Niklas Virtanen

Hybrid Machine Learning and Time-Series Modelling to Forecast Product Demand

A CRISP-DM Approach

School of Technology and Innovations
Master's Thesis
Industrial Systems Analytics

Vaasa 2026

UNIVERSITY OF VAASA			
School of Technology and Innovations			
Author:	Niklas Virtanen		
Title of the thesis:	Hybrid Machine Learning and Time-Series Modelling to Forecast Product Demand: A CRISP-DM Approach		
Degree:	Master of Science in Technology		
Discipline:	Industrial Systems Analytics		
Supervisor:	Mohammed Elmusrati		
Evaluator:	Emmanuel Ndzibah		
Instructor:	Rami Valonen		
Year:	2026	**Pages:**	101

ABSTRACT:

Efficient demand forecasting can yield significant competitive advantage for companies in today’s market, making it a crucial process in supply chain management. By exploring and implementing modern demand forecasting models, encompassing time-series, machine learning, deep learning and hybrid approaches, companies can gain benefits in important business areas such as strategic planning, operational efficiency and customer service.

This research is done in collaboration with a case company, and it aims to identify and develop a product demand forecasting model that improves forecasting accuracy compared against a baseline approach and to identify and analyze key demand drivers that influence demand. To meet these goals, an industry standard CRISP-DM framework is utilized to develop and compare multiple time-series, machine learning and their hybrid combination models to find the best performing forecasting solution. Key demand drivers are identified and analyzed by combining SHAP values with statistical analysis.

The results of this study indicate that a hybrid demand forecasting model combining Holt-Winters Exponential Smoothing and Elastic Net Regression outperformed the other models in the comparison and improved the forecasting accuracy significantly compared to the baseline approach. The results also suggest that product demand is driven by trend and seasonal factors, short-term sales pricing and momentum as well as macroeconomic leading indicators such as industry sentiment, interest rates and euro strength.

Despite the significant improvement in forecasting accuracy and practical insights through the identified demand drivers, research limitations such as limited amount of training data, small test set and lack of true causal analysis of the relationships between features and demand are acknowledged to emphasize the need for future research. The development and evaluation of the HW-Elastic Net model contributes to current research by highlighting the importance of exploring hybrid models in demand forecasting, especially on small and noisy real-world datasets.

KEYWORDS: Machine Learning, Demand Forecasting, Hybrid Models, CRISP-DM

VAASAN YLIOPISTO
Tekniikan ja Innovaatiojohtamisen Yksikkö

Tekijä:	Niklas Virtanen		
Tutkielman nimi:	Hybridi koneoppimis- ja aikasarjamallinnus tuotekysynnän ennustamisessa: CRISP-DM lähestymistapa		
Tutkinto:	Diplomi-insinööri		
Koulutusohjelma:	Tuotantotalous		
Opintosuunta:	Industrial Systems Analytics		
Työn ohjaaja:	Mohammed Elmusrati		
Työn arvioija:	Emmanuel Ndzibah		
Työn valvoja:	Rami Valonen		
Valmistumisvuosi:	2026	**Sivumäärä:**	101

TIIVISTELMÄ:

Tehokas kysynnän ennustaminen voi johtaa merkittävään kilpailuetuun nykypäivän yrityskentällä, tehden siitä tärkeän prosessin toimitusketjun hallinnassa. Tutkimalla ja tuotannollistamalla moderneja kysyntäennustemalleja, kuten aikasarja-, koneoppimis-, syväoppimis- ja hybridimalleja, yritykset voivat saada hyötyjä tärkeillä liiketoiminta-alueilla kuten strategisessa suunnittelussa, operatiivisessa tehokkuudessa sekä asiakaspalvelussa.

Tämä diplomityö on tehty yhteistyössä kohdeyrityksen kanssa. Tutkimuksen tavoitteena on tunnistaa ja kehittää yrityksen valitsemalle tuotteelle kysyntäennustemalli, joka parantaa ennustetarkkuutta verrattuna lähtökohtaennusteisiin sekä tunnistaa ja tulkita keskeisiä kysyntäajureita. Tutkimus hyödyntää CRISP-DM viitekehystä usean aikasarja-, koneoppimis- sekä hybridimallin kehittämiseen sekä vertailuun tunnistaakseen parhaiten suoriutuvan kysyntäennusteratkaisun. Keskeisten kysyntäajureiden tunnistaminen ja tulkinta perustuu SHAP-arvojen sekä tilastollisen analyysin yhdistelmään.

Tutkimuksen lopputulokset osoittavat, että Holt-Winters Exponential Smoothing sekä Elastic Net Regression malleja yhdistävä hybridimalli suoritui paremmin kuin muut vertailussa sisällytetyt mallit sekä paransi ennustetarkkuutta merkittävästi verrattuna lähtötasomalliin. Tämän lisäksi tulokset osoittavat, että trendi- ja kausittaisuustekijät, lyhyen välin tuotehinta- sekä myyntitasot sekä makroekonomiset ennakoivat indikaattorit kuten teollisuuden luottamustaso, korkotasot sekä euron vahvuus ovat keskeisiä ajureita tuotekysynnälle.

Diplomityönä kehitetyn mallin tuottamasta ennustetarkkuuden merkittävästä parannuksesta sekä tunnistettujen kysyntäajureiden tuottamasta ymmärryksestä huolimatta tutkimuksessa tunnistettiin rajoitteita. Harjoitusdatan rajallinen määrä, pieni testidatan koko sekä aitojen syy-seuraussuhteiden analysoinnin puute korostavat jatkotutkimuksen tarvetta. Tutkimuksen lopputuloksena kehitetty HW-Elastic Net malli edistää nykyistä tutkimusta osoittamalla hybridimallien tutkimisen tärkeyttä kysynnän ennustamisessa, erityisesti pienten ja kohinaa sisältävien datasarjojen yhteydessä.

AVAINSANAT: Koneoppiminen, Kysynnän ennustaminen, Hybridimallit, CRISP-DM

Contents

Figures

Tables

Abbreviations

ACF – Autocorrelation function

ANN – Artificial Neural Network

API – Application Programming Interface

ARIMA – AutoRegressive Integrated Moving Average

CRISP-DM – Cross-Industry Standard Process for Data Mining

DL – Deep Learning

ECB – European Central Bank

EDA – Exploratory Data Analysis

EMA – Exponential Moving Average

HW – Holt-Winters Exponential Smoothing

LIME – Local Interpretable Model-Agnostic Explanations

LSTM – Long Short-Term Memory

MAE – Mean Absolute Error

MAPE – Mean Absolute Percentage Error

MI – Mutual Information

ML - Machine Learning

MLR – Multiple Linear Regression

PACF – Partial Autocorrelation Function

RMSE – Root Mean Squared Error

RNN – Recurrent Neural Network

SHAP – SHapley Additive exPlanations

SVR – Support Vector Regression

TFT – Temporal Fusion Transformer

USD – United States Dollar

XGBoost – Extreme Gradient Boosting

YF – Yahoo Finance

1 Introduction

Efficient demand forecasting directly translates into enhanced performance in supply chain and inventory management, therefore it is determined as a crucial process for companies striving for competitive advantage in the modern market (Feizabadi, 2022; Harahap et al., 2025; Rahman Mahin et al., 2025). Demand forecasting methods and techniques have been extensively researched and developed with significant results in performance improvement. The methods include statistical, time-series, machine learning and deep learning models, each with their own respective advantages and limitations. The trade-off between the strengths and limitations motivates the study of hybridizing different techniques to account for the model limitations by combining their advantages to improve demand forecasting efficiency.

This thesis contributes to a gap in hybrid demand forecasting research by conducting a quantitative single case study comparing the performance of time-series models, machine learning models and hybridized combinations of the models on a small real-world demand forecasting dataset with high variability. The research is done in collaboration with the case company in the distribution business as proof of concept for data science, machine learning and advanced analytics solutions. Research background and problem definition, main research objectives and questions, keyword definitions, research scope as well as the structure of this thesis are presented in the following sections.

1.1 Background and problem definition

The case company aims to increase their analytical maturity and develop their data-driven decision-making capabilities by incorporating data science, machine learning and advanced analytics solutions into their core processes. The company operates as a distributor, where aligning supply with demand is essential for business performance, making demand forecasting a crucial process and thus the application area of this research project. Inaccurate forecasting can lead to over- or understocking situations.

Overstocking can increase holding costs, tie up capital and force the company to sell excess stock at lower prices than expected, lowering profit margins. Understocking can lead to lost sales and customer dissatisfaction, disrupted supplier relationships and decreased competitive advantage.

This thesis develops and compares multiple machine learning, time series and hybrid models to improve forecasting accuracy and interpret the demand of a single product that was selected by the company. The selected product has been shown to be difficult to forecast due to its spot sale nature and high demand variability. The lead time for the selected product is three months, meaning that demand needs to be forecasted well in advance for the insights to be actionable.

Currently, the baseline demand forecasting method the company uses for the selected product is judgmental forecasting, where demand planning is primarily based on the experience and intuition of experts supplemented with customer interviews and a basic historical approach that compares current demand to the same period last year. This approach has led to inaccurate forecasts, often overestimating the actual sales and leading to potential overstock issues. To counter this measure, the company must seldom purchase significantly less than the forecasted amount, making the current forecasting process unreliable.

1.2 Research objectives and questions

The first objective of this thesis is to identify and develop a demand forecasting model for the selected product to increase three-month forecasting accuracy against the baseline approach to support and optimize inventory planning processes. The second research objective is to identify and analyze external key demand drivers and leading indicators that influence product demand. In addition, this thesis acts as a proof of concept for data science, machine learning and advanced analytics solutions to enable further development in the company.

To achieve the abovementioned research objectives, the conducted research is guided by two main research questions. The questions are derived from the objectives to ensure the alignment of this thesis. The research questions are formed as:

1. What is the most efficient demand forecasting model for predicting the demand of the selected product?
2. What are the key demand drivers for the selected product?

1.3 Definitions and scope of the study

This section provides the definitions of the key concepts that form the theoretical foundation for this research and explains how they are utilized in the study. In addition, the methodological boundaries of this research are outlined to specify the scope of this study.

1.3.1 Theoretical definitions

Machine learning refers to the computational methods that can be used to learn patterns from historical data to improve performance and make accurate predictions without being explicitly programmed (Goar and Yadav, 2024; Samuel, 1959). The theoretical foundation for ML in this thesis consists of fundamentals, different types of learning and the core concepts of ML that are central for the development of the forecasting models compared in this research. However, the empirical ML modelling done in this research focuses solely on regression models, as the goal is to forecast a continuous variable. In addition, highly complex and data-intensive ML models and deep learning models were excluded from the scope of the empirical research due to insufficient amount of training data.

Demand forecasting refers to an analytical process that uses historical data, statistical tools, machine learning and market analysis to predict and understand future demand

(Harahap et al., 2025; Ingle, 2021). The demand forecasting theory in this thesis examines the benefits and challenges associated with the process. The quantitative research done in this study focuses on monthly demand forecasting of a single product with a three-month prediction horizon, which was set as a requirement by the case company.

Hybrid models refer to a family of predictive models that combine two or more algorithms into one model, so that the combined algorithms complement each other by using their respective advantages to account for the limitations of the other model to possibly generate more efficient and accurate outputs compared to the isolated models. (Aburto and Weber, 2006; Ahmed and Husien, 2024). The theory building of this thesis encompasses several different algorithms that can be used for demand forecasting, reviewing their advantages and limitations. Literature regarding hybrid models is also reviewed to clarify the potential benefits of the approach. In the empirical research conducted in this thesis, hybrid modelling is utilized to combine univariate time-series models with ML models to compare the performance of a hybrid model against the isolated models.

CRISP-DM (Cross-Industry Standard Process for Data Mining) is the most common framework for data science projects in industry (Shimaoka et al., 2024). This thesis utilizes it as the core methodological framework to guide model development and comparison. The research follows the CRISP-DM phases of business understanding, data understanding, data preparation, modelling and evaluation. The deployment phase is excluded from this thesis, as it is done separately by the case company.

1.3.2 Scope of the study

The scope of this research consists of forecasting the three-month demand of a single product selected by the case company by using monthly internal sales and pricing data complemented with external macroeconomic indicators. The dataset covers a timeframe from 2015-2025, from which the model is trained with data from 2015-2024 and the performance is tested on out-of-sample data from January to September 2025. The research is limited to classical time-series models, ML models and their hybrid combinations due to the lack of training data for more complex models.

The study also focuses on model interpretability by utilizing SHAP-values and MLR p-values to identify key demand drivers. These methods explain model predictions and provide correlation-based insights but are not able to establish causal relationships. The deployment of the best performing model is excluded from the scope of this thesis and will be done separately by the case company.

1.4 Thesis structure

This thesis consists of two main parts: A theory building section that forms the theoretical foundation for the research and a quantitative case study that aims to meet the research objectives and answer the research questions. Theory building is developed in chapter 2, where chapter 2.1 focuses on machine learning, its fundamentals and core concepts. Chapter 2.2 reviews literature regarding demand forecasting, emphasizing its potential benefits and common challenges, and chapter 2.3 examines the theory behind different demand forecasting algorithms and approaches. To finalize the theory building section, chapter 2.4 summarizes the theoretical framework of this study.

Chapter 3 presents and justifies the methodological approaches applied in this research, where chapter 3.2 walks through the CRISP-DM approach from business and data understanding to data preparation, modelling and evaluation. The results of this research are presented, interpreted and discussed in chapter 4. Lastly, in chapter 5, this thesis is summarized, and the main conclusions are presented along with discussion regarding the limitations of this study and future research suggestions.

2 Theory building

The aim of this chapter is to build a theoretical framework for this thesis and provide understanding of key concepts that are central to the topic of this research. The theory building begins with machine learning and its core concepts, followed by a theoretical examination of demand forecasting, focusing on its potential benefits and challenges. Next, different quantitative models for demand forecasting are explored, encompassing time-series models, machine learning models, deep learning models and hybrid models. The theory building concludes with a summary of the theoretical framework of this study, connecting the central concepts to the main research topic.

2.1 Machine learning

Recently, the field of machine learning has progressed rapidly, mainly due to the development of new algorithms, low-cost high performance computing and exponentially growing amount of data (Jordan & Mitchell, 2015). The large volume of data has made it possible for machines to transition from only processing information to creating intelligence by making sense of big data, and the applications of machine learning and computational intelligence are becoming more common and accessible to anyone (Joshi, 2023). According to the article authored by Jordan and Mitchell (2015), these applications can be found across various fields from healthcare and education to business and industry. To gain a deeper understanding of the topic, the following sections cover the theoretical foundations of machine learning, discussing fundamental principles, different types of learning and core concepts that are relevant to the field.

2.1.1 Fundamentals of machine learning

One of the earliest definitions of machine learning by Arthur Samuel (1959) states that machine learning is a “field of study that gives computers the ability to learn without being explicitly programmed”. More recently, modern research shows multiple definitions for machine learning. According to the book authored by Goar and Yadav (2024), machine learning can be broadly defined as computational methods that use historical information to improve performance and make accurate predictions. Mahesh (2020) defines machine learning in a manner similar to Samuel, as the scientific study of algorithms and statistical models that machines can use to perform specific tasks without being explicitly programmed. IBM (2021) states that machine learning is a “branch of artificial intelligence focused on enabling computers and machines to imitate the way that humans learn, to perform tasks autonomously, and to improve their performance and accuracy through experience and exposure to more data”.

2.1.1.1 Aim of machine learning

The main goal of machine learning is to learn from data (Mahesh, 2020). Machine learning combines core computer science principles with concepts from statistics, probability and optimization to design and develop efficient and accurate mathematical algorithms and build analytical models (Goar & Yadav, 2024). These algorithms uncover and learn the mathematical relationships and patterns found in data to generate predictions, recommendations, rules or other outputs that can be utilized to support advanced problem solving and data-driven decision-making (Janiesch et al., 2021; Goar & Yadav, 2024). The most common problems that can be solved utilizing machine learning are related to regression, classification, clustering, recommendation and ranking (Joshi, 2023). However, according to Mahesh (2020), not all machine learning algorithms fit all problems, and the choice of algorithm is done based on the problem and available data. Selecting the best algorithm for the problem is crucial for obtaining the best results (Alzubi et al., 2018).

2.1.1.2 Algorithms vs. models

To understand how machine learning works in practice, it is essential to discuss the roles of algorithms and models. Algorithms are precisely defined computational procedures for solving problems by taking inputs and transforming them into outputs (Manelli, 2020). According to Ra and Souza (2019), machine learning algorithms guide how machines learn from historical, present and future data. When developing a machine learning solution, algorithms are often trained with large amounts of data to optimize their internal parameters to find patterns, trends and relationships from data (Databricks, n.d.). The output of the training process is a machine learning model, that can be defined as a computer program with specific rules and structures derived from data (Databricks, n.d.). IBM (n.d.) defines a machine learning model as the output of a machine learning algorithm that has been applied to a dataset. A high-quality model should be able to generalize well, meaning it can provide accurate outputs on unseen data, not only the data it was trained on (Barbiero et al., 2020). In practical terms, machine learning models are used to make decisions and predictions, while machine learning algorithms provide the logic that determines how the outputs are generated (IBM, n.d.).

2.1.2 Machine learning paradigms and deep learning

According to Bi et al. (2019), machine learning can be classified into four main categories: supervised learning, unsupervised learning, semi-supervised learning and reinforcement learning. Each different machine learning paradigm is designed to solve different types of problems and are categorized depending on how the algorithm is trained and the properties of available training data (Alzubi et al., 2018; Mahesh, 2020). Deep learning is a subcategory of machine learning based on artificial neural networks that enable modelling complex patterns in big data (Janiesch et al., 2021). The different machine learning paradigms and deep learning are covered briefly in the next sections.

2.1.2.1 Supervised learning

Supervised learning is a machine learning approach where an algorithm is trained with labelled data to enable accurate predictions on new, unseen data (Cunningham et al., 2008). Supervised machine learning algorithms aim to accurately estimate the underlying function that transforms inputs into outputs by constructing mathematical models and uncovering patterns in the data (Liu, 2011; Nasteski, 2017). According to Nasteski (2017), supervised learning is mainly used to solve regression and classification problems. Regression models are trained and used to predict continuous variables, such as the future demand for a specific product (Liu, 2011). If a model is trained to predict output values from a finite set of predefined classes, it is defined as a classification model (Liu, 2011). Classification models can, for example, be used in email filtering to filter out spam messages. Some examples of widely used supervised learning algorithms are Linear Regression, Logistic Regression, Support Vector Machine, k-Nearest Neighbors, Decision Trees and Random Forests (Ra & Souza, 2019).

2.1.2.2 Unsupervised learning

Unsupervised learning is a machine learning approach where the algorithm is trained on unspecified data with no preexisting labels to uncover hidden structural information in the data (Goar & Yadav, 2024; Alzubi et al., 2018; Janiesch et al., 2021). According to Goar and Yadav (2024), unsupervised learning is mostly used for clustering and dimensionality reduction problems. Ezugwu et al. (2022) define clustering as a method of grouping data points into unspecified clusters depending on the characteristics and features of the data points. Clustering can be used, for example, in customer segmentation. Some examples of popular clustering algorithms are K-means and DBSCAN (Yin et al., 2024). According to Murel and Kavlakoglu (2024), dimensionality reduction can be used to project data in a lower dimension while still preserving the important information and properties of the original dataset. Dimensionality reduction methods like PCA or t-SNE can be used to enhance machine learning models by increasing accuracy while decreasing computation time (Murel & Kavlakoglu, 2024).

2.1.2.3 Semi-supervised learning

Semi-supervised learning combines aspects from both supervised and unsupervised learning (Mahesh, 2020). Usually, semi-supervised algorithms aim to improve the accuracy and performance of a supervised or unsupervised learning task by using information that is generally associated with the other (van Engelen & Hoos, 2020). According to Alzubi et al. (2018), these types of algorithms are suitable for building machine learning models in situations where a dataset contains both labelled and unlabelled data. Even though majority of research is focused on classification, semi-supervised learning can be also used to solve regression and prediction problems (Alzubi et al., 2018; van Engelen & Hoos, 2020). Semi-supervised learning can be categorized into transductive and inductive learning (Bergmann, 2023). According to Bergmann (2023), transductive learning methods use the labelled data to predict the labels of the unlabelled data, so the whole dataset can be used to train a supervised learning model. The goal of inductive training methods is to directly train a regression or classification model by using both the labelled and unlabelled data (Bergmann, 2023).

2.1.2.4 Reinforcement learning

In reinforcement learning, there are no input and output pairs for the machine to learn (Janiesch et al., 2021). Instead, according to Joshi (2023), in reinforcement learning, the system continuously interacts with the environment and searches for optimal behavior based on the feedback it gets from its actions. Reinforcement learning methods are used to solve various sequential decision-making tasks (Murphy, 2024). According to the paper authored by Huang (2020), reinforcement learning can be categorized into model-based methods and model-free methods depending on if the environment is understood beforehand or not. Model-free methods do not aim to understand the environment, but give actions according to environment changes and feedback (Huang, 2020). Model-based methods have a model that enables the agent to think in advance and make decisions based on the predicted outcome of different possible actions (Huang, 2020). Reinforcement learning applications are widely used in the fields of robotics and natural language processing (Murel & Kavlakoglu, 2024).

2.1.2.5 Deep learning

Deep learning can de defined as a specialized subset of machine learning that uses deep, multilayered artificial neural networks to mimic the complicated decision making power of a human brain (Shinde & Shah, 2018; Holdsworth & Scapicchio, 2024). Deep learning models consist of multiple processing layers to learn representations of data with multiple levels of abstraction by utilizing sophisticated algorithms, which optimize the internal parameters of the model by minimizing a loss function (Lecun et al., 2015; Mathew et al., 2021). Various types of neural networks can be used to solve supervised, unsupervised, semi-supervised and reinforcement learning problems (Mathew et al., 2021). Examples of widely used network architectures are CNNs and RNNs (Joshi, 2023). According to an article authored by Lecun et al. (2015), CNN models are efficient when working with spatial data like images while RNN models are used to deal with dynamic or sequential data like time series, speech and language. According to Mathew et al. (2021), deep learning methods often outperform traditional machine learning methods, especially when applied on large unstructured datasets.

2.1.3 Core concepts of machine learning

This section provides information about core concepts in the field of machine learning that affect the development and performance of machine learning models. The importance of high-quality data and feature engineering is emphasized first, followed by an overview of model training, testing and evaluation processes. Next, cross-validation and hyperparameter tuning are introduced as important techniques to ensure model generalization. The section then reviews the concepts of overfitting, underfitting and the bias-variance trade-off, which are imperative for balancing model complexity. Finally, the difference between interpretable and black-box models is examined to provide insight into the reasoning behind model selection.

2.1.3.1 Data quality and feature engineering

The quality and quantity of available training data has a crucial effect on the performance of a machine learning model (Gupta et al., 2021; Kariluoto et al., 2021). Data quality can be defined as the degree of data fitness for a given task (Gudivada et al., 2017). According to Mohammed et al. (2025), the lack of quality training data can lead to inaccurate machine learning models that can lead to unreliable decision-making. Missing and incomplete data, outliers, duplicates, inconsistencies and incorrectness are some of the most popular errors that contribute to poor quality of the data (Mohammed et al., 2025). According to Gudivada et al. (2017), data quality issues can be combatted by utilizing statistical methods such as imputation of missing data, outlier detection, different data transformations and dimensionality reduction techniques. These methods are often used when data is preprocessed for machine learning (Rahman, 2019).

Often in machine learning, raw data is not in a suitable form for model training, therefore relevant features need to be extracted from the data (Gudivada et al., 2017). Dong and Liu (2018) define a feature as an “attribute or variable used to describe some aspect of individual data objects”. For a feature to be useful, it needs to improve the performance of the machine learning model (Dong & Liu, 2018). The number of features is also imperative for model performance, as the lack of relevant features will most likely result in a poor performing model (Zheng & Casari, 2018). On the other hand, according to Zheng and Casari (2018), too many irrelevant features will lead to a model that is difficult and expensive to train. Feature engineering is a process where the most suitable features are formulated and selected according to the data, model and the problem at hand. The process requires strong data analysis skills and domain knowledge but can lead to a more optimized and accurate machine learning model (Murel & Kavlakoglu, 2024; Zheng & Casari, 2018).

