Lab 2

Damian Ke & Kyriakos Papadopoulos

2022-11-29

Assignment 2. Decision trees and logistic regression for bank marketing

Question 1:

Import the data to R, remove variable "duration" and divide into training/validation/test as 40/30/30: use data partitioning code specified in Lecture 2a.

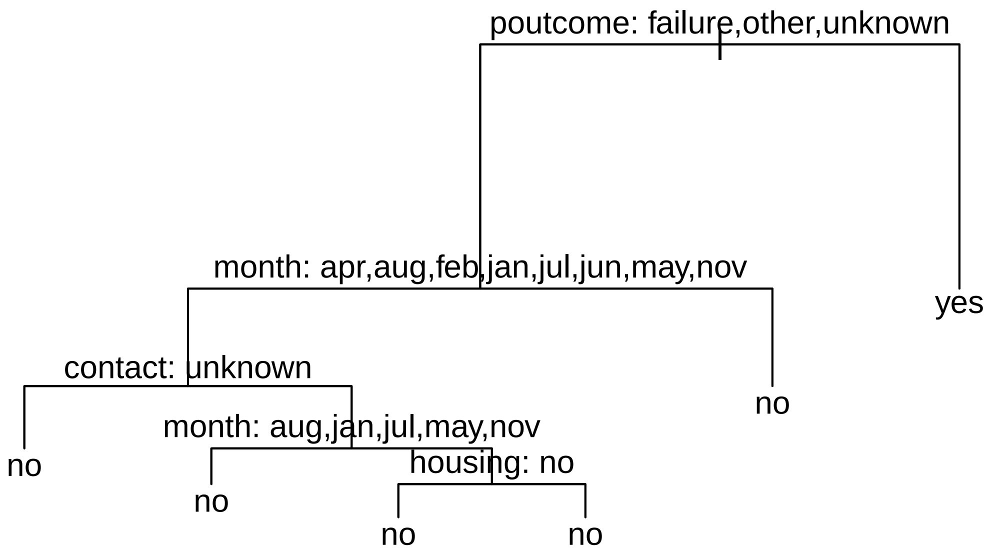
```
data = read.csv2("bank-full.csv",stringsAsFactors = TRUE)
#2.1
data = subset(data, select = -c(duration))
n <- dim(data)[1]
set.seed(12345)
id <- sample(1:n, floor(n*0.4))
train <- data[id,]

id1 <- setdiff(1:n, id)
set.seed(12345)
id2 <- sample(id1, floor(n*0.3))
validation <- data[id2,]

id3 <- setdiff(id1,id2)
test <- data[id3,]
```

Question 2:

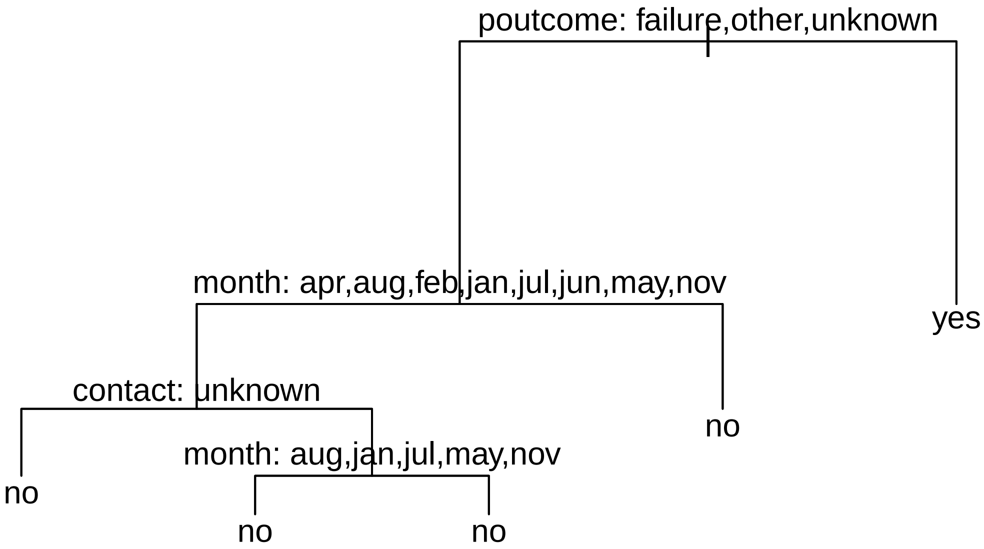
Fit decision trees to the training data so that you change the default settings one by one (i.e. not simultaneously): a. Decision Tree with default settings. b. Decision Tree with smallest allowed node size equal to 7000. c. Decision trees minimum deviance to 0.0005. and report the misclassification rates for the training and validation data. Which model is the best one among these three? Report how changing the deviance and node size affected the size of the trees and explain why.

```
## [1] "Default settings trees, Training missclassification:0.104844061048441"
```

