Journal of Statistics Applications & Probability
*An International Journal*

http://dx.doi.org/10.18576/jsap/120101

# Restaurant Revenue Prediction Applying Supervised Learning Methods

*Md Yasin Ali Parh*[1,2,*], *Mst Sharmin Akter Sumy* [1,2], *and Most Sifat Muntaha Soni* [2]

[1]Department of Bioinformatics and Biostatistics, University of Louisville, US
[2]Department of Statistics, Islamic University, Kushtia, Bangladesh

Received: 2 Feb. 2022, Revised: 22 Mar. 2022, Accepted: 27 May 2022
Published online: 1 Jan. 2023

**Abstract:** In the competitive world, it is difficult to make a decision where to open a restaurant outlet that produces maximum revenue. Especially, it is difficult to accurately extrapolate across geographies and culture based on the personal judgement and experiences. Supervised learning approach may play a vital role to determine the feasibility of a new outlet with the prediction of revenue. The goal of this study was to predict restaurant revenue of 100,000 regional tab food investment (TFI) restaurant locations across Turkey. Several supervised learning techniques were used to select the optimal model for prediction. The LASSO method was selected as the best supervised method for the prediction of revenue as determined by lowest test error. Other models were employed, but LASSO outperformed all other models and had the added benefit of simplicity and interpretability. The LASSO model was used to predict the revenue of 100,000 new restaurant site locations based on the coefficients termed using the training data.

**Keywords:** Revenue, Prediction, Supervised learning method, LASSO

## 1 Introduction

Opening new restaurants require large amounts of time, capital and difficult to make a decision where to open a restaurant outlet that produces maximum revenue. Based on the personal judgement and experiences, it is challenging to extrapolate across geographies and culture [1]. If the wrong location for a restaurant brand is chosen, the site closes soon, and operating losses are incurred [2]. It is also needed to objectify TFI's process of deciding where to open new sites. TFI is a leading quick-service restaurant (QSR) operator in Turkey and China and make significant daily investments in developing new restaurant sites. The company is behind well-known brands like Burger King, Popeyes, and Arby's [3]. A number of studies were performed to predict the annual restaurant revenue applying machine learning methods [4,5,6].

In this study, we want to compare the predictive power of several supervised learning techniques and predict restaurant revenue of 100,000 regional TFI restaurant locations across Turkey. We applied supervised learning techniques to select the optimal model for revenue prediction as supervised learning approach play a vital role to determine the feasibility of a new outlet with the prediction of revenue [6]. We applied supervised learning techniques to the dataset provided by TFI on Kaggle forum [7]. This study involved three main objectives: 1) handled and processed the data to apply the supervised learning methods, 2) trained and evaluate the model to compare the predictive power of supervised learning methods, and 3) predict the restaurant revenues based on the best model from the training dataset.

## 2 Data and Variables

The data provided by TFI and hosted on Kaggle forum. The data was split into a train and test dataset. The training dataset had 137 observations with 41 explanatory variables and revenue as the response variable. The explanatory variables are opening date, city, type of cities with two categories (big and other), type of the restaurant with three categories (Food Court, Inline and Drive Thru) and 37 obfuscated variables ($P1 - P37$). These obfuscated variables are three categories,

* Corresponding author e-mail: m0parh01@louisville.edu

demographic, real estate and commercial. Demographic data are gathered from third party providers with GIS systems. These include population in any given area, age and gender distribution, development scales. Real estate data mainly relate to the location, front facade of the location, car park availability. Commercial data mainly include the existence of points of interest including schools, banks, other QSR operators[7]. In the dataset, revenue is a transformed value and does not equate to the true dollar amount. The test dataset had 100,000 observations with the same 41 explanatory variables. The testing dataset contained no response, so it could not be used to evaluate the models. Therefore, the training dataset had to be broken up into two parts for model building and evaluation. Once the evaluation was done, the best model parameters and the entire training dataset were used to predict the response in the test dataset. One of the challenges in the dataset was that the categorical feature describing the city of the restaurant site had 40+ levels. We used both principal component analysis and clustering to reduce the levels. Since we did not know the possible number of clusters, we only considered hierarchical clustering. Both methods were unable to group the cities. Finally, we grouped the city variables into three categories using another dataset, GDP of the cities. The GDP data were provided by the Turkish Statistical Institute [8,9]. The categories were defined as 1) GDP greater than or equal \$100,000 treated as 'high GDP city', 2) GDP between USD 100,000 and USD 30,000 treated as 'mid GDP city' and, 3) GDP less or equal USD 30,000 treated as 'low GDP city'.

## 3 Methods to predict the revenue

Several machine learning methods are available in the literature to predict the restaurant revenue [4,5,6]. Kowsari et al. analyzed this dataset applying random forest and support vector machine (SVM) algorithms [1]. In this paper, we applied a bunch of standard supervised, both regression and tree-based, methods including random forest to predict the revenue and compared their predictive power in terms mean squared error. We start with the linear regression model.

