INVENTORY DEMAND FORECASTING

Radhika Pisipati - 1RVU22CSE126
SoCSE
RV University
Bengaluru, Karnataka

Renu Bojja - 1RVU22CSE129
SoCSE
RV University
Bengaluru, Karnataka

Karmishtha Patnaik - 1RVU22CSE077
SoCSE
RV University
Bengaluru, Karnataka

***Abstract*—In the modern evolving retail world, with demand drastically increasing and supply not being able to fulfill the demands, there is a need for a mechanism to predict demand in-order to accurately predict sales for the success of any business venture. In our project, we aim to use various machine learning regression algorithms, ranging from simple linear regression model to a complex neural network, to predict inventory demand. Our objective is to create a comparison metrics in-order to determine which model's approach works best to help accurately and efficiently predict inventory demand.**

I. Introduction

In today's retail world, understanding and predicting inventory is essential to ensure customer satisfaction, improving operational costs and maximising profit. This is where inventory demand forecasting becomes an extremely useful and essential tool as it enables businesses to maintain sufficient stock levels, minimize products going out of stock and prevents surplus stock for a product that has low demand. Traditionally, demand forecasting was done based on concepts such as probability and statistics and data analysis. However due to the uncertain nature of demand in today's world, and the advancements in technology have facilitated the use of machine learning and deep learning algorithms to better understand consumer behaviour and accurately predict demand, thus, helping in inventory forecasting. This paper focuses on the use of various regression based machine learning models to predict inventory demand. The objective of our research is to ultimately compile all the accuracy's of the models we have worked with into a comparison table to establish which machine learning model efficiently provides the best accuracy for this problem statement. In the following sections, we provide a comprehensive review of all previous related work in this domain. We then proceed to explain our methodology which details the theoretical framework, data collection methods and techniques used to analyse the data. Next comes our experimental details section where we provide detailed information about the experimental setup, dataset details, results and comparisons and all related information. We then conclude our paper by discussing the limitations of our project and future work and finally we end this paper with references that helped us during our project and research work and ultimately lead to the successful implementation of our project.

II. Related Work

Several studies have previously addressed the issue of inventory demand forecasting using various techniques. Wang et al. focused on the use of time series analysis and recurrent neural networks for demand forecasting in retail, while Smith and Jones utilized a combination of statistical methods and machine learning algorithms to predict inventory demand. These studies have provided valuable insights into the challenges and opportunities in inventory demand forecasting and have laid the foundation for our research. Other researchers have explored the use of data mining and customer segmentation to improve demand forecasting in retail stores (Guo et al., 2014).

III. Methodology

For this project, we have used 11 models to perform our evaluation and comparison metrics, 8 supervised learning models, 2 unsupervised learning models, 1 deep learning model. The models that have been implemented are listed below: Linear Regression, Polynomial Regression, KNN Regressor, Decision Tree Regressor, Random Forest Regressor, K-Means Clustering, Ridge Regression, Lasso Regression, XGB Regressor, Hierarchical Clustering, Neural Networks - Multi-layer Perceptron.

A. Theoretical Framework

Our project is built around the various applications of regression models to predict inventory demand. We have based our implementation in such a way that it handles the different aspects of the relationship between each of the dependent and independent variables in our dataset, i.e, linear relationships and non linear relationships.

B. Data Collection

The data used in this project was taken from taken from Kaggle's Walmart dataset. This dataset consists of historical data that covers sales from 05.02.2010 to 01-11-2012. It consists of the following information - store (the store number), date (the week of sales), weekly-sales (sales for the given store), Holiday flag (to indicate if the week has a special holiday or public holiday), temperature (temperature on the day of sales), fuel-price (cost of fuel in that region), CPI (prevailing customer price index), unemployment (prevailing unemployment rate), holiday-event (List of all national and public holidays)

C. Data Analysis

- Data preprocessing - This was done so as to perform preliminary investigations on data to discover all the patterns and outliers, to understand the data type of each column and to look for any/all null and unique values. Categorical values were also encoded and feature scaling and normalisation were also done to ensure consistency across the various features.
- Model Training - All the regression models that were implemented was trained on a subset of the data and the hyper-parameters were tuned using techniques like grid-search, randomised-search and cross-validation.
- Model Evaluation - The model was evaluated using metrics like: Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), R Squared (R2)
- Comparison and Analysis - The results that were obtained from each model were compared and analysed (ref: Results Section) and were also visualised using scatter-plot and distribution-plot graphs.

