Investing in Nashville

ALY6020: Predictive Analytics

Module 4 Assignment

Submitted by:

Naina Gupta

Submitted on:

6/18/2023

Introduction

This report is a summary of a comprehensive data analysis for the company looking for a large investment into the growing Nashville area. The acquired dataset pertains to information about recent sales. The project requires the building of an appropriate model to help the company accurately predict the best sale value deals during their visit in the following week. The emphasis will be given on the sale price compared to property value, to enable identification of the kind of properties that are being overvalued or undervalued. The goal of the project is to help the company look out for those factors which are crucial for finding the best house deal in Nashville.

In this project, with 'Sale Price Compared to Value' as the target variable, proper data cleansing techniques will be used to ensure highest quality data for building 4 models- Logistic Regression model, Decision Tree model, Random Forest model and Grade Boosting model. These models will be instrumental in accurately predicting the important factors for the best house deals and will subsequently help us learn about property valuation in the Nashville area. Finally, the project will compare the accuracy of all the models using at least two metrics and will narrow down the business problem into a solution based on the best model and the most significant variables.

Data cleaning

The original Nashville housing dataset consists of 22652 records spread under 26 columns. The cleaning of this dataset begins by first filtering the dataset for records for 'Nashville' city because that is the area we are particularly interested in this project. Next, we remove unwanted columns like 'City', 'Property city', 'State', 'Parcel ID', 'Legal Reference' and empty columns such as 'Suite/Condo #'. After checking for any null values, we find a total of 90 records under various columns with null values. Since this accounts for approximately less than 0.5% of the 18010 records, we choose to drop these records. Next, we convert the column 'Year Built' into a column showing the number of years old the each property is with respect to the current year so that we get a more specific variable for subsequent analysis. We also extract the year out of datetime column 'Sale year' to create cleaner time series plot and to be able to include this variable in our models.

Finally, all the unnecessary variables are dropped and as a result, we are left with 17 variables and 17926 records for the predictive analysis.

Exploratory Data Analysis and Visualization

The visualizations below give meaningful insight about the property valuation in Nashville area as entailed by this dataset:

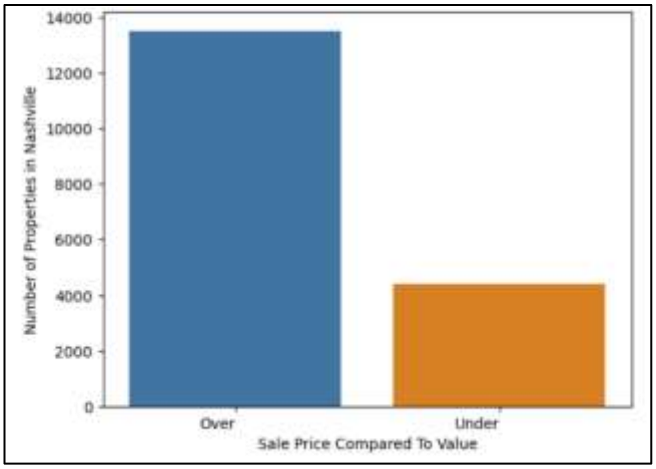


Figure 1: Bar plot for Number of Overvalued and Undervalued properties in Nashville:

The figure clearly shows that the number of properties in Nashville area having a sale price more than their actual valuation are far more than the undervalued ones. Nearly 14,000 properties are overvalued whereas only a little over 4000 properties are undervalued.

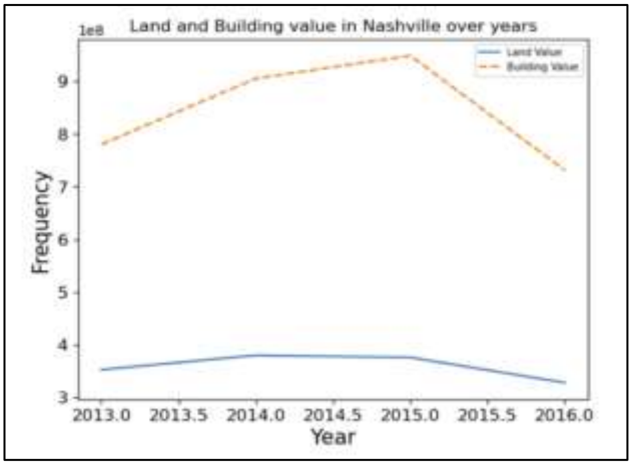


Figure 2: Multiple Time series line plot for Land and Building value in Nashville:

The data for Nashville properties are recorded from 2013 to 2016. It can be observed that the building value of properties are much higher than the Land value which is obvious. However, the building values have seen a major dip around mid-2015 and has been consistently falling sharply since then.

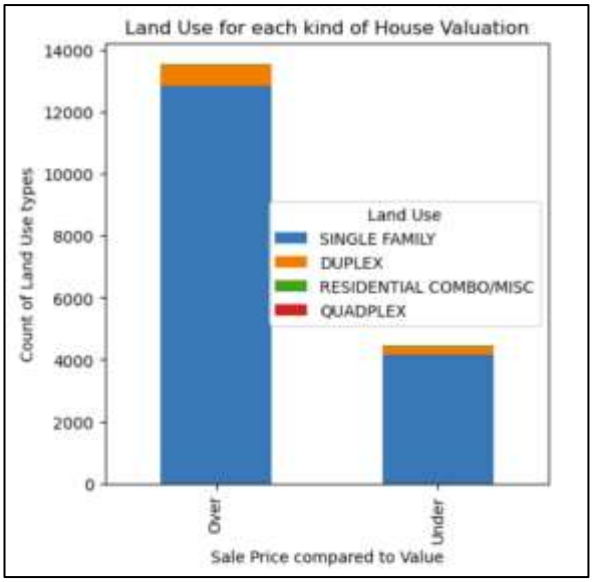


Figure 3: Stacked Bar chart for Land Use of Overvalued and Undervalued properties:

It is observed that for both overvalued and undervalued properties, the majority of properties in the Nashville area are Single family houses with negligibly few Duplexes as well.

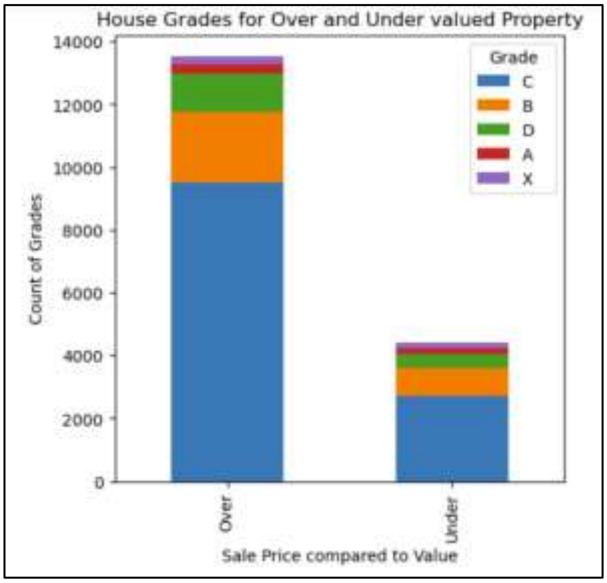


Figure 6: Stacked Bar chart for Grades of Overvalued and Undervalued Properties:

It is observed that majority of the properties in Nashville area, whether overvalued or undervalued, belong to Grade C, followed by a few from Grade B. The other grades hold a negligible constitution within the area.