**Linear Regression:** The linear regression model was first used as a quick investigation of the large dataset. The data have over 40 predictors, and linear regression is often useful to fit the model and predict the future observations. After defining the data, we started our analysis using linear regression model:

$$Y = \beta_0 + \beta_1 X_1 + ... + \beta_p X_p + \varepsilon \tag{1}$$

Where, $\varepsilon \sim N(0, \sigma^2)$. Since we had more than 40 predictors, the linear regression model could not identify the best subset of the predictors. Finally, we considered subset selection methods to identify the best predictors. This approach involves identifying a subset of the $p$ predictors that we believe to be related to the response. Here, we are discussing three subset selection methods.

**Best Subset Selection: Algorithm**

1.Let $M_0$ denote the null model, which contains no predictors. This model simply predicts the sample mean for each observation.
2.For $k = 1, 2, ...p$ :
  –Fit all $\binom{p}{k}$ models that contain exactly $k$ predictors.
  –Pick the best among these $\binom{p}{k}$ models and call it $M_k$. Here, 'best' is defined as having the smallest Residual Sum of Squares (RSS), or equivalently, the largest $R^2$.
3.Select a single best model from $M_0$, $M_1$, ... $M_k$ using $C_p$, Akaike information criterion$(AIC)$, Bayesian information criterion $(BIC)$, or adjusted $R^2$.

While best subset selection is a simple and conceptually appealing approach, it suffers from computational limitations. The number of possible models that must be considered grows rapidly as $p$ increases. In general, there are $2^p$ models that involve subsets of p predictors [10]. For our data $p = 41$, there are approximately $2.199512 \times 10^{12}$ possible models to be considered. Consequently, the best subset selection method becomes computationally infeasible for our data.

**Forward Stepwise Selection:** Forward stepwise selection is a computationally efficient alternative to best subset selection.

Algorithm

1.Let $M_0$ denote the null model, which contains no predictors.
2.For $k = 0, ..., p-1$ :
  –Consider all $p-k$ models that augment the predictors in $M_k$ with one additional predictor.
  –Choose the best among these $p-k$ models and call it $M_{(k+1)}$. Here, best is defined as having the smallest $RSS$ or largest $R^2$ value.
3.Select a single best model from among $M_0$, ..., $M_p$ using $C_p$, $AIC$, $BIC$, or adjusted $R^2$

Forward stepwise selection involves fitting altogether $1+p(p+1)/2$ models [10]. For our data, it requires fitting only $1+(41\times 42)/2=862$ models.

**Backward Stepwise Selection:** Like forward stepwise selection, the backward stepwise selection provides an efficient alternative to best subset selection. However, unlike the forward stepwise selection, it begins with the full least squares model containing all $p$ predictors, and then iteratively removes the least useful predictor, one-at-a-time.

Algorithm:

1.Let $M_p$ denote the full model, which contains all p predictors.
2.For $k=p,p-1,...,1$ :
–Consider all $k$ models that contain all but one of the predictors in $M_k$, for a total of $k-1$ predictors.
–Choose the best among these $k$ models and call it $M_{(k-1)}$. Here best is defined as having the smallest $RSS$ or highest $R^2$.
3.Select a single best model among $M_0$, ... ,$M_p$ using $C_p$, $AIC$, $BIC$, or adjusted $R^2$

Like forward stepwise selection, the backward selection approach searches through only $1+p(p+1)/2$ models [10].

**Choosing the optimal model**

To select the best model concerning test error, we estimated the test error. There are two common approaches:

1.We can indirectly estimate test error by adjusting the training error to account for the bias due to overfitting.
2.We can directly estimate the test error using either a validation set approach or a cross-validation approach.

In our project, we considered both approaches. For the first approach, there are several techniques for adjusting the training error for the model. We considered four such approaches: $C_p$, $AIC$, $BIC$, and adjusted $R^2$ for the best model. For the second approach, we computed the validation set error or the cross-validation error for each model under consideration and then selected the model with the lowest estimated test error. This procedure has an advantage relative to $AIC$, $BIC$, $C_p$, and adjusted $R^2$, in that it provides a direct estimate of the test error and makes fewer assumptions about the true underlying model [10].

As an alternative to least square fitting, we can fit a model containing all $p$ predictors using a technique that constrains or regularizes the coefficient estimates, or equivalently, that shrinks the coefficient estimates towards zero. The two best-known techniques for shrinking the regression coefficients towards zero are ridge regression and the LASSO. These methods are applicable even when $p>n$.

