project

Brian Lin

February 14, 2018

```r
library(rJava, warn.conflicts = FALSE, quietly=TRUE)
library(xlsx, warn.conflicts = FALSE, quietly=TRUE)
library(stringr, warn.conflicts = FALSE, quietly=TRUE)
library(dplyr, warn.conflicts = FALSE, quietly=TRUE)
library(readr, warn.conflicts = FALSE, quietly=TRUE)
library(randomForestSRC, warn.conflicts = FALSE, quietly=TRUE)
library(ggplot2, warn.conflicts = FALSE, quietly=TRUE)
library(ggthemes, warn.conflicts = FALSE, quietly=TRUE)
library(caret, warn.conflicts = FALSE, quietly=TRUE)
library(tidyr, warn.conflicts = FALSE, quietly=TRUE)
library(scales, warn.conflicts = FALSE, quietly=TRUE)
library(data.table, warn.conflicts = FALSE, quietly=TRUE)
library(effects, warn.conflicts = FALSE, quietly=TRUE)
library(gridExtra, warn.conflicts = FALSE, quietly=TRUE)
library(ggRandomForests, warn.conflicts = FALSE, quietly=TRUE )
library(ROCR, warn.conflicts = FALSE, quietly=TRUE)
library(ggpubr, warn.conflicts = FALSE, quietly=TRUE)
library(grid, warn.conflicts = FALSE, quietly=TRUE)
```

```r
#Functions

#AccuracyCutoffInfo, ConfusionMatrixInfo, ROCInfo function is completetly coded by github user ethen818

#The AccuracyCutoffInfo, ConfusionMatrixInfo, and ROCInfo functions in this Rmarkdown are a modified ve

#The following function can be found at the following github page
#https://github.com/ethen8181/machine-learning/blob/master/unbalanced/unbalanced_code/unbalanced_functi
#Credit to user ethen8181 for creating these functions.


# ------------------------------------------------------------------------------------------
# [AccuracyCutoffInfo] :
# Obtain the accuracy on the trainining and testing dataset.
# for cutoff value ranging from .4 to .8 ( with a .05 increase )
# @train   : your data.table or data.frame type training data ( assumes you have the predicted score in
# @test    : your data.table or data.frame type testing data
# @predict : prediction's column name (assumes the same for training and testing set)
# @actual  : actual results' column name
# returns  : 1. data : a data.table with three columns.
#                      each row indicates the cutoff value and the accuracy for the
#                      train and test set respectively.
#           2. plot : plot that visualizes the data.table

AccuracyCutoffInfo <- function( train, test, predict, actual )
{
  # change the cutoff value's range as you please
  cutoff <- seq( .05, 1, by = .025 )
```

```
  accuracy <- lapply( cutoff, function(c)
  {
    train_prediction <- as.factor(as.numeric( train[[predict]] > c ))
    test_prediction <- as.factor(as.numeric( test[[predict]] > c ))

    levels(train_prediction) <- c(levels(train[[actual]][1]),levels(train[[actual]])[2])
    levels(test_prediction) <- c(levels(test[[actual]][1]),levels(test[[actual]])[2])


    # use the confusionMatrix from the caret package
    cm_train <- confusionMatrix( train_prediction, train[[actual]] )
    cm_test  <- confusionMatrix( test_prediction, test[[actual]] )

    dt <- data.table( cutoff = c,
                      train  = cm_train$overall[["Accuracy"]],
                      test   = cm_test$overall[["Accuracy"]] )
    return(dt)
  }) %>% rbindlist()

  # visualize the accuracy of the train and test set for different cutoff value
  # accuracy in percentage.
  accuracy_long <- gather( accuracy, "data", "accuracy", -1 )

  plot <- ggplot( accuracy_long, aes( cutoff, accuracy, group = data, color = data ) ) +
    geom_line( size = 1 ) + geom_point( size = 3 ) +
    scale_y_continuous( label = percent ) +
    ggtitle( "Train/Test Accuracy for Different Cutoff" ) +
    scale_x_continuous(breaks=seq(0, 1, 0.1)) +
    theme_bw()

  return( list( data = accuracy, plot = plot ) )
}


#----------------------

# ------------------------------------------------------------------------------------------
# [ConfusionMatrixInfo] :
# Obtain the confusion matrix plot and data.table for a given
# dataset that already consists the predicted score and actual outcome.
# @data    : your data.table or data.frame type data that consists the column
#            of the predicted score and actual outcome
# @predict : predicted score's column name
# @actual  : actual results' column name
# @cutoff  : cutoff value for the prediction score
# return   : 1. data : a data.table consisting of three column
#                      the first two stores the original value of the prediction and actual outcome fro
#                      the passed in data frame, the third indicates the type, which is after choosing
#                      cutoff value, will this row be a true/false positive/ negative
#            2. plot : plot that visualizes the data.table

ConfusionMatrixInfo <- function( data, predict, actual, cutoff )
{
```

```r
  # extract the column ;
  # relevel making 1 appears on the more commonly seen position in
  # a two by two confusion matrix
  predict <- data[[predict]]
  temp_data <- as.factor( as.numeric(data[[actual]]) )
  levels(temp_data) <- c(0,1)
  actual  <- relevel(temp_data, "1")

  result <- data.table( actual = actual, predict = predict )

  # caculating each pred falls into which category for the confusion matrix
  result[ , type := ifelse( predict >= cutoff & actual == 1, "TP",
                            ifelse( predict >= cutoff & actual == 0, "FP",
                                    ifelse( predict <  cutoff & actual == 1, "FN", "TN" ) ) ) %>% as.fa

  # jittering : can spread the points along the x axis
  plot <- ggplot( result, aes( actual, predict, color = type ) ) +
    geom_violin( fill = "white", color = NA ) +
    geom_jitter( shape = 1 ) +
    geom_hline( yintercept = cutoff, color = "blue", alpha = 0.6 ) +
    scale_y_continuous( limits = c( 0, 1 ) ) +
    scale_color_discrete( breaks = c( "TP", "FN", "FP", "TN" ) ) + # ordering of the legend
    guides( col = guide_legend( nrow = 2 ) ) + # adjust the legend to have two rows
    ggtitle( sprintf( "Confusion Matrix with Cutoff at %.2f", cutoff ) )

  return( list( data = result, plot = plot ) )
}



# ------------------------------------------------------------------------------------------
# [ROCInfo] :
# Pass in the data that already consists the predicted score and actual outcome.
# to obtain the ROC curve
# @data    : your data.table or data.frame type data that consists the column
#            of the predicted score and actual outcome
# @predict : predicted score's column name
# @actual  : actual results' column name
# @cost.fp : associated cost for a false positive
# @cost.fn : associated cost for a false negative
# return   : a list containing
#            1. plot        : a side by side roc and cost plot, title showing optimal cutoff value
#                             title showing optimal cutoff, total cost, and area under the curve (auc)
#            2. cutoff      : optimal cutoff value according to the specified fp/fn cost
#            3. totalcost   : total cost according to the specified fp/fn cost
#            4. auc         : area under the curve
#            5. sensitivity : TP / (TP + FN)
#            6. specificity : TN / (FP + TN)

ROCInfo <- function( data, predict, actual, cost.fp, cost.fn )
{
  # calculate the values using the ROCR library
  # true positive, false postive
```

```r
pred <- prediction( data[[predict]], data[[actual]] )
perf <- performance( pred, "tpr", "fpr" )
roc_dt <- data.frame( fpr = perf@x.values[[1]], tpr = perf@y.values[[1]] )

# cost with the specified false positive and false negative cost
# false postive rate * number of negative instances * false positive cost +
# false negative rate * number of positive instances * false negative cost
cost <- perf@x.values[[1]] * cost.fp * sum( data[[actual]] == 0 ) +
  ( 1 - perf@y.values[[1]] ) * cost.fn * sum( data[[actual]] == 1 )

cost_dt <- data.frame( cutoff = pred@cutoffs[[1]], cost = cost )

# optimal cutoff value, and the corresponding true positive and false positive rate
best_index  <- which.min(cost)
best_cost   <- cost_dt[ best_index, "cost" ]
best_tpr    <- roc_dt[ best_index, "tpr" ]
best_fpr    <- roc_dt[ best_index, "fpr" ]
best_cutoff <- pred@cutoffs[[1]][ best_index ]

# area under the curve
auc <- performance( pred, "auc" )@y.values[[1]]

# normalize the cost to assign colors to 1
normalize <- function(v) ( v - min(v) ) / diff( range(v) )

# create color from a palette to assign to the 100 generated threshold between 0 ~ 1
# then normalize each cost and assign colors to it, the higher the blacker
# don't times it by 100, there will be 0 in the vector
col_ramp <- colorRampPalette( c( "green", "orange", "red", "black" ) )(100)
col_by_cost <- col_ramp[ ceiling( normalize(cost) * 99 ) + 1 ]

roc_plot <- ggplot( roc_dt, aes( fpr, tpr ) ) +
  geom_line( color = rgb( 0, 0, 1, alpha = 0.3 ) ) +
  geom_point( color = col_by_cost, size = 4, alpha = 0.2 ) +
  geom_segment( aes( x = 0, y = 0, xend = 1, yend = 1 ), alpha = 0.8, color = "royalblue" ) +
  labs( title = "ROC", x = "False Postive Rate", y = "True Positive Rate" ) +
  geom_hline( yintercept = best_tpr, alpha = 0.8, linetype = "dashed", color = "steelblue4" ) +
  geom_vline( xintercept = best_fpr, alpha = 0.8, linetype = "dashed", color = "steelblue4" ) +
  theme_bw()

cost_plot <- ggplot( cost_dt, aes( cutoff, cost ) ) +
  geom_line( color = "blue", alpha = 0.5 ) +
  geom_point( color = col_by_cost, size = 4, alpha = 0.5 ) +
  ggtitle( "Cost" ) +
  scale_y_continuous( labels = comma ) +
  geom_vline( xintercept = best_cutoff, alpha = 0.8, linetype = "dashed", color = "steelblue4" ) +
  theme_bw()

# the main title for the two arranged plot
sub_title <- sprintf( "Cutoff at %.2f - Total Cost = %.2f, AUC = %.3f",
                      best_cutoff, best_cost, auc )

# arranged into a side by side plot
```

```r
  plot <- arrangeGrob( roc_plot, cost_plot, ncol = 2,
                     top = textGrob( sub_title, gp = gpar( fontsize = 16, fontface = "bold" ) ) )

  return( list( plot          = plot,
               cutoff    = best_cutoff,
               totalcost   = best_cost,
               auc         = auc,
               sensitivity = best_tpr,
               specificity = 1 - best_fpr ) )
}

#delete_dup

#Some varaibles are forced into the model regardless of variable section result
#If the forced variable ended up being selected, this model will removed the duplicated variable.

delete_dup <- function(subset, data){
  remove <- c()
  for(i in 1:length(subset)){
    result <- str_detect(subset[i],names(data))
    for(j in 1:length(result)){
      if(result[j]){
        remove <- c(remove,i)
      }
    }
  }
  if(is.null(remove))
    return(subset)
  subset <- subset[-c(remove)]
  return(subset)
}


#data = data file
#Predition: predicted result
#response: The name of response variable
#cut_off: probabilty cut off point

Classify <- function(data, prediction,response, cut_off ){
  for(i in 1:length(prediction)){
    if(prediction[i] < cut_off){
      prediction[i] <- levels(data[[response]])[1]
    } else{
      prediction[i] <- levels(data[[response]])[2]
    }
  }

  prediction <- as.factor(prediction)
  levels(prediction) <- c(levels(data[[response]])[1],levels(data[[response]])[2])
  confuseion_matrix <- table(data[[response]],prediction)
  print(confuseion_matrix)
  Accuracy <- (confuseion_matrix[1,1] + confuseion_matrix[2,2])/sum(confuseion_matrix)
  TPR <- confuseion_matrix[2,2] / (confuseion_matrix[2,2] + confuseion_matrix[2,1])
```

```r
  return(cat(paste("The accuracy is", round(Accuracy*100,3),"%.\nThe True positive rate is", round(TPR*
}



#K fold K = 10

#data = data using for prediction
#response = name of the response variable
#cut off = probability cut off point
#interaction = you can type addition interaction term in text
#Example
#cv.error(CNP_logi_subset,"Subject_Type","+Age*Auditory.global_eff", 0.8)


cv.error <- function(data, response, interaction = "", cut_off = 0.5){

  #generate random seeds
  r <- runif(1,0,9999)
  set.seed(r)
  folds <- createFolds(data[[response]],k = 10)
  Accuracy <- rep(NA,10)
  TPR <- rep(NA,10)

  for(i in 1:10){

    #training and testing
    train <- data[-folds[[i]],]
    test <- data[folds[[i]],]

    levels(test[[response]]) <- c(levels(data[[response]])[1],levels(data[[response]])[2])

    logi_cv <-glm(paste(response,"~.",interaction), data = train, family = "binomial")

    prediction <- predict(logi_cv, test, type = "response")
    for(j in 1:length(prediction)){
      if(prediction[j] < cut_off){
        prediction[j] <- levels(test[[response]])[1]
      } else{
        prediction[j] <- levels(test[[response]])[2]
      }
    }
    prediction <- as.factor(prediction)
    levels(prediction) <- c(levels(data[[response]])[1],levels(data[[response]])[2])

    confuseion_matrix <- table(test[[response]],prediction)
    Accuracy[i] <- (confuseion_matrix[1,1] + confuseion_matrix[2,2])/sum(confuseion_matrix)
    TPR[i] <- confuseion_matrix[2,2] / (confuseion_matrix[2,2] + confuseion_matrix[2,1])
  }
  return(list(Accuracy, TPR))
}
```

```r
#Standardized variable

Standarize <- function(data){
  for(i in 1:ncol(data)){
    if(is.numeric(data[1,i])){
      data[,i] <- (data[,i] - mean(data[,i]))/sd(data[,i])
    }
  }
  return(data)
}
```

```r
#Load data

setwd("A:/Winter 2018/Stats 141SL/project/")

#load CNP data

CNP_between <- read.table("CNP_between_nets.txt", header =  TRUE)
CNP_within <- read.table("CNP_within_nets.txt", header = TRUE)
CNPDemographic <- read.xlsx("CNPDemographicMeasures.xlsx", sheetName = "SNF")


#load COBRE data


COBRE_between <- read.table("COBRE_between_nets.txt", header = TRUE)

COBRE_within <- read.table("COBRE_within_nets.txt", header = TRUE)

COBREDemographic <- read.xlsx("COBRE INDI Additional data.xls", sheetName = "NP")

COBRE_phenotypic <- read_csv("COBRE_phenotypic_data.csv")
```

```
## Parsed with column specification:
## cols(
##   X1 = col_integer(),
##   `Current Age` = col_character(),
##   Gender = col_character(),
##   Handedness = col_character(),
##   `Subject Type` = col_character(),
##   Diagnosis = col_character()
## )
```

```r
#Data cleaning process

#Removed character string

pattern <- "[a-z]*-"

CNP_within$Subject_ID <- as.numeric(str_replace_all(CNP_within$Subject_ID
, pattern,""))

CNP_between$Subject_ID <- as.numeric(str_replace_all(CNP_between$Subject_ID
, pattern,""))
```

```r
#Merge data
CNP_within_merge <- left_join(CNP_within,CNPDemographic, by = c("Subject_ID" = "PTID"))

#summary(CNP_within_merge)

CNP_between_merge <- left_join(CNP_between,CNPDemographic, by = c("Subject_ID" = "PTID"))

#summary(CNP_between_merge)




#Revmove character string

COBRE_between$Subject_ID <- as.numeric(str_replace_all(COBRE_between$Subject_ID
, pattern,""))

COBRE_within$Subject_ID <- as.numeric(str_replace_all(COBRE_within$Subject_ID
, pattern,""))




#remove 00

pattern <- "^00"

COBREDemographic$ID <- as.numeric(str_replace_all(COBREDemographic$ID, pattern,""))


#Merge data

COBRE_within_merge <- left_join(COBRE_within,COBREDemographic, by = c("Subject_ID" = "ID"))


#summary(COBRE_within_merge)

COBRE_between_merge <- left_join(COBRE_between,COBREDemographic, by = c("Subject_ID" = "ID"))

#summary(COBRE_between_merge)

COBRE_phenotypic$Gender <- as.factor(COBRE_phenotypic$Gender)

COBRE_phenotypic <- COBRE_phenotypic %>%
  filter(!(COBRE_phenotypic$Gender == "Disenrolled"))
```

```
## Warning: package 'bindrcpp' was built under R version 3.4.2
```

```r
COBRE_phenotypic$Gender <- droplevels(COBRE_phenotypic$Gender)

colnames(COBRE_phenotypic)[1:2] <- c("Subject_ID", "Age")
```

```r
COBRE_between_merge <- merge(COBRE_between_merge,COBRE_phenotypic, all = TRUE)
COBRE_within_merge <- merge(COBRE_within_merge,COBRE_phenotypic, all = TRUE)


table(COBRE_between_merge$Diagnosis)
```

```
##
##                290.3                 295.1                 295.2
##                    1                     3                     1
##                295.3                 295.6                 295.7
##                   41                    12                     5
##   295.70 bipolar type 295.70 depressed type                 295.9
##                    1                     1                     5
##               295.92                296.26                 296.4
##                    1                     1                     1
##                  311                  None
##                    1                    72
```

```r
table(COBRE_within_merge$Diagnosis)
```

```
##
##                290.3                 295.1                 295.2
##                    1                     3                     1
##                295.3                 295.6                 295.7
##                   41                    12                     5
##   295.70 bipolar type 295.70 depressed type                 295.9
##                    1                     1                     5
##               295.92                296.26                 296.4
##                    1                     1                     1
##                  311                  None
##                    1                    72
```

```r
#CNP filter


CNP_within_merge <- CNP_within_merge %>%
  filter(Subject_Type == "Control" | Subject_Type == "Schizophrenia")

table(CNP_within_merge$Subject_Type)
```

```
##
##          ADHD       Bipolar       Control Schizophrenia
##             0             0           115            42
```

```r
CNP_between_merge <- CNP_between_merge %>%
  filter(Subject_Type == "Control" | Subject_Type == "Schizophrenia")

table(CNP_between_merge$Subject_Type)
```

```
##
##          ADHD       Bipolar       Control Schizophrenia
##             0             0           115            42
```

```r
#COBRE filter

COBRE_between_merge <- COBRE_between_merge %>%
```

```r
  filter(!(Diagnosis == 290.3 | Diagnosis == 296.26 | Diagnosis == 296.4 | Diagnosis == 311))

COBRE_within_merge <- COBRE_within_merge %>%
  filter(!(Diagnosis == 290.3 | Diagnosis == 296.26 | Diagnosis == 296.4 | Diagnosis == 311))

table(COBRE_between_merge$Diagnosis)
```

```
##
##                 295.1                 295.2                 295.3
##                     3                     1                    41
##                 295.6                 295.7   295.70 bipolar type
##                    12                     5                     1
## 295.70 depressed type                 295.9                295.92
##                     1                     5                     1
##                  None
##                    72
```

```r
table(COBRE_within_merge$Diagnosis)
```

```
##
##                 295.1                 295.2                 295.3
##                     3                     1                    41
##                 295.6                 295.7   295.70 bipolar type
##                    12                     5                     1
## 295.70 depressed type                 295.9                295.92
##                     1                     5                     1
##                  None
##                    72
```

```r
#Recoding Patients to Schizophrenia in COBRE

pattern <- "Patient"

COBRE_between_merge$Subject_Type <- str_replace_all(COBRE_between_merge$Subject_Type,  pattern,"Schizop
COBRE_within_merge$Subject_Type <- str_replace_all(COBRE_within_merge$Subject_Type,  pattern,"Schizophre

table(COBRE_between_merge$Subject_Type)
```

```
##
##       Control Schizophrenia
##            72            70
```

```r
table(COBRE_within_merge$Subject_Type)
```

```
##
##       Control Schizophrenia
##            72            70
```

```r
CNP_between_merge$Subject_Type <- droplevels(CNP_between_merge$Subject_Type)
levels(CNP_between_merge$Subject_Type)
```

```
## [1] "Control"       "Schizophrenia"
```

```r
CNP_within_merge$Subject_Type <- droplevels(CNP_within_merge$Subject_Type)
levels(CNP_within_merge$Subject_Type)
```

```
## [1] "Control"       "Schizophrenia"
```

