Integrating Computational Tools in Interactive and Visual Methods for Enhancing High-dimensional Data and Cluster Analysis

Cagatay Turkay

Dissertation for the degree of Philosophiae Doctor (PhD)

Supervised by Helwig Hauser
Co-supervised by Peter Filzmoser

Institute for Informatics
University of Bergen
Norway

November 2013

To Burcu,
to my family,
to my friends,
and to all the nice people
who resist for
our rights in Turkey

Scientific Environment

The research carried out in this dissertation has been conducted in the Visualization group at the Department of Informatics, University of Bergen, Norway. My research have been facilitated with collaboration possibilities through the MedViz network in Bergen. Parts of my research has been conducted during my research stay at School of Engineering & Applied Sciences in Harvard University.

UNIVERSITY OF BERGEN

UNIVERSITAS BERGENSIS

ICT

Research School In
Information and Communication Technology

MEDVIZ FROM VISION TO DECISION

Acknowledgments

I would like to start by thanking my supervisor, Helwig Hauser, for the very fruitful, educative, and enjoyable four years during my PhD in Bergen. Helwig has always been there for me when I needed to have discussions and advise related to my research and other topics. I also thank Helwig for the nice trips, hikes, and very interesting philosophical discussions we have done together. I have learned a great deal from him to become a researcher in visualization. I am also very grateful to my co-supervisor, Peter Filzmoser. He provided me great insight on topics related to statistics and we had the chance to have very educating discussions on various topics in statistical data analysis.

I would like to specially thank Julius Parulek for the very productive collaboration we had over the years. I have been very lucky to dive into the research on interactive visual analysis together with Julius and I think it has been a pleasure for both of us to work together. I also thank Ivan Viola for making it possible for me to work on very interesting projects as the emergency statistician. I also thank him for his input and guidance in other projects we have done together. I would also like to express my gratitude to the excellent scientists that I had the chance to collaborate with from domains outside visualization. Many thanks to Nathalie Reuter, Arvid Lundervold and Astri Lundervold for their interest and support in our research. Along this line, I would like to express that I am grateful that I have the chance to work closely to the MedViz network in Bergen. I would also like to thank Hanspeter Pfister and Alexander Lex for making my research stay in Harvard University possible. I also thank the whole Caleydo team, Alexander Lex, Marc Streit, and Nils Gehlenborg for the very good collaboration we have started. I am also grateful for the mind-opening scientific discussions we had with Meister Eduard Gröller.

During my PhD project, I also had the chance to work closely and prepare submissions with the other members of the Visualization group. Special thanks to Veronika Šoltészová, Paolo Angelelli and Åsmund Birkeland for the productive collaboration. Many thanks to all the past and current members of the VisGroup for all the discussions and inputs I have got from all of you. It has been a great privilege for me to work with extremely talented colleagues.

I thank the VisGroup in Bergen for also making our VisCorridor a fun place to work in. I thank Andrea Brambilla for the discussions on beer making (and also for the beer he makes), Veronika for nice chats and hikes we did (and will be doing) together, Åsmund for making sure that we have enough grilled sausage in our blood, Mattia Natali for showing up every month at JazzBoks, Julius for the

several trips we made to the mountains and the bars together, Roger Bramon for the nice trip in the US, Johannes Kehrer and Ove Daae Lampe for introducing me to IVA in my first days, and Endre Lidal for the morning discussions over coffee. I like to extend my gratitudes to all the other members of VisGroup I have worked with, Paolo Angelleli Armin Pobitzer, Daniel Patel, Ola Kristoffer Øye, Anne-Kristin Stavrum, Linn Helljesen, Ivan Kolesar, and Pina Kingman. My special thanks also go to the administrative staff at the Department of Informatics at the University of Bergen for making our department a perfect place to work.

I thank my friends in Bergen outside the office for all the nice times we spent together. My thanks go to Michelle, Mike, Frederike, Joachim, Sonia, Anna-Caroline, Marc, Barbara, Maria, Ole, Maria Hauser, Umut, Tuba, Gard, and the amazing people from Nattjazz.

I want to send special thanks to all my friends in Turkey who I know that they are with me all the time: Yaser who is deeply missed, Muta, Seray, Serdar, Selcuk, Hasan Cem, Aziz, Cekdar, Burhan, Baris, Emrah, Duygu, Umut, Evrim, Aysan, Aylin, Mine, Tosun, Anna, Ramazan, Ulke, Kadir, Yelda, Julie, Uygar, Ilker, Onur, Seda, Yasemin, Ali Berk, Cihan, and others who I might be forgetting to mention here. You are my big family and I am lucky to have all of you in my life.

Special thanks go to my parents Nermin and Oktay, and my brother Koray Turkay for all their love and all they have done for me since I was a little kid. Many special thanks go to my newer family, Sehavet, Ahmet, and Bora Ceberler for all their love and support.

Finally I would like to thank my love Burcu for being together with me all the time and have gone through all the hurdles of my PhD period together with me. Without you none of this would be possible. I feel grateful for each day you are with me.

Abstract

With the advance of new data acquisition and generation technologies, our society is becoming increasingly information-driven. The datasets are getting larger and more complex as new technologies emerge and they are posing new challenges to the analysts who are trying to build an understanding of them. Automated computational approaches and interactive visual methods have been widely used to extract and interpret the relevant information in data analysis. However when these methods are used alone on complex datasets, their effectivity is limited due to several factors. Most of the commonly used computational tools often lead to hard to interpret results that may not be reliable most of the time.

This thesis aims to enhance data analysis procedures by integrating computational tools with interactive visual methodologies. The contributions of this thesis are mainly focused on the analysis of (very) high-dimensional data, i.e., hundreds and even thousands of dimensions, and cluster analysis. We introduce the dual analysis approach that makes it possible to analyze the items and the dimensions of a dataset in parallel in two linked visualization spaces. This methodology provides a basis to visually characterize and investigate dimensions as first-order analysis objects. We describe structure-aware analysis procedures that are facilitated by representative factors. Moreover, we present several mechanisms to achieve outlier-aware analysis routines. We describe the notion of *outlyingness* for the dimensions of a dataset and discuss how they can be determined and treated properly. We then focus on enhancing the dialogue between the analyst and the computer when computational methods are used interactively. We describe how different human factors come into play in visual analysis applications and propose optimized analytical processes that try to comply with the human capabilities. All these different approaches are demonstrated with various use-cases performed mostly together with experts from medical, genetic, and molecular biology domain.

Related Publications

This thesis is based on the following publications (see Part II of the thesis):

Paper A: C. Turkay, P. Filzmoser, and H. Hauser. **Brushing Dimensions-A Dual Visual Analysis Model for High-Dimensional Data**. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2591–2599, 2011.

Paper B: C. Turkay, A. Lundervold, A.J. Lundervold, and H. Hauser. **Representative Factor Generation for the Interactive Visual Analysis of High-Dimensional Data**. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2621–2630, 2012.

Paper C: C. Turkay, P. Angelelli, P. Filzmoser, and H. Hauser. **Outlier Dimensions – Outlier Aware Analysis of High-dimensional Data**. In submission to: *IEEE Transactions on Visualization and Computer Graphics*, 2013.

Paper D: C. Turkay and H. Hauser. **Optimizing Processes in Visual Analytics to Meet the Three Human Time Constants**. In submission to: *Computers and Graphics*, 2013.

Paper E: C. Turkay, A. Lundervold, A.J. Lundervold, and H. Hauser. **Hypothesis Generation by Interactive Visual Exploration of Heterogeneous Medical Data**. *Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data. Lecture Notes in Computer Science*, Volume 7947:1–12, 2013.

Paper F: C. Turkay, A. Lex, M. Streit, H.P. Pfister, and H. Hauser. **Characterizing Cancer Subtypes using the Dual Analysis Approach in Caleydo**. In submission to: *IEEE Computer Graphics and Applications*, 2013.

Paper G: C. Turkay, J. Parulek, N. Reuter, and H. Hauser. **Interactive Visual Analysis of Temporal Cluster Structures**. *Computer Graphics Forum*, 30(3):711–720, 2011.

The following publications are also related to this thesis:

Paper 1: C. Turkay, J. Parulek, and H. Hauser. **Dual analysis of DNA microarrays**. *Proceedings of the 12th International Conference on Knowledge Management and Knowledge Technologies, i-KNOW '12*, 26:1–26:8, 2012.

Paper 2: C. Turkay, J. Parulek, N. Reuter, and H. Hauser. **Integrating cluster formation and cluster evaluation in interactive visual analysis**. *Proceedings of the Spring Conference on Computer Graphics*, 77–86, 2011.

Paper 3: J. Parulek, C. Turkay, N. Reuter, and I. Viola. **Visual Cavity Analysis in Molecular Simulations**. To appear in: *BMC Bioinformatics*, 2013.

Paper 4: J. Parulek, C. Turkay, N. Reuter, and I. Viola. **Implicit surfaces for interactive graph based cavity analysis of molecular simulations**. *Proceedings of IEEE Symposium on Biological Data Visualization (BioVis 2012)*, 115–122, 2012.

Paper 5: V. Solteszova, C. Turkay, M.C. Price, and I. Viola. **A Perceptual-Statistics Shading Model**. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2265–2274, 2012.

Paper 6: A. Birkeland, C. Turkay, and I. Viola. **Perceptually Uniform Motion Space**. In submission to: *IEEE Transactions on Visualization and Computer Graphics*, 2013.

Paper 7: P. Angelelli, S. Oeltze, J. Haasz, C. Turkay, E. Hodneland, A. Lundervold, A.J. Lundervold, B. Preim, and H. Hauser. **Interactive Visual Analysis of Heterogeneous Cohort Study Data**. In submission to: *IEEE Computer Graphics and Applications*, 2013.

All the papers listed here have been written during the PhD period of the thesis author. The thesis author is the main author of the **Paper A** to **G**, **1** and **2**. **Paper A** to **G**, **1** and **2**, and **7** are co-authored by the main supervisor of the thesis, *Helwig Hauser*. Hauser contributed with ideas and inspiration in addition to the guidance and supervision.

Paper A and **Paper C** is written in collaboration with the co-supervisor of this thesis, *Peter Filzmoser* from the Department of Statistics and Probability Theory, Vienna University of Technology, Vienna. Filzmoser contributed with guidance on the statistical foundations of these articles. **Paper B** is coauthored by *Astri Johansen Lundervold* from the Department of Biological and Medical Psychology in University of Bergen, and by *Arvid Lundervold*, from the Department of Biomedicine in University of Bergen. Both A. Lundervold and A.J. Lundervold provided us with challenging problems in their domain and participated in joint sessions where we evaluated the analysis processes and findings. **Paper E** is a detailed report on our joint analysis sessions on the same problem domain where A. Lundervold and A.J. Lundervold provided us insight for generating and evaluating hypotheses within the data.

Paper C is co-authored by also *Paolo Angelelli*, a post-doctoral researcher from the Visualization Group in Bergen. Angelelli contributed to the paper by helping with the organization of the data and managing the communication with our collaboration partners.

Paper D has been submitted to the Computers and Graphics journal and currently it is under major revision. The included version of this paper is the state before the suggested revisions are done.

Paper F is the product of the research stay of the main thesis author in the School of Engineering & Applied Sciences in Harvard University. The paper is co-authored by *Alexander Lex*, a post-doctoral researcher from the VCG group at Harvard University, *Marc Streit* from Johannes Kepler University of Linz, and *Hanspeter Pfister* from Harvard University. Lex provided guidance regarding the technical aspects of the implementation. Both Lex and Streit contributed with discussions during the project development and also participated in the writing of the article. Both Pfister and Hauser provided supervision throughout the project.

Paper G, and **Paper 1, 2, 3, 4** are co-authored by Julius Parulek, a post-doctoral researcher from the Visualization Group in Bergen. In **Paper G** and **Paper 1, 2**, Parulek helped with the implementation of certain parts of the framework and contributed with discussions and in the write up of the paper.

In **Paper G**, and **Paper 2, 3**, I collaborated with Nathalie Reuter, a researcher and group leader from Department of Molecular Biology in University of Bergen. Reuter introduced several analytical problems from her domain and provided guidance and feedback on how to approach these problems with interactive visual methods.

In **Paper 3** and **4** I contributed with the development of the interactive mechanisms for the analysis of accompanying data in molecular dynamics. In these papers, I also actively took part in the writing of the paper.

In **Paper 5**, I contributed with the statistical modeling and the related analysis on the data from a user study on shape perception . Similarly in **Paper 6**, I helped with the statistical analysis of data related to a user study on motion perception. In **Paper 7**, I contributed with input and implementation on the interactive visual analysis functionality in the suggested framework.

Contents

II Papers 61

Part I

Overview

Chapter 1

Introduction

Due to recent advances in computing power and data acquisition methods, we are now living in an information-empowered society where the analysis of complex datasets are becoming increasingly important. One perspective on complexity is the growing size of the datasets in terms of number of the entities, i.e., rows of the data. This “big data” challenge is frequently investigated by researchers in visualization, data mining, and machine learning. However, there are other perspectives, those that are not addressed very often, which add to this complexity. This other form of complexity often stems from the fact that the data is collected/generated through several channels each of which carries different characteristics. In several domains of science, engineering, and business, such challenging datasets are becoming abundant. Analysts often refer to either automated computational methods or visualization techniques to explore and dig out the information in their data. While automated methods rely on the computational capabilities of the computer, visual analysis methods exploit the perceptual and cognitive strengths of humans in detecting structures and making associations. The successful analysis of the increasingly complex and heterogeneous datasets, on the other hand, calls for a *tight integration* of both of these methodologies [187].

The integration of capabilities of humans and computers has been one of the primary goals of the field of visual analytics [186, 113]. One common analysis pattern in visual analytics (VA) is the “Analyse first, show the important, zoom/filter, details on demand” mantra by Keim et al. [111]. This approach initiates the process by computational analysis, provides interactive support to investigate the important findings and then digs deeper into the data as the user sees fit. The research in VA brings together methods from visualization, data mining, data management, human-computer interaction, and human perception and cognition to devise powerful approaches to extract relevant information from data [111]. Several solutions from VA have been utilized in fields such as engineering, physics, medicine, or finance to aid the analysis of the nowadays highly challenging datasets [113]. The success of VA applications demonstrates that the integration of computational power and the strengths of humans has a huge potential in developing powerful analysis methods.

1.1 Problem statement and challenges

The research in this thesis is motivated by a number of challenges and problems arising in the *explorative data analysis* processes involving *high-dimensional* data and *cluster analysis*. The primary focus of our work is related to the analysis of *high-dimensional* data. With high-dimensional data, we refer to datasets with a (very) large number of dimensions, such as hundreds and even thousands, and in the context of this thesis, dimensions are considered as a mixture of dependent and independent variables. The abundance of dimensions distinguishes high-dimensional datasets from multi-dimensional (-variate) datasets which consist of a couple of dozens of dimensions at maximum. This particularly large number of dimensions in high-dimensional data leads to several challenges which we cover later in this section.

Datasets that have a large number of dimensions are becoming increasingly common in many application fields. One prominent field is *biology*, where high-throughput studies are producing data at different scales (from genetic sequencing to anatomical imaging) of the same samples [214]. For instance, the datasets to study the activity levels of genes often consist of measurements related to thousands of genes for a single sample [114]. In the field of *medicine*, large scale cohort studies involve the imaging of the participants using several modalities, such as magnetic-resonance or diffusion tensor imaging, complemented with a variety of clinical data on the patients. Other fields that deal with high-dimensional datasets include spectral imaging studies [71], large scale socio-economic surveys, or consumer activity data in business intelligence related analyses.

Most of the current computational and visual analysis approaches are tailored for multi-dimensional datasets and they easily fail to provide successful results when they are confronted with really high-dimensional data [2]. There are a number of factors that contribute to this limitation of the current approaches: *reliability and interpretability* of analysis results, the inherent *heterogeneity* within the dimensions, the *underlying assumptions* of computational tools, and no means to perform *local analysis* and merge the outcomes to build a big picture of the data. In the following, we discuss these observations and challenges in detail.

Reliability and interpretability: Both computational and visual methods do not scale with the large number of dimensions. In computational analysis, the results become hard to interpret and there are concerns about the reliability as the dimensionality of the data increases. Consider, for instance, the clustering of a 500-dimensional dataset (a 2D data table with 500 columns) using the popular K-means algorithm [181]. It is not straightforward at all to correctly interpret the resulting clusters when the computations are done on a 500-dimensional space, neither is it possible to judge the reliability of the clusters when the distances between the items are computed by a 500-dimensional distance metric [118]. This issue with distance measures is known as the "curse of dimensionality" that states the fact that distances between items lose their meaning in truly high-dimensional spaces [45]. On top of this, the number of samples could possibly be low in many cases. This results in datasets with small number of observations (small n) but a very high number of variables (large p). Since most of the statistical methods need a sufficiently large number of observations to provide reliable estimates, such "wide" data matrices lead to problematic computations [29].

In visualization, on the other hand, most of the methods that are widely used in the visual analysis of multivariate data, such as scatterplot matrices, parallel coordinates, or linked multiple views, can not successfully handle a large number of dimensions mainly due to the large physical screen space required to visualize the results, e.g., consider visualizing a 500 dimensional dataset where each dimension is an axis of a parallel coordinate plot. Although there has been significant research focusing on the scalability of visualizations in terms of data items that are visualized, truly high-dimensional datasets remain to be a challenge for most of the visual analysis approaches.

In order to address these issues listed above, there is the need to develop *methods that can easily cope with the high-dimensionality of the data.* Carefully designed interactive visual methodologies can guide users to give "informed" decisions while using computational analysis approaches.

Heterogeneity: The heterogeneous character of the set of dimensions is a challenge for both computational and visual analysis approaches. There are several causes of this heterogeneity. Dimensions can have difficult-to-relate *scales of measure*, such as categorical, discrete or continuous. Some can be replicates of other dimensions or encode exactly the same information acquired using a different method. There can be explicit relations between the dimensions that are known a priori by the expert. And there are usually inherent structures between the dimensions that could be discovered with the help of computational and visual analysis, e.g., correlation relations or common distributions types. Standard methods from data mining or statistics do not consider any known heterogeneity within the space of dimensions which could lead to results with limited quality. In order to achieve "successful" analysis sessions, *methods that enable an analyst to investigate the heterogeneous nature of high-dimensional datasets* should be developed.

Underlying assumptions: Most of the computational methods make assumptions on the structure of the data. Popular Multivariate analysis (MVA) methods such as PCA or regression analysis, for instance, assume that the data are normally distributed, or the variance is equal over all the data, known as the *assumption of homoscedasticity* [95]. Most of the methods also assume that the data is *clean of errors, missing values, and outliers.* The quality and reliability of the analysis relies heavily on whether such assumptions are met in the data. However, in real world cases, it is not often that such assumptions are met. Therefore there is the need for methods to check and validate whether the data conforms to such considerations. Moreover, it is also highly important to consider several methods/measures while performing the analysis to increase the reliability. For instance, when using descriptive statistics analyses can also incorporate *robust statistics* and methods that are resistant against outliers and problems in the data [59]. Along the lines of these issues, there is a need to devise *methods to enable analytical procedures that are aware of the different considerations related to the data and that can handle these properly.*

Local analysis: Due to the limitations of computational approaches, analysts have to perform their analysis on a subset of the data and thus losing the overall picture and having problems to relate the sub-analysis they carry out. On the other side, if the user decides to use the whole data for analytical operations, interpreting the results become a big challenge, i.e., applying dimension reduction on a 500-dimensional dataset. At this point, *mechanisms that enable analysts to merge the results of several local analyses performed on subsets of the data* can improve the analysis quality considerably.

In addition to the challenges related to high-dimensional data analysis, this thesis also focuses on problems related to *cluster analysis.* Cluster analysis divides data into groups (clusters) where data items within a group are similar with respect to certain criteria [181]. This analysis is one of the fundamental tasks in many data analysis scenarios and used widely in several domains [100]. Due to the variety of clustering algorithms and due to the fact that the notion of a cluster varies greatly from domain to domain, the *evaluation of clusters* is an essential step that needs to accompany cluster creation. Since the evaluation of clusters depends mainly on the expertise of the analyst, interactive visual methods can provide mechanisms to support this task.

Moreover, when the clustering of time series (temporal) data is considered, the above mentioned issues are even more critical. We observe that most of the algorithms developed for this task are either modifications of the static data clustering algorithms, or time-series are converted into static representations so that the existing algorithms can be used [125]. As a consequence, current methods are highly limited to properly aid the interpretation and evaluation of clusters of temporal data. There is a pressing need to develop *techniques that communicate the information in such temporal clusters and enable a comparative analysis of several of these structures.*

1.2 Contributions

The aforementioned problems and limitations in the current data analysis approaches motivate us to carry out the research in this thesis. With our contributions, we enhance the procedures involving high-dimensional data and cluster analysis. This is accomplished with a number of interactive and visual methodologies that make the *informed use* of computational tools possible throughout the interactive visual data analysis process. The contributions of this thesis can be investigated under a number of categories.

1. In order to consider the structured, heterogeneous nature of high-dimensional datasets, we proposed the *dual analysis approach* for the interactive visual analysis of very high-dimensional data. This method enables the simultaneous and linked visual analysis of both the dimensions and the items of a dataset. This methodology extends the domain of multiple linked views with visualizations that have the dimensions of a dataset as their main visual entities. This novel approach to visualize the dimensions enables the analyst to investigate the different characteristics of dimensions through the use of statistics and computational measures. Moreover, the proposed duality in interacting with both the data items and the dimensions leads to analyses that provide deeper insight on the relations between the items and the dimensions.
2. A method to enable the *structure-aware* analysis of high-dimensional datasets is proposed. We introduce the interactive visual exploration and creation of *representative factors* as a method to consider the structures in high-dimensional data analysis. This approach involves the creation of representative factors, each of which represents a sub-group of the dimensions. These representative factors are then analyzed together with the original dimensions through the same visualizations to understand the relations between the structures and the dimensions. We present a number of methods to create, to represent, and to evaluate the representative factors. These mechanisms provide the means to locally use computational tools and to visually compare and evaluate their results.
3. We present how an *outlier-aware analysis* of high-dimensional data can be carried out. With this work, we focus on the dimensions that carry "special" properties and thus *stand-out* from the rest of the dimensions. We describe the notion of *outlyingness for dimensions* through an according categorization. The proposed outlier-aware analysis process outlines how to characterize, how to determine, and how to treat outlier dimensions in high-dimensional data analysis. We demonstrate how important it is to consider the outlyingness of dimensions to achieve more reliable and insightful analyses.
4. We present a methodology that moderates the temporal aspects of the interactive visual steering of computational analysis tools. This moderation is done with the guidance of *human time constants* that enables us to address the perceptual capabilities of humans. Complementary to the other contributions of this thesis which focus more on improving the way computational tools are used interactively, this work focuses more on opti-

mizing how computational tools operate to conform to human capabilities. Our approach is realized through novel mechanisms such as the utilization of *online algorithms* together with a suitable *sampling mechanism*, the *keyframed brushing* technique, and the use of perceptually optimized, *animated transitions*.

5. We devise methods to visually support cluster analysis, especially within the domain of temporal data. Our approach enables analysts to both *evaluate* and *interpret* the clusters that are produced within the cluster analysis process. We utilize interactive visualizations together with measures that provide insight on the quality of clusters. Even more specifically, we propose novel and interactive visualization techniques to analyze *clusters of temporal data*. These views visualize the structural quality of temporal cluster sets and provide visual summaries of structures over time.

Thesis Structure

This thesis is composed of two main parts. In the first part, an overview of the research carried out within the course of this thesis is given. The second part consists of seven papers where the contributions in the overview part is described in detail.

The remainder of this thesis is structured as follows: In Chapter 2, the related state of the art in interactive and visual methods for of high-dimensional data and cluster analysis is discussed. The above listed contributions are detailed in Chapter 3. Demonstrations of the proposed ideas and methods are presented in Chapter 4. We discuss about the lessons learned during the thesis and conclude with perspectives on future research in Chapter 5.

The second part of the thesis includes seven papers to detail on the contributions listed above. Paper A and Paper F provide the details of the first contribution above. Paper B, Paper C and Paper D details on the contributions 2, 3, and 4 respectively. Paper F and Paper G correspond to the details of the fifth contribution. Paper E and also Paper F discuss how our methods are used in different application fields.

Chapter 2

State of the art: Interactive Visual Analysis of High-dimensional Data and Clusters

This chapter discusses the state of the art in the interactive and visual methods developed for high-dimensional data and cluster analysis. We start with a discussion on the research related to using a combination of automated and interactive visual methods and investigate the related studies in two categories. We then move on to discuss the research in the visual analysis of high-dimensional data with also a focus on the consideration of local structures and outliers. Section 2.3 discusses how interactive visual methods support the cluster analysis process. We then present how the interactivity is maintained within visual analysis frameworks.

2.1 Integrating Visual and Computational Analysis

Understanding the underlying information in the challenging datasets of nowadays have been in the focus of several research fields. Studies in statistics [100], data mining [181], machine learning [6], and certainly in visualization [172] have devised methods to help analysts in extracting information from the data. While the first three fields rely on computational power, visualization relies mainly on the perceptual and cognitive capabilities of the human in extracting information. Although these research activities have followed separate paths, there have been significant studies to bring together the strengths from these fields [110, 174, 129]. Tukey [188] led the way in integrating visualization and statistics with his work on exploratory data analysis. Earlier research on integrating statistics [32] and data mining [110] with information visualization have taken Tukey's ideas further.

This vision of integrating the best of both worlds has been a highly praised goal in visualization research [187, 113, 18] and led to the emergence of *visual analytics* as a field on its own. Visual analytics brings together research from visualization, data mining, data management, and human computer interaction [113]. In visual analytics research, the integration of automated and interactive methods is considered to be the main mechanism to foster the construction of knowledge in data analysis. In that respect, Keim [111] describes the details of a visual analysis process, where the data, the visualization, hypotheses, and interactive methods are integrated to extract relevant information. In their sense-making loop, based on the model introduced by van Wijk [196], the analytical process is carried out iteratively where the computational results are investigated through interactive visualizations. Such a loop aims to provide a better understanding of the data that will ultimately help the analyst to build new hypotheses.

There are different surveys that characterize how the integration of automated methods and interactive visualizations are accomplished. Crouser and Chang [37] characterize the human computer collaboration by identifying what contributions are made to the process by the two sides. In their survey, several papers are grouped according to these types of contributions. According to the authors, humans contribute to the analytical processes mainly by *visual perception*, *visuospatial thinking*, *creativity* and *domain knowledge*. On the other side, the computer contributes by *data manipulation, collection and storing*, and *bias-free analysis routines*. Bertini and Lalanne [18] categorize the methods involving data mining and visualization into three: *computationally enhanced visualization*, *visually enhanced mining*, and *integrated visualization and mining*. Their categorization depends on whether it is the visualization or the automated method that plays the major role in the analysis. In the following, we employ a simplified categorization and discuss the related works in integrated methods depending on the way the computational tool is utilized within the analysis: *using automated method as a standalone tool* and *interactive visual steering of the computation*. Kehrer et al. [106] demonstrates how statistical moments can be utilized to construct and navigate between visualizations. Their approach is a demonstration of how statistical aggregates facilitate the analysis of multi-faceted datasets.

Automated methods as a standalone tool

In this type of integration, the computational tool is used as a separate entity either implicitly or explicitly (refer to Chapter 3 for a related discussion) within the analysis and its inner working is not transparent to the user. In this setting, the user interacts with the computational mechanism either through *modifying parameters* or *altering the data domain* that the method is applied on. The results are then presented to the user through different visual encodings that are often accompanied by interaction. There are several examples along the lines of visual analytics that utilize such an integration. Perer and Shneiderman [144] discuss the importance of combining computational analysis methods, in particular statistics, with visualization to improve exploratory data analysis. Their study on a group of experts reveals that without interactive visualization, computational results can become very hard to interpret. Jänicke et al. [94] utilize a two-dimensional projection method where the analysis is performed on a projected 2D space called the attribute cloud. The resulting point cloud is then used as the medium for interaction where the user is able to brush and link the selections to other views of the data. The use cases in this work also demonstrate that the analysis performed at a projected space leads to successful results. Johansson and Johansson [99] enable the user to interactively reduce the dimensionality of a dataset with the help of quality metrics. The visually guided variable ordering and filtering reduces the complexity of the data in a transparent manner where the user has a control over the whole process. The authors later use this methodology in the analysis of high-dimensional datasets involving microbial populations [57]. Ingram et al. [92] present a system called DimStiller, where there are a selection of data transformations that are chained together interactively to achieve dimension reduction. The presented framework treats the computational tools as operators that perform particular tasks on data tables. Fuchs et al. [66] integrate methods from machine learning with interactive visual analysis to assist the user in knowledge discovery. Oeltze et al. [141] demonstrate how statistical methods, such as correlation analysis and principal component analysis, are used interactively to assist the derivation of new features in the analysis of multivariate data. Correa et al. [36] consider the uncertainties that arise while transforming the data. These uncertainties are integrated in the visualization to support the interpretation of statistical analysis results. Guo et al. [76] enable the interactive exploration of multivariate model parameters. They visualize the model space together with the data to reveal the trends in the data. Gosink et al. [70] use a query-driven visualization with a statistics-based framework. They utilize query distributions to estimate trends and features.

Interactive visual steering of computations

This mode of integration constitutes of mechanisms where the analyst interacts with the inner working of the algorithms. This is often achieved by displaying intermediate results where the user provides guidance for the algorithm to carry the computations further.

Although not as common as the solutions in the first category, there are several methods that fall under this category. In a recent paper, Endret et al. [50] describe such methods as enabling the *direct manipulation* for visual analytics. They describe three levels for interaction to enable such an integration: the manipulation of *spatial constraints*, *parameter weights* and *model steering*. They suggest that such a multi-level interaction facilitates the symbiotic relation between the computer and the analyst.

In MDSteer [210], an embedding is guided with user interaction leading to an adapted multidimensional scaling of multivariate datasets. Such a mechanism enables the analyst to steer the computational resources accordingly to areas where more precision is needed. Endert et al. [51] introduce observation level interactions to assist computational analysis tools to deliver more reliable results. May and Kohlhammer [133] present a conceptual framework that improves the classification of data using decision trees in an interactive manner. The results are iteratively improved through user input. Nam and Mueller [136] provides the user with an interface where a high-dimensional projection method can be steered according to user input. In the analysis of streaming text data, Jamal et al. [7] proposes a system that incorporates user input within the computations on-the-fly.

2.2 Visual Analysis of High-dimensional Data

Multi-dimensional datasets, where the dimension count is a few to several dozens approximately, have been studied widely in the visual analysis literature. Surveys by Wong and Bergeron [211] and more recently Fuchs and Hauser [65] provide an overview of multivariate analysis methods in visualization. The recent survey by Kehrer and Hauser [107] covers a wider spectrum of research and discusses the visual analysis of multifaceted data.

Frameworks with multiple coordinated views, such as XmdvTool [202], Jigsaw [177] or Polaris [178], are used quite commonly by now in visual multivariate analysis. Weaver [203] presents a method to explore multidimensional datasets, where the analysis is carried out by cross-filtering data from different views.

In these multiple view systems, data is visualized through 2D scatterplots, scatterplot matrices, parallel coordinate views, or histograms. One commonly employed interaction mechanism is the *linking & brushing* [14], where the user selects (or *brushes*) a subset of the data through one of the views and the same selection is then highlighted in the other views using a visualization method called *focus + context* [83, 43].

Compared to all these important related works there are however only few studies published where really high-dimensional data are analyzed. One example is the VAR display by Yang et al. [212], where the authors represent the dimensions as glyphs on a 2D projection of the dimensions. A multidimensional scaling operation is performed on the glyphs where the distances between the dimensions are optimally preserved in the projection.

There are methods to reduce the set of dimensions with the help of visual analysis and measures to evaluate and reduce the possible visualization space. May et al. [132] proposed a technique called SmartStripes where they investigated the relations between different subsets of features and entities. Their method guides the user in selecting suitable subsets for the analysis. Tatu et al. [182], on the other side employ automated methods to rank visualizations of high-dimensional dataset. Their approach suggest users a good subset of the dimensions to start the visual analysis analysis process.

There are few other works where a duality within the analysis proved to be useful. In parameter space exploration [17], the authors used two interaction spaces, one for the parameters and the other for system output in the form of predictions. In temporal data analysis, Andrienko et al. [9] perform the analysis both on the spatial and the temporal domains. The analysis on two separate data domains is carried out in parallel by the introduction of interfaces in the case of multi-run simulation data [109].

Performing the high-dimensional data analysis on derived attributes is a strategy utilized in a number of studies. Kehrer et al. [106] integrate statistical moments and aggregates to interactively analyze collections of multivariate datasets. Wilkinson et al. introduced graph-theoretic scagnostics [208] to characterize the pairwise relations on multidimensional datasets. Scagnostics are powerful measures that quantify the relations in 2D scatterplots. These measures makes it possible for the user to reduce the visualization space considerably via filtering non-interesting scatterplots. In a later work [209], the same authors used these features to analyze the relations between the dimensions. Scagnostics measures are also utilized to analyze multi-variate temporal datasets [38].

2.2.1 Visual Analysis of Structures

The structure of high-dimensional datasets and the relations between the dimensions have been investigated in a few studies, also. Seo and Shneiderman devise a selection of statistics to explore the relations between the dimensions in their Rank-by-Feature framework [168]. They rank 1D or 2D visualizations according to statistical features to discover relations in the data. In their method, the main focus is on the data items. One very interesting work is the visual hierarchical dimension reduction method by Yang et al. [213]. The authors build a hierarchy of the dimensions that is than used to create representatives and construct lower-dimensional spaces. In a similar work, Huang et al. [90] utilized the derived dimensions together with the original dimensions. The authors observed the output of several dimension reduction methods with a special focus on how they correlate with certain characteristics of the original dimensions. In an related paper from the analytical chemistry field by Ivosev et al. [93], the authors group variables depending on their inter-correlations and utilize them in dimension reduction and visualization. Their method is applied only to principal component analysis, however it demonstrates the benefit from a strategy that groups variables together.

2.2.2 Visual Analysis of Outliers

Outliers have been in the focus of research in data mining and statistics fields [88]. However, there is a limited number of studies in visual analysis. One of the most important papers that specifically address outliers in visualization is by Novotný and Hauser [140] where they visually separate trends and outliers in their extended version of a parallel coordinate plot. The trends in the data is represented as context and the outliers are separated in the visualization. This work demonstrates how visual analysis can benefit from the special treatment of outliers. Another important study on outlier analysis is by Kehrer et al. [106], where the authors can identify outlying observations through the exploration of multi-faceted data via aggregated statistics. In a recent study, Kandogan [101] discusses how trends and outliers can be detected in his visualization level approach. His image-based technique reveals outliers in a 2D visualization where the method automatically annotates the findings to make them apparent to the analyst. Liao et al. [126] introduced a visually-guided active learning mechanism to detect anomalies in GPS datasets. In all of these studies, however, the focus of the methods is on observations rather than on the dimensions. We have not come across any study where the outlyingness of dimensions has been investigated.

2.3 Visually Supported Cluster Analysis

Interactive techniques have proven to aid analysts in refining and building clustering results. Sprenger et al. [176] introduced a visually supported hierarchical clustering algorithm. Their visual clustering approach involves a two-stage procedure – a hierarchical clustering is followed by a visualization that uses blob objects to reveal cluster shapes. Rinzivillo et al. [154] use a visual technique called progressive clustering where the clustering is done using different distance functions in consecutive steps. The progressive clustering technique provides a convenient mechanism where potentially interesting portions of data are selected to direct the algorithms. Schreck et al. [164] propose a framework to interactively monitor and control Kohonen maps to cluster trajectory data. The authors state the importance of integrating the expert within the clustering process in achieving good results. Fua et al. [64] propose a technique based on parallel coordinates, which displays the required level of detail on the dataset using hierarchical clustering results. Another method for structure discovery in large datasets by means of clustering results and parallel coordinates is presented by Johansson et al. [98]. The authors exploit clusterings and high-precision textures to enhance apparent structures in parallel coordinates thus avoiding the cluttering issue.

Visualization has generally served as the final step of cluster analysis where it plays a critical role in enhancing the interpretation of clusters by enabling comparison and evaluation. gCluto [148] is an interactive clustering and visualization system where the authors incorporate a wide range of clustering algorithms. This system enables the user to store different clusterings and visualize the results in a matrix or mountain visualization. Rubel et al. [159] introduce a framework called PointCloudXplore that integrates clustering and visualization for the analysis of a dataset that has a spatial mapping.

In *Hierarchical Clustering Explorer* [167], Seo and Shneiderman describe the use of an interactive dendogram coupled with a colored heatmap to represent clustering information within a coordinated multiple view system. The authors enable the comparison of two clusters through a specialized comparison view. Lex et al. introduce MatchMaker [123], where they visualize and compare multiple groups of dimensions. In their work, they provide a use-case where they use their methods to compare clusters. In the follow-up of this work, StratomeX [124], they demonstrate how such a visual encoding facilitates the analysis of cancer subtypes.

Sharko et al. employed projections of data items on a vectorized radial visualizations to investigate several clustering results. Their method helps analysts to validate particular results when several clusterings of the same dataset exist. Bezdek and Hathaway [19] developed an interactive dissimilarity matrix that is extended by Siirtola [175] to analyze clustering results at different similarity levels. Specialized heat maps called cluster stability matrices are utilized by Sharko et al. [170] to visually determine most 'stable' clusters in clustering results. In the MultiClusterTree [195], Long and Linsen discuss how clusterings are utilized to analyze multi-dimensional data. A radial layout that is linked with several other views are utilized to explore hierarchical clusters. In the software visualization domain, Telea and Auber [185] represent the changes in code structures using a flow layout where they identify steady code blocks and when splits occur in the code of a software.

Analyzing temporal clusters

Wijk and Selow [198] presented one of the earliest works on cluster-based visualization of temporal data. The authors cluster time-series data and visualize the results on a calendar. In a paper by Andrienko et al. [10], the authors discuss how they perform the interactive clustering of trajectory data and they present a user-driven clustering methodology. They use graphical summaries of trajectory clusters to indicate the number of cluster members. These summaries provide valuable information when the analyst is interested in changes of the cluster sizes.

Dynamically evolving clusters, in the domain of molecular dynamics, are analyzed through interactive visual tools by Grottel et al. [74]. The authors describe flow groups and a schematic view that display cluster evolution over time. These groups are observed to validate the quality of clustering results.

Self organizing maps (SOM) have been used to visualize the temporal cluster changes by Denny et al. [41]. The authors create a set of SOMs for different time instances over time and compare these set of maps to explore structural changes in the cluster sets. However, this solution is limited to depicting only cluster-cluster relations. Another work where self organizing maps are utilized is by Andrienko et al. [9]. They propose the interactive utilization of SOMs that are integrated in a visual analysis framework. Their solution aims to discover spatiotemporal relations by analyzing the temporal evolution of a spatial situation and the distribution of temporal changes sequentially.

2.4 Maintaining the pace of interactivity in integrated systems

The integration of computational methods within interactive systems brings the challenge to maintain the pace of interactivity at an acceptable level, i.e., the user should not wait a very long time for the computational results to be computed. According to Shneiderman [173], interactive mechanisms need to give immediate feedback to user inputs within certain temporal limitations. One mechanism to maintain the interactivity of such systems is to improve the performance of the computationally heavy tasks. Along this line, Chan et al. [27] made use of predictive caching to improve the interaction with massive time series data. Piringer et al. [146] present a multi-threading architecture where the visualization and the background operations are carried out in separated threads to ensure interactive response. Fekete and Plaisant [55] focus on improving the scalability of visualization methods by incorporating GPU-supported computations.

Rosenbaum and Schumann [156] suggest a progressive refinement framework in order to achieve a scalable system in terms of response times, visual clutter, and available resources. The authors discuss that developing specific solutions that employ progressive refinement approaches still remains as an open challenge. Ahmed and Weaver [3] present an interactive cluster exploration system. The authors display approximate clustering results to maintain smoothly running interactivity. Similarly, Fisher et al. [61] present how an incremental sampling strategy can be employed in a database query system. The authors also perform a user study with analysts where they find out that their incremental approach enables them to give certain decisions early and update/remove their queries without waiting for the results to complete. In a recent work, Choo and Park [31] discuss the challenges brought up by very large datasets along the same lines. The authors suggest methods on how the responsiveness of systems can be improved through using less precision, using iterative refinement for the representation of results.

Chapter 3

Integrating Computational Methods in Interactive Visual Analysis

Both automated and visual analysis methods have exactly the same goal: helping the analyst to *build a better understanding of the data.* Automated computational analysis tools achieve this by performing tasks such as summarizing information, quantifying relations, finding structures, and classifying elements in datasets. However, due to several factors introduced in Chapter 1, analytical procedures that utilize these automated approaches alone suffer from certain limitations and pitfalls. On the other side, visual analysis methods need the speed and precision of automated approaches to carry on the complicated tasks that are listed above. This thesis aims to join the strengths of both interactive visual and automated methods and focuses on the integration of computational tools in the interactive and visual analysis of data to help analysts in gaining insight into complex datasets.

In this thesis, the integration of computational methods is achieved at two different levels by utilizing their output either *explicitly* or *implicitly.* We refer to the set of available automated methods as the *computational toolbox.* In the explicit use of computational tools, the output of the tool is treated as an extension of the raw data and is subject to the interactive visual analysis process together with the actual data. An example of such an integration is applying dimension reduction to project a high-dimensional dataset to a 2D space and visualizing the result within the visual analysis. The implicit use of computational tools on the other side, involves the use of computed measures to enhance interactive and visual methods. This approach uses the computational output inherently rather than making it explicitly available for the analysis. An example of such an implicit use can be coloring the data points in a scatterplot according to a computed measure, f.i., how central they are in the data distribution.

This approach to utilize computational tools at two levels facilitates our goal to tightly integrate automated methods with interactive visualizations. Figure 3.1 provides an overview of these two levels. Notice that, the interactive visual methods together with the implicitly used computational tools operate on the set of raw and derived data, i.e., *the data domain.* This data domain is extended iteratively with the use of computational tools explicitly. We follow a strategy

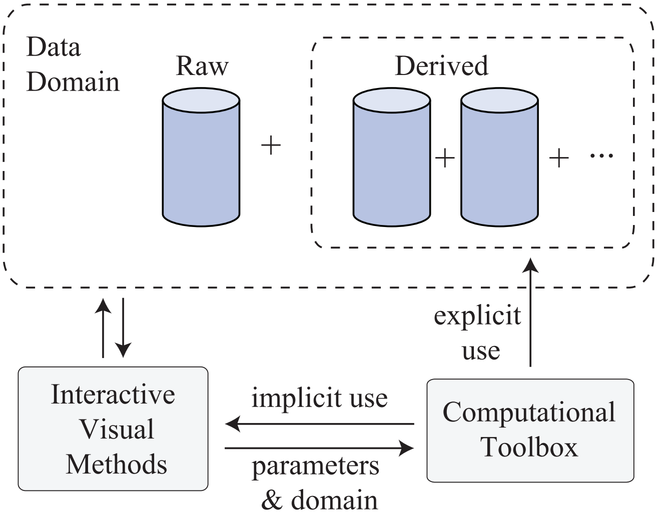


Figure 3.1: Integrating computational tools and interactive visual methods. One mechanism to use computational tools is to do it explicitly and extend the data domain with their output for further analysis. The implicit use of the computational toolbox enhances the interactive visual analysis approaches. The interactive visualizations are also used to determine specifications for the automated methods, such as interactively determining the data domain or parameters.

to make the output of several runs of computational tools available throughout the analysis together with the raw data. Our two-level approach enables the analyst to observe and interact with the results of particular computational tools in relation to the actual data. One important aspect to pinpoint here is that the explicit use of computational tools is supported by the feedback provided from the interactive visual analysis cycle. All the interaction in this cycle is also enhanced with the implicit use of specific computational methods. This mechanism in turn enables the *informed* and *reliable* use of computational tools.

This iterative loop that is facilitated by the integration of computational tools and interactive visual methods leads to an *enhanced data analysis processes.* In the following we discuss how this integration enhances the analysis of high-dimensional data and cluster analysis and we answer the question: *How does the integration of computational tools with interactive and visual methods lead to enhanced data analysis procedures?*

Here, we list the major points where the data analysis process benefits from the integration of automated approaches with interactive and visual methods.

- The *visual characterization* of the space of the data dimensions through *statistics* and *computed measures.* This enables an analyst to make *in-*

formed decisions in each step of the data analysis process involving high-dimensional data. These decisions involve initial data analysis steps from how the data is pre-processed, e.g., normalized, to core analysis steps on how a computational method is used most accurately, i.e., by checking for assumptions on data.

- The visual analysis of the *heterogeneous nature* of high-dimensional data spaces. This facilitates processes that are aware of local structures and outliers. This in turn improves the reliability and the interpretability of the analyses.
- The ability to *compare several results* from one or more computational tools. Since different automated methods have various strengths and drawbacks, resorting to several algorithms and comparing their results improves the confidence of the analyst on the resulting findings.
- The visual *communication of the quality* and the *certainty* of computational results. This provides the analyst to evaluate the findings and refine the process accordingly.

In the remaining of this chapter, we introduce our approach and give the details on how the above listed enhancements are accomplished. In Section 3.1, we introduce our interactive and visual methodologies for the analysis of high-dimensional data. We then continue to discuss visualization methods that improve the cluster analysis process in Section 3.2. Section 3.3 focuses on the human side of the use of integrated automated methods and discusses the importance of human factors in interactive visual analysis processes.

3.1 Interactive Visual Analysis of High-dimensional Data

In this thesis, we have a particular focus on high-dimensional datasets. As discussed earlier, when we mention high-dimensional data, we refer to datasets with hundreds or thousands of dimensions. In order to be able to analyze such datasets, we introduce the *Dual Analysis Methodology* that enables us to characterize and visually analyze the dimensions (Section 3.1.1). This methodology makes the heterogeneity within the set of dimensions accessible to the analysts. Building upon this mechanism, we extend our interest to the local (in Section 3.1.2) and “special” (in Section 3.1.3) structures. We provide analysis procedures that put special emphasis on such properties of high-dimensional spaces.

3.1.1 Dual Analysis Approach: Analyzing dimensions as first-order objects

In the course of this thesis, we consider high-dimensional data in a tabular form where items are the rows and dimensions are the columns. The conventional visual analysis of such data is to employ multiple coordinated views, where the data items are represented through visualizations such as scatterplots, histograms or parallel coordinates. Almost all the time, the data items are plotted in such views as opposed to the dimensions of the data, e.g., a scatterplot where the axes are two dimensions of the data and a point corresponds to a data item. The visual analysis of data items is often carried out using interactive mechanism such as *linking&brushing* and *focus+context visualization*. Here, we present a visual analysis model where the analysis of items and dimensions is carried out in two linked spaces, namely the *items space* and the *dimensions space*. We utilize the current knowledge about the interactive visual analysis of data items to also enable the interactive visual analysis of data dimensions. In our model, we suggest a setting of linked views, where the analyst interacts with the items in *items space*, e.g., *by brushing items*, and with the dimensions in *dimensions space*, f.i., *by brushing dimensions*.

As illustrated in Figure 3.2, the visual analysis space is structured into two spaces: *items space* I, and in *dimensions space* D. With items space we refer to a visualization domain where each visual entity in a visualization corresponds to a data item. In the dimensions space, however, each visual entity represents a dimension of the data.

To illustrate how we construct the views in both of these spaces to enable the dual analysis, consider a 2D table with n items (rows) and p dimensions (columns). In order to be able to construct visualizations of dimensions, we do the following: For each dimension, we derive a feature vector whose values are either selected statistics or derived information computed using the original data. In other words, we derive a $p \times k$ table S by assigning k values to each dimension. Once we construct this statistics table S, we use its values to build visualizations in the dimensions space. In Figure 3.3, the data is presented as a 2D table. Here, a conventional scatterplot visualizes the data items (each point is an item) over the values of two of the dimensions d_0 and d_1. Note that items space visualizations are characterized by a blue background. In order to construct a scatterplot of dimensions, we choose in this example to utilize the first two statistical moments μ and σ. For each dimension, we compute the μ and the σ of the values in the corresponding column. This gives us two values per dimension, which we then visualize on a scatterplot (each point is a dimensions) with a yellow background.

Such a visualization provides an overview of the characteristics of dimensions. For instance, the dimensions that are placed to the lower right corner have often higher values but at the same time show very small variety. And there is a single dimension that has a large variety in its values that makes it stand-out from the

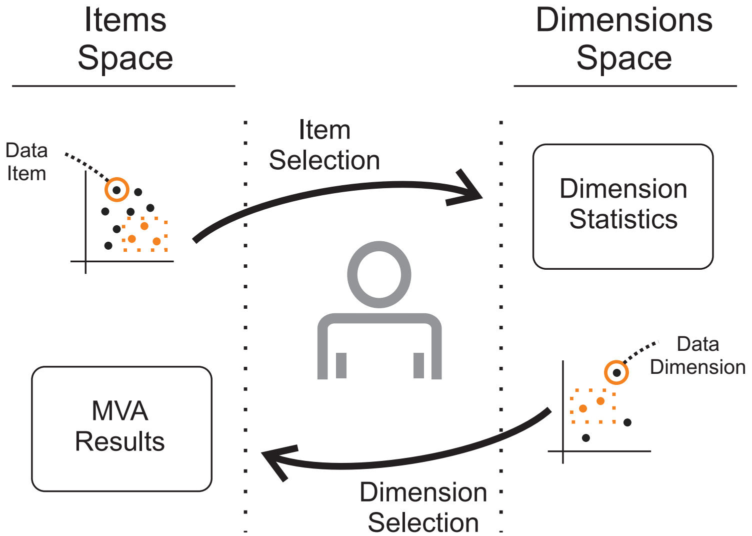


Figure 3.2: Visual analysis is performed over two spaces, items space and dimensions space. Visual entities correspond to items in items space and dimensions in dimensions space. Analysis advances iteratively by selecting items and dimensions. The interactions enable the joint and linked exploration of dimension statistics and multivariate analysis (MVA) results.

rest of the dimensions. The observations obtained from such visualizations get richer as we include more statistics into the analysis and interact with different visualizations of dimensions. The variety of insight provided by different statistics is discussed in the *Statistical and Computational Toolbox* section, Chapter 4 and in Papers A, B and C.

Using Computational Means Interactively within Dual Analysis Approach

The duality facilitated by our approach opens up for new possibilities to use computational tools both explicitly and implicitly in an interactive manner as introduced earlier in this chapter. The user can interact with both the items and the dimensions space visualizations. The brushing & linking mechanism across views from the different spaces enables the interactive use of computational tools in the analysis.

Interactive calculation of statistics – In order to link the selections in items space to the dimensions space, we introduce a view called *the difference view*. This view responds to a selection of items by recalculating the statistics/features and displaying the changes in the values. This provides an interactive mechanism to trigger statistical calculations on the data and assess the results instantly.

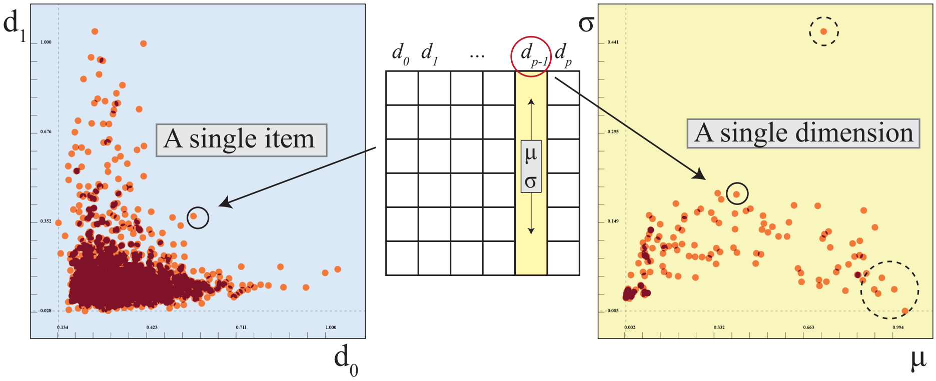


Figure 3.3: Setting up dual analysis views where the data is depicted as a 2D table for illustration. In an items space scatterplot (with a blue background), two dimensions are selected as the axes and each point is a data item. In a dimension space scatterplot (yellow background), each point is a dimension. The values for a single dimension are the μ and σ values computed over a single column.

In Figure 3.4-right, we see a difference view that displays the changes in μ and σ values. The user first selects (brushes) a subset of items and we denote the set of selected items as B. In response, the system automatically calculates the μ and σ values for each dimension using only the set of selected items B (μ^B and σ^B). At this point, we provide two options to build the difference view – two different *context subsets* to compare to. In the first option, the user compares μ^B and σ^B values to μ and σ values computed over *all the items*. As a second alternative, we compare μ^B and σ^B values to statistics computed using *the rest of items*. We denote the items that we compare against with C and the values computed for this context with μ^C and σ^C. We then compute the differences between the values with:

$$\Delta_\mu = \mu^B - \mu^C \quad , \quad \Delta_\sigma = \sigma^B - \sigma^C \tag{3.1}$$

Note that Δ_μ and Δ_σ are both data vectors of size p, the number of dimensions. When there is no difference for the values of a dimension for subsets B and C, it is placed at the origin $(0, 0)$ of the view. Similarly, for the dimensions marked in Figure 3.4-right, the dimensions have larger values and higher variance for the selection B.

The reason to enable two options for the context C is to provide suitable comparative tests for different tasks. For most of the instances of the difference view, we used the whole data to be the context. However, in specific cases where overlapping samples are not preferred, such as comparing statistics over different clusters, we choose to use the rest of the data items as the context. For instance,

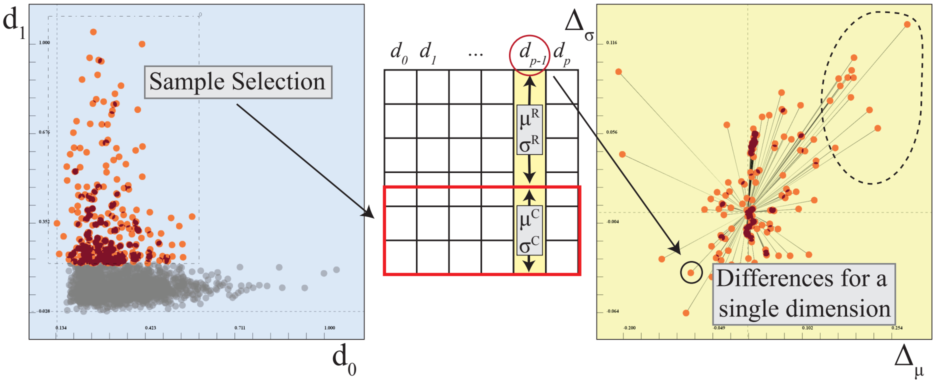


Figure 3.4: The difference view displays the changes in statistical computations after a selection is made. Here, we select the items with high d_1 values. μ and σ values are computed for each dimension twice: using only the selection and using the rest the data (or all, depending on the task). The differences between the two sets of values are computed and visualized. The dimensions on the right upper corner are those that have both higher values and higher variety for the selected items.

in the demonstration cases where we compare clusters using difference views in Section 4.2, we use the rest of the items as C.

One very important consideration when differences between two subsets are analyzed is the notion of *statistical significance*, i.e., whether the difference occurs by chance or not. A variety of *statistical hypothesis tests* are often employed to evaluate the significance of the differences between two groups of items, especially in terms of their central tendency, i.e., mean (or median). Since the comparison of means is one of the common tasks that is performed in several domains and different types of analysis, we enhance our difference view with the *implicit* use of the statistical hypothesis testing and introduce the *significant difference view*. In order to compute the significance, we utilize the *two-sample Welch's t-test* as the integrated hypothesis testing procedure [161]. We choose this test since it does not assume that the two subsets have equal variance, which makes it more suitable for our application. We perform the statistical test on the two subsets B and C (as introduced above), and test against the (*null*) hypothesis that these two subsets have equal central tendencies. This test is performed for each of the dimensions – showing whether there is a significant difference between the values of the two groups of items for a particular dimension. Each dimension is then colored accordingly. Dimensions that have significant differences are colored red, while the others are shown in blue. This addition to the difference view enables analysts to get immediate feedback on the significance of differences. This view is one of the examples where a computational tool, i.e., hypothesis testing, is utilized *implicitly* in a visualization to enhance the analysis process.

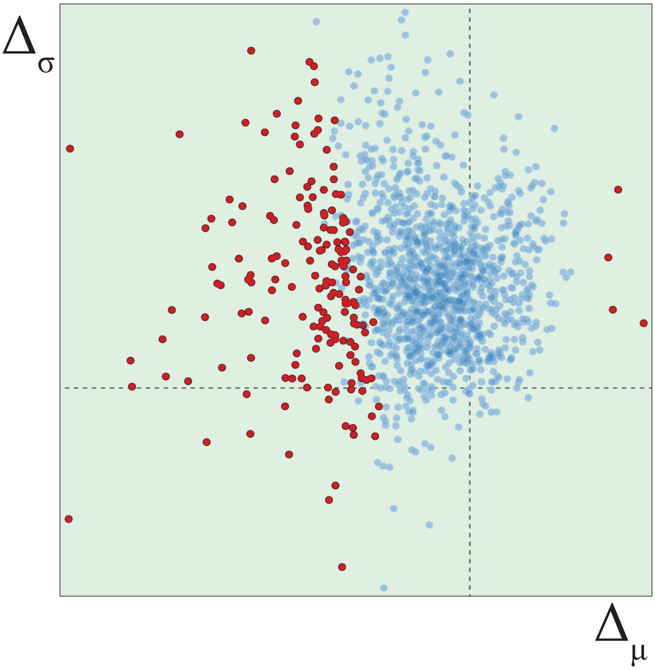


Figure 3.5: The significance of the differences between the μ values for two groups of items (not shown here) are depicted in the significant difference view. The dimension is highlighted with red color if the difference is significant and with blue otherwise.

Interactive use of computational analysis tools – The linking of the selections performed in the dimensions space to views in items space enables the user to interactively use computational analysis methods on high-dimensional datasets. This is achieved by the integrated use of several dimension reduction and clustering algorithms operating only on the dimensions selected by the user through the visualizations.

One of the examples of computational tools that we commonly use is principal component analysis (PCA). It provides a representation of the data in a lower dimensional space (often computed as a 2D projection). The result of PCA is then presented as a scatterplot of data items. In order to make PCA a part of interactive processes, the following steps are taken: The user makes a selection of dimensions (through a dimension view), the PCA is computed automatically using only the selected dimensions, and the resulting projection of the data items is automatically updated with the new results. Figure 3.6 illustrates how such computations are performed. We first bring up a dimensions view of μ against σ values and a scatterplot of items that displays the first two principal components

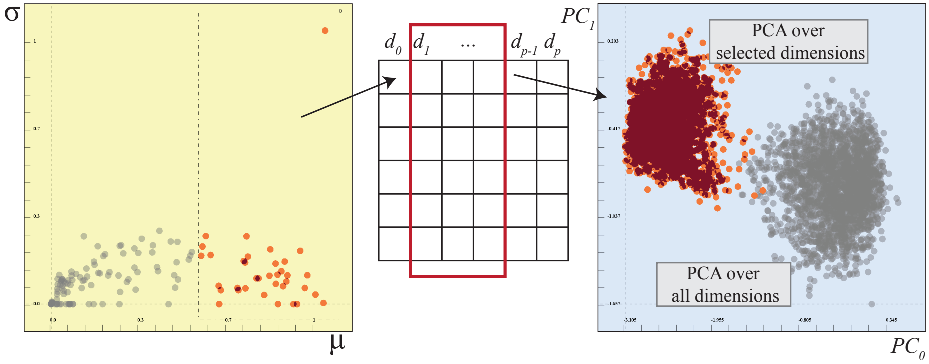


Figure 3.6: A μ vs. σ visualization of the dimensions and the data items on the first two principal components of a PCA computation using all the dimensions in the data (right). The system automatically responds to a selection of dimensions and re-applies PCA using only the selected dimensions. Both of the projections are visualized together in a single view. The new result (PCA on only the selected) is displayed in red while the rest (PCA on all the dimensions) is displayed in gray.

(PC) of PCA applied on all the dimensions (the initial state of this view is not shown in the figure). We start with a selection of the dimensions with higher average values. The system automatically applies PCA on the selection, projects the items to the newly computed PCs, and visualizes the result. The new results (shown in red color) are presented together with the previous computation (computed using all the dimensions) results (shown in gray).

In Figure 3.6, we display both of the results in a single plot to ease the comparison of different computations. However, we implemented another strategy that makes us of *animated transitions*. In this setting, the system responds to user inputs by performing the computations in the background and animating the data items from one result to the other. The details of these animated transitions are given in Paper D. In addition to PCA, we integrate a number of computational tools that can be utilized in the same manner as described above. Next section discusses these various tools.

This interactive mechanism is our main routine to utilize automated methods *explicitly*. Analysts are able to save any intermediate result for further investigation, i.e., the result of the computations are not lost as the user moves the selection. By saving the results, the user extends the data domain with these derived data columns and make them an integral part of the analysis.

Statistical and Computational Toolbox

The richness and the success of the analyses carried out by using the dual analysis approach depends on the variety of measures and statistics that are utilized to analyze the dimensions. We determine a number of measures (statistics/derived) that are important for different types of analysis we carry out in this thesis. We group the measures according to the type of information they provide and organize them in four categories: *characteristics of dimensions*, *summary of the distributions*, *type of the underlying model* and, *uniqueness of dimensions.* One important point to mention is that we also consider the robust versions of statistics. The field of robust statistics aims at statistical estimates and methods that are more resistant to outliers [59].

The first category of measures relate to the inherent characteristics of dimensions, such as the scales of measure (represented by the count of unique values in a column, *uniq*) or the percentage of 1D outliers $\%out$. The second class of measures provides insight on the shape of the distribution of values through summary statistics and their robust counterparts. The statistics in this category include, first of all, the basic statistical moments to measure centrality, i.e., the mean μ and the median *med*, and different measures of variability such as standard deviation σ, median absolute deviation *MAD*, and inter-quartile range *IQR*. In this category, we also have statistics (also robust counterparts) on the skewness of the distribution, i.e., skewness *skew*, octile-based skewness $skew_{oct}$, *MAD* based skewness $skew_{MAD}$. These values encode whether the center of the distribution leans to left or right. The fourth statistical moment, how steep the distribution of the values is also represented with a number of statistics: standard kurtosis *kurt*, octile-based $kurt_{oct}$, and *MAD*-based kurtosis $kurt_{MAD}$. The third category enables analyst to investigate the type of underlying distribution model. We check whether the data is coming from a normal distribution through a normality test score $norm_{shp}$, and check whether the distribution is uni-modal through test called dip test [82] *dip*. The fourth category investigates the correlation relation between the dimensions and aids an analyst to explore whether a dimension is unique or shares similar characteristics with the rest of the dimensions. In order to compute the measures in this category, we compute both the Pearson correlation [33] and the Spearman's rank-based correlation [33] between all the pairs of dimensions. For each dimension we find the minimum correlation (pr_{min}, sp_{min}), maximum correlation (pr_{max}, sp_{max}), and the number of significantly correlated dimensions (pr_{sign}, sp_{sign}). The statistics and the measures are listed in Table 3.1. For further details on how the measures are computed, refer to Paper C.

Within the context of this thesis, we use a number of computational analysis methods in addition to PCA. We use multi-dimensional scaling (MDS) as an alternative dimension reduction method that preserves the distances between the projected items as well as possible. We also use MDS directly on the dimensions,

Table 3.1: Statics and measures used in analyses are categorized depending on the type of insight they provide.

Category	Statistics/Measures
Characteristics of dimensions	$uniq$, $\%out$
Summary of the distributions	μ, σ, $skew$, $kurt$, med, MAD, IQR, $skew_{oct}$, $kurt_{oct}$, $skew_{MAD}$, $kurt_{MAD}$
Type of the underlying model	$norm_{shp}$, dip
Uniqueness of dimensions	pr_{max}, pr_{min}, pr_{sign}, sp_{max}, sp_{min}, sp_{sign}

similar to the VAR display by Yang et al. [212]. We use the correlations between the dimensions to compute a distance matrix, where this distance information is used as an input to MDS. Moreover, linear discriminant analysis (LDA) is used as a supervised discrimination algorithm and different clustering algorithms, such as k-means and hierarchical clustering, are utilized to find groups in the data. Note that, all these tools are integrated with the interactive mechanisms and operate only on the selected dimensions/items as described above for PCA.

3.1.2 Considering structures in high-dimensional data

Since the dual analysis approach enables an analyst to visually investigate the characteristics of dimensions, it provides us the foundations to discover and to consider the structures within the space of dimensions. The structures can be based on different properties, for instance, they can be an explicitly known categorization of the dimensions, e.g., collected through different data acquisition methods, or it can be dimensions sharing similar information, e.g., same measurements but in different scales. In order to achieve a *structure-aware* analysis of the data, we represent the underlying structures with *representative factors*, or factors, for short. We then analyze and evaluate these factors together with the original data to achieve a more informed use of the computational analysis tools. In the conceptual illustration Figure 3.7, we start by analyzing the dimensions on a s_1 vs. s_2 scatterplot (1). We notice a structure (a cluster in the lower right) which we then represent with a factor (2). With the help of a computational method, e.g., PCA, we generate the representative factor for the selected group of dimensions and replace these dimensions with the generated factor (3). We continue the analysis by exploring the relations between the factor and the represented dimensions, as well as the other dimensions (4).

Constructing factors that are useful for the analysis is crucial for our method.

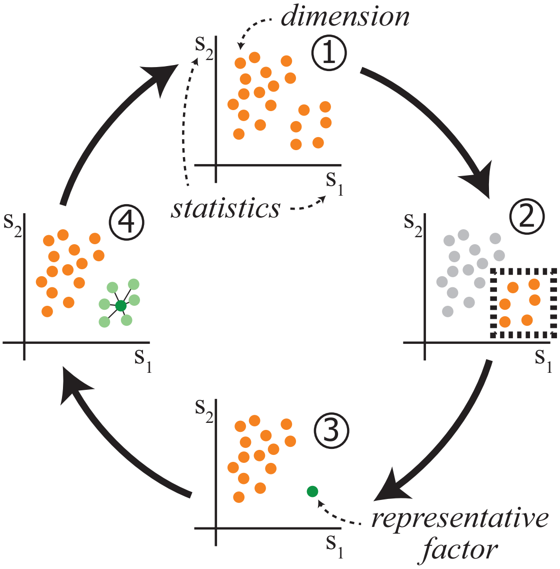


Figure 3.7: An illustration of our representative factor generation method. A view of the dimensions over two statistics s_1 and s_2 (1) reveals a group that shares similar values (2) and this group is selected to be represented by a factor. We generate a representative factor for this group and compute the s_1 and s_2 values for the factor (3). We observe the relation of the factor to the represented dimensions and the other dimensions (4) and continue iteratively.

Since factors are representatives for sub-groups of dimensions, they are constructed to preserve different characteristics of the underlying dimensions. We use three methods to construct representative factors where each method is a mapping from a subset of dimensions D' to a representative factor D_R. We describe three types of factors: *projection*, *distribution model*, and *medoid* factors.

Projection factors are generated using the output of projection-based dimension reduction methods that represent high-dimensional spaces with lower dimensional projections. Projection factors are preferred when we want the resulting factor(s) to represent most of the variance of the underlying dimensions [100]. This type of factors are suitable to apply computational analysis methods locally, especially concerning dimension reduction methods.

Distribution model factors represent the underlying dimensions with a known distribution where the distribution parameters are derived from the underlying dimensions. Distribution model factors are suitable to represent groups of dimensions that share similar underlying distributions. This type of factors are suitable for distribution fitting tasks.

