UNIVERSITY OF SOUTHAMPTON

Hybrid Machine Learning Models for Enhanced Sales Forecasting

by

Albert F. H. M. Lechner

A thesis submitted in partial fulfillment for the degree of Doctor of Philosophy

in the

Faculty of Engineering and Physical Sciences
School of Electronics and Computer Science

April 2021

Notice of Confidentiality

This report contains confidential information which is subject to nondisclosure. For this reason, this thesis may not be duplicated nor published without prior written consent from Rolls-Royce Motor Cars Ltd. This reservation expires on: 31.12.2023

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING AND PHYSICAL SCIENCES
SCHOOL OF ELECTRONICS AND COMPUTER SCIENCE

Doctor of Philosophy

Hybrid Machine Learning Models for Enhanced Sales Forecasting

by Albert F. H. M. Lechner

Sales forecasts are essential to every business's strategic plans and can both save the company money and increase its competitive advantage. However, many current businesses underestimate the opportunities which accurate forecasts provide and rely solely on judgemental forecasts from experts within the business. Machine learning and statistical forecasting methods are used by both academics and practitioners to increase the accuracy of these forecasting methods and can be further improved by combining knowledge from within the business with the statistical and machine learning techniques presented in this work. The models introduced in this study combine domain knowledge with data-driven approaches to improve forecasting on small datasets. The first approach in this work gathers global sales pipeline data to build a short-term sales forecast for a newly proposed dynamic cluster-based Markov (DCBM) model. By applying a newly developed algorithm, which first clusters the training and test set, the prediction of future sales for the next three months can be improved over a regular Markov transition model. The second proposed approach applies product lifecycle (PLC) information to improve the sales forecast. The accuracy of the sales forecast was increased for all 11 years for a luxury car manufacturer, comparing the newly developed PLC detrending approach to a common detrending by differencing approach in a seasonal autoregressive integrated moving average (SARIMA) framework. In a third model, the DCBM and PLC approaches are synthesised by using a SARIMA- long short-term memory (LSTM) framework capable of combining different data sources and thereby further increasing sales forecasting accuracy. The SARIMA-LSTM was able to predict the changes in sales occurring during the COVID-19 pandemic. All new models support short- and mid-term sales forecasting up to 12 months and represent an extension of knowledge in the area of sales forecasting.

Contents

List of Figures

List of Tables

Nomenclature

$\alpha_{hj}, {}_0()$	Baseline intensity function
$\alpha(t; Z)$	Hazard function with arbitrary covariate effect
$\alpha(t)$	Hazard function
β_{hj}	Vector of regression parameters
ϵ_t	Random noise at time t
γ	Threshold between cluster L and L_3
ϕ_1	First-order autoregressive coefficient
$\theta_q(B)$	Moving average operator of q-order
$\Theta_Q(B)$	Seasonal moving average operator
$\varphi_p(B)$	Autoregressive operator of p-order
B	Backwards shift operator
C_t	Cell state
$\tilde{C}_t$	Vector of new possible values
c	Constant
$d(x_i, x_j)$	Euclidean distance
d	Order of differencing
f_t	Forget gate
h, j	Transition states
H	History
i_t	Input gate
L_3	Opportunities likely to convert within three months
L	Opportunities likely to convert in more than three months
N	Opportunities not likely to convert
o_t	Output gate
p	Autoregressive order
p	Probability
q	Moving average order
S_t	Hidden state
S	Seasonal length
s	Time
t	Event occurrence
T	Time of event

X_t	Input at time t
Z	Covariate vector

Acknowledgements

Many thanks to Professor Steve R. Gunn from the University of Southampton for his excellent supervising and also to my supervisors Isobel Dando and Andrea Senso from Rolls-Royce Motor Cars for both overseeing and funding this work. I would like to show my appreciation to my mother and sister for supporting me throughout this time and for the guidance of my father, may he rest in peace.

Chapter 1

Introduction

This chapter serves as a general introduction and provides a brief overview of this thesis, starting with a short background of artificial intelligence in Section 1.1, followed by an overview of Rolls-Royce Motor Cars (which funded this research) as well as an introduction to their data in Section 1.2. Towards the end of this chapter, the research questions resulting from the problems faced within this research are presented in Section 1.3. The last section describes the contributions of this study as well as the published papers (in Section 1.4).

1.1 Artificial Intelligence

Artificial intelligence, or AI, has become increasingly important in everyday life and is being implemented in more daily applications, such as online searches, voice recognition, face detection, and traffic sign detection (Garnham, 2017). Not long ago, algorithms of machine learning (ML), one of the most prominent tools of AI, have managed to beat the world champions in the game GO, which is considered the most difficult game for a computer to win as the possible moves are nearly unlimited (Silver et al., 2016). Machine learning conducts tasks in a way humans consider smart in that machines are given specific data and learn for themselves (Michalski et al., 2013). Recently, self-driving cars, in which ML algorithms decide the vehicle's actions without a physical driver acting, have not only become a highly discussed topic but also a strongly researched area (Franke et al., 1998; Levinson et al., 2011; Chen et al., 2015). However, self-driving cars are not the only application of ML for manufacturers. Current research in that area not only focusses on autonomous driving but has also expanded to all areas of a business's value chain. As more data and computational power become available, ML is being used in many parts of a manufacturer's value chain, such as development, procurement, logistics, production, marketing, sales, and after-sales, followed by a connected customer after the purchase (Stock and Seliger, 2016).

Analysing past and current data to improve business is an important task, but predicting the future is even more crucial as several of a company's decision-making processes are based on forecasts. Vital decisions such as strategic planning, production planning, sales budgeting, marketing planning, and new product launches are influenced by forecasts. Therefore, many practitioners and researchers have focussed on new forecasting methods and improved forecasting accuracy as money can be saved and a business's competitive advantage could be improved (Wright et al., 1986; Armstrong, 2001).

Judgemental forecasts are based on human experience, which is applied to the time series and related future expectations and can thus suffer from personal bias (Webby and O'Connor, 1996). For that reason, companies use more data-driven approaches which reduce human bias from and include more features in the forecast than a human would be capable of. Machine learning models as well as traditional statistical models can outperform judgemental forecasts approximated by employees as they utilise more features of the available data with ease.

However, whilst emerging ML models increase accuracy, they also have challenges and shortcomings from a practical perspective as they can have a dense black-box structure, which often makes explanations difficult within a business environment. It is important for companies to not only increase the accuracy of a forecast but also to focus on explainability for the wider business network. Indeed, accuracy and explainability are key drivers within a business environment to ensure that the firm adopts new approaches and that managers gain confidence in the prediction. Managers can often be sceptical of new approaches like ML techniques, so they want evidence of improvements and an understanding of the key drivers behind the predictions; otherwise, they may not adopt new techniques into their daily work (Hagras, 2018; Holzinger et al., 2017).

Therefore, explainability in AI is an active area of research to make the black-box models more understandable (Zhang et al., 2018; Zhou and Gan, 2008). This can help workers better comprehend the logic behind models by recognising the patterns which drive a prediction and can also support the adoption of AI within businesses. Understanding the main features behind the prediction is not only essential for forecasting but also for other departments to focus on these features and thereby increase future sales (Langley and Simon, 1995).

Forecasting the buying process of a potential future customer supports short-term planning on a regional as well as a product-specific basis. Various techniques can model this process, such as seasonal autoregressive integrated moving average (SARIMA) models, but they suffer from drawbacks when being applied to real-world problems. For instance, one setback of SARIMA models is that they are not capable of capturing nonlinear patterns. For that reason, nonlinear models must be used alongside them (Gurnani et al., 2017). To overcome these problems, a new approach has been developed for this study based on a Markov model. Up to this point, Markov models have not been combined with a classification which has moving boundaries based on seasonal patterns for increased sales forecasting accuracy as well as easier business interpretability.

The second part of this work proposes a new way for including the age of a product defined over its product life cycle (PLC) into a forecast. Thus far, autoregressive integrated moving average (ARIMA) models have not been combined with PLCs detrending based on estimated parameters for increased sales forecasting accuracy as well as increased business interpretability. The final model proposed in this work, SARIMA-LSTM, combines both previously explained models as well as linear and nonlinear models to increase forecasting accuracy even during the COVID-19 pandemic.

1.2 Car Sales Predictions

Challenges introduced in the previous section are faced by companies such as Rolls-Royce Motor Cars (RRMC), which sponsored this research and provided the dataset used for the applications presented later in this thesis. RRMC is a super-luxury car manufacturer, with its factory and global headquarters located on the Goodwood Estate on the south coast of the United Kingdom. The iconic RRMC brand was first born in 1906, with Rolls-Royce Motor Cars Ltd. established as a wholly owned subsidiary of BMW in 1998. Their goal was, and still is, to take the best car in the world and make it even better. The company is composed of technical, administrative, and commercial departments, with the last group represented at market level by six regional teams, which manage a franchise network of 135 dealerships across 48 countries worldwide.

Business data within RRMC is collected, stored, and processed by a number of legacy systems which allow the organization to manage the flow of vehicles, parts, prospects, and customer data. As well as facilitating operations, the systems provide data for the measurement of business performance and, to a certain extent, to support decision making. The legacy systems and data are segregated and reflective of the organisational structure. The current system in RRMC for vehicle order to delivery is a software provider called SAP. For the customer relationship management (CRM) tool, the key system is Salesforce, a software company which provides an online tool to store all customer data in the cloud with access given to all employees in RRMC headquarters as well as dealers worldwide. These systems are not connected to each other and are thus limited by many factors such as partly automated and different human inputs which generate structured and unstructured datasets. Therefore, a promising goal is to enable the business to undertake intelligent decision making by combining existing data sources and using new ML techniques to improve current forecasting.

Up until now, RRMC's forecasting has mainly been dependent on judgemental forecasting combined with business targets, which is not able to adapt when the external environment undergoes rapid changes such as global economic crises or other events; it is also susceptible to introducing human bias into the predication (in that this type of forecasting can affect a manager's bonus, etc.). The results from this research will be applied to business decisions with an aim to optimise company performance. Once improved performance is demonstrated, the data-driven approaches will be automated and made available to key stake holders within the business through a user-friendly interface.

The data provided by RRMC dates back to 2003, when they started selling their first cars built in Goodwood. However, not all data sources go back that far; for example, Salesforce (their CRM tool) was introduced in 2009 and also underwent a drastic change in use in 2016, when dealers all over the world started using it as a business requirement to sell cars. For that reason, the data from Salesforce is only available from 2016 onwards, leading to a smaller dataset for the applications used in Chapters 4 and 6. Nonetheless, their sales have been recorded on a daily basis from 2003 until today and are broken down on a product as well as regional level. This breakdown is of special importance to understand how the business reacts to external changes in some parts of the world. In addition, understanding which products are of more or less interest to customers is important to steer marketing in the right direction and allocate budgets in the appropriate way.

Moreover, the sales data in the form of a time series is not linked to customer data at the moment. Thus, customer-specific data must be extracted from the CRM system, which can provide data on the location and features describing the customers. This data is currently not used to forecast the future behaviour of customers. Additionally, the data is different for all dealers globally because they are their own legal entity and follow their own guidance. Therefore, they use the system in various ways, which makes the data more difficult to use because the available information has different missing data points. The dealer is intended to fill out all data points in the system, but they often leave these blank for several reasons ranging from a lack of time to intentionally hiding information from the headquarters so they can keep more details about their own customers.

Another issue is that individual dealership managers have different requirements about what should or should not be in the system as some do not want their customers' personal data to be in the cloud or accessible to anyone outside their business. There are various approaches to address missing data, which are explained further in the related sections of this work. Regardless of missing data, the data fields filled in by dealers are used in this work to make more accurate predictions on both product and regional levels, which RRMC was not capable of before. This work provided the first approach at RRMC to combine different data sources to make better predictions which are automated and do not rely on judgemental forecasting.

The frameworks and models proposed in this research were created to deliver better forecasting accuracy not just for automotive companies but other industries as well.

The main research questions and contributions to the field are explained in more detail below.

1.3 Research Questions

As mentioned, the current forecasting at RRMC (as well as other manufacturers) is mainly based on judgemental forecasting without statistical or ML algorithms in place or the ability to generate data-driven forecasting models (Sanders and Manrodt, 2003; Reza et al., 2020). The reasons for this range from a lack of resources to the sheer quantity of data (which can be overwhelming), as well as a company's belief that judgemental forecasting is better and easier to adjust for unforeseeable changes. Therefore, the goal of this work is to increase forecasting capabilities whilst using all available data combined with newly developed ML algorithms as well as statistical models and the combination of both.

This research focusses in particular on improving sales and demand forecasting with the help of ML and answering the following research questions, the first of which is aligned with the future research of Niladri and Arun (2018):

1. In what way can ML enhance forecasting and demand estimation when the external environment undergoes dynamic, rapid and unforeseen changes?
2. How can PLC information be used to improve traditional sales forecasting methods?

This research introduces the following contributions to address these questions:

- An improved dynamic cluster-based Markov model (DCBM) exploiting clustered demand pipeline data, which is detailed in Chapter 4.
- A new approach to group demand data by clustering individuals' sales probability in combination with a conversion forecast to increase forecasting accuracy, which is detailed in Section 4.3.
- An increase in forecasting accuracy through the integration of PLC information in traditional forecasting methods, which is detailed in Chapter 5.
- A combination of both sales- and demand-affecting factors from the previous two points combined with a recurrent neural network, described in 6.

The remainder of the report is structured as follows. The following Chapter 2 presents an overview of previous work in the area of sales forecasting. Chapter 3 provides an overview of the technical background of statistical and ML models used within this work, ranging from ARIMA models and neural networks to time series validation methods and others, which are used as a foundation for the proposed approaches in later chapters. Chapter 4 introduces a new approach to forecast demand, which is evaluated on a dataset provided by RRMC. Chapter 5 introduces a new approach to detrend time series and improve their accuracy. This approach is evaluated with one time series of sales data from RRMC. A combination of the presented work from Chapters 4 and 5 is presented in Chapter 6 by using a SARIMA-LSTM framework to further improve the forecasting accuracy. Finally, in Chapter 7, conclusions are made based on the previous chapters, and suggestions for future research are described in detail.

1.4 Contributions

The contributions of this work are described in three papers, three of which were peer reviewed and published at international conferences in Dubai, Prague, and Spain in July 2021. The first paper, entitled 'Dynamic cluster based Markov model for demand forecasting', describes a new way of clustering the demand pipeline in a way that increases forecasting accuracy (further described in Chapter 4). The results of the DCBM model were presented at the 5th International Conference of Managing Value and Supply Chains 2019 in Dubai (Lechner and Gunn, 2019).

Albert Lechner and Steve Gunn. Dynamic cluster based markov model for demand forecasting. In *5th International Conference on Computer Science and Information Technology (CSTY-2019)*, pages 83–96, 11 2019.

This work's second contribution to the research field of time series forecasting is described in 'Product lifecycle de-trending for sales forecasting'. The main advantage of this approach is that it estimates a product's lifecycle by using ML techniques to include the age and specific features of a product which affect the number of sales over time into statistical time series forecasting methods. This new way of including external information into a time series forecast was peer reviewed and presented at the 2020 Complexis conference in Prague (Lechner and Gunn, 2020).

Albert Lechner and Steve Gunn. Product lifecycle de-trending for sales forecasting. In *5th International Conference on Complexity, Future Information Systems and Risk*, pages 25–33, 5 2020.

The third contribution of this work, entitled 'SARIMA-LSTM for sales forecasting', is a combination of both previously mentioned approaches into one SARIMA-LSTM model, which showed improved performance on the given dataset of car sales (also during and after the initial outbreak of COVID-19). The SARIMA-LSTM also introduced uncertainty to the predictions of the previous models and was accepted at the 7th International Conference on Time Series and Forecasting in Spain, where the conference date was postponed to July 2021, due to COVID-19.

Chapter 2

Sales Forecasting

Time series forecasting is a frequently researched topic with many extended branches for special cases. This topic solves problems in different areas such as forecasting financial markets or sales for a supermarket. For all these cases, there are various models to choose from, which also have different extensions and variables to improve their forecasting capabilities (Montgomery et al., 2008).

This chapter starts with an overview of the differences between sales and demand in section 2.1. Afterwards, this chapter provides an overview of related work in the area of sales forecasting related to this thesis, starting with statistical models. Sections 2.2 and 2.3 present a brief background of where statistical and ML models started to get used for forecasting and their evolution over time, followed by Section 2.4 which focusses on hybrid models mainly combining statistical and ML models to further improve forecasting. In Section 2.5, the current RRMC forecasting is explained in detail. The chapter concludes with Section 2.6,where all discussed models are contextualised within the work in this thesis. A deeper explanation of the foundation of the models used in this work is provided in Chapter 3, and the new approaches are introduced in Chapters 4, 5 and 6.

2.1 Sales and Demand

As the terms 'sales' and 'demand' are often used synonymously, an overview of the differences is introduced to better understand the distinct approaches used in this work. The demand for a product is what the end-use customer wants. Depending on the supply, demand can match the sales or be larger or smaller. Many products are not sold directly from the manufacturer to the end-use customer as there are subsequent businesses within the supply chain (Bernstein and Federgruen, 2005). Every supply chain has an independent demand, which is defined by the supply chain's end-use customer's demand for a product.

Therefore, there is a difference in business-to-business demand and business-to-customer demand. This is an important concept as the difference between demand and sales leads to promotions, marketing campaigns, or other methods to sell the remaining products. An incorrect sales forecast can lead to undersupply or a missed sales opportunity (Mentzer and Moon, 2005). By identifying the exact demand at the end-use customer level, a company can reduce the amount of unsold stock at every point in the supply chain and increase profit.

Also known as the bullwhip effect, every stage in the supply chain creates stock, which decreases the profit at every stage; this could be minimised by a better demand forecast (Lee et al., 1997). This unsold stock can only be sold by an effective marketing mix which creates new demand but reduces the profit (Hanssens, 1998). Most sales forecasting approaches consider the overall sold products, including even products sold through promotions or marketing activities which were not profitable when they were sold.

Depending on the type of product, the price point, and the frequency of a product purchase, identifying the demand at the end-use customer level can be difficult. Furthermore, many personal factors may play a role in an end-use customer's buying decision. Some personal factors can be determined through involvement; however, the involvement for low-price items bought with high frequency is short. For instance, the involvement for chewing gum is much shorter than that of expensive items bought infrequently, like cars or houses (Zaichkowsky, 1985; Mittal, 1989). The more time the sales personnel spend with possible future customers, the more information can be gained about the consumer's underlying decision process. With more interaction between salespeople and potential new customers (prospects), the size of the dataset grows over time. Whilst the interaction between the seller and buyer of chewing gum is short and cannot be used to gather much data, the interaction between a car dealer or real estate agent and the buyer contains much more information, leading to more data from the end-use customer (Dwyer et al., 1987). This data can be used to forecast the real demand for the product.

To forecast future sales, companies use different strategies which differ between top-down (TD) and bottom-up (BU). In the TD approach, the demand for the total sum of items is forecasted; in the BU approach, each individual item is separately forecasted and added to a total number (Schwarzkopf et al., 1988). There is no general agreement on which of the two approaches is better. Some authors argue for the BU approach (Dangerfield and Morris, 1992; Schwarzkopf et al., 1988), whilst others argue prefer TD (Fogarty and Hoffmann, 1991; Grunfeld and Griliches, 1960; Narasimhan et al., 1995). These findings lead to the assumption that the best approach depends on the data.

Williams and Waller (2011) have suggested that if point of sales (POS) data is not available, TD must be used, and if POS information is available, BU is the better option. As the individual forecasting for a large product portfolio becomes more complex and computationally expensive, and the amount of available data for each item becomes smaller, the accuracy of a prediction with the BU approach might decrease, unlike with the TD approach, because dependencies between items are lost with the BU approach (Williams and Waller, 2011). For many manufacturers, the individual dealer performance, as well as the regions in which they sell products, are of interest, especially on a short-term basis, where marketing can influence sales to achieve targets (Hanssens, 1998; Netzer et al., 2008). There are a number of approaches which could be applied in a TD or BU manner. The decision of which one to choose is dictated by the nature of the data. TD is preferable when only overall sales numbers are available, and a BU approach is better when individual customer data is available. In this work, TD is used in combination with aggregated sales numbers which are available per region or product and are forecasted using models like ARIMA; BU is used with a new Markov model (Chapter 4) where individual customer data at the POS is available.

2.2 Statistical Models

One of the most prominent and widely used approaches to statistical time series forecasting is presented in the work of Box and Jenkins (1976), which set a foundation for many derived models. Their work marked a new epoch in time series analysis. In contrast to the prevailing trend model, which assumed a deterministic process, Box and Jenkins assume a stochastic process to model a time series. An important consequence of this approach is that large increases or decreases in the time series can have a lasting effect on later time series values. This is a more realistic assumption, especially for economic time series. A significant amount of work was done after their study, resulting in many extensions of their model to improve forecasting accuracy.

Already in the late 1980s, researchers found that American companies were using computers and seasonal adjustments to make sales forecasts (Dalrymple, 1987). This trend continued with time series forecasting methods increasingly being used to improve sales forecasting accuracy within businesses.

As Webby and O'Connor (1996) found in their work in the beginning of statistical forecasting, judgemental forecasting was used not only as a benchmark but also to improve statistical forecasts and modify them according to the opinions of business managers. Over time, statistical forecasting methods were used more as a benchmark, comparing their results with state-of-the-art forecasting methods (Edmundson et al., 1988). The modification of statistical forecasting by external factors such as judgemental forecasting was the beginning of combined or hybrid models with the goal to improve forecasting accuracy by including additional information.

Davis and Mentzer (2007) have focussed on the evolution of sales forecasting techniques which more accurately reflect marketplace conditions. Their main focus is on the gap between theory and practise as a significant issue for sales forecasting research. They argue that although research has found many improvements for forecasting methods in theory, the practical improvements in forecasting accuracy did not increase by the same margin. Thus, they have proposed a theory-based framework to include organisational factors in sales forecasting by integrating research on organisational climate, capabilities, and learning. They conducted an extensive field study including 516 practitioners at 18 global manufacturing firms, resulting in empirical evidence of the fit between sales forecasting practises and the conceptual framework proposed in their work (Davis and Mentzer, 2007).

Sagaert et al. (2018) used forecasting in supply chain management to support planning for inventory, scheduling production, raw material purchases, and other functions. They typically refer to forecasts up to 12 months in the future, where traditional forecasting models consider univariate information which is extrapolated from the past but cannot anticipate macroeconomic events such as steep increases or declines in national economic activity. They use this additional information to increase the accuracy of the forecast, which, in practise, is made by managerial expert judgement. However, this judgement suffers from bias, is not scalable, and is expensive (Sagaert et al., 2018).

Fildes et al. (2019) have reviewed the research literature on forecasting retail demand. They introduce forecasting problems which retailers face, from the strategic to the operational, as sales of products are aggregated for stores and for the company overall. The factors which influence demand (in particular, promotional information) add considerable complexity so that forecasters potentially face the dimensionality problem of too many variables and too little data. Their review shows the importance of including various influential factors into a forecast (Fildes et al., 2019).

Out of many comparisons of statistical models with neural networks and other ML models, the study by Ainscough and Aronson (1999) was one of the first to compare them and find that neural networks outperform statistical methods based on a real-world dataset. For that reason, the following section presents an overview of important work in the area of ML-based forecasting techniques.

2.3 Machine Learning Models

Neural networks can model time series data and predict the future. However, after surveying the considerable amount of research conducted using neural networks for forecasting, the results are inconclusive as to whether they are more suitable than classical models such as ARIMA (Zhang et al., 1998). Poor performance could also result from small datasets or no experience in the setup of neural networks, so care must be taken in comparing various models on different or similar datasets.

Early adopters of neural networks for time series forecasting were Tang and Fishwick (1993). They studied neural networks as models for time series forecasting and compared their research with the Box-Jenkins method for small and larger time series datasets. They compared 16 different time series with different characteristics, finding similar performance compared to the Box-Jenkins method. Their experiments indicated that for time series with long memory, both methods produced comparable results. However, for series with short memory, neural networks outperformed the Box-Jenkins model. Because neural networks can be easily built for multiple-step-ahead forecasting, they may present a better long-term forecast model than the Box-Jenkins method. Neural networks are able to provide a promising alternative for time series forecasting (Tang and Fishwick, 1993).

For example, Chu and Zhang (2003) compare the accuracy of different linear and nonlinear models for forecasting aggregate retail sales, focussing on seasonal fluctuations. They compare several traditional seasonal forecasting methods with nonlinear versions implemented via neural networks which are generalised nonlinear functional approximators. Their results show that nonlinear models outperform linear models if the right prior seasonal adjustment of the data is conducted, which significantly improves forecasting performance of the neural network model (Chu and Zhang, 2003).

He et al. (2008) applied support vector regression (SVR) successfully for financial time-series forecasting. The challenge was to remove the noise from the data; therefore, they proposed a two-stage model using independent component analysis (ICA)and SVR. Their approach first uses ICA to generate the independent components of the forecasting variables and identify and remove the independent components containing the noise. After removing the components including noise, the residuals are used as an input for the SVR forecasting model. They validated their approach using financial time series, such as opening indexes, and the results show that their model outperformed the SVR model with non-filtered forecasting variables (He et al., 2008).

