project.

ramya

2024-11-29

###DATA PRE PROCESSING###

```
# dataset
data <- read.csv("~/Downloads/parkinsons_disease_data.csv")
# View the first few rows to confirm the data is loaded
head(data)
```

```
##   PatientID Age Gender Ethnicity EducationLevel      BMI Smoking
## 1      3058  85      0         3              1 19.61988       0
## 2      3059  75      0         0              2 16.24734       1
## 3      3060  70      1         0              0 15.36824       0
## 4      3061  52      0         0              0 15.45456       0
## 5      3062  87      0         0              1 18.61604       0
## 6      3063  68      1         2              1 39.42331       1
##   AlcoholConsumption PhysicalActivity DietQuality SleepQuality
## 1           5.108241        1.3806599    3.893969     9.283194
## 2           6.027648        8.4098041    8.513428     5.602470
## 3           2.242135        0.2132746    6.498805     9.929824
## 4           5.997788        1.3750452    6.715033     4.196189
## 5           9.775243        1.1886071    4.657572     9.363925
## 6          13.596889        7.7967040    7.070239     7.737549
##   FamilyHistoryParkinsons TraumaticBrainInjury Hypertension Diabetes Depression
## 1                       0                    0            0        0          0
## 2                       0                    0            0        0          0
## 3                       0                    0            0        1          0
## 4                       0                    0            0        0          0
## 5                       0                    0            0        0          0
## 6                       0                    0            0        0          0
##   Stroke SystolicBP DiastolicBP CholesterolTotal CholesterolLDL CholesterolHDL
## 1      0        129          60         222.8423      148.12562       37.86778
## 2      0        163          76         210.5011      153.75646       77.22812
## 3      0        113          93         287.3880      118.70260       85.58830
## 4      0        146          78         280.3395      136.29919       51.86963
## 5      0        115          94         284.0142      108.44945       25.06942
## 6      0        151          90         290.1331       91.75022       54.48892
##   CholesterolTriglycerides      UPDRS      MoCA FunctionalAssessment Tremor
## 1                 337.3071   6.458713 29.181289             1.572427      1
## 2                 264.6355  37.306703 12.332639             4.787551      0
## 3                 395.6626  67.838170 29.927783             2.130686      1
## 4                 362.1897  52.964696 21.304268             3.391288      1
## 5                 149.9566  21.804880  8.336364             3.200969      0
```

```
## 6                253.7973 101.912536 27.370580             6.824779      0
##   Rigidity Bradykinesia PosturalInstability SpeechProblems SleepDisorders
## 1        0            0                   0              0              0
## 2        1            0                   1              0              1
## 3        0            0                   0              1              0
## 4        1            1                   0              0              0
## 5        0            0                   1              0              1
## 6        0            0                   0              0              0
##   Constipation Diagnosis DoctorInCharge
## 1            0         0    DrXXXConfid
## 2            0         1    DrXXXConfid
## 3            1         1    DrXXXConfid
## 4            1         1    DrXXXConfid
## 5            0         0    DrXXXConfid
## 6            0         0    DrXXXConfid
```

#checking for null values

```
total_null_values <- sum(is.na(data))
print(total_null_values)
```

```
## [1] 0
```

#Showing the datatype and values.

```
str(data)
```

```
## 'data.frame':    2105 obs. of  35 variables:
##  $ PatientID              : int  3058 3059 3060 3061 3062 3063 3064 3065 3066 3067 ...
##  $ Age                    : int  85 75 70 52 87 68 78 70 80 71 ...
##  $ Gender                 : int  0 0 1 0 0 1 1 1 0 0 ...
##  $ Ethnicity              : int  3 0 0 0 0 2 0 0 2 3 ...
##  $ EducationLevel         : int  1 2 0 0 1 1 0 0 1 2 ...
##  $ BMI                    : num  19.6 16.2 15.4 15.5 18.6 ...
##  $ Smoking                : int  0 1 0 0 0 1 1 1 1 1 ...
##  $ AlcoholConsumption     : num  5.11 6.03 2.24 6 9.78 ...
##  $ PhysicalActivity       : num  1.381 8.41 0.213 1.375 1.189 ...
##  $ DietQuality            : num  3.89 8.51 6.5 6.72 4.66 ...
##  $ SleepQuality           : num  9.28 5.6 9.93 4.2 9.36 ...
##  $ FamilyHistoryParkinsons: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ TraumaticBrainInjury   : int  0 0 0 0 0 0 0 0 0 1 ...
##  $ Hypertension           : int  0 0 0 0 0 0 1 0 0 0 ...
##  $ Diabetes               : int  0 0 1 0 0 0 0 1 1 0 ...
##  $ Depression             : int  0 0 0 0 0 0 0 0 0 1 ...
##  $ Stroke                 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ SystolicBP             : int  129 163 113 146 115 151 122 129 133 169 ...
##  $ DiastolicBP            : int  60 76 93 78 94 90 60 99 113 105 ...
##  $ CholesterolTotal       : num  223 211 287 280 284 ...
##  $ CholesterolLDL         : num  148 154 119 136 108 ...
##  $ CholesterolHDL         : num  37.9 77.2 85.6 51.9 25.1 ...
##  $ CholesterolTriglycerides: num  337 265 396 362 150 ...
##  $ UPDRS                  : num  6.46 37.31 67.84 52.96 21.8 ...
```

```
##  $ MoCA                    : num  29.18 12.33 29.93 21.3 8.34 ...
##  $ FunctionalAssessment    : num  1.57 4.79 2.13 3.39 3.2 ...
##  $ Tremor                  : int  1 0 1 1 0 0 1 1 0 0 ...
##  $ Rigidity                : int  0 1 0 1 0 0 0 0 0 0 ...
##  $ Bradykinesia            : int  0 0 0 1 0 0 0 0 0 0 ...
##  $ PosturalInstability     : int  0 1 0 0 1 0 0 1 0 0 ...
##  $ SpeechProblems          : int  0 0 1 0 0 0 1 0 0 0 ...
##  $ SleepDisorders          : int  0 1 0 0 1 0 0 0 0 1 ...
##  $ Constipation            : int  0 0 1 1 0 0 0 1 0 0 ...
##  $ Diagnosis               : int  0 1 1 1 0 0 0 1 1 0 ...
##  $ DoctorInCharge          : chr  "DrXXXConfid" "DrXXXConfid" "DrXXXConfid" "DrXXXConfid" ...
```

#Checking for duplicate values.

```
# 1. Checking for any duplicate rows
any_duplicate_rows <- any(duplicated(data))
print(paste("Any duplicate rows found:", any_duplicate_rows))
```

```
## [1] "Any duplicate rows found: FALSE"
```

```
# 2. Checking for any duplicate columns
any_duplicate_columns <- any(duplicated(as.list(data)))
print(paste("Any duplicate columns found:", any_duplicate_columns))
```

```
## [1] "Any duplicate columns found: FALSE"
```

```
# 3. Removing duplicates (rows and columns)
if (any_duplicate_rows | any_duplicate_columns) {
  data <- data[!duplicated(data), ]                          # Remove duplicate rows
  data <- data[, !duplicated(as.list(data))]                  # Remove duplicate columns
}

# 4. Getting updated shape of the dataset
updated_shape <- dim(data)
print(paste("Updated shape:", updated_shape[1], "rows and", updated_shape[2], "columns"))
```

```
## [1] "Updated shape: 2105 rows and 35 columns"
```

#The datasets doesnot have any duplicate values. As the shape is same before and after update.

```
# Load necessary libraries
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
# Listing of numeric columns
numeric_columns <- data %>%
  select_if(is.numeric) %>%
  names()

# Function to calculate outliers using IQR
calculate_outliers <- function(x) {
  Q1 <- quantile(x, 0.25, na.rm = TRUE)
  Q3 <- quantile(x, 0.75, na.rm = TRUE)
  IQR <- Q3 - Q1
  lower_bound <- Q1 - 1.5 * IQR
  upper_bound <- Q3 + 1.5 * IQR
  sum(x < lower_bound | x > upper_bound, na.rm = TRUE) # Count of outliers
}

# Calculating outliers for each numeric variable
outlier_summary <- data.frame(
  Variable = numeric_columns,
  Outliers = sapply(data[numeric_columns], calculate_outliers)
)

# Filtering variables with zero outliers
zero_outlier_variables <- outlier_summary %>%
  filter(Outliers == 0)

# Displaying the results
print(zero_outlier_variables)
```

```
##                                         Variable Outliers
## PatientID                                PatientID        0
## Age                                            Age        0
## Gender                                      Gender        0
## EducationLevel                       EducationLevel        0
## BMI                                            BMI        0
## Smoking                                    Smoking        0
## AlcoholConsumption             AlcoholConsumption        0
## PhysicalActivity                 PhysicalActivity        0
## DietQuality                           DietQuality        0
## SleepQuality                         SleepQuality        0
## SystolicBP                             SystolicBP        0
## DiastolicBP                           DiastolicBP        0
## CholesterolTotal                 CholesterolTotal        0
## CholesterolLDL                     CholesterolLDL        0
## CholesterolHDL                     CholesterolHDL        0
## CholesterolTriglycerides CholesterolTriglycerides        0
## UPDRS                                       UPDRS        0
## MoCA                                         MoCA        0
## FunctionalAssessment         FunctionalAssessment        0
## Tremor                                     Tremor        0
## Rigidity                                 Rigidity        0
```

```
## SpeechProblems                     SpeechProblems        0
## Constipation                         Constipation        0
## Diagnosis                               Diagnosis        0
```

The following varaibles doesnot show any outliers. * Distribution:

```
# Load necessary libraries
library(ggplot2)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

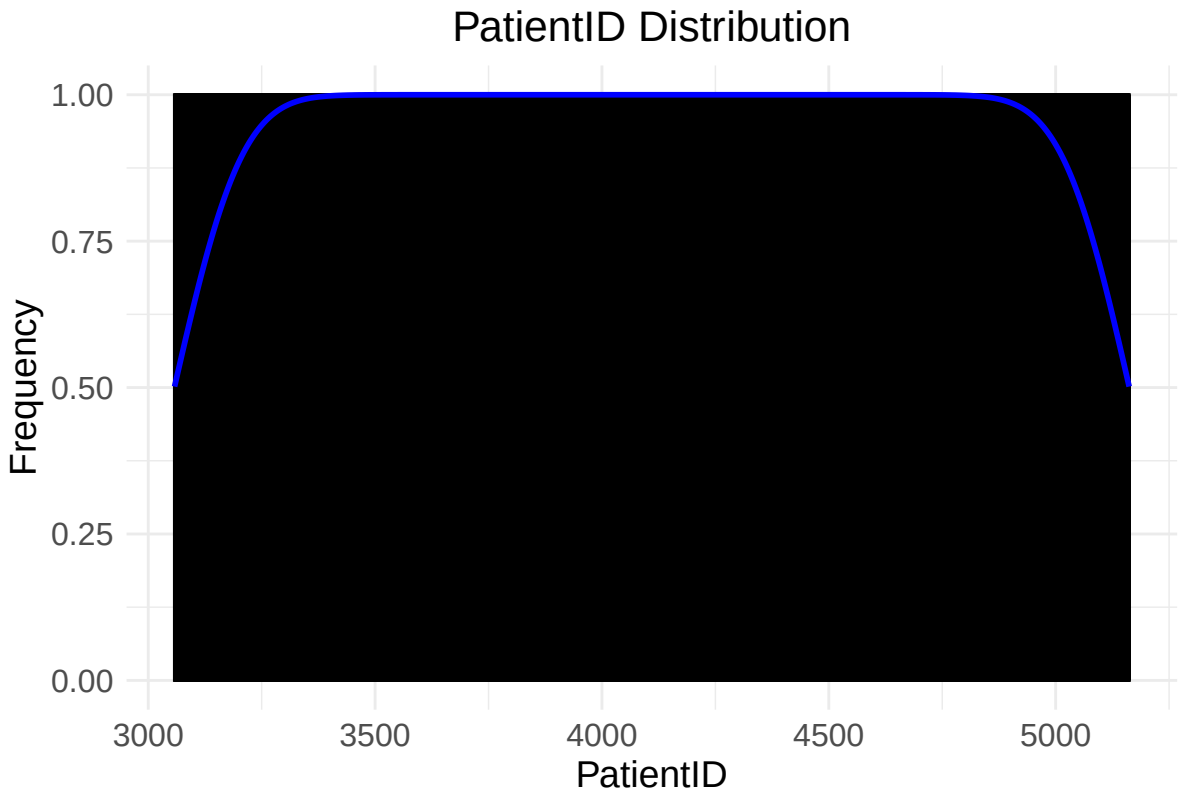
```
## The following object is masked from 'package:dplyr':
##
##     combine
```

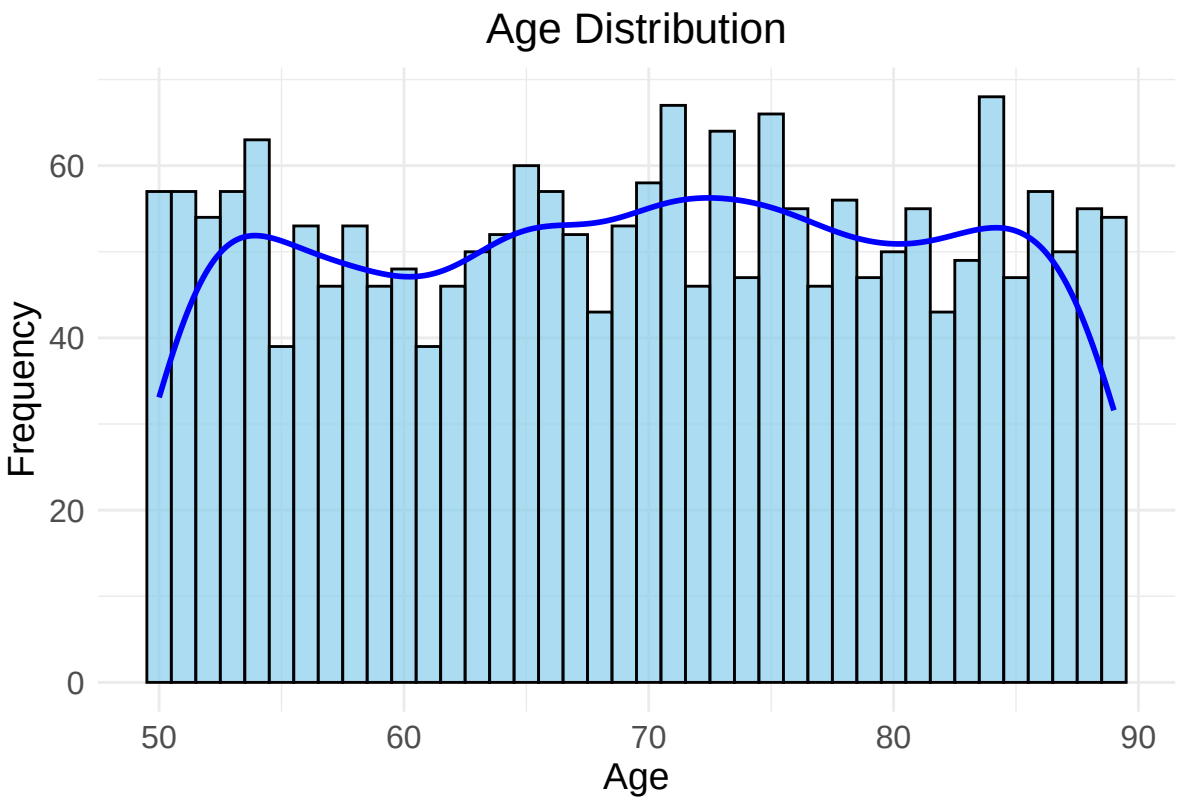
```
# Select only numeric columns
numeric_columns <- names(data)[sapply(data, is.numeric)]

# Create histograms for all numeric columns
histograms <- lapply(numeric_columns, function(col) {
  ggplot(data, aes_string(x = col)) +
    geom_histogram(binwidth = 1, fill = "skyblue", color = "black", alpha = 0.7) +
    geom_density(aes(y = after_stat(count)), color = "blue", size = 1) + # Overlay density plot
    labs(title = paste(col, "Distribution"), x = col, y = "Frequency") +
    theme_minimal() +
    theme(
      plot.title = element_text(hjust = 0.5, size = 16),
      axis.title = element_text(size = 14),
      axis.text = element_text(size = 12),
      plot.margin = margin(15, 15, 15, 15) # Add spacing around each plot
    )
})
```

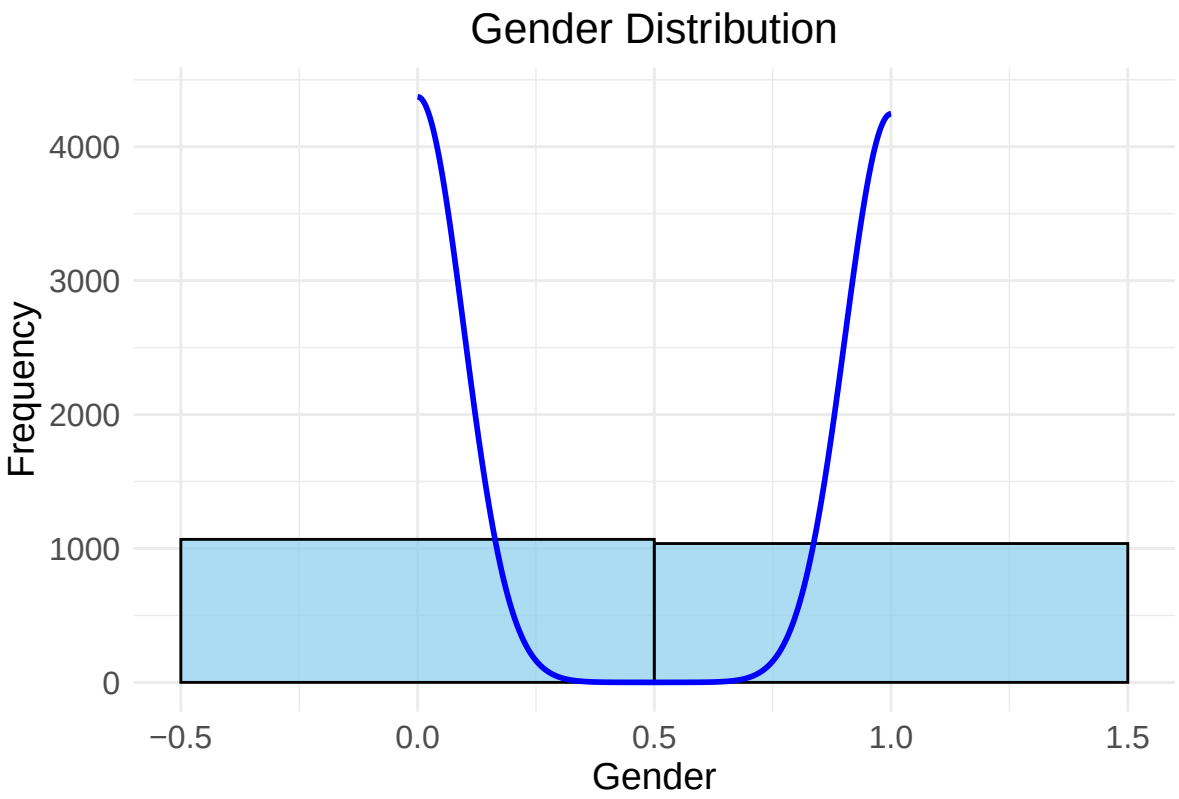
```
## Warning: `aes_string()` was deprecated in ggplot2 3.0.0.
## i Please use tidy evaluation idioms with `aes()`.
## i See also `vignette("ggplot2-in-packages")` for more information.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

