UNIVERSITY OF MARYLAND
ROBERT H. SMITH SCHOOL OF BUSINESS

BUDT758T

DATA MINING AND PREDICTIVE ANALYTICS

Individual Assignment 2

NAME (in capitals): _____SHRUTI GUPTA_____

- Please submit on Canvas.
- Your submission should consist of this document (with answers filled in in the appropriate places).
- Please ensure that answers are appropriately numbered and clearly legible.
- In the space below please enter the following text and initial below: "I pledge on my honor that I have not given or received unauthorized assistance on this assignment."

HONOR PLEDGE: I pledge on my honor that I have not given or received unauthorized assistance on this assignment. YOUR INITIALS: Shruti Gupta

The goal of this homework is to introduce you to classification concepts. You will develop (1) a linear probability model and (2) a logistic regression model. You will need to create random partitions of a data set, build your model on the training data set and then compute prediction errors using the test data set. There are a couple of helpful hints at the end of the assignment.

The Assignment

The data in the accompanying file "VoterPref.csv" (posted on Canvas) contains data from a survey of random sample of registered voters in a state. The subjects were asked whether they were "For" or "Against" a proposal on the ballot to increase the state sales tax by 0.5%, with the stipulation that the additional tax revenues be spent on education. In addition to their position on the proposition, some additional demographic information is collected. The variables in the data set are:

PREFERENCE	"For" or "Against"
AGE	Years of age at time of survey
INCOME	Annual income in thousands of US dollars
GENDER	"M" or "F"

The intent of the survey is to develop a strategy to target individuals for a marketing campaign designed to "get out the vote".

(1) Data Preparation

a. Read the data set in *R*. For the PREFERENCE variable ensure that "Against" is the success class (i.e. the class with higher level – e.g. "1" for binary variable)

```
setwd("~/Desktop/Sem2/Data MIning - Kislay Prasad/Assignment2")
df <- read.csv("VoterPref.csv")
df
df$PREFERENCE <- ifelse(df$PREFERENCE == "Against", "1", "0")
```

b. Set the seed to 71923

```
set.seed(71923)
```

c. Randomly partition the data set into the *training* and *test* data sets. The proportion of observations in the training data set should be 70%. The remaining 30% of observations should be in the test data set.

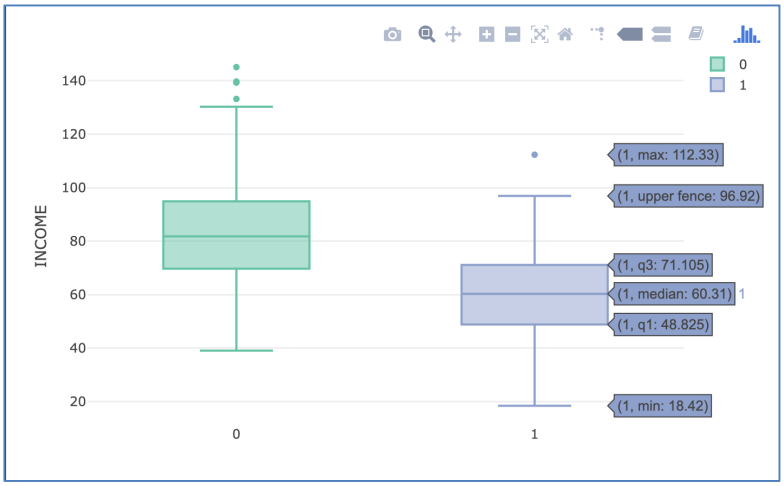
```
datasplit <- sample(nrow(df), 0.7*nrow(df))
df_train <- data.frame(df[datasplit,])
df_test <- data.frame(df[-datasplit,])
```

(2) Exploratory analysis of the *training* data set

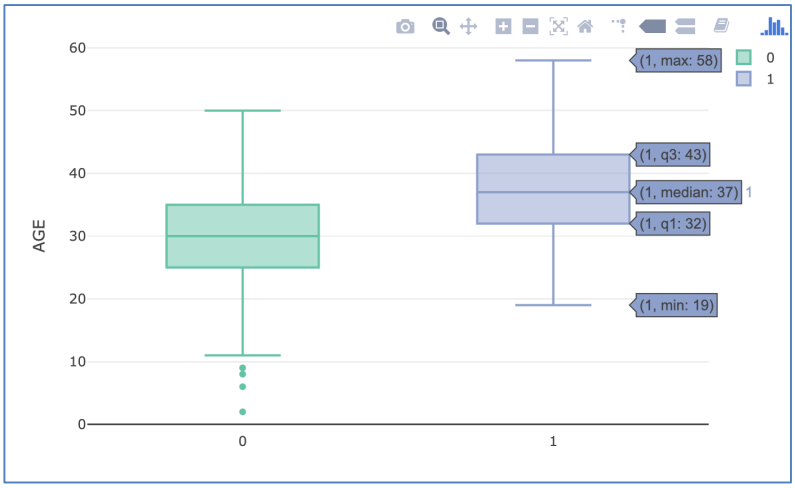
a. Construct boxplots of INCOME and AGE (broken up by values of PREFERENCE). Present the plot as **Exhibit A**. What do you observe?

```
library(plotly)
plot_ly(df_train, y = ~INCOME, color = ~PREFERENCE, type = "box")
plot_ly(df_train, y = ~AGE, color = ~PREFERENCE, type = "box")
```

