Decision Tree on Stroke Likelihood

Nathacia Nathacia

2022-08-13

```
library(tree)
dtree <- read.csv("/Users/nathacia/Desktop/r wd/bus315/stroke_data.csv")
View(dtree)
```

Loading packages and data

Investigating the data

```
class(dtree)
```

```
## [1] "data.frame"
```

```
names(dtree)
```

```
##  [1] "id"                "gender"            "age"              
##  [4] "hypertension"      "heart_disease"     "ever_married"     
##  [7] "work_type"         "Residence_type"    "avg_glucose_level"
## [10] "bmi"               "smoking_status"    "stroke"
```

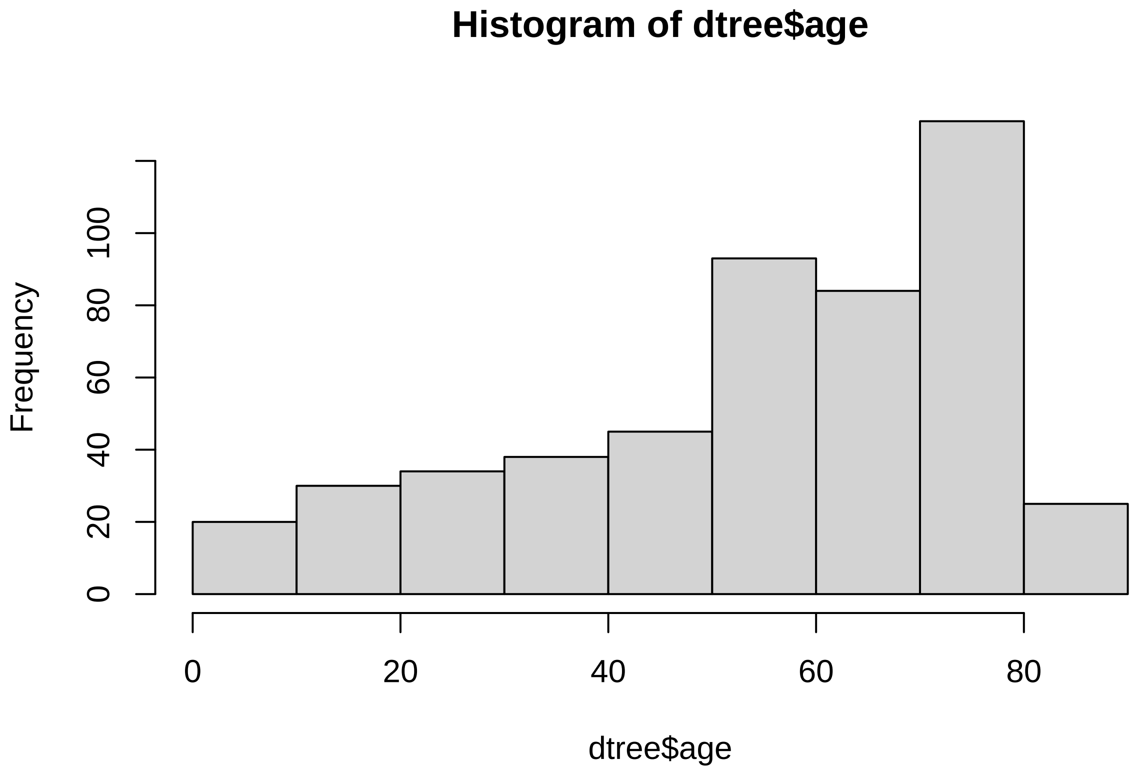
```
head(dtree)
```

```
##      id gender age hypertension heart_disease ever_married     work_type
## 1  9046   Male  67            0             1          Yes       Private
## 2 51676 Female  61            0             0          Yes Self-employed
## 3 31112   Male  80            0             1          Yes       Private
## 4 60182 Female  49            0             0          Yes       Private
## 5  1665 Female  79            1             0          Yes Self-employed
## 6 56669   Male  81            0             0          Yes       Private
##   Residence_type avg_glucose_level  bmi  smoking_status stroke
## 1          Urban            228.69 36.6 formerly smoked      1
## 2          Rural            202.21   NA    never smoked      1
## 3          Rural            105.92 32.5    never smoked      1
## 4          Urban            171.23 34.4          smokes      1
## 5          Rural            174.12 24.0    never smoked      1
## 6          Urban            186.21 29.0 formerly smoked      1
```

```
tail(dtree)
```

```
##        id gender age hypertension heart_disease ever_married work_type
## 495 29158 Female  55            0             0          Yes   Private
## 496 34299 Female  71            0             0          Yes   Private
## 497 54375   Male   5            0             0           No  children
## 498 37832 Female  14            0             0           No  children
## 499 21058 Female  15            0             0           No  children
## 500  7696 Female  66            0             0           No   Private
##     Residence_type avg_glucose_level  bmi  smoking_status stroke
## 495          Rural            111.19 39.7 formerly smoked      0
## 496          Urban             93.28 34.7    never smoked      0
## 497          Rural            122.19 35.0         Unknown      0
## 498          Rural            129.53 21.3    never smoked      0
## 499          Rural            114.53 29.1         Unknown      0
## 500          Urban             93.73 23.9          smokes      0
```

```
