Proceedings of the 2nd International Conference on Business and Policy Studies
DOI: 10.54254/2754-1169/17/20231094

Sales Prediction of Big Mart Based on Linear Regression, Random Forest, and Gradient Boosting

Ruiyun Kang[1,a,*]

[1] University of California, Berkeley, Berkeley CA 94720, United States
a. locke.k@berkeley.edu
**corresponding author*

Abstract: With rapid development of machine learning and data science approaches, many retailers are employing sales prediction to aid in strategy formulation and profit growth. However, it is challenging for some small merchants to access vast volumes of data for analysis. This paper investigates the viability of predict sales in a small-scale retail supermarket, and evaluates the performance of linear regression, random forest, and gradient boosting method in the corresponding limited dataset. The selected metrics for analysis are RMSE, MAE, and R^2 score. Experiment results indicate that the linear regression model has a relatively large error and suffers from underfitting. The two decision-tree based models, random forest, and gradient boosting, perform similarly, with gradient boosting model outperforms by a small margin. This study illustrates that it is feasible to perform sales forecasting by machine learning techniques on a small collection of data. It also identifies the issues in this process and suggests potential fixed, giving small retailers a guideline for making effective sales predictions.

Keywords: sales prediction, machine learning, linear regression, random forest, gradient boosting

1. Introduction

In the age of data technology, data acquisition and analysis are getting more and more important. As one of the core applications of data science, forecasting has become an indispensable tool in all industries. Decades ago, a survey of businessmen showed that 92% of the respondents regarded sales forecasting as being of great significance [1]. Additionally, among these merchants, the group of retailers expressed the most interest in and concern over this topic [1]. For the retail industry, accurate sales forecasting means capture of market trends in advance, better inventory, and pricing decisions, and maximized profits. In early days, small and medium-sized retailers might not have been able to make as good sales forecasts as large companies and banks due to a variety of constraints. However, decreased expenses and the development of numerical prediction methods have enabled even individual retailers to benefit from this amazing technique.

Various studies on sales forecasting through machine learning have been conducted, which yield many in-depth conclusions. Scholars examine the relationship between weather and food retailers' sales, and develop a deep-learning based model that uses LSTM and a stacked denoising autoencoder network [2]. Based on the model, they predict the sales of a Japanese chain supermarket and achieve an outcome superior than many conventional machine learning methods. Other researches carry out sales forecasting on Amazon's book sales using a variety of modeling techniques, such as regression, decision tree, and artificial neural network [3]. Features from multiple perspectives, including prices, sentiments and their interactions, are used in training and are shown to have various degrees of importance in different models. The result provides a guideline for online venders to precisely forecast their sales. In addition, other study discusses a concern that is frequently disregarded when predicting house selling prices through the random forest model [4]. By conducting analysis on the house sales data on Fairfax County, Virgina, they confirm the efficacy of variables related to the spatial correlation of houses, and therefore refine the random forest model. The research presented in Ref. [5] examines the significance of the ARIMA model and introduces a deep learning model built on top of it. The authors demonstrate the viability of this ARIMA-based model by analyzing five-year sales data from several stores and achieve a high accuracy. Other research introduces ForeXGBoost, which is a method based on XGBoost and takes the first place in a competition for machine learning models. ForeXGBoost model combines XGBoost, time series analysis, one-hot encoding techniques, and missing value filling [6]. Despite being specifically designed for the vehicle sales data used in the competition, the model also performs well in other sales forecasting scenarios, particularly when it comes to time series and data anomalies. Some of the scholars discuss a potential solution to the issue of some merchants not having enough data for sales forecasting [7]. They analyze short time-series data through neural network and utilize a multilayer-perceptron to do prediction, eventually demonstrating the adaptability of neural networks in sales prediction with small datasets.

