Capstone 2 Report

Problem Statement

- Used car market has always been frequently discussed as the prices and depreciation of used cars are very tricky for both sellers and buyers and there is a considerable amount of transaction cost.
- This project is dedicated to provide some insights on the pricing mechanism and help to reduce the information cost.
- For both dealers/sellers and buyers, this model would provide a good reference on pricing the vehicle and price transparency, which could improve the market efficiency by reducing the information cost and simplify the decision making process.

Data Cleaning

- Data Source

https://www.kaggle.com/lepchenkov/usedcarscatalog/data
Author: Kirill Lepchenkov
The data is scraped in Belarus (western Europe) on the 2nd of December 2019.

- Data Cleaning

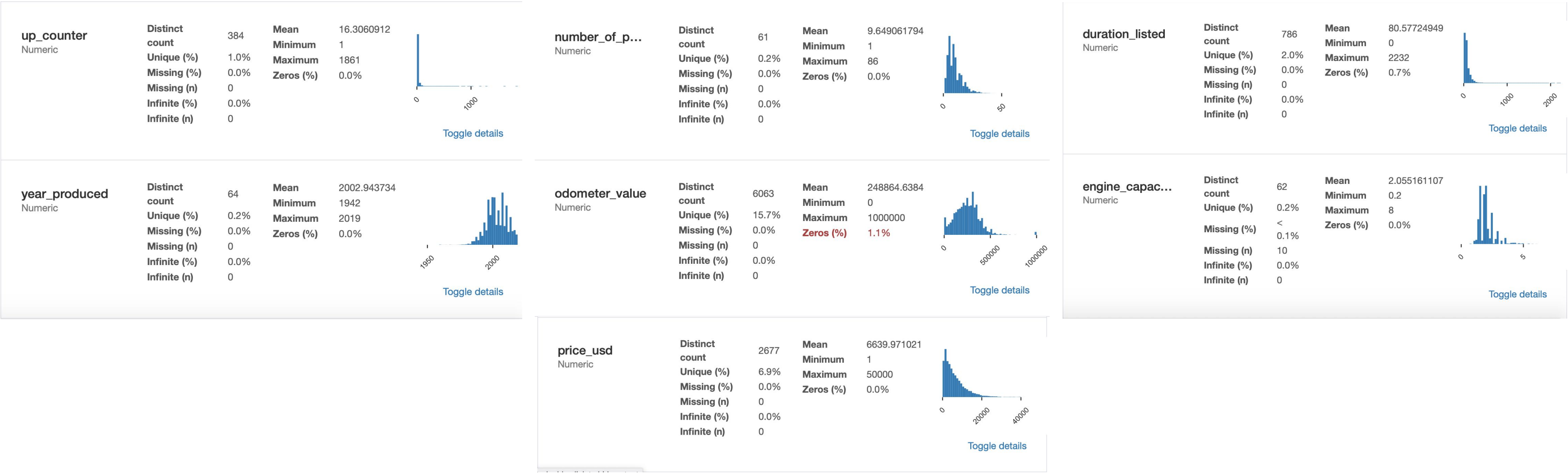
The original dataset has 38531 entries and 30 columns, all of them are non-null but 10 of them are nan and 2 columns are float64, 18 columns are int64, and 10 columns are object.

Data Wrangling

- Boolean columns were converted to numerical values.
- Categorical columns with less than 10 distinct values: one hot encoding was applied and then high correlation columns were dropped.
- Categorical columns with high cardinality: target encoding and hashing encoding were applied to generate 2 versions of the data, and missing values were filled with mean after splitting training and testing data.

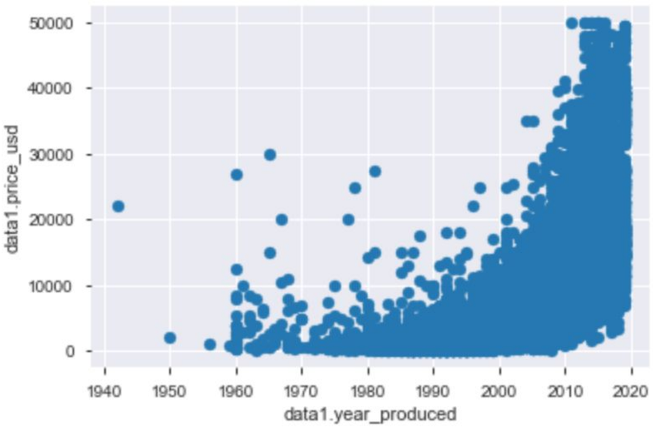
Exploratory Analysis

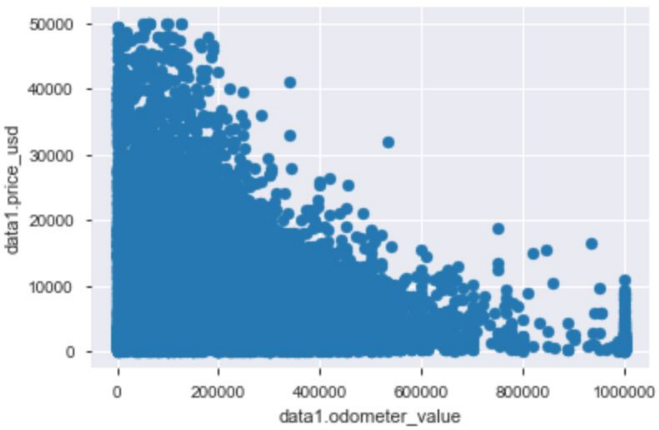
- Some columns such as price and odometer value seem to have distribution patterns.



