Used Car Price Prediction and Causal Inference in India

Yuhan Wang, Siming Yan

Agricultural and Applied Economics Department
University of Wisconsin Madison

Abstract- This is an article summarizing multiple trending machine learning methods by using *used car price* dataset in India (1996-2019). In this article, we discussed several presently trending machine learning tools, compared the models' performance in both prediction perspective and causal inference perspective. Methods for the prediction model quality including: ordinary linear regression with lasso and ridge penalty, support vector machine, random forests, boosting trees; Methods for the causal inference model quality including LTE lasso regression, orthogonal machine learning, causal tree, causal forests, double selection. Finally, we further discuss several promising future research directions.

Index Terms - *SVM, Random Forest, Stochastic Boosting Machine, Orthogonal Machine Learning, Double Selection, Causal Forest*

I. INTRODUCTION

The Indian automobile industry has incredibly large number of varieties [1]. India has surpassed Germany and become the fourth largest automobile market in the world [2]. With large number of brands and multiple types manufactory, we are able to analysis and build up the relationship models among car characteristics and price, by using data set from India used car price (1996-2019).

The major tool in this article is machine learning. Machine learning is a series methods of data analysis that automates analytical model building. Machine learning methods are considered to be a branch of artificial intelligence, which based on the idea that systems can learn from data, identify patterns and make decisions with

minimal human intervention [3]. We choose machine learning as it involves multiple different models, which enable us to build up variety models to compare our results and figure out pros and cons of these methods.

II. Data Descriptions

The data is obtained from India used car market, ranging from year 1996 to year 2019, 7253 observations are included. Original dataset includes 13 variables, including: car brand name, transaction location, year of the transaction, kilometers driven previous to the transaction, fuel type, transmission type, whether the car is first hand or else, mileage, engine capacity, power with unit of bhp, seats number, new car price and transaction price. We removed *new price* and all missing values in *price*, since missing values of *price* is randomly appeared in our dataset throughout all variables we take into consideration. Finally, we left with 5872 observations in our dataset.

Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage	Engine	Power	Seats	Price
Class :character	Class :character	Numeric	Numeric	Class :character	Class :character	Class :character	numeric	numeric	numeric	numeric	numeric
Car Brand	City in India	Year of transaction (1996-2019)	Log kilometers driven before transaction	takes value from one of (CNG, Diesel, Petrol, LPG, Electric)	takes value from one of (Manual, Automatic)	takes value from one of (First,Second,Fourth & Above,Third)	mileage in unit km/kg or kmpl	Capacity of Engine in unit CC	Power in unit bph	seats numer from 5 to 10	Price in INR Lakhs

figure 2.1 Data Describe

From our dataset, it is not hard to find used car prices are increasing from 1996 to 2019, in all major cities of India. We change variables of *year, location, owner type, transmission type, fuel type* into dummy variables, the total variable in the end is 47. [4]

III. Prediction Methods

In order to obtain best predictors, we randomly split our dataset in to 75% of training set and 25% of test set. While training prediction model, we use 10-folds validation to choose best model tuning parameters. This approach involves randomly dividing the set of observations into k folds of approximately equal size. The first fold is treated as a validation set, and the method is fit on the remaining $k - 1$ folds. The mean squared error, MSE, is then computed on the observations in the held-out fold [5]. The best model always has lowest CV_k, in our article, is represented by RMSE of model prediction on the test set.

$$CV_k = \frac{1}{k}\sum_{I=1}^{k} \mathrm{MSE_i} \tag{III-1}$$

1. Linear Regression with Ridge and LASSO Penalty

Linear regression with ridge and Lasso penalty can let us fit a model containing all predictors has significant influence on the dependent variable and several irrelevant predictors are removed by Lasso penalty. This is a technique that constrains or regularizes the coefficients estimates, or shrinks the coefficient estimates toward zero or exactly equal to zero. By using this method, we allow biasness of the model to be little larger the in OLS, to trade a significant decrease in variance of the predictors. [5]

$$\sum_{i=1}^{n}(y_t - b_0 - \sum_{j=1}^{P} b_j x_{jt})^2 + \lambda \sum_{p=1}^{P}[(1-\alpha)|b_p| + \alpha|b_p|^2] \tag{III-2}$$

Where the first term is residual standard error of OLS method, $\lambda \geq 0$ is a tuning parameter called regularization parameter. When λ is large, the two penalty will shrink coefficient estimates towards zero. $0 \leq \alpha \leq 1$ is mixing percentage stands for the mixing level of ridge and Lasso penalty. When choosing tuning parameters, as show in *figure 3.1*, we choose λ from 0 to 1, step is around to 0.05. α is choosing from 0 to 1, step is equal to 0.25.

