DSC412 Final Project Report

Demand Forecasting Analysis

Abstract

This project addresses a critical challenge in supply chain management: accurately forecasting product demand. Leveraging five years of sales data across multiple stores and products, we developed machine learning models, including the Random Forest Regressor, to forecast demand. The approach integrates feature engineering, time series analysis, and model tuning, achieving significant accuracy improvements over traditional methods. Results demonstrate potential for reducing inventory costs and improving operational efficiency.

Background

Demand forecasting plays a pivotal role in supply chain operations, directly impacting inventory control, cost optimization, and customer satisfaction. Poor demand predictions often result in stockouts or excess inventory, increasing operational costs. This project focuses on utilizing advanced machine learning models to address these issues and provide actionable insights.

Data Analysis

The data analysis followed the overall Exploratory Data Analysis to discover prominent patterns and trends within the dataset. To understand the temporal and categorical relationship, different visualization tools like histograms, line plots, and heat maps were put into action. Time-series decomposition has also been done to understand overall trends, seasonal variations, and residuals in the sales data. A deeper understanding of the variation and relationships among stores, products, and sales figures was obtained from statistical methods, including the calculation of averages, standard deviations, and correlations.

A number of key insights emerged from the analysis. Sales data showed strong seasonal patterns, with predictable peaks and troughs at specific times of the year. Variation across stores and products highlighted that certain entities consistently outperform others, pointing to a need for customized forecasting strategies. Therefore, when aggregated to weekly sales from daily data, smoother trends with reduced noise also made the forecasting process easier. The other important insight was that of lagged effects; sales of certain products depended on the sales volumes of previous weeks, indicating that lagged variables could be good predictors.

These insights found prominent mentions in the selection of models. Because non-linear relationships and temporal dependencies were observed in the data, this led to the selection of the Random Forest Regressor as the machine learning model most befitting such data complexity. The presence of seasonal trends and lagged variables as predictors further emphasized feature engineering, resulting in the addition of rolling averages and time-based indicators. While baseline models, such as moving averages and ARIMA, were initially tested to serve as benchmarks, their inability to fully capture complex relationships reaffirmed machine learning methods as the preferred choice.

Numerous analysis techniques were conducted in deriving these insights. Time-series visualizations were important for identifying trends, seasonality, and outliers, guiding feature engineering strategies. Heatmaps allowed for the analysis of variable correlations, guiding the identification of the strongest predictors of sales. Aggregation to weekly intervals helped to reduce noise and computational efficiency, whereas feature engineering added important variables, such as moving averages and lagged sales, in order to include temporal dynamics within the modelling process. These techniques had been used since they effectively underlined the underlying pattern of the data, informing directly model design and improving in such a way the general predictive accuracy.

Data Augmentation

The dataset provided by G. Biswal [2] was already clean and well-structured, which minimized the need for extensive cleaning. This ensured that no major inconsistencies, such as missing values or duplicates, needed to be addressed. The changes made to the dataset, such as encoding categorical variables and creating additional features like time-based indicators and lagged variables, align with the approach mentioned

earlier to enhance the dataset's usability for machine learning and forecasting purposes. These modifications were essential for adapting the dataset to the specific requirements of the Random Forest Regressor model.

The following modifications were made to increase the quality and predictive power of the dataset. Feature engineering was done to include time-based features such as day of the week, month, and year to capture seasonality and temporal trends. Lagged variables and rolling averages were also added to include the influence of past sales on current and future trends. These modifications provided the model with richer contextual information, improving its ability to forecast demand accurately.

Data aggregation and innovative organization were also done to speed up the analysis. Examples include the aggregation of daily sales into weekly totals to reduce noise and simplify the detection of long-term patterns. Sales statistics were also summarized by store and product to show performance variation and direct model customization. This structured organization of data not only enhanced the process of modeling but also availed actionable insights into store-product dynamics and demand variability. These steps collectively prepared the dataset for effective machine learning application.

Model Selection

Considering that this project will deal with large datasets that exhibit a wide variety of features, the Random Forest Regressor has been chosen as the main model. The reasons include the ability to work with such complex, nonlinear relationships and also to provide resistance to overfitting. This choice was influenced by the need to capture intricate patterns in sales data, such as seasonality, lag effects, and variability across stores and products. Additionally, the inherent feature importance metrics from Random Forest provided insightful information about which variables most greatly impacted sales predictions.

The model was built from scratch using Python's scikit-learn library and thus allowed full customization of the training process. Now, it was time to optimize the performance by tuning parameters such as the number of estimators, maximum tree depth, and the minimum samples per leaf. While the overall design followed a general pattern of most ensemble learning models, for this project, the implementation was done with feature engineering, such as adding lagged variables and rolling averages that

emanate from time series. This ensured the model had been tailored to the specific forecasting challenge presented by the dataset. The project developed a model that could give accurate and reliable demand predictions by leveraging the flexibility provided by Random Forest.

Training Methodology

The Random Forest Regressor model was trained on a preprocessed dataset that consisted of aggregated weekly sales data with engineered features like time-based indicators and lagged variables for capturing the historic trend. The data was divided into training and testing sets, with the training set consisting of data from 2013 to 2016, while the test set was reserved for 2017. This will ensure that the model learns patterns in the past years while being evaluated on unseen, future data.

I didn't attempt any hyperparameter tuning in this project, since the focus was on building and testing a baseline model with respect to performance. The default parameters for the Random Forest Regressor were utilized, which often provide a decent starting position for many datasets. This allowed quicker implementation and an initial understanding of the model's capability without extra computational complexity.