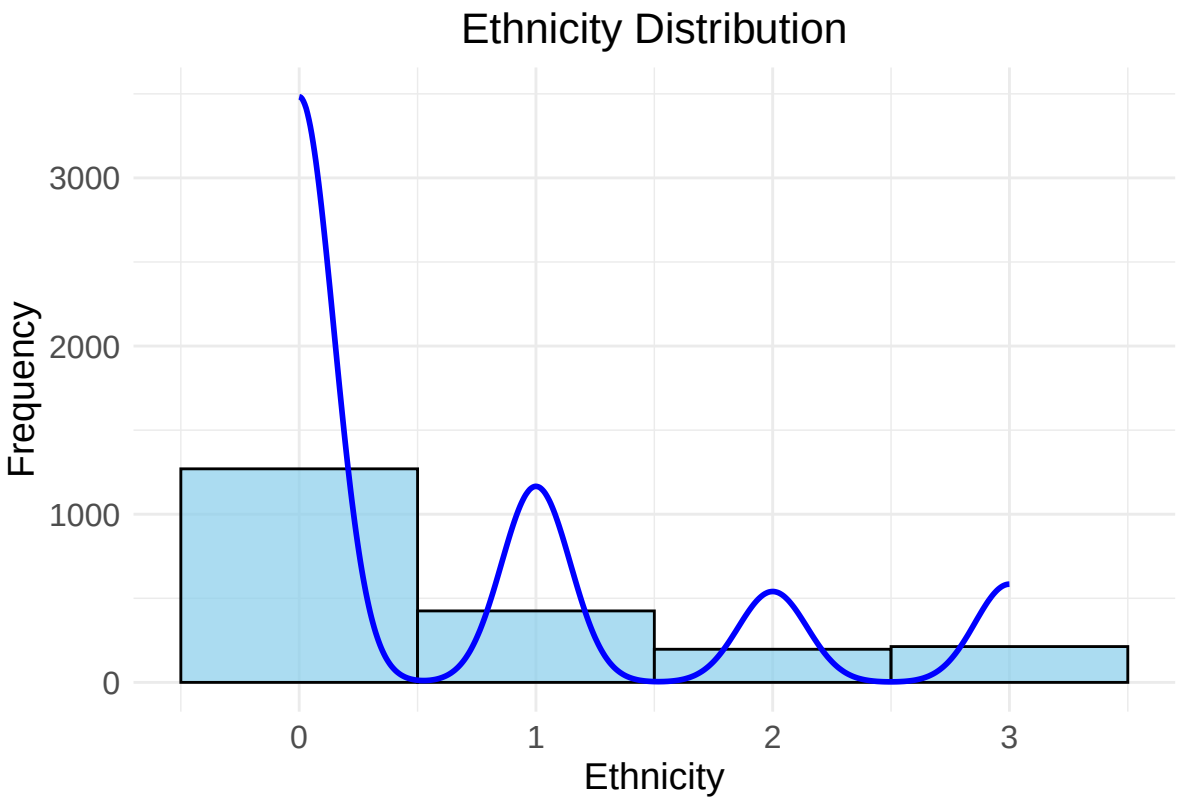
```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

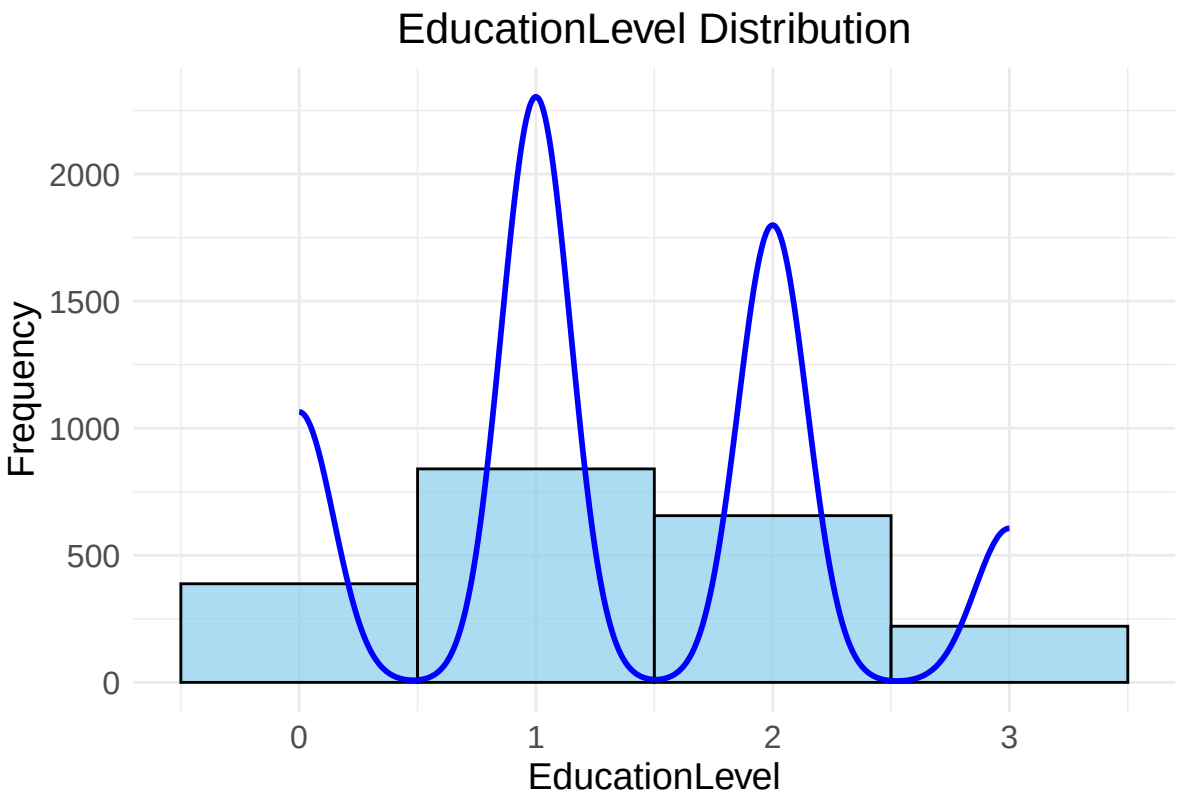
```
# Option 1: Display one plot at a time (for large datasets)
for (plot in histograms) {
  print(plot)
  Sys.sleep(1) # Pause to view each plot individually
}
```

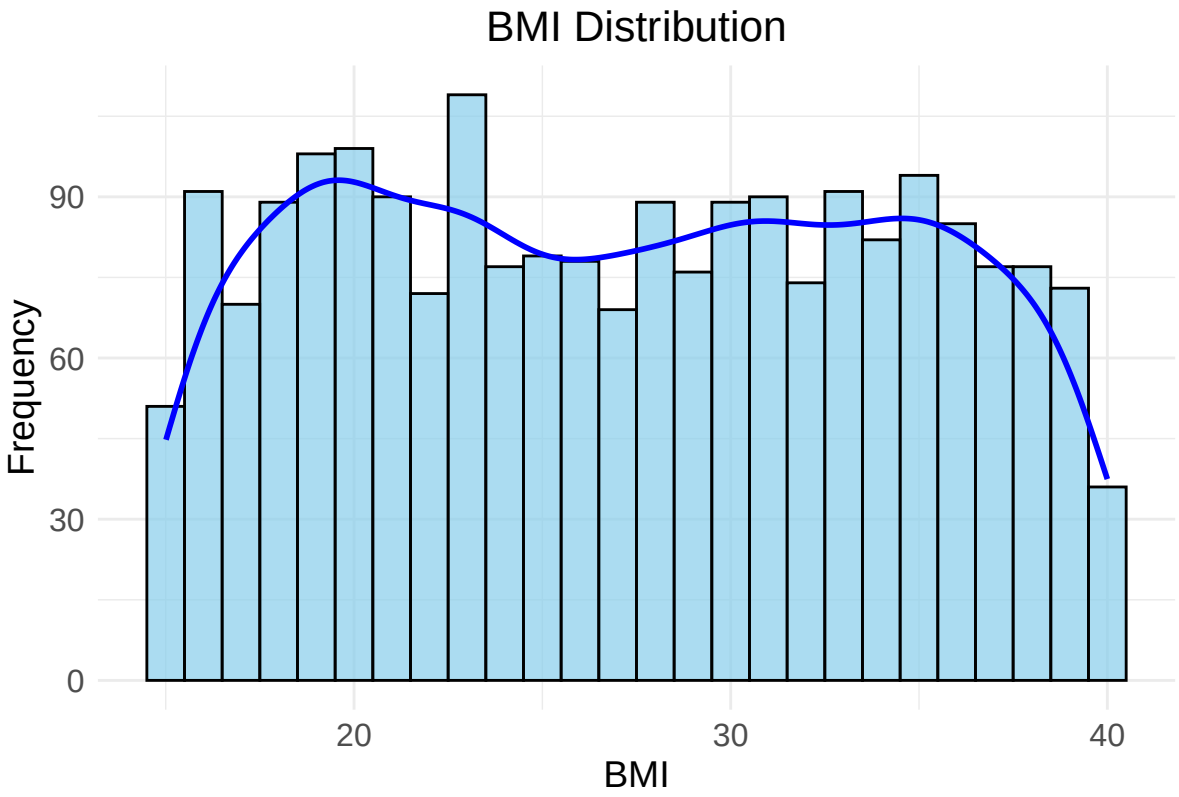

PatientID Distribution
1.00
0.75
0.50
0.25
0.00
Frequency
3000
3500
4000
4500
5000
PatientID

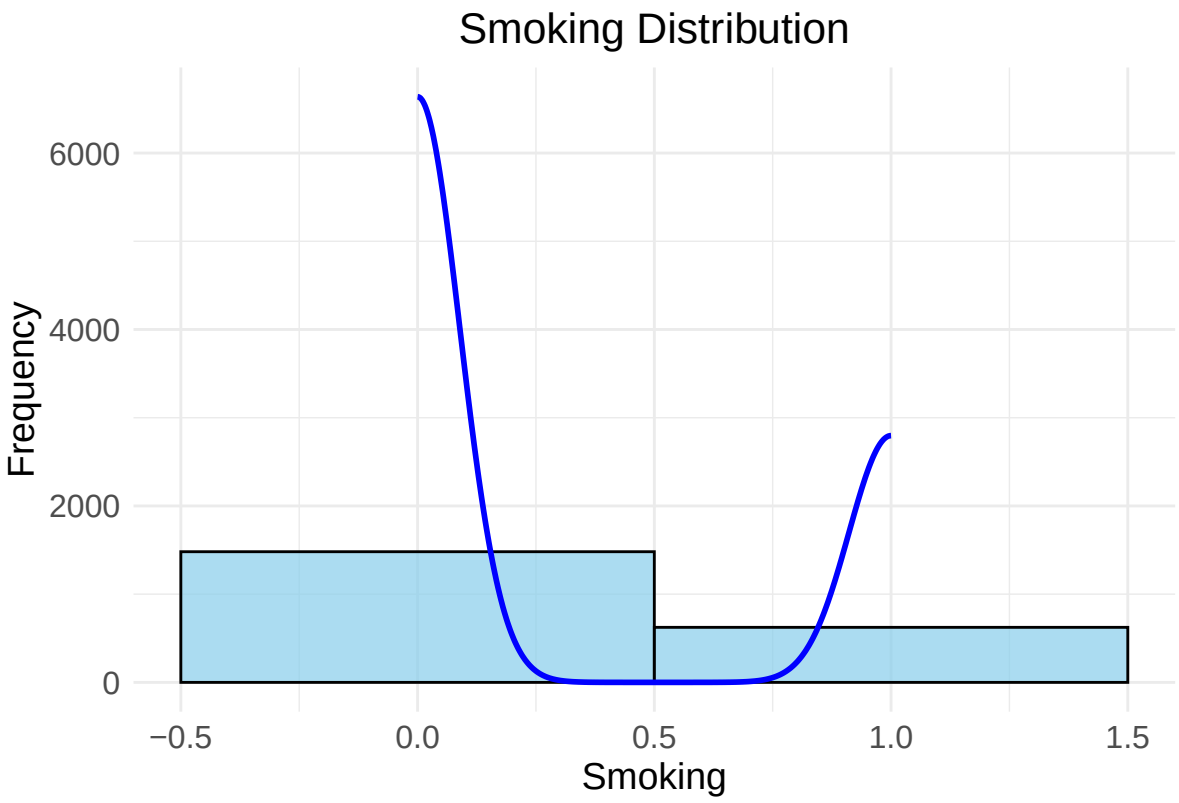

Age Distribution
Frequency
0
20
40
60
50
60
70
80
90
Age

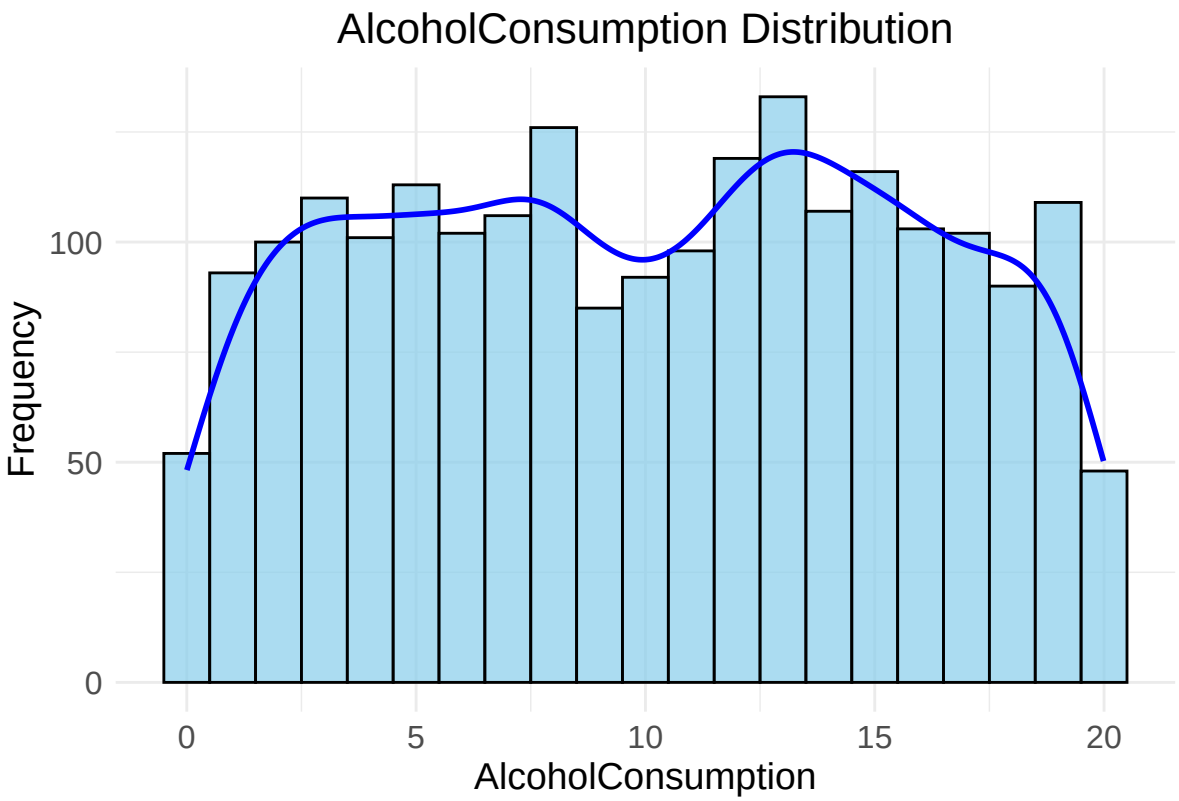

Gender Distribution
Frequency
4000
3000
2000
1000
0
−0.5
0.0
0.5
1.0
1.5
Gender

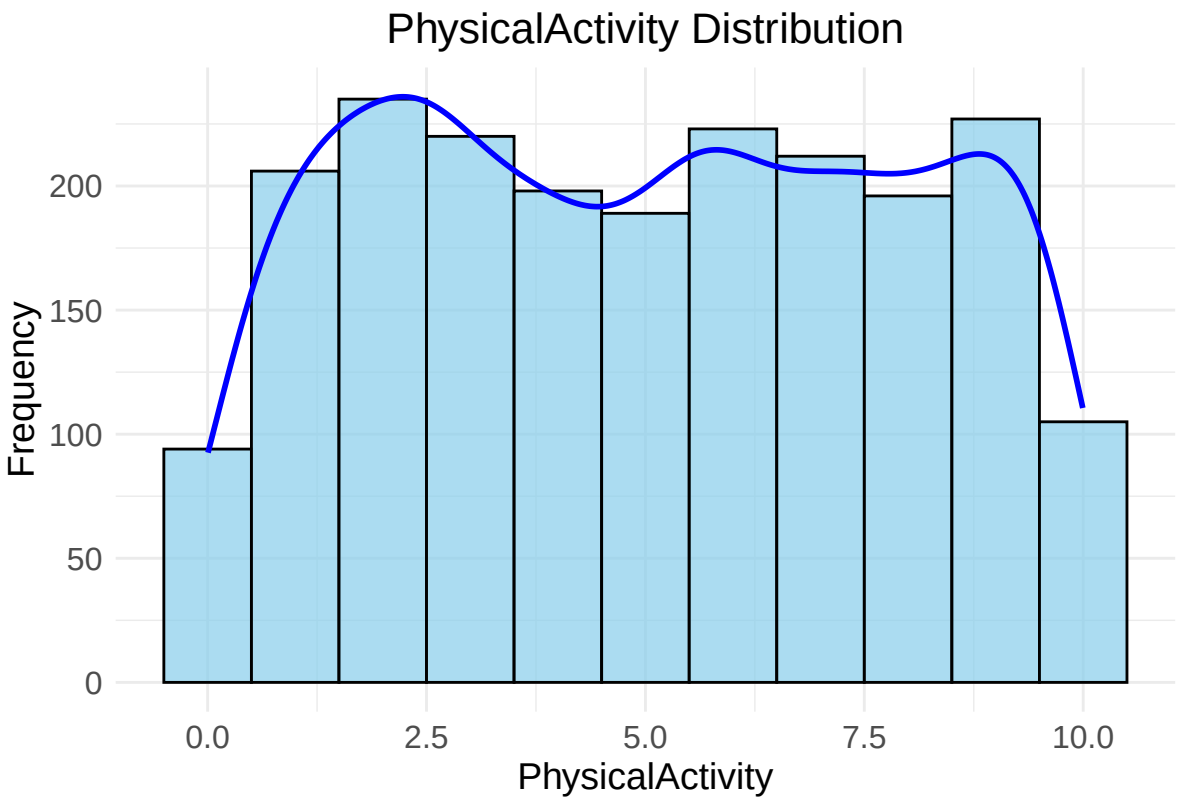

Ethnicity Distribution
Frequency
3000
2000
1000
0
0
1
2
3
Ethnicity

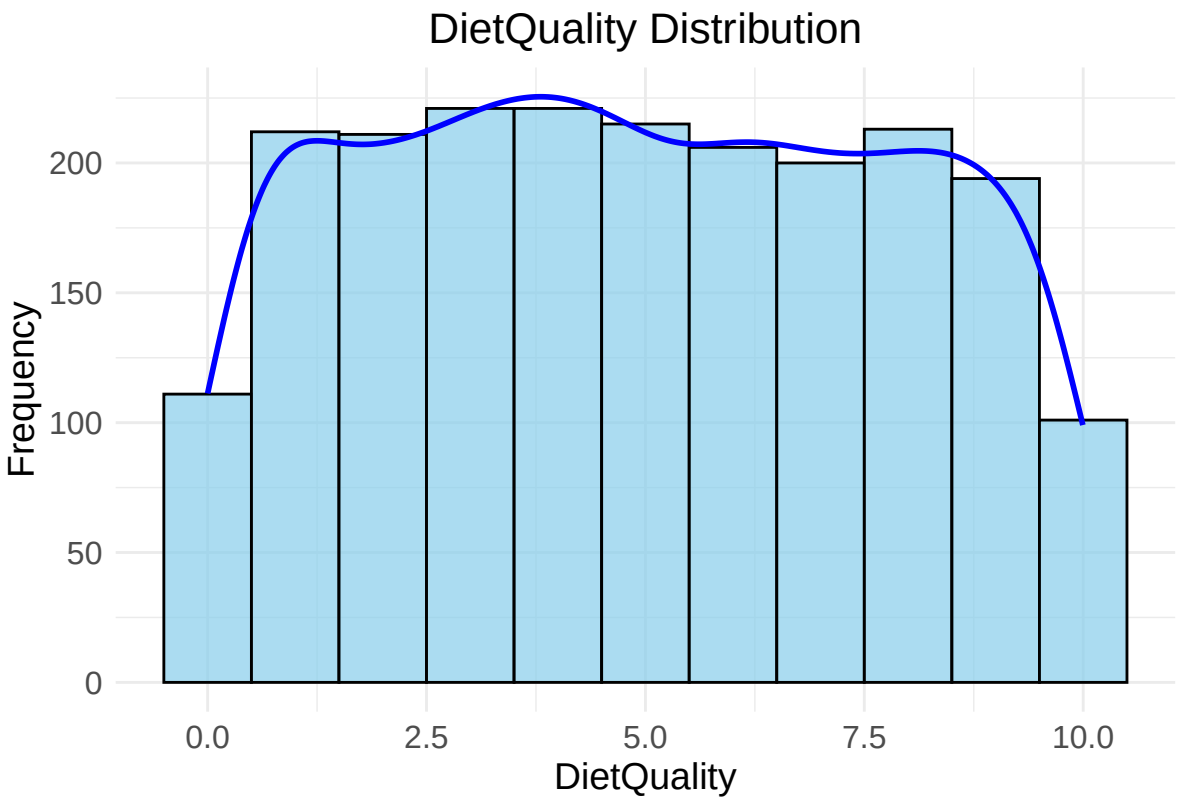

EducationLevel Distribution
Frequency
2000
1500
1000
500
0
0
1
2
3
EducationLevel

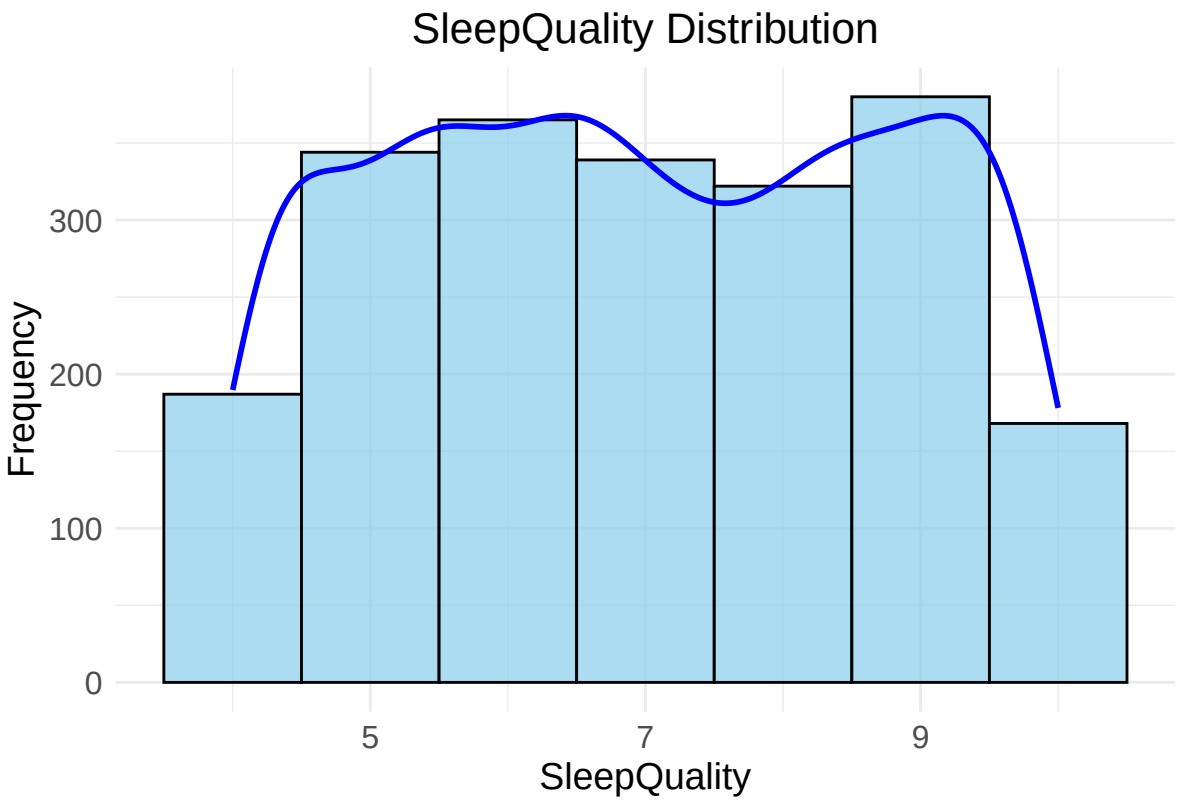

BMI Distribution
Frequency
90
60
30
0
20
30
40
BMI

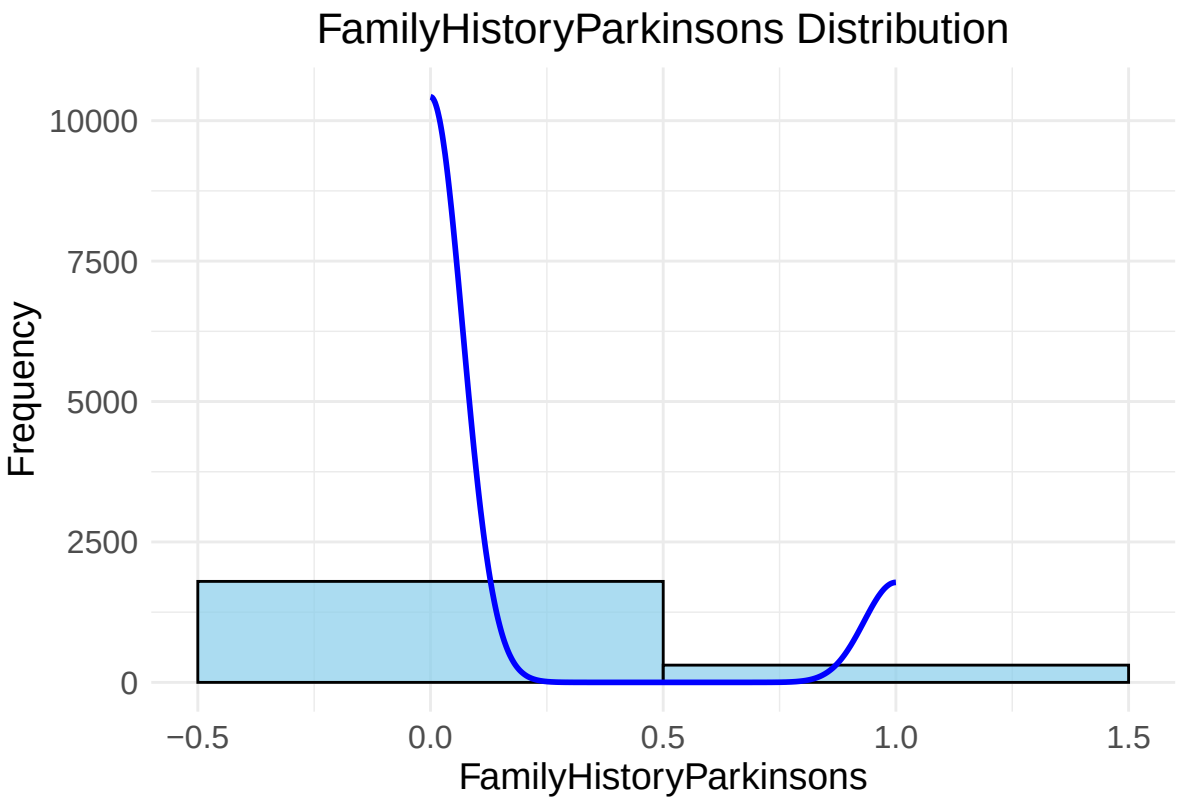

Smoking Distribution
Frequency
6000
4000
2000
0
−0.5
0.0
0.5
1.0
1.5
Smoking

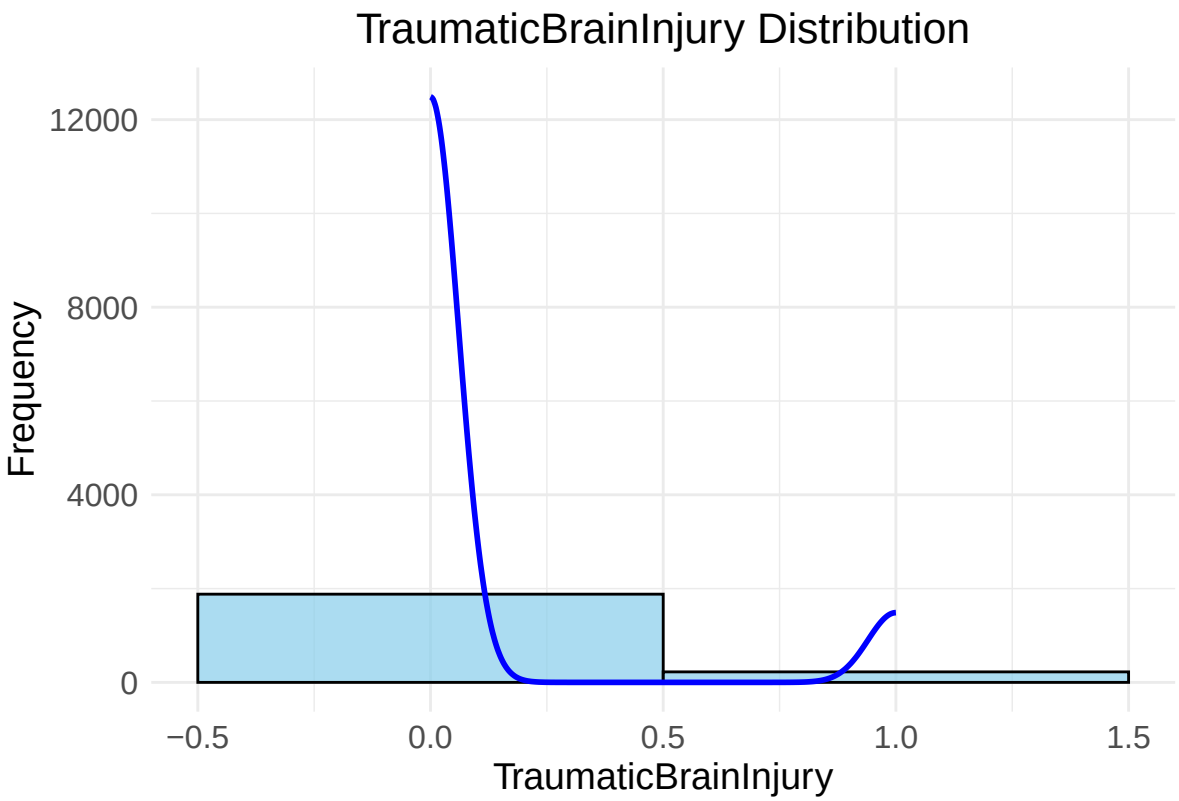

AlcoholConsumption Distribution
Frequency
100
50
0
0
5
10
15
20
AlcoholConsumption

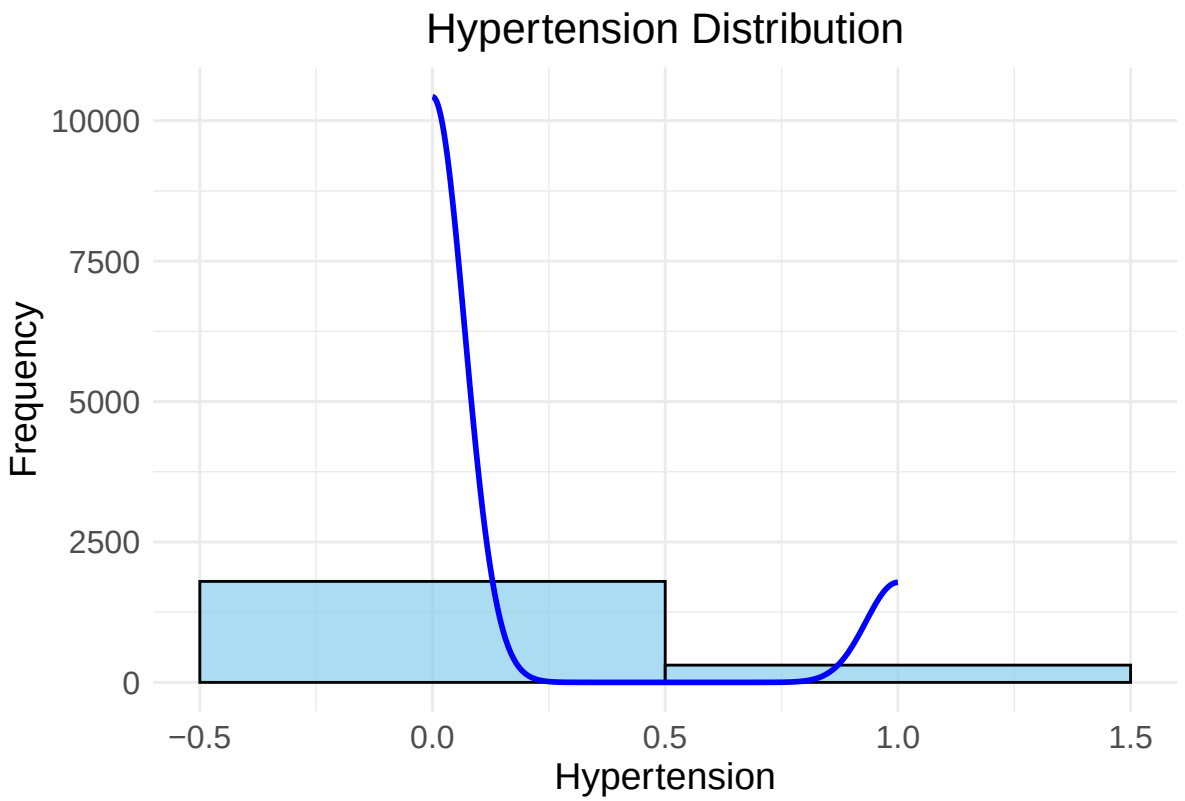

PhysicalActivity Distribution
Frequency
200
150
100
50
0
0.0
2.5
5.0
7.5
10.0
PhysicalActivity

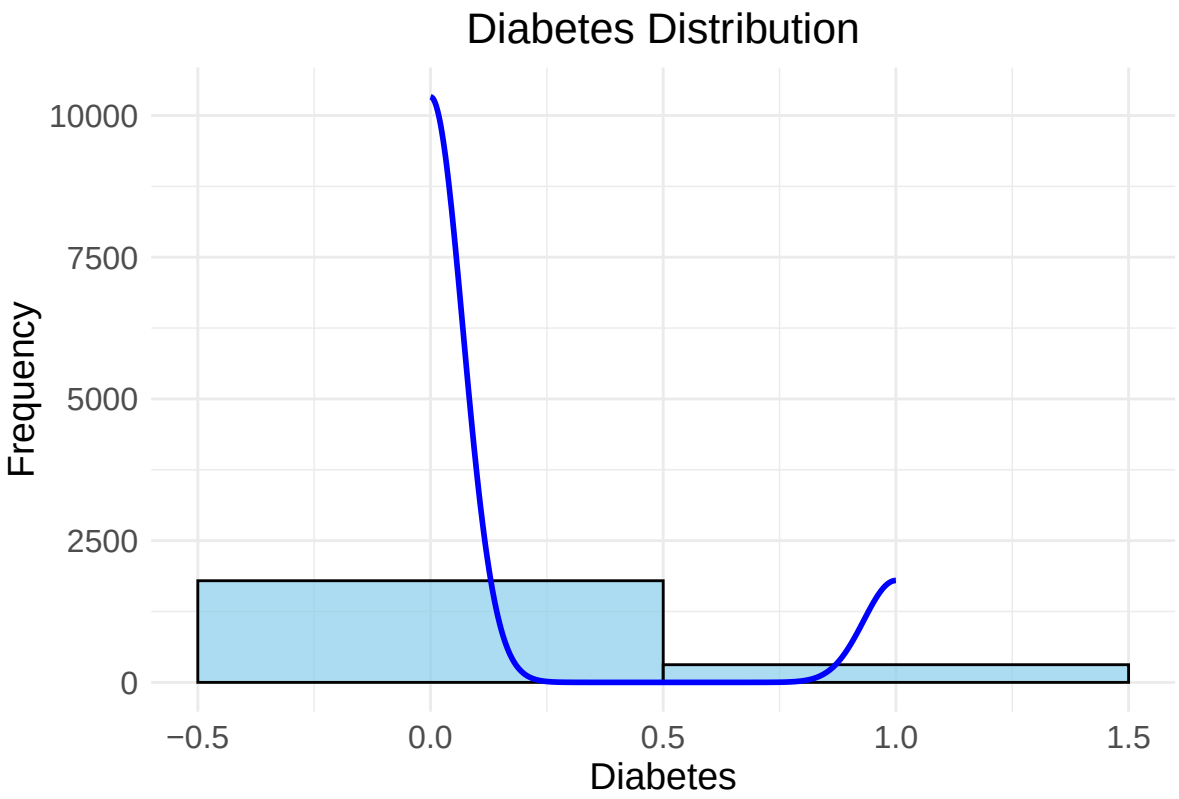

DietQuality Distribution
Frequency
200
150
100
50
0
0.0
2.5
5.0
7.5
10.0
DietQuality

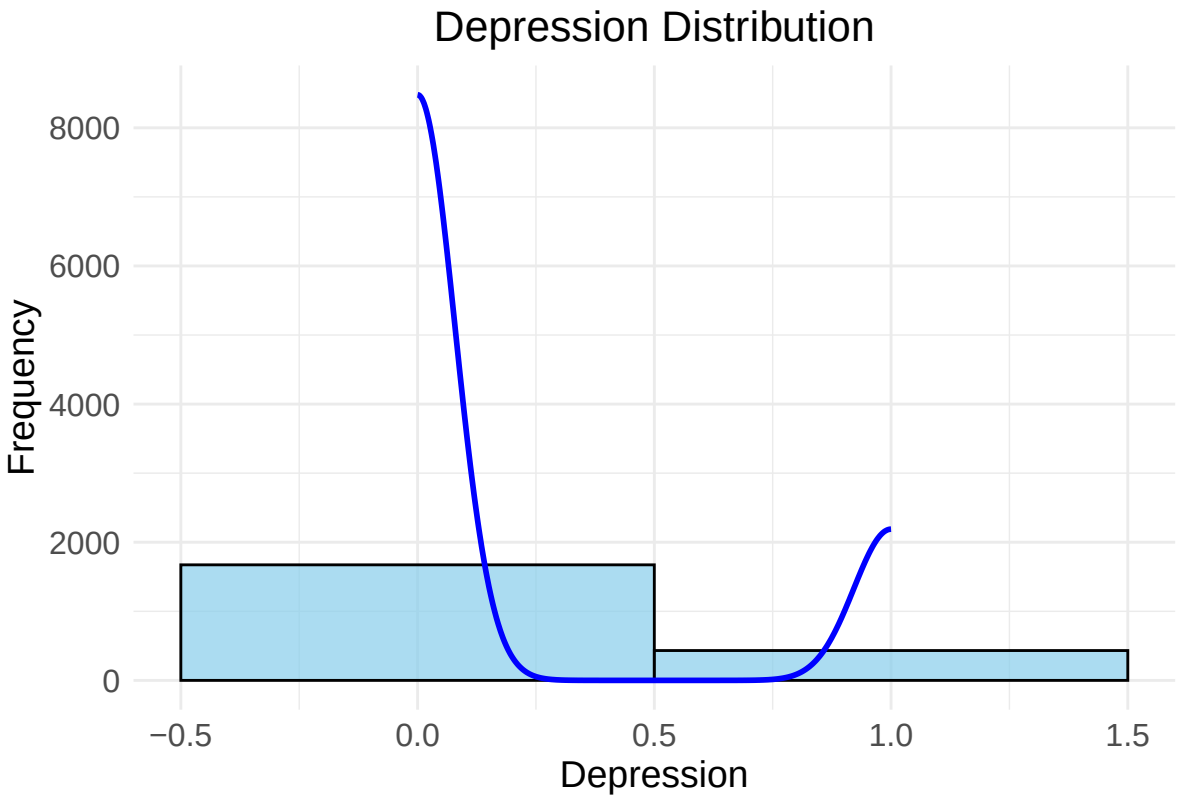

SleepQuality Distribution
Frequency
300
200
100
0
5
7
9
SleepQuality

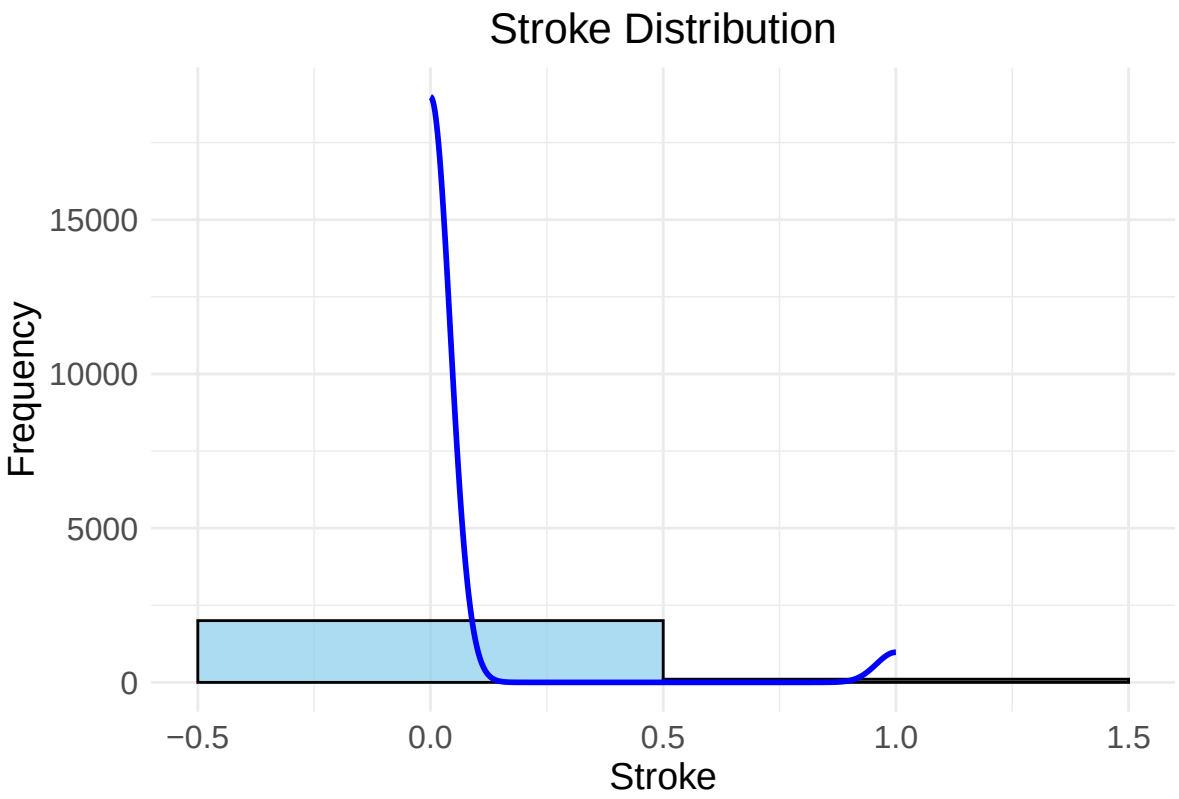

FamilyHistoryParkinsons Distribution
Frequency
10000
7500
5000
2500
0
−0.5
0.0
0.5
1.0
1.5
FamilyHistoryParkinsons

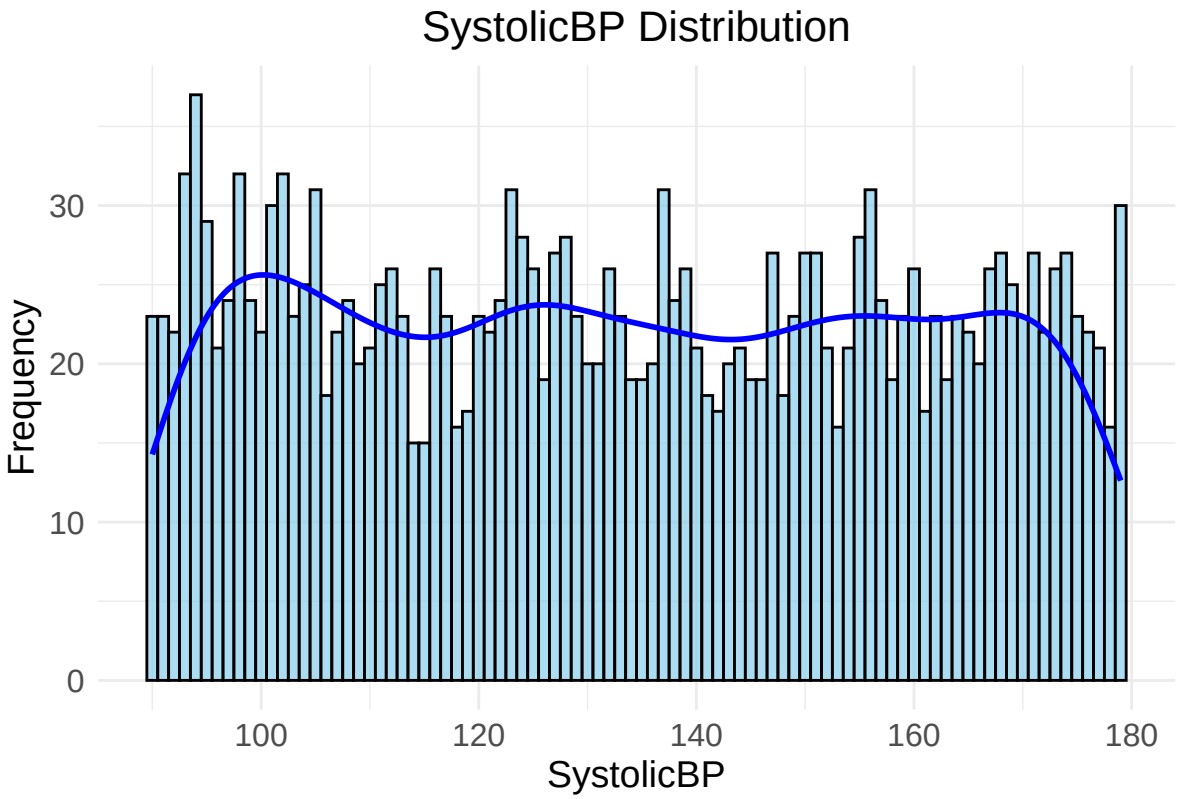

TraumaticBrainInjury Distribution
12000
8000
4000
0
Frequency
−0.5
0.0
0.5
1.0
1.5
TraumaticBrainInjury

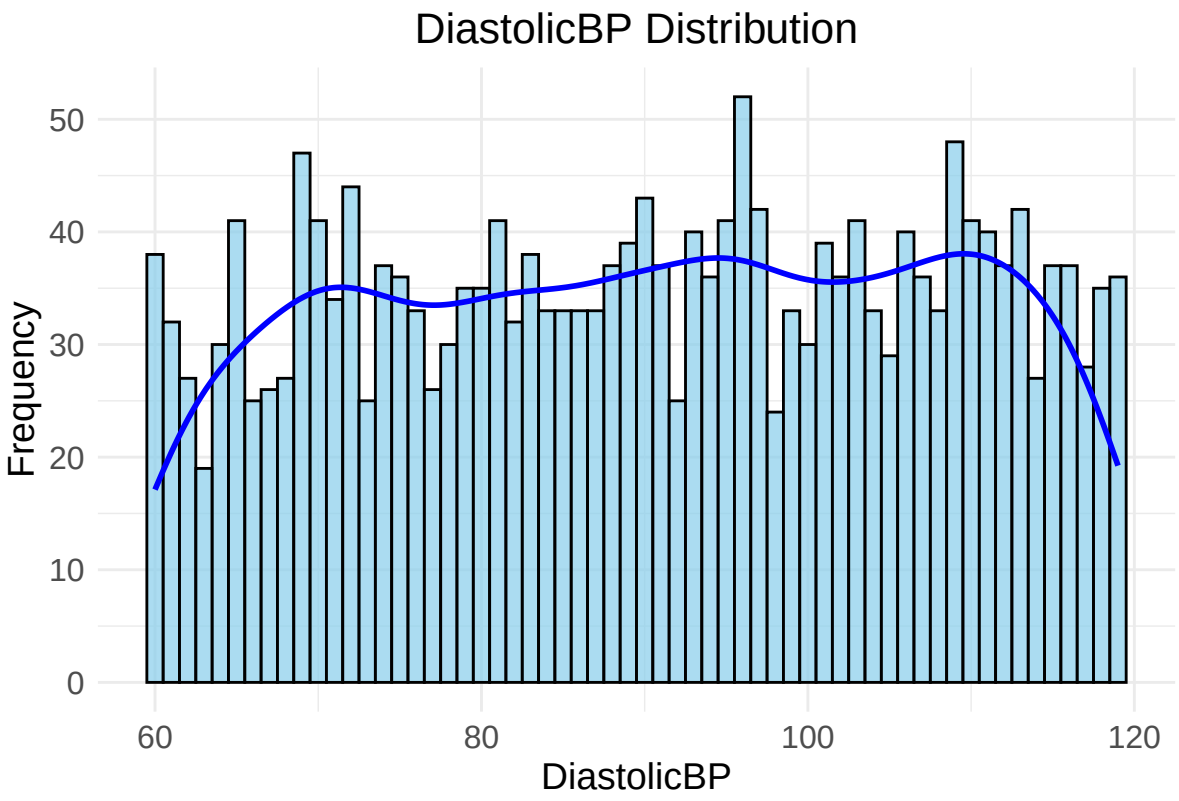

Hypertension Distribution
Frequency
10000
7500
5000
2500
0
−0.5
0.0
0.5
1.0
1.5
Hypertension

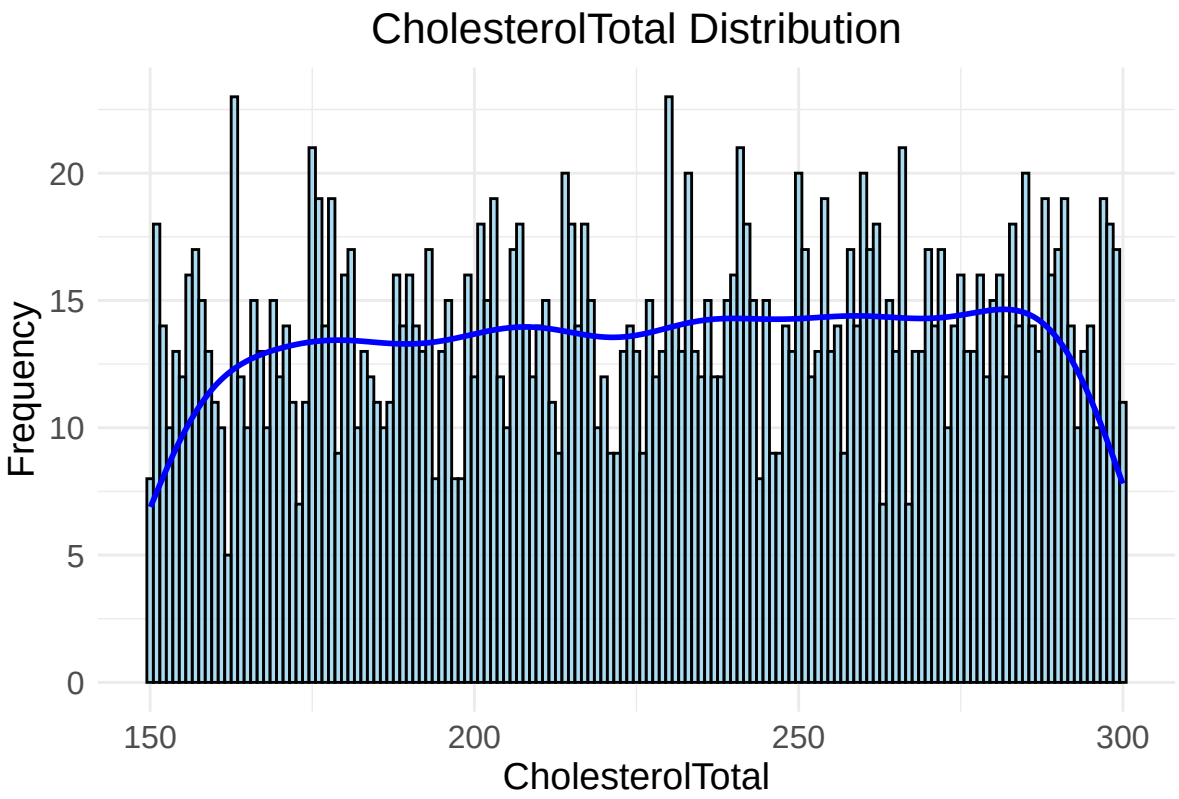

