Development of Predictive Analytics for Demand Forecasting and Inventory Management in Supply Chain using Machine Learning Techniques

Seyedehmahya Seyedan

A Thesis

In the Department

Of

Concordia Institute for Information Systems Engineering (CIISE)

Presented in Partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy
(Information & Systems Engineering)
at
Concordia University
Montreal, Quebec, Canada

December 2023

CONCORDIA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

This is to certify that the thesis prepared

By: **Seyedehmahya Seyedan**

Entitled: **Development of Predictive Analytics for Demand Forecasting and Inventory Management in Supply Chain using Machine Learning Techniques**

and submitted in partial fulfillment of the requirements for the degree of

Doctor Of Philosophy (Information & Systems Engineering)

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

_____ Chair
Dr. Andrea Schiffauerova

_____ External Examiner
Dr. Uday Venkatadri

_____ Arm's Length Examiner
Dr. Govind Gopakumar

_____ Examiner
Dr. Nizar Bouguila

_____ Examiner
Dr. Yong Zeng

_____ Thesis Supervisor
Dr. Fereshteh Mafakheri

_____ Thesis Supervisor
Dr. Chun Wang

Approved by _____
Dr. Jun Yan, Graduate Program Director

02/20/2024 _____
Dr. Mourad Debbabi, Dean
Gina Cody School of Engineering & Computer Science

ABSTRACT

Development of Predictive Analytics for Demand Forecasting and Inventory Management in Supply Chain using Machine Learning Techniques

Mahya Seyedan, Ph.D.

Concordia University, 2023

Forecasting demand effectively and managing inventories efficiently are critical components of modern supply chain management. By understanding full scope of demand possibilities, businesses gain ability to fine-tune inventory levels, navigate situations involving stockouts and overstock, and move toward a more resilient and precise supply chain. This thesis focuses on strategies to enhance these critical functions.

We start with examining impact of customer segmentation on forecasting precision by introducing a novel cluster-based demand forecasting framework that harnesses ensemble learning techniques. Our results showcase the effectiveness of the clustered-ensembled approach with minimal forecast errors. However, the constraints related to data availability and segmentation indicate areas that warrant further investigation in future research.

The significance of demand accuracy becomes most apparent when we consider its impact on safety stock. In second objective, we explore multivariate time series forecasting for optimal safety stock and inventory management, utilizing deep learning models and a cost optimization framework. This strategy outperforms individual models, demonstrating enhanced forecasting accuracy and stability across diverse product domains. Calculating safety stock based on proposed demand prediction framework leads to optimized safety stock levels. This not only prevents costly stockouts but also minimizes surplus inventory, resulting in reduced overall holding costs and improved inventory efficiency.

Although the first two objectives provided optimized results, relying on point predictions to calculate safety stock is not ideal. Unlike traditional point forecasting, distribution forecasting aims to cover the entire range of potential demand outcomes, essentially creating a comprehensive map of possibilities. The third objective of this thesis introduces recurrent mixture density networks (RMDNs) for refined distribution demand forecasting and safety stock estimation. These

innovative models consistently outperform traditional LSTM models, offering more precise stockout and overstock predictions. This approach not only reduces inventory costs but also enhances supply chain efficiency.

In summary, this thesis provides valuable insights and methodologies for businesses aiming to enhance demand forecasting accuracy and optimize inventory management practices in the retail industry. By leveraging customer segmentation, ensemble deep learning, and distribution forecasting techniques, organizations can enhance decision-making processes, reduce operational costs, and thrive in the dynamic landscape of supply chain operations.

ACKNOWLEDGEMENTS

First and foremost, I want to extend my deepest gratitude to Dr. Fereshteh Mafakheri and Dr. Chun Wang for their invaluable guidance during my Ph.D. journey. Their mentorship and unwavering support have been pivotal in shaping my growth and development as a Ph.D. candidate.

I also want to express my sincere appreciation to my committee members, Dr. Gopakumar, Dr. Bouguila, and Dr. Zeng. Their valuable insights and external perspectives provided crucial guidance during both my research proposal and comprehensive exam. Their advice and input have been instrumental in shaping the direction of my research.

To members both past and present of the Decision Modeling & Analytics Laboratory (DeciMAL) group, I extend my heartfelt gratitude. Your contributions to our scientific and nonscientific discussions have been invaluable. I would further like to extend a special thanks to those who have not only been colleagues but also become lifelong friends along this journey.

I also wish to express my deep appreciation to the Concordia Institute for Information Systems Engineering (CIISE) at Concordia University. Working within this department has been an enriching experience, and I attribute this to the welcoming and friendly work environment fostered by the institution.

Last but certainly not least, my profound thanks go to my husband, Milad, and my parents. Without Milad's unwavering support and my parents' constant encouragement, completing this journey would not have been possible. Along this path, we have shared moments of joyous laughter and heartfelt tears, and I am immensely grateful for having them all by my side through it all.

To my beloved parents, brothers, and husband

Farhang, Mahvash, Farid, Navid and Milad

Contribution of Authors

This thesis has been prepared in "Manuscript-based" format under the co-direction of Dr. Fereshteh Mafakheri and Dr. Chun Wang. All the articles presented in this thesis were co-authored and reviewed prior to submission for publication by Dr. Fereshteh Mafakheri and Dr. Chun Wang.

The author of this thesis acted as the principal researcher and performed the mathematical model's development, programming of the solution algorithms, analysis, and validation of the results, along with writing of the articles. All authors reviewed the final manuscript and approved the contents.

- The literature review article entitled "Predictive big data analytics for supply chain demand forecasting: methods, applications, and research opportunities", co-authored by Dr. Fereshteh Mafakheri was published in Journal of Big Data in Decembre 2020.
- The article entitled "Cluster-based demand forecasting using Bayesian model averaging: An ensemble learning approach", co-authored by Dr. Fereshteh Mafakheri and Dr. Chun Wang was published in Decision Analytics Journal in June 2022 (Chapter 3).
- The article entitled "Order-Up-To-Level Inventory Optimization Model using Time-Series Demand Forecasting with Ensemble Deep Learning", co-authored by Dr. Fereshteh Mafakheri and Dr. Chun Wang was published in Supply Chain Analytics in June 2023 (Chapter 4).
- The article entitled "Safety Stock Estimation Based on Forecasted Demand Distribution using Recurrent Mixture Density Networks", co-authored by Dr. Fereshteh Mafakheri and Dr. Chun Wang was submitted (Chapter 5).

TABLE OF CONTENTS

List of Figures

List of Tables

List of Symbols and Acronyms

Abbreviations	Definition
ANFIS	Adaptive neuro-fuzzy inference system
ANNs	Artificial neural networks
ARCH	Autoregressive conditional heteroscedastic
ARIMA	Auto-regressive integrated moving average
ANN	Artificial neural network
BMA	Bayesian model averaging
CNN	convolutional neural network
CSL	cycle service level
DT	Decision tree
d	Forecasted demand
DNNs	deep neural networks
EDA	Exploratory data analysis
EDMDRNN	Ensemble encoder-decoder mixture-density recurrent neural network
EOQ	economic order quantity
ERM	empirical risk minimization
GARCH	General autoregressive conditional heteroscedastic
GHSOM	Growing hierarchical self-organizing map
GMMs	Gaussian mixture models
GRUs	Gated recurrent units
KDE	kernel density estimators
KNN	K-nearest neighbor
KO	kernel-weights optimization
IID	independent and identically distributed
L	Lead time
LOO	Leave-One-Out cross-validation
LSTM	Long short-term memory
LR	Logistic regression
MA	moving average
MAE	Mean absolute error
MAPE	Mean absolute percentage error
MDNs	Mixture density networks
MDRNN	Mixture density recurrent neural network
MLP	multi-layer perceptron
MSE	mean squared error
μ	Mean of forecasted demand
NN	Neural networks
OUTL	order-up-to level
q	Order quantity
ReLU	Rectified linear unit

r	Reorder point
RFM	Recency, frequency and monetary
RMDNs	Recurrent mixture density networks
S	Replenishment level
RMSE	Root mean square error
RNNs	Recurrent neural networks
ss	Safety stock
σ	Standard deviation of forecasted demand
SOM	Self-organizing map
SAA	sample average approximation
SES	simple exponential smoothing
SKUs	stock-keeping units
SVM	Support vector machine
SVR	Support vector regression
SBA	Syntetos and Boylan approximation
WAIC	Widely applicable information criterion

Chapter 1. Introduction

1.1. Background and Motivation

In the ever-evolving retail industry, managing demand and supply planning processes has become increasingly complex. Retailers must address many operational aspects, ranging from purchase planning and distribution channels to labor availability and after-sales service. The most critical practice in retail management is demand forecasting, which lies at the center of this complexity (Pacheco et al., 2017). This thesis aims to address these challenges and understand the intricate world of demand forecasting and inventory management.

Demand forecasting is essential for retail operations because it is a key component of various critical activities, including optimizing production plans, managing services, orchestrating transportation logistics, achieving cost-effective inventory control, and ensuring efficient management of safety stock (Babai et al., 2011). Moreover, accurate demand forecasts lay the foundation for retailers to develop adaptive pricing strategies that optimize revenue management. The purpose of this research is to delve into the multifaceted world of demand forecasting in retail, addressing the challenges faced by retailers and leveraging data-driven methodologies to improve forecasting accuracy. The outcomes of this work, as presented in Chapters 3 to 5, highlight the findings of the study. Finally, in conclusion, we summarize important insights and offer useful directions for future studies in this area.

1.2. Problem Statement

As a result of a dynamic market, the retail industry faces a variety of challenges regarding demand forecasting, including handling demand fluctuations, mitigating various operational uncertainties, and optimizing the complex processes surrounding demand forecasting (Ge et al., 2019). Although numerous methods are available for demand forecasting, retailers must choose the approach most suitable to their unique data and business context (Seyedan & Mafakheri, 2020).

Moreover, with the rise of e-commerce and online shopping, retailers have gained access to a wealth of customer information, including demographics, historical spending patterns, and social media activities. However, effectively leveraging these data to improve demand forecasting

remains a challenge. The choice between traditional time-series forecasting methods and modern machine learning and deep learning approaches needs careful consideration, as each has strengths and limitations (Güven & Şimşir, 2020; van Steenbergen & Mes, 2020). Furthermore, the need to enhance forecast accuracy, robustness, and stability through ensemble learning methods and aggregation tools presents its own complexities.

1.3. Objectives and Purposes

The objectives and purposes of this thesis encompass a comprehensive exploration of various aspects of demand forecasting in the retail industry. First and foremost, the study aims to delve into the impacts of customer information, encompassing demographics and behavioral data, and how these can be effectively utilized to enhance the precision and customization of demand predictions. Additionally, the research seeks to investigate a variety of forecasting techniques, spanning from classical time-series methods to cutting-edge machine learning and deep learning approaches, with a specific focus on their capability in addressing the distinctive challenges inherent to retail demand forecasting. Furthermore, the thesis underscores the significance of ensemble learning methods and aggregation tools in augmenting the accuracy, resilience, and stability of forecasts. Ultimately, the primary goal is to contribute valuable insights, methodologies, and actionable recommendations that empower retailers to adeptly navigate the complexities of demand forecasting, optimize their operational strategies, and prosper in a competitive and perpetually evolving retail environment. In essence, this thesis is a comprehensive exploration of retail demand forecasting to equip retailers with the knowledge and tools to address multifaceted challenges and capitalize on the vast opportunities in retail management. To achieve the above objectives, we summarize our three main methodologies as follows:

1. Develop a cluster-based demand forecasting methodology using Bayesian model averaging.

To meet our objectives, we proposed a data-driven framework for ensemble cluster-based demand forecasting. The proposed approach was implemented in a real-world case with demand data of sports and electronic products. The aim was to forecast demand for the next cycle. We then conducted a sensitivity analysis to gain insights on the performance of the proposed demand forecasting approach. The scenarios included those with or without customer segmentation and with or without ensemble learning in the clustering and forecasting sections. This analysis showed

the proposed ensemble learning approach provides much more accurate predictions than conventional models. Chapter 3 provides further details.

2. Develop an order-up-to-level (OUTL) inventory optimization model using time-series demand forecasting with ensemble deep learning.

We delved through this process into the development of a comprehensive data-driven inventory optimization process crafted for online retailers. Our approach involved predicting daily demand and strategically optimizing stock levels. With heterogeneous deep neural networks (DNNs), we were able to capture a wider range of features, which enhanced forecasting accuracy. We proposed an ensemble deep learning model to capture global, temporal, and local patterns within supply chain data by combining the strengths of multilayer perceptrons (MLP), long short-term memory (LSTM), and 1D-vonvolutional neural networks (1D-CNN). To better manage inventory, we implemented an OUTL policy that minimizes opportunity costs while avoiding excessive inventories. We tested our research evaluation using real-world supply chain time series data from electronic and sports supplies. To showcase the effectiveness of our ensemble deep learning model, we conducted comparisons with individual base models (MLP, LSTM, and 1D-CNN), demonstrating superior performance across various accuracy metrics. Chapter 4 provides further details.

3. Develop a methodology for safety stock estimation based on forecasted demand distribution using recurrent mixture density networks.

In the context of supply chain management, this objective offers significant contributions that address key challenges. First, we used the mixture density network (MDN) component not only to make demand predictions but also to quantify uncertainty around these predictions. This is important because it helps supply chain professionals make informed decisions that account for demand fluctuations, improving the resilience of supply chains. Second, we introduced a novel RMDN model that combines LSTM and MDNs for probabilistic demand forecasting. This innovation allows for accurate demand predictions and the quantification of prediction uncertainty. These capabilities are crucial for optimizing inventory, logistics, and risk management in supply chains, ultimately enhancing their efficiency and competitiveness. Together, these contributions represent a significant advancement in supply chain management, providing new and essential tools for addressing its complexities. Chapter 5 provides further details.

1.4. Thesis Structure

This thesis is broken down into six chapters. Chapter 2 provides a thorough exploration of the literature, primarily focusing on examining modeling techniques and methodologies that are directly relevant to the research being conducted. In Chapter 3, we introduce an approach to cluster-based demand forecasting in the sports retail sector leveraging ensemble learning techniques, with the primary objective of enhancing the precision of daily demand predictions. In Chapter 4, we look into multivariate time series forecasting to optimize inventory management and safety stock. Using deep learning models such as MLP, LSTM, and 1D-CNN, we craft an ensemble forecasting strategy that outperforms individual models, elevating forecast accuracy, stability, and adaptability across both sports and electronics product domains. Chapter 5, we describe our exploration of RMDNs to further refine demand forecasting and safety stock estimation by predicting demand distribution rather than point forecasting. These innovative models consistently demonstrate their superiority to traditional LSTM models. Finally, Chapter 6 concludes the manuscript with final remarks and outlines several potential directions for future research.

Chapter 2. Literature Review

Inventory management is a critical aspect of supply chain operations. Demand forecasting and safety stock estimation are pivotal components of effective inventory management. Historically, forecasting and inventory control were not coordinated; the former has mainly focused on point forecasts, and the latter incorporates forecasted demands as exogenous inputs. However, research has evolved to recognize the interdependence of these two components. Prak et al. (2017) highlighted the need to account for forecast error autocorrelation in safety stock calculations, leading to the integration of forecasting and inventory models (Prak & Teunter, 2019). Recently, Goltsos et al. (2022) delved into the intricate relationship between demand forecasting and inventory control in the context of supply chain operations, introducing a framework to categorize integration levels and providing valuable insights into the challenges and opportunities of merging these two critical domains.

Foundational articles in inventory control dating back to the early 20th century include the work of Harris (1913), who introduced the economic order quantity (EOQ) model. These fundamental EOQ formulations typically assume that both demand and its true parameters remain known and constant, with exceptions noted (Andriolo et al., 2014; Glock et al., 2014). Approaches to handle demand uncertainty commonly fall into three main groups: The first category assumes demand distribution, and its parameters are known at the decision-making point in each time period (Axsäter, 2015; Goltsos et al., 2022). However, in practical scenarios, this is often not the case because decision makers lack prior knowledge about demand distribution and its dynamism (Arrow et al., 1958). Customer segmentation offers a potential solution to this limitation. By categorizing customers based on their distinct preferences, behaviors, and purchasing patterns, organizations can gain valuable insights into demand variations that might be obscured when treating all customers as a homogeneous group. The effect of customer segmentation on demand forecasting lies in its ability to unveil hidden patterns and trends within the data, allowing for more accurate predictions.

The second method category involves two phases: an estimation/forecasting phase and an optimization phase (Babai et al., 2020; Prak et al., 2017; Prak & Teunter, 2019). In this approach, researchers initially make distributional assumptions about historical data and employ statistical

methods to estimate parameters. These estimated values then inform the optimization of decision variables such as order quantity and safety stock. However, this approach may result in suboptimal solutions or difficulty in using forecasting methods appropriately (Liu et al., 2022).

The third method category focuses on distributional or parametric assumptions associated with historical demand data, with a particular emphasis on leveraging machine learning and deep learning–based techniques (Wang et al., 2022). These methods can be either parametric or nonparametric, with the latter referred to as nonparametric forecasting methods (Goltsos et al., 2022). In this context, recent advances in data analytics and artificial intelligence have revolutionized these areas, offering new avenues for achieving accurate forecasts and accordingly optimizing safety stock levels. Traditional time series models such as ARIMA and exponential smoothing have been widely used (Babai et al., 2013; Hyndman & Athanasopoulos, 2021; Thomopoulos, 2015), but their limitations in capturing complex and changing patterns have prompted the adoption of machine-learning methods (Güven & Şimşir, 2020; van Steenbergen & Mes, 2020). In recent years, deep learning methods such as recurrent neural networks (RNNs) and their variants have shown promise in demand forecasting (Rathipriya et al., 2023; Seyedan et al., 2023).

To address these challenges, each section of this literature review offers unique insights into approaches and methodologies of demand forecasting and inventory management. In Section 2.1, we explore the effect of customer segmentation on demand forecasting and how it can improve prediction accuracy. In Section 2.2, we investigate opportunities for improving point forecasting with deep learning models. Finally, in Section 2.3, we delve into distribution forecasting, emphasizing the interdependence of demand forecasting and inventory control and the potential of RMDNs in enhancing accuracy. These three sections collectively provide a comprehensive overview of the field, setting the stage for the proposed research framework to enhance inventory management practices.

2.1. Cluster-Based Forecasting

Demand forecasting is a highly needed and challenging issue in the retail industry. Demand forecasts are used to support inventory control (Barrow & Kourentzes, 2016), supply chain management (Kone & Karwan, 2011), and replenishment (Sillanpää & Liesiö, 2018) decisions. Lack of demand planning could lead to shortages and stock reduction or over storage resulting in

high storage costs (Turrado García et al., 2012), delays, the bullwhip effect (Carbonneau et al., 2008), need to reorder, and missed customers. Customer segmentation in demand forecasting is an approach that identifies customers with similar demand behavior and used as a means of improving prediction accuracy (Murray et al., 2018). According to McDonald (McDonald et al., 2003), market segmentation is "the process of splitting customers, or potential customers, within a market into different groups, or segments, within which customers have the same or similar requirements satisfied by a distinct marketing mix."

Online retailers can use customer segment profiles to predict the expected demand from each segment for their products and thus be able to adjust product processes accordingly (Q. Lu & Liu, 2013; Qu et al., 2017). Customer segmentation is often done based on individuals' purchasing power (Murray, Agard, & Barajas, 2015). It can improve the accuracy of demand predictions because the data in each segment belongs to a separate cluster of (similar) customers, making it easier for prediction models to extract the patterns in data (Espinoza et al., 2005; Kashwan & Velu, 2013; McCarty & Hastak, 2007; Wei et al., 2012). For example, forecasting for a cluster of customers who make seasonal purchases differs from customer segments that follow monotonically increasing and decreasing purchase patterns (Murray, Agard, Paul, et al., 2015). Several methods exist for customer segmentation (Kashwan & Velu, 2013). A basic segmentation can be done based on customers' geographic location. However, some demand characteristics of customers in the same location could still be different (Murray, Agard, & Barajas, 2015). Partitioning, hierarchical, density-based, grid-based, and model-based methods are some of more advanced segmentation methods (Collica, 2017). Many studies in the marketing area have focused on data-mining techniques for segmenting customers. The data mining is conducted through analysis of recency, frequency and monetary (RFM) value, partitioned clustering, logistic regression (LR), DT, and neural networks (NN) (Coussement et al., 2014; McCarty & Hastak, 2007; A. X. Yang, 2004). Partitioned clustering has low computational costs and is applicable when the data sets are large. K-means is one of the common partitioning methods (Murray, Agard, & Barajas, 2015).

After segmentation of customers, various algorithms can be used to forecast demand for each cluster of customers. These algorithms are generally categorized into statistical and artificial intelligence methods (Khashei & Bijari, 2011). ARIMA is a statistical method used for time-series forecasting. The method requires complete data sets (without missing information) and aims at

identifying linear relationships considering fixed temporal dependence and univariate data. It can only provide one-step forecasts (Jason, 2018). The autoregressive conditional heteroscedastic (ARCH) and general autoregressive conditional heteroscedastic (GARCH) models are two other statistical methods that can be used for time-series data with nonlinear patterns (Khashei & Bijari, 2011). Prophet is another time-series forecasting method that is attracting growing attention. It is an algorithm to forecast time-series data based on an additive model (adding a seasonal trend to forecast) where nonlinear trends are fitted with daily, weekly, yearly, and holiday seasonality effects. In addition, Prophet can handle data sets with missing data and outliers (Taylor & Letham, 2018).

Artificial intelligence techniques such as machine learning and deep learning are also effective in solving time-series forecasting problems (Jason, 2018). Artificial neural networks (ANN), SVM, KNN, and adaptive neuro-fuzzy inference systems (ANFIS) are examples of methods for predicting time-series data (Abbasimehr et al., 2020). Among these methods, ANNs have several advantages, including universal approximation, being data-driven, and better capturing of nonlinear patterns in data (Khashei & Bijari, 2011). Although different types of ANNs can capture nonlinear patterns in time-series data, researchers have indicated that ANNs with shallow architectures cannot accurately model time series with a high degree of nonlinearity, longer range, and heterogeneous characteristics (Sagheer & Kotb, 2019). A specific class of ANNs is RNNs which is capable of learning temporal representation of data (Parmezan et al., 2019). Unlike feed forward ANNs, the connections between nodes in an RNN establish a (feedback) cycle, allowing signals to move both forward and backward, thus providing a structure that supports a short-term memory suitable for processing sequential data. However, RNNs have a vanishing and exploding gradient problem for longer term predictions, making them sometimes hard to train. The prevalent solution to this problem is the addition of gated architectures such as LSTM that can exploit a long-range timing information (Abbasimehr et al., 2020; Bai et al., 2021). Table 2.1 presents the recent studies in demand forecasting domain along with the techniques adopted.

Table 2.1 Literature review on ensemble learning approach.

Authors	Clustering method	Ensemble learning in clustering	Forecasting method	Ensemble learning in forecasting
(Lemke & Gabrys, 2010)	-	-	ARIMA, Moving average, Single exponential smoothing, NN	Simple average, Trimming average Variance-based model, Outperformance method, Variance-based pooling, Regression combination
(Li et al., 2011)	-	-	Adaptive linear element network, Backpropagation network, Radial basis function network	BMA
(C.-J. Lu & Kao, 2016)	Single linkage, Complete linkage, Centroid linkage, Median linkage, Ward's linkage	Majority voting	Extreme learning machine	-
(Nilashi et al., 2017)	SOM, Expectation Maximization	Hypergraph-Partitioning Algorithm	ANFIS, SVR	-
(Raza et al., 2017)	-	-	Backpropagation neural network, Elman neural network, ARIMA, Feed forward neural network, Radial basis function, Wavelet transform	BMA
(Das Adhikari et al., 2018)	-	-	Time- Series Model, Regression Model	Weights Generation
(Papageorgiou et al., 2019)	-	-	Fuzzy cognitive maps, ANN	Bootstrapping method
(Bandara et al., 2020)	K-Means, PAM, DBSCAN, Snob	Boosting approach	LSTM	-
(Abbasimehr & Shabani, 2021)	Agglomerative hierarchical clustering	-	ARIMA, Simple moving average, KNN	Combined method (as described in their paper)
(Gastinger et al., 2021)	-	-	ARIMA, NN, Naive	Stacking ensemble-based
(Massaoudi et al., 2021)	-	-	Light Gradient Boosting Machine, eXtreme Gradient Boosting machine, Multi-Layer Perceptron	Stacking ensemble-based
(Zhang et al., 2021)	-	-	Multi-layer perceptron neural network, Convolutional neural network, LSTM	Stacking ensemble-based

2.2. Point Forecasting

In this context, literature review studies related to the integration of forecasting and inventory policies are provided, with a particular focus on the most relevant and recent papers that employ machine learning forecasting methods. In time-series forecasting, a linear function of the past observations is established following autoregressive (AR), moving average (MA) or auto-regressive integrated moving average (ARIMA) methods. Recent advances in data analytics and artificial intelligence have provided the practitioners with use of machine learning and deep learning approaches in time-series forecasting (Wang et al., 2021). Machine learning methods could evaluate the demand with or without assuming distribution or parameters. İfraz et al., (2023) investigated demand forecasting of spare parts in bus fleets and compared various forecasting methods, including regression-based, rule-based, tree-based, and artificial neural networks. Their results indicated that the artificial neural network outperformed all others, providing the highest accuracy rate and the least deviation in demand forecasting. Swaminathan & Venkitasubramony, (2023) reviewed forecasting techniques in predicting demand for fashion products focusing on advancements in artificial intelligence and machine learning methods considering various combinations of them.

The challenge of incorporating demand uncertainty in Inventory models without simplifying assumptions about distribution was investigated by (Trapero et al., 2019a). The widely accepted assumption that demand follows a normal distribution has been questioned. They suggested the use of a nonparametric kernel density approach for short lead times to fill this gap. In doing so, kernel density estimators (KDE), generalized autoregressive conditionally heteroscedastic (GARCH), and SES were considered in the order-up-to-level (OUTL) policy and newsvendor to estimate the safety stock. The authors also investigated the effects of sample size and demand distribution on safety stock. They concluded that as data-driven approaches, nonparametric methods perform less accurately when sample sizes are small, whereas parametric techniques perform well when the distribution is normal. Similarly, (Trapero et al., 2019b)) combined two quantile forecasts (KDE and CGARCH) to minimize the loss function and improve the empirical safety stock predictions using backorders to achieve a lower cost in their newsvendor model.

Another study (Cao & Shen, 2019) presents a data-driven approach, a double parallel feed-forward network, in determining stock levels for a newsvendor problem and its multiperiodic extension.

Demand is assumed time correlated following a normal distribution with a mean of zero and an unknown variance. The proposed approach is shown to be outperforming a number of statistical forecasting methods such as Holt-Winters' triple exponential smoothing. The proposed method captures both stationary and nonstationary time series (Cao & Shen, 2019). Babai, Dai, et al., (2022) provided a comprehensive investigation on estimating demand forecast error under stochastic lead times. The findings highlighted the limitations of the classical demand forecasting approaches and emphasized the importance of considering demand autocorrelation and lead-time variability to enhance the accuracy when selecting a forecasting strategy for inventory control.

Ban & Rudin, (2019) proposed a solution approach for a big data newsvendor problem using single-step machine-learning algorithms. Specifically, they presented two algorithms. The first was based on the empirical risk minimization (ERM) principle, with and without regularization; the second was based on kernel-weights optimization (KO). Furthermore, the authors showed several algorithm properties and tested them with empirical data in a newsvendor-type nurse staffing problem. A data-driven newsvendor problem to approximate demand for retailer's perishable items was proposed (Huber et al., 2019). They used and compared the performance of a number of machine learning-based approaches including a gradient-boosted decision tree, a single-layer neural network, and a nonparametric sample average approximation (SAA) method. Similarly, the application of deep neural networks in identifying the solutions for a newsvendor problem has been investigated (Oroojlooyjadid et al., 2020). Various features of demand data were investigated in terms of their effect on the optimal order quantities per product. Then, the algorithms were extended for (r, Q) policy. The authors validated their results by comparing the proposed methodology with other nonparametric machine learning algorithms including empirical quantile, quantile regression, kernel regression, k-nearest neighbor (KNN), and random forest.

To deal with forecasting intermittent characteristics of demand, bootstrapping methods, WSS (for Willemain, Smart and Schwarz) (Willemain et al., 2004), and VZ (for Viswanathan and Zhou) (C. Zhou & Viswanathan, 2011) was applied to identify lead time demand distribution parameters. Meanwhile, SES, Syntetos and Boylan approximation (SBA), Croston, and neural network (NN) methods were applied to estimate lead time demand and variance of lead time demand calculated through the MSE. The results were compared through an OUTL policy that operates under a cycle service level (CSL) objective. Babai et al. (2020) proposed NN approach with good training and

learning processes that could achieve higher inventory efficiency than the bootstrapping techniques. Inventory holding and back ordering volumes are compared to validate proposed NN approach (Babai et al., 2020). Omar et al. (2023) introduced a novel forecasting approach that leverages customer shopping basket data to predict both online and store sales. Their approach outperforms traditional benchmark methods, such as ARIMA, particularly for products with sporadic demand. Additionally, they demonstrated the advantages of joint forecasting and shared inventory in an omnichannel context, leading to reduced inventory shortages and improved service levels.

Recent studies have also highlighted the effectiveness of ensemble learning in improving demand forecasting and inventory management. Ensemble learning plays a crucial role in demand forecasting for inventory management by combining multiple forecasting models to improve accuracy and robustness. (Zhang et al., 2021) emphasize the advantage of ensemble learning in combining multiple models to enhance forecasting performance. By aggregating diverse predictions, ensemble learning can address individual model biases and capture a wider range of patterns, leading to more reliable and effective demand forecasting and inventory management strategies. Additionally, Zhou et al., (2002) have demonstrated that combining predictors partially can yield comparable or even superior generalization performance compared to combining all predictors simultaneously. Yang et al. (2022) explored the challenges and developments of ensemble deep learning in the era of deep learning, highlighting the need for more efficient methods to deploy ensemble learning in specific fields while reducing the associated time and space overheads. Mohammed & Kora (2023) discussed the potential of integrating ideas from traditional ensemble learning into deep learning to overcome the challenge of tuning optimal hyper-parameters, emphasizing the benefits of cost-effective ensemble deep learning approaches. These findings underscore the importance of adopting effective ensemble forecasting methods to enhance the accuracy of inventory optimization solutions. Despite the above benefits, to the best of authors' knowledge, very limited research has been reported in the literature focusing on the use of ensemble deep learning methods for demand forecasting in supply chain management and inventory optimization stocks (Varghese et al., 2022).