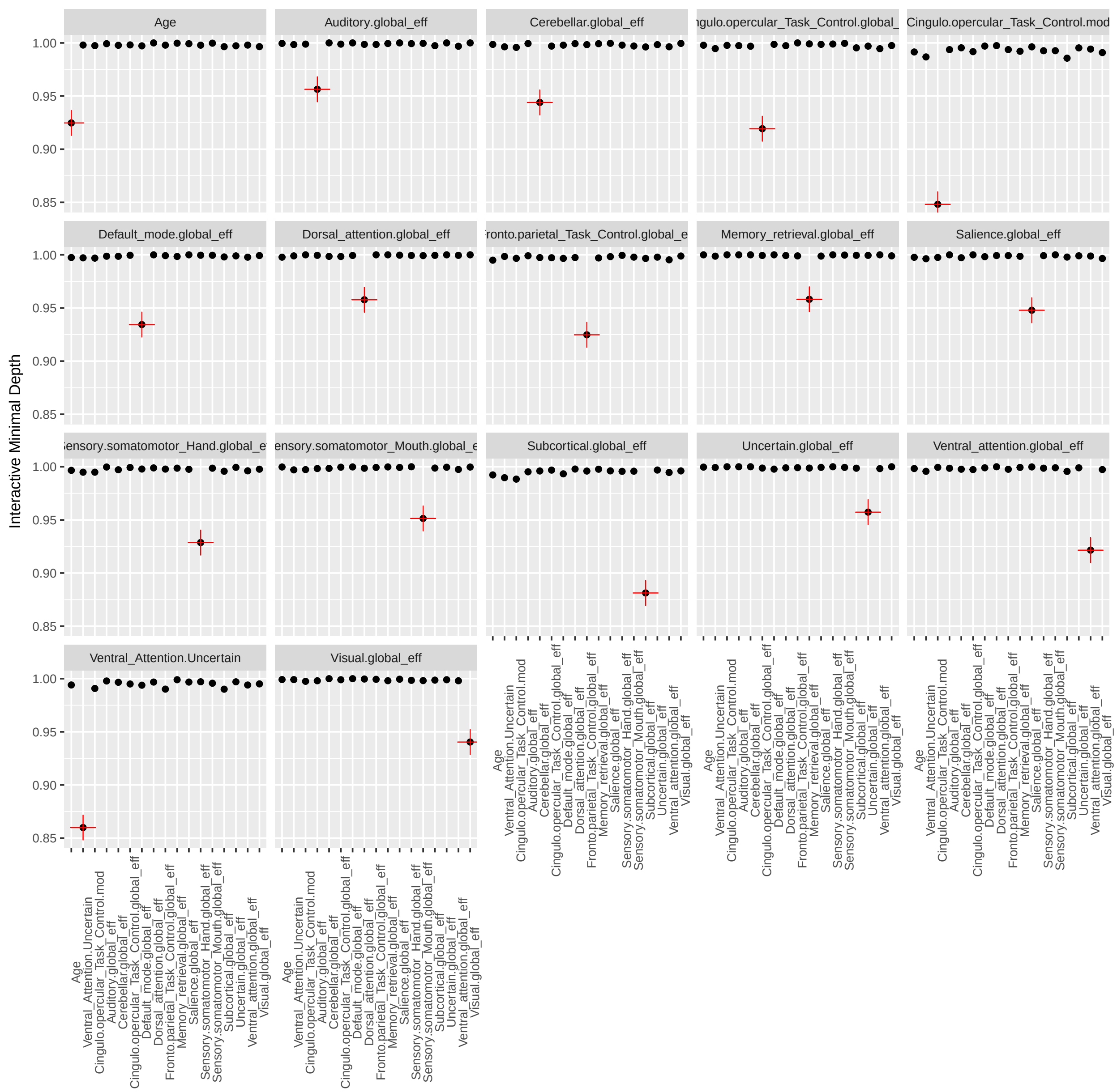
```r
#CNP between
#remove 96:98, 112
CNP_between_merge <- CNP_between_merge %>%
  select(-c(96:98,112))



#CNP within get rid of
#75 #76 #91
CNP_within_merge <- CNP_within_merge %>%
  select(-c(75:77,91))


#Merge both between and within data into CNP

CNP <- merge(CNP_between_merge,CNP_within_merge, all = TRUE)

CNP_RF_subset <- CNP %>%
  select(-c(1,5:41))





#Merge both between and within into COBRE

COBRE <- merge(COBRE_between_merge, COBRE_within_merge, all = TRUE)

#Use only the fMRI, MRI, and Age, keep global EFF

COBRE_RF_subset<- COBRE %>%
  select(-c(1,5:111))

COBRE_RF_subset$Subject_Type <- as.factor(COBRE_RF_subset$Subject_Type)
```

```r
#CNP data modeling

set.seed(4321)


rfsrc_m1 <- rfsrc(as.factor(Subject_Type)~.,data = CNP_RF_subset, na.action = c("na.omit"), ntree= 1000

max_var <- max.subtree(rfsrc_m1, conservative = TRUE)
max_var$topvars
```

```
## [1] "Ventral_Attention.Uncertain"
## [2] "Cingulo.opercular_Task_Control.mod"
```

```r
#delete duplicate entity


#Logistic Regression Model

subset1 <- as.vector(max_var$topvars)
```

```r
subset1 <- delete_dup(subset1,CNP_RF_subset[,c(1,137:150)])

CNP_logi_subset <- CNP_RF_subset[,c("Subject_Type",names(CNP_RF_subset[,c(1,137:150)]), subset1)]



#Using a previously grown forest, identify pairwise interactions for all pairs of variables from a spec

#method="maxsubtree"

#This invokes a maximal subtree analysis.

CNP_logi_subset <- na.omit(CNP_logi_subset) %>%
  Standarize()

#Find interaction
gg_int <- gg_interaction(find.interaction(rfsrc_m1,
                                          xvar.names = names(CNP_logi_subset[,-c(1)]),
                                          sorted = FALSE,
                                          verbose = FALSE))


plot(gg_int)
```




```
#Minimal depth variable interaction plot for all variables of interest.
#Higher values indicate lower interactivity with target variable marked in red.

#No interactioin found base on the result, we don't have to add interaction term


#Correlation check
high_cor <- findCorrelation(cor(CNP_logi_subset[,-c(1:2)]),cutoff = 0.75) + 2

#No potential multicollinearity problem

index <- sample(1:nrow(CNP_logi_subset), size = round(nrow(CNP_logi_subset)*0.7,0),replace = FALSE)

CNP_train <- CNP_logi_subset[index,]
```

```r
CNP_test <- CNP_logi_subset[-index,]
logi_m1 <-glm(Subject_Type~. , data = CNP_train, family = "binomial")
summary(logi_m1)
```

```
##
## Call:
## glm(formula = Subject_Type ~ ., family = "binomial", data = CNP_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6210  -0.6679  -0.3474   0.5809   3.0634
##
## Coefficients:
##                                           Estimate Std. Error z value
## (Intercept)                               -1.73948    0.36956  -4.707
## Age                                        0.83260    0.34036   2.446
## Auditory.global_eff                       -0.09186    0.35094  -0.262
## Cerebellar.global_eff                      0.15612    0.27872   0.560
## Cingulo.opercular_Task_Control.global_eff  0.03363    0.33229   0.101
## Default_mode.global_eff                    0.87062    0.42690   2.039
## Dorsal_attention.global_eff                0.43827    0.31808   1.378
## Fronto.parietal_Task_Control.global_eff    0.38999    0.42524   0.917
## Memory_retrieval.global_eff               -0.02610    0.29618  -0.088
## Salience.global_eff                        0.36833    0.34606   1.064
## Sensory.somatomotor_Hand.global_eff        0.30430    0.50027   0.608
## Sensory.somatomotor_Mouth.global_eff      -0.40188    0.37858  -1.062
## Subcortical.global_eff                     0.56609    0.36193   1.564
## Uncertain.global_eff                       0.78505    0.34559   2.272
## Ventral_attention.global_eff              -0.20361    0.31489  -0.647
## Visual.global_eff                          0.49797    0.38277   1.301
## Ventral_Attention.Uncertain               -2.28226    0.67824  -3.365
## Cingulo.opercular_Task_Control.mod         0.88433    0.32735   2.701
##                                           Pr(>|z|)
## (Intercept)                               2.51e-06 ***
## Age                                       0.014436 *
## Auditory.global_eff                       0.793522
## Cerebellar.global_eff                     0.575381
## Cingulo.opercular_Task_Control.global_eff 0.919375
## Default_mode.global_eff                   0.041408 *
## Dorsal_attention.global_eff               0.168252
## Fronto.parietal_Task_Control.global_eff   0.359080
## Memory_retrieval.global_eff               0.929783
## Salience.global_eff                       0.287175
## Sensory.somatomotor_Hand.global_eff       0.543008
## Sensory.somatomotor_Mouth.global_eff      0.288451
## Subcortical.global_eff                    0.117797
## Uncertain.global_eff                      0.023110 *
## Ventral_attention.global_eff              0.517884
## Visual.global_eff                         0.193274
## Ventral_Attention.Uncertain               0.000766 ***
## Cingulo.opercular_Task_Control.mod        0.006903 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 130.826  on 109  degrees of freedom
## Residual deviance:  91.494  on  92  degrees of freedom
## AIC: 127.49
##
## Number of Fisher Scoring iterations: 6
```

```r
round(exp(coef(logi_m1)),3)
```

```
##                               (Intercept)
##                                     0.176
##                                       Age
##                                     2.299
##                      Auditory.global_eff
##                                     0.912
##                    Cerebellar.global_eff
##                                     1.169
## Cingulo.opercular_Task_Control.global_eff
##                                     1.034
##                  Default_mode.global_eff
##                                     2.388
##            Dorsal_attention.global_eff
##                                     1.550
##   Fronto.parietal_Task_Control.global_eff
##                                     1.477
##            Memory_retrieval.global_eff
##                                     0.974
##                      Salience.global_eff
##                                     1.445
##        Sensory.somatomotor_Hand.global_eff
##                                     1.356
##       Sensory.somatomotor_Mouth.global_eff
##                                     0.669
##                   Subcortical.global_eff
##                                     1.761
##                     Uncertain.global_eff
##                                     2.193
##           Ventral_attention.global_eff
##                                     0.816
##                        Visual.global_eff
##                                     1.645
##            Ventral_Attention.Uncertain
##                                     0.102
##         Cingulo.opercular_Task_Control.mod
##                                     2.421
```

```r
anova(logi_m1, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: Subject_Type
##
```

```
## Terms added sequentially (first to last)
##
##
##                                          Df Deviance Resid. Df Resid. Dev
## NULL                                                        109    130.826
## Age                                       1   3.5821       108    127.244
## Auditory.global_eff                       1   0.3403       107    126.903
## Cerebellar.global_eff                     1   0.2195       106    126.684
## Cingulo.opercular_Task_Control.global_eff 1   0.0091       105    126.675
## Default_mode.global_eff                   1   0.0117       104    126.663
## Dorsal_attention.global_eff               1   0.2303       103    126.433
## Fronto.parietal_Task_Control.global_eff   1   0.6998       102    125.733
## Memory_retrieval.global_eff               1   0.0689       101    125.664
## Salience.global_eff                       1   0.0166       100    125.647
## Sensory.somatomotor_Hand.global_eff       1   1.5916        99    124.056
## Sensory.somatomotor_Mouth.global_eff      1   6.0241        98    118.032
## Subcortical.global_eff                    1   0.0072        97    118.025
## Uncertain.global_eff                      1   1.8716        96    116.153
## Ventral_attention.global_eff              1   1.1439        95    115.009
## Visual.global_eff                         1   0.1207        94    114.888
## Ventral_Attention.Uncertain               1  13.8856        93    101.003
## Cingulo.opercular_Task_Control.mod        1   9.5089        92     91.494
##                                           Pr(>Chi)
## NULL
## Age                                      0.0584057 .
## Auditory.global_eff                      0.5596684
## Cerebellar.global_eff                    0.6394285
## Cingulo.opercular_Task_Control.global_eff 0.9240192
## Default_mode.global_eff                  0.9137170
## Dorsal_attention.global_eff              0.6313287
## Fronto.parietal_Task_Control.global_eff  0.4028610
## Memory_retrieval.global_eff              0.7929193
## Salience.global_eff                      0.8975062
## Sensory.somatomotor_Hand.global_eff      0.2071033
## Sensory.somatomotor_Mouth.global_eff     0.0141116 *
## Subcortical.global_eff                   0.9321680
## Uncertain.global_eff                     0.1712969
## Ventral_attention.global_eff             0.2848198
## Visual.global_eff                        0.7282840
## Ventral_Attention.Uncertain              0.0001943 ***
## Cingulo.opercular_Task_Control.mod       0.0020447 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
#R-squared

R_squared <- 1 - (summary(logi_m1)[[4]]/summary(logi_m1)[[8]])
R_squared
```

