

Does NFL Combine Data Predict Rushing Performance

Audrey Holloman and Nicole Kadosh

9/26/2018

For this project we used Decision Trees to see how a player's NFL combine results effect the player's rushing production. We cleaned the data by gathering the information from the NFL combine stats with the rushing data from 2017 season of the 50 players.

For Scenario One we used all the data points from the NFL combine; Height, Weight, 40 Yard, Bench Press, Veritcal Leap, Broad Jump, Shuttle and 3 Cone Drill. We then determined the two best "Y variables" from the rushing data to compare to the 8 "X variables".

```
installed.packages("tree")
```

```
##      Package LibPath Version Priority Depends Imports LinkingTo Suggests
##      Enhances License License_is_FOSS License_restricts_use OS_type Archs
##      MD5sum NeedsCompilation Built
```

```
library(tree)
```

```
nfl_data <-read.csv("/home/students/kadoshn/GitHub/Sports Analytics/CombineData.csv")
str(nfl_data)
```

```
## 'data.frame':    50 obs. of  18 variables:
## $ X           : Factor w/ 50 levels "Aaron Jones",...: 29 48 34 37 38 28 41 36 8 18 ...
## $ College      : Factor w/ 34 levels "", "Alabama", "Alabama State",...: 31 12 18 1 2 14 34 15 7 22 ...
## $ POS          : Factor w/ 4 levels "", "ILB", "QB",...: 4 4 4 1 4 4 4 4 4 4 ...
## $ Height       : num  70.5 72.6 73.4 NA 69.1 ...
## $ Weight       : int  216 222 230 NA 215 230 215 240 224 225 ...
## $ X40Yard      : num  4.62 4.52 4.6 NA 4.62 NA 4.52 4.51 4.6 4.47 ...
## $ BenchPress   : int  18 17 24 NA 21 16 19 NA 17 NA ...
## $ VertLeap     : num  36.5 NA 31.5 NA 31.5 34 35 28.5 32 32.5 ...
## $ BroadJump    : int  119 NA 118 NA 113 122 126 NA 119 118 ...
## $ Shuttle      : num  NA NA 4.24 NA 4.62 NA 4.07 NA 4.12 NA ...
## $ X3Cone       : num  NA NA 6.75 NA 7.13 NA 7.04 NA 7.15 NA ...
## $ Gms          : int  16 15 15 16 16 16 16 13 16 10 ...
## $ Att          : int  272 279 321 287 230 276 284 268 245 242 ...
## $ Yds          : Factor w/ 50 levels "1,007", "1,040",...: 9 8 7 6 5 4 3 2 1 50 ...
## $ Avg          : num  4.88 4.68 4.02 3.97 4.89 4.07 3.89 3.88 4.11 4.06 ...
## $ YPG          : num  82.9 87 86.1 71.1 70.2 70.1 69.1 80 62.9 98.3 ...
## $ Lg           : int  69 57 27 48 72 53 87 90 40 30 ...
## $ TD           : int  8 13 9 6 12 9 8 9 3 7 ...
```

```
names(nfl_data)
```

```
## [1] "X"           "College"     "POS"         "Height"      "Weight"
## [6] "X40Yard"     "BenchPress" "VertLeap"    "BroadJump"   "Shuttle"
## [11] "X3Cone"      "Gms"         "Att"         "Yds"         "Avg"
## [16] "YPG"         "Lg"          "TD"
```

```
nrow(nfl_data)
```

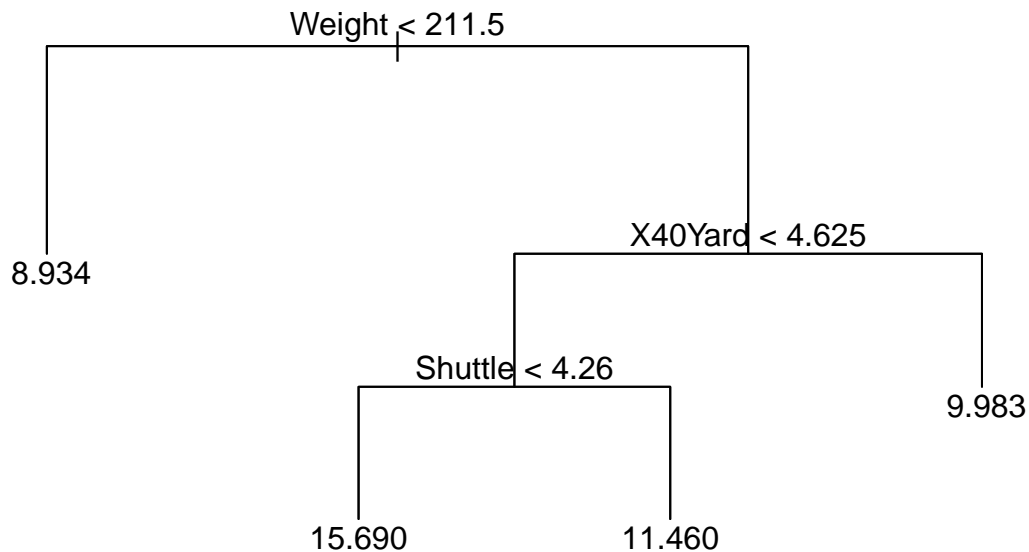
```
## [1] 50
```

```
ncol(nfl_data)
```

```
## [1] 18
set.seed(3)
select_rows <- sample(1:nrow(nfl_data),round(0.3*nrow(nfl_data)),replace=F)
select_rows

## [1] 9 40 19 16 28 46 6 13 25 26 21 20 49 38 32
test <- nfl_data[select_rows,]
train <- nfl_data[-(select_rows),]
nfl_data$new_column2 <- nfl_data$Att/nfl_data$Gms
nfl_data$new_column3 <- nfl_data$Yds/nfl_data$Att #no

## Warning in Ops.factor(nfl_data$Yds, nfl_data$Att): '/' not meaningful for
## factors
model_tree <- tree(nfl_data$new_column2~Height+Weight+X40Yard+BenchPress+VertLeap+BroadJump+Shuttle+X3Cor
plot(model_tree)
text(model_tree)
```



```
pred <- predict(model_tree,newdata = test)
pred
```

```
##          9          40          19          16          28          46          6
## 15.688627 11.455192 13.764339 13.764339 15.688627 8.933752 12.582626
##          13          25          26          21          20          49          38
## 13.764339 11.455192 11.455192 8.933752 11.269031 15.688627 9.982857
##          32
## 9.982857
```

For this Decision Tree the “Y variavble” we chose was attempts per game. We made this variable by inserting a new column into the data set that incorporatated attempts and games played per person from the rushing data. We did this because it allowed for more consistency since not all the players have played the same amount of games. The Decision Tree above shows that from the combine data weight is the first divisor making it of most importance and so on down the splits of the tree. If a player weighs less than 211.5 lbs then that player will have the lowest amount of attempts per game of 8.9. However, if a player weighs more than 211.5 lbs then the next divisor is the 40 Yard Dash. Now if the player runs more than 4.6 seconds for the 40 Yard Dash the player will have the next lowest attempts per game, 9.98. If the player runs the 40 Yard Dash less than 4.6 the next divisor occurs, the shuttle. If the player’s shuttle is greater than 4.3 then

the attempts per game is 11.5. Finally, If the players shuttle is less than 4.3 the attempts per game is 15.7. The best scenario for this tree is if the player weighs more than 211.5lbs, runs the 40 yard dash in less than 4.6 seconds and runs the shuttle in less than 4.3 seconds, as it gives you the highest attempts per game, 15.7. This makes sense as it is showing that the players who are faster and more agile, while having a heavier weight have a higher attempts per game.

```
installed.packages("tree")
```

```
##      Package LibPath Version Priority Depends Imports LinkingTo Suggests
##      Enhances License License_is_FOSS License_restricts_use OS_type Archs
##      MD5sum NeedsCompilation Built
```

```
library(tree)
```

```
nfl_data <-read.csv("/home/students/kadoshn/GitHub/Sports Analytics/CombineData.csv")
str(nfl_data)
```

```
## 'data.frame':    50 obs. of  18 variables:
## $ X      : Factor w/ 50 levels "Aaron Jones",...: 29 48 34 37 38 28 41 36 8 18 ...
## $ College : Factor w/ 34 levels "", "Alabama", "Alabama State",...: 31 12 18 1 2 14 34 15 7 22 ...
## $ POS     : Factor w/ 4 levels "", "ILB", "QB",...: 4 4 4 1 4 4 4 4 4 4 ...
## $ Height  : num  70.5 72.6 73.4 NA 69.1 ...
## $ Weight  : int  216 222 230 NA 215 230 215 240 224 225 ...
## $ X40Yard : num  4.62 4.52 4.6 NA 4.62 NA 4.52 4.51 4.6 4.47 ...
## $ BenchPress: int  18 17 24 NA 21 16 19 NA 17 NA ...
## $ VertLeap : num  36.5 NA 31.5 NA 31.5 34 35 28.5 32 32.5 ...
## $ BroadJump : int  119 NA 118 NA 113 122 126 NA 119 118 ...
## $ Shuttle  : num  NA NA 4.24 NA 4.62 NA 4.07 NA 4.12 NA ...
## $ X3Cone   : num  NA NA 6.75 NA 7.13 NA 7.04 NA 7.15 NA ...
## $ Gms      : int  16 15 15 16 16 16 16 13 16 10 ...
## $ Att      : int  272 279 321 287 230 276 284 268 245 242 ...
## $ Yds      : Factor w/ 50 levels "1,007", "1,040",...: 9 8 7 6 5 4 3 2 1 50 ...
## $ Avg      : num  4.88 4.68 4.02 3.97 4.89 4.07 3.89 3.88 4.11 4.06 ...
## $ YPG      : num  82.9 87 86.1 71.1 70.2 70.1 69.1 80 62.9 98.3 ...
## $ Lg       : int  69 57 27 48 72 53 87 90 40 30 ...
## $ TD       : int  8 13 9 6 12 9 8 9 3 7 ...
```

```
names(nfl_data)
```

```
## [1] "X"      "College" "POS"     "Height"  "Weight"
## [6] "X40Yard" "BenchPress" "VertLeap" "BroadJump" "Shuttle"
## [11] "X3Cone"  "Gms"      "Att"     "Yds"     "Avg"
## [16] "YPG"     "Lg"       "TD"
```

```
nrow(nfl_data)
```

```
## [1] 50
```

```
ncol(nfl_data)
```

```
## [1] 18
```

```
set.seed(2)
```

```
select_rows <- sample(1:nrow(nfl_data),round(0.3*nrow(nfl_data)),replace=F)
select_rows
```

```
## [1] 10 35 28 8 44 43 6 36 20 23 41 50 29 7 15
```

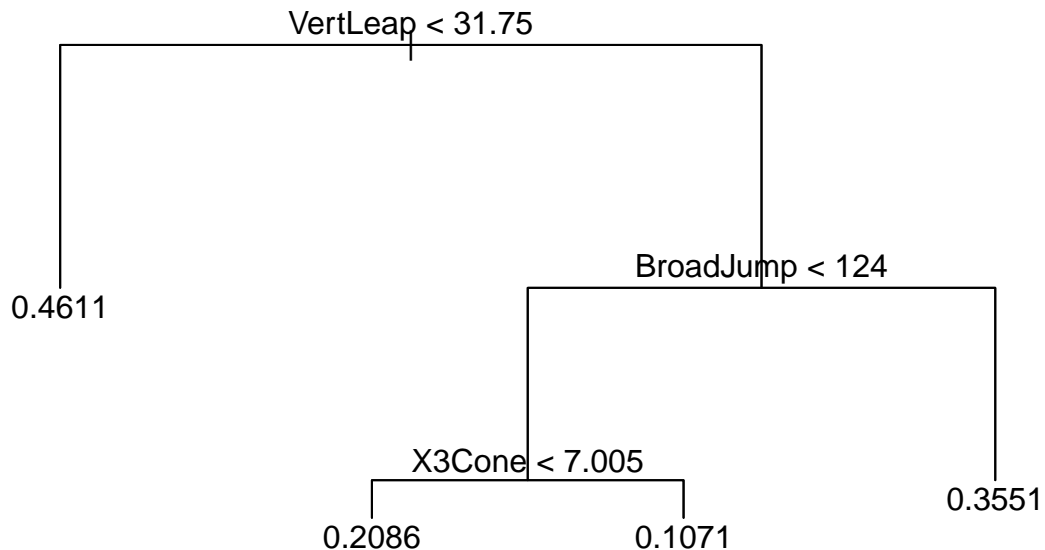
```

test <- nfl_data[select_rows,]
train <- nfl_data[-(select_rows),]
nfl_data$new_column <- nfl_data$TD / nfl_data$Gms
nfl_data$new_column4 <- (nfl_data$TD + nfl_data$Yds) /nfl_data$Gms #doesn't work

## Warning in Ops.factor(nfl_data$TD, nfl_data$Yds): '+' not meaningful for
## factors

model_tree <-tree(nfl_data$new_column~Height+Weight+X40Yard+BenchPress+VertLeap+BroadJump+Shuttle+X3Con
plot(model_tree)
text(model_tree)

```



```

pred <- predict(model_tree,newdata = test)
pred

##          10          35          28          8          44          43          6
## 0.1617757 0.3551020 0.3551020 0.4611111 0.2086039 0.3551020 0.1617757
##          36          20          23          41          50          29          7
## 0.2757742 0.2757742 0.3551020 0.2757742 0.4611111 0.1071429 0.3551020
##          15
## 0.3551020

```

For this Decision Tree the “Y variable” we chose was touchdowns per game. We made this variable by inserting a new column into the data set that incorporated touchdowns and games played per person from the rushing data. We did this because it allowed for more consistency since not all the players have played the same amount of games.

The best scenario from the decision tree above is that the player’s vertical leap is less than 31.8 inches giving the player .46 touchdowns per game. This decision tree shows that touchdowns per game is a lesser indicator than attempt per game, shown in the previous tree. This should result in scenerio 2 having touchdown

This decision tree should show that the correlating tree to these variables in Scenario Two will have a worse division rank than from the first tree’s variables.

Prior to starting to Scenario Two we gathering the infomation of the divisions that each of the players went to. We intered this data as a new column into the data set (nfl_data) to use as the “Y variable” for this scenario. This then lead us to Scenario Two, which is taking the variables from Scenario One, “X variables”, and compared them to that “Y variable” of the devision ranks. The goal is to determine what variables correspond to different college devisions.

```
installed.packages("tree")
```

```
##      Package LibPath Version Priority Depends Imports LinkingTo Suggests
##      Enhances License License_is_FOSS License_restricts_use OS_type Archs
##      MD5sum NeedsCompilation Built
```

```
library(tree)
```

```
nfl_data <- read.csv("/home/students/kadoshn/GitHub/Sports Analytics/CombineData.csv")
str(nfl_data)
```

```
## 'data.frame':    50 obs. of  18 variables:
## $ X      : Factor w/ 50 levels "Aaron Jones",...: 29 48 34 37 38 28 41 36 8 18 ...
## $ College : Factor w/ 34 levels "", "Alabama", "Alabama State",...: 31 12 18 1 2 14 34 15 7 22 ...
## $ POS     : Factor w/ 4 levels "", "ILB", "QB",...: 4 4 4 1 4 4 4 4 4 4 ...
## $ Height  : num  70.5 72.6 73.4 NA 69.1 ...
## $ Weight  : int   216 222 230 NA 215 230 215 240 224 225 ...
## $ X40Yard : num   4.62 4.52 4.6 NA 4.62 NA 4.52 4.51 4.6 4.47 ...
## $ BenchPress: int    18 17 24 NA 21 16 19 NA 17 NA ...
## $ VertLeap : num   36.5 NA 31.5 NA 31.5 34 35 28.5 32 32.5 ...
## $ BroadJump : int   119 NA 118 NA 113 122 126 NA 119 118 ...
## $ Shuttle  : num    NA NA 4.24 NA 4.62 NA 4.07 NA 4.12 NA ...
## $ X3Cone   : num    NA NA 6.75 NA 7.13 NA 7.04 NA 7.15 NA ...
## $ Gms      : int    16 15 15 16 16 16 16 13 16 10 ...
## $ Att      : int   272 279 321 287 230 276 284 268 245 242 ...
## $ Yds      : Factor w/ 50 levels "1,007", "1,040",...: 9 8 7 6 5 4 3 2 1 50 ...
## $ Avg      : num   4.88 4.68 4.02 3.97 4.89 4.07 3.89 3.88 4.11 4.06 ...
## $ YPG      : num   82.9 87 86.1 71.1 70.2 70.1 69.1 80 62.9 98.3 ...
## $ Lg       : int    69 57 27 48 72 53 87 90 40 30 ...
## $ TD       : int     8 13 9 6 12 9 8 9 3 7 ...
```

```
names(nfl_data)
```

```
## [1] "X"      "College" "POS"     "Height"  "Weight"
## [6] "X40Yard" "BenchPress" "VertLeap" "BroadJump" "Shuttle"
## [11] "X3Cone"  "Gms"      "Att"     "Yds"     "Avg"
## [16] "Lg"      "TD"
```

```
nrow(nfl_data)
```

```
## [1] 50
```

```
ncol(nfl_data)
```

```
## [1] 18
```

```
set.seed(3)
```

```
select_rows <- sample(1:nrow(nfl_data), round(0.3*nrow(nfl_data)), replace=F)
select_rows
```

```
## [1] 9 40 19 16 28 46 6 13 25 26 21 20 49 38 32
```

```
test <- nfl_data[select_rows,]
```

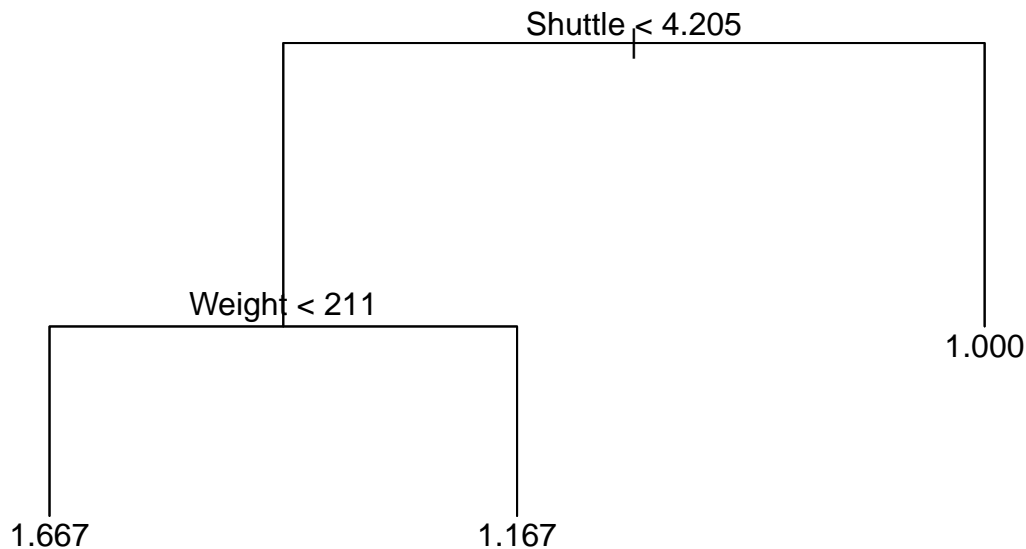
```
train <- nfl_data[-(select_rows),]
```

```
nfl_data$new_column <- nfl_data$TD / nfl_data$Gms
```

```
nfl_data$new_column4 <- (nfl_data$TD + nfl_data$Yds) / nfl_data$Gms #doesn't work
```

```
## Warning in Ops.factor(nfl_data$TD, nfl_data$Yds): '+' not meaningful for
## factors
```

```
nfl_data$Division <- c(1,1,1,NA,1,1,1,1,1,1,NA,1,1,1,1,1,NA,1,1,1,NA,1,1,1,1,1,1,1,1,1,3,NA,1,2)
model_tree <-tree(nfl_data$Division~Weight+X40Yard+Shuttle+BenchPress, data = nfl_data)
plot(model_tree)
text(model_tree)
```



```
pred <- predict(model_tree,newdata = test)
pred
```

```
##          9          40          19          16          28          46          6          13
## 1.166667 1.000000 1.192308 1.192308 1.166667 1.666667 1.192308 1.192308
##          25          26          21          20          49          38          32
## 1.000000 1.000000 1.192308 1.192308 1.166667 1.166667 1.000000
```

For this decision tree the “X variables” that we used were the divisors from the first tree of scenario one; Weight, 40 Yard Dash and Shuttle. We also added a 4th variable of Bench Press because we determined that this variable correlated with the previous 3 in testing a players strength, since Bench Press usually relates to weight. The best scenario from this decision tree is that if a player runs the shuttle greater than a 4.2 the player is more likely to be in deviation one. However, this does not show what you would expect since it is stating that a player who runs the shuttle at a slower speed is more likely to end up in the better deviation. We think this might have happened because this is based off of stronger and heavier players, which usually aren’t as fast of players.

```
installed.packages("tree")
```

```
##      Package LibPath Version Priority Depends Imports LinkingTo Suggests
##      Enhances License License_is_FOSS License_restricts_use OS_type Archs
##      MD5sum NeedsCompilation Built
```

```
library(tree)
nfl_data <-read.csv("/home/students/kadoshn/GitHub/Sports Analytics/CombineData.csv")
str(nfl_data)
```

```
## 'data.frame':   50 obs. of  18 variables:
## $ X          : Factor w/ 50 levels "Aaron Jones",...: 29 48 34 37 38 28 41 36 8 18 ...
## $ College     : Factor w/ 34 levels "", "Alabama", "Alabama State",...: 31 12 18 1 2 14 34 15 7 22 ...
## $ POS         : Factor w/ 4 levels "", "ILB", "QB",...: 4 4 4 1 4 4 4 4 4 4 ...
## $ Height      : num  70.5 72.6 73.4 NA 69.1 ...
## $ Weight       : int  216 222 230 NA 215 230 215 240 224 225 ...
```

```
## $ X40Yard : num 4.62 4.52 4.6 NA 4.62 NA 4.52 4.51 4.6 4.47 ...
## $ BenchPress: int 18 17 24 NA 21 16 19 NA 17 NA ...
## $ VertLeap : num 36.5 NA 31.5 NA 31.5 34 35 28.5 32 32.5 ...
## $ BroadJump : int 119 NA 118 NA 113 122 126 NA 119 118 ...
## $ Shuttle : num NA NA 4.24 NA 4.62 NA 4.07 NA 4.12 NA ...
## $ X3Cone : num NA NA 6.75 NA 7.13 NA 7.04 NA 7.15 NA ...
## $ Gms : int 16 15 15 16 16 16 16 13 16 10 ...
## $ Att : int 272 279 321 287 230 276 284 268 245 242 ...
## $ Yds : Factor w/ 50 levels "1,007","1,040",...: 9 8 7 6 5 4 3 2 1 50 ...
## $ Avg : num 4.88 4.68 4.02 3.97 4.89 4.07 3.89 3.88 4.11 4.06 ...
## $ YPG : num 82.9 87 86.1 71.1 70.2 70.1 69.1 80 62.9 98.3 ...
## $ Lg : int 69 57 27 48 72 53 87 90 40 30 ...
## $ TD : int 8 13 9 6 12 9 8 9 3 7 ...
```

```
names(nfl_data)
```

```
## [1] "X"           "College"     "POS"         "Height"      "Weight"
## [6] "X40Yard"     "BenchPress"  "VertLeap"    "BroadJump"   "Shuttle"
## [11] "X3Cone"      "Gms"         "Att"         "Yds"         "Avg"
## [16] "YPG"         "Lg"          "TD"
```

```
nrow(nfl_data)
```

```
## [1] 50
```

```
ncol(nfl_data)
```

```
## [1] 18
```

```
set.seed(3)
```

```
select_rows <- sample(1:nrow(nfl_data),round(0.3*nrow(nfl_data)),replace=F)
```

```
select_rows
```

```
## [1] 9 40 19 16 28 46 6 13 25 26 21 20 49 38 32
```

```
test <- nfl_data[select_rows,]
```

```
train <- nfl_data[-(select_rows),]
```

```
nfl_data$new_column <- nfl_data$TD / nfl_data$Gms
```

```
nfl_data$new_column4 <- (nfl_data$TD + nfl_data$Yds) /nfl_data$Gms #doesn't work
```

```
## Warning in Ops.factor(nfl_data$TD, nfl_data$Yds): '+' not meaningful for
```