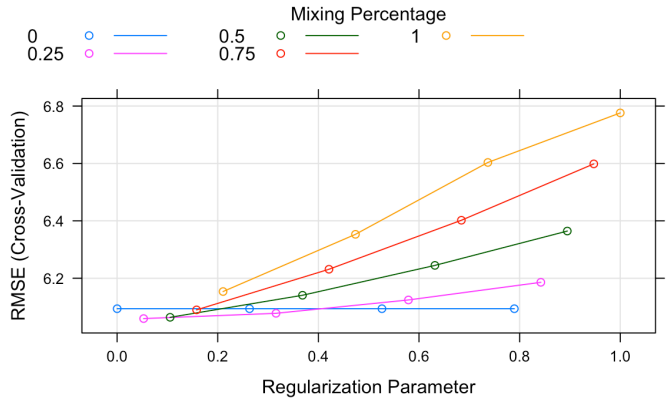

figure 3.1 choosing tuning parameters in OLS with ridge and Lasso penalty.

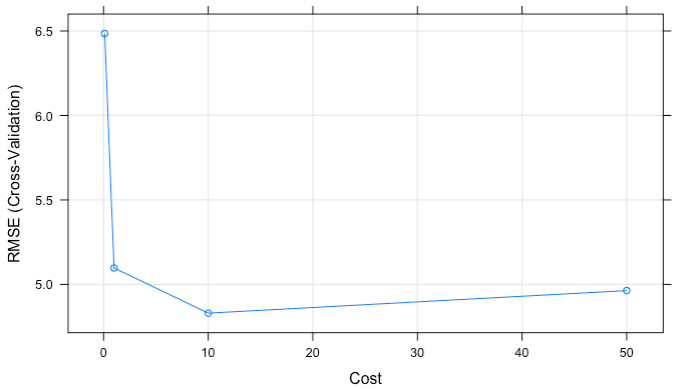

figure 3.3 choosing tuning parameters in SVM

After select out the best model with lowest RMSE in training data set when applying 10 folds cross validation, we use our model to predict price in the test dataset, to check the accuracy of our prediction. Figure 3.2 shows our results of predictions, we plot predicted price of cars on the x-axis, the real price in the test set on the y-axis,

the green line is the diagonal line which indicates the correct prediction where predicted price is equal to real price. The yellow line is fitted line. The OLS model with ridge and Lasso penalty does a great job when real price is not high, but when price is high, our model tends to underestimate the price. We generate a RMSE to clarify the model accuracy in formula III-3, The RMSE of our OLS model with ridge and lasso penalty is 5.332.

$$\text{RMSE} = \sqrt{\left(\sum((pred\_Price - real\_Price)^2)/n\right)} \tag{III-3}$$

2. Support Vector Machines with Radial basis function kernel

Given a set of training examples, support vector machine marks each point as belonging to one or the other categories, an SVM training algorithm builds a model that assigns new examples to one category or the other. It allows to enlarge the feature space used by the support vector classifier in a way that leads to efficient computations. Radial basis function kernel can implicitly map their inputs into high-dimensional feature spaces, in order to accommodate a non-linear boundary between the classes [5]. While applying SVM with radial kernel, we choose tuning parameter of cost from (0.1, 1, 10, 50)

$$\max_{\beta_0,\alpha_i} \quad M,\ Subjected\ to \sum_{j=1}^{p} \beta_j^2 = 1,$$
$$y_i\left(\beta_0 + \sum \alpha_i exp\left(-\gamma \sum_{j=1}\left(x_{ij} - xi'j\right)^2\right)\right) \leq M(1-\epsilon_i), \epsilon \geq 0, \sum \epsilon_i \leq C_i \tag{III-4}$$

