Project 1: Classification (Data Mining)

Name – Navtejinder Singh Brar
UTA ID – 1001874259

Description:

The dataset named bank.additional.full is provided, which has 21 attributes('y' as the class attribute) and 41188 objects. For the pre-processing we are asked to remove all the objects which had unknown values in it. After removing those objects, 30488 entries are left instead of 41188. Moreover, we are asked to remove five attributes named *marital, default, housing, loan* and *contact* from the dataset as the part of pre-processing. Then, we identified 4 attributes based on histogram analysis, which we will discuss as we move on in this report. After pre-processing we used the seed value of 170(group number is 17) as, asked in the description and created a random sample dataset of 10,000 objects/tuples. Then we divided the data into two splits (80(training):20(testing) and 50:50) and applied the GINI, Information gain and Naïve Bayes to the training data set and then predicted the class of the test data set with the decision tree from GINI and Information gain, and probabilistic model from naïve bayes. For GINI and information gain libraries rpart and rattle (for plotting) were used and the value of xval (cross validations) was set to 10 to get the best results. In addition, caret and e1071 libraries were used for naïve bayes.

GINI index classification using decision tree:

Gini index is the measure of impurity we use to make the decision tree using greedy algorithm (commonly used). GINI index is used when we split the tree into different nodes, GINI is used to determine the best split i.e. with the least impurity or max (GINI(parent) – GINI(child)). Splitting is done until we reach the conditions terminating the algorithm

- All samples for a given node belong to the same class.
- There are no remaining attributes for further partitioning.
- There are no samples left.

Coming to the given dataset:

For 80-20 split we applied the GINI index classification using decision tree we got the results as shown in the picture below:

	giniprediction	
	Predicted:no	Predicted:yes
Actual:no	1655	104
Actual:yes	103	138

Figure 1

Accuracy : 0.8965
Precision : 0.5702
Recall : 0.5726
F1 Score : 0.5714

Information gain (Entropy) classification using decision tree:

Here we construct the decision tree using the information gain. First the entropy of the parent node is found using the formula and then different splits are tried. After that, the split that has the maximum information gain that is max(Entropy(parent) – Entropy(child)) is chosen and the splits are further repeated until we get the ideal tree (Until algorithm terminates).

Coming to the given dataset:

For 80-20 split we applied the Information Gain classification using decision tree we got the results as shown in the picture below:

Figure 2

```
            igprediction
             Predicted:no Predicted:yes
Actual:no            1651            108
Actual:yes            102            139
```

Accuracy : 0.895
Precision : 0.5628
Recall : 0.5768
F1 Score : 0.5697

Comparison of GINI and Information Gain:

Without withholding any attribute (80:20 split):

As expected, when we do not remove any attribute from the dataset the accuracy, precision, recall and F1 score for both GINI and Information gain are almost identical. Duration attribute has the highest variable importance and day_of_week has the lowest variable importance in both GINI and Information Gain.

	Accuracy	Recall	Precision	F1 Score
GINI	0.8965	0.5726	0.5702	0.5714
Information Gain	0.895	0.5768	0.5628	0.5697

Figure 3

Without withholding any attribute (50:50 split):

Same pattern was seen while taking the 50:50 split. The accuracy, precision, recall and F1 score of both GINI and Information Gain was almost identical. Duration attribute has the highest variable importance and day_of_week has the lowest variable importance in both GINI and Information Gain.

	Accuracy	Recall	Precision	F1 Score
GINI	0.8982	0.5526	0.6506	0.5976
Information Gain	0.8972	0.5	0.6654	0.5709

Figure 4

Withholding duration attribute (80:20 split):

Duration attribute is the most important attribute in the data set (GINI (vi=373.6002), IG (vi=386.3308)). When we remove this attribute there is no significant change seen in the accuracy of both GINI and Information Gain because, the data is skewed towards the 'no' value of class attribute 'y' (around 87% are no). But, the recall of the GINI drops to 0.1618 and the recall of Information gain drops to 0.2282 because, recall= TP/TP+FN (tree is unable to get true positive values). Moreover, the precision for the GINI index has increased by 0.2 and F1 score for both GINI and IG reduced significantly (Because of huge fall in recall).

