BOSTON HOUSING PREDICTION AND CLASSIFICATION

Shravani Vaze

Muthouazhagi Dhanapal

CS5100

12/4/2019

Overview

This project is on Boston housing price prediction and area safety classification. The goal of this project is to develop a machine learning model that effectively estimates the price of the houses in Boston and also develop a machine learning model that determines how safe an area is to reside. We found this real estate domain to be interesting enough to apply various machine learning algorithms and analyze the performance of our model with respect to each of the algorithms. The reason why we focused on pricing and area safety is because these two factors plays an important role for people to decide on buying a house. We used the Boston housing dataset from Kaggle which provides housing values in Suburbs of Boston. It contains about fourteen features measuring various factors concerning the areas and houses in Boston.

We decided to use multiple linear regression to predict the prices of the houses in Boston. For area safety we decided to use various classification algorithms such as Decision Tree, Logistic Regression, Naïve Bayes and Random forest. The results of these algorithms will then be compared to determine which algorithm performed the best in classifying the areas with respect to its safety.

Data Set

We collected the Boston housing dataset from Kaggle which is derived from information collected by the U.S. Census Service concerning housing in the area of Boston MA. We chose this dataset because the dataset was clean and had enough number of records to both test and train the models. It contains fourteen features with respect to the housing in Boston.

Following are the features present in this dataset:

CRIM - This feature represents the per capita crime rate by town.

ZN - This feature represents the proportion of residential land zoned for lots over 25,000 sq. Ft.

INDUS – This feature represents the proportion of non-retail business acres per town.

CHAS – This feature represents the Charles River dummy variable (1 if tract bounds river; 0 otherwise)

NOX - This feature represents the nitric oxides concentration (parts per 10 million)

RM – This feature represents the average number of rooms per dwelling

AGE – This feature represents the proportion of owner-occupied units built prior to 1940

DIS – This feature represents the weighted distances to five Boston employment centers

RAD – This feature represents the index of accessibility to radial highways

TAX – This feature represents the full-value property-tax rate per $10,000

PTRATIO – This feature represents the pupil-teacher ratio by town

LSTAT - This feature represents the percentage of lower status of the population

MEDV – This feature represents the median value of owner-occupied homes in $1000's.

For the purpose of this project we have excluded certain features that were not required by our algorithms. The features that are used in this project are CRIM, RM, PTRATIO, LSTAT, TAX, MEDV. Certain columns with categorical variables were added for the purpose of Naïve Bayes classification algorithm:

residence: Represents the area safety and takes the values: 1 for safe and 0 for unsafe.

crime_category: Represents the severity of crime in the area and takes the values: low and high.

tax_category: Represents the rate of tax and takes the values: low, moderate and high.

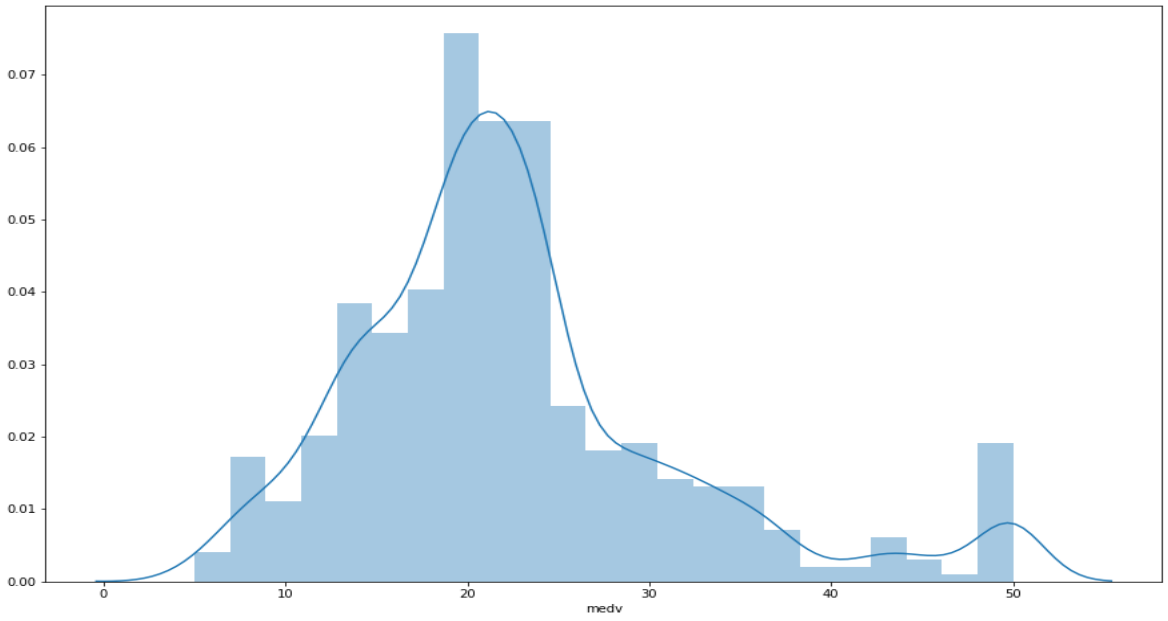
Boston housing dataset:

	crim	rm	tax	ptratio	lstat	medv	residence	crim_category	tax_category	rm_catego
2	0.00632	6.575	296	15.3	4.98	24	1	low	moderate	high
3	0.02731	6.421	242	17.8	9.14	21.6	1	low	moderate	high
4	0.02729	7.185	242	17.8	4.03	34.7	1	low	moderate	high
5	0.03237	6.998	222	18.7	2.94	33.4	1	low	low	high
6	0.06905	7.147	222	18.7	5.33	36.2	0	low	low	high
7	0.02985	6.43	222	18.7	5.21	28.7	1	low	low	high
8	0.08829	6.012	311	15.2	12.43	22.9	0	low	moderate	low
9	0.14455	6.172	311	15.2	19.15	27.1	0	low	moderate	low
10	0.21124	5.631	311	15.2	29.93	16.5	0	low	moderate	low
11	0.17004	6.004	311	15.2	17.1	18.9	0	low	moderate	low
12	0.22489	6.377	311	15.2	20.45	15	0	low	moderate	high
13	0.11747	6.009	311	15.2	13.27	18.9	0	low	moderate	low
14	0.09378	5.889	311	15.2	15.71	21.7	0	low	moderate	low
15	0.62976	5.949	307	21	8.26	20.4	0	low	moderate	low
16	0.63796	6.096	307	21	10.26	18.2	0	low	moderate	low
17	0.62739	5.834	307	21	8.47	19.9	0	low	moderate	low
18	1.05393	5.935	307	21	6.58	23.1	0	low	moderate	low
19	0.7842	5.99	307	21	14.67	17.5	0	low	moderate	low
20	0.80271	5.456	307	21	11.69	20.2	0	low	moderate	low
21	0.7258	5.727	307	21	11.28	18.2	0	low	moderate	low
22	1.25179	5.57	307	21	21.02	13.6	0	low	moderate	low
23	0.85204	5.965	307	21	13.83	19.6	0	low	moderate	low
24	1.23247	6.142	307	21	18.72	15.2	0	low	moderate	low
25	0.98843	5.813	307	21	19.88	14.5	0	low	moderate	low
26	0.75026	5.924	307	21	16.3	15.6	0	low	moderate	low
27	0.84054	5.599	307	21	16.51	13.9	0	low	moderate	low
28	0.67191	5.813	307	21	14.81	16.6	0	low	moderate	low
29	0.95577	6.047	307	21	17.28	14.8	0	low	moderate	low
30	0.77299	6.495	307	21	12.8	18.4	0	low	moderate	high
31	1.00245	6.674	307	21	11.98	21	0	low	moderate	high
32	1.13081	5.713	307	21	22.6	12.7	0	low	moderate	low
33	1.35472	6.072	307	21	13.04	14.5	0	low	moderate	low
34	1.38799	5.95	307	21	27.71	13.2	0	low	moderate	low
35	1.15172	5.701	307	21	18.35	13.1	0	low	moderate	low
36	1.61282	6.096	307	21	20.34	13.5	0	low	moderate	low

BostonHousing (1)

Dataset Analysis:

```
Statistics of Boston Housing Dataset
The minimum price of the house is 5.0
The maximum price of the house is 50.0
The mean price of the house is 22.532806324110698
The median price of the house is 21.2
```



Price Prediction

Technical Problem Statement

For price prediction of houses, the data has been formulized as a multiple linear regression which attempts to model the relationship between the predictor variables and the response variable thereby predicting the price of houses by fitting a linear equation to the observed data.