2.1.3.2 Training, testing and model evaluation

After the dataset is ready for modelling and an algorithm is selected, it is commonly split into training and testing sets (Joshi, 2023). Majority of the data is used for training the algorithm and building the model, but a sufficient number of samples need to be left for testing to provide statistically significant performance results (Joshi, 2023). Algorithm training, also called model fitting, can be defined as the learning process where the algorithm analyses the training data and provides the parameter estimates and a mathematical model representing the relationships between inputs and outputs (Liu, 2011; Genç & Tunç, 2019). The trained model is then tested on unseen data in the testing process, where the performance of the model is evaluated according to chosen criteria (Genç & Tunç, 2019; Uçar et al., 2020). Uçar et al. (2020) state that model training and testing processes are the most critical factor regarding the success of the machine learning model.

The process of validating machine learning models is called model evaluation, and optimally it should provide a representation on how the model would perform in real-world scenarios (Varoquaux & Colliot, 2023). According to Raschka (2020), models are evaluated to estimate their performance and generalization capability, to increase the performance by tweaking the algorithm and to identify the most suitable and best-performing machine learning algorithm for the given problem by comparing multiple options. Model performance can be assessed with various mathematical expressions, also called metrics (Joshi, 2023). For example, these include RMSE and MAPE for regression models and accuracy, recall, precision and F1 scores for classification models (Janiesch et al., 2021).

2.1.3.3 Cross-validation and hyperparameter tuning

Cross-validation is a statistical technique for resampling data to assess the machine learning model’s ability to generalize and perform on unseen data (Berrar, 2019). According to Raschka (2020), the main idea in cross-validation is that all the samples in the dataset can be used for testing. In cross-validation, the training data is divided into subsets and the model is trained and validated multiple times by using different combinations of the subsets (Joshi, 2023; Varoquaux & Colliot, 2023). The cross-validated performance of the model can then be computed as the arithmetic mean of performance metrics acquired on the test set across all iterations (Raschka, 2020). Most common cross-validation types are k-fold, nested and leave-one-out cross-validation (Berrar, 2019). K-fold cross-validation is visualized in figure 1, where R stands for the training subset and V for the validation subset. Berrar (2019) states that cross-validation techniques can be used to improve the generalization performance of machine learning models, tune their hyperparameters, compare algorithms and prevent overfitting.

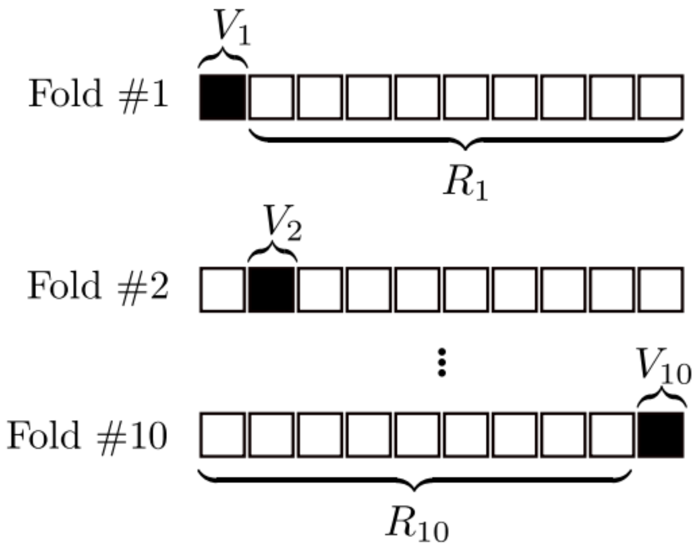


Figure 1 K-fold cross-validation (Berrar, 2019)

Model parameters are a set of internal variables that a machine learning model learns in the training process to minimize a loss function and improve performance (Joshi, 2023). Hyperparameters are free parameters that control the training process, and they are specified manually before model training (Berrar, 2019; Goar & Yadav, 2024). Hyperparameters control the behavior of algorithms and thus can be used to optimize the machine learning models' performance (Raschka, 2020). Some examples of hyperparameters include kernel type in SVM, the number of neurons in a neural network and the learning rate in XGBoost (Belcic & Stryker, 2024). According to Berrar (2019), optimal hyperparameter values can be found by utilizing resampling techniques like cross-validation. The process of identifying and selecting the best hyperparameters for training a machine learning model is called hyperparameter tuning, also referred to as hyperparameter optimization (Belcic & Stryker, 2024).

2.1.3.4 Overfitting, underfitting and the bias-variance trade-off

According to Berrar (2019), a major challenge in machine learning modelling is finding the correct balance between over- and underfitting. Overfitting can be defined as a phenomenon where a machine learning algorithm learns the noise and random errors of the dataset and fits too closely or exactly to the training data (Briscoe & Feldman, 2011; Joshi, 2023). This often leads to a model that is unable to generalize to unseen data (IBM, 2021). Overfitted models are often too complex relative to the amount and noisiness of training data and thus perform worse on new data compared to simpler models (Aliferis & Simon, 2024). Underfitting, on the other hand, refers to a phenomenon where an algorithm is incapable of capturing the relationships in the data thus often also results in a model that is unable to generalize well (Berrar, 2019). This often happens due to applying a learning algorithm that is too simple for the complexity of the data (Goar & Yadav, 2024). Underfitted, overfitted and optimal model fits are illustrated in figure 2.

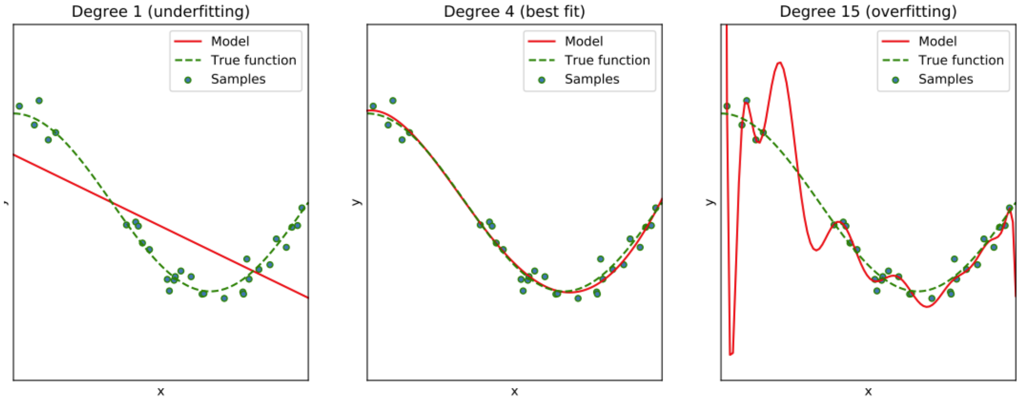


Figure 2 Overfitted, underfitted and optimal model fits (Barbiero et al., 2020)

As discussed in the previous chapter, over- and underfitting are highly dependent on balancing model complexity with its fitness for data. This trade-off is commonly defined as the bias-variance trade-off (Briscoe & Feldman, 2011). In the trade-off, bias represents the systematic error of a model and variance measures the model's sensitivity to variations in the training data (Dietterich & Kong, 1995). High bias is often connected with underfitting and simpler models that make more assumptions of the data (Briscoe & Feldman, 2011). On the other hand, according to Briscoe and Feldman (2011), high variance tends to lead to overfitting and is connected to complex models that are more flexible. According to Neal et al. (2018), the idea of the bias-variance trade-off is to indicate that as the complexity of the model increases, bias decreases and variance increases. This dynamic leads to a U-shaped test error curve that can be seen in figure 3.

Aliferis and Simon (2024) state that balancing the bias-variance trade-off is crucial to avoid under- and overfitting and leads to well-performing machine learning models. This can be done, for example, by using regularization methods, selecting the correct features and reducing the dimensionality of the data (Mucci, 2024). Additionally, according to Aliferis and Simon (2024), multiple different algorithms and their hyperparameters should be explored to see what the best fit for the data is. Model performance should also be validated with unbiased techniques such as cross-validation to ensure model generalization with unseen data (Mucci, 2024; Aliferis & Simon, 2024).

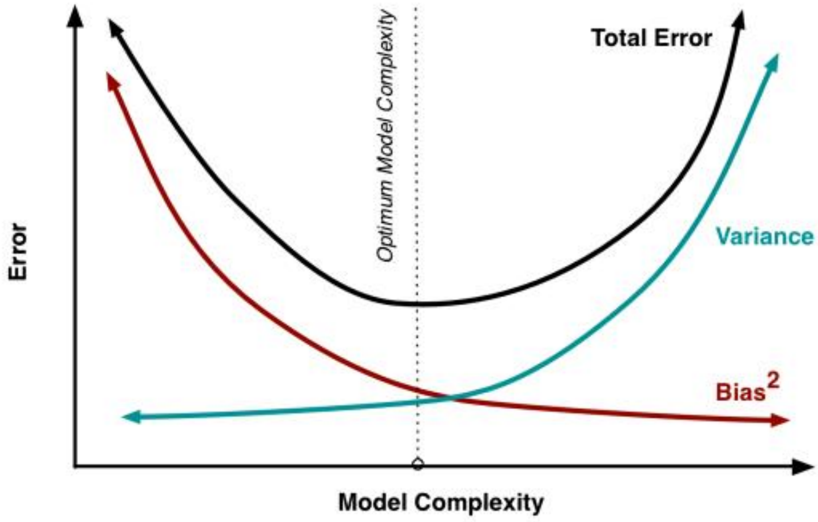


Figure 3 Bias-variance trade-off (Neal et al., 2018)

2.1.3.5 Accuracy vs. interpretability

Machine learning models can generally be categorized as interpretable or non-interpretable, with the latter often referred to as black-box models (Luo et al., 2019). Interpretable models provide understandable reasoning behind the model's predictions, leading to more straightforward and trustworthy insights to support decision-making (Atrey et al., 2025). Interpretability ensures that users can trust the model's outputs to be accurate, robust and unbiased while protecting the sensitive information in the data (Molnar, 2020). On the contrary, black-box models have proven to often provide high accuracy and outperform interpretable models, with the cost of being less interpretable and transparent (Atrey et al., 2025). To combat the lack of interpretability, recent development in explainable artificial intelligence has introduced methods such as LIME and SHAP to make black-box models more interpretable (Salih et al., 2025).

Atrey et al. (2025) state that even though black-box models are often regarded as the more accurate option, interpretable models can outperform them in some scenarios.

According to Bhattacharya (2022), an optimal model would be highly accurate and highly interpretable. In practice, this is often difficult, as there is always a trade-off between model accuracy and interpretability (Bhattacharya, 2022). The selection between a more interpretable or more accurate model needs to be done depending on the problem being solved and the users of the model (Bhattacharya, 2022). Figure 4 visualizes the accuracy-interpretability trade-off of popular machine learning algorithms.

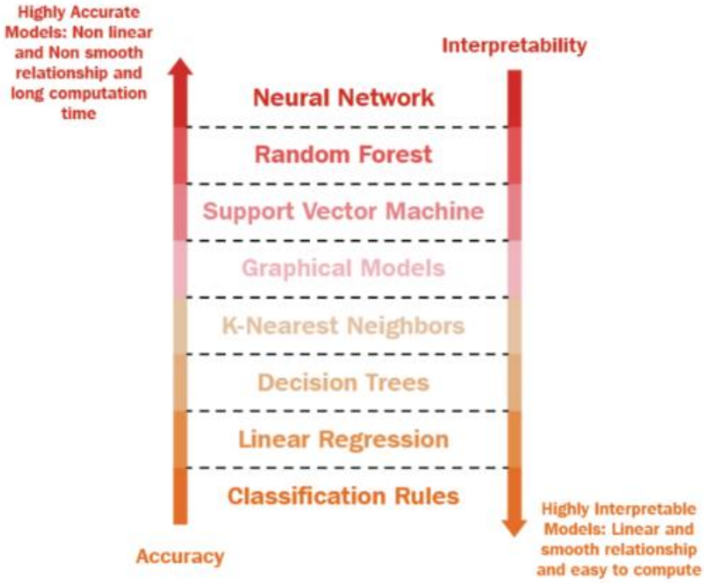


Figure 4 Accuracy-interpretability trade-off of popular machine learning algorithms (Bhattacharya, 2022)

2.2 Demand forecasting

Efficient inventory and supply chain management is imperative for businesses to be successful in the modern competitive market (Harahap et al., 2025; Rahman Mahin et al., 2025). Demand uncertainty has a significant effect on supply chain management, particularly in inventory planning, transportation and production scheduling (Seyedan & Mafakheri, 2020). According to Feizabadi (2022), this makes demand forecasting a crucial process in supply chain management, with high forecast accuracy directly translating into enhanced supply chain performance. Ingle et al. (2021) state that demand forecasting influences all parts in a supply chain from sourcing to production planning and inventory fulfilment. Demand forecasting can be defined as a process that uses historical data, statistical tools, machine learning and market analysis to predict future product demand (Harahap et al., 2025). Ingle et al. (2021) classify demand forecasting as a field of predictive analytics that is utilized to understand and predict customer demand.

According to Babai et al. (2022), demand can be forecasted in different granularities and time horizons depending on the specific objectives of the business. For example, short-term horizon demand predictions on a product level can be done to support operative and tactical decisions while long-term high-level forecasts can be used to guide strategic initiatives (Babai et al., 2022). There are multiple methods that can be applied to forecast demand, and method selection should be done based on data characteristics and the forecast horizon (Mediavilla et al., 2022). The methods are categorized into traditional statistical models, machine learning models, deep learning models and hybrid models (Ingle et al., 2021). Different demand forecasting models and algorithms are examined in more detail in chapter 2.3. This chapter focuses on the benefits that accurate demand forecasting can provide to businesses, followed by a review of the most common challenges associated with the process.

2.2.1 Benefits of efficient demand forecasting

Well-developed demand forecasting yields significant advantage and benefits for businesses (Feizabadi, 2022). According to Anitha et al. (2023), accurate forecasting of demand enables companies to optimize their operations, reduce costs and increase customer satisfaction**.** The main potential benefits are illustrated in figure 5. This thesis groups the benefits into three main areas: strategic planning, operational efficiency and customer service, which together contribute to cost reductions and translate into competitive advantage. The effects of accurate demand forecasting on these main areas are examined in more detail in the following sections.

Figure 5 Benefits of successful demand forecasting (Harahap et al., 2025)

2.2.1.1 Strategic planning

Adekunle et al. (2021) define demand forecasting as a critical component of business strategy. It can be stated that efficient demand forecasting is the foundation for strategic planning and decision-making, guiding long-term goal setting and influencing business areas such as product development, financial planning and growth strategy (Chandran & Khan, 2024; Harahap et al., 2025). In the modern competitive market, accurate forecasting provides strategic advantage by enabling businesses to predict future market fluctuations and adjust their operations proactively, mitigating the risk of market volatility (Adekunle et al., 2021; Chandran & Khan, 2024). According to Ahmaridad (2025), Walmart has based their strategic decisions such as optimal inventory placement and dynamic pricing strategy relying on accurate demand forecasting. As mentioned, demand forecasting is among the main drivers of business strategy, which is why organizations continuously invest in advanced analytics and research to enhance the accuracy of forecasts to guide better decision-making and ultimately gain competitive advantage (Waller & Fawcett, 2013).

2.2.1.2 Operational efficiency

Chandran and Khan (2024) found a clear correlation between forecasting accuracy and the operational efficiency of a company. This can be seen as improved inventory management, efficient resource allocation and significant cost reductions enabled by proactive decision-making (Harahap et al., 2025). According to Amosu et al. (2024), accurate demand forecasting can optimize replenishment processes and reduce over- and understocking situations by aligning purchasing and stock levels with future customer demand, which leads to improved inventory management. For instance, one retail company was able to reduce excess inventory by 30% with efficient demand forecasting, lowering holding costs significantly (Amosu et al., 2024).

Accurate forecasts are also crucial for business resource allocation to ensure the alignment of production, workforce and finances with the current market situation (Adekunle et al., 2021). Adekunle et al. (2021) state that in service and production industries, demand forecasting can guide workforce planning to ensure staff levels meet the customer and production demand levels, that can lead to reducing unnecessary labor costs and overproduction. Accurate forecasts are crucial for streamlining operational processes, often resulting in significant cost reductions and increased efficiency across the whole supply chain (Harahap et al., 2025).

2.2.1.3 Customer service

Customer service is a crucial element of supply chain management, with the aim to ensure the availability or products and services for purchase or use where and when they are needed, focusing to improve areas such customer experience, satisfaction and loyalty (Mas’udin & Kamara, 2017; Fawcett et al., 2007). Effective demand forecasting has a direct contribution to the improvement of these aspects (Chandran & Khan, 2024). Mandala (2024) states that the accuracy of demand forecasting impacts customer experience and brand image significantly, ultimately increasing customer loyalty. This is possible due to better availability, shorter lead times and lowered costs for customers (Chandran & Khan, 2024).

The accuracy of demand forecasting contributes directly to customer satisfaction through better availability of products and services with lowered costs due to optimized inventory management (Anitha et al., 2023; Böse et al., 2017). According to Adekunle et al. (2021), organizations can also improve customer retention by using insights derived from demand forecasts to identify customer preferences and buying patterns, that can then be utilized in developing personalized marketing strategies, such as targeted promotions. By improving the aspects of customer service, customers are more likely to leave positive reviews, become frequent buyers and brand advocates and less likely to turn to competitors, potentially leading to company growth (Amosu et al., 2024).

2.2.2 Challenges in demand forecasting

The benefits of demand forecasting come with a set of challenges. Often occurring challenges are related to data quality, fitness of the selected forecasting model, uncertainty and external factors as well as obstacles in implementation and scalability (Adekunle et al., 2021). The common challenges affecting forecasting accuracy and reliability are seen in figure 6. According to Adekunle et al. (2021), overcoming these challenges is crucial for developing effective demand forecasting systems. The technicalities of data quality issues and challenges in balancing model complexity and interpretability were reviewed in chapter 2.1.3. This section focuses on the impact of insufficient data quality and inaccurate modelling on demand forecasting, followed by discussion about challenges regarding external factors like unexpected events and market uncertainty and their effect on demand forecasting. Lastly, issues connected with demand forecasting implementation and scalability are reviewed.

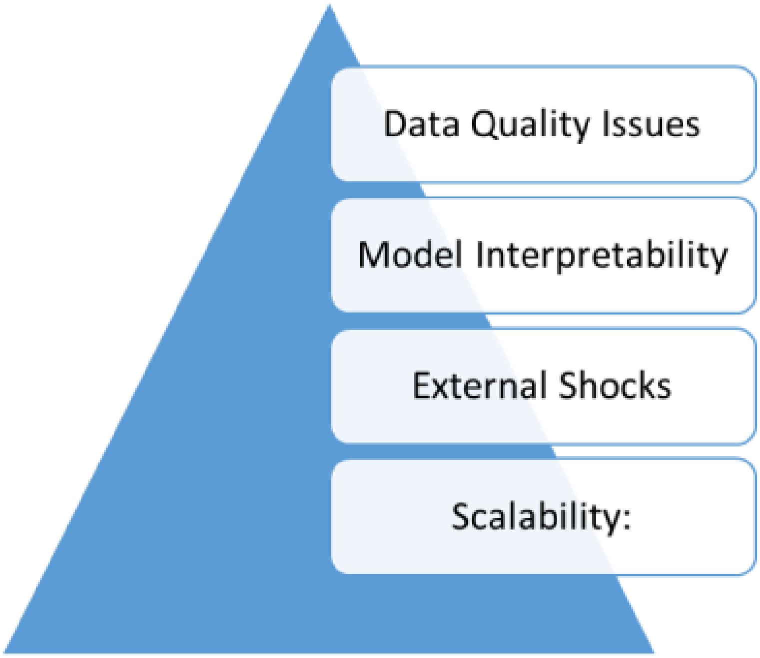


Figure 6 Common challenges in demand forecasting (Adekunle et al., 2021)

One of the core challenges in demand forecasting is the need for high-quality data (Amosu et al., 2024). Amosu et al. (2024) state that accurate forecasting of demand requires accurate and comprehensive historical demand data, market trends and external variables. Data quality issues such as inconsistency, missing values and inaccuracies can distort the patterns, trends and seasonality in the data, resulting in misleading and inaccurate forecasts (Adekunle et al., 2021). Ensuring high quality data for forecasting requires sophisticated processes for data collection and management, especially when integrating complex data from multiple sources such as internal data, demand records, market reports and social media trends (Amosu et al., 2024). Collecting and maintaining clean, structured and high-quality datasets is imperative for accurate demand forecasting. (Adekunle et al., 2021).

Another core challenge in demand forecasting is selecting a forecasting model that is suitable for the data and its purpose (Anitha et al., 2023). For example, a traditional statistical forecasting model like ARIMA may not be complex enough to capture the patterns in complicated data and thus risk underfitting, while a state-of-the-art ANN might tend to overfit or be difficult to interpret (Adekunle et al., 2021; Anitha et al., 2023). According to Amosu et al. (2024), organizations need to balance the complexity and accuracy of demand forecasting models with required levels of interpretability and transparency. This means balancing the bias-variance trade-off examined in chapter 2.1.3.4 by finding the optimal model complexity. Accomplishing this enables business stakeholders to trust the predictions and use them to make informed decisions (Amosu et al., 2024).

Traditional demand forecasting models make predictions based on historical patterns and relationships in the data (Amosu et al., 2024). However, according to Adekunle et al. (2021), unprecedented and abrupt external shocks like economic crises, pandemics, geopolitical events or natural disasters are difficult to predict with traditional models as they have often not occurred in the model's training data. Aforementioned events and quickly changing customer preferences can lead to significant alterations in demand trends and patterns, making past training data less relevant and leading to decreased forecasting performance (Adekunle et al., 2021; Anitha et al., 2023). This can be combatted by implementing complex dynamic models that can include real-time data in decision-making to predict and adjust to unprecedented changes (Adekunle et al., 2021). Amosu et al. (2024) suggest that forecasting models could be complemented with human expertise and judgment in scenarios where historical data is suspected to not fully capture the demand uncertainties.

Integrating and implementing advanced forecasting models with existing systems inside an organization is a critical challenge in demand forecasting (Amosu et al., 2024). Modern demand forecasting systems need to be designed to be scalable enough to process large volumes of real-time data reliably in production while simultaneously providing the flexibility to rapidly experiment and develop new algorithms (Böse et al., 2017; Adekunle et al., 2021). According to Adekunle et al. (2021) and Amosu et al. (2024), building, implementing and maintaining advanced forecasting systems is a complex task and requires a significant amount of computational resources, technical expertise and staff training, making it a costly investment to maximize the potential benefits. However, the long-term benefits of advanced demand forecasting often outweigh the initial investment (Amosu et al., 2024).

2.3 Models for demand forecasting

Demand forecasting models can be classified into four main categories: univariate time series models, machine learning models, deep learning models and hybrid models (Ingle et al., 2021). Univariate time series models such as ARIMA have been the foundation of demand forecasting for a long time because of their interpretability and ability to use demand trends and seasonality to generate predictions from datasets where time is the only independent variable (Hyndman & Athanasopoulos, 2018; Ingle et al., 2021; Zhang et al., 2024). ML algorithms like Random Forests and XGBoost transformed the demand forecasting field by making it possible to extract and analyze information from large multivariate datasets to uncover hidden patterns and relationships, leading to more robust and accurate models (Sistla et al., 2024).

The advancement has continued with DL demand forecasting models like LSTM networks that are computationally expensive and need large amounts of data, but offer the possibility to automatically learn important features and relationships from raw data while also capturing complicated temporal dependencies, often leading to better performance compared against time series and ML models (Ingle et al., 2021; Zhang et al., 2024). According to Chung et al. (2023), hybrid models aim to combine different models in a cooperative nature, addressing the limitations of individual models by supplementing them with another model. Combining the advantages of two models like ARIMA-LSTM has demonstrated excellent results in recent research (Chung et al., 2023). The following sections provide a more detailed review of the four demand forecasting model types along with examination of the most widely used algorithms associated with each category, describing their advantages and limitations.

