Logistic Regression Income Prediction:

Dataset: This file contains information on 30725 people and whether their annual income was below $50,000. This will be your master data set. Variables in the data set are:

1. Index: A sequential numbering of cases.
2. Age: The age of the person in years.
3. Sector: The general work sector.
4. Education: The highest level of education achieved.
5. Marital Status: The marital status of the person.
6. Race: The race classification of the individual.
7. Gender: Gender of a person.
8. Hours per week: The number of hours the person works in a week.
9. Bracket: The individual's annual compensation (1 = $50,000 and more).

Convert Sector, Education, Marital Status, Race, Gender and Bracket to factors.
Re-level the following attributes into the intercept as their primary base.

Attribute	Category
Sector	Unemployed
Education	Primary School
Marital Status	Never Married
Race	White

Problem Statement: Using a random number seed, take a random sample of 1600 cases from the full data set in such a way that there should be equal number of rows for income greater than 50k and less than 50k (balanced class).

Build a Logistic Regression Model in R and answer the following questions.

Analysis

1. Parameterize a full logistic regression model with Bracket as the dependent and all other variables as independent (excluding Index).
2. Report the results of the final recommended model from Step 1.
3. State whether the Residual Deviance of the model is markedly different from the Null Deviance.
4. State which variables will have the greatest influence in increasing and decreasing the modelled probability that a person has income greater than 50K?
5. Parameterize a new logistic regression model with the following variables as independents: Age, Education, Marital Status, and Hours per Week.
6. Use the *expand.grid()* command develop a prediction file with all independent variables in the Step 5 model. For binary independent variables use the *unique()* qualifier. For numerical (continuous) independent variables use the *quantile()* qualifier and set test levels at the 25^{th}, 50^{th}, 75^{th}, and 100^{th} percentiles for the variables as appearing in your reduced data set. Calculate and show independent variable values and predicted probabilities for ONLY the first five cases appearing in your prediction file.
7. Based on the predictions generated above, state the maximum and minimum predicted probabilities generated and the independent variable values which resulted in those predictions.

I. Preprocessing

```
#Author: Suryateja Chalapati

#Importing required libraries
rm(list=ls())
library(rio)
library(moments)
library(dplyr)
library(tidyverse)
library(magrittr)
```

```r
#Setting the working directory and importing the dataset
setwd("C:/Users/surya/Downloads")

df = import("Income Data.xlsx", sheet = "Sheet1")
colnames(df)=tolower(make.names(colnames(df)))
str(df)

## 'data.frame':    30725 obs. of  9 variables:
##  $ index         : num  1 2 3 4 5 6 7 8 9 10 ...
##  $ age           : num  39 50 38 53 28 37 49 52 31 42 ...
##  $ sector        : chr  "Government" "Self" "Private" "Private" ...
##  $ education     : chr  "Bachelors" "Bachelors" "High School" "High School" ...
##  $ marital.status: chr  "Never Married" "Married" "Divorced" "Married" ...
##  $ race          : chr  "White" "White" "White" "Black" ...
##  $ gender        : chr  "Male" "Male" "Male" "Male" ...
##  $ hours.per.week: num  40 13 40 40 40 40 16 45 50 40 ...
##  $ bracket       : num  0 0 0 0 0 0 0 1 1 1 ...

#Converting to factor variables and Re-levelling
cols <- c("sector", "education", "marital.status", "race", "gender", "bracket")

df %<>% mutate_at(cols, funs(factor(.)))
str(df)

## 'data.frame':    30725 obs. of  9 variables:
##  $ index         : num  1 2 3 4 5 6 7 8 9 10 ...
##  $ age           : num  39 50 38 53 28 37 49 52 31 42 ...
##  $ sector        : Factor w/ 4 levels "Government","Private",..: 1 3 2 2 2 2 2 3 2 2 ...
##  $ education     : Factor w/ 5 levels "Bachelors","High School",..: 1 1 2 2 1 3 4 2 3 1 ...
##  $ marital.status: Factor w/ 4 levels "Divorced","Married",..: 3 2 1 2 2 2 2 2 3 2 ...
##  $ race          : Factor w/ 4 levels "Asian","Black",..: 4 4 4 2 2 4 2 4 4 4 ...
##  $ gender        : Factor w/ 2 levels "Female","Male": 2 2 2 2 1 1 1 2 1 2 ...
##  $ hours.per.week: num  40 13 40 40 40 40 16 45 50 40 ...
##  $ bracket       : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 2 2 2 ...

df$sector <- relevel(df$sector, "Unemployed")
df$education <- relevel(df$education, "Primary School")
df$marital.status <- relevel(df$marital.status, "Never Married")
df$race <- relevel(df$race, "White")

#Setting seed and data sampling for equal distribution on brackets
set.seed(36991670)
data_sample = data.frame(df[sample(1:nrow(df), 1600, replace = FALSE),])
data_sample = df %>% group_by(bracket) %>% sample_n(800)
table(data_sample$bracket)

##
##   0   1
## 800 800

attach(data_sample)
```