Exploratory Analysis

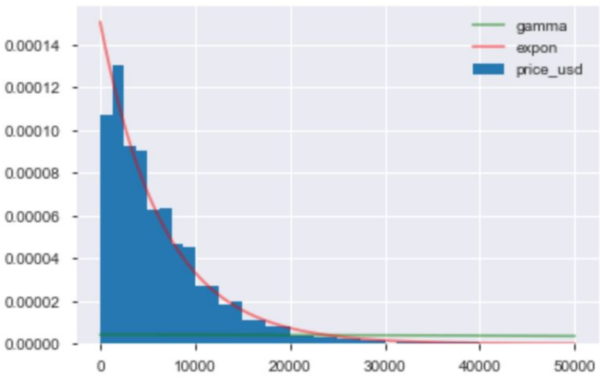
- Some columns seem to be correlated: year produced and price seem to be positively correlated while odometer value and price seem to be negatively correlated, which is aligned with common sense on used cars.
-

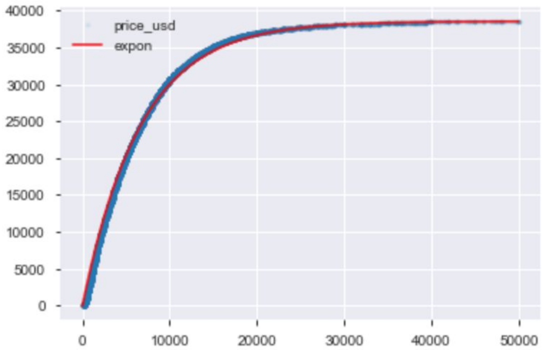


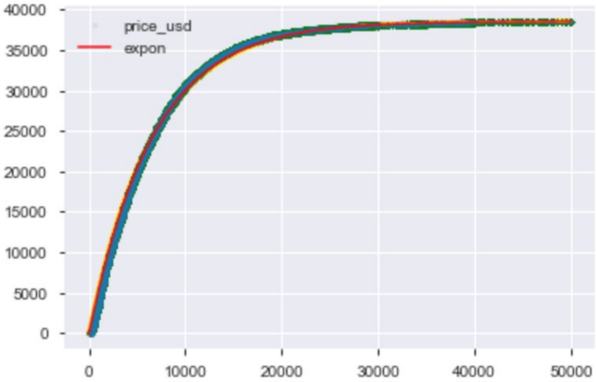


Exploratory Analysis

- From price's histogram and empirical cumulative distribution it looks like price is exponentially distributed, and the bootstrap test (n=1000) suggests so as well.

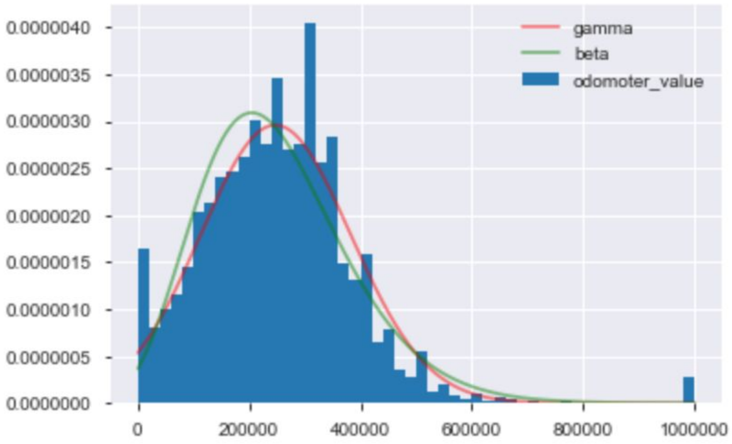


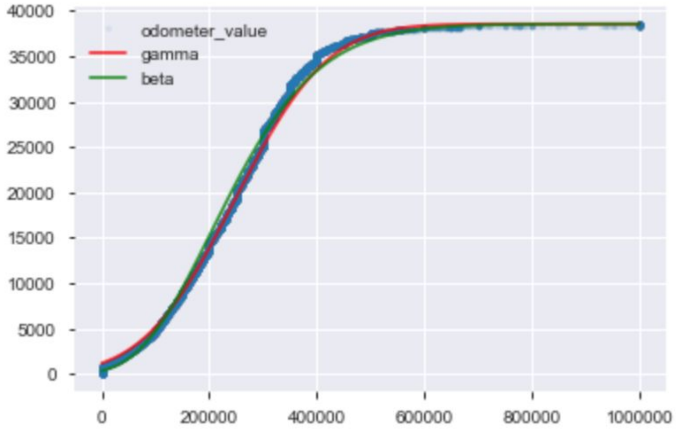




Exploratory Analysis

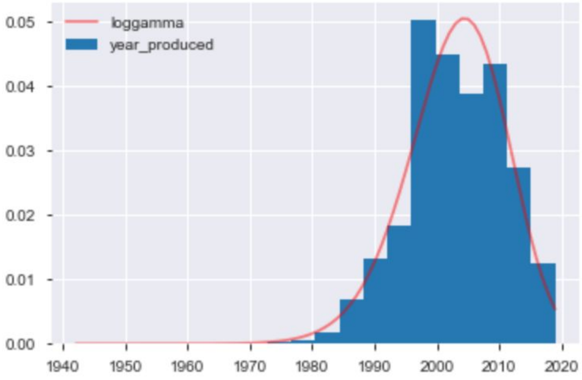
- Odometer value's histogram and empirical cumulative distribution seems to be gamma distributed.