The aim of this paper is to investigate the feasibility of machine learning for sales prediction in a small-scale retail supermarket, Big Mart, and to compare the performance of models used. The rest part of the paper is organized as follows. Section 2 presents the data source and its processing procedure, introduces the models and metrics, and outlines how the experiment was conducted. Section 3 introduces and analyzes the results. Section 4 examines potential limitations of this research and offers suggestions for improvement. Eventually, Section 5, summarizes the experiment and discusses its implication.

2. Data & Methods

2.1. Data

The dataset examined in this research was from the Kaggle stage and contains information about Big Mart, which was a supermarket brand with more than 60 stores currently operating in Nepal. It comprised sales data for 1559 products across 10 stores in different areas in 2013 and provided detailed information about the product, outlet information where the product was purchased, and its sales value. The provided train set that had 8523 observations was used to build and test the model. The dependent variable to be predicted was the sales of each product at an outlet with characteristics. There were 11 independent variables in total. Seven of them were categorical variables, including product ID, fat content, product type, outlet ID, outlet size, location type, and outlet type, while weight, visibility, maximum retail price (MRP), and year of establishment were the remaining 4 numerical variables. Some data had missing values in weight and outlet size, which needed to be fixed before model training.

2.2. Data Processing

Generally, four steps were taken during data processing: train-test split, handling missing values, consolidating redundant categories in each variable, and converting categorical variables into indicator variables. First, the training set provided was randomly divided into a new training set which comprised 80% of the 8523 data points, and a test set that contained the remaining 20%. Subseuqently, data cleaning and transformation were performed in these two sets, respectively. Missing values were found in two independent variables: weight and outlet size. For weight, the average of existing values was used to fill the gap. For outlet size, a more precise method was applied. The outlet size was found to have a relatively strong correlation (0.56) with outlet types, so the mode of outlet sizes in each outlet type was calculated first, and then the corresponding mode was filled in according to the outlet type of the missing data. Afterwards, redundant categories in the variable fat content were solved: duplicate labels such as "low fat", "LF", and "reg" were combined into "Low Fat" and "Regular" in respective. Lastly, all the categorical variables were converted into indicator variables by using the built-in function *get_dummies* in the Pandas library in Python.

2.3. Models and Metrics

Three determining models were chosen to fit the data: Linear Regression, Random Forest, and Gradient Boosting. Other methods such as Stochastic Gradient Descent were also scrutinized, but the results, probably due to the small training set size, were impractical, so they were not considered here. All models came from scikit-learn library in Python and were executed in Python 3.9. The metrics used to determine the performance of models were Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and the coefficient of determination (R^2).

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{1}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|x_i - \bar{x}| \tag{2}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y_i})^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{3}$$

Linear regression model with ordinary least squares (OLS) method was used. This model assumes that the relationship between the dependent variable y and the independent variable x is linear, and generates the set of coefficients β_i's which minimizes the residual sum of squares between the observed values in dataset and the targets predicted by the model. Here, the intercept term was included to achieve better result. The formular is as following:

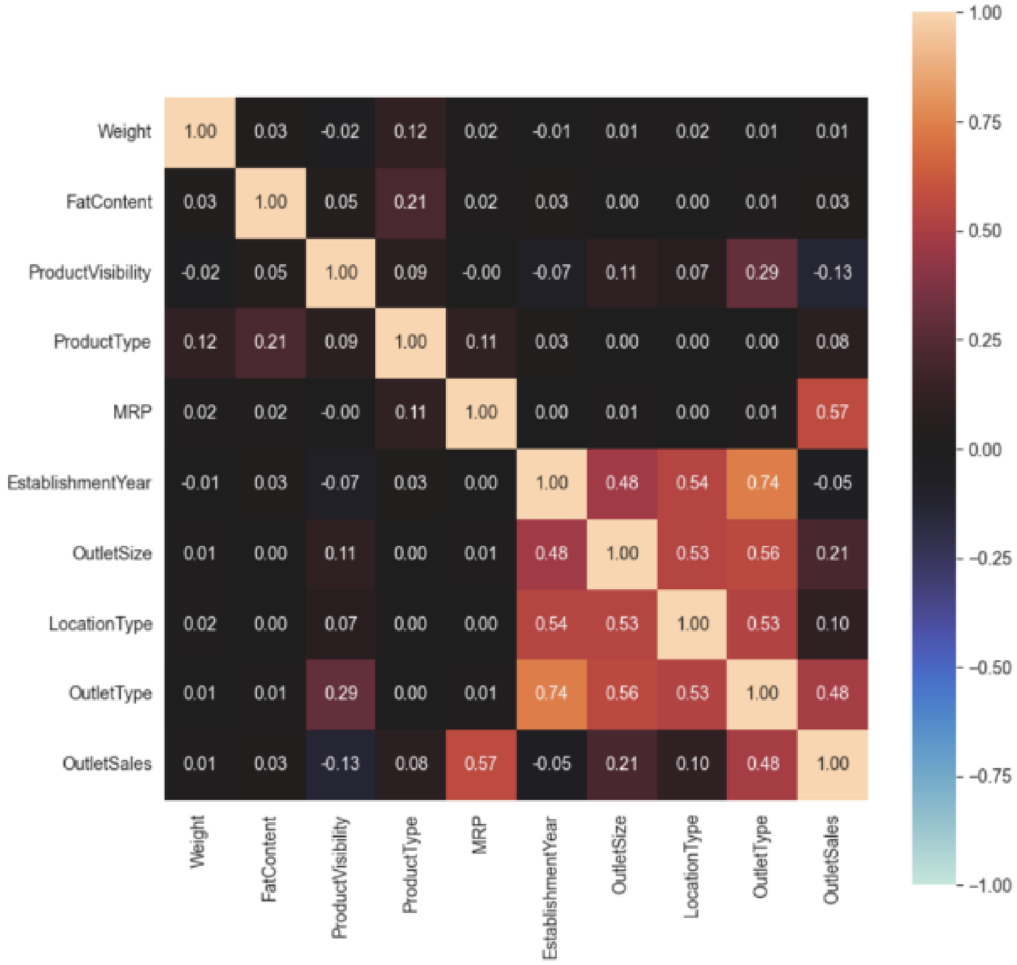
$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \cdots \tag{4}$$

Random Forest is an ensemble learning method that operates by building a number of decision trees at training time and uses averaging to improve prediction accuracy and reduce overfitting. For regression tasks, the average predicted values of individual trees are returned. In this experiment, n_estimator, the parameter that determined the number of trees in the forest, was set to 200 and the function that measured the quality of splits was selected to be mean squared error.

Gradient Boosting is a method of combining several simple models, which are typically decision trees, into a composite model. It has a forward stage-wise fashion since simple models are added one by one and each new model takes a step in the direction minimizing the prediction error. As more simple models are combined, the final completed model gets stronger. This algorithm uses gradient descent to minimize losses, which is where the term "gradient" comes from. When training the model, learning rate and the number of boosting stages were set to 0.1 and 100, respectively.

2.4. Procedure

After data processing and model selection, the next steps were to choose appropriate features and train the models. Features were selected based on their correlation with the target variable. In the calculation of correlation coefficients, Cramér's V (also known as Cramér's φ), the statistic that measured the association between nominal variables [8], was used as the criterion of measurement. In addition, the dython library in Python was applied for the calculation. Furthermore, different hyperparameters were tried during the learning process, and the final selection was shown in the previous section. When the training was completed, their results were compared and analyzed.



