SleepTrouble_Analysis

Krishangi

6/20/2020

This analysis predicts factors affecting sleep from data collected from NHANES : National Health And Nutrition Examination Survey.

```r
library(NHANES)
```

```
## Warning: package 'NHANES' was built under R version 3.6.3
```

```r
library(rpart)
library(partykit)
```

```
## Warning: package 'partykit' was built under R version 3.6.3

## Loading required package: grid

## Loading required package: libcoin

## Warning: package 'libcoin' was built under R version 3.6.2

## Loading required package: mvtnorm

## Warning: package 'mvtnorm' was built under R version 3.6.2
```

```r
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.6.3

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

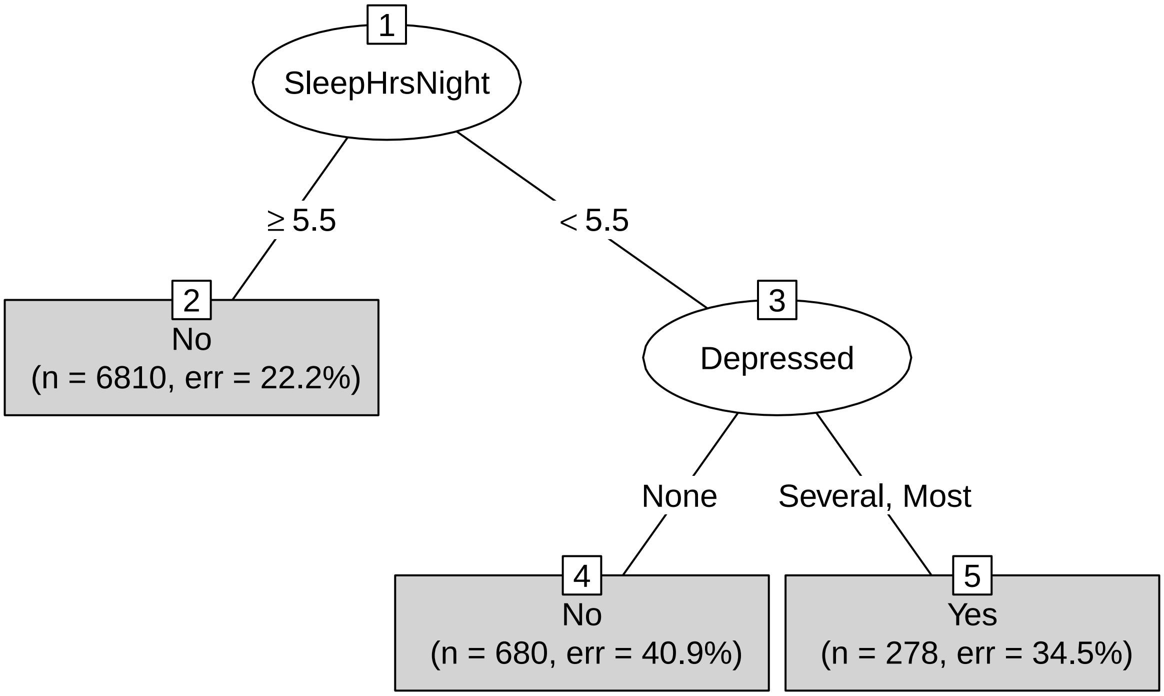
```r
glimpse(NHANES)
```

