Exploring correlated data: confidence bands and projections of shared variation

Jussi Korpela
Exploring correlated data: confidence bands and projections of shared variation

Jussi Korpela

A doctoral dissertation completed for the degree of Doctor of Science (Technology) to be defended, with the permission of the Aalto University School of Science, at a public examination held at the lecture hall T1 of the Computer Science building of the school on 15th March 2018 at 12 noon.

Aalto University
School of Science
Department of Computer Science
Supervising professor
Professor Kai Puolamäki, Aalto University, Finland

Thesis advisor
Professor Aristides Gionis, Aalto University, Finland

Preliminary examiners
Associate Professor Meelis Kull, University of Tartu, Estonia
Dr. Salvatore Rinzivillo, National Research Council of Italy, Italy

Opponent
Professor Mykola Pechenizkiy, Eindhoven University of Technology, The Netherlands
Abstract

The steady increase in automatic data collection and analysis creates new possibilities for data-driven decision making. Consequently there is a need for the development of new explorative data analysis methods. This thesis deals with two such methods: multivariate confidence intervals and finding shared variation between datasets.

First, we present a method to visualize the variation of a set of vector-valued data items. The visualization is a two dimensional confidence band, whose interpretation is similar to that of a one dimensional confidence interval. The goal is to have a band that covers a predefined fraction of the probability mass of the data vector distribution, such that the band can be used to assess likely values for a typical vector. We introduce new methods to compute the bands as well as describe in more detail the technical implementations of existing methods. In addition, we present a correction procedure that adjusts the coverage properties of the band when computed from a finite sample.

The second part of the work deals with finding shared variation between datasets of a data collection. The analysis is applied to data collections that describe a certain process from multiple views, and hence the shared variation becomes a measure of the underlying process. The method can be used to find the periods during which the datasets share variation with each other.

To solve the problem, we propose a filtering approach based on ordinary regression functions. The algorithm filters away all variation that is not shared by all of the datasets. Advantages of the method include easy implementation and adaptability – by changing the regression function one can easily change the definition of shared variation to match the problem at hand.

Confidence bands have many applications in expressing the variability of time series and other vector valued data. A prime example are time series model forecasts whose modeling uncertainty is often visualized using a confidence band. Analysis of shared variation, on the other hand, is often needed in conjunction with biosignal analysis where one might be, e.g., interested in finding shared and unshared changes in signal level between test subjects.

Keywords  time series, confidence band, simultaneous confidence interval, visualization, shared variation, regression

ISSN-L 1799-4934        ISSN (printed) 1799-4934        ISSN (pdf) 1799-4942
Location of publisher Helsinki        Location of printing Helsinki        Year 2018
Tekijä
Jussi Korpela

Väitöskirjan nimi
Korrolointunutta dataa tutkimassa: luottamusnauhat ja jaettu vaihtelun kuvaaminen

Julkaisija
Perustieteiden korkeakoulun Tietotekniikan laitos

Yksikkö
Aalto University publication series DOCTORAL DISSERTATIONS 30/2018

Tutkimusala
Informaatiotekniikka

Käsikirjoituksen pvm
29.01.2018

Väitöspäivä
15.03.2018

Julkaisuluvan myöntämispäivä
03.01.2018

Kieli
Englanti

Monografia

Artikkeliväitöskirja

Esseeväitöskirja

Tiivistelmä


Työn ensimmäisessä osassa esitämme visuaalisen menetelmän vektoriarvoisen datan vaihtelun kuvaamiseen. Vaihtelu visuaalisoitetaan kaksiulotteisena luottamusnauhana, joka tulkinnaltaan vastaa yksiulotteisen datan luottamusvälillä. Tavoitteena on, että nauha peittää ennalta määrätyn osuuden vektorideiden jakaumasta, jolloin nauhaa voi käyttää havainnollistamaan vektorijakauman tyypillisää arvoja.

Työssä esitetään uusia menetelmiä luottamusnauhoido laskemiseksi sekä käydään läpi aiempaa tarkemmin kirjallisuudessa jo esitettyjen menetelmien teknisiä toteutuksia. Lisäksi ehdotetaan menettelyä, jolla nauhan peitto saadaan halutun suuruiseksi myös äärellisillä havaintomääriillä.

Työn toisessa osassa etsitään jaettua vaihtelua samaan kokoelmasta kuuluvista datamatriiseista. Sovelluskohteena ovat samaa prosessia eri näkökulmista kuvaavat datamatriisikokoelmat, joiden jaettu vaihtelu antaa epäsuraatista tietoa taustalla vaikutavasta prosessista. Menetelmä löytyi ne ajanjakso, joilla datamatriiseissa tapahtuu yhtäaikaista vaihtelua.

Ongelman ratkaisuksi ehdotetaan tavanomaisiin regressiofunktioihin perustuva suodatusmenettelyä, joka suodattaa pois kaiken sellaisen vaihtelun, joka ei ole kaikille datamatriisille yhteistä. Menetelmenä etuja ovat helppo toteuttavuus sekä moottoravuus – regressiofunktioita vaihtamalla voidaan kätevästi muuttaa jaetuun vaihtelun määrittelmää kulloiseenkin tilanteeseen sopivaksi.

Luottamusnauholle on käytööä erityisesti aikasarjojen sekä muun vektoriarvoisen datan vaihtelun kuvaamisessa. Tyypillinen esimerkki ovat aikasarjamaallien tuottamat ennusteet, joiden epävarmuus havainnollistetaan usein käyttäen luottamusnauhanoja. Datamatriisien välästä jaettua vaihtelua puolestaan tutkitaan usein biosignaalien analysin yhteydessä, esimerkiksi selvitettäessä sitä, mitkä aktiivaatiot ovat tai eivät ole jollekin koehenkilöryhmälle yhteisiä.

Avainsanat
aikasarja, luottamusnauha, luottamusväli, visualisointi, jaettu vaihtelu, regressio

ISBN (painettu) 978-952-60-7854-0
ISBN (pdf) 978-952-60-7855-7
ISSN-L 1799-4934
ISSN (painettu) 1799-4934
ISSN (pdf) 1799-4942
Julkaisupäivä Helsinkin Painopäivä Helsinkin Vuosi 2018
Sivumäärä 188
Preface

This thesis is best categorized as applied computer science and the research themes have a close connection to the everyday problems I have encountered in my daily work of analyzing physiological signals. It has been put together while working as a research engineer at the Finnish Institute of Occupational Health (FIOH) in projects funded by Tekes (SalWe Research Program for Mind and Body, and The Revolution of Knowledge Work). I wish to thank FIOH for the opportunity to do research while being employed full-time. Especially my senior colleagues Prof. Dr. Kai Puolamäki, Dr. Kiti Müller, and Dr. Minna Huotilainen have significantly contributed to this possibility by successfully applying funding that allows method development.

Having a good instructor, preferably just a few rooms down the corridor, is a privilege. Accordingly, I am grateful to Kai Puolamäki for acting as the supervising professor for this thesis. Kai has provided computer science related expertise and connections that have greatly paved the way. In addition, I would like to thank Prof. Dr. Aristides Gionis for acting as the thesis advisor, for co-authorship and for the numerous constructive comments regarding the manuscript. I would also like to thank the pre-examiners Prof. Dr. Meelis Kull and Dr. Salvatore Rinzivillo whose detailed and rigorous comments where of great help in shaping the work.

This work would not have been possible without the co-authors either: Dr. Arto Klami, Dr. Antti Ukkonen, Dr. Emilia Oikarinen, Lauri Ahonen and Dr. Andreas Henelius – thank you. Special thanks to Andreas Henelius for all the peer-to-peer support and the nice conversations we have had over the years.

Furthermore I would like to thank all my past and present colleagues at FIOH – thank you for creating such a relaxed, innovative and productive atmosphere to work in. I would like to thank Teppo Valtonen, Kati Pet-
tersson, Jussi Virkkala, Kristian Lukander, Satu Pakarinen, Laura Sokka, Riku Louhimo, Aki Koskinen, Matti Gröhn, Marianne Leinikka, Miika Toivanen, Sharman Jagadeesan and Benjamin Cowley.

Finally, my warmest thanks to my parents Pirkko and Timo for all the support. Above all, I am grateful to my wife Elisa, who has always been willing to organize child care and everyday life in favor of the thesis, and to our children, Maisa and Martti, who have effectively prevented me from sinking too deep into this scientific endeavor.

Espoo, January 31, 2018,

Jussi Korpela
Contents

Preface 1
List of Publications 7
Author's Contribution 9

1. Notation 11

2. Introduction 13
   2.1 Multivariate Confidence Intervals 15
      2.1.1 Confidence Bands and Coverage 16
      2.1.2 Applications of Confidence Bands 18
      2.1.3 Confidence Bands and Multiple Hypothesis Significance Testing 18
      2.1.4 Current Methods 20
      2.1.5 Contribution 21
   2.2 Shared Variation Between Datasets 22
      2.2.1 Applications 23
      2.2.2 Current Methods 24
      2.2.3 Contribution 24
   2.3 Summaries of Publications 25
      2.3.1 Publication I 25
      2.3.2 Publication II 26
      2.3.3 Publication III 26
      2.3.4 Publication IV 27
   2.4 Outline of the Thesis 27

3. Multivariate Confidence Bands in Two Dimensions 29
   3.1 Statistical Intervals and Their Definitions 29
   3.2 Relation to Density Estimation 32
   3.3 Applications of Confidence Bands 34
   3.4 Properties of Confidence Bands 36
   3.5 Multiple Hypothesis Significance Testing (MHST) 38
### Preface

3.6 Coverage ................................................. 40
3.7 Methods for Computing Confidence Bands ................. 43
   3.7.1 Naïve Quantiles ................................. 44
   3.7.2 Iterative Quantiles .............................. 45
   3.7.3 Bonferroni Quantiles ............................. 45
   3.7.4 Inverting a Multiple Hypothesis Testing Procedure 46
   3.7.5 Multivariate Distance Measures .................... 46
   3.7.6 Curve Distances ................................ 47
   3.7.7 Rank Based Envelopes ............................. 49
   3.7.8 Minimum Width Envelope .......................... 49
3.8 Controlling Expected Coverage ................................ 52
   3.8.1 Cross Validation Procedure ....................... 55

4. Applications of Confidence Bands .................. 59
   4.1 Finding Structure in Interval Sequences ............. 59
      4.1.1 Randomization Methods ......................... 59
      4.1.2 Confidence Intervals and Hypothesis Testing .. 61
      4.1.3 Example: Words in a Book ....................... 61
   4.2 Analyzing Single-Trial ERPs .......................... 62

5. Shared Variation Between Datasets .............. 65
   5.1 Multi-View Learning ................................ 65
      5.1.1 Co-Training .................................. 66
      5.1.2 Multiple Kernel Methods ....................... 67
      5.1.3 Subspace Learning .............................. 67
   5.2 Notation and the General Shared Variation Problem ... 69
      5.2.1 Shared Variation ................................ 69
   5.3 Maximally Dependent Projections .................... 71
      5.3.1 Canonical Correlation Analysis ................ 71
      5.3.2 Multi-Set CCA .................................. 72
   5.4 Variance Factorization ................................ 72
      5.4.1 Restricting the Model .......................... 73
      5.4.2 Group Factor Analysis .......................... 74
   5.5 Regression Chain Filtering (COCOREG) ............. 75
      5.5.1 Unidentifiability of Shared Variation ........ 76
      5.5.2 Problem Definition ............................. 76
      5.5.3 Regression Chains .............................. 77
      5.5.4 COCOREG Algorithm ............................. 79
      5.5.5 Evaluation ...................................... 80
This thesis consists of an overview and of the following publications which are referred to in the text by their Roman numerals.


Author’s Contribution

Publication I: “Confidence bands for time series data”

The concept and basic approach was developed jointly by all authors. All algorithms, experiments and analyses were made by the present author.

Publication II: “Multivariate Confidence Intervals”

The concept of relaxing the within-band criterion was developed jointly by all authors. Antti Ukkonen was responsible for the complexity and approximability parts, while the present author implemented the greedy algorithm, prepared the stock market example as well as Figures 1 and 5. All authors took part in the writing.

Publication III: “Using regression makes extraction of shared variation in multiple datasets easy”

The concept of using regression to filter out shared variation was proposed by Kai Puolamäki. The implementation of the COCOREG framework was done jointly by Andreas Henelius and the present author. All experiments, analysis and writing were done by the present author.

Publication IV: “Explaining Interval Sequences by Randomization”

The use of randomization to generate data for statistical testing purposes has a long tradition, while the specific application to interval sequence data was a joint idea of all authors. The present author implemented all
randomization methods except the one related to MCMC sampling. He also prepared all experiments and visualizations related to the book words data. All authors took equal part in writing. This publication has been previously used in the doctoral thesis of Andreas Henelius, who made all fixed distance function randomizations using MCMC methods and the related experiments (Henelius, 2017). The inclusion of this publication to two theses is warranted as the contributions of Andreas Henelius and Jussi Korpela are non-overlapping.
1. Notation

Notation related to confidence bands:

\( F \) \hspace{1cm} \text{Any multivariate distribution}

\( \hat{F} \) \hspace{1cm} \text{Any empirical multivariate distribution}

\( X \) \hspace{1cm} \text{Observed data, } X \in \mathbb{R}^{N \times M}

\( x \) \hspace{1cm} \text{Row of } X, \ x \in \mathbb{R}^{M}

\( N \) \hspace{1cm} \text{Number of observations}

\( n \) \hspace{1cm} \text{Index over observations / curves, i.e., rows of } X

\( M \) \hspace{1cm} \text{Number of dimensions}

\( m \) \hspace{1cm} \text{Index over dimensions, i.e., columns of } X

\( \alpha \) \hspace{1cm} \text{Significance level}

\( \gamma \) \hspace{1cm} \text{Coverage}

\( \gamma_s \) \hspace{1cm} \text{In-sample coverage, set to } \gamma_s = 1 - \alpha

\( \gamma_t \) \hspace{1cm} \text{True coverage observed under } F

\( \gamma_e \) \hspace{1cm} \text{Expected true coverage}

\( K \) \hspace{1cm} \text{Number of observations (curves) to remove}

\( k \) \hspace{1cm} \text{Index over } K

\( L \) \hspace{1cm} \text{Number of dimensions in which the data may exit the band (MWE problem)}

\( B \) \hspace{1cm} \text{(Confidence) band, } B = (x_f, x_u) \text{ a tuple of two vectors from } \mathbb{R}^{M}
$x_l(m), x_u(m)$ Lower and upper bounds of a confidence band $B$ for all $m \in 1 : M$

$V(B)$ Size of the band (e.g., volume or area)

$\text{Err}(x|B)$ Error of curve $x$ given band $B$

$\phi_j(k)$ Coverage profile for $J$-fold cross validation

$L_p$ The $p$ norm: $||x||_p := \left( \sum_{m=1}^{M} |x(m)|^p \right)^{1/p}$

$[J]$ The set $\{1, \ldots, J\}$

NAIVE.QNT Naïve quantiles confidence bands

NAIVE.QNT.S Naïve quantiles confidence bands using S-Plus quantiles

BONF.QNT Bonferroni quantiles confidence bands

MAHA Mahalanobis distance confidence bands

MI Minimum interval confidence bands

MWE Minimum width confidence bands

MWEC Coverage corrected MWE bands

Notation related to COCOREG:

$x$ A vector of observed feature values from all views

$W$ Mixing matrix of a probabilistic model

$z$ A vector of latent variable scores

$R$ Correlation matrix

$\epsilon$ Noise

$\Psi$ Noise covariance matrix

PCA Principal component analysis

FD Factor analysis

GFA Group factor analysis

RGCCA Regularize generalized canonical correlation analysis

CCR COCOREG using all paths

CCRS COCOREG using path sampling
2. Introduction

We are living in an age of data. Data of all kinds are generated in an ever increasing pace as digitalization penetrates new businesses and areas of life. The data are often used to gain insight and this thesis proposes novel methods to extract this insight.

A large part of practical data analysis is exploratory and so are all methods in this thesis. Publications I and II introduce multivariate confidence intervals that can be used to quantify and visualize the quantiles of a multivariate probability density. These intervals find use in the rest of the thesis, which describes methods to extract the shared variation between datasets (Publication III) and methods to study interval sequences using randomization (Publication IV). The methods in this thesis are designed to be simple and understandable. This follows the principle of parsimony whereby simpler models should be favored over more complex ones when the two give comparable results. Simplicity also increases transparency which, in turn, makes the methods more acceptable to potential users in various disciplines. The proposed methods are also robust as there are no complex model structures that would lead into problems in learning and inference. In addition, all methods are straightforward to implement which also adds to their practical value. We also distribute R implementations of the methods as GitHub repositories\textsuperscript{1}. COCOREG is also available in CRAN as the package “cocoreg”.

After exploration we must ensure that the patterns we found did not occur by chance alone but that they are a real property of the data. Accordingly significance testing is a fundamental part of any meaningful data analysis. Until recently, most research has been relying on statistical tests and the P values they produce to establish the validity of the findings.\textsuperscript{1}https://github.com/jutako/multivariate-ci.git for confidence bands and https://github.com/bwrc/cocoreg-r for COCOREG
However, P values give information only of the statistical significance: it is possible to have a statistically significant result that is useless in practice due to a small effect size. For example, a small difference $\Delta d$ in the means of two distributions always becomes statistically significant, if the test is carried out with large enough samples. At the same time $\Delta d$ can be so small compared to the variances of the distributions that it has no practical value, e.g., in separating the distributions in a classification task. This problem has been long and acutely recognized, e.g., in medical research (Gardner and Altman, 1986). Some psychology journals have recently even banned the use of P values (Trafimow and Marks, 2015; Woolston, 2015). It has been suggested that the primary reported quantities should be confidence intervals and effect sizes rather than a test statistic and P values (Nuzzo, 2014). In this context confidence intervals can be seen as a general term that refers to any interval the purpose of which is to describe the probable values of some quantity. These include the confidence intervals of a statistical parameter, the tolerance intervals of some distribution, quantiles and percentiles as well as the credible intervals of Bayesian analyses – to name the most prominent ones.

The Publications I and II of this thesis consider non-parametric multivariate extensions of these intervals, which we call confidence bands. Intervals are practical since they describe the location and variance of the estimate at the same time. By comparing the interval to some baseline level or to another interval estimates of effect size can be made, both graphically and quantitatively. Provided that the intervals have been constructed such that they are guaranteed to cover some specific fraction of the value space they can also be used to perform statistical testing. There is extensive literature concerning the univariate intervals but much less so concerning the multivariate ones. Publications I and II are thus an attempt to fix this imbalance. We will introduce non-parametric methods to create the multivariate intervals, test their properties and tune them to data that may have spurious outliers.

