TU WIEN Informatics

Visuelle Analyse von periodischen Zeitreihen

Visual Analytics für die Modellauswahl, Vorhersage, Imputation und Ausreißererkennung

DISSERTATION

zur Erlangung des akademischen Grades

Doktor der Technischen Wissenschaften

eingereicht von

Dipl.-Ing. Markus Bögl, BSc
Matrikelnummer 00625252

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Univ.Prof. Mag.rer.soc.oec. Dr.rer.soc.oec. Silvia Miksch
Zweitbetreuung: Univ.Prof. Dipl.-Ing. Dr.techn. Peter Filzmoser

Diese Dissertation haben begutachtet:

Ross Maciejewski	Cagatay Turkay

Wien, 14. Oktober 2020

Markus Bögl

TU WIEN Informatics

Visual Analysis of Periodic Time Series Data

Supporting Model Selection, Prediction, Imputation, and Outlier Detection Using Visual Analytics

DISSERTATION

submitted in partial fulfillment of the requirements for the degree of

Doktor der Technischen Wissenschaften

by

Dipl.-Ing. Markus Bögl, BSc
Registration Number 00625252

to the Faculty of Informatics

at the TU Wien

Advisor: Univ.Prof. Mag.rer.soc.oec. Dr.rer.soc.oec. Silvia Miksch
Second advisor: Univ.Prof. Dipl.-Ing. Dr.techn. Peter Filzmoser

The dissertation has been reviewed by:

Ross Maciejewski	Cagatay Turkay

Vienna, 14th October, 2020

Markus Bögl

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.at

Erklärung zur Verfassung der Arbeit

Dipl.-Ing. Markus Bögl, BSc

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 14. Oktober 2020

Markus Bögl

Für Birgit und Frederik

Für meine Familie

Acknowledgments

I want to say a special thank you to my first advisor, Silvia Miksch, for her support, encouragement, enthusiasm, positive attitude, and final push to eventually finish my dissertation. In addition, I want to thank my second advisor, Peter Filzmoser. His lecture in *Exploratory Data Analysis and Visualization* started my journey to visual analytics, and the interdisciplinary cooperation between Silvia and Peter allowed me to continue my interest in statistics, exploratory data analysis and visual analytics while pursuing my PhD.

I want to thank all current and former colleagues from the CVAST research group at TU Wien: Alessio Arleo, Christian Bors, Davide Ceneda, Velitchko Filipov, Theresia Gschwandtner, Roger A. Leite, Nikolaus Piccolotto, Victor Schetinger, Bilal Alsallakh, Albert Amor-Amorós, Paolo Federico, and Tim Lammarsch.

I am also grateful to my collaborators from the Computational Statistics Department at TU Wien, Klaus Nordhausen and Christoph Mühlmann; my collaborators from the FH St. Pölten, Wolfgang Aigner, Alexander Rind, Christina Niederer, and Markus Wagner; my international collaborators from TU Darmstadt, Jürgen Bernard and Jörn Kohlhammer; and those from Universität Rostock, Martin Röhlig, Martin Luboschik, Christian Eichner, Christian Tominski, and Heidrun Schumann.

Lastly, I want to thank all people in the visualization and visual analytics community for the great time at the talks, workshops, conferences, doctoral colloquium, and other events as well as the reviewers for their time, efforts, and valuable feedback. I had the opportunity to participate in the last Visual Analytics Summer School at Middlesex University, UK, in 2013, during my first year as a PhD student. I am especially grateful to the organizers for this amazing opportunity to meet in person those famous researchers I knew from the books and papers—most notably, Jörn Kohlhammer, Ross Maciejewski, Natalia and Gennady Andrienko. This greatly helped build my research network and eventually resulted in cooperation for an international research project.

Finally, I want to express my great gratitude for several magnificent people from my extended family who helped and supported me during my PhD program. So many tiny gestures made the final phase of this dissertation possible: bringing/preparing food (lunch, dinner, pastries, and cake), bringing fresh fruits and vegetables, babysitting, helping me overcome my doubts, answering questions, proofreading, translating, and checking musical concepts. I will never forget everything you did for me.

My research was financially supported by the Centre for Visual Analytics Science and Technology (CVAST), funded by the Austrian Federal Ministry of Science, Research, and Economy in the

exceptional Laura Bassi Centre of Excellence initiative (#822746); the TU Wien by the Doctoral College for Environmental Informatics; the Austrian Science Fund (FWF) by the Lead Agency Procedure (DACH) “VISSECT: Visual Segmentation and Labeling of Multivariate Time Series” (#I2850-N31); the FWF by “HypoVis: Modeling Hypotheses with Visual Analytics Methods” (#P22883); the FWF by “Blind Source Separation in Time and Space” (#P31881-N32).

Kurzfassung

Zeitreihen sind in unterschiedlichen Forschungsbereichen präsent, sei es in Wirtschaft, Naturwissenschaften, Medizin und dergleichen mehr. Diese Art von Daten wird aufgezeichnet, um Messungen festzuhalten, zu analysieren und basierend darauf, Entscheidungen zu treffen. Periodizität ist eine Eigenschaft von Zeitreihen, die durch die natürliche Struktur der gemessenen Phänomene oder durch dahinterliegende Kalenderstrukturen bedingt ist. Diese strukturelle Eigenschaft der Zeit zeigt sich in vielen Datensätzen durch periodisch wiederkehrende Muster. Für die Analyse von Zeitreihen ist eben diese Eigenschaft von Vorteil, solange die Periodizität erkannt und richtig abgebildet wird. Dann kann sie helfen, passende Zeitreihenmodelle auszuwählen, Vorhersagen besser zu interpretieren, geschätzte fehlende Werte zu beurteilen und Ausreißer zu erkennen. Die periodischen Muster können direkt aus dem Zusammenhang der Daten gegeben oder in den Daten selbst versteckt sein. Um solche zu identifizieren, besteht eine Möglichkeit darin, sie mittels geeigneter visueller Darstellungen zu erkennen oder die Daten mithilfe von Visual Analytics zu explorieren, um die darunterliegenden Muster identifizieren und untersuchen zu können.

In der vorliegenden Dissertation werden verschiedenste Herausforderungen in der Zeitreihenanalyse betrachtet. Diese Herausforderungen betreffen im Speziellen periodische Zeitreihen und umfassen das Explorieren von Zeitreihen, die Auswahl möglichst passender Zeitreihenmodelle, die Unterstützung in der Parametrisierung, das Überprüfen der Vorhersagequalität der Modelle, das Berechnen und Ersetzen von fehlenden Werten sowie das Erkennen von Ausreißern. Für all diese Schritte wird die Anwendbarkeit von Visual Analytics-Methoden untersucht und es wird ermittelt, wie Nutzer*innen in diesen Aufgaben bestmöglich unterstützt werden können bzw. wie die Verflechtung neuer visueller Betrachtungswinkel auf periodische Zeitreihen gemeinsam mit menschlicher Wahrnehmung, mit Interaktionstechniken und mit statistischen Berechnungen, die herausfordernde Analyse von Zeitreihen positiv beeinflusst.

Als ersten Schritt stellen wir einen Visual Analytics-Ansatz vor, mit dem der gesamte Modellauswahlprozess unterstützt wird. Dazu ermöglicht der Ansatz die visuelle Analyse der Zeitreihe, bietet eine Orientierungshilfe für die Auswahl der Modelle und Modellparameter sowie für die genaue Diagnose bezüglich Angemessenheit der Modelle für die Daten. Weiters untersuchen wir die Zusammenführung der Vorhersagemöglichkeiten der gewählten Modelle und Modellparameter in den Modellauswahlprozess. Im nächsten Schritt verwenden wir eine spezielle Darstellungsform für periodische Zeitreihen (*Zyklengraph* bzw. *cycle plot*), um die Imputation, das heißt, die passenden Schätzungen von fehlenden Datenpunkten, zu verbessern und diese zu vervollständigen. Danach

zeigen wir unter Verwendung einer neuartigen Datenabstraktion, wie dieser Zyklengraph mithilfe dieser Abstraktion für multivariate Zeitreihen konstruiert und für das Erkennen von Ausreißern in multivariaten periodischen Zeitreihen verwendet werden kann. Für jeden dieser vorgeschlagenen Ansätze wurden iterative nutzer*innenzentrierte Designprozesse verwendet sowie Nutzen und Anwendbarkeit der Ansätze durch Nutzer*innenszenarien und gründlich beschriebene Funktionalitätstests bestätigt. Abschließend werden die Auswirkungen dieser Ansätze beschrieben sowie offene Fragestellungen und weitere Forschungsmöglichkeiten diskutiert. Das Miteinbeziehen von vorhandener Periodizität in der visuellen Darstellung erlaubt bei Visual Analytics-Ansätzen ein besseres Verstehen und Nachvollziehen von angewandten Zeitreihenmodellen, von Vorhersagen, von Imputationen und Ausreißern. Die Forschungsergebnisse zeigen, dass diese visuellen Darstellungsformen gemeinsam mit der passenden Datenabstraktion bei gleichzeitiger Berücksichtigung der speziellen periodischen Struktur der Zeit für die Analyse von Zeitreihen andere Sichtweisen auf die Daten bieten. Damit wird die Auswahl angemessener Modelle bzw. Modellparameter, das Miteinbeziehen der Vorhersagemöglichkeiten im Modellauswahlprozess, die Imputation fehlender Werte sowie das Identifizieren von Ausreißern verbessert bzw. erst ermöglicht.

Abstract

Time series data are essential in many fields, like economics, natural sciences, and medicine, to name a few. Measuring and recording these data allow us to document, analyze, and make decisions. One of the most natural structures in time series across all areas is periodicity, which stems from either the natural phenomena measured or the underlying calendar structure. One finds the structural property of time in many of these time series by periodic reoccurrences. In a time series analysis, these properties are mostly beneficial, if identified correctly and modeled adequately, for tasks like model selection, prediction, imputation, and outlier detection. These periodic patterns can be obvious due to the context or hidden in the data itself. Visualization is one way for human perception to identify such patterns easily and allow for the exploration, identification, and investigation of such underlying patterns using visual analysis.

In this dissertation, we consider the different stages of time series analysis for periodic time series, starting with exploring the time series, selecting appropriate time series models, supporting the parametrization, examining the prediction performance, imputing missing values, and detecting outliers. For all these steps, we investigate how visual analytics can support users in these tasks and how intertwining new perspectives on periodic time series using visualization together with user perception, interaction techniques, and statistical computations fosters the user in analyzing periodic time series.

We first propose a visual analytics approach for supporting the whole process of selecting appropriate time series models, allowing the visual exploration of time series while guiding the model selection, parametrization, and model diagnostics. We then investigate how to integrate the prediction capabilities of the model into the model selection process. Next, we employ a cycle plot representation to support the imputation of missing values in periodic time series. Thereafter, we present a novel abstraction method to use a cycle plot representation for multivariate time series as well in order to use it for outlier detection in periodic time series. For each of the proposed solutions, we employed an iterative user-centered design process; we showed the utility of the approaches in usage scenarios and thoroughly illustrated walk-throughs. Furthermore, we discussed the implications of such methods and concluded with open challenges in these topics. Integrating additional focus on visualizing periodicity into the visual analytics approaches allows for better comprehending the applied models, predictions, imputations, and outliers. The results indicate that adequate visual representation and abstraction, when considering the periodic structure of time, allows for the analysis of time series from different perspectives and provides

possibilities for identifying adequate time series models, supporting the imputation of missing values, and identifying outliers.

Contents

Part I

Exposition

CHAPTER 1

Introduction

"Once upon a time, statisticians only explored. Then they learned to confirm exactly—to confirm a few things exactly, each under very specific circumstances. As they emphasized exact confirmation, their techniques inevitably became less flexible. The connection of the most used techniques with past insights was weakened. Anything to which a confirmatory procedure was not explicitly attached was decried as 'mere descriptive statistics', no matter how much we had learned from it. [...]

Today, exploratory and confirmatory can—and should—proceed side by side."

John W. Tukey, 1977. [Tuk77, p. vii]

1.1 Motivation and Problem Analysis

No one can escape time. Everybody has to deal with it. Time is everywhere and everything. "The temporal entity [time] is a true enigma that neither scientists (from any field) nor philosophers even know how to define exactly" [CB14, p. 250]. "The fundamental phenomenon of time has always been of interest for mankind [... and is] discussed over literally thousands of years in philosophy, mathematics, physics, astronomy, biology, and many other disciplines" [AMST11, p. 45]. The oldest sources of documenting and structuring time are simple calendar concepts [AMST11, p. 46]; although these calendar concepts are human made, this very interesting periodic behavior of time is found everywhere in nature and in all kinds of natural phenomena.

Because time is so central to life, it is the basis of a vast amount of data collected and measured, from weather data of the past centuries to huge amounts of transaction data nowadays. "Data obtained from observations collected sequentially over time are extremely common" [CC08, p. 1]. Collecting these data, as well as analyzing the data for meaningful information and knowledge, is more important than ever and requires appropriate techniques to do so. In mathematics, specifically in statistics, this area of research and application is called time series analysis. A time series is a set of sequential measurements of variables over time [CC08, Cle94]. There is a great history of statistical analysis of time series data and generations of statisticians have contributed to this field. According to Tsay [Tsa00] time series analysis reaches back to 1927, although he states that forecasting has an even longer history.

According to Bisgaard and Kulahci [BK11], many practitioners find time series analysis complicated and frustrating. Time series analysis is applied "in many different fields such as finance, economics, engineering, healthcare, and operations management, to name a few" [BK11, p. 1]. In a brief history of time series and forecasting, Tsay listed the main purposes of time series analysis used in business and economics: "(a) to study the dynamic structure of a process, (b) to investigate the dynamic relationship between variables, (c) to perform seasonal adjustment of economic data [...], (d) to improve regression analysis [...], and (e) to produce point and interval forecasts [...]" [Tsa00, p. 638]. According to Tsay, the 1970 landmark work by Box and Jenkins, *Time Series Analysis: Forecasting and Control* [BJ70], "was an important milestone [...] It provided a systematic approach that enables practitioners to apply time series methods in forecasting" [Tsa00, p. 639]. The suggested model selection approach by Box and Jenkins is used in most textbooks on time series analysis, such as [BJR08, BD91, CC08, SS11, Ham94, BK11, BD02, CM09, Pfa08, Bro11].

In his famous book *The Visual Display of Quantitative Information*, Tufte defines *data graphics* as "visually display[ing] measured quantities by means of the combined use of points, lines, a coordinate system, numbers, symbols, words, shading, and color" [Tuf83, p. 9]. As Cleveland states, "[d]ata display is critical to data analysis. Graphs allow us to explore data to see overall patterns and to see detailed behavior; no other approach can compete in revealing the structure of data so thoroughly. Graphs allow us to view complex mathematical models fitted to data, and they allow us to assess the validity of such models" [Cle94, p. 5]. Cleveland further argues that "[t]he visualization of statistical data has always existed in one form or another in science and technology. [...] But with the appearance of John Tukey's pioneering 1977 book, *Exploratory Data Analysis*, visualization became far more concrete and effective [Tuk77]. Since 1977, changes in computer

systems have changed how we carry out visualization, but not its goals" [Cle93, p. 2]. Going back to an original quote by Tukey himself, "[a] basic problem about any body of data is to make it more easily and effectively handleable by minds—our minds, her mind, his mind" [Tuk77, p. v]. Tukey's proposed solution to make it more easily and effectively manageable was to introduce exploratory data analysis; a central aspect of exploratory data analysis is a visual representation of data, whether it is called statistical graphs or plots. As Cleveland explains, "[v]isualization is critical to data analysis. It provides a front line of attack, revealing intricate structure in data that cannot be absorbed in any other way. We discover unimagined effects, and we challenge imagined ones" [Cle93, p. 1].

Time series analysis is a tedious and challenging task that involves data, computations, visual representations, and practitioners—generally experts—in the application domain of the data at hand. Usually the purposes of time series analysis, as previously mentioned, are achieved by exploring the data numerically and visually, visualizing the data, computing metrics, adjusting data, visualizing these metrics, combining the insights with domain knowledge about the time series models, selecting adequate models, deciding on initial parameters of the models, computing/estimating model parameters, checking the model outcome, looking at model results, applying the models, looking at the outcome of these model applications, reselecting other models and/or adjusting the model parameters, and repeating this cycle. This kind of workflow is often repeated multiple times until an adequate model is found. Essentially, this described process asks not only for a combined exploratory and confirmatory data analysis, like Tukey demands in his famous book [Tuk77, p. vii], but also for combining computations and visual representations as well as multiple runs with iterative adjustments of this process. While presenting his famous Anscombe's quartet, Anscombe—another famous statistician—stated that "[a] computer should make *both* calculations *and* graphs. Both sorts of output should be studied; each will contribute to understanding" [Ans73, p. 17]. He also argued that a "[g]ood statistical analysis is not a purely routine matter, and generally calls for more than one pass through the computer" [Ans73, p. 17].

To deal with such challenges, the highly interdisciplinary research area of visual analytics [TC05, KKEM10] was introduced with the general idea to combine and intertwine computation, visualization, and interaction. The advancement of computation and interactive computer systems, together with research progress made in data visualization, human computer interaction, and visual analytics, have allowed completely new approaches and techniques to support such challenges, as apparent in time series analysis.

The aim of this dissertation is to present visual analytics approaches that support solutions to some of these challenges in the field of statistical time series analysis, including model selection, parametrization, exploration, outlier detection, imputation, and prediction.

1.2 Data Analysis and Visualization

As previously mentioned, in statistics, Tukey [Tuk77] introduced the area of exploratory data analysis, where the main idea was to shift the focus from mainly confirmatory data analysis—basically testing hypotheses based on the data—to also consider exploratory data analysis, where the central point is to look at the data (graphically or numerically) and determine the results in

order to generate hypotheses that can then be tested in order to confirm or reject them. Tukey's contributions to statistics, including the concepts of exploratory data analysis are central and still valid in data science nowadays. The technological and scientific advancement in mathematics, statistics, and computer science allowed for the creation of advanced visualizations, including advanced interactions, to create more complex yet powerful techniques, although the main aspects and goals of Tukey's ideas on exploratory data analysis still hold today.

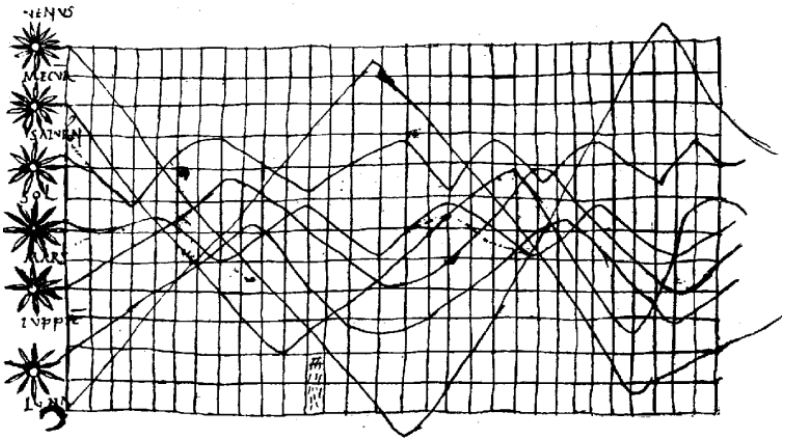


Figure 1.1: Possibly the first time series graph found in literature, author unknown, published by Funkhouser [Fun36]. It shows planetary orbits as a function of time. It dates from the 10th or possibly 11th century.
Image Source: Funkhouser (1936) [Fun36, p. 261], Osiris ©1936, courtesy of the University of Chicago Press.

According to Cleveland, as quoted earlier, the history of "visualization of statistical data has always existed in one form or another in science and technology" [Cle93, p. 2]. The literature about the history of graphical representations [RG10, Fun36, Fun37] and information visualization, such as [Tuf83, AMST11], as well as in textbooks about statistical time series analysis [CC08], all refer to a graph "meant to represent a plot of the inclinations of the planetary orbits as a function of time" [Fun36, p. 260] from the tenth or possibly eleventh century, which seems to be a "compiled text for use in monastery schools" [Fun36, p. 260]. The fascinating thing about this graph, shown in Figure 1.1, is that it not only uses this form of inclinations of the orbits as a function over time, essentially representing a time-line graph, but also captures the periodicity of many phenomena in nature, like the planetary orbits, and shows them in relation to each other. The x-axis is chosen in a way to show one full period of the longest periodic duration, which is by Venus, and one can easily recognize that the periodic pattern is meant to repeat on the left of the graph, when the line ends at the right.

For Tufte this oldest known graph "appears as a mysterious and isolated wonder in the history of data graphics, since the next extant graphic of a plotted time-series shows up some 800 years later"[Tuf83, p. 28]. Tufte refers to work by William Playfair and Johann Heinrich Lambert in the late 1700s, who are generally seen as the inventors of modern graphical design in the literature,

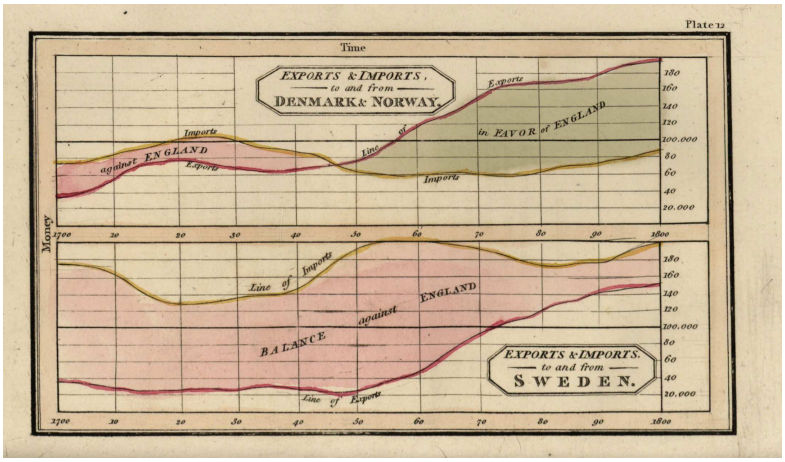


Figure 1.2: Time series line graph showing the imports and exports between England and Scandinavian countries, Playfair [Pla01]. Playfair's line graphs are considered the first representation of time series as we know and use it today. The first edition of his book *The Commercial and Political Atlas* was published in 1786. This graph is taken from the third edition published in 1801. The graphs improved over the years, but the core elements are the same.
Image Source: Playfair (1801) [Pla01, plate 12]. Image retrieved from archive.org: `https://archive.org/details/PLAYFAIRWilliam1801TheCommercialandPoliticalAtlas` (last visited on Sept. 24, 2020)

contributing to graphical representations of statistical data, specifically representing time series in the way we still use it today [Fun37, Til75, Tuf83, RG10, AMST11].

According to Funkhouser, William Playfair "may be called the father of the graphic method in statistics" [Fun37, p. 273] and as the "[a]pparent Inventor of Statistical Graphs" [Fun37, p. 280]. To clarify the significance of Playfair's work, Funkhouser explains that Playfair published his first edition of *The Commercial and Political Atlas* in 1786, a time when "[t]he term *statistics* had not yet appeared in the English language, few collections of reliable quantitative data were available and the development of statistical method was still far in the future [...]" [Fun37, p. 280]. He essentially contributed line graph, circle graph, bar graph, and pie diagram as we still know and use them nowadays for the visualization of data (see Figure 1.2). In addition, he accompanied these charts "with pointed expositions of the advantages of the new method for the discovery and analysis of economic trends" [Fun37, p. 280].

Tilling, in her article about early experimental graphs [Til75], discusses in detail Lambert's work and contributions to graphs for presenting or analyzing experimental results. One notable graph is the graphical analysis of periodic variation, as shown in Figure 1.3, which shows the variation in soil temperature. This graph by Lambert again focuses on the periodic pattern of a natural phenomenon. Tilling also refers to a paper by Lambert titled *Theorie der Zuverlässigkeit* (roughly translated as *Theory of Steadiness*), in which Lambert introduced graphical representations to detect a periodic variation and determine its period [Til75, p. 201].

Essentially, both Playfair and Lambert are presumed to be the first to apply visual data analysis,

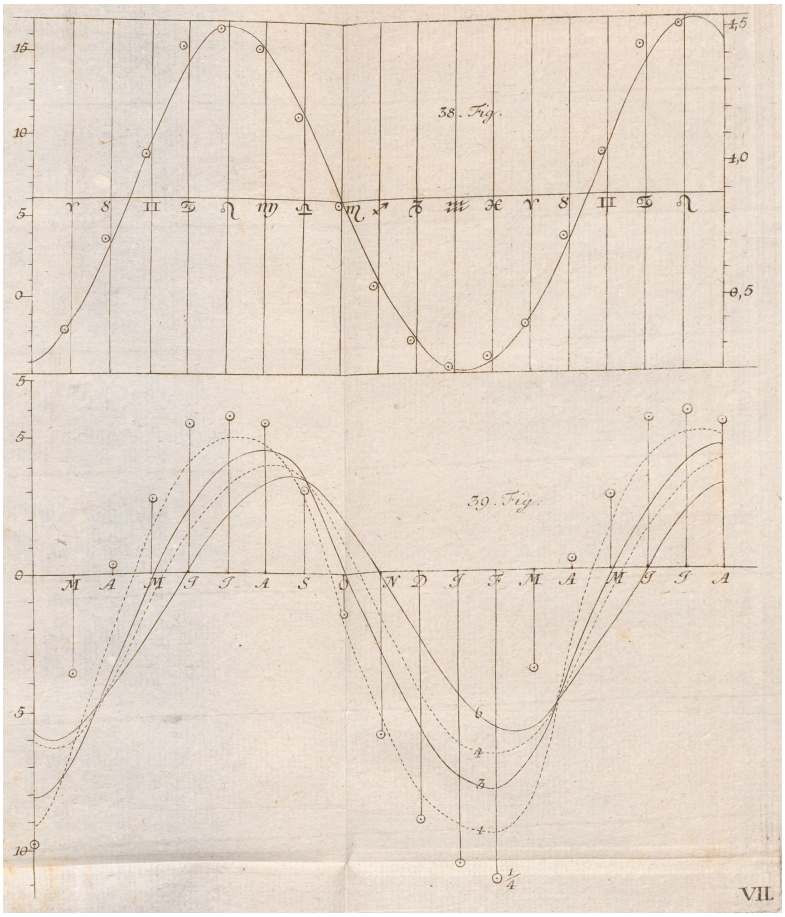


Figure 1.3: Periodic variation represented as line chart, Lambert [Lam79]. This graph is showing a graphical analysis of periodic variation of temperature in different depths of soil.
Image Source: Lambert (1779) [Lam79, p. 407], provided by Bayerische Staatsbibliothek München, 4 Phys.sp. 150, p. 407, Figure 39, Plate VII, urn:nbn:de:bvb:12-bsb10058497-5.

because they not only visualized the data at hand, but also used them for "[the] discovery and analysis of economic trends" [Fun37, 280] (Playfair) and to "detect a periodic variation and to determine its period" [Til75, p. 201] (Lambert). Lambert not only displayed his data in a graph, but as shown in Figure 1.3 "was accustomed to interpose a smooth curve amongst the observations" [Til75, p. 201] and in other graphs "used the technique of graphical differentiation" [Til75, p. 201]. "Lambert was particularly aware of the necessity, as a step in scientific reasoning, of a careful examination of the goodness of match between theory and data" [Til75, p. 204]. These findings suggest that he was already combining visualization and computation, as we are used to doing today in visual data analyses and how Tukey postulated it for exploratory data analysis and Cleveland for visualizing data for data analysis:

"It was an early lesson that we gain a lot by plotting the points. It was a later

> lesson, not emphasized as hard as it should be, perhaps, that you often do not learn enough until you smooth or middle the points. Plotting the points is often not enough." [Tuk77, p. 665]

> "There are two components to visualizing the structure of statistical data—graphing and fitting. Graphs are needed, of course, because visualization implies a process in which information is encoded on visual displays. Fitting mathematical functions to data is needed too. Just graphing raw data, without fitting them and without graphing the fits and residuals, often leaves important aspects of data undiscovered." [Cle93, p. 1]

As discussed earlier, Tukey's introduction of exploratory data analysis is widely considered as the cornerstone for modern visual data science. It is also well accepted in the visual analytics research community, cf. [KKEM10, p. 3], that the first step toward visual analytics research was moving from confirmatory data analysis to exploratory data analysis, as stated by Tukey [Tuk77]. Before we proceed with introducing the field of visualization briefly and visual analytics in more detail in the next section, we first want to quote some important statements from Tukey's book that are still very relevant in visual analytics and visual data science in general:

> "The greatest value of a picture is when it *forces* us to notice what we never expected to see." [Tuk77, p. vi]

> "Exploratory data analysis can never be the whole story, but nothing else can serve as the foundation stone—as the first step." [Tuk77, p. 3]

> "Summaries can be very useful, but they are not the details." [Tuk77, p. 27]

For the following sections, it is important first to specify what visualization in general actually means. According to some great researchers in the field of information visualization, the definition of *visualization* is "[t]he use of computer-supported, interactive, visual representations of data to amplify cognition" [CMS99, p. 6]. They then distinguish between the field of *scientific* visualization and *information* visualization, where the difference lies in the data that is visualized. For the first, it is scientific data, which is usually based on physical data; for the latter abstract data, it is nonphysical, abstract information, with no obvious spatial mapping. The field of *information visualization* is then defined by Card et al. in the same way as visualization, only changing *data* to *abstract data* [CMS99, p. 7]. For a definition of visualization, Keim et al. [KMS+08] refer to Colin Ware [War00], who understood the term visualization as "a graphical representation of data or concepts" [War00]. Over the years the (sharp) separation among the fields of information visualization, scientific visualization, and visual analytics (which we will introduce below) crumbled and broke off, and the fields moved closer to liaising with each other. There is a rumored tendency of combining all three of these fields under the term *visualization*. For the last several

years (at least since 2017), the major IEEE VIS conference has been advertised as "premier forum for advances in visualization".[1] Although it still has separate conference tracks for the three fields of visualization, in some near future this will likely change.

Thus far, we have already expressed our focus on challenges in time series analysis and dealing with the periodic structure of time. In the field of information visualization, a corresponding strain of research copes with challenges in how to visualize time and time series data as well as how to deal with the specific challenges in analyzing this type of data. In their terminology they call this type of data *time-oriented data* [AMM+07, AMST11]. The foundation of visualizing time-oriented data was set by Aigner et al. [AMM+07] when introducing a systematic view on diverse methods for visualizing this data and most importantly introduce a categorization that provides structure and supports in selecting appropriate visualization techniques for different purposes. These contributions were later extended and published in a book [AMST11] to provide a comprehensive view of the topic of visualizing time-oriented data. Most relevant for the content in this dissertation is the categorization into the structure of time, specifically linear and cyclic time.

The research area of information visualization contributes to Tukey's idea of seeing the data and the results. With novel graphical interfaces and interaction techniques for building a dialogue between data and users, advances in exploratory data analysis were made in a fashion never possible before. In the following section, we extend on these foundations and introduce the field of visual analytics.

1.3 Visual Analytics

Thus far, we have shown that the problem of data analysis and the visualization of (statistical) data have a long history of being a relevant topic in research and applications. Because of the technological advancement in computation, the demand to cope with the massive amount and increased complexity of the collected data, and the demand to make sense out of it, the research field of visual analytics was established; within this research field, scientists contribute to data analysis problems in many domains.

Thomas and Cook first defined the term visual analytics in 2004 as "the science of analytical reasoning facilitated by interactive visual interfaces" [TC05, p. 4]. This first book defining and describing the research and development agenda was heavily motivated, influenced, and focused on homeland security and the identification of terrorist threats. Within a few years, the research and application field of visual analytics evolved, and a European research community formed and contributed to research in the field. A paper about the scope and challenges of visual analytics by Keim et al. [KMS+08] was an important initial contribution to specify the field of visual analytics further. This was basically also the cornerstone for a two-year European Commission-funded project called the VisMaster initiative, in which experts from European academic and industrial R&D contributed a detailed review of all aspects of visual analytics from the European perspective to a consortium [KKEM10].

[1] `http://ieeevis.org/` (last visited on Oct. 11, 2020)

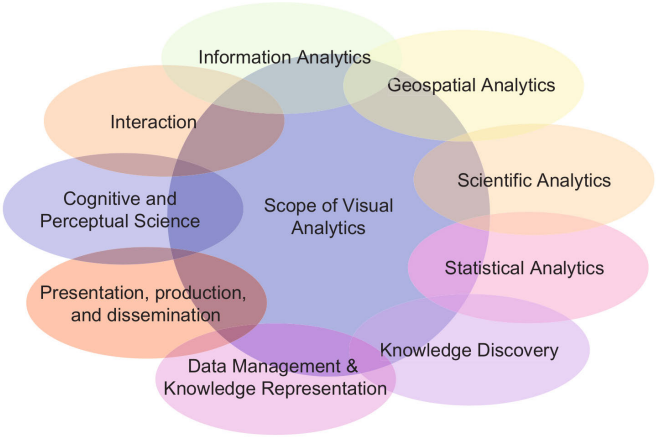


Figure 1.4: The scope of visual analytics, Keim et al. [KMS+08]. Illustration of the involved research and application fields in the scope of visual analytics.
Image Source: Keim et al. (2008) [KMS+08, p. 79], courtesy of Springer © 2008.

Keim et al. [KMS+08] adapted and extended the famous information-seeking mantra "overview first, zoom/filter, details on demand" [Shn96, p. 337] for visually exploring data to transform it into the visual analytics mantra of: "Analyze first, show the important, zoom/filter, analyse further, details on demand" [KKEM10, p. 11], first defined in [KMS+08, p. 82]. Keim et al. [KKEM10, p. 11] later argued that this adaptation is necessary because, when we are dealing with massive datasets, it is often not possible to create overview visualization without losing interesting patterns. In this case, we need to first analyze what the interesting aspects in these data are, show what seems to be important, allow interactions (e.g., zoom), and filter to drill down the analysis until we get to the details necessary.

Keim et al. [KMS+08] argue that visual analytics, by evolving from information and scientific visualization, is now much more than only visualization. Rather, the scope is a field that combines visualization, human factors, and data analysis. They illustrate the involved topics and areas by defining the detailed scope of visual analytics (cf. Figure 1.4). This indicates that visual analytics is a highly interdisciplinary field and covers a broad scope of challenges and opportunities. The ultimate goal of visual analytics is to "gain information, insights, and assessments from complex data" [ALA+18, p. 275]; to do so involves combining the strength of machines, or the computational power, with the strength of humans, like perception and cognition. The core elements in the computational automatic analysis part are data mining, machine learning, databases, statistics, and mathematics, whereas complementing these strengths with human capabilities to perceive, relate, and conclude is one of the reasons why visual analytics grew into a flourishing field of research [KMS+08].

In 2010, the results of the VisMaster initiative were published in the book *Mastering the information age: Solving problems with visual analytics* [KKEM10]. In this book, based on practical experience in visual analytics research, Keim et al. defined visual analytics with a more specific definition than Thomas and Cook as "visual analytics combines automated analysis

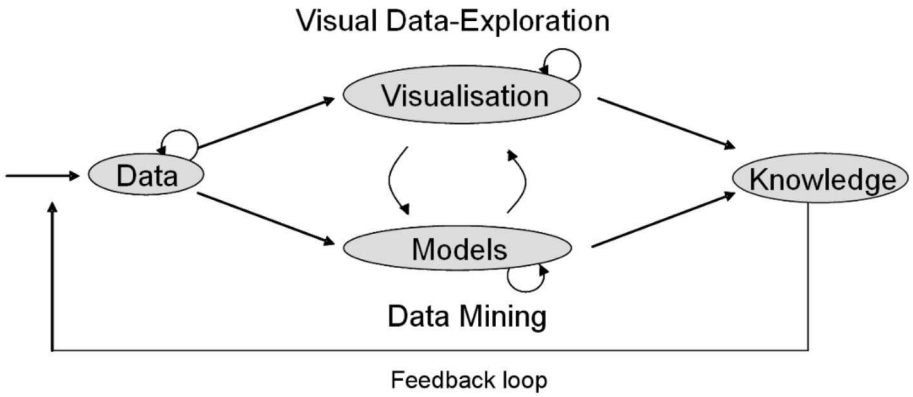


Figure 1.5: The visual analytics process, Keim et al. [KAF+08]. It shows an abstract overview of different stages and transitions in a visual analytics process.
Image source: Keim et al. (2008) [KAF+08, p. 156], courtesy of Springer © 2008.

techniques with interactive visualizations for an effective understanding, reasoning and decision making on the basis of very large and complex datasets" [KKEM10, p. 7]. They also defined the goal of visual analytics by the creation of tools and techniques, which serve the goal to enable people to

> "1. Synthesise information and derive insight from massive, dynamic, ambiguous, and often conflicting data.
> 2. Detect the expected and discover the unexpected.
> 3. Provide timely, defensible, and understandable assessments.
> 4. Communicate these assessment[s] effectively for action." [KKEM10, p. 7]

In their book, Keim et al. [KKEM10] defined a high-level process for achieving these goals by proposing a visual analytics reference model. A predecessor of this process (see Figure 1.5) was previously published by Keim et al. [KAF+08, KMS+08] in 2008, at the beginning of the VisMaster project. Essentially, this process describes a combination of automatic and visual analysis methods with a tight coupling involving human interaction in an iterative fashion to gain knowledge and insights from the data. Extending on this process definition for visual analytics, Sacha et al. [SSS+14] expanded the knowledge-generation process of human cognition activities. They argued the distinction between the computer and human parts and, in addition to the earlier process model, illustrated in more detail the knowledge-generation process of the human user. Sacha et al. emphasized the recent claims by Endert et al. [EHR+14] to go beyond the "human-in-the-loop" thinking to a "human-is-the-loop," so that the human work processes need to be directly integrated and recognized in the analytics. Sacha et al. [SSS+14] distinguished among an exploration, verification, and knowledge-generation loop, which comprise the overall reasoning process by human users of a visual analytics system (see Figure 1.6). The basic idea of modelling the knowledge generation in loops is to allow several loops in parallel, because analysts' cognitive processes are often rather chaotic and spontaneous, and they can work on different hypotheses,

tasks, or findings in parallel. The exploration loop is applied to generate new visualizations or models in order to analyze the data and allowing data exploration to find something interesting and build hypotheses. These findings and hypotheses are confirmed or new ones are formed with the combination of exploration and verification loop, whereas the verification provides guidance for the exploration. During this process insights are formed, which are collected and combined in the knowledge-generation loop to new knowledge, after the analysts have gained trust into these insights and hypotheses during the verification loop.