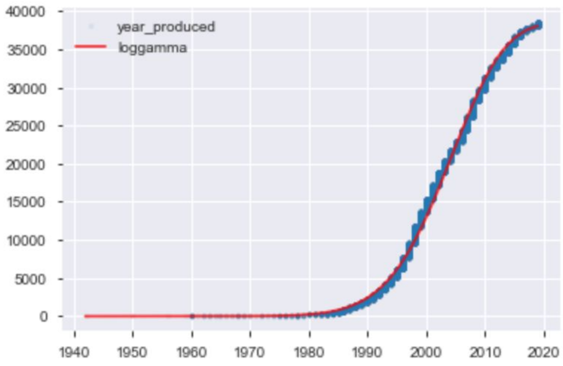


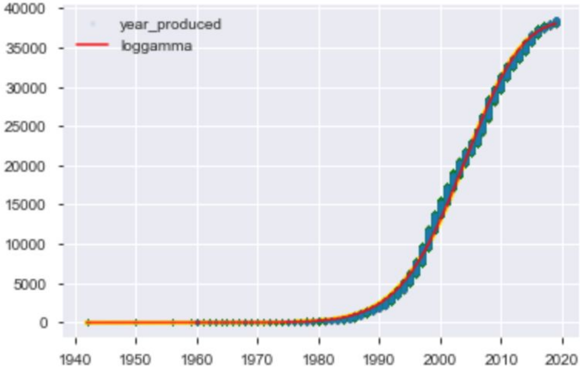


Exploratory Analysis

- Year produced’s histogram, empirical cumulative distribution and the bootstrap test (n=1000) suggest it is log gamma distributed.



