**Overview**

The main purpose of our project is to build a classification model for the dataset using decision tree and naive bayes method. Analyze the results of all models and understand the contribution of all attributes for the client to subscribe to term deposit. Also compare all models and do some analysis on top of it.

* Firstly, we went through the provided problem statement and the requirements to complete this project. So, we started learning R programming language and the theoretical concepts of decision tree and naïve bayes.
* We started cleaning the data by removing the rows containing unknown values. Also dropped the attributes which are not relevant.
* We have installed the packages that are required for developing the decision tree (rpart, rpart.plot).
* After that we set the seed and took the sample data from the given data set.
* We divided the sample data into train data and test data. After that, we applied rpart function on train data to generate two decision trees, one based on information gain as a splitting criterion and the other based on GINI Index. Visualize the tree using plot function.
* We have also built a naïve bayes model.
* We predicted y for the test data using the decision trees generated. After that, we compared the predicted values with the actual test data values and generated confusion matrix. We tested the performance of the tree based on analysis of some metrics like accuracy, precision, recall and F1 which are generated by using confusion matrix.
* Lastly, we went through the above process after with holding a column and analyzed the difference.

**File Description**

**Logic:** The process is common for the all the implementations. Below are the steps followed,

* First, we read the data set from the csv file in the local system.
* Import all the required libraries like required like dplyr, rpart and rpart.plot.
* Clean the data by removing the records containing “unknown” values.
* Remove the attributes “marital, default, housing, loan, contact “as they are not relevant for analysis.
* Factorizing education, job, month, and day\_of\_week.
* Setting seed and creating sample data as provided.
* Dividing the sample data for training and testing parts.
* Using the training data to plot the decision tree (or) Naive Bayes model and test the output with testing data.
* Calculating and Printing out confusion matrix, accuracy, precision, recall and F1 value.

**DecisionTreeGINI\_50\_50.R :** Builds decision tree model using GINI Index. Split for training and test data is 50/50.

**DecisionTreeGINI\_80\_20.R :** Builds decision tree model using GINI Index. Split for training and test data is 80/20.

**DecisionTreeWithholdGINI\_50\_50.R :** Drops the attribute ‘previous’ and builds decision tree model using GINI Index. Split for training and test data is 50/50.

**DecisionTreeWithholdGINI\_80\_20.R :** Drops the attribute ‘previous’ and builds decision tree model using GINI Index. Split for training and test data is 80/20.

**DecisionTreeInfG\_50\_50.R :** Builds decision tree model using Information gain. Split for training and test data is 50/50.

**DecisionTreeInfG\_80\_20.R :** Builds decision tree model using Information gain. Split for training and test data is 80/20.

**DecisionTreeWithholdInfG\_50\_50.R :** Drops the attribute ‘previous’ and builds decision tree model using Information gain. Split for training and test data is 50/50.

**DecisionTreeWithholdInfG\_80\_20.R :** Drops the attribute ‘previous’ and builds decision tree model using Information gain. Split for training and test data is 80/20.

**NaiveBayes\_50\_50.R:** Builds naïve bayes model. Split for training an test data is 50/50.

**NaiveBayes\_80\_20.R:** Builds naïve bayes model. Split for training and test data is 80/20.

**NaiveBayesWithhold\_50\_50.R:** Drops the attribute ‘previous’ and builds naïve bayes model. Split for training and test data is 50/50.

**NaiveBayesWithhold\_80\_20.R:** Drops the attribute ‘previous’ and builds naïve bayes model. Split for training and test data is 80/20.

**Division of Work:**

Sai krishna: Worked for Cleaning the data and implemented the decision tree. Also analyzed the outcomes of the tree.

Deepika: Worked for Cleaning the data and implemented the naive bayes model.

**Problems Faced:**

* While loading the data, we were unable to load it directly by using the delimiter because data has double quotes as qualifiers. There are few ways to deal with them but they were not working. So, we replaced double quotes with empty space and loaded the data.
* While splitting data between training and test sets, for categorical attributes few levels were not included in the training data but were included in test data. Due to this the model failed and we solved this issue by factorizing those attributes.

**Results:**

**Decision Tree Using Information Gain:**

**80/20 Split:** Training Data: 8000, Test Data: 2000

Confusion Matrix:

|  |  |  |
| --- | --- | --- |
|  | No | Yes |
| No | 1668 | 62 |
| Yes | 140 | 130 |

Precision: 0.9641618

Recall: 0.9225664

F1 score: 0.9429056

Accuracy: 0.899

**50/50 Split:** Training Data: 5000, Test Data: 5000

Confusion Matrix:

|  |  |  |
| --- | --- | --- |
|  | No | Yes |
| No | 4173 | 152 |
| Yes | 334 | 341 |

Precision: 0.9648555

Recall: 0.9258931

F1 score: 0.9449728

Accuracy: 0.9028

**Withholding ‘previous’ attribute:**

**80/20 Split:** Training Data: 8000, Test Data: 2000

Confusion Matrix:

|  |  |  |
| --- | --- | --- |
|  | No | Yes |
| No | 1668 | 62 |
| Yes | 140 | 130 |