hist(dtree$age)
```



```
str(dtree)
```

```
## 'data.frame':    500 obs. of  12 variables:
##  $ id               : int  9046 51676 31112 60182 1665 56669 53882 10434 27419 60491 ...
##  $ gender           : chr  "Male" "Female" "Male" "Female" ...
##  $ age              : num  67 61 80 49 79 81 74 69 59 78 ...
##  $ hypertension     : int  0 0 0 0 1 0 1 0 0 0 ...
##  $ heart_disease    : int  1 0 1 0 0 0 1 0 0 0 ...
##  $ ever_married     : chr  "Yes" "Yes" "Yes" "Yes" ...
##  $ work_type        : chr  "Private" "Self-employed" "Private" "Private" ...
##  $ Residence_type   : chr  "Urban" "Rural" "Rural" "Urban" ...
##  $ avg_glucose_level: num  229 202 106 171 174 ...
##  $ bmi              : num  36.6 NA 32.5 34.4 24 29 27.4 22.8 NA 24.2 ...
##  $ smoking_status   : chr  "formerly smoked" "never smoked" "never smoked" "smokes" ...
##  $ stroke           : int  1 1 1 1 1 1 1 1 1 1 ...
```

```
summary(dtree)
```

```
##        id           gender               age         hypertension
##  Min.   :  129   Length:500         Min.   : 0.64   Min.   :0.000
##  1st Qu.:17304   Class :character   1st Qu.:42.00   1st Qu.:0.000
##  Median :36404   Mode  :character   Median :59.00   Median :0.000
##  Mean   :36707                      Mean   :55.13   Mean   :0.184
##  3rd Qu.:55712                      3rd Qu.:74.00   3rd Qu.:0.000
##  Max.   :72918                      Max.   :82.00   Max.   :1.000
##
##  heart_disease  ever_married        work_type         Residence_type
##  Min.   :0.00   Length:500         Length:500         Length:500
##  1st Qu.:0.00   Class :character   Class :character   Class :character
##  Median :0.00   Mode  :character   Mode  :character   Mode  :character
##  Mean   :0.12
##  3rd Qu.:0.00
##  Max.   :1.00
##
##  avg_glucose_level      bmi        smoking_status        stroke
##  Min.   : 55.39    Min.   :13.80   Length:500         Min.   :0.000
##  1st Qu.: 79.11    1st Qu.:25.05   Class :character   1st Qu.:0.000
##  Median : 97.84    Median :28.90   Mode  :character   Median :0.000
##  Mean   :121.85    Mean   :30.13                      Mean   :0.498
##  3rd Qu.:167.20    3rd Qu.:34.40                      3rd Qu.:1.000
##  Max.   :271.74    Max.   :64.80                      Max.   :1.000
##                    NA's   :45
```

I analyzed the data by using various inspecting functions to get a better idea of the dataset. I used the histogram function to take a closer look at the age variable and from there, I can see that the data included mostly older people, particularly those between the age ranges of 50-80 years old. This made a lot of sense as stroke usually occurs within those age ranges.

