Werner Sturm

Discovering Medical Knowledge Using Visual Analytics

Master's Thesis

Graz University of Technology

Institute of Computer Graphics and Knowledge Visualization
Head: Wolf-Dietrich Fellner, Univ.-Prof. Dipl.-Ing. Dr.techn.

Supervisor: Torsten Ullrich, Dipl.-Math. Dr.techn.

Graz, May 2015

This document is set in Palatino, compiled with pdfLaTeX2e.

The LaTeX template from Karl Voit is based on KOMA script and can be found online: https://github.com/novoid/LaTeX-KOMA-template

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

Graz, ________________________ ____________________________

Date Signature

Eidesstattliche Erklärung[1]

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Graz, am ________________________ ____________________________

Datum Unterschrift

[1] Beschluss der Curricula-Kommission für Bachelor-, Master- und Diplomstudien vom 10.11.2008; Genehmigung des Senates am 1.12.2008

Abstract

Due to advanced technologies, the amount of biomedical data has been increasing drastically. Such large data sets might be obtained from hospitals, medical practices or laboratories and can be used to discover unknown knowledge and to find and reflect hypotheses. Based on this fact, knowledge discovery systems can support experts to make further decisions, explore the data or to predict future events.

To analyze and communicate such a vast amount of information to the user, advanced techniques such as knowledge discovery and information visualization are necessary. Visual analytics combines these fields and supports users to integrate domain knowledge into the knowledge discovery process.

This master's thesis reviews and categorizes state-of-the-art approaches of knowledge discovery and visual analytics in general and for the biomedical domain. It also reviews the novel biomedical approach of systems biology which makes use of "omics" data to analyze biological properties of genomes, proteins and metabolites to understand biological and pathological processes. There is still a lack of biomedical visual analytics systems which tightly integrate the user into the knowledge discovery process. Moreover, a performed state-of-the-art analysis revealed that most common visualization techniques for multivariate data are scatter plots, Parallel Coordinates and heat maps.

Last but not least, an implementation of a clustered heat map is presented to discuss a practical application of visual analytics.

Zusammenfassung

Die Menge von medizinischen Daten hat sich in den letzten Jahren drastisch erhöht. Datenmengen in größerem Ausmaß werden an unterschiedlichen Orten generiert. Darunter zählen unter anderem Krankenhäuser, Arztpraxen sowie biomedizinische Labore. Solch große Datenmengen, auch "big data" genannt, können als Wissensdatenbank genutzt werden, um Hypothesen zu finden bzw. zu überprüfen. Automatisierte Systeme anlysieren die Datenmenge und unterstützen Experten um Entscheidungen zu treffen, Daten zu erkunden bzw. Vorhersagen zu treffen.

Um große Informationsmengen zu analysieren, sowie verständlich für den/die BenutzerIn darzustellen, müssen spezielle Techniken wie Knowledge Discovery, Datenvisualisierung und Mensch-Maschine-Interaktion angewandt werden. Visual Analytics vereint diese Bereiche und im besten Fall ermöglicht es ExpertInnen das eigene Domänenwissen in den Analyseprozess zu integrieren.

Diese Arbeit untersucht state-of-the-art Methoden, welche Visual Analytics und Knowledge Discovery für biomedizinische Zwecke umsetzt. Aktuelle wissenschaftliche Arbeiten wurden analysiert und kategorisiert. Darüberhinaus wurde der Ansatz von Systems Biology untersucht, welcher alle "omics"-Daten (genomics, proteomics, metabolomics) vereint. Dies ermöglicht pathologische Prozesse zu analysieren und als Ganzes zu verstehen. Die Analyse hat ergeben, dass noch ein Mangel an tiefer Benutzerintegration herrscht und dass die meist angewandten Visualisierungsmethoden für multivariate Daten Scatter Plots, Parallel Coordinates und Heatmaps sind.

Anschließend wird eine Implementierung einer geclusterten Heatmap diskutiert, um einen praktischen Einsatz von Visual Analytics aufzuzeigen.

Acknowledgements

I would like to thank my supervisor Dipl.-Math. Dr.techn. Torsten Ullrich for his continuous guidance and support.

I also would like to express my gratitude to my family, especially to my parents, Hermine Kindermann and Josef Sturm, who have unconditionally supported me during my whole study.

Moreover, special thanks go to all people at the Institute of Computer Graphics and Knowledge Visualization for providing a very comfortable working environment, helpful tools and unlimited coffee.

Last but not least, I would like to thank my friends – especially Georg Reiter – who have supported me to proof-read my master's thesis.

Contents

1 Introduction

The amount of data has been increasing rapidly in various domains which leads to large data sets – so-called "Big Data" [58]. This recent trend can also be observed in medical and biomedical databases. Hospitals and medical practices collect information about every patient (patient records) including data from laboratories. Besides tabular data (e.g., measured values), doctors and physicians input various types of data such as natural text and images. As the amount of contained information is steadily increasing, corresponding databases are also becoming of greater value as such large collections of data can also be used as knowledge bases. Medical research appreciates new knowledge in this field, but it can also be used to improve clinical decision making for patient treatment, predict therapy outcomes and to find biomarkers indicating possible diseases.

As all results and decisions are based on empirical data, this approach is also called evidence-based medicine [20]. Its fundamental idea, which was called "medical arithmetic" dates back to the 18th century introduced by British physicians [143, 144]. In contrast to subjective "clinical judgment", evidence-based medicine optimizes decision-making by using knowledge gained from performed research.

Therefore, such knowledge needs to be made accessible to experts by a dedicated decision support system (DSS) [134]. This approach is also related closely to precision medicine (P4 Medicine: Predictive, Preventive, Participatory, Personalized) which aims the improvement of personalized patient care [105, 56].

While it is already a challenge for digital systems to organize and interlink such heterogeneous data, it is infeasible for humans to manually understand and analyze the entire data set as a whole. Therefore, digital systems are used to analyze large data sets automatically (knowledge discovery). Such knowledge discovery systems apply data mining techniques to find patterns and relations within the data set and present results to the user.

As mentioned in Section 2, data mining uses various methods of machine learning and statistics to generate a model for describing and approximating relationships and patterns within the observed data set. It is one of the key technologies in the knowledge discovery process and such data might be weakly structured, high dimensional and it is likely to hold sensitive information. Thus, prior steps such as data cleansing, data pre-processing are necessary to ensure results of higher quality [53]. Sophisticated techniques such as anonymization and pseudonymization must be applied to protect privacy [19, 79].

While the amount of the data is too large to be comprehended by humans, it is essential to make the data and discovered patterns more comprehensible and explorable. To transfer complex information and knowledge from the knowledge discovery system to the user, a tight interconnection between these two counterparts has to be established. For that, the research fields *human-computer interaction* and especially *information visualization* are fundamental.

As the name of *human-computer interaction* reveals, it is a field of research to analyze and optimize the interaction between humans and computer systems. It includes the design, implementation and evaluation of computer systems to enhance usability, efficiency, effectiveness and satisfaction of the user.

This thesis primarily focuses on *information visualization* combined with knowledge discovery and user involvement within the medical domain. This combination of analytical reasoning, visual data representations and user interactions is also called *visual analytics* [77]. As visual analytics systems integrate the user into the analytical reasoning process, such systems are usually not fully automated. This allows experts to steer the knowledge discovery process and to exert influence on outcome of the analysis with their individual domain knowledge.

Nowadays, many visual analytics systems focus on a particular data type to perform further analysis. A even more sophisticated tool interconnects and analyzes several related data sets at once to extend the analyzed amount of information. Concerning this approach, a relatively novel biomedical approach is called *systems biology*. As explained in Section 6, it aims modeling biological relationships and interactions between several data sets which contain information about proteins (proteomics), genes (genomics) and metabolites (metabolomics) [42]. Such data types are commonly called "omics" data and it can be used to analyze biological and pathogenic processes (e.g., finding biomarkers).

Therefore, the ultimate goal of bioinformatics is to combine several databases and heterogeneous data types into a single system to get insight into the whole biological processes (e.g., of the human body). Such a system might link x-ray scans to tissue samples and molecular data to support biologists and doctors to investigate several layers of the biological system at once.

In addition to that, Section 7 contains an analysis of recent visual analytics approaches based on the state-of-the-art review of Turkay et al. [147]. It reviews current research trends of visual analytics approaches of biomedical data. Integration level, visualization methods, data type and the class of analysis has been considered.

The performed analysis revealed, that heat maps are one of the most popular visualization techniques used for multivariate data. Section 8 discusses an implementation of a clustered heat map for several visual analytics tasks on molecular data sets for the Open Source project Scaffold Hunter[2]. The implementation supports the user to investigate relations between attributes of molecules by offering several interaction methods. Moreover, Scaffold Hunter provides several visualization methods to enhance the users insight into the data set.

[2] Scaffold Hunter - `http://scaffoldhunter.sourceforge.net/`

2 Data Mining and Knowledge Discovery

The aim of data mining is the extraction of unknown patters out of raw and complex data to gain new useful knowledge by applying specific data analysis and discovery algorithms. As data mining is one of the core steps in the KDD process, pre-steps such as data cleansing and data preprocessing are necessary to ensure data mining results of higher quality. In addition to that, prior knowledge about the domain is essential to make right interpretations of the result. An unexperienced application of data mining might lead to misinterpretations, unimportant and meaningless patterns [35].

Basically, data mining tasks are classified into *verification* and *discovery*. The aim of verification tasks is to check, whether a specific hypothesis is correct. Data mining tasks, which are used by discovery systems, aims to find new patterns autonomously. This type of task can be subdivided into *prediction* and *description*. Predictions intend to predict future behaviors based on the extracted knowledge from the present data set. Basically, this can be understood as a performed interpolation by using an approximated prediction model. The purpose of descriptive tasks is to present the extracted knowledge in a human-understandable way. Therefore, description models of lower complexity are preferable while the internal complexity of prediction models is irrelevant [35].

2.1 Knowledge Discovery in Databases

Knowledge discovery in Databases (KDD) is a hot topic because the amount of data has been increasing much quicker than the improvement of analyzing methods to extract knowledge out of the data. KDD is often being equated with data mining, but it includes more than just this field. While data mining is one of the core elements of KDD, it also targets the management of data (databases), the preparation of the data and selection of algorithms to perform data mining and the representation and interpretation of the final result.

Data mining is used to discover new knowledge for prediction and understanding by finding patterns within the data.

Fayyad et al. [37] mentions:

> “Knowledge Discovery is the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data”

To evaluate a pattern, interestingness and usefulness for a given task need to be considered. The discovered pattern should be valid for new data sets as well. The understandability of the found pattern is very important if the final goal of the analysis is more about understanding the data itself than pure prediction [36].

2.1.1 KDD Process

As depicted in 1, the KDD process gives a global overview of all needed steps of knowledge discovery in databases [36]:

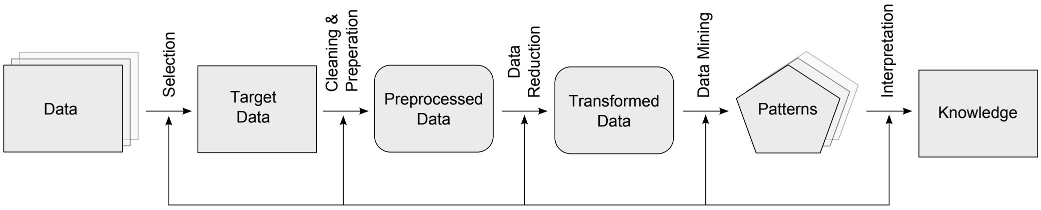


Figure 1: The simplified iterative KDD process depicts how new knowledge can be extracted from multiple data sources [36].

Domain Knowledge This step includes understanding of the domain by gathering necessary state-of-the-art information and defining a final goal of the process.

Target Data set Creating a data set by acquainting data from several sources and unify values. Moreover, making a selection of data and variables which should be used in the further process.

Data Cleaning and Preparation In general, large data sets noisy, inconsistent and might come from heterogeneous sources, cleansing of the data is essential. The quality of a performed knowledge discovery is directly dependent on the quality of the underlying data set [48]. Cleaning includes handling missing values, removing outliers, smoothing noise and resolving inconsistency. Missing values can be handled in different ways like simple removing of the data record, manual filling or using a constant (e.g., mean value) [48]. Removing the whole record might lead to a too small data set for further research, so the usage of surrogate values is preferable.

Data cleaning is an essential element of data mining but experts have to be aware that each manipulation of the data set might lead to a different result and interpretation of the data. Therefore, the final finding might deviate even more from the real model.

Data Reduction The data can be reduced by dimensionality reduction such as *Principle Component Analysis* (PCA) [167], *Multi-Dimensional Scaling* (MDS) [22] and *Independent Component Analysis* (ICA) [63]. These methods reduce the complexity by projecting the high dimensional data to a lower one while preserving the variance of the data [147]. Furthermore, additional approaches to reduce the number of variables are specific transformation methods and the assortment of features which represent the data set best. See Section 2.4 for more detailed information.

2.1.2 Function of Data Mining

This Section includes the selection of a data mining function (classification, clustering, regression, summarization) and its subsequent application. See Section 2.2.2 for further details.

Basic components of data mining are [36]:

- Model
  The model's function (e.g., clustering, classification)
  Representation of the model (e.g., density function,)
- Preference criteria (usage of goodness-of-fit function to select preferred parameters)

- Search algorithm (the algorithm for finding specific models and parameters, given a model, preference criteria and data)

Data Mining Algorithm A data mining algorithm defines which method should be used to find patterns. Depending on the type of data mining problem (e.g., clustering, classification, regression, etc.), an algorithm from the corresponding class has to be chosen (see Section 2.2.2). In addition to that, involved data types have to be considered as data mining algorithms can only handle a specific set of data types (e.g., numerical, nominal). See Section 2.1.2 for further details.

The resulting model type (e.g., decision tree, neuronal network) is also defined by the selected algorithm. In case that prediction is the overall goal, more complex and less interpretable models, which generally fit data more precisely, are usually preferred.

Data Mining This step performs the actual search for patterns within the data set by applying the selected data mining algorithm. Depending on the size of the data set, the application of the data mining algorithm might take a significant amount of time. Therefore, real-time calculations are not feasible in many cases and pre-calculations need to be performed instead.

The found model depends on the executed algorithm and it is an empirical approximation of the real theoretical model. Therefore, the quality of the trained model directly depends on the quality of the training data set. As mentioned above, data pre-processing also greatly influences the generated model and this has to be considered.

Interpretation The interpretation of the result can be enhanced significantly by specific visualization methods which transforms the pattern in a more comprehensible representation for users. Unimportant patterns might be removed and in case of a total unsatisfiable result, re-execution of the data ming algorithm with changed parameters or even choosing a different data mining method might be necessary. This means, that the KDD process is an iterative approach to find an appropriate method to discover new knowledge.

Using Discovered Knowledge and Provenance The usage of discovered knowledge includes taking further actions based on the knowledge (prediction support), reporting and documentation or finding conflicts with former knowledge. In addition to that, the expert needs to be able to trace which data has influenced the final decision (provenance). Due to the continuous addition of data to the database, a past decision is usually based on a different databases. Thus, changes of the data have to be documented as well to keep decisions traceable [164].

2.1.3 Data Types and Structures

Data from multiple databases is likely to be heterogeneous and individual data-mining algorithms are usually designed to analyze a specific data format or type. In general, data can be continuous, discrete or a more complex type such as text or multimedia data (e.g., photos, body scans, x-ray data, etc.). Therefore, individual solutions for different data types are often necessary.

Continuous data contains numerical values with an implicit order. Discrete data types usually represent categories. Such categories might have an ordering (so-called ordinary variables. e.g., "low", "middle", "high") or none (nominal variables: e.g., "animal", "plant", "human"). A common and specific type of discrete data is a binary variable (e.g., "yes" and "no").

Complex data structures such as graphs, text or even multimedia data require tailored algorithms to extract valuable information out of the given data structure, as common data-mining algorithms (e.g, k-means, decision trees, etc.) are designed to work on tabular data.

2.2 Human Computer Interaction and KDD

A novel approach is Human Computer Interaction and KDD is combining and emphasizing the research fields human computer interaction (HCI) and Knowledge discovery in databases (KDD). The ultimate goal of this approach is to enhance human intelligence by computational power and intelligence [52].

2.2.1 Human Computer Interaction

Human-computer-interaction is an essential research field to enhance visualizations. HCI investigates the interaction between people and computers for a specific task. It includes the design, implementation and evaluation to improve the system's usability implying its efficiency, effectiveness and satisfaction. Thus, HCI cooperates with multiple other research areas such as psychology, computing, behavior and ergonomics. In this context, efficiency describes the amount of resources the user needs to achieve a specific task with (e.g., time, mental effort). Effectiveness defines the accuracy the user achieves the goal with. Satisfaction represents how well the user felt during working with the system [29]. Therefore, to improve user experience for exploring the data, considering HCI is essential.

2.2.2 Combining HCI and KDD

The combination of HCI and KDD aims to delegate final decisions and judgments to the human intelligence while computers search for potentially useful and valid patterns within the data set (see KDD process 2.1.1). HCI focuses on the abilities of the human capacities which can be used to empower users to explore the data in a highly interactive and efficient way. In other words, HCI-KDD tackles this slogan accurately [132]:

> "Computers are incredibly fast, accurate, and stupid. Human beings are incredibly slow, inaccurate, and brilliant. Together they are powerful beyond imagination."

2.3 Classes of Data Mining

According to Fayyad et al., data mining tasks can be classified into six different types [35], namely *clustering*, *classification*, *association rule mining*, *regression* and *summarization*. Mostly, these techniques are derived or re-used from various research fields (e.g., machine learning, statistics and pattern recognition).

2.3.1 Clustering

Clustering algorithms assign every data item to one class of a predefined set of classes to describe the data. In other words, such algorithms determine a set of categories or clusters to distinguish and to heap together data points. Depending on the algorithm, clusters can be mutually exhaustive, hierarchical or overlapping [35]. *k-means*, *hierarchical clustering* or *clique* are just a few examples of clustering algorithms. Basically, clustering algorithms need a similarity and dissimilarity function, also known as distance function, to distinguish data points. Examples of distance functions are *Euclidean distance* or *Minkowski distance* [168].

2.3.2 Classification

Classification is about learning a function (classifier) which assigns new data items into one of the predefined classes. The decision is based on the learned knowledge from a labeled past data set. Thus, classification algorithms are trained by supervised learning techniques. There exist many applications of classification in various domains. Basically, algorithms are subdivided into binary classifications (positive and negative outcome) and multi class classifications [5]. Some examples of commonly accepted techniques are *Neural Networks* [44], *Naive Bayes Classifier* [123], *Decision Trees* [126], *K-nearest Neighbor* [21] and *Support Vector Machines* [50].

2.3.3 Association Rule Mining

Association rule mining (also known as Dependency modeling) intends to find a model which represents major dependencies between variables in large databases. Two levels of dependency models can be distinguished: the *structural* model shows local dependencies of variables while *quantitative* models describe the strength of dependency as a numerical value [35, 93].

2.3.4 Regression

Regression involves the search of a linear and higher dimensional function, which approximates the given data with a minimal distance error (e.g., mean square error). A so-called regression function models the relation between one or several predictor variables (multiple regression) and a single dependent response variable. Regressions are usually used for prediction tasks. However, a low-dimensional regression function can also represent the dependency in a human-understandable way (e.g, plot) [35, 5].

2.3.5 Summarization

Summarization aims to find a short description of the data which is commonly used for interactive exploratory data analysis and report generations [35]. Chandola et al. describe summarization as follows [17]:

> "Summarization is a key data mining concept which involves techniques for finding a compact description of a dataset. Simple summarization methods such as tabulating the mean and standard deviations are often applied for data analysis, data visualization and automated report generation."

For summarization, various values can be representative while preserving the most information. For example the centroid of a cluster of documents is a good representative of all items within the cluster. Another summarization approach uses aggregation functions (calculation of maximum, average, etc.) [4].

2.3.6 Sequential Patterns

The search for sequential patterns aims to find trends or to analyze the process generating patterns in time-dependent data sets [36].

2.4 Dimensionality Reduction

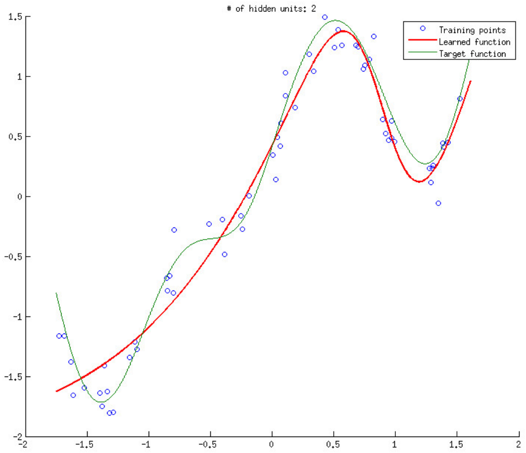
Data sets containing several hundred variables are very common. Unfortunately, such high dimensional spaces are not interpretable by humans and commonly accepted visualizations for multivariate data (e.g., scatter plot matrix, Parallel Coordinates) are not comprehensible when using more than 20 variables [68]. The reduction of the data sets dimensionality (variables) is an approach to proceed with a lower amount of variables while preserving as much information as possible. Every reduction leads to a loss of potentially valuable information contained in variables and hidden structures. Thus, as structures are based on a subset of variables, the amount of loss information does not only depend on the count of removed variables. In general, there exist two approaches for dimensionality reduction, namely *feature selection* and *feature extraction* [117].

Feature selection is generally about selecting the best subset of all features for a given task by using filters or by applying optimization methods. Feature extraction transforms the original data to a lower dimensional space while preserving as much information as possible. Common techniques are *Principle Component Analysis* (PCA) which preserves the data variance, *Multi Dimensional Scaling* (MDS), preserving dissimilarities within the data, and *Self Organizing Maps* (SOM) which aims to keep topological and metric relationships. As a result, feature transformations might make relations to the original data set not intuitive. Therefore, another approach are variable groupings which group similar variables or choose a representative one for each group. *Principle Component Variable Grouping* is an example which uses calculated principle components to group features [117].

To integrate the user into the process, there exist interactive feature reduction systems such as *Visual Hierarchical Dimension Reduction* (VHDR) [170]. This approach constructs a hierarchy based on similarities between variable pairs.

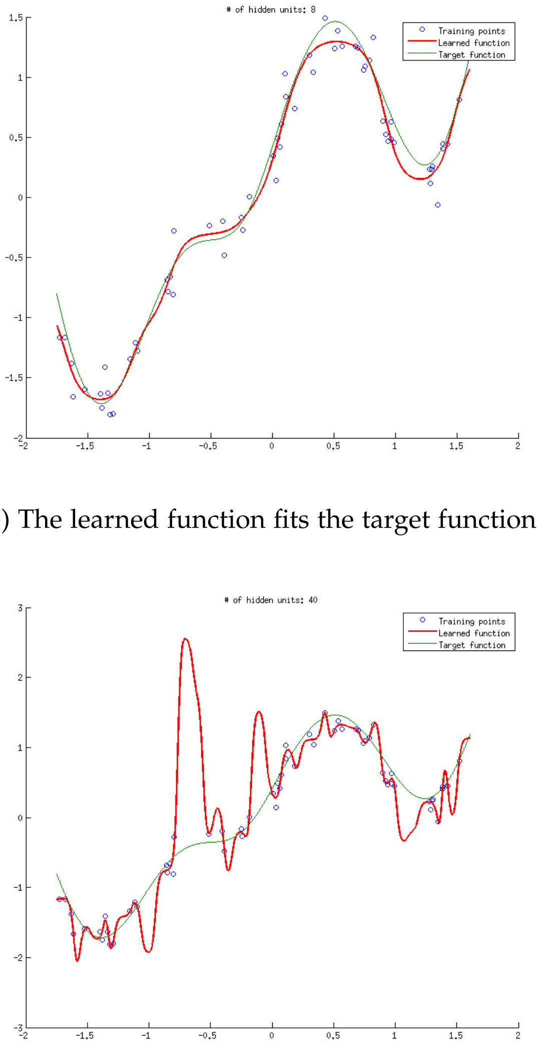
2.5 Challenges

Data mining faces several challenges which have to be considered by data analysts [35]: Large databases might contain several hundred fields and tables and millions of records. Thus, efficient algorithms are essential. Data of high dimensionality increases the search space dramatically, which leads to a higher chance of finding invalid patterns as well. This problem can be coped with selecting a subset of variables, based on prior knowledge, or with performing dimensionality reduction (see Section 2.4). Moreover, underfitting and overfitting of the learned model can also lead to a poor prediction quality (see Figure 2).



(a) The learned function underfits the target function.

An overfitted model might not fit the general data, since the learned model might be too complex (see Figure 2c) or it might fit noisy data points which results in a low quality when applying the model to new data sets. Therefore, missing and noisy data is a common challenge. Data might also change over time due to updates. Discovered patterns might not be valid after a significant change and new ones might appear. In such a case, an incremental update of patterns is needed. Many data mining algorithms are designed for attribute-value records, but databases might also contain more complex structures such as hierarchies or graphs [35]. For that, specialized algorithms are needed. For descriptive tasks, discovered patterns need to be understandable and data mining algorithms should be more interactive to allow the user to cooperate according to prior knowledge.



(b) The learned function fits the target function.

(c) The learned function overfits the target function.

Figure 2: Three trained functions learned by neural networks with different count of nodes. The usage of two hidden units leads to an underfitted trained function – the target function is too complex (a). The learned function fits the target function best with 8 hidden units (b). A neural network with 40 hidden units results to a heavily overfitted learned function – the neural network is too complex (c).

2.6 Selected Data Mining Techniques

2.6.1 Decision Trees

A Decisions Tree (DT) is a popular data-mining technique for prediction and to classify data records. In general, a DT is a directed-acyclic graph (tree) representing a collection of nested conditions. Each intermediate tree node (and root node) holds a condition which usually considers a single attribute. The condition aims to split the data into subsets while considering the target class. In other words, starting at the root and following appropriate branches of conditions leads to a final leaf node which represents a class. DTs can handle both numerical and categorical attributes to predict a class. While decision trees are known for their high classification performance, they can also be interpreted by humans easier in contrast to other classification techniques (see Figure 7). A tree-like presentation is understandable and intuitive [92]. To generate a new DT, learning algorithms need to find attributes which distinguish the classes the most. This approach is being applied to the resulting subsets in a recursive manner. To avoid learning noise (so-called overfitting), a stop criteria is needed.

2.6.2 Self-Organizing Map

A Self-Organizing Map (SOM) is a feedforward neuronal network, which is used to perform unsupervised clustering of numerical data. SOMs were developed by T. Kohonen in 1982 (thus, a SOM is also called Kohonen Map). In comparison to other neuronal networks, SOM do not have hidden layers. Nodes of the output layer have additional neighborhood relations defined by a link and a distance [83].