Precision: 0.9641618

Recall: 0.9225664

F1 score: 0.9429056

Accuracy: 0.899

**50/50 Split:** Training Data: 5000, Test Data: 5000

Confusion Matrix:

|  |  |  |
| --- | --- | --- |
|  | No | Yes |
| No | 4173 | 152 |
| Yes | 334 | 341 |

Precision: 0.9648555

Recall: 0.9258931

F1 score: 0.9449728

Accuracy: 0.9028

**Decision Tree Using GINI Index:**

**80/20 Split:** Training Data: 8000, Test Data: 2000

Confusion Matrix:

|  |  |  |
| --- | --- | --- |
|  | No | Yes |
| No | 1664 | 66 |
| Yes | 134 | 136 |

Precision: 0.9618497

Recall: 0.9254727

F1 score: 0.9433107

Accuracy: 0.9

**50/50 Split:** Training Data: 5000, Test Data: 5000

Confusion Matrix:

|  |  |  |
| --- | --- | --- |
|  | No | Yes |
| No | 4173 | 152 |
| Yes | 333 | 342 |

Precision: 0.9648555

Recall: 0.9260985

F1 score: 0.9450798

Accuracy: 0.903

**Withholding ‘previous’ attribute:**

**80/20 Split:** Training Data: 8000, Test Data: 2000

Confusion Matrix:

|  |  |  |
| --- | --- | --- |
|  | No | Yes |
| No | 1664 | 66 |
| Yes | 134 | 136 |

Precision: 0.9618497

Recall: 0.9254727

F1 score: 0.9433107

Accuracy: 0.9

**50/50 Split:** Training Data: 5000, Test Data: 5000

Confusion Matrix:

|  |  |  |
| --- | --- | --- |
|  | No | Yes |
| No | 4173 | 152 |
| Yes | 333 | 342 |

Precision: 0.9648555

Recall: 0.9260985

F1 score: 0.9450798

Accuracy: 0.903

**With holding attributes identified earlier:** Dropping the attributes age, day\_of\_week and pdays. Basis for dropping these is mentioned in the milestone report.

**80/20 Split:** Training Data: 8000, Test Data: 2000

Confusion Matrix:

|  |  |  |
| --- | --- | --- |
|  | No | Yes |
| No | 1665 | 65 |
| Yes | 135 | 135 |

Precision: 0.9624277

Recall: 0.925

F1 score: 0.9433428

Accuracy: 0.9

The accuracy of the model is not changed, and the results are identical. So, as identified these attributes as not relevant for classification.

**Naïve Bayes Model:**

**80/20 Split:** Training Data: 8000, Test Data: 2000

Confusion Matrix:

|  |  |  |
| --- | --- | --- |
|  | No | Yes |
| No | 1560 | 112 |
| Yes | 170 | 158 |

Precision: 0.9330144

Recall: 0.9017341

F1 score: 0.9171076

Accuracy: 0.859

**50/50 Split:** Training Data: 5000, Test Data: 5000

Confusion Matrix:

|  |  |  |
| --- | --- | --- |
|  | No | Yes |
| No | 3916 | 276 |
| Yes | 409 | 399 |

Precision: 0.9341603

Recall: 0.9054335

F1 score: 0.9195726

Accuracy: 0.863

**Withholding ‘previous’ attribute:**

**80/20 Split:** Training Data: 8000, Test Data: 2000

Confusion Matrix:

|  |  |  |
| --- | --- | --- |
|  | No | Yes |
| No | 1569 | 106 |
| Yes | 161 | 164 |

Precision: 0.9367164

Recall: 0.9069364

F1 score: 0.9215859

Accuracy: 0.8665

**50/50 Split:** Training Data: 5000, Test Data: 5000

Confusion Matrix:

|  |  |  |
| --- | --- | --- |
|  | No | Yes |
| No | 3934 | 249 |
| Yes | 391 | 426 |

Precision: 0.9404733

Recall: 0.9095954

F1 score: 0.9247767

Accuracy: 0.872

**Basis and Analysis of withholding the attribute ‘previous’:**

We have withhold previous attribute because in our sample data out of 10000 there are 8457 instances of ‘0’ occurred, and also in our training data out of 8000 there are 6739 of ‘0’ which makes our tree to make analysis by using other attributes rather than previous as it cannot classify the label on ‘0’ value. And also, it have less instances of other values so if any instance with other than ‘0’ value arises then there are high chances that it may classify it as wrong.

In decision tree models even after dropping the ‘previous’ attribute the results were identical. There is no change in the accuracy of all the models. Whereas in Naïve Bayes the accuracy increased very slightly.

**Analysis of Models:**

Below are the observations of decision tree using Information gain (80/20),

If nr.employed < 5088 and duration is greater than 826 then there is 60% chance of term deposit. If nr.employed >= 5088 below are the cases,

1. duration >=166 and pdays>=19 then there is 84% chance of term deposit
2. duration >=342 and pdays<19 then there is 65% chance of term deposit

In all remaining cases there is less chance for subscribing to term deposit. Similar conclusions can be drawn from all the models. The important attributes which determine whether the client subscribed to term deposit are nr.employed, duration and the others have minor effects. After comparing the results of 80/20 and 50/50 splits we observed that accuracy of models increased in 50/50 split.

**Comparision:**

Compared all the performance measures of 3 models in an excel sheet named ‘ComparisionDetails’.