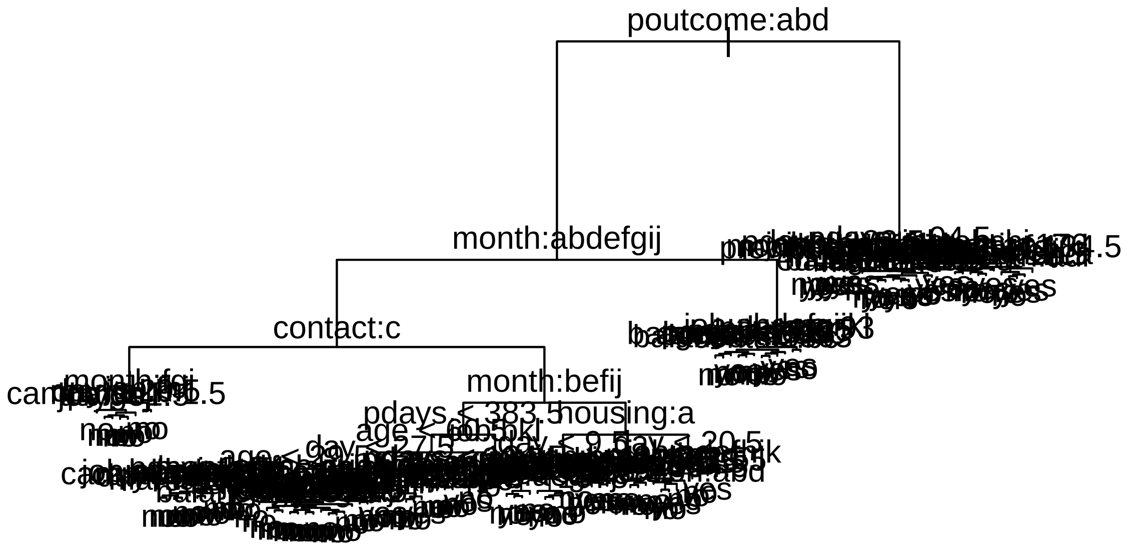
```
## [1] "Default settings trees, Validation missclassification:0.109267861092679"
```


poutcome: failure,other,unknown
month: apr,aug,feb,jan,jul,jun,may,nov
yes
contact: unknown
no
month: aug,jan,jul,may,nov
housing: no
no
no
no
no

```
##
## Classification tree:
## tree(formula = y ~ ., data = train)
## Variables actually used in tree construction:
## [1] "poutcome" "month"    "contact"  "housing"
## Number of terminal nodes:  6
## Residual mean deviance:  0.6022 = 10890 / 18080
## Misclassification error rate: 0.1048 = 1896 / 18084

## [1] "Smallest allowed node size trees, Training missclassification:0.104844061048441"

## [1] "Smallest allowed node size trees, Validation missclassification:0.109267861092679"
```


poutcome: failure,other,unknown
month: apr,aug,feb,jan,jul,jun,may,nov
yes
contact: unknown
no
month: aug,jan,jul,may,nov
no
no
no

```
##
## Classification tree:
## tree(formula = y ~ ., data = train, minsize = 7000)
## Variables actually used in tree construction:
## [1] "poutcome" "month"    "contact"
## Number of terminal nodes:  5
## Residual mean deviance:  0.6097 = 11020 / 18080
## Misclassification error rate: 0.1048 = 1896 / 18084

## [1] "Deviance trees, Training missclassification:0.0940057509400575"

## [1] "Deviance trees, Validation missclassification:0.111922141119221"
```