IV. Experimental Details

On the setting up of our experiments, we did a proper division of our dataset into training and testing sets which were split at 70:30. In order to determine how each model performed, we measured it using different performance metrics such as mean absolute error, mean squared error, R-squared, root mean squared error (RMSE). Furthermore, during the processing stage, data was pre-processed cautiously through procedures like normalization as well as feature scaling in order to help optimize performance of models. However, despite great progress about demand forecasting with regression models; there are still some notable gaps in our understanding. This makes it difficult for us to apply them broadly since many studies tend to focus more narrowly on specific products or industries. Apart from that, the computational requirements associated with certain advanced forms of regression are simply heavy making them unsuitable for accurate big retail real-time demand forecasts. And these complex neural networks when analyzed by businesses often need clear explanations regarding their predictions. So here is where our project comes in; it aims at covering a few crucial aspects that will fill these gaps. Essentially we are investigating how different regression models compare when predicting inventory demand in retailing intensively

A. Experimental Setup

We utilized the Panda, Numpy, Matplotlib, and Seaborn libraries in Python 3 to construct our project. Our code was executed on Google Colab. Eight supervised learning models, two unsupervised learning models, and one deep learning model were the eleven models we used to accomplish our assessment and comparison measures. The following is a list of the models that have been used: Linear Regression, Polynomial Regression, KNN Regressor, Decision Tree Regressor, Random Forest Regressor, K-Means Clustering, Ridge Regression, Lasso Regression, XGB Regressor, Hierarchical Clustering, Neural Networks - Multi-layer Perceptron — all relate to regression analysis. We have used Walmart Dataset that can be found on Kaggle's website.

B. Training Process

The training process involved a number of process, some of which are mentioned below:

- Data Cleaning: The implemented algorithm handles missing values, encodes categorical variables, scales numerical features etc as other ways for preprocessing data. This process is aimed at improving the performance of the model and preparing data for model training.
- Feature Engineering: year, quarter, month and week can be extracted from the date variable. The goal of feature engineering is to collect more information that may be related to a given target variable.
- Choosing Models: In this case machine learning models such as Linear Regression, Polynomial Regression, KNN Regressor, Decision Tree Regressor, Random Forest Regressor, K-Means Clustering, Ridge Regression, Lasso Regression, XGB Regressor, Hierarchical Clustering, Neural Networks - Multi-layer Perceptron have been utilized by the code. These models have been selected according to how well they perform on data and how well they could fit into the problem at hand.
- Model fine-tuning: In order to find the best hyperparameters for some of these models, the algorithm uses a combination of randomized and grid searches. They are pre-training variables that have a huge impact on how well the model functions. Randomized search takes hyper-parameters at random within a range, while in grid search we try every possible combination of hyper parameters within that range.
- Cross-Validation: To check if our model is overfitting our data, algorithms incorporate cross-validation techniques for example k-fold cross-validation technique. The data is divided into k folds for cross validation purposes only. Then choose one fold from this K fold which will be held out as test set while remaining k-1 folds used as training sample set. This process is repeated k times in order to get more accurate results with average obtained results being an estimate of performance of the model.
- Performance Evaluation: The code measures model accuracy using various evaluation metrics such as mean absolute error (MAE), mean square error (MSE), root mean square error (RMSE) and R^2 score. These metrics assess both correctness and generalizability of the model.

C. Results

The following is the conclusion that the team arrived at after implementing the project:

- Random Forest with Grid Search and Decision Tree with Random Search are the best unsupervised learning models. This is because, on both training and testing

data, the aforementioned models displayed low values for MAE, MSE, RMSE, and high R2 scores.
- The Hierarchical Clustering Supervised Learning Model Is the Best This is because K-means shows poorly divided and dense clusters due to its high inertia and low silhouette score. The modest cophenetic correlation and greater silhouette score of hierarchical clustering indicate better-defined groups.
- The Multi-Layer Perceptron deep learning model was the only one used. Low R2 values and large errors in the model suggest that it is not doing a good job of capturing the patterns in the data.
- When considering the model's best score and CV analysis, mean of scores and standard deviation of scores for the models namely: Polynomial Regression, Random Forest, and XGB Regressor models perform better than the others in terms of best scores and generalisation capabilities (perform well on unseen data also and prevent overfitting), making them the better options. Polynomial Regression outperforms the others, demonstrating higher performance on the dataset in question.