2.3.1 Univariate time-series models

Time series data can be defined as a set of quantitative observations measured over time and arranged in chronological order (Kirchgässner et al., 2012). In univariate time series, time is the only independent variable (Belcic and Stryker, 2025). According to Belcic and Stryker (2025), time series models analyze time series data and generate predictions of future values based on historical values in the sequence. Traditional time series models such as ARIMA and Holt-Winters Exponential Smoothing have widely been utilized for demand forecasting due to their interpretability and ability to capture trends and seasonality in the data (Zhang et al., 2024). Recently, a modern time series model Prophet has emerged, providing flexibility, ease of use and efficiency with irregular and incomplete data (Taylor & Letham, 2018). The three mentioned time series models are examined in the following sections.

2.3.1.1 ARIMA

ARIMA is one of the most frequently used time-series models in demand forecasting (Brown et al., 2013). According to Hyndman and Athanasopoulos (2018), the goal of ARIMA models is to describe the autocorrelations in the dataset by combining differencing (I) with autoregressive (AR) and moving average (MA) models. The combination is defined as an ARIMA(p, d, q) model, where p is the order of the AR model, d is the degree of first differencing involved and q is the order of the MA model (Hyndman & Athanasopoulos, 2018). The full model is mathematically illustrated as

$$y'_t = c + \phi_1 y'_{t-1} + \cdots + \phi_p y'_{t-p} + \Theta_1 \varepsilon_{t-1} + \cdots + \Theta_q \varepsilon_{t-q} + \varepsilon_t \, , \tag{1}$$

where y'_t is the differenced time series, c refers to the average change between consecutive data points, $\phi_1 \dots \phi_p$ are the AR model coefficients, $y'_{t-1} \dots y'_{t-p}$ are the lagged values of the differenced time series, $\Theta_1 \dots \Theta_q$ are the coefficients of the MA model, $\varepsilon_{t-1} \dots \varepsilon_{t-q}$ are the lagged forecast errors and ε_t is random noise.

ARIMA also has variations like Seasonal ARIMA, that extends ARIMA to support analysis of time series data that also has a seasonal component (Noble, 2024). According to Noble (2024), ARIMA models can be used to efficiently analyze time series data and predict future time series values. However, research has identified limitations such as linear assumptions, uncertainty in parameter selection, inaccuracy in long term forecasting as well as the inability to account for external factors besides historical target variable data (Liu, 2024). Liu (2024) suggests that ARIMA should be complemented with other models to account for the limitations.

2.3.1.2 Holt-Winters Exponential Smoothing

The Holt-Winters method, also known as triple exponential smoothing, is a time series forecasting method that can be applied to datasets that have trendy and seasonal characteristics (Ingle et al., 2021; Lima et al., 2019). Hyndman and Athanasopoulos (2018) state that it consists of the forecasting equation and three smoothing equations for the level ℓ_t, trend b_t and seasonal s_t components with respective smoothing parameters α, β and γ. There are two variations of the Holt-Winters model: additive and multiplicative, and they differ in the characteristics of the seasonal component (Ingle et al., 2021). The additive method is often used when seasonal variations are nearly constant and the multiplicative method is preferred when the seasonality varies proportional to the level of the series (Hyndman & Athanasopoulos, 2018).

The forecast at time $t + m$ in the additive method is computed with the following equation

$$F_{t+m} = \ell_t + b_t m + s_{t-L+m}\,, \tag{2}$$

while the same forecast in the multiplicative model is given by

$$F_{t+m} = (\ell_t + b_t m) s_{t-L+m}\,, \tag{3}$$

where m is the number of forecasts ahead, L is the length of the seasonal cycle, and ℓ_t, b_t and s_t are acquired from their respective smoothing equations.

The Holt-Winters method is effective and can provide reliable predictions in univariate demand forecasting even if the amount of available data is low, assuming that seasonality can be precisely determined (Berberich, 2020). Even though it has been applied for decades, its capability with modern data is limited due to similar reasons as ARIMA. Limitations include the assumption of linearity, difficulty in handling overlying seasonalities and the inability to include external factors for forecasting (Berberich, 2020). Berberich (2020) states that it is suggested to combine Holt-Winters with a suitable machine learning model to compensate for the limitations.

2.3.1.3 Prophet

Prophet is a modern time series forecasting model developed by Facebook that is built to focus on common features identified in a business context (Taylor & Letham, 2018). According to Taylor and Letham (2018), Prophet is designed to be effectively tuned with intuitive parameters that can be adjusted with no detailed knowledge of the underlying model. Prophet is a decomposable time series model that accounts for three main components: trend, seasonality and holidays by only using time as a regressor (Taylor & Letham, 2018). Prophet approaches forecasting problems by using a Bayesian-based curve fitting method to smooth the time series data and generate predictions (Guo et al., 2021). The Prophet model is mathematically illustrated as

$$y(t) = g(t) + s(t) + h(t) + \varepsilon_t \,, \tag{4}$$

where $g(t)$ represents the trend function that models non-periodic variations in the data, $s(t)$ models the seasonal variations, $h(t)$ represents the effect of potential irregular holidays, while ε_t stands for random noise or unexplained variability. Each component $g(t)$, $s(t)$ and $h(t)$ are derived from their individual underlying equations.

Prophet stands out from other traditional time series models due to its ease of implementation and flexibility with multiple seasonalities (Taylor & Letham, 2018). A significant advantage associated with Prophet is its robustness to outliers and its ability to provide fairly efficient forecasts even with incomplete and irregular data (Taylor & Letham, 2018). However, Guo et al. (2021) mention that Prophet encounters limitations when applied to complex forecasting problems due to its limited expression ability, lack of residual autocorrelation detection and its inability to utilize external features outside of the time series. Their article suggests that Prophet should be complemented with a suitable machine learning model to build a hybrid model, for example Prophet-SVR, to address the limitations.

2.3.2 Machine learning models

The concepts of machine learning and machine learning models have already been introduced in chapter 2.1. Demand forecasting typically deals with regression problems, meaning that the models are often classified under supervised learning (Ingle et al., 2021). According to Ingle et al. (2021), machine learning models often provide more accurate predictions than univariate time series models due to their increased adaptability to changes and ability to identify hidden patterns and relationships in multivariate data but require a bigger amount of data to train. This section begins by reviewing popular machine learning regression models used for demand forecasting, such as Linear Regression, Elastic Net Regression and SVR. It then examines widely used tree-based methods such as Decision Trees and Random Forests, followed by an assessment of XGBoost, a highly regarded gradient boosting model.

2.3.2.1 Linear Regression

Understanding Linear Regression is foundational to comprehend more complex machine learning models (James et al., 2023). It is a simple regression algorithm that linearly models the relationships between one or more independent variables and a continuous scalar dependent variable (Nasteski, 2017). The goal of Linear Regression is to provide accurate predictions by reducing the sum of square residuals between predicted and observed values (Hyndman & Athanasopoulos, 2018; Rahman Mahin et al., 2025). The model assumes a linear relationship between the independent variables and the dependent variable and is mathematically formulated as

$$y_t = \beta_0 + \beta_1 x_{1,t} + \beta_2 x_{2,t} + \cdots + \beta_k x_{k,t} + \varepsilon_t \,, \tag{5}$$

where y_t is the target variable at time t, β_0 is the intercept and ε_t represents the random error of the model. $x_1 \dots x_k$ are the numerical independent variables and $\beta_0 \dots \beta_k$ are the coefficients that quantify the influence each predictor variable has on the target variable.

The performance of Linear Regression is limited by its assumptions of linearity and homoscedasticity, and its sensitivity to outliers, correlated errors and multicollinearity (James et al., 2023). However, it has been widely used for statistical learning and inference for a long time (James et al., 2023). It is simple to implement and the results are easily interpretable, making it a good baseline model to compare against other demand forecasting models (Goar & Yadav, 2024). Linear regression is visually illustrated in figure 7.

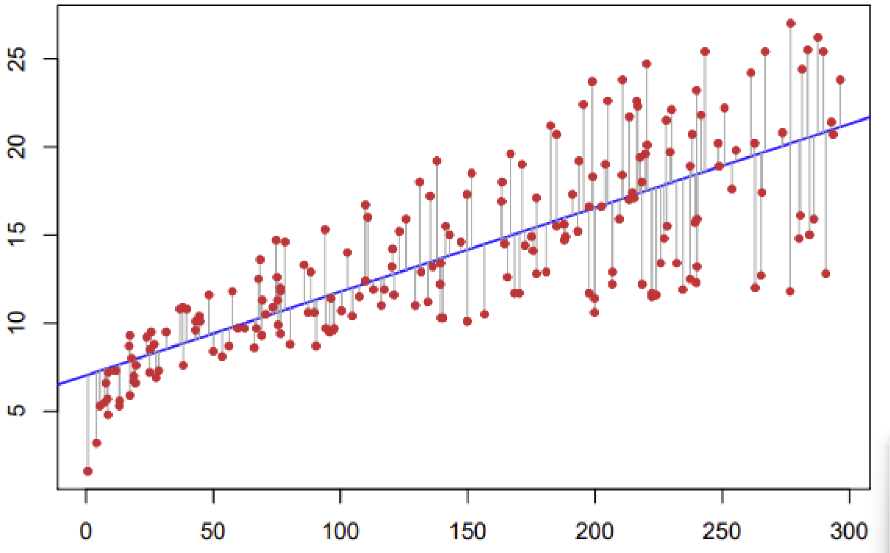


Figure 7 Linear Regression (James et al., 2023)

2.3.2.2 Elastic Net Regression

The Elastic Net algorithm is a regularized Linear Regression method that combines Lasso (L1) and Ridge (L2) penalties to provide an efficient solution for problems with high-dimensional data by managing multicollinearity and preventing overfitting (Chung et al., 2023; Rahman Mahin et al., 2025). These penalties address the multicollinearity limitation of Linear Regression. Lasso penalization is used for variable selection and Ridge penalization is used for coefficient shrinkage to reduce overfitting (Hans, 2011). Elastic Net identifies features that have a significant correlation with the dependent variable and assigns their weights accordingly, allowing the algorithm to automatically select or discard features from the model (Chung et al., 2023). According to Chung et al. (2023), this makes Elastic Net Regression an efficient tool to utilize with high-dimensional data that contains multicollinearity to discover the relationship between features and the target variable and prevent overfitting. However, besides decreased sensitivity to multicollinearity, Elastic Net Regression shares similar limitations as Linear Regression. The objective function of Elastic Net Regression is mathematically formulated as

$$\hat{\beta} = argmin_{\beta}(Y - X\beta)(Y - X\beta)^{T} + (1-\alpha)\sum_{i=1}^{n}|\beta_i| + \alpha\sum_{i=1}^{n}\beta^{2}{}_{i}\ , \qquad (6)$$

where $\hat{\beta}$ is the estimate of the regression coefficients that minimize the objective function, $(Y - X\beta)(Y - X\beta)^{T}$ is the residual sum of squares, $\sum_{i=1}^{n}|\beta_i|$ is the L1 norm, $\sum_{i=1}^{n}\beta^{2}{}_{i}$ is the L2 norm and α is the parameter that adjusts the trade-off between L1 and L2 penalties.

2.3.2.3 Support Vector Regression

SVR is a machine learning algorithm that aims to discover patterns in complex datasets and is often used in demand forecasting (Abolghasemi et al., 2020; Ingle et al., 2021). Ingle et al. (2021) state that SVR can be applied to nonlinear data by using a kernel function to project the data into a higher-dimensional space where the relationship can be linearly modelled. Polynomial, sigmoid and radial basis function (RBF) are among the most popular kernels (Ingle et al., 2021). According to Goar and Yadav (2024), SVR aims to estimate a function by fitting a tube with ε width to the data, splitting the data into two categories: data points inside and outside of the tube. This is visualized in figure 8.

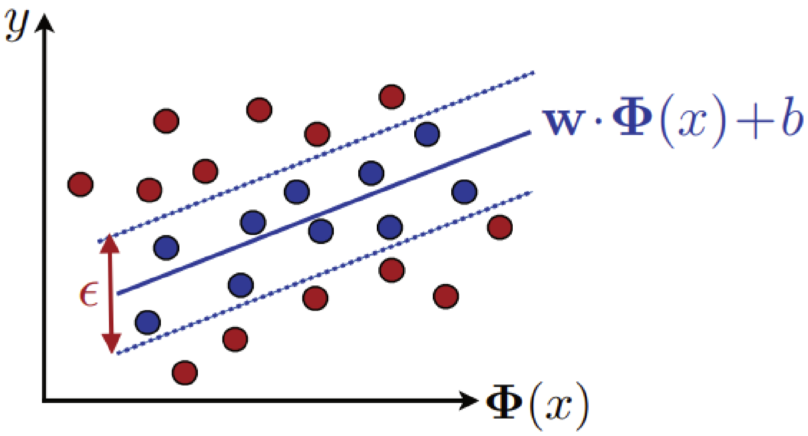


Figure 8 Support Vector Regression (Goar & Yadav, 2024)

SVR trains a symmetrical loss function that penalizes predictions that are located more than ε away from the estimated function, equally penalizing high and low misestimates (Awad & Khanna, 2015). The primal optimization problem of SVR is mathematically illustrated as

$$min\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i + \xi_i^* \,, \tag{7}$$

subject to

$$\begin{gathered} y_i - w^T x_i \leq \ \varepsilon + \xi_i^* \quad i = 1 \dots N \\ w^T x_i - y_i \leq \ \varepsilon + \xi_i \quad i = 1 \dots N \\ \xi_i, \xi_i^* \geq 0 \quad i = 1 \dots N \,, \end{gathered}$$

where w is the weight vector, C is the regularization parameter, ξ_i and ξ_i^* are slack variables that allow deviations beyond ε, x_i is the input feature vector and y_i is the actual output value. Solving this equation aims to approximate the target values while maintaining model simplicity and allowing error tolerance (Awad & Khanna, 2015).

According to Awad and Khanna (2015), advantages of SVR include its efficiency with high-dimensional data, ability to model nonlinear relationships, robustness to overfitting and good generalization ability. However, SVR highly depends on parameter selection, can be hard to interpret and may be computationally expensive when trained with a large amount of data (Goar & Yadav, 2024).

2.3.2.4 Decision Trees

Decision Trees is a machine learning algorithm that creates a tree-structured series of decision rules based on input variables to predict the target variable (Bi et al., 2019). The set of decision rules is derived from the training data by the algorithm (Joshi, 2023). The depth of the tree, its dimensions and termination point are parameters that the algorithm optimizes to minimize error (Joshi, 2023). A simple decision tree consists of a root node, branches, internal nodes and leaf nodes (IBM, n.d.). According to Goar and Yadav (2024), the prediction process of a Decision Tree starts by feeding the input data to the root node, from where the tree is recursively split to branches depending on the decision rules of internal nodes. This is repeated until a leaf is found, and the value of the leaf is then used as the predicted output of the given data point (Goar & Yadav, 2024). The hierarchical structure and prediction process of a simple Decision Tree can be seen in figure 9.

The prediction function of a regression tree that is partitioned into M leaf nodes $R_1 \dots R_m$ is mathematically illustrated as

$$f(x) = \sum_{m=1}^{M} c_m I(x \in R_m)\,, \tag{8}$$

where I is the indicator function and c_m is the constant prediction value, which is usually the mean of target variables in the training data that have been categorized into leaf node R_m (Hastie et al., 2009).

According to Louppe (2014), Decision Trees provide easily interpretable predictions with often reliable accuracy while also being fast to train. Their non-parametric nature enables them to model complex relationships, even in multicollinear data (Louppe, 2014; IBM, n.d.). However, Goar and Yadav (2024) state that the performance of Decision Trees often lacks when compared against state-of-the-art algorithms. This is mainly due to their greedy nature, tendency of overfitting and sensitivity towards noisy data (Bi et al., 2019). A simple Decision Tree is visually illustrated in figure 9.

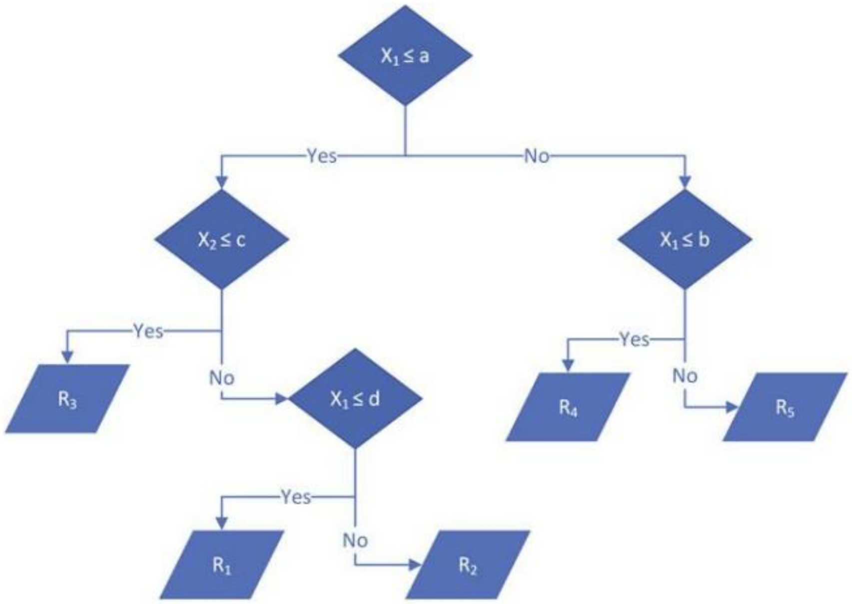


Figure 9 Simple decision tree (Joshi, 2023)

2.3.2.5 Random Forest

Random Forests are ensembles of Decision Trees and are applied widely in different industries, yielding efficient results (Cunningham et al., 2008). Random Forests use a bagging approach to build multiple de-correlated Decision Trees from bootstrapped training samples with a random sample of features (James et al., 2013; Hastie et al., 2009). The output of the Random Forest is then computed as the mean of the Decision Tree outputs (Hastie et al., 2009), and can be mathematically illustrated simply as

$$f(x) = \frac{1}{J}\sum_{j=1}^{J} h_j(x)\,, \tag{9}$$

where J is the number of decision trees and $h_j(x)$ represents the outputs of the trees reviewed in equation 8. The structure of a random forest is shown in figure 10.

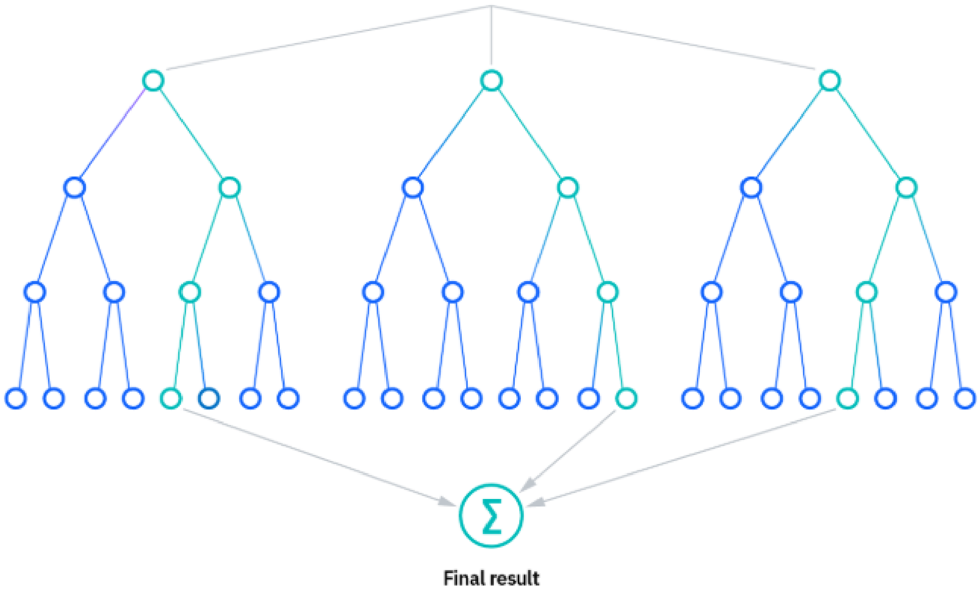


Figure 10 Random Forest (IBM, n.d.)

Key advantages of Random Forests are their robustness to overfitting, ease of implementation and their ability to assess feature importance and estimate generalization error (Cutler et al., 2011). On the other hand, IBM (n.d.) mentions complex interpretability and the time-consuming and resource-heavy nature of random forests as their limitation, especially with large datasets.

2.3.2.6 XGBoost

Extreme gradient boosting, known as XGBoost, is a state-of-the-art gradient boosting decision tree algorithm. According to Harikrishnan and Sreedharan (2025), it is an advanced machine learning algorithm that delivers efficient solutions for regression and classification tasks, especially with large datasets. This is supported by Ingle et al. (2021), who state that XGBoost models perform efficiently with predictive tasks like time series forecasting with big data. Ingle et al. (2021) define XGBoost as an ensemble tree technique that boosts multiple decision trees by constructing new trees using the residual errors from previous trees.

XGBoost model consisting of K Decision Trees can be mathematically expressed as

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), f_k \in F \ , \tag{10}$$

where $\hat{y}_i$ is the predicted output, f_k represent the Decision Trees and K is the total number of trees in the space of regression trees F.

The main goal of XGBoost is to optimize the objective function that consists of a loss function and a regularization term (Ahmetoglu & Das, 2022; Zhang & Jánošík, 2024). The loss function $l(y_i, \hat{y}_i)$ computes the difference between the predicted and actual values, and the regularization term $\Omega(f_k)$, which includes the number of leaves in a tree and the magnitude of leaf weights, controls model complexity and prevents overfitting (Ahmetoglu & Das, 2022). The objective function is mathematically formulated as

$$Obj = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{t} \Omega(f_k) \, . \tag{11}$$

XGBoost is known to have enhanced performance, generalization ability, scalability and computing speed compared to other ML methods (Lai et al., 2021; Zhang & Jánošík, 2024). However, XGBoost also has its limitations. Firstly, Velarde et al. (2023) state that XGBoost's predictive power quickly decreases on small datasets. XGBoost also needs efficient feature engineering to capture the temporal dependencies in demand forecasting training data (Thejovathi et al, 2024). Therefore, researchers suggest that XGBoost could be hybridized with time-series or DL models to account for these limitations (Islam et al., 2024).

2.3.3 Deep learning models

Foundations of deep learning models have been briefly reviewed previously in chapter 2.1. This section focuses on the applications of deep learning in the field of demand forecasting. According to Aguiar-Pérez and Pérez-Juárez (2023), deep learning models are efficient for demand forecasting due to their ability to automatically learn patterns and important features from large, complex and multisource data. Ingle et al. (2021) state that deep learning models have proven better predictive capabilities and better accuracy compared to machine learning models.

However, DL models require more available data for training and are often computationally expensive (Ingle et al., 2021). Every neural network consists of layers of neurons: input layer, weighted hidden layer(s) and an output layer (IBM, 2021). In a simple NN, the input layer takes the input data and passes it to the hidden layer neurons for computation, and the output layer provides the prediction (IBM, 2021). A simple feedforward ANN is visually illustrated in figure 11. The deep learning models that are introduced in this section include RNN and LSTM, followed by a review of a state-of-the-art TFT architecture.

