Combinations of Time Series Forecasts: When and Why Are They Beneficial?

Christiane Lemke

A thesis submitted in partial fulfilment of the requirements
of Bournemouth University for the degree of Doctor of Philosophy

January 2010

Bournemouth University in collaboration with Lufthansa Systems Berlin GmbH

Copyright statement

Abstract

Time series forecasting has a long track record in many application areas. In forecasting research, it has been illustrated that finding an individual algorithm that works best for all possible scenarios is hopeless. Therefore, instead of striving to design a single superior algorithm, current research efforts have shifted towards gaining a deeper understanding of the reasons a forecasting method may perform well in some conditions whilst it may fail in others. This thesis provides a number of contributions to this matter. Traditional empirical evaluations are discussed from a novel point of view, questioning the benefit of using sophisticated forecasting methods without domain knowledge. An own empirical study focusing on relevant off-the-shelf forecasting and forecast combination methods underlines the competitiveness of relatively simple methods in practical applications. Furthermore, meta-features of time series are extracted to automatically find and exploit a link between application specific data characteristics and forecasting performance using meta-learning. Finally, the approach of extending the set of input forecasts by diversifying functional approaches, parameter sets and data aggregation level used for learning is discussed, relating characteristics of the resulting forecasts to different error decompositions for both individual methods and combinations. Advanced combination structures are investigated in order to take advantage of the knowledge on the forecast generation processes.

Forecasting is a crucial factor in airline revenue management; forecasting of the anticipated booking, cancellation and no-show numbers has a direct impact on general planning of routes and schedules, capacity control for fareclasses and overbooking limits. In a collaboration with Lufthansa Systems in Berlin, experiments in the thesis are conducted on an airline data set with the objective of improving the current net booking forecast by modifying one of its components, the cancellation forecast. To also compare results achieved of the methods investigated here with the current state-of-the-art in forecasting research, some experiments also use data sets of two recent forecasting competitions, thus being able to provide a link between academic research and industrial practice.

Contents

List of Tables

List of Figures

Acknowledgements

To Bogdan: Thank you for being a great supervisor. For enabling me to get to know life in academia in all of its facets and providing interesting perspectives for the future. For motivating me and believing in me when I needed it. Your help, time and dedication are greatly appreciated.

To Silvia: Thank you for constant encouragement and your enthusiasm regarding the airline application of this work. For providing me a practical perspective and not letting me get lost in exclusively academic thoughts.

To my parents, sister and grandparents: Thank you for your unconditional love and support, as well as putting up with my ideas, which might have seemed crazy at times. I will probably keep them coming.

To my fellow PhD students: Thank you for making my time at Bournemouth University a very pleasant one, it was great getting to know and working with you.

To my friends in the UK: Vegard, thank you for going the better part of the PhD journey with me. For helping me overcome the little chicken inside of me at various occasions and for bearing with me when I banned PhD and work talk from conversations. Janko, thank you for your support during the crazy final PhD stages. I keep being amazed by how much we think alike and how you manage to make me feel happy. Kasia, I am so glad we met. Thank you for all the basketball, the conversations we had and for believing in me all the time. Isla, thank you for being genuine and bubbly and for getting me away from my computer to do fun stuff like climbing and kitesurfing.

To my friends "from home": Jana, thank you for being the strongest link to my life in Germany. Our telephone chats in good and bad times are one of my weekly highlights. Julia, thank you for your contagious restlessness and for the fun times we have whenever we meet. Ulli, thank you for being an inspiration with your unusually honest and emotional personality. Ralf, thank you for being so direct and our refreshingly blunt conversations. Lars, thank you for showing me the world of poetry slams and the beautifully melancholic hours we spend playing the guitar.

I would also like to thank Bournemouth University and Lufthansa Systems Berlin for the support and for providing a productive and enjoyable research and work environment.

Author's declaration

The work contained in this thesis is the result of my own investigations and has not been accepted nor concurrently submitted in candidature for any other award. Conference and journal publications related to this work are referenced. This thesis has been conducted in a collaboration with Lufthansa Systems Berlin GmbH and continues work of a previous project that resulted in the thesis of Riedel (2007).

List of Abbreviations and Symbols

ARIMA	Autoregressive integrated moving average
ARR	Adjust ratio of ratios
DCP	Data collection point
DOW	Day of the week
DT	Decision tree
F	Fareclass
GMDH	Group method of data handling
LSB	Lufthansa Systems Berlin GmbH
MSE	Mean squared error
NN	Neural networks
NN3/NN5	Neural network forecasting competitions 2006/2008
O&D	Origin and destination pair
ODO	Origin-destination opportunity
POS	Point of sale
RMSE	Square root of the mean squared error
S	Seasonal component of a time series
SETAR	Self-exciting threshold autoregressive models
SMAPE	Symmetric mean absolute percentage error
STAR	Smooth transition autoregressive models
SVM	Support vector machine
T	Trend component of a time series
α, β, γ	Smoothing parameters for smoothing based approaches
$\hat{\boldsymbol{y}}$	Vector of time series forecasts
$\boldsymbol{\omega}$	Combination weight vector
$\boldsymbol{e}$	Unity vector
ϵ	Forecast error
$\hat{y}$	Time series forecast
$\hat{y}^c$	Combined time series forecast
ω	Weights for linear forecast combination
ϕ	Dampening factor
$\sum$	Covariance matrix
b	Bookings
cr	Cancellation rate
ref	Reference curve in the airline application
f	Feature
l	Level of a time series
r	Growth rate of a time series
t	Trend of a time series
y	Time series observation

1 Introduction

Going beyond the well-known daily weather forecast, forecasts can be found in a wide variety of scenarios. Forecasting stock prices and exchange rates is common practice in finance as, for example, investigated in Gyorfi et al. (2006) and Kodogiannis & Lolis (2002). Forecasting variables like gross national product or unemployment is crucial for macroeconomics, for a recent example see Marcellino et al. (2006). As demonstrated in Weatherford & Kimes (2003) and Koutroumanidis et al. (2009), companies of all sizes use forecasts to predict demand for their products to support planning and decision-making. The lead time for decisions can vary significantly depending on the application; electrical load applications as described in Hippert et al. (2005) may require forecasts every few seconds while a few days are usually sufficient for transportation and production schedules as, for example, investigated in Cox Jr & Popken (2002). In the case of long-term investments based on macroeconomic data as used by Stock & Watson (2002), lead time can even amount to several years.

In general, forecasting describes a broad research area concerned with estimation of future events or conditions. According to Makridakis et al. (1998), forecasting approaches can be roughly divided into qualitative and quantitative models. Qualitative models assume sufficient knowledge of an underlying process and are often experts' judgements. Experts usually base their opinion on different sources of information, their intuition and subjective beliefs that cannot be easily quantified. The focus of this thesis however lies on quantitative forecasting, which mainly involves automatic prediction of numerical data. The data examined here consists of univariate sequences of data points, so called time series, that are investigated for regularities and patterns in their past to extract knowledge that can help to predict the future. In addition to looking at data sets publicly available from forecasting competitions, data has been provided by Lufthansa Systems Berlin GmbH (LSB), allowing an investigation of forecasting approaches in the industrial setting of the airline industry.

This chapter will give background information and motivations for this work. It will start to set the scene for the airline-specific part of the thesis by describing the importance of forecasting in the context of airline revenue management. It will continue to introduce the area of time series forecasting in general, including a brief look at forecast combinations and meta-learning, which are major topics in this thesis. The definition of aims and objectives as well as a description of the organisation of the thesis follow in the next sections. An overview of the original contributions and a list of publications conclude the chapter.

1.1 Background and motivation

A considerable part of the work presented in this thesis has been carried out in collaboration with Lufthansa Systems Berlin GmbH, a company providing revenue management software for airline carriers. The products the airline industry offers are seats on a plane which, contrary to the perception on first sight, do not only differ in being in the physically separated first or second class. Pak & Piersma (2002) rather describe it as a complex system of fareclasses that differ in various conditions, like refund availabilities, cancellation options or stopover arrangements. Customers can roughly be separated in two groups according to Zeni (2001): business and leisure passengers. While business passengers usually seek to make travel arrangements shortly before departure with little flexibility, leisure passengers tend to book their tickets well in advance while being more flexible with dates and booking conditions. In addition, business passengers are usually willing to pay a higher price for their tickets, thus contributing to more revenue than a leisure customer. The key to efficient capacity control is the determination of the point in time when it is beneficial to restrict bookings in a lower-fare class to leave space for later booking high-fare customers. This is of both economical and ecological interest, producing a higher revenue for a high demand flight and fewer unoccupied seats in a low demand one.

Accurate forecasting of anticipated booking, cancellation and no-show numbers is vital for revenue management. If the demand forecast of high-fare passengers is too high, seats go empty that could have been sold to low-fare passengers. On the other hand, if it is too low, passengers willing to pay a higher fare have to be turned away. Revenue management and forecasting do however not only support the decisions that have to be made on a daily basis, but also provide key information for strategic long-term decisions such as which itineraries to offer or how to change the size of the maintained fleet, see Zaki (2000).

Historic booking and cancellation numbers constitute time series, which might include valuable information for forecasting the future. Time series forecasting has been a very active area of research since the 1950's, and a variety of forecasting approaches have been introduced in the scientific literature and were used in many practical applications. Available forecasting algorithms can be roughly divided into a few groups: simple approaches are often surprisingly robust and popular, for example those based on exponential smoothing. Statisticians and econometricians tend to rely on complex ARIMA models and their derivatives, while the machine learning community mainly looks at neural networks. A review can be found in Gooijer & Hyndman (2006).

A few years after the first publications in the area of time series forecasting, research on combinations of forecasts became popular as well, with the seminal paper having been published by Bates & Granger (1969). It divided the community into researchers who believe that combinations are a great approach to decrease the risk of selecting the wrong individual model and better approximating a real world time series, and others who think that if a combination outperforms individual methods, it is only an indication of an individual method requiring better specification. However, especially in machine learning, combinations of methods have proven successful. As summarised in the review of Timmermann (2006), the choice of combination methods is extensive. Very simple methods average available individual forecasts with or without a certain degree of trimming, others take past

performance into account in different ways for calculating linear weights. Nonlinear forecast combination methods do exist, but literature seems comparatively sparse.

During the years, more focus has been put on the question of when a particular forecasting method works well. Promising work has been published on linking characteristics of time series to the performance of a forecasting algorithm, first mostly in the context of rule based systems as, for example, in Adya et al. (2001). More recently, the term "meta-learning" was adopted from the machine learning community, accommodating a wider range of learning methods as described in Prudencio & Ludermir (2004*a*) and Wang et al. (2009).

In a previous collaboration project resulting in the thesis of Riedel (2007), diversification procedures to extend the number of available individual forecasts similar to the ones introduced in Granger & Jeon (2004) have been investigated and applied to demand forecasting for airline data. The success of the work led to the belief that higher overall forecast accuracy can also be achieved by modifying the cancellation forecast, which is another important component in airline revenue management forecasting and which will be investigated in this thesis. Aims and objectives of this thesis in general and for the airline application in particular are described in the next section.

1.2 Aims and objectives

The main aim of the thesis is contributing to a better understanding of forecast model selection and combination approaches. On a general level, the following questions will be investigated:

- To what extent are expert contributions beneficial in empirical forecasting applications? Can adequate performance be achieved by combining simple individual predictors?
- How can a pool of individual methods be extended, and what characteristics are necessary to increase combination accuracy?
- Can situations in which a particular method works well be automatically identified and domain knowledge exploited for improved forecasting performance?

A major practical goal of this thesis is the improvement of the net booking forecast in the airline revenue management application of Lufthansa Systems by looking at modifications for one of its components, the cancellation forecast. To achieve this, several objectives are pursued:

- The design and implementation of a new forecast based on cancellation probabilities, enhancing the diversity of the individual forecasts available for combination.
- Investigation of the benefit of forecast combination for airline data and the extent of possible improvements while meeting application-specific requirements like time restrictions and coping with noisy data.
- Automatically generating and exploiting domain knowledge for method selection and more effective combination of individual predictors.

- Evaluation of diversification procedures for generating additional individual forecasts, by considering different functional approaches, different parametrisations and different aggregation levels of the data used for learning.

To achieve these contributions, relevant literature will be reviewed and discussed before conducting and analysing empirical investigations. Two different kinds of data sets will be used for increased value of the results: publicly available data sets obtained from forecasting competitions will allow comparison of the results given in this thesis to results obtained by experts in the field of time series forecasting and facilitate replication. The application of the investigated approaches to real-world airline data will give insights to the applicability of latest research results to an industry, in which accurate time series forecasting has a big impact on generated revenue.

1.3 Methodology and organisation of the thesis

Background knowledge in three different areas are relevant for this thesis: airline revenue management, time series forecasting and forecast combination. The introductory information will be extended in Chapter 2, providing a literature review and discussion of most important contributions and algorithms for each of the areas.

Chapter 3 investigates the question of how well off-the-shelf time series forecasting methods perform in empirical studies with the goal to assess the benefit of applying complex forecasting algorithms that usually have to be identified and fitted by experts. In the same context, the benefit of combinations of these simple forecasts, promising to provide a convenient way out of the dilemma of having to find and parametrise a suitable method for each forecasting problem is evaluated. After reviewing evidence from other empirical studies, an experiment designed for this specific point of view is presented and analysed, using publicly available datasets from forecasting competitions to allow comparison with contributions of different experts in the field. Chapter 4 then looks at forecasting methods currently used in the airline application and compares results obtained to those from the previous chapter.

Having provided the background and first empirical results of time series forecasting and combination approaches, the focus of this thesis shifts from investigating *which* methods work best to *why* some methods work well, and in which situations. Chapter 5 considers the forecasting problem from a higher level point of view. It investigates the automatic generation of domain knowledge to guide method selection and combination in the forecasting process, which can be summarised with the term of "meta-learning". Following an extensive review of work done in this particular area, different machine learning approaches are tested in an empirical study to evaluate possible performance improvements on all the data sets available.

Chapter 6 provides a thorough investigation of opportunities to increase forecast accuracy in the airline application. It looks at characteristics of individual forecasts that are necessary to contribute to an improved combination result, investigating the concept of diversity in the context of pools of time series forecasts. The benefits of functional, parameter and data aggregation level diversification is empirically evaluated and analysed using the airline data set.

Chapter 7 concludes by summarising results and findings and evaluating how the analysis of diversity and meta-learning has contributed to the understanding of forecast combination in general. An outlook on future work ends the thesis.

1.4 Original contributions

A comprehensive treatment of time series forecasting in airline revenue management is given in Chapters 4 and 6 and parts of Chapters 2 and 5, a treatment which has not been yet available in comparable detail. Data used for related empirical studies was kindly provided by Lufthansa Systems, and with airline revenue management applications being an extremely successful practical application area for time series forecasting algorithms, the thesis provides unique insights to the practical relevance of forecasting methods in this area. It complements the thesis of Riedel (2007), which resulted from the same collaboration in a previous project. A new algorithm for the prediction of airline cancellation rates is presented in Chapter 2.

A new perspective on empirical studies and the necessity of expert contributions in practical forecasting applications is given in Chapter 3, parts of which were published in Lemke & Gabrys (2007), Lemke & Gabrys (2008*a*) and Ruta et al. (2009).

The meta-learning discussion and empirical study in Chapter 5 is one of the most extensive works in this area to date, extending the features and method pool in comparison to previous work and reviewing a wider range of algorithms. Noteworthy is the use of diversity measures as inputs to the meta-learning algorithms, an original extension to meta-learning in a time series context. Some results have been published in Lemke & Gabrys (2009). The concept of meta-learning is furthermore applied to airline data for the first time.

Diversity is a concept that has its origin in the machine learning community and has not yet to the same extent been applied to time series forecast combination. Chapter 6 looks at the benefit of generating additional individual forecasts by diversifying procedures with the goal to improve the overall accuracy of a forecast combination. Parameter and functional diversifications are most common in the literature, this thesis additionally considers individual forecasts generated by building models on different data aggregation levels. Results on a smaller data set have been published in Lemke et al. (2009).

Overall, this thesis does not aim at adding just another method to the already large pool of available forecast and forecast combination approaches, thus increasing confusion of which method to chose in which situation. It is rather aimed at providing a deeper understanding of the dynamics of a combination of individual forecasts and of the value of domain knowledge and its automatic generation, which will contribute to knowledge on forecasting in general and facilitate improvements of forecast accuracy.

1.5 List of publications

A list of the publications resulting from this thesis in chronological order is provided below:

- Lemke, C. & Gabrys, B. (2007), Review of nature-inspired forecast combination techniques, in 'NiSIS 2007 Symposium'
- Lemke, C. & Gabrys, B. (2008a), Do we need experts for time series forecasting?, in 'Proceedings of the 16th European Symposium on Artificial Neural Networks', pp. 253-258

- Lemke, C. & Gabrys, B. (2008b), On the benefit of using time series features for choosing a forecasting method, in 'Proceedings of the European Symposium on Time Series Prediction', pp. 1-10
- Lemke, C. & Gabrys, B. (2009), 'Meta-learning for time series forecasting and forecast combination', accepted to a special issue of Neurocomputing
- Lemke, C., Riedel, S. & Gabrys, B. (2009), Dynamic combination of forecasts generated by diversification procedures applied to forecasting of airline cancellations, in 'Proceedings of the IEEE Symposium Series on Computational Intelligence', pp. 85-91
- Ruta, D., Gabrys, B. & Lemke, C. (2009), 'A generic multilevel architecture for time series prediction', accepted to IEEE Transactions on Knowledge and Data Engineering

2

Airline revenue management and forecasting

The introduction presented forecasting as an important topic in both research and industrial applications. However, the state-of-the art in industry and research often differ quite significantly for several reasons: sometimes research findings are purely academical and cannot easily be applied in real-world applications; or the requirement for robust and reliable systems in industry causes a certain reluctance to implement latest research outcomes. Furthermore, there is never a guarantee that approaches working well or badly on scientific data sets will perform similarly on data sets in industrial applications, especially since industrial data sets are usually hard to come by in the public domain for reasons of commercial sensitivity. This chapter provides an overview of the state-of-the art in forecasting algorithms from both a scientific perspective as well as from the point of view of the industry partner of this work.

Following an introduction to revenue management, the first section of this chapter highlights the importance of forecasting in the airline industry in particular and introduces forecasting algorithms used at Lufthansa Systems Berlin GmbH (LSB). The next sections take a step away from the specific application and provide a look at time series forecasting and forecast combination in general. A presentation of the most important algorithms is followed by an outlook on future work.

2.1 Airline revenue management

Revenue management has become a mainstream business practice with a growing importance in academic and industrial research and an increasing number of users in many key industries according to Talluri & van Ryzin (2005). Its goal is to maximise profits generated from limited perishable resources by optimising demand-related decisions, thus selling the right product to the right customer at the right time for the right price. Perishable resources can be as diverse as food, hotel rooms or train tickets.

This section extends the introductory information given on revenue management. Its significance particularly for the airline industry is emphasised in a brief historical overview showing that airlines were the first and remain one of most successful users of these systems till today. A look at the role of forecasting in revenue management then provides the connection to the presentation of forecasting methods currently used.

2.1.1 Background

The basic concept of revenue management is very old. However, Talluri & van Ryzin (2005) mention two factors that have considerably boosted its potential in

the last 50 years: scientific advances that facilitate more accurate models of real-world conditions and advances in information technology, that allow use of very complex algorithms on a very detailed level if necessary. This made way for modern automated revenue management on a very high scale and complexity. Revenue management is concerned with four major components: pricing, capacity control, overbooking and forecasting:

Pricing investigates the time-varying calculation of prices for a product. Usually, there is not only one product for one group of customers, but a portfolio of products targeting different customer groups. According to McGill & van Ryzin (1999), pricing is generally the most important and natural factor affecting customer demand behaviour and can be used to manipulate demand in the short run. For example, sufficiently raising the price of one product class will result in sales of this product approaching zero. A review of research done in the area of dynamic pricing policies in the context of revenue management can be found in Bitran & Caldentey (2003).

Capacity control manages the allocation of capacities within the bundle of products. It commonly distinguishes single-resource problems, where the goal is to optimally allocate capacities for a single resource to different classes of demand, and multiple-resource problems, where customers require a combination of resources, for example in a stay in a hotel lasting several days. If one product in the product bundle is not available, the sales of the whole bundle is affected, creating the need of jointly managing resources. Talluri & van Ryzin (2005) state that although multiple-resource problems are much more common in the industrial practice of revenue management, they are still often solved as a number of single-resource problems, treating the resources independently and ignoring network effects that might occur. Usual means for capacity control are limits, specifying how many products from a product class may be sold at most, or protection levels, reserving an amount of capacity for a particular class.

Overbooking is the oldest practice in revenue management and can only be applied in reservation-based systems. Its goal is to compensate for cancellations and no-shows by accepting more reservations than the capacity allows, hoping that the number of customers actually claiming the service or product will be within the capacity. An obvious danger in this respect is the chance of more customers turning up than anticipated, which means that additional revenue generated by overbooking is to be traded off against the risk and compensation of having to deny a service or product. According to Talluri & van Ryzin (2005), overbooking seems to be regarded as a quite mature research area and receives less attention in more recent revenue management research than capacity control or pricing, however, some examples of recent research are summarised in Chiang et al. (2007).

Forecasting quantities such as demand, cancellations, capacity limits and price sensitivity has a critical influence on the performance of a revenue management system. The other three revenue management areas all depend on accurate forecasts. Talluri & van Ryzin (2005) state that forecasting is a "high-profile task" of revenue management, requiring the majority of the development, implementation and maintenance effort. As a guideline, Poelt (1998) estimated that 20% reduction of forecast error in a revenue management system can translate into a 1% increase in

generated revenue. Although it is of course difficult to generalise this number, the importance of forecasting is generally recognised.

Examples of industries in which revenue management is successfully applied are:

- Hospitality industry (for example in hotels, restaurants and at conferences)
- Transportation industry (for example airlines, rental cars, cargo and freight)
- Subscription services (for example internet services)
- Miscellaneous industries (for example retail and manufacturing)

This work investigates forecasting and its application to airline revenue management, a history of which will be summarised in the next section before proceeding to investigate the problem of forecasting in more detail.

2.1.2 History

Airline companies were the first branch of industry applying revenue management according to Talluri & van Ryzin (2005). In the beginning of the 1970's, some airlines started offering restricted discount fares, for example for early bookings. This potentially reduced the number of empty seats on a flight, but introduced a central problem of airline revenue management: how many seats should be protected in the full fareclass so that no passenger willing to pay the full fare has to be turned away? McGill & van Ryzin (1999) state that no simple rule like reserving a fixed percentage could be applied as booking and cancellation behaviour varied considerably across the different flights, days of the week and other factors. Littlewood (1972) introduced a first simple formalism for a single-resource two-class problem, also called Littlewood's rule: bookings in a discount fareclass should be accepted as long as the resulting revenue value exceeds the revenue value of future expected bookings.

The potential influence of revenue management was boosted in the late 1970's, following a trend towards global airline liberalisation, which was for example illustrated by the 1978 Airline Deregulation Act in the United States, removing government control over fares, routes and services from commercial aviation.

Early airline revenue management was based on single-resource seat inventory control, meaning that only the capacity of one scheduled leg of a flight was considered at a time. Talluri & van Ryzin (2005) state that even though approaches to computing optimal booking limits do exist, it is mainly the heuristic approaches that are of great practical importance today, Belobaba (1987) providing an early, but still very popular example. A great number of itineraries however involve connecting flights, which can be booked as one entity at most larger airlines. Capacity control measures on one leg of the flight can thus have unforeseen effects on other flights in the respective itinerary. Modern revenue management systems therefore moved on to multiple-resource systems, which are also called origin-destination systems in the airline industry, considering multiple stops and accounting for network effects. A number of methods have been developed to address the needs of these multiple-resource systems; a review can be found in Chiang et al. (2007).

2.1.3 The role of forecasting

The importance of forecasting for general revenue management has been emphasised in Section 2.1.1. Airline revenue management is no exception to the general rule, on the contrary, forecasting is particularly critical in this area because forecasts directly influence booking limits that determine airline revenue. General revenue management forecasting may include demand forecasting, capacity forecasting and price forecasting, each of which has its specific requirements. The airline-specific part of this work investigates ways to improve the net booking forecast, which is the number of bookings remaining at departure reduced by the cancellations that occurred, by increasing the accuracy of the cancellation forecast.

Booking and cancellation forecasting and its interaction with optimisation as used by Lufthansa Systems are depicted in Figure 2.1. Normally, this revenue management cycle starts with the collection of relevant historic data. A history building process then estimates models and parameters that are necessary for forecasting. Forecasting generates numbers that guide optimisation decisions, like allocations, discounts and overbooking limits. These controls influence the actual booking numbers by closing and opening fareclasses. Once a flight has departed, past observations are again used to build and adjust models for the forecasting in the adaptive history building process.

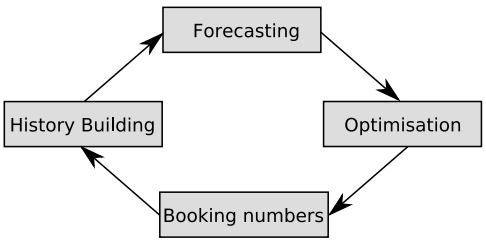

Figure 2.1: Interaction of forecasting with optimisation and past booking numbers

Talluri & van Ryzin (2005) mention that revenue management is mainly a professional practice, which unfortunately makes a lot of available knowledge inaccessible to the general research community. McGill & van Ryzin (1999) support that, adding that airlines are particularly reluctant to share knowledge of their forecasting methodologies due to commercial sensitivity. They also state that most forecasting systems employed by airlines depend on relatively simple moving average and smoothing techniques. Knowledge on critical market changes or anticipated structural breaks have to be realised by manual intervention to the system. Forecasting practice at Lufthansa Systems generally corresponds with these statements and will be described in the remainder of this section.

2.1.4 Lufthansa Systems forecasting basics

Forecasting at LSB involves several steps of preprocessing, postprocessing and actual calculations. Two forecast components are needed for a final booking forecast: the demand forecast and the cancellations forecast. After looking at the data, reference curves and preprocessing algorithms, this section describes current demand forecasting at LSB. The aim of this work however is the improvement of cancellation forecasting, which is why this area will be covered in an extra section.

2.1.4.1 Data and issues

At LSB, data is collected and forecasts are calculated at 23 data collection points (DCPs) atwith fixed distances prior to the departure date of a flight. The number of days to departure assigned to each of the DCPs is shown in table 2.1. At each of these points, booking and cancellation numbers are recorded.

DCP	0	1	2	3	4	5	6	7	8	9	10	11
days to departure	350	182	140	126	98	70	56	49	42	35	28	21

DCP	12	13	14	15	16	17	18	19	20	21	22
days to departure	14	12	10	8	6	5	4	3	2	1	0

Table 2.1: Days to departure for each data collection point (DCP).

Booking and cancellation data are furthermore collected for different dimensions:

- ODO - the origin-destination opportunity. One ODO holds past and present data of flights on the same routing with similar departure times, thus creating a stable history pool for a flight that is unaffected by flight number changes or minor time adjustments to the schedule.
- F - the fareclass (booking class). LSB distinguishes 20 fareclasses differing in price and booking conditions.
- DOW - the day of the week. Data is collected separately for each day of the week.
- POS - the point of sale. This indicates where a ticked was sold, it can be either the 'country of origin', the 'country of destination', or 'other'.

For capacity control, forecasts are generated on the finest possible level (for each ODO, F, DOW and POS combination), but are frequently aggregated to higher levels, for example for visualisation purposes or to support management decisions. The historical numbers for demand and cancellations are treated as univariate time series.

The airline industry environment comes with a few application-specific characteristics that have to be taken into account in the forecasting process: the fine level at which the forecasts are calculated causes a so called 'small number problem', which occurs due to the fact that for some combinations of ODO, F, DOW and POS, there might be only very few bookings, or no bookings at all. This means that small changes in the values lead to wide variances, so that the data is likely to be unstable and it becomes hard to build a model.

The fine level data is furthermore very noisy and susceptible to structural breaks, which are more or less abrupt changes in customer behaviour caused by seasonal effects, events or other unforeseen circumstances. A constantly changing environment requires the forecasts to have adaptation capabilities. In many cases, choices made when building a model (for example parameter values, predictive model or aggregation level used) can lead to deteriorating performance as time passes and the decisions become suboptimal very quickly. In the live application, strong time restrictions are another important factor, as a large number of forecasts needs to be generated in a limited amount of time.

This structure and quality of the existing data leads to the situation that in practice, only a few methods could be identified to produce fairly accurate forecasts for the LSB application investigated here. A number of LSB internal and commissioned studies on this topic have shown that simple and robust time series forecasting models such as simple average, different versions of exponential smoothing or regression models, which will be explained in more detail later, perform significantly better than a number of well known more sophisticated methods. This is not an observation specific to LSB, as a survey involving several other industries performed by Jain (2008) reveals. In the survey, most of the participants report using either simple trend/simple average models (57%) or models based on exponential smoothing (29%). The reason lies in the ability of the simple methods to make adequate forecasts even on limited historical data by reducing the danger of overfitting on the training data, because the number of parameters to be estimated is small.

The next section describes algorithms currently used along with alternative approaches implemented for the experiments presented at the end of this chapter.

2.1.4.2 Reference curves

In general, forecasting at LSB makes use of reference curves for modelling the typical booking and cancellation behaviour of customers. They are learnt on the finest possible level for each DCP and are periodically updated with actual bookings and cancellation rates. Figure 2.2 shows examples of aggregated booking and cancellation reference curves. Cancellation information is usually represented as cancellation rates by dividing cancellation numbers by bookings.

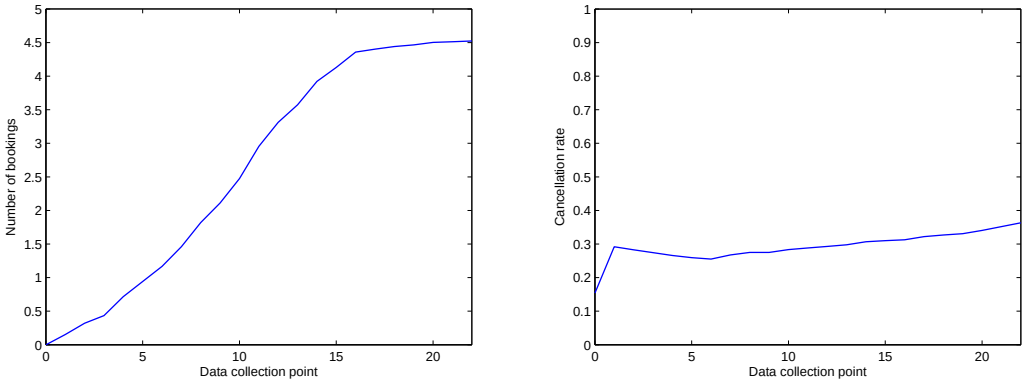

Figure 2.2: Example of reference curves, bookings (left) and cancellation rate (right), values given for each of the 23 DCPs prior to departure.

Reference curves are periodically updated in a history building step. If a new flight is introduced, an initial reference curve is generated using similar flights. The basic calculation uses an exponential smoothing approach to calculate the new value $\text{ref}_{\text{new}}(dcp)$, taking into account both the previous value of the reference curve at the same DCP $\text{ref}_{\text{old}}(dcp)$ and the current observation $\text{cr}(dcp)$, weighted by a smoothing factor α, which controls how much the curve adapts to newly available data. This is described in Equation 2.1.

$$\text{ref}_{\text{new}}(dcp) = \alpha \cdot \text{cr}(dcp) + (1 - \alpha) \cdot \text{ref}_{\text{old}}(dcp) \tag{2.1}$$

One of the problems arising in this context is the update of early parts of the cancellation rate reference curves, which only happens rarely: as long as booking numbers are zero and no rate can be obtained, the value of the old reference curve stays unchanged. To overcome this issue, changes in the reference curve at later data collections points are to a certain extent also used for updating the previous ones.

2.1.4.3 Unconstraining

As mentioned in the introduction, fareclasses are closed and opened during the optimisation process. This influences booking numbers, as bookings are not registered for a closed fareclass even though the demand may exist. Accepted bookings and cancellations are thus *constrained*, however, "unconstrained" numbers are needed for forecasting. Unconstraining is the process of eliminating the influence of capacity control from the data by approximating complete demand and cancellations. For bookings, unconstraining uses the reference curve in a simple additive manner to estimate demand that occurs during the time a fareclass is closed. For cancellations, a rate is calculated as a weighted sum of the actual cancellation rate applied to the actual bookings and the reference rate applied to the approximated rejected bookings. Consequently, for fareclasses that have at some point been closed before flight departure, the data used for evaluation and forecasting purposes is not real data, but an approximation.

2.1.4.4 Booking forecast

The central question of booking forecasting is how many people would make a booking if it was accepted. An important concept for forecasting the booking time series is decomposition. It is based on the assumption that time series are aggregates of a number of components, which can be modelled independently as separate time series. This approach is widely used in airline revenue management applications. In the most popular version, a time series is decomposed into a basic level, a trend and a seasonal component. The splitting of a series according to different factors allows separate treatment of each of the sub-series, with model and parameter choices being simpler and more adequate to the specific characteristics of a component.

At LSB, two major components are distinguished for bookings: the *attractiveness* represents a stable base component, subject only to general long term influences like demographic and economic conditions or time slot of the flight. Other influences only have a short term effect, some of which cannot be predicted using historic data, for example if they occur due to special events like a football world championship. However, other short term influences like seasonal behaviour can very well be modelled by examining the past. A previous project by Riedel (2007) provides a more detailed look at demand forecasting, with the focus on improving seasonality predictions as a big impact factor for final forecast accuracy. Overall accuracy improvements of 11% have been achieved.

2.1.5 Lufthansa Systems cancellation forecasting

Accurate cancellation predictions are vital for obtaining an accurate final net booking forecast, which is given by the difference of the booking and cancellation forecasts. In general, calculations concerning cancellations are carried out using cancellation rates, i.e. the number of cancellations divided by the number of bookings,

which has been shown to lead to more stable results in comparison to dealing with absolute cancellation numbers. This section first introduces the important concept of confidence limits for cancellation rates before moving on to the description of three traditional, rate-based algorithms.

2.1.5.1 Confidence limits

The small number issue described earlier introduces the problem of instability of cancellation rates. If, for example, only one booking exists in a specific fareclass, cancellation rates can be as extreme as zero or one, depending on whether or not the booking is cancelled. To prevent unstable cancellation rates especially at early data collection points where booking numbers are usually low, confidence limits[1] are introduced and used to constrain the currently observed cancellation rate for both history building and forecasting to a certain range around the reference curve. Figure 2.3 shows the concept: an upper limit restricts cancellation rates that are much higher than the reference curve, a lower limit does the same for cancellation rates that are too low in comparison to the historically learnt behaviour. For the calculation, the following guidelines apply:

The confidence limits get wider (less restrictive) with

- an increasing number of bookings, as more bookings lead to a more stable cancellation rate,
- an increasing DCP, as data from DCPs closer to departure is more trustworthy and
- decreasing difference between expected bookings and already accepted bookings.

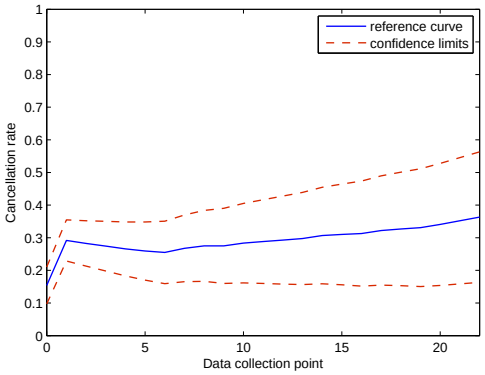

Figure 2.3: Reference curve and confidence limits

2.1.5.2 Forecasting based on cancellation rate reference curves

Based on the current cancellation rate cr_t and reference curve ref_t, t being the time index, two approaches exist for calculating the forecast. An additive approach is employed when

[1] Confidence limits are not to be confused with confidence intervals, which have a completely different meaning.

- $\text{cr}_t < \text{ref}_t$ and ref_t is ascending or
- $\text{cr}_t > \text{ref}_t$ and ref_t is descending

and a multiplicative approach is used in case

- $\text{cr}_t > \text{ref}_t$ and ref_t is ascending or
- $\text{cr}_t < \text{ref}_t$ and ref_t is descending.

The additive approach following Equation 2.2 enforces a bigger adjustment by just adding or subtracting the appropriate values of the reference curve, while the multiplicative approach ensures that the results are not below zero or above one by more gentle adjustments given in Equations 2.3 and 2.4, with index $t+h$ being the current time t plus the forecasting horizon h.

$$\hat{\text{cr}}_{t+h} = \text{cr}_t + (\text{ref}_{t+h} - \text{ref}_t) \tag{2.2}$$

$$\hat{\text{cr}}_{t+h} = 1 - \left[\frac{1 - \text{ref}_{t+h}}{1 - \text{ref}_t} \cdot (1 - \text{cr}_t)\right] \text{ for } \text{cr}_t > \text{ref}_t \tag{2.3}$$

$$\hat{\text{cr}}_{t+h} = \frac{\text{ref}_{t+h}}{\text{ref}_t} \cdot \text{cr}_t \text{ for } \text{cr}_t < \text{ref}_t \tag{2.4}$$

Default cancellation rates that are used to initialise reference curves are generated from similar flights and have been provided by LSB. Three ways of updating them are currently implemented:

- A model based on single exponential smoothing, where a new forecast is generated by adjusting the previous one by the error it produced. Reference curves are updated using Equation 2.1.
- A model based on Brown's double exponential smoothing according to Brown et al. (1961). The updated value ref_t is calculated using the equations below, with L being the value (level) of the series, T the trend component and α again the constant smoothing factor.

$$\begin{aligned} \text{ref}_t &= L_t + T_t \\ L_t &= \alpha \cdot \text{cr}_t + (1 - \alpha) \cdot (L_{t-1} + T_{t-1}) \\ T_t &= \alpha \cdot (L_t - L_{t-1}) + (1 - \alpha) \cdot (T_{t-1}) \end{aligned} \tag{2.5}$$

- A regression model that uses a regression line fitted to past observations of the same DCP. The new value for the reference curve is calculated by extrapolating the regression line into the future, a value for the trend curve is obtained from its slope.

Similar to the current algorithm used for the booking forecast, the current cancellation forecast is a mixture of some of these algorithms, with details being confidential. The forecasts of the currently used method will however be used as a baseline for comparison in the empirical experiments and will furthermore be referred to as the LSB cancellation forecast.

2.1.5.3 Probability forecast

In meetings with LSB, it was brought up that another cancellation forecast based on probabilities might be able to produce accurate results while still being simple and robust. As a reaction, a fourth forecast has been implemented in the scope of this work to add a functionally different approach to the method pool. The central question here is: for a booking at a certain DCP, what is the probability that the booking is cancelled at following DCPs if it has not been cancelled before?

