(https://lms.monash.edu/mod/assign/view.php?id=7560449)

# FIT5197 Assignment 3 Semester 2, 2020 (https://lms.monash.edu/mod/assign/view.php?id=7560449)

Authors: Dan Nguyen, Yun Zhao

Admins (Competition): Dr. Levin Kuhlmann, Yun Zhao, Anil Gurbuz

Proofreaders: Dr. Levin Kuhlmann, Yun Zhao, and other tutors

Date: Oct 2020

(https://lms.monash.edu/mod/assign/view.php?id=7560449)

# Assignment Instruction (https://lms.monash.edu/mod/assign/view.php?id=7560449)

Please read through the instructions carefully, by submitting the assignment, you are considered to have read all the instructions carefully and be aware of the penalties that entail.

# Part 1: Regression (50 Marks)

This part is about regression. Specifically, you will be `predicting the fuel efficiency` of a car (in kilometers per litre) based on its characteristics. This is a practical problem as Australia is one of the largest automobile markets in the world; thus, correctly predicting the fuel efficiency is necessary to control emission rates to the environment.

The dataset has many observations and predictors obtained from many retailers for car models available for sale from 2017 to 2020. The target variable is the fuel efficiency of the car measured in kilometers per litre. The higher this value, the better the fuel efficiency of the car.

PleaseProvide working/R code/justifications for each of these questions as required.

**Note:** If not explicitly mentioned, libraries are not allowed

In [1]:

```
# Read the data from students' side
remove(list = ls())
train <- read.csv("RegressionTrain.csv")
test <- read.csv("RegressionTest.csv")
```

In [ ]:

```
# PLEASE DO NOT ALTER THIS CODE BLOCK
# Please skip (don't run) this if you are a student
# Read in the data from marking tutors' side (ensure no cheating!)
remove(list = ls())
train <- read.csv("../data/RegressionTrain.csv")
test <- read.csv("../data/RegressionTest.csv")
label <- read.csv("../data/RegressionTestLabel.csv")
```

# Question 1 (5 Marks)

Fit a **multiple linear model** to the fuel efficiency data using the `train` dataset. By checking the summary information, which predictors/variables do you think are possibly associated with fuel efficiency (use `0.05` significant level), and why? Which `three predictors/variables` appear to be the strongest predictors of fuel efficiency, and why?

**Note**: You don't have to worry about categorical variables here since R can deal with this automatically, focus your efforts on interpretation. Additionally, when explaining why features are strongly associated with the target, please refrain giving one or two sentences answers, these answers are not descriptive enough and will result in deduction of marks. Finally, please name the model here `lm.fit` for future marking purposes.

**YOUR ANSWER HERE**

In [3]:

```
# Change this
lm.fit <-lm(Comb.FE ~ .,data = train)
summary(lm.fit)
```

```
Call:
lm(formula = Comb.FE ~ ., data = train)

Residuals:
    Min      1Q  Median      3Q     Max 
-4.0256 -0.9978 -0.0644  0.7006 11.3941 

Coefficients:
                           Estimate Std. Error t value Pr(>|t|)    
(Intercept)              -1.783e+02  8.587e+01  -2.076  0.03809 *  
Model.Year                9.640e-02  4.255e-02   2.266  0.02363 *  
Eng.Displacement         -1.364e+00  1.025e-01 -13.306  < 2e-16 ***
No.Cylinders              4.644e-02  6.769e-02   0.686  0.49282    
AspirationOT             -3.452e-01  6.352e-01  -0.543  0.58693    
AspirationSC             -9.197e-01  2.282e-01  -4.031 5.85e-05 ***
AspirationTC             -1.303e+00  1.288e-01 -10.111  < 2e-16 ***
AspirationTS             -1.149e+00  4.945e-01  -2.323  0.02035 *  
No.Gears                 -1.307e-01  2.995e-02  -4.364 1.37e-05 ***
Lockup.Torque.ConverterY -8.243e-01  1.117e-01  -7.377 2.78e-13 ***
Drive.SysA               -8.339e-02  1.521e-01  -0.548  0.58356    
Drive.SysF                1.441e+00  1.711e-01   8.419  < 2e-16 ***
Drive.SysP               -2.400e-01  2.980e-01  -0.805  0.42087    
Drive.SysR                4.328e-02  1.476e-01   0.293  0.76938    
Max.Ethanol              -7.076e-03  2.967e-03  -2.385  0.01722 *  
Fuel.TypeGM               5.706e-01  4.173e-01   1.368  0.17169    
Fuel.TypeGP               4.093e-01  1.369e-01   2.990  0.00284 ** 
Fuel.TypeGPR              1.363e-01  1.401e-01   0.973  0.33096    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.598 on 1382 degrees of freedom
Multiple R-squared:  0.6628,    Adjusted R-squared:  0.6586 
F-statistic: 159.8 on 17 and 1382 DF,  p-value: < 2.2e-16
```

Our goal is to make predictions, $\hat{y}_i$, to model a linear model using the train data set. We can get this using the lm function in R and using summary we can get the various stats like P-value, Residual standard error,Multiple R-squared etc. So looking at the summary we have a very important information i.e. P-value.

Thus, we know that p-value is a good indicator for determining which variable has the highest relation with the target varible and if the `p-value is less than 0.05` it can be said that the variable is possibly associated with the target class As we can see above there are 10 varibles that have p-value lower than `0.05` and these 10 variables will impact our target variable. These variables are:

- Model.Year with p-value as `0.02363`
- Eng.Displacement with p-value as `< 2e-16`
- AspirationSC with p-value as `5.85e-05`
- AspirationTC with p-value as `< 2e-16`
- AspirationTS with p-value as `0.02035`
- No.Gears with p-value as `1.37e-05`
- Lockup.Torque.ConverterY with p-value as `2.78e-13`

- Drive.SysF with p-value as `< 2e-16`
- Max.Ethanol with p-value as `0.01722`
- Fuel.TypeGP with p-value as `0.00284`

As we look at the model summary we can clearly see that **`Eng.Displacement`** , **`AspirationTC`** , **`Drive.SysF`** , has the lowest p-value in comparison to the other variables with a `p-value < 2e-16` Thus we can say that these three features are the strongest pridictors for `fuel efficiency(Comb.FE)` .

