In [1]:

```
setwd("H:/PS/Logistic_Regression")
adultCensus = read.csv("adultcensus.csv")
```

In [26]:

```
library(e1071)
library(stringr)
library(ggcorrplot)
library(stringi)
library(devtools)
library(InformationValue)
library(caret)
library(tidyverse)
library(caret)
library(ROCR)
options(scipen = 999)
```

```
Loading required package: gplots

Attaching package: 'gplots'

The following object is masked from 'package:stats':

    lowess
```

In [4]:

```
#Check for missing data. The dataset has "?" question marks in place of missing values.
#This has been taken care of using the following:
#Getting rid of such records:
sapply(adultCensus, function(x) sum(is.na(x)))
#checking for each column:
#adultCensus = adultCensus[adultCensus$occupation != "?", ]
```

age
0
workclass
0
fnlwgt
0
education
0
edu_num
0
marital_status
0
occupation
0
relationship
0
race
0
sex
0
capital.gain
0
capital.loss
0
hours_per_week
0
Native
0
income_class
0

In [5]:

```
#Checking through all the columns:
No_ques_marks <- function(x, column){
  for (column in colnames(adultCensus)){
    print("?" %in% x[, column])
  }
}
No_ques_marks(adultCensus, colnames(adultCensus))
```

```
[1] FALSE
[1] FALSE
[1] FALSE
[1] FALSE
[1] FALSE
[1] FALSE
[1] FALSE
[1] FALSE
[1] FALSE
[1] FALSE
[1] FALSE
[1] FALSE
[1] FALSE
[1] FALSE
[1] FALSE
```

In [6]:

```
#Viewing data type of each column:
sapply(adultCensus, class)
```

age
'integer'
workclass
'factor'
fnlwgt
'integer'
education
'factor'
edu_num
'integer'
marital_status
'factor'
occupation
'factor'
relationship
'factor'
race
'factor'
sex
'factor'
capital.gain
'integer'
capital.loss
'integer'
hours_per_week
'integer'
Native
'factor'
income_class
'factor'

In [7]:

```
head(adultCensus)
```

age	workclass	fnlwgt	education	edu_num	marital_status	occupation	relationship	race
39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White
50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White
38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White
53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black
28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black
37	Private	284582	Masters	14	Married-civ-spouse	Exec-managerial	Wife	White

In [8]:

```
tail(adultCensus)
```

	age	workclass	fnlwgt	education	edu_num	marital_status	occupation	relationship
38320	48	Local-gov	349230	Masters	14	Divorced	Other-service	Not-in-family
38321	33	Private	245211	Bachelors	13	Never-married	Prof-specialty	Own-child
38322	39	Private	215419	Bachelors	13	Divorced	Prof-specialty	Not-in-family
38323	38	Private	374983	Bachelors	13	Married-civ-spouse	Prof-specialty	Husband
38324	44	Private	83891	Bachelors	13	Divorced	Adm-clerical	Own-child
38325	35	Self-emp-inc	182148	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband

In [9]:

```
#Label Encoding for outcome variable:
unique(adultCensus$income_class)
adultCensus$income_class[adultCensus$income_class == "<=50K."] <- "<=50K"
adultCensus$income_class[adultCensus$income_class == ">50K."] <- ">50K"
```

<=50K >50K <=50K. >50K.

▶ **Levels**:

In [10]:

```
adultCensus$class <- factor(adultCensus$income_class,
                   levels = c('<=50K','>50K'),
                   labels = c(0,1))
#Verifying that the outcome variable is binary
unique(adultCensus$class)
```