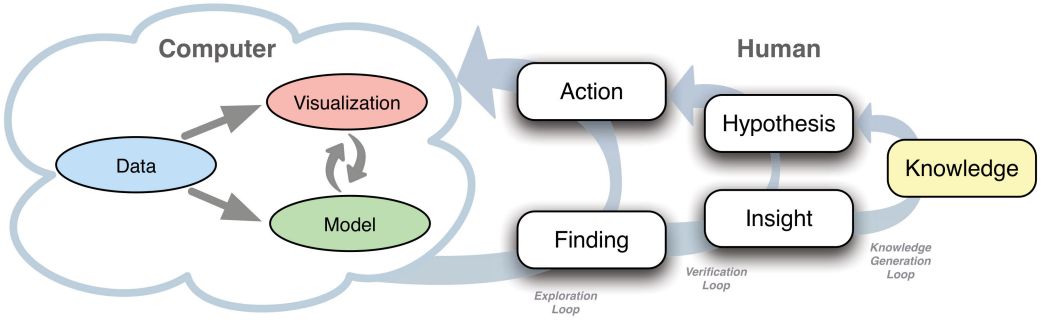


Figure 1.6: Knowledge-generation model for visual analytics, Sacha et al. [SSS+14]. In addition to the earlier visual analytics process definition, this process model illustrates the knowledge-generation process of the human in more detail.
Image Source: Sacha et al. (2014) [SSS+14, p. 1604], courtesy of IEEE © 2014.

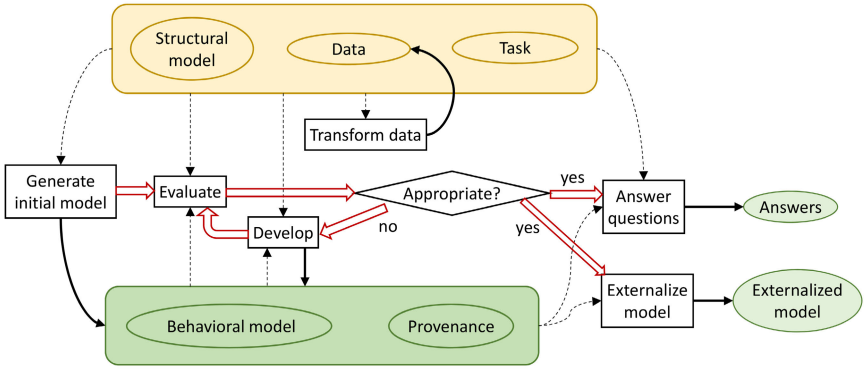


Figure 1.7: A visual analytics workflow for viewing visual analytics as model building, Andrienko et al. [ALA+18]. This visual analytics workflow illustrates how the final product—i.e., the expected results, or a *model*, as some piece of reality—of a visual analysis is created.
Image Source: Andrienko et al. (2018) [ALA+18, p. 276], courtesy of the Eurographics Association and John Wiley & Sons Ltd. © 2018.

Another addition to the family of visual analytics process definitions focuses on the hypothesis and model generation during a visual analytics process specifically for time-oriented data [LAB+11]. We show this process in Chapter 2 (especially Figure 2.4), where we use this process to describe the

visual analytics process for time series model selection, applying the Box-Jenkins methodology (cf. Section 2.3.1 in Chapter 2). This visual analytics process definition by Lammarsch et al. [LAB+11] can be considered an intermediate step, moving from the previously discussed general visual analytics process by Keim et al. [KKEM10] to a more recent visual analytics process definition by Andrienko et al. [ALA+18] that focuses on the final product (i.e., the expected results) of a visual analysis. Compared to the previous process definitions, the main idea is to complement existing definitions and conceptual frameworks proposed with this view, because these other processes instead focus on the activities performed and types of techniques used by analysts in visual analytics. Usually, the final product of visual analytics activities is described using the terms *information*, *knowledge*, or *insight*, which are rather general terms and, according to Andrienko et at. [ALA+18], do not clearly define the expected product but instead refer to the process of activities of a visual analytics analyst. They argued that visual analytics is a "purposeful activity directed to achieving a certain previously stated goal" [ALA+18, p. 275]. Consequently, the result should not be "any trustworthy insight but a knowledge product satisfying the analysis goal" [ALA+18, pp. 275–276]. They defined the overall goal of analysis, which is the product/result that needs to be created, or a *model*, as some piece of reality. Another emphasis in this work is that "*the primary interest of the analyst is not the data per se but the reality reflected in the data*" [ALA+18, p. 283]. Essentially, in their process definition the model is a generalization of the data and cannot represent all aspects, relationships, or details. An important step for complex models is to externalize and offload parts of the model because, in a complex model, it is occasionally difficult for an analyst to keep it fully in mind. The power and main benefit of visual analytics methods are that they allow for this process because visual analytics is exactly built around this idea. An important declaration in this paper is that the knowledge gained during the process of analysis is based on the built model of the subject, whether it is completely in the mind of the analyst or offloaded and distributed via different media. Using this concept of model building in visual analytics promotes better understanding of what kind of model development, evaluation, and externalization needs to be supported. To give more guidance for visual analytics researchers and developers for the design of visual analytics methods, procedures, and tools, Andrienko et al. [ALA+18] provided a summarized action list to perform.

Using visual analytics process definitions, like those by Andrienko et al. [ALA+18], Sacha et al. [SSS+14], Lammarsch et al. [LAB+11], and Keim et al. [KKEM10] as framework for defining and proposing visual analytics methods and solutions is already one part to the puzzle of the used research methodology in this dissertation. In addition, to consider the process definitions and follow the proposed frameworks as the guiding structure in visual analytics research, we will discuss the additional research methodology applied in this dissertation in the following section.

1.4 Methodology

In the field of visual analytics research, there is a reasonably accepted set of methodologies to follow for doing visual analytics research and proposing visual analytics methods, techniques, and solutions. The work in this dissertation is based on these methodological foundations.

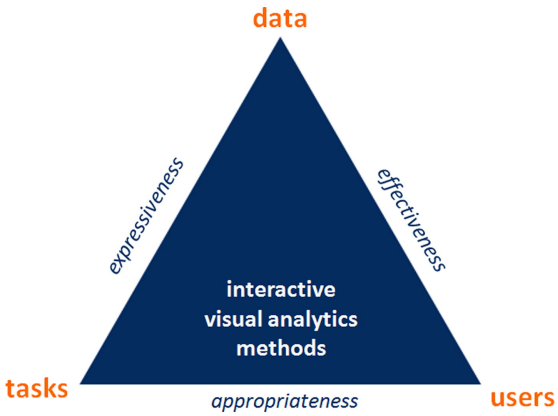


Figure 1.8: Design triangle framework, Miksch and Aigner [MA14]. It supports visual analytics researchers and designers in specifying requirements using a data–users–tasks paradigm together with expressiveness–effectiveness–appropriateness as quality criteria for interactive visual analytics methods.
Image Source: Miksch and Aigner (2014) [MA14, p. 286], courtesy of Elsevier © 2014.

Data–Users–Tasks Framework. By definition, a central part in each visual analytics solution is to validate whether the proposed solution can handle the large and complex data of the problem, which is useful for the user to gain insights, and satisfies the required tasks applied by the users facing the problem. For this reason, Miksch and Aigner [MA14] introduced the design triangle shown in Figure 1.8 as a high-level framework to support the design of interactive visual analytics methods, with a focus on time-oriented data. The idea is to use the design triangle to clarify the data with which the user is working, describe the user who will use the method, and specify the tasks that the users are about to apply to the data at hand. Specifying and answering this data–users–tasks abstraction gives indications of appropriate visual representations and what analytical and interaction methods are applicable. In the triangle in Figure 1.8, the authors also specify the major quality criteria for visual analytics methods, like expressiveness [Mac86], effectiveness [Mac86], and appropriateness [vW06]. In order to get useful visual analytics methods, Miksch and Aigner [MA14] ask for (a) only visualizing exactly what is contained in the data (expressiveness); (b) visualizations that are intuitive, recognizable, and interpretable (effectiveness); and (c) visualizations that help solve the given task and reach the defined goal (appropriateness).

Task Abstraction. Special focus needs to be put on the definition and abstraction of the tasks during the design of visual analytics methods. This is an important step in order to support the intended user in generating an appropriate interactive analysis of the given data and deriving insights and knowledge. To abstract and describe the tasks to be supported appropriately, a number of taxonomies and typologies are presented in the visual analytics and related literature. The most notable ones mentioned within the design triangle framework by Miksch and Aigner [MA14] are a set of taxonomies for visualization in general. These are the famous *task by data type taxonomy for information visualization* by Ben Shneiderman [Shn96] as well as the contributions by Brehmer

and Munzner [BM13] and Schulz et al. [SNHS13]. More specifically considering time-oriented data is the task typology by Andrienko and Andrienko [AA06]. These task taxonomies/typologies help visual analytics researchers structure, select, and describe information from the large set of possible tasks applied during an analytical reasoning process.

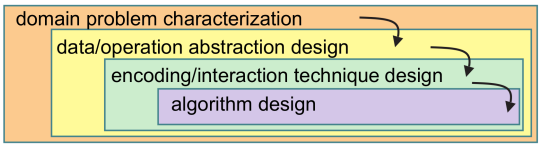


Figure 1.9: Nested model for visualization design and validation, Munzner [Mun09]. This model provides guidance for visualization designers by splitting the design process into four nested layers and suggests evaluation methodologies for possible threats to validity in each level.
Image Source: Munzner (2009) [Mun09, p. 922], courtesy of IEEE © 2009.

Nested Model. In addition to the visual analytics process models, an important aspect in visual analytics techniques is the human factor and, by definition, the tight integration of the user into the process. This requires a user-centered approach when designing and evaluating visual analytics methods. For this reason, Munzner introduced her nested model for visualization design and validation in 2009 [Mun09]. The general idea of this nested model, as shown in Figure 1.9, is to guide visualization designers in the design process by splitting it into four nested levels and provide suggestions for distinct evaluation methodologies for each level. The outer level is to characterize the domain problem space, like defining the tasks and data with the vocabulary of the domain. The second level's goal is to abstract the tasks and data into operations and data types. In the third level, the visual encoding and interaction techniques are designed, and the last level considers the definition of algorithms that need to execute these techniques efficiently. Munzner also identified possible threats to validity at each level and recommended evaluation methodologies to appropriately validate the different design choices. The possible threats to validity are also categorized into four levels: wrong problem, abstraction, encoding/interaction, and algorithm [Mun09, p. 921].

Design Study Methodology. During the years of visualization research, design studies—a form of problem-driven research—has become increasingly popular, and Sedlmair, Meyer, and Munzner condensed this ongoing practice into a design study methodology in 2012 [SMM12]. They proposed a holistic methodological approach for conducting design studies more effectively and provided guidance for applying their proposed nine-stage design framework. One important precondition for the suitability of design study methodologies is the clarity of the tasks to support and the location of the information with which to work. These preconditions can be best summarized in their task clarity and information location chart, as shown in Figure 1.10. The yellow area indicates the regions where design studies are a possible methodological choice. Sedlmair et al. defined a design study as a form of problem-driven research, where the researchers are working with real users and try to solve their real-world problems. Specifically, they define a design study as:

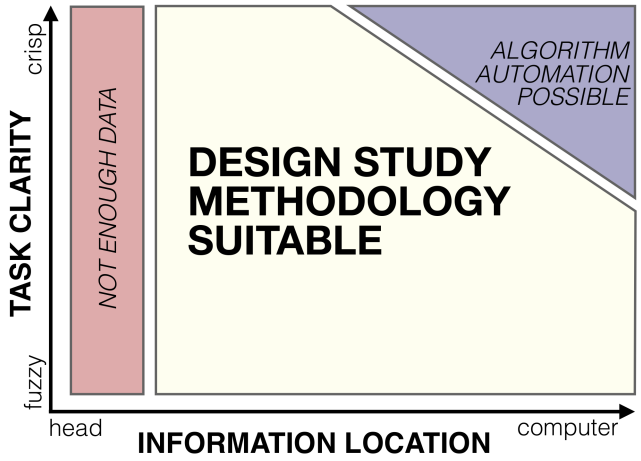


Figure 1.10: Design study methodology—task clarity and information location chart, Sedlmair et al. [SMM12]. The colored areas indicate where a design study methodology is (yellow) or is not (red and blue) suitable.
Image Source: Sedlmair et al. (2012) [SMM12, p. 2433], courtesy of IEEE © 2012.

> "a project in which visualization researchers analyze a specific real-world problem faced by domain experts, design a visualization system that supports solving this problem, validate the design, and reflect about lessons learned in order to refine visualization design guidelines." [SMM12, p. 2432]

In addition to defining a design study methodology, Sedlmair et al. proposed their nine-stage framework to help researches conduct design studies with a specific process. The nine-stage framework is illustrated in Figure 1.11. The framework consists of three phases: the precondition phase for personal validation, the core phase for inward-facing validation, and the analysis for outward-facing validation. The framework follows a highly linear process with overlapping stages, but includes backward jumps for adapting and refining ideas and forming understanding. Each of the stages is described with practical guidance based on reflections on earlier design study projects by the authors and other authors in the visualization community. The authors also discussed possible pitfalls in each of the stages. Munzner already contributed some critical aspects about design studies in her 2008 paper [Mun08], in which she discussed major pitfalls. Sedlmair et al. [SMM12] further elaborated on these pitfalls and structured them according to the design study methodology, especially in the nine-stage framework. They also provided strategies to avoid these pitfalls in the relevant stages.

Evaluation Methods. To validate visual analytics solutions, they need to be evaluated adequately. Because the human user is such a central figure in visualization and visual analytics, many aspects from the human–computer interaction field are relevant and are transferred to visualization and visual analytics, especially in the design and evaluation process of visual analytics solutions. Lam et al. [LBI+12] contributed a scenario-based look at the evaluation techniques of information

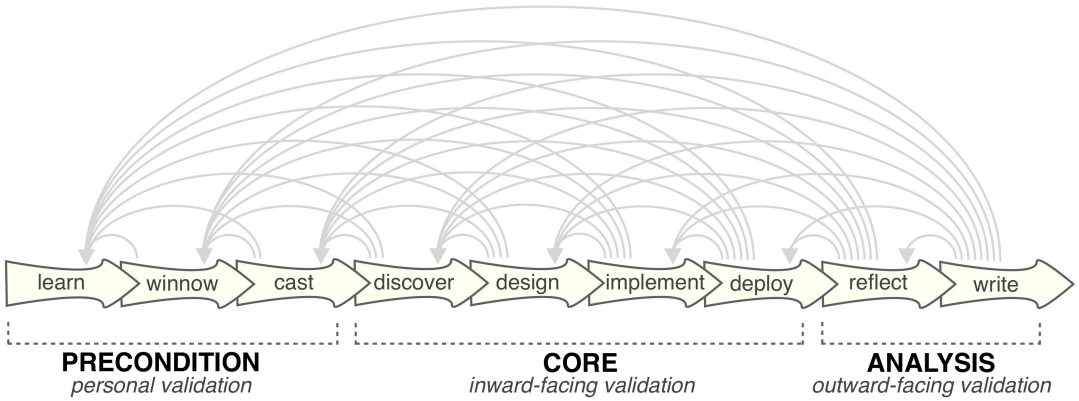


Figure 1.11: Design study methodology—a nine-stage framework for conducting design studies in research, Sedlmair et al. [SMM12]. The framework is considered linear, but because stages often overlap and the process is highly iterative, jumping backwards is very common to refine ideas and form understanding.
Image Source: Sedlmair et al. (2012) [SMM12, p. 2434], courtesy of IEEE © 2012.

visualization, proposing seven guiding scenarios for the evaluation. They derived these scenarios from extensive literature reviews and provided guidance in selecting appropriate evaluation techniques for a given information visualization. For each of these scenarios, they described the goals and outputs of evaluation in this type of scenario. They also provided possible sets of questions to be answered in such an evaluation as well as an overview of possible methods to apply and give examples to illustrate the practical application of each scenario. This work is widely used and considered a reference work in most evaluations in the field of visualization.

Another addition to the seven guiding scenarios by Lam et al. [LBI+12] is the systematic review on the practice of evaluating visualizations by Isenberg et al. [IIC+13]. The seven guiding scenarios are primarily focused on the literature of information visualization and, therefore, do not reflect the entire visualization community, which should also include scientific visualization and visual analytics. Isenberg et al. filled this gap by contributing their observations from their literature search covering the entire visualization field. They reported on the practices in evaluating visualizations and identified an emphasis on the evaluation of algorithmic performance and qualitative result inspections in the literature. They also drafted some trends in the evaluation practices, like the evaluation of user experience and user performance as well as reports on environmental and work practices as well as how a new visualization eases data analysis and reasoning. They still identified a lack of rigor in the evaluation of visualization solutions, although there was an improvement over the years.

Together, the papers by Lam et al. [LBI+12] and Isenberg et al. [IIC+13] provide a foundation for grounding the methods used and applied for evaluating visualizations and visual analytics solutions. They provide guidance on how to plan the evaluation as soon as the design phase and help carry out the evaluation in order to validate the visualization design and study the effectiveness and appropriateness of a proposed visual analytics approach.

Bridging from Goals to Tasks. A more recent addition to the previously mentioned set of papers is the contribution by Lam, Tory, and Munzner [LTM18] to help visualization and visual analytics researchers better bridge goals and tasks. They argued that the existing abstract task classifications and the usually applied bottom-up approach for task specification are in practice very challenging, because these low-level actions only make sense when put in perspective of the higher-level context of the analysis goals and the whole analysis process. The proposed framework relates the main analysis goals to lower-level tasks. This allows researchers to keep the bigger picture in mind when defining the low-level tasks during the design of visualizations.

In all our proposed visual analytics solutions discussed in Chapters 2, 3, 4, and 5, we used and applied the methodologies presented here. For all of them, we used the design study methodology and performed a task abstraction as well as described the data and the user by applying the previously discussed data–user–task framework. For the interactive model selection environment in Chapter 2 as well as the extension integrating the prediction capabilities in Chapter 3, we used a visual analytics process definition model, like the ones introduced in Section 1.3. For these two and the multivariate cycle plot for outlier detection in Chapter 5, we showed the utility through a illustrative walk-through and applied usage scenarios.

Following our discussion on the research methodology, we give some required background in time series analysis in the following section.

1.5 Time Series Analysis

Tsay [Tsa00] introduced the term *time series analysis* as the statistical analysis of time series data, with a history reaching back to Yule in 1927 [Yul27]; one of the goals of time series analysis—namely, forecasting—reaches back even longer. Tsay summarized the goals of time series analysis as analyzing the dynamic structures of the process, analyzing the dynamic relationship between variables, performing seasonal adjustment, improving regression analysis, and applying for forecasting. In addition, he identified the advances in methods and computing as having a major impact and allowing for outlier analysis and detection of structural breaks to get a central part in model diagnostics.

In the history of time series analysis (cf. [Tsa00, DGH06]), it is generally accepted that the breakthrough and most significant milestone was the book *Time Series Analysis: Forecasting and Control* published in 1970 by Box and Jenkins [BJ70], which even today in its fourth edition remains a reference work [BJR08]. Box and Jenkins integrated knowledge about time series analysis at that time and "delivered a coherent, versatile three-stage iterative cycle for time series identification, estimation, and verification" [DGH06, pp. 446–447]. It allowed practitioners to apply time series analysis by delivering this systematic approach, and it had an enormous impact on modern time series analysis and forecasting in terms of both theory and practice. We later employ this so-called Box-Jenkins methodology and use visual analytics techniques to support this kind of iterative model selection process (cf. Chapter 2).

There are many great and important textbooks on time series analysis, which build the basis of this section and are used as a foundation to most content in this dissertation. The main ones

are [BJR08, BD91, CC08, SS11, Ham94, BK11, BD02, Tsa10]. These are mostly based on the landmark work of Box and Jenkins [BJ70]. Before we give some details on model selection, we want to specify and define what we mean by time series data and what the terms *periodic*, *seasonal*, and *cyclic time series* mean. The exact definition and description of a time series varies in the vast number of books about time series analysis, but generally a time series is considered to be data collected or measured over time. These measurements can be a single value or multiple values and are, therefore, univariate or multivariate time series, respectively. According to the literature on statistical time series analysis, a time series is composed of 3 (or 4) components, which are trend, periodic, and irregular components [Cle93, Cle94, BD10, HA18]. The periodic component is the frequency of the number of observations before a periodic pattern repeats. By using the terms *periodic*, *periodic time series*, *periodic component*, etc., we subsume both periodic components—namely, the seasonal and cyclic components. According to Hyndman and Athanasopoulos [HA18], it is important to distinguish the terms *trend*, *seasonal*, and *cyclic* very carefully. In the following, we use the definitions of these terms from Hyndman and Athanasopoulos:

> "**Trend** A trend exists when there is a long-term increase or decrease in the data. It does not have to be linear. Sometimes we will refer to a trend as 'changing direction', when it might go from an increasing trend to a decreasing trend. [...]
>
> **Seasonal** A seasonal pattern occurs when a time series is affected by seasonal factors such as the time of the year or the day of the week. Seasonality is always of a fixed and known frequency. [...]
>
> **Cyclic** A cycle occurs when the data exhibit rises and falls that are not of a fixed frequency. These fluctuations are usually due to economic conditions, and are often related to the 'business cycle'. The duration of these fluctuations is usually at least 2 years." [HA18][2]

In addition to this definition, it should be noted that a possible seasonal and cyclic component could occur on any granularity level, such as days, weeks, years, decades, or centuries. The claimed duration of the fluctuation for a cyclic component of 2 years, should be defined more generally at a duration of at least 1.5–2 times the seasonal component. In addition, periodic components may occur nested on multiple granularity levels, such as seasonality on a daily level during a week and on a weekly/monthly level during the year. In summary, a seasonal time series has a fixed periodic length whereas a cyclic time series is a sequence with varying periodic lengths.

Historically, the field of time series analysis was originally divided into two schools of approaches: the time domain and the frequency domain [Tsa00]. The main idea of the time domain approach is to rely on the use of autocorrelation function and parametric models, whereas the frequency domain focuses on spectral analysis or power distributions. According to Tsay [Tsa00], the separation is gone and the determination of which approach to use is now actually based on the

[2] Website of the book by [HA18]. Quote from Section 2.3: Time series patterns, `https://otexts.com/fpp2/tspatterns.html` (last visited on Sept. 16, 2020)

objective of the analysis and the experience of the analyst. In the following, we focus on the time domain aspect of time series analysis and apply parametric models, as proposed and popularized by Box and Jenkins [BJ70]. These models are known as *autoregressive integrated moving average (ARIMA) models*. The systematic approach Box and Jenkins proposed allows practitioners to, quite simply, apply time series method for forecasting, and the success of these ARIMA models led to substantial research of time series analysis [Tsa00]. The basic model selection process by Box and Jenkins was a simple iterative process that included model identification, model estimation, and model diagnostics. We will describe this process in more detail later in Section 2.3.1 of Chapter 2. There we also elaborate on the challenges in the process of finding an adequate model for a given time series, but essentially it is necessary to find a "useful" and "adequate" model [BJ70] that can then be used and applied for tasks, like outlier detection, imputation, and prediction. In that chapter, we use and apply not only the previously mentioned ARIMA model, but also the extension for periodic time series, which are the class of seasonal ARIMA (SARIMA, or S-ARIMA) models. In this way we can fit the time series model to seasonal time series data.

In addition to the major impact by Box and Jenkins in 1970, Tsay identified advances in computing and time series methods as a reason for important developments around 1986–1988. In this period, there was a transition from "traditional" time series analysis to more advanced model diagnostics. This was achieved by integrating outlier analysis, applying the detection of structural breaks, the usage of model selection criteria, and some important advances in pattern identification methods [Tsa00]. In Chapter 2 we apply and use such model selection criteria to support the identification of adequate models. In addition to those criteria, we apply the model and use the results of the predictions as an integral part of the visual analytics model selection environment in Chapter 3.

In real-world applications and real-world data, data quality issues are of major concern [KHP+11, Sad13, Das13]. We focus in this dissertation on the issues of missing observations/data and outliers. As there is also a demand for still using time series analysis methods/models on such data, Tsay [Tsa00] mentioned the introduction of Kalman filters to time series analysis in order to cope with missing observations. The major contributions for handling such missing data in ARIMA models comes from the work based on alternative parameter estimation methods proposed by Jones [Jon80]. Jones was also concerned with using the time series models for unequally spaced time series and proposed similar techniques to apply such models in this case [Jon85]. The general idea of these techniques is to use specialized methods, like Kalman filtering, for the maximum likelihood estimation when estimating the model parameters. In addition, there are techniques to fill the missing values with arbitrary values and use a maximum likelihood estimation with additive outliers [GMPn99]. Another approach would be to apply imputation to fill in more appropriate estimated values for the missing ones instead of using arbitrary values. For this reason, in Chapter 4, we propose visually and statistically guided imputation by using visual analytics to better support the challenging task of imputation. We focus on seasonal time series data and apply a cycle plot representation, which allows for better judging the estimated value in relation to the seasonal structure in the time series.

Periodicity in time series data has always been of major interest in time series analysis. As mentioned before, Tsay [Tsa00] named Yule 1927 as the start of statistical analysis of time series data. In the referred paper [Yul27], Yule proposed a method for investigating periodicities, using

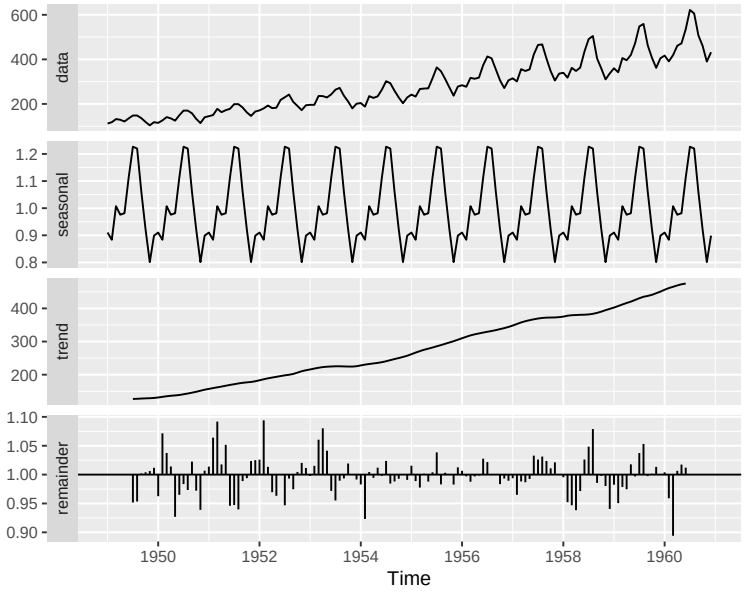


Figure 1.12: Seasonal decomposition of a periodic time series. The stacked line charts show the observed data, season, trend, and irregular components.
Image Source: Generated by Markus Bögl using `R` and `ggplot2`.
Data Source: The classical `AirPassengers` dataset by [BJR08], retrieved from the R `datasets` package.

the still well-known sunspot observation data. This dataset, which dates back to 1849, was introduced by Professor Wolf in Zürich [Yul27, Mor77]. The sunspot activity is known to have a periodic cycle of approximately 11 years [Mor77] and even in 1927 Yule [Yul27] was centrally concerned with the interesting periodic patterns in the sunspot data. He proposed a method to determine the periodic length of the sunspot cycle, but his method was challenged by the impact of the disturbances and fluctuations in the data. He stated that he was "attacking a problem which [...] was a new one, and used the methods that seemed best at the moment" [Yul27, p. 295].

Using our terminology of *periodic–cyclic–seasonal*, as defined before, dealing with periodic time series was and is a central concern in time series analysis. Dealing with cyclic periods that are not fixed in length was already considered from the beginning of the time series analysis. Dealing with seasonality, which means with fixed periodic length, is much more prominent and widely used and of major interest in many applications. There are seasonal fluctuations in many application domains, from economics over natural sciences to the health domain. Barnett and Dobson [BD10, p. vii] stated that "seasonality in disease was first recognised by Hippocrates (460–370 BC)," based on the information that "[a]ll kinds of diseases are produced in all seasons the year, yet some are caused and exacerbated rather in one than in another" [S+08, p. 55].

The common approach to dealing with seasonality in time series analysis is to apply seasonal decomposition [DGH06] for seasonal adjustment [CT82] to compensate for the seasonal fluctuations. Most of the economic time series data presented to us in different media, like unemployment rates,

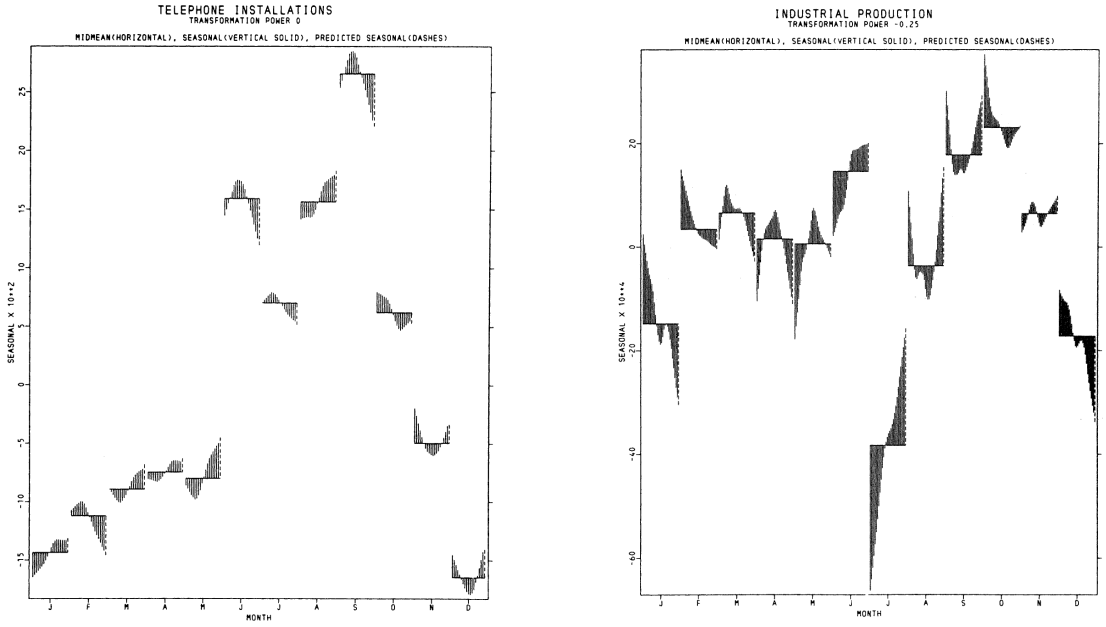


Figure 1.13: Seasonal subseries plot, later know as cycle plot, Cleveland and Terpenning [CT82]. This graph allows for investigating the seasonal pattern as well as the behavior in each subseries. Image Source: Cleveland and Terpenning (1982) [CT82, p. 55], courtesy of the Journal of the American Statistical Association © 1982.

price indices, production numbers, and shipments, are usually presented in this seasonally adjusted version [CT82]. The general idea is to separate the three components of a time series into trend, periodic (seasonal in most cases), and irregular (or remainder/residuals). In 1982, Cleveland and Terpenning [CT82] proposed some graphical methods for seasonal adjustment, where a first useful step is to show the results of the decomposition in a single display, as shown in Figure 1.12. A more detailed discussion on such displays and seasonal decomposition is presented in the later books by Cleveland [Cle93, Cle94]. Cleveland and Terpenning then introduced the so-called *seasonal subseries plot*, as shown in Figure 1.13, which allows for investigating the overall pattern of the seasonal component—in this case, the yearly pattern—as well as the behavior within each subseries plot—in this case, for each month. This type of graph was later adapted by Cleveland [Cle93, Cle94], who used the term *cycle plot*.

Bertin had already proposed a first indication of representing time series with obvious periodicity in a similar way in 1967 [Ber83, p. 214] (see Figure 1.14). He suggested that, "[i]f there is obvious periodicity [...], and the study involves a comparison of the phases of each cycle, it is preferable to break up the cycles in order to superimpose them [...]" [Ber83, p. 214]. Speaking in terms used in the visualization community today, Bertin is suggesting that the graph illustrates *superimposition* by *stacking* the lines above each other because superimposition would actually overlay the five lines in the correct position of the points on the y-axis. In contrast, the cycle plot proposed by Cleveland actually superimposes the data points and uses lines to connect the points

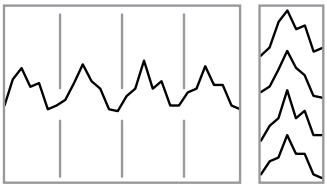


Figure 1.14: Showing obvious periodicity in cycles, Bertin [Ber83]. As early as 1967 Bertin [Ber83] suggested breaking up obvious periodicity and superimposing it to compare the phases of each cycle.
Image Source: Redrawn with different data by Markus Bögl, based on the graph by Bertin (1983, first published in 1967) [Ber83].

in a different way than Bertin is doing. In a cycle plot, the points in the same phase of the periodic cycle are connected (e.g., each month of a year). Together with the line representing the mean of each group, each of them actually represents small multiples of line charts, with the correct position on the y-axis and, like Bertin suggested, a restructured and differently arranged x-axis based on the periodicity.

We are using these techniques of representing periodic cycles in such a way in our proposed solution for visually and statistically guided imputation in Chapter 4. Because of the usefulness of this technique and the limitation of using univariate time series data, we propose a generalization to allow the usage of cycle plots for multivariate time series in Chapter 5. We then show the utility of this novel approach by applying it for multivariate outlier detection.

1.6 Research Questions

Thus far, we have illustrated and discussed the problems and challenges that have arisen in the field of time series analysis. We have also introduced the research field of visual analytics and the history of visualization, as statistical graphs, in time series analysis and exploratory data analysis. Applying visual analytics to the problem and challenges in time series analysis opens up numerous research opportunities. In this dissertation, we will answer the following research question:

> How can visual analytics support the challenges in statistical time series analysis of model selection, parametrization, prediction, imputation, and outlier detection?

In order to answer this comprehensive research question, it is necessary to break it down into sub-questions before consolidating them to answer the main question. These derived sub-questions are:

Sub-Question 1: Is visual analytics an adequate support for the challenges in statistical time series analysis dealing with periodic time series for both univariate and multivariate time series data?

Sub-Question 2: How can visualization and interaction improve the process of model selection and parametrization for time series prediction tasks?

Sub-Question 3: Is an adequate visual representation of periodic time series beneficial for imputation and outlier detection tasks in univariate and multivariate time series?

1.7 Aim and Contributions

The aim of this work is to provide an understanding of challenges in statistical time series analysis, introduce the principle ideas of visualization and visual analytics, and then apply visual analytics to provide support in visual time series analysis, applied to the problem of model selection, parametrization, prediction, imputation, and outlier detection. The larger structure of this thesis is encapsulated into a form inspired from classical music structure in music theory [MV15, p. 149]—namely, the sonata form of *exposition*, *development*, *recapitulation*, and *coda*. The background of this idea is that the classical sonata form is composed of these four parts, each of which has a specific purpose, as it is required in the structure of a dissertation. In the *exposition*, the thematic material is presented and introduced to prepare for the *development*, the lengthy main part where the themes are developed in great detail. In the *development*, the themes are changed, variegated, metamorphosed, contrasted, and opposed until they transform into *recapitulation*. The intention of *recapitulation* is to repeat and treasure the main thematic material and transfer it to the final part with no frills. The *coda* is the final part that, simply put, concludes the work.

In the first main part, *exposition*, we laid out the foundation by introducing the topic of visual analytics of periodic time series data in Chapter 1. After motivating the topic in general and describing the problem space, we provided the background on data analysis and visualization and offered a short history of time series analysis. We briefly summarized historic time series graphs and then focused on visualization in general before introducing visual analytics. We continued providing a foundation of the scientific methodology. Thereafter, we introduced some basic definitions in time series analysis, differentiating between periodic, cyclic, seasonal, and trend. Next, we stated our research questions tackled in this dissertation and described the aim of the work before summarizing the contributions together with an overview of the remaining structure of this document.

We then take the step to the second main part, *development*, were we propose our solutions for the problems stated earlier. The first chapter in this part covers the role of visual analytics for model selection in time series analysis in Chapter 2. The content of this chapter was published in Bögl et al. [BAF+13]. As previously explained, the first step in time series analysis is to find an appropriate model for the given data. There is a lack of adequate support for selecting such models and allowing parameter adjustment with immediate feedback on the fit of the model for the data. We introduce visual analytics methods to guide users in finding such adequate models and judging the adequateness of the model by incorporating visual cues for model diagnostic together with information criteria computations. We show that our prototype supports this model selection task through interactive visual interfaces with short feedback cycles.

In Chapter 3, we extend the model selection process and propose the integration of prediction

capabilities of time series models to improve the model selection process. The content of this paper was published in Bögl et al. [BAF+15]. In addition to the previous contribution, we extend the model diagnostic part for model selection and apply the model to compute prediction and provide the prediction visually in the context of the input time series to give additional insights into the adequateness and parsimony of the model for the model selection task.

We continue with introducing visually and statistically guided imputation of missing values in univariate seasonal time series in Chapter 4. The content of this paper was published in Bögl et al. [BFG+15]. The general idea of this contribution is to provide intuitive guidance for the task of imputing missing values in a time series. We utilize a cycle plot representation for visualizing results of statistical imputation for periodic time series data to benefit from different perspectives on the data and allow for a visual judgment of the adequateness of the computed values.

Finally, we propose extending the cycle plot representation to multivariate time series in Chapter 5 and utilize that for outlier detection in multivariate periodic time series data. The content of this paper was published in Bögl et al. [BFG+17]. In this contribution, we take the imputation idea using a cycle plot representation a step further and generalize the cycle plot for multivariate time series. We use a Mahalanobis-based distance measure to allow for multiple variables and represent this in our multivariate cycle plot. We then employ this representation to explore and interpret multivariate and univariate outliers and investigate seasonal cycles.