giniprediction

	Predicted:no	Predicted:yes
Actual:no	1743	16
Actual:yes	202	39

Figure 5(above) and 6(below)

igprediction

	Predicted:no	Predicted:yes
Actual:no	1716	43
Actual:yes	186	55

	Accuracy	Recall	Precision	F1 Score
GINI	0.891	0.1618	0.7091	0.2635
Information Gain	0.8855	0.2282	0.5612	0.3245

Figure 7

Withholding duration attribute (50:50 split):

Pattern similar to 80:20 split was seen in the 50:50 split by dropping the duration attribute. Precision for both GINI and Information Gain increased significantly and recall of both decreased significantly. Thus, decreasing the F1 score. In addition to this, surprisingly the values of all the four parameters (Accuracy, Precision, recall, F1 Score) for both GINI and Information Gain was exactly same.

```
          giniprediction
           Predicted:no Predicted:yes
Actual:no          4266            50
Actual:yes          543           141
          igprediction
           Predicted:no Predicted:yes
Actual:no          4266            50
Actual:yes          543           141
```

	Accuracy	Recall	Precision	F1 Score
GINI	0.8814	0.2061	0.7382	0.3223
Information Gain	0.8814	0.2061	0.7382	0.3223

Figure 8, 9 and 10.

Withholding day_of_week attribute (80:20):

day_of_week attribute is the least important attribute in the dataset (GINI (vi=0.9776), IG (vi=10.1780)). After removing this attribute change near to NIL was seen in the accuracy, precision, recall and F1. So, we can delete this attribute without affecting the prediction capability of the models.

```
          giniprediction
           Predicted:no Predicted:yes
Actual:no          1655           104
Actual:yes          103           138

          igprediction
           Predicted:no Predicted:yes
Actual:no          1655           104
Actual:yes          106           135
```

	Accuracy	Recall	Precision	F1 Score
GINI	0.8965	0.5726	0.5702	0.5714
Information Gain	0.895	0.5602	0.5646	0.5625

Figure 11, 12 and 13.

Withholding day_of_week attribute (50:50):

In case of 50:50 split pattern similar to 80:20 split was seen. Instead, there was a little bit of variation in all four values for the GINI, that is recall of the GINI decreased and precision increased thereby, decreasing the F1 score of GINI for 50:50 split (I think this is because of the small training data set).

```
          giniprediction
           Predicted:no Predicted:yes
Actual:no          4177           139
Actual:yes          368           316

          igprediction
           Predicted:no Predicted:yes
Actual:no          4145           171
Actual:yes          342           342
```

	Accuracy	Recall	Precision	F1 Score
GINI	0.8986	0.4619	0.6945	0.5549
Information Gain	0.8974	0.5	0.6667	0.5714

Figure 14, 15 and 16.

Classification using naïve bayes approach:

The naïve bayes is the probabilistic model in which we calculate the probability of every object with attributes and classify the object based on that probability. In the implementation I am leaving naïve bayes as naïve without adding the smoothing method.

Coming to the given dataset:

For 80-20 split we applied the Naïve Bayes classification we got the results as shown in the picture below: (no attributes deleted)

```
          naiveclassyprediction
           Predicted:no Predicted:yes
Actual:no          1587           172
Actual:yes           96           145
```

Figure 17.

Accuracy : 0.866
Precision : 0.4574
Recall : 0.6017
F1 Score : 0.5197

<u>Comparison of IG and GINI with Naïve Bayes:</u>
Without withholding any attribute (80:20):

When we compare the results produced by GINI and IG with Naïve Bayes, we notice that the accuracy of naïve bayes is slightly less (0.03/3%) than the accuracy of the GINI and IG. Moreover, it is also noticeable that the recall of the naïve bayes is slightly more (0.3) than the recall of GINI and IG. In addition, the precision of naïve bayes is significantly lower (0.11/11%) than the precision of GINI and IG. So, the F1 Score of naïve bayes is also less than GINI and IG.