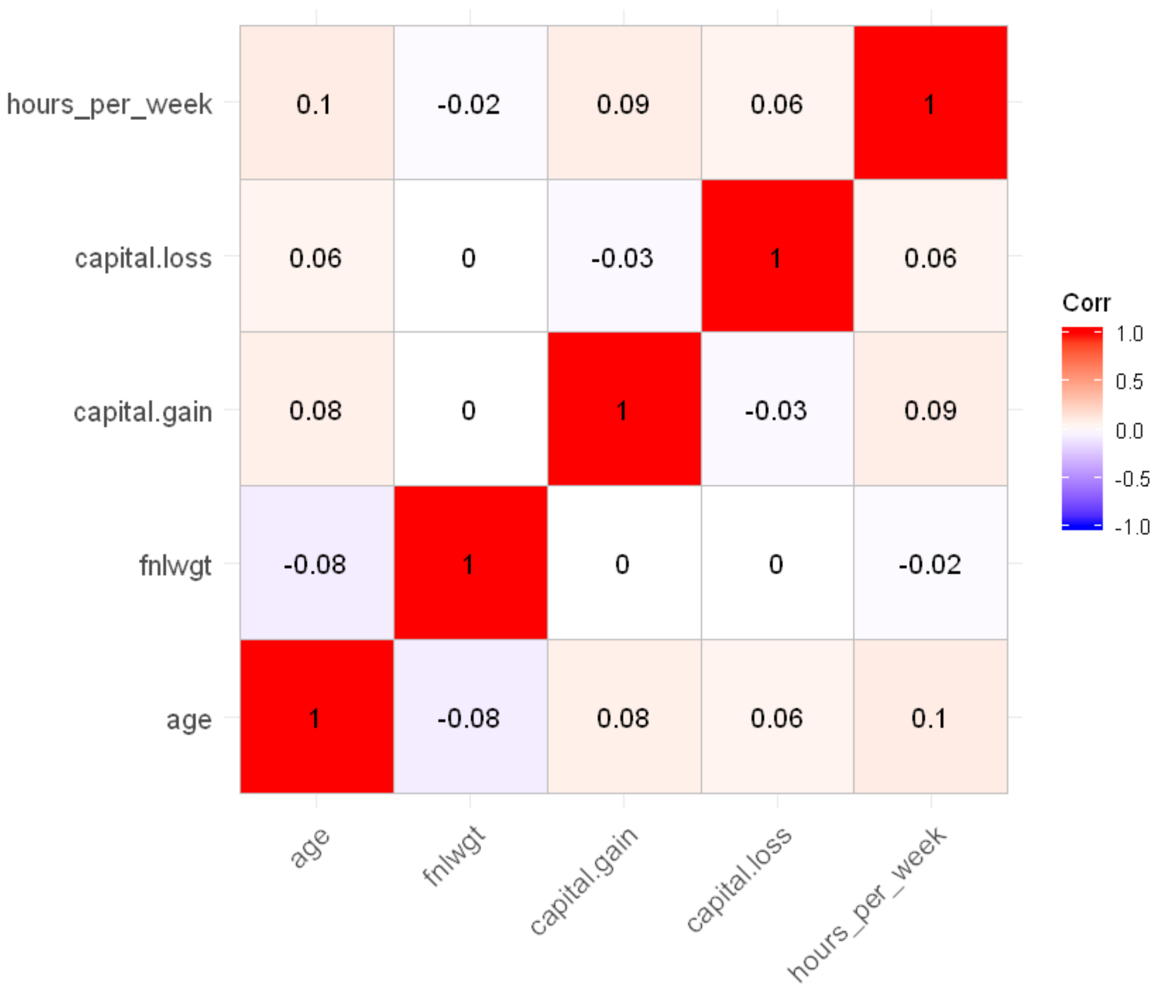
0 1

▶ **Levels**:

In [11]:

```
#Collecting only continuous variables:
acNumerical <- adultCensus[c(1,3,11,12,13)]
acNumerical <- scale(acNumerical)
acNumerical <- data.frame(acNumerical)
#Visualizing correlation amongst continuous variables:
acNum_cor <- cor(acNumerical)
acNum_cor
ggcorrplot(acNum_cor, lab = TRUE)
```

	age	fnlwgt	capital.gain	capital.loss	hours_per_week
age	1.00000000	-0.075607324	0.076862188	0.05804258	0.10144964
fnlwgt	-0.07560732	1.000000000	-0.002882717	-0.00385697	-0.01569907
capital.gain	0.07686219	-0.002882717	1.000000000	-0.03185829	0.08508155
capital.loss	0.05804258	-0.003856970	-0.031858291	1.00000000	0.05567208
hours_per_week	0.10144964	-0.015699068	0.085081554	0.05567208	1.00000000