```
## Rows: 10,000
## Columns: 76
## $ ID               <int> 51624, 51624, 51624, 51625, 51630, 51638, 51646, 5...
## $ SurveyYr         <fct> 2009_10, 2009_10, 2009_10, 2009_10, 2009_10, 2009_...
## $ Gender           <fct> male, male, male, male, female, male, male, female...
## $ Age              <int> 34, 34, 34, 4, 49, 9, 8, 45, 45, 45, 66, 58, 54, 1...
## $ AgeDecade        <fct>  30-39,  30-39,  30-39,  0-9,  40-49,  0-9,  0-9, ...
## $ AgeMonths        <int> 409, 409, 409, 49, 596, 115, 101, 541, 541, 541, 7...
## $ Race1            <fct> White, White, White, Other, White, White, White, W...
## $ Race3            <fct> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ Education        <fct> High School, High School, High School, NA, Some Co...
## $ MaritalStatus    <fct> Married, Married, Married, NA, LivePartner, NA, NA...
## $ HHIncome         <fct> 25000-34999, 25000-34999, 25000-34999, 20000-24999...
## $ HHIncomeMid      <int> 30000, 30000, 30000, 22500, 40000, 87500, 60000, 8...
## $ Poverty          <dbl> 1.36, 1.36, 1.36, 1.07, 1.91, 1.84, 2.33, 5.00, 5....
## $ HomeRooms        <int> 6, 6, 6, 9, 5, 6, 7, 6, 6, 6, 5, 10, 6, 10, 10, 4,...
## $ HomeOwn          <fct> Own, Own, Own, Own, Rent, Rent, Own, Own, Own, Own...
## $ Work             <fct> NotWorking, NotWorking, NotWorking, NA, NotWorking...
## $ Weight           <dbl> 87.4, 87.4, 87.4, 17.0, 86.7, 29.8, 35.2, 75.7, 75...
## $ Length           <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ HeadCirc         <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ Height           <dbl> 164.7, 164.7, 164.7, 105.4, 168.4, 133.1, 130.6, 1...
## $ BMI              <dbl> 32.22, 32.22, 32.22, 15.30, 30.57, 16.82, 20.64, 2...
## $ BMICatUnder20yrs <fct> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ BMI_WHO          <fct> 30.0_plus, 30.0_plus, 30.0_plus, 12.0_18.5, 30.0_p...
## $ Pulse            <int> 70, 70, 70, NA, 86, 82, 72, 62, 62, 62, 60, 62, 76...
## $ BPSysAve         <int> 113, 113, 113, NA, 112, 86, 107, 118, 118, 118, 11...
## $ BPDiaAve         <int> 85, 85, 85, NA, 75, 47, 37, 64, 64, 64, 63, 74, 85...
## $ BPSys1           <int> 114, 114, 114, NA, 118, 84, 114, 106, 106, 106, 12...
## $ BPDia1           <int> 88, 88, 88, NA, 82, 50, 46, 62, 62, 62, 64, 76, 86...
## $ BPSys2           <int> 114, 114, 114, NA, 108, 84, 108, 118, 118, 118, 10...
## $ BPDia2           <int> 88, 88, 88, NA, 74, 50, 36, 68, 68, 68, 62, 72, 88...
## $ BPSys3           <int> 112, 112, 112, NA, 116, 88, 106, 118, 118, 118, 11...
## $ BPDia3           <int> 82, 82, 82, NA, 76, 44, 38, 60, 60, 60, 64, 76, 82...
## $ Testosterone     <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ DirectChol       <dbl> 1.29, 1.29, 1.29, NA, 1.16, 1.34, 1.55, 2.12, 2.12...
## $ TotChol          <dbl> 3.49, 3.49, 3.49, NA, 6.70, 4.86, 4.09, 5.82, 5.82...
## $ UrineVol1        <int> 352, 352, 352, NA, 77, 123, 238, 106, 106, 106, 11...
## $ UrineFlow1       <dbl> NA, NA, NA, NA, 0.094, 1.538, 1.322, 1.116, 1.116,...