The third type of representative factors, *medoid factors*, are generated by selecting one of the members of D' as the representative of D'. Such factors are preferred when the dimensions in D' are known to share similar contextual properties or some of the dimensions could be filtered as redundant.

Performing analysis locally

The mechanisms to detect and create factors enable the analyst to use computational tools locally and represent/compare the results in a shared visualization.

We include the factors into the dimensions visualizations by computing all the statistics that we already computed for the original dimensions also for the representative factors. We add these values on D_R as a row to the table S. This enables us to plot the factors together with the original dimensions.

Figure 3.8-a shows the dimensions of a dataset with 264 dimensions in a plot of *med* vs. IQR. Here, it is known that the dimensions consist of 12 subgroups, which are represented explicitly in the form of meta-data. In order to apply the analysis locally on these 12 structures, we create a representative factor (of projection factor type) for each of these subgroups of 7 dimensions D'. Here, we prefer to use PCA to compute the representatives using the following steps: i) For each representative factor, PCA is applied on the 7 dimensions and the data is projected onto the first principal component. ii) *med* and IQR values are computed using the projected values. iii) The original dimensions (the 7 dimensions) are replaced in the visualization with this representative (Figure 3.8-b).

The representatives are colored in shades of green to distinguish them from the original data dimensions. In order to see how a single factor relates to the represented dimensions over the *med* and IQR values, the factor is expanded and connected with lines to the represented dimensions (Figure 3.8-c). The relations between the factor and the represented dimensions are also observed on a *skew* vs. *kurt* view (Figure 3.8-d). To communicate the quality of the constructed factors, two color mappings are used to indicate the strength of the relation (via correlation calculations detailed in Paper B) between the factor and the represented factors.

Our goal with representative factors is not to solely assist dimension reduction or the use of computational methods but rather to enable an informed use of automated approaches on the explicitly known or observed local structures to achieve a better understanding of the data. Moreover, this mechanism is one of the early examples where derived data attributes, i.e., the factors, are visually analyzed together with the actual data dimensions. This enables a seamless integration of computational results within the analysis of raw data.

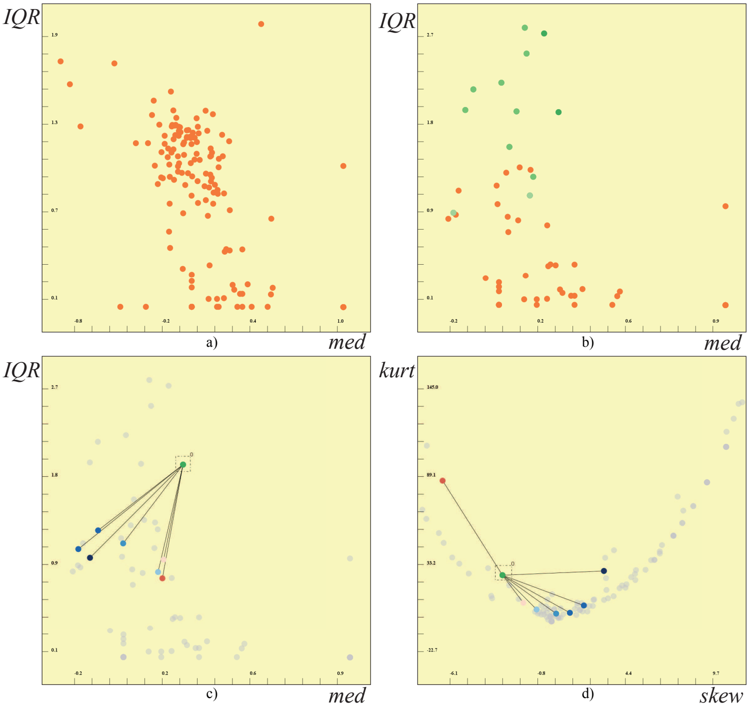


Figure 3.8: Integrating factors in the visual analysis. a) The normalized dimensions of a high-dimensional dataset (the ECG data introduced in Paper B) are visualized in a *med* vs. *IQR* scatterplot. b) Each sub-structure in the data (known explicitly by the analyst) is represented by a factor. The coloring is done based on the aggregated correlation. c) The factor for one structure is expanded and visually connected to the dimensions it represents. The coloring is done on the mutual correlations between the factor and the represented dimensions. d) The relation is different when *skew* and *kurt* values are considered.

3.1.3 Outlier-aware analysis of high-dimensional data

We have seen in the earlier parts of this thesis that the set of dimensions is usually heterogeneous. This might be due to the structures as discussed above – a single large subgroup or several smaller subgroups of these dimensions may contain related data and thus be highly correlated with each other. In addition to such heterogeneity, there may be dimensions that have “special” characteristics that are not shared with the others. When analyzing high-dimensional datasets, understanding the related groups of dimensions and those that *stand out* from the rest is highly important. In this section, we focus on understanding these *outlier* *dimensions.* We are motivated by the fact that outlier dimensions can easily skew and/or dominate the results of computational analysis tools. An example of this is PCA, where dimensions with very high variance tend to be highly expressed in the results, suppressing the structures in dimensions with low variances [30]. As for this example and for others that involve the use of computational tools, being aware of outlier dimensions could improve the analyses significantly.

Due to the significance of such special dimensions, we present a methodology to analyze high-dimensional datasets with a special consideration of *outlier dimensions.* An outlier-aware analysis process is possible by addressing three different stages: *characterizing*, *determining*, and *handling* outlier dimensions. These stages are important steps in an analysis session, where the analyst progresses through these stages with the help of the interactive visual methods introduced in the remaining of this section. We now follow with a detailed description of these stages and corresponding methods.

Characterizing outlier dimensions

We provide a concrete definition of outlier dimensions by a categorization of the dimensions based on the sources of outlyingness and propose three types: *characteristic*, *distribution based*, and *structural* outliers.

The first perspective in the evaluation of the outlyingness of dimensions is to consider their characteristic properties. With characteristic properties, we refer to the inherent properties of dimensions such as the type of data values (numeric, textual, etc.), the number of missing data values, or, the percentage of 1-dimensional outliers. If most of the dimensions in a dataset have continuous data values (e.g., floating point numbers) and two of them have categorical data, the latter ones can be considered as *characteristic outliers.*

The second perspective of outliers is related to the distribution of the items in a dimension. This type encompasses the dimensions that have distinct distributions compared to the rest of the dataset. For example, if most of the dimensions in one dataset are normally distributed and there is a couple of dimensions that are uniformly distributed, these dimensions could be considered as *distribution based outliers.*

As the third perspective, we consider the correlation relations within the dimensions. If a single dimension, or, a group of dimensions that are very strongly correlated, has very little correlation to the rest of the data, then this dimension(s) can be marked as of type *structural outliers.*

Methods to determine outlier dimensions

In order to facilitate the visual investigation of outlyingness of dimensions, we firstly make use of the categorization of statistics/measures introduced in Section 3.1.1 and determine which measures can provide insight on which type of

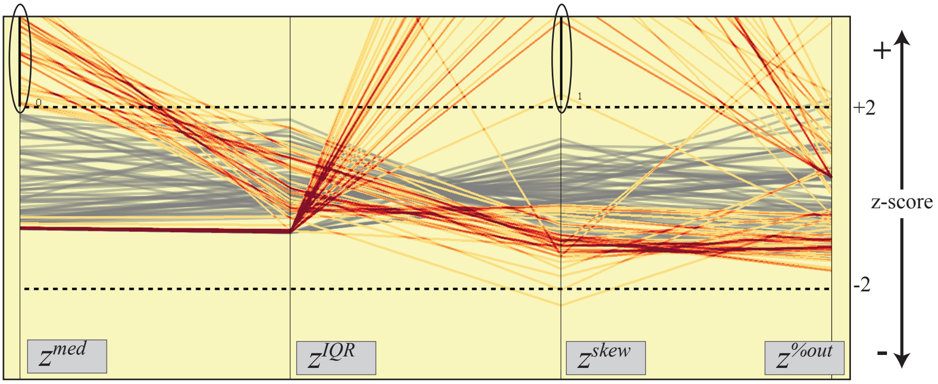


Figure 3.9: Our z-score view to visualize the z-scores for the dimensions over: *med*, *IQR*, *skew*, and, *%out*. Here, each line is a dimension and the dashed lines indicate the $[-2, 2]$ interval to ease the selection of potential outlier dimensions.

outliers. This mapping between the measures and the outlier types provide the analyst a guideline on which dimension space views to use while building the analysis setup. Details of this mapping can be found in Paper C.

In addition to the analysis of the dimensions through the multi-view setup, we also develop a number of novel interactive visual analysis mechanisms that make use of the state-of-the-art tools from statistics domain. These tools facilitate the outlier analysis performed on the statistics table S which has k values for each of the p dimensions. Depending on how many of the k statistics are considered, we resort to different methods for the evaluation of outlyingness. Notice that the following approaches can be considered as *implicitly* using computational tools.

z-Score view: Dimensions can be outlying with respect to a single statistic, e.g., if the σ values of all the dimensions are considered, dimensions with exceptional σ values are considered outliers with respect to σ. In order to determine the outlyingness of dimensions with respect to a single statistic, we compute the z-scores for all the dimensions for all the k statistics and visualize these values through an extended parallel coordinate plot called the *z-score view*. In this view dimensions with z-scores lying outside the $[-2, 2]$ range are highlighted as potential outliers. In Figure 3.9, each axis corresponds to the z-score values that are computed for 4 different statistics, *med*, *IQR*, *skew*, and, *%out*. Note that here, each line corresponds to a dimension. We enhance the view with two dashed lines that pass through -2 and 2.

Depth-based view and brushing: In order to support the identification of outlier dimensions through scatterplots, we enhance them with *data depth* calculations. Depth of a data item represents how central it is with respect to the distribution of the other items. The depth value computations are communicated

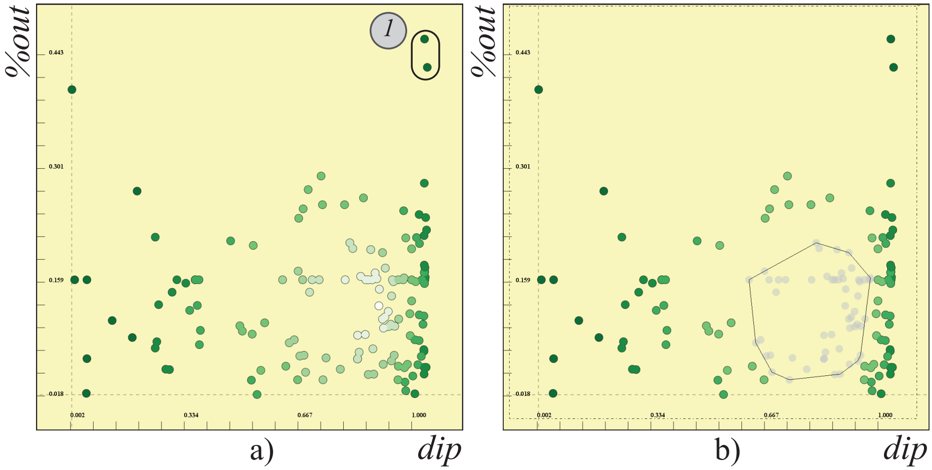


Figure 3.10: a) The dimensions are colored according to their depth values. The "deeper", i.e., central, points have a whitish color and the points on the outskirts (marked 1), i.e., possible outliers, have a saturated green color. b) Depth based brushes snap to different depth levels to aid the selection of different structures in the data.

via coloring: The possible outlier dimensions have saturated green colors (e.g., point marked 1 in Figure 3.10) and more central dimensions have less saturated colors.

We enhance the selection mechanism in scatterplots with depth-based brushes. These brushes enable us to easily (de)select points which are in the center or at the outskirts of the distribution of points. Since depth values are usually used to categorize data points into layers called depth contours [160], we develop brushes that are able to snap to such depth-contours. This mechanism can be seen as a step towards a *context-aware* interaction approach where the selections have an inherent "meaning", e.g., which depth layer the brush selects.

Outlier-aware analysis strategies

We describe a number of strategies to approach outlier dimensions to achieve the outlier-aware analysis of high-dimensional data. When an analyst determines outlier dimensions using the methods described above, we suggest four different approaches to treat the outliers: *leaving out*, *transforming*, *treating separately*, and *treating hierarchically*.

One of the first options that is commonly used in the analysis of the data items is to leave out outliers. Although this option could be practical for certain tasks, it might lead to the loss of relevant information. So this is a valid option for cases where the outlyingness is caused by severe problems in the data acquisition stage. The second alternative transforms the outlier dimension such that the source of outlyingness is "cured". The related literature in statistics and data mining suggests methods such as replacing missing data [163] or transforming data items via log or inverse transformations [149]. In certain cases, the outlier dimensions might be considered as the main focus of the analysis and treated separately in a parallel analysis session. One might gain further insight by performing the analysis with vs. without outliers. And a final approach involves an hierarchical consideration, where the analysis is carried out locally in sub-structures that contain outlier dimensions. Methods presented in the previous section, i.e., representative factors, could be incorporated to perform this strategy.

All these methods and approaches together are the enabling building blocks of outlier-aware analysis processes. Without properly determining and handling outlier dimensions, analysis results are often skewed. In Paper C, we include a number of cases where the careful consideration of outliers improve the analysis results.

3.2 Interactive and Visual Methods for Cluster Analysis

Cluster analysis is a widely used method that reveals underlying structures and relations of items by assigning them into several groups called clusters. The group of items in a cluster are similar with respect to certain features of the data. In the context of this thesis, we consider cluster analyses that are performed over both static and temporally varying data. Conventionally, cluster analysis starts with selecting a clustering algorithm and setting a set of parameters to produce an according clustering. To achieve a successful cluster analysis, however, it is of great importance to be able to both *evaluate* and *interpret* the resulting clusters–a task that analysts can benefit greatly from interactive and visual methods.

Ideally, any cluster formation step should be followed by an evaluation phase where the user decides whether she is satisfied with the clustering, or not. The evaluation of the clusters is important due to the fact that the choice of the algorithm and the parameters greatly affect the analysis outcome. Moreover, clustering algorithms provide the analyst with a grouping structure with limited information on what brings the members in the cluster together and what characteristics the grouping has. This makes it hard to interpret the resulting clusters. In the case of temporal data, these tasks are even more challenging. Unlike clusters of static data, temporal clusters have temporal spans in addition to the group of items they represent. Due to the fact that temporal clusters do not usually exhibit stable structures, both cluster-cluster relations and the structure of temporal clusters vary. Since current techniques do not address the challenges in analyzing the structural variations in temporal clusters, there is a need for methods to answer questions such as: "How does the quality of clusters vary over time?" and "What type of structural changes do clusters exhibit?".

We enhance the cluster analysis process by incorporating interactive and visual methods to aid the evaluation and interpretation of clusters. For temporal clusters, we propose two novel and interactive visualization techniques. Firstly we introduce the *temporal cluster view* that visualizes the structural quality of temporal cluster sets over time and secondly we present *temporal signatures* which are visual summaries of temporal cluster structures. In addition, we describe how significant difference views are utilized to characterize clusters in the analysis of heterogeneous data.

3.2.1 Analyzing Temporal Cluster Structures

Our solution for the analysis of temporal clusters is based on the temporal cluster view (in the following just "cluster view") and temporal signatures. The cluster view visualizes the quality of clusters together with structural changes that are related to item-cluster and cluster-cluster relationships. And temporal signatures are visual summaries of the statistical properties of clusters over time. The variations of these statistical properties reveal structural changes in groups of items.

A temporal cluster represents a group of items that display similar properties over a time interval. In order to generate such clusters, the clustering algorithm is applied to a temporal subset of the data. In this thesis, we use both hierarchical and k-means clustering [181]. As these algorithms are originally developed for static data, we modified the distance measures to incorporate the temporal nature of the data as suggested by Liao [125].

Temporal Cluster View

The temporal cluster view enables the visual exploration of clusters that are defined over time intervals. It depicts the evolution of cluster memberships and also encodes the commonly used cluster quality measure, *silhouette values*, in the visualization.

In a cluster view, each axis represents a clustering result over a different temporal span. Each curve between the axes represent a single data item and all the axes contain a set of clusters where each cluster is represented by a rectangle. The clusterings are ordered according to the start of their temporal span, i.e., clusterings applied on the beginning of the sequence is placed to the left. The duration of a clustering in time is visualized on top of the visualizations and connected to the axes with colored curves. This visualization enables the user to compare consecutive clusterings over time and observe how the membership

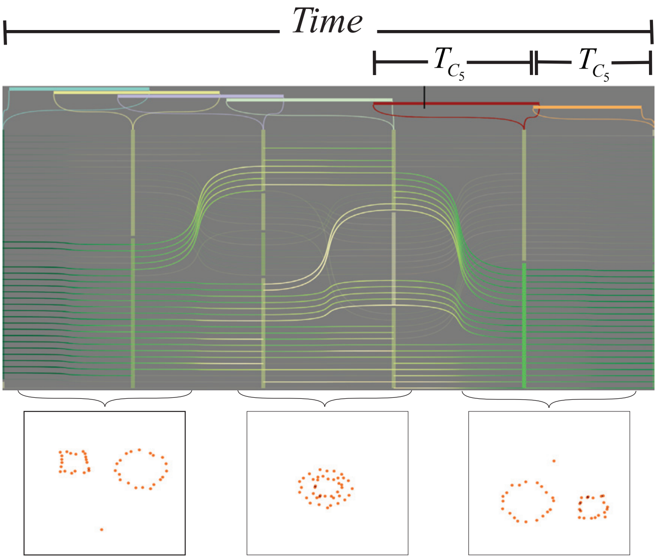


Figure 3.11: The temporal cluster view colored with silhouette values. The temporal ranges of clusters are ordered from left to right, earlier clusters to the left. Group structures change as items move over time. Since two separate groups merge and there is no clear clustering in the middle of the sequence, the silhouette values are low (yellowish color) – indicating that the clustering might have problems. In the beginning and the end, the two groups are well separated, so the clusters in these time zones have higher quality.

relations evolve. Figure 3.11 shows how the cluster structures and the members of these clusters change as the items move over time (two separate groups merge and split later within the sequence). The cluster members change when the two groups meet and form a group of items that is harder to cluster (the middle plot in Figure 3.11). Similar visualizations have been used in the literature to investigate the set of clusters [122] and we extend such visualizations with temporal clusters and the communication of quality measures.

In order to encode information about the structural quality of clusters, we utilize the *silhouette coefficient* [157] that indicates how well an item fits a particular cluster within the set of available clusters. Silhouette values are computed per each item of a cluster and they are in the range $[-1, 1]$. Items close to cluster centers have higher values, items on the borders of a cluster with close neighboring clusters have values close to 0, and items that are likely to be placed wrongly have values close to -1. In the cluster view, we use silhouette values to color code curves and cluster rectangles and the higher quality items/cluster are rendered in more saturated shades of green. This encoding enables an analyst to investigate the structural quality of the clusters over time. An example of how silhouette values vary can be seen in Figure 3.11. As the distribution of items where two groups meet is quite uniform, we see that the colors of items and clusters are not green – silhouette values mostly below 0, thus indicating not so strong cluster members. However, near the beginning and at the end of the sequences, the overall cluster quality is high, and this is clearly visible from the coloring where items have saturated green color, i.e., silhouette values close to 1 due to the nicely separated groups. This observation yields to the fact that clusters performed over the merging interval are lower in structural quality and therefore, have to considered with more care when further analysis is performed on them.

Temporal Signatures

Temporal signatures are visual representations of statistical properties of clusters over time. These structural properties are: *cluster cohesion* which represents the tightness of its items, and *cluster homogeneity* which correspond to the uniformity of the distribution of the member items [181]. These properties are important in detecting events such as cluster merging/splitting, and also in evaluating the stability of the cluster over time.

In order to construct these views, we rely on a qualitative approach and compute measures over time for each cluster. We compute the *minimum* and the *maximum of the distances* between each cluster member and the other members. We then aggregate these measures to estimate a *diameter of a cluster*. In addition, we compute how *compact* a cluster is by the *vicinity measure*. We compute these measures for each time frame independently.

A temporal signature represents the changes in these statistics over time to depict the structural variations within a cluster. Figure 3.12 shows how the temporal signature for the set of points mentioned earlier (separated in the beginning, merging and splitting later on). Here, the x-axis represents time and the y-axis the diameter of a cluster. The coloring between the upper and lower bands indicate how compact the cluster is, i.e., red indicates a compact and blue indicates a loose group. Notice in Figure 3.12 that as the two groups merge, the distribution of the points become more compact and small as indicated by the visualization.

When an analyst tries to analyze a large collection of clusters, these temporal signatures provide a quick overview on the set of temporal clusters. It is possible to quickly decide to either use, discard or update a cluster by observing their temporal signatures. Figure 3.13 shows how certain clusters within a selection of 15 clusters can be discarded (marked with X) due to their instable behavior.

In Section 4.3, we demonstrate the use of our methodology in the analysis of temporal clusters within molecular dynamics simulation data.

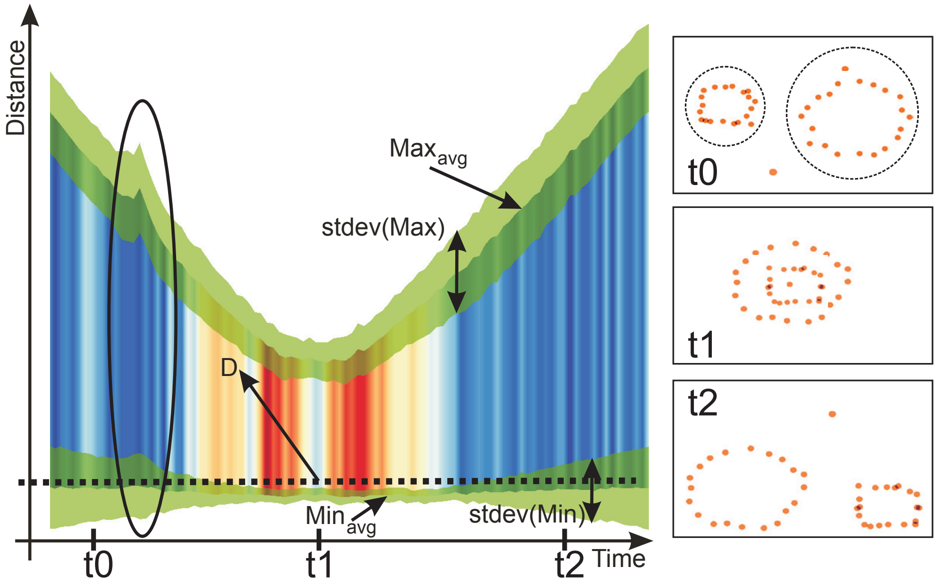


Figure 3.12: The temporal signature visually communicates the structure of the moving items. The top and bottom boundaries indicate the minimum and maximum distances within the items. In the beginning and the end, the two groups of items are further away from each other, and this is depicted with higher distance values at t_0 and t_2. At t_1, the group is tight (communicated with the red color) and the within item distances are minimal.

3.2.2 Characterizing clusters

Due to the increasing availability of different data acquisition methods, the analysis of groups over heterogeneous data is becoming common practice. With heterogeneous data, we refer to several datasets each of which is high-dimensional and linked with the other datasets over common identifiers. In order to understand the grouping structures, researchers apply clustering algorithms on each of these high-dimensional dataset separately and try to compare the results over different datasets.

The "Temporal Cluster View" introduced earlier and other similar visualization tools [116, 124] support this task to a certain level. In these approaches, the visualization provides insight only on the membership overlaps. However, in order to characterize a cluster, it is also important to analyze which dimensions contribute to the forming of a cluster and which characteristics the member items have in a cluster.

In order to support this task, we integrate the dual analysis views (of both the

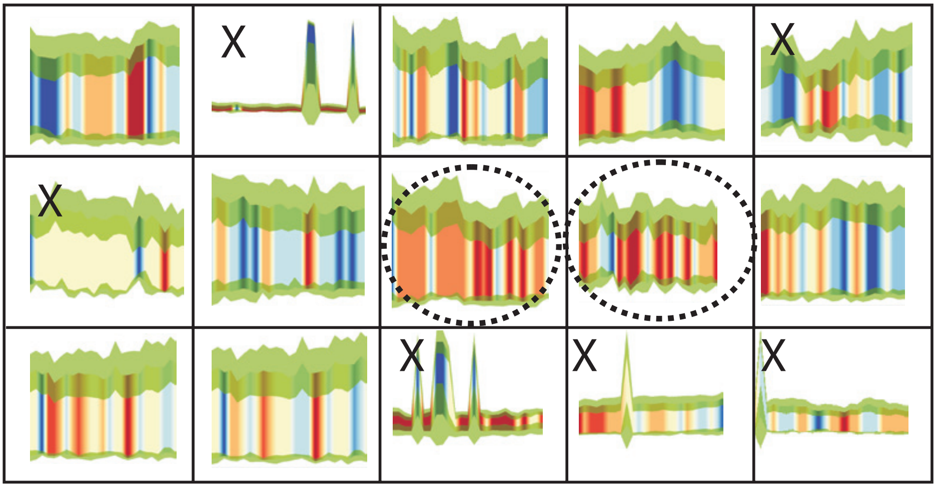


Figure 3.13: A number of clusters are evaluated by observing their temporal signatures. Depending on the observations, we discard some of them (marked with X). Some of these clusters show irregular structures, e.g., 2nd in 1st row, and others have loosely located members, 1st in 2nd row. Tightly distributed, stable clusters are selected for deeper investigation (marked with dotted circles).

items and the dimensions of a dataset) in a visualization framework that provides insight on the membership overlaps between clusters of heterogeneous datasets called StratomeX [124]. In StratomeX, clusterings are represented as columns. Each column consists of multiple stacked "bricks", where each brick corresponds to a group of item members (a cluster) in the column's clustering. Ribbons with varying width visualize the overlap between groups of neighboring clustering. We extend this by incorporating two types of views as *bricks* in StratomeX: i) scatterplots of statistics depicting either the genes or the samples, ii) significant difference plots. This integration provides a deeper characterization of clustering results by an analysis of distinctive elements and statistical profiles of cluster members.

The embedded dual analysis views in StratomeX can be seen in Figure 3.14. If the embedded scatterplot is a visualization of the samples (having a yellow background), it only displays those samples that are members of the represented cluster (see columns 1 and 2 in Figure 3.14). In this type of plots, the samples are visualized with respect to their *median* and *IQR* values computed for each row of the dataset. Notice that each point in a scatterplot represents a data item and the number of points in each of the scatterplot brick is equal to the number of members of the cluster.

On the other hand, if a scatterplot of dimensions is preferred, the brick dis-

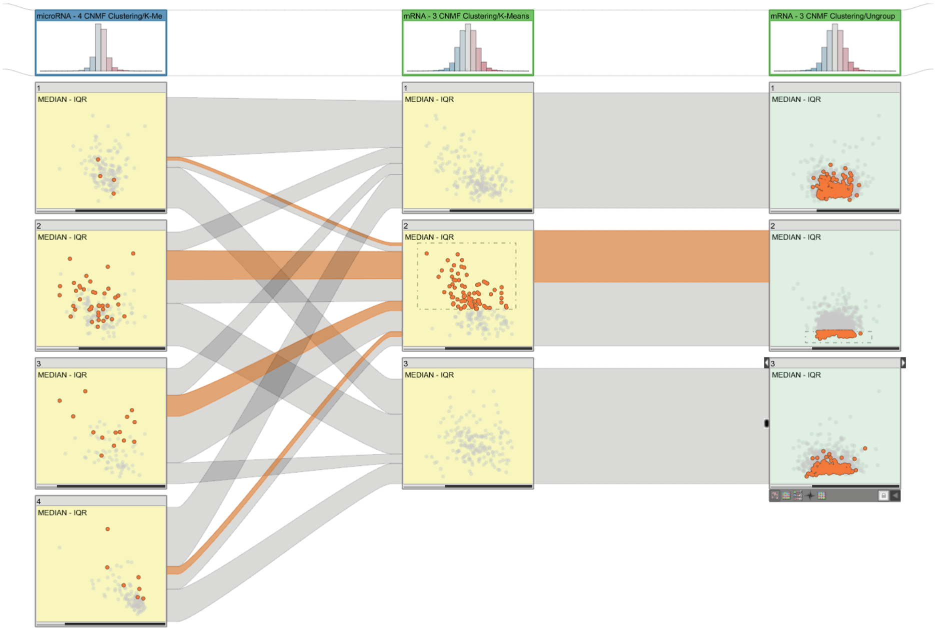

microRNA - 4 CNMF Clustering/K-Me
mRNA - 3 CNMF Clustering/K-Means
mRNA - 3 CNMF Clustering/Ungroup
1
MEDIAN - IQR
2
MEDIAN - IQR
3
MEDIAN - IQR
4
MEDIAN - IQR

Figure 3.14: Embedded dual analysis views in the StratomeX view [124]. The first column shows a 4-cluster stratification for a dataset. The scatterplots show median versus inter-quartile-range for the items in the cluster. The second column shows a 3-cluster stratification for another type of dataset, again showing items. The third column uses the same 3-cluster stratification for the same dataset, but shows the dimensions instead of the items. The scatterplots of items (yellow background) depict the statistical characteristics of the members of each cluster and the scatterplots of dimensions (light-green background) depict statistics computed for the dimensions using only the items from the cluster represented by the brick. The selection of items is highlighted in the first two columns and also in the ribbons. The selection of the dimensions makes it possible to investigate the distribution of the values for the dimensions for different clusters in a stratification.

plays the statistics (again *median* and IQR) for all the dimensions computed using *only the members of the cluster being represented*. In these embedded scatterplots with a blue background, each point represents a dimension of the data and each scatterplot contains the equal number of points, i.e., the total number of dimensions p. However, all the scatterplots look differently since the statistics for the dimensions in each plot are computed using the subset of items in the represented cluster. In the third column in Figure 3.14, dimensions with lower variety (lower IQR values) are selected in the second cluster and we see that these dimensions have usually higher variety for the first cluster. This observation can be interpreted as: the selected dimensions have common properties, i.e., lower values, for the second cluster and thus can be good discriminative features. However for the first cluster, these dimensions have no observable distinctive value due to the high variety. Such observations are not straightforward to make with conventional methods although they are critical to interpret the clusters.

We also embed difference plots as bricks in StratomeX. While doing this, we compute the Δ_μ and Δ_σ values for each of the dimensions using Equation 3.1 in Section 3.1.1. Here, however, B corresponds to the samples that are members of the cluster being represented while C corresponds to *the rest of the samples* in the dataset. We choose to use the rest to perform the comparisons on non-overlapping subsets, i.e., members are not repeated in the two subsets. The resulting difference view bricks communicate which dimensions are more distinctive for each cluster. Moreover, the selection mechanism enables the analyst to compare these distinctive dimensions between different clusters.

Our approach facilitates the characterization of clusters by enabling an investigation of them over both the items and the dimensions. This duality in representing clusterings provide deeper insight on the characteristics of clusters. This new approach not only leads to higher quality clusters but also provides a better reasoning why clusters exist and relate to each other. Demonstration of how this is achieved is discussed in Section 4.2.

3.3 Considering human factors to enhance interactive data analysis

Most of the contributions of this thesis up to now focus on the integration of computational methods within visual analysis without addressing how this integration can be achieved optimally according to cognitive and perceptual capabilities of the users. The three *human-time constants* (*perceptual processing*, *immediate response*, and *unit task*) introduced by Card et al. [25] provide us a solid basis to address this aspect of the integration we describe in this thesis. These constants determine the temporal characteristics of human-computer interaction (at three different time scales) such that an optimal communication between the human and the computer can be achieved. With our method, we show how interactive visualization processes can be realized in visual analytics such that they adhere to these human time constants.

It is of vital importance to properly address the perceptual and cognitive capabilities of humans in visual analytics (VA), since it is an interactive and iterative dialogue between the human and the computer [84]. With our *three levels of operation* for analytical processes, we aim to moderate the temporal aspects of such integrations in order to meet the three human time constants. The third level manages the time involved in *completing an analytical task*, e.g., observing the relations between several variables in a dataset. This level is based on the *unit task* constant. The second level moderates the human-computer dialogue and ensures that it occurs at a temporal pace where the human can give immediate responses, i.e., occurring within the limits of the second *immediate response constant* at which the parts in a communication are exchanging without being interrupted. The first level is responsible to make sure that the updates in the visualizations happen at a rate that is perceptually suitable for the human and is based on the *perceptual processing* constant. These levels of operation, the human constants and the corresponding temporal durations can be seen in Table 3.2.

The *unit task completion* level (Level 3) determines the temporal range in which an analytical unit task is completed. Such an analytical task is performed to answer a specific question related to the data. We moderate the activity at this level by a novel interaction mechanism called *keyframed brushing*. In this mechanism, the user defines two or more brushes (according to his/her analytical goal), similar to defining key frames in computer-assisted animation [26]. Using these *key brushes*, a sequence of *in-between* brushes is generated automatically. After the brush sequence is computed, the system starts traversing through this sequence without the need for further input by the user. The complete sequence is traversed in 10 sec., 20 sec., or 30 sec., and moving from one brush to the next takes 1 second in accordance with the human time constants. The sequences are generated using four different methods as seen in Figure 3.15: *moving brush*, *extending brush*, *no in-betweening*, *constrained brushing*. Keyframed brushing enables the user to focus on the linked views that display the results of the animation rather than paying attention to moving the brush in a particular fashion.

The human-computer dialogue level (Level 2) is mainly responsible to maintain the dialogue nature of the visual analysis process. It ensures that the communica-

Level	Operation Level	Human time constant	Response time (sec.)
Level 1	Visualization update	Perceptual processing	0.1
Level 2	Human-computer dialogue	Immediate response	1
Level 3	Analytical task completion	Unit task	10 - 30

Table 3.2: The three levels of operation, the corresponding human time constants [25], and the associated time limitations

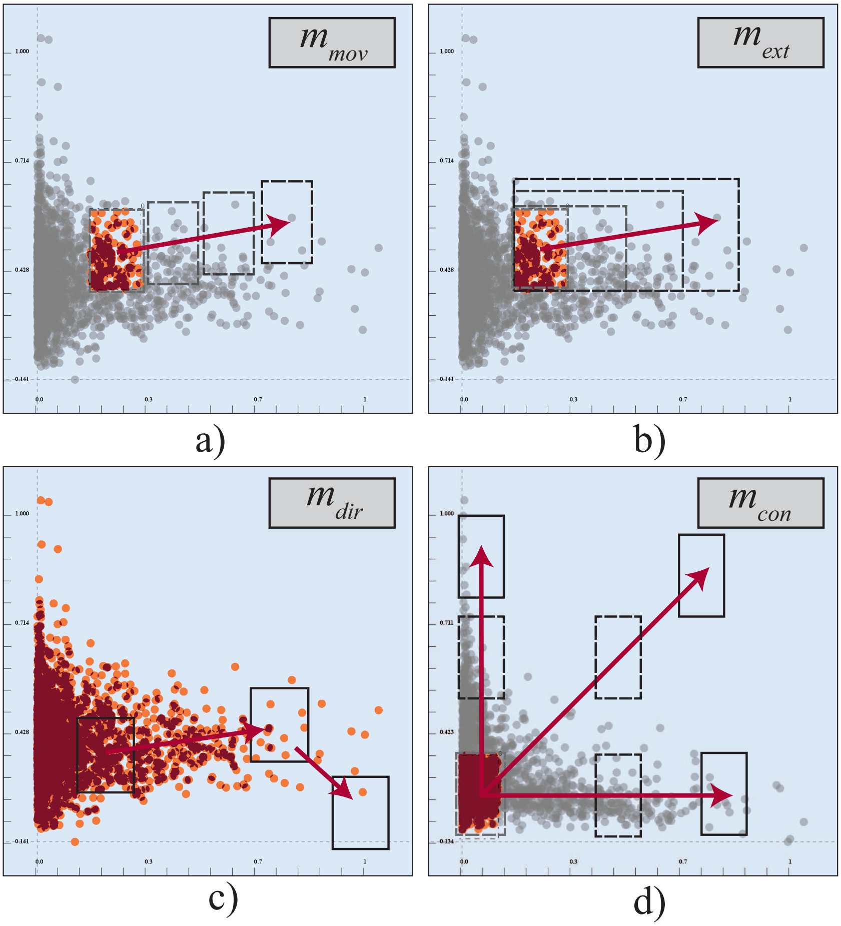


Figure 3.15: Four modes for keyframed brushing (according to the user interaction as illustrated in Figure 3-b). a) Moving brush mode: the position of the in-between brushes are linearly interpolated, b) Extending brush mode: the brush extends at every step, c) No in-betweening: the final sequence consists of only the three brushes, d) Constrained brushing mode: the path of the selection automatically snaps to one of the fixed lines (parallel to x-axis, y-axis, or to the diagonal)

tion between the user and the computer is not interrupted. This level focuses on maintaining a guaranteed response time (1 sec.) when integrated computational tools are utilized. Our solution to achieve this is to compromise the quality of the results by computing “only” the best possible result within the limited time frame. We achieve this by utilizing *online algorithms* that are capable of processing the data piece-by-piece sequentially [4] and do not need to access the whole

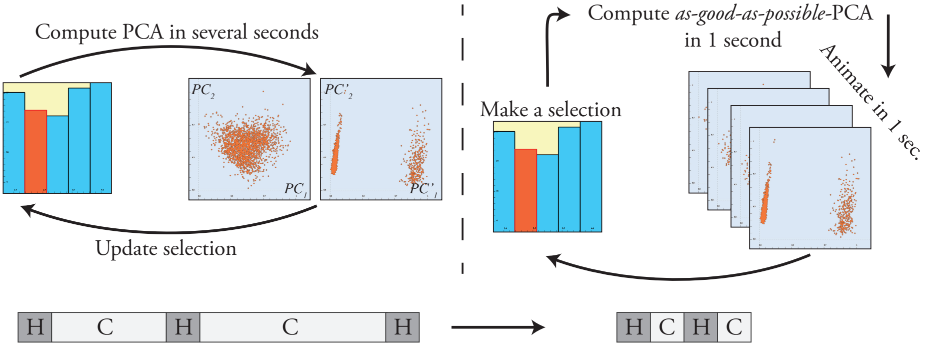


Figure 3.16: An example for optimizing an analytical process against the three human time constants [25]. In a conventional approach (left), a (re-)computation of PCA results is triggered (with a selection of variables), then the user waits a certain time for the results. When this time is long, this could potentially break the dialogue between the user and the computer. Our suggested optimization (right) addresses such issues by computing PCA results *as good as possible* within 1 sec. in response to a selection by the user. And whenever new input is received, the re-computation is done in no more than 1 sec. and the results are presented by animated transitions in 1 sec. The H–C–H–. . . - abstraction indicates the pattern of interaction (the lengths indicate the time spent).

data. We use the online algorithms with a suitable sampling strategy to provide the user the *best-possible approximate* result in no later than 1 second. And depending on the interpretation of these first approximate results, the user might either wait for more accurate results to compute or continue to explore the data by updating his/her interactive inputs. This temporally constrained mechanism enables the analysis to run smoothly at a pace that conforms to the *immediate response* time constant.

We make use of animated transitions between different computational results that are generated as a result of the dialogue occurring at the second level of operation. The visualization update level (Level 1) moderates the update rate of animated visualizations and secures the successful perceptual processing of the animations in the visualization. Smooth-in-the-eye animations are achieved by updating the visualizations at 10 Hz [25] or higher frame rates. Animations are either used to give *immediate responses* to user inputs, or they are constructed as a result of the keyframed brushing sequences. Animations aid the interpretation of the changes in the computational results in response to use inputs. We make enhancements to the animated views to support the analysis of changes even further. The details of these enhancements are in Paper D.

An example of how the three levels of operation optimizes an analytical process can be seen in Figure 3.16. Here, the analytical task involves the application of principal component analysis (PCA) to different subsets of the dimensions and observe whether there are interesting structures. The conventional process starts with an input from the user that selects a subset of the dimensions through an histogram depicting the set of dimensions (Figure 3.16-left). The computer responds to this user input by computing the PCA results and displaying them in a scatterplot. However, exactly at this point, there is an issue with the timing of the computations – the response time for the computer is undetermined, could be a millisecond or hours depending on the size of the data. This issue can easily break the communication between the human and the computer. On the other side, in the analytical process optimized by our three levels of operation, the computer responds to the user input in exactly one second by providing an *as-good-as-possible* result. The user uninterruptedly continues to observe other subsets until an interesting structure that needs further attention is spotted. With this optimized process, all the operations are performed with temporal characteristics that are in line with the communicative capabilities of the user. Such processes result in human computer dialogues that are not broken and likely to yield to more successful results.

Chapter 4

Demonstration cases

This section demonstrates the utilization of the methods introduced in the previous sections. During the research related to this thesis, we evaluated our methods on several different high-dimensional and temporal datasets. We have seen that our contributions enable analysts to perform tasks such as finding *relevant parts* of the data in very high-dimensional spaces, discovering *hidden relations* and *special features* when there is heterogeniety within the dimensions, and *quickly evaluating* results of several automated tools.

In the course of this thesis, our methods have been challenged by our collaborators working on medical, genetics, and molecular dynamics domain. Our methods provide them capabilities that were not possible in their previous analysis pipeline. In the analysis of heterogeneous medical data, being able to investigate the space of dimensions as a whole made it possible for the domain experts to be able to make observations that were not feasible with their current methods. This lead to a very productive hypothesis generation process as described in Paper E and a subset of the resulting hypotheses are discussed in Section 4.1. In the domain of genetics, and specifically in cancer subtype analysis, our approach makes it possible to analyze the relations both within the samples and the genes – insight very challenging to obtain with conventional approaches (Section 4.2). And in the analysis of clusters in molecular dynamics simulations, our collaborators were limited with simplistic measures to understand the structures in the data. Our methods provided them new ways to look at their data and opened up for new opportunities (Section 4.3). In all these cases, we have received very positive feedback and some of our collaborators expressed interest in making our methods a part of their daily scientific activity.

The papers in the second part of this thesis provide the details of analysis examples performed on a large variety of datasets. In the following cases, we present a selection of these analyses we have carried out and we showcase how our approaches enhance the analysis process with the integration of interactive and visual methods.

4.1 Hypothesis generation in heterogeneous medical data analysis

The analysis here demonstrates how interactive visual analysis methods, in particular using the dual analysis approach, facilitate the hypothesis generation process in the context of heterogeneous medical data. In this use case we analyze the data related to a longitudinal study of cognitive aging [8, 215]. In the study, all the participants were firstly subject to a neuropsychological examination, namely intellectual function (IQ), memory function, and attention/executive function, and later to multimodal imaging of the brain, i.e., 3D anatomical magnetic resonance imaging (MRI), followed by diffusion tensor imaging (DTI) and resting state functional MRI [89, 214]. One of the expected outcomes of the study is to understand the relations between image-derived features of the brain and cognitive functions in healthy aging [215]. The resulting dataset from the study contains information on 82 healthy individuals who took part in the first wave of the study in 2004/2005. MRI images were segmented into 45 anatomical regions, where seven features for each region were derived automatically. This process creates $45 \times 7 = 315$ dimensions per individual and with the neuropsychological examination, the resulting table has 373 dimensions, i.e., a 82×373 table.

In this study, we direct the analysis by treating age, sex, and the test scores as the dependent variables and try to investigate how they relate to the imaging based variables. As a group of visualization researchers and experts in neuroinformatics and neuropsychology, we perform several sub-analyses on this data where each of which results in an hypothesis, refer to Paper E for the details related to all the sub-analyses. Here, we comment on the findings related to the age of the participants.

In order to carry out this study, we interactively analyze the data through a combination of scatterplots depicting the items and difference views. And prior to our analysis we handle the missing values and perform normalization on the data. For numerical dimensions, we utilize z-standardization and we scaled the categorical data dimensions to the unit interval. We limit our interest to the elderly patients and aim to understand the effects of aging on the brain and the test results. We select the patients over the age of 60 (Fig. 4.1-a) and visualize how brain volumes and test scores change. We observe no significant difference in IQ & memory and attentive functions for the elderly patients (Fig. 4.1-b). However, when we observe the change in brain volumes, we observe that there is an overall *shrinkage in most of the brain segments with age.* This is clearly seen in Fig. 4.1-c, where most of the dimensions have smaller *median* values (i.e., to the left of the center line). Although most of the brain regions are known to shrink with age [200], some regions are reported to enlarge with age. When the dimensions that have a larger *median* value due to the selection (i.e., enlargement due to aging) are observed, they are found to be the *ventricles*

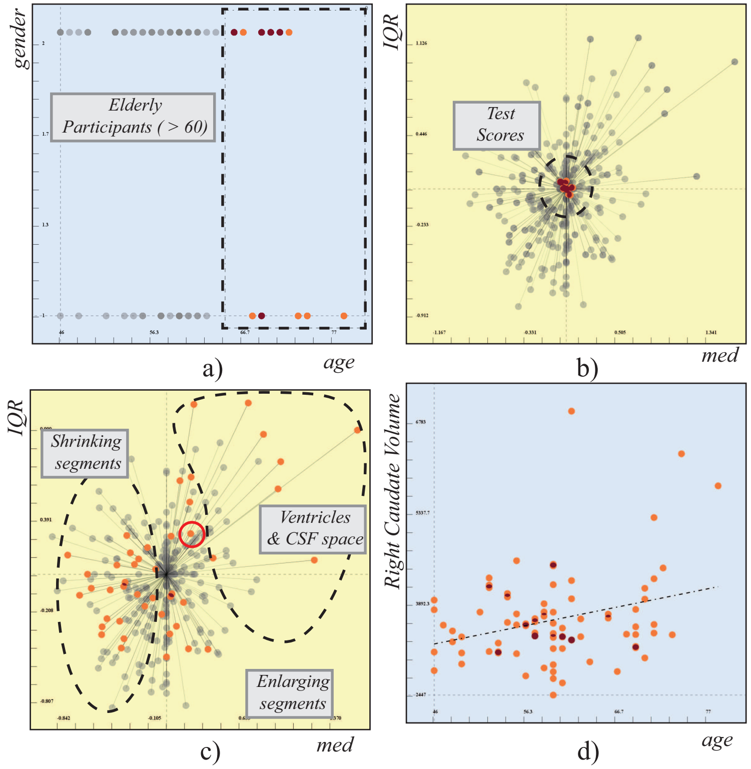


Figure 4.1: Elderly patients (> 60 years old) are selected (a). No significant relation is observed in the test scores (b). When we focus on the volumes of the segments, we see most of the regions are shrinking with age, but some, especially the ventricles, are enlarging (c). Apart from the expected enlargement of the ventricles, *the right caudate* is also found to enlarge with age (d).

(not the *4th ventricle*) and the *CSF space*. Since this is a known fact [200], we focused on the regions that show smaller enlargements and decide to look at *the right caudate* more closely. When *the right caudate* is visualized against age, a significant correlation is observed (Fig. 4.1-d). This is an unexpected finding that needs to be investigated further. With these above findings we build the following hypothesis: There is no significant relation between age and performance in IQ & memory and attentive & executive functions for individuals undergoing a healthy aging. Moreover, in contrast to most of the brain regions, there is a significant enlargement in *the right caudate* in healthy aging individuals.

We have seen that our explorative approach results in the generation of hypotheses quickly. The conventional routine to analyze this dataset is to physically limit the analysis to a subset of the dimensions and perform time-consuming, advanced statistical analysis computations on this subset. Our methods optimize this process by making the whole data available throughout the analysis and enabling the analyst to quickly swift through several subsets to generate new insight.

Representative Factors to Analyze Local Structures

In this use case, we continue to perform an analysis on the cognitive aging study data that is analyzed in the first use case. This time, however, we demonstrate how representative factors can be utilized to analyze such a structured data, i.e., brain regions vs. statistical features.

We start by investigating the 7 different features associated with the brain regions and generate 7 projection factors for these 7 sub-groups. We select these groups through the use of the available meta-data (not shown in the images here). Each factor here represents 45 dimensions, being the different brain regions, e.g., one sub-group contains all the *number of voxels* columns for the 45 brain regions. We visualize these factors over a *med* vs. IQR plot (Figure 4.2-a).

We mark the *standard deviation of intensities* as interesting, since the underlying dimensions have different correlation relations with the representative factor. This indicates that this feature is likely to show differences between the brain regions. In addition, we also consider the *range of intensity* feature to be important since it preserves most of the statistics in the underlying dimensions (this insight is made primarily by observing the profile plots introduced in Paper B).

We continue by delimiting the feature set for the brain regions to those two selected features, i.e., we delimit the operations to 45 × 2 dimensions. We first apply MDS on these using the correlation matrix as the distance values and we identify a group of dimensions that are highly correlated in the MDS plot (Figure 4.2-b). We find out that this group is associated with the sub-structures in the Cerebellum Cortex (CerCtx). We represent all the dimensions related to the CerCtx via a medoid factor and create a single projection factor for every other brain region. In Figure 4.2-c, we see the factors over a normality score vs. $\%out$ plot where each factor represents a single brain region. We select the representatives that show a normal distribution, since normally distributed dimensions provide a reliable basis to apply PCA on the participants. These regions of interest are *right and left lateral ventricle*, *brain stem*, *left and right choroid plexus* and *right inferior lateral ventricle*. In the resulting PCA visualization, we spot a group of outlier participants which we investigate further to evaluate the validity of our finding (Figure 4.2-d).

Firstly, we observe that these are mainly the elderly participants. And in order to investigate deeper, we refer to the internal reports written by our collaborators on the progression of the participants over the years. Through these reports, we observe that one of the nine participants is described as showing an

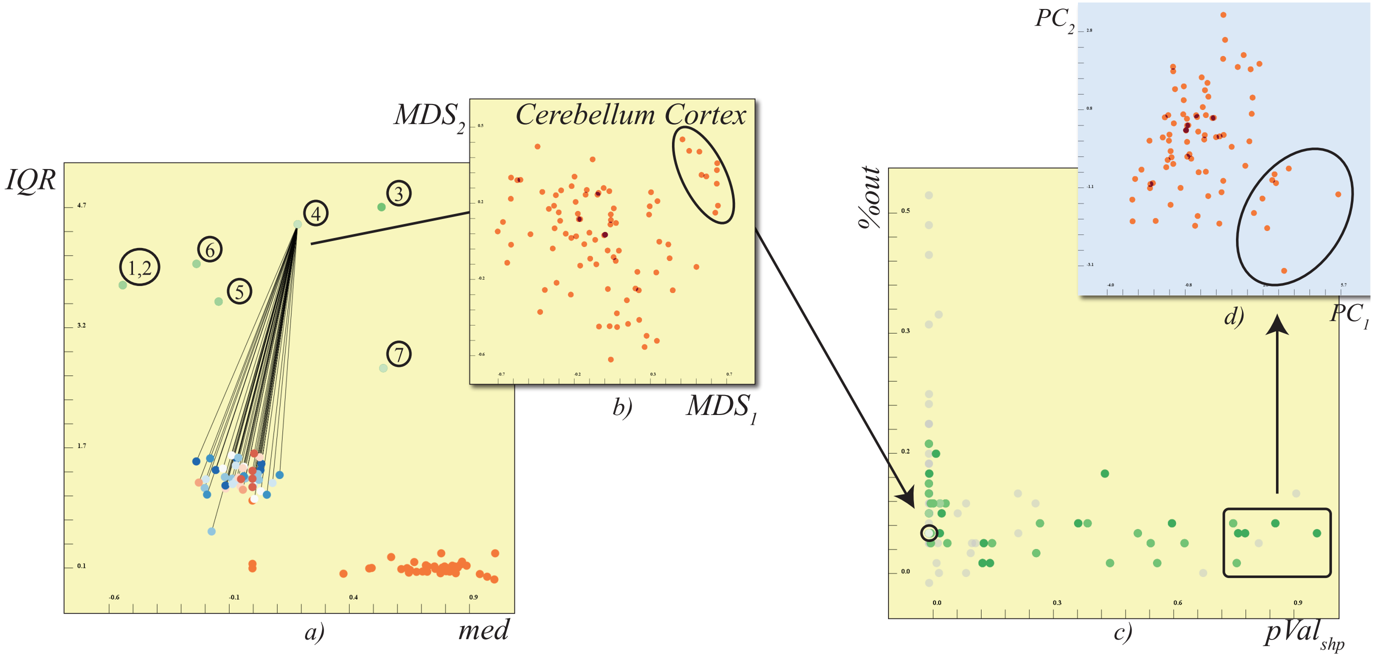


Figure 4.2: Analysis of the healthy brain aging dataset. We generate factors for the 7 types of features (a). Each factor represents 45 dimensions (the number of brain regions). One of the factors (4) has a varied correlation relation with the underlying dimensions and another factor (7) is a strong representative of the statistics over the brain regions (due to observations not shown here). For each brain region, we limit the features to these two and apply MDS on this subset of dimensions (b). The MDS reveals a tightly inter-related group of dimensions that is found to be associated with the Cerebellum Cortex (CerCtx). CerCtx is represented by a medoid factor and the rest with projection factors. These factors, each representing a brain region, are visualized on a $pVal_{shp}$ vs. $\%out$ plot (c). 6 of the "most normally" distributed factors are selected. PCA is applied on the participants. We notice a group of individuals with outlying values (d) and find out that this group consists of elderly subjects (not shown here). We can claim that the selected 6 brain regions are likely to be affected by aging.

older infarct (through MRI scans) and six of the remaining participants (75%) showed declining cognitive function during the study period. The percentage (of cognitive function decline) in the other participants is 28%. This shows a clinical importance of the selected participants. Moreover, this result supports the above hypothesis that the selected brain regions are related to age-related disorders. All in all, the above observations clearly suggest that the interactive visual analysis of the MRI dataset leads to significant and interesting results that are very unlikely to be achieved using conventional analysis methods.

4.2 Characterizing Cancer subtypes

In this case study, we utilize views constructed using our dual analysis approach together with StratomeX, a visualization method to represent overlaps of clusters (similar to the temporal cluster view introduced in Section 3.2). This case study is related to the cancer subtype analysis based on a variety of biomolecular datasets. The dataset is produced by “The Cancer Genome Atlas” project and captures several aspects of gene activity for a large number of participants. This gene activity is represented in the form of several datasets where one example is the mRNA data that measures the abundance of mRNAs in the cell. For a detailed description of the other datasets and details of a further analysis refer to Paper F.

In the analysis of cancer subtypes, analysts are faced with clusterings (stratifications) of several datasets. Although it is possible to investigate the membership overlaps with current methods [124], it is not possible to investigate what distinguishes a subgroup from another. Such an insight is possible by detecting the distinctive elements for a single group and observing these elements within the other groups. In order to address this task, we utilize our significant difference view as “bricks” within StratomeX.

We perform our analysis on the data related to a comprehensive breast invasive carcinoma (BRCA) dataset collected by the TCGA consortium. We use the mRNA expression data from over 800 breast cancer patients. In addition to the raw data, we load a recently published stratification of samples [114] that will serve as a basis for our analysis. The 4 subtypes that are reported in the reference study are: *Luminal-A*, *Basal-like*, *Luminal-B*, and *HER2-enriched*, as shown in Figure 4.3-a). Note that, in the difference views each point is a gene. The differences in each brick are computed between the members of the cluster being represented and the rest of the samples. Notice here that, in order to use non-overlapping sample sets, we prefer to compare each cluster to the rest of the data.

The reference study identified a list of genes that are differentially expressed for the *HER2-enriched* subtype by using unsupervised clustering (refer to supplementary Table 7 in [114]). We select the 7 most significantly under-expressed genes and 10 most significantly over-expressed genes as marked in Figure 4.3-a.

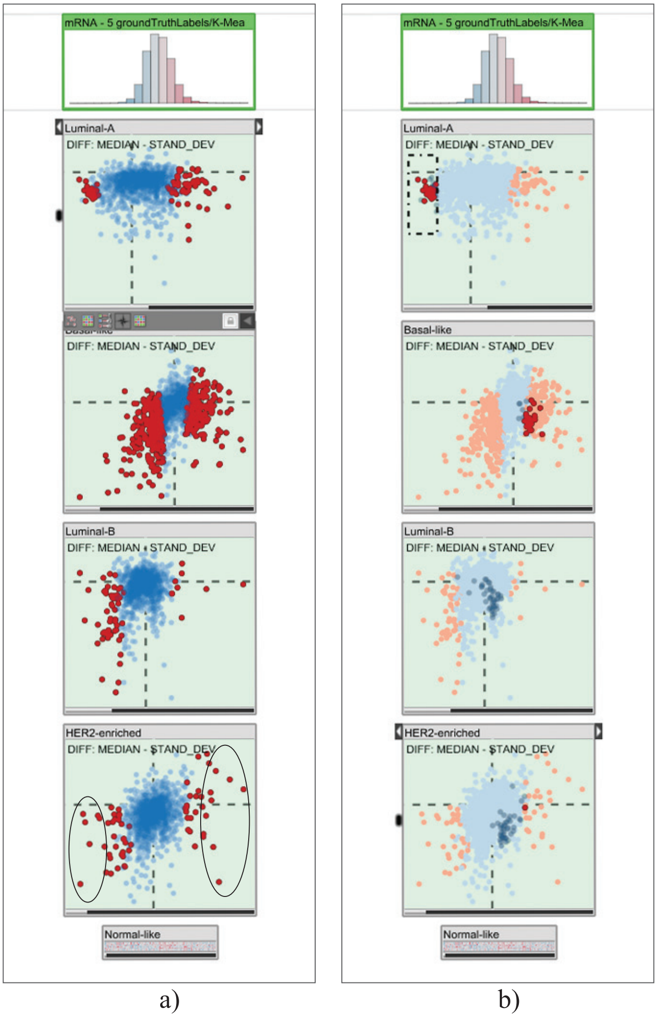


Figure 4.3: Using embedded difference plots to find descriptive genes. (a) Descriptive genes are marked for the *HER2-enriched* subtype. A comparison to the reference study shows the relevance of the marked genes (b) Under-expressed genes for the Luminal-A subtype are selected and we observe that they show over-expression for the Basal-like subtype, i.e., constitute good features to discriminate these two subtypes.

7 out of the 7 under-expressed and 6 out of 10 over-expressed genes are identical to the ones found in the reference study. This match demonstrates that our interactive visual analysis approach quickly yields relevant results in determining descriptive genes.

We then focus our attention in comparing distinctive genes between the *Luminal-A* and the other subtypes. We first select the significantly under-expressed genes for the *Luminal-A* subtype in Figure 4.3-b. We observe that the *significantly under-expressed genes for Luminal-A are often over-expressed for the Basal-like subtype.* This leads to the conclusion that *these genes are good markers to distinguish the Luminal-A from the Basal-like subtype.*

The capability to interactively select certain genes enables analysts to quickly get an understanding of the characteristics of a subgroup in relation to the other groups in the data. This enhancement to the subtype analysis demonstrates how the characterization of cancer subtypes might be facilitated through interactive visual methods. Paper F provides further utilization of our approach in supporting the characterization of cancer subtypes.

4.3 Analysis of Molecular Dynamics of Mixed Lipid Bilayers

In this use case, we perform an analysis on the data from molecular modeling of biological membranes. In a typical analysis, the focus is on the lipids that form the cell membrane. These lipids can form clusters with other membrane components which are relevant for signal transduction or cell apoptosis to name but a few [53]. Molecular dynamics (MD) simulations are often utilized to describe the atomic structure and dynamic behavior of lipids. After the simulation is completed, the analysis often starts with the generation of a grouping through clustering algorithms.

Our collaborators working in the field of biomolecular modeling state that they faced many limitations in performing an effective analysis on the group behaviors in previous work [23]. Due to the complexity of analyzing the clusters over time, they perform the clustering on individual time steps and average the clustering properties over time. This task is exactly where our temporal clusters analysis methods are particularly useful. We use our temporal cluster view and temporal signatures to investigate the set of temporal clusters and to determine “good” clusters that can provide insight on the structural changes within the lipids over time.

In this study, we work on a dataset obtained from a MD simulation of a mixed lipid bilayer [23], constituted of 2 types of lipids over 1640 time steps. We start the analysis by displaying the clusterings that are obtained by a clustering algorithm.

We assess the cluster quality, firstly, by brushing individual clusters and observing their silhouette values. And secondly, we assess the coherence of the clusters via the signature view. Fig. 4.4 displays a set of signatures for the observed clusters C_{1-5} defined over sequential time intervals $T_{C_{1-5}}$.

We mark two clusters to be interesting to analyze further. The first cluster,

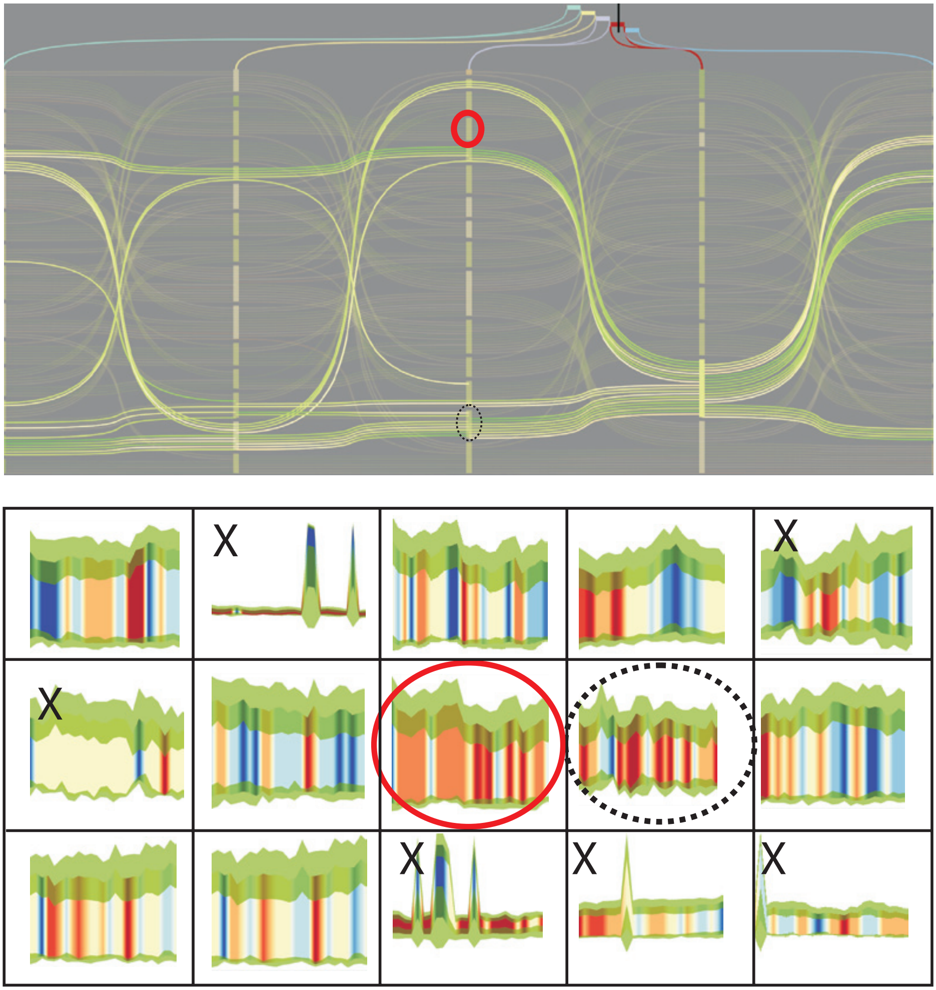


Figure 4.4: a) Cluster merging-splitting behavior. A cluster is selected with b_1 and the time selection is enlarged by brushes b_2 and b_3. Merging occurs around the smaller band in the middle, which gets larger at end of the sequence due to splitting in signature view. b) Searching for a plausible cluster. Two good signatures are identified (circles). The dashed circle is discarded due to its structural instability in cluster view (shown with the selection on the right). The red circled cluster is picked for further analysis. Moreover, the observed signatures allow to discard clusters (**X**) according to their structure.

marked with dotted circle can be considered a “good” cluster due to its temporal signature, i.e., stable and compact structure (mainly red). However, when the same cluster is observed in the cluster view, we see that there is a lot of branching visible for this cluster. This indicates that this is not a stable cluster over time and we do not pick that for further analysis (Discarded clusters are marked with an X in the figure). Nevertheless, we found a cluster (marked with a red circle in Fig. 4.4) that has both a plausible signature and also exhibits a stable structure in the neighboring clusterings. This cluster is a good candidate to build further analysis on. In Paper G, we discuss how the analysis is carried out further with such good clusters. In general, our collaborators find the procedure to be faster, more powerful and more reliable than traditional approaches which are usually based on distance criteria applied to each frame of the sequence.

Chapter 5

Conclusions and Future Work

This thesis aims to achieve the tight integration of computational methods within the interactive and visual analysis of data. We focused on high-dimensional data analysis and cluster analysis in different stages of the project. The current analysis pipeline that involves the use of automated methods has limitations regarding the *reliability and interpretability* of the results. This issue is due to several factors, such as the inherent *heterogeneity* within the dimensions, the *underlying assumptions* of computational tools, and the limited capability to perform *local analyses* on the data. With this PhD thesis, we managed to address these challenges and achieve analytical processes that yields results that are more reliable and easier to understand for the analyst.

During the research period of this thesis, we had the chance to collaborate with domain experts to work on challenging datasets and tried to solve analytical problems together. We have seen that our methods have improved their analysis pipeline. This improvement is mainly to due to *awareness* that the analysts gain with the help of our iterative and interactive visual methods. The awareness is facilitated by being able to understand the characteristics of the data and give *informed decisions* within the analysis in response to these investigations. During our research activities and our exchange with the collaborators, we made several observations regarding the analysis processes and practices.

- We have seen that our dual analysis approach enabled analysts, for the first time, to easily handle datasets with very high number of dimensions. Treating the dimensions as first-order analysis elements broadens the scope of the analysis to the space of dimensions. Our approach makes it possible to utilize most of the interactive visual analysis methodologies, e.g., *linking & brushing* and *focus + context* visualization, on the space of the dimensions. This new capability opens up for new possibilities that are not essentially limited with the number of the dimensions. Previous approaches, on the other hand, had to bring the size of the data to a manageable level by either sub-setting or by specialized methods such as dimension reduction – often resulting in significant information loss.
- We observed that the space of dimensions is highly heterogeneous. We have seen that this heterogeneity is due to several factors such as the interrelations within the data dimensions, the problematic nature of the data,

and various characteristics specific to the procedures that generate the data itself. Although we have seen such heterogeneity in almost all of the datasets we have analyzed, methods employed by the analysts often neglect these properties and treat all the dimensions equally. With our methods, we detect, analyze, and utilize these heterogeneous properties within the dimensions. As demonstrated by the several use cases and by the feedback from the domain experts, our contributions to the analysis process improve the productivity of the analyses significantly.