In [12]:

```
acNumerical$class <- adultCensus$class
#checking correlations with continuous variables
ConVar_cor <- manova(cbind(age, fnlwgt, capital.gain, capital.loss, hours_per_week) ~ class
summary(ConVar_cor)
summary.aov(ConVar_cor)
#Thus, variable "fnlwgt" fails to have a significant impact on income class. We exclude thi
```

```
             Df  Pillai approx F num Df den Df                Pr(>F)
class         1 0.14782   1329.3      5  38319 < 0.00000000000000022 ***
Residuals 38323
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

 Response age :
               Df Sum Sq Mean Sq F value                Pr(>F)
class           1   2087 2086.80  2206.9 < 0.00000000000000022 ***
Residuals   38323  36237    0.95
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

 Response fnlwgt :
               Df Sum Sq Mean Sq F value Pr(>F)
class           1      1  1.1361  1.1361 0.2865
Residuals   38323  38323  1.0000

 Response capital.gain :
               Df Sum Sq Mean Sq F value                Pr(>F)
class           1   1855 1854.68  1948.9 < 0.00000000000000022 ***
Residuals   38323  36469    0.95
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

 Response capital.loss :
               Df Sum Sq Mean Sq F value                Pr(>F)
class           1    830  829.80  848.14 < 0.00000000000000022 ***
Residuals   38323  37494    0.98
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

 Response hours_per_week :
               Df Sum Sq Mean Sq F value                Pr(>F)
class           1   1931 1930.53  2032.9 < 0.00000000000000022 ***
Residuals   38323  36393    0.95
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

In [15]:

```
#We first run the model with selected continuous and all categorical variables.
#Deleting edu_num as it is redundant, along with income_class and fnlwgt.
print("adultCensus:")
head(adultCensus,1)
ConVarScaled <- adultCensus[-c(3,5,15)]
print("ConVarScaled:")
head(ConVarScaled,1)
#Feature Scaling for continuous Variables:
ConVarScaled[c(1,9,10,11)] = scale(ConVarScaled[c(1,9,10,11)])
```

```
[1] "adultCensus:"
```

age	workclass	fnlwgt	education	edu_num	marital_status	occupation	relationship	race	s
39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	M

```
[1] "ConVarScaled:"
```

age	workclass	education	marital_status	occupation	relationship	race	sex	capital.gain	c
39	State-gov	Bachelors	Never-married	Adm-clerical	Not-in-family	White	Male	2174	

In [16]:

```
model_All <- glm(formula = class ~ ., data = ConVarScaled, family = binomial ,control = lis
summary(model_All)
#Checking Multi-collinearity by examining VIF values for Categorical Variables
car::vif(model_All)
```

```
NativeHong                                0.571620
NativeHungary                             0.859681
NativeIndia                               0.431906
NativeIran                                0.594665
NativeIreland                             0.229003
NativeItaly                               0.426222
NativeJamaica                             0.476200
NativeJapan                               0.536308
NativeLaos                                0.086739 .
NativeMexico                              0.190512
NativeNicaragua                           0.671552
NativeOutlying-US(Guam-USVI-etc)          0.679974
NativePeru                                0.194582
NativePhilippines                         0.801810
NativePoland                              0.936159
NativePortugal                            0.335818
NativePuerto-Rico                         0.748346
NativeScotland                            0.104916
NativeSouth                               0.044616 *
```

In [17]:

```
#We need to remove the variable with highest GVIF, which is "relationship".
ConVarScaled <- ConVarScaled[-c(6)]
model_All <- glm(formula = class ~ ., data = ConVarScaled, family = binomial ,control = lis
summary(model_All)
#Checking Multi-collinearity by examining VIF values for Categorical Variables
car::vif(model_All)
```

```
marital_statusWidowed                          0.959023
occupationArmed-Forces                         0.912513
occupationCraft-repair                         0.652730
occupationExec-managerial           < 0.0000000000000002 ***
occupationFarming-fishing           < 0.0000000000000002 ***
occupationHandlers-cleaners         0.000000028755090292 ***
occupationMachine-op-inspct                    0.000218 ***
occupationOther-service             0.000000000000000267 ***
occupationPriv-house-serv                      0.008852 **
occupationProf-specialty            0.000000000000011514 ***
occupationProtective-serv                      0.000108 ***
occupationSales                                0.002642 **
occupationTech-support              0.000000058147275000 ***
occupationTransport-moving                     0.280615
raceAsian-Pac-Islander                         0.002915 **
raceBlack                                      0.149951
raceOther                                      0.182479
raceWhite                                      0.011517 *
sexMale                                        0.004706 **
capital_gain                        < 0.0000000000000002 ***
```

In [18]:

```
Cat_vars <- c ("workclass", "education", "marital_status", "occupation", "race", "sex", "Na
Cat_infoval <- data.frame(VARS=Cat_vars, IV=numeric(length(Cat_vars)), STRENGTH=character(l
for (Cat_var in Cat_vars){
  Cat_infoval[Cat_infoval$VARS == Cat_var, "IV"] <- InformationValue::IV(X=adultCensus[, Ca
  Cat_infoval[Cat_infoval$VARS == Cat_var, "STRENGTH"] <- attr(InformationValue::IV(X=adult
}
Cat_infoval <- Cat_infoval[order(-Cat_infoval$IV), ]
Cat_infoval
```

	VARS	IV	STRENGTH
3	marital_status	1.32385744	Highly Predictive
4	occupation	0.74746648	Highly Predictive
2	education	0.71304603	Highly Predictive
6	sex	0.29981097	Highly Predictive
1	workclass	0.12423435	Highly Predictive
7	Native	0.08100767	Somewhat Predictive
5	race	0.06611061	Somewhat Predictive

In [20]:

```
#We would continue to include the bottom variables as well.
#Select significant features:
print("ConVarScaled:")
head(ConVarScaled,1)
Sig_Data <- ConVarScaled#[-c(6,11)]

#Check class distribution
#Baseline accuracy:
table(adultCensus$class)
```

```
[1] "ConVarScaled:"
```

age	workclass	education	marital_status	occupation	race	sex	capital.gain	capital.l
0.03042553	State-gov	Bachelors	Never-married	Adm-clerical	White	Male	0.1413175	-0.2182

```
    0     1
28875  9450
```

In [21]:

```
#Baseline accuracy
baseline <- round(28875/nrow(adultCensus),2)
baseline
```

0.75

```
Our model accuracy should at least be 75%
```

In [22]:

```
#Split into training and testing:
set.seed(123)
training_samples <- Sig_Data$class %>%
  createDataPartition(p = 0.70, list = FALSE)
train  <- Sig_Data[training_samples, ]
test <- Sig_Data[-training_samples, ]
head(test,1)
```

	age	workclass	education	marital_status	occupation	race	sex	capital.gain	ca
6	-0.1202864	Private	Masters	Married-civ-spouse	Exec-managerial	White	Female	-0.1459554	-0

In [23]:

```
#Applying model with selected features:
mod_sig <- glm(formula = class ~ ., data = train, family = binomial(link = "logit") ,contro
summary(mod_sig)
```

```
marital_statusSeparated                    0.403814
marital_statusWidowed                      0.750462
occupationArmed-Forces                     0.814173
occupationCraft-repair                     0.909088
occupationExec-managerial           < 0.0000000000000002 ***
occupationFarming-fishing            0.00000000000292980 ***
occupationHandlers-cleaners          0.00000906581999367 ***
occupationMachine-op-inspct                0.002342 **
occupationOther-service              0.00000000000298032 ***
occupationPriv-house-serv                  0.047004 *
occupationProf-specialty             0.00000000148845158 ***
occupationProtective-serv                  0.001614 **
occupationSales                            0.005079 **
occupationTech-support               0.00000264134385843 ***
occupationTransport-moving                 0.357050
raceAsian-Pac-Islander                     0.006220 **
raceBlack                                  0.419852
raceOther                                  0.462269
raceWhite                                  0.061710 .
```