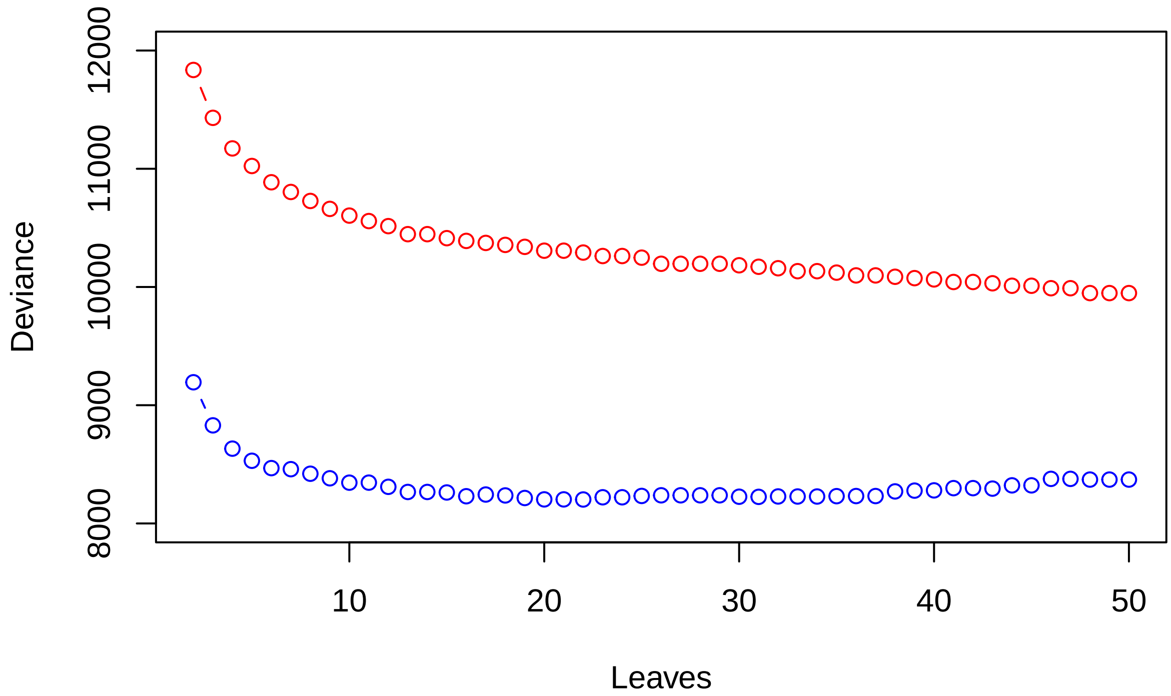


```
##
## Classification tree:
## tree(formula = y ~ ., data = train, mindev = 5e-04)
## Variables actually used in tree construction:
##  [1] "poutcome"  "month"     "contact"   "marital"   "day"       "campaign"
##  [7] "job"       "pdays"     "age"       "balance"   "housing"   "education"
## [13] "previous"
## Number of terminal nodes:  122
## Residual mean deviance:  0.5213 = 9363 / 17960
## Misclassification error rate: 0.09362 = 1693 / 18084
```

Answer Decision trees minimum deviance to 0.0005 gave the lowest missclassification rate for train data. Lowest validation error was given by default setting tree and tree with smallest allowed node. In addition both settings gave same missclassification error for both validation and train data. As it can be seen in the figure and summary. Min deviance has highest number of terminal nodes, equal to 122. This resulted in a biggest tree size. Thereafter, the tree with default settings has 6 terminal nodes and is 2nd largest. Lastly, min node size has 5 terminal nodes and is the smallest tree. It is important to examine both figure and number of terminal nodes as the tree can be unbalanced and have increased depth. The size of tree of min deviance can be due to minimal number of error of each node. This rule forces the tree to increase in the size and have more nodes. The opposite could be find with min node size, as all leaves have required number of size and it can be seen that is why it is a smallest tree.

Question 3.