D. Figures and Tables

TABLE I
REGRESSION MODEL PERFORMANCE

Model	Metric	Training	Testing
Linear Regression	MAE	358406.76	362471.07
	MSE	204470961002.3	210920722575.97
	RMSE	452184.65	459261.06
	R2 Score	31.08%	31.86%
Polynomial Regression	MAE	58374.79	69004.3
	MSE	6886879714.26	9318364369.55
	RMSE	82987.23	96531.68
	R2 Score	97.68%	96.99%

TABLE II
REGRESSION MODELS (CONTINUED)

Model	Metric	Training	Testing
Linear Regression	MAE	358406.76	362471.07
	MSE	204470961002.3	210920722575.97
	RMSE	452184.65	459261.06
	R2 Score	31.08%	31.86%
Polynomial Regression	MAE	58374.79	69004.3
	MSE	6886879714.26	9318364369.55
	RMSE	82987.23	96531.68
	R2 Score	97.68%	96.99%
Ridge Regression (Grid Search)	MAE	358542.43	362730.83
	MSE	204480782446.65	210893777737.26
	RMSE	452195.51	459231.73
	R2 Score	31.07%	31.87%
Ridge Regression (Randomized Search)	MAE	358484.39	362622.14
	MSE	204474451017.8	210901978470.13
	RMSE	452188.51	459240.65
	R2 Score	31.08%	31.86%
Lasso Regression (Grid Search)	MAE	358407.68	362472.09
	MSE	204470966370.87	210919193605.71
	RMSE	452184.66	459259.4
	R2 Score	31.08%	31.86%
Lasso Regression (Randomized Search)	MAE	358407.67	362472.07
	MSE	204470966241.68	210919212014.84
	RMSE	452184.66	459259.42
	R2 Score	31.08%	31.86%
Decision Tree (Grid Search)	MAE	27113.33	59095.59
	MSE	2256760434.69	8772553461.59
	RMSE	47505.37	93661.91
	R2 Score	99.24%	97.17%
Decision Tree (Randomized Search)	MAE	27000.89	58554.83
	MSE	2244776329.23	8673488120.51
	RMSE	47379.07	93131.56
	R2 Score	99.24%	97.2%
Random Forest (Grid Search)	MAE	24055.94	64607.3
	MSE	1802225821.92	13365236840.7
	RMSE	42452.63	115608.12
	R2 Score	99.39%	95.68%
Random Forest (Randomized Search)	MAE	40446.31	65851.59
	MSE	5087114784.26	13913959520.21
	RMSE	71324.01	117957.45
	R2 Score	98.29%	95.5%
KNN (Grid Search)	MAE	246614.87	256808.54
	MSE	101506755882.72	110053791468.37
	RMSE	318601.25	331743.56
	R2 Score	65.78%	64.45%
KNN (Randomized Search)	MAE	346471.61	358868.73
	MSE	178306750577.2	192819516586.96
	RMSE	422263.84	439112.19
	R2 Score	39.9%	37.71%
XGB (Grid Search)	MAE	65424.52	75429.65
	MSE	9102237939.09	12495116665.71
	RMSE	95405.65	111781.56
	R2 Score	96.93%	95.96%
XGB (Randomized Search)	MAE	0.09	214910.35
	MSE	0.02	82037108313.36
	RMSE	0.12	286421.21
	R2 Score	100%	73.5%
	Accuracy	-	33.33%

TABLE III
CLUSTERING PERFORMANCE

Model	Metric
K-Means	Inertia: 12026.78
K-Means	Silhouette Score: 0.1167
Hierarchical Clustering	Silhouette Score: 0.1592
Hierarchical Clustering	Cophenetic Cor. Coef.: 0.5210658972678718

Model	**Training Performance**	**Testing Performance**	**Difference (Testing - Training)**
Neural Networks			
Mean Absolute Error (MAE)	436710.29	442863.72	+6153.43
Mean Squared Error (MSE)	282714664199.16	291941102291.95	+9228438092.7
Root Mean Squared Error (RMSE)	531709.19	540315.74	+8606.55
R Squared (R2) score	4.7%	5.68%	+0.98%
Mean Absolute Percentage Error (MAPE)	64.02%	64.02%	-
Accuracy	12.85%	12.76%	-0.09%

