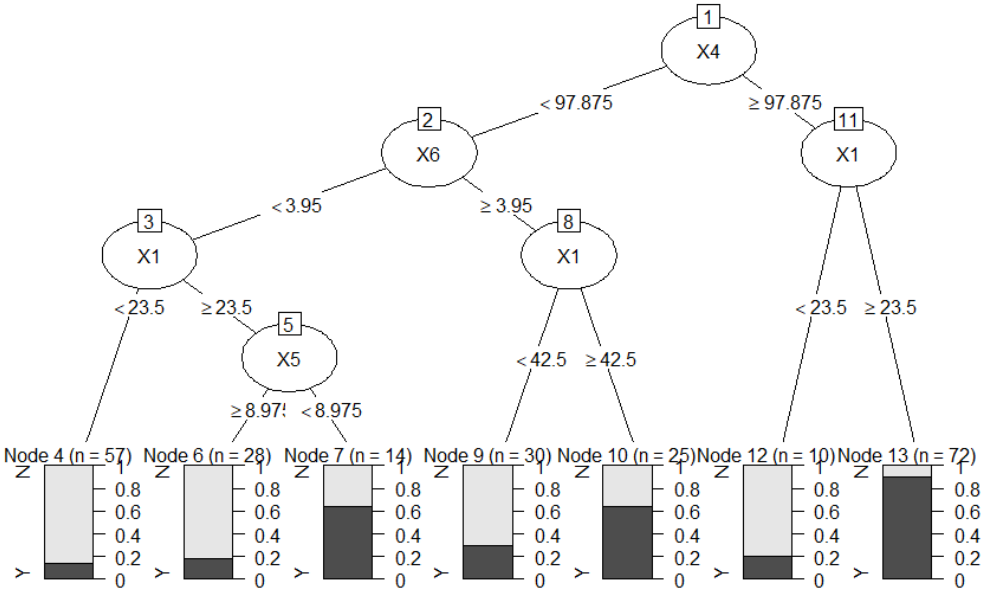
Project Report for CA 2

By Arambakam Mukesh – 19301497

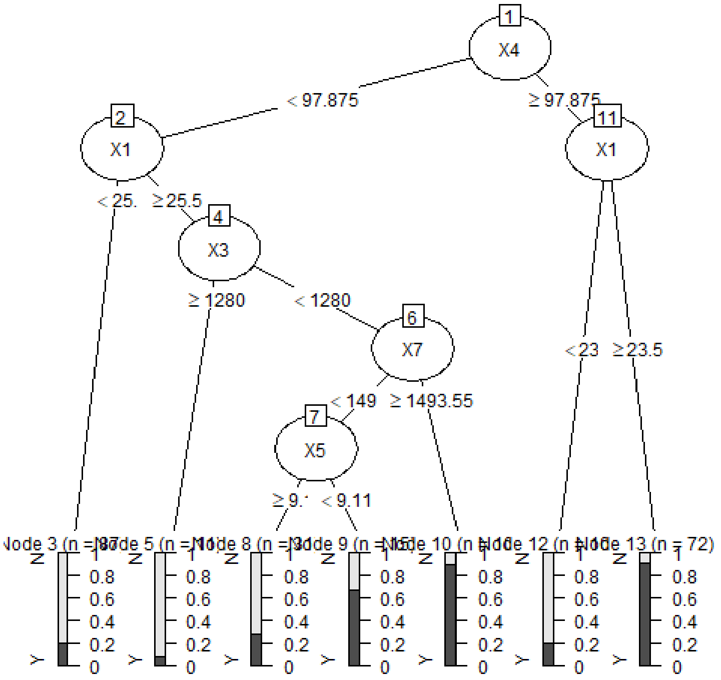
The code for this can be found on my GitHub, please find the link to the repo below.