	Weight	FatContent	ProductVisibility	ProductType	MRP	EstablishmentYear	OutletSize	LocationType	OutletType	OutletSales
Weight	1.00	0.03	-0.02	0.12	0.02	-0.01	0.01	0.02	0.01	0.01
FatContent	0.03	1.00	0.05	0.21	0.02	0.03	0.00	0.00	0.01	0.03
ProductVisibility	-0.02	0.05	1.00	0.09	-0.00	-0.07	0.11	0.07	0.29	-0.13
ProductType	0.12	0.21	0.09	1.00	0.11	0.03	0.00	0.00	0.00	0.08
MRP	0.02	0.02	-0.00	0.11	1.00	0.00	0.01	0.00	0.01	0.57
EstablishmentYear	-0.01	0.03	-0.07	0.03	0.00	1.00	0.48	0.54	0.74	-0.05
OutletSize	0.01	0.00	0.11	0.00	0.01	0.48	1.00	0.53	0.56	0.21
LocationType	0.02	0.00	0.07	0.00	0.00	0.54	0.53	1.00	0.53	0.10
OutletType	0.01	0.01	0.29	0.00	0.01	0.74	0.56	0.53	1.00	0.48
OutletSales	0.01	0.03	-0.13	0.08	0.57	-0.05	0.21	0.10	0.48	1.00

Figure 1: Heatmap of correlation coefficients.

3. Results

3.1. Correlation

The heatmap that displayed the correlation between each variable was generated by the function *nominal.association()* from python library. It should be noted that the product ID and outlet ID, the two features used for identification, were not considered because they were not relevant to the objective of this study. The correlation was presented in Fig. 1. Except for MRP and outlet type, most of the features had a low correlation with the target variable, outlet sales. The relatively high correlation (0.21) between outlet size and outlet sales was, to some extent, due to their high correlation with outlet type. Based on the obtained correlation coefficients, the three least correlated features - weight (0.01), fat content (0.03), and year of establishment (-0.05) - were tested during model training to decide whether to be applied. Test results showed that both fat content and the year of establishment had a positive effect on the accuracy of the model. Thus, excluding weight, the remaining 8 features were used in model training.

3.2. Model Performance

For each model, scatter plots with observed values on x-axis and predicted values on y-axis were used to illustrate the performance. Ideally, the model should have a high R^2 score, which meant that the arrangement of the scatter points should be close to a straight line with a slope of 1.

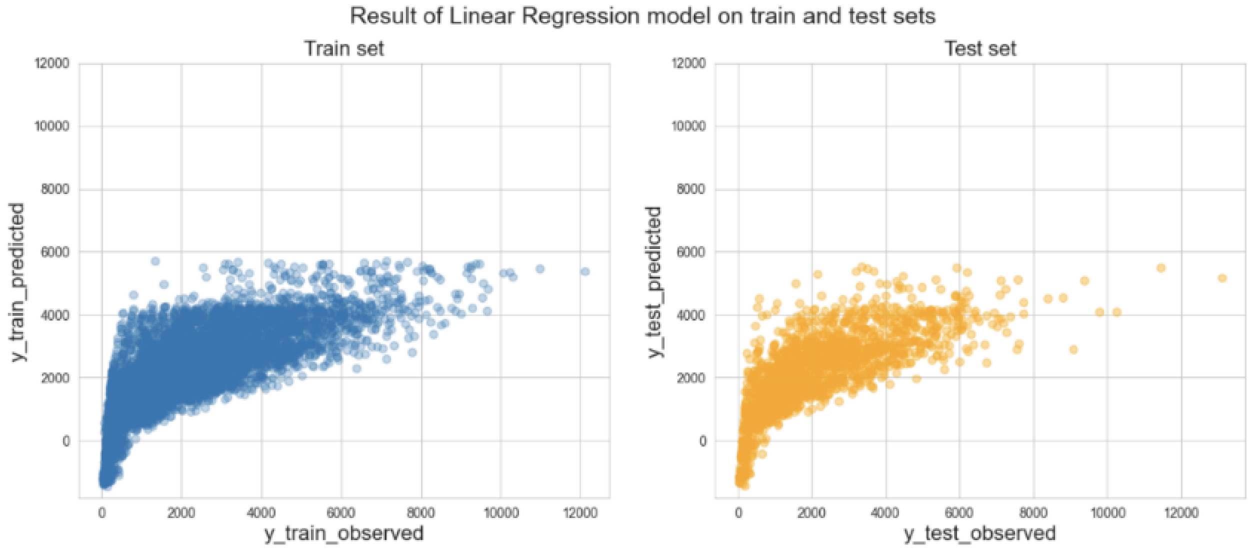


Figure 2: Scatter plots for results of linear regression.