Generally, output nodes are arranged in a grid where the shape of a cell depends on the defined topology of neighborhood (e.g., rectangle, hexagon). As depicted in Figure 3, each input node is dedicated to one dimension and each input node is connected to all output nodes (completely connected neuronal network).

Moreover, a spatial location within the input space is assigned to each node and therefore, each cell covers a specific area in the input space (see Figure

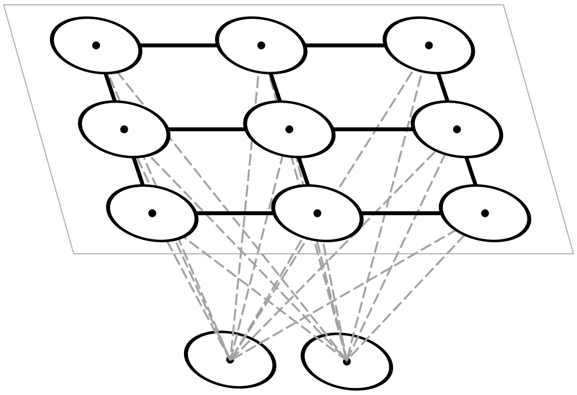


Figure 3: Architecture of a Self-Organizing Map of a rectangular neighborhood topology with 2 input and 9 output nodes.

4. In contrast to feedback networks, neighboring cells compete and adapt to a given input (self-organizing). Thus, cells can be used as a two-dimensional decoder for an arbitrary high-dimensional input. In other words, a SOM maps a high-dimensional input space to a 2D space.

While training, nodes and its neighbors (cells) adapt by moving their spatial location closer to the input. As a result, greater distances between cells can be interpreted as cluster boundaries.

2.7 Data Mining Tools

There exist several data mining tools which vary in ease-of-use and complexity. According to the user survey of Rexer Analysis [121], both open source and commercial products are used in common. Some of the most popular tools are STATISTICA, KNIME, R, SAS JMP, Rapid Miner, Weka and IBM SPSS Modeler, while the open source tool R has become the most-used since 2010. R is usually being used in combination with other tools. However, the usage of R as primary tool has also increased recently. In addition to that, STATISTICA, IBM SPSS Modeler, SAS JMP and KNIME

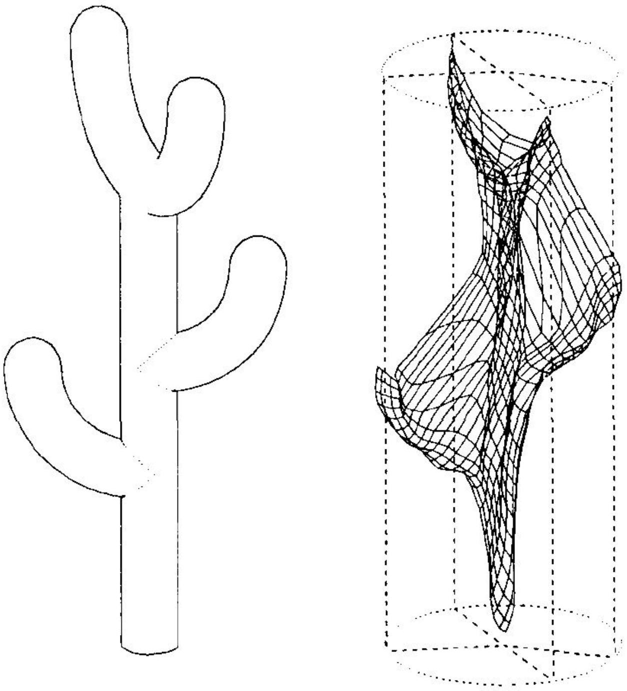


Figure 4: A two-dimensional SOM (right side) has been trained with a three-dimensional (uniform) density function as depicted on the left side. (Image source: Kohonen [83])

received high satisfaction ratings. The complete list can be found in data miner survey performed by Rexer Analysis in 2013 [121].

R

"R is a language and environment for statistical computing and graphics[3]. It is a GNU project which is similar to the S language and environment which was developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues. R can be considered as a different implementation of S. There are some important differences, but much code written for S runs unaltered under R." [1]

[3] The R Project for Statistical Computing - `http://www.r-project.org/`

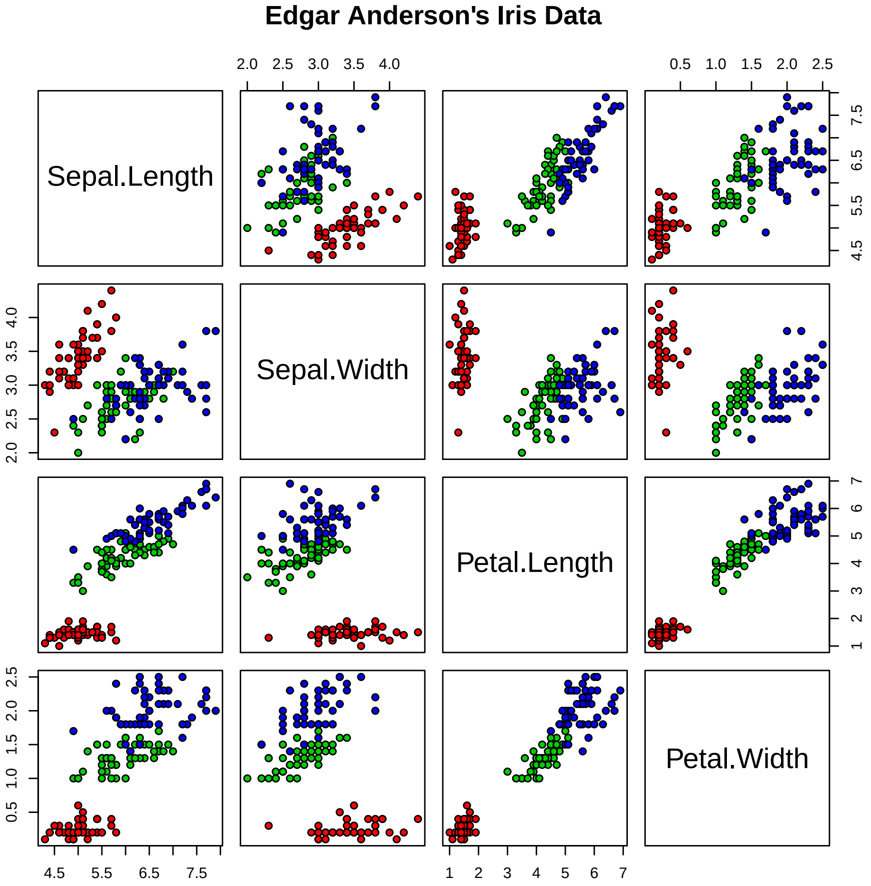


Figure 5: Scatter Plot Matrix depicting all dimension of the iris data set [39] (red=setosa, green=versicolor, blue=virginica) – generated with the command *pairs(x, . . .)* provided by the R environment.

The open source tool R offers the user freedom to modify, combine or extend given functionalities by writing own code using the programming language S. The R environment provides functions for data manipulation, performing calculations and generating graphical representations to analyze the data (see Figure 5). Data manipulations imply import and export functionalities and operators to perform calculations on arrays and matrices without the need of explicit loops. The environment can be understood as a coherent collection of functions to perform various statistical analysis or data mining. Moreover, R can be extended via packages or even by linking algorithms written in Fortran, C or C++.

STATISTICA

STATISTICA[4] is a suite of analytical software, which is split up into multiple products and developed by StatSoft since 1984 [136]. It offers functionalities to perform data analysis and management, data mining and statistics, and data visualization. According to Rexer Analysis, users report an outstanding overall satisfaction regarding the usage of STATISTICA. After R, it is the second most preferred primary tool for data analysis [121]. Fortunately, STATISTICA supports the integration of R to combine both systems.

Rapid Miner

Rapid Miner[5] is a commercial product (developed by the same-named company Rapid Miner) based on the open source software called YALE (Yet Another Learning Environment). It is implemented in Java and commonly used for industrial, educational and research purposes.

Moreover, it is designed to perform data mining experiments visually without the need of writing additional code and it supports both local and cloud based calculations (see Figure 6 and Figure 7). The product Rapid Miner Studio is free to use with several limitations (memory, supported data sources, etc.). For different limitation levels including the product Rapid Miner Server, annual fees are charged. However, Rapid Miner has shown significant increase of popularity in recent years and according to the survey of Rexer Analysis in the year 2013, more data analysts use it as their primary analytic tool [121].

[4] STATISTICA - `http://www.statsoft.com/`

[5] Rapid Miner - `https://rapidminer.com/`

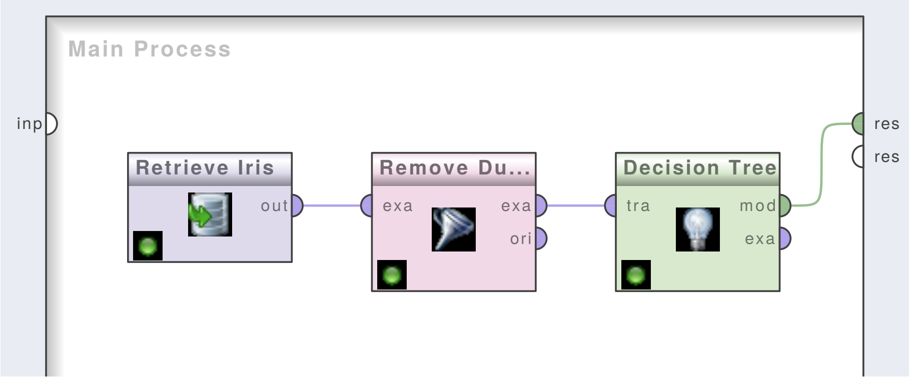


Figure 6: Graphical representation of the data mining process. The first step loads the iris data set (data access). The second step filters duplicates (data pre-processing). The last step trains a decision tree with the given data (data mining). The final result of the process is a trained decision tree model which can be applied to other data sets.

SAS

The Statistical Analysis System[6] (SAS), developed by SAS Institute, aims to enhance users to perform advanced and predictive analytics. The suite consists of more than 200 software components and it is one of the most popular tools with a high satisfaction rate. 9% of interviewed data analysts report SAS to be their primary analytics tool [121]. It offers both a graphical interface for standard users and a dedicated programming language (SAS programming language) to access more advanced functions and freedom.

KNIME

Similar to several other graphical data mining tools, the open platform KNIME[7] (the Konstanz Information Miner) supports users to design a modular data pipeline in a graphical manner. Such a pipeline consists of elements which are interconnected to each other to define a data flow. Each element (pipeline step) defines a specific task of the overall process. For example,

[6] SAS - http://www.sas.com/en_us/software/analytics/stat.html
[7] KNIME - https://www.knime.org/

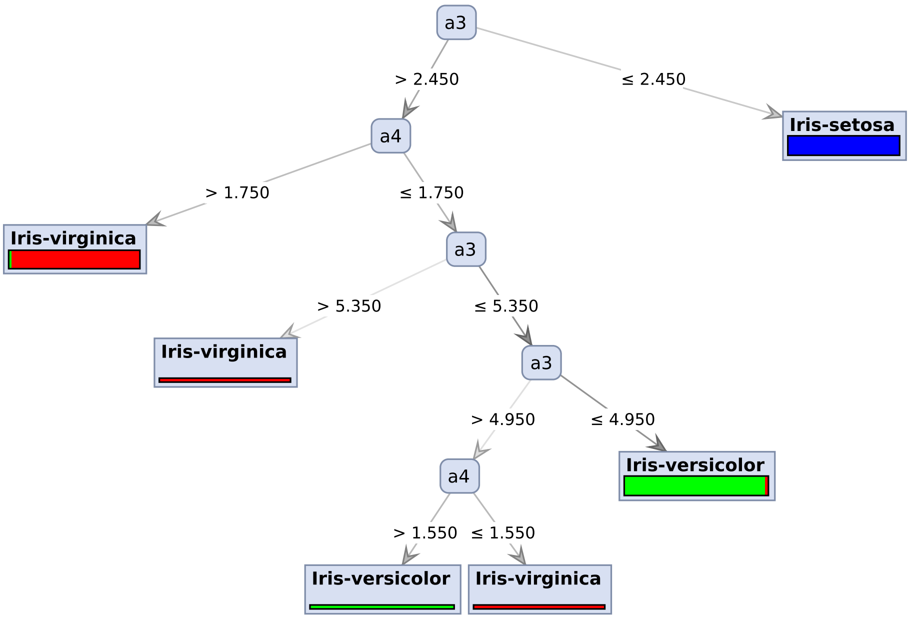


Figure 7: Rapid Miner enables users to design data mining processes in a graphical manner. Some trained models such as Decision Trees or Neuronal Networks can be visualized and investigated. The depicted exemplary Decision Tree, which is trained by using the iris data set, classifies three different species of Iris flowers.

such steps can be loading of data from a source, data pre-processing (e.g., removing duplicates, filling missing values, etc.), training a model, applying a trained model to data, calculating statistics or a visualization of the data or result (see Figure 8).

The environment is developed and supported by KNIME.com AG and aims primarily teaching and research purposes. It is written in Java and it is realized as an Eclipse plug-in (Eclipse Foundation). It can be easily extended by modules which implement new algorithms, visualization or other steps of the data pipeline [10]. Moreover, KNIME includes many extensions to enable the environment to include modules from other tools (e.g., Weka, R, Matlab, etc.).

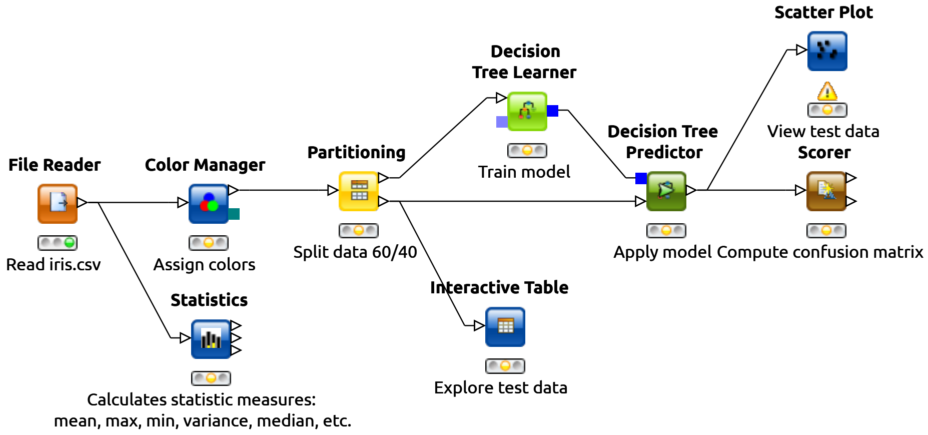


Figure 8: An example workflow of KNIME. It loads the iris data set, assigns individual colors to each species, splits the data set, trains a Decision Tree and applies the other data to the trained model. A Scatter Plot- and Scorer module show the final results and statistics.

IBM SPSS Modeler

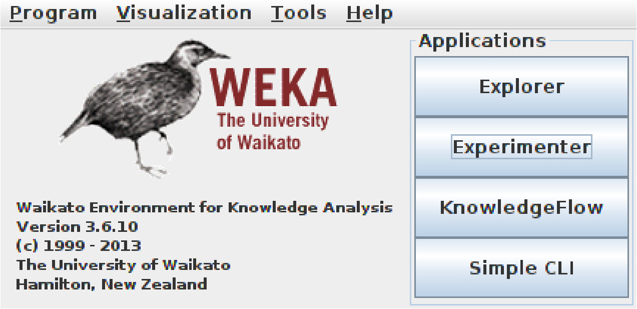
The SPSS Modeler is a proprietary software for data analysis developed by IBM[8]. Users can design data-mining processes in a graphical way as explained in 2.7. Moreover, the SPSS Modeler offers a generic model learner which tries to find the best model automatically. This might be a fast and convenient solution to obtain first results.

WEKA

The Waikato Environment for Knowledge Analysis[9] (WEKA) is a popular data mining tool and it offers a broad collection of features and algorithms. It is developed at the University of Waikato, New Zealand and published under the GNU General Public License [47].

[8] IBM SPSS Modeler - http://www-03.ibm.com/software/products/en/spss-modeler

[9] WEKA - http://www.cs.waikato.ac.nz/ml/weka/

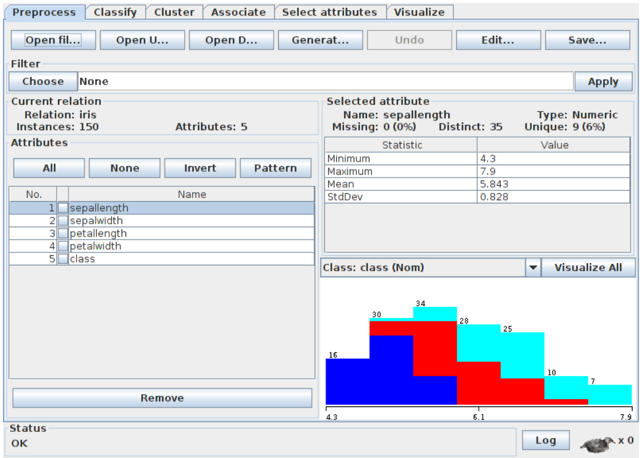


Waikato Environment for Knowledge Analysis
Version 3.6.10
(c) 1999 - 2013
The University of Waikato
Hamilton, New Zealand

Figure 9: The ”WEKA GUI Chooser” which is the initial user interface when starting WEKA.

WEKA offers a solid collection of data processing and machine learning algorithms. As depicted in Figure 9, the start window offers the user multiple modes to access these features. The “Explorer” is the main window. It consists of several panels which represent a single data-mining steps (see Figure 10 and Figure 11). The “Experimenter” supports users to execute and compare several experiments in a more convenient way. “KnowledgeFlow” offers the user to design and execute a graphically represented data-mining pipeline. Very similar to the approach of KNIME described in 2.7, algorithms and tasks can be connected to set up an individual pipeline. Last but not least, “Simple CLI” provides a command line to access features without the usage of the graphical user interface.

The Data Miner Survey 2013 of Rexer Analysis reports a high popularity of WEKA as secondary analytical tool [121].



Filter: None

Current relation
Relation: iris
Instances: 150 Attributes: 5

Selected attribute
Name: sepallength Type: Numeric
Missing: 0 (0%) Distinct: 35 Unique: 9 (6%)

Statistic	Value
Minimum	4.3
Maximum	7.9
Mean	5.843
StdDev	0.828

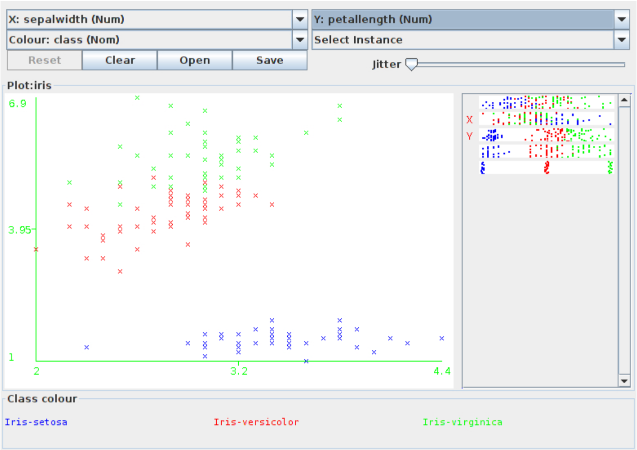
Attributes

No.		Name
1	□	sepallength
2	□	sepalwidth
3	□	petallength
4	□	petalwidth
5	□	class

Class: class (Nom)

Status
OK

Figure 10: The "Explorer" is the main graphical interface of WEKA. This screenshot shows the explorer with the loaded iris data set. It offers various features such as attribute selection, classification and regression, clustering and visualizations to explore the data set. After applying a data-mining algorithm, the trained model can be saved.



X: sepalwidth (Num)
Y: petallength (Num)
Colour: class (Nom)

Figure 11: To analyze the data graphically, 2D Scatter Plots and Scatter Plot matrices are offered by WEKA.

3 Visualization

The general goal of visualization is to provide a comprehensible and more abstract visual presentation of complex data to enable users to perform detailed investigations and to gain better insight into it [73]. This goal can be subdivided into explanatory and exploratory purposes. Visualizations have been used in human history in various ways. A famous example is Charles Minard's depiction of Napolean's invasion of Russia in 1812 (see Figure 13). It shows six types of data on a single map (the number of troops, temperature, distance, location relative to date, latitude and longitude).

Large data sets are likely to contain high dimensional data and humans are limited to perceive multiple attributes at once. Therefore, it is essential to consider which information is important to the user and to avoid overwhelming amount of irrelevant information.

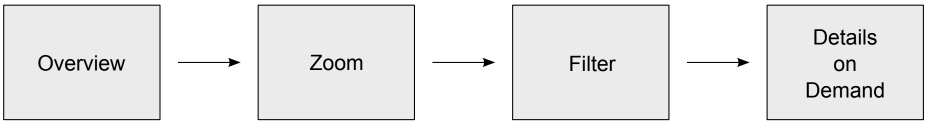


Figure 12: Shneiderman's mantra for visualization systems.

As depicted in Figure 12, many designs of visualization systems are influenced by Shneiderman's mantra [131, 73]:

> "Overview first, Filter and zoom, Details on demand"

It emphasizes how user interfaces should be designed so that users do not lose overview of the overall context or do not suffer from information overload.

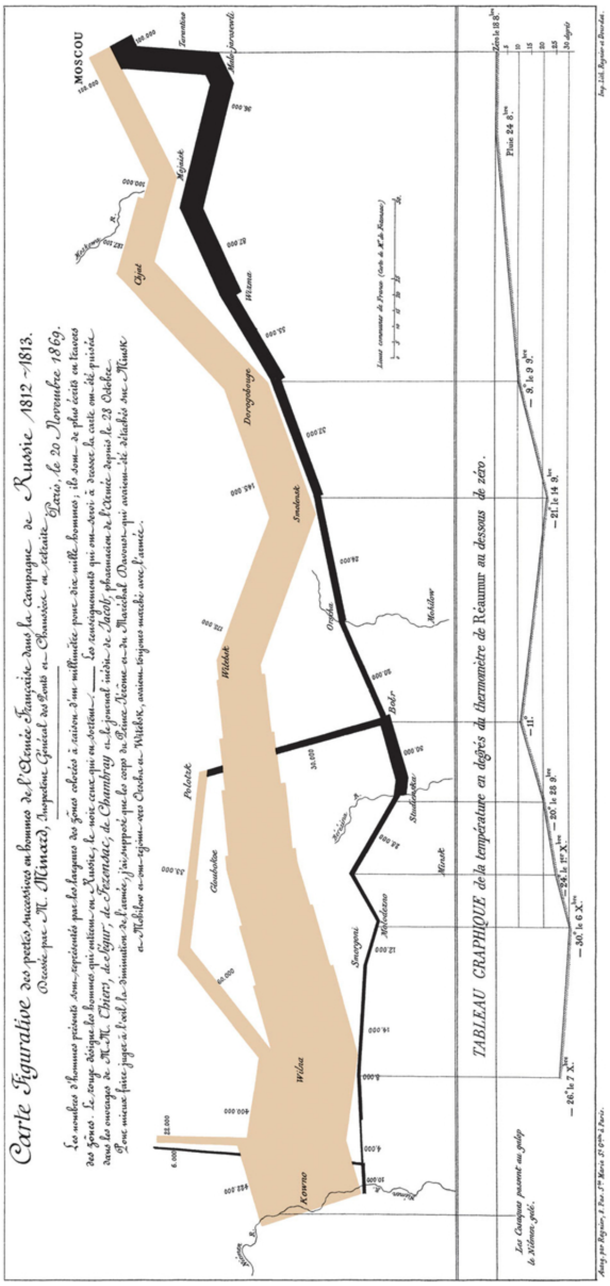


Figure 13: "Minard" by Charles Minard (1781-1870). Depiction of Napolean's invasion of Russia [104].

3.1 Categories of Visualization

In general, visualizations can be classified into *scientific visualization*, *information visualization*. Scientific visualization is about rendering data sets linked to time and space. Therefore, it often uses 3D renderings with additional (abstract) visual elements to depict the data (e.g., vectors, tensor data, volumes, etc.) [49].

Information visualization is usually about presenting abstract data which does not have references to the real-world dimensions space and time (e.g., business data, documents, software, etc.). Such data can be highly dimensional and due to it's abstract structure, it requires specific visual methods.

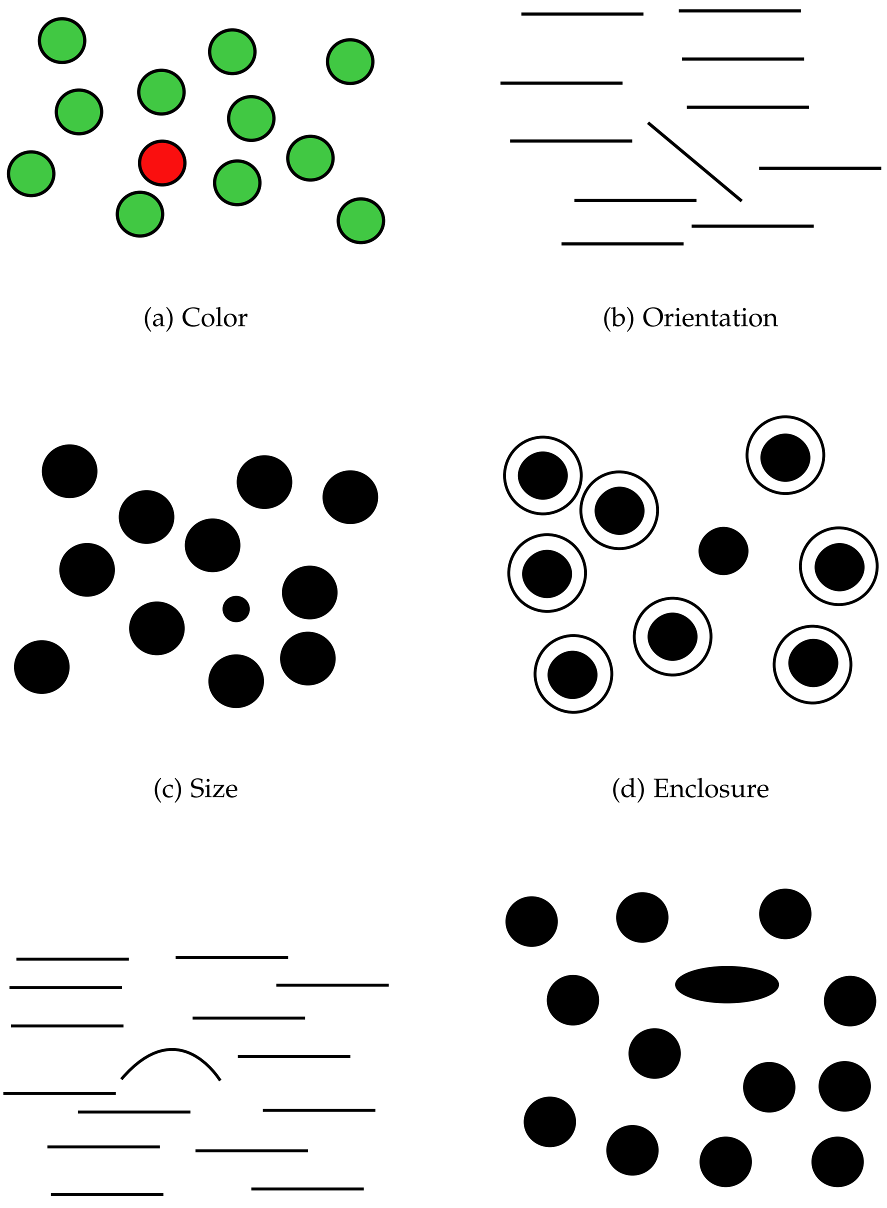
Some literature define a third categorization to organize the types of visualizations, namely *geovisualization*. Geovisualization has its roots in cartography and it refers to visualizations of geospatial data which includes attaching information to various maps [67].