https://github.com/mukeshmk/r-project

Plot A

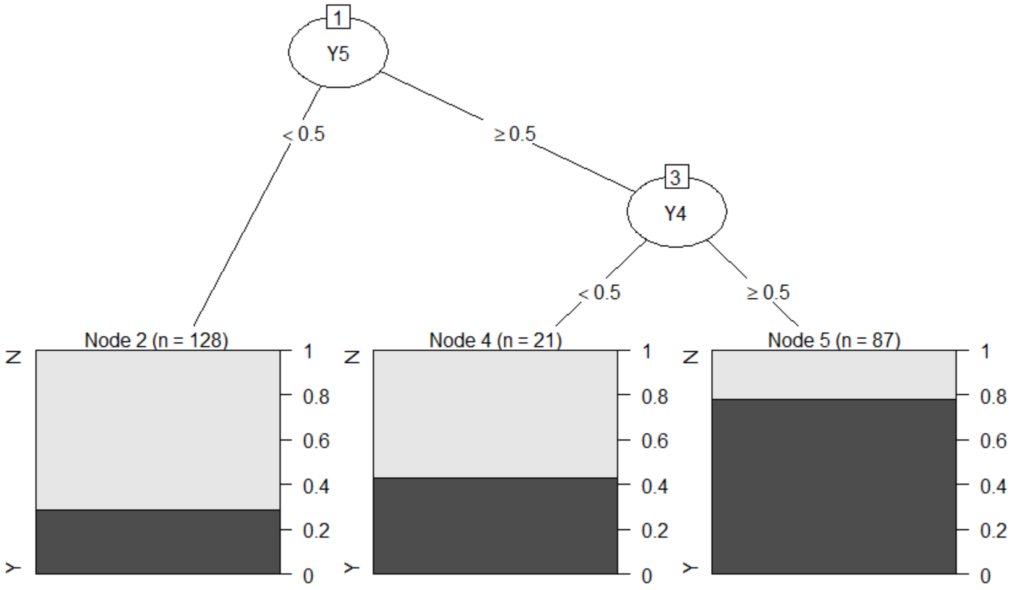


Plot B

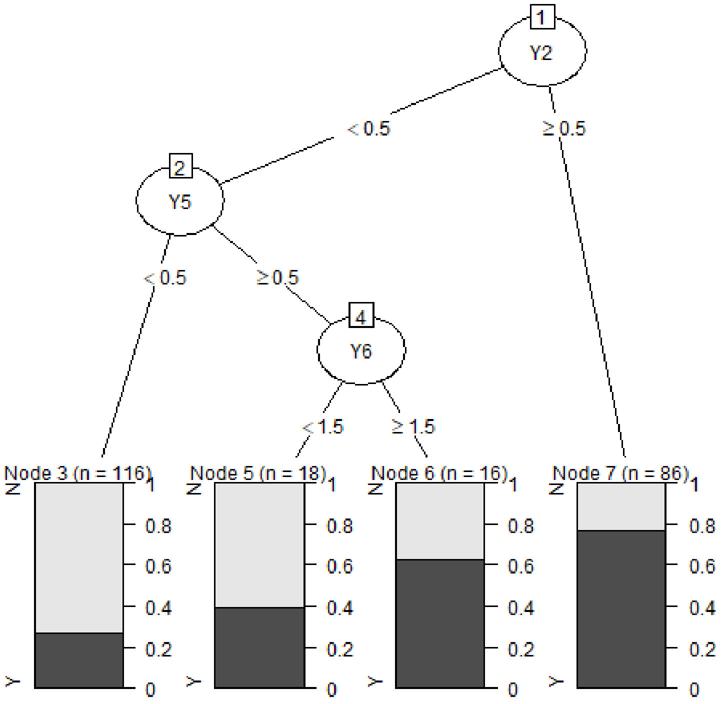


The above plots represent the Decision Tree over the entire data set but with the Predictors though X1-X7 before (Plot A) and after (Plot B) imputation respectively. The DT in Plot A predicts with an accuracy of 80% (0.8) and the DT in Plot B predicts with an accuracy of 82% (0.82).

