machine learning & knowledge extraction

MDPI

Article

Enhancing Hierarchical Sales Forecasting with Promotional Data: A Comparative Study Using ARIMA and Deep Neural Networks

Mariana Teixeira [1,†], José Manuel Oliveira [1,2,*,†] and Patrícia Ramos [2,3,†]

[1] Faculty of Economics, University of Porto, Rua Dr. Roberto Frias, 4200-464 Porto, Portugal; marianacteixeira@outlook.com
[2] Institute for Systems and Computer Engineering, Technology and Science, Campus da FEUP, Rua Dr. Roberto Frias, 4200-465 Porto, Portugal; patricia@iscap.ipp.pt
[3] CEOS.PP, ISCAP, Polytechnic of Porto, Rua Jaime Lopes Amorim s/n, 4465-004 Porto, Portugal
* Correspondence: jmo@fep.up.pt
† These authors contributed equally to this work.

Abstract: Retailers depend on accurate sales forecasts to effectively plan operations and manage supply chains. These forecasts are needed across various levels of aggregation, making hierarchical forecasting methods essential for the retail industry. As competition intensifies, the use of promotions has become a widespread strategy, significantly impacting consumer purchasing behavior. This study seeks to improve forecast accuracy by incorporating promotional data into hierarchical forecasting models. Using a sales dataset from a major Portuguese retailer, base forecasts are generated for different hierarchical levels using ARIMA models and Multi-Layer Perceptron (MLP) neural networks. Reconciliation methods including bottom-up, top-down, and optimal reconciliation with OLS and WLS (struct) estimators are employed. The results show that MLPs outperform ARIMA models for forecast horizons longer than one day. While the addition of regressors enhances ARIMA's accuracy, it does not yield similar improvements for MLP. MLPs present a compelling balance of simplicity and efficiency, outperforming ARIMA in flexibility while offering faster training times and lower computational demands compared to more complex deep learning models, making them highly suitable for practical retail forecasting applications.

Keywords: hierarchical forecasting; deep learning; multi-layer perceptrons; ARIMAX; supply chain management; promotions

Citation: Teixeira, M.; Oliveira, J.M.; Ramos, P. Enhancing Hierarchical Sales Forecasting with Promotional Data: A Comparative Study Using ARIMA and Deep Neural Networks. *Mach. Learn. Knowl. Extr.* **2024**, *6*, 2659–2687. https://doi.org/10.3390/make6040128

Academic Editor: Roozbeh Razavi-Far

Received: 13 September 2024
Revised: 3 November 2024
Accepted: 16 November 2024
Published: 19 November 2024

1. Introduction

The increasing complexity of modern supply chains underscores the critical role of accurate sales forecasting in informing retail planning and decision making. As highlighted by Villegas and Pedregal [1], accurate forecasts are essential for optimizing supply chain efficiency. Even minor forecasting errors can significantly impact retailers' revenue due to their substantial sales volumes. Retailers must carefully select forecasting methods that balance accuracy with practical considerations such as ease of use and computational demands. By mitigating stockouts, excess inventory, and product waste, accurate forecasts contribute significantly to cost reduction and customer satisfaction. As Fildes et al. [2] emphasize, robust forecasting is key to safeguarding against lost sales due to out-of-stock situations and the potential erosion of customer loyalty.

Given these challenges, this study aims to enhance sales forecasting accuracy by integrating promotional activities into hierarchical forecasting methods. The research specifically investigates how the inclusion of promotional data can improve forecast accuracy across multiple levels of aggregation, such as SKUs, stores, and distribution centers. This focus addresses a critical gap in the literature, as the impact of promotions on hierarchical sales forecasting remains underexplored. By doing so, this study seeks to provide a more

coherent and accurate forecasting system for retailers, which is essential for optimizing supply chain decisions and performance.

Retailers require sales forecasts at multiple organizational levels (SKUs, stores, and distribution centers) to inform effective decision making [3,4]. Hierarchical forecasting addresses this need by generating forecasts for different levels of aggregation. While independent generation of forecasts for each level is possible, it fails to capture hierarchical relationships and may lead to inconsistencies between aggregated and disaggregated forecasts. Maintaining forecast consistency across levels is crucial for coherent supply chain management. Reconciliation strategies offer a solution by aligning forecasts at different levels, potentially enhancing overall forecast accuracy. The optimal reconciliation approach depends on factors such as time-series characteristics, hierarchical structure, and forecast horizon.

Intensified competition within the retail sector has led to a surge in promotional activities, highlighting the need for forecasting models that accurately capture their impact on sales [5]. Incorporating promotional information into sales forecasts has been shown to enhance predictive accuracy [6,7], but its specific impact on hierarchical forecasting has not been extensively studied. Therefore, this research focuses on exploring how integrating promotional data can improve forecast accuracy within a hierarchical framework.

To achieve this, the study initially defines the most suitable hierarchical structure for data aggregation based on the retailer's needs. Base forecasts are then generated using traditional Autoregressive Integrated Moving Average (ARIMA) models and Multi-Layer Perceptrons (MLPs) across various hierarchical levels, while bottom-up, top-down, and optimal reconciliation strategies with OLS and WLS (struct) estimators are considered as reconciliation procedures. Additional factors, such as prices, day of the week, and calendar events, are integrated to further refine accuracy. The motivation for using MLPs stems from recent studies suggesting that deep learning methods may outperform traditional models. Although their potential is recognized, their application in sales forecasting is not yet well explored, indicating a need for further investigation to understand their comparative advantages. This study leverages real sales data from Pingo Doce stores provided by the Jerónimo Martins Group to support its analysis.

The main contributions of this research are highlighted as follows:

- This work contributes to the field by incorporating promotional data into hierarchical sales forecasting models. This addresses a gap in the existing literature and provides valuable insights into the impact of promotions on forecasting accuracy.
- This study compares the performance of traditional ARIMA models with more advanced MLP models, providing insights into their relative strengths and weaknesses in the context of hierarchical forecasting.
- This paper assesses the effectiveness of different reconciliation methods (bottom-up, top-down, and optimal reconciliation) in improving forecast accuracy and consistency across hierarchical levels.
- By utilizing a real-world dataset, this paper demonstrates the practical application of the proposed methodology and its potential benefits for retailers.
- This study contributes to the ongoing development of hierarchical forecasting methods by exploring the integration of additional variables and the application of advanced modeling techniques.

This paper is structured as follows. Following this Introduction, Section 2 provides a literature review, outlining key concepts and establishing a theoretical framework based on existing literature. Section 3 details the methodology used to address the study's objectives. Section 4 presents a case study conducted with real data from a Portuguese retailer and discusses the obtained results. Finally, Section 5 summarizes the main findings, addresses potential limitations of the study, and suggests areas for future research.

2. Related Work

2.1. Sales Forecasting in the Retail Sector

Sales forecasting involves estimating future sales values based on historical data and variables that may influence them. Given the uncertainty associated with demand, sales forecasting is essential for retailers to efficiently plan the distribution and restocking of their products in stores. Accurate forecasts help prevent profit reductions caused by inefficient stock management [8]. Typically, the sales forecasting process utilizes a Forecasting Support System (FSS), which generates initial forecasts that are subsequently refined by the company's demand management team. These adjustments account for factors not included in the initial forecast calculations [9]. However, it is essential to evaluate whether these adjustments improve forecast accuracy or, instead, introduce bias [10,11]. Moreover, the vast scale of product-level sales forecasting in the retail sector underscores the need for greater computational efficiency and speed [12,13], while manual adjustments of system-generated forecasts become increasingly impractical. Consequently, it is becoming imperative for forecasting processes to incorporate additional information, such as promotional activities, to reduce errors and the need for manual adjustments [7,14]. While each participant in the supply chain generally operates their FSS independently, collaborative schemes exist that enable the sharing of sales information between suppliers and retailers. Such partnerships are primarily established to reduce costs and manage inventory more effectively [15,16]. Several studies suggest that information sharing enhances forecast accuracy [17,18], which can help mitigate the bullwhip effect. As defined by Lee et al. [19], the bullwhip effect refers to the distortion of demand information along the supply chain, occurring when the variation in retailer sales is less than the variation in retailer orders to suppliers. This effect can lead to excessive raw material inventory; additional production, storage, and distribution costs due to overcapacity; and a decline in customer service quality [20]. Retailers typically rely on observed sales data to forecast future demand. However, when supply shortages result in stockouts, the true demand is not fully captured, potentially leading to a negative bias in future demand forecasts for the affected products [21]. Additionally, stockouts may drive customers toward substitute products, creating a positive bias in demand forecasts for those substitutes [22]. Kim et al. [23] highlight that stockouts complicate the demand estimation process for substitute products, demonstrating the intricacies involved in accurate demand forecasting. To mitigate the risk of stockouts, retailers often maintain safety stock [24]. The size of this stock is influenced by the level of demand uncertainty and the associated forecast errors [25]. Balancing the cost of maintaining safety stock against potential revenue losses due to stockouts remains a significant challenge for retailers. Another challenge in sales forecasting is predicting demand for new products, which lack historical sales data. This challenge is compounded by potential cannibalization effects, where the introduction of new products increases their sales at the expense of substitute products [26]. The most common strategy for forecasting new product demand relies on historical data from similar products. However, Kahn [27] emphasizes the importance of distinguishing between forecasting for existing products and new products, with the latter requiring qualitative analysis involving expert judgments and assumptions. In addition to historical sales data, retailers are increasingly leveraging information about customers and competitors' products and pricing to make more informed decisions and innovate their business models [28]. A deeper understanding of customer purchasing behaviors and preferences can significantly improve the accuracy of demand forecasting [29]. Recent studies have shown that incorporating new information sources, such as product reviews, online searches, and social media activity, can enhance sales forecasting performance [30–33].

Among the various forecasting approaches, the autoregressive integrated moving average (ARIMA) models developed by Box et al. [34] are widely utilized. These univariate models rely solely on time-series data for forecasting. In addition to producing forecasts, ARIMA models can generate prediction intervals, which are particularly valuable in the retail sector for determining appropriate safety stock levels. Deep Learning methods offer a promising alternative due to their ability to process large volumes of

data rapidly and create complex data representations [35]. Alon et al. [36] explored the use of Artificial Neural Networks (ANNs) for sales forecasting in the retail sector and demonstrated that these networks can capture nonlinear trends and seasonality patterns, often outperforming traditional statistical methods like ARIMA, particularly in volatile economic conditions. However, some studies suggest that without adequate preprocessing to account for trends and seasonality in sales data, ARIMA models may outperform Neural Networks (NNs) [37,38]. These findings align with the conclusions of Nelson et al. [39], who observed that when seasonality is pre-adjusted, NNs produce more accurate forecasts. Contrarily, Aras et al. [40] found no significant performance differences between ANNs and ARIMA models in sales forecasting. Aburto and Weber [41] developed a hybrid forecasting model that combines ARIMA with NNs, showing improvements in forecast accuracy compared to using either model individually. Given the mixed results regarding the performance of traditional ANNs, more advanced forecasting methods have recently gained attention. Vallés-Pérez et al. [42] demonstrated that Recurrent Neural Networks (RNNs) can be effectively used to forecast sales at the store or product level in the retail sector. Long Short-Term Memory (LSTM) networks, a subclass of RNNs introduced by Hochreiter and Schmidhuber [43], are particularly well-suited for learning long-term dependencies in time-series data, making them more effective for sales forecasting compared to traditional RNNs, which struggle with long-term connections. Studies by Abbasimehr et al. [44] and Ensafi et al. [45] have shown that multi-layered LSTM models outperform machine Learning methods and traditional forecasting models in terms of accuracy. However, Falatouri et al. [46] found that while the seasonal ARIMA model performs better for products with seasonal demand patterns, LSTM models are more effective for forecasting products with stable demand. The literature also includes hybrid models that combine LSTM models with machine Learning techniques [47,48]. Wang et al. [49,50] observed that although RNNs and LSTM models generally offer superior generalization and accuracy, ARIMA models have the advantage of shorter execution time and lower processing costs. Convolutional Neural Networks (CNNs), initially proposed by LeCun et al. [51] for image classification, have also been adapted for time-series forecasting due to their pattern recognition capabilities. Ma and Fildes [52] proposed a sales forecasting strategy based on CNNs, which improves the accuracy of product-level sales forecasts compared to a wide range of alternative methods. Additionally, hybrid approaches combining CNNs and LSTM models have been developed to capitalize on the strengths of both techniques [53–55]. In the context of online retail, Bandara et al. [56–58] developed an LSTM-based sales forecasting strategy that incorporates information about correlations between product demand patterns. Applied to Walmart's dataset, this approach outperformed traditional univariate models. Pan and Zhou [59] utilized CNNs for online sales forecasting based on data from Alibaba, including sales history and variables such as the number of searches and product views. Their method yielded better results than ARIMA models. Similarly, Chen [60] and He et al. [61] applied LSTM models to the same online retailer sales dataset, finding that these models reduced forecasting errors compared to other methods. In recent years, transformer models have emerged as a powerful approach for various tasks in time-series forecasting, including sales forecasting. Transformers, originally developed for natural language processing, leverage self-attention mechanisms to capture long-range dependencies in data more effectively than traditional methods, including RNNs and LSTM models [62]. Their architecture allows for parallel processing of input data, resulting in faster training times and improved performance on large datasets, making them well-suited for complex forecasting tasks in retail environments. Studies such as those by Wu et al. [63], Lim et al. [64], Zhou et al. [65,66], Nie et al. [67], and Tong et al. [68] have demonstrated the effectiveness of transformer models in forecasting applications, showing that they can outperform conventional methods by effectively modeling intricate relationships within the data. Transformers can also easily incorporate additional input features, such as promotional events and consumer sentiment data, enhancing their forecasting capabilities [69]. However, one significant drawback of transformer models is their sub-

stantial demand for computational resources, which can limit their accessibility for many retailers. The architecture's complexity requires significant memory and processing power, particularly when training on large datasets or incorporating multiple features [70]. This resource-intensive nature may deter smaller retailers from adopting transformer-based approaches, making it crucial to balance the model's benefits with the practical constraints of computational infrastructure. Recent advancements in artificial intelligence have led to the development of foundation models, which are large-scale models trained on massive datasets [71–75]. These models have demonstrated impressive performance in various tasks, including natural language processing, computer vision, and time-series forecasting. While foundation models offer significant potential for improving the accuracy and efficiency of time-series forecasting, their application in the retail domain is still in its early stages. Further research is needed to explore the potential benefits and challenges of leveraging foundation models for retail sales forecasting.