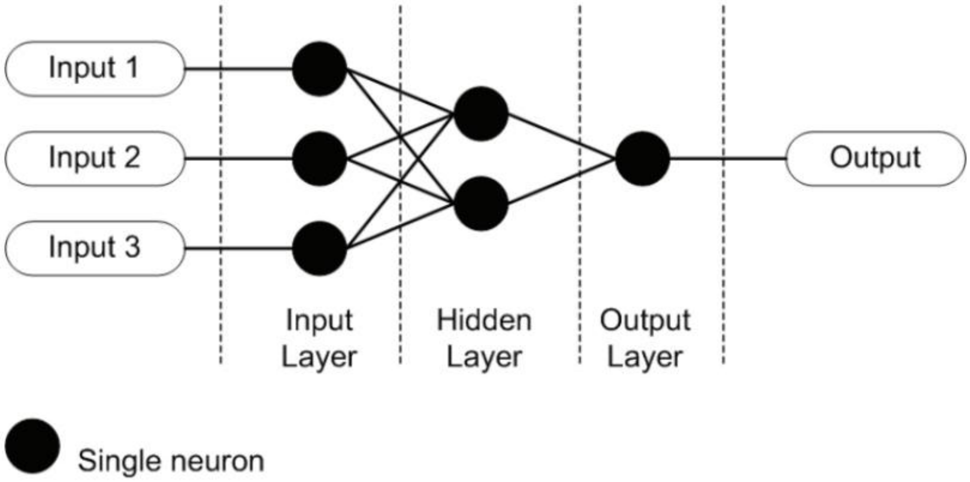


Figure 11 Simple ANN (Krenker et al., 2011)

2.3.3.1 RNN

Recurrent Neural Networks are deep learning models that are specifically designed to process sequential data (Mienye et al., 2024). According to Mienye et al. (2024), RNNs differ from feedforward ANNs due to them being able to memorize previous inputs by utilizing their internal state to efficiently process sequential input data, making them efficient for time series forecasting. As seen in figure 12, the neurons in RNNs have recurrent connections that allow the information to be cycled inside the network. In simple RNNs, the output is dependent on the latest input and the hidden state from the previous computation (Stryker, 2024).

The output of RNN is computed according to the following formula:

$$y_t = \sigma_y\left(W_{hy}h_t + b_y\right), \tag{12}$$

where W_{hy} is the weight matrix between the input and hidden layers, h_t is the hidden state, b_y is the bias vector and σ_y is the activation function, which is often the sigmoid function, hyperbolic tangent function or the rectified linear unit.

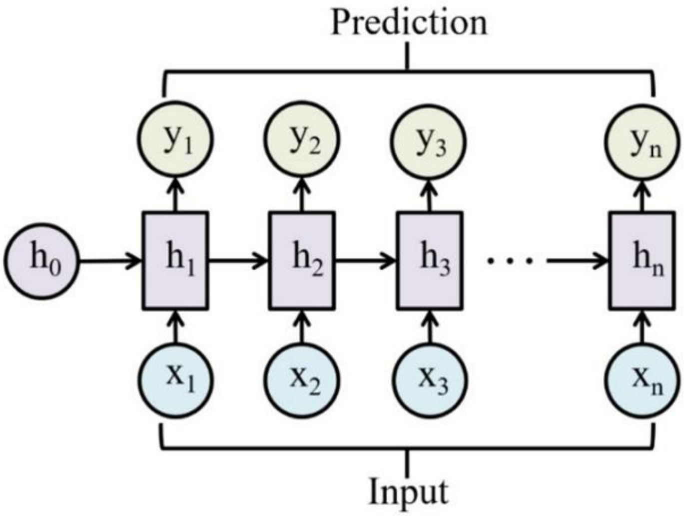


Figure 12 Simple RNN architecture (Mienye et al., 2024)

According to Stryker (2024), standard RNNs are highly efficient when processing sequential data in real time, where there is only one input and predictions are based on recent inputs. The ability to learn features and dependencies from data makes it possible for RNNs to model any dynamic system, leading to excellent performance in different fields (Salehinejad et al., 2018). According to Salehinejad et al. (2018), the main limitations of basic RNNs are their inefficient training and issues in learning long term dependencies due to the unstable gradient problem, which refers to the tendency of gradients to vanish or explode during backpropagation through time.

2.3.3.2 LSTM

Long Short-Term Memory networks are enhanced RNNs that are specifically designed to handle the unstable gradient problem (Berberich, 2020). This is possible due to gating mechanisms that control the information flow in the network, enabling LSTM to maintain and update the internal state over long periods (Mienye et al., 2024). Berberich (2020) describes this as the cell state (C_t) that represents the long-term memory, while the internal state (h_t) is the short-term memory. This innovative structure allows LSTM to model both short and long-term dependencies efficiently (Berberich, 2020; Mienye et al., 2024). Each LSTM cell consists of input (i_t), output (O_t) and forget (f_t) gates that control the cell state and hidden state to determine the amount of input to be considered, how much of the previous state is forgotten and how much of the cell state to output (Mienye et al., 2024; Salehinejad et al., 2018). The architecture of a LSTM network can be seen in figure 13. In the figure, X_t represents the input vector and L_t represents the candidate memory cell.

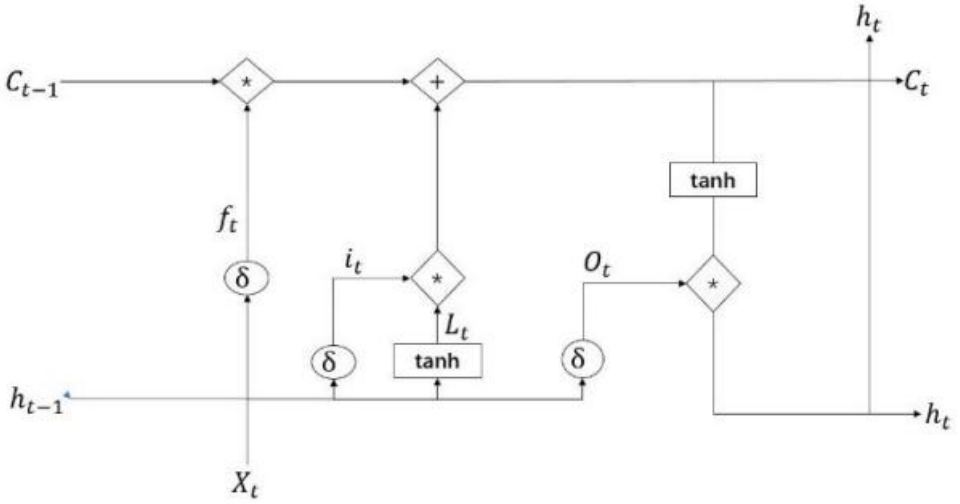


Figure 13 LSTM network (Wen & Li, 2023)

Ingle et al. (2021) states that LSTM is an efficient modelling approach for time series demand forecasting. This is because its flexibility and ability to handle unstable and inconsistent data as well as volatile demand scenarios (Ingle et al., 2021). It can also utilize multivariate input data to capture long and short term dependencies to provide accurate forecasts (Ingle et al., 2021). However, the advantages come with increased computational costs and high model complexity that results in predictions that are difficult to interpret (Berberich, 2020; Salehinejad et al., 2018). Berberich (2020) also states that LSTM is also not immune to problems in generalization, highlighting its tendency to overfit if the size of the dataset is not sufficient.

2.3.3.3 TFT

Temporal Fusion Transformer is a state-of-the-art attention-based deep neural network model designed specifically to process multivariate time series data where variables and features change over time (Karthika et al., 2025). It integrates LSTM in its complex architecture to provide high-performing multi-horizon forecasting and interpretable explanation for temporal dynamics (López Santos et al., 2022). According to Lim et al. (2020), TFT architecture consists of specialized blocks that select important features and exclude irrelevant information, enabling high performance in many applications.

Gating Residual Networks (GRN) allow TFT to exclude unused components and avoid unnecessary nonlinear processing, providing adaptive model depth and complexity to fit different datasets and scenarios (Lim et al., 2020). Variable selection networks allow the TFT model to select relevant input features for each time step (Lim et al., 2020; López Santos et al., 2022). Static covariance encoders incorporate static variables into the model and allows the network to learn the relationships between them in the past and the future (Huy et al., 2022). TFTs utilize temporal processing to learn long- and short-term dependencies from known and observed time-varying inputs (Lim et al., 2020). According to Lim et al. (2020), local processing is done in a sequence-to-sequence layer while long-term patterns are captured with an interpretable multi-head attention block. Finally, according to López Santos et al. (2022), TFTs minimize a quantile loss function, enabling them to provide a probabilistic forecast with confidence intervals to determine the likelihood of the target values. The architecture is visually illustrated in figure 14.

Huy et al. (2022) state that TFT models have proven to be more advanced in time series data processing and forecasting when compared against traditional RNN models. It also has better interpretability due to the ability to continuously identify relevant features, making it possible to explain the model’s predictions (López Santos et al., 2022). López Santos et al. (2022) imply that this property makes the model more trustworthy and easier to implement, making it a crucial part of explainable AI. The limitations of TFTs are mostly connected to the need for large amounts of data for model training, with insufficient data leading to challenges in temporal understanding, generalization and decline in model performance (Khanal et al., 2024).

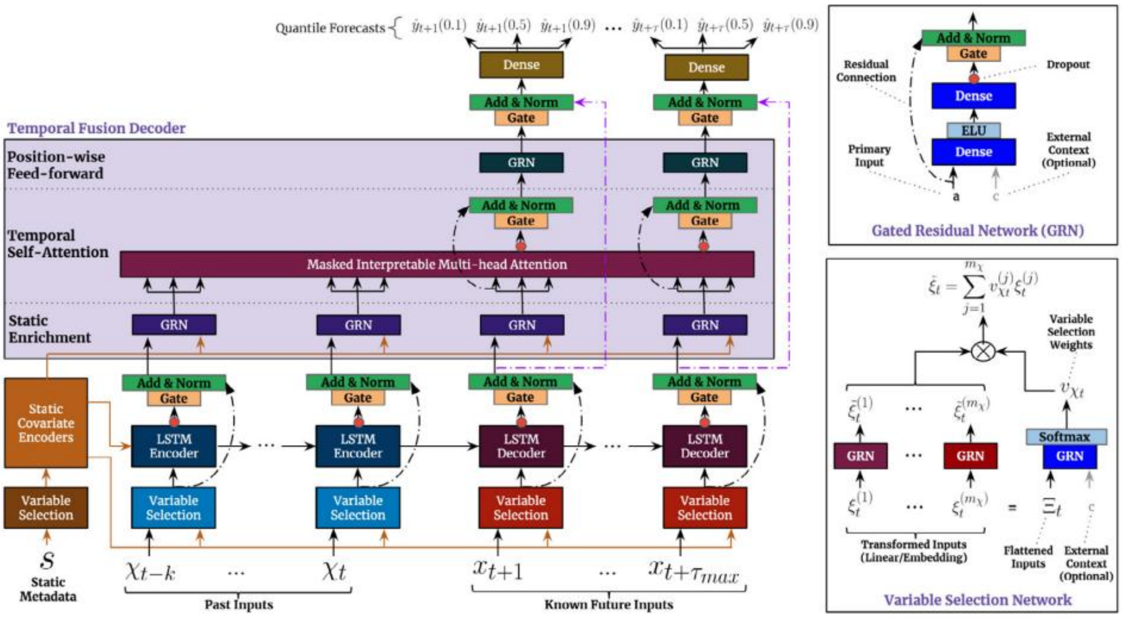


Figure 14 TFT architecture (Lim et al., 2020)

2.3.4 Hybrid models

Efficient model selection for a demand forecasting problem can be challenging in unique scenarios due to every model having their respective advantages and limitations (Ingle et al., 2021). According to Ahmed and Husien (2024), combining two or more algorithms into one forecasting model can address this issue and lead to novel solutions, possibly generating more efficient and accurate outputs compared to isolated models. These types of models are referred to as hybrid models (Ingle et al., 2021). Aburto and Weber (2007) state that the fundamental goal of hybrid models is to combine two or more different algorithms so that the models complement each other by using their advantages to compensate for the other algorithm's limitations. Examples of common limitations that can be covered with model combinations are insufficient training data, assumption of linearity and high computational cost (Berberich, 2020).

Hybrid models often combine traditional statistical models with machine learning or deep learning algorithms (Ingle et al., 2021). Instances of algorithmic combinations used in hybrid models are ARIMA-ANN, ARIMA-SVR, Holt-Winters-SVR and Holt-Winters-LSTM (Berberich, 2020; Brentan et al., 2017; Ingle et al., 2021). Exploration of hybrid models in demand forecasting offers multiple potential advantages. Ingle et al. (2021) found that hybrid models are an efficient approach in short-term demand forecasting when market conditions are unstable. Brentan et al. (2017) mention that hybrid models can increase model accuracy and reduce overfitting issues. Abolghasemi et al. (2020) found hybrid models to provide robust forecasts at different volatility levels. However, they are often complex to develop and understand (Ingle et al., 2021). According the Berberich (2020), researchers have stated that hybrid models have a promising future in time series forecasting and should always be assessed in model selection.

2.4 Summary of theoretical framework

The theoretical framework of this thesis uses machine learning fundamentals and core concepts and integrates them with time-series modelling to form the foundation for empirical research. This integration leads to hybrid models, which aim to combine the strengths of both modelling approaches to learn patterns from data and provide accurate outputs. CRISP-DM is the methodological framework that provides a systematic approach to guide the model development process from business and data understanding to data preparation, model development and finally model evaluation. The goal of model development in this thesis is to improve the efficiency of demand forecasting, which is an analytical process that aims to predict and understand future demand.

Together these foundational elements support the main topic of the thesis: Development and comparison of machine learning, time-series, and hybrid models to improve the accuracy of product demand forecasting. The synthesis of the theoretical framework is visually illustrated in figure 15, where the dark, bolded elements represent the core keywords of this thesis, and the lighter time-series modelling element serves as a supporting theoretical pillar for this research.

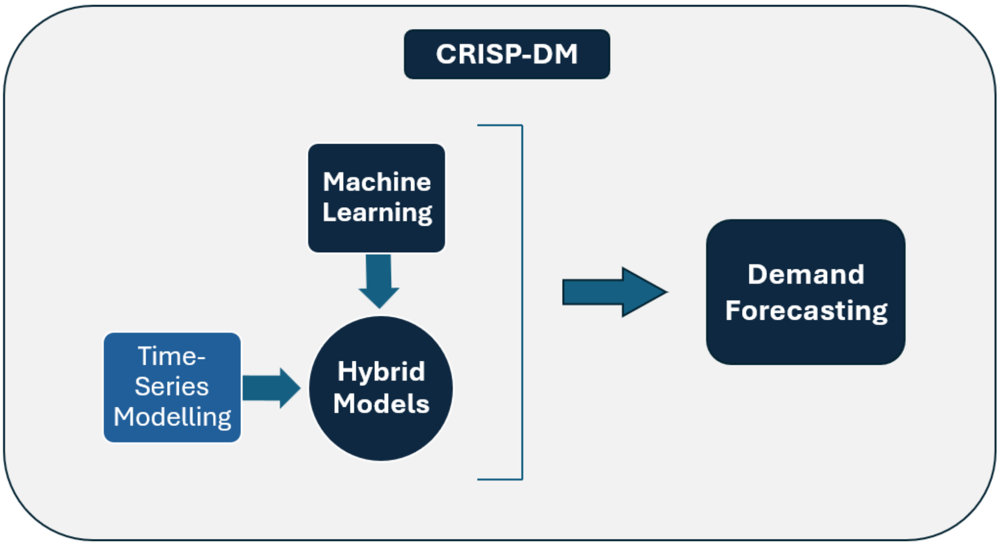


Figure 15 Synthesis of the theoretical framework

3 Research methodology

This chapter outlines the methodological framework of this study by adopting the research onion presented by Saunders et al. (2023). The research onion is visually shown in figure 16. The first section of this chapter describes and justifies the philosophy, approach, methodological choice, strategy and time horizon of the conducted research. The second section describes the techniques and procedures by walking through the CRISP-DM development methodology, focusing on data collection, analysis, preprocessing, modelling and evaluation. Lastly, in the third section, the quality of the study is discussed, describing the measures that were taken to ensure the accuracy, validity, reliability and reproducibility of the research process.

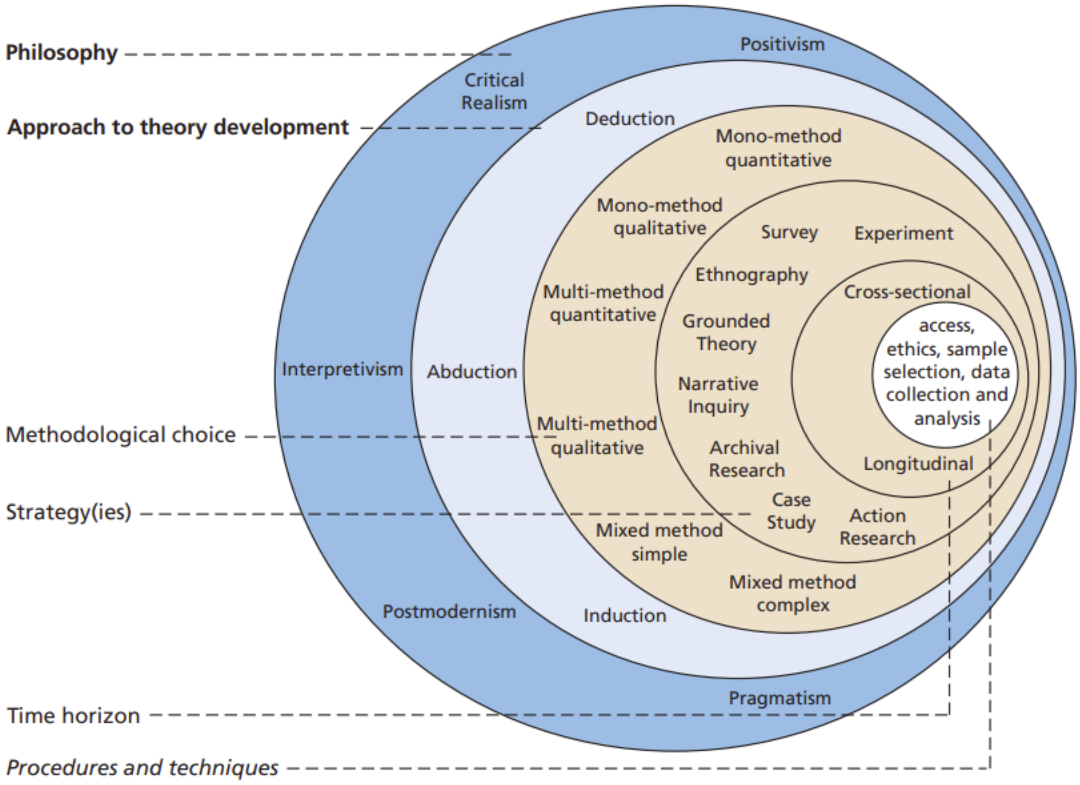


Figure 16 Research onion (Saunders et al. 2023)

3.1 Research design

The aim of this study was to develop a demand forecasting model that performs most efficiently on the data and to identify and analyze the key demand drivers of the selected product. This was done by collecting, processing and analyzing data, followed by developing, evaluating and comparing multiple time-series, machine learning and hybrid demand forecasting models. Therefore, the study is based on the positivism research philosophy, where the main goal is to find explanatory associations and causal relationships that lead to prediction and control of the studied phenomenon (Park et al., 2020). This thesis uses a deductive research approach, where known theoretical concepts from ML, time-series and hybrid modelling are applied to a real-world dataset to quantitatively evaluate their performance and effectiveness in improving demand forecasting accuracy.

The research was methodologically carried out as a single case study following a mono-method quantitative research design. A case study is an empirical research design that investigates a contemporary phenomenon within its real-life context (Schell, 1992). According to Yin (2009), a case study is particularly useful when the goal is to gain in-depth understanding of a phenomenon and the boundaries between phenomenon and context are not clearly evident or when multiple sources of information are used. Forecasting product demand is a contemporary topic, as it is crucial for companies to gain competitive advantage in the modern market. Demand is context sensitive and depends on the company, product and market conditions, making a case study a suitable method to gain deep understanding of demand as a real-life phenomenon.

Moutinho and Huarng (2013) define quantitative research as systematic empirical investigation of phenomena by using statistical, mathematical, or computational techniques. Similarly, Basias and Pollalis (2018) state that in quantitative research, the data is collected, processed and presented in a quantitative format and the features and their relations are also expressed quantitatively. ML and statistical analysis are methods that quantify the relationships between variables and transform quantitative inputs into quantitative outputs. The characteristics of quantitative research design are aligned with the nature of this research, making it an appropriate methodological approach. Regarding the quantitative data, this research uses a cross-sectional time-horizon, as the modelling and analysis is performed on a single collected dataset rather than across multiple repeated observations over time.

3.2 CRISP-DM approach

This research was mainly conducted as a proof of concept for data science, machine learning and advanced analytics for the case company, thus an industry standard framework CRISP-DM was applied. CRISP-DM is the most common methodology for data science projects in industry (Shimaoka et al., 2024). It describes the lifecycle of a data science project in six steps: Business understanding, data understanding, data preparation, modelling, evaluation and deployment (Martinez-Plumed et al., 2021). The CRISP-DM framework is visualized in figure 17. The deployment phase will be excluded from this thesis, as it is done separately by the company. This chapter will describe the procedures and techniques taken in detail from business and data understanding to data preparation and modelling, giving the reader a clear understanding of how the research project was executed. The evaluation phase will be described in the results section of this thesis.

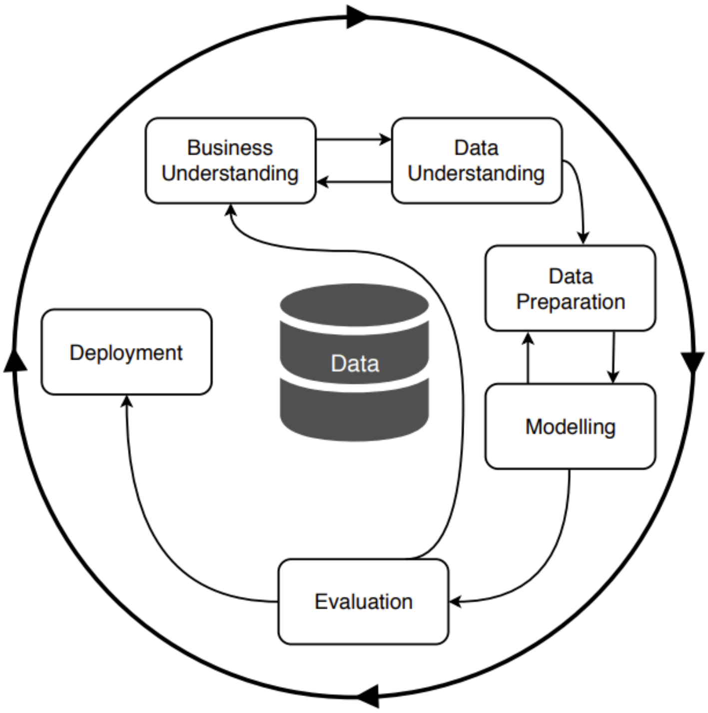


Figure 17 CRISP-DM framework (Martinez-Plumed et al., 2021)

3.2.1 Business understanding

The business understanding phase focused on defining the business problem and translating it to project objectives. The background and problem definition are described in the introduction chapter of this thesis. The foundation for the research project was established by discussing with the company’s sales managers to gain understanding about the demand of the product, current forecasting methods and the problems associated with them. The discussions were not documented for this thesis, as the goal was to gain preliminary understanding of the business case. The predictions generated with the current baseline forecasting method for January to September 2025 were available to be used in this study as a baseline against the alternative forecasting methods.

The company’s set objectives for this development project were similar to the research objectives of this study. The main objective was to develop an interpretable demand forecasting model with a three-month prediction horizon for the chosen product. The second objective was to identify and analyze key demand drivers to deepen the understanding of demand in the company and increase the trustworthiness of the developed model. The final industry objective of this project was to demonstrate the business value of data science, ML and advanced analytics solutions which was done by comparing the developed models against the baseline forecast and evaluating the differences in accuracy.

3.2.2 Data understanding

The data understanding phase included identifying and collecting relevant data for forecasting from internal and external sources as well as exploring the data through EDA. The goal was to gain deeper understanding of the data by describing the initial collected dataset, assessing its quality and analyzing it to identify trends, patterns and relationships that could have predictive potential for demand forecasting. The following sections describe these actions in more detail.

3.2.2.1 Data collection

The internal data used in this study was collected from the case company's database complemented with external data from Eurostat, ECB and Yahoo Finance APIs. The timeframe covered by the raw dataset was 1.1.2015-30.9.2025, as it was the available timeframe for demand and pricing data for the chosen product. The collected data was aggregated to a monthly level due to the forecasting horizon being three months. If there were multiple data points for a single month, the data points were summed or averaged to a monthly average. The result of collecting and aggregating data was a raw time-series dataset with 129 rows and 15 columns. The initial raw dataset consisted of variables listed in table 1 below.