Lastly, it should be noted that all methods in this thesis are inspired by time series problems and are applicable to time series data. The confidence bands are a perfect match for the way time series are plotted against time and the shared variation extraction is a way of detecting simultaneous changes in signal level in multiple sets of time series. Likewise, the interval sequences in Publication IV can be formed from time series data; one example would be the sequence of inter blink intervals to assess
vigilance. From this viewpoint the publications can be seen as a set of tools all working on the same problem.

However, in general, the methods presented in this thesis are applicable to any multivariate data with dependencies between the variates – time series just happen to be one prominent example of such data. Examples of non time series data for which the methods can be applied include, e.g., probability density estimates, signal power spectra and gait analysis (Lenhoff et al., 1999).

This thesis has two parts, both of which deal with explorative data analysis of multivariate data, but from different viewpoints. The first part deals with multivariate tolerance intervals, which are used to visualize the location of a multivariate distribution (Publications I and II). The second part introduces novel use for standard regression tools and describes how they can be used to extract shared variation from a collection of datasets (Publication III). Publication IV is discussed in less detail as an application example of part one.

2.1 Multivariate Confidence Intervals

A multivariate dataset is a dataset with multiple dimensions. Each dimension may correspond to a separate variable or there might be multiple observations of a single variable, for example over time, that can be treated as separate dimensions. Multivariate confidence intervals can be constructed for any multivariate dataset. However, they are especially suitable for data that exhibit some dependency structures between the dimensions. Time series are one abundant example of this type of data and they will be frequently used in this work to showcase the methods. A time series is a vector of real numbers in which each vector index is associated with a timestamp. Consider for example a time series of temperature data. Each entry of the series is a temperature value with an associated timestamp indicating the time of measurement. In what follows each timestamp is considered as a separate dimension: a series of length $M$ is considered to form a single point in a vector space of dimension $M$.

We will first explain some key concepts followed by motivation about why multivariate confidence intervals are useful. The chapter proceeds with discussion about current methods, gives a multiple hypothesis viewpoint on the problem and concludes with the contributions made in this thesis.
2.1.1 Confidence Bands and Coverage

A multivariate confidence interval is the generalization of a scalar confidence interval into multivariate setting. To see what this means, consider Figure 2.1. The left column shows how scalar confidence intervals are formed. A single observation (top left) gives little information on the distribution whereas a larger sample (middle left) gives a much finer grade histogram estimate of the distribution (bottom left). A 90% confidence interval (CI) for the data (bottom left) is computed by finding the shortest interval that contains 90% of the observations. Another option would have been to use the interval bounded by the 5:th and 95:th percentile as the CI. We chose to use the shortest interval since they are more consistent with the minimum width bands to be computed later. The shortest interval approximates a high density interval (see Section 3.1 as well as Liu et al. (2015) and Hyndman (1996)).

Note that our definition of a confidence interval differs slightly from that used in statistics: our intervals describe the variation of the data whereas in statistics confidence intervals quantify the uncertainty about the true value of an unobserved population parameter. Moreover, in our intervals the percentage refers to the fraction of the probability mass of the data covered, not the probability with which the interval contains the true value of an unobserved population parameter. We acknowledge that our use of the term confidence differs from the tradition of statistics and that this might cause misunderstanding, but we were unable to find a better name, see Section 3.1.

An equivalent treatment is given to time series data in the right column of Figure 2.1. A single observation (top right) is a vector of points, a curve, and a set of curves forms (middle right) an empirical distribution of the data. The last panel shows a multivariate confidence interval for the data (bottom right). As in the univariate case, the purpose is to show the whereabouts of the densest part of the distribution. The random variable is now a curve and hence a band shaped region is needed to visualize the quantiles of the distribution. Since the multivariate confidence intervals of unimodal distributions form a band shaped region we will call them shortly confidence bands.

The univariate confidence intervals are characterized by how much of the probability mass of the distribution lies within the interval. In the above example the 90% interval is wide enough to account for 90% of the
Figure 2.1. Comparison of univariate and multivariate confidence intervals using temperatures dataset from Milan, Italy. The left panel outlines the process of creating a confidence interval for univariate data (January observations), the right for vector valued data (one curve per year). The univariate confidence interval is shown as a horizontal blue bar (bottom, left) whereas a multivariate confidence intervals (blue lines, bottom right) form a band-like region. Confidence level 90 % is used throughout. The data are monthly average maximum temperatures (see Publication I).

probability mass. Similarly the confidence bands are constructed such that they contain a certain predefined percentage of the curves. This percentage is called coverage. For example, the confidence band in Figure 2.1 is constructed such that if new curves are sampled from the underlying distribution 90 % of them lie completely within the band and hence the band has a coverage of 90 %. Many authors speak of simultaneous confidence bands when entire curves are considered in assessing coverage (Akram et al., 2016; Kolsrud, 2007; Staszewska-Bystrova and Winker, 2013; Winker et al., 2016). A simpler option would be to assess coverage separately for each x-coordinate of the curves, which creates pointwise confidence bands. However, the pointwise control of the coverage is insufficient as will be explained below in Section 2.1.3.

Finding a band that has exactly the desired coverage is not straightforward. Sample size, smoothness of the data and the algorithm used to construct the bands all affect the outcome. Consequently, the main focus of the first part of this thesis will be on coverage and how to make sure that
the confidence band has the desired coverage for all datasets originating from the same distribution.

### 2.1.2 Applications of Confidence Bands

Time series is a common type of data and consequently confidence bands have many applications. One important application domain is forecasting, where a mathematical model produces forecasts and these are visualized to make them more interpretable to the end user. Time series prediction is one example of this, with applications such as impulse response functions of vector autoregressive (VAR) models (Lütkepohl et al., 2015b; Staszewska-Bystrova, 2011), autoregressive moving average (ARMA) models (Nolan and Ravishanker, 2009), fan charts in trend analysis (Britton et al., 1998) and weather forecasts.

As an example, Figure 2.2 shows how confidence bands can be used to visualize the uncertainty related to the temperature and precipitation forecasts. The bands nicely show how in general the uncertainty of the forecast increases with time. The bands augment the point estimate by providing visual cues on reliability. After seeing this kind of a plot, it should not come as a surprise that weather forecasts change frequently.

Another application domain where confidence bands are often used is regression analysis, where one might extrapolate the model into the future or study model stability with respect to the input data and model parameters. These and other applications of confidence bands are discussed in more detail in Chapter 3.

### 2.1.3 Confidence Bands and Multiple Hypothesis Significance Testing

The problem of constructing a confidence band to have a certain coverage is similar to that of the overall error rate control in multiple hypothesis significance testing (MHST). In MHST a statistical test is applied several times in sequence. Let us assume that each test makes the correct decision with 90% probability. In a sequence of, say, $M = 10$ independent tests we make the correct decision in all tests with the probability of $0.9^{10} \approx 0.35 = 35\%$. Turning things around, this means that we make at least one incorrect decision with $100\% - 35\% = 65\%$ probability. This probability, the family-wise error rate (FWER), is much higher than the 10% that applies for the individual tests. So clearly, when decisions are made in
sequence, the probability of making one or more false decisions quickly increases above the level employed in single decisions. This problem has been extensively studied in MHST literature. Several multiple hypothesis correction methods have been proposed to control the probability of making an incorrect decision over whole sequence of decisions (Dudoit et al., 2003; Westfall and Young, 1993). Some of these procedures, such as the well-known Bonferroni correction, can be inverted to produce a confidence band as shown in Chapter 3.7.3. Many step-wise procedures would have more statistical power than the Bonferroni method, but unfortunately there is no easy way of inverting these, as described in Chapter 3.7.4.

The need for multiple hypothesis correction applies directly to confidence bands as well. To see this, let us assume that we have been given a confidence band (two curves, upper and lower) and a candidate time series (one curve). The length of the time series (number of dimensions $M$) corresponds to the length of the decision sequence. A statistical test would be, e.g., to test at one time point whether the candidate time series exits the band at that point or not. Making this test on all available time points corresponds to an overall decision.
Thus far we have assumed the tests or time points to be independent. However, if the tests are somehow dependent, the FWER of the whole set changes in ways that are hard to predict. In an extreme case the tests are perfectly dependent and the FWER of the set is equal to the confidence level of a single test. For time series data this translates to a situation where knowing the value of the series at $t_0$ immediately defines the values at all subsequent time points as well. In practice most data are partially dependent and the confidence bands should adapt accordingly. For example, for time series data the bands should take the varying degrees of autocorrelation into account. Many widely used methods for constructing confidence bands do not have this property, whereas the methods proposed in this thesis do.

2.1.4 Current Methods

The formal literature on confidence bands is scarce and there seems to be a lack of general theory on the subject, as noted by Kolsrud (2007). It should be mentioned that there exists an extensive literature on multivariate confidence intervals for a vector-valued population parameter, such as a multivariate mean, but this literature is of limited interest with respect to confidence bands. First, the literature is about aggregate measures, not the data itself. The variance of an aggregate measure is much lower than that of the data and the central limit theorem is usable only with aggregate measures. Second, most of the approaches are parametric which we want to avoid. We favor the non-parametric approach, because, if no parametric model is assumed, we do not have to check whether the current data fulfills the model assumptions and no fitting of data to the model needs to be done. The range of possible applications is also larger and the result lets “the data speak for itself” since no subjective prior knowledge is added to the process in the form of a model assumption. Of course, in specific applications where the multivariate distribution of the data is well-known, a parametric approach would achieve greater accuracy with less data.

Often the problem of visualizing the variation of a set of curves pops up in different domains and researchers improvise a solution according to their best knowledge. Only few of the solutions end up being published as methods papers – the rest are buried deep in the appendices of research papers they were needed for. This makes comparison of results difficult, sometimes impossible. Therefore well-studied methods and software tools
are needed to harmonize the way results are presented. Accordingly, one objective of this thesis is to present an overview of the methods employed and introduce principled ways to solve the problem of confidence band estimation.

At least in the context of visualization, a common way to construct a confidence band is to connect the end points of several independent scalar confidence intervals into a band. We refer to these as the naïve confidence bands as they naïvely assume that joining together a set of univariate \((1 - \alpha) \cdot 100\%\) confidence intervals would form an \((1 - \alpha) \cdot 100\%\) confidence band. Unfortunately, this is not the case. As was shown above, the overall error rate of a sequence of independent tests can be much higher than that of a single test. Consequently the naïve bands are too narrow.

Economists working with time series prediction and researchers of functional analysis have been the most active in proposing methods to compute confidence bands. For example Hyndman and Shang (2010), Kolsrud (2007), Lütkepohl et al. (2015a), Schüssler and Trede (2016) and Wolf and Wunderli (2015) all recognize the fact that naïve bands are too narrow and propose alternatives. Most solutions follow the same logic: sample a set of curves, identify a set of most central/typical curves and use the envelope of these as the confidence band. What sets the methods apart is the definition of central. Economists tend to use multivariate distance measures or define various heuristics. Functional analysis favors more complicated measures of “deepness” to do the same. A thorough explanation of methods is given in Chapter 3.

### 2.1.5 Contribution

In this thesis we introduce methods to create minimal width confidence bands that take into account (a) multiple hypothesis corrections and (b) the bias caused by finite data. The difference is largest compared to naïve bands as these ignore both (a) and (b). A small number of methods exist that take into account (a) but only a few also consider (b). The contribution of this thesis is thus to take into account both the multiple hypotheses and bias. Additionally, we focus on minimal width bands which guarantees that data points lying within the band reside in areas of higher probability density than points lying outside of the band. In other words, our bands concentrate around the densest areas of the distribution.

Our methods can be applied to both fixed-size datasets as well as in situations where a generator process is available and new samples can
be drawn. All novel methods proposed in this thesis are non-parametric. Modeling assumptions are thus minimal but we do assume that the curves are sampled i.i.d. from a single unknown unimodal distribution.

Publication I introduces minimum width envelope (MWE) confidence bands that have the smallest width among all bands with the same coverage (see Section 3.7.8). The ideas in Publication I closely resemble those presented earlier by Kolsrud (2007) and Staszewska (2007). Publication I improves upon the earlier work by describing a detailed greedy algorithm to compute the MWE bands along with proper considerations of computational difficulty. Like Kolsrud, we used cross validation to guarantee that the coverage is correct also when tested with unseen test data.

Publication II extends the definition of MWE bands to allow room for outliers to appear in the data. A curve is allowed to exit the band at \( L \) time points and still be counted as being inside the band. In statistics the idea of relaxing the FWER control is called the generalized FWER or \( k \)-FWER (Lehmann and Romano, 2005). This formulation allows outliers to exist while minimizing their effect on the final bands. As an important practical improvement over existing literature, the R implementations of the algorithms in Publications I and II are released in GitHub under https://github.com/jutako/multivariate-ci.git.

### 2.2 Shared Variation Between Datasets

The second part of the thesis introduces a novel method to examine the shared variation between datasets. A frequently encountered situation in data exploration is that multiple sources of data are available to describe a set of objects. For example, pictures in the Internet (the objects) can be characterized both by the color intensities in their pixels and by their caption or other textual environment (the data sources). Each data source reveals different aspects about the objects and together the sources give a more comprehensive understanding about the data than any single source alone. A broad umbrella term for these types of analyses is multi-view learning.

In this thesis we focus on a sub-type of this problem, where a single phenomenon is described by several datasets. A typical example is an experiment with multiple participants. The experiment (the phenomenon) is described by the physiological responses of the subjects (the data sources). We organize the data from each subject into a separate dataset. As the
datasets all describe the same underlying phenomenon, it is reasonable to assume that they share patterns as well.

In this context a dataset is a collection of related features, organized into a matrix, in our case with one column per feature. Furthermore, we focus on applications where the features are time series and consequently the rows of the dataset index time. Within multi-view learning literature it has become customary to call the datasets views as each dataset gives a different angle on the problem at hand.

The basic idea behind multi-view learning is that the views share patterns because they all describe the same underlying phenomenon. This reasoning can also be reversed: if the views do not share patterns then there is no underlying phenomenon. Consider the physiological experiment example. If we measure a group of subjects, each in a different environment performing different tasks, there is no reason to believe that the resulting datasets would share patterns. Hence we assume, by default, that the datasets are independent. However, if the test subjects all undergo the same experimental paradigm it is natural to assume that their data show similar patterns induced by the underlying experiment. For example the auditive evoked response potentials (ERPs) from a group of test subjects all look similar since the primitive brain responses to a stimulus are rather constant across subjects. Note that the independence assumption is an important one: if we find some shared patterns, those patterns must be due to the experiment, since the views are otherwise unrelated to each other.

2.2.1 Applications

The problem of finding shared patterns in a set of time series datasets is a common one. In brain research shared stimuli, such as movies, have been successfully used to, e.g., study cortical activation patterns (Hasson et al., 2004; Jääskeläinen et al., 2008; Kauppi et al., 2010). Also applications where a group of people is studied together would benefit from methods that are able to pinpoint points in time when the group has responded simultaneously. Examples include the study of team performance (Henning et al., 2001), studies of social interaction (Hari et al., 2013), and affective computing (Ekman et al., 2012).
2.2.2 Current Methods

The method we are proposing is not a clear member of any of the existing sub-fields of multi-view learning, but it is most closely related to subspace learning. The goal of subspace learning is to find alternative representations of the data, the subspaces, that reveal shared patterns between views. Matrix projections such as canonical correlation analysis (CCA) (Hotelling, 1936) and multi-set CCA (MCCA) (Kettenring, 1971; Tenenhaus, 2011) are one option. The problem of CCA is that it generalizes poorly to multiple datasets. For example, for a set of three datasets A-B-C of which one, say C, is independent of the others, MCCA still finds shared patterns. These correspond to the shared variation between A and B. Hence the method completely misses the fact that C did not share any patterns with A and B.

Another alternative are methods that factorize the variance of the combined views, such as various factor analysis methods. The group factor analysis (GFA) by Klami et al. (2015) is one recent example of this kind. Factor analytic models are expressive and answer many questions at once. However, they are also complex, make many assumptions about the data, are often slow to train, and cannot be easily modified.

2.2.3 Contribution

To address these shortcomings we propose a new procedure to filter out variation that is not shared by all of the datasets. As usual, different definitions of shared give rise to different methods. Our method borrows the definition of sharedness from redundancy analysis (RDA) by Legendre and Legendre (1998). Given views A and B, the RDA simply states that the variation A shares with B is the part of A that can be explained using least squares regression from B to A. We extend this idea to situations with multiple datasets and to other forms of regression. The result is a regression based filter that dampens variation that is not shared by all views.

The filtering effect is achieved by chaining the regressors. This means that to extract the shared variation projection for one view information from all other views is used, one at a time, in sequence. This structure enforces an important and desirable property: if any of the views is independent of some other view, the chain does not pass information and the output is a noisy constant. In other words, the output reflects the “greatest common divisor” of the input views. Unlike in the case of MCCA,
a regression chain clearly reveals if one view is independent of the others. Moreover, as our approach uses regression functions there are no restrictions on how many features each view can have.

An additional convenient property of the RDA style shared variation definition is that it can be altered by changing the regressor used. Robust linear regression can be used to avoid biases due to outliers or the method can be made non-linear by choosing a non-linear regression function. With modern analysis environments packed with different alternative regressors, the characteristics of the method can be easily adapted to a wide range of research questions. This is an advantage with respect to probabilistic approaches where the modeling principle cannot be easily changed.

Publication III describes a “COmmon COmponents using REGression” (COCOREG) method that implements the ideas introduced above. A simple algorithm is given to compute the COCOREG projection along with description of its basic properties. The publication also compares COCOREG against CCA and GFA. An R implementation is released in CRAN (https://CRAN.R-project.org/package=cocoreg) and in GitHub (https://github.com/bwrc/cocoreg-r).

2.3 Summaries of Publications

2.3.1 Publication I


In Publication I we consider the problem of finding confidence bands to describe the variability of a time series. Given a set of N time series the task is to find a confidence band that contains a \((1 - \alpha)\) -fraction of the observations. We set \(K = \lfloor \alpha N \rfloor\) and construct a band by removing observations from the dataset to yield a set of \(N - K\) time series whose envelope is minimized. This is the minimum width envelope (MWE) problem, which we show to be NP-hard. We develop a greedy heuristic algorithm to solve it and compare against quantile and distance based confidence band methods. We also describe a method to find an effective significance level \(\alpha_{\text{eff}}\) and an effective number of observations to remove \(K_{\text{eff}}\), such that the coverage
of the resulting confidence band is below $\alpha$ also for unseen test data. We demonstrate that our method can be used to construct confidence bands with guaranteed coverage control, also when there is too little data for the quantile-based methods to work.

