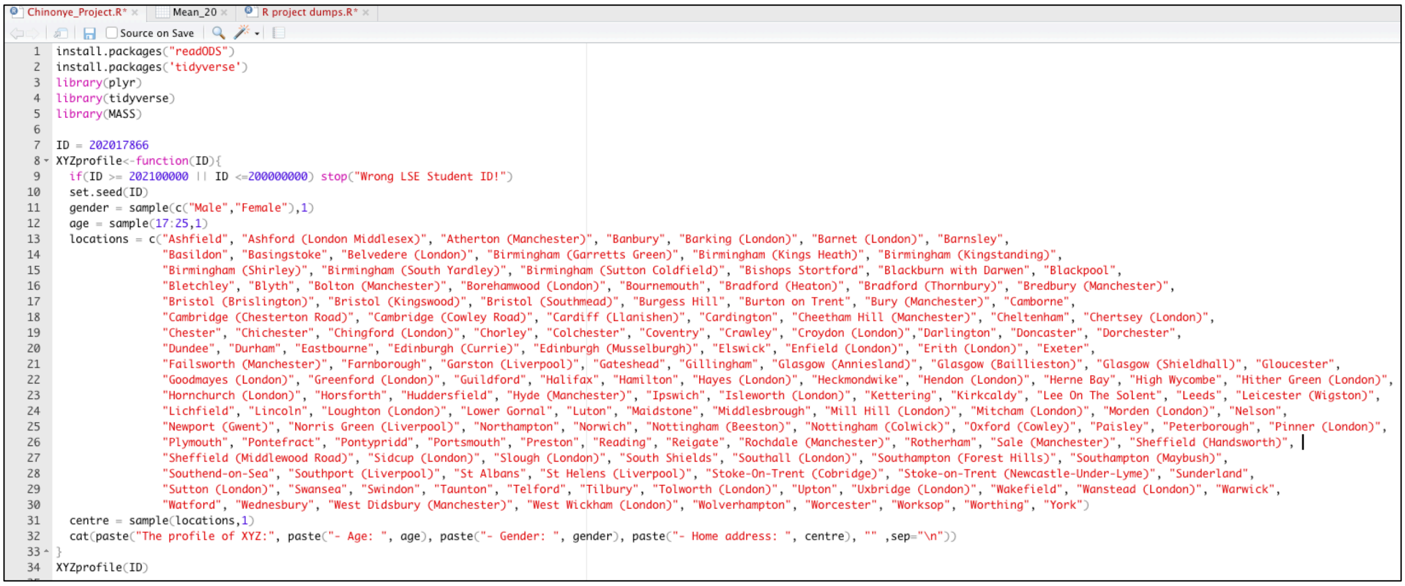
Project to forecast a student’s probability of success at the UK driving test by testing center

Problem Statement and Project Goal:

The goal of this project is to make a statistically backed suggestion to a friend, XYZ, on which driving center he/she should take their driving test. The two options for the driving centers where XYZ could test are the center closest to London School of Economics (Wood Green Driving Centre), or the center closest to his/her home. Additionally, the analysis should determine XYZ's passing rate at both centers

XYZ profile:

Using a random seed function in R and setting the seed value to my unique student ID number, XYZ's profile was generated to be a 20-year-old woman from Colchester. Based on this information, the driving test data from only the Colchester and Wood Green centers were analyzed to make suggestions to XYZ as data from all other centers were irrelevant. The code used to generate XYZ's profile is outlined in *figure 1* below:

