Title: Prediction of Car Sales

Authors:

Lyuang Fu, lf1664
Junge Zhang, jz3502
Yichao Shen, ys3197
Yuanxi Sun, ys1879

Abstract: While cars become more accessible to people, auto companies are challenged to make the car model which can attract more customers in the market. Considering the increasing competition in the auto market, our project aims to build an algorithm to predict the car sales as a reference before these car companies announce a new car in the market. With the both the car configuration and sales data, our team used car sales as the target variable to build and compare multiple supervised machine learning regression models to find the best algorithm. Throughout the project, our team decided Random Forest as the most appropriate model in this case. Random Forest will not only help auto companies predict car sales based on its configuration, but also give a feature importance reference of car products.

1. Introduction and Motivation

Facing plenty of choices, people today have great freedom to choose their favorite car. However, auto companies are challenged to make the car model that can attract more customers. Considering the increasing competition in the auto market, our project aims to build an algorithm to predict the car sales that help the company determine whether the upcoming car will have a good marketing performance. There are many factors that can decide the sales of a car in the market. Our original 14 features include 'manufacture', 'model', 'vehicle type' and etc. With both the car configuration and sales data, our team used car sales as the target variable to build and compare multiple supervised machine learning regression models to find the best algorithm.

2. Methodology

2.1 Data Preprocessing

The original dataset was downloaded from the Kaggle data named "Car_sales". The dataset contains fifteen columns, one of which is "Sales in thousands", the target variable. Table 1 in Appendix shows a complete description of all columns in the dataset. Our project basically had three main steps of the data cleaning process. Firstly, we dropped missing values and outliers. In the dataset, there are some "." values which are apparently missing data and some values obviously too far away from other data, so we dropped both types of these values. Secondly, we converted text data to numerical data. For instance, in the "Manufacture" column which contains the brand information of the car, we converted it into multiple columns of different brands. These columns are all binary values that are either true or false for the specific brand. Finally, since we do not expect an auto company to predict the sale of a new car by its model name, resale value or launch date, we dropped these three unrelated columns, including "Model", "4-year resale value" and "Latest launch".

2.2 Modeling

After doing the data preprocessing, we then started to build our regression model. The first model that we chose as our baseline was the k-Nearest Neighbor model. Here we used kNN as a regressor, so for each feature vector x, the model used the k neighbors around to predict. Then we applied a 5-fold cross validation grid search on the model (and on other models as well). After tuning the model we found that the number of neighbors k=9 gave us the best

score with Mean Square Error(MSE) on the validation set of 4310.66.

Then we used linear parametric models to improve our prediction, so we chose both the Linear Regression model as well as the Support Vector Machine regression. Our grid search on SVM regressor suggested us to have a C=1000 and gamma = 0.0003. This was very unnatural. It meant that the model needed an extremely high regularization on C and almost no regularization on the gamma of the kernel. However, in most cases, these two hyperparameters move in the same direction. The validation MSE of Linear Regression was 3686.41, and the MSE of SVM was 3755.30, the SVM was even worse than a Linear Regression, which means that SVM is not very suitable for this dataset.

Then we used the Decision Tree model, as well as the ensemble models with Decision Tree. We finally chose Random Forest and Gradient Boosting regression. After the grid search, Decision Tree chose to have "max_depth" to be just 1, and "min_samples_leaf" to be 32. After checking the feature importance of Decision Tree, we found that it only took one split on the "Price in thousands" feature, and got a validation score of 4049.06. Our random forest chose to have 100 trees with "max_depth" 3 and "min_samples_leaf" 4, which made sense that each tree in Random Forest needed to be a little bit overfitting for the entire model to behave better.

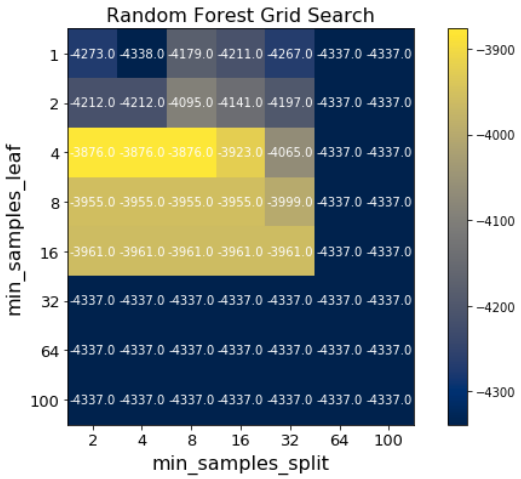

Figure 1: Random Forest Grid Search

Here is the grid search result for the Random Forest, and it provided us with an MSE of 3876.47. The Gradient Boosting model, which usually has the best performance, does not perform well this time. It used "trees with max_depth" only 1 and "min_samples_leaf" as 8, and had a MSE of 4184.15. We thought that Gradient Boosting was too complicated for this dataset, and did not behave well due to the limited size of our data.

Finally, we also tried to build a Stacking model, just by doing a linear Regression of all models above, and the test set MSE scores for all models shown in Table 2:

Table 2: Test Set Scores

Models	Test set MSE
LinReg	1645.59
kNN	2165.28
SVM	2187.52
Decision Tree	1835.95
Random Forest	1050.61
Gradient Boosting	1428.74
Stacking	1388.62

We can see that our best model is Random Forest.