Formally a multiple linear regression model with several predictor variables X_1, X_2, …., X_k and one response variable Y can be written as

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots \beta_k x_k + \epsilon.$$

Input

The features 'RM','LSTAT' and 'PTRATIO' were chosen as explanatory variables since they were found to be the best attributes in predicting the prices of houses. The feature 'MEDV' was chosen as the dependent variable which is the price of houses that we seek to predict.

Expected Output

The expected output from linear regression is the estimation of housing prices in Boston. The calculated values are compared with the actual prices present in the data set and this is visualized in the form of bar graph.

Method

Multiple Linear Regression

For our algorithm the least squares method was used with a 30% training size on 506 records of Boston housing data. The python library that was used for this model is sklearn. The independent variables RM, LSTAT, PTRATIO were taken for X values and the dependent variable 'MEDV' was chosen for Y value.

We made some intuitive assumptions on the features that were taken as X values:

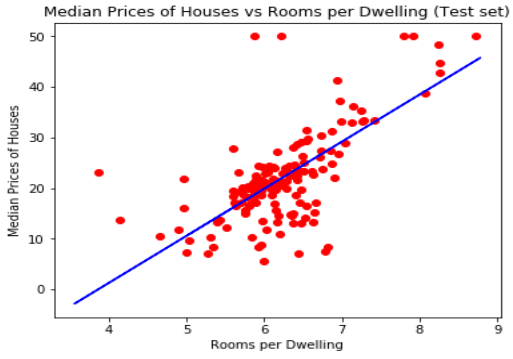
Houses with higher 'RM' value will be valued more , since more the number of rooms the prices of houses will be more . So the 'RM' and house cost will be directly proportional.

Higher 'LSTAT' value will be valued less, since higher the value of LSTAT denotes that the income of these people are low and hence their houses will be small. So the 'LSTAT' and house cost will be inversely proportional.

Higher 'PTRATIO' value will be valued less, This is because if the percentage of students to teachers ratio people is higher, it is likely that in the neighborhood there are less schools, this could be because there is less tax income which could be because in that neighborhood people earn less money. If people earn less money it is likely that their houses are worth less. They are inversely proportional variables. So the 'PTRATIO' and house cost will be inversely proportional.

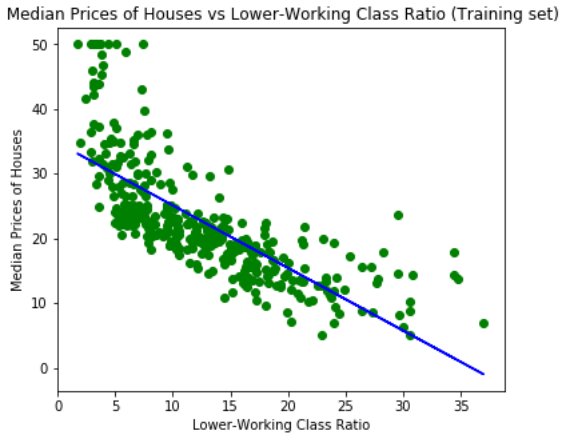
We verified each of our assumptions by generating the scatter plot of each feature against the MEDV value for training as well as test data set.

MEDV and RM:



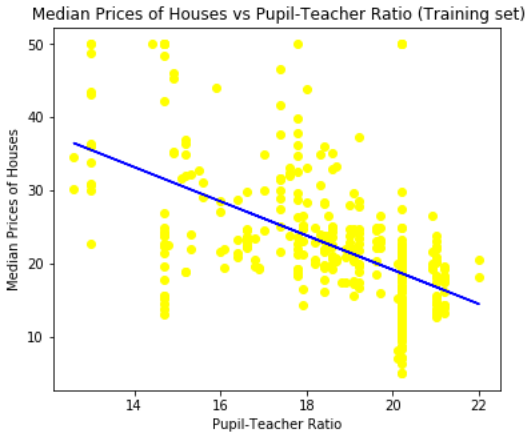
As seen from the graph , Rooms per Dwelling (RM) is directly proportional to Price cost.