Following the main part of this dissertation is the *recapitulation and coda*, in which we summarize, conclude, and discuss in Chapter 6 the main contributions of this thesis, answer the research questions, and propose open challenges and future research directions. We end with a list of relevant publications together with an overview of the contributions to these papers of the author of this dissertation, *Markus Bögl*.

Part II

Development

CHAPTER 2

Visual Analytics for Model Selection in Time Series Analysis

Model selection in time series analysis is a challenging task for domain experts in many application areas such as epidemiology, economy, or environmental sciences. The methodology used for this task demands a close combination of human judgement and automated computation. However, statistical software tools do not adequately support this combination through interactive visual interfaces. We propose a Visual Analytics process to guide domain experts in this task. For this purpose, we developed the TiMoVA prototype that implements this process based on user stories and iterative expert feedback on user experience. The prototype was evaluated by usage scenarios with an example dataset from epidemiology and interviews with two external domain experts in statistics. The insights from the experts' feedback and the usage scenarios show that TiMoVA is able to support domain experts in model selection tasks through interactive visual interfaces with short feedback cycles.

The content of this chapter was published in [BAF+13]. © 2013 IEEE. Reprinted, with permission, from the authors. We modified "[...] we refer to [SS11, p. 121–140]" to "[...] see [SS11, p. 121–140]" in the last sentence of the last paragraph on page 36 to prevent a pagebreak within the citation.

Markus Bögl, Wolfgang Aigner, Peter Filzmoser, Tim Lammarsch, Silvia Miksch, and Alexander Rind. Visual analytics for model selection in time series analysis. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2237–2246, 2013.

The original version is available at `https://doi.org/10.1109/TVCG.2013.222`

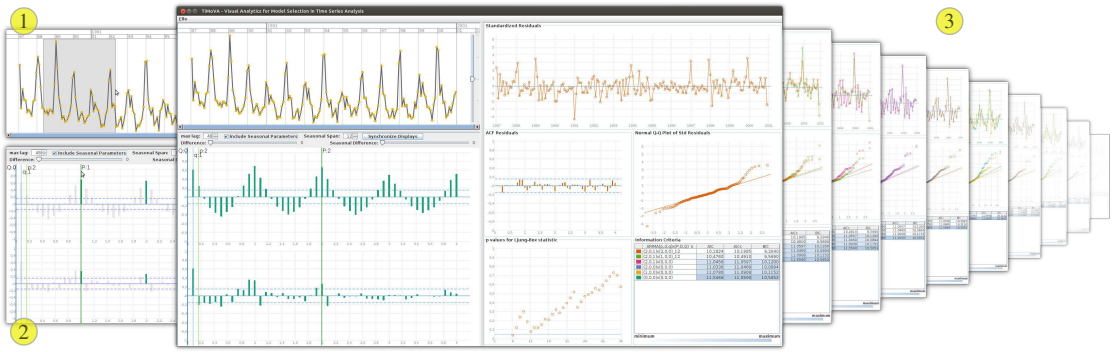


Figure 2.1: TiMoVA provides visual guidance for domain experts in the task of model selection, by (1) enabling to choose a certain range of interest in the time series, (2) supporting the model selection interactively via the visual interface, and (3) visualizing the model transitions and giving immediate visual feedback of the model output for the iterative refinements. For more details see Figure 2.5.

2.1 Introduction

Statistical time series analysis is a challenging task performed by experts in different domains. A practical application scenario is, for example, a public health official predicting the number of people that need to be treated because of cardiovascular reasons in the next year. Another scenario is to be prepared for the number of patients suffering from seasonal flu. The datasets to be analyzed are obtained from observations collected over time, optimally at periodic and equally spaced intervals and ideally without missing values. Such a dataset is called a time series. A range of methods, algorithms, and models to analyze these time series exist in the literature [BK11, BJ70, BJR08, CC08, SS11]. Moreover, they are implemented in most common software tools for statistical computing. In our work we focus on the most prominent and large class of models, namely ARIMA and seasonal ARIMA models [BJR08]. These models are applied in a variety of practical application domains. The huge amount of work discussing ARIMA models reflects the importance of this model class [BK11, BJ70, BJR08, CC08, Ham94, SS11] and there exists an established process for model selection known as Box-Jenkins methodology [BJ70]. We present and discuss this methodology and the theoretical underpinnings briefly in Section 2.3. However, currently available software tools do not appropriately support the workflow of the Box-Jenkins methodology, as we argue in Section 4.2. Therefore, support for domain experts would be beneficial for working on model selection in time series analysis.

A potential way to overcome the above-mentioned shortcomings is, in the spirit of Visual Analytics (VA), to "combine automated analysis techniques with interactive visualizations" [KKEM10, TC05]. This raises the question of how to use VA methods to support the task of model selection for time series analysis. According to the characteristics of design studies [Mun08, SMM12], we need a comprehensive understanding of the domain problem. We provide its characterization to judge our solution and analyze the tasks in Section 2.3. In Section 2.4 we

identify the target users, formulate the requirements for the tasks, and specify the data used in time series analysis.

The main objective of this work is to use well-established information visualization techniques [AMST11, Cle93] and apply them to the particular target problem. For this reason, we defined a VA process based on existing VA process descriptions [KKEM10, LAB+11] and implemented a prototype to facilitate it. The design and implementation of the VA process and the prototype were iteratively refined and judged by experts in information visualization and statistics using formative evaluation of user experience [LBI+12]. We present the results of this refinement process in Section 2.5, where we discuss the VA process description and our prototype. We named our prototype TiMoVA, which is an abbreviation of **Ti**me series analysis, **Mo**del selection, and **VA**.

In addition to the formative evaluation and the iterative design, we evaluated the final version of the prototype by defining usage scenarios and applying the prototype to an example dataset. We present the evaluation in Section 2.6, where we apply the usage scenarios on an example dataset. Furthermore, we also evaluated the user experience by performing a feedback session with two external domain experts. We used this informal user feedback to argue how our target users assess TiMoVA [LBI+12]. In the discussion of the results (Section 2.5) and the evaluation (Section 2.6), we describe the visual encodings and interaction mechanisms used in the prototype and how they fulfill our requirements.

In particular, the main contributions of our paper (Section 2.7) address issues of interactive visual guidance to ease the model selection in time series analysis by

- selecting the model order inside the autocorrelation and partial autocorrelation function plot, where the domain experts get the information about the model order,
- providing immediate visual feedback of the model results to the domain expert while adjusting the model order, and
- visualizing the model transitions to enable the domain experts to decide whether or not the model improves.

2.2 Related Work

Following the design study methodology [Mun08, SMM12], we apply existing and known VA methods and process descriptions to the domain problem of model selection in time series analysis. We based our work on the techniques for visualizing time-oriented data [AMST11, Cle93] and use state-of-the-art VA process descriptions [KKEM10, LAB+11, TC05].

All major mathematical and statistical software tools implement the state-of-the-art methods and models for time series analysis, which we describe briefly in Section 2.3. We considered tools, like the R project for statistical computing [R C20], SAS JMP [SAS12], MATLAB [The10], EVIEWS [IHS13], Mathematica [Wol13], Stata [Sta11], and Gretl [CL13]. Except for the R base package all these tools support time series analysis and models by menu driven user interfaces. The separate calculations and visualizations need to be initiated by the user either by menus and input forms, or by command line methods. What all these approaches are missing, is the guidance to browse and visually compare models directly when selecting the model. Instead it is

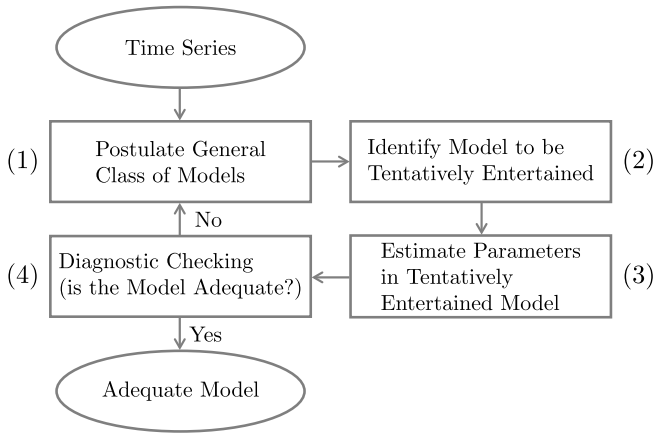


Figure 2.2: Box-Jenkins Methodology. An iterative process for model selection of time series [BJR08]. See Section 2.3.1 for details about the process.

necessary to either decide on a set of models or compute a whole bunch of models and arrange them in visualizations by hand to compare them. To summarize, they do not support the repeated execution of the separate steps in the iterative Box-Jenkins methodology [BJ70] well.

One notable solution is the `x12GUI` [KMST12] package for R. It offers an interactive graphical user interface for the `x12` [KM12] package, which provides a wrapper function to the X-12-ARIMA software. The focus of their approach is to explore the time series and the results of the seasonal time series adjustment as well as to enable the user to do interactive manual editing of outliers [KMST12]. However, the user interface supports the user in selecting a time series and adjusting the parameters for the X-12-ARIMA call using form-based input only. It also provides a history for computed models, which allows for loading previous settings but not to browse and directly compare them.

Because we have the very specific target problem of model selection in statistical time series analysis, to our knowledge, there is only distantly related work in VA, as for example TimeSearcher [BAP+05, BPS+07] and the work for visual-interactive time series preprocessing [BRG+12]. Both approaches apply VA methods to time-oriented data, but differ in the tasks and their solutions.

Motivated by these findings, we give the necessary background information in the following section to characterize and specify our target problem in more detail and ground our motivation further.

2.3 Problem Characterization

In this section we provide the characterization of the domain problem [SMM12] and discuss the tasks necessary for the model selection.

Example Dataset and Task. An important domain where time series analysis is used is public health and epidemiology. We have chosen a dataset from this domain for the evaluation in Section 2.6.1 and illustrate a possible analysis task in the following. The dataset contains the daily number of deaths from cardiovascular disease of people aged 75 and older in Los Angeles for the years 1987 to 2000 from the NMMAPS study [PW04, SDZ+00]. A possible scenario is that a public health official needs to predict the expected number of death from cardiovascular disease to start a prevention program. To do the prediction, the health official has to find a model based on the given data. The Box-Jenkins methodology is a standard method to solve this task. We describe this methodology in Figure 2.2 and in the following.

2.3.1 Box-Jenkins Methodology

The Box-Jenkins methodology [BJ70] is an iterative process to select an adequate model for a given time series (see Figure 2.2). It has been widely used in time series analysis and is an established method for model selection [BK11, BJR08, CC08, SS11]. To find or select a model for a given time series, it is according to Box et al. [BJR08] necessary to:

(1) Use the incomplete theoretical knowledge about the underlying mechanisms and the experience from theory and practice to consider a useful general class of models. By general class of models Box et al. [BJR08] mean any subclass combination of ARIMA and seasonal ARIMA model components. Fitting these models directly to data, would be too extensive and time consuming.
(2) Apply methods to select an appropriate parsimonious (see below) model by deciding on the model order. This determines the number of parameters in the model and gives some rough estimates for them.
(3) Fit the model to the data and estimate its parameters.
(4) Finally, check the model with diagnostics to uncover possible lacks of fit and find their causes.

These steps are repeated until an adequate model is found, which can subsequently be used for forecasting (see Figure 2.2).

The method is sometimes reduced to a simplified version [CC08], where the decision for a general class of models (1) and the identification of a model that can be tentatively entertained (2) are combined and entitled as *model specification*. Step (3) is renamed to *model fitting* and step (4) to *model diagnostics*. We present a more detailed description of these separate steps in Section 2.3.3, 2.3.4, and 2.3.5.

The crux of model selection is summarized in the famous quote of George Box that "Essentially, all models are wrong, but some are useful" [BD87, p. 424]. When introducing the Box-Jenkins methodology, Box and Jenkins [BJ70] use a language that indicates that there is a "useful" and "adequate" model for a time series, but we cannot assume that it is a "true" or "correct" model. The uncertainty of such models is put straight by using the term "tentatively entertaining a model" [BK11] (see Section 2.3.4).

Principle of Parsimony. An important principle in the model selection process is the principle of *parsimony* [BJR08]. It is described by the paraphrased quote of Albert Einstein "everything

should be made as simple as possible but not simpler" [BK11, p. 18]. In the process of model selection this means that if there are different candidate models to adequately represent the time series, the model with the least parameters is preferable [CC08].

In this section, we presented the methodology how to find an adequate model for a given time series. This methodology was introduced for a specific class of models [BJ70], which we describe in the following section.

2.3.2 ARIMA and Seasonal ARIMA Models

With classical regression it is often not possible to explain a time series sufficiently [SS11]. Therefore, alternative models exist. We briefly describe the key ideas of the different models and explain them without presenting the full details of the formal definition of these models. These formal definitions and the details are not necessary to understand and argue the design choices and explain the interpretation of the visual representations in Section 2.5 and 2.6. For more details and the formal definitions we refer to the literature in time series analysis [BK11, BJ70, BJR08, CC08, SS11].

Autoregressive (AR) models explain the current value x_t as a function of p past values in the time series. The number of past values p is also called model order, therefore an AR(p) model is an autoregressive model of order p. *Moving average* (MA) models explain the current value x_t as a linear combination of the current white noise term and the q past white noise terms. The number of past white noise terms is again called the model order, therefore, an MA(q) model is a moving average model of order q.

In some cases it is problematic to model a time series with only AR or only MA models, because it would demand a high-order model with many parameters. This is in conflict with the principle of parsimony. For these cases, Box and Jenkins [BJ70] presented *autoregressive moving average* (ARMA) models. To achieve parsimony, ARMA models combine the ideas of AR and MA models. An *autoregressive moving average* ARMA(p, q) model of order p and q, is the combination of an AR(p) model part of order p and a MA(q) model part of order q. It is possible to apply this class of models if the time series is stationary, which means that there is no seasonal effect or trend.

In many practical cases, a time series is non-stationary due to trends. It is possible to transform this time series to a stationary time series by applying a differencing operation, sometimes called detrending. To recover the original time series, the differenced time series needs to be aggregated, or also called integrated. These models are called *autoregressive integrated moving average* (ARIMA) models. An ARIMA(p, d, q) means that the dth difference of the time series is an ARMA(p, q) model.

To include seasonal terms in a model it is necessary to combine an ordinary non-seasonal ARIMA model with an ARIMA model that is extended to the seasonal period s. Therefore, the AR and the MA models are extended to the time shifts, called lag, of the seasonal period s and use capital letters P and Q for the seasonal model order. The seasonal difference D is also applied like the non-seasonal difference d, but with time shifts of the seasonal period s, called seasonal lag. Thus

the additive seasonal effects are removed. The resulting models are called *seasonal autoregressive integrated moving average* (SARIMA) models and are denoted as ARIMA$(p, d, q) \times (P, D, Q)_s$.

After introducing the Box-Jenkins methodology in Section 2.3.1 and the class of models used in that methodology in this section, we discuss the separate steps of this iterative model selection process in the following Sections 2.3.3, 2.3.4, and 2.3.5.

2.3.3 Model Specification

For the task of model specification, the goal is to decide on a class of models that could be appropriate for the given time series, select the level of differencing and determine the order of the model which specifies the number of parameters used in the model. The first step to achieve this goal is to take a look at the given time series. Usually this is done by viewing the time series in a line plot. After applying all required transformations, such as a difference operation or log transformation, the *autocorrelation function* (ACF) and *partial autocorrelation function* (PACF) plots are checked to support the decision of the model order.

ACF/PACF Plot. The ACF plot is a spike graph, which is a special type of bar chart, of the ACF as a function over lags. The PACF plot is likewise the PACF as a function over lags. For the formal definition of the ACF and the PACF we refer again to the literature in time series analysis [BK11, BJ70, BJR08, CC08, SS11] and give a description of the basic ideas in the following: The ACF is the correlation between any two values in a time series with a specific time shift, called lag. The PACF is the correlation between any two points with a specific lag, where the linear effects of the points in between is removed. This PACF plot combined with the ACF plot, where this linear dependence is included, is called ACF/PACF plot and enables us to choose the number of parameters for the model. In addition to the time series plot, the ACF/PACF plot provides a first idea for the level of difference and seasonal difference. In Figure 2.3 we show an example ACF/PACF plot. The ACF and PACF are plotted on the y-axes and the lags on the x-axes. In this case the labels are seasonal lags, which means that one lag represents one seasonal cycle. The non-seasonal lags are fractions of one, depending on the seasonal length. For example,

Table 2.1: ACF and PACF Behavior for ARMA and Seasonal ARMA Models [SS11]. The behaviors of the ACF and the PACF indicate which class of model and what number of parameters could be adequate for the non-seasonal and seasonal part of the model.

	AR(p)	**MA**(q)	**ARMA**(p, q)
ACF	Tails off	Cuts off after lag q	Tails off
PACF	Cuts off after lag p	Tails off	Tails off

	AR$(P)_s$	**MA**$(Q)_s$	**ARMA**$(P, Q)_s$
ACF*	Tails off at lags $k \cdot s$	Cuts off after lag Qs	Tails off at lags $k \cdot s$
PACF*	Cuts off after lag Ps	Tails off at lags $k \cdot s$	Tails off at lags $k \cdot s$

* where $k \cdot s$ are multiples of s, for $k = 1, 2, \ldots$

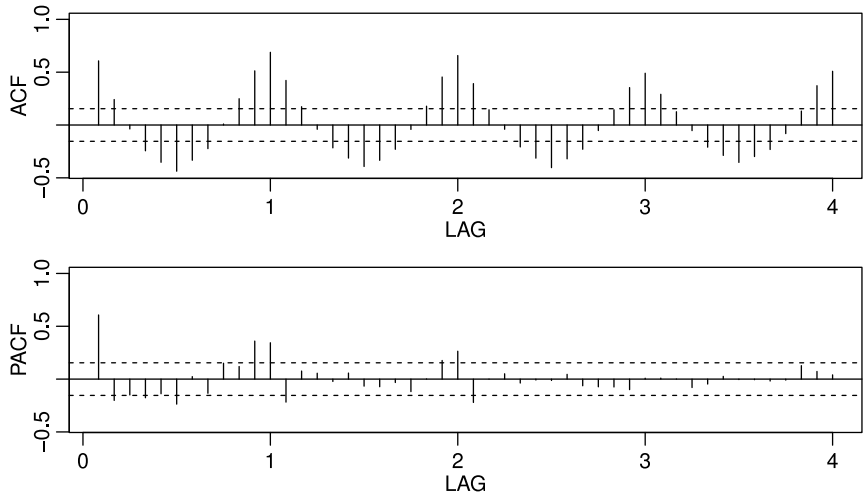


Figure 2.3: ACF and PACF over Lags. The behavior of the lags enables domain experts to decide on the order of the model according to Table 2.1. This plot displays the example dataset (see Section 2.6) from the NMMAPS study [PW04, SDZ+00].

in a dataset with 12 months in one year and a seasonal length of 12, the seasonal lags are $1, 2, \ldots$ and the non-seasonal lags are $\frac{1}{12}, \frac{2}{12}, \ldots$.

Using the definitions of the autoregressive models, moving average models, ACF, and PACF it is possible to identify the basic behavior of the ACF and the PACF for AR, MA, and ARMA models [SS11]. Likewise, it is possible to describe the behavior for the seasonal component of the model in a similar way. The behavior is shown in Table 2.1. If we consider the seasonal lags $1, 2, \ldots$ in Figure 2.3, we notice that the ACF plot is tailing off and in the PACF plot cuts off at lag 2. According to Table 2.1 this indicates, that an adequate model for the seasonal component could be an $\mathrm{AR}(2)_{12}$ model. Continuing this for the non-seasonal lags, we get a set of possible adequate models.

2.3.4 Model Fitting

In the previous section, we discussed how we select and configure the model order. The result is a model, for example a seasonal $\mathrm{ARIMA}(p, d, q) \times (P, D, Q)_s$ model, where the level of difference d, the level of the seasonal difference D, the seasonal length s, the number of parameters p and q, as well as the number of seasonal parameters P and Q are set according to the steps presented above. Note that the parameters p, q, P, and Q determine the order of the model, which is the number of parameters. The differences d and D are transformations to the time series. Box et al. [BJR08] use the term *tentatively entertained model* for this. Once the model is identified, it is fitted to the time series data to estimate the unknown parameters of the model. There are several methods to estimate the parameters. The most important one is the *maximum likelihood-estimation*. Other methods are the *method of moments*, the *least squares estimation*, and the *unconditional least squares*. For details and theoretical discussion see [SS11, p. 121–140].

2.3.5 Model Diagnostics

To evaluate how well the model represents the underlying time series, model diagnostic methods are applied. The model is diagnosed by analyzing the residuals, which means the remaining part that is not explained by the model. The exploratory analysis of the residuals is done by plots as shown in (4a-d) of Figure 2.5. If the model is well fitted to the time series, the remaining part is expected to behave like white noise. This is assessed in four ways: (4a) The standardized residuals are plotted over time. Any non-random episodes can unveil a remaining underlying process. (4b) The ACF of the residuals is calculated and plotted over the lags to check that there is no remaining structure in the residuals. (4c) If the model is well fitted, the standardized residuals are expected to be standard normally distributed. This distribution is checked by the quantile-quantile plot [Cle93]. (4d) The plot of the Ljung-Box statistic [BP70, LB78] is a test that helps to check if the residuals for each lag are independent. If for all lags the p-values are not significant for a pre-specified significance level, indicated by the dashed line, it can be assumed that there is no remaining autocorrelation within the residuals. If all this is fulfilled, the model is well specified, otherwise the model needs to be readjusted. A clear decision is often not possible and is based on human judgement and experience.

Information Criteria. In addition to the diagnostic plots it is possible to examine information criteria. They provide a good basis to decide on the fitness of the model. Criteria that are often used include Akaike's information criterion (AIC), its bias corrected form, the AICc, and the Bayesian information criterion (BIC) [SS11]. In contrast to the AICc, which does behave very well for smaller samples, the BIC is well suited for larger samples. In order to get an adequate model, as introduced in Section 2.3.1, the goal is to select a number of parameters for the model, thus minimizing the criteria. Based on these values for different models, it is possible to decide on one of them tentatively. For more details on the information criteria we refer to [SS11, p. 50–53]. We show such information criteria in the graphical user interface of TiMoVA in area (5) of Figure 2.5.

According to our findings about the domain problem, related work, and expert feedback, we conclude that these tasks are currently cumbersome to execute by domain experts. It is evident, that combining the computations and visualizations with additional intuitive interactions and visual feedback improves the way to accomplish these tasks and supports the domain experts in their work. To achieve this, we analyzed the requirements and present them in the following section.

2.4 Requirements Analysis

In this section we explicitly describe our target users, distill the tasks and challenges for model selection discussed in Section 2.3 and formulate them using user stories as well as present the data used.

Target Users. Our target users are domain experts in any field using time series analysis, for example biology, chemistry, epidemiology, economy, or environmental research. The users have

to be knowledgeable in statistics and time series analysis. These skills are required to interpret and understand the visual representations of the time series and the time series models as well as the model fitting and the selection criteria.

Tasks. User stories help to formulate the requirements in a way that is easy to use in discussions within the project team, with customers, or other stakeholders. User stories evolved from the extreme programming (XP) software development methodology [Bec00] and have an important role in other lean and agile software development methodologies [Coh04, Coh10]. Usually in the beginning only high-level goals and requirements with a broad coverage are known. These goals and requirements are formulated as user stories. Through the process of refinement, the high-level user stories are broken down iteratively to smaller user stories until they are very specific. The high-level user stories with the broad coverage are also called epics [Coh04, Coh10].

We formulated and refined the user stories in the repetitive meetings of the project team and throughout the formative evaluation. We used these user stories to formulate the VA process and implement the prototype. The stories are written from the perspective of our target users. The high-level goal is formulated in the following epic:

> *A*s a domain expert (user), I want to find an adequate model for a given time series so that I can use that model for different purposes, e.g., forecasting.

User stories are defined to get a more detailed understanding of the requirements [Coh10]:

- *A*s a domain expert (user), I want to select a certain region in the time series so that I can use any subregion of the time series for the model selection step.
- ..., I want to see all important visualizations of the time series and the model so that I can decide on the model and assess how well the model fits the time series with one glimpse.
- ..., I want to adjust the model orders at the place where the visualization provides the information about these model orders so that I can intuitively find an adequate model.
- ..., I want to include and exclude the seasonal components of the model and the seasonal parameter inputs so that I can compare the seasonal influence and if no seasonal components are needed, they do not distract me.
- ..., I want to see how a new selected model compares to the previous model so that I can decide if one model is better than the other.

Time Series Data. The considered data for our work are time series as introduced in Section 4.1. We assume to have univariate data values observed at equidistant discrete time intervals without missing values.

2.5 VA for Model Selection in Time Series Analysis

In Section 2.3 we discussed the characterization and tasks of the problem domain. We identified VA as a basis to define a VA process description to overcome the stated problems. In this section and the following Section 2.6 we rely on our findings to present the main contributions and results of our work. To do so, we provide the description of a tailored VA process in Section 2.5.1 that is

used for the implementation of the prototype. In Section 2.5.2 we provide the final design and a discussion of the design choices and interaction techniques in the prototype. These results are designed and implemented to fulfill the requirements specified in Section 2.4.

2.5.1 VA Process Description for Model Selection

The basis of our VA process description for model selection is the Box-Jenkins methodology presented in Figure 2.2 and discussed in Section 2.3.1. The process description we show in Figure 2.4 is the application of the VA process for time-oriented data [LAB+11] on the domain problem of model selection in time series analysis. The goal of the process is to select an adequate model for a given time series. The details of the theoretical underpinnings of this process in statistical time series analysis are briefly discussed in Section 2.3. In the following we describe the VA process for model selection in detail to prepare the reader for the discussion in Section 2.5.2 about the connection between the graphical user interface of the prototype (Figure 2.5), the VA process description for model selection (Figure 2.4) and the Box-Jenkins methodology (Figure 2.2).

The time series in Figure 2.2 is *Data* provided as *Input* in Figure 2.4. The *Domain Knowledge* is based on experience and *Prior Analyses*. The *Interactive Visual Interface* is used to visualize the *Data* (D_i) to decide on the class of models and adjust the number of parameters as well as the level of differencing. To interpret the *Interactive Visual Interfaces*, the *Domain Knowledge* about time series analysis (K_t) and about visualizations of time series and time series models (K_p) is used. Based on this knowledge and the visual representations of the time series and time series model, the *Hypotheses* are formed (V_t). By adjusting the level of differencing (A_d) and the order of the model (A_p), the *Hypotheses* are refined. Based on the *Hypotheses* the model is estimated with the given parameters (B_m) to build a model based on the input *Data* (A_i). The resulting model is analyzed using the *Domain Knowledge* about time series models and model

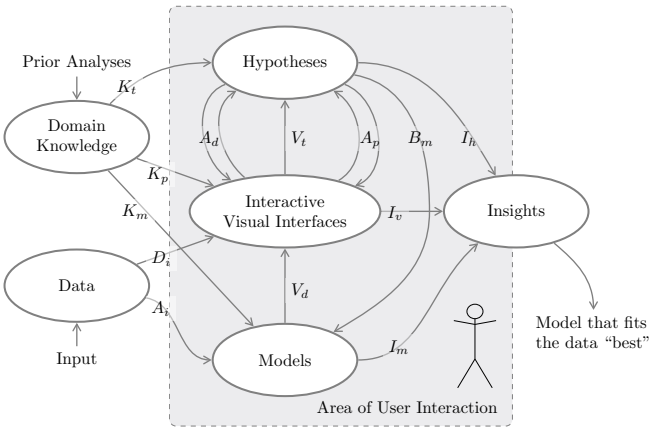


Figure 2.4: VA Process for Model Selection. The figure shows the VA process for selecting the model iteratively, to find an adequate model for a given time series. This process description is based on the VA process for time-oriented data [LAB+11].

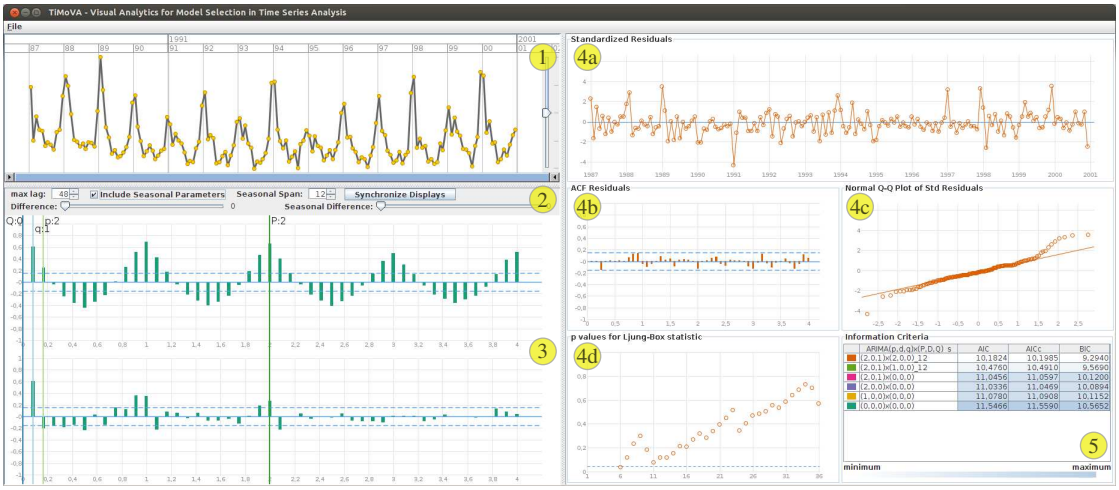


Figure 2.5: TiMoVA Overview. The figure is showing the coordinated and multiple views in the user interface, where (1) is the time series plot (input data), (2) the model selection toolbox, (3) the ACF/PACF plot as well as further model selection, (4a-d) the residual analysis plots, and (5) the model history including the information criteria. The plots in the area for the residual analysis are (4a) the standardized residuals over time, (4b) the ACF of the residuals over the lags, (4c) the quantile of the standardized residuals against the quantile of the standard normal distribution, and (4d) the p-values of the Ljung-Box statistics over lags.

diagnostics (K_m) as well as the visualizations of the standardized residuals, information criteria, and model parameters (V_d). In this iterative refinement of the process, *Insights* are gained by (1) interpreting the *Interactive Visual Interfaces* (I_v) deciding the fitness of the underlying model that is visualized, (2) the parameter estimations which lead to the adequate model (I_m), and (3) the refinement process of the *Hypotheses* building (I_h). The result is a model with estimated parameters, that is adequate for the given time series, and can be used for forecasting. The *Area of User Interaction* is highlighted in gray and indicates the process steps, where the user is part of the process through user interaction.

2.5.2 TiMoVA Prototype

Based on the VA process description above and the user stories (Section 2.4) we implemented our TiMoVA prototype. We refined our design along with the formulation of the user stories to quickly develop a working prototype and acquire user feedback [SMM12]. Here we present the final version.

Implementation Choices

Before we discuss the design choices and implementation ideas in detail, we highlight the most important design decisions for our prototype. We implemented the prototype in Java. For

the dynamic visualizations, we used the software framework prefuse [HCL05]. As an API for time-oriented data we used TimeBench, a software library for time-oriented data [RLA+13].

One important decision was to use the R project for statistical computing [R C20] as a comprehensive toolkit for time series analysis and other calculation tasks. R provides a broad variety of methods known from literature and our prototype is designed in a way that allows us to use these methods for the statistical computations. Using Java/R Interface (JRI) enables us to use R in combination with Java. JRI is part of the rJava package [Urb20] in R. We chose Java, because with prefuse, we have more possibilities regarding interactivity than implementing in R. Furthermore the extensibility and interconnectivity of our other projects using TimeBench is given.

Because the calculations in time series analysis can be very time-consuming, especially with large datasets, it is important that the user interface is still responsive to user input, while calculations are carried out. This is achieved by using Java threads and caching, which allow the computer to pre-compute models and provide them upon request. As a result, the user interface shows good reaction times for user input, even when the calculations are running in the background.

Graphical User Interface

The graphical user interface of the prototype is based on the workflow of the VA process description we defined in Section 2.5.1. The visualizations are inspired by the visualizations used in R and by well-established visualization techniques [AMST11, Cle93]. We extended these visualizations so that the user is able to interactively select the models. The result is a prototype that implements the VA process for model selection in time series analysis.

The TiMoVA prototype consists of coordinated and multiple views [Rob07]. An overview of the graphical user interface and the five areas is shown in Figure 2.5 and in the supplementary video of the usage scenarios in Section 2.6.1. In Figure 2.5, (1) displays the time series plotted over time (time series plot). In this view it is possible to explore the time series and select a certain range that is used for the model selection. This range selection is shown in the upper left corner (1) of Figure 2.1. The details of the interactions for the range selection are discussed in Section 2.5.2. The toolbox in area (2) and the ACF/PACF plot in area (3) are used for the model selection. In the ACF/PACF plot (3) the user can adjust the number of model parameters directly within the plot. The plots in area (4a-d) show the results of the parameter estimation as the plots for the residual analysis. The table in area (5) displays the model history including the information criteria.

In Figure 2.6 the model selection toolbox (2) and the ACF plot (3) are shown in more detail. These are the areas for the configuration of the model order. In the toolbox the *max lag* input changes the number of lags in the ACF/PACF plot below and in the ACF plot of the residuals in area (4b). The *Include Seasonal Parameters* check box enables or disables the configuration of the seasonal component in the model, which also enables or disables the input for the *Seasonal Span*, as well as the *Seasonal Difference* slider. With the *Difference* slider and the *Seasonal Difference* slider the numbers for the parameter d and seasonal parameter D are selected. The continuous vertical lines in Figure 2.6 can be dragged along the x-axis to select the order of the model, which is synonymous with the number of parameters. There is one vertical line for p, which is the order of the autoregressive part of the model AR(p). There is another vertical line for q, which is the

order of the moving average part of the model MA(q). If the seasonal components are enabled by the check box, two additional continuous vertical lines appear, one for P, which is the order of the autoregressive part of the seasonal component of the model AR$(P)_s$, and another one for Q, which is the order of the moving average part of the seasonal component of the model MA$(Q)_s$. The seasonal span s can be adjusted using the *Seasonal Span* spin box in the toolbox.

TiMoVA shows visual representations for the model diagnostics (4a-d) and (5), in order to evaluate the fitness of the model for the given time series. In this area the results of the parameter estimation are shown. We discussed the model diagnostics and the visual representations used for this task in Section 2.3.5. The goal is to visually explore the remaining part of the time series that is not described by the model, and check if it is likely to be white noise. The visual representation of the standardized residuals in the user interface of the prototype is inspired by the representation used in R. In Figure 2.5 the area displaying the plots for the analysis of the residuals are numbered with (4a-d). See Section 2.3.5 for the details on these plots.

In addition to the residual analysis, we included the information criteria, as introduced in Section 2.3.5, in the design of TiMoVA. The information criteria table (5) in Figure 2.5 shows a history of previously selected models. The first column represents the color used in the transitions of the residual plots. The second column describes the model and the other columns show the values of the model information criteria. The cells of the model criteria are colored according to their value indicating whether this criterion for this model is better (minimum) or worse compared to the others in the model history. The legend is shown below this table.

Residual analysis and tests for white noise, which are essentially tests for the randomness of a dataset, are manifold in statistics and there are many implementations of these methods in R. In our implementation of the prototype we focused on the standard tests and visualizations used in the literature for time series analysis [BK11, BJ70, BJR08, CC08, SS11]. It is desirable to enable the user to adjust and customize which tests and visualizations she or he wants to use in the process. This is a possible feature to include in the future work.

Connecting the VA Process Description and the Box-Jenkins methodology

In this paragraph, we describe how the VA process description defined in Section 2.5.1 is implemented in TiMoVA. We explain in detail how the user interface facilitates the VA process and creates short feedback cycles for the task of model selection. For each transition in the process, we provide the corresponding labels from Figure 2.4, the number of origin from the original Box-Jenkins methodology in Figure 2.2, and the number of the affected area in the user interface in Figure 2.5. By viewing the plots in the user interface, we decide on a general class of models (Figure 2.2: (1); D_i, V_t). By adjusting the level of difference and the number of model parameters (Figure 2.5: 2, 3; A_d, A_p, V_t), we identify a so called tentatively entertained model (Figure 2.2: (2); B_m). The adjustment of the relevant faders triggers the system to estimate the parameters of the model (Figure 2.2: (3); A_i) and show the resulting diagnostics immediately in the user interface (Figure 2.5: 4a-d, 5; Figure 2.2: (4); V_d, D_i, V_t). The insights gained (I_h, I_v, I_m) and the application of the domain knowledge (K_t, K_p, K_m) are part of the user interaction, but not part of the user interface.

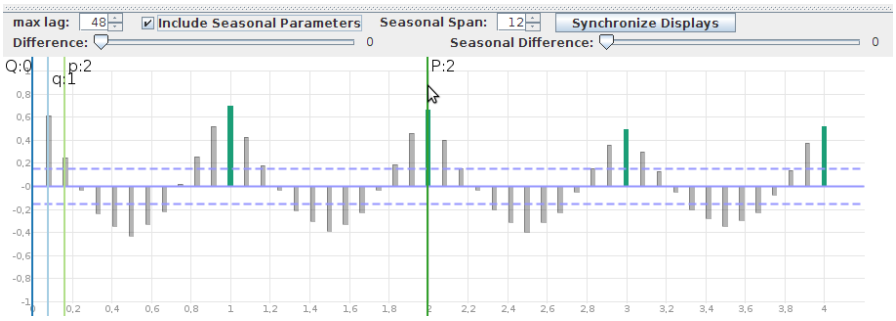


Figure 2.6: Model Selection Toolbox and ACF Plot. The toolbox at the top and the four continuous vertical lines are used to select the time series model. In this figure they are set to $p = 2$, $q = 1$, $Q = 0$, and P is currently moved from $P = 1$ to $P = 2$, which is the final model configuration for this time series. The user interface supports the user to focus on the seasonal lags by changing the color and reducing the opacity at the non-seasonal lags.