Plot A

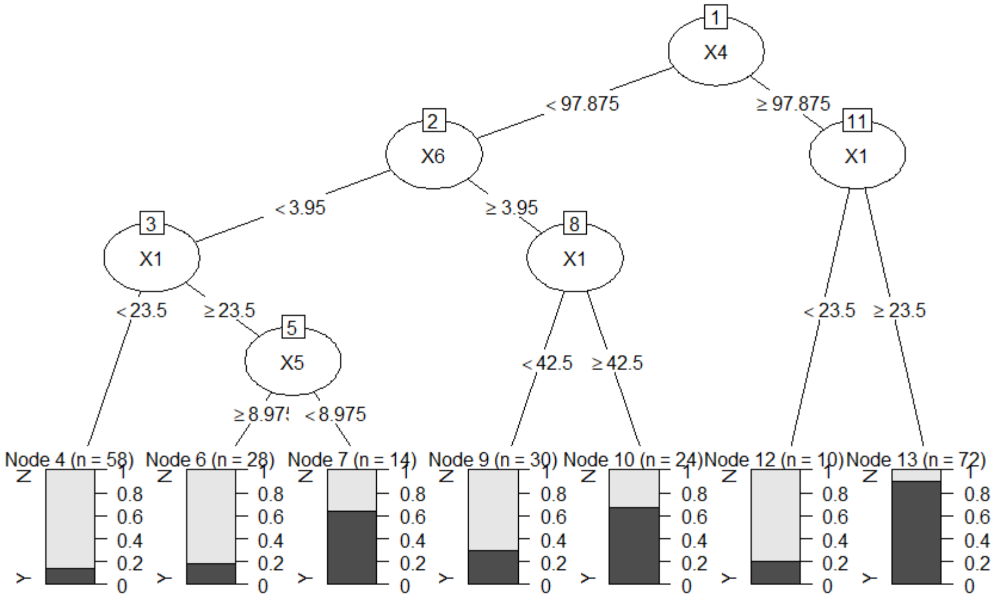


Plot B

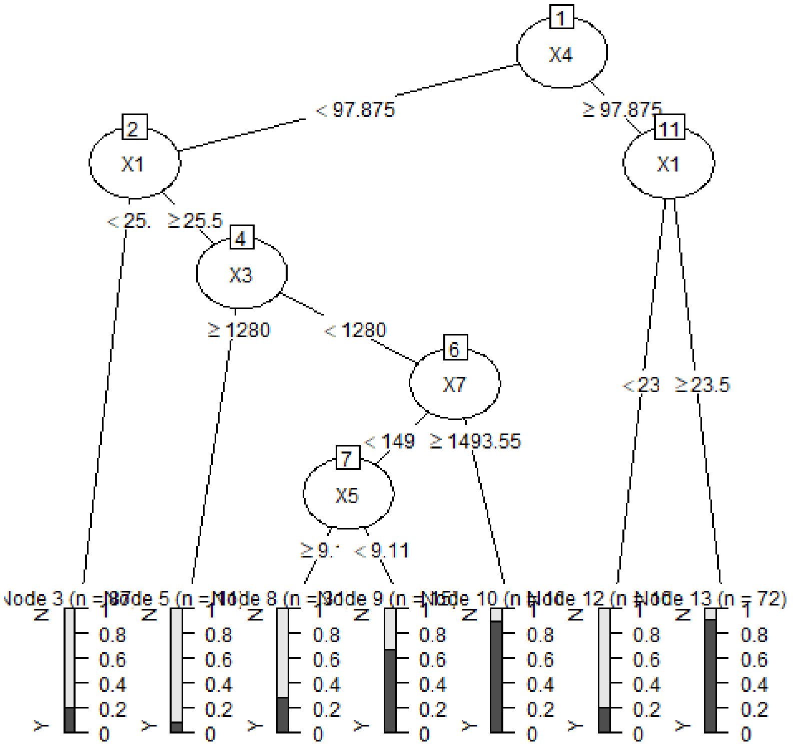


The above plots represent the Decision Tree over the entire data set but with the Predictors though Y1-Y7 before (Plot A) and after (Plot B) imputation respectively. The DT in Plot A predicts with an accuracy of 80% (0.8) and the DT in Plot B predicts with an accuracy of 75% (0.75).