In addition to that various sub-categories of visualizations exist. Some examples are software-, music-, biomedical-, social- or flow visualization.

3.2 Psychology of Perception

3.2.1 Pre-attentive Features

While statistics, machine learning and related research areas rely on computational power, visualization utilizes the human perception capabilities to communicate the content of the data. As shown in Figure 14, so-called pre-attentive features do not need cognitive effort for recognition [158, 142]. These properties make visualizations a powerful and appropriate tool to explore huge data sets faster.



(a) Color

(b) Orientation

(c) Size

(d) Enclosure

(e) Curvature

(f) Shape

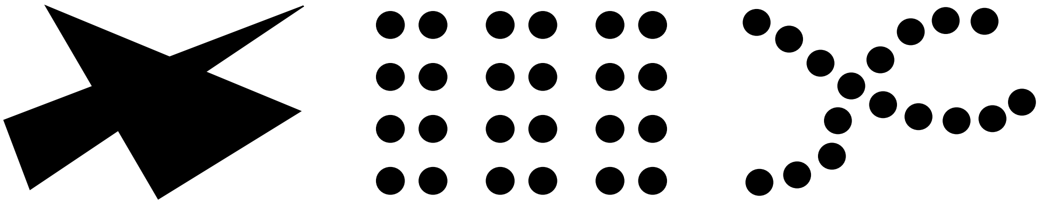
Figure 14: Pre-attentive features enable users to perceive differences in a glance without additional cognitive effort (popout effect). As an example, the change of color (a), orientation (b), size (c), enclosure (d), curvature (e) and shape (f) are strong pre-attentive features.

According to Ware [158], pre-attentive features can be categorized into the following categories:

- Line orientation
- Line lenght
- Line width
- Size
- Curvature
- Spatial grouping
- Blur
- Added marks
- Numerosity (one, two, etc.)
- Color
  - Hue
  - Intensity
- Motion
  - Flicker
  - Direction
- Spatial position
  - Two-dimensional position
  - Stereoscopic depth
- Convex/concave shape from shading

3.2.2 The Gestalt Laws

The Gestalt Laws are another theory which is relevant for visualization systems [75]. It implies that humans tend to perceive simple structures and shapes as shown in Figure 15. In addition to that, it says that the overall structure is more dominant than single parts (see Figure 15b and 15c). Nevertheless, pre-attentive features such as colors are even more dominant.



(a) The law of simplicity.

(b) The law of proximity.

(c) The law of continuity.

Figure 15: The Gestalt Laws tell that humans tend to perceive simple structures, grouped elements and continuous lines (a subset of all laws) [75]. The shown black area is perceived as two separate triangles (a). Due to alternating proximity, three columns are perceived in (b). The arrangement of circles in (c) let users perceive two continuous lines.

3.3 Visualization Stages

According to Ware [158], the general visualization process includes four stages (see Figure 16):

- Collection of data
- Preprocessing of data
- Mapping from data to a visual presentation
- Human perception and cognition

The stage *collection of data* includes storage, management and acquaintance of data to build a uniform basis for further processing. Next, *Preprocessing of data* describes the process of transforming and reducing the data. Data reduction is needed to limit the processed data to the current selection or context. Transformations can simplify the data format which makes it easier to work on it in the next stage. *Mapping from data to a visual presentation* maps the preprocessed data to a graphical representation. To maximize efficiency, the representation should make usage of pre-attentive features to outsource the users comprehension effort to the human perception system. Finally, the stage *Human perception and cognition* describes the processing of visual information of the human brain. To explore the data set and investigate further details, the analyst might change visualization settings or the current selection. This directly effects the mapping stage and in some cases even the

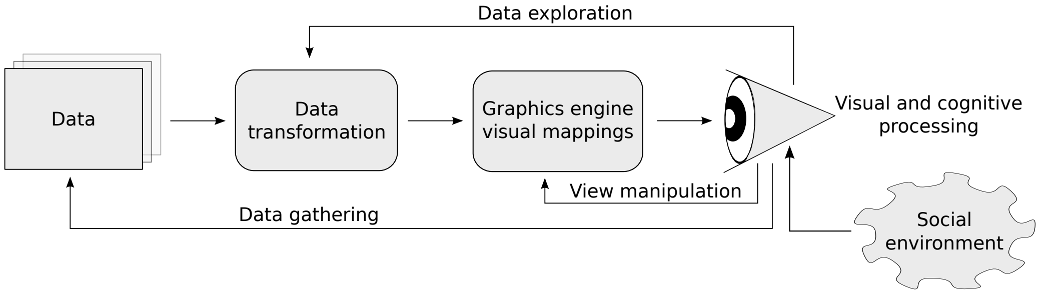


Figure 16: A depiction of the iterative visualization process. After preprocessing and transforming the raw data, a graphical mapping is performed and the result is shown. The user perceives the graphical representation and interprets it. The users final interpretation is highly depended on the users current domain knowledge. For further explorations, user interactions to change the view or data are essential.

preprocessing stage. Therefore, the general visualization process is iterative, while being coupled to the user.

3.4 Design of a Visualization System

A sophisticated visualization system meets several requirements to satisfy target users. The main purpose of a visualization is to support users to understand and explore the data for a given task. Therefore, the general functionality for usage and exploration of the data space requires to be relatively easy to use for main target users. To design functionalities to explore further details of the data, Shneiderman's mantra "Overview first, Filter and zoom, Details on demand" [131] should be considered well. Moreover, multiple data representations might increase effectiveness for different types of users or tasks [78].

According to Kerren et al. [78], the following questions need to be answered by performing a detailed investigation, to meet the above-mentioned requirements:

- Who will be the users of the visualization?
- What is the data that will be viewed?
- What tasks can be performed with the data?

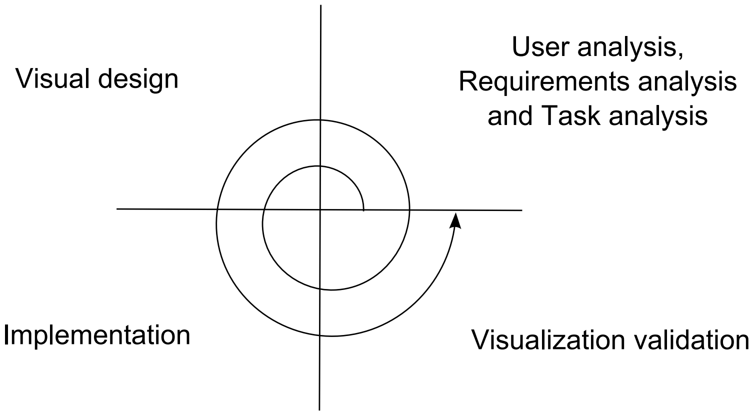


Figure 17: The design cycle is a generic approach to ensure quality of the design and implementation of visualization systems [78].

- What are some of the insights that the visualization will allow?

These questions imply user analysis, requirement analysis and task analysis. These analyzes are usually performed at the beginning of the project and after each visualization validation (discussed in Section 3.4.2). As depicted in Figure 17, this procedure results in an iterative design cycle. However, these steps are not necessary separated and the design cycle functions as a general guideline to develop visualization systems.

Moreover, it is essential to consider usability and the design of visualizations is strongly connected to the design of general interfaces. Thus, usability tests can be performed to find problems and to measure the general satisfaction of users citeAndrews2006. Problems might be the wrong type of visualization or even a too detailed representation of the data, which does not meet the users expectations to fulfill a specific task [78].

3.4.1 User Model and Requirements

An investigation of the user's mental model of the given data within the given context helps to derive and design a user model and find an appropriate visualization. A user model is an approximation of the mental model of a target user or even a group of users. The mental model of a user depends on the users individual domain knowledge, experience and expectations. It usually changes over time cause of gaining new knowledge or forgetting specific details. Moreover, the user might have wrong or outdated knowledge. Therefore, a designed user model does usually not match a user's mental model. However, the user model is the basis to derive a well defined visualization for a given task. If there are multiple users with different mental models of the same data, multiple user models and visualizations should be derived [78].

A proper definition of user requirements is essential to design a sophisticated visualization system. Its definition should contain the user needs and expectations of the visualization and corresponding interactions. Moreover, detailed investigation should be done on current activities and problems users deal with (including the strategy to solve it) [78].

3.4.2 Evaluation of Visualization Systems

An evaluation of the system is important to find needs of users and usability problems to enhance the visual system. Evaluations assist to check whether requirements are met and if target users accept the new visualization. In addition to that, the user can give hints for further improvements. Flexibility, freedom and usability commonly lead to a more successful system [15, 78].

In general, an evaluation of a system should imply the analysis of its interaction methods, usability and visualization [6]. As depicted in Figure 18, evaluations of a human-centered design involve users in early stages to ensure a higher final quality. In each iteration, evaluations are performed in each phase for the concept design, detailed design and implementation.

Evaluation methods can be divided into analytical methods and empirical methods [78]. Analytical methods include all methods to find problems without users. For that, experts investigate the design for possible problems and report these. Examples of analytical methods are heuristic evaluations, cognitive walk-through and early concept evaluation.

On the other side, empirical methods imply all methods, which observe users to measure parameters such as efficiency, effectiveness, satisfaction

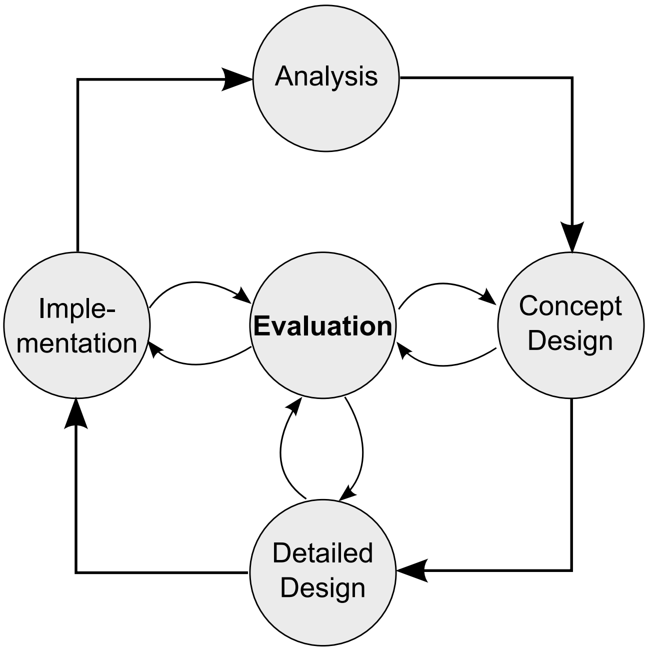


Figure 18: The evaluation cycle for a human-centered design emphasizes that the design and implementation has to be evaluated in each iteration [78].

and learnability. It is the most commonly used method, but its drawback is, that the system needs to be implemented to perform the evaluation.

3.5 Types of Interaction

Besides the actual visual representation of the data, user interactions are the second important part of visualizations systems. In contrast to a static representation, interaction techniques allow the user to navigate and explore the data set and change its presentation [75]. Yi et al summarized the most common interaction methods of information visualization techniques [171].

Selection

Selection of items implies a graphical highlighting to support the user to track selected data points while exploring the data space 19. Depending on the type of representation, selections are depicted in different ways (e.g., change of color, enclosure, etc.). A selection can also be combined with further operations.

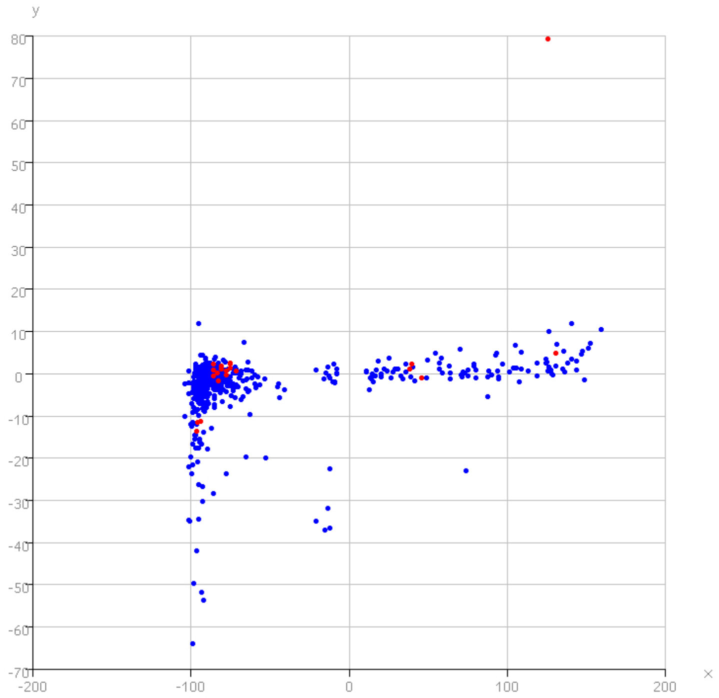


Figure 19: Selected data points are colored red for highlighting.

Exploration

Interaction methods to explore the whole data space are needed because the complete data set cannot be visualized in many cases. Reasons for that can be the size of the data, the size of the display or natural limitations of the human perception capabilities. Therefore, methods are needed to allow the user to change the currently viewed subset (see Figure 20). Common

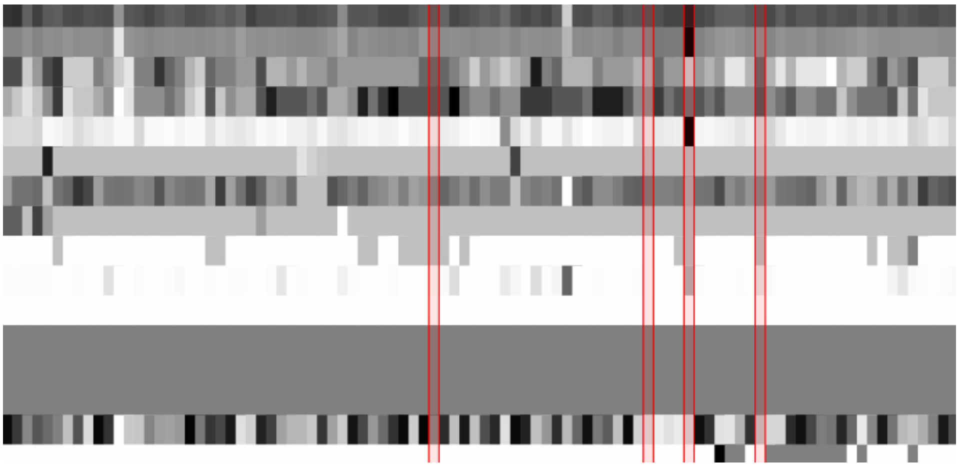


Figure 20: This gray scale heat map visualizes all attributes (rows) of all elements (columns) as a colored matrix. Selected elements are highlighted with red color. Often, the data set is too large to show all values simultaneously on the display. Therefore, a subset of the whole data set is shown and user interactions (e.g., panning, scaling) are used to explore the data.

methods are panning and direct walk. When changing the state of view to a new one, smooth animations support users to understand the relation to its previous view state.

Reconfiguration

A reconfiguration of a visualization describes the change of the perspective on the data. In other words, the data is being rearranged in some way. For example, this can be achieved by performing a sorting or by reconfiguring the view itself (e.g., changing the attribute of an axis of a scatter plot).

Encoding

An encoding enables the user to change the entire representation of the data by changing the visualization method itself (e.g., changing a histogram to a pie chart). An additional representation can emphasize different aspects of the same data set and therefore, users can gain deeper insight and further

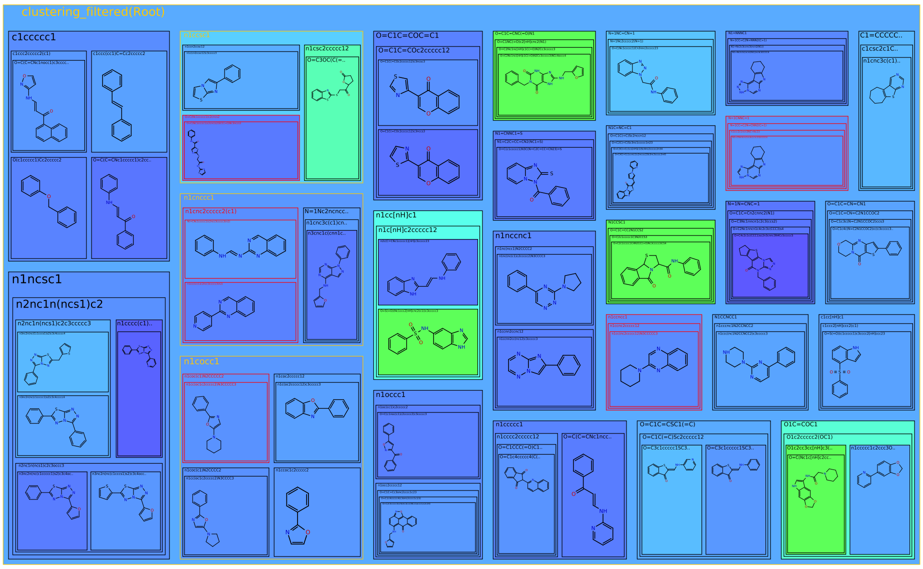


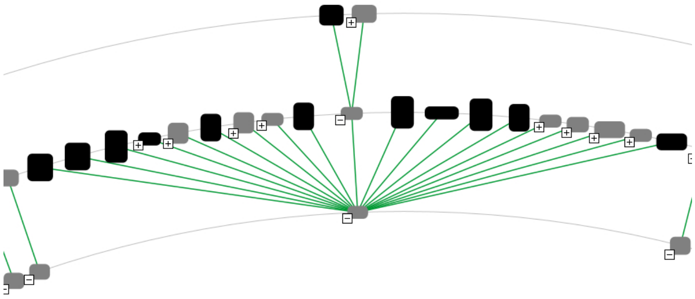
Figure 21: A tree map is a space filling visualization to represent hierarchies. One dimension of data set has been mapped to the color of the corresponding area. Selected elements are highlighted with a red borders. (generated with Scaffold Hunter [163]).

understanding. For example, the tree map depicted in 21 visualizes the same data set as the scatter plot in Figure 19 does.

Abstraction / Elaboration

Abstraction and elaboration methods are used to adjust the level of detail of a representation. In a common user scenario, the user wants to get an overview of the whole data at the beginning. Then, depending on the users goal, he or she might want to explore the data space in a more detailed way. This function involves the first, second and last phase of Shneiderman's mantra for visualization systems called *Overview*, *Zoom* and *Details on Demand*, respectively (see Figure 12). As shown in Figure 22, a practical method for this type of interactions are *semantic zoom* and *geometric zoom*.

Weaver states [161]:



(a) When zoomed out, less details are shown: This scaffold tree does not show detailed representations of its nodes (molecules) to avoid information overflow. In addition to that, nodes are too small to depict more details in a legible way and rendering performance is increased.

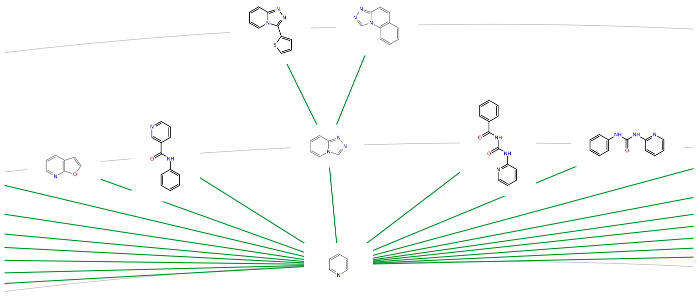
> “Semantic zoom is a form of details on demand that lets the user see different amounts of detail in a view by zooming in and out.”

Filter

Filter provide users an opportunity to change the presented data set by defining conditions without changing the actual visualization method. When applying new conditions, the original data stays unchanged but only a subset of the data, which applies to the defined conditions, is shown. Dynamic query control, by using sliders to set value limits, is an example for a common filter interaction. Filter interactions represent the third stage of Shneiderman’s mantra.

Connecting

Connecting interactions enable users to comprehend relations between different views and related elements. Several visualization systems use multiple views in parallel to enhance insight (multiple encodings). When selecting an element in one view, the element will be highlighted in all other



(b) When zoomed in, a more detailed representation is shown: Nodes of the scaffold tree are rendered as molecules instead of filled rectangles.

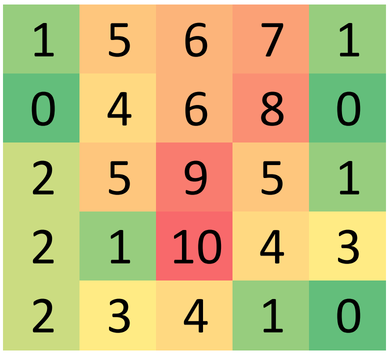
Figure 22: Semantic zoom is a technique to adjust the level of detail according to the geometric zoom level. When zoomed out, less detail is shown to not distract the user with too many details. This helps to get a better overview more easily (a). When zoomed in, less elements are shown but they can be rendered in a more detailed way (b).

views as well (compare Figures 19,20 and 21). This particular connecting interaction is called *linking and brushing*. Another approach is highlighting all neighboring (or similar) elements of a selected element to emphasize relations between data points.

3.6 Selected Examples

3.6.1 Heat Map

A heat map is table-based visualization technique and basically, it is a rectangular map consisting of rows and columns. It is a very popular visualization method and its purpose is to visualize a data matrix. For that, each cell of the heat map represents the appropriate value of the data matrix by color shading (see Figure 23). The color is determined by a color mapping function and it supports the user to comprehend and compare a huge amount of values more efficiently [165].



1	5	6	7	1
0	4	6	8	0
2	5	9	5	1
2	1	10	4	3
2	3	4	1	0

Figure 23: An example illustrating the basic principle of a heat map. Due to an applied color mapping function, each cell is shaded according to the appropriate value in the data matrix (low values are green, high values are red). Due to the use of colors, the observer recognizes high values in the center and above the center at a glance.

Thus, the main advantage of heat maps is the ability to visualize large data matrices on a single display (see Figure 20). Cells may be scaled down to the size of a single pixel and thus, the higher the resolution of the display, the larger the data matrix can be visualized as a whole.

Even if the matrix does not fit into the display (or the user prefers a larger cell size), heat maps allow several interactions to explore the data. The most common interactions are zooming and panning for further exploration. For coloring cells, different color mapping functions may be used to highlight different aspects. A more advanced feature is interchanging rows or columns to change the order. This might be done automatically or manually.

Selections need to be handled in a special way, because a selection of a single cell would mean the selection of a single property of one element. Thus, depending whether rows or columns represent selectable elements, the appropriate dimension needs to be selected as a whole. Selections can be highlighted by shading, drawing a border (see Figure 20) or by depicting an indicator along the border of the heat map.

3.6.2 Self-Organizing Map

As described in Section 2.6.2, a Self-Organizing Map represents a data space of an arbitrary high dimension by a two-dimensional grid. The neighborhood relations within the grid are being preserved during training. Therefore, the grids properties can be used to visualize the high dimensional data as a two-dimensional matrix. A known drawback of a planar 2D grid is, that nodes along the edge of the grid do not have the same count of neighbors as central nodes. Thus, the represented information of these nodes tends to be less. To tackle this problem, borderless manifolds (e.g., torus) should be used and to visualize its surface, a tiled representation of the minifold can be used [155, 151].

There exist multiple types of SOM visualizations. The most common is the so-called U-Matrix.

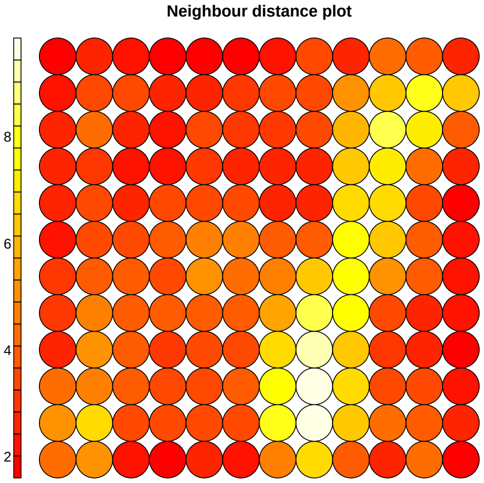


Figure 24: A colored U-Matrix visualization of a trained 12x12 SOM grid generated with R. The iris data set was used to train the SOM and distances are represented by cell colors. White and yellow indicates long distances between neighboring nodes and red indicates a high density of neighboring nodes. Thus, joint areas indicating long distances are interpreted as cluster borders.

U-Matrix The U-Matrix visualizes the distance of a node to all its neighbors (U-height) [152]. In other words, it visualizes the allocation of the input data in an abstract way using distance information.

Let N_m be the set of neighbors of a node m, $w(n)$ the associated vector of node n and $d(w(n), w(m))$ a distance function to calculate the distance between both given vectors, then

$$\text{U-height}(m) = \sum_{n \in \mathrm{N_m}} d(w(n), w(m))$$

As depicted in Figure 24 and 25, a large U-height indicates that trained nodes are located far away from each other, whereas a low U-height represents small distances to neighbor nodes. A small distance to a nodes neighborhood represents a high density of training data points within its surrounding area (cell). In Figure 24, a cluster boundary is shown by a yellow and white elongated area. The gray scale SOM visualization in Figure 25 indicates cluster boundaries by gray areas.

To depict a U-Matrix, a Heat map (using gray levels) or a landscape visualization is commonly used. Considering the landscape method, a U-Matrix has several properties [152]:

- A "mountain range" (large U-height values) represents a border of two clusters.
- A "valley" (low U-height values) indicates a cluster center.
- A "sinkhole" indicates outliers.
- Data points are typically in depressions.