Use training and validation sets to choose the optimal tree depth in the model 2c: study the trees up to 50 leaves. Present a graph of the dependence of deviances for the training and the validation data on the number of leaves and interpret this graph in terms of bias-variance tradeoff. Report the optimal amount of leaves and which variables seem to be most important for decision making in this tree. Interpret the information provided by the tree structure (not everything but most important findings).



```
## node), split, n, deviance, yval, (yprob)
##       * denotes terminal node
##
##    1) root 18084 12850.00 no ( 0.88576 0.11424 )
##      2) poutcome: failure,other,unknown 17468 11030.00 no ( 0.90422 0.09578 )
##        4) month: apr,aug,feb,jan,jul,jun,may,nov 16828  9772.00 no ( 0.91520 0.08480 )
##          8) contact: unknown 5130  1599.00 no ( 0.96374 0.03626 )
##           16) month: jul,jun,may 5074  1502.00 no ( 0.96610 0.03390 ) *
##           17) month: apr,aug,feb,jan,nov 56    62.98 no ( 0.75000 0.25000 ) *
##          9) contact: cellular,telephone 11698  7914.00 no ( 0.89391 0.10609 )
##           18) month: aug,jan,jul,may,nov 9284  5503.00 no ( 0.91265 0.08735 )
##             36) pdays < 383.5 9246  5373.00 no ( 0.91510 0.08490 )
##               72) age < 60.5 9097  5107.00 no ( 0.91920 0.08080 )
##                144) day < 27.5 7670  4588.00 no ( 0.91147 0.08853 )
##                  288) age < 29.5 754   637.10 no ( 0.85013 0.14987 )
##                    576) month: jan,jul,may 681   528.00 no ( 0.86931 0.13069 ) *
##                    577) month: aug,nov 73    92.46 no ( 0.67123 0.32877 ) *
##                  289) age > 29.5 6916  3918.00 no ( 0.91816 0.08184 )
##                    578) balance < 1493 5180  2635.00 no ( 0.92973 0.07027 )
##                     1156) month: aug,jul,may,nov 5164  2596.00 no ( 0.93087 0.06913 ) *
##                     1157) month: jan 16    21.93 no ( 0.56250 0.43750 ) *
##                    579) balance > 1493 1736  1249.00 no ( 0.88364 0.11636 ) *
##                145) day > 27.5 1427   472.40 no ( 0.96076 0.03924 )
##                   290) pdays < 184.5 1308   462.50 no ( 0.95719 0.04281 )
##                     580) pdays < 80.5 1277   402.60 no ( 0.96319 0.03681 ) *
##                     581) pdays > 80.5 31    37.35 no ( 0.70968 0.29032 ) *
##                   291) pdays > 184.5 119     0.00 no ( 1.00000 0.00000 ) *
##                73) age > 60.5 149   190.10 no ( 0.66443 0.33557 ) *
##              37) pdays > 383.5 38    47.40 yes ( 0.31579 0.68421 ) *
##            19) month: apr,feb,jun 2414  2262.00 no ( 0.82187 0.17813 )
##              38) housing: no 1123  1321.00 no ( 0.72484 0.27516 )
##                76) day < 9.5 691   668.20 no ( 0.81187 0.18813 )
##                 152) month: feb 505   364.20 no ( 0.88317 0.11683 ) *
##                 153) month: apr,jun 186   247.30 no ( 0.61828 0.38172 ) *
##                77) day > 9.5 432   586.10 no ( 0.58565 0.41435 ) *
##              39) housing: yes 1291   803.20 no ( 0.90627 0.09373 )
##                78) day < 20.5 1210   655.90 no ( 0.92314 0.07686 )
##                 156) month: apr,feb 1154   565.70 no ( 0.93328 0.06672 ) *
##                 157) month: jun 56    67.01 no ( 0.71429 0.28571 ) *
##                79) day > 20.5 81   104.40 no ( 0.65432 0.34568 ) *
##         5) month: dec,mar,oct,sep 640   852.70 no ( 0.61562 0.38438 ) *
##       3) poutcome: success 616   806.40 yes ( 0.36201 0.63799 )
##         6) pdays < 94.5 170   185.50 yes ( 0.23529 0.76471 ) *
##         7) pdays > 94.5 446   603.90 yes ( 0.41031 0.58969 )
##          14) job: admin.,blue-collar,entrepreneur,services,technician 213   295.20 no ( 0.50704 0.4929
##          15) job: housemaid,management,retired,self-employed,student,unemployed,unknown 233   292.80 y
```

Answer: For bias-variance trade off, it can be seen x-axis as model complexity. For higher number of leaves, the model get more complex. Deviance, is defined as measure of goodness of fit and can be used as an error. As model gets more complex, the deviance decreases. Although, it holds the same level at the optimal number of leaves which is around 22 leaves. Thereafter the blue points corresponding for validation, slowly increases. Therefore, it can be seen that bias decreases up till leaves 22 which is the optimal amount of leaves. Before leaves 22, the model is underfitted and after it is overfitted.

The most important variable is **poutcome** as it is one of the first nodes that is best at separating the tree and classes. Thereafter, by looking at the frequency of the nodes variables **month** and **pdays** are frequently used to split the classes.

By the provided tree it can be seen that the tree is not balanced, most of the nodes are going through one side and the depth of the tree is therefore deeper. It can be also seen that if **poutcome** is "success" the probability is around higher than 50% to classify it as yes. The opposite side with most number of observations is mostly classified as no if **poutcome** is defined as failure, other and unknown.