2.2. Hierarchical Forecasting

Supply chain management encompasses the coordination of actions among various participants, from producers to end consumers, with the primary objective of meeting consumer demand [3]. In the retail sector, sales forecasting is critical at different levels of aggregation, facilitating informed decision making throughout the supply chain [76]. Hierarchical forecasting has been extensively explored within this context, operating on the premise that time series can be disaggregated across various dimensions relevant to retailers, such as time intervals (temporal hierarchy) and product categories (cross-sectional hierarchy). When considering temporal aggregation, higher-level aggregates typically exhibit components like trend and cyclicality, while lower levels tend to display seasonal patterns. Kourentzes et al. [77] introduced the Multiple Aggregation Prediction Algorithm, a forecasting framework that employs temporal aggregation to model the distinct components of time series. Athanasopoulos et al. [78] further demonstrated that forecasting using temporal hierarchies provides superior results compared to traditional methods. For cross-sectional hierarchies, retailers must determine the appropriate level of aggregation for sales forecasts, considering three primary axes: product (SKU, category, and area), location (store, distribution center, region, and country), and time (day, week, month, and year) [2]. The choice of aggregation level significantly influences forecast accuracy [79]. The aim is to generate a consistent set of forecasts that align with the hierarchical structure, ensuring that the sum of forecasts at disaggregated levels equals the forecast at the corresponding aggregated level. To achieve this, various strategies are employed to reconcile base forecasts generated for different levels of the hierarchy. The three main reconciliation strategies are the bottom-up method, which involves generating forecasts at the most disaggregated level and aggregating them to obtain higher-level forecasts; the top-down method, which starts with forecasts at the most aggregated level and disaggregates them based on appropriate proportions; and the middle-out method, an intermediate approach between the two [80]. These methods utilize only a portion of the available information, as they generate forecasts for a single aggregation level. The bottom-up method, for example, does not account for correlations between time series, leading to suboptimal outcomes at higher aggregation levels. Conversely, the top-down method loses valuable information due to data aggregation, resulting in lower accuracy at lower levels of the hierarchy. To address this issue, Athanasopoulos et al. [81] proposed a top-down method that uses forecast-based proportions rather than historical proportions [82] to disaggregate forecasts generated at the highest level. However, Hyndman et al. [83] noted that none of the three basic reconciliation strategies adequately consider correlations between hierarchical levels. In response, they proposed an approach that involves independently generating forecasts for each hierarchical level, followed by a reconciliation process that aligns these forecasts with the aggregation structure. The reconciliation method used in their study was Ordinary Least Squares (OLS), and the results indicate that this approach outperforms traditional reconciliation methods. Hyndman et al. [84] later suggested the use of Weighted

Least Squares (WLS) to reconcile base forecasts, recommending algorithms to facilitate reconciliation in cases involving large numbers of time series. Wickramasuriya et al. [85] critiqued the practical feasibility of Hyndman et al. [83]'s approach, introducing the Minimum Trace (MinT) optimal reconciliation method as an alternative. Spiliotis et al. [86] developed a reconciliation strategy based on machine learning methods, enabling nonlinear combinations of base forecasts. This new approach consistently delivers better results in terms of forecast accuracy and bias compared to the basic and linear reconciliation strategies developed by Hyndman et al. [83,84] and Wickramasuriya et al. [85]. Pennings and van Dalen [87] presented an integrated hierarchical forecasting strategy based on the estimation of a multivariate state-space model with a Kalman filter. This approach considers the complementary and substitution relationships between products in sales forecasting, incorporating all available information and generating forecasts for all levels of aggregation while respecting the hierarchical structure. This method effectively overcomes the limitations of both bottom-up and top-down approaches. Villegas and Pedregal [1] offered a hierarchical forecasting approach grounded in state-space models, emphasizing that it ensures the consistency of forecasts across time. Despite the progress in hierarchical forecasting, there remains uncertainty regarding whether forecasts should be obtained directly for each hierarchical level or generated through hierarchical forecasting methods [88]. This question is particularly relevant when the forecast includes regressors, such as promotional activities, underscoring the need to explore which approach, when combined with regressors, enhances forecasting accuracy. Additionally, questions persist about the most suitable forecasting models for each hierarchical level and which reconciliation strategies produce the best results.

2.3. Determinants in Retail Product Sales

Customer purchasing decisions are influenced by a range of factors, including price, promotions, calendar events (such as holidays and festive seasons), weather conditions, and seasonality. To manage inventory efficiently and ensure high levels of customer satisfaction, retail sales forecasting processes must consider not only historical sales data but also these key drivers of product demand. Although numerous studies on the subject have been conducted, weather conditions remain underutilized in sales forecasting due to the inherent uncertainty of weather forecasts. For example, Divakar et al. [89] developed a sales forecasting model for beverages that incorporates variables such as prices, promotions, average temperature, calendar events, and new product launches. The rationale for including temperature forecasts lies in the increased demand for beverages as temperatures rise. Similarly, Ramanathan and Muyldermans [90] examined the factors affecting soft drink demand, including promotions, calendar events, and weather conditions, but found that promotions were the only factor consistently influencing demand across all products. In the context of online retail, Steinker et al. [91] demonstrated that integrating weather information into the forecasting process significantly reduces forecast errors, particularly for weekends and days with favorable weather. Additionally, Liu and Ichise [92] introduced a deep learning method that effectively forecasts beverage sales for a supermarket chain using weather data, outperforming traditional machine learning methods. Hirche et al. [93] employed non-seasonal Autoregressive Integrated Moving Average with exogenous variables (ARIMAX) models, incorporating temperature data and calendar events to forecast alcoholic beverage sales. Their findings indicate that the sensitivity of sales sensitivity temperature varies by region and beverage category, while festive seasons impact sales across all beverage categories. Verstraete et al. [94] proposed a methodology that accounts for the uncertainty of short-term and long-term weather forecasts when predicting retail product sales. However, Badorf and Hoberg [95] showed that the benefits of incorporating weather forecasts into sales predictions diminish as the forecast horizon extends. Based on data from a beverage company, Ramanathan and Muyldermans [96] found that promotional information and seasonal factors significantly influence sales, while the impact of calendar events is limited. Most models that incorporate sales determinants emphasize

the importance of promotional information due to the high frequency of retail promotions and their significant effect on customer behavior. For example, Özden Gür Ali et al. [97] demonstrated that simple forecasting methods perform well in the absence of promotions, but during promotional periods, more sophisticated methods that include additional promotional data significantly improve forecast accuracy. Arunraj and Ahrens [98] presented two linear regression models that integrate forecasts from a seasonal ARIMA model with variables such as promotions, calendar events, monthly seasonality, and weather conditions. These models outperformed the seasonal naïve method, the seasonal ARIMA model without regressor integration, and multi-layer perceptron. In a similar vein, Arunraj et al. [99] developed a seasonal ARIMAX model that includes promotions and calendar events for daily sales forecasting in retail stores, showing improved forecast accuracy over the traditional seasonal ARIMA model. Abolghasemi et al. [6] investigated forecasting demand series characterized by volatility due to promotional activities, proposing a hybrid strategy that combines the ARIMA model for non-promotional periods with segmented regression to predict demand spikes caused by promotions. This approach results in fewer forecast errors when demand volatility is high, while the ARIMAX model performs better in cases of low to moderate demand volatility. Abolghasemi et al. [7] further developed a demand forecasting model that integrates the effects of systematic events, such as promotions, to enhance forecast accuracy and reduce the need for manual adjustments. Huber and Stuckenschmidt [100] applied machine learning methods to sales forecasting based on external information related to calendar events, with results confirming the effectiveness of these methods and the decreasing necessity of manual adjustments after forecasts are generated. Overall, multivariate forecasting models have generally outperformed traditional models. However, integrating sales determinants into the forecasting process necessitates the use of more advanced forecasting methods and variable selection strategies to manage the complexity of numerous explanatory variables and data heterogeneity. For instance, Guo et al. [101] developed a multivariate forecasting model based on neural networks that incorporates a Harmony Search wrapper-based variable selection approach to identify the most relevant input variables for sales forecasting. The authors demonstrated that this variable selection strategy effectively reduces the number of model parameters, leading to greater forecast accuracy. Huang et al. [102] proposed a sales forecasting model at the SKU level, beginning with the selection of competitive explanatory variables, such as prices and promotions, and subsequently integrating this information into an Autoregressive distributed lag model. This model outperformed the Simple Exponential Smoothing (SES) method, even when SES forecasts were adjusted for past promotional effects. Ma et al. [103] used a model similar to that proposed by Huang et al. [102] but also accounted for promotional interactions between products from different categories, as well as those within the same category. To select relevant explanatory variables, they employed a more practical strategy based on least absolute shrinkage and selection operator regression. Trapero et al. [104] developed a multiple regression model that incorporates promotional variables and addresses challenges associated with this type of model, such as the high number of variables and multicollinearity, through the use of Principal Component Analysis. The authors observed that their proposed model generated more accurate forecasts than the naïve method, the SES method, and the multivariate last like promotion method developed by Özden Gür Ali et al. [97].

3. Forecasting Models

3.1. Hierarchical Forecasting

To illustrate a hierarchical structure, consider the example shown in Figure 1. At the top of the hierarchy (level 0) is the most aggregated series, labeled as *Total*. The *Total* series is then disaggregated into two series, A and B, which constitute level 1. Each of these series is further disaggregated into three series at the bottom level of the hierarchy (level 2). The value observed at time t for series i is represented as $y_{i,t}$, where $t = 1, \ldots, T$. For each

time period (t), n represents the total number of series, while m denotes the number of bottom-level series. In the example provided in Figure 1, $n = 9$ and $m = 6$.

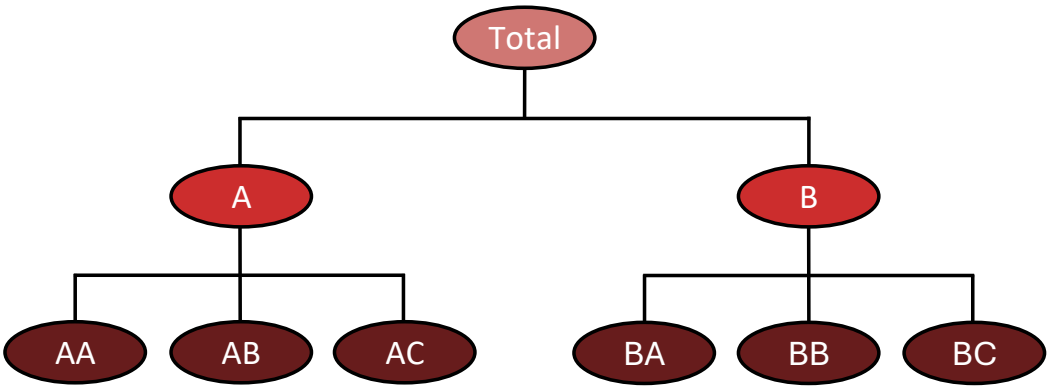

Figure 1. A three-level hierarchical structure.

In this case, at each time (t), the observations add up according to the following aggregation constraints:

$$\begin{aligned} y_{Total,t} &= y_{AA,t} + y_{AB,t} + y_{AC,t} + y_{BA,t} + y_{BB,t} + y_{BC,t}, \\ y_{A,t} &= y_{AA,t} + y_{AB,t} + y_{AC,t}, y_{B,t} = y_{BA,t} + y_{BB,t} + y_{BC,t}. \end{aligned} \tag{1}$$

Let y_t denote the vector containing the t-th observations of all series in the hierarchy, and let b_t represent the vector containing only the t-th observations of the bottom-level series. $\boldsymbol{S}$ is defined as the summing matrix of order $n \times m$, which reflects how the bottom-level series aggregate to the higher levels. Thus, the aggregation constraints can be expressed using matrix notation as follows [105]:

$$\boldsymbol{y}_t = \boldsymbol{S}\boldsymbol{b}_t. \tag{2}$$

For the scenario depicted in Figure 1, Equation (2) can be expressed as follows:

$$\begin{bmatrix} y_{Total,t} \\ y_{A,t} \\ y_{B,t} \\ y_{AA,t} \\ y_{AB,t} \\ y_{AC,t} \\ y_{BA,t} \\ y_{BB,t} \\ y_{BC,t} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} y_{AA,t} \\ y_{AB,t} \\ y_{AC,t} \\ y_{BA,t} \\ y_{BB,t} \\ y_{BC,t} \end{bmatrix}. \tag{3}$$

The goal is to produce coherent forecasts for each series within the hierarchy, ensuring that these forecasts align with the hierarchical structure and satisfy the aggregation constraints.

Let $\hat{\boldsymbol{y}}_{t+h|t}$ represent a vector containing the h-step-ahead forecasts (where $h = 1, 2, \ldots$) for all series in the hierarchy, generated based on observations up to and including time t. While any forecasting model can be used to independently generate these base forecasts, this approach is unlikely to ensure that the aggregation constraints are satisfied. To achieve coherence among the forecasts, a reconciliation method must be applied:

$$\tilde{\boldsymbol{y}}_{t+h|t} = \boldsymbol{S}\boldsymbol{P}\hat{\boldsymbol{y}}_{t+h|t}, \tag{4}$$

where $\boldsymbol{P}$ is an $m \times n$ matrix that maps the base forecasts ($\hat{\boldsymbol{y}}t + h|t$) into reconciled forecasts at the bottom level. These forecasts are then aggregated using the summing matrix ($\boldsymbol{S}$) to produce coherent forecasts ($\tilde{\boldsymbol{y}}t + h|t$). Matrix $\boldsymbol{P}$ depends on the chosen reconciliation

method. For the bottom-up method, $\boldsymbol{P} = \left[\mathbf{0}_{m\times(n-m)} | \boldsymbol{I}_m\right]$, where $\mathbf{0}_{m\times(n-m)}$ is the null matrix of order $m \times (n-m)$ and $\boldsymbol{I}_m$ is the identity matrix of order m. For the hierarchical structure illustrated in Figure 1, matrix $\boldsymbol{P}$ is expressed as follows:

$$\boldsymbol{P} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5}$$

For the top-down method, $\boldsymbol{P} = \left[\boldsymbol{p} | \mathbf{0}_{m\times(n-1)}\right]$, where $\boldsymbol{p}$ is a vector of dimension m that contains the proportions used to disaggregate the top-level forecast into forecasts at the bottom level. For the hierarchical structure depicted in Figure 1, matrix $\boldsymbol{P}$ is expressed as follows:

$$\boldsymbol{P} = \begin{bmatrix} p_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{6}$$

Traditional methods for calculating disaggregation proportions, as presented by Gross and Sohl [82], rely on historical data. In the first approach, each proportion (p_i) is computed as the average of the historical ratios of the bottom-level series ($y_{i,t}$) to that of the top-level series ($y_{Total,t}$) over the period of $t = 1, \ldots, T$:

$$p_i = \frac{1}{T} \sum_{t=1}^{T} \frac{y_{i,t}}{y_{Total,t}}, \quad i = 1, \ldots, m. \tag{7}$$

In the second approach, each proportion (p_i) is calculated as the ratio of the average historical values of the bottom-level series ($y_{i,t}$) to the average historical values of the top-level series ($y_{Total,t}$) over the same period:

$$p_i = \frac{\frac{1}{T}\sum_{t=1}^{T} y_{i,t}}{\frac{1}{T}\sum_{t=1}^{T} y_{Total,t}}, \quad i = 1, \ldots, m. \tag{8}$$

These top-down approaches are valued for their simplicity, but they are static and do not account for potential variations in proportions over time. Consequently, they may produce less accurate forecasts at lower levels of the hierarchy compared to the bottom-up method. To address this limitation, Athanasopoulos et al. [81] introduced an enhanced top-down approach that employs forecast-based proportions:

$$p_i = \prod_{l=0}^{k-1} \frac{\hat{y}_{i,t+h|t}^{(l)}}{\hat{S}_{i,t+h|t}^{(l+1)}}, \quad i = 1, \ldots, m, \tag{9}$$

where k represents the number of levels in the hierarchy. In this formulation, $\hat{y}_{i,t+h|t}^{(l)}$ denotes the h-step-ahead base forecast for the series corresponding to the node l levels above node i and $\hat{S}_{i,t+h|t}^{(l+1)}$ is the sum of the h-step-ahead base forecasts for the series associated with the nodes $l+1$ levels above node i.