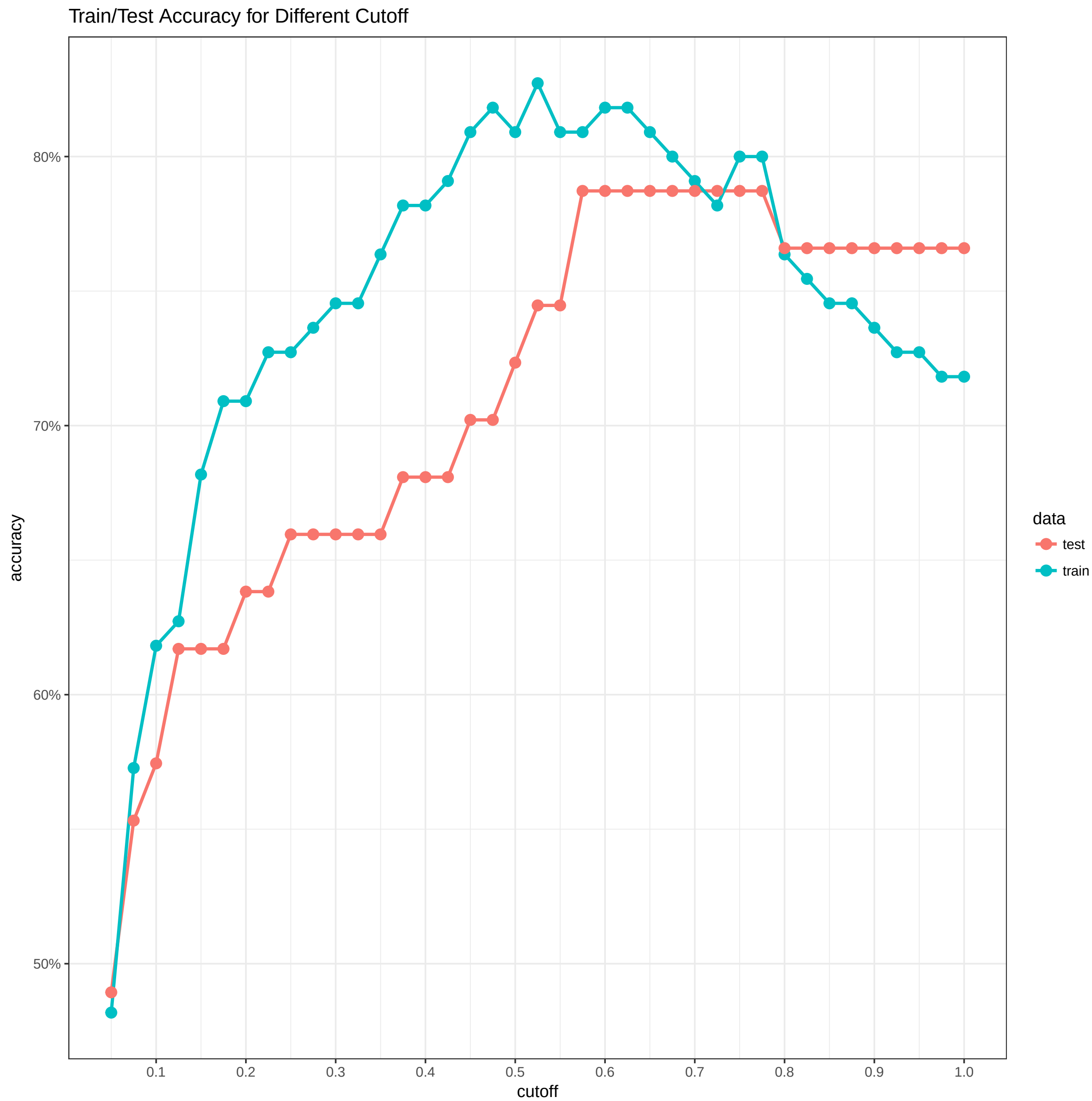
```
## [1] 0.3006434
```

```r
#70/30 CV check

#Train
CNP_train$prediction <- predict(logi_m1, CNP_train, type = "response")
```

```r
#Test
CNP_test$prediction <- predict(logi_m1, CNP_test, type = "response")

prop.table(table(CNP$Subject_Type))
```

```
##
##       Control Schizophrenia
##     0.7324841     0.2675159
```

```r
accuracy_info <- AccuracyCutoffInfo( train = CNP_train, test = CNP_test,
                                     predict = "prediction", actual = "Subject_Type" )

accuracy_info$plot
```



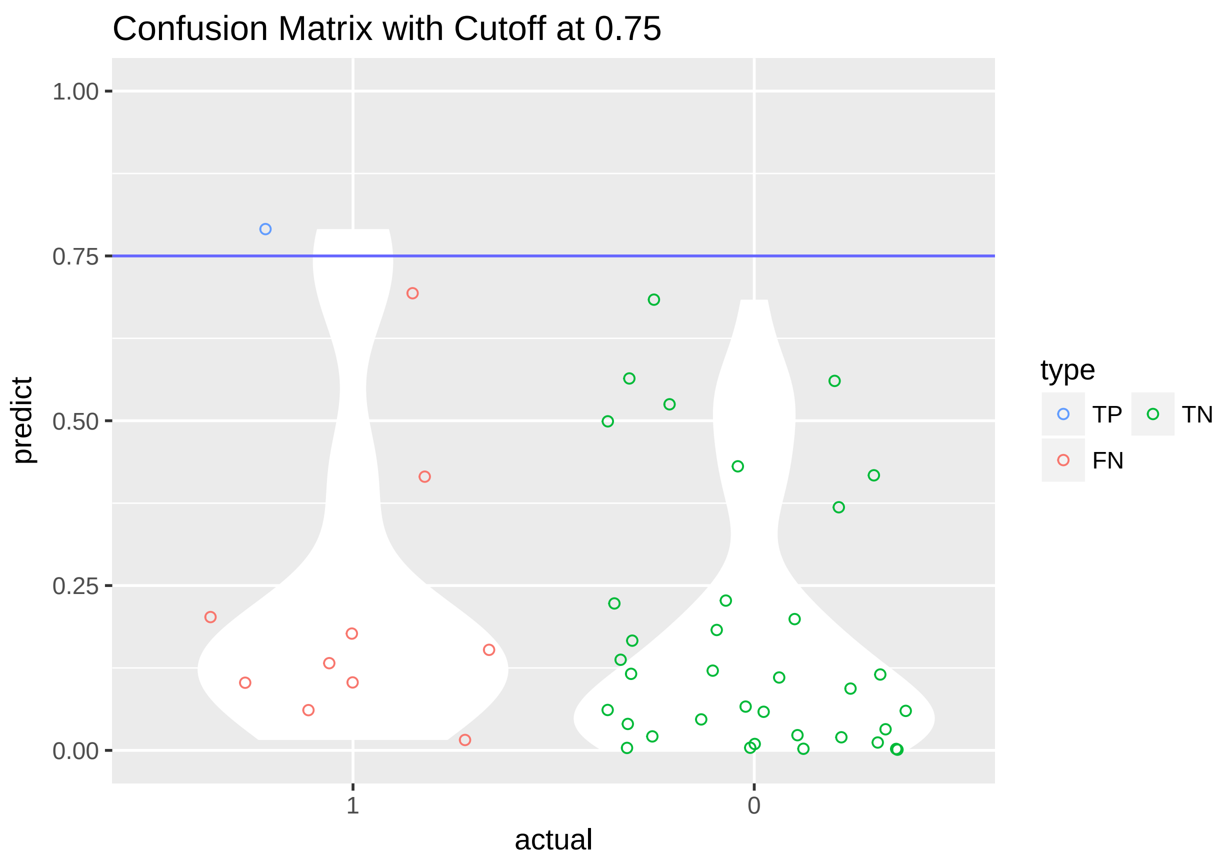
```r
Classify(CNP_train, CNP_train$prediction,"Subject_Type", 0.75 )
```

```
##                 prediction
##                  Control Schizophrenia
##   Control             79             0
##   Schizophrenia       22             9
## The accuracy is 80 %.
## The True positive rate is 29.032 %
```

```r
Classify(CNP_test, CNP_test$prediction,"Subject_Type", 0.75 )
```

```
##                 prediction
##                  Control Schizophrenia
##   Control             36             0
##   Schizophrenia       10             1
## The accuracy is 78.723 %.
## The True positive rate is 9.091 %
```

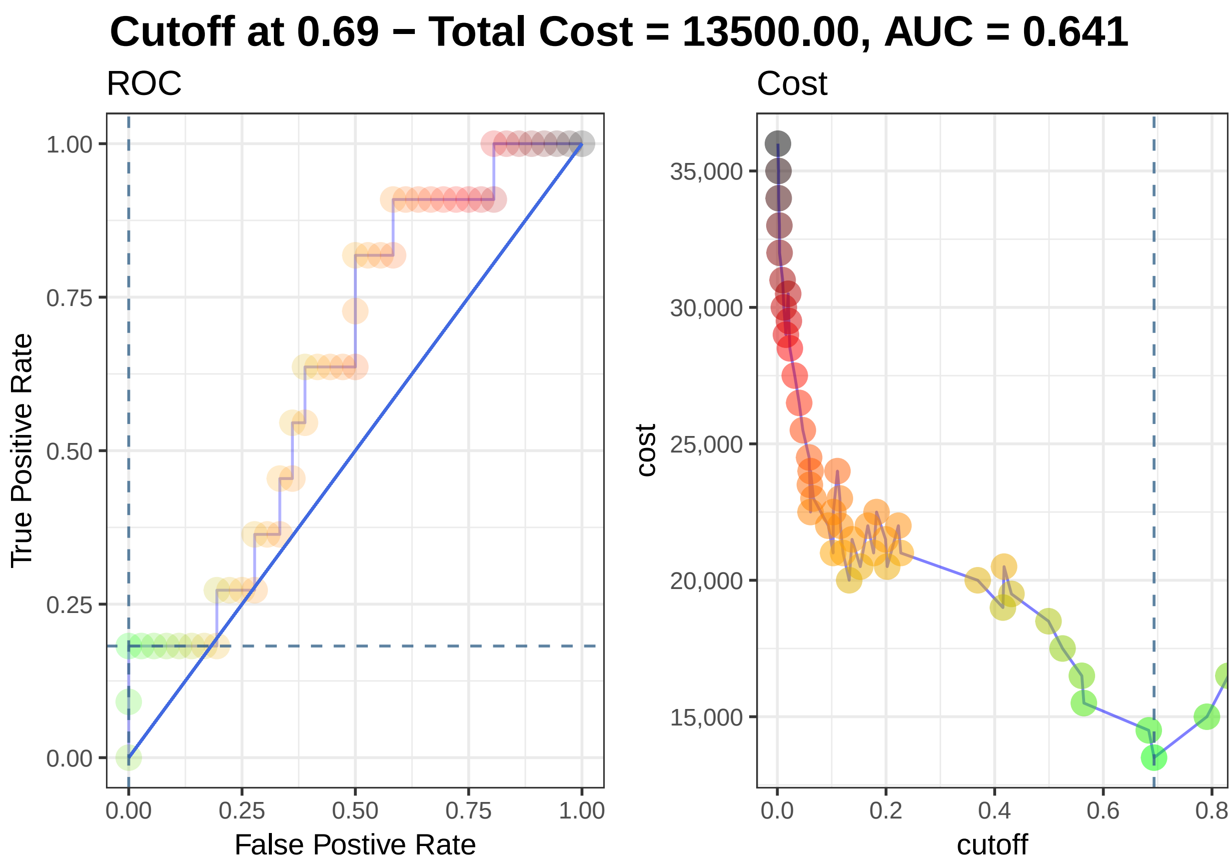
```r
set.seed(4321)
#CNP ROC search for better True positive rate.

#cutoff : Optimal cutoff value according to the specified FP and FN cost .
#totalcost : Total cost according to the specified FP and FN cost.
#auc : Area under the curve.
#sensitivity : TP / (TP + FN) for the optimal cutoff.
#specificity : TN / (FP + TN) for the optimal cutoff.


cm_info <- ConfusionMatrixInfo(data = CNP_test, predict = "prediction", actual = "Subject_Type", 0.75)

cm_info$plot
```




```r
invisible(dev.off())

roc_info <- ROCInfo( data = cm_info$data, predict = "predict",
                     actual = "actual", cost.fp = 1000, cost.fn = 1500 )

#Optimal cutoff for True positive rate
roc_info$cutoff
```

```
## [1] 0.6933646
```

```r
grid.draw(roc_info$plot)
```




```
#CNP model k fold CV check
set.seed(4321)

#Optimal cutoff for Accuracy
result <- cv.error(CNP_logi_subset, "Subject_Type",cut_off = roc_info$cutoff)
Accuracy.k <- result[[1]]
mean(Accuracy.k)
```

```
## [1] 0.7263725
```

```
TTP.k <- result[[2]]
mean(TTP.k)
```

```
## [1] 0.15
```

```
#Optimal cutoff for True positive rate
result <- cv.error(CNP_logi_subset, "Subject_Type",cut_off = roc_info$cutoff)
Accuracy.k <- result[[1]]
mean(Accuracy.k)
```

```
## [1] 0.7401716
```

```
TTP.k <- result[[2]]
mean(TTP.k)
```