P-Matrix Compared to the U-Matrix, the P-Matrix uses density values instead of distances. Therefore, the data density (P-height), within a predefined volume (e.g., Pareto sphere) is calculated for each node, where the volume is centered at the nodes position in the data space [152]. A large P-height represents a high density, whereas a low P-height represents a sparse environment. Considering these properties, the P-Matrix is a complementary (and compatible) visualization to the U-Matrix. Let n be a node of the SOM grid, $w(n)$ the associated vector of node n and $p(n, X)$ an empirical

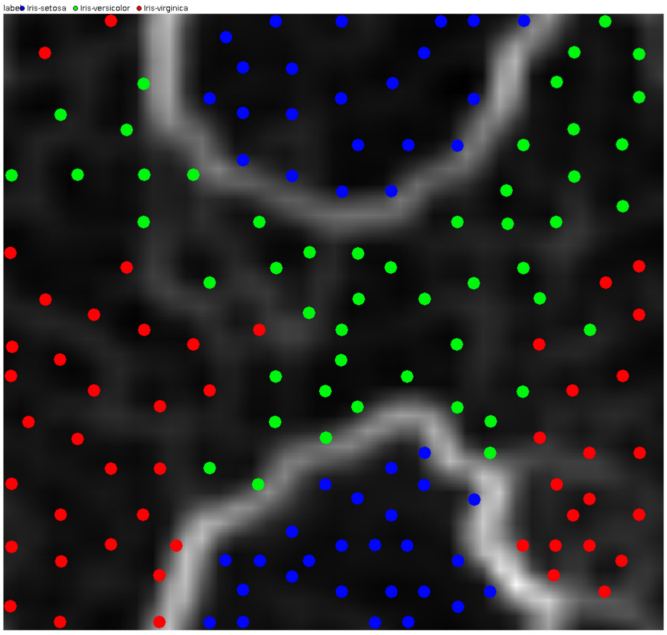


Figure 25: A gray scale U-Matrix visualization of a 40x40 SOM grid generated with Rapid Miner. The color of each node indicates its corresponding class, which is defined by the labeling of the data records (blue = iris-setosa, green = iris-versicolor, red = Iris-virginica). After performing an unsupervised training of the SOM with the iris data set, the U-Matrix yields the cluster border between the iris-setosa (blue) and the other types very clearly (bright areas represent longer distances between nodes). However, the cluster border between the classes iris-versicolor (green) and iris-verginica (red) has not been found.

density estimation of the node n in the input space X, then:

$$\text{P-height}(n) = p(w(n), X)$$

Spread of values Another common visualization of SOM is a color-coded matrix (heat map) representing a single dimension [155]. The cells color depicts the value of the appropriate attribute (see Figure 27). Thus, the distribution of values can be investigated over the whole allocated data space. When combined with other matrices, which depict different dimensions,

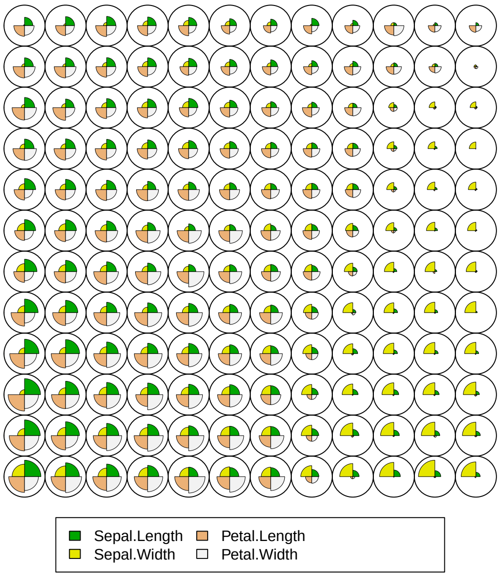
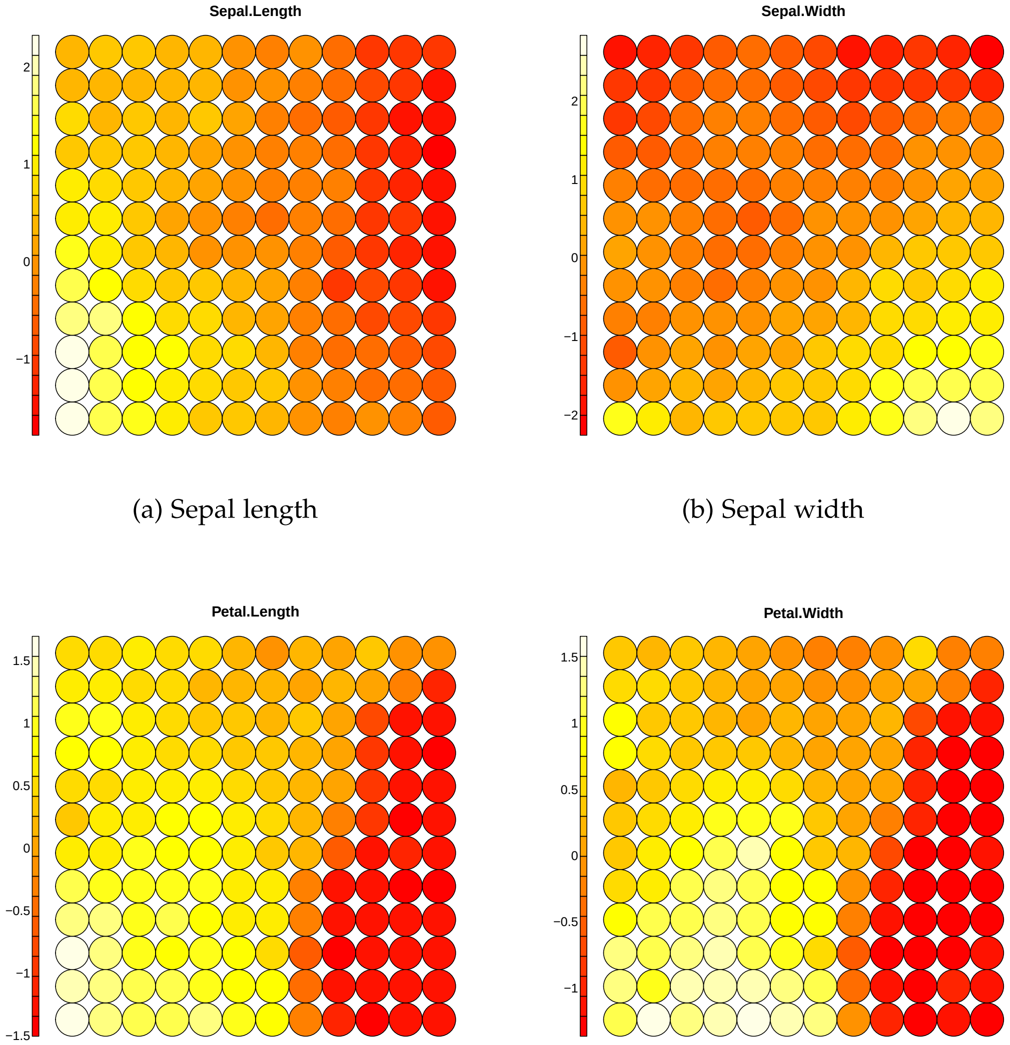


Figure 26: A representation of all trained SOM nodes interpreted as code book vectors. After a sufficient training, the SOM spans all values of the training set and this particular visualization supports the understanding of the distribution of values within the input space. In each cell, a segment plot is shown to depict all values of its corresponding code book vector. In the bottom left corner, all four dimensions are of high values. In contrast to that, the upper right corner represents low values of all dimensions. The bottom right corner depicts high values of the dimension *sepal width* (yellow) and low values of all remaining dimensions.

correlations and relations between dimensions can be analyzed. In addition to that, Figure 26 depicts all codebook vectors by using segment plots in a single matrix.



(a) Sepal length

(b) Sepal width

(c) Petal length

(d) Petal width

Figure 27: Heat maps depicting the value distribution of 4 dimensions over the whole SOM grid. Each matrix depicts a single attribute of the iris data set. Concerning the depicted SOM grid, values of the attribute *sepal length* decreases diagonally from the bottom left corner to the upper right corner (a), whereas the *sepal width* decreases from bottom right corner to the upper left corner (b). The *petal width* and *petal length* are of a similar distribution, where low values are concentrated on the right side (c,d).

4 Visual Analytics

There exist commonalities between the general processes of visualization and knowledge discovery in databases. Both processes are iterative and visualization is a key element of KDD to communicate the discovered knowledge to the user. Whereas the term *information visualization* describes the general visual presentation of data, KDD requires complex user interactions, which do not only affect the visualization itself. The user requires immersive insight and control of both the visualization and the data mining process to improve the ability to explore the data. This exceeds the definition of general information visualization, because this term does not necessarily imply data analytics. However, the demand for such interactive analytical solutions has been rising recently. Therefore, a new field called *visual analytics* has been announced. It combines several fields of data analytics, human-computer interaction and visualization for improved decision making and analysis (see Figure 28).

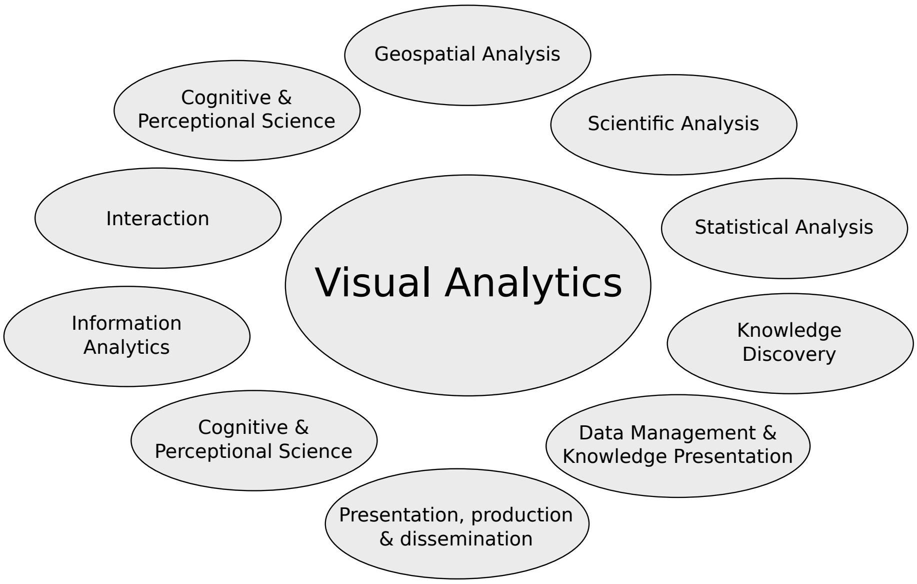

Geospatial Analysis
Cognitive & Perceptional Science
Scientific Analysis
Interaction
Statistical Analysis
Visual Analytics
Information Analytics
Knowledge Discovery
Cognitive & Perceptional Science
Data Management & Knowledge Presentation
Presentation, production & dissemination

Figure 28: The scope of visual analysis by Keim et al. emphasizes all research field which are involved in visual analytics [77].

4.1 The Visual Analytics Process

The visual analytics process implies the selection of automated data mining algorithms combined with an appropriate visual presentation [73, 75]. Therefore, it is a combination of traditional data mining and information visualization (see Figure 29). To emphasize the process, Keim extended Schneiderman's mantra as follows [77]:

> "Analyse First – Show the Important – Zoom, Filter and Analyse Further – Details on Demand".

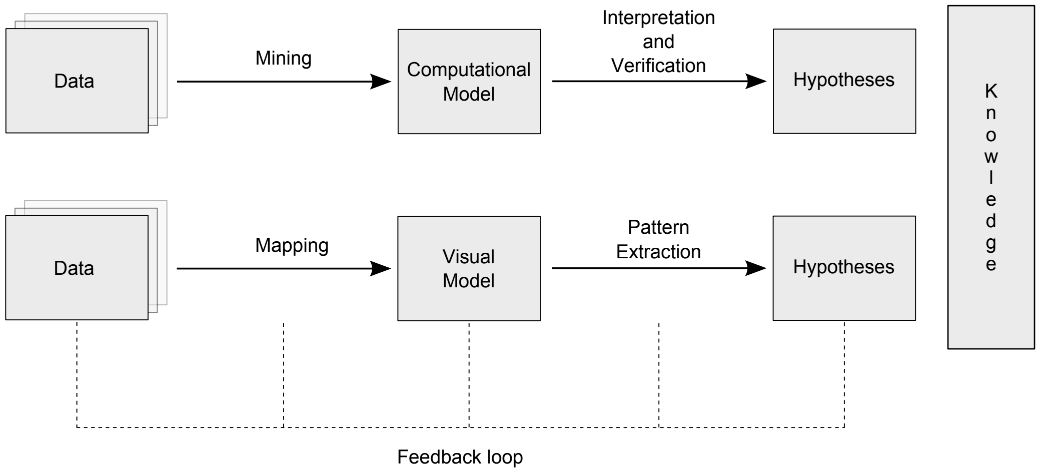


Figure 29: A comparison of analytic prosesses between conventional data mining (top) and information visualization (bottom) [75].

Moreover, an essential part of the overall visual analytics process is the sense-making loop [73]. As depicted in Figure 30, the visualization process is iterative, where the user interface acts as link between data and user.

Formal definition: Keim et al. presented a formal definition of the visual analytics process [77]:

The data might be acquired from multiple data sources. In case of m data sources, a heterogeneous data set $S = S_1, \ldots, S_m$ is given, where each $S_i, i \in (1, \ldots, n)$ has attributes $A_{i1}, \ldots, A_{ik}$. As the ultimate goal of data analytics is gaining a new insight I, the process can be understood as a

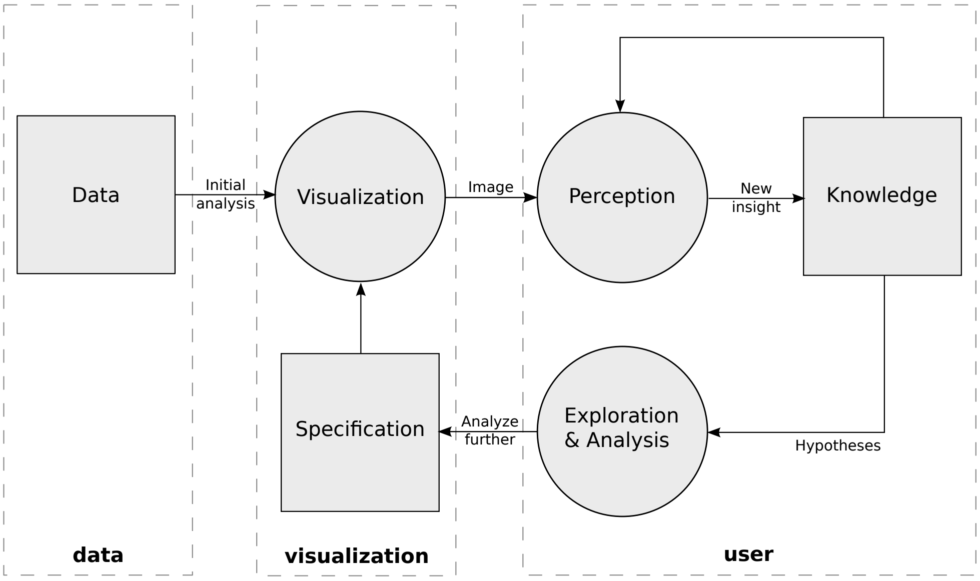


Figure 30: Depiction of the generic visualization process by Wijk [154]. Circles represent processes which transform inputs. It emphasizes the iterative user interaction with the visualization to explore the data.

transformation $F : S \rightarrow I$. In general, an insight I can be derived either directly from one or more visualizations V or from assumed hypotheses H derived from automated analysis methods. The transformation function f consists of multiple elements $f \in \{D_W, V_X, H_Y, U_Z\}$.

$D_W : S \rightarrow S, W \in \{T, C, SL, I\}$ describes the serial application of basic data pre-processing tasks $D_T(D_C(D_{SL}(D_I(S))))$, where D_I is a data-integration function, D_{SL} performs data selection tasks, D_C cleans the data and D_T performs data transformations.

$V_X, X \in \{S, H\}$ represents the visualization function to generate a set of visualizations from either the data itself ($V_S : S \rightarrow V$) or from a hypothesis ($V_H : H \rightarrow V$).

$H_Y, Y \in \{S, H\}$ describes the generation of a hypothesis which can be derived from the data ($H_S : S \rightarrow H$) or from a visualization ($H_V : V \rightarrow H$).

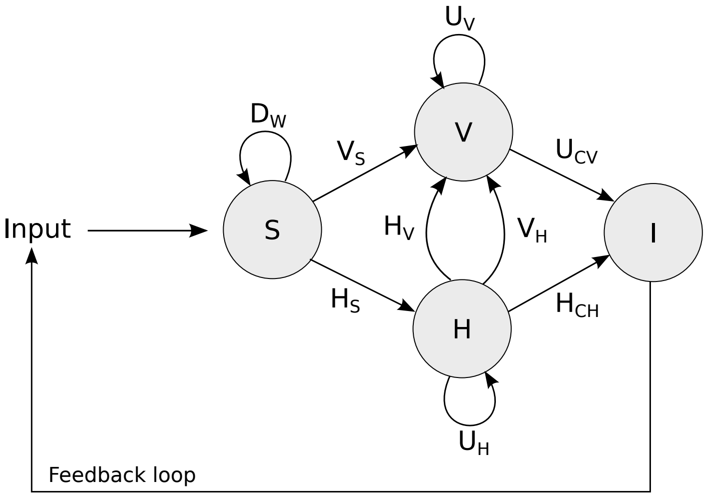


Figure 31: The formal process of visual analytics [77]

The user interaction function $U_Z, Z \in \{V, H, CV, CH\}$ is an essential part of the overall process. User interactions can manipulate only the visualization $U_V : V \rightarrow V$ or the derived hypothesis $U_H : H \rightarrow H$. In addition to that, further insight I can be provided by either the visualization $U_{CV} : V \rightarrow I$ or the current hypothesis $U_{CH} : H \rightarrow I$ (see Figure 31).

4.2 Categorization of Visual Analytics

Visual analytics techniques can be categorized in several ways. The categorization used by Bertini et al. [11] emphasizes whether the visualization or the analytical part plays the major role. For that, they used three categories, namely: *computationally enhanced visualization*, *visually enhanced mining* and *integrated visualization and mining*. Turkay et al. [147] presented a 2-dimensional classification scheme. The first categorization distinguishes the type of analytical task which is classified in *summarizing information*, *finding groups & classification* and *investigating relations & prediction*. The second one categorizes the applied visualization technique according to its integration level of analytical and computational tools: *visualization as a presentation medium*, *semi-interactive use of computational methods* and *tight integration of* *interactive visual and computational tools*. The transition between these levels is seamless. However, an investigation of state-of-the-art methods by applying this 2D categorization revealed, that there is still a lack of fully integrated methods.

4.3 Quality Metrics

Quality metrics are used to rate representations regarding to its usefulness for the user. In other words, quality metrics measure the relevance of a given representation and its settings. It can be used to find interesting projections, to reduce clutter or adjust the abstraction level of the data representation [114]. For example, when considering a clustering problem, a 2D projection (e.g., scatter plot) of a high dimensional data which shows separated clusters, is more relevant to the user (due to ease of cluster identification) than 2D projections, which show interleaved clusters (different dimensions for the x-axis and y-axis). Therefore, with the support of metrics, visual analytics systems can recommend highly rated representations to the user automatically [8].

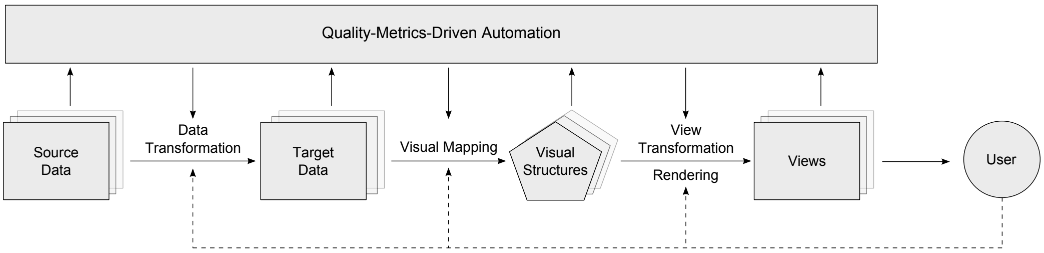
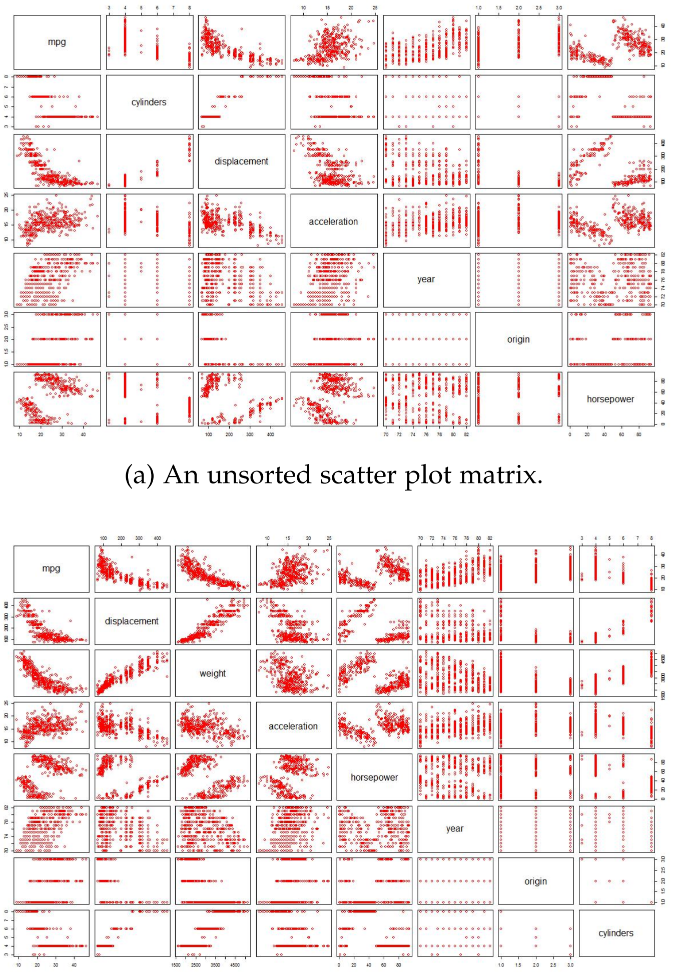


Figure 32: The visualization pipeline extended with a quality metrics measurement in each stage. Quality metrics are calculated in both data space and image space. The user can control and adjust the metrics system via user interactions.

As depicted in Figure 32, quality metrics can be calculated in any stage of the visualization pipeline while being under control of the user. Calculations are performed in both data and image space [12, 137]. Examples of measured values are correlations, outlier metrics, image quality (e.g., clutter) and feature preservation.



(a) An unsorted scatter plot matrix.

(b) A sorted scatter plot matrix.

Figure 33: (a) Dimensions of the scatter plot matrix are unsorted and therefore, correlations are harder to perceive. (b) Sorted dimensions support the user to recognize specific types of correlations more easily. (Image source: Behrischa et al. [8]).

There already exist several approaches for using quality metrics, namely the optimization of the projection, ordering, abstraction, visual mapping and the view itself.

As mentioned above, *optimizing the projection* includes finding the best combination of dimensions for the x-axis and y-axis to enhance the comprehensiveness of the presentations for a given task. *Ordering optimization* describes finding a sorting of the data which results to a better representation in the current view. Automatic adjustment of the *abstraction level* can enhance the understandability of a visualization. An example is the reduction of the data while preserving the overall structure (e.g., reducing the count of lines in a Parallel Coordinates plot). Furthermore, finding a better *visual mapping* from data features to visual features (e.g., colors, shapes) is another approach to enhance insight. A *view optimization* adjusts parameters of the view to emphasize interesting structures of the data set (see Figure 33).

As the system is not able to know what the user exactly does want to explore, some current approaches enable the user to interact with the quality measurement [12]. For that, a *threshold selection* changes parameters of the metrics calculation and a *metrics selection* combines or enables/disables individual quality metrics.

4.4 Selected Examples

4.4.1 Clustered Heat Map

A clustered heat map (CHM) is an extended version of a normal heat map which is discussed in Section 3.6.1. A CHM uses hierarchical clustering to order its rows or columns – or even both rows and columns. As clustering allocates similar elements next to each other, it supports to find patterns and relations within the data set (see Figure 34).

To define a similarity relation, a distance function (metric) needs to be defined. Some examples of common distance functions are *Euclidean distance*, *Squared Euclidean distance*, *Minkowski distance* [168] and *Manhattan distance* [85].

Moreover, a hierarchical clustering does also need a linkage criteria to define the distance of two sets. Common linkage criteria are *single-linkage* [133], *complete linkage* [23], *mean linkage* (also called *Unweighted Pair Group Method with Arithmetic Mean*) [135] or *centroid linkage* (also called *Unweighted Pair-Group Method using Centroids*) [71].

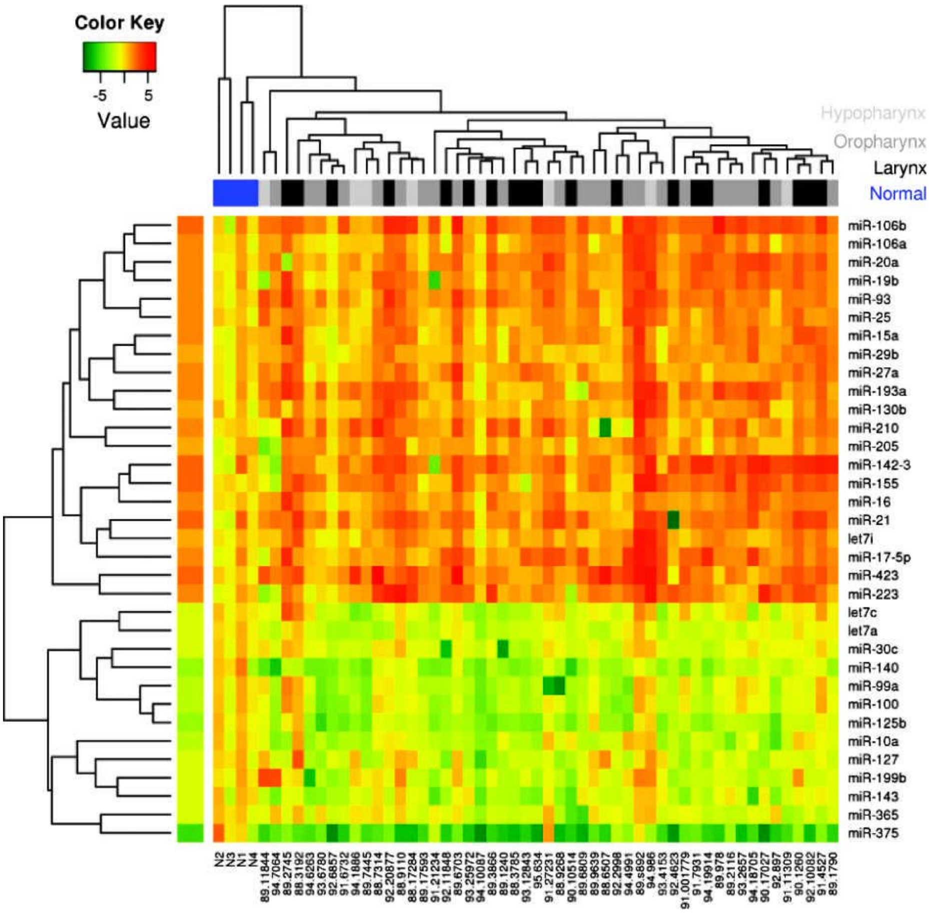


Figure 34: This clustered heat map uses clustered rows and columns to order both dimensions. Clustering supports users to identify relationships and patterns within the multivariate data set. Adjacent dendrograms on the top and to the left depict the cluster hierarchy. (Image source: Hui et al. [61]).