Figure 18.

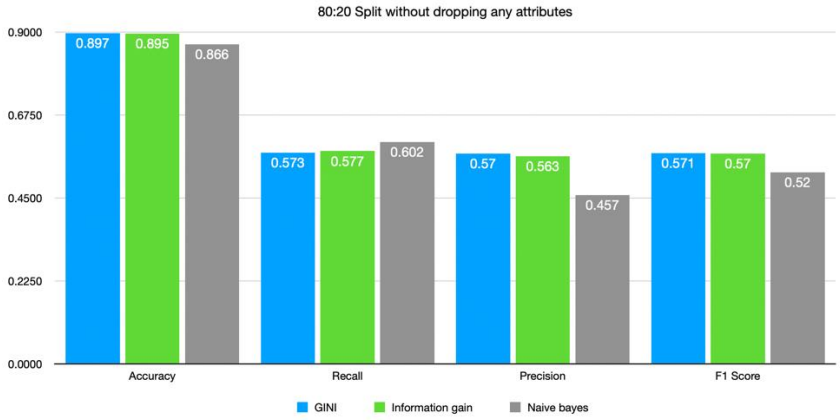


Table 1

	GINI	Information gain	Naive bayes
Accuracy	0.8965	0.895	0.866
Recall	0.5726	0.5768	0.6017
Precision	0.5702	0.5628	0.4574
F1 Score	0.5714	0.5697	0.5197

Without withholding any attribute (50:50):

While comparing the results of GINI and IG with Naïve Bayes for 50:50 split we notice the same pattern as of 80:20 split. But, with a little increase in the difference between recall value of GINI-IG and Naïve Bayes.

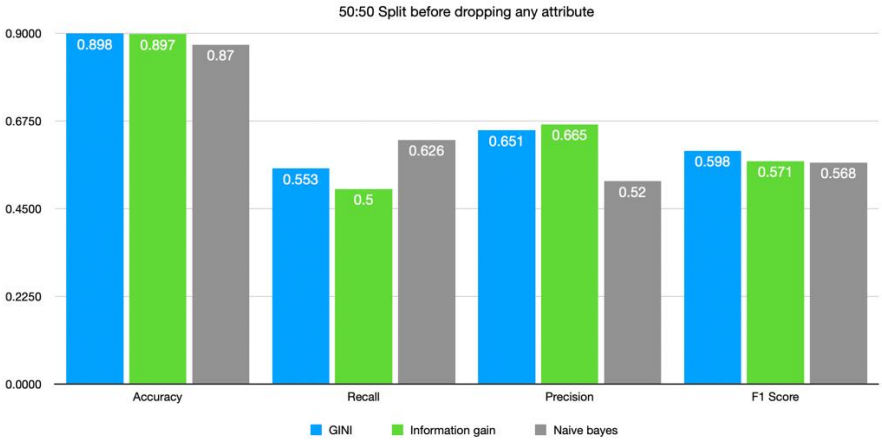


Table 1

	GINI	Information gain	Naive bayes
Accuracy	0.8982	0.8972	0.8698
Recall	0.5526	0.5	0.6257
Precision	0.6506	0.6654	0.5200
F1 Score	0.5976	0.5709	0.5680

Figure 19.

Withholding duration attribute (80:20):

When we observe the data after dropping duration attribute, we notice a huge change in the GINI and IG section (recall decreases drastically, precision increases significantly and F1 score decreases significantly). But, Naïve Bayes does not show much of a change in the accuracy, precision, recall and F1 Score. Thus, showing us that the duration attribute has very little importance for naïve bayes as compared to the importance it has for GINI and IG (most important attribute).