Diabetes Distribution
Frequency
10000
7500
5000
2500
0
−0.5
0.0
0.5
1.0
1.5
Diabetes

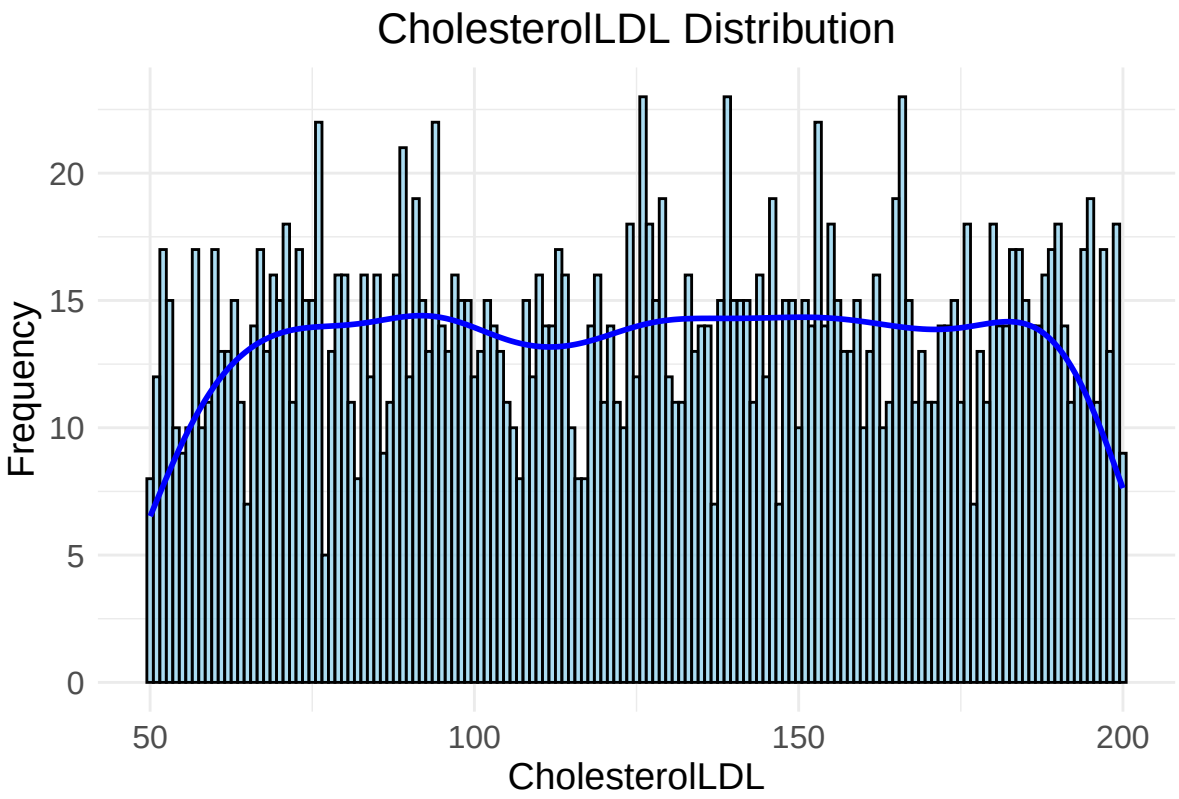

Depression Distribution
8000
6000
4000
2000
0
Frequency
−0.5
0.0
0.5
1.0
1.5
Depression

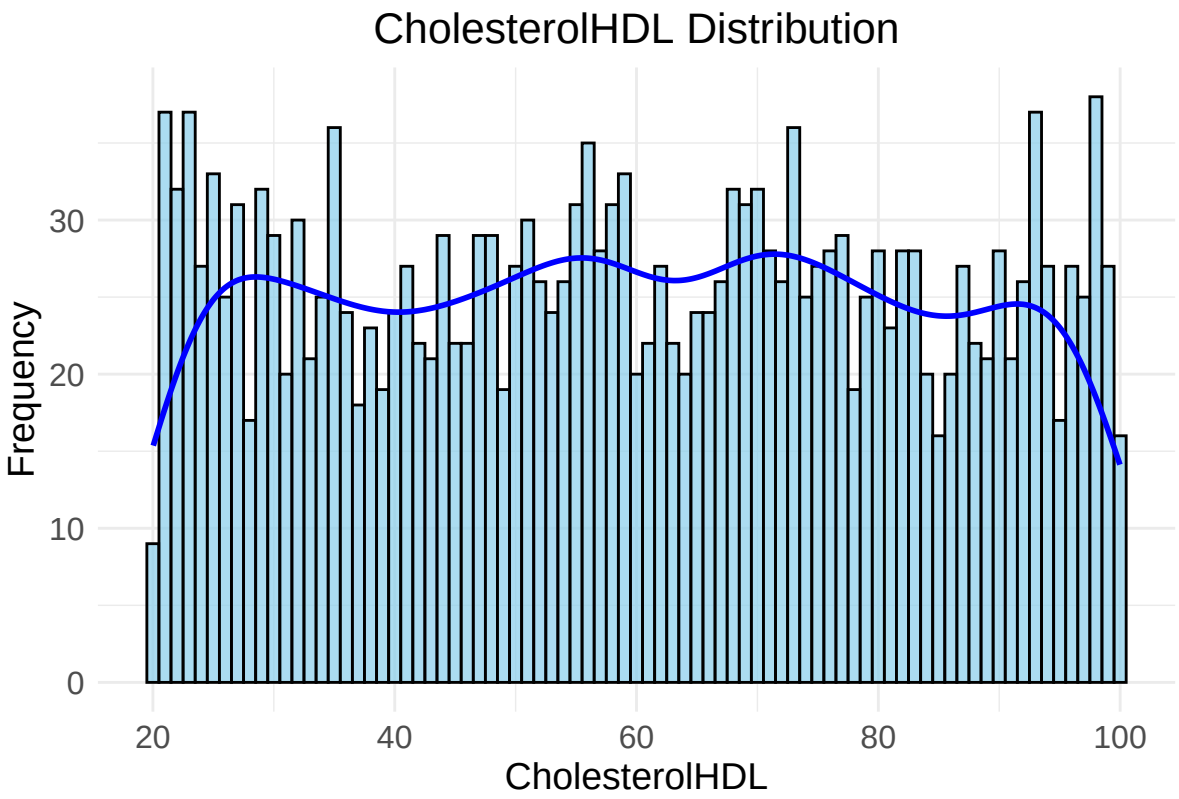

Stroke Distribution
Frequency
15000
10000
5000
0
−0.5
0.0
0.5
1.0
1.5
Stroke

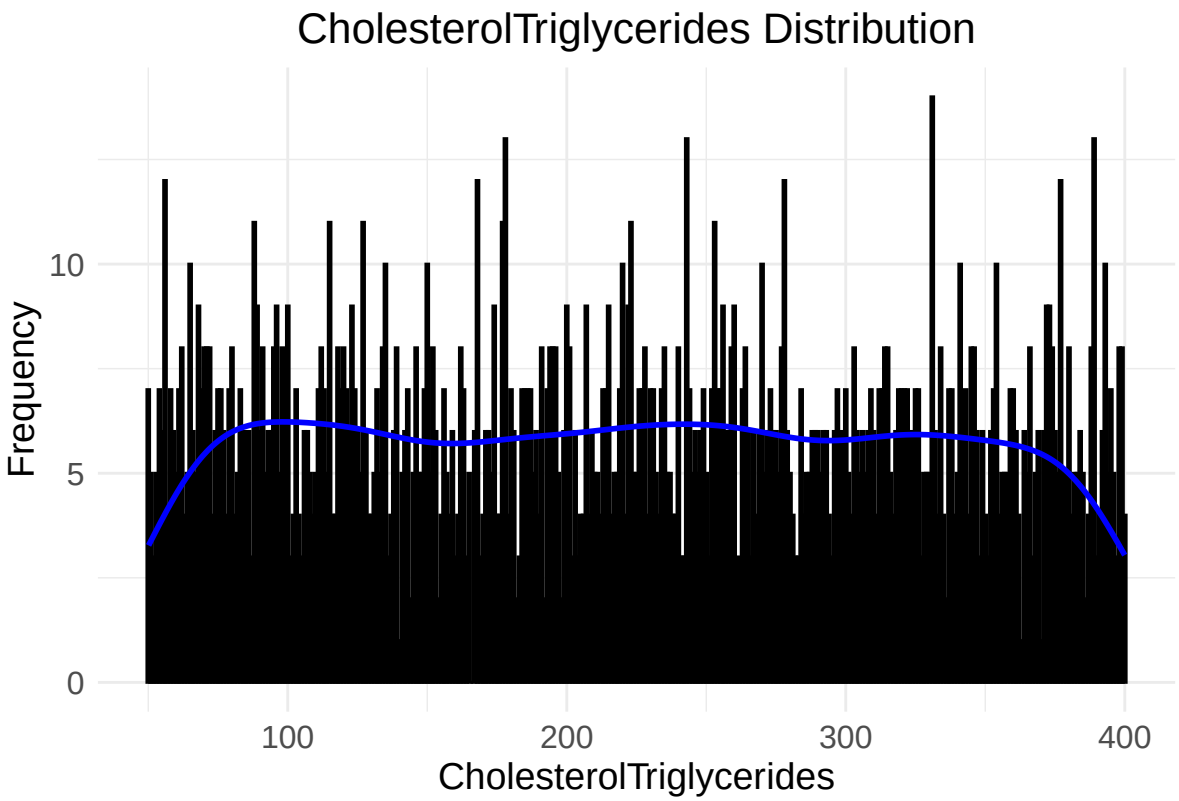

SystolicBP Distribution
Frequency
0
10
20
30
100
120
140
160
180
SystolicBP

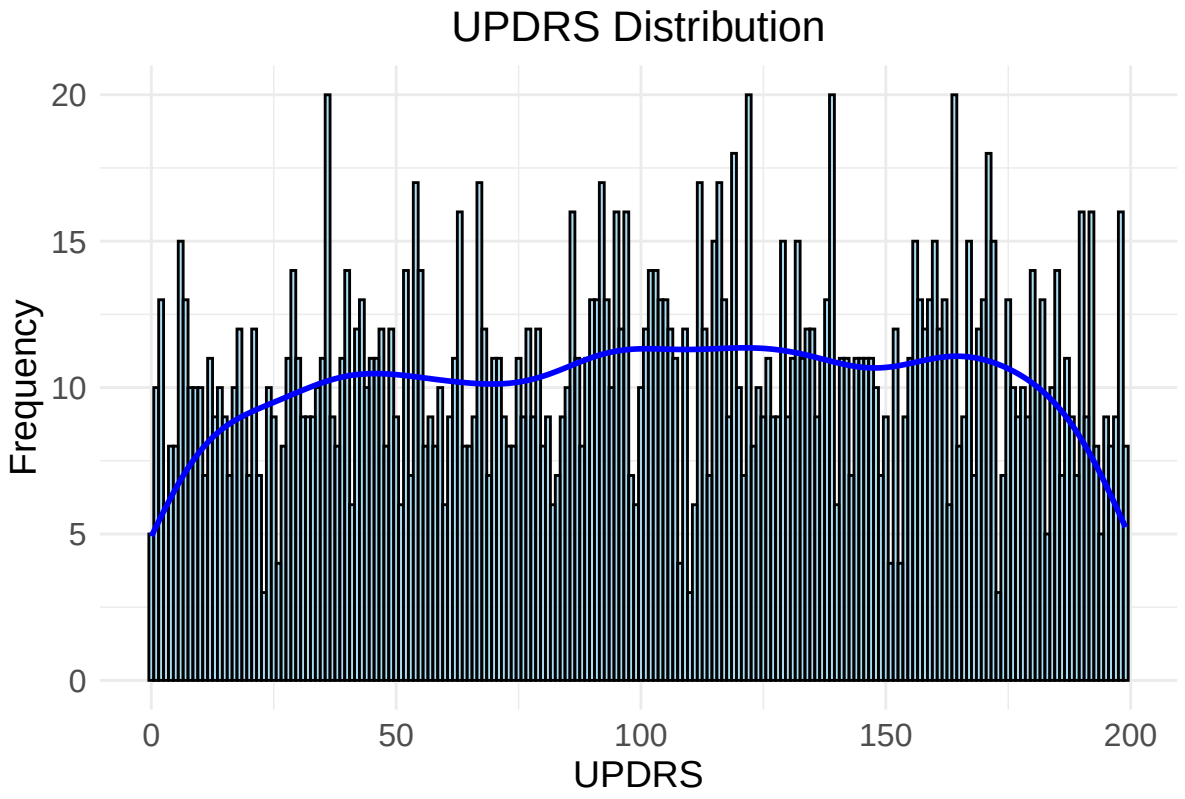

DiastolicBP Distribution
Frequency
0
10
20
30
40
50
60
80
100
120
DiastolicBP

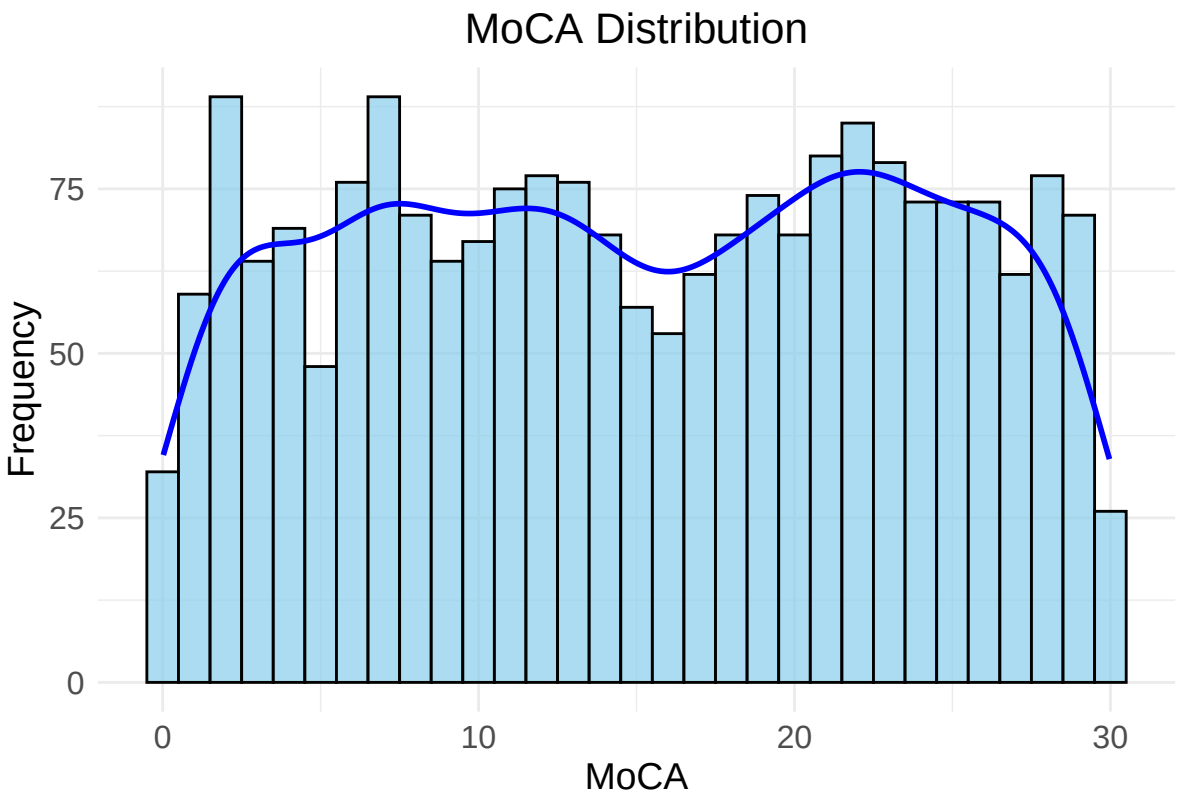

CholesterolTotal Distribution
Frequency
0
5
10
15
20
150
200
250
300
CholesterolTotal

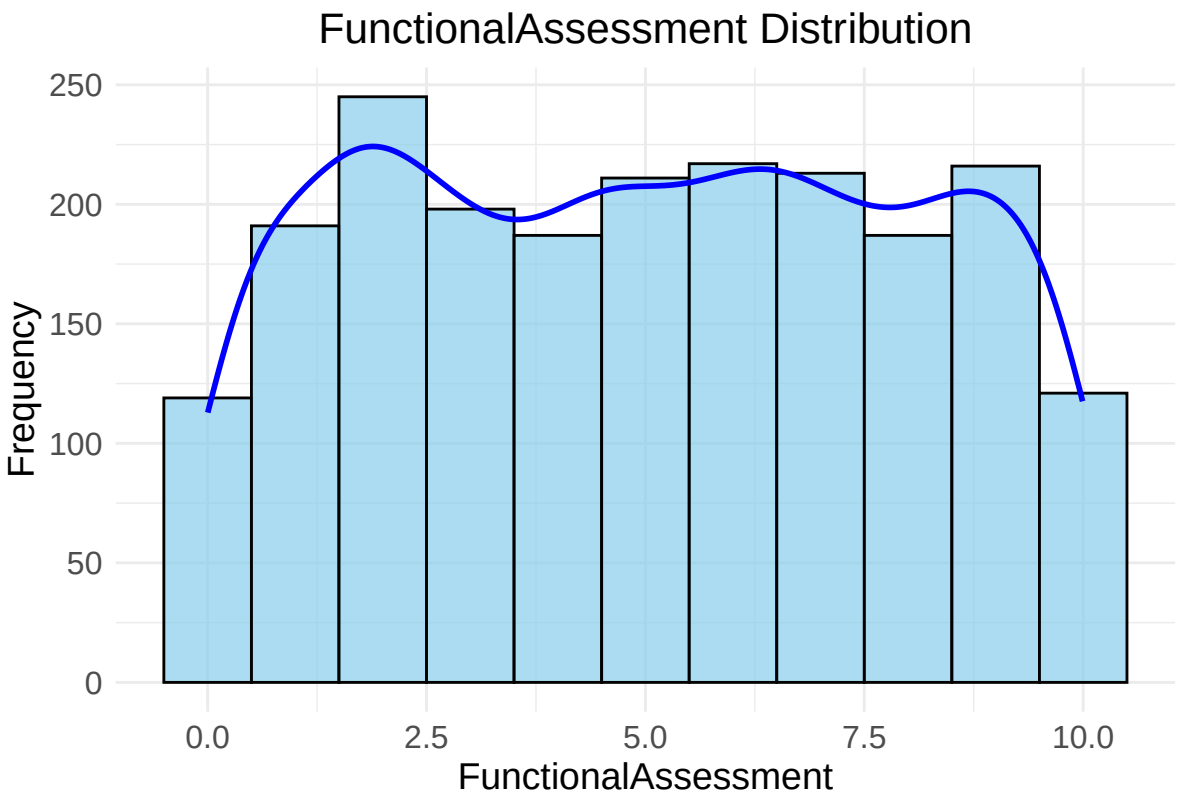

CholesterolLDL Distribution
Frequency
0
5
10
15
20
50
100
150
200
CholesterolLDL

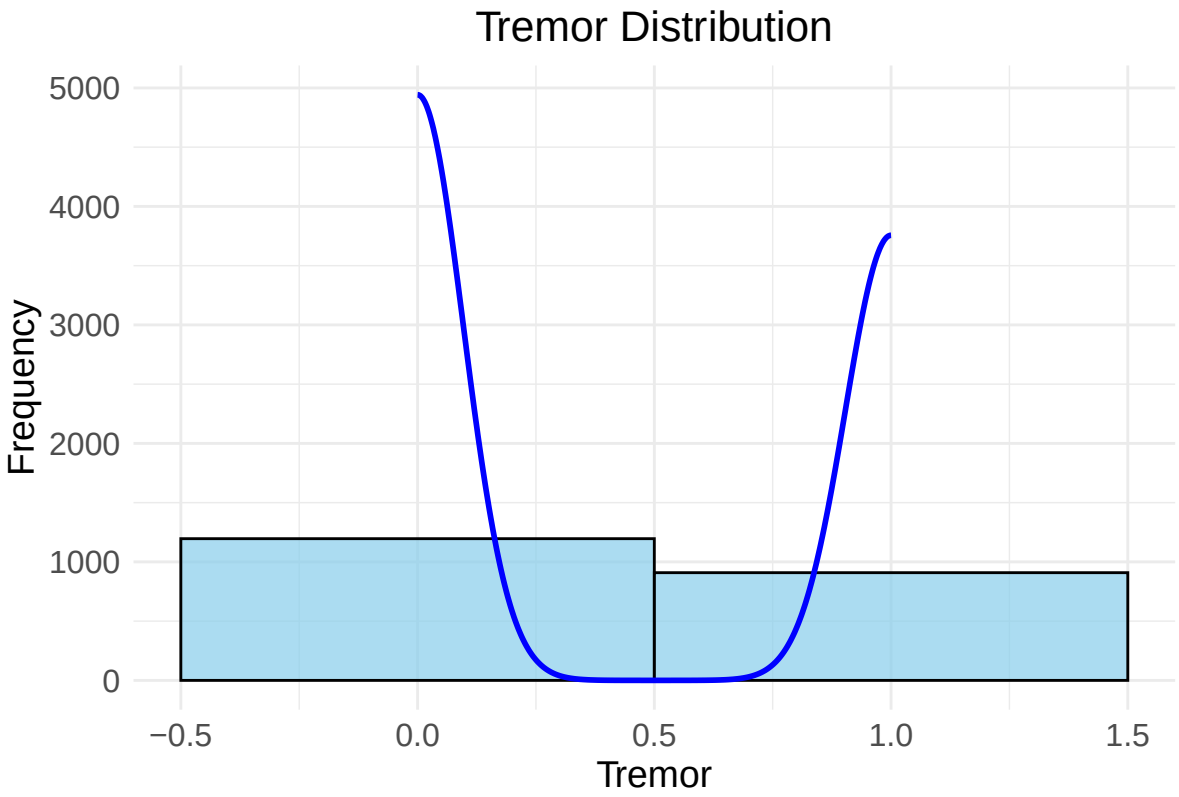

CholesterolHDL Distribution
Frequency
0
10
20
30
20
40
60
80
100
CholesterolHDL

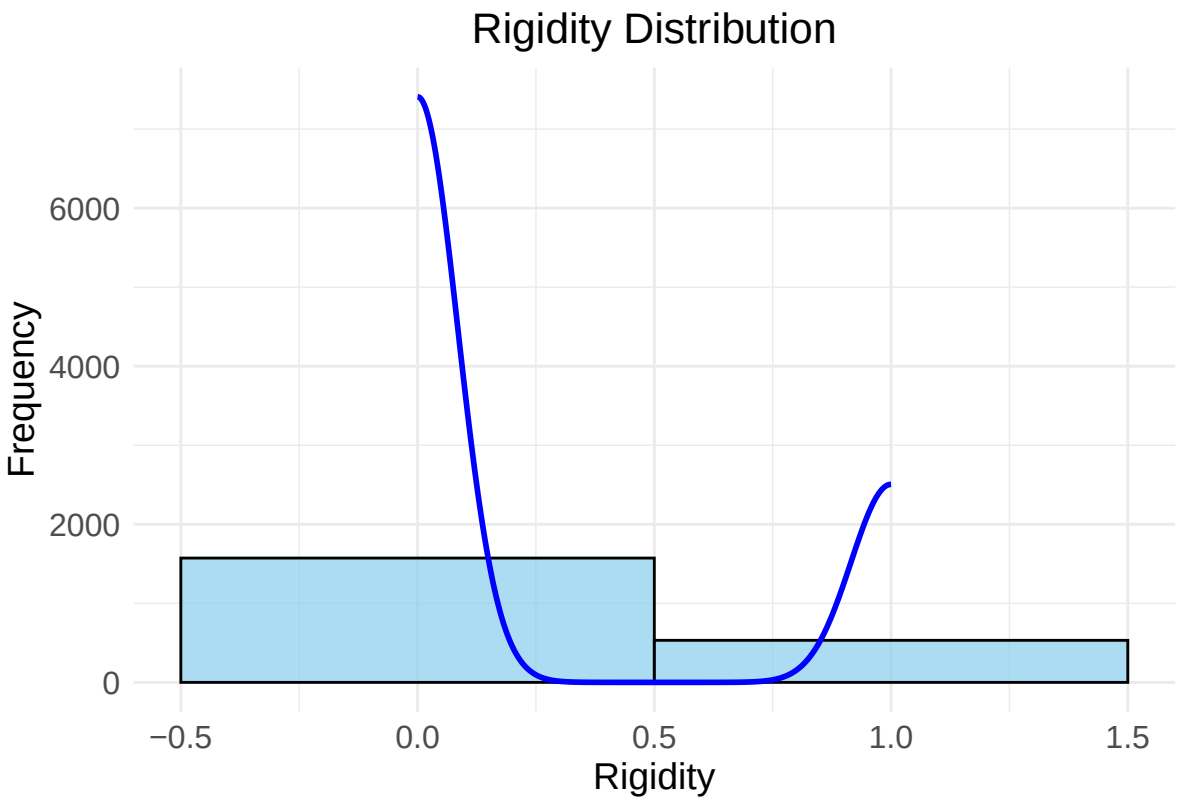

CholesterolTriglycerides Distribution
Frequency
10
5
0
100
200
300
400
CholesterolTriglycerides

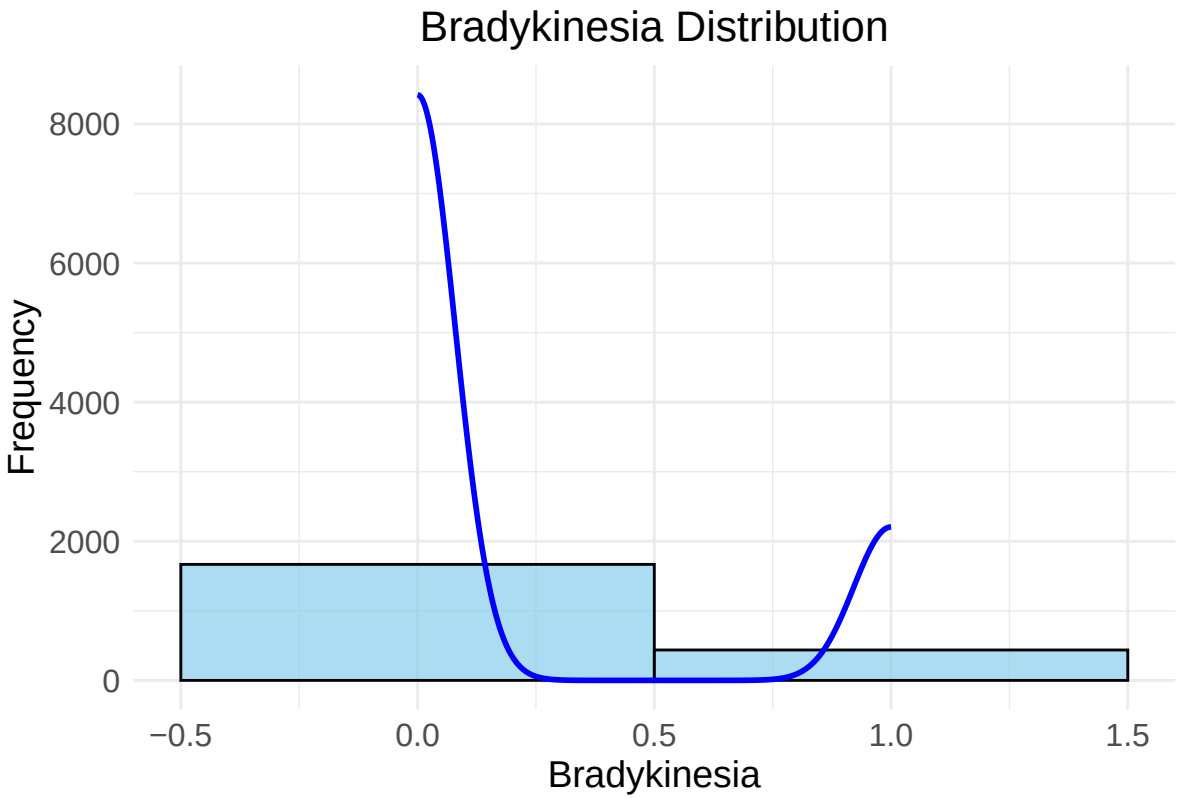

UPDRS Distribution
Frequency
20
15
10
5
0
0
50
100
150
200
UPDRS

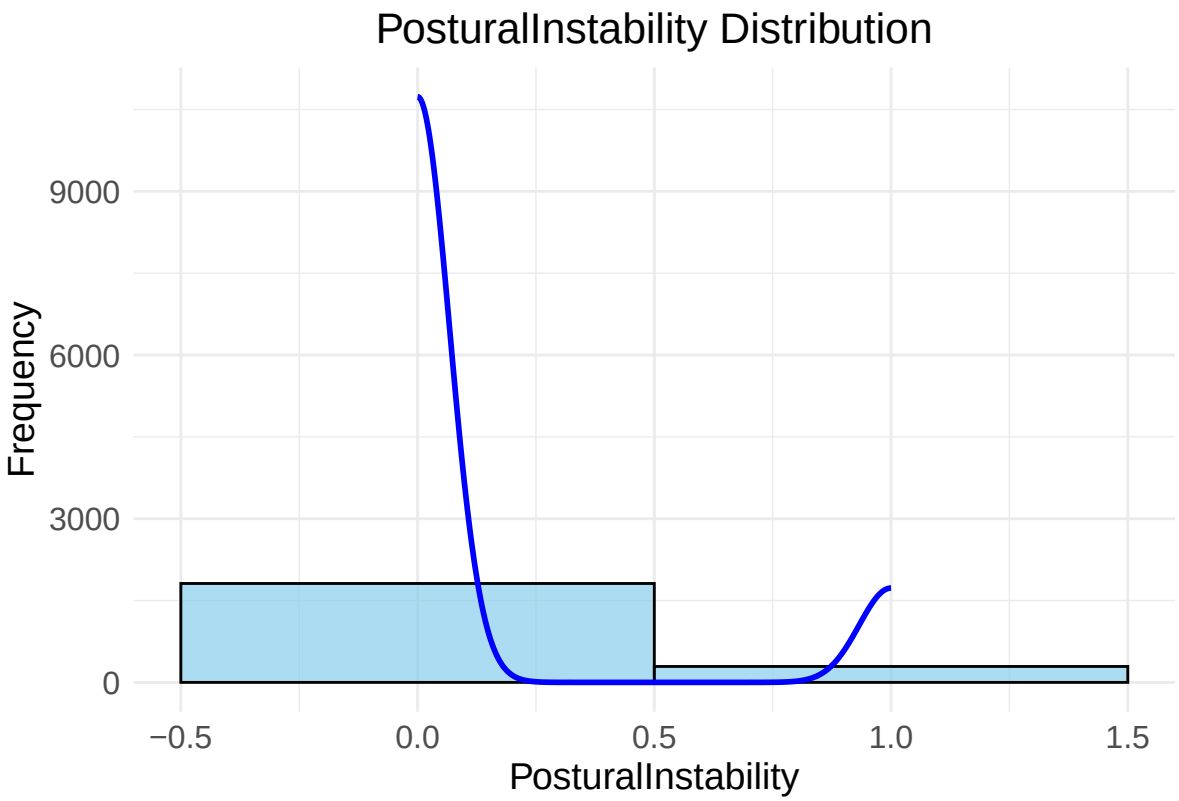

MoCA Distribution
Frequency
75
50
25
0
0
10
20
30
MoCA

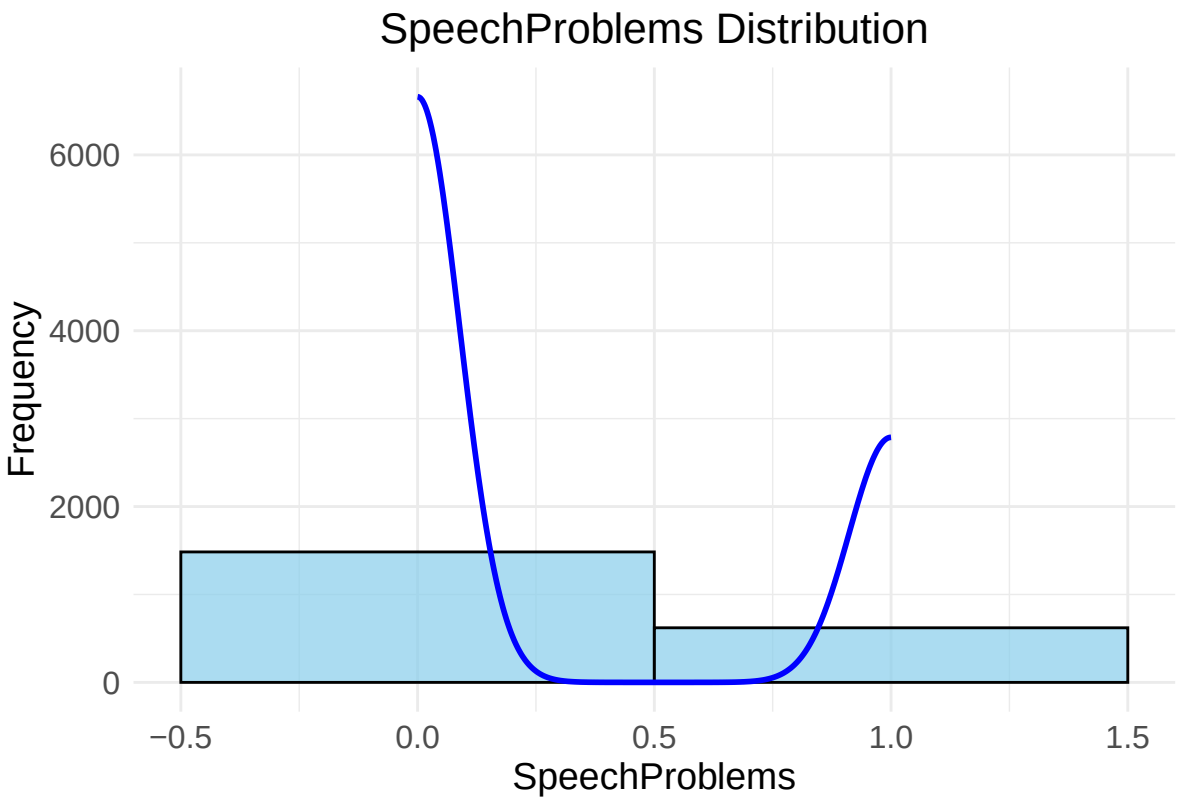

FunctionalAssessment Distribution
Frequency
250
200
150
100
50
0
0.0
2.5
5.0
7.5
10.0
FunctionalAssessment

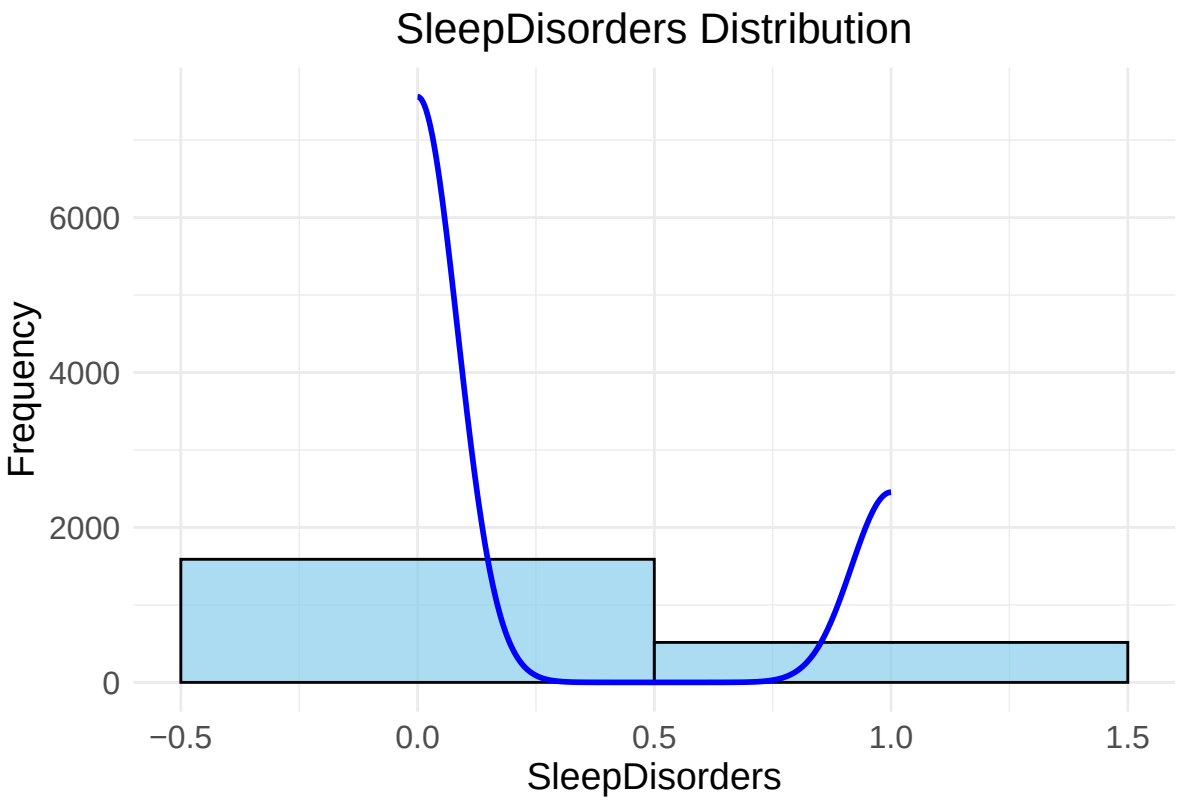

Tremor Distribution
Frequency
5000
4000
3000
2000
1000
0
−0.5
0.0
0.5
1.0
1.5
Tremor

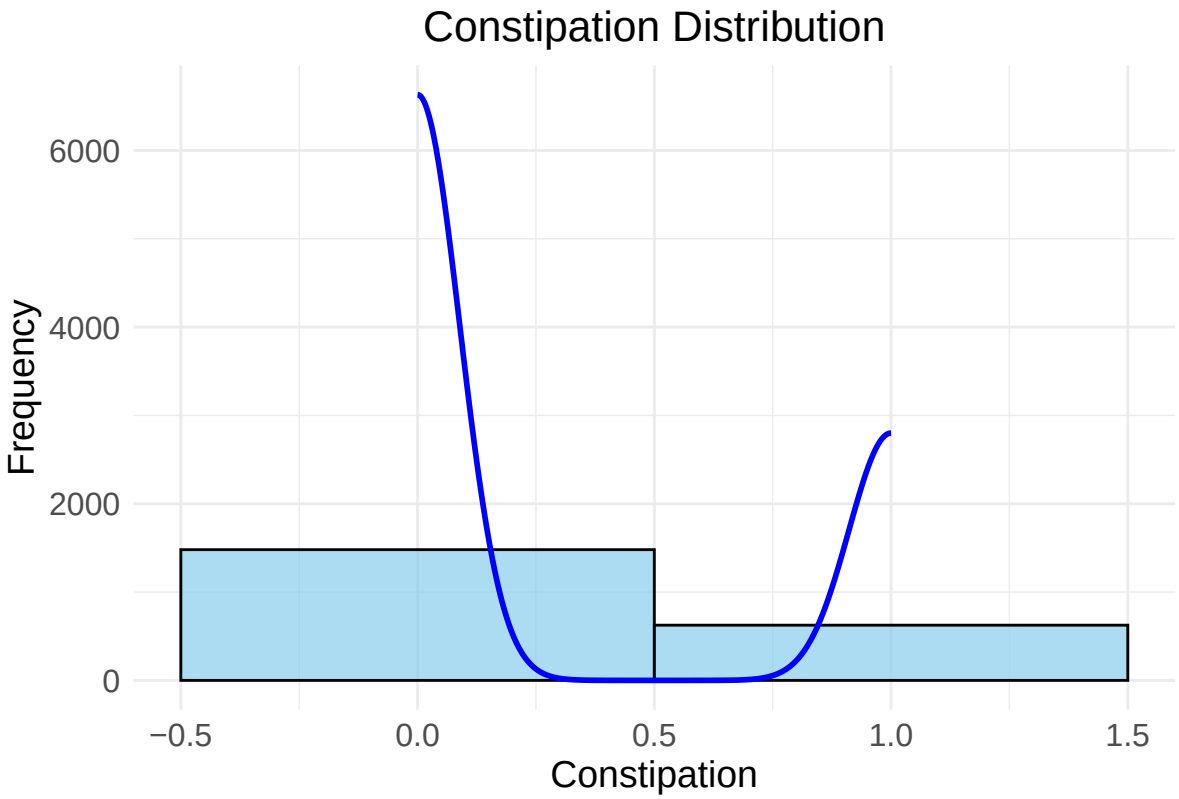

Rigidity Distribution
Frequency
6000
4000
2000
0
−0.5
0.0
0.5
1.0
1.5
Rigidity

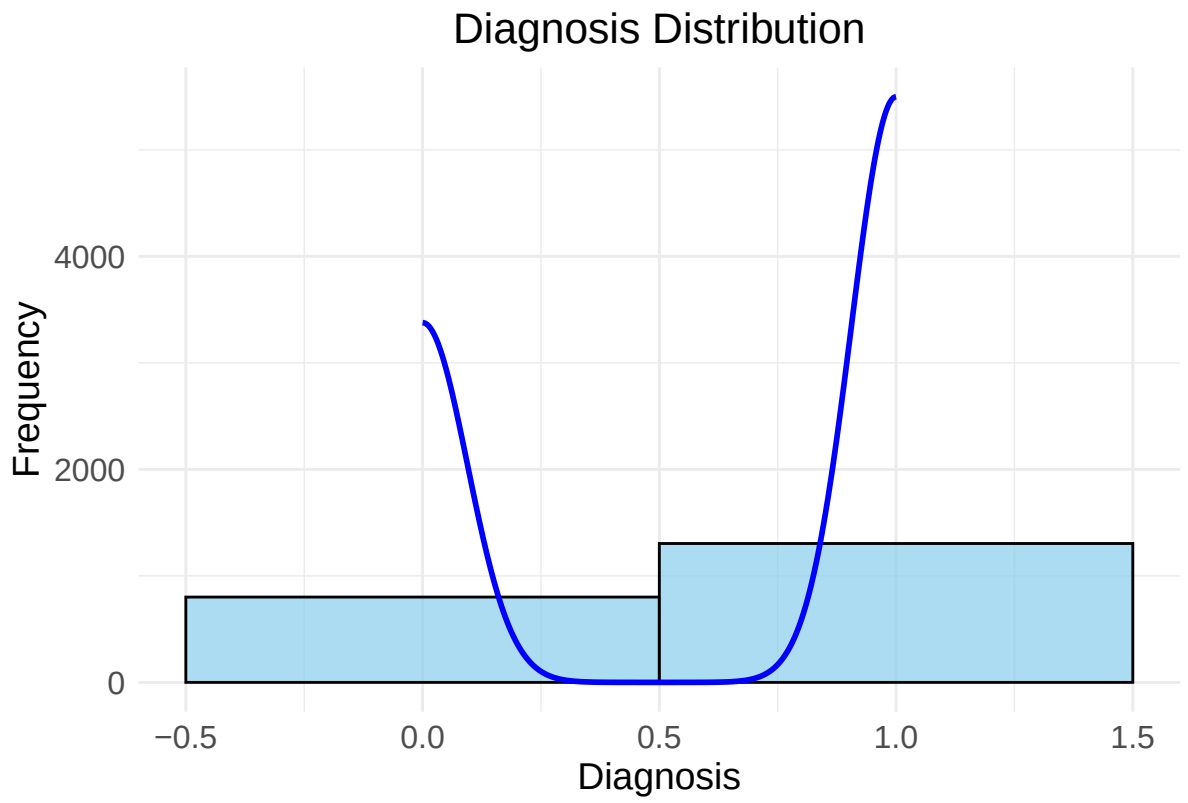

Bradykinesia Distribution
Frequency
8000
6000
4000
2000
0
−0.5
0.0
0.5
1.0
1.5
Bradykinesia


PosturalInstability Distribution
Frequency
9000
6000
3000
0
−0.5
0.0
0.5
1.0
1.5
PosturalInstability


SpeechProblems Distribution
Frequency
6000
4000
2000
0
−0.5
0.0
0.5
1.0
1.5
SpeechProblems


SleepDisorders Distribution
Frequency
6000
4000
2000
0
−0.5
0.0
0.5
1.0
1.5
SleepDisorders


