ST447: SUMMATIVE PROJECT 2021-22

Candidate Number 39719

December 4th 2021

INTRODUCTION :

XYZ has been learning to drive for a while and is considering taking the practical automobile test in the United Kingdom.There are two viable options:

1. Take the practical test at the nearest test centre to his or her residence

2. Take it at the LSE's nearest exam centre, i.e. Wood Green

PROFILE GENERATION :

The profile of XYZ :

Age: 21 | Gender: Male | Home address: Tolworth (London)

DATA PREPARATION :

The Data was extracted for both locations, i.e. Tolworth and Wood Green for a 7 year period. The data for the past 7 years only has been used mainly because there has been a significant change in modern automobiles in terms of driver and passenger safety features that have transformed the way modern vehicles are driven (increased number of sensors to assist drivers and better external cameras to see surroundings in clarity). Hence it made sense to use only the data from recent years for our analysis.

The data preparation was done in Excel entirely and the following transformations have been followed to manipulate the data for fitting the model

Gender	Value Specified
Male	1
Female	0

Outcome	Value Specified
Pass	1
Fail	0

Location	Value Specified
Tolworth	1
Wood Green	0

CREATING A DATAFRAME :

```
#FIRST LOAD THE DATA INTO A VARIABLE
combined_data = read.csv("CombinedData.csv", header = TRUE)

#SEE THE EXTRACTED DATA AND ITS STRUCTURE
head(combined_data)
```

```
##   SNO. YEAR AGE OUTCOME AGECAT GENDER LOC
## 1    1 2020  17       1      1      1   1
## 2    2 2020  17       1      1      1   1
## 3    3 2020  17       1      1      1   1
## 4    4 2020  17       1      1      1   1
## 5    5 2020  17       1      1      1   1
## 6    6 2020  17       1      1      1   1
```

```
str(combined_data)
```

```
## 'data.frame':    57687 obs. of  7 variables:
##  $ SNO.   : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ YEAR   : int  2020 2020 2020 2020 2020 2020 2020 2020 2020 2020 ...
##  $ AGE    : int  17 17 17 17 17 17 17 17 17 17 ...
##  $ OUTCOME: int  1 1 1 1 1 1 1 1 1 1 ...
##  $ AGECAT : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ GENDER : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ LOC    : int  1 1 1 1 1 1 1 1 1 1 ...
```

```
#CHECK FOR MISSINNG AND NA VALUES
```

DATA VISUALIZATION

```
# INDEXING DATA FOR VISUALIZATION

#First, lets extract data for 21 year old males in Tolworth and create a data frame from it
criterion1 =(combined_data$LOC == 1) & (combined_data$GENDER ==1) & (combined_data$AGE == 21)
df_c1 = data.frame(combined_data[criterion1,])
```