# Question 2 (5 Marks)

Describe/discuss the effect that the year of manufacture `(Model.Year)` variable appears to have on the mean `fuel efficiency` . Additionally, describe/discuss the effect that the number of gears `(No.Gears)` variable has on the mean `fuel efficiency` of the car.

**Note**: This asks for your descriptions, please refrain from using one or two lines to describe/discuss the effect. Keep answers to be 4 decimal places

### YOUR ANSWER HERE

After analysing the summary of model in question 1. We first look at the p-value of the variable to find out the importance of the varible in finding the target class. Also we know that if a variable has a p-value less than 0.05 then we know that the variable is of importance in finding the target variable. So we will check the the p-value for `Model.Year & No.Gears`

#### 1.) Model.Year

First we will inspect the `Model.Year` as we can see in the above summary that the `p-value: 0.02363` which is less that 0.05 which means that Model.Year is an important factor in determining the `Combe.FE(Fuel efficiency)` thus we can say that there is impact of Model.Year on Comb.FE. Also looking at the `Estimate` in our summary we can say that it refer to it as $\beta_1$. where $\beta_1$ is a regression coeffcient, i.e it is the amount the predicted value $\hat{y}_i$ changes with one unit change of the predictor Model.Year. So as we look at the Model.Year $\beta_1$ it is `9.640e-02` . Which means that For each year of manufacture the mean fuel efficiency of a car increases by `0.0964km/l` as the estimate value correspoding to it is positive that is why the fuel efficiency will increase.

#### 2.) No.Gears

Similarly for `No.Gears` as we check the p-value corresponding to it the `p-value: 1.37e-05` which is also less than 0.05 significance level. So we can say that No.Gears is also one of the important predictor in finding the target variable `Comb.FE` . Now looking at the $\beta_i$ for `No.Gears` where $\beta_i$ is the regression coeffcient for `No.Gears` we can see the correspoding estimate value is `-1.307e-01` . So this means that for each additional gear a car has the mean fuel efficiency decreases by `0.1307km/l` . This is because there is an inverse relation between the predictor and the target variable. and as number of gears increase the mean fuel efficiency will fall.

# Question 3 (5 Marks)

Apply the stepwise selection procedure with the **BIC** penalty to prune out potentially less significant variables. Write down the final regression equation obtained after pruning, please keep the values of the parameter coefficients to 2 decimal places. Finally, also describe the pruned model.

**Note**: please don't change the default direction `both` in the step function, this is so that we can check your work easily. Additionally, please name this model `sw.fit`

## YOUR ANSWER HERE

In [4]:

```
# Change this
sw.fit <- step(lm.fit, k = log(nrow(train)), trace = 0, direction = 'both')
summary(sw.fit)
```

```
Call:
lm(formula = Comb.FE ~ Eng.Displacement + Aspiration + No.Gears +
    Lockup.Torque.Converter + Drive.Sys + Max.Ethanol, data = train)

Residuals:
    Min      1Q  Median      3Q     Max
-4.0743 -0.9760 -0.0349  0.6566 11.3971

Coefficients:
                          Estimate Std. Error t value Pr(>|t|)
(Intercept)              16.196874   0.282901  57.253  < 2e-16 ***
Eng.Displacement         -1.277173   0.043418 -29.416  < 2e-16 ***
AspirationOT             -0.100081   0.626276  -0.160 0.873060
AspirationSC             -0.699137   0.213768  -3.271 0.001100 **
AspirationTC             -1.144227   0.107302 -10.664  < 2e-16 ***
AspirationTS             -1.122104   0.481471  -2.331 0.019919 *
No.Gears                 -0.113537   0.029183  -3.891 0.000105 ***
Lockup.Torque.ConverterY -0.825285   0.110202  -7.489 1.23e-13 ***
Drive.SysA                0.035013   0.145617   0.240 0.810020
Drive.SysF                1.480191   0.166847   8.872  < 2e-16 ***
Drive.SysP               -0.323201   0.292617  -1.105 0.269560
Drive.SysR                0.093779   0.146329   0.641 0.521710
Max.Ethanol              -0.008344   0.002934  -2.843 0.004528 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.605 on 1387 degrees of freedom
Multiple R-squared:  0.6586,    Adjusted R-squared:  0.6557
F-statistic:   223 on 12 and 1387 DF,  p-value: < 2.2e-16
```

In [5]:

```
coeff <- round(sw.fit$coefficients,digits=2)
coeff
```

**(Intercept)**
16.2
**Eng.Displacement**
-1.28
**AspirationOT**
-0.1
**AspirationSC**
-0.7
**AspirationTC**
-1.14
**AspirationTS**
-1.12
**No.Gears**
-0.11
**Lockup.Torque.ConverterY**
-0.83
**Drive.SysA**
0.04
**Drive.SysF**
1.48
**Drive.SysP**
-0.32
**Drive.SysR**
0.09
**Max.Ethanol**
-0.01

## Prune models:

Looking at the final pruned model `sw.fit` and on comparision with our main model `lm.fit` we can notice that out of 17 variable that were significant to our model we have only 12 variable that are left as significant that influences the target class variable these are: **Eng.Displacement, AspirationOT, AspirationSC, AspirationTC, AspirationTS, No.Gears, Lockup.Torque.ConverterY, Drive.SysA, Drive.SysF, Drive.SysP, Drive.SysR, Max.Ethanol**

As we drill down further in our Pruned model we can see that out of these 12 variable five of these variable have p-value less than 0.001 which are . `Eng.Displacement, AspirationTC,No.Gears, Lockup.Torque.ConverterY, Drive.SysF` This means these variable have the hieght impact on our target variable. So the final Linear equation will be as follows.

## Final Linear Equation:

Comb.FE= Eng.Displacement + Aspiration + No.Gears + Lockup.Torque.Converter + Drive.Sys + Max.Ethanol

**Comb.FE**=(16.2 × Intercept) + (-1.28 × Eng.Displacement) + (-1.14 × AspirationTC) + (-0.11 × No.Gears) + (-0.83 × Lockup.Torque.ConverterY) + (1.48 × Drive.SysF)

# Question 4 (5 Marks)

Say we are going to buy a new car and we want to improve the fuel efficiency of our new car, what does this `BIC model` suggest we should do? Provide a detailed answers of at least `150 words` .

### YOUR ANSWER HERE

Looking at the above `BIC model` that we have created in question-3 we can see that there are mainly `8` variables that have high effect on the models these variables are as follow: **Eng.Displacement, AspirationSC, AspirationTC, AspirationTS, No.Gears, Lockup.Torque.ConverterY, Drive.SysF, Max.Ethanol**. This is because all of these varaible have a p-value less than 0.05.