In ensemble learning, combining basic predictors is an essential task (Ganaie et al., 2022). The simple average (Zhou et al., 2018), weighted average (Hu & Chen, 2018), Bayesian model

averaging (BMA) (Seyedan et al., 2022), and meta-learning methods (Song & Dai, 2017) are some of the methods suggested to conduct such combinations. In particular, in many classification and regression tasks, the combinations using meta-learning methods using a stacking generalization technology have demonstrated satisfactory performance (He et al., 2018; Singh et al., 2019; Q. Wang et al., 2019; Zhang et al., 2021). The study of Andrade & Cunha (2023) proposed a methodology for disaggregated demand forecasting in the retail industry. The authors utilized XGBoost, a non-linear non-parametric ensemble-based model, and incorporated a structural change correction method to account for sudden changes in consumer behavior. Their approach demonstrated superior accuracy metrics, reduced stockouts and inventory costs, and a high degree of automation. Furthermore, most ensemble deep learning models focus on the ensemble of homogeneous DNNs and neglect the advantages of heterogeneous DNNs due to their rather complex architecture, reflecting a trade-off between forecast accuracy and model complexity.

Table 2.2 presents a summary of research in applications of machine learning methods in data-driven inventory optimization under demand uncertainty. Various forecasted variables are considered in these studies, including the amount of future demand (Babai et al., 2020; Cao & Shen, 2019), variance of lead time (Babai et al., 2022) and the order quantity (G. Y. Ban & Rudin, 2019; Huber et al., 2019; Oroojlooyjadid et al., 2020). The scarcity of papers included in the table, which specifically explore the application of machine learning methods in data-driven inventory optimization, highlights the relatively limited research efforts in this domain.

Table 2.2. Literature on inventory models with machine learning forecasting methods.

Authors	Review Interval	Inventory Policy	Decision Variables	Objective Function	Forecasting Machine Learning Methods	Estimated Variable	Demand Probability Distribution
(G. Y. Ban & Rudin, 2019)	Single periodic	Newsvendor	-	Min total cost	Quantile regression Kernel regression	Order quantity	Normal
(Cao & Shen, 2019)	Single periodic	Newsvendor	Order-up-to level Base-stock quantity	Min total cost	Double parallel feedforward network	Demand	Normal
(Huber et al., 2019)	Single periodic	Newsvendor	-	Min total cost	Neural networks Gradient-boosted decision trees Linear regression	Order quantity	Normal
(Trapero et al., 2019a)	Periodic Single periodic	(R, S) Newsvendor	Safety stock Order-up-to level Order quantity	Target service level	Kernel density estimation	Variance of lead time demand	Normal, lognormal, gamma
(Trapero et al., 2019b)	Single periodic	Newsvendor	Safety stock Backorders	Target service level	Kernel density estimation	Variance of lead time demand	Normal, lognormal
(Oroojlooyjadid et al., 2020)	Single periodic Continuous	Newsvendor (r, Q)	-	Min total cost	Deep neural network Empirical quantile Quantile regression Kernel regression k-nearest neighbor Random forest	Order quantity	Normal, beta, lognormal, uniform, exponential
(Babai et al., 2020)	Periodic	(R, S)	Safety stock Order-up-to level	Min total cost Target service level	Neural network	Variance of lead time demand	Negative binomial distribution
(Babai et al., 2022)	Periodic	Is not specified	Order-up-to level	Min total cost	Single exponential smoothing & minimum mean squared error forecasting	Demand autocorrelation and variance of lead-time	ARMA(1,1) demand process

2.3. Distribution Forecasting

Despite the extensive use of various neural networks and statistical and other forecasting approaches in inventory management, most of these methods have primarily focused on point forecasting, often centering on forecasting the mean demand (Babai et al., 2020; Goltsos et al., 2022). Other research mainly concentrated on approaches to calculate safety stock, without thoroughly examining the probabilistic characteristics of demand predictions (Trapero et al., 2019a, 2019b). This limited perspective fails to address the crucial need for predicting the complete demand distribution, a fundamental requirement for ensuring the reliability of safety stock estimation. Most of these methodologies also encounter difficulties in capturing the nonlinear relationships among variables (Oroojlooyjadid et al., 2020). Furthermore, Traditional forecasting techniques frequently assume Gaussian iid demand, an assumption that may not reflect the complexities of real-world scenarios (Lee, 2014; Porras & Dekker, 2008). This often leads to inaccuracies in safety stock calculations, resulting in discrepancies between achieved and target service levels, thereby escalating inventory costs.

Conversely, RMDNs, a cutting-edge forecasting technique, represent an advanced forecasting methodology that marks a significant departure from conventional point forecasts (Schittenkopf et al., 2000). As a versatile tool in time series data analysis, RMDNs excel in modeling probability distributions and are particularly effective in the context of safety stock calculations. The strength of RMDNs lies in their ability to model a wide range of distribution types, not limited to the normal distribution. This ability to adapt to various demand scenarios sets RMDNs apart from traditional forecasting methods and aligns closely with the emerging trends in demand forecasting that prioritize comprehensive distributional predictions over mere point estimates.

The true potential of RMDNs lies in their capacity to capture intricate patterns in historical data and adapt to changing market dynamics, thereby enabling organizations to proactively manage and mitigate inventory uncertainties. RMDNs can be applied effectively in distribution forecasting, ensuring that demands for products are not only accurately forecasted but also efficiently distributed to meet customer demand. Nikolaev et al. (2013) extended the capabilities of RMDNs by focusing on modeling time-dependent variances because time series data often exhibit changing variances. Understanding and interpreting these uncertainties are crucial for accurate forecasting.

In recent studies, RMDNs have showcased remarkable versatility across various domains. They excel in energy forecasting, with the presence of various uncertainties affecting demand. In this regard, RMDNs have neem applied in enhancing probabilistic household electrical load predictions for improved power system resource planning (Vossen et al., 2018). RMDNs are also pivotal in demand response within energy management, quantifying residential demand response potential (Shirsat & Tang, 2021). RMDNs have also played a vital role in short-term traffic flow prediction, aiding traffic management and congestion mitigation (M. Chen et al., 2021). Furthermore, linear pre-training for RMDNs holds promise in enhancing network training across various domains, including time series forecasting and natural language processing (Normandin-Taillon et al., 2023). These diverse applications highlight RMDNs adaptability and effectiveness in addressing forecasting challenges in capturing uncertainties.

In summary, the review of literature underscored the critical role of demand forecasting and safety stock estimation in modern inventory management. Although traditional approaches have predominantly centered on point forecasting, a significant research gap exists in predicting the complete demand distribution, a vital requirement for reliable safety stock estimation (Babai et al., 2020; Goltsos et al., 2022). Recent advances in deep learning, particularly RMDNs, show promise in addressing this gap by capturing uncertainties and adapting to a dynamic demand. RMDNs' successful application across various domains highlights their adaptability and effectiveness. Integrating RMDNs into inventory management practices can reduce costs, optimize safety stock levels, and enhance customer satisfaction. In the light of the above review, this study aims at proposing a novel framework utilizing RMDNs for demand forecasting and safety stock calculation. This approach holds substantial promise for retailers and businesses enhancing informed decision-making in inventory management.

In conclusion, this literature review underscores the vital role of demand forecasting and safety stock estimation in inventory management. Traditional approaches, often centered on point forecasting, have evolved to recognize the interdependence of these components and the need to address various challenges. Although the first category assumes perfect knowledge of demand distribution, practical scenarios often lack such certainty. Customer segmentation has thus emerged as a potential solution to improve prediction accuracy by acknowledging diversity in customer behavior. The second category, involving a two-phase process, may result in encounters

with suboptimal solutions. The third category leverages machine learning and deep learning techniques, particularly RMDNs, to model demand distributions. RMDNs offer promise in capturing complex patterns and adapting to dynamic demands. Integrating RMDNs into inventory management has the potential to reduce costs, optimize safety stocks, and enhance decision-making, highlighting the need for further research in this area. Therefore, in this review, we propose a novel framework utilizing RMDNs for improved inventory management practices.

Chapter 3. Cluster-Based Demand Forecasting Using Bayesian Model Averaging: An Ensemble Learning Approach[1]

Mahya Seyedan[a], Fereshteh Mafakheri[b,2], Chun Wang[a]
[a] Concordia Institute for Information Systems Engineering, Concordia University, Montreal, Canada
[b]École nationale d'administration publique (ENAP), Université du Québec, Montréal, Canada

ABSTRACT –Demand forecasting is an important aspect in supply chain management that could contribute to enhancing the profit and increasing efficiency by aligning the supply channels with anticipated demand. In the retail industry, customers and their needs are diverse making demand forecasting a challenging task. In this regard, this study aims at developing a three-step data-driven cluster-based demand forecasting approach for the retail industry. First, customers are segmented based on their recency, frequency, and monetary (RFM) characteristics. Customers with similar buying behaviors are recognized as a segment, creating an ordered relationship between transactions made by them. In the second step, time-series analysis techniques are used to forecast demand for each customer segment. Finally, Bayesian model averaging (BMA) is adopted to ensemble the forecasting results obtained from alternative time series techniques. The applicability of the proposed approach is presented through a comparative case study analysis with presented improvement in the accuracy of daily demand prediction.

Keywords: demand forecasting, customer segmentation, multivariate time-series forecasting, ensemble learning

[1] This paper is published in Decision Analytics Journal 3 (2022) 100033.
[2] Corresponding author: fereshteh.mafakheri@enap.ca

3.1. Introduction

Retail industry has to manage demand and supply planning processes at the operational level while dealing with demand fluctuations as well as uncertainties arising in purchase planning, distribution channels, availability of labor force, and demand for after-sales services (Fildes et al., 2019; Nguyen et al., 2018; G. Wang et al., 2016). Demand forecasting refers to an organization's demand estimation process (G. Wang et al., 2016) to support production, service, and transportation plans (Acar & Gardner, 2012), cost-effective inventory management (Athlye & Bashani, 2018), control the safety stock (Yu et al., 2018), and consequently lowering supply chain costs. Demand forecasting has attracted a lot of attention in the retail industry. (Fildes et al., 2019). A reliable demand forecasting model can help retailers increase profit, promote products, and prevent shortages (Yu et al., 2018). Furthermore, an accurate forecast consequently helps with developing an adaptive pricing strategy for improved revenue management. Time-series forecasting, clustering, K-nearest-neighbors (KNN), neural networks (NN), regression analysis (RA), decision tree (DT), support vector machines (SVM), and support vector regression (SVR) are common methods used to forecast demand (Bozkir & Sezer, 2011; Seyedan & Mafakheri, 2020; Ŝtepnicka et al., 2011).

In recent years, the emergence of online shopping and e-commerce has created a rich source of data on customer information and preferences, including customer demographics (postal code, date of birth, education/income status), past spending patterns, and social media activities (likes and dislikes) (G.-Y. Ban & Keskin, 2020). It is now possible for sellers to use these customer characteristics together with their purchase information to make accurate forecasts of the customers' needs and buying habits. One widely used customer information-based forecasting technique is clustering-based forecasting (López et al., 2015; Venkatesh et al., 2014).

Clustering-based forecasting involves separating customers into disjoint cluster with maximum within-cluster similarity and maximum intra-cluster dissimilarity and constructing a forecasting model on top of each cluster. Due to the inter-cluster similarity, each cluster's prediction models perform better than using the complete data set to build one prediction model. Each customer is assigned to a cluster, and the forecasting model corresponding to that cluster is used to obtain forecasting outcomes. Various factors affect the performance of the clustering-based approach, including the choice of clustering technique, the similarity measurement used, and the predictor (C.-J. Lu & Kao, 2016). In the literature, self-organizing map (SOM), growing hierarchical self-

organizing map (GHSOM), K-means clustering approach as well as various linkage methods have been used to cluster data (C.-J. Lu & Kao, 2016; Murray, Agard, & Barajas, 2015; Murray et al., 2017).

In this regard, depending on the nature of the dataset, various machine learning and statistical tools can be employed for forecasting. In other words, there is no "one fits all" solution. Time-series forecasting is mostly used when the data has an ordered relationship. For example, customer transactions are considered time-series data. Therefore, the forecasts depend on customers' previous purchase patterns (Athlye & Bashani, 2018; Jason, 2018). Machine learning and deep learning methods are mostly generating more accurate forecasts for large time-series data. However, in some forecasting problems, classical methods (such as seasonal autoregressive integrated moving average (ARIMA) and exponential smoothing) would outperform especially in case of one-step forecasting problems with univariate datasets (Jason, 2018; Makridakis et al., 2018). Therefore, it is important to understand how traditional time-series forecasting methods work and evaluate them before exploring more data-intensive techniques.

The literature suggests using ensemble learning to combine forecasting results as a means of achieving higher accuracy (Alqurashi & Wang, 2019; Galar et al., 2012; Gastinger et al., 2021; Nilashi et al., 2017). In the ensemble learning approach, different predictors (each with different performance) are combined to arrive at predictions. The predictors are combined such that each model covers the weakness(es) of another approach and improves the overall accuracy. A random forest algorithm is an example of ensemble learning where a collection of trees is used instead of a single tree predictor (C.-J. Lu & Kao, 2016). Majority voting is the most commonly used ensemble learning method, where no parameter tuning is required once each predictor is trained (C.-J. Lu & Kao, 2016). In addition to increased accuracy, superiorities in robustness, stability, confidence of estimation, parallelization, and scalability are other benefits of ensemble learning (Ghaemi et al., 2009). Bayesian model averaging (BMA) is also a well-known aggregation tool for combining results to improve forecast accuracy (Raza et al., 2017). BMA uses the entire data set for the inference-making process to avoid individual model dependence. In BMA aggregation process, combinational weights are assigned to each individual model. Models with higher accuracy gain a higher weight than the lower-performing ones (Li & Shi, 2010).

In this paper, we propose a multi-stage demand forecasting approach that combines clustering and ensemble learning techniques to improve the accuracy of the customer behavior forecasts for the

retail sector. The main novelty rests in the fact that by means of clustering, the dataset will be segmented to ensure generation of more accurate forecasts for each cluster. Then, using ensemble learning, the clustered forecasts will be combined to map the whole dataset again. A combinatorial forecasting model is then used to generate the combined forecasts. To show the applicability and usefulness of the proposed approach, it is implemented in a real-world case where demand data for sports products is used to provide forecasts for the next demand cycle.

The rest of this paper is structured as follows: Section 3.2 presents the proposed three-step methodology for demand forecasting. The results and related discussions will be presented in Section 3.3. We conclude the paper in Section 3.4, providing an overview of the approach and directions for future research.

3.2. Methodology

As mentioned earlier, the main objective of this study is to propose an effective demand forecasting approach with improved accuracy. The proposed approach has two main steps: customer segmentation and multivariate time-series demand forecasting. An overview of the proposed approach is shown in Figure 3-1. First, data is preprocessed, and the outliers interpolated (details provided in Section 3.2.1). Then, target and predictor variables are determined. The target variable is considered as sales quantity (demand). However, it could also be considered as sales in dollars, discounts in dollars, or demand in quantity. The training phase includes customer segmentation and demand forecasting. The segmentation deals with clustering of customers. First, data is clustered using different algorithms (K-means, single linkage, complete linkage, centroid linkage, and Ward's linkage). Then, the resulting clusters are combined to form three segments of low, mid, and high representing customer affordability (more details are provided in Section 3.2.2.1). After customer segmentation, LSTM and Prophet algorithms are used to establish the forecasts of demand corresponding to each customer segment. These algorithms are selected due to their superior performance in time-series forecasting problems (Abbasimehr et al., 2020; Rostami-Tabar & Rendon-Sanchez, 2021). LSTM is a recurrent neural network with long and short-term memory units suitable for forecasting order-dependent data. It is designed to direct the neural network in extracting longer-term trends and learning about longer-range dependencies in data (Parmezan et al., 2019). As purchasing habits usually follow a cyclic pattern, investigating the application of LSTM in our dataset is appealing. Also, Prophet is suitable for data containing seasonality and

holiday effects (Beneditto et al., 2021). The details on implementation of LSTM and Prophet methods are discussed in Section 3.2.2.2.

The testing phase consists of a procedure for evaluating the model's performance. The training data is fed into the clustering and segmentation algorithms. Then, trained prediction models are applied to each segment. The outputs of individual predictors are then ensembled using the BMA to generate forecast outputs (section 3.2.2 provides more details on testing phase).

3.2.1. Data Preprocessing

To ensure accuracy of the proposed demand clustering and forecasting, outlier detection and interpolation for collected test data are conducted using an unsupervised SVM. The data is scaled, and then, *OneClassSVM* is used to map the data into labels of 0 (for normal) or 1 (for abnormal) (see (Pedregosa et al., 2011)). The outliers can then be replaced by values obtained using linear interpolation. The selected data set have 52 features in total. However, not all these features are valuable for (and contribute to) demand forecasting. Each record of data set corresponds to a unique transaction ID and sale quantity (i.e., target variable). The selected set of predictor variables are date, time, sale price, holidays, and demand in the previous period.

3.2.1. Training Phase

The training phase includes two steps of 1) clustering and ensemble segmentation using majority voting, and 2) training of prediction models of each segment described as follows:

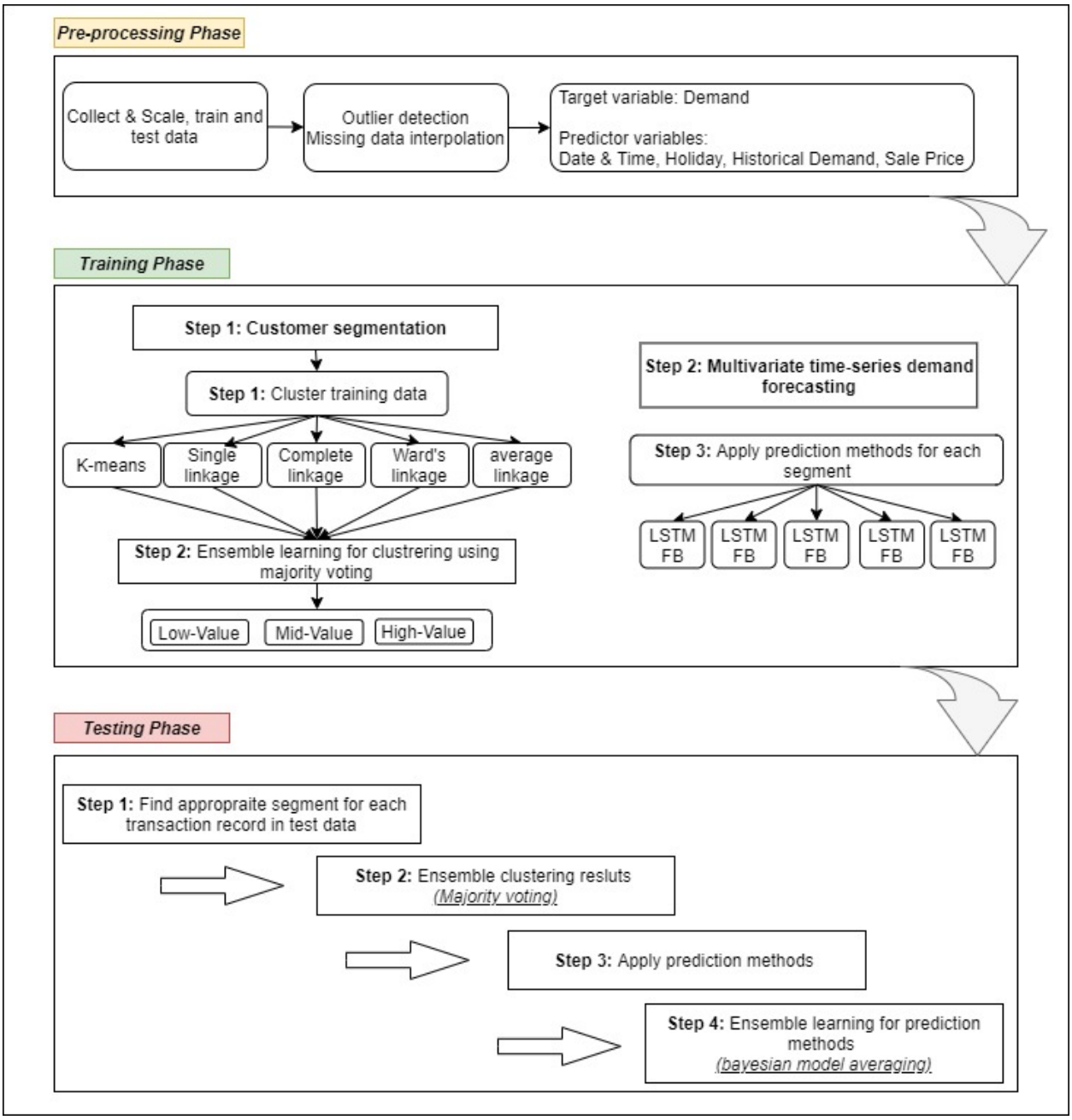


Figure 3-1 The Proposed Ensemble Modelling and Clustering-Based Multivariate Time-Series Demand Forecasting

3.2.2.1. Clustering

The choice of clustering methodology and feature selection depends on the business goals and needs. For example, if the goal is to increase the retention rate, segmentation can be done using churn probability (Murray et al., 2017). Among various existing segmentation methods, a RFM method is chosen due to its ability to segment customers who are highly likely to respond to a marketing campaign (Coussement et al., 2014). This clustering method classifies customers as low, mid, and high value-added customers. The customer segmentation is done using five clustering algorithms (K-means, single linkage, complete linkage, centroid linkage, and Ward's linkage), and then an ensemble learning will be done using majority voting method (C.-J. Lu & Kao, 2016). In doing so, three segments of customers are identified:

- Low-value: Those who are less active than others and less frequent buyers/visitors. They generate very low to zero or even negative revenue.
- Mid-value: These customers are in the middle. They often buy/visit (but not as much as the high-value customers) frequently and generate moderate revenue.
- High-value: These customers are most frequent buyers, and businesses do not want to lose them. They correspond to a high share of revenue and a high frequency of purchase.

Three features—recency I, frequency (F), and monetary (M)—were calculated for each row of data set as the basis for clustering. R implies the most recent purchase date of a customer. It is defined as the number of days since the last purchase of a customer. F is the total number of purchase orders for a customer in each period of time. The higher the frequency, the more active the customer. Similar to recency, a high-frequency customer is more value-added. M indicator shows how much money a customer has spent on his/her purchases to date.

Five clustering algorithms were then used, considering the above features. These are K-means, single linkage, complete linkage, centroid linkage, and Ward's linkage. We chose these algorithms to cover a broad range of selection criteria and to comparatively observe/evaluate their performance. As clustering methods could have different distance criteria, each could assign a record to a different cluster. We then use majority voting to combine the results of these clustering algorithms (C.-J. Lu & Kao, 2016). We chose the cluster with the highest vote and assigned it to

the customer ID. In case of equal votes, a random selection is made. In this process, the Elbow method is used to find the optimal number of clusters (Han et al., 2013).

At this point, the cluster numbers (please refer to section 3.3 for more details) are summed up for all features (using RFM method). The overall scores of segmentations were 0–2 (for low-value), 3–4 (for mid-value), and 5+ (for high-value). The distance criterion for each clustering algorithm will be discussed below. In this regard, C.-J. Lu and Kao (C.-J. Lu & Kao, 2016) could serve as a reference for more detailed information about the distance criteria and the corresponding clustering methods discussed as follows:

K-means method: K-means algorithm separates the data points into *k* cluster of equal variances. More precisely, the algorithm splits a set of *n* records into *j* = 1, 2, …*k* disjoint clusters I, each characterized by a mean of μ_j of samples x_i (values) in the cluster. The mean is called centroid and is representative of the cluster. The algorithm works based on minimizing a within-cluster sum of square (I. F. Chen & Lu, 2017) as follows:

$$\sum_{i=0}^{n} \min_{\mu_j} \|x_i - \mu_j\| \tag{3-1}$$

Single linkage method: In this method, the designated cluster for a sample is the one with minimum Euclidean distance. Assuming that $x_{kj}^{(d)}$ is the *k*th value of *j*th predictor in the cluster *d* identified in training phase, the Euclidean distance between the predictor x_i of testing data y_i^* and the predictors $x_{kj}^{(d)}$ of training data $y^{(d)}$ in cluster *d* can be computed as:

$$\lambda_{ik}^{(d)} = \sum_{j=1}^{p} \left(\sqrt{(x_{ij} - x_{kj}^{(d)})^2}\right) \qquad for\ i = 1,2,I,n,d = 1,2,I,g. \tag{3-2}$$

In single linkage method, distance $S_i^{(d)}$ between the test data y_i^* and the cluster *d* is the minimum value of $\lambda_{ik}^{(d)}$, i.e. $S_i^{(d)} = min(\lambda_{ik}^{(d)})$. In this sense, the best representative cluster for test data y_i^* is cluster SG_i that corresponds to smallest $S_i^{(d)}$, i.e. $SG_i = arg\ min\ (S_i^{(d)})$ (Carlsson et al., 2018; Jain & Dubes, 1988; C.-J. Lu & Kao, 2016).

Complete linkage method: The difference between a single linkage method and a complete linkage method is that the distance between test data y_i^* and cluster *d* is the maximum value of $\lambda_{ik}^{(d)}$, i.e. $C_i^{(d)} = max\ (\lambda_{ik}^{(d)})$. Thus, the best representative cluster for test data y_i^* will be a cluster

CG_i that corresponds to the smallest $C_i^{(d)}$, i.e. $CG_i = arg\ min\ (C_i^{(d)})$ (Jain & Dubes, 1988; C.-J. Lu & Kao, 2016).

Average linkage method: In contrary to the methods that use farthest or nearest points to calculate a similarity value, an average linkage method measures the distance between clusters using their averages, advocating the fact that a mean is an indicator of data centricity. In doing so, the mean of a *j*th predictor in cluster *d* is defined as:

$$\alpha_j^{(d)} = \frac{1}{n_d}\sum_{i=1}^{n_d} x_{ij}^{(d)} = mean(\{x_{ij}^{(d)}, \forall i = 1,2,..,n_d\}) \quad (3\text{-}3)$$

Thus, the best representative cluster for test data y_i^* will be a cluster AG_i that corresponds to the smallest $\alpha_j^{(d)}$, i.e. $AG_i = arg\ min\ (\alpha_j^{(d)})$ (C.-J. Lu & Kao, 2016).

Ward's linkage method: Ward's linkage method defines the distance between two clusters by calculating and minimizing a within-cluster variance. First, test data y_i^* is included in cluster *d* with a centroid value of $o_j^{(d)}$ calculated as follows:

$$o_j^{(d)} = \frac{x_{ij} + \sum_{k=1}^{n_d} x_{kj}^{(d)}}{1 + n_d} \qquad \forall j = 1,2,I,p. \quad (3\text{-}4)$$

Then, Ward's linkage distance $W_i^{(d)}$ between test data y_i^* and cluster d is calculated as:

$$W_i^{(d)} = \sum_{j=1}^{p}(\sqrt{(x_{ij} - o_j^{(d)})^2} + n_d \times \sqrt{(a_j^{(d)} - o_j^{(d)})^2}) \quad (3\text{-}5)$$

where n_d is number of data points in cluster *d*.

In this sense, the best representative cluster WG_i for test data y_i^* is determined as $WG_i = arg\ min\ (W_i^{(d)})$ (Jain & Dubes, 1988; C.-J. Lu & Kao, 2016).

3.2.2.2. Demand Forecasting

In previous section, we reviewed several clustering methods that can provide an analysis of transaction records and conduct a segmentation of customers based on RFM features. The output of clustering step (customer clusters) can be used as an input for demand forecasting. To predict daily demand for each segment, the following procedure is proposed in this paper. First, a

summation of daily transaction records is established. Then, other features such as sales price, holiday and other dates effect, as well as demand in the previous periods, are used as predictor variables to establish forecasts for the target variable, i.e., daily demand through alternative forecasting algorithms of LSTM and Prophet.

3.2.2.2.1. LSTM

LSTM method uses recurrent NNs with memory units and loops that allow the information to persist. LSTM has an input gate, a forget gate, an internal state, and an output gate. The terms used in LSTM formulation are:

$x(t_i)$: input value

$h(t_{i-1}), h(t_i)$: output values

$c(t_{i-1}), c(t_i)$: cell states

$b = \{b_a, b_f, b_c, b_o\}$ biases of input gate, forget gate, internal state and output gate

$W_1 = \{w_a, w_f, w_c, w_o\}$: weights of input gate, forget gate, internal state, and output gate

$W_2 = \{w_{ha}, w_{hf}, w_{hc}, w_{ho}\}$: recurrent weights

$a = \{a(t_i), f(t_i), c(t_i), o(t_i)\}$: output results for input gate, forget gate, internal state, and output gate

The following equations present the integration of the above elements in forming an LSTM cell:

$$a(t_i) = \sigma(w_a x(t_i) + w_{ha} h(t_{i-1}) + b_a) \quad (3\text{-}6)$$

$$f(t_i) = \sigma(w_f x(t_i) + w_{hf} h(t_{i-1}) + b_f) \quad (3\text{-}7)$$

$$c(t_i) = f_t \times c(t_{i-1}) + a_t \times tanh(w_c x(t_i)) + w_{hc}(h(t_{i-1}) + b_c) \quad (3\text{-}8)$$

$$o(t_i) = \sigma(w_o x(t_i) + w_{ho} h(t_{i-1}) + b_o) \quad (3\text{-}9)$$

$$h(t_i) = o(t_i) \times tanh(c(t_i)) \quad (3\text{-}10)$$

After identifying $c(t_i)$ and $h(t_i)$, error (variance) between the predicted data and input data is calculated. This error is propagated back to input gate, cell gate, and forget gate, as feedback, to update the weights following an optimization algorithm (such as gradient descent approach or similar) (Abbasimehr et al., 2020).