Another approach to forecasting sales in the retail industry, where a good sales-forecasting system is of high importance, is proposed by Sun et al. (2008). They applied a new neural network technique, named extreme learning machine, to examine the dependencies between sales and other factors which affect demand, such as product-specific features. Extreme learning machines are feed-forward neural networks without backpropagation that can achieve increased learning rates and do not need weight tuning. By evaluating the approach using real data from a fashion retailer in Hong Kong, they demonstrated increased performance compared to several sales-forecasting methods based on back-propagation neural networks (Sun et al., 2008).

Multi-layer perceptrons and other artificial neural networks have been widely used for time series forecasting since the 1980s. For some problems that exist in applications, such as initialization and local optima, however, the improvement of ANNs is (and will remain) the most interesting avenue not only for time series prediction, but also for other areas of intelligent computing. Kuremoto et al. (2014) proposed a method of time series prediction using Hinton and Salakhutdinov's deep belief networks, which are probabilistic, generative neural networks consisting of multiple layers of a constrained Boltzmann machine. They use a three-layer-deep network of RBMs to capture the features of the input space of the time series data. Then the RBMs, with their energy functions, use gradient-descent training to fine-tune the connection weights between the visible layers and the hidden layers of the RBMs (Kuremoto et al., 2014).

Various researchers and practitioners have used PLC curves to generate better sales forecasts. These usually include products where individual forecasting is not feasible for different reasons; therefore, clustering products into groups is an option developed by researchers (Solomon et al., 2000; Hu et al., 2017). To improve the forecasting for new products, researchers use an average PLC curve generated from sales numbers of clusters which share similar products. This type of forecasting is often used for products with short PLCs. Other related work uses different data sources to increase the accuracy of monthly car sales forecasts by including economic variables and Google online search data (Fantazzini and Toktamysova, 2015).

The ride-hailing service Uber used LSTM models to forecast extreme time series events for cases, such as holidays, when demand peaks. Therefore, they proposed a new LSTM-based architecture and trained a single model using heterogeneous time series. To model uncertainty, they combined Bootstrap and Bayesian approaches to produce a simple, robust and tight uncertainty bound, which was able to produce good coverage and provable convergence properties. First, the model primes the network by automatic feature extraction to capture complex time series dynamics. Then, feature vectors are aggregated via an ensemble technique. Finally, the vector is concatenated with new input and moved to the LSTM forecaster to predict demand. This procedure achieved over 14% improvement on average over the multilayer LSTM model they used for comparison (Zhu and Laptev, 2017).

Loureiro et al. (2018) explored the use of deep neural networks for sales forecasting in the fashion retail business. The deep learning approach was used to forecast sales in the fashion industry, predicting the sales of new individual products in future seasons. Lifecycles of fashion products are short and were quantified by large amounts of historical data that were collected and stored in the database of the company which supported the research. The model considered a wide range of variables, such as physical, product-specific features and the opinions of domain experts. The predictions were compared with different shallow techniques, such as decision trees, SVR, ANN and linear regression. The results showed good performance but did not perform significantly better than models such as random forest (Loureiro et al., 2018).

A new functional forecasting method was proposed by González et al. (2018), that attempts to universalise the standard seasonal ARMAX time series model to the L2 Hilbert space. The new model proposes a linear regression, in which functional parameters work on functional variables. The variables can be lagged values of the series (autoregressive terms), historically occurring innovations (moving average terms) or exogenous variables. The functional parameters used are integral operators whose kernels are modelled as linear combinations of sigmoid functions. Each sigmoid function is tweaked using a Quasi-Newton algorithm that reduces the sum of squared errors of the parameters. Their new model makes it possible to estimate the moving average terms in functional time series models. The new model was evaluated by forecasting the daily price profile of different electricity markets, showing improved capabilities (González et al., 2018).

Recently, DeepAR was proposed: a methodology for producing accurate probabilistic forecasts based on training an autoregressive recurrent neural network model on a large number of related time series. The model learns a global model effectively from related time series and is able, through rescaling and velocity-based sampling, to handle widely varying scales and generate calibrated probabilistic forecasts with improved accuracy. Furthermore, it is able to learn complex patterns, such as seasonality and growth in uncertainty over time. Empirical evaluations of different real-world datasets demonstrated that DeepAR generates improved accuracy compared to other state-of-the-art models (Salinas et al., 2020).

2.4 Hybrid Models

The foundation for hybrid models combining auto-regressive models with neural networks was laid by Zhang in 2003. He proposed a novel hybrid methodology combining both ARIMA and artificial neural network (ANN) models to use the advantages of both their linear and nonlinear modelling capabilities. He evaluated his proposed model on real datasets and showed an effective way of improving forecasting accuracy compared to the individual models. Following this work, a number of extensions were developed, some of which are explained below.

Dwivedi et al. (2013) evaluate the forecasting of sales data in the automobile industry using monthly sales from the automotive company Maruti. They primarily used moving average and exponential smoothing to forecast the past dataset and then used these forecasts as inputs for an adaptive neuro-fuzzy inference system. Empirical findings demonstrate that their model delivers better results than neural networks and linear regression (Dwivedi et al., 2013).

In addition, Arunraj and Ahrens (2015) extended their SARIMA model with external variables which reflect demand-influencing factors by using linear regression. The resulting new model was evaluated using data from the daily sales of banana from a discount retail store in Lower Bavaria, Germany. The results show improved forecasting accuracy than seasonal naïve forecasting, traditional SARIMA, and multi-layered perceptron neural network models. Their model also provides better prediction intervals and insights into the effects of demand-influencing factors for different quantiles (Arunraj and Ahrens, 2015).

Omar et al. (2016) propose a hybrid neural network model for sales forecasting based on the results of time series forecasting predictions and the popularity of new article titles. Their model combines historical sales data, popularity of article titles, and the prediction results of an ARIMA forecast, feeding into a back-propagation neural network forecasting model. They compared their model with conventional sales prediction techniques, which were outperformed by their new method (Omar et al., 2016).

Lu and Kao (2016) similarly introduced a new clustering-based sales forecasting method, using an extreme learning machine which also assembles the results of linkage methods. In a first step, they use the k-means algorithm to separate the training sales data into multiple disjointed clusters. In a second step, they use the extreme learning machine to construct a forecasting model. In a final step, they assign a test date to the best-suited cluster, which is identified by the result of combining five linkage methods. The identified cluster is then used to perform the final prediction. The proposed model is evaluated using two real sales datasets, with empirical results showing that model statistically outperforms eight benchmark models (Lu and Kao, 2016).

2.5 Historic Forecasting at RRMC

Forecasting at RRMC is done at the headquarters for all different regions and products, and historic sales numbers are mainly used. The forecasted numbers are not only a forecast, but they are also used as a target for the different regions and dealers. This already introduces bias as dealers go to great efforts to achieve that number, but they do not get any benefit from over-accomplishing. As RRMC is a business which aims to be profitable, they set their financial targets based on cars they need to sell in order to achieve profits. For setting the target, different external data sources such as wealth reports which reflect the wealth of potential customers all over the world are used to determine how many cars could be sold in total. The monthly forecasts are then broken down based on historical sales. This forecasting is done on a yearly basis for 12-months-ahead predictions, which are also used as a target.

During the year, these forecasts are adjusted by real sales numbers derived from the regions, where the overall goal is to achieve the yearly target set at the beginning of the year. If certain regions do not deliver the forecasted numbers, the goal is to achieve these numbers with sales from other regions to meet the yearly target. This procedure does not reflect in advance a realistic view of how many cars could be sold out of pure demand as it mainly includes historic data and current data from regions regarding how much they sold.

The shortcomings of this approach are that targets and forecasts are combined, which introduces bias towards the forecast. Currently, there is no ML or statistical approach used to include other data sources in the predictions to reflect a realistic market situation based on data. The reason for this is that, historically, the forecasting was always combined with target setting, and no resources were available to combine different data sources accessible to the business and use them to improve the forecasting to better reflect the market situation.

Thus, the motivation for this work was to combine the available data which RRMC owns in its different legacy systems to support better forecasting for the business. This work has helped RRMC change towards data-driven decisions and especially sales forecasting which includes sales and demand data as well as product-specific features which are included in the forecast.

2.6 Discussion

Utilising additional information to improve forecast accuracy is not new and has been undertaken by many researchers and practitioners (Sagaert et al., 2018; Omar et al., 2016). Indeed, the importance of PLC information as an additional source for forecasting was already proven by many researchers and is used in the approach of this work as well (Chu and Zhang, 2003; Solomon et al., 2000; Hu et al., 2017). In addition, Chu and Zhang (2003) have found that all forecasting methods, independent of statistical or ML, can benefit from better pre-processing in the form of de-seasonalising the time series prior to forecasting, which is also considered and implemented in this work.

These existing improvements have led to the idea of including product-specific factors which influence a forecast and are also known for the future. For that reason, this work focusses on including PLC-specific data known to a business in advance, thereby helping to remove seasonal effects occurring from new product launches or the end of an old PLC (further described in Chapter 5). The main advantages of this approach are improved detrending of a time series whilst adding features of the product into the forecast.

Moreover, including live demand information updated every month from different dealers worldwide can be useful to aggregate data and improve forecasting (Fildes et al., 2019). However, such information is hard to implement from outside of a business as not all data sources and their connections are available to the public, which was proven to be important in the past (Davis and Mentzer, 2007). Similarly, clustering time series data is also not new, but depending on the use case, it can be applied for different purposes (Lu and Kao, 2016) such as in the hybrid approach (Section 2.4), where it can help to combine different data sources. For this reason, one goal of this work is to combine both abovementioned models into one. Thus, this study identifies a promising area of research by combining live demand data with clustering the predicted outcome in the new proposed DCBM model (further described in Chapter 4).

2.7 Summary

Forecasting sales time series has attracted the attention of researchers in the area of machine learning to address the limitations of traditional forecasting methods. The work introduced in this chapter has already improved forecasting accuracy compared to judgemental forecasts and the first statistical forecasting methodologies. However, there is still room for improvement in different areas identified in this research. The two main contributors to improved forecasting are PLC information and live sales data from around the world. As both of these areas contribute to improved forecasting, it is logical to combine these concepts into one model that includes product-specific features and live sales data. This leads to a hybrid SARIMA-LSTM model (further explained in Chapter 6). The theoretical background of the proposed ideas is further described in the next chapter. This sets the foundation, from a technical point of view, for the different methodologies used.

Chapter 3

Technical Background

This chapter describes the background of the technical approaches used in the newly proposed models in this work. The three new approaches developed herein are separated from the already existing background of various statistical and ML models.

An overview of time series forecasting is given in Section 3.1. Throughout this work, nonlinear models are used for different purposes; therefore, as a starting point for non-linear models, a brief overview of neural networks is provided in Section 3.2. A number of nonlinear models were evaluated for the new models in this work, and an alternative for neural networks is further evaluated using classification and regression trees (Section 3.3). The performance of various models is evaluated in Section 3.4, and this chapter ends with a discussion of the technical background used for this work.

3.1 ARIMA Models

This section presents an overview of time series forecasting with a focus on ARIMA models as they are used in Chapters 4 and 5 for forecasting time series.

To maximise the available data, the DCBM model, introduced in Chapter 4, includes seasonal changes in sales, which can be forecasted by a time series of sales conversion over time. This forecast is accomplished using ARIMA models, which originated from the autoregressive moving average models. Autoregressive refers to the use of past values in the regression equation for the series; moving average specifies the error of the model as a linear combination of error terms which occurred at various times in the past (Ho et al., 2002). An ARIMA model is described by its values (p, d, q), where p and q are integers referring to the order of the autoregressive and moving average models, and d is an integer which refers to the order of differencing (Zhang, 2003). The equation for an $ARIMA(1,1,1)$ model is given by (Ho et al., 2002):

$$(1-\phi_1 B)(1-B)Y_t = (1-\theta_1 B)\epsilon_t \tag{3.1}$$

where ϕ_1 is the first order autoregressive coefficient, and B is a backwards shift operator given by $BY_t = Y_{(t-1)}$. The time series at time t is Y_t, Θ_1 is the first-order moving average coefficient, and ϵ_t is the random noise at time t (Arunraj and Ahrens, 2015). The ARIMA model can be used when the time series is stationary and there is no missing data within it. In the ARIMA analysis, an specified underlying process is derived based on observations of a time series to develop an accurate model which precisely illustrates the process-generating mechanism (Box and Jenkins, 1976).

An extension of this model is SARIMA, which relies on seasonal lags and differences to fit the seasonal pattern (Yaffee and McGee, 2009). By including seasonal autoregressive, seasonal moving average, and seasonal differencing operators, a $SARIMA(p,d,q)$ $(P,D,Q)_S$ can be stated as (Arunraj and Ahrens, 2015):

$$\varphi_p(B)\phi_p(B^S)(1-B)^d(1-B^S)^D Y_t = c + \Theta_q(B)\Theta_Q(B^S)\epsilon_t \tag{3.2}$$

where S represents the seasonal length, and B the backwards shift operator of a time series observation lag; k, symbolised by $B^k X_t = X_{t-k}, \varphi_p(B)$, represents the autoregressive operator of p-order $(1-\varphi_1(B)-\varphi_2(B^2)-\cdots-\varphi_p(B^p))$, $\phi_p(B)$ represents seasonal autoregressive operator with P-order $(1-\phi_1(B)-\phi_2(B^{2s})-\cdots-\phi_p(B^{2p}))$, $(1-B)^d$ represents the differencing operator of order d to remove non-seasonal stationarity, $(1-B^S)^D$ represents the differencing operator of order D to remove seasonal stationarity, c is a constant, $\Theta_q(B)$ represents the moving average operator of q-order $(1-\Theta_1(B)-\Theta_2(B^2)-\cdots-\Theta_q(B^q)$, and $\Theta_Q(B)$ represents the seasonal moving average operator with Q-order $(1-\Theta_1(B)-\Theta_2(B^{2s})-\cdots-\Theta_Q(B^{QS}))$.

There are various methods for model selection, with the most prominent ones being Akaike information criterion (AIC) and Bayesian information criterion (BIC). Complex models with a huge variety of parameters can lead to overfitting the data, which can in turn lead to a worse generalisation for unseen data. To control the effect of overfitting, an error measurement which includes the error itself and also the parameters to be chosen is of importance. Measures computing the error of fit with penalising the number of parameters are the AIC and BIC. Despite various theoretical differences, the main difference is that BIC more heavily penalises a model's complexity (Kuha, 2004). The AIC is used for model comparison in the next chapter's proposed models and is given by (Kuha, 2004):

$$AIC(k) = -2\hat{l}_k + 2\,|k| \tag{3.3}$$

where k is the number of model parameters, and $\hat{l}$ represents the log likelihood, a measure of model fit.

The forecasting horizon (used later in this work) was usually 12 months ahead in the future. This timeframe was chosen for reasons described in chapters 4 and 5. To forecast 12 months ahead, which is equivalent to a full year from January to December, the model was trained on a training dataset, and the prediction was made for the test set. To train the model, the training data was re-divided into a training part and a validation part, which consisted of the last 12 months of the full training dataset. This validation set was used to determine the parameters of the SARIMA model, using the AIC criterion described above. An illustrative example of the data is shown below in Figure 3.1.

FIGURE 3.1: Data split for training, validation, and test set

The available data is separated by years and subdivided by months, and the training dataset consists of the first six years followed by the validation dataset, consisting of one year, used for training the model's parameters; this is followed by the test dataset, consisting of the year seven in the example above. The data for the test set was unseen for the model prior to the prediction, so it was only used for model evaluation after the forecast was made, utilising different metrics defined in more detail in Section 3.4.

3.2 Artificial Neural Networks

The proposed DCBM model (introduced in Chapter 4) includes a new way of clustering data, where an ANN classification is used. For this reason, a short introduction to ANNs is provided in this section.

One of the most commonly used methods in ML is ANNs, which try to mimic the biological brain (Bishop, 1995). The equation for a simple neural network, the multilayer perceptron, is given by (Bishop, 1995):

$$y = \varphi(w^T x + b) \tag{3.4}$$

where w is vector of weights, x denotes the input vector, b the bias, and φ is a non-linear activation function. An ANN consists of several connected nodes, called neurons, which receive input from other neurons and send their output to the next neurons. The larger the network, the more input every neuron receives and the more neurons in the next layer receive their output (Bishop, 1995). An important feature of ANNs is that they are nonlinear models as well as universal approximators which provide competitive results by using effective training algorithms. Different training algorithms were used and developed over time, from back-propagation by E. Rumelhart et al. (1986) to newer methods which aim to accelerate the convergence of the algorithm.

Although ANNs do not need any prior assumption to build models, as a model is mainly determined by the characteristics of the data, the architecture of the network must be predefined (Zhang and Qi, 2005). In 1960, shallow neural networks with few neurons were used due to the difficulty of training deeper neural networks. More recently, new techniques have been found to train these networks and provide state-of-the-art performance. Different neural network architectures have evolved over the years with some adapted to specific applications. For example, convolutional neural networks are useful for vision problems (Goodfellow et al., 2016).

Over time, many suitable extensions have been developed, especially for time series forecasting, such as recurrent neural networks, which are designed to learn time varying patterns by using feedback loops (Fausett, 1994). Long short-term memory (LSTM) neural networks are an architecture of recurrent neural networks which essentially expand their memory. To know for how long information is to be stored in an LSTM and how to connect it to other neurons, each LSTM neuron consists of four individual components: the input gate, the memory and forgetting gate, the output gate and the interior of the cell with its linking logic. The input gate determines how and to what extent new values flow into a cell. The memory and forgetting gate determines whether information remains in a cell or is forgotten again. The output gate determines the extent to which values present or determined in the cell are output. Inside the cell, the components' interaction with each other is regulated to control information flow and storage (Hochreiter and Schmidhuber, 1997). The logic is implemented via neural functions with vector and matrix operations.

The input for the LSTM at time t is X_t with the hidden state from the previous time step, S_{t-1}, introduced to the LSTM block and computed for the hidden state S_t (Sagheer and Kotb, 2019). As a first step an LSTM decides which information is removed from the cell state by the forget gate f_t given by:

$$f_t = \sigma(X_t U^f + S_{t-1} W^f + b_f) \tag{3.5}$$

Afterwards, the LSTM decides which new information will be stored in the cell state by first deciding which values will be updated by the input gate i_t and, in a second step, by a hyperbolic tangent (tanh) layer that creates a vector of new possible values given by $\tilde{C}_t$ as follows

$$i_t = \sigma(X_t U^i + S_{t-1} W^i + b_i) \tag{3.6}$$

$$\tilde{C}_t = \tanh(X_t U^c + S_{t-1} W^c + b_c) \tag{3.7}$$

The new cell state C_t is updated from the old cell state C_{t-1}, given by:

$$C_t = C_{t-1} \otimes f_t \oplus i_t \otimes \tilde{C}_t \tag{3.8}$$

In the last step, the output is defined based on a filtered version of the cell state, where the output gate o_t chooses the parts of the cell state which are produced as the output by going through the tanh layer in order to get a value between -1 and 1 and multiply it by the output gate, given by:

$$o_t = \sigma(X_t U^o + S_{t-1} W^o + b_o) \tag{3.9}$$

$$S_t = o_t \otimes tanh(C_t) \tag{3.10}$$

Figure 3.2 shows the functioning of the LSTM cell (Sagheer and Kotb, 2019).

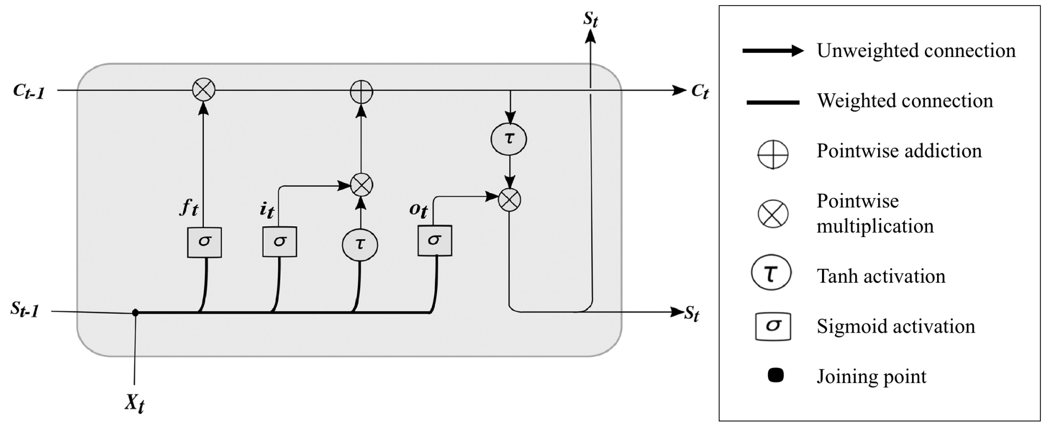

FIGURE 3.2: Structure of one LSTM cell (Sagheer and Kotb, 2019)

3.3 Classification and Regression Trees

Especially within business environments, the explainability of algorithms and models is of special importance. Neural networks are often referred to as black-box models which, once trained, are opaque. Other models are easier to explain to business stakeholders, and for that reason, this section introduces a widely applied model which provides an explanation of its decision making.

Decision trees have their origin in ML theory and can be used for classification and regression problems. They are based on a hierarchical decision scheme like a tree structure. Every tree has a root node, followed by internal nodes, which end at one point in terminal nodes. Each of these nodes takes a binary decision to decide which route to take in the tree until it ends in a leaf node. By splitting a complex problem into several

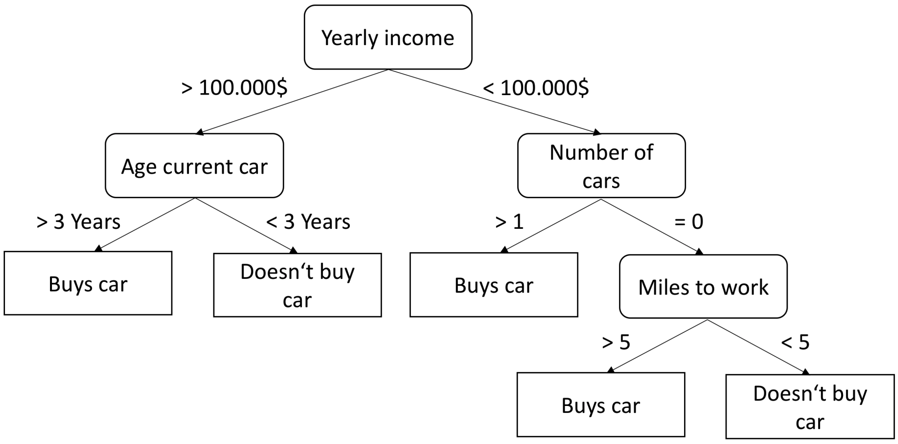

FIGURE 3.3: Example of regression tree with three variables

binary decisions, a decision tree breaks down the complexity into several simpler decisions. The resulting tree is easier to interpret and understand (Safavian and Landgrebe, 1991).

Classification and regression trees (CART) are a type of a decision tree which approximates real-valued functions. A regression tree shares similarities with a classification tree; however, the target variable takes ordered values with a regression model fitted to each node to give the estimated values of the target variable. The regression tree is constructed based on binary recursive partitioning in an iterative process. All training data is used to select the structure of the tree. The sum of the squared deviations from the mean is used to split the data into parts based on binary splits starting from the top. This process is continued until a user-defined minimum node size is reached, which leads to a terminal node (Breiman et al., 1984). Figure 3.3 illustrates a regression tree structure with a numeric output, given several input variables (yearly income, age of current car, and number of cars) and some nodes which take the binary decision, resulting in the leaf nodes if an individual buys a car or not. Within a business environment, the visual structure of the tree is important as it helps managers to understand the key drivers in the form of the sizes of all three variables to end with a certain prediction.

3.4 Time Series Validation

Different metrics are used throughout the literature by researchers as well as practitioners to evaluate the performance of forecasting models. Especially for time series forecasting, not all of them are of use as the temporal aspect of the data does not allow for the data to shuffle at random. For this reason, methods such as cross-validation do not work particularly well with all time series datasets due to the temporal aspect of the data. Thus, various evaluation methods are used, with the most prominent being the walk-forward validation, which maintains the temporal aspect of the data. The model is trained up to time t and then evaluated on $t+1$. In a next step, the model is trained up to $t+1$ and evaluated on $t+2$ and so on (Bergmeir and Benítez, 2012). Figure 3.4 shows how the time series is split up over time to evaluate model performance using the walk-forward approach.

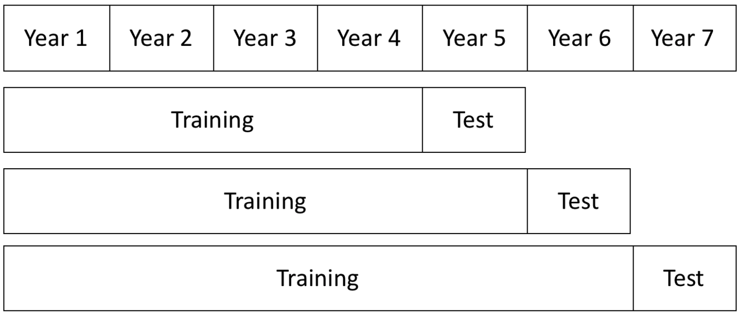

FIGURE 3.4: Walk forward validation for time series

Walk-forward validation is used for time series forecasting using models like ARIMA models. If other models are used for time series forecasting, different approaches can be used to evaluate their performance as well as choosing hyperparameters for the model.