## $ UrineVol2        <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ UrineFlow2       <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ Diabetes         <fct> No, No, No, No, No, No, No, No, No, No, No, No, No...
## $ DiabetesAge      <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ HealthGen        <fct> Good, Good, Good, NA, Good, NA, NA, Vgood, Vgood, ...
## $ DaysPhysHlthBad  <int> 0, 0, 0, NA, 0, NA, NA, 0, 0, 0, 10, 0, 4, NA, NA,...
## $ DaysMentHlthBad  <int> 15, 15, 15, NA, 10, NA, NA, 3, 3, 3, 0, 0, 0, NA, ...
## $ LittleInterest   <fct> Most, Most, Most, NA, Several, NA, NA, None, None,...
## $ Depressed        <fct> Several, Several, Several, NA, Several, NA, NA, No...
## $ nPregnancies     <int> NA, NA, NA, NA, 2, NA, NA, 1, 1, 1, NA, NA, NA, NA...
## $ nBabies          <int> NA, NA, NA, NA, 2, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ Age1stBaby       <int> NA, NA, NA, NA, 27, NA, NA, NA, NA, NA, NA, NA, NA...
## $ SleepHrsNight    <int> 4, 4, 4, NA, 8, NA, NA, 8, 8, 8, 7, 5, 4, NA, 5, 7...
## $ SleepTrouble     <fct> Yes, Yes, Yes, NA, Yes, NA, NA, No, No, No, No, No...
## $ PhysActive       <fct> No, No, No, NA, No, NA, NA, Yes, Yes, Yes, Yes, Ye...
## $ PhysActiveDays   <int> NA, NA, NA, NA, NA, NA, NA, 5, 5, 5, 7, 5, 1, NA, ...
## $ TVHrsDay         <fct> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ CompHrsDay       <fct> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ TVHrsDayChild    <int> NA, NA, NA, 4, NA, 5, 1, NA, NA, NA, NA, NA, NA, 4...
## $ CompHrsDayChild  <int> NA, NA, NA, 1, NA, 0, 6, NA, NA, NA, NA, NA, NA, 3...
## $ Alcohol12PlusYr  <fct> Yes, Yes, Yes, NA, Yes, NA, NA, Yes, Yes, Yes, Yes...
## $ AlcoholDay       <int> NA, NA, NA, NA, 2, NA, NA, 3, 3, 3, 1, 2, 6, NA, N...
## $ AlcoholYear      <int> 0, 0, 0, NA, 20, NA, NA, 52, 52, 52, 100, 104, 364...
## $ SmokeNow         <fct> No, No, No, NA, Yes, NA, NA, NA, NA, NA, No, NA, N...
## $ Smoke100         <fct> Yes, Yes, Yes, NA, Yes, NA, NA, No, No, No, Yes, N...
## $ Smoke100n        <fct> Smoker, Smoker, Smoker, NA, Smoker, NA, NA, Non-Sm...
## $ SmokeAge         <int> 18, 18, 18, NA, 38, NA, NA, NA, NA, NA, 13, NA, NA...
## $ Marijuana        <fct> Yes, Yes, Yes, NA, Yes, NA, NA, Yes, Yes, Yes, NA,...
## $ AgeFirstMarij    <int> 17, 17, 17, NA, 18, NA, NA, 13, 13, 13, NA, 19, 15...
## $ RegularMarij     <fct> No, No, No, NA, No, NA, NA, No, No, No, NA, Yes, Y...
## $ AgeRegMarij      <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, 20, 15...
## $ HardDrugs        <fct> Yes, Yes, Yes, NA, Yes, NA, NA, No, No, No, No, Ye...
## $ SexEver          <fct> Yes, Yes, Yes, NA, Yes, NA, NA, Yes, Yes, Yes, Yes...
## $ SexAge           <int> 16, 16, 16, NA, 12, NA, NA, 13, 13, 13, 17, 22, 12...
## $ SexNumPartnLife  <int> 8, 8, 8, NA, 10, NA, NA, 20, 20, 20, 15, 7, 100, N...
## $ SexNumPartYear   <int> 1, 1, 1, NA, 1, NA, NA, 0, 0, 0, NA, 1, 1, NA, NA,...
## $ SameSex          <fct> No, No, No, NA, Yes, NA, NA, Yes, Yes, Yes, No, No...
## $ SexOrientation   <fct> Heterosexual, Heterosexual, Heterosexual, NA, Hete...
## $ PregnantNow      <fct> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
```