Hyndman et al. [83] proposed an optimal reconciliation method based on a regression model:

$$\hat{\boldsymbol{y}}_{t+h|t} = \boldsymbol{S}\boldsymbol{\beta}_{t+h|t} + \boldsymbol{\varepsilon}_h, \tag{10}$$

where $\boldsymbol{\beta}_{t+h|t}$ represents the vector of unknown means for the most disaggregated series and $\boldsymbol{\varepsilon}_h$ is the reconciliation error with a mean of zero and a covariance matrix (Σ_h). When Σ_h is known, the Generalized Least Squares (GLS) estimator of $\boldsymbol{\beta}_{t+h|t}$ can be used to generate the following reconciled forecasts:

$$\tilde{\boldsymbol{y}}_{t+h|t} = \boldsymbol{S}\hat{\boldsymbol{\beta}}_{t+h|t} = \boldsymbol{SP}\hat{\boldsymbol{y}}_{t+h|t} = \boldsymbol{S}(\boldsymbol{S}'\Sigma_h^{-1}\boldsymbol{S})^{-1}\boldsymbol{S}'\Sigma_h^{-1}\hat{\boldsymbol{y}}_{t+h|t}. \tag{11}$$

Hyndman et al. [83] demonstrated that if the base forecasts ($\hat{\boldsymbol{y}}_{t+h|t}$) are unbiased, the reconciled forecasts $\tilde{\boldsymbol{y}}_{t+h|t}$ will also be unbiased, provided the condition ($\boldsymbol{SPS} = \boldsymbol{S}$) holds. This condition is met by both the optimal reconciliation approach and the bottom-up method. However, no top-down method satisfies this condition, which implies that top-down methods inherently produce biased reconciled forecasts. Wickramasuriya et al. [85] further showed that the reconciliation approach proposed by Hyndman et al. [83] is generally impractical because Σ_h is typically unknown and cannot be accurately determined. According to these authors, the covariance matrix of the h-step-ahead reconciled forecast errors is given by

$$\mathrm{Var}[\boldsymbol{y}_{t+h} - \tilde{\boldsymbol{y}}_{t+h|t}] = \boldsymbol{SPW}_h\boldsymbol{P}'\boldsymbol{S}', \tag{12}$$

where $\boldsymbol{W}_h = \mathrm{Var}[\boldsymbol{y}_{t+h} - \hat{\boldsymbol{y}}_{t+h|t}]$ is the variance–covariance matrix of the h-step-ahead base forecast errors. We seek to identify the $\boldsymbol{P}$ matrix that minimizes the error variances of the reconciled forecasts, which are represented by the diagonal elements of $\mathrm{Var}[\boldsymbol{y}_{t+h} - \hat{\boldsymbol{y}}_{t+h|t}]$. The optimal reconciliation approach, known as the MinT (minimum trace) method, was proposed by Wickramasuriya et al. [85]. This approach determines that matrix $\boldsymbol{P}$, which minimizes the trace of $\mathrm{Var}[\boldsymbol{y}_{t+h} - \hat{\boldsymbol{y}}_{t+h|t}]$ while satisfying the condition of $\boldsymbol{SPS} = \boldsymbol{S}$, is expressed as follows:

$$\boldsymbol{P} = (\boldsymbol{S}'\boldsymbol{W}_h^{-1}\boldsymbol{S})^{-1}\boldsymbol{S}'\boldsymbol{W}_h^{-1}. \tag{13}$$

Therefore, the reconciled forecasts generated by the MinT approach are expressed as follows:

$$\tilde{\boldsymbol{y}}_{t+h|t} = \boldsymbol{S}(\boldsymbol{S}'\boldsymbol{W}_h^{-1}\boldsymbol{S})^{-1}\boldsymbol{S}'\boldsymbol{W}_h^{-1}\hat{\boldsymbol{y}}_{t+h|t}. \tag{14}$$

This optimal reconciliation approach still requires the estimation of $\boldsymbol{W}_h$. Wickramasuriya et al. [85] proposed several alternatives for this purpose:

1. $\boldsymbol{W}_h = k_h\boldsymbol{I}_n$, $\forall h$, where $k_h > 0$. In this case, the estimator for $\boldsymbol{\beta}_{t+h|t}$ corresponds to the OLS estimator. Although this is the simplest estimation method, matrix $\boldsymbol{P}$ does not rely on the data, meaning it does not account for differences in scale between hierarchical levels or the relationships among the series. This specification is referred to as OLS.
2. $\boldsymbol{W}_h = k_h\mathrm{diag}(\widehat{\boldsymbol{W}}_1)$, $\forall h$, where $k_h > 0$ and $\widehat{\boldsymbol{W}}_1 = \frac{1}{T}\Sigma_{t=1}^{T}\boldsymbol{e}_t\boldsymbol{e}_t'$ is the sample covariance estimator of the one-step-ahead base forecast errors. This approach scales the base forecasts using the variance of the residuals ($\boldsymbol{e}_t$). This MinT estimator is referred to as WLS (var).
3. $\boldsymbol{W}_h = k_h\boldsymbol{\Lambda}$, $\forall h$, where $k_h > 0$, $\boldsymbol{\Lambda} = \mathrm{diag}(\boldsymbol{S1})$, with $\mathbf{1}$ being a unit vector of dimension m. This specification assumes that the variance of the base forecast errors at the bottom level is k_h and that these errors are uncorrelated across different nodes. The estimator relies solely on the aggregation constraints of the hierarchy rather than on the data, making it particularly useful when residuals are not available. This method is known as structural scaling and is denoted as WLS (struct).
4. $\boldsymbol{W}_h = k_h\widehat{\boldsymbol{W}}_1$, $\forall h$, where $k_h > 0$ represents the sample covariance estimator for $h = 1$. This estimator is straightforward to compute, but it may be unsuitable when the number of bottom-level series (m) exceeds the number of time periods (T). This specification is referred to as MinT (sample).
5. $\boldsymbol{W}_h = k_h\widehat{\boldsymbol{W}}_{1,D}^{*}$, $\forall h$, where $k_h > 0$ represents a shrinkage estimator. Here, $\widehat{\boldsymbol{W}}_{1,D}^{*} = \lambda_D\widehat{\boldsymbol{W}}_{1,D} + (1-\lambda_D)\widehat{\boldsymbol{W}}_1$ is designed to shrink the off-diagonal elements of $\widehat{\boldsymbol{W}}_1$ toward zero while leaving the diagonal entries unchanged. In this formulation, $\widehat{\boldsymbol{W}}_{1,D}$

is a diagonal matrix containing the diagonal elements of $\widehat{\boldsymbol{W}}_1$ and λ_D is the shrinkage intensity parameter. Assuming constant variances, Schäfer and Strimmer [106] proposed the following formula for the shrinkage intensity parameter:

$$\hat{\lambda}_D = \frac{\Sigma_{i \neq j} \widehat{\mathrm{Var}(\hat{r}_{ij})}}{\Sigma_{i \neq j} \hat{r}_{ij}^2}, \tag{15}$$

where $\hat{r}_{ij}$ represents the ijth element of $\widehat{\boldsymbol{R}}_1$, the sample correlation matrix of the one-step-ahead base forecast errors. This approach is referred to as MinT (shrink).

3.2. ARIMA Models

The seasonal ARIMA model, denoted as ARIMA$(p,d,q) \times (P,D,Q)_m$, is expressed as follows [107]:

$$\phi_p(B)\Phi_P(B^m)(1-B)^d(1-B^m)^D\eta_t = c + \theta_q(B)\Theta_Q(B^m)\varepsilon_t, \tag{16}$$

$$\begin{aligned} \phi_p(B) &= 1 - \phi_1 B - \cdots - \phi_p B^p, & \Phi_P(B^m) &= 1 - \Phi_1 B^m - \cdots - \Phi_P B^{Pm}, \\ \theta_q(B) &= 1 + \theta_1 B + \cdots + \theta_q B^q, & \Theta_Q(B^m) &= 1 + \Theta_1 B^m + \cdots + \Theta_Q B^{Qm}. \end{aligned}$$

In this equation, η_t represents the time series being modeled, while m denotes the seasonal period. The terms D and d correspond to the degrees of seasonal and ordinary differencing, respectively. B is the backward shift operator, and $\phi_p(B)$ and $\theta_q(B)$ are the regular autoregressive and moving average polynomials of orders p and q, respectively. Similarly, $\Phi_P(B^m)$ and $\Theta_Q(B^m)$ represent the seasonal autoregressive and moving-average polynomials of orders P and Q, respectively. The constant (c) is defined as $c = \mu(1 - \phi_1 - \cdots - \phi_p)(1 - \Phi_1 - \cdots - \Phi_P)$, where μ is the mean of $(1-B)^d(1-B^m)^D\eta_t$ and ε_t denotes a white-noise series that is uncorrelated over time, with zero mean and constant variance. To ensure the stationarity and invertibility of the model, the roots of polynomials $\phi_p(B)$, $\Phi_P(B^m)$, $\theta_q(B)$, and $\Theta_Q(B^m)$ must all lie outside of the unit circle. Non-stationary time series can be made stationary by applying transformations such as logarithms to stabilize the variance and by using proper degrees of differencing to stabilize the mean. Once the values of p, q, P, and Q are specified, the model parameters—$c, \phi_1 \ldots, \phi_p, \theta_1, \ldots, \theta_q, \Phi_1 \ldots, \Phi_P, \Theta_1, \ldots, \Theta_Q$—can be estimated by maximizing the log-likelihood function. To select the optimal values for p, q, P, and Q, the Akaike Information Criterion (AIC) is typically used. The AIC balances model fit with complexity by penalizing the number of parameters, helping to prevent overfitting.

While pure forecasting models such as ARIMA leverage historical values of the time series to generate forecasts, they do not account for external factors that might influence the dependent variable. In contexts like retail sales forecasting, where promotional activities, marketing campaigns, calendar events, and school holidays can significantly impact demand, it is crucial to consider these external effects. Conversely, regression models can integrate exogenous variables but are not designed to capture the time-series dynamics on their own. To address this limitation, ARIMA models can be extended to include explanatory variables by adopting a regression framework with ARIMA errors. This approach is expressed as follows [108]:

$$\begin{aligned} y_t &= \delta_0 + \sum_{k=1}^{K} \delta_k x_{k,t} + \eta_t, \\ \phi_p(B)\Phi_P(B^m)(1-B)^d(1-B^m)^D\eta_t &= \theta_q(B)\Theta_Q(B^m)\varepsilon_t, \end{aligned} \tag{17}$$

where y_t represents the target time series; $x_{1,t}, \ldots, x_{K,t}$ are the explanatory variables; and $\delta_0, \delta_1, \ldots$, and δ_K denote the coefficients of the regression model. The error term (η_t) is modeled using an ARIMA process.

To estimate all parameters in this framework, including those of both the regression and ARIMA components, one can maximize the log-likelihood function, as with a standard ARIMA model. However, it is crucial to ensure that both the dependent variable (y_t) and the explanatory variables ($x_{1,t}, \ldots, x_{K,t}$) are stationary to obtain consistent estimates. Consistency requires that the same differencing procedure be applied uniformly to all series involved to preserve the relationship between the dependent and independent variables.

ARIMA models are particularly effective at capturing linear relationships in time-series data and are well-suited for scenarios where trends and seasonality need to be explicitly modeled. Their ability to incorporate external factors, such as promotional activities, makes them valuable for adjusting forecasts based on variables that influence sales. This is especially important in retail, where accurate forecasting must account for fluctuations driven by promotions, pricing, and other market dynamics. By leveraging ARIMA, we mitigate the risk of under- or over-estimating sales in response to these factors, thereby improving the precision of our base forecasts.

3.3. Multi-Layer Perceptrons for Time Series Forecasting

Deep learning algorithms are a subset of machine learning and mimic the structure of the human brain. They employ multi-layer neural networks to process data through successive transformations, ultimately achieving optimal representations [109]. Learning occurs through the adjustment of these neural networks. The most fundamental deep neural networks are known as multi-layer perceptrons (MLPs). These networks consist of several layers of neurons (or nodes), where each neuron in one layer is fully connected to all neurons in the layer below and those in the layer above. At its core, a multi-layer perceptron includes three layers: an input layer that handles the raw data, a hidden layer where the learning primarily occurs, and an output layer that produces the predictions. Designing an MLP involves determining the number of layers and nodes, which is often more art than science. Typically, the number of nodes in each hidden layer is related to the number of input variables. The complexity of an MLP—reflected by the number of hidden layers and nodes—affects its ability to learn complex features from the data. The configuration of the output layer depends on the nature of the task; for forecasting tasks that typically involve predicting continuous values, the output layer consists of one or more nodes that provide the final prediction.