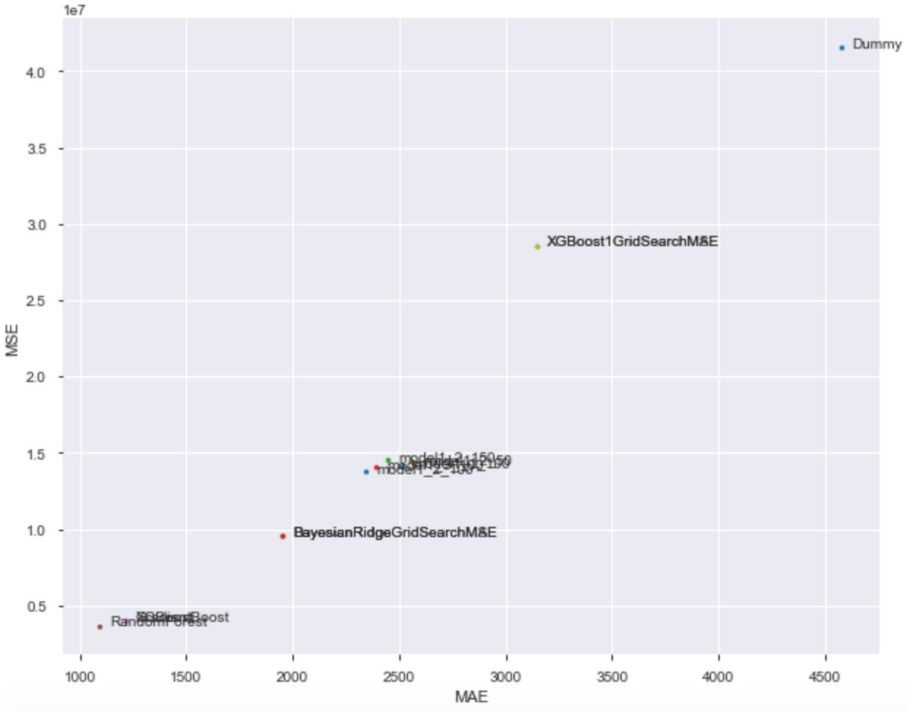


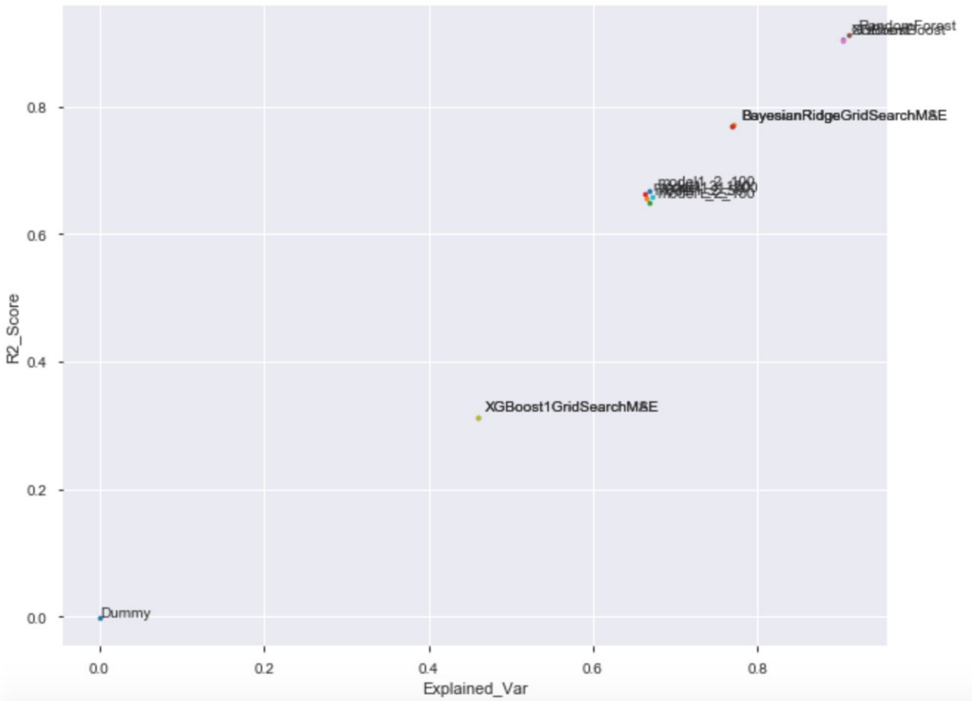


In-Depth Analysis

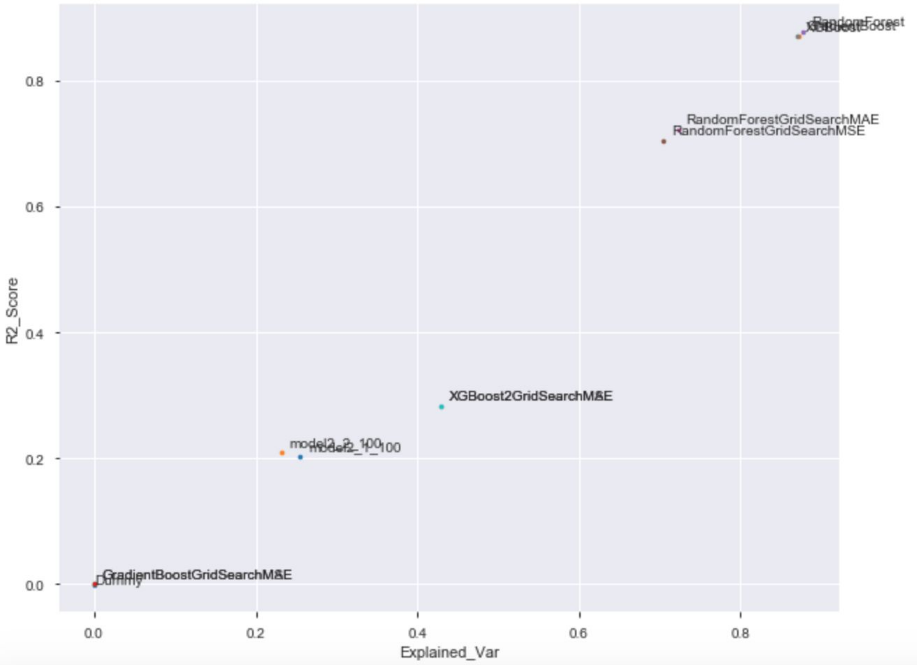
- Datasets: hashing encoding version and target encoding version; training 70%, testing 30%
- Metrics: MSE, MAE, R2 score, explained variance score. The baseline for comparison is from fitting the data into a dummy model.
- For target encoded data, Bayesian Ridge models have good performance in general.
- For hashing encoded data, Gradient Boost, XG Boost and Random Forest models all have better performance compared to Bayesian models with target encoded data, Random Forest seem to have slightly better performance than the other two ensemble methods.
- A few more ensemble models on target encoded data were tried for comparison purposes.With the same training and tuning steps, the models' performance improved and Random Forest still seems to be the better method.
- A few dense network models were tried on both target encoded and hashing encoded datasets. Models with up to 3 Dense layer and 100 nodes were trained and in general models trained with target encoded data have better performance compared to those with hashing encoded data. However, ensemble and Bayesian methods have better performance over the network models.
-