- One of the analysis strategies we developed in this thesis involved the treatment of local structures within the analysis. In a typical analysis session, analysts would discard most of the underlying structures and do not perform any local operations on the data. We observed that considering local structures through our methods improved the interpretability of analyses considerably. One example of this was our study carried out together with neuropsychology and biomedicine experts (refer to Section 4.1 in Chapter 4). In this study, we made use of the inherent structure of the data by performing local operations on each of the brain segments. We have seen that such a structured approach to data analysis enabled the domain experts to easily make understandable local observations, which they can later on merge to build a big picture of the data.
- We have seen that considering the characteristics of the dimensions in different stages of the data analysis guides the user in giving the appropriate decisions when applying automated methods on the data. For instance, during the pre-processing of the data, the right normalization method can be chosen based on a visual investigation of the dimensions. Similarly, the careful inspection of the characteristics of the dimensions enables the analyst to check for the different assumptions regarding the data, e.g., visually evaluating for normality before a dimension reduction is applied.
- In our methods, we have made it possible to access the whole raw data and the analysis results at all times during the analysis. Being able to access the whole data domain enables the analyst to switch the focus of the analysis from one subset to the other. This enhancement to the analysis practice has been regarded to be important by our collaboration partners since it enables them to quickly shift the focus of the analysis in a single analysis session. This also saves them from performing tedious steps to subset the data prior to the analysis and in addition enables them to compare different results in a common framework.
- In the more conventional practice, the analyst starts with one or more hypotheses in mind that are built based on his/her prior knowledge on the field. It continues with the usually time-consuming steps to evaluate these ideas one-by-one. In our approach, we suggest that an analyst quickly evaluates several ideas through the visual analysis cycle and only spends

additional effort on those ideas that look promising in the first place. This new approach amounts to a more efficient mechanism than spending a very long time on each and every single idea.

Along this line, being able to *quickly prototype ideas* is one of the main strengths of our interactive visual methods. The analysts can easily confirm existing knowledge, build new insight, and evaluate different ideas to generate new hypotheses. This explorative stage should ideally be followed by referring to more robust and advanced automated mechanisms to confirm the validity of these hypotheses.

- We have seen that it is highly important to communicate the certainty of the findings. The analyst should be given indications of whether the finding is reliable or needs further refinement. One way to achieve this is to employ quality measures and evaluation methods that have proven to be useful in other domains related to data analysis, i.e., statistics, data mining. One example in this thesis is the use of *silhouette values* (Chapter 3, Section 3.2) to evaluate the quality of cluster memberships. Moreover, we incorporated hypothesis testing results to communicate the significance of differences in our difference view (Chapter 3, Section 3.1). Both of these additions provide immediate visual feedback and improves the rate that the explorative process converges to useful insight.
- One feature that we consider very important within this thesis is to be able to use and compare the results of different algorithms or several runs of a single algorithm. This is a very powerful mechanism to overcome the limitations and parameter dependence of automated methods. Moreover, incorporating several methods and measures makes the whole analysis more resistant to problems in the computations. In this thesis, for instance, we have demonstrated the importance of such a capability while applying clustering on molecular dynamics simulations (Chapter 4, Section 4.2).
- We observed that moderating the interaction pace of visual analytics applications is very critical to maintain a healthy dialogue between the human and the computer. Most of the visual analytics applications do not fully address the different human factors. However, as the complexity and the size of datasets gets more challenging, it is becoming more and more important to consider different aspects of the human user in data analysis.

Future work

The lessons learned listed above and our exchanges with analysts motivates us to carry this research even further. Below we discuss a number of possible directions to extend this work.

One aspect that needs to be investigated further in the integration of interactive and automated methods is the *issue of usability.* Our solutions require significant literacy in statistics and skills in using different computational methods. Our collaborators mentioned that these requirements can lead to a demanding learning curve. We consider this to be very important in order to increase the impact of visual analytics applications. Along this line, we plan to incorporate mechanisms to improve the interpretability and usability of our methods. Possible methods could be to employ smart labeling and annotation, creating templates that analysts can follow for easier progress, and computationally guided interaction mechanisms where automated methods are integrated seamlessly.

Moreover, practical methods to evaluate the insight gained through integrated and interactive systems needs to be developed. Such mechanisms could be utilized to validate our improvements to the analysis process based on the human-time constants. Moreover, such methods will improve how we design the interaction and the visualizations within the analytical processes. This motivation is also underlined by North [139] where he emphasizes the importance of devising direct methods to evaluate visualizations.

Another topic that needs further attention is to address the uncertainty within the analysis process. We plan to implement mechanisms that communicate the reliability of the observations made through interactive visualizations, e.g., what happens to my observation if I move my selection slightly along the x-axis? If such questions are addressed, interactive and visual methods could easily place themselves in the everyday routine of analysts that require precise results.

As more technical improvements of our work, we plan to extend the dual analysis framework to operate on several datasets simultaneously. Currently the dual analysis approach operates over the dimensions of a single data table, however in many fields an analysis of several datasets is becoming highly needed. We consider that there is great potential in extending our approach to a third space in addition to the items and the dimensions space – the *dataset space.* We briefly investigated how such an improvement is possible by integrating our methods within the multi-dataset analysis framework Caleydo (Chapter 3, Section 3.2). However, the current functionality is limited in terms of the analysis of inter-dataset relations and there is the need to formally define the operations in this third level.

Part II

Papers

Paper A

Brushing Dimensions – A Dual Visual Analysis Model for High-dimensional Data

Cagatay Turkay[1], Peter Filzmoser[2], and Helwig Hauser[1]

[1]Department of Informatics, University of Bergen, Norway

[2]Department of Statistics and Probability Theory, Vienna University of Technology, Austria

Abstract

In many application fields, data analysts have to deal with datasets that contain many expressions per item. The effective analysis of such multivariate datasets is dependent on the user's ability to understand both the intrinsic dimensionality of the dataset as well as the distribution of the dependent values with respect to the dimensions. In this paper, we propose a visualization model that enables the joint interactive visual analysis of multivariate datasets with respect to their dimensions as well as with respect to the actual data values. We describe a dual setting of visualization and interaction in *items space* and in *dimensions space*. The visualization of items is linked to the visualization of dimensions with brushing and focus+context visualization. With this approach, the user is able to jointly study the structure of the dimensions space as well as the distribution of data items with respect to the dimensions. Even though the proposed visualization model is general, we demonstrate its application in the context of a DNA microarray data analysis.

This article was published in *IEEE Trans. Visualization and Computer Graphics*, 17(12):2591–2599, 2011.

1 Introduction

The rapid development of increasingly powerful computers and the improving methods for data acquisition lead steadily to more challenging datasets with respect to their analysis. On the one side, the large number of items in datasets is challenging. On the other side, the increased complexity of datasets, in particular in terms of larger numbers of expressions (dimensions) per item, is posing highly interesting questions. Both challenges have been addressed for many years in statistics research, data mining, machine learning, and visualization. With respect to related visualization research, and in particular with respect to recent activities in visual analytics, a somehow skewed picture appears. There is ample work on items-based visualization approaches, where the data items in a dataset are represented either explicitly or implicitly in the visualization. On the contrary, there is much less work, which addresses the dimensions as first-order objects of the visualization. Understanding a dataset's dimensions, however, such as its intrinsic dimensionality, for example, is often also important for an effective analysis of the data. Accordingly, we see a pressing need to also support this task (understanding the dimensions of a dataset) with means of interactive visual analysis.

In the context of this paper, dimensions are considered as a mixture of dependent and independent variables. An example would be a cars dataset about a number of cars (as the items), each of which being associated with several values, such as gas mileage, price, engine size, i.e., the dimensions in this data. Analysts often use multivariate statistical analysis (MVA) techniques, for example, principal component analysis (PCA), linear discriminant analysis (LDA), clustering, etc., to understand the underlying relations between the dimensions and the data items [100]. However, as the dimension count gets larger, and noisy values in dimensions (e.g., outliers) influence the represented information, the output of these methods becomes harder to interpret and occasionally less reliable [2].

Also it is often so that high-dimensional datasets come with a number of dimensions which are more important in order to explain the underlying phenomena than others. Datasets are also often populated with dimensions which are derived from each other or which carry no additional information about the phenomenon being explored (but are included for other reasons, e.g., their own absolute scale). If we refer to the cars dataset again, examples of derived dimensions could be the price of the same car in different currencies. Analysts are often, for example, interested in discovering the *intrinsic dimensionality* of the data which corresponds to the minimum number of dimensions which can explain the relations in the data [105]. Accordingly, multivariate statistical analysis is often preceded by a *dimension reduction* phase where the main goal is to create a lower dimensional space [100] that still contains the essential information from the original dataset. One of the most popular methods for dimension reduction is principal component analysis (PCA). PCA can be used to create a lower-dimensional representation of the data that still captures most of the variance in the data. However, the resulting dimensions are usually difficult to interpret. In this respect, there are studies in statistics research to improve the interpretability of the results by filtering the dimensions prior to PCA [54]. These studies try to create sparse representations of principal components by identifying and leaving out "redundant" dimensions that do not contribute to the overall variance of the dataset [54].

Another important consideration in most of the MVA methods is their assumptions on the underlying data distributions. Popular MVA methods such as PCA or regression analysis, for instance, assume that the data are normally distributed with respect to their dimensions. However, many of the high-dimensional datasets in practice fail to fulfill this assumption, for instance, due to outliers. Handling of outliers and observing the descriptive statistics of dimensions to assess their normality is crucial when considering the reliability of MVA results. This aspect of MVA is, therefore, subject to many studies under the name of "robustness" in statistics. Such studies try to improve the resistance of analysis methods to outliers and try to make them less dependent on the distribution of dimensions [58].

There are several application fields where the relations between the items are at least as important as the relation between the dimensions, such as DNA microarray data analysis [47]. In such areas, methods that operate on items and dimensions at the same time are of great potential interest. Most of the existing MVA methods, however, operate either on items or on the dimensions and the joint interpretation of these separate results is not always straight forward. Accordingly, there is a need for methods that enable the joint analysis of items and dimensions in such datasets, also by considering the effects of dimensionality and variable distributions.

Interactive visual analysis has been used extensively to visualize high-dimensional data and MVA results [65]. The common approach in the visual analysis of high-dimensional data is to visualize the items as opposed to different dimensions in linked views and to support the discovery of relations between expressions by means of interaction. This approach also provides an aid to derive hypotheses on the intrinsic dimensionality of the data. Unless supported by MVA tools, however, interactive methods alone fail to provide a comprehensive insight on the data, especially as the dimension count gets larger and as the relations between the dimensions become more complex. A more "fruitful" analysis requires the integration of computational tools in the visual analysis cycle as suggested, for example, by Keim et al. [112]. Moreover, an interactive visual analysis solution should also enable the exploration of the dimensionality of the data by considering the "redundancy" and "robustness" constraints throughout the analysis.

In this paper, we now present a visual analysis model where the analysis of items and dimensions is carried out in two linked spaces, namely *items space* and *dimensions space*. We utilize the current knowledge about the interactive visual analysis of data items to also enable the interactive visual analysis of data dimensions. In our model, we suggest a setting of linked views, where the analyst interacts with the items in items space, e.g., *by brushing items*, and with the dimensions in dimensions space, f.i., *by brushing dimensions.* Firstly, our model aims to provide more insight with respect to the *intrinsic dimensionality* of the dataset based on interactions in both spaces. By selecting useful combinations of dimensions and leaving out redundant ones, the analyst can improve the MVA results according to prior knowledge and interpretation. Secondly, by interacting with the data items, the analyst has the opportunity to relate data subsets to MVA results. With such interactions, the analyst can modify the distribution of items, e.g., by removing outliers, and observe the change with respect to MVA results.

In the following, we first exemplify our approach in the context of an illustrative example (after having discussed related work), before we then present a model for a dual visual analysis of high-dimensional data. We describe how the data analysis is performed through transformations and how brushing and focus+context visualization is integrated in the model. Specifically, the contribution of this paper are:

- a novel method for the joint and linked analysis of items and dimensions of high-dimensional data,
- a formal model which describes the transformations, brushing operations, and focus+context visualizations in the dual analysis framework, and
- a set of procedures and guidelines to preform such a dual visual analysis of high-dimensional data.

2 Related Work

Interactive visual methods have been used extensively in the analysis of high-dimensional data. An overview of related studies is available in surveys by Wong and Bergeron [211] and by Fuchs and Hauser [65]. Coordinated multiple views have proven to provide insight into high-dimensional datasets by means of linking and brushing in views which display different aspects of the same data [173]. Examples of such approaches are realized in the XmdvTool [202], Polaris [178], and in ComVis [131]. Many efforts have been made to explore multivariate data with visualization. Jänicke et al. [94] propose the brushing of multivariate data after a projection to an attribute space which can be visualized in a 2D view. In cross-filtered views [203], Weaver enables the exploration of relations between dimensions by cross-filtering data values from different views.

In order to cope with the complexities as induced by a higher number of dimensions, dimension reduction methods have been integrated into the visual analysis pipeline. In VHDR [213], Yang et al. group dimensions in a hierarchy and create lower-dimensional spaces using representative dimensions. Their method also provides opportunities to manually reduce dimensions. Jeong et al. [96] provide a set of interaction mechanisms that operate on PCA results. With modifications of the parameters of PCA, it is possible to observe changes in the PCA results.

Visual analysis methods have been used jointly with a number of computational methods. Fuchs et al. [66] integrated machine learning with interactive visual analysis to support hypothesis generation. In MDSteer [210], Williams and Munzner present a steerable multidimensional scaling computation where it is possible to steer the analysis to the areas which are interesting for the user.

A number of different statistical tools have been integrated into visualization systems. Guo et al. [76] enable the interactive exploration of multivariate model parameters. They visualize the model space together with the data to reveal the trends in the data. Gosink et al. [70] use a query-driven visualization with a statistics-based framework. They utilize query distributions to estimate trends and features. Correa et al. [36] consider the uncertainties that arise while transforming the data. These uncertainties are integrated in the visualization to support the interpretation of statistical analysis results.

There are a number of studies where the joint analysis of data items and dimensions have been investigated. In the Rank-by-Feature framework [168], Seo and Shneiderman rank the relations between dimensions according to user-defined statistical features. The authors present how a joint analysis framework is useful to steer certain statistical processes. However, their approach is limited to computations on the whole dataset. In our model, we enable the interactive exploration and comparison of statistical features under different subset selections. Moreover, we treat dimensions as any other data item and present them with visual entities in the proposed dimensions space. The successful utilization of joint analysis of two different spaces in the context of parameter space navigation is presented by Berger et al. [17]. In another study, Andrienko et al. [9] describes how a dual analysis scheme is utilized in spatio-temporal datasets. Their approach involves the dual analysis of spatio-temporal datasets over spatial distributions and temporal variations. Unlike our model, their approach is specific to spatio-temporal datasets. In our model, we utilize a similar dual analysis idea for the general case of high-dimensional datasets.

Another important related work is the Value and Relation (VaR) display by Yang et al. [212]. In this work, the authors represent the dimensions with glyphs, which are projected to a 2D layout using multi-dimensional scaling. In this work, the actual data items are only represented through glyphs and the interactive analysis of items together with dimensions is not possible.

Another important study in relation to our model is by Kehrer et al. [106], where the authors compute statistical moments from the data and plot data aggregates as opposed to these moments. In their work, a set of scatterplots and transformations between them are defined. Their framework provides mechanisms to explore trends and outliers in aggregated datasets. This framework displays the benefits of using statistics in the visual analysis of data aggregates

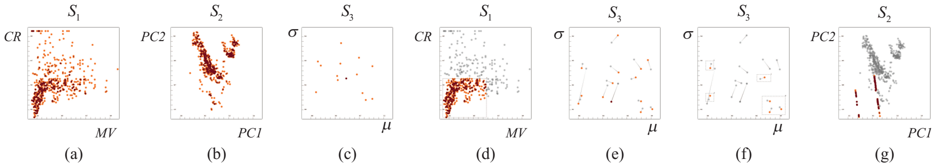


Figure 1: An illustrative example for a joint visual analysis of items and dimensions of the "Boston Housing Prices" dataset. Three scatterplots are set up first: a) S_1: house prices (MV) vs. crime rate (CR), b) S_2: the first two principal components ($PC1$ vs. $PC2$), c) S_3: mean (μ) vs. standard deviation (σ) values for all the dimensions of the data. d) The main trend in the data is selected in S_1. e) μ and σ values are re-computed for the selected items and changes are visualized in S_3. f) Dimensions that deviate less are selected for a re-computation of the PCA. g) PCA results (before and after) are visualized in a F+C style.

together with data items. In our work, we define a more general model which operates on high-dimensional data using statistical analysis methods together with statistics computations. With our model, we extend the current approach to the visual analysis of high-dimensional data with the idea of a joint and linked analysis of data items and dimensions.

Throughout this paper, we utilize a number of multivariate statistical analysis methods such as principal component analysis (PCA) and linear discriminant analysis (LDA). PCA is a popular, unsupervised dimension reduction method that is widely used in multivariate statistical analysis [100]. The goal of PCA is to create a lower-dimensional projection of an originally high-dimensional dataset while preserving as much of the variance in the data as possible. PCA creates an orthogonal coordinate system where the axes are called principal components (PC). These PCs are all linear combinations of the original dimensions where the weights are referred to as the *loadings*. LDA is a supervised dimension reduction method that finds a linear combination of the original dimensions by considering class labels [100]. LDA attempts to maximize the class discrimination while reducing the dimensionality of the data. LDA is used as a classifier or as a dimension reduction method. One important point is that both methods assume the data to be normally distributed.

In addition to PCA and LDA, we also make use of certain descriptive statistics, namely the mean (μ), the standard deviation (σ), the skewness ($skew$), the kurtosis ($kurt$) and interquartile-range (IQR). μ can be estimated by the average of the values in the data, σ is the standard measure of variability, $skew$ indicates if a distribution is centered, or not, while $kurt$ indicates the peakedness of a distribution and IQR is a robust statistics that also describes the variance of a distribution.

3 An Illustrative Dual Analysis Example

Before we present our more formal model further below, we first describe an illustrative example where a visual analysis of data items is carried out together with a visual analysis of the dimensions. Our aim here is not to already provide a comprehensive guide, but to informally demonstrate the basics of our dual analysis model.

As also generally in this paper, we assume that our datasets come in a tabular form with n items (rows) $x_j \in \Omega$ (set of items), each of which with values in m dimensions (columns) $d_k \in \Delta$ (set of dimensions). In the following, we denote the k^{th} value of the j^{th} item as $x_{j,k}$. For this first illustration, we study the well-known 'Boston Neighborhood Housing Prices' dataset [80]. This dataset contains information gathered by the U.S Census Service to understand the relation between housing prices and other factors in the area of Boston, Massachusetts. It consists of 506 samples x_j and 14 dimensions d_k (i.e., $|\Omega| = 506$, $|\Delta| = 14$). Some of the dimensions that we refer to later are: 'median value of owner-occupied homes' (MV),'crime rate by town' (CR), 'proportion of houses built before 1940' (AG) and 'proportion of lower status of the population' (LS).

In our analysis, we utilize PCA to understand the intrinsic dimensionality of this dataset. To reduce the effects of outliers on PCA, we analyze the data to determine outlier-free dimensions. We compare PCA results based on all dimensions and those computed for only selected dimensions, in order to achieve a better interpretation of the analysis results.

To enable the comparability of dimensions, the analysis starts with a normalization of the dimensions. To normalize the dimensions, we apply linear scaling to the unit interval in this case. We then estimate the mean (μ) and standard deviation (σ) of all the columns (dimensions), in order to get a first impression of the included data distributions. We apply PCA to all the dimensions and project the data onto the first two principal components ($PC1$, $PC2$). We continue with the visualization of the items in a scatterplot S_1 (Figure 1-a) with axes CR and MV and another scatterplot S_2 (Figure 1-b) with axes $PC1$ and $PC2$. Additionally, we plot the μ and σ values of all dimensions in a scatterplot S_3 (Figure 1-c).

We then start the interactive analysis by brushing (selecting) a subset of items in S_1. This brush leaves out the larger values of MV and CR and selects the items which (roughly) amount to the main trend in the data (Figure 1-d). As a next step, the μ and σ values are estimated (automatically) for the selected items and sent to S_3. As a result, S_3 gets updated to show the dimensions' statistics with respect to both the items selection as well as with respect to all of the items (Figure 1-e). The μ and σ values corresponding to the selected subset are highlighted (with orange color), while the original μ and σ values (corresponding to the entire dataset) are presented as reference (in gray). The two points in the scatterplot which correspond to the same dimension (entire dataset vs. selected subset) are connected with a tapered line to ease their identification. In Figure 1-e, we see that while the values for some of the dimensions changed prominently, some of them are not much affected by the selection. A simple first interpretation of the resulting visualization is that the dimensions that did not deviate so much due to the selection, possibly can be considered to be less sensitive to non-standard values of MV and CR. We then select the most "stable" dimensions in S_3 and PCA is applied automatically using only the dimensions selected in Figure 1-f. We then project all the items to the newly computed principal components and send the resulting values to S_2. Through a focus+context visualization of the two different projections of the items in S_2, we can clearly see that the projection results changed dramatically (Figure 1-g). An interesting split into two groups with respect to the new PC1, for example, can be observed. In such an explorative setting, the analysis may not always converge to the mathematically best-possible result. However, through the selection of suitable statistics and the use of interactive brushing, the analysis leads to both additional insight on the data and results that are easier to interpret. Guidelines for a robust analysis process are provided in Section 6.

The above presented short illustration brings up new opportunities for the analysis of high-dimensional data. Such a dual visual analysis of both items and dimensions leads to a novel perspective on looking at high-dimensional data. In the following section, we formalize this dual analysis idea in the form of a model by defining the underlying linking&brushing and focus+context (F+C) visualization mechanisms.

4 The Dual Analysis Model

Analysts are often faced with high-dimensional data which comes in a tabular form where items are rows and dimensions are columns. In conventional visual analysis approaches that involve multiple coordinated views, items are visualized using visualizations like scatterplots, histograms or parallel coordinates. In such visualizations, the items are plotted in the views as opposed to the dimensions of the data. The visual analysis of data items is often carried out using linking&brushing and focus+context visualization. Our dual visual analysis concept builds upon these conventional practices and proposes the visual analysis of data in two linked spaces, namely in *items space I*, and in *dimensions space D*. With items space we refer to a visualization domain where each visual entity in a visualization corresponds to a data item. In the dimensions space, however, each visual entity represents a dimension of the data. To illustrate, if we visualize the housing data in both of the spaces, using scatterplots, a point in items space corresponds to a single house, whereas in the dimensions space, a single point represents a dimension, crime rate by town, for instance. By separating the

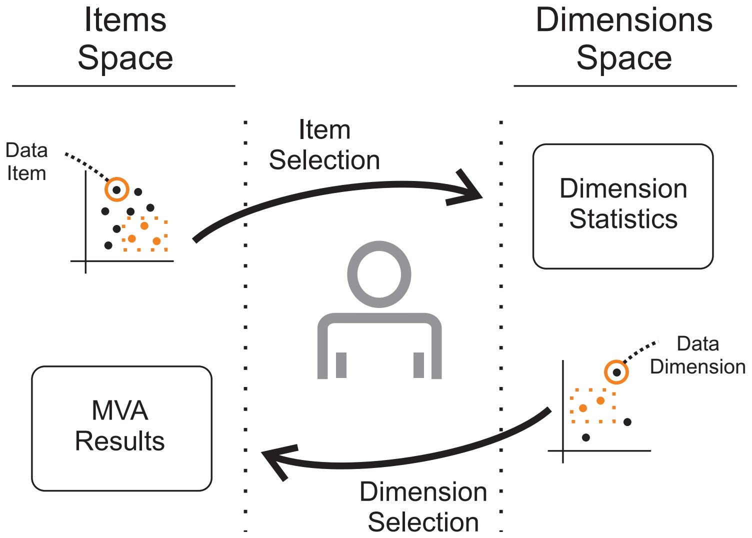


Figure 2: The dual analysis model sketched. Visual analysis is performed over two spaces, items space and dimensions space. Visual entities correspond to items in items space and dimensions in dimensions space. Analysis advances iteratively by selecting items and dimensions. The interactions enable the joint and linked exploration of dimension statistics and multivariate analysis (MVA) results.

visual analysis space into two, we provide opportunities for the joint and parallel analysis of items and dimensions.

A conceptual sketch of our model is depicted in Figure 2. Here, items space includes the visualizations of MVA results (such as a projection on principal components). The analyst iteratively performs item and dimension selections in order to observe the changes in dimension statistics as well as MVA results. The duality in the model is achieved by linking the visualizations in the two spaces. In order to fully accomplish this link, we formulate *brushing* and *focus+context* visualization mechanisms, as well as transformations which are needed to establish the relation between the two spaces.

4.1 Data Transformations

The iterative analysis of items and dimensions is at the core of our model. During a typical iteration, the focus of the analysis moves from one space to the other. In order to achieve the transitions between items and dimensions space, our model requires a set of data transformations.

From dimensions space D to items space I: The basis for the first type of transformations relates to the MVA methods that operate on the dimensions Δ. Such methods are here denoted by f. We generalize transformations f to operations that create l new data dimensions when applied. In the illustrative example in Section 3, PCA is an example of such an f transformation. Throughout the iterative analysis loop, the i^{th} transformation of data through f is defined as: $T_D^i(f) : \Delta' \xrightarrow{f} \Delta^i$ where $\Delta^i = \{d_{c+1}, ..., d_{c+l}\}$ with any d_a being a full new column $d_a = \{x_{1,a}, ..., x_{n,a}\}^T$ and $c = \sum_{t=0}^{i-1} |\Delta^t|$. Note that, in these transformations, all the items are projected onto the new dimensions and $\Delta' \subseteq \Delta$ represents a selection of dimensions of the data before the transformation. At a certain point in the iterative loop, where the analyst have made y of these transformations, the final set of dimensions is denoted as $\Delta^+ = \{\Delta^0, ..., \Delta^y\}$ with $\Delta^0 = \Delta$, i.e., the original data dimensions.

Although we exemplify PCA as one f method, it can also be any other MVA tool which creates a mapping of the original dimensions. It is possible to consider methods like multidimensional scaling (MDS) and factor analysis (which are other dimension reduction techniques), clustering (which maps the data items to class labels), and LDA (which maps the data items to known classes) [100].

As an initial transformation, which usually precedes the statistical analysis as well as the visualization, we normalize the dataset so that values in all the dimensions are quantitative and comparable. Normalization also ensures that all of our dimensions are suitable for visualization in a scatterplot, histogram, etc. Moreover, it is an essential step for most of the MVA processes [147]. This normalization step is denoted with $T_D^1(N)$ where N is a normalization method, such as linear normalization to the unit interval or z-standardization [147]. The results of $T_D^1(N)$ is denoted with Δ^1 where $\left|\Delta^1\right| = |\Delta|$.

From I to D: We use transformations s to iterate from items space to dimensions space. Examples of s can be descriptive statistics or an aggregation of data items. Here, we mainly consider statistics as s. If we consider σ as s, the result of the transformation are the σ values for each and every dimension in the data. In the r^{th} iteration of the analysis the transformation which computes g new values per dimension using s is defined as: $T_I^r(s) : \Omega' \xrightarrow{s} \Omega^r$ where $\Omega^r = \{x_{e+1}, ..., x_{e+g}\}$ with any x_a being a full new row $x_a = \{x_{a,1}, ..., x_{a,m}\}$ and $e = \sum_{t=0}^{r-1} |\Omega^t|$. Here, $\Omega' \subseteq \Omega$ represents a selection of items. In the course of the analysis, the analyst can make z of these transformations where she produces the final set of computed values $\Omega^+ = \{\Omega^0, ..., \Omega^z\}$. To generalize, regarding the set of possible s functions or statistics, it is possible to consider descriptive statistics such as mean, variance, skewness, kurtosis and more elaborate values like statistical test results or robust estimates.

The selection of dimensions Δ' and items Ω' is formulated through a degree-of-interest (doi) mechanism. Similar to fuzzy set definitions, we define $\Delta' = (\Delta, doi_\Delta)$ and $\Omega' = (\Omega, doi_\Omega)$ where doi_Δ and doi_Ω are mappings to define selection degrees. In the case of binary selections, where an item is either selected

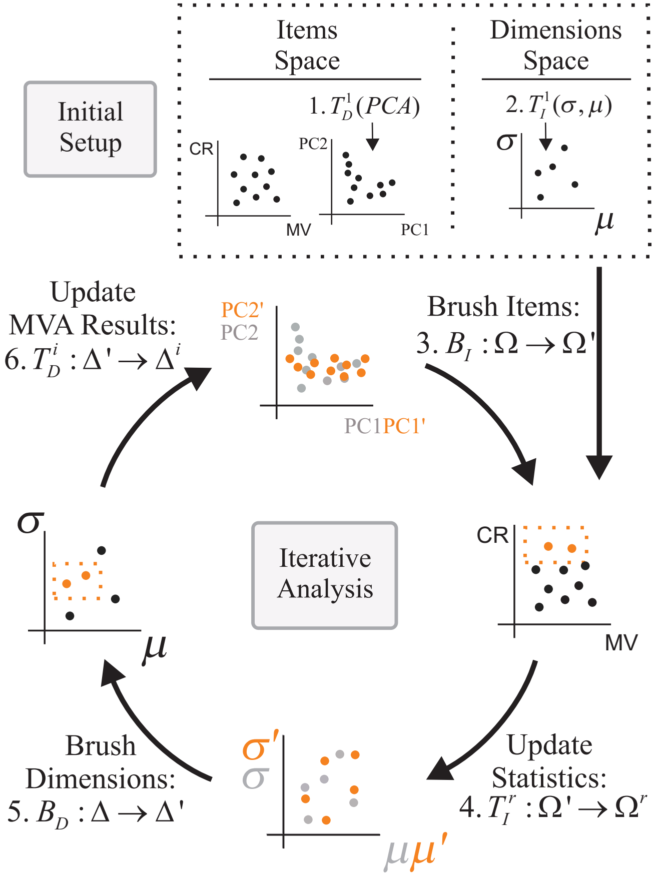


Figure 3: Items space views both visualize normalized dimensions, e.g., CR or MV in housing data, and derived dimensions, e.g., PCA results $PC1$ or $PC2$. Dimensions space views visualize dimensions as opposed to statistics, such as μ or σ. Here, the initial setup is done by computing PCs (1), μ and σ (2). Brushes from items space (3) triggers F+C visualizations in dimensions space by going through transformations (4). Similarly, brushes from dimensions space (5) updates the MVA result visualization through transformations (6). This interactive loop continues iteratively by modifying the selections on both sides.

or not, selections are defined as $doi_\Omega : \Omega \rightarrow \{0, 1\}$. In the case of continuous *doi* values, where items are selected to a certain degree, selections are defined as $doi_\Omega : \Omega \rightarrow [0, 1]$. Such a continuous selection mechanism can be achieved through smooth brushes [43]. The addition of smooth brushes brings the possibility of weighing the dimensions prior to a dimension reduction operation, for instance.

4.2 Brushing & Focus+Context Visualization

The conventional visualization of high-dimensional data in items space is achieved by plotting the items with respect to the original dimensions and the derived dimensions, i.e., Δ^+. The visualizations in dimensions space, however, visualize dimensions Δ as opposed to the statistics computed by $T_I(s)^r$ operations, i.e., Ω^+. We denote the views in items space with V_I and views in dimensions space with V_D. It is worthwhile to mention that the columns of our dataset are treated as rows in dimensions space. Accordingly, our approach can also be thought of as transposing the dataset and performing the visual analysis using a different perspective in dimensions space. In the illustrative example in Section 3, S_1 and S_2 are examples of V_I and S_3 is an example of V_D.

We follow the conventional linking&brushing mechanism between the views that are in the same space; i.e., when certain items in a V_I are brushed, the same items are highlighted in other V_Is using a focus+context visualization and the same mechanism works also for V_Ds. In order to define the links between views from different spaces, we extend this mechanism by handling the brushes through the f and s transformations. The transitions between the two spaces and illustrations for the associated F+C visualizations scheme are illustrated in Figure 3.

A brush in V_I is defined as $B_I : \Omega \rightarrow \Omega^{'}$ where $\Omega^{'} \subseteq \Omega$. In order to transfer B_I to dimensions space, brushed items $\Omega^{'}$ are transformed by $T_I(s)^r$ using the current s. The resulting values Ω^+ update visualizations in dimensions space. An example of such a brushing operation can be seen in Figure 1-d,e. Here, σ and μ values (i.e., s transformations) are re-computed for the selected items in S_1 and the computations update S_3.

A brush in V_D is defined as $B_D : \Delta \rightarrow \Delta^{'}$ with $\Delta^{'} \subseteq \Delta$. B_D is transferred to items space by going through the transformation $T_D(f)^i$. And, the resulting $\Delta^{'}$ update V_Is accordingly. An example for this type of operation can be seen in Figure 1-f,g. Here, the dimensions are selected in S_3 and the selection of dimensions is an input to the PCA operation.

In a typical F+C visualization, the common interpretation of focus are the selected items and the context is the rest. In our model, we slightly extend this definition of F+C visualization. Focus and context are two different visualizations of the same items, that are computed using different subsets of the dataset. The results of the last transformation (f or s) is set as the focus and those of the preceding one as the context. Notice that each point in a scatterplot is drawn twice, once with the old and once with the new value. Here, we follow a simple strategy to show the results. If the point count is large, we plot focus and context in different colors (Figure 4-a). If the point count is small, we additionally

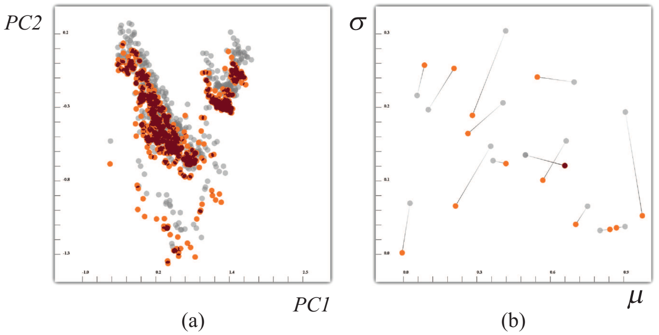


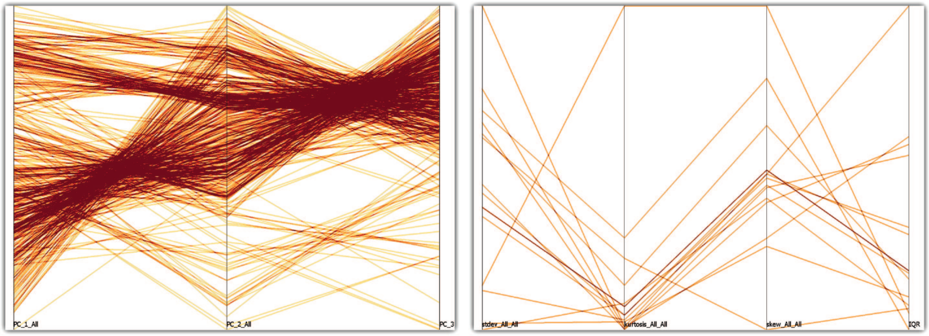
Figure 4: Focus+context visualizations in scatterplots of two different PCA results (a) and of two sets of statistics σ, μ (b). The recomputed values are in focus after the selection, and the values from before the selection are provided as context. Depending on the point count, two different styles are employed (with and without lines).

connect the related points with a tapered line (Figure 4-b). Although this simple solution is adequate for illustrative purposes in this paper, one should think of more intelligent ways to achieve comparative visualizations, e.g., difference views [120].

One important point to mention, also, is that, in the F+C visualizations of the first type of views, the focus is computed as a "lazy evaluation", i.e., the focus of a view, is linked to a brush and it is computed automatically as the brush moves. This approach is necessary for the sake of interactivity in the model. Additionally, the context of the views can be updated at any point throughout the analysis. With such an extension, it is possible to compare the statistics and analysis results of any different item-dimension subsets.

4.3 Extensions to the Model

It is possible to extend the proposed dual analysis method to also incorporate different visualization techniques, e.g., parallel coordinates plots (PCP). While lines in a PCP represent data items in items space, they represent dimensions in dimensions space. Accordingly, axes of a PCP in items space are the original dimensions of the dataset and they correspond to different Ω^+ in dimensions space. An example of these dual PCPs can be seen in Figure 5. In order to visualize the deviations and employ our dual focus+context approach in a PCP, comparative visualization methods, like Temporal Parallel Coordinates [97] can be utilized. Another possible extension is to employ glyphs as the visual entities in dimensions space [212]. One can think of glyphs where each visual channel represent different Ω^+ values.



(a) (b)

Figure 5: The proposed dual analysis extended to parallel coordinates plots (PCP). a) PCP from items space visualizing items over the first three principal components. b) PCP from dimensions space visualizing σ, $kurt$, $skew$ and IQR values for the dimensions.

In its current state, the model is designed for datasets that come in a 2D tabular form. However, it is possible to extend the model to 3D data tables, e.g., to datasets where the third dimension is time. In the dual analysis of such datasets, visualizations in items space are conventional visualizations of temporal data, i.e., each data item is represented by a curve over time in a function plot. In dimensions space, however, each curve represents a dimension over time. We perform s transformations on each temporal dimension and visualize the results in a function plot in dimensions space. In Figure 6, this mechanism is illustrated. Here, we visualize measurements from a weather station in Bergen, Norway. The dataset contains daily measurements, such as temperature, pressure, precipitation, for all the years between 2000 and 2010. In Figure 6-a, each curve represents the temperature values for one year. On the other side, in dimensions space, we compute σ values for each dimension over time. And the result is a curve for each dimension plotted against σ values as seen in Figure 6-b.

5 Prototype Implementation of the Model

We implemented our model in an interactive visual analysis environment where we enable linking&brushing and focus+context visualizations of data in scatterplots and other views. We implemented two types of scatterplots, with two types of F+C visualization, as already discussed above. Our aim with the prototype implementation is to showcase the utilization of the system using simple visualization solutions.

Our implementation utilizes composite brushing, as proposed by Allen and

Ward [130], as the underlying brushing mechanism. In this mechanism, each brush is combined with existing brushes by a Boolean operator op with $op \in \{\cup, \cap, \neg\}$, where $\cup$ represents the union, $\cap$ represents the intersection and $\neg$ represents the not operator. To ensure an easier utilization of different types of views, the visualization space is physically divided into two, one to show items space and the other one for dimensions space. Additionally, to include a wider range of MVA tools into the system, we integrate the R statistical computation package into our system [184].

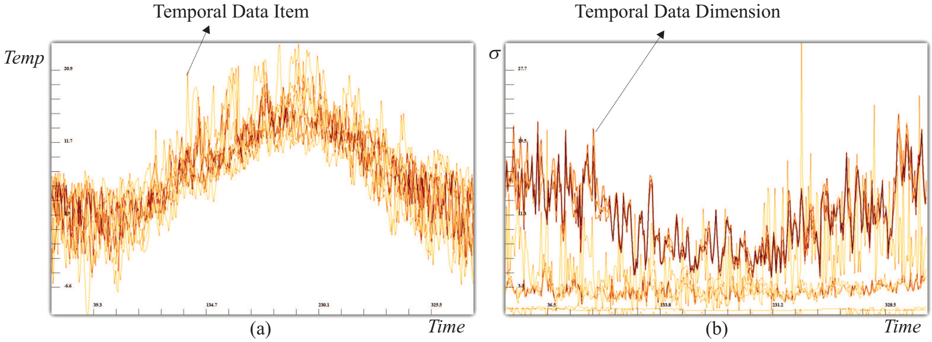
6 Dual Analysis Procedures

The dual analysis process provides a number of opportunities in the visual analysis of high-dimensional data. Here, we provide a guide for selecting and using the transformations and visualizations in the proposed dual setting.

6.1 Selecting Transformations

Depending on the type and the goal of the analysis, the analyst determines the multivariate statistical analysis tools and statistics to utilize. The selected tools and statistics then correspond to the transformations in our model. In Table 1 we provide a non-exhaustive list of common MVA tools f and statistics s that are suitable for the dual analysis scheme. Note that the dual analysis model is not specific to any of these methods.

One important type of f transformations are unsupervised dimension reduction methods such as PCA and MDS. The reliability of the results of such methods



(a) (b)

Figure 6: A dual analysis of temporal data. a) An items space visualization of daily average-temperature values from a weather station in Bergen, Norway. b) A dimensions space visualization where each curve corresponds to a dimension. The values are σ values that are computed for each time-step.

Table 1: Possible multivariate statistical tools (f transformations) and corresponding statistics s for the dual analysis setting

Analysis	f	s
Dimension reduction (unsupervised)	PCA, MDS	loadings, mean, variance, median, skewness, kurtosis, IQR
Dimension reduction (supervised)	LDA, SVM	variance, information theory
Finding groups in data	Clustering	mean, variance, median, IQR

depend on the normality and "outlier-freeness" of the data columns. Additionally, to improve the interpretability of the results, redundant dimensions should be discarded. Principle component loadings, σ and the interquartile range (IQR) can be used to assess the dimensions' redundancy while μ, σ, skewness and kurtosis can be used to evaluate normality and the existence of outliers. Similar s transformations are preferred for clustering, where the quality of the results is affected by a high number of dimensions as well as outliers in the data.

In supervised dimension reduction methods like LDA and Support Vector Machines (SVM), the normality of the data is not required. However, the selection of dimensions is crucial with respect to the quality of the results, also. In order to determine important dimensions, σ, IQR or information theoretic measures can be utilized [77].

In all of these methods, filtering dimensions prior to the analysis both improves the quality and interpretability of the results. Therefore, dimensions need to be evaluated in terms of their variance (saliency) and/or entropy [77]. Dimensions that are poor in information content, i.e., with a low variance, low entropy, near-zero loadings in PCs, can be marked as "redundant" and left out from the analysis.

6.2 The Analysis Process

In the following, we provide a task-based guideline to carry out an analysis in the proposed dual framework:

- To understand the relations between dimensions: A subset of items are selected first. As a result, the changes in s values in dimensions space reveal the correlation between dimensions with respect to the selections. Larger deviations in s values indicate a higher correlation.

- To explore the dimensions that determine the main trend or the outliers in the data: Items that correspond to the main trend or outliers are selected in a lower-dimensional projection of the data. Deviations in dimensions space reveal such dimensions.

- To leave out/select dimensions: Dimensions are evaluated in terms of the information they contain through the use of certain s such as σ, principal component loadings and entropy.

We follow these guidelines and go through the steps of a detailed analysis process that is similar to the one we presented earlier in Section 3.

In this analysis, we aim to explore the relation between dimensions and find lower-dimensional representations of the data to derive new hypotheses. Hence, we set PCA to be our main f and σ, μ, *skew*, and, *kurt* to be s transformations.

The analysis starts with the normalization step (T_D^1), where the data is scaled, for example, to the unit interval and followed by the computation of σ, μ, *kurt* and *skew* values for all the dimensions using all the items. Additionally, we perform PCA on the data using all the dimensions.

In the next part of the analysis, we try to understand the relations between dimensions. The changes in basic descriptive statistics (such as μ and σ) due to brushes in items space are easy to interpret and provide information on the correlations between dimensions. Therefore in this step, we choose μ and σ as the visualization axes in dimensions space. We visualize the items in a scatterplot with axes CR vs. AG (V_I^0) and dimensions in a scatterplot of μ vs. σ (V_D^0).

We select the areas with old houses in V_I^0 in Figure 7-a. In dimensions space (in V_D^0), we observe how σ and μ values deviate after the brushing operation. Here, we see that σ values for LS dropped significantly, this is due to the fact that the selection of high AG values is sampling the lower population (LS) dimension unevenly. We interpret this observation as follows:

High values of AG are related to very low values of LS, while low AG values lead to a much broader range of values for LS. In other words, only a very low proportion of the lower status of the population is living in areas with old houses. When focusing on areas with a lower proportion of old houses, there is no limitation with respect to the proportion of the lower status population. This "change point" in the relation between AG and LS was thus discovered by the big deviation of μ and σ when using all or just the selected data. On the contrary, we see that there is almost no change in the μ and σ values on the dimension MV, indicating about the same behavior of the selected and the original data points.

In order to verify these impressions, we visualize the AG dimension as opposed to both LS and MV (V_I^1, V_I^2). We see in V_I^2 in figure 7-a that in areas with old houses, the proportion of lower society is also very low. In V_I^1, we see that MV values vary over a wide range of values for the selected houses (i.e., in areas with older houses). Therefore, it is not possible to talk about a correlation between MV and AG.

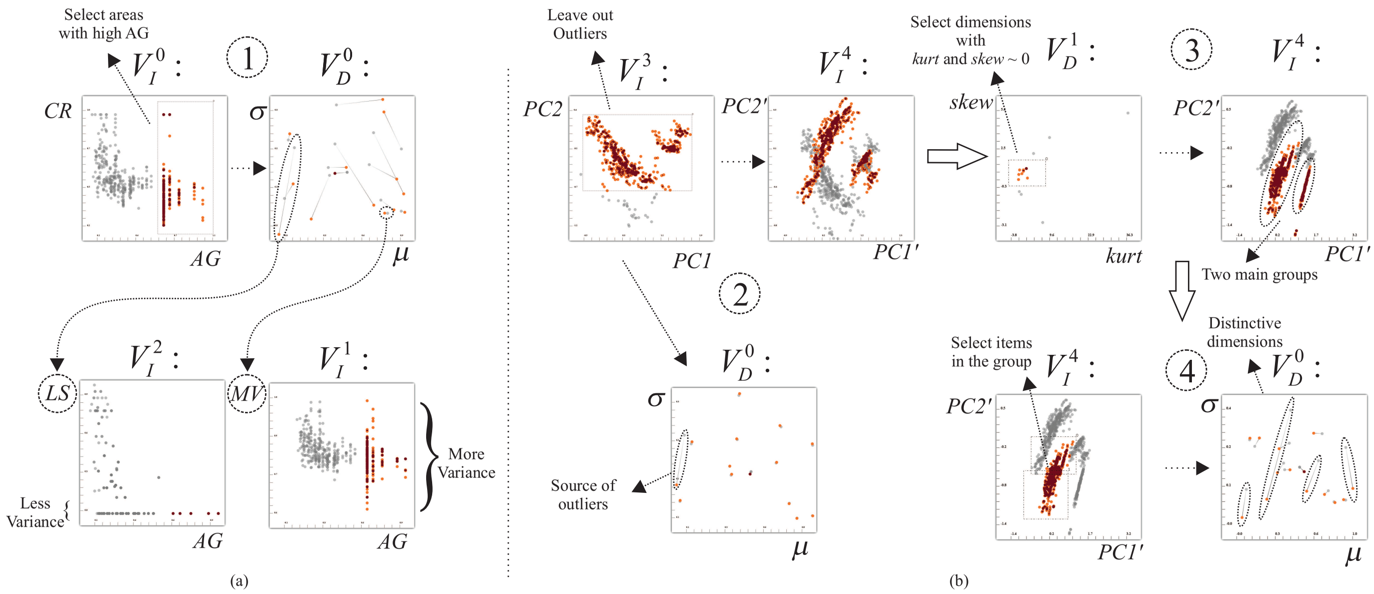


Figure 7: A dual analysis of the housing dataset. a) Houses in areas that have a large proportion of old houses (high AG values) are selected in V_I^0. V_D^0 is updated using new μ and σ values (1). Deviations in V_D^0 indicate a correlation between dimensions w.r.t. the selection. The most deviating (LS) and the least deviating (MV) dimensions are plotted for a deeper analysis. The variance of the selections (in V_I^1 and V_I^2) justifies the deviations in V_D^0. b) Outliers are removed in V_I^3 and PCA is applied with the selected items. V_I^4 is updated with the new results (2). As a result of the selection in V_I^3, one of the dimensions is marked in V_D^0 as the source of the outliers. Before operation (3), the current PCA results are set as the context of V_I^4. Normally distributed dimensions, w.r.t. *kurt* and *skew* values in V_D^1, are selected (3). Updated PCA results now display two groups. One of the groups is selected in V_I^4 and V_D^0 now reveals the dimensions that distinguish the selected group.

The second phase of the analysis involves the elimination of outliers to refine the PCA results. To determine outliers, we use the PCA results (which are already biased by the outliers) that are obtained earlier (V_I^3). V_I^4 in Figure 7-b shows how PCA results change after removing the outliers with the brush in V_I^3. The updated PCA results now display two groups of items, however there is still substantial variation in the groups.

Additionally, the effects of outlier removal are observed through the changes in dimensions space. In Figure 7-b (2), we observe that μ vs. σ values for the Tax-rate (TAX) dimension changed significantly. We mark the TAX dimension as the source of these outliers and remove this dimension (with a $\neg$ brush which is not shown in the image) from the analysis before we move on to the next step. As an intermediate operation, we set the current PCA results (obtained by removing the outliers) as the context of our new visualization (V_I^4).

We would now like to evaluate the dimensions' normality to decide whether to include them in the analysis. Therefore, we continue the analysis in dimensions space. Since *kurt* and *skew* values are indicators of normality, i.e., both the skewness and kurtosis for normal distribution are 0, we select dimensions through the *kurt* vs. *skew* plot (V_D^1). We select dimensions (marked with 3 in the figure) which are more likely to follow a normal distribution by selecting dimensions with values around 0. The updated PCA plot displays two well-separated groups that have less variance throughout the group.

We perform a final brush in V_I^4 to understand which of the dimensions are more distinctive for these groups (Figure 7-b, 4). We select the larger group on the left and observe the changes in μ vs. σ values. Here, we discover four dimensions: "nitric oxides concentration", "number of rooms", "pupil-teacher ratio", "proportion of black by town" to be the distinctive dimensions. These dimensions can now be used for further analysis, e.g., in clustering the houses.

The proposed dual analysis method continues iteratively with interactions between the two spaces. Since the analyst gets an immediate feedback of the interactions, item and dimension selections are refined iteratively until the analyst is satisfied with the results. Note that, the above analysis presents the interpretations of a set of specific statistics and statistical tools. The interpretations of the views and interactions needs to be formulated on the nature of the problem and statistics used.

7 Use Case: Molecular Classification using DNA Microarrays

DNA microarrays and high-density oligonucleotide chips are important monitoring technologies used in cancer research [47]. This monitoring is applied to different tissue samples which are known to be taken from a specific type of tu-

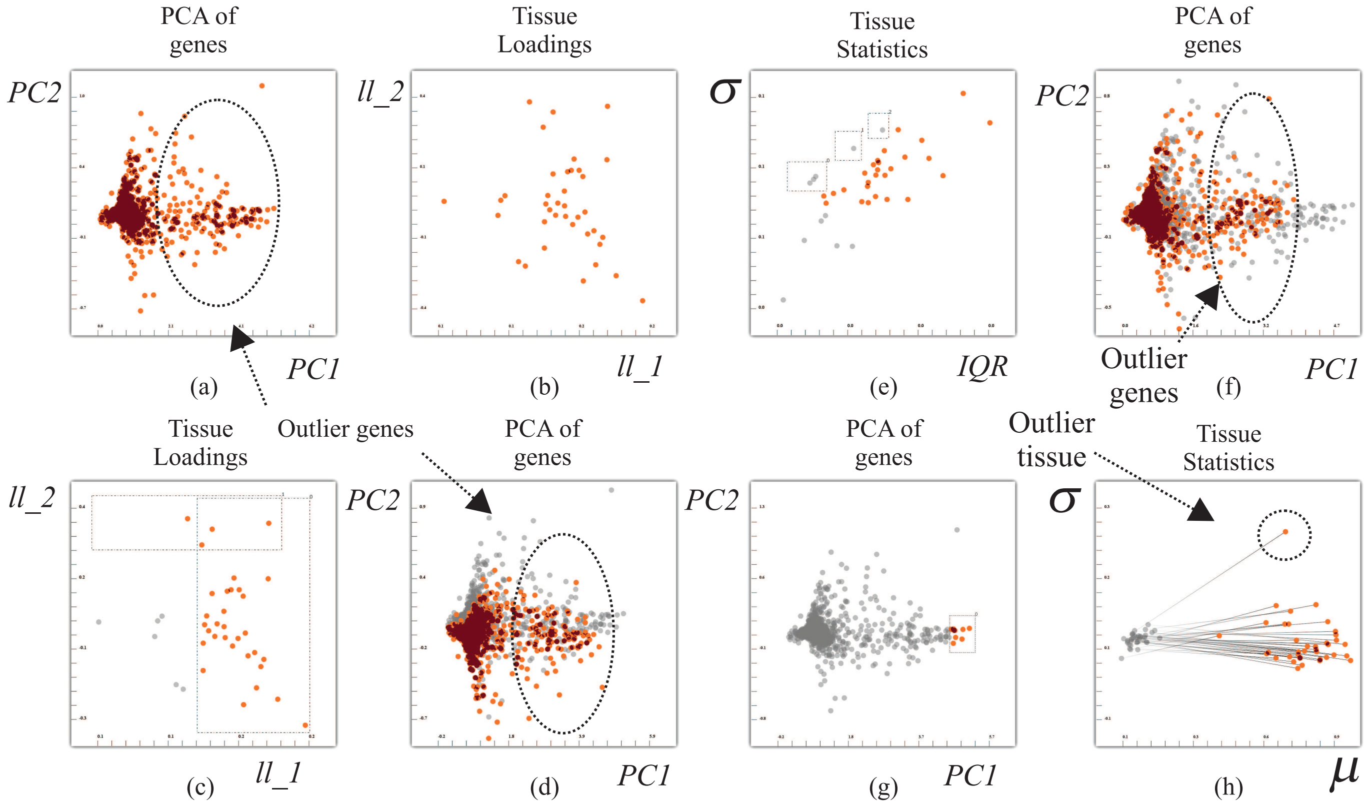


Figure 8: An analysis of microarray data. The task is to select a small number of genes (preferably outliers) for the discrimination of tissues. a) PCA is applied on the genes. There is a large variation and a large number of outliers. b) Tissues are plotted against their PCA loadings *ll*s for PC1 and PC2, where zero loadings indicate redundancy. c) Tissues with large loadings are selected. d) Less number of outlier genes due to the new PCA results. e) Tissues are visualized in a σ vs. IQR plot for the selection of tissues with a smaller number of outliers. f) PCA is computed using the selected tissues. g-h) Analyzing the properties of tissues w.r.t. the genes. For a selected group of genes, an outlier tissue is discovered.

mor. The resulting dataset then contains the expression levels of thousands of genes for these different samples. In molecular level cancer research, these datasets are analyzed to distinguish between cancer classes or even to discover new types of cancers. Two of the main goals in this research which involves statistical approaches are: classifying the samples into classes of tumors and identifying important genes which plays a role in this classification [47]. The statistical analysis of such data has always been a challenge as the dataset contains a very large number of genes (dimensions) compared to the number of tissue samples (items). As the analysts are interested in identifying both the groups of genes and the groups of samples, in the analysis of microarray data, one has to analyze both the original and the transposed version of the dataset.

In this use-case, we work on a gene expression dataset provided by Golub et al. [69]. Here, the samples are known to come from two types of acute leukemia, namely acute lymphoblastic leukemia (ALL) and acute myeloid leukemia (AML). The dataset consists of 7129 genes taken from 38 different tissue samples where 27 are known to be ALL and the rest AML. We treat the dataset in the form that, genes are items (Ω) and samples are dimensions (Δ) as it is the standard way in statistical analysis of microarray data [59].

The task in this use-case is to find a good classifier that distinguish the tissue samples into ALL and AML types. In order perform the classification, we use LDA as an integrated MVA tool. Our aim is to select a number of genes that are more important in the classification of the tissues and thus, improve the performance of the classifier. Without any modification, i.e., using all the samples and all the genes, LDA is able to classify 29 of the 38 samples correctly.

In DNA microarray data analysis, outlier genes are of more importance in the classification of the tissues [59]. Therefore, we focus the analysis on selecting the genes. We, firstly, plot the genes in a scatterplot using PCA and secondly, select outlier genes from the plot to perform the classification with the selected genes. We utilize our model to achieve more reliable PCA results, thus improving the classification performance.

We observe the genes in a visualization of PC1 vs. PC2 in items space. With such a visualization, we aim to separate the more "important" genes and filter out the less interesting ones (Figure 8-a). We visualize the tissues in dimensions space and update PCA results by selecting the tissues (dimensions in this case). To visualize the tissues, we utilize the loadings ll of the PCs as our s function. The loadings are the weights of each single tissue (dimension) in the resulting PCs and they indicate how much a tissue contributes to the principal component. In Figure 8-b, tissues are plotted against ll values (for PC1 and PC2). Here, the ones with higher loadings (in absolute values) are more important variables and the ones with close-to-zero loadings are considered as redundant. We leave out redundant samples (Figure 8-c) and visualize the updated PCA results (Figure 8-d). Here, we see that, we get a smaller number of outlier genes. We select the outlier genes and apply LDA using only these genes. We observe that with this setting, LDA is able classify 30 samples correctly.

We continue the analysis by visualizing the tissues in a interquartile-range (IQR) vs. σ scatterplot. Both σ and IQR are measures of variability, however σ is easily affected by outliers. As a result, if there is a large deviation between IQR and σ values of a dimension, this dimension is likely to contain outliers. In Figure 8-e, we remove such dimensions and re-compute PCA with the selected dimensions. As a result, we observe that we get a more reliable PCA result (Figure 8-f). By selecting the outliers, we observe that LDA classified 34 samples correctly. Additionally, we select a group of outlier genes (Figure 8-g) to explore how the tissues relate to this selected group. In Figure 8-h, we see that while the μ and σ values for most of the tissues change in a similar manner, one tissue is clearly an outlier.

In this use-case, we demonstrate how our model brings new possibilities to the analysis of DNA microarrays. Additionally, we demonstrate how a statistical tool LDA, is used as a validation step. At each iteration, LDA results provides an immediate feedback if the current selection improved the results or not.

8 Conclusion

In this paper, we introduce a visual analysis model that enables the dual analysis of items and dimensions of high-dimensional data. The iterative and joint analysis of the data is performed over two linked spaces: items space and dimensions space. The analysis iterates through the interaction with the items in items space and with the dimensions in dimensions space. In our model, dimensions are the basic visual entities of the visual analysis in dimensions space. Such an approach enables us to extend the knowledge in the interactive visual analysis of data items to the visual analysis of dimensions. To the best of our knowledge, our model is one of the first IVA approaches, where the dimensions are interactively and iteratively analyzed as first-order visual entities together with the actual data items.

We present a formal definition of our model by defining: i) the data transformations that are used to iterate from one space to the other; ii) brushing and F+C visualization to achieve the linking of views. We define how MVA tools and statistics are tightly integrated into the dual analysis concept. Additionally, we present a set of possible analysis procedures that involve the joint interaction of items and dimensions. Finally, we evaluate the model in the context of a DNA microarray data analysis, where the analysis of data items and dimensions is equally important.

MVA tools provide elaborate mechanisms to explore high-dimensional data. They are used for several purposes such as explaining the relations between dimensions, classifying items into groups or predicting the classes of items. One of the problems with these methods is that, they treat all the dimensions of the data equally and consider them in the computations even though they may not be relevant. In certain cases, the relevance of the dimensions can be computationally determined, e.g., by looking at the correlation between dimensions. In some other cases, however, the relevance of a dimension can only be determined by the analyst's preferences or prior knowledge about the data. Moreover, the effects of data item distributions need careful attention while dealing with MVA tools. Such considerations are only possible with the careful inspection of data subsets by an expert. With the presented model, we exploit the tight integration of MVA tools in the visual analysis process and enable the user to reflect her preferences to the analysis. Here, the analyst is given the possibility to steer the MVA tool by means of interactivity and as a result, both the outcome of visual analysis and the performance of MVA methods are improved.

In this paper, we do not focus on specific MVA tools or specific statistics. Therefore, we picked some of the well-known tools and statistics such as PCA, LDA, μ, σ, $skew$, $kurt$, and IQR. The concept of dual analysis can have utilizations with different MVA tools. We plan to work on visualizations and advanced interaction mechanisms that are more specific to certain MVA tools. We will further investigate the utilization of our model in the context of other application domains where the dual analysis concept could prove to be helpful.

As a future work, we will extend our model to include statistics that consider pairs of dimensions, e.g., correlation, regression. Additionally, as another extension, we plan to include visualizations that can provide a formal validation for the interactions, e.g., projection precision [165].

We think that the presented model brings up new opportunities in the analysis of high-dimensional data. By looking at the data from two different perspectives with the help of MVA tools, it is possible to build elaborate and specialized visual analysis frameworks.

Paper B

Representative Factor Generation for the Interactive Visual Analysis of High-Dimensional Data

Cagatay Turkay[1], Arvid Lundervold[2],
Astri Johansen Lundervold[2], and Helwig Hauser[1]

[1]Department of Informatics, University of Bergen, Norway
[2]Department of Biomedicine, University of Bergen, Norway
[3]Department of Biological and Medical Psychology, University of Bergen, Norway

Abstract

Datasets with a large number of dimensions per data item (hundreds or more) are challenging both for computational and visual analysis. Moreover, these dimensions have different characteristics and relations that result in sub-groups and/or hierarchies over the set of dimensions. Such structures lead to heterogeneity within the dimensions. Although the consideration of these structures is crucial for the analysis, most of the available analysis methods discard the heterogeneous relations among the dimensions. In this paper, we introduce the construction and utilization of representative factors for the interactive visual analysis of structures in high-dimensional datasets. First, we present a selection of methods to investigate the sub-groups in the dimension set and associate representative factors with those groups of dimensions. Second, we introduce how these factors are included in the interactive visual analysis cycle together with the original dimensions. We then provide the steps of an analytical procedure that iteratively analyzes the datasets through the use of representative factors. We discuss how our methods improve the reliability and interpretability of the analysis process by enabling more informed selections of computational tools. Finally, we demonstrate our techniques on the analysis of brain imaging study results that are performed over a large group of subjects.

This article was published in *IEEE Trans. Vis. and Comp. Grap.*, 18(12):2621–2630, 2012.

1 Introduction

High-dimensional datasets are becoming increasingly common in many application fields. Spectral imaging studies in biology and astronomy, omics data analysis in bioinformatics, or cohort studies of large groups of patients are some examples where analysts have to deal with datasets with a large number of dimensions. It is not even uncommon that such datasets have more dimensions than data items, which generally makes the application of standard methods from statistics substantially difficult (i.e., the "$p >> n$ problem"). Most of the available analysis approaches are tailored for multidimensional datasets that consist of multiple, but not really a large number of dimensions and they easily fail to provide reliable and interpretable results when the dimension count is in the thousands or even hundreds [2].

In addition to the challenge that is posed by a truly large number of dimensions, it is often the case that dimensions have properties and relations that lead to structures between the dimensions. These structures make the space of dimensions heterogeneous and can have different causes. Dimensions can have difficult-to-relate scales of measure, such as categorical, discrete and continuous. Some can be replicates of other dimensions or encode exactly the same information acquired using a different method. There can be explicit relations in-between the dimensions that are known a priori by the expert. Some of these relations are likely to be represented as meta-data already. Very importantly also, there are usually inherent structures between the dimensions that could be discovered with the help of computational and visual analysis, e.g., correlation relations or common distributions types. Standard methods from data mining or statistics do not consider any known heterogeneity within the space of dimensions – while this might be appropriate for certain cases, where the data dimensions actually are homogeneous, it is obvious that not considering an actually present heterogeneity must lead to analysis results of limited quality.

A natural approach to understanding high-dimensional datasets is to use multivariate statistical analysis methods. These tools provide the analyst with the most essential measures that help with the extraction of information from such datasets. However, a major challenge with these tools is that their results are likely to become inefficient and unreliable when the dimension count gets substantially large [162]. Take, for instance, principal component analysis (PCA), i.e., a method that is a widely used for dimension reduction [100]. If we apply PCA to a dataset with, for example, 300 dimensions, understanding the resulting principal components is a big challenge, even for the most experienced analysts.

Exactly at this point, the exploitation of any known structure between the dimensions can help the analyst to make a more reliable and interpretable analysis. With an interactive visual exploration and analysis of these structures, the analyst can make informed selections of subgroups of dimensions. These groups provide sub-domains where the computational analysis can be done locally. The outcomes of such local analyses can then be merged and provide a better overall understanding of the high-dimensional dataset. Such an approach is very much in line with the goal of visual analytics [111], where the analyst makes decisions with the support of interactive visual analysis methods.

In this paper, we present an approach that enables a *structure-aware* analysis of high-dimensional datasets. We introduce the interactive visual identification of *representative factors* as a method to consider these structures for the interactive visual analysis of high-dimensional datasets. Our method is based on generating a manageable number of representative factors, or just factors, where each represents a sub-group of dimensions. These factors are then analyzed iteratively and together with the original dimensions. At each iteration, factors are refined or generated to provide a better representation of the relations between the dimensions.

To establish a solid basis for our method, we borrow ideas from factor analysis in statistics and feature selection in machine learning. Factor analysis aims at determining *factors*, representing groups of dimensions that are highly interrelated (correlated) [78]. These factors are assumed to be high-level structures of dimensions, which are not directly measurable. Similar to our motivation of an analysis of the structures in the dimensions space, factor analysis also assumes that there are inherent relations between the dimensions. However, factor analysis operates solely on the correlation relation between the dimensions and does not allow the analyst to incorporate a priori information on the structures. Moreover, similar to the other multivariate analysis tools, the resulting factors become harder to interpret as the variable count gets large [78]. A second inspiration for our approach are the feature subset selection techniques, where variables (dimensions) are ordered and grouped according to their relevance and usefulness to the analysis [77]. Similarly, we interactively explore the set of dimensions to extract sub-groups that are relevant for the generation of factors in our method.

In order to visually analyze dimensions through the generation of factors, we make use of visualizations where the dimensions are the main visual entities. We analyze the generated factors together with the original dimensions and make them a seamless part of the analysis. Due to the iterative nature of our analysis pipeline, a number of factors can be generated and refined as results of individual iterations. We present techniques to compare and evaluate these factors in the course of the analysis. Our factor generation mechanism can be both considered as a method to represent the aggregated information from groups of dimensions and a method to apply computational analysis more locally, i.e., to groups of dimensions. Altogether, we present the following contributions in this paper:

- Methods to create representative factors for different types of dimension groups
- A visual analysis methodology that jointly considers the representative factors and the original dimensions

- Methods to assess and compare factors

2 Related Work

In many recent papers, it has been reported repeatedly that the integration of computational tools with interactive visual analysis techniques is of key importance in extracting information from the nowadays highly challenging datasets. In that respect, Keim [111] describes the details of a visual analysis process, where the data, the visualization, hypotheses, and interactive methods are integrated to extract relevant information. Perer and Shneiderman [144] also discuss the importance of combining computational analysis methods, such as statistics, with visualization to improve exploratory data analysis.

There are interesting examples of works where such an integration has been done. In MDSteer [210], an embedding is guided with user interaction leading to an adapted multidimensional scaling of multivariate datasets. A two-dimensional projection method, called the attribute cloud, is employed in the interactive exploration of multivariate datasets by Jänicke et al. [94]. Endert et al. [51] introduce observation level interactions to assist computational analysis tools to deliver more reliable results. Johansson and Johansson [99] enable the user to interactively reduce the dimensionality of a dataset with the help of quality metrics. In these works, interactive methods are usually used to refine certain parameters for the use of computational tools. Our method, differently, enables the integration of the computational tools by interactively determining local domains where these tools are then applied on. Fuchs et al. [66] integrate methods from machine learning with interactive visual analysis to assist the user in knowledge discovery. Oeltze et al. [141] demonstrate how statistical methods, such as correlation analysis and principal component analysis, are used interactively to assist the derivation of new features in the analysis of multivariate data. With our work, we contribute to this part of the literature by having the computational tools as inherent parts and integrating their results seamlessly to the interactive visual analysis cycle. Moreover, we bring together the local structures and the related analysis results to construct a complete image of the relations in high-dimensional datasets.

Multi-dimensional datasets, where the dimension count is a few to several dozens approximately, have been studied widely in the visual analysis literature. Frameworks with multiple coordinated views, such as XmdvTool [202] or Polaris [178], are used quite commonly by now in visual multivariate analysis. Weaver [203] presents a method to explore multidimensional datasets, where the analysis is carried out by cross-filtering data from different views. Surveys by Wong and Bergeron [211] and more recently Fuchs and Hauser [65] provide an overview of multivariate analysis methods in visualization. Compared to all these important related works there are however only few studies published where really high-dimensional data are analyzed. One example is the VAR display by Yang et al. [212], where the dimensions are represented by glyphs on a 2D projection of the dimensions. In order to lay out these glyphs in the visualization, multidimensional scaling is used based on the distances between the dimensions. Fernstad et al. [57] demonstrate their quality metric based reduction in the analysis of high-dimensional datasets involving microbial populations.