Interactive Guided Model Selection Environment

In Section 2.5.2 we introduced the time series plot and the range selection as shown in the upper left corner (1) of Figure 2.1. The main interaction in this area is the navigation through the time series and the selection of a specific time interval. The horizontal range slider on the bottom allows the user to zoom in and navigate through the time series. When changing the zoom level on the range slider, the time axis is adjusted to show a suitable resolution of time. Details about the time points are provided on demand when moving the mouse cursor to its position. It is possible to specifically select a time interval that is used for the model selection. The user can select, resize, move, and remove the selection using the mouse cursor. The user can select whether or not the selection is connected to the other views using the *Synchronize Displays* button shown in Figure 2.6. If it is linked to the other views, they are recomputed and the visualizations are updated as soon as the region changes. This feature enables the user to select a certain smaller region of interest from a larger time series for the model selection task and keep a fast reaction time for the model parameter estimation even for larger time series.

Another important design requirement was to visualize the change in the plots when adjusting the model order and give the user the control over these transitions. This is supported by direct manipulation [Shn83] using sliders for the level of difference and continuous vertical lines for the model parameters. To focus on the change in the resulting plots, we use animated transitions [HR07] and different colors that are consistent in the coordinated and multiple views [Rob07]. This process is shown in Figure 2.7 and the supplementary video. Once the slider or a vertical line is dragged from one value in the direction of the next value, the parameters of the new model are calculated and the plots are seamlessly faded from one display to another by using alpha blending of the bars, points, and lines. The colors for the plots are selected by using ColorBrewer2 [HB03]. We used this tool to get a qualitative color scheme, which is easy to distinguish on a screen. This set of colors is used as an endless cyclic sequence for the coloration of the plots. This ensures that each parameter combination is a different color, and the fading

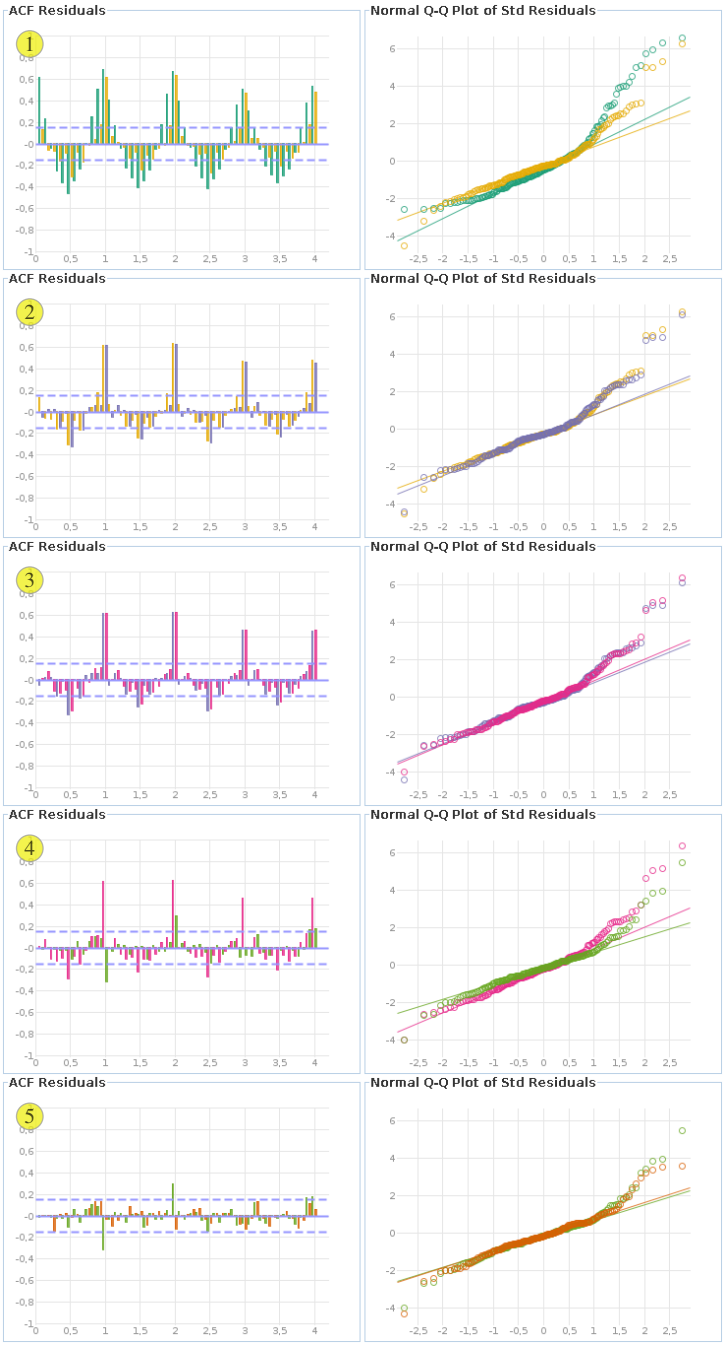


Figure 2.7: Transitions of Model Selection in Elected Residual Plots. We show the change when selecting a new model in two elected residual visualizations, the ACF and the normal quantile-quantile plot of the standardized residuals. This enables the user to evaluate whether or not the new model improves. Each numbered row represents one transition.

process uses always two separate colors.

The toolbox to adjust the parameters and the continuous vertical lines in the ACF/PACF plot are shown in Figure 2.6. By default the check box to include the seasonal parameters is disabled. In this case, the seasonal span and the seasonal difference input are disabled and the vertical lines for the order of the seasonal autoregressive and the moving average component of the model are not visible. This ensures that the user does not accidentally fit a seasonal model, if a non-seasonal model is needed. By ticking the check box the inputs are enabled and the vertical lines in the ACF/PACF plot for the seasonal order appear. That also ensures to first consider simpler non-seasonal models according to the principle of parsimony and keep the users attention to the relevant class of models.

In Figure 2.7, we show the fading process when changing the order of the model by dragging one of the continuous vertical lines in the ACF/PACF plot. When sliding the vertical lines for the seasonal order P and Q in the ACF/PACF plot shown in Figure 2.6, the prototype supports focusing the seasonal lags in the ACF/PACF plot by setting the non-seasonal lags to another color and opacity level. Thus the user can more easily decide the seasonal order of the model. When adjusting the level of differencing we also change the underlying data for the ACF/PACF plot. In this case the ACF/PACF plot and the residual plots are fading from the current to the new configuration.

All four residual analysis plots, area (4a-d) in Figure 2.5, are included in the interactive fading process presented before. When the user modifies the model configuration, the residual plots are fading from one to the other continuously. This enables the user to see the changes of the model configuration and evaluate if the model fitness improves or worsens. This process is shown in Figure 2.7.

The information criteria for all previously and currently selected models are shown in area (5) of Figure 2.5 as described in Section 2.5.2. This history stack is filled during the model selection process. The coloring to immediately find the minimal information criteria is readjusted if a new model is added to the table. So for each transition we can see the values of the criteria which are supported by the color if it is better or worse than the previous one. Additionally, it is possible to see which are the best models according to the information criteria at each point in the model selection task.

Discussion of Design Rationales

The key idea for the layout of the TiMoVA user interface is to map the Box-Jenkins methodology and its workflow. This way of working and thinking is well-established and known by our target users, the domain experts. The main intention to use established standard visualization techniques for the separate steps in the Box-Jenkins methodology, is to avoid confusion and benefit from the experience the domain experts already have. By using familiar visualizations that they know really well, the domain experts can profit from the combination, layout, and especially the interactions of TiMoVA. In this stage of our work this was our goal and is our contribution. For future work it would be exciting to experiment with more advanced visualization techniques, use and include them in TiMoVA in order to further improve the task of model selection in time series analysis.

Another requirement was to use appropriate interactions in TiMoVA, so that it is easy and intuitive for the target users and they can concentrate on their task of model selection. According to Heer and Robertson [HR07] animated transitions in statistical data graphics are a way to engage the users and improve the perception of changes. They also suggest using alpha blending as a solution for possible occlusion. Because of their findings, we apply animated transitions in TiMoVA and use alpha blending because occlusions may occur in the transitions. These transitions are triggered and steered directly by the user inside the ACF/PACF plot.

One limitation of showing the transitions with animated diagnostic plots is that you actually compare the current model to one of the next possible models. Usually this is good enough because you immediately see if the more specific sub-model you think of is better or worse and by this preview the domain expert can decide if it is worth to go into this direction. To overcome the limitation of comparing only two models we provide a history table with all previously and currently considered models and the corresponding information criteria as shown in area (5) of Figure 2.5 and in more detail in the supplementary material. With this overview it is possible to compare more than two models according to their information criteria. In our design we planned for future work to enable the user to select up to three models and load their standardized residuals in the diagnostic plots to directly compare them visually. This should be sufficient, because according to Nazem [Naz88, p. 307] in about 87% of the time series only one or two models remain in the shortlist for adequate models and in about 97.6% three or less models remain.

2.6 Evaluation

During the design and implementation phase we evaluated the results by formative evaluations during repetitive meetings of the design team. This iterative refinement process was judged by the team members, experts in information visualization and statistics, and the user experience [LBI+12] was discussed by performing demonstrations. In addition to this repetitive internal assessment, the user experience was evaluated by informal user feedback [LBI+12] consulting two external domain experts. Besides this first level of evaluation, we evaluated the prototype by defining usage scenarios and applying the prototype on an example dataset. We explain the example dataset below and apply the usage scenarios in Section 2.6.1. We discuss the insights gained from both levels of evaluation, what we learned, and how we can improve our solution further in Section 2.6.2. Accordingly we assess the usability and applicability of our solution for its target users.

Example Dataset. In Section 2.3 we introduced the example dataset of the daily number of deaths from cardiovascular disease from the NMMAPS study [PW04, SDZ+00]. The original dataset contains the data of different cities in the United States of America, but we focused on the number of cardiovascular disease deaths in Los Angeles only. The relevant columns in the dataset are *date* and *cvd*, which is the daily number of deaths from cardiovascular disease. There are no missing values in the dataset. For the evaluation of the prototype, we aggregate the time series to get monthly sums.

2.6.1 Usage Scenarios

Based on the requirement analysis in Section 2.4 we used the user stories and defined two usage scenarios for the evaluation. The first one is the high-level task of model selection. The second one is the task of selecting a range in the time series before selecting a model. We describe how the prototype is applied on the example dataset to solve these tasks. We demonstrate the usage scenarios from the perspective of a fictional domain expert. Because it is difficult to show the interactivity and visual feedback in static pictures, we support the textual description with the transitions in two of the residual plots in Figure 2.7. To clarify the interactions and how all visual representations behave during the transitions, we provide a supplementary video[1] with audio narration. Another supplementary material shows the model diagnostic area from TiMoVA for each of the five transitions that we discuss in the following.

Model Selection

Following the Box-Jenkins methodology presented in Section 2.3.1, we first consider the time series plot and the autocorrelation function (ACF) and partial autocorrelation function (PACF) plot. According to the time series plot in TiMoVA (Figure 2.5), we assume that no differencing operation (see Section 2.3.2) may be needed, because the time series seems to be stationary. Moving the difference slider confirms that the change in the ACF/PACF plot, as well as the residual analysis plots is marginal and, therefore, supports this hypothesis.

We evaluate the ACF/PACF plot, (3) in Figure 2.5, according to the behavior of the non-seasonal order of the model in Table 2.1. We show the transitions of this usage scenario in Figure 2.7 beneath each other with the color code explained in the information criteria table in area (5) of Figure 2.5. For the non-seasonal component of the model we decide to have a mixed ARMA model. When sliding the parameter p, the diagnostic plots shows that the adjustment of the non-seasonal AR model to order $p = 1$ results in a more random appearance of the residual time series plot, a more straight line behavior in the normal quantile-quantile plot, and lower lags in the non-seasonal lags of the ACF plot (Transition 1 in Figure 2.7). In addition to further improvement of those residual plots, the Ljung-Box statistics shows more lags with p-values significantly different from zero, if the order is changed to $p = 2$ (Transition 2 in Figure 2.7). See the supplementary image and video for further details on this. For both transitions we can see in area (5) of Figure 2.5 how the information criteria improve. The result is the model configuration $p = 2$, which is an AR(2) model. By adding a MA component of order $q = 1$ to the model (Transition 3 in Figure 2.7) we get the diagnostic plots for the assumed mixed ARMA model with $p = 2$ and $q = 1$. This configuration advances the model to show more randomness in the diagnostic plots of the standardized residuals, which strengthens the assumption that they are standard normal distributed. The information criteria in area (5) of Figure 2.5 get minimal worse when adding the MA component $q = 1$, but according to the behavior of the ACF/PACF plot in Figure 2.5 we assume a mixed ARMA model for the non-seasonal part. To see more details of these transitions, we provide all of them for the diagnostic plots in a supplementary image and in the supplementary video.

[1] `http://www.cvast.tuwien.ac.at/TiMoVA` (July 30, 2013) [revisited on Oct. 12, 2020]

The seasonal behavior is not covered by the model yet. Therefore, the next step is to adjust the seasonal parts of the model. We consider an autoregressive model, because sliding the parameter P highlights the seasonal lags and unveils the cut off on seasonal lag 2 in the PACF. This indicates, when consulting Table 2.1 for the behavior of the seasonal order of the model, that the seasonal component is likely to have order $P = 2$. Sliding from seasonal order $P = 0$ to $P = 1$ (Transition 4 in Figure 2.7) and from $P = 1$ to $P = 2$ (Transition 5 in Figure 2.7), shows the improvement of the selected model. In area (5) of Figure 2.5 we also recognize the abrupt improvement in the information criteria when including the seasonal component with $P = 1$ and the gradual improvement for the transition from $P = 1$ to $P = 2$. With this configuration, we get a seasonal model of the following form:

$$\text{ARIMA}(p = 2, d = 0, q = 1) \times (P = 2, D = 0, Q = 0)_{s=12}$$

Including the estimated parameters of the model, we get the following time series model:

$$(1 - 0.3068B^{12} - 0.5444B^{24})(1 + 0.3143B - 0.3112B^2)x_t = (1 - 0.9072B)w_t$$

In addition, the information criteria in area (5) of Figure 2.5 indicate that we found an adequate model. We selected the model with the minimum values for the AIC, AICc and BIC using TiMoVA. This goes along with the diagnostics based on the visualization of the standardized residuals.

Range Selection

TiMoVA enables the user to select a range of the time series, as we show in the upper left corner (1) of Figure 2.1. A possible usage scenario is to select a trend that starts and ends at a defined point in the time series and to consider only this trend for model selection.

In the following usage scenario we want to consider only the range starting with November 1994 and ending with the last data point in the time series. We select the model as described in the previous section. Compared to the complete time series we get a slightly different model, which is simpler and with fewer parameters. We get an intermediate model with $p = 2$ and seasonal $P = 1$. The ACF plot of the residuals indicates a remaining seasonal autocorrelation. In this case we consider a seasonal difference of $D = 1$ as a possible solution to remove this autocorrelation in the residuals. Sliding the seasonal difference fader, improves this model. Finally, we get a model with the following configuration:

$$\text{ARIMA}(p = 2, d = 0, q = 0) \times (P = 1, D = 1, Q = 0)_{s=12}$$

Consulting the information criteria for this model supports that we have found an adequate model.

In addition to the insights gained from the application of TiMoVA in these two usage scenarios, we discuss the results of the user feedback in the following section.

2.6.2 Evaluating User Experience

For the evaluation of the user experience [LBI+12] we rely on the insights gained from the internal formative evaluation and iterative design during the design and implementation phase, where

we had repetitive meetings and discussions on the intermediate stages, and on the demonstration session with two external domain experts who are employed as scientists in an institution for statistics research. They both hold a master in mathematics and one a PhD in statistics. The informal evaluation [LBI+12] was performed by demonstrating the prototype with a well-known dataset for time series analysis. The domain experts gave immediate feedback to the features of TiMoVA. This feedback was noted on paper and reflected after the demonstration session, which lasted about one hour. We included the feedback in the design and implementation. We discuss the remaining suggestions in this section and consider them for future work in Section 2.7. In this reflection our findings from the usage scenarios are included as well.

A very useful feature according to the domain experts is the overview displaying all separate steps of the Box-Jenkins methodology. It provides all information necessary to decide on an adequate model. Because the model selection relies on the behavior of the ACF/PACF plot, it is very helpful to directly select the model order inside this ACF/PACF plot using the continuous vertical lines. The visual support of focusing on the seasonal lags for the selection of the seasonal component in the model was considered very beneficial. Another beneficial feature according to the domain experts is the immediate visualization and preview of the model results when changing the model order and especially the visualization of the transition from one model to another. This enables the domain experts to directly compare the current model to the new model and decide whether the model improves or not. A further benefit is the possibility to select a certain region of interest from a larger time series and consider only this subregion for the model selection task. This is not only very nice for selecting a representative or interesting subregion, but also to have faster reaction times even for originally larger time series.

There are also suggestions to further improve our work. The domain experts would like to see more statistics of the residuals on demand. The numbers are available in memory as a result of the computations and including them in the graphical user interface is going to be future work. Domain experts sometimes prefer to perform different tests and statistics for the residual analysis and would like to customize the selection from a set of test statistics using the graphical user interface. Our solution is prepared for this kind of request and implemented in a way to ensure exchangeable test statistics.

One suggestion during the iterative design and implementation phase was to include a history of the previously selected models and enable the user to get an overview and reload certain models. We included this concept in the design and the result is shown in area (5) of Figure 2.5. This feature of getting an overview on the previously selected models and compare more than two models on a more abstract level is very beneficial. The concept of loading certain models as an overlay in the diagnostic plots is considered as useful and a way to further improve the implementation.

Another feature demanded, is to show how the model performs. This is usually done by taking the first part of the time series and use the model to forecast a certain time range or repetitive single step ahead forecasts. This forecast is then shown along the given time series with the according confidence boundaries. At the moment we focused on the first part in time series analysis, which is finding an adequate model, and not the application of this model. Although the extension of using these forecasts as overlay in the time series plot is considered for further work.

What we learned from the evaluation is that for the target users, with at least a basic knowledge in statistics and time series analysis (ARIMA models), TiMoVA is easy to learn and understandable. For others it is necessary to study time series analysis in order to understand and interpret the visualizations. We achieved this in TiMoVA, as discussed already in Section 2.5.2, by focusing on the knowledge and experience the domain experts already have and support them with a well-established way of working and thinking, familiar visualizations, and appropriate interactions.

2.7 Conclusion and Future Work

The goal of our work is to use VA to support domain experts in the process of model selection. We identified that for the class of ARIMA and seasonal ARIMA models in the Box-Jenkins methodology there is no technique or tool that supports the workflow of the process in an intuitive and user-friendly way. Applying VA methods to this domain problem showed that we can support this task with interactive visual interfaces, short feedback cycles, and the visualization of the model transitions.

By evaluating our work, we discovered that the resulting VA process description and the TiMoVA prototype enables the domain expert to do easy and intuitive visual exploration and selection of time series models. These benefits are achieved by

- enabling the domain expert to select the model order interactively via the visual interface, inside the ACF/PACF plot, which provides a first idea of the model order,
- giving the domain expert immediate visual feedback of the model results while selecting the model order, and
- helping domain experts with the visualization of the model transitions to decide whether or not the model improves.

We also showed that the interactions are appropriate for the task and that the domain experts profit from the usage of a well-established model selection methodology and visualizations from their domain.

In Section 2.6.2 we discussed the insights gained from the evaluation of our results and developed ideas for future work. One improvement was to include information criteria measures in the graphical user interface in a history. Another idea for future work is to enable using this history to load any previous model and compare two of them. Further improvement is to extend the prototype to directly support different statistical methods for the residual analysis and enable the user to customize the residual analysis for their needs. The diagnostic of the time series model is currently limited to the diagnostic plots. For future work, it would be interesting to include the performance of the model for forecasting the diagnostic step.

Our solution is limited to data with equally spaced time series without missing values. In practical applications, the data often contains missing values and/or are not equally spaced. For future research we consider these limitations as inspiration to apply VA for model selection in time series with missing values and provide visual support for the methods to estimate them. Another interesting challenge for future work is to foresee the model selection support for multivariate

time series. Therefore, it is necessary to consider appropriate visualization techniques for this kind of data.

The final model we found in our usage scenarios is rated as a rather complex model by our domain expert. Finding this model using existing statistical software tools would have been very cumbersome and time consuming. Working with TiMoVA reduced the number of models considered, because the immediate visual feedback excluded already certain subclasses of models early in the model selection process.

Based on the insights from the evaluation, we discovered that the well-established visualizations used in the prototype have the benefit that the domain experts are used to work with these visual encodings. Therefore, they can focus on the task of model selection, which is guided by TiMoVA and improves their work.

CHAPTER 3

Integrating Predictions in Time Series Model Selection

Time series appear in many different domains. The main goal in time series analysis is to find a model for given time series. The selection of time series models is done iteratively based, usually, on information criteria and residual plots. These sources may show only small variations and, therefore, it is necessary to consider the prediction capabilities in the model selection process. When applying the model and including the prediction in an interactive visual interface it is still difficult to compare deviations from actual values or benchmark models. Judging which model fits the time series adequately is not well supported in current methods. We propose to combine visual and analytical methods to integrate the prediction capabilities in the model selection process and assist in the decision for an adequate and parsimonious model. In our approach a visual interactive interface is used to select and adjust time series models, utilize the prediction capabilities of models, and compare the prediction of multiple models in relation to the actual values.

The content of this chapter was published in [BAF+15]. © 2015 Eurographics. Reprinted, with permission of the authors, in accordance with the retained rights defined in the Eurographics exclusive license form. All parts of the article are used without revision or modification to the content, it was only adapted to fit to the overall formatting style of this dissertation. Original full bibliographic reference:

Markus Bögl, Wolfgang Aigner, Peter Filzmoser, Theresia Gschwandtner, Tim Lammarsch, Silvia Miksch, and Alexander Rind. Integrating predictions in time series model selection. In *Proceedings of the 6th International EuroVis Workshop on Visual Analytics, EuroVA@EuroVis 2015, Cagliari, Sardinia, Italy, May 25-26, 2015*, pages 73–77. The Eurographics Association, 2015.

The original version is available at `https://doi.org/10.2312/eurova.20151107`

3.1 Introduction

In time series analysis, the main goal is to find a model for a given time series and to apply this model to predict future values [BK11, BJR08]. In previous work [BAF+13], we introduced a Visual Analytics (VA) approach to support domain experts in the task of selecting adequate seasonal autoregressive integrated moving average (ARIMA) models. This class of time series models are widely used for prediction tasks, for instance predicting electricity prices [CENC03], system failure analysis [HXG02], and in different financial and medical domains [SS11].

During evaluation of the prototype it became apparent that including the possibility to perform actual prediction would improve the model selection process considerably. Integrating the prediction capabilities into the exploration environment offers another perspective on the adequacy of the model for a given time series and raises the confidence in the resulting model. In addition to our previous work [BAF+13], we integrate the prediction functionality in the model selection process (Section 3.2). This work is a refined and extended version of our preliminary ideas presented in [BAF+14]. Based on feedback and discussions we focus our contribution to integrate the prediction capabilities in the model selection process and to compare the prediction of multiple model candidates. We demonstrate the benefit of this approach in a usage scenario using a dataset about the water quality in the San Francisco bay [JC14] (Section 3.3).

To support domain experts in the task of model selection, we propose a VA approach that utilizes the prediction capabilities of the models. Our approach therefore provides visual interactive means to

- explore different types of predictions,
- explore differences of predicted and actual values, and
- compare the prediction of multiple time series models.

This helps to adjust and re-select the time series models.

3.2 Visual Analytics Approach

In our VA approach we propose a close coupling of the prediction capabilities with the visual model selection interface. Including predictions in the interactive exploration environment during the iterative model refinement enables domain experts to judge the prediction capabilities and select a parsimonious model with fewer parameters. The principle of parsimony [BJR08] needs to be considered during the model selection, to prevent models from getting too complex.

Our approach is based on the Box-Jenkins methodology [BJR08], which describes how to find an adequate ARIMA model for a given time series. A seasonal ARIMA$(p,d,q) \times (P,D,Q)_s$ model combines a non-seasonal ARIMA(p,d,q) with a seasonal ARIMA$(P,D,Q)_s$ model multiplicatively. Both have an autoregressive component (AR(p), AR(P)), a moving average component (MA(q), MA(Q)), a difference transformation (d, D). The seasonal length is specified by s. The parameters p, P and q, Q describe the model order of the AR and MA components and specify the number of parameters that are estimated by, for instance, a maximum likelihood estimator. For more details about the ARIMA models cf. [BJR08, SS11].

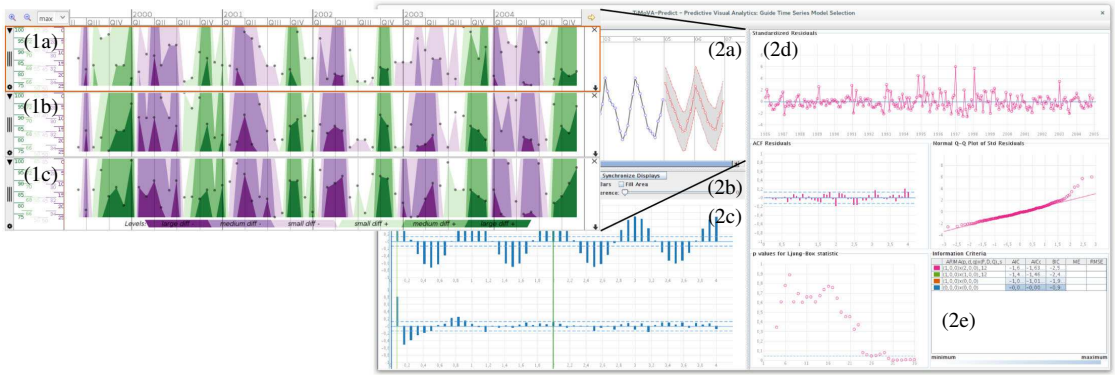


Figure 3.1: Interactive model selection environment, displaying the example data used in the usage scenario (Section 3.3). (2a-e) shows our prototype, where (2a) is the time series display showing the prediction of future values, (2b) is the toolbox for model selection and prediction, (2c) are the autocorrelation and partial autocorrelation plots for selecting the model orders, (2d) are the diagnostic plots for the residual analysis, (2e) is the model selection history including the information criteria. In our approach users can change the view of (2a) to the Qualizon Graph view, as shown in (1a-c) for visualizing the difference between the one-step-ahead prediction and the actual values. Each line, (1a, 1b, 1c) shows these differences for a different model (m1, m2, m3) respectively. In Section 3.3 we discuss the interpretation of these three possible models.

In general, the application of an ARIMA model for prediction is based on the available observations $x_1, x_2, \ldots, x_n$ at the time points $t_1, t_2, \ldots, t_n$. The predictions of the next m time points $t_{n+1}, \ldots, t_{n+m}$ are then denoted by $\hat{x}_{n+1}, \ldots, \hat{x}_{n+m}$, where m is an integer ≥ 1. Thus, the term predict refers to the predictions of these values, using the corresponding time series model. If we want to compare predictions with actual values, we mimic this process: The time series is split at time point t_k, with $1 < k < n$, the model parameters are estimated based on $x_1, \ldots, x_k$, and the predicted values $\hat{x}_{k+1}, \ldots, \hat{x}_n$ are computed. These values can be compared with the observed values $x_{k+1}, \ldots, x_n$. A variant is the one-step-ahead prediction, where k is set, e.g., to $n/2$ and step-by-step increased by one until $k = n$. In each step, the model is fit to the data points $x_1, \ldots, x_k$ and the predicted value $\hat{x}_{k+1}$ is derived. In that way, prediction is successively done at only one time point using all previous information. For more details about the estimation of the parameters and the error terms, cf. [SS11].

To support the ARIMA model selection with the prediction capabilities of the model, we combine both in our interactive model selection environment. The graphical user interface (cf. Figure 3.1; for details see [BAF+13]) consists of five main areas: (2a) the time series display showing the input time series and, if applied, the predicted values, (2b) the model selection and prediction toolbox, (2c) the autocorrelation and partial autocorrelation (ACF/PACF) plots (the model selection is steered by interactively moving the vertical lines), (2d) the residual plots to perform the model diagnostics and decide for an adequate model, and (2e) the information criteria for the same purpose as (2d) and for investigating the model selection history.

In this paper, we focus on the relevant elements for the prediction and how the prediction is

integrated in our approach. After the exploration of the input data, the user iteratively increases the model order and applies transformations. The model is applied to the time series and the resulting visual representations to judge the adequateness of the model are consulted. To couple the prediction with the model selection process, we provide prediction controls. The user can apply a model candidate to show the prediction of future values and one-step-ahead prediction that predict values within the given time series. The time series display is used to show the predicted values. This can be done any time throughout the model selection process. Therefore, the user selects either the prediction of future values or the one-step-ahead prediction, which triggers the computation of the predicted values using the currently selected model. For both types of prediction, the predicted values are represented as points connected with a differently colored line. In addition a dashed line shows the upper and the lower prediction error boundary, cf. Figure 3.1.2a.

To emphasize on the accuracy of the prediction, there are several ways to visually support this. We showed different ways of highlighting the difference between actual and predicted values in the one-step-ahead prediction [BAF+14]. There are two limitations we want to address here. First, it is not possible to judge details on what the difference actually means, mainly in context of the error boundaries of the prediction. Second, it is limited in the number of model predictions that can be compared. In Section 3.4, we describe the work by [HJS+09, HJM+11], where the authors use a diverging color scale to encode the difference between the predicted and the actual values with respect to the standard deviation. Using such an accuracy color band, enables to save vertical space and use this to stack visual representations for multiple models. Usually domain experts are interested in how much predicted and actual values differ, for example if the distance is small, medium, or large, considering the standard error of the prediction. Therefore, we suggest to use a categorical diverging color scale instead of a continuous one, like in CareCruiser [GAK+11], where the authors use a diverging color scale to highlight the progress of parameter values from the initial value toward the intended value of applied treatments. They use the full height to encode the difference with color. Like in [JSMK14], either the background of a plot can be used to encode the deviation of actual and predicted value, or just a small color band below and/or above the line to avoid visual clutter.

If multiple model predictions need to be shown, the proposed approach above would limit the vertical space and skew the line of the line plots. As another variant for analyzing the difference we propose to display the difference using Qualizon Graph [FHR+14], as we show in Figure 3.1 (1a-c) for three different models. Qualizon Graphs are extensions of Horizon Graphs [Rei08] and two-tone pseudo coloring [SMY+05] with qualitative abstractions. In our case, we use these qualitative abstractions for the differences between the predicted and the actual values. By using a diverging color scale, this can unveil more radical changes.

The benefit of this visual representation is the vertical space-efficiency, which enables the comparison of predictions of more than one time-series model. In Figure 3.1.1a-c we show how the stacking of the predictions of multiple models can be integrated in the prototype for model selection. Each line (1a, 1b, 1c) represents a Qualizon Graph showing the difference between the one-step-ahead prediction and the actual value of one time series model (m1, m2, m3) respectively. The user can select the model candidates using the model selection history in Figure 3.1.2e. In

addition to the residual plots, this graph gives an impression on how well the models are able to predict the data.

The qualitative abstraction unveils the deviation of the prediction to the actual values in relation to the standard error of the prediction. The difference $d_i = \hat{x}_i - x_i$ between the predicted values $\hat{x}_i$ and the observed values x_i for $k \leq i \leq n$ is calculated and shown in the Qualizon Graph. We use a diverging color scale with six colors, as shown in Figure 3.1.1a-c. In the negative direction, meaning that the predicted value is below the observed value $\hat{x}_i < x_i$, we use three violet color classes, in the positive direction, meaning that the predicted value is above the observed value $\hat{x}_i > x_i$, we use three green color classes. The directions are indicated in the labels as + and -. For our example we use two boundary levels. This results in three classes of difference for each direction (+/-). For the qualitative abstraction we use the distance between the one-step-ahead prediction and the actual value. Based on the chosen boundaries, in our example $x_i \pm 0.84$*standard error and $x_i \pm 1.96$*standard error, the distance is used to assign the color class for the difference based on the direction (+/-) and the distance to the actual value. The light color is used for close predictions, where the difference is within the first boundary. If the difference is larger, meaning outside the first boundary, but inside the second boundary, we use medium color, and dark color for differences where the distance is outside the second boundary. In the following section, we discuss the details of each line in Figure 3.1.1a-c in a usage scenario.

3.3 Usage Scenario

To illustrate how the prediction of different models are compared and how the prediction is integrated in the model selection process, we use a usage scenario from the environmental domain. The dataset is about the water quality in the San Francisco bay area [JC14]. We use the measurement from one station in depths above 5 meters. We aggregate the data to monthly averages and interpolate missing month with linear interpolation. We use a calculated measurement based on the water temperature and salinity of the water from the years 1986 to 2004. In our scenario an analyst from an environmental department in the city council needs to model the time series to predict the expected water quality based on this measurement.

As a first step the analyst loads the dataset and explores the raw data in the time series display (Figure 3.1.2a) and the behaviour of the ACF/PACF in the corresponding plot (Figure 3.1.2c), which suggests an AR(p) component with order $p = 1$. Adding this component improves the residuals (Figure 3.1.2d) and shows the remaining seasonal dependency clearly. Adjusting the model order to $P = 1$ for the seasonal AR(P) component improves the residuals further (m1). The residual plots (Figure 3.1.2d) indicate a remaining seasonal structure, but the analyst first applies the model to predict two seasonal cycles in the future (Figure 3.1.2a), which shows already a good pattern and behaviour. Because of the remaining seasonal structure, the analyst increases the model order of the seasonal AR(P) to $P = 2$ (m2). The analyst recognizes in the residuals, that there may be an improvement of the model by adding an MA(q) component of model order $q = 1$ (m3). The residuals show a model that fits the time series adequately.

The analyst applies again the model to predict two seasonal cycles in the future, and recognises that the error boundary got slightly narrower for this model. Furthermore, in the information

criteria (Figure 3.1.2e) there are three model candidates quite close together. The analyst selects these three models and switches the time series view (Figure 3.1.2a) to the Qualizon Graph view (Figure 3.1.1a-c). In this view the analyst recognizes that there is still a seasonal reoccurring pattern in the prediction, but interestingly the model (m1) in (1a) has smaller differences in the one-step-ahead prediction as the others. The models (m2) and (m3), which where superior according to the residual plots and the information criteria, do not perform so well in the one-step-ahead prediction. This is visible by the increase in dark colored areas from (m1) to (m2) and finally (m3).

Although the error boundary for the prediction of future values of model (m3) narrowed compared to the one of model (m1), it is good enough for the analyst's purpose and according to the principle of parsimony [BJR08] the analyst decides for the least complex model (m1) with only two parameters. Furthermore, the analyst recognizes that all three models underestimate the first and second quarter, but overestimate the third and fourth quarter of each year. This under- and overestimation is minor in model (m1) compared to the others, and therefore a better choice for the analyst.

3.4 Related Work

TimeSearcher [BPS+07] is a visualization tool to search and explore time series data. With dynamic queries it finds patterns and displays multiple forecasts, provided by similarity-based prediction. TimeSearcher uses a data-driven approach that needs exceptional events to be excluded and requires large datasets compared to model driven methods, like ARIMA [BPS+07].

Hao et al. [HJS+09, HJM+11] use a heat-band with a diverging color scale to indicate the accuracy of the prediction compared to the actual values using the normalized differences according to the standard deviation. They apply a moving average smoothing with peak preserving algorithm for the prediction and do not support the selection of ARIMA models. In [JSMK14], the authors use a similar metaphor to encode anomaly scores along the underlying time series. They use the full height in the background of the line chart for the more compact stripe view, but can also switch to encode the anomaly scores in stripes below and above the time series to avoid visual clutter.

The x12GUI [KMST12] package for R offers an interactive tool for the X-12-ARIMA software for seasonal adjustment. The focus is on the exploration of the time series and the results of the seasonal adjustment as well as the manual editing of outliers [KMST12]. For selecting a time series model and adjusting the parameters for the X-12-ARIMA call, form-based input is used. For the computed models there is a history, which allows for loading previous settings, but not to browse and directly compare them. For single models x12GUI provides also the possibility to predict and visualize future values, but this is not integrated in the model selection process.

3.5 Discussion and Conclusion

Predicting future values is one of the main goals of time series analysis [BJR08, BK11]. Integrating the prediction capability of time series models for analyzing and selecting ARIMA models,

supports users in finding a parsimonious and adequate model. Our approach uses an integrated analysis workflow with visual feedback on the selected models and human involvement in the selection process. This enables users to directly examine and judge the prediction capability of one or more models, and choose an adequate model, even if the residual plots do not show recognizable structures and the information criteria differ only slightly.

The usage scenario illustrates how our approach supports users in comparing different models regarding the one-step-ahead prediction. Our approach assists in choosing a model by considering factors, which have not been taken into account in state-of-the-art tools. The visual comparison of the prediction of multiple time series models using Qualizon Graphs resulted in a less complex model, which satisfies the principle of parsimony.

Our approach relies on user expertise to judge the adequateness of the model candidates and does not automatically propose an adequate model. The approach is also limited to the class of seasonal ARIMA models and univariate time series. It is possible to enable a comparison against other time-series models during model selection. A full integration of these models in our interactive environment might require adaptation of the approach, as they do not follow the same model selection process. Finally, it is important to assess the prediction quality in our approach and to evaluate its applicability via user studies with statisticians, and to validate it on new usage scenarios.

CHAPTER 4

Visually and Statistically Guided Imputation in Univariate Seasonal Time Series

Missing values are a problem in many real world applications, for example failing sensor measurements. For further analysis these missing values need to be imputed. Thus, imputation of such missing values is important in a wide range of applications. We propose a visually and statistically guided imputation approach, that allows applying different imputation techniques to estimate the missing values as well as evaluating and fine tuning the imputation by visual guidance. In our approach we include additional visual information about uncertainty and employ the cyclic structure of time inherent in the data. Including this cyclic structure enables visually judging the adequateness of the estimated values with respect to the uncertainty/error boundaries and according to the patterns of the neighbouring time points in linear and cyclic (e.g., the months of the year) time.

The content of this chapter was published in [BFG+15]. We fixed one small mistake on page 66 and changed “cyclic” to “seasonal”.

Markus Bögl, Peter Filzmoser, Theresia Gschwandtner, Silvia Miksch, Wolfgang Aigner, Alexander Rind, and Tim Lammarsch. Visually and statistically guided imputation of missing values in univariate seasonal time series. In *Proceedings of the IEEE Conference on Visual Analytics Science and Technology, VAST – Posters, Chicago, IL, USA, October 25-30, 2015*, pages 189–190. IEEE, 2015. **Best Poster Award VAST 2015.**

The original version is available at `https://doi.org/10.1109/VAST.2015.7347672`

4.1 Introduction

In various application domains data analysts face the problem of missing data. Missing values constitute a data quality problem that needs to be considered in data wrangling [KHP+11]. For example, when measuring water quality in rivers, values may be missing because of particles plugging the sensor.

