FIN 550: Big Data Project

EXECUTIVE SUMMARY

Group 6: Shammy Ho, Hyun Ji Lee, Ruilin Ni, Vishwas Rao, Zhijie Jin, Akshaya Suresh, Purva Vaswani, Manvi Nagpal

Case Overview

Concisely describe the problem and your objectives as a data scientist on the team.

As a data scientist on the team, the central challenge lies in enhancing the accuracy and transparency of property valuation for the Cook County Assessor's Office (CCAO), a task complicated by the diversity of parcels across over 130 municipalities, including the City of Chicago. Historically, the CCAO's valuation process was obscure and inaccurate, prompting the need for improved methodologies. The project involves three key datasets: "predict_property_data.csv" for 10,000 properties to be assessed, "historic_property_data.csv" with information on 50,000 recently sold properties, and "codebook.csv" detailing variables. The primary goal is to predict the value of each property in "predict_property_data.csv" using data from the historic set, optimizing accuracy through data science techniques in R.

The data scientist's role encompasses model development, training, and selection, with a focus on establishing relationships between property characteristics and their values based on historical sales prices. The aim is to create a singular, comprehensive script for the entire analysis, ensuring clarity and reproducibility. The outcome will contribute to a more transparent and reliable property tax assessment process for Cook County, leveraging data science to address the intricacies of property valuation in a diverse and complex landscape.

Methodology

Describe the data approach and methodology you use. Justify your choice of methodology. You should be precise but avoid jargon. This is a document that will potentially be shared with the data science team at CCAO, so make the language is clear professional and appropriate for them.

Preprocessing

In the preprocessing phase of this data analysis, the dataset was refined by dropping variables that were missing more than 50% of data and eliminating variables not identified as predictors according to the codebook. Any negative housing prices, likely due to data entry errors, were removed. To mitigate the influence of extreme values, the data underwent winsorization. In dealing with missing data, non-missing values from the same location group were used as substitutes, with categorical and numerical variables being treated differently; the mode was used to fill gaps in categorical variables, while the mean served this purpose for numerical variables. This comprehensive preprocessing ensured the dataset was primed for accurate and meaningful analysis.

Selecting Predictors

To address the issue of multicollinearity, we conducted a careful examination of variable correlations and decided to exclude "char_beds" and "char_fbath," given their correlation values exceeded 0.75 with other variables. We retained essential variables for our model, such as the number of rooms and square footage, among others.

In attempting to employ forward selection, backward elimination, and exhaustive search for predictor selection, we encountered a significant challenge due to the presence of categorical variables like "geo_school_elem_district" and "geo_school_hs_district," each with over hundreds of levels. Recognizing the potential impact of these variables on the sales price, we needed to include dummy variables for them. However, as the number of predictors reached 600, traditional selection methods became impractical due to computational burden and time constraints.

To address this issue, lasso regression emerged as a viable alternative, efficiently handling the high-dimensional dataset. Lasso regression proved effective in the given situation, allowing us to determine the optimal lambda value (60.8) for building a predictive model.

Fitting Models

Linear Regression:

Initially, we applied linear regression using variables selected by Lasso Regression. Afterward, we converted categorical variables into factors and divided the dataset into training and test sets. Subsequently, we built the linear model and calculated the Mean Squared Error (MSE), yielding a value of 11,423,649,406.

Bagging & Boosting:

We employed identified by Lasso Regression with correctly transformed factor variables and partitioned the data into training and testing sets. Ultimately, bagging and boosting did not yield results within a reasonable amount of time for decreased number of variables or decreased level of factors.

We utilized predictors identified through Lasso Regression, ensuring proper transformation of factor variables, and subsequently divided the data into distinct training and testing sets. However, the application of bagging and boosting techniques did not produce results within a reasonable timeframe for decreased number of variables or decreased level of factors.

Regression Tree:

Applying a set.seed(70) for data partition, we used variables identified by Lasso Regression and set a complexity parameter of 0.01 for the regression tree, resulting in a Mean Squared Error (MSE) of 15,503,818,679. Through pruning at a complexity parameter of 0.001 and employing 5-fold cross-validation, the MSE was reduced to 12,504,983,754.

