Exhibit A(rt)

Adrianne Fu, Junyue Wang, Xuanhui Chen

ORIE 5740: Statistical Data Mining 2022SP

Cornell University

May 5, 2022

Abstract

Some great auction houses, like Sotheby's, Christie's and Phillps, sell sculptures that are acquired from various artists around the world. However, shipping depictions or transporting collectibles isn't as simple as shipping consumer goods. The major objective of this project is to figure out the predicted cost required to ship these sculptures to customers based on varies types of information about the artist, sculpture, customer, and the order type. We also explored what factors are the foremost crucial ones to shipping cost, as well as a few interesting findings about their relation. The results will help auction houses to make cost analysis and corresponding decisions.

1 Introduction

Knowing and understanding art is definitely an indispensable part of our life, while art inspires as well as comforts our mind. Given this, art sculptures have become mainstream choices for people to decorate their houses. However, it can be difficult to navigate the logistics when it comes to buying art, especially when many of them may be exceedingly valuable, and come from exhibitions or auctions.

In this project, we try to make predictions of sculpture shipping cost and discern most related variables with it. We utilized multiple machine learning models, including but not limited to: linear models such as Linear Regression and Ridge Regression; non-linear models such as PCR, MARS and several tree-based models (Boosting Tree, Random Forest, and XGBoost).

The results suggest that the best result (test MSE) is achieved by XGBoost (0.03) and then random forest (0.037). It shows that tree-based model has overall better performance than linear models (0.3) and other types of non-linear models. The prediction is solid, and such model can be applied in real cases for calculating sculpture shipping cost.

2 Data Overview

In this part, we will describe the data source, variables explanation, data cleansing and feature engineering of this project. We used multiple visualization methods and statistical distribution to help make decisions.

2.1 Variables

The dataset used for this project is from HackerEarth Machine Learning : Exhibit A(rt). It contains 19 variables, including 6 numerical variables, 11 categorical variables, and 2 date variables representing scheduled and delivery date. There are a total of 6,500 data points.

Essential features include:

- Features of sculpture: Height, Width, Weight, Material, Fragile, and Price.
- Features of delivery order: Base Shipping Price, Customer Location, Express Shipment, International, Transport, Scheduled Date and Delivery Date.
- Customer Information (represents details about a customer, wealthy or working class) and Artist Reputation (the greater the reputation value, the higher the reputation of the artist in the market).

2.2 Data Processing

2.2.1 Data Cleansing and Missing Value Imputation

We first removed 659 observations with negative shipping cost, then took 80% of the remaining data points as training set for the following exploratory data analysis. Table 1 lists out all the variables containing missing value. Missing value of numerical variable, i.e., "Artist.Reputation", "Height", "Width", and "Weight", are replaced by the median of the corresponding train/test set. And that of categorical variable are imputed using Missing Forest which uses random forest to do predictions in completing missing values by

Column name	Missing value
Artist.Reputation	750
Height	375
Width	584
Weight	587
Material	764
Transport	1392
Remote.Location	771

Table 1: Summary of missing value.

recursively splitting complete data into training and test set.

2.3 EDA

In this section, we implemented exploratory data analysis (EDA) to investigate the data distribution and the relationship among variables.

First, let's take a look at the response column, "Cost". Figure 1 shows the average shipping cost of each state, excluding cost that exceeds 1 million to avoid the effect of extreme values. We can see that eastern regions like PA, NC, and DC tend to have higher average shipping cost, while the opposite is true for the central regions like SD, CO, and OK. We also examined the geographical distribution of number of orders, and found that the pattern is not the same as the average price – some central regions, for example IL, AR, and KS, are on a par with eastern regions in terms of order quantity.

As for the statistical distribution of response, as shown in figure 2, the shipping cost is highly skewed with a median of 458.8. And after applying similar analysis to all the variables, we noticed that the feature "Price of Sculpture" also appears to be extremely skewed.

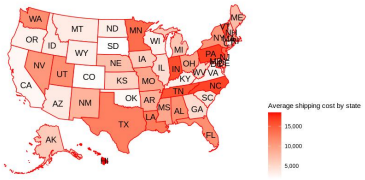


Figure 1: Average shipping cost by states

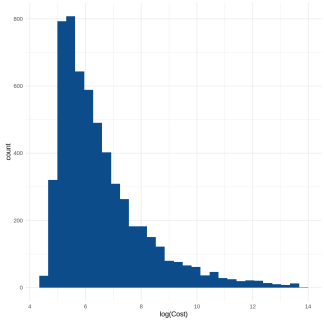


Figure 2: Distribution of log(Cost)

We are also interested in the relation between the shipping cost and material that the sculpture is made of. As in Figure 3, sculptures made of marble and stone have a more balanced shipping cost distribution, whereas sculptures made of other types of materials typically have lower shipping cost, this finding is justified by the result of box-plot.