```r
NHANES_new <- NHANES[complete.cases(NHANES$SleepTrouble), ]
str(NHANES_new)
```

```
## tibble [7,772 x 76] (S3: tbl_df/tbl/data.frame)
##  $ ID              : int [1:7772] 51624 51624 51624 51630 51647 51647 51647 51654 51656 51657 ...
##  $ SurveyYr        : Factor w/ 2 levels "2009_10","2011_12": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Gender          : Factor w/ 2 levels "female","male": 2 2 2 1 1 1 1 2 2 2 ...
##  $ Age             : int [1:7772] 34 34 34 49 45 45 45 66 58 54 ...
##  $ AgeDecade       : Factor w/ 8 levels " 0-9"," 10-19",..: 4 4 4 5 5 5 5 7 6 6 ...
##  $ AgeMonths       : int [1:7772] 409 409 409 596 541 541 541 795 707 654 ...
##  $ Race1           : Factor w/ 5 levels "Black","Hispanic",..: 4 4 4 4 4 4 4 4 4 4 ...
##  $ Race3           : Factor w/ 6 levels "Asian","Black",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ Education       : Factor w/ 5 levels "8th Grade","9 - 11th Grade",..: 3 3 3 4 5 5 5 4 5 2 ...
##  $ MaritalStatus   : Factor w/ 6 levels "Divorced","LivePartner",..: 3 3 3 2 3 3 3 3 1 3 ...
##  $ HHIncome        : Factor w/ 12 levels " 0-4999"," 5000-9999",..: 6 6 6 7 11 11 11 6 12 10 ...
##  $ HHIncomeMid     : int [1:7772] 30000 30000 30000 40000 87500 87500 87500 30000 100000 70000 ...
##  $ Poverty         : num [1:7772] 1.36 1.36 1.36 1.91 5 5 5 2.2 5 2.2 ...
##  $ HomeRooms       : int [1:7772] 6 6 6 5 6 6 6 5 10 6 ...
##  $ HomeOwn         : Factor w/ 3 levels "Own","Rent","Other": 1 1 1 2 1 1 1 1 2 2 ...
##  $ Work            : Factor w/ 3 levels "Looking","NotWorking",..: 2 2 2 2 3 3 3 2 3 3 ...
##  $ Weight          : num [1:7772] 87.4 87.4 87.4 86.7 75.7 75.7 75.7 68 78.4 74.7 ...
##  $ Length          : num [1:7772] NA NA NA NA NA NA NA NA NA NA ...
##  $ HeadCirc        : num [1:7772] NA NA NA NA NA NA NA NA NA NA ...
##  $ Height          : num [1:7772] 165 165 165 168 167 ...
##  $ BMI             : num [1:7772] 32.2 32.2 32.2 30.6 27.2 ...
##  $ BMICatUnder20yrs: Factor w/ 4 levels "UnderWeight",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ BMI_WHO         : Factor w/ 4 levels "12.0_18.5","18.5_to_24.9",..: 4 4 4 4 3 3 3 2 2 3 ...
##  $ Pulse           : int [1:7772] 70 70 70 86 62 62 62 60 62 76 ...
##  $ BPSysAve        : int [1:7772] 113 113 113 112 118 118 118 111 104 134 ...
##  $ BPDiaAve        : int [1:7772] 85 85 85 75 64 64 64 63 74 85 ...
##  $ BPSys1          : int [1:7772] 114 114 114 118 106 106 106 124 108 136 ...
##  $ BPDia1          : int [1:7772] 88 88 88 82 62 62 62 64 76 86 ...
##  $ BPSys2          : int [1:7772] 114 114 114 108 118 118 118 108 104 132 ...
##  $ BPDia2          : int [1:7772] 88 88 88 74 68 68 68 62 72 88 ...
##  $ BPSys3          : int [1:7772] 112 112 112 116 118 118 118 114 104 136 ...
##  $ BPDia3          : int [1:7772] 82 82 82 76 60 60 60 64 76 82 ...
##  $ Testosterone    : num [1:7772] NA NA NA NA NA NA NA NA NA NA ...
##  $ DirectChol      : num [1:7772] 1.29 1.29 1.29 1.16 2.12 2.12 2.12 0.67 0.96 1.16 ...
##  $ TotChol         : num [1:7772] 3.49 3.49 3.49 6.7 5.82 5.82 5.82 4.99 4.24 6.41 ...
##  $ UrineVol1       : int [1:7772] 352 352 352 77 106 106 106 113 163 215 ...