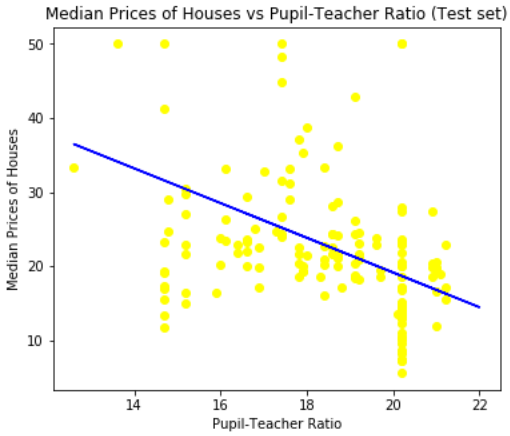
MEDV and LSTAT :



As seen from the graph , LSTAT is inversely proportional to House prices.

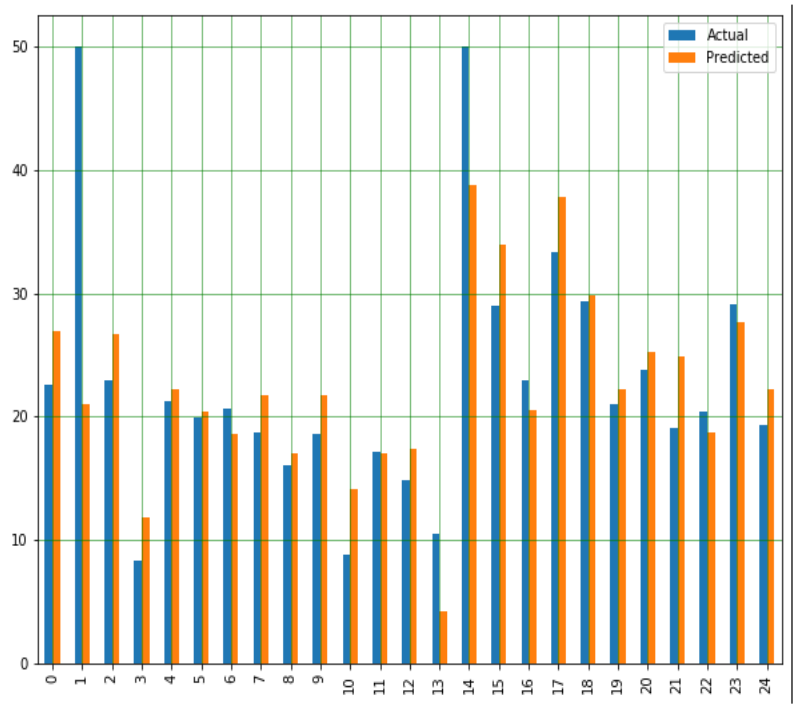
MEDV and PTRATIO:





As seen from the graph, PTRATIO is inversely proportional to house prices.

COMPARISON OF RESULTS OF MULTIPLE LINEAR REGRESSION WITH THE EXPECTED PRICE PREDICTION:




```
Mean Absolute Error: 3.9758660362578895
Mean Squared Error: 33.99646726021234
Root Mean Squared Error: 5.8306489570383455
```

The orange bars represents the output of our multiple linear regression model and the blue bar represents the expected output.

Result Analysis

On analyzing the result of comparison between the actual and predicted values of our model , we found that the accuracy of our model is lower than expected. The reasons for this is that the dataset chosen is scattered. Also, If the dataset contained more values then then the split ratio for training and testing could be improved. Hence if more values are given to the model better results could be obtained.

AREA SAFETY CLASSIFICATION

Technical Problem Statement

For area safety classification, four classification algorithms will be used.

Logistic Regression:

Logistic Regression is used to predict the probability of a categorical dependent variable Y which is the area safety in our case . In logistic regression, the dependent variable Y is a binary variable that contains data coded as 1 (safe area.) or 0 (unsafe area). In other words, the logistic regression model predicts P(Y=1) as a function of X. In this problem , the dependent variable ‘ residence’ denotes the area safety which is predicted using the categorical independent variables ‘RM’ , ‘LSTAT’ , ‘TAX.

Decision Tree:

A decision tree is a tree in which each internal (non-leaf) node is labeled with an input feature. The arcs coming from a node labeled with a feature are labeled with each of the possible values of the feature. Each leaf of the tree is labeled with a class or a probability distribution over the classes. In this problem , the features used are ‘RM’ , ‘LSTAT’ , ‘TAX’ to predict the area safety ‘ residence’.