Now as we look at the intercept for these 8 variables which will impact the fuel efficiency the most we can say that in what manner the variable effects the target variable i.e. if positive intercept will increase the fuel efficiency on the target class and negative intercept will leave us to reduced fuel efficiency.

Thus we have

1.) **Variables with positive coefficients:** `Drive.SysF` with an intercept value of **`1.480191`**

2.) **Variables with negative coefficients:** Eng.Displacement, AspirationSC, AspirationTC, AspirationTS, No.Gears, Lockup.Torque.ConverterY, Max.Ethanol

`Eng.Displacement, AspirationTC, Drive.SysF` has the lowest P-value thus they have the highest impact on fuel efficiency i.e. the target variable it is suggested that if we buy a new car then we must focus on these features and in order to increase fuel efficiency we should focus on `reducing` the Eng.Displacement, AspirationTC and have the `higher` Drive.SysF.

# Question 5 (5 Marks)

Imagine that you are looking for a new car to buy to replace your existing car. Use the **test** dataset to inspect the first car fuel efficiency and see whether it is a good fit for you or not.

```
(a) Use your BIC model to predict the mean fuel efficiency for this new car. Provi
de a 95% confidence interval for this prediction. [2 mark]
(b) Following the previous estimation, given that the current car that you own has
a mean fuel efficiency of 9.5 km/l (measured over the life time of your ownershi
p), does your model (BIC) suggest that the new car will have better fuel efficienc
y than your current car? Why? [3 marks]
```

In [6]:

```
data<-test[1,]
mean.efficiency <- predict(sw.fit, data, interval = 'confidence')
mean.efficiency
#mean(mean.efficiency[,'fit'])
#mean(mean.efficiency[,'lwr'])
#mean(mean.efficiency[,'upr'])
fit <- (mean.efficiency[,"fit"])
cat("Predicted mean fue efficiency of the new car:", fit)
```

| fit | lwr | upr |
|---|---|---|
| 9.287257 | 9.052956 | 9.521557 |

```
Predicted mean fue efficiency of the new car: 9.287257
```

### YOUR ANSWER HERE

Solution A.) The 95% confidence interval is (9.05295, 9.5215)km/l

Solution B.) The interval containing 9.5 km/l. Thus we cannot say that the new car will have better fuel efficiency compared to the current one. But there are higher changes that the new car will have a lesser fuel economy as compared to the current car. This is because would confidence interval i.e 95% have a boundary of (9.05295, 9.5215)km/l.

## Question 6 (Libraries are allowed) (25 Marks)

As a Data Scientist, one of the key tasks is to build models **most appropriate/closest** to the truth; thus, modelling will not be limitted to these steps in the assignment. To simulate for a realistic modelling process, this question will be in the form of a competition among students to find out who has the best model.

Thus, You will be graded by the performance of your model compared to your classmates', the better your model, the higher your score. Additionally, you need to write a short paragraph describing/documenting your thought process in this model building process `(300 words)` . Note that this is to explain to us why you build your current model so that we can verify that you understand the model you build and not just copy from other people.

**Note** Please make sure that we can install the libraries that you use in this part, the code structure can be:

```
install.packages("some package", repos='http://cran.us.r-project.org')

library("some package")
```

Remember that if we cannot run your code, we will have to give you 0 marks, our suggestion is for you to use the standard `R version 3.6.1`

You also need to name your final model `fin.mod` so we can run a check to find out your performance. A good test for your understanding would be to set the previous **BIC model** to be the final model to check if your code works Appropriately.

20 Marks for the model performance in the competition

5 Marks for logically writing down the thought process in building the final model

This is the [link (https://www.kaggle.com/t/0a3c0fc91b074816a6315bb4e9b42602)](https://www.kaggle.com/t/0a3c0fc91b074816a6315bb4e9b42602) to the competition

## YOUR ANSWER HERE

In [7]:

```
install.packages("randomForest", repos='http://cran.us.r-project.org')
library(randomForest)
```

```
Installing package into 'C:/Users/Shranja Sharma/Documents/R/win-library/3.
6'
(as 'lib' is unspecified)

package 'randomForest' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\Shranja Sharma\AppData\Local\Temp\RtmpmgibrU\downloaded_pac
kages

Warning message:
"package 'randomForest' was built under R version 3.6.3"randomForest 4.6-14
Type rfNews() to see new features/changes/bug fixes.
```

In [12]:

```
# Use this function to check the performance of your model
rmse <- function(pred.label, truth.label){
    # Lower is better
    return(sqrt(mean((pred.label - truth.label)^2)))
}
```

In [13]:

```
# Build your final model here, use additional coding block if you need

fin.mod <- randomForest(Comb.FE ~ . ,train[-1],ntree=100,mtry=7,importance=TRUE)
levels(test$Aspiration) <- levels(train$Aspiration)
levels(test$Lockup.Torque.Converter) <- levels(train$Lockup.Torque.Converter)
levels(test$Drive.Sys) <- levels(train$Drive.Sys)
truth.label<-train$Comb.FE
# An example would be use the previous model as your final one (make sure to delete this in
#fin.mod <- sw.fit
```

In [14]:

```
# If you are using any packages that perform the prediction differently, please change the
pred.label <- predict(fin.mod, test)
```

In [15]:

```
# PLEASE DO NOT ALTER THIS CODE BLOCK
# put this label in a csv file to commit to the Leaderboard
write.csv(data.frame("RowIndex" = seq(1, length(pred.label)), "Prediction" = pred.label),
          "RegressionPredictLabel.csv", row.names = F)
```

In [ ]:

```
## PLEASE DO NOT ALTER THIS CODE BLOCK
## Please skip (don't run) this if you are a student
## For teaching team use only
RMSE.fin <- rmse(pred.label, label$Label)
cat(paste("RMSE is", RMSE.fin))
```

**ANSWER:**

Since the data deal with categorical data we have used Random forest classifier to predict our test data as in regression random forces works by genralising the mean prediction of each individual tree.