Plot A

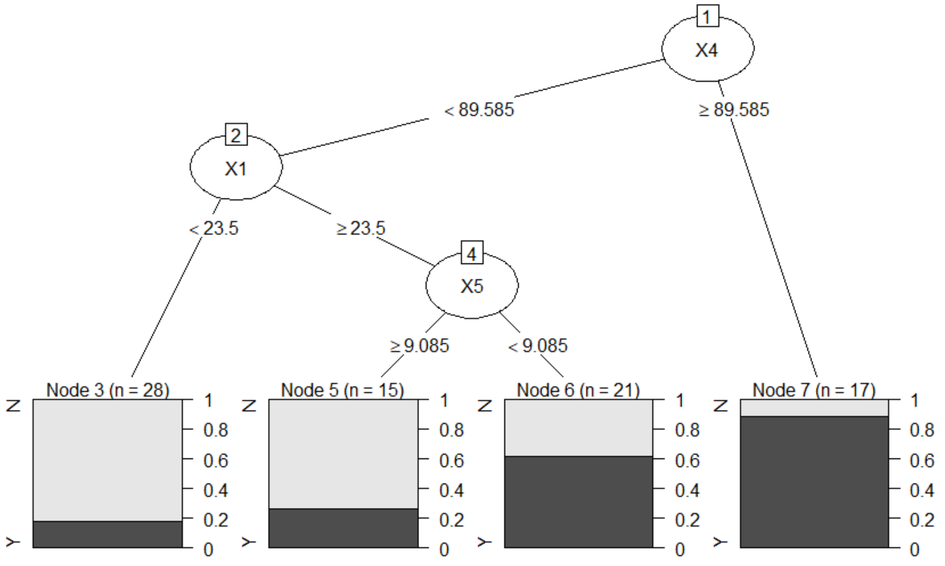


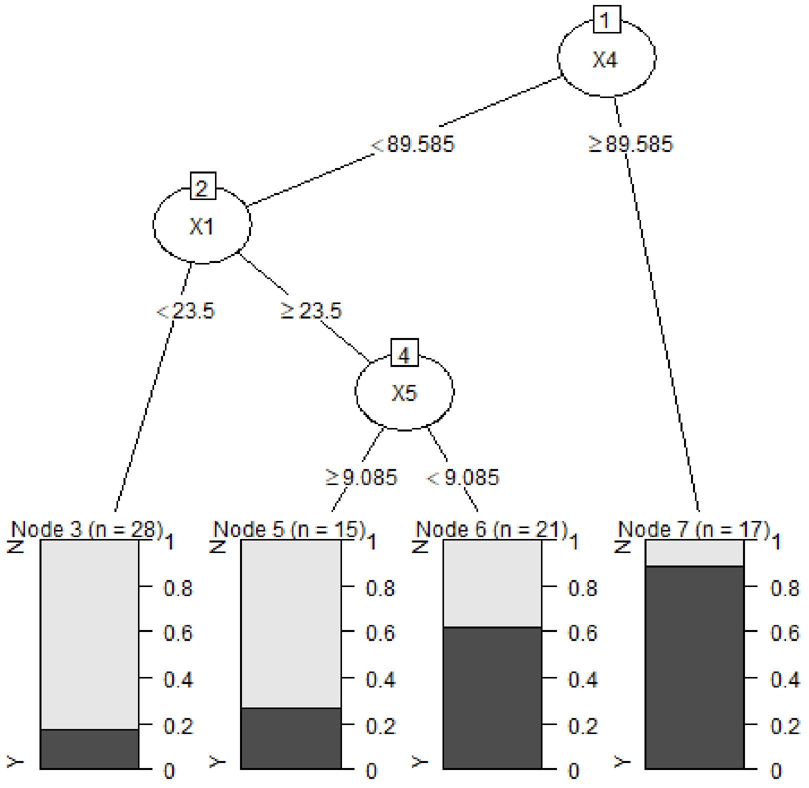
Plot B



The above plots represent the Decision Tree over the entire data set but with the Predictors though X1-X7 and Y1-Y7 before (Plot A) and after (Plot B) imputation respectively. The DT in Plot A predicts with an accuracy of 80% (0.8) and the DT in Plot B predicts with an accuracy of 82% (0.82).