Constipation Distribution
Frequency
6000
4000
2000
0
−0.5
0.0
0.5
1.0
1.5
Constipation



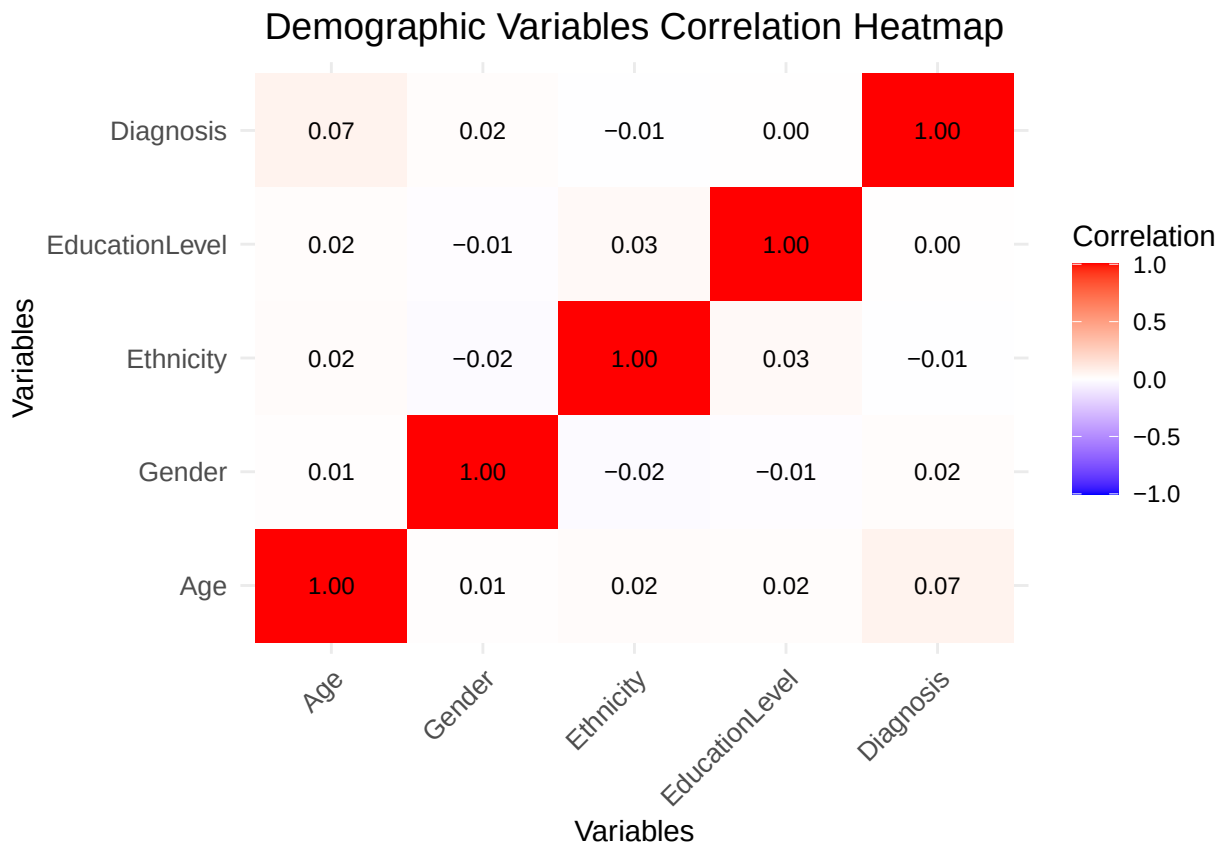
The average age of patients with Parkinson's Disease is 70 years, aligning with the disease's typical onset in middle or late life, with an average diagnosis age of 60. Most patients with Parkinson's Disease are likely Caucasian, with Asians having the lowest likelihood of being affected. Patients who completed high school or a bachelor's degree are more likely to have Parkinson's Disease, although there seems to be no strong correlation between education level and disease likelihood. Despite the BMI column having a uniform distribution, the average BMI of Parkinson's Disease patients is 27. Observing the graph suggests that many patients tend to consume more alcohol. Physical activity or exercise helps manage Parkinson's symptoms and improves quality of life, with at least 2.5 hours per week recommended by the Parkinson's Foundation. The average diet quality of patients is generally below average, also showing a uniform distribution. Sleep quality, like diet quality, follows a uniform distribution, with most patients not reporting any family history of Parkinson's Disease. Most patients do not have traumatic brain injury, hypertension, diabetes, depression, or stroke. Six clinical measurements also show uniform distribution patterns. The Unified Parkinson's Disease Rating Scale (UPDRS) scores range from 0 to 199, with higher scores indicating greater disease severity; most patients have high UPDRS scores, reflecting severe conditions. The MoCA scores, ranging from 0 to 30, indicate cognitive function, with lower scores suggesting impairment, and most patients with Parkinson's Disease have very low scores. Functional assessment scores range from 0 to 10, with lower scores indicating greater impairment; most patients with Parkinson's Disease have very low scores, indicating significant impairment.

*Correlation Heatmaps for each cluster:

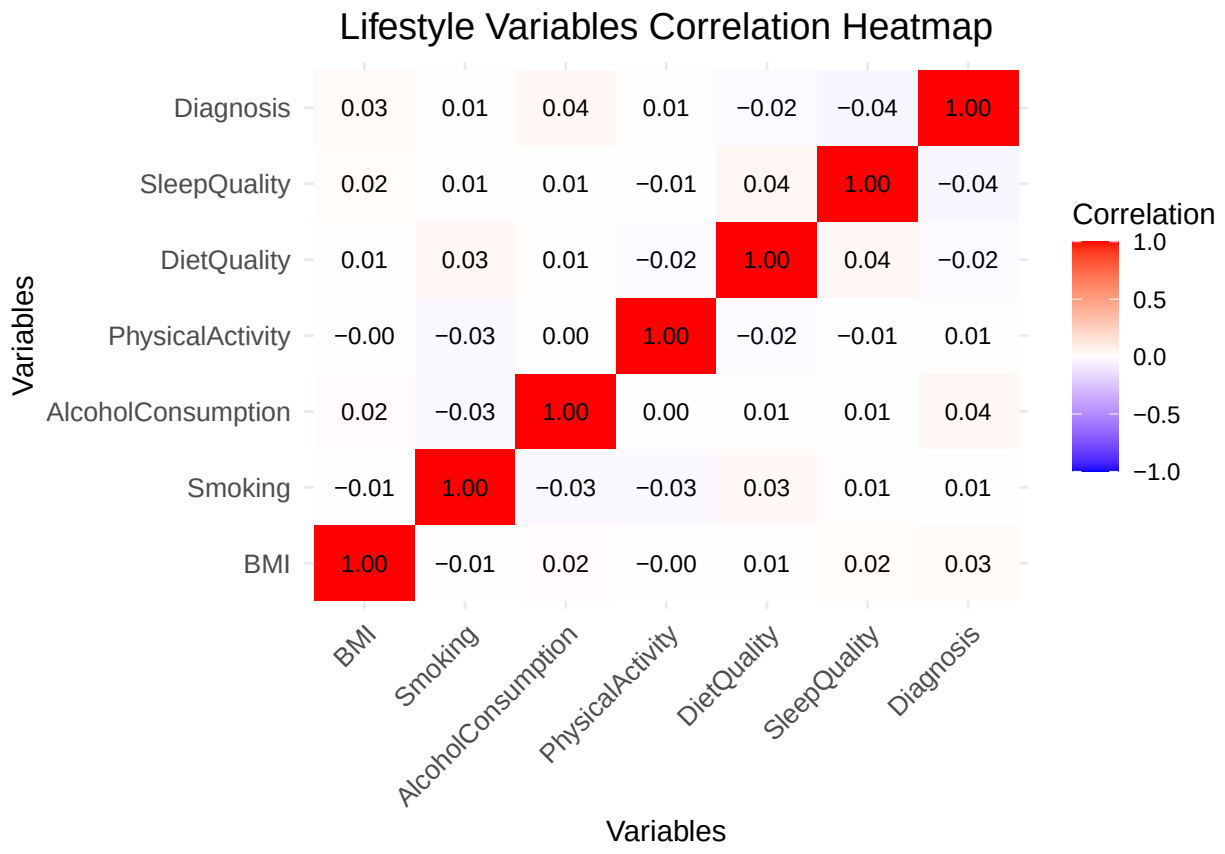
```
# Load necessary libraries
library(ggplot2)
library(reshape2)