Consider a matrix ($\boldsymbol{X} \in \mathbb{R}^{n \times d}$) representing a minibatch of n time series, each with d features. For a one-hidden-layer MLP with h hidden units, let $\boldsymbol{H} \in \mathbb{R}^{n \times h}$ denote the outputs of the hidden layer, which are the hidden representations. The weights for the hidden layer are $\boldsymbol{W}^{(1)} \in \mathbb{R}^{d \times h}$, and the biases are $\boldsymbol{b}^{(1)} \in \mathbb{R}^{1 \times h}$, while the weights of the output layer are $\boldsymbol{W}^{(2)} \in \mathbb{R}^{h \times q}$ and biases are $\boldsymbol{b}^{(2)} \in \mathbb{R}^{1 \times q}$. The outputs of the MLP ($\boldsymbol{O} \in \mathbb{R}^{n \times q}$) can be calculated as follows [110]:

$$\boldsymbol{H} = \boldsymbol{X}\boldsymbol{W}^{(1)} + \boldsymbol{b}^{(1)}, \tag{18}$$

$$\boldsymbol{O} = \boldsymbol{H}\boldsymbol{W}^{(2)} + \boldsymbol{b}^{(2)}. \tag{19}$$

To fully leverage multi-layer architectures, a nonlinear activation function (σ) must be applied to each hidden unit following the affine transformation [111]. These differentiable functions introduce nonlinearity to the network, enabling it to model more complex relationships. Among the most common activation functions is ReLU (Rectified Linear Unit):

$$\mathrm{ReLU}(x) = \max(x, 0), \tag{20}$$

which retains only positive values and sets negative values to zero. ReLU is favored for its well-behaved derivatives, which either vanish or pass through the argument, improving optimization and mitigating the vanishing gradient problem. Another activation function is the sigmoid function:

$$\text{sigmoid}(x) = \frac{1}{1 + e^{-x}}. \tag{21}$$

The sigmoid function maps input values to the interval of (0, 1), making it useful for tasks involving probabilities. However, it has largely been replaced by ReLU in many applications due to vanishing gradients for extreme values. The tanh (hyperbolic tangent) activation squashes input values to lie between -1 and 1:

$$\tanh(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}}, \tag{22}$$

The output of the activation function ($\sigma(\cdot)$) is referred to as the activation. By introducing activation functions, an MLP cannot be reduced to a simple linear model:

$$\boldsymbol{H} = \sigma(\boldsymbol{X}\boldsymbol{W}^{(1)} + \boldsymbol{b}^{(1)}), \tag{23}$$

$$\boldsymbol{O} = \boldsymbol{H}\boldsymbol{W}^{(2)} + \boldsymbol{b}^{(2)}. \tag{24}$$

Each row in $\boldsymbol{X}$ represents a single time series from the minibatch and the nonlinearity (σ) applied to each row individually. To create more complex MLPs, additional hidden layers can be stacked as follows:

$$\boldsymbol{H}^{(1)} = \sigma_1(\boldsymbol{X}\boldsymbol{W}^{(1)} + \boldsymbol{b}^{(1)}), \tag{25}$$

$$\boldsymbol{H}^{(2)} = \sigma_2(\boldsymbol{H}^{(1)}\boldsymbol{W}^{(2)} + \boldsymbol{b}^{(2)}), \tag{26}$$

which leads to increasingly expressive models. Given a dataset, the goal is to determine the weights ($\boldsymbol{W}$) and bias ($\boldsymbol{b}$) that minimize the prediction error. For a given set of features ($\boldsymbol{X}$) and known labels ($\boldsymbol{y}$), the objective is to find parameters ($\boldsymbol{W}$ and $\boldsymbol{b}$) that produce the most accurate predictions for new time series. To optimize the model, we need (1) a loss function to measure model performance and (2) an update procedure to improve the model. Loss functions quantify the difference between actual and predicted values. For forecasting tasks, the Mean Squared Error (MSE) is a commonly used loss function:

$$L = \frac{1}{q}\sum_{i=1}^{q}\left(y^{(i)} - \hat{y}^{(i)}\right)^2, \tag{27}$$

where $y^{(i)}$ denotes the true label and $\hat{y}^{(i)}$ represents the predicted value. During training, the objective is to minimize the loss function by adjusting the model parameters. Stochastic Gradient Descent (SGD) is the primary technique for optimization, involving iterative adjustments of model parameters to reduce the loss function. Instead of processing the entire dataset at once, SGD uses minibatches of data. The size of each minibatch is determined by factors such as memory capacity, computational resources, and dataset size. In each iteration, a minibatch is randomly sampled, and the gradient of the loss with respect to model parameters is computed. This gradient is scaled by a learning rate and subtracted from the current parameter values. After completing a specified number of iterations or meeting a stopping criterion, the estimated parameters ($(\hat{\boldsymbol{W}}, \hat{\boldsymbol{b}})$) are obtained. Due to the stochastic nature of minibatch sampling, these parameters might not be exact minimizers. Batch normalization helps address internal covariate shift by normalizing layer inputs to maintain consistent statistics (mean and variance), thereby enhancing training efficiency. Ensuring that MLPs generalize well to new, unseen data is crucial. Underfitting, where

the model fails to capture the underlying patterns in the data, can often be mitigated by increasing model complexity. Overfitting, where the model performs well on training data but poorly on new data, typically requires regularization techniques such as L1 or L2 regularization, early stopping, or dropout to manage high variance and improve generalization. In the MLP, external information at different times is treated as input features alongside the target variable (sales). The MLP architecture can handle multiple input variables simultaneously. Each input, whether it is the target variable (historical sales) or an external variable (e.g., promotional status, price, calendar events), is fed into the input layer of the neural network. The network processes these data through the hidden layers using weights and activation functions, allowing it to learn complex nonlinear relationships between sales and the external factors.

MLP neural networks offer a powerful tool for modeling complex, nonlinear relationships in the data. Retail environments often involve intricate interactions between factors such as consumer behavior, promotional campaigns, and seasonal effects. MLPs are particularly adept at uncovering these hidden patterns and relationships, which are difficult to capture using traditional statistical methods like ARIMA. This makes MLPs highly valuable in addressing the problem of forecasting under uncertainty, particularly when multiple variables interact in unpredictable ways. By employing both ARIMA and MLP models, this study directly tackles the challenges of sales forecasting through a combination of linear time-series analysis and nonlinear pattern recognition. This dual approach ensures that the forecasting problem is addressed from multiple angles, improving the overall accuracy and robustness of the forecasts across different hierarchical levels.

4. Empirical Study

In this empirical study, we aim to evaluate the performance of ARIMA and MLP models in the context of hierarchical sales forecasting for the retail sector. The main goal is to investigate how the integration of promotional activities into these models can enhance forecast accuracy across multiple hierarchical levels (SKU, region, and store). The study focuses on the ability of both traditional and deep learning methods to handle the complexities of retail sales data, especially in the presence of promotions, seasonal effects, and other external factors. By comparing ARIMA and MLP models and applying reconciliation strategies such as bottom-up, top-down, and optimal reconciliation, we seek to determine the most effective approach for ensuring coherent forecasts across different levels of aggregation. This research is motivated by the need for accurate and consistent forecasts at multiple organizational levels (e.g., SKU, store, and distribution center), which are crucial for optimizing supply chain management and decision making. Hierarchical forecasting addresses this need by capturing relationships between aggregated and disaggregated data, ensuring that forecasts are aligned across levels. Our experimental setup uses real-world data from a Portuguese retailer, Jerónimo Martins, and evaluates the models' ability to forecast sales while incorporating key external factors such as prices, promotions, and calendar events.

4.1. Case Study Data

Jerónimo Martins Group is a Portugal-based international company with extensive retail experience. Primarily focused on food distribution, the Group dominates Portugal's supermarket segment through its Pingo Doce chain. This case study aims to enhance sales forecast accuracy in the retail sector by integrating promotional activity into hierarchical forecasting methods. To this end, Jerónimo Martins provided daily sales and product price data from 10 Pingo Doce stores, spanning from 3 January 2012 to 27 April 2015 (1211 days). We analyzed 38 SKUs across five product categories: non-specialized perishables, grocery, beverages, specialized perishables, and detergents/cleaning products. These SKUs exhibited diverse sales patterns and price sensitivities, providing a robust dataset for evaluating forecasting methods. A three-level hierarchical structure was created for each SKU (see Figure 2). The top level (level 0) represents total sales, divided into sales for regions A and B at the second level (level 1). The bottom level (level 2) details sales for the 10 stores,

with 2 in region A and 8 in region B. The number of levels was determined based on the operational structure of the retailer. We used three levels—total, regional, and store levels—because these levels align with the retailer's organizational hierarchy. At the top level, total sales represent the overall demand. The regional level captures the sales at the data warehouse level, which supports logistics and distribution decisions. Finally, the store level provides granular insights into local demand, which are critical for inventory management and promotions at individual locations. This three-level structure reflects the natural flow of products and decision making within the retail organization, making it an appropriate choice for our hierarchical forecasting model. Each hierarchical structure contains 13 time series. With 38 SKUs and a data period of 1211 days, this study analyzed 494 series, comprising a total of 598,234 observations. Table 1 outlines the series distribution across hierarchical levels.

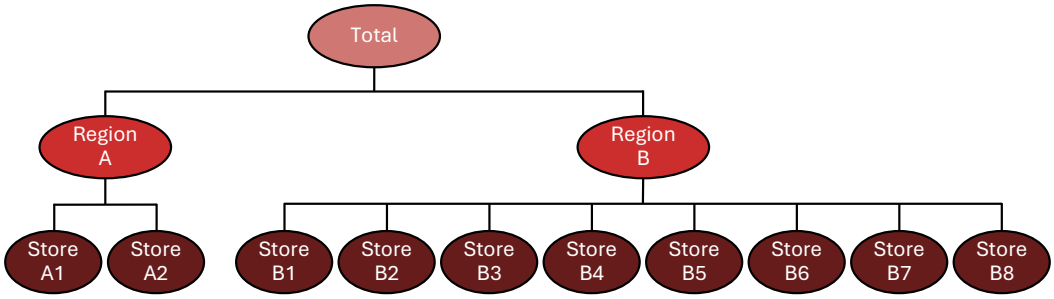

Figure 2. Hierarchical structure of Pingo Doce sales data by SKU, illustrating three levels: total, regional (A, B), and store-level sales.

Table 1. Distribution of time series across hierarchical levels.

Hierarchical Level	Number of Series per SKU	Total Number of Series
Level 0	1	38
Level 1	2	76
Level 2	10	380
Total	13	494

Figure 3 illustrates data aggregation by presenting the daily sales series of a representative SKU across the three hierarchical levels from 2012 to 2015. The top level displays total sales, the middle level shows sales by region, and the bottom level provides sales by store. The sales data exhibit a strong seasonal pattern at all levels, with peaks occurring at certain times of the year due to factors such as holidays or seasonal product demand. Specifically, the sharp peaks around December are attributed to Christmas, when consumer demand typically surges. Additionally, the retailer implemented a 50% discount promotion on 1 May (Labor Day), leading to a noticeable spike in sales across all stores during that period. A hierarchical structure is clearly depicted, with the top level showing the overall trend, the middle level showing regional differences, and the bottom level providing detailed store-level information. Sales data at the store level (Level 2) exhibit the highest variability, indicating significant differences in performance among individual stores. As we move up the hierarchy, the data become smoother, reflecting aggregated sales. In Figure 3, each line at Level 1 represents the sales performance of a specific region, showcasing the differences in regional demand patterns. These regional-level sales trends highlight distinct fluctuations, with certain regions displaying stronger peaks during holiday seasons or promotional periods. At Level 2, each line corresponds to the sales of an individual store within the respective region. The variability in sales performance is most pronounced at this level, indicating significant differences between stores in terms of customer traffic, local demand, and store-specific factors. This detailed granularity at Level 2 helps to illustrate how localized dynamics contribute to overall regional and national sales trends.

Figure 3. Hierarchical daily sales data for a representative SKU from 2012 to 2015, illustrating total, regional, and store-level sales with seasonal patterns. Different colors represent distinct aggregation levels and the individual time series within each level.

4.2. Experimental Setup

Base forecasts for all time series within the hierarchical structures were generated using both ARIMA models and multi-layer perceptrons (MLPs). ARIMA models were estimated and forecasted using the *ARIMA()* and *forecast()* functions from the *fable* and *fabletools* R packages [112]. MLPs were built using Python's Keras and TensorFlow libraries.

The multi-layer perceptron (MLP) architecture consisted of an input layer, multiple hidden layers, and an output layer. Models were trained for 100 epochs using a 28-day sliding window. The rectified linear unit (ReLU) activation function was employed in the hidden layers, while the Huber loss function was utilized to balance sensitivity to outliers and robustness to noise. Hyperparameter tuning, including the number of hidden layers, number of nodes per layer, batch size, and learning rate, was conducted through grid search. The optimal configuration was determined to be a single hidden layer with 80 nodes, a batch size of 2, and a stochastic gradient descent (SGD) optimizer with a learning rate of 0.001. To mitigate overfitting, early stopping and L2 regularization were implemented. Early stopping monitored the validation loss and halted training when no further improvement was observed. L2 regularization added a penalty term to the loss function, discouraging excessively large weights. A regularization parameter of 0.001 was found to be effective in balancing model complexity and generalization.

Initially, only historical sales data were used as input for forecasting. To enhance the model's robustness and interpretability, a set of nine additional regressors was carefully selected and incorporated into both ARIMA and MLP models. These regressors include the SKU price; six binary variables representing the days of the week; and two binary variables indicating major holidays, namely Christmas and Easter. The inclusion of price serves as a proxy for promotions, as price reductions typically signal promotional activity, offering insight into how discounts affect sales patterns. The weekday variables capture weekly seasonality, helping the model account for fluctuations in demand across different days.

The holiday indicators are critical for modeling sales spikes during periods of heightened consumer activity.

To ensure forecast coherence within the hierarchical structure, base forecasts were reconciled using both bottom-up and top-down methods [81]. These methods rely on proportions calculated from forecasts. Additionally, optimal reconciliation strategies employing ordinary least squares (OLS) and weighted least squares (WLS) estimators were applied. The *htsrec()* and *tdrec()* functions from the R package *FoReco* were used to implement these reconciliation techniques.

To evaluate the performance of a forecasting model, it is common practice to split the dataset into two parts: a training set and a testing set. The training set is used to estimate the model's parameters, while the testing set is used to assess the accuracy of the forecasts generated by the model. In this study, the ARIMA models were estimated based on 1085 days of data from 3 January 2012 to 22 December 2014. The subsequent period, from 23 December 2014 to 27 April 2015 (126 days), served as the testing set. Conversely, the MLPs utilized only the initial 847 days for training, with the following 238 days designated as a validation set to optimize model parameters. The testing set for the MLPs remained identical to that of the ARIMA models.

Forecasts were generated using a rolling-window approach, as depicted in Figure 4. The training set is indicated by dark-red bars, and the testing set by light-red bars. The training set was iteratively expanded by one day at each step, successively shifting the forecast origin. This process yielded seven-day-ahead forecasts.

Figure 4. Rolling-window approach for forecasting with 120 rolling steps. The training set is represented by dark-red bars and the testing set by light-red bars. At each step, the training set is incrementally extended by one day, producing forecasts for a 7-day horizon.