Our now proposed method is realized through a visualization approach, where dimensions are the main visual entities and the analysis is carried out together with the data items as recently presented by Turkay et al. [189]. In this (dual analysis) approach, dimensions are analyzed along with the data items in two dedicated linked spaces. This concept enables us to include the representative factors, that we identify, tightly into the analysis. There are few other works where similar dual analysis methods already proved to be useful, such as in parameter space exploration [17], temporal data analysis [9], and multi-run simulation data analysis [109]. Kehrer et al. [106] integrate statistical moments and aggregates to interactively analyze collections of multivariate datasets. Wilkinson et al. introduced graph-theoretic scagnostics [208] to characterize the pairwise relations on multidimensional datasets. In a later work [209], the same authors used these features to analyze the relations between the dimensions. Similar to our work where we analyze the feature space describing dimensions, Wilkinson et al. perform the analysis on the feature space that describes the pairwise relations.

The structure of high-dimensional datasets and the relations between the dimensions have been investigated in a few studies, also. Seo and Shneiderman devise a selection of statistics to explore the relations between the dimensions in their Rank-by-Feature framework [168]. They rank 1D or 2D visualizations according to statistical features to discover relations in the data. However, in their method the main focus is on the data items, not so much the dimensions. One very relevant related work for us is the visual hierarchical dimension reduction method by Yang et al. [213]. They analyze the relations between the dimensions to create a hierarchy that they later use to create lower-dimensional spaces. In our method, we build upon this idea of constructing representative dimensions. However, their method mainly involved an automatic derivation of the dimension hierarchy and the representative dimensions were used as the new visualization domain. In our approach, we treat the representative factors as objects of a dedicated analysis by embedding them into the visualization together with the original dimensions. Moreover, we provide different methods to generate, compare and evaluate the representative factors. In a similar work, Huang et al. [90] utilized the derived dimensions together with the original dimensions. The authors used several dimension reduction methods to derive new dimensions and observed how these dimensions correlate with certain characteristics of the original dimensions. In an interesting paper from the analytical chemistry field by Ivosev et al. [93], the authors present the idea to group variables according to their inter-correlations and utilize them in dimension reduction and visualization. Although their method is applied only to principal component analysis, it clearly demonstrates that grouping of variables indeed improves the analysis of high-dimensional datasets.

Our work now contributes to the literature with a structure-aware interactive visual analysis scheme for high-dimensional datasets. Moreover, we demonstrate that the visually-guided use of computational analysis tools can provide more reliable and interpretable results.

3 Representative Factors

With our method, we explore and consider the structures in the dimensions space during the high-dimensional data analysis. In order to achieve a *structure-aware* analysis of the data, we represent the underlying structures with *representative factors*, or factors, for short. We then analyze and evaluate these factors together with the original data to achieve a more informed use of the computational analysis tools.

A conceptual illustration of our approach is presented in Figure 1. Here, we start by computing statistics s_1 and s_2, e.g., mean and standard deviation, for each of the dimensions in the dataset. We analyze the dimensions by visualizing them in a s_1 vs. s_2 scatterplot, where each visual entity (i.e., point) is a dimension (1). We notice some structure (a cluster in the lower right), which we then represent with a factor (2). With the help of a computational method, e.g., PCA, we generate the representative factor for the selected group of dimensions and replace these dimensions with the generated factor (3). We continue the analysis by exploring the relations between the factor and the represented dimensions, as well as the other dimensions (4). The analysis continues iteratively with the generation of new factors and/or the refinement of the existing ones.

Our method operates (in addition to the original dataset) on a data table dedicated specifically to the dimensions. We construct this dimensions-related data table by combining a set of derived statistics with available meta-data on the dimensions. In order to achieve this, we assign a feature vector to each dimension, where each value is a computed statistic/property or some meta-data about this dimension. If we consider the original dataset to consist of n items (rows) and p dimensions (columns), the derived data table has a size of $p \times k$, i.e., each dimension has k values associated to it. The set of dimensions is denoted as D and the new dimensions properties table as S.

Through a visual analysis of S, we determine structures within the dimensions that then result in a number of sub-groups. We represent these sub-groups of dimensions with representative factors and assign feature vectors to these factors by computing certain features, e.g., statistics. Since factors share the same features as the original dimensions, this enables the inclusion of the factors into the visual analysis process. Moreover, these factors are also used to visually represent

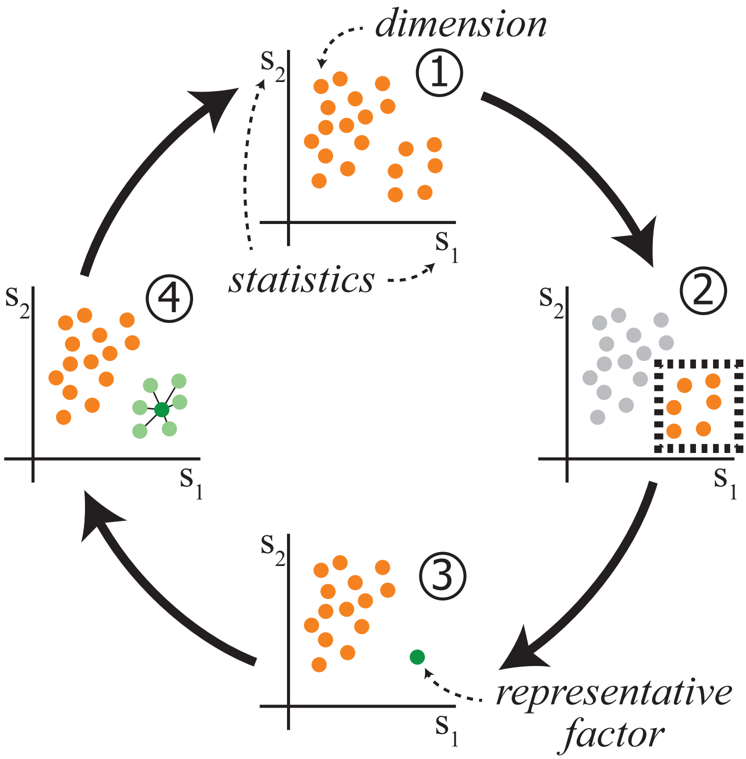


Figure 1: An illustration of our representative factor generation method. Two statistics s_1 and s_2 are computed for all the dimensions and dimensions are plotted against these two values (1). This view reveals a group that shares similar values of s_1 and s_2 (2) and this group is selected to be represented by a factor. We generate a representative factor for this group and compute the s_1 and s_2 values for the factor (3). We observe the relation of the factor to the represented dimensions and the other dimensions (4). The analysis continues iteratively to refine and compare other structures in the data.

the associated sub-group of dimensions. Factors serve both as data aggregation and as a method to apply computational tools locally and represent their results in a common frame together with the original dimensions.

As an illustrative example, we analyze an electrocardiography (ECG) dataset from the UCI machine learning repository [12] in the following sections. The dataset contains records for 452 participants, some of whom are healthy and others with different types of cardiac arrhythmia. There are 16 known types of arrhythmia and a cardiologist has indicated the type of arrhythmia for all the records in the dataset. This dataset is analyzed to determine the features that are helpful in discriminating patients with different arrhythmia types.

The raw ECG measurements are acquired through 12 different channels, and for each single channel 22 different features (a mixture of numerical and nominal attributes) are calculated (leading to $12 \times 22 = 264$ values per individual). Already this description reveals an important inherent structure within all dimensions, i.e., that they form kind of a 2D array of dimensions (channels vs. features). In addition to the above ECG measurements, 11 additional ECG-based features are derived and 4 participant specific pieces of information are included. The result is a 452×279 table ($n = 452$ and $p = 279$).

3.1 Computational and Statistical Toolbox

In order to generate and integrate representative factors into the visual analysis process, we need methods to visually determine the factors and to analyze them together with the other dimensions in D. The dual analysis framework as presented by Turkay et al. [189] provides us with the necessary basis to visually analyze the dimensions together with the data items. We make use of visualizations, where the dimensions are the main visual entities, as well as (more traditional) visualizations of the data items. In order to make the distinction easier, the visualizations with a blue background are visualizations of data items and those with a yellow background are visualizations of the dimensions. For the construction of the factors, we determine a selection of computational tools and statistics that can help us to analyze the structure of the dimensions space.

As one building block, we use a selection of statistics to populate several columns of the S table. In order to summarize the distributions of the dimensions, we estimate several basic descriptive statistics. For each dimension d, we estimate the mean (μ), standard deviation (σ), skewness ($skew$) as a measure of symmetry, kurtosis ($kurt$) to represent peakedness, and the quartiles (Q_{1-4}) that divide the ordered values into four equally sized buckets. We also include the robust estimates of the center and the spread of the data, namely the median (med) and the inter-quartile range (IQR). Additionally, we compute the count of unique values ($uniq$) and the percentage of univariate outliers ($\%out$) in a dimension. $uniq$ values are usually higher for continuous dimensions and lower for categorical dimensions. We use a method based on robust statistics [106] to determine $\%out$ values. In order to investigate if the dimensions follow a normal distribution, we also apply the Shapiro-Wilk normality test [158] to the dimensions and store the resulting p-values ($pVal_{shp}$) in S. Higher $pVal_{shp}$ indicate a better fit to a normal distribution. In the context of this paper, we limit our interest to the normal distribution due to its outstanding importance in statistics [100].

One common measure to study the relation between dimensions is the correlation between them. We compute the Pearson correlation between the dimensions to determine how the values of one dimension relate to the values of another dimension. Correlation values are in the range [-1, +1] where -1 indicates a perfect negative and +1 a perfect positive correlation.

Additionally, we use multidimensional scaling (MDS) to help us to investigate the structure of the dimensions space. MDS is a method that projects high-dimensional data items usually to a 2D space by preserving the distances between them as good as possible. Here, we use MDS directly on the dimensions, similar to the VAR display by Yang et al. [212]. We use the correlations between the dimensions to compute a distance matrix, where this distance information is used as an input to MDS. As a result, MDS places the highly inter-correlated groups close to each other. All these computational analysis tools are available through the integration of the statistical computation package R [184]. This mechanism enables us to easily include a variety of tools in the analysis.

3.2 Factor Construction

Constructing factors that are useful for the analysis is crucial for our method. Since factors are representatives for sub-groups of dimensions, they are constructed to preserve different characteristics of the underlying dimensions. The machine learning and data mining literature provides us with valuable methods and concepts under the title of feature (generally called an attribute in data mining) selection and extraction [77]. Feature extraction methods usually map the data to a lower dimensional space. On the other hand, feature subset selection methods try to find dimensions that are more relevant and useful by evaluating them with respect to certain measures [20].

Here, we introduce three different methods to construct representative factors using a combination of feature extraction and selection techniques. Each factor construction method is a mapping from a subset of dimensions D' to a representative factor D_R. The mapping can be denoted as $f : D' \to D_R$, where $D' \in 2^D$. The t dimensions that are represented by D_R are denoted as $d_0^R, \ldots, d_t^R$. Each factor creation is followed by a step where we compute a number of statistics for D_R and add these values to the S table. In other words, we extend the D table with a D_R column and the S table with a row associated with D_R. Notice that each D_R column consists of n values similar to the other columns of the D table.

Projection Factors

The first type of representative factor is the *projection factors*. Such factors are generated using the output of projection-based dimension reduction methods that represent high-dimensional spaces with lower dimensional projections. Projection factors are preferred when we want the resulting factor(s) to represent most of the variance of the underlying dimensions [100]. In order to determine structures that are suitable to be represented via this type of factors, we analyze the correlation

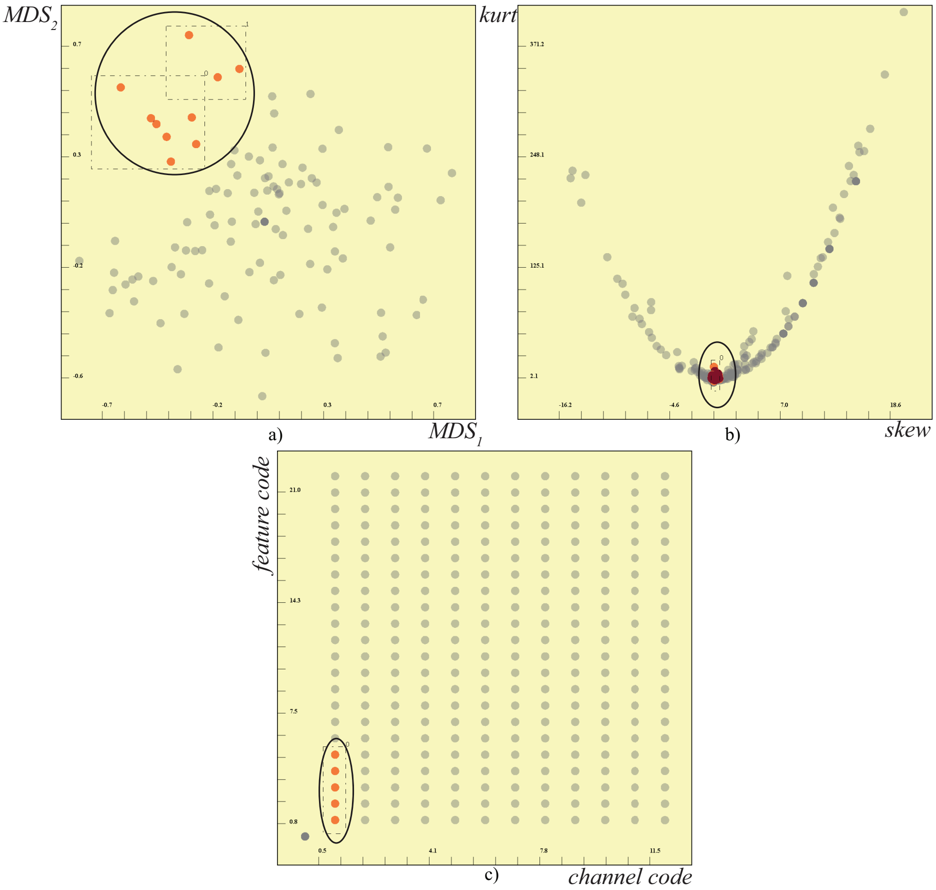


Figure 2: Groups of dimensions that are suitable to be represented by different types of factors. a) MDS is applied to the dimensions using the correlation information. A highly inter-correlated group is selected to be represented by a projection factor. b) A group of dimensions that are likely to come from a normal distribution (*skew* and *kurt* ~ 0) is to be represented by a distribution model factor. c) Meta-data is utilized to select a group of dimensions (same channel, different features) that then can be represented by a medoid factor.

relations between the dimensions. Subsets of dimensions that are highly inter-correlated are good candidates to be represented by a projection factor.

In the context of this paper, we use principal component analysis as the underlying reduction method. However, depending on the nature of the data and the analysis, different reduction methods [100] could be employed here, too.

During each projection-factor generation we create two factors, being the first two principal components here. We choose to include two components in order to be able to visualize also the data items in a scatterplot when needed. For D', where the variance structure cannot be well captured by two components, we suggest two options. The first option is to apply PCA to several subsets of D' and create factors for each of these subsets. These subsets can be determined by observing the inter-correlations between the dimensions in D' and separating the sub-groups with stronger inter-correlations. The second option is to use more components (factors) than two where a more accurate number can be determined by certain methods suggested in the literature, such as observing a scree-plot [100]. In our analysis, we prefer the first method instead of creating a larger number of factors per D', since it creates easier to interpret factors.

In order to determine sub-groups of dimensions that are suitable to be represented with projection factors, we can make use of MDS. If we apply MDS on the dimensions using the correlation matrix as the distance function and visualize the results, the clusters in such a view corresponds to highly inter-correlated sub-groups, i.e., suitable for a projection factor. In Figure 2-a, we see such a sub-group of dimensions (consisting of 10 dimensions) that is suitable to be represented with a projection factor. We then apply PCA to these 10 selected dimensions and store the first two principal components as the representative factors for these 10 dimensions.

Projection factors are the most suitable factors when the goal of the analysis is dimension reduction. Since different dimension reduction methods have different assumptions regarding the underlying data, evaluating these assumptions leads to more reliable results. In that respect, dimensions can be analyzed in terms of their descriptive statistics, normality test scores and *uniq* values to determine their suitability.

Distribution Model Factors

The second type of representative factor is the *distribution model factors*. These factors represent the underlying dimensions with a known distribution where the distribution parameters are derived from the underlying dimensions. Distribution model factors are suitable to represent groups of dimensions that share similar underlying distributions. In the context of this paper, we limit our investigation of the underlying distributions to the normal distribution. If a group of dimensions are known to come from a normal distribution, these dimensions can be represented by a normal distribution where the modeled distribution parameters are derived from the group. The representative normal distribution can be written as:

$$\mathcal{N}(\sum_{i=0}^{t-1} \frac{med_i}{t}, \sum_{i=0}^{t-1} \frac{IQR_i}{t})$$

Here, med_i is the median and IQR_i is the inter-quartile range of the dimension d_i where $d_0, \ldots, d_i \in D'$. We prefer the robust estimates of the center and the spread of the distributions to make our distribution generation step more resistant to outliers. As a final step, we draw n values from $\mathcal{N}$ to generate the representative factor D_R. Notice that, here, the $\mathcal{N}$ distribution is one dimensional, thus we create a single factor for the underlying t dimensions. In other words, D_R is a new artificial dimension, where the data items are known to come from the modeled distribution $\mathcal{N}$.

In Figure 2-b, we visualize the dimensions by a *skew* vs. *kurt* scatterplot. Normal distributions tend to have *skew* and *kurt* values very close to 0. This view enables us to select a group that is likely to follow a normal distribution, and thus, suitable to be represented via a distribution model factor.

Distribution model factors are suitable for distribution fitting tasks. To extend the applicability of this type of factors, different types of known distributions could be considered as well, such as Student's t-distribution or the chi-square distribution. Depending on the distribution type to be tested, dimensions can be visualized either over descriptive statistics or fitness scores to known distributions.

Medoid Factors

The third type of representative factor is the *medoid factors*, that are generated by selecting one of the members of D' as the representative of D'. Such factors are preferred when the dimensions in D' are known to share similar contextual properties or some of the dimensions could be filtered as redundant. The user may prefer to select one of the dimensions and discard the rest due to redundancy. Meta-data on the dimensions provide a good basis to determine and select the suitable dimensions to be represented by medoid factors.

In order to automatically determine one of the dimensions as the representative, we employ an idea from partitioning around medoids (PAM) clustering algorithm [103]. In this algorithm, cluster centers are selected as the most central element of the cluster. Similarly, to find the most central element, we choose the dimension $d \in D'$ that has the minimum total distance to the other dimensions, computed as:

$$\arg\min_{d}(\sum_{j=0}^{t-1} dist(d, d_j)), d \neq d_j, (d, d_j \in D')$$

where $dist$ is chosen as the Euclidean distance and t is the total number of dimensions in D'. This dimension d is then selected as the representative. In Figure 2-c, we make use of the meta-data information to determine a group that is suitable to be represented via a medoid factor. Here, we plot the channel codes and the feature codes on a scatterplot. The first five features associated

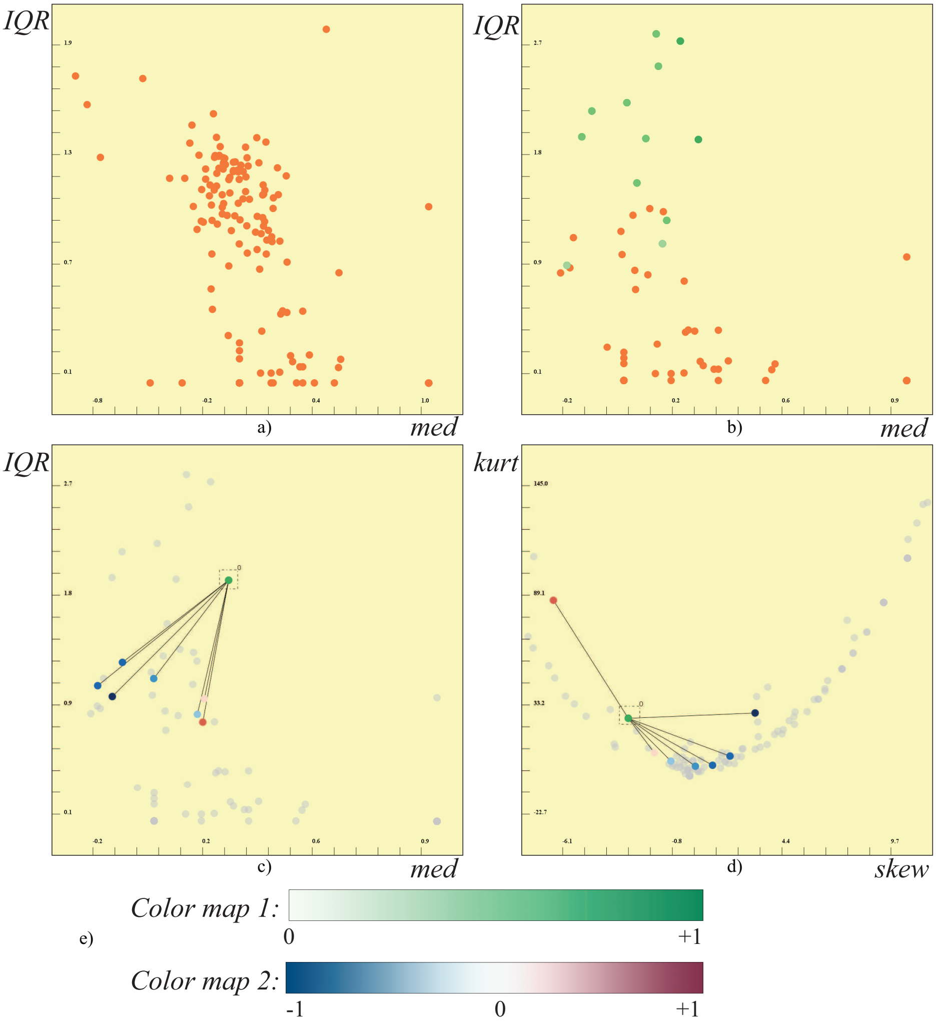


Figure 3: Integrating factors in the visual analysis. a) The normalized dimensions of the ECG data are visualized in a *med* vs. IQR scatterplot. b) Each channel in ECG is represented by a factor. The coloring is done based on the aggregated correlation. c) The factor for channel DI is expanded (D_R^{DI}) and visually connected to the dimensions it represents (d^R). The coloring is done on the mutual correlations between D_R^{DI} and d^R. d) The relation between D_R^{DI} and d^R are different for *skew* and *kurt* values. e) Two color maps are used to map correlation information, the first is used to color representative factors using the aggregated correlation and the second for the represented dimensions.

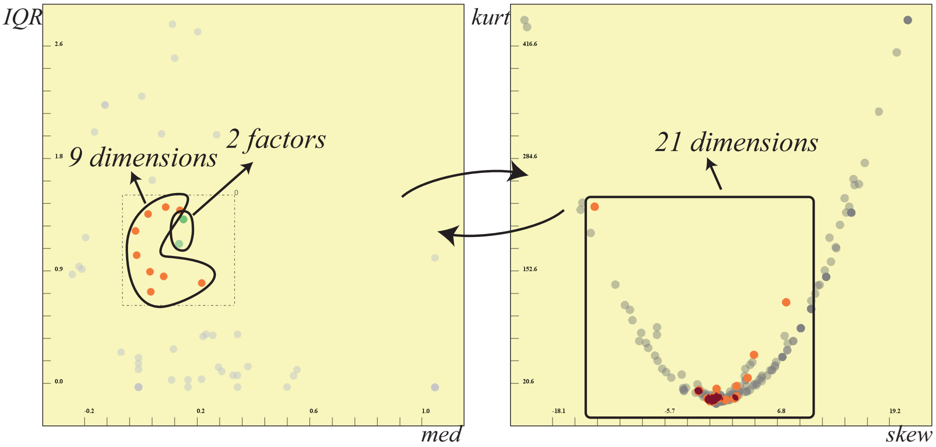


Figure 4: Representative factors can be brushed together with the original dimensions. When a factor is selected, all the dimensions that are represented by the factor are highlighted in the other views. And similarly, when one of the represented dimensions is selected in another view, the associated factor is highlighted. Here, 9 raw dimensions and 2 factors (each representing 6 dimensions) are brushed. A total of $9 + 2 \times 6 = 21$ dimensions are highlighted in the other views.

with a channel are known to be associated with the width of sub-structures in the channel, thus they can be represented by a medoid factor.

3.3 Integrating Factors in the Visual Analysis

In order to include the factors into the dimensions visualizations, we compute all the statistics that we already computed for the original dimensions also for the representative factors. We add these values on D_R as a row to the table S. This enables us to plot the factors together with the original dimensions.

Figure 3-a shows the dimensions in a plot of *med* vs. IQR. We then select all the continuous dimensions that are related to the first channel DI and apply a local PCA to the selected dimensions. We leave out the categorical data dimensions since they are not suitable to be included in PCA calculations. We perform the same operation also for the other 11 channels. This leaves us with a total of 12 representatives, each of which represents 16 dimensions. We compute the *med* and IQR values also for the D_Rs and replace the original dimensions with their representatives in Figure 3-b. The representatives are colored in shades of green to distinguish them from the original data dimensions. Here, we see the relation between different channels through the distribution of the factors over the *med* vs. IQR plot. In order to see how a single factor relates to the represented

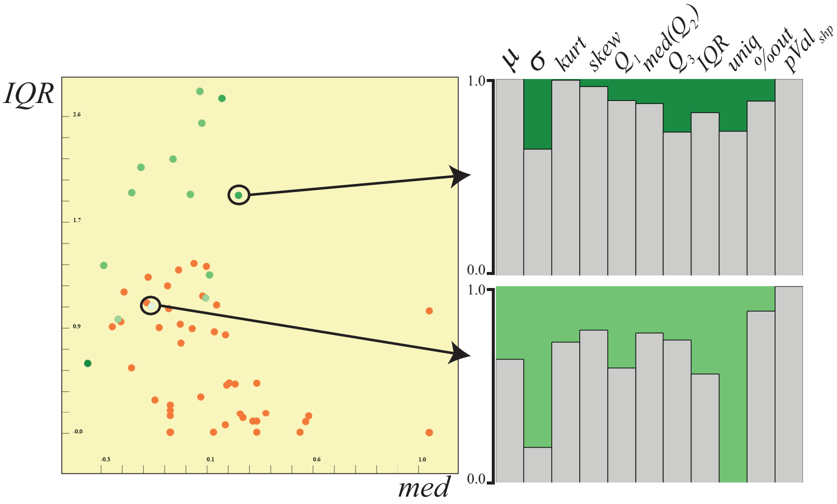


Figure 5: Two profile plots for two different representative factors (visible in the *med* vs. IQR plot) are visualized. Each bin in the profile plots is associated with the listed statistics. The profile plot for the first factor shows that most of the features of the represented dimensions are preserved. However, the second profile indicates that the factor fails to represent the features.

dimensions over the *med* and IQR values, the factor is expanded and connected with lines to the represented dimensions (Figure 3-c). The relations between the factor and the represented dimensions are also observed on a *skew* vs. *kurt* view (Figure 3-d).

Brushing representative factors: Representative factors require a different way of handling in the linking and brushing mechanism. When the user selects a representative factor D_R in a view, all the dimensions d_i^R that are represented by D_R in the other views are highlighted. Similarly, when the user selects one of the d_i^R dimensions, the related D_R is highlighted in the other views. Figure 4 illustrates how the selections of factors are linked to the other views. Here, for each factor selected in the *med* vs. IQR view, 6 associated dimensions are selected in the second *skew* vs. *kurt* view. Therefore there are 21 selected dimensions in total in the right view. This mechanism enables us to interact with information at both the original dimension level and the aggregated level.

3.4 Evaluation of the representatives

The evaluation and a more quantitative comparison of the factors is an essential part of a representative factor based analysis pipeline as presented here. We provide two different mechanisms to evaluate the factors using quantitative measures.

The first method is related to the correlation based coloring of the factors and the represented dimensions. As an inherent part of the factor generation, we compute the Pearson correlation between D_R and the dimensions that it represents d_i^R. The result is a set of t values $corr_R$, where each value is in the range [-1, +1] as described already. We color-code these pieces of correlation information in the views using two different color maps (Figure 3-e). Firstly, we represent the aggregated correlation values as shades of green. For each D_R, we find the average of the absolute values of $corr_R$. More saturated green represent higher levels of correlation (either positive or negative) and paler green represent lower levels. Secondly, we encode the individual values of $corr_R$ when a factor is expanded. Each represented dimension d_i^R is colored according to the correlation with D_R. Here, we use a second color map where negative correlations are depicted with blue and positive correlation with red.

The second mechanism to evaluate the factors is called *profile plots*. When the set of statistics associated with dimensions is considered, factors do not represent all the properties equally. If we consider again how the same factor relates to the represented dimensions over *med* and *IQR* in Figure 3-c and *skew* vs. *kurt*, in Figure 3-d, we see different levels of similarity between D_R and the represented dimensions. Since these relations for all the statistics, i.e., columns of S, are different, we build profile plots to visually represent this difference information. In order to find the similarity between D_R and d_i^i with respect to the statistic s, we compute the following value:

$$sim_s = 1 - \frac{\frac{1}{t}\sum_{i=0}^{t-1}|s(D_R) - s(d_i^R)|}{max(s(d_i^R)) - min(s(d_i^R))}$$

The *sim* values are in the range [0, 1] where higher values indicate that the representative has similar s values as the represented dimensions. We present the sim_s values for all the different statistics in a histogram-like view called profile plots as seen in Figure 5-right. Here, each bin of the plot corresponds to a different s (as listed in the figure) and the sim_s value determines the height of the bin. Additionally, we color-code the average of sim_s values as the background to the profile plots, with the color map (marked 1) in Figure 3. In Figure 5, we see two examples of factors where the profile plot for the first factor preserves most of the features of the underlying dimensions. However, the second profile plot shows that the factor has different values for most of the features of the underlying dimensions.

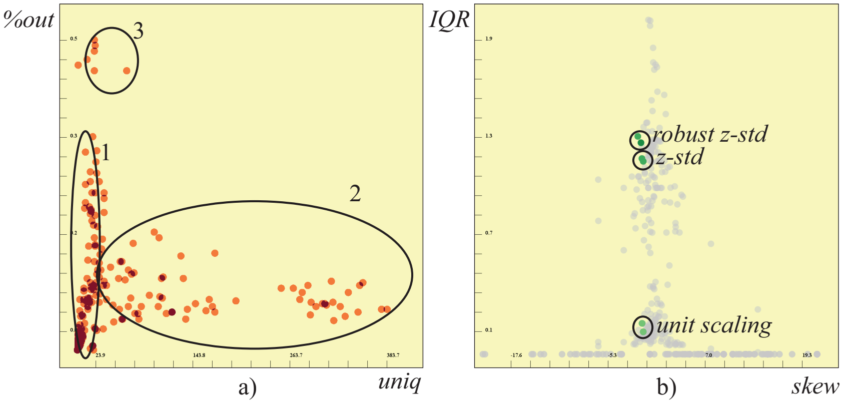


Figure 6: a) Different normalization methods could be suitable for different types of dimensions. We use unit scaling for group 1, z-standardization for group 2 and robust standardization for group 3. b) Three different normalizations are applied on the same group of dimensions and three sets of factors (using PCA) are generated accordingly for the same group. The differences between the results show that transformations can affect the outcomes of computational tools.

4 Analytical Process

The structure-aware analysis of the dimensions space through the use of these factors involves a number of steps. In the following, we go through the steps and exemplify them in the analysis of the ECG data. Still, these steps are general enough to provide a guideline for the analysis of heterogeneous high-dimensional data using the representative factors.

Step 1: Handling missing data – Missing data are often marked prior to the analysis and available as meta-data. It is important to handle missing data properly and there are several methods suggested in the corresponding literature [78]. We employ a simple approach here and replace the missing values with the mean value of continuous dimensions prior to the normalization step. Similarly, in the case of categorical data, we replace the missing values with the mode of the dimension, i.e., the most frequent value in the dimension. Moreover, we store the number of missing values per each dimension in S for further reference.

Step 2: Informed normalization – Normalization is an essential step in data analysis to make the dimensions comparable and suitable for computational analysis. Different data scales require different types of normalization (e.g., for categorical variables scaling to the unit interval can be suitable, but not

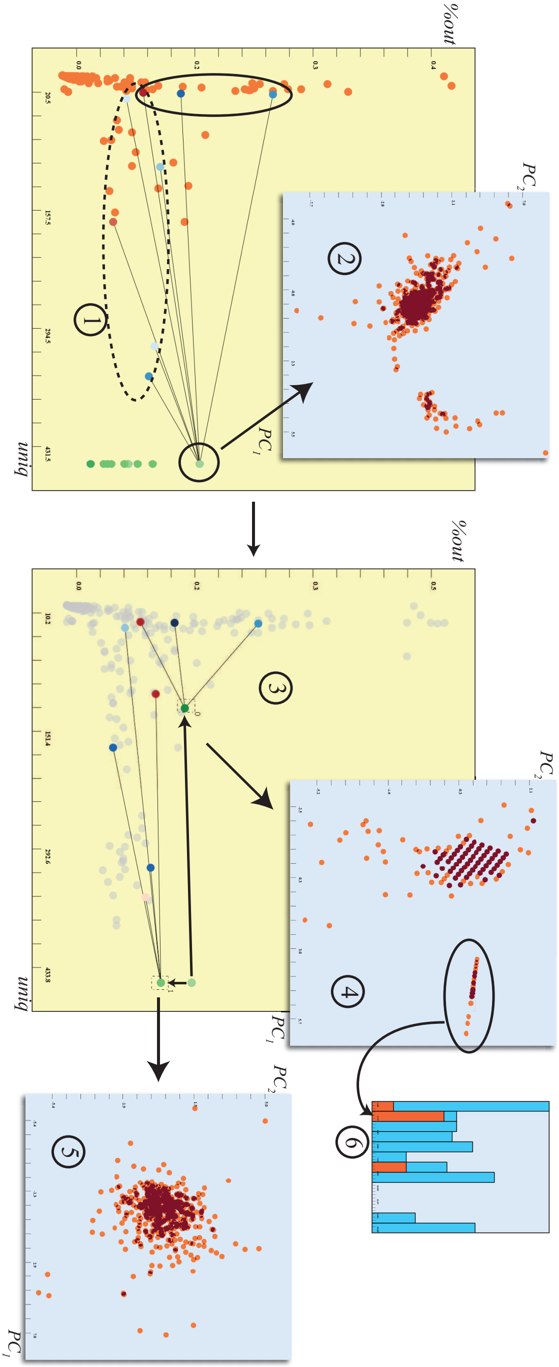


Figure 7: A sample analysis of the ECG dataset. One factor for each of the channels is created and displayed in a *uniq* vs. *%out* plot (1). One channel V2 has a high *%out* value. The expanded dimensions shows that it has strong correlations with some of the dimensions (solid ellipse) and less with the other (dashed ellipse). We use all the underlying dimensions to apply PCA to the subjects and observe two groups (2), however with some noise. We analyze further and create new factors for the two sub-groups (marked with the ellipses) (3). When we apply PCA using these subgroups separately, we see that the grouping is due to the strongly correlated dimensions (4) and there was no distinctive information in the other ones (5). We bring up a histogram where bins are different arrhythmia types. We observe that the left group in plot 4 is mainly the subjects with coronary artery disease. This means that V2 is a good discriminator for such types of arrhythmia.

z-standardization) and different analysis tools require different normalizations, e.g., z-standardization is preferred prior to PCA. We enable three different normalization options, namely, scaling to the unit interval [0,1], z-standardization, and, robust z-standardization. In the robust version, we use med as the robust estimate of the distribution's center and IQR for its spread. In order to determine which normalization is suitable for the dimensions, we compute certain statistics, namely $uniq$, $pVal_{shp}$ and $\%out$, prior to normalization. We visualize $uniq$ vs. $\%out$ (Figure 6-a) to determine the groups of dimensions that are suitable for different types of normalizations. Dimensions with low $uniq$ values (marked with 1 in figure) are usually categorical and scaling to the unit interval is suitable. Dimensions with higher $uniq$ values (marked 2) are more suitable for z-standardization. And, for those dimensions that contain larger percentage of one dimensional outliers (marked 3), a robust normalization is preferable. We normalize the same sub-group of dimensions using all the three methods and apply PCA separately on the three differently normalized groups. Figure 6-b shows the first two principal components factors. We observe that non-robust and robust normalizations resulted in similar outputs, however the unit scaling resulted in PCs that carry lower variance.

Step 3: Factor generation – In this step, we analyze the structures in the dimensions space firstly through the help of meta-data information. We choose to represent each channel only by the first principal component. Each channel in the ECG data has 22 dimensions associated, however, we select a sub-group of these features (the continuous features (dimensions) that have larger $uniq$ values) and then construct projection factors for each channel. The resulting groups are now displayed on a $uniq$ vs. $\%out$ plot (Figure 7).

Step 4: Evaluating and refining factors iteratively – In figure 7-1 we notice that the factor that is representing the $V2$ channel (denoted as D_R^{V2}), has a higher percentage of 1D outliers. This is interpreted as a sign of an irregular distribution of items in this factor and we decide to analyze this factor further. First, we have a look at the items in a scatterplot of the first two components of D_R^{V2} and we clearly see that there are two separate groups (figure 7-2). However, when we expand the selected factor to see its relation with the underlying dimensions, we observe that there are dimensions that the factor has strong correlations (D'_1) and some other that have weak correlations (D'_2). We decide to refine this factor further by creating two smaller groups D'_1 and D'_2 and visualize the new factors in the same view (Figure 7-3). When we observe the items in visualizations of the first two components of the new factors (Figure 7-4,5), we see that the grouping is solely due the dimensions in D'_1. The dimensions in D'_2 carry no significant information.

In order to the analyze the separated group of patients in Figure 7-5, we observe the arrhythmia class label column in a histogram. We find out that the selected group accounts for almost all the patients with coronary artery disease (Figure 7-6). This shows that these three dimensions associated with the $V2$ channel are distinctive features for coronary artery disease.

Here, we present a step-by-step iterative analysis where at each iteration we refine the factors and dig deeper into the data. The above example demonstrates how the representative factors enables a more controlled use of computational tools and a better understanding of the relations in-between the dimensions.

5 Use Case: Analysis of Healthy Brain Aging Study Data

In this use case we analyze the data related to a longitudinal study of cognitive aging [8, 215]. The participants in the study were healthy individuals, recruited through advertisements in local newspapers. Individuals with known neurological diseases were excluded before the study. All participants took part in a neuropsychological examination and a multimodal imaging procedure, with about 7 years between the first and third wave of the study. One purpose of the study was to investigate the association between specific, image-derived features and cognitive functions in healthy aging [215]. In the study, 3D anatomical magnetic resonance imaging (MRI) of the brain has been complemented with diffusion tensor imaging (DTI) and resting state functional MRI [89, 214]. Here we are interested in the analysis of the anatomical MRI recordings. These recordings are segmented automatically [62], and statistical measures, such as surface area, thickness and volume (among several others) are computed for each of the segmented cortical and subcortical brain regions. The neuropsychological examination covered tests of motor function, attention/executive function, visual cognition, memory- and verbal function. The participants' results on these tests are evaluated by a group of neuropsychologists.

The dataset covers 83 healthy individuals with the measurements from the first wave of the study in 2005. For each subject, a T1-weighted image was segmented into 45 anatomical regions, and 7 different measures were extracted for each region. For a complete list of brain regions, refer to the work by Fischl et al. [60]. These computations are done automatically using the software called Freesurfer [62]. The 7 features associated with each brain region are *number of voxels*, *volume* and *mean, standard deviation, minimum, maximum and range of the intensity values in the region*. This information on the brain regions and the features is represented in the meta-data file, which is then used in the analysis. The above operation creates $45 \times 7 = 315$ dimensions per subject. In addition, details about each individual, such as age and gender, and the results of the neuropsychological examination are added to this dataset. With this addition,

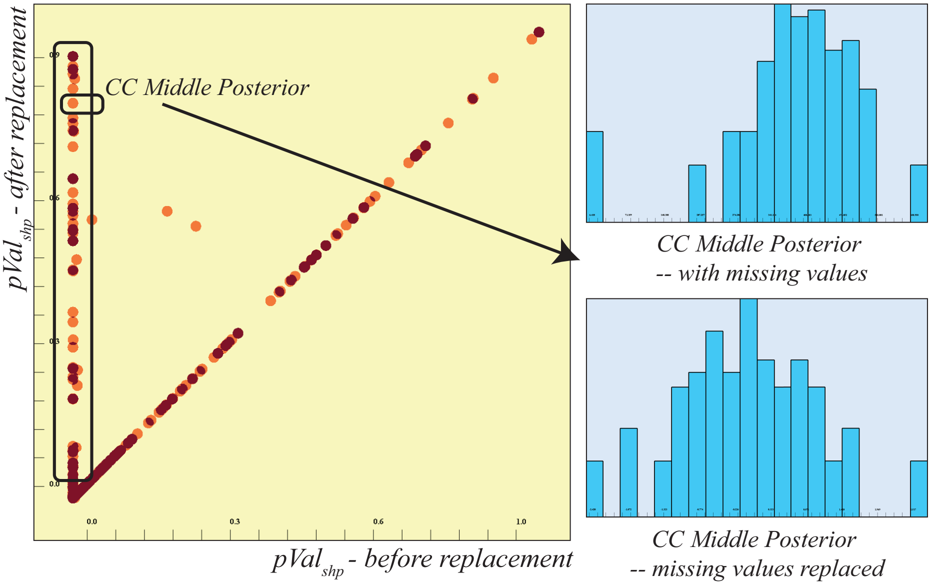


Figure 8: Missing values are handled automatically in our system, and the effects of this transformation is observed here. Normality test scores before and after the transformation are to the left. For a large number of dimensions, the normality test scores improved. On the right, the dimension *Cerebellum Cortex middle Posterior* is inspected before and after missing values are replaced.

the resulting dataset has 357 dimensions. In other words, the resulting table's size is 83 × 357 – a great challenge for visual as well as computational analysis. Such a high dimensionality usually requires analysts to delimit the analysis to a selected subset of segments, based on an a priori specified hypothesis. Our aim here is to discover different subsets of individuals and brain regions that are relevant for building new hypotheses.

We start our analysis with the missing value handling and the normalization step. Missing values in the dataset are identified with different strings in different columns of our dataset. And these identifiers (specific for each dimension) are recorded in the meta-data file. We replace the missing values with the mean (or mode) of each column. In Figure 8, we see the normality test values before and after the replacement. It is seen that some of the dimensions (marked with the big rectangle) have a large number of missing values which affect their fitness to normality. One example is the selected CC-middle posterior dimension (histograms in Figure 8), which shows a skewed histogram first (the binning of the histogram is distorted by missing values), and then, nicely fits to a normal distribution after the replacement. We continue with the normalization where we prefer different normalizations for different types. Here, dimensions related to participant specific information and the memory test are scaled to the unit interval and the rest of the dimensions are z-standardized.

After these initial steps, we start by investigating the 7 different features associated with the brain regions and generate 7 projection factors for these 7 sub-groups. We select these groups through the use of the available meta-data (not shown in the images here). Each factor here represents 45 dimensions, being the different brain regions, e.g., one sub-group contains all the *number of voxels* columns for the 45 brain regions. We visualize these factors over a *med* vs. IQR plot (Figure 9-a) and bring up a matrix of profile plots, Figure 9-b, for these factors. The first observation we make through the profile plots is that the *number of voxels* (marked 1) and *volume* (2) features carry identical information. We decide that one of these features needs to be left out. In this specific example it is, of course, clear that number of voxels is equal to the volume. However, such relations may not be always easily derived from the names of the features and require visual feedback to be discovered. Moreover, the profile plot reveals that the *range of intensity* feature (7) preserves most of the statistics in the underlying dimensions. We also mark the *standard deviation of intensities* as interesting, since the underlying dimensions have different correlation relations with the representative factor. This indicates that this feature is likely to show differences between the brain regions.

We continue by delimiting the feature set for the brain regions to those two selected features. This means that we delimit the operations to 45×2 dimensions and apply MDS on these 90 dimensions using the correlation matrix as the distance values. We identify a group of dimensions that are highly correlated in the MDS plot (Figure 9-c). We find out that this group is associated with the substructures in the Cerebellum Cortex (CerCtx) and CerCtx is represented with 5 sub-regions in the dataset. We decide to represent all the dimensions related to the CerCtx via a medoid factor.

As the next step, we create factors to represent each brain-region (not CerCtx, since it is already represented by a medoid factor). We compute a PCA locally for each brain region and create representative factors. In Figure 9-d, we see the factors (using only the first component) over a normality score vs. $\%out$ plot. Here, each factor represents a single brain region. We select the brain regions, where the representative shows a normal distribution. Such a normally distributed subset provides a reliable basis to apply methods such as PCA on the participants. From this analysis, the regions of interest are *right and left lateral ventricle*, *brain stem*, *left and right choroid plexus* and *right inferior lateral ventricle*. Using only the selected regions, we apply PCA on the subjects (Figure 9-e). We select a group of outlier participants and visualize them on a scatterplot of *birth year* vs. *gender*. We observe that this group is mainly composed of older participants. This observation leads to the hypothesis that the selected brain structures are affected by aging.

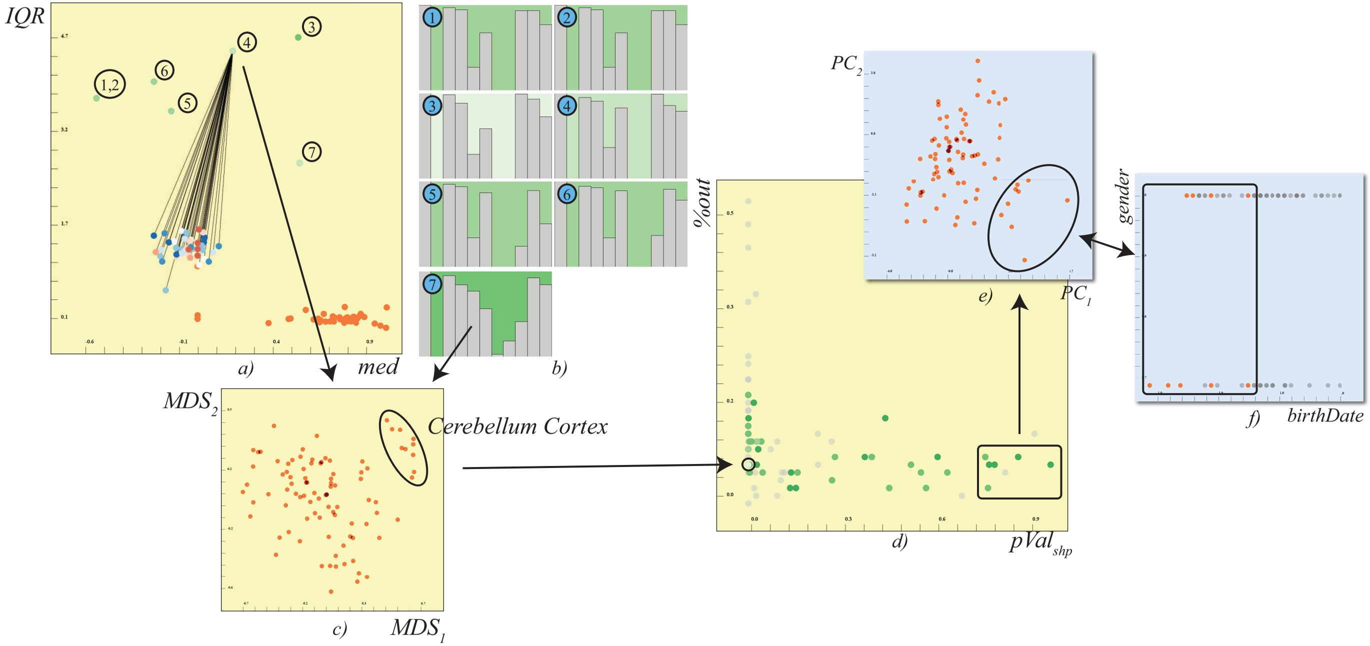


Figure 9: Analysis of the healthy brain aging dataset. We generate factors for the 7 types of features (a). Each factor represents 45 dimensions (the number of brain regions). We observe the profile plots for these seven factors (b). The profile plots for *number of voxels* and *volume* (1 and 2) reveal that these two features are identical, thus we discard one of them. One of the factors (4) has a varied correlation relation with the underlying dimensions and another factor (7) is a strong representative of the statistics over the brain regions. For each brain region, we limit the features to these two and apply MDS on this subset of dimensions (c). The MDS reveals a tightly inter-related group of dimensions that is found to be associated with the Cerebellum Cortex (CerCtx). CerCtx is represented by a medoid factor and the rest with projection factors. These factors, each representing a brain region, are visualized on a $pVal_{shp}$ vs. *%out* plot (d). 6 of the “most normally” distributed factors are selected. PCA is applied on the participants. We notice a group of individuals with outlying values (e) and find out that this group consists of elderly subjects (f). We conclude that the selected 6 brain regions are likely to be affected by aging (this hypothesis would still have to be tested to make a more definite statement).

Here, we comment on the findings related to the the selected brain regions. *Right and left lateral ventricle* are part of the ventricular system that are filled with cerebrospinal fluid (CSF). These regions are interesting and expected findings, and they are known to increase with age (since the brain tissue parenchyma shrinks and the intracranial volume remains constant). *Brain stem* image information might not be so reliable in the periphery of the core magnetic field homogeneity of the scanner, thus needs to be left out from the hypothesis. *Left and right choroid plexus* are small protuberations in the ventricles' walls/roof that produces CSF. It is unexpected for these structures to influence interesting age-related associations. However, this is an unexpected and important finding that our analysis can provide and can be subject to further investigation.

In order to validate the significance of our findings, we focused on the nine participants that we selected in Figure 9-e. As mentioned above, we analyzed the data from 2005, i.e., when all the participants are known to be healthy. Since the data is from a longitudinal study, there are internal reports on how the cognitive function of the participants evolved over time in the next waves of the study. Through these reports, we observe that one of the nine participants is described as showing an older infarct (through MRI scans) and six of the remaining participants (75%) showed declining cognitive function during the study period. The percentage (of cognitive function decline) in the other participants is 28%. This shows a clinical importance of the selected participants. Moreover, this result supports the above hypothesis that the selected brain regions are related to age-related disorders. All in all, the above observations clearly suggest that the interactive visual analysis of the MRI dataset leads to significant and interesting results that are very unlikely to be achieved using conventional analysis methods.

Above, we have presented only a subset of the analytical studies that we performed on this dataset. The overall analysis benefits highly from the comparison and the evaluation of the computational analysis results that are performed locally. We demonstrate that our methods are helpful in exploring new relations that provide a basis for building new hypotheses.

6 Discussions

To adopt our approach, the experts need to have a deep understanding of the statistics and computational tools that are employed in the analysis. This makes the learning curve of our system steeper than classical visual analysis systems. However, we observed that our tool could easily be integrated into the working pipeline of neuroinformaticians and neuropsychologists. These experts who analyze such complex datasets normally make use of computational analysis tools such as Matlab or R [184] and have an overall understanding of computational analysis. And compared to these systems, our solution is much more intuitive thanks to the support from interactive visual methods in the use of computational tools. We even state that such a tool can easily serve as an educative tool to train scientists in multivariate computational analysis. However, clear instructions and a video demonstration of an analysis of a simple dataset is regarded as highly important. One suggestion to improve the usability of the system is to further exploit the integration of R and develop a modular system that is accessible also for the domain experts. In order to get a clearer image of the requirements, a formal user study is needed. Such a study could lead to simplifications in the analysis process. To make the high-level operations more accessible and traceable, we need to devise special methods where the outcomes of the iterative steps are visually abstracted through a work-flow like interface. Such abstractions can also play a role in the presentation of the results and improve the usability of our system.

Different visualization methods such as parallel coordinate plots could also be incorporated to visualize the factors together with the original dimensions. One possible method to achieve this is to use hierarchical parallel coordinates, suggested by Fua et al. [64]. At several stages in our analysis, we are building new factors using a subset of factors, which implies that we are creating a hierarchy of factors. In our present realization, we only visualize the relations between the factors and the raw dimensions. Augmenting the visualization with such a hierarchy can likely lead to additional insight. Hierarchical difference scatterplots, as introduced by Piringer et al. [145], is a powerful technique to visualize such hierarchies.

Apart from the present case of healthy aging, the applicability of our tool could also be explored in the broader context of open access brain mapping databases such as BrainMap [119] and NeuroSynth [138]. These databases provide imaging data and meta-data from several thousand published articles available for meta-analyses and data mining, and thus are suitable for visual and explorative analysis methods.

7 Conclusion

With our method, we present how the structures in high-dimensional datasets can be incorporated into the visual analysis process. We introduce representative factors as a method to apply computational tools locally and as an aggregated representation for sub-groups of dimensions. A combination of the already available information and the derived features on the dimensions are utilized to discover the structures in the dimensions space. We suggest three different approaches to generate representatives for groups with different characteristics. These factors are then compared and evaluated through different interactive visual representations. We mainly use dimension reduction methods locally to extract the information from the sub-structures. Our goal is not to solely assist dimension reduction but rather to enable an informed use of dimension reduction methods at different levels to achieve a better understanding of the data. In both of the analysis examples, we observe that the results of the analysis become much more interpretable and useful when the analysis is carried iteratively on local domains and the insights are joined at each iteration.

The usual work flow when dealing with such complex datasets is to delimit the analysis based on known hypotheses and try to confirm or reject these using computational and visual analysis. With the advent of data generation and acquisition technologies, new types of highly complex datasets are produced. However, when these datasets are considered, little is known a priori, thus data driven, explorative methods are becoming more important. Our interactive visual analysis scheme proved to be helpful to explore new relations between the dimensions that can provide a basis for the generation of new hypotheses.

Acknowledgments

We would like to thank Peter Filzmoser for the discussions and comments on the project. We would also like to thank Paolo Angelelli for helping with the communication and preparation of the dataset. The aging study was supported by grants from the Western Norway Regional Health Authority (# 911397 to AJL and # 911593 to AL).

Paper C

Outlier Dimensions: Outlier Aware Analysis of High-dimensional Data

Cagatay Turkay[1], Paolo Angelelli[1],
Peter Filzmoser[2], and Helwig Hauser[1]

[1]Department of Informatics, University of Bergen, Norway

[2]Department of Statistics and Probability Theory, Vienna University of Technology, Austria

Abstract

In many application fields, datasets with a large number of dimensions per data item, such as hundreds or more, are posing special challenges for analysts. Very few visual analysis methods can handle such high numbers of dimensions effectively. Moreover, the dimensions usually have diverse characteristics due to the distribution of the values they contain. As a result of their characteristics, dimensions tend to form groups and hierarchies based on their similarities, or some are outliers due to special properties. In this paper, we investigate the relations within the dimensions with a special focus on those which stand out and thus can likely be considered as *outlying dimensions*. We determine the factors that lead to the outlyingness of dimensions and suggest a selection of relevant statistics to investigate these factors. We then devise interactive visual analysis methods to enhance the inspection of outlier dimensions. This new notion of outlier dimensions together with the interactive methods lead to statistics-assisted, interactive, visual, and, in particular, outlier-aware analysis strategies. We demonstrate the applicability of our approach to different types of datasets, especially in the context of a study of healthy brain aging, performed over a large group of participants.

This article is in submission to: *IEEE Transactions on Visualization and Computer Graphics*, 2013.

1 Introduction

High dimensional datasets with very large dimension counts, such as hundreds or thousands, pose special challenges for analysts. Very few visualization methods exist, for example, to properly encode the much lower-dimensional information in such data. Thus, analysts often refer to computational analysis tools to achieve a better understanding. However, these methods also often fail to provide reliable insight into such datasets. Consider, for instance, clustering a 500-dimensional dataset (a 2D data table with 500 columns) using the popular K-means algorithm [181]. It is not straightforward at all to correctly interpret the resulting clusters when the computations are done on a 500-dimensional space, neither is it possible to judge the reliability of the clusters when the distances between the items are computed by a 500-dimensional distance metric [118]. This issue with distance measures is known as the "curse of dimensionality" that states the fact that distances between items lose their meaning in high dimensional spaces [45]. Similar problems arise with most of the common computational tools. Hence, using these tools "securely" on such datasets calls for methods where the analyst has a better understanding of the set of dimensions and steers the analysis accordingly.

One of the key observations that we have made regarding the set of dimensions is that this set is usually heterogeneous. A single large subgroup or several smaller subgroups of these dimensions may contain related data and thus be highly correlated with each other. Also, there may be dimensions that have "special" characteristics that are not shared with the others. When analyzing high dimensional datasets, understanding the related groups of dimensions and those that *stand out* from the rest is highly important. In this paper, we focus on understanding these *outlier dimensions.* We are motivated by the fact that outlier dimensions can easily skew and/or dominate the results of computational analysis tools. An example of this is PCA, where dimensions with very high variance tend to be highly expressed in the results, suppressing the structures in dimensions with low variances [30]. As for this example and for others that involve the use of computational tools, being aware of outlier dimensions could improve the analyses significantly.

The idea of considering outlier dimensions is intriguing, however analyzing the outlyingness of dimensions is not straightforward. This is mainly due to the fact that the current outlier analysis methods operate mostly on data items [88]. To overcome this, we need methods to characterize dimensions and carry out the analysis by taking these characteristics into account.

In this paper, we now present a methodology to analyze high dimensional datasets with a special consideration of *outlier dimensions.* The contribution of this paper is an outlier aware analysis process for high dimensional datasets that answers three fundamental questions:

- How to define and characterize an outlier dimension?
- How to determine outlier dimensions?
- How to approach outlier dimensions once they are determined?

In order to define outlier dimensions, we suggest a categorization of outlier dimensions based on the different characteristics they carry. In the light of these categories, we determine a selection of statistics and features to characterize the different types of outliers. To investigate the *outlyingness of dimensions*, we introduce a number of interactive visual analysis methods, such as *z-Score view* and *data depth brushing*, that incorporate state-of-the-art mechanisms from statistics and the data mining literature. We then discuss analysis strategies to handle outlier dimensions. We relate these strategies to the categorization of outliers and demonstrate their utilization through several analysis examples.

Throughout the paper, we analyze a number of datasets. From Section 3 to Section 6, we describe the details of our approach along with an analysis of the *US communities and crime data* [12]. We then analyze an *artificial dataset* in Section 7, the *communities and crime* dataset again in Section 8, and a *cognitive aging study* dataset [215] in Section 9 to demonstrate our methods.

2 Related Work

Multidimensional data analysis has been one of the most important problems in visualization. Surveys by Wong and Bergeron [211] and by Fuchs and Hauser [65] provide an overview over the spectrum of available techniques in visualization. There are a number of visual analysis frameworks that enable the analysis of high-dimensional data by linking & brushing coordinated multiple views. Examples of such frameworks are the XmdvTool [202] or Polaris [178], now Tableau [180]. An elaborate mechanism for multivariate data analysis is proposed by Weaver [203]. In his work, he presents a methodology to explore the cross-filtering of data that are visualized in different types of views.

Statistical analysis methods are increasingly often integrated into the visual analysis process to help the analysts cope with the high number of dimensions. Jänicke et al. [94] present a method where two-dimensional projections of multivariate datasets, called attribute clouds, are the main medium for exploration. Williams and Munzner [210] present how computational power is guided via user-interaction in the multidimensional scaling of a multivariate dataset. A statistics-based framework that utilizes a query-driven analysis pipeline is presented by Gosink et al. [70] to explore how combinations of variables behave under different queries. In a recent work [51], Endert et al. propose observation level interactions to steer statistical analysis tools. Johansson and Johansson [99] propose the interactive dimension reduction through quality metrics. They also extended this method with measures specific to microbial populations in a later work [57].

The structure of high-dimensional datasets and the relations between the dimensions have been investigated in a number of studies. Seo and Shneiderman devise a selection of statistics to explore the relations between the dimensions in their Rank-by-Feature framework [168]. They rank 1D or 2D visualizations according to statistical features to discover relations in the data. However, in their method, the main focus are the data items rather than the dimensions. In VHDR [213], Yang et al. analyze the relations between the dimensions to create a hierarchy which they later use to create lower dimensional spaces. In their approach, they study the relations between the dimensions only in terms of a similarity measure. In the Value and Relation (VaR) display by Yang et al. [212], the authors represent the dimensions with glyphs projected into a 2D visualization. However, in terms of investigating the relations between the dimensions, their method is limited to displaying the correlation relations between the dimensions.

In order to apply our methods jointly on data items and the dimensions, we base our work on the dual analysis framework proposed by Turkay et al. [189] which we also discuss later in the paper in more detail. Such a joint analysis is also utilized in other problem domains, such as parameter space navigation [17], temporal data [9] and multi-run simulation data analysis [109].

Although outliers are frequently studied in data mining and statistics [88], they have not been the focus of many studies in visual analysis. One of the most prominent works concerning the treatment of outliers is by Novotný and Hauser [140] where they distinguish trends and outliers in their parallel coordinate plot. They represent the trends in the data as context and handle the outliers separately in the visualization. This work clearly shows that outliers need a special treatment in visual analysis. Another important study on outlier analysis is by Kehrer et al. [106], where the authors put a special emphasis on outlying observations by exploring the datasets through the use of robust statistics. In a recent study, Kandogan [101] discusses how trends and outliers can be detected via the Just-in-Time analytics. In all of these studies, however, the focus of the methods is on observations rather than on the dimensions. In our paper, we extend the literature by the visual analysis of outlier dimensions and present their utilization in the analysis of high-dimensional data.

3 Outlier Dimensions

In order to analyze the outlyingness of dimensions, we investigate the different properties of dimensions in comparison to set of dimensions in the data. The dimensions that are "special" and stand out with respect to certain features can be regarded as *outlier dimensions*. We construct a concrete definition of outlier dimensions by presenting a categorization of outlier dimensions with respect to different characteristics they carry.

The investigation of the characteristics of dimensions is based on the construction of a statistics space using a set of carefully selected statistics. In other words, for each dimension we derive a feature vector, whose values are either selected statistics or derived information computed using the original data. If we assume that our dataset is a two-dimensional table with n items (rows) and p dimensions (columns), we derive a $p \times k$ table S by assigning k values to each dimension. Once we construct the statistics table S, we analyze this table together with the data items using the dual analysis framework [189].

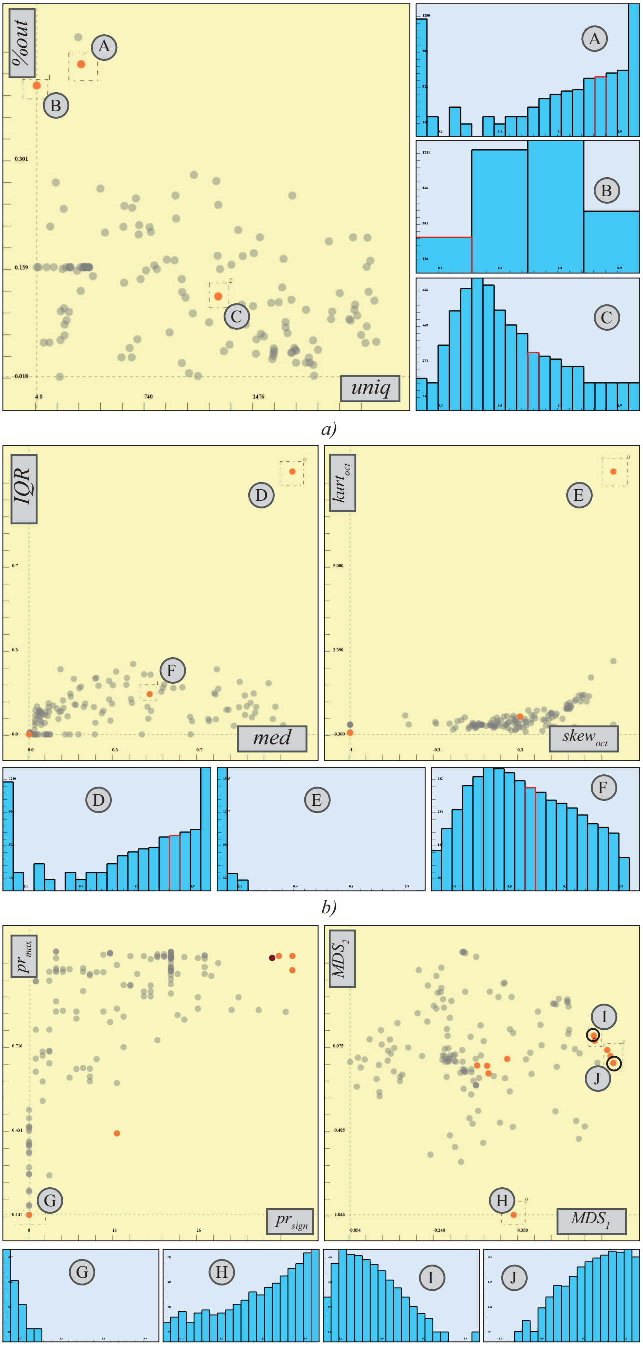
However, prior to building this derived table S, we need to determine the proper set of statistics. Therefore, we start by introducing the different perspectives that could lead to an outlier dimension. After we determine the different characteristics of outlier dimensions, we populate S with appropriate statistics that enable us to observe the dimensions in these perspectives.

In order to illustrate our methods in the following sections, we analyze a high-dimensional dataset as an example, obtained from the UCI machine learning data repository, which represents the socio-economic values and crime statistics for the communities in the US in the year 1990 [12]. The dataset consists of 142 dimensions (variables) and 2215 items, where each item corresponds to a community in the US. 18 of the dimensions contain the crime statistics, such as the number of murders or robberies, and, they are considered as the dependent variables, i.e., variables to be predicted. The other 124 variables contain the socio-economic statistics and can be grouped in five semantic groups, namely, *demographic*, *income*, *accommodation*, *family life*, and, *security force*. The main goal of the analysis of this data is to understand the relations between socio-economic indicators and crime statistics.

3.1 Types of Outlier Dimensions

Here we present three categories of outlier dimensions based on the sources of outlyingness. In addition, we give examples for each category through the use of appropriate statistics. The categorization provides a guideline on how to choose proper statistics and tools that will enable the analysis of outlier dimensions.

Characteristic outliers – The first perspective in the evaluation of the outlyingness of dimensions is to consider their characteristic properties. With characteristic properties, we refer to the inherent properties of dimensions such as the type of data values (numeric, textual, etc.), the number of missing data values, or, the percentage of 1-dimensional outliers. If we consider a 22-dimensional dataset where 20 of the dimensions have continuous data values (e.g., floating point numbers) and two of them have categorical data, the latter ones can be considered as *characteristic outliers*.



a)

b)

c)

Figure 1: Three types of outlier dimensions: a) Characteristic outliers b) Distribution based outliers c) Structural outliers. For each category, examples of dimensions which can and cannot be considered outliers are marked with corresponding histograms. For the marked dimensions in the scatterplots, an histogram is shown with the same label.