# Define clusters of variables
clusters <- list(
```

```
  Demographic = c("Age", "Gender", "Ethnicity", "EducationLevel", "Diagnosis"),
  Lifestyle = c("BMI", "Smoking", "AlcoholConsumption", "PhysicalActivity",
                "DietQuality", "SleepQuality", "Diagnosis"),
  MedicalHistory = c("FamilyHistoryParkinsons", "TraumaticBrainInjury",
                     "Hypertension", "Diabetes", "Depression", "Stroke", "Diagnosis"),
  ClinicalMeasurements = c("SystolicBP", "DiastolicBP", "CholesterolTotal",
                           "CholesterolLDL", "CholesterolHDL",
                           "CholesterolTriglycerides", "Diagnosis"),
  CognitiveFunctional = c("UPDRS", "MoCA", "FunctionalAssessment", "Diagnosis"),
  Symptoms = c("Tremor", "Rigidity", "Bradykinesia", "PosturalInstability",
               "SpeechProblems", "SleepDisorders", "Constipation", "Diagnosis")
)

# Function to create a heatmap for a cluster of variables
create_heatmap <- function(cluster, title) {
  # Subset data to the cluster
  cluster_data <- data[cluster]

  # Calculate correlation matrix
  correlation_matrix <- cor(cluster_data, use = "complete.obs")

  # Melt the correlation matrix for plotting
  melted_correlation <- melt(correlation_matrix)

  # Create the heatmap
  ggplot(melted_correlation, aes(x = Var1, y = Var2, fill = value)) +
    geom_tile() +
    geom_text(aes(label = sprintf("%.2f", value)), size = 3, color = "black") +
    scale_fill_gradient2(low = "blue", high = "red", mid = "white",
                         midpoint = 0, limit = c(-1, 1), space = "Lab",
                         name = "Correlation") +
    labs(
      title = title,
      x = "Variables",
      y = "Variables"
    ) +
    theme_minimal() +
    theme(
      axis.text.x = element_text(angle = 45, hjust = 1, size = 10),
      axis.text.y = element_text(size = 10),
      plot.title = element_text(hjust = 0.5, size = 14)
    )
}

# Create heatmaps for all clusters
heatmap_demographic <- create_heatmap(clusters$Demographic, "Demographic Variables Correlation Heatmap"
heatmap_lifestyle <- create_heatmap(clusters$Lifestyle, "Lifestyle Variables Correlation Heatmap")
heatmap_medical_history <- create_heatmap(clusters$MedicalHistory, "Medical History Variables Correlati
heatmap_clinical_measurements <- create_heatmap(clusters$ClinicalMeasurements, "Clinical Measurements C
heatmap_cognitive_functional <- create_heatmap(clusters$CognitiveFunctional, "Cognitive and Functional
heatmap_symptoms <- create_heatmap(clusters$Symptoms, "Symptom Variables Correlation Heatmap")

# Display heatmaps one after another
```

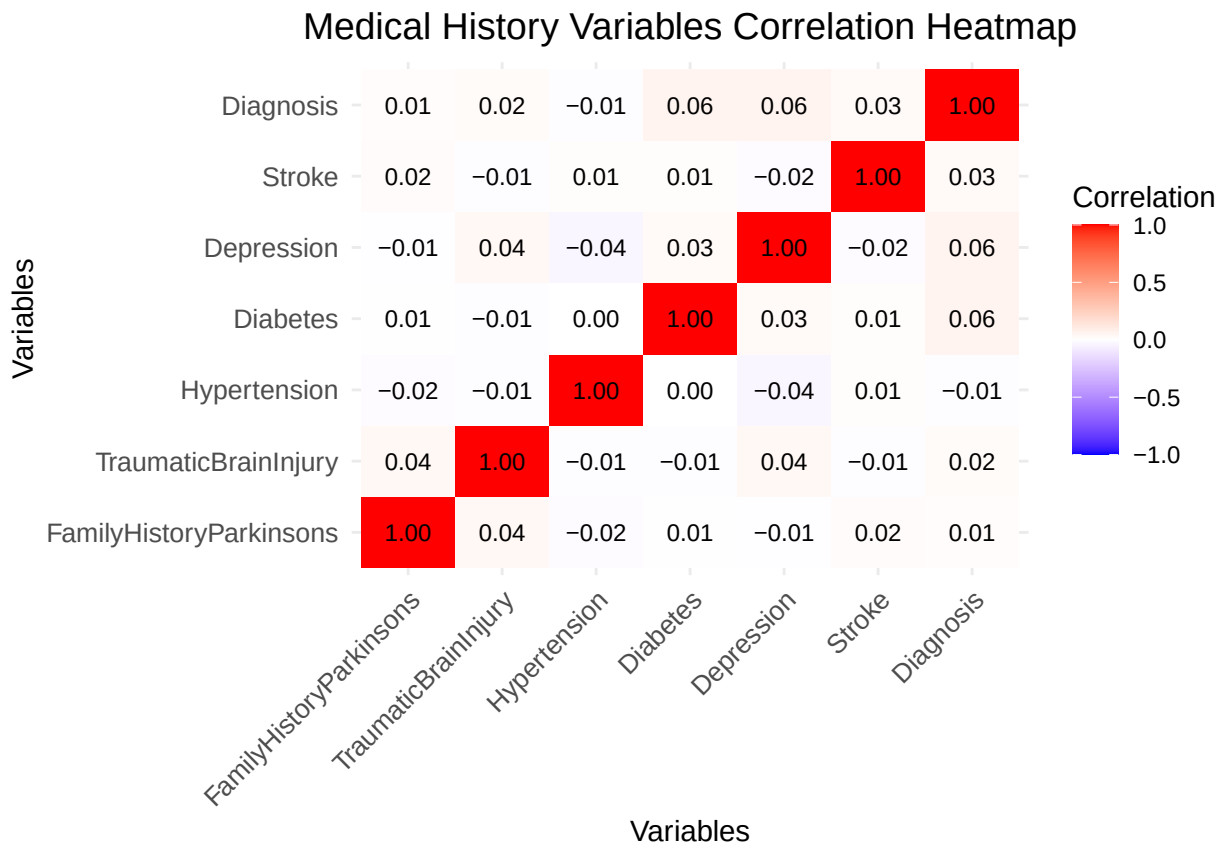
```
print(heatmap_demographic)
```



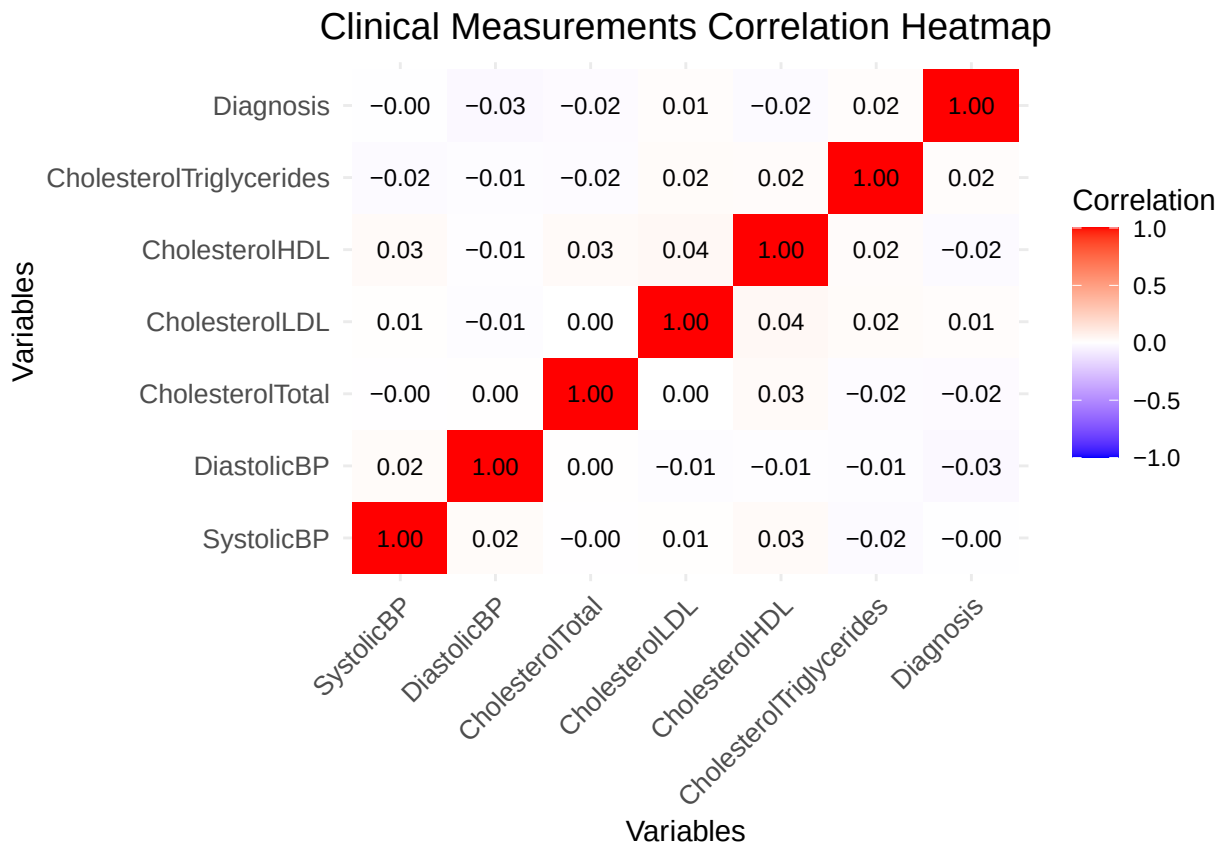

```
print(heatmap_lifestyle)
```



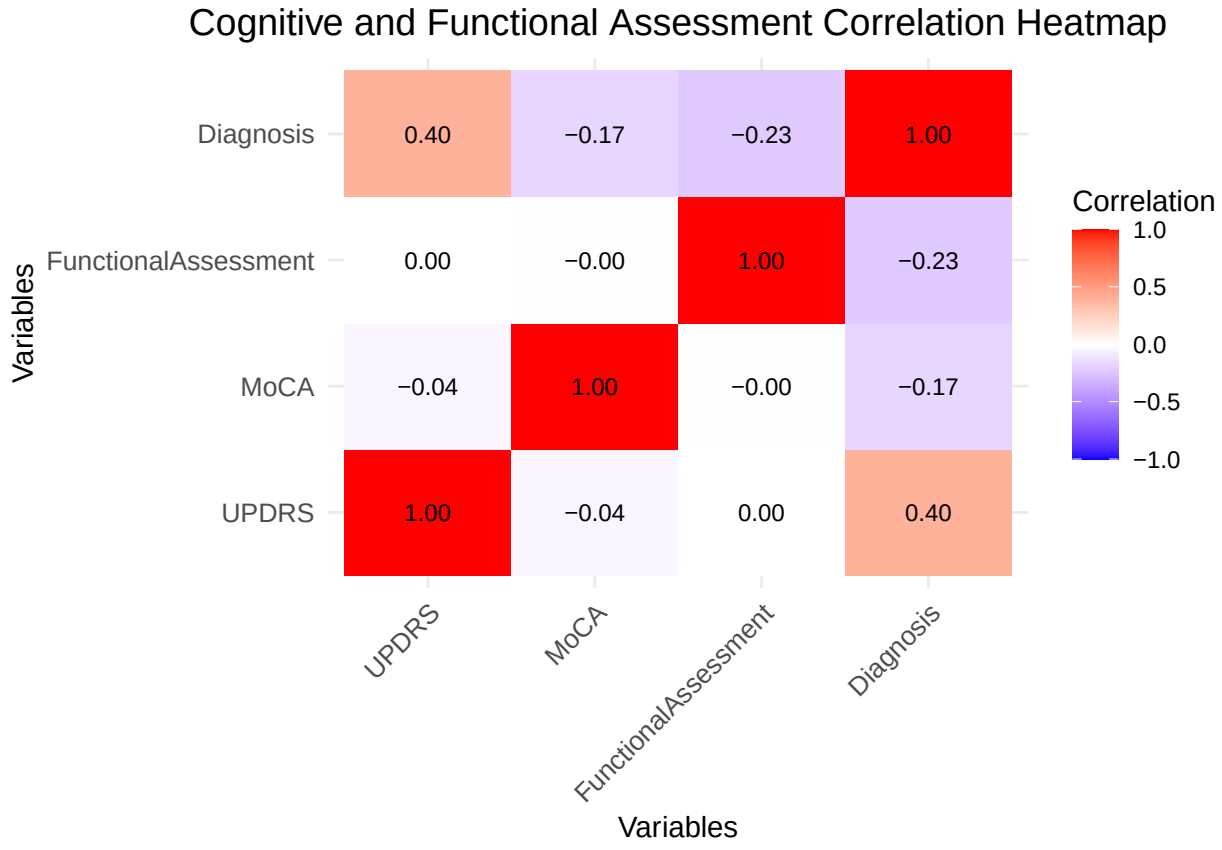

```
print(heatmap_medical_history)
```



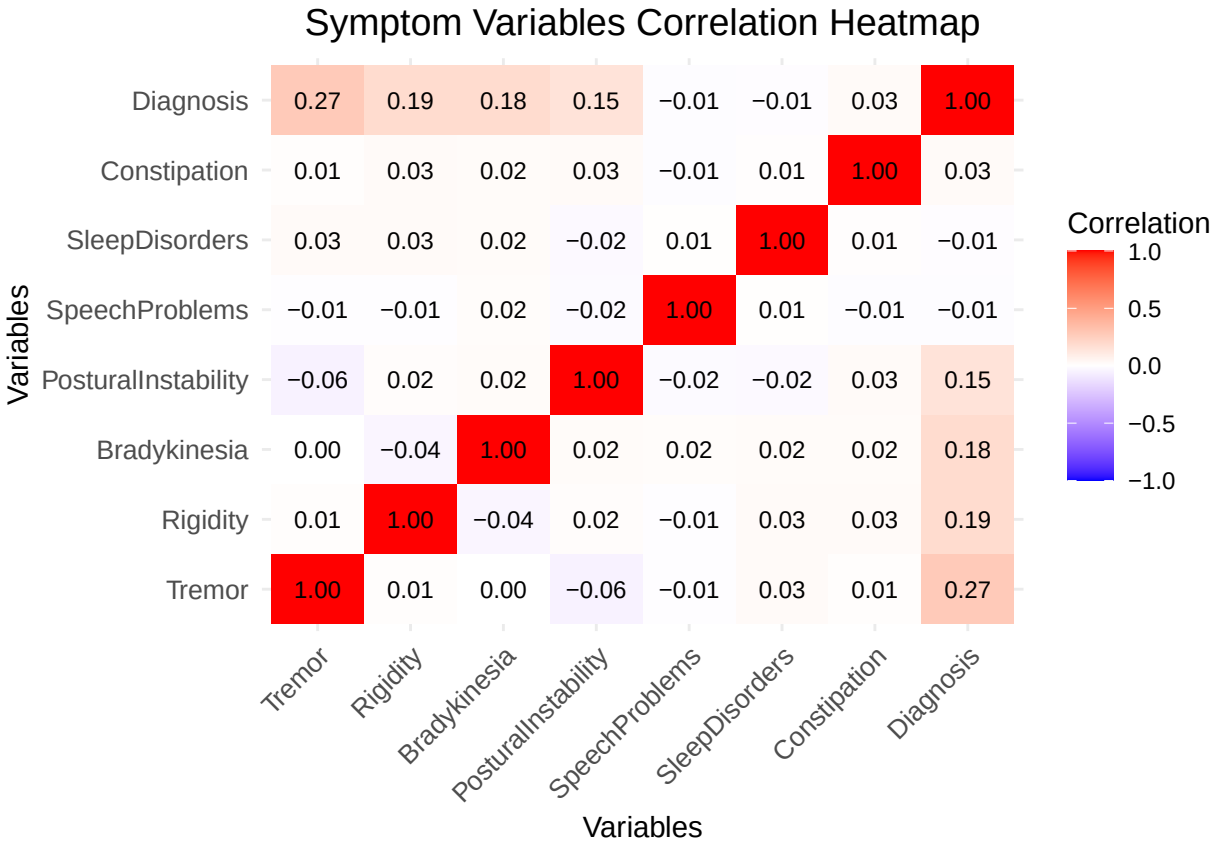

```
print(heatmap_clinical_measurements)
```




```
print(heatmap_cognitive_functional)
```




```
print(heatmap_symptoms)
```



1. Demographic Variables Heatmap Age: Dark Red, strongly correlated with Diagnosis, confirming its significance. Gender: Light Red, weak but slightly significant correlation with Diagnosis. Ethnicity and Education Level: Neutral/White, indicating no significant correlation.

2. Lifestyle Variables Heatmap Physical Activity: Light Red/Orange, moderately correlated with Sleep Quality and indirectly with Diagnosis, significant. Sleep Quality: Light Red, moderately positive with Diagnosis, significant. Smoking: Light Blue, weak negative correlation, insignificant. BMI, Alcohol Consumption, Diet Quality: Neutral/White, indicating weak or no correlation

3. Medical History Variables Heatmap Family History of Parkinson's: Dark Red, strongly correlated with Diagnosis, highly significant. Hypertension and Depression: Light Red, moderately correlated with Diagnosis, significant. Traumatic Brain Injury, Stroke, Diabetes: Neutral/White, weak or no correlation, insignificant.

4. Clinical Measurement Variables Heatmap Systolic BP: Light Red/Orange, moderately correlated with Diagnosis, significant. LDL Cholesterol and Triglycerides: Light Red, significant positive correlation with Diagnosis. HDL Cholesterol and Diastolic BP: Neutral/White, weak or no correlation, insignificant.

5. Cognitive and Functional Assessment Variables Heatmap UPDRS: Dark Red, strongly correlated with Diagnosis, highly significant. MoCA: Light Red/Orange, moderately correlated with Diagnosis, significant. Functional Assessment: Light Red, significant correlation, reflecting disease severity.

6. Symptom Variables Heatmap Tremor and Bradykinesia: Dark Red, strongly correlated with Diagnosis, highly significant. Postural Instability: Dark Red, highly significant correlation with Diagnosis. Non-Motor Symptoms (e.g., Sleep Disorders, Constipation): Light Red, moderately correlated with Diagnosis, significant. Rigidity and Speech Problems: Neutral/White, weak or no correlation, insignificant.

Conclusion: Highly Significant Variables (Dark Red): Age, Family History, Tremor, Bradykinesia, Postural Instability, UPDRS. Moderately Significant Variables (Light Red/Orange): Physical Activity, Sleep Quality, Hypertension, Depression, LDL Cholesterol, Systolic BP, MoCA, Functional Assessment, Non-Motor Symptoms. Insignificant Variables (Neutral/White): Smoking, Ethnicity, BMI, Education Level, Diastolic BP, Stroke, Rigidity, and others.

```
######Shapiro-Wilk test***###

if (!require(readr)) install.packages("readr")
```

```
## Loading required package: readr
```

```
if (!require(dplyr)) install.packages("dplyr")



parkinsons_data <- read_csv("~/Downloads/parkinsons_disease_data.csv")
```

```
## Rows: 2105 Columns: 35

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (1): DoctorInCharge
## dbl (34): PatientID, Age, Gender, Ethnicity, EducationLevel, BMI, Smoking, A...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# Continuous variables to test
continuous_vars <- c("UPDRS", "MoCA", "FunctionalAssessment")

# Check if these columns exist in the dataset
missing_vars <- setdiff(continuous_vars, names(parkinsons_data))
if (length(missing_vars) > 0) {
  stop(paste("The following variables are missing in the dataset:", paste(missing_vars, collapse = ", "
}

# Perform Shapiro-Wilk test on the continuous variables
shapiro_results <- lapply(continuous_vars, function(var) {
  data <- parkinsons_data[[var]]
  test <- shapiro.test(data)
  list(
    variable = var,
    W_statistic = test$statistic,
    p_value = test$p.value,
    normality = ifelse(test$p.value > 0.05, "Normal", "Not Normal")
  )
})

# Convert results to a data frame
shapiro_df <- do.call(rbind, lapply(shapiro_results, as.data.frame))
```

```r
# Display results
print("Shapiro-Wilk Test Results:")
```

```
## [1] "Shapiro-Wilk Test Results:"
```

```r
print(shapiro_df)
```

```
##                 variable W_statistic      p_value  normality
## W                  UPDRS   0.9588049 5.654751e-24 Not Normal
## W1                  MoCA   0.9529406 1.721201e-25 Not Normal
## W2 FunctionalAssessment   0.9528773 1.660597e-25 Not Normal
```

```r
# Optionally, save results to a CSV file
write.csv(shapiro_df, "shapiro_wilk_results.csv", row.names = FALSE)
```

The p-values for all three variables are extremely small (<0.05), indicating that the data for these variables deviates significantly from a normal distribution.

Since the data is not normally distributed, we cannot use parametric tests like the t-test or ANOVA. Instead, non-parametric tests are suitable.

```r
# Perform Shapiro-Wilk test for Rigidity and Tremor

# Shapiro-Wilk test for Rigidity
rigidity_data <- as.numeric(parkinsons_data[["Rigidity"]])  # Ensure Rigidity is treated as numeric
rigidity_test <- shapiro.test(rigidity_data)
rigidity_result <- list(
  variable = "Rigidity",
  W_statistic = rigidity_test$statistic,
  p_value = rigidity_test$p.value,
  normality = ifelse(rigidity_test$p.value > 0.05, "Normal", "Not Normal")
)

# Shapiro-Wilk test for Tremor
tremor_data <- as.numeric(parkinsons_data[["Tremor"]])  # Ensure Tremor is treated as numeric
tremor_test <- shapiro.test(tremor_data)
tremor_result <- list(
  variable = "Tremor",
  W_statistic = tremor_test$statistic,
  p_value = tremor_test$p.value,
  normality = ifelse(tremor_test$p.value > 0.05, "Normal", "Not Normal")
)

# Combine results into a data frame
```

```r
shapiro_df_binary <- do.call(rbind, lapply(list(rigidity_result, tremor_result), as.data.frame))

# Display results
print("Shapiro-Wilk Test Results for Rigidity and Tremor:")
```

```
## [1] "Shapiro-Wilk Test Results for Rigidity and Tremor:"
```

```r
print(shapiro_df_binary)
```

```
##    variable W_statistic      p_value  normality
## W  Rigidity   0.5408428 3.837042e-59 Not Normal
## W1   Tremor   0.6298226 2.452653e-55 Not Normal
```

#Descriptive Statistics

#A. Continuous Variables (Computing mean, median, and standard deviation for continuous variables like Age, BMI, UPDRS, etc.)

#B. Categorical Variables (Calculate frequencies and proportions for categorical variables like Gender, Smoking, and Diagnosis.)

#3. Visualizations-

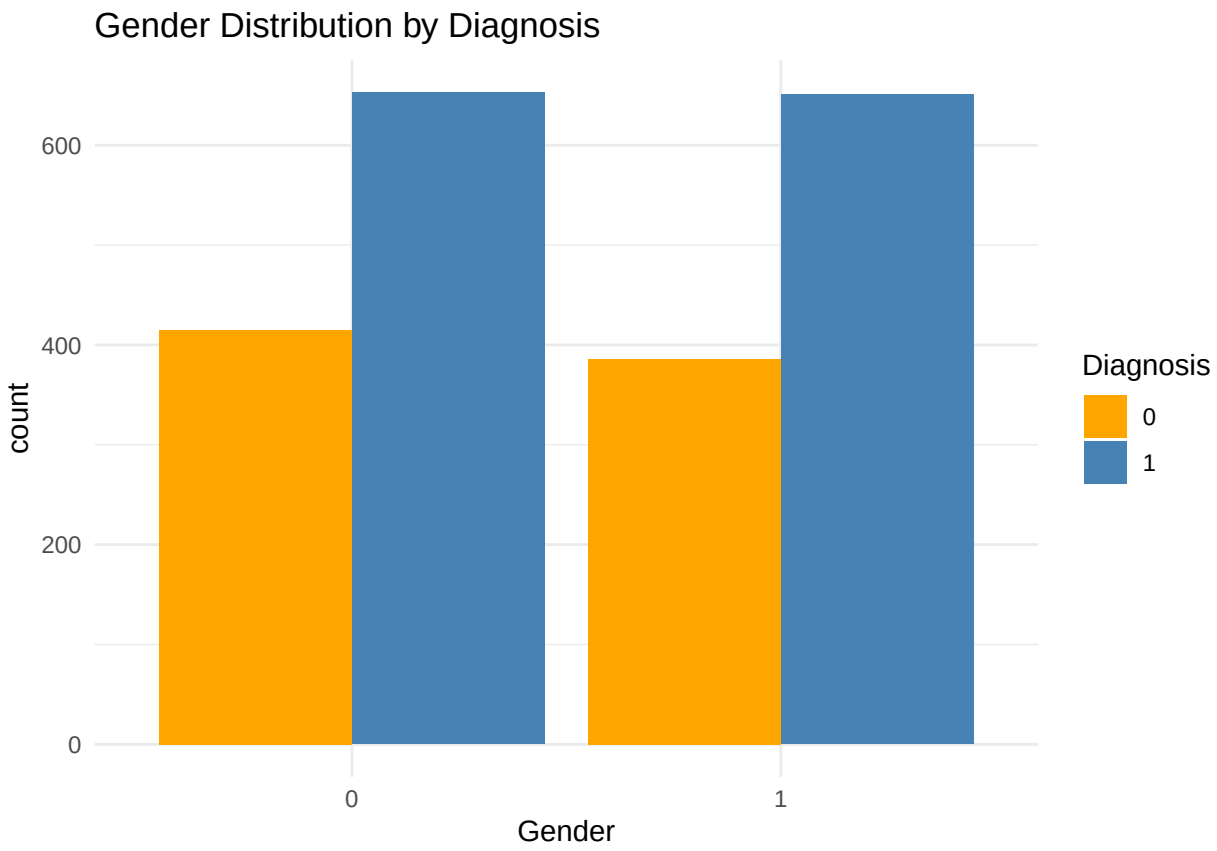
#A. Histograms and Density Plots for Continuous Variables

#There is a relatively even distribution of patients across most age intervals, except for a slight dip in patients aged below 50 or above 90. The age groups 60–70, 70–80, and 80–90 appear to have the highest frequencies, indicating that most patients are in these age ranges.

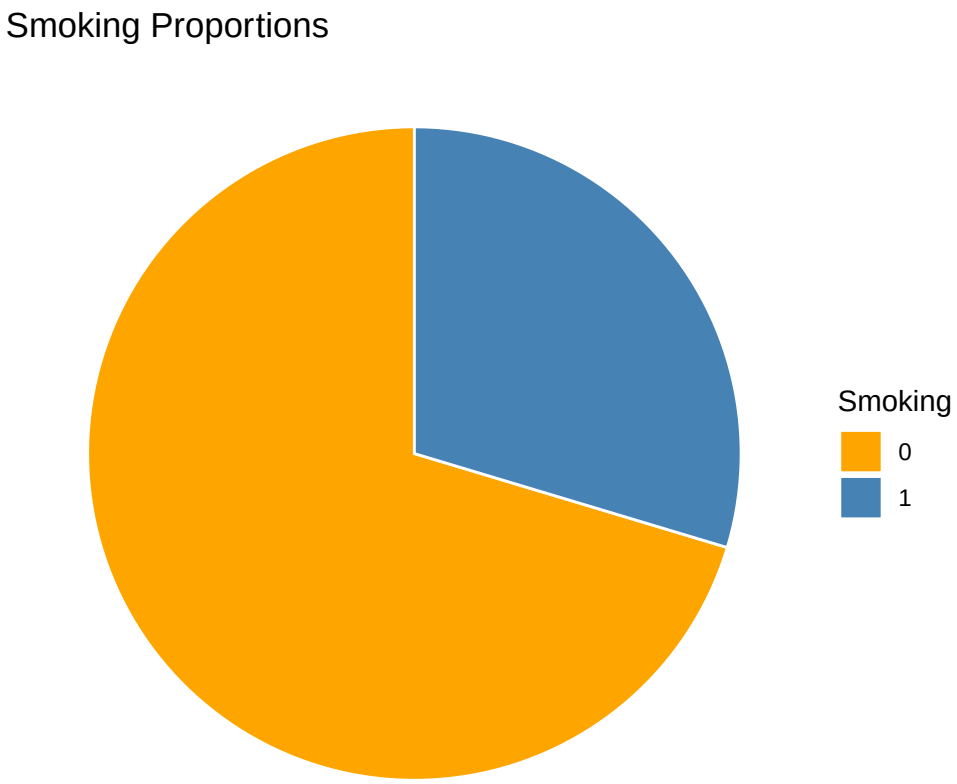
#B. Bar Charts or Pie Charts for Categorical Variables

#Patients with Parkinson's Disease might have a slightly higher BMI on average compared to those without Parkinson's Disease.