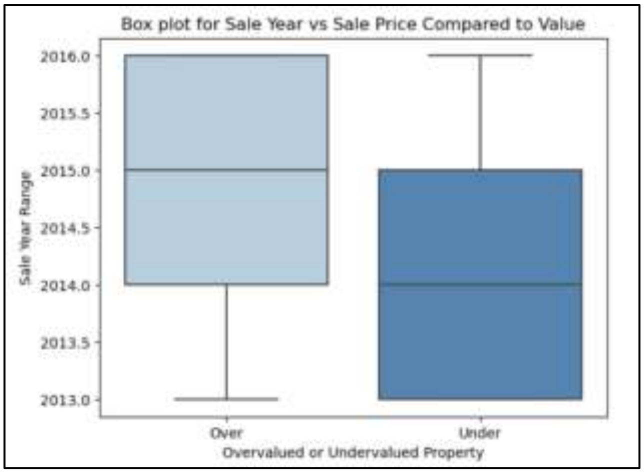
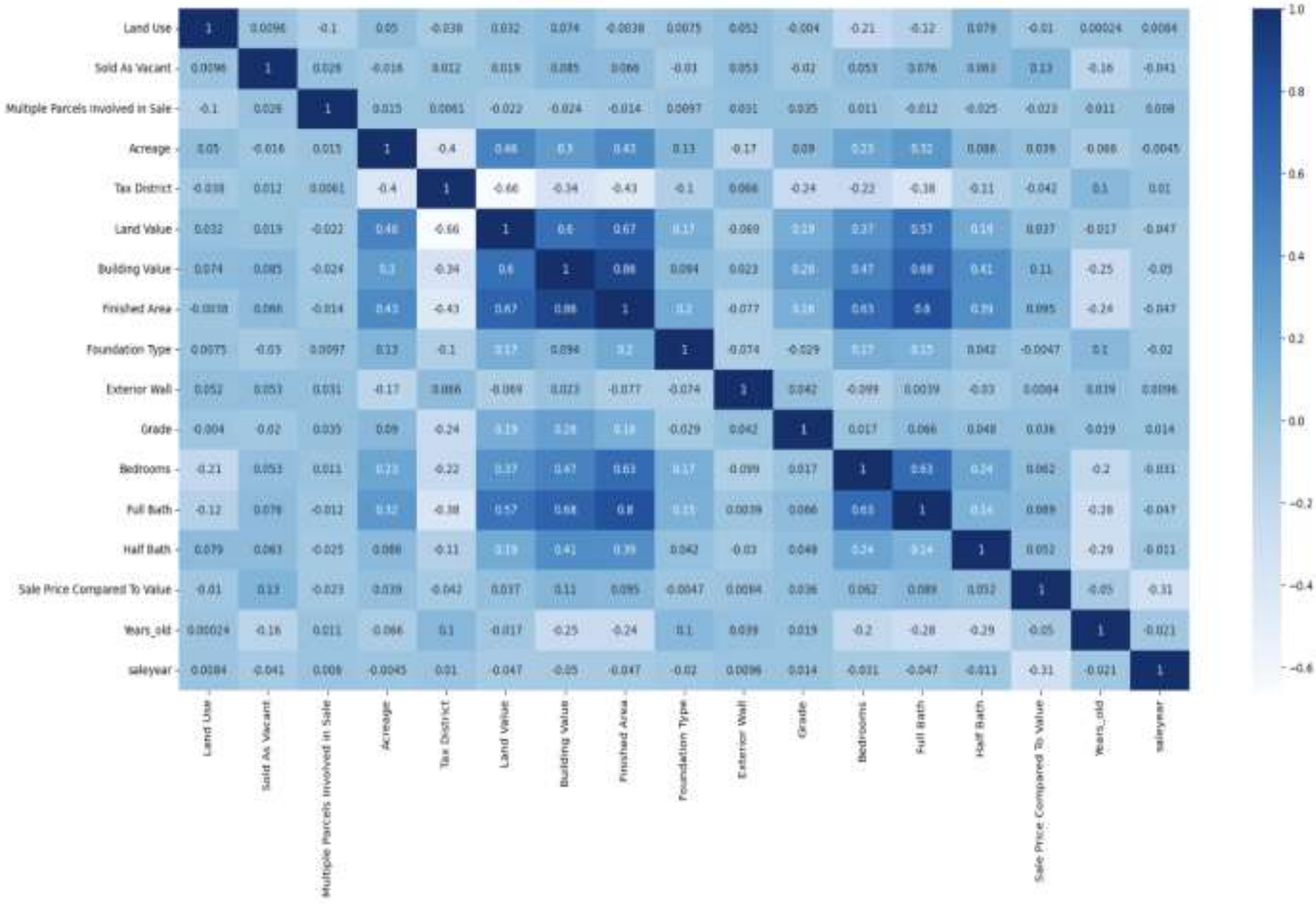


Figure 4: Boxplot of Sale year for Overvalued and Undervalued properties:

The boxplot shows the distribution of years of sale for both overvalued and undervalued properties separately. Evidently, the Overvalued ones in Nashville are typically the ones being sold most recently, roughly between 2014 to 2016. However, the undervalued ones are those sold between 2013 to 2015. Houses sold in 2015 are overvalued compared to the houses sold in 2014.

Figure 7: Correlation Matrix of all the relevant variables



The above matrix shows that Building value and Full Bath have a significant positive correlation with Finished Area. Other than that, there is no significant correlation observed between all the other independent variables. In the next section for model building, our target variable is 'Sale Price compared to Value' and the correlation plot shows no significant correlation between the target and all the other variables included in the dataset.

Prediction Models

The ‘Sale Price compared to Value’ is the targeted categorical variable, where class ‘0’ means ‘Overvalued Property’ and class ‘1’ means ‘Undervalued property’. Since no significant correlation was found between the target and other variables, we include all the 16 variables to predict the target. For model training, the entire dataset is split into train and test datasets by 80:20 ratio. This ratio was chosen in lieu of a modest data size and to avoid overfitting of the model. The remaining test data will be used to evaluate the performance of the trained model.

Logistic Regression Model

Under this section, we construct a logistic regression model with null hypothesis stating that none of the predictor variables have a statistically significant relationship with the response variable, ‘Sale price compared to Value’. The alternative hypothesis assumes the opposite, that is, variables have a significant impact on predicting the odds of a property being undervalued or overvalued.