In Figure 1-a, examples of characteristic outliers are determined through a scatterplot of *the number of unique values in each dimension* ($uniq$) and *the percentage of univariate outliers* ($\%out$) (refer to Section 3.2 for details on $uniq$ and $\%out$). Histogram A (for the dimension marked A in the scatterplot) shows the frequency of the values of the *percentage of people living in urbanized areas*. The reason that this dimension has a high $\%out$ value (i.e., contains many items classified as 1-dimensional outliers) is the bi-modal shape of the distribution. And this bi-modality is due to the fact that all the people live either in urbanized areas or all in the countryside in many communities. Whereas Histogram B shows that the dimension *median number of bedrooms* has a limited value range and has a different characteristic, i.e., being categorical, compared to the rest of the dimensions. *The percentage of population that is 12-29 in age* dimension cannot be considered as a characteristic outlier and the distribution of its values is shown in Histogram C.

Distribution based outliers – The second type of outliers is related to the distribution of the items in a dimension. This type encompasses the dimensions that have distinct distributions compared to the rest of the dataset. For example, if most of the dimensions in one dataset are normally distributed and there is a couple of dimensions that are uniformly distributed, these dimensions could be considered as *distribution based outliers*.

To exemplify distribution based outliers, we make use of the median (med), the inter-quartile range (IQR), a robust version of skewness ($skew_{oct}$) and kurtosis ($kurt_{oct}$) (details in Section 3.2) in Figure 1-b. Here, one dimension that clearly stands out is the *percentage of people living in urbanized areas* (Histogram D) and this is due to its bimodal distribution as discussed in the previous outlier type. This dimension could thus be considered both as a characteristic and as a distribution based outlier. The number of homeless people counted in the streets dimension (E) is affected by 1-dimensional outlier items (most strong being NYC) which make the distribution stand out in terms of its skewness and kurtosis. In contrast to these two dimensions, Histogram F depicts the values of the dimension, *percentage of people employed in management or professional occupations*, which has a more expected distribution.

Structural outliers – In high-dimensional datasets, it is often the case that dimensions have various forms of correlation with each other. If a single dimension, or, a group of dimensions that are very strongly correlated, with very little correlation to the rest of the data, then this dimension or these dimensions can be marked as of type *structural outliers*.

In order to exemplify such outliers here, we make use of the pairwise correlation values between the dimensions and compute statistics, pr_{sign} vs. pr_{max}, to indicate how much a dimension is correlated with the others (details in Section 3.2). Moreover, we also use the pairwise correlation matrix as a distance matrix for a *multidimensional scaling* (MDS) operation. In Figure 1-c, we identify a dimension with very low values that indicates very low correlations with the

Table 1: Types of outlier dimensions with statistics/methods used to determine the different types

Type	**Statistics/Methods to determine**
Characteristic outliers	$uniq$, $\%out$
Distribution based outliers	μ, σ, $skew$, $kurt$, med, MAD, IQR, $skew_{oct}$, $kurt_{oct}$, $skew_{MAD}$, $kurt_{MAD}$, $norm_{shp}$, dip
Structural outliers	pr_{max}, pr_{min}, pr_{sign}, sp_{max}, sp_{min}, sp_{sign}, MDS, Clustering

rest of the data (Histogram G). This indicates that this dimension, *percentage of people who speak only English*, has a distinctive distribution. Similarly, using a 2-dimensional MDS projection of the dimensions, we figure out two groups that are correlated with each other but not so much with the rest (marked with circles, I&J). Histograms I and J reveal that one of the groups consist of dimensions with positive skewness while the other with negative skewness. Additionally, the dimension *per capita income for native Americans* has very low correlation with the rest of the data, possibly due to the 1-dimensional outlier items it contains (Histogram H).

3.2 Constructing the Statistics Table

The above classification of outlier dimensions provides a guideline on what type of statistics are needed to investigate the outlyingness of dimensions. Here, we list a number of measures (statistics/derived) that are useful for the analysis. We characterize the measures according to the type of information they provide and organize them in four categories. Table 1 lists the outlier types together with measures that determine the three types.

One important point to mention is that we also consider the robust versions of statistics. The field of robust statistics aims at statistical estimates and methods that are more resistant to outliers [59]. In the categories below, we accordingly include also the robust versions of statistics.

Category 1 - Characteristics of dimensions – In order to distinguish between different scales of measures, i.e., whether dimensions are categorical or continuous, we count the number of unique values in each dimension ($uniq$). For categorical dimensions, the $uniq$ value tends to be low and for continuous dimensions $uniq$ is often close to the number of rows n. We also compute the percentage of univariate outliers in each dimension, denoted by $\%out$. This value represents how "contaminated" a single dimension is. We use a median/MAD based method [106] to determine the percentage of outlier items in each of the dimensions. In this method, for a single dimension, each data item is assigned a robust z-score [106] and those that fall outside the $[2, -2]$ interval are marked as potential outliers. Afterwards we count the number of potential outlier items in each dimension and set the $\%out$ value accordingly. Moreover, the associated meta-data on dimensions, when available, could also be used to characterize the dimensions. The measures in this category help the analyst to determine *characteristic outliers.*

Category 2 - Summary of the distributions – Descriptive statistics are used frequently in data analysis to summarize much of the information in the data [100]. The basic descriptive statistics that we consider are: mean (μ) that is estimated by the average value, standard deviation (σ) that is a measure of dispersion, skewness ($skew$) that is a measure of a-symmetry, kurtosis ($kurt$) that is a measure of peakedness and the quartiles that divide the ordered distribution into four equal groups. As a better estimate of the "central value" than the average value for approximating μ, we include the median (med) and for a robust estimate of the standard deviation, we consider the inter-quartile range (IQR) and the *median absolute deviation* (MAD) [91]. For robust versions of skewness and kurtosis, we include *octile-based* ($skew_{oct}$, $kurt_{oct}$) and *median/MAD-based* ($skew_{MAD}$, $kurt_{MAD}$) estimates [59]. For a more detailed description of these measures we refer to the paper by Kehrer et al. [106]. All of these measures carry important information on the shape of the distribution of items in each of the dimensions and they provide an intuitive basis to determine *distribution based* outlier dimensions.

Category 3 - Type of the underlying model – Most of the statistical analysis tools assume that the modeling distribution of a dataset is normal [58]. We include the p-value of the Shapiro-Wilk normality test [158] and denote it by $norm_{shp}$. The higher p-values indicate a better fit to the normal distribution. These values gives us the chance to compare the dimensions in terms of their underlying distribution model. We prefer normality test scores due to the fact that most of the multivariate analysis tools assume that the data is normally distributed. Therefore, it is important to assess the normality of dimensions. However, the list of tests can be extended with respect to other types of distributions. In addition, we test the distribution of dimensions for uni-modality using a method called dip test [82]. This test results in a high p-value when the distribution is uni-modal and lower values otherwise, we denote this by dip. All the statistics in this category help the analyst to detect *distribution based* outlier dimensions.

Category 4 - Uniqueness of dimensions – Correlations between the dimensions are important in understanding the relations between the dimensions. To represent the correlation relation between the dimensions, we first calculate

Pearson correlation coefficients [33] between all pairs of dimensions, denoted by $pr(d_i, d_j)$ where $1 \leq i, j \leq p$ and $i \neq j$. For each dimension d_i, we then find *the maximum correlation* and *the minimum correlation* values to all the other dimensions, represented by pr_{max} and pr_{min}. Moreover, for each d_i, we count the number of dimensions that are correlated with d_i above a certain threshold. This threshold can either be set by the user or computed automatically depending on the distribution of the correlations. In our analysis, we set this threshold to 0.6 and check this against the absolute values of the correlations. We denote this value as *significant correlation count* pr_{sign}. Additionally, we perform all these computations using Spearman's rank correlation coefficient, which is a robust measure and also considers non-linear relationships between the dimensions [33]. These values, then, are denoted by sp_{max}, sp_{min} and sp_{sign}. All the values in this category enable us to determine the dimensions which are unique or share common structures with the others, i.e., *structural outliers.*

4 State of the art methods to determine outlier dimensions

In order to enrich the outlier-aware analysis, we incorporate three different methods to facilitate the determination of outliers. These methods are based on the utilization of different outlyingness measures (for data items) from statistics. Since we focus on dimensions in this paper, we use these measures to evaluate the outlyingness of dimensions. All these measures are computed using the S table which has k values (i.e., statistics) for each of the p dimensions. Depending on how many of the k statistics are considered, we resort to different methods for the evaluation of outlyingness.

4.1 z-Score view

Dimensions can be outlying with respect to a single statistic, e.g., if the σ values of all the dimensions are considered, dimensions with exceptional σ values are considered outliers with respect to σ. In order to determine the outlyingness of a dimension d_i with respect to a single statistic s, we compute the z-scores using the robust median/MAD method as

$$z_i^s = \frac{d_i^s - med(d_1^s, \cdots, d_p^s)}{MAD(d_1^s, \cdots, d_p^s)}$$

where $i = 1, \ldots, p$ and $s = 1, \ldots, k$. Moreover, d_i^s denotes the values of the statistics s for dimension i. Also note that med is the median and MAD is the median absolute deviation (introduced in 3.2) of the s values of all the dimensions. Similar to the literature where z-scores are used to mark certain data

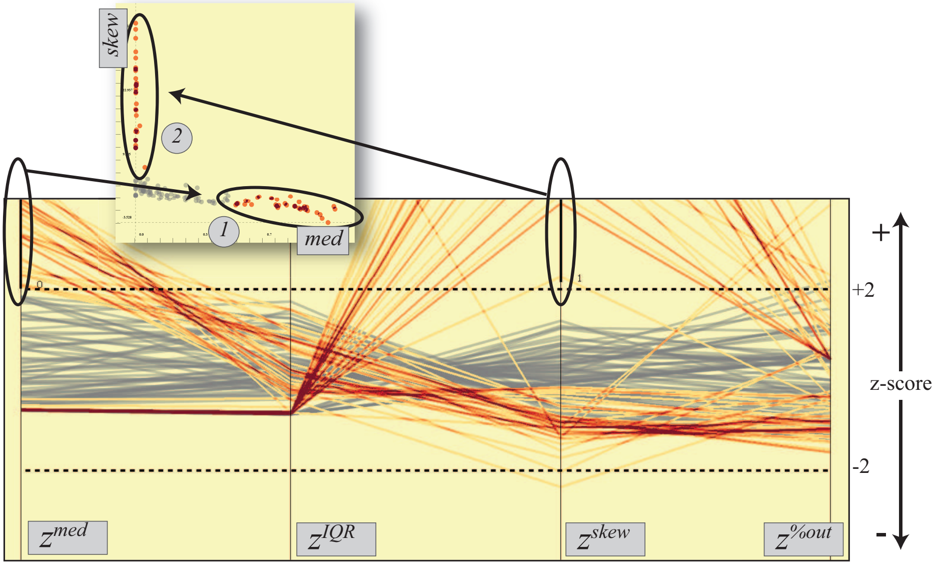


Figure 2: Our z-score view to visualize the z-scores for the dimensions over: *med*, *IQR*, *skew*, and, *%out*. Here, each line is a dimension and the dashed lines indicate the $[-2, 2]$ interval to ease the selection of potential outlier dimensions. The accompanying scatterplot shows the selected dimensions that are possible outliers w.r.t. *med* (marked 1) and *skew* (2).

items as potential outliers [150], dimensions with z-scores lying outside the $[-2, 2]$ range can be treated as potential outliers. We compute the z-scores for all the dimensions for all the k statistics and visualize these values through an extended parallel coordinate plot called the *z-score view*. In this view, shown in Figure 2, each axis corresponds to the z-score values that are computed for 4 different statistics, *med*, *IQR*, *skew*, and, *%out*. Note that here, each line corresponds to a dimension. We enhance the view with two dashed lines that pass through -2 and 2. This means that the dimensions that are above or below these lines are candidates for being an outlier for a particular statistic. In the figure, a group of possible outliers w.r.t. *med* (marked 1) and *skew* (2) are selected. The scatterplot reveals that the selected dimensions indeed have *med* and *skew* values that are much higher compared to the most of the dimensions.

4.2 2-Dimensional data depth

When we observe the outlyingness of dimensions with respect to two statistics, we usually visualize the dimensions on a scatterplot as opposed to these statistics.

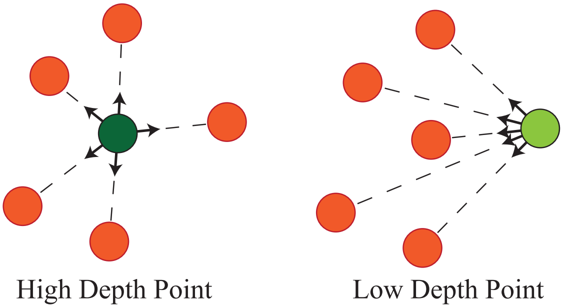


Figure 3: Illustration of the L_1 depth computations. The unit vectors between a dimension d_i (i.e., a point on the scatterplot) and all the other dimensions are found. The average of these vectors are used to compute data depth (see Eq. 1). Central data points have higher D values (left) and points on the edges of the distribution have lower D values (right).

In order to support the identification of outlier dimensions through scatterplots, we enhance them with *data depth* calculations. In the outlier analysis of data items, data depth is one of the widely used methods [88]. Depth of a data item represents how central it is with respect to the distribution of the other items. In the literature, there are a number of suggested methods to compute the depth of data items and in this paper we employ the L_1 depth [199] to compute the depth of dimensions. The L_1 depth for a dimension d_i with respect to statistics s_1 and s_2 is computed with:

$$D_i = 1 - \left\| \frac{1}{p} \sum_{j=1}^{p} e_{ij} \right\| \tag{1}$$

where $\|.\|$ is the Euclidean norm and e_{ij} is the unit vector between d_i and d_j with respect to their s_1 and s_2 values , computed as $e_{ij} = (d_i - d_j)/\left\|d_i - d_j\right\|$. The D value is close to 1 if the dimension lies at the center of the plot and close to 0 if it is on the edge. Figure 3 illustrates how the above formula distinguishes between points that are central and that are lying on the edges of the distribution of points on the scatterplot.

Whenever we bring up a scatterplot of dimensions with depth enhancement, we compute the depth values for all the dimensions. We then map the color of the points on the scatterplot to the associated data depth values. On Figure 4, the depth values for the dimensions are utilized to color the points in the scatterplot. The possible outlier dimensions have saturated green colors (e.g., point marked 1) and more central dimensions have less saturated colors. Notice that the depth computations are also linked to the selection mechanism (Figure 4-b). In this

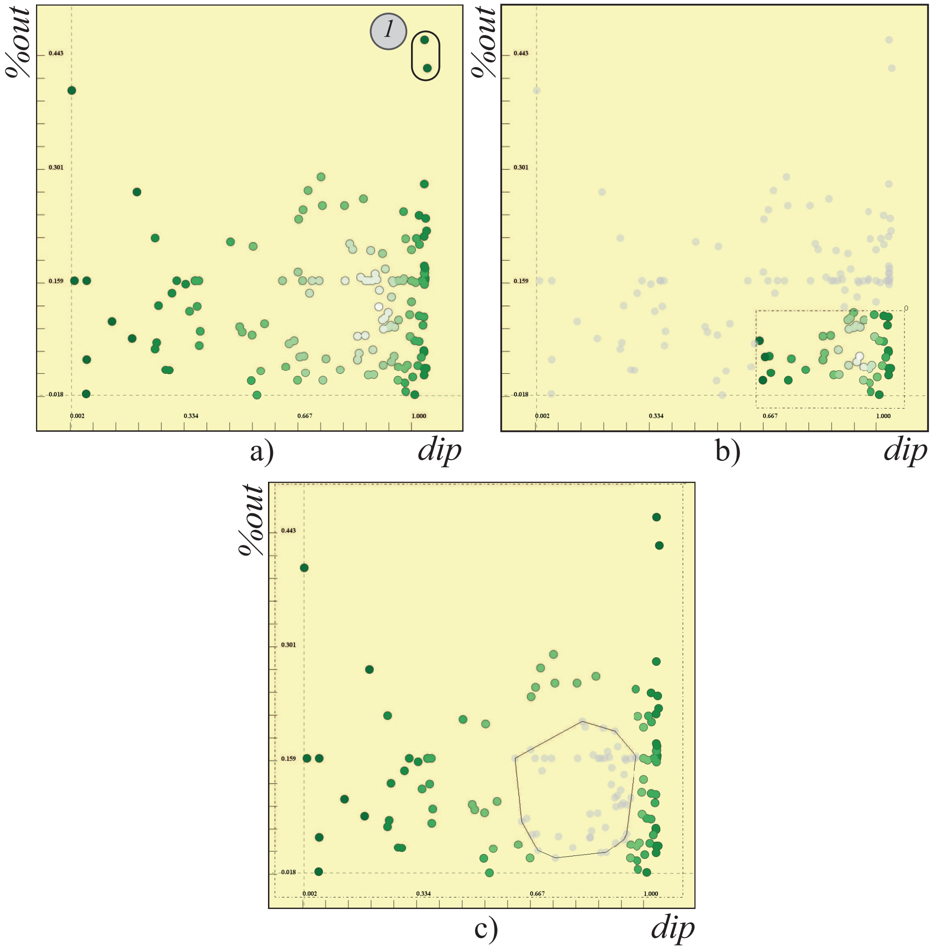


Figure 4: a) The dimensions are colored according to their depth values. The "deeper", i.e., central, points have a whitish color and the points on the outskirts (marked 1), i.e., possible outliers, have a saturated green color. b) Depth calculations are also done locally within the selection. c) Depth based brushes snap to different depth levels called depth contours to aid the selection of different structures in the data.

setting, the depth computations are done locally, i.e., limited to the selected points. This enables the user to assess the depth of the dimensions in local structures such as clusters.

Depth based brushing – In statistical analysis literature, depth values are usually used to categorize data points into layers called depth contours [160].

The inner contours consist of central points with high depth values and the points that lie in the outer contours are candidates for being outliers. Since selecting the outlier dimensions is highly critical in our analysis, we enhance the selection mechanism in scatterplots with depth-based brushes. These brushes enable us to easily (de)select points which are in the center or at the outskirts of the distribution of points. In order to achieve this, the user interactively determines a depth contour level l and the points with $D_i < l$ are selected. For convenience, we limited the number of levels to 10, i.e., intervals of 0.1 over the D values. In Figure 4-c, the depth contour brush leaves out the most central data points (the inner four levels, i.e., $l = 0.4$) and easily selects the dimensions with more *special dip* and *%out* values.

Although these types of depth contour selections could be done as a combination of smaller brushes, it is challenging to select these structures with conventional rectangular brushes. This is mainly due to the fact that depth values tend to create elliptical contours. More importantly, the suggested advanced brush automatically snaps to the depth contours and thus has a contextual mapping. Such a mechanism is much more meaningful and robust in selecting interesting structures compared to conventional brushes.

4.3 Mahalanobis distance

Mahalanobis distance computation (or scores) is one of the common multivariate outlier analysis method [88] for multi-dimensional datasets. The Mahalanobis distance is a multivariate distance measure, that gives the distance of a data sample to the center of a distribution by taking the covariance structure into account. In order to determine potential outliers with respect to 3 or more statistics at once, we refer to the Mahalanobis scores. Similar to the previous methods, the Mahalanobis distances for dimensions are computed using the derived statistics table S. Mahalanobis distance for dimension d_i is then computed by, $MD_i = \sqrt{(d_i - \mu_S)^T C_S^{-1} (d_i - \mu_S)}, 1 \leq i \leq p$ where μ_S is the k-dimensional mean vector and C_S is a $k \times k$ covariance matrix of the data table S (recall that we have k columns in S). These scores are then visualized and dimensions with exceptional values could be marked as outliers. And here, values larger than the 0.975 quantile of a chi-square distribution with k degrees of freedom could be considered as exceptional [88].

5 Interactive Visual Analysis Framework

The analysis of high dimensional datasets is performed through a coordinated multiple view setup that incorporates a linking & brushing mechanism with composite brushes, i.e., a combination of selections through Boolean operators. Additionally, we integrate multivariate analysis tools such as principal component analysis (PCA), multidimensional scaling (MDS), and, clustering in our analysis. We are able to apply these operations both on dimensions and data items. While we apply them on dimensions, we use either the transpose of the actual data (after normalization) or the statistics table S. And additionally, for MDS applied on dimensions, we use the pairwise correlations as an input distance matrix. In the resulting projection of the dimensions (assuming a 2D projection), the highly correlated dimensions are placed closed to each other.

One important point to mention is that all the computational tools operate on the current selections of items/dimensions, e.g., PCA is computed on only the selected subset of dimensions. In order to provide a wide selection of statistics and computational tools, we use *R - the statistical computing project* [184] as an integrated module.

5.1 Dual Analysis Model

In our system, we use the dual analysis model proposed by Turkay et al. [189] to perform the joint analysis of the items and the dimensions. In this model, the analysis is carried out in two linked visualization spaces, items space and dimensions space. In Figure 5-a we see the distribution of communities according to the *median income* and the *percentage of large households* variables. In the second scatterplot (Figure 5-b) each dimension is plotted against the the mean (μ) and standard deviation (σ) values that are estimated for all the normalized columns. A subset of the communities with lower incomes and larger households is selected (Figure 5-c). μ and σ values are re-estimated for all the dimensions using only the selected items. In dimensions space, the view (Figure 5-d) now displays the difference between the two estimated statistics for each dimension, i.e., computed using all and only the selected items. The visualization now displays the changes in the values. In such views, the dimensions that change the most (*outliers* in the context of this view) could be considered to show a strong relation to the selection of items. For instance, the *percentage of the population with Hispanic heritage* dimension (marked in Figure 5-d) shows a positive correlation with lower income and larger families. For the details of this view that shows the differences in statistics, please refer to the paper by Turkay et al. [192].

Preparing the analysis setup – Prior to starting the analysis, there are a number of steps that is taken. In addition to the loading of the data, we also load meta-data on the dimensions that also contains information on how missing values are encoded in the data. After the data loading, the initial step is the normalization of the data items in order to make them comparable. We follow the "Informed Normalization" steps discussed in our earlier work [191]. In this normalization scheme, depending on the analysis and the type of the dimension, one can prefer methods like scaling to unit interval or z-standardization. We then continue with the population of the statistics table S using the already mentioned

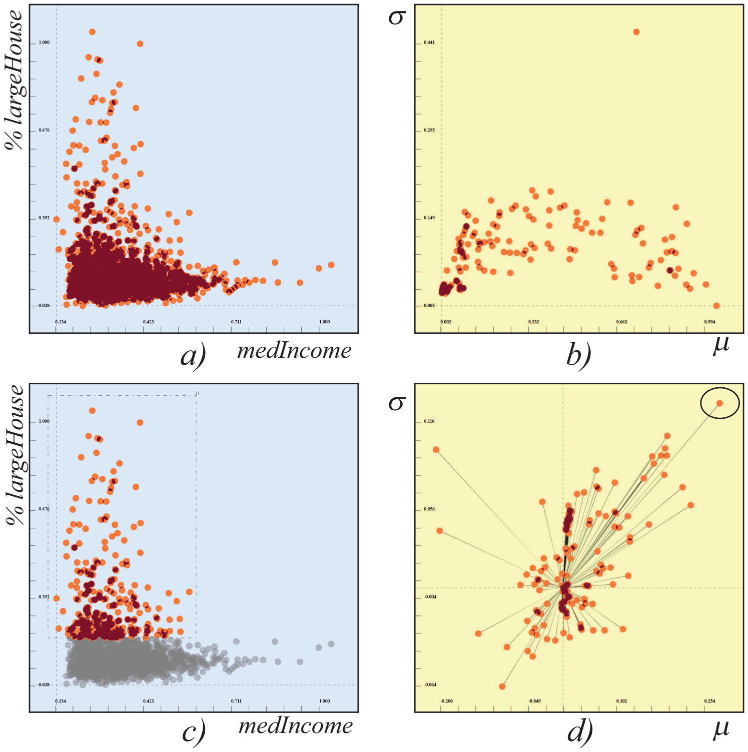


Figure 5: a) Scatterplot of data items with *median income* vs. *percentage of large households* values (blue background). b) A scatterplot where each point is a dimension with the estimated μ and σ values (yellow background). c) Communities with lower income and large households are selected. d) μ and σ values are re-estimated using the selected items. The change between the two values (after/before selection) are shown. The marked dimension has both higher μ and σ values for the selected subset.

statistics in Section 3.2 and investigate the outlyingness of the dimensions using the methods discussed in Section 4.

6 Outlier-aware analysis strategies

For a successful consideration of outlier dimensions in the course of high-dimensional data analysis, analysts need a number of strategies to handle the outlier dimensions. When we refer to high-dimensional data analysis, we refer to analysis routines that involves the integrated use of computational analysis tools, such as clustering or dimension reduction methods. Currently, there is only a literature on treating item-based outliers and two main strategies are suggested: removing the outliers or employing robust statistics to accommodate the outliers in the analysis [88]. In order to utilize outliers within the set of dimensions, however, we need a different set of strategies due to the characteristics that outlier dimensions carry as introduced in Section 3.1. We discuss 4 different strategies in relation to the categories of outlier dimensions.

S1: Leave out – This strategy simply suggests to leave out an outlier dimension from the analysis. This option is preferred in cases where it is known that the outlier dimension can cause the results of a computational tool to be less reliable, such as in the case of PCA or clustering. This strategy can be the main approach when dealing with *characteristic* outliers. As described earlier, this type of outliers either have distinct data types or contain many missing points/1D outliers which leads to unstable results when used with computational tools.

S2: Transform – This strategy involves the transformation of data items in a dimension when the outlyingness of a dimension is known to be caused by a problem in the data values, such as missing values or measurement errors. When such problems exist in the data, they usually cause dimensions to be either *characteristic* and *distribution-based* outliers. In this cases, we refer to the literature on “curing” item-based outliers [149]. Possible approaches include replacing missing data [163], transforming data items via methods such as log or inverse transformations [149], and fitting data distributions [149].

S3: Treat separately – In most of the cases, dimensions are considered outliers not due to problems in the data but due to their distinctive characteristics. In such cases it makes sense to handle the outlier dimensions separately and analyze them closely to understand the nature of their outlyingness. One can refer to conventional visual analysis methods and use histograms, scatter plots, or depth plots of the data items as introduced in Section 4.2. This strategy can be taken when *distribution-based* or *structural* outliers are determined in the analysis.

S4: Treat hierarchically – This analysis strategy involves the analysis of sub-structures and is related to the handling of *structural* outliers. Since structural outliers amount to highly-correlated sub-domains in the set of dimensions, they provide additional insight when handled separately. Such groups of dimensions can be used to create sub-domains where computational tools are applied locally,

Outlier Type	Strategy
Characteristic outliers	S1, S2
Distribution based outliers	S2, S3, S4
Structural outliers	S3, S4

Table 2: Strategies based on the type of the outlier dimension

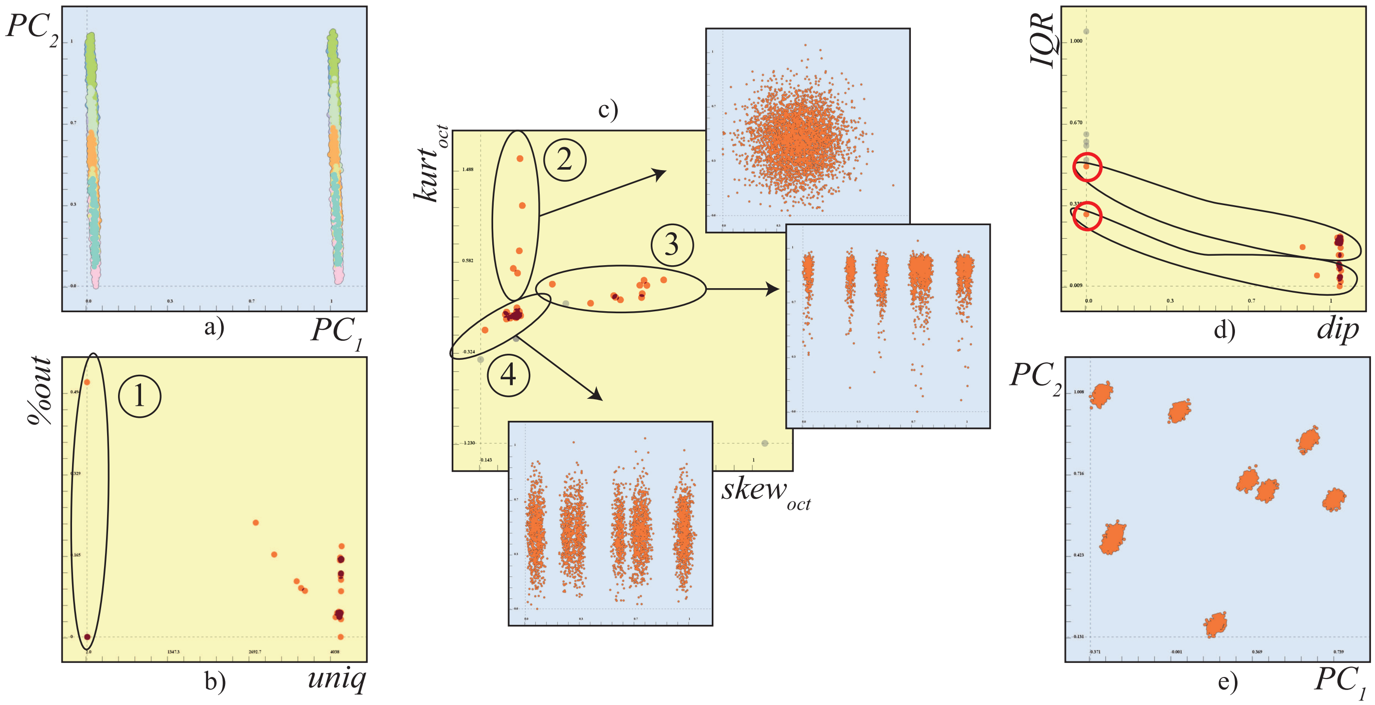


Figure 6: Analysis of a synthetic dataset with 8 clusters. The clusters are not represented in the PCA of all the dimensions (a). A number of characteristic outliers (marked 1) are removed in a *uniq* vs. *%out* plot (b). Three subgroups (2,3,4) are detected in the $skew_{oct}$ vs. $kurt_{oct}$ plot (c). PCA is applied locally on these groups and two of them (3, 4) contain apparent structures. When we observe the subgroups separately in the *dip* vs. *IQR* plot, there is a single dimension in each group (red circles) with a low *dip* value (indicating modality) and a high variance (d). These dimensions are those that encode the clusters (e).

e.g., applying PCA locally to find representative dimensions [191]. The analysis then continues by combining and comparing these local analysis results.

In the following analysis cases, we use a combination of these four strategies. We mark the strategy taken with the strategy name to make it easier to refer to the above strategies.

7 Analysis of a contaminated dataset

We demonstrate how the analysis strategies are utilized in an analysis of synthetic dataset with 37 dimensions and 4049 data items ($n \times p = 4049 \times 37$). We created this dataset by compiling together several modeled data dimensions that have different characteristics. We started with 2 dimensions which together encode 8 clusters. We extended this dataset by sampling 4049 data items from well-known distributions. We added the following 35 dimensions: 15 normally distributed, 10 log-normally distributed (5 left-skewed, 5 right-skewed), 5 categorical, and, 5 normally distributed with missing values. The initial clustering could be considered as the "hidden", relevant information and we demonstrate how this information could be extracted through our approach.

When we apply PCA using all the 37 dimensions in the data, we observe, in Figure 6-a, that the initial clustering information (denoted with the colors here) is not visible. Instead, there are two strong clusters, possibly due to one of the categorical dimensions. We start the investigation of the dimensions by checking for characteristic outliers through a *uniq* vs. *%out* plot. The five categorical dimensions stands out (marked 1) and we leave out these characteristics outliers from the PCA calculations (**Strategy S1**). We continue by analyzing the subgroups in the dimensions. In the $skew_{oct}$ vs. $kurt_{oct}$ plot, we detect three groups of dimensions with similar characteristics. A group of dimensions with high kurtosis but no skewness (marked 2, dimensions with missing values), another one with high skewness but no kurtosis (marked 3, the log-normal distributions), and, the last group showing no skewness or kurtosis (marked 4, the normally distributed dimensions). When we apply PCA on these groups locally, we observe that in two of these subgroups (3,4) there are apparent structures (**Strategy S4**). We observe the dimensions in these two subgroups separately over a *dip* vs. IQR plot. The statistics reveal one dimension per each group with a low *dip* value and a high variance, which indicates the existence of modalities, i.e., suitable to use in clustering. A final application of PCA shows that these two dimensions are indeed the two artificial dimensions that contain the clustering information (**Strategy S3**). The above example demonstrates how outlier dimensions could easily distort computational results and how a careful consideration of these outlier dimensions with our methods leads to a reliable analysis.

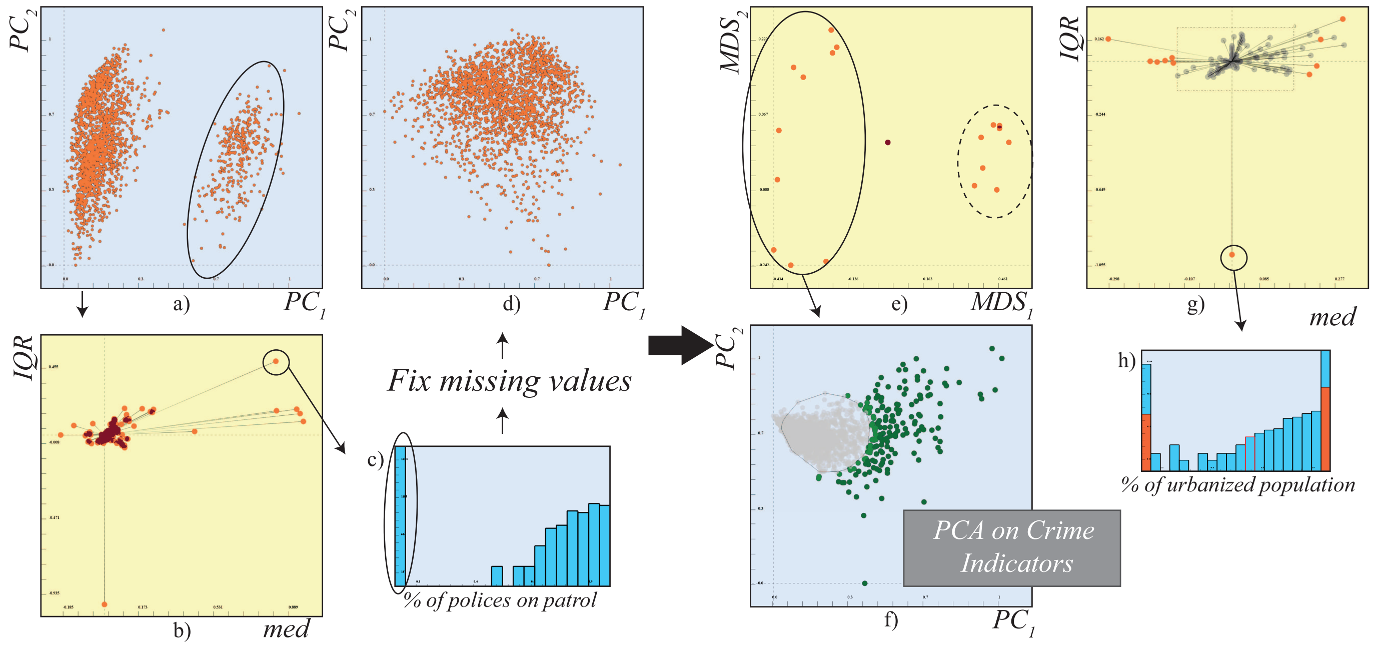


Figure 7: An analysis of communities and crime data. Only the independent dimensions are selected (not shown here) and PCA on these dimensions reveal two groups in the resulting first 2 PCs (a). After one of the groups (circled) is selected, the changes in *med* and *IQR* values are shown (b). The histogram for one of the dimensions (c) that "cause" the grouping reveals that there is a large portion of missing values (circled). After the missing values are handled, the grouping disappears (d). We continue by applying MDS on the dependent variables (e) and observe one highly correlated group (dashed circle). This group contains the actual numbers for crime indicators while the rest (circled) encodes the ratio of crime to the population. PCA is used to summarize the second group of dependent, crime-related variables, where a depth brush is used to select the item outliers, i.e., with higher crime rates (f). The changes in the statistics due to this selection reveals a number of variables that are related to crime (increasing/ decreasing *med* values) (g). A side finding is that crime statistics are only highly either %100 urban or %100 rural areas (h).

8 Use Case: Analysis of US Communities and Crime Data

In the analysis of the "communities and crime" data (introduced in Section 3), the goal is to find the relations between crime statistics and the socio-economic information. With the help of meta-data on the dimensions, we interactively select the independent dimensions and start the analysis by applying PCA to these dimensions. The resulting first 2 principal components (PCs) in Figure 7-a reveal two distinct groups. A visual inspection (not shown in the image) shows that there is no relation between the resulting principal components (PC) and the crime statistics. We select one of the groups and observe the changes in *med* and *IQR* values in a linked view (Figure 7-b). A number of dimensions, with higher *med* values for the selection, stand out to be the cause of the clustering. The histogram for one of these dimensions, *the percent of polices on patrol*, is inspected, we observe that the distribution has a bi-modal distribution with a gap between the values. When we refer to the actual data, we find out that this is due to missing values. Before we proceed with the analysis, we replace the missing values (**Strategy S3**) with the median of each dimension [163]. In addition, a number of these dimensions are left out, due to the high number of missing values, i.e., 85% of all items (**Strategy S1**). The effect of missing value replacement can be observed on the new PCA results (Figure 7-d). The updated results show that the initial grouping is only due to special artifacts in the data.

We then focus our attention to the sub-group of dimensions related to crime statistics (18 dependent variables). An MDS of these results indicates structural outlier dimensions which are highly correlated (marked in Figure 7-e). Upon inspection we see that this group consist of dimensions which contain the absolute values of different crime types, whereas the rest of the dependent dimensions contain percentage values. We leave out the marked dimensions in Figure 7-e due to the high in-between correlation in order to achieve easier to interpret PCA results (**Strategy S1**). An alternative method at this point is to represent this group with a representative factor [191] and continue the analysis in a hierarchical way (**Strategy S4**). However, since the two groups carry similar information (absolute vs. percentage), we decide to leave this group out. We continue to compute PCA using the percentage based crime related dimensions to summarize the information in these variables (Figure 7-f). Using depth brushing on the projected items, we select the items with higher crime rates in Figure 7-f to investigate the relation between high crime rate and the independent variables. The *med* vs. *IQR* view (Figure 7-g) shows a number of dimensions that have lower or higher values in *med* (indicates negative or positive correlation). We found out that *the percentage of people under poverty level* is positively and *the percentage of the population that is Caucasian* is negatively correlated with crime levels (in this dataset). There are also dimensions related to the family structure, such as two parent families with kids, that show weaker negative relation to crime levels. This finding can be interpreted as an indication of the relation between the neighborhood and crime, i.e., less crime in places where families are more common.

In addition, one of these dimensions, *percentage of population living in urban areas*, shows a big drop in the variance of the values (lower IQR). When we observe this dimension closely, we find out that the higher crime rates occur in communities that are 100% urban or 100% rural (**Strategy S3**).

The above analysis shows that a careful consideration of outlier dimensions leads to the controlled use of computational tools. Which in turn, results in findings that could otherwise be hidden by features in the data that do not carry relevant information.

9 Use Case: Analysis of Cognitive Aging Data

Cognitive studies investigate the role that specific neural substrates have in cognitive functions, as well as the impact of healthy aging, dementia or other pathologies on human cognition [127].

In this use case, we work on a multi-modal dataset related to a longitudinal study of cognitive aging. The study involves healthy participants who underwent a neuropsychological examination (tests on IQ, memory, attentive and executive function) and were subjected to imaging of their brain with different modalities, namely, 3D anatomical magnetic resonance imaging (MRI), diffusion tensor imaging (DTI) and functional MR (fMRI). Here, we focus on the analysis of anatomical MRI images and try to investigate the relations between image derived measures and cognitive abilities [215].

In order to carry out the analysis, a set of structural statistical measures for brain regions are extracted automatically using a software called Freesurfer [62]. With each segment in the cortical and subcortical brain regions, a number of measures, such as *surface area*, *thickness* or *volume*, are associated. These processes create a very high dimensional dataset (hundreds of dimensions). This high-dimensionality of the resulting dataset poses a big challenge for the involved scientists. They usually have to delimit the analysis to a selected subset of segments based on an a priori hypothesis. However, when the analysis has more of an explorative nature, they need methods to understand the relations between the dimensions and discover different subsets to investigate further. In that sense, our method is beneficial to the researchers in discovering artifacts, groups and important features in this high-dimensional healthy brain aging study dataset.

The dataset involves 83 healthy individuals recruited through advertisements in local newspapers, as part of a larger study on cognitive aging [215]. For each subject, a T1-weighted image was segmented into 49 anatomical regions, and 7

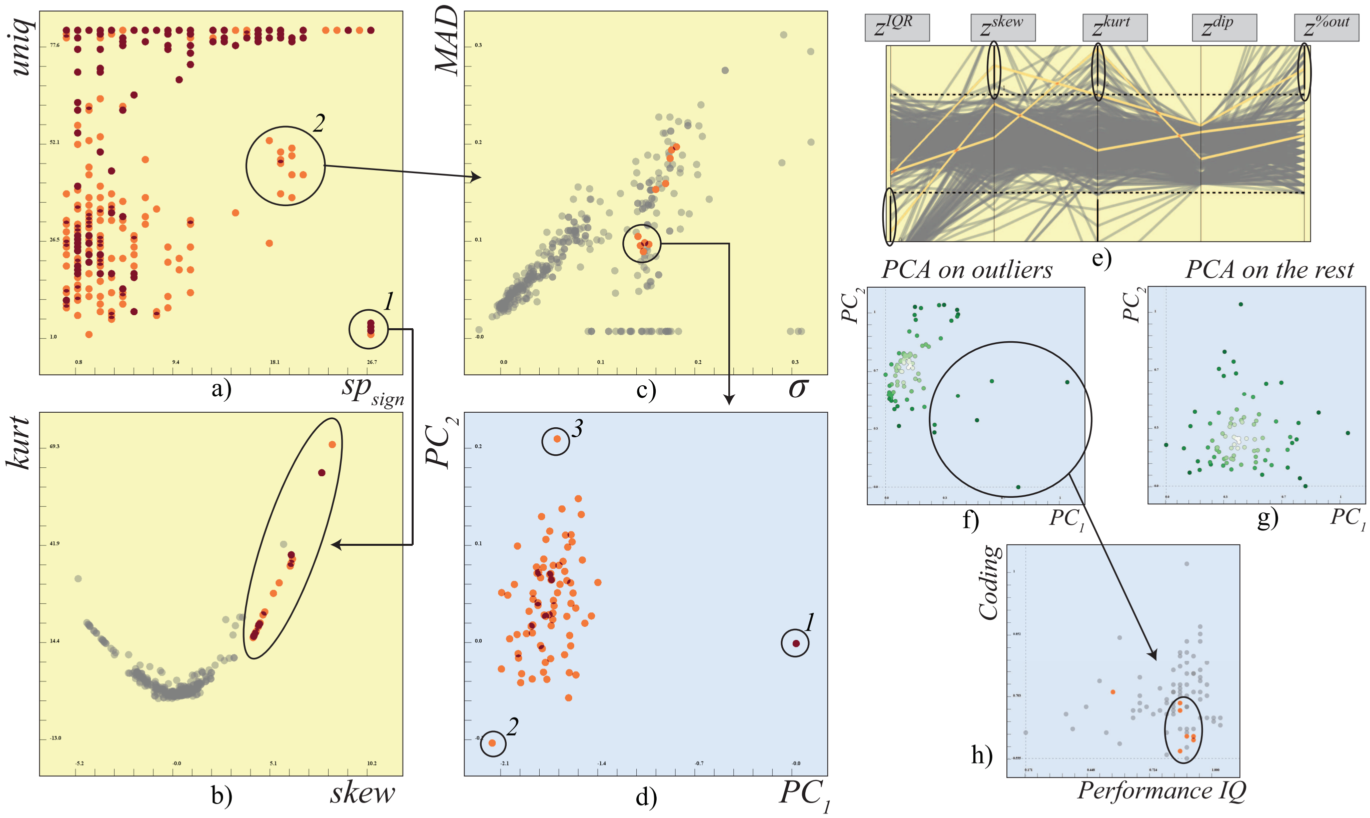


Figure 8: An analysis of the cognitive aging study dataset. In a plot of sp_{sign} vs. $uniq$, we observe a group of dimensions (marked 1) that are correlated with many dimensions, but have an exceptionally low item count. We check their $skew$ and $kurt$ values and see that they are also outliers in this plot (b). By referring to the data, we find out that they contain missing values for almost all the items. We focus on a second group of dimensions with high correlations (a-2) through a plot of σ vs. its robust counterpart MAD. We select the ones which are likely to contain interesting data items and apply PCA on the data using only the selected subset (d). The result now reveals participants (marked 1,2,3) that needs further investigation. In addition, the selections on z-score (e) reveals three outlier dimensions. We perform PCA on only those three (f) and the rest of the dimensions (g) and observe that the 7 outlying participants performed good in IQ tests while poorly in the coding test (h).

different measures were extracted for each region, automatically. This creates $49 \times 7 = 343$ dimensions per patient. In addition, details about each patient, such as age or gender, and the results of a memory test is added to this dataset. With this addition, the resulting dataset has 385 dimensions, i.e., the resulting table's size is 83×385.

We start the analysis by looking at the correlation structure between the dimensions. We bring up a view of sp_{sign} vs. *uniq* (Figure 8-a). The first group of dimensions that pops out in the visualization are those which have a very low number of unique values and a very high correlation count (circle-1 in Figure 8-a). We observe these dimensions further by looking at their shape characteristics via a *skew* vs. *kurt* plot (Figure 8-b) and see that they have abnormal values in this view as well (**Strategy S4, S3**). We see that these dimensions are related to a specific segment in the brain, white matter hypo-intensities of the left and right brain lobes. We then turn to the actual data items (the patients), and we observe that these dimensions are full of missing values for almost all the patients. This is very likely due to an artifact in the computation of these features, therefore the researchers need to double-check the related computations. In the current analysis we discard these dimensions since most of the values in these dimensions are missing (**Strategy S1**).

We continue the analysis by observing a group of dimensions which have similar unique value counts and high numbers of significant correlations (circle-2 in Figure 8-a). We find out that the selected dimensions all relate to the Cerebellum Cortex segment. We investigate this sub-group of dimensions further by observing them as according to σ and its robust counterpart MAD (Figure 8-c). For some of the selected dimensions, there is a difference between the robust and non-robust estimates of the variance. This indicates that these dimensions are likely to be contain abnormal measurements and this requires further investigation. We continue by applying PCA on the dimensions by only using the dimensions marked with a circle in (Figure 8-c). The results of the PCA now reveals some interesting patterns in Figure 8-d. The data items marked 1 (multiple items mapping to the same point) are again due to missing values in the computations. However, data items (i.e., patients) that are marked with 2 and 3 have abnormal statistics about their cerebellum cortex which makes them interesting subjects for a deeper analysis.

We elaborate our analysis by using the z-Score view to visualize the z-scores for IQR, *skew*, *kurt*, *dip*, and $\%out$ (Figure 8-e). Here, we first delimit our interest to the dimensions related to the volume of the brain segments (by selecting meta-data on dimensions through a histogram not shown in the image) and the selections on z-Score view (circled) reveals three dimensions: *left inferior lateral ventricle*, *5th ventricle* and *non white matter hypointensities*. In order to see the distribution of items in relation to these segments, we apply PCA on these 3 segments (Figure 8-f) and the rest of the segments (only using those related to volume) (Figure 8-g). We see that while the first projection contains outlier items, the second has a more regular distribution. We select the 7 outlier items (participants marked with a circle in Figure 8-f) in the first projection. After an inspection of the neuropsychological test scores, we see that 6 of these participants performed very good at IQ tests but performed poorly in coding test that is an indication of memory function. This result can be interpreted as an indication of the relation between the 3 selected brain regions and performance of the participants in IQ and memory function tests (Figure 8-h).

Even though we cannot present all the analytic procedures which we executed on this highly interesting dataset, we still understood very quickly that our methods were helpful in discovering groups in dimensions that are relevant for an analysis w.r.t. their different characteristics. And we see that when we utilize these discovered structures, we are able to single out data items with special properties.

10 Discussions

Finding the dimensions (variables) that carry distinctive or redundant information is an important research in fields such as machine learning and data mining and usually referred to as feature selection [77]. In these methods, the selection of features is done algorithmically based on specific measures. For instance, there are variable ranking criteria, such as correlation, predictive power, or, information theoretic, on which the feature selector is constructed upon. All these different perspectives on the variables (dimensions) could easily be incorporated in our framework to describe the dimensions. Our methods enables the interactive and visual inspection of these different perspectives and lead to a more diverse analysis of the dimensions.

There is an extensive collection of outlier analysis methods in statistics and data mining. The most common approaches can be categorized as: statistical tests, data depth based, deviation-based and density-based approaches [88]. However, these methods focus only on the observations, therefore our new analysis perspective on the dimensions broadens the focus of current analysis literature.

Although the selected statistics in the presented method already provide a quite general framework for a wide range of analysis tasks, this selection is by far not exhaustive. Depending on the nature of the analysis, the statistics space (S) can be populated with alternative features. Especially when dealing with domain-specific problems, one can include specific measures in S to aid the analysis.

The notion of outlier dimensions is slightly different than the common interpretation of item based outliers. In the context of outlier analysis of items, apart from a few special cases such as intrusion detection, outliers are usually treated as artifacts and left out of the data. However within the set of dimensions, outliers usually play an important role and we consider these dimensions by introducing analysis strategies other than just discarding the outlier.

11 Conclusion

In this paper, we present the concept of outlier dimensions and demonstrate the role of these dimensions in high-dimensional data analysis. With an outlier dimension, we refer to a dimension that stands out with respect to certain statistics and/or derived features. To the best of our knowledge, our method is the first method that enables an outlier analysis of dimensions as first-order analysis objects.

We discuss a number of perspectives to evaluate the outlyingness of dimensions and present a classification. We categorize the outlier dimensions in three categories, namely *characteristic*, *distribution based* and *structural* outliers. This categorization motivates the selection of different statistics and measures to use in our analysis. We use these to construct a derived data table for the analysis of the dimensions. Different statistics/features provide different insight on the dimensions. Moreover, we bring state-of-the-art measures and methods from outlier analysis of items in statistics into the analysis of dimensions. We introduce novel interactive visualizations to incorporate these measures into the analysis, such as the *z-score* and the *data depth* views. We then present a number of analysis strategies based on the different categories of outliers. These strategies serve as a guidance for analysts in handling outlier dimensions. All these building blocks enable the outlier-aware analysis of high dimensional data. Through a number of use cases, we showcase how these analysis are applied to high dimensional datasets.

The outlier analysis of the dimensions is an attempt to understand the structures within the set of dimensions. With the datasets getting larger in terms of variables and modalities, methods that help the analysts gain an understanding of the different aspects of these dimensions is becoming more important. In this respect, interactive methods, such as the one presented here, opens up new possibilities to make analysis routines that are more structured and reliable.

Acknowledgments

We would like to thank Astri Johansen Lundervold and Arvid Lundervold from University of Bergen for enabling access to the data related to the cognitive aging study supported by grants from the Western Norway Regional Health Authority (# 911397 and #911687 to AJL and 911593 to AL).

Paper D

Optimizing Processes in Visual Analytics to Meet the Three Human Time Constants

Cagatay Turkay[1] and Helwig Hauser[1]

[1]Department of Informatics, University of Bergen, Norway

Abstract

In their seminal work [25], Card et al. identified three human time constants and they discussed how the response times of information visualization should relate to them. These constants have great importance in that they represent the temporal human-computer interaction characteristics (at three different time scales) such that an optimal communication between the human and the computer can be achieved. In this paper, we show how interactive visualization processes can be realized in visual analytics such that they adhere to these human time constants. We define in particular how to meet these constraints when integrating computational tools in visual analysis that usually do not guarantee a real-time response. We describe how online machine learning algorithms enable the design of systems that are respectful to the perceptual characteristics of their users. This new way of modeling visual analysis processes is inline with a paradigm shift: instead of forcing the user to adjust to the temporal and cognitive capabilities of visual analysis solutions, we should orient the technical solutions at the communication characteristics of the users. We reason and demonstrate that such a process design can contribute to optimizing the efficiency and effectivity of interactive visual data analysis.

This article is in submission to: *Computers and Graphics*, 2013.

1 Introduction

The seminal work by Card et al. [25] investigates different aspects of human capabilities in our communication with the outer world, be it another human or a computer. In their paper, they describe the human computer interaction process as a dialogue between the user and the computer. Based on their investigation of psychology literature, they present three *human time constants* that characterize the temporal characteristics of our related human capabilities. These constants are reported to be highly important to achieve an optimal communication between the user and the computer. The first constant relates to the *perceptual processing* level at which humans are able to perceive changes in consecutive images as visually continuous animation. To achieve a visually smooth animation, the images need to be updated at least 10 times per second. The second constant addresses the *immediate response* level at which the parts in a communication are exchanging, forming a dialogue. The communication is interrupted if there is no response from the other party within about 1 second. The third time constant is the *unit task* constant which determines the limits for an elementary task to be completed during such a dialogue. This constant is reported to be more flexible and defined in an interval between 10 to 30 seconds.

Visual analytics (VA) is, in particular, an interactive and iterative dialogue between the human and the computer [84]. The interactive analysis process is a sequence of actions by the user and responses by the computer. The successful outcome of this process depends on the interpretation of these responses by the user. As such, it is of vital importance to think of visual analytics as a dialogue and to properly address the perceptual and cognitive capabilities of humans in this dialogue in the light of the already mentioned three time constants. By doing so, visual analysis sessions can be designed to be cognitively uninterrupted, which, in turn, can lead to optimized processes.

As a result of recent research activities, powerful processes in visual analysis in-

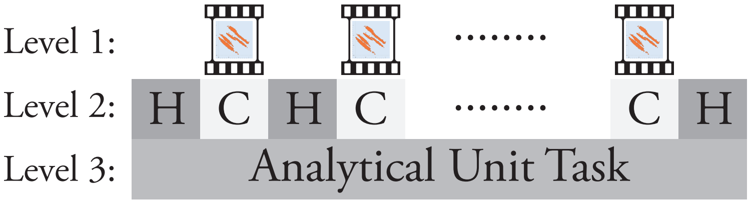


Figure 1: The unit task level is determined by an analytical task that answers a particular question about the data. The second level represents the dialogue between the human and the computer and moderates the response times of the computer. Responses can be given through an animation controlled by the visualization update level (running at 10 Hz or faster). (H: Human, C: Computer)

volve the tight integration of computational tools and interactive methods [112]. The careful design of the temporal characteristics of this integration is of great importance for optimizing the targeted analytical processes. However, the temporal aspects of such integrations have not been investigated in many studies. With this paper, we aim to fill this gap by describing a new approach to designing the temporal aspects of visual analytics processes in order to meet the three human time constants as described by Card et al. [25]. We describe *three levels of operation* for analytical processes, in particular for those that involve the integration of computational tools and interactive methods. An illustration of how these levels operate can be seen in Figure 1. Here, the third level manages the time involved in completing an analytical task, e.g., observing the relations between several variables in a dataset. The second level moderates the human-computer dialogue. It ensures that the dialogue occurs at a temporal pace where the human can give immediate responses, i.e., occurring within the limits of the second human time constant. The first level is responsible to make sure that the updates in the visualizations happen at a rate that is perceptually suitable for the human.

In essence, our solution moderates the temporal aspects of the interactive visual steering of computational analysis tools. Instead of forcing the user to wait for an interactive computation to finish, our methodology aims to present a best possible result within an acceptable time frame. And depending on the interpretation of these first approximate results, the user might either wait for more accurate results to compute or continue to explore the data by updating his/her interactive inputs. This approach is inline with the suggestion by Card et al. [25] that reads *"... When the cycle time becomes too high, cooperating rendering processes reduce the quality of rendering ... "*. Similarly, Jean-Daniel Fekete, in his Dagstuhl talk on the integration of computational tools with information visualization, commented that integrated methods should provide *any-time responses*, sometimes also allowing for some sacrifices on quality [56]. To the best of our knowledge, our work is now one of the first studies to directly address the temporal human comprehension capabilities by introducing immediate responses in the interactive use of integrated computational analysis tools.

In this paper, we present a methodology on how an interactive visual analysis system could be designed to conform to the human time constants and how such a system could be realized with the use of appropriate techniques. We firstly describe the three levels of operation that constitute the fundamental blocks to achieve optimized processes. We then introduce a number of novel computational and interaction techniques to achieve these three levels. We then demonstrate how these techniques enhance interactive analysis processes. The contributions of this paper can be listed as:

- The three levels of operation to consider the three human constants in visual analytics processes

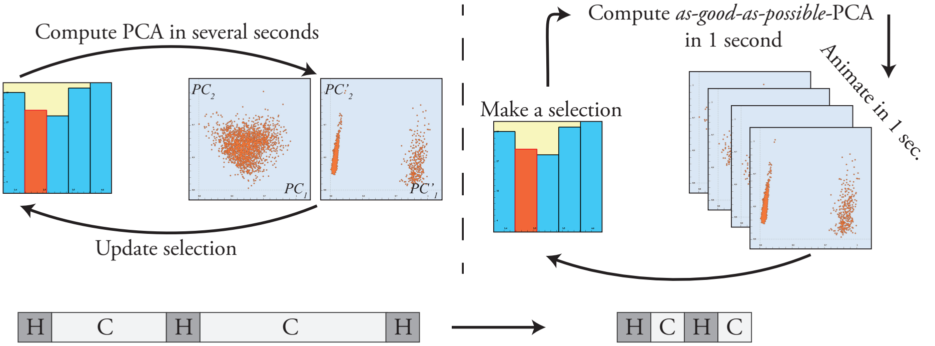


Figure 2: An example for optimizing an analytical process against the three human time constants [25]. In a conventional setup (left), the user first requests a (re-)computation of PCA results (with a selection of variables), then he/she waits a certain time for the results. This waiting could potentially interrupt the dialogue between the user and the computer (let's check my Inbox in the meantime!). In order to address this, our suggested optimization (right) computes PCA results *as good as possible* within 1 sec. in response to a selection by the user. And whenever new selections are made, the results are re-computed in no more than 1 sec. and the visualizations animate in 1 sec. to display the new results. The shown H–C–H–. . . - abstraction indicates the pattern of interaction (the lengths indicate the time spent).

- The *keyframed brushing* technique to improve the human-computer dialogue
- The utilization of *online algorithms* together with a suitable *sampling mechanism* to moderate the response time of computational tools
- The integrated use of perceptually optimized, animated transitions to communicate different computational results

After a discussion of related work, we give an example of how an analytical process can be optimized in Section 3, we continue with the description of the three levels of operation and present the keyframed brushing method in Section 5. We describe how online algorithms and animated transitions can be utilized within visual analysis applications in Sections 6 and 7. We then present sample analysis scenarios, evaluate and discuss the quality/limitations of online computations, and close with discussions on the methods and conclusions.

2 Related Work

The careful consideration of human factors is one of the central aspects of visual analytics [112]. Green et al. [72] discuss the central role of human computer interaction and suggest that the interaction has to be designed as a seamless cycle of give and take. *Levels of interaction* that are in line with the human time constants are also discussed in *Illuminating the Path* [187] by Thomas and Cook.

They emphasize the role of human time constants when designing interaction mechanisms for visual analytics. In our methods, we are motivated by the recommendations in these works in our effort to design optimized interactive visual analysis processes.

The integration of interactive methods with computational tools is becoming increasingly common in visual analysis [210, 144, 190]. However, there are not so many examples of studies on how such an integration should be carried out in the best possible setting for the user. With this work, we emphasize the temporal aspects that need to be considered for a successful design of such an integration.

Shneiderman states that interactive mechanisms need to give immediate feedback to dynamic queries within certain temporal limitations [173]. High performance techniques, such as predictive caching [27], a multi-threading architecture [146], or GPU-supported methods [55] have shown to improve the scalability of interactive visual analysis. Rosenbaum and Schumann [156] suggest a general progressive refinement framework to achieve a scalable system in terms of response times, visual clutter, and available resources. The authors also state that developing specific progressive refinement solutions for visualization systems is an open question. In a recent work, Ahmed and Weaver [3] discuss the details of a highly interactive cluster exploration system and one important aspect in their work is that the authors also display approximate clustering results to maintain smoothly running interactivity. In another recent work, Fisher et al. [61] present a database query system running on incremental samples. Their user study with analysts reveals that such an incremental approach enables analysts to give certain decisions early and update/remove their queries without waiting for the results to complete. This paper supports our motivation to suggest a methodology that incorporates incremental sampling and online computations. Our work adds to this part of the literature with the use of online algorithms to ensure time bounded computations. We make use of a sampling strategy together with the online computational tools. Although sampling has not been subject to many studies in visualization yet, it has been proven to be helpful in clutter reduction [48].

Although the usefulness of animations in visualization has also been disputed [155], there are several good examples where animations proved to be useful. Heer and Robertson [87] employ animated transitions between different statistical graphics. Animated transitions between different projections of high-dimensional datasets have also been used with success [49, 35, 16, 96]. We contribute to this part of the literature with a novel mechanism to generate animated transitions and to perform the animations seamlessly within the human-computer dialogue.

3 Example of Optimizing an Analytical Process

Here, we first go through an example of how a typical analytical process can be optimized against the three human time constants. We first describe the corresponding analysis problem and discuss how a typical visual analysis environment would provide a solution where human time constraints are not met. We then explain how a solution can be designed which respects the human time constants. In this example, and for the rest of the paper, we utilize the dual analysis approach [189] to analyze both the dimensions (visualized over a yellow background) and the data items (blue background) in parallel.

We analyze a high-dimensional dataset on the socio-economic values and crime statistics for the communities in the US in the year 1990 [12]. The dataset contains 128 variables, and we group these variables in five semantic categories, namely, *demographic*, *income*, *accommodation*, *family life*, and, *crime*. The analytical goal in this illustrative example is to observe if there are relations between the categories, and for the sake of simplicity we limit our interest to income- and crime-related data. One possible approach to perform this analysis is to summarize each category with a lower-dimensional representation, using a method such as principal component analysis (PCA) [100], and to compare projections of these representations. At this stage, we assume that the VA system in use offers PCA as a built-in computation [189]. Accordingly, the user selects a subset of the variables and the system computes the principal components for the selected subset of the data (i.e., using only the selected variables). A scatterplot visualizes the results using the first two principal components (Figure 2 - left).

In order to carry out this analysis "traditionally", the following steps could be followed. The user starts with selecting the subset of variables in the *income* category and triggers a PCA computation with this subset. Depending on the duration of the calculation, the user waits a certain time for the results to be computed (which could vary between milliseconds and minutes, or even more). According to Card et al. [25], however, in a dialogue, the user needs to get at least some feedback in less than one second, otherwise the dialogue (and the related cognitive process) can get interrupted. As a result, the user would have to re-orient himself/herself to continue the task, making a second iteration with a new input to the system, and so on. (Figure 2 - left).

An optimized version of this analysis, that respects the human time constants, can be carried out as follows. The user starts with selecting the same subset of variables in the *income* category by brushing on the histogram. In response, the system computes the PCA results *as good as it can* in no more than 1 sec. and displays the results. Likely, the results are then only approximate. However, if the user does not spot any interesting structure in this immediate result, he/she does not need to wait for the precise PCA computations to finish and immediately can make another interactive selection to update the visualizations. In our example, the user spots no interesting structure in the PCA results for the *income* category and immediately continues to selecting the *crime-related* category. Within no more than one more second, the PCA for this second selection is computed (as good as possible) and the points in the scatterplot animate to their new location. This animation is rendered at a minimum rate of 10 Hz where the total animation takes 1 second (Figure 2 - right). Only in the case that some interesting structure is observed in one of the categories, the user can then refer to a more precise PCA computation, that usually takes longer to compute, and then waits for the results, accordingly. We refer to the accompanying video for an immediate impression of this example.

Notice that after the optimization of this analysis process, all the operations are performed with temporal characteristics that are in line with the communicative capabilities of the user. This ensures that the dialogue between the user and the system through the session is not broken and we make sure to effectively use the perceptual and cognitive capabilities of the user in gaining insight.

4 Respecting the Human Time Constants

With the guidance of the human time constants, we aim to improve the dialogue between the human and the computer during visual analysis sessions. We achieve this by adjusting the system to operate at three levels (at three time scales of interaction). These levels correspond to the three human time constants and thus, are associated with certain temporal limits as shown in Table 1.

In this paper, we limit our discussion to a subset of visual analytics methods, including i) linking & brushing, ii) the integration of computational tools and interactive methods, iii) a visual representation of the computational analysis results. We argue, however, that our model is more general and that also other processes in visual analytics fit well into it. Briefly, we consider a *unit task in visual analytics as a sequence of actions and reactions where the reactions can be given by animated visualizations* and the three levels of operation moderate how such a task can be carried out at an optimized fashion by respecting the human time constants (Figure 1).

Level 3: Unit task completion – This level determines the temporal range in which an analytical unit task is completed. Such an analytical task is performed to answer a specific question related to the data. Such a task involves a sequence of inputs from the user and corresponding responses from the computer. Examples of such a unit task include: i) changing a selection of items in one view (in order to explore inter-dimensional relations) and observing how the according focus+context visualization in a linked view is changing, ii) observing the grouping structure of data items when different subsets of the variables are used, for instance, in clustering. The time constraints for this level of operation are more flexible according to Card et al. [25] and depend on the analytical unit task. In order to ease the construction of different patterns of selections, we developed an interaction mechanism called *keyframed brushing* (see Section 5). With this mechanism, it is possible to frame unit tasks with fixed completion times, i.e., 10 sec., 20 sec., or 30 seconds. Such a unit task is then a sequence of actions and reactions between the human and the computer as discussed in the following.

Level 2: Human-computer dialogue – This level is mainly responsible to maintain the dialogue nature of the visual analysis process. It ensures that the communication between the user and the computer is not interrupted. Specifically, this level focuses on maintaining a guaranteed response time (1 sec.) when integrated computational tools are utilized. This mechanism realizes an uninterrupted dialogue by making sure that the immediate response capability of the user is exploited.

Maintaining the 1 sec. response time is not straightforward when the computations are complex and the data is large. Our solution to approach this problem is to compromise the quality of the results by computing "only" the best possible result within the limited time frame. Similarly, in computer graphics, reducing the quality of the rendering process to maintain an interactive frame rate is common practice and related methods are usually referred to as *progressive refinement.* [34]. In order to achieve this, we make use of online algorithms together with a suitable sampling strategy (more in Section 6).

Level 1: Visualization update – Animated transitions have been proven to be helpful when the motion of the data items are of importance [87] and can help to avoid *change blindness* [152]. Therefore, we make use of animated transitions between the different computational results that are generated as a result of the dialogue occurring at the second level of operation. The visualization update level moderates the update rate of animated visualizations and secures the perceptual processing of the animations in the visualization. In order to create animations that are smooth in the eye, the lower bound for the update rate should be 10

Level	Operation Level	Human time constant	Response time (sec.)
Level 1	Visualization update	Perceptual processing	0.1
Level 2	Human-computer dialogue	Immediate response	1
Level 3	Analytical task completion	Unit task	10 - 30

Table 1: The three levels of operation, the corresponding human time constants [25], and the associated time limitations

Hz [25]. According to a study on the effects of update rate on the sense of presence in virtual environments [13], 15 Hz is an optimum rate for updating animations. In this work, we include the different update rates as reported in literature, and animate the visualizations at either 10, 15, or 20 Hz. For a more detailed discussion on why animated transitions are suggested, see Section 7.1.