### 2.3.2 Publication II


In Publication II the MWE problem from Publication I is re-examined while allowing a time series to exit the band at $L$ time points and still be counted as a within-band observation. The idea is to allow certain dimensions of the data to have spurious observations without affecting the confidence bands at those dimensions. The reformulated problem is NP-hard. To solve it two general algorithms are presented: minimum intervals (MI) and a greedy top-down algorithm. The first is a baseline method while the second is a modified version of the solution proposed in Publication I. Furthermore, a polynomial time approximation algorithm is described for the sub-problem where $K = 0$ but $L$ is not constrained.

### 2.3.3 Publication III


The Publication III presents a regression based algorithm for the extraction of variation that is common to all datasets in a given collection of arbitrary size. The algorithm, named COCOREG, can be seen to extend redundancy analysis to more than two datasets. It utilizes chains of regression functions to extract the shared variation in the original data space. The algorithm can be used with any regression function, both linear and non-linear. This allows flexibility for the analyst to select which types of relationships to model.
2.3.4 Publication IV


The Publication IV extracts significant temporal patterns from interval sequences. An interval sequence is an ordered sequence of some intervals, such as the time lags between successive heart beats in heart rate variability analysis. The analysis is based on first defining a null model that represents the experimenters’ prior knowledge of the data. To spot patterns that do not fit the null model we need ways to generate data from the null model. For this purpose we provide five ways to resample data: uniform, interval, fixed subsequence, fixed Fourier parameters, and fixed distance function randomizations. Temporal patterns in observed data that are not explained by the null model can be spotted by comparing the observed Fourier spectrum to that of the null model. The described approach allows for a nested continuum of analyses: starting from a totally structureless null model constraints can be gradually added until the observed data fits the model. The methods presented are applied to two real-life datasets; a medical heart inter-beat interval dataset and a word dataset from a book.

2.4 Outline of the Thesis

The thesis is organized as follows. First, theory and methods to compute confidence bands are given in Chapter 3, followed by selected applications in Chapter 4. An overview of multi-view learning and the details of the COCOREG algorithm are presented in Chapter 5. Discussion related to the strengths and weaknesses of the proposed methods as well as directions for future work appear in Chapter 6. The thesis concludes with the original publications.
3. Multivariate Confidence Bands in Two Dimensions

This chapter discusses confidence bands, two-dimensional visualizations of multivariate distributions. Publications I and II introduce methodology for computing the bands and methods in Publications III and IV benefit from the bands as tools that visualize variation and help in finding statistically significant patterns. Therefore, confidence bands form a basis of this thesis, tying together the publications.

The chapter starts with a short discussion about statistical interval terminology and continues by laying out the relationship between density estimation and confidence bands followed by a taxonomy of desired properties for the bands. The chapter continues with an outline of the basic concepts of multiple hypothesis significance testing (MHST), some definitions related to coverage and descriptions of the methods available for computing confidence bands. Thereafter procedures for the estimation of expected coverage are presented. The chapter concludes with two real data applications that showcase the potential of confidence bands as a visualization and statistical tool.

3.1 Statistical Intervals and Their Definitions

Statistical literature recognizes several different intervals with fixed definitions and use cases: probability intervals define limits for some fraction of the probability mass of some distribution, confidence intervals apply to estimates of population parameters, prediction intervals concern future observations, tolerance or enclosure intervals are probability intervals with an associated measure of confidence and credible intervals are the Bayesian counterpart of confidence intervals (Hahn and Meeker, 1991; Willink, 2012). Of these the term confidence interval is the most widely used and also misused. The Publications I-IV, too, have essentially rede-
fined the term confidence interval to be used in a new context. We will next briefly define all of the intervals mentioned above, followed by discussion regarding why we have decided to call our intervals confidence intervals.

The definitions below follow those given by Willink (2012) and are generally accepted in the statistics community. They apply either for a continuous scalar random variable $X$ that follows a distribution $F = \Pr(X \leq x)$ or for a statistic $\theta$, an (unknown) parameter of $F$ that can be estimated from a random sample from $F$. We mark fractions of probability content with $\gamma$, the statistical confidence level with $\eta$ and generic probabilities with $p$. Parameters $\gamma$ and $\eta$ can be associated with a distinct source of variation: $\gamma$ is related to the variance of $X$ and $\eta$ bounds the sampling error. Parameter $p$ can be associated with any source of variation depending on the application. Capital letters stand for random variables whereas lower-case indicates non-random.

The first three definitions are statements related to the random variable $X$:

**Definition 1** (Probability interval). A $\gamma \cdot 100\%$ probability interval for $X$ is a non-random interval $[x_l, x_u]$ such that the probability that $X$ lies between these limits is $\gamma$.

**Definition 2** (Tolerance interval). A $\gamma \cdot 100\%$-content tolerance interval for $X$ with confidence level $\eta \cdot 100\%$ is a random interval $[X_l, X_u]$ that has probability $\eta$ of covering at least a fraction $\gamma$ of the probability content of the distribution of $X$.

**Definition 3** (Prediction interval). A $p \cdot 100\%$ prediction interval for a single future observation of $X$ is a random interval $[X_l, X_u]$ with probability $p$ of covering the value that will be taken by $X$.

The following two intervals concern an unknown parameter $\theta$ of the probability distribution $F$:

**Definition 4** (Confidence interval). A $\eta \cdot 100\%$ confidence interval for an unknown non-random statistic $\theta$ is a random interval $[\Theta_l, \Theta_u]$ with probability $\eta$ of covering the true value $\theta$.

**Definition 5** (Credible interval). The $\gamma \cdot 100\%$ credible interval for a random parameter $\Theta$ is a non-random interval $[\theta_l, \theta_u]$ of the posterior probability distribution of $\Theta$ such that

$$\Pr[\theta_l(x) \leq \Theta \leq \theta_u(x)|X = x] = \gamma,$$
where $X = x$ emphasizes that the interval strongly depends on the observed data.

The interval definition used in this thesis is the following:

**Definition 6** (High density confidence interval). A $\gamma \cdot 100\%$ (high density) confidence interval for $X$ is the shortest contiguous non-random interval $[x_l, x_u]$ such that the probability that $X$ lies between these limits is $\gamma$.

The high density confidence interval may also be seen as the shortest $\gamma \cdot 100\%$ probability interval.

Note that while there exist many probability intervals for any given distribution (Definition 1) there is usually only one high density confidence interval (apart from uniform distributions where all intervals contain equal densities). For a unimodal distribution, every point inside a high density confidence interval has a higher or equal probability density than any point outside the interval. Therefore the high density interval is centered around the densest parts of the data, not necessarily true for probability intervals. Given a sample we approximate a high density confidence interval by computing the shortest interval that covers $\gamma \cdot 100\%$ of the observations (Liu et al., 2015).

When comparing the Definitions 1-5 from statistics to the Definition 6 adopted here, it becomes clear that there are some fundamental differences. First, our intervals do not apply to an unobserved population parameter but to the random variable $X$. Second, we make statements about the coverage of the intervals with respect to the underlying probability distribution but we do not make direct claims as to how confident we are that the intervals we produce actually possess the desired properties. Third, in Publications I and II we have used $\alpha$ to parametrize the probability mass covered whereas in statistics $\alpha$ is usually reserved for statistical confidence.

So why do we keep on calling our intervals confidence intervals? Of the intervals above probability, tolerance and prediction intervals are closest to the Definition 6. The first option, probability interval, is poorly known by the general audience, the term is very uninformative and the interval is not unique. The second option, tolerance interval, has two parameters, the fraction covered $\gamma$ and the confidence $\eta$, yet our interest lies mainly on establishing a correct $\gamma$ and we treat the confidence part more pragmatically in Section 3.8. The third alternative, prediction interval, has a strong emphasis on predicting the future, whereas confidence bands are equally useful in describing the present.
We could make up a new term *high density interval* drawing parallel to the highest density regions studied by Hyndman (1995), Hyndman (1996) and Winker et al. (2016), but this would probably just add to the confusion. In addition, there happens to be a frequent misunderstanding that a confidence interval is an interval that covers a fraction \( \eta \) of a distribution. Definition 6 is actually quite close to this inaccurate meaning, and, considering the options, we think it is for the non-statistician most informative to call our intervals confidence intervals. To maintain consistency with the publications, we keep on using \( (1 - \alpha) \) to parametrize the percentage of the probability mass that is covered by our intervals, despite its different use in statistics.

### 3.2 Relation to Density Estimation

In univariate statistics knowing the probability density function (pdf) immediately allows us to generate confidence intervals or highest density regions for the distribution (Hyndman, 1996). Similarly, when constructing a visualization for an \( M \) dimensional multivariate distribution \( F \), a natural starting point is to model or estimate the probability density function of \( F \) directly and produce the visualization based on this model. This modeling process is referred to as multivariate density estimation. As with most modeling tasks both parametric and non-parametric methods exist (Gelman et al., 2003; Silverman, 1986). Parametric approaches describe the data using a mathematical model and try to select the model parameters in a way that makes the model fit the observations. They provide a generative model for \( F \) that can be further used to answer questions about the data. An example would be the fitting of a multivariate normal distribution to a set of samples from \( F \).

In contrast, non-parametric methods make much milder assumptions about the form of \( F \). A typical requirement is the independence of the samples and that the data comes from a smooth distribution. Non-parametric methods provide an approximation of the probability density function based on the idea that regions with many observations must have higher density. Kernel density estimates are an example of this (Silverman, 1986) and extensions to multivariate distributions exist (Scott, 2015). The estimate can further be used for visualization purposes.

The focus of this thesis is in two-dimensional visualizations of the data, where the data values are plotted against the \( M \) dimensions. Figure 3.1
shows an example, where a collection of electrocardiographic (ECG) waveforms is plotted against time. In particular, we focus on bands that graphically show the variation in the dataset. A band consists of two vectors that define its upper and lower limit and the bulk of the data lies between these limits. If the band is guaranteed to contain a certain percentage of the data we speak of confidence bands. Figure 3.1 gives an idea of how the bands look like.

![Figure 3.1](image)

**Figure 3.1.** Confidence bands for a set of $N = 1507$ normal heart beats. Colored lines are the boundaries of the naïve quantiles (NAIVE.QNT) and minimum width envelope (MWE) confidence bands (introduced in Section 3.7), while a subset of 30 random observations are shown in light gray. Heart beats have been aligned using the R-peak of the QRS complex.

From the viewpoint of density estimation a multivariate observation is just a single point in $\mathbb{R}^M$. Likewise, a confidence band is equivalent to a $M$ dimensional rectangular region, a hyper-rectangle, that contains a predefined percentage of the observations. Additionally, we would like to focus on areas of high probability density. Accordingly, the optimal hyper-rectangle should be the smallest one that completely contains the highest density region of the probability distribution (Hyndman, 1996; Hyndman and Shang, 2010). Computing such a region in high dimensional data is difficult: for a given accuracy, the need for i.i.d. samples grows exponentially with dimension (see also Section 3.8). Hence most datasets are too small to allow for a true multivariate density estimation. Consequently, due to the size of the datasets and the target being a 2D visualization, most of the methods presented in this thesis do not rely on direct multivariate density estimation. Instead, quantile based reasoning is applied by defining an ordering of the multivariate observations in the 2D space: some percentage of the most “central” observations (curves) form the confidence band. It is the definition of “centrality” that sets the methods apart.
It is clear that a hyper-rectangle is a rather crude approximation, e.g., for a spherical probability distribution, but for the reasons explained below, the bands can still be useful. Two-dimensional visualizations have many convenient properties. First, many datasets can be naturally expressed in 2D. This is especially true for time series data, which is abundant in all fields of research. Second, humans are good at evaluating 2D patterns making confidence bands an intuitive tool for graphical pattern recognition. Third, like their univariate counterparts, confidence bands provide a practical way of doing graphical hypothesis testing. By using a band it is for example easy to spot areas where observations differ from the null model – an indication for the rejection of the null hypothesis. Last, most written communication still consists of 2D graphics, a perfect match for a 2D visualization.

Confidence bands can be computed for all kinds of data. However, as the methods we propose are best suited for datasets that exhibit some form of correlation between the dimensions, time series provide a natural application domain. Moreover, the standard visualization for time series is a two-dimensional plot, which matches our approach perfectly.

### 3.3 Applications of Confidence Bands

As pointed out in the introduction, confidence bands are a method to visualize the variation of time series and similar datasets. Time series being a common type of data there are numerous applications of confidence bands as well.

One important application domain is forecasting, especially time series prediction. Applications include forecasts for ARMA (Chatfield, 1993; Box et al., 2015) and ARCH-models (Reeves, 2005), fan charts in trend analysis (Akram et al., 2016; Britton et al., 1998) and weather forecasts (FMI, 2017).

Figure 3.2 illustrates a fan chart as used by the Bank of England in their inflation reports. The chart is constructed to reflect the collective judgment of a monetary policy committee (MPC) such that a “mature estimate” of the parameter in question, in this case the gross domestic product (GDP), lies within the darkest band 30 %, within two darkest bands 60 % and within all green bands 90 % of the time. These percentages apply for each quarter separately (Carney et al., 2016).

Another application domain where confidence bands are often used is
regression analysis, where one might extrapolate the model into the future or study model stability with respect to the input data and model parameters. A review of the regression related literature can be found in Liu et al. (2008).

Examples of more specific applications include density estimation (Bickel and Rosenblatt, 1973; Bissantz et al., 2007; Hall, 1992; Owen, 1995), spectral density estimation (Neumann and Paparoditis, 2008), receiver operating characteristic (ROC) curves (Hall et al., 2004) and impulse response functions (Kilian, 1998; Pesavento and Rossi, 2006), to name a few.

A common denominator for all these applications is that a single observation is vector valued, a curve, and that there exists some underlying distribution $F$, from which samples can be drawn. If $F$ is specified by some model, samples are easy to generate. If $F$ is governed by some natural process, the sampling happens by taking repeated measurements. In either case, a dataset is produced and this data can be used to produce an empirical estimate of the distribution $\hat{F}$, which can further be used to construct a confidence band.
3.4 Properties of Confidence Bands

Some notation will be introduced next in order to simplify the description of band related properties and computation methods. In the context of confidence bands the observational unit is an $M$-dimensional random row vector $x \in \mathbb{R}^M$ that follows an unknown distribution $x \sim F$. In practice $F$ can be approximated using the empirical distribution $\hat{F}$ of a sample of $N$ realizations of $x$. Let the observed data be organized into a matrix $X \in \mathbb{R}^{N \times M}$. The rows of $X$ are data vectors in $\mathbb{R}^M$ denoted by $x_n$, and the vector of values in column $m$ is marked $X(:, m)$. The notation $x_n(m)$ marks a scalar value at the $n$-th row and $m$-th column of $X$. The row index $n = 1 \ldots N$ indexes the samples of the data (draws from $x \sim F$), while the column index $m = 1 \ldots M$ runs over time for time series data and over variables for other types of data.

Let us plot the vectors $x_n$ onto a plane with horizontal axis spanning over the dimensions $M$ and vertical over the values $x_n(m)$. A band $B$ is a bounded area within this plot. We define:

**Definition 7 (Band).** Given $X \in \mathbb{R}^{N \times M}$, a band is a pair $B = (x_\ell, x_u)$ of lower and upper bound vectors $x_\ell(m), x_u(m) \in \mathbb{R}^M$ satisfying $x_\ell(m) \leq x_n(m) \leq x_u(m)$ for all $m = 1, \ldots, M$.

Unless stated otherwise, a data vector $x_n$ is within or inside the band $B$, if $x_n$ does not break out of the band at any location, i.e., if $x_\ell(m) \leq x_n(m) \leq x_u(m)$ for all $m \in M$.

Consider a subset of indices $I \subseteq [N]$, where $[N]$ is a shorthand for $\{1, \ldots, N\}$. An envelope of $X(I, \cdot)$ is a band that contains all of the data vectors $x_n$ for which $n \in I$:

**Definition 8 (Envelope).** The envelope of $X(I, \cdot)$ is a band denoted by $\text{env}(X, I) = (x_\ell, x_u)$, where $x_\ell(m) = \min_{n \in I} x_n(m)$, $x_u(m) = \max_{n \in I} x_n(m)$ and $I \subseteq [N]$.

Depending on the application, the band might be expected to have different properties. These properties are direct generalizations of the respective properties of univariate prediction intervals or highest density intervals: symmetry, centeredness and shortness (Hyndman, 1995, 1996). For example, Kolsrud (2007) identified five properties for the bands. We will next present a similar set of band properties to show the design trade-offs related to constructing a confidence band.
Let \( w(m) = x_u(m) - x_\ell(m) \) be the width of the band \( B \). Defining symmetricity and centeredness in absolute units gives rise to the following definitions:

**Definition 9** (Constant width band). A band \( B = (x_\ell, x_u) \) is of constant width, if \( w(m) = x_u(m) - x_\ell(m) = c \), for all \( m \), where \( c \) is a constant and \( x_\ell(m), x_u(m) \in \mathbb{R} \).

**Definition 10** (Symmetric band). A band \( B = (x_\ell, x_u) \) is called symmetric with respect to mean/median, if \( x_u(m) - \bar{x}(m) = \bar{x}(m) - x_\ell(m) = w(m)/2 \), for all \( m \), where \( \bar{x}(m) \) is the mean or median of observations in \( X \) and \( x_\ell(m), x_u(m) \in \mathbb{R} \).

Also probability mass can be used to define symmetricity and centeredness. A band is called trajectory-centered if it is equally probable for \( x \) to break out of the band in either direction:

**Definition 11** (Trajectory-centered band). A band is called trajectory-centered, if \( \Pr_{x \sim F}[x < x_\ell] = \Pr_{x \sim F}[x_u < x], \) where \( x < y \) if \( x(m) < y(m) \) for at least one \( m = 1, \ldots, M \) (and respectively for \( x > y \)).

By requiring the pointwise tail probabilities to be equal, we arrive at a point-centered band:

**Definition 12** (Point-centered band). A band is called point-centered, if \( \Pr_{x \sim F}[x(m) < x_\ell(m)] = \Pr_{x \sim F}[x_u(m) < x(m)] = \alpha/2 \), holds independently for all \( m \). The parameter \( \alpha \) is the desired significance level.