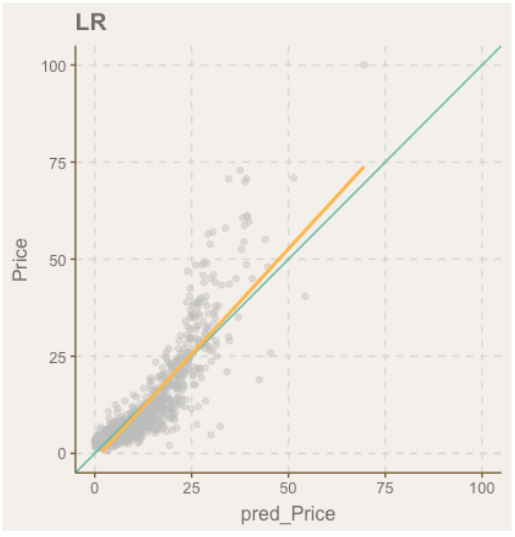

figure 3.2 predicted price versus real price in LR.

figure 3.4 predicted price versus real price in SVM

We again use our model obtained from training, predict car price in the test dataset, plot predicted value on the x-axis and real price on the y-axis, the RMSE now is 3.607, and from the figure we can easily find out that points are closer to the diagonal line even when the real price is large.

3. Random Forests

Random forests are a method operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. When building these decision trees, each time a split in a tree is considered, a random sample of m predictors is chosen as split candidates from the full set of p predictors, the algorithm is not even allowed to consider a majority of the available predictors. [5] Our tuning parameter is number of randomly selected predictors, when it is equal to 16, from figure 3.5, we can obtain the model with lowest RMSE in training process.

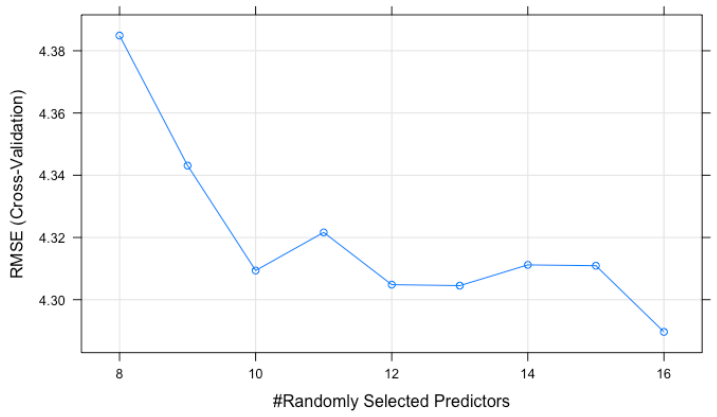

figure 3.5 choosing tuning parameters in Random Forests

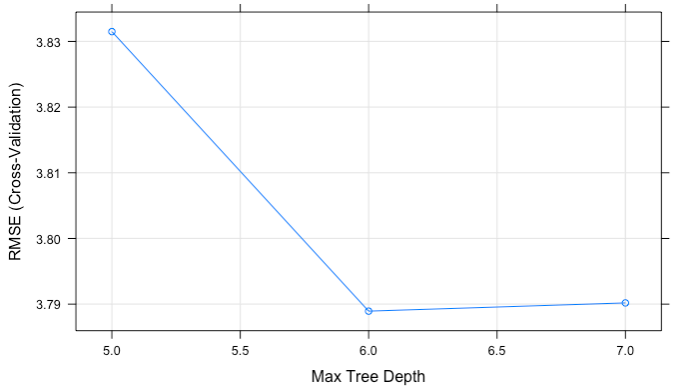

figure 3.7 choosing tuning parameters in SVM

We again plot the predicted price and real price together and calculate RMSE for the test set (figure 3.6), which is equal to 3.221.