Same is the case with 50:50 split

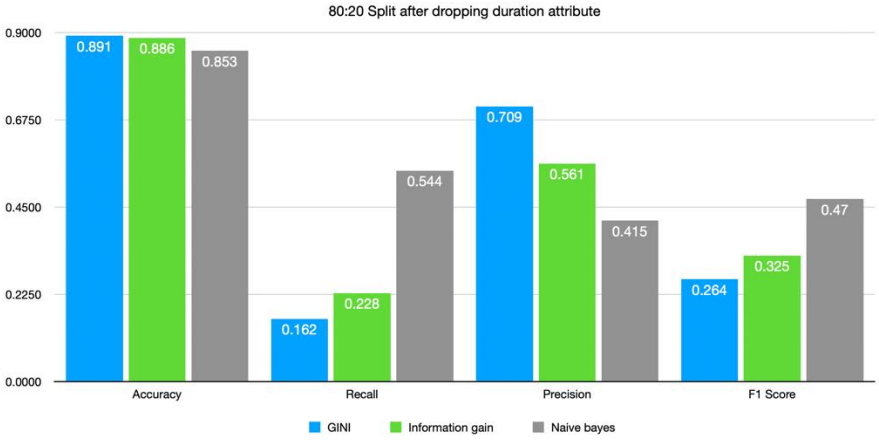


Table 1

	GINI	Information gain	Naive bayes
Accuracy	0.891	0.8855	0.8525
Recall	0.1618	0.2282	0.5436
Precision	0.7091	0.5612	0.4146
F1 Score	0.2635	0.3245	0.4704

Figure 20.

Withholding days_of_week attribute (80:20):

Naïve Bayes shows the similar effect to GINI and IG after deleting the days_of_week attribute. That is there is change near to NIL with or without the attribute days_of_week. So, Naïve Bayes, GINI and IG models are not dependent on days_of_week attribute for 80:20 split.

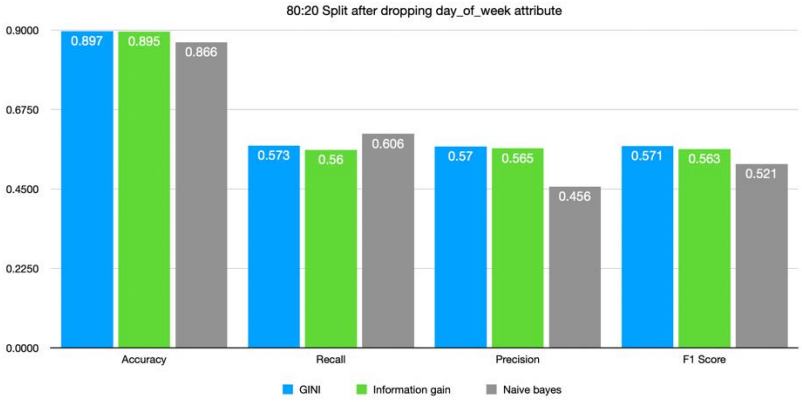


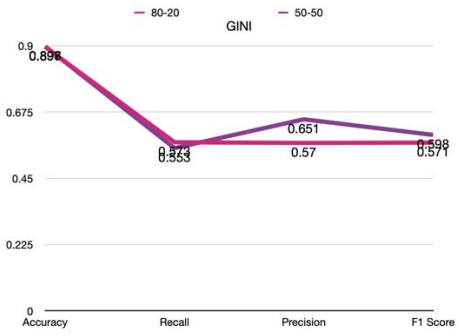
Table 1

	GINI	Information gain	Naive bayes
Accuracy	0.8965	0.895	0.8655
Recall	0.5726	0.5602	0.6058
Precision	0.5702	0.5646	0.4563
F1 Score	0.5714	0.5625	0.5205

Figure 21.

Analysis of training/test splits:

There are two splits in the data set that are taken:
GINI: We are using the two splits 80:20 and 50:50. When we observe the graph below we notice that the accuracy of both splits is almost same. But, there is significant difference between the precision of two splits (50:50 being higher) this means that because the data is imbalanced the accuracy of minority class ('yes') in 50:50 is higher as compared to 80:20 Split.