The results of the linear regression model (seen in Fig. 2) did not appear to be linear. Instead, it was more like a curve with a positive but decreasing slope, which gave an upper limit that was around 6000 to predicted values. In addition, for part of the data with observed values less than 1000, the model's predictions were less than 0, which was inaccurate because the value of sales would never be negative. The reason for the unsatisfactory results might be related to the nature of the features used. Most of the features involved in training were categorical variables, which were not well suited for linear regression in the case of low correlation.

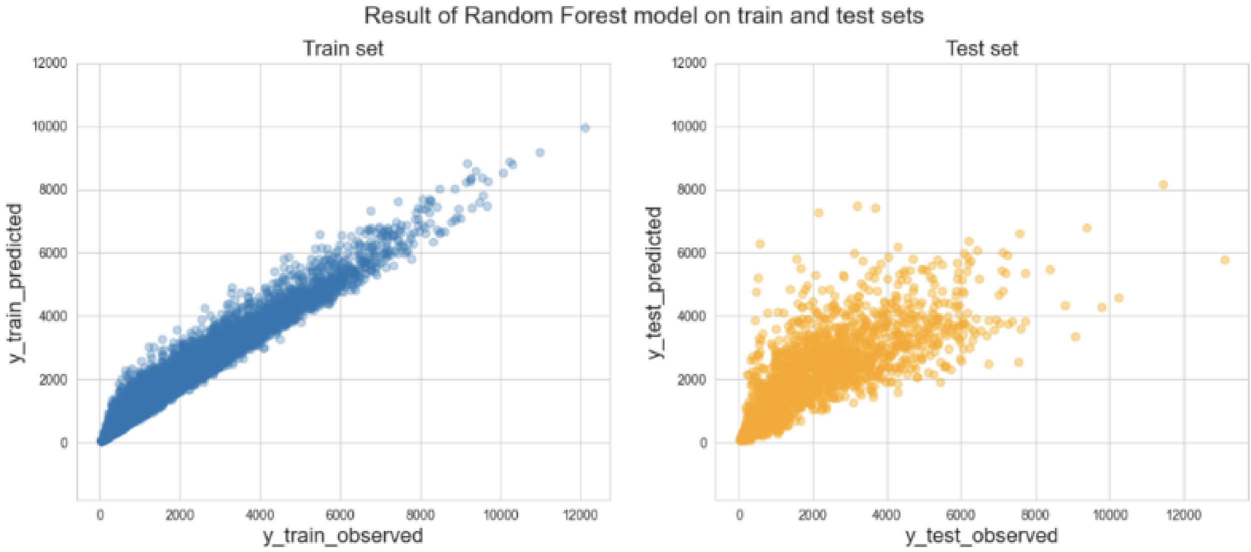


Figure 3: Scatter plots for results of random forest.

As presented in Fig. 3, the result of Random Forest was more reasonable than that of the linear regression model. The model performed very well in the training set due to the characteristics of decision trees and was therefore not informative, while the result for test set was not as perfect, but a linear trend still existed. From the density of data points, it could be seen that the random forest model performed better for data with observed sales between 0 and 6000, and it had a large error for data that had sale values greater than 6000. A possible reason for this phenomenon was that most of the samples used for training were concentrated between 0 and 6000, which made the model unable to effectively classify data beyond this interval.