EXHIBIT A



We observe that people with higher income have voted in favor of increase in sales tax while people with lower income have voted against it



We observe that people with higher ages have voted against increase in sales tax while people with lower ages have voted in favor of it

b. Construct a table for PREFERENCE showing proportions for and against.

```
proportn_pref <- prop.table(table(df_train$PREFERENCE))
rownames(proportn_pref) <- c("For","Against")
proportn_pref
      For      Against
0.8128571  0.1871429
```

c. Construct a two-way table for count of PREFERENCE broken up by GENDER (i.e. what are the numbers of men and women who are for and against the proposition).

```
genderproportn <-table(df_train$PREFERENCE, df_train$GENDER)
genderproportn
```

```
colnames(genderproportn) <- c("Female","Male")
rownames(genderproportn) <- c("For","Against")
genderproportn

        Female Male
For        276  293
Against     76   55
```

(3) Run a linear regression model of PREFERENCE on the demographic variables. Use only the training data set for fitting the model.

```
library(Metrics)
training_model <- lm(PREFERENCE ~ AGE + GENDER + INCOME, data = df_train)
summary(training_model)

Call:
lm(formula = PREFERENCE ~ AGE + GENDER + INCOME, data = df_train)

Residuals:
    Min      1Q  Median      3Q
-0.74013 -0.20850 -0.06941  0.16001
    Max
 0.89611

Coefficients:
            Estimate Std. Error t value
(Intercept)  0.3490527  0.0632336   5.520
AGE          0.0202591  0.0014690  13.791
GENDERM     -0.0721760  0.0234616  -3.076
INCOME      -0.0096474  0.0005916 -16.308
            Pr(>|t|)
(Intercept) 4.79e-08 ***
AGE          < 2e-16 ***
GENDERM      0.00218 **
INCOME       < 2e-16 ***
---
Signif. codes:
 0 '***' 0.001 '**' 0.01 '*' 0.05 '.'
 0.1 ' ' 1

Residual standard error: 0.3102 on 696 degrees of freedom
Multiple R-squared:  0.371,          Adjusted R-squared:  0.3683
F-statistic: 136.8 on 3 and 696 DF,  p-value: < 2.2e-16
```

a. Compute the average error, RMSE and the mean absolute error (MAE) for both in-sample predictions (i.e. for the training data set) and the out-of-sample predictions (i.e. for the test data set). Use predicted values from the regression equation (do **not** do the classification for this yet).

```
library(Metrics)
training_model <- lm(PREFERENCE ~ AGE + GENDER + INCOME, data = df_train)
summary(training_model)
training_AvgError <- mean(as.numeric(df_train$PREFERENCE) - training_model$fitted.values)
training_RMSE <- rmse(as.numeric(df_train$PREFERENCE), training_model$fitted.values)
training_MAE <- mae(as.numeric(df_train$PREFERENCE), training_model$fitted.values)
cat("training_AvgError", training_AvgError,"\ntraining_RMSE", training_RMSE, "\ntraining_MAE", training_MAE)
```

```
training_AvgError -4.855687e-18
training_RMSE 0.3093274
training_MAE 0.2429269
```

```
test_model <- lm(PREFERENCE ~ AGE + GENDER + INCOME, data = df_test)
summary(test_model)
test_AvgError <- mean(as.numeric(df_test$PREFERENCE) - test_model$fitted.values)
test_RMSE <- rmse(as.numeric(df_test$PREFERENCE), test_model$fitted.values)
test_MAE <- mae(as.numeric(df_test$PREFERENCE), test_model$fitted.values)
cat("test_AvgError", test_AvgError,"\ntest_RMSE", test_RMSE, "\ntest_MAE", test_MAE)
```

```
test_AvgError 2.309531e-18
test_RMSE 0.3198901
test_MAE 0.2625628
```

b. For which data set are these errors smaller?