The algorithm is based on the assumption that bookings occurring at different DCPs will have different probabilities of being cancelled prior to departure of the flight. It was, for example, previously observed, that bookings from early DCPs tend to be cancelled more often than the bookings from later ones. Similarly to the previously introduced approaches, reference curves are generated in the history building step. In the probability reference curve, each booking is assigned to a DCP, and the probability that it will be cancelled on a later DCP is given. A simplified example in form of a table is shown in Table 2.2. It says, for example, that a booking made at DCP0 will be cancelled at DCP1 with a probability of 0.15, if it has not been cancelled before. A booking from DCP2 obviously does not have probabilities for cancellation at DCP0 and 1, as it only occurred at a later DCP.

	P(Canc. at DCP0)	P(Canc. at DCP1)	P(Canc. at DCP2)
Booking at DCP0	0.1	0.15	0.1
Booking at DCP1	-	0.2	0.1
Booking at DCP2	-	-	0.15

Table 2.2: Simplified example of a cancellation probability reference curve.

Apart from a few special cases where booking numbers are extremely small, it is not possible to tell which occurring cancellation belongs to which booking. This is why the previously generated probabilities $P(j,t)$, giving the probability that a booking from DCP j will be cancelled at DCP t, $j \leq t$, are used to distribute cancellations to previous bookings as demonstrated in Equations 2.6 and 2.7. Assume a number of cancellations c_t occur at DCP t. All remaining bookings b_i at the previous DCPs $0 \leq i \leq t$ are then adjusted according to Equation 2.6 in case that the number of cancellations is greater than the sum of the product of relevant bookings and probabilities ($\sum_{j=0}^{t} b_j P_{j,t} > c_t$) and according to Equation 2.7 otherwise.

$$b_i^{new} = b_i - \lambda(b_i \cdot P_{i,t}), \quad \lambda = \frac{c_t}{\sum_{j=0}^{t} b_j P_{j,t}} \tag{2.6}$$

$$b_i^{new} = b_i - b_i(1 - \lambda(1 - P_{i,t})), \quad \lambda = \frac{\sum_{j=0}^{t} b_j - c_t}{\sum_{j=0}^{t} b_j (1 - P_{j,t})} \tag{2.7}$$

The cancellation probability that is eventually used for updating the probability reference curves is then obtained by dividing the new value of remaining bookings b_i^{new} by the old value b_i and subtracting it from 1. Initial probability reference curves are generated in a history building period, where cancellations are evenly distributed on previous bookings.

The forecasting process then consists of two steps: for each DCP, the current cancellations are first distributed to past bookings with the same algorithms as

used in the history building. Secondly, the booking reference curve estimates the development of the bookings for the future DCPs. Applying the probabilities from the reference curves and summing up the remaining bookings creates the final net booking forecast.

In summary, it can be said that demand and cancellation forecasting methods used at LSB have been heavily tuned to account for application-specific characteristics and requirements. The basis of the presented algorithms are simple averages and exponential smoothing approaches. However, many more methods have been investigated in the scientific literature, which will be reviewed in the remaining sections of this chapter.

2.2 Time series forecasting

This section looks at forecasting from a more general point of view and investigates traditional time series forecasting to provide the basis for the remainder of the thesis, aiming at organising and highlighting the most important methods and results of half a century's research done in this area.

The most basic time series forecasting method is called the naive forecast and sets the forecast to the last time series observation. Another simple method is the moving average, where the forecast is the arithmetic mean of the most recent values of the time series, discarding old and potentially inapplicable observations. Beyond these, the first more sophisticated methods date back to the 1950's and 1960's, with new approaches and extensions constantly being investigated until today. This section provides a literature review of the most important and popular approaches to time series forecasting. The choice of publications to cite has been difficult, as a vast majority of contributions are smaller case studies applied to a specific application area, which mostly provide results that contradict each other. As individual forecasting algorithms are not the primary focus of this thesis, the literature review will be confined to five big groups of forecasting algorithms, discussing the seminal contributions, publications on genuinely new forecasting algorithms and extensive review papers. Only a selection of the approaches mentioned here are actually implemented for the empirical studies in this thesis, which is why equations and more detailed descriptions of only these will be given along with the methodology of the experiments in Section 3.2.2.

2.2.1 Exponential smoothing

Exponential smoothing methods apply weights that decay exponentially with time and thus also rely on the assumption that more recent observations are likely to be more important for a forecast than those lying further in the past. Smoothing methods originated in the 1950's and 1960's, with the methods of Brown, Holt and Winter still being of considerable importance today as summarised and referenced in Makridakis et al. (1998). The seemingly only originally new smoothing method since the classic approaches was introduced by Taylor (2003), who suggested using a damped multiplicative trend; details are given in Section 3.2.2.1. A taxonomy of exponential smoothing methods has first been presented by Pegels (1969), distinguishing between nine models with different seasonal effects and trends, which can be additive, multiplicative or non-existent. Gardner (1985) extended this classification by including damped trends, increasing the number of models to twelve.

Abraham & Ledolter (1986) showed that some exponential smoothing methods arise as special cases of ARIMA models. Apart from this, these methods have been lacking a sound statistical foundation for a long time, which prevented a uniform approach to calculation of prediction intervals, likelihood and model selection criteria. Hence, many publications were concerned with investigating the stochastical framework of the exponential smoothing methods. The most thorough and recent work in this context has been published by Hyndman et al. (2002), who fitted all of the twelve models of Gardner (1985) into a state space framework, giving state space equations for each of them using both an additive and a multiplicative error approach. They furthermore fit a model selection strategy to the framework in order to allow for automatic forecasting.

A recent variation drawing some attention is the Theta-model proposed by Assimakopoulos & Nikolopoulos (2000). It decomposes seasonally adjusted series into short and long term components by applying a coefficient θ to the second order differences of the time series, thus modifying its curvature as described in Section 3.2.2.1. Hyndman & Billah (2003) show that this method is equivalent to single exponential smoothing with drift.

Extensive state-of-the art reports on exponential smoothing can be found in Gardner (1985) and Gardner (2006), citing over 100 and 200 relevant papers, respectively.

Exponential smoothing methods have a reputation of performing remarkably well for their simplicity as summarised in Gooijer & Hyndman (2006). In an extensive competition conducted by Makridakis & Hibon (2000), the authors recommend Taylor's exponential smoothing method with dampened trend as a method that is very easy to implement and gives robust performance. Chatfield et al. (2001) argue that the robust nature of these models is due to the fact that they are the best choice for a large class of problems. Hyndman (2001) picks up on that and adds that more complex models are subject to performance instabilities caused by a more complex model selection and parameter estimation process, which exponential smoothing models do not suffer from to this extent.

2.2.2 ARIMA models

One of the most influential publications in the area of time series forecasting is Box & Jenkins (1970), having an extraordinary impact on forecasting theory and practice until today. The authors introduced the group of autoregressive integrated moving average (ARIMA) models, which can simulate the behaviour of diverse types of time series. An ARIMA model consists of an autoregressive and a moving average part whose orders have to be estimated and involves a certain degree of differencing; general equations are given in Section 3.2.2.2.

Selection of an appropriate model can be done judgementally. A strategy mainly based on examining (partial) autocorrelation values can be found in Makridakis et al. (1998). Alternatives have been suggested, for example using information criteria like Akaike's information criterion (AIC)[2] introduced in Akaike (1973) and Bayes information criterion (BIC)[3] introduced in Raftery (1986). More recent publications mostly apply ARIMA models in a hybrid approach, for example in combination

[2] $AIC = 2k - 2 \cdot ln(L)$, with k being the number of parameters in the model and L the maximized value of the likelihood function for the estimated model.

[3] $BIC = -2 \cdot ln(L) + k \cdot ln(n)$, where k and L are the same as for the AIC and n denotes the number of observations available.

with neural networks as in Zhang (2004) and Koutroumanidis et al. (2009) or as an individual method as part of a more general combination approach as in Anastasakis & Mort (2009).

The performance and benefits of ARIMA models has been fiercely discussed in the aftermath of the M3-competition, whose results have been published in Makridakis & Hibon (2000). In this publication, the organisers criticise the approach of building statistically complex models like the ARIMA model, disregarding all empirical evidence that simpler ones predict the future just as well or even better in real life situations, for example provided by their competition. Makridakis et al. (1998) furthermore add that the only advantage a sophisticated model has compared to a simple one is the ability to better fit historical data, which is no guarantee for a better out-of-sample performance. Results of the M3 competition are discussed from a more general point of view in the next chapter.

2.2.3 State-space models

State space models provide a framework that can accommodate any linear time series model. The seminal work in this area was published by Kalman (1960), giving a recursive procedure for computing forecasts known as the Kalman filter. Originally mainly used in control and engineering applications, its usage for time series forecasting only started in the 1980's according to Gooijer & Hyndman (2006). Generally, two equations are part of a state space model: the observation and the state equation. While the state equation models the dependency of the current to the previous state, the observation equation provides the observed variables as a function of the state variables. A state-space model was implemented for the empirical experiments of this thesis as described in Section 3.2.2.3.

Gooijer & Hyndman (2006) mention that publications from practitioners concerning the use of the state space framework for time series forecasting applications are surprisingly rare, even though some books do exist, for example Durbin & Koopman (2001). Two significant contributions can however be mentioned: Harvey (2006) provides a comprehensive review and introduction to treating "structural models", i.e. time series given in terms of components such as trends and season, with the help of the state space concept. As already cited above, Hyndman et al. (2002) fit exponential smoothing methods into state space framework, providing them with a statistical foundation.

Comparing the performance of state-space models is not straightforward by looking at available literature, as they do not seem to appear in the major forecasting competitions. Since a large number of models have state-space formulations, the performance will naturally vary according to the model used. However, the benefits related to their statistical theory, e.g. well-defined strategies for model selection, likelihood estimation and prediction interval calculation are bound to make them attractive for forecasting researchers and practitioners.

2.2.4 Regime switching

Regime switching models are a class of model-driven forecasting methods originally introduced by Tong (1990). Most of them belong to the class of so-called self-exciting threshold autoregressive models (SETAR) and their variants, where a number of regimes consists of one autoregressive model each. The order and coefficients of the autoregressions vary for each regime. Which set of equations to apply for a

forecasting situation is then determined by trying to identify the regime or state the system is likely to be in, which is normally done by looking at past values of the time series, hence the term "self-exciting". Some SETAR models have been compared in Clements & Smith (1997).

As abruptly changing regimes are not always desirable, smooth switching of regimes is promoted in smooth transition autoregressive (STAR) models, where switching can, for example, be done with a logistic function. A publication by van Dijk & Franses (2000) gives a survey of developments in this area at this time. More recent work was published in Fok et al. (2005), where a STAR model is complemented with a meta-model linking parameters of the logistic switching functions to state characteristics and other regressors.

Empirical evaluation of regime switching models has mainly been conducted in comparison to neural networks and linear models and produced mixed results. In Stock & Watson (2001), STAR models generally performed worse than neural networks and did not outperform linear models either. Teräsvirta et al. (2004) however question results of this study, stating that nonlinear forecasting methods should only be considered at all if the data shows nonlinear characteristics. A re-examination of the performances of linear, STAR and neural network approaches on time series that rejected the statistical test for linearity has consequently been carried out. By a small margin, the STAR model had the best performance. Marcellino (2005) finds that STAR models generally outperform linear models, performance evaluation comparing them to neural networks however remains undecided.

2.2.5 Artificial neural networks

Looking at the area of computational intelligence models, it is neural networks that have most frequently and successfully been used for time series forecasting purposes. Neural networks represent a nonlinear data-driven technique and can, as universal approximators, approximate any continuous function to any required accuracy without the need of extensive knowledge on the underlying data generation process or modelling relationships explicitly. They do however come with the well-known risks of over-parametrisation, overfitting and the issue of choosing an optimal topology.

An extensive summary of work done in the area of multilayer perceptrons can be found in Zhang et al. (1998), which is somewhat outdated but still frequently cited and very relevant in terms of guidelines given. Not only reviewing a large number of related publications, the authors also make recommendations concerning network architecture (number of hidden/input and output nodes and their interconnection), activation functions, training algorithms, data normalisation, training/testing sample size and performance measures. Another comprehensive literature survey, albeit limited to the application area of electrical load forecasting, can be found in Hippert et al. (2001), who motivated more rigorous research in the area by stating that many publications present seemingly misspecified models that have not been sufficiently tested.

Work on other types of neural networks for time series forecasting seems sparse, although existent. The main focus lies on employing evolutionary algorithms for various purposes, for example for training a recurrent Elman network as described in Cai et al. (2007). Rivas et al. (2004) suggest evolving the number of hidden nodes, centres and radii for a radial basis function network. For the empirical study in this thesis, two neural networks have been implemented, with further information given in Section 3.2.2.4.

Following the initial enthusiasm about applying neural networks to time series forecasting, more critical voices like Chatfield (1995) soon began pointing out that results of empirical studies have always been both discouraging and encouraging, which applies till today. One important issue has been pointed out by Zhao et al. (2003): due to many parameters and architectural choices involved in an application of neural networks, it is hard to replicate results of important studies, especially if the description of the methodology does not provide sufficient details. Despite considerable effort and using the same data sets and the same setup, the authors failed to even come close to obtaining similar results to one of the most cited publications in this area written by Hill et al. (1996). Since the examined work represented one of the many studies where neural networks performed extremely well, the need for a transparent methodology and a critical evaluation of empirical studies becomes obvious, even more so because of the black-box nature of neural networks. Zhang (2007) looks at common mistakes in the design of experiments with neural networks from a general perspective, but frequently mentions issues specific to time series forecasting. The number of empirical studies on neural networks with varying degrees of success remains large till today, a big percentage of which seem to deal with electricity load forecasting as, for example, in Hippert et al. (2005). No general implications can be given on their performance apart from that there seem to be many open questions regarding their use and implementation.

2.3 Forecast combinations

More than one forecast for the same variable is often available, leading to the question if one should choose one single model or try to combine several to obtain a forecast with more accuracy. In the 30 years that passed since the seminal paper on forecast combination by Bates & Granger (1969), an impressive amount of work has been done in this area. Extensive reviews and summaries can be found in Clemen (1989), Granger (1989) and Timmermann (2006).

The value of a target variable y at time t is to be predicted[4]. Let $\{\hat{y}_1, \hat{y}_2, ..., \hat{y}_m\}$ be m individual forecasts for that variable. The linear combination forecast $\hat{y}^c$ can then be given by their weighted linear sum

$$\hat{y}^c = \sum_{i=1}^{m} \omega_i \hat{y}_i. \tag{2.8}$$

Linear models were the first proposed combination models in the 1960's and 70's and remain very popular until today. More recent related publications do generally not propose new linear models, but address adaptivity of the models and parameters and the situations in which a particular model should or should not be applied, which are issues addressed later in this thesis.

Potentially nonlinear relationships among forecasts are not considered in linear forecast combination, providing the main argument for usage of nonlinear combination methods. A nonlinear approach combines individual forecasts by a nonlinear function Ψ:

$$\hat{y}^c = \Psi(\hat{y}_1, \hat{y}_2, ...\hat{y}_m) \tag{2.9}$$

[4]The considerations can be applied to the more general case of predicting the target variable $y_{t+h,t}$ for time $t+h$ at time t. The time indices have been omitted for notational convenience.

Many ways to calculate the linear combination weights ω_i (or the weight vector $\boldsymbol{\omega}$ in case of vector notation), or to design function Ψ have been proposed in the scientific literature. This section provides a literature review on linear and nonlinear forecast combination to complete the background information necessary for the main part of this thesis.

2.3.1 Nonparametric methods

The simplest way to compute linear combination weights ω_i is by taking the simple average of all m available individual forecasts, giving each of them the weight

$$\omega_i = \frac{1}{m} \tag{2.10}$$

This is particularly attractive if the length of the time series is short in comparison to the number of individual forecasts, because combination weights derived from these short samples tend to be unstable as last shown by Smith & Wallis (2009).

Techniques based on ranks of the individual forecasts have first been proposed by Bunn (1975) who expressed combination weights as probabilities for a model producing the lowest loss in his outperformance model. The probabilities are estimated using the proportion of times a method has performed best up to the current time. This approach has for example been adopted by Aiolfi & Timmermann (2006), using the equation

$$\omega_i = \frac{R_i^{-1}}{\sum_{j=1}^{m} R_j^{-1}} \tag{2.11}$$

where R_i is the rank of model i based on its past performance.

Ranking-based approaches are comparatively robust to outliers in forecast performance and promise a stable combination performance especially if data is sparse, unstable or nonstationary and approaches based on the statistical moments of the error distribution provide unreliable results. However, they ignore correlations between forecasts and the extent of the differences in the relative performance, which are aspects considered in the next group of approaches.

2.3.2 Variance-covariance based methods

As originally suggested by Bates & Granger (1969), a combination approach can calculate weights in order to minimize the combination error variance using the $(m \times m)$ covariance matrix of forecast errors $\sum$ according to the equation

$$\boldsymbol{\omega} = \frac{\sum^{-1} \boldsymbol{e}}{\boldsymbol{e}' \sum^{-1} \boldsymbol{e}} \tag{2.12}$$

where $\boldsymbol{\omega}$ is the combination weight vector and $\boldsymbol{e}$ denotes the $(m \times 1)$ unit vector. In this so-called "optimal model", the results will be optimal with regard to the error measure used if each individual forecast is unbiased. The drawback is the difficulty to reliably calculate a covariance matrix: it can of course be estimated based on past error variances, which leads to high estimation errors if the data sample is changing or not stationary, or if the time series is short and many individual forecasts are to be combined. Many authors, most recently Smith & Wallis (2009), hence suggested to disregard correlations among forecast errors and only take individual error variances into account. In Stock & Watson (2001), this idea was implemented using the mean

squared error (MSE) which is calculated for each model i by taking the average of the sum of the past squared forecast errors over a certain period of time with the length v:

$$MSE_i = \frac{1}{v} \sum_{j=t-v}^{t-1} \epsilon_{i,j}^2 \tag{2.13}$$

The forecast error ϵ is given by the difference between forecast and realisation of the corresponding target variable $\epsilon_i = \hat{y}_i - y_i$. Past MSE performance is raised to various powers of a parameter k, with m again being the number of individual forecasting models:

$$\omega_i = \frac{MSE_i^{-k}}{\sum_{j=1}^{m} (MSE_j)^{-k}} \tag{2.14}$$

In this approach, setting $k = 0$ produces an equally weighted combination, $k = 1$ weights forecast by the inverse of their MSE. With increasing k, an increasing weight is assigned to models that performed well.

Forecasts of the historically best performing cluster are then averaged to obtain a final forecast. A relatively recent approach to the combination of forecasts was introduced by Aiolfi & Timmermann (2006). Following empirical results saying that a good or bad performing forecast is more likely to keep performing well or badly instead of changing its performance, they group a number of forecasts that are diversified in functional approach and model parameters in two or three clusters using a k-means algorithm on their past error variance. Forecasts are then pooled within the groups before combining them with one of the following strategies:

- selecting the previously best performing cluster and averaging the forecasts contained in it,
- excluding the cluster that performs worse and averaging forecasts from the other clusters,
- combining forecast averages of each of the clusters using least squares regression or
- doing the same as in the previous bullet point but shrinking weights towards equal weights.

2.3.3 Regression

In regressing realizations of the target variable on forecasts over past periods, linear combination weights can be estimated by least squares approaches. Let $\hat{\mathbf{y}}_j$ be the $m \times 1$ column vector of forecasts from the m different models and y_j the observation at time j. The least squares estimator for the $m \times 1$ weight vector $\boldsymbol{\omega}$ is given by

$$\boldsymbol{\omega_t} = (\sum_{j=t-v}^{t-1} \hat{\mathbf{y}}_j \hat{\mathbf{y}}_j')^{-1} \sum_{j=t-v}^{t-1} \hat{\mathbf{y}}_j y_j. \tag{2.15}$$

A linear regression has the general form

$$\hat{y}^c = \omega_0 + \boldsymbol{\omega}' \hat{\mathbf{y}} + \epsilon \tag{2.16}$$

where ω_0 is an intercept value, ϵ the error term and $\hat{\mathbf{y}}$ again the column vector of forecasts. Granger & Ramanathan (1984) first evaluated three different versions of the regression approach: a standard regression with an intercept, a standard regression without an intercept and a restricted regression without intercept where the combination weights sum to one. The authors suggest to use unrestricted combination weights and to include an intercept term because in that case, any bias in the individual forecasts can be compensated for with the intercept term. Furthermore, the authors of the same publication showed that the optimal model presented in the previous section is analogous to a least squares regression with weights constrained to sum to one, proposing to discard the optimal approach favouring the simpler and more flexible regression method. In de Menezes et al. (2000) the usage of unrestricted weights is challenged quoting Diebold (1988), who states that the resulting combination errors are likely to be serially correlated, indicating inefficient combination weights.

2.3.4 Nonlinear combinations

Although nonlinear models are widely known and popular for individual forecasting techniques, literature on nonlinear forecast combination seems sparse, a fact that has also been mentioned by Timmermann (2006). One reason can be found in the error potential related to parameter estimation, which is already large for linear models and which is even larger for nonlinear ones that usually come with higher degrees of freedom and more parameters to estimate. This section provides a review of the few publications identified on combination techniques using neural networks, self-organising algorithms, fuzzy systems and genetic programming approaches. A more extensive review has been previously published in Lemke & Gabrys (2007).

2.3.4.1 Neural Networks

The general idea of neural network forecast combinations is straightforward: an instance of the training set consists of forecasts of the individual models at a certain time in the past. The output is the observed value of the variable to be forecasted at the time. The neural network (regression type) then approximates the nonlinear function Ψ in the training period and can be used for obtaining a combined forecast afterwards.

The first use of neural networks for combining forecasts appears in Shi & Liu (1993). The authors use a four layer feedforward network trained with the backpropagation algorithm using only one time series with 120 values. Three individual forecasts obtained using one exponential smoothing, one trend analysis and one ARIMA model are combined. Diagrams imply that the neural network has been trained well and shows a good out-of-sample performance. Numerical results of the same experiment can be found in Shi et al. (1999), showing that the neural network performs extremely well. Details on the architecture have however been omitted in the publications.

Less favourable results can be found in more extensive studies. In Donaldson & Kamstra (1996), a data set consisting of four macroeconomic series over almost 18 years is used. Two individual forecasts are combined in a neural network with one hidden layer that consists of four nodes using one linear and three nonlinear logistic transfer functions. Furthermore, two linear combinations are used for comparison. Looking at a squared error measure, the neural network combinations perform at

least as well as individual forecasts and similar to the linear combinations. This is however not the case for an absolute error measure. On another note, the authors point out that neural networks are the only model in the evaluation that was not encompassed[5] by at least one other forecast. The same experimental setup with similar results is used in Harrald & Kamstra (1997), with the difference that some parameters of the neural network are determined using evolutionary programming.

Another empirical study can be found in Hu & Tsoukalas (1999). Four individual forecasts are combined with a one layered neural network with four nodes, using a data set of one and a half years of financial data including a crisis period. The neural network performs particularly well in the crisis period, but, in contrast to the results described in the previous paragraph, a simple average of individual forecasts or individual models themselves always outperform this approach if squared prediction errors are used for comparison. If the performance of the neural network is assessed by absolute error values, it gets significantly better, but it is still not always superior.

In Donaldson & Kamstra (1999), the authors explain potential benefits of the neural network approach graphically, plotting surfaces of the combining functions in relation to two input forecasts. While linear approaches create a flat surface with a constant response to changing input forecasts, a neural network shows a highly flexible response function where the impact of a change in one of the individual forecasts is influenced by the value the other forecast assumes at the same time. Neural network combination models can thus account for interactions between individual forecasts which would be omitted when using linear combination schemes.

More recent work can be found in Liu (2005) and Ozun & Cifter (2007), which successfully apply neural networks trained with genetic algorithms for forecast combinations in the application area of predicting market risk.

2.3.4.2 Self-organising algorithms

Research on a popular inductive self-organising method called the Group Method of Data Handling (GMDH) was started by Ivakhnenko (1970). This algorithm aims at finding the structure of a model by generating candidates in an iterative process, sorting out possible solutions according to an external criterion in each step.

An application of this concept to forecast combination can be found in He & Xu (2005). A so-called statistical learning network is built, starting with individual forecasts as input variables. A transfer function, usually a polynomial, is used to generate a first layer consisting of a set of model candidates. Parameter estimation for the models takes place on training data, for example using regression. The resulting candidates are ranked according to an external criterion, which could be the mean squared forecast error on unseen test data. Only models that are able to improve the value of the external criterion are selected as inputs for the subsequent layer, where the generation, estimation and selection process starts again. The value of the external criterion will pass through a minimum, which is when the optimal model has been found.

Only a very small empirical experiment is presented to evaluate the performance of the self-organising approach, just one macroeconomic series over 20 months is used. Compared to a neural network combining approach, all individual models and

[5] If a forecast encompasses rival forecasts, it includes all the information other models give and dominates them, making them redundant for a forecast combination. Section 5.3.1.4 gives more details of the encompassing concept.

a linear least squares combination, the proposed algorithm outperforms every single one of them.

In comparison to neural networks, the self-organising approach has several advantages. It does not require time-consuming training and gives an explicit model of the system. Furthermore, the topology of the network does not need to be determined in advance, which is still an issue of ongoing research in the area of neural networks.

2.3.4.3 Fuzzy systems

Fuzzy systems account for vagueness in knowledge and situations by using membership functions that translate values of input variables into values representing their degree of truth for belonging to a concept. Based on these, the degree of applicability of each rule in an expert system is calculated before a defuzzification procedure translates these applicabilities back to a crisp output value.

Fuzzy systems for forecast combination can be found following two different paradigms. Fiordaliso (1998) use fuzzy systems similar to regime models, with the difference that two or more different forecasting models can be active at one time. They apply a first order Takagi-Sugeno fuzzy system, which represents conclusions in the rules by linear functions of the input variables. Parameters in the system are estimated using the gradient descent algorithm to minimise its squared forecast error. Tests are carried out using three time series with 200 to 312 values, combining one ARIMA and one k-nearest neighbours model. Looking at four different error measures, the fuzzy system almost always outperforms or draws level with the individual forecasts and linear forecast combination methods. The authors furthermore suggest using a linear mixture of models, which is simpler than one working with neural networks, however, they do admit at the same time that the computational complexity of building their system is very high and suggest simplifications.

Two more publications emphasise a different aspect of fuzzy systems - the possibility of modelling linguistic and subjective knowledge. A case study favouring a fuzzy logic combination system for forecasting the demand for signal transmission products is presented in Frantti & Mähönen (2001). A similar hierarchical rule base combining subjective forecasts from experts with time series forecasts is presented in Petrovic et al. (2006), the individual time series forecasts being provided by an ARMA and a decomposition forecasting model. Only a hypothetical example is used for evaluation, significantly improving the individual time series forecast error. Noteworthy about the presented system is its periodically adapting rule base, which adjusts confidence in rules according to their past performance.

2.3.4.4 Genetic Programming

Riedel & Gabrys (2005) look at forecast combination for airline ticket demand data. In this area, forecasts are available on different levels because of the complex network of flights, routings and itineraries which contain different fareclasses or points of sale. A large number of predictions can thus be generated on these various levels with different forecasting models. To avoid the drawbacks that arise from using too many input forecasts without taking the risk of choosing among them, the authors propose hierarchical structures for forecast combination. Genetic programming is used to evolve the tree-like combination structures where individual forecasts are represented by leafs, different combination methods by nodes and the final combined forecast

is obtained at the root. As the fitness function, i.e. the criterion to optimize, the mean absolute deviation forecasting error on a previously unseen test data set has been chosen. The initial population is either a random subset or selected by experts; the standard operators for genetic programming are applied in the following way: crossover exchanges two subtrees and mutation exchanges a node or leaf. Variations of this algorithm are assessed on demand rates for airline tickets, leading to consistent improvements compared to the simple average combination.

2.3.5 Adaptivity

A constantly changing environment is a typical characteristic for an area in which forecasts are applied. Changes can be of different nature; one type of change is structural breaks in the data generation process, caused by events like macroeconomic shocks or an alteration of the political situation. Other changes are smoother and can for example be a function of the state of the economy. Assuming that no individual model can be a perfect model of the true data generation process and considering that each individual model has a different speed to adapt to changes, there is a reason to believe that forecast combination will perform well whenever adaptivity is needed. Hendry & Clements (2002) support this hypothesis for simple combination methods in the presence of structural breaks, considering a number of designs for the breaks that occur in a simulation. Terui & van Dijk (2002) contrast an individual autoregressive forecast with time varying weights with time varying forecast combinations, coming to the conclusion that the latter are always superior.

2.3.5.1 Time-Varying Combination Weights

Motivated by the fact that the performance of individual forecasts changes over time as, for example, shown in Aiolfi & Timmermann (2006), many publications investigate time varying combination weights. One of the initial papers on that matter by Diebold & Pauly (1986) proposes modelling a bigger impact of more recent observations and letting the combination weights be a function of time. Results of a simulation study shows significantly decreasing error values compared to an ordinary least squares regression based combination approach, but no further work with studies on real data has been found though announced in the conclusions.

One plausible method in the context of regression and variance-covariance based methods is using a moving window of fixed size to determine the number of latest data collection points to include in the calculation, as first thought of by Bates & Granger (1969) and Granger & Ramanathan (1984). Structural breaks can degrade the performance of these approaches; Pesaran & Timmermann (2007) proposed a varying window size following a known structural break by minimizing the expected mean forecast error in an iterative procedure. The approach is presented for forecasting a single autoregressive time series, but could be extended to guide forecast combination weights.

Deutsch et al. (1994) apply regime switching models to combine forecasts for the US and UK inflation rates. Two sets of combination weights for two forecasts are estimated, establishing two different regimes. The regime the economy might be in is then determined dependent on different functions of the lagged forecast error of the alternative model. The best performing method is reported to be related to the relative size of one of the forecast errors and reduces the out-of-sample MSE compared to the results obtained by linear combination of the models. The indi-

vidual forecasts are always outperformed. The more extensive empirical evaluation of Elliott & Timmermann (2005) gives contradictory results for the same approach. Out of six macroeconomic series investigated, this method only outperforms both of the individual forecasts in 2 out of 6 cases.

A different approach with even less promising results is followed by Terui & van Dijk (2002) and Stock & Watson (2004). Both use a state-space approach with a regression using time-varying weights in the measurement equation and evolve the weights using a random walk in the state equation. State and weights are then computed using the Kalman Filter using an expanding window of observations. Terui & van Dijk (2002) report results of an empirical study based on two data sets for natural sciences and 16 macroeconomic series, where the time varying model fails to consistently outperform the individual forecasts out-of-sample. However, the authors remark that the combined forecasts work well for highly nonlinear series. Stock & Watson (2004) report that too much adaptivity in the parameters worsens performance and stability of the combined forecasts, while least possible adaptivity only yields small improvements compared to simple averaging.

Elliott & Timmermann (2005) compare several time-varying and static forecast combinations, including the method by Deutsch et al. (1994), a rolling regression and a state-space approach similar to the one described in the previous paragraph. A new approach to regime switching is introduced by a latent state variable that governs the estimation weights. The empirical results for six macroeconomic time series are ambiguous; the proposed approach only outperforms the individual forecasts in three out of the six cases. It does though perform better comparing it only to the other time-varying methods in most of the cases. Furthermore, Elliott & Timmermann (2005) generate six data series by Monte Carlo simulations with different characteristics to find out for which series combination methods are successful. In the presence of persistent structural breaks, regime switching methods are reported to have the best performance for bigger sample sizes.

2.3.5.2 Adaptive Intelligent Models

Two of the presented fuzzy systems include a learning mechanism: Petrovic et al. (2006) periodically adapt the rule bases of their proposed fuzzy system by calculating the confidence of an individual forecast based on their past performance. In the hypothetical experiment, the learning mechanism lowers the forecast error compared to the system with learning turned off. In Frantti & Mähönen (2001), membership functions are automatically generated by processing incoming data, and thus adapted if new data arrives. No comparison between the adaptive and a non-adaptive system is given.

2.3.6 Combining or not combining?

Although empirical studies usually favoured forecast combinations over individual approaches, the question of whether or not to combine a pool of available forecasts remains controversial till today. Supporters of the combining approach give a number of reasons for the benefits of this approach. One of the first arguments was given in Bates & Granger (1969), based on the fact that some forecasts might come from closed, unobserved or private sources, where underlying variables, assumptions and models are not available. If one was to construct one single "super-model",

the hidden knowledge will be a major obstacle. A forecast combination offers a straightforward solution for taking the forecast into account.

A second reason is the presence of structural breaks in real-world data generation processes, which can be caused by institutional change, economic crises and many other factors. These breaks are very hard to detect in real-time. Some models will adapt quickly to the change, while others need longer to adjust their parameters. A combination of methods with different adaptation capabilities and adaptation speed is likely to outperform individual models as confirmed by Pesaran & Timmermann (2007).

Another argument is related to the fact that in practice, it is usually simply impossible to be able to correctly model a data generation process in only one model, it is furthermore highly unlikely that one single model will dominate others for all points in time. Single models are most likely to be simplifications of a much more complex reality, so different models might be complementary to each other and be able to approximate the real process better. This is also connected to the concern about model risk: even if a single best model might be available, it will usually require specialist knowledge to find the right model and its parameters. Forecast combinations decreases this model-risk by diversification and can help to achieve good results without in-depth knowledge about the application and time consuming computationally complex fine-tuning of a single model. Based on the same diversification argument, Newbold & Harvey (2006) describe the intuitive appeal of forecast combination as analogous to the investment in a portfolio of securities rather than a single stock. Hibon & Evgeniou (2005) reported that combining forecasts does not always give the best accuracy, but is far less risky in terms of performance outliers than choosing just one method and provides an alternative in cases where it is not feasible or the expertise is missing to reliably identify and design a single best forecasting model.

The first publication questioning the general approach of forecast combination were the discussions following Newbold & Granger (1974). More recently, Timmermann (2006) summarises reasons for and against the combination approach. Opponents of forecast combination argue that most combination algorithms are improvised and lack a sound statistical background. Instabilities and errors in estimating the combination weights can compromise performance, which is a danger that increases with increasing complexity of the forecast combination model. Another argument states that it is preferable to collect all the information the individual forecasts are built on and construct a single super-model which will provide the optimal forecast. Huang & Lee (2007) recently discussed this question under the title "To combine forecasts or to combine information?" and concluded that the super-model approach (combining information) is superior for in-sample evaluation. Circumstances of when the forecast combination work better than information combination for out-of-sample evaluation are analytically identified, and an empirical evaluation shows general superiority of the forecast combination approach in real-time forecasting.

2.4 Chapter summary and future work

This chapter provided a literature review in the areas of revenue management, time series forecasting and forecast combination, looking at both nonlinear and linear

models with a special section on adaptivity of the combination models. The publications mentioned in this chapter highlight a number of future research directions.

Chiang et al. (2007) identify possible future research in the area of applying revenue management systems to industries that do not traditionally use them, relaxing assumptions and requirements of the original approaches. They also see potential in the integration with relatively new technologies as the internet. A very promising research topic also pursued by Lufthansa Systems is concepts of competition and alliances - for example, one could model the impact of a competing company offering a similar portfolio on the demand of a product.

Gooijer & Hyndman (2006) and Clive Granger in Ord (2001) mention multivariate forecasting models: despite theoretical advances in the last two decades, they are still not widely applied in practice or in empirical studies, and the authors expect that this will change in the future. Makridakis & Hibon (2000) advocate putting more emphasis on real-life behaviour of data when studying ways to improve forecasting accuracy. Nonlinear methods deserve more attention and need more thorough research concerning their design, parametrisation, applicability and performance compared to linear methods as agreed upon by both Gooijer & Hyndman (2006) and Teräsvirta et al. (2004). Gooijer & Hyndman (2006) furthermore expect to see more research in the areas of density forecasting and improved forecasting intervals.

One conclusion motivating the line of research for this thesis is consistent for both forecasting and forecast combinations: there is no single method consistently providing more accuracy than another looking at empirical studies. Fred Collopy in Ord (2001) states, that the times where forecasting researchers are mainly looking for a better general method are over. The article of Robert J. Hyndman in the same publication has the same notion, one of his conclusions being: "*Makridakis & Hibon (2000) show us* ***what*** *works well and what does not. Now it is time to identify* ***why*** *some methods work well and others do not.*" This is a question still extremely relevant to time series forecasting, forecasting combination and their application to airline industry and one of the central questions in this thesis.

3

Do we need experts for time series forecasting?

Forecasting practitioners are faced with a tricky question: which approach is the best for the problem they need to solve? The number of available methods and parametrisations can be confusing and there appears to be no general guideline on when to pick which. In the 50 years of time series forecasting research, the question has naturally been addressed in a number of comparative empirical studies and other publications, some of which will be discussed in the first part of this chapter.

Motivated by the results of this discussion, this chapter will proceed to present an empirical study examining a selection of off-the-shelf forecasting and forecast combination algorithms with a focus on assessing their practical relevance by drawing conclusions for non-expert users. Considering the advances in forecasting techniques, this analysis addresses the question whether we need human expertise for practical forecasting applications or whether the investigated methods provide comparable performance. Parts of the results have been published in Lemke & Gabrys (2008*a*).

3.1 Choosing a forecasting approach

Forecasting and forecast combination has been extensively researched in the last decades and a large number of empirical studies have been conducted to compare out-of-sample accuracy of various methods. Many of these studies have been very limited in terms of the methods examined and the number and nature of time series used. However, there are a few exceptions that gained considerable reputation and provided the basis of many discussions of forecasting experts. This first section discusses influential empirical studies conducted in the last decade before Section 2 looks more closely at combinations of forecasts, summarising guidelines and recommendations on their usage given in literature.

3.1.1 Empirical studies

The three so called M-competitions consist of the M-competition (Makridakis et al. (1982)), the M2-competition (Makridakis et al. (1993)) and the M3-competition Makridakis & Hibon (2000). Competitions have a few advantages in comparison to ordinary empirical studies: the conclusions are not solely based on the forecasting skills of a small number of individuals, but on many experts in the field with different research expertise who are willing to take part. Furthermore, the same data set is used to compare methods using the same methodology and accuracy measures, facilitating more objective conclusions that cannot usually be drawn from a number of independent small empirical studies. Results of the latest M-competition have been published in Makridakis & Hibon (2000). The data set consisted of 3003 mainly yearly, quarterly or monthly business and economic time series with different

numbers of available observations and required forecasting horizons. 24 competing methods grouped into the six categories naive, smoothing, decomposition, ARIMA, expert systems and neural networks have been evaluated. The competition came to four major conclusions that have also been confirmed by the previous M1 and M2 competitions:

1. Statistically complex models do not necessarily outperform less sophisticated ones.
2. Forecasting performance depends on the accuracy measure used.
3. Forecasting performance depends on the length of the forecasting horizon.
4. Combinations of forecasts do outperform the individual methods involved on average.

Conclusion number four has already been discussed in Section 2.3.6. The second and third conclusion were largely undisputedly accepted by the forecasting community. Conclusion number one however has been subject to fierce discussions in the commentaries to the competition published in Ord (2001). In the introduction, Keith Ord mentions that more complex methods like ARIMA need at least 50 observations to build a model that produces good forecasting results, which was not given for many of the series in the data set and is the reason for simple methods seeming advantageous. The same reason is given by Sandy D. Balkin as an explanation for the mediocre performance of the only neural network in the competition. In addition, Mr Balkin criticises the choice of the M3-competition data set as consisting of financial and economic time series only, and thus being too limited to come to any general conclusions. Clive W.J. Granger points out that looking at individual series, the proportion of times that a simple method will be the best is a lot smaller than for a complex method, indicating that although their average performance seems promising, simple methods are not the best choice for the majority of the series.