Naïve confidence bands (NAIVE.QNT) and Bonferroni bands (BONF.QNT) would be examples of point-centered bands, if the required quantiles are estimated such that the tail probabilities are equal. However, the quantiles in this thesis are computed such that the interval \( [x_\ell(m), x_u(m)] \) is the shortest possible, which breaks point-centeredness but guarantees the densest part of the data to be inside the interval.

If instead of tail probabilities we focus on the probability mass inside the band, we get a uniform band:

**Definition 13** (Uniform band). A band is called uniform, if \( \Pr_{x \sim F}[x_\ell(m) \leq x(m) \leq x_\ell(m)] = 1 - \alpha \), independently for all \( m \).

A point-centered band is also a uniform band. However, a uniform band might not be point-centered as the tail probabilities \( \Pr[-\infty < x(m) \leq x_\ell] \) and \( \Pr[x_u \leq x(m) < \infty] \) might not be equal. The NAIVE.QNT and BONF.QNT bands in this thesis are uniform.
Lastly, a band is called *minimal* if it is the smallest of all bands that have the same coverage. The exact definition is given later in Section 3.6 (see Definition 15).

Note that the definitions are not mutually exclusive: a band might simultaneously have several of these properties.

### 3.5 Multiple Hypothesis Significance Testing (MHST)

Before going into details about band coverage, certain concepts related to statistical testing need to be explained. Hypotheses are exact statements about data, such as "the mean $\mu$ of a random variable $X$ is $\mu_x$". The usual setup is to have a null hypothesis $H_0$ and the observed data is used to find evidence against $H_0$. The strength of this evidence is measured using a statistical test and a corresponding test statistic $t$. The statistic $t$ is chosen such that it measures the deviation of the observed data from the $H_0$ and its distribution under $H_0$ is known. The observed value $t^*$ of $t$ is computed from the sample and compared to the distribution of $t$ under the null hypothesis $H_0$. The goal is to find out, how often does the null hypothesis produce values for $t$ that are as or more extreme than the observed $t^*$. A p-value is computed, indicating the probability of observing $t$ that is as or more extreme than $t^*$ assuming $H_0$ is true. Small p-values are interpreted as evidence against $H_0$. In the above example we would compare the observed sample mean of $X$ to the distribution of sample means under $H_0 : \mu = \mu_x$.

Some common terminology related to hypothesis testing is given in Table 3.1. Two types of errors can happen: rejection and acceptance errors. A rejection error (type I error) is committed, if a true $H_0$ is erroneously rejected. Similarly an acceptance error (type II error) happens, if a false $H_0$ fails to get rejected. To set an upper bound for the probability of making a rejection error a significance level $\alpha$ is set for the test. If p-value $\leq \alpha$ the null hypothesis is rejected and we conclude that the decision to reject $H_0$ might be false with probability at most $\alpha$. However, no such claim can be made regarding the acceptance error. If p-value $> \alpha$ we “fail to reject the null hypothesis”, but we do not know what is the probability of committing an acceptance error.

When $M$ independent tests are made, such as when screening a set of $M$ variables, the probability of committing a type I error becomes large very fast as $M$ increases. For example, when performing $M = 10$ independent
Table 3.1. Test outcomes of a traditional hypothesis test along with associated terminology.

<table>
<thead>
<tr>
<th>Null hypothesis is in reality:</th>
<th>Test outcome</th>
<th>valid / true</th>
<th>invalid / false</th>
</tr>
</thead>
<tbody>
<tr>
<td>reject</td>
<td>rejection error</td>
<td>true positive</td>
<td>type I error / false pos.</td>
</tr>
<tr>
<td>fail to reject</td>
<td>true negative</td>
<td>acceptance error</td>
<td>type II error / false neg.</td>
</tr>
</tbody>
</table>

test at once the probability of making at least one type I error becomes $1 - (1 - \alpha)^M$, where $\alpha$ is the significance level. For a typical $\alpha = 0.05$ this gives a 40% probability of committing at least one type I error. If the dimension of the data grows to, say, $M = 100$ this probability rises to 99%. The exact same problem is observed for confidence bands, if the band is constructed by joining pointwise confidence intervals: even though the individual intervals cover $1 - \alpha$ fraction of the distribution for each dimension, a much lower fraction of the curves stay completely within the naïve band. Consequently several authors make a distinction between pointwise and simultaneous confidence bands (Akram et al., 2016; Kolsrud, 2007; Staszewska-Bystrova and Winker, 2013; Winker et al., 2016).

To make robust decisions one therefore must control the overall error rate of the tests. There are several different definitions of error rate available, of which the most important are the family wise error rate (FWER) and the false discovery rate (FDR) (Benjamini and Hochberg, 1995; Dudoit et al., 2003; Storey and Tibshirani, 2003). The FWER limits the probability of making one or more false rejections to be less than $\alpha$, while the FDR states that the proportion of false rejections among all rejections should be at most $\alpha$. FWER is often said to be a conservative approach in the sense that it makes sure that with probability $1 - \alpha$ there are no false rejections. This is adequate in applications where a false rejection is considered much more harmful than a false acceptance. On the other hand, FDR allows false rejections to happen, it just limits their number relative to all rejections made. Consequently FDR is often used if false rejections are not that harmful in comparison to false acceptances.

A confidence band can also be considered as a graphical version of a multivariate statistical test. It can be used to answer questions such as “does the candidate curve come from the same distribution $F$ that was used
to construct the confidence band”?
If the candidate curve exits the band at
one or more time points, we have reason to believe that the candidate curve
did not come from \( F \). On the contrary, if the candidate curve is completely
within the band we make the opposite decision: the candidate curve could
come from \( F \). However, this is not the same thing as stating that “the
candidate curve comes from \( F \)”.

### 3.6 Coverage

Coverage indicates how large portion of the distribution \( F \) is covered by
the band. It is a very important property for a confidence band as the
interpretation and usefulness of the band rely on proper coverage.

To compute coverage the word “cover” needs to be defined exactly. To this
end we define the error of \( x \) with respect to a band \( B = (x_\ell, x_u) \) to be the
number of times \( x \) exits the band, i.e.,

\[
\text{Err}(x \mid B) = \sum_{m=1}^{M} I [x(m) < x_\ell(m) \lor x_u(m) < x(m)].
\]

(3.1)

Here the indicator function \( I[\square] \) equals one if the condition \( \square \) is satisfied
and zero otherwise. An observation \( x_n \) is inside the band \( B \), if the error
stays below a predefined value.

Using this definition of error we can define coverage:

**Definition 14 (Coverage).** Let \( F \) be the multivariate distribution function
of \( x \in \mathbb{R}^M \). The coverage of a band \( B \) is defined as

\[
\gamma = \Pr_{x \sim F} [\text{Err}(x \mid B) \leq L],
\]

where \( L \) is a predefined error level.

In the vast majority of applications it has been chosen \( L = 0 \), which
corresponds to not allowing \( x \) to exit the band. This convention also
corresponds to the FWER error rate discussed earlier: a \( L = 0 \) band with
\( 1 - \alpha \) coverage is a FWER controlling confidence band. Using non-zero
values for \( L \) is less common. At the time of writing the only other reference
proposing this is in the context of confidence bands is by Wolf and Wunderli
(2015), who proposed it in their bootstrap method related to path forecasts.

A band with \( 1 - \alpha \) coverage is called a \((1 - \alpha)\)-confidence band. Such a band
is guaranteed to cover repeated draws from the underlying distribution \( F \)
with probability \( 1 - \alpha \). Among all bands with the same coverage, it makes
sense to search for the smallest one to achieve high statistical power. We define a minimal band to be

**Definition 15** (Minimal band). A band $B$ is called minimal with respect to a measure of size $V \geq 0$, if $V(B) \leq V(B')$ for all bands $B'$ having the same coverage as $B$.

In this thesis we define $V$ to be the overall width of the band:

$$V_{\text{area}}(B) = \sum_{m=1}^{M} (x_u(m) - x_l(m)),$$  \hspace{1cm} (3.2)

but other options such as volume

$$V_{\text{volume}}(B) = \prod_{m=1}^{M} (x_u(m) - x_l(m))$$  \hspace{1cm} (3.3)

are also possible.

Smallest bands play the role of a high density confidence interval (see Definition 6) in the multivariate domain. Like the high density confidence interval encloses areas of highest probability density, the smallest bands are the smallest box shaped regions that cover regions of highest probability density.

Note however, that if we would allow the intervals or confidence bands to be disjoint, i.e., consist of multiple separate parts, we could achieve even shorter intervals or smaller bands, e.g., for multimodal distributions. Hyndman (1996) uses the term highest density region (HDR) to refer to regions that can reveal multi modality in a distribution and provides methods to find these in one and two dimensions. Extending confidence bands to account for multimodality remains a future challenge.

In practice confidence bands are constructed from finite datasets, which gives rise to some additional definitions for coverage (Kolsrud, 2007). Consider a sample $X_0$ from distribution $F$, the corresponding empirical distribution $\hat{F}_0$ and a related confidence band $B_0 = B(\gamma_0, \hat{F}_0)$. The band $B$ is a function of the distribution $F$, the nominal coverage $\gamma_0$ (desired coverage) and the actual sample $\hat{F}_0$ drawn from $F$. In addition, the band depends on the algorithm used to compute it, but we omit this to make the notation simpler. The band gets wider as $\gamma_0$ is increased and more accurate as the sample size $N$ is increased.

Most confidence bands are constructed such that their coverage within the learning sample $\hat{F}_0$ is fixed to $\gamma_0 = 1 - \alpha$. The _in-sample coverage_ is
defined as
\[ \gamma_s = \Pr_{x \sim \hat{F}_0} [\text{Err}(x \mid B_0) \leq L]. \] (3.4)

When B is tested against new unseen values from F, the observed coverage tends to be lower than the in-sample coverage. We define the true coverage of B under F to be
\[ \gamma_t = \Pr_{x \sim F} [\text{Err}(x \mid B_0) \leq L], \] (3.5)

which is a fixed but unknown quantity. Equations (3.4) and (3.5) are very similar, the only difference being the distribution over which the probability is taken. As F is usually an unknown distribution, \( \gamma_t \) cannot be computed directly but it can be estimated using resampling. We consider the band \( B_0 \) to be fixed and define its expected coverage to be:
\[ \gamma_e = \gamma_e(\gamma_0, F, N) = \mathbb{E}_{\hat{F} \sim F_N} \left[ \Pr_{x \sim \hat{F}} [\text{Err}(x \mid B_0) \leq L] \right]. \] (3.6)

The notation \( \gamma_e(\gamma_0, F, N) \) underlines the fact that the expected coverage depends on \( \gamma_0 \), the distribution F and the size of the sample N drawn to generate \( \hat{F} \) (Kolsrud, 2007). In practice the expectation is approximated by drawing repeated samples from F (if available) or by means of some cross validation scheme as later described in Chapter 3.8.

Literature on univariate confidence intervals suggests that intervals computed on a finite sample tend to be too short (Burr, 1994; Clements and Taylor, 2001; Kim, 2004; Royall and Cumberland, 1985; Stine, 1985). Likewise, confidence bands tend to be too narrow as observed in Publication I and also by several other authors (Lütkepohl et al., 2015a; Kolsrud, 2007; Staszewska-Bystrova and Winker, 2013; Winker et al., 2016). To quantify the error we define the following bias:
\[ \Delta \gamma = \gamma_e - \gamma_0 < 0. \] (3.7)

The bias \( \Delta \gamma \) is caused by the finite sample size N and it is reduced if more data are used. For most bands \( B(\gamma_0, \hat{F}_0) \) the bias is negative meaning that the band is actually narrower than intended.

Equation (3.7) describes an empirical observation that is strongly backed up by similar observations in the univariate confidence intervals literature. For an intuition into why this is the case, consider a univariate uniform random variable \( X \sim U(0, 1) \). In this case the order statistic \( X_{(i)} \), that is the \( i \):th value in an ordered set of N observations, has a beta distribution with the expected value of \( \mathbb{E}[X_{(i)}] = \frac{i}{N+1} \) (Arnold et al., 2008). We get for
the expected coverage

\[ \gamma_c = E[X(j) - X(i)] = \frac{j}{N+1} - \frac{i}{N+1} = \gamma_s - \frac{1}{N+1}, \tag{3.8} \]

where \( \gamma_s = \frac{i-j+1}{N} \) is the in-sample coverage of the band \([X(i), X(j)]\) (Kolsrud, 2007). So clearly, the expected coverage is \(1 + \frac{1}{N+1}\) smaller than the in-sample coverage. For example, the expected coverage for a univariate interval of \(N = 100\) samples using \(\gamma_s = 0.9\) becomes \(\gamma_e = \frac{95}{101} \approx 0.94\). For a \(M = 10\) dimensional set of independent variables this decreases further to \(\gamma_e = 0.88^{10} = 0.28\). Although the effect is smaller if the dimensions correlate (see Section 3.8), it is clear that controlling the in-sample coverage alone is not enough.

However, as Equation (3.8) suggests, gathering more data helps. Increasing the denominator makes \(\gamma_c\) converge towards \(\gamma_s\) – at least in the independent variables case. That is, as \(N \to \infty\), \(\hat{F} \to F\) and \(\gamma_l \to \gamma_0\) and \(\gamma_e \to \gamma_0\) (Kolsrud, 2007).

### 3.7 Methods for Computing Confidence Bands

Confidence bands are a very intuitive construct and therefore independently used in many applications. However, there seems to be a lack of general theory on the subject and application domain has developed its own definitions as well as domain-specific methods for computation (Kolsrud, 2007). This causes difficulties in interpretation of the bands and consequently lowers the general appeal towards the method. This chapter gives an overview of the existing literature and the variety of methods employed.

Theoretically a straightforward way to obtain a confidence band would be to first estimate the joint probability density of the underlying multivariate distribution, determine its highest density region and take the bounding hyper-rectangle as the band. First attempts of doing so have indeed appeared recently, see, e.g., Winker et al. (2016) for highest density regions and bands based on kernel density estimates. However, in general, computing such a region for high dimensional data is difficult: for a given accuracy, the need for i.i.d. samples grows exponentially with dimension (Silverman, 1986). In addition, from a time series point of view, a direct multivariate density estimation is unnecessarily complex as the dimensions are correlated leaving much of the space empty. Besides, visualizing curves as dots in vector space is somewhat unintuitive, apart from
some outlier detection applications (Miguez, 2016). For these reasons the methods proposed below are closely tied to the two-dimensional representation, where time series are represented as curves. The extensive literature related to both parametric and non-parametric density estimation (Scott, 1992; Silverman, 1986) is thus considered to be largely out of scope.

3.7.1 Naïve Quantiles

The most natural extension of a univariate confidence interval into multiple dimensions is to compute the one-dimensional \((1 - \alpha)\)-confidence interval independently for each variable \(m\) and connect these into a band. We refer to this method as the **NAIVE.QNT** approach and it is an example of a pointwise confidence band.

The univariate quantiles can be computed using a wide variety of different methods (Hyndman and Fan, 1996). The methods differ in two main aspects: (i) allowed endpoints and (ii) symmetricity. Regarding endpoints, some methods restrict the quantile endpoints to be observed data points while others use models to estimate the tail probabilities. Restricting the endpoints to actual observations is a more data driven alternative, the downside being that the smallest attainable quantile is \(1/N\), which might be a problem in small datasets.

Symmetricity, on the other hand, refers to the position of the interval with respect to the data. The estimation process can either find the shortest interval with the given coverage or extract equal probability masses from both tails of the distribution. This makes a difference for strongly skewed distributions. By focusing on the shortest interval that contains \(1 - \alpha\) fraction of the central mass, an interval is produced that contains the densest part of the empirical distribution, but the left and right tail probabilities might differ. The other option is to set the left and right tail probabilities to be equal and consider the remainder to be the area inside the confidence interval. In this thesis all quantiles coincide with actual observed data points and the intervals are the shortest intervals that contain \(1 - \alpha\) fraction of the sample.

**NAIVE.QNT** confidence bands are fairly often used in practice, but, despite its popularity, the method has severe shortcomings. Most importantly, the band does not contain a \(1 - \alpha\) fraction of the observations. This follows directly from the fact that the coverage is computed pointwise leaving the simultaneous coverage uncontrolled. In other words, as the dimensions are treated independently, the method does not take the correlation structure
of the data into account at all.

### 3.7.2 Iterative Quantiles

The NAIVE.QNT approach can be improved by decreasing the quantile parameter $q$, until the in-sample coverage $\gamma_s = 1 - \alpha$ is reached. In other words, instead of using $\alpha/2$ we find $q < \alpha/2$ such that the desired in-sample coverage is achieved. Following the terminology of Publication II where this approach was introduced, we will refer to it as *minimum intervals* (MI). In practice, MI can be computed using standard machinery for finding the root of a function. Unlike NAIVE.QNT, MI can be applied for all values of $L$.

Previously Härdle and Marron (1991) and Kolsrud (2007) have described similar approaches where the univariate quantiles were widened by one observation at a time for both directions (up and down) to create a band that reaches desired in-sample coverage. These approaches differ from MI in that the confidence bands are symmetric with respect to the median (not shortest/highest density) and that the widening process preserves this symmetricity. The cost of having a symmetric band is that often the exact desired in-sample coverage cannot be reached as two observations are added each round.

Note that, despite using the highest density quantiles, MI does not find the band with the smallest total width. Depending on the data, the MWE band may be much more narrow, especially if the variance of the data varies a lot between dimensions.

### 3.7.3 Bonferroni Quantiles

The simplest way to add error rate control to NAIVE.QNT bands is to apply Bonferroni correction. This correction guarantees family-wise error rate control (FWER) by using a lower significance level $\alpha' = \alpha/M$ for each test. The divisor is the number of tests, in our application the dimension of the data $M$. We refer to this method BONF.QNT. It seems to be in use at least among economics time series forecasting, albeit often considered as a baseline to beat (Lütkepohl et al., 2015a,b; Staszewska-Bystrova, 2011; Winker et al., 2016).

Unfortunately, the applicability of this method is impaired by the fact that most datasets do not have enough data to reliably estimate the required quantiles $\alpha/(2M)$ and $1 - \alpha/(2M)$. Moreover, the FWER control enforced by the Bonferroni correction is known to be conservative, i.e., the
band could be narrower without violating the coverage target (Lütkepohl et al., 2015a; Perneger, 1998). This problem gets more severe if the $M$ tests are correlated.