##  $ UrineFlow1      : num [1:7772] NA NA NA 0.094 1.116 ...
##  $ UrineVol2       : int [1:7772] NA NA NA NA NA NA NA NA NA NA ...
##  $ UrineFlow2      : num [1:7772] NA NA NA NA NA NA NA NA NA NA ...
##  $ Diabetes        : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 1 1 ...
##  $ DiabetesAge     : int [1:7772] NA NA NA NA NA NA NA NA NA NA ...
##  $ HealthGen       : Factor w/ 5 levels "Excellent","Vgood",..: 3 3 3 3 2 2 2 2 2 4 ...
##  $ DaysPhysHlthBad : int [1:7772] 0 0 0 0 0 0 0 10 0 4 ...
##  $ DaysMentHlthBad : int [1:7772] 15 15 15 10 3 3 3 0 0 0 ...
##  $ LittleInterest  : Factor w/ 3 levels "None","Several",..: 3 3 3 2 1 1 1 1 1 1 ...
##  $ Depressed       : Factor w/ 3 levels "None","Several",..: 2 2 2 2 1 1 1 1 1 1 ...
##  $ nPregnancies    : int [1:7772] NA NA NA 2 1 1 1 NA NA NA ...
##  $ nBabies         : int [1:7772] NA NA NA 2 NA NA NA NA NA NA ...
##  $ Age1stBaby      : int [1:7772] NA NA NA 27 NA NA NA NA NA NA ...
##  $ SleepHrsNight   : int [1:7772] 4 4 4 8 8 8 8 7 5 4 ...
##  $ SleepTrouble    : Factor w/ 2 levels "No","Yes": 2 2 2 2 1 1 1 1 1 2 ...
##  $ PhysActive      : Factor w/ 2 levels "No","Yes": 1 1 1 1 2 2 2 2 2 2 ...
##  $ PhysActiveDays  : int [1:7772] NA NA NA NA 5 5 5 7 5 1 ...
##  $ TVHrsDay        : Factor w/ 7 levels "0_hrs","0_to_1_hr",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ CompHrsDay      : Factor w/ 7 levels "0_hrs","0_to_1_hr",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ TVHrsDayChild   : int [1:7772] NA NA NA NA NA NA NA NA NA NA ...
##  $ CompHrsDayChild : int [1:7772] NA NA NA NA NA NA NA NA NA NA ...
##  $ Alcohol12PlusYr : Factor w/ 2 levels "No","Yes": 2 2 2 2 2 2 2 2 2 2 ...
##  $ AlcoholDay      : int [1:7772] NA NA NA 2 3 3 3 1 2 6 ...
##  $ AlcoholYear     : int [1:7772] 0 0 0 20 52 52 52 100 104 364 ...
##  $ SmokeNow        : Factor w/ 2 levels "No","Yes": 1 1 1 2 NA NA NA 1 NA NA ...
##  $ Smoke100        : Factor w/ 2 levels "No","Yes": 2 2 2 2 1 1 1 2 1 1 ...
##  $ Smoke100n       : Factor w/ 2 levels "Non-Smoker","Smoker": 2 2 2 2 1 1 1 2 1 1 ...
##  $ SmokeAge        : int [1:7772] 18 18 18 38 NA NA NA 13 NA NA ...
##  $ Marijuana       : Factor w/ 2 levels "No","Yes": 2 2 2 2 2 2 2 NA 2 2 ...
##  $ AgeFirstMarij   : int [1:7772] 17 17 17 18 13 13 13 NA 19 15 ...
##  $ RegularMarij    : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 NA 2 2 ...
##  $ AgeRegMarij     : int [1:7772] NA NA NA NA NA NA NA NA 20 15 ...
##  $ HardDrugs       : Factor w/ 2 levels "No","Yes": 2 2 2 2 1 1 1 1 2 2 ...
##  $ SexEver         : Factor w/ 2 levels "No","Yes": 2 2 2 2 2 2 2 2 2 2 ...
##  $ SexAge          : int [1:7772] 16 16 16 12 13 13 13 17 22 12 ...
##  $ SexNumPartnLife : int [1:7772] 8 8 8 10 20 20 20 15 7 100 ...
##  $ SexNumPartYear  : int [1:7772] 1 1 1 1 0 0 0 NA 1 1 ...
##  $ SameSex         : Factor w/ 2 levels "No","Yes": 1 1 1 2 2 2 2 1 1 1 ...
##  $ SexOrientation  : Factor w/ 3 levels "Bisexual","Heterosexual",..: 2 2 2 2 1 1 1 NA 2 2 ...
##  $ PregnantNow     : Factor w/ 3 levels "Yes","No","Unknown": NA NA NA NA NA NA NA NA NA NA ...
```