Naïve Bayes:

Naive Bayes is a probabilistic machine learning model that’s used for classification. Using Bayes theorem, we can find the probability of A happening, given that B has occurred. Here, B is the evidence and A is the hypothesis. The assumption made here is that the predictors/features are independent. In this problem the independent features are ‘RM’ , ‘LSTAT’ and ‘TAX’ which are used to predict the area safety ‘ residence’. As per the Bayes’ theorem :

$$P(y|x_1,...,x_n) = \frac{P(x_1|y)P(x_2|y)...P(x_n|y)P(y)}{P(x_1)P(x_2)...P(x_n)}$$

The variable y is the class variable – as per this problem it would be ‘residence’ which represents if an area is safe or not given the conditions. Variable $x_1,\ldots,x_n$ represent the various features used in this problem : RM , LSTAT and TAX.

Random Forest:

The Random Forest classifier works by constructing a series of decision trees to classify the area safety using the features RM , LSTAT and TAX. It then goes down

each of the trees in the forest and gets a classification from each of them, ultimately giving the area the label (Safe / Unsafe) with the most votes.

Input

For the classification algorithms : Decision Tree , Logistic Regression and Random Forest we decided to use - 'CRIM' , 'RM' and 'TAX' as the features responsible for predicting area safety 'residence'. The variable 'residence' represents the categorical variable which denotes whether the area is safe or not.

To apply Naïve Bayes algorithm each of these features : CRIM , RM and TAX need to be categorical variables due to which we added the 'crime_category' , 'tax_category', 'rm_category' to our dataset to determine the area safety 'residence' (dependent variable) . These categorical variables were used as input to Naïve Bayes algorithm.

Output

To decide the area safety, the output of decision tree / logistic regression / Naïve Bayes and Random forest is visualized in the form of confusion matrix.

Confusion matrix is a table that evaluates the accuracy of our classification algorithms. The four different combinations of predicted and actual values are True Positive , False Positive , False Negative , True negative.

True Positive: You predicted positive and it's true.

True Negative: You predicted negative and it's true.

False Positive: You predicted positive and it's false.

False Negative: You predicted negative and it's false.

Based on the above values form confusion matrix of each of the classification algorithms, the performance of the algorithms are compared.

Methods

For Linear regression, Decision tree and Naïve Bayes : CRIM , RM and TAX are considered as independent input features and we made certain assumptions on these features.

Feature Assumptions:

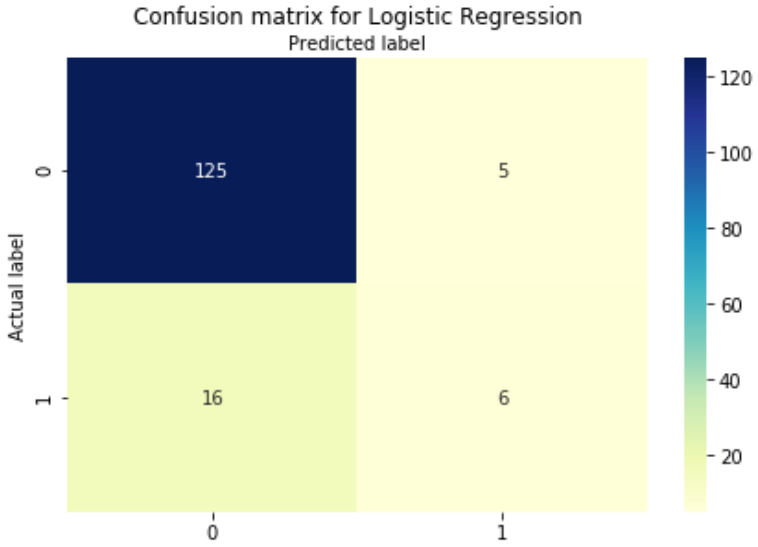
‘CRIM’: When the crime rate is higher , then the area will be considered unsafe.

‘RM’: Less number of rooms per dwelling denotes that the houses are smaller and people with less income reside in the area. Low income might tend people to get involved in crimes, So the lower RM value will classify the area as unsafe.

‘Tax’: Lower property tax indicates that the smaller houses which in turn indicates that people with less income reside in the area. Low income might tend people to get involved in crimes, So the lower Tax value will classify the area as unsafe.

These features were used to predict the categorical dependent variable ‘residence’. This variable contains values 0 for unsafe and 1 for safe.