Looking into the M3 competition in more detail, the authors recommend the dampened trend exponential smoothing method by Taylor (2003) as a method that is very easy to implement and still gives good performance. Among the best performing methods was also the Theta-model by Assimakopoulos & Nikolopoulos (2000).

Another frequently cited extensive empirical study has been carried out by Stock & Watson (2001) and has already been mentioned in Section 2.2.4. This contribution is still being discussed and of significance in the forecasting literature. The authors compared 49 linear and nonlinear forecasting methods using a data set consisting of 215 U.S. macroeconomic series. In general, they came to a similar conclusion as Makridakis & Hibon (2000): compared to simple methods, in their case represented by an autoregressive model of order four, there are very few times where a complex method clearly and consistently performs better. The authors furthermore aimed to compare linear with nonlinear methods, where no clear-cut winner could be identified: the forecasting accuracy differed significantly across forecast horizons and series, confirming two more of the M3 competition conclusions. The re-examination of Teräsvirta et al. (2004) using similar models as Stock & Watson (2001) was based on a data set of 47 time series and put considerably more effort into defining a suitable architecture and parametrisation of the nonlinear models, however, results still did not overly favour nonlinear models and the question of which class generally performs better remains undecided.

Summarising, it can be said that the results of the enormous amount of empirical studies on time series forecasting have been mixed and sometimes contradictory. Although the three big sample studies mentioned here come to the same very general conclusions, no best overall method could be clearly identified. A rough guideline which method to choose in which situation has been given in Ord (2001). Generally, a small number of observations, very erratic process behaviour and no or weak seasonal pattern for a given time series are strong indicators that simple methods should be used. As the number of observations grows and the series exhibit a stable stochastic and a strong seasonal pattern, statistical criteria and contextual information should be used to identify an appropriate, possibly sophisticated and more complex model.

3.1.2 Evidence on using combinations of forecasts

Although there is a vast amount of literature available about linear forecast combination, no straightforward method of choosing the right approach can be determined. The relative performance of the models depends on the error variance of the individual forecasts, the correlation between forecast errors and the sample size for estimation according to de Menezes et al. (2000). Timmermann (2006) summarises a number of general outcomes that tend to be consistent in the majority of empirical studies:

- **Simple combination schemes are hard to beat.** Stock & Watson (2004) coined the term "the forecast combination puzzle", referring to the fact that simple combinations repeatedly outperform more sophisticated ones in empirical experiments, although in theory, a combination which puts more weight on a well-performing forecast should be superior. Smith & Wallis (2009) recently re-visited this issue, mainly blaming it on parameter estimation errors in the more complex combination strategies and stating that it is error-prone to assess forecast accuracy for estimating combination weights on limited past and possibly noisy data. Especially when combining a large number of forecasts, the simple average provides a trade-off between a small bias and a larger estimation variance.

- **Trimming often improves performance.** Trimming refers to discarding the individual models that performed worst in the past before combining the others and has been recommended by many authors, among which are Jose & Winkler (2008), Stock & Watson (2004) and Granger & Jeon (2004) with recommendations of trimming percentages usually ranging from 5-30%.

- **Shrinkage often improves performance.** The idea of shrinkage is to trade off weight estimation errors against biased weights and has been found beneficial by, for example, Aiolfi & Timmermann (2006) and Stock & Watson (2004). A simple and popular approach is shrinking combination weights ω_i towards equal weights, proposed by, amongst others, Stock & Watson (2004) as shown in the following equation:

$$\omega_i = \lambda \cdot \hat{\omega}_i + (1 - \lambda) \cdot \frac{1}{m}, \tag{3.1}$$

  where m is the number of methods involved, λ controls the extent of shrinkage and ω_i is the linear combination weight of model i as in the previous chapter.

Parameter λ depends on the size of the estimation window in relation to the number of models; with increasing window size, the estimated weights become more important.

- **Limited time-variation in the combination weights may be helpful.** Empirical results reported on time varying parameters are at best ambiguous, if not discouraging. Among the authors acknowledging a small positive impact of time varying parameters are Stock & Watson (2004) and Elliott & Timmermann (2005), both of which however stress that the extent of the time-variation should be limited.

Concerning linear forecast combination methods, de Menezes et al. (2000) give practical guidelines of which linear model to choose in which situation. According to these, rank-based approaches work well for small sample sizes, while variance-covariance and regression methods are more suitable for longer time series. Similar error variances of individual forecasts indicate that simple averages might be a good choice. Concerning regression, there seems to be a general disagreement on whether or not to use the unrestricted version, and on whether or not to favour regression models over variance-covariance based methods.

Compared to literature on linear forecast combination, the number of publications about nonlinear methods appears small. Empirical results are mostly smaller case studies; the majority only use one time series like, for example, He & Xu (2005), See & Openshaw (2000) and Li & Tkacz (2001), some of them with less than 300 observations for assessment of algorithms. The choice of individual forecasts used for combination is different in each study, making it difficult to compare results. Evaluations across methods have only been found in one publication by Palit & Popovic (2000) with no clear result given. The most investigated of these methods is clearly forecast combination with neural networks. While two publications with smaller empirical evaluations report a significant improvement over individual forecasting and linear forecast combination models, others find that performance is mixed dependent on forecasting horizons or the error measure used. Indications also exist that neural networks perform better than other methods if there are structural breaks in the data generation process. The mixed results and the traditional drawbacks of neural networks might favour self-organising approaches, although only one publication mentioned them in the context of forecast combination. Fuzzy systems are the method to use if subjective or linguistic forecasts have to be included in the combination. If this is not the case, the computationally complex process of generating and applying a fuzzy system can provide a reason against it. However, some publications on fuzzy systems, namely Frantti & Mähönen (2001) and Petrovic et al. (2006) also consider adaptivity to a changing environment thoroughly, which has not been done for any of the other nonlinear methods yet.

3.1.3 Conclusions

In the studies presented and summarised in the previous section, forecasting experts spent time and used accumulated knowledge designing and tuning methods with different degrees of complexity only to come to the same conclusion: no single best method that works well on all time series can be identified. Consequently, in practical applications, one would need forecasting experts to investigate specific time series and suggest a forecasting model. However, the fact that experts with sufficient

application-specific and forecasting expertise are mostly rare and expensive leads to the question of how much loss in forecast accuracy, if any, one might expect from failing to consult experts, but using off-the-shelf methods instead. The importance of this question is furthermore enforced by the well-known finding that simple methods like exponential smoothing for forecasting and simple average for combinations tend to work just as well as the more complex competing methods in practice.

3.2 Empirical study

Based on the conclusions of the previous section, an empirical study using a number of well-known and easily accessible forecasting and combination methods has been conducted. Data sets have been obtained from two recent forecasting competitions, to provide a good basis for comparison of results achieved here to results of independent researchers from diverse backgrounds. This study gives a contribution to answering the question if complex methods are of practical relevance by looking at the following issues:

- Do the more complex methods used outperform the simple models?
- Can forecast combination increase the accuracy of the simple individual predictors?
- Can the performance of expert contributions to the competitions be matched with off-the-shelf methods?

The initial presentation and justification of data set and methodology used will be followed by the analysis of the results.

3.2.1 Data sets

Data sets have been obtained from the NN3 and NN5 neural network forecasting competitions Crone (2006/2007) and Crone (2008). These competitions took place fairly recently and included 111 time series of varying lengths and forecasting horizons each, which will allow analysis of the results regarding these different aspects. Submissions had been invited for methods from computational intelligence area and performances of well-known statistical benchmarks were provided for comparison. To the best of the author's knowledge, there is no publication analysing the results, but rankings, performances and descriptions of the various contestants are given on the cited websites. The choice of the NN3/NN5 data over data from the M3 competition has another advantage: M3 data was often criticised for including series that are too short (some of the series including as few as 14 observations) with too few complex characteristics for being a suitable data set for both linear and nonlinear or simple and complex model. NN3/NN5 data seemed more promising and balanced in this respect.

Figure 3.1 shows one sample series for each of the data set. NN3 data includes monthly empirical business time series with 52 to 126 observations, while the NN5 series consist of daily time series from cash machine withdrawals with 735 observations each. The competition task was to predict the next 18 or 56 observations, respectively. While NN3 data did not need specific preprocessing, NN5 data included some missing values, which were substituted by taking the mean of the value of the corresponding weekday of the previous and the following week.

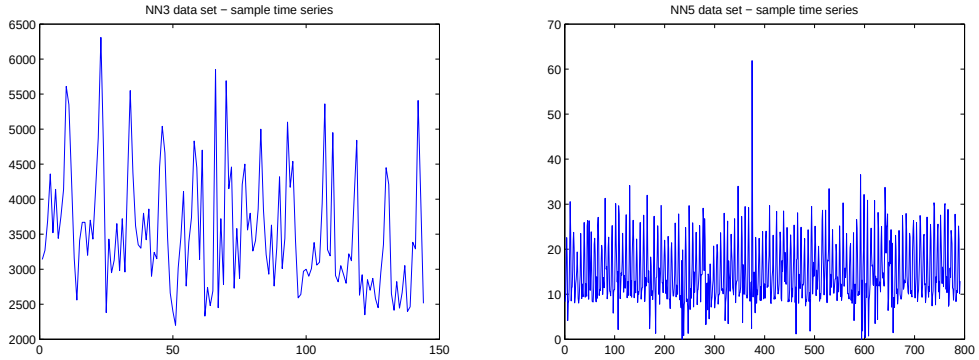

Figure 3.1: Examples of time series, left: NN3 competition, right: NN5 competition.

3.2.2 Methodology

The experiments will be conducted in a single-step-ahead and a multiple-steps-ahead version. The multi-step approach allows comparison to the results of the competitions. Many empirical studies are however based on single-step-ahead predictions, and results might vary to the multi-step approach because of the missing accumulation of errors and an increased influence of chance. In each of the cases, the last 18 or 56 values of each series were not used for training the models to enable pseudo-out-of-sample error evaluation.

Forecasts will be evaluated using two error measures of a forecast $\hat{y}$ forecasting the target variable y on a test set with n observations. The first one is the symmetric mean absolute percentage error (SMAPE), which is given by

$$\text{SMAPE} = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{(y_i + \hat{y}_i)/2} * 100, \tag{3.2}$$

and has been used for evaluating a wide variety of methods, which allows comparisons between series on different scales. As the SMAPE was the error measure used for the NN3/NN5 competitions, this was an obvious choice for this study. Anne Koehler in Ord (2001) however shows that the SMAPE is asymmetric, punishing methods giving low forecasts more than methods that produce high forecasts. Consequently, an additional error measure, the square root of the mean squared error (RMSE) given by

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3.3}$$

is also calculated. The RMSE is a widely used error measure in many empirical forecast evaluations and, as an absolute measure, depends on the scale of the time series. Because errors are squared before they are averaged, more weight is put on large errors than on small ones, which is a useful characteristic whenever large errors are particularly undesirable.

It was aimed to create a functionally diverse pool of automatic individual forecasting algorithms, representing both the basic simple and complex, linear and nonlinear models with at least one model from each of the groups presented in the

literature review Section 2.2. The forecasting and combination algorithms implemented are explained in the remainder of this section. As not all of the individual algorithms provide native multi-step-ahead forecasting, some of them are implemented using two approaches: an iterative approach, where the last prediction is fed back to the model to obtain the next forecast, or a direct approach, where n different predictors are trained for each of the 1 to n steps ahead problem. In the equations given, $\hat{y}_{t+1}$ will denote the one-step-ahead prediction with current time t and y_i the observation of the time series at time i.

3.2.2.1 Simple forecasting models

The power and popularity of simple forecasting algorithms has been discussed in Section 3.1.1. Five algorithms that can be considered "simple" have been implemented

- For the **moving average** according to, for example, Makridakis et al. (1998), the arithmetic mean of the last v observations according to Equation 3.4 is calculated. An appropriate time window is found by grid-searching v-values from 1 to 20 and choosing the v with the lowest error on a validation set prior to the test set.

$$\hat{y}_{t+1} = \frac{1}{v} \sum_{i=t-v+1}^{t} y_i \tag{3.4}$$

- **Single exponential smoothing**, which is also described in Makridakis et al. (1998), is the simplest representative of smoothing methods. According to one interpretation, it is calculated by adjusting the previous forecast by the error it produced. The parameter α controls the extent of the adjustment and is determined again by minimising pseudo-out-of-sample forecast errors.

$$\hat{y}_{t+1} = \alpha y_t + (1 - \alpha)\hat{y}_t \tag{3.5}$$

- **Taylor's exponential smoothing** is an exponential smoothing algorithm more recently introduced by Taylor (2003). Here, a multiplicative growth rate r is applied to the current level L_t of the time series to obtain the one-step-ahead forecast, t being the time index. Furthermore, Taylor uses a factor ϕ that takes values between 0 and 1 and dampens the growth rate with ascending periods ahead to be forecast by raising it to the power of the forecast horizon in each step. Equation 3.6 provides the details, with h being the number of periods ahead to be predicted. The parameters α and β are smoothing constants taking values between zero and one, which are again determined by grid search.

$$\begin{aligned} \hat{y}_{t+h} &= L_t r_t^{\sum_{i=1}^{h} \phi^i}, \\ L_t &= \alpha y_t + (1 - \alpha)(L_{t-1} r_{t-1}^{\phi}) \\ r_t &= \beta(L_t / L_{t-1}) + (1 - \beta) r_{t-1}^{\phi} \end{aligned} \tag{3.6}$$

- **Polynomial regression** fits a polynomial to the time series by regressing time series indices against time series values. In this experiment, a suitable order of the polynomial between two and six is grid-searched and the resulting curve

extrapolated into the future; Equation 3.7 shows the example of a regression of order three, where ω_i are parameters estimated using the training set and t is the current time index.

$$\hat{y}_t = \omega_0 + \omega_1 t + \omega_2 t^2 + \omega_3 t^3 \tag{3.7}$$

- The **Theta-model** according to Assimakopoulos & Nikolopoulos (2000) works on the curvature of a time series. If $\{y_1, ..., y_n\}$ again constitutes the observed univariate time series, a new series $\{z_1(\theta), ..., z_n(\theta)\}$ called theta-line is constructed by applying a coefficient θ to the second order differences of the original time series so that

$$z_t^{''}(\theta) = \theta y_t^{''}, \tag{3.8}$$

where

$$y_t^{''} = y_t - 2y_{t-1} + y_{t-2} \tag{3.9}$$

with t once more being the time index. This is a second order difference equation which, according to e.g. Box & Jenkins (1970) pp.114-119, has the solution

$$z_t(\theta) = \hat{a}_\theta + \hat{b}_\theta(t-1) + \theta y_t \tag{3.10}$$

with $\hat{a}_\theta$ and $\hat{b}_\theta$ being constants that can be determined according to Hyndman & Billah (2003) who derived their equations by finding the values $z_1(\theta)$ and $z_2(\theta) - z_1(\theta)$ that minimise the sum of squared differences between the original series and the theta-lines.

In the original setup in Assimakopoulos & Nikolopoulos (2000), forecasts of two theta-lines are calculated and averaged. The first forecast uses $\theta = 0$ which, looking at Equation 3.10, results in a linear regression problem. The forecast for a theta-line with $\theta = 2$ is obtained by single exponential smoothing on the theta-line as shown in Equation 3.5.

Hyndman & Billah (2003) remark that this model is equivalent to a special case of single exponential smoothing with drift.

3.2.2.2 Automatic Box-Jenkins Models

Box-Jenkins models are given by the notation ARIMA (p,d,q) and consist of the following three parts:

- AR(p) denotes the autoregressive part of order p. Autoregression defines a regression $y_t = \omega_0 + \omega_1 y_{t-1} + ...\omega_p y_{t-p} + \epsilon_t$ where the target variable depends on p time-lagged values of itself weighted by ω_i.
- I(d) defines the degree of differencing involved. Differencing is a method of removing non-stationarity in a time series by calculating the change between each observation. The first difference of a time series is thus given by $y'_t = y_t - y_{t-1}$.

- MA(q) indicates the order of the moving average part of the model, which is given as a moving average of the error series. It can be described as a regression against the past error values of the series $y_t = \omega_0 + \omega_1\epsilon_{t-1} + ...\omega_q\epsilon_{t-q} + \epsilon_t$.

The identification of an appropriate ARIMA model for a specific time series is not straightforward as discussed in Makridakis et al. (1998) and usually involves expert knowledge and intervention. Two automatic approaches have been implemented for this study:

- The original time series as well as two series representing its first and second differences are submitted to the automatic ARMA selection process of a MATLAB toolbox published in Delft Center for Systems and Control (2007), subsequently choosing the approach that produced the lowest pseudo-out-of-sample error. The maximum number of time lags used is an input parameter of the toolbox and has been set to two, which is sufficient in practice according to Makridakis et al. (1998). Furthermore, the same authors state, that it is almost never necessary to generate more than second-order differences of a time series, because data usually only involves nonstationarity of the first or second level.
- An alternative automatic approach for ARMA-modelling is included in the State-Space-Models Toolbox as described in Peng & Aston (2007). In this case, an appropriate order for p and q was chosen using the Akaike's Information criterion (AIC)[1], while a suitable order of differencing was determined by the log-likelihood of a model given the corresponding differenced series.

3.2.2.3 State space model

The State-Space-Models Toolbox by Peng & Aston (2007) has been used for estimation and implementation of a local level model with a dummy seasonal component of either twelve for the monthly NN3 data or seven for the weekly NN5 data. A basic local level consists of a random walk with noise as described by the following equations:

$$y_t = \mu_t + \epsilon_t \tag{3.11}$$
$$\mu_t = \mu_{t-1} + \eta_t \tag{3.12}$$

where ϵ_t and η_t are normally and independently distributed error terms. Variable μ_t represents a stochastic trend component in the more general structural time series definition, for the random walk it simply denotes last time series observations.

3.2.2.4 Neural networks

According to the widely accepted guidelines given by Zhang et al. (1998), a feed-forward neural network was designed. The authors stated that a backpropagation algorithm with momentum is successfully used by the majority of empirical studies, which is also implemented here. One hidden layer is generally considered sufficient for forecasting purposes. It is furthermore suggested to use as many hidden neurons

[1] $AIC = 2k - 2 \cdot ln(L)$, with k being the number of parameters in the model and L the maximized value of the likelihood function for the estimated model.

as input variables, with input variables being chosen with regard to the application. In the experiments presented here, the number of lagged input variables and thus the number of hidden neurons has been set to 12 for NN3 data and 14 to NN5 data, to capture the yearly or fortnightly seasonality, if present. Ten neural networks have been trained and their predictions averaged to obtain the final prediction and protect it from outliers. Since overfitting seems to be a problem according to, for example, Zhang (2007), a number of rules for early stopping were used in order to prevent overtraining on the test set and loss of generalisation performance: training will stop, if

- the maximum number of training epochs (500) is reached,
- performance is minimized to the goal of 0.1,
- the performance gradient falls below 10^{-10} or
- since its last decrease, validation performance error has increased more than 5 times.

Mainly, the parameters of the model have been left at their default values from the Matlab neural network toolbox, with a few exceptions: the maximum number of training epochs was raised from ten to 500 due to a small number of observations in some of the time series. However, it is assumed that the other early stopping mechanisms prevent the network to always be trained for this maximum number. The performance goal was increased from 0 to 0.1 to allow for early stopping even if performance is not perfect. The value for the momentum and the learning rate of the training algorithm has been kept at default values 0.9 and 0.1, respectively, as preliminary experiments provided no reason to change them. Sigmoid activation functions for the neurons have been used, along with a linear function for the output layer.

Furthermore, a recurrent neural network of the Elman-type has been employed. There seem to be no general guidelines in literature about the architecture of a recurrent network for time series forecasting, but one thing seems to be a common agreement: they need more hidden nodes than the feedforward neural network because the temporal relationship has to be modelled as well. In these experiments, the number of hidden nodes was set to 24, thus doubling the amount of hidden nodes for the feedforward network, otherwise, parameter values remain the same.

3.2.2.5 Forecast combinations

Concerning the choice of forecast combination approaches, only linear models are considered here for reasons given in Section 2.3. Considering all groups of linear combination approaches presented in the previous chapter, the following methods have been implemented:

- Using **simple average**, all available forecasts are averaged according to Equation 2.10.
- The **simple average with trimming** averages individual forecasts as well, but without taking the worst performing 20% of the models into account. This is in accordance with guidelines given in Jose & Winkler (2008), where 10-30% of trimming are recommended.

- In the **variance-based model**, weights for a linear combination of forecasts are determined using past forecasting performance according to Equation 2.14.
- The **outperformance method** is implemented using Equation 2.11 and determines weights based on the number of times a method performed best in the past.
- Two **regression** approaches following Equation 2.16 were implemented, one version with convex weights, meaning the weights are non-negative and sum to one, the other with unrestricted weights. Both approaches include an intercept to compensate for biases in the individual forecasts.
- The idea of **variance-based pooling** was introduced by Aiolfi & Timmermann (2006) and described in Section 2.3.2. In the experiments here, past performance is used to group forecasts into two or three clusters by a k-means algorithm. Forecasts of the historically best performing cluster are then averaged to obtain a final forecast.

For the multi-step-ahead forecast, combination weights are determined using a validation period prior to the test set wherever applicable, with the size of the forecasting horizon. In the single-step case, a rolling validation window of the same size is applied.

3.2.3 Results (single-step-ahead)

A general difference in the performances of the single-step-ahead predictions can be observed when comparing results for the NN3 and NN5 data set in Table 3.1: the very simple methods (moving average, smoothing approaches and the Theta model) perform extremely well for the NN3 data set, while the ARIMA models, the neural networks and the structural model work much better on the NN5 data set containing longer and less erratic time series. The worst performing individual method is the regression approach, suffering from a number of outliers in both data sets.

A look at the combinations of forecasts shows that the convex regression and the variance-based pooling approaches outperform the best individual predictors consistently on the SMAPE error measure, and in most of the cases also for the RMSE. Comparing combinations methods in general, it becomes obvious that the unrestricted regression suffers from parameter instabilities, especially on the NN3 data set. The outperformance model performs very well only on the NN5 data set, where relative forecast performance seems to be of lesser importance than in the NN3 data set, where it is the second worst performing combination. Trimming improves the performance of the simple average, but both average approaches are outperformed by other models giving different weights to methods according to their past performance.

3.2.4 Results (multi-step-ahead)

As only a few of the algorithms provide native multi-step-ahead forecasting (the ARIMA models, the state space model and polynomial regression), some of them are implemented using two approaches: an iterative approach, where the last prediction is fed back to the model to obtain the next forecast, or a direct approach, where n different predictors are trained for each of the 1 to n steps ahead problem. The selection of models used in this work is presented in the next paragraphs.

Individual methods	**SMAPE**	σ**(SMAPE)**	**RMSE**	σ**(RMSE)**
1. Last observation	17.4	14.1	1	0
2. Moving average	**14.7**	12.3	**0.86**	0.13
3. Single exponential smoothing	**14.7**	12.0	0.87	0.12
4. Taylor smoothing	**14.7**	12.9	**0.86**	0.12
5. Polynomial regression	22.8	19.0	2.05	4.39
6. Theta-model	**14.7**	12.1	**0.86**	0.12
7. ARIMA v1	16.0	12.5	0.94	0.24
8. ARIMA v2	15.0	12.4	0.87	0.16
9. State space structural model	16.6	21.6	0.89	0.34
10. Feedforward neural network	15.3	12.0	1.01	0.95
11. Elman neural network	15.8	12.6	1.05	1.09
Combinations	**SMAPE**	σ**(SMAPE)**	**RMSE**	σ**(RMSE)**
1. Simple average	14.2	11.6	0.87	0.40
2. Simple average with trimming	13.7	11.4	0.80	0.12
3. Variance-based	**13.2**	10.8	**0.78**	0.15
4. Outperformance	15.5	11.7	1.86	5.85
5. Regression (unrestricted)	31.3	64.6	1.38	0.70
6. Regression (convex weights)	**13.2**	11.5	**0.78**	0.20
7. Variance-based pooling 2 clusters	13.4	11.0	0.79	0.16
8. Variance-based pooling 3 clusters	13.4	11.3	**0.78**	0.18

Individual methods	**SMAPE**	σ**(SMAPE)**	**RMSE**	σ**(RMSE)**
1. Last observation	37.4	8.9	1	0
2. Moving average	34.6	7.7	0.72	0.13
3. Single exponential smoothing	35.8	8.1	0.78	0.15
4. Taylor smoothing	36.4	7.7	0.91	0.12
5. Polynomial regression	39.5	25.1	0.91	0.61
6. Theta-model	35.8	8.1	0.79	0.15
7. ARIMA v1	31.1	7.4	0.61	0.08
8. ARIMA v2	30.9	7.5	0.60	0.08
9. State space structural model	**28.2**	12.1	**0.39**	0.16
10. Feedforward neural network	32.3	7.6	0.69	0.24
11. Elman neural network	32.3	7.1	0.69	0.24
Combinations	**SMAPE**	σ**(SMAPE)**	**RMSE**	σ**(RMSE)**
1. Simple average	30.3	6.9	0.57	0.09
2. Simple average with trimming	29.7	6.8	0.54	0.10
3. Variance-based	28.3	6.6	0.47	0.11
4. Outperformance	25.8	6.2	0.39	0.11
5. Regression (unrestricted)	27.9	9.3	0.40	0.17
6. Regression (convex weights)	**24.0**	6.2	**0.33**	0.13
7. Variance-based pooling 2 clusters	26.9	7.0	0.43	0.13
8. Variance-based pooling 3 clusters	26.5	8.3	0.38	0.14

Table 3.1: Forecast performances and standard deviation on NN3 data (top) and NN5 data (bottom), single-step-ahead

Tables 3.2 and 3.3 show the error values and standard deviations for the different models. There is one clearly best performing methods for the NN5 data set: the state space structural model. For the NN3 data set, the ranking is less defined, a few methods like the structural model, the iterated neural networks and the direct moving average perform similarly well, with small differences across the two error measures. The ARIMA models do not provide convincing performance, suffering from performance outliers from a few series. This is not surprising regarding the NN3 data set, since most of the series are too short to properly fit a model, but unexpected for NN5 data. Most of the simple smoothing and averaging methods perform reasonably well for NN3 data, but considerably worse than the other models on NN5 data, which can be attributed to the longer forecast horizon, where errors can accumulate quickly.

Individual methods	**SMAPE**	**σ(SMAPE)**	**RMSE**	**σ(RMSE)**
1. Iterated moving average	19.2	18.8	0.92	0.28
2. Iterated single exponential smoothing	19.0	15.3	0.95	0.43
3. Iterated Taylor smoothing	23.3	20.5	1.17	0.86
4. Polynomial regression	27.1	23.1	1.79	2.31
5. Iterated Theta	20.6	16.7	1.01	0.64
6. Direct Theta	20.3	16.0	1.04	0.55
7. ARIMA v1	20.9	18.6	1.05	0.70
8. ARIMA v2	28.5	78.6	1.05	0.45
9. State Space structural model	18.0	17.7	**0.86**	0.34
10. Iterated feedforward neural network	**17.0**	12.6	0.93	0.56
11. Iterated elman network	17.3	13.3	0.93	0.56
12. Direct moving average	17.2	12.9	0.88	0.28
13. Direct single exponential smoothing	19.0	14.6	0.95	0.25
14. Direct Taylor	18.0	14.4	0.88	0.22
15. Direct feedforward neural network	18.0	14.9	0.96	0.54
Combinations	**SMAPE**	**σ(SMAPE)**	**RMSE**	**σ(RMSE)**
1. Simple average	17.4	13.8	0.87	0.31
2. Simple average with trimming	17.3	13.5	0.88	0.29
3. Variance-based	16.5	12.3	**0.86**	0.36
4. Outperformance	17.8	13.2	0.99	0.75
5. Regression (unrestricted)	315	397	2546	18485
6. Regression (convex weights)	18.9	17.0	1.01	0.77
7. Variance-based pooling 2 clusters	**16.4**	12.0	**0.86**	0.31
8. Variance-based pooling 3 clusters	16.7	12.2	0.97	1.07

Table 3.2: Forecast performances and standard deviation on NN3 data (SMAPE), multi-step-ahead

For the multi-step-ahead case, the unrestricted regression combination is completely unsuitable, producing large weights that fail to generalise the behaviour of the forecasts. The restricted regression produces much better results. The best combination approaches on NN3 data are the variance-based ones, both in the pooling and the conventional version. On NN5 data, variance-based pooling and the outperformance model perform best. Again, some combinations are able to outperform even the best individual predictors, although this is not as often the case as for the single-step-ahead problem.

Individual methods	**SMAPE**	**σ(SMAPE)**	**RMSE**	**σ(RMSE)**
1. Iterated moving average	35.8	8.4	0.92	0.14
2. Iterated single exponential smoothing	35.3	7.6	0.95	0.14
3. Iterated Taylor smoothing	41.1	12.0	1.17	0.14
4. Polynomial regression	49.4	108	1.79	0.24
5. Iterated Theta	35.4	7.8	1.01	0.13
6. Direct Theta	34.4	7.6	1.04	0.13
7. ARIMA v1	36.8	8.3	1.05	0.15
8. ARIMA v2	40.3	10.4	1.05	0.18
9. State space structural model	**26.5**	13.1	**0.86**	0.16
10. Iterated feedforward neural network	34.2	6.9	0.93	0.15
11. Iterated elman network	34.6	7.2	0.93	0.14
12. Direct moving average	33.4	7.3	0.88	0.14
13. Direct single exponential smoothing	34.3	7.1	0.95	0.14
14. Direct Taylor	33.5	7.6	0.88	0.14
15. Direct feedforward neural network	29.1	6.3	0.96	0.15
Combinations	**SMAPE**	**σ(SMAPE)**	**RMSE**	**σ(RMSE)**
1. Simple average	32.2	7.3	0.73	0.13
2. Simple average with trimming	31.8	7.2	0.72	0.13
3. Variance-based	29.5	6.9	0.78	0.13
4. Outperformance	26.5	6.8	0.61	0.14
5. Regression (unrestricted)	299	555	6545	22053
6. Regression (convex weights)	27.5	12.3	0.64	0.25
7. Variance-based pooling 2 clusters	28.4	8.6	0.65	0.16
8. Variance-based pooling 3 clusters	**25.9**	9.9	**0.60**	0.17

Table 3.3: Forecast performances and standard deviation on NN5 data, multi-step-ahead

A look at the histograms of the best performing individual methods in Figure 3.2 show additional interesting aspects: While the number of times an individual method performed best for a series is quite well spread across the methods on the NN3 data set, it is almost only the structural model and the direct neural network that perform best for NN5. Even the very simple methods have a number of series for which they perform best within the NN3 data set, which can be either a matter of luck, a matter of robustness due to a small number of available observations or, as mentioned above, a matter of the differing forecasting horizons.

The combination histograms in Figure 3.3 reveal another fact: although the regression combination with convex weights is outperformed by two or more methods regarding the average SMAPE, it performs best or second best in terms of the number of series for which it produces the most accurate forecasts.

3.2.5 Outcomes

The following general statements can be made:

- The conclusions two and three of the M3 competition have been revalidated - forecasting performances differed across error measures and for different forecasting horizons in this study as well.

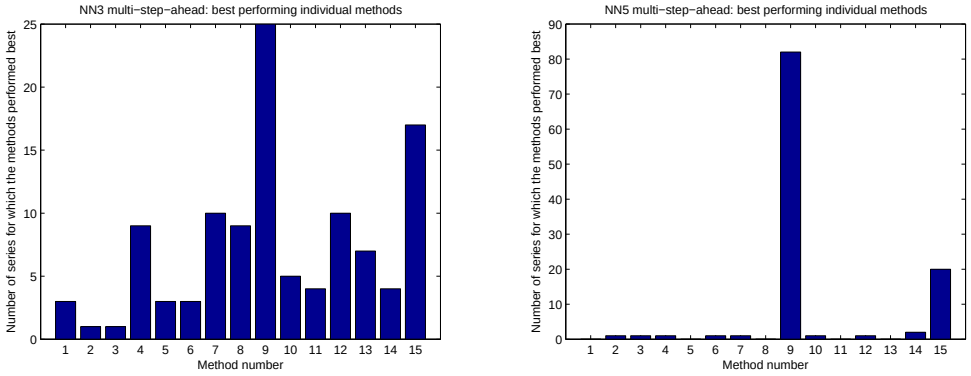

Figure 3.2: Histogram showing number of series for which a method performed best, left: NN3 competition, right: NN5 competition.

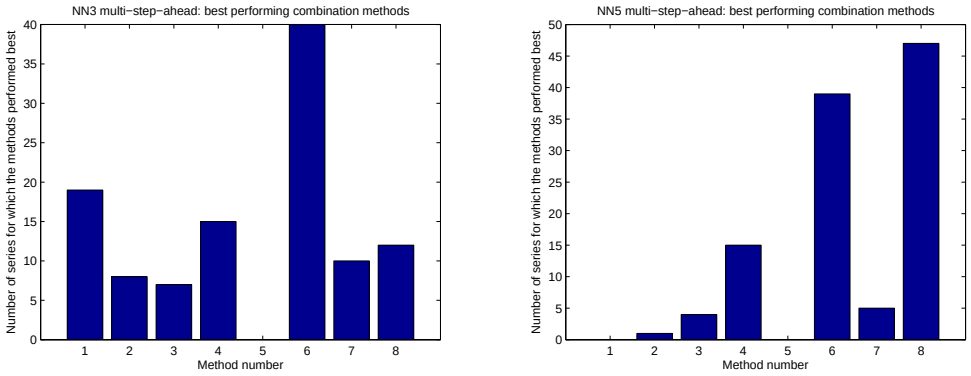

Figure 3.3: Histogram showing number of series for which a combination method performed best (SMAPE), left: NN3 competition, right: NN5 competition.

- The automatic approaches to ARIMA modelling, representing the statistically most complex individual predictors in this study, did not provide convincing performance on any of the data sets.
- Very simple methods only performed very well on the NN3 data set for single-step-ahead predictions. With longer series to forecast and a bigger forecasting horizon, their performance deteriorated.
- For individual models, the state space model performs best in the studies, dominating the majority of the given error values. As the basic algorithm corresponds to an exponential smoothing method, it shows that the state space framework with its defined parameter and model estimation algorithms is very beneficial even for simple methods.
- Combinations have proven their potential to outperform individual predictors. However, they do so less consistently on longer time series and the multi-step-ahead case.

There were 25 competitors in the NN3 competition and 19 in NN5, with SMAPEs ranging from 15.2 to 27.5 and 20.4 to 53.5, respectively. Taking the results of the

variance-based pooling with three clusters from this study, which performed well on average, participation in the competitions would have obtained rank 6 for NN3, and rank 12 for NN5. This illustrates how the relatively simple, out-of-the box automatic methods used here perform comparatively well on the data set with the shorter, more erratic and diverse time series of the NN3 data set, but can also still compete with the submissions to the NN5 competition.

3.3 Chapter summary

This chapter empirically investigated a diverse set of off-the-shelf approaches for forecasting time series. Answering the questions asked in the beginning of the section, it can be said that the results of previous studies have generally been confirmed: complex methods do not necessarily outperform simple ones and performance of methods differs according to error measures and forecasting horizons used. Especially in the single-step-ahead case for the short and erratic time series of the NN3 data, extremely simple methods such as single exponential smoothing and moving average performed very well.

Regarding combination methods, the superiority of the simple average combination approach could not be confirmed in this study. Combinations giving weights according to past performance produce better forecasts in this study, which can probably be attributed to the relatively big pool of individual predictors, where chances of some negative performance outliers are high. Some of the combinations were able to outperform even the best individual predictor.

Comparing results achieved in this study to the performance of the competition submissions, the off-the-shelf algorithms seem to provide a forecast accuracy that compares well with the performance of methods the contestants used in the NN3 and NN5 forecasting competitions, with some combinations increasing performance even further. This study hence illustrates the point that automatic off-the-shelf forecasting methods can compete well with contributions of experts in practical applications. In the next chapter, similar experiments will be conducted for the airline application. Although the setup and choice of individual predictors is quite different, results regarding forecast combinations can be compared between the two chapters. Investigating if some of the outcomes of this chapter are applicable in this particular industrial application.

According to the no-free-lunch theorem of Wolpert (1996), there is no guarantee that any method, however complex it may be, performs better for a bigger number of series than another method, which renders results of traditional empirical studies useless regarding general applicability of the outcomes, as they will only be valid for the specific data set used. So how can it at all be possible to reliably choose a well-performing method or combination? One approach is to identify and describe groups of time series within the set of all time series, generating domain knowledge and finding situations in which one method consistently performs better than others. Chapter 5 will investigate the automatic generation of such knowledge in a meta-learning context.

4

Forecast combination for airline data

The question asked in the title of the previous chapter can be answered with "yes" when it comes to the airline application investigated in this thesis, because the data does no longer only consist of univariate time series with very limited additional information, but comes with multiple dimensions, several known components that have to be modelled and strong restrictions regarding the choice of forecasting method imposed by the structure of the data and requirements regarding computational complexity. A number of applicable individual forecasting methods has been introduced in Section 2.1.4. This chapter provides an evaluation of their performance, giving details on the data set and the methodology used throughout this thesis when investigating airline data. The connection between the cancellation and the net booking forecast is examined, which will be an aspect investigated in the discussions in Chapter 6. Consistent with the previous chapter, a number of linear combination approaches is evaluated to provide first insights into the benefit of combinations for this specific application.

4.1 Data set and methodology

Evaluation of the methods for airline cancellation forecasting will take place on a data set provided by Lufthansa Systems. For this purpose, the research software "Avanti" described in Riedel (2007) was extended to allow for cancellation forecasting, which was previously not considered. Figure 4.1 shows an overview of the steps and components involved together with their inputs and outputs. A thorough description of the software modifications and components additionally developed is provided in the Appendix A.

The airline data set includes 155 weeks of booking and cancellation data and consists of

- 5 European Origin and Destination pairs (O&D),
- 1 intercontinental O&D from Asia to America,
- 1 intercontinental O&D from Europe to Asia.

Each of the origin and destination pairs includes up to three origin and destination opportunities, which represent actual scheduled flights going from the origin to the destination. The whole data set consists of 14 of these origin-destination opportunities, all selected with the objective of obtaining a data set that represents the real composition of flights as well as possible.