5 Keyframed Brushing

The *keyframed brushing* mechanism is intended to reshape (a certain subset of) analytical tasks as a dialogue while keeping the user engaged. The user defines two or more brushes (according to his/her analytical goal), similar to defining key frames in computer-assisted animation [26]. Using these *key brushes*, a sequence of *in-between* brushes is generated automatically. After the brush sequence is computed, the system starts traversing through this sequence without the need for further input by the user. Depending on the user's preference, the complete sequence is traversed in 10 sec., 20 sec., or 30 sec., and moving from one brush to the next takes 1 second. Here, traversing the whole sequence is considered as a task operating at Level 3 and moving from one brush to the next as operating at Level 2 as defined above. Keyframed brushing enables the user to focus on the linked views that display the results of the animation rather than paying attention to moving the brush in a particular fashion. Refer to Section 8.1 for a demonstration of cases where keyframed brushing proves to be helpful in cases that are hard to investigate with manually modified brushes. This mechanism has a utilization both as an automated linking & brushing operation and as a method to interact with computational tools.

In order to construct brushing-based animations, we enable the specification of key brushes through conventional visualizations, such as scatter-plots and histograms. In Figure 3, the interface to define a brush sequence can be seen. We draw overlays (in pinkish color) to abstract the range of the final brush sequence. We suggest four different modes to create brush sequences when the user is finished with the key brush definition: *moving* (m_{mov}), *extending* (m_{ext}), *no in-betweening* (m_{dir}), and *constrained* (m_{con}) (Figure 4). We refer to the key brushes as $B = \langle b_1, \ldots, b_t \rangle$ where b_1 is the first and b_t is the last brush made by the user. In the m_{mov}, m_{ext}, and m_{con} modes, the user defines only the start and the end brush, i.e., $t = 2$, while in the m_{dir} mode the user defines as many brushes as he/she wants. In the m_{mov} mode, the user creates a brush sequence where the brush moves from b_1 to b_2 through a linear interpolation. In the m_{ext} mode the borders of the brush b_1 get extended at each iteration so that the final brush covers both b_1 and b_2. In the m_{dir} mode, we create no in-betweens and the selection moves directly from b_i to b_{i+1} at each iteration, where each iteration is performed in 1 second. Notice that the total time for such a m_{dir} sequence depends on the number of brushes made. The fourth mode (m_{con}) is the *con-*

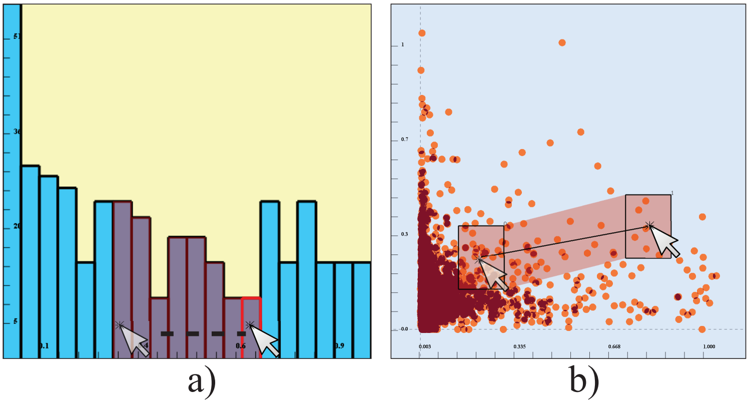


Figure 3: Keyframed brushing can be performed in a histogram (a) or in a scatterplot (b). The user interactively determines the start and end selections (“key brushes”), which are then accompanied with computed “in-between” brushes.

strained brushing mode, where the path of the brush sequence snaps to a fixed, predetermined line. We enable three fixed paths to make selections in scatterplots, namely, parallel to the x-axis, parallel to the y-axis, and to the diagonal of the scatterplot. In order to activate this mode, the user presses the shift key on the keyboard while moving the mouse to make the second brush b_2. The path then automatically snaps to one of the closest fixed paths. For instance consider Figure 4-d, that displays the *% of African Americans* vs. *median income*. When the user constructs a path parallel to y-axis, the resulting brush sequence precisely selects higher and higher income levels for a fixed *% of African Americans* values. This mode thereby helps the user to move a brush in a precise manner over paths that carry specific properties and leads to brush sequences that are less arbitrary as compared to those that are constructed manually.

The total duration (10, 20, or 30 seconds) of the brush sequence (m_{mov}, m_{ext}, and m_{con}) determines how much the selection changes between two (keyframed) brushes. For instance, when 30 secs. are used, then the difference in the positions of two consecutive brushes is low and the overlap of the selections is therefore higher, leading to a more coherent visualization (and its according transition). We also provide the flexibility to modify the time that it takes in moving from one brush to the other, which becomes useful when the speed of the selection is in focus (see Section 8.1).

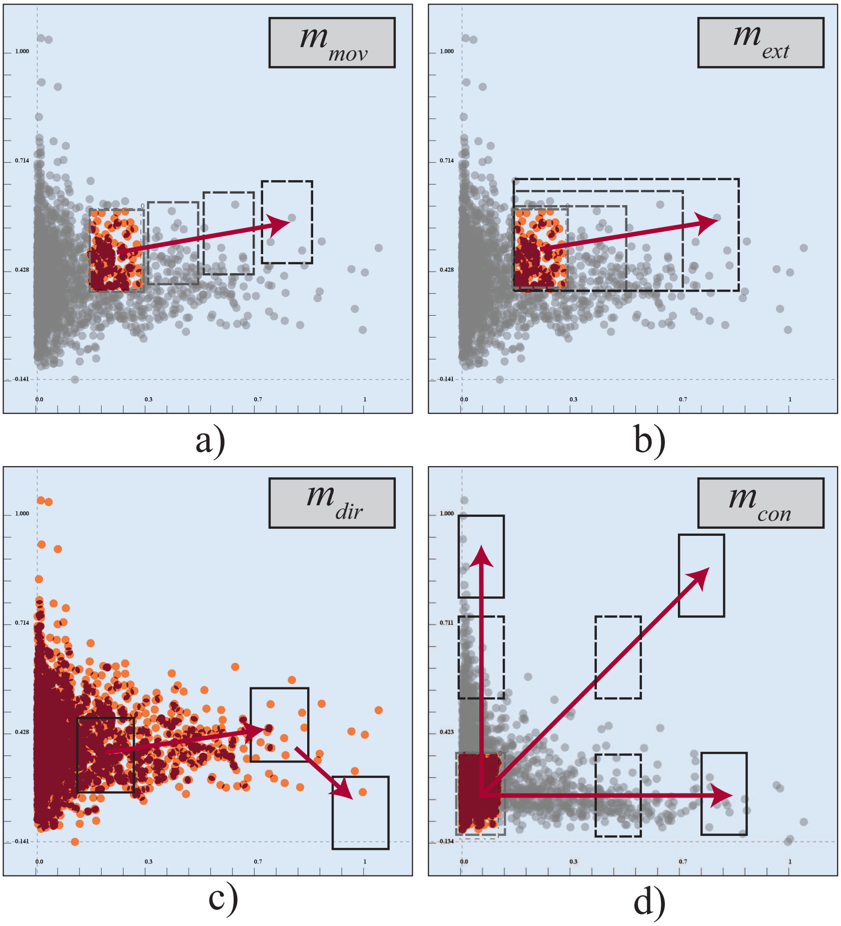


Figure 4: Four modes for keyframed brushing (according to the user interaction as illustrated in Figure 3-b). a) Moving brush mode: the position of the in-between brushes are linearly interpolated, b) Extending brush mode: the brush extends at every step, c) No in-betweening: the final sequence consists of only the three brushes, d) Constrained brushing mode: the path of the selection automatically snaps to one of the fixed lines (parallel to x-axis, y-axis, or to the diagonal)

6 Online Computations

In order to maintain the temporal limitations set forth by the human time constants, we develop a mechanism where the computational tool guarantees to respond within a fixed time, i.e., 1 second. To achieve this, we make use of *online algorithms*, which are capable of processing the data piece-by-piece se-

Algorithm 1 Online computation with random sampling

```
procedure ComputeInFixedTime
    $O :\ Online\ computation\ module$
    $D :\ Data,\ size :\ n \times p$
    $Q :\ Random\ sampling\ queue,\ size : n$
    $t_{lim} :\ human - time\ constant$                ▷ Fixed to 1 sec.
    $t_0 :\ currentTime()$
    $timeLeft :\ t_{lim}$
    while $Q.notempty()$ do                          ▷ Until all samples are used
        while $timeLeft > 0$ do
            $i \leftarrow Q.pop()$
            $x \leftarrow D[i]$
            $O.update(i)$
            $timeLeft :\ t_{lim} - (currentTime() - t_0)$
        end while
        $O.returnResults()$                          ▷ Visualization is updated
    end while
end procedure
```

quentially [4]. These algorithms do not need the whole data to operate and can update the results as new data becomes available. In the machine learning literature, there are online versions of computational tools that are frequently used in visual analytics, such as principal component analysis [79] and clustering [75]. One common method to use online algorithms is to pass the data items row by row, so that each update cycle of the algorithm is performed in a limited time [4]. To be able to utilize this incremental computing nature of online algorithms, we use them in combination with a sampling method.

In this paper, we use this row-by-row computation strategy to achieve guaranteed response times. We determine which rows to use by a random sampling of the data. The main reason to prefer a random sampling method is the positive effect of randomization on on-line computational methods [4]. Certainly, it is possible to alternatively also use more sophisticated sampling methods such as stratified sampling or selective sampling [143]. For a detailed discussion on the scope and limitations of online algorithms, see Section 9.

In short, our approach makes sure that the employed computational tool responds in a fixed time by limiting the size of the portion of the data that it processes. This portion is made as large as possible so that it can be handled within the given time limitations. The details of this approach are given as Algorithm 1. This approach assures that the computations are finished and the associated visualization is updated within the temporal limits. However, due to the fact that the results are computed on a limited sample, the results are usually

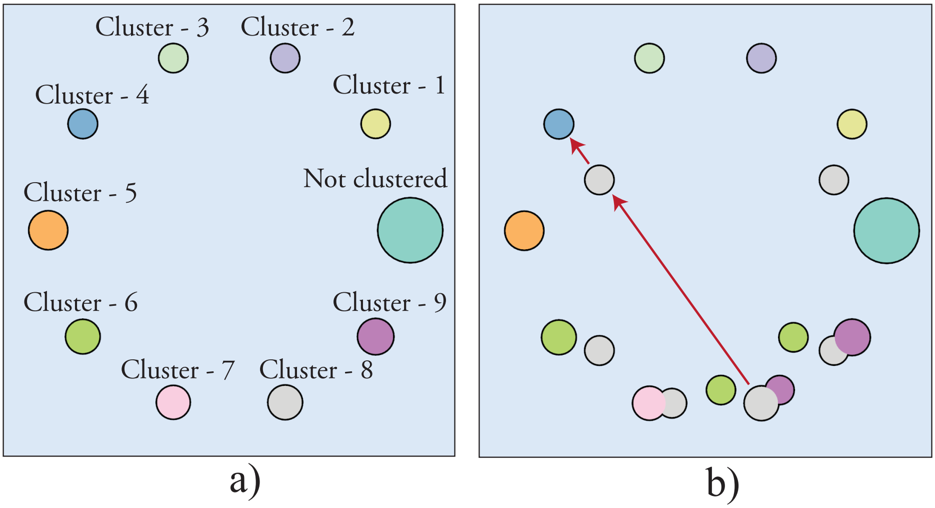


Figure 5: a) An abstract visualization to show clustering results where each cluster is represented by a circle with a distinct color. The visualization shows a clustering result with 9 clusters (+1 for the non-clustered items) where the size of the points denote the number of members of the cluster. b) When a new clustering result is present, the changes in the clusters of items are represented by animated points between the clusters. Here, amongst other transitions, some items of cluster-8 (grayish) are joining cluster-4 (blue)

not as accurate as one would achieve if the whole dataset was used. Therefore the algorithm continues to run (in a separate background thread) after the first response is given and consumes more and more of the data every second. This implies that the results are getting more and more accurate every second as the user observes the result without being disengaged from the communication. In the context of this work, we incorporated the online versions of two popular computational tools, PCA and clustering (a version with similar principles as the k-means algorithm). In algorithm 1, these tools correspond to the module O.

Incremental PCA – Online PCA algorithms make use of an incremental updating of the singular value decomposition (SVD) of the data matrix [79]. In this paper, we refer to online PCA and incremental PCA interchangeably. Here, we use the SVD updating methodology described by Hall et al. [79]. Notice that the resulting PCs are computed based only on the sampled points, however, the final projections are applied to the whole dataset. The results are then visualized through scatterplots where the axes are the first two principal components (see Figure 2).

Online clustering – Similarly for clustering, we use an online clustering algorithm, defined by Guedalia et al. [75]. This algorithm takes a k parameter as an upper bound on the number of clusters. At each iteration it includes the new item as a new cluster in the data and appropriately merges/splits the new clusters. Notice that in clustering, at each iteration of Algorithm 1, only a subset of the items is clustered. Although such an approach does not provide the labeling for all the items, it provides an overview of the clustering structure, i.e., the number and relative sizes of clusters. Therefore, we visualize the results of such a clustering using an abstract visualization, where each cluster is represented by a circle with a distinct color (taken from ColorBrewer [81]) where the size represents the number of items it includes (Figure 5-a). This view is also animated and shows how many items are migrating between different clustering results (Figure 5-b). Here, we prefer an abstract visualization due to the lack of an inherent spatial mapping of clusters to 2D.

Both of these methods are used in integration with the conventional linking & brushing and the keyframed brushing mechanism. We are running separate computational threads to carry out these two operations. Both of these threads manage the time limitations as outlined in Algorithm 1. When the results of the computations are ready, a signal is sent to the visualizations to update the results in the corresponding visualizations. In Section 9, we evaluate the performance of online computations in terms of the stability of the results and the amount of data that could be processed within the time limitations.

7 Animated Transitions

In our approach, we use animated transitions to support the interpretations of changes while comparing different results of a computational tool. Each computational result can be thought of as a key frame in an animation and the in-betweened frames are computed by the animation module. Animated transitions are controlled by the first level of operation and are done at 10 Hz or faster. A single animation sequence takes 1 sec. For the sake of simplicity, we focus in the following on animations that display PCA results. Assume that we start with a view V that shows the PCA projection of the data based on all the dimensions in the US census dataset.

Immediate response animations – Our online computation mechanism immediately responds to user input such as a new selection of a group of dimensions (similar to Figure 2). The interactive input triggers Algorithm 1 which returns the first, approximate result within 1 second. If the user spots an interesting structure in this first result, he/she can wait for the algorithm to compute a more precise result. And in order to maintain the human-computer dialogue also in this case, the visualization V is fed with new, more accurate computation results every second. As a result, the points in V start animating to their new positions in the newly available PCA projection. However, if there is no apparently

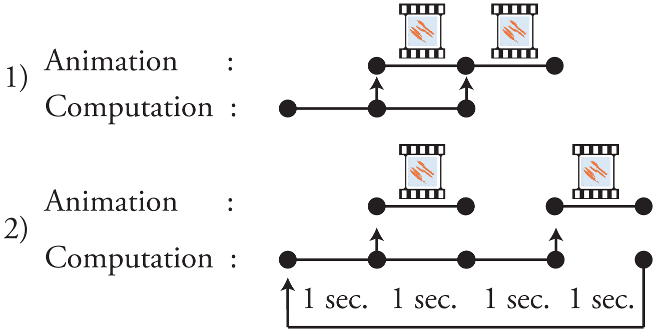


Figure 6: Two different animation patterns. The first pattern moderates "immediate response animations", where every 1 sec. a new computational result is computed and a new animation sequence is played one after the other (1). In the "keyframed brushing animations", each animation sequence is followed by a 1 sec. of a pause to give the user enough time to familiarize with the computation results (2) – for such animations looping is possible, as well.

interesting structure in the first results, or at any instance, the user can update the selection. In this case, the current animation is stopped immediately and the view animates to the new computation results, instead. The animation ends when all the items are processed and Algorithm 1 terminates. An illustration of the pattern of such an animation is shown in Figure 6-1. Instead of alternative methods such as rotating the axes of the projections [49], we prefer (non)linear interpolations for the animations since the transitions carry an incremental nature in immediate response animations.

In order to visually encode the accuracy of the computation, we adjust the size of the points in the scatterplots (Figure 7). The size of the points are inversely proportional to the proportion of the samples that are already processed. This means that when a little number of samples are used, the points are larger. This creates visualizations with more overlapping points, which makes it possible to see only the overall structures and no details. As the computation becomes more reliable at each iteration, the points get smaller, enabling more detailed readings from the visualization as it sharpens. Alternative methods, such as blurring in the *Semantic Depth of Field* technique [117], can also be incorporated here to encode the uncertainty in the results.

Keyframed brushing animations – These animations are triggered when the user performs a keyframed brush operation. A typical use of this animation is as follows: firstly, the user makes a keyframed brush sequence that selects different subsets of dimensions to then observe the differences between the PCA computations that are done for each of these selections in the sequence. Referring back to the example in Section 3, a keyframed brush could span each of the 5

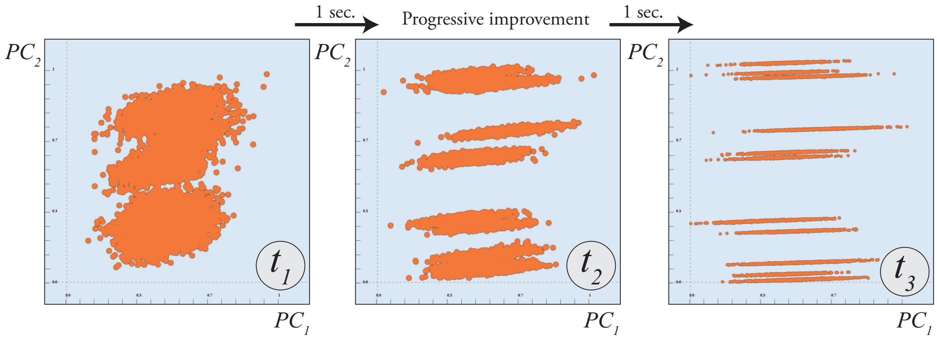


Figure 7: Our interactive PCA computation module responds to any selection by computing a best possible result within the given temporal constraints. If the user spots an interesting structure, he/she can observe a progressively improving PCA animation as new results are computed by Algorithm 1. If no interesting structure is observed, the user can immediately make a new selection and stop the current animation and trigger a new one. Since there is an interesting structure in the first frame above, we decide to observe the whole animation as the results become more accurate. Here, the point sizes are proportional to the uncertainty related to the computations. This mechanism ensures that the user makes only overall readings when the results are less accurate and detailed readings (through smaller points) as the quality of the results improve. In the animation sequence (3 key frames are shown), only three bigger groups are visible in the beginning. As more and more accurate results arrive, sub-groups in each of the larger groups can be observed.

category bins in the histogram. As a result of this input, the PCA visualization would animate within these five projection results, computed on the run by the online algorithm.

In these animations, the PCA is computed using the first keyframe of the brush sequence and V is animated accordingly. Then the system waits for one second to give the user enough time to observe the results. This, in particular, is the user part of the human-computer dialogue. At the end of this time, the system starts animating again to the next selection in the sequence until all the brushes are processed. For this type of animation, we also include a looping function, so that the changes in the sequence can be observed more than once. This pattern is illustrated in Figure 6-2. We achieve the looping, pausing, and rewinding operations by using a fixed sampling array (array Q in Algorithm 1). For each loop, the results are recomputed instead of being stored to avoid the memory overhead that it can cause.

7.1 Why Animated Transitions?

Due to the conflicting reports on the successful utilization of animations [193, 155], we carefully consider our design choice related to the use of animated transitions. We also make sure that our animated transitions follow the *congruence principle*, that requires a natural mapping between the changes in the visualization and the information to be conveyed, and the *apprehension principle* which requires that the changes and relations depicted in the animations are easily perceivable and understandable by the user [193].

In our visualization approach, PCA projection results carry spatial characteristics, i.e., have a meaningful mapping to the x and y coordinates and all the changes between different projections, which carry valuable information for the analysis, happen within this spatial mapping. It is reported that in tasks where the objects' spatial positions are of importance, the utilization of animations is suggested [15]. Therefore we reckon that it is suitable to utilize animation for the visualization of change in such views. In the case of clustering related animations, the focus is on the membership changes and the clusters are mapped to physical positions in the abstract view. Therefore, the membership changes also carry spatial properties, e.g., from cluster A (located at x_1, y_1 in the view) to cluster B (x_2, y_2). This makes such changes also suitable to be visualized by animated transitions. This presence of such meaningful spatial mappings in the animated views maintains the *congruence* principle, accordingly. Further, it is stated that interacting with animated views, for instance via pausing or looping, is an important extensions to achieve the *apprehension* principle [193]. In our system, we enable this functionality such that the user can pause, stop, and loop animations interactively. Inline with the apprehension principle, one might argue that animations put additional attentional and memory load to the user. This may suggest the use of static visualizations such as small multiples, although there are conflicting reports on their efficiency [155, 73]. In our approach, we utilized animations as our main medium largely due to the fact that the user continuously interacts with the views to trigger new computations and generating new animations. Using a set of small multiples in response to all the selections done by the user is not space efficient in a multiple view environment. However, to support the user with this attentional and memory load, and to get together the best of two worlds, we include a mechanism that enables the user save any step of the current visualization animation as a separate interactive view that is then available for an individual analysis. In addition, we made improvements to the animated transitions to improve the readability of the animations. These improvements are discussed in detail in the following section.

7.2 Improving Animated Transitions

In the following, we present selected improvements to animated transitions. The first improvement is related to maintaining the coherence between two key frames (two computational results) of an animation. Such an improvement is important in order to preserve the mental map of the user [11] and similar challenges have been studied in other domains, such as in graph drawing [63]. In the case of PCA, the resulting principal components (PCs) are known to have arbitrary rotations and signs due to the nature of PCA [100, 22] . Due to this fact, we observed that although the structure of the point distribution does not change, i.e., item neighborhoods stay the same, the PCs can come out flipped and/or mirrored. This makes it very hard to follow the animations and creates rotations that carry no significant meaning. We solve this by checking the correlations ρ (using Pearson's correlation measure) of the axes between the first, x_1, y_1, and the second PCs x_2, y_2. If $\rho(x_2, y_1) > \rho(x_2, x_1)$, we flip the axes, and if $\rho(x_1, x_2) < 0$ (negatively correlated) we mirror the axis (mirroring check is done also for y). Refer to the accompanying video on how this affects the results.

Similarly, in the case of cluster computations, the resulting cluster labels are in principle arbitrary. In order to make coherent transitions between key frames (two clustering results), we need to find a mapping between two consecutive labellings. We use a metric called Jaccard coefficient [181], which measures the

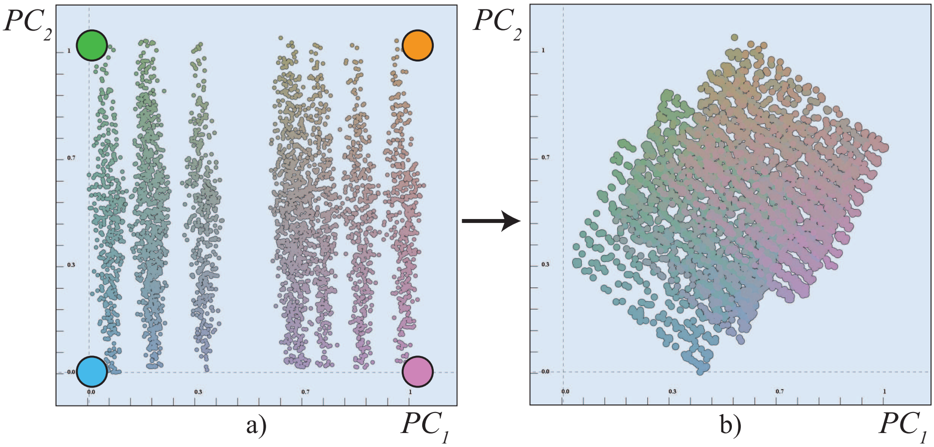


Figure 8: Coloring is used to enhance the perception of change between animation frames. We use a 2D color map taken from the CIELUV color space using a fixed lightness value (corners of the 2D map are shown). The points are colored according to their position at the beginning of the animation sequence (a) and this color stays the same for the whole animation (b).

overlap between two sets. For each cluster c_i in the current result, we compute the Jaccard values with all the clusters in the next frame c'. And for each cluster, we find the corresponding cluster c'_j with the highest $Jaccard(c_i, c'_j)$ and update the mapping accordingly.

Coloring is used additionally to support the tracking of changes in the animations in scatterplots. We map the color of each point based on their x, y coordinates in the beginning of an animation sequence (Figure 8-a) and the coloring stays constant for all points through the animation. The corners of the 2D color map and the resulting colors can be seen in Figure 8. The color map is constructed using an isoluminant slice of the CIELUV color space [207]. With this coloring mechanism, we enable a mixture of dynamic and static techniques in the visualization of change.

We also improved the animations by using a non-linear interpolation while constructing the in-betweens. We create animations that are slow both at the start and towards the end of the animation (also known as "ease-in and ease-out"). This update reserves more time for the user to observe the configurations of the plots at the key frames rather than at the in-betweens. The effect of such transitions has been also exploited by van Wijk and Nuij [197] and the study by Dragicevic et al. [46] states that the use of such non-linear methods improve the efficiency of animations.

8 Sample Analysis Scenarios

The new interaction mechanisms and the animated transitions enable a new set of analysis routines that lead to optimized analytical processes. Here, we exemplify three scenarios where our methods are used to investigate high-dimensional datasets having typical analytical tasks in mind. For the animations related to these tasks in this section, we refer to the accompanying video.

8.1 Observe the regression relations

In this example, we analyze the US Census dataset (see Section 3). We try to understand the relations between several dimensions using a multiple windows setting (Figure 9). Here, we make use of the speed of keyframed brush animations to investigate the regression relations and therefore reduce the time it takes to move from one brush to the next (from 1 to 0.2 second). We first bring up three linked views (the top three in Figure 9), where the first one shows the *% collage graduates* (*% coll*) vs. *income of African Americans* and the other two shows the *% of African Americans* against two values, *% not high school graduates* (*% not_HS*) and *% males that are divorced* (*% m_Div*). We perform a keyframed brush operation using the constrained mode, that precisely selects increasing *% coll* values. Then we observe the animated brushes to loop several times in the two linked views. We see that the selection in the left-top view is accelerating and moving in a higher speed with decreasing *% not_HS* values. In the top-right view, on the other hand, the selection moves at a constant and a slower speed with decreasing *% m_Div* values. These observations indicate that the relation between *% coll* and *% not_HS* is a negative non-linear (due to the acceleration) and stronger (due to the speed) regression. However, the relation between *% coll* and *% m_Div* seems to be a linear (constant speed) and a weaker regression. Refer to the accompanying video for an immediate impression of these temporal and visual effects.

In order to confirm this finding, we visualize the *% not_HS* and *% m_Div* values against *% coll* values. We indeed see that the scatterplots (the two below in Figure 9) confirm our observation. Such an analysis would not be easily possible by modifying the brushes manually since the attention of the user would be on manipulating the brush (instead of realizing the change of speeds in the linked focus+context visualization). One additional benefit of the constrained brushing mode in this example is that the selection sequence selects increasing values precisely, leaving the other dimension unchanged.

8.2 Determine dimensions with structure

Here, we analyze a dataset on protein homologies that was made available in the 2004 KDD Contest [1]. The dataset consist of 10498 rows and 77 columns, i.e., $n \times p = 10498 \times 77$. The analytical unit task here is to investigate the set of dimensions to determine those with an apparent structure in them and to use them for further analysis. Here, we make use of PCA calculations on different subsets of the dimensions and observe the changes to spot interesting structures. We use a visualization of the dimensions over two statistics, skewness and kurtosis (Figure 10-left). We observe that most of the dimensions have similar kurtosis values but varying skewness. To investigate the dimensions over their skewness values, we build a keyframed animation that starts with dimensions that are left-skewed animating to those that are right-skewed.

When we observe the resulting animation with the key frames shown in Figure 10-right, we see that there are structures appearing between frame 7 and 8. After further inspection through different statistics and the dual analysis framework (introduced in Section 3) [189], we find out that there is a single dimension that is included as a category field in the data and it causes these structures.

One important observation in this study is that even when the dataset size is relatively large (close to a million of values), the PCA calculations are done in the temporal limits and the animations run smoothly.

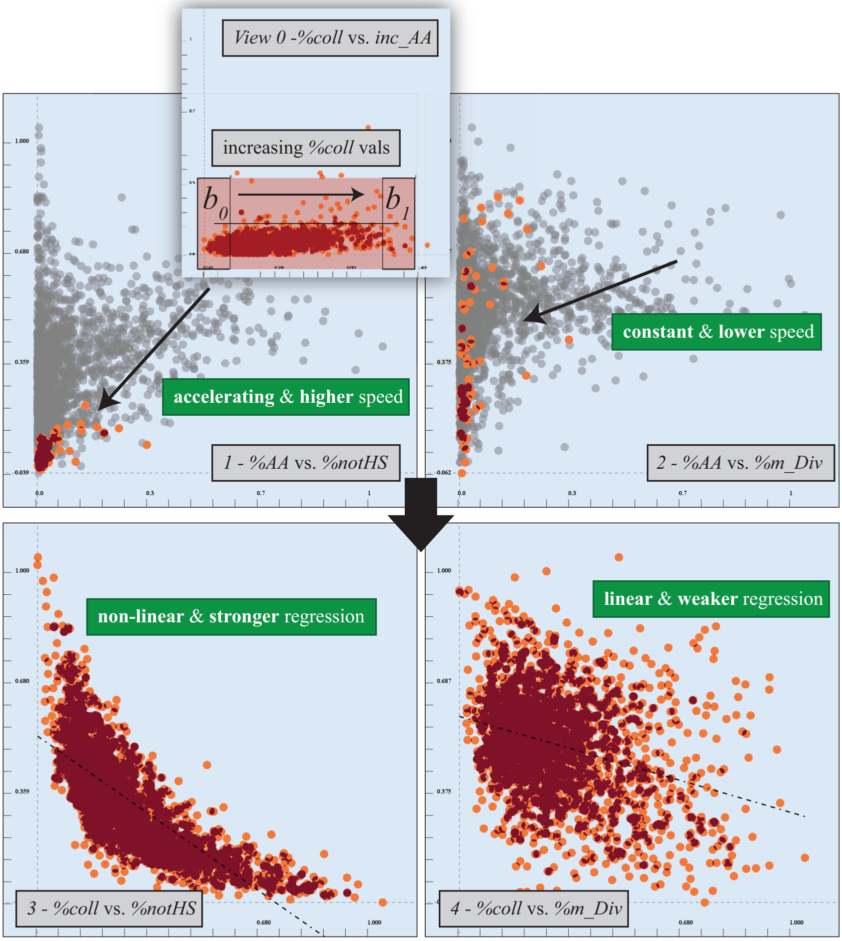


Figure 9: Using a constrained keyframed brush animation in a multiple views setting. A brush sequence is set up to follow increasing *% coll* values (view-0). When the brush sequence animation is looped several times, a difference in the speed of the selection is spotted between the two scatterplots. The selection accelerates at a high speed in view-1 indicating a non-linear, strong regression. And it moves at a constant, lower speed in view-2 indicating a linear and weaker regression. This finding is confirmed through views 3 & 4.

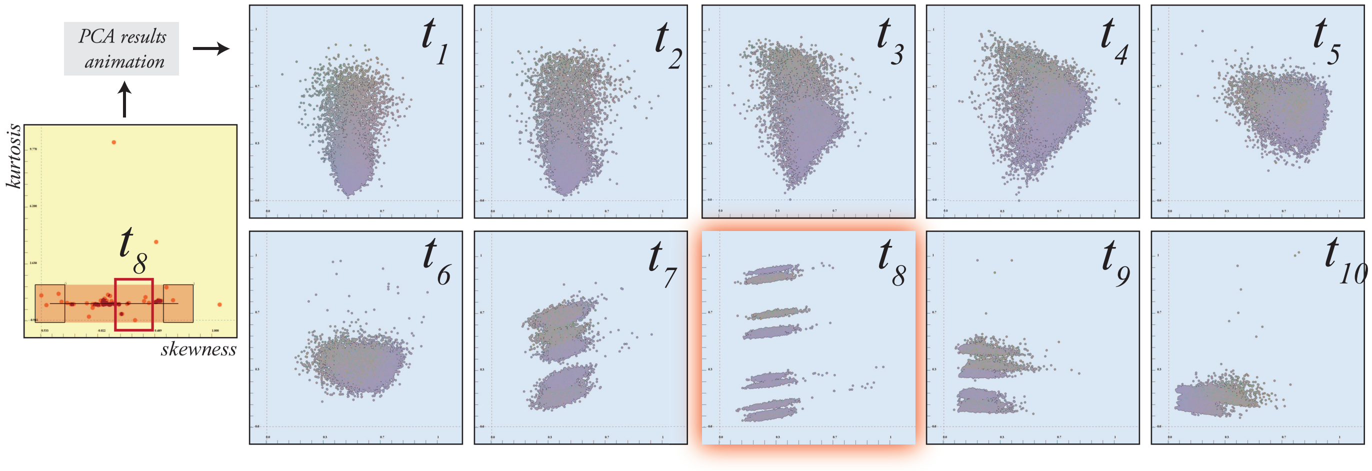


Figure 10: An analysis of the dimensions of the protein homology dataset. The aim is to find dimensions that carry structured information. A keyframed brush is generated through a *skewness* vs. *kurtosis* plot (the dimensions have similar kurtosis but varying skewness). The resulting brush sequence traverses from the left-skewed dimensions to the right-skewed ones. When the resulting animation of PCA results is observed (only the key frames here), a strong structure is spotted at key frame t_8. We pause the animation there and inspect the selected dimensions closely. As a result, the structures are found to be due to a dimension that contains categorical information.

8.3 Observe structures in the data

In this analysis example, we analyze data from The Cancer Genome Atlas (TCGA) project [183] related to a study on sub-types of breast cancer [137]. This dataset contains the gene expression levels of 1500 genes for 529 samples from tumor tissues. The comprehensive study of this dataset reports 4 breast cancer subtypes, namely, *Basal-like*, *HER2-enriched*, *Luminal A*, and *Luminal B* [137] and these labels are provided as meta-data. We load this dataset such that the genes are the rows and the tissue samples are the columns, i.e., $n = 1500$, $p = 529$.

The analysis question here is to find genes that are common or distinctive within the subtypes. In order to achieve this, we select, one by one, the four subtypes (i.e., 4 groups of the dimensions of our 2D table). In response to each of these selections, PCA is applied automatically. We observe the animated transitions between these four projections and check for structures within these transitions.

Figure 11 displays the PCA computation results (four *key frames*) for these four groups. When the transitions between the first three projections are observed, a group of 82 genes is found to be moving together and stable within the projections (marked with a black ellipse). This implies that this group of genes have similar expression patterns for all the three subtypes that are *Basal-like*, *HER2-enriched*, and *Luminal A*. Although this group of genes have little discriminative information, it amounts to the common characteristics that are shared within the subtypes.

To confirm this hypothesis, we refer to the gene expression values for these 82 genes. Similarly to doing multiple two- sample t-tests for these groups, we compare the center and the spread of the expression values for the 82 genes over these three groups. We have seen that there are in fact no significant differences in the expression levels of these genes within these three groups. This deeper statistical analysis supports our hypothesis on this group of genes. Now, if the analyst needs to run further classification algorithms on the tissue samples, this group of genes can safely be left out from the computations to achieve more precise classifications.

In addition to the above result, we observe that another group of genes (red-dashed ellipse in Figure 11) is positioned close to the first gene group in the projection for the second subtype *HER2-enriched*. However, the animated transitions reveal that this structure is only visible for the *HER2-enriched* subtype and not shared with the other subtypes (observing how this second group behaves is unfortunately not possible to show in static images). Similarly when the transitions for *Luminal A* (marked 4) subtype is viewed, it is seen that none of these structures are preserved.

In this analysis, we have seen that the animated transitions help the observer to determine structures that are preserved (or not) within the computation results. Our online computation scheme quickly leads to a hypothesis on structures within the genes, which is then confirmed through more precise but time-consuming computations. This demonstrates a typical optimized analysis pipeline, where the capabilities of the human is used optimally to detect structures which then provide the basis to refer to advanced computations.

9 Evaluation of Online Computations

Here we evaluate the quality of the results that are produced by the incremental PCA module and observe how quickly the computations converge to stable results in comparison to an offline algorithm. Our evaluation strategy is the following. We first compute PCA using a conventional offline approach and project the data items to the first two principal components and denote this as ρ. Secondly, we

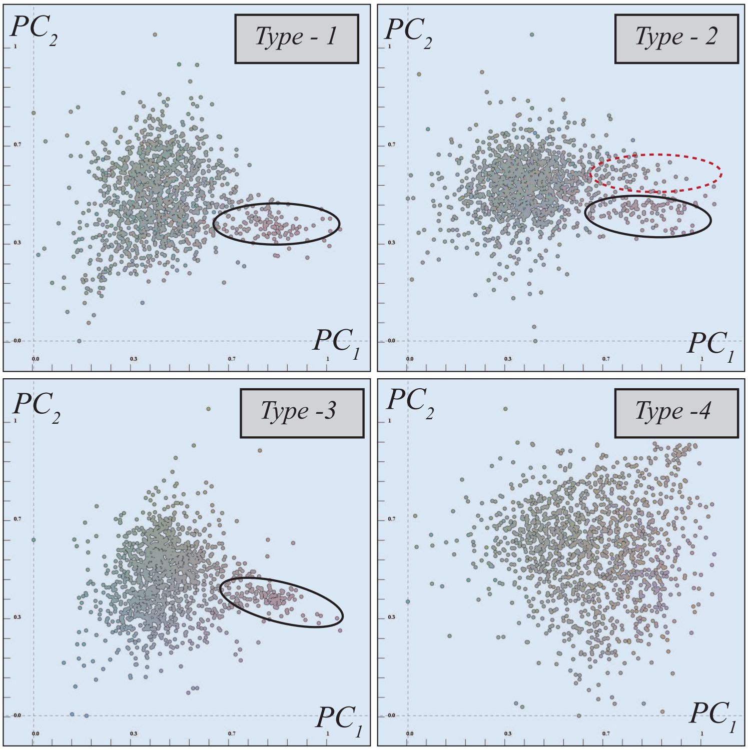


Figure 11: The analysis of data from the TCGA project on breast cancer subtypes [137]. Animated transitions (only the keyframes are shown here) reveal a subgroup of genes that are stable and similarly expressed within three subtypes of cancer (black circle). A group (red dashed circle) is found to be only formed in Type-2.

compute incremental PCA results in 10 iterations by growing the sampling size by 10% at each step. At the end of each such iteration, we project all the data items to the first two principal components ρ' and compare ρ' to the initially computed ρ. The comparison is done by a similarity metric called *neighborhood preservation ratio* (*NPR*) suggested by van der Maaten and Hinton [194]. This metric is computed by finding the set of k nearest neighbors of each point x_i in both of the projections ρ and ρ', which are denoted as G_i and $G_i^{'}$ respectively. We then compute the *NPR* between ρ and ρ' as: $1/n \cdot \sum_{i=1}^{n}(\left\|G_i \cap G_i^{'}\right\| /k)$. Here, if the two projections are completely the same, the *NPR* score is 1 and gets closer to zero as two projections differ from each other.

We run 5 comparative tests (using $k = 10$) with 5 different datasets which are either artificial or taken from the UCI repository [12]. The datasets are:

1. Artificial dataset $n = 4050$, $p = 35$ where the dimensions have distinct characteristics, e.g., normal, log-normal, uniform.
2. *US Census Dataset* (see Section 3) with $n = 2216$, $p = 86$.
3. An artificial dataset with $n = 1024$, $p = 256$ where the dimensions all together encode 16 clusters.
4. *Low Resolution Spectrometer* dataset with $n = 532$, $p = 97$.
5. *Protein Homology Dataset* (see Section 8) with $n = 10498$, $p = 77$.

Figure 12 displays the *NPR* scores for each of the dataset for 10% sample size increments. We observe that with the datasets that are taken from the UCI repository, even with 10% of the data, we obtained *NPR* scores close to 1, meaning that there is little difference with the projections computed by using only 10% of the data and those computed by an offline algorithm. However, for artificial datasets with structures, i.e., such as the 16 clusters in Test-3, or with dimensions that have very skewed distributions, i.e., such as log-normal distributions in Test-1, the *NPR* scores tend to be lower. This is due to the variability in sampling from these structured dimensions and more advanced sampling schemes may be utilized to overcome these problems [143]. These results show that even with very small portions of the data used, the online algorithm manages to provide approximate results that are reasonably reliable. It has been reported for PCA that in order to obtain reliable results, one has to use at least around 400 items or keep a 10:1 item to dimension ratio, i.e., at least 10 items per dimension [142]. When the amount of data that can be processed by our algorithm in 1 sec. is considered in the above tests (listed in Table 2), we observe that our algorithm manages to process sample sizes that are sufficient to achieve reliable results. The results also show that the number of data items consumed by the algorithm depends on the number of dimensions of the data. As a result, for the 256 dimensional dataset, our sampling method was not able to keep the 10:1 item to dimension ratio in 1 sec. but was able to maintain the 400 items consideration.

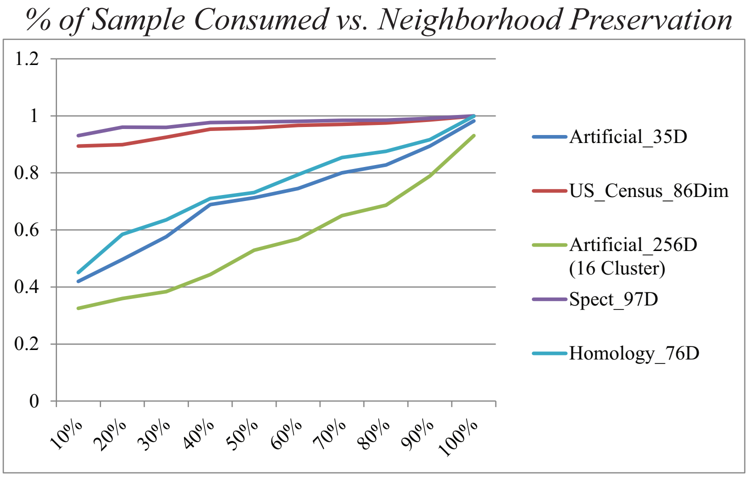


Figure 12: *Neighborhood preservation ratio* values computed for 10 different sample sizes for 5 datasets.

This implies that for datasets with very large dimension counts, the results of the algorithm may be unstable due to the low number of samples that can be consumed within the temporal limitations.

Moreover, we cross the % of samples in Table 2 with the NPR scores in Figure 12. We observe that for most of the datasets, our algorithm manages to reach high NPR scores either in a single iteration (tests 1, 2 and 4) or three to four iterations (tests 3 and 5).

Scope & Limitations of Online Algorithms Researchers in machine learning and in data mining have been developing online algorithms mainly due to their time and memory efficiency when dealing with large datasets, and, the updating capabilities due to changes in the data [102]. However compared to offline versions, i.e., algorithms that process the whole data in a complete *batch*, online algorithms can lead to inaccurate results and might suffer from over-fitting to the data that has been processed [24]. Such problems are tackled via error-bounded methods [44] and advanced sampling strategies [68].

Depending on the task and type of computational tool that needs to be employed, there are incremental versions of different algorithms in literature that can be integrated as the computation module (O in Algorithm 1). In addition to PCA and clustering, online algorithms have been used in classification [68], time series analysis [68], regression analysis [128], and, even for non-linear projection methods such as ISOMAP [121]. However, there are, of course, tasks that are not suitable to approach with online algorithms. Outlier analysis in nov-

Test ID	# of dimensions	# of processed	% of data
1	35	1700	41%
2	86	1200	55%
3	256	210	20%
4	97	532	100%
5	77	1330	22%

Table 2: Performance evaluation for online PCA computations (the # and % of items processed in 1 second).

elty detection or statistical procedures that require highly precise results such as hypotheses testing, may not be easily possible with incremental methods. In such cases, it is advisable for visualization designers to focus on improving the performance of the system, using methods such as pre-computing or caching, to maintain interactivity within the limits of human time constants, one successful example in this line of work is the ATLAS system by Chan et al. [27].

10 Discussion

When working with massive datasets, our methods can enable the user to get an understanding of the data very quickly. After finding interesting relations in the data, e.g., which dimension subset to use for clustering, the user can refer to a sophisticated offline algorithm (which potentially takes a long time to compute) to get more accurate results. This amounts to a more efficient pipeline compared to using the costly algorithms without any prior investigation of the data.

Our online algorithm utilization methodology (Algorithm 1) could be improved further by incorporating a sampling of the dimensions (variables) in addition to the sampling over the data items. This could be achieved through a preprocessing step of the dataset that creates a hierarchical clustering of the variables. This hierarchy can then be used to progressively improve the results similar to a level-of-detail rendering mechanism in computer graphics [67]. Such an addition can improve the robustness and performance of the method when the dimension count gets very large.

When realizing our methods, we focused on the functionality of the mechanisms rather than their performance. Our approach could certainly benefit from a mechanism that involves high-performance computing techniques, for instance those introduced by Piringer et al. [146].

We see a number of visualization problems that can benefit from the suggested three levels of operation. In this paper, we investigated the utilization of our methods on static datasets. However, in the visual analysis of dynamic data, such as temporal datasets, dynamic visualizations are employed frequently. The consideration of the levels of operation in such dynamic systems can lead to visualizations that are perceptually more suitable for the analysis.

As immediate future extension, our sampling and online algorithm approach can be extended to handle streaming data. In essence, our proposed sampling strategy turns a static dataset into a data stream and feeds the data in smaller chunks to the computational tools and the visualization system. In the case of streaming data, however, further improvements to the visualization system need to be incorporated. These include the progressive computation of descriptive measures such as statistics, and the representation of how the computations evolve as new data becomes available.

11 Conclusions

In this paper, we introduce three levels of operation for visual analysis tasks that involve the integration of computational tools and interactive methods. The three levels address important characteristics of humans when they are engaged in a communication (dialogue). We respect the three human time constants [25] and design the temporal characteristics of the three levels, accordingly. With our approach, we take a solid step to realize one of the recommendations in *Illuminating the Path* by Thomas and Cook [187], that reads "*... identify and develop interaction techniques that address the rational human timeframe.*".

We observe that analysis processes can be improved when human factors are considered. As the data size and the complexity of typical analytical questions increase, the careful consideration of human characteristics plays an important role in achieving efficient and effective results.

Paper E

Hypothesis Generation by Interactive Visual Exploration of Heterogeneous Medical Data

Cagatay Turkay[1], Arvid Lundervold[2],
Astri Johansen Lundervold[2], and Helwig Hauser[1]

[1]Department of Informatics, University of Bergen, Norway
[2]Department of Biomedicine, University of Bergen, Norway
[3]Department of Biological and Medical Psychology, University of Bergen, Norway

Abstract

High dimensional, heterogeneous datasets are challenging for domain experts to analyze. A very large number of dimensions often pose problems when visual and computational analysis tools are considered. Analysts tend to limit their attention to subsets of the data and lose potential insight in relation to the rest of the data. Generating new hypotheses is becoming problematic due to these limitations. In this paper, we discuss how interactive analysis methods can help analysts to cope with these challenges and aid them in building new hypotheses. Here, we report on the details of an analysis of data recorded in a comprehensive study of cognitive aging. We performed the analysis as a team of visualization researchers and domain experts. We discuss a number of lessons learned related to the usefulness of interactive methods in generating hypotheses.

This article was published in *Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data. Lecture Notes in Computer Science*, Volume 7947:1–12, 2013.

1 Introduction

As in many other domains, experts in medical research are striving to make sense out of data which is collected and computed through several different sources. Along with new imaging methodologies and computational analysis tools, there is a boom in the amount of information that can be produced per sample (usually an individual in the case of medical research). This increasingly often leads to heterogeneous datasets with very large number of dimensions (variables), up to hundreds or even thousands. This already is a challenging situation since most of the common analysis methods, such as regression analysis or support vector machines [134], for example, do not scale well to such a high dimensionality. Consider for instance applying factor analysis to understand the dominant variations within a 500-dimensional dataset. It is a great challenge to correctly interpret the resulting factors even for the most skilled analyst.

On top of this challenge, the number of samples is usually very low in medical research due to a number of factors such as the availability of participants in a study or high operational costs. This results in datasets with small number of observations (small n) but a very high number of variables (large p). Since most of the statistical methods need sufficiently large number of observations to provide reliable estimates, such "long" data matrices lead to problematic computations [29]. Both the high dimensionality of the datasets and the "$p \gg n$ problem", pose big challenges for the analyst and the computational tools. These challenges lead to the fact that the experts tend to limit their analyses to a subset of the data based on a priori information, e.g., already published related work. Limiting the analysis to a subset of the data dimensions hides relations in the data that can potentially lead to new, unexpected hypotheses.

At this stage, the field of visual analytics can offer solutions to analysts to overcome these limitations [111] [108]. The visual analysis methods enable analysts to quickly build new hypotheses through interaction with the data. The user also gets immediate feedback on whether or not these hypotheses call for a further investigation. Moreover, the interactive tools enable analysts to check for known hypotheses and relationships that have been already studied and reported in the related literature.

In this application paper, we discuss how interactive visual analysis methods facilitate the hypothesis generation process in the context of heterogeneous medical data. We discuss how we utilize the *dual analysis* of items and dimensions [189] in the interactive visual analysis of high dimensional data. We report on the analysis of data related to a longitudinal study of cognitive aging [8] [215]. We demonstrate how our explorative methods lead to findings that are used in the formulation of new research hypotheses in the related study. We additionally showcase observations that are in line with earlier studies in the literature. We then comment on a number of lessons learned as a result of the analysis sessions that we performed as a team of visualization researchers and domain experts.

2 Interactive Visual Analysis Environment

The analysis of the cognitive aging study data is performed through a coordinated multiple view system [202], that primarily makes use of scatterplots. The user is able to make selections in any of the views and combine these selections through Boolean operators, i.e., $\cup, \cap, \neg$. In order to indicate the selections and achieve the focus+context mechanism, we employ a coloring strategy, i.e, the selected points are in a reddish color and the rest is visualized in gray with a low transparency (see Fig. 1-b) to aid the visual prominence of the selection. One additional note here is that we use a density based coloring such that overlapping points lead to a more saturated red color. We use Principal Component Analysis (PCA) – on demand – to reduce the dimensionality of the data when needed. Additionally, we use Multidimensional Scaling (MDS) directly on the dimensions similar to the *VAR display* by Yang et al. [212]. In this visualization approach, the authors represent a single dimension by a glyph that demonstrates the distribution of the items in the dimension. Later authors apply MDS on the dimensions to lay them out on a $2D$-display. Similarly in this work, we feed the correlations between the dimensions as a distance metric to MDS and as a result, it places the highly inter-correlated groups close to each other. These computational analysis tools are available through the integration of the statistical computation package R [184].

The analysis approach employed in this paper is based on the dual analysis method by Turkay et al. [189]. In this model, the visualization of data items is accompanied by visualizations of dimensions. In order to construct visualizations where dimensions are represented by visual entities, a number of statistics, such as mean (μ), standard deviation (σ), *median*, inter-quartile-range (IQR), *skewness*, and, *kurtosis* are computed for each dimension (i.e., column of the data). These computed statistics are then used as the axes of a visualization of dimensions. In Fig. 1-a, the dimensions are visualized with respect to their skewness and kurtosis, where each dot here represents a dimension.

An additional mechanism we employ is the *deviation plot*, which enables us to see the changes in the statistical computations for dimensions in response to a subset selection of items [192]. In Fig. 1-b, we select a sub-group of participants (from the study data) who are older and have a lower education. We now compute the μ and σ values for each dimension twice, once with using all the items (participants) and once with using only the selected subset. We then show the difference between the two sets of computations in a deviation plot (Fig. 1-c). The dashed circle shows the dimensions that have larger values for the selected subset of items, i.e., for the elderly with lower education. Such a visualization shows the relation between the selection and the dimensions in the data and provides a quick mechanism to check for correlations. Throughout the paper, the views that show items have blue background and those that visualize the

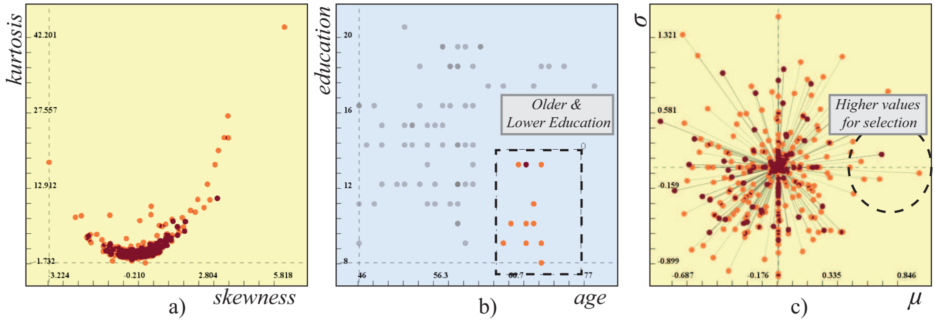


Figure 1: Dual analysis framework where visualizations of items have a blue and those of dimensions a yellow background. a) We employ a visualization of the dimensions over their *skewness* and *kurtosis* values, where each dot represents a single dimension b) We select a group of participants who are older and have a lower education. c) The deviation plot shows how the μ and σ values change when the selection in (b) is made.

dimensions have a yellow background. Further details on the methods could be found in the related references [189] [192].

3 Cognitive Aging Study Data

We analyze the data from a longitudinal study of cognitive aging where the participants were chosen among healthy individuals [8] [215]. All the participants were subject to a neuropsychological examination and to multimodal imaging. One of the expected outcomes of the study is to understand the relations between image-derived features of the brain and cognitive functions in healthy aging [215]. The study involves 3D anatomical magnetic resonance imaging (MRI) of the brain, followed by diffusion tensor imaging (DTI) and resting state functional MRI in the same imaging session [89] [214]. In this paper, we focus on the anatomical MRI recordings together with the results from the neuropsychological examination. The examination included tests related to intellectual function (IQ), memory function, and attention/executive function. IQ was estimated from two sub tests from the Wechsler Abbreviated Scale of Intelligence [206]. The total learning score across the five learning trials of list A (learning), the free short and long delayed recall and the total hits on the Recognition scores from the California Verbal Learning Test (CVLT) II [39] were together with the subtest Coding from Wechsler Adult Intelligence Scale-III [205] used to assess memory function. The Color Word Interference Test from the Delis-Kaplan Executive Function System [40] and the Trail Making Test A and B from the Halstead-Reitan Test Battery [151] were used to assess attention/executive function.

The resulting dataset from the study contains information on 82 healthy individuals who took part in the first wave of the study in 2004/2005. T1-weighted MRI images were segmented into 45 anatomical regions. For each segmented brain region, seven features were derived automatically, namely: *number of voxels*, *volume* and *mean, standard deviation, minimum, maximum and range of the intensity values in the regions.* All these automated computations were done in the FreeSurfer software suite [62]. This automated process creates $45 \times 7 = 315$ dimensions per individual. Additional information on the participants, such as age and sex, and, the results of two neuropsychological tests are added to the data. With this addition, the resulting dataset has 373 dimensions, i.e., the resulting table's size is 82×373. Moreover, meta-data on the dimensions is also incorporated. This meta-data contains whether each dimension is a test score or a brain segment statistic, which brain regions that dimension is related to, and, which statistical feature (e.g., volume or mean intensity) is encoded.

4 Analysis of Cognitive Aging Study Data

In this study, our analysis goal is to determine the relations between age, sex, neuropsychological test scores, and the statistics for the segmented brain regions. The conventional routine to analyze this dataset is to physically limit the analysis to a subset of the dimensions and perform time-consuming, advanced statistical analysis computations on this subset, e.g., loading only the data on specific brain regions and training a neural network with this data. In this setting, if the same analysis needs to be applied on a slightly different subset (which is often the case), all the operations need to be redone from the beginning – a considerably long time to build/evaluate a single hypothesis. On the contrary, in our interactive methods, the whole data is available throughout the analysis and analysts switch the current focus quickly through interactive brushes.

In order to direct the analysis, we treat age, sex, and the test scores as the dependent variables and try to investigate how they relate to the imaging based variables. Moreover, we investigate the relations within the brain segments. In each sub-analysis, we derive a number of observations purely exploratively. We then discuss these findings as an interdisciplinary team of visualization researchers, experts in neuroinformatics and neuropsychology. We comment on the observations using a priori information and suggest explanations/hypotheses around these new findings. These hypotheses, however, needs to be confirmed/rejected through more robust statistical and/or clinical tests to be considered for further studies. Our aim here is to enable analysts to generate new hypotheses that could potentially lead to significant findings when careful studies are carried out.

Prior to our analysis we handle the missing values and perform normalization on the data. To treat missing values, we apply one of the methods known as statistical imputation and replace the missing values with the mean (or mode)

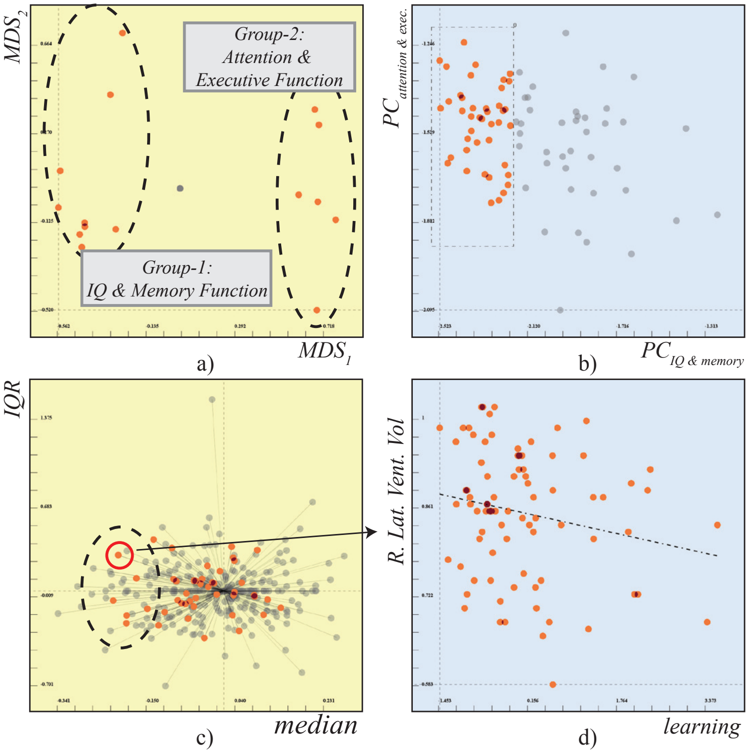


Figure 2: a) MDS is applied on the *test score* dimensions, where related dimensions are placed close to each other. Two groups for the test scores (Group-1: IQ and memory related, Group-2: attention) show up in the results. b) Each group is represented through an application of PCA and the resulting first principal components are mapped to the axes of the scatterplot. A group of participants, who are better in learning and attentive function is selected. c) Some brain regions are smaller for this subgroup, i.e., have smaller *median* value. d) We select one of the dimensions that shrink the most, *right lateral ventricle volume* (red circle), and visualize these values against the learning scores from CVLT. We notice that there is indeed a negative correlation with the learning score from the CVLT.

of each column [163]. We continue with a normalization step where different normalization schemes are employed for different data types. Here, dimensions related to the imaging of the brain are z-standardized and the rest of the columns are scaled to the unit interval.

Inter-relations in Test Results.

We start our analysis by looking at the relations between the test scores. We first focus our attention on the results related to IQ & Memory function and attention/executive functions related tests and apply a correlation-based-MDS on the 15 dimensions. The rest of the dimensions are not used in the computation and are placed in the middle of the view and colored in gray in Fig. 2-a. Here, we choose to focus on the two large groups, that are to the left and to the right of the view. For a micro analysis, one can focus on the sub-groupings that are visible in both of the clusters. The first group relates to test results assessing IQ and memory function (Group-1). The second group relates to test scores assessing attention and executive function (Group-2). This grouping is in line with the interpretation of these scores and we investigate these two sub-groups separately in the rest of the analysis. We interactively select these sub-groups and locally apply PCA on them. We then use the resulting principal components (PC) to represent these two groups of test scores. We observed that for both of the groups much of the variance is captured by a single PC, so we decide to use only the first PC for each group.
Hypothesis 1: There are two dominant factors within the test results, *IQ & memory* and *attention & executive function.*

Findings Based on Sex.

As a continuation of our analysis, we now focus on available meta-data on patients, such as age and sex, to derive interesting relations. We begin by a visualization of *age* vs. *sex* and select the male participants (Fig. 3-a) with a brush and observe how the test scores change in the linked deviation view (Fig. 3-b). The visualization shows that the male participants performed worse in *IQ & memory function* related tasks. In tests related to attention and executive function, however, there were no significant changes between sexes. This is a known finding that has been already observed throughout the study. Another observation that is also confirmed by prior information is the differences in brain volumes between sexes. An immediate reading in Fig. 3-c is that *male participants have larger brains (on average) compared to women*, which is a known fact. We analyze further by selecting one of the regions that changed the most, *Thalamus volume*, and look at its relation with sex (Fig. 3-d). We see that there is a significant change, however, this apparent sex difference in thalamic volume has shown to be negligible when the intracranial volume (ICV) difference between sexes are taken into account [179]. This finding could probably be further explored by normalizing segmented brain volumes with the subject's ICV (if this measure is available).
Hypothesis 2: Males perform worse in *IQ & memory* related tests but not in those related to *attention & executive function.*

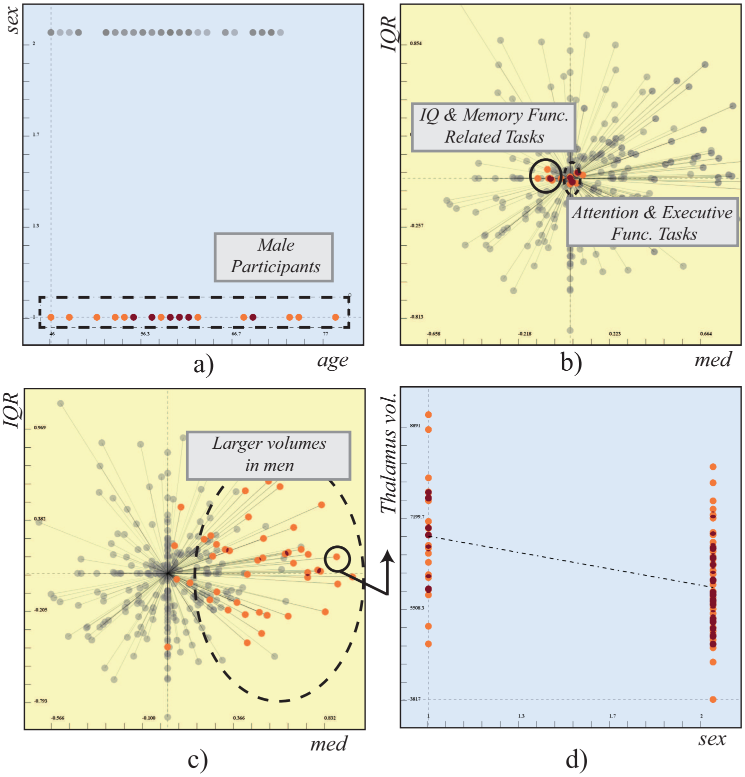


Figure 3: Male participants are selected (a) and the deviation plot shows that for IQ & memory related tasks, males generally perform worse. However, for attentive and executive function related tests, there is no visible difference (b). When the changes in volume for the brain segments are observed, it is clearly seen that males have larger brains (c). When the volume of one of the segments, thalamus, is visualized with a linear regression line, the sex based difference is found to be significant.

Findings Based on Age.

We continue our investigation by limiting our interest to the elderly patients to understand the effects of aging on the brain and the test results. We select the patients over the age of 60 (Fig. 4-a) and visualize how brain volumes and test scores change. We observed no significant difference in IQ & memory and attentive functions for the elderly patients (Fig. 4-b). However, when we observe the change in brain volumes, we observe that there is an overall *shrinkage in most of the brain segments with age.* This is clearly seen in Fig. 4-c, where most of the dimensions have smaller *median* values (i.e., to the left of the center

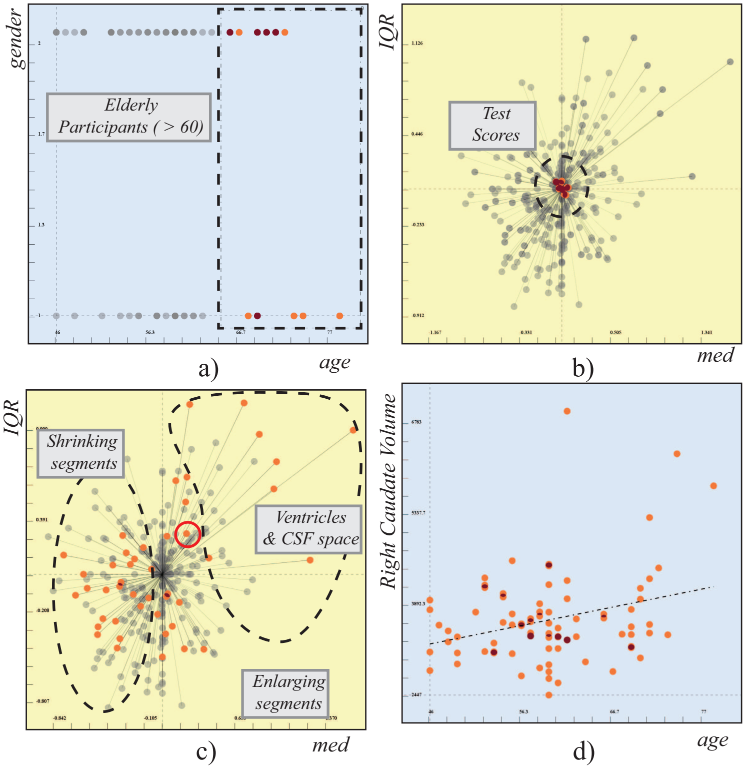


Figure 4: Elderly patients (> 60 years old) are selected (a). No significant relation is observed in the test scores (b). When we focus on the volumes of the segments, we see most of the regions are shrinking with age, but some, especially the ventricles, are enlarging (c). Apart from the expected enlargement of the ventricles, *the right caudate* is also found to enlarge with age (d).

line). Although most of the brain regions are known to shrink with age [200], some regions are reported to enlarge with age. When the dimensions that have a larger *median* value due to the selection (i.e., enlargement due to aging) are observed, they are found to be the *ventricles* (not the *4th ventricle*) and the *CSF space*. Since this is a known fact [200], we focused on the regions that shows smaller enlargements and decide to look at *the right caudate* more closely. When *the right caudate* is visualized against age, a significant correlation is observed (Fig. 4-d). This is an unexpected finding that needs to be investigated further.

Hypothesis 3: There is no significant relation between age and performance in IQ & memory and attentive & executive functions for individuals undergoing a healthy aging. Moreover, in contrast to the most of the brain regions, there is a significant enlargement in *the right caudate* in healthy aging individuals.

IQ & Memory Function vs. Brain Segment Volumes.