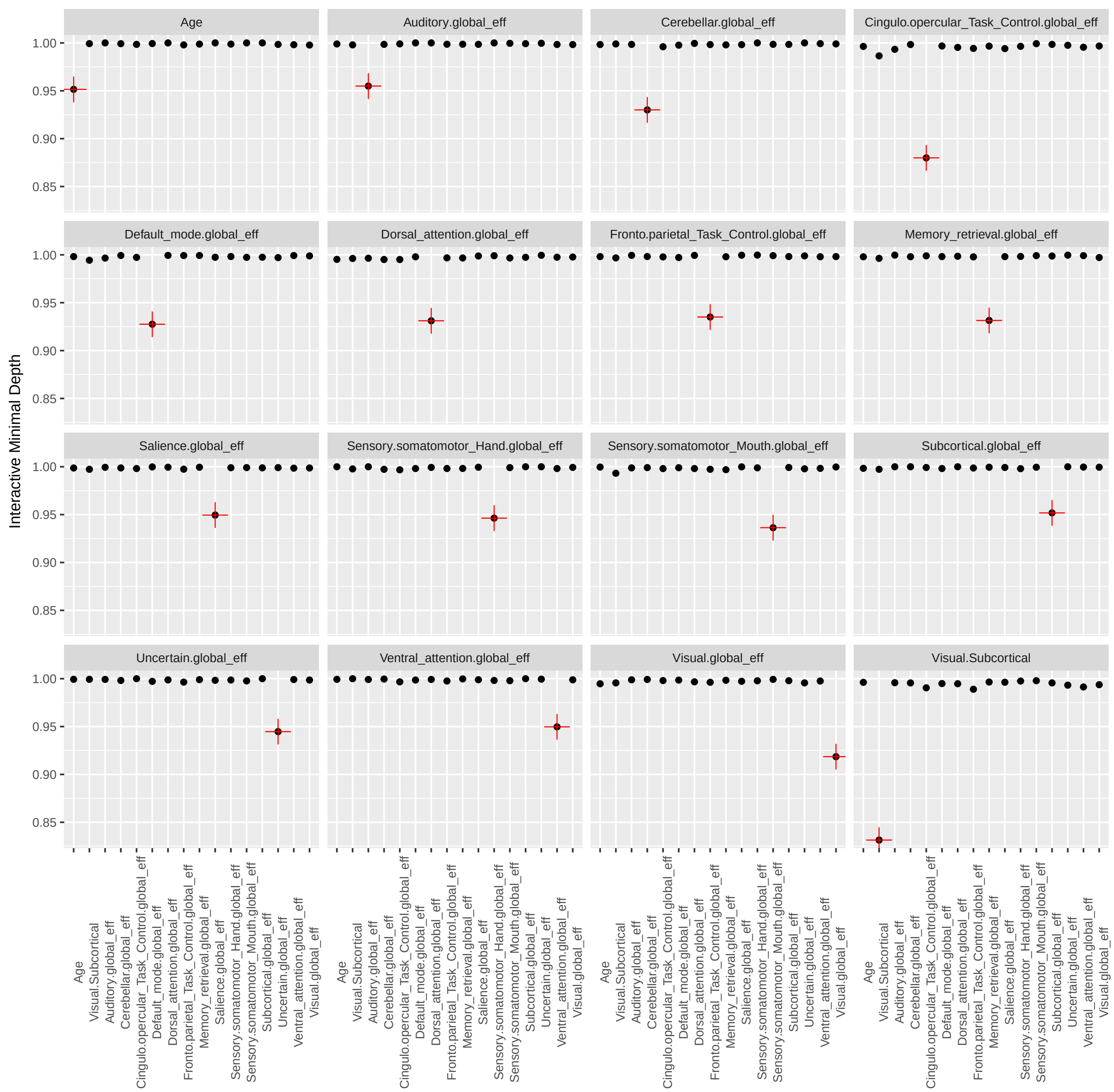
```
## [1] 0.195
```

```
set.seed(4321)
```

```r
#Random Forest variable section
rfsrc_m2 <- rfsrc(Subject_Type~.,data = COBRE_RF_subset, na.action = c("na.omit"), ntree= 1000)


max_var <- max.subtree(rfsrc_m2, conservative = TRUE)
max_var$topvars
```

```
## [1] "Visual.Subcortical"
```

```r
#delete duplicate entity


subset2 <- as.vector(max_var$topvars)

subset2 <- delete_dup(subset2,COBRE_RF_subset[,c(1,137:150)])


#Logistic Regression model


COBRE_logi_subset <- COBRE_RF_subset[,c("Subject_Type",names(COBRE_RF_subset[,c(1,137:150)]), subset2)]


COBRE_logi_subset <- na.omit(COBRE_logi_subset) %>%
  Standarize()


#Find interaction
gg_int <- gg_interaction(find.interaction(rfsrc_m2,
                                          xvar.names = names(COBRE_logi_subset[,-c(1)]),
                                          sorted = FALSE,
                                          verbose = FALSE))
plot(gg_int)
```




```
#No interactioin fund base on the result, we don't have to add interaction term


#Correlation check
high_cor <- findCorrelation(cor(COBRE_logi_subset[,-c(1:2)]),cutoff = 0.75) + 2

#No potential multicollinearity problem


index <- sample(1:nrow(COBRE_logi_subset), size = round(nrow(COBRE_logi_subset)*0.7,0),replace = FALSE)

COBRE_train <- COBRE_logi_subset[index,]
COBRE_test <- COBRE_logi_subset[-index,]
logi_m2 <-glm(Subject_Type~. , data = COBRE_train, family = "binomial")
```

```r
summary(logi_m2)
```

```
##
## Call:
## glm(formula = Subject_Type ~ ., family = "binomial", data = COBRE_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7740  -0.8927  -0.3029   0.8723   2.7946
##
## Coefficients:
##                                           Estimate Std. Error z value
## (Intercept)                               -0.13280    0.24848  -0.534
## Age                                       -0.47861    0.31704  -1.510
## Auditory.global_eff                        0.27499    0.28158   0.977
## Cerebellar.global_eff                     -0.02965    0.27095  -0.109
## Cingulo.opercular_Task_Control.global_eff -0.64555    0.33684  -1.916
## Default_mode.global_eff                   -0.41007    0.35894  -1.142
## Dorsal_attention.global_eff               -0.16624    0.28377  -0.586
## Fronto.parietal_Task_Control.global_eff   -0.59059    0.28493  -2.073
## Memory_retrieval.global_eff               -0.13569    0.27979  -0.485
## Salience.global_eff                       -0.20282    0.31132  -0.652
## Sensory.somatomotor_Hand.global_eff        0.10884    0.36779   0.296
## Sensory.somatomotor_Mouth.global_eff      -0.30390    0.31851  -0.954
## Subcortical.global_eff                    -0.27574    0.29112  -0.947
## Uncertain.global_eff                       0.35474    0.28462   1.246
## Ventral_attention.global_eff               0.40859    0.29147   1.402
## Visual.global_eff                         -0.74487    0.38911  -1.914
## Visual.Subcortical                         1.09163    0.34568   3.158
##                                           Pr(>|z|)
## (Intercept)                                0.59303
## Age                                        0.13114
## Auditory.global_eff                        0.32877
## Cerebellar.global_eff                      0.91286
## Cingulo.opercular_Task_Control.global_eff  0.05530 .
## Default_mode.global_eff                    0.25327
## Dorsal_attention.global_eff                0.55801
## Fronto.parietal_Task_Control.global_eff    0.03820 *
## Memory_retrieval.global_eff                0.62769
## Salience.global_eff                        0.51472
## Sensory.somatomotor_Hand.global_eff        0.76728
## Sensory.somatomotor_Mouth.global_eff       0.34001
## Subcortical.global_eff                     0.34356
## Uncertain.global_eff                       0.21263
## Ventral_attention.global_eff               0.16097
## Visual.global_eff                          0.05559 .
## Visual.Subcortical                         0.00159 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 136.42  on 98  degrees of freedom
## Residual deviance: 105.62  on 82  degrees of freedom
```

```
## AIC: 139.62
##
## Number of Fisher Scoring iterations: 5
```

```r
round(exp(coef(logi_m2)),3)
```

```
##                               (Intercept)
##                                     0.876
##                                       Age
##                                     0.620
##                      Auditory.global_eff
##                                     1.317
##                    Cerebellar.global_eff
##                                     0.971
## Cingulo.opercular_Task_Control.global_eff
##                                     0.524
##                  Default_mode.global_eff
##                                     0.664
##              Dorsal_attention.global_eff
##                                     0.847
##   Fronto.parietal_Task_Control.global_eff
##                                     0.554
##              Memory_retrieval.global_eff
##                                     0.873
##                      Salience.global_eff
##                                     0.816
##       Sensory.somatomotor_Hand.global_eff
##                                     1.115
##      Sensory.somatomotor_Mouth.global_eff
##                                     0.738
##                   Subcortical.global_eff
##                                     0.759
##                     Uncertain.global_eff
##                                     1.426
##             Ventral_attention.global_eff
##                                     1.505
##                        Visual.global_eff
##                                     0.475
##                       Visual.Subcortical
##                                     2.979
```

```r
anova(logi_m2, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: Subject_Type
##
## Terms added sequentially (first to last)
##
##
##                                           Df Deviance Resid. Df Resid. Dev
## NULL                                                         98     136.42
## Age                                        1   0.0417        97     136.38
```

```
## Auditory.global_eff                        1   0.3108         96      136.07
## Cerebellar.global_eff                      1   1.8761         95      134.19
## Cingulo.opercular_Task_Control.global_eff  1   3.9597         94      130.24
## Default_mode.global_eff                    1   1.2625         93      128.97
## Dorsal_attention.global_eff                1   0.4660         92      128.51
## Fronto.parietal_Task_Control.global_eff    1   1.0467         91      127.46
## Memory_retrieval.global_eff                1   0.4759         90      126.98
## Salience.global_eff                        1   0.0280         89      126.96
## Sensory.somatomotor_Hand.global_eff        1   0.0038         88      126.95
## Sensory.somatomotor_Mouth.global_eff       1   1.9567         87      125.00
## Subcortical.global_eff                     1   0.0634         86      124.93
## Uncertain.global_eff                       1   1.4932         85      123.44
## Ventral_attention.global_eff               1   2.6970         84      120.74
## Visual.global_eff                          1   2.0923         83      118.65
## Visual.Subcortical                         1  13.0328         82      105.62
##                                            Pr(>Chi)
## NULL
## Age                                       0.8382831
## Auditory.global_eff                       0.5771834
## Cerebellar.global_eff                     0.1707794
## Cingulo.opercular_Task_Control.global_eff 0.0466033 *
## Default_mode.global_eff                   0.2611782
## Dorsal_attention.global_eff               0.4948425
## Fronto.parietal_Task_Control.global_eff   0.3062618
## Memory_retrieval.global_eff               0.4902644
## Salience.global_eff                       0.8671524
## Sensory.somatomotor_Hand.global_eff       0.9505287
## Sensory.somatomotor_Mouth.global_eff      0.1618715
## Subcortical.global_eff                    0.8011719
## Uncertain.global_eff                      0.2217196
## Ventral_attention.global_eff              0.1005368
## Visual.global_eff                         0.1480421
## Visual.Subcortical                        0.0003061 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
#R-squared
R_squared <- 1 - (summary(logi_m2)[[4]]/summary(logi_m2)[[8]])
R_squared
```