Preprocessing

```
dtree$bmi <- as.numeric(dtree$bmi)
dtree <- na.omit(dtree)
dtree$gender <- as.factor(dtree$gender)
dtree$ever_married <- as.factor(dtree$ever_married)
dtree$work_type <- as.factor(dtree$work_type)
dtree$Residence_type <- as.factor(dtree$Residence_type)
dtree$smoking_status <- as.factor(dtree$smoking_status)
stroke_likelihood <- ifelse(dtree$stroke>=1, "Yes", "No")
stroke_likelihood <- as.factor(stroke_likelihood)
dtree <- data.frame(dtree, stroke_likelihood)
dtree <- subset(dtree, select=-stroke)
dtree <- subset(dtree, select=-ever_married)
dtree <- subset(dtree, select=-id)
dtree <- subset(dtree, select=-work_type)
dtree <- subset(dtree, select=-Residence_type)
summary(dtree)
```

```
##     gender          age          hypertension    heart_disease
##  Female:276   Min.   : 0.64   Min.   :0.0000   Min.   :0.0000
##  Male  :179   1st Qu.:39.00   1st Qu.:0.0000   1st Qu.:0.0000
##               Median :58.00   Median :0.0000   Median :0.0000
##               Mean   :53.99   Mean   :0.1846   Mean   :0.1143
##               3rd Qu.:73.00   3rd Qu.:0.0000   3rd Qu.:0.0000
##               Max.   :82.00   Max.   :1.0000   Max.   :1.0000
##  avg_glucose_level      bmi                smoking_status stroke_likelihood
##  Min.   : 55.39    Min.   :13.80   formerly smoked: 95    No :246
##  1st Qu.: 79.16    1st Qu.:25.05   never smoked   :176    Yes:209
##  Median : 97.55    Median :28.90   smokes         : 78
##  Mean   :121.47    Mean   :30.13   Unknown        :106
##  3rd Qu.:161.71    3rd Qu.:34.40
##  Max.   :271.74    Max.   :64.80
```

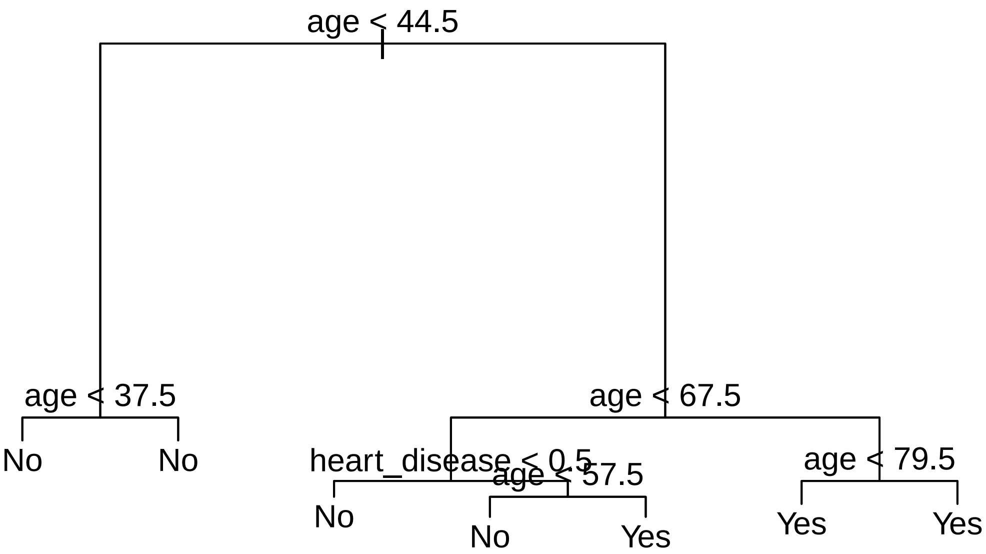
In the preprocessing stage, I had to change the attributes gender, "evermarried", work type, residence type, and smoking status to factor as they were originally character data types. I then created a new variable named 'stroke_likelihood' using the ifelse() function, based on the stroke variable. If 'stroke' is larger than or equal to one (which means yes, stroke), the result will be 'YES' and otherwise is 'NO'. After adding the stroke_likelihood to the data set (dtree) using the data.frame() function, I removed the stroke variable from the dataset because it is highly correlated to the stroke_likelihood variable. If both were present in the dataset, the decision tree will swing in favor of the stroke variable and thus, the results will be inflated/deflated and be inaccurate. After some analysis, I decided to also remove the ever married, work type, residence type, and id variables. The id is unique to each person, so it is an unsuitable variable to be tested on. The ever married, work type, and residence type does not contribute to the prediction of the likelihood of having a stroke, which is why I decided to leave them out of the data tree. By doing so, I am reducing the multidimensionality of the analysis and keeping only the ones that I feel are important to the analysis. After removing the variables and changing the variables' data types, I looked at the summary to see what it looked like after the preprocessing. After the preprocessing, it is easier to observe certain variables such as the gender and smoking status compared to the summary before preprocessing.