We oppose the first principal components for the two groups of test scores (Fig. 2-a) and select the participants that show better IQ & memory function performance (Fig. 2-b). A linked deviation plot shows the change in *median* and *IQR* values where we observe the change in the imaging related variables (Fig. 2-c). We limit our interest to the variables that are the *volumes of the brain segments* by selecting the volume category through a histogram that displays the related meta-data (not shown in the image). In the deviation plot, we see a sub-group of segments (dashed circle) that have lower volumes for the selected participants (i.e., those that showed better performance). Among those segments are the lateral ventricles that show a significant change. Lateral ventricles are filled with cerebrospinal fluid and have no known function in learning and IQ. We use the integrated linear regression computation on a scatterplot showing *learning* vs. *right lateral ventricle volume* and observe that there is in fact a negative correlation. This could be explained such that, when the ventricles have larger sizes, it indicates less gray matter volume in the brain parenchyma responsible in cognitive function, and is thus associated with reduced performance in IQ & memory function. However, although ventricles tend to grow with age, we observed no significant relation between aging and the performance (See Hypothesis 3). These are now two related observations that leads to an interesting hypothesis.
Hypothesis 4: Regardless of age, the larger sizes of the ventricles are associated with low performance. However, the (expected) enlargement of the ventricles with aging does not directly influence the overall performance.

Relations within Brain Segments.

We continue by delimiting the feature set for the brain regions to their *volume* and apply MDS on the 45 dimensions (one for each segment) using the correlation between the dimensions as the distance metric. We identify a group of dimensions that are highly correlated in the MDS plot (Fig. 5-a). This group consists of the volumes for different *ventricles* (lateral, inferior) and *non-white matter hypointensities*. We investigate this finding closely by looking at the relations between *left lateral ventricle* and *non-WM-hypointensities* and found a positive correlation relation (Fig. 5-b) due to a sub-group of patients that have outlying values. This is an interesting finding since non-white matter hypointensities (as segmented by FreeSurfer) might represent local lesions in gray matter such as vascular abnormalities that have a predilection for involving the thalamus and the basal ganglia. Such vascular abnormalities in deeper brain structure could then lead to substance loss and enlarged lateral ventricles. One might further

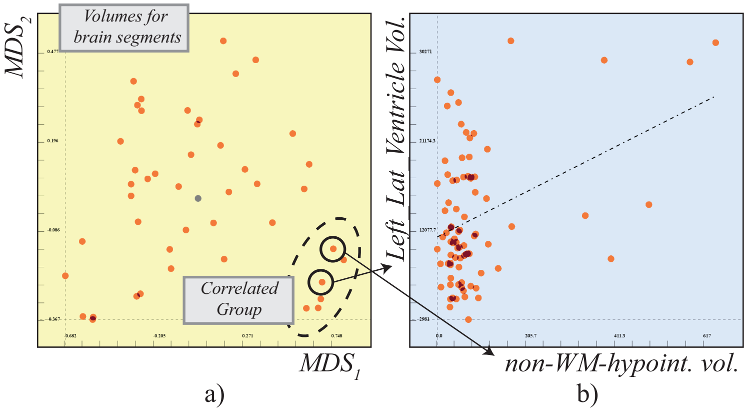


Figure 5: After MDS is applied on the volume dimensions for brain segments, a correlated group of brain segments is observed (a). Although most of these dimensions are related to the volume of different parts of *the ventricles* (which is expected), *non white matter hypointensities* (scars on the white matter) is also related. This is an interesting finding which led to an hypothesis on the relation between the enlargement of the scars on the white matter and the ventricles.

expect that this pathophysiological process would be increasingly frequent with age, but such relationship between age and non-white matter hypointensities was observed to be insignificant in our analysis.

Hypothesis 5: There is a positive relation between lesions on brain tissue and the volume of the ventricles. However, no significant relation with such lesions and age has been detected, this is likely due to the fact that the study involves only participants going through healthy aging.

5 Discussions, Lessons Learned & Conclusions

In a typical analysis of this data, domain experts usually utilize complex machine learning methods, such as neural networks [134], to analyze the data and confirm hypotheses. With such methods however, the process is not transparent and the results can be hard to interpret.

Explorative methods, such as this one presented here, offers new opportunities in building hypotheses. However, the hypotheses built in such systems may suffer from over-fitting to the data, i.e., the finding could be a great fit for a specific selection but harder to generalize [86]. In order to provide feedback on this problem of over-fitting, interactive systems could include cross-validation (or bootstrapping) functionalities to report on the sensibility of the results [115]. In these methods, the hypotheses are tested for several subsets of the data to check the validity of the findings [115]. Another important feature that needs to be present in such interactive systems is the immediate use of more robust and solid statistical verification methods. In our current framework, we employ linear regression to check for the statistical significance of certain relations (see Fig. 4-d). Such functionalities, and even more advanced inferential statistics, are feasible to incorporate through the embedding of R. Such extensions are desirable for domain experts and can increase the reliability of the results considerably in interactive frameworks.

In this work, we only employed scatterplots and the deviation plot. One can easily extend the selection of visualizations using more advanced methods discussed in the literature. The changes can be encoded by flow-based scatterplots [28] and the comparison of groups can be enhanced by using clustered parallel coordinates [98].

In a significantly short analysis session, we were able to build 5 hypotheses from the healthy aging data. Building this many potential hypotheses using the conventional analysis process would require a considerable amount of time. Throughout the analysis, we discovered relations that lead to novel hypotheses for the healthy aging domain. In addition, we came up with a number of findings that have been already confirmed in the related literature.

Acknowledgments

We would like to thank Peter Filzmoser for the valuable insights on the statistical foundations of this work. The study on cognitive aging was supported by grants from the Western Norway Regional Health Authority (# 911397 and #911687 to AJL and 911593 to AL).

Paper F

Characterizing Cancer Subtypes using the Dual Analysis Approach in Caleydo

Cagatay Turkay[1], Alexander Lex[2], Marc Streit[3], Hanspeter Pfister[2], and Helwig Hauser[1]

[1]Department of Informatics, University of Bergen, Norway

[2]School of Engineering and Applied Sciences, Harvard University, Cambridge, MA, USA

[3]Institute of Computer Graphics, Johannes Kepler University Linz, Linz, Austria

Abstract

The comprehensive analysis and characterization of cancer subtypes is an important problem to which significant resources have been devoted in recent years. In this paper we integrate the dual analysis method, which uses statistics to describe both dimensions and rows of a high dimensional dataset, into *StratomeX*, a *Caleydo* view tailored to cancer subtype analysis. We introduce *significant difference plots* for showing the elements of a candidate cancer subtype that differ significantly from other subtypes, thus enabling analysts to characterize cancer subtypes. We also enable analysts to investigate how samples relate to the subtype they are assigned and to the other groups of samples. Furthermore, our approach give analysts the ability to create well-defined candidate subtypes based on statistical properties. We demonstrate the utility of our approach in three case studies, where we show that we are able to reproduce findings from a published cancer subtype characterization.

This article is in submission to: *IEEE Computer Graphics and Applications*, 2013.

1 Introduction

Cancer is one of the most-common causes of death and virtually everyone is or will be either directly or indirectly affected by it. While there has been significant progress in the diagnosis, prevention, and treatment of cancer, there are still many open questions to be answered, methods to be improved, and drugs to be developed. While cancer is a multi-factorial disease, involving environmental factors and lifestyle choices, it has a strong genetic component. In the post-genomic age research on cancer is largely conducted using methods of molecular biology to record and analyze the genetic alterations responsible for cancer. One important field in cancer research is the analysis and characterization of cancer subtypes. While cancers are colloquially referred to by the tissue they originate from (e.g., lung cancer because it occurs in the lung), there are in fact significant differences between cancers from the same tissue, which are characterized by various biomolecular properties. These different forms of cancer are called subtypes. Large scale research projects such as *The Cancer Genome Atlas (TCGA)*[1] elicit comprehensive genomic and clinical datasets with the goal of characterizing the molecular alterations responsible for cancer; and of identifying and characterizing cancer subtypes.

Due to next-generation sequencing and micro-array technology, these projects can utilize large and heterogeneous datasets capturing more aspects of the complex process from the genomic information to the functional consequences than ever before. However, deriving insight from these complex datasets remains a challenging task. Current analysis largely relies on custom scripts to find interesting genes or clusters of patients in these datasets. To remedy this, we have developed *Caleydo StratomeX* [124], an interactive visualization method to analyze and discover relationships within large and heterogeneous biomolecular datasets. StratomeX can be used to evaluate overlaps and relationships of *stratifications* of patients, i.e., groupings or clusterings of patients.

However, StratomeX does not enable analysts to identify the characteristic genes of candidate subtypes, nor does it communicate how patients relate to a given subtype. The former is important since the characteristic genes are also potentially causally involved in a subtype and thus may be a target for a therapeutic or diagnostic approach. The latter, investigating how sample relates to a subtype, can be used to estimate the quality of candidate subtypes and to build a deeper characterization of a subtype.

In this paper, we address these limitations by integrating the *dual analysis approach* [189], a general high-dimensional data analysis methodology, into StratomeX. Our primary contribution is the embedded use of dual analysis views and *significant difference plots*, a novel visual representation of the differences between data subsets, within StratomeX. This approach enables domain scientists

[1] http://cancergenome.nih.gov

to (1) discover genes that are distinctive for specific subtypes, and (2) observe the properties of the member samples of a cluster and compare how they behave in different datasets and clusters. With these, we provide a deeper understanding of the stratifications of heterogeneous genomics datasets. As a secondary contribution, we investigate the potential of the dual analysis approach to interactively generate patient stratifications in StartomeX.

We demonstrate our application in three case studies with data from TCGA and validate our findings against those published by the TCGA consortium.

2 Biological Background and Analysis Tasks

Modern cancer subtype analysis is based on a variety of biomolecular datasets that capture different aspects of the process of life, starting with the information stored in the genome to the functional products that trigger biochemical reactions in the cells. Projects such as TCGA capture information on gene activity, on factors influencing the process of expression, and on the actual structure and sequence of the genome. An example for gene activity data is mRNA data ("gene expression"), which measures the abundance of mRNA in the cell. mRNA is translated into proteins, which are the functional products. Methylation and miRNAs influence the process of gene expression in various ways and thus are an important factor in many processes and diseases.

All these processes play a role in the development of certain cancers, and consequently, a comprehensive analysis solution needs to take all these datasets, in addition to meta-data, such as clinical data about patients, into account. In this paper, we demonstrate our method by investigating mRNA, miRNA, and methylation data. However, in a comprehensive analysis one would also incorporate other datasets, for instance, related to structural variations occurring on various scales in the genome. Such datasets are equally important to get a full picture of the disease.

In previous work, we have elicited analysis tasks for cancer subtype analysis [124]. These tasks are concerned with finding and evaluating stratifications of patients based on multiple datasets. We recently revisited these requirements in collaboration with domain scientists and found the need to supplement them with the following tasks to characterize the stratifications further:

T1 **Find Distinctive Elements**

Identifying distinctive elements of clusters in a stratification provides a deeper understanding of why a particular cluster exists and how it relates to other clusters within the analysis. Distinctive elements are also good candidates to investigate as diagnostic markers or may even be causally involved in the disease.

T2 **Compare Samples**

Investigating the characteristics of the samples over several datasets and in comparison to other stratifications is important in building a more complete picture of the properties of a group of samples. One can observe how strongly the members of a cluster are related and explore whether they show similar properties in a dataset that is different than the one used for clustering.

T3 **Create Clusters**

Analysts should be able to create clusters in an exploratory manner and interactively compare the intermediate results to meta-data such as clinical data. Moreover, this manual clustering process should enable the analyst to merge observations made using different datasets. The thus created clusters are well defined in terms of statistical properties and richer in terms of the sources of information included in the construction phase.

Combined with the previously elicited tasks, this makes it possible to analyze, create, and characterize cancer subtypes based on multiple biomolecular datasets.

3 Methodological Building Blocks

Our solution that enables the aforementioned tasks is based on an integration of two visual analysis methodologies, *Caleydo StratomeX* and the *Dual Analysis Approach*. Before introducing the details of how we improve these methodologies by joining their strengths, we provide brief descriptions of them.

3.1 Caleydo and StratomeX

Caleydo[2] is an open-source visualization framework focused on biomolecular data analysis. Caleydo provides rich functionality for loading and handling multiple heterogeneous datasets as well as stratifications defined on the data. A core strength of Caleydo is the ability to slice datasets into meaningful subsets and to flexibly combine multiple small visualizations of these subsets, using views such as histograms or heat maps, to a fully integrated composite visualization [122]. Caleydo is one of the examples where visual methods have shown to improve the analysis of genomics data [167, 42].

StratomeX is a comparative visualization technique that makes use of this mechanism and enables analysts to investigate the relationships between multiple stratifications. In StratomeX stratifications are represented as columns. Each column consists of multiple stacked “bricks”, where each brick corresponds to a group of patients in the column’s stratification. Ribbons with varying width visualize the overlap between groups of neighboring stratifications, resulting in an overall appearance similar to *Parallel Sets* [116] or *Sankey Diagrams* [153].

[2] http://www.caleydo.org

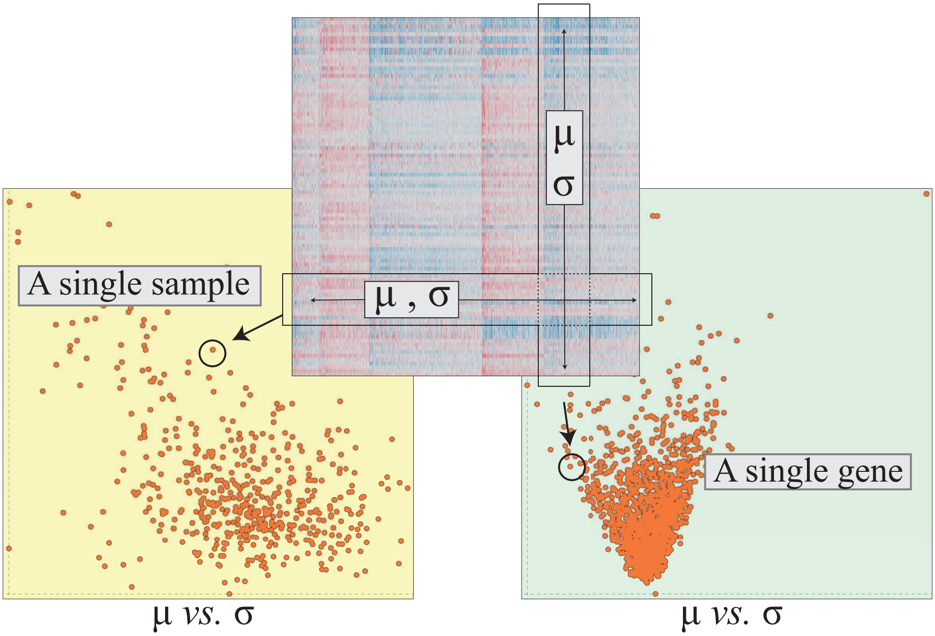


Figure 1: Setting up dual analysis views where the data is depicted as a 2D heatmap for illustration. Samples and genes are visualized in separate views over statistical measures. In order to construct a view that depicts samples (yellow background), statistics, μ and σ in this case, for each sample are computed using a *row* of the data. A visualization for the genes (light-green background), on the other hand, is constructed with statistics computed over a *column* of the data.

Wide ribbons indicate a strong overlap between two groups and thin or absent ribbons correspond to only a few or no shared patients. Each brick contains a visualization showing the data of the patients in that group. Analysts can switch between different types of visualizations on-demand. For numerical data we use clustered heat maps as default views within the bricks in StratomeX, since they are very effective for communicating global trends and patterns in the data.

3.2 The Dual Analysis Approach

The dual analysis approach [189] was shown to be effective in the analysis of high dimensional data. In this method, the visual analysis is carried out in parallel on both the data items and the dimensions. This duality is achieved by using statistics computed both over the rows and the columns of a dataset. The utilization of statistical measures have been shown to be effective in the analysis of datasets from different domains [106].

As an example, consider a mRNA gene expression dataset given as a 2D data table with n rows and p columns, where each row corresponds to a single sample (patient) and each column to a single gene. The expression values are contained in the cells of the matrix.

After appropriate normalization is applied on the data, we calculate the central tendency (μ or $median$) and the spread (standard deviation σ or inter-quartile range IQR) using each one of the n samples and p genes separately. Notice that we calculate the robust counterparts of statistical moments to increase the resistance of the statistics to outlier values. Figure 1 illustrates how the dual analysis views are constructed. Notice that visualizations of samples have a yellow background with each point representing a sample, and visualizations of genes have a light-green background with each point depicting a gene. The location of a single point in a scatterplot is determined by the computed statistics. The analysis process can be elaborated through the use of statistics other than the first two statistical moments. For the analysis that are carried out in this paper, we also compute the *skewness* (*skew*) that indicates how asymmetric a distribution of values is (and also in which direction) and the *kurtosis* (*kurt*) that characterize the "peakedness". Utilization of these measures are demonstrated later in the case studies.

4 Characterizing Cancer Subtypes through Visual Analysis

To facilitate the characterization of cancer subtypes in heterogeneous genomic and clinical datasets, we introduce a visual analysis methodology that makes use of the dual analysis approach to construct specialized views that represent clusters in Caleydo. We achieve this by incorporating two different visualizations as *bricks* in StratomeX: (1) dual analysis based scatterplots depicting either the genes or the samples, and (2) *significant difference plots*. In addition, we also use these visualizations as separate linked views to enhance the interactive visual exploration process and achieve tasks such as manual creation of clusters (Task T3 in Section 2).

4.1 Embedded dual analysis views

In this work, we extend the visualization options for *bricks* in StratomeX with scatterplots of either the genes or the samples constructed using the dual analysis approach. The embedded dual analysis views in StratomeX can be seen in Figure 2. If the embedded scatterplot is a visualization of the samples (having a yellow background), it only displays those samples that are members of the represented cluster (see columns 1 and 2 in Figure 2). On the other hand, if a

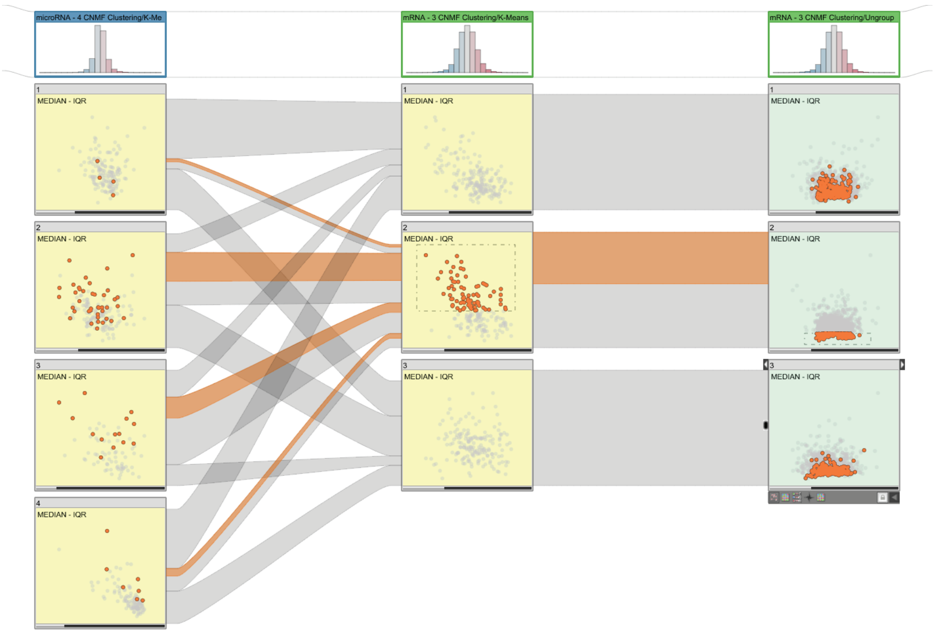

microRNA - 4 CNMF Clustering/K-Me
mRNA - 3 CNMF Clustering/K-Means
mRNA - 3 CNMF Clustering/Ungroup
1
MEDIAN - IQR
2
MEDIAN - IQR
3
MEDIAN - IQR
4
MEDIAN - IQR

Figure 2: Embedded dual analysis views in StratomeX. The first column shows a 4-cluster stratification for a microRNA dataset. The scatterplots show median versus inter-quartile-range for the samples in the cluster. The second column shows a 3-cluster stratification for an mRNA dataset, again showing samples. The third column uses the same 3-cluster stratification for the same dataset, but shows genes instead of samples. The scatterplots of samples (yellow background) depict the statistical characteristics of the members of each cluster and the scatterplots of genes (light-green background) depict statistics computed for the genes using only the samples from the cluster represented by the brick. The selection of samples is highlighted in the first two columns and also in the ribbons. The selection of the genes makes it possible to investigate the distribution of expression values for the genes for different clusters in a stratification.

scatterplot of genes is preferred, the brick displays the statistics for all the genes computed using only the members of the cluster being represented.

We enhance the interactive exploration functionalities by enabling a selection mechanism that is linked with all the views in StratomeX. It is possible to select both samples (selection in the second cluster in the second column of Figure 2) and genes (selection in the second cluster in the third column in Figure 2) at the same time. Also note that the ribbons in StratomeX highlight the selection of the samples in Figure 2.

4.2 Significant difference plots

Since the comparison of subsets is one of the fundamental tasks in tumor subtype analysis, we facilitate the visual comparison of subsets with a novel visualization called *significant difference plot*. In previous work, we used similar plots to effectively display the changes in statistical computations in response to a selection made by the user [192]. In this paper, we extend this approach with the determination and the communication of the significance of the differences being visualized.

Figure 3 illustrates how significant difference plots (or, shortly difference plots) are constructed. The user first selects (brushes) a subset of samples (we denote the set of selected samples as B and the rest as R). In response, the system automatically calculates the μ and σ values for each gene using only the set of selected samples B (μ^B and σ^B) and the rest of samples R (μ^R and σ^R) separately. We then compute the differences between the values with:

$$\Delta_\mu = \mu^B - \mu^R \quad , \quad \Delta_\sigma = \sigma^B - \sigma^R \tag{1}$$

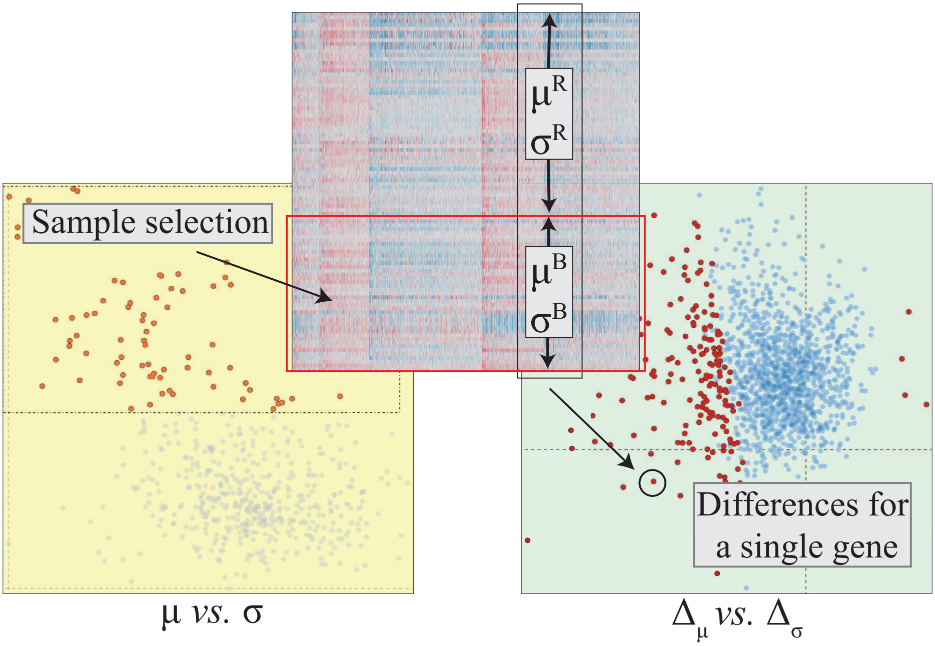


Figure 3: Significant difference view. A set of samples is selected. The differences of the selected samples (B) compared to the not-selected sample (R) is plotted for the genes. Genes that show significant differences are depicted in red and all others in blue.

Note that Δ_μ and Δ_σ are both data vectors of size p, the number of genes. The difference plot then visualizes these values for all the p genes. When there is no difference for the expression values of a gene for subsets S and R, it is placed at the origin $(0, 0)$ of the view.

The difference plot in Figure 3 (right) displays the distribution of the differences in the statistic computations in response to the selection in the scatterplot. Notice that in this example most genes have lower μ values for the selected items, i.e., placed to the left of the y-axis.

Communicating significance – One very important consideration when differences between two subsets are analyzed is the notion of *statistical significance*, i.e., whether the difference occurs by chance or not. As in many other domains, *statistical hypothesis test*s are employed to test for significance in the analysis of genomic data [5]. In this work we enhance difference plots with the integrated use of the statistical hypothesis testing.

In order to compute the significance, we utilize the *two-sample Welch's t-test* as the integrated hypothesis testing procedure [161]. We choose this test since it does not assume that the two subsets have equal variance, which makes it more suitable for our application. We perform the statistical test on the two subsets B and R (as introduced above), and test against the (*null*) hypothesis that these two subsets have equal central tendencies. We compute the t statistic and the degrees of freedom $d.f.$ with:

$$t = \frac{\overline{\mu}_B - \overline{\mu}_R}{\sqrt{\frac{s_B^2}{N_B} + \frac{s_R^2}{N_R}}} \tag{2}$$

$$d.f. = \frac{(s_B^2/N_B + s_R^2/N_R)^2}{(s_B^2/N_B)^2/(N_B - 1) + (s_R^2/N_R)^2/(N_R - 1)} \tag{3}$$

where $\overline{\mu}_i$ is the sample mean, s_i^2 is the sample variance and N_i is the sample size of subsets B and R.

We then use these values together with the t-distribution and test the null hypothesis with a significance level of 0.05 and using a two-tail strategy. This test is performed for all the p genes in the data. For each gene, we store whether it shows a significant difference between the two subsets B and R. We communicate this significant difference information by modifying the color of each gene in the difference plot. Genes that have significant differences are colored red, while the others are shown in blue, as can be seen in Figure 3 (right). This enhancement to the difference view enables analysts to get immediate feedback on the significance of differences. Based on this initial assessment, analysts can employ more advanced routines to confirm the significance of the changes between the two subsets.

Difference plots as bricks – Similar to scatterplots, we also embed difference plots as bricks in StratomeX. While constructing the difference views as bricks, we again compute the Δ_μ and Δ_σ values for each of the genes using Equation 1. Here, however, B corresponds to the samples that are members of the cluster being represented while R corresponds to the rest of the samples in the dataset. In addition, we also compute the significance of the differences and color the visualization accordingly. The resulting difference view bricks communicate which genes are more distinctive for each cluster. Moreover, the selection mechanism enables the analyst to compare these distinctive genes between different clusters. For an utilization of this feature, refer to the first part of Section 5.

5 Case Studies

We demonstrate the effectiveness of our approach by analyzing a comprehensive breast invasive carcinoma (BRCA) dataset collected by the TCGA consortium. We use the mRNA expression data, miRNA sequencing data, and DNA methylation data from over 800 breast cancer patients. The goal of the case studies is to demonstrate how the proposed visual analysis approach enables analysts to execute the three tasks described in Section 2. To begin with, we load the BRCA data which is available pre-packaged for Caleydo. In addition to the raw data, we load a recently published stratification of samples [114] that will serve as a basis for comparisons.

5.1 T1 Case Study: Find Distinctive Elements

We start our analysis by comparing the significantly distinctive genes that are suggested by our computations and those that have been identified in the aforementioned article. The 4 subtypes that are reported in the reference study are: *Luminal-A*, *Basal-like*, *Luminal-B*, and *HER2-enriched*, as shown in Figure 4-a). The reference study identified a list of genes that are differentially expressed for the *HER2-enriched* subtype by using unsupervised clustering (refer to supplementary Table 7 in [114]). We select the 7 most significantly under-expressed genes[3] and 10 most significantly over-expressed genes[4] as marked in Figure 4-a. 7 out of the 7 under-expressed and 6 out of 10 over-expressed genes are identical to the ones found in the reference study. This match demonstrates that our interactive visual analysis approach quickly yields relevant results in determining descriptive genes.

We continue our analysis with the investigation of distinctive genes between particular subtypes (see task **T1**). We focus our attention on the *Luminal-A* subtype and explore the expression characteristics of distinctive genes for *Luminal-A*

[3] *AGR3, ESR1, GFRA1, NPY1R, PGR, SERPINA3, SUSD3*

[4] *ABCA12, CALML5, CLCA2, CRYM, DCD, GLYATL2, MUCL1, NXPH1, PNMT, SOX11*

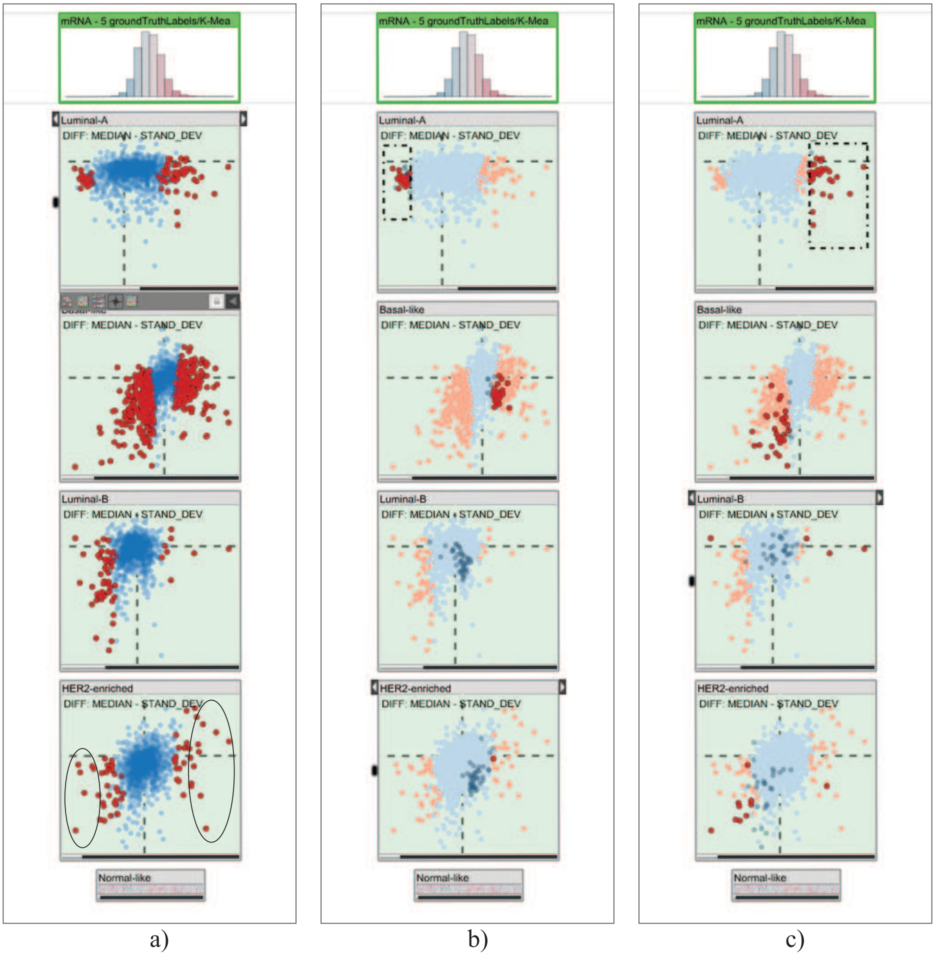


Figure 4: Using embedded difference plots to find descriptive genes. (a) Descriptive genes are marked for the *HER2-enriched* subtype. A comparison to the reference study shows the relevance of the marked genes (b) Under-expressed genes for the Luminal-A subtype are selected and we observe that they show over-expression for the Basal-like subtype, i.e., constitute good features to discriminate these two subtypes. (c) The over-expressed genes for Luminal-A could also be considered good discriminators for this subtype but show similar expression profiles for Basal-like and HER2-enriched subtypes.

in comparison to the other subtypes. We first select the significantly under-expressed genes[5] for the *Luminal-A* subtype in Figure 4-b. We observe that the

[5] *AQP9*, *FAM83D*, *GGH*, *MCM10*, and *MMP1* being some of the lowest

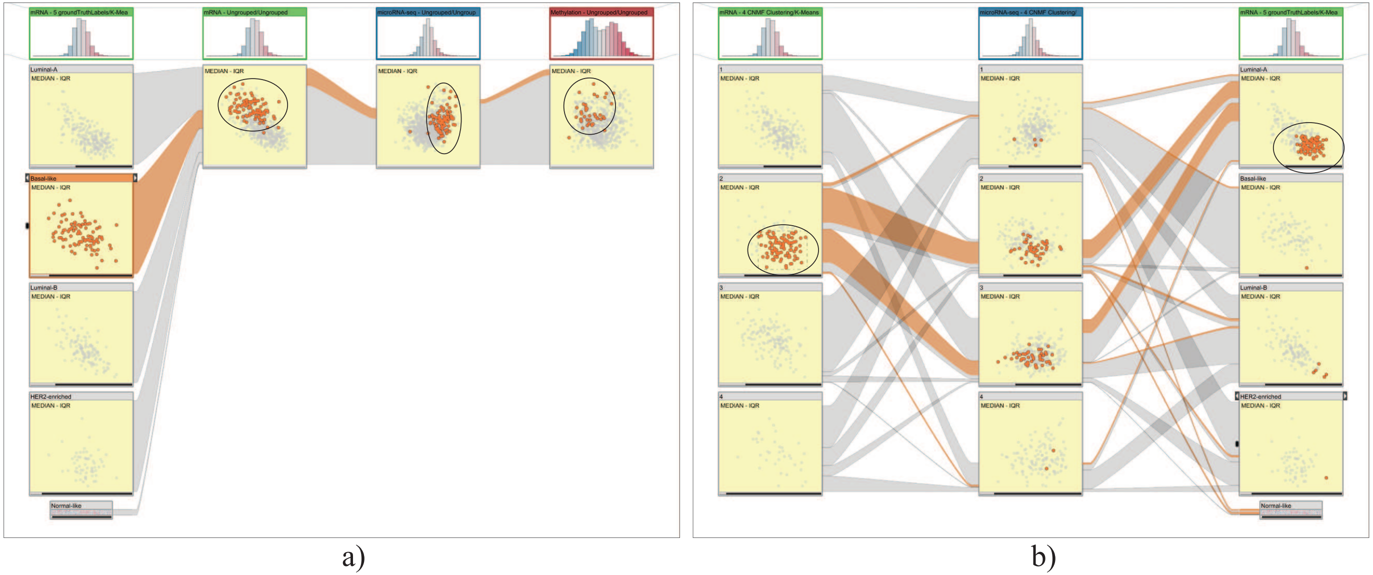

mRNA - 5 groundTruthLabels/K-Mea
mRNA - Ungrouped/Ungrouped
microRNA-seq - Ungrouped/Ungroup
Methylation - Ungrouped/Ungrouped
Luminal-A
Basal-like
Luminal-B
HER2-enriched
Normal-like
MEDIAN - IQR
a)
mRNA - 4 CNMF Clustering/K-Means
microRNA-seq - 4 CNMF Clustering/
mRNA - 5 groundTruthLabels/K-Mea
1
2
3
4
MEDIAN - IQR
Luminal-A
Basal-like
Luminal-B
HER2-enriched
Normal-like
b)

Figure 5: (a) Investigating the sample profiles for *Basal-like* subtype (column 1) over three different datasets (left-to-right: mRNA, microRNA, and methylation). The subtype contains samples with lower values and high variance for mRNA data and usually higher values in the microRNA data. In the methylation data, however, no dominant characteristic is observed. (b) “Core” members of a cluster from a unsupervised stratification of mRNA data are selected (marked, left) and visualized with a microRNA stratification (column 2) and the subtypes. We observe that the selected members correspond to a subgroup in Luminal-A subtype (marked, right).

significantly under-expressed genes for Luminal-A are often over-expressed for the Basal-like subtype. This leads to the conclusion that *these genes are good markers to distinguish the Luminal-A from the Basal-like subtype.* Similarly, when the over-expressed genes are selected for the *Luminal-A* subtype (Figure 4-c), we observe that these genes are under-expressed for *Basal-like* subtype. However unlike the previous set, these genes also show similar expression profiles for the *HER2-enriched* subtype. Consequently, these genes carry less distinctive characteristics compared to the previous set.

5.2 T2 Case Study: Compare Samples

In the second case study we investigate how certain properties of samples from a particular subtype, for instance outliers or trends, are shared among different datasets (**T2**). We start with an investigation of the characteristics of samples from the *Basal-like* subtype by considering the *mRNA*, *microRNA*, and *DNA methylation* datasets. We bring up a StratomeX view with the subtypes from the reference study as the first column and unstratified versions of the datasets mRNA, microRNA, and methylation from left to right, as shown in Figure 5-a. When all the samples from the *Basal-like* subtype are selected, we can observe the following that further characterizes this subtype: samples from the *Basal-like* subtype have *lower expression values with a high variance in mRNA* and have *higher expression values in the microRNA* dataset. When looking at their *DNA methylation values, however, we do not observe any dominant characteristics.*

We use the same approach to determine the characteristics of a cluster that is computed as a result of an unsupervised clustering of the mRNA dataset (first column in Figure 5-b). We select the “core members” of the second cluster, i.e., those which have similar expression values and variance. We observe that these samples do not show any dominant characteristics in an unsupervised clustering of microRNA data (second column in Figure 5-b). However, when considering the reference subtypes from the BRCA study, we observe that the selected samples constitute a subgroup of the *Luminal-A* subtype. We can also see that these samples are the over-expressed *Luminal-A* members with a lower variance. Based on this observation, we can claim that *cluster-2* from the mRNA stratification can be utilized to determine a subgroup of *Luminal-A*.

5.3 T3 Case Study: Create Clusters

In certain cases in tumor subtype analysis, the stratification information is not readily available. In these cases, we make use of the dual analysis methodology to manually create stratifications as an alternative to automated methods. This mechanism enables the analyst to discover structures through different views of multiple datasets and represent these structures as a stratification.

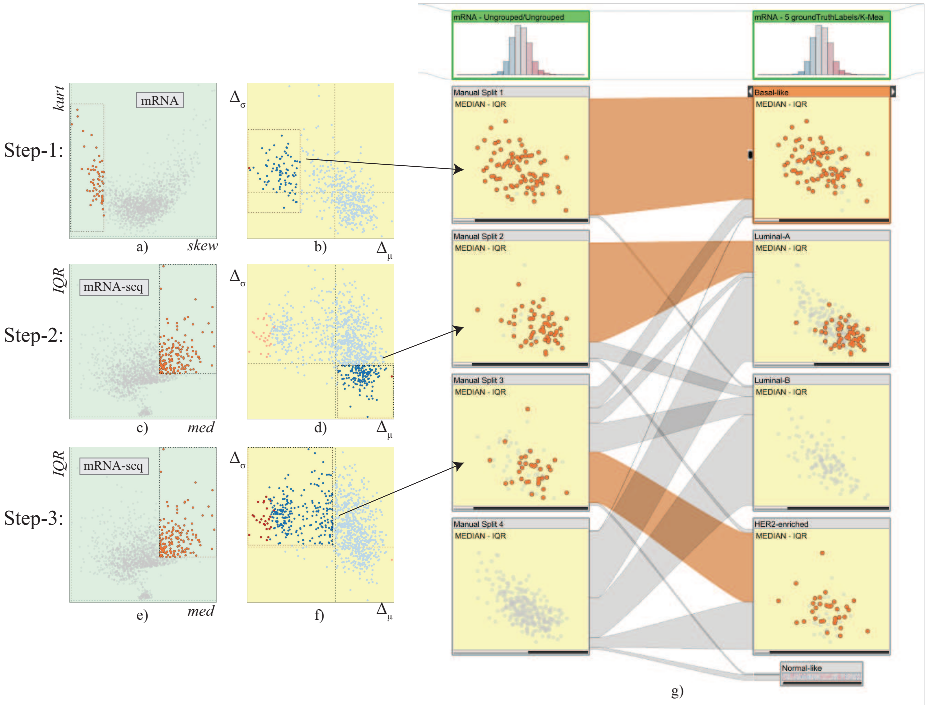


Figure 6: Manual clustering of unstratified mRNA dataset using dual analysis views. Negatively skewed genes are selected through *skew* vs. *kurt* visualization (a) and the difference plot for the samples is updated automatically (b) where we observe a group of samples with lower values and mark them as our first cluster (b). We then switch to mRNA-seq dataset and select the genes that are higher-expressed with a large variety within the values (c,e). We identify two groups and mark them as clusters 2 (d) and 3 (f). For validation, we compare our stratification with the subtypes from the reference study and observe a significant overlap with the subtypes.

For demonstration, we perform such a manual clustering process on the BRCA data. In this process, we use dual analysis views as separate linked views rather than embedded in StratomeX. We bring up two linked views of the *mRNA dataset*: *skew* vs. *kurt* visualization of the genes (Figure 6-a) and a difference plot for the samples for Δ_μ vs. Δ_σ (Figure 6-b). Also, we add two other views of the *mRNA-seq dataset*: *median* vs. *IQR* visualization of the genes (Figure 6-c,e) and a difference plot for the samples for Δ_μ vs. Δ_σ (Figure 6-d,f).

We start by marking an unstratified mRNA dataset as the target for the manual clustering (through a user interface not shown in the images) and clustering process is then as follows:

Step-1: We select the genes that are left-skewed (negative skew values) (Figure 6-a) and select a group of samples that are separated from the rest to the left of the difference view (Figure 6-b). At this point we mark this subset of samples as a stratification of the mRNA dataset (the first cluster in the first column of StratomeX in Figure 6-g). This operation is performed through the UI which is not shown in the image.

Step-2: We now switch to the mRNA-seq dataset and select those genes that have higher expression values and higher variety (Figure 6-c). The difference view is updated automatically and we select those samples which have higher expression values and lower variance (Figure 6-d). We make this selection due to the fact that one would expect to see higher variance and higher values for the samples in response to the selection of genes in Figure 6-c. We finish this step by marking the selection of samples as a second cluster.

Step-3: Without updating the selection of genes, we move on by selecting the samples that have higher variety but smaller mRNA-seq values for the selected genes (Figure 6-f). This last selection of samples is marked as the third cluster in the data. The rest of the samples are left as an unclustered set.

In order to evaluate our custom stratification, we compare it against the classification from the reference study (Figure 6-g). We observe that the cluster made in Step-1, characterized with genes that have negative skewness, has almost a complete overlap with the *Basal-like* subtype. The second cluster from Step-2 largely corresponds to a subgroup of *Luminal-A* subtype. Finally, more than half of the samples from the third cluster belong to the *HER2-enriched*. This overlap between the manually created clusters and the reference subtypes show that the manual clustering leads to relevant results. We have also seen that considering different data sources in the manual clustering steps, e.g., mRNA and mRNA-seq in this case, enables the analyst to merge interesting structures observed in different datasets.

6 Conclusion

In this paper, we integrate dual analysis views and significant difference plots within Caleydo StratomeX, a state-of the art cancer subtype visualization tool. Our approach facilitates the characterization of cancer subtypes by enabling an investigation of them over both the samples and the genes. Such a duality in representing stratifications provide deeper insight on the characteristics of subtypes. The ability to handle multiple datasets in Caleydo extends such insights over to different datasets (such as **T2** in Section 5).

We have also demonstrated how the dual analysis approach can be used to create clusters based on statistical properties and merge structures from different datasets, a challenging task to achieve through automated methods. We have demonstrated the utility of our approach in three case studies. In concert with the existing StratomeX functionality, we believe that we have created a powerful tool for experts to analyze and characterize cancer subtypes.

In the future, we aim to integrate advanced statistical tests and procedures, such as the analysis of variance (ANOVA), or Bonferroni correction [5]. We also consider to extend the capability of difference view to depict the comparison of more than two groups. Furthermore, instead of comparing one cluster to all the other elements, we plan to implement mechanisms to compare clusters with each other.

Acknowledgments

We thank Nils Gehlenborg, Samuel Gratzl, and Christian Partl for their input. This work is supported by the grant J 3437-N15 by the Austrian Science Fund (FWF) and the Air Force Research Laboratory and DARPA grant FA8750-12-C-0300.

Paper G

Interactive Visual Analysis of Temporal Cluster Structures

Cagatay Turkay[1], Julius Parulek[1],
Nathalie Reuter[2], and Helwig Hauser[1]

[1]Department of Informatics, University of Bergen, Norway [2]Department of Molecular Biology, University of Bergen, Norway

Abstract

Cluster analysis is a useful method which reveals underlying structures and relations of items after grouping them into clusters. In the case of temporal data, clusters are defined over time intervals where they usually exhibit structural changes. Conventional cluster analysis does not provide sufficient methods to analyze these structural changes, which are, however, crucial in the interpretation and evaluation of temporal clusters. In this paper, we present two novel and interactive visualization techniques that enable users to explore and interpret the structural changes of temporal clusters. We introduce the temporal cluster view, which visualizes the structural quality of a number of temporal clusters, and temporal signatures, which represents the structure of clusters over time. We discuss how these views are utilized to understand the temporal evolution of clusters. We evaluate the proposed techniques in the cluster analysis of mixed lipid bilayers.

This article was published in *Computer Graphics Forum*, 30(3):711–720, 2011.

1 Introduction

With the advance of data acquisition and simulation systems, large amounts of data with a high number of dimensions and temporally varying values are produced. In various fields like bioinformatics, financial analysis and engineering, it is of great importance to explore and understand the groups of data which share common characteristics over time. These groups are usually analyzed further to gain insight into the processes that are governed by these common characteristics. Cluster analysis is a widely used method to discover grouping structures in both static and time-varying data. This analysis results in a set of clusters, each of which represents a group of similar items with respect to certain features of the data. However, when performing cluster analysis on temporal datasets, interpreting and evaluating the resulting clusters is not as straightforward as it is with static data.

Most of the algorithms developed for clustering time series (temporal) data are either modifications of the static data clustering algorithms, or time-series are converted into static representations such that existing algorithms can be used [125]. Therefore, these clustering algorithms focus mainly on the design of a proper distance function to use in clustering or in the conversion of the data into feature vectors of lower dimensionality. These custom distance functions and conversion operations applied to large, high-dimensional time series may easily produce low-quality clusters [201]. As a consequence, the interpretation and evaluation of clusters become a very important part of cluster analysis. Current methods for cluster assessment, however, are mainly tailored for static data [125], yielding a need for new mechanisms to analyze temporal clusters.

In the following, we illustrate a simple situation where advanced analysis techniques are required to understand the variation of time-dependent cluster structures. We consider a simple scenario as illustrated in Fig. 1. In this setting, two well separated and equally sized groups merge into a single, heterogeneous group at time t_1 and split into two groups again at time t_2. This simple scenario demonstrates a typical example of structural changes which clusters can exhibit over time. Also note that, clustering different time intervals (i.e., t_0, t_1 or t_2) yields completely different clusters.

As the overall clustering structure changes temporally in time-series data, cluster analysis of such data is generally performed over intervals of time [166]. Therefore, unlike clusters of static data, temporal clusters have temporal spans in addition to the group of items they represent. Due to the fact that temporal clusters do not exhibit stable structures usually, both cluster-cluster relations and the structure of temporal clusters vary over time. However, if an experienced user could evaluate such variations, then she/he could consequently discard or update the clusters. The analysis of these variations are not really addressed by the current methods and techniques in cluster analysis. In order to interpret and evaluate temporal clusters, the analyst has to answer at least two questions;

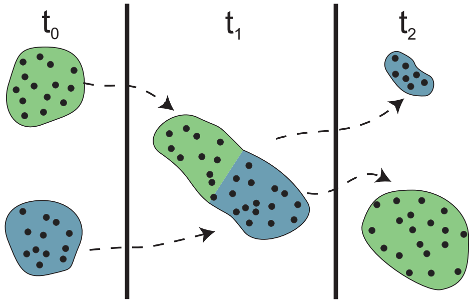


Figure 1: An example of structural changes in clusters of temporal data. Two well-separated clusters (at t_0) merge into a single group at t_1 and split into two groups again at t_2.

firstly, "How does the quality of clusters vary over time?" and secondly, "What type of structural changes do clusters exhibit?".

In this paper, we propose two novel and interactive visualization techniques to analyze temporal clusters. We firstly introduce the *temporal cluster view* that visualizes the structural quality of temporal cluster sets over time. Secondly, we present *temporal signatures* which are visual summaries of temporal cluster structures. The cluster view provides mechanisms to visualize and interactively analyze a set of temporal clusters that are computed from different time intervals. This view also encodes *silhouette coefficients* [157], which are quite widely used cluster structure metrics. They are used to evaluate the structural quality of cluster sets. Temporal signatures are representations of statistical properties of clusters over time. These properties are based on *cluster cohesion* which represents the tightness of its items, and *cluster homogeneity* which correspond to the uniformity of the distribution of the member items [181].

When used in conjunction, these two views provide intuitive mechanisms to analyze and evaluate temporal clusters. They are utilized to explore structural changes in clusters; namely, splitting, merging, and changes in cluster size. We present these two views in an interactive visual analysis framework. To summarize, our contributions in this paper are:

- The temporal cluster view, visualizing a number of temporal clusters together with their structural quality variation.
- Temporal signatures, that are visual representations of the structural changes of groups over time.
- Interactive visual analysis procedures for temporal cluster analysis with the help of these two views.

2 Related Work

Our work relates to the analysis of temporal clusters using interactive techniques and visual representations for temporally varying structures. Thus, the related literature is presented in three subsections:

Analyzing clusters – Vectorized radial visualizations are used in exploring different clustering results by projecting data records on a vectorized cluster space [169]. This approach proves to be useful in validating the clusters when a number of cluster sets for the same dataset exist. Rinzivillo et al. proposes a visually guided clustering called progressive clustering [154], where the clustering is done with different distance functions in successive steps. In hierarchical clustering explorer [167], Seo and Shneiderman use an interactive dendogram, coupled with a color mosaic to represent clustering information in a linked visualization. They propose a cluster comparison view where two clustering results can be compared. However, their method is only suited for clusters of static data. In a recent study, Lex et al. introduce the MatchMaker [123], visualizing and comparing multiple groups of dimensions to represent cluster memberships. Their cluster visualization method is similar to our temporal cluster view, however, their solution does not provide information on the structural quality of clusters over time. Moreover, their method is designed for static clusters only. In the MultiClusterTree [195], Long and Linsen discuss how clusterings are utilized to analyze multi-dimensional data. They use a radial layout, linked with several other views to explore hierarchical clusters. Telea and Auber [185] visualize changes in code structures using a flow layout where they try to identify steady code blocks and when certain splits in the code occur.

Cluster analysis of temporal data – One of the earliest works on cluster-based visualization of temporal data is by Wijk and Selow [198], where they cluster time-series data and visualize them on a calendar. Interactive clustering of trajectory data is discussed in a paper by Andrienko et al. [10], where they describe a user-driven clustering methodology. They use graphical summaries of trajectory clusters to indicate the number of cluster members. These summaries are sufficient when the analyst is interested in changes of the cluster sizes only. In an application of molecular dynamics analysis, Grottel et al. [74] use interactive visual tools to analyze clusters. The authors introduce the concept of flow groups and a schematic view, which displays cluster evolution over time. In a recent study, Rubel et al. [159] introduce a framework that integrates clustering and visualization for the analysis of 3D gene expression data. The authors integrate the data clustering for 3D gene expression analysis into their PointCloudXplore visualization tool. The approach in this study is application oriented, limiting a utilization in other fields. Self organizing maps (SOM) have been utilized in a recent study by Andrienko et al. [9]. They propose the interactive utilization of SOMs that are integrated in a visual analysis framework. Their solution aims to discover spatiotemporal relations by analyzing the temporal evolution of a spatial situation and the distribution of temporal changes sequentially.

Visual representations of temporal data – In this paper, we provide visual a representation of the structural changes of temporal clusters. There is a large number of studies on how to represent temporal data in visualization [204, 135]. One of the important studies which represents temporal changes visually is the ThemeRiver [85] by Havre et al. The authors provide a visual representation of thematic changes in document collections over time. The ThemeRiver visualizes a single value per item and proposes a cumulative representation for each time step. In our temporal signatures, however, we encode a number of temporally varying statistics that are not suitable for a cumulative visualization due to their different scales.

In this paper, we extend the state of the art in the visual analysis of temporal clusters with the temporal cluster view, that integrates temporal clusters into interactive visual analysis procedures, and temporal signatures that visualize the temporal structure of clusters.

3 Overview

The proposed solution for analyzing temporal clusters is based on a new temporal cluster view (in the following just "cluster view") and temporal signatures. Firstly, we introduce the cluster view, that visualizes the quality of clusters together with structural changes that are related to item-cluster and cluster-cluster relationships. Secondly, we present temporal signatures, which are visual summaries of the statistical properties of clusters over time. The variations of these statistical properties reveals structural changes in groups of items.

These two views are utilized in an interactive visual analysis (IVA) cycle to analyze temporal clusters. Prior to the analysis, the analyst constructs a set of temporal clusters using a clustering algorithm. Information from the cluster view and the temporal signatures are combined with information on properties of items as provided by conventional views. The resulting insight is used to interpret and/or validate the clusters. This analysis is performed iteratively until sufficient clusters and insight in group relations is achieved. Fig. 2 is an overview illustration of our solution.

We present our solution in an IVA framework where we incorporate different types of linked views: histograms, scatterplots, parallel coordinates, (for regular variables), and functions graphs and animated scatterplots for temporal variables. In order to update these temporally varying views synchronously, we use a global time parameter τ. We define the dataset of independent variables (items) as $O = \{o_1, \ldots, o_n\}$, where each item has a set of $m = p + q$ dependent values $F(o_i) = [f_1(o_i), \ldots, f_p(o_i), g_{p+1}(o_i, t), \ldots, g_{p+q}(o_i, t)]$. Here, f represents regular

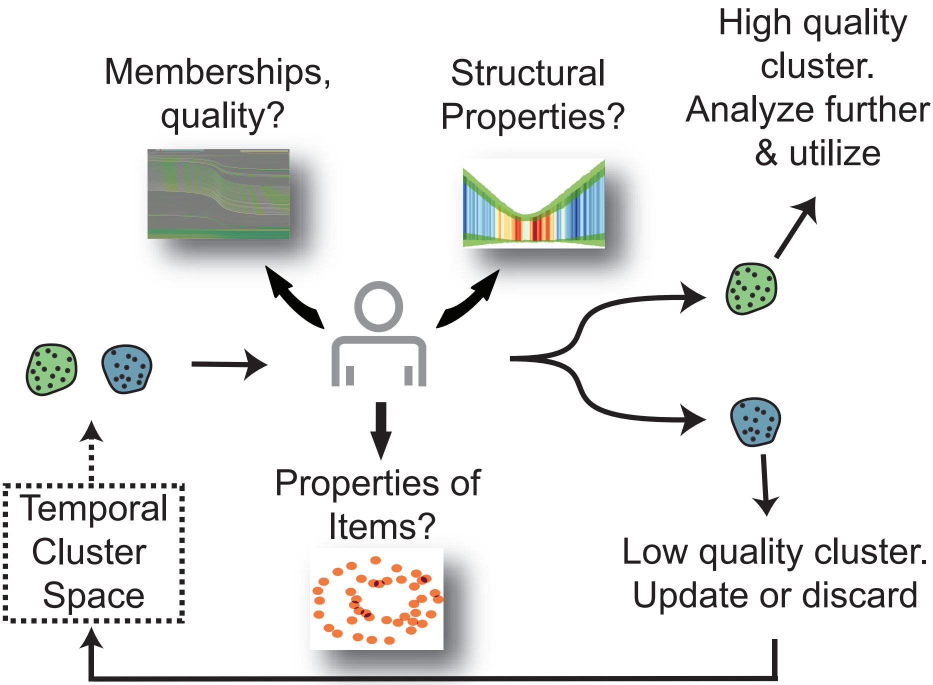

Memberships, quality?
Structural Properties?
High quality cluster. Analyze further & utilize
Temporal Cluster Space
Properties of Items?
Low quality cluster. Update or discard

Figure 2: An overview of our approach. A subset of temporal clusters are analyzed using our techniques and conventional IVA tools in terms of their structural changes and quality variations. Plausible clusters are analyzed to derive more insight on data. Low quality clusters are updated or discarded.

variables and g represents time-series which are defined over time interval $[0, t']$. We define a temporal cluster c_i as:

$$c_i = \{I_{c_i}, T_{c_i} : I_{c_i} \subseteq O, T_{c_i} = [t_0, t_1], 0 \leq t_0 \leq t_1 \leq t'\} \qquad (1)$$

In order to obtain such clusters, the analyst first defines a time interval T and then uses a clustering algorithm to cluster the data in T. This clustering operation is performed k times using different time intervals and/or item subsets which are determined by the user. We refer to the set of clusters obtained at each such step as a *clustering* C_j and the set of all the clusterings as $U = \{C_0, .., C_k\}$ where C_j is defined as:

$$C_j = \{c_1, .., c_{n_j} : \forall c_a, c_b (T_{c_a} = T_{c_b} \wedge c_a \neq c_b \Rightarrow c_a \cap c_b = \emptyset)\} \qquad (2)$$

with n_j as the total number of clusters in C_j. Additionally, we do not necessarily expect C_j to include all the items in O, i.e., $\bigcup_{c \in C_j} \subseteq O$. Note that in a clustering, there are no overlapping clusters in terms of their items. However, it is possible that temporal spans of clusterings can overlap. In this paper, we use both hierarchical and k-means clustering [181]. As these algorithms are originally developed for static data, we modified the distance measures as suggested by Liao [125]. Our solution is well-suited to temporal versions of hierarchical and partitioning clustering algorithms, as they operate on distances between items. However, there exist also other algorithms which operate on densities and statistical models [125]. To generalize our approach to a wider-variety of algorithm results, different quality metrics needs to be included into the analysis procedure.

In our framework, we utilize a brushing mechanism which is similar to *composite brushing* as proposed by Allen and Ward [130]. We extend this mechanism with selections over time. A brush $b = \{I, T\}$ is composed of an item selection, I ($I \subseteq O$), and a time interval selection, T ($[t_0, t_1]$). Each brush is combined with existing brushes by a Boolean operator S with $S \in \{\cup, \cap, \neg\}$, where $\cup$ represents the union, $\cap$ represents the intersection and $\neg$ represents the not operator. The result of this combination is a composite brush B, which is computed "in parallel" as the user makes brushes. Individual brushes b_i are combined into composite brushes B_i using the selected S by $B_i = S(B_{i-1}, b_i)$ starting with $B_1 = S(b_0, b_1)$. For simplicity, in the following, we denote the final set of brushed items as $B_L = \{I_L, T_L\}$. Note that, our definitions of a brush and a cluster (1) is the same, i.e., $b = \{I, T\} = c$. This enables the interpretation of clusters directly as brushes in our system. Due to the fact that non-continuous selections with respect to time would cause an additional complexity in the temporal analysis and related calculations, $\cup$ operator on time results in a single continuous time interval. The resulting time interval encapsulates both input intervals, i.e., $[t_0, t_1] \cup [t_2, t_3] = [min(t_0, t_2), max(t_1, t_3)]$. One other exception in the brushing mechanism is related to the $\cap$ operator in the temporal cluster view. In this view, when two brushes are combined using $\cap$, the item groups are intersected as expected with $\cap$. The temporal spans, however, are joined using $\cup$. This modification enables the use of the $\cap$ operator between clusters defined over non-overlapping temporal spans.

In order to demonstrate our approach in the following, we consider an artificial dataset (Fig. 3). In this dataset, two groups, composed of 20 points each, merge and split at certain points in time. There is a point that moves vertically from the bottom to the top. Additionally, one point shortly gets away from its group and returns back at the first half of the sequence. Prior to the analysis of this dataset, a number of clusterings are added to U. In order to avoid extra complexity in the analysis, we use all the items in consecutive clustering operations.

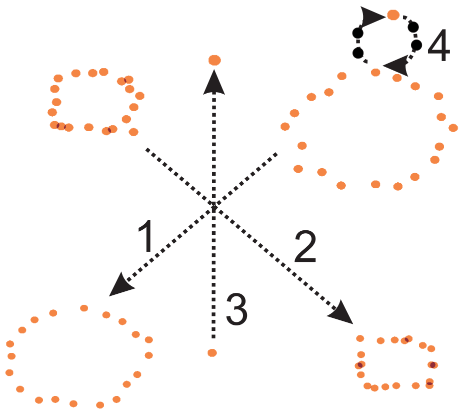


Figure 3: In this artificial dataset, two groups move towards each other following the paths 1 and 2. One point follows path 3 and one item shortly gets away from its group (4).

4 The Temporal Cluster View

The proposed temporal cluster view enables the visual exploration of clusters which are defined over different time intervals. It visually depicts how cluster memberships evolve over time. Moreover, it encodes cluster quality metrics and enables cluster level selections.

In the cluster view, each vertical axis visualizes a clustering C_k, where k indicates the order of the clustering in the view, i.e., for the leftmost axis, $k = 1$ (Fig. 4 a). Each *rectangle* on an axis corresponds to cluster c_i^k in C_k and each curve between the axes represents a single data item, o_i. When the user selects a cluster c in this view, I_c and T_c are handled by the selection mechanism as any other brush b with the above mentioned exception related to the $\cap$ operator.

We visualize the temporal span of clusters in order to link this view to the other temporally updating views. In Fig. 4 a, five clusterings C_{1-5}, performed on different time intervals, are visualized together with their temporal span on top. A black cursor is displayed at the top of the view to indicate τ. Temporal span of the clusterings, which are defined at τ, are highlighted by a saturated red color at the top of the view, e.g., C_2 in Fig. 4 a. Here, brushes b_1 and b_2 are combined using the $\cap$ operator, selecting the intersection of the items and the union of the temporal spans.

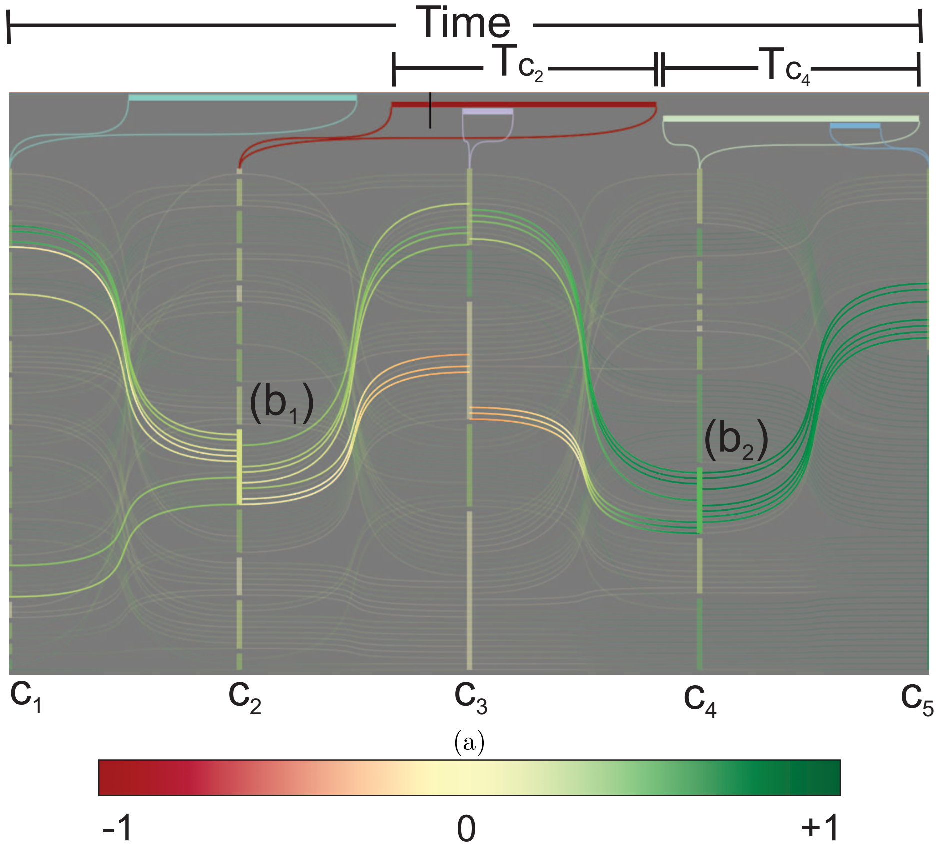
In order to encode information about the structural quality of clusters, we utilize the *silhouette coefficient* [157], which is a popular method in data mining for evaluating the structural quality of clusters. Silhouette values s_i^k are computed per each item of cluster c_i^k and they are in the range $[-1, 1]$. Items close to cluster centers have higher values, and items on the borders of a cluster with close neighboring clusters have values close to 0. Moreover, when an item has a silhouette value close to -1, this item is wrongly placed in this cluster as an artifact of the clustering algorithm. In cluster view, we use silhouette values to color code curves and cluster rectangles. The color coding map, extracted from ColorBrewer [21], is included in Fig. 4 b. The color of a single curve is interpolated between s_i^k and s_i^{k+1} and each cluster rectangle is colored according to the average of the s_i^k values of its members. Here, green colored curves and/or *rectangles* represent high-quality clusters (with respect to silhouette values).

In the cluster view, ordering is crucial for the ease of interpretation. Firstly, we order clusterings C_k according to the "start" of their time intervals T_{C_k}. Secondly, the c_i^k on each axis are ordered with a greedy algorithm in order to minimize overlapping curves between clusters. This ordering starts with the first clustering C_1 placed randomly. The algorithm then continues with the bottom-most cluster c_1^1 of C_1 and finds the cluster $x \in C_2$ which has the biggest overlap with c_1^1, i.e., $arg\,max_{x \in C_2} \left|c_1^1 \cap x\right|$. Then x is placed to the first available position on the second axis. The algorithm continues with c_2^1 and traverses all the clusters on the first axis. The same procedure is then applied for all the axes up to C_{n-1} where n is the number of axes. This crossing minimization problem is a well-known problem called "two layer crossing reduction problem" and more optimized solutions exist in literature [104]. Although it does not provide the optimum solution, we use the presented greedy algorithm due to its low computational complexity and its sufficient outcome for the requirements of our solution. Finally, we order the items in the clusters. All the members of the clusters are first grouped according to the *branches* between C_k and C_{k+1}, where a branch represents overlapping items between two clusters, i.e., $c_i^k \cap c_j^{k+1}$. As the final step in this ordering, all the items in a single branch are organized in an ascending order with respect to s_i^k values. The effect of ordering on the perception of cluster relations and cluster quality is illustrated in Fig 5.

Although our clustering definition (2) allows for items that are not members of any clusters, the clustering algorithms we use in this paper assigns all the items to clusters. In case of items which are not in a cluster (can be referred to as outliers), these items are grouped together and visualized just like any other cluster in the cluster view. If the analyst plans to focus on these outliers, this group of outlier items can be visualized in a distinctive color in the cluster view.