### 3.7.4 Inverting a Multiple Hypothesis Testing Procedure

In univariate statistical testing it is quite common to derive confidence intervals for a parameter by inverting a statistical test (Cox and Hinkley, 1979). For example, one might derive a confidence interval for the hypothesis that a population mean (the parameter) is zero by mapping the acceptance region of the test statistic back to the actual parameter value. Similar reasoning can be applied to multiple hypothesis correction procedures: the task would be to find a region that corresponds to all parameters being within the null hypothesis acceptance range. A corresponding confidence band would be a band that encloses the region.

The Bonferroni corrected NAIVE.QNT confidence bands are an example of this. A simple correction to the pointwise intervals guarantees simultaneous FWER control over all $M$ variables. Another commonly used class of MHST corrections are step-wise procedures that operate on a set of observed $p$-values \{\(p_m : m = 1, \ldots, M\}\}. One well-known example is the Holm-Bonferroni method (Holm, 1979). Let \([p_{m_j} : j = 1, \ldots, M]\) denote an ordered sequence of the unadjusted $p$-values such that \(p_{m_1} \leq \cdots \leq p_{m_M}\). In the Holm-Bonferroni correction a threshold index \(j^* = \min\{j : p_{m_j} > \alpha/(M - j + 1)\}\) is determined and hypotheses corresponding indices \(j = 1, \ldots, j^* - 1\) are rejected. Unfortunately, this kind of procedure cannot be turned into a confidence band. A confidence band is required to be a static construct, whereas in the procedure above the rejection criterion depends on the observed data. There is no obvious way to transform a constantly changing rejection criterion into a confidence band. Additionally, many powerful step-wise procedures make strong assumptions about the independence or correlation relations between the tests (in our case the dimensions). Since all step-wise procedures involve similar difficulties, confidence bands corresponding to step-wise MHST methods cannot be found in the literature.

### 3.7.5 Multivariate Distance Measures

As noted earlier, observations of $M$ dimensional multivariate data can be seen as points in the vector space \(\mathbb{R}^M\). Hence a dataset of $N$ observations
forms a cloud of points in the vector space. As in the context of outlier
detection, the distance of a data point from the center of this cloud can be
used as a measure of extremeness. A natural definition is to select a set of
central observations and use their envelope to describe the joint distribu-
tion of the variables. This approach has been used by several authors over
the years. Kolsrud experimented with various distance measures (Kolsrud,
2007, 2015) as did Staszewska-Bystrova (2011). Also we compared against
these in Publication I.

The choice of distance measure is critical as different measures assume
the data to have different kinds of properties. If the data are nearly normal,
then the Mahalanobis distance is a good choice since it estimates the
correlation structure and measures distances in units that are independent
of the measurement scale, which might vary across dimensions. The
equidistance contours of the Mahalanobis distance are ellipsoids. If sphere
shaped regions seem appropriate, the $L_2$ norm, also known as Euclidean
distance, can be used. Moreover, the $L_\infty$ norm, also known as Chebyshev
distance, produces rectangular regions (Kolsrud, 2007, 2015). Note that
depending on the distance measure used the observations that become
marked as extreme may be quite surprising; e.g., a noisy curve among a
set of smooth ones can be spotted as outlying even though the individual
values would be typical for the dataset (see Figure 6 in Publication I).

### 3.7.6 Curve Distances

The analysis of vector valued data has given rise to a very dedicated sub-
field of statistics that treats curves as the data atom – functional data
analysis (FDA). Alternative formulations of various basic statistic concepts
have been generated to be able to compute on curves anything that is
computable on scalars. One important notion are measures of centrality
that, given a set of curves, describe which are typical curves within the
set and which not (López-Pintado and Romo, 2007, 2009; Miguez, 2016). Func-
tional data analysts speak of functional depth: curves with higher
depth are more central. Statistics such as h-modal depth (Cuevas et al.,
2006) and various band depths (López-Pintado and Romo, 2009) have been
used to measure it. Also the neighboring paths approach by Staszewska
(2007) fits into this category of methods as it chooses the paths to remove
based on $L_1$- or $L_2$-distance from an average curve. In essence, these
measures provide an ordering of the curves, which, in turn, can be used to
generate confidence bands (Goldsmith et al., 2013; Yao et al., 2005).
Figure 3.3. Bag plots and functional boxplots of the Milan temperature data using the R package *rainbow* (Shang and Hyndman, 2016). Plots on the left columns use bag plot conventions for marking outliers while the right column uses highest density region. Both operate on a two-dimensional PCA space. For the HDR plots the inner region covers 50% of the observations while the outer covers 99%. A large coverage was selected for the outer region to reduce the number of outliers (cleaner plot) and to account for the fact that when using MWE we end up at roughly 90% in-sample coverage when constructing the band with $K/N \approx 0.01$. See also Figure 2.1 and Publication I, Fig. 9.

One option to visualize the quantiles of the depth distribution is a *functional boxplot* (Hyndman and Shang, 2010; Shang and Hyndman, 2016; Sun and Genton, 2011). Figure 3.3 shows examples of this for the Milan temperature data of Publication I. Note that Figure 3.3 uses the R package *rainbow*, which computes the curve distances in two-dimensional principal components space (upper row), either using bag plot heuristics or HDR of a kernel density estimate (Shang and Hyndman, 2016). Due to differences in distance definitions, only year 2007 is identified as extreme both in Publication I and in Figure 3.3. As a curiosity, the two dots forming their own “island” in the top-right sub-plot are the remaining two outliers identified in Publication I (years 2003 and 2006). Note that, in principle, a functional boxplot could be constructed for any depth ordering of the curves. The bands in a functional boxplot are just envelopes of groups of curves and in that sense do not differ from confidence bands.
3.7.7 Rank Based Envelopes

Like multivariate distance measures, rank based envelopes construct confidence bands by excluding extreme observations from the dataset and defining the envelope of the remaining central observations to be the confidence band. The difference is that the latter measures the extremeness of an observation with two-dimensional rather than multidimensional measures.

One of the earliest references to these ideas can be found in the book by Davison and Hinkley (1997) in which independently created variable-wise confidence intervals were joined to form a confidence band. The authors point out the coverage problem and suggest a way to compute "an overall simultaneous two-sided error rate" in order to adjust the in-sample coverage. The error rate computation used the ranks of the observations within each data matrix column $X^{(\cdot, m)}$ to identify violations of the bands. However, Davison et al did not suggest an algorithm for finding a band with predefined coverage.

The first actual implementation of an algorithm to compute the bands as suggested by Davison and Hinkley (1997) was provided by Mandel and Betensky (2008). Their algorithm ranks the curves in the learning sample “according to the coordinate which is most discrepant from the pointwise medians”. These ranks are then used to define the “simultaneous confidence interval” by discarding the curves with the most extreme ranks and letting the envelope of the remaining data define the confidence band.

The band is claimed to control the FWER but this property seems to be asymptotic in nature, i.e., achieved only with very large $N$. The authors assume that the data come from an application where new samples can be drawn endlessly from the underlying distribution. Therefore no methods are given to achieve FWER in a fixed size sample.

One significant drawback of the method is that it does not scale nicely to multiple dimensions. As the dimension $M$ grows, ties in column ranks start appearing more frequently. To circumvent the problem of multiple observations having the same rank, the authors suggest ad hoc heuristics such as picking the extreme observation at random.

3.7.8 Minimum Width Envelope

From the theory of univariate confidence intervals we know that, among intervals of given coverage, the shortest one contains the densest part
of the data. Similarly for multiple dimensions the confidence band with the smallest area is centered around the densest parts of the data, when compared against all other confidence bands with the same coverage (Hyndman, 1996). We therefore argue that a minimum width confidence band is a band that is most suitable for finding interesting patterns in data and for performing graphical hypothesis testing.

We define a minimum width envelope (MWE) to be the smallest envelope that meets the required coverage criterion. In more exact terms, we are solving the following problem:

**Problem 1 (Minimum width envelope (MWE)).** Given a dataset $X$ with $N$ observations of dimension $M$ and integers $K$ and $L$, find a confidence band $B = (x_l, x_u)$ by minimizing $\sum_{m=1}^{M}(x_u(m) - x_l(m))$ subject to the constraint

$$\sum_{n=1}^{N} I[\text{Err}(x_n | x_l, x_u) \leq L] \geq N - K,$$

(3.9)

where $\text{Err}()$ is the error as defined in Equation (3.1).

The idea is to remove $K = \lfloor \alpha N \rfloor$ observations from the dataset such that the remaining set of $N - K$ observations has the smallest envelope possible. Curves inside the band violate the confidence band at most $L$ times per curve. The method is referred to as MWE.

Setting $L = 0$ gives the error criterion used in Publication I. This criterion and the greedy algorithm were motivated by the observation rank based procedure loosely outlined by Davison and Hinkley (1997), pages 150–154. Similar ideas have been presented by others as well. Kolsrud (2007) used the exact same criterion as did Staszewska (2007) (algorithm H2). Mandel and Betensky (2008) based their band on distributions of most extreme rank observations. Staszewska-Bystrova and Winker (2013) introduced MWE band search algorithm based on threshold accepting, which is a local search heuristic that somewhat resembles simulated annealing. However, their objective function was more complicated including also a penalty term. Lütkepohl et al. (2015a) used similar reasoning to adjust Bonferroni bands.

Recently Schüssler and Trede (2016) and Berg et al. (2017) have proposed exact solutions to the MWE problem based on constraint programming approaches. Schüssler and Trede (2016) solve the MWE problem for $L = 0$ and the solution is feasible for real life datasets. The search space for the MWE problem with $L > 0$ is too large for practical applications, but a column-wise restriction for the number of points outside the band reduces...
the search space (Berg et al., 2017). They also report that globally optimal solutions to the MWE problem are sometimes extremely narrow for some dimensions. This was also the motivation for constraining the number of outlying points column-wise. Constraint programming approaches deliver reasonably good solutions quite fast while most of the computation time is spent proving optimality. These methods depend upon specialized software and tuning them for maximal efficiency requires expert knowledge.

The novelty in Publications I and II is to give a computer science viewpoint to the problem by discussing and proving the hardness of the problem. Publication I provides a detailed greedy algorithm to solving the MWE problem while Publication II extends this to $L > 0$. In addition, an iterative quantile based algorithm is suggested in Publication II. R implementations of the algorithms are released as a public repository under https://github.com/jutako/multivariate-ci.git to facilitate a wider adoption of the methods.

In Publication II the FWER criterion $L = 0$ is relaxed such that the curves may exit the band in $L$ locations and are still counted as within-band. In other words, we let $L > 0$ which is analogous to the generalized FWER or $k$-FWER in statistics (Finos and Farcomeni, 2011; Lehmann and Romano, 2005; Romano and Wolf, 2007). The idea of $k$-FWER is to control the probability of observing $k$ or more false positives. The motivation is similar to the reasons why many bioinformatics applications control FDR instead of FWER. For example gene studies often involve hundreds of features and corresponding significance tests from which robust discoveries are to be made (Dudoit et al., 2003). False positives (type I errors) are not critical as the end result is just that a few unnecessary genes are given a closer look. In contrast, false negatives (type II error) are more critical, since all possibly useful genes need to be identified. Under these circumstances allowing some extra false positives is an acceptable choice, since it increases the chance of discovering all truly significant genes.

The increase of $L$ in the error criterion has a similar effect. It allows some dimensions to be outside the bands while requiring other to be within band. This can help in controlling the effect of spurious outliers, which would otherwise widen the bands unnecessarily. In the context of confidence bands a similar idea has previously only been presented by Wolf and Wunderli (2015).

Unfortunately, the problem of finding the minimum width envelope is an NP-hard combinatorial problem as shown in Publications I and II. Exact
global optimum might be possible to find for small datasets, but in general we have to resort to approximate algorithms. The greedy implementation in Publication I is a top-down approach that begins with the whole data envelope. Observations (rows of $X$) are removed one-by-one such that, upon removal, the width of the whole data envelope shrinks as much as possible. The process is repeated $K$ times and the envelope of the remaining observations define the confidence band. This basic approach can be used also when $L > 0$.

For the case $L = 0$ the computational complexity of this algorithm can be computed as follows. Constructing the helper data structure (see Appendix of Publication I) requires $O(MN \log N)$ time. Within this structure, finding the smallest and second smallest elements from each dimension can be performed in $O(1)$ time, and the operation to remove a curve from the structure takes $O(M)$ time. The removal of the $K$ curves can therefore be computed in $O(MK)$ time, resulting to a total time complexity of $O(MN \log N + MK)$ and memory requirement of $O(MN)$ for the whole algorithm.

The greedy algorithm performs well in practice. However, it can be shown that it does not provide any approximation guarantee. An adversarial example is given by a setup with $N = 5, M = 1, K = 2$, and a data matrix given by $x^T = (1, 1 - \epsilon/2, 2\epsilon, \epsilon, 0)$, with $\epsilon$ arbitrarily small. The optimal solution is given by $I_{opt} = \{3, 4, 5\}$ with cost $U(I_{opt}) = 2\epsilon$, while the greedy gives a solution $I_{alg} = \{1, 2, 3\}$ with cost $U(I_{alg}) = 1 - 2\epsilon$. In other words the greedy solution is unlikely to remove a set of two or more extreme curves that lie close to one another; it has no mechanism to “look ahead” and remove multiple curves at once reach a better solution.

### 3.8 Controlling Expected Coverage

Most methods described above produce bands with in-sample coverage of $\gamma_s = 1 - \alpha$. However, as described before in Section 3.6, several authors note that the true coverage often tends to be lower than the in-sample coverage, both for univariate confidence intervals (Burr, 1994; Clements and Taylor, 2001; Efron and Tibshirani, 1993; Kim, 2004; Royall and Cumberland, 1985; Stine, 1985) and for multivariate bands (see Publications I and II as well as Kolsrud (2007)). In other words, the bands do not generalize for unseen data from the same distribution. The reasons for this are outlined next together with possible solutions to overcome the issue.
All density estimation methods, including the confidence band methods above, suffer from a phenomenon called the *curse of dimensionality*. This refers to the observation that in many cases when generalizing a univariate analysis method to multiple dimensions, the amount of data needed to achieve a given accuracy of estimates increases exponentially with \( M \) (see Silverman (1986) for an example). In other words, as the dimension of the problem increases, so does the complexity of the joint distribution of the dimensions, and more data are needed to maintain accuracy. For example, in many methods the band boundary is restricted to observed data points and only the inclusion of new points can move the boundary and give a better estimate. Additionally, as the estimation concerns the tails of the distribution, new data points are not easy to get. When sampling, the probability of a new observation to hit the tails of the distribution is low and therefore the majority of new data points do not contribute to the accuracy of the band.

A standard approach to alleviating these effects is to reduce the dimensionality of the data. This is usually done either by selecting a subset of the most important variables (variable selection) or by transforming the data to a new lower dimensional basis that somehow better captures the essential relationships of the data. The latter is often called dimension reduction with a vast range of methods, such as principle components analysis (PCA), to choose from (Lee and Verleysen, 2007; Leskovec et al., 2014). However, these methods are not applied to compute confidence bands in this thesis, mainly because time series data are usually best interpreted in its original basis. Time series also exhibit strong autocorrelation structures, which significantly alleviate the problem.

A feasible number of dimensions for a given method depends on both the data and the estimation method. Some data types such as time series inherently suffer less from the curse of dimensionality, since the dimensions are correlated. To give an example, consider a uniformly distributed random variable \( x \sim U(0, 1)^M \) with \( M \) dimensions. The expected width of the envelope of a single dimension is \( (N - 1)/(N + 1) \) (Lemma 4.2 in Publication II) and thus \( N = (2 - \alpha)/\alpha \) samples are needed for this envelope to approximate a \( 1 - \alpha \) confidence interval. In an extreme case with perfect correlation, e.g., the series is \( x_n(m) = c_n \) for all \( m = 1 \ldots M \) and \( c_n \sim U(0, 1) \), the density estimation problem is effectively one dimensional and \( N = (2 - \alpha)/\alpha \) observations are needed to approximate a FWER controlling confidence band for the distribution. Using \( \alpha = 0.1 \) this yields
$N = 19$ regardless of $M$. On the other hand, in Publication II it is shown that if the dimensions are uncorrelated, $L = 0$ and we assume all $M$ intervals to be of the same width then $N \approx 2M/\alpha$ observations are needed. For $M = 100$ and $\alpha = 0.1$ this results into $N \approx 2000$ samples!

If we allow some dimensions to exit the band ($L > 0$), fewer samples are needed to achieve the same accuracy. In Publication II it is shown that if we let $L = \lfloor \beta M \rfloor$ then the number of observations required in the uncorrelated case above drops to $N \approx 2/\beta$. With $\beta = 0.1$ this gives only $N \approx 20$. Hence allowing $L > 0$ leads to more accurate bands at smaller sample sizes. In addition to the analytical examples, this was also observed for multivariate normal data in Publication II.

Besides dimensionality, also the use of data affects the generalizability of the bands. In the methods above, the observed data have a dual role: it is used both in computing the bands and for testing their coverage. This essentially overfits the bands to the observed data causing the true coverage to be smaller than intended. This problem is considered next.

The solution to the coverage problem depends on whether new samples are easy to get. If new samples can be easily generated, the coverage can be made more accurate by simply using more data. Also the estimation of expected true coverage is straightforward, since a test dataset can be generated whenever needed. On the other hand, if new samples are not available or the increase of sample size causes running time issues, several resampling based approaches have been suggested.

Kolsrud (2007) introduces four different resampling methods for the estimation of expected coverage, all of which are adaptations of the respective univariate methods (Efron, 1983; Efron and Tibshirani, 1993). First of these is leave-one-out cross validation, in which one observation at a time is left out of the learning sample and used for testing. Kolsrud notes that the method is unbiased but has large variability (Efron, 1983; Kolsrud, 2007).

The second method is ordinary bootstrap, where $b$ new datasets $X^*_b$ are sampled with replacement from $X$, confidence bands computed and coverage tested using $X$. This method has low variability but has bias due to the double use of data in both training and testing (Efron, 1983; Kolsrud, 2007).

To reduce bias, the third variant combines the two previous ones creating a scheme traditionally known as $k$-fold cross validation. The data are divided into $k$ subsets and each subset is used once as a test set while the
rest serve as the learning sample. Also Publication I uses this method

to estimate the expected coverage. As an improvement over the existing
literature we also give an estimate of an effective in-sample coverage $\gamma_s'$ to
use in order to achieve $\gamma_e \approx 1 - \alpha$.