For instance, cross-validation is a technique whereby the dataset is split in k different splits, where $k-1$ splits are used for training and the remaining part is used for testing of the model. This is repeated k times until every split is used once for testing. One goal of this procedure is to test the model's ability to predict unseen data in order to prevent overfitting or selection bias and thereby assess if the model is capable of generalising for new unseen data. Another advantage of cross-validation is that it shows the stability of a model and the sensitivity to training data. The results of the k different outcomes are averaged to get an estimation of the model's performance (Kohavi, 1995). An example of a fourfold cross-validation is shown in Figure 3.5, where the test set is moved throughout the whole dataset in four steps, covering all the available data.

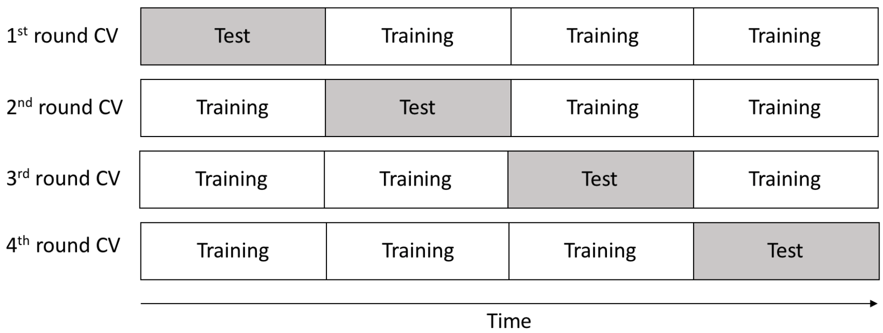

FIGURE 3.5: Example of 4-fold cross validation over time

The grey boxes show that the test data is moved from the beginning of the dataset until the end to test the model on the whole dataset. This procedure helps in understanding if the model fits the overall dataset compared to only training it on one part of the data and testing it on the rest, which could lead to wrong predictions if the data changed over time.

There are different methods of cross-validation; for instance, the one presented is suitable for temporal data, but for nontemporal data, different cross-validation methods could be considered as well (Roberts et al., 2017). Although cross-validation is hard to use for classic time series data, some of the data used in this work is used for training in such a way that cross-validation is feasible and used for model evaluation. This is true if the time series data is converted to a supervised learning problem where the model receives as input several lagged time series observations and as output the current timestamp.

In a next step, the performance of the different approaches presented must be evaluated by error measurements, such as the ones presented in the following. The forecasting evaluation can be done using different metrics which describe the error between the predicted value p_i and the observed value o_i. Thus, the error is defined by $e_i = p_i - o_i$. The metrics used in this work are the root mean squared error (RMSE) and the mean absolute error (MAE). The RMSE is given by (Murphy, 2013):

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} e_i^2} \tag{3.11}$$

In comparison, the MAE is given by (Murphy, 2013):

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |e_i| \tag{3.12}$$

As the RMSE squares the errors before they are averaged, it gives outliers a relatively higher weight, which can be useful within time series, especially for sales forecasts, where each over- or underestimation can cause financial risk. However, the downside is that one outlying datapoint can potentially have a large influence on the model, and if that datapoint is uncertain, it might be undesirable. The MAE also penalises large outliers but to a lesser extent than the RMSE. Theory shows that the optimal error measure can be determined by the additive noise distribution on the data, such that when it is Gaussian (Chai and Draxler, 2014), the RMSE is optimal, and when it has larger tails, the MAE (Willmott and Matsuura, 2005) is often more appropriate. However, in practise, often the noise distribution is not known and may not even be additive.

This work adopts the RMSE in the training process but also compares the test results under the MAE to highlight any large outliers in the predictions. The MAE is sometimes preferred within business environments as it can be more intuitive. The error can also be measured by dividing it with the reference value and thus defining the percentage error to overcome scale dependency (Bergmeir and Benítez, 2012). This was not done here because RRMC only uses total numbers for sales forecasting.

3.5 Discussion

All models explained in this chapter have been used widely in many companies and institutions all over the world. Although they improved forecasting over the years, there is still room for improvement in many areas. In recent times, a common thought is that neural networks and other universal approximators are able to solve all upcoming questions if there is enough data, but this is not always the case, and one must be very careful with using, for example, neural networks to solve new problems. For some problems, other models suit the purpose better; thus, it is important to understand the data and the problem that should be solved before applying any kind of algorithms. There is a broad variety of algorithms, and there are always other ways of using data to make predictions or support decision making.

Nonetheless, ML has emerged as a promising approach for many data modelling problems, but its application to many real-world issues within companies is a new endeavour. Many of the algorithms are quite complex and require experts in ML applications to apply them. Moreover, employees might be sceptical of any new approach until its demonstrated and validated to perform well. They must be convinced that the approaches are robust and applied well so that they not only perform well in theory but also on an ongoing basis.

To that end, this work has helped RRMC to break up the existing barriers which stopped them before as the results of this study are explainable and made them more comfortable adapting new ML techniques to their existing approaches. Thus, this research has helped RRMC to implement previous research as well as new ideas generated from this work into their daily forecasting and, with that, improve their sales forecasting capabilities. This study has also assisted them in making better use of their existing data as well as more fully understanding it. This was possible by combining different datasets retrieved from various legacy systems and applying new proposed algorithms to them, which is explained in the following chapter.

Different academics and practitioners have used the techniques presented in this chapter to improve forecasting. Based on their previous work, this study identifies three new approaches to improve forecasting accuracy. The forecasting methods range from classification algorithms to pure time series methods, like ARIMA models. The classification, using neural networks, is used in this research to separate a single time series and categorise different customer groups into ones which share similar buying behaviour. Since ARIMA models are appropriate for forecasting time series, they are used to separate clusters. Findings from these models support sales and demand forecasting for a variety of different businesses.

3.6 Summary

This chapter has surveyed central ideas behind research in the areas of statistical forecasting and machine learning that can be used for time series forecasting and many other applications. Many different approaches have been proposed, both in the statistical field and the machine learning field. All can be used for different problems, but it is rare for a single model to be the perfect fit for one problem. It is important to mention that the same problem can be solved with different methodologies; for that reason, a range of algorithms was introduced although not all can be listed here. There may be other algorithms that could solve the challenges presented in this work in a similar way but which were not listed here due to the large number of different algorithms available. Throughout this thesis, the main focus is on improving sales and demand forecasting, building on previous techniques and improving them to further enhance forecasting. Therefore, combinations of existing knowledge were used to create a hybrid model resulting from other new improvements in the area of time series forecasting, with a special focus on sales and demand forecasting.

Chapter 4

Dynamic Cluster-Based Markov Model

Thus far, to my knowledge, research has not applied multistage models to forecast demand for unknown prospects in the database, possibly because the process of sales is usually shorter than the prediction timeframe, and therefore this method cannot be used. However, depending on the type of product, prospects can be included in the data months before they actually buy. This data can then be used to model the process through various stages to forecast the demand. For products like real estate, boats, or luxury goods, the buying process can take weeks to several months. During this time, the collected data can be used for forecasting.

The new DCBM approach can be used to predict the sales pipeline for the future, and it includes a timeframe of conversion to the prediction's cluster. This added information can be used to improve forecasting. With the prediction, it is possible for a company to focus the forecast on a product- as well as regional-specific level, which supports short-term sales and marketing activities in order to sell more products. Furthermore, the proposed approach scores sales opportunities in different clusters, which can be used to influence the conversion by incorporating the resulting clusters into the business's marketing activities. The stages are used to represent an individual's buying process, and they are explained in Section 4.1. After modelling the stage transitions, the stages are clustered, and therefore a brief introduction to clustering is given in Section 4.2. The methodology of the DCBM model is introduced in Section 4.3, and its application to RRMC's sales pipeline is described in Section 4.4. The results of the new approach are evaluated in Section 4.5, followed by a discussion in Section 4.6.

4.1 Stage Transitions

Modelling transitions amongst stages and individuals' transitions within a given framework, like patients moving from the 'alive' to 'dead' stage, is a widely applied approach as it helps in predicting future events. Especially for small datasets, it can be advantageous to build domain knowledge about the underlying process of the individuals' transitions into the prediction and thereby improve forecasting. Therefore, this work investigates different stage transition modelling techniques for implementation into the final model.

Patient transitions in a medical environment is the most common topic within multistate modelling research, but other fields have been researched within similar frameworks (Cox and Reid, 1984; Lee and Wang, 1971; Andersen and Keiding, 2002; Collett, 2014). For example, Sullivan and Woodall (1994) used a Markov model to forecast educational enrolments over time. Zhu and Ching (2010) used multivariate Markov chain models for demand prediction, focussing on customers who already had a large individual sales history within a company's database. The transition between the two states can be measured with a hazard rate (Cox and Reid, 1984). By adding more stages, such as 'unwell', to the two-stage model, the standard survival model must be extended with a multistate model (MSM). An MSM is a time stochastic process which allows individuals to move between finite numbers of states (Hougaard, 2000).The different stages can be transient or absorbing, depending on whether a transition could emerge from the state. The transition between the stages will often be incomplete due to left-censored observation times, where the event occurred before the study started, or right-censored, when the study ends before an event occurs. Incomplete data occurs when the process is not observed from the origin or, for the right-censored case, if the individual is not in an absorbing stage when the data was captured. The presence of incomplete data must be captured when the likelihood functions are constructed (Meira-Machado et al., 2009).

The following section provides an overview of MSMs, which incorporates more than two stages (Andersen et al., 1993; Hougaard, 2000). Within this research, the stages represent a company's sales pipeline and are defined by a company's customer relationship management (CRM) system, which could vary by company and with the system used.

A multi-state process is stochastic and has a finite state space where T is a time interval, and the state occupied at that time is represented by the value of the process at time t. Over time, a history H is generated containing the previously visited states up to time t. The multi-state process is defined through transition probabilities between states h and j by (Andersen et al., 1993):

$$p_{hj}(s,t) = p(X(t) = j \mid X(s) = h, Hs-) \tag{4.1}$$

for $h, j \epsilon S, s, t \epsilon T, s \leq t$ where $s = s_1 < \cdots < s_{i-1} < \cdots < s_i = t$ is a part of the distance from s to t (Andersen et al., 1993). The transition rates' dependence on time leads to different model assumptions, including time homogeneous models, Markov models, and semi-Markov models. Time homogeneous models assume that the intensities are constant over time, which is independent of t. In Markov models, only the history of the process through the current stage is important for the transition intensities (Chiang, 1968).

Markov models are frequently used because of their simplicity. A Markov chain is a sequence of random variables (X_1, X_2, X_3) where the probability of moving from one state to another only depends on the previous state, the so-called Markov property. The Markov property is formulated as (Schweitzer, 1996):

$$p(X_{t+1} = s \mid X_t = s_t, X_{t-1} = s_{t-1}, \ldots, X_0 = s_0) = p(X_{t+1} = s \mid X_t = s_t) \tag{4.2}$$

for all times $t = 1, 2, 3, \ldots$ and for all states $s_0, s_1, \ldots, s_t, s$. In Markov models, the transition probabilities can be computed from the intensities by calculating the forward Kolmogorov differential equation (Kolmogoroff, 1931). Markov models can be differentiated into time homogeneous models and non-time homogeneous models. In time homogeneous Markov models, all transition intensities are supposed to be constant functions of time, however in some cases, the assumption of homogeneity may be unrealistic, and a nonhomogeneous model is needed (Gardiner, 2009).

To make the previous approach more explainable within a business environment and include many features without suffering from the curse of dimensionality, the new DCBM model (introduced in Chapter 4) is based on clustering the data. There are a number of ways to cluster data, and the following section presents a brief overview of these approaches.

4.2 Clustering

Clustering is an unsupervised learning problem which aims to segment a heterogeneous dataset into homogeneous clusters. As clusters are usually unknown before, it is different than the supervised learning problem classification. The data clustering problem has been addressed by different researchers in many contexts and is an important task in exploratory data analysis (Hartigan, 1975; Spath, 1980; Jain et al., 1999; Berkhin, 2006). To assign data points to a cluster, it is essential to measure their distance to other data points. Different similarity measures, like the Euclidean distance, are used in the literature and are explained further below. Various clusters are formed so the distances between data points in the same cluster are minimal and distances between data points of different clusters are maximal. Clustering techniques can broadly be categorised between partitional and hierarchical clustering (Jain et al., 1999).

To cluster data, a measurement for similarity between two data points, drawn from the same feature space, is essential. Usually, one calculates the dissimilarity between two data points using a measurement carefully chosen depending on the feature space. The most popular metric for continuous features is the Euclidean distance, which takes the ordinary distance between two points and is calculated by (Danielsson, 1980)

$$d(x_i, x_j) = \sqrt{\sum_{i=1}^{k} (x_i - x_j)^2} = \|x_i - x_j\|_2 \tag{4.3}$$

The Euclidean distance calculates the distance by taking the root of square differences between the coordinates of a pair of data points. However, depending on the probability density function, which describes the pattern representation, other measurements are more suitable than the Euclidean distance (Cha, 2007), especially in a higher dimensional space. To cluster the data by their similarity or dissimilarity, different clustering techniques can be used. These techniques are explained in the following.

At a top level, differentiations can be made between hierarchical clustering (which produces a nested series of partitions) and partitional clustering (which only produces one). The literature compares clustering in different ways, so Figure 4.1 was chosen, in accordance with Saxena et al. (2017) to provide an overview of the clustering techniques' taxonomy.

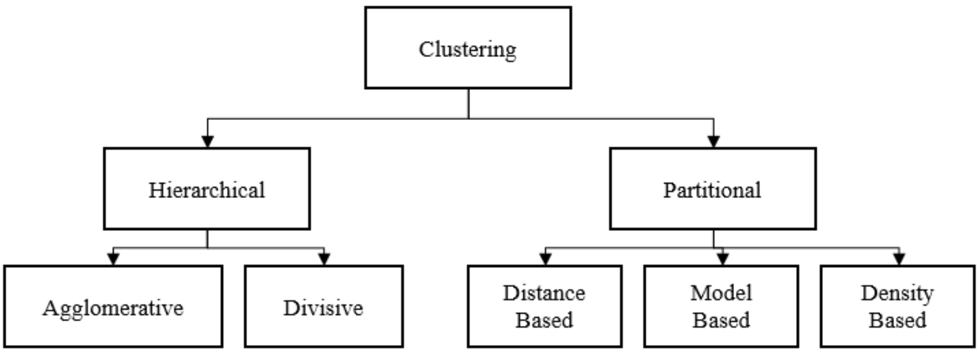

FIGURE 4.1: Taxonomy of clustering approaches

Hierarchical clusters are formed by iterative divisions of the data using either a TD or BU approach. The technique behind the TD approach is called divisive hierarchical clustering and breaks up larger clusters into smaller ones until every data point is its own cluster. The BU approach, called agglomerative clustering, starts with a single data point and merges the data points into increasingly larger clusters until all data points are in a single cluster (Murtagh, 1983). The two methods lead to a dendrogram which represents the split of the clusters. Both techniques can be further split into single, complete, or average linkage based on the similarity measures (Jain et al., 1999). As hierarchical clustering computes the distance between all data points for every cluster, it is computationally expensive and does not scale well with large datasets, resulting in $O(N^2)$ compared to partitional clustering with $O(N)$ (Steinbach et al., 2000).

Another technique, partitional clustering, scales better with large datasets. Partitional clustering assigns the data into k-clusters, which must be predefined without any hierarchical structure like in hierarchical approaches. The most common algorithm here is the k-means algorithm, which uses a squared error criterion like the Euclidean distance (Jain, 2010). Given a set of data points and the number of k clusters, the algorithm iterates with an expectation maximisation approach and splits the points into k clusters so that the total sum of the distances between the points within one cluster is minimised. The initialisation of the k points in this approach is random, potentially leading to different results over time. Further challenges are the choice of k and the vulnerability to outliers in the dataset (MacQueen, 1967; Saxena et al., 2017).When prelabelled data is available, the k-means algorithm can be initialised using a small amount of labelled data to generate an initial seed and thus create a semi-supervised learning problem (Basu et al., 2002).

As the algorithm will be trained on historic sales transactions, approaches like hierarchical modelling would take too much time and computational power to execute. Most of the described clustering techniques are unsupervised, but within the new approach, a combination of supervised approaches is used for grouping the data into different clusters. The new approach is introduced in the following section.

4.3 New Forecasting Approach

Within the given dataset, potential customers can go through various stages (introduced in Figure 4.4). Within the stage process, individuals can go from one stage to another but only in one direction as the system which manages the transitions does not allow going back to a previous stage. Not every individual starts the stage process in the first stage as this depends on the sales personnel and how they create the opportunity. Based on current knowledge from within the business, the stage transitions of an individual mainly depend on their current state and the information behind every opportunity, such as the time the customer has been known to the brand. Markov models are dependent on the last stage in the chain of different stages and therefore are a reasonable model for this problem.

A possible solution to the problem of applying a Markov process onto new unknown prospects is to compare them with existing customers who share similarities. For CRM data, Markov models are used to model the transitions between the recently observed purchase states to capture customer dynamics. Thereby, a key feature of the company's relationship with the potential customer is that the future prospects for that relationship are a function only of the current state of the relationship, which is defined by the potential customer's recent activity and not the particular path the opportunity took to reach its current stage (Pfeifer and Carraway, 2000). The information behind every opportunity which goes through different stages rarely changes over time, so the decision about which cluster an opportunity belongs to does not change over time. For that reason, a Markov models supports simplifying the stage transitions as the transition to a next stage only depends on the previous stage. If the opportunities would go through every stage and enrich the data behind them in this process, it could be possible to build a more complex model, which was not the case in the car sales dataset used in this work.

The Markov model factorises the problem in that the number of different models to train does not explode. If more data between potential customer and manufacturer interaction would have been available hidden Markov models could be useful to determine their effect on the impact on shifting potential customers to different unobservable stages (Netzer et al., 2008).

When information can be gained about the prospect during the buying process, this can be used to forecast, for example, buying a house or selecting features of a car or a boat. Here, the potential customer is usually known in the CRM system before he or she actually buys. All the information retrieved in this process could be used to forecast, on an individual level, if the prospect will buy or not and thereby support the demand forecasting on an individual level.

Thus far, sales and demand forecasting approaches have been mainly based on time series forecasting approaches, like ARIMA models, if only historic sales numbers are available. Machine learning was used to predict the outcome of the sales pipeline in the past but without an estimated timeframe of when the sales opportunities convert (Yan et al., 2015). In their research, Yan et al. (2015) assumed that the opportunities within the pipeline are more likely to convert by the end of the quarter, as business targets, and have only predicted the conversion for a two-week future period using a profile-specific two-dimensional Hawkes processes model.

In survival analysis, however, an occurrence of an event can be seen as a transition from one state to another, for example, from enquiry to retailed in a sales funnel. Within this research, Markov models are used to forecast sales out of the manufacturer's sales pipeline. The different stages which prospects go through in their buying process can be modelled as described in Chapter 3. The new DCBM is suggested for this: instead of modelling a covariate Z for every prospect, the approach here is to cluster the new incoming opportunities in the sales pipeline, described further in stage 3 of the DCBM algorithm. This approach was developed because many prospects in the pipeline have no sales, or even stage, history, but their features can be used to cluster them according to features from people who already went through the pipeline. To better explain the final model within a business environment and to reduce the curse of dimensionality, a problem occurring due to too many covariates, the DCBM is proposed (Meira-Machado et al., 2009). Each cluster is then modelled with a non-time homogenous Markov model to forecast the number of opportunities which convert over different timeframes. The time of conversion information can be gained by calculating the transition matrix for different timeframes. For a monthly forecast, the transition matrix for every cluster is broken down on a monthly basis for the next months, depending on the chosen timeframe.

All clustering approaches in Chapter 3 require the number of clusters to be predefined. This requirement leads to the question of how many clusters are suitable for the proposed approach. Since this question can be a whole research topic itself, it is not discussed in full detail. Nonetheless, clustering the data based on similarity is possible with approaches like k-means (mentioned in Chapter 3). As the purpose of every transition matrix is to represent similar buying probability, the clusters should be related to customer groups which are similar in their buying behaviour. Therefore, the main difference between every cluster should be the sale probability instead of pure similarities in the raw data. Thus, a new approach for dividing data into different clusters is proposed, which starts by estimating the sales probability (described further in stage 2 of the DCBM algorithm). Afterwards, the data is clustered by their sales probability. This approach has the same problem with the number of clusters compared to other clustering approaches but can improve the prediction of the Markov-based transition model.

Another approach to splitting the data based on sales probability is binary classification, which naturally results in two clusters. To improve the two-cluster system and the accuracy in a sales forecast, it is possible to split the first cluster which is likely to sell by using a forecast of the conversion number out of opportunities and divide it into two clusters, which is described further in stage 1 of the DCBM algorithm. The motivation for the second split is that most sales teams have targets by month, quarter, and year which drive their sales, and if there is no additional bonus for selling more than the target, sales teams tend to move potential sales to the next month, quarter, or year if they are close to or have fulfilled their quotas (Bouwens and Kroos, 2010). Splitting the first cluster supports the seasonal aspect of conversion and adds that information to the forecast. Therefore, the first group is separated into two clusters. One is the size of the prediction from the historic conversion, and the other is defined by the rest of the first cluster: potential sales which are moved to the future to fulfil the next month's target or are not treated at all.

Figure 4.2 describes the three clusters and their relationships to car sales per month. The blue line represents opportunities created by month as well as their conversion to a sold car on a rolling three-month basis after the opportunity was created (shown in yellow). The orange line represents the classification boundary, where everything above it represents a total number of opportunities likely not to convert, and below the orange line represents a total number of opportunities likely to convert to a sale. The new idea here is to split the opportunities below the orange line to create two further groups, divided by the yellow line. Everything below the yellow line represents opportunities highly likely to convert even more than the ones between orange and yellow. For comparison, the grey line represents car sales.

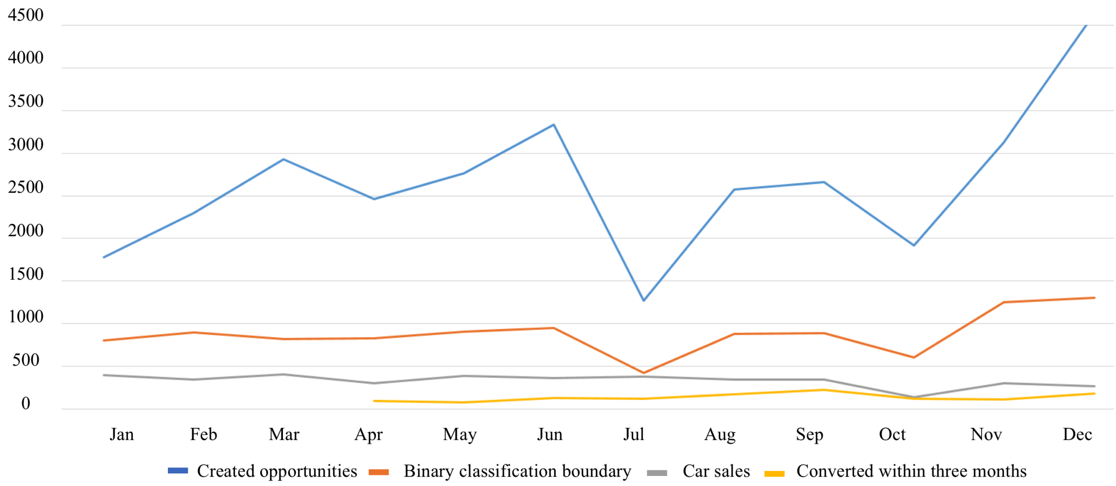

FIGURE 4.2: Amount of created opportunities (blue) compared to the binary classification boundary (orange), sales (grey), and converted opportunities within three months (yellow) from January to December

The size of the first cluster is defined by a time series forecast of historic opportunity conversion within a predefined timeframe, in this case, three months as this is the chosen forecasting timespan. The number of historic conversions can be forecasted by classical time series approaches like seasonal ARIMA models, which also contain information about seasonally driven sales by targets. The calculation between the factors above is provided in the DCBM algorithm, consisting of three stages as shown in Figure 4.3. This diagram shows the data used for training and the flow of information passing between each stage.

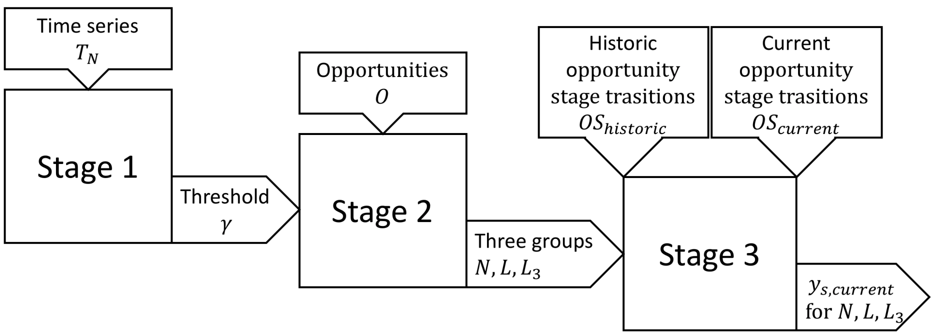

FIGURE 4.3: Three stages of the DCBM algorithm and their inputs (coming from above) and outputs (directing to the right)

Stage 1 describes a seasonal ARIMA model used for forecasting the future conversion of created opportunities per month within the next three months. The model selection is based on AIC of parameters between 0 and 2. The output of stage 1 is a forecast for the next month of opportunities which will be converted within three months. This output is used in stage 2 to split the opportunities into three separate groups which are predicted to convert.