Dendrograms A dendrogram is a tree-like visual presentation of a hierarchy. As shown in Figure 34, the dendrogram on the left side represents the result of a clustering of all rows. Each element is treated as a single cluster and the height of a cluster represents the dissimilarity of both cluster members. Dendrograms are also commonly used as stand-alone visualizations for clustered elements (e.g., molecules, proteins, etc.) to investigate differences and similarities.

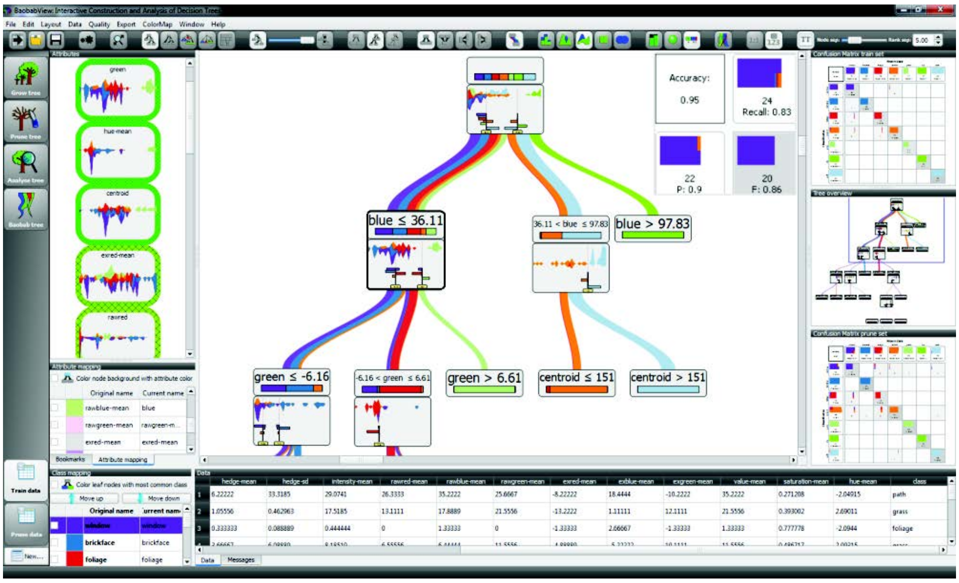
Configuration of Clustering To control the clustering process, the user can configure the clustering parameters (distance function, linkage criteria) to achieve different ordering of rows and columns. In addition to that, the user might select a sub-cluster of the dendrogram to perform a subsequent clustering on it. This enables the expert to analyze the multivariate data combined with the support of clustering techniques.

4.4.2 Interactive Decision Trees

Several approaches exist to construct decision trees interactively to support experts to contribute their domain knowledge to the creation process [7, 153, 159]. Ankerst et al. [7] and Ware et al. [159] introduced methods to support users to create decision trees based on visualized data using circle segments and scatter plots respectively. Both approaches are based on drawing separating elements (e.g., lines) to split the data in a graphical manner. The separation is interpreted as a new predicate resulting to a new node being attached to the decision tree. The user might perform another split on several subsequent data sets to enhance the classification performance. An advantage of this approach is that users do not tend to cause overfitting and performance evaluations showed that users are able to create decision trees of almost the same classification performance as trees generated automatically by algorithms [7].

Another approach was introduced by Elzen et al. [153]. It supports users to create, optimize, prune and analyze decision trees. Therefore, the visualization of the decision tree itself has been enhanced instead of visualizing the data (see Figure 35). In addition to that, each node of the tree visualizes the data set, which is being divided by the node's predicate, in a compact way by using distinct colors for each class.

Areas within a horizontal bar depict the quantity of elements of each class and a stream graph plots the distribution of all elements over the attribute's value range. Each colored stream represents a class and the overall graph supports users to determine whether an additional division of the data set is reasonable. Furthermore, a histogram depicts the amount of elements of each class on both sides of the split point. These combined visualizations, which are shown in every tree node, enable analysts to investigate the classification



Attribute mapping

☐ Color node background with attribute color

		Original name	Current name
☐	🟩	rawblue-mean	blue
☐	🟪	rawgreen-mean	rawgreen-m...
☐	⬜	exred-mean	exred-mean

Class mapping

☐ Color leaf nodes with most common class

Move up Move down

		Original name	Current name
☐	🟦	window	window
☐	🟦	brickface	brickface
☐	🟥	foliage	foliage

Data

	hedge-mean	hedge-sd	intensity-mean	rawred-mean	rawblue-mean	rawgreen-mean	exred-mean	exblue-mean	exgreen-mean	value-mean	saturation-mean	hue-mean	class
1	6.22222	33.3185	29.0741	26.3333	35.2222	25.6667	-8.22222	18.4444	-10.2222	35.2222	0.271208	-2.04915	path
2	1.05556	0.462963	17.5185	13.1111	17.8889	21.5556	-13.2222	1.11111	12.1111	21.5556	0.393002	2.69011	grass
3	0.333333	0.088889	0.444444	0	1.33333	0	-1.33333	2.66667	-1.33333	1.33333	0.777778	-2.0944	foliage

Figure 35: The Baobabview is a visual analytics solution using interactive decision trees [153]. An unique color is assigned to each class and compared to standard trees, it depicts the amount of class members of each data subset applied to a particular tree node. Besides the condition, each node visualizes the distribution of all classes by using a stream graph to increase comprehensibility of the conditions impact on the data set. Node links consist of colored sub-threads where each thread represents one class and its thickness is coherent to the count of its class members. (Image source: Elzen et al. [153])

performance of the corresponding predicate. This visualizations function as decision support for further interactive adjustments. To examine the overall classification performance of the decision tree, a visual confusion matrix is provided. It depicts the count of misclassifcations in a graphical manner. Moreover, node links are represented as a bundle of colored streams and multiple tree layouts are used for further investigations. For example, a clustering and placement of all tree leaves on the same level enable analysts to identify which classes are easier to classify (represented by a low count of leaves). The thickness of each stream is relative to the amount of data entries of the corresponding class to depict the amount of elements.

4.4.3 Visual Cluster Analysis with Interactive Kohonen Maps

The Visual Cluster Analysis with Interactive Kohonen maps by Schreck et al. is a visual analytics approach to involve user knowledge in the training process of Kohonen maps [129]. The standard training of SOMs only supports the configuration of certain learning parameters (e.g., learning rate, neighborhood topology, count of iterations, grid size, etc.) before the training has been started. The training process of a SOM is an unsupervised method and the final outcome depends on the data and the random initial state of the grid. Thus, the final result might conflict with the users expectations based on the present domain knowledge. Therefore, it would be an advantage to allow users to steer the learning process.

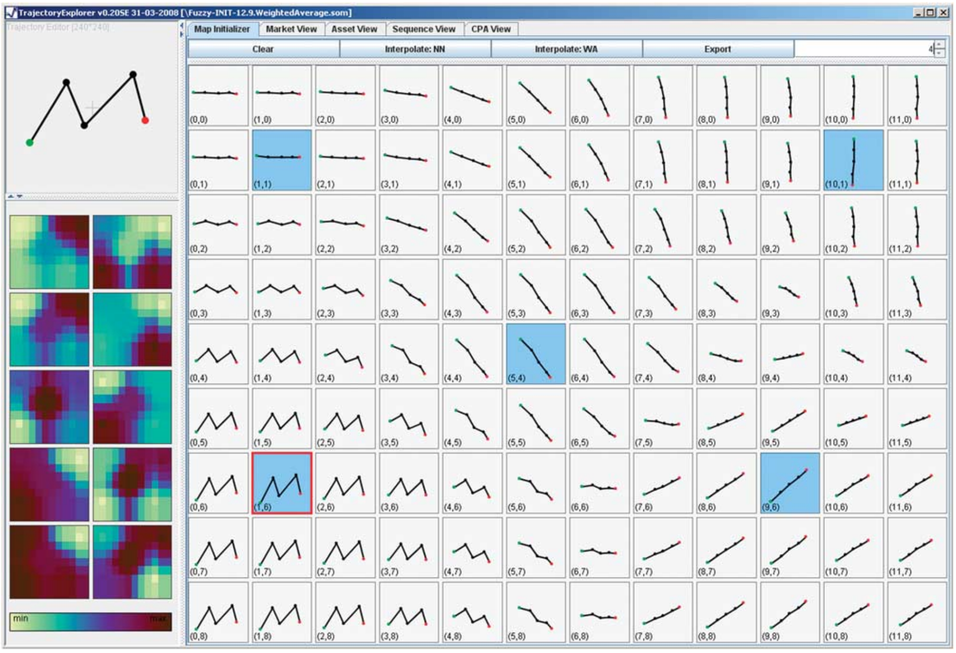


Figure 36: Visual clustering of trajectory data with five fixed user-defined trajectories [129]. (Image source: Schreck et al. [129])

In detail, the approach aims to cluster normalized trajectory data. An advantage of this particular data type is, that it can be visualized compactly by drawing the corresponding path (see Figure 36). The user can define an initial state manually and in addition to that, the user can pause, modify and resume the learning process at any time. When stopped, learning parameters and even individual trajectories can be modified. In case of having a non-satisfying result (e.g., poor classification performance or significant drift from user expectations), the approach also offers an undo function to rewind the learning process to a certain degree. To allow finer granularity, individual learning parameters can be set and modified for a single cell.

This approach can also be applied to other data types and the combination of all these functions enables the user to steer and modify the learning process of a Self-Organizing Map in a detailed way. However, too many configurations might lead to a low classification performance.

4.5 Open Problems

In general, visualizations still face several challenges and open problems [73]. First of all, the quality of the analyzed data must be sufficient to realize a visualization representing reliable knowledge. Obviously, even highly sophisticated visualizations cannot compensate data of low quality leading to a wrong result.

Another challenge is the scalability of such systems concerning the volume and dimensionality of the data. Specific data sets require tailored visualizations and nowadays, there is no generally accepted framework for a given problem. Therefore, there does no encompassing evaluation of visualization techniques exist.

One of the main problems are usability issues which are still recognized as an optional add-on. Most visualizations techniques can only unfold their effectiveness in combination with sophisticated realizations of human-computer interactions. Moreover, specialized data presentations often require much knowledge about statistics resulting in a steep learning curve. Many systems are based on statistical and stochastic results, but several approaches lack of informing users about the certainty of the presented result.

Regarding visual analysis, HCI plays even a more important role as the user wants to explore the data space and the result of the analysis. Interactions might adjust the underlying analytical process leading to further calculations. As mentioned before, there is a lack of a seamless integration of such complex analytical process. In addition to that, as the data set is usually of huge size, depending on its size, standard algorithms might take several hours or even days to finish all performed calculations. These circumstances do not allow any reasonable user interactions. Therefore, specialized infrastructures and adopted algorithms which support adjustments are essential [147].

4.6 Outlook

As one of the main problems are usability issues, further research about Human Computer Interaction is required. Future visual analytics systems might integrate the overall process more seamlessly and focus on the questions "What is interesting?". The answer depends on the whole context including the current task, the users preferences and knowledge, etc. Therefore, the computer needs to interpret and understand the current context and adjust to it in an appropriate way [147]. Novel approaches in this field are Programming by Feedback [128] and Attention Routing [147]. Both approaches try to adapt to the users behavior and perform appropriate adjustments to offer the user more interesting results.

5 Application of KDD in Biomedical Domain

This Section discusses necessary working steps to realize a visualization of biomedical data to allow knowledge discovery and decision support for experts (illustrated in Figure 37). This analysis is based on multiple studies, which present different approaches to extract new knowledge from medical data sets [94, 145, 173].

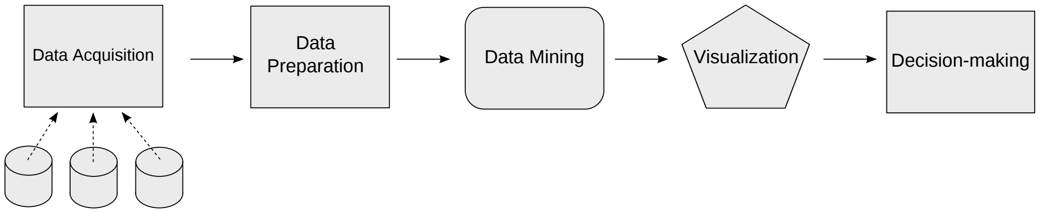


Figure 37: This Figure illustrates the general process of an application of KDD in the biomedical domain. Data can be acquired from multiple biomedical databases and a strong emphasis is put on data security, anonymity in the data preparation phase and on all subsequent steps.

5.1 Data Acquisition

Useful data sets can be obtained from multiple sources such as public sources offered by companies or research institutes (open data), biomedical research (e.g., biobank, gene bank), patient databases in hospitals or laboratories [79]. Therefore, the data might contain various images (e.g., scans, photos, etc.), physician's observations and reports, laboratory data and patient interviews [19].

According to Cios et al., heterogeneous data can be categorized into several classes [19]:

- Heterogeneity of medical data
- Ethical, legal, and social issues
- Statistical philosophy
- Special status of medicine

“Omics” Data

Commonly, biomedical data sets are generalized by the term “omics” data which covers many research fields of biology (e.g, genomics, proteomics, metabolomics, microbiomics, etc.) [53]. For example, the largest biobank in Europe is located in Graz at the Medical University of Graz [62]:

> “It houses nearly 6 million samples including formalin-fixed paraffin embedded (FFPE) tissue samples kept at room temperature, fresh frozen tissue samples kept in the vapor phase of liquid nitrogen and samples of body fluids (blood, serum, plasma, buffy coat, urine, liquor) kept at minus 80°C.”

Digital information (e.g, facts about the donor) is attached to each sample, which results in a huge “omics” data storage.

Patient Records

Patient records, which store treatment histories of diagnoses and prognoses (implying clinical and laboratory values), are very useful for research and predicting outcomes of similar future patient cares. As the data was collected for a specific treatment, such records are usually incomplete (missing parameter values), incorrect (systematic or random noise in the data), sparse (few or non-representable records) or inexact (inappropriate selection of parameters for a specific task) for a different task [89]. These circumstances might require further data acquisitions (e.g., interviews) to increase quality of the data set for a specific task.

In addition to that, many physicians input a significant amount of information about their patients by typing free text. This text, commonly in native language, might contain keywords or natural language which is highly unstructured in the sense of data. Therefore, a strong motivation to extract data out of text automatically has been developed. This problem is being tackled by biomedical text mining [55].

Open Data

Another way to obtain useful information is public open data, which can be reused and redistributed freely by everyone [139]. Open data is a great opportunity to share medical data internationally for further research. However, as anonymized open data might be incomplete for a given task, there is no possibility to obtain further information about the data set [172].
All these heterogeneous data sources provide a huge and complex amount of information. There already exist approaches for translational medicine to bridge these sources to build an integrative knowledge base [108].

5.2 Data Preparation

5.2.1 Data Integration

The combination of multiple data sets is often necessary and the data formats tend to be as diverse as its sources. Therefore, data pre-processing, while protecting privacy, is needed to obtain a uniformly structured data set for performing further analysis. Each data source is likely to contain different records or some sources might be incomplete as discussed in Section 5.2.3. Values may be continuous or discrete, stored in varied dimensions or even be acquainted under different measurement standards and conditions. Such conditions imply technical and environmental aspects (e.g., used equipment, ambient temperature, etc.) and require particular data transformations [82, 48]. If these influences are not considered carefully, the combined data set might lead to harmful divergences of values and furthermore to distorted results of the performed analysis. Logically, the quality of the analysis directly depends on the quality of the analyzed data.

5.2.2 Protecting Anonymity

Biomedical data sets usually contain personal information which has to be protected by applying to ethical policies. Third parties must not be able to identify patients in a single data set or even by linking multiple accessible data sets combined with potential background knowledge (linkage attack). To emphasize sensitivity, linkage-relevant attributes are divided into identifiers and so-called quasi identifiers (QI) [79]. While pure identifiers uniquely identify a person, a combination of QIs is needed for a confident identification (e.g., ZIP-code and birthday). Therefore, its consideration is of high importance when publishing and promoting open data. There exist multiple approaches to achieve anonymity like anonymization and pseudonymization.

Anonymization describes, besides the removal of personal information, the fragmentation of attributes and addition of ambiguity to protect privacy while retaining the data's quality for performing knowledge discovery.

Pseudonymization replaces all identifiers with non-related pseudonyms or hashes. Another approach is the generalization of values (e.g., usage of the birth year instead of the exact date) which weakens identifiers efficiently but might influence the data quality for further research as well.

5.2.3 Data Cleansing

Data cleansing includes removing noise, handling and mapping missing values within the data set to achieve better quality in knowledge discovery. Therefore, data cleansing is an essential step and it might take up to 80% of the time of the overall process [31, 95]. Besides the general data cleansing tasks of the KDD process (see Section 2.1.1), missing data fields can be filled by performing further manual information acquisitions (e.g., interviews, phone calls, questionnaires). This approach does only make sense if the data set is not too large, since manual value filling is very time consuming and expensive. For larger data sets, automated methods are preferable and instead of using a constant value, a predictive density distribution for unobserved values can improve finding an appropriate substitutionary value. An example is Expectation Maximization (EM). It is a method to compute a fitting model to predict likely values for missing attributes based on observed ones [25, 120]. As data cleansing modifies the original data set, experts need to be aware of the fact, that any modification leads to a deviated interpretation of the data set.

5.3 Knowledge Discovery

Knowledge discovery implies the selection and application of data mining and machine learning algorithms to search for new patterns. Such patterns support experts to discover new knowledge and unknown relations within the data set. See Section 2.1 for general information about knowledge discovery in databases.

5.4 Visualization and User Interaction

The result of the applied algorithm has to be visualized in a comprehensible way to allow experts to investigate the discovered knowledge. The visualization system should offer sophisticated interaction methods to explore the data set and adjust granularity. The biomedical domain challenges visualizations in multiple ways. First, because of the trend to data-centric medicine, systems have to cope with huge, complex and multidimensional volumes which are likely to include unstructured and noisy data. Furthermore, precision medicine aims to integrate multiple data sources (e.g, "omics"-data, etc.) [147]. This fact dramatically increases complexity of the data set and adds an additional challenge for data analysts and appropriate visualizations. Many data analyses create a trained model or use statistical calculations to approximate relations within the data (see Section 2.1 for further information). As these models are approximations, the visualization should inform the user about the certainty of the presented result to avoid misinterpretations (see Section 4.5).

5.5 Supported Decision-making

Users and experts may use the discovered knowledge to make decisions for further actions or document the result. Generally, decision support systems represent extracted knowledge from the analyzed data, so it does not offer a complete solution for a given problem. The main expertise for making further decisions and solving problems is still the experts experience and knowledge [53, 134].

6 Systems Biology - Visualization of Omics Data

The term "omics" describes the combination of several research fields which are called *genomics*, *transcriptomics*, *proteomics* and *metabolomics* [57]. Lately, these research fields have advanced significantly due to high-throughput technologies such as *microarray technology* [51], *Next-Generation Sequencing* (NGS) [97] and *mass spectrometry* [2]. Due to these techniques, a vast amount of data has been generated and enables experts to perform detailed research.

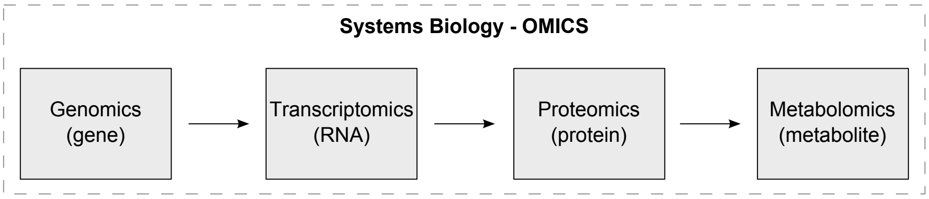
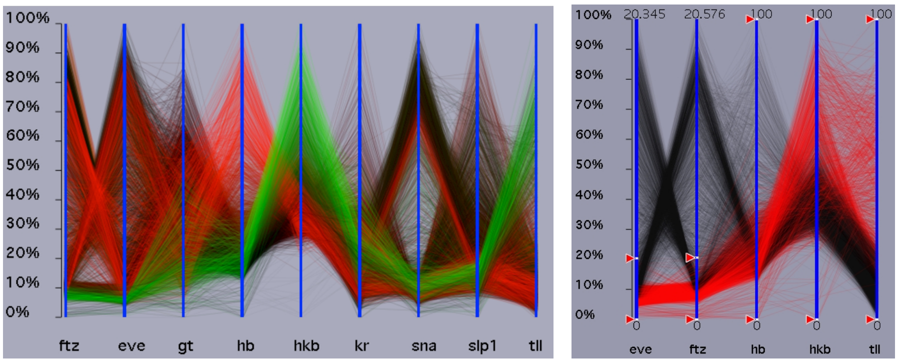


Figure 38: This figure illustrates relations between different types of omics data. Gene data (genomics) is transcribed to transcriptomics (RNA). RNA can be broken down to all proteins it consists of (proteomics) and each protein can be described by motabolites and its corresponding chemical process (metabolomics).

As depicted in Figure 38, all mentioned types of "omics" data depend on each other in a sequential manner. *Genomics* is the study of all genes in an organism and genes holding DNA can be transcribed to RNA transcripts. The study of RNA within an organism (*transcriptomics*) uses these transcripts to perform research on a more detailed level.

Systems biology is an approach to put all together and aims to understand interactions and relations between all "omics" research fields as a whole (high-dimensional biology). This complex approach supports experts to understand physiological and disease processes in a new way. It can improve personalized diagnosis, prognosis and drug therapies for patient treatment as well. In addition to that, it can also be used for biomarker discovery, drug discovery or developing new approaches to prevent or predict diseases [57].



(a) (b)

Figure 39: Two Parallel Coordinates plots depicting expression data for genes. (a) Depiction of expression data for 9 genes where *hkb* (green) and *hb* (orange) is highlighted. (b) A Parallel Coordinate plot comparing five genes. A configurable brush is used to highlight all cells expressing *eve* and *ftz* at a level between 0% and 20%. (Image source: Rübel et al. [125]).

Moreover, systems biology is about modeling biological relationships and interactions between proteins (proteomics), genes (genomics) and metabolites (metabolomics). Therefore, biological networks are established. Examples of common networks are:

- Gene regulatory networks
- Cell signaling networks
- Protein - protein interaction networks
- Metabolic networks

Multiple representations and data formats for biological networks exist. One of the most common formats for systems biology is *SBML* [59] (Systems Biology Markup Language) which is based on XML and able to represent metabolic networks, cell signaling networks etc. Another common representation format is *BioPax* (Biological Pathway Exchange) to depict molecular and cellular pathways [43] in a RDF/OWL-based format [24]. As networks are represented as graphs, graphs are still a hot research topic to improve visualizations and interactions [54].

6.1 Genomics

In general terms, genomics is the research field of genes and gene expressions (DNA). Microarray techniques are one of the key technologies which significantly advanced genomics. As microarray data sets usually are of high dimensionality, visualization methods for high dimensional data are used to depict the multivariate data. In addition to that, dimensionality reduction (see Section 2.4) may be applied to simplify the data set before using it for further analysis [157]. The most common visualization techniques are scatter plots, Parallel Coordinates plots [65, 30] and heat maps [42].

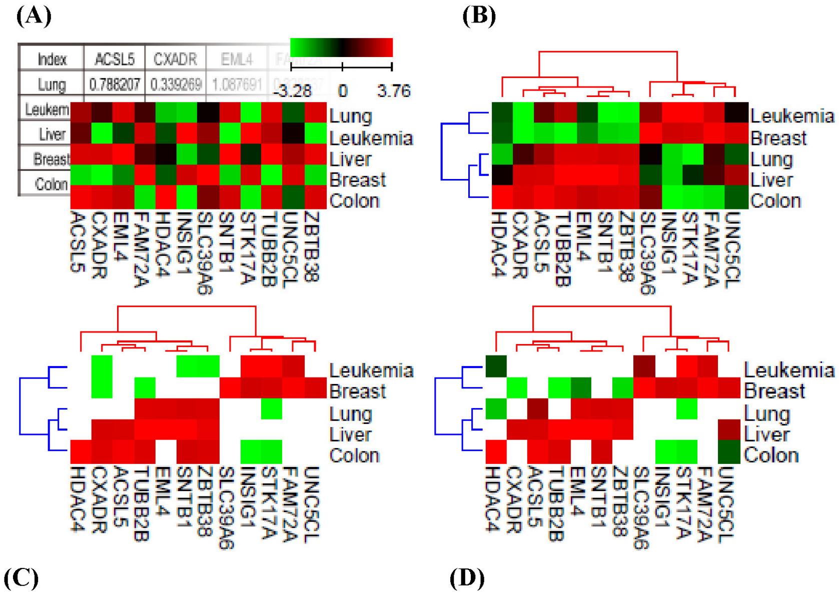


Figure 40: Illustration of heat maps depicting microarray data for 12 genes and 5 cancer samples. Up-regulated gene expressions are shown in red and down-regulated ones in green. (a) The input data is shown as a standard heat map. (b) Cancer samples (rows) and genes (columns) have been reordered by clustering. Adjacent dendrograms represent the cluster result. (c) Selective depiction of high and low expressions. (d) Selected depiction of genes controlled by a threshold value. (Image source: Kim et al. [80]).

As Gehlenborg et al. [42] mentions:

> "The initial goal in analyzing expression profiles is usually to find a set of genes or, less typically, proteins that share a related pattern of expression—for example, genes that are up- or down-regulated in a certain genotype, disease model or human disease, or in response to a drug treatment."

As shown in Figure 39, Parallel Coordinate plots are a flexible way to analyze multivariate gene data. It supports users to find correlations between samples and expression levels. Conditions (brushes) are used to highlight a specific subset of the data (see Figure 39b).

A disadvantage of the Parallel Coordinate plot is that the order of the axes influences the graphical representation significantly. To avoid too many intersections, a limited amount of samples may be used. Morover, quality metrics (see Section 4.3) can support the system to find a more preferred order.