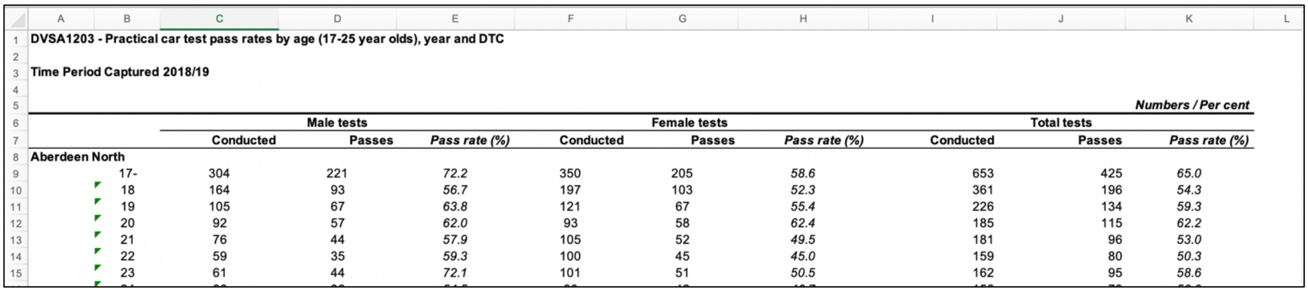


```r
install.packages("readODS")
install.packages('tidyverse')
library(plyr)
library(tidyverse)
library(MASS)

ID = 202017866
XYZprofile<-function(ID){
  if(ID >= 202100000 || ID <=200000000) stop("Wrong LSE Student ID!")
  set.seed(ID)
  gender = sample(c("Male","Female"),1)
  age = sample(17:25,1)
  locations = c("Ashfield", "Ashford (London Middlesex)", "Atherton (Manchester)", "Banbury", "Barking (London)", "Barnet (London)", "Barnsley",
                "Basildon", "Basingstoke", "Belvedere (London)", "Birmingham (Garretts Green)", "Birmingham (Kings Heath)", "Birmingham (Kingstanding)",
                "Birmingham (Shirley)", "Birmingham (South Yardley)", "Birmingham (Sutton Coldfield)", "Bishops Stortford", "Blackburn with Darwen", "Blackpool",
                "Bletchley", "Blyth", "Bolton (Manchester)", "Borehamwood (London)", "Bournemouth", "Bradford (Heaton)", "Bradford (Thornbury)", "Bredbury (Manchester)",
                "Bristol (Brislington)", "Bristol (Kingswood)", "Bristol (Southmead)", "Burgess Hill", "Burton on Trent", "Bury (Manchester)", "Camborne",
                "Cambridge (Chesterton Road)", "Cambridge (Cowley Road)", "Cardiff (Llanishen)", "Cardington", "Cheetham Hill (Manchester)", "Cheltenham", "Chertsey (London)",
                "Chester", "Chichester", "Chingford (London)", "Chorley", "Colchester", "Coventry", "Crawley", "Croydon (London)","Darlington", "Doncaster", "Dorchester",
                "Dundee", "Durham", "Eastbourne", "Edinburgh (Currie)", "Edinburgh (Musselburgh)", "Elswick", "Enfield (London)", "Erith (London)", "Exeter",
                "Failsworth (Manchester)", "Farnborough", "Garston (Liverpool)", "Gateshead", "Gillingham", "Glasgow (Anniesland)", "Glasgow (Baillieston)", "Glasgow (Shieldhall)", "Gloucester",
                "Goodmayes (London)", "Greenford (London)", "Guildford", "Halifax", "Hamilton", "Hayes (London)", "Heckmondwike", "Hendon (London)", "Herne Bay", "High Wycombe", "Hither Green (London)",
                "Hornchurch (London)", "Horsforth", "Huddersfield", "Hyde (Manchester)", "Ipswich", "Isleworth (London)", "Kettering", "Kirkcaldy", "Lee On The Solent", "Leeds", "Leicester (Wigston)",
                "Lichfield", "Lincoln", "Loughton (London)", "Lower Gornal", "Luton", "Maidstone", "Middlesbrough", "Mill Hill (London)", "Mitcham (London)", "Morden (London)", "Nelson",
                "Newport (Gwent)", "Norris Green (Liverpool)", "Northampton", "Norwich", "Nottingham (Beeston)", "Nottingham (Colwick)", "Oxford (Cowley)", "Paisley", "Peterborough", "Pinner (London)",
                "Plymouth", "Pontefract", "Pontypridd", "Portsmouth", "Preston", "Reading", "Reigate", "Rochdale (Manchester)", "Rotherham", "Sale (Manchester)", "Sheffield (Handsworth)",
                "Sheffield (Middlewood Road)", "Sidcup (London)", "Slough (London)", "South Shields", "Southall (London)", "Southampton (Forest Hills)", "Southampton (Maybush)",
                "Southend-on-Sea", "Southport (Liverpool)", "St Albans", "St Helens (Liverpool)", "Stoke-On-Trent (Cobridge)", "Stoke-on-Trent (Newcastle-Under-Lyme)", "Sunderland",
                "Sutton (London)", "Swansea", "Swindon", "Taunton", "Telford", "Tilbury", "Tolworth (London)", "Upton", "Uxbridge (London)", "Wakefield", "Wanstead (London)", "Warwick",
                "Watford", "Wednesbury", "West Didsbury (Manchester)", "West Wickham (London)", "Wolverhampton", "Worcester", "Worksop", "Worthing", "York")
  centre = sample(locations,1)
  cat(paste("The profile of XYZ:", paste("- Age: ", age), paste("- Gender: ", gender), paste("- Home address: ", centre), "" ,sep="\n"))
}
XYZprofile(ID)
```

Figure 1: Code to generate XYZ's profile

The Data:

Data collected by the Driver and Vehicle Standards Agency (DVSA) from 2007 to 2019 which lists the car pass rates of all test takers by gender and age (17-25) across all centers in the UK was used.



DVSA1203 - Practical car test pass rates by age (17-25 year olds), year and DTC

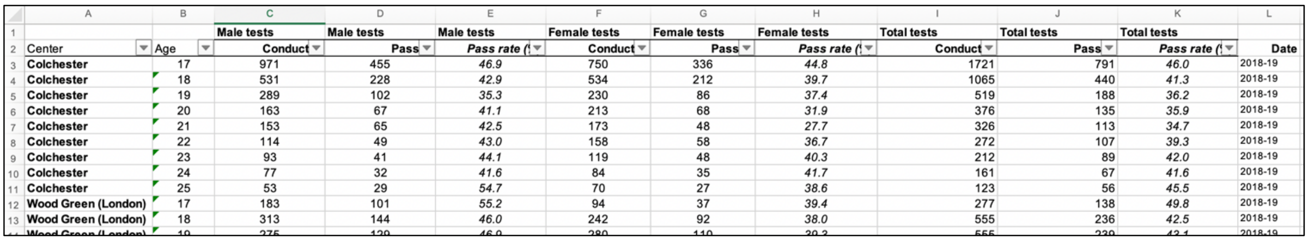
Time Period Captured 2018/19

Numbers / Per cent

	Male tests			Female tests			Total tests		
	Conducted	Passes	*Pass rate (%)*	Conducted	Passes	*Pass rate (%)*	Conducted	Passes	*Pass rate (%)*
Aberdeen North									
17-	304	221	72.2	350	205	58.6	653	425	65.0
18	164	93	56.7	197	103	52.3	361	196	54.3
19	105	67	63.8	121	67	55.4	226	134	59.3
20	92	57	62.0	93	58	62.4	185	115	62.2
21	76	44	57.9	105	52	49.5	181	96	53.0
22	59	35	59.3	100	45	45.0	159	80	50.3
23	61	44	72.1	101	51	50.5	162	95	58.6

Figure 2: Snippet of Raw data in original format

The data was stored in an ODS file and was split into tabs according to test year from 2007 to 2018. Partial data cleaning was performed outside of R in the ODS file to remove data from irrelevant test centers. The final format before loading the data into R is shown below:



		Male tests	Male tests	Male tests	Female tests	Female tests	Female tests	Total tests	Total tests	Total tests	
Center	Age	Conduct	Pass	*Pass rate (*	Conduct	Pass	*Pass rate (*	Conduct	Pass	*Pass rate (*	Date
Colchester	17	971	455	*46.9*	750	336	*44.8*	1721	791	*46.0*	2018-19
Colchester	18	531	228	*42.9*	534	212	*39.7*	1065	440	*41.3*	2018-19
Colchester	19	289	102	*35.3*	230	86	*37.4*	519	188	*36.2*	2018-19
Colchester	20	163	67	*41.1*	213	68	*31.9*	376	135	*35.9*	2018-19
Colchester	21	153	65	*42.5*	173	48	*27.7*	326	113	*34.7*	2018-19
Colchester	22	114	49	*43.0*	158	58	*36.7*	272	107	*39.3*	2018-19
Colchester	23	93	41	*44.1*	119	48	*40.3*	212	89	*42.0*	2018-19
Colchester	24	77	32	*41.6*	84	35	*41.7*	161	67	*41.6*	2018-19
Colchester	25	53	29	*54.7*	70	27	*38.6*	123	56	*45.5*	2018-19
Wood Green (London)	17	183	101	*55.2*	94	37	*39.4*	277	138	*49.8*	2018-19
Wood Green (London)	18	313	144	*46.0*	242	92	*38.0*	555	236	*42.5*	2018-19
Wood Green (London)	19	275	129	*46.9*	280	110	*39.3*	555	239	*43.1*	2018-19

Figure 3: Snippet of data for relevant test centres in year 2018-2019

To carry out this analysis, the following assumptions were made:

1. The number of tests conducted are for unique test takers (i.e. there is no repetition)
2. The male and female data were independent of each other
3. The year-on-year data were independent of each other
4. The data within groups (e.g., pass rate data for women of different ages in either test center) were independent of each other

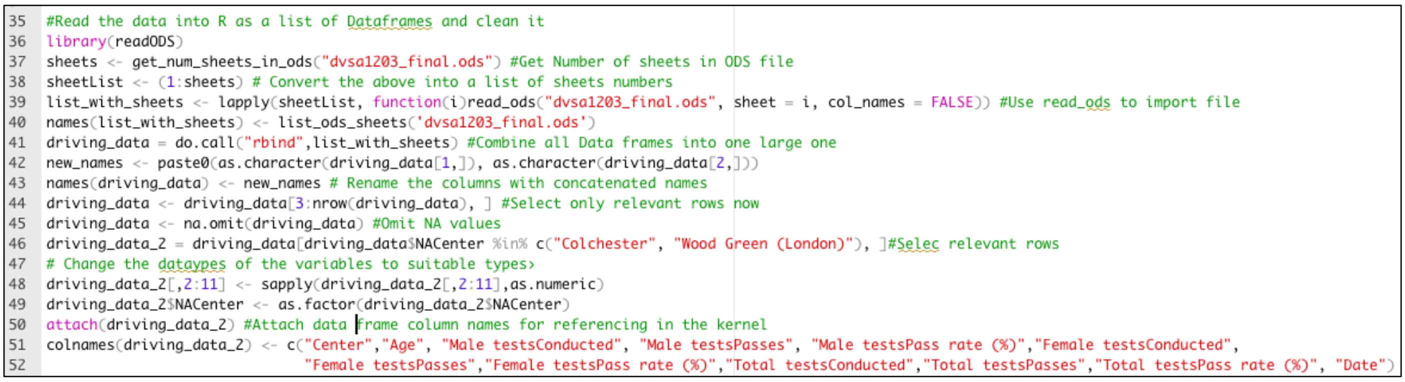
Summary:

This analysis was conducted on UK DVSA driving data from 2007 – 2019 with the goal of suggesting a driving center to a friend, XYZ, to take her test based on her features. In order to do this, the data was cleaned and prepared for analysis, and appropriate statistical methods i.e. Expected Value calculation and Logistic Regression were used to compare the pass rates and odds of passing at both centers. The results showed a higher pass rate at Wood Green and a higher odds of XYZ falling within the percentage of testers who pass at Wood Green. Therefore, the conclusion of this analysis was that XYZ should take her test at the Wood Green center.

Methodology:

Cleaning:

The data - as shown in fig. 2 - was loaded into R using the *read_ods* function embedded in a for loop to enable reading from each tab in the file. Further data preparation was performed to combine data from all tabs into a single data frame, rename columns, change variable types, remove rows with missing data and select relevant data. Data from all 12 years were used in this analysis to provide a large enough number of observations, and the year variable was not considered given the assumption stated above.



```
#Read the data into R as a list of Dataframes and clean it
library(readODS)
sheets <- get_num_sheets_in_ods("dvsa1203_final.ods") #Get Number of sheets in ODS file
sheetList <- (1:sheets) # Convert the above into a list of sheets numbers
list_with_sheets <- lapply(sheetList, function(i)read_ods("dvsa1203_final.ods", sheet = i, col_names = FALSE)) #Use read_ods to import file
names(list_with_sheets) <- list_ods_sheets('dvsa1203_final.ods')
driving_data = do.call("rbind",list_with_sheets) #Combine all Data frames into one large one
new_names <- paste0(as.character(driving_data[1,]), as.character(driving_data[2,]))
names(driving_data) <- new_names # Rename the columns with concatenated names
driving_data <- driving_data[3:nrow(driving_data), ] #Select only relevant rows now
driving_data <- na.omit(driving_data) #Omit NA values
driving_data_2 = driving_data[driving_data$NACenter %in% c("Colchester", "Wood Green (London)"), ]#Selec relevant rows
# Change the dataypes of the variables to suitable types›
driving_data_2[,2:11] <- sapply(driving_data_2[,2:11],as.numeric)
driving_data_2$NACenter <- as.factor(driving_data_2$NACenter)
attach(driving_data_2) #Attach data frame column names for referencing in the kernel
colnames(driving_data_2) <- c("Center","Age", "Male testsConducted", "Male testsPasses", "Male testsPass rate (%)","Female testsConducted",
                               "Female testsPasses","Female testsPass rate (%)","Total testsConducted","Total testsPasses","Total testsPass rate (%)", "Date")
```

Figure 4: Code for data cleaning and preparation

Exploratory Data Analysis (EDA):

An EDA was carried out using boxplots to visualize the performance of female test takers by age in both test centers. Male test takers were not considered in the EDA as our focus was on female test takers, and also given assumption 2.