DCBM algorithm stage 1: SARIMA opportunity conversion forecast

Input	Time series T_n = historic sales conversions out of opportunities per month (2015 to today)
Create	Construct $SARIMA(p, d, q)(P, D, Q)m$
Initialise	Set SARIMA parameters based on grid search of AIC result between 0 and 2
Algorithm	Time series forecast for following three months
Output	Threshold γ

Stage 2 of the DCBM describes a neural network-based clustering which gives every opportunity a probability to convert. Opportunities with a probability over 0.5 are predicted to convert, and opportunities below 0.5 are predicted not to convert. The first group of the new DCBM model is defined by opportunities with a probability up to 0.5. The DCBM approach categorises the group likely to convert another time with the output of stage 1. Ordering the opportunities by their probability with the highest first, the third group consists of the top γ opportunities, and the second group consists of opportunities greater than or equal to 0.5; probabilities lower than that form the third group. All three groups are used in stage 3 of the DCBM algorithm as for every single group, a Markov transition model is applied.

DCBM algorithm stage 2: neural network-based opportunity clustering

Input	Opportunities $O = (x_1, y_1), \ldots, (x_n, y_n)$
Create	Construct neural network with specified number of inputs, hidden nodes, and outputs
Initialise	Randomly initialise weights for neural network
Algorithm	Use gradient descent algorithm to optimise ω_{M_P} Obtain predictions $y = f(x; \omega_{M_P})$ Cluster Opportunities O into three groups based on predictions N (not likely to convert): $0 \leq f(x; \omega_{M_P}) < 0.5$ L (likely to convert in more than three months): $0.5 \leq f(x; \omega_{M_P}) < \gamma$ L_3 (likely to convert within three months): $\gamma \leq f(x; \omega_{M_P}) < 1$
Output	N, L, L_3

All three groups from DCBM algorithm stage 2 are now used to create a Markov transition model for all various stages of the opportunities. Stages 1 and 2 are applied to opportunities older than 12 months as they will not convert anymore in order to get $OS_{historic}$ and on all new incoming opportunities to get $OS_{current}$. The final goal is to calculate the outcome of $y_{s,current}$, representing which stages the opportunities will be in within the next three months.

DCBM algorithm stage 3: time homogeneous Markov transitioning model

Input	Historic monthly opportunity stage transitions $OS_{historic} = (s_1, s_2, s_n, y)$ Current month opportunity stage transitions $OS_{current} = (s_1, s_2, s_n)$
Initialise	Construct historic transition matrix ϕ_{O_S} $\phi_{O_S} = \begin{matrix} s_1 s_2 & s_1 s_3 & s_1 y \\ & s_2 s_3 & s_2 y \\ & & s_3 y \end{matrix}$
Algorithm	For all three groups N, L, L_3 from DCBM algorithm stage 2: $y_s = OS_{current} \times \phi_{O_S}$
Output	$y_{s,current}$ for N, L, L_3

If the number of opportunities categorised as a sale in the binary classification is lower than the forecasted number of conversions, the target of the next period likely cannot be achieved from pure demand.

As there is a difference between the demand and actual sales, one can use the demand information to improve the sales forecast. An ARIMA ANN (Zhang, 2003) can be used to model the residual errors of the SARIMA model with the demand information given above. This solution can also address the potential problem of information asymmetry between dealer and manufacturer as there can be delayed or missing information within the data. The transition from demand to sales forecast was not applied in the proposed approach but is included in the work done in Chapter 6.

The result of the different approaches is compared in the next section using an example dataset.

4.4 Application and Data Pre-processing

The main focus of this application is to improve the sales and demand forecasting on a regional as well as model-specific level. This gives the sales and marketing teams worldwide a chance to steer sales by using marketing tools to reach their sales targets. The following sections describe the car sales channel data from RRMC and the features and their pre-processing.

RRMC's cars are sold exclusively through a dealer network of over 100 dealers worldwide. To sell a car, a dealer must create an opportunity in the system, either through converting a lead or manually creating a new opportunity. Therefore, before selling a car, there must be an opportunity in the system. If the dealer only creates the opportunity to directly sell the car or the prospect walks into the showroom and directly orders the car, the conversion time from an opportunity to an order is under one month. About 40% of opportunities are created within the same month as the orders. In this case, using opportunities to model individual transitions is difficult as they are created within the last stage. Because every opportunity is created by a human, the quality of the opportunity is much better than online-generated leads, which are not prequalified. However, if the manufacturer creates opportunities as intended, then it is possible to model the transition amongst the stages from enquiry (En), qualification (Qu), configure (Co), test drive (TD), negotiate (Ne), commitment (Co), and ordered (Or) to closed (Close), which can either be a sold car or a lost opportunity and can transit from stage to stage. Figure 4.4 illustrates this concept but does not depict all possible transitions, only including those from the first enquiry stage and all possible stage transitions to the last stage (Close) for easier understanding.

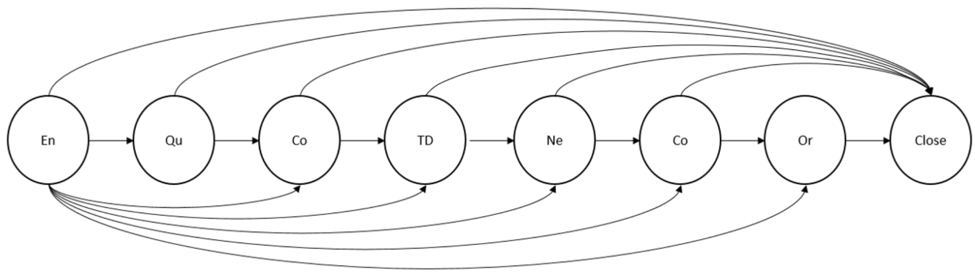

FIGURE 4.4: Opportunity stage transitions from enquiry to a closed opportunity

Based on all possible transitions, the stages proceed as illustrated in Table 4.1 between opportunity stages created in 2016 and 2017, in which every value above a diagonal line between the top left corner and the bottom right corner displays transitions forwards and every value below transitions backwards. The darker the cell, the more transitions occurred from and to that stage. The company has sold cars for over a decade, but the stages of the opportunities changed over time as a new CRM system was introduced. Thus, for this study, only data from 2016 onwards is used. Data used for training was gathered on a monthly basis between January 2016 and the end of December 2017, for 24 months of total data. In total, the forwards transitions for the years 2016 and 2017 add up to over 100,000, whereas the transitions backwards, displayed under the grey diagonal line, total around 3,500.

	Enquiry	Qualification	Configure	Test Drive	Negotiate	Commitment/Order	Retailed/Delivered	Lost/Cancelled	Total
Enquiry	566	1237	58	6801	289	560	694	17235	27440
Qualification	77	653	531	923	454	188	335	2934	6095
Configure	10	60	76	415	100	104	45	422	1232
Test Drive	80	115	97	1099	718	485	545	12289	15428
Negotiate	47	99	133	215	148	1549	248	982	3421
Commitment/Order	24	38	33	39	500	686	7381	471	9182
Retailed/Delivered	264	181	29	202	96	563	4492	121	5953
Lost/Cancelled	45	77	18	51	68	65	117	33529	33970
Total	22341	7182	1614	18849	3548	9146	14483	68163	102721

TABLE 4.1: Transitions from and to opportunity stages created in 2016 and 2017, with darker grey highlighting more transitions

This matrix is not time dependent, and the transitions from and to the same stage reflect a problem within the system, which has since been fixed. The stages moving backwards were excluded because the system rules prevent them from occurring again, and as there are few, the bias created by deleting them is acceptable. Therefore, these transitions were not included, and the final used data is shown in Table 4.2. As the CRM system changed over time, over 62,000 opportunities could be used out of approximately 170,000 created thus far. This reduction in usable opportunities is due to the changes within the CRM system occurring before 2016. In addition, around 1% of the opportunities are aftersales related and are not used as they are not related to new car sales. Approximately 5% of opportunities could not be used due to a problem within the system as every opportunity should go from one stage to the next and not back to the stage at which the opportunity originated. This mistake happened for around 5% of the opportunities because users sometimes employed the system incorrectly, and no rule within the system blocked opportunities from moving backwards. With new rule implementations within the system, this error can no longer happen. Therefore, the opportunities moving backwards were excluded. Of the 62,000 opportunities, around 13,000 converted to a sale. The first three months of 2018 were used as a test set, and the results were compared on a monthly basis.

	Qualification	Configure	Test Drive	Negotiate	Commitment/Order	Retailed/Delivered	Lost/Cancelled	Grand Total
Enquiry	1160	48	6721	242	536	430	17190	26327
Qualification		471	808	355	150	154	2857	4795
Configure			318	33	71	16	404	842
Test Drive				503	446	343	12238	13530
Negotiate					1039	152	914	2105
Commitment/ Order						13599	694	14293
Retailed/ Delivered							13	13
Grand Total	1160	519	7847	1133	2242	14694	34310	61905

TABLE 4.2: Cleaned transitions amongst opportunity stages created in 2016 and 2017

Giving the transitions from Table 4.1 (a temporal aspect of the transitions), it takes, on average, 57 days to change from one stage to another. As many opportunities are located in a stage but are in reality lost, every opportunity which did not retail or was cancelled is assumed to be lost after six months. The goal of modelling the transitions is to improve forecasting on a short-term basis (within the next three months) on both a regional and product level.

The dataset consisted of the features displayed in Table 4.3 for all opportunities, which were pre-processed (as described in the following section) before use in the algorithm from Section 4.3.

The categorical variables *region*, *product* and *lead source* were pre-processed by using one-hot encoding, where the categorical variable is replaced with a new binary variable added for each unique categorical value. *related contact type* was converted to either a 0 for prospect or a 1 for existing customers. *description*, *primary campaign source*, and *secondary campaign source* were converted to a 0 if they were blank and a 1 if they were filled in. As *sale probability* had missing values, these values were replaced with either the average of probabilities for retailed opportunities or the average probability of lost opportunities. This approach makes best use of *sale probability* compared to leaving it out completely or replacing it with zero. *stage* was converted from a categorical variable to a numerical variable between 0 and 1 with steps of 0.1, starting from enquiry with (0.1) and ending with sold with a value of (1) as this is the natural ordering of the stages. An additional feature, *days known*, was created to replace *created date* and *contact created date* and is calculated by using the difference in days between the day the opportunity was created and the day the contact behind the opportunity was created. This new feature was created to determine whether the person behind the opportunity was known to the business before and, if so, for how long.

Feature	**Example**	**Data type**	**Missing data**
Region	North America	Categorical	0%
Related Contact Type	Prospect	Categorical	0%
Contact Created Date	03/10/2017	Numerical	0%
Created Date	03/10/2018	Numerical	0%
Days Known	365	Numerical	0%
Product	Range	Categorical	0%
Stage	Test drive	Categorical	0%
Lead Source	locator.com	Categorical	36%
Last Stage Change Date	03/10/2018	Numerical	0%
Description	Wants to buy coupe	Categorical	0%
Primary Campaign Source	2018 CRE campaign	Categorical	31%
Secondary Campaign Source	Launch campaign	Categorical	78%
Sale Probability(%)	50%	Numerical	32%

Table 4.3: Overview of used features with examples, data type and % of missing data

4.5 Results

The prediction aims to discover how many cars out of the opportunities will sell in the following three months. This prediction can support the sales team on a short-term basis and reflects a live view of sales on a regional and product level. With that information, the sales team can allocate a budget to the regions and products which struggle with future sales, helping them to achieve their targets.

The regression, used for clustering the opportunities, was initially done using a neural network. This approach has the drawback of being a black-box model where the importance of features is not identifiable. Within a business environment, it is important to understand the key features which drive the prediction to understand the process better and influence it in the future. For that reason, tree-based methods were used within the DCBM model to make the approach more explainable to business stakeholders. The results of tree-based methods also showed slightly better performance. The ANN model results for 12 months MAE is 19.25% compared to the tree model with 18.71%. As both models are universal function approximators, they are able to model the regression with the difference that the tree model can output the feature importance as well, which is an important advantage within this framework. For that reason, from this point on, the DCBM model is based on a decision tree regression.

The DCBM utilises three clusters with dynamic boundaries. Other cluster sizes are compared in Table 4.4 to determine whether the results could be improved by increasing the cluster size. In theory, splitting it in more than one cluster should give better results as every single cluster represents the different buying processes which prospective customers go through. Therefore, more clusters could result in a better prediction in theory, but it was shown that for the given dataset, this was not the case. This creates a trade-off between the cluster size and the gained forecasting accuracy. An increasing number of clusters enables one to investigate an increasing number of behaviours, but it also gives the algorithm less data to estimate the behaviour of each group. In the limiting case with a single data point per 'cluster', it is not possible to estimate the behaviour of new individuals anymore.

Table 4.4 compares the results for different cluster sizes ranging from one cluster, which represents a traditional Markov model, up to 20 clusters. The first row shows the yearly error for the prediction when adding all 12 months of individual forecasts together. The second row shows the mean absolute error of all 12 individual months. The cluster size was equally distributed within the prediction range from zero to one. The results indicate that the DCBM with its three dynamic clusters showed increased performance compared to the other equally distributed clusters with sizes ranging from 1, 2, 3, 5, and 10 to 20. With the given data, more than 20 clusters lead to an error as there were clusters with little to no data.

	DCBM	1 cluster	2 clusters	3 clusters	5 clusters	10 clusters	20 clusters
Yearly error	-1.06%	29.25%	35.82%	27.79%	25.49%	20.48%	20.48%
Monthly error	18.71%	39.08%	43.55%	37.03%	36.42%	31.27%	31.27%

Table 4.4: Comparison of the three-cluster DCBM approach to different cluster sizes ranging from 1 to 20, compared on a yearly as well as monthly error basis

A more detailed comparison of the 12 months for the year 2018 (from Table 4.4) is shown in Figure 4.5. The conversion to a retailed/delivered car is shown in numbers on the left axis through all 12 months of 2018, represented on the bottom axis. It is shown that the DCBM performs better for seven months and shows comparable results for the remaining months.

Month four shows an especially large difference in the prediction between the dynamic boundary and a fixed boundary. The reason here is that in month four, more opportunities were created than usual, and the fixed boundary approach from a traditional Markov model cannot adapt to the high number of opportunities and predicts a higher conversion as well. Using a tenfold cross-validation it was illustrated that the DCBM provides stable results. By randomly leaving out 10% of the data and using it as a test set, the predictions stay within a range of 5% deviation.

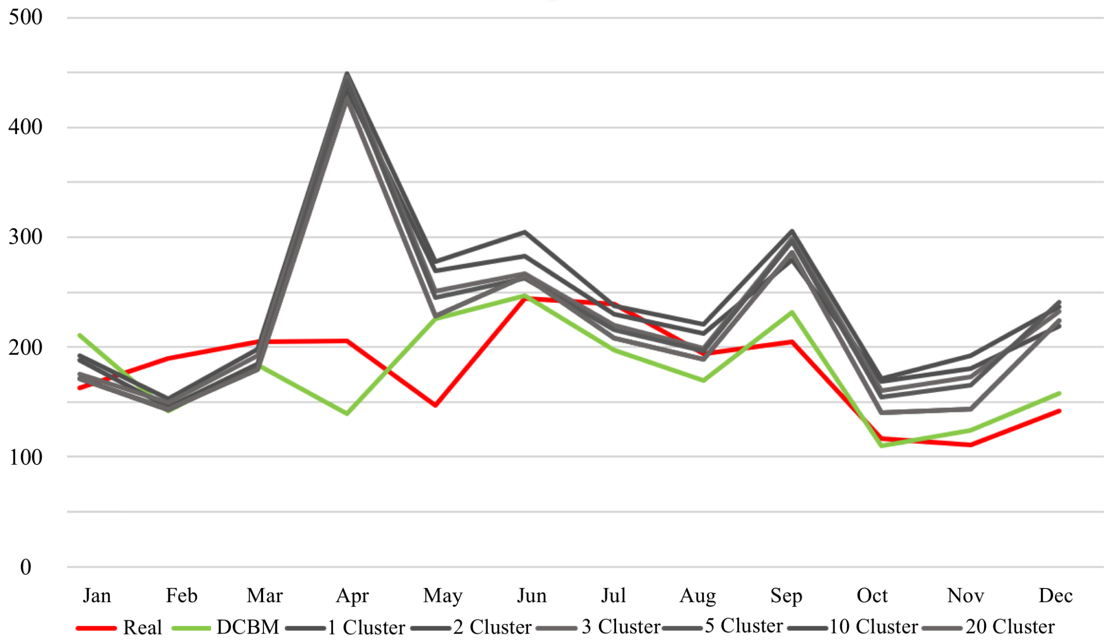

FIGURE 4.5: Comparison of the three-cluster DCBM approach (green) to different cluster sizes ranging from 1 to 20 (different shades of grey) predicting future demand in total car sales on the vertical axis compared on a monthly basis

Using a decision tree model, it was possible to identify the most important features which were reported back to the business. The features *sale probability*, *model* and *days between contact and opportunity* were proven to be the main drivers behind predicting the conversion of opportunities into sold products. This enables RRMC to focus on these three important drivers when evaluating their incoming opportunities.

4.6 Discussion

A drawback of the DCBM is that the time conversion is not completed at the opportunity level. Because it is completed on a cluster level, the approach cannot track when each opportunity will convert. However, the DCBM supports the regions and products potentially at risk and can be supported by marketing activities. Clustering the pipeline as proposed can also be beneficial for marketing purposes, especially the proposed model which splits the binary classified opportunities based on a time series forecast. For example, one could immediately approach the second stage of the cluster and wait on the first cluster for two months if the opportunities did not convert by then. As the first cluster has the highest probability to convert to a sale, no immediate additional action is required since the customer is predicted to buy. In contrast, in the second cluster, whilst the probability for a sale is still over the cut-off of 0.5, the realistic chances of a sale are lower and could benefit from marketing activities towards the prospects to increase the chance of a sale. The effect of such clusters on marketing could be a future research question related to the marketing department.

The proposed approach was introduced at the RRMC Motor Cars sales channel development conference in January 2019. Afterwards, one sales region requested the approach to be introduced at every dealer in their region to start a cleaning process based on which clusters the existing opportunities belonged to. This was also used to gain further insights into how the data is created and used for the proposed algorithm and to evaluate the reasoning behind the new three-cluster approach. The three clusters were found useful within this framework. Although the proposed approach was designed for forecasting and not cleaning data, splitting opportunities into three groups was appreciated by sales personnel as this approach made targets and promotions easier to understand and manage on a monthly basis. In addition, after a maximum of six months, dealerships tend to open new opportunities for those which already exist, which supports the assumption that an opportunity is lost after six months.

As the proposed approach uses opportunities, it does not cover all sales as some of them happen before the opportunity is created in the system. To address this, a new approach of forecasting a total sales number is proposed in Chapter 6, where a SARIMA-LSTM is applied to solve this. This approach was chosen as it is more straightforward from a technical point of view to first forecast the opportunities and afterwards use them to forecast overall sales numbers.

The car sales data for the application in this chapter used information from previous years to predict the outcome of future years. This was subject to the assumption that the training data comes from the same distribution as the test data and does not change over time. However, this assumption is not always true. If training and test data are not in the same feature space and are differently distributed, standard classifiers cannot perform well (Pan and Yang, 2010). This might be the case for the given data if all dealers would change all sales personnel and use the system in a completely new way. For other datasets, this might also be the case if, for example, no data for some regions or products are available. To make predictions outside of the training data, there are ways which address the change in data distribution, such as transfer learning.

Transfer learning is a subfield of ML which tries to solve the question of how a classifier is able to generalise from a source to a target domain (Kouw, 2018). There are different solutions for how to use training data which does not come from the same feature space, and might be differently distributed, to predict future targets. To use the DCBM model for such data, it is necessary to make use of transfer learning techniques to obtain accurate predictions. Transfer learning was used for this dataset to initialise the ANN with older data from times when the system was used in a slightly different way and not in all regions. This did not improve the final results, so it was not used in the final model as there was no significant improvement, and the potential exists to add additional bias and issues to the model.

The results of the DCBM model show better forecasting accuracy mainly through a closer interaction with the end customer via the dealer's sales personnel. As Figure 4.6 shows, the data used is close to the customer but was not directly taken from the customer. Instead, information is collected through the dealer's staff, who interact with the customers and afterwards store the gathered information into the CRM system, where the arrows represent the data flow for the current approach and a possible future approach.

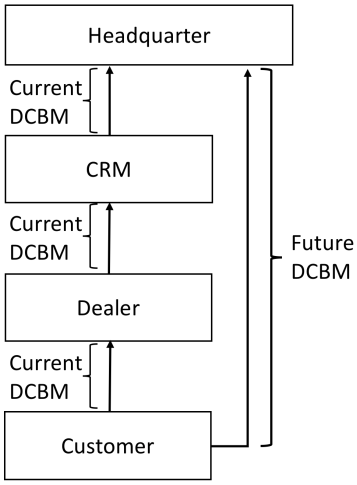

FIGURE 4.6: Comparison of current (left) and potential future (right) data gathering of the DCBM model

However, there are several potential downfalls with this approach as it is not ensured that the dealer's staff input all gathered information into the CRM system, and also different dealers all over the world use the system in various ways. To get direct customer data, it would be necessary to gather live data directly from the consumer. Therefore, a possible alternative would be contact with the customer directly from the headquarters or through other personal communication such as email or user-friendly options like an application on smartphones, which create more touch points over time with the customer. This would enhance the DCBM model to make even better predictions which could also include more personalised predictions such as preferred customisation of the product (e.g. colour or trim of the vehicle). This would not only help to forecast what and when the customer will purchase, but it could also assist in not losing the customer to a competitor, which might occur due to a missed opportunity to sell a new or revised product which the customer would not be aware of by themselves or through the dealer. With a closer connection to the customer, privacy of the customer's data must always be kept in mind to apply with the law (GDPR) and also with ethical constraints which might occur by tracking all interactions between the customer and the company.

4.7 Summary

This chapter introduced a new approach to cluster demand data into three clusters with dynamic boundaries. The DCBM gathers global sales pipeline data to build a short-term sales forecast. Evaluating its accuracy based on a dataset from RRMC shows that the prediction of future sales for the next three months is improved compared to a regular Markov transition model. The boundaries between the clusters depend on a SARIMA forecast to include the seasonal aspect of sales throughout the year. The new model can support short-term planning, supporting regional and product-specific forecasting to steer business activities to achieve their targets and remain profitable. The DCBM model is further used in Chapter 6 to model the residuals of the PLC model introduced in Chapter 5.

Chapter 5

Product Lifecycle Detrending

A new approach developed in this chapter is based on PLCs (introduced in Section 5.1), and Section 5.2 explains how PLC information can be used to improve sales forecasting. The parameters are estimated using a new ML approach (explained further in Section 5.3), and the improvements are outlined in Section 5.4, based on an application using car sales data from RRMC. Results of the proposed approach are evaluated in Section 5.5, and implications and future improvements are discussed in Section 5.6. For statistical models like ARIMA models, there is no extension, to my knowledge, which includes PLC into the prediction based on an ML estimation of its future sales. For that reason, an overview about time series forecasting with a focus on ARIMA models is presented in Section 3.1, an introduction to neural networks in Section 3.2, and decision trees in Section 3.3.

5.1 Product Lifecycle

To include domain knowledge in small-sales time series datasets, the new PLC detrending approach is introduced in this chapter. It is based on the PLCs which every manufacturer's products go through. Figure 5.1 depicts this process over time. After a product idea goes through research, development, production, and market rollout, it is in the introduction phase. If the product is successful, sales increase in the second growth phase. When the product is widely available on the market and sales stop increasing, the product is in the maturity stage. The demand for the product eventually declines, and the product reaches its last phase: decline (Vernon, 1966).

If the product is successful or the manufacturer sees it becoming more successful with improvements, a new product will replace the old one, which restarts the first phase. The restarts of PLC curves result in an up and down movement in sales for a particular product over time. The timeframe of a PLC varies and depends on product, market,

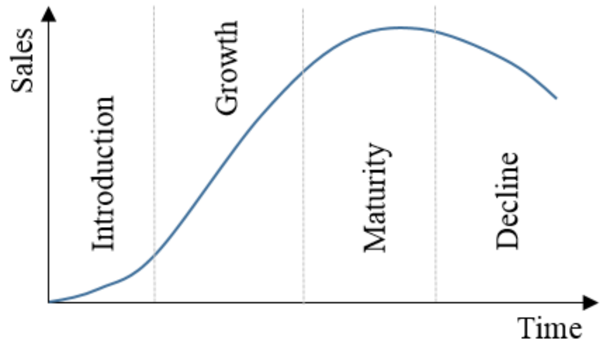

FIGURE 5.1: PLC curve over time

and industry aspects (Meyer, 1997). If a manufacturer produces more than one product, this information is hard to include and filter out in classic time series approaches like ARIMA models.

5.2 Sales Detrending

This section introduces a new way to improve the forecasting accuracy of time series models using PLC information. Bringing a product to market requires a business plan, which must contain not only an estimated production number over time to justify financial costs but also an estimated timeframe of production until a new product launches (Stark, 2015). Both numbers are based on forecasts and have limitations, but the important factor is that they tend to be consistent over time and give a rough estimate about the time and volume of the product. The proposed approach leverages this information and uses it to detrend the time series, consisting of all products offered by the manufacturer.