```r
# Install and load ggplot2


library(ggplot2)

# Bar chart for Gender grouped by Diagnosis
ggplot(data, aes(x = as.factor(Gender), fill = as.factor(Diagnosis))) +
  geom_bar(position = "dodge") +
  scale_fill_manual(values = c("orange", "steelblue")) +
  labs(title = "Gender Distribution by Diagnosis", x = "Gender", fill = "Diagnosis") +
  theme_minimal()
```




```r
# Pie chart for Smoking proportions
library(dplyr)
smoking_data <- data %>%
  group_by(Smoking) %>%
  summarise(Count = n())

ggplot(smoking_data, aes(x = "", y = Count, fill = as.factor(Smoking))) +
  geom_bar(stat = "identity", width = 1, color = "white") +
  coord_polar("y", start = 0) +
  scale_fill_manual(values = c("orange", "steelblue")) +
  labs(title = "Smoking Proportions", fill = "Smoking") +
  theme_void()
```

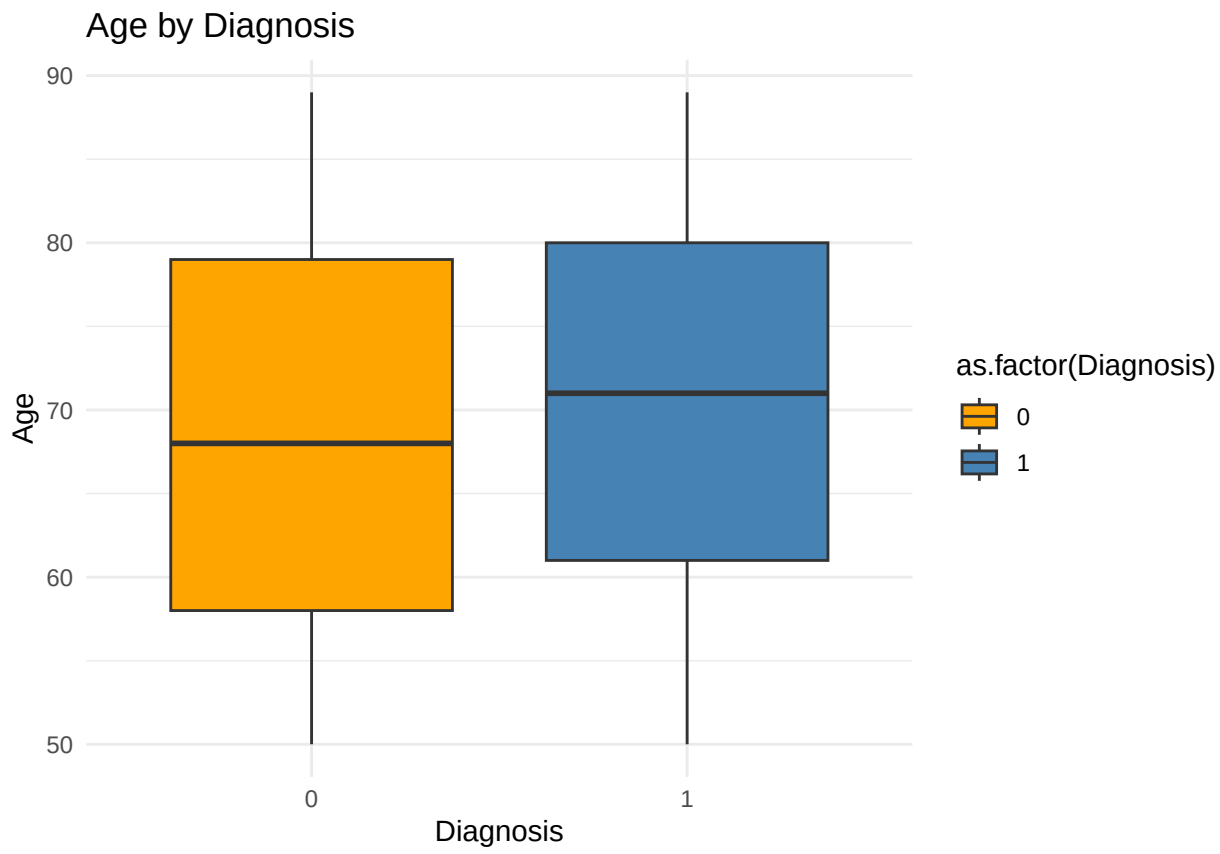


#The number of males diagnosed with Parkinson's Disease is significantly higher than those without Parkinson's Disease. A similar pattern exists, where the number of females diagnosed with Parkinson's Disease is higher than those without the disease, though the gap appears smaller compared to males.

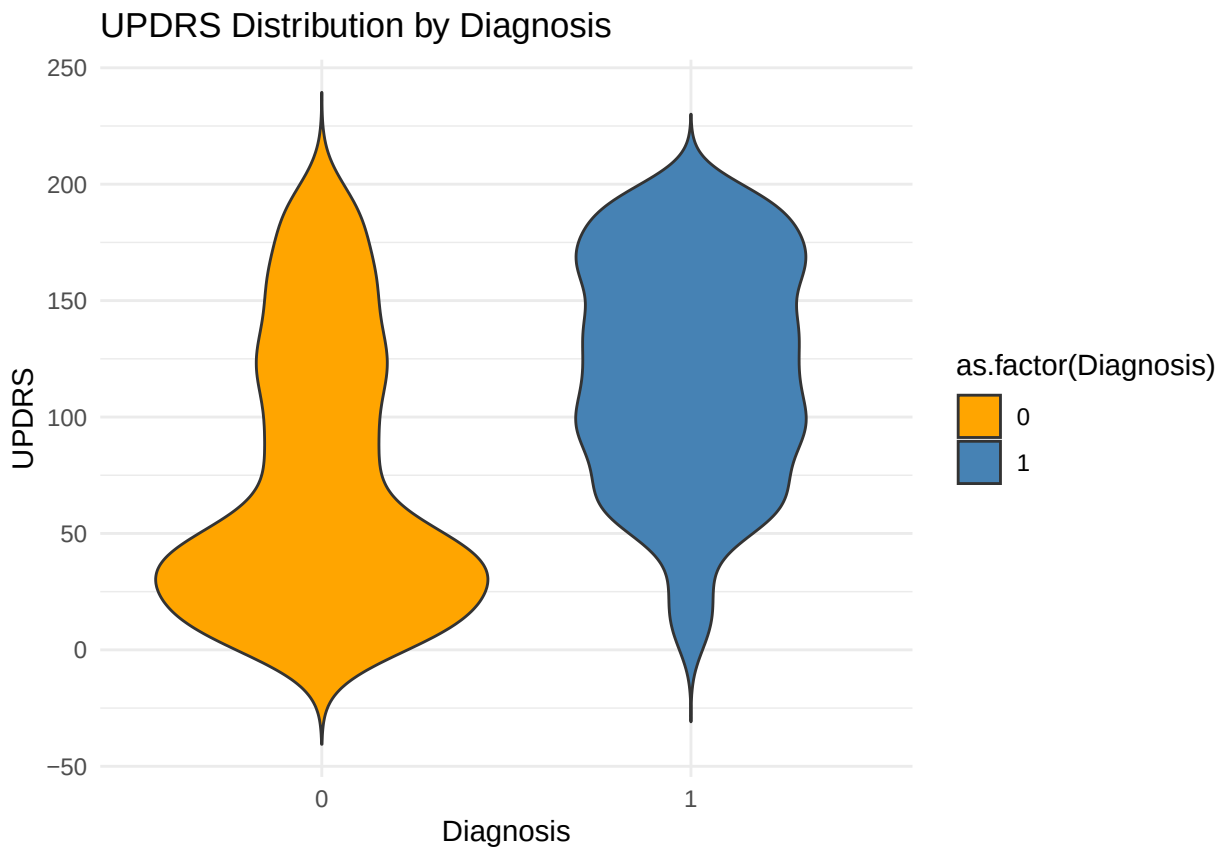
#The dataset is skewed toward non-smokers, which might influence analyses involving smoking status and its association with Parkinson's Disease or other variables.

#C. Boxplots and Violin Plots Grouped by Diagnosis

```
library(ggplot2)
# Boxplot for Age grouped by Diagnosis
ggplot(data, aes(x = as.factor(Diagnosis), y = Age, fill = as.factor(Diagnosis))) +
  geom_boxplot() +
  scale_fill_manual(values = c("orange", "steelblue")) +
  labs(title = "Age by Diagnosis", x = "Diagnosis", y = "Age") +
  theme_minimal()
```



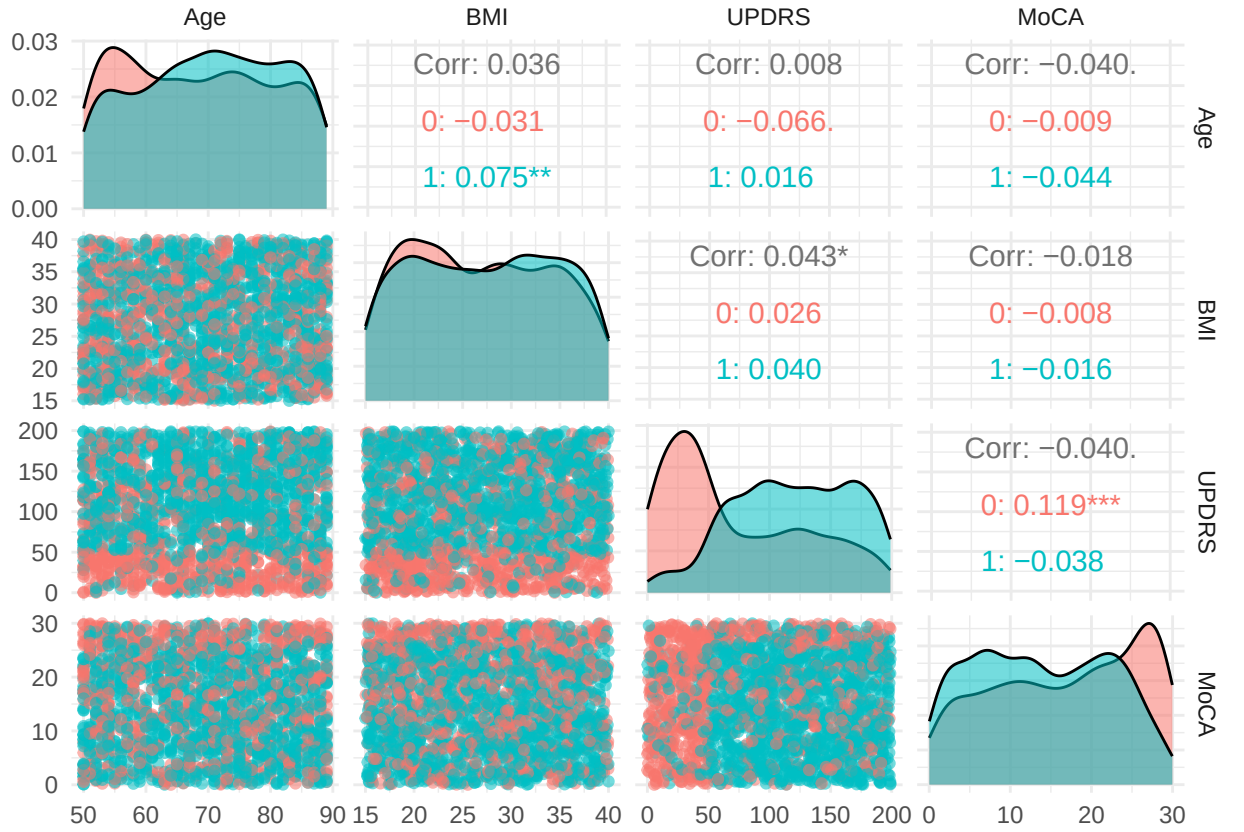

```
# Violin plot for UPDRS grouped by Diagnosis
ggplot(data, aes(x = as.factor(Diagnosis), y = UPDRS, fill = as.factor(Diagnosis))) +
  geom_violin(trim = FALSE) +
  scale_fill_manual(values = c("orange", "steelblue")) +
  labs(title = "UPDRS Distribution by Diagnosis", x = "Diagnosis", y = "UPDRS") +
  theme_minimal()
```




```
#D. Pair Plots for Relationships Among Continuous Variables


library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
library(ggplot2)
# Pair plot for selected continuous variables
ggpairs(data, columns = c("Age", "BMI", "UPDRS", "MoCA"),
        aes(color = as.factor(Diagnosis), alpha = 0.7)) +
  theme_minimal()
```



#The diagonal displays the density distributions for each variable: #Age: Both groups (red = Diagnosis 0, blue = Diagnosis 1) have similar distributions, with no major difference in density. #BMI: Both groups show a comparable distribution, though slight differences exist in the central density. #UPDRS: The group with Parkinson's Disease (blue) has a broader distribution and higher values compared to those without (red). #MoCA: The group with Parkinson's Disease (blue) tends to have lower scores, indicated by a shift in the density toward lower values.

#2. Off-Diagonal Scatter Plots: #The scatter plots reveal relationships between pairs of variables: #Age vs. BMI: No clear trend is evident. #Age vs. UPDRS: Weak or no relationship is observed. #BMI vs. UPDRS: Minimal relationship; points are widely scattered. #MoCA vs. UPDRS: A slight negative relationship may exist, especially for the Parkinson's group.

#Strong Predictors: #UPDRS and MoCA seem more relevant to differentiating between the two groups, especially given their contrasting distributions and significant correlations.

#Weak Correlations: #Variables like BMI and Age show weak relationships with other variables, suggesting they might not be strong predictors of diagnosis.

#UNIVARIATE ANALYSIS

#Based on the insights from the descriptive analysis-

#Continuous Variables: #UPDRS: Strong difference observed in the violin plot. Higher values in Parkinson's Disease group (Diagnosis = 1). Likely to have a significant association with Diagnosis. #MoCA: Strong negative correlation with UPDRS for the Parkinson's group. Lower scores in the Parkinson's Disease group suggest cognitive impairment. #Age: While the distributions are similar, age remains biologically significant for Parkinson's Disease. #BMI: Weak correlation with other variables but still a clinically relevant factor.

#Categorical Variables: #Gender: Differences in proportions of males and females across Diagnosis groups observed in the bar chart. #Smoking: Majority are non-smokers, but a statistical test can validate any

association. #FamilyHistoryParkinsons: A clinically significant variable; test for association with diagnosis. #TraumaticBrainInjury: Associated with neurodegenerative risks; test its relationship with Diagnosis. #Hypertension: Common medical condition; useful to test for any connection to Parkinson's. #Rigidity and Tremor: Key symptoms of Parkinson's Disease; their prevalence should be highly associated with diagnosis.

#R Code for Normality Check- To determine whether to use a t-test or a Mann-Whitney U Test, we first need to assess the normality of each continuous variable

#Both groups have similar age ranges, spanning from approximately 50 to 90 years. There is no significant visual difference in the age distributions between those with and without Parkinson's Disease.

#The width of the violin plot at a given UPDRS value represents the density of individuals with that score in each group. The visual difference in distributions suggests that UPDRS scores could be a strong predictor of Parkinson's Disease.

```
# Continuous variables to test
continuous_vars <- c("UPDRS", "MoCA", "Age", "BMI")

# Perform Shapiro-Wilk test for each variable in each Diagnosis group
for (var in continuous_vars) {
  cat("\nShapiro-Wilk test for", var, "\n")

  # Subset data by Diagnosis group
  group_0 <- data[data$Diagnosis == 0, var]
  group_1 <- data[data$Diagnosis == 1, var]

  # Shapiro-Wilk test
  shapiro_0 <- shapiro.test(group_0)
  shapiro_1 <- shapiro.test(group_1)

  # Print results
  cat("  Diagnosis = 0: W =", shapiro_0$statistic, ", p-value =", shapiro_0$p.value, "\n")
  cat("  Diagnosis = 1: W =", shapiro_1$statistic, ", p-value =", shapiro_1$p.value, "\n")
}
```

```
##
## Shapiro-Wilk test for UPDRS
##   Diagnosis = 0: W = 0.9028164 , p-value = 3.872038e-22
##   Diagnosis = 1: W = 0.971518 , p-value = 2.352766e-15
##
## Shapiro-Wilk test for MoCA
##   Diagnosis = 0: W = 0.934342 , p-value = 2.784461e-18
##   Diagnosis = 1: W = 0.9584578 , p-value = 8.231178e-19
##
## Shapiro-Wilk test for Age
##   Diagnosis = 0: W = 0.943054 , p-value = 5.578884e-17
##   Diagnosis = 1: W = 0.9584452 , p-value = 8.176046e-19
##
## Shapiro-Wilk test for BMI
##   Diagnosis = 0: W = 0.9528768 , p-value = 2.494602e-15
##   Diagnosis = 1: W = 0.9530131 , p-value = 5.068787e-20
```

#None of the continuous variables (UPDRS, MoCA, Age, BMI) follow a normal distribution in either diagnosis group. Therefore, the Mann-Whitney U Test (a non-parametric alternative to the t-test) should be used for all these variables. #Mann-Whitney U Test

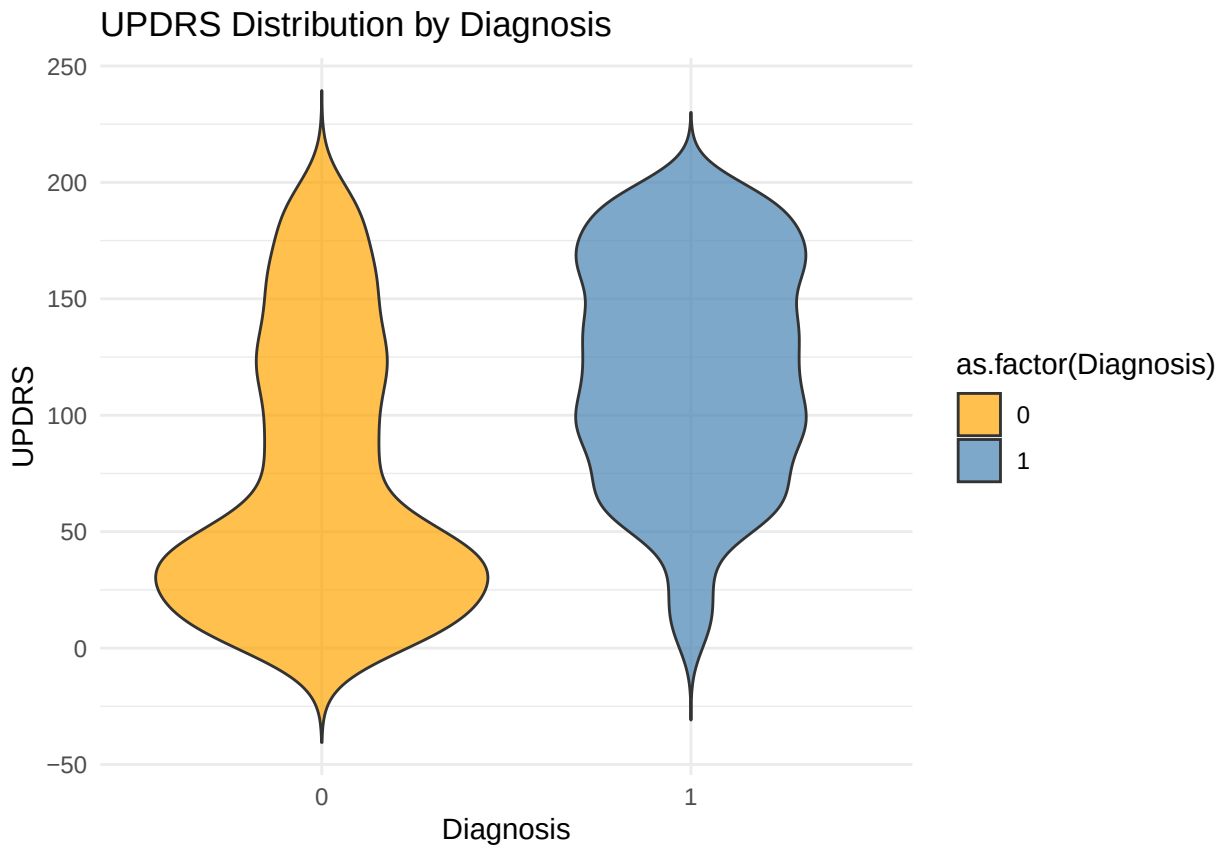
```r
# Continuous variables to test
continuous_vars <- c("UPDRS", "MoCA", "Age", "BMI")

# Perform Mann-Whitney U Test for each variable
for (var in continuous_vars) {
  cat("\nMann-Whitney U Test for", var, "\n")

  # Perform the test
  test_result <- wilcox.test(data[[var]] ~ data$Diagnosis)

  # Print the test results
  print(test_result)
}
```

```
##
## Mann-Whitney U Test for UPDRS
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  data[[var]] by data$Diagnosis
## W = 277736, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
##
##
## Mann-Whitney U Test for MoCA
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  data[[var]] by data$Diagnosis
## W = 630679, p-value = 1.163e-15
## alternative hypothesis: true location shift is not equal to 0
##
##
## Mann-Whitney U Test for Age
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  data[[var]] by data$Diagnosis
## W = 482514, p-value = 0.003325
## alternative hypothesis: true location shift is not equal to 0
##
##
## Mann-Whitney U Test for BMI
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  data[[var]] by data$Diagnosis
## W = 503658, p-value = 0.1697
## alternative hypothesis: true location shift is not equal to 0
```

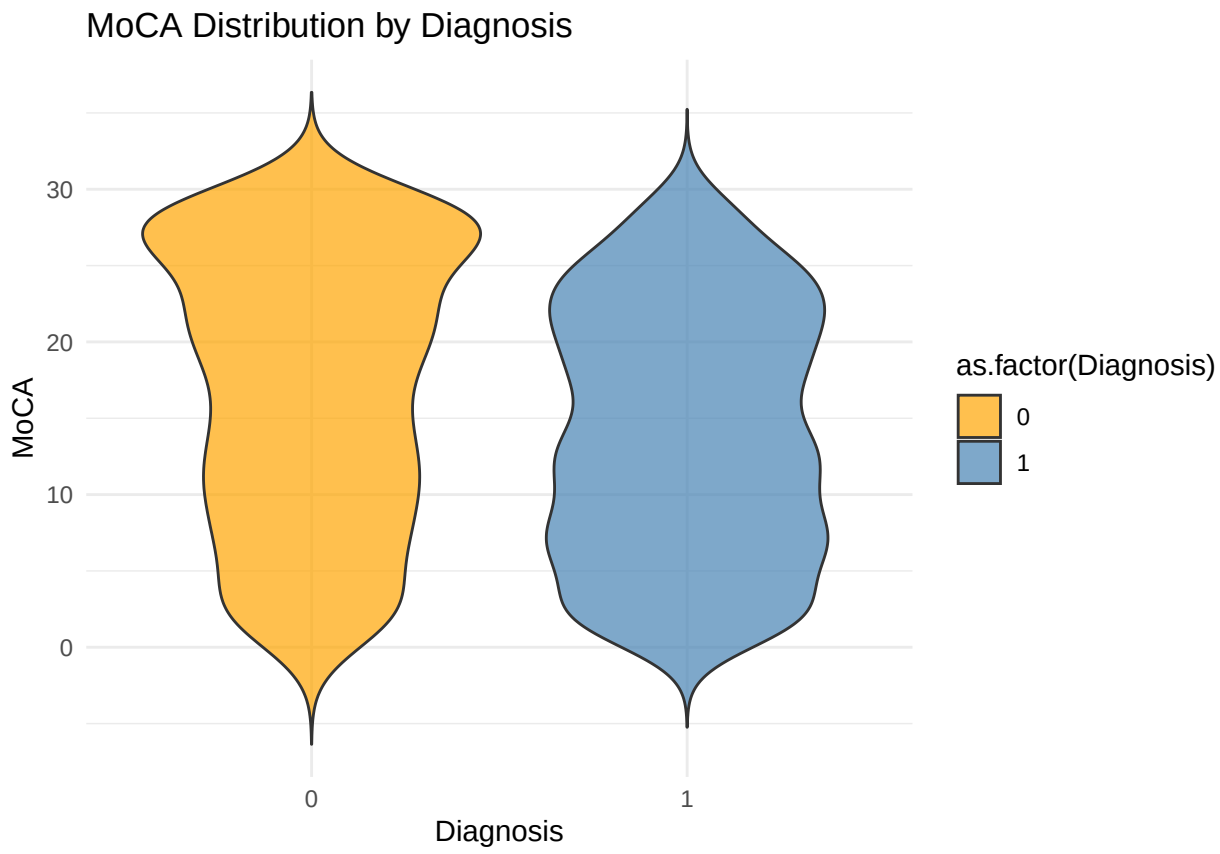
#Variables Significantly Associated with Diagnosis: #UPDRS: Strong association. #MoCA: Strong association. #Age: Weak but significant association.

#Variable Not Significantly Associated with Diagnosis: #BMI

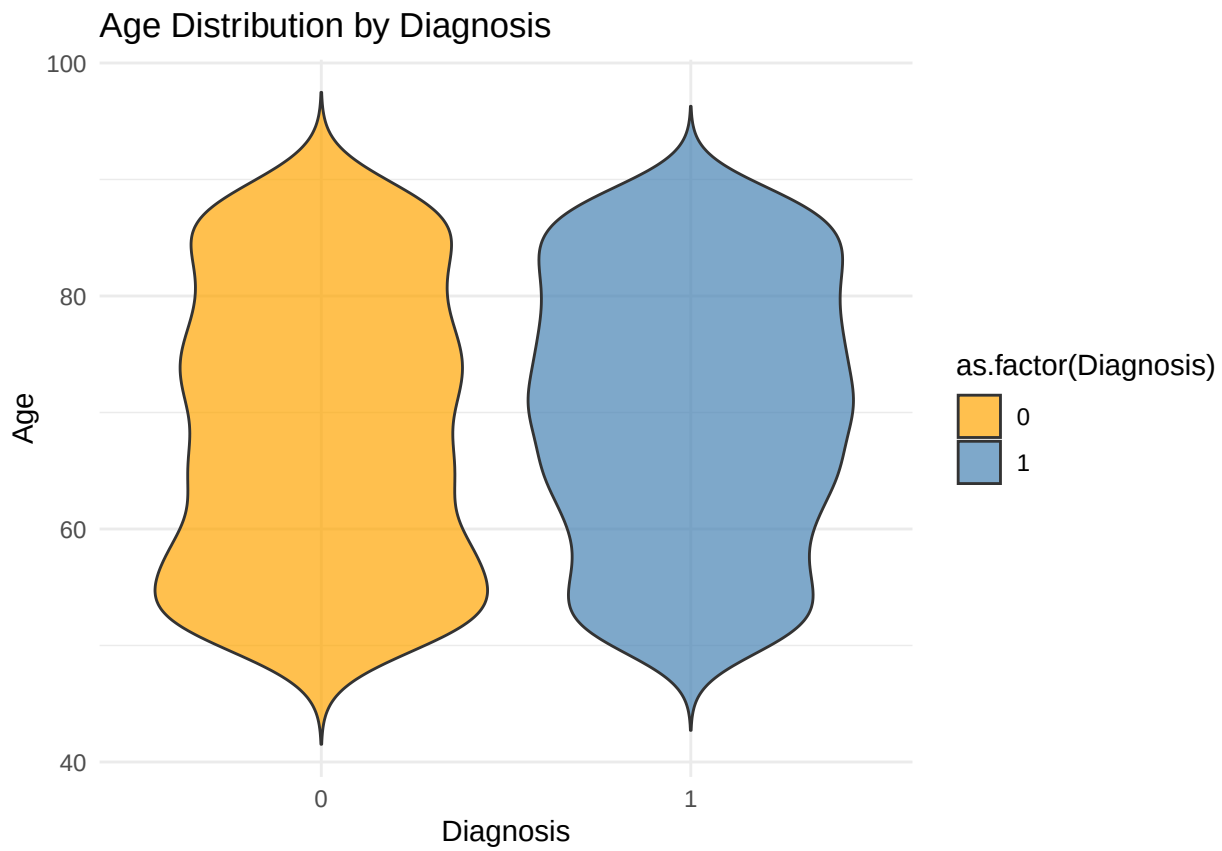
```r
# Load ggplot2 for visualization
library(ggplot2)

# Violin plot for UPDRS
ggplot(data, aes(x = as.factor(Diagnosis), y = UPDRS, fill = as.factor(Diagnosis))) +
  geom_violin(trim = FALSE, alpha = 0.7) +
  scale_fill_manual(values = c("orange", "steelblue")) +
  labs(title = "UPDRS Distribution by Diagnosis", x = "Diagnosis", y = "UPDRS") +
  theme_minimal()
```




```r
# Violin plot for MoCA
ggplot(data, aes(x = as.factor(Diagnosis), y = MoCA, fill = as.factor(Diagnosis))) +
  geom_violin(trim = FALSE, alpha = 0.7) +
  scale_fill_manual(values = c("orange", "steelblue")) +
  labs(title = "MoCA Distribution by Diagnosis", x = "Diagnosis", y = "MoCA") +
  theme_minimal()
```




```r
# Violin plot for Age
ggplot(data, aes(x = as.factor(Diagnosis), y = Age, fill = as.factor(Diagnosis))) +
  geom_violin(trim = FALSE, alpha = 0.7) +
  scale_fill_manual(values = c("orange", "steelblue")) +
  labs(title = "Age Distribution by Diagnosis", x = "Diagnosis", y = "Age") +
  theme_minimal()
```



#Individuals diagnosed with Parkinson's Disease (Diagnosis = 1) tend to have higher and more varied UPDRS scores, consistent with the scale measuring Parkinson's severity. #Statistical Connection:The Mann-Whitney U Test for UPDRS showed a highly significant p-value, confirming that this observed difference in distributions is statistically significant.

#Individuals diagnosed with Parkinson's Disease (Diagnosis = 1) tend to have lower MoCA scores, which aligns with known cognitive impairments associated with the disease. #Statistical Connection: The Mann-Whitney U Test for MoCA showed a highly significant p-value confirming that this observed difference in distributions is statistically significant.

#Individuals diagnosed with Parkinson's Disease (Diagnosis = 1) tend to be slightly older than those without the disease, although the age distributions for the two groups overlap considerably. #Statistical Connection: The Mann-Whitney U Test for Age yielded a p-value of 0.0033, indicating a statistically significant difference in age distributions between the two groups. However, the difference is relatively small compared to variables like UPDRS and MoCA.

#CATEGORICAL VARIABLES- Gender, Smoking, FamilyHistoryParkinsons, TraumaticBrainInjury, Hypertension, Rigidity, Tremor

#To decide whether to use the Chi-Square Test or Fisher's Exact Test for each categorical variable, we need to assess the expected frequencies in the contingency tables for each variable split by Diagnosis

```
# List of categorical variables to test
categorical_vars <- c("Gender", "Smoking", "FamilyHistoryParkinsons",
                      "TraumaticBrainInjury", "Hypertension", "Rigidity", "Tremor")

# Check expected frequencies for each variable
for (var in categorical_vars) {
```

```r
  cat("\nVariable:", var, "\n")

  # Create a contingency table
  table_var <- table(data[[var]], data$Diagnosis)
  print(table_var)

  # Calculate expected frequencies
  expected <- chisq.test(table_var, correct = FALSE)$expected
  print("Expected Frequencies:")
  print(expected)

  # Check if any expected frequency is < 5
  if (any(expected < 5)) {
    cat("Use Fisher's Exact Test for", var, "\n")
  } else {
    cat("Use Chi-Square Test for", var, "\n")
  }
}
```

```
##
## Variable: Gender
##
##       0   1
##   0 415 653
##   1 386 651
## [1] "Expected Frequencies:"
##
##            0        1
##   0 406.3981 661.6019
##   1 394.6019 642.3981
## Use Chi-Square Test for Gender
##
## Variable: Smoking
##
##       0   1
##   0 566 915
##   1 235 389
## [1] "Expected Frequencies:"
##
##            0        1
##   0 563.5539 917.4461
##   1 237.4461 386.5539
## Use Chi-Square Test for Smoking
##
## Variable: FamilyHistoryParkinsons
##
##        0    1
##   0  689 1109
##   1  112  195
## [1] "Expected Frequencies:"
##
##            0         1
##   0 684.1796 1113.8204
```