Data is available for 20 fareclasses, the seven days of the week and three points of sale ("country of origin", "country of destination", "rest of the world"). The

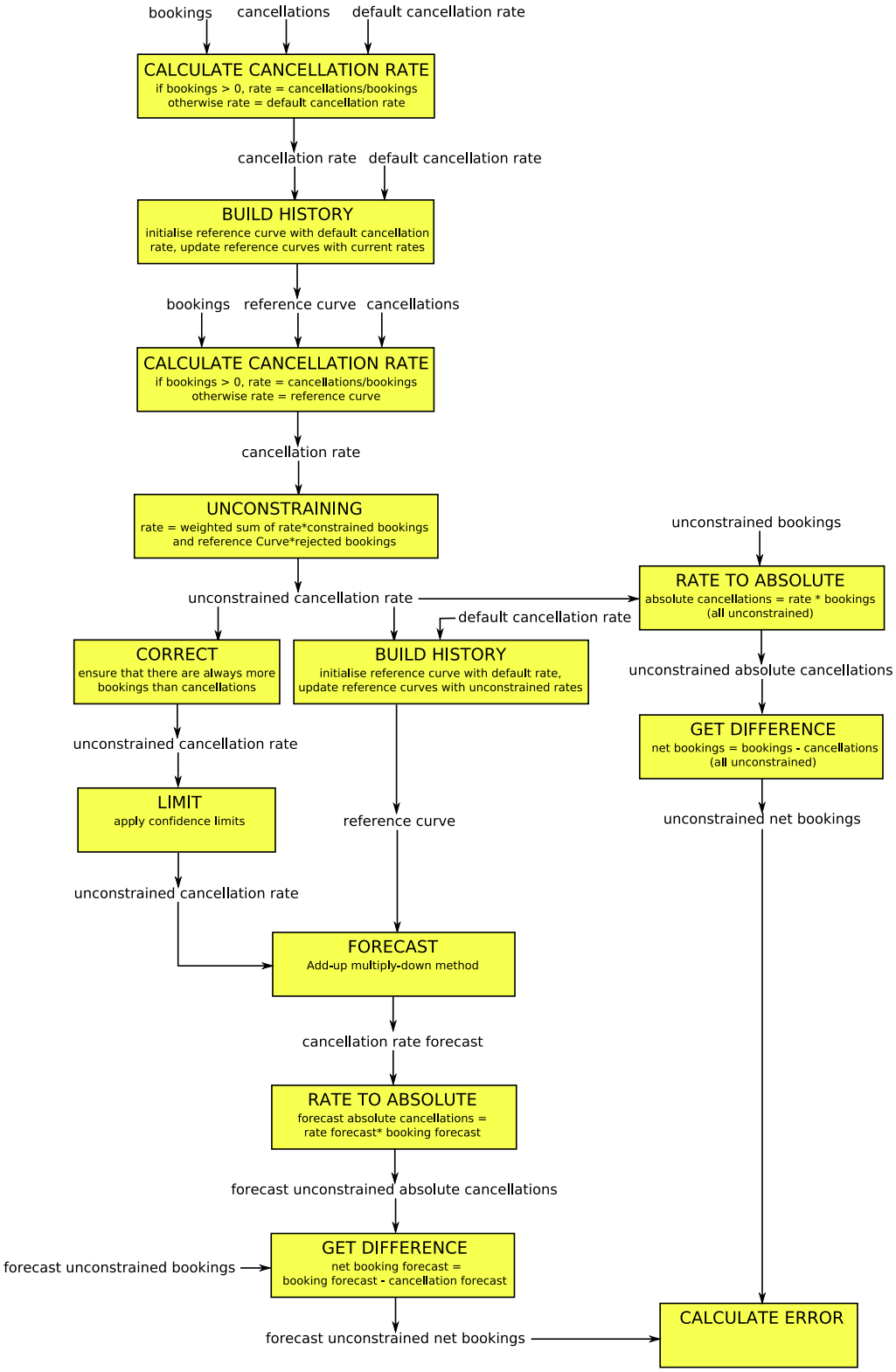

Figure 4.1: Steps and components of the cancellation forecasting procedure.

data has been cleared from extraordinary influences like long term changes of the base component or additional demand occurring due to special events, which would normally be forecasted with other approaches.

The first 52 weeks of the available data was used as an initial period for obtaining stable reference curves. This was decided because of the need for at least a year's data to calculate seasonal factors for the modified booking forecast presented in previous work, which will be combined with the new cancellation forecasts later in this thesis in order to assess overall performance gain. Forecasts are then generated for the weeks 53 to 155, with the last 31 weeks (20% of the available data) being used for an out-of-sample error estimation. The error measure used is the mean absolute deviation MAD.

Results are given for each DCP and for two different aggregation levels: the low level, which is the finest possible level on which the forecasts are normally generated, and the high level, which is an aggregation of the error values over fareclasses and points of sale that is frequently used by analysts for visualisation purposes and strategic decision making. Aggregating results on a higher level allows drawing conclusions on the bias and variance error components of the individual predictors, which will be further explored in Chapter 5.

4.2 Individual forecasting methods

Performance results for the three cancellation forecasting methods based on reference curves introduced in Section 2.1.5.2 and the probability forecast described in Section 2.1.5.3 are presented in this section, providing a baseline for the airline-application related selection and combination approaches discussed in the remainder of this thesis. Table 4.1 shows a comparison of net booking forecast performance (unconstrained bookings minus cancellations) of the exponential smoothing (exp), Brown's smoothing (brown), regression (regr) and probability forecast (prob) next to the LSB-reference-forecast (lsb) on the given data set. Performance of the reference forecast is given as mean absolute deviation, while the other methods are assessed by the percentage of improvement relative to it, meaning that positive values denote a better performance while negative indicate performance deterioration. The best method per level and DCP is printed in bold.

The first remarkable result is the similarity of the LSB and the exponential smoothing based method, differing only marginally at later decimal points. The values for these two methods outperform the other original methods for most of the DCPs. It can furthermore be noted that the newly introduced probability forecast performs well in comparison, on average outperforming all other methods for all DCPs apart from the last three DCPs at the low level.

Table 4.2 shows performance for forecasts of absolute cancellations. The new probability forecast again performs well in comparison, but not as well as for the net booking forecast, which is not surprising as the probability forecast works with net booking values and has to be converted to cancellation numbers for comparison, introducing an additional error source by dividing it by the current booking forecast.

Another issue that should be addressed is the general relationship between cancellation and net booking forecast. Intuitively, one would assume that an improved cancellation forecast leads to a better net booking forecast, but this is not always the case, in our example especially not at the later DCPs. When calculating absolute cancellations in the traditional way by multiplying the rate with the corresponding

	high level					low level				
DCP	**lsb**	**exp**	**brown**	**regr**	**prob**	**lsb**	**exp**	**brown**	**regr**	**prob**
0	21.0	-0.16	-1.88	-0.26	**7.30**	1.2	-0.04	-7.75	-1.79	**3.91**
1	19.0	-0.06	-0.12	1.70	**8.94**	1.1	-0.01	-4.19	-3.65	**1.94**
2	17.2	-0.04	-8.71	-0.39	**7.28**	1.1	-0.01	-5.86	-4.98	**1.04**
3	16.3	-0.04	-10.17	-1.31	**6.26**	1.0	-0.01	-7.04	-5.27	**1.01**
4	15.2	-0.02	-9.06	-3.21	**5.04**	0.9	-0.01	-9.55	-6.12	**1.20**
5	14.6	-0.02	-7.10	-3.30	**5.26**	0.9	-0.01	-10.85	-5.96	**1.60**
6	13.5	-0.02	-8.81	-4.72	**3.42**	0.9	-0.01	-11.64	-6.04	**1.95**
7	12.3	-0.01	-10.23	-4.23	**0.01**	0.8	-0.01	-11.93	-5.57	**1.77**
8	11.2	**-0.01**	-11.91	-3.97	-2.17	0.7	-0.01	-12.43	-5.62	**1.43**
9	10.5	**-0.01**	-13.19	-3.40	-2.47	0.7	-0.01	-12.66	-5.43	**1.88**
10	9.5	**-0.00**	-15.65	-2.43	-5.69	0.6	-0.01	-12.80	-5.55	**1.60**
11	8.4	**-0.01**	-16.05	-3.21	-8.54	0.6	-0.01	-12.53	-5.67	**1.90**
12	7.7	**0.00**	-15.79	-2.98	-8.96	0.6	-0.01	-12.28	-5.48	**1.86**
13	7.3	**-0.01**	-13.00	-2.18	-6.81	0.5	-0.01	-11.91	-5.78	**1.69**
14	6.4	**-0.02**	-14.84	-3.70	-8.65	0.5	-0.01	-13.09	-5.97	**1.16**
15	5.9	**-0.01**	-15.24	-3.04	-8.75	0.4	-0.01	-13.78	-6.04	**0.72**
16	5.1	**-0.01**	-10.92	-7.55	-9.28	0.3	**-0.01**	-10.17	-11.26	-1.14
17	4.7	**-0.00**	-10.39	-7.98	-9.73	0.3	**-0.01**	-10.51	-11.47	-1.60
18	4.4	**0.00**	-9.17	-8.57	-8.56	0.3	**-0.01**	-10.97	-11.66	-2.23
19	3.9	**0.00**	-7.57	-8.32	-7.70	0.2	**-0.00**	-11.67	-11.94	-3.12
20	3.3	**-0.01**	-7.66	-8.47	-7.24	0.2	**-0.00**	-13.56	-12.94	-4.83
21	2.2	**-0.00**	-8.33	-7.70	-10.21	0.1	**-0.01**	-16.16	-14.36	-9.06
avg	-	**-0.02**	-10.26	-4.06	-2.78	-	-0.01	-11.06	-7.21	**0.21**

Table 4.1: Mean absolute deviation of reference net booking forecast and percentage of relative improvement of four individual forecasting algorithms for each DCP. Left: high aggregation level, right: low aggregation level.

bookings, the resulting absolute cancellations will generally be strongly correlated with the bookings, as they are a factor in the equation. When absolute cancellations are subtracted from the bookings to obtain net booking values, errors hence always compensate each other to a certain extent. This effect is weaker, if existing at all, when directly working on net bookings as done in the probability forecast. However, if there is a considerable booking error, an accurate cancellation forecast will not lead to a better net booking performance than a cancellation forecast with an error positively correlated with the booking error which would lead to a compensation effect.

Figure 4.2 illustrates this: forecast errors are plotted on the finest possible level for a fareclass/point of sale/day of week configuration in a calendar week with zero bookings. Naturally, the booking forecast overestimates the bookings, indicated by a positive booking error that is drawing closer to zero with ascending DCPs, the cancellation error does the same. If the cancellation forecast is quite accurate as in the right part of the figure, the overestimation of the booking forecast is not compensated as well as if the cancellation forecast is less accurate with higher overestimation. This relationship will be further investigated in Section 6.3.1.

	high level					low level				
DCP	**lsb**	**exp**	**brown**	**regr**	**prob**	**lsb**	**exp**	**brown**	**regr**	**prob**
0	30.3	-0.03	-3.71	-9.76	-4.36	1.47	**-0.01**	-6.87	-6.68	-2.94
1	28.0	**0.01**	-7.18	-2.83	-4.70	1.43	**0.00**	-10.26	-1.16	-2.02
2	26.3	-0.00	-3.46	-0.64	-4.10	1.38	**-0.00**	-9.31	**0.26**	-1.73
3	25.1	**0.00**	-1.77	-0.18	-3.91	1.34	-0.00	-8.57	**0.13**	-1.79
4	23.8	-0.00	-1.98	**0.34**	-3.44	1.27	-0.00	-6.88	**0.04**	-2.15
5	22.7	0.00	-2.80	**0.14**	-3.53	1.20	-0.00	-5.07	**0.03**	-2.19
6	21.2	**-0.00**	-2.11	-0.16	-2.57	1.13	**-0.00**	-4.88	-0.24	-2.27
7	19.1	**-0.00**	-1.77	-0.82	-1.88	1.03	**-0.00**	-5.49	-0.73	-2.43
8	16.5	**-0.00**	-2.36	-1.32	-3.43	0.91	**-0.00**	-6.28	-1.11	-2.82
9	14.9	**-0.00**	-2.51	-1.28	-3.78	0.83	**-0.00**	-6.54	-1.49	-2.62
10	13.2	**-0.00**	-2.15	-1.67	-4.07	0.74	**-0.01**	-7.21	-1.70	-2.75
11	11.2	**-0.01**	-2.26	-1.89	-4.60	0.63	**-0.01**	-8.31	-2.59	-2.76
12	9.8	**-0.01**	-2.58	-2.48	-5.52	0.56	**-0.01**	-8.44	-3.01	-2.50
13	8.7	**-0.01**	-3.52	-3.14	-5.28	0.50	**-0.01**	-8.80	-3.45	-1.99
14	6.9	**-0.01**	-6.54	-4.10	-8.70	0.41	**-0.01**	-11.22	-4.39	-2.59
15	5.9	**-0.02**	-9.20	-4.40	-9.94	0.35	**-0.01**	-12.89	-5.09	-2.24
16	4.7	**-0.02**	-6.64	-8.65	-9.57	0.28	**-0.01**	-10.88	-10.88	-2.20
17	4.2	**-0.02**	-6.18	-9.19	-9.23	0.24	-0.01	-11.83	-12.07	-1.16
18	3.6	**-0.01**	-7.57	-9.91	-8.63	0.21	-0.01	-13.50	-13.05	**0.16**
19	2.9	**-0.02**	-8.93	-10.61	-7.53	0.16	-0.00	-15.32	-14.68	**2.76**
20	2.3	**-0.02**	-12.09	-10.56	-3.55	0.12	0.00	-19.66	-17.22	**7.38**
21	1.3	-0.03	-16.13	-9.52	**7.95**	0.06	-0.01	-26.81	-20.42	**26.55**
avg	**-**	**-0.01**	-5.16	-4.21	-4.74	-	**-0.00**	-10.23	-5.43	-0.20

Table 4.2: Mean absolute deviation of reference cancellation forecast and percentage of relative improvement of four individual forecasting algorithms for each DCP. Left: high aggregation level, right: low aggregation level.

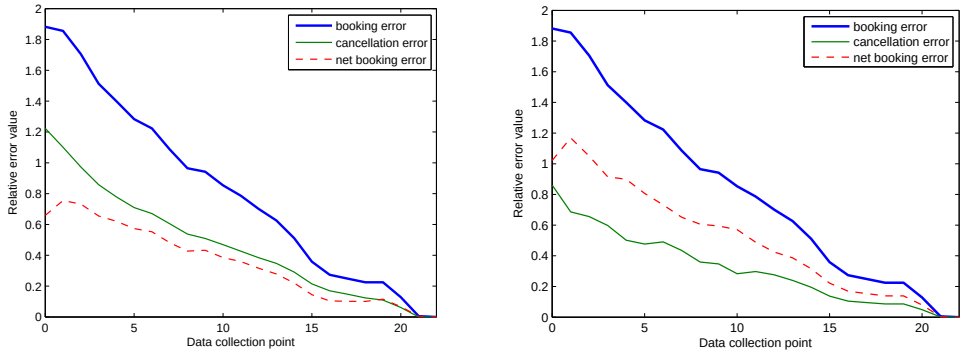

Figure 4.2: Example of the relation between booking, cancellation and net booking errors, left: a cancellation forecast with a higher error leading to a better net booking forecast because it compensates the booking error, right: a cancellation forecast with a lower error leading to worse net booking forecast because compensation does not have the same extent.

The number of net bookings are however the value of interest at Lufthansa Systems, which is why results for the following experiments will be given for net booking prediction accuracy.

4.3 Combinations

The four available individual predictors have been combined using similar approaches as used in the experiments presented in the previous chapter: the simple average (avg) averages all available forecast, the simple average with trimming (sat) does the same but trims the worst method. The outperformance model (outp), the variance based approach (var) and the optimal model (opt) have been explained in Section 2.3, as well as the regression with convex weights (regr). Results for the high and low aggregation level are given in the Tables 4.3 and 4.4.

On the low level, the combinations clearly fail to outperform the best individual predictor, while on the high level, all of the combination methods apart from the optimal model and the unrestricted regression beat the best individual forecast on average. The two methods with the bad performance are the approaches that require the most preferably stable training data for robust estimation of the parameters, which apparently cannot be guaranteed in this experimental setup. The

DCP	avg	sat	outp	var	opt	regr
0	5.48	5.40	6.17	5.50	**7.36**	6.13
1	5.47	5.68	**7.18**	5.69	5.95	6.11
2	2.11	**3.84**	3.25	2.82	1.77	2.17
3	1.55	**3.62**	2.60	2.33	0.74	1.60
4	1.41	**2.88**	2.39	2.14	-0.81	1.56
5	2.04	2.99	**3.20**	2.65	-0.38	1.83
6	1.28	**2.71**	2.44	1.89	-2.10	1.20
7	0.84	**2.06**	1.69	1.52	-2.15	0.83
8	0.68	**1.78**	1.18	1.33	-4.54	0.45
9	0.73	**1.90**	0.97	1.31	-4.26	0.77
10	-0.09	**1.18**	0.07	0.60	-4.32	-0.19
11	-0.40	**1.35**	-0.45	0.32	-6.43	-0.51
12	0.08	**1.33**	-0.19	0.70	-6.78	0.02
13	1.47	**2.22**	1.12	1.93	-6.04	1.39
14	0.27	**1.27**	-0.20	0.74	-6.99	0.09
15	0.27	**1.46**	-0.14	0.66	-6.28	0.18
16	-0.24	-0.32	-0.76	-0.03	-8.49	-0.28
17	-0.48	-0.21	-0.85	**-0.25**	-9.70	-0.54
18	-0.87	**-0.47**	-1.08	-0.67	-8.54	-0.92
19	-0.73	**-0.18**	-0.70	-0.44	-8.74	-0.85
20	-1.49	**-0.82**	-1.58	-1.00	-8.29	-1.57
21	-2.56	-2.00	-2.41	**-1.66**	-11.77	-2.63
avg	0.76	**1.71**	1.09	1.28	-4.13	0.77

Table 4.3: Flat forecast combination: percentage of relative performance improvement compared to reference forecast (lsb), high level

DCP	avg	sat	outp	var	opt	regr
0	1.39	**1.99**	1.44	1.69	0.84	1.44
1	1.28	1.37	2.27	**1.59**	-6.25	1.37
2	0.43	0.87	**1.32**	0.83	-10.87	0.32
3	0.21	0.85	**1.11**	0.66	-9.39	0.17
4	-0.32	**0.59**	0.35	0.20	-8.25	-0.45
5	-0.45	**0.33**	0.00	0.00	-6.96	-0.53
6	-0.47	**0.28**	0.04	-0.00	-7.43	-0.48
7	-0.30	**0.48**	0.18	0.18	-6.43	-0.31
8	-0.13	**0.70**	0.48	0.39	-6.25	-0.22
9	0.01	**0.98**	0.70	0.56	-5.72	-0.02
10	0.00	**1.03**	0.84	0.63	-5.39	-0.05
11	0.41	**1.49**	1.47	1.08	-5.77	0.27
12	0.58	1.46	**1.67**	1.28	-5.75	0.50
13	0.69	1.50	**1.77**	1.36	-5.88	0.59
14	0.51	1.47	**1.65**	1.27	-5.84	0.40
15	0.39	1.40	**1.48**	1.23	-6.15	0.30
16	-0.28	0.44	**0.87**	0.51	-8.60	-0.40
17	-0.63	0.19	**0.39**	0.14	-10.05	-0.73
18	-1.02	-0.24	**-0.15**	-0.23	-10.83	-1.13
19	-1.59	**-0.47**	-0.92	-0.72	-12.62	-1.69
20	-2.57	**-1.31**	-2.10	-1.50	-16.77	-2.63
21	-3.92	-2.45	-3.28	**-2.39**	-21.02	-3.99
avg	-0.26	**0.59**	0.53	0.40	-8.24	-0.33

Table 4.4: Flat forecast combination: percentage of relative performance improvement compared to reference forecast (lsb), low level

simple average with trimming performs best, with the outperformance model and the variance-based approach following closely in the next positions.

4.4 Conclusions

This chapter provides a detailed description of the methodology and steps involved in generating airline net booking forecasts. Individual methods presented in the previous chapter have been assessed on the airline data set used throughout the thesis. This includes a first evaluation of the newly introduced probability forecast, which performed very well compared to the other methods.

Forecast combinations were not able to improve on the best individual predictor on the fine level, however, improvements could be observed when forecasts were aggregated on a higher level. In contrast to the results of Chapter 3, a simple combination method with trimming clearly outperformed methods that require more sophisticated parameter estimation. This contradiction illustrates the fact that method performances will vary depending on the problem they are used for and provides another motivation for the meta-learning investigations following in the next chapter.

A few other questions can be raised when looking at the results provided in this chapter: what exactly are the benefits of including more forecasts in the method

pool? How can performance differences between the high and the low level be explained? What can be done if the choice of methods is so restricted that only few individual methods are available, reducing opportunities for combination approaches? What are the general mechanisms that make a combination method successful? These questions will be investigated in Chapters 5 and 6 of this thesis.

5 Meta-learning

What is it that determines the success or failure of a forecasting model? Chapter 3 summarised a number of studies that fail to provide consistent results as to which actual method generally performs best, which can also be concluded when comparing the empirical experiments conducted in the Chapters 3 and 4 of this thesis. The well-known no-free-lunch theorem, for example described in Wolpert (1996), provides an explanation by stating that there are no algorithms that perform better or worse than random when looking at all possible data sets or learning tasks. This implies that no assumptions on the performance of an algorithm can be made if nothing is known about the problem that it is applied to, but that there will of course be specific problems for which one algorithm performs better than another in practice. In accordance to this, this chapter investigates approaches to relax the assumption that nothing is known about a problem by extracting domain knowledge from data, linking it to well-performing methods and drawing conclusions for a similar set of time series. It identifies an extensive novel feature set describing both the time series and the pool of individual forecasting methods for the competition data sets and finds application-specific variables to characterise airline data. The applicability of different meta-learning approaches is investigated using both classic techniques and a newer ranking algorithm, first to gain knowledge on which model works best in which situation, later to improve forecasting performance.

Traditionally, experts visually inspect time series characteristics and fit models according to their judgement. The approaches investigated here are purely automatic, since a thorough time series analysis by humans is often not feasible in practical applications that process a large number of time series in very limited time.

5.1 Background

According to Vilalta & Drissi (2002), meta-learning in the broadest sense tries to answer the question of *"..how can we exploit knowledge about learning (i.e. meta-knowledge) to improve the performance of learning algorithms?"* and has mainly been investigated in a machine learning context. Meta-knowledge can have different origins as summarised in the same publication, most straightforward is the extraction of general information of the problem; for time series forecasting, this could be the series' length, its seasonality or the length of the forecasting horizon. Statistical summary measures, for example the variance or kurtosis of a time series can be used as well. Alternatively, information of individual algorithms and how they solved the problem can be considered, for example, their predicted confidence intervals or the depth of a generated decision tree. A different approach is called landmarking,

using the performance of simple algorithms to describe the problem and correlating these information with the performance of more advanced learning algorithms as described in Pfahringer et al. (2000).

In this work, meta-learning is referred to as the process of linking the characteristics of the problem described by meta-knowledge to the performance of the individual algorithms as formulated by Prudencio & Ludermir (2004*a*), with the goal of selecting the best model or providing a ranking of models for the problem under study. This provides the means of adaptation of algorithms at different levels of abstraction. One difference to the general perception of meta-learning is that the individual methods used here are not necessarily machine learning algorithms themselves, but include other approaches as well.

A classic and straightforward classification for time series has been given by Pegels (1969). Time series can thus have patterns that show different seasonal effects and trends, both of which can be additive, multiplicative or non-existent. Gardner (1985) extended this classification by including damped trends. Time series analysis in order to find an appropriate ARIMA model has been discussed since the seminal paper of Box & Jenkins (1970). Guidelines are summarised in Makridakis et al. (1998) and rely heavily on visually examining autocorrelation and partial autocorrelation values of a series.

The idea of using characteristics of univariate time series to select an appropriate forecasting model has been pursued since the 1990s. The first systems were rule based and built on a mix of judgemental and quantitative methods. Collopy & Armstrong (1992) use time series features to generate 99 rules for weighting four different models; features were obtained judgementally, by both visually inspecting the time series and using domain knowledge. Adya et al. (2000) and Adya et al. (2001) later modified this system and reduced the necessary human input, yet did not abandon manual intervention completely. Vokurka et al. (1996) extract features automatically to weight between three individual models and a combination in a rule-base that was built automatically, but required manual review of the outputs. Completely automatic systems have been proposed by Arinze et al. (1997), where a generated rule base selects between six forecasting methods. Discriminant analysis to select between three forecasting methods using 26 features is used in Shah (1997). A similar study with bigger data sets and an extended method pool is provided by Meade (2000), who uses ordinary least squares regression to map 25 descriptive statistics to a performance index.

The phrase meta-learning in the context of time series was first adopted from the general machine learning community in Prudencio & Ludermir (2004*b*), where two case studies are presented: in the first one, a C4.5 decision tree is used to link six features to the performance of two forecasting methods; in the second one, the NOEMON approach introduced by Kalousis & Theoharis (1999) is used for ranking three methods. NOEMON builds classifiers for each pair of base forecasting methods, using the six features to predict the more promising algorithm for each two-class problem. A ranking is generated using the classifiers' outputs.

The most recent and comprehensive treatment of the subject can be found in Wang et al. (2009), where time series are clustered according to nine data characteristics including measures for chaos, self-similarity and traditional statistics like trend, seasonality and kurtosis. In a first step, rules are generated judgementally by looking at performances of clusters of methods identified on a self-organising map. Furthermore, a C4.5 decision tree is automatically generated for the same

purpose. The approach is extended to determine weights for a combination of individual models based on data characteristics. Table 5.1 summarises some facts about the related work presented here for a better overview of approaches and methods used. The calculation of features and meta-learning method listed are implemented automatically if not otherwise stated.

Year	Authors	Features	Meta-learning method	Time Series	Model pool
1992	Collopy & Armstrong (1992)	18 (judgemental)	rule base (judgemental)	126 (M1)	2 exp. smoothing, random walk, linear regression
1996	Vokurka et al. (1996)	5	rule base (partly automatic)	126 (M1)	2 exp. smoothing, structural and a combination of the three
1997	Arinze et al. (1997)	6	rule base	67	2 exp. smoothing, adaptive filtering, three "hybrids" of the previous
1997	Shah (1997)	26	discriminant analysis	203 (M1)	2 exp. smoothing, structural
2000	Adya et al. (2001)	26 (mainly automatic)	rule base (judgemental)	3003 (M3)	2 exp. smoothing, random walk, linear regression
2000	Meade (2000)	26	regression	1001 (M1) + 263 + 6144 generated	3 naive, 3 exp. smoothing, 2 ARIMA
2004	Prudencio & Ludermir (2004*b*)	6 / 7	decision tree / NOEMON	99 / 645 (M3)	exp. smoothing, neural network / random walk, exp. smoothing, auto-regressive
2009	Wang et al. (2009)	9	decision tree	315	random walk, smoothing, ARIMA, neural network

Table 5.1: Time series model selection - overview of literature

5.2 Methodology for empirical studies

The remainder of this chapter investigates different meta-learning approaches, first for the competition data sets, later for the airline application. The features used to describe each data set are different and will be presented in the corresponding sections. However, there are methodological similarities of the studies that will be described here.

Figure 5.1 shows a general overview of the applied process: characteristics of time series are extracted and used as features in a classification problem, with an empirical evaluation of the performances providing the class labels. Several meta-learning approaches can then be used on the resulting data set.

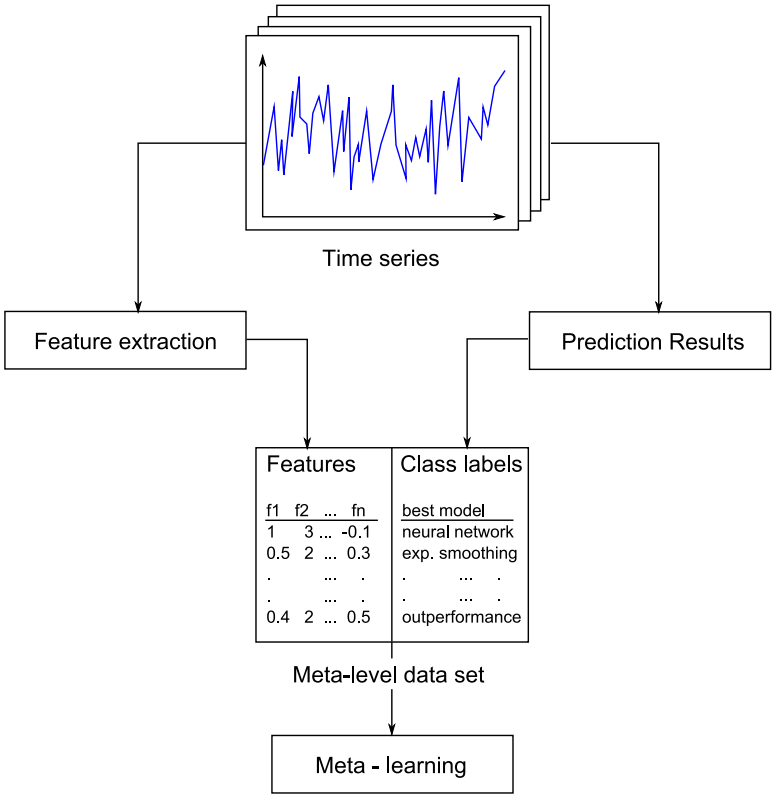

Figure 5.1: Meta-learning overview

5.2.1 Exploratory analysis

Both studies start with an exploratory analysis, providing a better understanding of the data and the potential of meta-learning for the specific problems. Because of the need for producing understandable results, decision trees are used as a machine learning method. The best performing method for each series provides its class label; features and performances are obtained for the whole data set in this experiment. The trees were built using the Matlab statistics toolbox, which implements classification and regression trees according to Breiman (1984), calculating the feature and split point at each node according to the highest resulting reduction of impurity, which denotes its degree of heterogeneity regarding the target variable. Gini's diversity index[1] (GI) is used as the impurity measure, which is a common criterion used for many problem domains, a recent discussion can be found in Sen (2005). The pruned final tree is obtained by choosing the minimum-cost-tree of a ten-fold crossvalidation.

5.2.2 Comparing meta-learning approaches

Moving away from the exploratory nature of the first experiments, a number of machine learning algorithms for meta-learning are tested in the subsequent ones, adopting the leave-one-out methodology that has, for example, been applied in Prudencio

[1] Let y be the target variable of a classification problem which can assume n different values, one specific value being denoted with index i. For a subset of all instances, Gini's diversity index is calculated using the fractions fr_i of y assuming the value i with the equation $GI = 1 - \sum_{i=1}^{n} \mathrm{fr}_i^2$.

& Ludermir (2004*b*). Of the n available series, or origin/destination opportunities for airline data, only $n-1$ are used as a training set before the remaining one is used to test the resulting meta-model. This process is repeated n times, until every series/opportunity has been the test set once. Features for the test set were now only calculated using the training set, which means that the last observations were held back from the series.

5.2.2.1 Classic machine learning

Three classic machine learning approaches have been implemented using the leave-one-out methodology, dealing with the classification problem of linking time series features to the class label of one of the most promising forecasting algorithms:

- Following the same methodology described in Section 5.2.1, a pruned **decision tree** is the first classic approach used.
- A feedforward **neural network** with one hidden layer was implemented. The number of hidden neurons was set to the number of features selected. Parameters and early stopping conditions were implemented as described in the experiments of Chapter 3.
- The third method investigated is a **support vector machine** in the least-squares version provided by Suykens et al. (2002) with a radial basis function as the kernel. Two parameters have to be set for the implementation used: the regularisation parameter γ, which trades off training error and model complexity thus providing a mechanism against overfitting, and the kernel bandwidth σ which controls the nonlinear mapping from input space to feature space. Both have been set following a grid search on the validation set.

5.2.2.2 Zoomed Ranking

Only selecting one model that is applied to a problem as in the three more traditional meta-learning approaches presented above has the obvious limitation of bearing a certain risk, even if one of the selected algorithms is a combination of predictors as in the case of our experiments. A newer approach that facilitates combinations on a higher level, the meta learning level, is presented in Brazdil et al. (2003) and applied to time series forecasting in Maforte dos Santos et al. (2004). It allows taking relations of individual performances into account by providing a ranking of methods for a particular problem. The problem space is divided using clustering on a distance measure that was calculated using the time series features. Details of this so-called zoomed ranking and our implementation of it follow.

In the first step of the zoomed ranking algorithm, divergence in the set of time series are calculated. With the normalised features f_{x,s_i}, where x is the meta-attribute number and s_i denotes the series under study, the divergence of two series s_i and s_j is given by the unweighted L_1 norm:

$$dist(s_i, s_j) = \sum_x \frac{\left| f_{x,s_i} - f_{x,s_j} \right|}{max_{k \neq i}(f_{x,s_k}) - min_{k \neq i}(f_{x,s_k})} \tag{5.1}$$

The distances are then clustered using the k-means algorithm, and the series in the cluster closest to the test series are identified for further inspection. The ranking is then generated by a variation of the Adjust Ratio of Ratios (ARR), which is applied

in a classification context in the original paper of Brazdil et al. (2003) and extended by a penalty term for time intensity in Maforte dos Santos et al. (2004). However, in this experiment, the time dimension was discarded and an absolute error measure denoted by $ERROR$ in the equation was used instead of classifier success rates to adapt the ranking to regression problems. The measure used for the competition data is the SMAPE. For the airline data, the mean absolute deviation was used to allow for comparability with previous studies. The pairwise ARR for forecasting models $\hat{y}_p$ and $\hat{y}_q$ on series s_i is

$$ARR^{s_i}_{\hat{y}_p,\hat{y}_q} = \frac{ERROR_{\hat{y}_q}}{ERROR_{\hat{y}_p}}. \tag{5.2}$$

A high ARR indicates that model p performs better than model q. To aggregate all rankings over the selected series and the pairwise ranking to one number per method, the following equation is used:

$$ARR_{\hat{y}_p} = \frac{\sum_{\hat{y}_q} \sqrt[n]{\prod_{s_i} ARR^{s_i}_{\hat{y}_p,\hat{y}_q}}}{m} \tag{5.3}$$

where m is the number of models and n the number of time series. The method with the best ranking then gets selected, or, in an alternative approach, the rankings are then used to calculate convex weights for the algorithms considered in each experiment.

5.3 Meta-learning for competition data

This section presents a number of meta-learning experiments for the NN3/NN5 competition data with the aim to provide insights into the question of which method to pick in which situations. An extensive feature set describing both the time series and the pool of individual forecasting methods will be identified in the first part of this section. These global characteristics are used as a set of descriptors to analyse time series data with different meta-learning approaches, first to gain knowledge on which model works best in which situation. Following the exploratory experiment, a different set-up is used to analyse effects of different meta-learning approaches on forecasting performance. Finally, the experimental set-up is changed to reflect the NN5 competition conditions to compare the achieved performances.

5.3.1 Time series features

Some time series features presented here are similar to the ones used in literature, but other novel and different features are introduced extending previous work published in Lemke & Gabrys (2009) and Lemke & Gabrys (2008*b*). In particular, features concerning the diversity of the ensemble of algorithms are included, which is facilitated by adding a number of popular forecast combination algorithms to the feature pool.

5.3.1.1 General statistics

Time series are often assumed to consist of different components, the most common ones being a trend component T and a seasonal component S. The series is then commonly formulated by the additive equation

$$y_t = S_t + T_t + E_t, \tag{5.4}$$

with E_t being the remaining irregular component at time index t which represents the series without seasonal or trend influences and will be referred to as the adjusted series.

For some time series features, the original time series should be the basis for calculations, for others, it is more sensible to use the adjusted version. For estimating the trend component, a piecewise polynomial curve with each of the polynomials having order three (cubic spline) with three knots as suggested in Wang et al. (2009) is fitted, providing a more flexible curve compared to a polynomial regression approach. The seasonal component is extracted using the algorithm suggested by Mohr (2005), which basically models a stochastic seasonal process with an autoregressive moving average (ARMA) component.

Metrics for trend and seasonality are then calculated as suggested in Wang et al. (2009), using the equations

$$trend = 1 - \frac{Var(E_t)}{Var(y_t - S_t)} \tag{5.5}$$

$$season = 1 - \frac{Var(E_t)}{Var(y_t - T_t)} \tag{5.6}$$

quantifying the amount of the variance that can be attributed to the trend or the season, respectively.

General descriptive statistics for a time series are standard deviation, skewness and kurtosis of the adjusted series as well as its length. Furthermore, the ratio of the standard deviation of the first and second half of the series is calculated, in order to capture changing behaviour.

Turning points and step changes are adapted from Shah (1997) describing oscillating behaviour and structural breaks in the adjusted series, respectively. A turning point for series with observations $y_i = \{y_1 ... y_t\}$ is given if y_i is a local maximum or minimum for its two closest neighbours. A step change is counted whenever

$$\left| y_i - \overline{\{y_1 ... y_{i-1}\}} \right| > 2\sigma(y_1 ... y_{i-1}), \tag{5.7}$$

where $\overline{\{y_1 ... y_{t-1}\}}$ is the mean and $\sigma(y_1 ... y_{i-1})$ the standard deviation of the series up to point $i-1$. Both measures are divided by the number of observations to ensure comparability.

Two measures of interest have been published in Gautama et al. (2004): the deterministic component of a time series measure is measured by representing a time series as a number of delay vectors of embedding dimension v, denoted by $\mathbf{y}_t = [y_{t-1} ... y_{t-v}]$. Delay vectors are grouped according to their similarity, so that the variances of the targets provides an inverse indication of predictability. Furthermore, using the iterative amplitude adjusted Fourier Transform according to Schreiber & Schmitz (1996), nonlinearity is estimated by generating 99 surrogate time series for linearised versions of the data as the realisation of the null hypothesis that the series is linear. If the delay vector representations of original and surrogate series are significantly different, the time series is considered to be nonlinear.

The largest Lyapunov exponent is a measure for the separation rate of trajectories of observations that are initially close to each other and the observations a

number of periods ahead, quantifiying chaos in a time series. The average of the Lyapunov exponents calculated using software provided in University of Goettingen (2009) was added to the feature set. All features have been listed and summarised in Table 5.2.

General statistics	
abbreviation	description
trend	Trend measure
season	Seasonality measure
length	Length of series
std	Standard deviation
skew	Skewness of series
kurt	Kurtosis of series
stdratio	Standard deviation(first half)/standard deviation(second half)
turn	Number of turning points
step	Number of step changes
pred	Predictability measure
nonlin	Nonlinearity measure
lyap	Largest Lyapunov exponent

Table 5.2: Summary of features - general statistics

5.3.1.2 Frequency domain

A number of features have been extracted from the Fast Fourier Transform of the adjusted series as summarised in Table 5.3. The frequencies at which the three maximum values of the power spectrum occur are intended to give an indication of periodicity additional to seasonality, the maximum value of the power spectrum should give a measure of the general strength of the strongest periodic component. The number of peaks in the power spectrum that have a value of at least 60% of the maximum value quantify how many strong recurring components the time series has.

Frequency domain	
abbreviation	description
ff[1-3]	Power spectrum frequencies of three biggest values
ff[4]	Power spectrum: maximal value
ff[5]	Number of peaks not lower than 60% of the maximum

Table 5.3: Summary of features - frequency domain

5.3.1.3 Autocorrelations

Autocorrelation and partial autocorrelation give indications on stationarity and seasonality of a time series; both of the measures have been included for the lags one and two, for the original and the adjusted series. Furthermore, domain knowledge on seasonality is exploited by including the autocorrelations of lag 12 for the NN3

data set which consists of monthly data and the partial autocorrelation of lag 7 for the NN5 data, which consists of weekly time series.