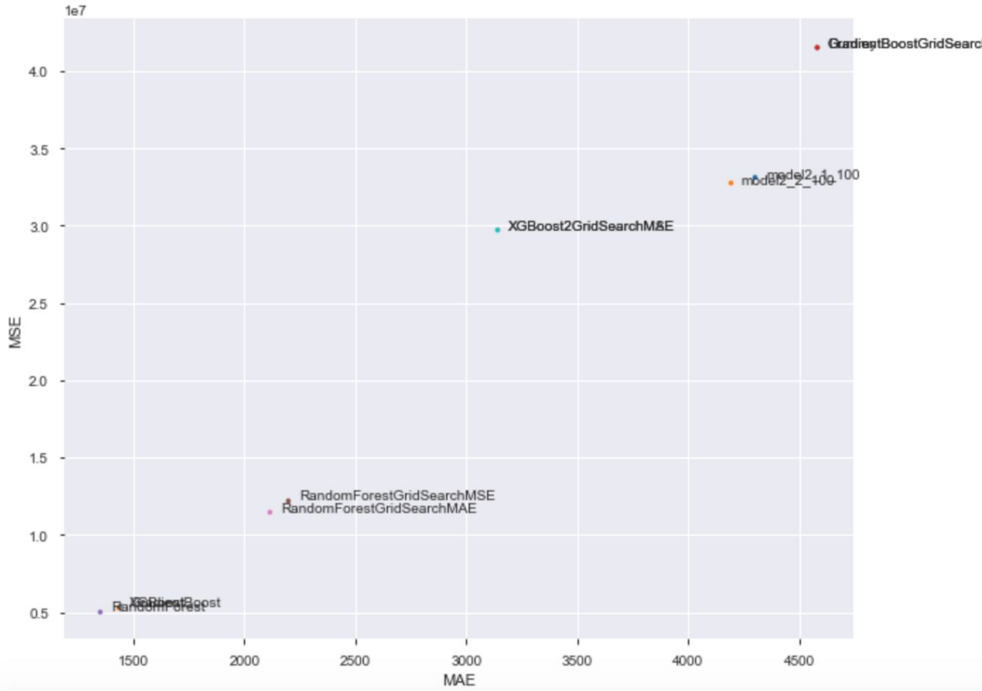
- Target encoding version





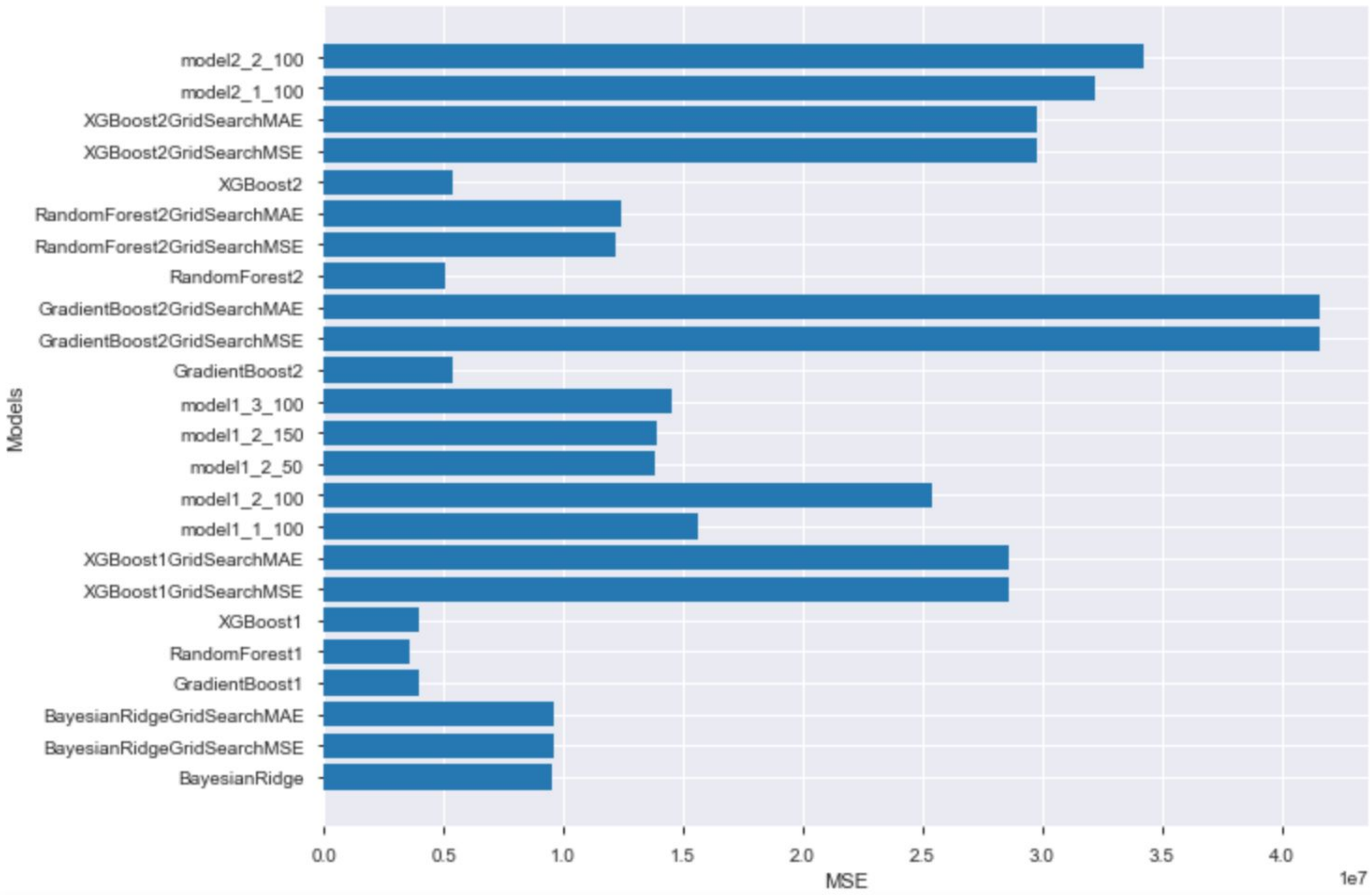
- Datasets: hashing encoding version and