5 Temporal Signatures

In order to explore the structural changes in temporal clusters, we rely on a qualitative approach based on structural statistics, which is easy to interpret,

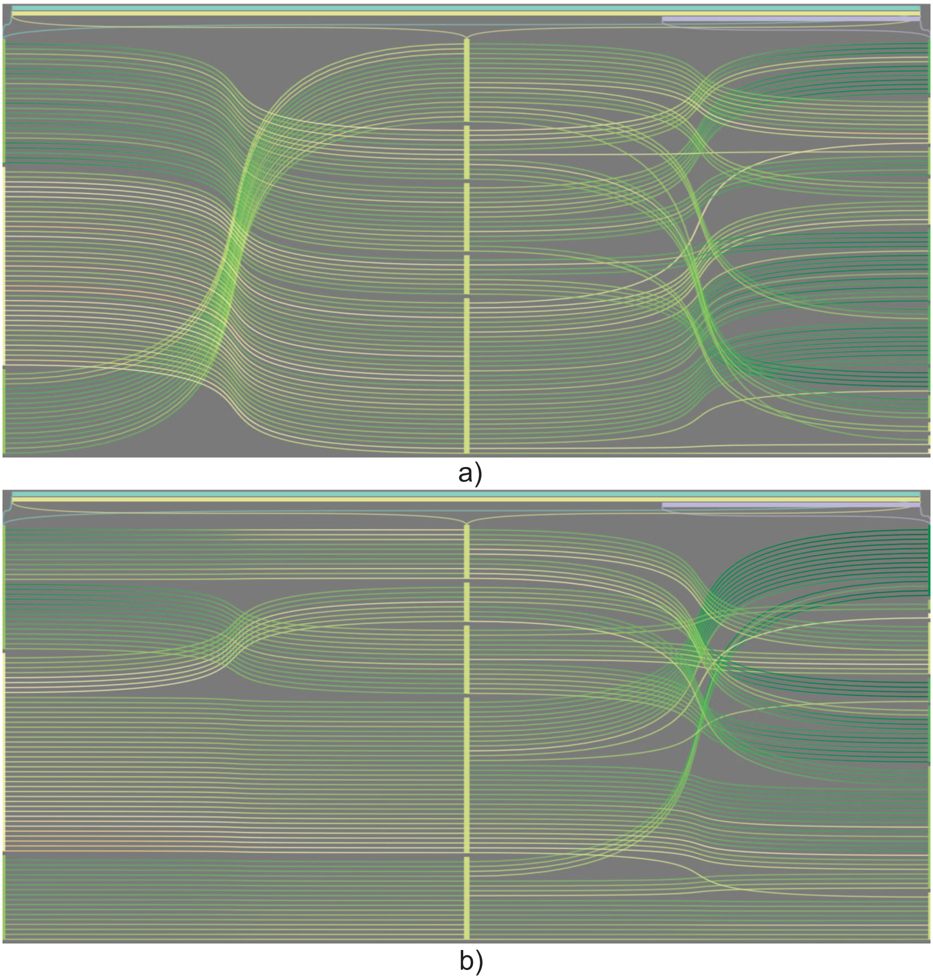


(a)

(b)

Figure 4: (a) Five clusterings visualized in the cluster view. The temporal span of each clustering is visualized on top. Brushes b_1 and b_2 are made to select two clusters. (b) Color coding for silhouette values.

calculate, and visualize. Fig. 6 demonstrates the proposed measures. We utilize a group coherence measure that is based on mutual distances between items in I_L for every time step in T_L. Note that I_L can consist of any group of items that are selected by the brush combinations in the framework. Here, we compute average distance boundaries, which can be thought of as computing the extent covered by points I_L—referred to as *cluster diameter* [52]. The minimum average distance


a)

b)

Figure 5: Ordering cluster view improves the overall perception of cluster quality. Before (a) and after ordering (b).

Min_{avg}^{t} and maximum average distance Max_{avg}^{t} are calculated for all time steps separately as follows:

$$Min_{avg}^{t} = \frac{\sum_{i=1}^{|I_L|} d_{min}^{t}(o_i)}{|I_L|}, \qquad (3)$$

where $d_{min}^{t}(o_i) = min(\{d^t(o_i, o_j) | o_{i,j} \in I_L \wedge o_j \neq o_i\})$ and t represents a single time step. Max_{avg}^{t} is computed likewise with max instead of min in equation (3).

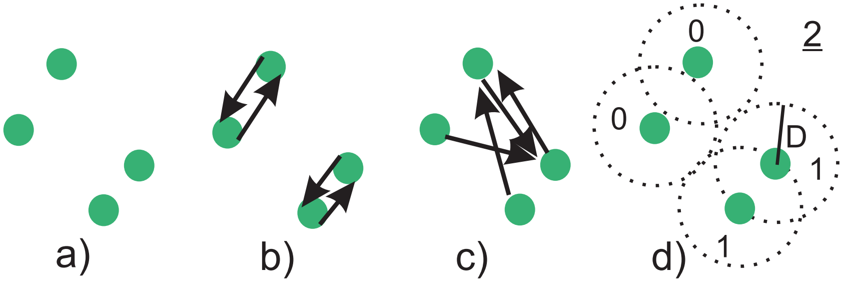


Figure 6: For four 2D points (a), we compute minimum distances (b), maximum distances (c), and vicinity measure V (d). V is the sum of neighboring items within a sphere of radius D $(0 + 0 + 1 + 1 = 2)$.

Per each time step, we additionally compute the sum of number of items "closer" to each other than a distance threshold D. This number, which we refer to as *vicinity measure* V, describes the *compactness* (cohesion) of the group [181]. D is a free parameter, which users can interactively change according to the Min_{avg} and Max_{avg} values. $V^t(D)$ is defined by:

$$V^t(D) = \sum_{i=1}^{|I_L|} \left|\{j | o_j \in I_L \wedge o_j \neq o_i \wedge d^t(o_i, o_j) < D\}\right|. \tag{4}$$

For equations (3) and (4), the Euclidean distance is preferred for $d^t(\cdot, \cdot)$, which is defined as: $d^t(o_i, o_j) = \sqrt{\sum_{k=1}^{q} (g_k(o_i, t) - g_k(o_j, t))^2}$ where g are the temporal variables in our dataset. The selection of distance functions is an essential element of cluster analysis and the utilization of several distance functions can be found in the literature [171]. Therefore, the distance function should be chosen to fulfill domain specific constraints.

The temporal signature view computes the above defined metrics for the currently selected group of items (not necessarily from a cluster) over the selected time interval to construct the visualization. Fig. 7 (left) shows an example of such a temporal signatures view, where I_L contains all the items for the whole time span of the dataset. The upper bound represent maximum average distances, while the lower one represent minimum average distances. We also compute the standard deviations of these distances and render them in a transparent green band around the actual minimum and maximal values. Moreover, we utilize the space between the boundaries to display V values by color intensities. The saturated blue colors represent sparsely distributed items, while the saturated red colors represent packed items, i.e., higher number of neighboring items. The color scaling is done according to the minimum and the maximum values of V for the current I_L and T_L. In Fig. 7 (left), we can observe an instability between

Min_{avg} and Max_{avg} values, where the band gets thinner in the middle as time progresses. This is due to the fact that the groups at t_0 cross each other at t_1 making the overall cluster diameter smaller.

Both standard deviations, $stdev(Min_{avg})$ and $stdev(Max_{avg})$, encodes cluster homogeneity. In Fig. 7, we select first $I_L = c_1$, then $I_L = c_2$, and eventually $I_L = c_1 \cup c_2$ over $T_L = [t_0, t_1]$. The signature of cluster c_1 indicates a high quality cluster due to the stable values of the metrics. However, cluster c_2 contains an outlier (Fig. 3-4), that is recognized through the peaking standard deviations. In general, $stdev(Min_{avg})$ reveals outliers. $stdev(Max_{avg})$ is mainly associated with cluster homogeneity where lower values identify tightly packed items or groups of such tightly packed items. For instance, although group $c_1 \cup c_2$, separates at t_0, the resulting $stdev(Max_{avg})$ values do not vary when c_1 and c_2 merges at t_1, except for the outlier in c_2.

For all the views in Fig. 7, we specify $D = max\{Min_{avg}\}$, which means that there is a number of items above D for all the time steps. This choice of D reveals only the most compact configuration of the items over the whole time interval. In the rightmost signature view in Fig. 7, it can be seen that items are in the most compact form at t_2 (saturated red color) where c_1 and c_2 merges.

Instead of arbitrary groups of items, the analyst can prefer to directly brush clusters. In this case, the signature view enables the user to perform a number of analysis tasks on clusters:

- A single cluster can be visualized to evaluate its temporal structural variations.
- A number of clusters can be brushed using S operators to explore the resulting group's behaviors.
- While a single cluster is selected, the temporal selection (T_L) can be expanded using other brushes. The resulting signature view visualizes how this cluster behaves over time intervals where it is not defined.

6 Temporal Cluster Analysis Procedures

Temporal-cluster analysis aims to find a plausible set of clusters and understand the structural variations of these clusters. The analysis starts with visualizing the selected clusterings in the cluster view and continues with selecting a number of clusters and investigating the corresponding temporal signatures. As a result of the interpretations of these two views, the analyst draws one of these conclusions; validate a cluster, update the temporal span of a cluster or discard a cluster. In order to draw such conclusions, interpretation of silhouette values and discovering where structural changes (like splitting and merging) take place is quite important.

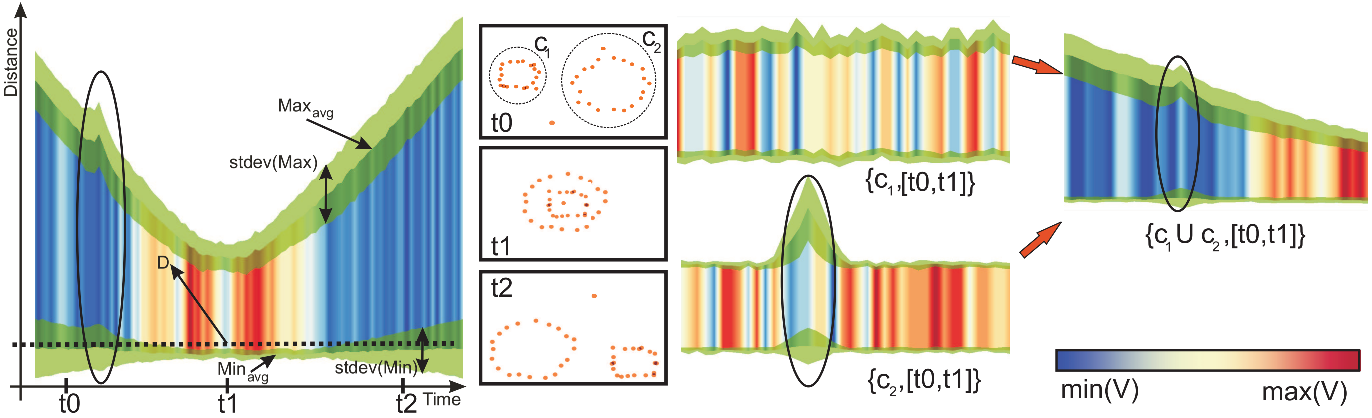


Figure 7: Left: Temporal signatures view. The upper bound represents maximum average distance, the lower represents minimum average distance and the vicinity measure is represented with the color map depicted on the bottom right. The dotted line represents the threshold distance D. The standard deviation is rendered through the transparent green color. Circles mark changes due to movement 4 in Fig. 3. Right: Signature views computed for clusters c_1, c_2 and $c_1 \cup c_2$ over time interval $[t_0, t_1]$.

Interpreting silhouette values – Silhouette values are higher when the clusters are well-separated and more coherent. Therefore, in regions with not so apparent clusters (i.e., where the distribution of items is more uniform), the silhouette values are generally close to zero or even below zero. In Fig. 8, we can see a clear example of such a situation. Here, the example dataset (Fig. 3) is clustered over consecutive time intervals (C_{1-6}). As the distribution of items where two groups meet is quite uniform, we see that the colors of items and clusters turn to yellow. However, near the beginning and at the end of the sequences, the overall cluster quality is high, and this is clearly visible from the colors of C_1 and C_6. This observation yields to the fact that clusters performed over the merging interval are lower in structural quality and therefore, have to considered with more care when further analysis is performed on them.

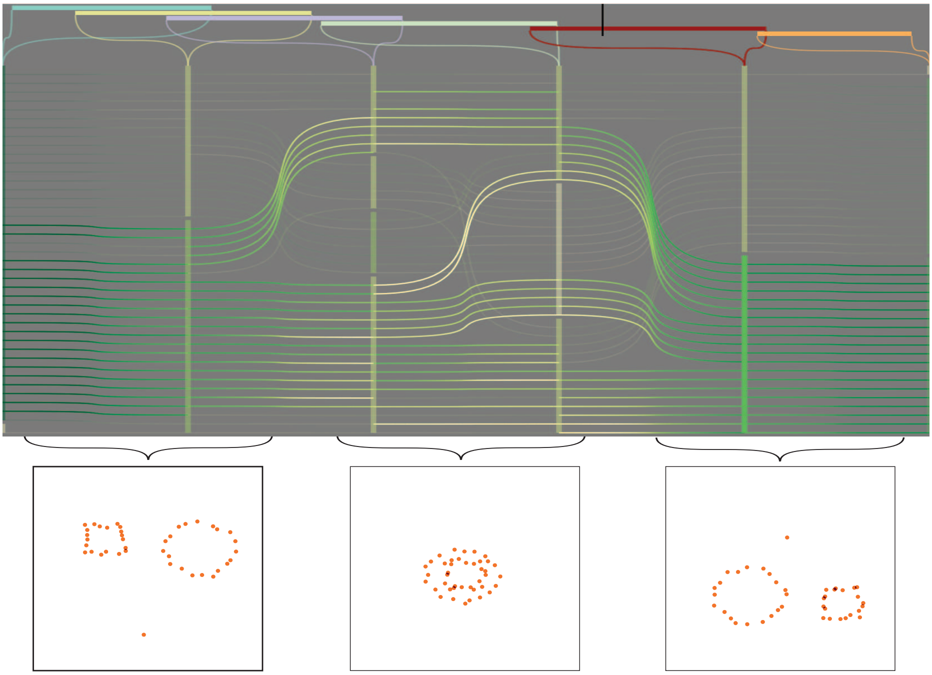


Figure 8: Variation of silhouette values. Group structures change as items move over time. These variations are clearly visible in cluster view by observing the color changes.

Merging and splitting – Two of the important behavior of clusters are merging and splitting. To analyze these behaviors, we firstly brush a cluster by ∩ operation, which may represent a cluster that is about to split or to be created as a union of several other clusters. Secondly, we observe the accompanied temporal signatures view, which reveals this structural tendencies.

In Fig. 9 a, we visualize three sequential clusterings, C_1, C_2 and C_3. We brush a cluster (b_1) in C_2 by $\cap$ brush and by brushes b_2 and b_3 ($\cup$) we extend time selection T_L to contain also time intervals of C_1 and C_3. This extension of time interval is crucial to show the behavior of cluster b_1 in C_1 and C_2. The signatures view is then automatically updated for $I_L = I_{C_2}$ and $[T_{C_1} \cup T_{C_3}]$. A notable tendency is that the band between the Min_{avg} and Max_{avg} gets smaller towards T_{C_2} (*cluster merging*) and gets large again at T_{C_3} (*cluster splitting*). We can additionally observe that I_L has the most compact form where both groups merge and a sparse form where the groups are separated by observing V values.

To generalize, we follow a set of informal rules in evaluating the clusters using our views:

- Items in a cluster should not have many branchings in cluster view
- Cluster rectangle and item curves should be in saturated green
- In a signature of a cluster, values of Min_{avg} and Max_{avg} and the thickness of the band between them should not deviate
- Signatures should mostly contain red values in the band (i.e., high V values)

7 Case Study: Analysis of Molecular Dynamics of Mixed Lipid Bilayers

Molecular modeling of biological membranes is one of the application fields where analysis of temporal clusters is particularly useful. Cell membranes separate the interior of cells from the environment and are mostly constituted of a mixture of different lipids. The lipids can form microdomains or clusters with other membrane components. Such microdomains are relevant for signal transduction or cell apoptosis to name but a few [53]. Lipid bilayers are widely used to model and study cell membranes, and molecular dynamics (MD) simulations are utilized as powerful tools to describe their atomic structure and dynamic behavior. These simulations run on a mixture of different types of lipids that form different cluster sets. These lipid clusters can lead to inhomogeneity in biological membranes [23].

Here we use a dataset obtained from MD simulation of a mixed lipid bilayer [23], constituted of DMPC (dimirystoilphosphatidylcholine) and DMPG (dimirystoilphosphatidylglycerol) lipids composed of 1640 time steps. Each lipid is represented by one particle, localized at the position of the phosphorus atom. Additionally, we work on a set of clusterings $\{C_1, \ldots, C_n\}$ that are computed as the final step of the simulation phase.

Our aim here is to evaluate clusters by their stability over time. In case of a plausible cluster (with respect to cluster view and to signatures view), we perform

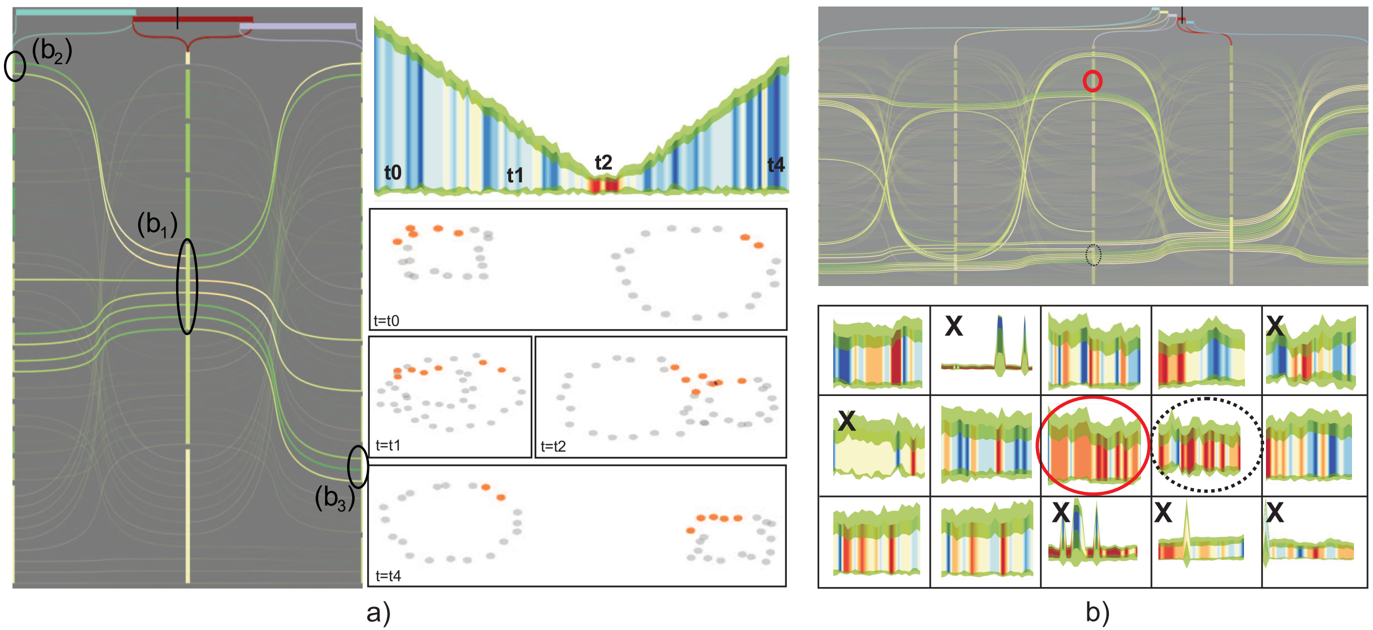


Figure 9: a) Cluster merging-splitting behavior. A cluster is selected with b_1 and the time selection is enlarged by brushes b_2 and b_3. Merging occurs around the smaller band in the middle, which gets larger at end of the sequence due to splitting in signature view. b) Searching for a plausible cluster. Two good signatures are identified (circles). The dashed circle is discarded due to its structural instability in cluster view (shown with the selection on the right). The red circled cluster is picked for further analysis. Moreover, the observed signatures allow to discard clusters (**X**) according their structure.

additional IVA analyses to specify the time span where the cluster preserves its structure.

We start the analysis by displaying the clusterings in the cluster view. Then we assess the cluster quality, firstly, by brushing individual clusters by $\cap$ operation according to silhouette values, and secondly, assessing the cluster coherence via the signature view. Here, we use the set of rules described in Section 6. Fig. 9 b displays a set of signatures for the observed clusters C_{1-5} defined over sequential time intervals $T_{C_{1-5}}$.

In Fig. 9 b, although the signature for the cluster marked with dotted circle represents a good cluster; the cluster structure over time is not stable due to branching in cluster view. Therefore, this cluster is not picked for further analysis. Discarded clusters are marked with an X in the figure. Nevertheless, we found a cluster (marked with a red circle in Fig. 9 b) that has a plausible signature and exhibits a stable structure in the neighboring clusterings. We continue our analysis with this cluster c in C_3 (Fig. 10). As the next step, we enlarge the time selection, from $T_L = T_{C_3}$ to $T_L = [T_{C_1} \cup T_{C_5}]$. The corresponding signature Fig. 10 (left-bottom) depicts the stability of cluster c even for the remaining intervals. The stability is observed by the band width between minimum average distance Min_{avg} and maximum average distance Max_{avg}. The group extend is preserved over T_L since $stdev(Max_{avg})$ has the same width for the whole time. However, $stdev(Min_{avg})$ exhibits certain instabilities which are caused by oscillation movements of cluster boundary lipids that gets away from the group for a few time steps. Additionally, we continue by extending time interval T_L with a brush on time domain (not shown in the figure) to analyze how stable this group is over a larger time interval. With this update, we observe that the signature changes rapidly for latter regions (Fig. 10 (top-right)). This limits the time extend of this cluster to the first peak (arrow). However, later on, we can see that the vicinity values, depicted by colors, are close to red again, identifying that the same group is forming. Since we observe this region where the cluster can be defined, we add clustering C_6 for this region of interest. We see in Fig. 10 (bottom-right) that cluster c is formed again, even for this small interval.

Our collaborators working in the field of biomolecular modelling state that, in their previous work on a similar dataset [23] they faced many limitations in performing analysis on group behaviors. Due to the complexity of analyzing the clusters over time, they were doing the clustering on individual time steps and average the clustering properties over time. As they were not able to relate the structure of these separate clusterings, they were computing properties of them and analyze the changes of these values over time. These statistics involve basic properties like the number of clusters and the number of items in clusters at each time step. In their analyses, it was not possible to explore the behavior and quality of clusters over time. They state that our framework provides significant improvements in the analysis of MD simulations of lipid bilayers. The proposed framework enables the discovery of grouping behaviors which can lead to new

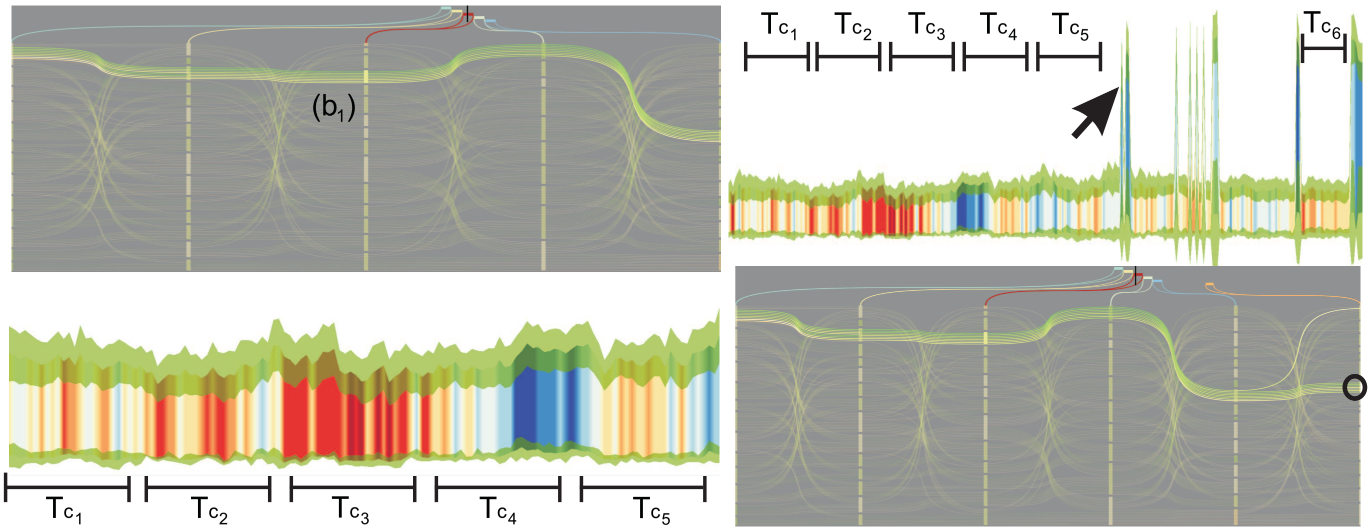


Figure 10: Lipid cluster development. Top left: Coherent cluster $c = b_1$ in all clusterings, C_{1-5}. Bottom left: The signature view for c with extended time interval to showcase signatures in remaining clustering intervals $T_L = [T_{C_1} \cup T_{C_5}]$, where it expresses high stability. Top right: We extend T_L to search for “existing” boundaries (arrow) for cluster c, where we mark another coherent interval T_{C_6}. We add cluster C_6, where we observe that items in b_1 reforms cluster c again at T_{C_6} (circle).

hypotheses on the relations of lipids in lipid bilayers. During this case study, we came across a number of additional analysis tasks like: identifying the threshold deviations for the “good” lipid clusters, analysis of vanishing clusters and determining the time when the overall system stabilization takes place. These are potential tasks where our analysis framework can be utilized. In general, our collaborators find the procedure to be faster, more powerful and more reliable than traditional approaches which are usually based on distance criteria applied to each frame of the sequence.

8 Conclusion

In this paper, we introduce two novel visualization techniques for the interactive visual analysis of temporal clusters. We firstly introduce cluster view, which interactively visualizes a number of clusters defined on temporal intervals. This view visualizes the variation of the structural quality of clusters by representing the changes of silhouette coefficients. Cluster view visualizes the temporal span of clusters in order to enable the exploration of clusters over time. Secondly, we present temporal signatures which are visual representations of the structure of a group of items over time. This view encodes a number of time-varying statistical properties of a group to depict its structural transformations. We show how these views enable an intuitive analysis of temporal clusters, where the analyst is able to determine the validity of the clusters and interpret the relations that cause structural changes in clusters. To the best of our knowledge, our solution is the first interactive visual approach to analyze the structural changes in cluster-cluster and item-cluster relations of temporal datasets.

We integrated our visualizations into an IVA environment where we performed visual analysis of temporal clusters. Cluster view enables cluster level interactions and when used in combination with temporal signatures view, it provides a mechanism to explore temporal clusters in terms of their structural properties. We describe analysis procedures which enables the analyst to explore the quality of clusters over time and explore the structural changes exhibited by clusters. As a consequence of these analyses, the clusters are either validated, updated or discarded. The analyst then continues with the further analyses of high quality clusters.

We evaluated our methodologies on the analysis of molecular dynamics simulation, where the analyst is trying to build hypotheses on the grouping behaviors of lipid-bilayers. We show that our methods reveals certain groups which exhibit stable behavior over distinct time intervals. Such behavior patterns provides the basis to make hypotheses on the behavioral properties of lipid bilayers.

As a future work, we plan to extend our temporal signatures with more robust statistics and different quality metrics, which can provide deeper insight on the structure of groups of items. Another future direction is to create abstract representations of the structural changes and encode them in the form of an event based visualization system.

Acknowledgments

NR acknowledges funding from the Bergen Research Foundation (BFS; Bergens Forskningstiftelse) and the University of Bergen. We thank Torben Broemstrup for molecular dynamics simulation data.

Bibliography

[1] ACM. KDD Cup 2004. http://www.sigkdd.org/kddcup/, 2004.

[2] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. In *Proceedings of the 1998 ACM SIGMOD international conference on Management of data*, pages 94–105. ACM, 1998.

[3] Z. Ahmed and C. Weaver. An Adaptive Parameter Space-Filling Algorithm for Highly Interactive Cluster Exploration. In *Procedings of IEEE Symposium on Visual Analytics Science and Technology (VAST)*, 2012.

[4] S. Albers. Online algorithms: A survey. *Mathematical Programming*, 97(1):3–26, 2003.

[5] D. B. Allison, X. Cui, G. P. Page, and M. Sabripour. Microarray data analysis: from disarray to consolidation and consensus. *Nature Reviews Genetics*, 7(1):55–65, 2006.

[6] E. Alpaydin. *Introduction to machine learning*. MIT press, 2004.

[7] J. Alsakran, Y. Chen, Y. Zhao, J. Yang, and D. Luo. Streamit: Dynamic visualization and interactive exploration of text streams. In *Pacific Visualization Symposium (PacificVis), 2011 IEEE*, pages 131–138. IEEE, 2011.

[8] M. Andersson, M. Ystad, A. Lundervold, and A. Lundervold. Correlations between measures of executive attention and cortical thickness of left posterior middle frontal gyrus - a dichotic listening study. *Behavioral and Brain Functions*, 5(41), 2009.

[9] G. Andrienko, N. Andrienko, S. Bremm, T. Schreck, T. Von Landesberger, P. Bak, and D. Keim. Space-in-time and time-in-space self-organizing maps for exploring spatiotemporal patterns. *Computer Graphics Forum*, 29(3):913–922, 2010.

[10] G. Andrienko, N. Andrienko, S. Rinzivillo, M. Nanni, D. Pedreschi, and F. Giannotti. Interactive visual clustering of large collections of trajectories. In *Visual Analytics Science and Technology, 2009. VAST 2009. IEEE Symposium on*, pages 3–10. IEEE, 2009.

[11] D. Archambault, H. Purchase, and B. Pinaud. Animation, small multiples, and the effect of mental map preservation in dynamic graphs. *Visualization and Computer Graphics, IEEE Transactions on*, 17(4):539–552, 2011.

[12] K. Bache and M. Lichman. UCI machine learning repository, 2013.

[13] W. Barfield and C. Hendrix. The effect of update rate on the sense of presence within virtual environments. *Virtual Reality*, 1(1):3–15, 1995.

[14] R. A. Becker and W. S. Cleveland. Brushing scatterplots. *Technometrics*, 29(2):127–142, 1987.

[15] B. B. Bederson and A. Boltman. Does animation help users build mental maps of spatial information? In *Information Visualization, Proceedings. IEEE Symposium on*, pages 28–35. IEEE, 1999.

[16] C. Bentley and M. Ward. Animating multidimensional scaling to visualize n-dimensional data sets. In *Information Visualization, Proceedings IEEE Symposium on*, pages 72–73, 126, 1996.

[17] W. Berger, H. Piringer, P. Filzmoser, and E. Gröller. Uncertainty-aware exploration of continuous parameter spaces using multivariate prediction. *Computer Graphics Forum*, 30(3):911–920, 2011.

[18] E. Bertini and D. Lalanne. Investigating and reflecting on the integration of automatic data analysis and visualization in knowledge discovery. *SIGKDD Explor. Newsl.*, 11(2):9–18, 2010.

[19] J. Bezdek and R. Hathaway. Vat: a tool for visual assessment of (cluster) tendency. In *Neural Networks, 2002.*, volume 3, pages 2225 –2230, 2002.

[20] A. Blum and P. Langley. Selection of relevant features and examples in machine learning. *Artificial intelligence*, 97(1-2):245–271, 1997.

[21] C. A. Brewer. http://www.colorbrewer.org/, 2009.

[22] R. Bro, E. Acar, and T. G. Kolda. Resolving the sign ambiguity in the singular value decomposition. *Journal of Chemometrics*, 22(2):135–140, 2008.

[23] T. Broemstrup and N. Reuter. Molecular Dynamics Simulations of Mixed Acidic/Zwitterionic Phospholipid Bilayers. *Biophysical journal*, 99(3):825–833, 2010.

[24] L. Carbonara and A. Borrowman. A comparison of batch and incremental supervised learning algorithms. *Principles of Data Mining and Knowledge Discovery*, pages 264–272, 1998.

[25] S. K. Card, G. G. Robertson, and J. D. Mackinlay. The information visualizer, an information workspace. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '91, pages 181–186, New York, NY, USA, 1991. ACM.

[26] E. Catmull. The problems of computer-assisted animation. *SIGGRAPH Comp. Graph.*, 12(3):348–353, 1978.

[27] S. Chan, L. Xiao, J. Gerth, and P. Hanrahan. Maintaining interactivity while exploring massive time series. In *Visual Analytics Science and Technology. IEEE Symposium on*, pages 59–66. IEEE, 2008.

[28] Y.-H. Chan, C. D. Correa, and K.-L. Ma. Flow-based scatterplots for sensitivity analysis. In *Visual Analytics Science and Technology (VAST), 2010 IEEE Symposium on*, pages 43–50. IEEE, 2010.

[29] J. Chen and Z. Chen. Extended bic for small-n-large-p sparse glm. *Statistica Sinica*, 22(2):555, 2012.

[30] A. Cheriyadat and L. Bruce. Why principal component analysis is not an appropriate feature extraction method for hyperspectral data. In *Geoscience and Remote Sensing Symposium, 2003. IGARSS'03. Proceedings. 2003 IEEE International*, volume 6, pages 3420–3422. IEEE, 2003.

[31] J. Choo and H. Park. Customizing computational methods for visual analytics with big data. *Computer Graphics and Applications, IEEE*, 33(4):22–28, 2013.

[32] W. S. Cleveland and M. E. Mac Gill. *Dynamic graphics for statistics.* CRC Press, 1988.

[33] J. Cohen, P. Cohen, S. G. West, and L. S. Aiken. *Applied multiple regression/correlation analysis for the behavioral sciences.* Lawrence Erlbaum, 2003.

[34] M. F. Cohen, S. E. Chen, J. R. Wallace, and D. P. Greenberg. A progressive refinement approach to fast radiosity image generation. *SIGGRAPH Comput. Graph.*, 22(4):75–84, 1988.

[35] D. Cook and D. Swayne. *Interactive and Dynamic Graphics for Data Analysis: with R and GGobi.* Springer, 2007.

[36] C. D. Correa, Y.-H. Chan, and K.-L. Ma. A framework for uncertainty-aware visual analytics. In *Proc. IEEE Symposium on Visual Analytics Science and Technology VAST 2009*, pages 51–58, 2009.

[37] R. J. Crouser and R. Chang. An affordance-based framework for human computation and human-computer collaboration. *Visualization and Computer Graphics, IEEE Transactions on*, 18(12):2859–2868, 2012.

[38] T. N. Dang, A. Anand, and L. Wilkinson. Timeseer: Scagnostics for high-dimensional time series. *Visualization and Computer Graphics, IEEE Transactions on*, 19(3):470–483, 2013.

[39] D. C. Delis. *California verbal learning test.* Psychological Corporation, 2000.

[40] D. C. Delis, E. Kaplan, and J. H. Kramer. *Delis-Kaplan executive function system.* Psychological Corporation, 2001.

[41] Denny, G. J. Williams, and P. Christen. Visualizing temporal cluster changes using relative density self-organizing maps. *Knowl. Inf. Syst.*, 25(2):281–302, 2010.

[42] J. Dietzsch, N. Gehlenborg, and K. Nieselt. Mayday-a microarray data analysis workbench. *Bioinformatics*, 22(8):1010–1012, 2006.

[43] H. Doleisch, M. Gasser, and H. Hauser. Interactive feature specification for focus+context visualization of complex simulation data. In *Proceedings of the symposium on Data visualisation 2003*, VISSYM '03, pages 239–248. Eurographics Association, 2003.

[44] P. Domingos and G. Hulten. A general method for scaling up machine learning algorithms and its application to clustering. In *In Proceedings of the Eighteenth International Conference on Machine Learning*, pages 106–113. Morgan Kaufmann, 2001.

[45] D. L. Donoho et al. High-dimensional data analysis: The curses and blessings of dimensionality. *AMS Math Challenges Lecture*, pages 1–32, 2000.

[46] P. Dragicevic, A. Bezerianos, W. Javed, N. Elmqvist, and J.-D. Fekete. Temporal distortion for animated transitions. In *Proceedings of the 2011 annual conference on Human factors in computing systems*, pages 2009–2018. ACM, 2011.

[47] S. Dudoit, J. Fridlyand, and T. Speed. Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American statistical association*, 97(457):77–87, 2002.

[48] G. Ellis and A. Dix. A taxonomy of clutter reduction for information visualisation. *Visualization and Computer Graphics, IEEE Transactions on*, 13(6):1216–1223, 2007.

[49] N. Elmqvist, P. Dragicevic, and J. Fekete. Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. *Visualization and Computer Graphics, IEEE Transactions on*, 14(6):1539–1148, 2008.

[50] A. Endert, L. Bradel, and C. North. Beyond control panels: Direct manipulation for visual analytics. *Computer Graphics and Applications, IEEE*, 33(4):6–13, 2013.

[51] A. Endert, C. Han, D. Maiti, L. House, and C. North. Observation-level interaction with statistical models for visual analytics. In *Visual Analytics Science and Technology (VAST), 2011 IEEE Conference on*, pages 121–130. IEEE, 2011.

[52] B. S. Everitt, S. Landau, and M. Leese. *Cluster Analysis.* Wiley Publishing, 4th edition, 2009.

[53] J. Fan, M. Sammalkorpi, and M. Haataja. Formation and regulation of lipid microdomains in cell membranes: Theory, modeling, and speculation. *FEBS letters*, 584(9):1678–1684, 2010.

[54] A. Farcomeni. An exact approach to sparse principal component analysis. *Computational Statistics*, 24(4):583–604, 2009.

[55] J. Fekete and C. Plaisant. Interactive information visualization of a million items. In *Information Visualization, 2002. IEEE Symposium on*, pages 117–124. IEEE, 2002.

[56] J.-D. Fekete. How to achieve interactive visual analysis for visual analytics, 2012. Slides of a talk given at Dagstuhl Seminar 12081, Information Visualization, Visual Data Mining and Machine Learning.

[57] S. Fernstad, J. Johansson, S. Adams, J. Shaw, and D. Taylor. Visual exploration of microbial populations. In *Biological Data Visualization (BioVis), 2011 IEEE Symposium on*, pages 127 –134, 2011.

[58] P. Filzmoser, K. Hron, and C. Reimann. Principal component analysis for compositional data with outliers. *Environmetrics*, 20(6):621–632, 2009.

[59] P. Filzmoser, R. Maronna, and M. Werner. Outlier identification in high dimensions. *Computational Statistics & Data Analysis*, 52(3):1694–1711, 2008.

[60] B. Fischl, D. Salat, E. Busa, M. Albert, M. Dieterich, C. Haselgrove, A. Van Der Kouwe, R. Killiany, D. Kennedy, S. Klaveness, et al. Whole brain segmentation: automated labeling of neuroanatomical structures in the human brain. *Neuron*, 33(3):341–355, 2002.

[61] D. Fisher, I. Popov, S. Drucker, et al. Trust me, i'm partially right: incremental visualization lets analysts explore large datasets faster. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems*, pages 1673–1682. ACM, 2012.

[62] FreeSurfer. http://surfer.nmr.mgh.harvard.edu, 2012.

[63] Y. Frishman and A. Tal. Online dynamic graph drawing. *Visualization and Computer Graphics, IEEE Transactions on*, 14(4):727–740, 2008.

[64] Y.-H. Fua, M. O. Ward, and E. A. Rundensteiner. Hierarchical parallel coordinates for exploration of large datasets. In *Proceedings of the conference on Visualization'99: Celebrating ten years*, VIS '99, pages 43–50. IEEE Computer Society Press, 1999.

[65] R. Fuchs and H. Hauser. Visualization of multi-variate scientific data. *Computer Graphics Forum*, 28(6):1670–1690, 2009.

[66] R. Fuchs, J. Waser, and M. E. Gröller. Visual human+machine learning. *IEEE TVCG*, 15(6):1327–1334, 2009.

[67] T. Funkhouser and C. Séquin. Adaptive display algorithm for interactive frame rates during visualization of complex virtual environments. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pages 247–254. ACM, 1993.

[68] M. M. Gaber, A. Zaslavsky, and S. Krishnaswamy. Mining data streams: a review. *ACM Sigmod Record*, 34(2):18–26, 2005.

[69] T. Golub, D. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. Mesirov, H. Coller, M. Loh, J. Downing, M. Caligiuri, et al. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531, 1999.

[70] L. J. Gosink, C. Garth, J. C. Anderson, E. W. Bethel, and K. I. Joy. An application of multivariate statistical analysis for query-driven visualization. *IEEE Transactions on Visualization and Computer Graphics*, 17:264–275, 2011.

[71] R. O. Green, M. L. Eastwood, C. M. Sarture, T. G. Chrien, M. Aronsson, B. J. Chippendale, J. A. Faust, B. E. Pavri, C. J. Chovit, M. Solis, et al. Imaging spectroscopy and the airborne visible/infrared imaging spectrometer (aviris). *Remote Sensing of Environment*, 65(3):227–248, 1998.

[72] T. Green, W. Ribarsky, and B. Fisher. Visual analytics for complex concepts using a human cognition model. In *Visual Analytics Science and Technology. IEEE Symposium on*, pages 91–98. IEEE, 2008.

[73] A. L. Griffin, A. M. MacEachren, F. Hardisty, E. Steiner, and B. Li. A comparison of animated maps with static small-multiple maps for visually identifying space-time clusters. *Annals of the Association of American Geographers*, 96(4):740–753, 2006.

[74] S. Grottel, G. Reina, J. Vrabec, and T. Ertl. Visual verification and analysis of cluster detection for molecular dynamics. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1624–1631, 2007.

[75] I. D. Guedalia, M. London, and M. Werman. An on-line agglomerative clustering method for nonstationary data. *Neural Comput.*, 11(2):521–540, 1999.

[76] Z. Guo, M. O. Ward, and E. A. Rundensteiner. Model space visualization for multivariate linear trend discovery. In *Proc. IEEE Symp. Visual Analytics Science and Technology VAST 2009*, pages 75–82, 2009.

[77] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *The Journal of Machine Learning Research*, 3:1157–1182, 2003.

[78] J. Hair and R. Anderson. *Multivariate data analysis*. Prentice Hall, 2010.

[79] P. Hall, D. Marshall, and R. Martin. Incremental eigenanalysis for classification. In *British Machine Vision Conference*, volume 1, pages 286–295, 1998.

[80] D. Harrison et al. Hedonic housing prices and the demand for clean air. *Journal of Environmental Economics and Management*, 5(1):81–102, 1978.

[81] M. Harrower and C. Brewer. Colorbrewer. org: an online tool for selecting colour schemes for maps. *Cartographic Journal, The*, 40(1):27–37, 2003.

[82] J. Hartigan and P. Hartigan. The dip test of unimodality. *The Annals of Statistics*, 13(1):70–84, 1985.

[83] H. Hauser. Generalizing focus+context visualization. In *Scientific visualization: The visual extraction of knowledge from data*, pages 305–327. Springer, 2006.

[84] H. Hauser. The iterative process of interactive visual analysis. Keynote talk at the EuroVA 2012 workshop, 2012.

[85] S. Havre, E. Hetzler, P. Whitney, and L. Nowell. Themeriver: Visualizing thematic changes in large document collections. *IEEE Transactions on Visualization and Computer Graphics*, 8:9–20, 2002.

[86] D. M. Hawkins et al. The problem of overfitting. *Journal of chemical information and computer sciences*, 44(1):1–12, 2004.

[87] J. Heer and G. Robertson. Animated transitions in statistical data graphics. *Visualization and Computer Graphics, IEEE Transactions on*, 13(6):1240–1247, 2007.

[88] V. Hodge and J. Austin. A survey of outlier detection methodologies. *Artificial Intelligence Review*, 22(2):85–126, 2004.

[89] E. Hodneland, M. Ystad, J. Haasz, A. Munthe-Kaas, and A. Lundervold. Automated approaches for analysis of multimodal mri acquisitions in a study of cognitive aging. *Comput. Methods Prog. Biomed.*, 106(3):328–341, 2012.

[90] S. Huang, M. Ward, and E. Rundensteiner. Exploration of dimensionality reduction for text visualization. In *Coordinated and Multiple Views in Exploratory Visualization, 2005.(CMV 2005). Proceedings. Third International Conference on*, pages 63–74. IEEE, 2005.

[91] P. Huber and E. Ronchetti. *Robust statistics.* Wiley, 2009.

[92] S. Ingram, T. Munzner, V. Irvine, M. Tory, S. Bergner, and T. Möller. Dimstiller: Workflows for dimensional analysis and reduction. In *Visual Analytics Science and Technology (VAST), 2010 IEEE Symposium on*, pages 3 –10, 2010.

[93] G. Ivosev, L. Burton, and R. Bonner. Dimensionality reduction and visualization in principal component analysis. *Analytical chemistry*, 80(13):4933–4944, 2008.

[94] H. Jänicke, M. Böttinger, and G. Scheuermann. Brushing of attribute clouds for the visualization of multivariate data. *IEEE Transactions on Visualization and Computer Graphics*, pages 1459–1466, 2008.

[95] C. M. Jarque and A. K. Bera. Efficient tests for normality, homoscedasticity and serial independence of regression residuals. *Economics Letters*, 6(3):255–259, 1980.

[96] D. Jeong, C. Ziemkiewicz, B. Fisher, W. Ribarsky, and R. Chang. ipca: An interactive system for pca-based visual analytics. *Computer Graphics Forum*, 28(3):767–774, 2009.

[97] J. Johansson, P. Ljung, and M. Cooper. Depth cues and density in temporal parallel coordinates. In *Proceedings of the 9th Joint Eurographics/IEEE VGTC conference on Visualization*, pages 35–42. Eurographics Association, 2007.

[98] J. Johansson, P. Ljung, M. Jern, and M. Cooper. Revealing structure within clustered parallel coordinates displays. In *Information Visualization, 2005. INFOVIS 2005. IEEE Symposium on*, pages 125–132, 2005.

[99] S. Johansson and J. Johansson. Interactive dimensionality reduction through user-defined combinations of quality metrics. *Visualization and Computer Graphics, IEEE Transactions on*, 15(6):993–1000, 2009.

[100] R. Johnson and D. Wichern. *Applied multivariate statistical analysis*, volume 6. Prentice Hall Upper Saddle River, NJ:, 2007.

[101] E. Kandogan. Just-in-time annotation of clusters, outliers, and trends in point-based data visualizations. In *Visual Analytics Science and Technology (VAST), 2012 IEEE Conference on*, pages 73–82. IEEE, 2012.

[102] R. M. Karp. On-line algorithms versus off-line algorithms: How much is it worth to know the future? In *Proceedings of the IFIP 12th World Computer Congress on Algorithms, Software, Architecture*, pages 416–429, 1992.

[103] L. Kaufman and P. J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis.* Wiley, 2005.

[104] M. Kaufmann and D. Wagner. *Drawing graphs: methods and models.* Springer Verlag, 2001.

[105] B. Kégl. Intrinsic dimension estimation using packing numbers. *Advances in Neural Information Processing Systems*, pages 697–704, 2003.

[106] J. Kehrer, P. Filzmoser, and H. Hauser. Brushing moments in interactive visual analysis. *Computer Graphics Forum*, 29(3):813–822, 2010.

[107] J. Kehrer and H. Hauser. Visualization and visual analysis of multifaceted scientific data: A survey. *Visualization and Computer Graphics, IEEE Transactions on*, 19(3):495–513, 2013.

[108] J. Kehrer, F. Ladstädter, P. Muigg, H. Doleisch, A. Steiner, and H. Hauser. Hypothesis generation in climate research with interactive visual data exploration. *IEEE Transactions on Visualization and Computer Graphics (IEEE TVCG)*, 14(6):1579–1586, 2008.

[109] J. Kehrer, P. Muigg, H. Doleisch, and H. Hauser. Interactive visual analysis of heterogeneous scientific data across an interface. *IEEE Transactions on Visualization and Computer Graphics*, 17(7):934–946, 2011.

[110] D. Keim. Information visualization and visual data mining. *IEEE transactions on Visualization and Computer Graphics*, 8(1):1–8, 2002.

[111] D. Keim, G. Andrienko, J. Fekete, C. Görg, J. Kohlhammer, and G. Melançon. Visual analytics: Definition, process, and challenges. *Information Visualization*, pages 154–175, 2008.

[112] D. Keim, F. Mansmann, J. Schneidewind, J. Thomas, and H. Ziegler. Visual analytics: Scope and challenges. *Visual Data Mining*, pages 76–90, 2008.

[113] D. A. Keim, J. Kohlhammer, G. Ellis, and F. Mansmann. *Mastering The Information Age-Solving Problems with Visual Analytics.* Florian Mansmann, 2010.

[114] D. Koboldt, R. Fulton, M. McLellan, H. Schmidt, J. Kalicki-Veizer, J. McMichael, L. Fulton, D. Dooling, L. Ding, E. Mardis, et al. Comprehensive molecular portraits of human breast tumours. *Nature*, 490(7418):61–70, 2012.

[115] R. Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *International joint Conference on artificial intelligence*, volume 14, pages 1137–1145. Lawrence Erlbaum Associates Ltd, 1995.

[116] R. Kosara, F. Bendix, and H. Hauser. Parallel sets: Interactive exploration and visual analysis of categorical data. *IEEE Transactions on Visualization and Computer Graphics*, 12(4):558–568, 2006.

[117] R. Kosara, S. Miksch, and H. Hauser. Semantic depth of field. In *Information Visualization, 2001. INFOVIS 2001. IEEE Symposium on*, pages 97–104, 2001.

[118] H.-P. Kriegel, P. Kröger, and A. Zimek. Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. *ACM Transactions on Knowledge Discovery from Data*, 3(1):1–58, 2009.

[119] A. Laird, J. Lancaster, and P. Fox. Brainmap. *Neuroinformatics*, 3(1):65–77, 2005.

[120] O. D. Lampe, J. Kehrer, and H. Hauser. Visual analysis of multivariate movement data using interactive difference views. In *Proceedings of Vision, Modeling, and Visualization (VMV 2010)*, pages 315–322, 2010.

[121] M. H. C. Law, N. Zhang, and A. K. Jain. Nonlinear manifold learning for data stream. In *In Proc. SIAM International Conference for Data Mining*, pages 33–44, 2004.

[122] A. Lex, H.-J. Schulz, M. Streit, C. Partl, and D. Schmalstieg. VisBricks: multiform visualization of large, inhomogeneous data. *IEEE Transactions on Visualization and Computer Graphics (InfoVis '11)*, 17(12):2291–2300, 2011.

[123] A. Lex, M. Streit, C. Partl, K. Kashofer, and D. Schmalstieg. Comparative analysis of multidimensional, quantitative data. *Visualization and Computer Graphics, IEEE Transactions on*, 16(6):1027 –1035, 2010.

[124] A. Lex, M. Streit, H. Schulz, C. Partl, D. Schmalstieg, P. J. Park, and N. Gehlenborg. StratomeX: visual analysis of Large-Scale heterogeneous genomics data for cancer subtype characterization. *Computer Graphics Forum (EuroVis '12)*, 31(3):1175–1184, 2012.

[125] W. Liao. Clustering of time series data–a survey. *Pattern Recognition*, 38(11):1857–1874, 2005.

[126] Z. Liao, Y. Yu, and B. Chen. Anomaly detection in gps data based on visual analytics. In *Visual Analytics Science and Technology (VAST), 2010 IEEE Symposium on*, pages 51–58. IEEE, 2010.

[127] A. Lundervold. On consciousness, resting state fmri, and neurodynamics. *Nonlinear biomedical physics*, 4:1–18, 2010.

[128] J. Ma, J. Theiler, and S. Perkins. Accurate on-line support vector regression. *Neural Comput.*, 15(11):2683–2703, 2003.

[129] K.-L. Ma. Machine learning to boost the next generation of visualization technology. *Computer Graphics and Applications, IEEE*, 27(5):6–9, 2007.

[130] A. R. Martin and M. O. Ward. High dimensional brushing for interactive exploration of multivariate data. In *VIS '95: Proceedings of the 6th conference on Visualization '95*, page 271, Washington, DC, USA, 1995. IEEE Computer Society.

[131] K. Matkovic, W. Freiler, D. Gracanin, and H. Hauser. Comvis: A coordinated multiple views system for prototyping new visualization technology. *Information Visualisation, International Conference on*, 0:215–220, 2008.

[132] T. May, A. Bannach, J. Davey, T. Ruppert, and J. Kohlhammer. Guiding feature subset selection with an interactive visualization. In *Visual Analytics Science and Technology (VAST), 2011 IEEE Conference on*, pages 111–120. IEEE, 2011.

[133] T. May and J. Kohlhammer. Towards closing the analysis gap: Visual generation of decision supporting schemes from raw data. In *Computer Graphics Forum*, volume 27, pages 911–918. Wiley Online Library, 2008.

[134] T. M. Mitchell. *Machine learning*. McGraw Hill series in computer science. McGraw-Hill, 1997.

[135] A. Moere. Time-varying data visualization using information flocking boids. In *Information Visualization, 2004. INFOVIS 2004. IEEE Symposium on*, pages 97–104. IEEE, 2005.

[136] J. Nam and K. Mueller. Tripadvisorn-d: A tourism-inspired high-dimensional space exploration framework with overview and detail. *Visualization and Computer Graphics, IEEE Transactions on*, 19(2):291–305, 2013.

[137] T. C. G. A. Network. Comprehensive molecular portraits of human breast tumours. *Nature*, 490(7418):61–70, 2012.

[138] NeuroSynth. www.neurosynth.org, 2012.

[139] C. North. Toward measuring visualization insight. *Computer Graphics and Applications, IEEE*, 26(3):6–9, 2006.

[140] M. Novotný and H. Hauser. Outlier-preserving focus+context visualization in parallel coordinates. *Visualization and Computer Graphics, IEEE Transactions on*, 12(5):893–900, 2006.

[141] S. Oeltze, H. Doleisch, H. Hauser, P. Muigg, and B. Preim. Interactive visual analysis of perfusion data. *Visualization and Computer Graphics, IEEE Transactions on*, 13(6):1392 –1399, 2007.

[142] J. Osborne and A. Costello. Sample size and subject to item ratio in principal components analysis. *Practical assessment, research & evaluation*, 9(11):8, 2004.

[143] M. Pechenizkiy, S. Puuronen, and A. Tsymbal. The impact of sample reduction on pca-based feature extraction for supervised learning. In *Proceedings of the 2006 ACM symposium on Applied computing*, pages 553–558. ACM, 2006.

[144] A. Perer and B. Shneiderman. Integrating statistics and visualization for exploratory power: From long-term case studies to design guidelines. *Computer Graphics and Applications, IEEE*, 29(3):39 –51, 2009.

[145] H. Piringer, M. Buchetics, H. Hauser, and M. E. Gröller. Hierarchical difference scatterplots: Interactive visual analysis of data cubes. *ACM SIGKDD Explorations Newsletter*, 11(2):49–58, 2010.

[146] H. Piringer, C. Tominski, P. Muigg, and W. Berger. A multi-threading architecture to support interactive visual exploration. *Visualization and Computer Graphics, IEEE Transactions on*, 15(6):1113–1120, 2009.

[147] D. Pyle. *Data preparation for data mining.* Morgan Kaufmann, 1999.

[148] M. Rasmussen and G. Karypis. gCLUTO–An Interactive Clustering, Visualization, and Analysis System., University of Minnesota, Department of Computer Science and Engineering, CSE. Technical report, UMN Technical Report: TR, 2004.

[149] R. Ratcliff et al. Methods for dealing with reaction time outliers. *Psychological bulletin*, 114:510–510, 1993.

[150] C. Reimann, P. Filzmoser, and R. Garrett. Background and threshold: critical comparison of methods of determination. *Science of the Total Environment*, 346(1):1–16, 2005.

[151] R. Reitan and L. Davison. *Clinical neuropsychology: current status and applications.* Series in Clinical and Community Psychology. Winston, 1974.

[152] R. Rensink, J. O'Regan, and J. Clark. To see or not to see: The need for attention to perceive changes in scenes. *Psychological Science*, 8(5):368–373, 1997.

[153] P. Riehmann, M. Hanfler, and B. Froehlich. Interactive sankey diagrams. In *Proceedings of the IEEE Symposium on Information Visualization (InfoVis '05)*, pages 233–240. IEEE, 2005.

[154] S. Rinzivillo, D. Pedreschi, M. Nanni, F. Giannotti, N. Andrienko, and G. Andrienko. Visually driven analysis of movement data by progressive clustering. *Information Visualization*, 7(3):225–239, 2008.

[155] G. Robertson, R. Fernandez, D. Fisher, B. Lee, and J. Stasko. Effectiveness of animation in trend visualization. *Visualization and Computer Graphics, IEEE Transactions on*, 14(6):1325 –1332, 2008.

[156] R. Rosenbaum and H. Schumann. Progressive refinement: more than a means to overcome limited bandwidth. In *SPIE Proceedings of the Visualization and Data Analysis*, volume 7243, page 72430, 2009.

[157] P. Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

[158] J. Royston. An extension of shapiro and wilk's w test for normality to large samples. *Applied Statistics*, pages 115–124, 1982.

[159] O. Rubel, G. Weber, M.-Y. Huang, E. Bethel, M. Biggin, C. Fowlkes, C. Luengo Hendriks, S. Keranen, M. Eisen, D. Knowles, J. Malik, H. Hagen, and B. Hamann. Integrating data clustering and visualization for the analysis

of 3d gene expression data. *Computational Biology and Bioinformatics, IEEE/ACM Transactions on*, 7(1):64 –79, 2010.

[160] I. Ruts and P. Rousseeuw. Computing depth contours of bivariate point clouds. *Computational Statistics & Data Analysis*, 23(1):153–168, 1996.

[161] G. D. Ruxton. The unequal variance t-test is an underused alternative to student's t-test and the mann–whitney u test. *Behavioral Ecology*, 17(4):688–690, 2006.

[162] H. Samet. *Foundations of multidimensional and metric data structures.* Morgan Kaufmann, 2006.

[163] J. Scheffer. Dealing with missing data. *Research letters in the information and mathematical sciences*, 3(1):153–160, 2002.

[164] T. Schreck, J. Bernard, T. Tekusova, and J. Kohlhammer. Visual cluster analysis of trajectory data with interactive Kohonen Maps. In *IEEE Symposium on Visual Analytics Science and Technology, 2008. VAST'08*, pages 3–10, 2008.

[165] T. Schreck, T. von Landesberger, and S. Bremm. Techniques for precision-based visual analysis of projected data. *Information Visualization*, 9(3):181–193, 2010.

[166] A. Seborg and A. Singhal. Clustering multivariate time-series data. *Journal of chemometrics*, 19(8):427, 2005.

[167] J. Seo and B. Shneiderman. Interactively exploring hierarchical clustering results. *IEEE Computer*, 35(7):80–86, 2002.

[168] J. Seo and B. Shneiderman. A rank-by-feature framework for unsupervised multidimensional data exploration using low dimensional projections. In *Proc. IEEE Symposium on Information Visualization INFOVIS 2004*, pages 65–72, 2004.

[169] J. Sharko, G. Grinstein, and K. Marx. Vectorized radviz and its application to multiple cluster datasets. *IEEE transactions on Visualization and Computer Graphics*, pages 1444–1427, 2008.

[170] J. Sharko, G. Grinstein, K. Marx, J. Zhou, C.-H. Cheng, S. Odelberg, and H.-G. Simon. Heat map visualizations allow comparison of multiple clustering results and evaluation of dataset quality: Application to microarray data. In *Information Visualization, 2007. IV '07. 11th International Conference*, pages 521 –526, 2007.

[171] K. Shi, H. Theisel, H. Hauser, T. Weinkauf, K. Matkovic, H. Hege, and H. Seidel. Path line attributes-an information visualization approach to analyzing the dynamic behavior of 3d time-dependent flow fields. *Topology-Based Methods in Visualization II*, pages 75–88, 2009.

[172] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Visual Languages, 1996. Proceedings., IEEE Symposium on*, pages 336–343. IEEE, 1996.

[173] B. Shneiderman. The eyes have it: a task by data type taxonomy for information visualizations. In *Visual Languages, 1996. Proceedings., IEEE Symposium on*, pages 336 –343, 1996.

[174] B. Shneiderman. Inventing discovery tools: combining information visualization with data mining1. *Information Visualization*, 1(1):5–12, 2002.

[175] H. Siirtola. Interactive cluster analysis. In *IV '04: Proceedings of the Information Visualisation, Eighth International Conference*, pages 471–476, Washington, DC, USA, 2004. IEEE Computer Society.

[176] T. Sprenger, R. Brunella, and M. Gross. H-BLOB: a hierarchical visual clustering method using implicit surfaces. *Proceedings Visualization 2000.*, pages 61–68,, 2000.

[177] J. Stasko, C. Görg, and Z. Liu. Jigsaw: supporting investigative analysis through interactive visualization. *Information visualization*, 7(2):118–132, 2008.

[178] C. Stolte, D. Tang, and P. Hanrahan. Polaris: a system for query, analysis, and visualization of multidimensional relational databases. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):52–65, 2002.

[179] E. V. Sullivan, M. Rosenbloom, K. L. Serventi, and A. Pfefferbaum. Effects of age and sex on volumes of the thalamus, pons, and cortex. *Neurobiology of aging*, 25(2):185–192, 2004.

[180] Tableau Software. http://www.tableausoftware.com/, 2011.

[181] P.-N. Tan, M. Steinbach, and V. Kumar. *Introduction to Data Mining.* Addison-Wesley Longman Publishing Co., Inc., 2005.

[182] A. Tatu, G. Albuquerque, M. Eisemann, P. Bak, H. Theisel, M. Magnor, and D. Keim. Automated analytical methods to support visual exploration of high-dimensional data. *Visualization and Computer Graphics, IEEE Transactions on*, 17(5):584–597, 2011.

[183] TCGA. The cancer genome atlas. http://cancergenome.nih.gov, 2013.

[184] R. D. C. Team. *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria, 2009. ISBN 3-900051-07-0.

[185] A. Telea and D. Auber. Code flows: Visualizing structural evolution of source code. *Computer Graphics Forum*, 27(3):831–838, 2008.

[186] J. Thomas and P. C. Wong. Visual analytics. *IEEE Computer Graphics and Applications*, 24(5):0020–21, 2004.

[187] J. J. Thomas and K. A. Cook. *Illuminating the Path: The Research and Development Agenda for Visual Analytics.* National Visualization and Analytics Ctr, 2005.

[188] J. W. Tukey. *Exploratory Data Analysis.* Addison-Wesley, 1977.

[189] C. Turkay, P. Filzmoser, and H. Hauser. Brushing dimensions – a dual visual analysis model for high-dimensional data. *Visualization and Computer Graphics, IEEE Transactions on*, 17(12):2591 –2599, 2011.

[190] C. Turkay, A. Lundervold, A. Lundervold, and H. Hauser. Representative factor generation for the interactive visual analysis of high-dimensional data. *Visualization and Computer Graphics, IEEE Transactions on*, 18(12):2621–2630, 2012.

[191] C. Turkay, A. Lundervold, A. J. Lundervold, and H. Hauser. Representative factor generation for the interactive visual analysis of high-dimensional data. *Visualization and Computer Graphics, IEEE Transactions on*, 18(12):2621–2630, 2012.

[192] C. Turkay, J. Parulek, and H. Hauser. Dual analysis of dna microarrays. In *Proceedings of the 12th International Conference on Knowledge Management and Knowledge Technologies*, i-KNOW '12, pages 26:1–26:8, 2012.

[193] B. Tversky, J. B. Morrison, and M. Betrancourt. Animation: can it facilitate? *International journal of human-computer studies*, 57(4):247–262, 2002.

[194] L. van der Maaten and G. Hinton. Visualizing non-metric similarities in multiple maps. *Machine learning*, pages 1–23, 2011.

[195] T. Van Long and L. Linsen. MultiClusterTree: Interactive Visual Exploration of Hierarchical Clusters in Multidimensional Multivariate Data. In *Computer Graphics Forum*, volume 28, pages 823–830. John Wiley & Sons, 2009.

[196] J. J. van Wijk. The value of visualization. In *Visualization, 2005. VIS 05. IEEE*, pages 79–86. IEEE, 2005.

[197] J. J. van Wijk and W. A. A. Nuij. Smooth and efficient zooming and panning. In *Proceedings of the Ninth annual IEEE conference on Information visualization*, INFOVIS'03, pages 15–22, Washington, DC, USA, 2003. IEEE Computer Society.

[198] J. J. van Wijk and E. Van Selow. Cluster and calendar based visualization of time series data. In *infovis*, page 4. Published by the IEEE Computer Society, 1999.

[199] Y. Vardi and C. Zhang. The multivariate l1-median and associated data depth. *Proceedings of the National Academy of Sciences*, 97(4):1423–1426, 2000.

[200] K. B. Walhovd, A. M. Fjell, I. Reinvang, A. Lundervold, A. M. Dale, D. E. Eilertsen, B. T. Quinn, D. Salat, N. Makris, and B. Fischl. Effects of age on volumes of cortex, white matter and subcortical structures. *Neurobiology of aging*, 26(9):1261–1270, 2005.

[201] X. Wang, K. Smith, and R. Hyndman. Characteristic-based clustering for time series data. *Data Mining and Knowledge Discovery*, 13(3):335–364, 2006.

[202] M. O. Ward. Xmdvtool: integrating multiple methods for visualizing multivariate data. In *Proceedings of the conference on Visualization '94*, VIS '94, pages 326–333. IEEE Computer Society Press, 1994.

[203] C. Weaver. Cross-filtered views for multidimensional visual analysis. *IEEE Transactions on Visualization and Computer Graphics*, 16:192–204, 2010.

[204] M. Weber, M. Alexa, and W. Müller. Visualizing time-series on spirals. In *proceedings of the IEEE Symposium on Information Visualization*, page 7. Citeseer, 2001.

[205] D. Wechsler. Wechsler adult iintelligence scale-iii (wais-iii). *Psychological Corporation, San Antonio*, 1997.

[206] D. Wechsler. *Wechsler abbreviated scale of intelligence.* Psychological Corporation, 1999.

[207] M. Wijffelaars, R. Vliegen, J. J. van Wijk, and E. Van Der Linden. Generating color palettes using intuitive parameters. *Computer Graphics Forum*, 27(3):743–750, 2008.

[208] L. Wilkinson, A. Anand, and R. Grossman. Graph-theoretic scagnostics. In *Proceedings of the Proceedings of the 2005 IEEE Symposium on Information Visualization*, INFOVIS '05, pages 157–164, Washington, DC, USA, 2005. IEEE Computer Society.

[209] L. Wilkinson, A. Anand, and R. Grossman. High-dimensional visual analytics: Interactive exploration guided by pairwise views of point distributions. *Visualization and Computer Graphics, IEEE Transactions on*, 12(6):1363–1372, 2006.

[210] M. Williams and T. Munzner. Steerable, progressive multidimensional scaling. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 57–64, Washington, DC, USA, 2004. IEEE Computer Society.

[211] P. C. Wong and R. D. Bergeron. 30 years of multidimensional multivariate visualization. In *Scientific Visualization, Overviews, Methodologies, and Techniques*, pages 3–33, Washington, DC, USA, 1997. IEEE Computer Society.

[212] J. Yang, D. Hubball, M. Ward, E. Rundensteiner, and W. Ribarsky. Value and relation display: Interactive visual exploration of large data sets with hundreds of dimensions. *Visualization and Computer Graphics, IEEE Transactions on*, 13(3):494 –507, 2007.

[213] J. Yang, M. O. Ward, E. A. Rundensteiner, and S. Huang. Visual hierarchical dimension reduction for exploration of high dimensional datasets. In *VISSYM '03: Proceedings of the symposium on Data visualisation 2003*, pages 19–28. Eurographics Association, 2003.

[214] M. Ystad, T. Eichele, A. J. Lundervold, and A. Lundervold. Subcortical functional connectivity and verbal episodic memory in healthy elderly – a resting state fmri study. *NeuroImage*, 52(1):379 – 388, 2010.

[215] M. Ystad, A. Lundervold, E. Wehling, T. Espeseth, H. Rootwelt, L. Westlye, M. Andersson, S. Adolfsdottir, J. Geitung, A. Fjell, et al. Hippocampal volumes are important predictors for memory function in elderly women. *BMC medical imaging*, 9(1):17, 2009.