The Ljung-Box-Test provides a measure of randomness for autocorrelations acf of time series. It is given by

$$lb = n(n+2)\sum_{i=1}^{v}\frac{acf_i^2}{n-i} \tag{5.8}$$

where n is the length of the series and v the number of lags investigated. A higher value of the lb measure indicates a lower probability of the autocorrelations being random. All of the measures are calculated on the original and the adjusted series and are summarised in Table 5.4.

Autocorrelations	
abbreviation	description
acf[1,2]	Autocorrelations at lags one and two
acf[s]	acf[7] for NN5, acf[12] for NN3
pacf[1,2]	Partial autocorrelations at lags one and two
pacf[s]	pacf[7] for NN5, pacf[12] for NN3
acfa[1,2]	Autocorrelations of adjusted series at lags one and two
acfa[s]	acfa[7] for NN5, acfa[12] for NN3
pacfa[1,2]	Partial autocorrelations of adjusted series at lags one and two
pacfa[s]	pacfa[7] for NN5, pacfa[12] for NN3
lb	Ljung-Box test statistic

Table 5.4: Summary of features - autocorrelations

5.3.1.4 Diversity features

When dealing with combinations of forecasts, it is crucial to look at characteristics of the available individual forecasts. It is desirable to have a diverse pool of individual predictors, ideally with the strengths of one forecast compensating weaknesses of another. On another note, if there is one extremely superior forecast in the ensemble, it is unlikely that a combination with other forecasts will outperform it. All diversity features are listed in Table 5.5.

The standard ways to look at diversity for a number of methods is examining correlation coefficients. The feature pool here includes mean and standard deviation of the error correlation coefficients of the forecast pool. Other diversity measures have mainly been discussed in the context of classification tasks, for example in Kuncheva & Whitaker (2003) and Gabrys & Ruta (2005). One of the few publications dealing with diversity in a regression context is Brown, Wyatt & Tino (2005), where an error function e for training regression ensembles is introduced following the equation

$$e = \frac{1}{m}\sum_i(\hat{y}_i - y)^2 - \kappa \cdot \frac{1}{m}\sum_i(\hat{y}_i - \hat{y}^c)^2, \tag{5.9}$$

where $\hat{y}_i$ denotes the prediction of the ith of m models, y an actual observation (the target value) and $\hat{y}^c$ the combined output of the ensemble members. In the

publication, parameter κ has been added arbitrarily to control the impact of the second term of the equation.

The first term in the equation can be interpreted as the mean error of the individual methods, while the second term is the variability of the ensemble members in relation to the combined output. Brown, Wyatt & Tino (2005) relate these two components to a bias-variance-covariance error decomposition and show analytically that the first term of Equation 5.9 contains the bias and variance error terms, while the second error term contains the bias and variance error term as well, but includes the covariance of the errors of the ensemble members in addition to these. Hence, parameter κ controls the extent of the covariance impact on the error and, when used for training regression ensembles, can enforce diversity of the individual predictors. To exploit these findings for experiments with time series forecasts and the extraction of time series features, two values have been added to the feature set: the error measure in Equation 5.9 in its original form and the quotient of the mean error (first term of equation) and the variability between the ensemble members (second term). In this way, the trade off between individual accuracy and diversity can be measured.

The clustering combination method inspires a different approach on quantifying diversity. A k-means clustering algorithm is used to assign individual forecasts to one of three groups. The number of methods in the top performing cluster is then taken as a feature, that will identify if there are few or many equally well performing methods, or even just a single one. Additionally, the distance of the mean of the errors of methods in the top performing cluster to the mean of the second best is added to the feature set, in order to put the two performances into relation.

Fang (2003) state that one characteristic of a superior individual forecast is its encompassing of rival forecasts, i.e. it includes all the information other models give and dominates them. Forecasts that are encompassed by others are redundant for a forecast combination. The absence of an individual forecast that encompasses all of the others is also an indicator that combining might be beneficial. A simple test for encompassing is proposed by the regression

$$y = \beta_1 \hat{y}_1 + \beta_2 \hat{y}_2 + \epsilon$$

where y is the target variable, $\hat{y}_1$ and $\hat{y}_2$ are two forecasts and ϵ represents the error component. Successful testing for $\beta_1 = 1$ and $\beta_2 = 0$ means that forecast 2 is encompassed by forecast 1. Adopting the methodology of Kisinbay (2007), this encompassing test is conducted starting with the forecast performing best in

Diversity features	
abbreviation	description
div1	mean(SMAPEs)-mean(SMAPEs deviation from average SMAPE)
div2	mean(SMAPEs)/mean(SMAPEs deviation from average SMAPE)
div3	mean(correlation coefficients of ensemble errors)
div4	std(correlation coefficients of ensemble errors)
div5	Number of methods in top performing cluster
div6	Distance top performing cluster to second best
div7	Number of forecasts that are not encompassed by others

Table 5.5: Summary of features - diversity

the validation period, performing a pairwise comparison to the other forecasts and excluding the ones that are encompassed. The process is repeated with the next best forecast until no worse performing forecast is left in the pool. The number of remaining forecasts is another diversity characteristic.

5.3.1.5 Feature selection

Including irrelevant features in a machine learning algorithm can cause degrading performance of the resulting model according to Witten & Frank (2005). The use of redundant attributes may have the same effect. The features presented in this section were submitted to an automatic feature selection algorithm.

The method chosen is "Subset Selection", which was proposed in Hall (1998) and is implemented in the Weka collection of machine learning algorithms described in Witten & Frank (2005). It belongs to the so-called filter methods, which are known for fast and efficient selection of features in a preprocessing step, independent of a learning algorithm. The quality of a feature subset is measured by two components: the individual predictive power given by correlation values and the level of inter-correlation among them. Searching the feature space is carried out using a greedy Best First algorithm with an empty feature set as a starting point. All possible expansions are then evaluated and the best one is picked to be expanded again.

5.3.2 Exploratory analysis - decision trees

In the exploratory analysis following the description given in Section 5.2.1, using all available methods as class labels did not yield interpretable results, which is why, concentrating on the best performing approaches in the underlying experiments, the classification problem has been reduced to three different questions:

1. When is it better to use the structural time series model and when is it better to use a neural network approach? (two class labels)

2. How can one decide between pooling and outperformance model as a combination approach? (two class labels)

3. Which of the two individual and two combination methods work best for which kind of time series (four class labels)?

Results given in this section do not claim to be universally applicable, they merely provide an insight on the existence of rules for the specific data set used. However, if the meta-features describe the series well and a series with similar characteristics can be found, it is likely that the guidelines are generalisable.

In the figures given in the following results, the leaf to the left of a node represents the data that fulfils its condition, the leaf to the right hand side represents data that does not. The numbers following the methods in the leafs denote the fraction of times this particular method performed best on the data subset.

The first classification problem concerned a decision between using the structural model and the neural network, the resulting tree is shown in Figure 5.2. The top node divides the series according to their Ljung-Box statistic of the autocorrelations, stating in the right branch that the structural model works well on series where this value is higher, indicating a lower probability of the autocorrelations being random. For series with autocorrelations with higher probability of being random in the left

branch, neural networks usually beat the structural model. However, if there is a higher remaining seasonality in the adjusted series, the structural model again performs better.

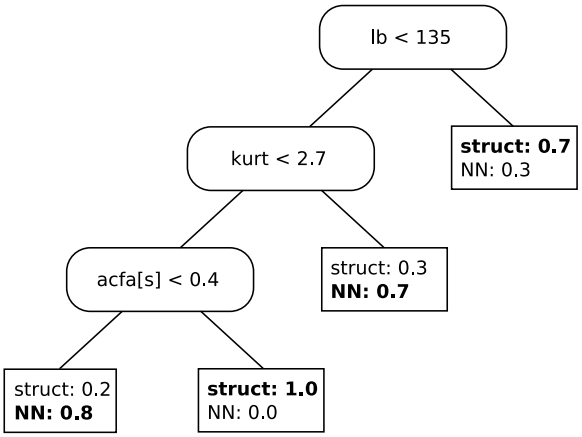

Figure 5.2: Decision tree one - which individual method?

In the second tree in Figure 5.3, two different combination approaches are compared. Variance-based pooling seems to perform better on seasonal series indicated by a higher partial autocorrelation at the seasonal lag. The left part of the tree shows an interesting outcome: if the individual accuracy is high in relation to the diversity in the pool of forecasts, an outperformance model works better; if individual accuracy however is small compared to the diversity, the variance-based pooling is the better choice. This is rather intuitive as low individual accuracy with high diversity of individual forecasts means a high risk for instabilities in the combination weights using a simple approach like the outperformance model, whereas the variance-based pooling with the strong but flexible trimming of methods may provide a higher security.

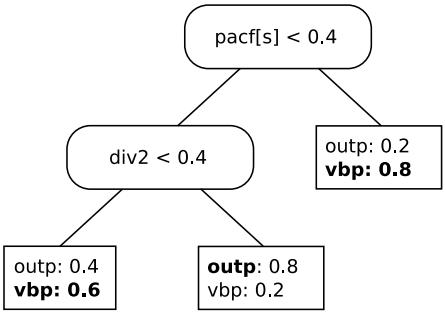

Figure 5.3: Decision tree two - which combination method?

The last decision tree in Figure 5.4 looks at how to decide between the structural model, the neural networks, the outperformance combination and the variance-based pooling approach. It shows that for the data sets used here, variance-based pooling works best for series with a low or negative autocorrelation at lag one, while the structural model is able to more accurately predict series with a high trend measure. The outperformance model works best for series with a higher autocorrelation at lag one and a lower trend measure.

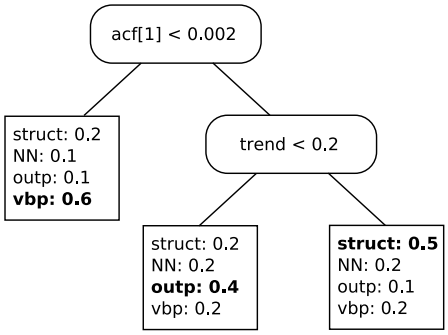

Figure 5.4: Decision tree three - which method?

In summary, this experiment has shown that rules can be generated for the NN3 and NN5 data sets, some of which have a straightforward interpretation, others of which seem fairly random. The next experiment investigates if forecast performance can be improved using decision trees and other meta-learning techniques.

5.3.3 Comparing meta-learning approaches

This experiment follows the methodology described in Section 5.2.2. For the diversity features of the test series, only the validation set forecasts were used. The validation set of the NN5 algorithm was also used for running the same basic empirical experiments described in Chapter 3, in order to help guiding decisions concerning the selection of class labels. Running the baseline experiments on the validation set of the NN3 series is impossible with some of the algorithms used here, which is due to the shortness of some of the time series included, however, it is anticipated that a preselection of methods on the NN5 data will also be beneficial for the NN3 data.

In the NN5 validation set, four methods performed best with a big performance gap to the next best methods, namely the structural model, the neural network, the restricted regression combination and the variance-based pooling, which will be used as class labels referred to as label set one. Since the restricted regression often performed well or badly together with one of the other three methods in label set one, the outperformance model was chosen to replace the regression combination in the second set of class labels. It showed reasonable performance as well, while providing a different functional combination approach and thus promising different insights into the problem. Table 5.6 summarises the two label sets.

In the first part of this section, classic machine learning approaches are investigated along with the ranking-based approach presented in Section 5.2.2.2. It has to be stated that the performances given in the table cannot directly be compared to the competition results, as the whole time series were used in the training set for building the models.

Even though all series are treated as one data set in the experiment, performances in Table 5.7 are given separately for NN3 and NN5 data. The results show that the meta-learning approach investigated here, even when using the whole time series as the training series, can only match performance of the underlying approaches in a few cases for the NN5 data, where the best method's SMAPE was 25.9, and can only slightly outperform them for the NN3 competition, where the best SMAPE was 16.3. The best overall performance is achieved by the decision tree, closely followed by the support vector machine.

Label set 1
structural model
neural network
restricted regression combination
variance-based pooling with three clusters
Label set 2
structural model
neural network
outperformance combination
variance-based pooling with three clusters

Table 5.6: Label sets for the meta-learning classification problem

	Label set 1		Label set 2	
Method	NN3	NN5	NN3	NN5
Decision tree	16.7	**25.9**	16.3	**25.9**
Neural network	17.1	29.8	16.8	29.6
Support vector machine	**16.2**	26.3	16.7	26.0

Table 5.7: SMAPE error measures applying three classic meta-learning techniques

One of the open questions using the zoomed ranking approach is determining the number of clusters to use for the k-means algorithm. However, trying different values for the number of clusters, it becomes clear that the impact on the performance is small as can be seen in Figure 5.5, so that it is safe to set the number arbitrarily, within reason.

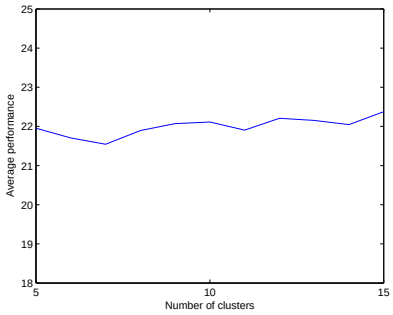

Figure 5.5: Average performance in relation to number of clusters

Performance results for the modified zoomed ranking presented in this section can be found in Table 5.8. Results are again given separately for the NN3 and the NN5 data sets, allowing better comparison with the performances given in Chapter 3. Four different ways of using the obtained rankings have been investigated, namely picking the method with the best ranking or calculating weights for a linear combination of the two (three, four) best methods.

The weighted combinations of three or four individual methods outperform all other meta-learning approaches and also improve upon the best individual predic-

tors. It can thus be seen, that some combinations outperformed model selection on a meta-learning level, which underlines the need for approaches providing a ranking of models as opposed to just recommending one of the available approaches.

	All methods		Label set 1		Label set 2	
Method	NN3	NN5	NN3	NN5	NN3	NN5
Pick best	17.0	26.1	17.8	26.5	17.0	25.7
Weighted best 2	15.8	25.9	16.2	26.1	15.8	25.5
Weighted best 3	15.6	25.4	15.5	25.3	15.8	24.0
Weighted best 4	15.7	24.6	**15.4**	**23.7**	15.5	24.2

Table 5.8: SMAPE error measures applying the meta-learning ranking algorithm

For the combination of three and four methods, using all methods as class labels provides performance that beats the best individual predictors. The best performance is obtained by using label set one, which consisted of the best performing methods on the NN5 validation set.

5.3.4 Ranking in the NN5 competition

For the previous experiments, time series observations that were not available at the time of the competition were used for training the meta-models. In this experiment, the zoomed ranking approach was evaluated on features that were calculated excluding the test data, hence the obtained forecast could have participated in the competitions. This caused problems for the NN3 data set, as the necessary validation periods for the combination approaches would reduce the observations available for individual model building to only 15 for the shortest of the series, which is too few for some of the methods to work. This experiment therefore only considers the NN5 competition, resulting performances are given in Table 5.9.

	No preselected labels	Label set 1	Label set 2
Pick best	26.6	26.7	26.5
Weighted best 2	25.6	26.1	25.3
Weighted best 3	25.4	25.3	23.9
Weighted best 4	24.5	23.8	24.1

Table 5.9: Performances applying different meta-learning techniques, competition conditions

Applying the zoomed ranking approach, the resulting out-of-sample SMAPE of 23.8 is similar to the performance in the previous experiment, showing that the approach was successful also in competition conditions and improving the 12th rank of the best individual method to rank 9 of 20 competitors. Even without pre-selection of methods for the combination, ranking-based weight calculation proved beneficial.

5.4 Meta-learning for the airline application

The benefit of meta-learning has been demonstrated on a range of different time series in the previous section. This section investigates meta-learning in the context

of the special application area of airline demand and cancellation forecasting, where, in contrast to the NN3 and NN5 competition data, extensive domain knowledge is available. Time series are no longer one-dimensional but can be aggregated on various levels, with the time series predicting booking and cancellation numbers for one specific flight being as short as 23 data points. The shortness of the time series, their special characteristics and the additional domain knowledge completely change the set of features that can be potentially relevant for characterising the data.

In the experiments presented here, global characteristics will be extracted from the data set with the purpose of exploring the impact of application-specific processes like booking control on the error values. Similar to the last experiments, the features will then be linked to the best performing method. In the second part, meta-learning will be performed on a global level, building meta-models on one set of flights and applying them to another. The third section then looks at meta-learning opportunities in each of the flights in the data set, attempting to improve forecasting performance at runtime of the forecasting system.

5.4.1 Exploratory analysis - the data

A number of features is extracted in order to gain a better understanding of the data, the influence of booking control and application-specific preprocessing as well as of the strengths and weaknesses of the forecasting algorithms. The characteristics are disaggregated on the level of origin-destination-opportunity and fareclasses and are calculated for the test set, the training set and the whole data set. Straightforward numbers extracted from the data include

- the number of data collection points (DCPs) with no values (defaults),
- the number of DCPs at which the fareclass was closed due to booking control,
- the overall number of bookings and cancellations.

In order to assess how much the cancellation reference curve has changed from the initial default reference, the difference of the most recent reference curve and the default reference for a fareclass has been taken. Both the original and the absolute values have been summed up to obtain two measures of the change, which will be referred to as the default reference change, abbreviated by deltaRef.

As fareclasses include different numbers of default values, for example if an origin and destination has no associated flights for certain days of the week, normalisation procedures need to be employed. As an illustration: if a fareclass with no defaults has been closed for half of the DCPs, the same number of closed DCPs will mean more for a fareclass with a significant portion of default values. Furthermore, if a fareclass usually only has a few cancellations, a certain change in the learnt reference curve means a higher deviation from the default reference as if the same change occurs in a fareclass with many cancellations. To address these two scenarios, the number of DCPs with non-default values is used to calculate the following measures:

- the percentage of DCPs at which the fareclass was closed due to booking control ($\frac{\text{closed DCPs}}{\text{non-defaults}}$),
- the number of average cancellations per DCP where the fareclass was open for the whole data set and the test set ($\frac{\text{cancellations}}{\text{non-defaults}}$),

- the default reference change, relative and absolute per open DCP, ($\frac{\text{deltaRef}}{\text{non-defaults}}$)
- default reference change, relative and absolute per average cancellations at open DCPs ($\frac{\text{deltaRef}}{\text{canc. per open DCP}}$)

Another issue having an influence on forecasting performance could be the difference between the training and test sets. Measures used to quantify this are

- closed DCPs (training) / closed DCPs (test)
- bookings (training) / bookings (test)
- cancellations (training) / cancellations (test)

Summary statistics of the described features are given in Table 5.10. A few interesting things can be seen here; for example does the negative mean of the relative reference curve change measure indicate that the default reference curve tends to overestimate cancellation rates on the given data. Some fareclasses do not have any bookings and cancellations in the test set, but in the training set as indicated by the minimum numbers of features 11 and 12.

For the whole available data set, correlation values were subsequently determined between measures mentioned and the sum of the net booking and cancellation errors per fareclass and origin-destination-opportunity that were obtained by evaluating the exponential smoothing forecasting method. The errors used are absolute values of the sum of relative errors, because potential systematic errors are to be investigated. Errors are given without any normalisation as well as divided by non-default values and average cancellations per open DCP. The resulting correlation coefficients are given in Table 5.11 with absolute values above 0.5 printed in bold.
High correlation values exist between the errors and the booking and cancellation numbers per nondefault DCP. This is rather intuitive, as higher booking and cancellation numbers do lead to risks of errors on a bigger scale. Normalising the error with the number of cancellations per open DCP however compensates this effect.

Feature	Mean	Max	Min	Std. Dev.	Kurtosis	Skewness
1. nondefault values	68515.48	74865	30981	11885	1.50	-1.75
2. % closed DCPs	22.74	94.53	0.63	23.37	0.53	1.26
3. $\frac{\text{Bookings}}{\text{non-defaults}}$	0.15	1.86	0.00	0.23	20.25	3.92
4. $\frac{\text{Cancellations}}{\text{non-defaults}}$	0.09	1.53	0.00	0.16	34.60	5.09
5. $\frac{\text{Cancellations}}{\text{non-defaults}}$ test set	1.69	28.79	0.00	3.20	34.75	5.14
6. $\frac{\text{deltaRef}}{\text{non-defaults}}$	-0.02	0.17	-0.16	0.06	1.22	0.64
7. $\frac{\text{deltaRef(abs)}}{\text{non-defaults}}$	0.08	0.18	0.01	0.03	0.25	0.46
8. $\frac{\text{deltaRef}}{\text{canc. per open DCP}}$	-10.08	131.58	-256.56	28.75	23.51	-2.85
9. $\frac{\text{deltaRef (abs)}}{\text{canc. per open DCP}}$	28.90	275.49	0.38	35.03	10.46	2.65
10. DCPs closed $\frac{\text{training}}{\text{test}}$	6.11	30.84	0.89	4.12	9.83	2.58
11. Bookings $\frac{\text{training}}{\text{test}}$	4.82	28.76	0.00	3.67	21.14	4.17
12. Cancellations $\frac{\text{training}}{\text{test}}$	4.71	22.35	0.00	2.73	11.46	2.78

Table 5.10: Airline data features summary statistics

	Error	$\frac{\text{Error}}{\text{non-defaults}}$	$\frac{\text{Error}}{\text{canc. per open DCP}}$
1. nondefault values	0.12	0.06	-0.08
2. % closed DCPs	0.17	0.19	0.34
3. $\frac{\text{Bookings}}{\text{non-defaults}}$	**0.65**	**0.64**	-0.14
4. $\frac{\text{Cancellations}}{\text{non-defaults}}$	**0.53**	**0.52**	-0.16
5. $\frac{\text{Cancellations}}{\text{non-defaults}}$ test set	0.45	0.43	-0.16
6. $\frac{\text{deltaRef}}{\text{non-defaults}}$	0.32	0.33	0.07
7. $\frac{\text{\|deltaRef\|}}{\text{non-defaults}}$	0.21	0.25	-0.17
8. $\frac{\text{deltaRef}}{\text{canc. per open DCP}}$	0.15	0.15	-0.48
9. $\frac{\text{\|deltaRef\|}}{\text{canc. per open DCP}}$	-0.30	-0.30	**0.56**
10. DCPs closed $\frac{\text{training}}{\text{test}}$	-0.14	-0.13	0.04
11. Bookings $\frac{\text{training}}{\text{test}}$	0.33	0.39	-0.05
12. Cancellations $\frac{\text{training}}{\text{test}}$	0.23	0.27	-0.13

Table 5.11: Correlation coefficients of airline data features and net booking errors

Another high correlation can be found for the absolute change of the reference curve and the error divided by the cancellations per open DCP. This indicates an increasing error with a higher deviation of the reference curve to the default reference. Interesting is the absence of a strong correlation between errors in general and the percentage of DCPs where a fareclass was closed. It was suspected that the unconstraining procedure described in Section 2.1.4.3 necessary to appropriately deal with the effects of closing fareclasses due to booking control will increase errors, but this is not the case.

Predicting a best performing method per origin-destination-opportunity and fareclass, the subset feature algorithm on the data set only recommends using features number 6 and 7 from table 5.11, corresponding to relative and absolute change from the default reference curve to the learnt reference curve. A decision tree was built on these features with the best performing method as a class label. The tree in Figure 5.6 suggests using the probability forecast if the absolute reference change measure is below a certain value, and Brown's exponential smoothing otherwise.

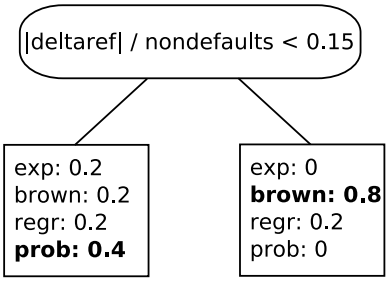

Figure 5.6: Decision tree - best method for airline net booking forecast

5.4.2 Global meta-learning

The leave-one-out methodology described in Section 5.2.2 is again employed to assess the performance of the following meta-learning approaches: decision trees (DT), neural networks (NN), support vector machines (SVM), ARR ranking method picking the best method (R-PB), ARR ranking with weighted combinations of the 2 (3,4) best models (R-T2, R-T3, R-T4). The three features capturing the relation of booking/cancellation/closed DCP numbers between training and test set had to be omitted, as they cannot be determined on the test series. A meta-model is built on 13 of the available origin-destination opportunities and then applied to the remaining one. Results for the high aggregation level can be found in Table 5.12 and for the fine level in Table 5.13.

Neural networks and the ranking approach just selecting the best method perform worst in this experiment. The decision tree and the support vector machine approach provide performances very similar to the performance delivered by the probability forecasting method, which indicates that this method gets selected as a class label in the great majority of the cases; a look at the actual class labels confirmed this suspicion. The combinations with weights obtained by the ranking algorithm perform better on average on the high level, with the ranking combination of all available methods performing best overall. It is able to outperform the reference method by 0.74% on average, which is better than the average performance of the new probability forecast, but worse than the best flat combination as presented in the previous chapter. The performance improvement is biggest for the first DCPs with improvements of up to 6.8% per DCP, which is not as good as the improve-

DCP	DT	NN	SVM	R-PB	R-T2	R-T3	R-T4
0	**7.30**	1.88	7.09	1.88	5.92	5.73	5.30
1	**8.94**	6.19	8.51	0.60	6.78	7.14	5.59
2	**7.28**	2.74	6.87	-3.28	3.84	4.14	4.42
3	**6.26**	1.37	5.92	-3.79	3.27	3.45	3.79
4	**5.04**	-1.27	4.70	-2.98	2.56	2.86	3.06
5	**5.26**	-2.33	5.05	-2.37	3.08	3.66	3.95
6	**3.42**	-4.23	3.31	-3.44	1.75	2.40	**2.62**
7	0.01	-4.03	-0.03	-4.51	0.61	**1.11**	1.03
8	-2.17	-3.67	-1.97	-5.27	0.21	**0.64**	0.50
9	-2.47	-2.95	-2.29	-6.25	0.69	**0.91**	0.53
10	-5.69	-2.16	-5.48	-6.81	-0.30	**-0.25**	-0.66
11	-8.54	-2.98	-8.14	-7.31	**-0.57**	-0.90	-1.58
12	-8.96	-2.93	-8.48	-6.38	**-0.28**	-0.76	-1.62
13	-6.81	-2.11	-6.50	-4.91	**1.16**	0.60	-0.49
14	-8.65	-3.66	-8.32	-5.78	**0.28**	-0.23	-1.44
15	-8.75	-2.81	-8.39	-5.80	**-0.17**	-0.45	-1.68
16	-9.28	-7.23	-8.96	-5.69	**-0.91**	-1.46	-2.87
17	-9.73	-7.39	-9.30	-5.43	**-1.26**	-1.93	-3.40
18	-8.56	-8.26	-8.28	-5.19	**-1.87**	-2.48	-3.58
19	-7.70	-7.70	-7.56	-4.65	**-1.68**	-2.62	-3.86
20	-7.24	-7.90	-7.24	-5.28	**-2.75**	-3.70	-4.79
21	-10.21	-7.15	-10.13	-5.95	**-4.09**	-5.56	-6.81
avg	-2.78	-3.12	-2.71	-4.48	**0.74**	0.56	-0.09

Table 5.12: Global meta-learning: percentage of relative performance improvement compared to reference forecast, high level

DCP	DT	NN	SVM	RPB	R-T2	R-T3	R-T4
0	**3.91**	-0.79	3.77	-2.67	1.87	2.08	2.07
1	1.94	-0.18	1.74	-2.97	1.64	**1.98**	0.10
2	1.04	-1.73	0.83	-3.88	0.87	**1.21**	-0.74
3	**1.01**	-2.54	0.76	-4.06	0.66	0.93	-0.75
4	**1.20**	-4.63	0.92	-4.36	0.07	0.15	-0.80
5	**1.60**	-5.31	1.30	-4.11	-0.18	-0.27	-0.49
6	**1.95**	-5.88	1.67	-4.45	-0.20	-0.24	-0.25
7	**1.77**	-5.60	1.50	-4.58	-0.18	-0.16	-0.08
8	**1.43**	-5.80	1.13	-5.14	-0.19	-0.12	-0.14
9	**1.88**	-5.49	1.57	-5.44	0.04	0.32	0.36
10	**1.60**	-5.66	1.32	-5.76	0.03	0.42	0.38
11	**1.90**	-5.68	1.57	-5.74	0.37	0.83	0.69
12	**1.86**	-5.47	1.53	-5.57	0.53	0.93	0.72
13	**1.69**	-5.71	1.35	-5.56	0.49	0.97	0.66
14	**1.16**	-5.95	0.79	-6.14	0.34	0.85	0.54
15	**0.72**	-6.08	0.36	-6.38	0.10	0.59	0.22
16	-1.14	-10.77	-1.55	-6.24	-0.98	**-0.79**	-1.86
17	-1.60	-10.96	-1.98	-6.47	-1.27	**-1.24**	-2.20
18	-2.23	-11.25	-2.62	-6.84	**-1.75**	-1.77	-2.80
19	-3.12	-11.46	-3.50	-7.38	**-2.32**	-2.54	-3.49
20	-4.83	-12.48	-5.18	-8.31	**-3.36**	-3.68	-4.84
21	-9.06	-14.27	-9.40	-11.00	**-4.97**	-5.85	-7.32
avg	**0.21**	-6.53	-0.10	-5.59	-0.38	-0.25	-0.91

Table 5.13: Global meta-learning: percentage of relative performance improvement compared to reference forecast, low level

ments achieved by the probability forecast, but more consistent over all DCPs. On the low level, improvements are less obvious, with only the decision tree producing a slight overall performance gain.

5.4.3 Local meta-learning

Global features of the data set are currently not available in the Lufthansa Systems Forecasting kernel, which is the reason why this section investigates the applicability of meta-learning on a smaller level. The main objective remains to link features of the data instances to the performance of four individual forecasting algorithms in order to build a meta-model that predicts which method will work best in a certain situation. The following features describe data on this low level and were extracted from the training and the test set on the finest possible level:

- the current data collection point
- the current number of bookings
- the current fraction of expected bookings ($\frac{\text{current bookings}}{\text{booking forecast}}$)
- the difference between the current reference curve and the currently measured cancellation rate
- the availability of the fareclass (0 for closed, 1 for open)
- the percentage of previous DCPs at which the fareclass was closed

The instances of the feature set were then labelled with the number of the individual forecasting method that performs best in the training set, which again formulates the problem as a classification task with four classes. Training features were extracted for the weeks 52 to 123 of the available 155 calendar weeks, test features were calculated on the remaining 31 weeks. For reasons of computational complexity, the support vector machine had to be removed from the pool of investigated meta-learning methods, because the time to compute results became too long. The remaining methods being compared are decision trees (DT), neural networks (NN) and the ranking algorithm (R) with results given in Tables 5.14 and 5.15. The suffix "T2" denotes trimming of the 2 methods that performed worst on the training set.

DCP	DT	DT-T2	NN	NN-T2	R	R-T2
0	6.12	**6.46**	4.03	3.12	5.50	5.62
1	6.32	**8.47**	2.14	2.61	5.78	5.89
2	3.21	**4.86**	-4.34	-1.52	1.90	1.97
3	3.01	**3.68**	-5.72	-1.52	1.74	1.81
4	2.29	**3.51**	-6.58	-2.20	1.24	1.31
5	2.66	**3.67**	-4.56	-2.28	1.81	1.81
6	1.69	**2.72**	-4.93	-3.63	0.67	0.70
7	0.74	**2.30**	-4.66	-4.29	0.29	0.29
8	-0.70	**1.85**	-5.64	-4.33	0.24	0.14
9	-0.91	**1.45**	-5.56	-4.83	0.26	0.19
10	-1.83	**1.69**	-6.13	-4.69	-0.48	-0.54
11	-2.85	**0.59**	-6.24	-5.00	-0.78	-0.81
12	-2.73	**0.34**	-5.00	-4.37	0.07	0.03
13	-1.53	0.68	-3.79	-3.55	**1.56**	1.55
14	-3.25	-1.01	-4.87	-4.60	**0.62**	0.61
15	-2.61	-0.26	-4.57	-4.63	**0.09**	0.06
16	-2.28	-0.83	-5.02	-4.50	**0.28**	0.27
17	-3.33	-1.08	-5.67	-5.67	**-0.31**	-0.35
18	-2.61	-1.29	-5.81	-5.56	**-0.69**	-0.75
19	-3.10	**-0.90**	-5.55	-4.79	-0.96	-0.96
20	-3.80	**-1.24**	-5.02	-4.70	-1.67	-1.66
21	-4.83	**-1.96**	-4.23	-4.37	-2.64	-2.64
avg	-0.47	**1.53**	-4.44	-3.42	0.66	0.66

Table 5.14: Local meta-learning: percentage of relative performance improvement compared to reference forecast, high level

For this experiment, the ranking algorithms do not provide the best performance. While slightly improving average forecasting performance on the high level, average performance on the low level is worse than for the reference forecasts. The best performing method for this experiment is decision trees with trimming, achieving overall improvements of 1.21% on the low level and 1.5% on the high level on average, with improvements at the first DCPs amounting to up to 8.5% for the second DCP.

5.5 Chapter summary

This chapter investigated meta-learning approaches on both publicly available time series competition data and the airline data set. Following a review of literature relevant to the area, empirical experiments have been conducted with the main

DCP	DT	DT-T2	NN	NN-T2	R	R-T2
0	-0.26	**1.95**	-1.62	-1.06	1.07	1.14
1	1.22	**3.33**	-2.21	-1.44	1.54	1.00
2	0.53	**2.55**	-3.74	-2.23	0.88	0.24
3	0.63	**2.13**	-4.17	-2.70	0.59	-0.02
4	-0.21	**1.66**	-5.53	-3.56	-0.04	-0.61
5	-0.76	**1.01**	-6.11	-4.22	-0.17	-0.57
6	-0.81	**0.99**	-6.22	-4.91	-0.21	-0.62
7	-0.76	**0.91**	-5.82	-4.85	-0.12	-0.47
8	-0.72	**1.10**	-6.13	-5.26	0.03	-0.44
9	-0.31	**1.34**	-6.37	-5.26	0.12	-0.16
10	-0.11	**1.29**	-6.41	-5.31	0.14	-0.19
11	0.62	**1.70**	-5.85	-4.96	0.62	0.21
12	0.72	**1.63**	-5.67	-4.81	0.66	0.39
13	0.82	**1.81**	-5.43	-4.81	0.73	0.52
14	0.82	**1.47**	-5.76	-5.04	0.64	0.31
15	0.67	**1.28**	-6.13	-5.33	0.49	0.19
16	1.29	**1.89**	-7.37	-5.33	0.47	-0.63
17	0.81	**1.35**	-7.61	-5.87	0.07	-0.94
18	0.46	**0.95**	-7.78	-6.32	-0.37	-1.37
19	-0.39	**0.18**	-8.55	-6.60	-0.99	-1.92
20	-2.00	**-0.87**	-9.78	-7.95	-2.17	-2.96
21	-3.77	**-3.08**	-12.19	-9.57	-4.58	-4.51
avg	-0.07	**1.21**	-6.20	-4.88	-0.03	-0.52

Table 5.15: Local meta-learning: percentage of relative performance improvement compared to reference forecast, low level

objective of creating domain knowledge by extracting descriptors on a training set of time series in order to train meta-models to be used on a test set.

For the competition data, an extensive number of features has been identified, some of which are novel and have not been used in this context before. Noteworthy is the description of not only the series itself, but also of the pool of available forecasting methods in order to guide decisions on whether or not a forecast combination is the best approach for a combination. For the airline data, the extracted features have been chosen differently to reflect the special characteristics of the data set and the application.

Meta-learning proved extremely valuable for the competition data when using a newer ranking approach giving weights for a linear combination of methods as opposed to only suggesting one of them. The SMAPE error values could substantially be improved with combinations of the three or four methods having the best performance on the test set. The ranking algorithm also performs best of all investigated meta-learning approaches on the airline data from a more global perspective, but does not outperform the classic decision tree when meta-learning is applied on a smaller scale. The reason for lack of success for the global airline experiment may be found in the data set: since origin-destination-opportunities were chosen to be representative of all the flights offered by the airline, the flight characteristics are necessarily quite different from each other. Since meta-learning relies on having seen a similar problem before, the diversity of the individual flights are most likely detrimental for this experiment. However, using meta-learning on a smaller scale does not provide convincing results either, which underlines a second prerequisite

for meta-learning: the data needs to be stable enough to extract meaningful meta-features, and meta-feature identification needs to be suitable and reliable. The data characteristics on the finest possible level might be too noisy and limited and the problem domain too confined to provide meta-data that can be used to obtain efficient and generalisable combination weights. A third reason for meta-learning being less successful than expected could be found in the changing environment, which probably could not be grasped with the static meta-learning approach used here. More investigations in the area of adaptive meta-learning, starting with rebuilding the model periodically, could be beneficial.

Summarising, the big potential for the meta-learning concept has been illustrated on the competition data, underlining the fact that the problem domain is an important factor in determining when a forecasting method works well and when it will fail. Furthermore, it was shown that this knowledge can be automatically extracted and lead to better forecasting performance.

Meta-learning experiments on the airline data however showed less convincing results. Reasons can be found in the nature of the data, but recalling the mediocre performances of flat combinations given in the previous chapter, the question also arises why forecast combination as a concept seems to fail in the airline application. The next chapter will be dedicated to exploring ways of further improving airline net booking forecasting performance by generating additional cancellation forecasts with diversification procedures, examining their impact on the composition of the forecast error and investigating alternative forecast combination approaches.

6

Diversification strategies for the airline application

In the previous chapter, meta-learning proved very successful when used on competition data, where domain knowledge was limited and time series were diverse. However, performances on the airline data set were not convincing; possible reasons were discussed. The next logical step is looking at alternative ways to improve forecast accuracy for the airline application, possibly exploiting the abundance of domain knowledge available and the special characteristics of the data set. Riedel (2007) provides an extensive treatment of forecasting strategies for the airline demand forecast, whereas this work tries to improve the net booking forecast by changing forecasts of its cancellation component.

Comparing the experiments for the NN3/NN5 competitions with experiments on the airline data, a major difference seems to be the number of individual forecasts considered: the competition application had over three times more individual forecasts at its disposal. This chapter looks at means of generating more individual cancellation forecasts by diversification procedures in a manner that adds value for a combination. In the process, additional insights into the dynamics of forecast combinations are sought, trying to identify beneficial characteristics of individual forecasts in general and for this specific application.

The next section will provide the necessary background, explaining different error decompositions for ensembles and combinations and their implications. Diversification methods are then discussed in Section 6.2, followed by a look at effects of airline-specific calculations on the error components identified. The last part of this chapter presents empirical experiments studying the effects of the diversifications and investigating alternative combination methods including novel forecast combination structures.

6.1 Background and motivation

Multiple classifier and prediction systems in machine learning are often referred to as ensembles and consist of a number of models sharing the same functional approach as opposed to combinations, where individual methods are usually built using different prediction models. Ensemble learning is a term describing strategies for training these models and combining their outputs to obtain a single prediction as, for example, described in Dietterich (2000) and Yao & Islam (2008). Diversity as a quantification of the disagreement between the models is crucial for the success of a combination as acknowledged in Gabrys & Ruta (2005) and Brown, Wyatt & Tino (2005) which can be illustrated based on the rather intuitive fact that a combination

of predictors that completely agree in all situations will not be more accurate than any of the individual predictors.