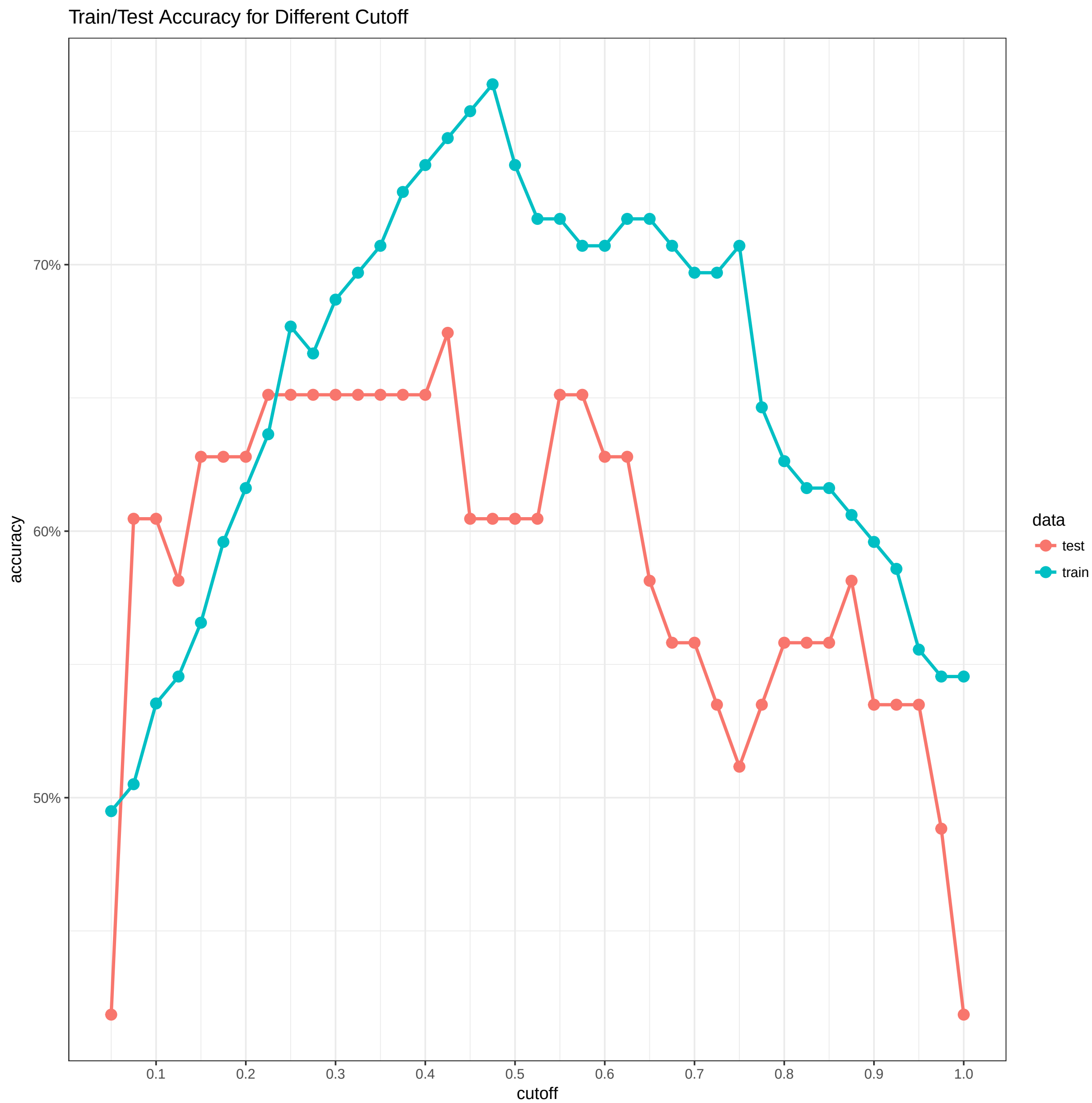
```
## [1] 0.2258151
```

```r
#70/30 CV check

#Train
COBRE_train$prediction <- predict(logi_m2, COBRE_train, type = "response")


#Test
COBRE_test$prediction <- predict(logi_m2, COBRE_test, type = "response")

prop.table(table(COBRE$Subject_Type))
```

```
##
##       Control Schizophrenia
```

```
## 	0.5070423 	0.4929577
```

```
accuracy_info <- AccuracyCutoffInfo( train = COBRE_train, test = COBRE_test,
                                     predict = "prediction", actual = "Subject_Type" )

accuracy_info$plot
```



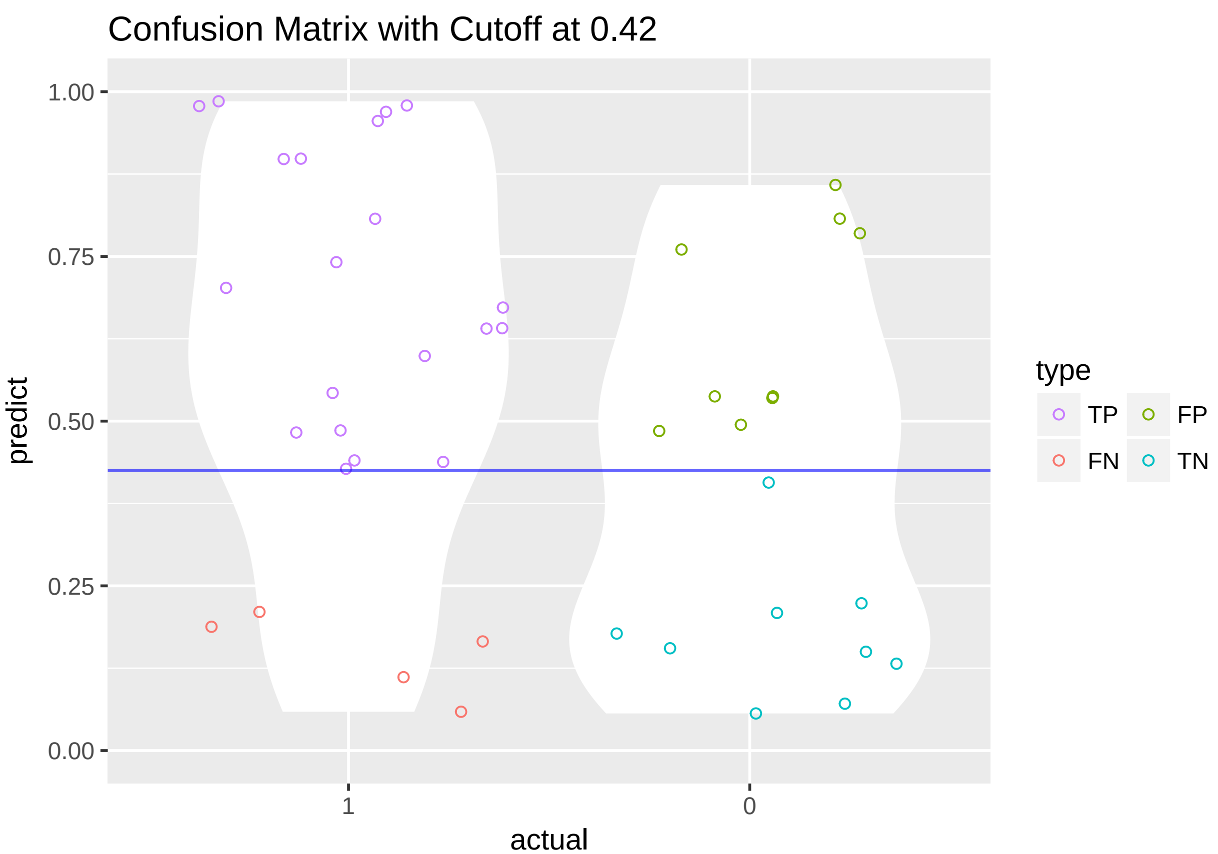

```
Classify(COBRE_train, COBRE_train$prediction,"Subject_Type", 0.425)
```

```
##                  prediction
##                   Control Schizophrenia
##    Control             37            17
##    Schizophrenia        8            37
## The accuracy is 74.747 %.
## The True positive rate is 82.222 %
```

```
Classify(COBRE_test, COBRE_test$prediction,"Subject_Type", 0.425)
```

```
##                  prediction
##                   Control Schizophrenia
##   Control               9             9
##   Schizophrenia         5            20
## The accuracy is 67.442 %.
## The True positive rate is 80 %
```

```
#COBRE ROC search for better True positive rate.

#cutoff : Optimal cutoff value according to the specified FP and FN cost .
#totalcost : Total cost according to the specified FP and FN cost.
#auc : Area under the curve.
#sensitivity : TP / (TP + FN) for the optimal cutoff.
#specificity : TN / (FP + TN) for the optimal cutoff.


cm_info <- ConfusionMatrixInfo(data = COBRE_test, predict = "prediction", actual = "Subject_Type", 0.425

cm_info$plot
```

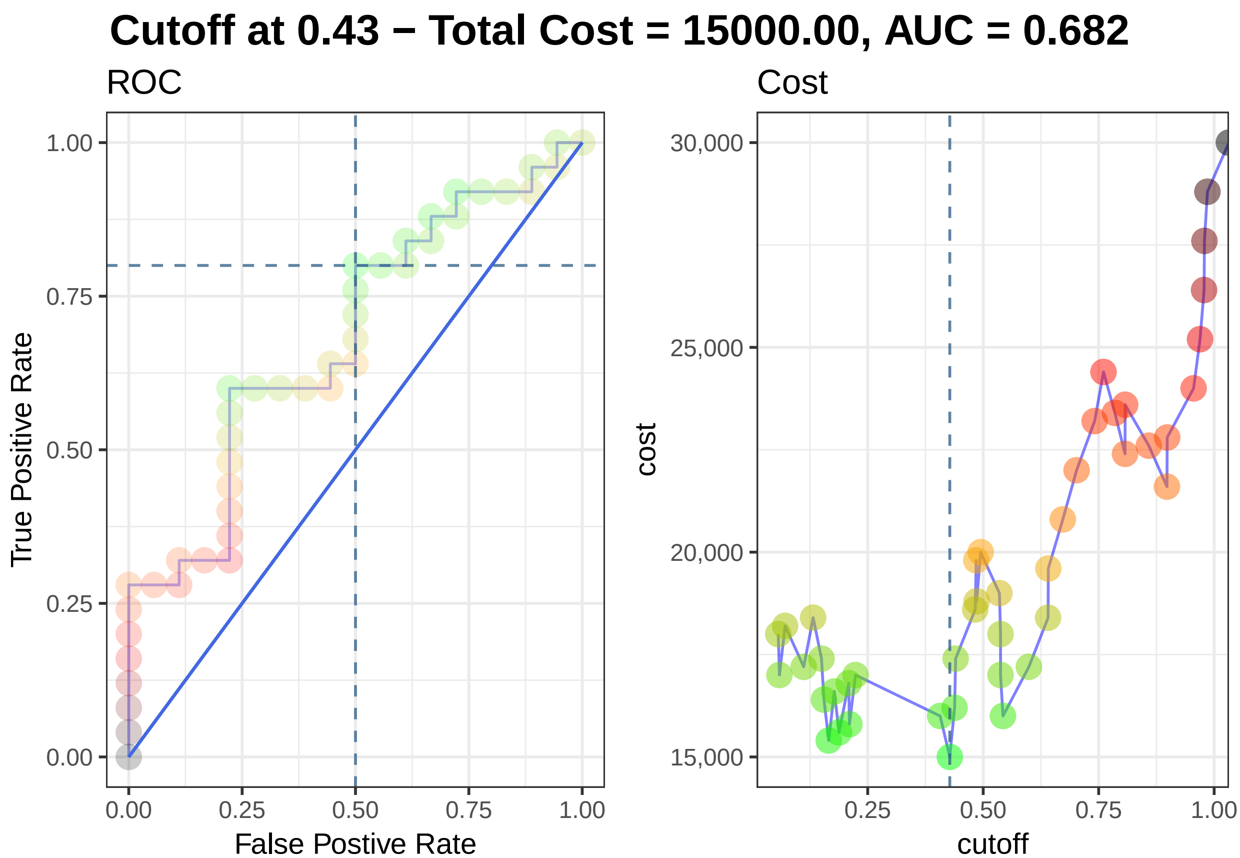



```
invisible(graphics.off())
roc_info <- ROCInfo( data = cm_info$data, predict = "predict",
                     actual = "actual", cost.fp = 1000, cost.fn = 1200 )

#Optimal cutoff for True positive rate
```

```r
roc_info$cutoff
```

```
## [1] 0.4276777
```

```r
grid.draw(roc_info$plot)
```




```r
#COBRE model k fold CV check

set.seed(4321)
#Optimal cutoff for Accuracy
result <-  cv.error(COBRE_logi_subset, "Subject_Type", cut_off = 0.425)
Accuracy.k <- result[[1]]
mean(Accuracy.k)
```

```
## [1] 0.6185714
```

```r
TTP <- result[[2]]
mean(TTP)
```

```
## [1] 0.7285714
```

```r
#Optimal cutoff for True postitive rate
result <-  cv.error(COBRE_logi_subset, "Subject_Type", cut_off = roc_info$cutoff)
Accuracy.k <- result[[1]]
mean(Accuracy.k)
```

```
## [1] 0.6480952
```

```r
TTP <- result[[2]]
mean(TTP)
```

```
## [1] 0.7285714
```

```r
#When we want optimize True positive rate, we gave up about 10% of accuracy.
```

```r
set.seed(4321)
#Fit Data into model build base on other study to test how it handles data from different study


#Fit COBRE data into CNP Model
Fit_COBRE_logi_subset <- COBRE_RF_subset[,c("Subject_Type",names(COBRE_RF_subset[,c(1,137:150)]), subse
  Standarize()

Fit_COBRE_test <- Fit_COBRE_logi_subset

invisible(rm(Fit_COBRE_logi_subset))

Fit_COBRE_test$prediction <- predict(logi_m1, Fit_COBRE_test, type = "response")
Classify(Fit_COBRE_test, Fit_COBRE_test$prediction,"Subject_Type", 0.17 )
```

```
##                 prediction
##                  Control Schizophrenia
##   Control             39            33
##   Schizophrenia       38            32
## The accuracy is 50 %.
## The True positive rate is 45.714 %
```

```r
#Fit CNP data into COBRE model
Fit_CNP_logi_subset <- CNP_RF_subset[,c("Subject_Type",names(CNP_RF_subset[,c(1,137:150)]), subset2)] %
  Standarize()

Fit_CNP_test <- Fit_CNP_logi_subset

invisible(rm(Fit_CNP_logi_subset))

Fit_CNP_test$prediction <- predict(logi_m2, Fit_CNP_test, type = "response")
Classify(Fit_CNP_test, Fit_CNP_test$prediction,"Subject_Type", cut_off = 0.69 )
```

```
##                 prediction
##                  Control Schizophrenia
##   Control             84            31
##   Schizophrenia       29            13
## The accuracy is 61.783 %.
## The True positive rate is 30.952 %
```

```r
#When we introduce data from the other study, the both model has a a low testing accuracy.
#This hint us that the two studys are different.
```

```r
#Combine data

#Further data cleaning to merge CNP and COBRE data
Study <- rep("CNP",nrow(CNP))

CNP <- data.frame(CNP,Study)
```

```r
CNP <- CNP %>%
  select(-c(7:41))

colnames(CNP)[5:6] <- c("Ethnicity","Education")

levels(CNP$Gender) <- c("Female","Male")

Study <- rep("COBRE",nrow(COBRE))
COBRE <- data.frame(COBRE,Study)

COBRE <- COBRE %>%
  select(-c(5,8:111))

# CNP Ethinicty
#1=Hispanic origin
#2=Not of Hispanic origin

#COBRE Ethinicty
#Caucasian = 1
#African-American   = 2
#Hispanic   = 3

#Recoding required

table(COBRE$Ethnicity)
```

```
##
##  1  2  3
## 69  9 53
```

```r
for(i in 1:length(COBRE$Ethnicity)){
  if(!is.na(COBRE$Ethnicity[i])){
    if(COBRE$Ethnicity[i] == 1 | COBRE$Ethnicity[i] == 2)
      COBRE$Ethnicity[i] <- 4
  }
}
COBRE$Ethnicity <- COBRE$Ethnicity - 2

table(COBRE$Ethnicity)
```

```
##
##  1  2
## 53 78
```