Given a training set $(y_1, y_2, \ldots, y_T)$ and a testing set $(y_{T+1}, y_{T+2}, \ldots)$, for a given time period ($T + h$, where h denotes the forecast horizon), the forecast error e_{T+h} is the difference between the observed value (y_{T+h}) and the predicted value ($\hat{y}_{T+h|T}$):

$$e_{T+h} = y_{T+h} - \hat{y}_{T+h|T}. \tag{28}$$

To compare forecast errors across different data scales at various hierarchical levels, model performance was evaluated using scaled error metrics: the Mean Absolute Scaled Error (MASE) proposed by Hyndman and Koehler [113] and the Root Mean Squared Scaled Error (RMSSE). In the MASE, errors are scaled by the Mean Absolute Error (MAE) of the seasonal naïve method computed on the training set:

$$\text{SAE}_j = \frac{|e_j|}{\frac{1}{T-m}\sum_{t=m+1}^{T}|y_t - y_{t-m}|}, \tag{29}$$

where m denotes the seasonal period, set to $m = 7$ to capture the inherent weekly seasonality of the data. Similarly, in the RMSSE, errors are scaled by the Mean Squared Error (MSE) of the seasonal naïve method computed on the training set:

$$\text{SSE}_j = \frac{e_j^2}{\frac{1}{T-m}\sum_{t=m+1}^{T}(y_t - y_{t-m})^2}. \tag{30}$$

The MASE and RMSSE forecast accuracy metrics were calculated as the mean and square root of the mean of all SAE_j and SSE_j values, respectively:

$$\mathrm{MASE} = \mathrm{mean}\left(\mathrm{SAE}_j\right), \tag{31}$$

$$\mathrm{RMSSE} = \sqrt{\mathrm{mean}\left(\mathrm{SSE}_j\right)}. \tag{32}$$

4.3. Results

The evaluation of the forecasting models is presented progressively, focusing first on the most significant outcomes, then expanding to more detailed comparisons across models, forecasting horizons, and hierarchical levels. Figure 5 demonstrates the training progress for MLP models with and without the inclusion of regressors. Both Huber loss and mean absolute error (MAE) progressively decline over the course of training in both the training and validation datasets, with a more pronounced reduction during the initial stages. This trend is characteristic of neural networks, where early learning typically captures significant patterns before fine tuning later in the training process. Notably, the MLP model with regressors converged more rapidly, triggering early stopping after 29 epochs, whereas the standard MLP persisted to the maximum number of epochs. This behavior aligns with expectations, as the early-stopping criterion prevents overfitting, especially in the presence of added regressors, which can accelerate convergence. As is typical, the training set achieved lower loss and error metrics compared to the validation set, given that the model was directly optimized using the training data while remaining blind to the validation set.

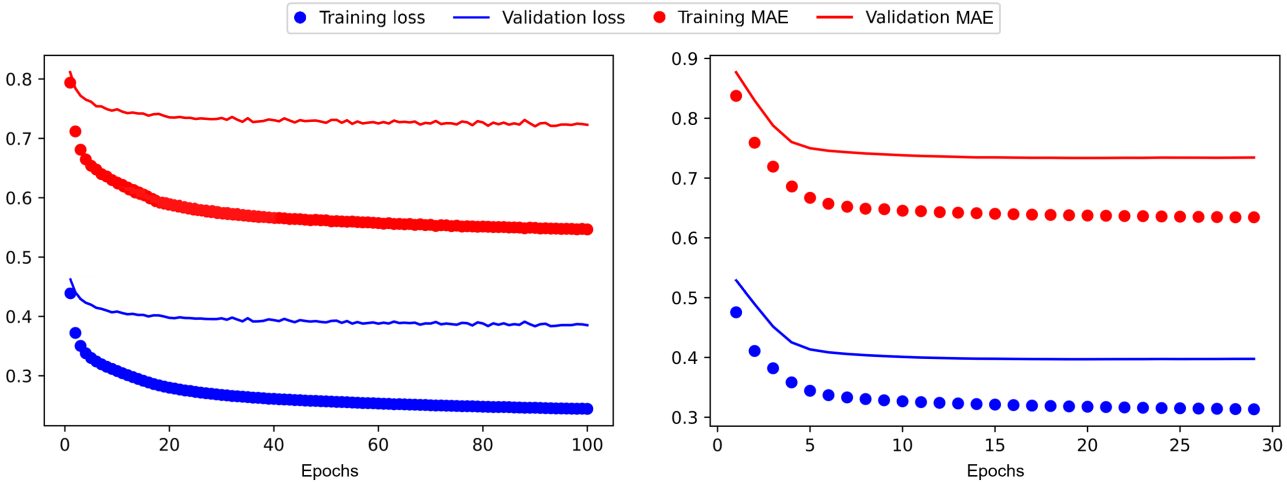

Figure 5. Huber loss (blue) and mean absolute error (MAE, red) during MLP training for a typical SKU, comparing the standard MLP (**left**) and an MLP with regressors (**right**).

Figure 6 offers a comparative visual analysis of sales forecasts for a representative SKU over the test period from 23 December 2014 to 27 April 2015. This figure juxtaposes actual sales data with forecasted values produced by ARIMA, ARIMAX, MLP, and MLP models incorporating regressors. Despite the inherent difficulty of forecasting highly volatile sales patterns, models augmented with regressors exhibit a closer approximation to the true data compared to their non-regressor counterparts. This suggests that external factors captured by regressors contribute meaningfully to the model's ability to mitigate error in the face of variability, although no model perfectly captures the extremes of sales fluctuations.

The comparative forecast accuracy of the four models—ARIMA, ARIMAX, MLP, and MLP with regressors—is presented in Tables 2–5. These tables evaluate performance across various hierarchical levels and forecast horizons, spanning one to seven days ($h = 1$ to $h = 7$), as well as the overall average. For each forecast horizon and hierarchical level, the most accurate model is highlighted in bold. Clear trends emerge: ARIMAX consistently outperforms ARIMA across all horizons and hierarchical levels, highlighting the beneficial impact of incorporating regressors into model performance.

Notably, while ARIMA-based models typically struggle with longer-term forecasts, MLP models demonstrate superior performance beyond the one-day forecast horizon. This finding is particularly significant for retail applications, where medium-term forecasts are critical for effective inventory management. The advantage of MLPs is further amplified when base forecasts are reconciled across hierarchical levels, with the reconciliation process effectively enhancing forecast accuracy for longer-term horizons.

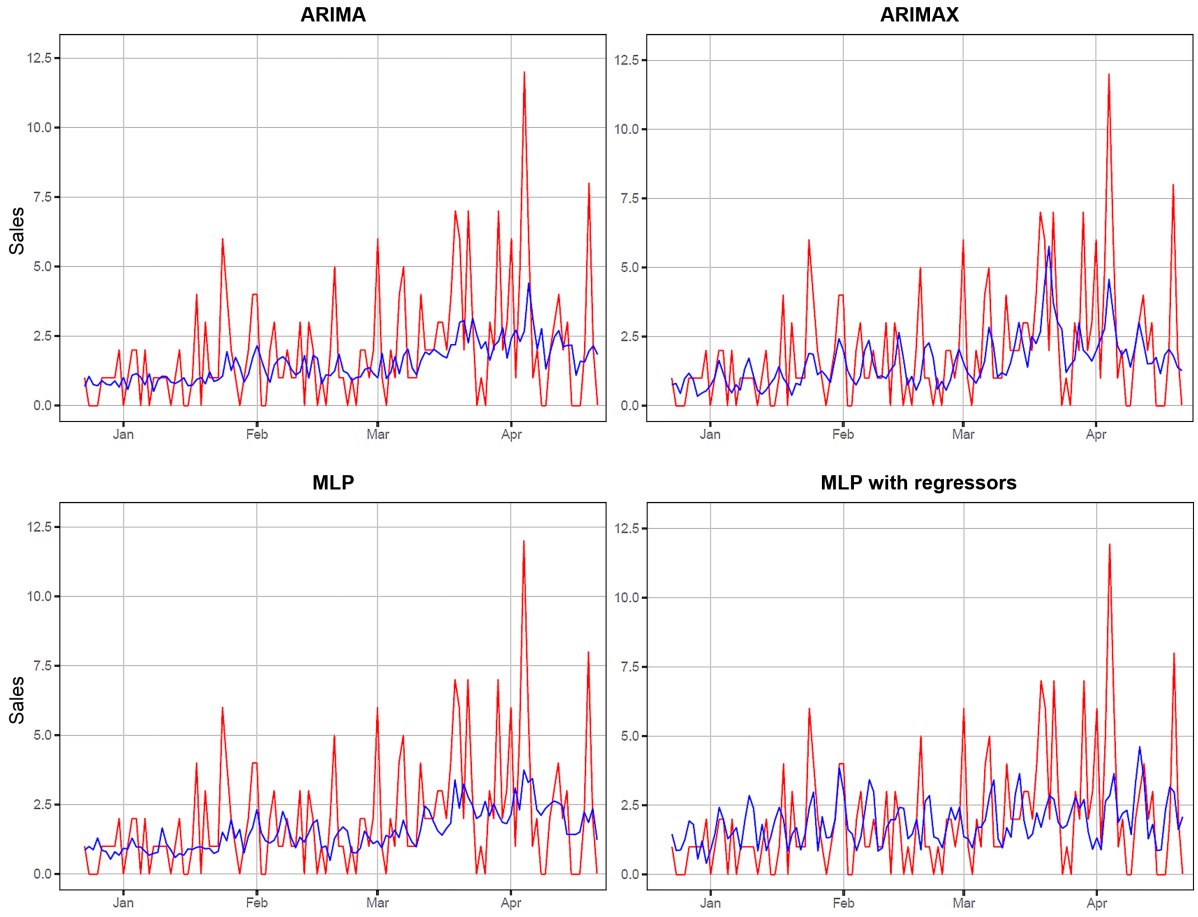

Figure 6. Actual and forecasted sales for a representative SKU, comparing ARIMA, ARIMAX, MLP, and MLP with regressors models (red: actual; blue: forecast).

Interestingly, the inclusion of regressors does not universally improve MLP performance, in stark contrast to their impact on ARIMA models. The ARIMAX model's advantage likely stems from its ability to incorporate real-time regressor data during forecast generation, whereas MLP models, constrained by their architecture, utilize only historical regressor data. This limitation may account for the diminished effectiveness of regressors in the MLP framework. Exploring more sophisticated neural architectures, which allow for dynamic incorporation of external data, could potentially bridge this gap, although at the cost of increased computational demand—a tradeoff less desirable in retail environments where speed and efficiency are paramount, given the vast number of SKUs to forecast.

In terms of reconciliation strategies, the WLS (struct) estimator consistently provides superior results, although it is occasionally outperformed by the bottom-up method, particularly at lower levels of the forecasting hierarchy. This result suggests that while sophisticated reconciliation methods offer benefits, simpler approaches may still yield competitive accuracy in specific contexts, especially when more granular forecasting is required.

Table 2. MASE and RMSSE of forecasts generated by ARIMA models.

ARIMA								
MASE	$h = 1$	$h = 2$	$h = 3$	$h = 4$	$h = 5$	$h = 6$	$h = 7$	$h = 1$–7
Level 0: Total								
Base	0.9811	1.0442	1.0650	1.0773	1.0950	1.1047	1.1041	1.0673
Bottom-up	1.0090	1.0438	1.0398	1.0327	**1.0336**	**1.0292**	**1.0228**	1.0301
Top-down	0.9811	1.0442	1.0650	1.0773	1.0950	1.1047	1.1041	1.0673
OLS	0.9671	1.0260	1.0463	1.0531	1.0670	1.0743	1.0735	1.0439
WLS (struct)	**0.9609**	**1.0131**	**1.0256**	**1.0288**	1.0376	1.0407	1.0382	**1.0207**
Level 1: Region								
Base	0.8975	0.9381	0.9496	0.9502	0.9570	0.9592	0.9590	0.9444
Bottom-up	0.9157	0.9439	0.9427	**0.9388**	**0.9409**	**0.9377**	**0.9337**	0.9362
Top-down	0.9118	0.9595	0.9733	0.9814	0.9921	0.9987	0.9979	0.9735
OLS	0.8976	0.9455	0.9605	0.9660	0.9761	0.9811	0.9812	0.9583
WLS (struct)	**0.8875**	**0.9274**	**0.9367**	0.9399	0.9471	0.9490	0.9479	**0.9336**
Level 2: Store								
Base	**0.7974**	**0.8113**	**0.8106**	**0.8109**	**0.8120**	**0.8107**	**0.8083**	**0.8088**
Bottom-up	**0.7974**	**0.8113**	**0.8106**	**0.8109**	**0.8120**	**0.8107**	**0.8083**	**0.8088**
Top-down	0.8136	0.8412	0.8523	0.8627	0.8724	0.8796	0.8810	0.8575
OLS	0.8115	0.8402	0.8523	0.8611	0.8697	0.8762	0.8775	0.8555
WLS (struct)	0.7981	0.8221	0.8307	0.8369	0.8433	0.8470	0.8473	0.8322

RMSSE	$h = 1$	$h = 2$	$h = 3$	$h = 4$	$h = 5$	$h = 6$	$h = 7$	$h = 1$–7
Level 0: Total								
Base	1.6757	1.7551	1.7809	1.7963	1.8277	1.8445	1.8332	1.7876
Bottom-up	1.7602	1.7792	1.7691	1.7734	1.7792	1.7709	1.7613	1.7705
Top-down	1.6757	1.7551	1.7809	1.7963	1.8277	1.8445	1.8332	1.7876
OLS	**1.6607**	1.7311	1.7557	1.7629	1.7869	1.8006	1.7921	1.7557
WLS (struct)	1.6637	**1.7140**	**1.7264**	**1.7329**	**1.7505**	**1.7567**	**1.7481**	**1.7275**
Level 1: Region								
Base	1.4903	1.5373	1.5538	1.5519	1.5621	1.5670	1.5639	1.5466
Bottom-up	1.5476	1.5609	1.5555	1.5617	1.5673	1.5619	1.5547	1.5585
Top-down	1.5004	1.5575	1.5737	1.5865	1.6105	1.6230	1.6152	1.5810
OLS	1.4836	1.5377	1.5567	1.5645	1.5854	1.5944	1.5885	1.5587
WLS (struct)	**1.4825**	**1.5185**	**1.5273**	**1.5351**	**1.5496**	**1.5538**	**1.5480**	**1.5307**
Level 2: Store								
Base	1.2645	1.2757	**1.2723**	**1.2764**	**1.2796**	**1.2785**	**1.2736**	**1.2744**
Bottom-up	1.2645	1.2757	**1.2723**	**1.2764**	**1.2796**	**1.2785**	**1.2736**	**1.2744**
Top-down	1.2591	1.2999	1.3147	1.3285	1.3455	1.3578	1.3541	1.3228
OLS	1.2494	1.2897	1.3071	1.3173	1.3310	1.3415	1.3387	1.3107
WLS (struct)	**1.2381**	**1.2672**	1.2771	1.2860	1.2963	1.3030	1.2998	1.2811

Table 3. MASE and RMSSE of forecasts generated by ARIMAX models.