Random Forest:

Utilizing set.seed(1) as the random seed, we employed variables identified by Lasso Regression, ensuring uniform factor levels in both test and training sets. Applying the random forest algorithm to the training set, with 'sale_price' as the response variable and 'mtry' restricted to 4 - which means at each split, the algorithm randomly selects four variables from the dataset for making the split - the model yielded an MSR of 9,293,780,067 and an MSE of 8,716,355,447, the lowest among all generated models. Consequently, we selected the Random Forest model for predictions, culminating in the creation of the *Property Assessment Prediction* CSV file with PID and assessed_value columns.

Conclusion

Describe your results, including summary statistics (e.g., min, max, mean, and quartiles) of the distribution of assessed property values. Describe your data file which reports the assessed property values you have generated.

The results of the assessment of property values reveal a diverse distribution, as evidenced by the summary statistics. The assessed values span a range from a minimum of $18,320 to a maximum of $1,524,386, illustrating the diversity within the dataset. The distribution is characterized by a median value of $250,671, indicating that half of the assessed values fall below this point. The mean assessed value is $315,411, reflecting the average value across all properties. Additionally, the first quartile (25th percentile) is at $154,312, and the third quartile (75th percentile) stands at $379,366, providing insights into the spread and variability of values in the lower and upper segments of the distribution.

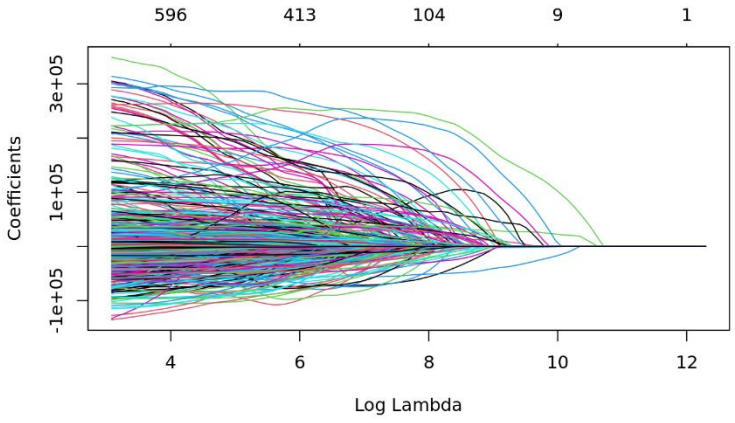
Appendix

Include tables and plots here. Enumerate each table and plot, give each a descriptive title, and make sure elements are labeled clearly. Tables and plots should correspond to output from your code.

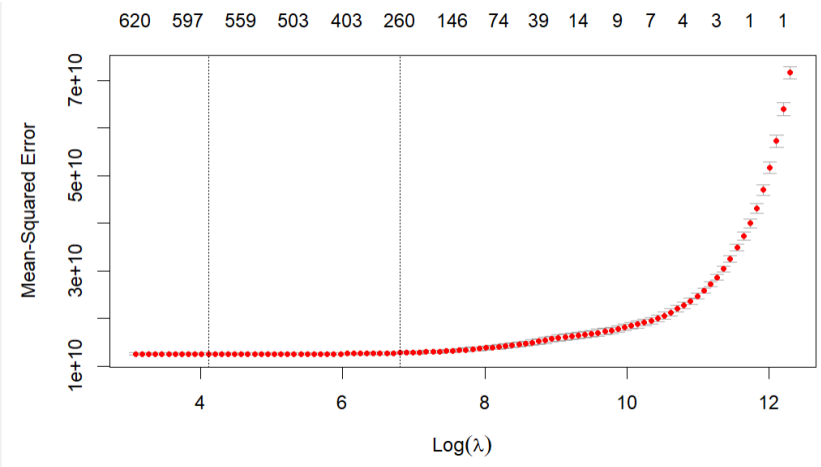
Correlation values between most highly correlated variable pairs

```
Correlation between meta_certified_est_bldg and sale_price : 0.8225603
Correlation between sale_price and meta_certified_est_bldg : 0.8225603
Correlation between char_beds and char_rooms : 0.9003237
Correlation between char_fbath and char_rooms : 0.7577657
Correlation between char_bldg_sf and char_rooms : 0.7856339
Correlation between char_rooms and char_beds : 0.9003237
Correlation between char_rooms and char_fbath : 0.7577657
Correlation between char_bldg_sf and char_fbath : 0.809186
Correlation between char_rooms and char_bldg_sf : 0.7856339
Correlation between char_fbath and char_bldg_sf : 0.809186
```