Figure 4 shows the pairwise correlation and scatter plot between all the numerical variables, as well as the distribution of them. According to the result, cost is highly correlated with base ship-

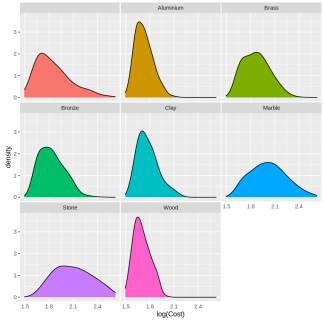


Figure 3: Distribution of log(Cost) by material

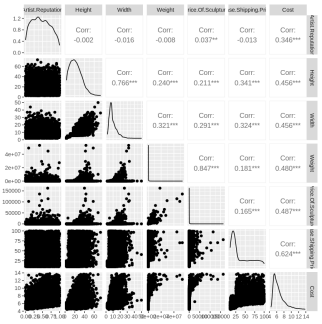


Figure 4: Correlation and pairwise scatter plot of numerical variables

ping price with a correlation coefficient of 0.624, and has moderate correlation with all the other numerical variables. As for the relation between features, unsurprisingly, we noticed that weight is closely related to the price of the sculpture and so is height and width.

2.3.1 Feature Engineering

Based on our understanding of this shipping problem, external references, and exploratory data analysis results (in the next section), we built the following new features:

- Average base shipping price grouped by mode of transport and whether it's a remote location.
- Average base shipping price grouped by material of sculpture.
- Average base shipping price grouped by state.
- Weight per base shipping price.
- Volume of sculpture ($Width * Height^2$).
- Difference between delivery date and scheduled date.

This step gives us a cleaned data set with 5841 observations, 4672 in training set and 1169 in testing set. There are 20 features, consists of 12 numerical variables and 8 categorical variables. We then used ordinal encoding to convert categorical variables, because features like "Material" is related to the density of different substances, which has a natural order.

3 Models

3.1 Feature Selection

In this section, we adopted two methods to process with feature selection to improve the machine learning process and increase the predictive power of algorithms by selecting the most important variables and eliminating redundant and irrelevant features.

First, *Best Subset Selection* with exhaustive search enumerates all possible subsets of variables. Results were selected the based upon different index, one with the largest adjusted R^2, lowest Cp and lowest BIC. It turned out that we had quite close subsets of features, with 16 variables suggested by adjusted R^2, 15 by C_p and 14 by BIC (as shown in Figure 5). Therefore, we picked the case with the largest adjusted R^2 and saved as "best1".

Further, we compared this result with *LASSO*, which has a nature to shrink variable size by using ten-folds cross validation and train MSE to select variables. All numeric variables had been standardized, and categorical variables were kept unchanged because they were already in 0-1 range.

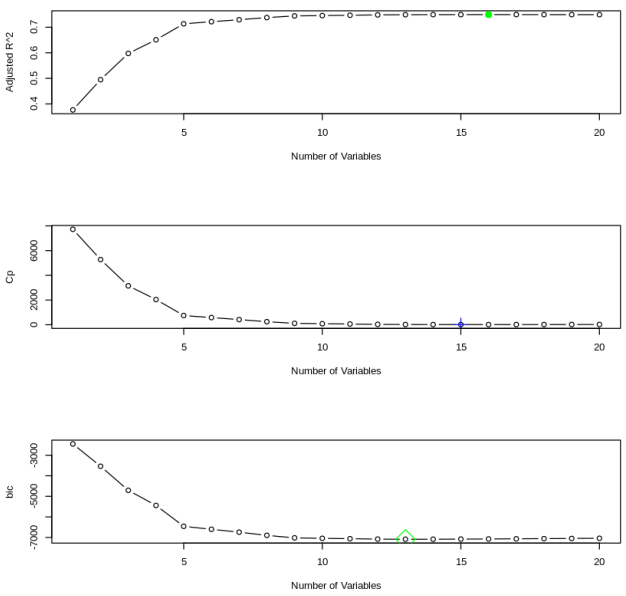


Figure 5: Best subset results with on different index

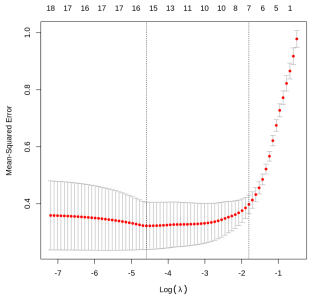


Figure 6: Lasso Lambda value vs MSE

According to Figure 6, minimum value of MSE is 0.3 with lambda = 0.009. In this case, variables with non-zero coefficients were selected and saved as "lasso1".