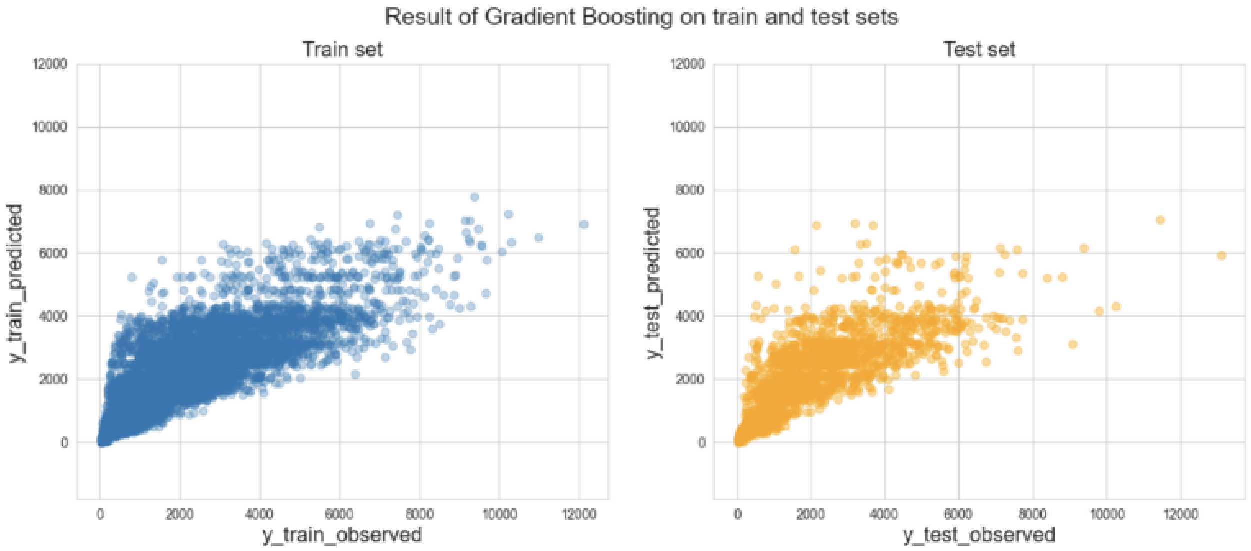


Figure 4: Scatter plots for results of gradient boosting.

The results of Gradient Boosting were illustrated in Fig. 4. Also based on decision trees, Gradient Boosting performed worse than Random Forest in fitting the training set due to its additive nature. However, its performance in the test set was slightly better, i.e., the linear trend of data points was more obvious. The predicted values obtained from this model were lower than the true values. Besides, the same issue in the prediction for data with true values larger than 6000 existed as well. The errors and R2 scores of different models were generally consistent with the trends exhibited by the scatter plots, as presented in Table. 1.

Table 1: Errors and R^2 scores of models.

	Linear Regression	Random Forest	Gradient Boosting
RMSE_train	1124.926	425.269	1029.046
MAE_train	834.714	293.638	727.225
R2_train	0.562	0.937	0.633
RMSE_test	1142.004	1133.550	1086.291
MAE_test	840.730	790.548	753.112
R2_test	0.567	0.574	0.609

Overall, Gradient Boosting performed the best among the three models. It had the highest R2 score and lowest errors in testing. Random Forests performed slightly worse than Gradient Boosting, which was consistent with the fact that gradient-boosted trees generally outperform random forests [9]. The linear regression model had relatively the worst accuracy, which could be determined by both the RMSE and MAE. Its RMSE was close to that of the other two models because there were more outliers in the results of them, which was because of the lack of accuracy in their prediction of data with large values.

3.3. Interpretation

The results demonstrated that the sales prediction by machine learning techniques was feasible but ineffective for the chosen small retail markets. In machine learning, though it was not always true that the more training data used, the better the resulting model would be. But it was obvious that the data collected from a small retail market, such as Big Mart, was insufficient to train a model that performs well. This was due to the characteristics of small markets, i.e., low retail traffic, a narrow range of available goods, and a limited number of data categories. This experiment suggested that for small volume sales data, doing sales prediction only by machine learning was not a good approach. Such sales forecasting was not particularly informative and could only be considered as an aid for small retailers to make decisions.

