**ASSIGNMENT 4**

**INVESTING IN NASHVILLE**

Trilok Palla

Northeastern University [College of Professional Studies](https://cps.northeastern.edu/)

**ALY 6020- Predictive Analytics**

Justin Grosz

**INTRODUCTION:**

A solid investment has a high likelihood of success or return on your investment. A lot of websites and ad campaigns have been developed to target potential customers to provide investment opportunities. In general terms, Real Estate Asset Management and the Nashville housing market are expanding.

**About Decision Trees:**

Decision trees are a non-parametric supervised learning approach that can be used for both regression and classification applications. It has an internal root node, branches, internal nodes, and leaf nodes in a hierarchical tree structure. A tree can be "learned" by dividing the source set into subgroups based on an attribute value test. This method is repeated recursively on each derived subset, which is known as recursive partitioning. When the subset at a node all has the same value of the target variable, or when splitting no longer adds value to the predictions, the recursion is finished.

**About Random Forest Trees:**

The output of multiple decision trees is combined by Random Forest to produce a single result. Its ease of use and flexibility, as it handles both classification and regression problems, has fueled its adoption.

**About Gradient Boosting Method:**

A gradient boosting classifier is a collection of machine learning techniques that merge numerous weaker models into a powerful big one with highly predictive output. Such models are popular due to their ability to properly classify datasets. Decision trees are commonly used in the construction of gradient-boosting classifier models.

**About the data:**

This dataset consists of 22651 data points with 26 features. These 26 features consist of 15 object-type data and the rest 11 are numeric. With the help of these features, we are going to train our data using Logistic Regression, Decision Tree, Random Forest, and Gradient Boosting methods.

**Libraries Used:**

* Numpy
* Pandas
* Matplotlib
* Seaborn
* Sklearn methods
* Requests

**Part 1:**

**BASIC DATA CLEANING:**

* The first step here is to check if the data consists of any NULL values in it. Several columns have NULL values on them:

1. 22651 NULL values found in the Suite/Condo Number column.
2. 108 NULL values found in half bath.
3. 3 NULL values are found in the Bedroom column.
4. 2 found in property city and property address.
5. 1 NULL value found in full bath, finished area, and foundation type.

* These missing data points have been filled with mean and median values. In this manner, we can avoid removing these individuals from the dataset and also not skewing the column values. Although not ideal, this approach enables you to include values that do not affect the entire dataset because the average remains constant regardless of how many averages are added.
* Median Value is considered for ‘Half Bath’, ‘Full Bath’, ‘Bedrooms’, and ‘Foundation Type’ As these values are usually whole numbers, taking the middle value (median) would be ideal.
* Mean Value is considered for ‘Finished Area’, and ‘Building Area’. Since these values can be till decimals also, direct average(Mean) can be taken.
* The Column ‘Suite/Condo #’ is dropped as all the columns were found empty.

**DATA VISUALIZATIONS:**

In this step, a few pie charts, bar plots, and box plots of object variables have been depicted to get a better understanding of the data in hand.

Given below are the observations that can be seen from the above plots:

* From the Pie chart of ‘Sold as Vacant’, we can see that almost (99%) of investors have said ‘No’.
* The majority of buildings have exterior walls made of brick (51%).
* Most of the buildings owned are by single families(94%).
* Most of the Houses have either 1 or 2 baths.
* Almost half of the data consists of buildings with 3 bedrooms.
* 75% of the houses are overpriced.

The plot Images have been attached in the appendix.

**DATA PREPROCESSING:**

A few categorical columns have been converted into numerical (‘Foundation type’, ‘Sale Price Compared To Value’, ‘Land Use’,’ Sold As Vacant’,’ Multiple Parcels involved in Sale’, ‘Exterior Wall’, ‘Grade’) to run methods smoothly. In this, we have split the training and test data set in an 80 to 20 ratio with ‘Sale Price Compared To Value’ as the dependent variable under data frame ‘y’ and the rest of the variables under a different data frame ‘x’.

**MODEL BUILDING AND COMPARING:**

The data now which we currently have is having 22649 data points with 25 features ready to be built.

The Logistic regression, Decision Trees, Random Forest, and Gradient Boosting here can be built using both “sklearn” and “statsmodels”. But with “sklearn” we will be able to get a better summary of our classification models.

**Part 2:**

**Model Building using Logistic Regression:**

A screenshot of a computer

Description automatically generated with medium confidence

From the **first round of model building**, given below are the results obtained,

A screenshot of a computer

Description automatically generated with low confidence

Observations:

* Many variables are found to have p-value greater than 0.05.
* Model accuracy of 76.1% is observed.
* Columns **Foundation Type(P- 0.026), and Acreage (P- 0.034)** are found to have p-values close to 0.05.
* The model is having a precision of 60%.
* The model is found to have 1068 false positives.