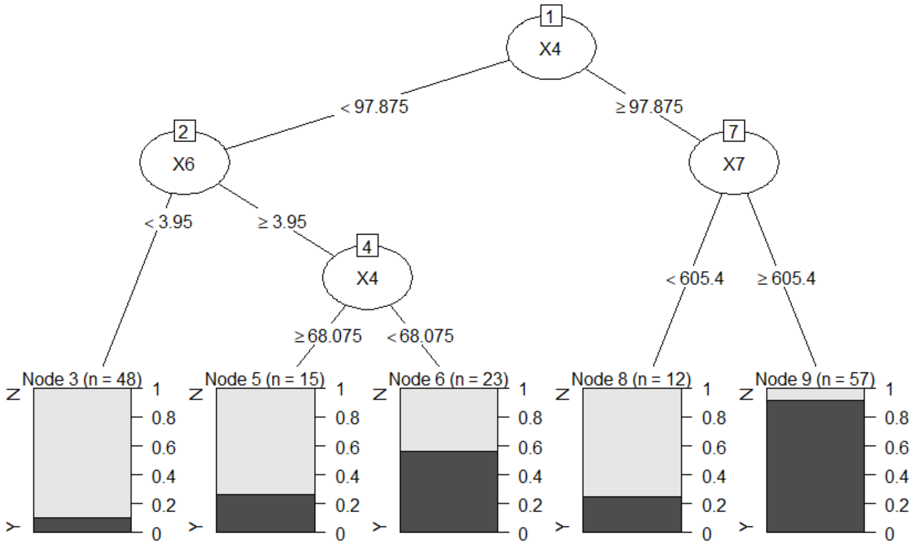
Plot A



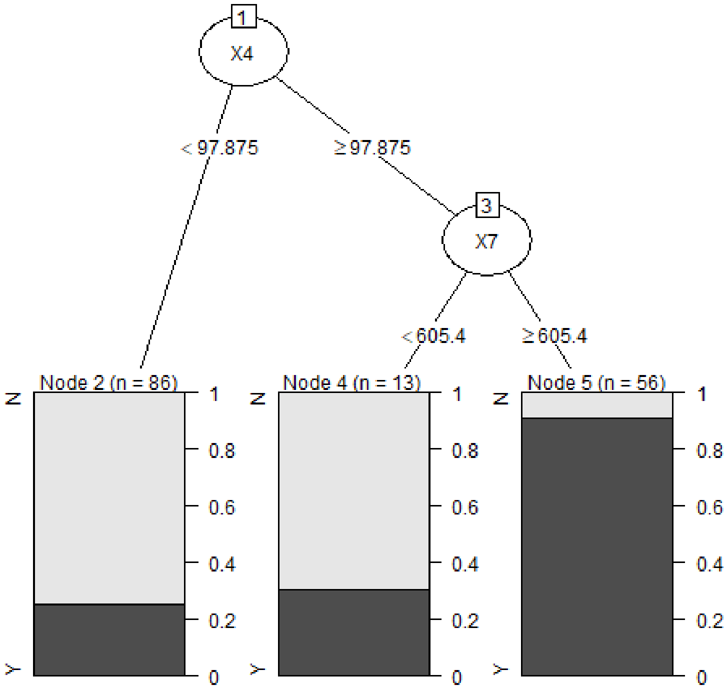


The above plots represent the Decision Tree over Group 0 set but with the Predictors though X1-X7 before (Plot A) and after (Plot B) imputation respectively. The DT in Plot A predicts with an accuracy of 77% (0.77) and the DT in Plot B predicts with an accuracy of 77% (0.77).