**Ridge Regression**

The ridge regression coefficient estimates $\hat{\beta}_R$ are the values that minimize

$$\sum_{i=1}^{n}\left(y_i-\beta_0-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2+\lambda\sum_{j=1}^{p}\beta_j^2=RSS+\lambda\sum_{j=1}^{p}\beta_j^2 \tag{2}$$

where $\lambda\geq 0$ is a tuning parameter. Ridge regression seeks coefficient estimates that fit the data well by making the $RSS$ small. The second term, $\lambda\sum_{j=1}^{p}\beta_j^2$ is the shrinkage penalty, which is small when $\beta_1$ ,..., $\beta_p$ are close to zero. The tuning parameter $\lambda$ serves to control the relative impact of these two terms on the regression coefficient estimates Ridge regression produces a different set of coefficient estimates, $\hat{\beta}_R$, for each value of $\lambda$[11]. Implementing ridge regression requires a method for selecting a value for the tuning parameter. Cross-validation provides a simple way to tackle this problem. We choose a grid of $\lambda$ values and compute the cross-validation error for each value of $\lambda$, we then select the tuning parameter value for which the cross-validation error is smallest. We choose the best lambda from our training dataset and used this best lambda for the prediction of revenue in our test data.

**LASSO**

The obvious disadvantage of ridge regression is that it includes all p predictors in the final model. The penalty $\lambda\sum_{j=1}^{p}\beta_j^2$ will shrink all the coefficients towards zero, but it will not set any of them exactly to zero. Model interpretation is difficult when $p$ is large. The lasso alternative to ridge regression overcomes this disadvantage. The LASSO coefficients, $\hat{\beta}_L$, minimize the quantity

$$\sum_{i=1}^{n}\left(y_i-\beta_0-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2+\lambda\sum_{j=1}^{p}|\beta_j|=RSS+\lambda\sum_{j=1}^{p}|\beta_j| \tag{3}$$

In the case of the lasso, the $l_1$ penalty has the effect of forcing some of the coefficient estimates to be exactly equal to zero when the tuning parameter $\lambda$ is sufficiently large. The process to find the best lambda is the same as ridge regression [11].

**Principle Component Regression**

The principal components regression (PCR) approach involves constructing the first $M$ principal components, $Z_1, ..., Z_M$, and then using these components as the predictors in a linear regression model that is fitted by least squares. The key idea is that a small number of principal components suffice to explain most of the variability in the data, as well as the relationship with the response.

**Tree-based Methods**

For prediction, we also considered tree-based methods. They are often the simplest models and useful for interpretation of the data. Tree-based methods involve stratifying or segmenting the predictor space into several simple regions. For a regression tree, to predict a given observation, we typically use the mean of the training observations in the region to which it belongs. However, a tree without encountering any restriction may grow very complex which may produce good predictions on the training set but is likely to overfit the data, leading to poor test set performance. In this case, pruning the tree may lead to lower variance and better interpretation. Through the cross-validation approach, we can easily find the best tree that will minimize the test error rate. Unfortunately, trees generally do not have the same level of predictive accuracy as some of the other regression approaches mentioned above. However, by aggregating many decision trees, using methods like bagging, random forests, and boosting, the predictive performance of trees can be substantially improved.

Bagging (Bootstrap aggregating) reduces the variance and increases the prediction accuracy by taking many training sets from the population, then building a separate prediction model using each training set and averaging the resulting predictions.

On the other hand, the random forest algorithm does not consider all the predictors at each split in the tree. The logic behind it is that most or all the trees will use the strong predictor, along with some other moderately strong predictors to split the tree. Consequently, all the bagged trees will look quite similar to each other. Hence the predictions from the bagged trees will be highly correlated. This approach decorrelates the trees and improves the test error rate.

The last approach used in our study, boosting, learns slowly. Each tree grows using information from a previously formed tree. At step m, the model increases the weights for the observations that were misclassified and decreases the weights for the observations that were classified correctly [10].

## 4 Results

**Exploratory analysis**

For the exploratory analysis of the data, box plots were used to describe the range of revenue values and to compare revenue between categorical variables like the type of city (big city or other) and the restaurant type (drive-thru, inline, or food court). The boxplot in Figure 1 representing the range of the response variable, revenue, shows that there are some outliers in our response variable that drive up the mean revenue.

The boxplot of city group shows that revenue is driven higher in big cities as compared to smaller cities (Figure 2A) and the boxplot of revenue per restaurant type shows that there was only one observation for drive-thru restaurants and that there was little no difference in the mean revenue between food court and inline restaurants (Figure 2B).

Because there was only one observation of drive-thru restaurants in our training dataset, we merged it with the type 'inline' restaurants so that there were only two categories of restaurant type. The variables were re-coded and given dummy variables for further analysis.

To manage the category 'cities', which had over 40 levels, we attempted to regroup the levels using unsupervised learning methods, including principal component analysis (PCA) and hierarchical clustering. Because we did not know the number of categories, we did not consider $k-means$ clustering and only employed hierarchical clustering methods. Using $PCA$ and all explanatory variables, no clear pattern emerged in our data (Figure 3A). We used a sub-sample of our dataset to see if a pattern emerged, but again, there was no clear grouping of cities (Figure 3B). Hierarchical clustering was also employed and still, no clear patterns emerged to define groupings of cities (Figure 4).