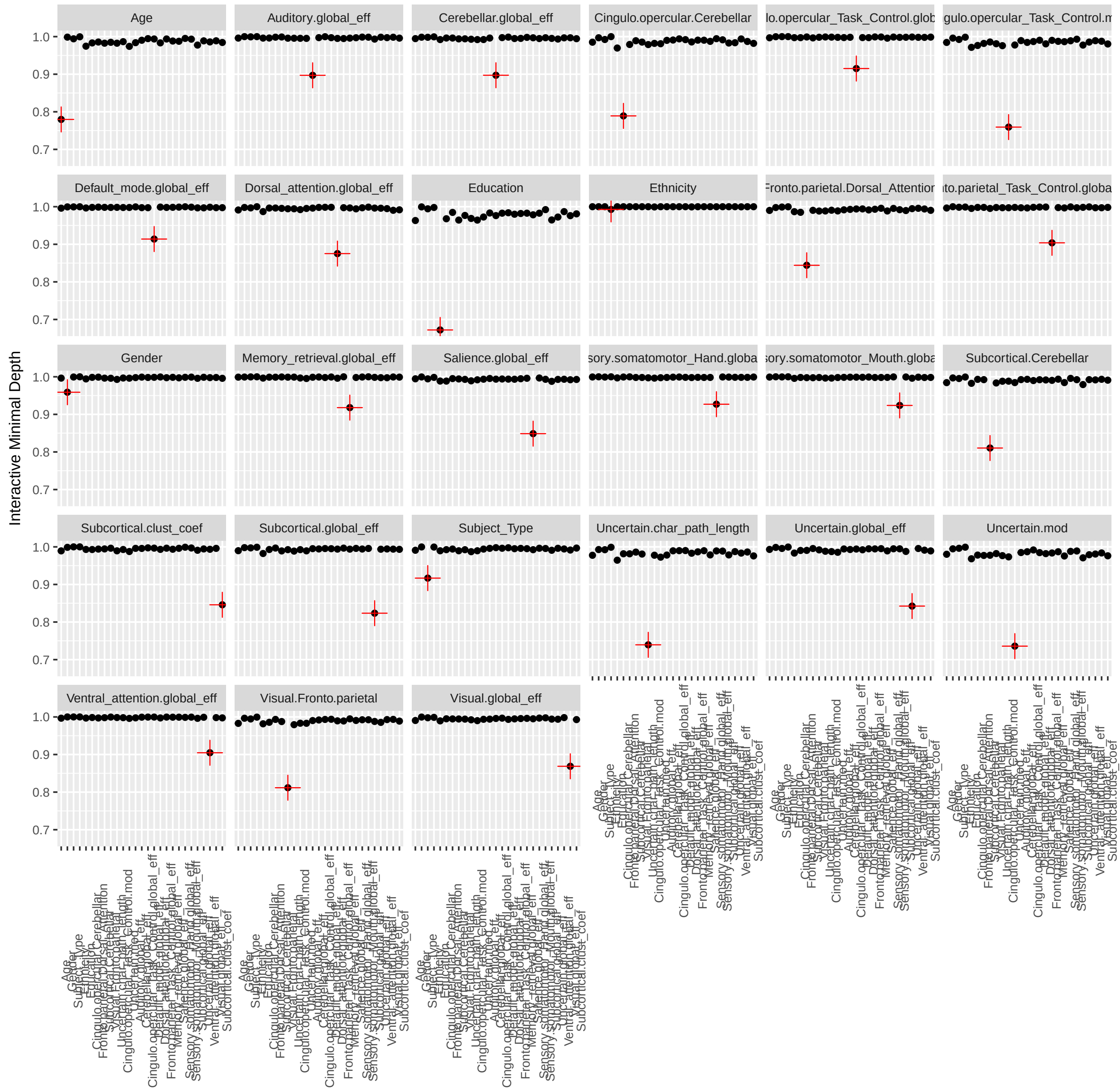
```r
Data <- merge(CNP,COBRE, all = TRUE) %>%
  select(-c(1))
Data$Ethnicity <- as.factor(Data$Ethnicity)
levels(Data$Ethnicity) <- c("Hispanic", "non-Hispanic")
```

```r
set.seed(4321)

# Combine Data modeling


#Random Forest variable selection
```

```r
rfsrc_m3 <- rfsrc(Study~.,data = Data, na.action = c("na.omit"), ntree= 1000)


max_var <- max.subtree(rfsrc_m3, conservative = TRUE)

max_var$topvars
```

```
##  [1] "Age"
##  [2] "Education"
##  [3] "Cingulo.opercular.Cerebellar"
##  [4] "Fronto.parietal.Dorsal_Attention"
##  [5] "Subcortical.Cerebellar"
##  [6] "Visual.Fronto.parietal"
##  [7] "Uncertain.char_path_length"
##  [8] "Cingulo.opercular_Task_Control.mod"
##  [9] "Uncertain.mod"
## [10] "Subcortical.global_eff"
## [11] "Uncertain.global_eff"
## [12] "Subcortical.clust_coef"
```

```r
#delete duplicate entity


subset3 <- as.vector(max_var$topvars)

subset3 <- delete_dup(subset3,Data[,c(1:5,139:152)])



#Logistic Regression model

Data_logi <- Data[,c("Study",names(Data[,c(1:5,139:152)]), subset3)]

Data_logi <- na.omit(Data_logi) %>%
  Standarize()


#find interaction
gg_int <- gg_interaction(find.interaction(rfsrc_m3,
                                          xvar.names = names(Data_logi[,-c(1)]),
                                          sorted = FALSE,
                                          verbose = FALSE))
plot(gg_int)
```




```
#No interactioin fund base on the result, we don't have to add interaction term

#check correlation

high_cor <- findCorrelation(cor(Data_logi[,-c(1,3:5)]),cutoff = 0.75) + 4

#Remove variables to prevent multicollinearity problem
Data_logi <- Data_logi %>%
  select(-c(high_cor))

index <- sample(1:nrow(Data_logi), size = round(nrow(Data_logi)*0.7,0),replace = FALSE)

Data_train <- Data_logi[index,]
Data_test <- Data_logi[-index,]
```

```r
logi_m3 <-glm(Study~. + Subject_Type*Age , data = Data_train, family = "binomial")
summary(logi_m3)
```

```
##
## Call:
## glm(formula = Study ~ . + Subject_Type * Age, family = "binomial",
##     data = Data_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7700  -0.6746  -0.3092   0.7643   2.4375
##
## Coefficients:
##                                          Estimate Std. Error z value
## (Intercept)                              -0.53169    0.42039  -1.265
## Age                                       0.36891    0.31853   1.158
## GenderMale                                0.46669    0.43638   1.069
## Subject_TypeSchizophrenia                 0.38913    0.47163   0.825
## Ethnicitynon-Hispanic                    -0.21121    0.44299  -0.477
## Education                                -0.48969    0.24136  -2.029
## Auditory.global_eff                       0.22337    0.23506   0.950
## Cerebellar.global_eff                    -0.29040    0.23386  -1.242
## Cingulo.opercular_Task_Control.global_eff  0.33690    0.26957   1.250
## Default_mode.global_eff                  -0.09649    0.28185  -0.342
## Dorsal_attention.global_eff               0.25471    0.24005   1.061
## Fronto.parietal_Task_Control.global_eff    0.01053    0.23382   0.045
## Memory_retrieval.global_eff              -0.11565    0.21575  -0.536
## Salience.global_eff                       0.33326    0.24352   1.368
## Sensory.somatomotor_Hand.global_eff       0.03733    0.30262   0.123
## Sensory.somatomotor_Mouth.global_eff     -0.12186    0.25212  -0.483
## Subcortical.global_eff                    0.35924    0.26237   1.369
## Uncertain.global_eff                     -0.33765    0.24043  -1.404
## Ventral_attention.global_eff             -0.11511    0.22425  -0.513
## Visual.global_eff                        -0.17071    0.26198  -0.652
## Cingulo.opercular.Cerebellar             -0.59461    0.28980  -2.052
## Subcortical.Cerebellar                   -0.31993    0.31572  -1.013
## Visual.Fronto.parietal                    0.15127    0.31430   0.481
## Cingulo.opercular_Task_Control.mod        0.40354    0.23812   1.695
## Uncertain.mod                             0.39313    0.24439   1.609
## Subcortical.clust_coef                    0.50978    0.23507   2.169
## Age:Subject_TypeSchizophrenia            -0.09416    0.42299  -0.223
##                                          Pr(>|z|)
## (Intercept)                                0.2060
## Age                                        0.2468
## GenderMale                                 0.2849
## Subject_TypeSchizophrenia                  0.4093
## Ethnicitynon-Hispanic                      0.6335
## Education                                  0.0425 *
## Auditory.global_eff                        0.3420
## Cerebellar.global_eff                      0.2143
## Cingulo.opercular_Task_Control.global_eff  0.2114
## Default_mode.global_eff                    0.7321
## Dorsal_attention.global_eff                0.2887
## Fronto.parietal_Task_Control.global_eff    0.9641
```

```
## Memory_retrieval.global_eff                  0.5919
## Salience.global_eff                          0.1712
## Sensory.somatomotor_Hand.global_eff          0.9018
## Sensory.somatomotor_Mouth.global_eff         0.6289
## Subcortical.global_eff                       0.1709
## Uncertain.global_eff                         0.1602
## Ventral_attention.global_eff                 0.6077
## Visual.global_eff                            0.5146
## Cingulo.opercular.Cerebellar                 0.0402 *
## Subcortical.Cerebellar                       0.3109
## Visual.Fronto.parietal                       0.6303
## Cingulo.opercular_Task_Control.mod           0.0901 .
## Uncertain.mod                                0.1077
## Subcortical.clust_coef                       0.0301 *
## Age:Subject_TypeSchizophrenia                0.8239
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 265.96  on 193  degrees of freedom
## Residual deviance: 178.00  on 167  degrees of freedom
## AIC: 232
##
## Number of Fisher Scoring iterations: 5
```

```
round(exp(coef(logi_m3)),3)
```

```
##                              (Intercept)
##                                    0.588
##                                      Age
##                                    1.446
##                               GenderMale
##                                    1.595
##                  Subject_TypeSchizophrenia
##                                    1.476
##                     Ethnicitynon-Hispanic
##                                    0.810
##                                Education
##                                    0.613
##                      Auditory.global_eff
##                                    1.250
##                    Cerebellar.global_eff
##                                    0.748
## Cingulo.opercular_Task_Control.global_eff
##                                    1.401
##                  Default_mode.global_eff
##                                    0.908
##             Dorsal_attention.global_eff
##                                    1.290
##   Fronto.parietal_Task_Control.global_eff
##                                    1.011
##             Memory_retrieval.global_eff
##                                    0.891
##                      Salience.global_eff
```

```
##                                    1.396
##        Sensory.somatomotor_Hand.global_eff
##                                    1.038
##       Sensory.somatomotor_Mouth.global_eff
##                                    0.885
##                     Subcortical.global_eff
##                                    1.432
##                       Uncertain.global_eff
##                                    0.713
##               Ventral_attention.global_eff
##                                    0.891
##                          Visual.global_eff
##                                    0.843
##               Cingulo.opercular.Cerebellar
##                                    0.552
##                     Subcortical.Cerebellar
##                                    0.726
##                     Visual.Fronto.parietal
##                                    1.163
##         Cingulo.opercular_Task_Control.mod
##                                    1.497
##                              Uncertain.mod
##                                    1.482
##                     Subcortical.clust_coef
##                                    1.665
##              Age:Subject_TypeSchizophrenia
##                                    0.910
```

```r
anova(logi_m3, test = "Chisq")
```

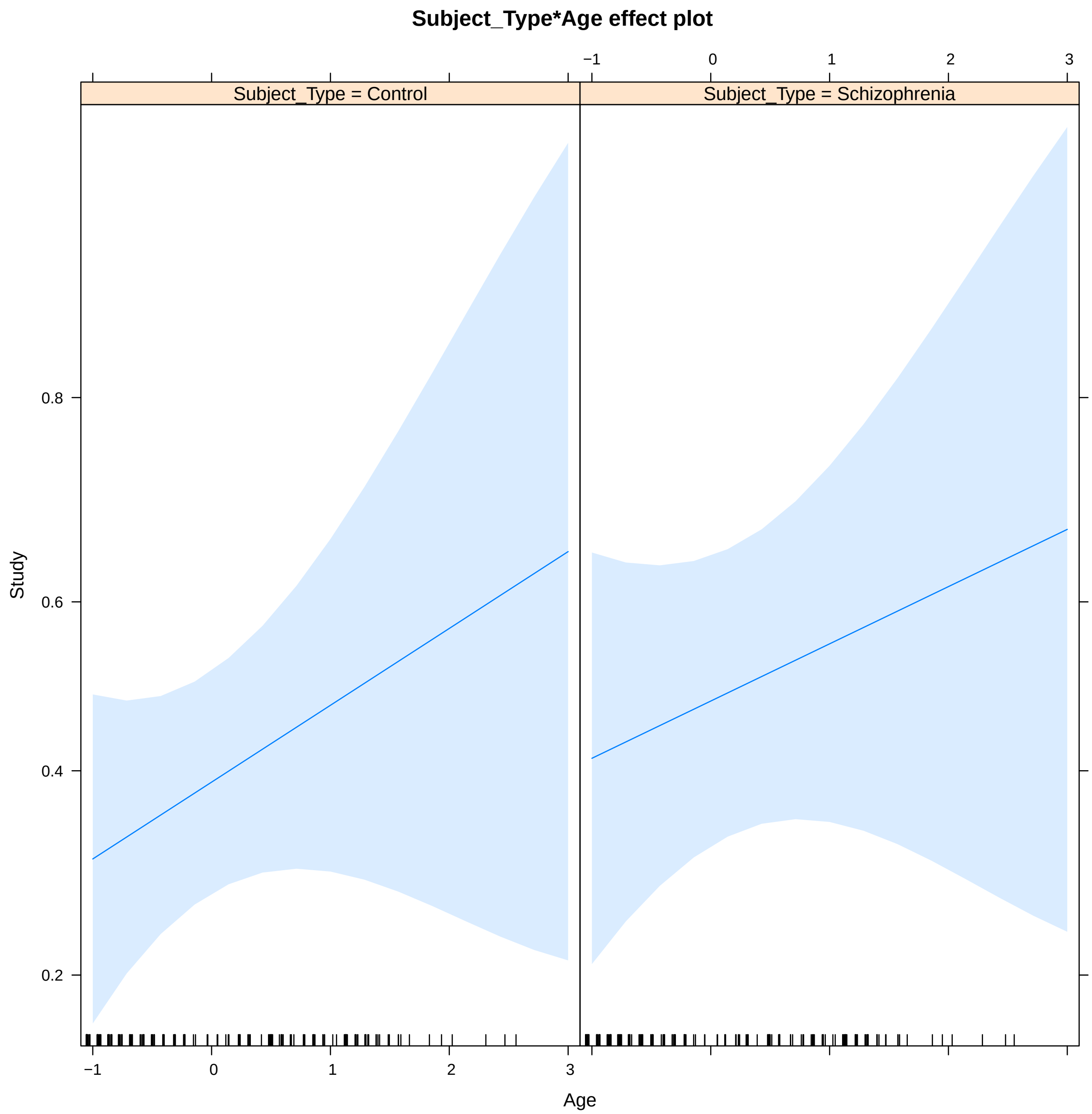
```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: Study
##
## Terms added sequentially (first to last)
##
##
##                                          Df Deviance Resid. Df Resid. Dev
## NULL                                                       193     265.96
## Age                                       1   3.3371       192     262.63
## Gender                                    1   6.2191       191     256.41
## Subject_Type                              1   7.2228       190     249.19
## Ethnicity                                 1   0.5869       189     248.60
## Education                                 1  12.7223       188     235.88
## Auditory.global_eff                       1   3.6053       187     232.27
## Cerebellar.global_eff                     1   6.4624       186     225.81
## Cingulo.opercular_Task_Control.global_eff 1   0.2843       185     225.52
## Default_mode.global_eff                   1   3.6685       184     221.86
## Dorsal_attention.global_eff               1   1.5829       183     220.27
## Fronto.parietal_Task_Control.global_eff   1   0.0726       182     220.20
## Memory_retrieval.global_eff               1   0.0668       181     220.13
## Salience.global_eff                       1   3.7711       180     216.36
## Sensory.somatomotor_Hand.global_eff       1   0.0170       179     216.34
```

```
## Sensory.somatomotor_Mouth.global_eff       1   0.4466       178      215.90
## Subcortical.global_eff                     1   3.5802       177      212.32
## Uncertain.global_eff                       1   5.7028       176      206.62
## Ventral_attention.global_eff               1   0.4207       175      206.19
## Visual.global_eff                          1   2.0756       174      204.12
## Cingulo.opercular.Cerebellar               1  14.1765       173      189.94
## Subcortical.Cerebellar                     1   1.8643       172      188.08
## Visual.Fronto.parietal                     1   0.0379       171      188.04
## Cingulo.opercular_Task_Control.mod         1   2.3040       170      185.74
## Uncertain.mod                              1   2.3636       169      183.37
## Subcortical.clust_coef                     1   5.3243       168      178.05
## Age:Subject_Type                           1   0.0495       167      178.00
##                                            Pr(>Chi)
## NULL
## Age                                       0.0677343 .
## Gender                                    0.0126379 *
## Subject_Type                              0.0071985 **
## Ethnicity                                 0.4436360
## Education                                 0.0003613 ***
## Auditory.global_eff                       0.0575963 .
## Cerebellar.global_eff                     0.0110179 *
## Cingulo.opercular_Task_Control.global_eff 0.5938826
## Default_mode.global_eff                   0.0554491 .
## Dorsal_attention.global_eff               0.2083497
## Fronto.parietal_Task_Control.global_eff   0.7875658
## Memory_retrieval.global_eff               0.7959919
## Salience.global_eff                       0.0521457 .
## Sensory.somatomotor_Hand.global_eff       0.8961752
## Sensory.somatomotor_Mouth.global_eff      0.5039533
## Subcortical.global_eff                    0.0584732 .
## Uncertain.global_eff                      0.0169375 *
## Ventral_attention.global_eff              0.5165702
## Visual.global_eff                         0.1496697
## Cingulo.opercular.Cerebellar              0.0001664 ***
## Subcortical.Cerebellar                    0.1721237
## Visual.Fronto.parietal                    0.8457015
## Cingulo.opercular_Task_Control.mod        0.1290434
## Uncertain.mod                             0.1241928
## Subcortical.clust_coef                    0.0210304 *
## Age:Subject_Type                          0.8239025
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
#R-squared

R_squared <- 1 - (summary(logi_m3)[[4]]/summary(logi_m3)[[8]])
R_squared
```

```
## [1] 0.3307403
```

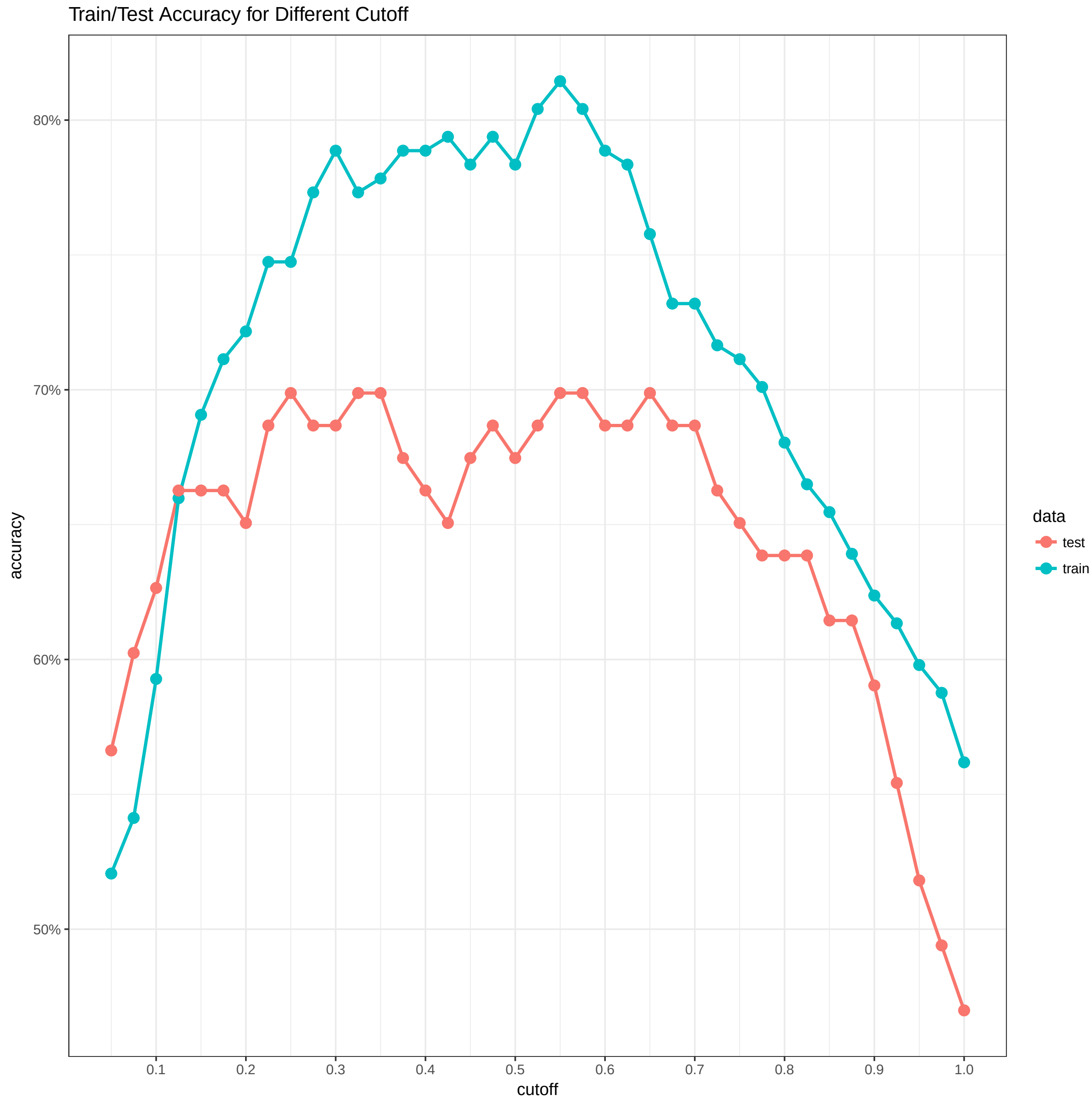
```r
#Effect plot
plot(Effect(c("Subject_Type", "Age"), logi_m3),ask = FALSE)
```




```r
#70/30 CV check

#Train
Data_train$prediction <- predict(logi_m3, Data_train, type = "response")


#Test
Data_test$prediction <- predict(logi_m3, Data_test, type = "response")

prop.table(table(Data$Study))
```

```
##
##       CNP     COBRE
## 0.5250836 0.4749164
```

```r
accuracy_info <- AccuracyCutoffInfo( train = Data_train, test = Data_test,
                                     predict = "prediction", actual = "Study" )

accuracy_info$plot
```




```r
Classify(Data_train, Data_train$prediction,"Study", 0.55)
```

```
##         prediction
##           CNP COBRE
##    CNP     95    14
##    COBRE   22    63
## The accuracy is 81.443 %.
## The True positive rate is 74.118 %
```

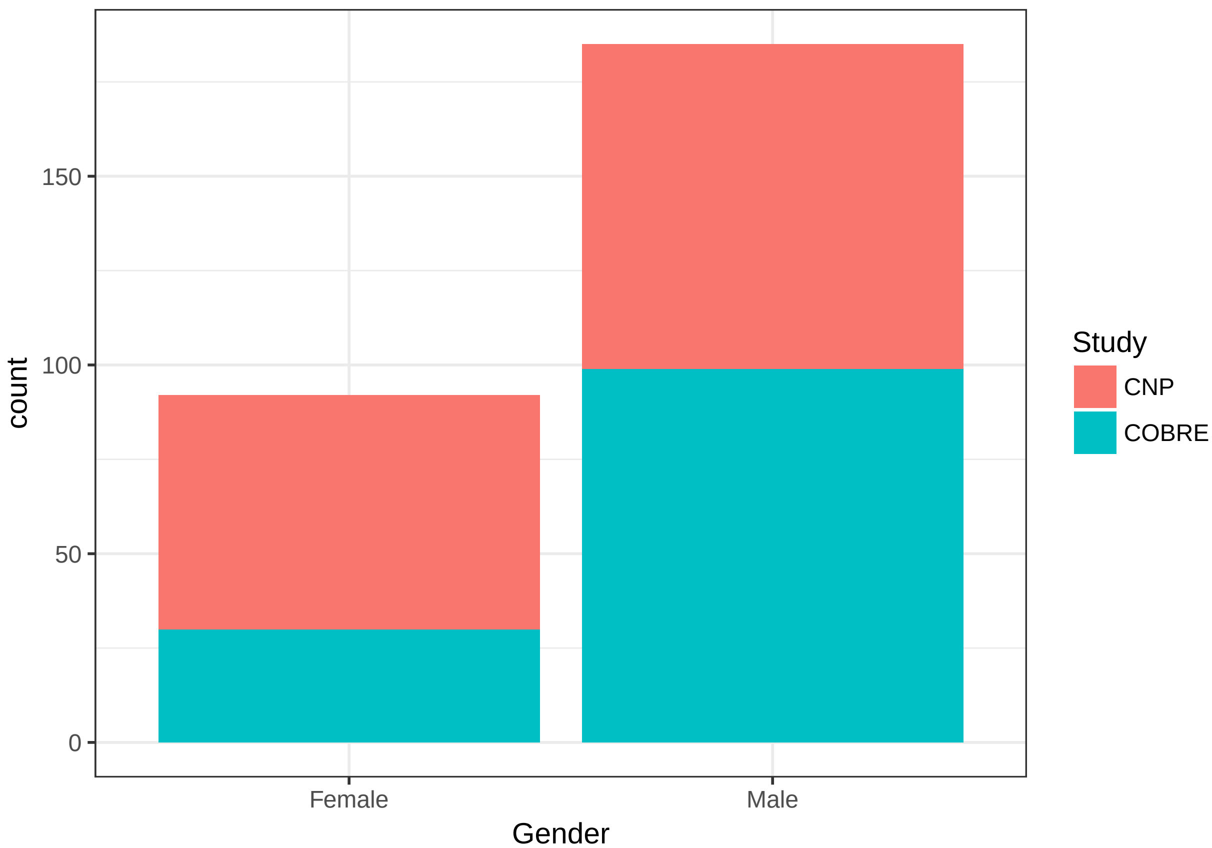
```r
Classify(Data_test, Data_test$prediction,"Study", 0.55)
```

```
##         prediction
##          CNP COBRE
##   CNP     30     9
##   COBRE  16     28
## The accuracy is 69.88 %.
## The True positive rate is 63.636 %
```

```r
#Combine data model k fold CV check
#Here we are not interesting in looking at the True positive rate
set.seed(4321)

Accuracy.k <- cv.error(Data_logi, "Study", cut_off = 0.55)[[1]]
mean(Accuracy.k)
```