3.2.2.2.2. Prophet

Turning to Prophet, this forecasting tool consists of autoregression models to predict time-series data capable of handling seasonality, holidays effect, strong shifts in trends, and outlier existence in data (Rostami-Tabar & Rendon-Sanchez, 2021; Taylor & Letham, 2018). This forecasting tool is automated in tuning time-series (Beneditto et al., 2021). Prophet fits several linear and nonlinear time functions according to the following equation:

$$y(t) = g(t) + s(t) + h(t) + e(t) \tag{3-11}$$

where $g(t)$ is a trend representing non-periodic changes (such as growth or decay over time), $s(t)$ is a seasonality term representing periodic changes (such as daily, weekly, or monthly), $h(t)$ represents the holiday effects, and $e(t)$ represents noise in data. The $g(t)$ follows either a saturating growth model or a piecewise growth model. As such, the saturating trend is given by:

$$g(t) = \frac{\mathrm{C}}{1 + \exp\left(-\mathrm{k}(\mathrm{t} - \mathrm{m})\right)} \tag{3-12}$$

where C is carrying capacity, *k* is the growth rate, and *m* is an offset parameter.

In the case of a piecewise growth model (i.e. the default method in Prophet), the trend is defined as:

$$g(t) = (k + a(t)^T \delta)t + (m + a(t)^T \gamma) \tag{3-13}$$

where k is the growth rate, and δ is a vector of rare adjustments, where δ_j is the change in rate at time t. The rate at time t is represented by a base rate k plus all adjustments up to that point. This is represented by the following function $a(t)$

$$a_j(t) = \begin{cases} 1, & if\ t \geq s_j, \\ 0, & otherwise. \end{cases}$$

As for the seasonality trend $s(t)$, the aim is to provide adaptability in the model to periodic changes following a sub-daily, daily, weekly, and yearly seasonality. Use of a standard Fourier model is advocated to establish a seasonality trend. Holiday effects $h(t)$ is also incorporated by generating a matrix of regressors. More details about Prophet and the above procedure can be found in Taylor & Letham (Taylor & Letham, 2018).

3.2.2. Testing Phase

The objective of testing phase is to ensemble predicted values of the forecasting methods using BMA. In this phase, firstly, the above reviewed clustering methods are implemented for test data set and the results are ensembled by a majority voting. At the end of clustering section, three customer segments (low, mid, and high) are identified. Secondly, a pre-trained forecasting method (LSTM and Prophet) is applied to test data in each cluster. Finally, ensembling the results from different forecasting methods is done by combining them into a single predictor. The BMA method weighs these forecasts according to their posterior model probabilities. It, therefore, forms an averaged model with better-performing predictions having higher weights than worse-performing predictions (Li et al., 2011). BMA is implemented using widely applicable information criterion (WAIC), or leave-one-out cross-validation (LOO) approaches to estimate the above-mentioned weights as follows:

$$w_i = \frac{e^{-\frac{1}{2}dIC_i}}{\sum_j^M e^{-\frac{1}{2}dIC_j}} \tag{3-14}$$

where dIC_i is the difference between the i-th information criterion value and the lowest value. In general, a lower dIC is better. This approach is called pseudo-Bayesian model averaging, or Akaike-like weighting, and it is a heuristic way to calculate the relative probability of each model (given a fixed set of models) from the information criteria values. The denominator ensures that the weights sum up to one (Raza et al., 2017; Y. Zhou et al., 2020).

3.2.3. Assumptions

- RFM method assumes that customers are sensitive to marketing campaigns but react differently to marketing strategies. RFM aims at ranking the customers based on their purchasing habits, in order for those with high ranks being targeted with marketing campaigns (Coussement et al., 2014). This concept is considered applicable to the dataset presented in this study. Thus, an RFM-based clustering is employed for demand prediction by focusing on customers with similar RFM ranks.
- It is assumed that no unprecedented events will happen in the future (such as pandemics, global recession, international shipping problems, crashes in shopping platforms, etc.).
- The demand is predicted on a daily basis.

3.3. Results Analysis and Discussion

3.3.1. Data Description and Performance Criteria

The time-series data was collected from an open-source data set available in (Constante et al., 2019). It consisted of the supply chain data of three products: clothing, sports, and electronics supplies. The focus of this study is on sports products as the corresponding dataset was of larger size and with fewer missing values. Although, this dataset consisted of 52 features (Figure 3-2), most of these features were not valuable (needed) for demand forecasting. The dataset contains of records from January 2015 to September 2017. Figure 3-3 shows the daily demand for sports products corresponding to the last three months of dataset. The demand varied from 250 to 500 depending on time, sales price, and discounts.

The accuracy of the predictions was evaluated using Mean absolute percentage error (MAPE), Mean absolute error (MAE), and Root mean square error (RMSE). In doing so, a lower value indicates a better prediction performance (or less inaccuracy). The definitions of these criteria are provided as follows:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{3-15}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{3-16}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3-17}$$

where y_i and $\hat{y}_i$ show the actual and predicted values at day i, respectively. Also, n is the total number of testing days (30 days in this case).

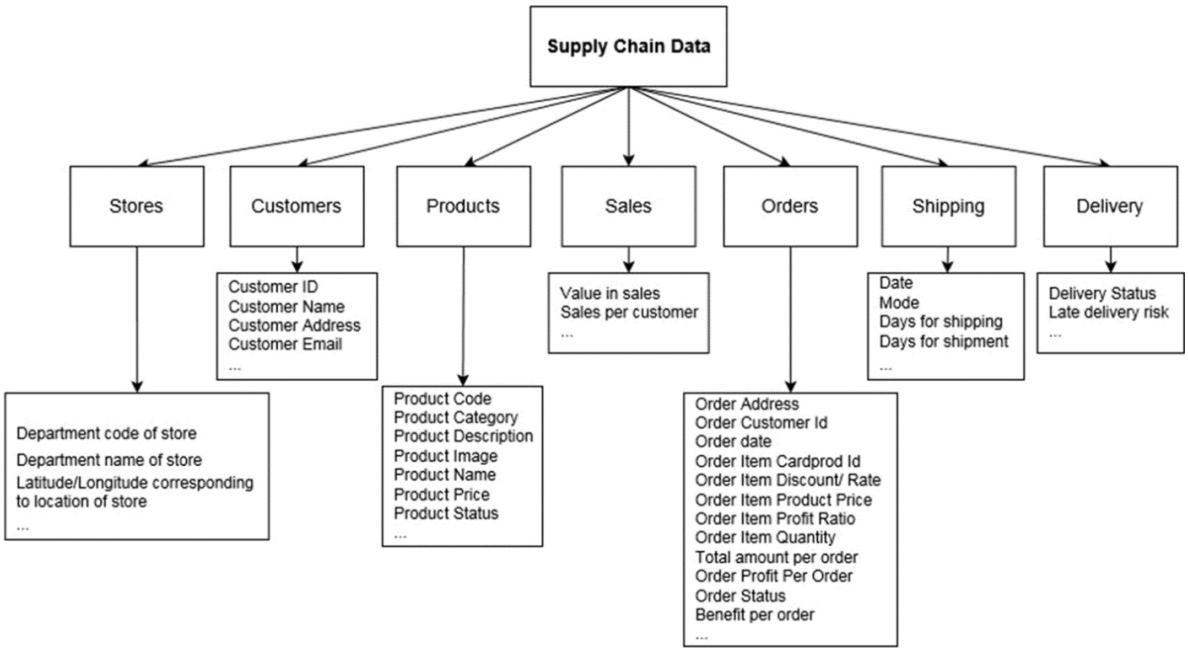


Figure 3-2 Taxonomy of Features in Sports Data Set

Note: Demand, customer ID, purchase date, sales price, amount purchased, frequency of customer visits, and holidays were used or extracted from the features (Seyedan & Mafakheri, 2020).

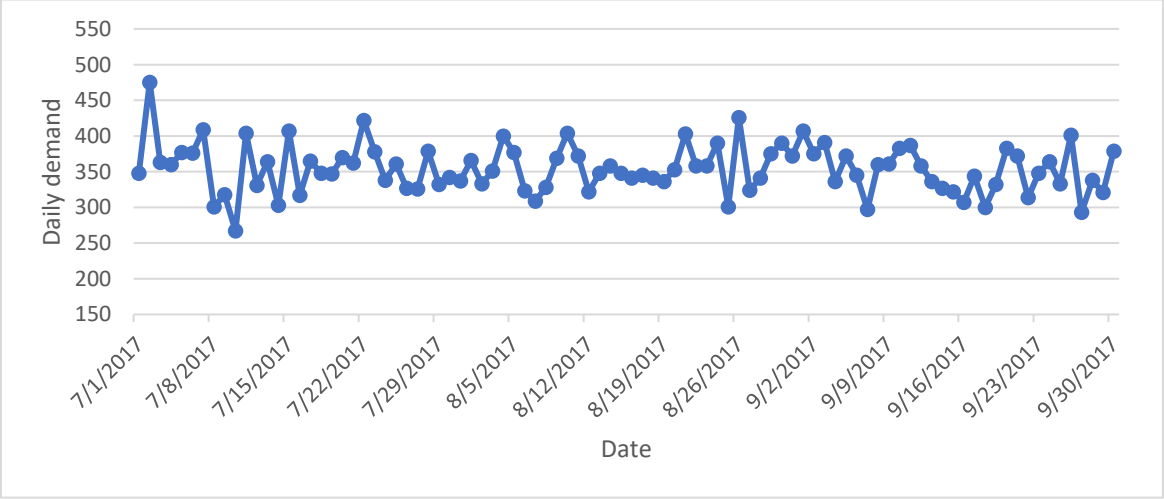


Figure 3-3 Daily Demand (quantity) for Sports Products

3.3.2. Clustering and Predictions

To calculate recency, for each transaction, we subtracted the snapshot day (the current day) from the date the transaction was performed. Frequency was also calculated as the number of transactions made by each customer. For monetary value (of a customer), we summed up all transactions by each customer. In order to establish the above features, it is essential to identify the optimum number of clusters to ensure minimization of each cluster's within-cluster variance. The elbow method (Han et al., 2013) was employed to identify the optimum value of k for each feature (i.e. optimal number of clusters). Results are presented in Figure 3-4, showing that $k = 4$ is optimal for recency, frequency, and monetary clusters. For each cluster, one score (from 0 to 3) is assigned according to their cluster centroid values. In recency, this score is 0 for the highest cluster centroid and 3 for the lowest cluster centroid, while in case of frequency and monetary; the situation is in opposite. Finally, by calculating the overall scores of customers (summing up all scores for recency, frequency, and monetary), they are clustered into three customer segments: low-value customers, mid-value customers, and high-value customers. Figure 3-5 shows these customer segments based on the two-dimensional features and K-means clustering. High-value customers are those who purchased more recently, tend to buy more frequently, and spend more than other clusters. After applying different clustering methods, using majority voting, we combined the outcomes.

In the next step, we aggregated daily demand and average daily sale prices for each customer segment. We then used them as input to predict the next month's daily demand. We extracted two additional features (besides historical daily demand and sale prices)—of dates and holidays—because they strongly affect future demand. The average daily demand was calculated for holidays (364) and non-holidays (362). This shows that customers were more inclined to buy products during holidays rather than non-holidays. To predict the future daily demand, we turned to Prophet and LSTM algorithms as described in Section 3.2.2.2.

Using an exhaustive search approach, the optimal parameters chosen for our LSTM were established as follows: number of hidden layers = 50, number of epochs = 50, batch size = 72, with *adam* selected as the accuracy optimizer function. In addition, for Prophet, we considered daily, weekly, and yearly seasonality data forming a multiplicative seasonality mode (Beneditto et al., 2021).

After applying LSTM and Prophet, each method established a predicted value for the daily demand for next (following) month. To ensemble the two forecasting results, we used simple average and BMA with the results described in the following sections.

3.3.3. Accuracy Test

To validate the proposed clustering-based forecasting approach with ensemble learning, we considered three different scenarios. In the first scenario, the forecasting methods were applied to raw data without clustering. The results of three performance evaluation criteria (MAPE, MAE, and RMSE) were presented in Table 3.1, showing that the lowest RMSE was achieved in forecasting results ensembled by BMA.

In the second scenario (

Table 3.2), five clustering methods were considered, and the results were shown for each customer segment. The LSTM and Prophet results were ensembled using simple averaging and BMA. These results were shown in Table 3.3. We combined the clustering results using a majority voting. Then, we forecasted the daily demand by applying alternative multivariate time-series forecasting. Finally, we ensembled the results using BMA. The error was less than other scenarios with MAPE = 8.12%, MAE = 27.1, and RMSE = 32.8. The results showed that customer segmentation did increase the accuracy of the predictions. In addition, by adopting multivariate forecasting, we could integrate the impact of additional features on daily demand predictions, making them more accurate compared to univariate forecasting. Furthermore, use of ensemble learning and combining the results of different clustering and forecasting models, by means of a majority voting in the clustering section and BMA in the prediction section, has further improved the accuracy of predictions making them more representative of reality.

3.3.4. Sensitivity Analysis

This section discusses the influence of prediction variables and how they contribute to demand forecasts. The effects of holidays and average daily sales on demand were presented in Figure 3-6, with average daily sales and holiday variables changing across their ranges (minimum and maximum values derived based on historical data) while keeping the rest of the variables fixed. Figure 3-6 shows that:

- There is an increasing trend for sales in both holidays and weekdays. When the daily demand rises, the average price of the products rises as well leading to higher sales.

Average price of products increases, daily demand rises as well. This incrimination shows that when the market is hot (high demand), people have higher preference for online shopping (with willingness to pay a higher price in such a hot market). This reflects the demand and sale direct dependency.

- The graph also shows that buyers shop online more on holidays than weekdays when the average daily sale is less than a certain threshold (about $200). This outcome may be due to the customers' preference to buy from retail stores in person and avoid online shopping when the prices are higher.

Table 3.1 Forecasting Performance Criteria—Without Clustering

Methods of Forecasting	MAPE %	MAE	RMSE
LSTM	8.169	67.147	111.927
Prophet	9.920	304.366	316.661
Simple average	8.753	29.041	35.813
BMA	8.177	27.393	32.887

Table 3.2 Forecasting Performance Criteria—Using Clustering Methods

Methods of Clustering	Methods of Forecasting	MAPE %	MAE	RMSE
K-means clustering	LSTM	8.213	205.340	248.496
	Prophet	8.296	155.665	214.737
	Simple average	8.153	27.775	33.339
	BMA	8.252	28.234	34.836
Single linkage clustering	LSTM	8.434	75.153	124.783
	Prophet	10.016	293.855	309.561
	Simple average	8.967	29.738	36.266
	BMA	8.494	28.433	33.675
Complete linkage clustering	LSTM	8.680	103.018	160.688
	Prophet	10.720	267.247	293.250
	Simple average	9.501	31.590	37.914
	BMA	8.666	29.140	34.271
Average linkage clustering	LSTM	8.506	65.114	109.038
	Prophet	9.898	307.646	321.007
	Simple average	9.074	30.217	36.523
	BMA	8.446	28.373	33.890
Ward's linkage clustering	LSTM	8.885	143.755	202.902
	Prophet	8.622	212.575	258.640

Simple average	8.703	29.224	34.849
BMA	8.655	29.293	34.948

Table 3.3 Forecasting Performance Criteria— with Majority Voting in Clustering

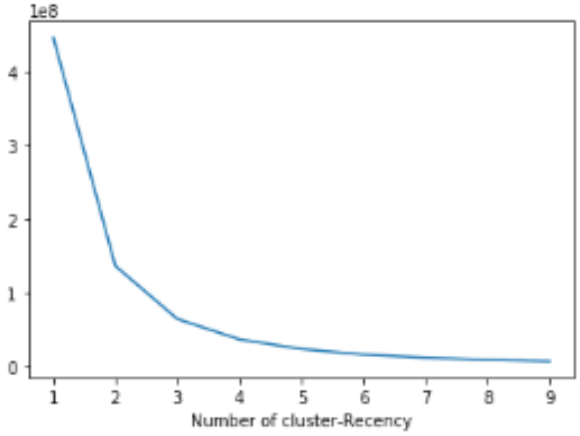
Methods of Forecasting	MAPE %	MAE	RMSE
LSTM (univariate)	8.126	125.300	150.835
Prophet (univariate)	8.951	216.550	225.696
Simple average	8.464	28.524	33.739
BMA	8.219	27.760	32.984
LSTM (multivariate)	8.076	62.723	104.985
Prophet (multivariate)	9.928	306.254	319.201
Simple average	8.842	29.314	35.815
BMA	8.120	27.106	32.801

In addition, to test the robustness of the methodology proposed in section 3.2, it is applied to an alternative dataset of electronic products (Constante et al., 2019). The proposed methodology also was well-performed with the electronic products dataset. For instance, the proposed majority voting method with ensemble forecasting results showed a 5.65% accuracy improvement compared to the scenario without clustering. The results obtained for the alternative dataset are shown in Table 3.4.

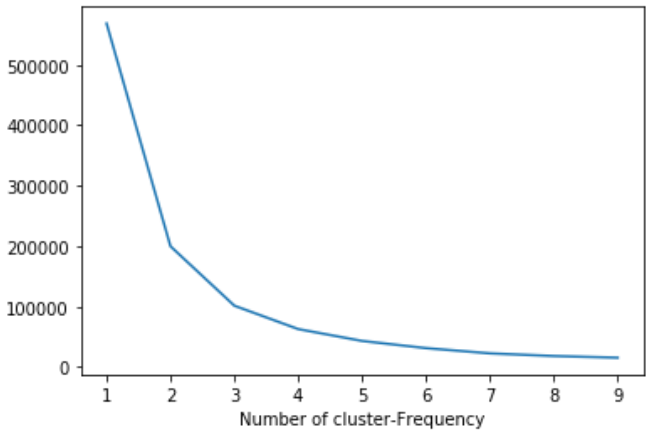
Table 3.4 Performance Improvement by Ensemble clustering (for Electronic products dataset)

Methods of Clustering	Methods of Forecasting	Performance Improvement by Ensemble Clustering (%)
Without clustering	BMA (LSTM & Prophet)	5.65
K-means clustering		6.08
Single linkage clustering		4.58
Complete linkage clustering		9.12
Average linkage clustering		6.75
Ward's linkage clustering		2.84

(a)



(b)



(c)

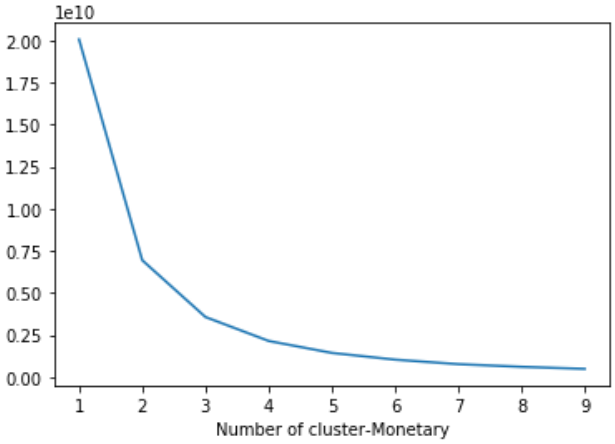
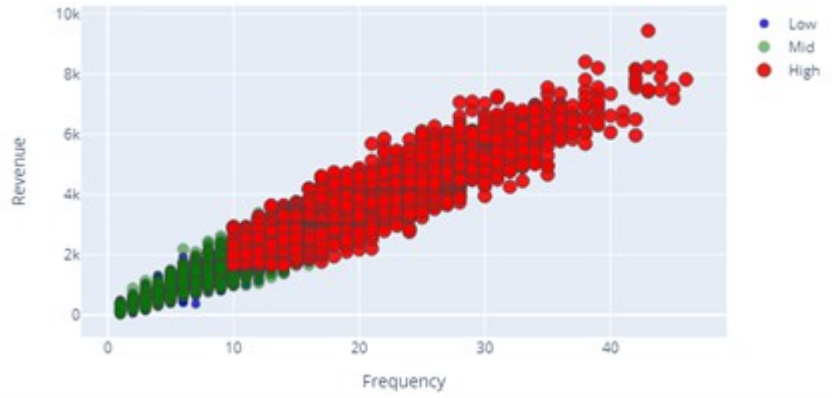
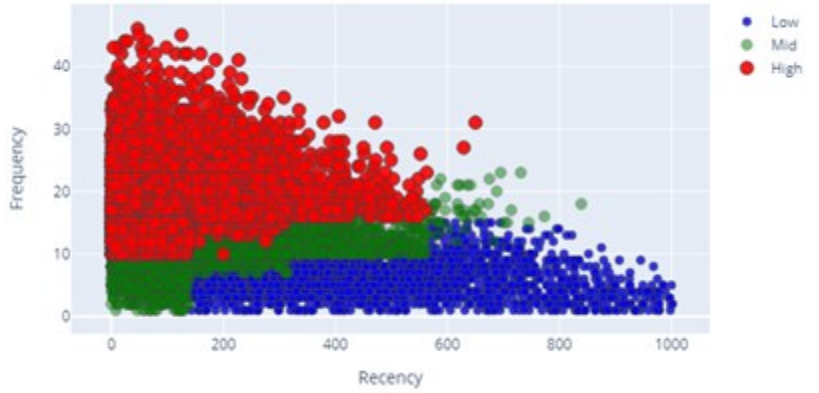


Figure 3-4 Optimal Number of Clusters for a) Recency, b) Frequency, & c) Monetary

(a)



(b)



(c)

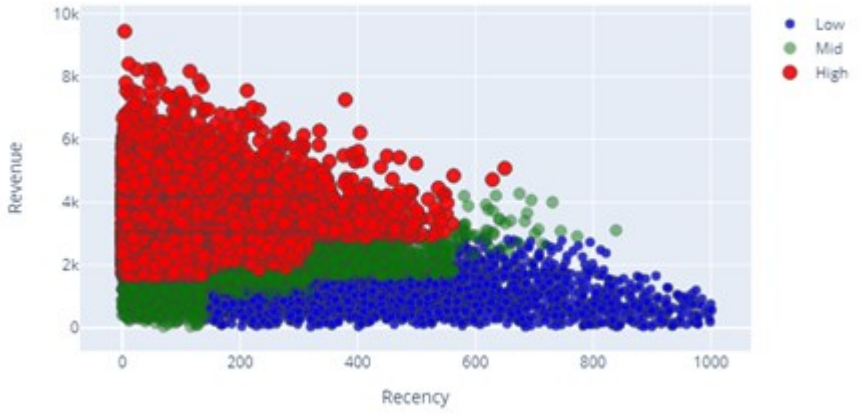


Figure 3-5 Customer Segmentations using K-means: a) Revenue vs Frequency, b) Frequency vs Recency, c) Revenue vs Recency.

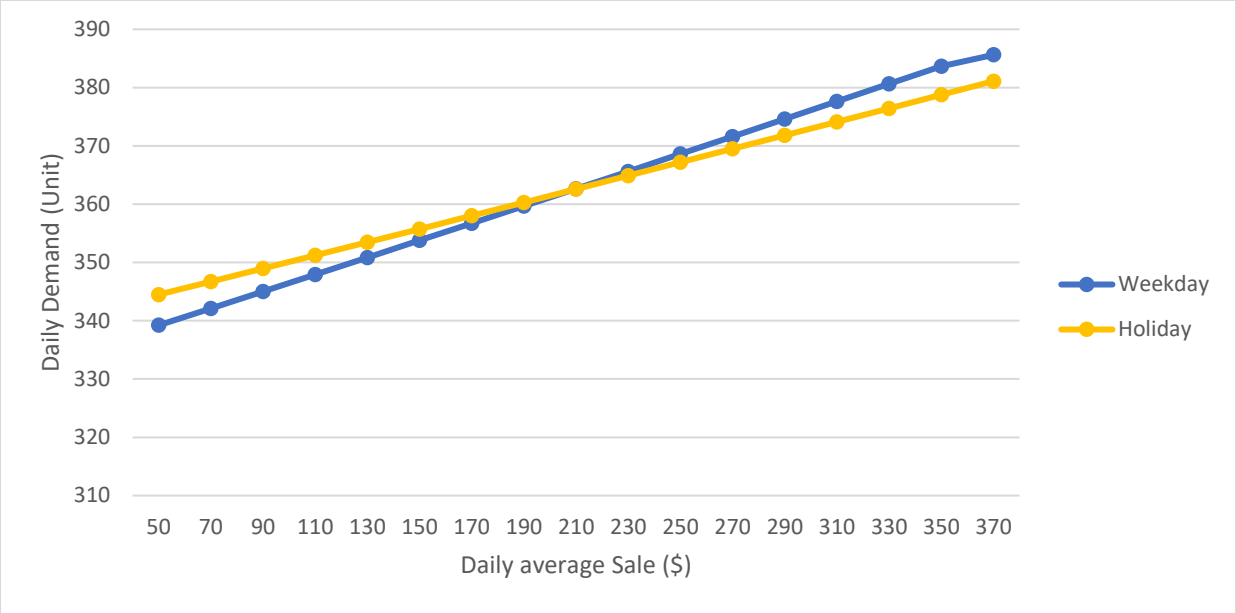


Figure 3-6 Effect of Holiday and Average Sales on Demand Forecasts

3.4. Conclusions

Demand forecasting highly affects business decisions, such as production planning, inventory management, financial planning, and sales. As a result, enhancing forecasting accuracy can improve a firm's decision-making efficacy, reduce risk of unmet demand, and lower operational costs by preventing over production. This study proposed a methodology for demand forecasting using ensemble learning in case of sports retail industry to improve the accuracy of future daily demand forecasting.

The proposed forecast framework provided a cluster-based demand prediction using time-series forecasting methods of LSTM and Prophet. In addition, we used majority voting and BMA as ensemble learning techniques in clustering and forecasting, respectively. The aim was to achieve a higher forecast accuracy in demand prediction by intelligently combining different models' performances and assigning higher weights to the better-performing models. The prediction performance of the proposed forecasting framework was presented for sports products data set.

The proposed framework was analyzed under different forecasting scenarios, including with clustered and without clustered customers as well as with or without ensemble learning. The sensitivity analysis of forecast results showed an improvement in prediction accuracy of the clustered-ensembled approach compared to use of single models, resulting in the minimum values

of MAPE (8.12%), MAE (27.1), and RMSE (32.8) for daily forecasts. BMA effectiveness was also observed in terms of forecast error reduction. The proposed cluster-based forecasting framework had considerably increased prediction accuracy across various seasonal and monthly cases.

The proposed framework has a number of limitations. We made efforts to develop a generic methodology applicable to different sets of supply chain data while using a minimum number of variables. Access to more information about customers such as income rate, age, location, monthly budget, and similar personalized data could significantly improve the segmentation.

Access to a larger data set could also help better project a multi-period demand forecast and improve forecasting accuracy. We used a dataset of sports products. However, having multiple retail data from different sources could enhance the performance of predictions. Just to present the robustness of our proposed approach, it was also tested on an alternative dataset (electronic products).

Future avenues of research could include combining the proposed forecasting model with an optimization model with prescriptive capabilities in response to future demand scenarios and expectations. The literature has indicated that while predictive analytics have been utilized in demand management and procurement, prescriptive analytics have rarely been applied to demand forecasting.

Chapter 4. Order-Up-To-Level Inventory Optimization Model using Time-Series Demand Forecasting with Ensemble Deep Learning[1]

Mahya Seyedan[a], Fereshteh Mafakheri[b,2], Chun Wang[a]
[a] Concordia Institute for Information Systems Engineering, Concordia University, Montreal, Canada
[b]École nationale d'administration publique (ENAP), Université du Québec, Montréal, Canada

ABSTRACT – Inventory control aims to meet customer demands at a given service level while minimizing cost. As a result of market volatility, customer demand is generally changing, and ignoring this uncertainty could lead to under or over-estimation of inventories resulting in shortages or inefficiencies. Inventory managers need batch ordering such that the ordered items arrive before the depletion of stocks due to the lead time between the ordering point and delivery. Therefore, to meet demand while optimizing the cost of the inventory system, firms must forecast future demands to address ordering uncertainties. Traditionally, it was challenging to predict such uncertainties with high accuracy. The availability of high volumes of historical data and big data analytics have made it easier to overcome such a challenge. This study aims to predict future demand in the case of an online retail industry using ensemble deep learning-based forecasting methods with a comparison of their performance. Compared to single-model learning, ensemble learning could improve the accuracy of predictions by combining the best performance of each model. Also, the advantages of deep learning and ensemble learning are combined in ensemble deep learning models, allowing the final model to be more generalizable. Finally, safety stocks are estimated using the forecasted demand distribution, optimizing the inventory system under a cycle service level objective.

Keywords: Inventory management; demand forecasting; optimization; ensemble deep learning; multivariate time-series forecasting.

---

[1] This paper is published in the journal of Supply Chain Analytics 3 (2023) 100024.
[2] Corresponding author: fereshteh.mafakheri@enap.ca

4.1. Introduction

Inventory control is associated with decisions on ordering time and quantities of multiple stock-keeping units (SKUs) as well as related materials and parts. Inventory control aims to ensure satisfying customer demand at a determined service level. Reviewing the past literature shows that, by assuming a deterministic demand, most studies seek to minimize the sum of expected ordering and inventory carrying costs (Goltsos et al., 2022). However, in reality, due to the volatility of markets, customer demand could become highly uncertain (Hançerlioğulları et al., 2016). Companies are required to establish forecasts of future demand in order to foresee capabilities to satisfy an uncertain demand and optimize costs of the supporting inventory system. In addition, due to the current lead time between the ordering point and the delivery, which necessitates batch-ordering, companies need to order items ahead of demand.

Nowadays, enterprises benefit from the availability of big data and the use of predictive analytics with a growing number of applications reported in the literature (Bradlow et al., 2017). Although achieving accurate demand prediction has been challenging in the presence of market uncertainties, the availability of a large volume of historical data and the use of big data analytics helped improve the accuracy of demand forecasting (Wang et al., 2021). Such forecasts contribute to improving customer service and reducing costs resulted from supply-demand mismatches (Ge et al., 2019). Despite such an aim, however, there is no clear path from data to a good decision, and as such, failure in determining appropriate forecasting methods can lead to suboptimal solutions. In inventory optimization, to take advantage of growing availability of data, appropriate methods and algorithms shall be adopted for a data-driven inventory management. This study examines how big data can be utilized to handle demand uncertainty while optimizing supply chain cost.