So far we had two sets of variables, and we are going to test which one performs better with basic linear regression model. The variable size we had is not overwhelming, and we believed that each variable can contribute more or less to the model, therefore, we kept the full size of these two sets to feed into later algorithms.

3.2 Linear Model

3.2.1 Linear Regression

To serve as a building block and baseline of model comparison, we here built a basic linear regression model with both "best1" variables and "lasso1" variables over the training set. Here, we calculated the test MSE for two purposes: 1) determine the performance of union variables set, and 2) serve as a reference for later models.

Lasso model has slightly lower train MSE value, hence we decided to stick with Lasso sets to build models. The final dataset contains 16 dependent variables. Considering certain algorithm may filter variables out automatically, here we just try to keep constant input for each model, and exactly same variable being used throughout following analysis is not guaranteed.

Figure 7 shows the fitting result of basic linear regression with "lasso1" variables. The predicted value falls roughly diagonally with the true value, yet a potential non-linear trend is observed. To further confirm, we then drew a residual plot in Figure 8, and clearly, the fitted model doesn't capture the non-linear relation between the response and features. Thus, we move on to further analysis.

3.2.2 Ridge Regression

Next we fed these variables into Ridge regression, which meant to maintain variable size when all are important. Tuning parameter λ with 10-folds cross validation was applied with 100 different

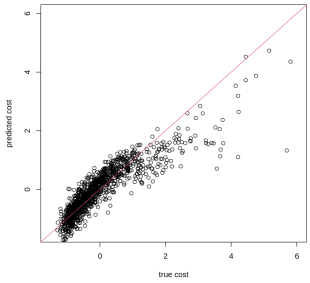


Figure 7: Fitting result of basic linear regression with "lasso1" variables

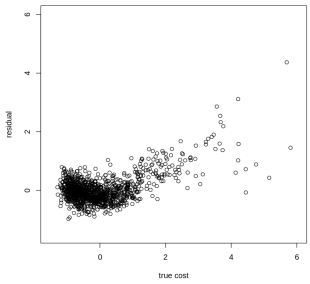


Figure 8: Residual plot of linear regression with "lasso1" variables

values. Given that ridge regression adds a regularization term to ordinary OLS with L_2 norm $\hat{\beta}^{ridge} = \underset{\beta}{argmin} \sum_{i=1}^{n}(y_i - (\beta_0 + \beta^T x_i))^2 + \lambda||\beta||_2^2$ as the tuning parameter increases, the L_2 norm of ridge coefficient decreases towards zero. This pattern is due to the increasing weight of regularization term in solving the ridge optimization problem. As the regularization term weighs more than the squared loss, the objective function can be minimized with β staying smaller. The best lambda value here is 0.163 with test MSE = 0.263. This is slightly higher than linear regression model.

3.3 PCR

We then applied Principal Component Regression (PCR) model, as a comparison to linear model fitted on features selected in section 3.1. As shown

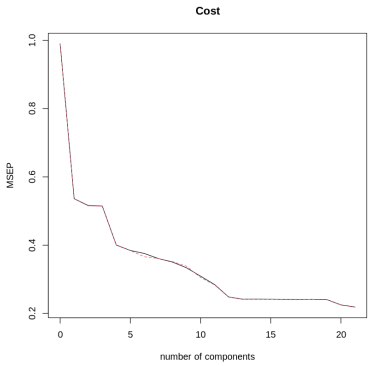


Figure 9: Cross validation RMSE of PCR model

in Figure 9, we chose the model with first 16 principal components according to cross validation results. The model retains 94.42% variance of all the features and explains 76.28% variance of the response (compared to full model explains 78.92% variance of the response). The best model returns train MSE = 0.4866 and test MSE = 0.3396. This result indicates that linear regression outperformed PCR model.

3.4 MARS Model

In order to identify potential interaction terms and discover non-linear relationship between the features and response, we performed a 10-fold cross-validation grid search on MARS (Multivariate Adaptive Regression Splines) model. The goal of this analysis is to identify the optimal combination of degree of interactions and the number of retained terms.