Results of Logistic Regression



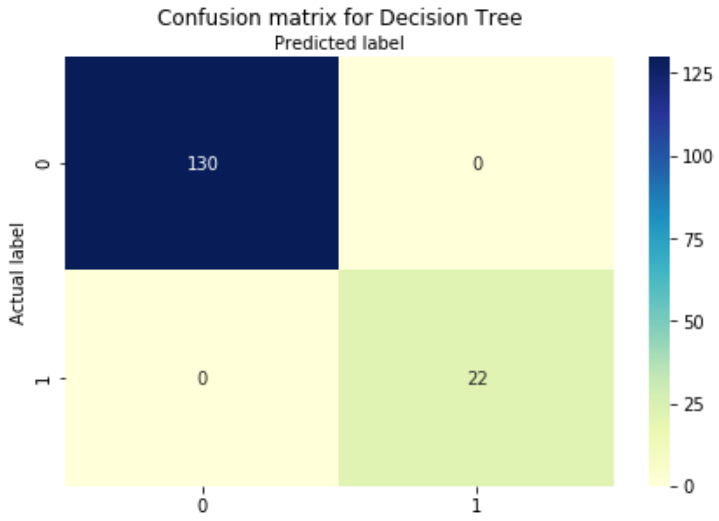
Analysis

The accuracy results of our model using logistic regression contains:

125 records classified as True positive , 5 records classified as False Positive ,

16 records classified as False negative and 6 records classified as True negative.

Results of Decision Tree:



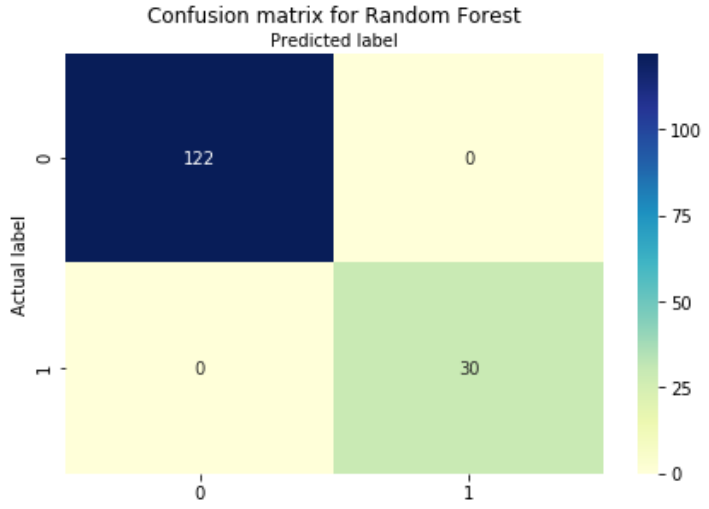
Analysis

The accuracy results of our model using logistic regression contains:

130 records classified as True positive , 0 records classified as False Positive ,

0 records classified as False negative and 22 records classified as True negative.

Results of Random Forest



Analysis

The accuracy results of our model using logistic regression contains:

122 records classified as True positive , 0 records classified as False Positive ,

0 records classified as False negative and 30 records classified as True negative.

Naïve Bayes

For Naïve Bayes algorithm: Three categorical independent variables were used : ‘rm_category’ , ‘tax_category’,’crim_category’. The possible values of each of these variables are :

rm_category: high, low.

High represents that the number of rooms are high and low represents that the number of rooms are lower.

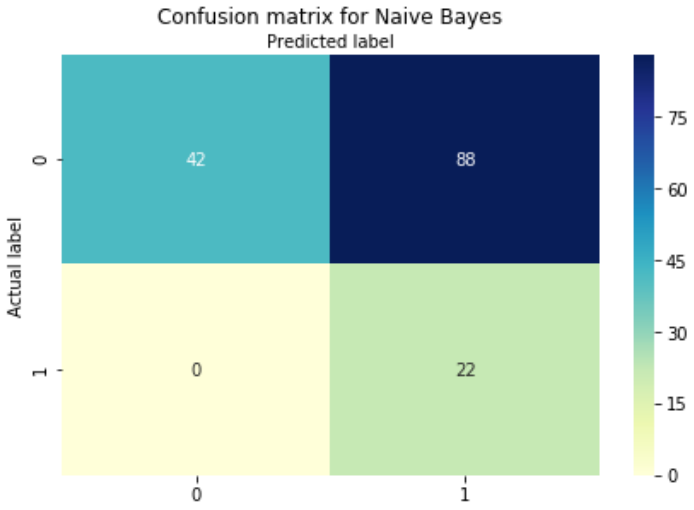
tax_category: moderate, low, high.