Members of an ensemble for prediction can be designed for a classification and a regression problem. For classification problems, where individual predictors produce crisp class labels, the question of diversity still remains an open research issue due to the lack of a concept of distance, although diversity measures for classification have been discussed by, for example, Kuncheva & Whitaker (2003), Gabrys & Ruta (2005) and Tang et al. (2006). For ensembles of regressors, the understanding of diversity is quite mature due to the involvement of several research communities, for example the forecasting community. Regression ensemble diversity can be quite easily quantified by covariance values of the methods in the pool. Two error decompositions applicable for ensembles and combinations are common, which will be presented in the following sections.

6.1.1 The ambiguity decomposition

Krogh & Vedelsby (1995) published an important contribution for the understanding of regression ensembles. The main finding is expressed with Equation 6.1, which is given in the forecast notation with y denoting the actual observation or the target value and $\hat{y}_i$ the individual forecast of model number i.

$$(\hat{y}^c - y)^2 = \sum_i \omega_i (\hat{y}_i - y)^2 - \sum_i \omega_i (\hat{y}_i - \hat{y}^c)^2 \tag{6.1}$$

Furthermore, $\hat{y}^c$ is the linear combination of individual forecasts,

$$\hat{y}^c = \sum_i \omega_i \hat{y}_i, \tag{6.2}$$

with combination weights ω_i being non-negative and summing to one:

$$\sum_i \omega_i = 1,\ \omega_i \geq 0. \tag{6.3}$$

Several proofs for Equation 6.1 have been given, for example in the original paper of Krogh & Vedelsby (1995), in Hansen (2000) and in Brown, Wyatt & Tino (2005). Here, the proof of Brown, Wyatt & Tino (2005) will be given for a better understanding. The starting point is a simple manipulation of the quadratic error equation:

$$\begin{aligned}
\sum_i \omega_i (\hat{y}_i - y)^2 &= \sum_i \omega_i (\hat{y}_i - \hat{y}^c + \hat{y}^c - y)^2 \\
&= \sum_i \omega_i \left[(\hat{y}_i - \hat{y}^c)^2 + (\hat{y}^c - y)^2 + 2(\hat{y}_i - \hat{y}^c)(\hat{y}^c - y) \right] \\
&= \sum_i \omega_i (\hat{y}_i - \hat{y}^c)^2 + \sum_i \omega_i (\hat{y}^c - y)^2 \\
&\quad + \sum_i 2\omega_i (\hat{y}_i - \hat{y}^c)(\hat{y}^c - y)
\end{aligned} \tag{6.4}$$

Now, Equation 6.2 can be used to reduce the second term of Equation 6.4:

$$\begin{aligned}\sum_i \omega_i(\hat{y}^c - y)^2 &= (\hat{y}^c - y)^2 \sum_i \omega_i \\ &= (\hat{y}^c - y)^2 \end{aligned} \tag{6.5}$$

and Equations 6.2 and 6.3 help to eliminate the third term:

$$\begin{aligned}\sum_i 2\omega_i(\hat{y}_i - \hat{y}^c)(\hat{y}^c - y) &= 2(\hat{y}^c - y)\sum_i \omega_i(\hat{y}_i - \hat{y}^c) \\ &= 2(\hat{y}^c - y)\left[\sum_i \omega_i \hat{y}_i - \hat{y}^c \sum_i \omega_i\right] \\ &= 2(\hat{y}^c - y)(\hat{y}^c - \hat{y}^c) \\ &= 0 \end{aligned} \tag{6.6}$$

then

$$\sum_i \omega_i(\hat{y}_i - y)^2 = \sum_i \omega_i(\hat{y}_i - \hat{y}^c)^2 + (\hat{y}^c - y)^2$$

which turns out to be equivalent to Equation 6.1.

Having a closer look at the right hand sight of the Equation 6.1, the first term represents the weighted average error of the individuals. The second term can be referred to as ambiguity, providing a measure for the variability among the ensemble members. The latter term is non-negative, leading to the conclusion that the quadratic error of the ensemble estimator, given by the left side of the equation, will always be less or equal to the average quadratic error of the individual predictors, which is represented by the first term of the right hand side of the equation. This decomposition is also referred to as the ambiguity decomposition and holds for convex weights and mean squared error loss. The bigger the ambiguity term is, the bigger is the overall error reduction. However, an increasing disagreement in the ensemble will also affect the first term of the decomposition, so that only the right balance between the two terms can produce the lowest overall ensemble error. This decomposition provides the basis for a successful ensemble training technique "negative correlation learning" first introduced in Liu & Yao (1999).

Brown, Wyatt & Tino (2005) relate the ambiguity decomposition to a bias-variance-covariance decomposition and provide an error function that can be used for explicitly controlling the diversity of an ensemble, which has been exploited as a diversity measure for the meta-learning experiments in Section 5.3.1.4 and will be further explored in the next section. Hansen (2000) provides a more comprehensive treatment of the ambiguity decomposition and its implications for machine learning, however, in the context of the thesis, the concept that individual accuracy of ensemble members has to be balanced with their diversity in an effective combination shall suffice.

6.1.2 Bias/variance/covariance

The bias/variance dilemma is well-known and studied in estimation theory and modelling; one of the seminal papers relating it to machine learning in general and neural network ensembles in particular has been published by Geman et al. (1992).

It assumes that the estimation error can be decomposed into a bias and a variance component:

- The **bias** component represents the expected error over all trained ensembles, i.e. the expected loss when using the average combined predictor for forecasting target y. This part of the error results from incorrect models and the fact that the class of functions used for modelling a data generation process may not include the correct one. Since the correct models are hard to find and identify for real-world problems, a bias component will be inherent in a forecasting error.

- The **variance** denotes the degree of variability between one particular combined predictor and the average over all combined predictors. It can be attributed to parameter sets not being perfectly estimated due to limitations of the available training data.

In very simple models, errors will usually have a large bias component, as the predictors are not sufficiently complex to model the data generation process. However, they will agree in their predictions as there are not many parameters to estimate, producing a low variance component. On the other hand, if more complex models are chosen for modelling, the number of parameters to estimate becomes large. Extremely and sometimes prohibitively large training sets are necessary for an appropriate parameter estimation, with the estimation being extremely slow to converge. In this scenario, the bias component of the error would be small at the expense of a much bigger variance. This illustrates the trade-off between the two components; reducing one of them will cause an increase in the other. It also provides an explanation for the fact that a simple model can outperform a more complex one when training data is limited or noisy, causing an increased risk of overfitting and making the parameter estimation too dependent on the actual training sample.

When not considering a combination approach as a single learning unit, Brown, Wyatt, Harris & Yao (2005) add another component to the error decomposition: the covariance.

- The **covariance** component of a combination forecast error is the averaged covariance of the individual methods. Highly correlated individual forecasts increase the error, but again the covariance cannot be reduced without affecting bias and variance values. The two-way bias/variance trade-off when designing an individual predictor thus becomes a three-way trade-off for combinations.

6.1.3 Motivation for diversification

Flat combinations and meta-learning of methods for airline cancellation forecasts showed interesting results in the previous sections, but did not produce consistent improvements over the individual predictors. For further improvements, the generation of individual forecasts additional to the presented four methods seems most promising. Unfortunately, the choice of available methods is limited, as only a few simple and robust methods can deal with the noisy data, the small number problems and comply with the time restrictions of the airline application. A number of other suggestions regarding the creation of additional forecasts for an ensemble can be found in the scientific literature, which will be investigated in the next section and fitted to the airline application whenever applicable. As discussed previously

in this section, individual forecasts need to have certain characteristics to provide a beneficial contribution in a combination, which can be explained with the help of error decompositions. Diversification ideas are examined in this respect as well, with the aim of exploring the resulting forecasts and their characteristics.

6.2 Generating forecasts by diversification procedures

Diversification algorithms have been studied in different research and application areas, with the most mature contributions seeming to be available in the machine learning community. Publications especially relevant to classifier diversity and methods of how to encourage generation of diverse ensembles can be found in Kuncheva & Whitaker (2003) and Kuncheva (2004). Brown, Wyatt & Tino (2005) propose a taxonomy for creating diversity in neural network ensembles, distinguishing

- different starting points in the hypothesis space, for example by random weight initialisation,
- manipulating accessible hypotheses by changing training data or network architectures and
- changing the way the hypothesis space is traversed, for example by negative correlation learning as mentioned earlier.

Sharkey & Sharkey (1997) separate "blind" approaches of random methods to achieve diversity (random architectures, weights, initialisations) from active attempts to create an ensemble of predictors that can effectively be combined, which also fits the concept of implicit and explicit diversity creation methods discussed in Tang et al. (2006).

The following sections will investigate diversification approaches by looking at literature relevant in the context of this thesis, and, similar to the approach used in Riedel & Gabrys (2009), relating them to the bias-variance-covariance decomposition of errors. Furthermore, the possible applicability for the airline data set is explored at the same time.

6.2.1 Decomposing data

In Section 5.3.1.1, it was discussed to decompose time series into a trend component and a seasonal component. In a more general case, the existence of different components in a time series can lead to a situation where input forecasts predict one or more of the components in a similar manner. This will lead to an increased correlation between the individual forecasts which is generally undesirable for a combination. Decomposing time series and predicting components separately as, for example, described in Makridakis et al. (1998) can thus provide a way to avoid these difficulties common in industrial practice as shown in Jain (2008).

At Lufthansa Systems, decomposition is one of the key concepts used. As shown in Figure 6.1, historical data is split into a demand and a cancellation component, with the demand further being split into a seasonality and an attractiveness component as described in Section 2.1.4.4 which are all predicted separately.

As this thesis is concerned with improvements of the cancellation forecast which is not further split into components, this type of diversification is not used.

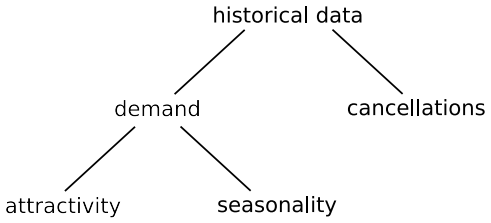

Figure 6.1: Decomposition of data in the airline application

6.2.2 Diversifying functional approaches

One of the first explicit discussions of diversity in the context of combinations of forecasts can be found in Batchelor & Dua (1995), who suggest that functionally different approaches are very unlikely to produce errors that are strongly correlated and are thus suitable for forecast combination. A number of empirical studies like Zhang (2004), Terui & van Dijk (2002) and Swanson & Zeng (2001) illustrate the benefits of including models clearly differing in complexity in a combination of forecasts, the complexity difference mostly being achieved by a pool of linear and nonlinear models. This generates models with different trade-offs between their error bias and variance components, ideally leading to uncorrelated errors and compensation effects in a combination. Another general approach is using a strongly restricted approach as prediction model, producing forecast errors with a high bias and a low variance component with a combination increasing complexity and reducing the bias term in the process, which will however only work if the biases are not strongly correlated.

As discussed in Section 2.1.4, individual algorithms employed at Lufthansa Systems have to be fairly simple due to the nature of the data, so that including a nonlinear method as an additional method in the pool is not an option. However, as described in the same section, three different methods are readily available for cancellation forecasting, two being based on exponential smoothing and one on regression. An algorithm based on cancellation probabilities was added in the scope of this thesis as described in Section 4.2 to further enforce diversity in the functional approaches and potentially benefiting the accuracy of the combined forecast.

6.2.3 Diversifying parameters

Granger & Jeon (2004) criticise ordinary model-building approaches consisting of a single specification and its estimation as unrealistic. They suggest taking into account alternative specifications of similar quality and combine the different results, a methodology referred to as “thick modelling”. Even though alternative specifications may mean using different models or data sets for training to obtain different forecasts, a major application of the concept is using different parameters for the same model, which is also how thick modelling is used for airline demand forecasting in Riedel (2007). Fixing certain parameters which would normally be learnt or estimated reduces the error variance at the expense of an increased bias, which can however again be compensated for in the combination process. On the other hand, if more specifications are considered as opposed to only using one model, again different models will emerge with different bias/variance trade-offs. If learning of the models takes place on the same data, highly correlated variance components are very likely, but uncorrelated parts can exist in the bias component. This means that

the bias component of the error needs to be relevant in order to generate uncorrelated forecasts with this type of diversification. Related experiments are presented in Section 6.4.2.

6.2.4 Diversifying training data

Linear and nonlinear transformations as reviewed in Fodor (2002) provide different ways of representing data and using it for building the models, often reducing dimensionality at the same time. Principal component analysis as, for example, described in Jolliffe (2002) is a very popular approach belonging to this group of methods.

The bias-variance decomposition explains the variance component as a result of using one particular limited sample of the training data. Manipulating training data by resampling techniques and training a different model on each one of the sampled data partitions has the potential of reducing the variance component when combining the resulting predictions. The approach of resampling is used by a number of ensemble generation techniques:

- Bagging as introduced by Breiman (1996) denotes random sampling with replacement with the number of samples having the same size as the original training set.
- Manipulating the probability which samples are drawn from the original training data, boosting according to Freund & Schapire (1997) sequentially trains classifiers and iteratively changes the probabilities of selecting specific data points from the training set. Data instances which were previously misclassified by the other predictors, get an increased selection probability.

Hansen (2000) states that ensemble generation techniques based on resampling are mainly useful for individual predictors that are complex and flexible enough to potentially overfit on training data and hence have a low bias and a high variance component. In this case, resampling reduces the overfitting risk and potentially provides higher estimation accuracy. The forecasting techniques used at Lufthansa Systems however provide a different approach to ensure generalisation ability and prevent overfitting in using reference curves, confidence limits and strongly restricted prediction models as discussed in Section 2.1.4, so that resampling is not considered in this work.

However, the fact that airline data can be aggregated and disaggregated on different levels can be exploited for a diversification of the training data. Figure 6.2 illustrates a decomposition: A time series is first decomposed into the three different points of sale, with each of them furthermore being decomposed regarding day of the week. A recent example of a publication investigating approaches of hierarchical forecasting in the area of tourism can be found in Athanasopoulos et al. (2009).

Currently, history for airline cancellation forecasts is built on the finest possible level, which means using data collected per fareclass, day of week, point of sale and origin-destination-itinerary. On this finest level, important characteristics that are only visible when looking at the bigger picture, i.e. a higher aggregation level, might be lost, and when the training data is noisy, a high error variance component becomes likely. On the other hand, using the high level for model building might omit characteristics specific to certain parts of the data although having a variance-stabilising effect, leading to inferior forecasts as well. Generating forecasts learnt on a higher aggregation level of the data will not affect the bias component compared

to the low level forecasts, as the prediction model used remains the same, but an impact on the variance error component can be expected. Hence, the error variance component needs to be relevant in comparison to the bias component for a bigger potential for accuracy improvement. Experiments concerning multi-level learning are described in Section 6.4.1.

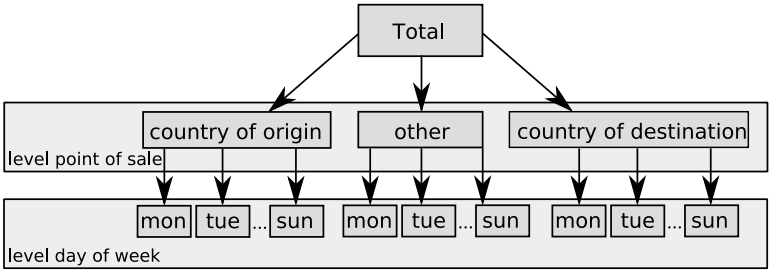

Figure 6.2: Example of different aggregation levels of airline data

6.2.5 Summary

Approaches to generate additional forecasts for inclusion in a combination process have been discussed in this section, including a discussion of the effects the diversification procedures will have on the different components of the error. Some of the methods presented are applicable to the airline application investigated here: diversifying the functional approaches, diversifying parameter sets and diversification of sources of training data by using data aggregated from different aggregation levels. The classic approach of using functionally different approaches has been pursued in Chapter 4. Before introducing and analysing experiments regarding the other two diversification strategies in the remainder of this chapter, additional explanations about application-specific calculations and their effect on the error composition are given.

6.3 Application-specific dynamics of the error components

Improving airline net booking forecasts by investigating the modification of the cancellation component is a complex procedure as described in Chapter 4, with the different steps having various effects on the error decomposition. This section will relate the bias/variance error decomposition to two calculations specific to the application and thus provide the basis for the discussion of the diversification experiments presented later in this chapter.

6.3.1 The interaction with the booking forecast

The first aspect to look at is the interaction of the predicted cancellations with the predicted bookings. Both are components needed to calculate the net booking forecast, which is the basis for assessing forecasting performance in this work as explained in Section 4.2. Both booking and cancellation forecast naturally have errors that can be split into a bias and a variance component. Since the approaches used for each of the forecasts differ in their functional approach, strong correlations of the bias components are unlikely, which will not lead to consistent compensation

or accumulation effects when subtracting the cancellation forecast from the booking forecast to obtain the prediction of net bookings. However, it is likely that the error variance of the two components is positively correlated, as the training data for both components is subject to similar influences of noise, changing environment or structural breaks at the same time. This will lead to compensation of this part of the error when subtracting the two and create a situation in which a reduced variance component in the cancellation forecast does not necessarily generate a better net booking forecast, which was already observed in Section 4.2.

6.3.2 Aggregating

Performance results for the airline application are normally evaluated on the finest possible aggregation level, which is the level on which forecasts are needed in the productive system. However, for visualisation and decision making purposes, forecasts are often also aggregated to a higher level, which has already been done for the empirical experiments in the Chapters 4 and 5 and will be continued in this chapter. The improvement of the low level performance is the main objective of this work, however, an improved high-level forecast is certainly desirable as well.

Riedel (2007) gives an analysis of the dynamics of the different error components when aggregating forecasts on a higher level. In general, the aggregation process sums up relative values of errors from different subspaces, with the subspaces being given by the different fareclasses and points of sale in this application. Positively correlated errors in the different subspaces will lead to undesirable accumulation effects while uncorrelated or negatively correlated errors will cause beneficial compensation. As discussed previously, the error variance component is related to parameter estimation error which is strongly affected by noisy data. According to Riedel (2007), the noise tends to be highly correlated between the different subspaces in the airline application, which can result in highly correlated error variance terms which will accumulate on the high level. The bias error component on the other hand has a bigger chance of being compensated in the aggregation process, however, positive correlations might still exist to a lesser extent.

6.4 Flat combinations of diversified forecasts

This section will look at flat combinations of a pool of individual methods, which was, in comparison to the experiments presented in Chapter 4, extended by a number of diversification approaches explained previously in this chapter. The experiments presented in this section have been conducted using the same methodology as described in Chapter 4 with a minor change in the choice of combination methods: the unrestricted regression method has been omitted from the combination pool, as weight estimation turned out to be extremely unstable for the increased number of individual methods. Instead of the simple average taking all available methods into account, which consistently provided worse results than the simple average with trimming of 20% on this data set, a simple average using only the four historically best performing methods labelled as “avg4” has been included.

6.4.1 Diversifying level of learning

The first experiment of this chapter looks at diversifying the aggregation level of the data used for building a forecasting model as motivated in Section 6.2.4. Similar to

the work done in Riedel (2007), the training data for building the reference curves has been used on the finest possible level as well as aggregated over the different compartments. Each of the four forecasting models is therefore built using data from two different aggregation levels, producing eight individual forecasts as inputs for the combination.

Table 6.1 shows that level diversifications fail to produce successful combination results. Results on the low level outperformed the flat combinations of Chapter 4 in some cases, but not the best individual predictor. The idea of including high-level information in building cancellation models was to stabilise the error variance component at the expense of the bias component. This has most probably backfired when subtracting the cancellation forecast from the booking forecast, where the error variances of the two components were not able to compensate each other any longer as described in Section 6.3.1. The forecasts added to the pool were thus inferior, not being able to produce a performance gain by better balancing individual accuracy and diversity in the ensemble. Furthermore, the composition of the errors of the resulting low level forecasts was less beneficial for aggregation to the higher level. The optimal model again performs worst of all, suffering from a badly estimated covariance matrix due to limited training data and a higher number of individual forecasts.

The graphs in Figure 6.3 illustrate how including high level information can be beneficial in some cases and detrimental in others. Both pictures show reference curves from high and low levels, averaged over calender weeks. The lack of change in the first 52 weeks appears due to the initialisation period. The left picture shows a fareclass in which it is clear that the fine level curve fits the data much better and the high level curve differs significantly from the actual data. The right picture shows a shift in the data, which makes the high level curve more accurate in some cases. However, these beneficial cases seem to be the exception, explaining the bad performance of the resulting additional forecasts.

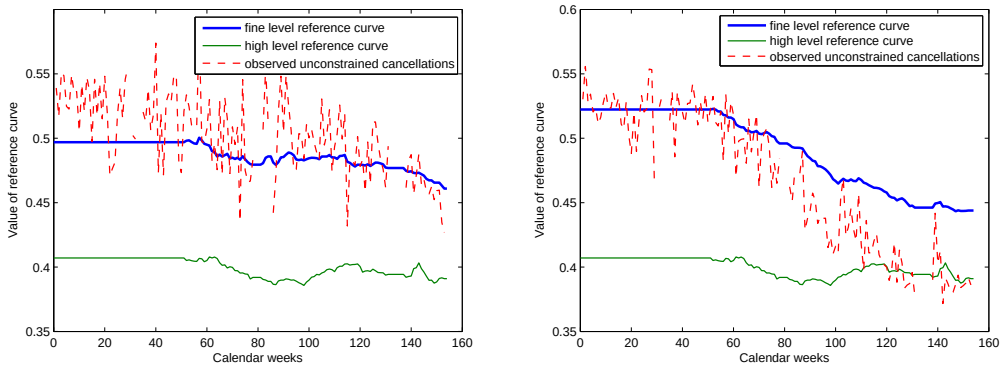

Figure 6.3: Reference curves learnt on high level and low level and actual data averaged per calendar week. Left: scenario in which the low level curve corresponds to data much better, right: scenario where higher level information can be beneficial towards the end

DCP	avg4	sat	outp	var	opt	regr
0	6.01	4.53	**7.53**	4.42	-9.28	6.46
1	4.17	1.56	**7.38**	1.26	-5.23	4.31
2	0.95	-1.86	**1.26**	-3.17	-11.68	0.95
3	0.14	-2.37	**0.65**	-3.53	-16.77	0.23
4	-1.24	-2.72	**0.21**	-3.43	-22.87	-1.23
5	-1.42	-2.83	**1.33**	-2.98	-10.56	-1.35
6	-2.74	-3.51	**0.42**	-3.83	-12.05	-2.65
7	-2.45	-3.47	**-0.45**	-3.84	-12.62	-2.46
8	-2.78	-3.10	**-0.85**	-3.28	-14.58	-2.74
9	-2.15	-2.35	**-0.85**	-2.79	-11.45	-2.20
10	-1.51	-2.16	**-1.38**	-2.50	-12.40	-1.52
11	**-0.62**	-1.18	-1.41	-1.65	-14.69	-0.60
12	**-0.28**	-0.81	-1.02	-0.90	-14.22	-0.32
13	0.73	0.41	**0.85**	0.71	-11.59	0.71
14	**0.73**	0.21	0.27	0.63	-15.48	0.59
15	0.53	0.60	0.44	**0.74**	-13.93	0.54
16	0.44	0.11	0.40	**0.56**	-15.33	0.38
17	0.24	-0.03	**0.33**	0.29	-16.18	0.23
18	-0.11	-0.08	-0.22	**0.06**	-18.79	-0.21
19	0.19	-0.08	-0.47	0.02	-13.49	**0.09**
20	-0.41	-0.91	-1.27	-0.77	-15.13	**-0.39**
21	-0.66	-1.13	-1.67	-0.88	-13.82	**-0.64**
avg	**-0.10**	**-0.96**	**0.52**	**-1.13**	**-13.73**	**-0.08**

DCP	avg4	sat	outp	var	opt	regr
0	4.60	3.33	4.15	2.97	-23.34	**4.73**
1	2.20	0.96	**2.82**	0.83	-23.76	2.04
2	0.79	-0.59	**0.99**	-0.85	-27.36	0.71
3	-0.01	-1.09	**0.24**	-1.43	-28.79	-0.03
4	**-0.93**	-1.87	-1.12	-2.37	-29.71	-0.95
5	**-1.45**	-2.29	-1.62	-2.75	-28.76	-1.46
6	-1.89	-2.61	**-1.77**	-2.79	-27.93	-1.96
7	**-1.70**	-2.39	-1.89	-2.57	-25.36	-1.78
8	**-1.68**	-2.03	-1.71	-2.24	-23.59	-1.75
9	**-1.14**	-1.61	-1.52	-1.86	-21.76	-1.20
10	**-0.76**	-1.24	-1.40	-1.60	-20.06	-0.77
11	**-0.25**	-0.67	-0.90	-0.98	-19.88	-0.31
12	**0.17**	-0.21	-0.78	-0.64	-19.98	0.10
13	**0.41**	0.08	-0.79	-0.35	-19.54	0.33
14	**0.64**	0.34	-1.19	-0.24	-19.34	0.54
15	**0.81**	0.57	-1.76	-0.13	-19.70	0.75
16	**-0.52**	-0.95	-3.89	-1.80	-20.69	-0.62
17	**-0.36**	-0.67	-4.79	-1.71	-22.25	-0.46
18	**-0.22**	-0.70	-6.07	-1.83	-26.17	-0.30
19	**0.09**	-0.37	-7.71	-1.83	-25.39	-0.00
20	**0.48**	-0.23	-9.67	-1.85	-29.97	0.47
21	**0.86**	-0.15	-11.89	-1.33	-30.75	0.80
avg	**0.01**	**-0.65**	**-2.38**	**-1.24**	**-24.28**	**-0.05**

Table 6.1: Level diversification: percentage of relative performance improvement compared to reference forecast, top: high level, bottom: low level

6.4.2 Diversifying the smoothing parameter

Looking at the four individual forecasting methods described in Section 2.1.4, two parameters are suitable for diversification:

- The smoothing parameter α for the reference curve update, which is present in the two smoothing models and the probability forecast and
- the width of the confidence limits applied to the observed data before forecasting and history building, used in the two smoothing models and the regression model.

Both of these choices affect the adaptation capabilities of the data in a similar manner. With a higher smoothing parameter, the adaptation of the reference curve is stronger, consequently believing the new data to a higher extent. By applying wider confidence limits around the reference curve, again belief in the current data is stronger by allowing a wider range of values around the current reference curve.

However, it is expected that diversification of the smoothing parameter will produce more diverse forecasts than diversification of the confidence limits, because manipulating the reference curves directly affects all data at all DCPs, while manipulating confidence limits only influences predictions under certain conditions as described in Section 2.1.4. A look into Figure 6.4 illustrates this hypothesis by giving average error values for the diversified forecasts for a sample origin-destination opportunity, clearly showing a bigger impact of the smoothing parameter.

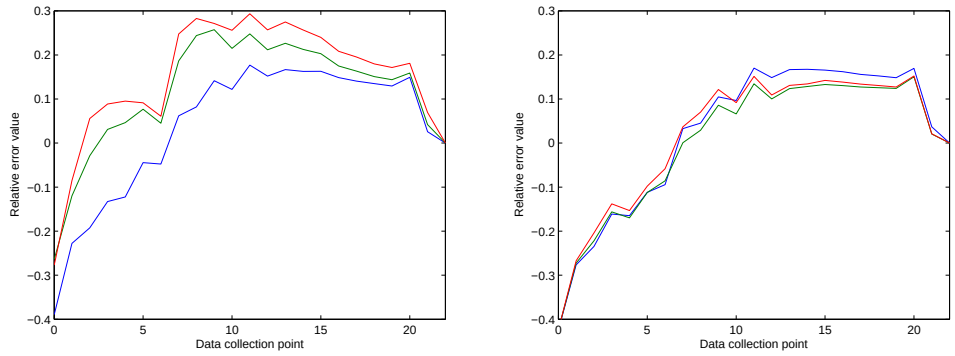

Figure 6.4: Average cancellation forecast errors, with the different curves corresponding to three different parameters used, smoothing factor (left) and confidence limit width (right).

Parameter diversification experiments have been conducted using the same methodology and combination methods as described in Chapter 4. As justified above, the smoothing parameter has been diversified using three values, including the one it was previously fixed at for the two smoothing based methods and the method based on probability. The regression approach however does not have a smoothing parameter, which is why the confidence limits have been diversified in this method's case, also using three different values. The four methods with three different parameters produce twelve different individual forecasts for the combination methods investigated, which are the same as in the level diversification experiment. Combination performance results for the low and high level are given in the Table 6.2.

DCP	avg4	sat	outp	var	opt	regr
0	**6.71**	6.13	3.72	5.46	1.91	6.67
1	**7.31**	5.93	4.24	5.51	2.71	4.48
2	**5.02**	3.68	1.64	3.05	-0.82	2.50
3	**3.93**	3.04	0.69	2.46	-2.85	1.48
4	**3.25**	2.32	-0.32	1.50	-0.87	0.47
5	**2.45**	1.42	-1.40	0.51	-4.37	-0.84
6	**1.82**	1.24	-1.33	0.23	-8.08	-0.89
7	**2.08**	1.20	-2.16	0.04	-4.82	-1.38
8	**1.48**	1.29	-2.27	0.00	-9.88	-1.43
9	**2.20**	1.81	-1.18	0.86	-8.97	-0.29
10	1.61	**1.68**	-1.19	0.70	-9.66	-0.62
11	1.17	**1.33**	-1.19	0.51	-9.76	-0.37
12	1.53	**2.09**	-0.06	1.50	-10.17	0.79
13	2.25	**2.85**	1.16	2.59	-10.96	1.86
14	1.15	**2.18**	0.86	1.94	-12.38	1.01
15	0.84	**1.46**	0.62	1.42	-13.21	0.89
16	**9.87**	7.23	8.95	7.46	1.11	5.59
17	**9.78**	6.41	8.28	6.79	-0.18	5.07
18	**10.23**	6.31	8.21	6.64	-10.78	4.66
19	**9.83**	6.53	8.22	6.59	1.49	4.51
20	**9.22**	5.72	7.42	6.03	-0.76	4.07
21	**6.68**	3.72	5.00	4.19	-3.94	2.37
avg	**4.56**	**3.43**	**2.18**	**3.00**	**-5.24**	**1.85**

DCP	avg4	sat	outp	var	opt	regr
0	**2.92**	1.81	-1.31	1.24	-8.70	0.64
1	**2.68**	1.87	0.44	1.80	-14.17	0.96
2	**2.39**	1.48	-0.09	1.36	-16.13	0.66
3	**2.03**	1.43	-0.45	1.15	-17.73	0.37
4	**1.63**	0.93	-1.51	0.44	-15.79	-0.46
5	**1.21**	0.58	-1.94	0.05	-14.78	-0.86
6	**1.06**	0.65	-1.95	0.00	-15.41	-0.94
7	**1.19**	0.78	-1.75	0.11	-13.32	-1.30
8	**1.02**	0.92	-1.58	0.19	-13.86	-0.76
9	**1.50**	1.07	-1.30	0.39	-14.00	-0.63
10	**1.72**	1.20	-0.87	0.55	-13.67	-0.43
11	**2.06**	1.53	-0.11	1.00	-13.73	0.03
12	**2.22**	1.77	0.36	1.31	-13.30	0.39
13	**2.41**	1.88	0.94	1.52	-13.13	0.71
14	**2.22**	1.92	1.16	1.65	-14.59	0.65
15	**2.16**	1.78	0.98	1.52	-15.47	0.59
16	**15.78**	10.49	13.73	10.49	9.31	7.40
17	**15.69**	10.25	13.49	10.38	9.08	7.30
18	**15.29**	9.88	12.89	9.96	3.62	6.90
19	**14.52**	9.33	11.92	9.36	6.33	6.36
20	**12.89**	8.17	10.20	8.17	1.68	5.25
21	**9.43**	5.95	7.14	5.99	-8.34	3.18
avg	**5.18**	**3.44**	**2.74**	**3.12**	**-9.37**	**1.64**

Table 6.2: Parameter diversification: percentage of relative performance improvement compared to reference forecast, top: high level, bottom: low level

The results look promising. With the variance-based method and the simple average with different degrees of trimming, the individual forecasts could be outperformed consistently for all data collection points on the low and high level. The high performance gains at the later DCPs can however be a bit misleading due to the absolute numbers being very small, especially at the low level. The outperformance and regression model again suffer from parameter estimation instabilities.

6.5 Advanced combination techniques

The experiments presented in the previous section showed some promising results for flat combinations of forecasts generated by a number of diversification procedures. This section now investigates if further improvements can be achieved by using more advanced combination techniques. In order to obtain a combination result with increased accuracy, Riedel (2007) analyses a number of desirable characteristics that a forecast ensemble should have, which can be summarised in the following points:

- Individual forecasts should contain diverse information but still have reasonable individual performance. This has also been described using the ambiguity decomposition in formula 6.1 at the beginning of this chapter. The total number should be restricted using trimming approaches, which has for example been reviewed in Timmermann (2006).
- In general, individual forecasts should have a low error variance component. If error variance is high in relation to the bias, impact of the training data would increase and so would the positive correlation of forecast errors, which is detrimental for the combination, especially for approaches that perform some kind of parameter estimation.
- Homogeneous error variance and correlation values are desirable. Both Riedel (2007) and Timmermann (2006) come to the conclusion that loss expected from not taking variance and/or covariance information into account when using a simpler combination model as opposed to the optimal model is high when error variances and covariances are inhomogeneous.

These criteria are not sufficiently met with the experiments described above: with the diversification procedures, the number of forecasts to combine becomes rather big and the noisy data will cause high error variance components. As described when discussing diversification approaches in Section 6.2, each has the potential to produce forecast errors that are at least uncorrelated in some parts, so that a homogeneous covariance matrix is not realistic. As a solution to this problem, the next sections are looking at clustering approaches, multi-step combination structures and evolutionary computation in relation to the airline application. The approaches are empirically evaluated and analysed.

6.5.1 Pooling and multilevel structures

Pooling denotes the division of a combination task into several subtasks called pools, with a final combination generating the overall prediction from each of the pool's outputs. This can be implemented on several levels, so that multi-step structures are generated. In each of the pools, the available input forecasts can be combined by a combination approach as well. This provides a number of advantages: it limits

the number of forecasts to combine in each step, which can be further enforced by using trimming. Ideally, the covariance matrix of the forecasts of one pool would also be homogeneous, and a combination with a simple method would reduce total error.

A popular pooling approach introduced by Aiolfi & Timmermann (2006) has already been mentioned in Chapter 3 and was very successful in the experiments on the competition data. It groups forecasts into clusters using the k-means algorithm on their past forecast errors. Figure 6.5 shows a sample combination structure with three clusters, where the number of forecasts in each cluster is restricted to five and the cluster including the forecasts with the highest error is discarded. The individual forecasts are again given by $\hat{y}_i$ with i being the forecast number.

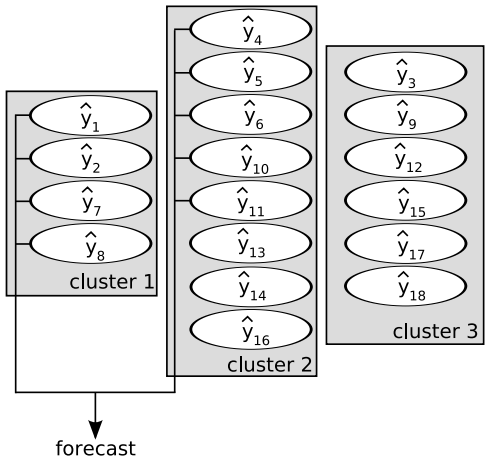

Figure 6.5: Example of a combination structure generated by variance-based pooling

Riedel & Gabrys (2007) however argue that the variance-based combination approach risks high losses of accuracy if the covariance matrices are not homogeneous, as it does not take correlation between forecasts into account. As a remedy, clustering based on the covariance matrix is recommended if possible. Looking back at the discussion about the "forecast combination puzzle" in Section 3.1.2, this brings up a familiar dilemma: using a simpler combination method will have accuracy losses compared to a more complex method taking covariance into account, but since it is error-prone and time-consuming to estimate covariance information, the theoretically optimal weights do not necessarily perform best.

Riedel & Gabrys (2007) hence suggest taking knowledge of the forecast generation process of the individual forecasts into account when generating pools. According to the discussion, only forecasts differing in one aspect of their forecast generation process should be pooled, otherwise, the risk of strongly inhomogeneous covariance matrices in each of the pools increases and will have a negative impact on performance of resulting output forecasts and overall forecast accuracy. An illustration can be found in Figure 6.6: eight forecasts have been generated using two different methods $m1$ and $m2$, two different parameter sets $p1$ and $p2$ and two levels of learning $l1$ and $l2$. In the first step, the level of learning is chosen as first pooling "dimension". The pooling process aggregates this level dimension and produces four forecasts with differing methods and parameter sets. The next level of pooling then

aggregates the parameter dimension before combining the resulting two forecasts that now only differ in the method used.

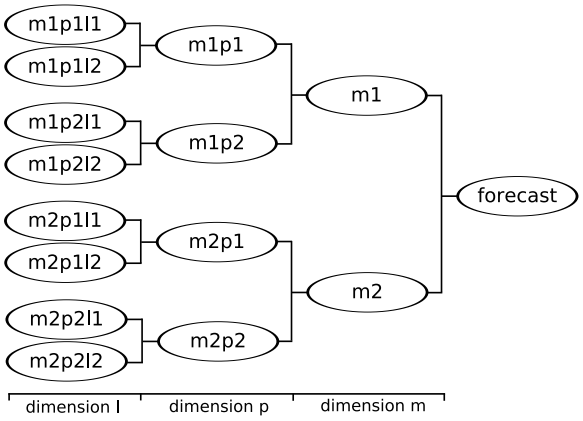

Figure 6.6: Illustration of a combination structure generated by dimension-specific pooling

A number of parameters have to be identified when generating multi-level structures for combining forecasts: which dimension should first be aggregated? How many methods should be allowed per cluster, how many trimmed? Which combination method is most beneficial for the combinations in the sub-problems? Riedel & Gabrys (2004) and Riedel & Gabrys (2005) propose solutions for these questions, however, it is also stated that predefining structures requires a lot of expert knowledge and would have to be verified in trial and error procedures. Using automatically evolved structures is suggested to overcome these difficulties in Riedel (2007), which will be discussed and implemented in the next section.

6.5.2 Evolving multilevel structures

Evolutionary computation is an area of artificial intelligence that was inspired by evolutionary biology. It involves metaheuristic optimisation techniques and includes, among others, the subfields of genetic algorithms as first introduced by Goldberg & Holland (1988) and genetic programming as described in Koza (1992). Initially, a population of individuals representing potential solutions to a given optimisation problem is randomly generated. In subsequent steps, new and better sets, so called generations, of individuals are evolved using genetic operators like mutation, crossover and selection. A fitness function evaluates the quality of the candidate solutions, until a certain fitness level has been reached or a predefined number of generations has been produced. The next sections will fit the problem of finding well-performing combination structures into the framework of an evolutionary approach and present results of empirical experiments before more closely analysing the resulting combination structures.