Model output: The model shows that since the intercept is equal to approximately 1438.052 with a positive sign, when none of the 8 variables are in play, the probability of having an undervalued property is likely to be more than 50%. Moreover, the significant variables that are most likely to affect this prediction of our categorical target variable are- ‘Sold as Vacant’, ‘Multiple Parcels Involved in Sale’, ‘Acreage’, ‘Tax District’, ‘Land Value’, ‘Building Value’ and ‘Sale year’.

Taking the exponential of these coefficients to represent them as the odds of having the outcome of an undervalued property, we will discuss those significant variables which have a larger coefficient and hence, a larger impact on the house price compared to valuation. If the property is sold as vacant, a unit increase in this feature of a house, increases the odds of the property being undervalued by 28.44 times. Similarly, for a unit increase in Acreage, the chances of a property being undervalued increases by approximately 1.15 times. For an incremental increase in year of sale, the odds of the house being undervalued will decrease by 0.49, since the coefficient is negative. Thus, properties sold in more recent years are more likely to be overvalued. If multiple parcels are involved in the sale of a property, an incremental increase in the count of such properties will decrease the odds of getting an undervalued house by 0.56 times.

Model Performance: When all the variables are included in the model to predict the chances of overvaluation or undervaluation of property in Nashville area, the accuracy of the model is roughly 76%. However, from our exploratory analysis, we know that the actual data is highly imbalanced because most of the properties recorded in the dataset are overvalued in the Nashville area. This implies that the dataset is skewed towards one class of the target variable more than the other. As a result, there is a good chance that despite this accuracy level, the model may have failed to accurately predict the less represented class of ‘Undervaluation’ or ‘1’. Thus, accuracy as a metric could be misleading, and we check other metrics. Precision and recall are two metrics used to assess the quality of positive predictions made by the model.

‘Precision’ tells us that out of all the predictions, 77% responses were actually accurate for ‘overvalued’ class, whereas ‘Recall’ tells us that out of all the responses under each class, 98% of them were correctly predicted. Here, the precision and recall values for Overvalued properties are far better than that for undervalued properties. Thus, we can say that this model will perform better at accurately predicting the outcome of having an overvalued property. The F1 score confirms this result.

Decision Tree Model

The goal for a decision tree model is to use the earlier decisions to create a tree-like stepwise movement from one feature to another to predict the target class (classification). Here, our target variable is 'Sale Price compared to Value' using different feature subsets and decision rules at different stages of classification. The depth of the decision tree is programmed at 3 to increase its interpretability and reduce the chances of model-overfitting as far as possible.

Model Output: To estimate feature importance, we use the Gini Index, the probability of misclassifying a randomly chosen record in a sample set. In our model, the decision tree (see appendix) begins with Sale year as the root node. Since this is the first step, all the 14340 records are included in the sample since it constitutes 80% of the data reserved for model training. For properties not having Sale year less than or equal to mid-2015, are more likely to be overvalued (classified as '0'), with only 12.8% chances of being misclassified in a sample of 3295.

The most important predictor variables as per the model output are year of sale, age of the property with respect to current year and the building value, in that order.

Model Performance: When all the variables are included in the model, the accuracy of the model is roughly 77%. Since accuracy as a metric could be misleading for imbalanced data, Precision and recall will be used to assess the quality of positive predictions made by the model. As per 'Precision' out of all the predictions, 98% of responses were actually accurate for the 'Undervalued' class, whereas 'Recall' tells us that out of all the responses under 'Overvalued' class, 100% of them were correctly predicted. These precision and recall values favor opposite classes. Thus, we check the F1 score, which is the harmonic mean of precision and recall. With a 0.87 value, it confirms that the model is better at predicting the 'Overvalued' class of properties in Nashville area.

Model comparison so far: The accuracy level for decision tree has increased by 1% over the logistic regression model. In terms of precision, logistic model indicates better accuracy in predicting 'Undervalued properties' while decision tree does so for 'Overvalued' properties. In terms of recall, both show a higher prediction accuracy for 'Overvalued' houses.

Random Forest Model

Using random forest classification, multiple decision trees are created from different random subsets of the train dataset, at a depth of 3 features. Predictions for the 'Sale price compared to value' are made by calculating the prediction for each decision tree, then taking the most popular result. The other hyperparameter 'n_estimators', that is, the number of decision trees in the forest, has been tuned to 100 since it improves the performance of the model but at the same time increases the computational cost of training and predicting.

Model Output: To estimate feature importance, we use the Gini Index, the probability of misclassifying a randomly chosen record in a sample set. In our random forest model (see appendix), if the sale year is not less than or equal to 2014.5, Acreage is less than or equal to 0.975, and Finished Area is less than or equal to 4037.46, then the property is predicted to be overvalued over the sale price with roughly 22% chances of a misclassification in a sample of 4144. The most important predictor variables as per the model output are year of sale, vacancy status of the property being sold, age of the property with respect to current year, the building value and Land value, in that order.

Model Performance: When all the variables are included in the model, the accuracy of the model is roughly 76.44%. Since accuracy as a metric could be misleading for imbalanced data, Precision and recall will be used to assess the quality of positive predictions made by the model. As per 'Precision' out of all the predictions, 95% of responses were actually accurate for the 'Undervalued' class, and 'Recall' tells us that out of all the responses under 'Overvalued' class, 100% of them were correctly predicted. These precision and recall values favor opposite classes. The F1 score, being the harmonic mean of the two metrics, confirms that the model is better at predicting the 'Overvalued' class of properties in Nashville area.

Model comparison so far: The predictive accuracy level for decision tree has improved by 1% over logistic regression and random forest model. In terms of recall value, the decision tree and random forest models improve to 100% in predicting 'Overvalued properties' while Logistic regression model does so for 'Undervalued' properties. In terms of precision both decision tree and random forest model show a higher prediction capability of up to 98% for 'undervalued' houses, whereas logistic regression model show higher prediction capability for 'Overvalued' properties. For comparison, we will have to use F1 scores for all the three models and find that these are more than or equal to 86% for 'Overvalued' class.

Gradient Boosting Model

In the previous random forest model, the results of decision trees are aggregated at the end of the process to give the popular result. However, in Gradient boosting model, the decision trees are aggregated one after the other such that each new tree builds on the deficiencies of the previous trees and boosts the result. Gradient boosting usually performs better than random forests but since they're prone to overfitting, we have tuned its depth to 3 and number of trees to 100, like the previous model.

Model Output: To estimate feature importance, we use the Friedman Mean of Squared Errors. As per our Gradient Boosting model (see appendix), if the sale year of the property is not less than or equal to 2015.5 and the property is not less than or equal to 7.5 years old, then the property is predicted to be overvalued over the sale price with roughly 0.143 average squared deviation between the observed and the predicted value. The most important predictor variables as per the model output are year of sale, age of the property with respect to current year, Land value and the building value, in that order.