Moderate denotes that the tax is generally moderate in these areas , low denotes that the property tax is low and high represents that the property tax is higher.

crim_category: low, high.

Low represents that the crime is generally low in these areas , high represents that the crim is generally high in these areas.

Based on these categorical independent variables , the response variable ‘residence’ is predicted.



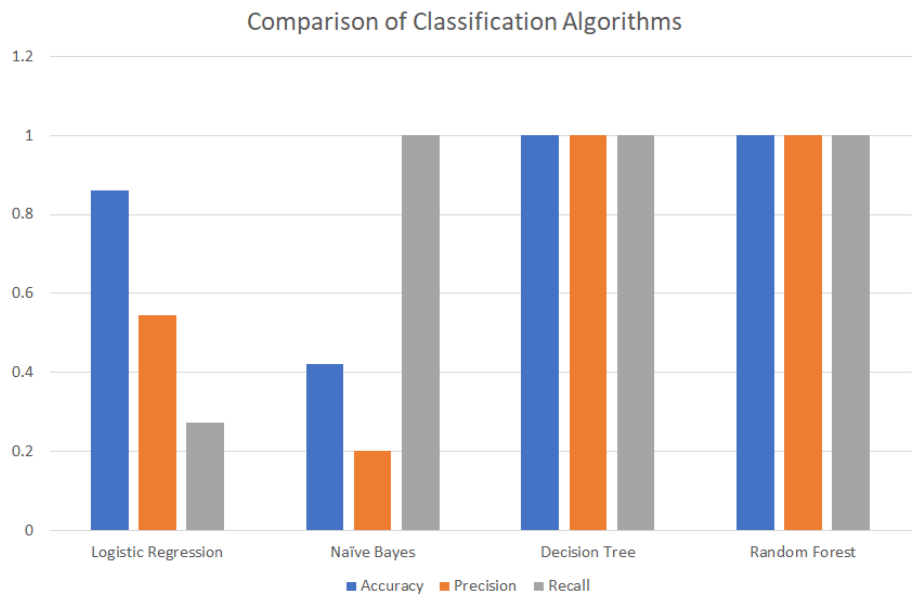
Analysis

The accuracy results of our model using Naïve Bayes contains:

42 records classified as True positive , 88 records classified as False Positive ,

0 records classified as False negative and 22 records classified as True negative.

Performance comparison of Classification algorithms:




```
('Accuracy for Logistic Regression:', 0.8618421052631579)
('Precision:', 0.5454545454545454)
('Recall:', 0.2727272727272727)
('Accuracy for decesion tree:', 1.0)
('Precision:', 1.0)
('Recall:', 1.0)
('Accuracy for Naive Bayes:', 0.42105263157894735)
('Precision:', 0.2)
('Recall:', 1.0)
```

```
('Accuracy for Random Forest:', 1.0)
('Precision:', 1.0)
('Recall:', 1.0)
```

Where,

Accuracy represents the performance measure and it is simply a ratio of correctly predicted observation to the total observations.

Accuracy = TP+TN/TP+FP+FN+TN

Precision represents the ratio of correctly predicted positive observations to the total predicted positive observations.

Precision = TP/TP+FP

Recall represents the ratio of correctly predicted positive observations to the all observations in actual class.

Recall = TP/TP+FN

On analysis of performance comparison , we found that decision tree and random forest algorithms performed the best with accuracy rate of 1.0. The reason that decision tree and random forest performed better than logistic regression is that the more number of X features , decision tree and random forest performs well as it trisects the data into more parts to take a decision, if there had been only one X feature , Logistic regression would have performed better. Note that Random forest is a strong modelling technique and much more robust than a single decision tree as it predicts based on the majority of votes from each of the decision trees.

Future Directions:

Using more features

We used the features 'RM' , 'PTRATIO' , LSTAT' for predicting the prices of house , we believe that more features could have been taken into account. For example ,

the features : Tax , accessibility to radial highways could have also been included as independent variables to predict the price using linear regression .

Better training and testing data split

The training size used for this project is 30%. We felt that a training size of 50% could have been used which could have improved the accuracy of our models using linear regression.