As the PCA and clustering methods did not define any reasonable groupings for our city variable, the cities were reclassified by their gross domestic product (GDP). Cities with GDP of 100,000 USD or greater were classified as 'high-GDP', cities with GDP between 30,000 and 100,000 USD were classified as 'mid-GDP', and cities with a GDP of fewer than 30,000 USD were classified as 'low-GDP'. This new category was used throughout the remaining analyses.

**Results from Regression models**

Several regression methods were utilized for modeling the revenue. The first regression method was a linear regression model which found that only two obfuscate variables, $P8$ $(p < 0.05)$ and $P17$ $(p < 0.1)$ were significant in explaining revenue. The $R^2$ of the regression model was 39% with a residual squared error of $7.2 \times 10^{12}$. The test error in the predictions based on linear regression was $8.01 \times 10^{12}$. For forward stepwise selection, we considered $C_p$, $AIC$, $BIC$, and adjusted $R^2$ as measurements to determine the optimal model. Figure 5 shows that BIC selected one variable with mean squared error (MSE) $= 3.5 \times 10^{12}$, $C_p$ selected 4 variables with $MSE = 5.9 \times 10^{12}$, and adjusted $R^2$ selected 18 variables with $MSE = 7.4 \times 10^{12}$ for the prediction of revenue.

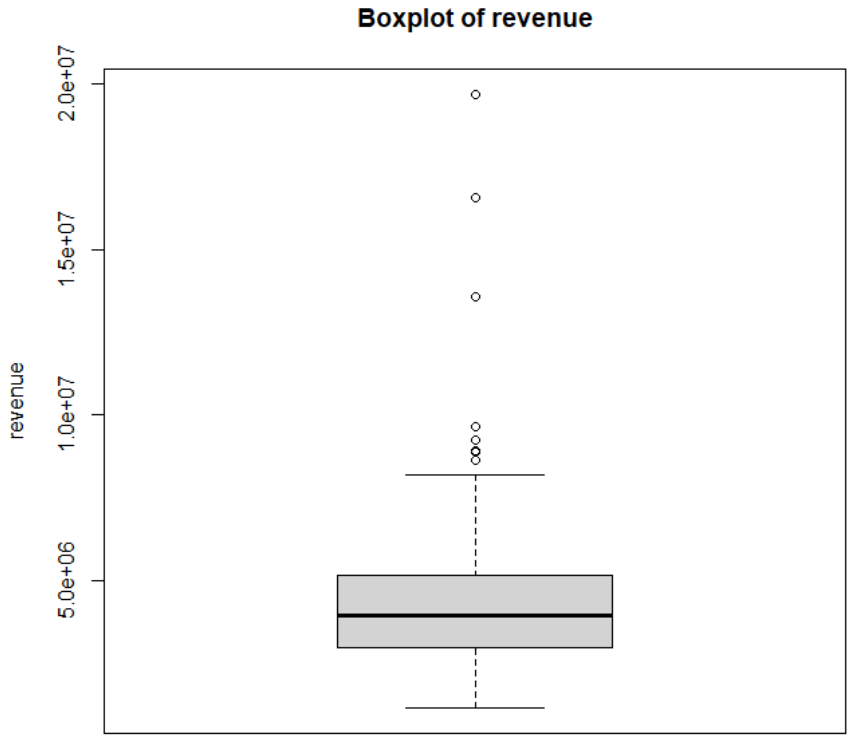


**Fig. 1:** Boxplot showing the range of values for the response variable revenue.

Backward stepwise model selection, Figure 6, found optimization with two variables under BIC ($MSE = 3.1 \times 10^{12}$), seven variables under $C_p$ ($MSE = 6.4 \times 10^{12}$), and 16 variables under adjusted $R^2$ ($MSE = 6.1 \times 10^{12}$). The two-variable model using backward stepwise selection over BIC out-performed the optimization under $C_p$ and adjusted $R^2$.

The forward and backward stepwise selection methods were also evaluated using validation and cross-validation methods. Validation of forward stepwise selection was optimized with one variable and had a test error of $3.6 \times 10^{12}$ and validation of backward stepwise selection also chose one variable and had a test error of $2.8 \times 10^{1}2$.

LASSO and ridge regression were used as shrinkage methods to minimize the residual sum of squares. The best $\lambda$ for LASSO was $186,698.9$ and the mean squared error rate was $2.7 \times 10^{12}$. Ridge regression produced the best $\lambda$ of $3,751,216$ and a mean squared error rate of $3.46 \times 10^{12}$. LASSO outperformed the ridge regression method and all previously used regression methods.

**Results from tree-based methods:**

Several tree-based methods were used including pruning, bagging, random forest, and boosting. A tree was produced in Figure 7, and it predicted revenue with a test error of $7.6 \times 10^{12}$ and assumed $P28$ to be the variable most important to predicting revenue. We pruned the tree to improve test error and to simplify the model, but it did not change the test error.

Bagging outperformed the tree with a mean squared error of $3.8 \times 10^{12}$. Random forest model was even better with a prediction mean squared error of $3.7 \times 10^{12}$. Boosting was the last tree-based method used and predicted revenue with a test error of $4.7 \times 10^{12}$. We presented the MSE from all the supervised learning method in Table 1.