Training

```
dtree.stroke <- tree(stroke_likelihood~., dtree)
summary(dtree.stroke)
```

```
##
## Classification tree:
## tree(formula = stroke_likelihood ~ ., data = dtree)
## Variables actually used in tree construction:
## [1] "age"           "heart_disease"
## Number of terminal nodes:  7
## Residual mean deviance:  0.9238 = 413.9 / 448
## Misclassification error rate: 0.244 = 111 / 455
```

```
plot(dtree.stroke)
text(dtree.stroke,pretty=0)
```



```
print(dtree.stroke)
```

```
## node), split, n, deviance, yval, (yprob)
##       * denotes terminal node
##
##  1) root 455 627.80 No ( 0.54066 0.45934 )
##    2) age < 44.5 140  61.33 No ( 0.94286 0.05714 )
##      4) age < 37.5 105  19.80 No ( 0.98095 0.01905 ) *
##      5) age > 37.5 35  32.07 No ( 0.82857 0.17143 ) *
##    3) age > 44.5 315 412.30 Yes ( 0.36190 0.63810 )
##      6) age < 67.5 153 212.10 No ( 0.50327 0.49673 )
##       12) heart_disease < 0.5 138 190.60 No ( 0.53623 0.46377 ) *
##       13) heart_disease > 0.5 15  15.01 Yes ( 0.20000 0.80000 )
##         26) age < 57.5 5   6.73 No ( 0.60000 0.40000 ) *
##         27) age > 57.5 10   0.00 Yes ( 0.00000 1.00000 ) *
##      7) age > 67.5 162 174.10 Yes ( 0.22840 0.77160 )
##       14) age < 79.5 120 143.10 Yes ( 0.28333 0.71667 ) *
##       15) age > 79.5 42  21.61 Yes ( 0.07143 0.92857 ) *
```

In the training part, I assigned the tree algorithm with the stroke_likelihood being predicted based on the dtree dataset into the variable named dtree.stroke. By doing so, we can build a decision tree based on all 455 observations (not 500 because NA values have been omitted). The misclassification rate is 24.4%, meaning that based on the observations, 24.4% of the class labels are misclassified. I also plotted and added text (with 'pretty=0' to make the labels more meaningful) to add labels.

Splitting

```
set.seed(123)
train.index <- sample(1:nrow(dtree), 300)
train.set <- dtree[train.index,]
class(train.set)
```

```
## [1] "data.frame"
```

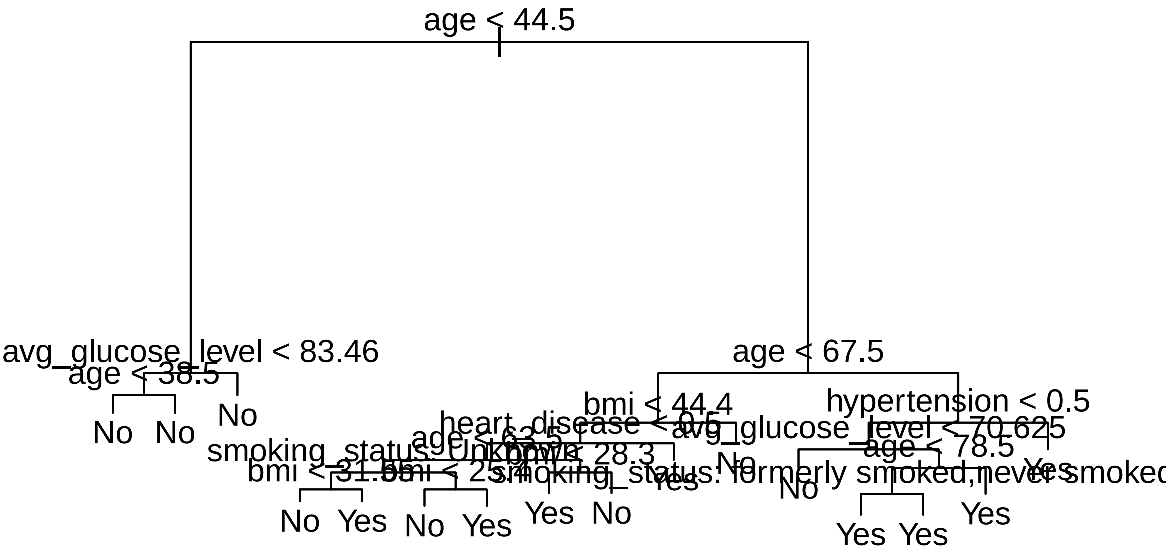
In the splitting process, I first used the function set.seed(123) in order to be able to replicate the random sampling. Creating a new variable called train.index, I used the sample() function in order to be able to randomly sample 300 records. In order to do the splitting, I created a train.set variable (to assign as the dataset for training), and used the train.index to extract the random 300 sample records into the training set.