Question 4

Estimate the confusion matrix, accuracy and F1 score for the test data by using the optimal model from step 3. Comment whether the model has a good predictive power and which of the measures (accuracy or F1-score) should be preferred here.

```
##       pred
##          yes    no
##   yes    214  1371
##   no     107 11872
```

```
## [1] "Accuracy of the model is equal to:0.891035092892952"
```

```
## [1] "F1 score of the model is equal to:0.224554039874082"
```

Answer For F1, the score should be between 0 to 1 and higher score reflects over better predictive power. Then the F1 score of 0.22455 can be seen as a bad result. Because F1 formula ignores TN, where most of predictions were made it can be quite misleading. Therefore, F1 formula depends mostly on how imbalanced the data is of TP. For accuracy, it shows the percentage of correctness and score of 0.89103 can be seen as quite high. The preferred measure depends on the goal of the prediction and on the data. But the preferred method is accuracy as it checks TP and TN.

Question 5

Perform a decision tree classification of the test data with the following loss matrix . . . , and report the confusion matrix for the test data. Compare the results with the results from step 4 and discuss how the rates has changed and why.

```
##      fit_3_5
##         yes    no
##   yes     0  1585
##   no      0 11979
```

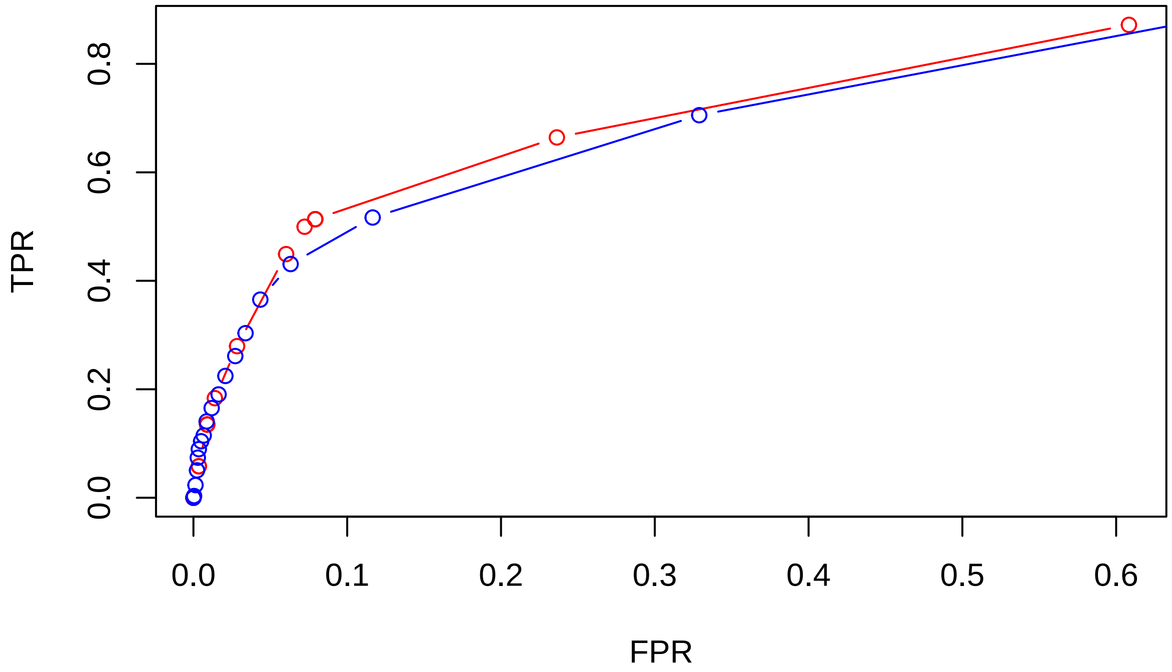
```
## [1] NaN
```

```
## [1] 0.8831466
```

Answer F1, has NaN values as most of the yes prediction are now 0. The accuracy went from 0.89103 to 0.88315, as all values from yes prediction, moved to the no prediction column. As it was described earlier, F1 ignores TN, but most of the calculations are done on yes column, which for this assignment are 0. For the accuracy as only TN and FN exists, the results are therefore TN/FN.

Question 6

Use the optimal tree and a logistic regression model to classify the test data by using the following principle: . . . Compute the TPR and FPR values for the two models and plot the corresponding ROC curves. Conclusion? Why precisionrecall curve could be a better option here?



Answer The optimal tree model as red line, performs better than logistic regression in the ROC graph. Although, both of the models have similar results. There is a delay in FPR, as the values go from around 0.15 to 1. Meanwhile for TPR the values go quickly from 0.4 to 0.9 or 1.

As it was mentioned earlier there is imbalance in the data, therefore precision-recall curve may give a better overview of the model's predictive power.