Since random forest aggregated prediction of all the decision trees and different depth so it trains the model on small pecieces of our training data

number of trees are set as `100` for our model to reduce over fitting of model in fin.mod variable mtry is 7 because it splits the data for 7 variables at each tree considering all the variables combinations

Further we have set the levels in testing dataset same as in our training dataset so that all the levels are available while predicting the target class in testing dsata set these variables are as follow `Aspiration, Lockup.Torque.Converter, Drive.Sys`

So Random forest gives us advantage in predicting our target class from our train data we have not considered model year as if is an insignificant varaible to our predictor class and set number of trees as 100 to avoid over fitting the model.

So concluding my model random forest has given me an optimum RMSE as `.22308` so it can be said that it is a good model predictor algorithm for our data.

# Part 2: Classification (50 Marks)

In this part, you are going to work with "Census Income Dataset" which was originally donated by Ronny Kohavi and Barry Becker to UCI (University of California, Irvine) in 1996. This is a trimmed dataset used for machine learning students to study classification.

This dataset has collected over 40,000 records (we excluded some data in our version) regarding personal yearly income with 12 attributes (predictors). The attributes comprise many aspects of a person that may contribute to the yearly income. You can use summary() function to obtain the attributes information. Your prediction task is to determine whether a person makes over 50K a year.

We have splitted the dataset into a trainning and a testing set. There are 27245 records in the training set while 13631 records in the testing set. Besides the 12 predictors, there is one more column named Salary indicating whether a person's yearly income is over 50K. The label information is a seperated file for the testing set and will be used by us to asess your performance later. Note the label TRUE means an individual's yearly salary exceeds 50K while FALSE means an individual's yearly salary is under 50K.

**Note:** If not explicitly mentioned, libraries are not allowed

In [16]:

```
# Read the data from students' side
remove(list = ls())
train <- read.csv("ClassTrain.csv")
test  <- read.csv("ClassTest.csv")
```

In [ ]:

```
## PLEASE DO NOT ALTER THIS CODE BLOCK
# Please skip (don't run) this if you are a student
# Read in the data from marking tutors' side (ensure no cheating!)
remove(list = ls())
train <- read.csv("../data/ClassTrain.csv")
test  <- read.csv("../data/ClassTest.csv")
label <- read.csv("../data/ClassTestLabel.csv")
```

# Question 1 (10 Marks)

Fit a **Generalized Linear Model (Logistic Regression)** to predict level of income (salary) ( $\geq 50$ K, or $< 50$ K ) using the `train` dataset. Using the results of fitting this model, which predictors do you think are possibly associated with the level of Salary (use `0.05` significant level), and why? Which `three variables` appear to be the strongest predictors of salary, and why?

Furthermore, you can see that you have much more predictors in this part than in the `linear model` from Part 1 $\Rightarrow$ manually checking information is counterproductive. Thus, please write a function to automate these processes **(1)** selecting important feature against 0.05 threshold and **(2)** Selecting three most important features.

**Note**: You don't have to worry about categorical variables here since R can deal with this automatically, focus your efforts on interpretation. Additionally, when explaining why features are strongly associated with the target, please refrain from giving one or two sentences answers, these answers are not descriptive and will result in a deduction of marks. Finally, please name the model here `glm.fit` and have the parameter in the model set to `family = binomial` .

**YOUR ANSWER HERE**

In [17]:

```
# Build your model, keep family = binomial, ignore the warnings, they are benign
glm.fit <-glm(Salary~.,data=train,family = binomial)
summary(glm.fit)
```

```
Warning message:
"glm.fit: fitted probabilities numerically 0 or 1 occurred"


Call:
glm(formula = Salary ~ ., family = binomial, data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-5.1013  -0.5296  -0.1926   0.0276   3.4349

Coefficients:
                                  Estimate Std. Error z value Pr(>|z|)
(Intercept)                     -7.614e+00  4.525e-01 -16.826  < 2e-16 **
*
Age                              2.626e-02  1.779e-03  14.762  < 2e-16 **
*
WorkClassLocal-gov              -7.214e-01  1.168e-01  -6.179 6.46e-10 **
*
WorkClassPrivate                -4.734e-01  9.693e-02  -4.884 1.04e-06 **
*
WorkClassSelf-emp-inc           -2.974e-01  1.283e-01  -2.317 0.020506 *
WorkClassSelf-emp-not-inc       -9.994e-01  1.139e-01  -8.772  < 2e-16 **
*
WorkClassState-gov              -7.757e-01  1.294e-01  -5.996 2.03e-09 **
*
FinalWeight                      7.896e-07  1.822e-07   4.334 1.46e-05 **
*
Education11th                    6.909e-02  2.201e-01   0.314 0.753589
Education12th                    5.005e-01  2.940e-01   1.702 0.088676 .
Education7th-8th                -6.213e-01  2.592e-01  -2.397 0.016530 *
Education9th                    -2.472e-01  2.856e-01  -0.865 0.386877
EducationAssoc-acdm              1.302e+00  1.843e-01   7.066 1.60e-12 **
*
EducationAssoc-voc               1.263e+00  1.772e-01   7.127 1.02e-12 **
*
EducationBachelors               1.931e+00  1.647e-01  11.724  < 2e-16 **
*
EducationDoctorate               3.076e+00  2.380e-01  12.926  < 2e-16 **
*
EducationHS-grad                 7.790e-01  1.598e-01   4.874 1.09e-06 **
*
EducationMasters                 2.319e+00  1.767e-01  13.126  < 2e-16 **
*
EducationProf-school             2.874e+00  2.145e-01  13.396  < 2e-16 **
*
EducationSome-college            1.108e+00  1.622e-01   6.832 8.36e-12 **
*
MaritalStatusMarried-civ-spouse  2.345e+00  3.050e-01   7.687 1.51e-14 **
*
MaritalStatusMarried-spouse-absent -3.345e-02  2.697e-01  -0.124 0.901286
MaritalStatusNever-married      -4.513e-01  9.187e-02  -4.912 9.01e-07 **
*
MaritalStatusSeparated          -9.621e-02  1.733e-01  -0.555 0.578829
MaritalStatusWidowed             1.484e-01  1.656e-01   0.896 0.370163
```

```
OccupationCraft-repair              5.906e-02  8.379e-02   0.705 0.480884
OccupationExec-managerial           7.693e-01  8.089e-02   9.511  < 2e-16 **
*
OccupationFarming-fishing          -9.919e-01  1.457e-01  -6.805 1.01e-11 **
*
OccupationHandlers-cleaners        -7.641e-01  1.529e-01  -4.999 5.77e-07 **
*
OccupationMachine-op-inspct        -2.794e-01  1.073e-01  -2.605 0.009191 **
OccupationOther-service            -8.967e-01  1.300e-01  -6.900 5.22e-12 **
*
OccupationProf-specialty            4.654e-01  8.613e-02   5.403 6.54e-08 **
*
OccupationProtective-serv           6.229e-01  1.302e-01   4.784 1.72e-06 **
*
OccupationSales                     2.770e-01  8.625e-02   3.211 0.001322 **
OccupationTech-support              6.359e-01  1.159e-01   5.488 4.07e-08 **
*
OccupationTransport-moving         -1.027e-01  1.032e-01  -0.995 0.319541
RelationshipNot-in-family           6.652e-01  3.021e-01   2.202 0.027683 *
RelationshipOther-relative         -4.067e-01  2.918e-01  -1.394 0.163395
RelationshipOwn-child              -6.044e-01  2.943e-01  -2.054 0.039994 *
RelationshipUnmarried               5.707e-01  3.174e-01   1.798 0.072185 .
RelationshipWife                    1.332e+00  1.103e-01  12.071  < 2e-16 **
*
RaceAsian-Pac-Islander              9.879e-01  2.997e-01   3.296 0.000979 **
*
RaceBlack                           3.929e-01  2.427e-01   1.619 0.105400
RaceOther                           1.524e-01  4.397e-01   0.347 0.728862
RaceWhite                           5.396e-01  2.305e-01   2.340 0.019266 *
GenderMale                          8.679e-01  8.403e-02  10.328  < 2e-16 **
*
CapitalGain                         3.191e-04  1.107e-05  28.830  < 2e-16 **
*
CapitalLoss                         6.503e-04  3.999e-05  16.264  < 2e-16 **
*
HoursWork                           2.965e-02  1.774e-03  16.714  < 2e-16 **
*
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 31005  on 27244  degrees of freedom
Residual deviance: 17976  on 27196  degrees of freedom
AIC: 18074

Number of Fisher Scoring iterations: 7
```