A wide range of research has been conducted on developing effective methods for demand forecasting (Seyedan & Mafakheri, 2020). Machine learning algorithms, such as K-nearest neighbor, Gaussian naive Bayes, and decision trees, are used to establish forecasts of future demand based on historical patterns of time-series data (Varghese et al., 2022). Deep learning approaches are also gaining significant attention, including RNN, LSTM, gated recurrent units (GRU), autoencoders, and convolutional neural networks (CNN). These approaches achieve higher accuracy in predictions (Ju et al., 2018). However, such deep-learning models cannot

maintain high forecasting accuracy and robustness in dealing with applications that are subject to dynamic environments.

Ensemble learning is emerging as an approach that can address the above-mentioned challenge by benefiting from combining several models, and benefit from their collective superiorities, to improve forecasting performance . Zhou et al., (2002) have demonstrated that combining partial basic predictors can yield comparable or even superior generalization performance compared to combining all predictors simultaneously. These findings underscore the importance of adopting effective ensemble forecasting methods to achieve more accurate demand forecasting and inventory management solutions. Through the aggregation of diverse predictions, ensemble learning can mitigate individual model biases and capture a wider range of patterns, ultimately enhancing the reliability and effectiveness of demand forecasting and inventory management strategies. Reviewing the literature, as reported in the next section, reveals a scarcity of research in utilization of ensemble deep-learning methods for demand forecasting in supply chain management and inventory optimization stocks (Varghese et al., 2022). Moreover, the majority of ensemble deep learning models primarily concentrate on ensembling homogeneous deep neural networks (DNNs), while disregarding the benefits offered by heterogeneous DNNs due to their intricate architecture. This has created a compromise between forecast accuracy and model complexity, as noted by (Zhang et al., 2021).

This study advocates using heterogeneous DNNs in inventory optimization. In this sense, prediction algorithms of MLP CNN, and LSTM are investigated and combined in forming an ensemble learning approach that enhances the forecasting accuracy of multivariate time series data used for safety stock optimization achieving minimal total inventory cost. This research builds upon recent progress on DL-based forecasting approaches that 1) highlight the superior efficiency of LSTM over RNN in handling vanishing gradient issues (Du et al., 2023), 2) superiority of LSTM and CNN in time-series and spatial features modeling, respectively, and 3) MLP capability in extracting global features (Barrow & Crone, 2016). Furthermore, the proposed ensemble approach leverages distinct features extracted from two real-world supply chain time series datasets by employing these different types of DNNs.

The rest of the paper is organized as follows: Section 04.2 presents the proposed framework to incorporate an ensemble deep learning approach in demand forecasting for inventory optimization.

Section 4.3 presents the results. Discussions and further analysis are presented in Section 4.4. Finally, paper concludes in Section 4.5, providing a summary of the proposed approach, key assumptions and limitations as well as directions for future research.

4.2. Methodology

A typical data-driven inventory optimization process is presented in Figure 4-1. First, it is to decide how to handle demand uncertainty. Then, demand distribution is formulated as the outcome of the forecasting component. In doing so, demand is forecasted by developing an ensemble deep learning framework. These predictions will serve as an input for the inventory optimization component with a number of decision variables being adjusted to optimize the total inventory cost.

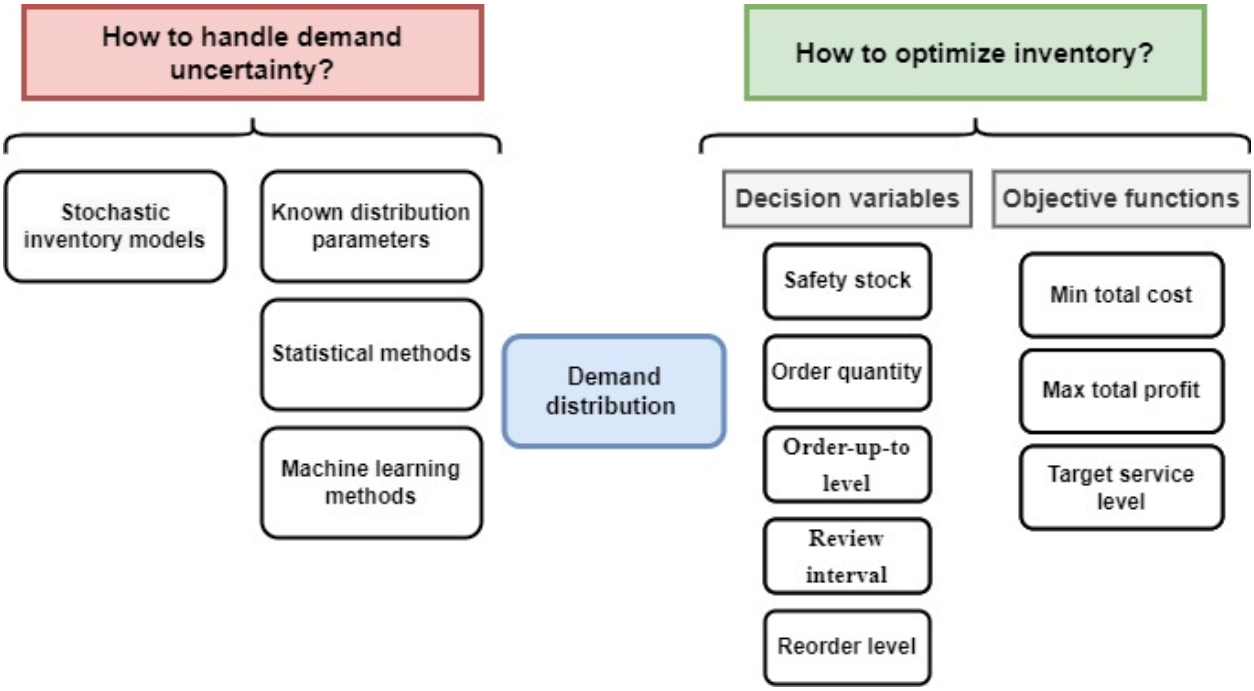


Figure 4-1. A schematic of data-driven inventory optimization process.

In doing so, this study contributes to the body of literature by considering ensemble deep learning approaches to predict daily demand and optimizes the stock levels for the case of online retailers. Additionally, in terms of the prediction algorithm, this study advocates addressing a research gap identified in the literature review in having very limited research on using heterogeneous deep neural networks (DNNs) in inventory optimization. The rationale behind the proposed methodology will be further discussed below.

This section aims at proposing an ensemble deep-learning approach to improve multivariate time series forecasting performance. Currently, most ensemble deep learning models focus on the ensemble of homogeneous DNNs and neglect the advantages of heterogeneous DNNs due to their rather complex architecture; reflecting a trade-off between forecast accuracy and model complexity (Zhang et al., 2021). For example, LSTM and CNN have an advantage in extracting time and space-related features from data, respectively, compared to MLP that can only extract global features from data (Barrow & Crone, 2016). In this sense, a comparative investigation of MLP, CNN, and LSTM methods (forming the ensemble model) is advocated by leveraging the different features extracted using different types of DNNs.

There are several assumptions in this study. First, the safety stock evaluation will be done through an OUTL policy that operates under a CSL objective. In the prediction section, it is assumed that no unprecedented events will happen in the future (e.g., pandemics, global recession, international shipping problems, crashes in shopping platforms). Furthermore, demand is predicted daily and is incorporated to estimate future order quantity, safety stock, and all related inventory scheduling. In the following subsections, methods adopted in constructing and combining the basic predictors will be described in detail. We further investigate how demand predictions can affect the total cost of inventory.

4.2.1. Demand forecasting

Figure 4-1 Figure 4-2 presents the proposed ensemble deep learning approach for demand time series forecasting. The aim is to establish forecasts of daily demand using an ensemble deep-learning neural network and then calculate safety stock and order quantity to minimize the total cost. Data preparation, base model construction (MLP, LSTM, and 1D-CNN), and meta-learner combination are the three stages of the proposed approach.

In the first step, data is split into the following three sections (Tan, 2021): (a) holding the latest 10% of the data as the holdout test set, (b) splitting the remaining 90% into an earlier grid-search cross-validation training set (2/3*90%), and (c) a later metamodel training set (1/3*90%). Figure 4-3 shows such split data for training and testing. Data preparation involves partitioning the original time series data sets using cross-validation. Cross-validation determines the depth and number of nodes in each hidden layer, as is the best practice when optimizing hyperparameters (Clausen & Li, 2022).

In order to guarantee the precision of the suggested demand forecasting, a robust process is implemented that involves outlier detection and interpolation for the collected test data. To achieve this, an unsupervised Support Vector Machine (SVM) is utilized. Firstly, the data is appropriately scaled to ensure compatibility with the SVM algorithm. Subsequently, the OneClassSVM technique is employed to map the data into two labels: 0 for normal data points and 1 for abnormal data points (Pedregosa et al., 2011). This classification enables the identification of outliers within the dataset. Once the outliers have been detected, a strategy of replacing these anomalous values with more accurate estimations is implemented. Specifically, linear interpolation is employed to estimate the missing or erroneous values caused by the outliers. By leveraging this interpolation technique, the gaps resulting from the outliers can be managed, ensuring a more complete and reliable dataset for demand forecasting purposes.

In the second step, the feature extraction process involves utilizing three different deep learning models: MLP, LSTM, and 1D-CNN. These models run in parallel to extract informative features from the input time-series data. Each model specializes in capturing different aspects of data.
The first model, MLP, is a feedforward neural network with fully connected layers. It operates on (input) time series data and leverages its ability to learn complex relationships between the input features and the target variable. The MLP extracts feature from a global perspective using a nonlinear activation function (Bishop, 1995; Hornik, 1991):

$$h_i = \varphi[W_i \times h_{i-1} + b_i] \tag{4-1}$$

Where:
h_i is the output of the i-th layer.
φ represents the activation function applied element-wise.
W_i is the weight matrix of the i-th layer.
h_{i-1} is the input to the i-th layer.
b_i is the bias vector of the i-th layer.

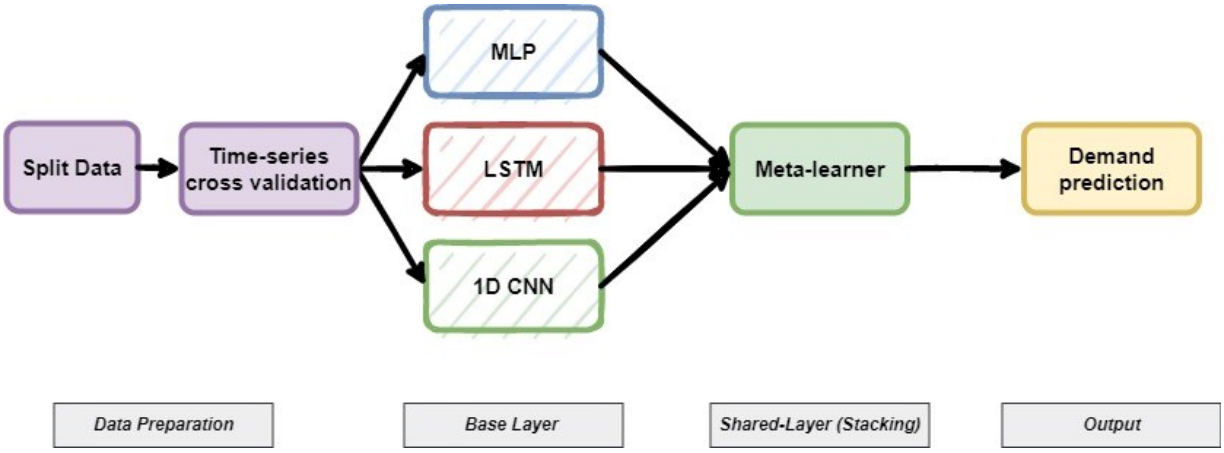


Figure 4-2. Proposed time-series demand forecasting approach using ensemble deep learning.

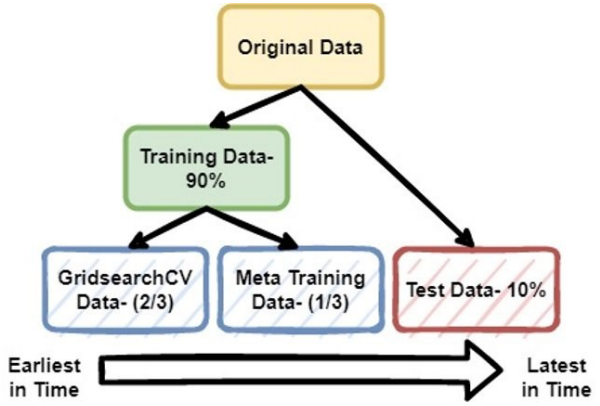


Figure 4-3. Data pipeline for different stages of ensemble methodology (Tan, 2021).

The second model, LSTM, utilizes recurrent neural networks with memory units and loops. This architecture allows the LSTM to capture long-term dependencies and retain information over time. It consists of input gates, forget gates, internal states, and output gates, which help in controlling the flow of information and preserving relevant temporal patterns (Abbasimehr et al., 2020; Hochreiter & Schmidhuber, 1997).

$$a(t_i) = \sigma[w_a x(t_i) + w_{ha} h(t_{i-1}) + b_a)] \quad (4\text{-}2)$$

$$f(t_i) = \sigma\big[w_f x(t_i) + w_{hf} h(t_{i-1}) + b_f\big)] \quad (4\text{-}3)$$

$$c(t_i) = f_t \times c(t_{i-1}) + a_t \times tanh(w_c.x(t_i)) + w_{hc}[h(t_{i-1}) + b_c)] \quad (4\text{-}4)$$

$$o(t_i) = \sigma[w_o x(t_i) + w_{ho} h(t_{i-1}) + b_o] \quad (4\text{-}5)$$

$$h(t_i) = o(t_i) \times tanh(c(t_i)) \quad (4\text{-}6)$$

Where:

$x(t_i)$ is the input value at time step i.

$h(t_{i-1})$ & $h(t_i)$ are the output values at time steps (i-1) and i, respectively.

$c(t_{i-1})$ & $c(t_i)$ are the cell states at time steps (i-1) and i, respectively.

b_a, b_f, b_c & b_o are the biases of the input gate, forget gate, internal state, and output gate, respectively.

w_a, w_f, w_c & w_o are the weights of the input gate, forget gate, internal state, and output gate, respectively.

w_{ha}, w_{hf}, w_{hc} & w_{ho} are the recurrent weights.

$a(t_i), f(t_i), c(t_i)$ & $o(t_i)$ are the output results for the input gate, forget gate, internal state, and output gate, respectively.

σ represents the sigmoid activation function.

$tanh$ represents the hyperbolic tangent activation function.

The third model, 1D-CNN, applies convolutional operations along the temporal dimension of the time series data. By sliding filters over the data, the 1D-CNN captures local patterns and extracts relevant features. Activation functions introduce non-linearity into the network, enabling it to learn complex representations (Kim, 2014).

$$h_i = \varphi[W \times x_{i:i+k-1} + b)] \quad (4\text{-}7)$$

Where:

h_i is the output at position i.

φ represents the activation function applied element-wise.

W is the weight matrix of the i-th layer.

$x_{i:i+k-1}$ is the input to the i-th layer.

b is the bias vector.

The above models run simultaneously, processing the input data and extracting different sets of features. In the training phase, a grid-search approach can be applied to find the optimal hyperparameters for each of the base models using a subset of the training data. Once the optimal hyperparameters are determined, the base models are trained on the full grid-search training data. The predictions obtained from the base models on the meta-training set form the explanatory variables for training the metamodel. The base models are essentially stacked or integrated, where their predictions serve as input to the shared layer. This step results in the creation of more information-rich features due to the dense structure of the shared layer (Mohammed & Kora, 2023).

The metamodel, which is an MLP, is then constructed based on the predictions generated by the base models using a holdout test set. These predictions are fed into the metamodel, and a nonlinear activation function is applied to produce the final demand forecasts.

Mathematically, the ensemble process can be represented as follows:

Base Model 1 (MLP): $f_1(X) = MLP(X; \theta_1)$

Base Model 2 (LSTM): $f_2(X) = LSTM(X; \theta_2)$

Base Model 3 (1D-CNN): $f_3(X) = 1D - CNN(X; \theta_3)$

Meta Model (MLP): $Meta(X) = MLP([f_1(X), f_2(X), f_3(X)]; \theta_{meta})$

Ensemble Prediction : Ensemble(X) = Meta(X)

Where X represents the input time series data, θ_1, θ_2, and θ_3 are the parameters of the base models, and θ_{meta} represents the parameters of the metamodel.

The choice of the above-mentioned three models for ensemble is based on their complementary strengths. MLP excels in capturing global patterns and complex relationships, LSTM is proficient in modelling temporal dependencies, and 1D-CNN is effective in capturing local patterns. By combining their individual strengths, the ensemble can benefit from a broader range of features and a more comprehensive understanding of the time series data. The ensemble process allows for the nonlinear mapping of the outputs of the base models using the metamodel. The ensembling helps reduce bias, leveraging diverse model architectures, and improving the overall predictive performance by combining the knowledge and insights from the base models.

In summary, the ensemble deep learning model combines the predictions of the base models (MLP, LSTM, and 1D-CNN) through a metamodel (MLP) to generate demand forecasts. The chosen

models provide complementary capabilities in capturing different aspects of the time series data, leading to a more comprehensive representation and improved predictive accuracy.

4.2.2. Inventory optimization

Most comparative demand forecasting studies focus on comparing accuracy of various methods rather than comparing their implications for inventory control (Babai et al., 2020). In this study a dynamic multiple-period inventory management problem is examined by adopting an OUTL inventory policy. We assume a nonperishable product with a starting inventory each period. Inventory replenishment may occur at each period. The decision variable will be the number of units that are to be ordered each time subject to demand variations. To incorporate such uncertainty, an inventory system with stochastic demand is formulated. An OUTL policy is used to control the stock based on the forecasted demand resulting from the proposed approach in Section 4.2.1.

Figure 4-4 presents the details of an OUTL policy with a variable order quantity (q) placed at a fixed time period(R). To reach the desired quantity(S), a number of units as inventory are needed and ordered subject to a lead time(L). This policy avoids accumulating excessive inventories and therefore could contribute to opportunity costs (i.e. resulting from unmet demand). In case of higher average inventories, the OUTL policy could lead to relatively higher capital commitments including holding costs (Lowalekar et al., 2016). In such a case, large quantities could only be acquired by placing several orders leading to higher ordering costs. In addition, shortages could happen if there are long delays in fulfilling the target inventory. As such, companies with cycled replenishment are recommended to use OUTL policies (Ivanov et al., 2019).

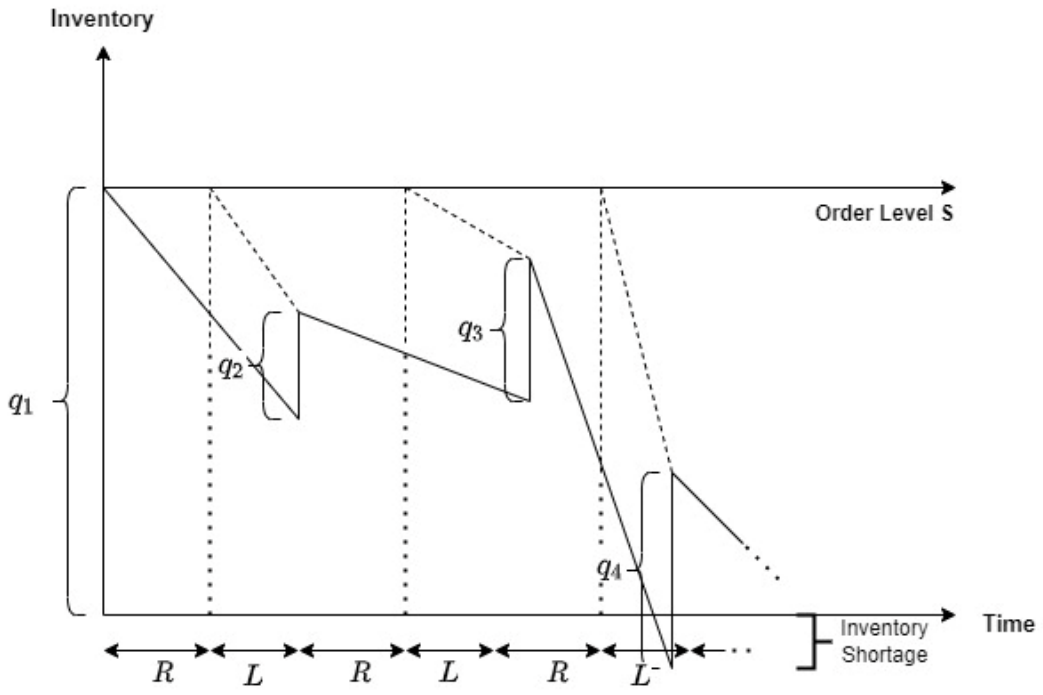


Figure 4-4 OULP – Order-up-to level policy (Ivanov et al., 2019).

In the realm of inventory control, one of the most widely used methods for determining the order quantity is the economic order quantity (EOQ) model. The proposed model provides a framework for estimating the optimal order quantity minimizing the total inventory costs. Traditionally, an EOQ model assumes a constant demand rate. However, in the context of demand forecasting, it is more realistic to replace the constant demand with an average forecasted demand. By incorporating the average forecasted demand into EOQ, businesses can better align their order quantities in line with expected demand variations, thereby avoiding excess inventory or stockouts.

When implementing EOQ models, items within the same inventory control group are usually considered to have similar ordering costs. This grouping allows businesses to streamline their procurement processes and benefit from economies of scale. By consolidating the ordering process for similar items, companies can achieve cost savings and operational efficiency. In contrast, the holding cost component in EOQ models is primarily calculated as a percentage of the value of items. This approach takes into account the carrying costs associated with storing and maintaining inventory over time. Holding costs include expenses such as warehousing, insurance, obsolescence, and the opportunity cost of tying up capital in inventory. By considering the item's

value as a factor in calculating the holding cost, businesses can reflect the financial implications of carrying inventory.

As a rule of thumb, the carrying charge, which represents the sum of capital and holding costs, is commonly estimated to range from 10% to 15% of the total annual costs (Ivanov et al., 2019). This percentage serves as a guideline for businesses to evaluate the overall cost impact of carrying inventory and make informed decisions regarding inventory control strategies. By understanding the carrying charge, companies can assess the trade-offs between holding costs and ordering costs and optimize their inventory management practices accordingly. In doing so, safety stock (ss), reorder point (r) and replenishment level (S) are defined as follows:

$$ss = z.\sigma\sqrt{(R+L)} \tag{4-8}$$

$$r = d(R+L) + z.\sigma\sqrt{(R+L)} \tag{4-9}$$

$$S = r + q \tag{4-10}$$

Estimating the distribution of lead time also plays a crucial role in determining the appropriate reorder and order-up-to levels in inventory management. The lead time refers to the time interval between placing an order and receiving it. By understanding the statistical characteristics of lead time, businesses can make informed decisions about inventory replenishment to ensure a smooth flow of goods. To assess the impact of lead time on inventory control, the mean and variance of the demand distribution need to be estimated. These estimations provide valuable insights into the expected demand levels and the associated uncertainty. Equations (1) and () presented by Babai et al. (2020) outline the methodology for these estimations:

$$\mu_{L+1,t} = (L+1) * \widehat{D}_t \tag{4-11}$$

$$\sigma^2_{L+1,t} = (L+1) * MSE_t. \tag{4-12}$$

To improve the accuracy of demand forecasting, this study proposes utilization of an ensemble deep-learning forecasting approach, as discussed in Section 4.2.1. This approach combines multiple deep-learning models to generate a more robust and reliable demand forecast. In this context, $\widehat{D}_t$ represents the estimated demand forecast at period t.

In order to assess the accuracy of the demand forecasts and estimate the variance of the mean demand per period, this study uses a smoothed mean squared error (MSE_t), which is derived using equation (4-13) as proposed by (Babai et al., 2020). This calculation provides a measure of the accuracy and reliability of demand forecasts by considering the squared differences between the forecasted and actual demand values.

$$MSE_t = \alpha(D_t - \hat{S}_t)^2 + (1 - \alpha)MSE_{t-1}. \quad (4\text{-}13)$$

However, determining the lead-time distribution presents difficulties, particularly when stochastic variations occur. To overcome this challenge, this study assumes fixed lead times when facing uncertain or varying lead-time durations. This simplification could help address the complexity of lead-time variations.

In the following section, the results and analyses related to the proposed ensemble deep learning forecasting approach and its integration into the inventory control system will be discussed. These results will shed light on performance and effectiveness of the approach in improving demand forecasting accuracy and optimizing inventory management decisions. By leveraging the ensemble deep learning forecasting approach and addressing the challenges associated with lead-time distribution, the proposed methodology aims to enhance the accuracy of demand forecasting and provide more reliable inputs for inventory control decisions. The subsequent section will delve into the empirical findings and provide insights into practical implications of the proposed approach.

4.3. Results Analysis

To analyze accuracy, stability, and generalization ability of the proposed ensemble deep learning approach, two types of products and three widely used evaluation metrics are used to conduct comparison experiments. Same experimental parameters are used for each model to compare their generalization performance on different data sets. Detailed descriptions of data sets, evaluation metrics, and experimental results and analyses are presented in the following subsections.

4.3.1. Data description & performance criteria

Two real-world supply chain time series data (with records from January 2015 to September 2017) of electronic and sports supplies are used in this study to show the effectiveness and practicality of the proposed model (Constante et al., 2019). There are 52 features in the data set, but many of

these features are not linked to or can contribute to demand forecasting. The daily demand for all products varied from 250 to 500 depending on time, sales price, and discounts. Forecasts for the target variable are correlated to a small set of these features including sales price, holidays and other timing effects, as well as past demand.

Figure 4-5 and Figure 4-6 depict specific subsets of data, focusing on Sports and Electronics Products, respectively. These figures provide an insight into the observed patterns within the data, revealing numerous fluctuations characterized by considerable uncertainties associated with seasonality and cyclic behavior. Given the inherent volatility and irregularities observed in the ups and downs of the dataset, traditional predictive approaches may struggle to capture the complexities and intricacies effectively. Ensemble learning, on the other hand, has the potential to handle such challenges by aggregating the predictions from multiple models and capturing diverse patterns and trends within data.

Based on mean absolute percentage error (MAPE), mean absolute error (MAE), and root mean square error (RMSE), the predictions were compared in terms of accuracy. Consequently, a lower RMSE value indicates better predictability (or less inaccuracy). These criteria are calculated as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{4-14}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{4-15}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{4-16}$$

where y_i and $\hat{y}_i$ show the actual and predicted values at day *i*, respectively. In this case, *n* represents the number of days in a testing period (82 days in this example).

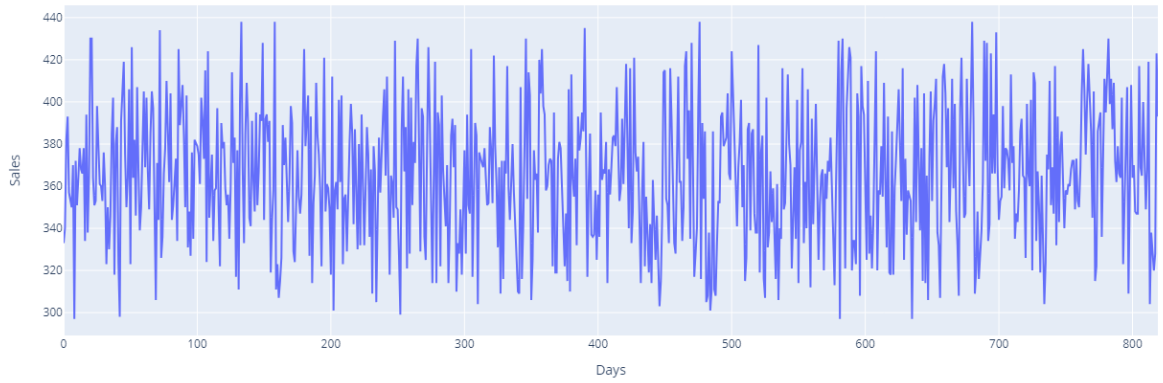


Figure 4-5 Daily Sports Products Sales

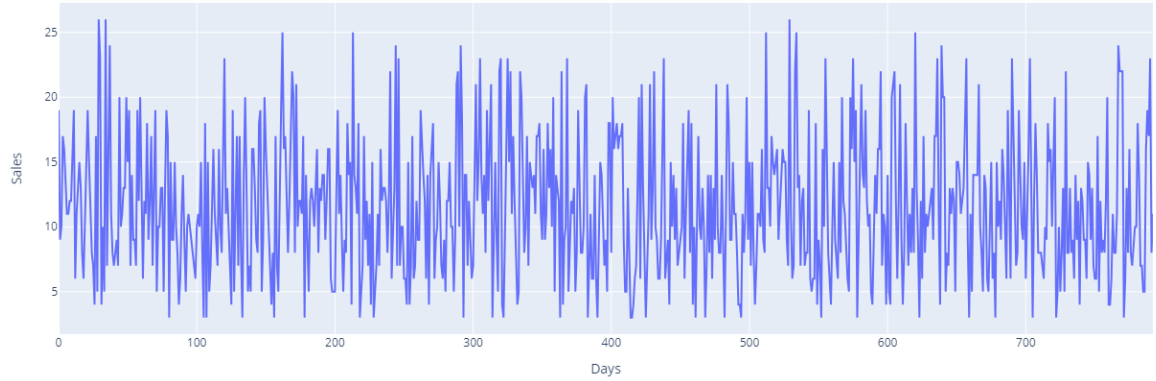


Figure 4-6 Daily Electronics Products Sales

Table 4.1 Cross-validation scores and statistics for sports products.

	MLP			LSTM			1D-CNN			Proposed model		
	MAE	**RMSE**	**MAPE%**	**MAE**	**RMSE**	**MAPE%**	**MAE**	**RMSE**	**MAPE%**	**MAE**	**RMSE**	**MAPE%**
1st fold	36.43	45.50	9.66	24.99	30.77	6.83	35.42	42.46	10.69	23.24	23.24	6.28
2nd fold	54.40	66.26	15.64	34.03	40.04	9.14	34.52	42.06	9.60	30.74	30.74	8.32
3rd fold	50.85	61.67	16.08	29.82	36.01	8.17	28.90	36.79	8.24	28.72	28.72	7.90
4th fold	118.78	131.70	13.17	31.85	36.85	8.66	38.33	47.49	11.56	30.14	30.14	8.41
5th fold	125.90	138.37	13.03	31.79	36.93	8.89	43.12	52.45	12.53	28.75	28.75	8.08
Mean	77.27	88.70	13.52	30.49	36.12	8.34	36.06	44.25	10.52	28.32	28.32	7.80
Median	54.40	66.26	13.17	31.79	36.85	8.66	35.42	42.46	10.69	28.75	28.75	8.08
Standard Deviation	41.76	43.06	2.56	3.41	3.35	0.91	5.22	5.94	1.67	2.97	2.97	0.87

Table 4.2. Cross-validation scores & statistics for electronic products.