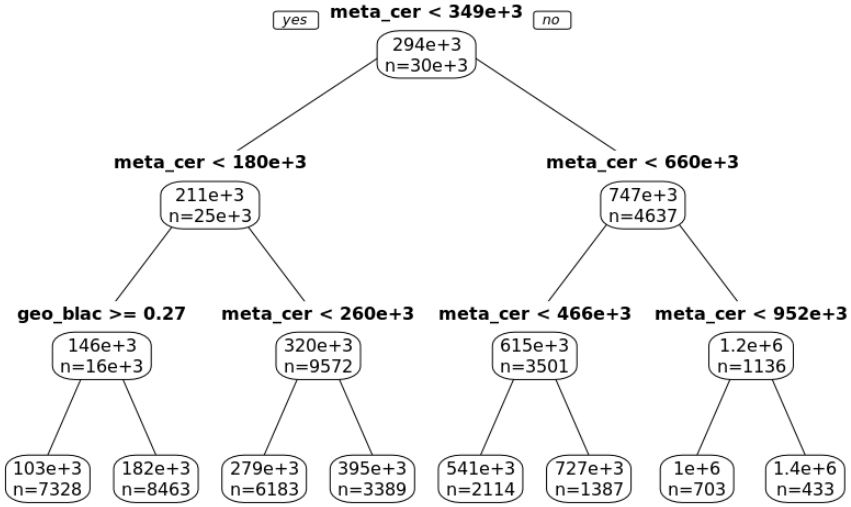
Coefficients on log of lambda values



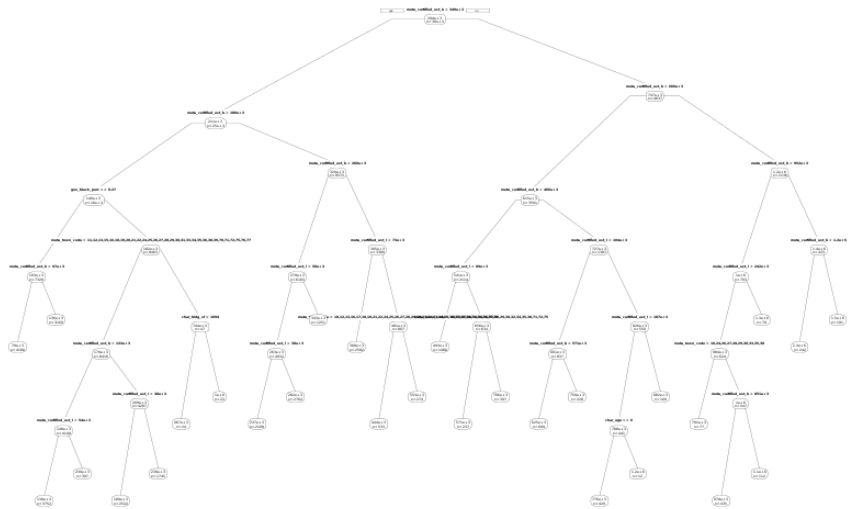
Lambda value that corresponds to lowest cross-validated MSE



Regression tree with cp = 0.01



Pruned regression tree with cp = 0.001, xval = 5



Random Forest model's MSE & MSR

```
Call:
 randomForest(formula = sale_price ~ ., data = train.df, mtry = 4)
               Type of random forest: regression
                     Number of trees: 500
No. of variables tried at each split: 4

          Mean of squared residuals: 9293780067
                    % Var explained: 87.11
```

```r
# Make predictions on the test data
predictions <- predict(rf, newdata = test.df)

# Calculate the Mean Squared Error
mse <- mean((test.df$sale_price - predictions)^2)

# Print the MSE
print(mse)
```

```
[1] 8716355447
```

Summary statistics of assessed property data

```
 assessed_value
 Min.   :  18320
 1st Qu.: 154312
 Median : 250671
 Mean   : 315411
 3rd Qu.: 379366
 Max.   :1524386
```