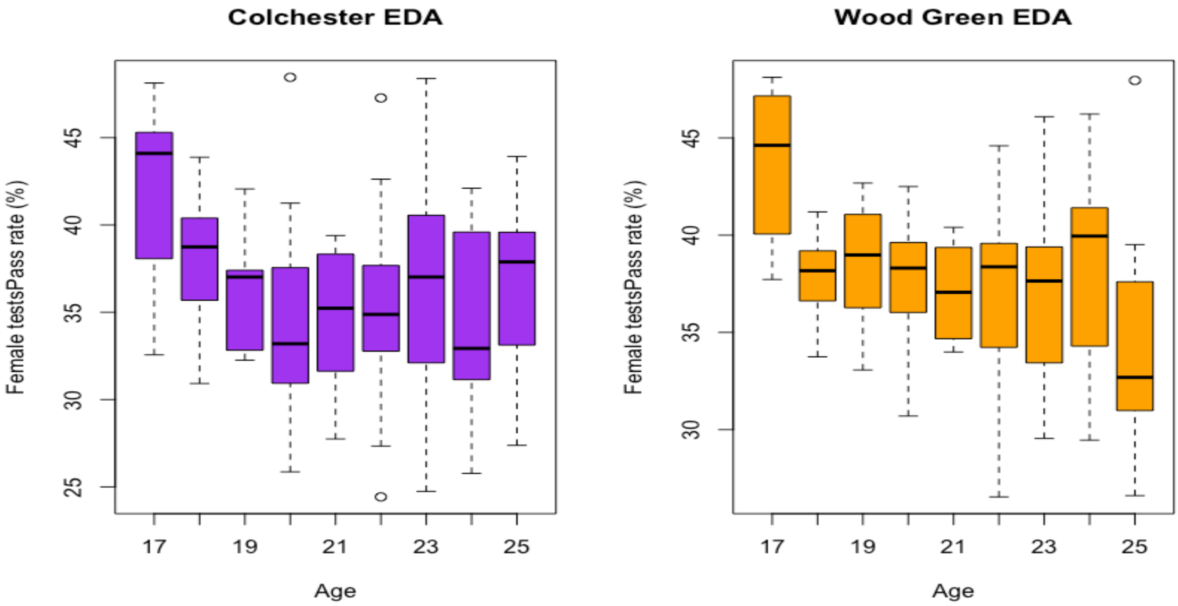


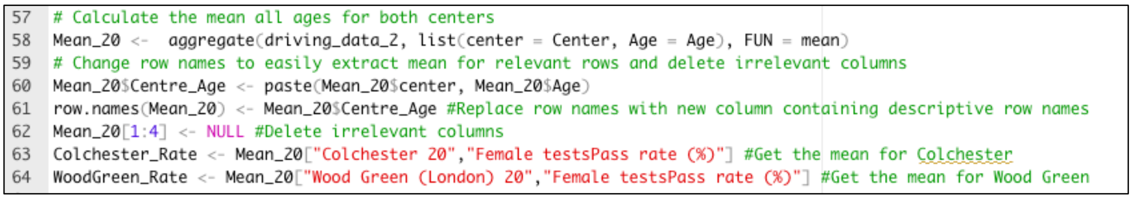
Figure 5: Boxplots of Colchester and Wood Green data by female pass rate and age

Figure 5 shows a slightly linear inverse relationship between age and pass rates. The plots also suggest that 20-year-old women at Wood Green do better than counterparts at Colchester as all values in the statistics are higher. A further in-depth data analysis was carried out below to examine these suggestions.

<u>Expected Value for Pass Rates at both centers:</u>

To obtain XYZ's expected passing rates at both centers, the 'expected value' equation *E(X) = ΣX * P(X)* was used. 'Pass rates' in the dataset indicate the *probability of a pass(P(X))* for each sample size e.g., the data says that of the 260 20-year-old women who tested at Wood Green in 2018-2019, 99 of them passed. 99/260 = 0.381 which was the probability of a pass at that center for that year. This value corresponds to the provided pass rate of 38.1%.

The expected value equation above is synonymous with the mean pass rates of 20-year-old females, thus XYZ's expected pass rates at both centers were obtained using the mean function in R shown in *figure 6* below to yield **37.649%** at Wood Green and **34.615%** at Colchester**.** This supports the suggestions from the EDA carried out above.