There is no clear definition of detrending a time series as there are various approaches (Fritts, 1976; Anderson, 1977; Chatfield, 1975). The most common approach is to fit a straight line to the data and then remove it to yield a zero-mean residue. Another commonly used procedure is to take the moving mean of the time series and remove it. This operation needs a predefined time scale, which is often difficult to determine. Regression analysis or Fourier-based filtering are examples of more sophisticated trend extraction methods, which share the problem of justifying their usage as they are based on many assumptions (Wu et al., 2007).

With the new approach introduced herein, every product needs a lifecycle curve to be fitted based on the expected production number and the timeframe of production as well as two shape parameters. Figure 5.2 visualises the effect of the different shape parameters p, q with values of 0.01, 0.05, 0.1, and 0.5 on the Bass curve. m stays constant as it represents the area under the curve. The coefficient of innovation p represents external influences like advertising and is typically very low with values ranging from 0.01 to 0.05. The larger p gets, the steeper is the initial increase in sales in the beginning of the Bass curve. The coefficient of imitation q typically ranges from 0.3 to 0.5 (Bass et al., 1994). With q increasing the height of the Bass curve increases.

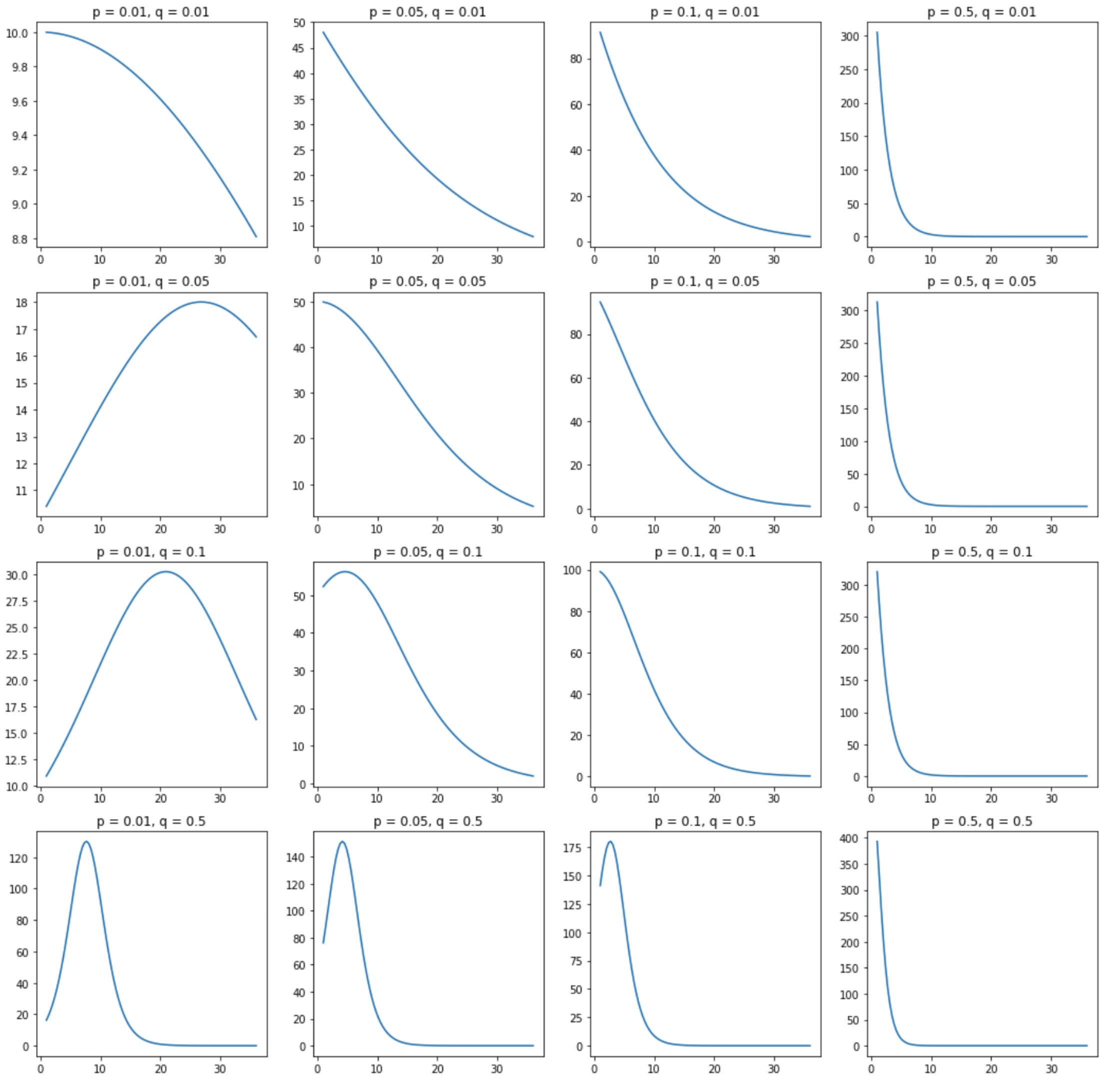

FIGURE 5.2: Different shape parameters and their effects on the Bass curve

By adding all PLC numbers together, the PLC detrending curve is created and, in a second step, is removed from the sales time series history. As this information is also available for a limited time in the future as well, the new approach also adds the lifecycle information to the forecast. There are various ways to fit the PLC curve to the sales data. For instance, in a different approach using the PLC for new product forecasting, Hu et al. (2017) have used three ways to fit a curve to the sales numbers. They compare piecewise linear curves with polynomial approaches and with the Bass diffusion model (Bass, 1969). Since their approach clusters the resulting PLC curves, they choose a piecewise linear model over a polynomial or Bass curve. However, for this research, the Bass diffusion model fits the data best as there are fewer products (five compared to hundreds) on the market which have longer lifecycles (seven to 15 years compared to half a year). The Bass curve was chosen in this work as it is a simple two-parameter model which can be estimated with small amounts of data.

Other approaches could be used with more parameters to fit a more complex model, but the downside is a higher chance for overfitting the data. Piecewise linear curves and polynomial approaches were explored as well but have not delivered better results than the Bass curve. In addition, the estimation of parameters is not as straightforward as the ML Bass curve parameter estimation described later in this section. The Bass diffusion curve draws a smooth diffusion curve, including the slow rise of sales in the beginning and a saturation after the demand increases over time (Massiani and Gohs, 2016).

The Bass diffusion curve is fitted to the available yearly sales numbers from 2003 to today. Yearly sales numbers, instead of monthly, were used because of the large seasonality of car sales, which is not caused by a product's lifecycle but instead is the result of targets within the business. The seasonality for demand is much flatter throughout the year as the main impulse for demand is new model introductions, which vary in time around the world, thus flattening the real demand. There is also seasonality within the demand; for example, convertibles have higher demand in summer, but summer varies around the world. The resulting Bass curve is then split per month for the proposed approach by converting the sales numbers from a yearly to a monthly basis, based on the Bass function. The Bass curve consists of three parameters, m, p, q where m represents the lifetime sales volume, and p, q are shape parameters which represent the coefficient of innovation and imitation. Sales at time T are given by (Bass, 1969) as:

$$S(T) = pm + (q - p)Y(T) - \frac{q}{m}[Y(T)]^2 \tag{5.1}$$

Given the yearly sales numbers, the curve was fitted using a nonlinear least squares fitting. As the cars used for training were already sold, m was calculated by the sum of all past sales. For the years when no sales numbers were available as a new product was launching, the Bass curve was calculated using the parameters predicted by the newly developed approach (explained later in this section). The sales, m, for new products were calculated using lifecycle business plan sales numbers (LCBPSN), which are only available to the business itself. The LCBPSN are used to calculate the business case of a new car over its entire lifecycle and are a good approximation of how many cars will be sold from this model.

Figure 5.3 depicts the monthly sales numbers in dark grey as well as the PLC curves fitted with the Bass diffusion model for two products in blue and orange. The green curve represents the sum of all PLC curves for 10 years and is used later for division of the sales numbers to generate a new time series for forecasting with improved PLC information by car sales:

$$\text{PLC detrended time series} = \frac{\text{Car sales}}{\sum \text{Bass curves}} \tag{5.2}$$

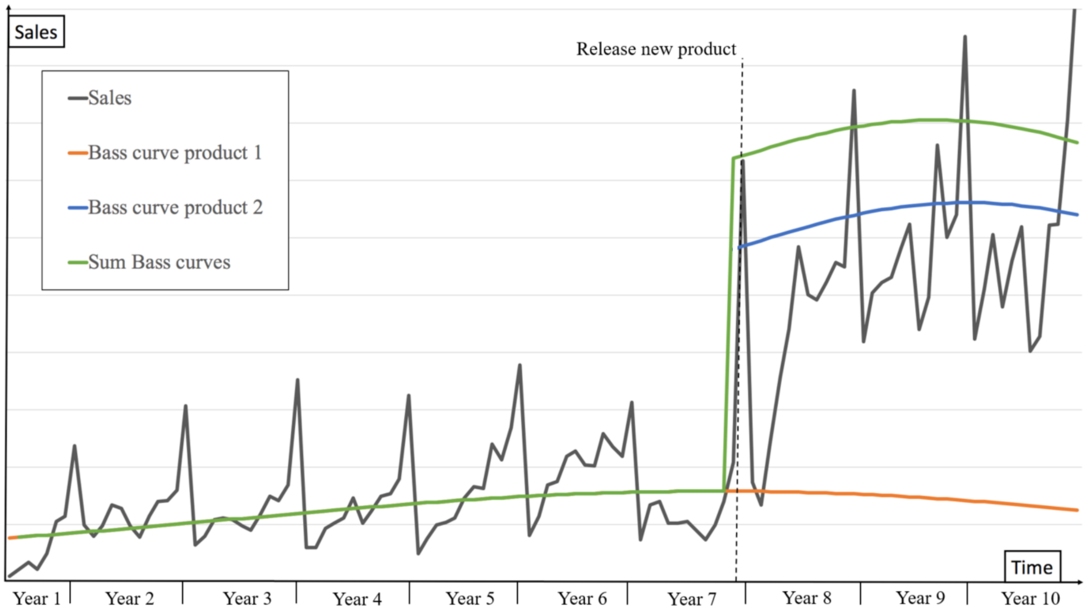

FIGURE 5.3: PLC curves for monthly sales data with total sales/Bass curves on the vertical axis and years on the horizontal axis

5.3 Bass Parameter Estimation

Although the bass curve can be fitted based on assumptions, as described above, there is a new way of estimating the parameters p and q developed within this research. An innovative approach of fitting the Bass curve for new products is proposed. By using sales data from sold products with features like size and weight, it is possible to predict the parameters of the Bass curve for new products by using ML.

The data used in this research is a combination of all car models' sales numbers, Bass parameters calculated for every model based on past sales numbers, and car-specific features from Car Database API Seo and Ltd. (2019) for more than 1000 different car models. The Car Database API features of *power, length, width, height, weight, wheelbase, and displacement* are numeric, and minimal pre-processing was necessary; *coupe* and *drive* features were pre-processed by using one-hot encoding. In total, these features are available for over 28,000 car models. Yearly sales numbers are taken from a dataset available within the company, which consists of all car models sold since 1990. Of these sales numbers, the Bass parameters m, p, q were calculated as described in Section 5.2. Table 5.1 shows an extract from the combined datasets, historic car sales, and car features from Car Database API for two different car models.

Table 5.1: Data extract for Bass curve fitting for two different car models from historic car sales (grey), calculated fields (light grey), and Car Database API (grey)

	Historic car sales						
name	year 1	year 2	year 3	year 4	year 5	year 6	year 7
Suzuki Ertiga	59467	62220	61154	60194	63850	68355	56408
Subaru Legacy	219945	280027	244614	244749	228710	198540	187271
	sum year 1-7	least-squares fitting from year 1-7		Car Database API			
	m	p	q	power	rpm	coupe	length
Suzuki Ertiga	431648	0.055	0.112	105	4500	1	4395
Subaru Legacy	1603856	0.089	0.208	156	5000	4	4685
	Car Database API						
Name	width	height	displacement	fuelsystem	drive	weight	
Suzuki Ertiga	1735	1690	1462	1	2	1180	
Subaru Legacy	1745	1415	2457	2	4	1589	

The resulting dataset consists of over 1,000 car models. The decrease in total car models from 28,000 is due to merging both datasets and removing entries with missing values. The internal dataset containing car model sales numbers only has around 1,400 car models, whereas the Car Database API data is larger because it lists cars with different engines as separate models. By fitting a Bass curve to the yearly sales numbers of products already sold, it is possible to use them as a desired output for the model. As input, the features of the car models described above are used.

For the prediction of p and q, a multi-regressor approach using a random forest regressor was compared to a neural network. The results indicate that the underlying problem can be modelled with a simple neural network with one hidden layer consisting of 20 neurons. Table 5.2 compares the mean absolute error (MAE) for the neural network and the multi-regressor approach with 100 estimators and a maximal depth of 30, where the multi-regressor approach shows slightly better performance.

Table 5.2: MAE comparison for ANN and multi-regressor modelling of Bass curve parameters

	MAE ANN	MAE Multi-regressor
p	0.06	**0.02**
q	0.29	**0.11**

Within the business environment where this model is used, it is also vital to obtain the feature importance for every single feature as this information is useful for stakeholders to understand the model and use the data for the product development of future cars. This could be delivered from the multi-regressor approach (Section 5.4). Table 5.3 summarises the steps of the new PLC approach and gives a broad overview of all stages.

All of the stages explained above are conducted on a real-world application described in further detail in the next section.

Table 5.3: PLC algorithm steps with the needed input for the performed algorithm and its output

Step	Input	Algorithm	Output
1.	Features of over 1,000 cars	Decision tree regression	p and q for Bass curve
2.	Company's car model features	Decision tree regression (1)	p and q for company's car model
3.	p and q (2) + m from business plan	Bass curve	PLC curve (cumulated Bass curves)
4.	Sales time series/ PLC curve (3)	SARIMA model	Forecast of PLC de-trended time series
5.	Forecast (4)	Multiplication with PLC curve (3)	Final forecast

5.4 Application

The proposed approach of PLC detrending is presented for several different time horizon sales forecasts of RRMC which made the data available. The total sales numbers are aggregated through the sales of five different products through 135 dealers worldwide in six regions (shown in Figure 5.4).

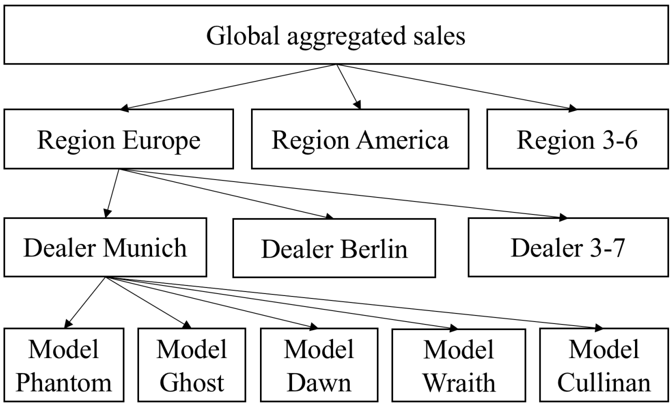

Figure 5.4: Global sales numbers by aggregation over region, dealer and model

To better understand the distribution of the sold products, Figures 5.5 and 5.6 present an overview of the products sold per product and region. Both figures show the distribution of sales per month for the years 2003 until the end of 2018, either split per region in Figure 5.5 or product in Figure 5.6. The regional split shows increased sales over time for all regions as well as the seasonality of the sales always peaking in December. It is also apparent that not all regions share similar behaviour as some markets are bigger than others, and their cultural and climatic differences affect the seasonality in various ways. All regions sell the same products but with a different number of dealers. This is clear in Region 6, with less than 10 dealers, compared to Region 2, with more than three times the number of dealerships. All regions increased dealerships over time and also sold more cars over time; therefore, the trend of sales rose over time.

Figure 5.6 shows four products the manufacturer sold from 2003 to 2019. It is apparent that the overall sales are affected by new products entering the market; especially in the beginning of 2010, sales doubled with the introduction of Product 2. Similar behaviour

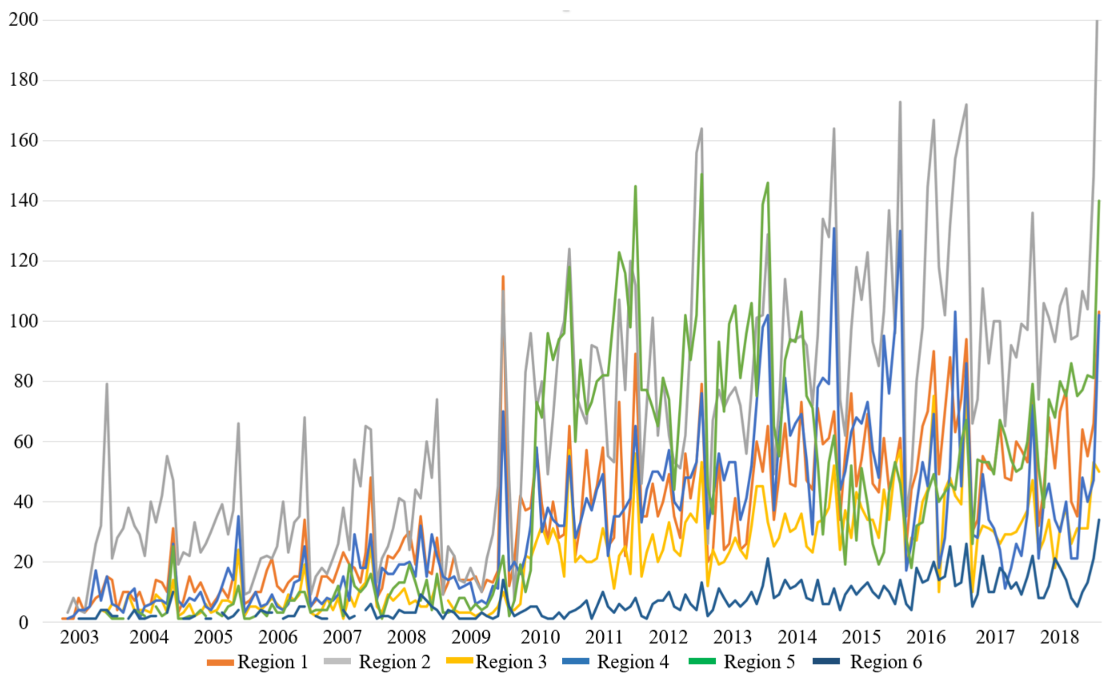

FIGURE 5.5: Monthly car sales split per sales region from 2003 to 2018

is apparent for the other product launches. In addition, the age of a product influences its sales over time as older products tend to sell less after a certain period.

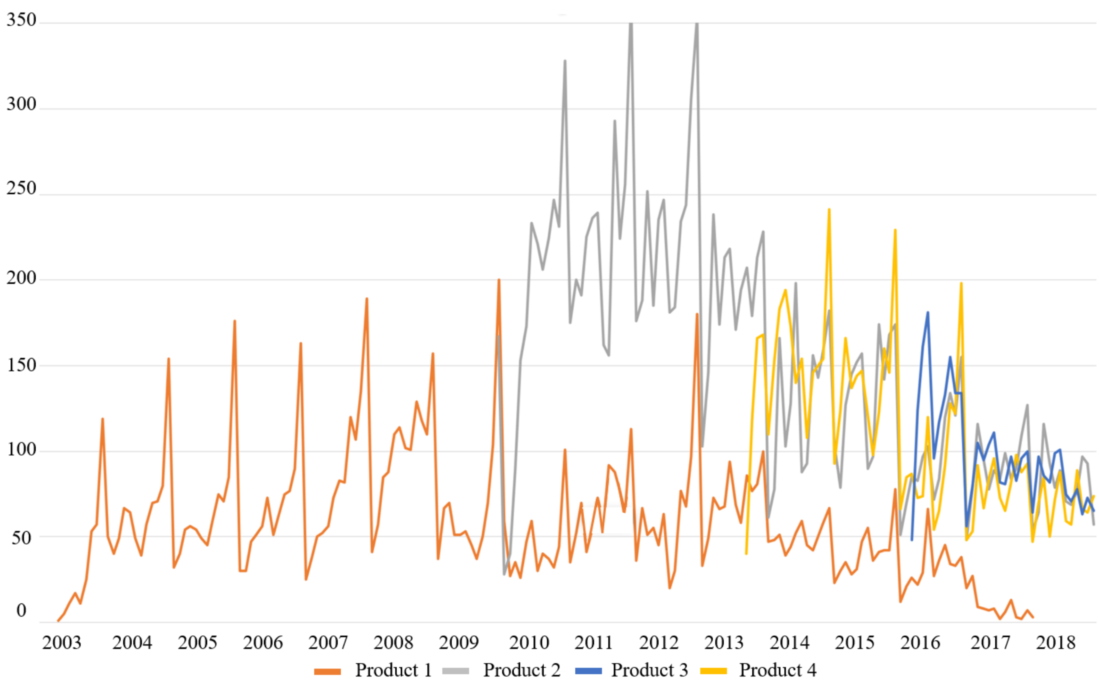

FIGURE 5.6: Monthly car sales split per car model from 2003 to 2018

Following this data overview, the next section further details the results delivered by the new PLC approach for the monthly car sales dataset described above.

5.5 Results

The proposed approach outperforms detrending by differencing, which is a common method to detrend a time series (Solo, 1984). The sales data used ranges from May 2003 to December 2018. To build a reasonable SARIMA model, a minimum of 50 observations was needed, so more than four years' worth of historical monthly data was used (Wei, 1990); hence, only predictions for years after 2007 were considered in the evaluation. The SARIMA model creation itself was completed following the classic Box-Jenkins methodology (Box and Jenkins, 1976).

The model's hyper parameters p, d, q, P, D, and Q were chosen from a grid search between zero and 3 based on the AIC score. The forecasts are compared in Table 5.4 using the RMSE and MAE for the years from 2008 to 2018, where the lower error is highlighted in bold. As the business which generated the sales numbers measures their forecasting accuracy in absolute terms, the MAE was used for comparison. As the MAE does not penalise huge outliers as much as other metrics, the RMSE was used as well, so both measurements are in the same units as the forecasted values of car sales.

As Figure 5.3 shows, the company introduced a new product at the end of Year 7, which led to an increase in sales. Additionally, in 2013, 2015, and 2018, new products were released. Table 5.4 compares the RMSE and MAE of a classic SARIMA forecast with detrending by differencing to the proposed approach, highlighting the lower error in bold. All years indicate an improvement with the proposed PLC detrending approach. For 2010, a large difference is also apparent because it was the year in which the PLC of one product started with the introduction of a new product, resulting in higher sales, which were covered within the new approach. The monthly sales per product are described in the previous section in Figure 5.6, which shows the new product starting at the end of 2009/ beginning of 2010 in grey. For all 11 years, the new approach resulted in an increased accuracy for the MAE as well as the RMSE.

Table 5.4: PLC detrending error compared to normal detrending (NDT) from 2008 to 2018

Error\Year	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018
PLC RMSE	**30.6**	**32.7**	**188.5**	**57.5**	**69.1**	**65.2**	**56.8**	**25.9**	**85.9**	**76.2**	**80.9**
PLC MAE	**22.7**	**25.1**	**167.5**	**49.2**	**49.7**	**55.5**	**50.9**	**22.1**	**73.3**	**68.9**	**61.7**
NDT RMSE	46.1	119.2	2205.2	145.3	124.1	70.9	71.7	46.5	96.8	103.9	192.1
NDT MAE	35.2	77.2	1949.6	117.6	109.7	64.5	62.7	41.9	84.1	89.5	183.6

For all years combined, the improvement of the PLC model for the RMSE is 77% and for the MAE is 78%. Implications for the business include not only increased accuracy in their monthly forecasting, but it also delivered new insights into which features were most predictive within the decision tree regression and how they affect shape parameters. The most predictive features are *length*, *height*, and *weight*, displayed in Table 5.5, where the percentage on the right side indicates how much each feature contributes to decreasing the weighted impurity.

Table 5.5: Feature importance in the decision tree regression

Feature	Feature importance
length	18.0%
height	17.0%
curb weight	14.1%
engine displacement	12.4%
width	12.2%
powerHp	11.8%
powerRpm	7.6%
coupe	4.5%
drive	1.4%
fuel system	1.1%

With the features potentially having a correlation amongst themselves, one must be careful arguing that changing one feature might have a large effect on the overall outcome. Several, like the features referring to the overall size of the car, may correlate in such a way that their effect depends on the changes of other features as well.

For illustration purposes, a simplified version of the chosen tree with a depth of only two is shown in Figure 5.7, which also helps to visualise the effect of certain features, compared to other ML models which lack explainability. The tree shows the splits based on the different features (first line), the error (second line), and also the samples per split and their respective predictions.

Trees like in Table 5.7 can support the adoption of emerging ML algorithms by companies as they support an easy understanding of the decisions taken by the algorithm at every point. Most managers are not experts in the field of ML and struggle to understand more complex models in a short timeframe, so they naturally chose simpler models over complex ones like neural networks. Thereby, the decision is mainly influenced by supporting models they understand compared to those they do not understand and also cannot describe to their supervisors. The tree above is one example of a simplified model which managers may prefer over other models.

Finally, newer body types, such as sport utility vehicles, have different lifecycle curves than traditional sedan models. They result in a steeper increase at the beginning of the PLC, which was also felt in reality. As a side effect of the new model, this information can be used for the future planning of new products and support other business decisions based on the predicted PLC curve.