CONCLUSION

The project investigated various machine learning models reliant on regression to forecast inventory demand. It is possible because of the successful use of a wide range of Machine Learning Models based on Regression in forecasting for demand for inventory. This helped us to understand how these models perform and when they should be used. In terms of methodology, linear regression, polynomial regression, decision tree regressors, random forest regressors, and neural network were some of the broadest set of regression models that were employed. Project was able to divide data in Kaggle Walmart dataset into train-test sets before preprocessing it and fitting last select best-performing models among them. Mean absolute error, mean squared error root mean square error and R-squared were some performance metrics used in this study. Some performed well while others did not; therefore the results provided by each model are its strengths and weaknesses as seen from Table 2 above. For example Random Forests Decision Trees showed superior performance compared with other algorithms but neural networks had lower accuracy as compared to other models. Polynomial Regression stood out with regard to capturing complex relationships between variables in this context.

REFERENCES

[1] M. Owais, A. S. Ahmed, G. S. Moussa, and A. A. Khalil, "An optimal metro design for transit networks in existing square cities based on Non-Demand criterion," *Sustainability*, vol. 12, no. 22, p. 9566, Nov. 2020, doi: 10.3390/su12229566.

[2] K. N. Sukhia, A. A. Khan, and M. Bano, "Introducing Economic Order Quantity Model for Inventory Control in Web based Point of Sale Applications and Comparative Analysis of Techniques for Demand Forecasting in Inventory Management," *International Journal of Computer Applications*, vol. 107, no. 19, pp. 1–8, Dec. 2014, doi: 10.5120/18856-7385.

[3] J.-J. Chanaron, "Pricing innovation: state of the art and automotive applications," *ideas.repec.org*, 2008, [Online]. Available: https://ideas.repec.org/p/hal/gemptp/halshs-00371047.html.

[4] P. K. Bala, "Improving inventory performance with clustering based demand forecasts," *Journal of Modelling in Management*, vol. 7, no. 1, pp. 23–37, Mar. 2012, doi: 10.1108/17465661211208794.

[5] T. E. Goltsos, A. A. Syntetos, C. H. Glock, and G. Ioannou, "Inventory – forecasting: Mind the gap," *European Journal of Operational Research*, vol. 299, no. 2, pp. 397–419, Jun. 2022, doi: 10.1016/j.ejor.2021.07.040.

[6] M. Abolghasemi, E. J. Beh, G. Tarr, and R. Gerlach, "Demand forecasting in supply chain: The impact of demand volatility in the presence of promotion," *Computers & Industrial Engineering*, vol. 142, p. 106380, Apr. 2020, doi: 10.1016/j.cie.2020.106380.

[7] "Decision tree based demand forecasts for improving inventory performance," *IEEE Conference Publication — IEEE Xplore*, Dec. 01, 2010. https://ieeexplore.ieee.org/abstract/document/5674628.

[8] Y. Tadayonrad and A. B. Ndiaye, "A new key performance indicator model for demand forecasting in inventory management considering supply chain reliability and seasonality," *Supply Chain Analytics*, vol. 3, p. 100026, Sep. 2023, doi: 10.1016/j.sca.2023.100026.

[9] M. Z. Babaï, M. Ali, J. E. Boylan, and A. A. Syntetos, "Forecasting and inventory performance in a two-stage supply chain with ARIMA(0,1,1) demand: Theory and empirical analysis," *International Journal of Production Economics*, vol. 143, no. 2, pp. 463–471, Jun. 2013, doi: 10.1016/j.ijpe.2011.09.004.

[10] N. Kourentzes, J. R. Trapero, and D. K. Barrow, "Optimising forecasting models for inventory planning," *International Journal of Production Economics*, vol. 225, p. 107597, Jul. 2020, doi: 10.1016/j.ijpe.2019.107597.

[11] A.-L. Beutel and S. Minner, "Safety stock planning under causal demand forecasting," *International Journal of Production Economics*, vol. 140, no. 2, pp. 637–645, Dec. 2012, doi: 10.1016/j.ijpe.2011.04.017.

[12] R. Van Steenbergen and M. Mes, "Forecasting demand profiles of new products," *Decision Support Systems*, vol. 139, p. 113401, Dec. 2020, doi: 10.1016/j.dss.2020.113401.

[13] J. J. Bergman, J. S. Noble, R. G. McGarvey, and R. L. Bradley, "A Bayesian approach to demand forecasting for new equipment programs," *Robotics and Computer-integrated Manufacturing*, vol. 47, pp. 17–21, Oct. 2017, doi: 10.1016/j.rcim.2016.12.010.