Model Performance: When all the variables are included in the model, the accuracy of the model is roughly 78%. Since accuracy as a metric could be misleading for imbalanced data, Precision and recall will be used to assess the quality of positive predictions made by the model. As per 'Precision' out of all the predictions, 79% of responses were actually accurate for the 'Overvalued' class, and 'Recall' tells us that out of all the responses under 'Overvalued' class, 97% of them were correctly predicted. These precision and recall values favor the same class and the F1 score, further confirms that the model is better at predicting the 'Overvalued' class of properties in Nashville area.

Model Comparison (all models)

As per the accuracy levels, the Gradient Boosting model performs the best with 78% overall accuracy in predicting the overvaluation or undervaluation class of a property in Nashville. This is a reliable model as it combines several weak learner decision trees into strong learners, such that each new model is trained to minimize the mean squared error. However, since it is an unbalanced dataset, we confirm the validity using another metric. ‘Precision’ and ‘Recall’ alone cannot be used for comparing all the 4 models because they favor opposite classes in case of decision tree model. Thus, we move to F1 score, which is a harmonic mean of Precision and Recall values under each class. According to the F1 score, all the models show a higher predictive accuracy for the ‘Overvalued’ property class. On comparison, we conclude that while, both Decision tree model and Gradient boosting model show equal F1 score of 0.87 under the ‘Overvalued’ or ‘0’ class of property, the Gradient boosting model is the best fit for predictive modeling of target variable in Nashville since its accuracy level is higher too.

The feature importance charts (see appendix) for decision tree, random forest and gradient boosting model, give only the order of importance of all the variables in predicting the target variable. However, the logistic regression model gives the statistical significance of variables by means of p-value and the importance of variables by means of their coefficients.

Conclusion

It is concluded that based on this dataset, all the 4 models are trained to make more reliably accurate predictions for outcomes pertaining to a response of ‘0’ or ‘Overvalued’ class of properties in Nashville area. However, the Gradient Boosting model is an optimized model as it increases the ‘Accuracy’ as well as the ‘F1’ score for class ‘0’, implying that for all the instances of this class, there is an 87% accurate prediction rate. To predict the categorical target variable ‘Sale Price compared to Value’, the two most important features given by the Gradient boosting model while also considered statistically significant by the logistic regression model are year of sale and the building value.

Generally, for making a huge investment into a growing city like Nashville, undervalued properties should be favored over overvalued ones, as the company can buy low and sell high. If the city continues to grow at the same pace, the undervalued property can give promising returns. However, buying an overvalued property doesn't have this advantage, as the sale price would eventually return to its intrinsic value, which is lower. For an incremental increase in year of sale, the odds of the house being undervalued decrease by 0.49. Thus, properties sold in more recent years are more likely to be overvalued. It is recommended that the company should focus on investing their resources in properties that have not been sold in a while. This is because the properties sold most recently could be witnessing a real estate bubble due to many possible reasons such as temporary increase in demand, limited supply, sub-normal interest rates and so on. Since an incremental increase in the building value makes the odds of an undervalued Nashville property increase by 1 time the odds, the company must focus on properties with high or increasing building values. To identify such undervalued properties, the company must look out for properties with a promising location but are hard to sell, have low transaction volumes and have motivated sellers.

References

Python, Real. “Logistic Regression in Python – Real Python.” *Realpython.com*, realpython.com/logistic-regression-python/

“Guide to Accuracy, Precision, and Recall.” *Mage*, www.mage.ai/blog/definitive-guide-to-accuracy-precision-recall-for-product-developers.

Hoare, Jake. “Gradient Boosting Explained - the Coolest Kid on the Machine Learning Block.” *Displayr*, 5 June 2017, www.displayr.com/gradient-boosting-the-coolest-kid-on-the-machine-learning-block/.

scikit learn. “1.10. Decision Trees — Scikit-Learn 0.22 Documentation.” *Scikit-Learn.org*, 2009, scikit-learn.org/stable/modules/tree.html.

Shafi, Adam. “Sklearn Random Forest Classifiers in Python Tutorial.” *Www.datacamp.com*, Feb. 2023, www.datacamp.com/tutorial/random-forests-classifier-python.

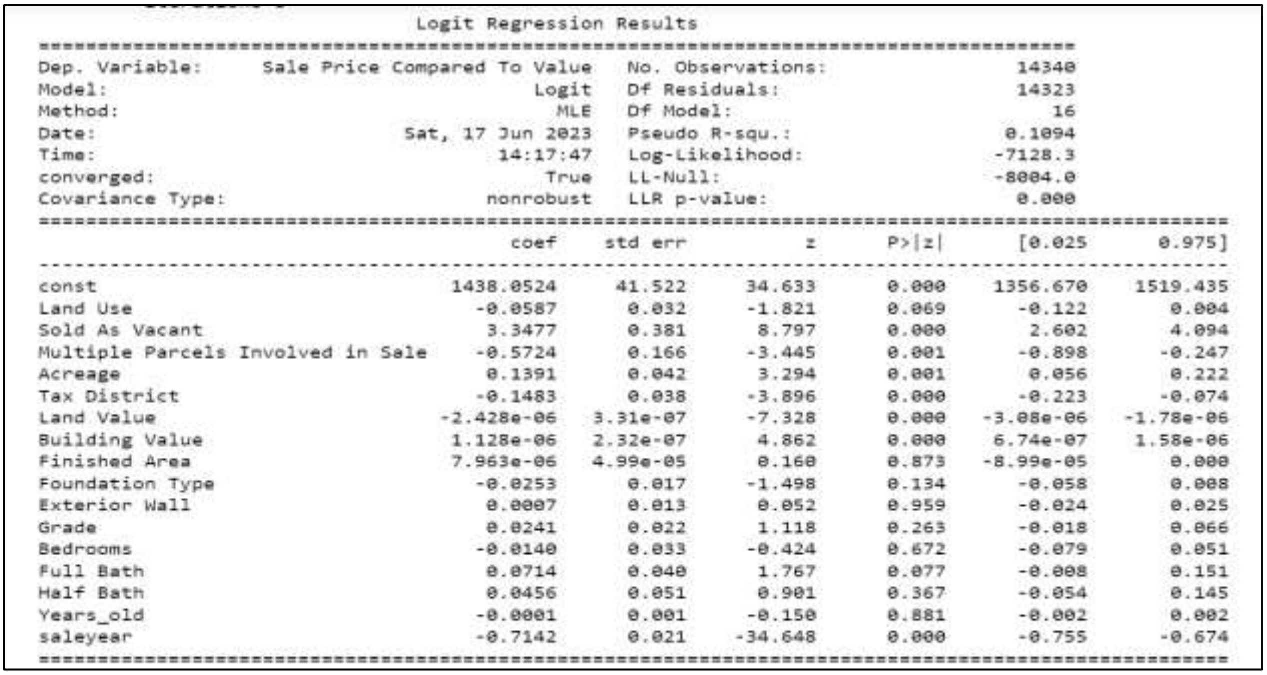
“Overvalued.” *Corporate Finance Institute*, corporatefinanceinstitute.com/resources/capital-markets/overvalued/.