	ARIMAX							
MASE	$h = 1$	$h = 2$	$h = 3$	$h = 4$	$h = 5$	$h = 6$	$h = 7$	$h = 1$–7
	Level 0: Total							
Base	0.7795	0.7863	0.7875	0.7799	0.7795	0.7756	0.7720	0.7800
Bottom-up	0.8141	0.8221	0.8221	0.8157	0.8161	0.8163	0.8151	0.8173
Top-down	0.7795	0.7863	0.7875	0.7799	0.7795	0.7756	0.7720	0.7800
OLS	0.7671	0.7744	0.7763	0.7693	0.7692	0.7659	0.7625	0.7692
WLS (struct)	**0.7587**	**0.7671**	**0.7697**	**0.7636**	**0.7643**	**0.7630**	**0.7609**	**0.7639**
	Level 1: Region							
Base	0.7406	0.7463	0.7457	0.7423	0.7433	0.7408	0.7377	0.7424
Bottom-up	0.7625	0.7679	0.7682	0.7643	0.7664	0.7666	0.7657	0.7659
Top-down	0.7532	0.7585	0.7579	0.7544	0.7545	0.7510	0.7483	0.7540
OLS	0.7422	0.7482	0.7473	0.7439	0.7445	0.7417	0.7395	0.7439
WLS (struct)	**0.7329**	**0.7386**	**0.7391**	**0.7361**	**0.7372**	**0.7361**	**0.7340**	**0.7363**
	Level 2: Store							
Base	**0.7268**	**0.7301**	**0.7291**	**0.7301**	**0.7318**	**0.7309**	**0.7295**	**0.7298**
Bottom-up	**0.7268**	**0.7301**	**0.7291**	**0.7301**	**0.7318**	**0.7309**	**0.7295**	**0.7298**
Top-down	0.7479	0.7513	0.7506	0.7511	0.7523	0.7505	0.7489	0.7504
OLS	0.7471	0.7502	0.7487	0.7492	0.7505	0.7490	0.7470	0.7488
WLS (struct)	0.7330	0.7362	0.7352	0.7360	0.7376	0.7363	0.7344	0.7355

RMSSE	$h = 1$	$h = 2$	$h = 3$	$h = 4$	$h = 5$	$h = 6$	$h = 7$	$h = 1$–7
	Level 0: Total							
Base	1.3318	1.3396	1.3441	1.3459	1.3519	1.3415	1.3320	1.3410
Bottom-up	1.4153	1.4075	1.4088	1.4093	1.4165	1.4141	1.4077	1.4113
Top-down	1.3318	1.3396	1.3441	1.3459	1.3519	1.3415	1.3320	1.3410
OLS	1.3207	1.3258	1.3298	1.3309	1.3375	1.3288	1.3187	1.3274
WLS (struct)	**1.3196**	**1.3205**	**1.3239**	**1.3249**	**1.3323**	**1.3260**	**1.3169**	**1.3234**
	Level 1: Region							
Base	1.2312	1.2279	1.2288	1.2304	1.2387	1.2344	1.2273	1.2312
Bottom-up	1.2811	1.2714	1.2710	1.2741	1.2814	1.2794	1.2755	1.2763
Top-down	1.2395	1.2404	1.2430	1.2468	1.2544	1.2473	1.2418	1.2447
OLS	1.2244	1.2244	1.2258	1.2293	1.2368	1.2305	1.2247	1.2280
WLS (struct)	**1.2208**	**1.2171**	**1.2182**	**1.2217**	**1.2297**	**1.2253**	**1.2196**	**1.2218**
	Level 2: Store							
Base	1.1414	1.1400	1.1375	1.1417	1.1455	1.1442	1.1411	1.1416
Bottom-up	1.1414	1.1400	1.1375	1.1417	1.1455	1.1442	1.1411	1.1416
Top-down	1.1440	1.1476	1.1488	1.1515	1.1563	1.1529	1.1496	1.1501
OLS	1.1329	1.1355	1.1328	1.1371	1.1412	1.1382	1.1341	1.1360
WLS (struct)	**1.1212**	**1.1226**	**1.1200**	**1.1245**	**1.1287**	**1.1264**	**1.1224**	**1.1237**

Table 4. MASE and RMSSE of forecasts generated by MLPs.

	MLP							
MASE	$h = 1$	$h = 2$	$h = 3$	$h = 4$	$h = 5$	$h = 6$	$h = 7$	$h = 1$–7
Level 0: Total								
Base	1.0199	1.0363	1.0260	1.0077	1.0131	1.0192	1.0110	1.0190
Bottom-up	1.0046	1.0109	1.0003	0.9931	0.9966	0.9980	0.9900	0.9991
Top-down	1.0199	1.0363	1.0260	1.0077	1.0131	1.0192	1.0110	1.0190
OLS	0.9999	1.0132	1.0071	0.9915	0.9948	1.0007	0.9931	1.0001
WLS (struct)	**0.9898**	**1.0005**	**0.9938**	**0.9816**	**0.9846**	**0.9881**	**0.9815**	**0.9886**
Level 1: Region								
Base	0.9263	0.9349	0.9340	0.9223	0.9263	0.9287	0.9225	0.9279
Bottom-up	0.9210	0.9249	**0.9176**	0.9150	0.9171	0.9180	0.9139	0.9182
Top-down	0.9383	0.9508	0.9457	0.9328	0.9392	0.9413	0.9371	0.9407
OLS	0.9299	0.9402	0.9360	0.9242	0.9301	0.9329	0.9275	0.9315
WLS (struct)	**0.9130**	**0.9217**	0.9177	**0.9092**	**0.9127**	**0.9143**	**0.9100**	**0.9141**
Level 2: Store								
Base	**0.8101**	**0.8118**	**0.8085**	**0.8090**	**0.8109**	**0.8112**	**0.8086**	**0.8100**
Bottom-up	**0.8101**	**0.8118**	**0.8085**	**0.8090**	**0.8109**	**0.8112**	**0.8086**	**0.8100**
Top-down	0.8306	0.8355	0.8325	0.8286	0.8319	0.8339	0.8284	0.8316
OLS	0.8311	0.8340	0.8321	0.8276	0.8313	0.8338	0.8280	0.8311
WLS (struct)	0.8195	0.8221	0.8200	0.8174	0.8200	0.8217	0.8171	0.8197

RMSSE	$h = 1$	$h = 2$	$h = 3$	$h = 4$	$h = 5$	$h = 6$	$h = 7$	$h = 1$–7
Level 0: Total								
Base	1.7521	1.7263	1.7094	1.7024	1.7084	1.7157	1.7050	1.7170
Bottom-up	1.7832	1.7510	1.7314	1.7253	1.7332	1.7333	1.7282	1.7408
Top-down	1.7521	1.7263	1.7094	1.7024	1.7084	1.7157	1.7050	1.7170
OLS	**1.7423**	**1.7089**	**1.6953**	**1.6879**	**1.6927**	**1.7014**	**1.6896**	**1.7026**
WLS (struct)	1.7466	1.7114	1.6970	1.6892	1.6945	1.7015	1.6918	1.7046
Level 1: Region								
Base	1.5545	1.5221	1.5145	1.5096	1.5141	1.5203	1.5109	1.5209
Bottom-up	1.5723	1.5443	1.5293	1.5290	1.5350	1.5346	1.5315	1.5394
Top-down	1.5549	1.5348	1.5239	1.5201	1.5275	1.5289	1.5238	1.5306
OLS	1.5514	1.5279	1.5160	1.5131	1.5204	1.5218	1.5155	1.5237
WLS (struct)	**1.5466**	**1.5185**	**1.5070**	**1.5047**	**1.5099**	**1.5131**	**1.5072**	**1.5153**
Level 2: Store								
Base	1.2776	1.2669	1.2582	1.2605	1.2642	1.2638	1.2608	1.2646
Bottom-up	1.2776	1.2669	1.2582	1.2605	1.2642	1.2638	1.2608	1.2646
Top-down	1.2753	1.2682	1.2597	1.2620	1.2666	1.2673	1.2613	1.2658
OLS	1.2749	1.2645	1.2564	1.2578	1.2622	1.2640	1.2574	1.2625
WLS (struct)	**1.2703**	**1.2589**	**1.2509**	**1.2528**	**1.2564**	**1.2580**	**1.2525**	**1.2571**

Table 5. MASE and RMSSE of forecasts generated by MLPs with regressors.

MLP with Regressors								
MASE	$h = 1$	$h = 2$	$h = 3$	$h = 4$	$h = 5$	$h = 6$	$h = 7$	$h = 1$–7
Level 0: Total								
Base	1.2115	1.1893	1.1752	1.1636	1.1617	1.1567	1.1527	1.1729
Bottom-up	**1.0954**	**1.0709**	**1.0555**	**1.0439**	**1.0418**	**1.0377**	**1.0337**	**1.0541**
Top-down	1.2115	1.1893	1.1752	1.1636	1.1617	1.1567	1.1527	1.1729
OLS	1.1962	1.1738	1.1596	1.1483	1.1467	1.1424	1.1390	1.1580
WLS (struct)	1.1580	1.1349	1.1203	1.1089	1.1072	1.1029	1.0994	1.1188
Level 1: Region								
Base	1.0562	1.0382	1.0278	1.0202	1.0196	1.0167	1.0139	1.0275
Bottom-up	**0.9910**	**0.9714**	**0.9603**	**0.9524**	**0.9511**	**0.9480**	**0.9447**	**0.9598**
Top-down	1.0665	1.0484	1.0380	1.0298	1.0282	1.0241	1.0203	1.0365
OLS	1.0578	1.0397	1.0292	1.0212	1.0198	1.0157	1.0122	1.0279
WLS (struct)	1.0288	1.0102	0.9995	0.9915	0.9904	0.9873	0.9842	0.9988
Level 2: Store								
Base	**0.8559**	**0.8454**	**0.8389**	**0.8365**	**0.8365**	**0.8347**	**0.8324**	**0.8401**
Bottom-up	**0.8559**	**0.8454**	**0.8389**	**0.8365**	**0.8365**	**0.8347**	**0.8324**	**0.8401**
Top-down	0.8927	0.8827	0.8767	0.8740	0.8738	0.8718	0.8694	0.8773
OLS	0.8933	0.8834	0.8774	0.8748	0.8748	0.8727	0.8703	0.8781
WLS (struct)	0.8776	0.8675	0.8613	0.8588	0.8588	0.8570	0.8548	0.8623

RMSSE	$h = 1$	$h = 2$	$h = 3$	$h = 4$	$h = 5$	$h = 6$	$h = 7$	$h = 1$–7
Level 0: Total								
Base	1.9013	1.8255	1.7965	1.7830	1.7824	1.7774	1.7739	1.8057
Bottom-up	**1.8506**	**1.7698**	**1.7369**	**1.7247**	**1.7246**	**1.7218**	**1.7194**	**1.7497**
Top-down	1.9013	1.8255	1.7965	1.7830	1.7824	1.7774	1.7739	1.8057
OLS	1.8896	1.8140	1.7851	1.7724	1.7727	1.7696	1.7672	1.7958
WLS (struct)	1.8705	1.7934	1.7633	1.7509	1.7513	1.7487	1.7465	1.7750
Level 1: Region								
Base	1.6630	1.6004	1.5786	1.5713	1.5726	1.5702	1.5679	1.5891
Bottom-up	**1.6326**	**1.5651**	**1.5404**	**1.5327**	**1.5332**	**1.5304**	**1.5279**	**1.5517**
Top-down	1.6670	1.6028	1.5804	1.5715	1.5711	1.5659	1.5621	1.5887
OLS	1.6599	1.5969	1.5745	1.5661	1.5660	1.5611	1.5577	1.5832
WLS (struct)	1.6431	1.5779	1.5548	1.5469	1.5476	1.5447	1.5423	1.5653
Level 2: Store								
Base	**1.3141**	**1.2834**	**1.2698**	**1.2684**	**1.2693**	**1.2677**	**1.2651**	**1.2768**
Bottom-up	**1.3141**	**1.2834**	**1.2698**	**1.2684**	**1.2693**	**1.2677**	**1.2651**	**1.2768**
Top-down	1.3296	1.3005	1.2882	1.2862	1.2867	1.2842	1.2811	1.2938
OLS	1.3268	1.2981	1.2858	1.2840	1.2847	1.2823	1.2794	1.2916
WLS (struct)	1.3180	1.2884	1.2757	1.2741	1.2750	1.2732	1.2706	1.2821

5. Conclusions

The inherent uncertainty in consumer demand requires that retailers employ robust sales forecasting methods to manage stock efficiently and maintain high-quality customer service. While traditional forecasting primarily relies on historical data, our findings emphasize the value of integrating additional influencing factors, such as promotional activities, to improve accuracy. However, incorporating these variables increases forecast complexity and calls for more advanced methodologies.

In the retail environment, sales forecasts must accommodate a diverse range of products organized hierarchically, allowing for forecasts at various levels of aggregation. Our study highlights the effectiveness of hierarchical forecasting in this context while also

noting the lack of consensus on the optimal forecasting model for each level or the most effective strategy for reconciling forecasts.

The primary aim of this research was to improve forecasting accuracy within the retail sector by integrating additional information into a hierarchical forecasting framework, thereby enhancing decision-making efficiency, particularly for larger retailers. Using data from the Jerónimo Martins Group encompassing sales and pricing at Pingo Doce stores from 3 January 2012 to 27 April 2015, we generated base forecasts across various hierarchical levels employing ARIMA models alongside multi-layer perceptrons (MLPs). Our approach began with sales data alone and progressively included additional variables such as pricing and calendar events. The selection of MLP was motivated by recent research that highlights deep learning's ability to model complex, nonlinear relationships, thereby providing potential advantages over traditional forecasting models. One of the key strengths of MLPs is their relative simplicity and ease of training, which make them accessible to a wider range of practitioners. In contrast to more complex architectures, MLPs demand fewer computational resources and require less intensive hyperparameter tuning. This streamlined approach allows for rapid prototyping and experimentation, establishing MLPs as a compelling option for initial exploration in numerous deep learning applications.

In order to reconcile base forecasts, our study implemented both bottom-up and top-down methods, in addition to optimal reconciliation strategies utilizing ordinary least squares (OLS) and weighted least squares (WLS) estimators. We assessed the performance of the forecasting models using mean absolute scaled error (MASE) and root mean squared scaled error (RMSSE) as accuracy metrics.

Our results indicate that for forecast horizons extending beyond one day, MLPs yield more accurate forecasts than ARIMA models. Additionally, we found that incorporating promotional activities into ARIMA models enhances forecast accuracy, although this benefit did not extend to the MLP model. This discrepancy may arise from differences in methods of integrating regressors into each forecasting approach.