Missing values cause difficulties for many statistical methods, since they usually rely on complete data information [All09]. There are a few specialized methods to analyze data with missing values [LR02], but the common way to enable the application of established statistical methods is to impute these missing values. Imputation methods are categorized by the type of method itself and the kind of output they provide [All09, GH07, HK10, LR02]. Some methods only impute a single value and replace the missing value, which neglects the uncertainty that is introduced in the data. Others apply repeated resampling or use multiple imputation techniques to compute the imputation uncertainty [LR02]. In case of repeated resampling it is possible to compute the standard error from the variability of estimates [LR02]. Multiple imputation techniques, e.g., Monte Carlo based simulations allow to compute estimates and confidence intervals [Sch99]. Depending on the method, the appropriate error boundary or confidence interval can be used to communicate the uncertainty of the imputation.

We propose an approach that makes the uncertainty inherent in imputed values visible and allows for comparing them to neighbouring values in linear and cyclic time.

4.2 Related Work

In this section we give brief discussion about the background and the relevant related work. We present the main concept important regarding missing values, the structure of time-oriented data, and the relevant visual representations used in our approach.

Missing Values are elements in a dataset, where a observation is missing, which means that it is not available. As discussed in the introduction 4.1, this can be caused by various reasons, e.g. by a failing sensor measurement. According to the statistics literature about missing data [LR02, HK07, GH07, All09] there are different types for the source of missing data: missing completely at random (MCAR), missing at random (MAR), and not missing at random (NMAR). For the definitions and more details see the relevant literature.

Imputation Methods are on a higher level grouped into 4 categories [LR02, HK07]: complete case/record methods, weighting methods, imputation-based methods, and model-based methods. Horton and Kleinman [HK07] provide a overview about imputation methods including a discussion about implementations of these methods. We focus on implementations and packages available in the R project for statistical computation [R C20]. Van Burren provides a list of implementations of multiple imputation methods in the appendix of his book [vB12] and updates in [vB14].

Time-Series Data is data obtained from observations collected over time. Using time granularities and the design concept of cyclic time arrangement in our approach, we consider time-series data related as part of time-oriented data [AMST11]. The literature in statistics [Cle93, Cle94, BD10, HA18] considers time-series as composed by 3 (or 4) components. This are trend, season, and remaining variation (and cyclic component). If included, the cyclic part is defined to be a longer duration than the seasonal part, but shorter than a trend (usually more than 1.5 years if the seasonal part is yearly).

Visual Representations used in our approach are straight forward and well-established. To visually indicate missing values in both views 4.3, time-series line plot with linear time axis [AMST11] and cycle plot [Cle93], we use color and confidence intervals as used by [HKB11] and [TAKP13, STA+13]. To visually explore the missing values or rather the imputations of such, we use confidence intervals and box-plots [Tuk77] depending on the imputation method. This enables to analyse the level of uncertainty in the estimated values. To embrace Keim's Visual Analytics mantra [KMS+08] in our approach we apply various interaction techniques [HS12] like focus+context, direct manipulation, coordinated and multiple view, as well as linking and brushing. To facilitate this techniques, it is necessary to use some variations of the box-plots [Pot06, PKRJ10], like a simplified version [Tuf83], an abbreviated version [PKRJ10], or an extended, more informative version like the violin plots [HN98]. Another important visual representation we use is the cycle plot [CT82, Cle93]. This makes it possible (1) to discern the trend and the seasonal component in one single plot [AMST11] and (2) compare each data value to the values that are related through their proximity in the seasonal cycle. A cycle plot is shown in (b) of Figure 4.1.

To the best of our knowledge, there is no approach for visually and statistically guiding analysts in the imputation of missing values in time-series data. As farther related work, we consider some graphical user interface (GUI) solutions that support the analysis and exploration of missing and imputed values for more general data, see [STA+13, CCH13]. Amelia [HKB11] is a exeptional package that provides multiple imputation methods also for time-series data, but only a menu-driven and form-based GUI. None of them compares the outcome of different imputation methods, nor uses the seasonal cycle for additionally judging the adequateness of the estimated values. This motivates us to propose the TIMI approach.

4.3 Time-Series Imputation Approach

The task we support with our approach, is to impute missing values with a suitable imputation method and provide visual and statistical guidance for judging the adequateness of the imputed values.

The general idea is to benefit from displaying the imputed values from two different perspectives, namely linear and cyclic time. Figure 4.1 shows the design of our approach. For the two perspectives, we use coordinated views, (a) the time series line plot, and (b) the *cycle plot* [Cle93]. The cycle plot is a technique to visualize the sesonal cycle within time series data [AMST11, Cle93]. The seasonal component, e.g. the month in monthly data, is grouped and

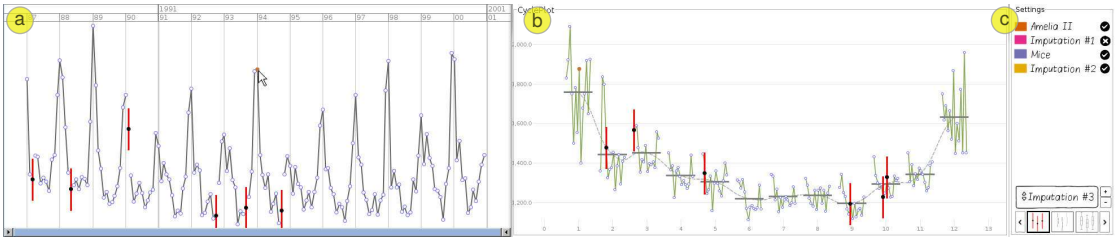


Figure 4.1: Overview of our approach for visually and statistically guided imputation. Coordinated views with (a) a time series line plot using a linear time axis, (b) the corresponding cycle plot (for details cf. supplementary), and (c) a configuration panel. The estimated values (black dots) of missing values and boundaries (red bars) are displayed. Upon request, more details are shown in (a) and (b), either by explicitly selecting the level of detail in (c), or by interaction as described in Figure 4.2. The latter allows the user to adjust the estimated value by dragging the dot up/down. When clicking a point in one window, (a) or (b), the corresponding point in the other window gets highlighted as well.

visualized like in our example data in Figure 4.1 (b), showing first values for Jan., then Feb. etc. More details are explained in the supplementary. The control panel (c) shows a list of imputation methods with an assigned color to indicate the corresponding error boundary or confidence interval in the detail view (Figure 4.2). In this panel it is possible to activate/deactivate, as well as add/remove different imputation methods. Initially, we use a preselected set of imputation methods implemented in the statistical environment R [R C20] and R packages [HK07, vB12]. The missing values are estimated using these initial methods and are shown as black dots together with the error boundaries or confidence interval, represented by red vertical bars. Combining the estimated values from the different imputation methods allows to quantify and communicate the uncertainty of the imputation methods, for instance using error boundaries, confidence intervals or box-plots [LR02].

Seasonal time series are very common in real world applications and their behaviour is considered as a cyclic time structure [AMST11]. Arranging the data points, especially the missing ones, in the representation as described above and linking them using coordinated views, allows to compare them to their neighbouring values in linear time, but also besides the time points close to each other in a seasonal cycle. This enables the user to judge the adequateness of the imputed values. To link corresponding points in these views, we apply bi-directional linking and brushing. When hovering/selecting a point in one view, the corresponding point gets highlighted in the other view. Hovering/selecting the horizontal bar in the cycle plot representing the month's mean highlights all points of this part-of-the-season in the linear time series view.

Details about a specific imputed value can be expanded in both views, by either setting the level of detail in the configuration panel (Figure 4.1c), by hovering the area around the missing value with the mouse cursor, or by zooming within the temporal axis. This shows the results of different imputation methods next to each other (cf. Figure 4.2). These details are represented by error boundaries, confidence intervals, or modified box-plot versions, depending on the outcome of the imputation method (e.g., time series models or multiple imputation). Colors are assigned to the

imputation methods (Figure 4.1c), which allows for comparing estimates of different imputation methods and further fine tune and adjust the imputed values if necessary. For adjusting the imputed value, the dot can be dragged and moved directly, which also changes the value in the other view. By highlighting and simultaneously moving the selected value, it is possible to consider neighbouring values in both the linear and cyclic time.

By providing these details about the uncertainty in different imputation methods, the user can consider these uncertainties when deciding which value is most plausible. In addition, the user is aware of the uncertainty involved and can judge the adequateness of the imputed values more accurately. The user can adjust values through drag-and-drop within the suggested spread of the imputation methods. It allows comparing how the imputation methods impute values differently, e.g. if one method has a wider error boundary or one method over- or under-estimates the missing values.

To preserve the context also in the detailed view (Figure 4.2, step (3)) we use a semantic zoom using a bifocal display. This provides an overview on the imputed value on a higher level and details on demand. All these above described interactions are supported in both views. Moving the mouse to a missing value in the cycle plot or in the line plot shows the details and aids in adjusting the imputed value accordingly.

4.4 Discussion and Conclusion

We proposed a visually and statistically guided approach for the imputation of missing values in univariate time series with seasonal cycles. We discussed how our approach enables the user to gain confidence in how adequate the imputed values are. By combining statistical imputation

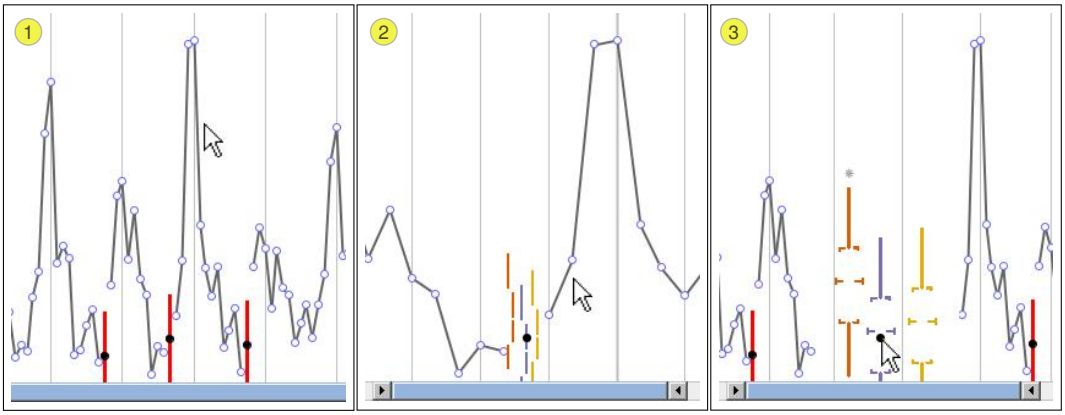


Figure 4.2: Sequence of interactions for more details on demand. This interactions with missing values and their imputed values, are possible in both views, (a) and (b) in Figure 4.1. To provide more details, the representation varies according to zoom level and mouse interaction. The transition in zoom level is shown between image (1) and (2), as well as (1) and (3), depending on the level of detail requested. The color encodes the imputation method, cf. Figure 4.1c.

methods with an interactive visual interface, we provide a view for displaying the time series with a linear time axis coordinated with a view in a cyclic arrangement, side by side. Using linking and brushing helps keeping track of these two different arrangements. The outcome of the imputation methods is visually embedded directly into both views and provides detailed information about the uncertainty and variation of the imputed values in box-plot representations. This enables a better judgement of the adequacy of the imputed values, raise the confidence about these values, and adjust unsuitable values.

There are several possibilities to extend our approach. For multivariate time series a possible correlation between the variables can be used to improve the imputed values. For this extension one needs to think about more appropriate techniques to visually representing the cyclic structure. One limitation is that imputations based on outliers will not provide a good estimate for a missing value. Indicating the time points involved in the imputation may help identifying suspicious values, which may then be excluded in order to improve the imputation. Furthermore, the approach can be used to impute a suspicious value and compare the outcome of the imputation method to judge whether the value really is an outlier. Another limitation is that our approach is not applicable in case the time series has a very strong trend. One idea is to extend our approach and make use of decomposed time series with several views for each component, for instance, separate views for trend and seasonal components.

As laid out in the introduction, missing values are a big issue in time series data from real world applications. Our approach expands on the possibilities of imputation methods by incorporating domain knowledge and an optimized visual representation for seasonal time series.

CHAPTER 5

The Multivariate Cycle Plot

The cycle plot is an established and effective visualization technique for identifying and comprehending patterns in periodic time series, like trends and seasonal cycles. It also allows to visually identify and contextualize extreme values and outliers from a different perspective. Unfortunately, it is limited to univariate data. For multivariate time series, patterns that exist across several dimensions are much harder or impossible to explore. We propose a modified cycle plot using a distance-based abstraction (Mahalanobis distance) to reduce multiple dimensions to one overview dimension and retain a representation similar to the original. Utilizing this distance-based cycle plot in an interactive exploration environment, we enhance the Visual Analytics capacity of cycle plots for multivariate outlier detection. To enable interactive exploration and interpretation of outliers, we employ coordinated multiple views that juxtapose a distance-based cycle plot with Cleveland's original cycle plots of the underlying dimensions. With our approach it is possible to judge the outlyingness regarding the seasonal cycle in multivariate periodic time series.

The content of this chapter was published in [BFG+17]. Original full bibliographic reference:

Markus Bögl, Peter Filzmoser, Theresia Gschwandtner, Tim Lammarsch, Roger A. Leite, Silvia Miksch, and Alexander Rind. Cycle plot revisited: Multivariate outlier detection using a distance-based abstraction. *Computer Graphics Forum*, 36(3):227–238, 2017.

The original version is available at `https://doi.org/10.1111/cgf.13182`

5.1 Introduction

In this paper we propose an interactive environment utilizing cycle plots to explore patterns and to detect multivariate as well as univariate outliers. For the construction of our distance-based cycle plot we use an abstraction based on a multivariate distance measure (Mahalanobis distance), to visualize patterns in multivariate seasonal time series, like trends and seasonal cycle. We build upon the established and effective *cycle plot* by Cleveland [Cle93], which is limited to univariate time series.

Time series often follow a periodically reoccurring pattern, called periodic or seasonal pattern. An example are monthly averages of temperatures over multiple years, with a yearly low, a yearly high, and smooth transitions in between. Such seasonal time series appear in various domains, like ecology, economics, or health. Examples of seasonal time series may be univariate, like number of influenza cases, but many real world examples are multivariate, like number of deaths caused by cardiovascular disease connected with air pollution data, or water quality measures [BD10].

An important objective in time series analysis is the detection of outliers, which in multivariate seasonal time series requires to consider seasonal pattern and trends, both for the several underlying variables and for the multivariate space. The cycle plot described by Cleveland [Cle93] is an effective visualization technique, which facilitates the identification of these seasonal pattern and trends in univariate data, and it allows for comparing data points within the same seasonal cycle (e.g., month of year) in close proximity. These subgroups within the seasonal cycle enable the detection of outliers and extreme values within the groups or whole groups that do not follow the behavior of the seasonal pattern. To achieve the same effect for multivariate seasonal time series, each variable can be represented by one original cycle plot. Although this allows the human analyst to analyze seasonal patterns, trends, and extreme/outlying values of each dimension, building an overview mentally by observing multiple cycle plots is a difficult, time-consuming, and unreliable task. Single data points may behave abnormal in just some or even none of these dimensions, but stand out in multivariate space. Furthermore considering multiple such cycle plots for each variable separately takes additional time and increases cognitive load to combine and transfer the individual dimensions in a multivariate mental model. For detecting anomalies like outliers in the multivariate space, the aid of further abstraction, introduced in Section 5.5, allows to ease this reasoning, as illustrated in Section 5.6 and discussed in Section 5.7. Empirical evidence for an increased performance of our approach is beyond the scope of this paper, but we consider comprehensive user studies of the task performance in future work. The main contributions of this paper are:

- The construction of a Mahalanobis-distance-based cycle plot that
  - involves an additional abstraction step based on generalized multivariate distances and
  - uses a modified visual encoding for these distances, but retains the idea of the original cycle plot.
- An interactive exploration environment of coordinated multiple views combining the distance-based cycle plot with original cycle plots to

- identify outliers in multivariate time series considering the seasonality,
- support the interpretation of multivariate outliers,
- reduce the information loss inevitably accompanying the multivariate data abstraction.

5.2 Related Work

A variety of approaches have been proposed to visualize time-oriented multivariate data by Aigner et al. [AMST11]. Suitable approaches can be categorized into techniques, which provide (1) visualizations for multivariate data, mainly using projections and other aggregation methods, (2) visualizations, which take the structure of time into account, and (3) statistical methods for outlier detection.

Visualizations for Multivariate Data. A frequently used approach is using several line plots [Pla86] either in one coordinate system or as small multiples [Tuf83]. An alternative, already introduced by Playfair in 1786, is the stacked graph [Pla86, BW08]. Wu et al. [WWS+16] incorporate additional information in the stacked graph and discuss clustering and visual arrangement for detecting multivariate patterns. For using small multiples, space-efficient visualizations are well-suited, like Horizon Graphs [Rei08, Few08] and Qualizon Graph [FHR+14]. Javed et al. [JME10] compare the traditional line plot, small multiples, Horizon Graphs, and a new visualization called braided graphs. Another space-efficient visualization for multivariate data are CloudLines [KBK11], which are inspired by ideas of EventRiver [LYK+12]. Tominski et al. [TAS04] compare axes-based visualizations with radial layouts (for example, the time wheel) and discovered that all approaches are suitable for showing multiple variables at the same time and temporal trend detection, but are less appropriate for seasonal cycles. Another technique going back to Playfair et al. [Pla86] for the special case of bivariate data are connected scatter plots, for example Haroz et al. [HKF16]. A very similar concept called trajectories is used in small multiples by Schreck et al. [SBVLK09]. Visually similar are time curves [BSH+16] that project time series in a 2D space based on similarity measures. It is a strong visualization method for finding both regular and irregular temporal patterns. However, the projection makes it hard to compare the length of intervals, and there is no visual representation of the data underlying the distance measure.

The Structure of Time in Visualization. While the structure of time has many different aspects [AMST11], the aspects of granularities and cycles are the most important ones in the context of our work, as the cycle plot [Cle93, Cle94] supports them (see Section 5.3 for a detailed explanation). For pixel-based visualizations, the original work by Keim et al. [KKA95] (which also includes multivariate data), as well as related work by Van Wijk and van Selow [vWvS99], has been the basis for several further publications [SFdOL04, LAB+09, KJL14]. Borgo et al. [BPC+10] evaluate the performance of pixel-based visualizations according to the task complexity and cognitive load. Even though they only tested univariate data, we assume that including multiple variables is a task aspect that creates exactly a task complexity that worsens performance of pixel-based visualizations. Besides the visual representation of periodicity as in the approaches above, it is possible to isolate the seasonal component of time series and to perform further

analysis such as detection of abnormal events [CTB+12] on residuals. Such seasonal time series models can as well be used for prediction [BAF+15, MHR+11].

Statistical Oriented Approaches. Outlier detection has been considered a foremost challenge in statistics for a long time and visual methods a possible solution. There are varying definitions of the term outlier in literature [Agg13, BL98, BG05]. They can be summarized by "an outlier is a data point which is significantly different from the remaining data" [Agg13]. A broad spectrum of methods for outlier detection in time series is available. Primarily, we refer to surveys, taxonomies, or other works which cover the breadth of the topic. Hodge and Austin [HA04] provide a good starting point for an overview on different types of methods, namely statistical models, neural networks, machine learning, and hybrid systems for outlier detection. The recent work by Aggarwal [Agg13], gives an in depth overview on outlier detection in general, with a particular part on outlier detection in time series. Ben-Gal [BG05] gives an overview and a taxonomy on statistical methods for outlier detection. The Mahalanobis distance is a distance-based outlier detection method in the class of parametric outlier detection methods [BG05]. It is an established method and commonly used in statistics to handle multivariate outliers [BG05, HA04, PnP01]. To avoid the influence of outliers on the estimation of the required variables, robust procedures have to be used to identify multivariate outliers [FGR05]. For analyzing and visualizing more than 3 dimensions with basic visualization methods, dimensionality reduction methods can be used, e.g., principal component analysis [Agg13] or multidimensional scaling [BBH11]. However, applying dimensionality reduction, the context of time, especially the periodicity, is lost, and the meaning of the principal components is difficult to interpret intuitively.

In summary, we could identify visualization approaches, which take into account the structure of time and different approaches for multivariate data. Moreover, we found several statistical approaches for multivariate outlier detection, specifically the well-established Mahalanobis distance. We could not find methods that can deal with the structure of time in relation within multivariate time series, neither visualize them in an intuitive and compact way, nor include both at the same time in a statistical oriented approach.

5.3 Background

Before we explain how to compute and construct the distance-based cycle plot, we briefly introduce the original cycle plot by Cleveland [Cle93] and define some variables.

5.3.1 Cycle Plot

The *cycle plot* is a representation described by Cleveland [Cle93] for time series that contains a reoccurring cycle, like a seasonal cycle, and a trend component, which often appear in time series. It was presented as an alternative visualization for this type of data based on the seasonal subseries plot by Cleveland and Terpenning [CT82]. Cycle plots [AMST11, Cle93] are used to investigate the seasonal cycle and the trend along time *granularities*. The concept of granularities is explained in detail by Bettini et al. [BJW00]. Essentially, a granularity is a grouping of discrete points in time to larger units. For example, hours can be grouped into days. 'Day' is a granularity,

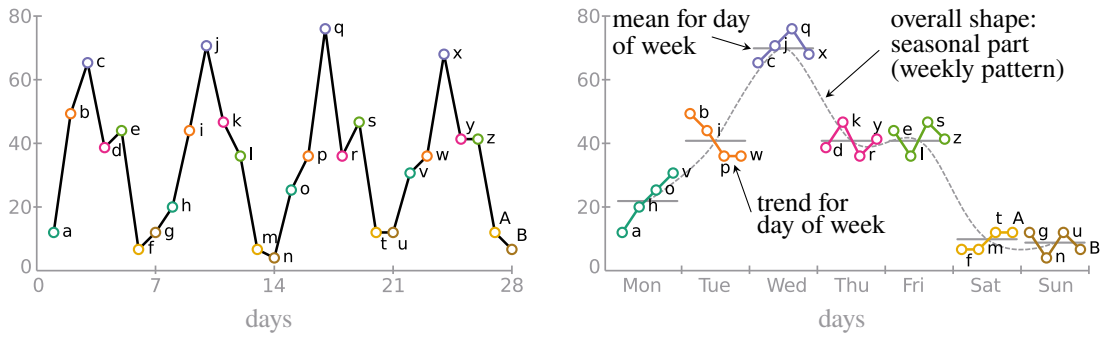


Figure 5.1: Explanation of Cleveland's original cycle plot (adapted from Aigner et al. [AMST11]). Each individual day is labeled with the same letter in the conventional line plot (left) and the cycle plot (right). In the cycle plot the days within one group are connected to form a line. The average value for this day of week is indicated by a horizontal line.

while each specific day is called one *granule* of the granularity 'day'. The cycle plot inverts the order of grouping of two granularities: we illustrate this in Figure 5.1, using the granularities 'day' and 'week'. In the conventional line plot (see Figure 5.1, left) each granule of the granularity 'week' is used to create the tick marks on the horizontal axis. The data points are shown for each granule of each granularity, following the normal order of time. In the cycle plot (see Figure 5.1, right) the horizontal axis is grouped by day of week (Monday, Tuesday, etc.). Hence, the group 'Monday' contains all Mondays of these four weeks. All other days of the week are grouped accordingly. Aigner et al. state that the objective of the cycle plot is: "To make seasonal and trend components visually discernable", and the individual trends are shown "as line plots embedded within a plot that shows the seasonal pattern" [AMST11, p. 176]. In earlier work by Cleveland and Terpenning [CT82], the values of the subseries are plotted using vertical lines on the horizontal line representing the mean, in the later work by Cleveland [Cle93], a line is used for the subseries, like it is commonly known and used today. Yet Cleveland's original cycle plot represents univariate time series data only. In the following we use the term *original cycle plot*, whenever we want to explicitly refer to the original technique [Cle93] as described in this section.

5.3.2 Variable Specification

For the remaining part of the paper we specify variables and sets for the explanations. We will refer to p-dimensional time series data by $X = \{x_1, \ldots, x_n\}$ measured at time point $t_1, \ldots, t_n$. For simplification we use x_k to refer to the p-dimensional measurement at time point t_k, for $k = 1, \ldots, n$. Given a **seasonal length** (s), time points t_{i+j*s}, where $i = 1, \ldots, s$ and $j = 0, \ldots, \lfloor \frac{n-1}{s} \rfloor$ are in position i of the seasonal cycle and adding $j * s$ gives a time point in the same position, but j times further in the coarser granularity, such as the same month in different years. For instance, given monthly measurements x_k over 8 years, $s = 12$ represents the 12 months assembling a year. For $i = 1$ and $j = 0, \ldots, 7$, x_{i+j*s} would represent 8 January values: one measurement for each January of these 8 years. Likewise, the measures for month February, March, and April would be indexed by $i = 2, 3, 4$ respectively. In the following we

refer to these bins as **groups (X_i)** within the seasonal cycle. By writing X_i we indicate the data points within one of the $i = 1, \ldots, s$ groups, where $X_i \subseteq X$, $\bigcap_{i=1}^{s} X_i = \emptyset$ and $\bigcup_{i=1}^{s} X_i = X$. In our example above, the groups represent the months of a year: $\{X_1 = \text{January}, X_2 = \text{February}, \ldots, X_{12} = \text{December}\}$. For each of the groups we can define a group reference point $\mu_1, \ldots, \mu_s$, which can be the mean or median of the data points within the group and will be referred to as **group center (μ_i)**. Moreover, we define a global reference point μ, named **global center (μ)** for mean or median of the whole dataset.

5.4 Task Abstraction and Requirements for Distance Measures

As a basis for discussing the design decisions and reasons for how to apply the additional distance-based abstraction, we first derive the tasks that are supported by the original cycle plot (Tasks T1-T5). Then we derive the tasks for outlier detection, going beyond the tasks supported by the original cycle plot (Tasks T6-T6). Theoretically our abstraction is independent from a specific distance measures, as long as it meets the specified requirements. For our prototypical implementation we apply the Mahalanobis distance, which is an example that meets these criteria.

5.4.1 Tasks

In the following we will use the terms *pattern* of the seasonal cycle and *behavior* within each group. By pattern of the seasonal cycle, we mean the shape formed by the group center perceived in the visual representation. For example in the cycle plot of Figure 5.1, the group of Mondays is generally a lower value followed by an steady increase until the peak on Wednesdays, saddling lower on Thursday and Friday with a drop to the lowest points on Saturday and Sunday. The behavior within a group means basically the pattern of the points or bars representing the data within the group.

The first set of tasks is derived from the tasks supported by the original cycle plot (Cleveland [Cle93], Cleveland and Terpenning [CT82]), as well as from our experience in applying the cycle plot [BFG+15].

T1: Identify the overall pattern of the seasonal cycle. Given a time series with a seasonal component, one wants to get an idea of the overall pattern of the seasonal cycle. This corresponds to an analysis of the finer granularity, e.g., patterns of the months' average over the year.

T2: Identify the behavior within each group. Beside the overall pattern of the seasonal cycle, one needs to assess the behavior of the subseries within each group. For univariate time series, this is often done to identify a larger trend corresponding to the coarser granularity, e.g., patterns within months over several years.

T3: Compare changes within each group to the seasonal cycle and across groups. Besides the individual behaviors of these aspects of the time series, one wants to know which of them drives the patterns of the whole time series and to which extent. It is also interesting how the behavior of one group compares to the behavior of another group.

T4: Detect extreme/outlying values within each group. The way the data is arranged in the cycle plot allows to identify extreme/outlying values with respect to data points within the same seasonal cycle. One wants to detect such extreme values and consider them as possible outliers.

T5: Identify whole groups that deviate from the seasonal cycle. When the overall pattern of the seasonal cycle (T1) is detected, one wants to identify groups within the seasonal cycle that deviate from this behavior.

Additionally, we specify tasks required for outlier detection in multivariate seasonal time series. These tasks are derived from domain knowledge about robust statistics and outlier detection both from literature and from the long-lasting experience of one of our co-authors [FGR05, FRGTA14], who is a statistician.

T6: Detect multivariate and univariate outliers based on the specified boundary. One wants to specify a tolerance boundary and easily detect data points outside this boundary. This needs to be possible for univariate and multivariate outliers.

T7: Detect outliers that are univariate as well as multivariate outliers. Extending task T6, one needs to detect data points that constitute outliers in both, multivariate and univariate context.

T8: Detect multivariate outliers and explore the respective data points in the univariate space. One needs to make selections of data points, in order to explore multivariate outliers and analyze the corresponding values in the univariate plots.

T9: Detect univariate outliers and explore the corresponding data points in the other variables as well as in multivariate space. This task is similar to T8, but one wants to start the exploration with selecting a data point in one (univariate) dimension and see its position in other dimensions as well as its representation in multidimensional space.

T10: Adjust outlier-boundaries and track the resulting outlyingness of data points. The boundaries specify what separates normal from outlying data points. One wants to adjust these boundaries in order to detect borderline outliers.

5.4.2 Requirements for a Distance Measure

To visualize multivariate time series in a cycle plot, we need an additional abstraction step. We decided to use a distance measure, because they are easy to compute, applicable for multivariate data, and a well-known concept. Distance measures, also allows us to retain a visual representation similar to the original cycle plot. Moreover, distance measures are commonly used in outlier detection. To support the tasks described above, the distance measurement for the data abstraction needs to meet the following requirements:

R1: Applicable for multivariate data.

R2: Robust against outliers.

R3: Specific cut-off value exists.

R4: Incorporates the correlation of the data.

5.4.3 Distance Measure

A distance measure quantifies the distance between two points in multivariate space. For our prototypical implementation we decided to use the Mahalanobis distance [Mah36], a generalized multivariate distance, which is an established method in statistics for multivariate outlier detection [BG05, HA04, PnP01] and meets our requirements on a distance measured described above.

In contrast to a basic distance measure, like the Euclidean distance, the Mahalanobis distance considers also the correlation of the data, which meets our requirement R4. A covariance matrix specifies the covariance structure of the data, which involves the correlation and the spread of the p dimensions. In 2-dimensions the spread can be illustrated with ellipses, see the data points and ellipses in Figure 5.3a. Given a p-dimensional dataset with n observations, $X = \{x_1, \ldots, x_n\}$, with the data center μ, and a covariance matrix Σ, the Mahalanobis distance between points x_k, for $k = 1, \ldots, n$, and the center μ is defined as

$$MD(x_k, \mu, \Sigma) = \sqrt{(x_k - \mu)^{\mathrm{T}} \Sigma^{-1} (x_k - \mu)}. \tag{5.1}$$

The center μ and covariance matrix Σ need to be estimated based on the dataset X. To estimate them there are different methods, ranging from classical to robust estimation methods [BG05, FRGTA14]. Even though, the specific method is not relevant for the construction of the distance-based cycle plot, but if used for outlier detection, robust methods are required.

Our main reason for using the Mahalanobis distance is that it is an established distance measure in statistics and used in multivariate outlier detection. By definition, see Equation (5.1), the Mahalanobis distance is applicable for multivariate data, fulfilling our requirement R1.

According to Filzmoser et al. [FGR05], if estimated with robust procedures, the Mahalanobis distance can be used to identify multivariate outliers, using quantiles of the chi-squared distribution. In more detail, in case of multivariate normal distribution, the squared Mahalanobis distance of the data points to the center, with respect to the covariance matrix of the data, are approximately chisquare-distributed with p degrees of freedom, χ^2_p. Thus, a potential multivariate outlier has a higher squared Mahalanobis distance than a certain quantile, e.g., the quantile 0.975, of the χ^2_p. We can use this quantile as a boundary for deciding whether a data point is an outlier or not, which meets the requirement R3.

The center μ as well as the covariance matrix Σ required to calculate the Mahalanobis distances need to be estimated based on the data, and when using classical methods for the estimation, these methods are influenced by outliers. Thus, to avoid the influence of outliers, we use robust methods for the estimation of μ and Σ, which meet requirement R2. In the statistics literature, compare [BG05, FGR05, FRGTA14], the most commonly used methods are the *median* as a

robust estimator for the center of the data and the *minimum covariance determinant (MCD)* estimator [Rou85] for estimating the covariance matrix.

5.5 Features of the Interactive Exploration Environment

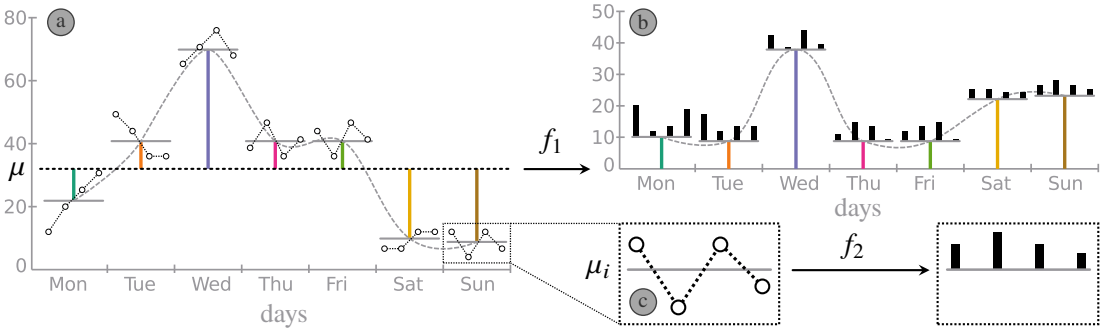


Figure 5.2: Transformation of an original cycle plot to a distance-based cycle plot. Considering the group centers $\mu_1, \ldots, \mu_s$ as points forming a time series line plot and using the distance to the global center μ, we construct the base of the groups, transformation f_1, as described in Section 5.5.1. We do the same transformation f_2 for the pattern within each group. Both, the group centers and the points within each group, can form different patterns that are comparable to a seasonal pattern or trend in the original cycle plot.

The main element in our interactive exploration environment is the distance-based cycle plot, which shows an abstraction of a multivariate time series using distances. This section is aimed to ease the understanding of the construction of the distance-based cycle plot by first explaining the transformation of an original cycle plot to a distance-based cycle plot. We then illustrate its construction in a bivariate case. Finally it is generalized for the multivariate case and integrated into our interactive exploration environment.

5.5.1 Seasonal Cycle (Inter-Group Distance)

Our goal is to support the same tasks as the original cycle plot, but for multivariate seasonal time series, cf. Section 5.4.1. The original cycle plot shows the pattern of the reoccurring cycle, like the season over the year, which is required for tasks T1, T3, and T5 in the distance-based cycle plot. The identification of this overall pattern is supported by showing vertical lines that indicate the group centers $\mu_1, \ldots, \mu_s$, see Figure 5.1.

In the original cycle plot, each group center is a real number, where the absolute value can be considered as distance between this group center and the zero line. Instead of the zero line as central reference, we use by default the global center of all groups μ for constructing distance-based cycle plot. During the exploration process this global reference point can be changed interactively (see Section 5.6). In particular, we compute the Mahalanobis distance *MD* between each group center μ_i, $i = 1, \ldots, s$, and the global center μ. The result is one horizontal line for each group, serving as group base line. In Figure 5.2 we use the original cycle plot to

illustrate how the distance-based cycle plot is constructed. In the original cycle plot (Figure 5.2a), the global center μ is indicated by a horizontal dashed line. We compute the distance of each group center $\mu_1, \ldots, \mu_s$ to this global center (indicated by the colored vertical lines). For the construction of this generalized distance based cycle plot, we apply these distances (group μ_i to global center μ) on the y-axis. Thus, the global center is represented by the x-axis itself (see transformation f_1 in Figure 5.2). This is due to the fact that these distance measures are always non-negative. The distinction between 'above' and 'below' the global center may be applicable in the univariate example given in Figure 5.2, but does not make sense in an actual multivariate scenario (as explained in Figure 5.3). We discuss this information loss due to the data abstraction Section 5.7 and describe how our interactive exploration environment allows to reduce this information loss.

In Figure 5.3 we show the construction in the bivariate case. We consider a small example of daily measurements of two variables: temperature and humidity. In this example the seasonal cycle $s = 2$ is grouping the data points into measurements during the *day* and measurements during the *night*. For each group – day and night – we calculate a bivariate average for temperature and humidity values combined, which is equivalent to the group centers μ_{day} and μ_{night}. In addition, we calculate the global center μ of the whole dataset, i.e., the bivariate (temperature and humidity) mean of all data points (8 days and 8 nights combined). In this bivariate example these two group centers and the global center are points in two-dimensional space (see Figure 5.3a). Given the group centers μ_{day} and μ_{night}, the global center μ, and the covariance matrix Σ (see Section 5.5.4), we calculate the Mahalanobis distance $MD(\mu_{\text{night}}, \mu, \Sigma)$ of the night-group center μ_{night} to the global center μ and $MD(\mu_{\text{day}}\mu, \Sigma)$ of the day-group center μ_{day} to the global center μ. This distance is used as the position of the group-base line on the y-axis shown by transformation g_2 and g_3 in Figure 5.3b.

In case of more than two dimensions, the group centers μ_i, the global center μ, and distances between them are calculated accordingly. By using these distance values, we are able to represent p-dimensional datasets in our distance-based cycle plot. For example, consider the monthly temperature (see Figure 5.4b'): there are low values in winter, high values in summer, and average values in spring and fall. This seasonal pattern is reflected in the position of the group center lines in a similar pattern like in the distance-based cycle plot (see Figure 5.4a). The winter months, like January as coldest month, and the summer months, like August as hottest month, have large distances to the global center and, therefore, appear as peaks in the distance-based cycle plot, whereas the average spring/fall months have small distances from the global center and therefore are closer to the x-axis.