Brownlee, J. (2021). How to Develop a Gradient Boosting Machine Ensemble in Python. *MachineLearningMastery.com*. https://machinelearningmastery.com/gradient-boosting-machine-ensemble-in-python/

Appendix

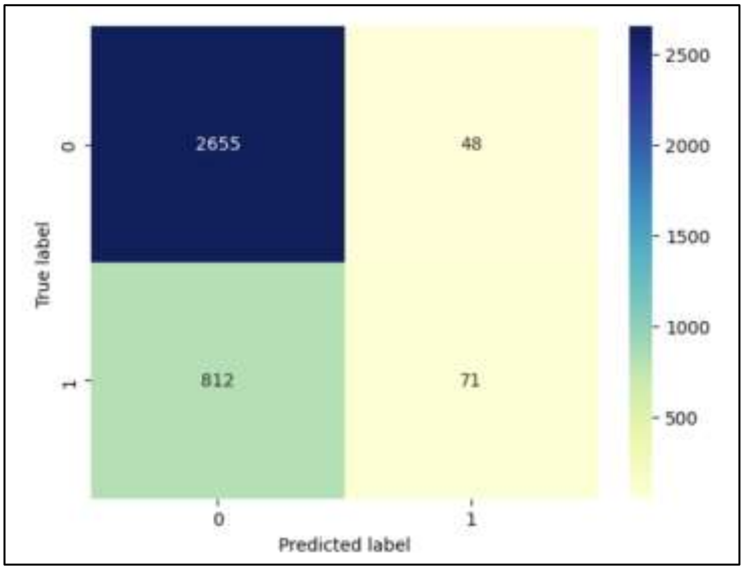
Logistic Regression Model

Model Summary:

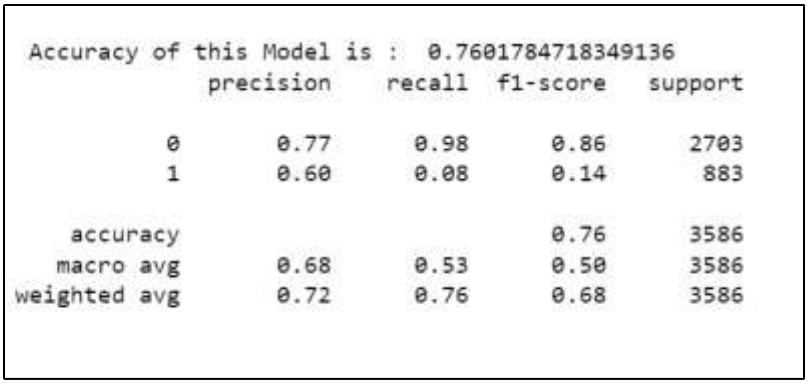


```
                           Logit Regression Results
==============================================================================
Dep. Variable:     Sale Price Compared To Value   No. Observations:                14340
Model:                                    Logit   Df Residuals:                    14323
Method:                                     MLE   Df Model:                           16
Date:                          Sat, 17 Jun 2023   Pseudo R-squ.:                  0.1094
Time:                                  14:17:47   Log-Likelihood:                -7128.3
converged:                                 True   LL-Null:                       -8004.0
Covariance Type:                      nonrobust   LLR p-value:                     0.000
=====================================================================================================
                                        coef    std err          z      P>|z|      [0.025      0.975]
-----------------------------------------------------------------------------------------------------
const                              1438.0524     41.522     34.633      0.000    1356.670    1519.435
Land Use                             -0.0587      0.032     -1.821      0.069      -0.122       0.004
Sold As Vacant                        3.3477      0.381      8.797      0.000       2.602       4.094
Multiple Parcels Involved in Sale    -0.5724      0.166     -3.445      0.001      -0.898      -0.247
Acreage                               0.1391      0.042      3.294      0.001       0.056       0.222
Tax District                         -0.1483      0.038     -3.896      0.000      -0.223      -0.074
Land Value                        -2.428e-06   3.31e-07     -7.328      0.000   -3.08e-06   -1.78e-06
Building Value                     1.128e-06   2.32e-07      4.862      0.000    6.74e-07    1.58e-06
Finished Area                      7.963e-06   4.99e-05      0.160      0.873   -8.99e-05       0.000
Foundation Type                      -0.0253      0.017     -1.498      0.134      -0.058       0.008
Exterior Wall                         0.0007      0.013      0.052      0.959      -0.024       0.025
Grade                                 0.0241      0.022      1.118      0.263      -0.018       0.066
Bedrooms                             -0.0140      0.033     -0.424      0.672      -0.079       0.051
Full Bath                             0.0714      0.040      1.767      0.077      -0.008       0.151
Half Bath                             0.0456      0.051      0.901      0.367      -0.054       0.145
Years_old                            -0.0001      0.001     -0.150      0.881      -0.002       0.002
saleyear                             -0.7142      0.021    -34.648      0.000      -0.755      -0.674
=====================================================================================================
```

Confusion Matrix:

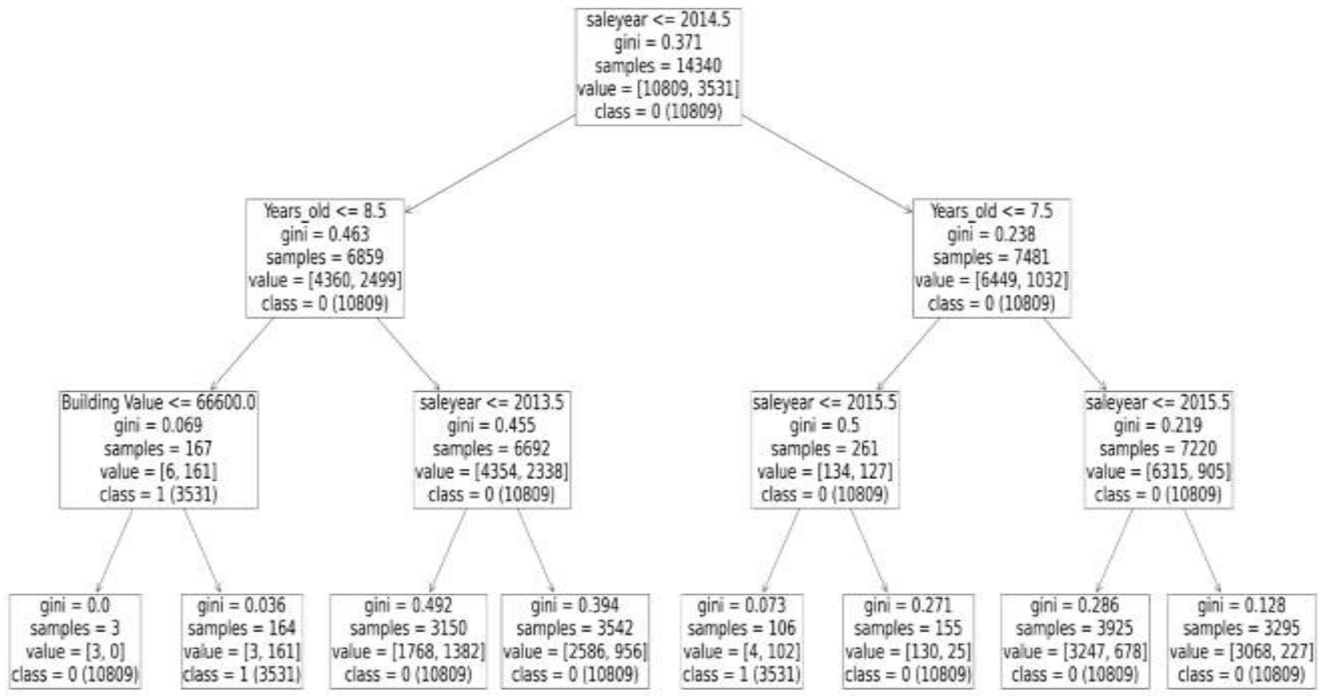


Performance Metrics:

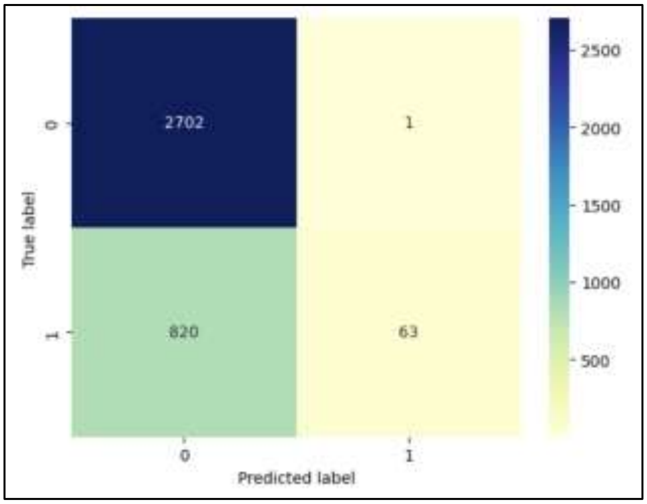


```
 Accuracy of this Model is :  0.7601784718349136
              precision    recall  f1-score   support

           0       0.77      0.98      0.86      2703
           1       0.60      0.08      0.14       883

    accuracy                           0.76      3586
   macro avg       0.68      0.53      0.50      3586
weighted avg       0.72      0.76      0.68      3586
```

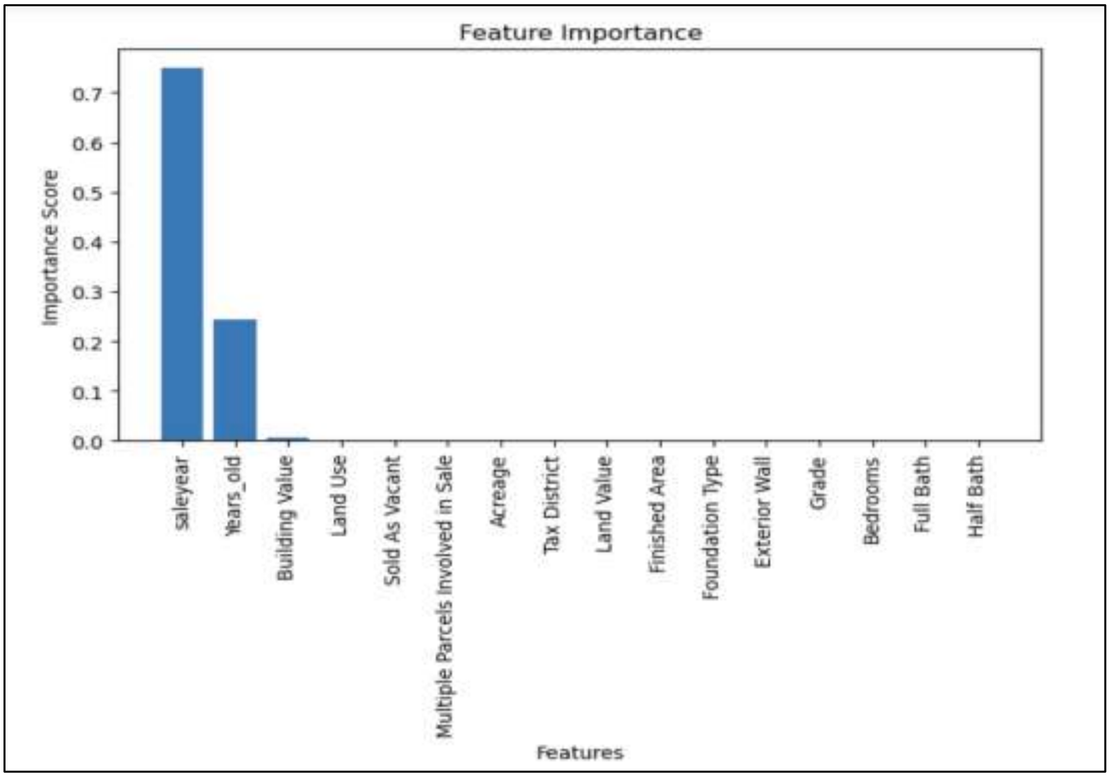
Decision Tree



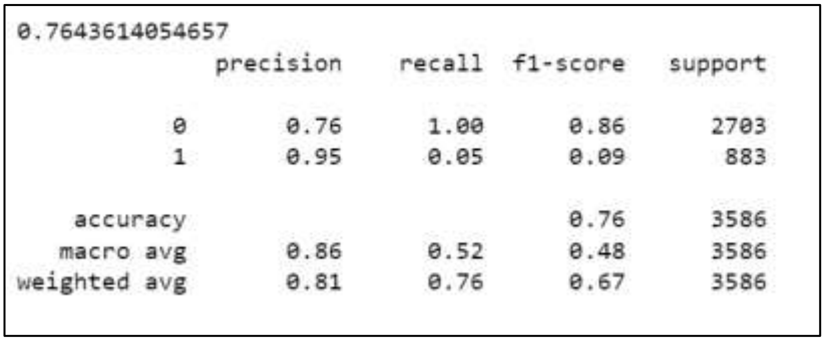
Confusion matrix:



Feature Importance:



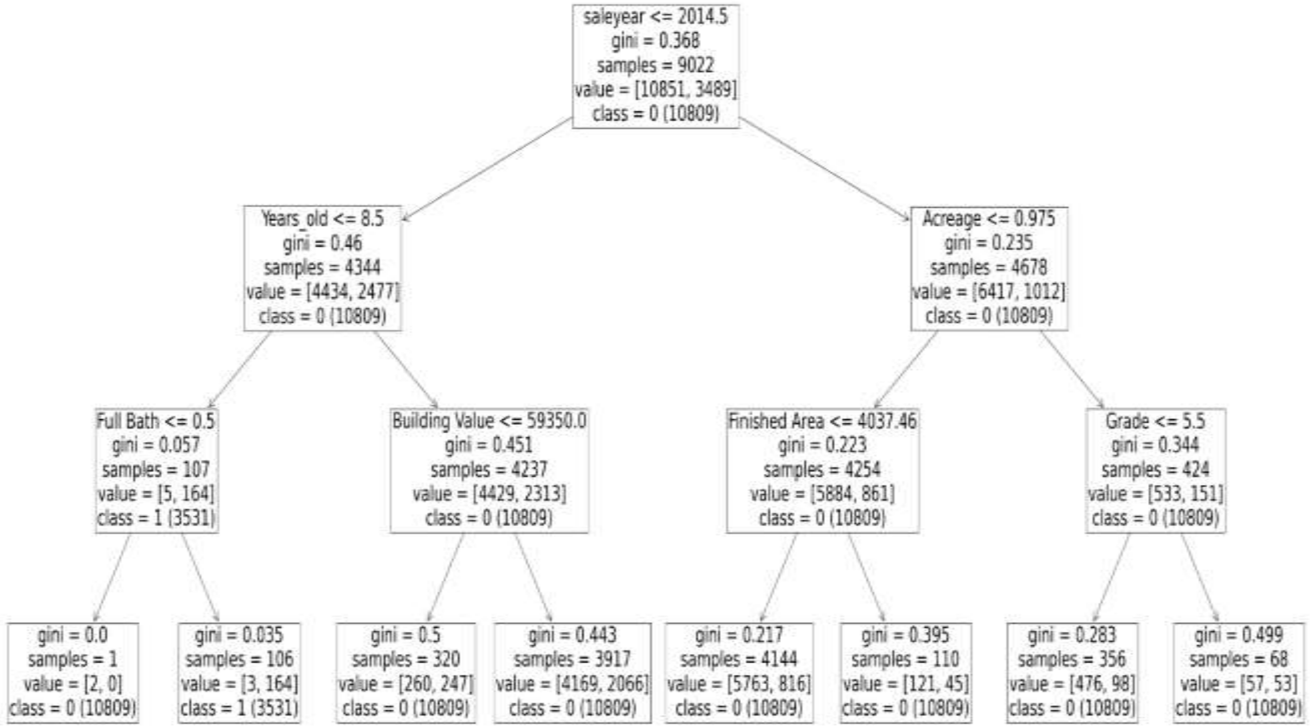
Performance Metrics:



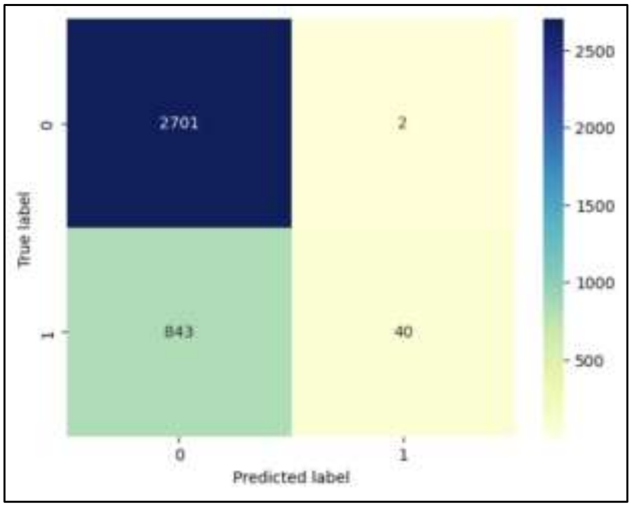
```
0.7643614054657
              precision    recall  f1-score   support

           0       0.76      1.00      0.86      2703
           1       0.95      0.05      0.09       883

    accuracy                           0.76      3586
   macro avg       0.86      0.52      0.48      3586
weighted avg       0.81      0.76      0.67      3586
```

Random Forest Model

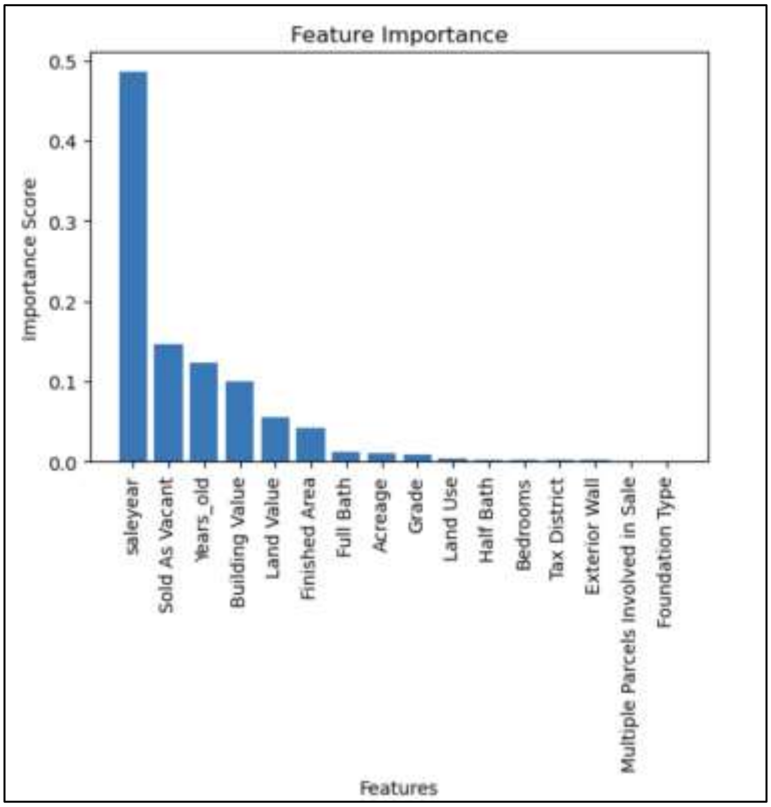
Model Summary:



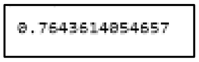
Confusion Matrix:



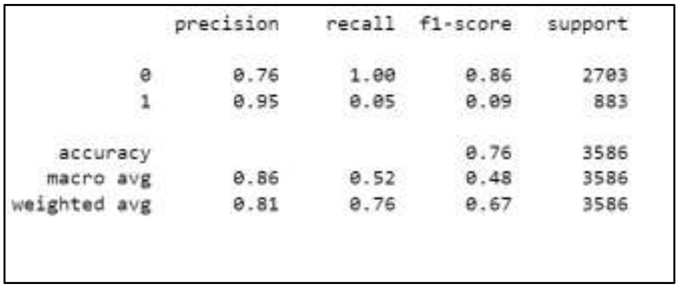
Feature Importance:



Performance Metrics:

Accuracy: 