4. Stochastic Gradient Boosting Machine

Stochastic Gradient boosting constructs additive regression models by sequentially fitting a simple parameterized function (base learner) to current “pseudo”-residuals by least squares on a training data set at each iteration. The pseudo-residuals are the gradient of the loss functional being minimized, with respect to the model values at each training data point evaluated at the current step. The training data is drawn at random (without replacement) from

the full training data set. [6] Our tuning parameter is max tree depth here, and from figure 3.7 we can see with tree depth equal to 6, we can obtain best model of training data set.

figure 3.6 predicted price versus real price in Random Forests

figure 3.8 predicted price versus real price in SGB.

We again plot the relationship between predicted price and real price, SGB shows the best ability of prediction in this data set, with the lowest RMSE equal to 2.644.

IV. Causal Inference Methods

On the causal inference part, we choose firsthand ownership type as our interest parameter, use multiple methods to figure out whether it can positively effect price.

We clean data as the same way we did in prediction part. Our data set contains 5872 observations, 4839 of them are firsthand before the transaction happens, and 1033 are second hand or more. We assume *Kilometers Driven; Year of manufacture; Model* are the confounding variables which influence both ownership type and price. There are other unobservable confounders like credit information about the car owner are not available for our analysis, which makes our inference may suffer from bias.

1. Ordinary Least Squares

We regression *price* on dummy variable *first hand,* factor variables *years* and *locations* and *car brands etc.*

Obtained following outputs:

	Est.	Std. Error	t-value	Pr (>|t|)
(Intercept)	30.6750967	8.293225	3.69881	2.20E-04
First hand	-0.0002001	0.002976	-0.06722	9.46E-01

From the output we can easily find the *first hand* parameter is negative and statistically significant at any level, which means, if the car is first hand when the transaction happens, it might decrease the car price by 0.0002.

2. Orthogonal Machine Learning

We randomly split data into roughly equally sized 5 folds, then use linear regression to fit the prediction functions with all data except k-th fold, calculate out of sample residuals for these fitted predictions on the k-th fold. The second step we collect all of the OOS residuals from the last stage, and use OLS to fit the regression. [7]

$$E[y|d] = \alpha + d \cdot \gamma \tag{IV-1}$$

The resulting γ estimate can be paired with heteroskedastic consistent standard errors obtain a confidence interval for the treatment effect.[8] The result of treatment effect is 0.8398 and the standard error is 0.1833, hence it is statistically significant.

3. LTE Lasso Regression

Our goal of using LTE Lasso Regression is to estimate the treatment effect on price when *first hand* moves independent of all other influences, and to remove the effect of other influences that are correlated with *first hand* from the treatment effect estimate. [8] In the first step we use AICc lasso to select predictors, there are about 400 predictors out of 1858 in the original data set have been selected to predict $\hat{d}$. The in-sample R square is around 0.38. In the second regression, we get the estimated treatment effect of 0.4268.

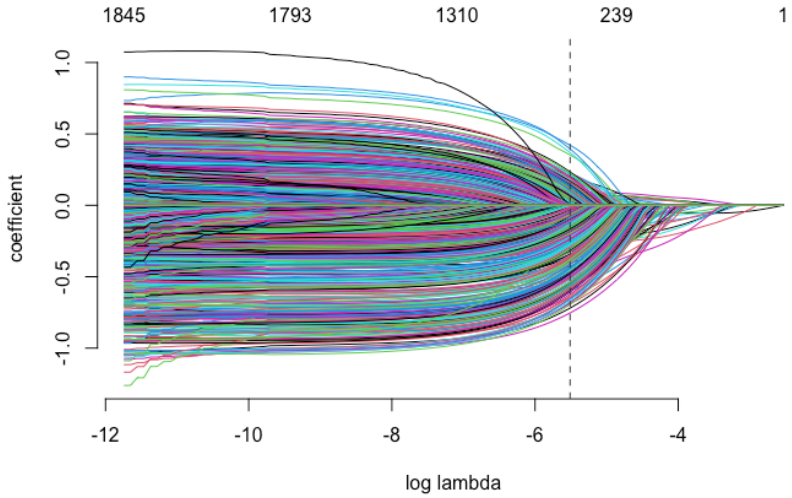

figure 4.1 Variable selection in LTE Lasso Regression.