Table 1 Raw dataset variable information

Variable	**Description**	**Type**	**Source**
sales	Monthly sales	Numeric	Internal
sales price	Monthly average sales price	Numeric	Internal
year	Year of the sample	Numeric	Internal
month	Month of the sample	Categorical	Internal
covid	Covid indicator	Binary	Internal
crisis	Crisis indicator	Binary	Internal
cl=f	Monthly average crude oil price	Numeric	YF
qg=f	Monthly average natural gas price	Numeric	YF
eur/usd	Monthly average EUR/USD index	Numeric	YF
construction confidence	Construction confidence indicator	Numeric	Eurostat
industry confidence	Industry confidence indicator	Numeric	Eurostat
consumer confidence	Consumer confidence indicator	Numeric	Eurostat
economic sentiment	Economic sentiment indicator	Numeric	Eurostat
interest	3-month average interest rate	Numeric	Eurostat
inflation	HICP inflation rate	Numeric	ECB

3.2.2.2 Exploratory data analysis

This section covers the EDA that was conducted to understand demand as a statistical phenomenon and to guide the modelling decisions. The demand was first described using core descriptive statistics shown in table 2. The standard deviation of the demand is nearly as large as the mean as can be seen from the coefficient of variation, indicating high variability and noisiness of demand. The high variability can also be seen from the min-max and interquartile ranges. Skewness and kurtosis values indicate a positively skewed distribution with heavy tails and outliers. The distribution and prominence of outliers are visually illustrated in the sales distribution and boxplot in figure 18. The main conclusion that was drawn from these statistics was that the nature of demand is volatile and it contains extreme outliers, making accurate forecasting challenging.

Table 2 Core descriptive statistics

Statistic	**Value**
Mean	67714,32
Standard deviation	65581,56
Coefficient of variation	0,97
Min - Max	25 - 393250
Interquartile range	28875 – 84675
Skew	1,87
Kurtosis	4,00

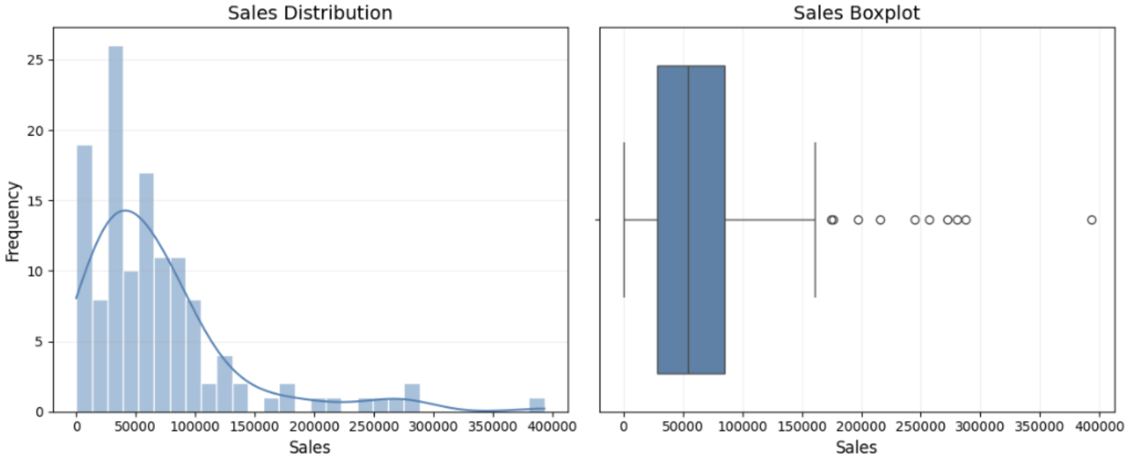


Figure 18 Distribution and boxplot of sales

A seasonal decomposition of the demand time series data is displayed in figure 19. It separates demand into trend, seasonal and residual components. The observed plot shows the demand as a time-series, highlighting the continuous fluctuations and increasing volatility. The trend component illustrates a clear upward trajectory in demand starting from mid-2024. The seasonal component indicates that the demand has historically followed a yearly seasonal pattern. Lastly, the residual component shows random demand fluctuations that are not captured by the trend or seasonal components, confirming that the demand variability is also subject to external and unpredictable factors and including them in a predictive model can potentially improve performance. The residual component also has higher fluctuations starting from mid-2024. The growth trend and larger residual fluctuations indicate that a regime change has occurred in 2024, making forecasting the demand more challenging.

Figure 19 Seasonal decomposition of demand

Figure 20 visualizes the autocorrelation and partial autocorrelation of demand. ACF measures the correlation between demand and its past values, also referred to as lags (Kis, 2024). PACF measures the correlation between demand at two time points, accounting for the intermediary demand values (Kis, 2024). The ACF in figure 20 shows a strong positive correlation at lags 1-6, indicating that the time-series has short-term memory with lags 1-6 influencing current demand. The PACF plot shows significant spikes at lags 1-3 and then declines quickly in larger lags, suggesting that most of the direct autocorrelation influence comes from the first lag, followed by lags 2 and 3.

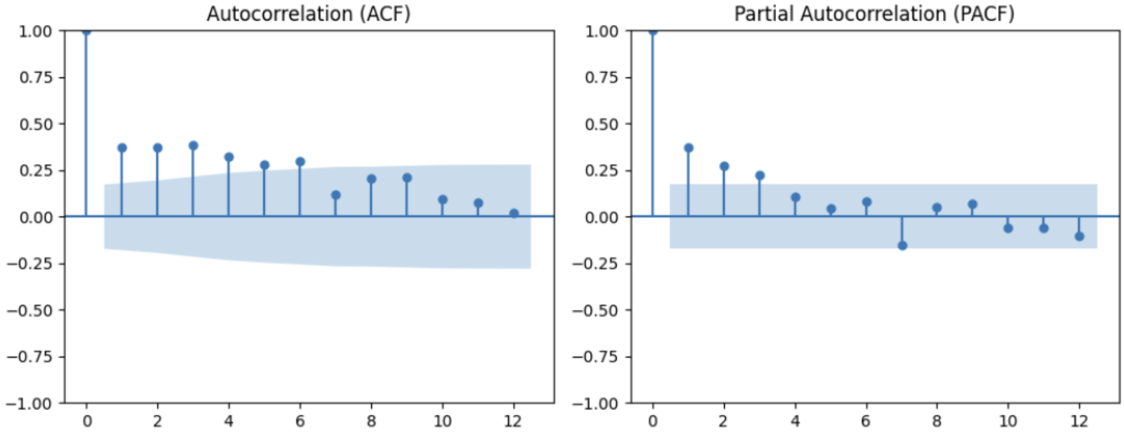


Figure 20 ACF and PACF plots of demand

After univariate analysis of demand, the initial dependencies between raw input variables and demand were measured using Spearman correlation to gain understanding of potential key variables. The correlation heatmap is shown in figure 21. The correlations suggest that sales price and year are the variables that have the most influence regarding sales. However, macroeconomic indicators such as confidence indexes, inflation and interest rates may act as leading variables that need to be lagged to have predictive power. The correlation heatmap also shows significant multicollinearity between the input variables, highlighting the importance of feature selection in the data preparation phase.

The exploratory data analysis provided crucial information regarding the demand time series and its dependencies with suspected explanatory variables. These insights were used to guide the data preparation phase, including feature engineering and feature selection as well. The analysis also provided direction for the modelling phase.

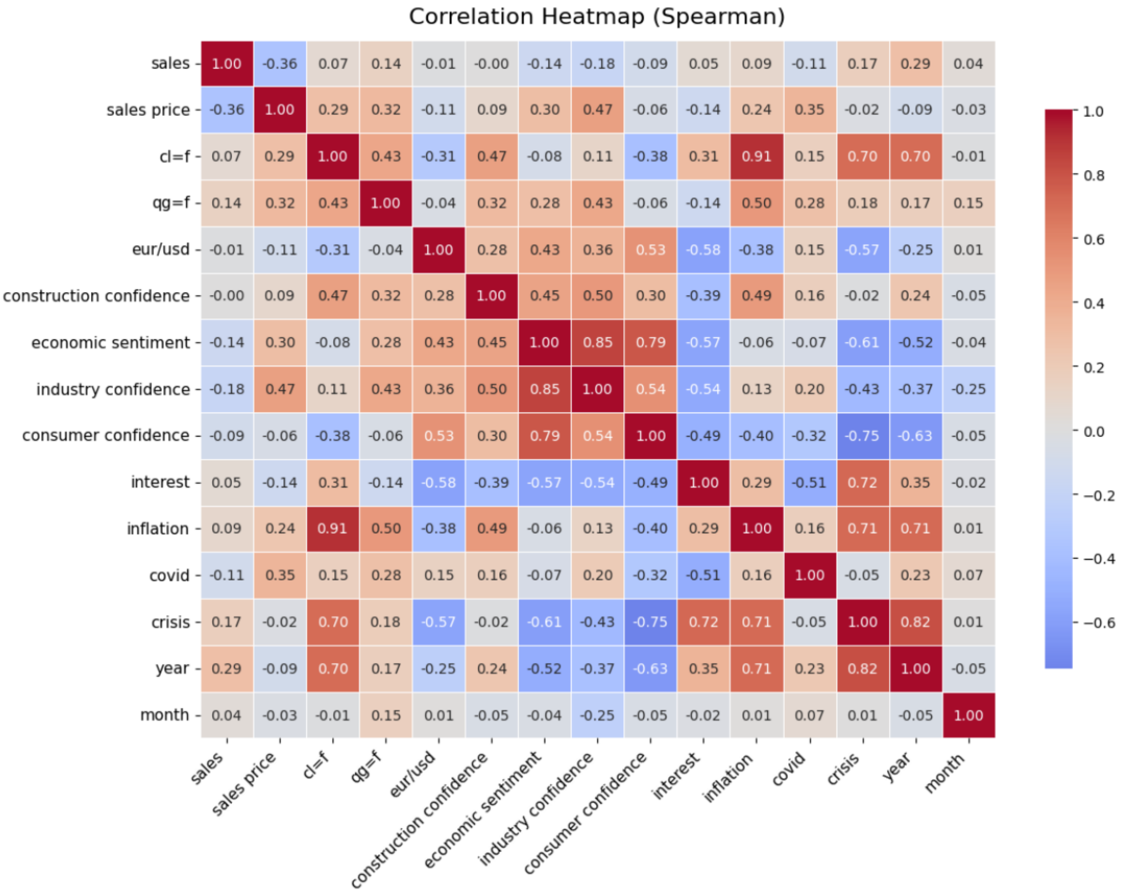


Figure 21 Spearman correlations between raw input variables

3.2.3 Data preparation

The aim of the data preparation phase was to prepare the raw dataset for modelling. In this study, data preparation consisted of data preprocessing and feature selection. Data preprocessing included the creation and transforming or features, ensuring that the dataset is correctly structured for forecasting. Feature selection was performed by combining correlation filtering and mutual information filtering to remove redundant features and to reduce multicollinearity and dimensionality in the dataset. These steps are described in more detail in the following sections.

3.2.3.1 Data preprocessing

The first step in data preprocessing was to create lagged features from the raw input variables. This was done to ensure feature availability at the time of forecasting, model the predictive power of leading variables and to ensure real-world performance. For this study, the input variables were shifted back in time only according to their availability, getting the latest available value at the time of forecasting. The target variable, monthly sales, was shifted three months forward. This made it possible to use monthly historical data to forecast the demand three months in advance. The final shifted features can be seen in table 3, which describes the structure of the final dataset used for modelling.

The ACF and PACF plots in figure 20 suggested that recent demand values affect current demand. This was considered by computing a 4-month EMA of the *sales t-1* feature. EMA is a moving average that places greater emphasis on recent data points. A 4-month EMA of the *sales price t-1* was also computed as a feature to possibly capture the temporal effects of pricing dynamics in the model. The time series decomposition visualized in figure 19 showed indication of yearly seasonality in demand, which was accounted for by creating one-hot-encoded variables of the month column to act as control features and removing the initial numeric month column. This was done to ensure the models could capture the cyclical seasonality of demand. However, this increased the dimensionality of the dataset significantly, which had to be considered in model selection.

The high variance, skewness and outliers in the demand data, indicated in table 2 and figure 18, increase the difficulty of accurate forecasting. This was assessed by using a Box-Cox transformation on the target variable *sales t+3* to improve normality, stabilize variance and to reduce skewness and the impact of outliers. The Box-Cox transformation is a family of power transformations that aims to make data more normally distributed and stabilize variance by finding the optimal λ, which is essential for many machine learning algorithms and statistical models (Osborne, 2010). The Box-Cox transformation of y is mathematically expressed as

$$y(\lambda) = \begin{cases} \dfrac{y^{\lambda} - 1}{\lambda}, & \text{if } \lambda \neq 0 \\ \ln(y), & \text{if } \lambda = 0 \end{cases}, \tag{13}$$

For the sales data in this research, the optimal λ proved to be 0,3546. The effectiveness of the Box-Cox transformation on the target variable is visually illustrated in figure 22, showing clear improvement in normality of *sales t+3*, decreasing the skewness, stabilizing variance and reducing the influence of outliers.

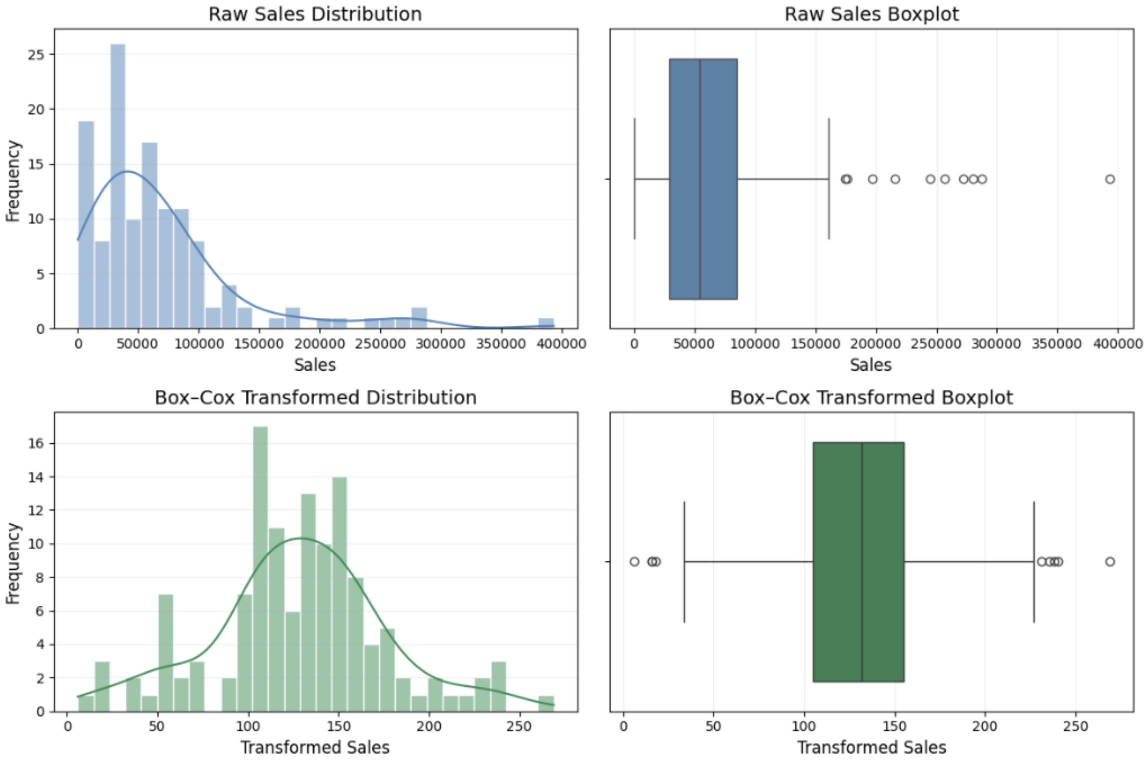
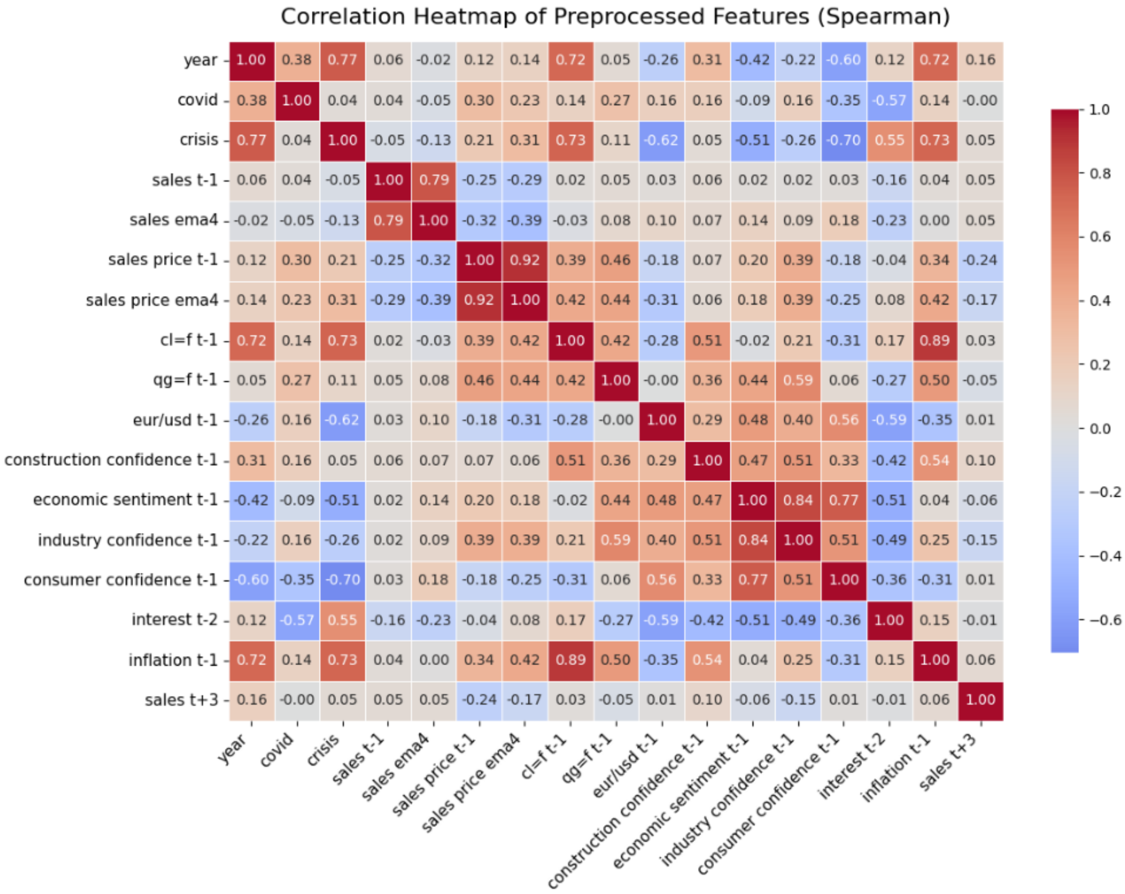


Figure 22 Effectiveness of Box-Cox transformation on the target variable

3.2.3.2 Feature selection

In this study, feature selection was performed by combining two filter methods: Correlation filtering and mutual information. Only training data was used in the feature selection process to prevent any data leakage. Additionally, the monthly one-hot-encoded variables were excluded from the feature selection process, as they were used in modelling as control variables due to their importance for capturing seasonality. Spearman correlation filtering was used to remove multicollinear features from the dataset with a correlation threshold of 0,75. The spearman correlation matrix used for filtering is visualized in figure 23.



Correlation Heatmap of Preprocessed Features (Spearman)

	year	covid	crisis	sales t-1	sales ema4	sales price t-1	sales price ema4	cl=f t-1	qg=f t-1	eur/usd t-1	construction confidence t-1	economic sentiment t-1	industry confidence t-1	consumer confidence t-1	interest t-2	inflation t-1	sales t+3
year	1.00	0.38	0.77	0.06	-0.02	0.12	0.14	0.72	0.05	-0.26	0.31	-0.42	-0.22	-0.60	0.12	0.72	0.16
covid	0.38	1.00	0.04	0.04	-0.05	0.30	0.23	0.14	0.27	0.16	0.16	-0.09	0.16	-0.35	-0.57	0.14	-0.00
crisis	0.77	0.04	1.00	-0.05	-0.13	0.21	0.31	0.73	0.11	-0.62	0.05	-0.51	-0.26	-0.70	0.55	0.73	0.05
sales t-1	0.06	0.04	-0.05	1.00	0.79	-0.25	-0.29	0.02	0.05	0.03	0.06	0.02	0.02	0.03	-0.16	0.04	0.05
sales ema4	-0.02	-0.05	-0.13	0.79	1.00	-0.32	-0.39	-0.03	0.08	0.10	0.07	0.14	0.09	0.18	-0.23	0.00	0.05
sales price t-1	0.12	0.30	0.21	-0.25	-0.32	1.00	0.92	0.39	0.46	-0.18	0.07	0.20	0.39	-0.18	-0.04	0.34	-0.24
sales price ema4	0.14	0.23	0.31	-0.29	-0.39	0.92	1.00	0.42	0.44	-0.31	0.06	0.18	0.39	-0.25	0.08	0.42	-0.17
cl=f t-1	0.72	0.14	0.73	0.02	-0.03	0.39	0.42	1.00	0.42	-0.28	0.51	-0.02	0.21	-0.31	0.17	0.89	0.03
qg=f t-1	0.05	0.27	0.11	0.05	0.08	0.46	0.44	0.42	1.00	-0.00	0.36	0.44	0.59	0.06	-0.27	0.50	-0.05
eur/usd t-1	-0.26	0.16	-0.62	0.03	0.10	-0.18	-0.31	-0.28	-0.00	1.00	0.29	0.48	0.40	0.56	-0.59	-0.35	0.01
construction confidence t-1	0.31	0.16	0.05	0.06	0.07	0.07	0.06	0.51	0.36	0.29	1.00	0.47	0.51	0.33	-0.42	0.54	0.10
economic sentiment t-1	-0.42	-0.09	-0.51	0.02	0.14	0.20	0.18	-0.02	0.44	0.48	0.47	1.00	0.84	0.77	-0.51	0.04	-0.06
industry confidence t-1	-0.22	0.16	-0.26	0.02	0.09	0.39	0.39	0.21	0.59	0.40	0.51	0.84	1.00	0.51	-0.49	0.25	-0.15
consumer confidence t-1	-0.60	-0.35	-0.70	0.03	0.18	-0.18	-0.25	-0.31	0.06	0.56	0.33	0.77	0.51	1.00	-0.36	-0.31	0.01
interest t-2	0.12	-0.57	0.55	-0.16	-0.23	-0.04	0.08	0.17	-0.27	-0.59	-0.42	-0.51	-0.49	-0.36	1.00	0.15	-0.01
inflation t-1	0.72	0.14	0.73	0.04	0.00	0.34	0.42	0.89	0.50	-0.35	0.54	0.04	0.25	-0.31	0.15	1.00	0.06
sales t+3	0.16	-0.00	0.05	0.05	0.05	-0.24	-0.17	0.03	-0.05	0.01	0.10	-0.06	-0.15	0.01	-0.01	0.06	1.00

Figure 23 Spearman correlation heatmap of preprocessed features

The correlation heatmap in figure 23 showed high multicollinearity between multiple feature pairs. Starting from the most collinear feature pair, if their correlation exceeded the 0,75 threshold, the feature with weaker correlation to the target variable was removed from the dataset. This process was iterated until all the pairwise correlations were below the threshold. As exception, the multicollinearity between *sales t-1* and *sales ema4* was resolved by dropping *sales ema4* due to the features having a similar correlation with the target variable, but the PACF plot in figure 20 indicated that the first sales lag had the most predictive power.

After multicollinearity was removed from the dataset, mutual information was used to identify and remove redundant features from the dataset. MI is a statistical measure of the amount of information that is shared between two random variables (Vergara & Estévez, 2014). It is useful in feature selection because it quantifies the relevance of a feature for predicting the target variable. It is also capable of measuring any kind of relationship between the variables (Estevez et al., 2009). Higher MI value indicates a greater amount of information a feature provides about the target, indicating a stronger relationship. The MI values between the features and the target value are visualized in figure 24.