However, univariate studies show that even the $k$-fold cross validation
has bias (Efron, 1983; Efron and Tibshirani, 1997). Inspired by these
results Kolsrud (2007) suggests yet another correction, the so called
$.632+$ bootstrap estimator. However, both Efron and Kolsrud note that
despite good actual performance the theoretical justifications behind $.632$
and $.632+$ estimators are weak and more research is needed to verify the
results (Efron, 1983).

We will now move on to explaining how cross validation is used in esti-
mating the expected coverage and in adapting the number of observations
to remove.

3.8.1 Cross Validation Procedure

This section explains how $J$-fold cross validation can be used to estimate
an effective number of observations to remove, $K'$, such that the desired
expected coverage $\gamma_e \approx 1 - \alpha$ is reached. We speak of $J$-fold cross validation,
to avoid notational clashes with the index $k$ of the MWE method. The
process starts by dividing the dataset into $J$ subsets (folds) where $J \leq N$.
For each fold, data in the current fold $j$ are used as test data while data
from the other $J - 1$ folds are used to construct a sequence of confidence
bands with a variable number of curves to remove $k = \{1, \ldots, \alpha K\}$. The
output for each fold is a coverage profile $\phi_j(k)$ that indicates the observed
in-sample coverage $\gamma_s$ within the test set $j$ at any given $k$. After computing
these profiles for all $J$ folds the profiles are averaged to give an overall
coverage profile $\bar{\phi}(k)$. The subscript underlines the fact that the result
depends on the number of folds $J$ used. By finding the largest $k$ for which
$\bar{\phi}(k) \geq (1 - \alpha)$, we get an number of observations $K'$ to remove, such that
the expected true error rate stays controlled at level $\alpha$.

This procedure is a generic solution that can be applied to all of the
methods described above. For quantile methods that use $\alpha$ instead of $K$ the coverage profile just becomes a function of $\alpha$. Also the error rate
criterion can be freely changed. A MWE band that is computed using $K'$ is
referred to as MWEC.

Figure 3.4 shows confidence bands for the ECG data. The top part is
computed without corrections whereas the bottom part has been computed
using the cross validation procedure just described. Note how in the top part NAIVE.QNT is too narrow and MAHA produces a qualitatively very different band. After the procedure (bottom) all bands resemble each other and are actually very close to the data envelope.

As the process is computationally quite heavy, it makes sense to limit the span of $K$ (or $\alpha$). Since we know that setting $K = \lfloor \alpha N \rfloor$ produces bands that are already a bit too narrow, we do not need to test further than that. Note also that using smaller subsets provides more accurate results. A detailed description of the process is given in Publication I.
**Figure 3.4.** Different confidence bands for the ECG data. This figure is a zoomed-in version of Figure 2.1. **TOP:** Bands with in-sample coverage of 90%. Notice how NAIVE.QNT is always narrowest whereas MWE and MI are close to each other. **BOTTOM:** Bands with expected coverage of 90%. MWE and MI are shown for reference – they do not control expected coverage. BONF.QNT ends up excluding only one observation from each tail and is clearly the most conservative. Other bands are very similar, partly because there is not enough observations for precise estimation. For MWEC four cross validation folds reduce $K$ from 45 to 34.
Datasets in which the observational unit is a vector valued quantity rather than a scalar are common. Examples include time series analysis, spectral analysis and several applications of biosignal analysis. Because the observations are multivariate, confidence bands provide an intuitive way of visualizing the data. We will next present two applications for confidence bands: finding structure in interval sequences and the analysis of evoked response potentials of the brain.

4.1 Finding Structure in Interval Sequences

The first application comes from Publication IV, where we study the structure of interval sequences using a variety of randomization methods. An interval sequence is an ordered sequence of interval lengths. Consider a sequence of $N + 1$ events that occur at times $t_0, t_1, \ldots, t_N$, where $t_0 \leq t_1 \leq \ldots \leq t_N$. An interval sequence of length $N$ is defined by

$$S = (x_0, x_1, \ldots, x_{N-1}),$$

where $x_n = t_{n+1} - t_n$. In Publication IV we use the logarithmic interval sequence $S_z = (z_0, z_1, \ldots, z_{N-1})$, where $z_n = \log x_n$. The logarithmic scale is appropriate for two reasons: without it long intervals would receive larger weight in the analysis, and logarithms of intervals can take any value, including negative, which is numerically convenient.

4.1.1 Randomization Methods

Publication IV defines several distributions of interval sequences, and the respective randomization methods. Each of the distributions preserves some aspect of the original sequence.
Uniform Randomization. The UNIFORM distribution is a uniform distribution over all sequences of $N$ intervals in which the duration is fixed to $t_N - t_0$.

Interval Randomization. The INTERVAL distribution is a uniform distribution over all permutations of sequence $S$. A sample $S^*$ from the INTERVAL distribution can be drawn by permuting $S$ uniformly in random.

Fixed Subsequence Randomization. The SUBSEQUENCE distribution is a uniform distribution over all permutations of the sequence $S$ where a given subsequence $G_x \subseteq \{0, \ldots, N-1\}$ of the intervals is kept fixed. A sample $S^*$ is obtained by permuting all intervals in $S$ that are not in $G_x$ uniformly in random.

Fixed Fourier Parameters Randomization. The FOURIER distribution is a distribution of interval sequences in which given subsets of Fourier amplitudes and phases have been fixed. The FOURIER distribution is obtained by fixing a subset $G_c \subseteq \{1, \ldots, K\}$ of Fourier amplitudes $c_k$ where $k \in G_c$, and a subset $G_\varphi \subseteq \{1, \ldots, K\}$ of Fourier phases $\varphi_k$ where $k \in G_\varphi$. A sample $S^*$ from FOURIER is obtained by first taking a sample $S'^*$ from the INTERVAL distribution, and then replacing the Fourier amplitudes $c_k$ and Fourier phases $\varphi_k$ not in $G_c$ and $G_\varphi$, respectively, by the respective Fourier parameters of the sample $S'^*$. The sample $S^*$ is then obtained by applying the inverse Fourier transformation to the randomized Fourier parameters.

Fixed Distance Function Randomization. The DISTANCE randomization method approximately preserves any arbitrary constraint. We define the constraint by a distance function $d(S')$ which is a non-negative function of permutations of the original interval sequence and zero for the original non-permuted sequence, $d(S) = 0$. We define a distribution DISTANCE by $f(S') \propto e^{-d(S')}$, where $S'$ is a permutation of the original interval sequence $S$. A sample from the distribution $f$ is likely to include intervals which are close to the original interval sequence in terms of the distance function. A sample $S^*$ from DISTANCE can be obtained via Markov chain Monte Carlo (MCMC) integration (see Publication IV and Henelius (2017) for details).

These randomization methods are essential in finding interesting patterns: they provide a versatile set of distributions that can be used as null models when assessing the significance of patterns, e.g., with confidence
4.1.2 Confidence Intervals and Hypothesis Testing

Confidence intervals cannot always be determined analytically (e.g., for Fourier amplitudes, Fourier phases or the autocorrelation structure) and must hence be obtained by simulation. Given a (null) distribution to sample from, confidence intervals for any arbitrary statistic can always be computed by drawing several samples, computing the value of the statistic for each sample and using this sampling distribution to derive intervals. The confidence intervals in Publication IV are defined to be the $\alpha/2$ and $1 - \alpha/2$ quantiles of the sampling distribution, where we always use $\alpha = 0.05$ as the significance level. Observed statistic values that are outside the confidence intervals of the chosen null distribution are significant, i.e., their value is not explained by the null model. These represent the interesting structure in the data.

The statistics studied in Publication IV are all vector valued and hence confidence bands can be used. Publication IV uses naïve quantiles for computing confidence bands. In what follows we compare these to the FWER controlled bands introduced in Publication I.

4.1.3 Example: Words in a Book

This example shows how the INTERVAL randomization can be used to break all temporal structures in the data, while maintaining the values $x_n$ of the original sequence. Temporally significant patterns are detected by comparing Fourier cosine amplitudes from the original sequence $S$ to the amplitude distribution of the INTERVAL randomized data $S^\ast$. Maintaining the original values is important as, e.g., a single deviant value $x_n$ causes low frequency activity to the power spectrum. The purpose of INTERVAL randomization is to find out which spectral components are due to the ordering of the data and which are just artifacts of the value distribution.

An example is shown in Figure 4.1 for the word “met” from the book dataset (see Publication IV). The amplitude distribution related to the INTERVAL randomization $S^\ast$ is visualized as a confidence band and observed amplitudes from $S$ are overlaid on top (black dots). Figure 4.1 is a re-analysis of Publication IV Figure 1 (without matching random number generation).

In order to use Figure 4.1 to make robust decisions about which observed
Fourier amplitudes $c_k$ differ from typical null model values, a confidence band with known coverage properties is needed. For example, the observed amplitude at $k = 18$ seems significant with respect to the naïve quantiles but not in comparison to the MWE bands. As this example illustrates, a researcher using only naïve quantiles makes more rejection errors than one using coverage controlled methods.

Note that there are two versions of naïve quantiles confidence bands in Figure 4.1. The NAIVE.QNT.S corresponds to a symmetric tail definition of a quantile used in S-Plus and in Publication IV, whereas NAIVE.QNT uses the shortest interval that covers $(1 - \alpha)$ fraction of the data. The difference is clear: the shortest interval is indeed much shorter.

### 4.2 Analyzing Single-Trial ERPs

The second example comes from neuroscience where the electrical activity of the brain is measured using electroencephalography (EEG). One subfield of EEG research is the analysis of event-related potentials (ERPs). These are brain waves that occur in response to some stimulus and they convey information related to the neural processes associated with the stimulus and the task at hand.

The example comes from a study where stress related burn-out symptoms
Applications of Confidence Bands

were studied using cognitive tests. The subject performed a mismatch negativity (MMN) auditory ERP protocol, where the task is simply to passively listen to sounds while watching a video (Pakarinen et al., 2014; Sokka et al., 2014). No reaction to the sounds is required. The goal is to study how the MMN ERP response changes when characteristics of the stimulus (such as frequency, amplitude, etc.) are changed.

ERPs were extracted from the data by extracting data segments from \(-100\) to \(800\) ms around the stimulus giving on average 160 single trial ERPs for the standard stimulus. An example of the ERP from a single subject from channel Cz is shown in Figure 4.2, along with various 50% confidence bands. A low coverage was selected to focus on the most central observations assuming that they have a higher signal to noise ratio. It is evident from the Figure 4.2 that the signal-to-noise ratio in an ERP is very low. In fact, even the very narrow naïve 50% confidence band contains the \(y = 0\) line throughout, although the distribution is slightly skewed towards the positive y-axis. The bands also reveal that signal variation is lower before time zero. This is deliberate, since the ERPs have been baseline corrected to have zero mean in the \(-100\ldots0\) ms range, as customary in ERP analyses.

In this example the confidence bands reveal that single subject ERP averages are to be interpreted with extreme caution. It further makes clear why it is customary to average the trials within each test subject and again over all test subjects to produce a group average ERP. These averages reflect activity that remains constant across trials while noise gets averaged out.
Figure 4.2. Single-trial mismatch negativity ERP waves from a single subject from channel Cz. The average ERP is shown in black and a 10 % sample of the single-trial ERPs in gray. Four different 50 % confidence bands are shown in color, computed as in Fig. 4.1. Remember that only the MWEC band controls the expected coverage while the others are too narrow.
5. Shared Variation Between Datasets

Publication III presents an algorithm for the extraction of shared variation between datasets. It is difficult to position this work into any of the existing data analysis genres as the problem setup resembles that of multi-view learning, but the modeling principle, based on redundancy analysis (RDA) and chained regression, does not fit into the existing categories. However, many of the subspace learning methods in multi-view learning can be used to produce similar results. Especially factor analysis provides a well studied reference point for comparison. To set the COCOREG algorithm of Publication III into perspective, we will next first introduce multi-view learning with focus on the problem setup and subspace learning methods. The chapter then continues by describing the COCOREG algorithm and concludes with examples.

5.1 Multi-View Learning

Multi-view learning refers to a recent trend in machine learning in which multiple feature sets are used to analyze a single set of objects (Sun, 2013). The feature sets, usually referred to as views, correspond to different measurement modalities or information sources. For example, an image is characterized both by its pixels and its caption or other textual environment. This gives us two views: the raw pixel content in one and the related text in the other. Another example is the human brain, which can be measured using different imaging modalities such as EEG, magnetoencephalography (MEG), and functional magnetic resonance imaging (fMRI) providing three views. In all cases, an object of interest is given several alternative descriptions using different sources of information. The views complement each other and together they provide a more comprehensive understanding about the objects than a single view does.
A view is a data matrix where rows correspond to objects and columns to different features. Similar features are grouped together in a single view and the set of all available views constitute a data collection. Given a data collection with i.i.d. sampled rows a typical task is to classify the objects (rows) using the features (columns). For example, one might want to classify a set of images into classes “scenery” and “portrait” based on both the pixel content as well as the image caption.

The problem setup of COCOREG is a variant of this basic setup with time series as the feature vectors (columns). The rows of each view now index time and are correlated rather than i.i.d. sampled. Within this setting a typical research question is to determine a new set of time series that somehow capture the essential structure present in the data. An example would be an experiment with multiple participants, where the physiological responses of the subjects are organized into the views, one view per participant, rows indexing time. In this case the rows of the views are correlated; both within and between views. Since all views describe the same experiment, it is reasonable to assume that they share patterns as well. By studying these patterns we get information about the experiment, but less about the participants.

Note, however, that the results of this physiological experiment could have been presented such that the views have i.i.d. rows. If time is averaged out by computing some aggregate feature of the signal, the resulting data can just as well be organized such that subjects map to rows and the aggregate features to columns. Each view could contain certain types of aggregate features, for example one for ERPs and another for the signal power spectrum. For data organized this way the research question might be, e.g., to perform an unsupervised classification on the participants. This provides information mainly about the participants and much less so about the experiment. In sum, the way the data are organized into views affects the range of questions we are able to address.

Following the classification by Xu et al. (2013) we classify the multi-view learning methods into three broad categories: co-training, multiple kernel learning and subspace learning. Each of these is briefly reviewed next to give an overview of the field.

5.1.1 Co-Training

Co-training has a relatively short history dating back to Blum and Mitchell (1998), who used it to classify web pages by combining information on the
page with information found on links pointing to the page. The idea is to use two views of data to complement each other in situations where there exists little labeled data but large amounts of unlabeled data. The success of co-training depends crucially on whether the views are independent or not (Nigam and Ghani, 2000). Independence is needed for the views to complement each other, without it the method works poorly (Krogel and Scheffer, 2004).

5.1.2 Multiple Kernel Methods

Multiple kernel learning refers to using several kernel methods, each with different kernel, for solving a single multi-view problem (Gönen and Alpaydın, 2011). The advantage is that kernels are easy to combine but, at the same time, the type of the input data can vary between the kernels. Hence multiple kernel learning provides a natural framework for data fusion.

Multiple kernel methods are based on ordinary kernel methods, such as support vector machines (SVMs), that learn classification rules for single datasets (Cortes and Vapnik, 1995). These methods are able to learn non-linear decision boundaries by projecting the data into a higher dimensional space, where linear separation is possible. As it turns out, neither the actual coordinates of the data after projection nor the projection function itself are ever needed. The learning is based on computing similarities between data points in the higher dimensional space, but, for many projections, these can be easily computed in the original space as well. The similarity function, expressed using coordinates of the original data space, is called a kernel. Multiple kernel learning combines several, possibly different, ordinary kernel methods into a larger model. A common approach is to use a linear combination of kernels as the target function to optimize. This enables data fusion, since a different type of kernel can be used for each view and yet the output of all kernels is a similarity score that can be compared against other similarity scores (Yu et al., 2010).

5.1.3 Subspace Learning

In subspace learning the goal is to reveal interesting patterns by finding subspaces of the original data, in which the patterns become clearly visible. In many cases, this new representation is of lower dimension than the feature space the data were measured in. Many familiar and well-known
analysis methods such as principal component analysis (PCA) (Hotelling, 1933), canonical correlation analysis (CCA) (Hotelling, 1936) and factor analysis (FA) (Thurstone, 1931) belong to this category.

In practice an interesting pattern is often a correlation between the views of the data collection. In general the original feature space is a poor choice for finding patterns, since usually some combination of features gives a higher correlation between the views. To give an intuition of what is happening, we will next briefly describe the process of combining features.

The feature space of a dataset is defined by the columns of the data. Each column contains data from one feature which are either directly measured or computed from a raw signal and organized into a matrix. The columns define a set of coordinate axes - the original feature space. However, any linear combination of the original features defines a direction in this vector space and this can be used to define a new coordinate axis. A popular choice is to search for orthogonal directions to get uncorrelated coordinate axes.

To get the coordinate values for each observation along a new axis, we must compute the orthogonal projections of the data points onto the axis. Given a unit vector in the direction of the new axis, the coordinates are given simply by vector dot product (matrix multiplication). Figure 5.1 shows an example of redefining coordinate axes using PCA, which defines a new set of coordinate axes such that the variance of the data along each axis becomes maximized. By comparing the red and the blue dot it can be seen that the relative positions of the data points remain unaltered even though the coordinate values change.

To conclude, multi-view learning has been successfully applied to a wide variety of machine learning problems including dimensionality reduction, clustering, supervised learning, active learning, transfer learning and ensemble learning (Sun, 2013). These problems are encountered in a wide range of application domains such as neurophysiology (Remes et al., 2016), high-throughput data in genetics (Ritchie et al., 2015), computer vision (Su et al., 2009), natural language processing (Dhillon et al., 2011), image classification (Hardoon et al., 2004; Zhu et al., 2016) and web page classification (Blum and Mitchell, 1998). In fact, it has been observed that even when the data naturally consist of only one view, the generation of artificial views can improve performance (Sun, 2013).

The novel approach in Publication III is closest related to subspace learning and therefore existing subspace learning methods are reviewed.
in more detail in Chapters 5.3 and 5.4.