5.5.2 Data Within Groups (Intra-Group Distance)

For representing the data points $x_1, \ldots, x_n$ within the groups $X_{i,\ldots,s}$, we apply a similar approach, like in the original cycle plot. In the original cycle plot, the data points are represented by points on a line following the order of the coarser granularity, and the position on the y-axis given by their values. One feature of this representation in the original cycle plot, is that it is possible to see the trend over the coarser granularity, for example over the years. For the distance-based cycle plot we compute the Mahalanobis distance of the data points in X_i to their group center μ_i. In

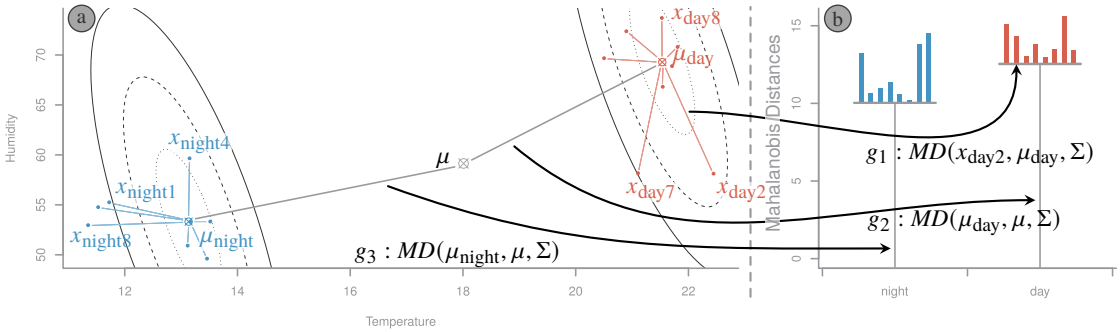


Figure 5.3: Construction of the distance-based cycle plot with a bivariate example. The construction of the bivariate cycle plot on the right is based on Mahalanobis distances of the bivariate data points to the respective group center (e.g., transformation g_1) and on Mahalanobis distances of the group centers to the global center (transformation g_2 and g_3). The usage of distance measures that are applicable to p-dimensional space, is a key aspect applied in our data abstraction. The ellipses illustrate the spread of the data captured in the covariance matrix.

contrast to the original cycle plot we represent distances instead of actual data points, and thus, we chose to use bars instead of connected points. This also picks up the original design by Cleveland and Terpenning [CT82], using vertical lines.

In Figure 5.1 we illustrate how the data points are binned and arranged in the original cycle plot. With univariate data the transformation to the distance-based cycle plot is similar to the construction of the horizontal lines for the group centers. Using the illustration from Figure 5.2, the data points in each group form a time series line plot with a horizontal line representing the group center. Computing the distance of each point to this group center allows to draw them as bar chart within each group, like we illustrate for one group in Figure 5.2c by transformation f_2.

In the bivariate example (Figure 5.3) we compute the Mahalanobis distance of each data point from the subset of measures during day $x_l \in X_{day}$ and night $x_m \in X_{night}$, where $l, m = 1, \ldots, 8$, to the respective group centers μ_{day}, μ_{night}, $MD(x_l, \mu_{day}, \Sigma)$ and $MD(x_m, \mu_{night}, \Sigma)$. These distances are ordered according to the coarser granularity, in our example calendar days, and represented as bars within their group, as shown by transformation g_1 in Figure 5.3b. As discussed before, the distance can be computed for any p-dimensional multivariate data set. However, in contrast to the original cycle plot, the seasonal pattern and trends need to be interpreted differently. In the following section we discuss the interpretation of these patterns in the distance-based cycle plot.

5.5.3 Design and Interactions

In Figure 5.4 we show the design of our interactive exploration environment. Our goal is to support users in exploring seasonal patterns and trends as well as detecting and exploring univariate and multivariate outliers (see tasks T1–T10 in Section 5.4.1). Using distances to construct the distance-based cycle plot (Figure 5.4a) allows for representing an arbitrary number of dimensions. To gain further insights into the multivariate dataset, we provide interactive exploration means

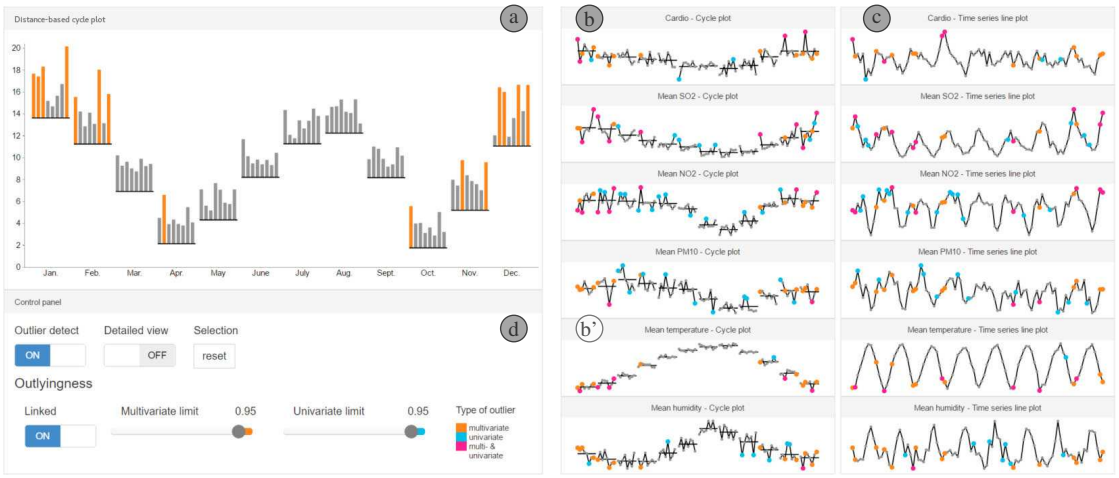


Figure 5.4: Prototype implementation of our interactive exploration environment utilizing the Mahalanobis-distance-based cycle plot. The prototype employs coordinated multiple views with the distance-based cycle plot (a) next to the underlying univariate plots: an original cycle plot (b) followed by the univariate time series line plot (c). In this screenshot we use space-efficient sparkline representations, in order to provide a comprehensive overview. The small plots on the right side (b+c) can be changed to a more detailed view with a scroll bar for detailed exploration. The bottom left shows the control panel for interactive exploration (d). Color encodes the type of outlier and sliders are used to specify outlier-boundary values. (b') is the original cycle plot of variable temperature, used as example in Section 5.5.1.

that allow for switching back and forth between the distance-based visualization and the multiple underlying univariate representations. For visualizing the multiple variables of the multivariate time series, we provide the original cycle plot next to a time series line plot. This allows two perspectives on the same univariate dimension, like it is used by Bögl et al. [BFG+15] for time series containing missing values. In Figure 5.4, we show sparklines [Tuf06] for the univariate visualizations (b & c), to fit more variables on the screen. In the control panel (d), the user can switch between detailed view and sparkline view and adjust parameters. In the detailed view, the sparklines are replaced by more detailed line plots.

To explore patterns and outliers in the multivariate space and the underlying univariate dimensions, we provide multiple linked views with highlighting triggered by hovering and selection, including multiple selection. Highlighting and selection are supported in each of the visualizations. To allow the exploration of more dimensions, the univariate plots are scrollable.

We encode three types of outliers using color. We selected three distinguishable colors according to the L*a*b* color space and maximized the perceptual distance of the selected colors, compare [HSA+10]. The three types of outliers and respective colors are: (1) univariate outliers represented by cyan ■, (2) multivariate outliers represented by orange ■, and (3) outliers in univariate and multivariate represented by magenta ■.

Our interactive exploration environment is independent of a specific method for computing

univariate or multivariate outliers. For doing so, a lot of methods exist in statistics literature, see [BG05, HA04] for an overview. In our environment, we highlight the identified outliers in univariate as well as multivariate space and allow to adjust the parameters for outlier detection. For example, in Figure 5.4d the user can adjust the boundaries used for outlier detection. In the following section, we will give details on the calculations used for our prototype.

5.5.4 Robust Calculations

We use the median to compute the global center $\mu = \text{colMedian}(X)$, and the group centers $\mu_i = \text{colMedian}(X_i)$. For constructing the distance-based cycle plot, there are two possibilities to estimate the covariance matrix Σ. Either to estimate a separate covariance matrix for each of the groups Σ_i, or to center the data points on their group median and estimate a global covariance matrix Σ using the centered data points $\bar{X} = X_i - \mu_i$. Testing with some data sets showed an instability in the estimation of separate covariance matrices Σ_i. This is due to an often low number of data points in each group compared to the number of dimensions. Therefore, we apply the MCD method to compute the global covariance matrix $\Sigma = \text{covMcd}(\bar{X})$ with all centered data points $\bar{X}$. For a dataset with $\lfloor \frac{n-1}{s} \rfloor \gg p$ one can decide for any of these methods as needed.

For univariate outlier detection, we use a similar approach as described above. If the covariance matrix Σ is estimated robustly, the diagonal consists of robust estimates of the variance σ_r^2 for each variable $r = 1, \ldots, p$ in the p-dimensional dataset. Using the centered data points $\bar{X}$, the center of $\bar{X}$, $\mu_r = 0$, and the variance σ_r^2, we compute the outlier based on the selected quantiles of the underlying univariate distribution. In case the absolute value of a centered univariate data point is higher than a certain quantile, it is identified as univariate outlier. If the univariate data is normally distributed, we compute the quantile of $N(\mu, \sigma^2)$. Like for the multivariate boundary, we provide an interactive slider in our exploration environment for selecting the quantile, see Figure 5.4d.

Note that the assumed distributions (chi-square, normal) for the distances will most likely not be met because the observations are time-dependent, and thus not independent from each other. However, the quantiles of these distributions still serve as an indication of outlyingness of the data points. The goal of outlier detection is thus more in an exploratory context, namely to draw the attention of the user to these highlighted points.

5.6 Usage Scenario

We implemented the interactive exploration environment utilizing the distance-based cycle plot in a prototype and apply the prototype in a usage scenario. We use this usage scenario to illustrate how the distance-based cycle plot visualizes real data and how the interactive exploration environment advances the possibilities to explore patterns and outlying values in multivariate seasonal time series data. Throughout the usage scenario, we refer to the related tasks T1–T10 described in Section 5.4.1. We support the reader in following the usage scenario with additional figures provided in the supplementary material and refer to our prototype available online at `http://cycleplot.net`.

Mortality & Air Pollution Dataset. The dataset is about the mortality, air pollution, and meteorological data for major cities in South Korea [LOK13]. It is available in the R project for statistical computing [R C20] as library named *HEAT* [LOK13]. The dataset contains several air quality indicators together with meteorological data as well as the number of deaths caused by cardiovascular diseases and respiratory diseases. The dataset consists of daily measurements for several years (2000-2007). For the illustration, we select a subset of 6 variables, cardio (deaths caused by cardiovascular diseases), SO_2 (Sulfur dioxide), NO_2 (Nitrogen dioxide), PM_{10} (particulate matter), temperature, and humidity from the city Seoul aggregated to monthly averages.

User. As a possible user is a public health official, who analyzes and explores the seasonal pattern, trends, and the outliers in the dataset described above.

Goal/Tasks. The overall goal is to get insights into the seasonal patterns, trends, extreme, and outlying values of the dataset. For details on the particular tasks to achieve this goal, we refer to the tasks T1–T10 described in Section 5.4.1.

As a proof of concept, we separated the preprocessing of the dataset and the prototype of the interactive exploration environment. The preprocessing of the dataset (HEAT library [LOK13]) was done in R [R C20]. The computation of global and group centers, Mahalanobis distances, and outlyingness values was done as described above (see Sections 5.4.3 and 5.5.4). The implementation of the interactive exploration environment was done as web application using *JavaScript*, where we imported the precomputed data file. For future work, one may combine the computations in R with the interactive visualization in a web application, by using appropriate libraries to connect them.

The user first wants to get an overview of the seasonal behavior of the time series in the multivariate space (compare Task T1). According to the group centers of the months, see Figure 5.4a, the user identifies that there are peaks in summer as well as in winter. This means that summer and winter months are on average more extreme than the global center, which basically represents an average month, e.g., spring (April) or autumn (October). Selecting one month center as the global reference point, e.g., January, shows that the other winter months are closer to January than the summer month (see the supplementary material for more details). Considering the transitions between high and low peaks of the season in the original cycle plot representation of each variable, the seasonal pattern of the Mahalanobis-distance-based abstraction follows a similar smooth behavior. The user then considers the behavior within the groups according to their position in the seasonal cycle (Tasks T2 & T3). He/She identifies a tendency in some of the peak months (Dec., Jan., & Feb.), that the data values within the group vary more than in others. Especially, when comparing to the other peak in summer, the user detects this additional variation with larger distances to the group center (Task T4). Next, the user compares the variations within the groups and across groups in more detail (Tasks T2 & T3). When looking at the months June and March, he/she spots distances with roughly the same length, except for the first year. These months seem to be quite stable months across all dimensions. Even without highlighting the user can easily identify extreme values by large bars, that may be possible outliers (Task T4). Amongst others, the user considers the last year in January, first in June, and several in December, as possible outliers. In our example, the user cannot find any group that deviates from the seasonal cycle

(Task T5), but one can imagine one whole month, that stands out of the multivariate seasonal pattern.

The user activates the highlighting of outliers and selects a certain quantile for the univariate and multivariate boundaries to indicate outlyingness of the data points. The user selects the 0.95 quantile, shown in Figure 5.4, and gets an overview of patterns in the outlyingness that allows to detect multivariate as well as univariate outliers easily (T6). For example, the user considers interesting that there are multivariate outliers only in months Oct.–Apr., and an exceptionally large number in Dec.–Feb. Knowing that, the user detects the same pattern in the original cycle plots and recognizes that there are more data points in these winter months highlighted in magenta (T7), indicating outliers in both, uni- and multivariate space. The user immediately recognizes that the months Nov.–Feb. in the last year are all multivariate outliers. To further investigate the outlyingness in the univariate space, he/she selects the outliers (T8), which highlights the corresponding data points in the univariate plots. This exploration reveals that in some variables, e.g., cardio and PM_{10}, they are indicated as multivariate outliers only, yet in others, e.g., SO_2 and NO_2, they are highlighted as outliers in both, univariate and multivariate space. Looking at the original cycle plot for the variable cardio, the user detects two extreme data points in Nov. and Dec., highlighted in magenta. Selecting them shows that in the distance-based cycle plot, they can also be recognized as data points with large distance to the center (T9). The user also recognizes that besides being multivariate outliers, the variable cardio is also a univariate outlier in Nov. and Dec., but the variable temperature is a univariate outlier only in Nov. not in Dec. By changing the outlier boundary with the slider, the user can track the data points that are borderline and are indicated as outliers, when the boundary is decreased. For example, the first bar in month Mar. and Jun. in the distance-based cycle plot, see Figure 5.4a, are only highlighted as outliers, when changing the threshold from the 0.95 to the 0.9 quantile (T10). This allows to interactively get an impression about how extreme the outliers are.

In contrast to using only multiple original cycle plots, the user is able to explore the seasonal pattern and patterns within and across groups directly in the multivariate space. Obviously, it is required to also consult the underlying univariate visualizations, but combined in the interactive exploration environment, the distance-based cycle plot is vital for getting insight in the overall picture of the multivariate seasonal time series.

5.7 Discussion

So far, we introduced our interactive exploration environment for exploring patterns and outliers in multivariate seasonal time series and explained the construction of the utilized distance-based cycle plot. We abstracted the tasks relevant to do so together with the requirements for a distance measure in Section 5.4 and argued the construction of the visualization and the design of our environment, accordingly (Section 5.5). In these sections we briefly discussed the benefits and limitations of specific decisions for the construction and the design. In this section we will continue this discussion of benefits and limitations in more depth, cover the performance of our approach regarding the specified tasks, and give an outlook on future work.

5.7.1 Benefits and Limitations

One main benefit of the way we construct the distance-based cycle plot is the independence from the number of dimensions. This is achieved by constructing a distance based cycle plot using Mahalanobis distances (see Sections 5.4.3 and 5.5). This causes a different representation of patterns (i.e., the seasonal pattern and patterns within groups), and therefore, the distance-based cycle plot needs to be interpreted differently. A distance is a non-negative number by definition. As a consequence, all distances are represented above the group center lines (see Figure 5.2). We, thus, loose the information about the exact position of that data point, for the sake of being able to represent multiple dimensions. While the information about the actual position appears to be important for one-dimensional space (maybe even for two-dimensional, and three-dimensional space), it is very difficult to represent this in multivariate space. One commonly used method in this case is dimensionality reduction, like principal component analysis [Agg13] or multidimensional scaling [BBH11]. However, one limitation of this technique is, that it is difficult to interpret the meaning of the principal components, e.g., first and second for 2-dimensional visualizations. By applying dimensionality reduction, the context of time, especially the periodicity, is lost. Our approach, i.e. using the Mahalanobis distances, is also a type of abstraction from multivariate space to less dimensions. To reduce this information loss our interactive exploration environment, see Section 5.5.3, allows further investigations in both, the distance-based cycle plot and each of the single dimensions in multiple linked views. By taking the structure of time into account, and wisely selecting the granularity levels according to the seasonal time series, this abstraction still retains the temporal context.

While the idea of using intra-group vs. inter-group distances is a well-known strategy, we did not find this applied in cycle plots for multivariate time series (see Section 4.2). The full explorational power of our distance-based cycle plot for multivariate time series needs to be seen in context of the interactive exploration environment, where it is possible to connect the multivariate intra- and inter-group distances to the visualizations of the single dimensions in original cycle plots and line plots and thus, to investigate anomalies, like outliers.

Using distances, however, may lead to a setting where the global center has the same distance to multiple or even all of the group centers. This would result in a representation where all group base lines lie on the exact same position on the y-axis. In Figure 5.3 the global center μ is close to have the same Mahalanobis distance to both groups, and therefore, the base lines in Figure 5.3b are nearly on the same horizontal level. It is then difficult to judge if there is no seasonal pattern at all or if the group centers span a multidimensional sphere around the global center. The same drawback is true for original cycle plots. In case there is no seasonal pattern in univariate seasonal time series data, also the original cycle plot would show group centers aligned on a horizontal line. One way to tackle this problem is to relate the group centers to a different reference point. Instead of a mathematical global center we can define a global reference point. This reference point could be, for instance, the basis, the zero point, of the multivariate coordinate system. Another potential reference point would be choosing one representative group, i.e. a reference month whose group center line would then lie on the x-axis, and relate the other groups to this reference group. The same can be applied for group centers, by defining a group reference point instead.

For both issues mentioned above, we introduced the possibility to select any group center (μ_i) as global reference point and therefore investigate the relation of all other groups to the selected group in our interactive exploration environment. We use the multiple linked views to enable the selection of single or multiple points either in the distance-based cycle plot or in one of the univariate representations. Brushing and linking allows to further explore the relations of a point in the multidimensional space and the connection to the single dimensions. This feature helps to reduce the information loss introduced by the distance based abstraction. The flexibility of interactively adjustable global as well as group reference points, allows a further investigation of relations between data points within and even across groups. The exploration of different aspects of locality (in linear and periodic time, as well as across groups) is possible because of using Mahalanobis distances. There is relatively few work done considering this local aspects in outlier detection, see [FRGTA14].

Another possible limitation is demonstrated by a case in which there are very large distances between the global center and the group centers and only small variation of the data points within the groups. A common scale (y-axis) would thus lead to a distance-based cycle plot that shows mainly long vertical lines representing the distance of group centers to the global center. In consequence the small variations within the groups would not be visible. However, this can also happen in the original cycle plot. To tackle this problem, we propose using data transformations, such as a log scale. Another solution would be to use separate scales. Using one scale for the distance of group centers to the global center, and another scale for the distances within groups would allow for exploring the smaller variations within groups, while preserving the overall picture and comparison between groups.

A problem related to the previous one, would be a very strong trend within the data (e.g., monthly data over several years). A steep trend over the years would distort the patterns within groups of months. This, again, is a problem that affects the original cycle plot as well. In time series decomposition the time series is split up into trend-, seasonal-, and the error- or irregular-component [BD10]. These can then be analyzed separately with appropriate visualizations, e.g., using our distance-based cycle plot for the seasonal-component. Seasonal adjustment is usually applied to remove the seasonal component in order to analyze the other components. There are also recent approaches to seasonal adjustment for multivariate time series (see [GM13]), that can also be used to separate the seasonal component for a more detailed analysis, which again could be supported by using our distance-based cycle plot.

5.7.2 Task Performance

A formal cost-benefit analysis of our approach on the basis of Chen and Golan [CG16] is beyond the scope of this paper, but the following discussion should better explain the benefit of our approach when performing the task outlined in Section 5.4.

Using only an original cycle plot for each variable (only the plots in column (b) of Figure 5.4), one can easily identify overall patterns of the seasonal cycle (Task T1), identify the behavior within each group (Task T2), and compare changes within each group to the seasonal cycle and across groups (Task T3), but only for each of the variables separately. For picturing these patterns and

behaviors within the multivariate space, one can do this to a certain extend by mental aggregation. If there are very similar and smooth transitions and patterns in each of the variable, one can imagine or derive mentally a similar pattern in multivariate space. On the other hand, when using only the distance-based cycle plot (Figure 5.4a) for these tasks, it is possible to identify the patterns and behavior of the abstraction only. One can identify peaks and transitions as well as intra- and inter-group similarities in the multivariate space, but it is not possible to break down any information for individual variables. While the tasks of detecting extreme/outlying values within each group (Task T4) and the identification of whole groups that deviate from the seasonal cycle (Task T5) can be done for each variable separately, multivariate outliers are possibly not outliers in any of the variables or outliers just in single variables. Also by summarizing mentally the individual variables, data points that are outliers in single or many individual variables, may or may not be multivariate outliers (see Section 5.6). For these tasks (Tasks T1-T5) a combination the distance-based cycle plot (Figure 5.4a) and the several original cycle plots (Figure 5.4b) is necessary. The support of interaction is also beneficial to keep track of single points, when switching the focus between the single variables and the distance-based cycle plot.

If we consider the Tasks T6-T10 for multivariate and univariate outlier detection that are going beyond the detection of extreme/outlying values only within groups (Task T4) and identification of whole groups that deviate from the seasonal cycle (Task T5), one also requires the classic line plot representation of each variable (Figure 5.4c) to also identify peaks and anomalies, like possible outliers, and how they relate to each other in linear time, e.g. the larger values within the last year of variable NO_2 and SO_2 in Figure 5.4b and c. This side-by-side presentation of the same data in different representations, cycle plot and traditional line chart, adds an additional perspective that allows to investigate the connections of outliers in each of the variables. Highlighting univariate outliers in original cycle plots and line plots only, does not allow for investigating data points that are no outliers in any of the single variables but are multivariate outliers. The distance-based cycle plot allows to investigate the data points relations with different aspects of locality, e.g. to data points in the same position of the periodicity, but also the distances compared to other groups as well as interactively changing global and local (group) reference points. Only the possibility to select and highlight the data points in these coordinated multiple views of different perspectives, allows to easily switch from the abstracted multivariate data to the single dimensions, which eases the external memorization of the analysis. This affects all the tasks for all combinations of only multivariate or only univariate outliers in one or more dimensions, as well as outliers that are both, multivariate and univariate outliers in single or multiple dimensions.

The main advantage of the distance-based cycle plot is the aggregated overview of all dimensions combined, to show directly the patterns and anomalies like outliers in a condensed view. The full exploration power is only achieved by the interactions, highlighting, and coordinated multiple views in combination with the twofold visualization of each variable in a classical cycle plot and a line plot. Another benefit of this combination of different representations and the additional abstraction of the distance-based cycle plot is, that it enables the investigation of each data point. First, locally according to the normal linear time scale for each variable, second, locally according to the periodicity in the several original cycle plots for each variable, and third, locally in context of the distance-based abstraction from the multivariate data, again according to the periodicity in

the distance-based cycle plot.

5.7.3 Future Work

There are several questions with regard to the usability of our approach that remain open. First, the distance-based cycle plot, although encoding an abstraction using a multivariate distance, uses a very similar design to the original cycle plot. This may cause confusion and needs to be learned by the user. Furthermore, we do not know about the performance regarding the abstracted tasks and whether the combination of different cycle plots discussed above helps users to identify multivariate outliers. For multivariate time series involving periodicity, this remains an open question. Only future evaluation with real users can answer these questions. We plan to tackle this issue first by providing additional use case examples, and by formal user studies focusing on the correct interpretation of the plots and task performance. Due to the number of tasks and their complexity, this will require more than one study. For the time being, we demonstrate the applicability of our approach by the walk-through in the usage scenario (Section 5.6), the comprehensive discussion above, the provided supplementary material, and the possibility to test the prototype in an online demo.

5.8 Conclusion

Our interactive exploration environment utilizes a distance-based cycle plot for identifying seasonal patterns and outliers in multivariate seasonal time series. It revisits and retains a visual representation similar to the original cycle plot by Cleveland [Cle94]. The construction of the distance-based cycle plot includes an additional abstraction step using the Mahalanobis distances, which enables the generalization to an arbitrary number of dimensions. With our interactive exploration environment we combine statistics and visualization techniques and balance their benefits and limitations for visually analyzing patterns and outliers in multivariate seasonal time series, with respect to the structure of time and the relations among multiple dimensions.

5.9 Appendix: Supplementary Material for Usage Scenario

Supplementary material to support the reader in the usage scenario discussed above in Section 5.6.

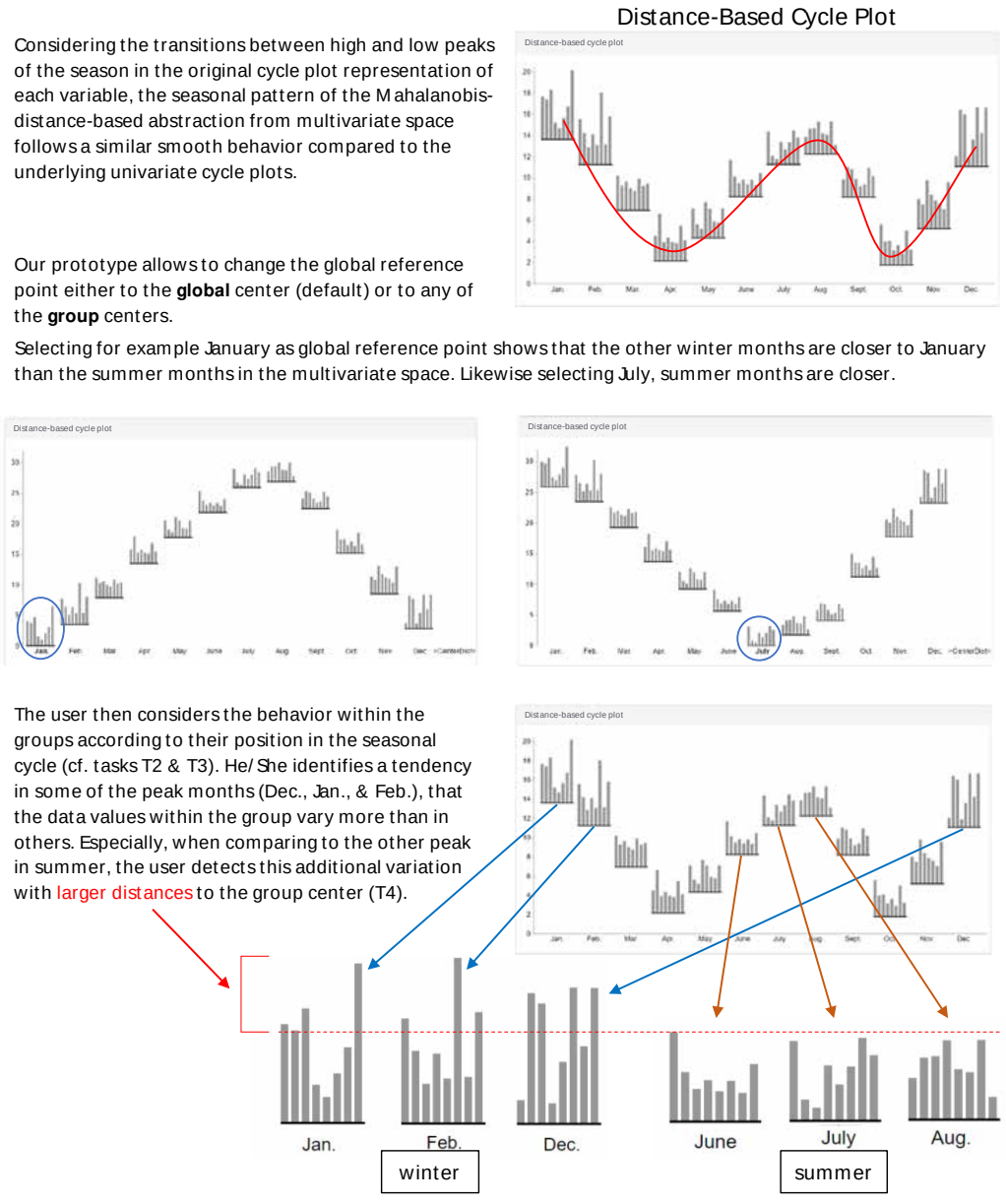
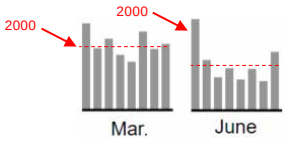
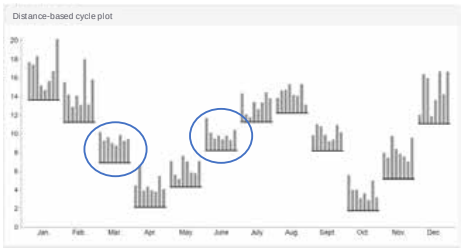


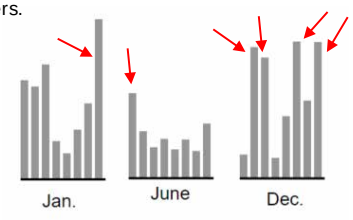
Figure 5.5: Supplementary material for the usage scenario, page 1.

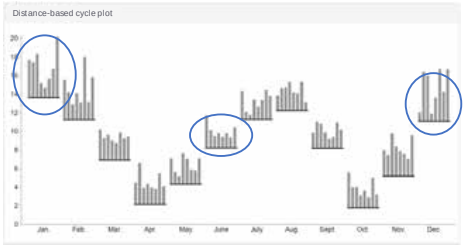
Next, the user compares the variations within the groups and across groups in more detail (T2 & T3). When looking at the months June and March, he/she spots distances with roughly the same length, except for the first year. According to this pattern, these months seem to be quite stable months across all dimensions.



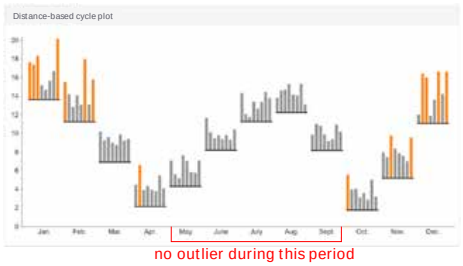


Even without highlighting the user can easily identify extreme values by large bars, that may be possible outliers (T4). Amongst others, the user considers the last year in January, first in June, and several in December, as possible outliers.





For example, the user finds interesting that there are multivariate outliers only in months Oct.–Apr., and an exceptionally large number in Dec.–Feb.



Knowing that, the user detects the same pattern in the original cycle plots and recognizes that there are more data points in these winter months highlighted in magenta (T7), indicating outliers in both, uni- and multivariate space.

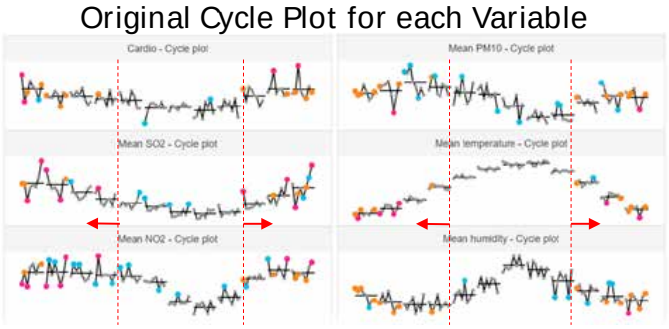
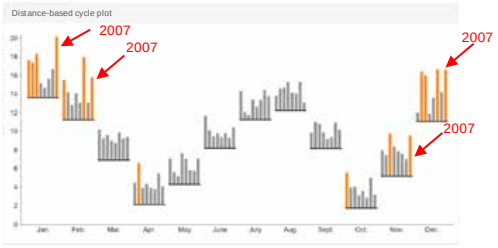


Figure 5.6: Supplementary material for the usage scenario, page 2.

The user immediately recognizes that the last year (2007) in Nov.–Feb. are all multivariate outliers.



Looking at the original cycle plot for the variable **cardio**, the user detects two extreme data points in Nov. and Dec., highlighted in magenta. Selecting them shows that in the distance-based cycle plot, they can also be recognized as data points with large distance to the center (T9). The user also recognizes that besides being multivariate outliers, the variable cardio is also an univariate outlier in Nov. and Dec., but the variable temperature is an univariate outlier only in Nov. not in Dec.

Remark 1: the colored arrows (,) are not part of the prototype.
Remark 2: selected items have their border increased (Nov. 2002 and Dec. 2002).

By changing the outlier boundary with the slider, the user can track the data points that are borderline and are indicated as outliers, when the boundary is decreased. For example, the first bar in month Mar. and Jun. in the distance-based cycle plot are only highlighted as outliers, when changing the threshold from the 0.95 to the 0.9 quantile (T10). This allows to interactively get an impression about how extreme the outliers are.

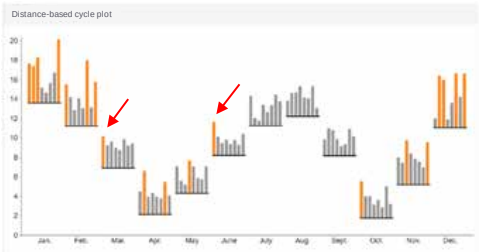


Figure 5.7: Supplementary material for the usage scenario, page 3.

Part III

Recapitulation and Coda

CHAPTER 6

Conclusion

“ [...] [W]e believe, [there] is a clear demand that pictures based on exploration of data should *force* their messages upon us. Pictures that emphasize what we already know—‘security blankets’ to reassure us—are frequently not worth the space they take. Pictures that have to be gone over with a reading glass to see the main point are wasteful of time and inadequate of effect.

The greatest value of a picture is when it *forces* us to notice what we never expected to see.”

John W. Tukey, 1977. [Tuk77, p. vi]

"It's the end of the world as we know it [...]"
— R.E.M. (1987), from the Album *Document*

In spring 2020, the global world as we knew it started falling apart. People were looking at line charts showing numbers of infections by SARS-CoV-2 (COVID-19) increasing steeply into the sky. These charts are omnipresent in the media and are presented by politicians, medical professionals, and epidemiologists. If somebody had forgotten what exponential growth is, they now had it illustrated and taught in these charts together with predictions of what the exponential growth meant for the future. Different measures to intervene were presented together with the expected outcome in predicted number of cases. Everybody was watching the line in the chart progress while the message "flatten the curve" was propagated on all different media channels. Perhaps most famously, the comic by Siouxsie Wiles and Toby Morris was published on Twitter (see Figure 6.1). Using such illustrative graphs was important to sensitize the public to the necessity and effects of the measures taken and to be more careful in order to prevent a collapse of the healthcare system. Suddenly, people had to deal with statistical and epidemiological data and graphs while often being left alone to interpret them. Therefore, it is important to teach visual literacy and, more importantly, for those using such charts to communicate their agenda, to use adequate representations of the message to be conveyed. Furthermore, we also recognized how important "good" predictions are for decision making and planning and how central the power of data sovereignty is to convey and control the general public in such pandemic situations.

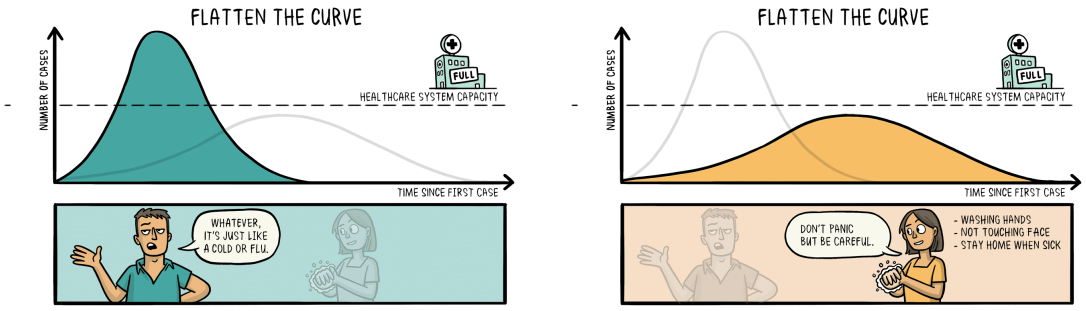


Figure 6.1: A comic about flattening the curve of COVID-19 cases, Wiles and Morris. The illustration was intended to engage people in precautionary measures for this pandemic situation, thereby preventing the collapse of the healthcare system.
Image Source: Siouxsie Wiles and Toby Morris (2020). Published under the Creative Commons Attribution-Share Alike 4.0 International license (CC-BY-SA). Images extracted from a gif animation retrieved from Wikipedia Commons: `https://commons.wikimedia.org/wiki/File:Covid-19-curves-graphic-social-v3.gif` (last visited on Sept. 26, 2020)

Visual literacy is also important to allow for questioning some visual representations while teaching some skepticism as well. Going back to the "inventor of statistical graphs" [Fun37, p. 280], William Playfair, we refer to his graph about government spending [Pla01, Plate 20], in which he wisely chose the aspect ratio of the graph to make the graph look like skyrocketing (see Figure 6.2 left). Tufte [Tuf83] mentioned that displaying government spending and debt over years is often done in a printed graphic. Most such information is displayed in a way like Playfair

used to do, to make them look like rapidly increasing. Tufte also noted that, in the text attached to this graph, Playfair wrote polemically about the "ruinous folly" [Tuf83, p. 65], meaning the British government, for financing colonial wars with debt, to underscore this intention. Yet in the next paragraph, Tufte referred to Playfair's integrity to have another graph showing the data with a better aspect ratio (see Figure 6.2 right). For this reason, it is important to consider appropriate representations of the data. Tufte was not the only one centrally concerned with the correct aspect ratio and context for graphic integrity of time series graphs; Cleveland [Cle93, Cle94] also introduced the concept of banking local segments to an angle of 45° for choosing an adequate aspect ratio. Ignoring such best practices helps tweak the visualization in a way to tell a different story as the data actually does and is at best misleading the interpretation of the audience.