The results showed that the LASSO method outperformed all other methods. Therefore, we used the LASSO to train our model on the entire training dataset and determine the coefficients for the model to be used on the test dataset for prediction of revenue. Table 2 shows the coefficients for the model using the LASSO. The coefficients from the LASSO model trained with the training dataset were used to predict revenue for the $100,000$ observations in the test dataset. Table 3 shows the first 6 revenue predictions from the test observations.

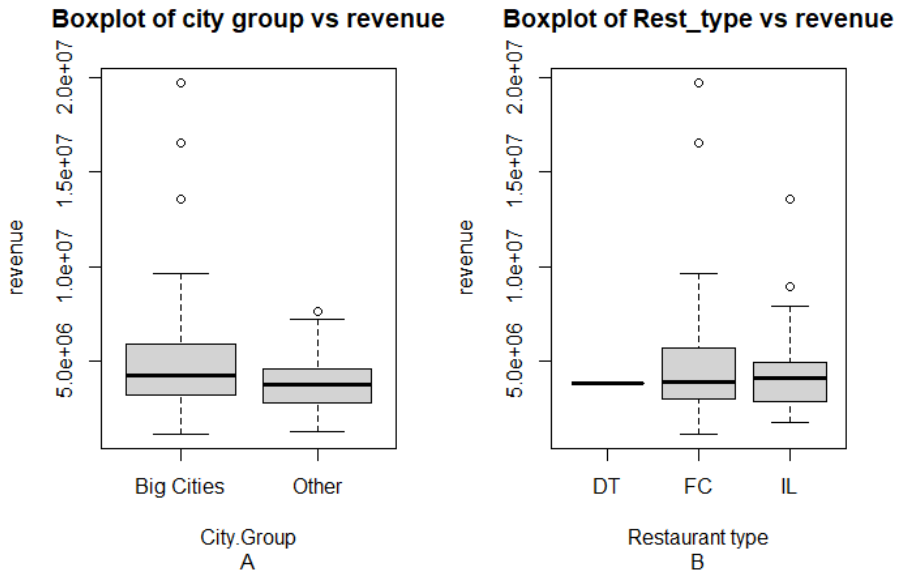


**Fig. 2:** Box plot of revenue by city type (A) and Box plot of revenue by restaurant type (B)

**Table 1:** A comparison of the test error for all methods evaluated in this study

| Methods | Test Error |
|---|---|
| Regression Methods | |
| Linear Regression | $8.0 \times 10^{12}$ |
| Forward Stepwise – BIC | $3.6 \times 10^{12}$ |
| Forward Stepwise – $C_p$ | $5.9 \times 10^{12}$ |
| Forward Stepwise – adjusted $R^2$ | $7.4 \times 10^{12}$ |
| Backward Stepwise – BIC | $3.1 \times 10^{12}$ |
| Backward Stepwise – $C_p$ | $6.4 \times 10^{12}$ |
| Backward Stepwise – adjusted $R^2$ | $6.1 \times 10^{12}$ |
| Backward Stepwise Validation | $3.6 \times 10^{12}$ |
| Ridge Regression | $3.5 \times 10^{12}$ |
| LASSO | $2.9 \times 10^{12}$ |
| PC regression | $3.7x10^{1}2$ |
| Tree-based Methods | |
| Tree | $7.6 \times 10^{12}$ |
| Bagging | $3.8 \times 10^{12}$ |
| Random Forest | $3.7 \times 10^{12}$ |
| Boosting | $4.7 \times 10^{12}$ |

## 5 Discussion

In this study, we predicted restaurant revenue of 100,000 regional tab food investment (TFI) restaurant locations across Turkey using the data supplied by TFI and hosted on Kaggle forum.

While the data was an overall complete and "clean" dataset, we were able to bring in other data to try and complement our study and attempted some recategorizing to increase the effectiveness of the model. The GDP data was added to try and help with recategorizing the 'city' variable that contained over 40 levels. Unsupervised learning techniques were also employed to try and reduce the levels by grouping similar cities together, but no clear patterns emerged in the grouping, so these methods were abandoned. Upon the addition of the GDP data, we found that it was not significant in predicting revenue. Only two of the obscured variables representing commercial, real estate, and demographic data were found to be significant in the linear regression. There were a few outliers, as seen in the boxplot representing the range of values