```
AIC has reduced from 25290 to 17755
```

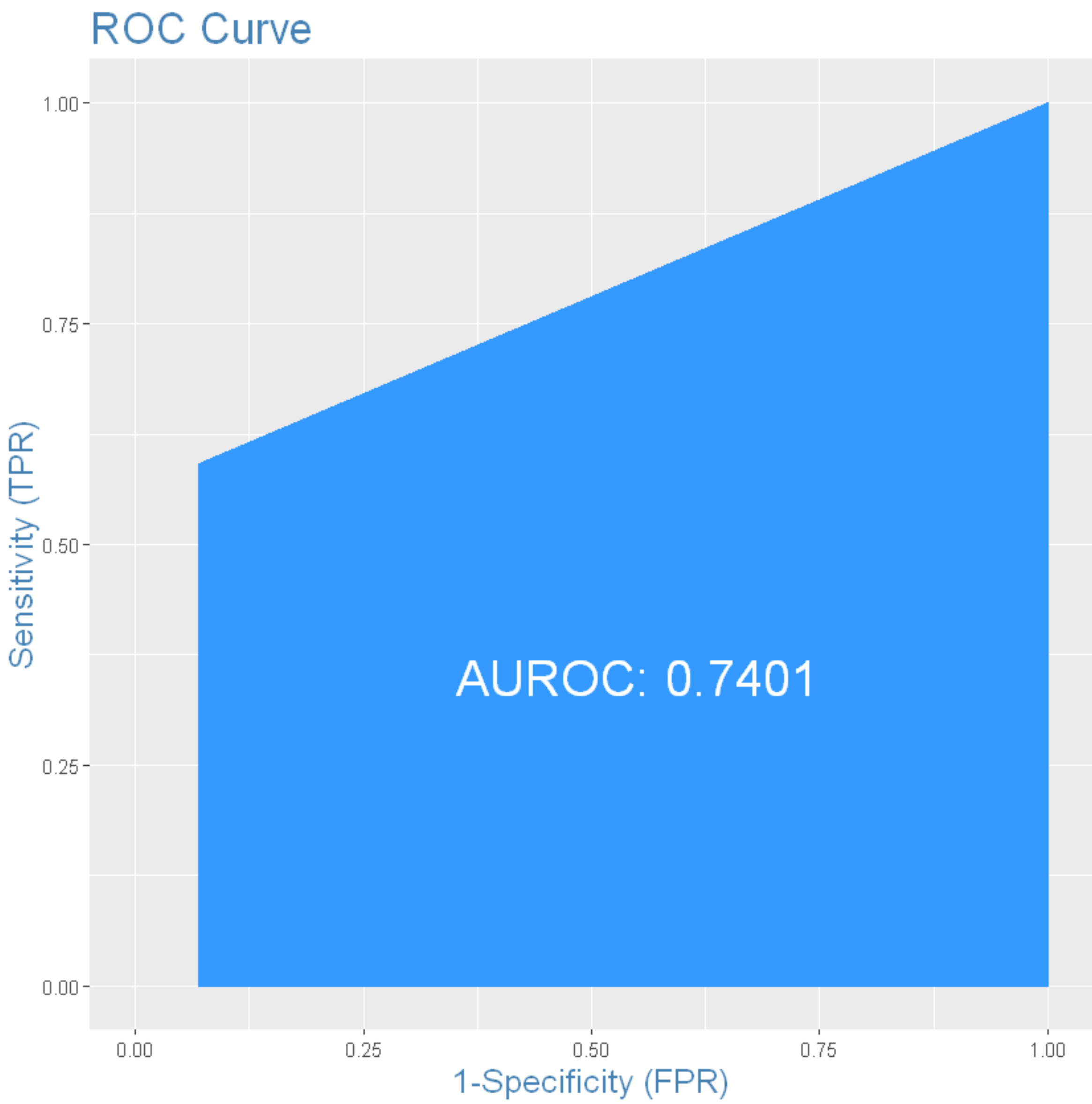
In [24]:

```
# Predicting the Test set results
pred_modsig = predict(mod_sig, type = 'response', newdata = test[-12])
#find optimal threshold:
library(InformationValue)
oc <- optimalCutoff(test$class, pred_modsig)[1]
oc
```

0.51

In [30]:

```
p_class_modsig = ifelse(pred_modsig > oc, 1, 0)
#roc.curve(test$class, p_class_modsig)
plotROC(test$class, p_class_modsig)
```



In [32]:

```
ypred <- as.data.frame(p_class_modsig)
sapply(ypred, class)

unique(ypred$p_class_modsig)
ypred$p_class_modsig <- factor(ypred$p_class_modsig,
                               levels = c('0','1'),
                               labels = c(0,1))
sapply(ypred, class)
```

p_class_modsig: 'numeric'

1 0

p_class_modsig: 'factor'

In [33]:

```
confusionMatrix(reference = test$class, data = ypred$p_class_modsig, positive = '1')
```

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 8058 1159
         1  604 1676

               Accuracy : 0.8467
                 95% CI : (0.8399, 0.8532)
    No Information Rate : 0.7534
    P-Value [Acc > NIR] : < 0.00000000000000022

                  Kappa : 0.5582
 Mcnemar's Test P-Value : < 0.00000000000000022

            Sensitivity : 0.5912
            Specificity : 0.9303
         Pos Pred Value : 0.7351
         Neg Pred Value : 0.8743
             Prevalence : 0.2466
         Detection Rate : 0.1458
   Detection Prevalence : 0.1983
      Balanced Accuracy : 0.7607

       'Positive' Class : 1
```

```
The Model has performed better than the baseline Accuracy of 75% to now 84.67%.
However, the accuracy can be misleading when we have imabalanced classes.
Sensitivity has taken a hit, being at 0.5912
Specificity looks good at 0.9303
Precision would just be average at 0.7351

I would like to treat class imbalance and examine results for new training and testing
set, and provide EDA as well.
This file will be updated.
```