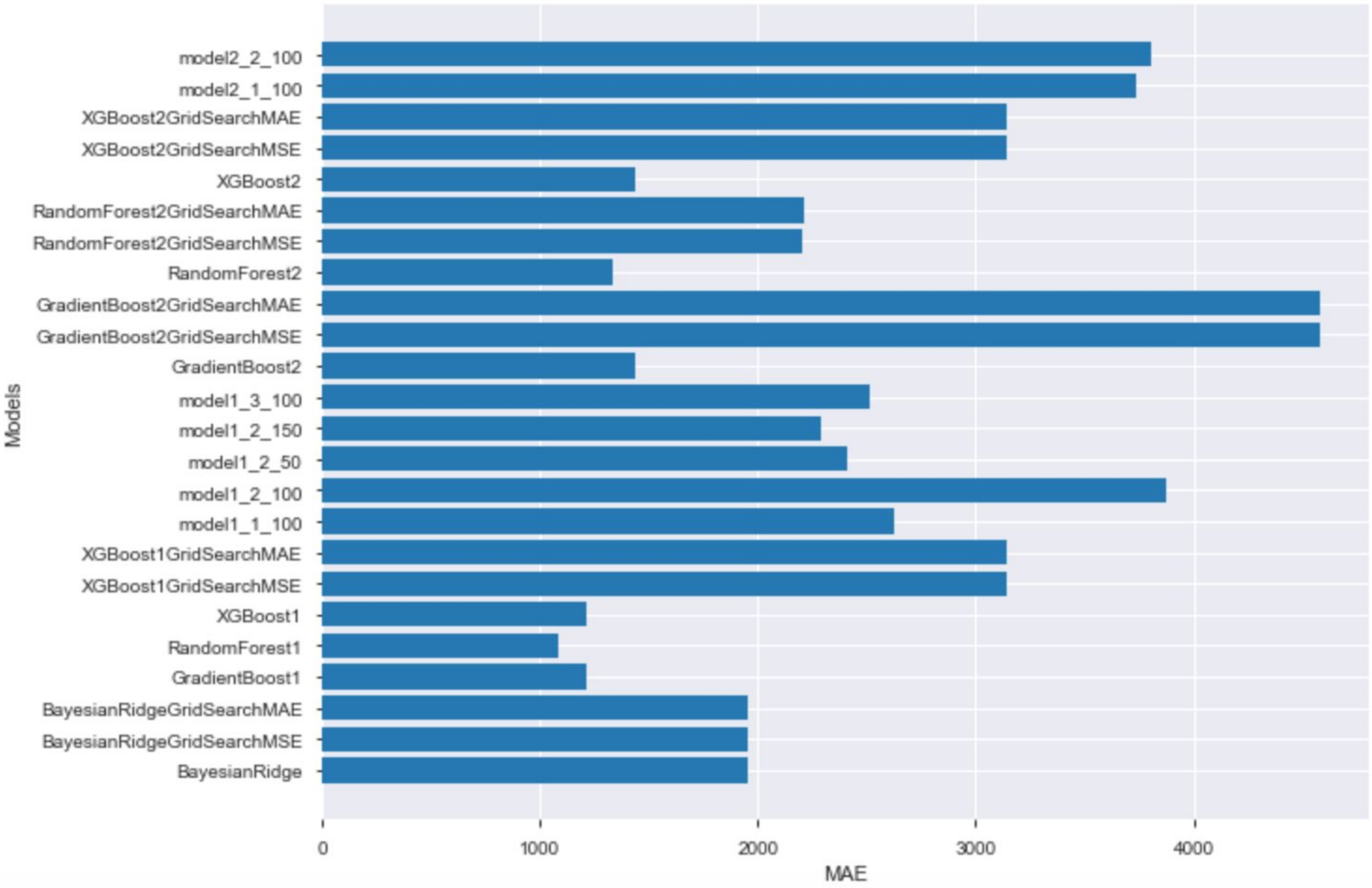


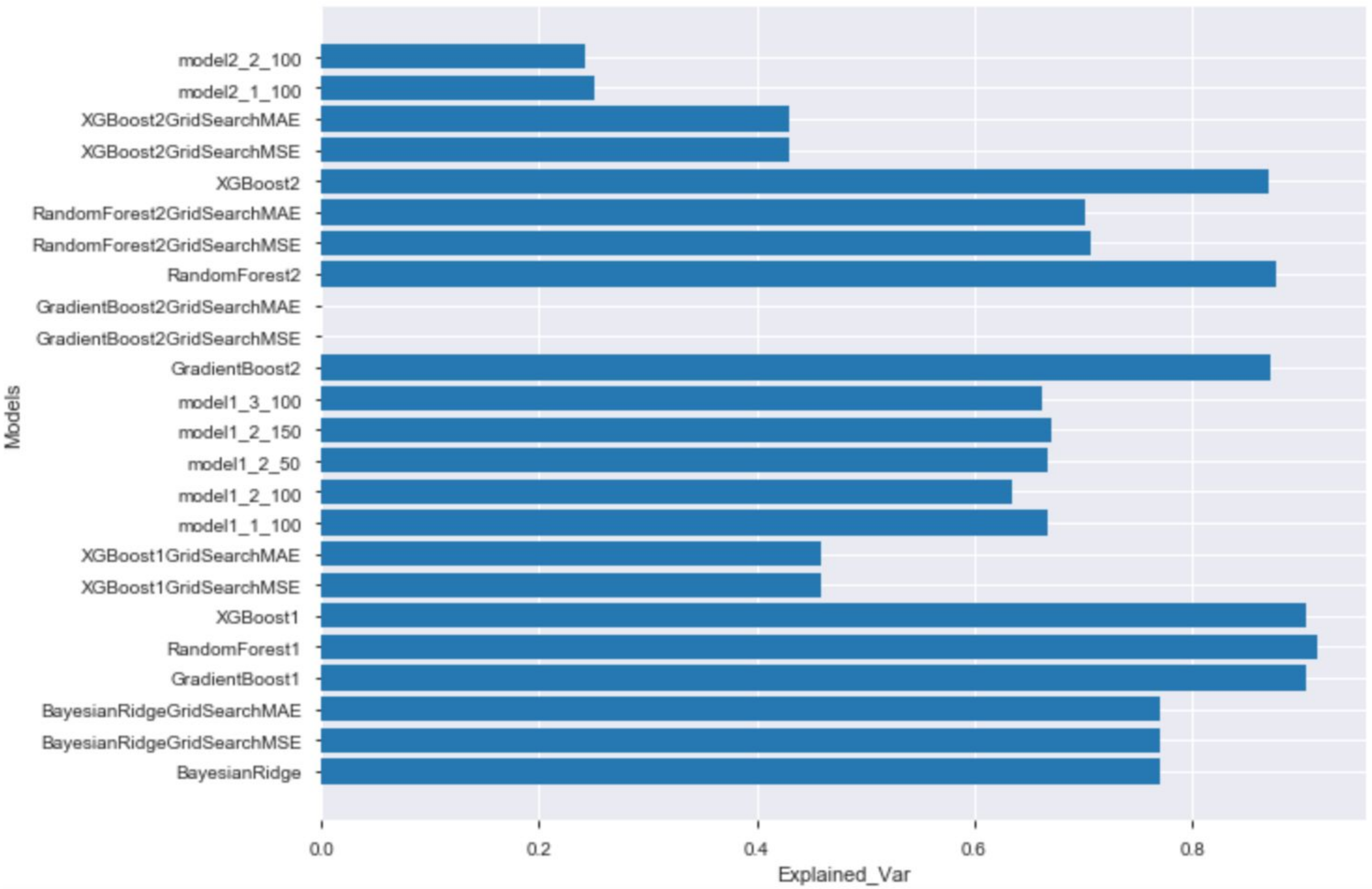


Results

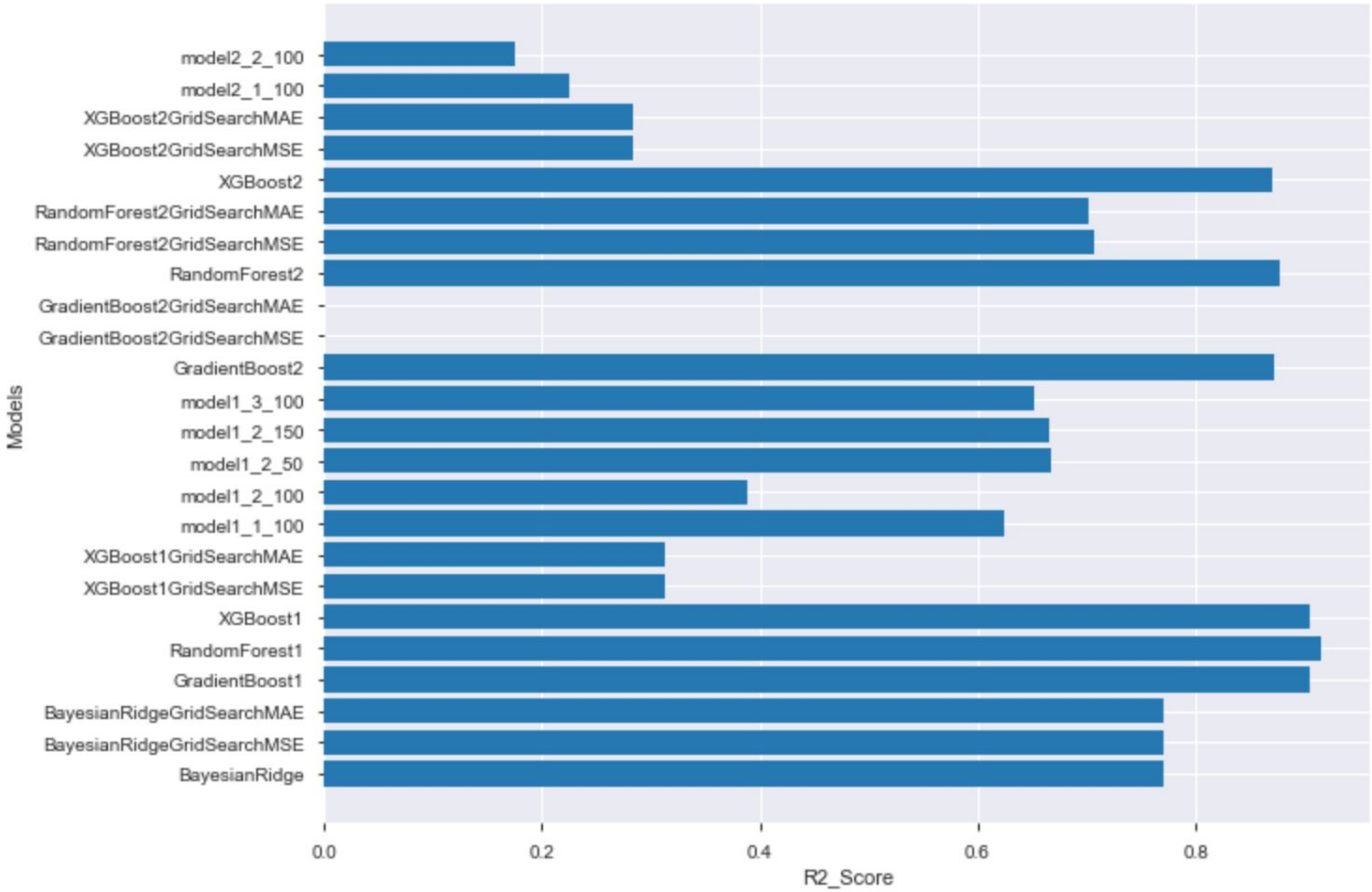
- For target encoded data, all models perform better than the dummy model, Gradient Boost, Random Forest and XG Boost have the best performance out of all models listed, Bayesian Ridge and tuned Bayesian Ridge models are less accurate but still better than network models.
- For hashing encoded data, one Gradient model with tuned hyperparameters has the same performance as the dummy model while other original ensemble models have the best performance out of all models listed. Bayesian models still seem to be better than network models.