4. Double Selection

Apply variable selection to each of the two reduced form equations and then use all of the selected controls in estimation of α. First, select variables for predicting y from (2): x_{yi}. Second, select variables for predicting d from (1): x_{di} Finally, estimate α by OLS of y on d and $x_{yi} \cup x_{di}$. Our first selection picks 505 predictors, second selection picks 50, the union has 520 predictors. Our results show that treatment effect is 0.4973 and it is statistically significant.

	Estimate	Std. Error	t-value	Pr (>|t|)
(Intercept)	6.581285	1.385004	4.751816	2.08E-06
D	0.008045	0.001377	5.841198	5.55E-09

5. Causal Tree

Apply honest causal tree to our dataset and get the conditional ATE for each subgroup. Form the figure we can see that most node have a positive treatment effect. The estimate ATE of all units is 2.5, which is significant and much higher than that of other method. From the plot we can draw three major conclusions:

1. For cars manufactured before 2011, the positive effect of the first hand on the price is very significant
2. For cars which engine higher than 1600cc the ATE is more significant.

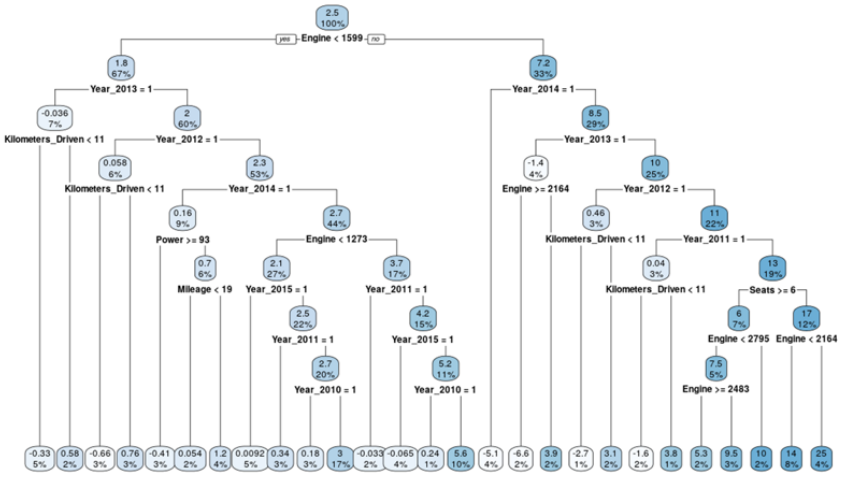

figure 4.3 Causal Tree

6. *Causal Forests*

Finally, we use causal forest to estimate ATE. We grow 2000 trees and the predictors in each tree is 64. We divide half of the trainset to estimate set and use honest method. The estimation of CATE is 2.4491 and the standard error is 0.320. The estimation of CATT is 2.4312 and the standard error is 0.321.

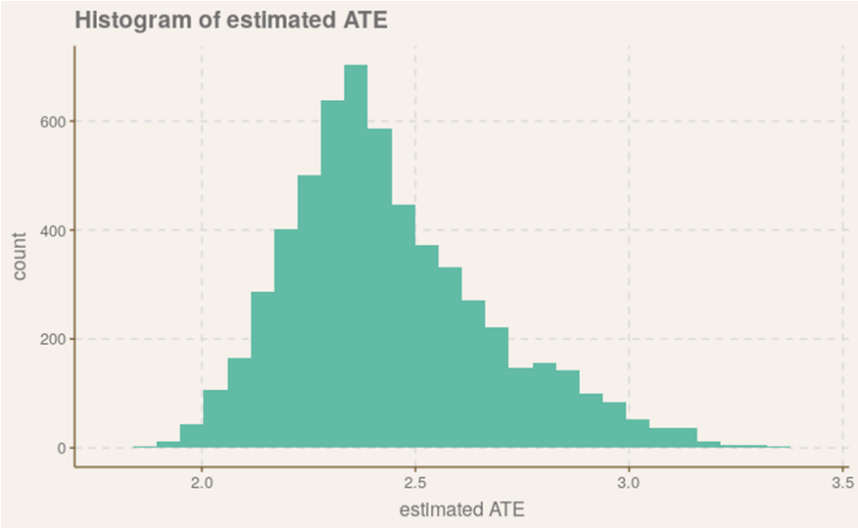

figure 4.3 Causal forests estimated ATE.