Figure 40 shows various examples of using heat maps to analyse microarray gene expression data. A clustering of rows and columns leads to an ordered matrix, which simplifies the investigation of relations and values. In addition to that, threshold values can be used to hide uninteresting values and highlight a specific range of values [80].

As Weinstein in 2008 [162] mentions:

> "In the case of gene expression data, the color assigned to a point in the heat map grid indicates how much of a particular RNA or protein is expressed in a given sample. The gene expression level is generally indicated by red for high expression and either green or blue for low expression. Coherent patterns (patches) of color are generated by hierarchical clustering on both horizontal and vertical axes to bring like together with like. Cluster relationships are indicated by tree-like structures adjacent to the heat map, and the patches of color may indicate functional relationships among genes and samples."

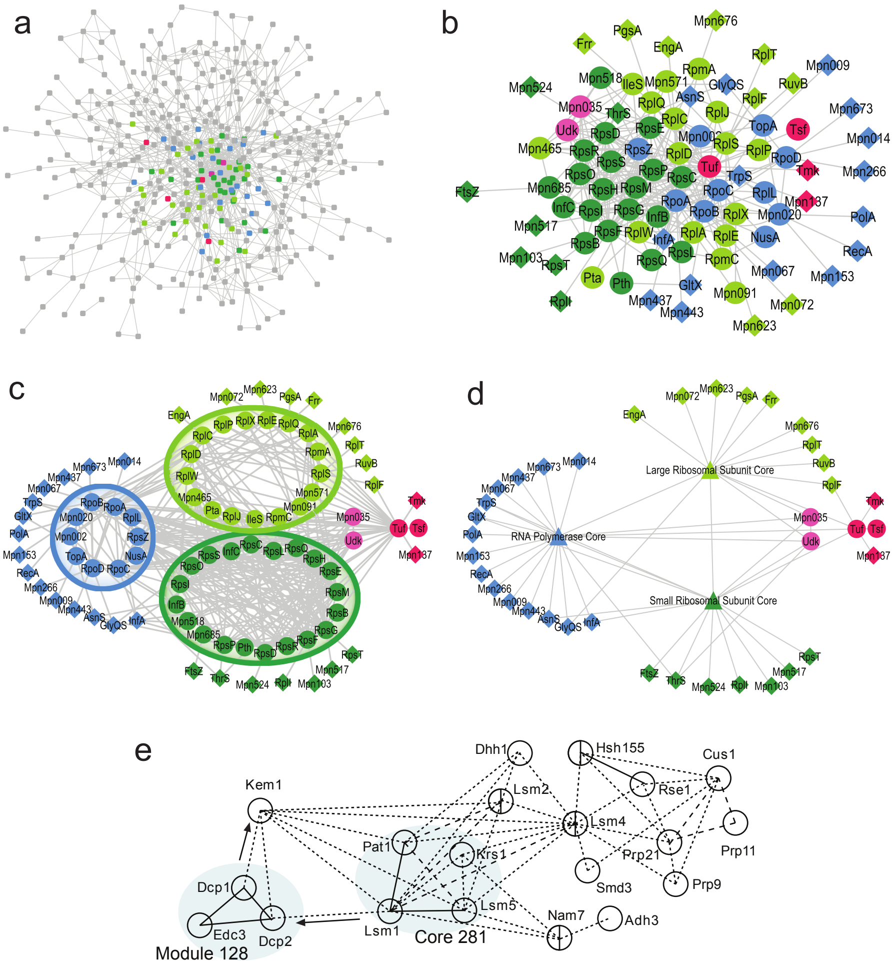


Figure 41: Examples of visualized protein interaction networks. (a) A protein interaction network with more than 400 proteins placed by using a force-directed algorithm. (b) Simplified graph by removing unimportant nodes. The shape of nodes depicts different roles. A circle represents a core protein while a diamond represents a non-core element. (c) Manual replacement of nodes of the network to emphasize structure and interactions. (d) All core nodes of one type have been collapsed to a single meta node to simplify the network (e) A representation of stages in deadenylation-dependent mRNA degradation. The order of the cellular process is shown by arrows and the color of nodes represents the localization of the associated protein. Shaded circles represent protein complexes and edge styles illustrate socia-affinity indices. (Image source: Gehlenborg et al. [42]).

6.2 Proteomics

An understanding of relations between proteins is essential in systems biology as biological processes of a cell are controlled by protein interactions. Data sets containing information about protein interactions are usually large and complex because a single protein can interact with up to several dozens proteins [118, 127].

Bu et al. states [14]:

> "It is believed that all biological processes are essentially and accurately carried out through protein–protein interactions."

As protein–protein interactions are usually visualized by graphs, a complete representation of all interactions is overwhelming for users. Therefore, tools try to visualize specific proteins or important subsets at a time (see Figure 41). Due to its high complexity, common tools use very different methods to visually represent such graphs (no standard method has been recognized yet) [14, 127]. A drawback of visualized protein interactions is the fact, that only already-known interactions can be visualized. If the underlying protein complex purification techniques (e.g., *mass spectrometry* [2], *correlated messenger RNA expression profiles* [60]) does not detect any interaction, it will not be visualized afterwards. However, protein networks can still be used to understand and to find biological functions by graph mining. For example, finding quasi-cliques or quasi-bipartites might reveal unknown knowledge [14].

6.3 Metabolomics

Metabolomics is about analyzing metabolites and their associated chemical reactions within a cell. To represent such chemical chain reactions, metabolic pathways are used. Such pathways are usually represented as acyclic graphs (see Figure 42).

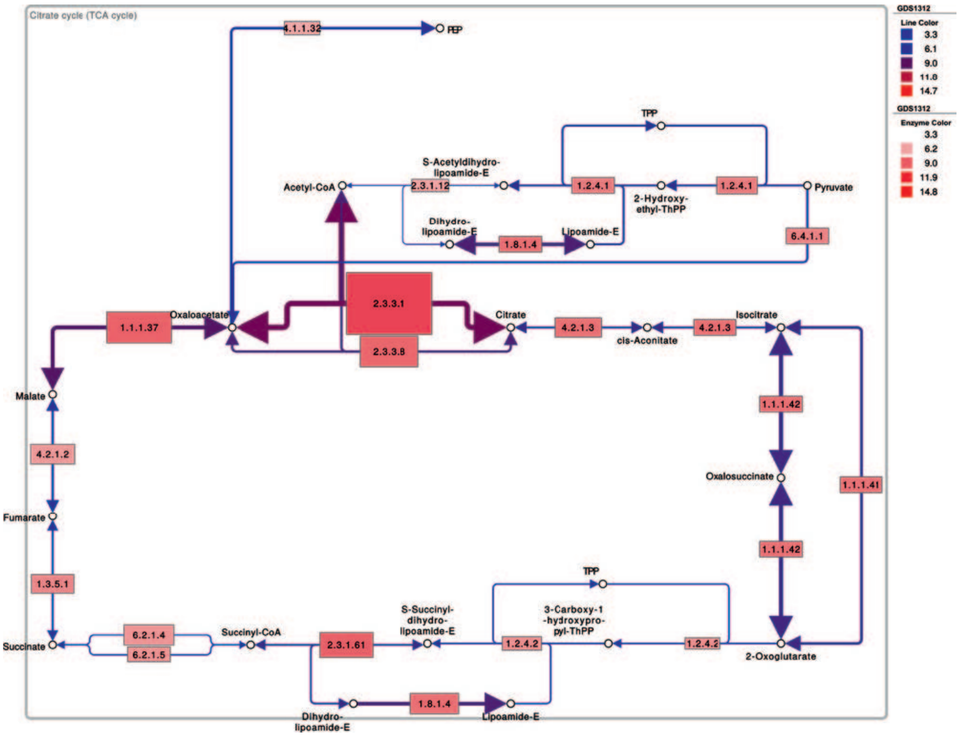


Figure 42: A visualization of the KEEG citrate cycle pathway. The corresponding gene expression data is additionally mapped to the nodes color, size and line thickness. (Image source: Gerasch et al. [43]).

Gehlenborg et al. [42] explains the general goal of metabolomics:

> "The general goal in analyzing metabolite profiles is to gain detailed insight into the molecular mechanisms of cellular metabolic pathways. The identification of molecules that may be used as reliable biomarkers of disease is also of great interest."

6.4 Open Problems and Outlook

A present issue is the fact, that many visual representations are not standardized and therefore, the user faces multiple visual representations of the same data. Variety of supports different view points, but standardized representations help a community to maintain the same common mental model. For

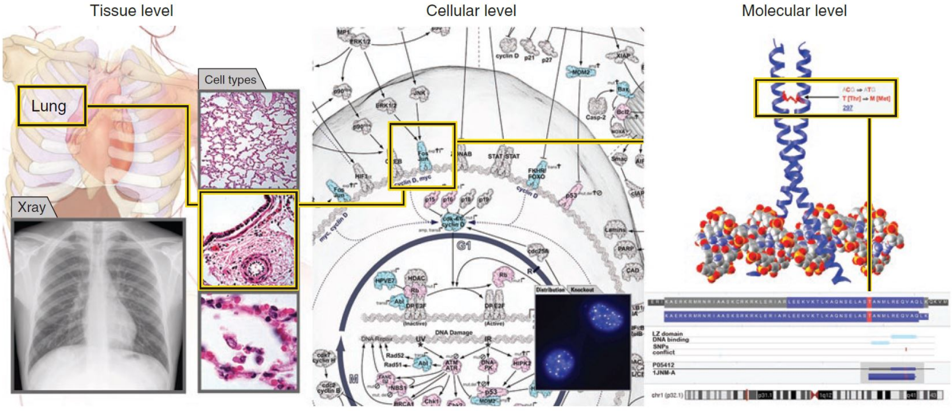


Figure 43: The ultimate goal of systems biology is to link heterogeneous data sets to support biologists to gain insight into the whole biological system. Such visualizations might depict X-ray scans, tissues, cellular and molecular data, genomes and metabolic pathways. (Image source: O'Donoghue et al. [110]).

example, several standardized representations exist for molecules nowadays (e.g., spheres connected by sticks). Another famous example is the double helix proposed by Watson and Crick to visualize DNA [160]. However, systems biology still lacks of well known representations for various other omics data types [110].

There exist many stand-alone tools to explore a specific type of data but it does not support the user to link the gained knowledge to other data sets [156]. Therefore, the ultimate goal of systems biology is to support biologists to gain insight into whole organisms by linking all abstraction levels to a single system (e.g., from organs to molecules). This can only be achieved by an integrative framework which combines several visualizations of interlinked heterogeneous data sets (see Figure 43).

First approaches do already exist. Ding et al. developed an integrative visual analytics system to analyze genomic based cancer. Figure 44 shows that the system combines multiple visualization techniques (graphs, Parallel Coordinates, heat maps) combined with unsupervised cluster analysis to compare clinical outcomes. It supports experts to explore multiple gene expression data sets to find biomarkers.

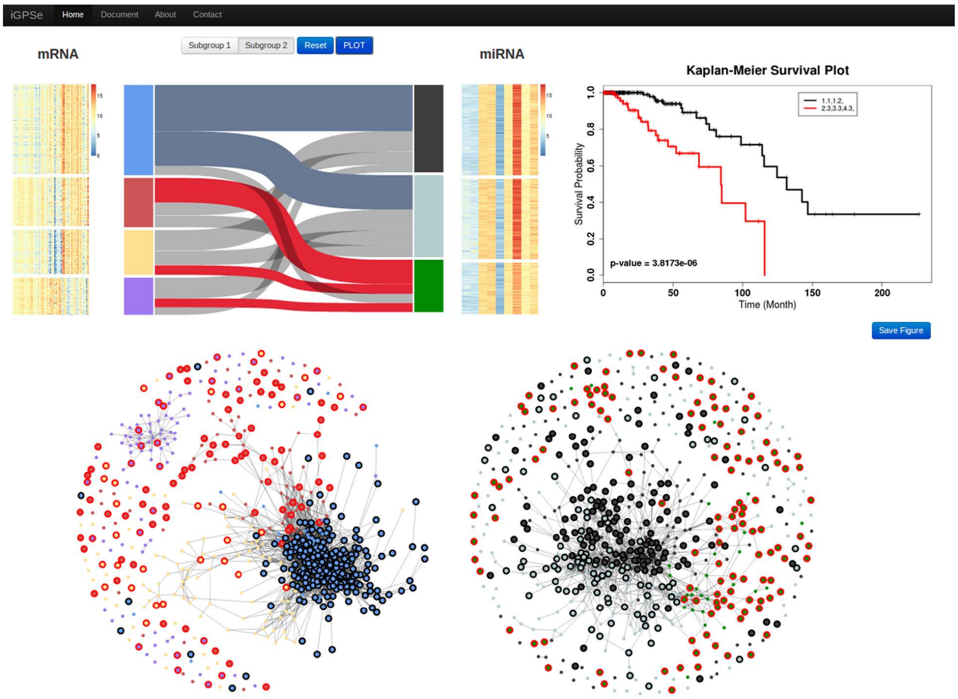


Figure 44: This visual analytics approach supports experts to find biomarkers interactively within data holding genetic information about patients who suffer from cancer. Clustered mRNA and miRNA expression data is visualized by heat maps in the top left corner. Between both heat maps, links are used to represent relations between grouped patient samples. In the top right corner, a plot depicts the probability of survival after a given time. Moreover, the statistical significance (p-value) is shown. At the bottom, force-directed graphs visualize similarities of patient records. (Image source: Ding et al. [28]).

7 State-of-the-art of Visual Analytics in Biomedical Domain

This analysis of 73 recent visual analytics papers is based on the state-of-the-art report of Turkay et al. [147]. It extends the given analysis by classifying all scientific papers into the categorizations *data type* and *visualization techniques*. Moreover, several additional visual analytics papers are included.

7.1 Dimensions and Categories

All papers are categorized into for 4 dimensions, where the first two are inherited from the analysis of Turkay et al. [147]:

- Type of analysis
- Level of integration
- Visualization technique
- Data type

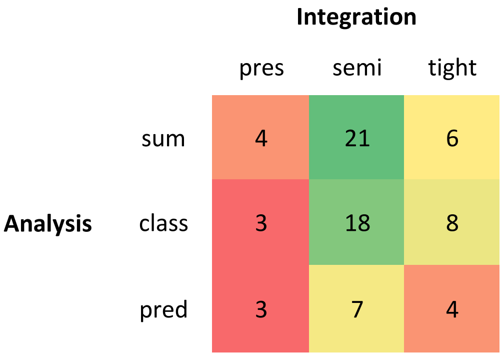
Each dimension is divided into the following subcategories:

Type of analysis: Summarizing information, groups & classification, dependence & prediction.

The type of analysis categorizes papers according to analytical task which the presented approach is supposed to carry out (see Section 4.2).

Level of integration: Visualization as presentation, semi-interactive methods, tight integration.

The level of integration describes how tightly computational tools and algorithms are integrated into the visual analytics system to enable the user to steer the automated analytical process (see Section 4.2).



		Integration		
		pres	semi	tight
Analysis	sum	4	21	6
	class	3	18	8
	pred	3	7	4

Figure 45: Integration level vs. type of analysis: Most visual analytics systems are of the integration level *semi-interactive methods* for both analysis task *summarizing information* and *groups & classification*. There is still a lack of prediction systems which tightly integrate the user.

Visualization technique: Geometric, table-based, icon/glyph-based, pixel-based, graph

Visualization techniques are categorized according to Keim et al. [76, 74] and in addition to that, the category *table-based* has been added to emphasize common table-based visualizations such as table lens and heat maps.

Data type: Genomics, proteomics, metabolomics, text, graph, image, multivariate data

Besides common data types in the biomedical domain (text, image), the category *data type* contains all main omics-data types (genomics, proteomics, metabolomics). For general and novel visual analytic approaches, which do not target the biomedical domain or graph analysis, the category *multivariate data* is used.

7.2 Classifications and Results

Several visualizations combine multiple visualization methods (brushing and linking) and therefore, a single paper might fall into several categories

	Visualization as Presentation	**Semi-interactive Methods**	**Tight Integration**
Summarizing Information	[24], [98], [109], [127]	[8], [12], [14], [16], [17], [38], [40], [41], [60], [66], [68], [72], [81], [88], [101], [111], [115], [140], [141], [161], [169], [170]	[33], [34], [64], [107], [146], [166]
Groups & Classification	[26], [84], [138]	[7], [27], [45], [46], [69], [80], [81], [90], [91], [99], [100], [112], [119], [122], [129], [130], [159], [174]	[3], [18], [28], [113], [124], [149], [150], [153]
Dependence & Prediction	[70], [87], [88]	[13], [32], [102], [103], [106], [116], [170]	[9], [28], [96], [148]

Table 1: Level of integration vs. type of analysis

at once. Similar circumstances occur when categorizing approaches by data type because several biomedical visualizations combine several omics data sources.

Level of Integration vs. Type of Analysis

The cross section of integration level and type of analysis reveals that most visual analytics solutions provide a semi-interactive method. In other words, it allows the user to configure computational tools before the actual visual exploration of the data. There is still a large lack of predictive and tightly integrated systems (see Figure 45 and Table 1).

Figure 46 shows, that there exists an increasing trend of performed research on highly integrated visual analytics systems.

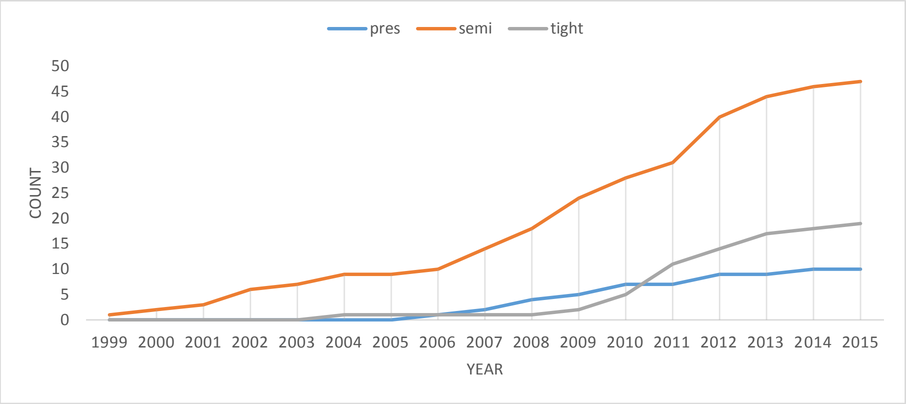
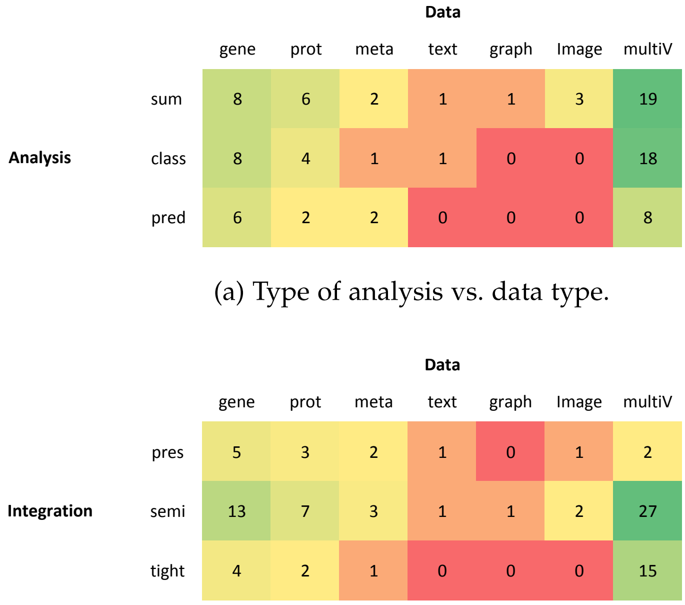


Figure 46: According to this analysis, research on highly integrated visual analytics approaches has been increased. For completeness, papers published in the current year 2015 are included but are not representative.

Data Type vs. Type of Analysis and Level of Integration

Cross intersections with the dimension *data type* reveal, that most novel approaches are designed for multivariate data sets (see Figure 47, Table 2 and Table 3). The has not been presented for the biomedical domain



		Data						
		gene	prot	meta	text	graph	Image	multiV
Analysis	sum	8	6	2	1	1	3	19
	class	8	4	1	1	0	0	18
	pred	6	2	2	0	0	0	8

(a) Type of analysis vs. data type.

		Data						
		gene	prot	meta	text	graph	Image	multiV
Integration	pres	5	3	2	1	0	1	2
	semi	13	7	3	1	1	2	27
	tight	4	2	1	0	0	0	15

(b) Integration level vs. data type.

Figure 47: Both cross intersections show, that most novel visual analytics systems are designed for general multivariate data sets. Regarding omics-data, genome and protein data sets are the most supported data types (due to its multivariate nature). Tightly integrated systems to mostly exist for general multivariate data sets. Moreover, there exists a lack of text and image analysis systems.

but can be applied on biomedical data sets. Data sets containing information about gene expressions and proteins are commonly multivariate data sets, therefore, a considerable amount of techniques do already exist. Tools which analyze metabolites use graphs to depict biological pathways. Therefore, techniques based on multivariate data sets can not be applied.

Level of Integration vs. Visualization Method

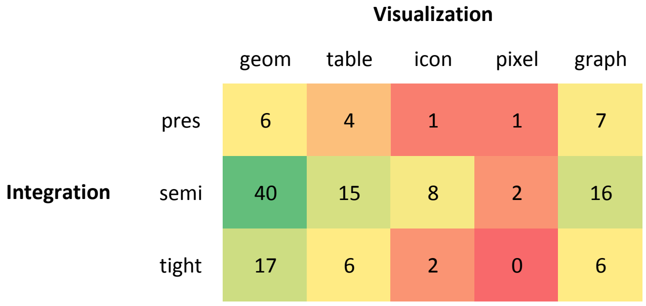
An analysis of used visualization techniques reveals, that the most common methods are geometric/projective visualizations. Common techniques are for example scatter plots and Parallel Coordinates (also called profile plots).

	Summarizing Information	**Groups & Classification**	**Dependence & Prediction**
Genomics	[16], [24], [60], [88], [101], [109], [140], [146]	[27], [28], [80], [90], [91], [124], [130], [174]	[28], [70], [87], [88], [102], [103]
Proteomics	[14], [16], [24], [40], [60], [127]	[28], [80], [112], [174]	[28], [70]
Metabolomics	[24], [60]	[112]	[70], [103]
Text	[40]	[138]	
Graph	[115]		
Image	[41], [98], [141]		
Multivariate	[8], [12], [17], [33], [34], [38], [41], [64], [66], [68], [72], [81], [107], [111], [115], [146], [161], [166], [169]	[3], [7], [18], [26], [46], [69], [81], [84], [90], [99], [113], [119], [122], [129], [149], [150], [153], [159]	[9], [13], [32], [96], [106], [116], [148], [170]

Table 2: Type of analysis vs. data type

	Visualization as Presentation	Semi-interactive Methods	Tight Integration
Genomics	[24], [70], [87], [88], [109]	[16], [27], [60], [80], [88], [90], [91], [101], [102], [103], [130], [140], [174]	[28], [43], [124], [146]
Proteomics	[24], [70], [127]	[14], [16], [40], [60], [80], [112], [174]	[28], [43]
Metabolomics	[24], [70]	[60], [103], [112]	[43]
Text	[138]	[40]	
Graph		[115]	
Image	[109]	[41], [141]	
Multivariate	[26], [84]	[7], [8], [12], [13], [17], [32], [38], [41], [46], [66], [68], [69], [72], [81], [90], [99], [106], [111], [115], [116], [119], [122], [129], [159], [161], [169], [170]	[3], [9], [18], [33], [34], [64], [96], [107], [113], [146], [148], [149], [150], [153], [166]

Table 3: Level of integration vs. data type



		Visualization				
		geom	table	icon	pixel	graph
	pres	6	4	1	1	7
Integration	semi	40	15	8	2	16
	tight	17	6	2	0	6

Figure 48: Integration level vs. visualization technique: There is a clear indicator, that a major part uses geometric visualization techniques (e.g., scatter plot). In addition to that, table-based (e.g., heat map)and graph-based techniques are common. Despite that huge data sets need to be visualized, pixel-based visualization methods are rarely used.

As shown in Figure 48 and in Table 4, especially tightly integrated systems integrate such methods often.

Besides geometric techniques discussed above, this analysis reveals that the table-based technique called *heat map* is one of the most common visualization technique for multivariate data.

These results conform to the statement of Gehlenborg et al. [42]:

> "Multivariate data, for instance from gene expression studies, are very common in systems biology, [...]. The three most commonly used visualization methods are scatter plots, profile plots and heat maps."

Last but not least, graphs are popular to depict found relations within the multivariate data set, represent hierarchies, analyzed text or biological pathways.

Commonly, visual analytics systems combine several visualization techniques (linking and brushing) to enhance the users insight. In this case, a single approach will be categorized into several visualization techniques.