In [26]:

```
top_feature<-function(model){
    stats<-coef(summary(model))
    bestFeature<-stats[-1,]
    bestFeature<-bestFeature[bestFeature[,"Pr(>|z|)"]<0.05,]
    bestFeature<-sort(bestFeature[,4],decreasing=FALSE)
    top3<-names(bestFeature)[1:3]
    print(bestFeature)
    cat("Top 3 Features are: ",top3)
}
```

In [27]:

```
top_feature(glm.fit)
```

```
                   CapitalGain                     HoursWork 
                  9.086632e-183                  1.033315e-62 
                    CapitalLoss                           Age 
                  1.791718e-59                   2.591338e-49 
           EducationProf-school                EducationMasters 
                  6.362770e-41                   2.349945e-39 
             EducationDoctorate                RelationshipWife 
                  3.224490e-38                   1.499247e-33 
             EducationBachelors                    GenderMale 
                  9.636007e-32                   5.272260e-25 
     OccupationExec-managerial       WorkClassSelf-emp-not-inc 
                  1.887085e-21                   1.747638e-18 
MaritalStatusMarried-civ-spouse              EducationAssoc-voc 
                  1.510914e-14                   1.023955e-12 
           EducationAssoc-acdm         OccupationOther-service 
                  1.599694e-12                   5.217529e-12 
         EducationSome-college       OccupationFarming-fishing 
                  8.363157e-12                   1.007389e-11 
            WorkClassLocal-gov              WorkClassState-gov 
                  6.457094e-10                   2.026573e-09 
        OccupationTech-support        OccupationProf-specialty 
                  4.065421e-08                   6.543650e-08 
    OccupationHandlers-cleaners      MaritalStatusNever-married 
                  5.765385e-07                   9.009754e-07 
             WorkClassPrivate                EducationHS-grad 
                  1.039016e-06                   1.094343e-06 
      OccupationProtective-serv                    FinalWeight 
                  1.718776e-06                   1.464846e-05 
         RaceAsian-Pac-Islander                  OccupationSales 
                  9.792498e-04                   1.321770e-03 
    OccupationMachine-op-inspct                Education7th-8th 
                  9.191178e-03                   1.653000e-02 
                     RaceWhite           WorkClassSelf-emp-inc 
                  1.926637e-02                   2.050583e-02 
      RelationshipNot-in-family           RelationshipOwn-child 
                  2.768284e-02                   3.999385e-02 
Top 3 Features are:  CapitalGain HoursWork CapitalLoss
```

In the above part we have train our train data using the `logistic Model` and from that we got 47 features that are impacting our model but there are `36` variable that have the p-value less than 0.05 significant level thus this means that these features are the most impacting features on our model. These variable are stated below: `Age, WorkClassLocal-gov, WorkClassPrivate, WorkClassSelf-emp-inc, WorkClassSelf-emp-not-inc, WorkClassState-gov, FinalWeight, Education7th-8th, EducationAssoc-acdm,`

`EducationAssoc-voc, EducationBachelors, EducationDoctorate, EducationHS-grad, EducationMasters, EducationProf-school, EducationSome-college, MaritalStatusMarried-civ-spouse, MaritalStatusNever-married, OccupationExec-managerial, OccupationFarming-fishing, OccupationHandlers-cleaners, OccupationMachine-op-inspct, OccupationOther-service, OccupationProf-specialty, OccupationProtective-serv, OccupationSales, OccupationTech-support, RelationshipNot-in-family, RelationshipOwn-child, RelationshipWife, RaceAsian-Pac-Islander, RaceWhite, GenderMale, CapitalGain, CapitalLoss, HoursWork .`

We have created a function to list all the features that have a p-value less than 0.05 and the top 3 features based on their p-value. These top3 features are CapitalGain, HoursWork and CapitalLoss They are the highest impacting variables on our target class that is `Salary` as there p values is the minimum. P-value for these variable are as follow.

- **CapitalGain-** `9.086632e-183`
- **HoursWork-** `1.033315e-62`
- **CapitalLoss-** `1.791718e-59`

As it signifies these values are very very low and thus the impact on Target variable is major and the highest.

# Question 2 (10 Marks)

Firstly, please use the model created in the previous question to predict for the labels of the **train** data. Consequently, our objective is to compare this `predict.label` with the `truth.label` from the **test** data. However, as we don't know the **test** label, we have to estimate model performance using **train** data at this moment.

