Car Price Prediction with Ensemble Models

Emircan Erol
Faculty of Computer and Informatics Engineering
Istanbul Technical University
erole20@itu.edu.tr
150200324

***Abstract*—Comparison between Ensemble, XGBOOST, Random Forest, Decision Tree models implemented for car price prediction. Categorical features frequency encoder performs better than label encoding. Stack similar method is better than using best model alone.**

***Index Terms*—Ensemble learning, Car price prediction, Machine Learning**

I. Introduction

Car prices can be manipulated by some vendors easily, and providing a reference price for a specific car can be a good solution for this reason. Also, countless organizations in some fields such as insurance, banking car price prediction is highly demanded model to insure and mortgage.

II. Related Work

Xiong et al. found that [1] using stack ensemble method is the best match for house price prediction, with a slight improvement than single model. Assuming house and car price predictions are similar in development perspective, similar performance can be achieved.

III. Data

Two different datasets are used in the making of this project, first one was small and could not meet the need, for this purpose current large dataset is used.
Data obtained from Kaggle [2] and has 372 thousand samples.

Data	Description	Type
Vehicle Type	Type of vehicle (e.g. SUV, sedan, etc.)	Categorical
Brand	Brand of the car	Categorical
Fuel Type	Type of fuel	Categorical
Model	Model name	Categorical
Seller	Type of seller (private or dealer)	Binary
Offer Type	Type of offer (e.g. sale, repair, etc.)	Binary
Gearbox	Type of gearbox (manual or automatic)	Binary
Not Repaired Damage	Whether the car has any damage or not	Binary
Price	Price of the car	Ordinal
Power	Power of the car in PS	Ordinal
Kilometer	Mileage	Ordinal
Created Timestamp	Date the car was created	Ordinal
Age	Age of the car	Ordinal

Preprocess

Binary and ordinal data directly converted to numbers, categorical data could be processed in three ways first, as it is (then will be used with XGBOOST categorical), secondly with label encoding, thirdly with frequency encoding. Missing values are imputed with iterative imputer. Using KNN imputer was computationally very expensive because KNN is a lazy algorithm. IQR and manual cutting methods are used in outlier detection of ordinal features.

IV. Implementation

A. Decision Tree

Leaning capacity of the model was not satisfactory, therefore this method is not used in ensemble part.

B. Random Forest

Size of the model and training time was dramatically higher than other models. Maximum tree depth is optimized as seen in figure 1. Random forest was remarkably better than decision tree. In order to understand categorical data handling effects, random forest is implemented with 2 different data mentioned in the data section.

Dataset with label encoder was not an appropriate approach because it does not use any information to represent categorical features. Label encoder is randomly choosing sequential integers for encoding. Label encoder gives misleading information to loss function.

Frequency encoding is a better approach for categorical data, especially for trees because it uses more information than the label encoder and loss between frequent or infrequent classes are low but between frequent and infrequent loss is high.

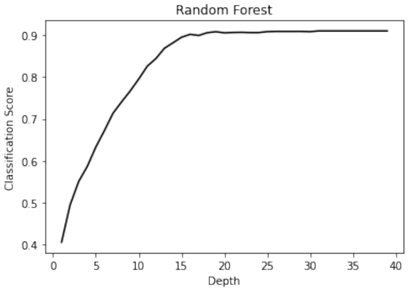

Fig. 1. Feature Importance

C. XGBOOST

In order to understand categorical data handling effects, XGBOOST is implemented with 2 different data mentioned in the data section. Frequency encoder and categorical feature of XGBOOST, gave very similar results with a slight difference, XGBOOST feature was better. PCA is implemented with all possible dimensions but did not give any better result as seen in figure 2, which yields that curse of dimensionality is avoided.

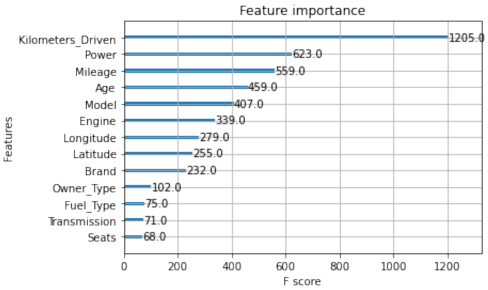

Fig. 2. Feature Importance

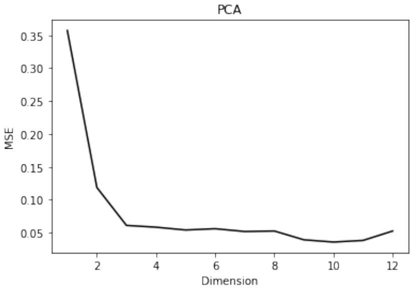

Fig. 3. PCA

D. Neural Network

Neural network did not give any meaningful result.

E. Blend Ensemble

Random forest and XGBOOST results are blended with 0.6 and 0.4 ratio. The result was slightly worse than random forest regressor, since the gap between random forest and XGBOOST was significant.

F. Boosting

Output of random tree is fed to XGBOOST training as another column, XGBOOST output is chosen as final result. Although the model was overfitted in the training set, it achieved the best score for the test set.

V. Results

Model	RMSE	MAPE
Decision Tree	30665	%6.99
Random Forest with LE	48005	%9.72
Random Forest with FE	24915	%5.95
XGBOOST	27677	%6.63
Blending	27677	%6.04
Boosting	27677	%5.60

VI. Feature Work

Many websites have textual data which contains adequate information for price prediction, mining of these data could result in better performance.

Classification of cars as luxury, economic or SUV, sedan and then training a regression model could perform better especially in more data.

VII. Conclusion

Finally, car price prediction is indispensable for stability, and it facilitates the work in the banking, insurance and car trade industries. Results shows that boosting is the best method and frequency embedding for categorical features increase the performance.

References

[1] Xiong, S., Sun, Q., Zhou, A. (2020). Improve the House Price Prediction Accuracy with a Stacked Generalization Ensemble Model. In: Hsu, CH., Kallel, S., Lan, KC., Zheng, Z. (eds) Internet of Vehicles. Technologies and Services Toward Smart Cities. IOV 2019. Lecture Notes in Computer Science(), vol 11894. Springer, Cham. https://doi.org/10.1007/978-3-030-38651-1_32

[2] Devastator, T. (2022, December 6). Used Cars. Kaggle. Retrieved January 1, 2023, from https://www.kaggle.com/datasets/thedevastator/uncovering-factors-that-affect-used-car-prices