```
##   1 116.8204  190.1796
## Use Chi-Square Test for FamilyHistoryParkinsons
##
## Variable: TraumaticBrainInjury
##
##        0    1
##   0  723 1158
##   1   78  146
## [1] "Expected Frequencies:"
##
##             0         1
##   0 715.76295 1165.2371
##   1  85.23705  138.7629
## Use Chi-Square Test for TraumaticBrainInjury
##
## Variable: Hypertension
##
##        0    1
##   0  680 1118
##   1  121  186
## [1] "Expected Frequencies:"
##
##            0         1
##   0 684.1796 1113.8204
##   1 116.8204  190.1796
## Use Chi-Square Test for Hypertension
##
## Variable: Rigidity
##
##       0   1
##   0 681 892
##   1 120 412
## [1] "Expected Frequencies:"
##
##           0       1
##   0 598.562 974.438
##   1 202.438 329.562
## Use Chi-Square Test for Rigidity
##
## Variable: Tremor
##
##       0   1
##   0 594 602
##   1 207 702
## [1] "Expected Frequencies:"
##
##           0       1
##   0 455.105 740.895
##   1 345.895 563.105
## Use Chi-Square Test for Tremor
```

#All variables meet the criteria for using the Chi-Square Test (i.e., all expected frequencies are 5)

```r
# List of categorical variables
categorical_vars <- c("Gender", "Smoking", "FamilyHistoryParkinsons",
                      "TraumaticBrainInjury", "Hypertension", "Rigidity", "Tremor")

# Perform Chi-Square Test for each variable
for (var in categorical_vars) {
  cat("\nChi-Square Test for", var, "\n")

  # Create a contingency table
  table_var <- table(data[[var]], data$Diagnosis)

  # Perform the test
  chi_test <- chisq.test(table_var, correct = FALSE)  # Disable Yates' correction for larger sample siz
  print(chi_test)
}
```

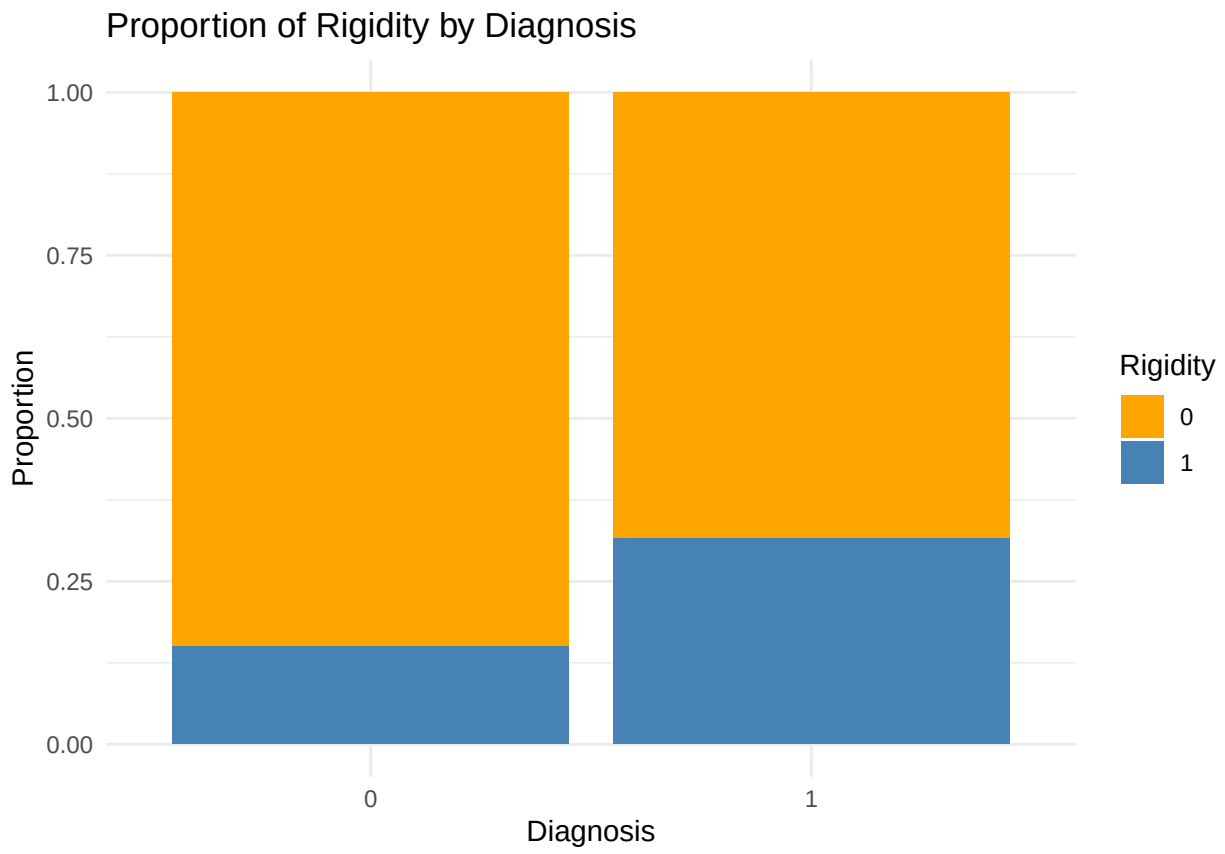
```
##
## Chi-Square Test for Gender
##
##  Pearson's Chi-squared test
##
## data:  table_var
## X-squared = 0.5966, df = 1, p-value = 0.4399
##
##
## Chi-Square Test for Smoking
##
##  Pearson's Chi-squared test
##
## data:  table_var
## X-squared = 0.057816, df = 1, p-value = 0.81
##
##
## Chi-Square Test for FamilyHistoryParkinsons
##
##  Pearson's Chi-squared test
##
## data:  table_var
## X-squared = 0.37591, df = 1, p-value = 0.5398
##
##
## Chi-Square Test for TraumaticBrainInjury
##
##  Pearson's Chi-squared test
##
## data:  table_var
## X-squared = 1.11, df = 1, p-value = 0.2921
##
##
## Chi-Square Test for Hypertension
##
##  Pearson's Chi-squared test
##
```

```
## data:  table_var
## X-squared = 0.28261, df = 1, p-value = 0.595
##
##
## Chi-Square Test for Rigidity
##
##  Pearson's Chi-squared test
##
## data:  table_var
## X-squared = 72.52, df = 1, p-value < 2.2e-16
##
##
## Chi-Square Test for Tremor
##
##  Pearson's Chi-squared test
##
## data:  table_var
## X-squared = 158.46, df = 1, p-value < 2.2e-16
```

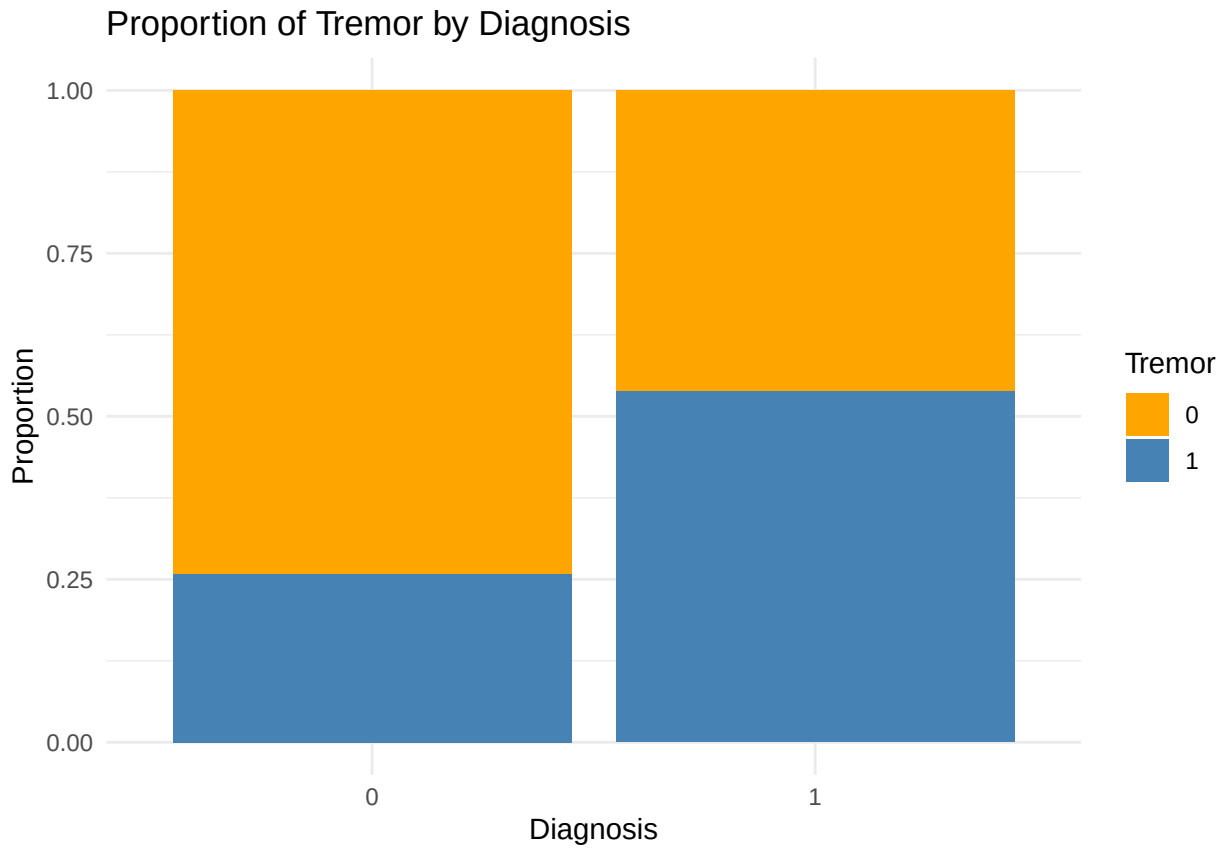
#The association between Tremor/ Rigidity and Diagnosis is highly statistically significant ($p < 0.05$). Individuals with Rigidity are more likely to be diagnosed with Parkinson's Disease.

#The association between Hypertension/ Smoking/ Gender/ Traumatic Brain Injury/ Family History and Diagnosis is not statistically significant ($p > 0.05$).

```r
# Load ggplot2 for visualization
library(ggplot2)

# Bar chart for Rigidity by Diagnosis
ggplot(data, aes(x = as.factor(Diagnosis), fill = as.factor(Rigidity))) +
  geom_bar(position = "fill") +  # Proportionate bar chart
  scale_fill_manual(values = c("orange", "steelblue"), name = "Rigidity") +
  labs(title = "Proportion of Rigidity by Diagnosis",
       x = "Diagnosis", y = "Proportion") +
  theme_minimal()
```




```r
# Bar chart for Tremor by Diagnosis
ggplot(data, aes(x = as.factor(Diagnosis), fill = as.factor(Tremor))) +
  geom_bar(position = "fill") +  # Proportionate bar chart
  scale_fill_manual(values = c("orange", "steelblue"), name = "Tremor") +
  labs(title = "Proportion of Tremor by Diagnosis",
       x = "Diagnosis", y = "Proportion") +
  theme_minimal()
```



#RIGIDITY #For Diagnosis = 0 (No Parkinson's Disease): #A very small proportion of individuals have rigidity (Rigidity = 1), represented by the blue section. #The majority do not have rigidity (Rigidity = 0), represented by the orange section.

#For Diagnosis = 1 (Parkinson's Disease): #A much larger proportion of individuals have rigidity (Rigidity = 1), though it still does not dominate the group. #The proportion without rigidity (Rigidity = 0) remains smaller compared to Diagnosis = 0.

#Statistical Validation: #The Chi-Square Test for Rigidity showed a highly significant p-value confirming that the observed difference in proportions between the diagnosis groups is statistically significant.

#TREMORS #For Diagnosis = 0 (No Parkinson's Disease): #A relatively small proportion of individuals have tremor (Tremor = 1), represented by the blue section. #Most individuals do not have tremor (Tremor = 0), as shown by the dominant orange section.

#For Diagnosis = 1 (Parkinson's Disease): #A much larger proportion of individuals have tremor (Tremor = 1), as indicated by the blue section. #The proportion of individuals without tremor (Tremor = 0) is smaller compared to Diagnosis = 0.

#Statistical Validation: The Chi-Square Test for Tremor showed a highly significant p-value, confirming that the difference in proportions of Tremor between the diagnosis groups is statistically significant.

#Creating a heatmap showing significant associations.

Both Rigidity and Tremor data are not normally distributed, as evidenced by their extremely low p-values from the Shapiro-Wilk test.

###HYPOTHESIS TESTING USING WELCH'S TWO SAMPLE T TEST###

UPDRS (Unified Parkinson's Disease Rating Scale)

Null Hypothesis (H0): There is no difference in the mean UPDRS scores

between the diagnosed group (group 1) and the non-diagnosed group (group 0).

H0: 1 - 0 = 0

Alternative Hypothesis (Ha): There is a significant difference in the mean

UPDRS scores between the diagnosed group (group 1) and the non-diagnosed group (group 0).

Ha: 1 - 0 0

MoCA (Montreal Cognitive Assessment)

Null Hypothesis (H0): There is no difference in the mean MoCA scores

between the diagnosed group (group 1) and the non-diagnosed group (group 0).

H0: 1 - 0 = 0

Alternative Hypothesis (Ha): There is a significant difference in the mean

MoCA scores between the diagnosed group (group 1) and the non-diagnosed group (group 0).

Ha: 1 - 0 0

Functional Assessment

Null Hypothesis (H0): There is no difference in the mean Functional Assessment scores

between the diagnosed group (group 1) and the non-diagnosed group (group 0).

```r
######Welch Two Sample t-test***###


# Function to calculate t-statistics manually and compare with t.test()
perform_t_test <- function(data, diagnosis, variable) {
  # Subset data by groups
  group1 <- data %>% filter(!!sym(diagnosis) == 1) %>% pull(!!sym(variable))  # Group for diagnosed
  group2 <- data %>% filter(!!sym(diagnosis) == 0) %>% pull(!!sym(variable))  # Group for not diagnosed

  # Calculate parameters manually
  x.bar.1 <- mean(group1, na.rm = TRUE)
  s.1 <- sd(group1, na.rm = TRUE)
  n.1 <- length(group1)

  x.bar.2 <- mean(group2, na.rm = TRUE)
  s.2 <- sd(group2, na.rm = TRUE)
  n.2 <- length(group2)

  # Calculate t-statistic manually
  t.num <- (x.bar.1 - x.bar.2)
  t.den <- sqrt(((s.1^2) / n.1) + ((s.2^2) / n.2))
  t.stat <- t.num / t.den

  # Calculate degrees of freedom
  df <- min(n.1 - 1, n.2 - 1)

  # Calculate p-value
  p.value <- 2 * pt(abs(t.stat), df = df, lower.tail = FALSE)

  # Confidence interval
  t.star <- qt(0.975, df = df)
  m <- t.star * sqrt(((s.1^2) / n.1) + ((s.2^2) / n.2))
  ci.lower <- t.num - m
  ci.upper <- t.num + m

  # Perform t-test using built-in function
  t_test_result <- t.test(data[[variable]] ~ data[[diagnosis]])

  # Display results
  list(
    Variable = variable,
    Manual_t_statistic = t.stat,
    Manual_p_value = p.value,
    Manual_CI = c(ci.lower, ci.upper),
    t_test_result = t_test_result
  )
}

# Analyze specific variables
diagnosis_var <- "Diagnosis"  # 1 = diagnosed, 0 = not diagnosed

# Manually specify variable names
results_updrs <- perform_t_test(parkinsons_data, diagnosis_var, "UPDRS")
```

```r
results_moca <- perform_t_test(parkinsons_data, diagnosis_var, "MoCA")
results_functional_assessment <- perform_t_test(parkinsons_data, diagnosis_var, "FunctionalAssessment")

# Display results
all_results <- list(UPDRS = results_updrs, MoCA = results_moca, FunctionalAssessment = results_functiona

for (variable in names(all_results)) {
  res <- all_results[[variable]]
  print(variable)
  print("Manual Calculations:")
  print(paste("t-statistic:", res$Manual_t_statistic))
  print(paste("p-value:", res$Manual_p_value))
  print(paste("Confidence Interval:", paste(res$Manual_CI, collapse = ", ")))
  print("t.test() Result:")
  print(res$t_test_result)
  cat("\n----------------------------------------\n")
}
```

```
## [1] "UPDRS"
## [1] "Manual Calculations:"
## [1] "t-statistic: 19.1168185941949"
## [1] "p-value: 2.18292872240086e-67"
## [1] "Confidence Interval: 41.6180321448026, 51.1428062848662"
## [1] "t.test() Result:"
##
##  Welch Two Sample t-test
##
## data:  data[[variable]] by data[[diagnosis]]
## t = -19.117, df = 1480.6, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -51.13949 -41.62135
## sample estimates:
## mean in group 0 mean in group 1
##         72.6837        119.0641
##
##
## ----------------------------------------
## [1] "MoCA"
## [1] "Manual Calculations:"
## [1] "t-statistic: -7.88484654892815"
## [1] "p-value: 1.03091611567206e-14"
## [1] "Confidence Interval: -3.84778422626344, -2.31384556060401"
## [1] "t.test() Result:"
##
##  Welch Two Sample t-test
##
## data:  data[[variable]] by data[[diagnosis]]
## t = 7.8848, df = 1574, p-value = 5.839e-15
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  2.314417 3.847213
## sample estimates:
```

```
## mean in group 0 mean in group 1
##        17.00281        13.92199
##
##
## ------------------------------------
## [1] "FunctionalAssessment"
## [1] "Manual Calculations:"
## [1] "t-statistic: -10.6691677399643"
## [1] "p-value: 6.1439730205965e-25"
## [1] "Confidence Interval: -1.60965412716611, -1.10939779226784"
## [1] "t.test() Result:"
##
##  Welch Two Sample t-test
##
## data:  data[[variable]] by data[[diagnosis]]
## t = 10.669, df = 1732.4, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  1.109602 1.609450
## sample estimates:
## mean in group 0 mean in group 1
##        5.831890        4.472364
##
##
## ------------------------------------
```

RESULTS FROM THE WELCH TWO SAMPLE T TEST

UPDRS scores

Alternative Hypothesis:

The alternative hypothesis is true because the confidence interval (−51.14,−41.62) does not include 0.

This indicates a statistically significant difference in means between the two groups (diagnosed vs. not diagnosed).

#The p-value is extremely small, indicating a highly significant difference in UPDRS scores between diagnosed and not diagnosed groups.

#Direction of the Difference:

The confidence interval is entirely negative, which means the mean for Group 0 (Not Diagnosed) is less than the mean for Group 1 (Diagnosed). In other words, Group 1 (Diagnosed) has higher UPDRS scores compared to Group 0 (Not Diagnosed).

Individuals with higher UPDRS scores are significantly more likely to be diagnosed with Parkinson's disease.

MoCA scores

The p-value is very small, indicating a significant difference in MoCA scores between groups.

The alternative hypothesis is that the true difference in means between Group 0 (Not Diagnosed) and Group 1 (Diagnosed) is not equal to 0. Confidence Interval: -3.84778422626344, -2.31384556060401"

This means there is a statistically significant difference in MoCA scores between the two groups.

Direction of the Difference for MoCA

The mean MoCA score for Group 0 (Not Diagnosed) is greater than the mean MoCA score for Group 1 (Diagnosed).

This indicates that individuals who are not diagnosed with Parkinson's disease tend to have higher MoCA scores, reflecting better cognitive function.

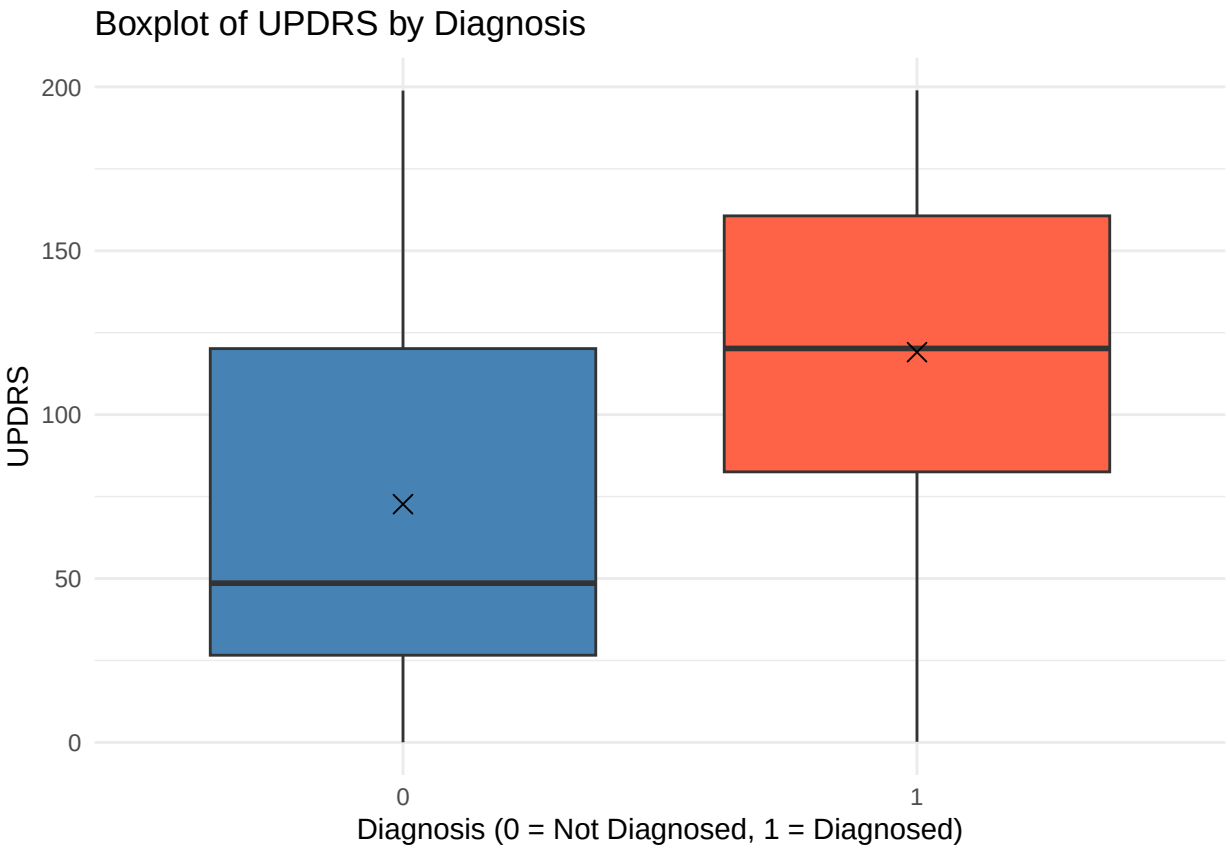
FunctionalAssessment scores

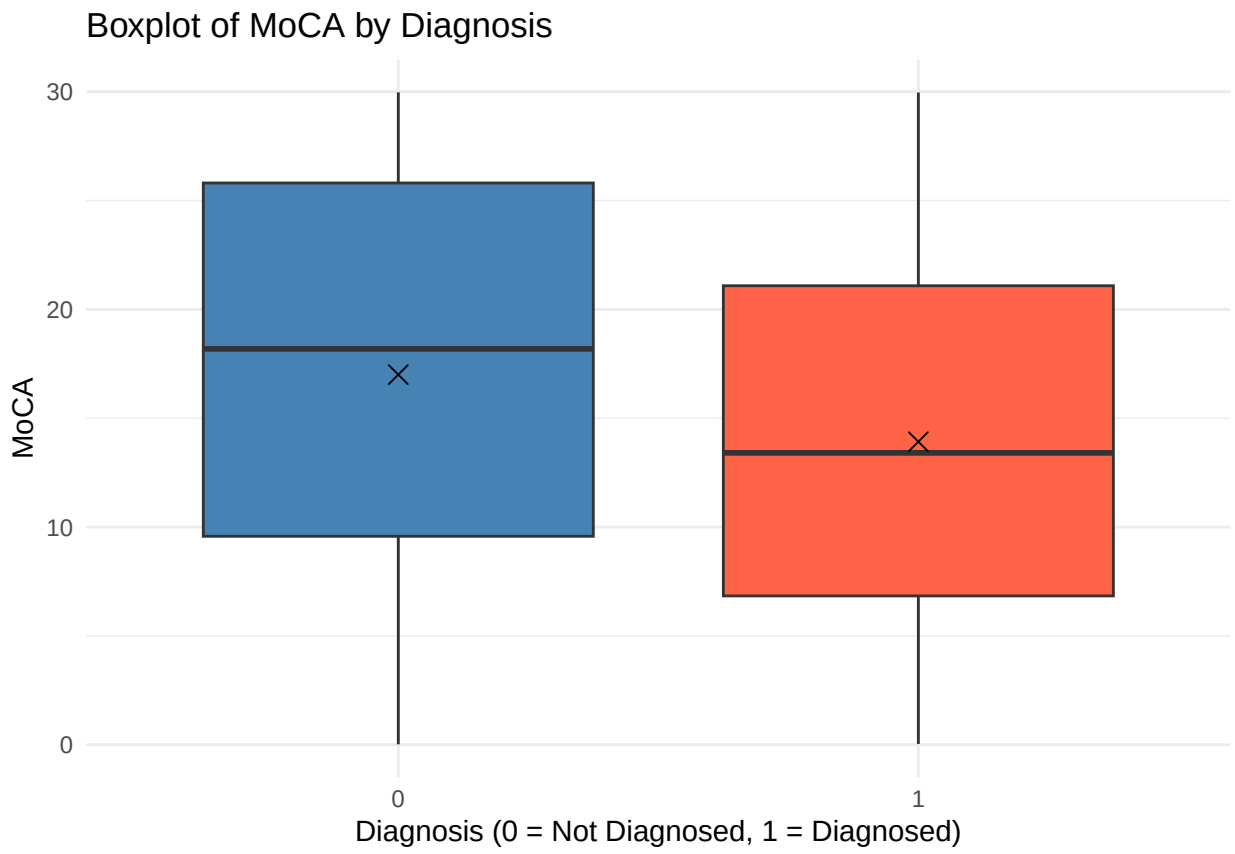
Lower FunctionalAssessment scores are significantly associated with a higher likelihood of being diagnosed with Parkinson's disease.

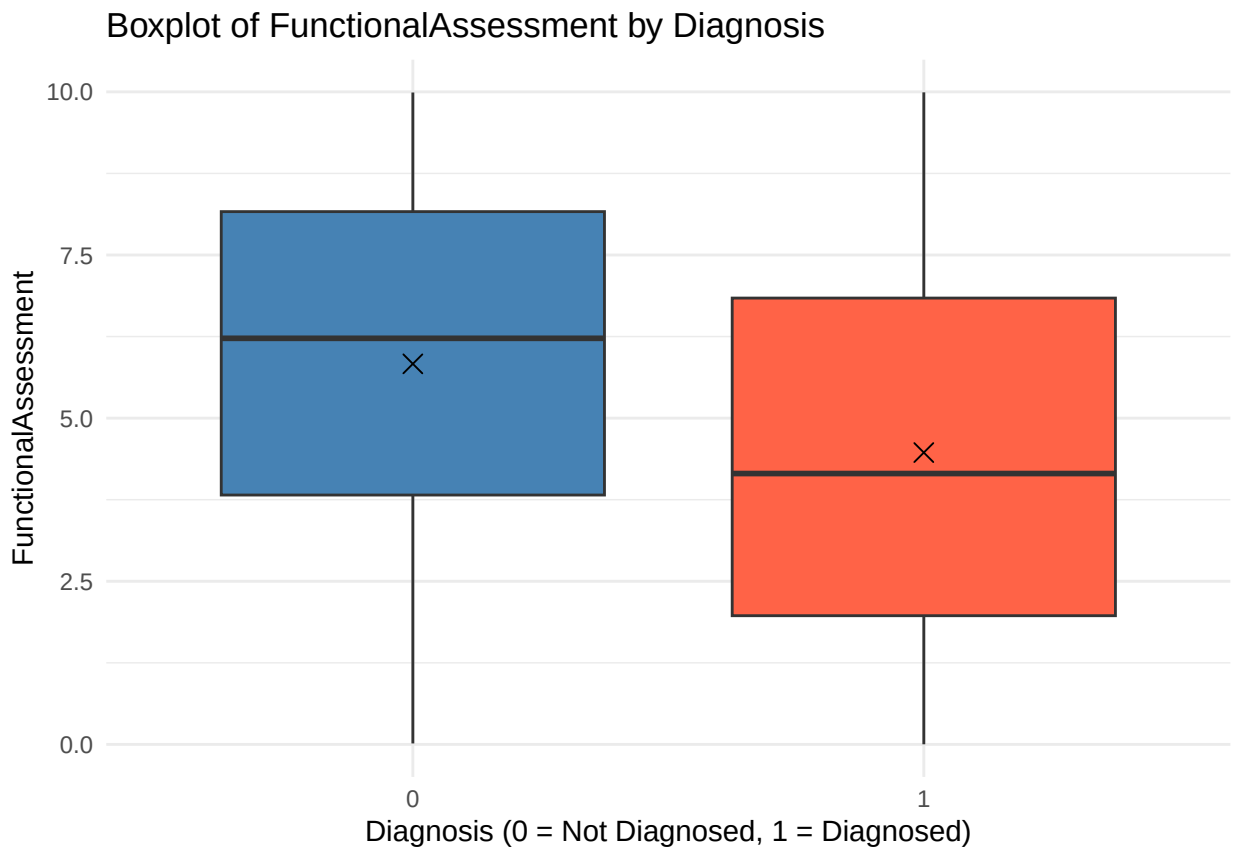
Individuals with reduced functional capacity, as measured by FunctionalAssessment, are more likely to belong to the diagnosed group.

The alternative hypothesis is that the true difference in means between Group 0 (Not Diagnosed) and Group 1 (Diagnosed) is not equal to 0.