```
tree <- rpart(SleepTrouble ~ SleepHrsNight + Depressed, data = NHANES_new,
              parms = list(split = "gini"))
plot(as.party(tree),type = "simple")
```



The best splits were: sleeping at least 5.5 hours per night, which resulted in a prediction of No sleep trouble; sleeping less than 5.5 hours with no depression resulted in a prediction of no sleep trouble; and sleeping less than 5.5 hours with several or majority of days depressed predicted sleep trouble.

```
NHANES_new %>%
  filter(is.na(SleepTrouble) == F) %>%
  group_by(SleepTrouble) %>%
  summarise(n = n()) %>%
  mutate(pct = n/sum(n))
```

```
## # A tibble: 2 x 3
##   SleepTrouble     n   pct
##   <fct>        <int> <dbl>
## 1 No            5799 0.746
## 2 Yes           1973 0.254
```

```
set.seed(1234)
r <- nrow(NHANES_new)
test <- sample.int(r, size = round(0.25 * r))
train <- NHANES_new[-test, ]
#nrow(train)
#is.na(NHANES_new$SleepTrouble)
test_data <- NHANES_new[test, ]
#nrow(test_data)
```

```
c_tree <- rpart(SleepTrouble ~ SleepHrsNight + Depressed, data = train,
                parms = list(split = "gini"))
p_tree <- predict(object = c_tree, newdata = test_data, type = "prob")
confusion_matrix <- table(p_tree[,2] >= 0.5,test_data$SleepTrouble)
row.names(confusion_matrix) <- c("No","Yes")
confusion_matrix
```

```
##
##         No  Yes
##   No  1415  449
##   Yes   27   52
```

```
sensit_50 <- confusion_matrix[4]/(confusion_matrix[4] + confusion_matrix[3])
print(paste(’sensitivity:’,sensit_50))
```

```
## [1] "sensitivity: 0.103792415169661"
```

```
specif_50 <- confusion_matrix[1]/(confusion_matrix[1] + confusion_matrix[2])
print(paste("specificity:",specif_50))
```

```
## [1] "specificity: 0.98127600554785"
```

```
fpr_50 <- 1 - specif_50
print(paste("false positive rate:",fpr_50))
```

```
## [1] "false positive rate: 0.0187239944521498"
```

```
fnr_50 <- 1 - sensit_50
print(paste("false negative rate:",fnr_50))
```

```
## [1] "false negative rate: 0.896207584830339"
```

```
accuracy_50 <- (confusion_matrix[1] +
                  confusion_matrix[4])/sum(confusion_matrix)
print(paste("Accuracy:",accuracy_50))
```

```
## [1] "Accuracy: 0.755018013381369"
```

Cut-point using 0.25

```
confusion_matrix_2 <- table(p_tree[,2] >= 0.25,test_data$SleepTrouble)
row.names(confusion_matrix_2) <- c("No","Yes")
confusion_matrix_2
```

```
##
##          No  Yes
##   No   1320  371
##   Yes   122  130
```

```
sensitivity_data2 <- confusion_matrix_2[4]/(confusion_matrix_2[4] + confusion_matrix_2[3])
print(paste('Sensitivity:',sensitivity_data2))
```

```
## [1] "Sensitivity: 0.259481037924152"
```

```
specificity_data2 <- confusion_matrix_2[1]/(confusion_matrix_2[1] + confusion_matrix_2[2])
print(paste('Specificity:',specificity_data2))
```

```
## [1] "Specificity: 0.915395284327323"
```

```
fpr2 <- 1 - specificity_data2
print(paste('false positive rate:',fpr2))
```

```
## [1] "false positive rate: 0.0846047156726768"
```

```
fnr <- 1 - sensitivity_data2
print(paste('false negative rate:',fnr))
```

```
## [1] "false negative rate: 0.740518962075848"
```

```
accuracy2 <- (confusion_matrix_2[1]
              + confusion_matrix_2[4])/sum(confusion_matrix_2)
print(paste('accuracy:',accuracy2))
```

```
## [1] "accuracy: 0.746268656716418"
```

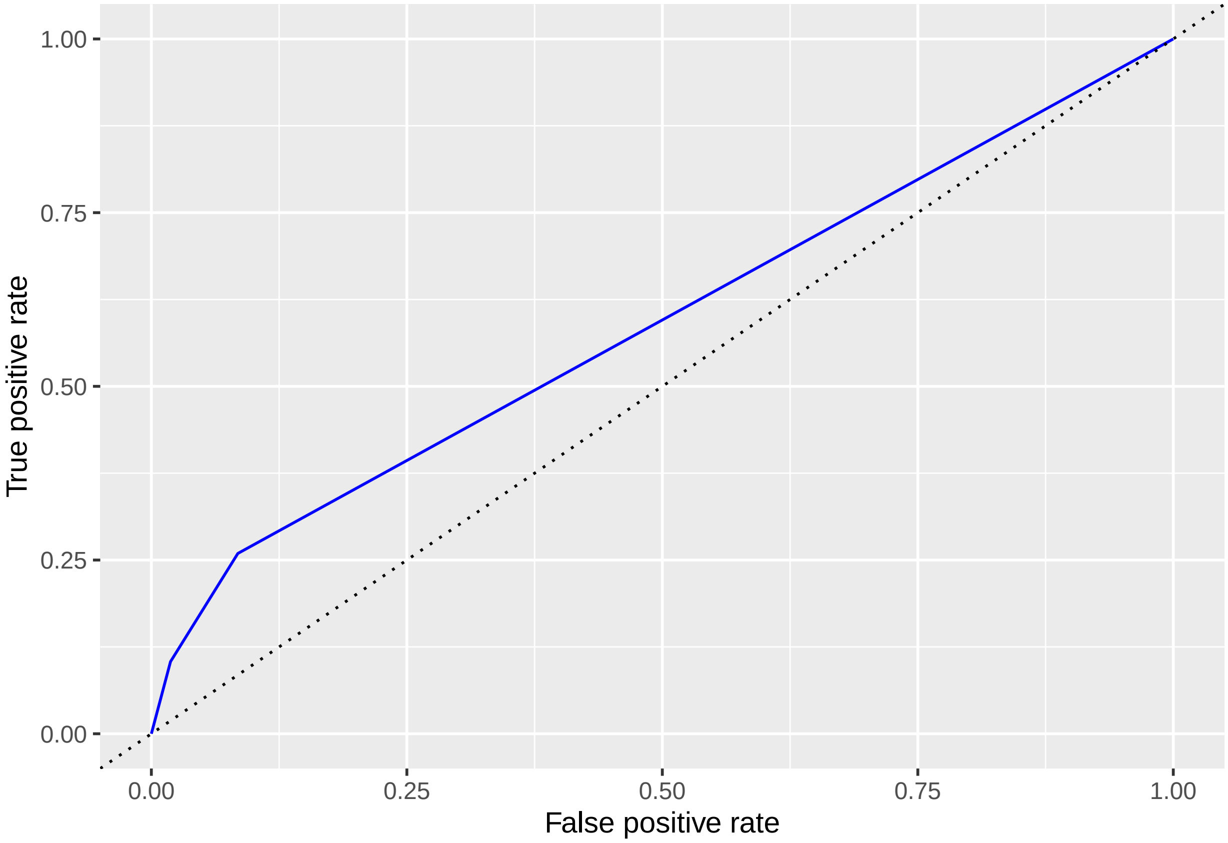
```
library(ROCR)
```

```
## Warning: package 'ROCR' was built under R version 3.6.2

## Loading required package: gplots

## Warning: package 'gplots' was built under R version 3.6.3

##
## Attaching package: 'gplots'

## The following object is masked from 'package:stats':
##
##     lowess
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.6.3
```

```
p_tree <- predict(object = tree, newdata = test_data, type = "prob")
predicted <- ROCR::prediction(predictions = p_tree[,2], test_data$SleepTrouble)
perf <- ROCR::performance(predicted, 'tpr', 'fpr')
pf <- data.frame(perf@x.values, perf@y.values)
names(pf) <- c("fpr", "tpr")
roc <- ggplot(data = pf, aes(x = fpr, y = tpr)) +
  geom_line(color = "blue") + geom_abline(intercept = 0, slope = 1, lty = 3) +
  ylab(perf@y.name) + xlab(perf@x.name)
roc
```



```
opt.cut <- function(perf){
  cut.ind <- mapply(FUN = function(x,y,p){d=(x-0)^2+(y-1)^2
  # We compute the distance of all the points from the corner point [1,0]
  ind<- which(d==min(d)) # We find the index of the point that is closest to
  #the corner
  c(recall = y[[ind]], specificity = 1-x[[ind]],cutoff = p[[ind]])},perf@x.values,
  perf@y.values,perf@alpha.values)
}
new_cut <- opt.cut(perf)
new_cut
```

```
##                   [,1]
## recall      0.2594810
## specificity 0.9153953
## cutoff      0.4088235
```

Cutoff is at 0.408. Since better recall and lower FPR.

```
tree_full <- rpart(SleepTrouble ~ ., data = train, parms = list(split = "gini"))
predicted_tree_full <- predict(object = tree_full, newdata = test_data,type = "prob")
confusion_matrix3 <- table(predicted_tree_full[,2] >= 0.5,test_data$SleepTrouble)
row.names(confusion_matrix3) <- c("No","Yes")
confusion_matrix3
```

```
##
##         No  Yes
##   No  1407  436
##   Yes   35   65
```

```
Acc<-(confusion_matrix3[1] + confusion_matrix3[4])/sum(confusion_matrix3)
Acc
```