	Visualization as Presentation	Semi-interactive Methods	Tight Integration
Geometric	[26], [70], [87], [88], [98], [109]	[8], [12], [13], [14], [16], [17], [27], [32], [38], [41], [45], [46], [60], [66], [68], [69], [72], [81], [88], [91], [99], [100], [102], [101], [103], [106], [111], [112], [115], [116], [119], [122], [129], [130], [140], [159], [161], [169], [170], [174]	[3], [9], [18], [28], [33], [34], [64], [96], [107], [113], [124], [146], [148], [149], [150], [153], [166]
Table-based	[70], [87], [88], [109]	[16], [27], [60], [80], [81], [88], [90], [91], [99], [102], [119], [130], [140], [161], [174]	[18], [28], [33], [96], [124], [149]
Icon-based	[88]	[16], [45], [88], [101], [122], [129], [140], [169]	[33], [107]
Pixel-based	[87]	[7], [169]	
Graph	[24], [26], [70], [84], [87], [127], [138]	[14], [27], [40], [60], [81], [90], [91], [102], [103], [112], [115], [119], [130], [159], [161], [170]	[3], [28], [43], [113], [149], [153]

Table 4: Level of integration vs. Visualization technique

8 An Implementation of a Clustered Heat Map

This section describes an implementation of a configurable clustered heat map for the Open Source project called Scaffold Hunter[10] [81, 163]. Scaffold Hunter (SH) is a visual analytics tool to explore the chemical space containing molecules which is commonly required for drug discovery. Scaffold Hunter is freely available under GNU GPL v3 and it is implemented in Java. The current version Scaffold Hunter 2.4.1 supports the following visualization methods:

- Scaffold tree
- Table
- Dendrogram
- Scatter plot (2D and 3D)
- Tree map
- Heat map

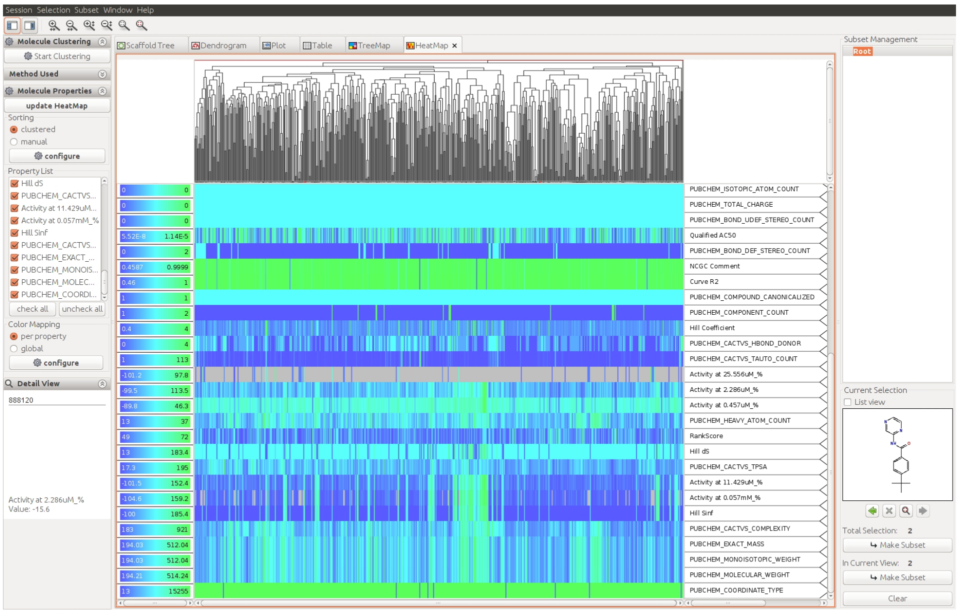
8.1 Framework and Architecture

Figure 50 shows that the architecture of Scaffold Hunter consists of three main parts:

- Data integration & management
- Analysis
- Interactive Visualization

Data Integration & Management As depicted in Figure 50, the layer *data integration & management* provides all functions to import and access the data. The central database (eg., MySQL, HSQLDB) is accessed via Hibernate, which abstracts the databases access by an object-relational mapping. Besides importing new data from various file sources (SDF, CSV and SQL), several calculations can be performed on the data. Such calculations might be a computation of fingerprints or a scaffold tree. The support of further

[10] Scaffold Hunter - `http://scaffoldhunter.sourceforge.net/`



Sorting
● clustered
○ manual

Color Mapping
● per property
○ global

888120

Activity at 2.286uM_%
Value: -15.6

□ List view

Total Selection: 2

In Current View: 2

Figure 49: An overview of the heat map view and its user interface implemented for the Open Source project Scaffold Hunter. In the center, the actual heat map visualizes a data array (molecules vs. molecule properties) by using a three-colored color mapping. Above the heat map, a dendrogram is shown to depict the hierarchy of clustered molecules (columns). The sidebar on the left provides functions to configure the clustering and heat map. Between the left sidebar and the actual heat map, a legend for each row is shown to depict the rows value range. To the right of the heat map, the name of a molecule property is shown in each appropriate row. User interactions such as zoom and panning are supported to explore the data set.

data sources and calculations can be easily extended by a flexible plugin system.

Analysis The central component manages data and creates subsets. Such sets are created by applying filters, performing a substructure search or by using the current selection defined by the user. A special type of sets is the result of a clustering. As the whole data set is often too large, subsets are important to support the user to refine the explorable data space. Moreover, for several internal data structures and calculations, the library *Chemistry Development Kit* is used.

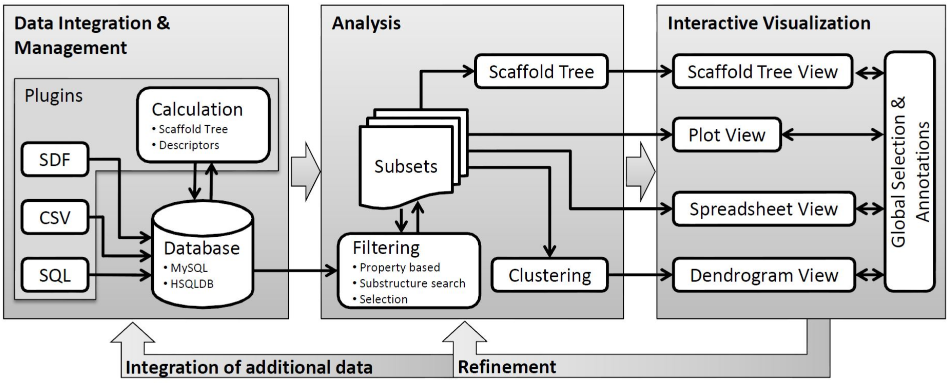


Figure 50: The architecture of Scaffold Hunter for visual analysis of chemical space. Several data formats can be used to import data into the central database. On the molecules data, further calculations can be performed. An integrated subset and filter management supports users to refine the currently investigated data set. To analyze the data visually, Scaffold Hunter offers a broad set of interactive visualizations to analyze the current data set. (Image source: Kriege [86])

Interactive Visualization This component provides a framework to manage and integrate new views and appropriate controls seamlessly to the current system. For large and zoomable user interfaces, the 2D scene graph library *Piccolo2D* is used. *Batik* provides functions to render and store SVG. In addition to that, the library *Guava* supports programmers to tackle common tasks (e.g., numerical functions, null checks, creation of complex structures, etc.).

The framework provides read-to-use functions for subset management and current selections (see right sidebar in Figure 49). A new visualization view only needs to implement all corresponding callbacks. As selection management is supported by the framework, a synchronization of selections between all active views is realized (linking and brushing).

The synchronization of selections is realized by an observable set. This set contains all currently selected molecules and when the set is being changed, all observers (views) will be notified. When a view is notified, it will highlight all selected molecules in its representation space. In case of a heat map, each column of the map represents one molecule and therefore, the corresponding column will be highlighted.

8.2 Scope of the Implementation

This section explains what I have implemented to realize an interactive clustered heat map for Scaffold Hunter. Section 8.1 describes that the architecture of SH consists of three main parts, namely *data integration & management*, *analysis* and *interactive visualization*. Besides several small code extensions, the whole implementation of the heat map is located in the latter layer. SH provides a generic framework to implement a new data view. Every view needs to inherit from the generic view class which manages all essential interfaces and callbacks for data access and global features to preserve consistency. Two essential global features are selection management (for linking and brushing) and subset management (to create subsets by using filters or manual selection). All controls for these two features are shown in the right sidebar of every data view (Figure 49). New views do not have to implement or adjust these functions. In contrast to the right sidebar, the left one is a placeholder for individual user interface elements to control the data view. Therefore, all required elements have to be implemented or ported from other views. The property list in the left sidebar and additional elements, which are needed to configure the heat map (e.g., color coding, sorting) are new components.

I ported the clustering configuration, clustering execution management and the rendering component (canvas) of the dendrogram view to render the dendrogram above the heat map. In addition to that, the clustering is also used to cluster properties and for that, several generalizations and dependency resolutions were needed.

All other components (between the left and right sidebar) are completely new. These parts include the actual heat map, a rendering of color legend for each row on the left side of the heat map and the rendering of property names (including an optional vertical dendrogram tree) on the right side.

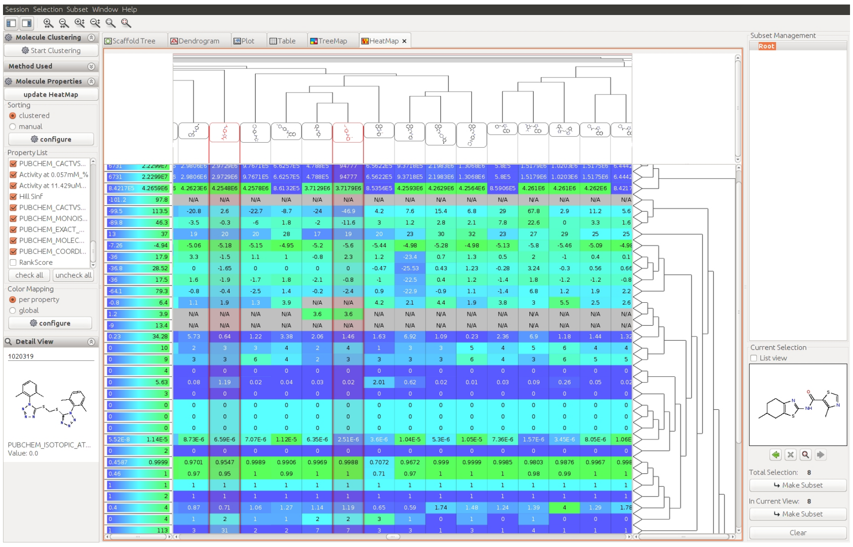


Figure 51: Semantic zoom: When zoomed in, values are rendered in each cell to show additional information.

8.3 Clustered Heat Map

8.3.1 Interactions

In addition to all user interactions mentioned in Figure 52, the clustered heat map supports the following user interactions:

- Panning
- Zoom
- Resizing
- Selection
- Tagging molecules
- Detail view and tooltip

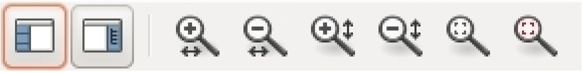
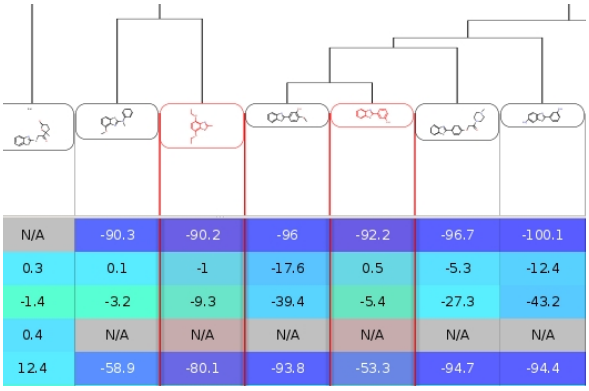


Figure 52: The heat map view supports several user interactions. From left to right: hide left and right sidebar, horizontal zoom in and zoom out, vertical zoom in and zoom out, zoom to overview, zoom to fit selection.



N/A	-90.3	-90.2	-96	-92.2	-96.7	-100.1
0.3	0.1	-1	-17.6	0.5	-5.3	-12.4
-1.4	-3.2	-9.3	-39.4	-5.4	-27.3	-43.2
0.4	N/A	N/A	N/A	N/A	N/A	N/A
12.4	-58.9	-80.1	-93.8	-53.3	-94.7	-94.4

Figure 53: Selections are highlighted as red dendrogram leaves and columns with red borders and a transparent red overlays. (Image source: Scaffold Hunter manual[11]).

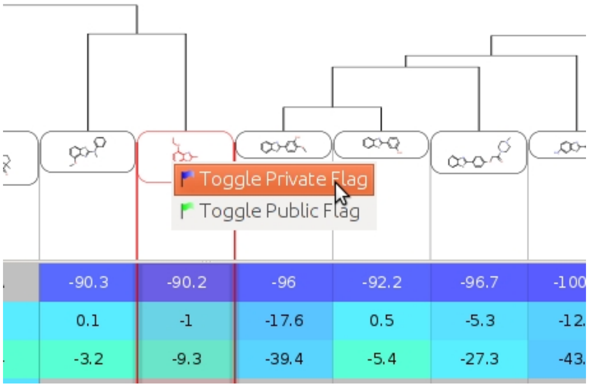
Panning Panning is important when the heat map is larger then the actual display. Horizontal and vertical panning can be performed by dragging the heat map or using the corresponding scroll bar. Vertical panning can also be carried out by using the mouse wheel on both adjacent legends on the left and right side.

Zoom By using the mouse wheel over the heat map, a horizontal zoom in or zoom out is performed. An additional press of the Ctrl-key performs a vertical zoom. Moreover, the dendrogram above the heat map performs a horizontal zoom when using the mouse wheel.

Figure 51 shows that the heat map uses *semantic zoom* to adjust the level of detail according to the zoom level.

Resizing Besides hiding the left and right panel, it is possible to resize the heat map canvas panel by moving the borders between heat map and dendrogram or legends. This enables the user to distribute the available space to several graphical elements.

[11] Scaffold Hunter - `http://scaffoldhunter.sourceforge.net/`



-90.3	-90.2	-96	-92.2	-96.7	-100
0.1	-1	-17.6	0.5	-5.3	-12.
-3.2	-9.3	-39.4	-5.4	-27.3	-43.

Figure 54: Scaffold Hunter supports users to tag single molecules with a flag to find it in other data views. (Image source: Scaffold Hunter manual[12]).

Selection In the heat map view, a molecule can be selected by clicking on a column or on a dendrogram node. The dendrogram also enables users to select a whole subtree of the dendrogram. As shown in Figure 53, selected molecules are highlighted as red dendrogram leaves and corresponding columns have additional red borders. Each selected column is additionally highlighted with a red overlay.

Tagging Molecules The framework of Scaffold Hunter provides a feature to tag molecules to support the user to recognize tagged molecules easily within other visualizations (linking And brushing). A molecule can only be tagged via the dendrogram view as shown in Figure 52.

Detail View and Tooltip A detail view of the currently hovered molecule is shown in the left sidebar. When pointing at a column of the heat map or a dendrograms node for 3 seconds, a tooltip window will be shown. It shows all properties of the current molecule and it is a global feature of Scaffold Hunter.

[12] Scaffold Hunter - `http://scaffoldhunter.sourceforge.net/`

8.3.2 Configuration

Sorting of columns and rows Before a heat map can be rendered, a clustering of all molecules of the selected subset has to be performed. For that, a clustering configuration is shown. The Scaffold Hunter framework provides several parameters to configure a clustering. Basic parameters are the type of clustering (normal exact clustering or heuristic clustering), the linkage (e.g., *complete linkage*, *group average linkage*, *single linkage*, etc.) and the distance function (e.g., *Euclide*, *Tanimoto*, *Jaccard*). Besides these parameters, a set of properties containing all properties which shall be used for distance calculation, need to be selected. This allows an even more specific clustering.

In addition to that, it is possible to configure the ordering of rows manually while columns have to be clustered anyway. All properties are shown in a list in the left side bar. It is possible to exclude individual properties to be shown as a row in the rendered heat map by unchecking it in the properties list. For manual ordering, items in the list can be moved and reorder by drag and drop.

Color Mapping The heat map view provides several possibilities to configure the color mapping function for each row individually (per property) or as a global function applied to each row. As shown in Figure 55, the assignment of an individual color mapping function to each heat map row enables the user to configure individual color ranges to highlight specific properties. The heat map supports three different mapping functions: a two-colored gradient mapping, a three-colored gradient mapping and an interval mapping.

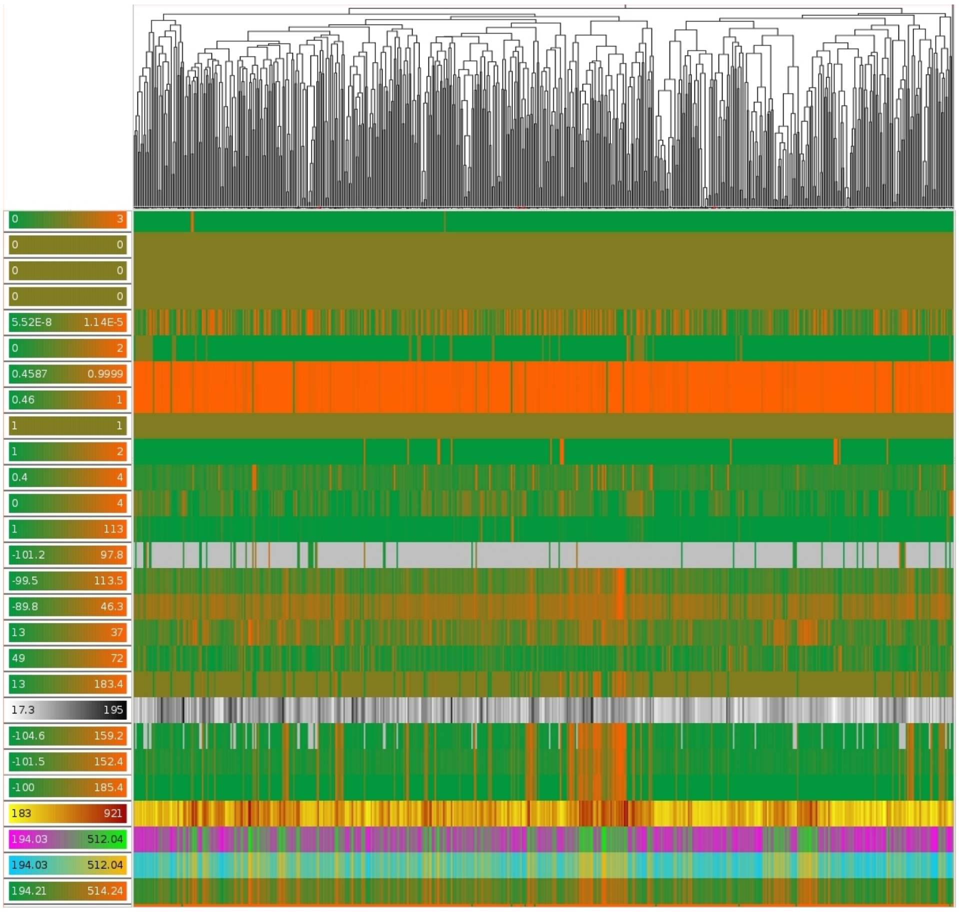


Figure 55: Individual color mapping functions are used to highlight specific properties in the heat map.

9 Contribution and Benefit

This master's thesis reviews state-of-the-art approaches for visual analytics in the biomedical domain. As visual analytics combines knowledge-discovery, visualization and human-computer-interaction, I examined each research field separately to collect broad and fundamental knowledge.

Knowledge Discovery: I reviewed the complete knowledge discovery process and analyzed its application in the biomedical domain. As data mining is the core technique of every knowledge discovery process, I discuss all related classes. Common data mining classes are clustering, classification, association rule mining, regression, summarization and searching of sequential patterns. As multivariate data usually is of very high dimensionality, dimensionality reduction is discussed as a pre-processing step. Moreover, biomedical data might contain personal information about patients and therefore, techniques for privacy protection (e.g., anonymization and pseudonymization) are essential. In addition to that, selected examples of practical knowledge discovery are discussed (decision trees and self-organizing maps).

Decision trees are an efficient and intuitive technique to perform classifications. Besides its high classification performance, it is easy to interpret by humans because decisions trees can be visualized as trees.

High-dimensional data sets always challenge low-dimensional visualizations to depict the data in a representative way. For that, a discussion about self-organizing maps shows a sophisticated approach to represent such data sets by two-dimensional visualizations.

This review of a knowledge discovery process supports experts to understand major obstacles and common techniques to design a sophisticated knowledge-based system.

Last but not least, I reviewed and discussed several commonly-used data mining tools.

Visual Analytics: The next major part of this thesis is a review of current visualization and visual analytics methods. For that, fundamental and psychological aspects, the design and architecture of visualizations are discussed. Psychological aspects include pre-attentive features such as color, size and other visual properties which generate a so-called "pop-out" effect to the human visual perception. The design of visualizations is generally conducted by Shneiderman's mantra "Overview first, Filter and zoom, Details on demand". Moreover, I discussed the influence of the users mental model on the final design of a visualization. As human-computer-interaction is also of high importance to improve the users insight, general interaction methods for visualization systems are discussed.

In addition to that, I discussed the iterative process visual analytics between the system and the user. The more the user can interact and steer the analytical process, the higher the integration of the user is. Quality metrics are a technique to judge visualization settings to support the automated visual analytics system to find highly relevant visualizations for the user.

Selected examples of visual analytics systems include a clustered heat map, interactive decision trees and a visual cluster analysis based on interactive Kohonen maps.

Analysis of state-of-the-art Papers: I performed an extended analysis of 73 state-of-the-art papers based on the review of Turkay et al. [147]. There is still a lack of highly integrative visual analytics systems – especially in the biomedical domain. However, the analysis shows an increasing trend of performed research in this field can be identified. The analysis shows clearly, that a major part of visual analytics systems is designed to analyze multivariate data. In addition to that, geometric, table-based and graph-based visualizations are the most common techniques to represent multivariate data. Regarding geometric techniques, the most common ones are scatter plots, Parallel Coordinates and heat maps are the most popular table-based visualization technique.

Analysis of Systems Biology: I analyzed the novel approach of bioinformatics, which aims to combine all omics-data types. The most common visualization for genomics and proteomics are geometric visualization techniques. For metabolomics, graph-based visualizations are used to depict biological pathways.

Implementation of a Clustered Heat Map: I implemented a clustered heat map for Scaffold Hunter to support experts to perform visual analytics on molecular data. In this thesis, I discussed the architecture and implementation while focusing on research field of visualization and interaction techniques. It is a semi-integrated approach, because the user configures the clustering and heat map before the actual heat map is shown. While exploring the data, the clustering result can not be changed but various interaction methods such as brushing-and-linking and semantic zoom are available.

10 Outlook

There is still a huge demand for specialized and highly integrative visual analytics approaches in the biomedical domain. Many highly integrative approaches are general approaches, but it can also be applied on particular sub-fields of bio-medicine. Therefore, there is a need of further research on specialized applications which integrate the users knowledge to the analytical process.

As many approaches support a single data type, there is even a larger lack of solutions, which integrate multiple data sets to analyze them in parallel. Based on this analysis, an even broader and more detailed investigation of current research would reveal how many systems already support multiple data sets.

As users input therapy outcomes as natural text and a lot of medical knowledge is located in books, the automated analysis of text is still a hot topic and needs further research. In addition to that, new approaches for graph analysis and graph mining are needed to analyze complex graphs (hairballs) in a comprehensible way.

However, systems biology aims to combine multiple data sets to analyze multiple layers of a biological system at once. The ultimate goal of such biomedical systems is to understand biological or pathological processes as a whole. Such a system would interlink all related data sets (e.g., images, text, measured values, scans) and offer visual analytics to support experts to explore the data while integrating personal domain knowledge (see Section 6.4). Such sophisticated visual analytics systems will boost evidence-based medicine to a new level.

List of Figures

References

[1] The r project for statistical computing. `http://www.r-project.org/`. Accessed: 2014-11-27.

[2] R. Aebersold and M. Mann. Mass spectrometry-based proteomics. *Nature*, 422(6928):198–207, 2003.

[3] Z. Ahmed and C. Weaver. An adaptive parameter space-filling algorithm for highly interactive cluster exploration. In *Visual Analytics Science and Technology (VAST), 2012 IEEE Conference on*, pages 13–22. IEEE, 2012.

[4] R. Alfred and D. Kazakov. Data summarization approach to relational domain learning based on frequent pattern to support the development of decision making. In *Advanced Data Mining and Applications*, pages 889–898. Springer, 2006.

[5] E. Alpaydin. *Introduction to machine learning*. MIT press, 2004.

[6] K. Andrews. Evaluating information visualisations. In *Proceedings of the 2006 AVI workshop on BEyond time and errors: novel evaluation methods for information visualization*, pages 1–5. ACM, 2006.

[7] M. Ankerst, C. Elsen, M. Ester, and H.-P. Kriegel. Visual classification: an interactive approach to decision tree construction. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 392–396. ACM, 1999.

[8] M. Behrisch, L. Shao, B. C. Kwon, T. Schreck, I. Sipiran, and D. Keim. Quality metrics driven approach to visualize multidimensional data in scatterplot matrix. 2014.

[9] W. Berger, H. Piringer, P. Filzmoser, and E. Gröller. Uncertainty-aware exploration of continuous parameter spaces using multivariate prediction. In *Computer Graphics Forum*, volume 30, pages 911–920. Wiley Online Library, 2011.

[10] M. R. Berthold, N. Cebron, F. Dill, T. R. Gabriel, T. Kötter, T. Meinl, P. Ohl, C. Sieb, K. Thiel, and B. Wiswedel. *KNIME: The Konstanz information miner*. Springer, 2008.

[11] E. Bertini and D. Lalanne. Investigating and reflecting on the integration of automatic data analysis and visualization in knowledge discovery. *ACM SIGKDD Explorations Newsletter*, 11(2):9–18, 2010.

[12] E. Bertini, A. Tatu, and D. Keim. Quality metrics in high-dimensional data visualization: an overview and systematization. *Visualization and Computer Graphics, IEEE Transactions on*, 17(12):2203–2212, 2011.

[13] M. Booshehrian, T. Möller, R. M. Peterman, and T. Munzner. Vismon: Facilitating analysis of trade-offs, uncertainty, and sensitivity in fisheries management decision making. In *Computer Graphics Forum*, volume 31, pages 1235–1244. Wiley Online Library, 2012.

[14] D. Bu, Y. Zhao, L. Cai, H. Xue, X. Zhu, H. Lu, J. Zhang, S. Sun, L. Ling, N. Zhang, et al. Topological structure analysis of the protein–protein interaction network in budding yeast. *Nucleic acids research*, 31(9):2443–2450, 2003.

[15] S. Carpendale. Evaluating information visualizations. In *Information Visualization*, pages 19–45. Springer, 2008.

[16] T. Carver, S. R. Harris, M. Berriman, J. Parkhill, and J. A. McQuillan. Artemis: an integrated platform for visualization and analysis of high-throughput sequence-based experimental data. *Bioinformatics*, 28(4):464–469, 2012.

[17] V. Chandola and V. Kumar. Summarization–compressing data into an informative representation. *Knowledge and Information Systems*, 12(3):355–378, 2007.

[18] J. Choo, H. Lee, J. Kihm, and H. Park. ivisclassifier: An interactive visual analytics system for classification based on supervised dimension reduction. In *Visual Analytics Science and Technology (VAST), 2010 IEEE Symposium on*, pages 27–34. IEEE, 2010.

[19] K. J. Cios and G. William Moore. Uniqueness of medical data mining. *Artificial intelligence in medicine*, 26(1):1–24, 2002.

[20] J. A. Claridge and T. C. Fabian. History and development of evidence-based medicine. *World journal of surgery*, 29(5):547–553, 2005.

[21] T. Cover and P. Hart. Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on*, 13(1):21–27, 1967.

[22] T. F. Cox and M. A. Cox. *Multidimensional scaling*. CRC Press, 2000.

[23] D. Defays. An efficient algorithm for a complete link method. *The Computer Journal*, 20(4):364–366, 1977.

[24] E. Demir, M. P. Cary, S. Paley, K. Fukuda, C. Lemer, I. Vastrik, G. Wu, P. D'Eustachio, C. Schaefer, J. Luciano, et al. The biopax community standard for pathway data sharing. *Nature biotechnology*, 28(9):935–942, 2010.

[25] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 1–38, 1977.

[26] J. Demšar, G. Leban, and B. Zupan. Freeviz—an intelligent multivariate visualization approach to explorative analysis of biomedical data. *Journal of biomedical informatics*, 40(6):661–671, 2007.

[27] J. Dietzsch, N. Gehlenborg, and K. Nieselt. Mayday-a microarray data analysis workbench. *Bioinformatics*, 22(8):1010–1012, 2006.