Secondly, since our objective is to estimate the performance of this model in making correct predictions; thus, this question also asks you to explore different performance metrics (https://en.wikipedia.org/wiki/Precision_and_recall) for classification models. The metrics we will use are **Accuracy, Precision, Recall, and F1 Score**, please create a function to calculate these value and print them out properly using the given structure.

Additionally, please also discuss the results of these values in the context of your model.

**Note**: This asks for your descriptions, please refrain from using one or two lines to describe/discuss the effect. Keep answers to be 4 decimal places

### YOUR ANSWER HERE

In [20]:

```
# Apply your previous model to perform prediction, keep type = "response"
# Don't worry if you receive some warnings, they are benign
predict.label <-predict(glm.fit,type = "response")
# Truth label from train data
truth.label <- train$Salary
```

In [21]:

```
# Model statistics function
mod.stat <- function(predict.label, truth.label){
    # instantiate the variables
    accuracy <- NULL
    precision <- NULL
    recall <- NULL
    F1 <- NULL

    ##############################
    #Your calculatation here


    glm.pred <- ifelse(predict.label > 0.5, "True", "False")
    confumat=table(truth.label,glm.pred)

    TN = confumat[1,1]
    TP = confumat[2,2]
    FN = confumat[2,1]
    FP = confumat[1,2]


    accuracy = (TP + TN)/(TP + TN + FP + FN)#WRITE CODE : formula to find accuracy
    precision = TP / (TP + FP)#WRITE CODE : formula to find precision
    recall = TP / (TP + FN)#WRITE CODE : formula to find recall
    F1=(2)*((precision*recall)/(precision+recall))

    ##############################

    # Return a list of value
    return(list("accuracy" = accuracy, "precision" = precision, "recall" = recall, "fscore"
}
```

In [22]:

```
# Run the function to get statistics, provide description/discussion after this
mod.stat(predict.label, truth.label)
```

**$accuracy**
0.845182602312351
**$precision**
0.739545375672393
**$recall**
0.610689210488609
**$fscore**
0.668968764715115

In the above part we have calculated the most important Stats for our predicted Model. `Accuracy, Precision, Recall, F1` . In order to find these Stats we need to get the confusion matrix to find our the True Positive, True negative, False Negative and False Positive Values. Using these values we can find out our **Accuracy, Precision, Recall, F1**. These Values Can be described as.

`True Positives (TP):` - TP are the correctly predicted positive values which means that the value of actual class is True and the value of predicted class is also True.

`True Negatives (TN):` - TN are the correctly predicted negative values which means that the value of actual class is False and value of predicted class is also False.

**False positives and false negatives, these values occur when your actual class contradicts with the predicted class.**

`False Positives (FP):` FP are when actual class is False and predicted class is True.

`False Negatives (FN):` FN are when actual class is True but predicted class in False.

So Using the confusion matrix and the TP,TN,FP and FN we have calculated our Stats.

**Accuracy:**

Accuracy is the performance measure and it is a ratio of correctly predicted labels to the total observations. Accuracy is a great measure of analysing model performance but when we have symmetric datasets where values of false positive and false negatives are almost same. If we have asymetric data then it might not give us the correct picture. Therefore we find out precision recall and F1 as well.

And Looking at our model We have found out that our Accuracy is around `84.5182%` Which is very good and it means that out model is able to predict approx around 85% correct labels. But we will still find out our Precision recall and F1

**Precision:**

Where as precision is the `ratio of correctly predicted positive observations with total positive predicted observations` .Also high precision refers to low false positive rate. We have got `0.7395` precision which is a good indication.

**Recall:**

Recall is the `ratio of correctly predicted positive observations to the all observations in actual class - True` . We have got recall `of 0.6106` which is good for this model as it's above 0.5

**F1:**

F1 Score is the weighted average of Precision and Recall. Therefore, this score consideres both false positives and false negatives into account. F1 is usually more useful than accuracy, especially if you have an asymetric data. Accuracy is good if we have false positives and false negatives equally in our data. If there are uneven of `false positives and false negatives` , it's ideal to look at both `Precision and Recall` .

In our case, `F1 score is: 0.66896`

# Question 3 (5 Marks)

Use the stepwise selection procedure with the **BIC** penalty to prune out potentially unimportant variables. Checking the performance of your model using the created `mod.stat()` function, please give your discussion as how this model is compared with the `glm.fit` (you can run the `mod.stat()` function for this as well if you want to).

**Note**: please don't change the default direction `both` in the step function, this is so that we can check your work easily. Additionally, please name this model `sw.fit` . Don't worry about the warnings, they are benign