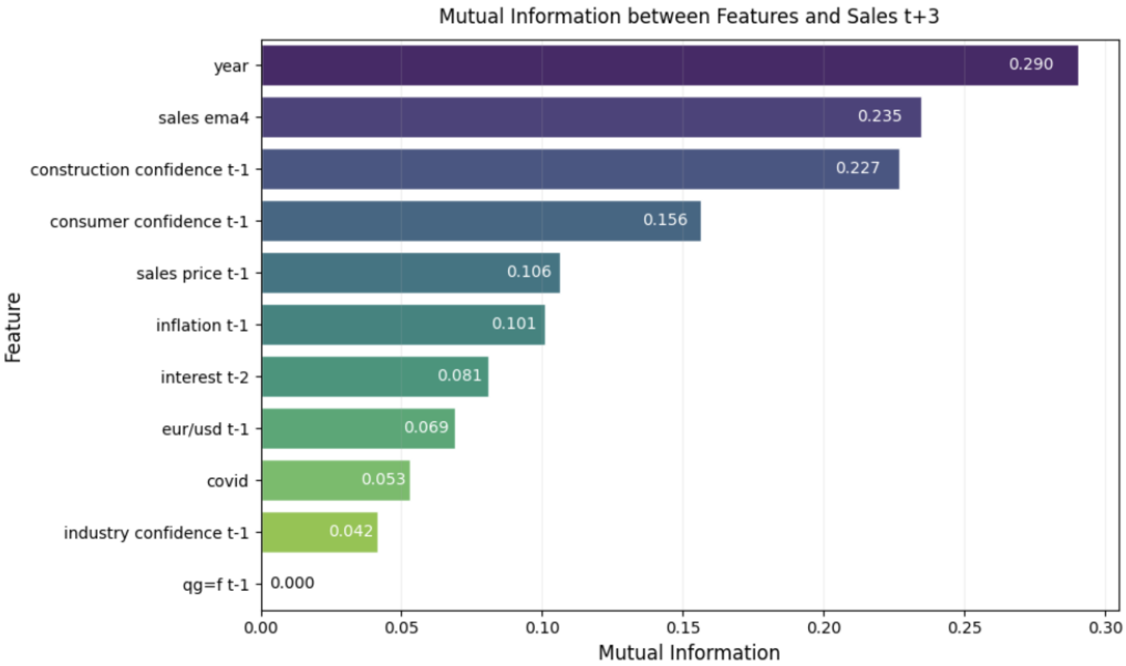


Figure 24 MI values for feature selection

The MI values indicated that feature *qg=f t-1* is less likely to have predictive power than the other features, thus it was dropped from the dataset. After the two-step feature selection process, the final dataset that was used for modelling contained 124 rows and

23 columns, of which 12 were the one-hot encoded month features. Additionally, due to the different feature value ranges, standardization was implemented to enhance the performance of the models and ensure the features contribute equally to the model training process. The final set of features used to model *sales t+3* is described in table 3 below.

Table 3 Final feature set (t = time of forecasting)

Feature	**Description**	**Type**
sales ema4	t-1 sales 4-month EMA	Continuous
sales price t-1	t-1 sales price	Continuous
eur/usd t-1	t-1 EUR/USD index	Continuous
construction confidence t-1	t-1 construction confidence indicator	Continuous
industry confidence t-1	t-1 industry confidence indicator	Continuous
consumer confidence t-1	t-1 consumer confidence indicator	Continuous
inflation t-1	t-1 HICP inflation rate	Continuous
interest t-2	t-2 3-month average interest rate	Continuous
year	Year at the time of forecasting	Continuous
covid	Covid indicator	Binary
month	Month at the time of forecasting	One-hot encoded

3.2.4 Modelling

The modelling phase consisted of selecting and developing the potential demand forecasting models as well as designing the setup for model training, hyperparameter tuning and testing. This section first outlines the model development workflow and describes the developed demand forecasting models and their hyperparameters, followed by assessing the model evaluation metrics and how key demand drivers are classified. The models selected for development and comparison represent univariate time-series forecasting models, machine learning models and hybrid models combining the strengths of both. Deep learning models were excluded from model comparison due to the insufficient amount of training data. The test set performance of the developed models is evaluated and compared both quantitatively and visually later in chapter 4.

As mentioned in chapter 3.2.2, the raw dataset consisted of monthly sales data from January 2015 to September 2025, with the final dataset containing 124 samples. Due to the small sample size and recent upward trend change in demand, the split choice between training, hyperparameter tuning and testing sets was crucial. In the end, the models were first trained with data from January 2015 to May 2024 and hyperparameters were tuned on data from June 2024 to December 2024. The models were then trained with the tuned parameters on data from January 2015 to December 2024. Final validation was performed on the test set from January 2025 to September 2025 and compared against a baseline forecast that was only available for that timeframe. The model development workflow is visually illustrated in figure 25.

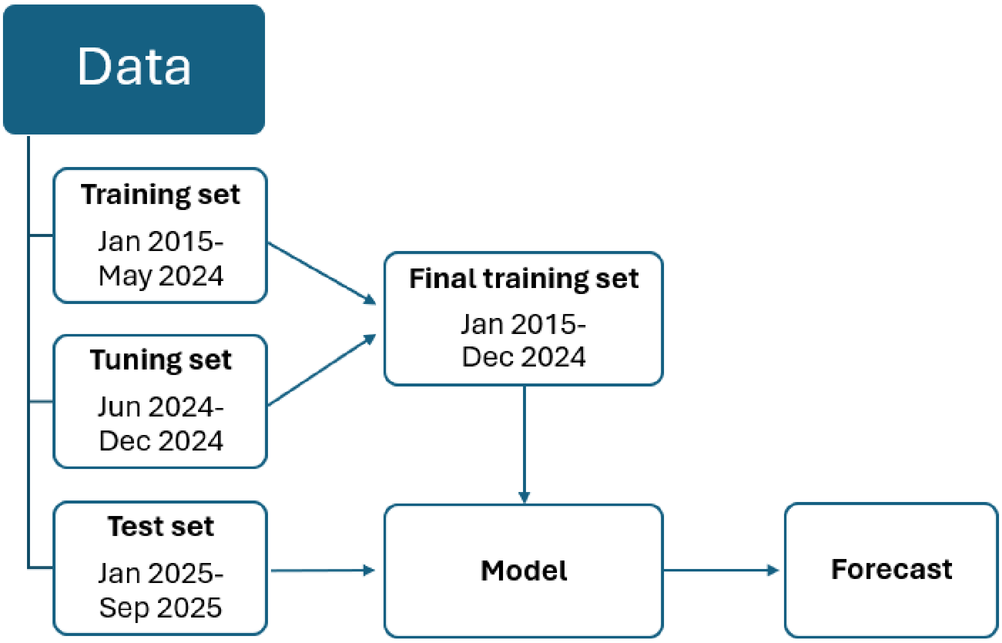


Figure 25 Model development workflow

3.2.4.1 Holt-Winters Exponential Smoothing

As described in chapter 2.3.1.1, Holt-Winters Exponential Smoothing is a time-series forecasting method that can work well with small datasets that have trendy and seasonal characteristics. Therefore, it was selected for modelling as a baseline univariate time-series model to compare against machine learning and hybrid models. The final model hyperparameters that were used for test set forecasting are seen below in table 4.

Table 4 Holt-Winters hyperparameters

Hyperparameter	**Value**
trend	’mul’
damped_trend	True
seasonal	’add’
seasonal_periods	12

3.2.4.2 Prophet

In this study, Prophet was selected for model comparison to represent the more modern univariate time-series forecasting methods. It was chosen to bridge the gap between classical, statistical time-series forecasting and machine learning as it combines concepts from both. The tuned hyperparameters of the Prophet model are listed in table 5.

Table 5 Prophet hyperparameters

Hyperparameter	**Value**
seasonality_mode	’multiplicative’
changepoint_prior_scale	0.5
n_changepoints	25
seasonality_prior_scale	5.0
changepoint_range	0.95

3.2.4.3 Elastic Net Regression

Elastic Net Regression was selected for comparison as a classical machine learning model due to it being a simple, efficient and interpretable forecasting solution for small, high-dimensional datasets that contain multicollinearity. The regularization and feature selection properties of Elastic Net Regression make it a suitable choice for this study to prevent overfitting. The hyperparameters of the tuned Elastic Net model are shown in table 6.

Table 6 Elastic Net hyperparameters

Hyperparameter	**Value**
alpha	0.01
l1_ratio	1.0

3.2.4.4 XGBoost

In this study, XGBoost was included in model comparison to represent state-of-the-art machine learning techniques. Similar to Elastic Net Regression, XGBoost offers built-in regularization to control model complexity and prevent overfitting. Due to the limited amount of training data, the XGBoost model had to be kept simple to decrease the risk of overfitting. The final tuned hyperparameters of the XBoost model are presented in table 7.

Table 7 XGBoost hyperparameters

Hyperparameter	**Value**
n_estimators	100
max_depth	2
learning_rate	0.05
reg_alpha	1
reg_lambda	1
subsample	0.6
gamma	0
min_child_weight	1
colsample_bytree	0.6

3.2.4.5 Hybrid models

As was stated in chapter 2.3.4, exploring hybrid models in demand forecasting can lead to efficient and robust solutions, especially in unstable market conditions with high volatility. In this study, the first hybrid model that was explored combined two of the simpler models that were also developed in isolation to see if combining them could increase the accuracy. The first hybrid model combined Holt-Winters Exponential Smoothing and Elastic Net Regression by first generating the Holt-Winters forecasts and then using the forecasts as a feature in Elastic Net Regression to generate the final forecasts. This approach to combining the models ensured the final forecast remained interpretable. The hyperparameters of the separate models were kept the same as in the isolated models for unbiased performance comparison. The diagram in figure 26 illustrates the framework of the Holt-Winters – Elastic Net hybrid model.

Figure 26 Holt-Winters - Elastic Net Regression hybrid model framework

The same hybridization approach as above was also applied to combine the two more advanced models in this study. In the second hybrid model, Prophet and XGBoost were combined by first generating Prophet forecasts and then adding them to the feature set that XGBoost was trained on to generate the final forecasts. As with the first hybrid, the hyperparameters of the models were retained from the isolated models to ensure fair model comparison. Figure 25 presents the framework of the Prophet – XGBoost hybrid model.

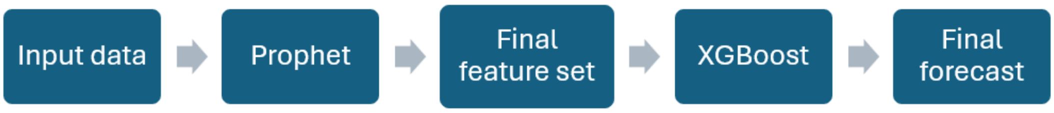


Figure 27 Prophet - XGBoost hybrid model framework

3.2.5 Model evaluation metrics and demand driver classification

The evaluation metrics used for assessing and comparing model performance on unseen data in this study were MAE, RMSE, MAPE and R^2 score, which are all widely recognized evaluation metrics in demand forecasting. MAE computes the average of the absolute difference between actual and predicted values (Tatachar, 2021). RMSE is the square root of the average of the mean of squares of errors (Tatachar, 2021). MAPE is the percentage equivalent of MAE, computing the mean of the absolute percentage errors between actual and predicted values (de Myttenaere et al., 2016). R^2 is the coefficient of determination, measuring the goodness-of-fit of the model by determining the proportion of variance in the target variable that is explained by the independent variables included in the model (Tatachar, 2021). To detect overfitting, training and test set MAE and RMSE were compared when evaluating each model. A significant increase in test set errors in relation to training set errors often indicate overfitting and low generalizability, while similar error metrics for training and testing sets suggest high model generalizability on unseen data, indicating better real-world performance.

In this study, the impact of features was analyzed, and key demand drivers were identified by combining SHAP values with MLR p-values and feature coefficients. SHAP values provide information about the contribution of each independent variable to specific model predictions (Wang et al., 2024). A feature with a higher SHAP value can be determined as more influential in the model when generating forecasts. SHAP is a model-agnostic, additive feature attribution technique, thus it was chosen for this study to model the feature relationships with the target variable (Wang et al., 2024). MLR p-values and coefficient signs were used to strengthen the SHAP results by providing evidence if the features were statistically significant predictors. This dual approach combining model-based interpretation and statistical inference was chosen to increase the robustness of the identification and analysis of demand drivers.

3.3 Research validity and reliability

The quality of quantitative research can be assessed by using reliability and validity as criteria (Golafshani, 2003). Validity can be defined as the extent to which a study measures what it is claimed to measure (Andersson et al., 2024). According to Andersson et al. (2024), validity can be classified into three types: Internal, external and construct validity. In this study, internal validity was assessed by splitting the data into training and testing sets to ensure the model performance was caused by predictive power of features, not data leakage. External validity was assessed by applying hold out validation to the trained models on the test set against a baseline forecast, providing indication of model generalization to unseen real-world data. Construct validity was assessed by measuring model forecast accuracies with widely accepted metrics such as MAE, RMSE, MAPE and R^2 score, aligning the results with existing demand forecasting research.

Research reliability can be defined as the extent to which the results are consistent and stable over time (Golafshani, 2003; Andersson et al., 2024). Andersson et al. (2024), state that reliable research should yield similar results under consistent conditions. This was ensured by using a standardized CRISP-DM framework to systematically approach each stage of the research from data understanding and preparation to modelling and evaluation. In addition, each step was documented to ensure that the research process was reliable and reproducible.

4 Results and discussion

This chapter presents, interprets and discusses the results of this thesis as well as provides the answers to the research questions of this study. This chapter is divided into two sections: model performance comparison, and identification and analysis of key demand drivers. In the model performance comparison section, the developed models are quantitatively compared through the chosen evaluation metrics and visually assessed by plotting the test set forecasts of all compared models against actual demand. By evaluating the models, the best performing approach is identified and will be implemented in the case company.

The identification and analysis of key demand drivers section provides answers to the second research question. The features were classified as key demand drivers if they ranked among the most impactful predictors based on SHAP values of the best performing model and if they were either statistically significant or approached significance according to their MLR p-values. The effects of the key demand drivers were interpreted in depth by analyzing the local SHAP values of the test set predictions to see how the features impacted the prediction. The SHAP analysis was supported with statistical evidence from MLR coefficients of significant features.

4.1 Model performance comparison

The computed test set evaluation metrics of the models included in this study are presented in table 8 below. Table 9 contains the comparison of training and test set MAE and RMSE values of the models to detect possible under- or overfitting. The test set performance of the models is visually presented in figures 28-33. The baseline forecast is included in both the numeric and visual comparisons for reference to show how the models performed against it.

Table 8 Test set performance comparison of forecasting models

Model	**MAE**	**RMSE**	**MAPE**	R^2
Baseline	*119822.22*	*141255.00*	*92.65 %*	*-1.48*
Holt-Winters	183273.05	204114.57	126.53 %	-4.18
Prophet	82108.31	104728.94	43.26 %	-0.36
Elastic Net Regression	54627.58	78383.27	**28.98 %**	0.24
XGBoost	65226.84	94390.54	38.41 %	-0.11
HW - Elastic Net	**51326.67**	**67917.65**	33.58 %	**0.43**
Prophet - XGBoost	71956.37	95151.18	45.12 %	-0.13

Table 9 Training and test set error rate comparison to detect overfitting

Model	**MAE (Train)**	**MAE (Test)**	**RMSE (Train)**	**RMSE (Test)**
Holt-Winters	58402.99	183273.05	71273.01	204114.57
Prophet	47362.20	82108.31	53047.25	104728.94
Elastic Net Regression	30899.44	54627.58	49088.08	78383.27
XGBoost	21311.32	65226.84	35784.30	94390.54
HW - Elastic Net	31034.25	51326.67	49314.04	67917.65
Prophet - XGBoost	19419.84	71956.37	31896.82	95151.18

The results obtained through this research suggest that a hybrid demand forecasting model combining Holt-Winters Exponential Smoothing and Elastic Net Regression performs best for the selected product, given the feature set selected to be included in this study. This was identified both quantitatively from the evaluation metrics presented in table 8, where the HW-Elastic Net hybrid model outperformed other models in MAE, RMSE and R^2 score, and qualitatively from the HW-Elastic Net forecast plot displayed in figure 28, which indicated that the HW-Elastic Net was the best fit for the hold-out test set. The HW-Elastic Net model did not show significant signs of overfitting, indicating good generalizability and real-world performance. The hybridization of Holt-Winters and Elastic Net Regression showed improved performance compared to independent models by combining Holt-Winters’ ability to capture the trend and seasonality and complementing it with Elastic Net Regression to utilize external features and built-in regularization. In addition, the HW-Elastic Net improved forecasting MAE by approximately 57 % compared to the baseline approach. This supports the findings made by Ahmed and Husien (2024) and Ingle et al. (2021), which state that hybrid models can perform more efficiently and accurately compared to isolated models, especially in volatile market conditions.

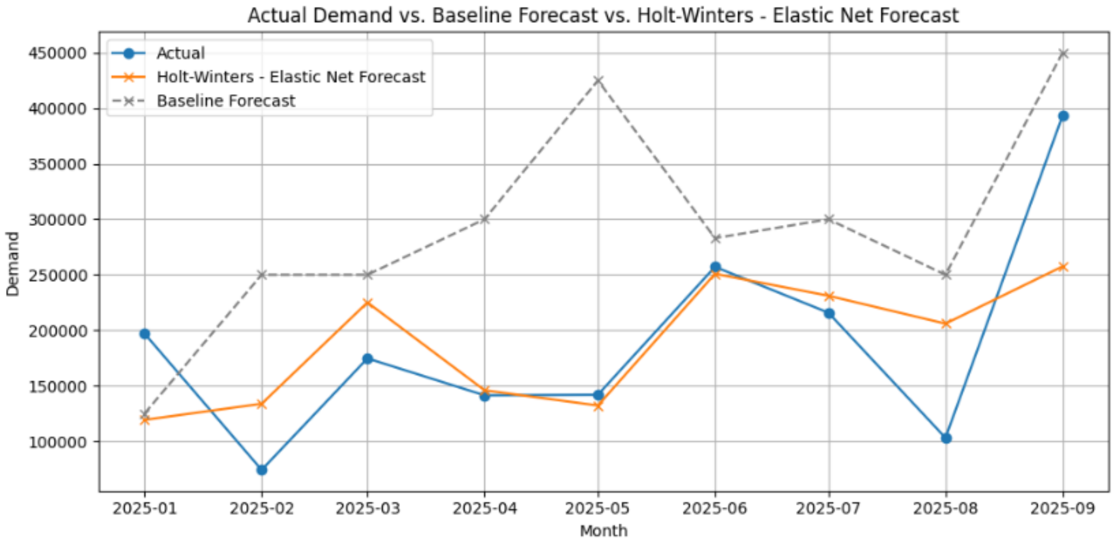


Figure 28 HW-Elastic Net hybrid forecast

Elastic Net Regression in isolation also performed well on the test set with good generalization, being only slightly outperformed by the HW-Elastic Net hybrid. Quantitatively it showed the lowest MAPE and was closest to HW-Elastic Net in the other metrics than the other models included in this study. Visually it performed similarly to the HW-Elastic Net hybrid, as can be seen in figure 29. The Elastic Net also showed significant forecasting accuracy improvements compared to baseline. HW-Elastic Net was slightly more accurate due to the slightly higher level of the predicted demand values, that was most likely driven by the Holt-Winters forecast as a feature. The tuned hyperparameters of the Elastic Net show that a pure Lasso regression with light regularization works best for this study's data by prioritizing feature selection and simplifying the model. This supports Chung's et al. (2023) statement that Elastic Net is efficient with high-dimensional data.

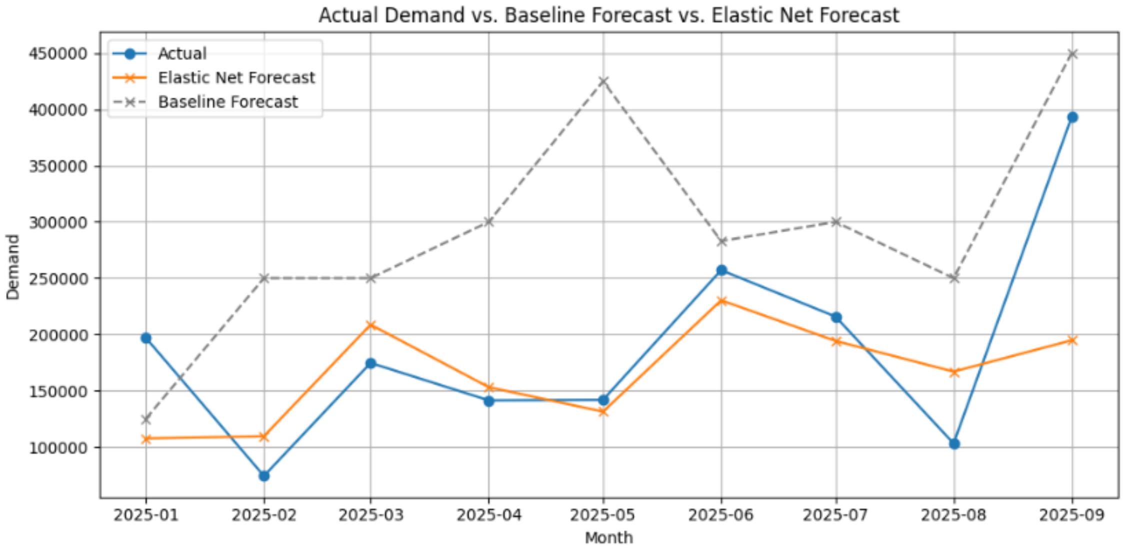


Figure 29 Elastic Net forecast

Holt-Winters Exponential Smoothing showed weakest quantitative performance amongst the compared models, having clearly inferior evaluation metrics and indicating significant overfitting. However, by inspecting figure 30, it appears that Holt-Winters is able to capture the underlying seasonal patterns in the demand, but the magnitude of the forecasts is significantly higher than actual demand. The seasonality captured by the Holt-Winters model proved to be useful in hybrid modelling, as it increased the performance of the isolated Elastic Net model. This finding is in line with Berberich's (2020) findings, suggesting that Holt-Winters can be hybridized with a machine learning model to compensate for its limitations.

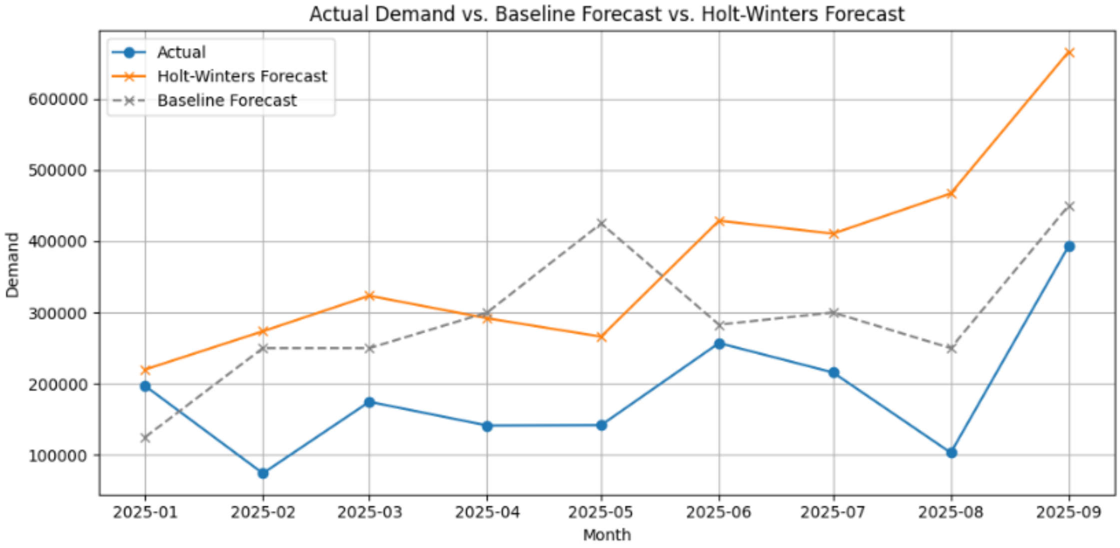


Figure 30 Holt-Winters forecast

The Prophet demand forecasting model showed weak performance numerically and moderate performance visually as can be observed from figure 31. It also indicated significant overfitting according to the difference in training and test set MAE and RMSE. Likely reasons for overfitting are the limited amount of data and the structural demand shift that happened in 2024, making the demand forecasting problem complex. This supports Guo's et al. (2021) statement that Prophet encounters limitations with more complex forecasting problems due to its limited expression ability and inability to utilize external features to generate forecasts and thus should be complemented with a machine learning model to build a hybrid model.