	MLP			LSTM			1D-CNN			Proposed model		
	MAE	**RMSE**	**MAPE%**	**MAE**	**RMSE**	**MAPE%**	**MAE**	**RMSE**	**MAPE%**	**MAE**	**RMSE**	**MAPE%**
1st fold	5.19	6.18	14.63	4.93	42.03	12.03	4.72	5.77	13.55	5.58	6.38	12.65
2nd fold	3.71	4.71	13.66	4.03	35.53	13.53	4.15	4.94	13.07	4.82	5.61	13.86
3rd fold	4.81	6.10	16.96	4.59	39.78	11.78	4.51	5.60	12.95	4.96	5.90	12.15
4th fold	5.20	6.68	14.07	4.87	42.51	13.51	5.13	6.62	14.18	5.03	6.29	11.12
5th fold	4.40	5.51	15.84	4.03	33.36	14.36	3.80	4.78	11.20	3.87	4.83	13.50
Mean	4.66	5.84	15.03	4.49	38.64	13.04	4.46	5.54	12.99	4.84	5.80	12.65
Median	4.81	6.10	14.04	4.59	39.78	13.51	4.51	5.60	13.07	4.90	5.90	12.66
Standard Deviation	0.62	0.75	1.35	0.43	4.04	1.10	0.51	0.73	1.11	0.61	0.62	1.09

4.3.1. Demand forecasting

To determine the score of holdout test set, a 5-fold cross-validation is performed using modified post-gridsearch hyperparameter settings. During the cross-validation, both batches (base + meta) of the training set are combined to reconstruct the full 90% of training data. Table 4.1 and Table 4.2 present cross-validation scores for three base models and the proposed ensemble deep learning model for sports and electronics products, respectively.

The base models are only trained once using the gridsearch training set to predict the target variable's values on the holdout test set. Metamodels are built from predictions made on meta-training sets, which are used to decide what to predict on test sets based on the meta-training model. Figure 4-7 and Figure 4-8 are forecasted demand versus actual demand in the sports and electronic data sets, respectively. These figures show that the predicted demand and the actual demand have close proximity confirming a good accuracy for predictions. It is shown that the proposed model achieved the lowest MAE, RMSE, and MAPE% scores on the holdout test set.

Moreover, compared to the LSTM (which has the best performance among the three base models presented in Table 4.3 and Table 4.4), the proposed ensemble deep learning model achieved an improvement of 22% in the RMSE, 21.7% in the MAE, and 22.3% in the MAPE% and demonstrated improvement in sports products. In other words, the proposed ensemble methodology generates more accurate results than individual base models in the ensemble. Table

4.3 and Table 4.4 provide details related to the above-mentioned forecasting performance criteria for sports and electronics data sets, respectively.

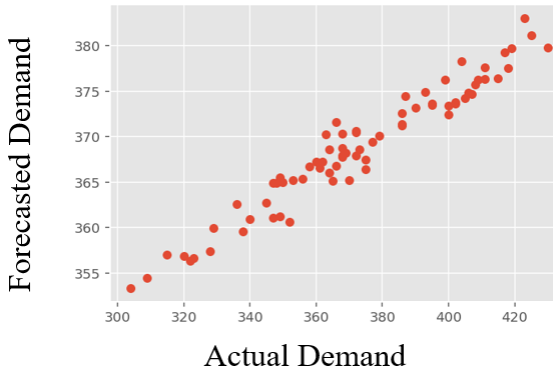


Figure 4-7 Forecasted vs. actual sport product demand.

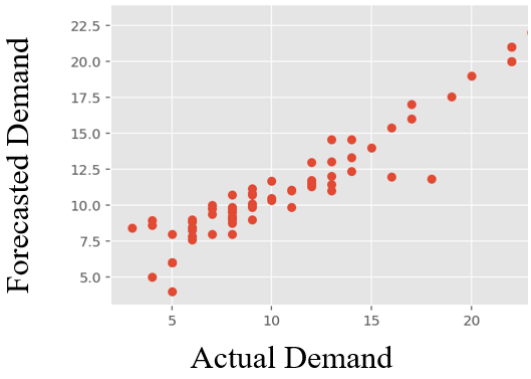


Figure 4-8 Forecasted vs. actual *electronic product* demand.

Table 4.3 Test results using holdout data set for sports product.

	MAE	**RMSE**	**MAPE%**
MLP	*26.26*	*32.25*	*7.29*
LSTM	24.85	30.78	6.72
1D-CNN	*25.24*	*31.06*	*6.78*
Proposed model	**19.46**	**24.02**	**5.22**

Table 4.4 Test results using holdout data set *for electronic product.*

	MAE	**RMSE**	**MAPE%**
MLP	*5.05*	*5.93*	*12.64*
LSTM	4.18	5.21	10.12
1D-CNN	*6.20*	*7.96*	*14.76*
Proposed model	**3.56**	**4.99**	**9.58**

Additionally, the RMSE of the proposed model is within the range of the 5-fold cross-validation procedure, while the MAE is below it. Holdout test errors were generally lower than those observed in the cross-validations across all models, indicating that the test set is more predictable.

This emphasizes the need to conduct multifold cross-validation to evaluate model performance in such multivariate time-series forecasting.

4.3.2. Inventory Optimization

Total cost includes fixed cost, holding cost, and shortage cost which should be optimized to determine the optimum values of the order-up-to-level and review interval (Lowalekar et al., 2016). Sports and electronics products are assumed without a lifetime. A periodic review setting is considered, where reviews are conducted regularly. In OUTL policy, order-up-to level is *S* and the inventory position is determined after a review I. Then, an amount equal to the difference between the order-up-to level (*S*) and on-hand stock is ordered at the beginning of each review period. The order quantity is received instantly and reflected in the stock level after the review. All of these calculations are based on the distribution of forecasted demand in the review intervals.

Figure 4-9. Comparison of total cost resulted in using different demand forecasting methods for sportFigure 4-9 and Figure 4-10 represent the total cost in OUTL policy with different base forecasting methods (MLP, LSTM, and 1D-CNN) for sports products and electronic products, respectively. It is shown that the proposed ensemble deep learning model can reach a lower total cost compared to other forecasting methods in a review interval. In this online shopping case study, data shows that demand for sports products are higher than that for electronic products. Lower pricing could be contributed as one of the reasons for this higher daily demand.

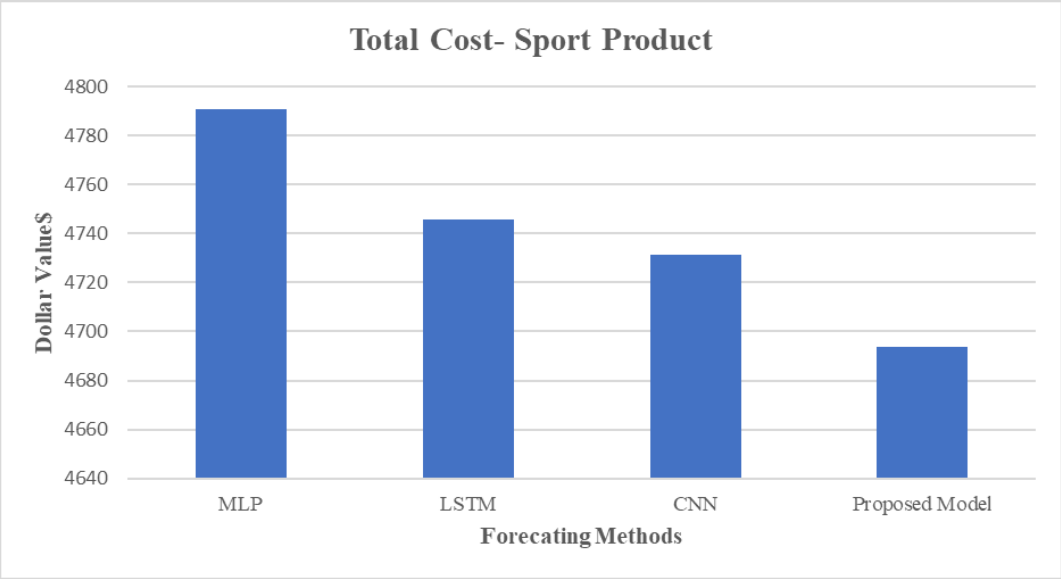


Figure 4-9. Comparison of total cost resulted in using different demand forecasting methods for sports products.

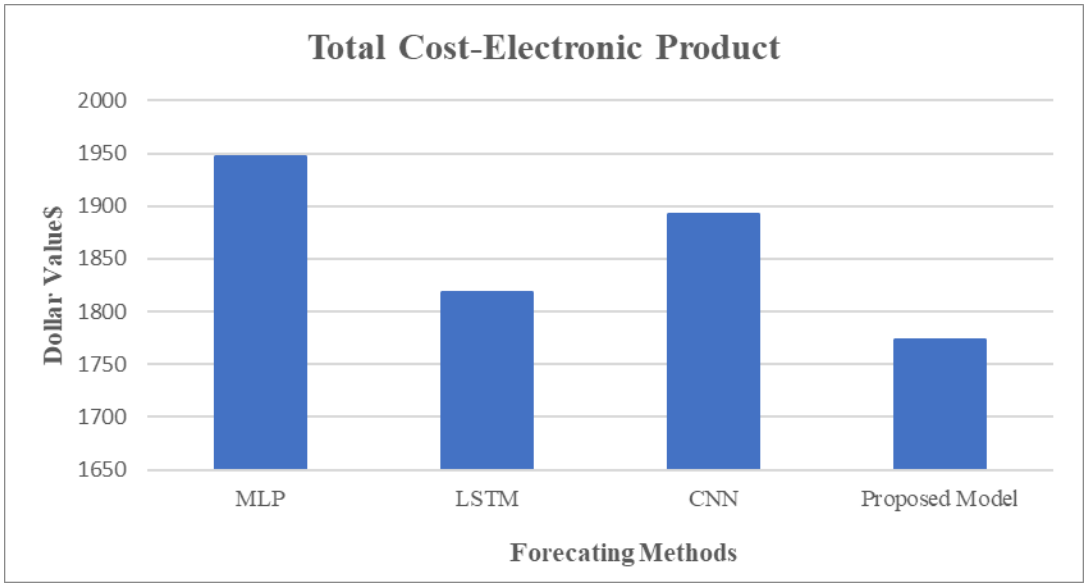


Figure 4-10. Comparison of total cost resulted in using different demand forecasting methods for electronic products.

4.4. Discussion

The proposed research utilized an ensemble deep learning approach, comprising three neural network base models (MLP, LSTM, and 1D-CNN), to predict demands for sports and electronic products for one period ahead. A meta-layer was incorporated, modelling a nonlinear relationship with the MLP model. The findings demonstrated that employing a machine learning ensemble stack approach for multivariate time series analysis resulted in more accurate demand predictions. Furthermore, by implementing an OUTL policy, the demand distribution was leveraged to optimize the overall cost.

The application of the proposed forecasting approach to an OUTL policy led to noticeable improvements in inventory performance metrics. The ensemble deep learning approach achieved a mean absolute percentage error (MAPE) of 5.22% and 9.58% for sports and electronic products, respectively, in predicting demand uncertainty. Consequently, when compared to its three individual base models, the ensemble learning approach exhibited enhanced forecasting accuracy.

In Table 4.5 and Table 4.6, the changes in order quantity, safety stock, and reorder points for the base models and the proposed ensemble deep learning model are presented. Notably, the ensemble deep-learning forecasting resulted in slightly higher safety stocks. Additionally, both order quantity and reorder points exhibited similar patterns across the datasets, with the proposed model yielding the lowest values compared to the individual base models. This optimization of safety stock led to a reduction in shortage costs and, consequently, a decrease in the total cost.

The results warrant a more detailed discussion to enhance their interpretation and significance. Firstly, in relation to literature, this study contributes to the existing knowledge by showcasing the effectiveness of an ensemble deep learning approach for demand forecasting in the context of sports and electronic products. Evidence of the improved forecasting accuracy was presented through comparing the proposed model with the individual base models. Moreover, the findings highlight the importance of considering an OUTL policy in inventory management. The optimization of safety stock based on demand distribution resulted in cost savings, demonstrating the practical value of leveraging demand insights in decision-making.

Table 4.5. Sensitivity analysis of inventory parameters in different forecasting methods for Sports products.

Forecasting method	Order Quantity	Safety Stock	Reorder Point
MLP	*623*	*136*	*2761*
LSTM	618	177	2753
CNN	*616*	*170*	*2731*
Proposed Model	**611**	**197**	**2717**

Table 4.6. Sensitivity analysis of inventory parameters in different forecasting methods for Electronics products.

Forecasting method	Order Quantity	Safety Stock	Reorder Point
MLP	*105*	*12*	*108*
LSTM	98	10	94
CNN	*102*	*12*	*103*
Proposed Model	**96**	**14**	**93**

Another analysis has been done to validate the model in both Sports and Electronic products. The following tables provide a sensitivity analysis of the total cost across different lead time scenarios, comparing the proposed model with MLP, LSTM, and CNN models. Table 4.7 focuses on sports products. Across all lead time scenarios of 7 days, 10 days, and 14 days, the proposed model consistently achieves lower total costs compared to the MLP, LSTM, and CNN models. Specifically, the proposed model demonstrates cost savings of 97 units, 105 units, and 115 units, respectively, compared to the MLP model. Table 4.8 examines electronic products. In this category, the proposed model also outperforms the other models in terms of cost across all lead time scenarios. Under a 7-day lead time, the proposed model achieves cost savings of 175 units compared to the MLP model. For a 10-day lead time, the savings increase to 90 units, and for a 14-day lead time, the savings amount to 98 units.

In summary, the sensitivity analysis highlights that the proposed model consistently delivers cost savings and outperforms the MLP, LSTM, and CNN models in both sports and electronic product categories. The savings range from approximately 90 to 175 units in electronic products and from 97 to 115 units in sports products, depending on the lead time scenario. These findings underscore

the effectiveness and competitiveness of the proposed model in optimizing costs across different lead-time situations. By adopting the proposed model, businesses in both sports and electronic industries have the opportunity to achieve significant cost savings and enhance their overall financial performance.

Table 4.7. Sensitivity analysis under different lead time situations- Sports products

		MLP	LSTM	CNN	Proposed Model
Total Cost	*7 Days Lead Time*	*4791*	*4746*	*4731*	*4694*
	10 Days Lead Time	**5207**	**5158**	**5143**	**5102**
	14 Days Lead Time	*5704*	*5651*	*5634*	*5589*

Table 4.8. Sensitivity analysis under different lead time situations- Electronic products

		MLP	LSTM	CNN	Proposed Model
Total Cost	*7 Days Lead Time*	*1948*	*1819*	*1893*	*1773*
	10 Days Lead Time	**998**	**931**	**970**	**908**
	14 Days Lead Time	*1093*	*1020*	*1062*	*995*

4.5. Conclusions

The application of multivariate time series forecasting holds great potential for enhancing the efficiency and robustness of inventory systems. In this particular study, a data mining approach was proposed, aiming to analyze supply chain data and extract valuable information for optimizing safety stock and improving inventory management. The study focused on two streams of real-world time series data representing Sports and Electronics products, employing three state-of-the-art deep learning models: Multilayer Perceptron (MLP), Long Short-Term Memory (LSTM), and 1D Convolutional Neural Network (1D-CNN). To demonstrate the superiority of the proposed ensemble forecasting approach, comparative experiments were conducted. The forecasting performance of each individual model was evaluated using commonly used statistical metrics. Additionally, the demand uncertainty, managed under an OUTL (Order-Up-To Level) policy, was addressed by adjusting safety stock and order quantity. Through these experiments, several noteworthy conclusions were drawn:

- The proposed approach successfully leveraged the strengths of both deep learning and ensemble learning techniques, allowing for the extraction of implicit features within the time series data.
- The ensemble model outperformed the baseline models in both the Sports and Electronics product domains, exhibiting superior forecasting accuracy, stability, and generalization capabilities.
- The ensemble deep learning method significantly improved the forecasting accuracy of the underlying base predictors.
- The OUTL policy, coupled with the optimized safety stock and order quantity, resulted in the minimum total cost.

While the proposed approach showcased promising results, there are certain limitations that could inspire future research directions. The construction and combination stages of the basic predictors were found to be complex and computationally demanding, requiring significant memory and computational resources. Exploring better strategies for building and pruning ensembles of basic predictors can enhance computational efficiency and scalability.

Moreover, there is room for further exploration and design of more effective ensemble schemes to enhance forecasting accuracy and stability. Researchers have already demonstrated the potential of such approaches in related studies (Abbasi et al., 2020; Abbasimehr et al., 2020). Additionally, it would be valuable to investigate the broader impacts of the proposed approach and the resulting improvements in demand forecasting accuracy on supply chain management. Furthermore, the proposed framework can be tested under different inventory policies, beyond the OUTL policy utilized in this study. Future work may also consider the optimization of inventory for perishable products incorporating constraints related to product life expectancy and associated spoilage costs.

Acknowledgement

The authors are grateful to the editor and three anonymous reviewers whose comments and suggestions were very helpful in improving the quality of this manuscript.

Chapter 5. Safety Stock Estimation Based on Forecasted Demand Distribution using Recurrent Mixture Density Network[1]

Mahya Seyedan[a], Fereshteh Mafakheri[b,2], Chun Wang[a]
[a] Concordia Institute for Information Systems Engineering, Concordia University, Montreal, Canada
[b]École nationale d'administration publique (ENAP), Université du Québec, Montréal, Canada

ABSTRACT – Accurate demand forecasting is critical for a reliable estimation of safety stock and proper inventory management. A significant amount of research has focused on demand forecasting aimed at estimating the average and variation of demand using point forecasting. However, demand distribution forecasting has received limited attention. Recently, research is emerging on the use of recurrent mixture density networks (RMDNs) to model complex and nonlinear patterns exhibited in time series when estimating data distribution. In this sense, this study is proposing a novel framework using RMDNs for demand forecasting and safety stock calculation. The performance of the proposed approach is compared to a number of well-known time series forecasting models. In addition, a comparative analysis, using case study data, showcases how retailers can effectively predict the likelihood of experiencing stockouts (or overstock situations) in order to adjust the estimates for the safety stock, reducing inventory management costs and enhancing customer satisfaction.

Keywords: Safety stock, demand forecasting, recurrent mixture density network, demand distribution, inventory management

[1] This paper was submitted.
[2] Corresponding author: fereshteh.mafakheri@enap.ca

5.1. Introduction

Safety stock is crucial in inventory management as it guards against unexpected fluctuations in demand, disruptions in supply chains, and variations in lead times. Accurate demand forecasting is critical to calculating safety stock levels to ensure that the right amount of inventory is available to meet customer demand. A company may lose sales, have unsatisfied customers, or even damage its reputation without sufficient safety stock on hand. Alternatively, excess safety stock increases carrying costs, reduces profitability, and ties up capital. It is therefore essential to strike a balance between safety stock and inventory levels so inventory management can be efficient and effective. Several researchers have found accurate demand forecasting is critical for the reliable estimation of safety stock and proper inventory management. Pacheco et al. (2017) found demand forecasts have a significant impact on safety stock levels. Likewise, Babai et al. (2011) demonstrated accurate demand forecasting can reduce safety stock levels and associated inventory costs.

Demand forecasting techniques have been improved over the past few decades by researchers to forecast demand more accurately using statistical models, machine-learning approaches, and deep-learning algorithms. A number of statistical models have been used to forecast demand, including auto-regressive integrated moving averages (ARIMAs) and exponential smoothing (Babai et al., 2013). However, use of machine-learning methods, such as random forests (van Steenbergen & Mes, 2020), support vector machines, and neural networks (Güven & Şimşir, 2020), has surged thanks to their ability to handle intricate data patterns and capture nonlinear relationships. Recently, deep-learning methods such as RNNs and their variants have demonstrated promising results in demand forecasting (Salinas et al., 2020).

Despite the extensive use of various machine-learning, deep-learning, and statistical approaches in inventory management, most of these methods have primarily been used for point forecasting (Babai et al., 2020). This narrow focus neglects the critical aspect of predicting the entire demand distribution, which is essential for robust safety stock estimation. Furthermore, many proposed forecasting applications have either disregarded the importance of incorporating uncertainty into demand forecasting (Rostami-Tabar & Rendon-Sanchez, 2021) or struggle to capture the nonlinear relationships among variables (Oroojlooyjadid et al., 2020).

Demand distribution plays a pivotal role in safety stock calculation, enabling companies to assess the probability of specific demand levels and adjust their safety stock accordingly. Disruptions in supply chains (Snyder et al., 2016) or unforeseen shifts in demand patterns (Tarim & Kingsman, 2006) can have significant implications for inventory management. Traditional inventory management often relies on the assumption that demand is Gaussian and iid (independently and identically distributed), which may not always align with real-world complexities. Recent studies have challenged this norm and introduced more sophisticated forecasting methods. (Trapero et al., 2019b) critiques the iid assumption and suggests kernel density estimation and GARCH (1,1) models for better safety stock calculation. Similarly, (Trapero et al., 2019a) highlight the limitations of Gaussian iid assumptions and propose a combination of empirical methods to enhance safety stock accuracy. These papers predominantly focused on methods for estimating safety stock and did not specifically explore the probabilistic aspects of demand forecasting. These studies represent a shift towards empirical, data-driven forecasting, emphasizing the need for methods that acknowledge and adapt to demand variability. This context sets the stage for the introduction of Recurrent Mixture Density Networks (RMDNs).

RMDNs, an extension of RNNs, RMDNs, with their ability to model complex patterns and various distribution types, address these identified gaps. This flexibility is crucial in accurately capturing the real-world complexities of demand data, which often exhibit non-linear patterns, multimodality, and other non-standard characteristics. By effectively modeling these diverse distributions, RMDNs provide a more nuanced and realistic forecast of demand, enhancing the accuracy of safety stock calculations. They offer an advanced solution for robust demand forecasting and safety stock calculation, aligning with the evolving needs of modern supply chain management. RMDNs have emerged as a promising approach for modeling probability distributions in time series data, moving beyond traditional point forecasts (Razavi et al., 2024; Schittenkopf et al., 2000). This development is highly relevant in inventory management, where precise demand forecasting is crucial for determining optimal safety stock levels (Huber et al., 2019). By leveraging insights from related domains, RMDNs offer valuable distribution forecasts that aid in safety stock calculations. These demand probability distributions that RMDNs provide empower managers to make well-informed decisions about safety stock levels and inventory ordering policies.

In this study, we propose a data-driven framework to assist retailers in reducing inventory management costs and enhancing customer satisfaction by effectively mitigating stockouts and overstocks. Central to this framework is the accurate prediction of demand, which is crucial for calculating safety stock. To address these challenges, we estimate demand distribution based on historical demand data and utilize this information to determine optimal safety stock levels.

The rest of the paper is organized as follows: In Section 5.2, the proposed method of forecasting demand distributions and calculating safety stock is presented in details. Sections 5.3 and 5.4 present results and discussions, respectively. We then conclude by summarizing the research approach and suggesting directions for future research in Section 5.5.

5.2. Methodology

A time-series probabilistic forecasting method, RMDNs, which incorporate LSTM units is proposed for sales demand prediction. The MDNs combine the power of neural networks with Gaussian mixture models (GMMs) to provide probabilistic forecasts. Whereas regular neural networks produce deterministic outcomes, MDNs capture the stochastic behavior of models, making them valuable when multiple predicted outcomes are needed. This feature allows easy integration with stochastic programming models.

In the proposed approach, MDNs are utilized to parameterize a Gaussian mixture distribution, generating the mixed coefficient (weight), mean, and variance for each Gaussian kernel. A GMM is a semiparametric approach that stores the predicted probability distribution (Schittenkopf et al., 2000). Given an input, X, the GMM can formulate a conditional density function, as shown in Equation 5-1:

$$p(y|X,\theta) = \sum_{i=1}^{K} \pi_i(X) \mathrm{N}_i(y|\mu_i(X), \sigma_i(X)) \tag{5-1}$$

Where θ represents a set of parameters including π, μ, and σ, and K denotes the number of Gaussian components in the model. Each Gaussian distribution (labeled i-th Gaussian) is determined by its weight, mean, and a covariance matrix, $\sum_i$ (i.e. variance if using a univariate Gaussian). The weight of each Gaussian component, π_i, is calculated using the softmax function (Equation 5-2), ensuring the sum of component weights equals 1:

$$\pi_i = softmax(h)_i = \frac{e^{h_i^{\pi}}}{\sum_{k=1}^{n} e^{h_k^{\pi}}} \tag{5-2}$$

The variables h_i^u, μ_i and σ_i^2 represent the outputs of the hidden layers preceding the GMM components, and the mean and variance values are obtained using Equations 5-3 and 5-4, respectively:

$$\mu_i = h_i^{\mu} \tag{5-3}$$

$$\sigma_i = \exp\left(h_i^{\sigma}\right) \tag{5-4}$$

Whereas traditional neural networks use mean squared error (MSE) as a deterministic loss function, MDNs employ the negative logarithm of the likelihood function (Equation 5-5) as their loss function. By minimizing this loss function and optimizing the neural network parameters represented by *w*, the GMM parameters can be calibrated using input data X and *w*:

$$E(w) = -\sum_{n=1}^{N} ln\left\{\sum_{k=1}^{K} \pi_k(X_n, w)\mathrm{N}(y|\mu_k(X_n, w), \sigma_k^2(X_n, w))\right\} \tag{5-5}$$

To enhance MDNs for time-series sales demand forecasting, a recurrent extension of them, an RMDN model, is proposed. By incorporating LSTM units, RMDNs can model sequential data, making it ideal for time-series forecasting tasks. LSTM units can retain important historical information and capture long-term dependencies, allowing RMDNs to better understand temporal patterns in the sales demand data.

The LSTM unit incorporates three essential gates: the input gate (i_t), the forget gate (f_t), and the output gate (o_t) (Bai et al., 2021). These gates control the flow of information through the cell state, allowing the LSTM to learn and retain relevant information while disregarding irrelevant or noisy inputs. The updated cell state ($\tilde{C}_t$) calculates the new candidate cell state, which, in combination with forget and input gates, determines the final cell state (C_t). Finally, the hidden state (h_t) is updated by applying the output gate to the updated cell state.

The LSTM model is expressed as:

i. Input Gate (i_t):

$$i_t = \sigma(W_{xi}.x_t + W_{hi}.h_{t-1} + b_i) \tag{5-6}$$

ii. Forget Gate (f_t):

$$f_t = \sigma(W_{xf}.x_t + W_{hf}.h_{t-1} + b_f) \quad (5\text{-}7)$$

iii. Updated Cell State ($\tilde{C}_t$):

$$\tilde{C}_t = tanh(W_{xc}.x_t + W_{hc}.h_{t-1} + b_c) \quad (5\text{-}8)$$

iv. Final Cell State (C_t):

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \quad (5\text{-}9)$$

v. Output Gate (o_t):

$$o_t = \sigma(W_{xo}.x_t + W_{ho}.h_{t-1} + b_o) \quad (5\text{-}10)$$

vi. Hidden State (h_t):

$$h_t = o_t \odot \tanh(C_t) \quad (5\text{-}11)$$

By incorporating LSTM units into an MDN architecture, RMDNs can effectively model the temporal dependencies in time-series sales demand data, enabling more accurate and robust probabilistic forecasts. Trained with historical sales data, RMDNs generate future sales demand forecasts, along with associated uncertainties, providing valuable insights for decision-making in inventory management, sales planning, and resource allocation. This approach can handle the complexities of time-series sales demand forecasting with probabilistic outcomes, empowering businesses to make informed decisions.

5.2.1. Forecasting Model Architecture

To forecast demand for each product, an RMDN model is employed. RMDNs are a deep-learning architecture that capitalize on the strengths of both RNNs and MDNs. RNNs, with their sequential memory capabilities, can capture temporal dependencies in the data, whereas MDNs enable the generation of probabilistic forecasts with multiple modes, making them ideal for handling uncertainties in demand patterns.

The RMDN model comprises three main components, as depicted in Figure 5-1:

1. LSTM Layer: At the forefront of the architecture, the input time series is fed into an LSTM layer with 128 units and a rectified linear unit (ReLU) activation function. The unites of LSTM layer is (2^n) that n can be optimized by grid search approach (It is discussed in section 5.3.6.1). The primary role of this LSTM layer is to capture the intricate temporal dependencies present in the time series data, enabling the model to comprehend long-term patterns and trends.

2. Mixture Density Layer: Following the LSTM layer, the output is passed through a dense layer with a softmax activation function, producing a one-hot encoding representing the mixture component used in the output distribution. This encoding is further fed into another dense layer with 50 units and a hyperbolic tangent activation function.

3. Output Layers: The output from the dense layer is then processed through three separate output layers, each dedicated to computing the mean, variance, and mixing coefficient for each Gaussian distribution. The parameters K and I determine the number of Gaussians representing the multimodal distribution and the number of input features in each model, respectively.