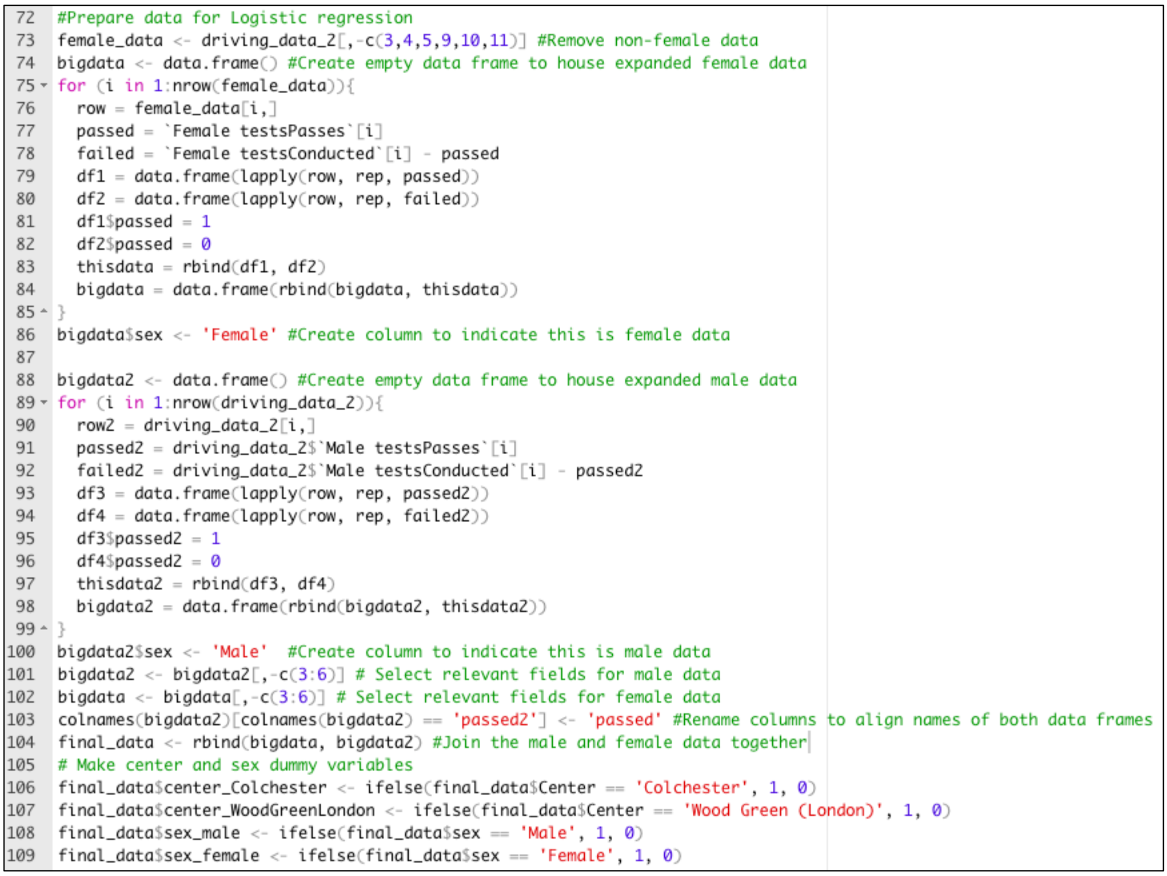
```
# Calculate the mean all ages for both centers
Mean_20 <-  aggregate(driving_data_2, list(center = Center, Age = Age), FUN = mean)
# Change row names to easily extract mean for relevant rows and delete irrelevant columns
Mean_20$Centre_Age <- paste(Mean_20$center, Mean_20$Age)
row.names(Mean_20) <- Mean_20$Centre_Age #Replace row names with new column containing descriptive row names
Mean_20[1:4] <- NULL #Delete irrelevant columns
Colchester_Rate <- Mean_20["Colchester 20","Female testsPass rate (%)"] #Get the mean for Colchester
WoodGreen_Rate <- Mean_20["Wood Green (London) 20","Female testsPass rate (%)"] #Get the mean for Wood Green
```

Figure 6: Code used to calculate Expected Value for 20 y/o females

The Expected Values provide a statistic for a target a sample of a population (in this case only 20-year-old females), but it does not examine the relationship between or likelihood of XYZ passing a test based on her features. To make a stronger case, a statistical method that can be used to examine trend and make inference for a population was required to buttress the results obtained from the Expected Value, hence the use of logistic regression below.

Logistic Regression:

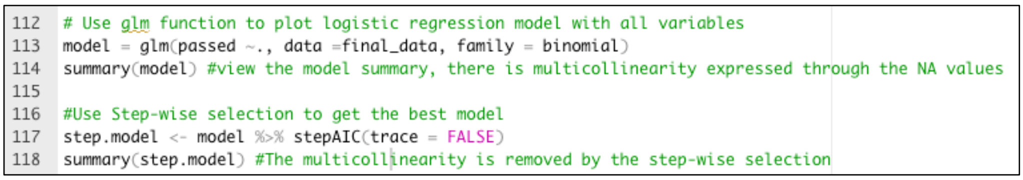
Logistic regression (logit) was used to analyze this data due to the binary nature of the dependent variable (i.e. pass or fail) and the highly categorical nature of the data. Logistic regression is a model that allows for a dependent variable to be transformed into its corresponding probabilities between 0 and 1 through the use of a *sigmoid function*. In this analysis the model took all predictor variables into account to produce an output of 1 or 0 and the *sigmoid function* was used to convert those 1 and 0 values into probability values. Several data cleaning and preparation methods shown in *figure 7* below were used to prepare the data frame '*final_data'* on which the logistic regression was carried out.
To fit the model, all predictor variables i.e. sex, center and age were chosen.
The dependent variable chosen was 'Passed' (represented by 1 for pass and 0 for fail). The data was expanded to produce 1 row for each test taker and assign a pass or fail label to them based on the pass rate. E.g., if there was a 20% pass rate for 100 test takers, 100 rows would be created with 20 testers with a pass and 80 with a fail. The variables 'Center' and 'Sex' were split into dummy variables (i.e., a row of 0 for false and 1 for true) to enable the model coefficients and outputs to be more readable.



```r
#Prepare data for Logistic regression
female_data <- driving_data_2[,-c(3,4,5,9,10,11)] #Remove non-female data
bigdata <- data.frame() #Create empty data frame to house expanded female data
for (i in 1:nrow(female_data)){
  row = female_data[i,]
  passed = `Female testsPasses`[i]
  failed = `Female testsConducted`[i] - passed
  df1 = data.frame(lapply(row, rep, passed))
  df2 = data.frame(lapply(row, rep, failed))
  df1$passed = 1
  df2$passed = 0
  thisdata = rbind(df1, df2)
  bigdata = data.frame(rbind(bigdata, thisdata))
}
bigdata$sex <- 'Female' #Create column to indicate this is female data

bigdata2 <- data.frame() #Create empty data frame to house expanded male data
for (i in 1:nrow(driving_data_2)){
  row2 = driving_data_2[i,]
  passed2 = driving_data_2$`Male testsPasses`[i]
  failed2 = driving_data_2$`Male testsConducted`[i] - passed2
  df3 = data.frame(lapply(row, rep, passed2))
  df4 = data.frame(lapply(row, rep, failed2))
  df3$passed2 = 1
  df4$passed2 = 0
  thisdata2 = rbind(df3, df4)
  bigdata2 = data.frame(rbind(bigdata2, thisdata2))
}
bigdata2$sex <- 'Male'  #Create column to indicate this is male data
bigdata2 <- bigdata2[,-c(3:6)] # Select relevant fields for male data
bigdata <- bigdata[,-c(3:6)] # Select relevant fields for female data
colnames(bigdata2)[colnames(bigdata2) == 'passed2'] <- 'passed' #Rename columns to align names of both data frames
final_data <- rbind(bigdata, bigdata2) #Join the male and female data together
# Make center and sex dummy variables
final_data$center_Colchester <- ifelse(final_data$Center == 'Colchester', 1, 0)
final_data$center_WoodGreenLondon <- ifelse(final_data$Center == 'Wood Green (London)', 1, 0)
final_data$sex_male <- ifelse(final_data$sex == 'Male', 1, 0)
final_data$sex_female <- ifelse(final_data$sex == 'Female', 1, 0)
```

Figure 7: Code used to prepare male and female data for logistic regression

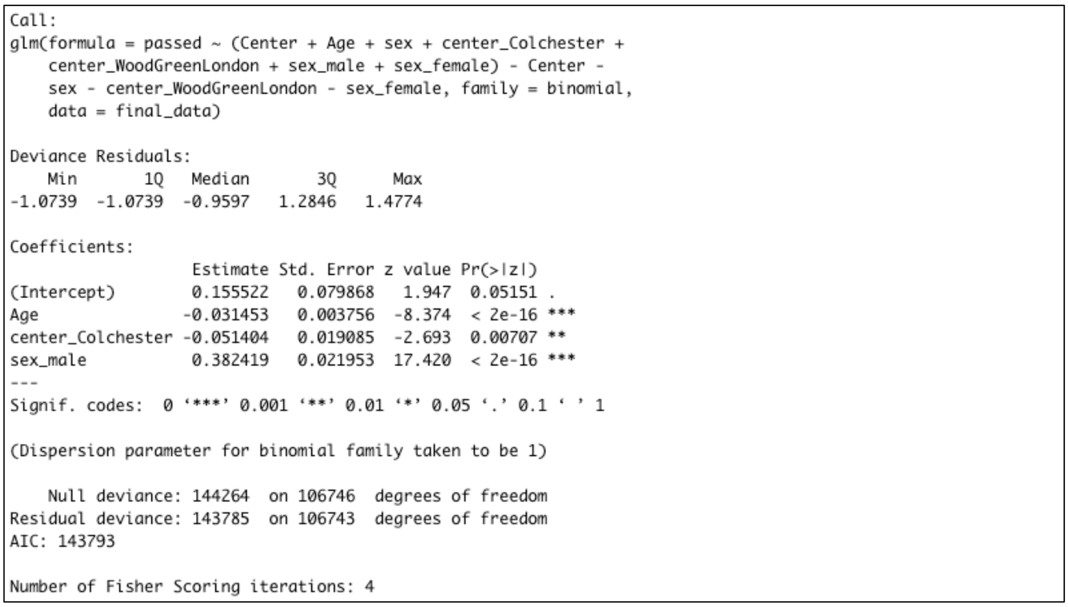
To get the best possible fitted model, the S*tepwise Selection method* was used after the model was fitted with all the predictor variables. This method selects the best model by iteratively adding and removing predictor variables and selecting the model that best fits our data.



```
# Use glm function to plot logistic regression model with all variables
model = glm(passed ~., data =final_data, family = binomial)
summary(model) #view the model summary, there is multicollinearity expressed through the NA values

#Use Step-wise selection to get the best model
step.model <- model %>% stepAIC(trace = FALSE)
summary(step.model) #The multicollinearity is removed by the step-wise selection
```

Figure 8: Code used to fit the logistic regression model and use stepwise selection

The model summary for the stepwise logistic regression model yielded the following output:

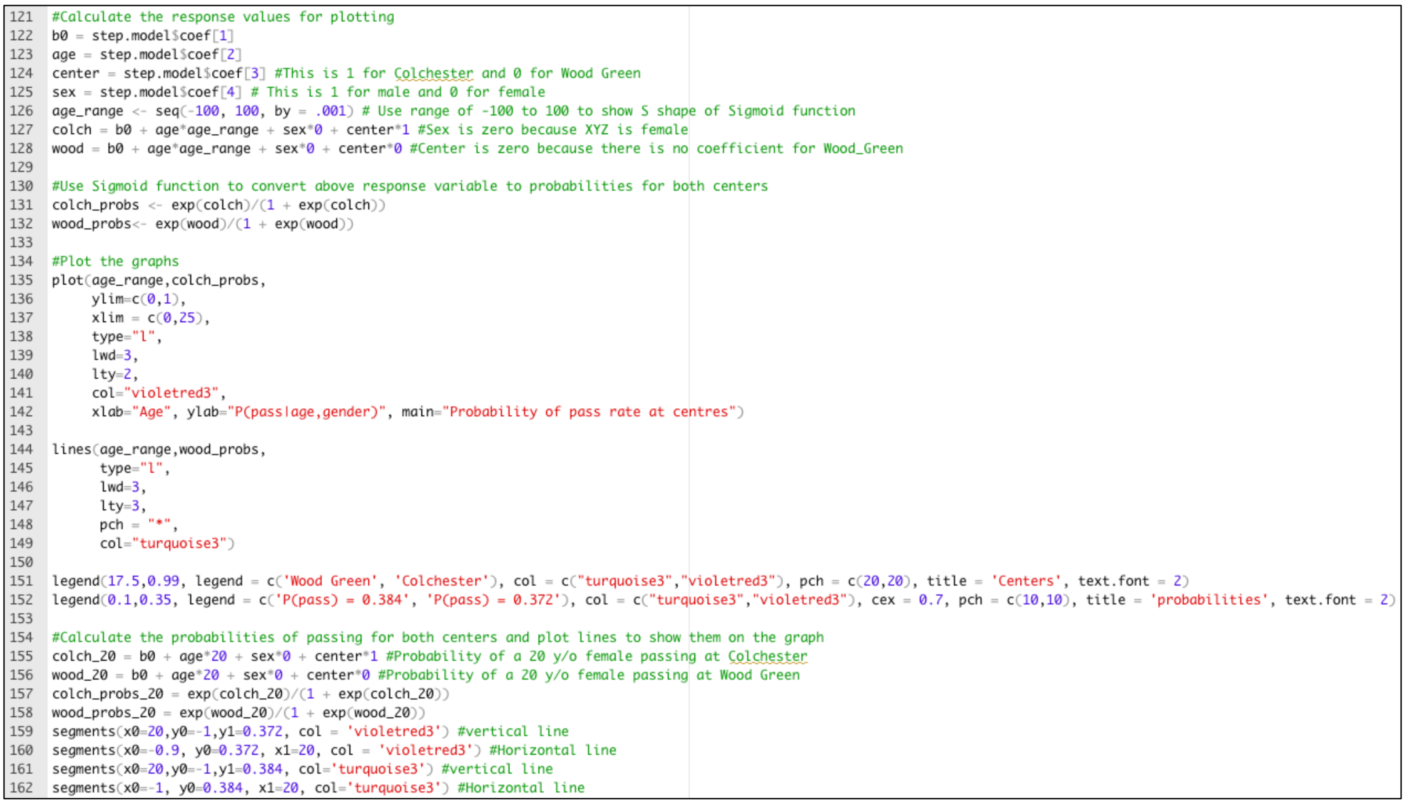


```
Call:
glm(formula = passed ~ (Center + Age + sex + center_Colchester +
    center_WoodGreenLondon + sex_male + sex_female) - Center -
    sex - center_WoodGreenLondon - sex_female, family = binomial,
    data = final_data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.0739  -1.0739  -0.9597   1.2846   1.4774

Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)        0.155522   0.079868   1.947  0.05151 .
Age               -0.031453   0.003756  -8.374  < 2e-16 ***
center_Colchester -0.051404   0.019085  -2.693  0.00707 **
sex_male           0.382419   0.021953  17.420  < 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: 144264  on 106746  degrees of freedom
Residual deviance: 143785  on 106743  degrees of freedom
AIC: 143793

Number of Fisher Scoring iterations: 4
```

Figure 9: Logistic Regression Step-wise Model Summary

The coefficient estimates in the first column show the relationship (in terms of direction and magnitude) between the dependent variable and each of the independent variables. The negative coefficient for age is in line with the already established trend observed earlier that pass rates decrease as age increases. The negative coefficient for Colchester shows that there is a lesser probability of passing if the test is taking at Colchester which also corresponds to previously observed trends. The positive coefficient for Males indicates that men are more likely to pass than women at an older age. It also infers that men generally have higher pass rates than women and since our analysis is only for women, this coefficient would not contribute to the pass rate for females since logically, it would be multiplied by 0 as the opposite of male (1) is female (0). The ‘Wood Green’ and Female’ variables were not included in the output because they share multicollinearity with the ‘Colchester’ and ‘Male’ variables. This means one variable can be predicted from the others (if one variable is 1, the other has to be 0 so if a person is not male, she must be female). All 3 variables had p values smaller than 0.05 meaning that the null hypothesis that suggests there is no relationship between the independent variable and the dependent variable (in this case, ‘pass’ or ‘fail’) could be rejected. Simply put, it means that historically, all variables (sex, test center and age) are significant to the probability of a student passing or failing at either center.

To plot the logit graphs and visualize XYZ’s chances of passing at either center, the *sigmoid function* equation was used to derive the conditional probabilities of each outcome (pass or fail) from the dependent variable as shown in *figure 10* below.



```r
#Calculate the response values for plotting
b0 = step.model$coef[1]
age = step.model$coef[2]
center = step.model$coef[3] #This is 1 for Colchester and 0 for Wood Green
sex = step.model$coef[4] # This is 1 for male and 0 for female
age_range <- seq(-100, 100, by = .001) # Use range of -100 to 100 to show S shape of Sigmoid function
colch = b0 + age*age_range + sex*0 + center*1 #Sex is zero because XYZ is female
wood = b0 + age*age_range + sex*0 + center*0 #Center is zero because there is no coefficient for Wood_Green

#Use Sigmoid function to convert above response variable to probabilities for both centers
colch_probs <- exp(colch)/(1 + exp(colch))
wood_probs<- exp(wood)/(1 + exp(wood))

#Plot the graphs
plot(age_range,colch_probs,
     ylim=c(0,1),
     xlim = c(0,25),
     type="l",
     lwd=3,
     lty=2,
     col="violetred3",
     xlab="Age", ylab="P(pass|age,gender)", main="Probability of pass rate at centres")

lines(age_range,wood_probs,
      type="l",
      lwd=3,
      lty=3,
      pch = "*",
      col="turquoise3")

legend(17.5,0.99, legend = c('Wood Green', 'Colchester'), col = c("turquoise3","violetred3"), pch = c(20,20), title = 'Centers', text.font = 2)
legend(0.1,0.35, legend = c('P(pass) = 0.384', 'P(pass) = 0.372'), col = c("turquoise3","violetred3"), cex = 0.7, pch = c(10,10), title = 'probabilities', text.font = 2)

#Calculate the probabilities of passing for both centers and plot lines to show them on the graph
colch_20 = b0 + age*20 + sex*0 + center*1 #Probability of a 20 y/o female passing at Colchester
wood_20 = b0 + age*20 + sex*0 + center*0 #Probability of a 20 y/o female passing at Wood Green
colch_probs_20 = exp(colch_20)/(1 + exp(colch_20))
wood_probs_20 = exp(wood_20)/(1 + exp(wood_20))
segments(x0=20,y0=-1,y1=0.372, col = 'violetred3') #vertical line
segments(x0=-0.9, y0=0.372, x1=20, col = 'violetred3') #Horizontal line
segments(x0=20,y0=-1,y1=0.384, col='turquoise3') #vertical line
segments(x0=-1, y0=0.384, x1=20, col='turquoise3') #Horizontal line
```

Figure 10: Code to calculate and plot logit graphs

This was done for both the Wood Green and Colchester centers and the outcomes were plotted against age. The *sigmoid* graph gives an S shaped curve that is bound between 0 (fail) and 1(pass). Anything above 0.5 is a pass and below it is a fail.

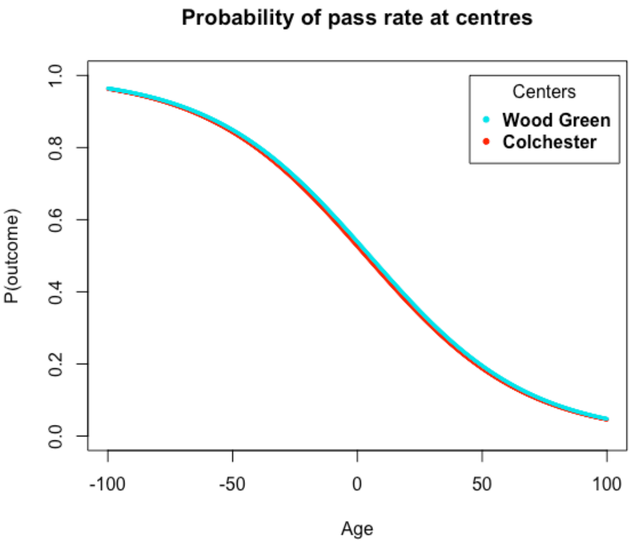


Figure 11: Showing S-Shaped curve for a wide age range.

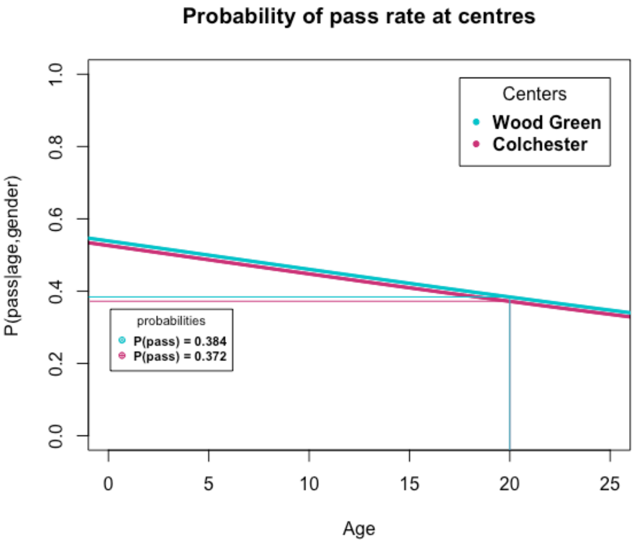


Figure 12: Zoomed in Section of ages 0-25 from fig. 4

The graph in *figure 11* was plotted with age range -100 to 100 to show that the *sigmoid function* produced the expected S-curve. However, the results we are interested in lie in a much smaller range of 17 – 25. *Figure 12* shows the plots for 0 – 25 years old taken from *figure 11*. There is a marginal but present difference between pass rates at Wood Green and Colchester for all ages, with Wood Green having the slightly higher rate.

Results and Conclusion

The analysis concludes that XYZ has a 0.384 probability of falling within the 37 % of test takers that pass the test at Wood Green and a 0.372 probability of falling within the 34% of females that pass the test at Colchester, given her features of being a 20-year-old woman. Therefore, while the overall chances of her passing at all are sadly quite low in both centers, she should take the test at Wood Green and take it soon as possible, because the older she gets the lower her chances of passing are.

Strengths and Weaknesses of the Data and Methodology:

The data used for this analysis was sparse and gave no distinctive details of testers that could have been used in model fitting or prediction. Variables specific to the individual test takers such as familiarity of the routes, time spent learning to drive for individual test takers etc. would have been useful to build a more accurate model to depict the probability of XYZ passing based on features she shared with other test takers. There is also no proof as to whether one test taker took the test and failed multiple times, which could possibly skew the data as the best analysis would be done for 1 record per individual. The strength of the data was its large number of observations compared to its variables (p) which allows for a lower chance of overfitting thus eliminating the need to apply regularization techniques to our model. The data had no missing values and it was an organized and simple dataset. Logit was a good model for this analysis due to the binary nature of the outcome i.e., 'pass' or 'fail' and the linearity between the pass rates and the independent variables. Additionally, logit allows us to easily compare where XYZ has higher chances of passing as the model outputs probabilities. This allowed for easy comparison as we plotted the logit function for both centers and advised XYZ based on the higher of the two. Logit is suitable for data of all distribution types and thus was suitable for our data which was not normally distributed. Logit is an easy model to train, implement and interpret. The weakness of the logit model is that it does not do well with high multicollinearity which was present in our data given the dummy variables that were created (i.e., new columns were created for sex and center and filled with ones and zeros). However, the colinear variables were ultimately removed in the stepwise selection process making logit a good model to use. It also assumed linearity between dependent and independent variables and the data provided, while somewhat linear, did not have a perfectly linear fit.

Possible Improvements:

For the data, the DVSA could collect information for individual test takers. They could also collect information about their demographic in order to have a robust dataset that would lead to better data analysis. There could also be synthetic IDs placed beside each tester to allow for analysis of unique testers. Information about the centers and routes themselves would also have improved the analysis and allowed for more precise suggestions to give XYZ.

The analysis could have been improved by fitting the data to other models such as Mean Difference, Wald test etc. and comparing the accuracy of those models to the chosen logistic regression model. Through this, a model that more accurately fit our data could be used to advise XYZ. This could also have been done on this model itself by predicting the outcome of the data used to fit it and creating a confusion matrix to test its accuracy. However, as this model is a supplement to the Expected Value test and is used for inference rather than prediction, that check was not strictly required.