6.5.3 Experimental setup

Genetic programming is an extension of genetic algorithms, where individuals in the population are tree-like structures with the leafs representing arguments that are passed to the nodes which then apply a primitive function. Traditionally, it is

used for evolving actual computer programs with the fitness function assessing a program's ability to perform a given task. These tree-like structures do however correspond quite well to the forecasting structures described in the previous section, which is why Riedel (2007) formulate the problem of finding suitable combination structures as an optimisation problem that can be solved with an evolutionary algorithm similar to genetic programming. The following details of the evolution have been identified:

- The leafs of the generated tree represent the input forecasts of the ensemble.
- The nodes represent the forecast combination algorithms available, each of which can apply a certain degree of trimming.
- The fitness function evaluates the mean absolute deviation of the generated structures in a validation period. The possibility of using penalty terms for enforcing diversity has been discarded as other ways of dealing with diversity are applied, for example following the concept of dimension-specific pooling as described previously.
- An initial population consisting of eight individual combination structures is generated either randomly or corresponding to the idea of dimension-specific pooling from Section 6.5.1.
- The standard genetic operators crossover and mutation can be used without application-specific changes.
- A crossover randomly exchanges subtrees in two parents, which can either be a single combination procedure or a substructure. A maximum crossover number of 100 has been used in line with the experiments presented in Chapter 7 of Riedel (2007).
- Mutation randomly exchanges a terminal (input forecast) or a primitive function (combination algorithm) with a 20% probability in each step.
- The algorithm stops early if performance has not improved in the last 50 generations.

The following list enumerates actual methods that have been evaluated in this empirical experiment with the abbreviations used in the tables.

- **ev1**: Dynamic structures with varying complexity are evolved with this algorithm. A population of eight structures is randomly initialised, each level containing between two and five combination procedures. Combination methods are determined randomly as well, choosing between the average, outperformance, variance-based and optimal model with a random degree of trimming per pool. One parameter differs in comparison to the experiments in Riedel (2007): the maximum number of combination levels allowed. It has been reduced from four to two here to avoid problems that occured with overfitting.
- **ev2**: In order to avoid covariance inhomogeneities, dimension-specific pooling according to Section 6.5.1 has been implemented. Evolution takes place modifying the order in which the dimensions are aggregated. Additionally, a global percentage of trimming is evolved, while the combination method is fixed to the variance-based one.

- **ev3**: This method is similar to ev2, with the only difference of now additionally including the combination method in the aspects that are randomly generated and changed during the evolution.
- **ev4**: Again, this methods corresponds to the previous one, but also evolves the maximum number of forecasts allowed per pool.
- **tim**: For comparison, the pooling method introduced by Aiolfi & Timmermann (2006) is considered. This approach only takes past error variance of forecasts into account, thus discarding information on the forecast error correlations.

6.5.4 Results

Experiments have been run using only parameter diversified forecasts as well as using parameter and level diversified forecasts. For the first experiments, the standard booking forecast was used as a basis for calculating net bookings, results are given in Table 6.3. For the final experiments, an improved booking forecast obtained by the EV8 algorithm of the experiments in Chapter 7 of the thesis of Riedel (2007) has been used in order to assess the interaction between the algorithms presented in this thesis and the previous project, with results given in Table 6.4. Again, it has to be emphasized that performance gains at the later DCPs should be taken with some care, as the absolute numbers are very small and lead to big percentage improvements. The discussion of results will thus focus on the first half of the DCPs.

Looking at the performances of combination structures of parameter and level diversified forecasts in the top part of Table 6.3, a number of improvements can be observed. On the low level, improvements of up to 8% for the first DCPs are achieved, but deteriorate quickly with ascending DCPs. On the high level, consistent improvements can be found using the variance-based pooling and the ev3 algorithm with a peak improvement of 10.7% on the second DCP of the ev1 algorithm. More consistent improvements can be seen for only using parameter-diversified individual forecasts in the bottom part of Table 6.3, allthough peak improvements only as high as 4% on the low level and 9.9% on the high level are achieved. Using the improved booking forecast does not notably change results for the low level parameter and level diversified forecasts. However, considerable improvements can be seen on the high level for both sets of individual forecasts, with improvements of now up to 14.8% on the high level and 6% for the low level structures using the parameter diversified forecasts.

Comparing the different algorithms used, it can be said that the variance-based pooling and the ev2 algorithm using dimension-specific pooling with a fixed combination method are usually outperformed by the other methods. It is thus safe to say that dynamically evolving a combination method for the clusters has proven to be very beneficial in all cases. Algorithms ev3 and ev4 only differ in their approach to perform trimming within the clusters, where ev3 fixes a maximum number of forecasts and ev4 dynamically evolves it. Comparing performances of the two, results are inconclusive.

The algorithms ev1, ev3 and ev4 generally perform quite similarly, with advantages of the ev1 algorithm especially when using the improved booking forecast. This is a contradiction to the results of Riedel (2007), where dimension-specific pooling clearly outperformed the more flexible approach. A few reasons can be given for this: in the previous work, four different diversifications were used (4 methods, 4

	high level					low level				
DCP	**tim**	**ev1**	**ev2**	**ev3**	**ev4**	**tim**	**ev1**	**ev2**	**ev3**	**ev4**
0	8.3	5.0	5.8	**9.6**	5.6	6.1	5.5	4.8	**8.0**	**6.7**
1	9.4	**10.7**	6.2	4.0	5.9	**3.6**	3.5	2.2	0.9	0.4
2	3.2	**5.7**	2.2	3.1	-0.8	2.8	**3.0**	0.6	2.3	0.9
3	3.3	**4.8**	2.1	3.9	-0.1	1.2	2.2	0.1	**2.6**	1.8
4	2.5	2.2	0.4	1.4	**2.9**	**1.1**	**1.1**	-0.7	0.1	0.9
5	2.3	**3.1**	-0.1	1.4	2.9	-1.0	**0.9**	-1.5	-0.9	**0.9**
6	1.2	**2.2**	-0.0	0.6	1.8	-0.4	**0.8**	-0.8	-0.2	0.7
7	**1.1**	-0.1	-2.2	0.5	0.9	-0.6	**0.4**	-1.5	-0.7	-0.3
8	**0.4**	0.3	-1.8	0.1	0.4	-0.5	-0.2	-1.3	-0.7	-0.2
9	0.5	-0.0	-0.8	**0.6**	0.1	**0.2**	-0.1	-0.5	-0.1	**0.2**
10	0.7	**1.1**	-0.3	0.9	0.8	0.2	0.2	0.0	-0.3	**0.3**
11	0.2	-0.1	-0.2	0.1	0.3	1.2	1.0	**1.3**	1.0	1.2
12	0.4	0.1	0.6	**1.6**	1.3	1.7	1.8	1.8	1.8	**1.9**
13	1.8	2.0	1.9	**2.7**	2.1	2.3	2.1	**2.5**	2.1	2.3
14	1.1	1.7	1.7	**2.1**	1.8	2.4	2.7	**3.1**	2.8	2.8
15	1.2	1.5	1.4	**1.8**	1.4	2.7	**3.4**	3.2	2.8	3.0
16	12.9	12.3	12.6	12.5	**13.5**	23.2	22.4	22.0	**23.3**	23.1
17	12.7	**13.6**	13.0	13.1	12.8	**23.3**	22.9	22.1	**23.3**	23.1
18	14.1	**14.6**	14.0	14.4	14.5	**23.0**	22.7	21.7	22.7	22.9
19	14.0	**14.5**	14.2	14.1	**14.5**	**22.1**	21.6	20.9	21.8	**22.1**
20	**14.8**	14.2	14.5	14.6	14.7	20.5	20.2	19.6	20.3	**20.7**
21	13.5	11.0	11.9	**14.6**	14.3	16.8	16.0	15.9	16.4	**17.0**
avg	**5.4**	**5.5**	**4.4**	**5.3**	**5.1**	**6.9**	**7.0**	**6.2**	**6.8**	**6.9**

	high level					low level				
DCP	**tim**	**ev1**	**ev2**	**ev3**	**ev4**	**tim**	**ev1**	**ev2**	**ev3**	**ev4**
0	4.5	8.6	5.0	**9.4**	7.8	1.9	**3.3**	1.9	3.4	3.2
1	6.0	9.6	5.6	**9.9**	9.6	1.7	1.6	2.0	2.1	**2.3**
2	4.0	6.5	3.8	6.8	**6.9**	1.7	**2.6**	1.7	**2.6**	3.1
3	4.0	6.0	2.9	**6.3**	6.0	1.9	**4.1**	1.7	3.2	3.4
4	2.3	**5.2**	1.8	4.4	5.1	1.5	**3.0**	1.1	2.3	2.8
5	2.7	**5.2**	1.8	4.3	4.4	1.3	2.7	0.9	2.5	**2.9**
6	2.2	4.1	1.3	3.6	**4.4**	1.2	**2.6**	1.1	2.5	**2.6**
7	1.4	2.4	0.8	**2.6**	**2.6**	1.4	**2.9**	1.1	2.2	2.0
8	2.1	**2.7**	1.0	**2.7**	2.5	1.7	**2.6**	1.4	2.0	2.2
9	2.5	2.3	2.1	**3.2**	3.0	2.0	**2.9**	1.8	2.4	2.6
10	1.9	1.4	1.3	**2.0**	1.9	2.0	**3.0**	2.0	2.5	2.8
11	**1.6**	0.2	1.4	1.3	1.2	2.6	**3.2**	2.4	3.1	3.1
12	2.0	1.0	1.7	**2.2**	0.8	3.1	**3.5**	2.6	3.1	3.2
13	**3.1**	2.8	2.8	2.6	2.7	3.2	**3.7**	2.9	3.4	3.3
14	1.6	1.5	**2.1**	1.6	1.9	3.3	**3.9**	3.2	3.4	3.8
15	1.2	1.4	**1.5**	1.4	1.3	3.1	3.3	3.0	3.3	**3.6**
16	11.9	12.6	12.0	**12.9**	12.7	19.8	23.0	20.8	22.9	**23.4**
17	12.6	12.2	**12.8**	12.7	12.6	19.9	23.0	20.7	**23.1**	**23.1**
18	12.8	**13.8**	13.2	13.7	13.5	19.5	22.3	20.3	23.1	**22.7**
19	12.6	13.4	13.3	**14.1**	**14.1**	18.8	21.6	19.7	21.7	**22.0**
20	12.5	14.4	13.4	13.0	**14.8**	17.4	**20.7**	18.1	19.7	20.6
21	7.8	10.9	10.5	12.0	**13.7**	12.7	15.9	13.8	16.3	**17.1**
avg	**5.2**	**6.3**	**5.1**	**6.5**	**6.5**	**6.4**	**8.0**	**6.6**	**7.8**	**8.0**

Table 6.3: Percentage of relative net booking forecast improvement of combination structures compared to the reference forecast. Top: parameter and level diversified forecasts, bottom: parameter diversified forecasts, left: high aggregation level, right: low aggregation level.

	high level					low level				
DCP	**tim**	**ev1**	**ev2**	**ev3**	**ev4**	**tim**	**ev1**	**ev2**	**ev3**	**ev4**
0	7.0	9.3	7.5	7.2	**11.0**	5.7	4.9	6.6	6.0	**6.7**
1	6.4	14.3	10.0	8.4	**15.4**	2.3	3.0	2.8	**3.3**	0.4
2	2.0	**12.7**	6.5	9.0	11.3	0.8	**4.6**	1.3	3.4	0.9
3	0.9	**8.8**	2.3	7.7	6.3	-0.5	1.6	-0.7	0.4	**1.8**
4	-1.5	5.5	1.2	**6.3**	5.5	-1.8	-0.5	-1.1	**1.1**	0.9
5	-2.9	**5.7**	0.1	5.4	5.4	-3.0	**0.9**	-2.1	0.3	**0.9**
6	-4.4	**5.0**	-0.4	4.2	4.1	-2.9	**0.9**	-1.8	-0.2	0.7
7	-4.7	2.4	-2.2	2.3	**3.3**	-2.5	-0.6	-2.3	-0.4	**-0.3**
8	-2.9	2.4	-1.1	2.1	**2.8**	-2.3	-0.4	-2.0	-0.5	**-0.2**
9	-1.1	3.1	-0.2	**3.4**	3.1	-1.5	-0.3	-1.0	-0.2	**0.2**
10	1.1	3.5	1.0	3.0	**3.6**	-0.7	-0.5	-0.8	-0.2	**0.3**
11	0.9	2.9	1.4	2.8	**3.8**	0.3	0.5	0.6	0.9	**1.2**
12	1.8	**3.6**	2.5	3.5	**3.6**	0.6	1.8	1.4	1.1	**1.9**
13	3.0	4.5	3.0	4.3	**5.0**	1.3	2.2	**2.3**	2.1	**2.3**
14	2.6	3.4	2.8	**4.3**	4.1	1.9	**3.3**	3.0	3.1	2.8
15	2.0	2.6	3.1	3.2	**3.9**	2.4	3.2	3.4	**3.5**	3.0
16	15.0	15.4	**16.1**	16.0	15.8	20.2	23.7	22.7	**24.2**	23.1
17	15.4	15.8	16.0	16.1	**16.4**	20.3	23.7	22.8	**24.1**	23.1
18	16.1	16.6	16.9	16.8	**17.0**	20.5	23.8	22.6	**24.4**	22.9
19	15.8	16.7	16.7	**17.0**	16.8	19.6	**23.5**	22.1	23.3	22.1
20	15.8	**17.4**	17.1	17.2	17.1	18.8	**22.4**	21.2	22.3	20.7
21	15.1	17.9	17.0	18.1	**18.5**	16.2	**19.7**	19.0	**19.7**	17.0
avg	**4.7**	**8.6**	**6.2**	**8.1**	**8.8**	**5.3**	**7.3**	**6.4**	**7.4**	**6.9**

	high level					low level				
DCP	**tim**	**ev1**	**ev2**	**ev3**	**ev4**	**tim**	**ev1**	**ev2**	**ev3**	**ev4**
0	6.0	**9.4**	6.6	9.2	8.6	4.6	**5.9**	4.6	5.6	**5.9**
1	9.5	**14.8**	8.8	12.2	12.8	3.6	**5.0**	3.6	3.3	3.7
2	7.0	**13.4**	6.9	10.9	11.8	3.2	**4.7**	2.5	3.6	3.5
3	5.6	**10.3**	4.6	9.1	10.1	2.1	**4.9**	1.7	3.4	4.2
4	4.8	**9.0**	3.9	8.9	**9.0**	1.7	**3.7**	1.0	2.4	2.6
5	3.8	**8.7**	3.0	8.2	7.2	1.0	**2.9**	0.6	2.1	2.4
6	3.6	**8.4**	2.9	6.7	8.0	1.1	**2.4**	0.7	2.3	2.2
7	3.3	5.6	2.3	5.7	**6.4**	1.1	**2.0**	0.6	1.4	1.5
8	3.1	5.6	3.2	5.3	**6.1**	1.0	**2.0**	0.9	1.5	1.8
9	4.9	6.0	4.4	5.8	**6.3**	1.5	**2.3**	1.1	1.6	2.0
10	4.3	5.1	3.7	5.0	**5.5**	1.6	**2.4**	1.1	1.9	2.0
11	4.3	4.2	3.8	4.5	**4.8**	2.3	**2.6**	1.8	2.4	2.5
12	4.9	4.7	5.0	5.3	**4.8**	2.7	**3.2**	2.3	2.7	2.8
13	5.5	5.7	5.3	5.7	**5.8**	2.9	**3.3**	2.6	3.0	3.2
14	4.3	4.5	**4.8**	4.3	4.7	3.3	**3.6**	2.9	3.3	3.5
15	3.7	4.0	3.7	**4.1**	4.0	3.3	3.5	3.0	3.4	**3.6**
16	14.6	14.8	14.9	14.7	**15.2**	20.6	23.4	21.3	23.6	**23.6**
17	15.0	**15.2**	**15.2**	14.5	14.8	20.7	**23.5**	21.4	**23.5**	**23.5**
18	15.7	15.6	**15.9**	15.4	15.5	20.8	23.5	21.4	23.5	**23.8**
19	15.6	15.9	15.7	15.8	**16.1**	20.2	22.9	20.7	23.0	**23.2**
20	16.5	**16.8**	16.3	16.5	16.5	19.6	**22.3**	19.7	21.9	22.2
21	15.6	17.3	15.5	**17.9**	17.3	17.4	**19.9**	16.7	19.7	19.4
avg	**7.8**	**9.8**	**7.6**	**9.3**	**9.6**	**7.1**	**8.6**	**6.9**	**8.1**	**8.3**

Table 6.4: Percentage of relative net booking forecast improvement of combination structures compared to the reference forecast using the improved booking forecast. Top: parameter and level diversified forecasts, bottom: parameter diversified forecasts, left: high aggregation level, right: low aggregation level.

parameters, 2x2 levels), whereas in this work, fewer of them are considered (4 methods, 3 parameters, 2 levels). Together with the fact that the maximum number of combination levels has been reduced, overfitting to the training data due to too complex structures has most probably become less of an issue. Furthermore, generating additional cancellation forecasts by diversification will have the effects on the error components described above, however, subtracting them from the booking forecast to generate net booking forecasts before evolving combination structures can induce unanticipated side effects, so that more flexible combination structures as in ev1 turn out to be more successful.

6.5.5 Analysis of generated structures

This section will have a closer look at the generated structures to investigate why they were able to outperform flat combinations. All the numbers given are obtained by taking the example of one flight of the data set, as the characteristics examined were similar for the other flights as well. The experiment investigated is the last one presented in the previous section, using a set of parameter and level diversified individual forecasts and the improved booking forecast.

First of all, it is interesting to look at a few sample structures that were generated. In the figures included later, individual forecasts are described using certain abbreviations with mappings given in Table 6.5.

Method	Param.	Level	code	Method	Param.	Level	code
0 (exp)	0	0 (low)	m0p0l0	2 (regr)	0	0 (low)	m2p0l0
0 (exp)	0	1 (high)	m0p0l1	2 (regr)	0	1 (high)	m2p0l1
0 (exp)	1	0 (low)	m0p1l0	2 (regr)	1	0 (low)	m2p1l0
0 (exp)	1	1 (high)	m0p1l1	2 (regr)	1	1 (high)	m2p1l1
0 (exp)	2	0 (low)	m0p2l0	2 (regr)	2	0 (low)	m2p2l0
0 (exp)	2	1 (high)	m0p2l1	2 (regr)	2	1 (high)	m2p2l1
1 (brown)	0	0 (low)	m1p0l0	3 (prob)	0	0 (low)	m3p0l0
1 (brown)	0	1 (high)	m1p0l1	3 (prob)	0	1 (high)	m3p0l1
1 (brown)	1	0 (low)	m1p1l0	3 (prob)	1	0 (low)	m3p1l0
1 (brown)	1	1 (high)	m1p1l1	3 (prob)	1	1 (high)	m3p1l1
1 (brown)	2	0 (low)	m1p2l0	3 (prob)	2	0 (low)	m3p2l0
1 (brown)	2	1 (high)	m1p2l1	3 (prob)	2	1 (high)	m3p2l1

Table 6.5: Mapping of forecast representations in the figures to the actual forecast generation

Figure 6.7 shows a sample structure generated by the ev1 algorithm. On the first combination level to the left of the picture, two to four individual forecasts are grouped into clusters. Eleven of the 23 available forecasts have been used as inputs, with six of them being used twice. The combination models in the second and third level of the combination vary as they were dynamically evolved. Many structures similar to this one were evolved with the ev1 algorithm, but a big number of them were also bigger and more complicated than the example given here.

Figure 6.8 shows a combination structure evolved by the ev4 algorithm. On the first level, all 24 input forecasts differing only in the level dimension are pooled. The pooling process then aggregates this dimension as described in Section 6.5.1 and produces 12 output forecasts that can be interpreted as having been reduced by the level dimension. The next dimension to be aggregated is the parameter dimension

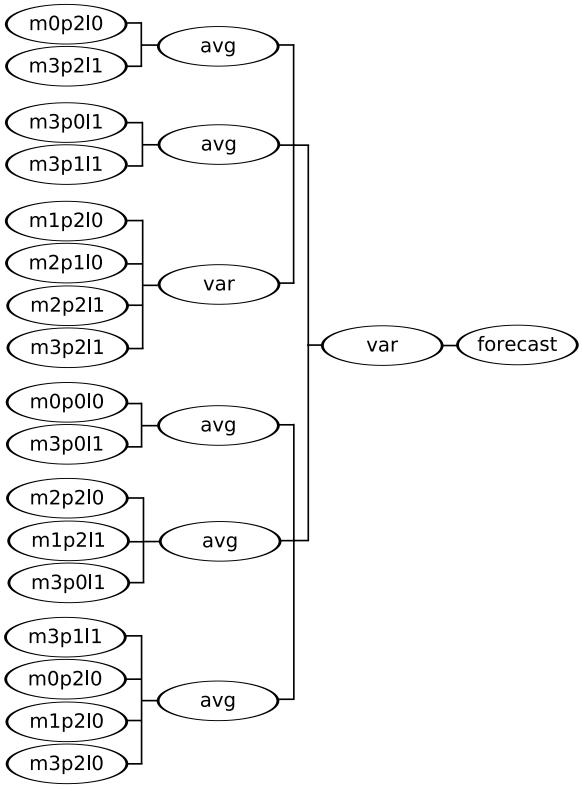

Figure 6.7: Sample combination structure generated by the ev1 algorithm

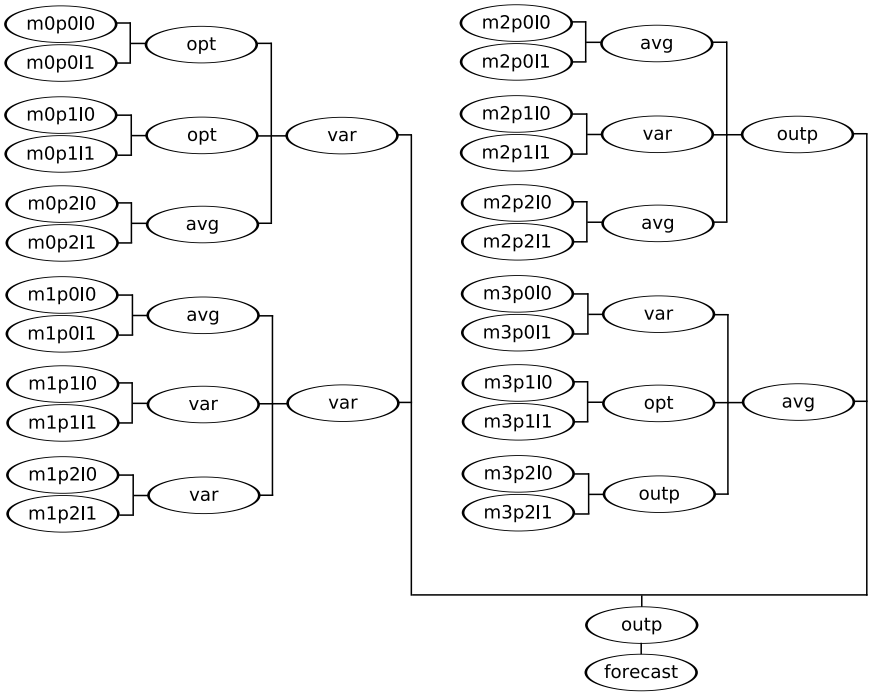

Figure 6.8: Sample combination structure generated by the ev4 algorithm

before moving on to the method dimension. As a side effect, this structure illustrates why evolving or not evolving the trimming parameter does not make a big difference in the results, as the number of individual methods per pool will always be quite small with two to four, resulting from the size of the dimensions.

It is now interesting to investigate a number of facts about the generated structures:

What is the predominant combination method used for the pools? Table 6.6 shows the percentages each of the available combination methods was used for the pools, for both the ev1 and ev4 algorithm. No predominant method can be identified, although it can be seen that the optimal model is slightly more often present in the final evolved structure. This shows that problems encountered with the optimal model in flat combinations can possibly be reduced by pooling, as the number of forecasts in each pool can be considerably smaller and the covariance estimates are less error-prone.

Method	ev1	ev4
avg	24%	24%
outp	24%	25%
var	24%	24%
opt	27%	26%

Table 6.6: Percentage of times a particular combination method is present in the final evolved structure in ev1 and ev4

Are there input forecasts that get selected more often than others? Table 6.7 shows the percentage of times an individual forecasts was selected as an input variable for a pool in the final evolved structure in the ev1 algorithm. Again, no obvious trend can be seen, as the numbers all range between 4.0 and 4.5%.

code	% chosen	code	% chosen
m0p0l0	4.1	m2p0l0	4.1
m0p0l1	4.1	m2p0l1	4.1
m0p1l0	4.2	m2p1l0	4.2
m0p1l1	4.1	m2p1l1	4.2
m0p2l0	4.2	m2p2l0	4.3
m0p2l1	4.1	m2p2l1	4.2
m1p0l0	4.1	m3p0l0	4.5
m1p0l1	4.1	m3p0l1	4.3
m1p1l0	4.2	m3p1l0	4.2
m1p1l1	4.1	m3p1l1	4.1
m1p2l0	4.3	m3p2l0	4.3
m1p2l1	4.0	m3p2l1	4.2

Table 6.7: Percentage of times an individual forecast is present in the final structure in ev1

Is there a typical order of dimensions that has been evolved for the dimension-specific pooling? In the ev4 algorithm, the order of the dimensions for

the multi-step pooling process was evolved. Table 6.8 shows the percentage a particular aggregation dimension is picked in relation to a combination level. It can be seen that the parameter dimension is a little bit less likely to be the first aggregation dimension, however, results in general are quite similar.

Dimension	1st level	2nd level	3rd level
Method	35%	31%	33%
Parameter	29%	35%	35%
Level	35%	32%	32%

Table 6.8: Percentage of times a particular aggregation dimension is selected for a combination level in ev4

Which degree of trimming is used most often? The maximum number of forecasts allowed per cluster has been evolved for the ev1 algorithm. With numbers from one to ten being allowed, the percentage of each of the possibilities being present in the final combination structure is around 10%, again not revealing any obvious trends.

The numbers given in this section reveal mainly one fact: there is nothing to reveal. The generated structures seem random with no considerable characteristics that could be identified. However, the structures do still outperform the flat combinations of forecasts investigated earlier in this chapter. This means, that the structures were able to adapt to specific situations and to produce better performing forecasts, but were different from situation to situation, so that no visible trends could be found.

6.6 Chapter summary

The generation of additional forecasts by diversification procedures was investigated in this chapter, linking properties of the resulting forecasts to error decompositions of both the individual forecasts themselves as well as to the error components of a combination it is part of. Characteristics specific to the airline application have been discussed.

In a first set of empirical experiments, benefits from using additional forecasts generated by diversification of parameters and level of learning have been evaluated. It shows that, contrary to results of a previous project investigating the demand forecast only, combinations of parameter-diversified individual forecasts provide better results than using different levels of learning, which has been attributed to the interactions between the booking and the cancellation forecast when calculating net bookings.

The chapter continued to have a look at advanced combination strategies, motivating and justifying the need for more complex and flexible approaches in the form of combination structures for the airline application. The second set of experiments then formulated the forecasting problem as an optimisation task searching for the most suitable combination structure by using evolutionary approaches. The resulting structures have been analysed and showed very few common characteristics, which underlines their flexibility and the need for different structures for different situations, also providing another explanation for the lack of success of meta-learning on

this data set. Including forecasts diversified in their level of learning improves peak performances, but is less consistent in the improvements over all DCPs compared to structures only using forecasts diversified in their parameter sets. For parameter-diversified individual forecasts using the improved booking forecast, performances improved by up to 14.4% and 6% on the high and low level, respectively.

7 Conclusions and future work

With the abundance of time series forecasting algorithms available, solely concentrating on developing new methods, improving existing ones and conducting countless empirical studies on different data sets does not seem sufficient. This thesis investigated time series forecasting from a different point of view and made contributions to answering the question of *why* methods perform well in some situations and fail to provide reasonable predictions in others. This chapter provides a summary of the thesis; its findings, conclusions and original contributions, linking it to the research questions given in the introductory chapter. A discussion of opportunities for future research will round up this chapter and this thesis.

7.1 Summary of the chapters

In the introduction, the need for a deeper understanding of forecasting and forecast combination methods and their usage was motivated. At the same time, forecasting of airline booking, cancellation and net booking values was introduced as a practical application of great importance for airline revenue management. The next chapter provided the background of forecasting in the context of airline revenue management and a more general treatment of time series forecasting and forecast combination.

Chapter 3 then started by critically looking into the major empirical studies published in the literature. As a consequence of the main findings being very general and evidence frequently suggesting that simple methods can perform better than more sophisticated ones, our own empirical investigation was conducted and described using forecast competition data. The main objective here was to investigate a pool of off-the-shelf forecasting methods that are practically very relevant because of their relatively straightforward parameterisation and implementation. The role of forecast combination as an approach to increase complexity of the system and automatically weight different individual models has been investigated. Chapter 4 once more turned the attention to the airline application and presented baseline forecasting and forecast combination results using methods applicable to this particular problem.

Chapter 5 looked at meta-learning in the time series forecasting context and gave results on empirical investigations, both for the competition and the airline data set. Following data analyses of a more exploratory nature, possible forecast accuracy improvement by several meta-learning approaches was investigated in normal forecasting conditions.

Chapter 6 was then dedicated to an investigation of the generation of additional individual forecasts specific to the airline application and provided a discussion on the effects on the different error components and potential benefits. An empirical

study in that chapter looked at flat combinations similar to the ones used in Chapters 3 and 4, but extended the experiments to include multi-level structures that were evolved using genetic algorithms.

7.2 Findings and conclusions

In section 1.2 of the introduction, a number of research questions were formulated, which will be repeated and commented on at this point. The main general questions asked are listed below:

To what extent are expert contributions beneficial in empirical forecasting applications?

This issue has been investigated in Chapter 3. The off-the-shelf algorithms investigated there were able to outperform many of the expert contributions to the NN3/NN5 competition, which supports the conclusion that sophisticated methods do not have an edge in empirical studies. This does of course not imply that the author suggests discouraging research regarding more sophisticated methods, which is naturally crucial for a better understanding of time series and forecasting. However, in purely practical applications, simpler methods have a great chance of performing just fine.

Can adequate performance be achieved by combining simple individual predictors?

Combinations have once again proven to outperform individual predictors on average throughout the thesis. In some cases, even consistently outperforming the best individual predictor is possible for both the competition and the airline data as seen in Chapters 5 and 6. Caution is however necessary when applying combinations, as consistent outperformance has only been achieved using more advanced techniques like meta-learning in the case of the competition data or generating additional forecasts by diversification procedures and evolving combination structures in the case of the airline data.

Can situations in which a particular method works well be automatically identified and domain knowledge be exploited for improved forecasting performance?

Chapter 5 discussed meta-learning as an approach to generate domain knowledge on the performance of methods and exploit it to improve forecasting performance, which has proven to be very successful for the competition data and resulted in consistent performance gains compared to the experiments described in Chapter 3. It however failed to provide consistent improvement for the airline application due to certain characteristics of the airline data set including the higher probability of a changing data generation process due to a longer forecasting horizon, lack of stability in the low level data preventing meaningful extraction of meta-features and missing generalisability from characteristics of one flight to the other.

How can a pool of individual methods be extended, and what characteristics are necessary to increase combination accuracy?

This question has been investigated in the context of the airline application in

Chapter 6, describing the generation of additional individual forecasts by diversifying method parameters and learning reference curves on different data aggregation levels. Especially using diversified parameters was successful in increasing the accuracy of the combination results. It is however important to consider the effects that diversification has on the error components of both the individual methods and the ensembles to anticipate possible positive and negative application-specific side effects, which has also been discussed in Chapter 6.

7.3 Original contributions

Original contributions of the thesis were discussed in the introduction, but will be summarised a second time with the benefit of hindsight.

Chapter 3 did not only describe another empirical study conducted in the area of time series forecasting. It critically investigated major past empirical studies and as a result questioned the need for sophisticated forecasting methods in practical applications. Consequently, the experiments have been conducted with off-the-shelf forecasting and forecast combination techniques, underlining their competitiveness with contributions that require more expert effort for modelling and parametrisation as well as computational power. Relatively simple combinations were able to further improve results and decrease the risk of picking a badly performing model from the pool of individual predictors.

A comprehensive review of meta-learning for time series forecasting has been provided in Chapter 5. A new empirical experiment has been conducted, extending the feature and method pools of previous work. Furthermore, a ranking algorithm for meta-learning was investigated, which is an approach that has hardly been looked at previously in this context. Evidence from both literature and the experiments presented suggest that domain knowledge is one of the keys for a better understanding of the dynamics of time series forecasting performance and to provide a solution to the dilemma of the no-free-lunch theorem.

Extensive work on industrial applications of time series forecasting is rare. Due to a collaboration with Lufthansa Systems in Berlin, this work has been able to provide unique insights into the practical applicability of forecasting algorithms and current industrial practice in the Chapters 2, 4, 5 and 6. A novel probability based forecasting algorithm for airline cancellations has been introduced in Chapter 2. Chapter 6 discussed means of generating additional forecasts by diversification procedures and discussed impact on error components of different decompositions, thus helping to understand why one method might be more successful than another under certain circumstances. Flexible combination structures were evolved for the airline data set, extending previous work by evolving more parameters for the combinations. An analysis of the generated structures provided insights as to why the combination structures were successful in improving forecast accuracy.

In summary, this work gave a unique treatment of time series forecasting, contributing to a better understanding of what makes the combination of forecasts beneficial. With the transferability of latest research outcomes into industrial practices always having been a rather problematic issue in research, this thesis' main strength is looking at the research area from two points of view: on the one hand, investigating new techniques on data sets from recent forecasting competitions fitted the results into current academic research, while, on the other hand, the big focus on the airline application provided a link to practical applications in industry.

7.4 Future work

Of course there can always be a wider range of forecasting/combination/meta-learning methods, more parametrisations, more features and different data sets that can be investigated for any empirical study; the same applies to the empirical studies presented in this thesis. The future in time series forecasting according to Ord (2001) however does not lie in increasing the number of comparative empirical studies, but in gaining a better understanding of the behaviour of existing forecasting methods in different scenarios. In this respect, extending experiments on meta-learning as presented in Chapter 5 seem to be particularly promising for future investigations, especially if dealing with ranking approaches on which literature is very sparse to date.

There are many directions in which research for the airline application could be pursued. As explained in Chapter 2, the airline net booking forecasting process follows the concept of decomposition. As seen in Chapter 6, the different components interact with each other when eventually calculating the final forecast. To completely understand these interactions and exploit them for better accuracy, investigations going further than the ones presented here would be useful.

The question of aggregating forecasts to different levels certainly has potential for further research. High level forecasts are currently aggregated from forecasts obtained on the low level. This bottom-up approach is compared to different versions of a top-down approach in Athanasopoulos et al. (2009) in the context of a tourism application. Similar investigations could be carried out for airline data. Furthermore, since higher level data tends to be more stable, it might be beneficial to generate forecasts directly on the levels they are needed for.

Evolving combination structures has proven to be very successful for airline data. However, only the availability of domain knowledge and special characteristics of the data set, for example the presence of different aggregation levels, facilitated a discussion of the impacts of diversification procedures on the forecast error components. The link to more general data sets is not entirely straightforward, but its investigation would be novel and promising.

Another interesting issue is adaptivity. The forecasting process of Lufthansa Systems automatically adapts to the data by updating the reference curves of the individual algorithms as time goes by. The combination weights and the meta-learning weights were then only calculated on a training set and assessed on an out-of-sample test set, however, investigating adaptivity on the level of the combination methods would be worth looking at, as methods are likely to change their relative performance to each other as the data generation process changes. A periodic rebuild of the combination weights would be an easy option to start with.

Due to its great practical relevance and the diversity of application areas, time series forecasting has been a very active research area for about half a century now, and it is very likely to remain this way. As mentioned in several places of the thesis, research however started to shift into the direction of why methods work, moving away from trying to find the one superior algorithm. More research is to be expected in this context, with this thesis providing a contribution.

A

Description of the software

A software tool called "Avanti" has been developed in the scope of a previous collaboration project of Bournemouth University and Lufthansa Systems Berlin. Avanti is implemented in C++ and uses the Microsoft Foundation Classes (MFC) for the graphical user interface. It can be seen as a software spinoff primarily used for research, which is strongly linked to the forecasting kernel developed by Lufthansa Systems Berlin that is used in the productive systems offered to the customers.

Avanti provides the possibility to visualise and analyse results according to selected dimensions of the data. Its design has been kept modular, so that calculations are broken down to components that facilitate easier understanding and reusability. An extensive manual on how to install and use Avanti for running experiments on airline data has been provided in Riedel (2007). In this section, only additional components that were developed to allow for experiments involving cancellation forecasting will be described.

A.1 Preprocessing

A.1.1 ABS_TO_RATE and RATE_TO_ABS

Given absolute booking numbers, these two components calculate a cancellation rate given absolute cancellation numbers and vice versa. A few application-specific corrections are carried out, for example restricting the maximum number of bookings and ensuring the monotony of the absolute cancellations for a flight.

ABS_TO_RATE	
Input	bookings cancellations
Output	cancellation rate
Parameters	pCycleSize: number of DCPs before departure

RATE_TO_ABS	
Input	bookings cancellation rate
Output	absolute cancellations
Parameters	pCycleSize: number of DCPs before departure pMaxBkg: maximum number of bookings

A.1.2 CONSTRAIN_RATE

As described in Section 2.1.5.1, the actual observed cancellation rate is constrained using confidence limits. This component is used for correcting the observed cancellation rate before both the history building and the forecasting calculations.

CONSTRAIN_RATE	
Input	bookings reference cancellation rate observed cancellation rate booking forecast (if used for forecasting)
Output	corrected cancellation rate
Parameters	pCycleSize: number of DCPs before departure pCancInitialBound: initial bounds for cancellation rate pCancLowerBound: width of confidence limits pSuffBkgs: parameter for sufficient bookings pUseFc: indicating if rate will be used for forecasting or history building

A.1.3 UNCONSTRAINING_CANC

Section 2.1.4.3 explains the need for unconstraining booking and cancellation numbers due to fareclasses closing and opening because of booking control. Unconstraining for cancellations is implemented in this component. It is carried out by calculating the weighted sum of the cancellation reference curve applied to the constrained bookings and the default cancellation rate applied to the estimated denied bookings.

UNCONSTRAINING_CANC	
Input	observed cancellation rate reference cancellation rate bookings denied bookings
Output	unconstrained cancellation rate

A.2 Data analysis

A.2.1 DEFAULT_PROB

Default probabilities for the new individual forecasting method are calculated in this component. Data from the initialisation period is used for this purpose. The algorithm is equivalent to the one used for the ordinary history building, which was described in Section 2.1.5.3, with the difference that the distribution of an occurring cancellation to bookings at previous DCPs is not weighted, but assumes an equal probability for the cancellation belonging to any of the previous bookings.

DEFAULT_PROB	
Input	bookings cancellations
Output	default probability
Parameter	pCycleSize: number of DCPs before departure pHistCycles: length of initialisation period

A.2.2 DATA_ANALYSE

This component is used for the exploratory airline data analysis presented in Section 5.4.1.