3. Results

Since our dataset is very small, there are only 29 instances in the test set. Therefore, we used the validation prediction method to split the dataset into 5 folds. Then we trained models separately and combined the prediction results back together to get an MSE score on the entire dataset. Figure 2 shows the results of cross validation prediction:

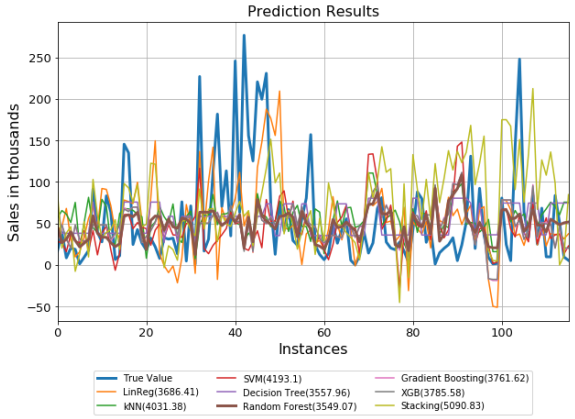

Figure 2: Prediction Results of Models

The blue line represents the actual "Sales in thousands" values of each car model, and the thick line on the very top is the prediction of the Random Forest. The best model under the cross validation prediction is still the Random Forest, but it is only slightly better than Decision Tree. Our stacking model turned out to behave much worse in this situation. It seems to have some problems while predicting after Instance 80. This is probably because the model in that fold took the high peak between Instance 30 and Instance 50 as a training set, and decided to predict much higher in the test set. We can see that our models yielded some tendencies while predicting the sales, but did not perform very well with those cars with large sales.

4. Discussion

The main strength of Random Forest is that not only we can get a reasonable performance of predicting the sales of a car, but also we are able to generate a corresponding feature importance analysis, which is a natural derivative from the model. This can help us understand what factors can potentially affect the sales of the car and the car manufacturer can design a new car based on such reference in the future. For instance, in Figure 3, we can see the price of the car accounts for 27% of the importance, which means the car company should pay sufficient attentions to label a reasonable price of a new car.

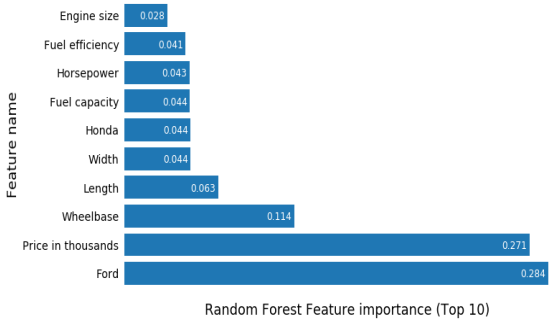

Figure 3: Feature Importance

Also, we can see that the wheelbase and length of car account for about 25% in total, which means customers care a lot about the size of the car based on their personal needs. How to design a car with appropriate size is a big challenge but also the key to increase the sales of car. However, the most important factor shown in the bar chart is namely Ford, possibly because of the distribution of data so that it is not so interpretable in the analysis.

The cons of Random Forest are reflected in two aspects. One is that we cannot predict the sales over 200 thousand accurately, which is shown in Figure 4.

	Manufacturer	Model	Sales in thousands
40	Dodge	Ram Pickup	227.061
49	Ford	Taurus	245.815
52	Ford	Explorer	276.747
55	Ford	Ranger	220.650
58	Honda	Accord	230.902
137	Toyota	Camry	247.994

Figure 4: Result of Prediction

The corresponding reason is probably due to the few such instance trained for the model and the model does not learn much about the information how a car can reach the sales over 200 thousand. Since we have little data in hand (around 150), the model still has space to improve. However, for auto companies, this will not be the ceiling and it is much more convenient for them to collect relevant data. Another drawback of the model is that it is not so easy to understand and edit the model without specific knowledge. The result is easy to understand but how to improve the performance requires professional understanding and experience.

However, generally, the results are reasonable enough and the relevant issue discussed above can be solved or improved. The limitation of our model is that we have no access to update the data of any new car model, thus we are not able to update the model over regular period. Any new changes in the car mode can have influence on the performance to predict the car sales. In addition, the model we finally get is a bit naïve and basic, which cannot completely fit the practical demand of the auto companies. Therefore, it is necessary for the auto companies to hire a data science team to tune, modify and update the model.

5. Conclusion

In summary, our group applies several machine learning algorithms to predict the sale of a car based on relevant features of the vehicles. We find out that Random Forest achieves the best performance and it can naturally provide us the summary of importance of every feature. It is crucial for auto companies to learn what factors can affect the sales and design the car with the help of the analysis. However, due to the limitation of data we can access, the model cannot predict the sales over 200 thousand accurately and also we cannot update the model in a timely manner. In the future, we will try to improve the model by getting more new samples. Moreover, with more advanced knowledge we learned, our basic and naïve model can fit to more practical industry scenario.

References

Kaggle. Car_sales. [online] Available at: https://www.kaggle.com/sunyuanxi/car-salespython/data [Accessed 16 Nov. 2018].

Foster, P., and Tom, F. Data Science for Business. New York: O'Reilly Media. 2013.

Ndaye, E., Le, T., Fercoq, O., Salmon, J., and Takeuchi, I. Safe Grid Search with Optimal Complexity. *AISTATS,* 2018.

He, J., Levine, R. A., Fan, J., Beemer, J., and Stronach, J. (2018). Random Forest as a Predictive Analytics Alternative to Regression in Institutional Research. *Practical Assessment, Research & Evaluation*, 23(1):1-16,2018.

Appendix

Table 1: Dataset Feature Description

Manufacture	Str
Model	Str
Sales in thousands	Float
4-year resales value	Float
Vehicle type	Str
Price in thousands	Float
Engine size	Float
Horsepower	Float
Wheelbase	Float
Width	Float
Length	Float
Curb weight	Float
Fuel capacity	Float
Fuel efficiency	Float
Latest launch	Datetime