V. CONCLUSIONS

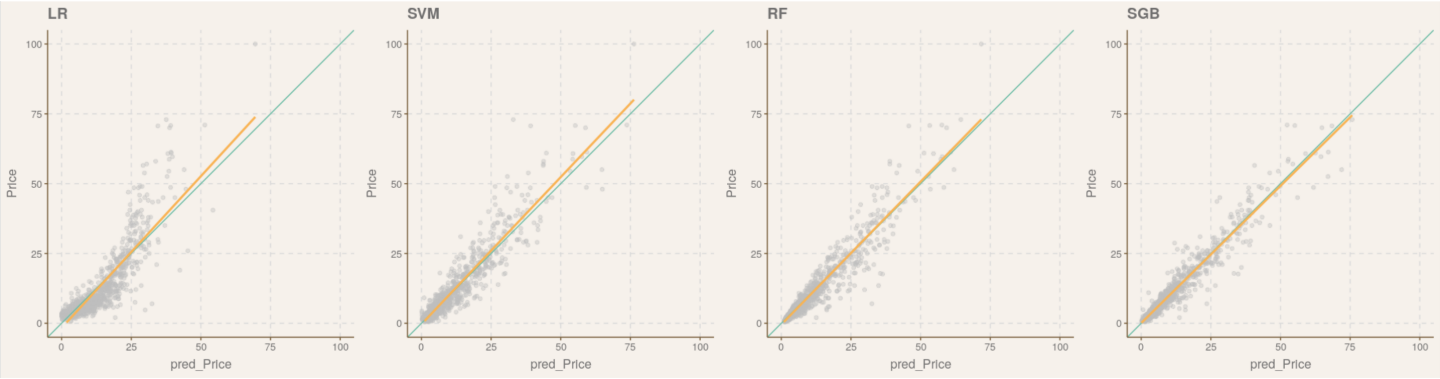

figure 5.1 Comparation of prediction methods.

In figure 5.1, we plot four types of prediction methods we used in this project, with methods change, the fitted line changes its direction towards right side of the figure with RMSE decreasing. The dots of prediction-real pair are getting closer to the diagonal line. We can assert that, in this project, stochastic gradient boosting has the best performance. These models can be very useful in India car market, the prediction is close to the real price. In the future studies, the researchers can modify these models to fit in other countries.

	OLS	Orthogonal ML	LTE Lasso	Double Selection	Causal Tree	Causal Forest
Estimation	0.5581	0.8398	0.3844	0.5141	2.5419	2.4491
Standard Error	0.1163	0.1833	——	0.1056	——	0.3205

figure 5.2 Comparation of Causal inference methods.

The estimated ATE of causal effect is significantly different between regression method and tree method. It may due to unobservable confounders which introduce biasness to our model. In order to get an unbiased result, we need more information or get a proper instrument variable. The relationship of variable *first hand* shows it for first hand cars, their price might be slightly higher than those are not. For further research, researchers can add more variables, enlarge the data set, and check other causal relationships which can influence price of used cars.

VI. References

1. Sharma, K. PRE-OWNED CAR MARKET IN INDIA: A STUDY OF MARKETING STRATEGIES OF CAR MAKERS. **2**, 5 (2012).
2. Sharmistha, M. India pips Germany, ranks 4th largest auto market now. *Econ. Times* **Industry**, (2018).
3. Hui Li. Machine Learning: What it is and why it matters. (2020).
4. Chen, C., Hao, L. & Xu, C. Comparative analysis of used car price evaluation models. in 020165 (2017). doi:10.1063/1.4982530.
5. James, G., Witten, D., Hastie, T. & Tibshirani, R. *An Introduction to Statistical Learning*. vol. 103 (Springer New York, 2013).
6. Friedman, J. H. Stochastic gradient boosting. *Comput. Stat. Data Anal.* **38**, 367–378 (2002).
7. Semenova, V. *et al.* Orthogonal Machine Learning for Demand Estimation: High Dimensional Causal Inference in Dynamic Panels. 66.
8. Taddy, M. *Business Data Science*. (McGraw-Hill Education, 2019).