**YOUR ANSWER HERE**

In [23]:

```
# Setting to suppress warnings
options(warn=-1)
# Fit a stepwise model
sw.fit <- step(glm.fit, k = log(nrow(train)), direction = 'both')
# Setting to suppress warnings
options(warn=0)
# Getting the summary to understand the result
summary(sw.fit)
```

```
Start:  AIC=18476.42
Salary ~ Age + WorkClass + FinalWeight + Education + MaritalStatus +
    Occupation + Relationship + Race + Gender + CapitalGain +
    CapitalLoss + HoursWork

                Df Deviance   AIC
- Race           4    17992 18451
<none>                17976 18476
- FinalWeight    1    17995 18485
- MaritalStatus  5    18072 18521
- WorkClass      5    18094 18544
- Gender         1    18087 18577
- Relationship   5    18222 18672
- Age            1    18196 18686
- CapitalLoss    1    18249 18739
- HoursWork      1    18263 18753
- Occupation    11    18490 18878
- Education     12    18871 19249
- CapitalGain    1    19580 20071

Step:  AIC=18451.26
Salary ~ Age + WorkClass + FinalWeight + Education + MaritalStatus +
    Occupation + Relationship + Gender + CapitalGain + CapitalLoss +
    HoursWork

                Df Deviance   AIC
<none>                17992 18451
- FinalWeight    1    18009 18458
+ Race           4    17976 18476
- MaritalStatus  5    18087 18496
- WorkClass      5    18110 18519
- Gender         1    18104 18553
- Relationship   5    18238 18646
- Age            1    18215 18664
- CapitalLoss    1    18266 18715
- HoursWork      1    18279 18728
- Occupation    11    18513 18860
- Education     12    18900 19237
- CapitalGain    1    19593 20042


Call:
glm(formula = Salary ~ Age + WorkClass + FinalWeight + Education +
    MaritalStatus + Occupation + Relationship + Gender + CapitalGain +
    CapitalLoss + HoursWork, family = binomial, data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-5.0961  -0.5291  -0.1936   0.0279   3.4393
```

```
Coefficients:
                                  Estimate Std. Error z value Pr(>|z|)
(Intercept)                     -7.130e+00  3.929e-01 -18.146  < 2e-16 **
*
Age                              2.644e-02  1.778e-03  14.869  < 2e-16 **
*
WorkClassLocal-gov              -7.159e-01  1.163e-01  -6.154 7.54e-10 **
*
WorkClassPrivate                -4.588e-01  9.626e-02  -4.766 1.88e-06 **
*
WorkClassSelf-emp-inc           -2.776e-01  1.278e-01  -2.173  0.02977 *
WorkClassSelf-emp-not-inc       -9.857e-01  1.133e-01  -8.703  < 2e-16 **
*
WorkClassState-gov              -7.653e-01  1.290e-01  -5.930 3.02e-09 **
*
FinalWeight                      7.496e-07  1.800e-07   4.164 3.13e-05 **
*
Education11th                    7.190e-02  2.201e-01   0.327  0.74395
Education12th                    5.065e-01  2.939e-01   1.724  0.08479 .
Education7th-8th                -6.220e-01  2.593e-01  -2.399  0.01643 *
Education9th                    -2.417e-01  2.851e-01  -0.848  0.39663
EducationAssoc-acdm              1.323e+00  1.841e-01   7.187 6.60e-13 **
*
EducationAssoc-voc               1.277e+00  1.770e-01   7.215 5.38e-13 **
*
EducationBachelors               1.947e+00  1.645e-01  11.838  < 2e-16 **
*
EducationDoctorate               3.089e+00  2.377e-01  12.998  < 2e-16 **
*
EducationHS-grad                 7.881e-01  1.596e-01   4.937 7.95e-07 **
*
EducationMasters                 2.337e+00  1.765e-01  13.239  < 2e-16 **
*
EducationProf-school             2.894e+00  2.145e-01  13.495  < 2e-16 **
*
EducationSome-college            1.117e+00  1.621e-01   6.893 5.47e-12 **
*
MaritalStatusMarried-civ-spouse  2.357e+00  3.045e-01   7.743 9.75e-15 **
*
MaritalStatusMarried-spouse-absent -3.049e-02  2.690e-01  -0.113  0.90977
MaritalStatusNever-married      -4.482e-01  9.172e-02  -4.886 1.03e-06 **
*
MaritalStatusSeparated          -1.101e-01  1.728e-01  -0.637  0.52405
MaritalStatusWidowed             1.497e-01  1.655e-01   0.904  0.36583
OccupationCraft-repair           6.350e-02  8.372e-02   0.759  0.44812
OccupationExec-managerial        7.726e-01  8.083e-02   9.558  < 2e-16 **
*
OccupationFarming-fishing       -9.869e-01  1.456e-01  -6.779 1.21e-11 **
*
OccupationHandlers-cleaners     -7.678e-01  1.528e-01  -5.026 5.02e-07 **
*
OccupationMachine-op-inspct     -2.857e-01  1.072e-01  -2.665  0.00770 **
OccupationOther-service         -9.059e-01  1.298e-01  -6.981 2.92e-12 **
*
OccupationProf-specialty         4.656e-01  8.599e-02   5.414 6.15e-08 **
*
OccupationProtective-serv        6.192e-01  1.301e-01   4.760 1.94e-06 **
*
OccupationSales                  2.797e-01  8.618e-02   3.246  0.00117 **
OccupationTech-support           6.444e-01  1.157e-01   5.568 2.57e-08 **
*
```

```
OccupationTransport-moving         -1.082e-01  1.031e-01  -1.050  0.29391
RelationshipNot-in-family           6.761e-01  3.016e-01   2.242  0.02499 *
RelationshipOther-relative         -4.031e-01  2.920e-01  -1.381  0.16742
RelationshipOwn-child              -5.877e-01  2.935e-01  -2.002  0.04525 *
RelationshipUnmarried               5.744e-01  3.168e-01   1.813  0.06984 .
RelationshipWife                    1.332e+00  1.103e-01  12.076  < 2e-16 **
*
GenderMale                          8.747e-01  8.401e-02  10.412  < 2e-16 **
*
CapitalGain                         3.184e-04  1.106e-05  28.784  < 2e-16 **
*
CapitalLoss                         6.509e-04  3.998e-05  16.281  < 2e-16 **
*
HoursWork                           2.968e-02  1.774e-03  16.735  < 2e-16 **
*
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 31005  on 27244  degrees of freedom
Residual deviance: 17992  on 27200  degrees of freedom
AIC: 18082

Number of Fisher Scoring iterations: 7
```

In [24]:

```
top_feature(sw.fit)
```

```
                    CapitalGain                     HoursWork
                  3.406183e-182                  7.289596e-63
                    CapitalLoss                            Age
                   1.356204e-59                  5.261119e-50
           EducationProf-school              EducationMasters
                   1.671541e-41                  5.215997e-40
             EducationDoctorate              RelationshipWife
                   1.252708e-38                  1.410651e-33
             EducationBachelors                  GenderMale
                   2.483657e-32                  2.193947e-25
     OccupationExec-managerial     WorkClassSelf-emp-not-inc
                   1.197051e-21                  3.233980e-18
MaritalStatusMarried-civ-spouse            EducationAssoc-voc
                   9.747441e-15                  5.380651e-13
           EducationAssoc-acdm       OccupationOther-service
                   6.602259e-13                  2.921988e-12
          EducationSome-college     OccupationFarming-fishing
                   5.470929e-12                  1.208215e-11
            WorkClassLocal-gov            WorkClassState-gov
                   7.539377e-10                  3.021284e-09
        OccupationTech-support      OccupationProf-specialty
                   2.573001e-08                  6.151150e-08
    OccupationHandlers-cleaners              EducationHS-grad
                   5.015644e-07                  7.951699e-07
     MaritalStatusNever-married              WorkClassPrivate
                   1.027815e-06                  1.875884e-06
      OccupationProtective-serv                   FinalWeight
                   1.936994e-06                  3.126466e-05
                OccupationSales    OccupationMachine-op-inspct
                   1.170383e-03                  7.696691e-03
               Education7th-8th      RelationshipNot-in-family
                   1.642543e-02                  2.498770e-02
          WorkClassSelf-emp-inc          RelationshipOwn-child
                   2.976914e-02                  4.524916e-02
Top 3 Features are:  CapitalGain HoursWork CapitalLoss
```

In [25]:

```
# Making prediction using train data and view the statistics
predict.label.sw <- predict(glm.fit,type = "response")
# Only run the below if you have labels, in your submission, this must be UNCOMMENTED
mod.stat(predict.label.sw, truth.label)
```

**$accuracy**
0.845182602312351
**$precision**
0.739545375672393
**$recall**
0.610689210488609
**$fscore**
0.668968764715115

PROVIDE DISCUSSION HERE

In this part we are asked to use stepwise BIC pruning on our `Logistic model` that we have fitted in question1 so using the model glm.fit we are pruning our model to remove all the less important variable i.e. those variable that have less impact on our model from that we are left with 33 variables that have p-value less than 0.05 which are stated below.