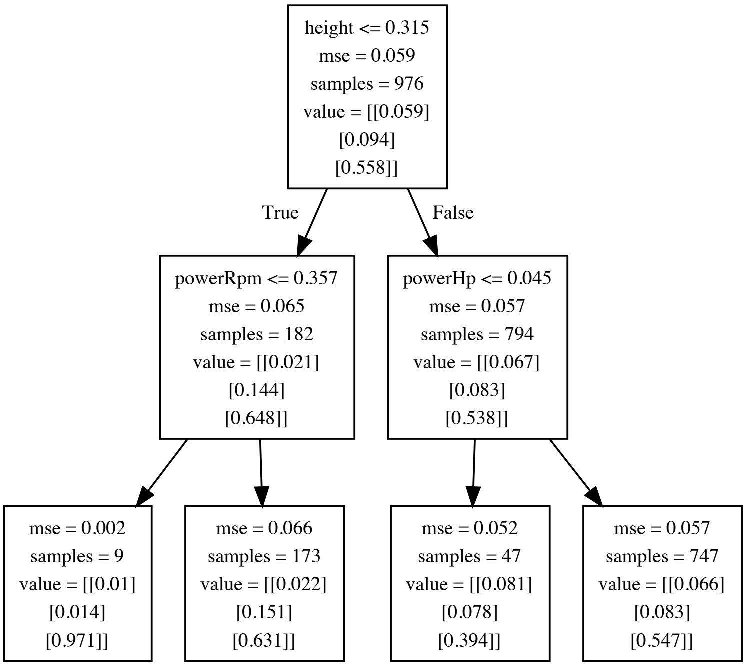

FIGURE 5.7: Simplified decision tree with a depth of two

5.6 Discussion

The proposed approach has the advantage of including information on PLCs in a sales forecast. Other methods also include new product information to improve forecast accuracy. However, these methods are often based on the marketing department's forecasts (Kahn, 2002). Hierarchical procedures, like those proposed by Lenk and Rao (1990) and Neelamegham and Chintagunta (1999), use a Bayesian modelling framework to include various information sources to make new product forecasts but focus more on new products than existing ones, unlike the new PLC approach proposed in this work.

Forecasting every product is also possible, but this approach has two drawbacks. The first is the limited amount of historic data for a new product, and the second is that new products influence other products, so forecasting the total number of sales includes the influence of a new product on other products. In particular, cannibalisation from one product to another product from the same manufacturer is not included, which is an issue which could further improve the accuracy. As products from other manufacturers influence the sales of a product as well, a general PLC curve containing information about the lifecycles of all products in the same market could improve forecast accuracy even more. Typically, start and end dates for competitor's PLCs and their business cases are not publicly available, and therefore including this information was not possible. Other approaches to fit the PLC curve could be considered as well, such as extended Bass diffusion models which include supply constraints, which have not been accounted for so far (Kumar and Swaminathan, 2003).

Overall sales numbers reflected by parameter m were determined from the business case, which makes it difficult for people outside of a business to use the same approach. Therefore, an approach which estimated m using a similar approach to the estimation for p and q was attempted, but the estimate had a large error which is a consequence of the limited available data. Further work is required to establish whether the sales numbers can be estimated reliably from new car features, which would make the approach more widely applicable. This work would need to explore larger feature sets as well as suitable modelling approaches. Although the dataset used car sales data, it should also work for other products if there is enough data about their features. As the required data contains confidential information, it was not possible to access datasets from other industries and products which could open the proposed approach to a wider variety of implementations.

Moreover, one potential drawback of tree models is that their accuracy could be improved by more splits, which at some point can potentially decrease the explainability. Neural networks do not suffer from this issue regarding accuracy; however, in general, they lack explainability. This imposes the problem of a trade-off between accuracy and explainability (Frosst and Hinton, 2017). Therefore, both neural networks and tree models were considered in this work to provide the opportunity to choose the preferred one. From a business point of view, the resulting feature importance given by the decision tree regression is more important than the accuracy; therefore, the decision tree was used. As the data contains confidential features unique to each business, it was not possible to obtain different datasets on which the algorithms could be compared and tested. For that reason, the time series given by the car manufacturer was reversed and then forecasted from the opposite side. The results were even better when comparing the PLC approach to the classic detrending by differencing.

Neural networks can also be used to forecast time series and not only for modelling the shape parameters of the Bass curve. However, they are not well suited for capturing seasonal or trend variations for data which were not pre-processed, but by detrending or de-seasonalisation, their performance could be increased drastically (Zhang and Qi, 2005). This could be another approach to change the used SARIMA model into a neural network model to improve its forecast accuracy even more with the proposed PLC detrending as a pre-processing step. Nonetheless, the problem of de-seasonalisation would not be solved here, so this would need a different pre-processing step. Furthermore, other forecasting methods could be considered as the model is not limited to ARIMA models in general, which makes it easier to implement for other researchers and practitioners in their current way of forecasting.

Although the proposed approach performed better than the present forecasting by RRMC, there is room for improvement, especially in how the code is currently executed. Running the system in a cloud-based environment would decrease the time spent running the code with extracting all the data from different sources. This would allow outsourcing of the work into the cloud, which has proven to be more efficient for data scientists within a company (Aulkemeier et al., 2016). This would not only save time, but it could also be run more often throughout the month to obtain an actual live status from all regions, including updated features of new car models.

5.7 Summary

This chapter introduced a new way of improving the sales forecasting accuracy of time series models using product lifecycle information. Most time series forecasts utilize historical data for forecasting because there is no data available for the future. The PLC model changes this process and utilizes a product's lifecycle-specific data to obtain future information, including product lifecycle changes. Therefore, a decision tree regression was used to predict the shape parameters of the bass curve, which reflects a product's lifecycle over time. This curve is used to detrend the time series to exclude the underlying trend created by the age of a product. The sales forecasts' accuracy was increased for all 11 years of RRMC's forecasts, comparing the newly developed product lifecycle detrending approach to a common detrending by differencing approach in a seasonal autoregressive integrated moving average framework.

Chapter 6

SARIMA-LSTM

As described, sales and demand forecasts are of high value for businesses. Thus, this thesis has developed two new approaches to demand and sales forecasts. Both have improved forecasting accuracy, so it is logical to assess whether combining them can further improve a sales forecast by using PLC information and live sales pipeline data.

This chapter synthesises ML and statistical forecasting methods to increase the accuracy of forecasting. The new approach combines linear and nonlinear forecasting models to create a SARIMA-LSTM model. The linear part is provided by a SARIMA forecast of a detrended PLC time series with improved forecasting abilities due to the included product features. The nonlinear part of the model takes the output of a demand forecasting model, which creates different clusters varying in size, describing the demand. Combining both approaches into one hybrid SARIMA-LSTM model results in improved forecasting accuracy when data is limited by incorporating a seasonal model. The approach was evaluated on a real-world dataset and was shown to improve accuracy. Furthermore, the model was also able to predict the changes in sales due to COVID-19.

The final model should use the three clusters from the DCBM model as well as the residuals from the SARIMA forecast of the PLC model to predict even closer to the actual sales numbers than either model on its own. The residuals are calculated by the actual sales minus the forecasted values generated from the SARIMA-PLC forecast. A reasonable approach to utilise both models is a combination of the PLC model with a long short-term memory neural network (LSTM) which models the residuals of the SARIMA model.

The following section more closely examines the combination and architecture of the SARIMA-LSTM model (Section 6.1). Not only is the accuracy of a model important to business stakeholders, but the model's certainty of the prediction also plays a key role in the adaptation of ML approaches within companies. Therefore, the model uncertainty is approximated within the SARIMA-LSTM framework (explained further in Section 6.2).

The dataset used in the previous work is also applied here and described in Section 6.3, with the results presented in Section 6.4. A discussion of the SARIMA-LSTM model is presented in Section 6.5.

6.1 SARIMA-LSTM Framework

This section describes the new architecture of the SARIMA-LSTM model, which combines two approaches which provide complementary information about the PLC and live sales pipeline data to include demand and sales factors in the forecasts. This is achieved by synthesising both key drivers behind changing trends in the sales numbers by combining the information from the PLC and the DCBM model. The final model uses the three clusters from the DCBM model as well as the residuals from the SARIMA forecast of the PLC model to provide an enhanced prediction of actual sales numbers compared to both models applied individually.

Both approaches have unique information about either demand or product specific features, both of which affect the overall sales numbers; thereby, combining them is crucial for a sales forecast to reflect the current situation between products on the market and the market situation itself. Whilst the SARIMA can capture seasonality, a nonlinear model is needed to include nonlinear patterns into the prediction. Therefore, LSTM was chosen as it can not only incorporate nonlinear patterns but also add memory by including a feedback loop. In the case of an LSTM cell, it is able to create both a short- and long-term memory component which supports predictions on sequential predictions like time series (Gers et al., 1999). The LSTM models the residuals of the PLC model given the three clusters from the DCBM model and afterwards adjusts the SARIMA forecast, with the resulting prediction of the residuals, as depicted in Figure 6.1.

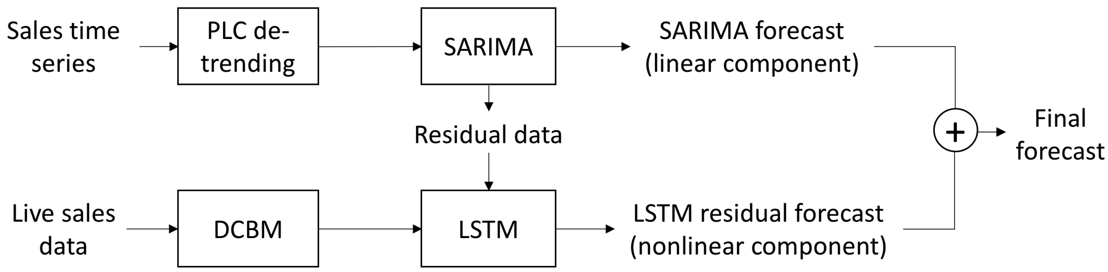

FIGURE 6.1: Newly proposed SARIMA-LSTM pipeline model

The sales time series is detrended by the PLC detrending approach, and afterwards a SARIMA model generates a forecast for the linear part of the newly proposed approach as well as residuals of the forecast compared to the actual sales. The residual data is then used as a target variable for the LSTM part of the model. The input variables for the LSTM model are generated by the DCBM model, resulting in three different clusters. The predicted values from the LSTM are used to adjust the given SARIMA forecast for a final forecast combining live demand data and PLC information. Table 6.1 summarises the data flow within the proposed new SARIMA-LSTM framework.

Step	Input	Algorithm	Output
1.	Sales time series	PLC detrending	Detrended time series
2.	Detrended time series	SARIMA	Sales forecast + residuals
3.	Demand data	DCBM	Three clusters
4.	Residuals + three clusters	LSTM	Forecasted residuals
5.	Sales forecast (2) + forecasted residuals (4)	Addition	Final forecast

TABLE 6.1: SARIMA-LSTM framework

To execute step number four from Table 6.1 to forecast a multivariate time series with an LSTM, the moving window approach is used for input and output. Figure 6.2 describes the multivariate forecasting using moving windows. The red box shows the used input and the green box the output; both are moved whilst training to include the whole time series.

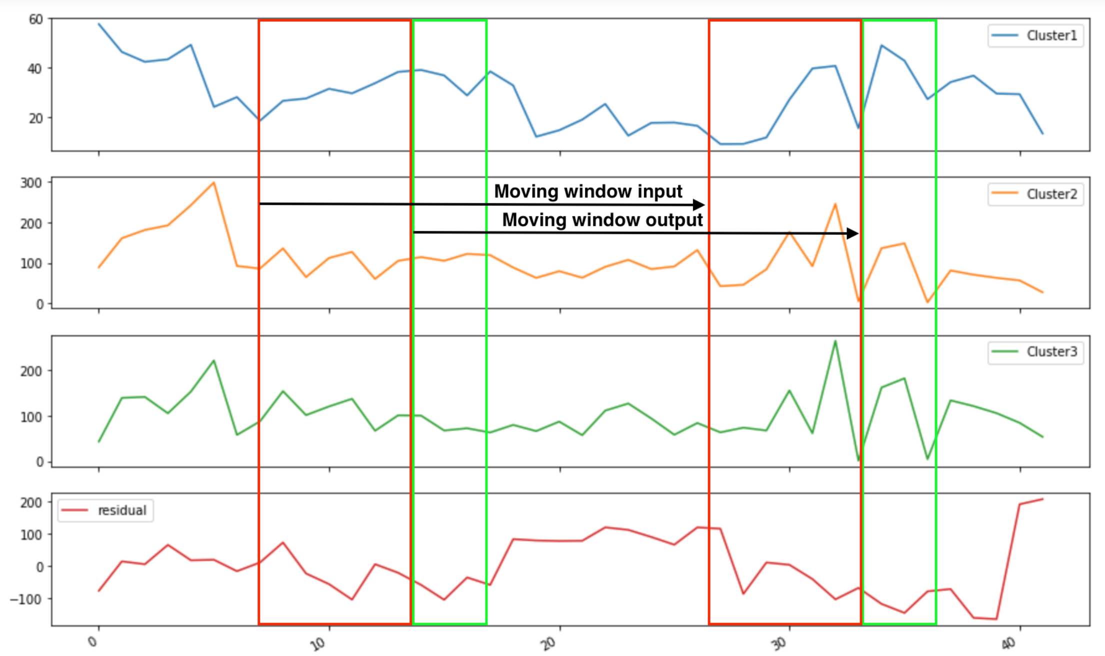

FIGURE 6.2: Moving window forecasting

The size of the window can be defined to include only one timestamp or increased to include more timestamps for the input as well as the output. Moving window forecasting is further described in Section 6.3.

6.2 Model Uncertainty

Section 3.1 discussed how to control model parameters using errors such as the BIC. This is used to specify the fit of the model to the data. The resulting model is then able to make a prediction. However, it can be useful to also have an estimate of the model uncertainty in the prediction. Therefore, various approaches were developed over time to not only make predictions for time series forecasting but to also provide a confidence interval of the model prediction. This can be achieved in different ways, but for neural networks, this process is not as straightforward as for ARIMA or tree models. For this reason, an overview of estimating the model uncertainty for neural networks is provided. This can be done using standard deviation and relating the model uncertainty to it (Gal and Ghahramani, 2016).

One approach to the uncertainty problem employs Bayesian inference by using the posterior distribution $p(W|X,Y)$. Monte Carlo dropout is utilised to approximate the model's uncertainty in the prediction. Dropout randomly drops units and their connections from the neural network during the training process with a certain predefined dropout rate p (Srivastava et al., 2014). The idea here is for a new input x^*; the neural network computes the output with stochastic dropouts at every hidden layer with probability p. This process is repeated N times, and from that we obtain $\hat{y}_1^*, ..., \hat{y}_N^*$ predictions. With that, we can approximate the model uncertainty by the sample variance given by:

$$\hat{Var}(f^W(x^*)) = \frac{1}{N}\sum_{n=1}^{N}(\hat{y}_{(n)}^* - \bar{\hat{y}}^*)^2 \tag{6.1}$$

where $\bar{\hat{y}}^* = \frac{1}{N}\sum_{n=1}^{N}\hat{y}_{(n)}^*$. Choosing the optimal dropout probability p is not straightforward, but in practise, a range around 0.1 tends to be robust (Zhu and Laptev, 2017).

Ideally, the uncertainty interval is as small as possible but still contains all targeted values in between the lower and upper boundaries. To quantify the accuracy of the uncertainty interval, there are many different properties to choose from; the coverage probability and width are chosen here. The prediction interval coverage probability (PICP) describes the total number of points which lie within the uncertainty interval c_i divided by the total number of forecasted points n and is calculated by:

$$PICP = \frac{1}{n}\sum_{i=1}^{n} c_i \tag{6.2}$$

The mean prediction interval width (MPIW) describes the average width of the uncertainty interval over the forecasted time horizon and is given by:

$$MPIW = \frac{1}{n}\sum_{i=1}^{n}(U(X_i) - L(X_i)) \tag{6.3}$$

where $U(X_i)$ and $L(X_i)$ describe the upper and lower boundaries of the uncertainty interval corresponding to the respective sample (Shrestha and Solomatine, 2006).

6.3 Application and Data Pre-processing

The dataset used in the following section was provided by RRMC and reflects their available sales data for all products and all regions worldwide. As the demand data used dates back only to 2016, the data used here are the three clusters from mid-2016 to mid-2020 as well as the residuals of the PLC model from the same timeframe. This observation timeframe adds up to 48 data points in total. An extract of the data used herein is shown in Table 6.2.

Date	Cluster 1	Cluster 2	Cluster 3	Residuals
June 2016	57.67	88.75	42.92	-77.87
July 2016	46.45	160.62	138.31	13.18
August 2016	42.47	180.96	140.43	4.3

Table 6.2: Data extract for SARIMA-LSTM

As the first output in the series cannot be used for the new proposed moving window approach, both approaches have different dataset sizes. The classic moving average version has 48 observations available, with 42 for training and 6 for testing. The following passages explain in more detail how the data is pre-processed for the neural network.

In a first step, the time series is converted into a supervised learning problem of one month as input and one as output. This is done for all three clusters as well as the residuals and is presented in Table 6.3. The three clusters and their respective residuals from the previous month (t-1) are used to predict the current month(t) clusters and residuals. With the shift of the moving window approach, the input from month two becomes the output of month one as the past values are used to forecast the current values.

Input

Month	Cluster 1 (t-1)	Cluster 2 (t-1)	Cluster 3 (t-1)	Residual(t-1)
1	57.67	88.76	42.93	-77.87
2	46.46	160.63	138.32	13.19
3	42.47	180.96	140.44	4.30

Output

Month	Cluster 1 (t)	Cluster 2 (t)	Cluster 3 (t)	Residual (t)
1	46.46	160.63	138.32	13.19
2	42.47	180.96	140.44	4.30
3	43.49	192.42	104.73	64.29

Table 6.3: Conversion of time series data into supervised learning problem

In the previous case, it was chosen to take one timestamp in the past (t-1) as an input and the current timestamp as output (t). This could be changed to include more timestamps in the past and future, depending on the application and available data. As the dataset has only a limited amount of data, the windows were chosen to be as small as possible as with an increasing size of the moving window, observations are lost due to the shift of observations into the window. If more data were available, the moving window size would have been increased and tested for performance increase. Moreover, the features have different value ranges, which can introduce bias for neural networks. For this reason, the data is scaled to similar range values using the MinMaxScaler from sklearn in Python (Pedregosa et al., 2011). MinMaxScaler scales each feature individually so it is in the range $[-1, 1]$ in order to improve the neural network performance. Scaling is important when training neural networks as unscaled data can lead to a slow or unstable learning process or result in exploding gradients for regression problems (Bishop, 1995). Scaling was used to transform the input and output of the given dataset where -1 and 1 were used, although other intervals could be chosen, zero to 1.

As a next step, the scaled dataset was split into a training and test set (87.5%:12.5%) before training up the LSTM architecture. The LSTM neural network was chosen because of its ability to include a feedback loop which serves as a kind of memory as it belongs to the class of recurrent neural networks. The LSTM extends the abilities by including a short- as well as long-term memory part in its model. This is of special importance for time series forecasting problems (Gers et al., 1999; Greff et al., 2017). The model is sequential, containing one LSTM input layer with seven neurons followed by two hidden LSTM layers consisting of 14 neurons. A dense layer was used as an output layer consisting of one neuron, providing the model output. The loss function used was the MAE, and the optimiser used was the Adam optimiser (Kingma and Ba, 2014). The application was programmed using Python 3.5, utilising the Keras library for the LSTM models and the Statsmodels package for the SARIMA models.

To validate the performance of the proposed framework, the next section evaluates the results of the SARIMA-LSTM model on the real-world car sales dataset.

6.4 Results

Depending on the business objective, different time horizons might be of varying importance. For instance, forecasting can be done by predicting one step ahead or multiple steps ahead if the aim is to predict more steps into the future (Boné and Crucianu, 2002). This section evaluates the performance of the SARIMA-LSTM framework for the application explained in the previous section for 12 months using multistep ahead forecasts. Twelve months was chosen as it is the most important timeframe from a business perspective to plan a full year ahead. However, it can also easily be adapted to other forecasting horizons or throughout-the-year forecasts. A broader view into the future supports different purposes such as production planning, sales channel development, or product development. For that reason, the SARIMA-LSTM framework was used for multistep ahead forecasting.

To create a multi-step forecast, one can rely on one-step forecasts iterated for one step after another, always including the latest prediction into the dataset or directly forecasting several steps into the future in one prediction (Marcellino et al., 2006). The model used here predicts multiple steps ahead by iterating over one-step ahead forecasts for a full 12-month ahead prediction, where the model only received data up until the previous December. This could be changed to a 12-month forecast without iterating over one-step forecasts if more data were available, which was not the case for the dataset used here.

The results of the proposed SARIMA-LSTM model are compared to a SARIMA forecast with normal detrending by differencing (SARIMA NDT) and the already proposed SARIMA PLC detrended approach (SARIMA PLC). All models are compared using the RMSE and MAE as evaluation metrics, shown in the following table.

Month	Sales	SARIMA NDT	SARIMA PLC	SARIMA-LSTM
January	289	250.20	312.20	329.90
February	343	287.90	373.84	333.48
March	415	399.28	404.16	396.33
April	365	359.45	363.59	344.46
May	414	405.40	433.62	421.26
June	435	466.26	486.85	456.13
July	356	325.47	359.12	312.45
August	342	321.11	363.16	360.95
September	420	430.48	470.76	437.01
October	414	439.35	483.49	449.53
November	516	488.60	551.45	513.82
December	758	652.85	775.86	776.58
	RMSE	40.62	34.30	**24.48**
	MAE	31.24	27.97	**21.15**

TABLE 6.4: RMSE and MAE comparison of sales to predictions of SARIMA NDT, SARIMA PLC, and SARIMA-LSTM from January to December 2019

The results indicate that the SARIMA model influences about 80% of the forecast, whereas the LSTM part of the new hybrid model accounts for about 20% of the forecast as it only makes smaller corrections to the SARIMA forecast by incorporating live sales data from all over the world into the model. The proposed SARIMA-LSTM model improves forecasting accuracy over SARIMA-PLC and traditional SARIMA-NDT forecasting whilst including demand data from the DCBM model (combining worldwide live data with current PLC information). The SARIMA-LSTM was evaluated on a real dataset and finds a reduction in RMSE and MAE for multistep ahead forecasting over the PLC forecasting process and an overall average reduction for one-step ahead forecasting.

The results show an increase in performance of the SARIMA-LSTM model for the last year, but it is also important to evaluate how the algorithm performs on other parts of the dataset; for that reason, an eightfold cross-validation was performed (further explained in Section 3.4). This was done by splitting the dataset into eight folds with all splits consisting of six months of data. Thereby, the RMSE of the training and test set was compared using the standard deviation (STD) and mean of the eight different folds. Table 6.5 shows the different RMSEs for the training and test sets through the different folds and their STD as well as their mean.

Fold	RMSE training	RMSE test
1	77.4	95.7
2	14.9	80.1
3	55.0	81.1
4	64.9	76.4
5	75.7	78.4
6	81.6	76.0
7	65.7	157.0
8	70.4	140.6
STD	19.9	30.1
Mean	63.2	98.2

Table 6.5: Comparison of RMSE, STD, and mean for the training and test sets of the SARIMA-LSTM model through eight folds

The results show that the training error is lower than the test error, which is typically the case within a reasonable range; this holds true for the STD and mean of the different folds as well. This procedure ensures that the algorithm does not overfit the data in the training process and thus makes worse predictions on the different test sets.

As stated, an important factor in sales forecasting and in forecasting in general is the model certainty in the prediction. For this reason, the SARIMA-LSTM model's uncertainty interval was approximated using dropout (Gal and Ghahramani, 2016). For the model uncertainty, a fourfold cross-validation was used to compare the results. The 48 months of data were split in a way that the test set always consisted of one full year from July to June, and the training set contained the other three years of data. Figure 6.3 illustrates the results of the different uncertainties in the prediction for two standard deviations, which equal 95% certainty over 12 months for four different folds, or four full years. The numbers shown reflect the error band which represents the certainty in the prediction of sales numbers per month. The results of the LSTM model predictions are shown as well as the lower and upper boundaries of the uncertainty interval from the LSTM prediction and the actual residuals from the SARIMA-PLC prediction.

The model successfully captures the trends of the data, and the uncertainty band is accurate in bounding the confidence. The few cases where the prediction just pushes outside the confidence interval coincide with production line problems; this was especially the case for the peaks in month five, when a new product was launched, and months 45 to 47, when a global pandemic affected sales and the production plant was closed for six weeks. To account for these problems, the model would need to include data from the production to determine if production constraints occur.