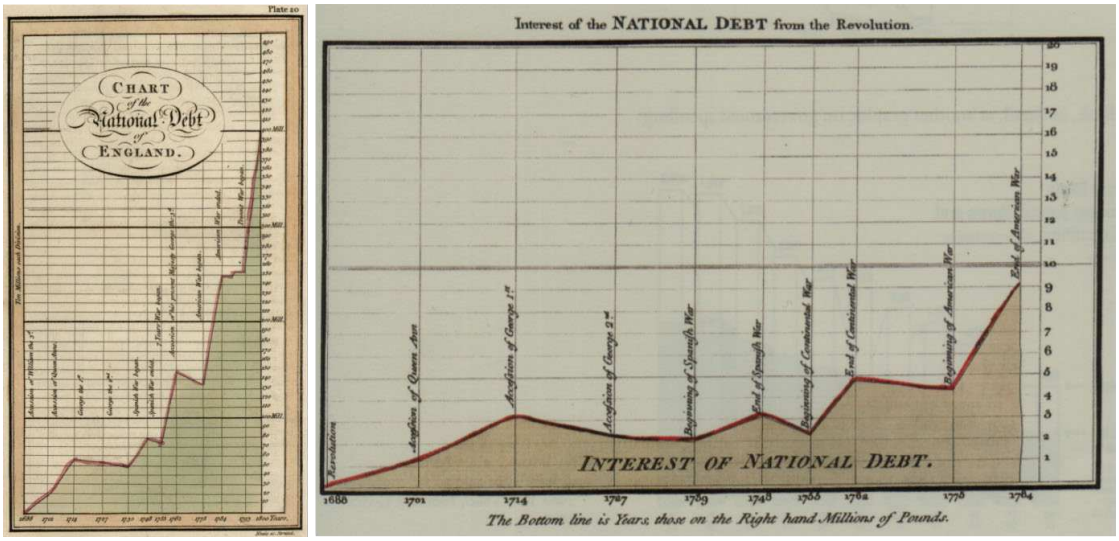


Figure 6.2: National debt of England in two different graphs, Playfair [Pla01]. The left one serves the intention to illustrate the "skyrocketing government debt" due to financing the British colonial wars [Tuf83, p. 65]. Image Source: Playfair (1801) [Pla01, plate 20 & p. 129]. First image retrieved from archive.org: `https://archive.org/details/PLAYFAIRWilliam1801TheCommercialandPoliticalAtlas` (last visited on Sept. 24, 2020); The second image was retrieved from Wikipedia Commons: `https://commons.wikimedia.org/wiki/File:Playfair_interest_national_debt.png` (last visited on Sept. 24, 2020)

Apparently, time series analysis and prediction as well as imputation and outlier detection are an important topic. In this dissertation, we applied visual analytics techniques to support users in doing so and embedded them into the context of historic visual analyses and recent challenges.

6.1 Summary

In the main body of this dissertation, we introduced visual analytics techniques and approaches to deal with certain aspects of the challenges in statistical time series analysis. In the Introduction (Chapter 1), we motivated these challenges in time series analysis; we then proposed our visual analytics techniques and approaches in Chapters 2–5 to provide support in solving them.

Specifically, we tried to support the tasks of time series model selection, parametrization, prediction, imputation, and outlier detection partly for univariate and multivariate periodic time series data. In the following sub-sections, we summarize and discuss our contributions to these challenges.

Time Series Model Selection. In Section 1.1 of Chapter 1, we presented the challenging and tedious tasks involved in time series analysis in general and in the iterative model selection process in particular and illustrated this model selection process, known as Box-Jenkins methodology (Section 2.3.1), and the related tasks, like model specification, model fitting, and model diagnostics in Section 2.3. The class of ARIMA/SARIMA models is applied in many different domains, such as epidemiology, economy, and environmental sciences. Essentially, the systematic approach of model selection proposed by Box and Jenkins [BJ70] demands a highly iterative process of multiple runs of parameter adjustments, recomputation, and analysis of the outcome, the model diagnostics, which involves a close intertwining of expert and domain knowledge, human judgment, as well as automated analysis and computation. We proposed a visual analytics process for guiding domain experts by combining these parts through interactive visual interfaces. We implemented our visual analytics process in a prototype and iteratively refined the prototype based on user stories and expert feedback on user experience. We applied the process and the prototype using an epidemiological dataset and provided a detailed walk-through using usage scenarios and experts' feedback. The results of the evaluation are the basis for answering and validating our research questions—specifically, sub-questions 1 and 2.

Using Prediction in Model Diagnostics. According to Tsay's history of time series analysis and forecasting [Tsa00], forecasting has an even longer history then statistical analysis of time series data, dating back to Yule 1927 [Yul27]. Time series models (e.g., ARIMA) are intended to produce prediction, which is synonymous with forecasting. In particular, the systematic approach proposed by Box and Jenkins [BJ70] allowed practitioners to apply such models for forecasting [Tsa00]. Using information criteria and residual plots, like we used in the overall ARIMA model selection process in Chapter 2, may only show small variations. Because the goal is to apply such a model for prediction, we integrate the prediction capabilities of the model into the process of model selection in Chapter 3. When including only predictions in the interactive visual interface, it is still difficult to compare deviations from actual values or benchmark models. In our visual analytics approach, we combine visual and analytical methods to integrate the prediction capabilities in the model selection process. This provides guidance in the decision for an adequate and parsimonious model by enabling the user to examine and judge prediction capabilities directly for one or multiple models. We proposed using a Qualizon graph representation and demonstrated in a usage scenario that this integration and adequate representation results in a less complex model selected. The discussion and the findings from this chapter were used to answer sub-questions 1 and 2 from our set of research questions.

Imputation of Missing Values in Periodic Time Series. In most real-world applications, missing data is a frequent data quality problem to address [KHP+11] before analysis methods can be applied. The issue with missing values for statistical methods is that these usually rely

on complete data [All09], and only a few specialized methods are applicable in the case of missing values [LR02, Jon80]. Imputation is a common way to bypass this issue and replace the missing values in order to apply established statistical methods. One concern is the uncertainty introduced in the data that is neglected in most cases of applying imputation to replace the missing data. Most commonly, repeated resampling [LR02] and multiple imputation techniques [Sch99] help calculate an error measure; for example, Monte Carlo-based simulations can be used to calculate confidence intervals. In this way, it is possible to communicate the uncertainty of the imputation using appropriate error boundaries or confidence intervals. In Chapter 4 we proposed an approach to integrate such uncertainty inherent in the imputed values and employed a cycle plot representation linked with a standard line chart to interactively compare imputed data points in the context of their neighboring values in linear and periodic time. By using an optimized visual representation for periodic time, we expanded on the possibilities of imputation and incorporated domain knowledge and contextualized neighboring values in linear and periodic time for better judgment of the appropriateness of the imputed values. The contribution of this chapter is directly linked to research sub-questions 1 and 3.

Outlier Detection in Multivariate Periodic Time Series. Another issue in real-world data is the data points that deviate significantly from the other values, which are called *outliers* [Agg13]. We already learned from the benefit of cycle plots for periodic time series to support the task of missing value imputation (Chapter 4). Based on this experience, we had the idea to apply cycle plot representations in an interactive exploration environment to identify outliers in periodic time series. Because analysts in most application domains are dealing with multivariate data, we extended the idea to support the outlier detection in multivariate periodic time series. The cycle plot is well established and effective for identifying and comprehending patterns in univariate periodic time series and allows for visually identifying and contextualizing extreme values and outliers from different perspectives. Because it is defined only on univariate data, we proposed in Chapter 5 a modified cycle plot using a distance-based abstraction to reduce it to one overview dimension and retain the established representation of the original cycle plot. In addition to the construction of this Mahalanobis distance-based cycle plot, we also integrated this novel type of cycle plot together with multiple classical cycle plots with multiple coordinated views in an interactive exploration environment. Using this approach, we could identify outliers in multivariate time series while considering the periodicity and support the interpretation and contextualization of multivariate outliers. It also helped reduce the information loss inevitably accompanying the multivariate data abstraction used. The reflections and lessons learned from this chapter contribute the last missing part to answer sub-questions 1 and 3 as well as the main research question.

6.2 Research Questions Revisited

We now revisit the research question stated in Section 1.6 and answer them based on the findings and argumentation presented in the main body of this dissertation. We start by answering the subordinate questions to subsequently derive from them an answer to the main question.

Sub-Question 2 How can visualization and interaction improve the process of model selection and parametrization for time series prediction tasks?

The highly iterative process of the ARIMA model selection, known as Box-Jenkins methodology, is usually applied in script-based computation environments, such as the R project for statistical computing, and mainly uses static plots (visualization) and numerical comparison of information criteria. If these suggest an adequate model, it is used for prediction. Our proposed interactive exploration environment built upon the R project for statistical computing in order to combine the strength of both worlds: the computational power of R, including the large amount of available packages for time series analysis, and an interactive exploration. To improve the model selection, we introduced intuitive and interactively linked views, known as multiple coordinated views, and direct comparison possibilities to immediately grasp the improvements in the model diagnostics when adapting the model's parameters. In addition, our visual analytics approach allowed us to incorporate the prediction capabilities and prediction performance during the model selection process and use additional measures to judge the appropriateness of the model for the given dataset. This integration of the prediction together with the input time series meant that we could also check how well the seasonal cycle is reproduced by the seasonal component of a SARIMA (seasonal ARIMA) model. In Chapters 2 and 3, we showed that these interactive exploration environments improved the model selection process, including the adjustment of parameters and integration of the prediction, using the means summarized herein.

Sub-Question 3 Is an adequate visual representation of periodic time series beneficial for imputation and outlier detection tasks in univariate and multivariate time series?

One common issue in a graphical representation aimed at detecting outliers in multivariate data is choosing the wrong representations or dimensions, which leads to hiding the outliers in the visual representation. Tufte [Tuf83, p. 14] used the famous example of a bivariate scatterplot, which immediately shows an outlier that would otherwise be hidden in the marginal distributions. We employed a multiple view approach in our interactive exploration environment (Chapter 5) to try to overcome this issue. In contrast to normal distance measures, like the Euclidean distance, the Mahalanobis distance considers the multivariate distribution of the data to compute the distance. We employed robust techniques for the calculation of distances so that it is less error prone and not influenced by outliers. In contrast to using multiple univariate cycle plots for each variable only, which would be prone in hiding outliers, the addition of the multivariate distance-based cycle plot allowed a separate perspective of the data and helped overcome this issue. Furthermore, we could find multivariate outliers that are not outliers in any of the separate variables. By using this distance-based abstraction together with a known and established visual representation of periodic time series in our novel multivariate cycle plot, users familiar with this representation can adapt to the exploration environment easily. It also makes it intuitive for exploring multivariate periodic time series and identifying possible outliers by comparing neighboring values in linear and periodic time in the multivariate and each of the univariate dimensions. The distance-based abstraction can apply this technique to an arbitrary number of dimensions. The combination of statistical and visualization techniques facilitates the balancing of benefits and limitations for both to visually analyze patterns and outliers in multivariate seasonal time series. This is done with

respect to the periodic structure and the relationship among multiple dimensions. Similarly, an adequate representation of periodic time series helps improve the imputation of missing values. In Chapter 4, we used the cycle plot to represent estimated values for missing values and similarly could investigate the imputed values in the context of their neighboring values in linear and periodic time. Although we introduced this for univariate periodic time series data and showed that it is an adequate support, we consider the techniques applied in Chapter 5 for outlier detection to be applicable to imputation tasks in multivariate periodic time series as well, similar to our work in Chapter 4. Essentially, this argumentation and discussion allows us to answer this research sub-question with **yes**.

Sub-Question 1 Is visual analytics an adequate support for the challenges in statistical time series analysis dealing with periodic time series for both univariate and multivariate time series data?

In Chapter 2, we demonstrated how an interactive visual exploration environment supports the established Box-Jenkins methodology for ARIMA/SARIMA model selection of univariate data by closely integrating and visualizing model diagnostics and animated transitions when adapting the model parameters. We showed how to adequately support the analysis of the model diagnostics using residual plots and information criteria. Integrating the prediction capabilities, as we introduced in Chapter 3, extended this approach and added adequate support for prediction tasks of univariate time series data. We also illustrated that, for periodic time series, the appropriate visual representation (cycle plot) for imputation tasks (Chapter 4) is inevitably beneficial to compare the estimated values for missing values in the context of linear and periodic time, which enabled us to judge the adequateness of the imputed values immediately. Based on the findings from using a cycle plot for imputation, we extended the cycle plot representation using a distance-based abstraction in order to use the cycle plot for multivariate data (Chapter 5). We combined such a multivariate distance-based cycle plot with univariate cycle plots and line charts in an interactive exploration environment to support the detection of outliers in multivariate periodic time series. For analysts, this additional perspective into multivariate periodic time series provides a superior investigation of patterns and outliers in this data. In essence, we can answer this sub-question with **yes**.

Main Research Question How can visual analytics support the challenges in statistical time series analysis of model selection, parametrization, prediction, imputation, and outlier detection?

A close integration of statistical computation into an intuitive, highly responsive, and interactive exploration environment, using adequate visual representation to support the interactive exploration, adaptation, and investigation of time series data and models, allows visual analytics to support the challenges in time series analysis. Specifically, this combination enables better support of the model selection, adapting the parameters, exploring the time series data, models, and model diagnostics, as well as integrating and exploring the prediction capabilities. In addition, we provided visual analytics techniques to explore periodic time series and the estimated values for imputed values in the relevant context of linear and periodic time and detected outliers in an adequate visual representation of multivariate periodic time series. Using sophisticated

data abstractions with well-established visual representations to visualize multivariate periodic structures, together with the underlying univariate components in linear and periodic time, we could investigate these outliers and/or find outliers that would otherwise be missed. Applying visual analytics for challenges in time series analysis extrapolates the idea of Tukey's ideas on exploratory data analysis and provides completely new possibilities for supporting the challenges of model selection, parameterization, prediction, imputation, and outlier detection.

6.3 Conclusion

Using visual analytics together with statistics allows novel solutions to the challenges in time series analysis that were not possible before. With today's advanced interactive systems and fast computation, it is possible to allow reasonably fast responses for even computationally expensive calculations and make it possible to do model adjustments and previews on the run. Opening the black box of models is one significant challenge that has been tackled in visual analytics research for several years already. Although opening the black box fully has still not been achieved, important steps have already been taken. The adequateness of models, judging the outlying nature of values and how well imputed values match the patterns in the time series, is better judged by a human analyst. Therefore, the integration of his/her expertise is critical for success in this regard. Although much of the outlier detection and imputation ultimately need to be automated, visual analytics is inevitably useful for adapting and parametrizing the algorithms, methods, and models for new or different data.

In the main body of this dissertation (Chapters 2, 3, 4, and 5), we specifically showed how visual analytics contributes to the challenges in time series analysis introduced in Chapter 1. We presented the answers to our research questions in the previous section, where we stated that applying visual analytics approaches to model selection and integrating prediction capabilities improve the process of finding an adequate model for a given dataset by allowing direct investigation of the diagnostic measures and prediction capabilities for the user. Specifically, for periodic time series, the adequate representation using cycle plots for univariate data and imputation tasks as well as a distance-based abstraction to visualize a multivariate cycle plot for investigating outliers in an interactive exploration environment using robust statistical methods for this abstraction allows us to detect multivariate and univariate outliers. When dealing with periodic time series, it is important to consider the periodic structure of time prominently in the visual analytics approach.

6.4 Open Challenges and Future Opportunities

Although we proposed and introduced solutions for solving some of the stated problems in time series analysis, there are still open challenges that need to be considered, and each of them indicates research opportunities for future work. Two articles [Tsa00, DGH06] have discussed the history of time series analysis and forecasting, and both include a discussion about future research in that field. We consider some of the discussed challenges as great opportunities for visual analytics research in time series analysis as well. In addition, there are some recent articles about the vision and challenges in visual analytics research related to the topics covered in this dissertation

that we consider in the following discussions—namely, spatio-temporal visual analytics [AA20], visual analytics and machine learning [ERT+17, HKPC19, JLC19], and predictive visual analytics [LGH+17, KPB16].

Cyclic Time Series Without Fixed Periodic Length. Strictly speaking, our proposed solutions presented in the main body of this dissertation mainly considered seasonal cycles, meaning periodic cycles with fixed periodic length (frequency). Although in a large portion of practical applications and real-world data this is the case, we consider our proposed techniques to be applicable for periodic time series in general. We expect that adaptations and extensions are required to deal with cycles having different periodic lengths and consider this to be an opportunity for further research. Most likely, small variations in periodic length may be possible to handle with simple adaptations, but for larger variations and/or multiple nested cycles and seasons, it may require advanced and specialized models and visual representations or even the embedding of other approaches from, for example, machine learning. Dealing with such advanced models and methods creates additional challenges, which we discuss in the following sub-section.

Machine Learning and Advanced Models for Prediction. De Gooijer and Hyndman [DGH06] dedicated a whole section to artificial neural networks (ANN) for predicting and considering the model complexity, over-parametrization, and risk of overfitting as major challenges in addition to their power in forecasting. Since 2006, research in the area of ANN has been enhanced, and we reached the deep learning arena in the meantime. Although they allow new possibilities for prediction, the complexity of networks has increased drastically. In recent years, there has been a strong demand for better explaining and understanding how such complex machine learning techniques, like deep learning approaches, come to their results while visualization, especially visual analytics, is a means for communicating that and allows exploration for building an understanding of how the results are reached. For simpler time series models, like the ARIMA and SARIMA models we used in our visual analytics approach in Chapter 2, the interactive adjustment of model parameters and the diagnostic plots allowed a comprehensible understanding of the model components and how the model maps to the input time series. More research is needed to find appropriate techniques and methods to achieve this for more complex models. The ability to explain and understand such complex models is a widely discussed topic, including in the visualization research community. Interpretability and trust building are of central concern and challenge to which visualization and visual analytics can contribute. According to Krause et al. [KPB16], one desirable reason for human involvement in visual analytics is when understanding and interpretation are required. They identified three main needs of interpretability of machine learning models, where visual analytics can contribute "data understanding and discovery; trust building and accountability; model comparison and diagnostics" [KPB16]. Recent state-of-the-art research on machine learning combined with visual analytics [ERT+17] provides a more detailed discussion on enhancing the trust and interpretability as open challenges and opportunities. Jiang et al. [JLC19] also included the ability to explain in their research challenges about interactive machine learning. Meanwhile, Andrienko and Andrienko, in their recent vision of spatio-temporal visual analytics for 2020 [AA20], posed the issue of uncritical trust in results produced by computers or models. They argued that naive analysts may not be aware of the big changes in

results when adapting model parameters slightly while experienced analysts may just trust the numbers and not make the effort to get a better understanding. In this situation, interpretability and trust building are essential, but the effort to gain that trust and interpretations needs to be accepted by analysts. Hohman et al. [HKPC19] specifically discussed the issues in understanding and interpretability using visual analytics in deep learning approaches. Because of the high complexity and size of deep learning approaches, it is especially challenging to achieve. From their comprehensive survey, they derived the most pressing challenges, where visual analytics can contribute to solving them. These are in concordance to the previously mentioned literature concerned primarily with interpretability, human involvement in interpretability, trust, and bias detection [HKPC19, pp. 17–18]. When considering extending our visual analytics approaches using more complex and advanced models and machine learning methods for tasks, like outlier detection, imputation, and prediction, it is important to pay attention to these extensive challenges that come with this decision.

Integrating Prior Knowledge and Specifying Objectives for Prediction. De Gooijer and Hyndman [DGH06] identified the need for model selection procedures to use the data together with prior knowledge and also enable the definition of objectives for the forecasts that are considered in the model selection procedure. A state-of-the-art survey of predictive visual analytics by Lu et al. [LGH+17] also identified the integration of user knowledge as a challenge for future research. They also discussed the issues that arise when integrating user knowledge and allowing them to adapt predictions. Lu et al. specifically raised unanswered questions regarding how it is possible to "regulate or constrain knowledge integration so that we get the benefits of domain knowledge, social and emotional intuition, and minimize the costs of introducing bias? How much human-in-the-loop is the right amount?" [LGH+17, pp. 554–555] We consider our contribution in Chapter 3 to integrate the prediction capabilities into the model selection process as a simple first step into this direction. Our approach simply allows an adaptive way to adjust the model based on the data and prior knowledge about the data and domain to select the model, but also to form an informal form of objective and expectation into the prediction capabilities of the model. It is only a first step because, in their understanding, the objectives to be met need to be more formally defined and integrated too. We still consider visual analytics methods to be a good fit for defining, expressing, and judging such objectives as well as integrating prior knowledge into such a model selection process.

Parametrization of Multivariate Time Series Models. Another challenge mentioned by De Gooijer and Hyndman [DGH06] was multivariate time series models. Although extensive research in theory and practice has already been done and there are methods to use, these are still very difficult and complex to apply. In particular, there is appropriate software support missing for applications. In addition, Tsay [Tsa00] foresaw the mixtures of discrete and continuous variables in multivariate time series as a challenge for future time series research. One other specific challenge with existing multivariate methods is the large number of parameters involved and the resulting difficulty for parameter estimation, computational complexity, and time needed for computation [DGH06]. The parametrization of models and algorithms is of central concern in many applications, and visual analytics research has picked up this issue and provided support in

parametrization for different methods and application domains. For example, the author of this dissertation was involved in a work by Röhlig et al. [RLK+15] that employed a visual analytics approach for segmenting and labeling time series data and investigating the influence of parameters to the results. With this experience, we can state that visual analytics is a possible solution to support a large number of parameters and help shape an understanding of the influence of the parameters on the results. Tsay [Tsa00] explored the importance of multivariate models in order to satisfy the interest in investigating the dynamic relationships between variables, and he stated that, with the advances of computational methods, these vector ARMA and state-spaced models can be more practically applied. Future research could investigate how visual analytics can be applied and support the challenges of multivariate time series model selection.

Detection of Interesting Structures in Time Series. In practical applications and real-world data, there are several interesting other structures in time series that need to be considered. In this work, we have focused on missing values of equally spaced time series as well as on simple outlier detection (Chapters 4 and 5). Tsay [Tsa00] identified some special features in the data that challenge existing methods and will require additional research to cope with them. For example, he mentioned unequally spaced observations as one specific challenge. Tsay also foresaw a trend to investigate in more detail the time duration between observations in such time series and stated that the times of occurrence will be more important for the analysis and prediction. In addition to specific time series model approaches [Jon85], one possible approach is to apply rastering to transform such time series. This approach has its own issues and concerns, but allows the application of standard methods afterwards. The author of this dissertation contributed to a work providing visual support for such a rastering of unequally spaced time series [BBGM17], but there is more research and work to be done. This research should not only look more into supporting such transformations, but also help investigate and explore such unequally spaced time series as well as apply specialized models and methods compensating these unequally spaced time series. Another interesting challenge in addition to outliers is structural breaks, level shifts, and location shifts [DGH06]. These structural breaks and shifts essentially mean that a time series model that would fit some parts of the time series adequately is not appropriate in other segments, such as exactly around the shifts and breaks or the segments following them. A possible solution is to partition or segment parts of the time series according to their similarities and then apply different models to each of the segmented classes. The author of this dissertation already contributed to some work about such visual support for segmenting and labeling time series [ABG+14, RLS+14, RLK+15, BDB+16, BBB+18, BBGM18], but this idea of using such an approach and then applying different time series models for each resulting group, has not been pursued yet. In addition, with the high-volume and long-term data available nowadays, there is a higher chance of these kinds of effects in the data. Therefore, it is important to look into methods to detect and understand them in order to adapt the applied models accordingly. In order to detect such effects, patterns, and behavior, it is necessary to combine statistical computational methods with visualization to provide a visual analytics solution for this challenge.

Predictive Visual Analytics. In an extensive survey article on predictive visual analytics, Lu et al. [LGH+17] presented recent advances in visual analytics support for predictive analytics, where

predictive analytics is used as an umbrella term for prediction techniques from statistical modeling, machine learning, and data mining. In addition to their overview on the visual analytics systems and techniques applied for predictive analytics found in the literature, they also offered an outlook for future challenges and research directions in predictive visual analytics based on their findings from the literature survey and internal discussions. Some of their named challenges correspond to the challenges mentioned herein. They also identified the integration of user knowledge as well as the scaling to larger and more complex models. Furthermore, Lu et al. considered deep learning approaches for prediction as a current trend. This trend of such increasingly large and complex models increases the challenges in interpretability and trust in such models, as we have discussed. The buzzword *explainable AI (artificial intelligence)* has been used for research efforts in trying to deal with the challenge of interpretability and ability to explain in this regard.

Appropriate Tools Depending on Type of User. Lu et al. [LGH+17] raised an important discussion on the relationship between the type of user interacting with a predictive visual analytics system and how his/her knowledge can contribute to the process. They identified three types of users in the scope of predictive visual analytics, depending on their knowledge: end-users, domain experts, and modeling experts. End-users are neither experts in the specific domain of application nor have knowledge about predictive models or methods. Domain experts have a great understanding of the domain and the data involved. Modeling experts, on the other hand, have advanced knowledge about predictive models and techniques, but no in-depth understanding of the domain. Visual analytics methods for supporting modeling and prediction can of course target users with different backgrounds, but this needs to be considered and distinguished from the beginning of the design. In addition, in visual analytics research, this distinction needs to be individually investigated and to determine what the appropriate methods and techniques for each user type are. Failing to do so may lead to faulty results and decisions and, consequently, mistrust. Andrienko and Andrienko [AA20] stated the danger of uncritical trust versus being overly critical of results generated by an analysis algorithm. This can be caused, for example, by mixing up or not strictly specifying the intended user type and providing an inadequate method or technique for the wrong type of user. For instance, an end-user might use only an initial parametrization of a model because s/he is unable to cope with the advance adaptation possibilities of the parametrization and, as a result, uncritically trusts the outcome. Therefore, Andrienko and Andrienko emphasized the underlying philosophy of visual analytics, the "primacy of human understanding and reasoning and awareness of the weaknesses of computers, which cannot see, understand, and think, and thus need to be led and controlled by humans" [AA20, p. 92]. De Gooijer and Hyndman [DGH06, p. 461] also mentioned the danger of the misuse of time series methods, such as if there are outliers or shifts that are difficult to detect. They suggested that the advances in robust statistics need to be considered with more attention in the forecasting community. Another related challenge is availability and accessibility of appropriate practical visual analytics solutions, as discussed by Andrienko and Andrienko [AA20]. They claimed that the vast amount of contributions in visualization and visual analytics techniques is developed as research prototypes and do not seem likely to be transferred into accessible and reliable software in the near future. Although there is a current trend in publishing these techniques, methods, and approaches in open-source libraries, mainly for current state-of-the-art data science languages like R and Python, to make

them accessible through web-based environments there is again a danger of misuse, because such libraries are often adapted by people who lack experience in appropriate visualization and visual analytics techniques [AA20]. For a broader utilization of the predictive visual analytics system, Lu et al. [LGH+17] also identified the challenge of improving the user experience, which has a connection to appropriate support depending on the type of user.

Responsibility—Balancing Human and Machine Effort. Recent papers presenting research challenges in visual analytics for spatio-temporal data [AA20], machine learning [ERT+17, HKPC19], and predictive visual analytics [LGH+17] have discussed the issue of economics in balancing the effort between human and machine in visual analytics approaches. The costs of effort and time needed between human users and machines are becoming more relevant to consider when creating visual analytics solutions. Andrienko and Andrienko called it the "effective division of labor between the human and the computer" [AA20, p. 91]. This challenge has increased in recent years because of the ever-growing massive amounts of big data available, the natural limitations of human capabilities to comprehend information, and the centrality of human involvement ("human-is-the-loop") in visual analytics. Since the beginning of visual analytics research, there has already been an emphasis on automated analysis as much as possible and human involvement where necessary. As prominently represented by Daniel Keim, "[i]n many cases automated analytics is favored towards interactive visual analysis since getting the user involved in the analysis process can be an unpredictable and cost-intensive undertaking" [KMT09, p. 6]. This has been widely discussed in recent papers, including those referred to herein. Endert et al. [ERT+17] argued that, although there are established methodologies to decompose tasks and divide them into sub-tasks that are better done by the user or faster by the computer, there is still no generalizable empirical evidence on how to balance the effort between these two entities. They even argue the lack of metrics for measuring the effort of users compared to the systems in such mixed-initiative systems. Andrienko and Andrienko embarked on this discussion with an exciting topic—what they call "orchestrated automatic model adaptation mechanisms" [AA20, p. 91] reacting to data dynamics. This idea helps move more tasks to the computer and reduce the effort of human involvement, costs, and reaction times in changing environments. The basic idea is not to readjust a model in a visual analytics model building environment every time new data arrives or data changes and the model no longer maps the data adequately, which would always require a human analyst to execute the adaption, but rather—based on the knowledge of the expected changes in the data—to foresee such changes in the model and, if the data changes accordingly, trigger such model adaptations automatically. This is a fascinating approach that we could imagine being beneficial in time series modelling and prediction scenarios with data containing structural breaks and shifts.

We extensively discussed open challenges and future research opportunities that opened up during the research on this dissertation. As usual in research projects, the further you investigate into one topic, the more you realize what is left open to be done. In the following section, we list the publications published by the author of this dissertation, Markus Bögl, during his PhD and discuss his contributions to each of the papers.

The work on this dissertation took more time and resources than expected and occasionally dominated the author's free time, yet the author learned a lot personally, scientifically, and

occupationally, moving forward plenty of work and overcoming obstacles. The result was a feeling of accomplishment that evolved in the final phase. Serendipitously, it is nearly complete now, allowing me to set off for new shores and horizons. To conclude, I want to come back to the quote from the R.E.M. song from the beginning of this section.

> *"It's the end of the world as we know it and I feel fine—time I had some time alone"*
>
> — R.E.M. (1987), from the Album *Document*

6.5 Publications

The findings of the research conducted by Markus Bögl during his PhD program were presented at international conferences and symposia, like IEEE VIS, Eurographics (EG) EuroVIS, and EuroVA, in front of a high-profile scientific audience. The results were published in international journals, like *Computer Graphics Forum* and *IEEE Transactions in Visualization and Computer Graphics*, as well as in conference proceedings (EG EuroVIS, IEEE VIS). The following list summarizes the main publications and explains the roles and contributions of the author of this dissertation in each of the publications. Chapters 2, 3, 4, and 5 are self-contained and are each based on the research publications listed in the main publications below. In addition, we list publications to which the author of this dissertation contributed as a co-author during his PhD program.

6.5.1 Main Publications

[BFG+17] Markus Bögl, Peter Filzmoser, Theresia Gschwandtner, Tim Lammarsch, Roger A. Leite, Silvia Miksch, and Alexander Rind. Cycle plot revisited: Multivariate outlier detection using a distance-based abstraction. *Computer Graphics Forum*, 36(3):227–238, 2017 – **Journal**

[BAF+15] Markus Bögl, Wolfgang Aigner, Peter Filzmoser, Theresia Gschwandtner, Tim Lammarsch, Silvia Miksch, and Alexander Rind. Integrating predictions in time series model selection. In *Proceedings of the 6th International EuroVis Workshop on Visual Analytics, EuroVA@EuroVis 2015, Cagliari, Sardinia, Italy, May 25-26, 2015*, pages 73–77. The Eurographics Association, 2015 – **Workshop**

[BAF+14] Markus Bögl, Wolfgang Aigner, Peter Filzmoser, Theresia Gschwandtner, Tim Lammarsch, Silvia Miksch, and Alexander Rind. Visual analytics methods to guide diagnostics for time series model predictions. In *Proceedings of the 2014 IEEE VIS Workshop on Visualization for Predictive Analytics*, 2014 – **Workshop**

[BFG+15] Markus Bögl, Peter Filzmoser, Theresia Gschwandtner, Silvia Miksch, Wolfgang Aigner, Alexander Rind, and Tim Lammarsch. Visually and statistically guided imputation of missing values in univariate seasonal time series. In *Proceedings of the IEEE Conference on*

Visual Analytics Science and Technology, VAST – Posters, Chicago, IL, USA, October 25-30, 2015, pages 189–190. IEEE, 2015 – **Poster**

[BAF+13] Markus Bögl, Wolfgang Aigner, Peter Filzmoser, Tim Lammarsch, Silvia Miksch, and Alexander Rind. Visual analytics for model selection in time series analysis. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2237–2246, 2013 – **Journal**

Contributions. *Markus Bögl* was the lead author of all the above listed papers [BFG+17, BAF+15, BFG+15, BAF+14, BAF+13] and was responsible for most of the scientific contributions, including the main idea, visualization design, data abstraction, organization of the paper, writing of the paper, and the entire submission process as well as the presentation in front of the scientific community at international conferences. He was also responsible for project meetings about the respective topics and discussions with co-authors for feedback and iterative improvements. While writing the papers, he gathered feedback from the co-authors and integrated the suggested improvements as they fit the overall ideas. For the paper [BFG+17], he implemented the data abstraction method and computed and prepared the data for the visualization. He advised the prototypical implementation for the showcase, which was implemented by one of the co-authors. The supplementary material attached in Section 5.9 to support the reader in following the discussion on the usage scenario (Section 5.6) was prepared by the same co-author based on the text and explanations from the main text written by Markus Bögl. For the papers [BAF+14, BAF+15, BFG+15], Markus Bögl drafted the figures for the design and computed and prepared the data for the visualization. For the paper [BAF+13], he implemented the prototype as showcased and computed and prepared the data for the visualization. For the paper [BFG+15], he designed and prepared a poster for the presentation at the international scientific conference. This poster was awarded the Best Poster Award at the 2015 IEEE VAST conference in Chicago, Illinois, USA.

6.5.2 Additional Publications

[BBB+19] Christian Bors, Jürgen Bernard, Markus Bögl, Theresia Gschwandtner, Jörn Kohlhammer, and Silvia Miksch. Quantifying uncertainty in multivariate time series pre-processing. In *Proceedings of the 10th International EuroVis Workshop on Visual Analytics, EuroVA@EuroVis 2019, June 3, 2019, Porto, Portugal*, pages 31–35. The Eurographics Association, 2019

[BBGM18] Markus Bögl, Christian Bors, Theresia Gschwandtner, and Silvia Miksch. Categorizing uncertainties in the process of segmenting and labeling time series data. In *Proceedings of the 20th Eurographics Conference on Visualization, EuroVis 2018 – Posters, Brno, Czech Republic, June 4-8, 2018*, pages 45–47. The Eurographics Association, 2018

[BBB+18] Jürgen Bernard, Christian Bors, Markus Bögl, Christian Eichner, Theresia Gschwandtner, Silvia Miksch, Heidrun Schumann, and Jörn Kohlhammer. Combining the automated segmentation and visual analysis of multivariate time series. In *Proceedings of the 9th International*

EuroVis Workshop on Visual Analytics, EuroVA@EuroVis 2018, Brno, Czech Republic, June 4, 2018, pages 49–53. The Eurographics Association, 2018

[BBGM17] Christian Bors, Markus Bögl, Theresia Gschwandtner, and Silvia Miksch. Visual support for rastering of unequally spaced time series. In *Proceedings of the 10th International Symposium on Visual Information Communication and Interaction, VINCI 2017, Bangkok, Thailand, August 14-16, 2017*, pages 53–57. ACM, 2017

[GBFM16] Theresia Gschwandtner, Markus Bögl, Paolo Federico, and Silvia Miksch. Visual encodings of temporal uncertainty: A comparative user study. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):539–548, 2016

[BDB+16] Jürgen Bernard, Eduard Dobermann, Markus Bögl, Martin Röhlig, Anna Vögele, and Jörn Kohlhammer. Visual-interactive segmentation of multivariate time series. In *Proceedings of the 7th International EuroVis Workshop on Visual Analytics, EuroVA@EuroVis 2016, Groningen, The Netherlands, June 6-7, 2016*, pages 31–35. The Eurographics Association, 2016

[RLK+15] Martin Röhlig, Martin Luboschik, Frank Krüger, Thomas Kirste, Heidrun Schumann, Markus Bögl, Bilal Alsallakh, and Silvia Miksch. Supporting activity recognition by visual analytics. In *Proceedings of the 2015 IEEE Conference on Visual Analytics Science and Technology, VAST 2015, Chicago, IL, USA, October 25-30, 2015*, pages 41–48. IEEE, 2015

[RLS+14] Martin Röhlig, Martin Luboschik, Heidrun Schumann, Markus Bögl, Bilal Alsallakh, and Silvia Miksch. Analyzing parameter influence on time-series segmentation and labeling. In *Proceedings of the 2014 IEEE Conference on Visual Analytics Science and Technology, VAST – Posters, Paris, France, October 25-31, 2014*, pages 269–270. IEEE, 2014

[ABG+14] Bilal Alsallakh, Markus Bögl, Theresia Gschwandtner, Silvia Miksch, Bilal Esmael, Arghad Arnaout, Gerhard Thonhauser, and Philipp Zöllner. A visual analytics approach to segmenting and labeling multivariate time series data. In *Proceedings of the 5th International EuroVis Workshop on Visual Analytics, EuroVA@EuroVis 2014, Swansea, UK, June 9-10, 2014*. The Eurographics Association, 2014

[LAB+13] Tim Lammarsch, Wolfgang Aigner, Alessio Bertone, Markus Bögl, Theresia Gschwandtner, Silvia Miksch, and Alexander Rind. Interactive visual transformation for symbolic representation of time-oriented data. In *Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data - Third International Workshop, HCI-KDD 2013, Held at SouthCHI 2013, Maribor, Slovenia, July 1-3, 2013. Proceedings*, volume 7947 of *Lecture Notes in Computer Science*, pages 400–419. Springer, 2013

List of Figures

List of Tables

Bibliography

[AA06] Natalia Andrienko and Gennady Andrienko. *Exploratory Analysis of Spatial and Temporal Data: A Systematic Approach*. Springer Science & Business Media, 2006.

[AA20] Natalia Andrienko and Gennady Andrienko. Spatio-temporal visual analytics: a vision for 2020s. *Journal of Spatial Information Science*, 2020(20):87–95, 2020.

[ABG+14] Bilal Alsallakh, Markus Bögl, Theresia Gschwandtner, Silvia Miksch, Bilal Esmael, Arghad Arnaout, Gerhard Thonhauser, and Philipp Zöllner. A visual analytics approach to segmenting and labeling multivariate time series data. In *Proceedings of the 5th International EuroVis Workshop on Visual Analytics, EuroVA@EuroVis 2014, Swansea, UK, June 9-10, 2014*. The Eurographics Association, 2014.