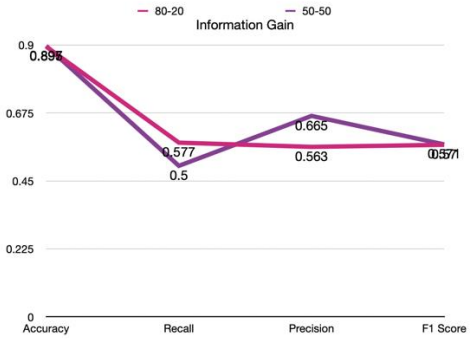


GINI

	80-20	50-50
Accuracy	0.8965	0.8982
Recall	0.5726	0.5526
Precision	0.5702	0.6506
F1 Score	0.5714	0.5976

Figure 22

IG: When we look at the graph below we notice that the recall of 50:50 split is significantly lower than the recall of 80:20 split this means that the coverage of minority class (as the data set is imbalanced minority class is 'yes') is more in the case 80:20 split. Moreover, the precision of 50:50 is around 0.10 higher than 80:20 which says that the accuracy of minority class is higher in the case of 50:50 split as compared to 80:20 split. As we go further, we notice that the F1 score of both splits are almost same and similar is the case with accuracy.

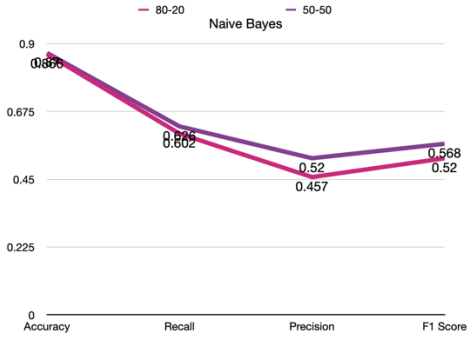


Information Gain

	80-20	50-50
Accuracy	0.895	0.8972
Recall	0.5768	0.5
Precision	0.5628	0.6654
F1 Score	0.5697	0.5709

Figure 23.

Naïve Bayes: In case of naïve bayes accuracy of both splits are same but the recall, precision and F1 score of 50:50 split is higher than the 80:20 split. From this we can infer that, 50:50 split gives better accuracy and better coverage of minority class as compared to 80:20 split.



Naive Bayes

	80-20	50-50
Accuracy	0.866	0.8698
Recall	0.6017	0.6257
Precision	0.4574	0.5200
F1 Score	0.5197	0.5680

Figure 24.

Analysis of dropping attributes:

Duration: Duration is the most important attribute of the data set with highest variable importance (vi (GINI=373.6002, IG=386.3308)) for GINI and IG. That is the reason when we drop the attribute duration there is a drastic drop seen in the recall and F1 score of the GINI and IG in both the splits as shown in figures 7 and 10. But, in the case of naïve bayes not much of a change was seen in any of the values because of its probabilistic approach to classify. That means duration was not an important attribute in naïve bayes classification.

Day_of_week: In contrast day_of_week is the least important attribute in the data set with least variable importance for GINI and fourth last for IG (vi (GINI=0.9776, IG= 10.1780)). As expected, when we dropped this attribute literally no change was seen in values of GINI but, an insignificant change was seen in IG (refer to figure 13). In terms of Naïve bayes change in the values was next to NIL. We can drop this attribute successfully.

Euribor3m: Euribor3m is the third most important attribute for both GINI and IG(vi (GINI=304.844, IG=309.216)). But, when we drop this attribute no significant change was seen in F1 score, accuracy and precision (accuracy is never effected because the given data set is skewed towards 'no') but recall of GINI, IG and naïve bayes was dropped by 6-8% (0.06-0.08) marking the importance of this attribute to cover the minority class that is 'yes' in our case.

Discussion of 3 problems encountered and solved:

1) Not having enough domain knowledge was one of the main problems that I faced in this dataset. It was a learning curve for me which I resolved by going through different banking related articles for starters on internet.

2) Not having a project mate was challenging for me, I overcame that by working double time. Moreover, I arrived late to the US due to covid so, catching up with the class was a task.

3) I had no knowledge about how to use R language. With this project I have learned that and I think I am ready for the upcoming tasks.