Cross-validation was employed to ensure that the model's performance was consistent across subsets of the data, which inherently avoids underfitting and overfitting into the training set. In addition, out-of-sample testing with the 2017 dataset acted as a final check for the generalizability of the model. Careful tuning of the model was performed by incorporating these validation methods into the training process, hence striking a balance between capturing complex patterns in the data with predictively reliable unseen scenarios.

Results

The performance of the model is quite good for a baseline implementation as shown by the evaluation metrics. The MAE and RMSE values of 12.90 and 18.17, respectively, indicate that the model makes predictions of weekly sales reasonably well by capturing important trends and seasonal patterns. These values indicate that, in general, the model performed well but still has scope for improvement in terms of minimizing the prediction error of a particular store-product combination.

The model was tested by splitting the data into a train-test split: from 2013 to 2016 for training and from 2017 for testing. In this way, the model learned the historical patterns and was tested on unseen, future data. Besides, visual comparisons of predicted versus actual weekly sales were drawn to validate that the model follows the observed trends-qualitatively assessing the effectiveness of the model.

The model outputs will be the forecasted weekly sales for each product in every store within the chain. These forecasts are actionable and offer insight into future demand to drive inventory optimization and operational planning.

Data analysis played a vital role in the model. In fact, the exploratory data analysis uncovered critical patterns such as seasonality and lagged effects that drove feature engineering. Including lag variables and rolling averages significantly enhances the temporal dependency captured by the model, as shown by a reasonable match between the predicted values and the actual values. The data analysis thus not only enhanced the predictive power of the model but also ensured the fact that the predictions have a basis in the underlying structure of the dataset.

Future Work

The model performed well and captured the overall sales pattern, with fairly accurate forecasts. However, there is room for improvement, especially in refining the predictions and addressing potential areas of underperformance. For example, the results could be further improved by hyperparameter tuning, which would optimize the configuration of the Random Forest Regressor and potentially improve the accuracy of the forecasts. Moreover, the inclusion of more robust cross-validation techniques would help to ensure better generalization across different subsets of data.

The following steps could be done to further improve the project: first, by integrating external data on holidays, promotions, or weather conditions, the dataset would be enriched, and the model would have more chances to catch the exogenous factors of demand; second, further research could be done by using more advanced models such as Gradient Boosting or neural networks like LSTMs. On the data analysis side, more sophisticated feature engineering and deeper statistical analysis might reveal a number of patterns that could help improve predictive performance.

I would like to explore some more advanced machine learning techniques on future projects: ensemble models such as Random Forests combined with boosting methods, or neural networks tailored for time-series data. Besides, developing an interactive dashboard for real-time forecasting and visualization might bridge the gap between technical implementation and stakeholder usability. In this project, I have learnt the importance of thorough data analysis and how feature engineering can impact model performance-lessons that will guide future work in predictive modeling and demand forecasting.

Stakeholder Acknowledgements

The major stakeholders for this project are retailers, manufacturers, logistics companies, and supply chain managers. Retailers and manufacturers can benefit from better demand forecasting to optimize inventory levels, minimize stockouts, and reduce the costs associated with overstocking. Logistics companies can use the demand data to plan transportation and warehousing resources more efficiently, thereby reducing operational inefficiencies. For the supply chain manager, it serves as a decision-support tool to improve inventory control and optimize order fulfillment to match demand with supply.

However, it might have certain negative impacts in case of misuse or relying only on this model without taking into consideration the influence of exogenous shocks or singular market conditions. For example, if it heavily relies on historical data, there may be inaccurate predictions when such sudden changes occur as unforeseen surges in demand or disruptions in supply. Such instances could lead to under-preparedness, missed sales opportunities, or overstock of inventories if predictions don't consider non-historical elements.

At this stage, while there has not been any direct testing of the model by stakeholders, it is intuitive, easy to interpret, and provides actionable insights. With further development-such as a user interface or dashboard-participants could directly interact with the model to validate practical utility. Stakeholder feedback regarding the predictions and usability could further guide the improvements, so that the model is aligned with their operational needs and expectations. It would make the model more relevant and effective in applied settings.

Conclusion

This project successfully portrayed the application of machine learning in demand forecasting, covering two major pain points in any supply chain: inventory optimization and cost reduction. Feature engineering, combined with the time-series analysis and systematic testing of the Random Forest Regressor, made this model really powerful for capturing seasonal trends and variability across stores and products. Evaluation metrics, such as MAE and RMSE, were good for a baseline implementation, with actionable insights that could guide inventory management and operational planning.

Although the project met its initial goals, it also showed several avenues for further improvement: hyperparameter tuning, incorporating external data, and exploring advanced models such as Gradient Boosting and LSTMs. The lessons learned from exploratory data analysis were highly instrumental in enhancing the predictive accuracy of the models, hence justifying the importance of adequate data preparation and feature engineering.

This project really showcases the power of machine learning in demand forecasting and will provide much-needed tools for stakeholders within retail, manufacturing, and logistics. Further development should be devoted to developing more user-friendly dashboards and incorporating stakeholder feedback to further enhance the usability and effectiveness of the model in practical applications.

Citations

[1] Kaggle, "Demand forecasting - kernels only," [Online]. Available: https://www.kaggle.com/code/graisybiswal/demand-forecasting-in-supply-chain-analytics. [Accessed: Nov. 26, 2024].

[2] G. Biswal, "Demand forecasting in supply chain analytics," Kaggle, [Online]. Available: https://www.kaggle.com/code/graisybiswal/demand-forecasting-in-supply-chain-analytics/notebook. [Accessed: Nov. 26, 2024].