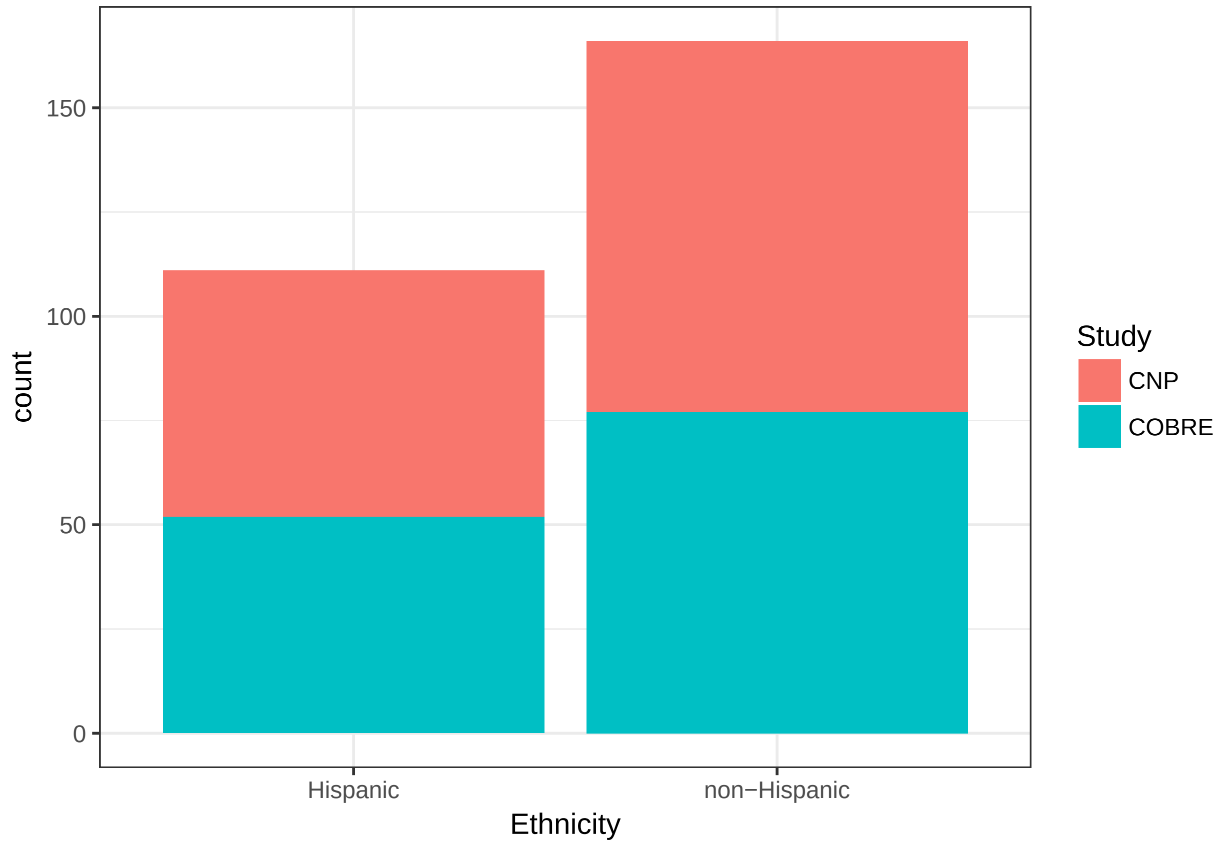
```
## [1] 0.7439153
```

```r
par(mfrow = c(2,2))


ggplot(data = na.omit(Data), aes(x = Gender, fill = Study)) +
  geom_bar() +
  theme_bw()
```

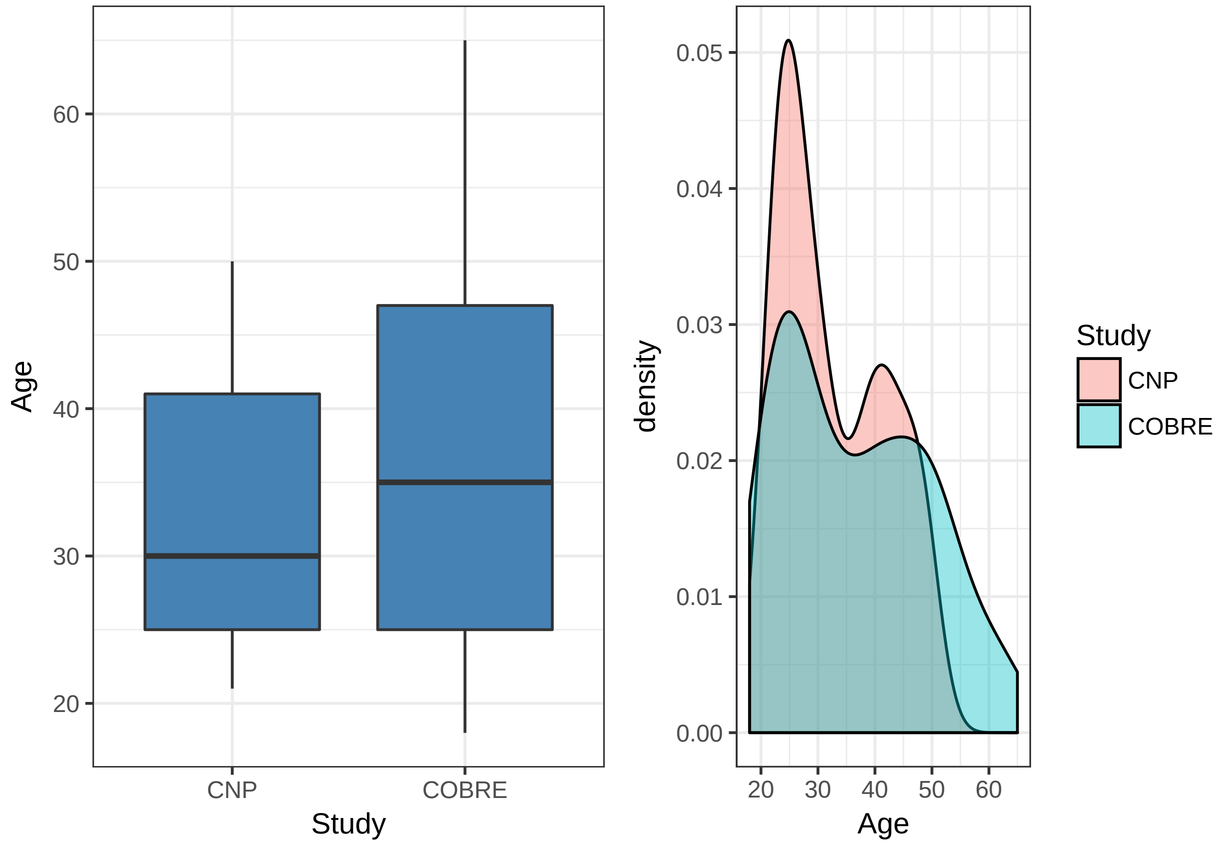



```r
ggplot(data = na.omit(Data), aes(x = Ethnicity, fill = Study)) +
  geom_bar() +
```

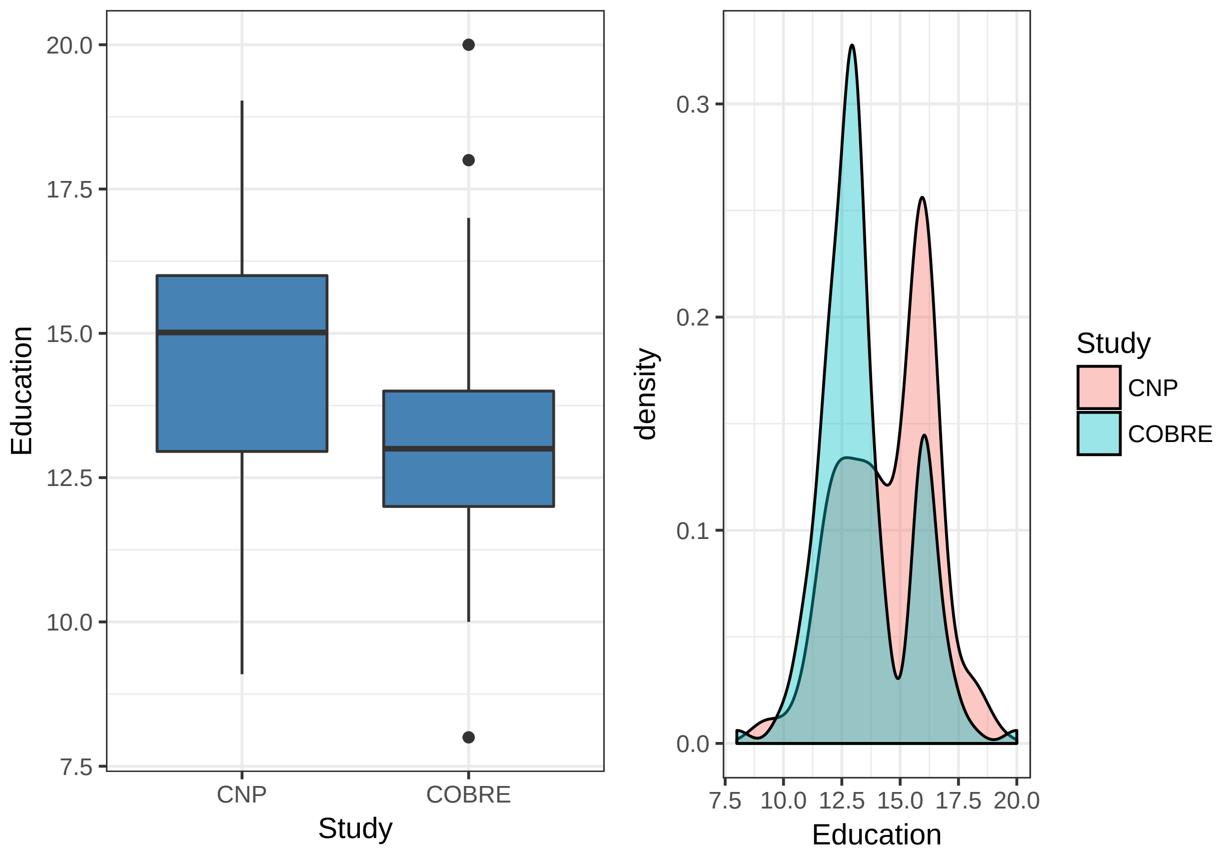
```
  theme_bw()
```




```
plot1 <- ggplot(data = na.omit(Data), aes(x = Study, y = Age)) +
  geom_boxplot(fill = "steelblue") +
  theme_bw()

plot2 <- ggplot(data = na.omit(Data), aes(x = Age, fill = Study)) +
  geom_density(alpha = 0.4) +
  theme_bw()

grid.arrange(plot1,plot2, nrow = 1, ncol = 2)
```




```
plot3 <- ggplot(data = na.omit(Data), aes(x = Study, y = Education)) +
  geom_boxplot(fill = "steelblue") +
  theme_bw()

plot4 <- ggplot(data = na.omit(Data), aes(x = Education, fill = Study)) +
  geom_density(alpha = 0.4) +
  theme_bw()

grid.arrange(plot3,plot4, nrow = 1, ncol = 2)
```




```r
#Recall the anova output for the combined data set logistic model

anova(logi_m3, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: Study
##
## Terms added sequentially (first to last)
##
##
##                                          Df Deviance Resid. Df Resid. Dev
## NULL                                                       193     265.96
## Age                                       1   3.3371       192     262.63
## Gender                                    1   6.2191       191     256.41
## Subject_Type                              1   7.2228       190     249.19
## Ethnicity                                 1   0.5869       189     248.60
## Education                                 1  12.7223       188     235.88
## Auditory.global_eff                       1   3.6053       187     232.27
## Cerebellar.global_eff                     1   6.4624       186     225.81
## Cingulo.opercular_Task_Control.global_eff 1   0.2843       185     225.52
## Default_mode.global_eff                   1   3.6685       184     221.86
## Dorsal_attention.global_eff               1   1.5829       183     220.27
## Fronto.parietal_Task_Control.global_eff   1   0.0726       182     220.20
```

```
## Memory_retrieval.global_eff               1   0.0668       181      220.13
## Salience.global_eff                       1   3.7711       180      216.36
## Sensory.somatomotor_Hand.global_eff       1   0.0170       179      216.34
## Sensory.somatomotor_Mouth.global_eff      1   0.4466       178      215.90
## Subcortical.global_eff                    1   3.5802       177      212.32
## Uncertain.global_eff                      1   5.7028       176      206.62
## Ventral_attention.global_eff              1   0.4207       175      206.19
## Visual.global_eff                         1   2.0756       174      204.12
## Cingulo.opercular.Cerebellar              1  14.1765       173      189.94
## Subcortical.Cerebellar                    1   1.8643       172      188.08
## Visual.Fronto.parietal                    1   0.0379       171      188.04
## Cingulo.opercular_Task_Control.mod        1   2.3040       170      185.74
## Uncertain.mod                             1   2.3636       169      183.37
## Subcortical.clust_coef                    1   5.3243       168      178.05
## Age:Subject_Type                          1   0.0495       167      178.00
##                                           Pr(>Chi)
## NULL
## Age                                      0.0677343 .
## Gender                                   0.0126379 *
## Subject_Type                             0.0071985 **
## Ethnicity                                0.4436360
## Education                                0.0003613 ***
## Auditory.global_eff                      0.0575963 .
## Cerebellar.global_eff                    0.0110179 *
## Cingulo.opercular_Task_Control.global_eff 0.5938826
## Default_mode.global_eff                  0.0554491 .
## Dorsal_attention.global_eff              0.2083497
## Fronto.parietal_Task_Control.global_eff   0.7875658
## Memory_retrieval.global_eff              0.7959919
## Salience.global_eff                      0.0521457 .
## Sensory.somatomotor_Hand.global_eff      0.8961752
## Sensory.somatomotor_Mouth.global_eff     0.5039533
## Subcortical.global_eff                   0.0584732 .
## Uncertain.global_eff                     0.0169375 *
## Ventral_attention.global_eff             0.5165702
## Visual.global_eff                        0.1496697
## Cingulo.opercular.Cerebellar             0.0001664 ***
## Subcortical.Cerebellar                   0.1721237
## Visual.Fronto.parietal                   0.8457015
## Cingulo.opercular_Task_Control.mod       0.1290434
## Uncertain.mod                            0.1241928
## Subcortical.clust_coef                   0.0210304 *
## Age:Subject_Type                         0.8239025
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#Hypothesis testing for demographic variables in the combined data set


t.test(Age~Study, data = Data_logi)
```

```
##
##  Welch Two Sample t-test
##
## data:  Age by Study
```

```
## t = -2.8159, df = 227.22, p-value = 0.005292
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.5820745 -0.1028142
## sample estimates:
##   mean in group CNP mean in group COBRE
##          -0.1594777           0.1829667
```

```r
t.test(Education~Study, data = Data_logi)
```

```
##
##  Welch Two Sample t-test
##
## data:  Education by Study
## t = 4.5959, df = 272.59, p-value = 6.597e-06
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.3046507 0.7612365
## sample estimates:
##   mean in group CNP mean in group COBRE
##           0.2481939          -0.2847496
```

```r
#Pearson's chi-squared test

#H_{0} = there is no difference between the distributions
#H_{1} = there is a difference between the distributions

chisq.test(table(Data_logi$Study, Data_logi$Gender))
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(Data_logi$Study, Data_logi$Gender)
## X-squared = 9.9677, df = 1, p-value = 0.001593
```

```r
chisq.test(table(Data_logi$Study, Data_logi$Ethnicity))
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(Data_logi$Study, Data_logi$Ethnicity)
## X-squared = 1.2223e-30, df = 1, p-value = 1
```

```r
chisq.test(table(Data_logi$Study, Data_logi$Subject_Type))
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(Data_logi$Study, Data_logi$Subject_Type)
## X-squared = 16.988, df = 1, p-value = 3.762e-05
```