Plot A

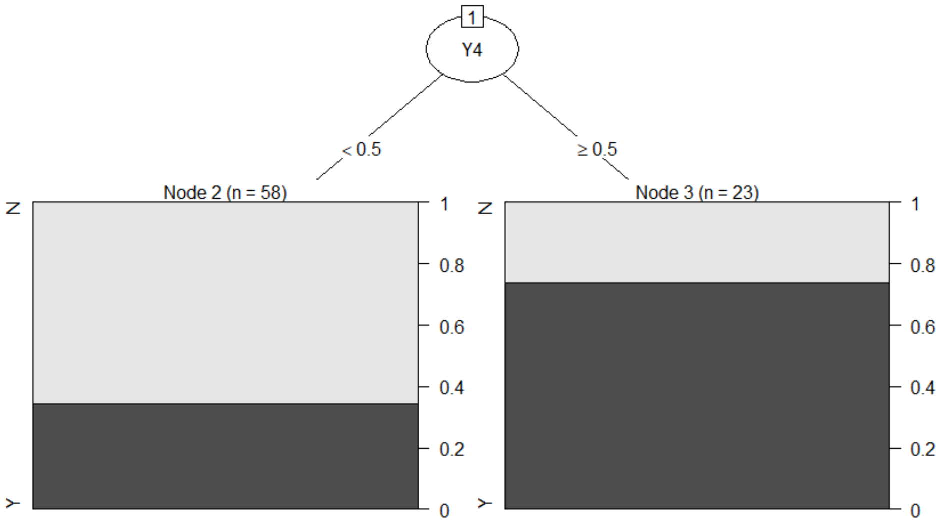


Plot B

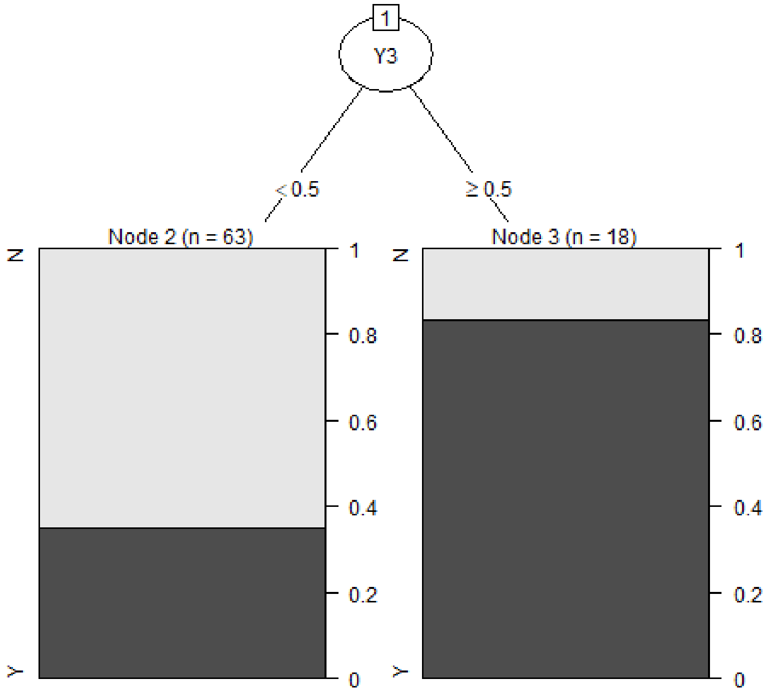


The above plots represent the Decision Tree over Group 1 set but with the Predictors though X1-X7 before (Plot A) and after (Plot B) imputation respectively. The DT in Plot A predicts with an accuracy of 78% (0.78) and the DT in Plot B predicts with an accuracy of 82% (0.82).

Plot A

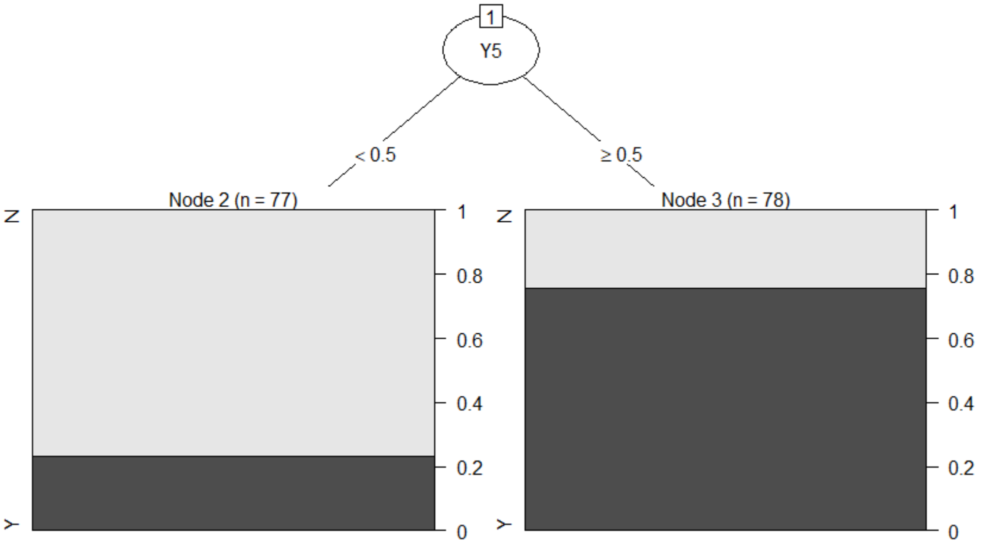


Plot B

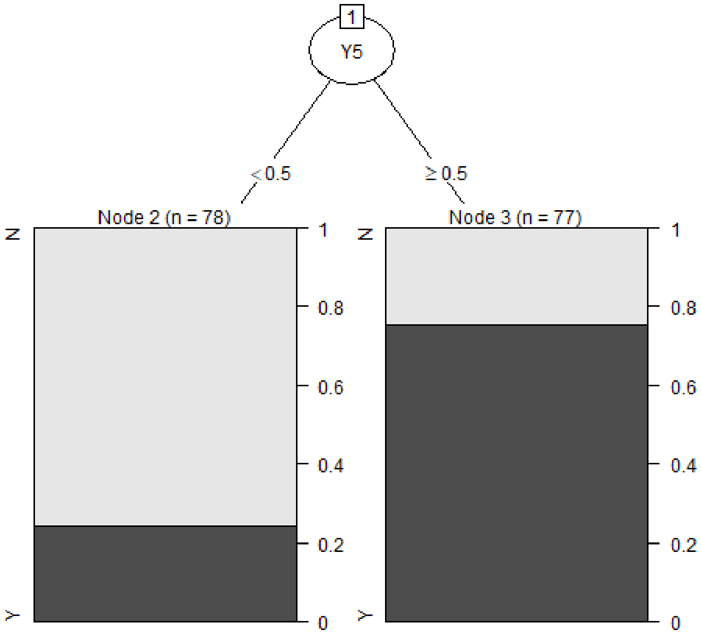


The above plots represent the Decision Tree over Group 0 set but with the Predictors though Y1-Y7 before (Plot A) and after (Plot B) imputation respectively. The DT in Plot A predicts with an accuracy of 72% (0.72) and the DT in Plot B predicts with an accuracy of 65% (0.65).