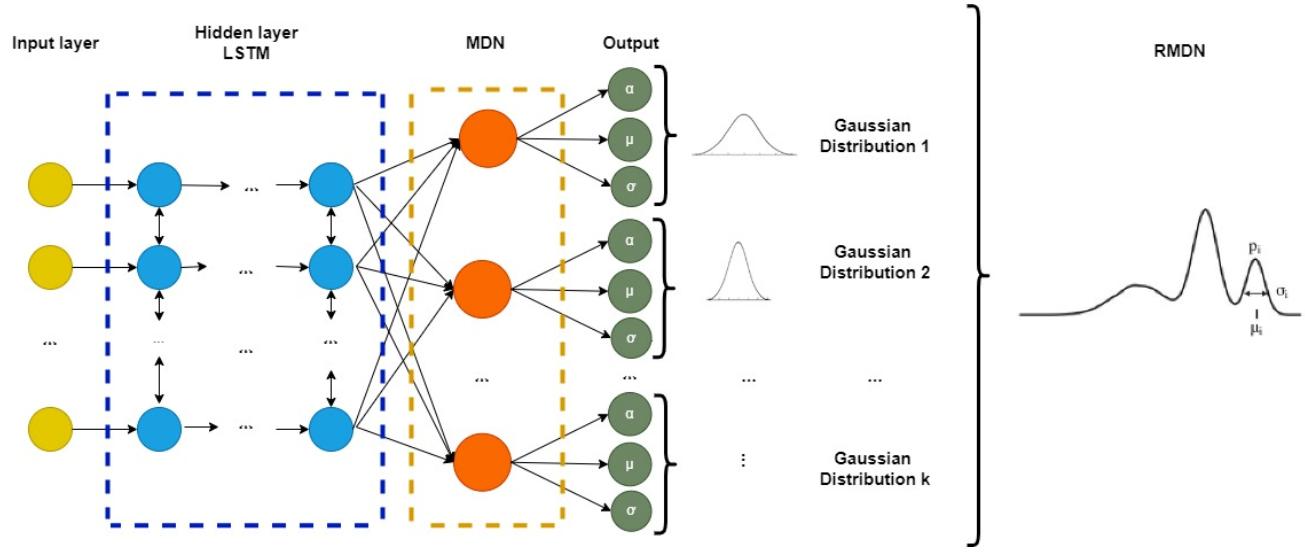


Figure 5-1 RMDN Structure

Through this architecture, our RMDN model efficiently combines the temporal understanding of RNNs with the probabilistic capabilities of MDNs, making it a powerful tool for generating reliable and comprehensive probabilistic forecasts.

5.2.1.1. Training and Validation

Before training the RMDN model, data is prepared by creating distinct training, validation, and test sets. The training set encompassed 70% of the preprocessed data, and the validation and test sets each comprised 15%. During the training phase, the RMDN model was fine-tuned using the Adam optimizer (Kingma & Ba, 2014) with a learning rate of 0.001 and a batch size of 64. It is found that the choice of the negative log-likelihood loss function was suitable for the nature of probabilistic forecasting tasks.

To ensure the model did not overfit the training data, regularization, a technique used to prevent overfitting in neural networks, is applied. Regularization can itself impact the training process: Too much regularization may lead to underfitting, where the model does not capture the underlying patterns in the data. Conversely, too little regularization might not prevent overfitting. It is thus a good idea to experiment with different levels of regularization and monitor the model's performance on both training and validation sets to find the right balance. In the context of LSTM and other deep-learning models, two regularization methods are primarily used: dropout and L1/ L2 regularization(Koivu et al., 2022; M. Yang et al., 2023).

- Dropout: Dropout is a regularization technique that involves dropping out (i.e., setting to zero) a random fraction of input units during training to prevent over-dependence on any particular neuron. This helps the model generalize better.
- L1/L2 Regularization: L1 and L2 are regularization techniques that add a penalty to the loss function. L1 regularization adds a penalty equal to the absolute value of the magnitude of coefficients, and L2 adds a penalty for the square of the magnitude of coefficients. You can adjust the regularization strength by changing the values in the "L1L2" function. Higher values mean stronger regularization.

5.2.1.2. Deployment

After a thorough evaluation, the RMDN model is deployed to generate daily demand forecasts. Automated forecasts play a pivotal role in informing the retailer's inventory management and purchasing decisions. Updated daily based on the latest sales data, the forecasts help optimize inventory levels, production planning, and resource allocation.

By employing our forecasting approach, demand can be accurately predicted for the next time step and several future time steps. This foresight empowers businesses to make informed decisions, proactively manage inventory, and allocate resources strategically. For example, if the forecast indicates an upcoming spike in demand a few months from now, businesses can plan ahead with additional production runs, raw material procurement, and resource allocation. The improved accuracy in demand forecasting helps adjust safety stock calculation, which is crucial for ensuring firms have sufficient inventory to meet unexpected spikes or disruptions in demand. With precise demand forecasting, businesses can adjust safety stock levels effectively, striking the right balance between stock availability and excess inventory.

In summary, with the powerful RMDN model at its core, the proposed methodology provides businesses with a data-driven approach to demand forecasting, enabling efficient inventory management strategies, reducing stockouts, and enhancing overall supply chain resilience. By leveraging the benefits of probabilistic forecasts, businesses can make confident decisions and navigate the complexities of demand fluctuations in today's dynamic marketplace.

5.2.2. Inventory Optimization

To optimize inventory management based on the demand forecasts generated by the RMDN model, an order-up-to level (OUTL) inventory policy (Ivanov et al., 2019) is adopted. When implementing the OUTL inventory policy, the primary goal is to replenish the stock to a predetermined inventory level. This makes the ability to forecast demand and its variability crucial for avoiding stockouts or overstocking. The RMDN model, with its provision for uncertainty, becomes particularly useful in this scenario. To implement this policy effectively, the weighted mean and variance of the lead time demand distribution from the RMDN outputs are used to set the OUTL. First, the weighted mean for lead time demand is calculated by utilizing the mixture weights (π_i) and means (μ_i) from the RMDN model. The overall weighted mean for the lead time demand can be computed as the following:

$$\mu_{LTD} = \sum_i \pi_i \mu_i \tag{5-12}$$

Then the weighted variance for lead time demand (σ^2_{LTD}) is calculated as:

$$\sigma^2_{LTD} = \sum_i \pi_i (\sigma_i^2 + \mu_i^2) - \mu^2_{LTD} \tag{5-13}$$

Here, σ_i^2 is the variance for each Gaussian component, which is the square of its standard deviation. Finally, the standard deviation for lead time demand is established as follows:

$$\sigma_{LTD} = \sqrt{\sigma^2_{LTD}} \tag{5-14}$$

By using μ_{LTD} and σ_{LTD} the OUTL is set, factoring in the desired service level and the variability in demand. This ensures avoiding reliance on a point estimate (as with traditional LSTM models), and instead, incorporating demand variability into our inventory decisions. To calculate the safety stock (SS) level, businesses must consider the variability in demand and lead time (L). Safety stock acts as a buffer to account for uncertainties and unexpected fluctuations in demand and lead time.

One commonly used approach to calculate safety stock is based on the service level and demand variability:

$$SS = z \times \sqrt{R + L} \times \sigma \tag{5-15}$$

Where L is the average lead time after placing an order until receiving the inventory, σ is the standard deviation of lead time and Z is the Z-score associated with the desired service level. A Z-score is calculated as the number of standard deviations from the mean needed to achieve a desired service level. An approximate 95% service level would be represented by a Z-score of 1.645, for instance.

In summary, the combination of a RMDNs demand forecasting model and an OUTL inventory policy offers a comprehensive data-driven approach to inventory optimization. By leveraging cutting-edge deep-learning techniques for demand forecasting, adopting an adaptive OUTL inventory policy, and calculating safety stock based on demand variability and lead time, businesses can achieve more efficient inventory management, reduced costs, and enhanced customer satisfaction through reliable product availability. This integrated approach paves the way for streamlined operations, improved decision-making, and greater overall competitiveness in today's dynamic and demanding market landscape. A summary of the methodology and experiments conducted in the study is illustrated in Figure 5-2.

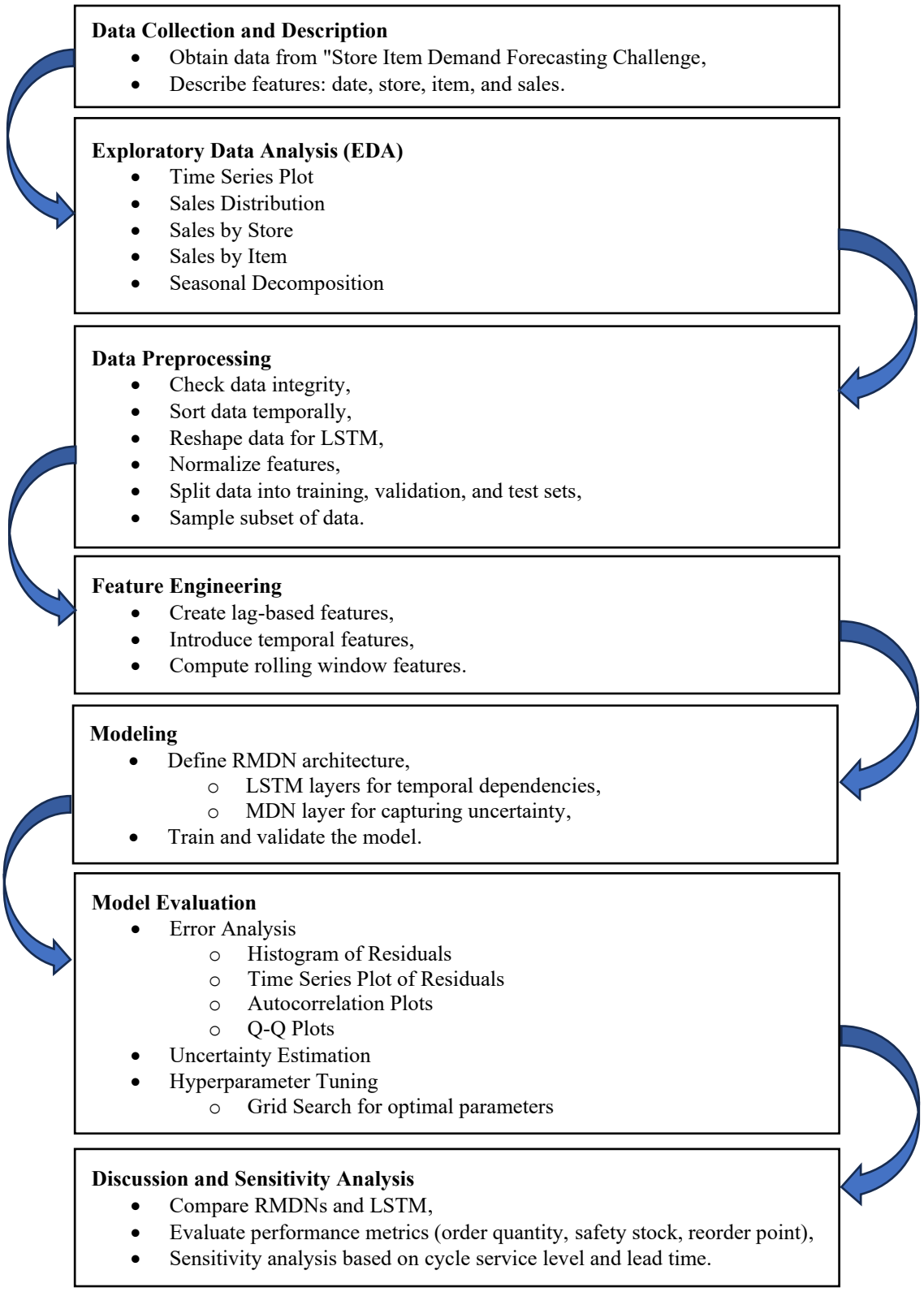


Figure 5-2 Summary of the proposed methodology and conducted experiments.

5.3. Results

After advocating the use of an RMDN model in demand forecasting, the results obtained by employing the proposed approach are presented in this section. The results will be followed by discussion on the accuracy measures, visualizations of predicted versus actual demand, and the effect of parameter tuning on model performance. Furthermore, the RMDN model's forecasting ability will be compared to alternative methods regarding calculating safety stock.

5.3.1. Data Description and Performance Criteria

A case study dataset from the Kaggle competition "Store Item Demand Forecasting Challenge" is explored containing historical sales data for 50 products sold at 10 different stores. The data span a period of approximately 5 years, from January 2013 to December 2017 (inversion, 2018). In the first step of the analysis, the focus is on data description, which was crucial for preparing the data set for RMDN modeling. The features used in the dataset includes *date*, which specifies the date of the sale; the *store*, indicating the ID of the store where the sale occurred; the *item*, which is the ID of the item sold; and finally, *sales*, representing the number of that particular item sold in the specified store on that date.

To gain a comprehensive understanding of the data set, various EDA techniques are employed. Below are the key analyses and the resulting insights.

5.3.1.1. Time Series Plot

To get an initial sense on how sales have evolved, a time series graph is plotted for a specific store–item combination, such as Store 1–Item 1 (Figure 5-3). This visualization reveals three important aspects: (1) Seasonality: The graph vividly displays a recurring pattern of peaks and valleys, indicating strong seasonality in the sales data; (2) Trend: Sales appear to be fairly stable over time, without any significant upward or downward shifts; and (3) Variability: Although there are fluctuations in sales, they seem to align well with the seasonal patterns already noted.

5.3.1.2. Sales Distribution

Understanding the distribution of daily sales is crucial for grasping its spread and central tendencies, to achieve this, a histogram plot is used (Figure 5-4). Observations from the histogram are as follows: (1) Common Sales Figures: Most daily sales cluster within a specific range, marking

the most frequently occurring sales values; (2) Skewness: The distribution exhibits a noticeable right skew, meaning there are fewer days with exceptionally high sales compared to the average.

5.3.1.3. Sales by Store

To evaluate the performance of individual stores, total sales are aggregated and visualized by store through a bar chart (Figure 5-5). Key observations include the following: (1) Uniform Performance: All stores display similar sales figures, suggesting a level of consistency across the board; (2) Minor Variations: Some stores have slight deviations in sales, but none stand out as either top or bottom performers relative to others.

5.3.1.4. Sales by Item

Furthermore, sales data are aggregated by item to identify which products drive the most revenue. Another bar chart serves this purpose (Figure 5-6). The bar chart revealed the following: (1) Consistent Item Performance: Just as with stores, sales across different items are relatively even; and (2) Slight Variations: A few items register slightly higher sales, but these differences are not substantial enough to be of concern.

This uniformity in sales across both stores and items suggests other factors—possibly external conditions or inherent qualities of the items—may be influencing sales more than the store or item itself.

5.3.1.5. Seasonal Decomposition

Finally, a seasonal decomposition of time series is applied to dissect the sales data into its core components—trend, seasonal, and residual. A representative store–item combination for this analysis is investigated (Figure 5-7). The decomposition revealed the following: (1) Original Time Series: The raw sales data, inclusive of both trend and seasonal elements; (2) Trend Component: A stable, nonfluctuating trend without any drastic inclines or declines; (3) Seasonal Component: Seasonal variations that recur with a predictable pattern, displaying noticeable peaks and troughs; and (4) Residual Component: What remains after accounting for trend and seasonal factors, representing random or irregular fluctuations. These decompositions offer valuable insights into the inherent patterns within our data set, which will be particularly useful when constructing and fine-tuning predictive models.

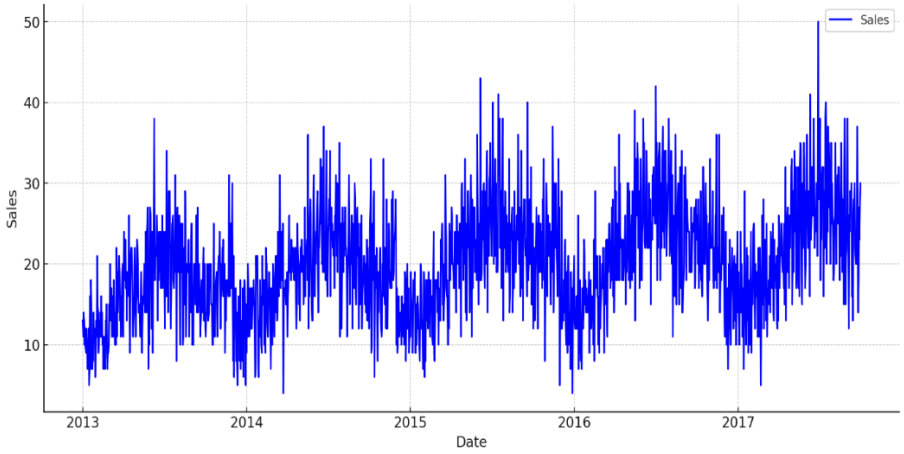


Figure 5-3 Time Series Plot: Sales Trend for Store 1–Item 1

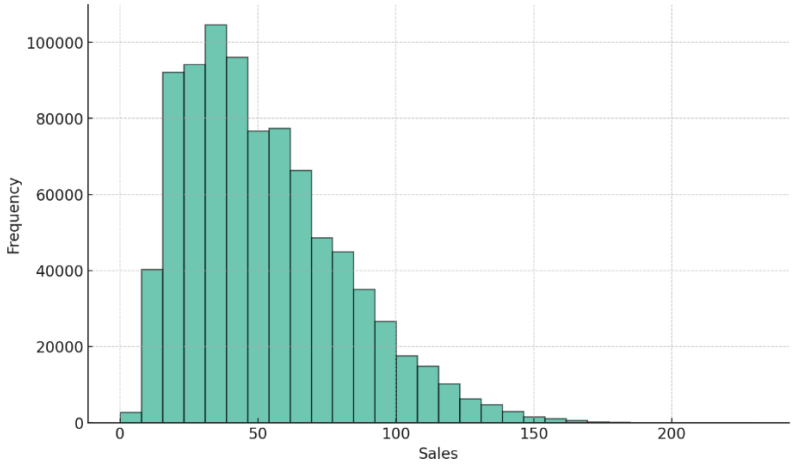


Figure 5-4 Distribution of Daily Sales

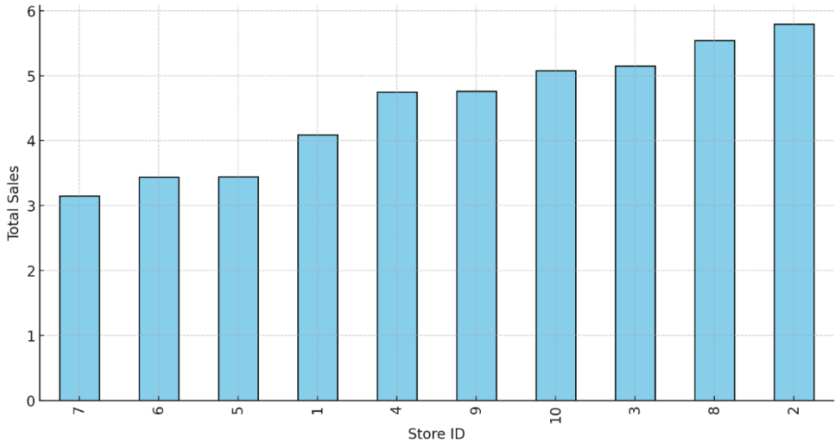


Figure 5-5 Total Sales by Store

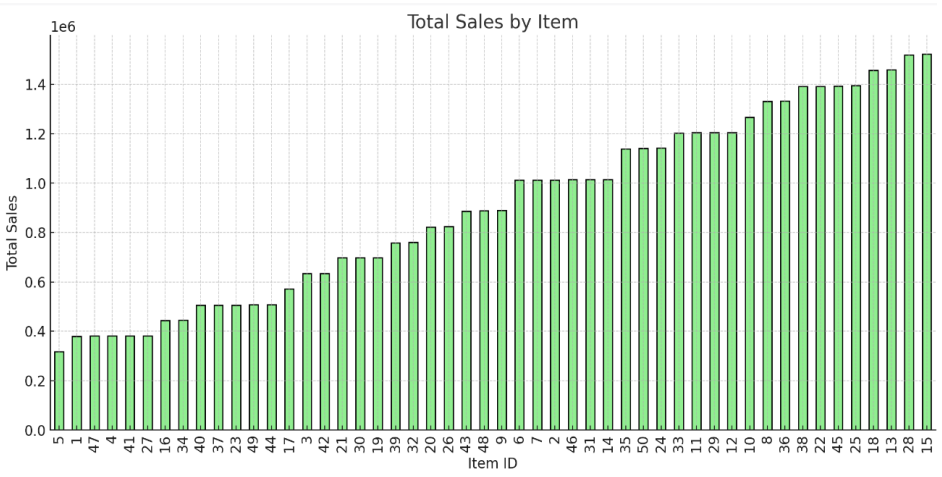


Figure 5-6 Total Sales by Item

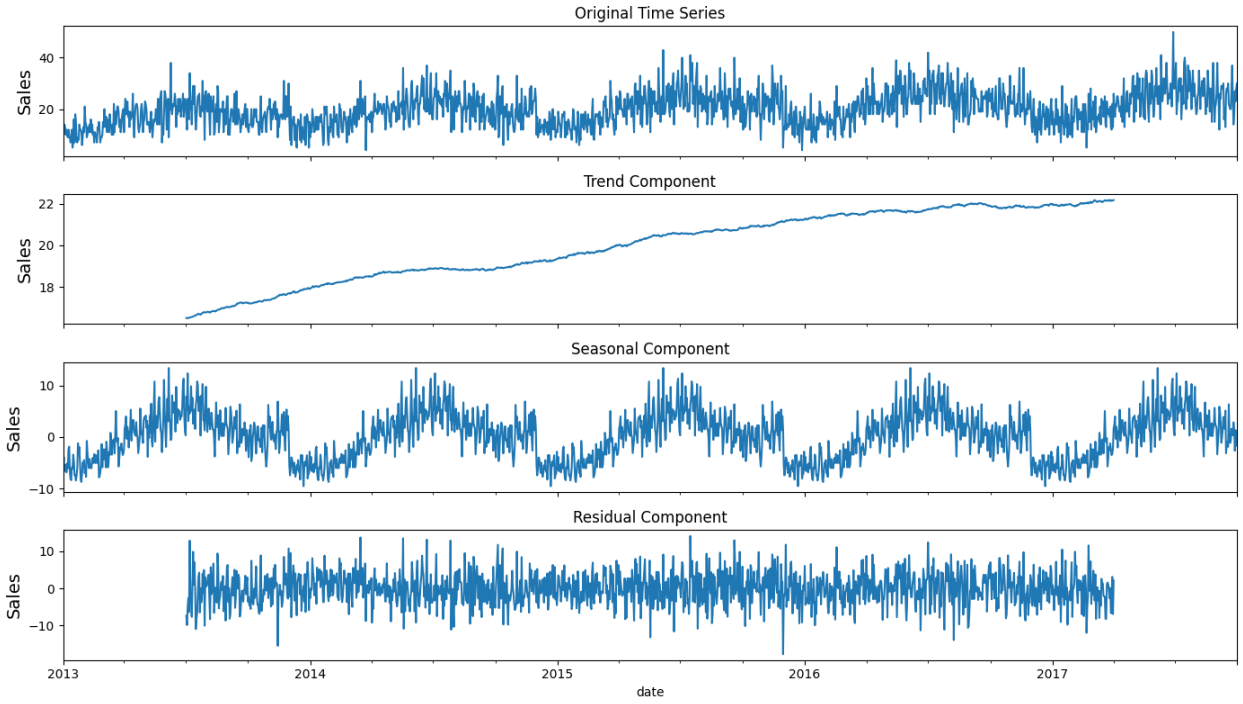


Figure 5-7 Seasonal Decomposition

5.3.2. Data Preprocessing

After completing the EDA, the next step is to preprocess the data to make it suitable for LSTM models. The sequences of length 7 (representing one week) can be used to predict sales for the subsequent day. The sequence length can be customized according to the specific requirements of the problem and dataset. Even though the dataset seemed to be aggregated daily for each store–item combination, it is required to double-checked to ensure only one record exists per day for each combination. Additionally, to preserve the temporal integrity of the data set, the records are sorted by store, item, and date. Then, the data is reshaped into sequences suitable for LSTM models. For example, using a sequence length of 7 days, the input sequence includes sales data from Day 1 to Day 7, where the target output is the sales data for Day 8.

For LSTM models, the features are normalized using mean and standard deviation. To evaluate the model effectively, the dataset is partitioned into training, validation, and test sets by 70%, 15% and 15%, respectively. Finally, given the hierarchical nature of the data—organized by store, then item, and finally date—a sample subset of stores and items is used for retaining the time series structure. For demonstration purposes, the focus is placed on the first five stores and the first 10

items within those stores. After sampling, the dataset consisted of 85,993 input sequences and an equal number of target values, each corresponding to the following day's sales. Now after preparation of data, feature engineering starts, followed by structuring and training of the RMDN model.

5.3.3. Feature Engineering

In feature engineering, the focus is on creating lag-based and temporal features to help the LSTM model capture various dependencies. The lagged features are introduced based on sales data from the past 7 days. This provided a historical context that could help the model understand short-term temporal patterns. Additional temporal features, such as the day of the week, the day of the month, and the month of the year, are included. These features helped the model better understand weekly, monthly, and annual sales patterns.

The rolling window features, such as the rolling mean and rolling standard deviation, are calculated over the past 7 days. These features offered insights into short-term trends and variations, giving the model a richer context for predictions. After data preprocessing and features engineering, the next step will be to proceed with defining and training our LSTM model. Given the complexity of adding an MDN component, the model's architecture is simplified for the sake of demonstration.

5.3.4. Modeling

An RMDN model is employed to fit the training data. The architecture consisted of two primary components, as discussed in the Methodology section: (1) LSTM Layers: These are utilized to capture the temporal dependencies inherent in our time-series data; and (2) The MDN Layer: Unlike traditional models that predict a single outcome, the MDN layer estimates parameters for a mixture of Gaussian distributions, thereby capturing uncertainty in predictions. The model is validated against the final 15% of the dataset.

5.3.5. Model Evaluation

5.3.5.1. Error Analysis

To conduct an error analysis, the model's residuals (i.e., true values minus predicted values) are checked for existence of a pattern. Ideally, residuals should be random. If they are not, this might be an indicator that the model is not capturing all the information. To perform error analysis on the residuals, predictions for the validation set are generated using the model with K = 2, which is

identified as the best value based on the loss curves. Then different values for K between 2 and 5 are used (as presented in section 835.3.6.1) to generate predictions for the validation set and calculate the residuals. The computed residuals are then analyzed (Figures 5.8-5.11) using the following means:

- Histogram of Residuals: This plot gives a visual sense of the distribution of residuals. Ideally, residuals should have a roughly normal (Gaussian) distribution centered around zero. From the histogram, the residuals seem somewhat normally distributed, but there might be a slight skewness.
- Time Series Plot of Residuals: This plot displays residuals over time. It is possible that the model is missing some time-dependent information if there is a clear pattern or trend in this plot. According to the plot, there is no consistent trend. However, spikes indicate specific periods where the model's predictions were off. Investigating these spikes could provide insights into specific scenarios where the model is underperforming.
- Autocorrelation Plots: This plot helps determine whether residuals are correlated with their own previous values (i.e., lagged values). Ideally, residuals should not show any autocorrelation. If they do, it indicates that there is some information in the data that the model is not capturing. The plot seems to show some significant spikes, suggesting some level of autocorrelation. This might be an area to investigate further.
- Q-Q Plots: This plot compares the shapes of distributions, providing a graphical view of how properties such as location, scale, and skewness are similar or different in the two distributions. The residuals are used in this context to determine whether they are normally distributed. As long as the residuals lie on the red line, they are perfectly normal. A plot like this indicates the residuals are reasonably close to normal but have some deviations, especially in the tails.

Based on time series plot (Figure 5-9), the model appears to perform reasonably well, but there may be periods in which it underperforms. Additionally, the presence of some autocorrelation in the residuals suggests there might still be some temporal information the model is not capturing.

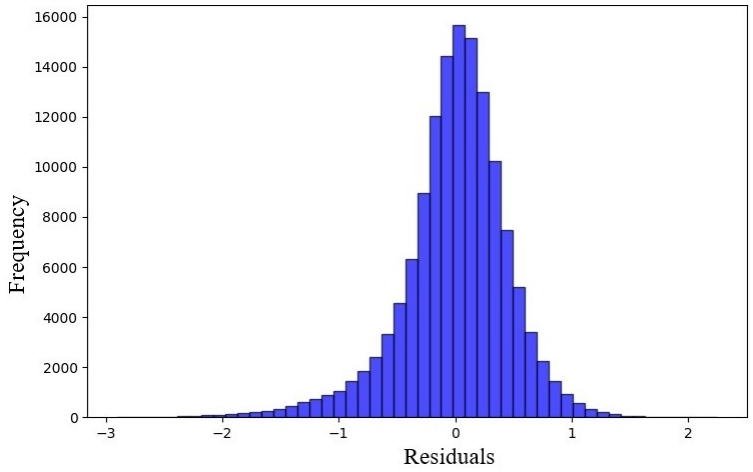


Figure 5-8 Histogram of Residuals

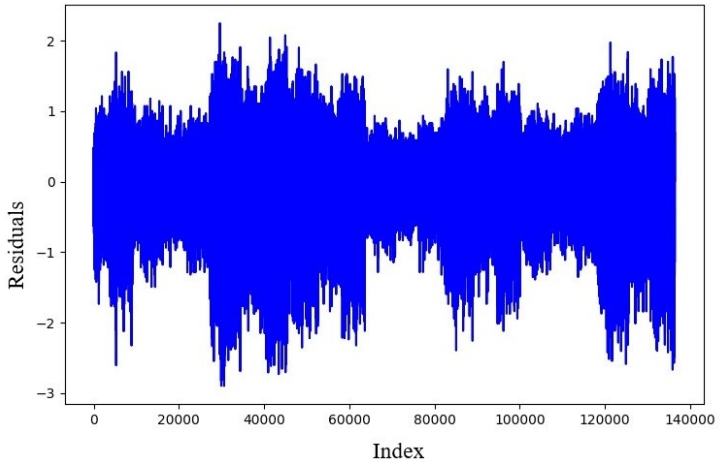


Figure 5-9 Time Series Plot of Residuals

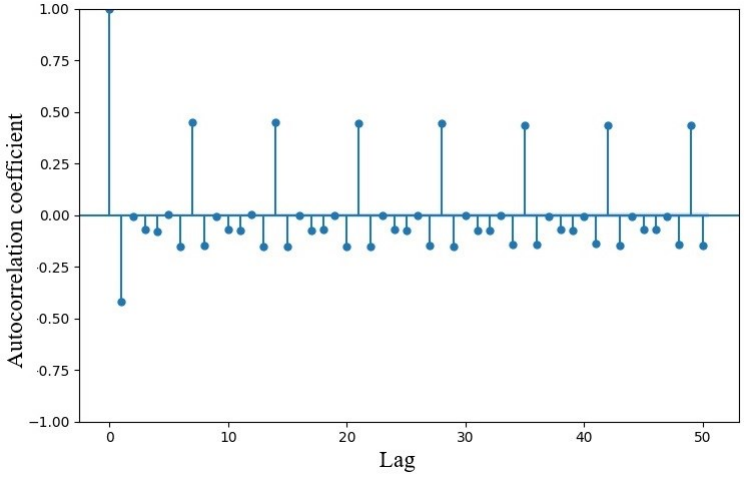


Figure 5-10 Autocorrelation Plot

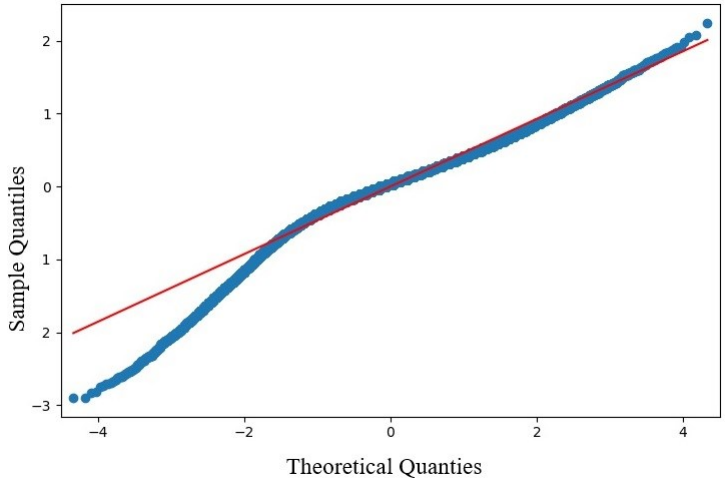


Figure 5-11 Q-Q Plot

5.3.5.2. Uncertainty Estimation

One of the advantages of MDNs is that they can provide a representation of uncertainty in predictions. For each prediction, not only the most likely value but also the spread or variance of the possible values can be evaluated. To estimate and interpret uncertainty with MDNs, mixture components are extracted. From the MDN output, the mixture weights π, the means μ, and the standard deviations σ are extracted. Recall that, for each data point, the MDN generates these parameters for each mixture component *i*. The expected value (i.e., mean) and variance for the mixture model can be computed from these parameters, where K is the number of mixture components:

$$\mu = \sum_{i=1}^{K} \pi_i \mu_i \tag{5-16}$$

$$\sigma^2 = \sum_{i=1}^{K} \pi_i (\sigma_i^2 + \pi_i^2) - \mu^2 \tag{5-17}$$

Where the Expected Value provides the most likely prediction (akin to the mean prediction in standard regression), and Variance provides a measure of the spread or uncertainty of the prediction. A higher variance indicates greater uncertainty in the prediction. When the variance is low, the model is quite confident about its predictions.