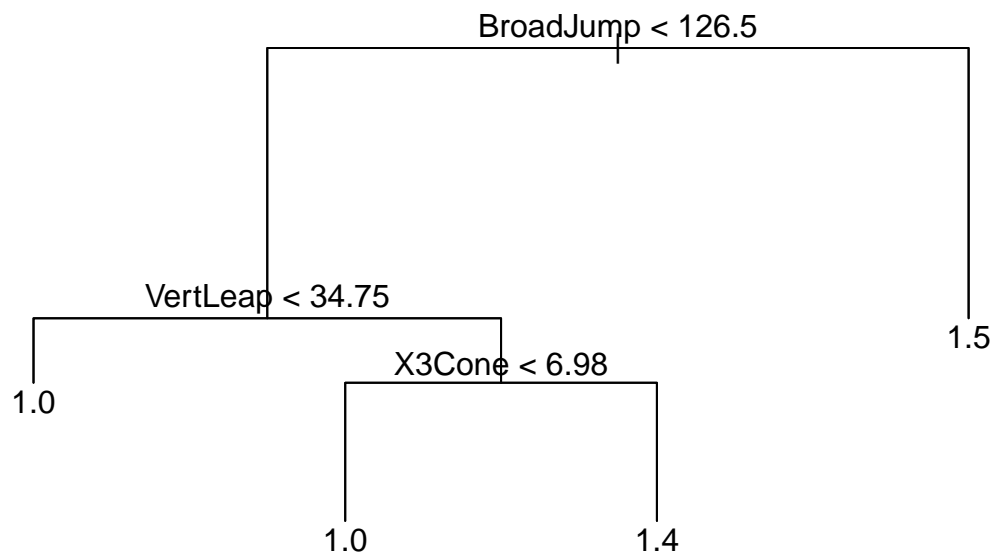
```
## factors
```

```
nfl_data$Division <- c(1,1,1,NA,1,1,1,1,1,1,1,NA,1,1,1,1,1,1,NA,1,1,1,NA,1,1,1,1,1,1,1,1,1,1,1,3,NA,1,2
```

```
model_tree <-tree(nfl_data$Division~VertLeap+BroadJump+X3Cone+X40Yard, data = nfl_data)
```

```
plot(model_tree)
```

```
text(model_tree)
```



```
pred <- predict(model_tree,newdata = test)
pred
```

```
##          9          40          19          16          28          46          6          13
## 1.000000 1.000000 1.181818 1.161290 1.500000 1.400000 1.000000 1.161290
##          25          26          21          20          49          38          32
## 1.500000 1.500000 1.161290 1.161290 1.000000 1.400000 1.000000
```

For this decision tree the “X variables” that we used were the divisors from the second tree of scenario one; Broad Jump, Veritcal Leap and 3 Cone. We also added a 4th variable of 40 Yard Dash because we determined that this variable correlated with the previous 3 in testing a players speed and agility, since the skill of the 40 Yard Dash relate to the 3 Cone as they both test similar objectives. For this decision tree there are two scenarios where the outcome is optimal. For the first scenario, if the player’s Broad Jump is less than 126.5 and Veritcal Leap is less than 34.8 inches then the player is more likely to be in devision one. The second scenario, if the player’s Broad Jump is less than 126.5, Veritcal Leap is less than 34.8 inches, and 3 Cone Drill is less than 6.98 then the player is more likely to be in division one.

The trees from scenario one and scenario two correlate to show that higher divisions usually have the better players in the league and the better players are usually in better divisions (division 1). This project is also showing that the nfl combine is a good indicator of how a player will perform in regards to rushing yards within his NFL career. Certain aspects of the combine data show to be more significant in perdicting the players performance.