5.2 Notation and the General Shared Variation Problem

We are interested in a collection of \( K \) sets of features which we call views. This collection of \( K \) views is referred to as a data collection. The \( k \):th view is a \( N \times M_k \) data matrix \( X_k \in \mathbb{R}^{N \times M_k} \) with \( N \) observations of \( M_k \) features. A row of features in \( X_k \) is marked \( X_k(n, \cdot) \) and a column of observations for feature \( m \) as \( X_k(\cdot, m) \). The \( n \):th observed data point in the \( m \):th column is denoted \( X_k(n, m) \), which is a scalar. In this work the focus is on time series data and hence \( n = 1, \ldots, N \) indexes time. The number of features \( M_k \) may vary from view to view.

5.2.1 Shared Variation

Our interest will be in examining to which extent the views share variation. By variation we mean the fluctuations of \( X_k(n, m) \) around the mean \( X_k(m) = \sum_{n=1}^{N} X_k(n, m) / N \). To gain insight into what is meant by shared variation, let us look at the definition of correlation:

\[
\text{corr}(X, Y) = \frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y} = \frac{\mathbb{E}[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y},
\]

(5.1)

where \( X \) and \( Y \) are random variables, \( \mu \) the distribution mean and \( \sigma \) the standard deviation. Equation (5.1) essentially quantifies to which extent variables \( X \) and \( Y \) deviate simultaneously from the mean, scaled to the interval \([-1, 1]\). Both deviations to the same direction (positive correlation)
as well as to opposite directions (negative correlation) count as correlation. The same is true for shared variation: simultaneous fluctuations that consistently happen in either same or opposite direction count as shared variation.

While for the random variables \( X \) and \( Y \) correlation is a fixed quantity, the shared variation for \( X_k \) with respect to the rest of the data collection may not be constant over time. Consider a data collection consisting of views that contain the physiological response of a test subject, one view per subject. If at time instance \( n_1 \) the subjects all experience a powerful stimulus, there might be a lot of shared variation between the views. On the other hand, at time instance \( n_2 \) the subjects might be in rest and not interacting with each other, causing the views to share no variation at all. Consequently shared variation is a quantity that changes over time.

As will be shown in detail later, all methods discussed in this chapter have some notion of shared variation attached to them. For CCA the projection of the views onto the directions of largest correlation define the canonical components, which quantify one form of shared variation. Another definition is used in factor analysis and related methods, where a basis of unobserved latent variables is used to model the correlation structure of the views. These latent variables represent variation that is, in varying degrees, shared between all views. The definition of shared variation used in Publication III is an extension of the one used in RDA, in which linear regression is used to define redundancy – also a form of shared variation. According to the RDA definition, variation in view \( A \), that can be explained using view \( B \) and a linear model, is considered shared (see Definition 16 in Section 5.5.2 below for details).

In general the total variation of a view can be divided in three parts: shared-by-all, shared-by-some and private. The shared-by-all variation is common to all views of the data collection, whereas shared-by-some is only common to some subset of the views. By private variation we mean variation unique to a single view only. The following notation will be used: the variation in view \( X_{k_1} \) that is shared between another view \( X_{k_2} \) is marked \( X_{k_1,k_2}^{all} \). The variation in \( X_{k_1} \) that is shared with all other views \([K] \setminus k_1 \) is marked \( X_{k_1}^{all} \), where \([K]\) is used to denote the set \( \{1, \ldots, K\} \).

We consider the problem of dividing the variation of the view \( X_k \) into two parts: one that contains variation that is shared with all of the other \( K \) views \( X_k^{all} \) and another that contains rest of the variation \( X_k^{residual} \). Together these sum up to the whole view \( X_k = X_k^{all} + X_k^{residual} \). In this
notation \( X_k^{\text{residual}} = X_k^{\text{shared-by-some}} + X_k^{\text{private}} \).

The general shared variation extraction problem becomes:

**Problem 2** (Shared variation extraction). *Given a data collection with \( K \) views \( X_k \), extract from each of the views \( X_k, k \in [K] \), the variation shared by all views \( X_k^{\text{all}} \).*

This definition applies to all methods and therefore the definition of *shared* is not exact. Method specific definitions will be given as the methods are introduced. However, all of the definitions are consistent with the fact that high correlation implies a large fraction of shared variation.

### 5.3 Maximally Dependent Projections

Matrix projections are an established and well researched class of multi-view learning methods. This section introduces two basic formulations of canonical correlation analysis (CCA) that can be used to extract shared variation.

#### 5.3.1 Canonical Correlation Analysis

Canonical correlation analysis (CCA) was first introduced by Hotelling (1936) as a method to find correlation maximizing projections between two views. The idea is to find a linear combination of columns (features), one for each view, such that the correlation between the resulting one-dimensional projections is as large as possible. Once a pair of projections has been found, their contribution to the data variation is subtracted from the data and the process starts over. As the input data matrices are in general of different rank, the process ends when the lower rank matrix has no free dimensions left. The output is a sequence of orthogonal canonical directions that represent the data from a shared correlation viewpoint. In other words, CCA creates a new set of coordinate axes for both views such that the correlation between the coordinate axes is maximized.

Over the years several variants of the basic procedure have been introduced, typically by either replacing the linear projection with a non-linear alternative (e.g., Hardoon et al. (2007)) or by changing the dependency measure (e.g., Dehon et al. (2000)). However, the main limitation of the standard CCA approach is that the input is limited to two datasets. A basic trick to overcome this limitation is presented next.
5.3.2 Multi-Set CCA

Multi-set CCA (MCCA) methods extend CCA to situations with more than two views (Hwang et al., 2013; Kettenring, 1971; Tenenhaus, 2011; Tenenhaus et al., 2015). The underlying principle remains the same but maximization criterion is changed to account for the greater number of views. Instead of maximizing a single correlation, a sum of correlations forms the maximization cost function. The result is a set of canonical directions that maximize the average correlation among the views.

If all views share some variation with all others, the MCCA approach is reasonable: the selected direction is the one that gives, on average, maximal correlation with others. However, if one view happens to be completely independent of the others and therefore there is no shared variation, the result can be misleading. To see why, consider a set of three views A, B and C, where C is independent of A and B. The question is whether the views contain variation that is shared between all of the views? The correlation sum to be maximized contains all pairs of views, i.e., \( \{A, B\} \), \( \{A, C\} \) and \( \{B, C\} \). Since C is independent, only the pair \( \{A, B\} \) contributes to the maximization. Hence the resulting canonical directions are arbitrary for C (depending only on small by-chance correlations) and close to the two-view CCA solution for A and B. Hence the result, a set of canonical directions and an average correlation, does not reveal the fact that the amount of variation shared by all views is zero. However, in many applications we would like to find not average correlations but the smallest common correlation between the views. For these situations other methods than MCCA are needed.

5.4 Variance Factorization

Another class of subspace learning methods are those that aim at factorizing the whole data variance. The most common approach is to assume that a set of unobserved latent variables exist and that a linear combination of these produces the observed data. The task is to search for a suitable set of latent variables and associated combination weights. Models of this kind are members of a broader class called latent variable models (LVMs). Any model with unobserved variables can be counted as an LVM. In what follows, we focus on LVMs that have continuous observed and unobserved variables.
Due to advances in computational methods, it has become mainstream to analyze LVMs under a probabilistic or, equivalently, Bayesian framework. To simplify the notation of this section and to highlight similarities between methods, we use the probabilistic notation throughout. Fortunately, many widely applied methods, such as principal component analysis (PCA) and factor analysis (FA), have well studied probabilistic interpretations. We are interested in a generative model (Bishop, 2006):

\[
x_n = z_n W + \mu + \epsilon_n
\]

(5.2)

where \( x_n = [X_1(n, \cdot), \ldots, X_K(n, \cdot)] \) is a row vector of observed features from all views for sample index \( n \), \( M = \sum_k M_k \), \( z_n \) a vector of latent variable scores, \( W \) a matrix of mixing coefficients, \( \mu \) the mean vector (can be subtracted) and \( \epsilon_n \) a noise variable distributed as \( N(\epsilon|0, \Psi) \). The number of latent variables \( P \leq M \) is a parameter of the model. The idea is to model dependencies between observed variables using \( zW \) and let the noise part \( \epsilon \) take care of the variable specific variation. The rows of \( W \) define the new coordinate basis for the data. In factor analysis literature they are called factor loadings and whereas the diagonal elements of \( \Psi \) are called uniquenesses (Bishop, 2006).

### 5.4.1 Restricting the Model

The model defined in Equation (5.2) is unidentifiable. This means that there are infinitely many solutions to Equation (5.2). Reasons for this are two-fold: rotational invariance and ambiguity of associating variable to factors (Bishop, 2006; Klami et al., 2015).

By rotational invariance we refer to the fact that the solution is redundant with respect to rotations of the latent space coordinates (Bishop, 2006). In other words, there are infinitely many matrices of the form \( \bar{W} = WT \), where \( T \) is an orthogonal matrix, that result in the exact same covariance matrix for the marginal distribution of \( x \). Consequently many analysis methods have factor rotations as a post-processing step.

The other problem is related to how \( W \) and \( \Psi \) are used to model the data. In general, it would be nice to have at least two types of latent variables: those that are common to all views and those that contribute only to a subset of the views. In addition, the model should contain some view-specific private variation and room for noise. The problem is that a shared-by-all factor can be replaced by a set of view-specific latent variables...
Shared Variation Between Datasets

if the model is flexible enough (Klami et al., 2015). Hence restrictions need to be posed on both $W$ and $\Psi$ to end up with useful models.

Let us first consider a model where restrictions are posed only on noise covariance $\Psi$. Klami et al. (2015) lists three options of which the least useful is to pose no restrictions at all, which would cause some dependencies between the views to be modeled as noise. Forcing $\Psi$ to be diagonal, gives rise to a probabilistic version of FA. Further restricting $\Psi = \sigma^2 I$ amounts to probabilistic PCA.

We move now on to models where both $W$ and $\Psi$ are restricted in some way. Note that it is not fruitful to restrict only $W$ as that would only cause some of the dependencies between views to be modeled as noise. Instead, $\Psi$ is kept diagonal to let the latent variables model the dependencies. The models that follow extend factor analysis into the multi-view learning setting by explicitly modeling view dependencies using the latent variables.

A classical example is Multi-Battery Factor Analysis (MBFA) (Browne, 1980; McDonald, 1970). MBFA allows two types of latent variables to exist: those that are shared by all views and those that are private to a single view only.

A more complex model is defined by Inter-Battery Factor Analysis (IBFA). In addition to what MBFA provides, IBFA also explicitly models latent variables that are shared by a subset of the views. IBFA was first introduced by Tucker (1958) in the context of experimental psychology, but the method has been rediscovered in machine learning several times since (Ek et al., 2008; Klami and Kaski, 2006).

Group Factor Analysis (GFA) by Klami et al. (2015) provides a recent, fully probabilistic, linear solution to the IBFA modeling problem. A probabilistic, non-linear alternative is given by Damianou et al. (2012) in their Manifold Relevance Determination (MRD) algorithm. A recent extension of MRD uses Gaussian process priors to select dimensionality of the latent subspace (Damianou et al., 2016). We focus next on the GFA, which was also used as the primary method of comparison in Publication III.

### 5.4.2 Group Factor Analysis

Group factor analysis (GFA) by Klami et al. (2015) is a recent probabilistic solution to the IBFA problem. A “structural sparsity prior” is placed on the mixing matrix $W$ in Equation (5.2) to specifically search for sparse solutions and by explicitly modeling the relationships between the views. A sparse solution is such that not all latent variables load onto all views.
The noise covariance matrix $\Psi$ is kept diagonal (as in factor analysis) with unique variances for each view.

The GFA model provides detailed information on how the variance is shared between the views. A factor can be active in only one view, in some subset or in all of them. However, defining “active” requires one to set an arbitrary threshold to define how much a factor needs to contribute in order to be counted as active.

We can use this threshold to define those factors that represent variation shared by all views. If the threshold is set very low, almost all factors count as being active in all views and, as a result, most of the data variation gets explained by the model. On the other hand, raising the threshold a bit might rule out many factors changing the view entirely.

### 5.5 Regression Chain Filtering (COCOREG)

As pointed out above, the existing methods involve various problems when used to extract variation shared by all views in a data collection. MCCA provides only averaged results that may hide the fact that one of the views shares nothing with the rest. Variance factorization delivers more precise information, but at the cost of increased model complexity and time-consuming training phase. Recent probabilistic extensions of factor analysis, such as GFA, are able to explicitly model the relationships between views, again at the cost of even more model complexity and very slow convergence in training. Due to model complexity, adapting the models to the specific needs of a research question at hand is difficult and requires considerable expertise.

To strike a balance among these limitations we would like to have an algorithm that has the following properties:

a) if one of the views is independent of the others, the output should be constant

b) if the views are perfectly dependent and there is zero noise, the output should reconstruct the data without error

c) if the views are partially dependent, the output should suppress variation that is not shared by all views

d) it should utilize standard tools for modeling dependency

e) it should be possible to change the dependency model without re-
designing the algorithm

Criteria (a) – (c) state the desired behavior at the extreme cases of dependency and under normal conditions. Criteria (d) – (e) express the wish to find a simple model that builds upon existing work (e.g., regression models).

5.5.1 Unidentifiability of Shared Variation

Before describing an algorithm that has these properties, let us take a closer look into the particular definition of shared variation used in COCOREG. In the case of two views we would like to divide the views into two parts:

\[ X_1 = X_1^{[1,2]} + X_1^{[1]} = A_{12}z_{12} + A_1z_1^{[1]} \]
\[ X_2 = X_2^{[1,2]} + X_2^{[2]} = A_{21}z_{12} + A_2z_2^{[2]} \]  
(5.3)

where \( X_1^{[1,2]} \) is the variation in \( X_1 \) that is shared and \( X_1^{[1]} \) is the residual variation. The latter part of the Equation (5.3) shows the latent variable representation with latent variables \( z \) and related loadings \( A \).

Intuitively, the shared component of view 2 is given by \( \hat{X}_2^{[1,2]} = B_{12}X_2 = R_{12}R_{22}^{-1}X_2 \) and the residual component by \( \hat{X}_2^{[2]} = B_{21}X_1 = R_{21}R_{11}^{-1}X_1 \). However, a problem with this construct is that the partition is not unique. Variance may be freely shifted between the components of Equation (5.3) without any change to the covariance matrix of the data collection \( X = [X_1, X_2] \) (see Publication III).

5.5.2 Problem Definition

To solve the non-uniqueness problem COCOREG borrows and extends a definition for shared variation from redundancy analysis (RDA) (Legendre and Legendre, 1998). In RDA the shared variation components of views 1 and 2 are

\[ \hat{X}_1^{[1,2]} = B_{12}X_2 = R_{12}R_{22}^{-1}X_2, \]  
(5.4)
\[ \hat{X}_2^{[1,2]} = B_{21}X_1 = R_{21}R_{11}^{-1}X_1, \]  
(5.5)

where \( B_{ij} \) are set of linear ordinary least squares (OLS) regression coefficients when explaining \( X_i \) with \( X_j \). Hence RDA defines the shared variation of, view 1 to be that part of the signal that can be modeled using view 2 and vice versa.
Definition 16 (Shared variation using regression). The variation that view \( b \) shares with view \( a \) is defined as \( X_b^{(a,b)} = f_{a\rightarrow b}(X_a) \), where \( f_{a\rightarrow b} : \mathbb{R}^{N \times d_a} \rightarrow \mathbb{R}^{N \times d_b} \) is some regression model that estimates \( X_b \) using \( X_a \).

We solve the general shared variation extraction problem (Problem 2) using Definition 16 as the definition of shared. This leads to a solution that can be seen as a generalization of RDA for data collections with more than two views. Next, we will explain how this is done.

5.5.3 Regression Chains

This section describes regression chains which are used to find shared variation. A chain that finds the shared variation for view \( X_k \) starts from some view \( X_i \) and always ends in \( X_k \), visiting each of the other \( K - 2 \) views \( X_{[K] \setminus \{i,k\}} \) exactly once. Let \( l \) be a random permutation of the integers \([K] \setminus \{i,k\}\). To extract the shared variation in \( X_k \) we can now define the regression chain from \( X_i \) to \( X_k \) as

\[
X_k^{(all)} = f_{l_{K-2}\rightarrow k} \circ f_{l_{K-3}\rightarrow l_{K-2}} \circ \cdots \circ f_{l_1\rightarrow l_2} \circ f_{l_1\rightarrow l_1}(X_i),
\]

(5.6)

where \( \circ \) denotes function composition. This process is also illustrated in Figure 5.2.

The regression function \( f \) in Equation (5.6) is general, and can be any function that allows a mapping from one multidimensional view to another, such as ordinary linear regression, regression using support vector machines, random forests etc.

We will next introduce some properties of a regression chain that consist of consistent regression functions. Let us first define a consistent regression function:

Definition 17 (Consistent regression function). A consistent regression function \( f_{a\rightarrow b} : \mathbb{R}^{N \times d_a} \rightarrow \mathbb{R}^{N \times d_b} \) outputs a constant value when the distributions of the views \( X_a \) and \( X_b \) are independent, i.e., \( \Pr(X_a \mid X_b) = \Pr(X_a) \).

Chains of consistent regression functions have two important properties:
**Theorem 1.** \textit{Full loss of shared variation.} For a consistent regression function the regression chain of Equation (5.6) outputs a constant value, if the distribution of any of the views is independent of the other views.

\textit{Proof.} Consider a regression chain from view $X_i$ to view $X_k$. Let the $j$:th view $X_j$, $j \in [K] \setminus \{i, k\}$, be independent of the others. According to the definition of a consistent regression function the regression from view $X_j$ to the next outputs a constant value. All subsequent steps receive this constant as input and output a constant, since a constant-valued variable is independent of any random variable. All chains contain $X_j$ and consequently all chains output a constant value. \hfill \Box

\textbf{Theorem 2.} \textit{Full transfer of shared variation.} If all of the regression functions in the regression chain of Equation (5.6) have zero error, then the output of the chain also has zero error.