The class of the train.set is data frame.

Training

Now that I have the 300 records to use for training, I train my model by creating a new variable named dtree.tree and build the decision tree to predict the likelihood of stroke (stroke_likelihood variable) based on the training dataset (train.set). The plot of the tree with text and summary looks like:

```
dtree.tree <- tree(stroke_likelihood~., train.set)
plot(dtree.tree)
text(dtree.tree, pretty=0)
```



```
summary(dtree.tree)
```

```
##
## Classification tree:
## tree(formula = stroke_likelihood ~ ., data = train.set)
## Variables actually used in tree construction:
## [1] "age"               "avg_glucose_level" "bmi"
## [4] "heart_disease"     "smoking_status"    "hypertension"
## Number of terminal nodes:  16
## Residual mean deviance:  0.6735 = 191.3 / 284
## Misclassification error rate: 0.1567 = 47 / 300
```

The misclassification error rate has improved slightly from 24.4% to 15.7% now.

Testing

```
test.set <- dtree[-train.index,]
stroke.test <- stroke_likelihood[-train.index]
tree.predict <- predict(dtree.tree, test.set, type="class")
table(tree.predict, stroke.test)
```

```
##             stroke.test
## tree.predict No Yes
##          No  55  18
##          Yes 27  55
```

```r
accuracy = (55+55)/155
accuracy
```

```
## [1] 0.7096774
```

After creating the training dataset, I then created the testing dataset. Creating the new variable named test.set, I assigned the dtree and excluded the 300 random sample records, which leaves me with 155 records for the test.set. I then created another variable known as stroke.text to include the known variables in the stroke_likelihood in order to be able to make comparisons and observe how many of the records are correctly predicted based on the actual values. In the stroke.test, I also excluded the 300 records in the train.index. When testing, I would like to test the predictive power of the model and I do this by creating the variable tree.predict and use the function predict() to predict based on the dtree.tree model to test classes in the test dataset (test.set). I then used the table() function to tabulate the predicted classes. The table will include the predictive classes and the actual classes. This gives us an accuracy of (55+55)/155 which is approximately 71%.