[28] H. Ding, C. Wang, K. Huang, and R. Machiraju. igpse: A visual analytic system for integrative genomic based cancer patient stratification. *BMC bioinformatics*, 15(1):203, 2014.

[29] A. Dix. *Human-computer interaction*. Springer, 2009.

[30] M. d'Ocagne. Coordonnèes parallèles et axiales. 1885.

[31] A. Duhamel, M. Nuttens, P. Devos, M. Picavet, and R. Beuscart. A preprocessing method for improving data mining techniques. application to a large medical diabetes database. *Studies in health technology and informatics*, 95:269–274, 2002.

[32] N. Elmqvist, P. Dragicevic, and J.-D. Fekete. Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. *Visualization and Computer Graphics, IEEE Transactions on*, 14(6):1539–1148, 2008.

[33] 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.

[34] A. Endert, C. Han, D. Maiti, L. House, S. Leman, 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.

[35] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. From data mining to knowledge discovery in databases. *AI magazine*, 17(3):37, 1996.

[36] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. The kdd process for extracting useful knowledge from volumes of data. *Commun. ACM*, 39(11):27–34, Nov. 1996.

[37] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy. Advances in knowledge discovery and data mining. 1996.

[38] S. J. 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. IEEE, 2011.

[39] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188, 1936.

[40] A. Franceschini, D. Szklarczyk, S. Frankild, M. Kuhn, M. Simonovic, A. Roth, J. Lin, P. Minguez, P. Bork, C. von Mering, et al. String v9. 1: protein-protein interaction networks, with increased coverage and integration. *Nucleic acids research*, 41(D1):D808–D815, 2013.

[41] R. Fuchs, J. Waser, and M. E. Groller. Visual human+ machine learning. *Visualization and Computer Graphics, IEEE Transactions on*, 15(6):1327–1334, 2009.

[42] N. Gehlenborg, S. I. O'Donoghue, N. S. Baliga, A. Goesmann, M. A. Hibbs, H. Kitano, O. Kohlbacher, H. Neuweger, R. Schneider, D. Tenenbaum, et al. Visualization of omics data for systems biology. *Nature methods*, 7:S56–S68, 2010.

[43] A. Gerasch, J. Küntzer, P. Niermann, D. Stöckel, M. Kaufmann, O. Kohlbacher, and H.-P. Lenhof. Network-based interactive navigation and analysis of large biological datasets. *it-Information Technology*, 57(1):37–48, 2015.

[44] S. Grossberg. Nonlinear neural networks: Principles, mechanisms, and architectures. *Neural networks*, 1(1):17–61, 1988.

[45] S. Grottel, G. Reina, J. Vrabec, and T. Ertl. Visual verification and analysis of cluster detection for molecular dynamics. *Visualization and Computer Graphics, IEEE Transactions on*, 13(6):1624–1631, 2007.

[46] Z. Guo, M. O. Ward, and E. A. Rundensteiner. Model space visualization for multivariate linear trend discovery. In *Visual Analytics Science and Technology, 2009. VAST 2009. IEEE Symposium on*, pages 75–82. IEEE, 2009.

[47] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.

[48] J. Han and M. Kamber. *Data Mining, Southeast Asia Edition: Concepts and Techniques*. Morgan kaufmann, 2006.

[49] C. D. Hansen and C. R. Johnson. *The visualization handbook*. Academic Press, 2005.

[50] M. A. Hearst, S. T. Dumais, E. Osman, J. Platt, and B. Scholkopf. Support vector machines. *Intelligent Systems and their Applications, IEEE*, 13(4):18–28, 1998.

[51] M. J. Heller. Dna microarray technology: devices, systems, and applications. *Annual review of biomedical engineering*, 4(1):129–153, 2002.

[52] A. Holzinger. Human-computer interaction and knowledge discovery (hci-kdd): What is the benefit of bringing those two fields to work together? In A. Cuzzocrea, C. Kittl, D. Simos, E. Weippl, and L. Xu, editors, *Availability, Reliability, and Security in Information Systems and HCI*, volume 8127 of *Lecture Notes in Computer Science*, pages 319–328. Springer Berlin Heidelberg, 2013.

[53] A. Holzinger and I. Jurisica. *Interactive Knowledge Discovery and Data Mining in Biomedical Informatics: State-of-the-Art and Future Challenges*, volume 8401. Springer Berlin Heidelberg, 2014.

[54] A. Holzinger, B. Ofner, and M. Dehmer. Multi-touch graph-based interaction for knowledge discovery on mobile devices: State-of-the-art and future challenges. In *Interactive Knowledge Discovery and Data Mining in Biomedical Informatics*, pages 241–254. Springer, 2014.

[55] A. Holzinger, J. Schantl, M. Schroettner, C. Seifert, and K. Verspoor. Biomedical text mining: State-of-the-art, open problems and future challenges. In *Interactive Knowledge Discovery and Data Mining in Biomedical Informatics*, pages 271–300. Springer Berlin Heidelberg, 2014.

[56] L. Hood, R. Balling, and C. Auffray. Revolutionizing medicine in the 21st century through systems approaches. *Biotechnology journal*, 7(8):992–1001, 2012.

[57] R. P. Horgan and L. C. Kenny. 'omic' technologies: genomics, transcriptomics, proteomics and metabolomicssbml):. *The Obstetrician & Gynaecologist*, 13(3):189–195, 2011.

[58] D. Howe, M. Costanzo, P. Fey, T. Gojobori, L. Hannick, W. Hide, D. P. Hill, R. Kania, M. Schaeffer, S. St Pierre, et al. Big data: The future of biocuration. *Nature*, 455(7209):47–50, 2008.

[59] M. Hucka, A. Finney, H. M. Sauro, H. Bolouri, J. C. Doyle, H. Kitano, A. P. Arkin, B. J. Bornstein, D. Bray, A. Cornish-Bowden, et al. The systems biology markup language (sbml): a medium for representation and exchange of biochemical network models. *Bioinformatics*, 19(4):524–531, 2003.

[60] T. R. Hughes, M. J. Marton, A. R. Jones, C. J. Roberts, R. Stoughton, C. D. Armour, H. A. Bennett, E. Coffey, H. Dai, Y. D. He, et al. Functional discovery via a compendium of expression profiles. *Cell*, 102(1):109–126, 2000.

[61] A. B. Hui, M. Lenarduzzi, T. Krushel, L. Waldron, M. Pintilie, W. Shi, B. Perez-Ordonez, I. Jurisica, B. O'Sullivan, J. Waldron, et al. Comprehensive microrna profiling for head and neck squamous cell carcinomas. *Clinical Cancer Research*, 16(4):1129–1139, 2010.

[62] B. Huppertz and A. Holzinger. Biobanks — a source of large biological data sets: Open problems and future challenges. In A. Holzinger and I. Jurisica, editors, *Interactive Knowledge Discovery and Data Mining in Biomedical Informatics*, volume 8401 of *Lecture Notes in Computer Science*, pages 317–330. Springer Berlin Heidelberg, 2014.

[63] A. Hyvärinen, J. Karhunen, and E. Oja. *Independent component analysis*, volume 46. John Wiley & Sons, 2004.

[64] S. Ingram, T. Munzner, V. Irvine, M. Tory, S. Bergner, and T. Moller. Dimstiller: Workflows for dimensional analysis and reduction. In *Visual Analytics Science and Technology (VAST), 2010 IEEE Symposium on*, pages 3–10. IEEE, 2010.

[65] A. Inselberg. The plane with parallel coordinates. *The Visual Computer*, 1(2):69–91, 1985.

[66] H. Janicke, M. Bottinger, and G. Scheuermann. Brushing of attribute clouds for the visualization of multivariate data. *Visualization and Computer Graphics, IEEE Transactions on*, 14(6):1459–1466, 2008.

[67] B. Jiang and Z. Li. Geovisualization: design, enhanced visual tools and applications. *The Cartographic Journal*, 42(1):3–4, 2005.

[68] 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.

[69] 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.

[70] J. R. Karr, J. C. Sanghvi, D. N. Macklin, M. V. Gutschow, J. M. Jacobs, B. Bolival, N. Assad-Garcia, J. I. Glass, and M. W. Covert. A whole-cell computational model predicts phenotype from genotype. *Cell*, 150(2):389–401, 2012.

[71] L. Kaufman and P. J. Rousseeuw. *Finding groups in data: an introduction to cluster analysis*, volume 344. John Wiley & Sons, 2009.

[72] J. Kehrer, P. Filzmoser, and H. Hauser. Brushing moments in interactive visual analysis. In *Computer Graphics Forum*, volume 29, pages 813–822. Wiley Online Library, 2010.

[73] D. Keim, G. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer, and G. Melançon. *Visual analytics: Definition, process, and challenges*. Springer, 2008.

[74] D. A. Keim. Visual exploration of large data sets. *Communications of the ACM*, 44(8):38–44, 2001.

[75] D. A. Keim, J. Kohlhammer, G. Ellis, and F. Mansmann. *Mastering The Information Age-Solving Problems with Visual Analytics*. Florian Mansmann, 2010.

[76] D. A. Keim and H.-P. Kriegel. Visualization techniques for mining large databases: A comparison. *Knowledge and Data Engineering, IEEE Transactions on*, 8(6):923–938, 1996.

[77] D. A. Keim, F. Mansmann, J. Schneidewind, J. Thomas, and H. Ziegler. *Visual analytics: Scope and challenges*. Springer, 2008.

[78] A. Kerren, A. Ebert, and J. Meyer. *Human-centered visualization environments*. Springer-Verlag, 2006.

[79] P. Kieseberg, H. Hobel, S. Schrittwieser, E. Weippl, and A. Holzinger. Protecting anonymity in data-driven biomedical science. In *Interactive Knowledge Discovery and Data Mining in Biomedical Informatics*, pages 301–316. Springer Berlin Heidelberg, 2014.

[80] N. Kim, H. Park, N. He, H. Y. Lee, and S. Yoon. Qcanvas: an advanced tool for data clustering and visualization of genomics data. *Genomics & informatics*, 10(4):263–265, 2012.

[81] K. Klein, N. Kriege, and P. Mutzel. Scaffold hunter: facilitating drug discovery by visual analysis of chemical space. In *Computer Vision, Imaging and Computer Graphics. Theory and Application*, pages 176–192. Springer, 2013.

[82] M. Kobayashi. Resources for studying statistical analysis of biomedical data and r. In A. Holzinger and I. Jurisica, editors, *Interactive Knowledge Discovery and Data Mining in Biomedical Informatics*, volume 8401 of *Lecture Notes in Computer Science*, pages 183–195. Springer Berlin Heidelberg, 2014.

[83] T. Kohonen. The self-organizing map. 78(9):1464–1480, 1990.

[84] R. Kosara, F. Bendix, and H. Hauser. Parallel sets: Interactive exploration and visual analysis of categorical data. *Visualization and Computer Graphics, IEEE Transactions on*, 12(4):558–568, 2006.

[85] E. F. Krause. Taxicab geometry. *The Mathematics Teacher*, pages 695–706, 1973.

[86] N. Kriege. Visual analysis of chemical space with scaffold hunter. `http://www.opentox.org/data/documents/development/meeting/opentoxeuro2013_submission_8.pdf`, 2013. Accessed: 2015-04-15.

[87] M. Krzywinski, J. Schein, I. Birol, J. Connors, R. Gascoyne, D. Horsman, S. J. Jones, and M. A. Marra. Circos: an information aesthetic for comparative genomics. *Genome research*, 19(9):1639–1645, 2009.

[88] R. M. Kuhn, D. Haussler, and W. J. Kent. The ucsc genome browser and associated tools. *Briefings in bioinformatics*, page bbs038, 2012.

[89] N. Lavrač. Selected techniques for data mining in medicine. *Artificial intelligence in medicine*, 16(1):3–23, 1999.

[90] A. Lex, M. Streit, C. Partl, K. Kashofer, and D. Schmalstieg. Comparative analysis of multidimensional, quantitative data. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1027–1035, 2010.

[91] A. Lex, M. Streit, H.-J. Schulz, C. Partl, D. Schmalstieg, P. J. Park, and N. Gehlenborg. Stratomex: Visual analysis of large-scale heterogeneous genomics data for cancer subtype characterization. In *Computer Graphics Forum*, volume 31, pages 1175–1184. Wiley Online Library, 2012.

[92] L. Liu and M. T. Zsu. *Encyclopedia of database systems*. Springer Publishing Company, Incorporated, 2009.

[93] B. L. W. H. Y. Ma. Integrating classification and association rule mining. In *Proceedings of the 4th*, 1998.

[94] L. Majnarić-Trtica and B. Vitale. Systems biology as a conceptual framework for research in family medicine; use in predicting response to influenza vaccination. *Primary health care research & development*, 12(04):310–321, 2011.

[95] J. I. Maletic and A. Marcus. Data cleansing: A prelude to knowledge discovery. In *Data Mining and Knowledge Discovery Handbook*, pages 19–32. Springer, 2010.

[96] A. Malik, R. Maciejewski, N. Elmqvist, Y. Jang, D. S. Ebert, and W. Huang. A correlative analysis process in a visual analytics environment. In *Visual Analytics Science and Technology (VAST), 2012 IEEE Conference on*, pages 33–42. IEEE, 2012.

[97] E. R. Mardis. The impact of next-generation sequencing technology on genetics. *Trends in genetics*, 24(3):133–141, 2008.

[98] M. E. Martone, J. Tran, W. W. Wong, J. Sargis, L. Fong, S. Larson, S. P. Lamont, A. Gupta, and M. H. Ellisman. The cell centered database project: an update on building community resources for managing and sharing 3d imaging data. *Journal of structural biology*, 161(3):220–231, 2008.

[99] 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.

[100] 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.

[101] M. Meyer, T. Munzner, A. DePace, and H. Pfister. Multeesum: A tool for comparative spatial and temporal gene expression data. *IEEE transactions on visualization and computer graphics*, 16(6):908, 2010.

[102] M. Meyer, T. Munzner, and H. Pfister. Mizbee: a multiscale synteny browser. *Visualization and Computer Graphics, IEEE Transactions on*, 15(6):897–904, 2009.

[103] M. Meyer, B. Wong, M. Styczynski, T. Munzner, and H. Pfister. Pathline: A tool for comparative functional genomics. In *Computer Graphics Forum*, volume 29, pages 1043–1052. Wiley Online Library, 2010.

[104] C. Minard. "minard" by charles minard, 1781-1870. Licensed under Public domain via Wikimedia Commons.

[105] R. Mirnezami, J. Nicholson, and A. Darzi. Preparing for precision medicine. *New England Journal of Medicine*, 366(6):489–491, 2012. PMID: 22256780.

[106] T. Muhlbacher and H. Piringer. A partition-based framework for building and validating regression models. *Visualization and Computer Graphics, IEEE Transactions on*, 19(12):1962–1971, 2013.

[107] J. E. Nam and K. Mueller. Tripadvisor^{ND}: A tourism-inspired high-dimensional space exploration framework with overview and detail. *Visualization and Computer Graphics, IEEE Transactions on*, 19(2):291–305, 2013.

[108] H. Nguyen, J. D. Thompson, P. Schutz, and O. Poch. biological knowledge bases: bridging genomics, integrative biology and translational

medicine. In *Interactive Knowledge Discovery and Data Mining in Biomedical Informatics*, pages 255–270. Springer, 2014.

[109] C. B. Nielsen, M. Cantor, I. Dubchak, D. Gordon, and T. Wang. Visualizing genomes: techniques and challenges. *Nature methods*, 7:S5–S15, 2010.

[110] S. I. O'Donoghue, A.-C. Gavin, N. Gehlenborg, D. S. Goodsell, J.-K. Hériché, C. B. Nielsen, C. North, A. J. Olson, J. B. Procter, D. W. Shattuck, et al. Visualizing biological data – now and in the future. *Nature methods*, 7:S2–S4, 2010.

[111] 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.

[112] C. Partl, A. Lex, M. Streit, D. Kalkofen, K. Kashofer, and D. Schmalstieg. enroute: Dynamic path extraction from biological pathway maps for in-depth experimental data analysis. In *Biological Data Visualization (BioVis), 2012 IEEE Symposium on*, pages 107–114. IEEE, 2012.

[113] J. Parulek, C. Turkay, N. Reuter, and I. Viola. Visual cavity analysis in molecular simulations. *BMC bioinformatics*, 14(Suppl 19):S4, 2013.

[114] W. Peng, M. O. Ward, and E. A. Rundensteiner. Clutter reduction in multi-dimensional data visualization using dimension reordering. In *Information Visualization, 2004. INFOVIS 2004. IEEE Symposium on*, pages 89–96. IEEE, 2004.

[115] A. Perer and B. Shneiderman. Integrating statistics and visualization for exploratory power. 2009.

[116] H. Piringer, W. Berger, and J. Krasser. Hypermoval: Interactive visual validation of regression models for real-time simulation. In *Computer Graphics Forum*, volume 29, pages 983–992. Wiley Online Library, 2010.

[117] P. Pudil and J. Novovičová. Novel methods for feature subset selection with respect to problem knowledge. In *Feature Extraction, Construction and Selection*, pages 101–116. Springer, 1998.

[118] R. Rameshwari and T. Prasad. Systematic and integrative analysis of proteomic data using bioinformatics tools. *arXiv preprint arXiv:1211.2743*, 2012.

[119] M. Rasmussen and G. Karypis. gcluto: An interactive clustering, visualization, and analysis system. *UMN-CS TR-04*, 21, 2004.

[120] A. R. Razavi, H. Gill, H. Åhlfeldt, and N. Shahsavar. A data preprocessing method to increase efficiency and accuracy in data mining. In *Artificial Intelligence in Medicine*, pages 434–443. Springer Berlin Heidelberg, 2005.

[121] K. Rexer. Rexer analytics 2013 data miner survey. `http://www.rexeranalytics.com/Data-Miner-Survey-2013-Intro.html`, 2013. Accessed: 2014-11-27, report requested per e-mail.

[122] 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(3-4):225–239, 2008.

[123] I. Rish. An empirical study of the naive bayes classifier. In *IJCAI 2001 workshop on empirical methods in artificial intelligence*, volume 3, pages 41–46. IBM New York, 2001.

[124] O. Rubel, G. H. Weber, M.-Y. Huang, E. W. Bethel, M. D. Biggin, C. C. Fowlkes, C. L. Luengo Hendriks, S. V. Keranen, M. B. Eisen, D. W. Knowles, et al. 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.

[125] O. Rübel, G. H. Weber, S. V. Keränen, C. C. Fowlkes, C. L. L. Hendriks, L. Simirenko, N. Shah, M. B. Eisen, M. D. Biggin, H. Hagen, et al. Pointcloudxplore: visual analysis of 3d gene expression data using physical views and parallel coordinates. In *EuroVis*, pages 203–210, 2006.

[126] S. R. Safavian and D. Landgrebe. A survey of decision tree classifier methodology. 1990.

[127] G. A. Salazar, A. Meintjes, G. Mazandu, H. A. Rapanoël, R. O. Akinola, and N. J. Mulder. A web-based protein interaction network visualizer. *BMC bioinformatics*, 15(1):129, 2014.

[128] M. Schoenauer, R. Akrour, M. Sebag, and J.-c. Souplet. Programming by feedback. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1503–1511, 2014.

[129] T. Schreck, J. Bernard, T. Von Landesberger, and J. Kohlhammer. Visual cluster analysis of trajectory data with interactive kohonen maps. *Information Visualization*, 8(1):14–29, 2009.

[130] J. Seo and B. Shneiderman. Interactively exploring hierarchical clustering results [gene identification]. *Computer*, 35(7):80–86, 2002.

[131] 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.

[132] B. Shoemate. Einstein never said that... `http://www.benshoemate.com/2008/11/30/einstein-never-said-that`. Accessed: 2014-12-10.

[133] R. Sibson. Slink: an optimally efficient algorithm for the single-link cluster method. *The Computer Journal*, 16(1):30–34, 1973.

[134] I. Sim, P. Gorman, R. A. Greenes, R. B. Haynes, B. Kaplan, H. Lehmann, and P. C. Tang. Clinical decision support systems for the practice of evidence-based medicine. *Journal of the American Medical Informatics Association*, 8(6):527–534, 2001.

[135] R. R. Sokal. A statistical method for evaluating systematic relationships. *Univ Kans Sci Bull*, 38:1409–1438, 1958.

[136] StatSoft. Statistica. `http://www.statsoft.com/`. Accessed: 2014-11-30.

[137] A. Tatu. *Visual Analytics of Patterns in High-Dimensional Data*. PhD thesis, Universität Konstanz, 2013.

[138] A. Telea and D. Auber. Code flows: Visualizing structural evolution of source code. In *Computer Graphics Forum*, volume 27, pages 831–838. Wiley Online Library, 2008.

[139] M. Thompson and C. Heneghan. We need to move the debate on open clinical trial data forward. *BMJ: British Medical Journal*, 345, 2012.

[140] H. Thorvaldsdóttir, J. T. Robinson, and J. P. Mesirov. Integrative genomics viewer (igv): high-performance genomics data visualization and exploration. *Briefings in bioinformatics*, page bbs017, 2012.

[141] K. D. Toennies, O. Gloger, M. Rak, C. Winkler, P. Klemm, B. Preim, and H. Völzke. Image analysis in epidemiological applications. *it-Information Technology*, 57(1):22–29, 2015.

[142] A. Treisman. Preattentive processing in vision. *Computer vision, graphics, and image processing*, 31(2):156–177, 1985.

[143] U. Tröhler. " to improve the evidence of medicine": Arithmetic observation in clinical medicine in the eighteenth and early nineteenth centuries. *History and philosophy of the life sciences*, pages 31–40, 1988.

[144] U. Tröhler, R. C. of Physicians of Edinburgh, et al. *To improve the evidence of medicine: the 18th century British origins of a critical approach*. Royal College of Physicians of Edinburgh, 2000.

[145] L. Trtica-Majnaric, M. Zekic-Susac, N. Sarlija, and B. Vitale. Prediction of influenza vaccination outcome by neural networks and logistic regression. *Journal of biomedical informatics*, 43(5):774–781, 2010.

[146] 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.

[147] C. Turkay, F. Jeanquartier, A. Holzinger, and H. Hauser. On computationally-enhanced visual analysis of heterogeneous data and its application in biomedical informatics. In *Interactive Knowledge Discovery and Data Mining in Biomedical Informatics*, pages 117–140. Springer, 2014.

[148] 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.

[149] C. Turkay, J. Parulek, N. Reuter, and H. Hauser. Integrating cluster formation and cluster evaluation in interactive visual analysis. In *Proceedings of the 27th Spring Conference on Computer Graphics*, pages 77–86. ACM, 2011.

[150] C. Turkay, J. Parulek, N. Reuter, and H. Hauser. Interactive visual analysis of temporal cluster structures. In *Computer Graphics Forum*, volume 30, pages 711–720. Wiley Online Library, 2011.

[151] A. Ultsch. Maps for the visualization of high-dimensional data spaces. In *Proc. Workshop on Self organizing Maps*, pages 225–230, 2003.

[152] A. Ultsch. *U*-matrix: a tool to visualize clusters in high dimensional data*. Fachbereich Mathematik und Informatik, 2003.

[153] S. van den Elzen and J. J. van Wijk. Baobabview: Interactive construction and analysis of decision trees. In *Visual Analytics Science and Technology (VAST), 2011 IEEE Conference on*, pages 151–160. IEEE, 2011.

[154] J. J. Van Wijk. The value of visualization. In *Visualization, 2005. VIS 05. IEEE*, pages 79–86. IEEE, 2005.

[155] J. Vesanto. Som-based data visualization methods. *Intelligent data analysis*, 3(2):111–126, 1999.

[156] I. Viola and J. Parulek. Interactive and exploratory visual analysis in biology. 2011.

[157] H. Wang and M. J. van der Laan. Dimension reduction with gene expression data using targeted variable importance measurement. *BMC bioinformatics*, 12(1):312, 2011.

[158] C. Ware. *Information visualization: perception for design*. Elsevier, 2013.

[159] M. Ware, E. Frank, G. Holmes, M. Hall, and I. H. Witten. Interactive machine learning: letting users build classifiers. *International Journal of Human-Computer Studies*, 55(3):281–292, 2001.

[160] J. D. Watson, F. H. Crick, et al. Molecular structure of nucleic acids. *Nature*, 171(4356):737–738, 1953.

[161] C. Weaver. Building highly-coordinated visualizations in improvise. In *Information Visualization, 2004. INFOVIS 2004. IEEE Symposium on*, pages 159–166. IEEE, 2004.

[162] J. N. Weinstein. Biochemistry. a postgenomic visual icon. *Science (New York, NY)*, 319(5871):1772–1773, 2008.

[163] S. Wetzel, K. Klein, S. Renner, D. Rauh, T. I. Oprea, P. Mutzel, and H. Waldmann. Interactive exploration of chemical space with scaffold hunter. *Nature chemical biology*, 5(8):581–583, 2009.

[164] A. Wheat. External representation of provenance in intelligence analysis. 2014.

[165] L. Wilkinson and M. Friendly. The history of the cluster heat map. *The American Statistician*, 63(2), 2009.

[166] M. Williams and T. Munzner. Steerable, progressive multidimensional scaling. In *Information Visualization, 2004. INFOVIS 2004. IEEE Symposium on*, pages 57–64. IEEE, 2004.

[167] S. Wold, K. Esbensen, and P. Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1):37–52, 1987.

[168] R. Xu, D. Wunsch, et al. Survey of clustering algorithms. *Neural Networks, IEEE Transactions on*, 16(3):645–678, 2005.

[169] J. Yang, D. Hubball, M. O. Ward, E. A. 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.

[170] J. Yang, M. O. Ward, and E. A. Rundensteiner. Visual hierarchical dimension reduction for exploration of high dimensional datasets. 2002.

[171] J. S. Yi, Y. ah Kang, J. T. Stasko, and J. A. Jacko. Toward a deeper understanding of the role of interaction in information visualization. *Visualization and Computer Graphics, IEEE Transactions on*, 13(6):1224–1231, 2007.

[172] P. Yildirim, M. Bloice, and A. Holzinger. Knowledge discovery and visualization of clusters for erythromycin related adverse events in the fda drug adverse event reporting system. In A. Holzinger and I. Jurisica, editors, *Interactive Knowledge Discovery and Data Mining in Biomedical Informatics*, volume 8401 of *Lecture Notes in Computer Science*, pages 101–116. Springer Berlin Heidelberg, 2014.

[173] P. Yildirim, L. Majnarić, O. I. Ekmekci, and A. Holzinger. Knowledge discovery of drug data on the example of adverse reaction prediction. *BMC Bioinformatics*, 15(Suppl 6):S7, 2014.

[174] H. Younesy, C. B. Nielsen, T. Möller, O. Alder, R. Cullum, M. C. Lorincz, M. M. Karimi, and S. J. Jones. An interactive analysis and exploration tool for epigenomic data. In *Computer Graphics Forum*, volume 32, pages 91–100. Wiley Online Library, 2013.