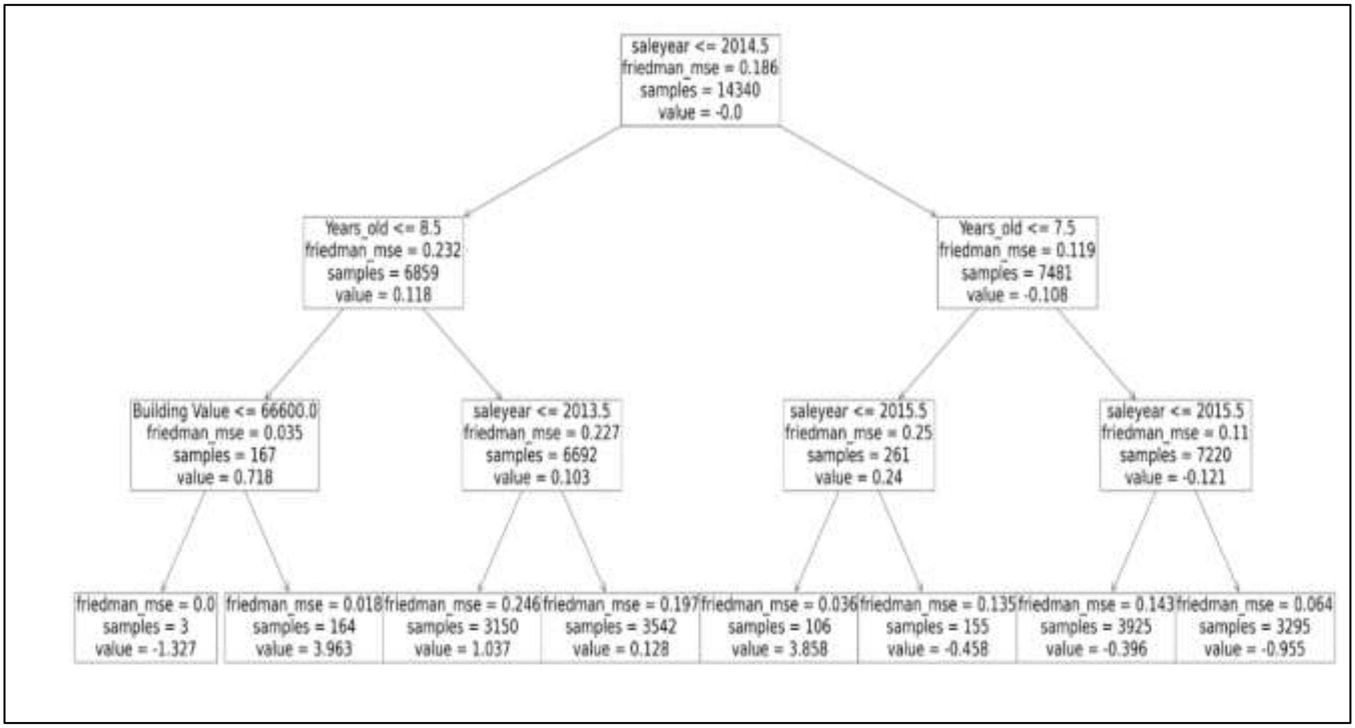
```
0.7643614054657
```



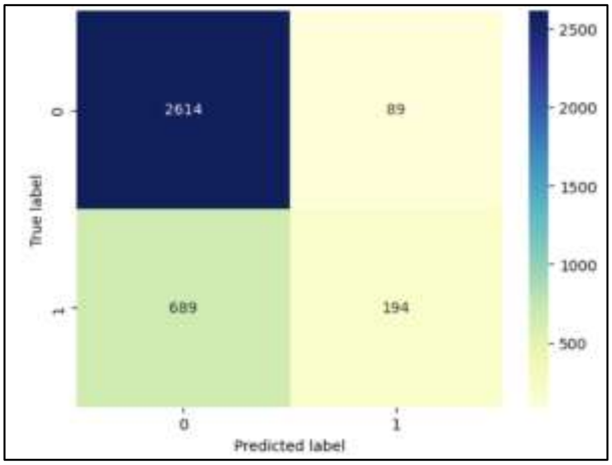
```
              precision    recall  f1-score   support

           0       0.76      1.00      0.86      2703
           1       0.95      0.05      0.09       883

    accuracy                           0.76      3586
   macro avg       0.86      0.52      0.48      3586
weighted avg       0.81      0.76      0.67      3586
```

Gradient Boosting Model

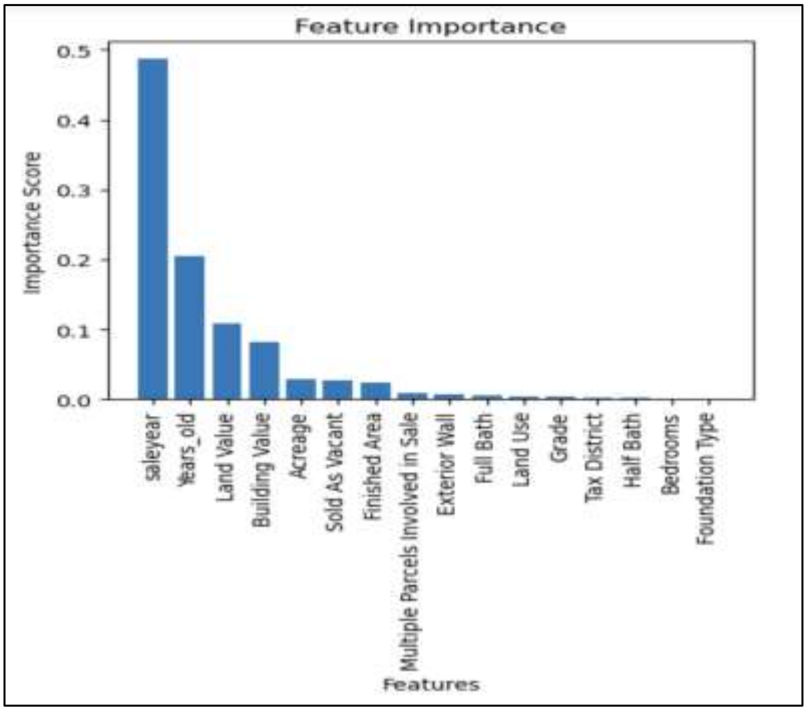
Model Summary:



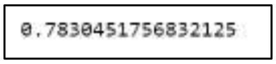
Confusion Matrix:



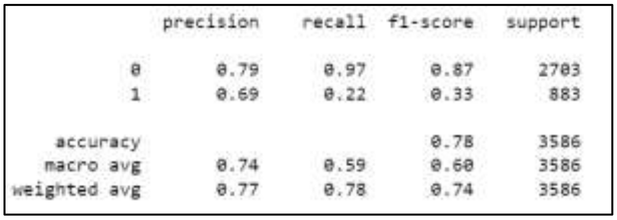
Feature Importance:



Performance Metrics:

Accuracy: 

```
0.7830451756832125
```



```
              precision    recall  f1-score   support

           0       0.79      0.97      0.87      2703
           1       0.69      0.22      0.33       883

    accuracy                           0.78      3586
   macro avg       0.74      0.59      0.60      3586
weighted avg       0.77      0.78      0.74      3586
```