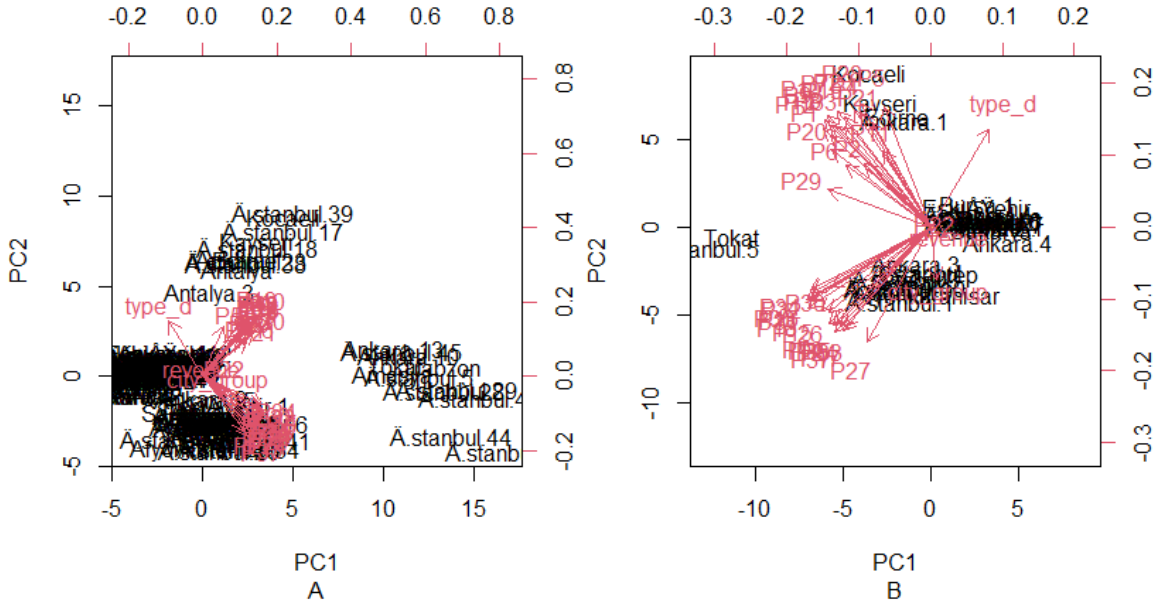


**Fig. 3:** Principal component analysis using all variables to group cities (A) and a subset of the data (B).

**Table 2:** Coefficients from the LASSO model

| Coefficient | estimate |
|---|---|
| Intercept | 3924142.53 |
| P2 | 141707.01 |
| P8 | -64007.08 |
| P13 | -61486.60 |
| P28 | 43137.33 |
| P29 | -30321.35 |
| City group | 649723.42 |

for revenue. Because the training data had only 137 observations, the test error for model evaluations was affected by these few outliers. Test error was consistently a large value in our model evaluations. This was not unexpected because our response variable contained very large values and most of the predictors are not related with response.

LASSO produced the best model for predicting the revenue in the training data based on calculations of test error. LASSO is often a very interpretable model because it uses linear regression to show the relationship between the explanatory variables and the response. This also makes LASSO a very restrictive model. Restrictive models can be especially useful if the goal of the study is inference [10]. In this study, the end goal was just to predict revenue with the most accuracy. Sometimes when prediction accuracy is the goal and interpretability is not important, more flexible models are used. However, restrictive models like LASSO can excel at prediction accuracy because the flexible models, like bagging and boosting, could lead to overfitting in the model. We believe it is for these reasons that LASSO was able to outperform the more flexible models.

We found that lasso outperformed the other shrinkage method used, ridge regression, and this is not surprising because of the large number of explanatory variables in our model. Ridge regression utilizes all the variables, while the lasso performs the variable selection by shrinking some coefficients to zero when the tuning parameter *lambda* is sufficiently

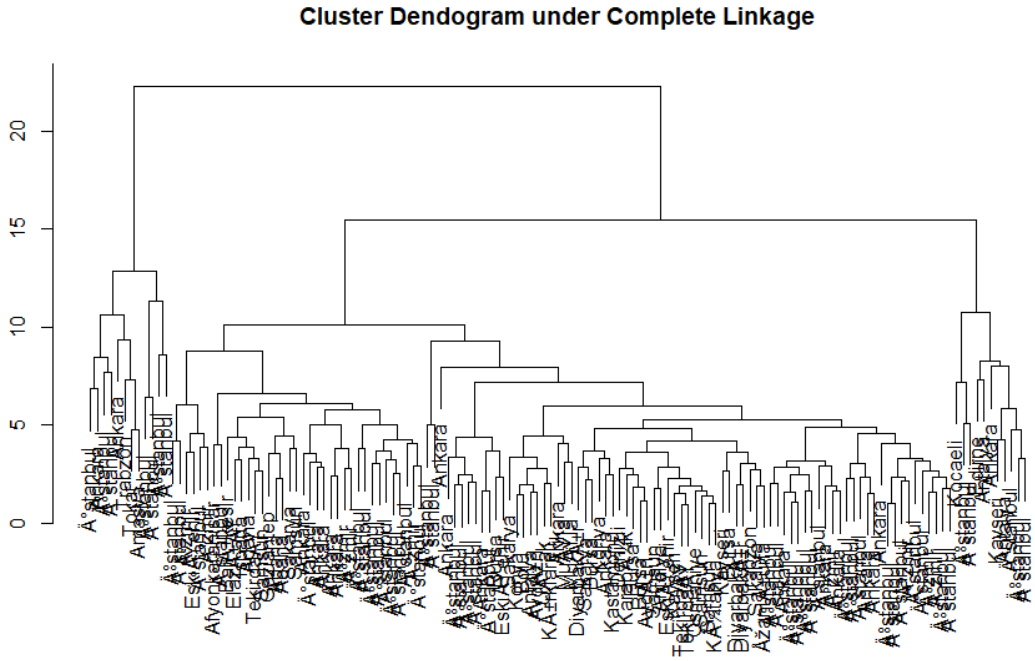


**Fig. 4:** Cluster dendrogram formed using complete linkage methods. The cities are randomly distributed throughout the clusters in the dendrogram.