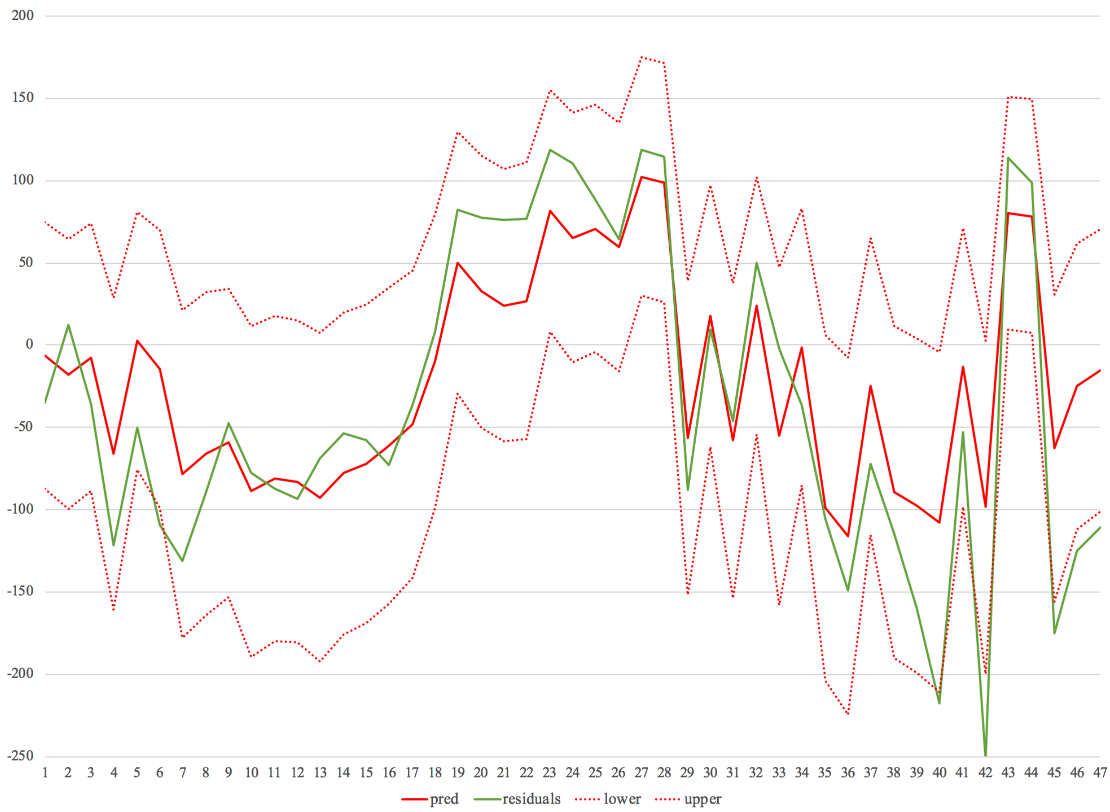

FIGURE 6.3: LSTM prediction (red) with uncertainty interval (dotted red) of SARIMA residuals (green) for four years with the residuals on the vertical axis and the months on the horizontal axis

Due to the pandemic, the production plant of RRMC shut down from mid-March to the beginning of May, which resulted in stopping production for new cars, so dealers could only sell cars they had in stock. They could also only sell cars when they were open, which was not always the case during the lockdown in different countries for various timeframes. This resulted in all-time low sales numbers, especially for April, when the model was able to predict nearly half of the shortcomings in sales. If the overall production numbers were included, this would have probably increased the model's prediction accuracy as no new cars were available for sale. This data was not available for this work and is also hard to obtain as production constraints often occur without any lead time.

The following figure (6.4) illustrates the model uncertainty and how it matches with the predictions using a scatter plot. The vertical axis shows the difference between the residuals and the prediction in absolute terms, whereas the horizontal axis reflects the uncertainty calculated by the difference of the upper boundary of the prediction and the prediction itself. Everything below the blue line from the bottom left to top right corresponds to where the residual error falls outside of the uncertainty interval, and everything above the blue line corresponds to predictions within the uncertainty interval.

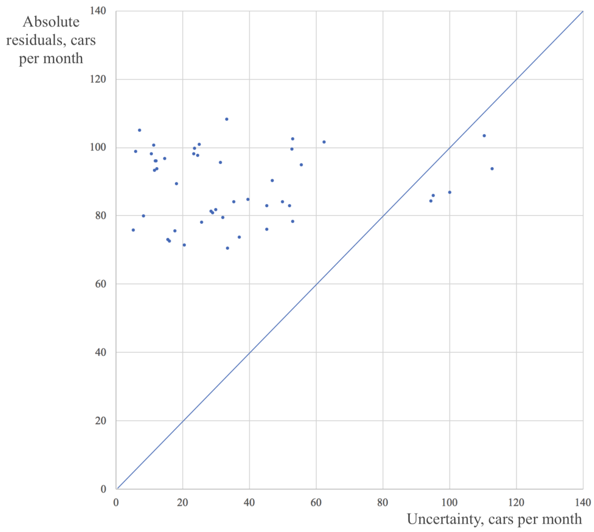

FIGURE 6.4: Absolute residuals on the vertical axis compared to uncertainty on the horizontal axis in cars per month

Data points outside of the uncertainty interval are close to the interval with the exception of one point, also due to production problems which the model was not able to capture as well as no production at all due to COVID-19. Although the model could not fully predict the massive drop in sales due to COVID-19 and the closed dealers and production stoppage, it did predict the underlying trend in sales throughout this pandemic. As the model was not trained on production shortages or global crises, such the one the world faced in 2020, the overall performance of the model shows that incorporating live sales data on a global level helps to forecast even such dramatic changes in sales.

The SARIMA-LSTM model was used to forecast 12 months ahead from June to July of the consecutive year and showed improved results. To assess whether the data split affected the prediction, the same framework was used again with a full year, including all 12 months from January to December, for three years. This was done to check whether this affects the predictions, and it was found that the model is not sensitive to the timeframe itself.

A common approach to dealing with limited data is to use bootstrapping to determine how resampling affects the predictions and their corresponding uncertainty intervals. Bootstrapping was performed whereby 48 datapoints were chosen at random with sampling, with replacement resulting in a new dataset of 48 data points (one datapoint can be picked twice or more, and some of the initial datapoints might not be in the new dataset). The chosen datapoints were used for training, and the ones not chosen were used for testing (Chernick et al., 2011). This was done 100 times to create different datasets. The results of the bootstrapping are illustrated in Figure 6.5, which shows only the LSTM part of the prediction modelling the residuals, and Figure 6.6, which shows the combined SARIMA-LSTM modelling the sales.

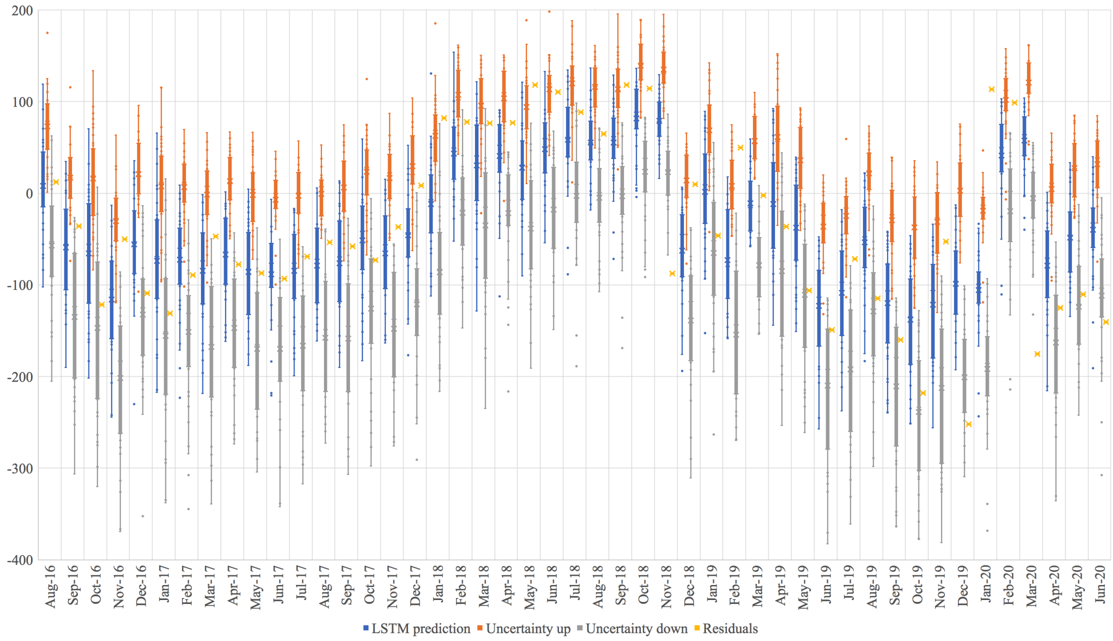

FIGURE 6.5: LSTM results of 100 runs bootstrapping with the LSTM prediction (blue), the uncertainty band (up in orange and down in grey), and the residuals (yellow)

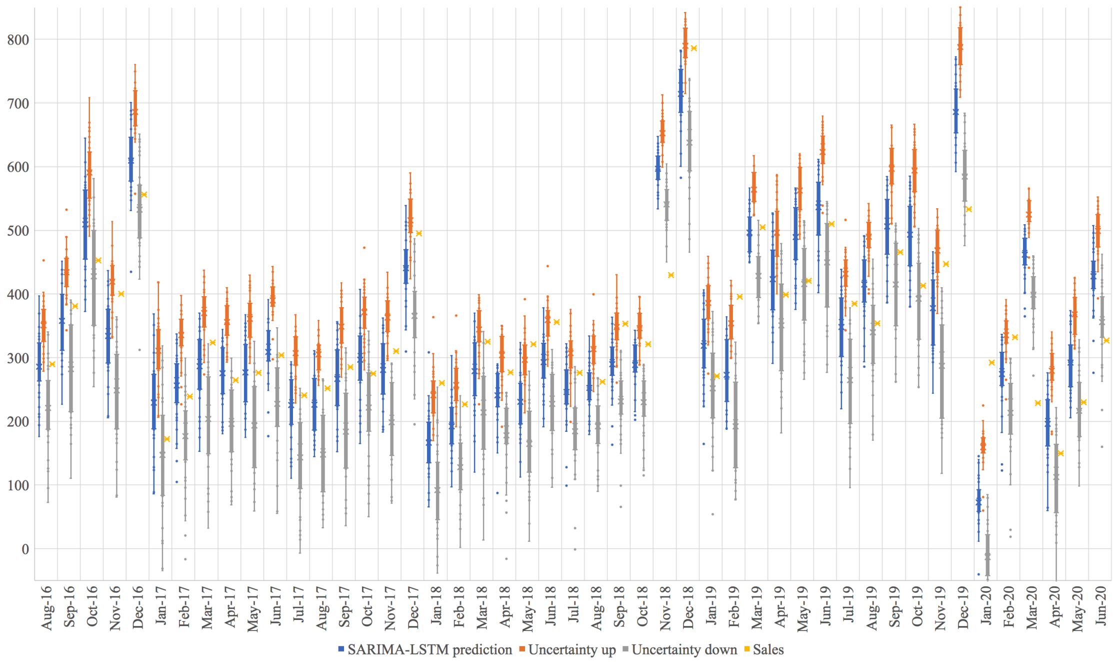

FIGURE 6.6: SARIMA-LSTM results of 100 runs bootstrapping with the combined SARIMA-LSTM prediction (blue), the uncertainty band (up in orange and down in grey), and the real sales (yellow)

The data for training was not in the initial temporal order as it was mixed through the bootstrapping. For time series forecasting, it can be advantageous if the temporal order is not mixed up in order to maximise the accuracy of the model. The results indicate that even through resampling of the data, most of the predicted points lie within the uncertainty interval and the peaks and troughs are captured within the model even for extreme cases such as sales during the outbreak of COVID-19.

The bootstrapping approach was compared with the one from the beginning of this section, where the dataset was divided by a fourfold cross-validation, using the PICP and MPIW. The cross-validation approach resulted in a PICP of 85.1% and MPIW of 177.7, compared to the bootstrapping approach with a PICP of 83% and MPIW of 150.8. The PICP of the cross-validation performed slightly better, which is reasonable as the data was in the right temporal order compared to the bootstrapping. As the bootstrapping approach could utilise more data, it is also reasonable that the uncertainty interval is smaller compared to the cross-validation result.

Through the combination of the DCBM and PLC models depicted in Figure 6.1, it was possible to enhance the ability of the PLC model even further. In summary, the SARIMA-LSTM model increased the forecasting accuracy on a yearly basis for four consecutive years compared to a SARIMA-NDT model and SARIMA-PLC model, as depicted in Table 6.6.

Year	SARIMA-NDT	SARIMA-PLC	SARIMA-LSTM
2017 (last six months)	96.5 (82)	74.9 (68.7)	**30.6 (27.3)**
2018	174.5 (166.4)	68.2 (49.3)	**47.7 (32.1)**
2019	40.6 (31.2)	34.3 (27.9)	**24.5 (21.2)**
2020 (first six months)	119.8 (116.2)	107.7 (82.7)	**83.1 (72.8)**
Mean RMSE (MAE)	107.7 (99)	71.3 (57.2)	**46.5 (38.4)**

Table 6.6: Error comparison of the SARIMA-NDT, SARIMA-PLC, and the SARIMA-LSTM for 2017-2020 using RMSE and MAE (in brackets)

In summary, the RMSE and MAE of the SARIMA-NDT can be improved by using the PLC model for every year from mid-2017 until mid-2020. Furthermore, the SARIMA-LSTM showed even more improvement over the PLC model and, with that, increased the performance, equating it to the initial NDT model. Comparing the results of the RMSE and MAE of all models shows that the SARIMA-NDT has potentially more large outliers in the prediction as the MAE is closer to the RMSE in all cases, whereas for the PLC and SARIMA-LSTM models, the MAE is significantly lower than the RMSE.

6.5 Discussion

In addition to improved accuracy, another advantage of the model is that it is able to include more features into the prediction without much effort if the data is available on a monthly basis for the given time horizon. However, one must be careful as just adding new data does not mean that the accuracy of the forecast is automatically improved. As different features are brought together within the SARIMA-LSTM, it would be possible to include external factors such as economic data to make the forecast more robust for external changes. As an example, this would especially be a benefit during economic crises occurring due to pandemic diseases like the corona virus spreading all over the world in the beginning of 2020. Within the setup of the model, the following economic variables were used without success: the global economic policy uncertainty index, which is a GDP-weighted average of national indices for 20 countries, as well as the crude oil prices. A suggestion for future research would be to also include several economic indicators on a regional level to better reflect economic changes. As under 50 months were used for the whole model, it would also be interesting to find if this would increase the accuracy for a larger dataset.

The dataset had 48 months of observations in total; however, if larger datasets were available, it would be interesting to investigate whether the LSTM model could potentially learn the seasonal aspect of the data and compete with the SARIMA-LSTM model. Nonetheless, such large datasets for monthly sales data are often hard to obtain as most companies do not have them. For RRMC, the SARIMA-LSTM model provides a combination of different data sources which were not juxtaposed before, thus enabling the business to combine different data sources into one prediction. This is an innovative approach as, in the past, their forecasts were mainly judgemental and not automated. The automation and aggregation of the data through CRM systems as well as product-specific sources not only improve the current forecasting accuracy but also save valuable time.

However, the proposed SARIMA-LTSM model also has the drawbacks of both the PLC and DCBM models, and within the current framework, it does not include external car market data sources as this data is not publicly available except from the past sales numbers used to train the PLC model. In addition, including economic variables did not improve the forecasting for the given dataset, which cannot be generalised for other datasets. Therefore, the SARIMA-LSTM framework should be applied to different datasets to determine whether the same results are reproducible on other datasets. This should be possible not only for other car manufacturers but also for different industries where lead times of products are longer than those for daily purchased products like chewing gum or other fast-moving consumer goods.

In general, ML algorithms need a significant amount of data to perform well, but there is no general size to assess what is sufficient (Stephen, 2009). As the used dataset covers 48 months of data, it is nearer the lower boundary of what is too small or too large. Nevertheless, as the SARIMA-LSTM increased forecasting ability, it would be interesting to note how much more accuracy the model would gain with more data.

However, with additional data, the computational resources needed to run the algorithm would increase as well. This was tested to determine how the computational expense would change by doubling the amount of data through appending the original dataset to itself and thus resulting in 96 observations. The model uncertainty approximation using dropout and running several times especially increased the computational cost for a larger dataset size, requiring a linear computational cost rise of $\mathcal{O}(n)$. Nonetheless, as increased sales forecasting saves the business money, and the added required cost for running the algorithms is manageable, the additional time and money are worth it from a business perspective since the savings are much greater. Thus, for the SARIMA-LSTM, the algorithmic scalability is worth the effort if more data becomes available.

Finally, the uncertainty in the prediction reflects the uncertainty of the LSTM part of the model, estimated using dropout as a Bayesian approximation. The SARIMA model's uncertainty is already reflected in the residuals of the LSTM models. In this way, the uncertainty approximated by the LSTM reflects the overall uncertainty of the model as the residuals capture the uncertainty of the SARIMA aspect. As the LSTM models the residuals of the SARIMA prediction, the overall uncertainty is incorporated within this approach.

6.6 Summary

The new approach presented combines linear and nonlinear forecasting elements within one SARIMA-LSTM model. The linear part is a SARIMA forecast of a product lifecycle detrended time series with improved forecasting abilities due to the inclusion of product features. The nonlinear part of the model takes the output of a demand-forecasting model that creates different clusters, varying in size, which describe the demand. It has been shown that the SARIMA-LSTM framework can improve upon the individual performances of SARIMA and LSTM models and exploit the additional domain knowledge provided by the DCBM and PLC approaches. Combining both approaches into a hybrid SARIMA-LSTM model resulted in improved forecasting accuracy, evaluated on RRMC's dataset. It was able to improve the forecasting accuracy and was robust even in the extreme period of the COVID-19 pandemic.

Chapter 7

Conclusions and Future Research

7.1 Conclusions

In most retail industries, sales forecasts are important for many reasons, from budgeting to production planning. This research has thus developed new approaches to improve the sales and demand forecasting by using global live data as well as PLC information. The results indicate that forecasting accuracy can be improved on a regional and product-specific level. The two proposed approaches are new ways of implementing a Markov transition model by clustering the data beforehand to boost performance and using a PLC detrending approach, where the launch of new products and the age of the product influence the forecast. This research also combined both proposed models into one SARIMA-LSTM model, which further improved forecasting accuracy.

For a company selling products globally, gaining accurate live forecasts from all countries for different products is difficult, but it is important for reaching business targets. Using sales pipeline data, which includes stage transitions reflecting the buying process, the DCBM performed better than a common Markov transition approach. Within this approach, a new way of clustering sales data was proposed, which creates three clusters, each differentiated by the customers' likelihood to buy a product. The novelty of this approach is that the boundary between the first and second cluster varies based on a time series forecast of conversion, which includes seasonal variations in buying behaviour, present in most sales businesses which receive sales targets from top management.

The first research question regarding whether ML can enhance forecasting and demand estimation when the external environment undergoes dynamic, rapid, and unforeseen changes can indeed be answered positively as this can be predicted with the DCBM model. Through evaluation, the DCBM approach demonstrated an increased performance over a traditional Markov model on the car sales dataset.

The second research question regarded how PLC information can be used to improve traditional sales forecasting methods, and a new PLC approach was proposed. Moreover, classic time series forecasting provides strong results for most cases, but it has one major drawback: only considering historic data in the forecast. The new PLC detrending approach improves sales forecasting by using information about the products' lifecycles, which is also available for the future, thereby increasing forecast accuracy. For years when new products are launched and old products run out of production, the PLC approach outperformed a SARIMA forecast with common detrending by differencing. For a 12-month sales forecast for RRMC, all 11 years were improved. The PLC approach detrends the time series by dividing it through the PLC curve, which represents a product's lifecycle, which was fitted by several Bass diffusion curves. The Bass model parameters were estimated by a new ML framework. Thus far, research in that area has focussed on forecasting sales for completely new products on the market, where no data was available. The main difference with the new approach is that the forecasting of existing products can also be improved. This can be applied by any business selling luxury and expensive goods as the buying process is usually longer for these items than for cheaper, fast-selling products.

Given the complementary strengths of the DCBM and PLC models, it was logical to find a way to combine both. The solution was to create a SARIMA-LSTM to model the linear and nonlinear parts of both models and thereby further increase forecasting accuracy. Not only was the accuracy improved, but it also made it possible to include other external variables which influence the forecast, such as economic factors.

This work further found that the hyperparameter tuning of the various models can result in different findings; however, the focus was not on tuning hyperparameters or the architecture of the models. Instead, the focus was on the methods behind the models. Thus, for different datasets or problems, it might be useful to further explore other architectures and their parameter estimation to optimise results.

On a field trip to different countries, the live sales data was analysed with the help of local dealers, which supported previous assumptions made from the data and provided new insights into how RRMC's data is collected. Feedback was used to better adapt the model to the problems occurring in real data applications. The findings demonstrate, in theory as well as the actual business case, that the forecasting was improved and supports the company's planning for upcoming months and years.

Nevertheless, whilst this work was being written, a global pandemic started and spread all over the world, affecting sales in all markets, ranging from cars to nearly all consumer goods (Kim, 2020). Companies could not sell cars whilst their dealers or sales outlets were closed, so the DCBM model was able to predict that sales would drop (dealers still had some enquiries during this time, which lead to several sales). The SARIMA-LSTM model predicted the change in sales regarding the direction of overall sales per region and model. The results show that the model not only reflects changes in sales but could predict a large drop in sales during the outbreak of COVID-19, which was not built into the model. This finding illustrates the power of ML for sales forecasting if the right input is selected concerning the levers which drive sales and demand. SARIMA-LSTM makes accurate sample predictions, indicating the robustness of the architecture combined with the chosen input of PLC and global live sales pipeline data.

The initial research questions focussed in particular on improving sales and demand forecasting with the help of ML. These questions have been answered by showing how ML can enhance forecasting and demand estimations when the external environment undergoes dynamic, rapid, and unforeseen changes as well as how PLC information can be used to improve traditional sales forecasting methods. Both questions were investigated, and evidence from practical experiments on real data from RRMC Motor Cars showed improved results compared to their current forecasting methods. The proposed models were also compared to current statistical and ML approaches.

The DCBM model improved clustering the demand data and making predictions from it, which was later used in the SARIMA-LSTM model; however, the DCBM model also improved forecasting on its own. The PLC model included new data into classical statistical models by using ML techniques which generate product-specific information, already available for the future, into a forecast. The final contribution of this work is the SARIMA-LSTM model, which combines both the DCBM and PLC models and improves forecasting in general, as well as forecasting the direction of sales during the COVID-19 pandemic within the uncertainty band of the model. Predicting sales affected by this pandemic is an extreme case which clarifies the model's ability to predict sales with the external environment undergoing dynamic, rapid, and unforeseen changes.

7.2 Future Research

The proposed models in this research all showed improved accuracy on the given car sales dataset. The approaches would also benefit from comparison to other datasets, but such data contains confidential company information, so this was not possible to obtain. However, it would be insightful for future research to apply the approaches introduced in this work to other datasets currently not available.

Moreover, the DCBM and PLC models also have limitations for the existing car sales dataset. For instance, the PLC will most likely not change, but with the introduction of competitors' products or new laws, the sales dynamic could change. Technological leaps, like combustion engine to electric propulsion, could interfere with the past sales behaviour and PLCs. Additionally, the DCBM approach uses live sales data, which is dependent on the end user, in this case, every dealer's sales teams using the system as intended. These limitations are beyond the scope of this research but might be addressed in future studies.

Sa-ngasoongsong et al. (2012) have forecasted sales in the automotive world with similar timeframes (from six to 24 months) as this research but with a focus on the economic indicators and their effect on sales and demand. This current research touched on the effect of economic variables, but these were not used for the proposed models as their impact was not significant on the given dataset. However, especially for the rise of alternative propulsion systems such as electric cars, government subsidies can affect sales and demand more than any promotions or marketing activities by car manufacturers themselves. This will be an important topic in the future of car sales and demand forecasting and thus should be included in future research in that specific area.

Furthermore, Fildes et al. (2019) already raised the issue of changes in retail shopping for store locations as well as the size of the individual stores. Especially in the automotive market, where nearly all cars are currently sold via a dealer network, location is key to a successful sales strategy. The DCBM model could be extended in the future to not only include which regions perform best, but it could also include the location of the potential customers to determine if the distance from their home to the dealer affects the sales outcome and, if so, propose a better location for future dealers. Another point that could be addressed in future research is that the DCBM model is based on a Markov model, which only utilises the current stage to predict future stages. It would be interesting to investigate whether the performance could be improved by using higher-order models which include more than the previous stage in their predictions. To test this hypothesis robustly, a greater quantity of data would be required than what was available for this work.

Another consideration is that much work in the forecasting area focusses on stock keeping units which are hard to sell and remain too long in sales outlets (Özden Gür Ali et al., 2009; Ma et al., 2016). This area of forecasting research was not included in this research but could be an important future research field as many car dealerships struggle with expensive cars in their courtyard which do not sell and tie up large amounts of money which could depreciate over time. This is especially the case if successor models are released by the car manufacturer. Such information could be added through the DCBM model into a company's CRM system to promote already produced cars to other dealerships which would have potential customers for them. As for the given dataset, all dealers are their own legal entity and do not send customers to other dealerships in most cases, so this was not investigated further. However, such a study could be applied to other companies which have a different legal system in order to decrease the time which products stay at the dealerships.

Other important factors are production capacities as well as logistics. Neither aspect was considered in this work as the data was not available or was too complex to include.

However, both are limiting factors which could affect sales on a global level. If there are production shutdowns, or unexpected maintenance is needed on the production lines, sales will be affected by limited supply. Throughout a year, this effect might be flattened, but for specific months, this will affect the forecasts. Nonetheless, obtaining this kind of data is not straightforward as production or logistical problems may occur due to unforeseeable external changes with suppliers or other factors such as climate changes, which can impact production and logistic capabilities, especially for a small manufacturer like RRMC, which has only one production line.

As a last consideration, future data sources should be discussed briefly as they will increase in size and variety. Looking forwards, there will be more communication channels amongst customer, dealer, manufacturer, and everyone in between this supply chain. Currently, most forecasting methods make use of the data of one of these channels to forecast certain outcomes, and this work combined different data sources to improve forecasting.

In the future, there will be more channels such as mobile apps or websites which will gather more data about customer interactions, which might improve the overall knowledge about a customer or potential customer. Due to legal restrictions like GDPR, it is not straightforward to combine the data of different communication channels and to identify specific individuals amongst different datasets and collect their information under one customer ID. Therefore, it will be increasingly important to establish communication channels with customers in a way that makes it possible to identify them as a specific user (whilst respecting their fundamental right to anonymity if they do not want to be identified and traced). This creates a new challenge for businesses from a technical perspective but also gives them the opportunity to learn more about their customers and their behaviour.

The implementation of new technologies and communication channels can gather data to make better future predictions not only about sales but also for other important areas such as predictive maintenance. Different communication channels such as WhatsApp and WeChat or mobile applications like BMW Connected or Mercedes Me are just the start of new ways to generate more data. Live production data from the production machines could also be included in forecasts and predictions, supporting better decision making for businesses. These channels generate new touchpoints between customers and companies, and future forecasting models could benefit from the data generated from them. Therefore, an important future research topic will be how to include all different data streams into one prediction, and the proposed SARIMA-LSTM model could be used as a starting point.

References

Thomas L. Ainscough and Jay E. Aronson. An empirical investigation and comparison of neural networks and regression for scanner data analysis. *Journal of Retailing and Consumer Services*, 6(4):205 – 217, 1999. ISSN 0969-6989.

Per Kragh Andersen, Ornulf Borgan, Richard D. Gill, and Niels Keiding. *Statistical Models Based on Counting Processes*. Springer-Verlag New York, 1993.

P.K. Andersen and N. Keiding. Multi-state models for event history analysis. *Statistical Methods in Medical Research*, 2002.

O. D. Anderson. The box-jenkins approach to time series analysis. *RAIRO - Operations Research - Recherche Opérationnelle*, 11(1):3–29, 1977.

J.S. Armstrong. *Principles of Forecasting: A Handbook for Researchers and Practitioners*. Kluwer, 2001.

Nari Sivanandam Arunraj and Diane Ahrens. A hybrid seasonal autoregressive integrated moving average and quantile regression for daily food sales forecasting. *International Journal of Production Economics*, 170(PA):321–335, 2015.

Fabian Aulkemeier, Roman Daukuls, Maria-Eugenia Iacob, Jaap Boter, Jos van Hillegersberg, and Sander de Leeuw. Sales forecasting as a service. In *Proceedings of the 18th International Conference on Enterprise Information Systems*, ICEIS 2016, page 345–352. SCITEPRESS - Science and Technology Publications, Lda, 2016. ISBN 9789897581878.

Frank M. Bass. A new product growth for model consumer durables. *Management Science*, 15(5):215–227, 1969. ISSN 00251909, 15265501.

Frank M. Bass, Trichy V. Krishnan, and Dipak C. Jain. Why the bass model fits without decision variables. *Marketing Science*, 13(3):203–223, 1994.

Sugato Basu, Arindam Banerjee, and Raymond J. Mooney. Semi-supervised clustering by seeding. In *Proceedings of the Nineteenth International Conference on Machine Learning*. Morgan Kaufmann Publishers Inc., 2002.

Christoph Bergmeir and José M. Benítez. On the use of cross-validation for time series predictor evaluation. *Information Sciences*, 191:192 – 213, 2012. ISSN 0020-0255. Data Mining for Software Trustworthiness.

P. Berkhin. *A Survey of Clustering Data Mining Techniques*, pages 25–71. Springer Berlin Heidelberg, 2006.

Fernando Bernstein and Awi Federgruen. Decentralized supply chains with competing retailers under demand uncertainty. *Management Science*, 51(1):18–29, 2005.

Christopher M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, Inc., New York, NY, USA, 1995.

R. Boné and M. Crucianu. Multi-step-ahead prediction with neural networks: a review. In *9emes rencontres internationales: Approches Connexionnistes en Sciences*, pages 97–106, 2002.

Jan Bouwens and Peter Kroos. Target ratcheting and effort reduction. *Journal of Accounting and Economics*, 51:171–185, 05 2010.

George E. P. Box and Gwilym M. Jenkins. *Time series analysis, forecasting and control*. Holden-Day, 1976.

L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone. *Classification and regression trees*. Routledge, New York, NY, USA, 1984.

Sung-Hyuk Cha. Comprehensive survey on distance/similarity measures between probability density functions. *International Journal of Mathematical models and Methods in Applied Sciences*, 1(4):300–307, 2007.

T. Chai and R. R. Draxler. Root mean square error (rmse) or mean absolute error (mae)? – arguments against avoiding rmse in the literature. *Geoscientific Model Development*, 7(3):1247–1250, 2014.

C. Chatfield. *The analysis of time series: Theory and practice*. Chapman and Hall, 1975.

Chenyi Chen, Ari Seff, Alain L. Kornhauser, and Jianxiong Xiao. Deepdriving: Learning affordance for direct perception in autonomous driving. *CoRR*, abs/1505.00256, 2015.

Michael R. Chernick, Wenceslao González-Manteiga, Rosa M. Crujeiras, and Erniel B. Barrios. *Bootstrap Methods*, pages 169–174. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. ISBN 978-3-642-04898-2.

Chin Long Chiang. *Introduction to stochastic processes in biostatistics*. Wiley, New York, 1968.

Ching-Wu Chu and Guoqiang Peter Zhang. A comparative study of linear and nonlinear models for aggregate retail sales forecasting. *International Journal of Production Economics*, 86(3):217 – 231, 2003. ISSN 0925-5273.

D Collett. *Modelling Survival Data in Medical Research*. Chapman and Hall, 2014.

D. Cox and N Reid. *Analysis of Survival Data*. Routledge, 1984.

Douglas J. Dalrymple. Sales forecasting practices: Results from a united states survey. *International Journal of Forecasting*, 3(3):379 – 391, 1987. ISSN 0169-2070. Special Issue on: Forecasting in Marketing.

Byron J. Dangerfield and John S. Morris. Top-down or bottom-up: Aggregate versus disaggregate extrapolations. *International Journal of Forecasting*, 8(2):233–241, 1992.

Per-Erik Danielsson. Euclidean distance mapping. *Computer Graphics and Image Processing*, 14(3):227 – 248, 1980. ISSN 0146-664X.

Donna F. Davis and John T. Mentzer. Organizational factors in sales forecasting management. *International Journal of Forecasting*, 23(3):475 – 495, 2007. ISSN 0169-2070.

Alekh Dwivedi, Maheshwari Niranjan, and Kalicharan Sahu. A business intelligence technique for forecasting the automobile sales using adaptive intelligent systems (anfis and ann). *International Journal of Computer Applications*, 74:7–13, 07 2013.

F. Robert Dwyer, Paul H. Schurr, and Sejo Oh. Developing buyer-seller relationships. *Journal of Marketing*, 51(2):11–27, 1987. ISSN 00222429.

David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back propagating errors. *Nature*, 323:533–536, 10 1986.

Bob Edmundson, Michael Lawrence, and Marcus O'Connor. The use of non-time series information in sales forecasting: A case study. *Journal of Forecasting*, 7(3):201–211, 1988.

Dean Fantazzini and Zhamal Toktamysova. Forecasting german car sales using google data and multivariate models. *International Journal of Production Economics*, 170: 97 – 135, 2015. ISSN 0925-5273.

Laurene Fausett. *Fundamentals of Neural Networks: Architectures, Algorithms, and Applications*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1994.

Robert Fildes, Shaohui Ma, and Stephan Kolassa. Retail forecasting: Research and practice. *International Journal of Forecasting*, 2019. ISSN 0169-2070.

Donald W Fogarty and Thomas Russell Hoffmann. *Production & inventory management*. Cincinnati : South-Western Publishing Co, 2nd ed edition, 1991.

Uwe Franke, Dariu Gavrila, Steffen Görzig, Frank Lindner, Frank Paetzold, and Christian Wöhler. Autonomous driving goes downtown. *IEEE Intelligent Systems*, 13: 40–48, 11 1998.

H.C. Fritts. *Tree rings and climate*. Academic Press, 1976.

Nicholas Frosst and Geoffrey E. Hinton. Distilling a neural network into a soft decision tree. *CoRR*, abs/1711.09784, 2017.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1050–1059, New York, New York, USA, 20–22 Jun 2016. PMLR.

Crispin Gardiner. *Stochastic Methods*. Springer-Verlag Berlin Heidelberg, 2009.

Alan Garnham. *Artificial Intelligence: An Introduction*. Routledge, 2017.

F.A. Gers, J. Schmidhuber, and F. Cummins. Learning to forget: continual prediction with lstm. *IET Conference Proceedings*, pages 850–855(5), January 1999.

J. P. González, A. M. S. Muñoz San Roque, and E. A. Pérez. Forecasting functional time series with a new hilbertian armax model: Application to electricity price forecasting. *IEEE Transactions on Power Systems*, 33(1):545–556, 2018.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber. Lstm: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, 28(10):2222–2232, 2017.

Y. Grunfeld and Z Griliches. Is aggregation necessarily bad? *Review of Economic Statistics*, 42:1–13, 1960.

M. Gurnani, Y. Korke, P. Shah, S. Udmale, V. Sambhe, and S. Bhirud. Forecasting of sales by using fusion of machine learning techniques. In *2017 International Conference on Data Management, Analytics and Innovation (ICDMAI)*, pages 93–101, Feb 2017.

H. Hagras. Toward human-understandable, explainable ai. *Computer*, 51(9):28–36, 2018.

Dominique M. Hanssens. Order forecasts, retail sales, and the marketing mix for consumer durables. *Journal of Forecasting*, 17(3-4):327–346, 1998.

John A. Hartigan. *Clustering Algorithms*. John Wiley & Sons, Inc., New York, NY, USA, 99th edition, 1975. ISBN 047135645X.

Wenwu He, Zhizhong Wang, and Hui Jiang. Model optimizing and feature selecting for support vector regression in time series forecasting. *Neurocomputing*, 72(1):600–611, 2008. Machine Learning for Signal Processing (MLSP 2006) / Life System Modelling, Simulation, and Bio-inspired Computing (LSMS 2007).

S. L. Ho, M. Xie, and T. N. Goh. A comparative study of neural network and box-jenkins arima modeling in time series prediction. *Comput. Ind. Eng.*, 42(2-4):371–375, June 2002. ISSN 0360-8352.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

Andreas Holzinger, Chris Biemann, Constantinos S. Pattichis, and Douglas B. Kell. What do we need to build explainable AI systems for the medical domain? *CoRR*, abs/1712.09923, 2017.

Philip Hougaard. *Analysis of Multivariate Survival Data*. Springer-Verlag New York, 2000.

Kejia Hu, Jason Acimovic, francisco erize, Douglas Thomas, and Jan Van Mieghem. Forecasting product life cycle curves: Practical approach and empirical analysis. *Manufacturing & Service Operations Management*, To appear, 07 2017.

A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: A review. *ACM Comput. Surv.*, 31(3):264–323, September 1999. ISSN 0360-0300.

Anil K. Jain. Data clustering: 50 years beyond k-means. *Pattern Recogn. Lett.*, 31(8): 651–666, June 2010. ISSN 0167-8655.

Kenneth B. Kahn. An exploratory investigation of new product forecasting practices. *Journal of Product Innovation Management*, 19(2):133–143, 2002.

R. Y. Kim. The impact of covid-19 on consumers: Preparing for digital sales. *IEEE Engineering Management Review*, pages 1–1, 2020.

Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 12 2014.

Ron Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'95, page 1137–1143, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc. ISBN 1558603638.

A. Kolmogoroff. Über die analytischen methoden in der wahrscheinlichkeitsrechnung. *Mathematische Annalen*, 1931.

Wouter M. Kouw. An introduction to domain adaptation and transfer learning. *CoRR*, abs/1812.11806, 2018.

Jouni Kuha. Aic and bic: Comparisons of assumptions and performance. *Sociological Methods & Research*, 33(2):188–229, 2004.

Sunil Kumar and Jayashankar Swaminathan. Diffusion of innovations under supply constraints. *Operations Research*, 51:866–879, 12 2003.

Takashi Kuremoto, Shinsuke Kimura, Kunikazu Kobayashi, and Masanao Obayashi. Time series forecasting using a deep belief network with restricted boltzmann machines. *Neurocomputing*, 137:47–56, 2014.

Pat Langley and Herbert A. Simon. Applications of machine learning and rule induction. *Commun. ACM*, 38(11):54–64, November 1995.

Albert Lechner and Steve Gunn. Dynamic cluster based markov model for demand forecasting. In *5th International Conference on Computer Science and Information Technology (CSTY-2019)*, pages 83–96, 11 2019.

Albert Lechner and Steve Gunn. Product lifecycle de-trending for sales forecasting. In *5th International Conference on Complexity, Future Information Systems and Risk*, pages 25–33, 5 2020.

Elisa T. Lee and John Wenyu Wang. *Statistical Methods for Survival Data Analysis*. J. Wiley, 1971. ISBN 978-1-118-09502-7.

Hau L. Lee, V. Padmanabhan, and Seungjin Whang. Information distortion in a supply chain: The bullwhip effect. *Management Science*, 43(4):546–558, 1997.

Peter J. Lenk and Ambar G. Rao. New models from old: Forecasting product adoption by hierarchical bayes procedures. *Marketing Science*, 9(1):42–53, 1990. ISSN 07322399, 1526548X.

Jesse Levinson, Jake Askeland, Jan Becker, Jennifer Dolson, David Held, Sören Kammel, J Zico Kolter, Dirk Langer, Oliver Pink, Vaughan Pratt, Michael Sokolsky, Ganymed Stanek, David Stavens, Alex Teichman, Moritz Werling, and Sebastian Thrun. Towards fully autonomous driving: Systems and algorithms. In *2011 IEEE Intelligent Vehicles Symposium (IV)*, pages 163 – 168, 07 2011.

A.L.D. Loureiro, V.L. Miguéis, and Lucas F.M. da Silva. Exploring the use of deep neural networks for sales forecasting in fashion retail. *Decision Support Systems*, 114: 81–93, 2018.

Chi-Jie Lu and Ling-Jing Kao. A clustering-based sales forecasting scheme by using extreme learning machine and ensembling linkage methods with applications to computer server. *Engineering Applications of Artificial Intelligence*, 55:231 – 238, 2016. ISSN 0952-1976.

Shaohui Ma, Robert Fildes, and Tao Huang. Demand forecasting with high dimensional data: The case of sku retail sales forecasting with intra- and inter-category promotional information. *European Journal of Operational Research*, 249(1):245 – 257, 2016. ISSN 0377-2217.

J. B. MacQueen, editor. *Some Methods for classification and Analysis of Multivariate Observations*, 1967. Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability, University of California Press.

Massimiliano Marcellino, James H. Stock, and Mark W. Watson. A comparison of direct and iterated multistep ar methods for forecasting macroeconomic time series. *Journal of Econometrics*, 135(1):499 – 526, 2006. ISSN 0304-4076.

Jérôme Massiani and Andreas Gohs. The choice of Bass model coefficients to forecast diffusion for innovative products: An empirical investigation for new automotive technologies. Working Papers 2016: 37, Department of Economics, University of Venice "Ca' Foscari", 2016.

Luís Meira-Machado, Jacobo de Uña-Alvarez, Carmen Cadarso-Suárez, and Per K Andersen. Multi-state models for the analysis of time-to-event data. *Statistical Methods in Medical Research*, 18(2):195–222, 2009. ISSN 0962-2802.

John Mentzer and Mark Moon. *Sales Forecasting Management: A Demand Management Approach*. SAGE Publications, Inc., 2005.

Marc H. Meyer. Revitalize your product lines through continuous platform renewal. *Research-Technology Management*, 40(2):17–28, 1997.

R. Michalski, J. Carbonell, and T. Mitchell. *Machine Learning: An Artificial Intelligence Approach*. Springer, 2013.

Banwari Mittal. Measuring purchase-decision involvement. *Psychology & Marketing*, 6 (2):147–162, 1989.

Douglas C. Montgomery, Cheryl L. Jennings, and Murat Kulahci. *Introduction to Time Series Analysis and Forecasting*. Wiley, 2008.

Kevin P. Murphy. *Machine learning : a probabilistic perspective*. MIT Press, Cambridge, Mass. [u.a.], 2013.

Fionn Murtagh. A survey of recent advances in hierarchical clustering algorithms. *Comput. J.*, 26:354–359, 11 1983.

S.L. Narasimhan, D.W. McLeavey, and P.J. Billington. *Production Planning and Inventory Control*. Prentice Hall, 1995.

Ramya Neelamegham and Pradeep Chintagunta. A bayesian model to forecast new product performance in domestic and international markets. *Marketing Science*, 18 (2):115–136, 1999.

Oded Netzer, James M. Lattin, and V. Srinivasan. A hidden markov model of customer relationship dynamics. *Marketing Science*, 27(2):185–204, 2008.

Syam Niladri and Sharma Arun. Waiting for a sales renaissance in the fourth industrial revolution: Machine learning and artificial intelligence in sales research and practice. *Industrial Marketing Management*, 69:135 – 146, 2018. ISSN 0019-8501.

Hani Omar, Van Hoang, and Duen-Ren Liu. A hybrid neural network model for sales forecasting based on arima and search popularity of article titles. *Computational Intelligence and Neuroscience*, 2016:1–9, 01 2016.

S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Phillip E. Pfeifer and Robert L. Carraway. Modeling customer relationships as markov chains. *Journal of Interactive Marketing*, 14(2):43–55, 2000.

Salehzadeh Reza, Tabaeeian Reihaneh Alsadat, and Esteki Farahnaz. Exploring the consequences of judgmental and quantitative forecasting on firms' competitive performance in supply chains. *Benchmarking: An International Journal*, 27(5):1717–1737, 2020/07/14 2020.

David R. Roberts, Volker Bahn, Simone Ciuti, Mark S. Boyce, Jane Elith, Gurutzeta Guillera-Arroita, Severin Hauenstein, José J. Lahoz-Monfort, Boris Schröder, Wilfried Thuiller, David I. Warton, Brendan A. Wintle, Florian Hartig, and Carsten F. Dormann. Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure. *Ecography*, 40(8):913–929, 2017.

Akkarapol Sa-ngasoongsong, Satish T.S. Bukkapatnam, Jaebeom Kim, Parameshwaran S. Iyer, and R.P. Suresh. Multi-step sales forecasting in automotive industry based on structural relationship identification. *International Journal of Production Economics*, 140(2):875 – 887, 2012. ISSN 0925-5273. Sixteenth internationalworkingseminaronproductioneconomics, Innsbruck, 2010.

S. R. Safavian and D. Landgrebe. A survey of decision tree classifier methodology. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(3):660–674, 1991.

Yves R. Sagaert, El-Houssaine Aghezzaf, Nikolaos Kourentzes, and Bram Desmet. Tactical sales forecasting using a very large set of macroeconomic indicators. *European Journal of Operational Research*, 264(2):558 – 569, 2018. ISSN 0377-2217.

Alaa Sagheer and Mostafa Kotb. Time series forecasting of petroleum production using deep lstm recurrent networks. *Neurocomputing*, 323:203 – 213, 2019. ISSN 0925-2312.

David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. Deepar: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, 36(3):1181–1191, 2020. ISSN 0169-2070.

Nada Sanders and Karl Manrodt. The efficacy of using judgmental versus quantitative forecasting methods in practice. *Omega*, 31:511–522, 12 2003.

Amit Saxena, Mukesh Prasad, Akshansh Gupta, Neha Bharill, Om Prakash Patel, Aruna Tiwari, Meng Joo Er, Weiping Ding, and Chin-Teng Lin. A review of clustering techniques and developments. *Neurocomput.*, 267(C):664–681, December 2017. ISSN 0925-2312.

Albert B. Schwarzkopf, Richard J. Tersine, and John S. Morris. Top-down versus bottom-up forecasting strategies. *International Journal of Production Research*, 26 (11):1833–1843, 1988.

Paul Schweitzer. Stochastic models, an algorithmic approach, by henk c. tijms (chichester: Wiley, 1994), 375 pages, paperback. *Probability in the Engineering and Informational Sciences*, 10(3):463–464, 1996.

Seo and Web Ltd. Cars technical specifications database, https://www.car-database-api.com/, 2019.

Durga L. Shrestha and Dimitri P. Solomatine. Machine learning approaches for estimation of prediction interval for the model output. *Neural Networks*, 19(2):225 – 235, 2006. ISSN 0893-6080. Earth Sciences and Environmental Applications of Computational Intelligence.

David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489, January 2016. ISSN 0028-0836.

Victor Solo. The order of differencing in arima models. *Journal of The American Statistical Association - J AMER STATIST ASSN*, 79:916–921, 12 1984.

R. Solomon, P. A. Sandborn, and M. G. Pecht. Electronic part life cycle concepts and obsolescence forecasting. *IEEE Transactions on Components and Packaging Technologies*, 23(4):707–717, 2000.

H. Spath. *Cluster Analysis Algorithms.* Ellis Horwood, 1980.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014.

J. Stark. *Product Lifecycle Management.* Springer, 2015.

Michael Steinbach, George Karypis, and Vipin Kumar. A comparison of document clustering techniques. In *In KDD Workshop on Text Mining*, 2000.

Marsland Stephen. *Machine Learning: an algorithmic perspective.* Chapman and Hall CRC Machine Learning and Pattern Recognition. CRC Press, Boca Raton, FL, 2009.

T. Stock and G. Seliger. Opportunities of sustainable manufacturing in industry 4.0. *Procedia CIRP*, 40:536 – 541, 2016. ISSN 2212-8271. 13th Global Conference on Sustainable Manufacturing – Decoupling Growth from Resource Use.

Joe Sullivan and William Woodall. A comparison of fuzzy forecasting and markov modeling. *Fuzzy Sets and Systems*, 64:279–293, 06 1994.

Zhan-Li Sun, Tsan-Ming Choi, Kin-Fan Au, and Yong Yu. Sales forecasting using extreme learning machine with applications in fashion retailing. *Decision Support Systems*, 46(1):411–419, 2008.

Zaiyong Tang and Paul A. Fishwick. Feedforward neural nets as models for time series forecasting. *ORSA Journal on Computing*, 5(4):374–385, 1993.

R Vernon. International investment and international trade in the product cycle. *Quarterly Journal of Economics*, 80:190–207, 05 1966.

Richard Webby and Marcus O'Connor. Judgemental and statistical time series forecasting: a review of the literature. *International Journal of Forecasting*, 12(1):91 – 118, 1996. ISSN 0169-2070. Probability Judgmental Forecasting.

W Wei. *Time Series Analysis: Univariate and Multivariate Methods.* Addison-Wesley Publishing Company, 1990.

Brent D. Williams and Matthew A. Waller. Top-down versus bottom-up demand forecasts: The value of shared point-of-sale data in the retail supply chain. *Journal of Business Logistics*, 32(1):17–26, 2011.

Cort J. Willmott and Kenji Matsuura. Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Climate Research*, 30(1):79–82, 2005.

D.J. Wright, G. Capon, R. Pagé, J. Quiroga, A.A. Taseen, and F. Tomasini. Evaluation of forecasting methods for decision support. *International Journal of Forecasting*, 2 (2):139 – 152, 1986. ISSN 0169-2070.

Zhaohua Wu, Norden E. Huang, Steven R. Long, and Chung-Kang Peng. On the trend, detrending, and variability of nonlinear and nonstationary time series. *Proceedings of the National Academy of Sciences*, 104(38):14889–14894, 2007. ISSN 0027-8424.

R. Yaffee and M. McGee. *Introduction to time series analysis and forecasting*. Academic Press, 2009.

Junchi Yan, Chao Zhang, Hongyuan Zha, Min Gong, Changhua Sun, Jin Huang, Stephen Chu, and Xiaokang Yang. On machine learning towards predictive sales pipeline analytics. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, pages 1945–1951. AAAI Press, 2015. ISBN 0-262-51129-0.

Judith Lynne Zaichkowsky. Measuring the involvement construct. *Journal of Consumer Research*, 1985.

Peter Zhang. Time series forecasting using a hybrid arima and neural network model. *Neurocomputing*, 50:159–175, 2003.

Peter Zhang, Eddy Patuwo, and Michael Hu. Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting*, 14:35–62, 03 1998.

Peter Zhang and Min Qi. Neural network forecasting for seasonal and trend time series. *European Journal of Operational Research*, 160:501–514, 02 2005.

Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. Interpretable convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

Shang-Ming Zhou and John Q. Gan. Low-level interpretability and high-level interpretability: a unified view of data-driven interpretable fuzzy system modelling. *Fuzzy Sets and Systems*, 159(23):3091 – 3131, 2008. ISSN 0165-0114. Theme: Modeling.

Dong-Mei Zhu and Wai-Ki Ching. A new estimation method for multivariate markov chain model with application in demand predictions. In *Proceedings of the 2010 Third International Conference on Business Intelligence and Financial Engineering*, BIFE '10, pages 126–130, Washington, DC, USA, 2010. IEEE Computer Society. ISBN 978-0-7695-4116-7.

L. Zhu and N. Laptev. Deep and confident prediction for time series at uber. In *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 103–110, 2017.

Özden Gür Ali, Serpil Sayın, Tom van Woensel, and Jan Fransoo. Sku demand forecasting in the presence of promotions. *Expert Systems with Applications*, 36(10):12340 – 12348, 2009. ISSN 0957-4174.