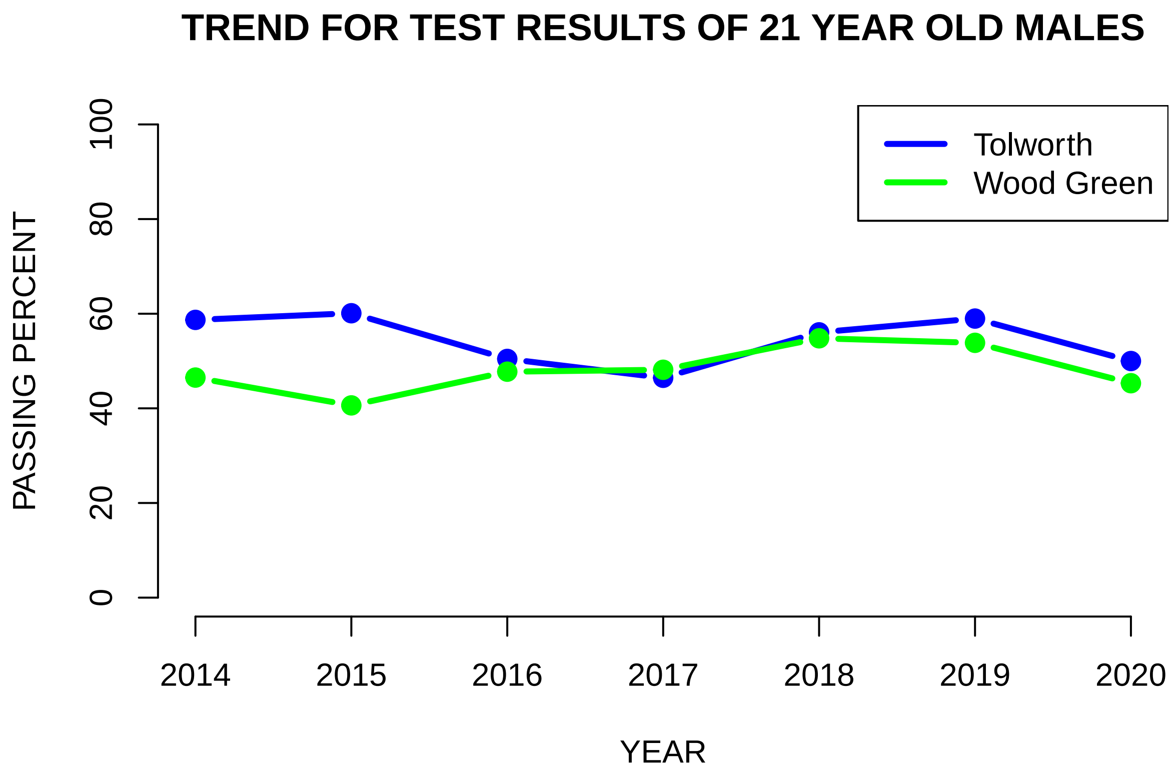
```r
#Now lets create a new data frame of mean passing rates in Tolworth
mean_tolworth = data.frame(aggregate(df_c1$OUTCOME, list(df_c1$YEAR), FUN=mean))
colnames(mean_tolworth)<- c("Year", "Pass Percentage")

#Lets convert this mean value to a percent value
mean_tolworth$`Pass Percentage` = mean_tolworth$`Pass Percentage`*100
head(mean_tolworth)
```

```
##   Year Pass Percentage
## 1 2014        58.69565
## 2 2015        60.09852
## 3 2016        50.44643
## 4 2017        46.45669
## 5 2018        56.06061
## 6 2019        58.97436
```

```r
#Next, we index data for 21 year old males in Wood Green and create its data frame
criterion2 =(combined_data$LOC == 0) & (combined_data$GENDER ==1) & (combined_data$AGE == 21)
df_c2 = data.frame(combined_data[criterion2,])

#Again, we create a new data frame of mean passing rates in Wood Green
mean_woodgreen = data.frame(aggregate(df_c2$OUTCOME, list(df_c2$YEAR), FUN=mean))
colnames(mean_woodgreen)<- c("Year", "Pass Percentage")
mean_woodgreen$`Pass Percentage` = mean_woodgreen$`Pass Percentage`*100
head(mean_woodgreen)
```

```
##   Year Pass Percentage
## 1 2014        46.50206
## 2 2015        40.62500
## 3 2016        47.74775
## 4 2017        48.14815
## 5 2018        54.80226
## 6 2019        53.84615
```

```r
#Finally, lets plot this on the graph and see any trends
plot(mean_tolworth, xlab = "YEAR", ylab = "PASSING PERCENT", col = "blue",
     type = "b" , main = "TREND FOR TEST RESULTS OF 21 YEAR OLD MALES", lwd =3,
     bty = "n", ylim = c(0,100), pch = 19 )

#Lets add Wood Green data on this and add a legend
lines(mean_woodgreen, col = "green", lwd = 3, type = "b", pch = 19)

#Lastly, we add a legend to our plot
legend(x = "topright", legend = c("Tolworth","Wood Green"), col = c("blue","green"), lwd = 3)
```



MODIFYING THE DATA :

```
#We need to convert some of our data points to factors before we model them

combined_data$OUTCOME <- as.factor(combined_data$OUTCOME) #To be predicted, dependent variable

combined_data$AGE <- as.factor(combined_data$AGE) # Since we will be performing a logistic regression,

#Check the Structure again to confirm the changes
str(combined_data)
```