4. Limitation & Future Enhancement

While this experiment met the study’s aim, it was not perfect and the outcomes still had some flaws. In general, the lack of training data put all three models at risk of underfitting. In particular, the underfitting of linear regression model was more severe, as shown in Fig. 2, possibly because of the data’s predominance in categorical variables that had low correlation. In contrast, the overfitting was also reflected in the random forest model to some extent. Additionally, the limitations in terms of the dataset were not only reflected in the insufficient amount. The lack of different aspects of dependent variables was also a major factor. As Armstrong mentioned in the research, factors such as the environment, customers, market conditions, and competitors could all cause changes in sales [10]. The data utilized in this experiment were all regarding the products and the stores and no external factors were included, which constrained the accuracy of the models. In future studies, when dealing with small-volume and less informative datasets, a smart strategy is to make judgmental extrapolations, as suggested by Armstrong [10].

This experiment also had space for improvements in terms of model selection and evaluation criteria. Models based on decision tree, like random forest and gradient boosting, did not work well for predicting new data beyond the existing data range. While the commonly used method for prediction, linear regression model, showed rather substantial underfitting. Therefore, experimenting with other hybrid models, such as linear model trees, might produce better results. Moreover, if the data is viable, incorporating time series analysis or using more complex models like LSTM were potential ways for improvement in future researches.

5. Conclusion

In summary, this paper investigated the feasibility of using machine learning approaches to forecast sales of each product in Big Mart. Based on correlation coefficients, appropriate features were selected and used to train three models – linear regression, random forests, and gradient boosting. Among these models, gradient boosting outperformed the other two models in terms of accuracy and R^2 score. According to the result, sales forecasting was moderately predictive and could serve as an aid for decision making. However, this experiment several data and model restrictions, and thus the most ideal outcomes were not obtained. For future studies, alternative hybrid models, such as the linear model tree, or time series models (e.g., the LSTM) would be good attempts for better sales forecasting models. Moreover, for small-volume and uninformative datasets, judgmental extrapolations based on knowledge about the products and the market would be a good strategy to increase the accuracy of sales forecasting. Overall, these results offered a guideline for small retailers to make sales prediction through machine learning with restricted datasets, and pointed to potential issues within the process.

References

[1] Dalrymple, D. J.: Sales forecasting practices: Results from a United States survey. International journal of Forecasting, 3(3-4), 379-391 (1987).

[2] Liu, X., Ichise, R.: Food sales prediction with meteorological data—a case study of a japanese chain supermarket. In International Conference on Data Mining and Big Data (pp. 93-104). Springer, Cham (2017, July).

[3] Sharma, S. K., Chakraborti, S., Jha, T.: Analysis of book sales prediction at Amazon marketplace in India: a machine learning approach. Information Systems and e-Business Management, 17(2), 261-284 (2019).

[4] Hu, C. Y., Griffith, D. A.: Incorporating spatial autocorrelation into house sale price prediction using random forest model. Transactions in GIS, 26(5), 2123–2144 (2022).

[5] Manikandan, S., Deetshiha, A., Sushmitha, D. J., et al. Intelligent sales prediction using ARIMA techniques. AIP Conference Proceedings, 2444(1) (2022).

[6] Xia, X., Wu, S., Sun, L., et al. ForeXGBoost: passenger car sales prediction based on XGBoost. Distributed and Parallel Databases: An International Journal, 38(3), 713–738 (2020).

[7] Canton, C. R., Gibaja, D. E., Caballero, S. O.: Sales Prediction through Neural Networks for a Small Dataset. International Journal of Interactive Multimedia and Artificial Intelligence, 5(4), 35–41 (2019).

[8] Cramér, H.: Mathematical Methods of Statistics (PMS-9), Princeton: Princeton University Press, 2016.

[9] Caruana, R., Alexandru, N. M.: An Empirical Comparison of Supervised Learning Algorithms. Proceedings of the 23rd International Conference on Machine Learning – ICML, 06, (2006).

[10] Armstrong, J. S.: Sales Forecasting. The IEBM Encyclopedia of Marketing, 278-290, 1999. Retrieved from https://repository.upenn.edu/marketing_papers/237