**Table 3:** Revenue predictions from the first six observations in the test dataset

| Observation | Predicted value |
|---|---|
| 1 | 4033120 |
| 2 | 3959183 |
| 3 | 4584691 |
| 4 | 3937077 |
| 5 | 4177478 |
| 6 | 3760423 |

large [10]. We can say that our model had a relatively small number of predictors with significant coefficients, which is why LASSO outperformed ridge regression.

Shrinkage methods have an advantage over the best subset selection models we used in our study (forward and backward stepwise selection) because shrinkage reduces the variance as compared to these models, although this comes with an increased bias. Backward stepwise under BIC was selected as the second-best model in terms of test error, and it reduced the model to two variables (P6 and P8). Overall, in this study, LASSO was effective at parsing out the important explanatory variables in our model to predict revenue for new restaurant site locations. Other methods could have been similarly effective at predicting revenue, but LASSO has the added advantage of interpretability and simplicity.

**Conflicts of interest** The authors declare that there is no conflict of interest regarding the publication of this article.

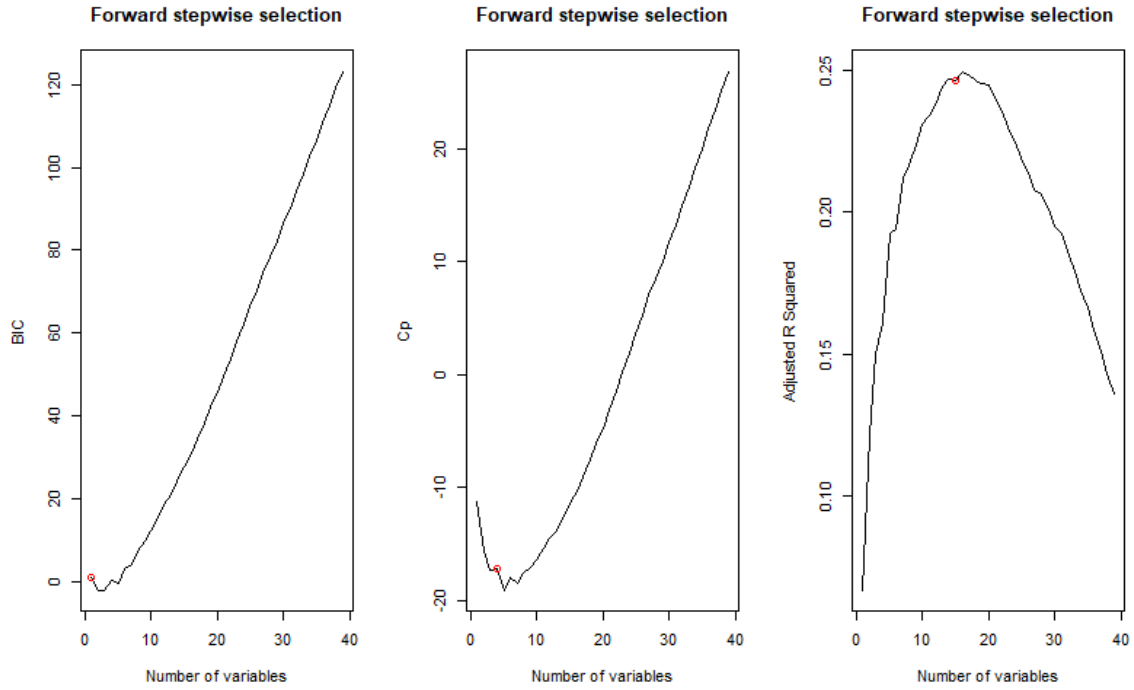


**Fig. 5:** Forward stepwise selection using BIC, $C_p$, and adjusted $R^2$ for optimizing the model.

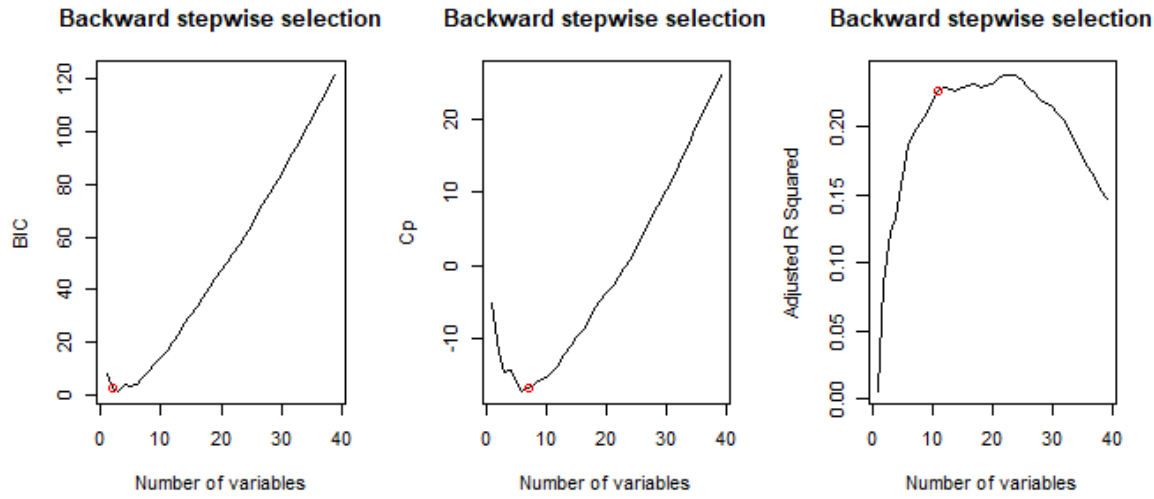


**Fig. 6:** Backward stepwise selection using BIC, $C_p$, and adjusted $R^2$ for optimizing the model.

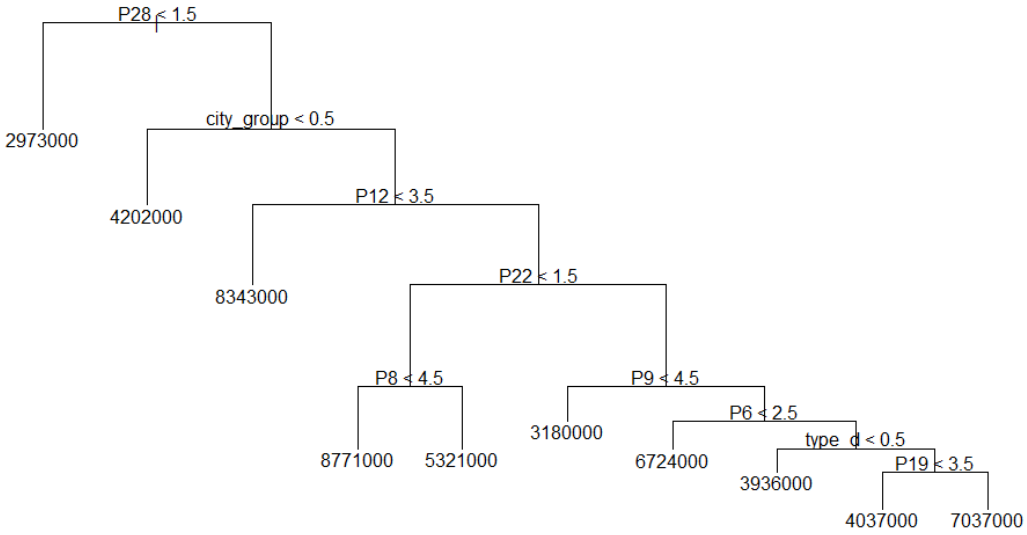