	Avg Error	RMSE	MAE
Test_model	*2.309531e-18*	*0.3198901*	*0.2625628*
Train_model	*-4.855687e-18*	*0.3093274*	*0.2429269*

It is lesser for the training model and more for the Test model

c. Use a cutoff of 0.5 and do the classification (i.e. make the class predictions). What proportions of predicted classes are for and against in each data set?

```
pred <- ifelse(training_model$fitted.values>0.5,1,0)
train_proportions <- prop.table(table(pred))
rownames(train_proportions) <- c("For","Against")
train_proportions
```

```
pred1 <- ifelse(test_model$fitted.values>0.5,1,0)
```

```
test_proportions <- prop.table(table(pred1))
rownames(test_proportions) <- c("For","Against")
test_proportions
```

```
pred
     For          Against
0.90285714    0.09714286

pred1
    For          Against
0.8966667     0.1033333
```

d. What proportion of class predictions are in error in each of the training and test data set?

```
predcn1 <- ifelse(training_model$fitted.values > 0.5, 1, 0)
confusionTabe <- table(df_train$PREFERENCE, predcn1)
rownames(confusionTabe) <- c("For","Against")
colnames(confusionTabe) <- c("For","Against")
prop.table(confusionTabe)

predcn2 <- ifelse(test_model$fitted.values > 0.5, 1, 0)
confusionTabe1 <- table(df_test$PREFERENCE, predcn2)
rownames(confusionTabe1) <- c("For","Against")
colnames(confusionTabe1) <- c("For","Against")
prop.table(confusionTabe1)
```

```
 predcn1
                    For            Against
 For         0.79571429       0.01714286
 Against   0.10714286       0.08000000

 predcn2
                     For              Against
 For           0.78333333       0.01666667
 Against     0.11333333       0.08666667
```

In the Training Data set,
Error in Against -> 10.71%
Error in For -> 1.71%

In the Test Data set,
Error in Against -> 11.33%
Error in For -> 1.67%

(4) Run a logistic regression model of PREFERENCE on the demographic variables. Use only the training data set for this.

a. Present the output as **Exhibit B**.

```
log_train <- glm(as.numeric(PREFERENCE) ~ AGE + GENDER + INCOME, data = df_train, family =
'binomial')
summary(log_train)
```

Exhibit B

```
Call:
glm(formula = as.numeric(PREFERENCE) ~ AGE + GENDER + INCOME,
    family = "binomial", data = df_train)

Deviance Residuals:
    Min       1Q   Median       3Q
-2.65726  -0.37215  -0.16886  -0.04293
    Max
 2.76761

Coefficients:
             Estimate Std. Error z value
(Intercept) -0.41788    0.76950  -0.543
AGE          0.22478    0.02331   9.642
GENDERM     -0.73941    0.27662  -2.673
INCOME      -0.11617    0.01129 -10.286
             Pr(>|z|)
(Intercept)  0.58710
AGE          < 2e-16 ***
GENDERM      0.00752 **
INCOME       < 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: 674.87  on 699  degrees of freedom
Residual deviance: 353.26  on 696  degrees of freedom
AIC: 361.26

Number of Fisher Scoring iterations: 7
```

b. Provide a precise interpretation of the coefficient of AGE.

With increase in Age there is on an average 22.47% increase in chance of the person voting Against the increase in sales tax, given that all other factors are constant.

c. Provide a precise interpretation of the coefficient of the gender variable.
With gender being Male, there is on an average 73.94% decrease in chance of the person voting Against the increase in sales tax, given that all other factors are constant.

d. Use a cutoff of 0.5 and do the classification. What proportion of predicted classes are in error (in the training and test data set)?

```
pred2 <- ifelse(log_train$fitted.values>0.5,1,0)
confusionTabeA <- table(df_train$PREFERENCE, pred2)
rownames(confusionTabeA) <- c("For","Against")
colnames(confusionTabeA) <- c("For","Against")
prop.table(confusionTabeA)

log_test <- glm(as.numeric(PREFERENCE) ~ AGE + GENDER + INCOME, data = df_test, family =
'binomial')
summary(log_test)

pred3 <- ifelse(log_test$fitted.values > 0.5, 1, 0)
confusionTabeB <- table(df_test$PREFERENCE, pred3)
rownames(confusionTabeB) <- c("For","Against")
colnames(confusionTabeB) <- c("For","Against")
prop.table(confusionTabeB)
```

```
      pred2
                For            Against
For         0.77857143   0.03428571
Against     0.08000000   0.10714286
      pred3
                For            Against
For         0.76              0.04
Against     0.07              0.13
```

In the Training Data set,
Error in Against -> 8%
Error in For -> 3.42%

In the Test Data set,
Error in Against -> 7%
Error in For -> 4%

e. Compare these error rates with those in question 3d (linear regression).
In general, the difference between the error rates of training data and test data is less for Logistic Regression compared to Linear Regression.

f. Compute the predicted probability for voting *against* the proposition for an individual who is a female, is 36 years old, and has an income $70,000.

```
data <- data.frame(AGE = 36, GENDER ="F", INCOME = 70)
pred4 <- predict(log_test, data, interval = "confidence", type = "response")
pred4
```

```
0.3495066
```

Hints: You may find the *R* **ifelse** function convenient for classification. Finally, the **predict** function that was used for regression will also work for the logistic case. Note however that, by default, it will give you the predicted logit. If you pass it an additional argument (type = "response") you will get predicted probabilities. E.g.

p <- predict(fit, newdata=df, type = "response")