Typically, uncertainty is visualized by plotting the expected value and shading the region that contains a certain percentage (e.g., 95%). This shaded region, often called a confidence or prediction interval, provides a range in which the true value is expected to lie with a specified probability.

The area between $mean - \sqrt{variance}$ and $mean + \sqrt{variance}$ provides a way to visualize a range that the model believes could likely contain the true value. This is essentially a prediction interval derived from the variance.

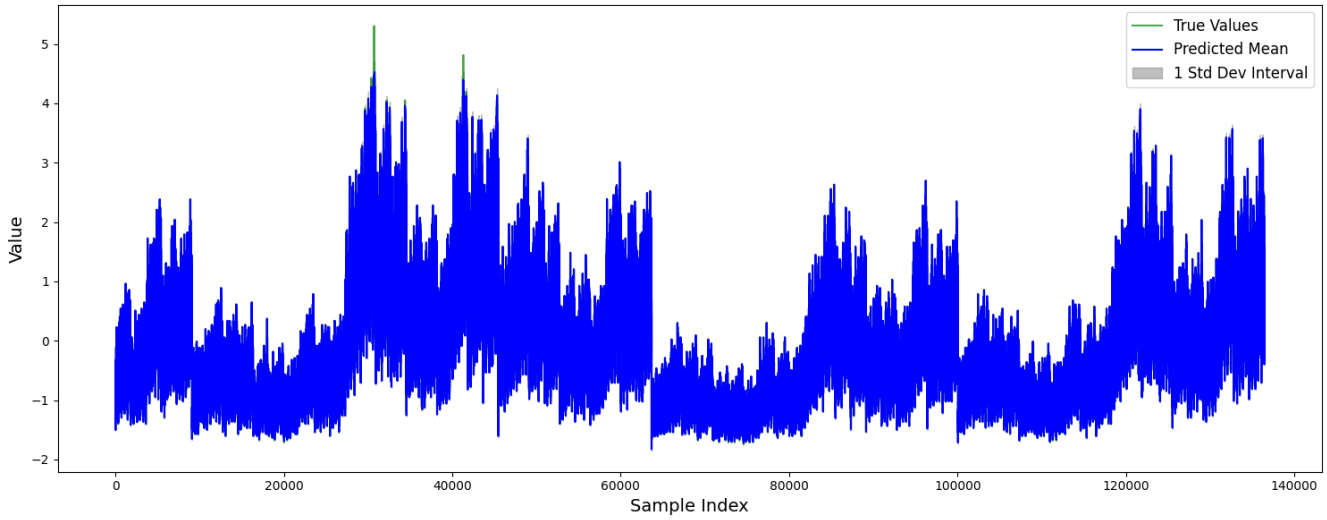


Figure 5-12 True Values vs. Predicted Means with Uncertainty

Figure 5-12 demonstrates true values versus predicted means subject to uncertainty. Some observations and interpretations based on this plot include the following:

- Prediction Accuracy: The blue line (i.e., predicted mean) closely follows the green line (i.e., true values) for a significant portion of the data, indicating the model's predictions are quite accurate for that portion.
- Uncertainty Intervals: The shaded region represents the range within one standard deviation of the predicted mean. The ideal case is to have the true values (i.e., green line) mostly placed within this shaded region because it indicates the model's confidence in its predictions. This is the case for most of the data points, suggesting the model's uncertainty estimates are reasonably calibrated. However, there are regions where the true values are placed outside the shaded region, indicating the model's uncertainty estimates can sometimes be undermined.
- Trends and Patterns: The overall trends in the data seem somewhat cyclic, with regular peaks and troughs. The model has captured this cyclic behavior reasonably well, as evidenced by the blue line's pattern. This indicates the model has learned some of the underlying patterns in the data.
- Model's Confidence: In areas where the shaded region is narrow, the model is more confident in its predictions. Conversely, where the shaded region is wide, the model is less certain. It

is interesting to note where the model feels more or less confident and compare that with where its predictions are accurate or off-mark.

- Potential Improvements: Although the model performs reasonably well, there are areas of improvement. Incorporating additional features, tweaking the architecture might lead to better performance. The regions where the predictions deviate from the true values or where the uncertainty is not well-calibrated can be studied in detail to determine whether there is a systematic pattern or if certain data characteristics are causing the model to falter.

In summary, RMDNs not only provide point predictions but also give a measure of uncertainty. This information can be invaluable in many real-world scenarios where understanding the prediction uncertainty is as crucial as the prediction itself.

5.3.6. Model Optimization

5.3.6.1. Hyperparameter Tuning

In the realm of neural networks and deep-learning models like the RMDNs, several hyperparameters can be fine-tuned to optimize performance. Starting with the number of units in the LSTM layers, it is crucial to strike a balance: too few units may fail to capture the data's complexity, and too many could lead to overfitting. Similarly, the learning rate, which dictates the step size at each iteration toward optimizing the loss function, needs careful calibration. A rate that is too high might cause overshooting of the optimal solution, whereas a rate that is too low could result in slow convergence or even getting stuck in a local optimum.

Batch size is another critical parameter that influences not only the speed of training but also the model's ability to generalize. Alongside these, the sequence length fed into the LSTM plays a role in how effectively the model captures temporal dependencies. Moreover, the number of mixtures in the MDN can significantly affect how well the model represents the target variable's distribution. To streamline this complex hyperparameter tuning process, a grid search approach was initially employed.

To determine which K (i.e., number of mixtures) is best, an evaluation is conducted based on the following criteria:

- Loss Curve Smoothness: A smooth loss curve generally indicates the model is learning consistently without many fluctuations.

- Convergence Speed: This refers to how quickly the loss decreases and stabilizes.
- Final Loss Value: The lower, the better, but it should be compared to how the model performs on validation and test data sets to ensure there is no overfitting.
- Validation Loss: Ideally, the validation loss shall be low and stable, without sharp increases that might indicate overfitting.

For the case study data, among K=1 to K=5; K = 2 results the best outcomes. It offers a balance where both training and validation losses decrease smoothly, suggesting a good fit without clear signs of overfitting.

5.4. Discussion

This This section includes a comparative analysis of the RMDNs and LSTM models, particularly concerning supply chain metrics such as order quantity, safety stock, and reorder point. The RMDN model, with its capacity to output a mixture of distributions, offers a more nuanced understanding of demand variability. This feature becomes invaluable when calculating safety stock, where a mere point estimate could lead to either overstocking or understocking. The RMDN model's ability to provide a measure of uncertainty around its predictions can be leveraged to set more accurate and dynamic safety stock levels, ultimately reducing holding costs and the risk of stockouts.

Conversely, although LSTM models are powerful in capturing temporal dependencies, they only provide point estimates for future sales. This lack of information about the distribution of possible outcomes can be a significant limitation when determining the reorder point and order quantity. Because the reorder point is often calculated based on lead time and demand variability, the point estimates from LSTM models can lead to static and sometimes inaccurate reorder points. In contrast, the RMDN model allows for dynamic adjustments by incorporating uncertainty in its forecasts. The RMDN model can help businesses make informed decisions by understanding the range of possible demand scenarios, thereby optimizing inventory costs. Therefore, regarding supply chain optimization, the RMDN model offers a more comprehensive and flexible approach over traditional LSTM models.

Figure 5-13 demonstrates inventory metrics such as order quantity, safety stock, and reorder point for different lead times (7 and 14), and Z = 95%. These inventory metrics under the OUTL policy can be defined as the following:

- Order Quantity (Q): This is the quantity that should be ordered to replenish inventory up to the target level. Under the OUTL policy, the order quantity would be the difference between the OUTL (S) and the current inventory level (I). In a more advanced setting, this can be adjusted based on the lead time demand mean.

$$Q = S - I \tag{5-18}$$

- Safety Stock (SS): This is the stock kept mitigating the risk of stockouts due to variability in demand and lead time. Under this policy, the safety stock can be calculated using the weighted standard deviation of the lead time demand (Equation (5-15)).
- Reorder Point (r): This is the inventory level at which a new order should be placed.

$$r = \mu_{LTD} + SS \tag{5-19}$$

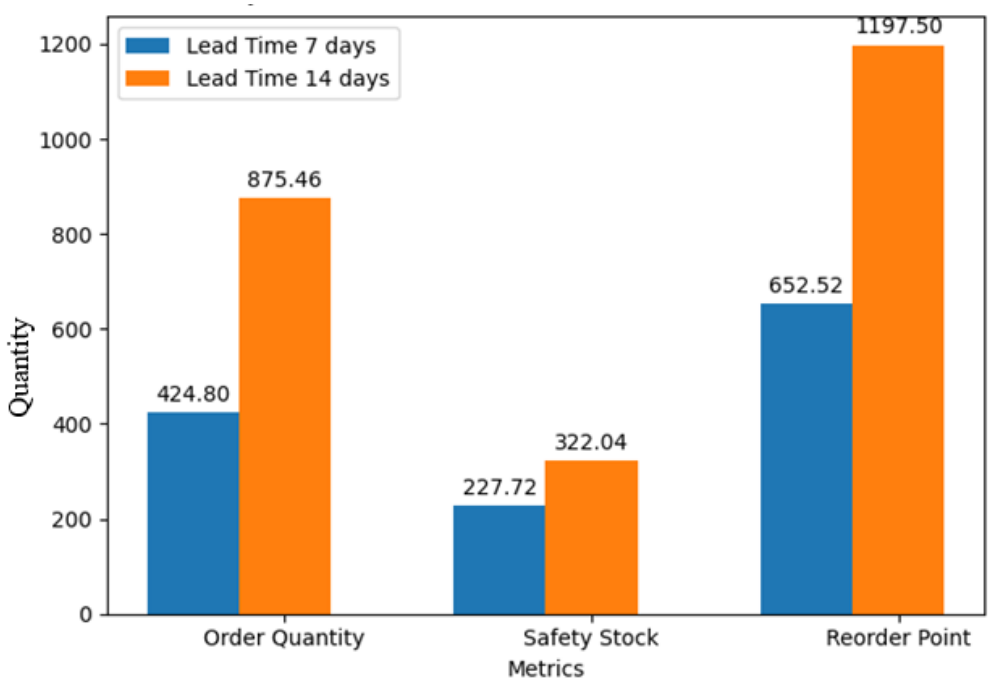


Figure 5-13 Inventory Metrics for Different Lead Times

5.4.1. Sensitivity Analysis (RMDNs)

Sensitivity analysis is used to understand how different values of an independent variable could affect (change) a particular dependent variable under a given set of assumptions. In the context of inventory management, parameters like lead time, service level, and demand forecasts can be changed to observe how they affect key metrics like order quantity, safety stock, and reorder point. In doing so, the following scenarios are considered with regard to forecasted demand based on RMDNs.

5.4.1.1. Varying Service Level

The safety stock and reorder point are calculated for different service levels of 90%, 95%, and 99%. Varied lead times of 7 and 14 days are also used for the analysis. The safety stock formula with varying service levels was based on Equation 5-15, where z was the Z-score corresponding to the service levels of (90%, Z = 1.28), (95%, Z = 1.64), and (99%, Z = 2.33).

The above Z-scores are used to calculate the safety stock and reorder point for the two different lead times (7 and 14 days).

5.4.1.2. Varying Demand Forecast Accuracy

The output from the RMDN model gives us a mean and standard deviation, representing a range of possible demands. Improving or worsening the model's accuracy would change these parameters. As the service level increases, the order quantity also increases. This is due to the fact that a higher service level necessitates more safety stock to meet demand, which in turn increases the order quantity. Additionally, the safety stock increases with both the service level and the lead time. A higher service level requires more safety stock to meet the risk of stockouts. The reorder point is essentially the sum of the mean demand and the safety stock for the given lead time. It increases with both the service level and the lead time.

If the RMDN model's accuracy improves, the standard deviation of the demand forecast decreases. This would, in turn, reduce the safety stock and the reorder point. Conversely, if the model's accuracy declines, the standard deviation would increase, raising both the safety stock and the reorder point. The sensitivity analysis, as presented in Figure 5-14, demonstrates how different service levels (90%, 95%, and 99%) and lead times (7 and 14 days) affect inventory metrics such as order quantity, safety stock, and reorder point. Here are some key observations:

- Order quantity increases with an increase in lead time for all service levels and is higher for higher service levels at both 7 and 14 days lead time.
- Safety stock also increases with an increase in lead time and is significantly higher for a 99% service level compared to 90% and 95%.
- Reorder points follow the same trend as the order quantity because they are equal in an OUTL policy and are higher for higher service levels.

It is worth mentioning that, although a higher service level (e.g., 99%) ensures better service, it requires maintaining a higher level of safety stock, which in turn increases the holding costs. Longer lead times require higher order quantities and safety stock levels to maintain the same service level, impacting both holding and ordering costs. There is a clear trade-off between the service level, safety stock, and costs.

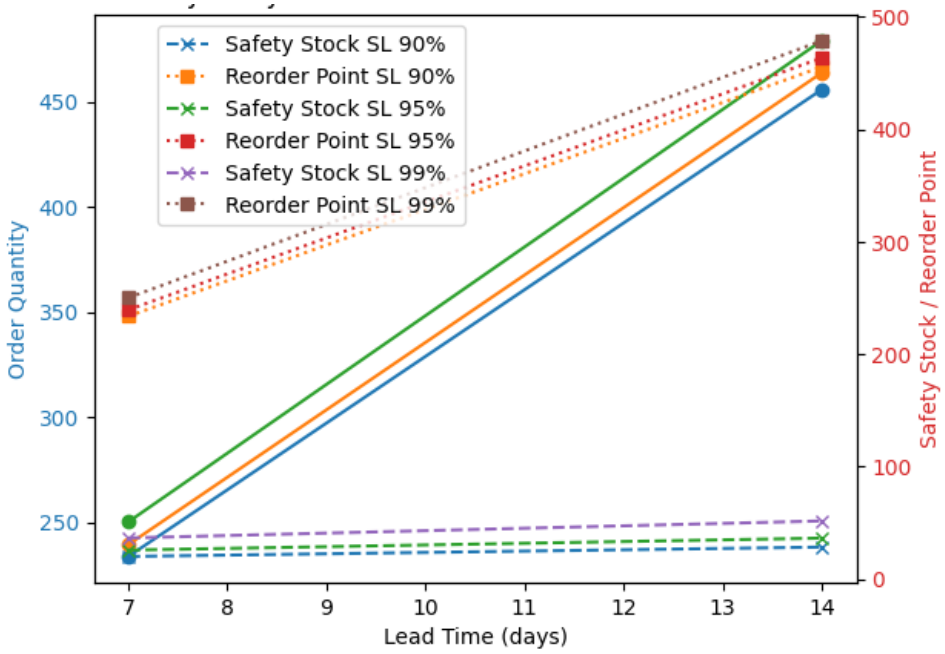


Figure 5-14 Sensitivity Analysis for Different Service Levels and Lead Times

5.4.2. Comparison study

In this section, we choose LSTM networks as a benchmark due to their methodological similarity to RMDNs. Both being advanced neural network models, they are well-equipped to handle non-linear dependencies inherent in time-series data, making LSTM an appropriate and direct

comparison for evaluating RMDNs' performance in our context. This comparison aims to shed light on the strengths and limitations of each approach in terms of inventory management, forecast accuracy, and computational efficiency. To this end, we have conducted a similar sensitivity analysis for inventory metrics using the LSTM model, with results compared against those obtained from RMDNs, as shown in Table 5.1.

Table 5.1 Sensitivity Analysis for Inventory Metrics Based on RMDNs and LSTM

		RMDNs			**LSTM**		
Service Level	**Lead Time (Days)**	**Order Quantity**	**Safety Stock**	**Reorder Point**	**Order Quantity**	**Safety Stock**	**Reorder Point**
90	7	234	20	234	425	85	425
90	14	456	28	456	800	121	800
95	7	239	26	239	449	110	449
95	14	464	36	464	834	155	834
99	7	250	36	250	495	155	495
99	14	479	52	479	899	219	899

Comparing the accuracy of demand forecasting using RMDNs and LSTM, it can be observed that:

- For RMDNs: The order quantity and reorder point are generally lower than the LSTM model, indicating the RMDN model is likely providing a more confident demand prediction.
- For LSTM: The higher values for these metrics suggest this model is more considerate of uncertainty, potentially overestimating demand.

Comparing safety stock calculations, the following results can be observed:

- For RMDNs: Safety stock values are significantly lower than those from the LSTM model, which could result in lower holding costs.
- For LSTM: Higher safety stock values indicate a more conservative approach, possibly leading to increased holding costs.

Comparing complexity and computational cost, it can be concluded that:

- For RMDNs: Although more computationally expensive, it provides a more nuanced inventory strategy.
- For LSTM: The model is less computationally expensive but may result in a less efficient inventory system.

With regard to flexibility and adaptability, it is observed that:

- For RMDNs: The model appears to adapt better to different service levels and lead times, as evidenced by the more moderate changes in safety stock and reorder point.
- For LSTM: The model shows more significant jumps in these values, indicating less flexibility.

Considering inventory costs, it is found that:

- For RMDNs: The model is likely to result in lower inventory costs because of lower safety stock and reorder point values.
- LSTM: The model may incur higher costs because of the overestimation of these metrics.

Regarding the risk of stockouts and overstocking, it is shown that:

- For RMDNs: The model has a lower risk because it has more accurate safety stock and order quantity estimates.
- For LSTM: The model has a higher risk of overstocking and possibly higher carrying costs because of the conservative nature of the model.

Regarding sensitivity to service level, it is observed that:

- RMDNs: The model shows moderate increases in safety stock and reorder point as the service level increases.
- LSTM: The model demonstrates more considerable increases, indicating higher sensitivity to changes in service level.

In summary, the RMDN model offers a more efficient and cost-effective inventory management solution in comparison with the traditional LSTM model, albeit at the cost of increased computational complexity.

5.5. Conclusion

Effective demand forecasting remains a cornerstone for ensuring successful inventory management and maintaining high levels of customer satisfaction. This study took a dual approach to demand forecasting, employing both LSTM and RMDNs to offer groundbreaking insights into safety stock calculation and demand prediction. By specifically focusing on the role of demand distribution in calculating safety stock, this research fills a gap often overlooked in the literature. It goes a step further by employing RMDNs to model the probability density function of demand, thereby facilitating more nuanced and informed decision-making in supply chain operations.

This comparative analysis of the LSTM and RMDN methodologies using actual retail data revealed important findings. RMDNs consistently yield more accurate safety stock and reorder point estimates than the LSTM model, leading to reduced inventory costs and improved customer satisfaction. Furthermore, LSTMs tend to overestimate these parameters, possibly increasing operational costs and reducing operational efficiency, even though they are easier to implement. These findings have considerable practical implications. By offering a precise prediction of stockout and overstock scenarios, our methodology allows retailers to fine-tune their safety stock estimates. In addition to inventory cost reductions, this could improve customer satisfaction and overall supply chain efficiency.

There are a number of avenues for future research. As it is still relatively new and underexplored to apply RMDNs to supply chain forecasting, it may be beneficial to examine the long-term impact of using RMDNs in these contexts. Furthermore, the role of external factors, such as seasonal variations and economic indicators, in improving the model's accuracy could be explored. Another interesting area for future work could be the integration of real-time data streams into the RMDN model to create a dynamic and responsive inventory management system.

This research made a substantial contribution to demand forecasting and inventory management by advocating for the use of RMDNs. This approach not only refined safety stock estimates through forming a richer understanding of demand distribution but also resulted in a cost-effective and operationally efficient inventory management. It offered a practical yet innovative solution for retailers and other similar demand-driven businesses in today's volatile and complex markets.

Chapter 6. Summary, Limitations and Future Work

6.1. Summary

This thesis comprised three distinct sections, each contributing to the advancement of demand forecasting and inventory management. In Chapter 3, we introduced a novel cluster-based demand forecasting methodology using ensemble learning techniques. Focused on the sports and electronics retail industry, this approach aimed to enhance the accuracy of daily demand forecasts. By combining time series forecasting methods like LSTM and Prophet with ensemble learning methods such as majority voting and BMA, we strived to create a robust framework. The results showcased substantial improvements in prediction accuracy, with the clustered-ensembled approach yielding a minimum MAPE of 8.12%, MAE of 27.1, and RMSE of 32.8 for daily forecasts. BMA further demonstrated its effectiveness in reducing forecast errors. However, it is important to acknowledge that this framework has limitations related to data availability and size, which could be addressed in future research.

In Chapter 4, we explored the application of multivariate time series forecasting to optimize safety stock and inventory management. Employing deep learning models like MLP, LSTM, and 1D-CNN, we developed an ensemble forecasting approach. This approach outperformed individual models, improving forecasting accuracy, stability, and generalization in both the sports and electronics product domains by comparing safety stock calculation and total cost. Yet we must recognize that constructing and combining the basic predictors posed computational challenges, warranting future research into more efficient strategies. Additionally, further exploration of ensemble schemes and their impact on supply chain management could yield valuable insights.

In Chapter 5, we delved into the world of RMDNs to refine distribution demand forecasting and safety stock estimation. These innovative models proved their capabilities by consistently outperforming traditional LSTM models, providing accurate stockout and overstock predictions, ultimately reducing inventory costs and enhancing supply chain efficiency. Nonetheless, there is potential for future research in exploring the long-term impact of RMDNs, integration of external factors, and the real-time application of data streams in dynamic inventory management.

6.2. Limitations

Despite the significant contributions of these sections, certain limitations must be acknowledged. The limitations of Chapters 3 to 5 include the following:

- In Chapter 3
    - The segmentation could have been further improved with access to more personalized customer data such as income rate, age, location, and monthly budget.
    - The size of the dataset was from January 2015 to September 2017. A larger dataset could enhance the multiperiod demand forecasts.
- In Chapter 4
    - The complexity of constructing and combining basic predictors posed computational challenges, which could be addressed to improve efficiency.
    - Further exploration of ensemble schemes and broader impacts on supply chain management are also avenues for future research.
- In Chapter 5
    - Although RMDNs showed promise, there is room for exploring their long-term impact, integration of external factors, and real-time data streams for dynamic inventory management.

6.3. Directions for Future Work

Future research in the domain of demand forecasting and inventory management could delve into several specific areas to advance the field and provide practical benefits to businesses:

1. Integration of Forecasting Models with Optimization: Investigate the seamless integration of proposed forecasting models with advanced optimization models that offer prescriptive capabilities. This research could focus on developing algorithms that not only predict demand but also suggest optimal inventory levels and ordering policies based on various constraints and objectives.
2. Prescriptive Analytics in Demand Forecasting: Explore the application of prescriptive analytics techniques to demand forecasting. This involves not only predicting future demand but also recommending specific actions to optimize inventory levels, procurement strategies,

and supply chain operations in real time. Reinforcement learning is an exiting method to be explored to address point 1 & 2.

3. Enhancing Computational Efficiency in Ensemble Modeling: Develop strategies to enhance the computational efficiency of ensemble modeling techniques. This could include exploring parallel computing, distributed computing, or other methods to speed up the processing of large datasets in ensemble forecasting.
4. Effective Ensemble Schemes: Investigate and design more effective ensemble schemes by combining various forecasting methods and models. Evaluate the performance of different ensemble combinations and identify those most suitable for specific business scenarios.
5. Impact of Approaches on Supply Chain Operations: Examine the broader impacts of these advanced forecasting and inventory management approaches on supply chain operations. Assess how improved demand forecasting and inventory optimization affect aspects like lead times, order fulfillment, and overall supply chain efficiency.
6. Inventory Policies for Perishable Products: Focus on optimizing inventory management for perishable products with life expectancy constraints. Develop inventory policies and models that account for product shelf life, demand variability, and dynamic pricing strategies to minimize waste and maximize profit.

The practical significance of these research directions is profound for businesses and retailers looking to elevate their demand forecasting and inventory management practices:

- Leveraging Ensemble Learning: Organizations can harness ensemble learning techniques like clustering and ensemble deep learning to enhance forecast accuracy. This leads to reduced forecasting errors, optimized inventory levels, and ultimately lower operational costs.
- Adoption of RMDNs: Implementing RMDNs for demand forecasting and safety stock estimation empowers more informed decision-making. This can result in reduced inventory holding costs and increased customer satisfaction because of higher stock availability.
- Integration into Supply Chain Operations: It is recommended that businesses consider integrating these innovative approaches into their supply chain operations. Adaptability to market changes and customer demands gives companies a competitive advantage. Advanced demand forecasting and inventory management can provide these advantages

and can result in improved customer service levels, reduced carrying costs, and more efficient supply chains.

REFERENCES

Abbasi, B., Babaei, T., Hosseinifard, Z., Smith-Miles, K., & Dehghani, M. (2020). Predicting solutions of large-scale optimization problems via machine learning: A case study in blood supply chain management. *Computers and Operations Research*, *119*, 104941. https://doi.org/10.1016/j.cor.2020.104941

Abbasimehr, H., & Shabani, M. (2021). A new framework for predicting customer behavior in terms of RFM by considering the temporal aspect based on time series techniques. *Journal of Ambient Intelligence and Humanized Computing*, *12*(1), 515–531. https://doi.org/10.1007/s12652-020-02015-w

Abbasimehr, H., Shabani, M., & Yousefi, M. (2020). An optimized model using LSTM network for demand forecasting. *Computers and Industrial Engineering*, *143*(July 2019), 106435. https://doi.org/10.1016/j.cie.2020.106435

Acar, Y., & Gardner, E. S. (2012). Forecasting method selection in a global supply chain. *International Journal of Forecasting*, *28*(4), 842–848. https://doi.org/10.1016/J.IJFORECAST.2011.11.003

Alqurashi, T., & Wang, W. (2019). Clustering ensemble method. *International Journal of Machine Learning and Cybernetics*, *10*(6), 1227–1246. https://doi.org/10.1007/s13042-017-0756-7

Andrade, L. A. C. G., & Cunha, C. B. (2023). Disaggregated retail forecasting: A gradient boosting approach[Formula presented]. *Applied Soft Computing*, *141*, 110283. https://doi.org/10.1016/j.asoc.2023.110283

Andriolo, A., Battini, D., Grubbström, R. W., Persona, A., & Sgarbossa, F. (2014). A century of evolution from Harris's basic lot size model: Survey and research agenda. *International Journal of Production Economics*, *155*, 16–38. https://doi.org/10.1016/j.ijpe.2014.01.013

Arrow, K. J., Karlin, S., Scarf, H., Beckmann, M. J., Gessford, J., & Muth, R. F. (1958). Studies in the mathematical theory of inventory and production. *Stanford, Calif., Stanford University Press*.

Athlye, A., & Bashani, A. (2018). *Multivariate Demand Forecasting of Sales Data*. *6*(X), 198–211.

Axsäter, S. (2015). Inventory Control. In *Springer eBooks*.