**Part 3:**

**Model Building Using Decision Trees:**

Given below is the decision tree obtained,

Qr code

Description automatically generated

Given below are the model scores,

A screenshot of a computer

Description automatically generated with medium confidence

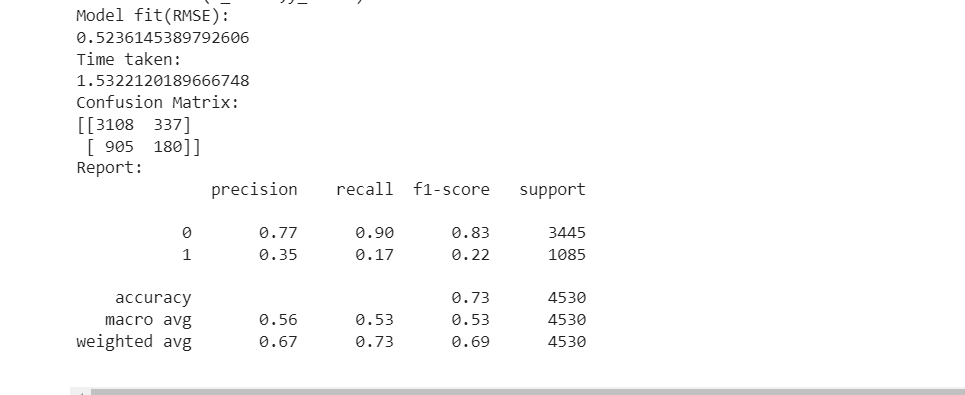
Observations:

* Good development in precession has been observed with **76.7%.**
* The model has a precession of **80%**.
* The most important variables (feature importance) are, ‘Sold as Vacant’, ‘Building Value’, and ‘Year Built’.
* Since ‘Build as Vacant’ almost has biased data consisting of 99% of ‘No’ Values, the most important feature in this dataset would be **‘Building Value.**
* **Lower false positives** than the previous model from the **Confusion Matrix.**

**Part 4:**

**Model Building Using Random Forest:**

Given below are the scores obtained by using Random Forest Classifier:



Observations:

* A model accuracy of **72.5%** has been obtained.
* A precision of only **34%** is observed.

**Part 5:**

**Model Building Using Gradient Boosting Classifier:**

Given below are the observations obtained from Gradient Boosting Classifier,

A screenshot of a computer

Description automatically generated with low confidence

Observations:

* A model accuracy of **76.9%** has been obtained.
* A precision of **72.5%** is obtained.

**Part 6:**

**Model Comparison:**

Given below is the table for comparing several models which have been evaluated,

Table

Description automatically generated

Final Observations:

* Gradient Boosting and Decision Tree methods are found to have the highest accuracy of almost around **76.8%.**
* Both Gradient boosting and Decision Tree methods are having similar low RMSE scores of **0.48.**
* Both of these models have been a great fit for the data set. But one major factor which can be observed here is the time taken to run the model.
* The decision Tree method is having a runtime of only **0.0534**, whereas the Gradient Boosting method is having a runtime of **6.74** seconds. This means that each time we run the gradient boosting algorithm, we can run more than 100 times of decision tree method.
* Also, the decision tree is having the highest precision among all the 4 methods with **80%.**

**CONCLUSION:**

So, after evaluating all 4 models on the given dataset, it is clear that the Decision Tress method is having an upper hand, with high precision, good accuracy scores, and a very low run time. In the real estate and investment planning business, there will be a lot of such data to deal with, and with the increasing competition among corporate companies, I believe our model needs to have more precision and lower runtime. As such even though the Gradient boosting model is having a little higher accuracy and better rmse, considering several constraints in the picture, the **ideal model recommendation** would be **Decision Tress.**

3 variables have been observed with the highest feature importance i.e **‘Sold as Vacant’**, **‘Building Value’, and ‘Year Built’.** The data obtained from us is biased in the case of the most important variable **‘Sold as Vacant’** with 99% of ‘No’. It can be observed from the tree that only 104(96) samples are having ‘Yes’, which then are depending on ‘Year Built’.

So from the above inference, it is clear that **‘Building Value’, and’ Year Built’** is the most important feature and directly relate to the target variable ‘**Sale Price Compared To Value’**.

**Business Recommendations:**

The most significant recommendations to the customer here would be the 2 given below,

🡺Showing underpriced buildings for investment opportunities: From the decision tree chart observed, 2 conditions can accompany the customer to buy an underpriced building,

* + When the building value is less than **309,950$** and is built before **2015**, the customer can look to invest in such a place.
  + For vacant buildings built after **1962** with 2 and more bedrooms, the customer can be directed to invest.

🡺Providing caution for overpriced buildings, as it can be observed from the decision tree that buildings with lower values must be carefully evaluated as these houses can be sold for prices more than they are worth. Customers would often be lured to such traps (low-priced buildings).

**Appendix:**

Chart, pie chart

Description automatically generatedChart, pie chart

Description automatically generated

*Fig1: A Piechart plot of Sold as Vacant Attribute. Fig2: A Piechart plot Exterior Wall Type*

Chart, pie chart

Description automatically generatedChart, pie chart

Description automatically generated

*Fig3: A Piechart of building grade type Fig4: A pie chart of usage of land*

*Chart, histogram

Description automatically generated*

*Fig5: Distribution of baths bar graph*

*Chart, histogram

Description automatically generated*

*Fig6: Distribution of bedrooms bar graph*

*Chart, histogram

Description automatically generated*

*Fig7: Property build a distribution bar graph*

*Shape

Description automatically generated*

*Fig8: Land value distribution bar graph*

*Shape

Description automatically generated*

*Fig9: Acerage Distribution bar graph*

*Chart, pie chart

Description automatically generated*

*Fig10: Piechart of Sale Compared to Value*

*A picture containing text, clipart

Description automatically generated*

*Fig11: Correlation HeatMap*

**References:**

<https://www.geeksforgeeks.org/decision-tree-introduction-example/>

<https://medium.com/geekculture/gradient-boosting-classifier-f7a6834979d8>

Thank you,

Trilok Palla