After a grid search of 60 hyper-parameter combinations, 3 different degree of interactions and 20 different prune parameters ranging from 2 to 30, we found that the optimal model includes second degree interactions, retains 25 terms, and selects 9 out of 21 predictors. The cross-validated

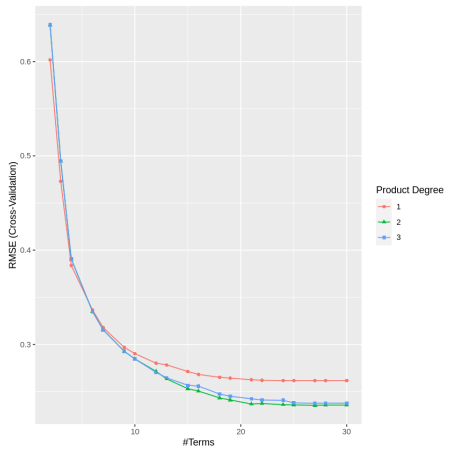


Figure 10: Cross validation RMSE of MARS model

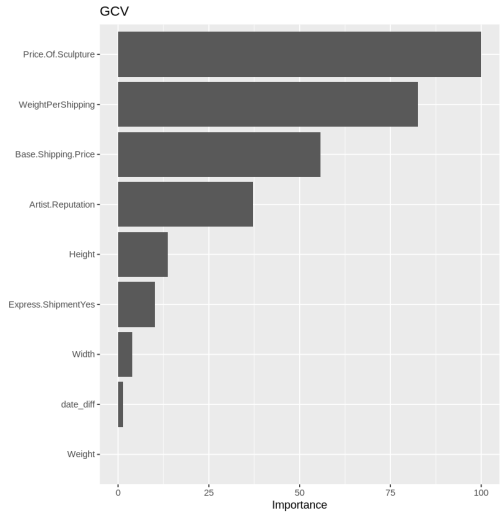


Figure 11: Variable importance based on impact to GCV

RMSE for these models are shown in Figure 10 and the optimal model's cross-validated MSE is 0.0529 with corresponding test MSE of 0.0641. The model includes interaction terms between multiple hinge functions, for example, the term *h(Artist.Reputation–0.4052) * h(Height–0.9263)* indicates an interaction effect for sculptures made by an artist with reputation rate less than 0.4052 (after scaling) and the height of sculpture is less than 0.9263 (after scaling).

Since MARS automatically includes and excludes terms during the pruning process, it also serves as a supplementary tool for feature selection in our analysis. Figure 11 shows the importance of variables based on impact to GCV. However, it's worth noticing that variable importance only measures the impact of included features and doesn't measure the impact of particular hinge functions created for a given feature, like the example stated earlier.

3.5 Tree Based Model

Tree based models have extensive usage in both classification and regression problems. Thus, in this section, we used Decision Tree with pruning, Random Forest, and XGBoost to predict shipping cost. Here we had overall better performance compared to linear model.

3.5.1 Decision Tree

We first computed the cross validation error and determined the best subtree with size of 7 as shown in Figure 12, the optimal model's cross-validated MSE is 0.1428. Then we tuned the hyperparameters with grid search over 64 different possible combinations and found that the best combination has cost complexity of 10^{-11}, tree depth of 10, and minimum number of data points in a node of 27. After tuning hyper-parameters, we observed that the test MSE was improved with the value of 0.0679, whereas the train MSE is 0.1077.

One limitation of decision trees is that they are

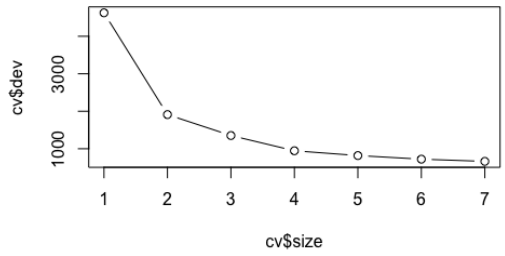


Figure 12: Cross validation MSE of Decision Tree model

strikingly unstable compared to other tree-based models, which may lead to overfitting after tuning hyper-parameters. Due to the potential of overfitting, please see the Appendix for visualizations of decision trees for reference only.

3.5.2 Random Forest

Next, we performed bagging method of decision tree, hopefully the variance could be reduced and hence achieve a lower error. Cross validation was conducted to select optimal "ntrees" and "mtry" variable in terms of test MSE. As the test error consists both bias and variance part, we chose parameters carefully rather than randomly choose large ntrees and overfit the model. The result from cross validation, as shown in Figure 13, suggests the test error continues to drop as we increase mtry and tree size and random forest achieved the best performance with mtry=10 and ntrees=1000.