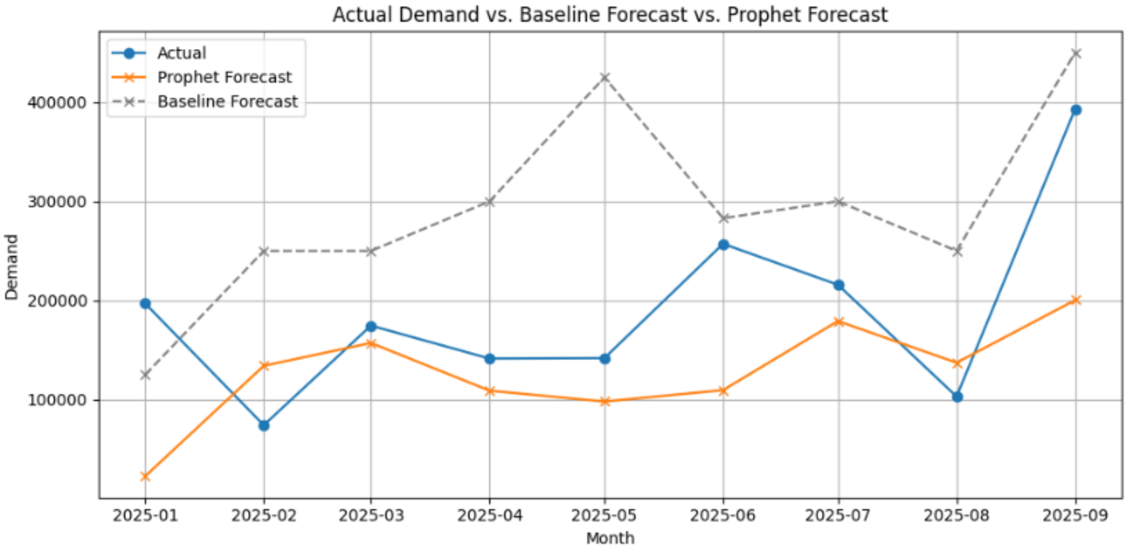


Figure 31 Prophet forecast

XGBoost, known for its state-of-the-art performance in demand forecasting with large datasets (Harikrishnan & Sreedharan, 2025), demonstrated clear signs of significant overfitting. It performed accurately on training data, but due to the lack of generalization, it was unable to perform well on the hold out test set. The overfitting can be seen visually in figure 32, where it can be observed that XGBoost is not able to predict the fluctuations in demand. In order to improve XGBoost's performance and prevent overfitting, more training data would be needed. The results obtained with the small dataset in this research support Velarde's et al. (2023) statement that the predictive power of XGBoost quickly decreases with insufficient training data.

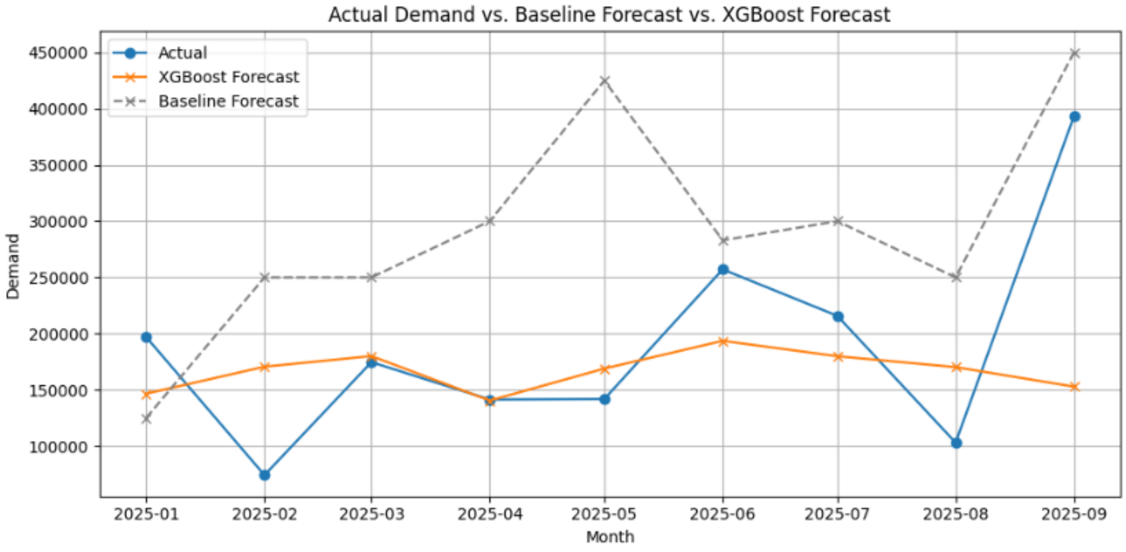


Figure 32 XGBoost forecast

The hybridization of Prophet and XGBoost did not show significant improvement to the performance of the models in isolation. The complex hybrid model also displayed more significant overfitting than the isolated models. The metrics show that it slightly improved the performance of the Prophet model but still performed worse than isolated XGBoost model. By visually inspecting figure 33, it seems that Prophet-XGBoost fits the data similarly to Prophet and slightly better than XGBoost. The weak performance of Prophet, XGBoost and the hybrid model combining them indicates that the models are too complex for the demand forecasting problem, mostly due to the lack of training data and insufficient model tuning.

After evaluating the models, their fit and test set performance, the results show that all of the developed models except for Holt-Winters Exponential Smoothing improved the forecasting metrics compared to the baseline approach. This indicates that there is a significant need for data-driven demand forecasting in the company, and the best performing HW-Elastic model should be implemented to support the current demand forecasting process.

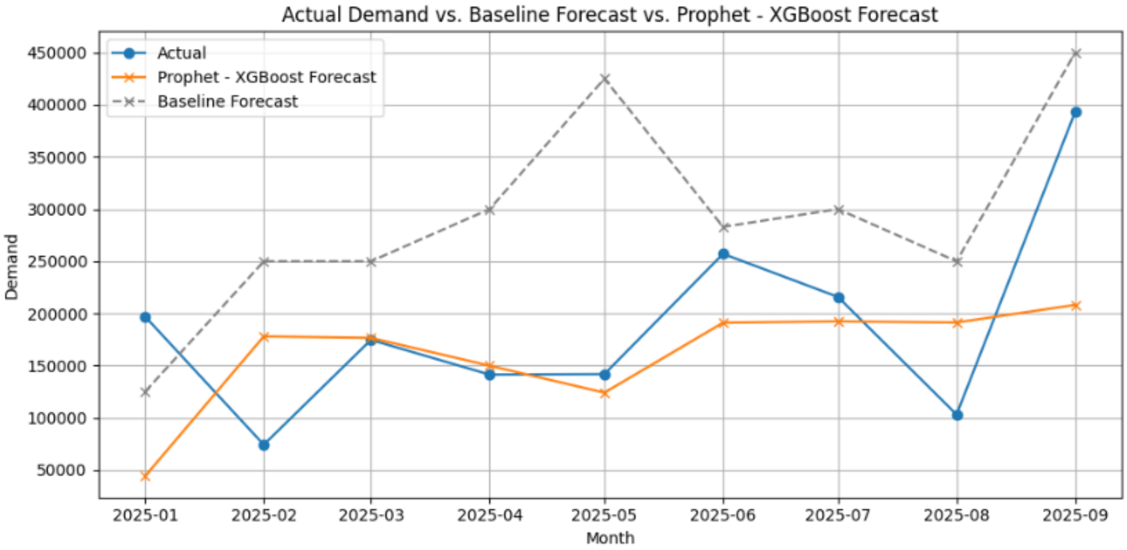


Figure 33 Prophet-XGBoost hybrid forecast

4.2 Identification and analysis of key demand drivers

This section identifies and analyzes the key demand drivers, using HW-Elastic Net SHAP values and MLR p-values as quantitative metrics. First, the key demand drivers are identified through analysis of SHAP values from the best performing HW-Elastic Net model and cross analyzing them with MLR p-values. Secondly, the impact of the key demand drivers in the HW-Elastic Net model is analyzed and interpreted with a similar dual approach to gain deeper understanding of the relationship between the features and demand.

4.2.1 Identification of key demand drivers

The most impactful features according to SHAP values are visualized in figure 34. The SHAP values indicate that the model emphasizes the upward trend of demand in its predictions, which can be seen as the high SHAP value of the *year* feature. Beyond trend, the model highlights the importance of macroeconomic indicators in predicting demand, giving weight to features such as industry confidence, historical interest rates, and the exchange rate between the Euro and USD. In addition, the SHAP values indicate that the past sales prices and short-term EMA of past sales volume affect future demand. The HW-Elastic Net model also uses the temporal components captured by the Holt-Winters forecast to complement the external features.

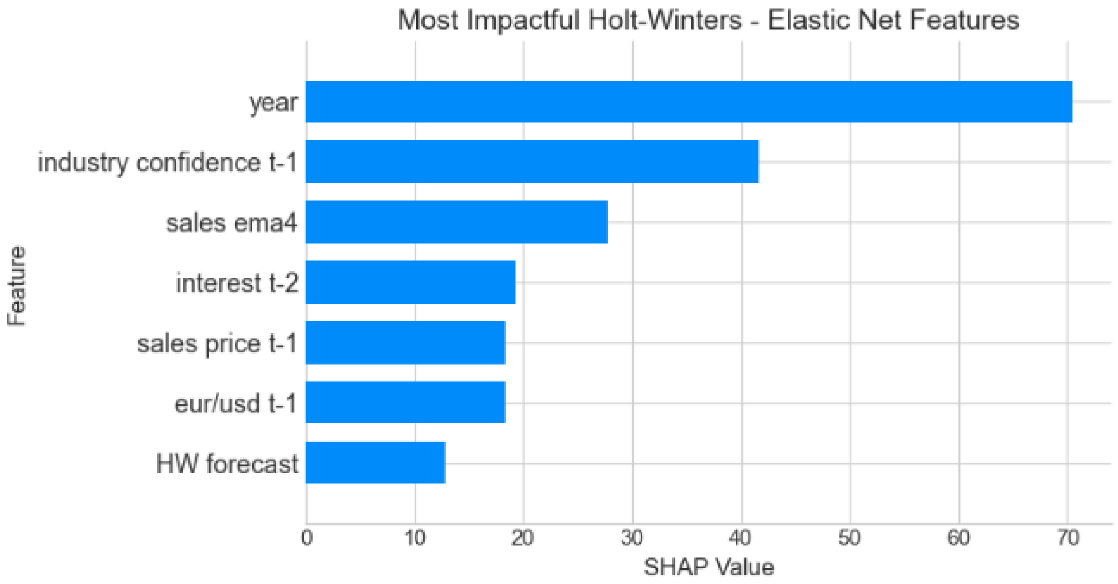


Figure 34 Most impactful HW-Elastic Net features according to SHAP value

To ensure robust identification of the key demand drivers, MLR was performed on the dataset to assess the significance of the features. It resulted in a statistically significant p-value for *sales price t-1*, *year*, *eur/usd t-1*, *industry confidence t-1* and *month_12*. In addition, the p-values of *month_7*, *interest t-2* and *month_6* were only slightly above the 5% significance level, thus they were also considered significant in this study. The MLR p-values of the significant features can be seen below in table 10.

Table 10 MLR p-values of the significant features

Feature	**p-value**
sales price t-1	0.000
year	0.002
eur/usd t-1	0.003
industry confidence t-1	0.01
month_12	0.025
month_7	0.063
interest t-2	0.063
month_6	0.07

By analyzing the HW-Elastic Net SHAP values from figure 34 and the MLR p-values from table 10, it can be concluded that they show similar results, both approaches highlighting the same key demand drivers. The identified key demand drivers and their justification are seen in table 11. Consistency across statistical significance and model-based feature importance suggests that product demand is driven by the combination of trend and seasonal patterns, short-term sales pricing and momentum as well as long-term economic indicators, explaining the efficiency of a hybrid model combining time-series forecasting and machine learning.

Table 11 Key demand drivers

Key demand driver	**SHAP justification**	**p-value justification**
Trend & Seasonality	*year, HW forecast*	*year, month_12, month_7, month_6*
Historical Sales Performance	*sales ema4, sales price t-1*	*sales price t-1*
Euro / USD Exchange Rate	*eur/usd t-1*	*eur/usd t-1*
Industry Confidence	*industry confidence t-1*	*industry confidence t-1*
Interest Rates	*interest t-2*	*interest t-2*

4.2.2 Analysis of the key demand drivers

This section contains the in-depth analysis of the key demand drivers identified in the previous section. The HW-Elastic Net model SHAP values and MLR coefficients of the significant features are analyzed to gain in-depth understanding of demand behavior and to explain model predictions. The features included in the analysis are the identified key demand drivers. The SHAP analysis was conducted on the test set to ensure the analysis reflects the model's generalization performance and interprets how the model uses each feature to make predictions on unseen data. The SHAP analysis plot is seen in figure 35. In the plot, blue dots represent relatively lower feature values while red dots represent higher feature values. A positive SHAP value on the x-axis indicates that the feature pushed the predictions higher while a negative value means that the feature decreased the predicted value.

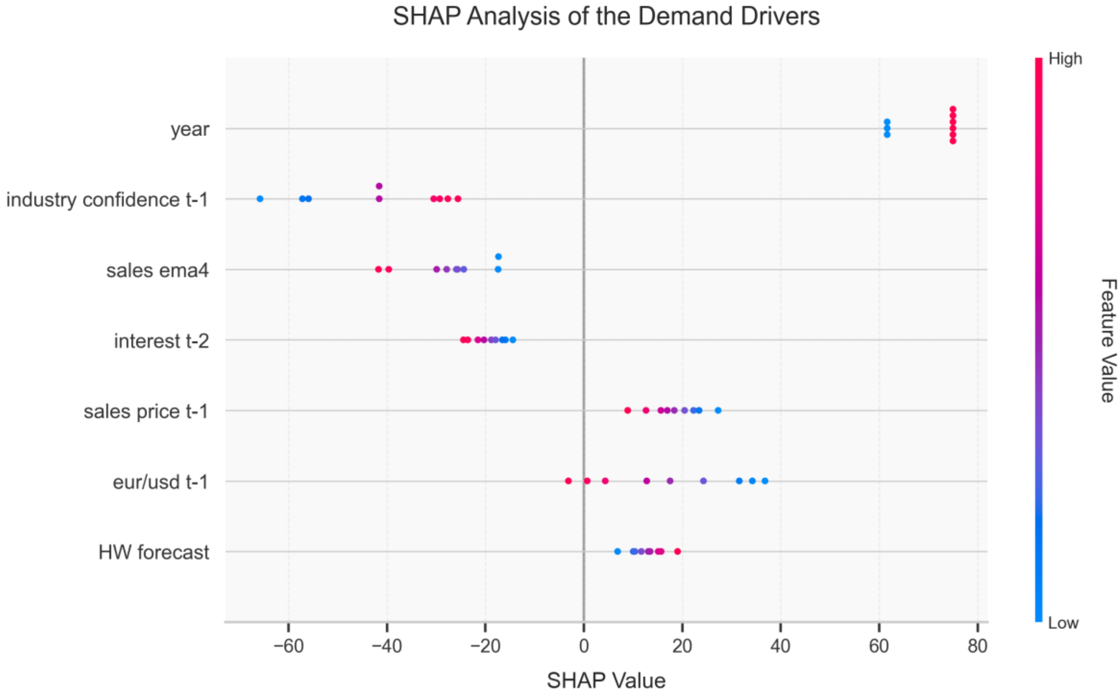


Figure 35 SHAP analysis of the identified demand drivers

As observed from the plot, the SHAP values of the test set predictions suggest that a higher year value pushes the predictions up, indicating the model has captured the growing trend in demand. The values also show that higher industry confidence levels push the demand forecasts higher. The model suggests that previous sales EMA and sales pricing are inversely proportional to three-month demand. Similarly, higher interest rates and the exchange rate between the Euro and USD lead to decreased demand forecasts.

The results of the model SHAP analysis are supported by the MLR variable coefficients that are visible below in table 12. Only the signs of the significant variables were included in the table. Again, the consistency across model-based results and statistical analysis strengthens the results of the demand driver analysis. These results increase the trustworthiness of the model and provide important insights for the company.

Table 12 MLR coefficient signs of the significant features

Feature	**Coefficient sign**
sales price t-1	-
year	+
eur/usd t-1	-
industry confidence t-1	+
month_12	+
month_7	+
interest t-2	-
month_6	+

5 Conclusion

This thesis acted as proof of concept for ML and advanced analytics in the case company. The research aimed to identify and develop a demand forecasting model for the company's chosen product to increase the forecasting accuracy against the baseline approach, as well as to identify and analyze key demand drivers that influence product demand by utilizing machine learning and statistical methods. To meet these objectives, an industry standard CRISP-DM framework was utilized to collect, process and analyze data and to develop and compare multiple demand forecasting models. Key demand drivers were identified by analyzing the SHAP values of the developed models and performing MLR to identify the most significant features for predicting demand. Research was guided by the following research questions:

1. What is the most efficient demand forecasting model for predicting the demand of the selected product?
2. What are the key demand drivers for the selected product?

The results of this study suggest that a hybrid model combining Holt-Winters Exponential Smoothing and Elastic Net Regression is the most efficient demand forecasting approach from the compared set of models on the current dataset, with a significant improvement of 57 % in MAE when compared against the baseline approach. The results of the key demand driver analysis in this study indicate that the demand of the selected product is driven by trend and seasonal factors, short-term sales pricing and momentum as well as macroeconomic leading indicators such as industry confidence, interest rates and the Euro/USD exchange rate.

The findings of this research contribute to machine learning and demand forecasting research by providing evidence that hybrid modelling can be an efficient demand forecasting solution with small and volatile real-world datasets. This was demonstrated by the HW-Elastic Net hybrid model that combined the temporal structure captured by Holt-Winters Exponential Smoothing with the multivariate inputs and regularization provided by the Elastic Net to outperform the isolated models. The results are in line with existing research, claiming that hybrid models can produce efficient results, especially in volatile market conditions by increasing accuracy and reducing overfitting issues. The results also suggest that model-based SHAP-values can be combined with MLR statistical inference to create a robust model interpretation framework.

The significant improvement in forecast MAE of approximately 57% in the test set suggests that the HW-Elastic Net model would enhance the inventory planning in the case company by reducing the risk of over- and understocking. More reliable three-month forecasts support purchasing decisions to help account for long lead-time uncertainty. In addition, the identified demand drivers provide actionable and monitorable indicators for demand planning, as well as increase the trust in the HW-Elastic Net model. The results enable the transition from judgmental forecasting to a robust, data driven process, ultimately advancing the overall analytical maturity of the case company.

The practical next step in the case company is to deploy the developed HW-Elastic Net demand forecasting model to production systems to support monthly supply planning processes. This enables the monitoring and validation of the model in a real-world operational environment to examine its business impact and long-term performance in varying scenarios. Data-driven demand forecasting will also be expanded to a wider range of products and product families.

5.1 Limitations and future research

While the hybrid model developed in this research improved demand forecasting accuracy significantly compared to baseline and provided relevant insights regarding variables that drive demand, there are limitations that need to be considered when evaluating the outcome of this research. The main limitation is the limited monthly demand data availability, leading to a small dataset. Small data meant that more complex machine learning models were prone for overfitting and deep learning models were excluded from the empirical study.

Another limitation was the structural change in demand in mid-2024 that led to a large upward trend and increased volatility in demand. To account for the structural change and to ensure the model could learn the changed structure, the size of the test set had to be decreased to only 9 samples that accounted for the timeframe from January to September in 2025. The small hold-out test set provided an initial indication on how the model performs on out-of-sample real world data, but more testing data is needed to robustly validate the model. In addition, the key demand drivers were identified by utilizing SHAP-values and MLR p-values, indicating correlation-based relationships and thus cannot be interpreted as evidence for true causal effects.

Future academic research should account for these limitations and extend the findings of this research. Firstly, a wider range of demand forecasting models should be compared to potentially identify more efficient forecasting solutions. This would require dataset expansion to also incorporate more complex ML models and deep learning models in the comparison. Secondly, taking the structural changes into account, future research should investigate forecasting methods that are built to be robust to regime shifts to potentially generate more accurate forecasts and better generalization.

Regarding the key demand drivers and their effects on model predictions, a more extensive set of explanatory variables could be investigated to potentially improve the performance of the model. Additionally, future research should assess how different feature selection and dimension reduction methods affect model performance and explainability. Future work should also emphasize performing extensive statistical analysis on the identified demand drivers to validate the causal relationships between variables and demand.

References

Aburto, L., & Weber, R. (2007). Improved supply chain management based on hybrid demand forecasts. *Applied Soft Computing*, *7*(1), 136-144. DOI: 10.1016/j.asoc.2005.06.001

Adekunle, B. I., Chukwuma-Eke, E. C., Balogun, E. D., & Ogunsola, K. O. (2021). Predictive Analytics for Demand Forecasting: Enhancing Business Resource Allocation Through Time Series Models. *ResearchGate, January*. DOI: 10.54660/.IJFMR.2021.2.1.32-42

Aguiar-Pérez, J. M., & Pérez-Juárez, M. Á. (2023). An insight of deep learning based demand forecasting in smart grids. *Sensors*, *23*(3), 1467. DOI: 10.3390/s23031467

Ahmed, M., & Husien, I. (2024). Heart disease prediction using hybrid machine learning: A brief review. *Journal of Robotics and Control (JRC)*, *5*(3), 884-892. DOI: 10.18196/jrc.v5i3.21606

Ahmetoglu, H., & Das, R. (2022). A comprehensive review on detection of cyber-attacks: Data sets, methods, challenges, and future research directions. *Internet of Things*, *20*, 100615. DOI: 10.1016/j.iot.2022.100615

Aliferis, C., & Simon, G. (2024). Overfitting, underfitting and general model overconfidence and under-performance pitfalls and best practices in machine learning and AI. *Artificial intelligence and machine learning in health care and medical sciences: Best practices and pitfalls*, 477-524. DOI: 10.1007/978-3-031-39355-6_10

Alzubi, J., Nayyar, A., & Kumar, A. (2018, November). Machine learning from theory to algorithms: an overview. In *Journal of physics: conference series* (Vol. 1142, p. 012012). IOP Publishing. DOI: 10.1088/1742-6596/1142/1/012012

Amosu, O. R., Kumar, P., Ogunsuji, Y. M., Oni, S., & Faworaja, O. (2024). AI-driven demand forecasting: Enhancing inventory management and customer satisfaction. *World Journal of Advanced Research and Reviews*, *23*(2), 100-110. DOI: 10.30574/wjarr.2024.23.2.2394

Andersson, M., Boateng, K., & Abos, P. (2024). Validity and Reliability: The extent to which your research findings are accurate and consistent. ResearchGate.