STATISTICAL METHOD USED : LOGISTIC REGRESSION

```
## 'data.frame':    57687 obs. of  7 variables:
##  $ SNO.   : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ YEAR   : int  2020 2020 2020 2020 2020 2020 2020 2020 2020 2020 ...
##  $ AGE    : Factor w/ 9 levels "17","18","19",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ OUTCOME: Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
##  $ AGECAT : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ GENDER : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ LOC    : int  1 1 1 1 1 1 1 1 1 1 ...
```

NOTE: Gender and Location were not converted to factors since they are already in binary form and changing them to factor will not effect the model coefficients

MODELLING THE DATA :

```
#We use the Generalized Linear Model function in R to do the regression on the combined data.
CO_MODEL = glm(OUTCOME ~ AGE + GENDER + LOC ,data = combined_data, family = binomial(link = logit))

#Now lets see the model results:
summary(CO_MODEL)
```

STATISTICAL METHOD USED : LOGISTIC REGRESSION

```
##
## Call:
## glm(formula = OUTCOME ~ AGE + GENDER + LOC, family = binomial(link = logit),
##     data = combined_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3911  -1.1181  -0.9734   1.2067   1.4042
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.09789    0.02568  -3.811 0.000138 ***
## AGE18       -0.28335    0.02761 -10.262  < 2e-16 ***
## AGE19       -0.35571    0.03054 -11.646  < 2e-16 ***
## AGE20       -0.40291    0.03298 -12.216  < 2e-16 ***
## AGE21       -0.34595    0.03398 -10.182  < 2e-16 ***
## AGE22       -0.40108    0.03465 -11.576  < 2e-16 ***
## AGE23       -0.38658    0.03633 -10.640  < 2e-16 ***
## AGE24       -0.34549    0.03826  -9.029  < 2e-16 ***
## AGE25       -0.42101    0.04000 -10.526  < 2e-16 ***
## GENDER       0.27479    0.01685  16.304  < 2e-16 ***
## LOC          0.31261    0.01782  17.546  < 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: 79899  on 57686  degrees of freedom
## Residual deviance: 78748  on 57676  degrees of freedom
## AIC: 78770
##
## Number of Fisher Scoring iterations: 4
```

```
#We store the predicted values in a vector
R = predict(CO_MODEL, newdata = combined_data, type = "response")

#Lets take a look at the head of our predicted values
head(R)
```

NOW LETS PREDICT VALUES FOR OUR DATASET USING OUR OWN MODEL

```
##         1         2         3         4         5         6
## 0.6199897 0.6199897 0.6199897 0.6199897 0.6199897 0.6199897
```

```r
#Now lets round up these values to compare it to our original model
R$converted.to.binary <- ifelse(R >= 0.5, 1, 0)

#Lets have one final look at our predicted values
head(R$converted.to.binary)
```

```
## 1 2 3 4 5 6
## 1 1 1 1 1 1
```

MODEL ACCURACY :

```r
#Lets calculate the correct predictions that were right and take the mean of all observations to see th

accuracy <- mean((combined_data$OUTCOME) == (R$converted.to.binary))
print(accuracy)
```

```
## [1] 0.5622237
```

Hence, our model has an accuracy of 56.2223725% !

EVALUATING BOTH THE OPTIONS :

```r
#TEST 1 - SUCCESS RATE FOR TOLWORTH DRIVING CENTER -

friend = data.frame(AGE = "21", GENDER = 1, LOC = 1)
predicted_value_tolworth = predict(CO_MODEL, friend, type = "response")
print(predicted_value_tolworth)
```

```
##         1
## 0.5358268
```

```r
#TEST 2 - SUCCESS RATE FOR WOOD GREEN DRIVING CENTER -

my_guy = data.frame(AGE = "21", GENDER = 1, LOC = 0)
predicted_value_woodgreen = predict(CO_MODEL, my_guy, type = "response")
print(predicted_value_woodgreen)
```

Hence, our model predicts a *53.5826848 %* chance of success at Tolworth!

```
##         1
## 0.4578371
```

Hence, our model predicts a *45.7837069 %* chance of success at Wood Green!