```
## [1] 0.7575914
```

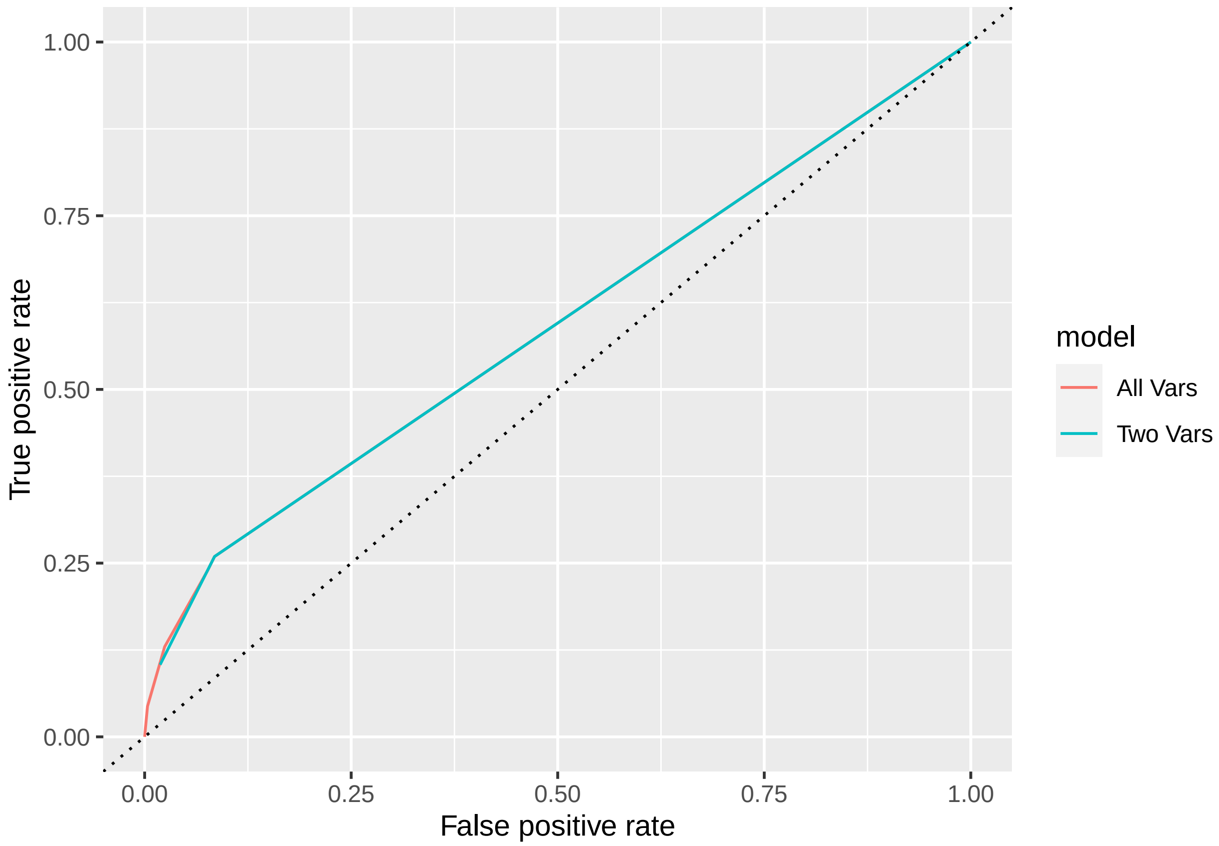
```
pred <- ROCR::prediction(predictions = predicted_tree_full[,2],
                         test_data$SleepTrouble)
perf <- ROCR::performance(pred, 'tpr', 'fpr')
pf_full <- data.frame(perf@x.values, perf@y.values)
names(pf_full) <- c("fpr", "tpr")
rbind(pf_full,pf)
```

```
##            fpr        tpr
## 1  0.000000000 0.00000000
## 2  0.003467406 0.04391218
## 3  0.024271845 0.12974052
## 4  0.076282940 0.23952096
## 5  0.084604716 0.25948104
## 6  1.000000000 1.00000000
## 7  0.000000000 0.00000000
## 8  0.018723994 0.10379242
## 9  0.084604716 0.25948104
## 10 1.000000000 1.00000000
```

```
c(rep("All Vars",7),rep("Two Vars",3))
```

```
##  [1] "All Vars" "All Vars" "All Vars" "All Vars" "All Vars" "All Vars"
##  [7] "All Vars" "Two Vars" "Two Vars" "Two Vars"
```

```r
plot_dat <- cbind(rbind(pf_full,pf), model = c(rep("All Vars",7),
                                                rep("Two Vars",3)))
ggplot(data = plot_dat, aes(x = fpr, y = tpr, colour = model)) +
  geom_line() + geom_abline(intercept = 0, slope = 1, lty = 3) +
  ylab(perf@y.name) +
  xlab(perf@x.name)
```



## Both trees have almost the same performance. Even though ROC curves can be seen overlapping but accuracy of model with all variables is slighlty more than the model with two variables.Difference in Accuracy: 0.758-0.746 = 0.012