**Fig. 7:** A tree without pruning

## References

[1] K. Kowsari, M. Migliardi, S. Baddar, A. Merlo, A. Ony, Restaurant Revenue Prediction using Machine Learning, International Journal of Engineering and Science, **6**, 91-94 (2016).
[2] R. Fields, Restaurant Success by the Numbers: A Money-guy's Guide to Opening the Next New Hot Spot, Ten Speed Press, Berkeley, CA (2014).
[3] https://www.tabfoods.com/en/tfi/overview.
[4] B.M Noone, R.C Coulter, Applying modern robotics technologies to demand prediction and production management in the quick-service restaurant sector, Cornell Hospitality Quarterly, **53**, 122-133 (2012).
[5] S. Gogolev, E.M Ozhegov, Comparison of Machine Learning Algorithms in Restaurant Revenue Prediction, Communications in Computer and Information Science, **1086**, 27-36 (2019).
[6] G. S Rao, K. A Shastry, S. Sathyashree, S. Sahu, Machine Learning based Restaurant Revenue Prediction in Evolutionary Computing and Mobile Sustainable Networks, Springer, Singapore, 363-371 (2021).
[7] https://www.kaggle.com/c/restaurant-revenue-prediction/data.
[8] https://*en.wikipedia.org/wiki/List_of_Turkish_provinces_by_GDP*.
[9] http://www.trukstat.gov.tr/start.do. Accessed on 16 November (2018).
[10] G. James, D. Witten, T. Hastie, R. Tibshirani, An introduction to statistical learning, **112**, Springer, New York (2013).
[11] B. Huang, H. Liao, Y. Wang, X. Liu, X. Yan, Prediction and evaluation of health state for power battery based on Ridge linear regression model, Science Progress, **104**, 1-16 (2021).