II. Analysis

```r
#Analysis_1
log.out = glm(bracket~.-index, data = data_sample, family = "binomial")
```

- Full logistic regression model above.

```r
#Analysis_2
```

```
#Logistics Regression
summary(log.out)

##
## Call:
## glm(formula = bracket ~ . - index, family = "binomial", data = data_sample)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3392  -0.6719   0.1171   0.7766   2.6463
##
## Coefficients:
##                           Estimate Std. Error z value Pr(>|z|)
## (Intercept)             -20.468495 324.745831  -0.063  0.94974
## age                       0.031504   0.006194   5.086 3.66e-07 ***
## sectorGovernment         13.188323 324.743803   0.041  0.96761
## sectorPrivate            12.814328 324.743771   0.039  0.96852
## sectorSelf               12.815144 324.743800   0.039  0.96852
## educationBachelors        3.389542   1.076469   3.149  0.00164 **
## educationHigh School      2.435642   1.074365   2.267  0.02339 *
## educationMasters          3.468146   1.080748   3.209  0.00133 **
## educationMiddle School    1.205526   1.147118   1.051  0.29330
## marital.statusDivorced    0.602939   0.257304   2.343  0.01911 *
## marital.statusMarried     2.395245   0.203866  11.749  < 2e-16 ***
## marital.statusWidowed     1.138771   0.449008   2.536  0.01121 *
## raceAsian                -0.294256   0.395198  -0.745  0.45653
## raceBlack                -0.427774   0.223189  -1.917  0.05528 .
## raceOther                -1.099723   0.816832  -1.346  0.17820
## genderMale                0.346140   0.166346   2.081  0.03745 *
## hours.per.week            0.033884   0.006041   5.609 2.03e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2218.1  on 1599  degrees of freedom
## Residual deviance: 1571.1  on 1583  degrees of freedom
## AIC: 1605.1
##
## Number of Fisher Scoring iterations: 11
```

- Summary output for full logistic regression above.

```
#Analysis_3
#Null Deviance vs Residual Deviance
```

- The difference between Null deviance and Residual deviance is [651]. The greater the difference the better. Null deviance value is when we only have intercept in the equation and no other variables and Residual deviance value is when taking all the other variables into account. This model can be considered as there is a significant difference between both deviances.

```
#Analysis_4
```

- The variables **sector** seems to have greatest increasing influence and **race** seems to have greater decreasing influence.

```
#Analysis_5
```

```r
log.out1 = glm(bracket~age+education+marital.status+hours.per.week, data = data_sample, family
= "binomial")
summary(log.out1)
```

```
##
## Call:
## glm(formula = bracket ~ age + education + marital.status + hours.per.week,
##     family = "binomial", data = data_sample)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.37694  -0.68663   0.08845   0.78067   2.68500
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)             -7.646998   1.142218  -6.695 2.16e-11 ***
## age                      0.032831   0.006026   5.448 5.09e-08 ***
## educationBachelors       3.393997   1.073833   3.161  0.00157 **
## educationHigh School     2.437063   1.072068   2.273  0.02301 *
## educationMasters         3.504582   1.077375   3.253  0.00114 **
## educationMiddle School  1.193440   1.146056   1.041  0.29772
## marital.statusDivorced  0.536191   0.254127   2.110  0.03486 *
## marital.statusMarried   2.470059   0.198277  12.458  < 2e-16 ***
## marital.statusWidowed   1.078091   0.442723   2.435  0.01489 *
## hours.per.week           0.037546   0.005883   6.382 1.74e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2218.1  on 1599  degrees of freedom
## Residual deviance: 1586.0  on 1590  degrees of freedom
## AIC: 1606
##
## Number of Fisher Scoring iterations: 5
```

```r
#Analysis_6
pred.sample = expand.grid(age = quantile(data_sample$age, c(.25,.50,.75,1)),
                          education = unique(data_sample$education),
                          marital.status = unique(data_sample$marital.status),
                          hours.per.week = quantile(data_sample$hours.per.week, c(.25,.50,.75,
1)))

pred.sample$pred.prob = predict(log.out1, newdata=pred.sample, type='response')

head(pred.sample,5)
```

```
##   age   education marital.status hours.per.week pred.prob
## 1  31   Bachelors  Never Married             40   0.15015
## 2  40   Bachelors  Never Married             40   0.19187
## 3  49   Bachelors  Never Married             40   0.24187
## 4  90   Bachelors  Never Married             40   0.55073
## 5  31 High School  Never Married             40   0.06355
```

- Showing top 5 results.

```r
#Analysis_7
max_row = pred.sample[pred.sample$pred.prob == max(pred.sample$pred.prob),]
max_row
```

```
##     age education marital.status hours.per.week pred.prob
## 296  90   Masters        Married             99 0.9933035
```

```
min_row = pred.sample[pred.sample$pred.prob == min(pred.sample$pred.prob),]
min_row
```

```
##    age      education marital.status hours.per.week   pred.prob
## 9   31 Primary School  Never Married             40 0.005896918
## 89  31 Primary School  Never Married             40 0.005896918
```

- Maximum predicted probability is 0.9933, for age=90, education=masters, marital status=married, hours per week=99.
- Minimum predicted probability for two cases is 0.0058, for age=31, education=primary school, marital status=never married, hours per week=40.