**CapitalGain, HoursWork , CapitalLoss, Age, EducationProf-school, EducationMasters , EducationDoctorate, RelationshipWife, EducationBachelors, GenderMale, OccupationExec-managerial, WorkClassSelf-emp-not-inc, MaritalStatusMarried-civ-spouse, EducationAssoc-voc, EducationAssoc-acdm, OccupationOther-service, EducationSome-college, OccupationFarming-fishing, WorkClassLocal-gov, WorkClassState-gov, OccupationTech-support, OccupationProf-specialty, OccupationHandlers-cleaners, EducationHS-grad, MaritalStatusNever-married, WorkClassPrivate, OccupationProtective-serv, FinalWeight, OccupationSales, OccupationMachine-op-inspct, Education7th-8th, RelationshipNot-in-family, WorkClassSelf-emp-inc, RelationshipOwn-child**

As we see that out of these 33 varaible 3 has the highest impact on our taget variable which is `Salary` since Salary is of binomial class in form of True and False values we are setting them as True for those that have probabilty greater than 0.5 and False for those that have probabilty less than 0.5. So these top three variables remains same as in part 1 which are `CapitalGain HoursWork CapitalLoss` as their p-value is extremely low signifying that these variable are the major impactor on our target class.

So when we look at our `pruned model` Stats we notices that the Accuracy, precision, recall, and F1 remains the same for the pruned model this is because almost all the variables are significant in deciding the target class variables and the removed variables are completely irrelevent to our model.

# Question 4 (Libraries are allowed) (25 Marks)

Similar to the first part, to simulate for a realistic modelling process, this question will be in the form of a competition among students to find out who has the best model.

Thus, You will be graded by the performance of your model compared to your classmates', the better your model, the higher your score. Additionally, you need to write a short paragraph describing/documenting your thought process in this model building process `(300 words)` . Note that this is to explain to us why you build your current model so that we can verify that you understand the model you build and not just copy from other people.

**Note** Please make sure that we can install the libraries that you use in this part, the code structure can be:

```
install.packages("some package", repos='http://cran.us.r-project.org')
```

```
library("some package")
```

Remember that if we cannot run your code, we will have to give you a deduction, our suggestion is for you to use the standard `R version 3.6.1`

You also need to name your final model `fin.mod` so we can run a check to find out your performance. A good test for your understanding would be to set the previous **BIC model** to be the final model to check if your code works perfectly.

20 Marks for the model performance in the competition

5 Marks for logically writing down the thought process in building the final model

This is the link (https://www.kaggle.com/t/1bdebc96607742dbaf47ab36cd3ae421) to the competition

## YOUR ANSWER HERE

In [31]:

```
install.packages("randomForest", repos='http://cran.us.r-project.org')
library(randomForest)
```

```
Installing package into 'C:/Users/Shranja Sharma/Documents/R/win-library/3.
6'
(as 'lib' is unspecified)
Warning message:
"package 'randomForest' is in use and will not be installed"
```

In [32]:

```
# Build your final model here, use additional coding block if you want to
fin.mod <-randomForest( as.factor(train$Salary) ~., train,ntree=200,mtry=7,importance=TRUE)
# An example would be use the previous model as your final one
#fin.mod <- sw.fit
```

In [33]:

```
# Getting the predict label for the TEST data
pred.label <- predict(fin.mod, test,type = "response")
```

In [34]:

```
# PLEASE DO NOT ALTER THIS CODE BLOCK
# Use this csv file to commit to the leaderboard
write.csv(data.frame("RowIndex" = seq(1, length(pred.label)), "Prediction" = pred.label),
          "ClassPredictLabel.csv", row.names = F)
```

In [ ]:

```
## PLEASE DO NOT ALTER THIS CODE BLOCK
## Please skip (don't run) this if you are a student
## For teaching team use only
source("../data/modassess.r")
model.perf <- mod.stat.test(pred.label,label$Label)
print(model.perf)
```

**ANSWER:**

In this question the best model predictor algorithm i found out was `random forest` as it used for classification and regression where it constructing a multitude of decision trees at training time as random tree can outperform a normal decision tree and can predict values more accuratly when there is a target class as binomial. I have used the target class as factors since it is of binomial class and the values that are returned are in True of False form.

the parameter set in the model are the `ntrees=200` which means that there will be 200 trees form untill the classifier stop i.e. after 200 iteration in tree the classifier will stop this is done to reduce the model to over fit and give us the optimum predicted value against the test data also we have set the importance as `True` which signifies that all varaibles considered are important in assessing the predicted class and train the model.

While `mtry` signifies that number of variables that are available for splitting on each tree node. so we have set it to 7 this is due to the fact that there are 47 variables to predict our model.

after training our model on train data and predicting class for salary in test dataset we found out that the f1 score for our model stands at `0.86891` which tells us that weighted average of the precision and recalls i.e ratio of correctly predicted positive observations with total positive predicted observations and ratio of correctly predicted positive observations to the all observations in actual class - True.

# References

[1] https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/ (https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/)

[2] https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9 (https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9)

[3] https://www.youtube.com/watch?v=S8CRgjJ_XWU&ab_channel=ironfrown (https://www.youtube.com/watch?v=S8CRgjJ_XWU&ab_channel=ironfrown)

[4] https://en.proft.me/2017/01/24/classification-using-random-forest-r/ (https://en.proft.me/2017/01/24/classification-using-random-forest-r/)

[5] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4842399/ (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4842399/)