Crossvalidation

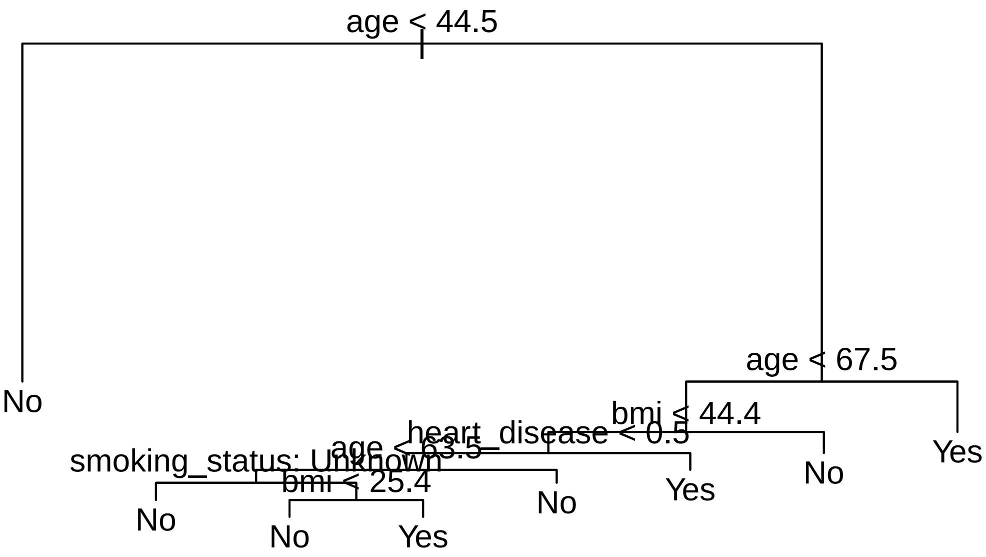
```r
crossvalidation.dtree <- cv.tree(dtree.tree, FUN=prune.misclass)
crossvalidation.dtree
```

```
## $size
## [1] 16 12 10  8  2  1
##
## $dev
## [1]  89  88  81  79  80 136
##
## $k
## [1]      -Inf  0.000000  1.000000  1.500000  3.833333 61.000000
##
## $method
## [1] "misclass"
##
## attr(,"class")
## [1] "prune"         "tree.sequence"
```

Following this is the cross validation step, and I created a new variable called crossvalidation.dtree and using the cv.tree() function, I am cross validating the trained model (dtree.tree) and base it on the prune.misclass function. By doing so, the tree that misclassifies the most will be removed and only the tree with the lowest misclassification will be kept. I will choose the 79 from the $dev since it is the smallest because it means that it has the least errors. This means I will be choosing the tree with the size 8 as it results in the least errors.

Pruning

```r
pruned.tree <- prune.misclass(dtree.tree, best=8)
plot(pruned.tree)
text(pruned.tree, pretty=0)
```



```
summary(pruned.tree)
```

```
##
## Classification tree:
## snip.tree(tree = dtree.tree, nodes = c(2L, 49L, 96L, 7L))
## Variables actually used in tree construction:
## [1] "age"            "bmi"            "heart_disease"  "smoking_status"
## Number of terminal nodes:  8
## Residual mean deviance:  0.8536 = 249.3 / 292
## Misclassification error rate: 0.1733 = 52 / 300
```

In the pruning step, I create the variable pruned.tree and prune the decision tree and look at the best of the size 8. I proceeded to plot the tree and add labels. I also looked at the summary to find the misclassification error rate of 17.3%.

Testing pruned tree

```
pruned.test.pred <- predict(pruned.tree, test.set, type="class")
table(stroke.test, pruned.test.pred)
```

```
##            pruned.test.pred
## stroke.test No Yes
##         No  57  25
##         Yes 16  57
```

```
(57+57)/155
```

```
## [1] 0.7354839
```

I then tested the pruned tree by creating the variable pruned.test.predict and predict based on the pruned.tree to test the class based on the test.set. I then tabulated the stroke.test based on the results gained from the pruned.test.pred. The accuracy is calculated as (57+57)/155 which is 73.5%.