Plot A

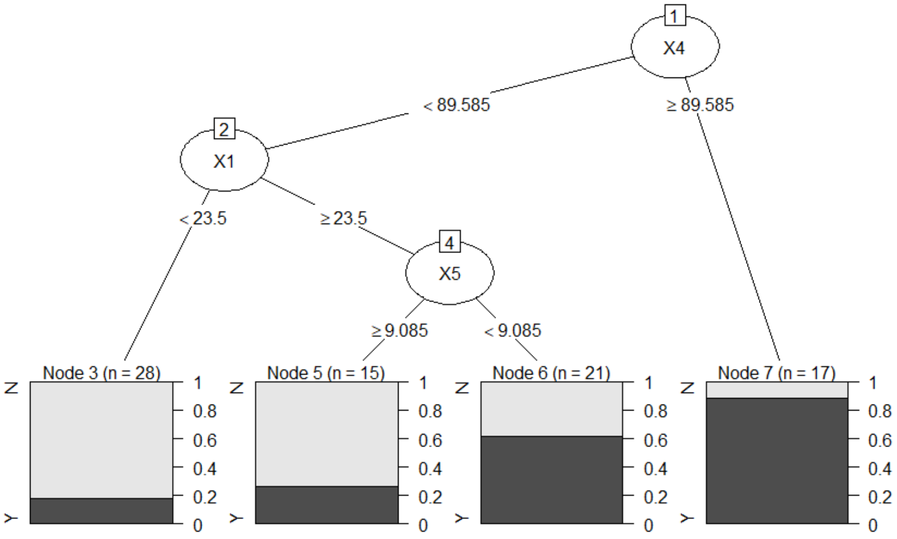


Plot B

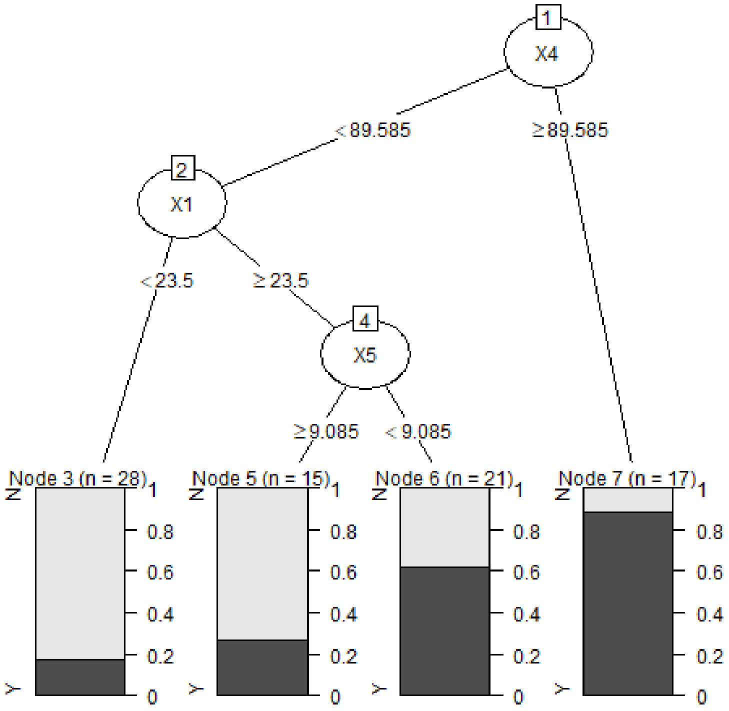


The above plots represent the Decision Tree over Group 1 set but with the Predictors though Y1-Y7 before (Plot A) and after (Plot B) imputation respectively. The DT in Plot A predicts with an accuracy of 75% (0.75) and the DT in Plot B predicts with an accuracy of 75% (0.75).