DATA_ANALYSE	
Input	bookings cancellations availability cancellation reference curve (exponential smoothing) default references
Output	summary data features
Parameter	pCycleSize: number of DCPs before departure pSplit: indicates position of split between training and test data pChunks: denotes block size of the signals provided, which are given by their level of aggregation

A.2.3 FEATURES

For meta learning, a number of features need to be extracted from the data set, which is carried out in this component. The features are described in Section 5.4.3.

FEATURES	
Input	bookings cancellations availability cancellation reference curve (exponential smoothing) cancellation forecast (exponential smoothing) block element shift
Output	data features
Parameter	pCycleSize: number of DCPs before departure

A.3 History building

A.3.1 HB_SMCANC

Two traditional ways of generating cancellation rate reference curves are using exponential smoothing and Brown's smoothing approach, which were described in Section

2.1.5.2. These are implemented in the component HB_SMCANC. Initially, the rate is set to a default cancellation rate provided by Lufthansa Systems, before observed cancellation rates are used to update it.

HB_SMCANC	
Input	bookings cancellations default cancellation rate
Output	cancellation rate reference curve trend for the reference curve
Parameters	pMethod: exponential or Brown's smoothing pSmoothingFactor: smoothing factor used pCycleSize: number of DCPs before departure pHistCycles: number of cycles in initialisation period pTrendMin: trend lower bound pTrendMax: trend upper bound pCancInitialBound: values of initial confidence limits pCancLowerBound: width of confidence limits pSuffBkgs: number of bookings that indicate sufficient information to relax confidence limits pLearningInfluence: determines the extent of the influence of changes at later DCPs has on earlier DCPs for the reference curve

A.3.2 HB_REGRCANC

Updating the reference curves using a regression approach, also described in Section 2.1.5.2, is implemented in this component.

HB_REGRCANC	
Input	bookings cancellations default cancellation rate
Output	cancellation rate reference curve trend for the reference curve
Parameter	pCycleSize: number of DCPs before departure pHistCycles: number of cycles in initialisation period pTrendMin: trend lower bound pTrendMax: trend upper bound

A.3.3 HB_PROBCANC

History building for the new probability forecast introduced in Section 2.1.5.3 is realised in this component. Cancellation probabilities are initialised using default probabilities estimated in a seperate component. Because the data cubes holding probability references tend to be too big, only data after a certain learning period is provided in the output.

HB_PROBCANC	
Input	bookings cancellations default probability
Output	probability references
Parameter	pSmoothing: smoothing factor used for reference curve update pInitProb: fixed probability used if no default probability curves are given pCycleSize: number of DCPs before departure pHistCycles: number of cycles in initialisation period pLearn: length of period not used for forecasting (used to limit size of the output reference) pLearningInfluence: determines the extent of the influence of changes at later DCPs has on earlier DCPs for the reference curve

A.4 Forecasting

A.4.1 FC_CANC

This component calculates forecasts based on reference curves as explained in Section 2.1.5.2, regardless if the curves were generated with the exponential smoothing, the Brown's smoothing method or the regression approach. The block element shift input signal denotes the number of time series "intervals" between one block element and the last block element, which is necessary due to the fact that the DCPs have different distances to the departure date as shown in Table 2.1. Providing these values as an input parameter is necessary to ensure that only values learnt before the time of forecast generation are used.

FC_CANC	
Input	bookings observed cancellation rate cancellation rate reference curve cancellation rate reference curve trend booking forecast block element shift
Output	cancellation rate forecast
Parameter	pUseTrend: indicates availability of trend information pDampTrend: indicates whether or not trend should be dampened pCycleSize: number of DCPs before departure

A.4.2 FC_PROBCANC

The probability forecast using the probabilty reference curves is implemented in this component.

FC_PROBCANC	
Input	bookings booking reference curve booking forecast observed cancellation rate cancellation probability reference curve cancellation forecast exponential smoothing block element shift
Output	cancellation rate forecast or net booking forecast
Parameter	pNetBkg: indicates whether to calculate the net booking or cancellation rate forecast

A.4.3 FC_META

This component is needed for the meta-learning experiments described in Section 5.4. The machine learning algorithms to build the meta-knowledge are implemented in matlab called using the C++ matlab engine. Three different ways of trimming are possible: providing the maximum number or the perventage of individual forecasts to consider or a variance ratio with regard to the best individual forecast that cannot be exceeded.

FC_META	
Input	data features forecast errors on training set individual forecasts
Output	cancellation rate forecast or net booking forecast
Parameter	pTrimmingMaxNbrFc: maximum number of individual forecasts to consider pTrimmingPerc: percentage of individual forecasts two consider pTrimmingMaxVarRatio: maximum variance ratio to best individual forecast pMethod: indicates which method to use: decision trees (0), neural networks (1), support vector machines (2) or ranking (3) pFeatures: number of features pSplit: calendar week at which the feature set is split into a training and a testing period pEnd: total number of feature records

B

Airline data experiments

Experiments on airline data analysed in this thesis have been carried out using the previously described Avanti software. Each of the experiments involve a big number of components which will not all be listed here. However, to provide an impression of the experiments, components and their parameters will be listed for the example of evaluating individual forecasting methods and simple combination approaches as presented in Chapter 3. Calculations related to the demand forecast, including decomposing data, estimating seasonality and calculating the booking forecast according to Experiment2 in Riedel (2007), will not be included.

FILE_INTERFACE	
Purpose	initial loading of data
Parameter	pCubesToLoad: bookings, cancellations, availablity pCubesToSave: none pAppliedDimInFile: DCP
FILE_INTERFACE	
Purpose	initial loading of auxilliary data
Parameter	pCubesToLoad: default cancellation rate, default probabilities, block Element shift pCubesToSave: none pAppliedDimInFile: none
ABS_TO_RATE	
Purpose	calculating constrained cancellation rate from bookings and cancellations using default reference
Input	bookings cancellations default cancellation rate
Output	constrained cancellation rate
Parameter	pCycleSize: number of DCPs (23)
HB_SMCANC	
Purpose	calculating a first estimate of the cancellation rate reference curve using constrained data
Input	bookings cancellation rate default cancellation rate
Output	reference cancellation rate reference cancellation rate trend (unused)
Parameter	pMethod: 0 (exponential smoothing) pSmoothingFactor: 0.1 pCycleSize: 23

	pHistCycles: 51 pMinTrend: unused pMaxTrend: unused pCancInitialBound: 0.3 pCancLowerBound: 0.05 pSuffBkgs: 15 pLearningInfluence: 0.5
ABS_TO_RATE	
Purpose	calculating constrained cancellation rate from bookings and cancellations using first reference curve estimate
Input	bookings cancellations reference cancellation rate
Output	constrained cancellation rate
Parameter	pCycleSize: 23
UNCONSTRAINING_CANC	
Purpose	unconstraining cancellation rate
Input	constrained cancellation rate reference cancellation rate constrained bookings rejected bookings
Output	unconstrained cancellation rate
RATE_TO_ABS	
Purpose	calculate absolute unconstrained cancellations for performance evaluation
Input	unconstrained bookings unconstrained cancellation rate
Output	unconstrained absolute cancellations
Parameter	pCycleSize: 23
Parameter	pMaxBkg: 2000
DIFFERENCE	
Purpose	calculate absolute unconstrained net bookings for performance evaluation
Input	unconstrained bookings unconstrained absolute cancellations
Output	unconstrained net bookings
HB_SMCANC	
Purpose	calculate the final estimate of the cancellation rate reference curve - exponential smoothing
Input	unconstrained bookings unconstrained cancellation rate default cancellation rate
Output	reference cancellation rate (exp) reference cancellation rate trend (unused)
Parameter	pMethod: 0 pSmoothingFactor: 0.1 pCycleSize: 23 pHistCycles: 51

	pMinTrend: unused pMaxTrend: unused pCancInitialBound: 0.3 pCancLowerBound: 0.05 pSuffBkgs: 15 pLearningInfluence: 0.5
HB_SMCANC	
Purpose	calculate the final estimate of the cancellation rate reference curve - Brown's smoothing
Input	unconstrained bookings unconstrained cancellation rate default cancellation rate
Output	reference cancellation rate (Brown) reference cancellation rate trend (Brown)
Parameter	pMethod: 1 pSmoothingFactor: 0.1 pCycleSize: 23 pHistCycles: 51 pMinTrend: -0.1 pMaxTrend: 0.1 pCancInitialBound: 0.3 pCancLowerBound: 0.05 pSuffBkgs: 15 pLearningInfluence: 0.5
HB_REGRCANC	
Purpose	calculate the final estimate of the cancellation rate reference curve - regression
Input	unconstrained bookings unconstrained cancellation rate default cancellation rate
Output	reference cancellation rate (regr) reference cancellation rate trend (regr)
Parameter	pCycleSize: 23 pHistCycles: 51 pMinTrend: -0.1 pMaxTrend: 0.1 pCancInitialBound: 0.3 pCancLowerBound: 0.05 pSuffBkgs: 15 pLearningInfluence: 0.5
HB_PROBCANC	
Purpose	calculate probability reference curves
Input	unconstrained bookings unconstrained cancellation rate default probabilities
Output	cancellation probability reference curve
Parameter	pSmoothingFactor: 0.1 pInitProb: unused

	pCycleSize: 23 pHistCycles: 51 pLearningInfluence: 0.9
CONSTRAIN_RATE	
Purpose	Apply confidence limits to the observed rate
Input	unconstrained bookings reference cancellation rate (exp) unconstrained cancellation rate booking forecast
Output	unconstrained corrected cancellation rate
Parameter	pCycleSize: 23 pCancInitialBound: 0.3 pCancLowerBound: 0.05 pSuffBkgs: 15 pUseFc: 1 (yes)
FC_CANC	
Purpose	Generate forecasts using the exponential smoothing (Brown, regression) reference curves
Input	unconstrained bookings unconstrained corrected cancellation rate cancellation rate reference curve exponential smoothing (Brown, regression) trend of cancellation rate reference curve exponential smoothing (Brown, regression) booking forecast block element shift
Output	cancellation rate forecast
Parameter	pCycleSize: 23 pUseTrend: 0 (1,1) pDampTrend: 0 (1,0)
FC_PROBCANC	
Purpose	Generate probability forecast
Input	unconstrained bookings booking reference curve booking forecast unconstrained corrected cancellation rate cancellation probability reference curve cancellation rate forecast (exponential smoothing) block element shift
Output	cancellation rate forecast
Parameter	pNetBkg: 1 (cancellation rate) pCycleSize: 23
RATE_TO_ABS	
Purpose	calculate absolute cancellation forecasts
Input	booking forecast cancellation rate forecasts
Output	absolute cancellations forecasts
Parameter	pCycleSize: 23

Parameter	pMaxBkg: 2000
DIFFERENCE	
Purpose	calculate net booking forecasts
Input	booking forecast absolute cancellations forecast
Output	net booking forecast
RATE_TO_ABS	
Purpose	calculate absolute cancellation forecasts
Input	booking forecast cancellation rate forecasts
Output	absolute cancellations forecasts
Parameter	pCycleSize: 23
Parameter	pMaxBkg: 2000
DIFFERENCE	
Purpose	calculate net booking forecasts
Input	booking forecast absolute cancellations forecast
Output	net booking forecast
VALID_FC_REF	
Purpose	calculate relative forecast error
Input	unconstrained net bookings net booking forecast
Output	net booking error
ERROR_COVAR	
Purpose	calculate absolute forecast error
Input	net booking error
Output	absolute net booking error
HB_LINEAR_COMBINATION	
Purpose	Calculate linear combination weights
Input	net booking error unconstrained net bookings
Output	linear combination weights
Parameter	pMethod: 0- simple average (1- outperformance, 2- variance-based, 3- restricted regression, 4- unrestricted regression) pTrimmingMaxNbrFc: -1 (unused) pTrimmingPerc: -1 (unused) pTrimmingMaxVarRatio: -1 (unused)
LINEAR_COMBINATION	
Purpose	Calculate combination forecast
Input	net booking forecast linear combination weights
Output	combined net booking forecast
VALID_FC_REF	
Purpose	calculate relative combination forecast error
Input	unconstrained net bookings combined net booking forecast
Output	combined net booking error

ERROR_COVAR	
Purpose	calculate absolute combination forecast error
Input	combined net booking error
Output	combined absolute net booking error
FILE_INTERFACE	
Purpose	saving error values for individual and combined methods
Parameter	pCubesToLoad: none pCubesToSave: individual and combined net booking error pAppliedDimInFile: DCPFC (forecast DCP)

References

Abraham, B. & Ledolter, J. (1986), 'Forecast functions implied by autoregressive integrated moving average models and other related forecast procedures', *International Statistical Review* **54**(1), 51–66.

Adya, M., Armstrong, J., Collopy, F. & Kennedy, M. (2000), 'An application of rule-based forecasting to a situation lacking domain knowledge', *International Journal of Forecasting* **16**(4), 477–484.

Adya, M., Collopy, F., Armstrong, J. & Kennedy, M. (2001), 'Automatic identification of time series features for rule-based forecasting', *International Journal of Forecasting* **17**(2), 143–157.

Aiolfi, M. & Timmermann, A. (2006), 'Persistence in forecasting performance and conditional combination strategies', *Journal of Econometrics* **127**(1-2), 31–53.

Akaike, H. (1973), Information theory and an extension of the maximum likelihood principle, *in* B. Petrov & F. Caski, eds, 'In Proceedings of the 2nd International Symposium on Information Theory', Akademiai Kiado, pp. 267–281.

Anastasakis, L. & Mort, N. (2009), 'Exchange rate forecasting using a combined parametric and nonparametric self-organising modelling approach', *Expert Systems with Applications* **36**(10), 12001–12011.

Arinze, B., Kim, S.-L. & Anandarajan, M. (1997), 'Combining and selecting forecasting models using rule based induction', *Computers & Operations Research* **24**(5), 423–433.

Assimakopoulos, V. & Nikolopoulos, K. (2000), 'The theta model: A decomposition approach to forecasting', *International Journal of Forecasting* **16**(4), 521–30.

Athanasopoulos, G., Ahmed, R. A. & Hyndman, R. J. (2009), 'Hierarchical forecasts for australian domestic tourism', *International Journal of Forecasting* **25**(1), 146–166.

Batchelor, R. & Dua, P. (1995), 'Forecaster diversity and the benefits of combining forecasts', *Management Science* **41**(1), 68–75.

Bates, J. & Granger, C. (1969), 'The combination of forecasts', *Operational Research* **20**(4), 451–468.

Belobaba, P. (1987), Air Travel Demand and Airline Seat Inventory Management, PhD thesis, Flight Transportation Laboratory, Massachusetts Institute of Technology.

Bitran, G. & Caldentey, R. (2003), 'An overview of pricing models for revenue management', *Manufacturing and Service Operations Management* **5**, 1407–1420.

Box, G. & Jenkins, G. (1970), *Time Series Analysis*, Holden-Day, San Francisco.

Brazdil, P., Soares, C. & Pinto de Costa, J. (2003), 'Ranking learning algorithms: Using ibl and meta-learning on accuracy and time results', *Machine Learning* **50**(3), 251–277.

Breiman, L. (1984), *Classification and Regression Trees*, Chapman & Hall.

Breiman, L. (1996), 'Bagging predictors', *Machine Learning* **24**(2), 123–140.

Brown, G., Wyatt, J., Harris, R. & Yao, X. (2005), 'Diversity creation methods: a survey and categorisation', *Journal of Information Fusion* **6**, 5–20.

Brown, G., Wyatt, J. & Tino, P. (2005), 'Managing diversity in regression ensembles', *The Journal of Machine Learning Research* **6**, 1621–1650.

Brown, R. G., Meyer, R. F. & D'Esopo, D. A. (1961), 'The fundamental theorem of exponential smoothing', *Operations Research* **9**(5), 673–687.

Bunn, D. (1975), 'A bayesian approach to the linear combination of forecasts', *Operational Research Quarterly* **26**(2), 325–329.

Cai, X., Zhang, N., Venayagamoorthy, G. K. & II, D. C. W. (2007), 'Time series prediction with recurrent neural networks trained by a hybrid PSO-EA algorithm', *Neurocomputing* **70**(13-15), 2342–2353.

Chatfield, C. (1995), 'Positive or negative?', *International Journal of Forecasting* **11**(4), 501–502.

Chatfield, C., Koehler, A. B., Ord, J. K. & Snyder, R. D. (2001), 'A new look at models for exponential smoothing', *Journal of the Royal Statistical Society. Series D (The Statistician)* **50**(2), 147–159.

Chiang, W.-C., C.H. Chen, J. & Xu, X. (2007), 'An overview of research on revenue management: current issues and future research', *International Journal Revenue Management* **1**(1), 97–128.

Clemen, R. (1989), 'Combining forecasts: A review and annotated bibliography', *International Journal of Forecasting* **5**, 559–583.

Clements, M. & Smith, J. (1997), 'The performance of alternative forecasting methods for SETAR models', *International Journal of Forecasting* **13**(4), 463–475.

Collopy, F. & Armstrong, S. J. (1992), 'Rule-based forecasting: Development and validation of an expert systems approach to combining time series extrapolations', *Management Science* **38**(10), 1394–1414.

Cox Jr, L. & Popken, D. (2002), 'A hybrid system-identification method for forecasting telecommunications product demands', *International Journal of Forecasting* **18**(4), 647–671.

Crone, S. (2006/2007), 'NN3 Forecasting Competition [Online]'. Available online: http://www.neural-forecasting-competition.com/NN3/ [02/06/2009].

Crone, S. (2008), 'NN5 Forecasting Competition [Online]'. Available online: http://www.neural-forecasting-competition.com/NN5/ [02/06/2009].

de Menezes, L. M., Bunn, D. W. & Taylor, J. W. (2000), 'Review of guidelines for the use of combined forecasts', *European Journal of Operational Research* **120**(1), 190–204.

Delft Center for Systems and Control (2007), 'Matlab toolbox ARMASA [Online]'. http://www.dcsc.tudelft.nl/Research/Software [13/06/2007].

Deutsch, M., Granger, C. W. J. & Teräsvirta, T. (1994), 'The combination of forecasts using changing weights', *International Journal of Forecasting* **10**(1), 47–57.

Diebold, F. (1988), 'Serial correlation and the combination of forecasts', *Journal of Business & Economic Statistics* **6**(1), 105–111.

Diebold, F. X. & Pauly, P. (1986), Structural change and the combination of forecasts, Special Studies Papers 201, Board of Governors of the Federal Reserve System (U.S.). available at http://ideas.repec.org/p/fip/fedgsp/201.html.

Dietterich, T. (2000), Ensemble methods in machine learning, *in* 'Proceedings of the First International Workshop on Multiple Classifier Systems', pp. 1–15.

Donaldson, R. G. & Kamstra, M. (1999), 'Neural network forecast combining with interaction effects', *Journal of The Franklin Institute* **336**(2), 227–236.

Donaldson, R. & Kamstra, M. (1996), 'Forecast combining with neural networks', *Journal of Forecasting* **15**(1), 49–61.

Durbin, J. & Koopman, S. (2001), *Time series analysis by state space methods*, Oxford University Press.

Elliott, G. & Timmermann, A. (2005), 'Optimal forecast combination under regime switching', *International Economic Review* **46**(4), 1081–1102.

Fang, Y. (2003), 'Forecasting combination and encompassing tests', *International Journal of Forecasting* **19**(1), 87–94.

Fiordaliso, A. (1998), 'A nonlinear forecasts combination method based on Takagi-Sugeno fuzzy systems', *International Journal of Forecasting* **14**(3), 367–379.

Fodor, I. K. (2002), A survey of dimension reduction techniques, Technical report, Center for Applied Scientific Computing, Lawrence Livermore National Laboratory.

Fok, D., van Dijk, D. & Franses, P. H. (2005), 'Forecasting aggregates using panels of nonlinear time series', *International Journal of Forecasting* **21**(4), 785–794.

Frantti, T. & Mähönen, P. (2001), 'Fuzzy logic-based forecasting model', *Engineering Applications of Artificial Intelligence* **14**(2), 189–201.

Freund, Y. & Schapire, R. E. (1997), 'A decision-theoretic generalization of on-line learning and an application to boosting', *Journal of Computer and System Sciences* **55**(1), 119–139.

Gabrys, B. & Ruta, D. (2005), 'Classifier selection for majority voting', *Information Fusion* **6**(1), 63–81.

Gardner, E. S. (1985), 'Exponential smoothing: The state of the art', *Journal of Forecasting* **4**(1), 1–28.

Gardner, E. S. (2006), 'Exponential smoothing: The state of the art–part ii', *International Journal of Forecasting* **22**(4), 637–666.

Gautama, T., Mandic, D. & Van Hulle, M. (2004), 'A novel method for determining the nature of time series', *IEEE Transactions on Biomedical Engineering* **51**, 728–736.

Geman, S., Bienenstock, E. & Doursat, R. (1992), 'Neural networks and the bias/variance dilemma', *Neural Computation* **4**(1), 1–58.

Goldberg, D. & Holland, J. (1988), 'Genetic algorithms and machine learning', *Machine Learning* **3**(2-3), 95–99.

Gooijer, J. G. D. & Hyndman, R. J. (2006), '25 years of time series forecasting', *International Journal of Forecasting* **22**(3), 443–473.

Granger, C. (1989), 'Invited review: Combining forecasts - twenty years later', *Journal of Forecasting* **8**, 167–173.

Granger, C. & Jeon, Y. (2004), 'Thick modeling', *Economic Modelling* **21**(2), 323–343.

Granger, C. & Ramanathan, R. (1984), 'Improved methods of combining forecasts', *Journal of Forecasting* **3**(2), 197–204.

Gyorfi, L., Lugosi, G. & Udina, F. (2006), 'Nonparametric kernel-based sequential investment strategies', *Mathematical Finance* **16**(2), 337–357.

Hall, M. A. (1998), Correlation-based Feature Subset Selection for Machine Learning, PhD thesis, University of Waikato, Hamilton, New Zealand.

Hansen, J. V. (2000), Combining Predictors: Meta Machine Learning Methods and Bias/Variance & Ambiguity Decompositions, PhD thesis, Department of Computer Science, University of Aarhus.

Harrald, P. & Kamstra, M. (1997), 'Evolving artificial neural networks to combine financial forecasts', *IEEE Transactions on Evolutionary Computation* **1**, 40–52.

Harvey, A. (2006), Forecasting with unobserved components time series models, *in* G. Elliott, C. Granger & A. Timmermann, eds, 'Handbook of Economic Forecasting', Elsevier, pp. 327–408.

He, C. & Xu, X. (2005), 'Combination of forecasts using self-organizing algorithms', *Journal of Forecasting* **24**, 269–278.

Hendry, D. & Clements, M. (2002), 'Pooling of forecasts', *Econometrics Journal* **5**, 1–26.

Hibon, M. & Evgeniou, T. (2005), 'To combine or not to combine: selecting among forecasts and their combinations', *International Journal of Forecasting* **21**(1), 15–24.

Hill, T., O'Connor, M. & Remus, W. (1996), 'Neural network models for time series forecasts', *Management Science* **42**(7), 1082–1092.

Hippert, H., Bunn, D. & Souza, R. (2005), 'Large neural networks for electricity load forecasting: Are they overfitted?', *International Journal of Forecasting* **21**(3), 425–434.

Hippert, H., Pedreira, C. & Souza, R. (2001), 'Neural networks for short-term load forecasting: a review and evaluation', *IEEE Transactions on Power Systems* **16**(1), 44–55.

Hu, M. & Tsoukalas, C. (1999), 'Combining conditional volatility forecasts using neural networks: an application to the ems exchange rates', *Journal of International Financial Markets, Institutions & Money* **9**(4), 407–422.

Huang, H. & Lee, T.-H. (2007), To combine forecasts or to combine information?, Working papers, University of California at Riverside, Department of Economics.

Hyndman, R. & Billah, B. (2003), 'Unmasking the Theta method', *International Journal of Forecasting* **19**(2), 287–290.

Hyndman, R. J. (2001), 'It's time to move from what to why', *International Journal of Forecasting* **17**, 567–570.

Hyndman, R. J., Koehler, A. B., Snyder, R. D. & Grose, S. (2002), 'A state space framework for automatic forecasting using exponential smoothing methods', *International Journal of Forecasting* **18**(3), 439–454.

Ivakhnenko, A. (1970), 'Heuristic self-organization in problems of engineering cybernetics', *Automatica* **6**(2), 207–219.

Jain, C. L. (2008), 'Benchmarking forecasting models', *Journal of Business Forecasting* **26**, 15–35.

Jolliffe, I. (2002), *Principal component analysis 2nd edition*, Springer.

Jose, V. R. R. & Winkler, R. L. (2008), 'Simple robust averages of forecasts: Some empirical results', *International Journal of Forecasting* **24**(1), 163–169.

Kalman, R. (1960), 'A new approach to linear filtering and prediction problems', *Journal of Basic Engineering* **82**(1), 35–45.

Kalousis, A. & Theoharis, T. (1999), 'NOEMON: design, implementaion and performance results of an intelligent assistant for classifier selection', *Intelligent Data Analysis* **5**(3), 319–337.

Kisinbay, T. (2007), The use of encompassing tests for forecast combinations, Technical report, International Monetary Fund Working Paper N. 07/264.

Kodogiannis, V. & Lolis, A. (2002), 'Forecasting financial time series using neural network and fuzzy system-based techniques', *Neural Computing and Applications* **11**(2), 90–102.

Koutroumanidis, T., Ioannou, K. & Arabatzis, G. (2009), 'Predicting fuelwood prices in Greece with the use of ARIMA models, artificial neural networks and a hybrid ARIMA-ANN model', *Energy Policy* **37**(9), 3627–3634.

Koza, J. R. (1992), *Genetic Programming*, MIT Press.

Krogh, A. & Vedelsby, J. (1995), 'Neural network ensembles, cross-validation and active learning', *Advances in Neural Information Processing Systems* **7**, 231–238.

Kuncheva, L. (2004), *Combining pattern classifiers: methods and algorithms*, John Wiley & Sons.

Kuncheva, L. I. & Whitaker, C. J. (2003), 'Measures of diversity in classifier ensembles', *Machine Learning* **51**(2), 181–207.

Lemke, C. & Gabrys, B. (2007), Review of nature-inspired forecast combination techniques, *in* 'NiSIS 2007 Symposium'.

Lemke, C. & Gabrys, B. (2008*a*), Do we need experts for time series forecasting?, *in* 'Proceedings of the 16th European Symposium on Artificial Neural Networks', pp. 253–258.

Lemke, C. & Gabrys, B. (2008*b*), On the benefit of using time series features for choosing a forecasting method, *in* 'Proceedings of the European Symposium on Time Series Prediction', pp. 1–10.

Lemke, C. & Gabrys, B. (2009), 'Meta-learning for time series forecasting and forecast combination', *accepted to a special issue of Neurocomputing* .

Lemke, C., Riedel, S. & Gabrys, B. (2009), Dynamic combination of forecasts generated by diversification procedures applied to forecasting of airline cancellations, *in* 'Proceedings of the IEEE Symposium Series on Computational Intelligence', pp. 85–91.

Li, F. & Tkacz, G. (2001), Evaluating linear and non-linear time-varying forecast-combination methods, Technical report, Bank of Canada.

Littlewood, K. (1972), 'Forecasting and control of passenger bookings', *AGIFORS Symposium Proceedings* **12**, 95–117.

Liu, Y. (2005), Value-at-risk model combination using artificial neural networks, Technical report, Emory University Working Paper Series.

Liu, Y. & Yao, X. (1999), 'Ensemble learning via negative correlation', *Neural Networks* **12**(10), 1399–1404.

Maforte dos Santos, P., Ludermir, T. & Cavalcante, R. (2004), Selection of time series forecasting models based on performance information, *in* 'Proceedings of the Fourth International Conference on Hybrid Intelligent Systems', pp. 366–371.

Makridakis, S., Andersen, A., Carbone, R., Fildes, R., Hibon, M., Lewandowski, R., Newton, J., Parzen, E. & Winkler, R. (1982), 'The accuracy of extrapolative (time series) methods: The results of a forecasting competition.', *Journal of forecasting* **1**, 111–153.

Makridakis, S., Chatfield, C., Hibon, M., Lawrence, M., Mills, T., Ord, K. & Simmons, L. F. (1993), 'The m2-competition: A real-time judgmentally based forecasting study', *International Journal of Forecasting* **9**(1), 5 – 22.

Makridakis, S. & Hibon, M. (2000), 'The M3-competition: Results, conclusions and implications', *International Journal of Forecasting* **16**(4), 451–476.

Makridakis, S., Wheelwright, S. & Hyndman, R. (1998), *Forecasting: Methods and Applications*, 3rd edn, John Wiley, New York.

Marcellino, M. (2005), Instability and non-linearity in the EMU, *in* C. Milas, P. Rothman & D. van Dijk, eds, 'Nonlinear Time Series Analysis of Business Cycles', Elsevier Amsterdam.

Marcellino, M., Stock, J. & Watson, M. (2006), 'A comparison of direct and iterated multistop AR methods for forecasting macroeconomic time series', *Journal of Econometrics* **135**, 499–526.

McGill, J. & van Ryzin, G. (1999), 'Revenue management: Research overview and prospects', *Transportation Science* **33**(2), 233–256.

Meade, N. (2000), 'Evidence for the selection of forecasting methods', *International Journal of Forecasting* **6**(19), 515–535.

Mohr, M. (2005), A trend-cycle(-season filter), Technical report, European Central Bank.

Newbold, P. & Granger, C. (1974), 'Experience with forecasting univariate time series and the combination of forecasts', *Journal of the Royal Statistical Society. Series A (General)* **137**(2), 131–165.

Newbold, P. & Harvey, D. I. (2006), Forecast combination and encompassing, *in* M. Clements & D. Hendry, eds, 'A Companion to Economic Forecasting', Blackwell Publishing.

Ord, K. (2001), 'Commentaries on the M3-competition', *International Journal of Forecasting* **17**, 537–584.

Ozun, A. & Cifter, A. (2007), Nonlinear combination of financial forecast with genetic algorithm, Technical report, University Munich.

Pak, K. & Piersma, N. (2002), Airline revenue management: an overview of OR techniques 1982-2001, Technical report, Econometric Institute Report EI.

Palit, A. & Popovic, D. (2000), Nonlinear combination of forecasts using artificial neural network, fuzzy logic and neuro-fuzzy approaches, *in* 'The Ninth IEEE International Conference on Fuzzy Systems', pp. 566–571.

Pegels, C. (1969), 'Exponential forecasting: Some new variations', *Management Science* **15**(5), 311–315.

Peng, J.-Y. & Aston, J. A. D. (2007), The SSM toolbox for MATLAB, Technical report, Institute of Statistical Science, Academia Sinica. http://www.stat.sinica.edu.tw/jaston/software.html.

Pesaran, M. & Timmermann, A. (2007), 'Selection of estimation window in the presence of breaks', *Journal of Econometrics* **127**(1), 134–161.

Petrovic, D., Xie, Y. & Burnham, K. (2006), 'Fuzzy decision support system for demand forecasting with a learning mechanism', *Fuzzy Sets and Systems* **157**, 1713–1725.

Pfahringer, B., Bensusan, H. & Giraud-Carrier, C. (2000), Meta-learning by landmarking various learning algorithms, *in* 'In Proceedings of the Seventeenth International Conference on Machine Learning', Morgan Kaufmann, pp. 743–750.

Poelt, S. (1998), Forecasting is difficult - especially if it refers to the future, *in* 'Proceedings of the Reservations and Yield Management Study Group Annual Meeting'.

Prudencio, R. B. & Ludermir, T. B. (2004*a*), 'Meta-learning approaches to selecting time series models', *Neurocomputing* **61**, 121–137.

Prudencio, R. & Ludermir, T. (2004*b*), Using machine learning techniques to combine forecasting methods, *in* 'Proceedings of the 17th Australian Joint Conference on Artificial Intelligence', pp. 1122–1127.

Raftery, A. E. (1986), 'Choosing models for cross-classifications', *American Sociological Review* **51**(1), 145–146.

Riedel, S. (2007), Forecast combination in revenue management demand forecasting, PhD thesis, Bournemouth University in collaboration with Lufthansa Systems Berlin GmbH.

Riedel, S. & Gabrys, B. (2004), 'Hierarchical multilevel approaches of forecast combination', *Proceedings of the GOR 2004 conference* pp. 1–8.

Riedel, S. & Gabrys, B. (2005), Evolving multilevel forecast combination models-an experimental study, *in* 'Proceedings of NiSIS 2005 Symposium'.

Riedel, S. & Gabrys, B. (2007), Dynamic pooling for the combination of forecasts generated using multi level learning, *in* 'Proceedings of the International Joint Conference on Neural Networks', pp. 454–459.

Riedel, S. & Gabrys, B. (2009), 'Pooling for combination of multi level forecasts', *IEEE Transactions on Knowledge and Data Engineering* **12**(21), 1753–1766.

Rivas, V. M., Merelo, J. J., Castillo, P. A., Arenas, M. G. & Castellano, J. G. (2004), 'Evolving RBF neural networks for time-series forecasting with EvRBF', *Information Sciences* **165**(3-4), 207–220.

Ruta, D., Gabrys, B. & Lemke, C. (2009), 'A generic multilevel architecture for time series prediction', *accepted to IEEE Transactions on Knowledge and Data Engineering* .

Schreiber, T. & Schmitz, A. (1996), 'Improved surrogate data for nonlinearity tests', *Physical Review Letters* **77**(4), 635–638.

See, L. & Openshaw, S. (2000), 'A hybrid multi-model approach to river level forecasting', *Hydrological Sciences-Journal* **45**, 523–536.

Sen, P. K. (2005), 'Gini diversity index, hamming distance and curse of dimensionality', *International Journal of Statistics* **63**(3), 329–349.

Shah, C. (1997), 'Model selection in univariate time series forecasting using discriminant analysis', *International Journal of Forecasting* **13**(4), 489–500.

Sharkey, A. J. & Sharkey, N. E. (1997), 'Combining diverse neural nets', *The Knowledge Engineering Review* **12**(03), 231–247.

Shi, S., Da Xu, L. & Liu, B. (1999), 'Improving the accuracy of nonlinear combined forecasting using neural networks', *Expert Systems with Applications* **16**(1), 49–54.

Shi, S. & Liu, B. (1993), Nonlinear combination of forecasts with neural networks, *in* 'Proceedings of the International Joint Conference on Neural Networks', Vol. 1, pp. 959–962.

Smith, J. & Wallis, K. F. (2009), 'A simple explanation of the forecast combination puzzle', *Oxford Bulletin of Economics and Statistics* **71**(3), 331–355.

Stock, J. & Watson, M. (2001), A comparison of linear and nonlinear univariate models for forecasting macroeconomic time series, *in* R. Engle & H. White, eds, 'Cointegration, causality and forecasting. A festschrift in honour of Clive W.J. Granger', Oxford University Press, pp. 1–44.

Stock, J. & Watson, M. (2002), 'Macroeconomic forecasting using diffusion indexes', *Journal of Business and Economic Statistics* **20**(2), 147–162.

Stock, J. & Watson, M. (2004), 'Combination forecasts of output growth in a seven-country data set', *Journal of Forecasting* **23**(6), 405–430.

Suykens, J., van Gestel, T., de Brabanter, J., de Moor, B. & Vandewalle, J. (2002), *Least Squares Support Vector Machines*, World Scientific.

Swanson, N. & Zeng, T. (2001), 'Choosing among competing econometric forecasts: Regression-based forecast combination using model selection', *Journal of Forecasting* **20**(6), 425–440.

Talluri, K. T. & van Ryzin, G. (2005), *The theory and practice of revenue management*, Springer.

Tang, E., Suganthan, P. & Yao, X. (2006), 'An analysis of diversity measures', *Machine learning* **65**(1), 247–271.

Taylor, J. W. (2003), 'Exponential smoothing with a damped multiplicative trend', *International Journal of Forecasting* **19**(4), 715–725.

Teräsvirta, T., van Dijk, D. & Medeiros, M. (2004), *Linear Models, Smooth Transition Autoregressions, and Neural Networks for Forecasting Macroeconomic Time Series: A Reexamination*, Pontifícia Universidade Católica de Rio de Janeiro.

Terui, N. & van Dijk, H. K. (2002), 'Combined forecasts from linear and nonlinear time series models', *International Journal of Forecasting, Volume 18, Issue 3* pp. 421–438.

Timmermann, A. (2006), Forecast combinations, *in* G. Elliott, C. Granger & A. Timmermann, eds, 'Handbook of Economic Forecasting', Elsevier, pp. 135–196.

Tong, H. (1990), *Non-linear time series: A dynamical system approach.*, Clarendon Press.

University of Goettingen (2009), 'TSTOOL software package for nonlinear time series analysis [online]'. Available online: http://www.dpi.physik.uni-goettingen.de/tstool/ [03/06/2009].

van Dijk, D. & Franses, P. (2000), *Smooth Transition Autoregressive Models: A Survey of Recent Developments*, Econometric Institute.

Vilalta, R. & Drissi, Y. (2002), 'A perspective view and survey of meta-learning', *Artificial Intelligence Review* **18**, 77–95.

Vokurka, R., Flores, B. & Pearce, S. (1996), 'Automatic feature identification and graphical support in rule-based forecasting: a comparison', *International Journal of Forecasting* **12**(4), 495–512.

Wang, X., Smith-Miles, K. & Hyndman, R. (2009), 'Rule induction for forecasting method selection: Meta-learning the characteristics of univariate time series', *Neurocomputing* **72**, 2581–2594.

Weatherford, L. & Kimes, S. (2003), 'A comparison of forecasting methods for hotel revenue management', *International Journal of Forecasting* **19**(3), 401–415.

Witten, I. H. & Frank, E. (2005), *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd edn, Morgan Kaufmann.

Wolpert, D. (1996), 'The lack of a priori distinctions between learning algorithms', *Neural Computation* **8**(7), 1341–1390.

Yao, X. & Islam, M. (2008), 'Evolving artificial neural network ensembles', *IEEE Computational Intelligence Magazine* **3**, 31–42.

Zaki, H. (2000), 'Forecasting for airline revenue management', *Journal of Business Forecasting Methods and Systems* **19**, 2–6.

Zeni, R. H. (2001), Improved forecast accuracy in airline revenue management by unconstraining demand estimates from censored data, PhD thesis, Graduate School-Newark.

Zhang, G. (2007), 'Avoiding pitfalls in neural network research', *Systems, Man and Cybernetics* **37**(1), 3–16.

Zhang, G. P. (2004), 'A combined arima and neural network approach for time series forecasting', *Neural Networks in Business Forecasting, Hershey, PA: Idea Group Publishing:* pp. 213–225.

Zhang, G., Patuwo, B. & Hu, M. (1998), 'Forecasting with artificial neural networks: The state of the art', *International Journal of Forecasting* **14**(1), 35–62.

Zhao, L., Collopy, F. & Kennedy, M. (2003), 'The problem of neural networks in business forecasting: An attempt to reproduce the Hill, O'Connor and Remus study', *Sprouts: Working Papers on Information Systems* **3**(18), 234–243.