Resulting decision tree

```
pruned.tree
```

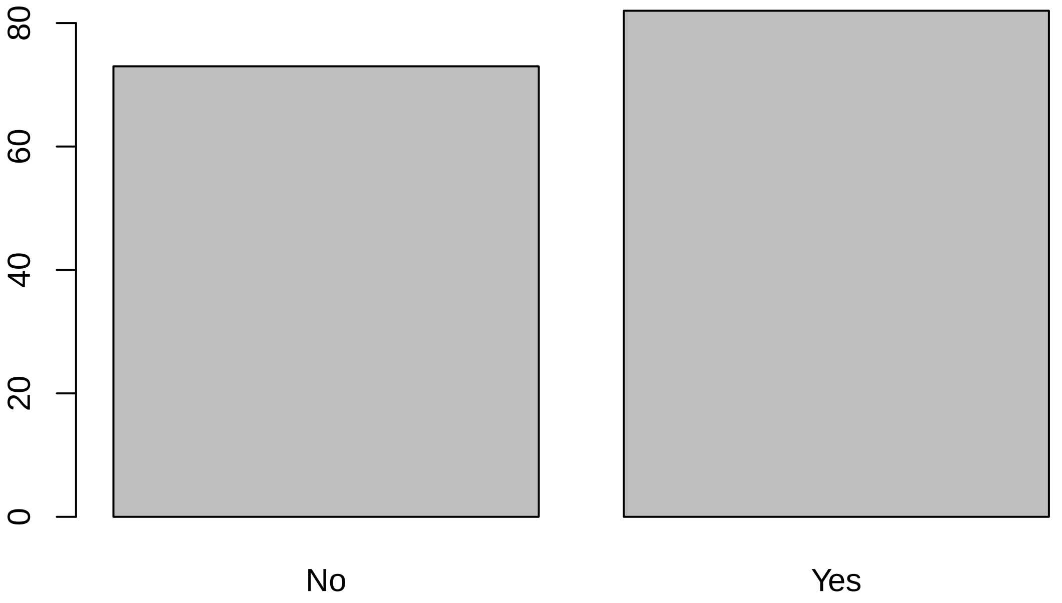
```
## node), split, n, deviance, yval, (yprob)
##       * denotes terminal node
```

```
##
##   1) root 300 413.300 No ( 0.54667 0.45333 )
##     2) age < 44.5 97  33.340 No ( 0.95876 0.04124 ) *
##     3) age > 44.5 203 262.800 Yes ( 0.34975 0.65025 )
##       6) age < 67.5 97 134.500 Yes ( 0.49485 0.50515 )
##        12) bmi < 44.4 92 127.100 Yes ( 0.46739 0.53261 )
##          24) heart_disease < 0.5 83 115.100 No ( 0.50602 0.49398 )
##            48) age < 63.5 70  96.530 Yes ( 0.45714 0.54286 )
##              96) smoking_status: Unknown 15  17.400 No ( 0.73333 0.26667 ) *
##              97) smoking_status: formerly smoked,never smoked,smokes 55  73.140 Yes ( 0.38182 0.61818
##               194) bmi < 25.4 6   5.407 No ( 0.83333 0.16667 ) *
##               195) bmi > 25.4 49  61.910 Yes ( 0.32653 0.67347 ) *
##            49) age > 63.5 13  14.050 No ( 0.76923 0.23077 ) *
##          25) heart_disease > 0.5 9   6.279 Yes ( 0.11111 0.88889 ) *
##        13) bmi > 44.4 5   0.000 No ( 1.00000 0.00000 ) *
##       7) age > 67.5 106 110.900 Yes ( 0.21698 0.78302 ) *
```

This is the optimal tree and we can see that it is much easier to deduce the likelihood of stroke. We can see that if the age is under 44.5, there is no chance (or little to none) for stroke to occur. If the age is equal to or larger than 67.5, there is a chance for stroke to occur. We can also see that if age is lower than 67.5 and bmi is higher than or equal to 44.4, there is also no chance for stroke. If age is between 44.5 and 63.5, bmi is less than 44.4, heart_disease is less than 0.5 and smoking status is unknown, we can also see that there is no chance for stroke to occur. These are the types of deductions that we can make from the optimized, pruned tree.

Predict stroke likelihood for test set

```
test.pred <- predict(dtree.tree, newdata = test.set, type = "class")
plot(test.pred)
```



Confusion matrix

```
table(test.set$stroke_likelihood, test.pred)
```

```
##       test.pred
##       No Yes
##   No  55  27
##   Yes 18  55
```

Performance metrics evaluation

Accuracy

The proportion of correct predictions out of total predictions made.

```
accc <- (55+55)/(55+55+27+18)
accc
```

```
## [1] 0.7096774
```

The accuracy of the decision tree model is 73.03%

Precision

The proportion of true positive predictions out of the total positive predictions made.

```
prec <- 55/(55+27)
prec
```

```
## [1] 0.6707317
```

The precision of the decision tree model is 67.07%.

Recall

The proportion of true positive predictions out of the total actual positive instances.

```
recalls <- 55/(55+18)
recalls
```

```
## [1] 0.7534247
```

The recall of the decision tree model is 75.34%.

F1 Score

The harmonic mean of precision and recall. It provides a balanced measure of both precision and recall.

```
f1s <- 2 * (0.6707 * 0.7534) / (0.6707 + 0.7534)
f1s
```

```
## [1] 0.7096487
```

The F1 score of the decision tree model is 70.90%.