model2_2_100
model2_1_100
XGBoost2GridSearchMAE
XGBoost2GridSearchMSE
XGBoost2
RandomForest2GridSearchMAE
RandomForest2GridSearchMSE
RandomForest2
GradientBoost2GridSearchMAE
GradientBoost2GridSearchMSE
GradientBoost2
model1_3_100
model1_2_150
model1_2_50
model1_2_100
model1_1_100
XGBoost1GridSearchMAE
XGBoost1GridSearchMSE
XGBoost1
RandomForest1
GradientBoost1
BayesianRidgeGridSearchMAE
BayesianRidgeGridSearchMSE
BayesianRidge
Models
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
1e7
MSE


model2_2_100
model2_1_100
XGBoost2GridSearchMAE
XGBoost2GridSearchMSE
XGBoost2
RandomForest2GridSearchMAE
RandomForest2GridSearchMSE
RandomForest2
GradientBoost2GridSearchMAE
GradientBoost2GridSearchMSE
GradientBoost2
model1_3_100
model1_2_150
model1_2_50
model1_2_100
model1_1_100
XGBoost1GridSearchMAE
XGBoost1GridSearchMSE
XGBoost1
RandomForest1
GradientBoost1
BayesianRidgeGridSearchMAE
BayesianRidgeGridSearchMSE
BayesianRidge
Models
0
1000
2000
3000
4000
MAE



model2_2_100
model2_1_100
XGBoost2GridSearchMAE
XGBoost2GridSearchMSE
XGBoost2
RandomForest2GridSearchMAE
RandomForest2GridSearchMSE
RandomForest2
GradientBoost2GridSearchMAE
GradientBoost2GridSearchMSE
GradientBoost2
model1_3_100
model1_2_150
model1_2_50
model1_2_100
model1_1_100
XGBoost1GridSearchMAE
XGBoost1GridSearchMSE
XGBoost1
RandomForest1
GradientBoost1
BayesianRidgeGridSearchMAE
BayesianRidgeGridSearchMSE
BayesianRidge
Models
0.0
0.2
0.4
0.6
0.8
Explained_Var



model2_2_100
model2_1_100
XGBoost2GridSearchMAE
XGBoost2GridSearchMSE
XGBoost2
RandomForest2GridSearchMAE
RandomForest2GridSearchMSE
RandomForest2
GradientBoost2GridSearchMAE
GradientBoost2GridSearchMSE
GradientBoost2
model1_3_100
model1_2_150
model1_2_50
model1_2_100
model1_1_100
XGBoost1GridSearchMAE
XGBoost1GridSearchMSE
XGBoost1
RandomForest1
GradientBoost1
BayesianRidgeGridSearchMAE
BayesianRidgeGridSearchMSE
BayesianRidge
Models
0.0
0.2
0.4
0.6
0.8
R2_Score

Summary

- Ensemble methods have better performance especially with target encoding, Bayesian Ridge is also a good alternative, and networks models I tried are not the best options.
- In general models with target encoding perform better. Due to time and computation cost concerns I didn’t tune each model very well and ARD models were not included in the results, it’s possible that there are better encoding methods such as different combinations of the encoding methods used here and models such as other Bayesian models and might give better predictions with more efforts spent on tuning the hyperparameters.

Summary

- If given limited time and computation power, I would suggest using Random Forest or XG Boost with one hot encoded and target encoded data, and update the model at least once a year and when prediction's scores fall close to a dummy model's prediction score.
- As for data collection, I suggest dropping the redundant or similar features such as engine type and if engine has gas, and maybe adding some new features such as if had major accident, if modified, times of being traded.
- I believe with good quality data, proper implementation, the suggested model would perform effectively on reducing information cost in the used car market and promote market efficiency.
- This model can be used on mobile apps and websites for consumers and combined with offers from dealers or customer reviews of dealers to help consumers make better decisions with less efforts. It can also be used for dealers to automate some processes during the transaction to cut the cost and make the price more transparent to consumers. And there are many more applications for this model that would generate economic and social benefits.