Babai, M. Z., Ali, M. M., Boylan, J. E., & Syntetos, A. A. (2013). Forecasting and inventory performance in a two-stage supply chain with ARIMA(0,1,1) demand: Theory and empirical analysis. *International Journal of Production Economics*, *143*(2), 463–471. https://doi.org/10.1016/J.IJPE.2011.09.004

Babai, M. Z., Dai, Y., Li, Q., Syntetos, A., & Wang, X. (2022). Forecasting of lead-time demand variance: Implications for safety stock calculations. *European Journal of Operational Research*, *296*(3), 846–861. https://doi.org/10.1016/j.ejor.2021.04.017

Babai, M. Z., Jemai, Z., & Dallery, Y. (2011). Analysis of order-up-to-level inventory systems with compound Poisson demand. *European Journal of Operational Research*, *210*(3), 552–558. https://doi.org/10.1016/j.ejor.2010.10.004

Babai, M. Z., Tsadiras, A., & Papadopoulos, C. (2020). On the empirical performance of some new neural network methods for forecasting intermittent demand. *IMA Journal of Management Mathematics*, *31*(3), 281–305. https://doi.org/10.1093/imaman/dpaa003

Bai, Y., Xie, J., Wang, D., Zhang, W., & Li, C. (2021). A manufacturing quality prediction model based on AdaBoost-LSTM with rough knowledge. *Computers and Industrial Engineering*, *155*(January), 1–10. https://doi.org/10.1016/j.cie.2021.107227

Ban, G. Y., & Rudin, C. (2019). The big Data newsvendor: Practical insights from machine learning. *Operations Research*, *67*(1), 90–108. https://doi.org/10.1287/opre.2018.1757

Ban, G.-Y., & Keskin, N. B. (2020). Personalized Dynamic Pricing with Machine Learning: High Dimensional Features and Heterogeneous Elasticity. *Management Science*. https://doi.org/10.2139/ssrn.2972985

Bandara, K., Bergmeir, C., & Smyl, S. (2020). Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach. *Expert Systems with Applications*, *140*. https://doi.org/10.1016/j.eswa.2019.112896

Barrow, D. K., & Crone, S. F. (2016). A comparison of AdaBoost algorithms for time series forecast combination. *International Journal of Forecasting*, *32*(4), 1103–1119. https://doi.org/10.1016/j.ijforecast.2016.01.006

Barrow, D. K., & Kourentzes, N. (2016). Distributions of forecasting errors of forecast combinations: Implications for inventory management. *International Journal of Production Economics*, *177*, 24–33. https://doi.org/10.1016/J.IJPE.2016.03.017

Beneditto, C., Satrio, A., Darmawan, W., Nadia, B. U., & Hanafiah, N. (2021). Time series analysis and forecasting of coronavirus disease in Indonesia using ARIMA model and PROPHET. *Procedia Computer Science*, *179*(2020), 524–532.

Bishop, C. M. (1995). Building Neural Network for Pattern Recognition. In *CLARENDON PRESS OXFORD*. https://doi.org/10.1109/RusAutoCon49822.2020.9208207

Bozkir, A. S., & Sezer, E. A. (2011). Predicting food demand in food courts by decision tree approaches. *Procedia Computer Science*, *3*, 759–763. https://doi.org/10.1016/J.PROCS.2010.12.125

Bradlow, E. T., Gangwar, M., Kopalle, P., & Voleti, S. (2017). The Role of Big Data and Predictive Analytics in Retailing. *Journal of Retailing*, *93*(1), 79–95. https://doi.org/10.1016/j.jretai.2016.12.004

Cao, Y., & Shen, Z. J. M. (2019). Quantile forecasting and data-driven inventory management under nonstationary demand. *Operations Research Letters*, *47*(6), 465–472. https://doi.org/10.1016/j.orl.2019.08.008

Carbonneau, R., Laframboise, K., & Vahidov, R. (2008). Application of machine learning techniques for supply chain demand forecasting. *European Journal of Operational Research*, *184*(3), 1140–1154. https://doi.org/10.1016/J.EJOR.2006.12.004

Carlsson, G., Mémoli, F., Ribeiro, A., & Segarra, S. (2018). Hierarchical clustering of asymmetric networks. *Advances in Data Analysis and Classification*, *12*(1), 65–105. https://doi.org/10.1007/s11634-017-0299-5

Chen, I. F., & Lu, C. J. (2017). Sales forecasting by combining clustering and machine-learning techniques for computer retailing. *Neural Computing and Applications*, *28*(9), 2633–2647. https://doi.org/10.1007/s00521-016-2215-x

Chen, M., Chen, R., Cai, F., Li, W., Guo, N., & Li, G. (2021). Short-Term Traffic Flow Prediction with Recurrent Mixture Density Network. *Mathematical Problems in Engineering*, *2021*. https://doi.org/10.1155/2021/6393951

Clausen, J. B. B., & Li, H. (2022). Big data driven order-up-to level model: Application of machine learning. *Computers and Operations Research*, *139*(November 2021), 105641. https://doi.org/10.1016/j.cor.2021.105641

Collica, R. S. (2017). *Customer Segmentation and Clustering Using SAS Enterprise Miner, Third Edition*. SAS Institute Inc.

Constante, F., Silva, F., & Pereira, A. (2019). *DataCo SMART SUPPLY CHAIN FOR BIG DATA ANALYSIS*. Mendeley Data. https://doi.org/10.17632/8gx2fvg2k6.5

Coussement, K., Van den Bossche, F. A. M., & De Bock, K. W. (2014). Data accuracy's impact on segmentation performance: Benchmarking RFM analysis, logistic regression, and decision trees. *Journal of Business Research*, *67*(1), 2751–2758. https://doi.org/10.1016/J.JBUSRES.2012.09.024

Das Adhikari, N. C., Garg, R., Datt, S., Das, L., Deshpande, S., & Misra, A. (2018). Ensemble methodology for demand forecasting. *Proceedings of the International Conference on Intelligent Sustainable Systems, ICISS 2017*, *Iciss*, 846–851. https://doi.org/10.1109/ISS1.2017.8389297

Du, S., Wang, J., Wang, M., Yang, J., Zhang, C., Zhao, Y., & Song, H. (2023). A systematic data-driven approach for production forecasting of coalbed methane incorporating deep learning and ensemble learning adapted to complex production patterns. *Energy*, *263*. https://doi.org/10.1016/j.energy.2022.126121

Espinoza, M., Joye, C., Belmans, R., & Moor, B. De. (2005). *Short-Term Load Forecasting , Profile Identification , and Customer Segmentation : A Methodology Based on Periodic Time Series*. *20*(3), 1622–1630.

Fildes, R., Ma, S., & Kolassa, S. (2019). Retail forecasting : Research and practice. *International Journal of Forecasting*, *xxxx*. https://doi.org/10.1016/j.ijforecast.2019.06.004

Galar, M., Fern, A., Barrenechea, E., & Bustince, H. (2012). *A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches*. *42*(4), 463–484.

Ganaie, M. A., Hu, M., Malik, A. K., Tanveer, M., & Suganthan, P. N. (2022). Ensemble deep learning: A review. *Engineering Applications of Artificial Intelligence*, *115*(June), 105151. https://doi.org/10.1016/j.engappai.2022.105151

Gastinger, J., Nicolas, S., Stepić, D., Schmidt, M., & Schülke, A. (2021). *A study on Ensemble Learning for Time Series Forecasting and the need for Meta-Learning*. *Ijcnn*.

Ge, D., Pan, Y., Shen, Z.-J. (Max), Wu, D., Yuan, R., & Zhang, C. (2019). Retail supply chain management: a review of theories and practices. *Journal of Data, Information and Management*, *1*(1–2), 45–64. https://doi.org/10.1007/s42488-019-00004-z

Ghaemi, R., Sulaiman, N., Ibrahim, H., & Mustapha, N. (2009). A survey: Clustering ensembles techniques. *World Academy of Science, Engineering and Technology*, *38*(February), 644–653. https://doi.org/10.5281/zenodo.1329276

Glock, C. H., Grosse, E. H., & Ries, J. M. (2014). The lot sizing problem: A tertiary study. *International Journal of Production Economics*, *155*, 39–51. https://doi.org/10.1016/j.ijpe.2013.12.009

Goltsos, T. E., Syntetos, A. A., Glock, C. H., & Ioannou, G. (2022). Inventory – forecasting: Mind the gap. *European Journal of Operational Research*, *299*(2), 397–419. https://doi.org/10.1016/j.ejor.2021.07.040

Güven, İ., & Şimşir, F. (2020). Demand forecasting with color parameter in retail apparel industry using artificial neural networks (ANN) and support vector machines (SVM) methods. *Computers and Industrial Engineering*, *147*(July). https://doi.org/10.1016/j.cie.2020.106678

Han, J., Kamber, M., & Pei, J. (2013). Data Mining: Concepts and Techniques. In *Morgan Kaufmann Publishers, USA*. Morgan Kaufmann Publishers, USA. https://doi.org/10.1016/B978-0-12-381479-1.00001-0

Hançerlioğulları, G., Şen, A., & Aktunç, E. A. (2016). Demand uncertainty and inventory turnover performance: An empirical analysis of the US retail industry. *International Journal of Physical Distribution and Logistics Management*, *46*(6–7), 681–708. https://doi.org/10.1108/IJPDLM-12-2014-0303

Harris, F. W. (1913). How Many Parts to Make at Once. *The Magazine of Management*, *10*(2), 135–136.

He, H., Zhang, W., & Zhang, S. (2018). A novel ensemble method for credit scoring: Adaption of different imbalance ratios. *Expert Systems With Applications*, *98*, 105–117. https://doi.org/10.1016/j.eswa.2018.01.012

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks*, *4*(2), 251–257. https://doi.org/10.1016/0893-6080(91)90009-T

Hu, Y. L., & Chen, L. (2018). A nonlinear hybrid wind speed forecasting model using LSTM network, hysteretic ELM and Differential Evolution algorithm. *Energy Conversion and Management*, *173*(May), 123–142. https://doi.org/10.1016/j.enconman.2018.07.070

Huber, J., Müller, S., Fleischmann, M., & Stuckenschmidt, H. (2019). A data-driven newsvendor problem: From data to decision. *European Journal of Operational Research*, *278*(3), 904–915. https://doi.org/10.1016/j.ejor.2019.04.043

Hyndman, R. J., & Athanasopoulos, G. (2021). *Forecasting : principles and practice* (3rd editio). OTexts.

İfraz, M., Aktepe, A., Ersöz, S., & Çetinyokuş, T. (2023). Demand forecasting of spare parts with regression and machine learning methods: Application in a bus fleet. *Journal of Engineering Research*, *11*(2), 100057. https://doi.org/10.1016/j.jer.2023.100057

inversion. (2018). *Store Item Demand Forecasting Challenge*. Kaggle. https://kaggle.com/competitions/demand-forecasting-kernels-only

Ivanov, D., Tsipoulanidis, A., & Schönberger, J. (2019). *Global Supply Chain and Operations Management* (Second). Springer International Publishing. https://doi.org/10.1007/978-3-319-94313-8

Jain, A. K., & Dubes, R. C. (1988). *Algorithms for clustering data*. Prentice-Hall, Inc.

Jason, B. (2018). Deep Learning For Time Series Forecasting. *Ml*, *1*(1), 1–50. https://doi.org/10.1093/brain/awf210

Ju, C., Bibaut, A., & van der Laan, M. (2018). The relative performance of ensemble methods with deep convolutional neural networks for image classification. *Journal of Applied Statistics*, *45*(15), 2800–2818. https://doi.org/10.1080/02664763.2018.1441383

Kashwan, K. R., & Velu, C. M. (2013). *Customer Segmentation Using Clustering and Data Mining Techniques*. *5*(6), 1–6. https://doi.org/10.7763/IJCTE.2013.V5.811

Khashei, M., & Bijari, M. (2011). A novel hybridization of artificial neural networks and ARIMA models for time series forecasting. *Applied Soft Computing Journal*, *11*(2), 2664–2675. https://doi.org/10.1016/j.asoc.2010.10.015

Kim, Y. (2014). Convolutional neural networks for sentence classification. *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 1746–1751. https://doi.org/10.3115/v1/d14-1181

Kingma, D. P., & Ba, J. L. (2014). Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 1–15. https://doi.org/https://arxiv.org/abs/1412.6980

Koivu, A., Kakko, J. P., Mäntyniemi, S., & Sairanen, M. (2022). Quality of randomness and node dropout regularization for fitting neural networks. *Expert Systems with Applications*, *207*(May), 117938. https://doi.org/10.1016/j.eswa.2022.117938

Kone, E. R. S., & Karwan, M. H. (2011). Combining a new data classification technique and regression analysis to predict the Cost-To-Serve new customers. *Computers & Industrial Engineering*, *61*(1), 184–197. https://doi.org/10.1016/J.CIE.2011.03.009

Lee, Y. S. (2014). A semi-parametric approach for estimating critical fractiles under autocorrelated demand. *European Journal of Operational Research*, *234*(1), 163–173. https://doi.org/10.1016/j.ejor.2013.10.055

Lemke, C., & Gabrys, B. (2010). Meta-learning for time series forecasting and forecast combination. *Neurocomputing*, *73*(10–12), 2006–2016. https://doi.org/10.1016/j.neucom.2009.09.020

Li, G., & Shi, J. (2010). Application of Bayesian model averaging in modeling long-term wind speed distributions. *Renewable Energy*, *35*(6), 1192–1202. https://doi.org/10.1016/j.renene.2009.09.003

Li, G., Shi, J., & Zhou, J. (2011). Bayesian adaptive combination of short-term wind speed forecasts from neural network models. *Renewable Energy*, *36*(1), 352–359. https://doi.org/10.1016/j.renene.2010.06.049

Liu, C., Letchford, A. N., & Svetunkov, I. (2022). Newsvendor problems: An integrated method for estimation and optimisation. *European Journal of Operational Research*, *300*(2), 590–601. https://doi.org/10.1016/j.ejor.2021.08.013

López, K. L., Gagné, C., Castellanos-Dominguez, G., & Orozco-Alzate, M. (2015). Training subset selection in Hourly Ontario Energy Price forecasting using time series clustering-based stratification. *Neurocomputing*, *156*, 268–279. https://doi.org/10.1016/J.NEUCOM.2014.12.052

Lowalekar, H., Nilakantan, R., & Ravichandran, N. (2016). Analysis of an order-up-to-level policy for perishables with random issuing. *Journal of the Operational Research Society*, *67*(3), 483–505. https://doi.org/10.1057/jors.2015.59

Lu, C.-J., & Kao, L.-J. (2016). A clustering-based sales forecasting scheme by using extreme learning machine and ensembling linkage methods with applications to computer server. *Engineering Applications of Artificial Intelligence*, *55*, 231–238. https://doi.org/10.1016/J.ENGAPPAI.2016.06.015

Lu, Q., & Liu, N. (2013). Pricing games of mixed conventional and e-commerce distribution channels. *Computers and Industrial Engineering*, *64*(1), 122–132. https://doi.org/10.1016/j.cie.2012.09.018

Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2018). Statistical and Machine Learning forecasting methods: Concerns and ways forward. *PLOS ONE*, *13*(3), 1–26. https://doi.org/10.1371/journal.pone.0194889

Massaoudi, M., Refaat, S. S., Chihi, I., Trabelsi, M., Oueslati, F. S., & Abu-Rub, H. (2021). A novel stacked generalization ensemble-based hybrid LGBM-XGB-MLP model for Short-Term Load Forecasting. *Energy*, *214*. https://doi.org/10.1016/j.energy.2020.118874

McCarty, J. A., & Hastak, M. (2007). *Segmentation approaches in data-mining : A comparison of RFM , CHAID , and logistic regression*. *60*, 656–662. https://doi.org/10.1016/j.jbusres.2006.06.015

McDonald, M., Christopher, M., & Bass, M. (2003). *Market segmentation. In: Marketing*. Palgrave, London. https://doi.org/https://doi.org/10.1007/978-1-4039-3741-4_3

Mohammed, A., & Kora, R. (2023). A comprehensive review on ensemble deep learning: Opportunities and challenges. In *Journal of King Saud University - Computer and Information Sciences*. King Saud bin Abdulaziz University. https://doi.org/10.1016/j.jksuci.2023.01.014

Murray, P. W., Agard, B., & Barajas, M. A. (2015). Forecasting Supply Chain Demand by Clustering Customers. *IFAC-PapersOnLine*, *48*(3), 1834–1839. https://doi.org/10.1016/J.IFACOL.2015.06.353

Murray, P. W., Agard, B., & Barajas, M. A. (2017). Market segmentation through data mining: A method to extract behaviors from a noisy data set. *Computers and Industrial Engineering*, *109*, 233–252. https://doi.org/10.1016/j.cie.2017.04.017

Murray, P. W., Agard, B., & Barajas, M. A. (2018). Forecast of individual customer's demand from a large and noisy dataset. *Computers & Industrial Engineering*, *118*, 33–43. https://doi.org/10.1016/J.CIE.2018.02.007

Murray, P. W., Agard, B., Paul, W., Agard, B., Paul, W., Agard, B., Marco, A., & Hc, Q. C. (2015). Forecasting Supply Chain Demand by Clustering Customers. *IFAC-PapersOnLine*, *48*(3), 1834–1839. https://doi.org/10.1016/j.ifacol.2015.06.353

Nguyen, T., ZHOU, L., Spiegler, V., Ieromonachou, P., & Lin, Y. (2018). Big data analytics in supply chain management: A state-of-the-art literature review. *Computers & Operations Research*, *98*, 254–264. https://doi.org/10.1016/J.COR.2017.07.004

Nikolaev, N., Tino, P., & Smirnov, E. (2013). Time-dependent series variance learning with recurrent mixture density networks. *Neurocomputing*, *122*, 501–512. https://doi.org/10.1016/j.neucom.2013.05.014

Nilashi, M., Bagherifard, K., Rahmani, M., & Rafe, V. (2017). A recommender system for tourism industry using cluster ensemble and prediction machine learning techniques. *Computers and Industrial Engineering*, *109*, 357–368. https://doi.org/10.1016/j.cie.2017.05.016

Normandin-Taillon, H., Godin, F., & Wang, C. (2023). *LINEAR PRETRAINING IN RECURRENT MIXTURE DENSITY NETWORKS*.

Omar, H., Klibi, W., Babai, M. Z., & Ducq, Y. (2023). Basket data-driven approach for omnichannel demand forecasting. *International Journal of Production Economics*, *257*(December 2022), 108748. https://doi.org/10.1016/j.ijpe.2022.108748

Oroojlooyjadid, A., Snyder, L. V., & Takáč, M. (2020). Applying deep learning to the newsvendor problem. *IISE Transactions*, *52*(4), 444–463. https://doi.org/10.1080/24725854.2019.1632502

Pacheco, E. de O., Cannella, S., Lüders, R., & Barbosa-Povoa, A. P. (2017). Order-up-to-level policy update procedure for a supply chain subject to market demand uncertainty. *Computers and Industrial Engineering*, *113*, 347–355. https://doi.org/10.1016/j.cie.2017.09.015

Papageorgiou, K. I., Poczeta, K., Papageorgiou, E., Gerogiannis, V. C., & Stamoulis, G. (2019). Exploring an ensemble of methods that combines fuzzy cognitive maps and neural networks in solving the time series prediction problem of gas consumption in Greece. *Algorithms*, *12*(11), 1–27. https://doi.org/10.3390/a12110235

Parmezan, A. R. S., Souza, V. M. A., & Batista, G. E. A. P. A. (2019). Evaluation of statistical and machine learning models for time series prediction: Identifying the state-of-the-art and the best conditions for the use of each model. *Information Sciences*, *484*, 302–337. https://doi.org/10.1016/j.ins.2019.01.076

Pedregosa, F., VAROQUAUX, G., Gramfort, A., Michel, V., & Thirion, B. (2011). Scikit-learn: Machine Learning in Python. *Journal OfMachine Learning Research*, *12*, 2825–2830. https://doi.org/10.1289/EHP4713

Porras, E., & Dekker, R. (2008). An inventory control system for spare parts at a refinery: An empirical comparison of different re-order point methods. *European Journal of Operational Research*, *184*(1), 101–132. https://doi.org/10.1016/j.ejor.2006.11.008

Prak, D., & Teunter, R. (2019). A general method for addressing forecasting uncertainty in inventory models. *International Journal of Forecasting*, *35*(1), 224–238. https://doi.org/10.1016/j.ijforecast.2017.11.004

Prak, D., Teunter, R., & Syntetos, A. (2017). On the calculation of safety stocks when demand is forecasted. *European Journal of Operational Research*, *256*(2), 454–461. https://doi.org/10.1016/j.ejor.2016.06.035

Qu, T., Zhang, J. H., Chan, F. T. S., Srivastava, R. S., Tiwari, M. K., & Park, W. Y. (2017). Demand prediction and price optimization for semi-luxury supermarket segment. *Computers and Industrial Engineering*, *113*, 91–102. https://doi.org/10.1016/j.cie.2017.09.004

Rathipriya, R., Abdul Rahman, A. A., Dhamodharavadhani, S., Meero, A., & Yoganandan, G. (2023). Demand forecasting model for time-series pharmaceutical data using shallow and deep neural network model. *Neural Computing and Applications*, *35*(2), 1945–1957. https://doi.org/10.1007/s00521-022-07889-9

Raza, M. Q., Nadarajah, M., & Ekanayake, C. (2017). Demand forecast of PV integrated bioclimatic buildings using ensemble framework. *Applied Energy*, *208*(December), 1626–1638. https://doi.org/10.1016/j.apenergy.2017.08.192

Razavi, S. F., Hosseini, R., & Behzad, T. (2024). FRMDN: Flow-based Recurrent Mixture Density Network. *Expert Systems with Applications*, *237*, 121360. https://doi.org/10.1016/j.eswa.2023.121360

Rostami-Tabar, B., & Rendon-Sanchez, J. F. (2021). Forecasting COVID-19 daily cases using phone call data. *Applied Soft Computing*, *100*. https://doi.org/10.1016/j.asoc.2020.106932

Sagheer, A., & Kotb, M. (2019). Time series forecasting of petroleum production using deep LSTM recurrent networks. *Neurocomputing*, *323*, 203–213. https://doi.org/10.1016/j.neucom.2018.09.082

Salinas, D., Flunkert, V., Gasthaus, J., & Januschowski, T. (2020). DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, *36*(3), 1181–1191. https://doi.org/10.1016/j.ijforecast.2019.07.001

Schittenkopf, C., Dorffner, G., & Dockner, E. J. (2000). Forecasting time-dependent conditional densities: A semi-non-parametric neural network approach. *Journal of Forecasting*, *19*(4), 355–374. https://doi.org/10.1002/1099-131X(200007)19:4<355::AID-FOR778>3.0.CO;2-Z

Seyedan, M., & Mafakheri, F. (2020). Predictive big data analytics for supply chain demand forecasting: methods, applications, and research opportunities. *Journal of Big Data*, *7*(1), 53. https://doi.org/10.1186/s40537-020-00329-2

Seyedan, M., Mafakheri, F., & Wang, C. (2022). Cluster-based demand forecasting using Bayesian model averaging: An ensemble learning approach. *Decision Analytics Journal*, *3*, 100033. https://doi.org/10.1016/j.dajour.2022.100033

Seyedan, M., Mafakheri, F., & Wang, C. (2023). Order-up-to-level inventory optimization model using time-series demand forecasting with ensemble deep learning. *Supply Chain Analytics*, *3*, 100024. https://doi.org/10.1016/j.sca.2023.100024

Shirsat, A., & Tang, W. (2021). Quantifying residential demand response potential using a mixture density recurrent neural network. *International Journal of Electrical Power and Energy Systems*, *130*(September 2020), 106853. https://doi.org/10.1016/j.ijepes.2021.106853

Sillanpää, V., & Liesiö, J. (2018). Forecasting replenishment orders in retail: value of modelling low and intermittent consumer demand with distributions. *International Journal of Production Research*, *56*(12), 4168–4185. https://doi.org/10.1080/00207543.2018.1431413

Singh, S. K., Bejagam, K. K., An, Y., & Deshmukh, S. A. (2019). Machine-Learning Based Stacked Ensemble Model for Accurate Analysis of Molecular Dynamics Simulations. *The Journal of Physical Chemistry A*, *123*(24), 5190–5198. https://doi.org/10.1021/acs.jpca.9b03420

Song, G., & Dai, Q. (2017). A novel double deep ELMs ensemble system for time series forecasting. *Knowledge-Based Systems*, *134*, 31–49. https://doi.org/10.1016/j.knosys.2017.07.014

Ŝtepnicka, M., Peralta, J., Cortez, P., Vavrícková, L., & Gutierrez, G. (2011). Forecasting seasonal time series with computational intelligence: Contribution of a combination of distinct methods. *Proceedings of the 7th Conference of the European Society for Fuzzy Logic and Technology, EUSFLAT 2011 and French Days on Fuzzy Logic and Applications, LFA 2011*, *1*(1), 464–471. https://doi.org/10.2991/eusflat.2011.7

Swaminathan, K., & Venkitasubramony, R. (2023). Demand forecasting for fashion products: A systematic review. *International Journal of Forecasting*, *xxxx*. https://doi.org/10.1016/j.ijforecast.2023.02.005

Tan, A. T. (2021, October 13). *Stacking Machine Learning Models for Multivariate Time Series*. Towards Data Science. https://towardsdatascience.com/stacking-machine-learning-models-for-multivariate-time-series-28a082f881

Tarim, S. A., & Kingsman, B. G. (2006). Modelling and computing (Rn, Sn) policies for inventory systems with non-stationary stochastic demand. *European Journal of Operational Research*, *174*(1), 581–599. https://doi.org/10.1016/j.ejor.2005.01.053

Taylor, S., & Letham, B. (2018). Forecasting at scale. *The AmericanStatisticia*, *72*(1), 37–45. https://doi.org/https://doi.org/10.1080/00031305.2017.1380080

Thomopoulos, N. T. (2015). Demand Forecasting for Inventory Control. In *Hospital Materiel Management Quarterly* (Vol. 4, Issue 2). Springer International Publishing. https://doi.org/10.1007/978-3-319-11976-2

Trapero, J. R., Cardós, M., & Kourentzes, N. (2019a). Empirical safety stock estimation based on kernel and GARCH models. *Omega (United Kingdom)*, *84*, 199–211. https://doi.org/10.1016/j.omega.2018.05.004

Trapero, J. R., Cardós, M., & Kourentzes, N. (2019b). Quantile forecast optimal combination to enhance safety stock estimation. *International Journal of Forecasting*, *35*(1), 239–250. https://doi.org/10.1016/j.ijforecast.2018.05.009

Turrado García, F., García Villalba, L. J., & Portela, J. (2012). Intelligent system for time series classification using support vector machines applied to supply-chain. *Expert Systems with Applications*, *39*(12), 10590–10599. https://doi.org/10.1016/J.ESWA.2012.02.137

van Steenbergen, R. M., & Mes, M. R. K. (2020). Forecasting demand profiles of new products. *Decision Support Systems*, *139*(September), 113401. https://doi.org/10.1016/j.dss.2020.113401

Varghese, R., Mohanty, A., Tyagi, S., Mishra, S., Kumar, S., & Nandan, S. (2022). A hybrid deep learning framework with CNN and Bi-directional LSTM for store item demand forecasting. *Computers and Electrical Engineering*, *103*(August), 108358. https://doi.org/10.1016/j.compeleceng.2022.108358

Venkatesh, K., Ravi, V., Prinzie, A., & Van Den Poel, D. (2014). Cash demand forecasting in ATMs by clustering and neural networks. *European Journal of Operational Research*, *232*(2), 383–392. https://doi.org/10.1016/j.ejor.2013.07.027

Vossen, J., Feron, B., & Monti, A. (2018, August 17). Probabilistic forecasting of household electrical load using artificial neural networks. *2018 International Conference on*

Probabilistic Methods Applied to Power Systems, PMAPS 2018 - Proceedings. https://doi.org/10.1109/PMAPS.2018.8440559

Wang, G., Gunasekaran, A., Ngai, E. W. T., & Papadopoulos, T. (2016). Big data analytics in logistics and supply chain management: Certain investigations for research and applications. *International Journal of Production Economics*, *176*, 98–110. https://doi.org/10.1016/J.IJPE.2016.03.014

Wang, J., Cevik, M., & Bodur, M. (2022). On the impact of deep learning-based time-series forecasts on multistage stochastic programming policies. *INFOR: Information Systems and Operational Research*, *60*(2), 133–164. https://doi.org/10.1080/03155986.2021.2015825

Wang, Q., Xu, W., Huang, X., & Yang, K. (2019). Enhancing intraday stock price manipulation detection by leveraging recurrent neural networks with ensemble learning. *Neurocomputing*, *347*, 46–58. https://doi.org/10.1016/j.neucom.2019.03.006

Wei, J., Lin, S., Weng, C., & Wu, H. (2012). Expert Systems with Applications A case study of applying LRFM model in market segmentation of a children ' s dental clinic. *Expert Systems With Applications*, *39*(5), 5529–5533. https://doi.org/10.1016/j.eswa.2011.11.066

Willemain, T. R., Smart, C. N., & Schwarz, H. F. (2004). A new approach to forecasting intermittent demand for service parts inventories. *International Journal of Forecasting*, *20*(3), 375–387. https://doi.org/10.1016/S0169-2070(03)00013-X

Yang, A. X. (2004). How to develop new approaches to RFM segmentation. *Journal of Targeting, Measurement and Analysis for Marketing*, *13*(1), 50–60. https://doi.org/10.1057/palgrave.jt.5740131

Yang, M., Lim, M. K., Qu, Y., Li, X., & Ni, D. (2023). Deep neural networks with L1 and L2 regularization for high dimensional corporate credit risk prediction. *Expert Systems with Applications*, *213*(December 2021). https://doi.org/10.1016/j.eswa.2022.118873

Yang, Y., Lv, H., & Chen, N. (2023). A Survey on ensemble learning under the era of deep learning. *Artificial Intelligence Review*, *56*(6), 5545–5589. https://doi.org/10.1007/s10462-022-10283-5

Yu, Q., Wang, K., Strandhagen, J. O., & Wang, Y. (2018). Application of Long Short-Term Memory Neural Network to Sales Forecasting in Retail—A Case Study. *Lecture Notes in Electrical Engineering*, *451*, 11–17. https://doi.org/10.1007/978-981-10-5768-7_2

Zhang, S., Chen, Y., Zhang, W., & Feng, R. (2021). A novel ensemble deep learning model with dynamic error correction and multi-objective ensemble pruning for time series forecasting. *Information Sciences*, *544*, 427–445. https://doi.org/10.1016/j.ins.2020.08.053

Zhou, C., & Viswanathan, S. (2011). Comparison of a new bootstrapping method with parametric approaches for safety stock determination in service parts inventory systems. *International Journal of Production Economics*, *133*(1), 481–485. https://doi.org/10.1016/j.ijpe.2010.09.021

Zhou, J., Peng, T., Zhang, C., & Sun, N. (2018). Data pre-analysis and ensemble of various artificial neural networks for monthly streamflow forecasting. *Water (Switzerland)*, *10*(5). https://doi.org/10.3390/W10050628

Zhou, Y., Chang, F. J., Chen, H., & Li, H. (2020). Exploring Copula-based Bayesian Model Averaging with multiple ANNs for PM2.5 ensemble forecasts. *Journal of Cleaner Production*, *263*, 121528. https://doi.org/10.1016/j.jclepro.2020.121528

Zhou, Z. H., Wu, J., & Tang, W. (2002). Ensembling neural networks: Many could be better than all. *Artificial Intelligence*, *137*(1–2), 239–263. https://doi.org/10.1016/j.artint.2010.10.001