Plot A

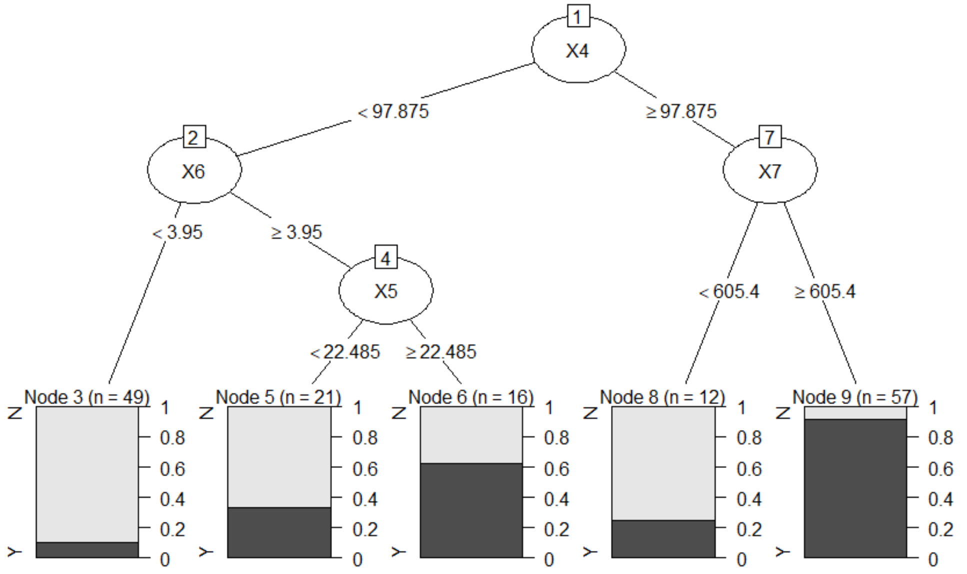


Plot B

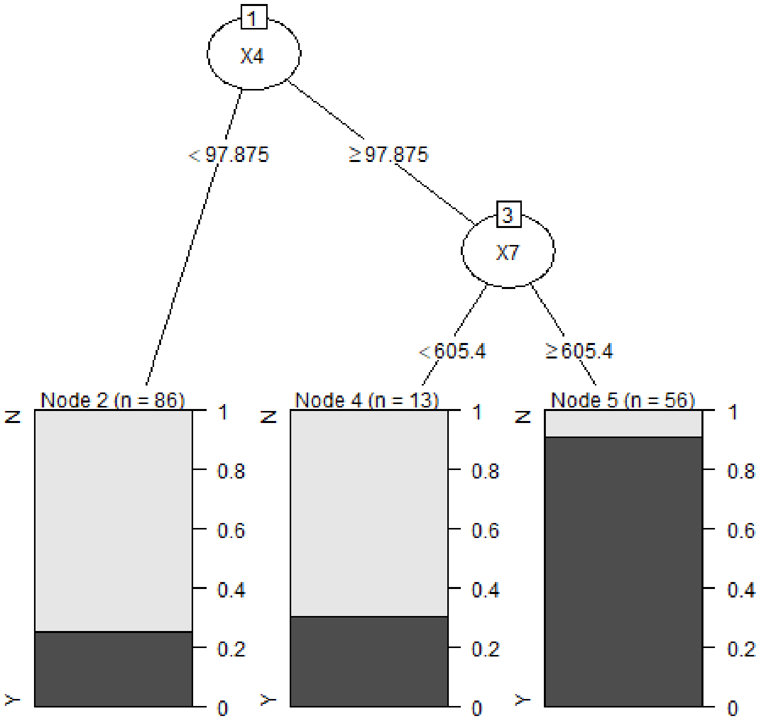


The above plots represent the Decision Tree over Group 0 set but with the Predictors though X1-X7 and Y1-Y7 before (Plot A) and after (Plot B) imputation respectively. The DT in Plot A predicts with an accuracy of 77% (0.77) and the DT in Plot B predicts with an accuracy of 77% (0.77).

Plot A



Plot B



The above plots represent the Decision Tree over Group 1 set but with the Predictors though X1-X7 and Y1-Y7 before (Plot A) and after (Plot B) imputation respectively. The DT in Plot A predicts with an accuracy of 80% (0.8) and the DT in Plot B predicts with an accuracy of 82% (0.82).

Conclusion:

The best Decision Tree generated is the DT generated over **Group 1 with the Predictors X1-X7 and Y1-Y7** as it has the highest **accuracy of 82%**. The below is the Decision Tree's summary, indicating the splits:

```
n= 155

node), split, n, loss, yval, (yprob)
      * denotes terminal node

1) root 155 77 N (0.50322581 0.49677419)  
  2) X4< 97.875 86 22 N (0.74418605 0.25581395) *
  3) X4>=97.875 69 14 Y (0.20289855 0.79710145)  
    6) X7< 605.4 13  4 N (0.69230769 0.30769231) *
    7) X7>=605.4 56  5 Y (0.08928571 0.91071429) *
```

Though the DT's in Plot 1 and Plot 3 also have an accuracy of 82% - Plot 9 is better because it gives a consistent accuracy of 82% when tested with different `rpart` configurations like minsplit, minbucket and maxdepth, more over it has a **Low Variance** between different decision tree when run with different seed values for the data and different rpart configuration.

The code for this can be found on my GitHub, please find the link to the repo below.

https://github.com/mukeshmk/r-project