While this study utilized a single dataset from a major Portuguese retailer, which may limit the generalizability of our findings, the hierarchical forecasting framework and the integration of promotional data provide a transferable methodology that can be applied to diverse retail contexts.

Our goal was to explore methods that balance accuracy with practical considerations, particularly for retailers managing large portfolios of products, where computational efficiency is critical. The multi-layer perceptron (MLP) model, which we used, is the simplest neural network architecture within the broader class of deep learning models. It is much easier to train, requires fewer computational resources, and has fewer hyperparameters to tune compared to more complex deep learning models such as LSTM or CNN models. This simplicity makes MLPs a practical and efficient solution for retailers, as they reduce training time and computing power demands while still capturing nonlinear relationships in data. For retailers with thousands of SKUs and limited computational resources, MLPs offers a highly accessible and scalable forecasting model, ensuring that forecasts can be produced quickly and without requiring extensive infrastructure. This approach fits the context of retail forecasting, where the focus is often on operational efficiency and delivering reliable results in a timely manner. Therefore, we prioritized computational simplicity and ease of deployment, which are key factors in real-world retail applications. As future work, we suggest exploring more complex deep learning models such as LSTM or CNN models to compare their forecasting accuracy and computational efficiency with those of MLPs, especially in contexts where additional computational resources are available and task complexity justifies their use. To further improve the model's robustness, future work could explore more advanced feature engineering techniques. For example, decomposing the promotional effect into different types of promotions (e.g., percentage discounts and multi-buy offers) could provide greater granularity and improve interpretability regarding the specific impact of each promotion. Additionally, incorporating external factors such as competitor pricing, store-specific events, and broader economic indicators could be valu-

able in capturing other nuances affecting sales performance. This approach would provide a more holistic understanding of the factors influencing demand, potentially improving both forecast accuracy and decision making. By addressing these avenues, we can continue to refine forecasting methodologies and improve decision-making processes for retailers operating in increasingly complex market environments.

Author Contributions: Conceptualization, M.T., J.M.O., and P.R.; methodology, M.T., J.M.O., and P.R.; software, M.T., J.M.O., and P.R.; validation, M.T., J.M.O., and P.R.; formal analysis, M.T., J.M.O., and P.R.; investigation, M.T., J.M.O., and P.R.; resources, M.T., J.M.O., and P.R.; data curation, M.T., J.M.O., and P.R.; writing—original draft preparation, M.T., J.M.O., and P.R.; writing—review and editing, M.T., J.M.O., and P.R.; visualization, M.T., J.M.O., and P.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Restrictions apply to the availability of these data due to privacy considerations.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Villegas, M.A.; Pedregal, D.J. Supply chain decision support systems based on a novel hierarchical forecasting approach. *Decis. Support Syst.* **2018**, *114*, 29–36. [CrossRef]
2. Fildes, R.; Ma, S.; Kolassa, S. Retail forecasting: Research and practice. *Int. J. Forecast.* **2022**, *38*, 1283–1318. [CrossRef]
3. Syntetos, A.A.; Babai, Z.; Boylan, J.E.; Kolassa, S.; Nikolopoulos, K. Supply chain forecasting: Theory, practice, their gap and the future. *Eur. J. Oper. Res.* **2016**, *252*, 1–26. [CrossRef]
4. Oliveira, J.M.; Ramos, P. Investigating the Accuracy of Autoregressive Recurrent Networks Using Hierarchical Aggregation Structure-Based Data Partitioning. *Big Data Cogn. Comput.* **2023**, *7*, 100. [CrossRef]
5. Petropoulos, F.; Apiletti, D.; Assimakopoulos, V.; Babai, M.Z.; Barrow, D.K.; Ben Taieb, S.; Bergmeir, C.; Bessa, R.J.; Bijak, J.; Boylan, J.E.; et al. Forecasting: Theory and practice. *Int. J. Forecast.* **2022**, *38*, 705–871. [CrossRef]
6. Abolghasemi, M.; Beh, E.; Tarr, G.; Gerlach, R. Demand forecasting in supply chain: The impact of demand volatility in the presence of promotion. *Comput. Ind. Eng.* **2020**, *142*, 106380. [CrossRef]
7. Abolghasemi, M.; Hurley, J.; Eshragh, A.; Fahimnia, B. Demand forecasting in the presence of systematic events: Cases in capturing sales promotions. *Int. J. Prod. Econ.* **2020**, *230*, 107892. [CrossRef]
8. Ramos, P.; Santos, N.; Rebelo, R. Performance of state space and ARIMA models for consumer retail sales forecasting. *Robot. Comput.-Integr. Manuf.* **2015**, *34*, 151–163. [CrossRef]
9. Fildes, R.; Goodwin, P.; Lawrence, M.; Nikolopoulos, K. Effective forecasting and judgmental adjustments: An empirical evaluation and strategies for improvement in supply-chain planning. *Int. J. Forecast.* **2009**, *25*, 3–23. [CrossRef]
10. Davydenko, A.; Fildes, R. Measuring forecasting accuracy: The case of judgmental adjustments to SKU-level demand forecasts. *Int. J. Forecast.* **2013**, *29*, 510–522. [CrossRef]
11. Franses, P.H.; Legerstee, R. Do experts' adjustments on model-based SKU-level forecasts improve forecast quality? *J. Forecast.* **2010**, *29*, 331–340. [CrossRef]
12. Seaman, B. Considerations of a retail forecasting practitioner. *Int. J. Forecast.* **2018**, *34*, 822–829. [CrossRef]
13. Seaman, B.; Bowman, J. Applicability of the M5 to Forecasting at Walmart. *Int. J. Forecast.* **2022**, *38*, 1468–1472. [CrossRef]
14. Trapero, J.R.; Pedregal, D.J.; Fildes, R.; Kourentzes, N. Analysis of judgmental adjustments in the presence of promotions. *Int. J. Forecast.* **2013**, *29*, 234–243. [CrossRef]
15. Lee, H.L.; So, K.C.; Tang, C.S. The Value of Information Sharing in a Two-Level Supply Chain. *Manag. Sci.* **2000**, *46*, 626–643. [CrossRef]
16. Yu, Z.; Yan, H.; Edwin Cheng, T. Benefits of information sharing with supply chain partnerships. *Ind. Manag. Data Syst.* **2001**, *101*, 114–121. [CrossRef]
17. Hosoda, T.; Naim, M.M.; Disney, S.M.; Potter, A. Is there a benefit to sharing market sales information? Linking theory and practice. *Comput. Ind. Eng.* **2008**, *54*, 315–326. [CrossRef]
18. Trapero, J.R.; Kourentzes, N.; Fildes, R. Impact of information exchange on supplier forecasting performance. *Omega* **2012**, *40*, 738–747. [CrossRef]
19. Lee, H.L.; Padmanabhan, V.; Whang, S. Information Distortion in a Supply Chain: The Bullwhip Effect. *Manag. Sci.* **1997**, *43*, 546–558. [CrossRef]
20. Lee, H.L.; Whang, S. Information sharing in a supply chain. *Int. J. Manuf. Technol. Manag.* **2000**, *1*, 79–93. [CrossRef]
21. Jain, A.; Rudi, N.; Wang, T. Demand Estimation and Ordering Under Censoring: Stock-Out Timing Is (Almost) All You Need. *Oper. Res.* **2014**, *63*, 134–150. [CrossRef]

22. Vulcano, G.; van Ryzin, G.; Ratliff, R. Estimating Primary Demand for Substitutable Products from Sales Transaction Data. *Oper. Res.* **2012**, *60*, 313–334. [CrossRef]
23. Kim, S.; Kim, H.; Lu, J.C. A practical approach to measuring the impacts of stockouts on demand. *J. Bus. Ind. Mark.* **2019**, *34*, 891–901. [CrossRef]
24. Boone, T.; Boylan, J.E.; Fildes, R.; Ganeshan, R.; Sanders, N. Perspectives on supply chain forecasting. *Int. J. Forecast.* **2019**, *35*, 121–127. [CrossRef]
25. Beutel, A.L.; Minner, S. Safety stock planning under causal demand forecasting. *Int. J. Prod. Econ.* **2012**, *140*, 637–645. [CrossRef]
26. Fisher, M.; Vaidyanathan, R. A Demand Estimation Procedure for Retail Assortment Optimization with Results from Implementations. *Manag. Sci.* **2014**, *60*, 2401–2415. [CrossRef]
27. Kahn, K.B. Solving the problems of new product forecasting. *Bus. Horizons* **2014**, *57*, 607–615. [CrossRef]
28. Fisher, M.; Raman, A. Using Data and Big Data in Retailing. *Prod. Oper. Manag.* **2018**, *27*, 1665–1669. [CrossRef]
29. Boone, T.; Ganeshan, R.; Jain, A.; Sanders, N.R. Forecasting sales in the supply chain: Consumer analytics in the big data era. *Int. J. Forecast.* **2019**, *35*, 170–180. [CrossRef]
30. Boone, T.; Ganeshan, R.; Hicks, R.L.; Sanders, N.R. Can Google Trends Improve Your Sales Forecast? *Prod. Oper. Manag.* **2018**, *27*, 1770–1774. [CrossRef]
31. Chern, C.C.; Wei, C.P.; Shen, F.Y.; Fan, Y.N. A sales forecasting model for consumer products based on the influence of online word-of-mouth. *Inf. Syst. e-Bus. Manag.* **2015**, *13*, 445–473. [CrossRef]
32. Cui, R.; Gallino, S.; Moreno, A.; Zhang, D.J. The Operational Value of Social Media Information. *Prod. Oper. Manag.* **2018**, *27*, 1749–1769. [CrossRef]
33. Lau, R.Y.K.; Zhang, W.; Xu, W. Parallel Aspect-Oriented Sentiment Analysis for Sales Forecasting with Big Data. *Prod. Oper. Manag.* **2018**, *27*, 1775–1794. [CrossRef]
34. Box, G.E.P.; Jenkins, G.M.; Reinsel, G.C.; Ljung, G.M. *Time Series Analysis: Forecasting and Control*, 5th ed.; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2015.
35. Najafabadi, M.M.; Villanustre, F.; Khoshgoftaar, T.M.; Seliya, N.; Wald, R.; Muharemagic, E. Deep learning applications and challenges in big data analytics. *Int. J. Manuf. Technol. Manag.* **2015**, *2*, 1. [CrossRef]
36. Alon, I.; Qi, M.; Sadowski, R.J. Forecasting aggregate retail sales: A comparison of artificial neural networks and traditional methods. *J. Retail. Consum. Serv.* **2001**, *8*, 147–156. [CrossRef]
37. Chu, C.W.; Zhang, G.P. A comparative study of linear and nonlinear models for aggregate retail sales forecasting. *Int. J. Prod. Econ.* **2003**, *86*, 217–231. [CrossRef]
38. Zhang, G.P.; Qi, M. Neural network forecasting for seasonal and trend time series. *Eur. J. Oper. Res.* **2005**, *160*, 501–514. [CrossRef]
39. Nelson, M.; Hill, T.; Remus, W.; O'Connor, M. Time series forecasting using neural networks: Should the data be deseasonalized first? *J. Predict.* **1999**, *18*, 359–367. [CrossRef]
40. Aras, S.; Kocakoç, İ.D.; Polat, C. Comparative study on retail sales forecasting between single and combination methods. *J. Bus. Econ. Manag.* **2017**, *18*, 803–832. [CrossRef]
41. Aburto, L.; Weber, R. Improved supply chain management based on hybrid demand forecasts. *Appl. Soft Comput.* **2007**, *7*, 126–144. [CrossRef]
42. Vallés-Pérez, I.; Soria-Olivas, E.; Martínez-Sober, M.; Serrano-López, A.J.; Gómez-Sanchís, J.; Mateo, F. Approaching sales forecasting using recurrent neural networks and transformers. *Expert Syst. Appl.* **2022**, *201*, 116993. [CrossRef]
43. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
44. Abbasimehr, H.; Shabani, M.; Yousefi, M. An optimized model using LSTM network for demand forecasting. *Comput. Ind. Eng.* **2020**, *143*, 106435. [CrossRef]
45. Ensafi, Y.; Amin, S.H.; Zhang, G.; Shah, B. Time-series forecasting of seasonal items sales using machine learning—A comparative analysis. *Int. J. Inf. Manag. Data Insights* **2022**, *2*, 100058. [CrossRef]
46. Falatouri, T.; Darbanian, F.; Brandtner, P.; Udokwu, C. Predictive Analytics for Demand Forecasting—A Comparison of SARIMA and LSTM in Retail SCM. *Procedia Comput. Sci.* **2022**, *200*, 993–1003. [CrossRef]
47. Punia, S.; Nikolopoulos, K.; Singh, S.P.; Madaan, J.K.; Litsiou, K. Deep learning with long short-term memory networks and random forests for demand forecasting in multi-channel retail. *Int. J. Prod. Res.* **2020**, *58*, 4964–4979. [CrossRef]
48. Weng, T.; Liu, W.; Xiao, J. Supply chain sales forecasting based on lightGBM and LSTM combination model. *Ind. Manag. Data Syst.* **2020**, *120*, 265–279. [CrossRef]
49. Wang, Y.; Smola, A.; Maddix, D.; Gasthaus, J.; Foster, D.; Januschowski, T. Deep Factors for Forecasting. In *Proceedings of Machine Learning Research , Proceedings of the 36th International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019*; Chaudhuri, K., Salakhutdinov, R., Eds.; PMLR: New York, NY, USA, 2019; Volume 97, pp. 6607–6617. [CrossRef]
50. Wang, J.; Liu, G.Q.; Liu, L. A Selection of Advanced Technologies for Demand Forecasting in the Retail Industry. In Proceedings of the 2019 IEEE 4th International Conference on Big Data Analytics (ICBDA), Suzhou, China, 15–18 March 2019; pp. 317–320. [CrossRef]
51. LeCun, Y.; Boser, B.; Denker, J.; Henderson, D.; Howard, R.; Hubbard, W.; Jackel, L. Handwritten Digit Recognition with a Back-Propagation Network. In *Advances in Neural Information Processing Systems*; Touretzky, D., Ed.; Morgan-Kaufmann: Burlington, MA, USA, 1989; Volume 2.
52. Ma, S.; Fildes, R. Retail sales forecasting with meta-learning. *Eur. J. Oper. Res.* **2021**, *288*, 111–128. [CrossRef]