3.5.3 XGBoost

Finally, we applied XGBoost method to further optimize tree-based models, because it works by fitting a new tree to the residual of the previous trees, and optimizes objective function with penalization towards tree size. The result is shown

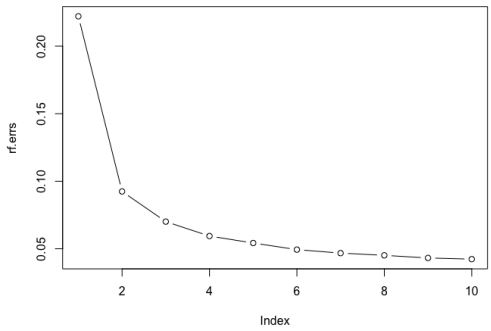


Figure 13: Test errors of mtry and treesize(*100) in Random Forest model

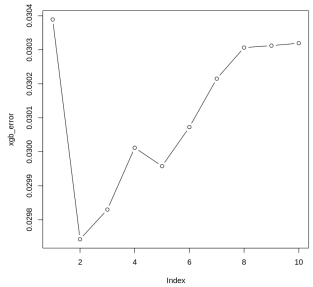


Figure 14: Test errors of XGB nrounds in XGB model

in Figure 14, the best test error we achieved is 0.03 with learning rates = 0.3, nrounds = 200, and maxdepth = 4. If learning rates were raised higher, test error would slightly increase. We also tried many other values of parameters and discovered that the results could not be greatly improved. Therefore, we believe that this reaches the limit to XGBoost.

4 Conclusion

In this project, we employed various machine learning techniques, including fitting linear and non-linear models, to predict sculpture shipping cost. Performances of models are displayed in Table 2.

The importance of features acquired from differ-

Model	Train MSE	Test MSE
Linear Regression	0.24	0.25
Ridge Regression	0.25	0.26
PCR	0.49	0.34
MARS	0.05	0.06
Decision Tree	0.11	0.07
Random Forest	0.045	0.04
XGBoost	0.001	0.03

Table 2: Summary of Model Performance.

ent feature selection methods also provides information on what factors have the largest impact on shipping cost. Specifically, for Lasso model, the optimal lambda value in 1 standard error presents fewer variables with closer prediction power.

MARS model and tree based models perform distinctly better than linear models, among which XGBoost especially has most predictive power and achieve the lowest error. Other linear or non-linear methods do not perform as well as expected. It turns out the shipping cost can be precisely predicted with machine learning model and this is useful in real business case as a reference. The predictions of our project could assist auction houses to conduct cost analysis and make decisions accordingly. Besides, buyers can run a similar test to see if he/she is overcharged.

5 Future Work

Both individual and interactive partial dependence plots (PDPs) could be performed to better understand the relation between features and the response. Because as stated in section 3.4, MARS model alone has certain limitations for a more comprehensive analysis.

We also tried boosting linear models, which takes residual to build models recursively, but the process is not explainable as expected. The stacking or ensemble methods could also be applied here with a combination of algorithm working together on each prediction, followed by averaging to reduce bias.

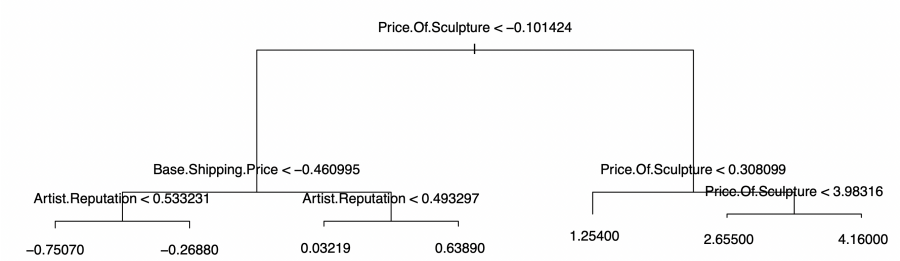
References

- http://uc-r.github.io/mars
- https://www.projectpro.io/recipes/apply-gradient-boosting-r-for-regression
- https://machinelearningmastery.com/xgboost-for-regression/
- https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data/
- https://www.tidymodels.org/start/tuning/

Appendices

A Decision Tree

Decision tree before hyper-parameter tuning:



Decision tree after hyper-parameter tuning: