MA335 - Modelling experimental and observational data

Final Project

Student ID: 2201687

Date: 26/06/2023

Abstract:

This report is the result of a data analysis project regarding an investigation into various features and their connection to a subject's dementia diagnosis. Some brief exploratory data analysis highlighted important connections between some of the variables and the "Demented" and "Nondemented" groups; namely, cognitive test scores, brain volume, and to a smaller degree, gender and years in education. Cleaning and preparing the data meant the removal of missing values, removing unnecessary rows, and converting certain columns to numerical values, among other things. Following a section of visualising the data to gain a deeper understanding of the data set, how the variables interacted, and the distribution of certain variables, a clustering algorithm was applied. The purpose of this was to establish if there were any clear patterns in the data set, and to see if they could be used to predict a subject's dementia diagnosis. K-means clustering was the algorithm chosen for this task, and while the optimal number of clusters was easily found, the effectiveness of the clustering algorithm was found to be limited; it was scarcely better than random chance when it came to classifying observations or predicting dementia diagnosis. In addition, logistic regression models were built to most accurately predict dementia diagnosis. Initially, a model including all variables was built, but it was observed that CDR as a predictor is redundant due to its correlation with the Group (dependent variable). Feature selection methods, such as variable inflation factor (VIF), McFadden pseudo R2 values, and Boruta further enhance the model's accuracy, Akaike information criterion (AIC), and area under curve (AUC) scores. Overall, this report highlights the importance of specific variables in the diagnosis of dementia, the value in feature selection methods, and the limited ability of K-means clustering for this dataset.

Contents:

1. Title Page
2. Abstract and Contents
3. Introduction & Preliminary Analysis
4 - 7. Analysis & Discussion
8. Conclusion
9+. Bibliography & Appendix

Report size limit: 1800 words or 6 pages.

● Due to the limited size of this report, few figures or tables will be shown in the text directly. Any figures or tables referenced but not in the report text can be found in the appendix.

Introduction:

The purpose of this report is to investigate the relationship between several characteristics and the diagnosis – either "Demented" or "Nondemented" – of Alzheimer's disease. The characteristics that will be analysed are Group/Group_Num (Diagnosis), M.F (Gender), Age, EDUC (Year of education), SES (Socioeconomic Status; 1-5, 1-low, 5-high), MMSE (Mini mental state examination), CDR (Clinical dementia rating), eTIV (Estimated total intracranial volume), nWBV (Normalize whole brain volume), and ASF (Atlas scaling factor). In the completion of this report's objective, after data cleaning and preparing, graphical and numerical descriptive statistics will be used to gain a better understanding of the data set; this is followed by use of a clustering algorithm, namely K-Means clustering, to attempt to predict diagnoses and group variables. Finally, a logistic regression model, with applied feature selection methods, will be used to find the most important variables in predicting diagnoses.

Preliminary Analysis:

The preliminary analysis revolves mainly around the use of figures and tables to breakdown and display the dataset. Figure 1 shows the majority of subjects were not diagnosed with Alzheimer's – 60% "Nondemented" and 40% "Demented.

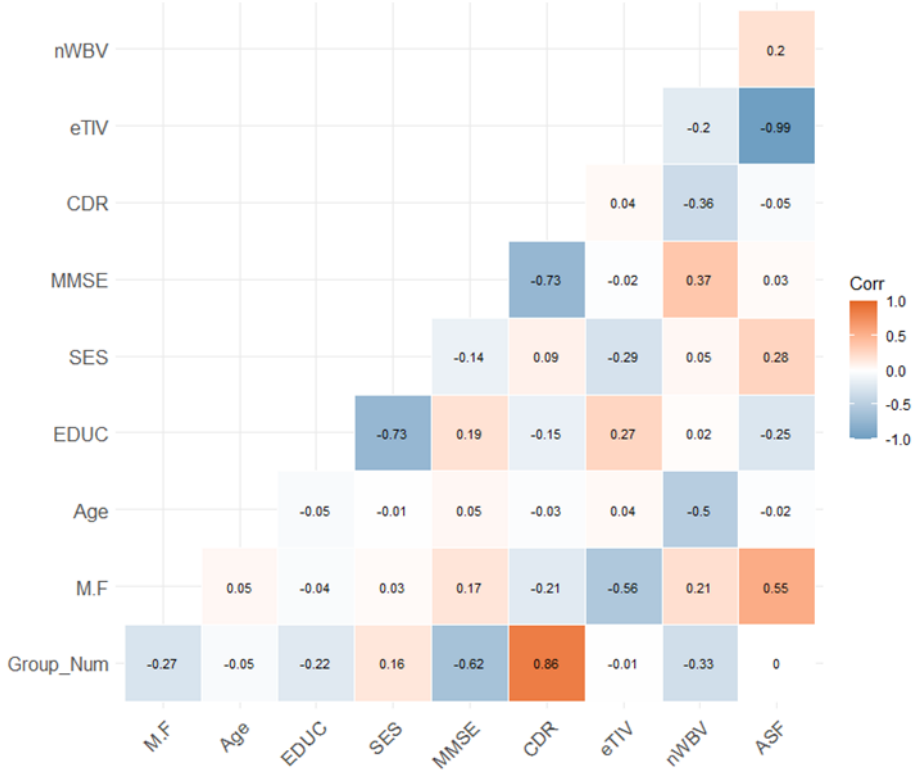

Figure 2: Correlation Matrix of All Variables

Figure 2 is incredibly useful as it allows a greater understanding of both how the predictors interact with the dependent variable, but also how they interact with each-other. Important aspects to highlight are the high correlation between CDR and the Group, and between ASF and eTIV; the latter due to ASF being used to estimate eTIV. The correlation between CDR and Group is further explored in Table 1, highlighting difficulties later faced when building a logistic regression model; as the population of Demented subjects and subjects with a CDR score above 0 is identical, apart from two anomalies. Additionally, Table 1 also looks at MMSE scores, which clearly closely align with diagnoses and CDR scores, suggesting it is a good predictor. Figure 2 also helps identify potential areas of multicollinearity between predictors, such as EDUC and SES.

Figures 3-8 show the density plots of all continuous variables. It can be seen that - excluding variables Age and nWBV - the remaining variables show a non-normal like distribution, although using the Shapiro-Wilk test shows that Age and nWBV are not exact normal distributions.

Studying Figure 9 it is obvious that Age has little baring on diagnosis – in-line with the low correlation - due to the very scattered datapoints for both Groups and the very limited differences between the box plots. Although, the average age of demented subjects is slightly lower. Figure 10 shows a “Demented” diagnosis is noticeably more likely for men, while the opposite is true for women. While Figure 11 shows EDUC and SES have some, but limited, influence on diagnosis, with EDUC seeming to have more influence than SES. Ultimately however, it is clear there are a good mix of both diagnosis Groups represented in every SES and EDUC range.

Analysis and Discussion:

Clustering Algorithm: K-Means Clustering

A K-Means clustering algorithm was implemented with the aim of seeing if the algorithm could successfully group the data points closely in-line with the ground truth, i.e., could the algorithm correctly predict a subject’s diagnosis by correctly clustering the Group variable. The first stage in K-Means clustering after data preparation is to find the optimal number of clusters. This was easily done and can be visualised in Figure 12: 3 clusters were chosen due to the reduction in the Total Within Sum of Squares (TWSS) rapidly falling off in efficiency after this point. Although there could be an argument made for using 5 clusters, due to TWSS reduction

efficiency flat lining after this point, early testing quickly showed this to be noticeably less efficient than 3 clusters.

Analysing the results of the clustering algorithm, it is obvious that K-Means clustering is an ineffectual method of clustering for this data set. Tables 2-4 show the breakdown of Group, CDR, and M.F between the clusters, while Table 5 shows the averages of the continuous variables between the clusters.

Cluster	Age	EDUC	MMSE	eTIV	nWBV	ASF
3	76.07971	14.44928	26.88406	1,489.109	0.7295072	1.179906
2	76.32039	13.78641	27.74757	1,307.194	0.7430874	1.346398
1	78.40789	16.03947	27.28947	1,754.289	0.7156447	1.003066

Table 5: Continuous Variable Averages Between Clusters

Table 5 shows that the averages of the continuous variables -apart from eTIV and ASF- change little between the clusters, highlighting the ineffectual nature of this algorithm at clustering different variables. Combined with the information shown in Tables 2-4 it is inferred there are no distinguishing differences beyond random chance.

In order to confirm the inferences from the above-mentioned tables several Adjusted Rand Index (ARI) tests were conducted. These confirmed the inferences with results of 7.725389e-05 for Group, -0.0072 for CDR, and 0.1611 for M.F. ARI tests give scores between 1 and -1, with 1 meaning perfect alignment with the ground truth of the dataset, while -1 is the opposite, a score of 0 means no better or worse than random chance. As demonstrated, the algorithm is roughly as accurate as random chance at assigning variables, and in some cases worse than chance.

Thus, it can be concluded that K-Means clustering is insufficient as a clustering algorithm for this dataset and cannot effectively assign a diagnosis based on the variables in the dataset.

Initial Logistic Regression:

Attempting to predict the Group variable, an initial logistic regression model is created using all other variables as predictors. However, as estimated earlier, the inclusion of CDR effectively stopped model convergence -due to the high correlation between the two- leading to the model not working as intended.

Removal of CDR allowed the model to converge. Model Two showed many of the predictors are statistically significant, with only EDUC, SES, eTIV, and ASF not having a p-value of <0.05. This model also had an AIC of 199.06, and a McFadden (pseudo R-Squared) value of 0.5758. Furthermore, VIF analysis showed eTIV and ASF have high levels of multicollinearity, as could be expected by their high correlation. This is the baseline of model effectiveness as it is the first working model, using all variables except CDR.

Feature Selection for Logistic Regression:

Within the feature selection portion of this project, several methods were used with the aim of increasing the accuracy of the logistic regression model, alongside other metrics which help identify effective models. VIF, AIC, AUC, and McFadden scores were also metrics used to assess the effectiveness of the model.

The first thing that was done to improve the model was to remove ASF as a predictor due to its high level of multicollinearity with eTIV. ASF was chosen over eTIV to be removed due its lower correlation with the dependent variable, and due to eTIV ability to lower AIC scores more than ASF. Doing this reduced AIC from 199.06 to 197.3 while only reducing the McFadden score from 0.5758 to 0.5752. A VIF check shows that there is no significant multicollinearity between any of the remaining predictors. This was labelled Model Three.

To further improve the model a backwards step function was used on model three. This function works by starting with a model full of all predictor variables and then proceeds to drop predictors to lower the AIC score, it continues doing this until dropping anymore predictors would no longer lower the AIC score. The backwards step function dropped first SES, then EDUC; the variables M.F, Age, MMSE, eTIV, and nWBV were left as the most important variables. Conducting a forward step function on the model also produced the same results; a forward step function works in a similar way to the backwards step, but starts from an empty model and adds predictors until doing so would no longer lower the AIC score. From this the model has had its AIC score reduced to a low of 196.99, and had a McFadden score of 0.5666. This created Mode Four.

Continuing to try and improve the model, a different feature selection method was used on Model Three – Boruta. Boruta uses a random forest to classify features by their importance within the model.

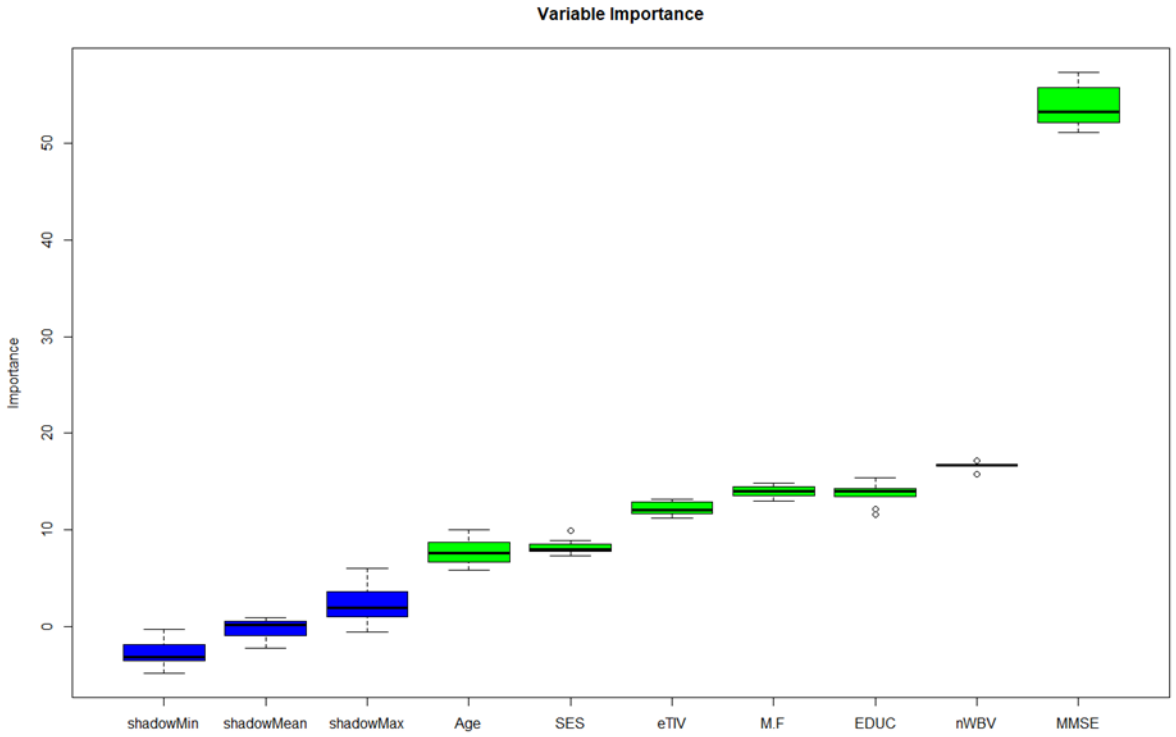

Figure 14: Boruta Feature Importance

Figure 14 above shows the results of this feature selection model. As can be seen Age has been given the lowest importance rating, thus it was dropped from the model. Furthermore, given the relatively close importance rating of Age and SES -which can be seen exactly on Table 6- it was decided to drop SES as well. This was surprising as the previously used feature selection techniques dropped EDUC and SES before dropping Age. However, to fully test Boruta both Age and SES were dropped, this created Model Five. Unfortunately, Boruta proved to be less effective than even the initial model created after dropping CDR, Model Two. Model Five resulted in an AIC of 215.88 and a McFadden score of 0.5224, noticeably worse than Model Four's previous score, and even worse than the original Model Two's AIC of 199.06 and McFadden value of 0.5758; clearly in this instance the simpler model is not as effective.

Thus, it has been established that Boruta is less effective at feature selection than a simple backwards/forwards step function, if the objective is to select the most important features for creating a more accurate and effective regression model. Although both functions did recognise MMSE, nWBV, eTIV, and Gender as important variables to retain.

With Model Four being the best model that could be found using the above techniques, a function was created to test the accuracy of the model. This uses a 70/30 train/test split from the dataset to train the regression model and then create predictions on subject's diagnosis, which is then tested on the smaller split of the dataset. It then loops 50 times and takes the percentage accuracy of all iterations and averages them.

new.pred	Var2	Freq
Demented	Demented	36
Nondemented	Demented	11
Demented	Nondemented	6
Nondemented	Nondemented	49

Table 7: Confusion Matrix showing Predicted Diagnosis against True Diagnosis

The above Table 7 shows the results of this function; the breakdown of predicted diagnosis (new.pred) and the true diagnosis (Var2). For Model Four the average consistently sits around 85%, and the AUC is 0.9369. This is generally quite good for this model and an improvement over the Model Two -the original model minus CDR- of roughly 1%.

Conclusion:

In conclusion, it is possible to build a model which has a good level of accuracy at predicting the diagnosis of Alzheimer's disease using the variables in available in the dataset. There are many strong connections between variables in the dataset, as the earlier descriptive statistics - notably the correlation matrix- identified, and which were later explored while building a logistic regression model. Noticeably, MMSE and nWBV were identified as important variables in the prediction of a subject's diagnosis, while ASF and SES were two of the variables first dropped during feature selection.

Unfortunately, the attempted using of K-Means clustering to successfully group variables and diagnoses proved highly ineffectual, being around as effective as random chance at grouping variables. Ultimately however, the successful creation of an effective logistic regression model counterbalances this as it allowed the central objective of this report to be completed; to successfully investigate the relationship between the variables and the diagnosis and create a model that can be used to predict diagnoses.

Bibliography:

Xavier Robin, Natacha Turck, Alexandre Hainard, Natalia Tiberti, Frédérique Lisacek, Jean-Charles Sanchez and Markus Müller (2011). pROC: an open-source package for R and S+ to analyze and compare ROC curves. BMC Bioinformatics, 12, p. 77. DOI: 10.1186/1471-2105-12-77

Adjusted Rand Index, (n.d.), *Package* pdfCluster *Index*. https://search.r-project.org/CRAN/refmans/pdfCluster/html/adj.rand.index.html

Buckner RL, Head D, Parker J, Fotenos AF, Marcus D, Morris JC, Snyder AZ. A unified approach for morphometric and functional data analysis in young, old, and demented adults using automated atlas-based head size normalization: reliability and validation against manual measurement of total intracranial volume. Neuroimage. 2004 Oct;23(2):724-38. doi: 10.1016/j.neuroimage.2004.06.018. PMID: 15488422.

Appendix:

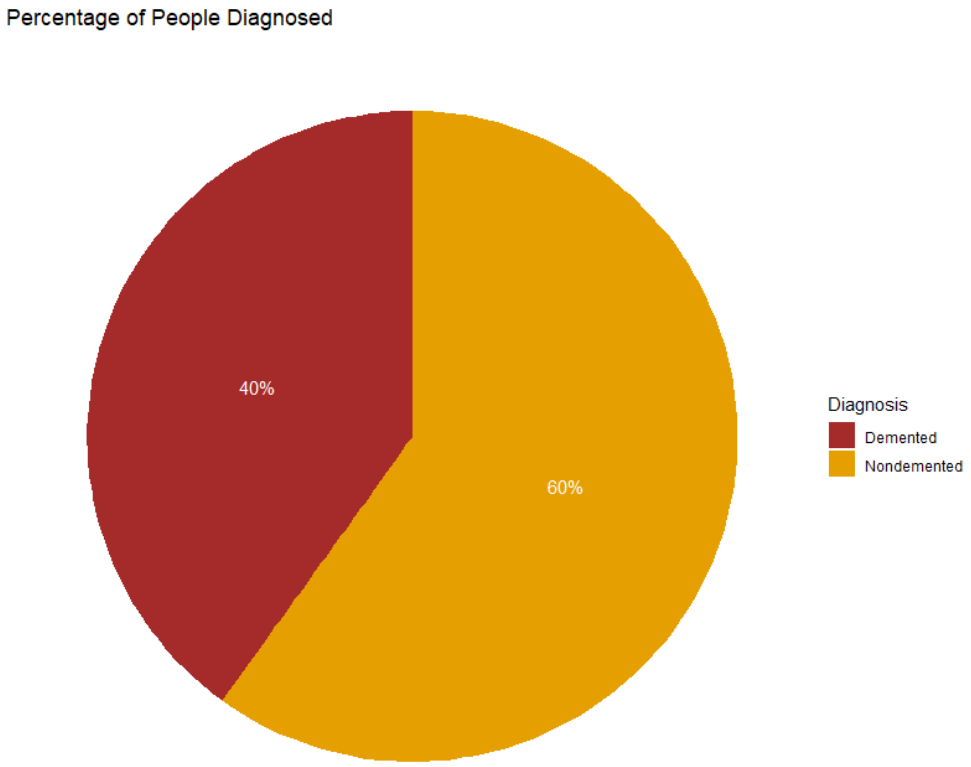

Figure 1: Percentage of People Diagnosed with Dementia

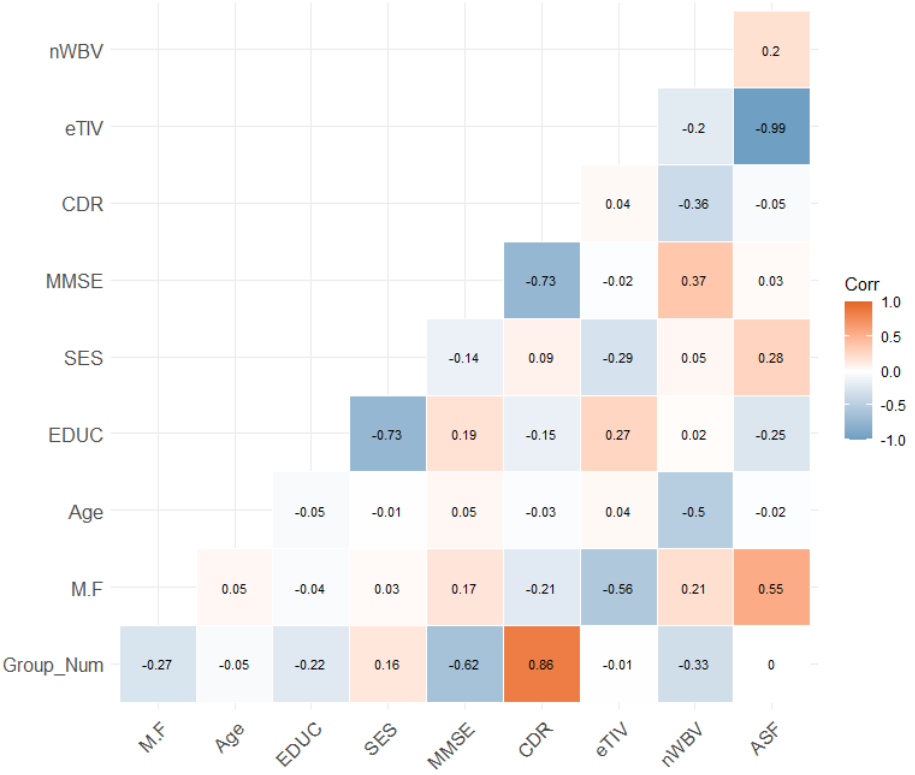

Figure 2: Correlation Matrix of All Variables

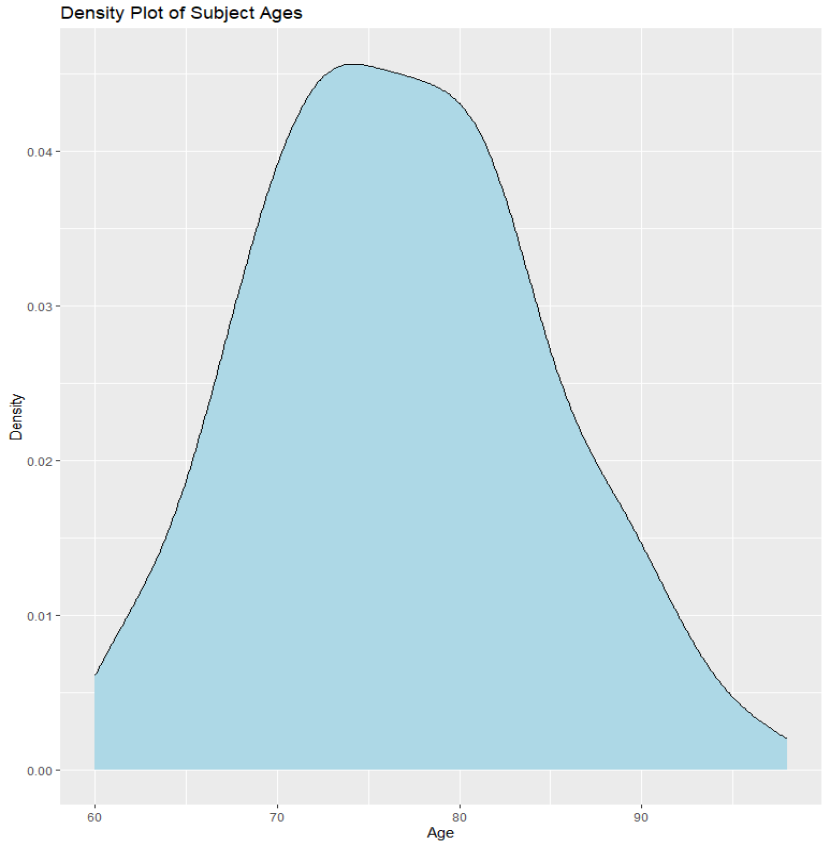

Figure 3: Density Plot of Age

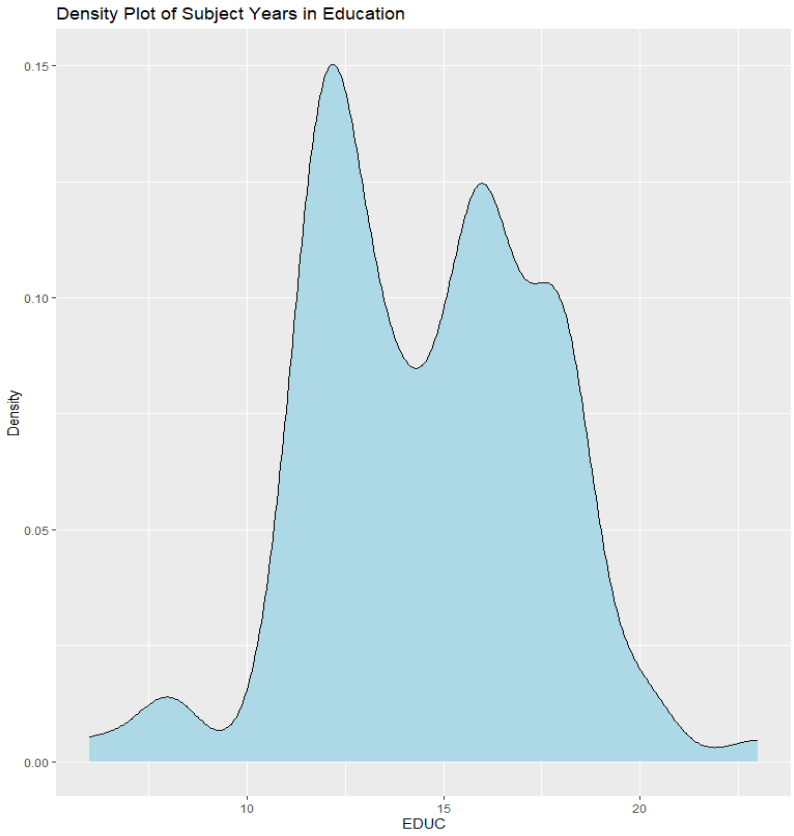

Figure 4: Density plot of Years in Education

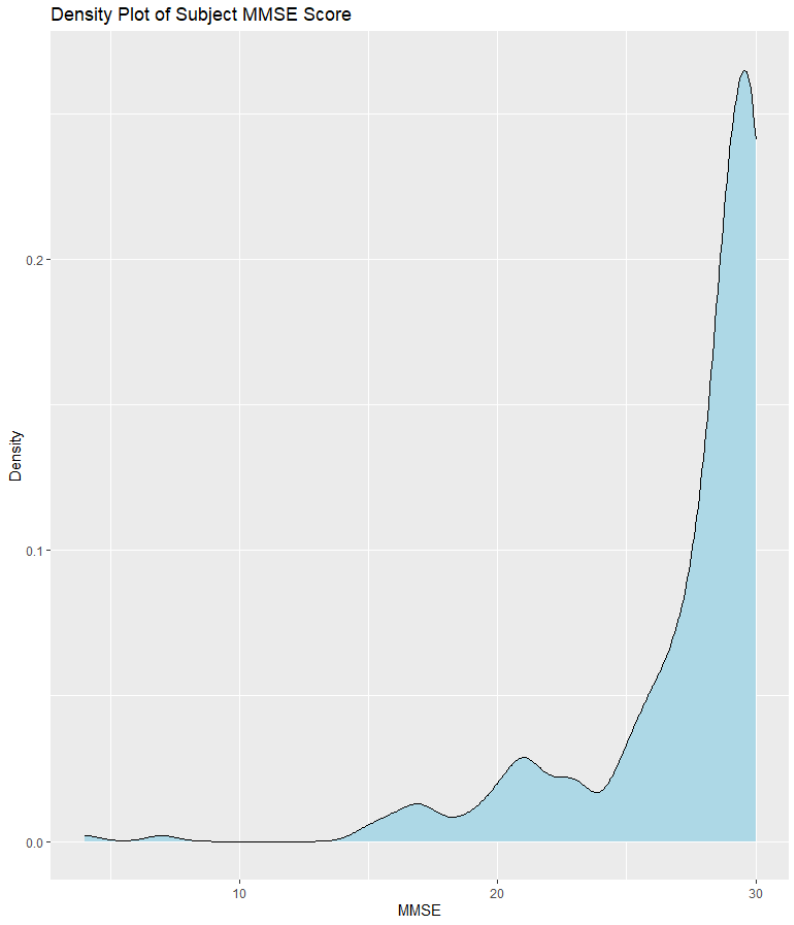

Figure 5: Density plot of MMSE score

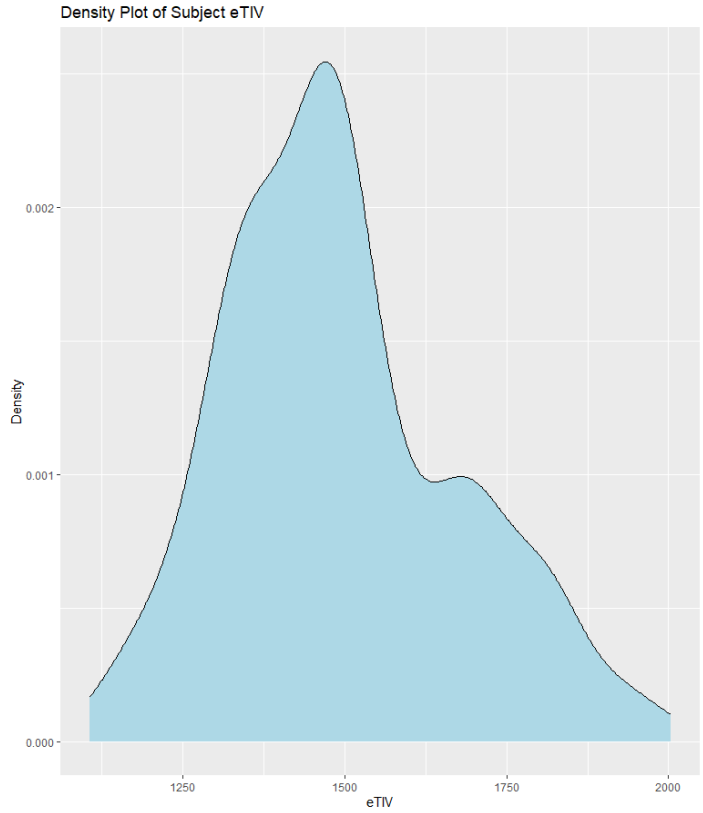

Figure 6: Density plot of subject eTIV

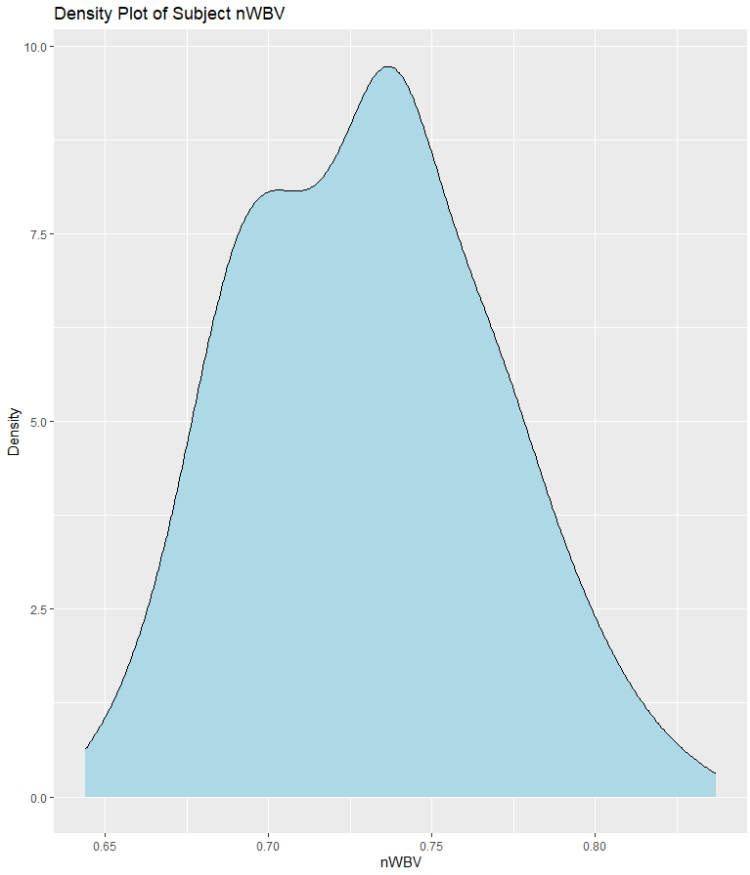

Figure 7: Density plot of subject nWBV

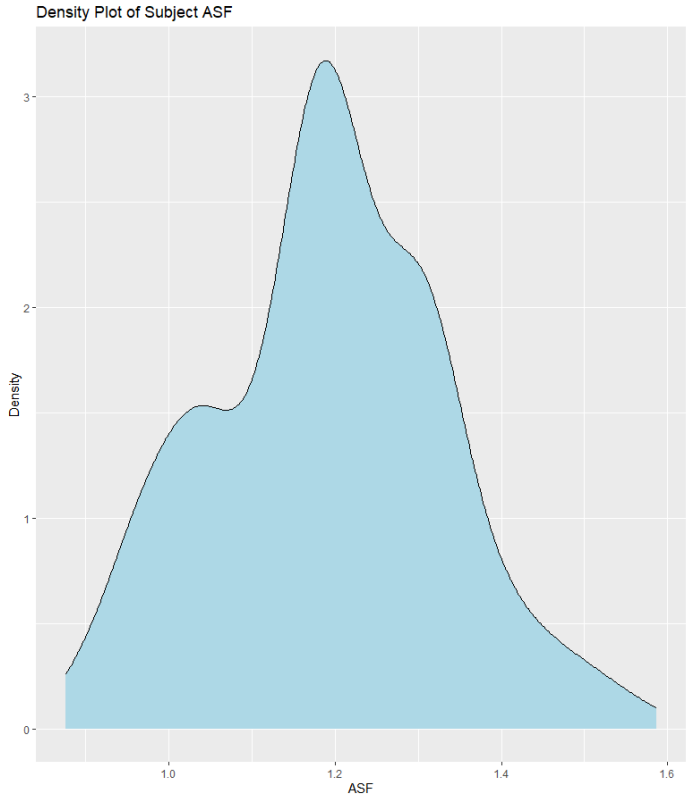

Figure 8: Density plot of subject ASF

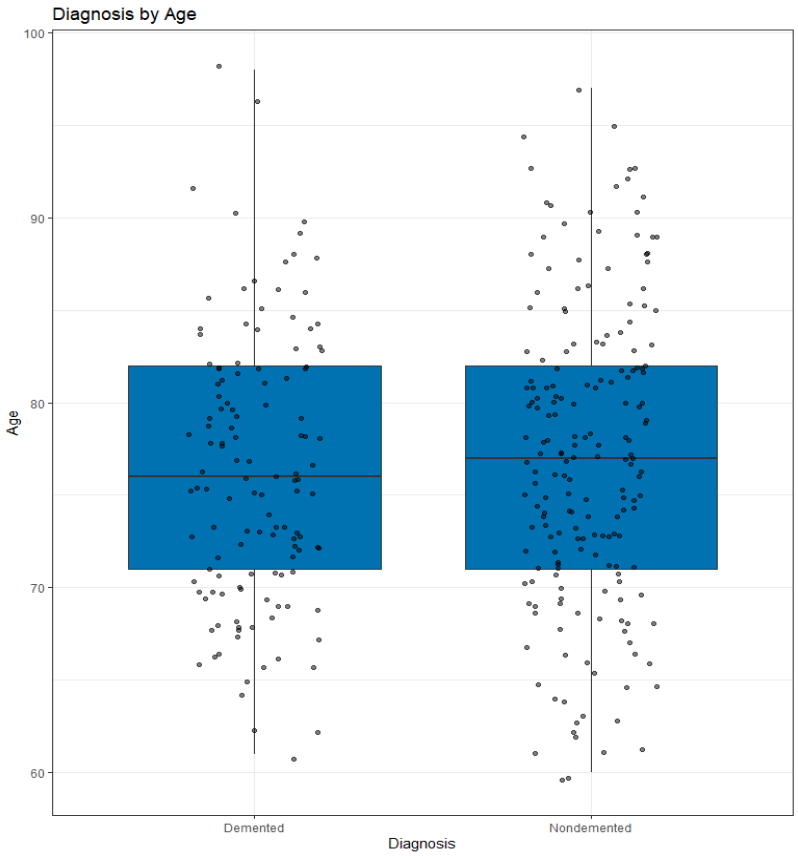

Figure 9: Boxplot of Diagnosis against Age

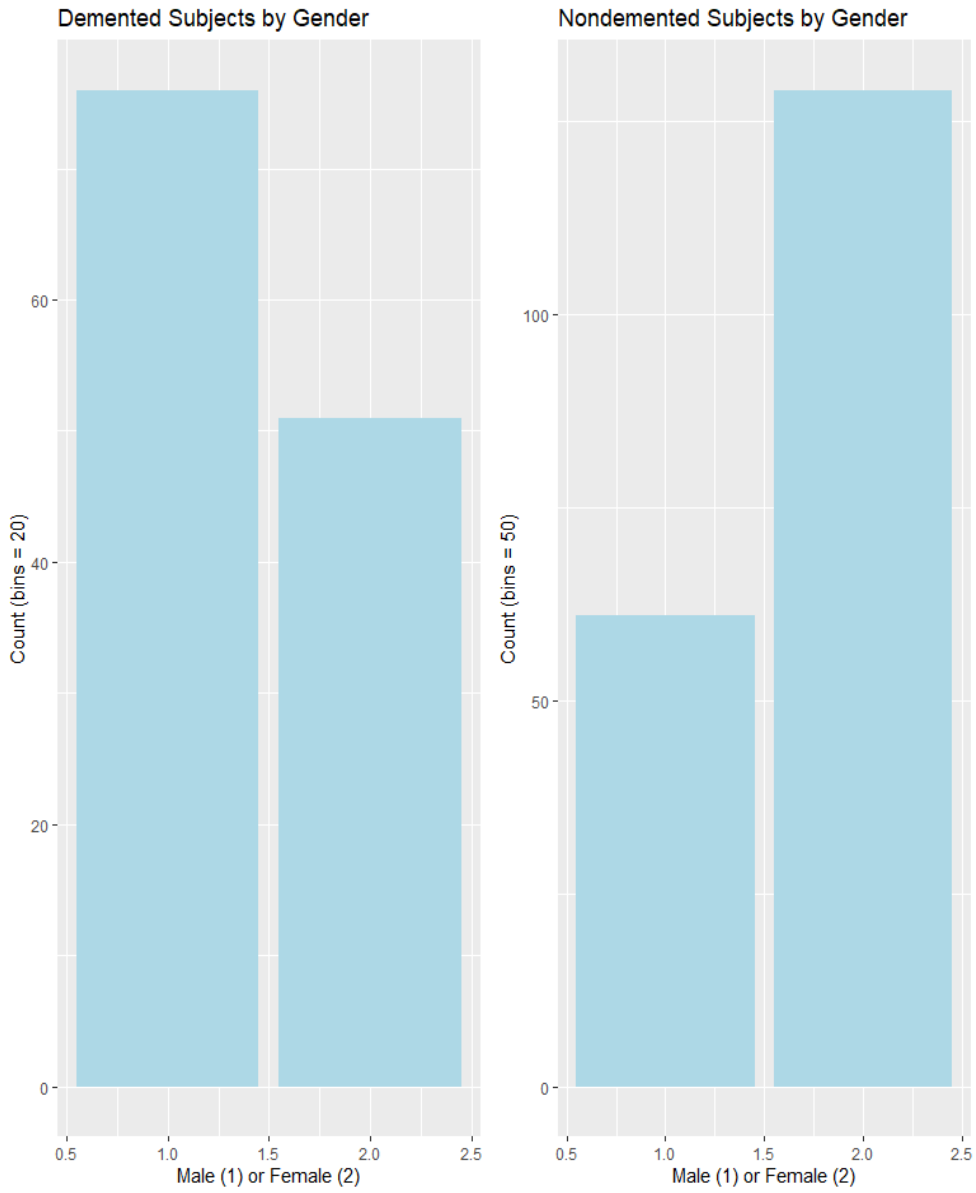

Figure 10: Diagnoses by gender

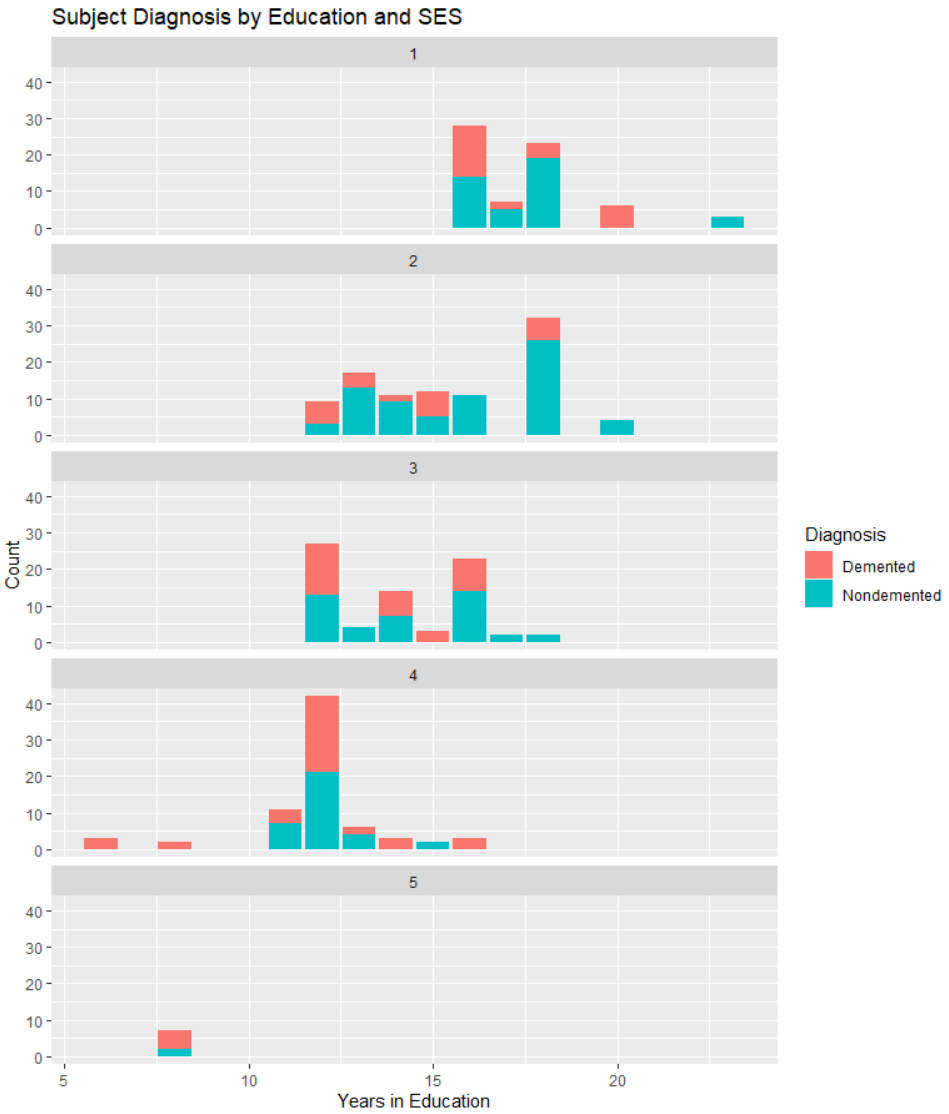

Figure 11: Diagnosis by Years in Education and SES

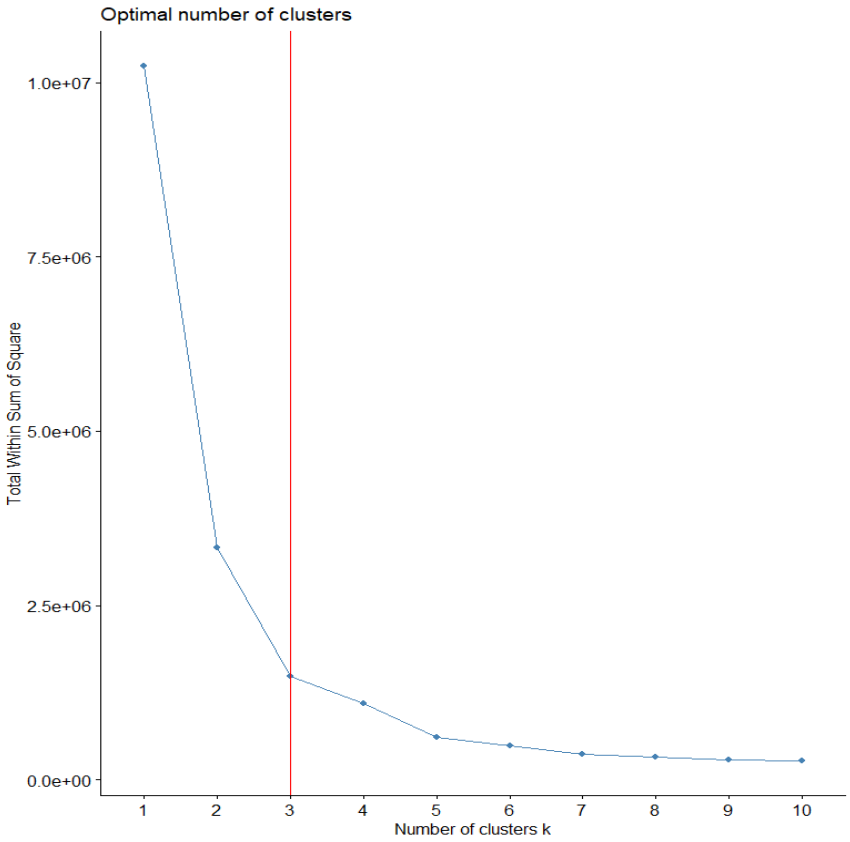

Figure 12: Optimal number of K clusters

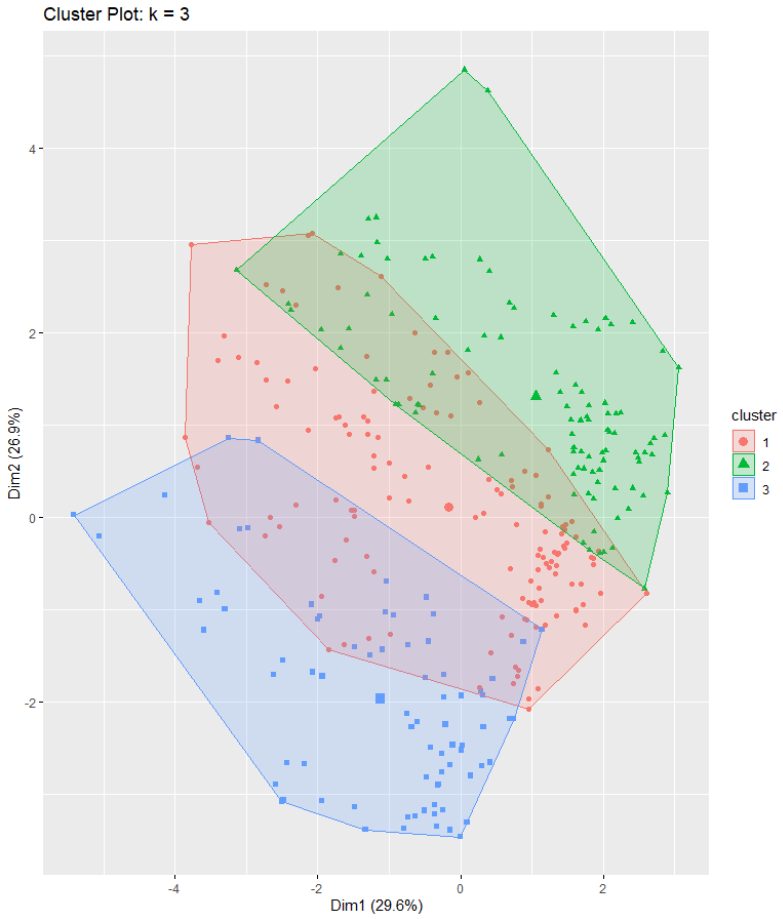

Figure 13: Cluster plot of K-means clustering

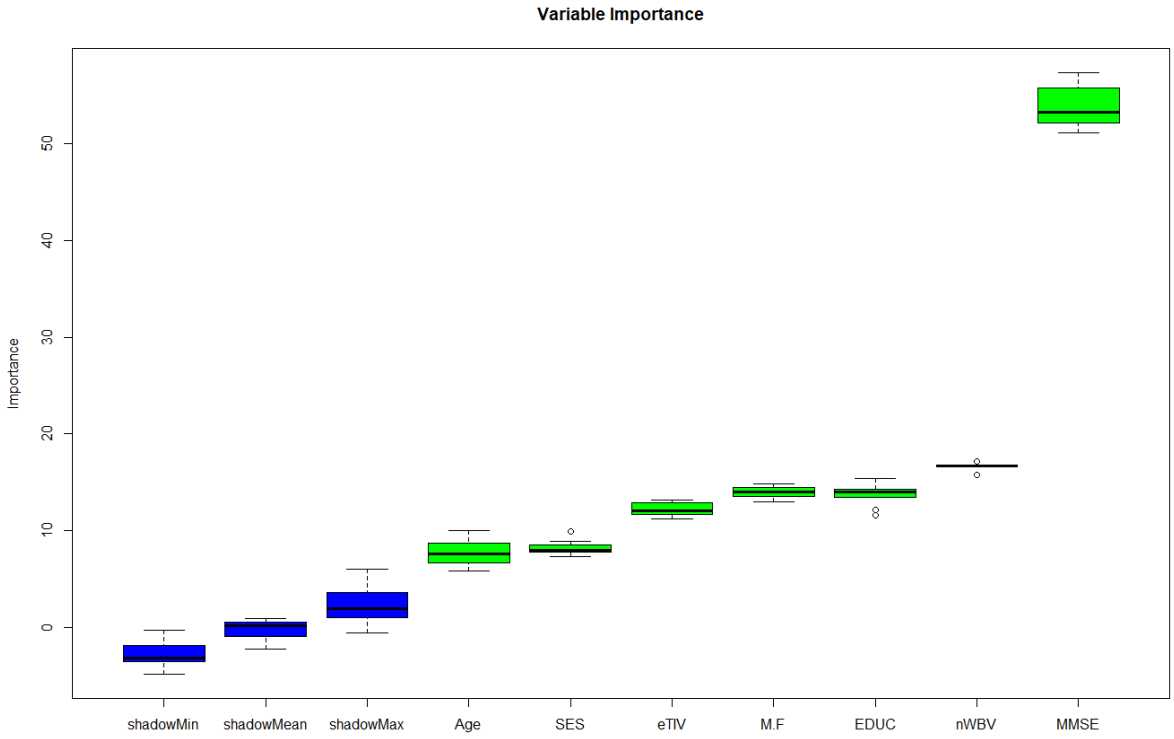

Figure 14: Variable Importance according to the Boruta method

CDR	MMSE_score	M.F	total_diagnoses	diagnosed	percentage
0.0	normal	1	59	0	0.00
0.0	normal	2	129	0	0.00
0.5	mild	1	8	8	100.00
0.5	mild	2	7	7	100.00
0.5	moderate	1	2	2	100.00
0.5	moderate	2	1	1	100.00
0.5	normal	1	48	46	95.83
0.5	normal	2	25	25	100.00
1.0	mild	1	10	10	100.00
1.0	mild	2	8	8	100.00
1.0	moderate	1	5	5	100.00
1.0	moderate	2	3	3	100.00
1.0	normal	1	2	2	100.00
1.0	normal	2	5	5	100.00
1.0	severe	1	2	2	100.00
2.0	mild	2	1	1	100.00
2.0	moderate	2	1	1	100.00
2.0	normal	1	1	1	100.00

Table 1: Diagnoses by CDR and MMSE score groups

clusterNorm	Var2	Freq
1	0	73
2	0	67
3	0	50
1	1	65
2	1	36
3	1	26

Table 2: Distribution of Diagnosis (Var2) by Cluster

clusterNorm	Var2	Freq
1	0	73
2	0	67
3	0	48
1	0.5	45
2	0.5	29
3	0.5	17
1	1	18
2	1	7
3	1	10
1	2	2
2	2	0
3	2	1

Table 3: Distribution of CDR (Var2) in Clusters

clusterNorm	Var2	Freq
1	1	55
2	1	16
3	1	66
1	2	83
2	2	87
3	2	10

Table 4: Distribution of Gender (Var2) in clusters

Cluster	Age	EDUC	MMSE	eTIV	nWBV	ASF
3	76.07971	14.44928	26.88406	1,489.109	0.7295072	1.179906
2	76.32039	13.78641	27.74757	1,307.194	0.7430874	1.346398
1	78.40789	16.03947	27.28947	1,754.289	0.7156447	1.003066

Table 5: Averages of each Continuous Variable in the Clusters

	meanImp	medianImp	minImp	maxImp	normHits	decision
nWBV	16.709270	16.688195	15.822014	17.204834	1	Confirmed
eTIV	12.228846	12.107828	11.267943	13.205527	1	Confirmed
MMSE	53.655179	53.208962	51.097104	57.308307	1	Confirmed
SES	8.221230	8.010502	7.389094	9.907376	1	Confirmed
EDUC	13.812935	14.041829	11.645208	15.413899	1	Confirmed
Age	7.798602	7.583959	5.849512	10.050239	1	Confirmed
M.F	13.991898	13.998280	12.977691	14.895615	1	Confirmed

Table 6: Predictor Variable Importance Measure According to Boruta Method

new.pred	Var2	Freq
Demented	Demented	36
Nondemented	Demented	11
Demented	Nondemented	6
Nondemented	Nondemented	49

Table 7: Confusion Matrix showing Predicted Diagnosis against True Diagnosis (Var2)

R Code:

```
#install required packages for this report
library(dplyr)
library(ggplot2)
library(ggcorrplot)
library(corrplot)
library(plotly)
library(tidyverse)
library(car)
library(pscl)
library(Boruta)
library(factoextra)
library(pdfCluster)
library(flextable)
library(pROC)
library(DT)

citation("pROC") #citation for package used
```

```
df1 <- read.csv("m:\\pc\\desktop\\Modelling\\project data.csv") #read in data file
df1 <- na.omit(df1) #drop missing values
df1$M.F <- ifelse(df1$M.F == "M", 1, 2) #change M to 1 and F to 2
df1 = filter(df1, Group != "Converted") #remove rows containing "converted" from the group column
df1$Group_Num <- ifelse(df1$Group == "Demented", 1, 0) #creates a new column with 0 = nondemented, and 1 = demented
head(df1)
str(df1)
unique(df1$M.F)
unique(df1$Group)#these are used to check the data cleaning is correctly done
unique(df1$CDR)
unique(df1$MMSE)
unique(df1$Age)


df1 <- df1 %>% mutate(MMSE_score = case_when(
  MMSE <= 9 ~ "severe",
  MMSE >= 10 & MMSE <= 18 ~ "moderate",
  MMSE >= 19 & MMSE <= 23 ~ "mild",
  MMSE >= 24 ~ "normal"
)) #creates a new column grouping MMSE scores into their diagnosis range to make visualisation easier

unique_count <- function(column) { #take the number of unique entries for each column and add them
  length(unique(column))
}
col_unique <- sapply(df1, unique_count)
print(col_unique) #this is used to get a better understanding of each column


##### DATA VISUALISATION
```

```
#Pie chart showing % of people diagnosed (Figure 1)
ggplot(df1, aes(x = "", fill = Group)) + #create a plot using ggplot2 from df1, leave x-axis empty
  geom_bar(width = 1, position = "fill") +
  scale_fill_manual(values = c("brown", "#E69F00")) + #set colours manually
  geom_text(stat = "count", aes(label = paste0(scales::percent(stat(count) / sum(stat(count))), "" , "")),
            position = position_fill(vjust = 0.5),color = "white") + #Add text labels to the plot displaying the percentage count of each Group
  coord_polar("y", start = 0) +
  theme_void() + ## Remove all unimportant elements from the plot
  ggtitle("Percentage of People Diagnosed") +
  guides(fill = guide_legend(title = "Diagnosis")) #set custom title and legend title
#40% diagnosed, 60% not.

#Correlation Matrix of all variables (Figure 2)
corrM <- cor(df1[, c("Group_Num", "M.F", "Age", "EDUC", "SES", "MMSE", "CDR", "eTIV", "nWBV", "ASF")]) #correlation matrix of all variables
ggcorrplot(corrM, type = "lower", outline.col = "white", lab = TRUE, lab_size = 3, colors = c("#6D9EC1", "white", "#E46726")) #plot corr matrix
#a correlation analysis is used to help identify potentially most useful predictor values
#and to see if there are any predictors which are highly correlated and could probably be removed from the model
#eTIV and ASF have effectively no correlation with Group, thus probably don't need to be in the model

## Density plots of all continuous variables

#for explanation of code for figures 3-8 see figure 3
#Density plot of age (Figure 3)
ggplot(data = df1, aes(x = Age)) + #using ggplot2 and df1 create density plots
  geom_density(fill = "lightblue") + #create density plot with lightblue fill colour
  labs(x = "Age", y = "Density") +
  ggtitle("Density Plot of Subject Ages") #custom title and axis labels
```

```
#Looks close to a normal distribution of ages between 60 and 98
shapiro.test(df1$Age)
#A quick test shows that Age is not normally distributed at p<0.05

#Density plot of Years in Education (Multiple peaks) (Figure 4)
ggplot(data = df1, aes(x = EDUC)) +
  geom_density(fill = "lightblue") +
  labs(x = "EDUC", y = "Density") +
  ggtitle("Density Plot of Subject Years in Education")

#Density plot of MMSE score (Highly skewed) (Figure 5)
ggplot(data = df1, aes(x = MMSE)) +
  geom_density(fill = "lightblue") +
  labs(x = "MMSE", y = "Density") +
  ggtitle("Density Plot of Subject MMSE Score")

#Density plot of subject eTIV (Figure 6)
ggplot(data = df1, aes(x = eTIV)) +
  geom_density(fill = "lightblue") +
  labs(x = "eTIV", y = "Density") +
  ggtitle("Density Plot of Subject eTIV")

#Density plot of subject nWBV (Figure 7)
ggplot(data = df1, aes(x = nWBV)) +
  geom_density(fill = "lightblue") +
  labs(x = "nWBV", y = "Density") +
  ggtitle("Density Plot of Subject nWBV")

#Density plot of subject ASF (Figure 8)
ggplot(data = df1, aes(x = ASF)) +
  geom_density(fill = "lightblue") +
  labs(x = "ASF", y = "Density") +
  ggtitle("Density Plot of Subject ASF")
```

```
#Boxplot of Diagnosis against Age (Figure 9)
ggplot(df1, aes(x = Group, y = Age)) + #set x-axis as Group and y-axis as Age
  geom_boxplot(fill = "#0072B2", outlier.shape = NA) + #create boxplot using ggplot2
  geom_jitter(position = position_jitter(width = 0.2), alpha = 0.5) + #jitter the datapoints to make it easier to see where they are located on the plot
  labs(x = "Diagnosis", y = "Age",
       title = "Diagnosis by Age") + #custom title and axis labels
  theme_bw()

#Diagnoses by gender (Figure 10)
gridExtra::grid.arrange(grobs = list( #set to plot multiple graphs
  ggplot(subset(df1, Group == "Demented"), aes(x = M.F)) + #plot 1 only show Demeted subjects
    geom_bar(fill = "Lightblue") + #ggplot2 bar plot with custom fill colour
    labs(title = "Demented Subjects by Gender",
         x = "Male (1) or Female (2)",
         y = "Count (bins = 20)"), #custom labels
  ggplot(subset(df1, Group == "Nondemented"), aes(x = M.F)) + #plot 2 only show Nondemeted subjects
    geom_bar(fill = "Lightblue") + #ggplot2 bar plot with custom fill colour
    labs(title = "Nondemented Subjects by Gender",
         x = "Male (1) or Female (2)",
         y = "Count (bins = 50)")), #custom labels
  nrow = 1) #plot should be arranged on a single row

#Bar graph: Diagnosis by EDUC and SES (Figure 11)
ggplot(df1, aes(x = EDUC, fill = Group)) + #ggplot2 barplot
  geom_bar(position = "stack") + #add another layer of bars stacked on top of each other
  facet_wrap(~ SES, ncol = 1) + #create facet plots arranged on a single column
  labs(title = "Subject Diagnosis by Education and SES",
       x = "Years in Education",
       y = "Count",
       fill = "Diagnosis") #custom labels
```

```
#table showing diagnoses by CDR and MMSE score groups (Table 1)
df_demented <- df1 %>% #create new dataframe, grouped by CDR, MMSE, and Gender
  group_by(CDR, MMSE_score, M.F) %>%
  summarize(total_diagnoses = n(), #summarise the data
            diagnosed = sum(Group == "Demented")) %>% #find total count of diagnoses and total count of Demented diagnoses within that
  ungroup() %>% #remove grouping
  mutate(percentage = round(diagnosed / total_diagnoses * 100,2 )) #calculate percentage of Demented diagnoses
datatable(df_demented, #create datatable visualising the above
          options = list(pageLength = 20, lengthChange = FALSE))
#If the person has a CDR score that is not 0, then they have dementia, and will likely be diagnosed



##### CLUSTERING ALGORITHMS
set.seed(123) # Set seed for reproducibility
df1Norm <- data.frame(df1) #new data frame for clustering
df1Norm <- subset(df1Norm, select = -c(Group, MMSE_score)) #remove redundant columns/duplicate information
str(df1Norm)
contVars <- c("Age", "EDUC", "MMSE", "eTIV", "nWBV", "ASF") #create list of continuous variables for use later to assess average values



#K-Mean Clustering
fviz_nbclust(df1Norm, kmeans, method = "wss")+ #find TWSS per cluster and plot
  geom_vline(xintercept = 3, color = "red") #(Figure 12)

kmeansNorm <- kmeans(df1Norm, centers = 3, nstart = 20) #run k-means function with clusters = 3
clusterPlot <- fviz_cluster(kmeansNorm, geom = "point", data = df1Norm) + ggtitle("Cluster Plot: k = 3")
plot(clusterPlot) #plot k-means clusters (Figure 13)
```

```
clusterNorm <- kmeansNorm$cluster #select cluster information
table(clusterNorm, df1Norm$Group_Num) #Distribution of diagnosis in clusters
table(clusterNorm, df1Norm$CDR) #Distribution of CDR in clusters
table(clusterNorm, df1Norm$M.F) #Distribution of Gender in clusters
#Create savable tables that look nice from the above information
ftable1 <- flextable(data.frame(table(clusterNorm, df1Norm$Group_Num))) #create a flextable showing breakdown of variable assignment to clusters
ftable1 #Table 2
save_as_docx(ftable1, path = "table_2.docx") #save flextables to a word doc for easy access
ftable2 <- flextable(data.frame(table(clusterNorm, df1Norm$CDR)))
ftable2 #Table 3
save_as_docx(ftable2, path = "table_3.docx")
ftable3 <- flextable(data.frame(table(clusterNorm, df1Norm$M.F)))
ftable3 #Table 4
save_as_docx(ftable3, path = "table_4.docx")

clusterAvg <- data.frame(Cluster = unique(clusterNorm)) #Average of each contVar in the clusters
for (var in contVars) { #Iterate over each continuous variable
  avg_values <- tapply(df1Norm[[var]], clusterNorm, mean) #Compute the average value of the variables in each cluster
  clusterAvg[[var]] <- avg_values #Add the average values to the new data frame
}
clusterAvg
ftableAvg <- flextable(clusterAvg) #use average values to plot to a table
ftableAvg #Table 5
save_as_docx(ftableAvg, path = "table_5.docx") #save to word

#Showing how effective the clustering is: <0 = worse than chance, 0= same as chance, 1= Perfect alignment with truth
ARI1 <- adj.rand.index(df1Norm$Group_Num, clusterNorm) #function identifying how effective the clustering is, these cover 3 variables
ARI1
ARI2 <- adj.rand.index(df1Norm$CDR, clusterNorm)
```

```
ARI2
ARI3 <- adj.rand.index(df1Norm$M.F, clusterNorm)
ARI3

#repeating the above but using 5 clusters instead of 3
df1Norm1 <- data.frame(df1)
df1Norm1 <- subset(df1Norm1, select = -c(Group, MMSE_score))
kmeansNorm1 <- kmeans(df1Norm1, centers = 5, nstart = 20) #clusters set to 5
clusterNorm1 <- kmeansNorm1$cluster
ARI15 <- adj.rand.index(df1Norm1$Group_Num, clusterNorm1)
ARI15
ARI25 <- adj.rand.index(df1Norm1$CDR, clusterNorm1)
ARI25
ARI35 <- adj.rand.index(df1Norm1$M.F, clusterNorm1)
ARI35

##### LOGISTIC REGRESSION

#Logistic regression model for all variables
#Code and functions will work the same for all "Model#" thus comments on Model 1 will not be repeated on following Model#
model1 <- glm(Group_Num~M.F+Age+EDUC+SES+MMSE+eTIV+nWBV+ASF+CDR, family="binomial", data=df1) #create logistic regression model using all variables
summary(model1) #show output of above regression model
vif(model1) #check for colinearity
pR2(model1)["McFadden"] #high McFadden rsqr val shows the usefulness of the model.psuedo R-Squared value
#This model does not work
#as previous analysis showed, if someone has a CDR rating above 0 then they were also demented (4 anomalies), hence the populations of Group and CDR are the same -chance


model2 <- glm(Group_Num~M.F+Age+EDUC+SES+MMSE+eTIV+nWBV+ASF, family="binomial", data=df1) #CDR removed from predictors
```

```
summary(model2)
vif(model2)
pR2(model2)["McFadden"]
#This model, while not perfect, does work using all variables (minus CDR)
#Although it can be seen that some variables have high levels of colinearity.



##### FEATURE SELECTION

#New model with ASF removed due to high multicollinearity.
model3 <- glm(Group_Num~M.F+Age+EDUC+SES+MMSE+eTIV+nWBV, family="binomial", data=df1)
summary(model3)
vif(model3)
pR2(model3)["McFadden"]
#This model has much lower levels of colinearity, and has a lower AIC score, without majorly affecting the pseudo-R2 val.
#AIC 197.3 and psudeo-R2 val 0.5753

#Now through feature selection functions, attempts will be made to improve the model
bkwrdstep <- step(model3, method = "backward") #uses all predictors then drops the predictor with the lowest SS score, reducing AIC
#Group_Num ~ M.F + Age + MMSE + eTIV + nWBV appears to be the best model according to the bkwrdstep algorithm
#a forward step on the same model to ensure the results are correct.
fwrdstep <- step(model3, method = "forward")

#Using another feature selection model lets see if the 'best' models align
#Boruta applies a random forest to rate features by their importance in the model
boruta1 <- Boruta(Group_Num~M.F+Age+EDUC+SES+MMSE+eTIV+nWBV, family="binomial", data = df1, doTrace = 1) #run boruta feature selection on Model 3
decision <- boruta1$finalDecision
```

```
signif <- decision[boruta1$finalDecision %in% c("Confirmed")] #select significant predictors according to boruta algorithm
print(signif) #print significant predictors
plot(boruta1, xlab="", main="Variable Importance") #(Figure 14) create plot showing figure importance
attStats(boruta1) #(Table 6) create table showing feature importance

#test the bkwrdstep and frwrdstep model
model4 <- glm(Group_Num~M.F+Age+MMSE+eTIV+nWBV, family="binomial", data=df1)
summary(model4)
pR2(model4)["McFadden"]
#AIC 196.99 and psudeo-R2 val 0.5666

#test the Boruta model (dropping lowest values Age and SES)
model5 <- glm(Group_Num~M.F+EDUC+MMSE+eTIV+nWBV, family="binomial", data=df1)
summary(model5)
pR2(model5)["McFadden"]
#AIC 215.88 and psudeo-R2 val 0.5224 (dropping anything else makes the model worse)

#the result of this analysis has created model4, the best from this testing

#Evalute model4's performance using several metrics
#building the #train/test split prediction model based off our previous logistic regression model
TTmodelsplit <- function(df) { #creates a function that runs the train and test split model 50 times to get an avg model accuracy
  results <- numeric(50)  #vector to store results

  for (i in 1:50) {
    sample <- sample(c(TRUE, FALSE), nrow(df), replace = TRUE, prob = c(0.7, 0.3)) #assigns 70/30 split of T/F vals to rows
    train <- df[sample, ]
    test <- df[!sample, ] #these check for T/F vals, then creates samples from df1 split into train and test sets based on that val
```

```
  modeltt <- glm(Group_Num~M.F+Age+MMSE+eTIV+nWBV, family = "binomial", data = train) #based on model4 but using the Train split
  new.probs <- predict(modeltt, test, type = "response") #uses the trained regression model to make predictions for test set

  test.length <- nrow(test) #counters the problem of the variable length of "test" from the way df1 is split
  new.pred <- rep("Nondemented", test.length) #new vector filled with Nondemented values
  new.pred[new.probs > 0.5] <- "Demented" #updates the vector, changing values to Demented if the prediction thinks its a more likely result

  results[i] <- mean(new.pred == test$Group) #store the results of each iteration, and proportion of correct predictions
 }

 avg_result <- mean(results) #averages the proportion of correct predictions across all iterations, to get an overall average accuracy
 return(avg_result)
}
avg_accuracy <- TTmodelsplit(df1)
print(avg_accuracy) #takes output of the function and prints it


#Area Under Curve (AUC) of ROC curve for test set
rocm4 <- roc(test$Group, new.probs)
rocm4

#Confusion Matrix for the test set
cMatrix <- table(new.pred, test$Group)
cMatrix
ftabCM <- flextable(data.frame(cMatrix)) #Table 7
save_as_docx(ftabCM, path = "table_7.docx") #save table as a word doc
```

```
#Compare the optimal model with the original model to assess how much it has improved
TTmodelsplitOG <- function(df) { #creates a function that runs the train and test split model 50 times to get an avg model accuracy
  resultsOG <- numeric(50)  #vector to store results

  for (i in 1:50) {
    sample <- sample(c(TRUE, FALSE), nrow(df), replace = TRUE, prob = c(0.7, 0.3)) #assigns 70/30 split of T/F vals to rows
    train <- df[sample, ]
    test <- df[!sample, ] #these check for T/F vals, then creates samples from df1 split into train and test sets based on that val

    modelOG <- glm(Group_Num~M.F+Age+EDUC+SES+MMSE+eTIV+nWBV+ASF, family = "binomial", data = train) #based on model2 but using the Train split
    OG.probs <- predict(modelOG, test, type = "response") #uses the trained regression model to make predictions for test set

    test.length <- nrow(test) #counters the problem of the variable length of "test" from the way df1 is split
    OG.pred <- rep("Nondemented", test.length) #new vector filled with Nondemented values
    OG.pred[OG.probs > 0.5] <- "Demented" #updates the vector, changing values to Demented if the prediction thinks its a more likely result

    resultsOG[i] <- mean(OG.pred == test$Group) #store the results of each iteration, and proportion of correct predictions
  }

  avg_resultOG <- mean(resultsOG) #averages the proportion of correct predictions across all iterations, to get an overall average accuracy
  return(avg_resultOG)
}
avg_accuracyOG <- TTmodelsplitOG(df1)
print(avg_accuracyOG) #takes output of the function and prints it
```