[Agg13] Charu C. Aggarwal. *Outlier Analysis*. Springer New York, 2013.

[ALA+18] Natalia Andrienko, Tim Lammarsch, Gennady Andrienko, Georg Fuchs, Daniel Keim, Silvia Miksch, and Alexander Rind. Viewing visual analytics as model building. *Computer Graphics Forum*, 37(6):275–299, 2018.

[All09] Paul Allison. Missing data. In *The SAGE Handbook of Quantitative Methods in Psychology*, chapter 4, pages 72–89. SAGE, 2009.

[AMM+07] Wolfgang Aigner, Silvia Miksch, Wolfgang Müller, Heidrun Schumann, and Christian Tominski. Visualizing time-oriented data—a systematic view. *Computers & Graphics*, 31(3):401–409, 2007.

[AMST11] Wolfgang Aigner, Silvia Miksch, Heidrun Schumann, and Christian Tominski. *Visualization of Time-Oriented Data*. Springer, London, UK, 2011.

[Ans73] Francis J. Anscombe. Graphs in statistical analysis. *The American Statistician*, 27(1):17–21, Feb. 1973.

[BAF+13] Markus Bögl, Wolfgang Aigner, Peter Filzmoser, Tim Lammarsch, Silvia Miksch, and Alexander Rind. Visual analytics for model selection in time series analysis. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2237–2246, 2013.

[BAF+14] Markus Bögl, Wolfgang Aigner, Peter Filzmoser, Theresia Gschwandtner, Tim Lammarsch, Silvia Miksch, and Alexander Rind. Visual analytics methods to guide diagnostics for time series model predictions. In *Proceedings of the 2014 IEEE VIS Workshop on Visualization for Predictive Analytics*, 2014.

[BAF+15] Markus Bögl, Wolfgang Aigner, Peter Filzmoser, Theresia Gschwandtner, Tim Lammarsch, Silvia Miksch, and Alexander Rind. Integrating predictions in time series model selection. In *Proceedings of the 6th International EuroVis Workshop on Visual Analytics, EuroVA@EuroVis 2015, Cagliari, Sardinia, Italy, May 25-26, 2015*, pages 73–77. The Eurographics Association, 2015.

[BAP+05] Paolo Buono, Aleks Aris, Catherine Plaisant, Amir Khella, and Ben Shneiderman. Interactive pattern search in time series. In *Proceedings of the Conference on Visualization and Data Analysis, VDA*, pages 175–186, 2005.

[BBB+18] Jürgen Bernard, Christian Bors, Markus Bögl, Christian Eichner, Theresia Gschwandtner, Silvia Miksch, Heidrun Schumann, and Jörn Kohlhammer. Combining the automated segmentation and visual analysis of multivariate time series. In *Proceedings of the 9th International EuroVis Workshop on Visual Analytics, EuroVA@EuroVis 2018, Brno, Czech Republic, June 4, 2018*, pages 49–53. The Eurographics Association, 2018.

[BBB+19] Christian Bors, Jürgen Bernard, Markus Bögl, Theresia Gschwandtner, Jörn Kohlhammer, and Silvia Miksch. Quantifying uncertainty in multivariate time series pre-processing. In *Proceedings of the 10th International EuroVis Workshop on Visual Analytics, EuroVA@EuroVis 2019, June 3, 2019, Porto, Portugal*, pages 31–35. The Eurographics Association, 2019.

[BBGM17] Christian Bors, Markus Bögl, Theresia Gschwandtner, and Silvia Miksch. Visual support for rastering of unequally spaced time series. In *Proceedings of the 10th International Symposium on Visual Information Communication and Interaction, VINCI 2017, Bangkok, Thailand, August 14-16, 2017*, pages 53–57. ACM, 2017.

[BBGM18] Markus Bögl, Christian Bors, Theresia Gschwandtner, and Silvia Miksch. Categorizing uncertainties in the process of segmenting and labeling time series data. In *Proceedings of the 20th Eurographics Conference on Visualization, EuroVis 2018 – Posters, Brno, Czech Republic, June 4-8, 2018*, pages 45–47. The Eurographics Association, 2018.

[BBH11] Kerstin Bunte, Michael Biehl, and Barbara Hammer. A general framework for dimensionality-reducing data visualization mapping. *Neural Computation*, 24(3):771–804, 2011.

[BD87] George E. P. Box and Norman R. Draper. *Empirical Model Building and Response Surfaces*. John Wiley & Sons, NY, USA, 1987.

[BD91] Peter J. Brockwell and Richard A. Davis. *Time Series: Theory and Methods*. Springer Series in Statistics. Springer, New York, NY, USA, 2nd edition, 1991.

[BD02] Peter J. Brockwell and Richard A. Davis. *Introduction to Time Series and Forecasting*. Springer Texts in Statistics. Springer, New York, NY, USA, 2nd edition, 2002.

[BD10] Adrian G. Barnett and Annette J. Dobson. *Analysing Seasonal Health Data*. Statistics for Biology and Health. Springer, Berlin Heidelberg, 2010.

[BDB+16] Jürgen Bernard, Eduard Dobermann, Markus Bögl, Martin Röhlig, Anna Vögele, and Jörn Kohlhammer. Visual-interactive segmentation of multivariate time series. In *Proceedings of the 7th International EuroVis Workshop on Visual Analytics, EuroVA@EuroVis 2016, Groningen, The Netherlands, June 6-7, 2016*, pages 31–35. The Eurographics Association, 2016.

[Bec00] Kent Beck. *Extreme Programming Explained: Embrace Change*. Addison-Wesley, Reading, MA, 2000.

[Ber83] Jacques Bertin. *Semiology of Graphics*. The University of Wisconsin Press, 1983. translated by William J. Berg, originally published in french "Sémiologie graphique" in 1967.

[BFG+15] Markus Bögl, Peter Filzmoser, Theresia Gschwandtner, Silvia Miksch, Wolfgang Aigner, Alexander Rind, and Tim Lammarsch. Visually and statistically guided imputation of missing values in univariate seasonal time series. In *Proceedings of the IEEE Conference on Visual Analytics Science and Technology, VAST – Posters, Chicago, IL, USA, October 25-30, 2015*, pages 189–190. IEEE, 2015.

[BFG+17] Markus Bögl, Peter Filzmoser, Theresia Gschwandtner, Tim Lammarsch, Roger A. Leite, Silvia Miksch, and Alexander Rind. Cycle plot revisited: Multivariate outlier detection using a distance-based abstraction. *Computer Graphics Forum*, 36(3):227–238, 2017.

[BG05] Irad Ben-Gal. Outlier detection. In Oded Maimon and Lior Rokach, editors, *Data Mining and Knowledge Discovery Handbook*, pages 131–146. Springer US, 2005.

[BJ70] George E. P. Box and Gwilym M. Jenkins. *Time Series Analysis: Forecasting and Control*. Holden-Day, San Francisco, CA, USA, 1970.

[BJR08] George E. P. Box, Gwilym M. Jenkins, and Gregory C. Reinsel. *Time Series Analysis: Forecasting and Control*. John Wiley & Sons, Hoboken, USA, 4th edition, 2008.

[BJW00] Claudio Bettini, Sushil G. Jajodia, and Sean X. Wang. *Time Granularities in Databases, Data Mining and Temporal Reasoning*. Springer, 2000.

[BK11] Søren Bisgaard and Murat Kulahci. *Time Series Analysis and Forecasting by Example*. John Wiley & Sons, Hoboken, USA, 2011.

[BL98] Vic Barnett and Toby Lewis. *Outliers in statistical data*. Wiley, 3rd edition, 1998.

[BM13] Matthew Brehmer and Tamara Munzner. A multi-level typology of abstract visualization tasks. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2376–2385, 2013.

[BP70] George E. P. Box and David A. Pierce. Distribution of residual autocorrelations in autoregressive-integrated moving average time series models. *Journal of the American Statistical Association*, 65(332):1509–1526, 1970.

[BPC+10] Rita Borgo, Karl Proctor, Min Chen, Heike Jänicke, Tavi Murray, and Ian Thornton. Evaluating the impact of task demands and block resolution on the effectiveness of pixel-based visualization. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):963–972, 2010.

[BPS+07] Paolo Buono, Catherine Plaisant, Adalberto Simeone, Aleks Aris, Ben Shneiderman, Galit Shmueli, and Wolfgang Jank. Similarity-based forecasting with simultaneous previews: A river plot interface for time series forecasting. In *Proceedings of the International Conference on Information Visualisation*, pages 191–196. IEEE, 2007.

[BRG+12] Jürgen Bernard, Tobias Ruppert, Oliver Goroll, Thorsten May, and Jörn Kohlhammer. Visual-interactive preprocessing of time series data. In Andreas Kerren and Stefan Seipel, editors, *Proceedings of SIGRAD, Interactive Visual Analysis of Data*, volume 81, pages 39–48. Linköping University Electronic Press, 2012.

[Bro11] Peter J. Brockwell. Time series. In Miodrag Lovric, editor, *International Encyclopedia of Statistical Science*, pages 1601–1605. Springer, Berlin, Heidelberg, 2011.

[BSH+16] Benjamin Bach, Conglei Shi, Nicolas Heulot, Tara Madhyastha, Tom Grabowski, and Pierre Dragicevic. Time Curves: Folding time to visualize patterns of temporal evolution in data. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):559–568, 2016.

[BW08] Lee Byron and Martin Wattenberg. Stacked graphs – geometry & aesthetics. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1245–1252, 2008.

[CB14] Li Choy Chong and Luisella Balestra. The problem of time. *Modern Economy*, 5(3):250–258, 2014. Number: 3 Publisher: Scientific Research Publishing.

[CC08] Jonathan D. Cryer and Kung-Sik Chan. *Time Series Analysis. With Applications in R*. Springer Texts in Statistics. Springer, New York, NY, USA, 2nd edition, 2008.

[CCH13] Xiaoyue Cheng, Dianne Cook, and Heike Hofmann. *MissingDataGUI: A GUI for Missing Data Exploration*, 2013. R package version 0.1-5.

[CENC03] Javier Contreras, Rosario Espinola, Francisco J. Nogales, and Antonio J. Conejo. ARIMA models to predict next-day electricity prices. *IEEE Transactions on Power Systems*, 18(3):1014–1020, 2003.

[CG16] Min Chen and Amos Golan. What may visualization processes optimize? *IEEE Transactions on Visualization and Computer Graphics*, 22(12):2619–2632, 2016.

[CL13] Allin Cottrell and Riccardo Lucchetti. *Gretl: GNU Regression, Econometric and Time-series Library - User's Guide, Version 1.9.12*, March 2013. `http://gretl.sourceforge.net` (last visited on Oct. 12, 2020).

[Cle93] William S. Cleveland. *Visualizing data*. Hobart Press, Summit, NJ, USA, 1993.

[Cle94] William S. Cleveland. *The elements of graphing data*. Hobart Press, Summit, NJ, USA, 1994.

[CM09] Paul S.P. Cowpertwait and Andrew V. Metcalfe. *Introductory Time Series with R*. Use R! Springer, New York, NY, USA, 2009.

[CMS99] Stuart K. Card, Jock D. Mackinlay, and Ben Shneiderman. Using vision to think. In *Readings in information visualization: using vision to think*, pages 579–581. Morgan Kaufmann Publishers Inc., 1999.

[Coh04] Mike Cohn. *User Stories Applied: For Agile Software Development*. Addison-Wesley, Redwood City, CA, USA, 2004.

[Coh10] Mike Cohn. *Succeeding with Agile: Software Development Using Scrum*. Addison-Wesley, Upper Saddle River, NJ, 2010.

[CT82] William S. Cleveland and Irma J. Terpenning. Graphical methods for seasonal adjustment. *Journal of the American Statistical Association*, 77(377):52–62, 1982.

[CTB+12] Junghoon Chae, Dennis Thom, Harald Bosch, Yun Jang, Ross Maciejewski, David S. Ebert, and Thomas Ertl. Spatiotemporal social media analytics for abnormal event detection and examination using seasonal-trend decomposition. In *Proceedings IEEE Conference on Visual Analytics Science and Technology, VAST*, pages 143–152, 2012.

[Das13] Tamraparni Dasu. Data glitches: Monsters in your data. In Shazia Sadiq, editor, *Handbook of Data Quality*, pages 163–178. Springer Berlin Heidelberg, 2013.

[DGH06] Jan G. De Gooijer and Rob J. Hyndman. 25 years of time series forecasting. *International Journal of Forecasting*, 22(3):443–473, 2006.

[EHR+14] Alex Endert, M. Shahriar Hossain, Naren Ramakrishnan, Chris North, Patrick Fiaux, and Christopher Andrews. The human is the loop: new directions for visual analytics. *Journal of Intelligent Information Systems*, 43(3):411–435, 2014.

[ERT+17] Alex Endert, William Ribarsky, Cagatay Turkay, William Wong, Ian Nabney, Ignacio Díaz Blanco, and Fabrice Rossi. The state of the art in integrating machine learning into visual analytics. *Computer Graphics Forum*, 36(8):458–486, 2017.

[Few08] Stephen Few. Time on the horizon. Visual Business Intelligence Newsletter, `http://www.perceptualedge.com/articles/visual_business_intelligence/time_on_the_horizon.pdf`, cited Mar. 20, 2016, (last visited on Oct. 13, 2020), June 2008.

[FGR05] Peter Filzmoser, Robert G. Garrett, and Clemens Reimann. Multivariate outlier detection in exploration geochemistry. *Computers & Geosciences*, 31(5):579–587, 2005.

[FHR+14] Paolo Federico, Stephan Hoffmann, Alexander Rind, Wolfgang Aigner, and Silvia Miksch. Qualizon Graphs: Space-efficient time-series visualization with qualitative abstractions. In *Proceedings of the 12th International Working Conference on Advanced Visual Interfaces, AVI*, pages 273–280. ACM, 2014.

[FRGTA14] Peter Filzmoser, Anne Ruiz-Gazen, and Christine Thomas-Agnan. Identification of local multivariate outliers. *Statistical Papers*, 55(1):29–47, 2014.

[Fun36] Howard Gray Funkhouser. A note on a tenth century graph. *Osiris*, 1:260–262, 1936.

[Fun37] Howard Gray Funkhouser. Historical development of the graphical representation of statistical data. *Osiris*, 3:269–404, 1937.

[GAK+11] Theresia Gschwandtner, Wolfgang Aigner, Katharina Kaiser, Silvia Miksch, and Andreas Seyfang. CareCruiser: Exploring and visualizing plans, events, and effects interactively. In *Proceedings of the 4th IEEE Pacific Visualization Symposium, PacificVis*, pages 43–50, 2011.

[GBFM16] Theresia Gschwandtner, Markus Bögl, Paolo Federico, and Silvia Miksch. Visual encodings of temporal uncertainty: A comparative user study. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):539–548, 2016.

[GH07] Andrew Gelman and Jennifer Hill. *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Cambridge University Press, New York, USA, 2007.

[GM13] Ryan Greenaway-McGrevy. A multivariate approach to seasonal adjustment. Technical report, Bureau of Economic Analysis, 2013.

[GMPn99] Víctor Gómez, Agustín Maravall, and Daniel Peña. Missing observations in ARIMA models: Skipping approach versus additive outlier approach. *Journal of Econometrics*, 88(2):341–363, 1999.

[HA04] Victoria J. Hodge and Jim Austin. A survey of outlier detection methodologies. *Artificial Intelligence Review*, 22(2):85–126, 2004.

[HA18] Rob J. Hyndman and George Athanasopoulos. *Forecasting: principles and practice*. OTexts, Melbourne, Australia, 2nd edition, 2018. `https://otexts.com/fpp2` (last visited on Oct. 12, 2020).

[Ham94] James D. Hamilton. *Time series analysis*. Princeton University Press, Princeton, NJ, USA, 1994.

[HB03] Mark Harrower and Cynthia Brewer. Colorbrewer.org: An online tool for selecting colour schemes for maps. *Cartographic Journal, The*, 40(1):27–37, 2003.

[HCL05] Jeffrey Heer, Stuart K. Card, and James A. Landay. Prefuse: A toolkit for interactive information visualization. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '05)*, pages 421–430. ACM, 2005.

[HJM+11] Ming C. Hao, Halldór Janetzko, Sebastian Mittelstädt, William Hill, Umeshwar Dayal, Daniel A. Keim, Manish Marwah, and Ratnesh K. Sharma. A visual analytics approach for peak-preserving prediction of large seasonal time series. *Computer Graphics Forum*, 30(3):691–700, 2011.

[HJS+09] Ming C. Hao, Halldór Janetzko, Ratnesh K. Sharma, Umeshwar Dayal, Daniel A. Keim, and Malu Castellanos. Poster: Visual prediction of time series. In *Poster presented at IEEE Symposium on Visual Analytics Science and Technology, 2009. VAST 2009*, pages 229–230, 2009.

[HK07] Nicholas Horton and Ken Kleinman. Much ado about nothing: A comparison of missing data methods and software to fit incomplete data regression models. *The American Statistician*, 61(1):79–90, 2007.

[HK10] James Honaker and Gary King. What to do about missing values in time-series cross-section data. *American Journal of Political Science*, 54(2):561–581, 2010.

[HKB11] James Honaker, Gary King, and Matthew Blackwell. Amelia II: A program for missing data. *Journal of Statistical Software*, 45(7):1–47, 2011.

[HKF16] Steve Haroz, Robert Kosara, and Steve Franconeri. The connected scatterplot for presenting paired time series. *IEEE Transactions on Visualization and Computer Graphics*, 22(9):2174–2186, 2016.

[HKPC19] Fred Hohman, Minsuk Kahng, Robert Pienta, and Duen Horng Chau. Visual analytics in deep learning: An interrogative survey for the next frontiers. *IEEE Transactions on Visualization and Computer Graphics*, 25(8):2674–2693, 2019.

[HN98] Jerry L. Hintze and Ray D. Nelson. Violin plots: A box plot-density trace synergism. *The American Statistician*, 52(2):181–184, May 1998.

[HR07] Jeffrey Heer and George Robertson. Animated transitions in statistical data graphics. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1240–1247, 2007.

[HS12] Jeffrey Heer and Ben Shneiderman. Interactive dynamics for visual analysis. *ACM Queue*, 10(2):30:30–30:55, 2012.

[HSA+10] Christophe Hurter, Mathieu Serrurier, Roland Alonso, Gilles Tabart, and Jean-Luc Vinot. An automatic generation of schematic maps to display flight routes for air traffic controllers: Structure and color optimization. In *Proceedings of the International Conference on Advanced Visual Interfaces (AVI'10)*, pages 233–240, 2010.

[HXG02] S. L Ho, Min Xie, and Thong N. Goh. A comparative study of neural network and Box-Jenkins ARIMA modeling in time series prediction. *Computers & Industrial Engineering*, 42(2):371–375, 2002.

[IHS13] IHS Global, Inc. *EViews, Version 8*. Irvine, CA, 2013. `http://www.eviews.com` (last visited on Oct. 12, 2020).

[IIC+13] Tobias Isenberg, Petra Isenberg, Jian Chen, Michael Sedlmair, and Thorsten Möller. A systematic review on the practice of evaluating visualization. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2818–2827, 2013.

[JC14] Alan D. Jassby and James E. Cloern. *wq: Exploring water quality monitoring data*, 2014. R package version 0.4-1.

[JLC19] Liu Jiang, Shixia Liu, and Changjian Chen. Recent research advances on interactive machine learning. *Journal of Visualization*, 22(2):401–417, 2019.

[JME10] Waqas Javed, Bryan McDonnel, and Niklas Elmqvist. Graphical perception of multiple time series. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):927–934, 2010.

[Jon80] Richard H. Jones. Maximum likelihood fitting of arma models to time series with missing observations. *Technometrics*, 22(3):389–395, 1980.

[Jon85] Richard H. Jones. Time series analysis with unequally spaced data. In *Handbook of statistics*, volume 5, pages 157–178. Elsevier, Amsterdam, Netherlands, 1985.

[JSMK14] Halldór Janetzko, Florian Stoffel, Sebastian Mittelstädt, and Daniel A. Keim. Anomaly detection for visual analytics of power consumption data. *Computers & Graphics*, 38:27–37, 2014.

[KAF+08] Daniel Keim, Gennady Andrienko, Jean-Daniel Fekete, Carsten Görg, Jörn Kohlhammer, and Guy Melançon. Visual analytics: Definition, process, and challenges. In Andreas Kerren, John T. Stasko, Jean-Daniel Fekete, and Chris North, editors, *Information Visualization: Human-Centered Issues and Perspectives*, Lecture Notes in Computer Science, pages 154–175. Springer, 2008.

[KBK11] Miloš Krstajić, Enrico Bertini, and Daniel A. Keim. CloudLines: Compact display of event episodes in multiple time-series. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2432–2439, 2011.

[KHP+11] Sean Kandel, Jeffrey Heer, Catherine Plaisant, Jessie Kennedy, Frank van Ham, Nathalie Henry Riche, Chris Weaver, Bongshin Lee, Dominique Brodbeck, and Paolo Buono. Research directions in data wrangling: Visualizations and transformations for usable and credible data. *Information Visualization*, 10(4):271–288, 2011.

[KJL14] Andreas Kerren, Ilir Jusufi, and Jiayi Liu. Multi-scale trend visualization of long-term temperature data sets. In *Proceedings of SIGRAD 2014 - Visual Computing*, 2014.

[KKA95] Daniel A. Keim, Hans-Peter Kriegel, and Mihael Ankerst. Recursive pattern: A technique for visualizing very large amounts of data. In *Proceedings Visualization '95 (Vis95)*, pages 279–286, 1995.

[KKEM10] Daniel A. Keim, Jörn Kohlhammer, Geoffrey Ellis, and Florian Mansmann. *Mastering The Information Age - Solving Problems with Visual Analytics*. Eurographics Association, Goslar, Germany, 2010.

[KM12] Alexander Kowarik and Angelika Meraner. *x12: X12 - wrapper function and structure for batch processing*, 2012. R package version 1.0-3. `http://CRAN.R-project.org/package=x12` (last visited on Oct. 12, 2020).

[KMS+08] Daniel A. Keim, Florian Mansmann, Jörn Schneidewind, Jim Thomas, and Hartmut Ziegler. Visual analytics: Scope and challenges. In Simeon Simoff, Michael H. Boehlen, and Arturas Mazeika, editors, *Visual Data Mining: Theory, Techniques and Tools for Visual Analytics*, Lecture Notes in Computer Science (LNCS). Springer, Berlin, Heidelberg, 2008.

[KMST12] Alexander Kowarik, Angelika Meraner, Daniel Schopfhauser, and Matthias Templ. Interactive adjustment and outlier detection of time dependent data in R. In *Conference of European Statisticians, Work Session on Statistical Data Editing*, Oslo, Norway, 2012. United Nations - Economic Commission for Europe.

[KMT09] Daniel A. Keim, Florian Mansmann, and Jim Thomas. Visual analytics: how much visualization and how much analytics? *SIGKDD Explorations*, 11(2):5–8, 2009.

[KPB16] Josua Krause, Adam Perer, and Enrico Bertini. Using visual analytics to interpret predictive machine learning models. In *ICML Workshop on Human Interpretability in Machine Learning (WHI2016)*, New York, NY, USA, 2016.

[LAB+09] Tim Lammarsch, Wolfgang Aigner, Alessio Bertone, Johannes Gärtner, Eva Mayr, Silvia Miksch, and Michael Smuc. Hierarchical temporal patterns and interactive aggregated views for pixel-based visualizations. In *Proceedings of the 13th International Conference on Information Visualisation, IV 2009*, pages 44–49, Los Alamitos, CA, USA, 2009. IEEE.

[LAB+11] Tim Lammarsch, Wolfgang Aigner, Alessio Bertone, Silvia Miksch, and Alexander Rind. Towards a concept how the structure of time can support the visual analytics process. In S. Miksch and G. Santucci, editors, *Proceedings of the 2nd International EuroVis Workshop on Visual Analytics, EuroVA@EuroVis 2011, Bergen, Norway, May 31 - June 3, 2011*, pages 9–12, Bergen, Norway, 2011.

[LAB+13] Tim Lammarsch, Wolfgang Aigner, Alessio Bertone, Markus Bögl, Theresia Gschwandtner, Silvia Miksch, and Alexander Rind. Interactive visual transformation for symbolic representation of time-oriented data. In *Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data - Third International Workshop, HCI-KDD 2013, Held at SouthCHI 2013, Maribor, Slovenia, July 1-3, 2013. Proceedings*, volume 7947 of *Lecture Notes in Computer Science*, pages 400–419. Springer, 2013.

[Lam79] Johann Heinrich Lambert. *Pyrometrie*. Berlin, 1779.

[LB78] Greta M. Ljung and George E. P. Box. On a measure of lack of fit in time series models. *Biometrika*, 65(2):297–303, 1978.

[LBI+12] Heidi Lam, Enrico Bertini, Petra Isenberg, Catherine Plaisant, and Sheelagh Carpendale. Empirical studies in information visualization: Seven scenarios. *IEEE Transactions on Visualization and Computer Graphics*, 18(9):1520–1536, 2012.

[LGH+17] Yafeng Lu, Rolando Garcia, Brett Hansen, Michael Gleicher, and Ross Maciejewski. The state-of-the-art in predictive visual analytics. *Computer Graphics Forum*, 36(3):539–562, 2017.

[LOK13] Youn-Hee Lim, Il-Sang Ohn, and Ho Kim. *HEAT: Health Effects of Air Pollution and Temperature (HEAT)*, 2013. R package version 1.2.

[LR02] Roderick Little and Donald Rubin. *Statistical Analysis with Missing Data*. John Wiley & Sons, Hoboken, USA, 2nd edition, 2002.

[LTM18] Heidi Lam, Melanie Tory, and Tamara Munzner. Bridging from goals to tasks with design study analysis reports. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):435–445, 2018.

[LYK+12] Dongning Luo, Jing Yang, Milos Krstajic, William Ribarsky, and Daniel Keim. EvenRriver: Visually exploring text collections with temporal references. *IEEE Transactions on Visualization and Computer Graphics*, 18(1):93–105, 2012.

[MA14] Silvia Miksch and Wolfgang Aigner. A matter of time: Applying a data–users–tasks design triangle to visual analytics of time-oriented data. *Computers & Graphics*, 38:286–290, 2014.

[Mac86] Jock Mackinlay. Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics*, 5(2):110–141, 1986.

[Mah36] Prasanta Chandra Mahalanobis. On the generalized distance in statistics. *Proceedings of the National Institute of Sciences (Calcutta)*, 2:49–55, 1936.

[MHR+11] Ross Maciejewski, Ryan Hafen, Stephen Rudolph, Stephen G. Larew, Michael A. Mitchell, William S. Cleveland, and David S. Ebert. Forecasting hotspots - a predictive analytics approach. *IEEE Transactions on Visualization and Computer Graphics*, 17(4):440–453, 2011.

[Mor77] M. J. Morris. Forecasting the sunspot cycle. *Journal of the Royal Statistical Society. Series A (General)*, 140(4):437–468, 1977.

[Mun08] Tamara Munzner. Process and pitfalls in writing information visualization research papers. In A. Kerren, J.T. Stasko, J.-D. Fekete, and C. North, editors, *Information Visualization: Human-Centered Issues and Perspectives*, volume 4950 of *LNCS*, pages 134–153, Heidelberg, 2008. Springer.

[Mun09] Tamara Munzner. A nested model for visualization design and validation. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):921–928, 2009.

[MV15] Ulrich Michels and Günter Vogel. *dtv-Atlas Musik: systematischer Teil: Musikgeschichte von den Anfängen bis zur Gegenwart*. Deutscher Taschenbuch Verlag; Bärenreiter, 2015. 4. Auflage.

[Naz88] Sufi M. Nazem. *Applied Time Series Analysis for Business and Economic Forecasting*. Statistics: Textbooks and Monographs. Marcel Dekker, Inc., NY, USA, 1988.

[Pfa08] Bernhard Pfaff. *Analysis of Integrated and Cointegrated Time Series with R*. Use R! Springer, New York, NY, USA, 2008.

[PKRJ10] Kristin Potter, Joe Kniss, Richard Riesenfeld, and Chris R. Johnson. Visualizing summary statistics and uncertainty. *Computer Graphics Forum (Proceedings of Eurovis 2010)*, 29(3):823–832, 2010.

[Pla86] William Playfair. *The Commercial and Political Atlas and Statistical Breviary*. edited and introduced by Wainer H. and Spence I. Cambridge University Press, 2005 edition, 1786.

[Pla01] William Playfair. *The commercial and political atlas : representing, by means of stained copper-plate charts, the progress of the commerce, revenues, expenditure and debts of England during the whole of the eighteenth century*. London: Printed by T. Burton for J. Wallis, etc, 3rd ed. edition, 1801.

[PnP01] Daniel Peña and Francisco J. Prieto. Multivariate outlier detection and robust covariance matrix estimation. *Technometrics*, 43(3):286–310, 2001.

[Pot06] Kristin Potter. Methods for presenting statistical information: The box plot. *Visualization of Large and Unstructured Data Sets, GI-Edition Lecture Notes in Informatics (LNI)*, S-4:97–106, 2006.

[PW04] Roger D. Peng and Leah J. Welty. The NMMAPSdata package. *R News*, 4(2):10–14, 2004.

[R C20] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2020.

[Rei08] Hannes Reijner. The development of the horizon graph. In *Proceedings of the VisWeek Workshop From Theory to Practice: Design, Vision and Visualization*, 2008.

[RG10] Daniel Rosenberg and Anthony Grafton. *Cartographies of Time: A History of the Timeline*. Princeton Architectural Press, 2010. Google-Books-ID: DqWqKVzipToC.

[RLA+13] Alexander Rind, Tim Lammarsch, Wolfgang Aigner, Bilal Alsallakh, and Silvia Miksch. TimeBench: A data model and software library for visual analytics of time-oriented data. *IEEE Transactions on Visualization and Computer Graphics*, 19:2247–2256, 2013.

[RLK+15] Martin Röhlig, Martin Luboschik, Frank Krüger, Thomas Kirste, Heidrun Schumann, Markus Bögl, Bilal Alsallakh, and Silvia Miksch. Supporting activity recognition by visual analytics. In *Proceedings of the 2015 IEEE Conference on Visual Analytics Science and Technology, VAST 2015, Chicago, IL, USA, October 25-30, 2015*, pages 41–48. IEEE, 2015.

[RLS+14] Martin Röhlig, Martin Luboschik, Heidrun Schumann, Markus Bögl, Bilal Alsallakh, and Silvia Miksch. Analyzing parameter influence on time-series segmentation and labeling. In *Proceedings of the 2014 IEEE Conference on Visual Analytics Science and Technology, VAST – Posters, Paris, France, October 25-31, 2014*, pages 269–270. IEEE, 2014.

[Rob07] Jonathan C. Roberts. State of the art: Coordinated & multiple views in exploratory visualization. In *Proceedings of the Conference on Coordinated and Multiple Views in Exploratory Visualization (CMV)*, pages 61–71. IEEE, 2007.

[Rou85] Peter J. Rousseeuw. Multivariate estimation with high breakdown point. *Mathematical Statistics and Applications*, 8:283–297, 1985.

[S+08] C. J. Sprengell et al. *Aphorisms of Hippocrates and the Sentences of Celsus: With Explanations and References to the Most Considerable Writers in Physick and Philosophy, Both Ancient and Modern: to which are Added, Aphorisms Upon the Small-pox, Measles, and Other Distempers, Not So Well Known to Former More Temperate Ages*. R. Bonwick, W. Freeman, T. Goodwin, and others, 1708.

[Sad13] Shazia Sadiq. Prologue: Research and practice in data quality management. In Shazia Sadiq, editor, *Handbook of Data Quality*, pages 1–11. Springer Berlin Heidelberg, 2013.

[SAS12] SAS Institute Inc. *JMP 10 Modeling and Multivariate Methods*. SAS Institute Inc., Cary, NC, November 2012. `http://www.jmp.com` (last visited on Oct. 12, 2020).

[SBVLK09] Tobias Schreck, Jürgen Bernard, Tatiana Von Landesberger, and Jörn Kohlhammer. Visual cluster analysis of trajectory data with interactive Kohonen maps. *Information Visualization InfoVis*, 8(1):14–29, 2009.

[Sch99] Joseph Schafer. Multiple imputation: a primer. *Statistical Methods in Medical Research*, 8(1):3–15, 1999.

[SDZ+00] Jonathan Samet, Francesca Dominici, Scott Zeger, Joel Schwartz, and Douglas Dockery. The national morbidity, mortality, and air pollution study. Part I: Methods and methodologic issues. Research Report 94, Health Effects Institute, Cambridge, 2000.

[SFdOL04] Milton H. Shimabukuro, Edilson F. Flores, Maria Christina F. de Oliveira, and Haim Levkowitz. Coordinated views to assist exploration of spatio-temporal data: A case study. In *Proceedings of the 2nd International Conference on Coordinated and Multiple Views in Exploratory Visualization, CMV04*, pages 107–117. IEEE, 2004.

[Shn83] Ben Shneiderman. Direct manipulation: A step beyond programming languages. *IEEE Computer*, 16(8):57–69, 1983.

[Shn96] Ben Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings 1996 IEEE Symposium on Visual Languages*, pages 336–343, Boulder, Colorado, USA, 1996. IEEE, IEEE.

[SMM12] Michael Sedlmair, Miriah Meyer, and Tamara Munzner. Design study methodology: Reflections from the trenches and the stacks. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2431–2440, 2012.

[SMY+05] Takafumi Saito, Hiroko Nakamura Miyamura, Mitsuyoshi Yamamoto, Hiroki Saito, Yuka Hoshiya, and Takumi Kaseda. Two-tone pseudo coloring: Compact visualization for one-dimensional data. In *IEEE Symposium on Information Visualization, InfoVis*, pages 173–180. IEEE, 2005.

[SNHS13] Hans-Jörg Schulz, Thomas Nocke, Magnus Heitzler, and Heidrun Schumann. A design space of visualization tasks. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2366–2375, 2013.

[SS11] Robert H. Shumway and David S. Stoffer. *Time Series Analysis and its Applications. With R examples.* Springer, New York, 3rd edition, 2011.

[SSS+14] Dominik Sacha, Andreas Stoffel, Florian Stoffel, Bum Chul Kwon, Geoffrey Ellis, and Daniel A. Keim. Knowledge generation model for visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1604–1613, 2014.

[Sta11] StataCorp. *Stata Statistical Software: Release 12*. StataCorp LP, College Station, TX, 2011. `http://www.stata.com` (last visited on Oct. 12, 2020).

[STA+13] Daniel Schopfhauser, Matthias Templ, Andreas Alfons, Alexander Kowarik, and Bernd Prantner. *VIMGUI: Visualization and Imputation of Missing Values*, 2013. R package version 0.9.0.

[TAKP13] Matthias Templ, Andreas Alfons, Alexander Kowarik, and Bernd Prantner. *VIM: Visualization and Imputation of Missing Values*, 2013. R package version 4.0.0.

[TAS04] Christian Tominski, James Abello, and Heidrun Schumann. Axes-based visualizations with radial layouts. In *Proceedings of the 2004 ACM symposium on Applied computing SAC'04*, pages 1242–1247. ACM, 2004.

[TC05] James J. Thomas and Kristin A. Cook. *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. IEEE, Los Alamitos, CA, USA, 2005.

[The10] The MathWorks Inc. *MATLAB (R2009a)*. Natick, MA, 2010. `http://www.mathworks.com` (last visited on Oct. 12, 2020).

[Til75] Laura Tilling. Early experimental graphs. *The British Journal for the History of Science*, 8(3):193–213, 1975.

[Tsa00] Ruey S. Tsay. Time series and forecasting: Brief history and future research. *Journal of the American Statistical Association*, 95(450):638–643, 2000.

[Tsa10] Ruey S. Tsay. *Analysis of Financial Time Series*. Wiley Series in Probability and Statistics. John Wiley & Sons, 3rd edition edition, 2010.

[Tuf83] Edward R. Tufte. *The Visual Display of Quantitative Information*. Graphics Press, Cheshire, CT, USA, 1983.

[Tuf06] Edward R. Tufte. *Beautiful Evidence*. Graphics Press, Cheshire, CT, USA, 2006.

[Tuk77] John W. Tukey. *Exploratory Data Analysis*. Addison-Wesley, 1977.

[Urb20] Simon Urbanek. rJava: Low-level R to java interface, 2020. R package version 0.9-13. `http://CRAN.R-project.org/package=rJava` (last visited on Oct. 12, 2020).

[vB12] Stef van Burren. *Flexible Imputation of Missing Data*. Chapman and Hall/CRC, Boca Raton, USA, 2012.

[vB14] Stef van Buuren. Multiple imputation. Online, 2014. [accessed Feb. 18, 2014] Available online at `http://www.multiple-imputation.com`, Update of Appendix A from [vB12]; [Update from Oct. 13, 2020: Website not accessible anymore. New website of the author `https://stefvanbuuren.name/` (last visited on Oct. 13, 2020); New edition of [vB12] available [vB18]].

[vB18] Stef van Buuren. *Flexible Imputation of Missing Data*. Chapman and Hall/CRC, 2nd edition, 2018.

[vW06] Jarke J. van Wijk. Views on visualization. *IEEE Transactions on Visualization and Computer Graphics*, 12(4):421–432, 2006.

[vWvS99] Jarke J. van Wijk and Edward R. van Selow. Cluster and calendar based visualization of time series data. In *Proceedings 1999 IEEE Symposium on Information Visualization (InfoVis'99)*, pages 4–9, San Francisco, CA, USA, 1999. IEEE.

[War00] Colin Ware. *Information Visualization - Perception for design*. Morgan Kaufmann Publishers Inc., 1st edition, 2000.

[Wol13] Wolfram Research, Inc. *Mathematica, Version 9.0.1*. Champaign, IL, 2013. `http://www.wolfram.com/mathematica` (last visited on Oct. 12, 2020).

[WWS+16] Tongshuang Wu, Yingcai Wu, Conglei Shi, Huamin Qu, and Weiwei Cui. PieceStack: Toward better understanding of stacked graphs. *IEEE Transactions on Visualization and Computer Graphics*, 22(6):1640–1651, 2016.

[Yul27] George Udny Yule. VII. on a method of investigating periodicities disturbed series, with special reference to wolfer's sunspot numbers. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 226(636):267–298, 1927.