\textit{Proof.} Again, consider a regression chain from view $X_i$ to view $X_k$. If the regressions have zero error, i.e., $X_{i_2} - f_{i_1 \rightarrow i_2}(X_{i_1}) = 0$ for any pair $i_1, i_2 \in [K]$, each function in Equation 5.6 receives its input with zero error and $X_k^{(all)} = X_k$ for all $k$ in all chains. \hfill \Box

In other words, if at least one of the views is totally unrelated to the rest of the views, the chain breaks and outputs a constant value (Theorem 1). On the other hand, if all of the views contain the same variation and this is captured by the regression function, the shared variation will be equal to the views themselves (Theorem 2). Intuitively, the chain of regressors suppresses the part of the signal that is not shared by all of the views and maintains the shared part.
5.5.4 COCOREG Algorithm

The algorithm that implements the principles discussed above and solves the shared variation extraction problem (Problem 2) is the following:

**Data:** $K$ matrices $X_k \in \mathbb{R}^{N \times M_k}$ with $N$ observations of $M_k$ features

**Result:** Filtered version of each $X_k$

Subtract mean from each time series;

Train the regressors for the mappings $f_{ij}$;

for $k \in 1 : K$ do

<table>
<thead>
<tr>
<th>Form $C$ regression chains;</th>
</tr>
</thead>
<tbody>
<tr>
<td>for $c \in 1 : C$ do</td>
</tr>
<tr>
<td>Compute output of chain $c$;</td>
</tr>
<tr>
<td>end</td>
</tr>
<tr>
<td>Average over chains;</td>
</tr>
<tr>
<td>end</td>
</tr>
</tbody>
</table>

**Algorithm 1:** The COCOREG algorithm. There are in total $(K^2 - K)$ mappings $f_{ij}$. The number of chains $C$ is $(K - 1)!$ for full and $K - 1$ for sampling COCOREG (see the end of this section for details).

The output of COCOREG is in the original data space. It can be thought of as a filtered version of the original data that shows only the variation that is present in all of the views. The extraction happens without explicitly optimizing complex criteria formulated over multiple views. In terms of modeling principle, COCOREG is somewhat of an outlier in the multi-view learning literature. It does not explicitly optimize a dependency measure, nor does it specify a generative process. Instead, COCOREG uses chains of ordinary regression models to dampen variation that is not shared by the views, leaving only the shared part as the output. To extract the variation that a view $X_k$ shares with others, it is modeled as the output of a chain of regressors going through the other views. Since the chain always contains every view, only variation shared by all views will be carried forward.

To form all possible regression chains for a collection of $K$ datasets a total of $K^2 - K$ mappings $f_{ij}$ need to be estimated. In addition, there are $(K - 1)!$ possible regression chains from $K - 1$ different $X_i$ that end at $X_k$.

The **full** COCOREG algorithm uses all of them and is referred to as CCR. To reduce the computational burden, we randomly sample one chain for each starting view $X_i$. This variant is called the **sampling** COCOREG and abbreviated CCRS.

In sum, the computational complexity of COCOREG depends on the regressor and whether all possible regression chains are used. For a linear
regressor it takes $O(M^2N)$ to train the model and $O(MN)$ to apply it. A conservative estimate is to use $M = \max_k (M_k) = M_{\text{max}}$. This gives for a COCOREG with a linear regressor a total time complexity of

$$O(\max^2 MN^2) = O(M_{\text{max}} N K!$$

for CCR and

$$O(M_{\text{max}}^2 N K^2)$$

for CCRS. As $K$ is typically fairly small even the full version is often computationally feasible despite the $K!$ term.

An R implementation of the algorithm is available in CRAN as the cocoreg package (Korpela and Henelius, 2016).

5.5.5 Evaluation

This section introduced COCOREG, a novel algorithm for the analysis of shared variation between views. COCOREG addresses the shortcoming of existing methods by providing a simple yet flexible framework for extracting the shared variation. The use of standard regression models means flexibility: depending on application a suitable class of regression models can be put to use. Extension to non-linear relationships is just as easy – just select a non-linear regressor. The basic linear regression version is also quick to compute, which makes the method suitable for exploratory analysis and online applications. COCOREG outputs its results in the original data space which is familiar and convenient for the end user. If a single time series describing the shared variation is needed, principal component analysis (PCA) can be used as a post-processing step.

COCOREG is best viewed as a filter that extracts variation that is common to all of the views, but gives no information on dependency structures concerning smaller sub-collections of data. This might appear to be a niche solution at first sight, but the approach has clear advantages. First, the idea is simple and consequently implementing the algorithm is straightforward.

Second, the algorithm works with any regression function hence giving the researcher a choice. For example, a robust linear regressor can be selected for noisy data or if particular non-linear dependencies are assumed, a non-linear regressor can be used. The ability to change the regressor also makes the method adaptable to a wide variety of research problems.
Third, there are no tunable parameters in the method itself. In contrast, to apply GFA one has to decide on a model rank, define thresholds for component activity, fear that the choice of prior has effect on the result and use multiple starting points to avoid local minima.

Last, the algorithm is computationally efficient. The linear CCRS version of COCOREG scales quadratically with maximum dimension $M_{\text{max}}$ and number of views $K$ and linearly with the number of observations $N$. As the most large views have much more observations than variables or views, CCRS remains feasible in many applications.

### 5.6 Applications

#### 5.6.1 Shared Variation in Single-Trial ERPs

This example uses data from the ERP experiment already described in Section 4.2. We use the data to show how COCOREG works and that confidence bands are useful in this context as well.

The data collection consists of average ERP waveforms from eight subjects. To keep the output simple, a representative subset of four channels is used. The input hence consist of eight matrices, each having four time series with 450 time points. The input data are shown in Figure 5.3 (left) along with its path-sampled COCOREG projection (middle). The projection has very low amplitude, possibly because the ERPs of subject one (sbj_1) seem to differ from the others. We test this by computing the CCRS projection for an input without this subject (right). The projection amplitude is now roughly an order of magnitude larger. The dataset of sbj_1 clearly does not share much variation with the rest and if present, it blocks the flow of information in the regression chain.
Figure 5.3. CCRS for average ERP data. Shown are original data from eight subjects (left), CCRS on this data (middle) and CCRS for subjects 2 to 8 (right). Note how the exclusion of sbj_1 causes an increase in the CCRS projection amplitude. Note also flat projections for channels that differ from the others.
The COCOREG output for subject sbj_2, channel Cz, is shown in Figure 5.4. Subject sbj_1 is not included in the analysis and the CCR variant of COCOREG is used, i.e., all \((7 - 1)! = 720\) paths are used to compute the average projection (thick black). A subset of the single-path projections are visualized in faint gray and confidence bands with in-sample coverage of 90\% are also shown. The Figure 5.4 shows that there is quite a lot of variation in the single-path projections and that confidence bands are a handy tool to pinpoint the densest areas of the single-path curves. The figure also illustrates how two confidence bands (MWE and MI) can have the same in-sample coverage and yet be different in shape and width.

![Figure 5.4](image_url)

**Figure 5.4.** Confidence bands with in-sample 90\% coverage for full cocoreg output at channel Cz for subject 2. Five percent of the paths are shown in light gray and their average in thick black.
6. Discussion

This thesis has two main lines of research both of which are related to explorative data mining. First we introduced new methods to draw multivariate confidence intervals for time series data or parallel coordinates plots. Completely new methods were introduced and several practical improvements were made related to existing methods. Thereafter we described a novel method to study shared variation between datasets. The idea is to use readily available regression tools to filter out variation that is not shared by the datasets. Since any regression function will do, the method can be easily modified according to problem specific needs. Discussion on the assumptions made, limitations of the methods and respective research needs are presented next, followed by a discussion on two promising application areas.

6.1 Limitations and Future Work: Confidence Bands

As with any analysis method there are some limitations that should be kept in mind when computing confidence bands. First of all, the MWE algorithms presented in this thesis are greedy and do not contain any mechanism to “look ahead”. In some rare cases this can make MWE to converge to a solution that is very far from the optimum. For example, two very similar time series can prevent each other from being removed, although removing both of them would yield a considerable shrinkage in the envelope width. These situations can be spotted by plotting a random sample of the time series along with the confidence bands to see if there are any areas of exceptionally low density inside the bands. Sudden large deviations between MWE and naïve bands might also indicate problems. Note that while distance measure based confidence handle this situation correctly they otherwise behave rather different than MWE as shown in
Publication I. A closely related problematic data type are non-unimodal distributions. However, these can be easily spotted by simply plotting the raw data.

Second, all confidence bands in this thesis are non-parametric or empirical, meaning that only observed data points can serve as confidence band boundary points. Further on, as multi-dimensional datasets are practically always relatively small compared to the size and complexity of the vector space they span, sparse areas cannot be avoided. In other words, the possible trajectories of the bands are dictated by the observed data points and, if higher resolution is needed, more data needs to be collected. Note however that if the cross validation procedure is used to guarantee coverage, the error made due to sparse data point grid is always towards the bands being too wide. Hence collecting more data shrinks the bands towards the intended size.

Parametric approach to confidence band computation might bring some benefits. It would be able to deliver more accurate bands since there would be no need to round up to nearest observed value. Depending on the model, the parametric approach might also be computationally more effective. However, it might be difficult to take the correlation structure of the data into account in a simple way. Ignoring it altogether leads to unnecessarily wide bands for correlated datasets. For example, a parametric model could make Bonferroni correction feasible for the quantile based bands.

Note also that in the basic MWE algorithm all dimensions are considered to have equal weight when computing the width of the band. Consequently, if variance changes from dimension to dimension, dimensions with high variance influence the result more than dimensions with low variance. If the data are time series, the dimensions usually have comparable variance as the measured variable and its statistical properties remain roughly the same throughout time. And if the variance happens to change as a function of time, this would be an interesting observation rather than a nuisance property of the data to control. On the other hand, if the data are not time series there might be considerable differences between the variances of the dimensions. In these cases it is up to the user to decide, if the variances need to be normalized prior to confidence band computation and whether denormalization is needed afterwards to make the bands reflect actual variances.

Moreover, it is worth reminding that confidence bands make most sense in situations where correlated variables are visualized in two dimensions.
The reasons are two-fold. First, the confidence band is an inherently two-dimensional visualization. Its multivariate counterpart is a hyper-rectangle, which is a very crude approximation for a general multidimensional distribution. Such an approximation is justified if it is native to the visualization in use, as is the case with time series. Second, the MWE algorithm takes the correlation structures into account in a natural way. The advantage is the larger the stronger the correlations and vice versa. Datasets with practically uncorrelated variables are best analyzed using traditional MHST methods.

Regarding future work, one line of research could be to more thoroughly study what kind of methods are best for establishing control over the true coverage of the band. Even defining “best” is an issue in itself – do we for example want the method to be fast and scalable or is perhaps accuracy the main concern. It is clear that corrections are needed to establish control over expected coverage but the required resampling procedures are in practice painfully slow. A trade-off between speed and accuracy could be established.

A more practical question is the one of how much data are needed to reach a desired level of accuracy. Especially in applications where new curves may be sampled without limit, the practitioner would benefit from knowing in advance roughly how much to sample. Clearly this number depends on the autocorrelation structure and the dimensionality of the data as well as variance. Nevertheless it might be possible to give some guidelines.

### 6.2 Limitations and Future Work: COCOREG

The assumption of constant relations between views is the biggest implicit assumption made in the COCOREG algorithm. It is a direct consequence of traditional regression analysis where the relation between independent and dependent variables remains fixed through time. However, in many real life applications the views are connected (share variation) only intermittently, usually driven by some possibly unobserved external factor. In these situations problems arise as the same set of regression coefficients is used to describe the relation at all times. One simple solution would be to perform the analysis in shorter windows for which the relations between views remain more constant. The implementation could for example involve a sliding window with some overlap between adjacent windows.
Properties and implementation details of such a method would be one direction of future work. Another option would be to use a regression model in which the mapping between views is itself a function of time such as with hidden Markov model regression (Trabelsi et al., 2013) or piecewise regression (Brailovsky and Kempner, 1992).

Another practical issue to solve is the fact that shared variation is a continuous concept: any percentage of variation might be shared. As this percentage falls into comparable levels with noise, it becomes difficult to distinguish what happened because the views share variation and what was just a coincidence. The GFA method has the same problem, there a threshold needs to be set in order to select components that contribute strongly enough to all views. A similar solution for COCOREG would be to use the shared variation projection to explain the original data and assess significance based on how much of the variance gets explained. This gives a overall measure but does not help in assessing which time ranges show shared variation and which don't. Post processing methods remain a future challenge for the analysis of shared variation.

6.3 Applications

In the following we present some current and topical application areas for the methods of this thesis. Although many of the examples concern time series data, the methods themselves are applicable in a much wider context. Neither confidence bands nor COCOREG assume any ordering for the data and therefore they can be applied to any dataset that contains unknown correlation structures. Examples of such non time series data include, e.g., univariate probability density estimates, signal power spectra, ROC-curves, impulse response functions as well as functional datasets like the ones produced in gait analysis.

6.3.1 Communicating Uncertainty in Time Series Forecasts

During the last decade a major trend in both public and private sector has been to strive for more exact knowledge, objective decisions and better control through the use of measurements and data. A majority of these undertakings yield numerical values at some stage of the process and usually the measurements are repeated at periodic intervals. The result is nothing else but a time series and these have become a very common type
One interesting time series problem is how to communicate uncertainties related to forecasts (Chatfield, 1993). This is also an application where confidence bands could be used far more extensively and systematically. A common approach even today is to give the single most probable trajectory as the forecast and forget uncertainty altogether. Such visualizations can be quite misleading as in many cases the single trajectory gives a false feeling of preciseness and may even have been manipulated to have a certain desired shape. To add realism, a set of worst/best case scenarios may also be also given, but these do not convey much information about where the bulk of the probability mass lies. A much more informative type of visualization is a fan chart (Figure 3.2), named by the Bank of England, which first started using them as part of their inflation report in 1993. A more detailed fan chart with color shaded percentiles was taken into use in 1996. The fan chart essentially visualizes the forecaster’s expectations regarding (i) the central tendency of the forecast trajectory, (ii) its degree of uncertainty (variance) and (iii) a view of the balance of the risks (a skew in the distribution) (Britton et al., 1998).

In this particular economics application the properties (i) to (iii) are constructed as a shared view of expert opinions, but in other applications the shape of the distribution might come from, e.g., multiple runs of a model with varying input parameters. The important aspect of the chart is that it seeks to visualize the distribution of possible future trajectories of some vector-valued variable. In this regard the methods introduced in this thesis are a perfect fit for forecasting applications where the empirical distribution of the forecast is known and the task is to visualize it. The only requirement is that a set of trajectories is available but this can usually be satisfied since, if a model exists, it is usually also possible to sample from it. Confidence bands also provide coverage guarantees, unlike the original fan charts that estimate the quantiles in a few points in time and interpolate the steps in between.

### 6.3.2 Analysis of Wearable Sensor Data

Small wearable electronics have made it feasible to track one’s daily activities closer than ever before. This opens up new possibilities to promote worker health and consequently these devices have been an active topic of research at my employer. Currently several devices are able to measure physiologic features such as activity (accelerometer), heart rate variability,
Discussion

Electrodermal activity (EDA) and temperature. Another daily companion, the mobile phone, provides an additional source of data providing the interested quantified self (QS) enthusiast with GPS position, velocity, visited places, calls, messages and many other signals. The technology has matured such that all this can be acquired without much extra effort. However, services around this pool of data are only emerging. The following envisions some applications that could benefit from the methods of this thesis.

Because the devices gather data at regular intervals practically all of them produce time series. Combined with the repetitive nature of our lives, sets of time series are created with natural alignment between curves. Examples include, e.g., activity measurement overnight or throughout a typical working day.

Confidence bands provide an important visualization tool to examine the variability of these curve sets. Once average profiles and simultaneous bands describing the quantiles of the data are found, comparison against the past or against other people can be made. These analyses form the input for personalized feedback – adaptive recommendations that steer the individual towards a healthier life.

Another possibility is to use the confidence bands to detect outliers and notify the user. The personal well-being app could for example note that you have slept poorly for several nights in a row and suggest measures to break the cycle.

Furthermore, almost certainly there will be an increasing number of applications where the curves are classified into groups. For these applications confidence bands provide a natural way of assessing overlap and distance between the groups.

From the viewpoint of multi-view learning the QS data provide interesting possibilities. One application could be the identification of co-occurring changes in signal level. An example analysis could, e.g., report how many arousals happened during the night due to external disturbances such as noise or light. In addition, the user could be presented with estimates about how much potential there is to improve sleep by just eliminating the external factors. This would, however, require a windowed version of the current COCOREG-algorithm such that the degree of shared variation is measured in smaller chunks.

Another application could mine for sub-groups of views that share variation most. A sequential algorithm to do this would be simple to implement,
for example by starting from a set of two groups and adding new ones until the projection amplitude suddenly drops as an independent view gets added. In reality this could be used to identify users with similar feature profiles.

Event sequences are another common form of QS data. They can be viewed as a special sub type of time series that can either be extracted from another series or created directly as events unfold in time. Publication IV provides randomization and surrogate data generation tools to explore event sequences, helping the analyst in decision making. A typical problem involves assessing the significance of a pattern: is it possible to observe the pattern purely by chance or is it so exceptional that a more careful look is warranted.

### 6.3.3 Future Work and Impact

As shown above, the methods presented in this thesis have many applications in the data-driven society. To promote a rapid adoption of the methods some marketing needs to be done. A comprehensive review article, including practical usage guidelines, could be the first step. This should be accompanied by an easy to use and access software package or packages. Lastly these tools need enough web presence to ensure a top-rank position in major search engine queries. An interconnected web of publications, software packages, development repositories and tutorial websites should be enough to draw the practitioners’ attention.

To sum up, both the size of datasets as well as the number of data producing devices have increased rapidly in recent years (Economist, 2015; van der Meulen, 2017; Szalay and Gray, 2006). This trend will continue into the foreseeable future. Also the volume of time series datasets grows together with the number of Internet of Things (IoT) devices producing and logging it at regular intervals. These trends underline the need for new and improved methods and software tools for visualization and pattern finding. Methods presented in this thesis fill this need by providing useful tools for the data scientist. Confidence bands allow the variability of vector-valued data to be visualized in a way that additionally allows for quantitative conclusions to be made. They also serve as a useful tool in rapidly communicating for example forecast uncertainty to the public. The COCOREG algorithm can be used to find patterns of shared variation, e.g., between a group of IoT devices which helps in understanding their relationships. Lastly, the study of interval sequences is widely applicable.
Discussion

whenever the occurrence rates of some incident are of interest, e.g., the analysis of eye blink rates to measure vigilance. These tools help the analyst to understand the data better, ultimately enabling him to make higher quality and more data driven decisions.


References


Errata

Publication III

On page 1117 the Equation (6) should be:

\[ R_{22} = E [x_2^a x_2^{aT}] = A_{21} A_{21}^T + A_2 A_2^T \]

On page 1119 in Figure 2 the rightmost mapping should be \( f_{I_{K-2} \to k} \) instead of \( f_{I_{K-2} \to l_k} \).