Anitha, E., Prasath, M. N., Sanjai, L., Shiny, J. A., & Varsini, P. (2023). Effective Food Demand Forecasting Using Machine Learning Algorithms. *2023 IEEE Engineering Informatics*, 1-7. DOI: 10.1109/IEEECONF58110.2023.10520567

Atrey, P., Brundage, M. P., Wu, M., & Dutta, S. (2025). Demystifying the Accuracy-Interpretability Trade-Off: A Case Study of Inferring Ratings from Reviews. *arXiv preprint arXiv:2503.07914*. DOI: 10.48550/arXiv.2503.07914

Awad, M., Khanna, R. (2015). Support vector regression. *Efficient learning machines: Theories, concepts, and applications for engineers and system designers*, 67-80. DOI: 10.1007/978-1-4302-5990-9_4

Babai, M. Z., Boylan, J. E., & Rostami-Tabar, B. (2022). Demand forecasting in supply chains: a review of aggregation and hierarchical approaches. *International Journal of Production Research*, *60*(1), 324-348. DOI: 10.1080/00207543.2021.2005268

Barbiero, P., Squillero, G., & Tonda, A. (2020). Modeling generalization in machine learning: A methodological and computational study. *arXiv preprint arXiv:2006.15680*. DOI: 10.48550/arXiv.2006.15680

Basias, N., & Pollalis, Y. (2018). Quantitative and qualitative research in business & technology: Justifying a suitable research methodology. Review of Integrative Business and Economics Research, 7, 91-105. Retrieved 10.9.2025 from https://www.proquest.com/scholarly-journals/Quantitativequalitative-research-business-amp/docview/1969776018/se-2?accountid=14797

Belcic, I., & Stryker, C. (2024). What is hyperparameter tuning?. IBM. Retrieved 28.5.2025 from https://www.ibm.com/think/topics/hyperparameter-tuning

Berberich, D. (2020). *Hybrid Methods for Time Series Forecasting* (Doctoral dissertation, Karlsruher Institut für Technologie (KIT)). DOI: 10.5445/IR/1000134512

Bergmann, D. (2023). *What is semi-supervised learning?.* IBM. Retrieved 23.5.2025 from https://www.ibm.com/think/topics/semi-supervised-learning

Berrar, D. (2019). Cross-validation. *Encyclopedia of bioinformatics and computational biology* (pp. 542–545). Academic Press. DOI: 10.1016/B978-0-12-809633-8.20349-X

Bhattacharya, A. (2022). *Applied Machine Learning Explainability Techniques: Make ML models explainable and trustworthy for practical applications using LIME, SHAP, and more*. Packt Publishing Ltd.

Bi, Q., Goodman, K. E., Kaminsky, J., & Lessler, J. (2019). What is machine learning? A primer for the epidemiologist. *American journal of epidemiology*, *188*(12), 2222-2239. https://doi.org/10.1093/aje/kwz189

Brentan, B. M., Luvizotto Jr, E., Herrera, M., Izquierdo, J., & Pérez-García, R. (2017). Hybrid regression model for near real-time urban water demand forecasting. *Journal of Computational and Applied Mathematics*, *309*, 532-541. DOI: 10.1016/j.cam.2016.02.009

Briscoe, E., & Feldman, J. (2011). Conceptual complexity and the bias/variance tradeoff. *Cognition*, *118*(1), 2-16. DOI: 10.1016/j.cognition.2010.10.004

Brown, S., Blackmon, K., Cousins, P., and Maylor, H. (2013). Operations management: policy, practice and performance improvement. Routledge.

Chandran, J. M., & Khan, M. R. B. (2024). A Strategic Demand Forecasting: Assessing Methodologies, Market Volatility, and Operational Efficiency. *Malaysian Journal of Business, Economics and Management*, 150-167. DOI: 10.56532/mjbem.v3i2.71

Chung, D., Lee, C. G., & Yang, S. (2023). A hybrid machine learning model for demand forecasting: Combination of k-means elastic-net and gaussian process regression. *International Journal of Intelligent Systems and Applications in Engineering*, *11*(6s), 325-336.

Cunningham, P., Cord, M., & Delany, S. J. (2008). Supervised learning. In *Machine learning techniques for multimedia: case studies on organization and retrieval* (pp. 21-49). Berlin, Heidelberg: Springer Berlin Heidelberg. DOI: 10.1007/978-3-540-75171-7_2

Cutler, A., Cutler, D. R., & Stevens, J. R. (2012). Random forests. *Ensemble machine learning: Methods and applications*, 157-175. DOI: 10.1007/978-1-4419-9326-7_5

Databricks. (n.d.). Machine Learning Models. Retrieved 16.5.2025 from https://www.databricks.com/glossary/machine-learning-models

De Myttenaere, A., Golden, B., Le Grand, B., & Rossi, F. (2016). Mean absolute percentage error for regression models. *Neurocomputing*, *192*, 38-48. https://doi.org/10.1016/j.neucom.2015.12.114

Dong, G., & Liu, H. (Eds.). (2018). *Feature engineering for machine learning and data analytics*. CRC press.

Estévez, P. A., Tesmer, M., Perez, C. A., & Zurada, J. M. (2009). Normalized mutual information feature selection. *IEEE Transactions on neural networks*, *20*(2), 189-201. DOI: 10.1109/TNN.2008.2005601

Ezugwu, A. E., Ikotun, A. M., Oyelade, O. O., Abualigah, L., Agushaka, J. O., Eke, C. I., & Akinyelu, A. A. (2022). A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects. *Engineering Applications of Artificial Intelligence*, *110*, 104743. DOI: 10.1016/j.engappai.2022.104743

Fawcett, S. E., Ellram, L. M., & Ogden, J. A. (2007). *Supply chain management: From vision to implementation*. Upper Saddle River, NJ: Pearson Prentice Hall.

Feizabadi, J. (2022). Machine learning demand forecasting and supply chain performance. *International Journal of Logistics Research and Applications*, *25*(2), 119-142. DOI: 10.1080/13675567.2020.1803246

Genc, B., & Tunc, H. Ü. S. E. Y. İ. N. (2019). Optimal training and test sets design for machine learning. *Turkish Journal of Electrical Engineering and Computer Sciences*, *27*(2), 1534-1545. DOI: 10.3906/elk-1807-212

Goar, V., & Yadav, N. S. (2024). Foundations of machine learning. In *Intelligent Optimization Techniques for Business Analytics* (pp. 25-48). IGI Global. DOI: 10.4018/979-8-3693-1598-9.ch002

Golafshani, N. (2003). Understanding reliability and validity in qualitative re-search. The qualitative report, 8(4), 597-607.

Gudivada, V., Apon, A., & Ding, J. (2017). Data quality considerations for big data and machine learning: Going beyond data cleaning and transformations. *International Journal on Advances in Software*, *10*(1), 1-20.

Guo, L., Fang, W., Zhao, Q., & Wang, X. (2021). The hybrid PROPHET-SVR approach for forecasting product time series demand with seasonality. *Computers & Industrial Engineering*, *161*, 107598. DOI: 10.1016/j.cie.2021.107598

Gupta, N., Mujumdar, S., Patel, H., Masuda, S., Panwar, N., Bandyopadhyay, S., ... & Munigala, V. (2021, August). Data quality for machine learning tasks. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining* (pp. 4040-4041). DOI: 10.1145/3447548.3470817

Hans, C. (2011). Elastic net regression modeling with the orthant normal prior. *Journal of the American Statistical Association*, *106*(496), 1383-1393. DOI: 10.1198/jasa.2011.tm09241

Harahap, A. Z. M. K., Rahim, M. K. I. A., Malinjasari, N., Salleh, S. M., & Ma'arof, R. A. (2025). Enhancing the Inventory Management through Demand Forecasting. *International Journal of Research and Innovation in Social Science*, *9*(1), 2737-2744. DOI: 10.47772/IJRISS.2025.9010221

Harikrishnan, G. R., & Sreedharan, S. (2025). Advanced short-term load forecasting for residential demand response: An XGBoost-ANN ensemble approach. *Electric Power Systems Research*, *242*, 111476. DOI: 10.1016/j.epsr.2025.111476

Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning.

Holdsworth, J. & Scapicchio, M. (2024). *What is deep learning?*. IBM. Retrieved 20.5.2025 from https://www.ibm.com/think/topics/deep-learning

Huang, Q. (2020, August). Model-based or model-free, a review of approaches in reinforcement learning. In *2020 International Conference on Computing and Data Science (CDS)* (pp. 219-221). IEEE. DOI: 10.1109/CDS49703.2020.00051

Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: principles and practice*. OTexts.

IBM. (2021). What is machine learning?. Retrieved 15.5.2025 from https://www.ibm.com/think/topics/machine-learning

IBM. (2021). What is overfitting?. Retrieved 30.5.2025 from https://www.ibm.com/think/topics/overfitting

IBM. (n.d.). What is a decision tree?. Retrieved 19.6.2025 from https://www.ibm.com/think/topics/decision-trees

IBM. (n.d.). What is an AI model?. Retrieved 16.5.2025 from https://www.ibm.com/think/topics/ai-model

IBM. (n.d.). What is random forest?. Retrieved 23.6.2025 from https://www.ibm.com/think/topics/random-forest

Ingle, C., Bakliwal, D., Jain, J., Singh, P., Kale, P., & Chhajed, V. (2021, July). Demand forecasting: Literature review on various methodologies. In *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)* (pp. 1-7). IEEE. DOI: 10.1109/ICCCNT51525.2021.9580139

Islam, M. T., Ayon, E. H., Ghosh, B. P., Chowdhury, S., Shahid, R., Rahman, S., ... & Nguyen, T. N. (2024). Revolutionizing retail: A hybrid machine learning approach for precision demand forecasting and strategic decision-making in global commerce. *Journal of Computer Science and Technology Studies*, *6*(1), 33-39. DOI: 10.32996/jcsts.2024.6.1.4

James, G., Witten, D., Hastie, T., Tibshirani, R., & Taylor, J. (2023). Linear regression. In *An introduction to statistical learning: With applications in python* (pp. 69-134). Cham: Springer international publishing. DOI: 10.1007/978-3-031-38747-0_3

Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine learning and deep learning. *Electronic markets*, *31*(3), 685-695. https://doi.org/10.1007/s12525-021-00475-2

Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, *349*(6245), 255-260. DOI: 10.1126/science.aaa8415

Joshi, A. V. (2020). Machine learning and artificial intelligence. DOI: 10.1007/978-3-031-12282-8

Kariluoto, A., Kultanen, J., Soininen, J., Pärnänen, A., & Abrahamsson, P. (2021, December). Quality of data in machine learning. In *2021 IEEE 21st international conference on software quality, reliability and security companion (QRS-C)* (pp. 216-221). IEEE. DOI: 10.1109/QRS-C55045.2021.00040

Karthika, K., Balasubramanie, P., Harishmitha, S., Shanmugapriya, P., & Ramya, T. E. (2025, January). Deep Learning Based Hybrid Transformer Model for Stock Price

Prediction. In *2025 International Conference on Multi-Agent Systems for Collaborative Intelligence (ICMSCI)* (pp. 1603-1608). IEEE. DOI: 10.1109/ICMSCI62561.2025.10894439

Khanal, S., Tirupathi, S., Zizzo, G., Rawat, A., & Pedersen, T. B. (2024). Domain Adaptation for Time series Transformers using One-step fine-tuning. *arXiv preprint arXiv:2401.06524*. DOI: 10.48550/arXiv.2401.06524

Kirchgässner, G., Wolters, J., & Hassler, U. (2012). *Introduction to modern time series analysis*. Springer Science & Business Media.

Kis, A. (2024). *Understanding Autocorrelation and Partial Autocorrelation Functions (ACF and PACF).* Medium. Retrieved 15.9.2025 from https://medium.com/@kis.andras.nandor/understanding-autocorrelation-and-partial-autocorrelation-functions-acf-and-pacf-2998e7e1bcb5

Kong, E. B., & Dietterich, T. G. (1995). Error-correcting output coding corrects bias and variance. In *Machine learning proceedings 1995* (pp. 313-321). https://doi.org/10.1016/B978-1-55860-377-6.50046-3

Krenker, A., Bešter, J., & Kos, A. (2011). Introduction to the artificial neural networks. *Artificial Neural Networks: Methodological Advances and Biomedical Applications. InTech*, 1-18. DOI: 10.5772/644

Lai, S. B. S., Shahri, N. H. N. B. M., Mohamad, M. B., Rahman, H. A. B. A., & Rambli, A. B. (2021). Comparing the performance of AdaBoost, XGBoost, and logistic regression for imbalanced data. *Mathematics and Statistics*, *9*(3), 379-385. DOI: 10.13189/ms.2021.090320

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436-444. DOI: 10.1038/nature14539

Lim, B., Arık, S. Ö., Loeff, N., & Pfister, T. (2021). Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, *37*(4), 1748-1764. DOI: 10.48550/arXiv.1912.09363

Lima, S., Gonçalves, A. M., & Costa, M. (2019, December). Time series forecasting using Holt-Winters exponential smoothing: An application to economic data. In *AIP conference proceedings* (Vol. 2186, No. 1). AIP Publishing. DOI: 10.1063/1.5137999

Liu, B. (2011). Supervised learning. In *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data* (pp. 63-132). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-19460-3_3

López Santos, M., García-Santiago, X., Echevarría Camarero, F., Blázquez Gil, G., & Carrasco Ortega, P. (2022). Application of temporal fusion transformer for day-ahead PV power forecasting. *Energies*, *15*(14), 5232. DOI: 10.3390/en15145232

Louppe, G. (2014). *Understanding random forests: From theory to practice* (Doctoral dissertation, Universite de Liege (Belgium)).

Luo, Y., Tseng, H. H., Cui, S., Wei, L., Ten Haken, R. K., & El Naqa, I. (2019). Balancing accuracy and interpretability of machine learning approaches for radiation treatment outcomes modeling. *BJR| Open*, *1*(1), 20190021. DOI: 10.1259/bjro.20190021

Mahesh, B. (2020). Machine learning algorithms-a review. *International Journal of Science and Research (IJSR).[Internet]*, *9*(1), 381-386. DOI: 10.21275/ART20203995

Mahin, M. P. R., Shahriar, M., Das, R. R., Roy, A., & Reza, A. W. (2025). Enhancing Sustainable Supply Chain Forecasting Using Machine Learning for Sales Prediction. *Procedia Computer Science*, *252*, 470-479. DOI: 10.1016/j.procs.2025.01.006

Mandala, V. (2024). Optimizing Retail Demand Forecasting: Big Data-Driven AI Models for Enhanced Customer Experience and Operational Efficiency. *Journal of Artificial Intelligence and Big Data Disciplines*, *1*(1), 16-26.

Manelli, L. (2020). Design of Algorithms. In *Introducing Algorithms in C: A Step by Step Guide to Algorithms in C* (pp. 13-68). Berkeley, CA: Apress. https://doi.org/10.1007/978-1-4842-5623-7_2

Martínez-Plumed, F., Contreras-Ochando, L., Ferri, C., Hernández-Orallo, J., Kull, M., Lachiche, N., ... & Flach, P. (2019). CRISP-DM twenty years later: From data mining processes to data science trajectories. *IEEE transactions on knowledge and data engineering*, *33*(8), 3048-3061. DOI: 10.1109/TKDE.2019.2962680

Mas' udin, I., & Kamara, M. S. (2017). Electronic data interchange and demand forecasting implications on supply chain management collaboration: a customer service perspective. *Jurnal Teknik Industri*, *18*(2), 138-148. DOI: 10.22219/JTIUMM.Vol18.No2.138-148

Mathew, A., Amudha, P., & Sivakumari, S. (2021). Deep learning techniques: an overview. *Advanced Machine Learning Technologies and Applications: Proceedings of AMLTA 2020*, 599-608. DOI: 10.1007/978-981-15-3383-9_54

Mediavilla, M. A., Dietrich, F., & Palm, D. (2022). Review and analysis of artificial intelligence methods for demand forecasting in supply chain management. *Procedia CIRP*, *107*, 1126-1131. DOI: 10.1016/j.procir.2022.05.119

Mienye, I. D., Swart, T. G., & Obaido, G. (2024). Recurrent neural networks: A comprehensive review of architectures, variants, and applications. *Information*, *15*(9), 517. DOI: 10.3390/info15090517

Mohammed, S., Budach, L., Feuerpfeil, M., Ihde, N., Nathansen, A., Noack, N., ... & Harmouch, H. (2025). The effects of data quality on machine learning performance on tabular data. *Information Systems*, *132*, 102549. DOI: 10.1016/j.is.2025.102549

Molnar, C. (2020). *Interpretable machine learning*. Lulu.com.

Moutinho & Huarng (2013). Conclusion in a book Quantitative Modelling In Marketing And Management, ed. by Luiz Moutinho, and Kun-Huang Huarng, World Scientific Publishing Company, 2013. ProQuest Ebook Central, http://ebookcentral.proquest.com/lib/tritoniaebooks/detail.action?docID=1080984.

Mucci, T. (2024). *Overfitting vs. underfitting: Finding the balance.* IBM. Retrieved 30.5.2025 from https://www.ibm.com/think/topics/overfitting-vs-underfitting

Murel, J. & Kavlakoglu, E. (2024). *What is dimensionality reduction?*. IBM. Retrieved 20.5.2025 from https://www.ibm.com/think/topics/dimensionality-reduction

Murel, J. & Kavlakoglu, E. (2024). *What is feature engineering?*. IBM. Retrieved 26.5.2025 from https://www.ibm.com/think/topics/feature-engineering

Murel, J. & Kavlakoglu, E. (2024). *What is reinforcement learning?*. IBM. Retrieved 23.5.2025 from https://www.ibm.com/think/topics/reinforcement-learning

Murphy, K. (2024). Reinforcement learning: an overview. *arXiv e-prints*, arXiv-2412. DOI: 10.48550/arXiv.2412.05265

Nasteski, V. (2017). An overview of the supervised machine learning methods. *Horizons. b*, *4*(51-62), 56. DOI: 10.20544/HORIZONS.B.04.1.17.P05

Neal, B., Mittal, S., Baratin, A., Tantia, V., Scicluna, M., Lacoste-Julien, S., & Mitliagkas, I. (2018). A modern take on the bias-variance tradeoff in neural networks. *arXiv preprint arXiv:1810.08591*. DOI: 10.48550/arXiv.1810.08591

Noble, J. (2024). *What are ARIMA models?.* IBM. Retrieved 10.6.2025 from https://www.ibm.com/think/topics/arima-model

Osborne, J. (2010). Improving your data transformations: Applying the Box-Cox transformation. *Practical Assessment, Research, and Evaluation*, *15*(1). DOI: https://doi.org/10.7275/qbpc-gk17

Park, Y. S., Konge, L., & Artino Jr, A. R. (2020). The positivism paradigm of research. *Academic medicine*, *95*(5), 690-694. DOI: 10.1097/ACM.0000000000003093

Ra, A., & Souza, R. (2019). A Review on Machine Learning Algorithms. *International Journal for Research in Applied Science & Engineering Technology*. DOI: 10.22214/ijraset.2019.6138

Rahman, A. (2019). Statistics-based data preprocessing methods and machine learning algorithms for big data analysis. *International Journal of Artificial Intelligence*, *17*(2), 44-65.

Raschka, S. (2018). Model evaluation, model selection, and algorithm selection in machine learning. *arXiv preprint arXiv:1811.12808*. DOI: 10.48550/arXiv.1811.12808

Salehinejad, H., Sankar, S., Barfett, J., Colak, E., & Valaee, S. (2017). Recent advances in recurrent neural networks. *arXiv preprint arXiv:1801.01078*. DOI: 10.48550/arXiv.1801.01078

Salih, A. M., Raisi-Estabragh, Z., Galazzo, I. B., Radeva, P., Petersen, S. E., Lekadir, K., & Menegaz, G. (2025). A perspective on explainable artificial intelligence methods: SHAP and LIME. *Advanced Intelligent Systems*, *7*(1), 2400304. DOI: 10.1002/aisy.202400304

Samuel, A. L. (1959). Machine learning. *The Technology Review*, *62*(1), 42-45.

Saunders, M., Lewis, P., & Thornhill, A. (2023). *Research methods for business students*. Pearson education.

Schell, C. (1992). The Value of the Case Study as a Research Strategy. Manchester Business School.

Shimaoka, A. M., Ferreira, R. C., & Goldman, A. (2024). The evolution of CRISP-DM for data science: Methods, processes and frameworks. *SBC Reviews on Computer Science*, *4*(1), 28-43. https://doi.org/10.5753/reviews.2024.3757

Shinde, P. P., & Shah, S. (2018, August). A review of machine learning and deep learning applications. In *2018 Fourth international conference on computing communication control and automation (ICCUBEA)* (pp. 1-6). IEEE. DOI: 10.1109/ICCUBEA.2018.8697857

Sistla, S. M. K., Krishnamoorthy, G., Jeyaraman, J., & Konidena, B. K. (2024). Machine learning for demand forecasting in manufacturing. *Int J Multidiscip Res (IJFMR)*, *6*, 1-11. DOI: 10.36948/ijfmr.2024.v06i01.14204

Stryker, C. (2024) *What is a recurrent neural network (RNN)?*. IBM. Retrieved 24.6.2025 from https://www.ibm.com/think/topics/recurrent-neural-networks

Tatachar, A. V. (2021). Comparative assessment of regression models based on model evaluation metrics. *International Research Journal of Engineering and Technology (IRJET)*, *8*(09), 2395-0056.

Taylor, S. J., & Letham, B. (2018). Forecasting at scale. *The American Statistician*, *72*(1), 37-45. DOI: 10.7287/peerj.preprints.3190v2

Thejovathi, M., ChandraSekharaRao, M. V. P., Priyadharsini, E. J., Siddi, S., Karthik, B., & Abbas, S. H. (2024). Optimizing Product Demand Forecasting with Hybrid Machine Learning and Time Series Models: A Comparative Analysis of XGBoost and SARIMA. *EJ and Siddi, Someshwar and Karthik, B. and Abbas, Syed Hauider, Optimizing Product Demand Forecasting with Hybrid Machine Learning and Time Series Models: A Comparative Analysis of XGBoost and SARIMA*

Uçar, M. K., Nour, M., Sindi, H., & Polat, K. (2020). The effect of training and testing process on machine learning in biomedical datasets. *Mathematical Problems in Engineering*, *2020*(1), 2836236. DOI: 10.1155/2020/2836236

Van Engelen, J. E., & Hoos, H. H. (2020). A survey on semi-supervised learning. *Machine learning*, *109*(2), 373-440. DOI: 10.1007/s10994-019-05855-6

Varoquaux, G., & Colliot, O. (2023). Evaluating machine learning models and their diagnostic value. *Machine learning for brain disorders*, 601-630. DOI: 10.1007/978-1-0716-3195-9_20

Velarde, G., Sudhir, A., Deshmane, S., Deshmunkh, A., Sharma, K., & Joshi, V. (2023). Evaluating XGBoost for balanced and imbalanced data: application to fraud detection. https://doi.org/10.48550/arXiv.2303.15218

Vergara, J. R., & Estévez, P. A. (2014). A review of feature selection methods based on mutual information. *Neural computing and applications*, *24*(1), 175-186. DOI: 10.1007/s00521-013-1368-0

Waller, M. A., & Fawcett, S. E. (2013). Data science, predictive analytics, and big data: a revolution that will transform supply chain design and management. *Journal of Business logistics*, *34*(2), 77-84. DOI: 10.1111/jbl.12010

Wang, H., Liang, Q., Hancock, J. T., & Khoshgoftaar, T. M. (2024). Feature selection strategies: a comparative analysis of SHAP-value and importance-based methods. *Journal of Big Data*, *11*(1), 44. https://doi.org/10.1186/s40537-024-00905-w

Wen, X., & Li, W. (2023). Time series prediction based on LSTM-attention-LSTM model. *IEEE access*, *11*, 48322-48331. DOI: 10.1109/ACCESS.2023.3276628

Yin, H., Aryani, A., Petrie, S., Nambissan, A., Astudillo, A., & Cao, S. (2024). A rapid review of clustering algorithms. *arXiv preprint arXiv:2401.07389*. DOI: 10.48550/arXiv.2401.07389

Yin, R. K. (2009). *Case study research: Design and methods* (Vol. 5). sage.

Zhang, L., & Jánošík, D. (2024). Enhanced short-term load forecasting with hybrid machine learning models: CatBoost and XGBoost approaches. *Expert Systems with Applications*, *241*, 122686. DOI: 10.1016/j.eswa.2023.122686

Zhang, X., Li, P., Han, X., Yang, Y., & Cui, Y. (2024). Enhancing Time Series Product Demand Forecasting with Hybrid Attention-Based Deep Learning Models. *IEEE Access*. DOI: 10.1109/ACCESS.2024.3516697

Zheng, A., & Casari, A. (2018). *Feature engineering for machine learning: principles and techniques for data scientists*. " O'Reilly Media, Inc.".