53. Kaunchi, P.; Jadhav, T.; Dandawate, Y.; Marathe, P. Future Sales Prediction For Indian Products Using Convolutional Neural Network-Long Short Term Memory. In Proceedings of the 2021 2nd Global Conference for Advancement in Technology (GCAT), Bangalore, India, 1–3 October 2021; pp. 1–5. [CrossRef]
54. Liu, Y.; Lan, K.; Huang, F.; Cao, X.; Feng, B.; Zhu, B. An Aggregate Store Sales Forecasting Framework based on ConvLSTM. In Proceedings of the 2021 5th International Conference on Compute and Data Analysis, ICCDA '21, New York, NY, USA, 2–4 February 2021; pp. 67–72. [CrossRef]
55. Nithin, S.S.J.; Rajasekar, T.; Jayanthy, S.; Karthik, K.; Rithick, R.R. Retail Demand Forecasting using CNN-LSTM Model. In Proceedings of the 2022 International Conference on Electronics and Renewable Systems (ICEARS), Tuticorin, India, 16–18 March 2022; pp. 1751–1756. [CrossRef]
56. Bandara, K.; Shi, P.; Bergmeir, C.; Hewamalage, H.; Tran, Q.; Seaman, B. Sales Demand Forecast in E-commerce Using a Long Short-Term Memory Neural Network Methodology. In *Neural Information Processing, Proceedings of the 26th International Conference, ICONIP 2019, Sydney, Australia, 12–15 December 2019*; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2019; Volume 11955, pp. 462–474. [CrossRef]
57. Bandara, K.; Bergmeir, C.; Smyl, S. Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach. *Expert Syst. Appl.* **2020**, *140*, 112896. [CrossRef]
58. Bandara, K.; Hewamalage, H.; Liu, Y.H.; Kang, Y.; Bergmeir, C. Improving the accuracy of global forecasting models using time series data augmentation. *Pattern Recognit.* **2021**, *120*, 108148. [CrossRef]
59. Pan, H.; Zhou, H. Study on convolutional neural network and its application in data mining and sales forecasting for E-commerce. *Electron. Commer. Res.* **2020**, *20*, 297–320. [CrossRef]
60. Chen, K. An Online Retail Prediction Model Based on AGA-LSTM Neural Network. In Proceedings of the 2020 2nd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI), Taiyuan, China, 23–25 October 2020; pp. 145–149. [CrossRef]
61. He, Q.Q.; Wu, C.; Si, Y.W. LSTM with particle Swam optimization for sales forecasting. *Electron. Commer. Res. Appl.* **2022**, *51*, 101118. [CrossRef]
62. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.u.; Polosukhin, I. Attention is All you Need. In *Advances in Neural Information Processing Systems*; Morgan-Kaufmann: Burlington, MA, USA, 2017; Volume 30, pp. 5998–6008.
63. Wu, H.; Xu, J.; Wang, J.; Long, M. Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting. In *Advances in Neural Information Processing Systems*; Morgan-Kaufmann: Burlington, MA, USA, 2021; Volume 34, pp. 22419–22430.
64. Lim, B.; Arık, S.Ö.; Loeff, N.; Pfister, T. Temporal Fusion Transformers for interpretable multi-horizon time series forecasting. *Int. J. Forecast.* **2021**, *37*, 1748–1764. [CrossRef]
65. Zhou, H.; Zhang, S.; Peng, J.; Zhang, S.; Li, J.; Xiong, H.; Zhang, W. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. *Proc. AAAI Conf. Artif. Intell.* **2021**, *35*, 11106–11115. [CrossRef]
66. Zhou, T.; Ma, Z.; Wen, Q.; Wang, X.; Sun, L.; Jin, R. FEDformer: Frequency Enhanced Decomposed Transformer for Long-term Series Forecasting. In *Proceedings of Machine Learning Research, Proceedings of the 39th International Conference on Machine Learning, Baltimore, MD, USA, 17–23 July 2022*; Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., Sabato, S., Eds.: PMLR: New York, NY, USA, 2022; Volume 162, pp. 27268–27286.
67. Nie, Y.; Nguyen, N.H.; Sinthong, P.; Kalagnanam, J. A Time Series is Worth 64 Words: Long-term Forecasting with Transformers. In Proceedings of the Eleventh International Conference on Learning Representations, Kigali, Rwanda, 1–5 May 2023.
68. Tong, J.; Xie, L.; Yang, W.; Zhang, K.; Zhao, J. Enhancing time series forecasting: A hierarchical transformer with probabilistic decomposition representation. *Inf. Sci.* **2023**, *647*, 119410. [CrossRef]
69. Oliveira, J.M.; Ramos, P. Evaluating the Effectiveness of Time Series Transformers for Demand Forecasting in Retail. *Mathematics* **2024**, *12*, 2728. [CrossRef]
70. Wellens, A.P.; Boute, R.N.; Udenio, M. Simplifying tree-based methods for retail sales forecasting with explanatory variables. *Eur. J. Oper. Res.* **2024**, *314*, 523–539. [CrossRef]
71. Ansari, A.F.; Stella, L.; Turkmen, C.; Zhang, X.; Mercado, P.; Shen, H.; Shchur, O.; Rangapuram, S.S.; Arango, S.P.; Kapoor, S.; et al. Chronos: Learning the Language of Time Series. *arXiv* **2024**, arXiv:2403.07815. [CrossRef]
72. Woo, G.; Liu, C.; Kumar, A.; Xiong, C.; Savarese, S.; Sahoo, D. Unified Training of Universal Time Series Forecasting Transformers. *arXiv* **2024**, arXiv:2402.02592. [CrossRef]
73. Das, A.; Kong, W.; Sen, R.; Zhou, Y. A decoder-only foundation model for time-series forecasting. *arXiv* **2024**, arXiv:2310.10688. [CrossRef]
74. Rasul, K.; Ashok, A.; Williams, A.R.; Ghonia, H.; Bhagwatkar, R.; Khorasani, A.; Bayazi, M.J.D.; Adamopoulos, G.; Riachi, R.; Hassen, N.; et al. Lag-Llama: Towards Foundation Models for Probabilistic Time Series Forecasting. *arXiv* **2024**, arXiv:2310.08278. [CrossRef]
75. Garza, A.; Challu, C.; Mergenthaler-Canseco, M. TimeGPT-1. *arXiv* **2024**, arXiv:2310.03589. [CrossRef]
76. Oliveira, J.M.; Ramos, P. Cross-Learning-Based Sales Forecasting Using Deep Learning via Partial Pooling from Multi-level Data. In *Engineering Applications of Neural Networks*; Iliadis, L., Maglogiannis, I., Alonso, S., Jayne, C., Pimenidis, E., Eds.; Springer Nature: Cham, Switzerland, 2023; pp. 279–290. [CrossRef]

77. Kourentzes, N.; Petropoulos, F.; Trapero, J.R. Improving forecasting by estimating time series structural components across multiple frequencies. *Int. J. Forecast.* **2014**, *30*, 291–302. [CrossRef]
78. Athanasopoulos, G.; Hyndman, R.J.; Kourentzes, N.; Petropoulos, F. Forecasting with temporal hierarchies. *Eur. J. Oper. Res.* **2017**, *262*, 60–74. [CrossRef]
79. Zotteri, G.; Kalchschmidt, M.; Caniato, F. The impact of aggregation level on forecasting performance. *Int. J. Prod. Econ.* **2005**, *93–94*, 479–491. [CrossRef]
80. Athanasopoulos, G.; Gamakumara, P.; Panagiotelis, A.; Hyndman, R.J.; Affan, M. Hierarchical Forecasting. In *Macroeconomic Forecasting in the Era of Big Data*; Fuleky, P., Ed.; Springer: Cham, Switzerland, 2020; Volume 52, Chapter 21, pp. 689–719. [CrossRef]
81. Athanasopoulos, G.; Ahmed, R.A.; Hyndman, R.J. Hierarchical forecasts for Australian domestic tourism. *Int. J. Forecast.* **2009**, *25*, 146–166. [CrossRef]
82. Gross, C.W.; Sohl, J.E. Disaggregation methods to expedite product line forecasting. *J. Forecast.* **1990**, *9*, 233–254. [CrossRef]
83. Hyndman, R.J.; Ahmed, R.A.; Athanasopoulos, G.; Shang, H.L. Optimal combination forecasts for hierarchical time series. *Comput. Stat. Data Anal.* **2011**, *55*, 2579–2589. [CrossRef]
84. Hyndman, R.J.; Lee, A.J.; Wang, E. Fast computation of reconciled forecasts for hierarchical and grouped time series. *Comput. Stat. Data Anal.* **2016**, *97*, 16–32. [CrossRef]
85. Wickramasuriya, S.L.; Athanasopoulos, G.; Hyndman, R.J. Optimal forecast reconciliation for hierarchical and grouped time series through trace minimization. *J. Am. Stat. Assoc.* **2019**, *114*, 804–819. [CrossRef]
86. Spiliotis, E.; Abolghasemi, M.; Hyndman, R.J.; Petropoulos, F.; Assimakopoulos, V. Hierarchical forecast reconciliation with machine learning. *Appl. Soft Comput.* **2021**, *112*, 107756. [CrossRef]
87. Pennings, C.L.; van Dalen, J. Integrated hierarchical forecasting. *Eur. J. Oper. Res.* **2017**, *263*, 412–418. [CrossRef]
88. Oliveira, J.M.; Ramos, P. Assessing the Performance of Hierarchical Forecasting Methods on the Retail Sector. *Entropy* **2019**, *21*. [CrossRef]
89. Divakar, S.; Ratchford, B.T.; Shankar, V. CHAN4CAST: A Multichannel, Multiregion Sales Forecasting Model and Decision Support System for Consumer Packaged Goods. *Mark. Sci.* **2005**, *24*, 334–350. [CrossRef]
90. Ramanathan, U.; Muyldermans, L. Identifying demand factors for promotional planning and forecasting: A case of a soft drink company in the UK. *Int. J. Prod. Econ.* **2010**, *128*, 538–545. [CrossRef]
91. Steinker, S.; Hoberg, K.; Thonemann, U.W. The Value of Weather Information for E-Commerce Operations. *Prod. Oper. Manag.* **2017**, *26*, 1854–1874. [CrossRef]
92. Liu, X.; Ichise, R. Food Sales Prediction with Meteorological Data—A Case Study of a Japanese Chain Supermarket. In *Data Mining and Big Data*; Tan, Y., Takagi, H., Shi, Y., Eds.; Springer International Publishing: Cham, Switzerland, 2017; pp. 93–104.
93. Hirche, M.; Haensch, J.; Lockshin, L. Comparing the day temperature and holiday effects on retail sales of alcoholic beverages—A time-series analysis. *Int. J. Wine Bus. Res.* **2021**, *33*, 432–455. [CrossRef]
94. Verstraete, G.; Aghezzaf, E.H.; Desmet, B. A data-driven framework for predicting weather impact on high-volume low-margin retail products. *J. Retail. Consum. Serv.* **2019**, *48*, 169–177. [CrossRef]
95. Badorf, F.; Hoberg, K. The impact of daily weather on retail sales: An empirical study in brick-and-mortar stores. *J. Retail. Consum. Serv.* **2020**, *52*, 101921. [CrossRef]
96. Ramanathan, U.; Muyldermans, L. Identifying the underlying structure of demand during promotions: A structural equation modelling approach. *Expert Syst. Appl.* **2011**, *38*, 5544–5552. [CrossRef]
97. Ali, Ö.G.; Sayın, S.; van Woensel, T.; Fransoo, J. SKU demand forecasting in the presence of promotions. *Expert Syst. Appl.* **2009**, *36*, 12340–12348. [CrossRef]
98. Arunraj, N.S.; Ahrens, D. A hybrid seasonal autoregressive integrated moving average and quantile regression for daily food sales forecasting. *Int. J. Prod. Econ.* **2015**, *170*, 321–335. [CrossRef]
99. Arunraj, N.S.; Ahrens, D.; Fernandes, M. Application of SARIMAX Model to Forecast Daily Sales in Food Retail Industry. *Int. J. Oper. Res. Inf. Syst.* **2016**, *7*, 1–21. [CrossRef]
100. Huber, J.; Stuckenschmidt, H. Daily retail demand forecasting using machine learning with emphasis on calendric special days. *Int. J. Forecast.* **2020**, *36*, 1420–1438. [CrossRef]
101. Guo, Z.; Wong, W.; Li, M. A multivariate intelligent decision-making model for retail sales forecasting. *Decis. Support Syst.* **2013**, *55*, 247–255. [CrossRef]
102. Huang, T.; Fildes, R.; Soopramanien, D. The value of competitive information in forecasting FMCG retail product sales and the variable selection problem. *Eur. J. Oper. Res.* **2014**, *237*, 738–748. [CrossRef]
103. Ma, S.; Fildes, R.; Huang, T. Demand forecasting with high dimensional data: The case of SKU retail sales forecasting with intra- and inter-category promotional information. *Eur. J. Oper. Res.* **2016**, *249*, 245–257. [CrossRef]
104. Trapero, J.R.; Kourentzes, N.; Fildes, R. On the identification of sales forecasting models in the presence of promotions. *J. Oper. Res. Soc.* **2015**, *66*, 299–307. [CrossRef]
105. Hyndman, R.J.; Athanasopoulos, G. *Forecasting: Principles and Practice*, 3rd ed.; Online Open-Access Textbooks; Monash University: Clayton, Australia, 2021. Available online: https://OTexts.com/fpp3/ (accessed on 1 April 2024).
106. Schäfer, J.; Strimmer, K. A Shrinkage Approach to Large-Scale Covariance Matrix Estimation and Implications for Functional Genomics. *Stat. Appl. Genet. Mol. Biol.* **2005**, *4*, 151–163. [CrossRef]

107. Ramos, P.; Oliveira, J.M. A procedure for identification of appropriate state space and ARIMA models based on time-series cross-validation. *Algorithms* **2016**, *9*, 76. [CrossRef]
108. Ramos, P.; Oliveira, J.M.; Kourentzes, N.; Fildes, R. Forecasting Seasonal Sales with Many Drivers: Shrinkage or Dimensionality Reduction? *Appl. Syst. Innov.* **2023**, *6*, 3. [CrossRef]
109. Ramos, P.; Oliveira, J.M. Robust Sales forecasting Using Deep Learning with Static and Dynamic Covariates. *Appl. Syst. Innov.* **2023**, *6*, 85. [CrossRef]
110. Zhang, A.; Lipton, Z.C.; Li, M.; Smola, A.J. *Dive into Deep Learning*; Cambridge University Press: Cambridge, UK, 2023. Available online: https://D2L.ai (accessed on 15 June 2024).
111. Goodfellow, I.J.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016. Available online: http://www.deeplearningbook.org (accessed on 15 June 2024).
112. R Development Core Team. *R: A Language and Environment for Statistical Computing*; R Foundation for Statistical Computing: Vienna, Austria, 2021.
113. Hyndman, R.J.; Koehler, A.B. Another look at measures of forecast accuracy. *Int. J. Forecast.* **2006**, *22*, 679–688. [CrossRef]