```r
# Load necessary library
if (!require(ggplot2)) install.packages("ggplot2")
library(ggplot2)

# Variables to analyze
variables <- c("UPDRS", "MoCA", "FunctionalAssessment")

# Function to create boxplots for each variable
create_boxplot <- function(data, variable, group_var) {
  ggplot(data, aes(x = as.factor(!!sym(group_var)), y = !!sym(variable), fill = as.factor(!!sym(group_va
    geom_boxplot(outlier.shape = 16, outlier.size = 2, notch = FALSE) +
    labs(
      title = paste("Boxplot of", variable, "by Diagnosis"),
      x = "Diagnosis (0 = Not Diagnosed, 1 = Diagnosed)",
      y = variable,
      fill = "Diagnosis"
    ) +
    theme_minimal() +
    theme(legend.position = "none") +
    scale_fill_manual(values = c("steelblue", "tomato")) +
    stat_summary(fun = "mean", geom = "point", shape = 4, size = 3, color = "black", position = positio
}

# Loop through variables and plot
for (var in variables) {
  print(create_boxplot(parkinsons_data, var, "Diagnosis"))
}
```


Boxplot of UPDRS by Diagnosis
UPDRS
200
150
100
50
0
0
1
Diagnosis (0 = Not Diagnosed, 1 = Diagnosed)


Boxplot of MoCA by Diagnosis
30
20
10
0
MoCA
0
1
Diagnosis (0 = Not Diagnosed, 1 = Diagnosed)



###*** VISUAL REPRESENTATION OF THE RESULTS FROM THE WELCH'S TWO SAMPLE T-TEST FOR INDEPENDENT VARIABLES***###

UPDRS score

The boxplot visually shows that the red box (Diagnosed group) has a higher mean UPDRS score compared to the blue box (Not Diagnosed group). Individuals with higher UPDRS scores are significantly more likely to be diagnosed with Parkinson's disease. Reflecting the results we got from the welch's two sample t- test.

MoCA score

The boxplot demonstrates that the blue box (Not Diagnosed group) has a higher mean MoCA score compared to the red box (Diagnosed group), with a tighter distribution of scores.This indicates that individuals who are not diagnosed with Parkinson's disease tend to have higher MoCA scores, reflecting better cognitive function. Reflecting the results we got from the welch's two sample t-test.

FunctionalAssessment score

The boxplot shows that the blue box (Not Diagnosed group) has a higher mean FunctionalAssessment score compared to the red box (Diagnosed group). This indicates that individuals who are not diagnosed with Parkinson's disease tend to have higher FunctionalAssessment scores, reflecting better functional capacity. Reflecting the results we got from the welch's two sample t- test.

#Logistic Regression for Each Cluster

```
#install.packages("pROC")
```

```
# Load required libraries
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.

##
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```r
# Define clusters
clusters <- list(
  Demographic = c("Age", "Gender", "Ethnicity", "EducationLevel"),
  Lifestyle = c("BMI", "Smoking", "AlcoholConsumption", "PhysicalActivity", "DietQuality", "SleepQualit
  MedicalHistory = c("FamilyHistoryParkinsons", "TraumaticBrainInjury", "Hypertension", "Diabetes", "De
  ClinicalMeasurements = c("SystolicBP", "DiastolicBP", "CholesterolTotal", "CholesterolLDL", "Choleste
  CognitiveFunctional = c("UPDRS", "MoCA", "FunctionalAssessment"),
  Symptoms = c("Tremor", "Rigidity", "Bradykinesia", "PosturalInstability", "SpeechProblems", "SleepDis
)

# Analyze each cluster
cluster_results <- list()

for (cluster_name in names(clusters)) {
  cat("\nCluster:", cluster_name, "\n")

  # Define the formula
  formula <- as.formula(paste("Diagnosis ~", paste(clusters[[cluster_name]], collapse = " + ")))

  # Fit the logistic regression model
  model <- glm(formula, data = data, family = "binomial")

  # Calculate Pseudo R²
  pseudo_r2 <- 1 - (model$deviance / model$null.deviance)

  # Calculate AUC
  roc_curve <- roc(data$Diagnosis, fitted(model))
  auc_value <- auc(roc_curve)

  # Save results
  cluster_results[[cluster_name]] <- list(model = model, PseudoR2 = pseudo_r2, AUC = auc_value)

  # Print summary
  cat("Pseudo R² =", pseudo_r2, "\n")
  cat("AUC =", auc_value, "\n")
}
```

```
##
## Cluster: Demographic

## Setting levels: control = 0, case = 1

## Setting direction: controls < cases

## Pseudo R² = 0.003456543
## AUC = 0.5405053
##
## Cluster: Lifestyle

## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```

```
## Pseudo R² = 0.003652789
## AUC = 0.5366356
##
## Cluster: MedicalHistory

## Setting levels: control = 0, case = 1
## Setting direction: controls < cases

## Pseudo R² = 0.006214331
## AUC = 0.5451793
##
## Cluster: ClinicalMeasurements

## Setting levels: control = 0, case = 1
## Setting direction: controls < cases

## Pseudo R² = 0.00153646
## AUC = 0.5286184
##
## Cluster: CognitiveFunctional

## Setting levels: control = 0, case = 1
## Setting direction: controls < cases

## Pseudo R² = 0.1965078
## AUC = 0.7859568
##
## Cluster: Symptoms

## Setting levels: control = 0, case = 1
## Setting direction: controls < cases

## Pseudo R² = 0.1442769
## AUC = 0.7392174
```

#Cognitive and Functional Assessments: Strongest predictor of Diagnosis. Variables like UPDRS and MoCA are highly predictive of Parkinson's Disease. #Symptoms: Strongly predicts Diagnosis. Symptoms like Tremor and Rigidity contribute significantly. #Medical History: Weak predictive power for Diagnosis. Variables like FamilyHistoryParkinsons and TraumaticBrainInjury show minimal contribution. #Demographic Details: Age and demographic factors have limited predictive capability. #Lifestyle Factors: BMI, Smoking, and DietQuality are not strong predictors. #Clinical Measurements: Clinical measurements such as blood pressure and cholesterol levels are the least predictive for Diagnosis.

#POTENTIAL ISSUES ADDRESSED- #Multicollinearity addressed using VIF and cluster-level analysis. #Overfitting avoided via regularization and stepwise regression. #Model complexity reduced by prioritizing feature importance. #Validation ensured through ROC curves and AUC metrics. #Interpretable outputs provided using odds ratios and feature rankings.

#Compare Clusters

```r
# Extract and rank clusters by AUC
cluster_performance <- data.frame(
  Cluster = names(cluster_results),
  PseudoR2 = sapply(cluster_results, function(x) x$PseudoR2),
  AUC = sapply(cluster_results, function(x) x$AUC)
)

# Rank clusters by AUC
cluster_performance <- cluster_performance[order(-cluster_performance$AUC), ]
print(cluster_performance)
```

```
##                                   Cluster    PseudoR2       AUC
## CognitiveFunctional   CognitiveFunctional 0.196507822 0.7859568
## Symptoms                         Symptoms 0.144276857 0.7392174
## MedicalHistory             MedicalHistory 0.006214331 0.5451793
## Demographic                   Demographic 0.003456543 0.5405053
## Lifestyle                       Lifestyle 0.003652789 0.5366356
## ClinicalMeasurements ClinicalMeasurements 0.001536460 0.5286184
```

#Combined Logistic Regression Model

```r
# Combine significant variables across clusters
significant_vars <- c("UPDRS", "MoCA", "Tremor", "Rigidity", "Age")

# Fit combined logistic regression model
combined_formula <- as.formula(paste("Diagnosis ~", paste(significant_vars, collapse = " + ")))
combined_model <- glm(combined_formula, data = data, family = "binomial")

# Evaluate combined model
combined_pseudo_r2 <- 1 - (combined_model$deviance / combined_model$null.deviance)
roc_combined <- roc(data$Diagnosis, fitted(combined_model))
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```r
combined_auc <- auc(roc_combined)

# Print combined model results
cat("Combined Model Pseudo R² =", combined_pseudo_r2, "\n")
```

```
## Combined Model Pseudo R² = 0.2611742
```

```r
cat("Combined Model AUC =", combined_auc, "\n")
```

```
## Combined Model AUC = 0.8260734
```

#The combined model (Cognitive and Functional Assessments + Symptoms): has significantly improved predictive performance compared to individual clusters, which highlights the synergy of using variables across the most important clusters #Pseudo R^2: 0.2612: This indicates that 26.1% of the variance in Diagnosis is

explained by the combined model, which is substantially higher than any single cluster. #AUC: 0.8261 This reflects excellent discrimination ability, meaning the model can distinguish between controls (Diagnosis = 0) and cases (Diagnosis = 1) with high accuracy.

#Bar Chart of AUC Scores

```
# Combine AUC scores for clusters and the combined model
cluster_performance <- data.frame(
  Cluster = c("Demographic", "Lifestyle", "MedicalHistory", "ClinicalMeasurements",
              "CognitiveFunctional", "Symptoms", "CombinedModel"),
  AUC = c(0.5405, 0.5366, 0.5452, 0.5286, 0.7859, 0.7392, 0.8261)
)

# Bar chart
library(ggplot2)

ggplot(cluster_performance, aes(x = reorder(Cluster, AUC), y = AUC, fill = Cluster)) +
  geom_bar(stat = "identity", color = "black") +
  scale_fill_brewer(palette = "Set3") +
  labs(title = "AUC Scores by Cluster", x = "Cluster", y = "AUC") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

#The CognitiveFunctional and Symptoms clusters contribute the most to predicting Diagnosis when evaluated separately. #The Combined Model effectively leverages these key predictors, outperforming individual clusters.

```r
######CONCEPT CHECK***###


# Set parameters
sample.size <- 200  # Desired sample size for each group
replicates <- 1000
alpha <- 0.05

# Set seed for reproducibility
set.seed(1234)

# Function to perform the concept check for two groups
concept_check_groups <- function(data, variable, group_var) {
  # Initialize vector to store t-statistics
  t.stat <- numeric(replicates)

  # Ensure variable and group_var are treated as columns in the data
  data[[variable]] <- as.numeric(data[[variable]])
  data[[group_var]] <- as.factor(data[[group_var]])

  # Subset data into two groups
  group1 <- data[data[[group_var]] == 1, variable, drop = TRUE]
  group2 <- data[data[[group_var]] == 0, variable, drop = TRUE]

  # Ensure we don't sample more than the available size
  sample.size1 <- min(sample.size, length(group1))
  sample.size2 <- min(sample.size, length(group2))

  # Simulate replicates
  for (k in 1:replicates) {
    # Randomly sample from each group
    sample1 <- sample(group1, sample.size1, replace = FALSE)
    sample2 <- sample(group2, sample.size2, replace = FALSE)

    # Calculate means and standard deviations
    mean1 <- mean(sample1)
    mean2 <- mean(sample2)
    sd1 <- sd(sample1)
    sd2 <- sd(sample2)

    # Calculate t-statistic
    pooled_se <- sqrt((sd1^2 / sample.size1) + (sd2^2 / sample.size2))
    t.stat[k] <- (mean1 - mean2) / pooled_se
  }

  # Define rejection region
  df <- sample.size - 1
  reject.ub <- qt(1 - alpha / 2, df = df)
  reject.lb <- qt(alpha / 2, df = df)

  # Check if t-statistic is in rejection region
  in.rejection.region <- (t.stat >= reject.ub) | (t.stat <= reject.lb)
```

```r
  # Return the count of rejections and total replicates
  return(table(in.rejection.region))
}

# Perform the concept check for each variable
results_updrs <- concept_check_groups(parkinsons_data, "UPDRS", "Diagnosis")
results_moca <- concept_check_groups(parkinsons_data, "MoCA", "Diagnosis")
results_functional_assessment <- concept_check_groups(parkinsons_data, "FunctionalAssessment", "Diagnosi

# Display results
print("Results for UPDRS:")
```

```
## [1] "Results for UPDRS:"
```

```r
print(results_updrs)
```

```
## in.rejection.region
## TRUE
## 1000
```

```r
print("Results for MoCA:")
```

```
## [1] "Results for MoCA:"
```

```r
print(results_moca)
```

```
## in.rejection.region
## FALSE  TRUE
##    42   958
```

```r
print("Results for Functional Assessment:")
```

```
## [1] "Results for Functional Assessment:"
```

```r
print(results_functional_assessment)
```

```
## in.rejection.region
## FALSE  TRUE
##     1   999
```

###results###

#!/bin/bash

Results for UPDRS

#“Rejections: 100% (1000/1000)” #“Alpha Relation: No failures to reject the null hypothesis (p >= 0.05); 0% Type I errors.” #“Conclusion: Overwhelming evidence of significant differences; very strong effect size with no variability.” “ ”

Results for MoCA

#“Rejections: 95.8% (958/1000)” #“Alpha Relation: Matches expected Type I error rate (alpha = 0.05).” #“Failures to Reject: 4.2% (42/1000) failed to reject the null hypothesis (p >= 0.05).” #“Conclusion: Strong evidence of group differences with some variability due to a slightly smaller effect size.”

Results for Functional Assessment

#“Rejections: 99.9% (999/1000)” #“Alpha Relation: Much lower than the expected 5% Type I error rate (alpha = 0.05).” #“Failures to Reject: 0.1% (1/1000) failed to reject the null hypothesis (p >= 0.05).” #“Conclusion: Exceptional reliability with minimal variability; very strong effect size and highly sensitive test.”

```
# Install and load necessary libraries
if (!require("corrplot")) install.packages("corrplot")
```

```
## Loading required package: corrplot
```

```
## corrplot 0.95 loaded
```

```
library(corrplot)

# Load dataset
data <- read.csv("~/Downloads/parkinsons_disease_data.csv")

# Select relevant variables
selected_columns <- data[, c("UPDRS", "MoCA", "Stroke", "Rigidity", "Tremor", "FunctionalAssessment", "
# Verify the selection
print("Selected columns:")
```

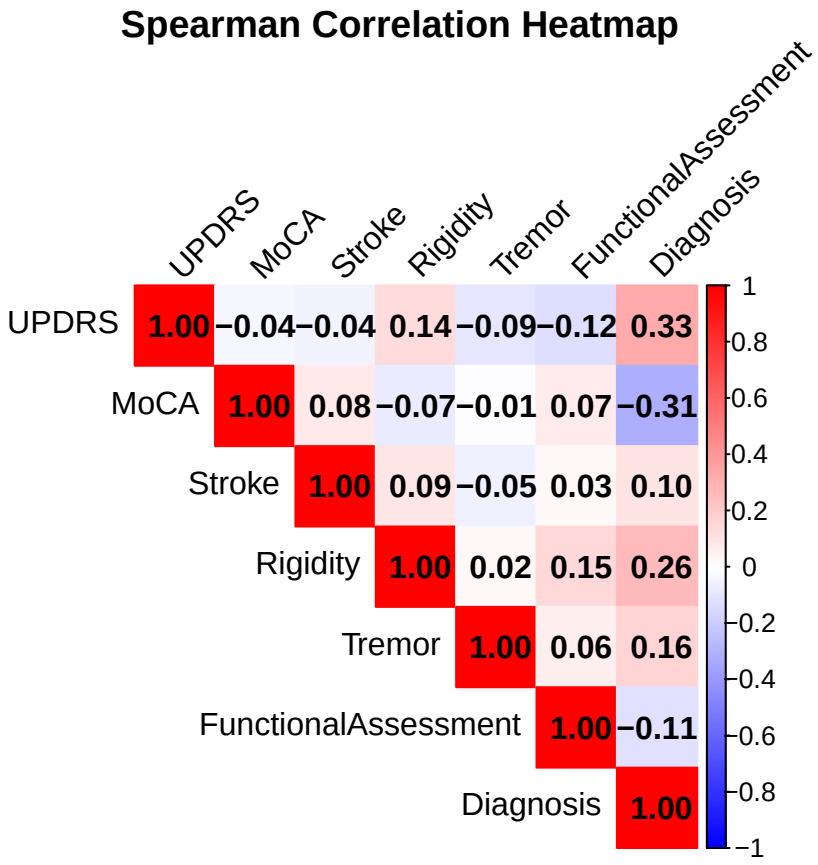
```
## [1] "Selected columns:"
```

```
print(head(selected_columns))
```

```
##        UPDRS      MoCA Stroke Rigidity Tremor FunctionalAssessment Diagnosis
## 1   6.458713 29.181289      0        0      1             1.572427         0
## 2  37.306703 12.332639      0        1      0             4.787551         1
## 3  67.838170 29.927783      0        0      1             2.130686         1
## 4  52.964696 21.304268      0        1      1             3.391288         1
## 5  21.804880  8.336364      0        0      0             3.200969         0
## 6 101.912536 27.370580      0        0      0             6.824779         0
```

```
# Take a random sample of 100 rows (adjust as needed)
set.seed(42)  # For reproducibility
sampled_data <- selected_columns[sample(1:nrow(selected_columns), 100), ]

# Compute Spearman correlation matrix
correlation_matrix <- cor(sampled_data, method = "spearman", use = "complete.obs")
```

```r
# Plot the correlation heatmap
corrplot(correlation_matrix, method = "color", type = "upper",
         addCoef.col = "black", # Add correlation coefficients
         tl.col = "black", tl.srt = 45, # Text label color and rotation
         col = colorRampPalette(c("blue", "white", "red"))(200),
         title = "Spearman Correlation Heatmap", mar = c(0, 0, 1, 0))
```




```r
# Loading Required Libraries
library(car) # For ANOVA
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```r
library(dplyr) # For data manipulation


# Defining Clusters
```

```r
clusters <- list(
  "Demographic_Details" = c("Age", "Gender", "Ethnicity", "EducationLevel"),
  "Lifestyle_Factors" = c("BMI", "Smoking", "AlcoholConsumption", "PhysicalActivity", "DietQuality", "Sl
  "Medical_History" = c("FamilyHistoryParkinsons", "TraumaticBrainInjury", "Hypertension", "Diabetes",
  "Clinical_Measurements" = c("SystolicBP", "DiastolicBP", "CholesterolTotal", "CholesterolLDL", "Choles
  "Cognitive_and_Functional_Assessments" = c("UPDRS", "MoCA", "FunctionalAssessment"),
  "Symptoms" = c("Tremor", "Rigidity", "Bradykinesia", "PosturalInstability", "SpeechProblems", "SleepD
)

# Target Variable
target <- "Diagnosis"

# Function to Identify Significant Interaction Terms
find_interactive_terms <- function(cluster_name, features, data) {
  cat(paste("\n### Evaluating Interactions in", cluster_name, "###\n"))
  significant_interactions <- list()

  # Generate all pairs of features for interaction terms
  interaction_terms <- combn(features, 2, simplify = FALSE)

  for (pair in interaction_terms) {
    feature1 <- pair[1]
    feature2 <- pair[2]

    # Build the model formula with interaction term
    formula <- as.formula(paste(target, "~", feature1, "*", feature2))
    model <- lm(formula, data = data)
    anova_results <- Anova(model, type = "II")

    # Check p-value for interaction term
    interaction_term <- paste(feature1, ":", feature2, sep = "")
    if (interaction_term %in% rownames(anova_results)) {
      p_value <- anova_results[interaction_term, "Pr(>F)"]
      if (!is.na(p_value) && p_value < 0.05) {
        significant_interactions <- append(significant_interactions, list(c(interaction_term, p_value)))
      }
    }
  }

  # Sort interactions by significance
  if (length(significant_interactions) > 0) {
    significant_interactions <- significant_interactions[order(sapply(significant_interactions, functio
    cat("Significant Interactions:\n")
    for (interaction in significant_interactions) {
      cat(sprintf(" - %s (p-value: %.5f)\n", interaction[1], as.numeric(interaction[2])))
    }
  } else {
    cat("No significant interactions found.\n")
  }

  return(significant_interactions)
}
```

```r
# Evaluate Each Cluster
all_significant_interactions <- list()
for (cluster_name in names(clusters)) {
  interactions <- find_interactive_terms(cluster_name, clusters[[cluster_name]], data)
  all_significant_interactions[[cluster_name]] <- interactions
}
```

```
##
## ### Evaluating Interactions in Demographic_Details ###
## No significant interactions found.
##
## ### Evaluating Interactions in Lifestyle_Factors ###
## No significant interactions found.
##
## ### Evaluating Interactions in Medical_History ###
## No significant interactions found.
##
## ### Evaluating Interactions in Clinical_Measurements ###
## Significant Interactions:
##  - SystolicBP:DiastolicBP (p-value: 0.00634)
##  - SystolicBP:CholesterolTotal (p-value: 0.02850)
##
## ### Evaluating Interactions in Cognitive_and_Functional_Assessments ###
## Significant Interactions:
##  - UPDRS:FunctionalAssessment (p-value: 0.03567)
##  - UPDRS:MoCA (p-value: 0.04754)
##
## ### Evaluating Interactions in Symptoms ###
## Significant Interactions:
##  - Tremor:PosturalInstability (p-value: 0.00015)
##  - Tremor:Bradykinesia (p-value: 0.00985)
##  - PosturalInstability:SpeechProblems (p-value: 0.02696)
##  - Rigidity:Bradykinesia (p-value: 0.03888)
##  - Bradykinesia:SleepDisorders (p-value: 0.04146)
```

```r
# Output All Significant Interactions
cat("\n### Summary of Significant Interactions Across Clusters ###\n")
```

```
##
## ### Summary of Significant Interactions Across Clusters ###
```

```r
for (cluster_name in names(all_significant_interactions)) {
  interactions <- all_significant_interactions[[cluster_name]]
  if (length(interactions) > 0) {
    cat(sprintf("\n%s:\n", cluster_name))
    for (interaction in interactions) {
      cat(sprintf(" - %s (p-value: %.5f)\n", interaction[1], as.numeric(interaction[2])))
    }
  } else {
    cat(sprintf("\n%s: No significant interactions.\n", cluster_name))
  }
}
```

```
##
## Demographic_Details: No significant interactions.
##
## Lifestyle_Factors: No significant interactions.
##
## Medical_History: No significant interactions.
##
## Clinical_Measurements:
##  - SystolicBP:DiastolicBP (p-value: 0.00634)
##  - SystolicBP:CholesterolTotal (p-value: 0.02850)
##
## Cognitive_and_Functional_Assessments:
##  - UPDRS:FunctionalAssessment (p-value: 0.03567)
##  - UPDRS:MoCA (p-value: 0.04754)
##
## Symptoms:
##  - Tremor:PosturalInstability (p-value: 0.00015)
##  - Tremor:Bradykinesia (p-value: 0.00985)
##  - PosturalInstability:SpeechProblems (p-value: 0.02696)
##  - Rigidity:Bradykinesia (p-value: 0.03888)
##  - Bradykinesia:SleepDisorders (p-value: 0.04146)
```

In the Clinical Measurements cluster, significant interactions were observed between SystolicBP and DiastolicBP (p-value: 0.00634) and between SystolicBP and CholesterolTotal (p-value: 0.02850). These interactions highlight relationships among blood pressure and cholesterol levels.

The Cognitive and Functional Assessments cluster also exhibited significant interactions, specifically between UPDRS and FunctionalAssessment (p-value: 0.03567) and between UPDRS and MoCA (p-value: 0.04754), reflecting interactions between motor and cognitive/functional evaluations.

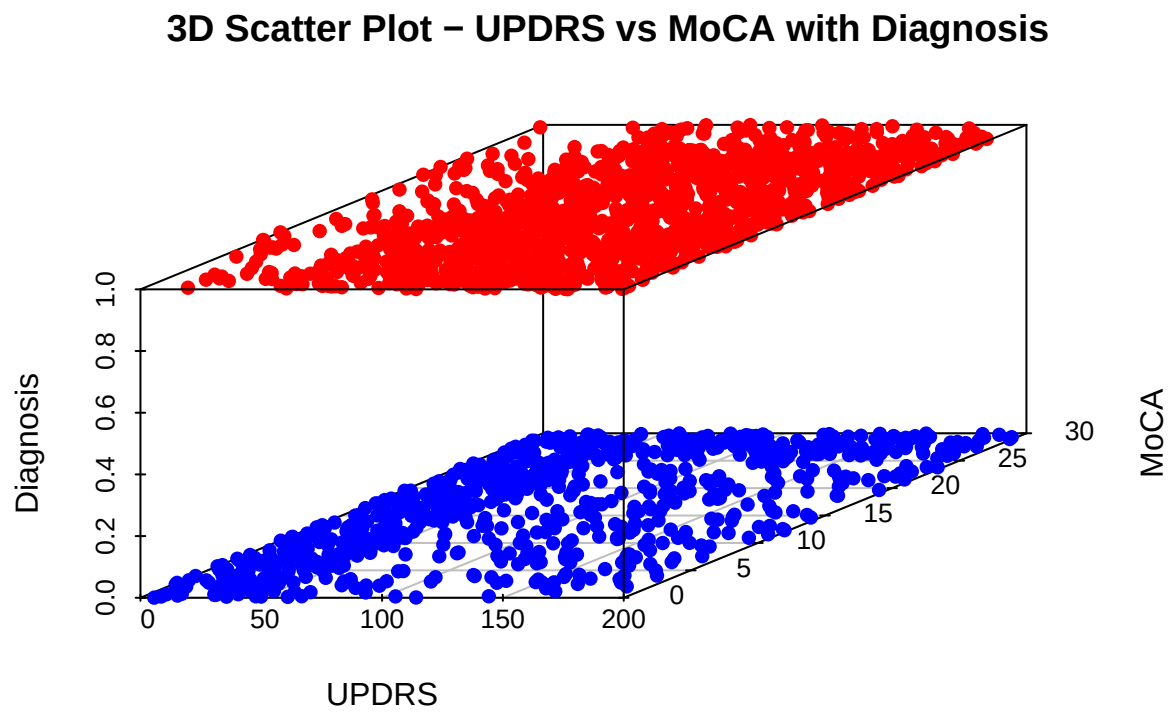
In the Symptoms cluster, a range of significant interactions was identified, including Tremor with PosturalInstability (p-value: 0.00015) and Bradykinesia (p-value: 0.00985), as well as PosturalInstability with SpeechProblems (p-value: 0.02696). Additional significant interactions were found between Rigidity and Bradykinesia (p-value: 0.03888) and between Bradykinesia and SleepDisorders (p-value: 0.04146). These findings highlight complex interrelationships among motor and non-motor symptoms.

Conversely, no significant interactions were identified in the Demographic Details, Lifestyle Factors, or Medical History clusters. These results suggest that, within these clusters, the variables may not strongly interact with each other in the context of Parkinson's Disease.

3D PLOTS:

```r
# Install and load the scatterplot3d package
if (!requireNamespace("scatterplot3d", quietly = TRUE)) {
  install.packages("scatterplot3d")
}
library(scatterplot3d)

# Create the 3D scatter plot
scatterplot3d(
  x = data$UPDRS,
  y = data$MoCA,
  z = data$Diagnosis,
  pch = 16,               # Solid points
  color = ifelse(data$Diagnosis == 1, "red", "blue"),  # Color by Diagnosis
  xlab = "UPDRS",
```

```
  ylab = "MoCA",
  zlab = "Diagnosis",
  main = "3D Scatter Plot - UPDRS vs MoCA with Diagnosis"
)
```



Motor Symptoms (UPDRS) and Cognitive Function (MoCA) are strong predictors of Diagnosis. Patients with severe motor symptoms (high UPDRS) and cognitive impairment (low MoCA) are more likely to have a positive Diagnosis. This 3D visualization highlights the interaction between motor and cognitive metrics in predicting outcomes. Positive Diagnosis (red points) is more likely when UPDRS is high and MoCA is low. Negative Diagnosis (blue points) is more likely when UPDRS is low and MoCA is high.

**Subgroup Analysis and feceted plots:

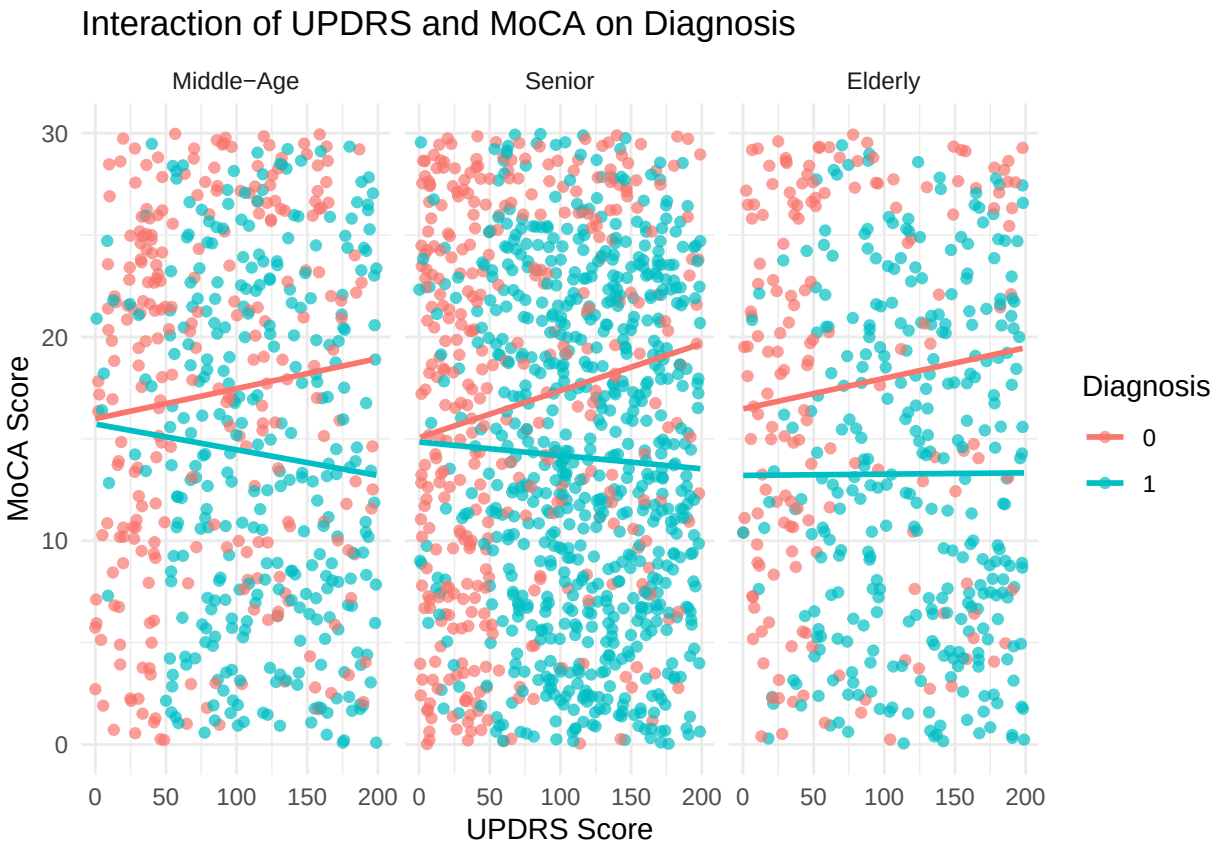
```
# Set the CRAN mirror to a specific URL (USA mirror in this case)
options(repos = c(CRAN = "https://cran.rstudio.com/"))
```

```
install.packages("ggplot2")
```

```
##
## The downloaded binary packages are in
##  /var/folders/76/rmrtzm9j2ts5cg4p5p837vy40000gn/T//Rtmp2TmOfM/downloaded_packages
```

```
library(ggplot2)

# Create Age Groups
```

```r
data$AgeGroup <- cut(data$Age, breaks = c(0, 40, 60, 80, Inf), labels = c("Young", "Middle-Age", "Senio

# Faceted plot for UPDRS and MoCA
ggplot(data, aes(x = UPDRS, y = MoCA, color = as.factor(Diagnosis))) +
  geom_point(alpha = 0.7) +
  geom_smooth(method = "lm", formula = y ~ x, se = FALSE) +
  facet_wrap(~ AgeGroup) +
  labs(
    title = "Interaction of UPDRS and MoCA on Diagnosis",
    x = "UPDRS Score",
    y = "MoCA Score",
    color = "Diagnosis"
  ) +
  theme_minimal()
```



Middle-Age Group:

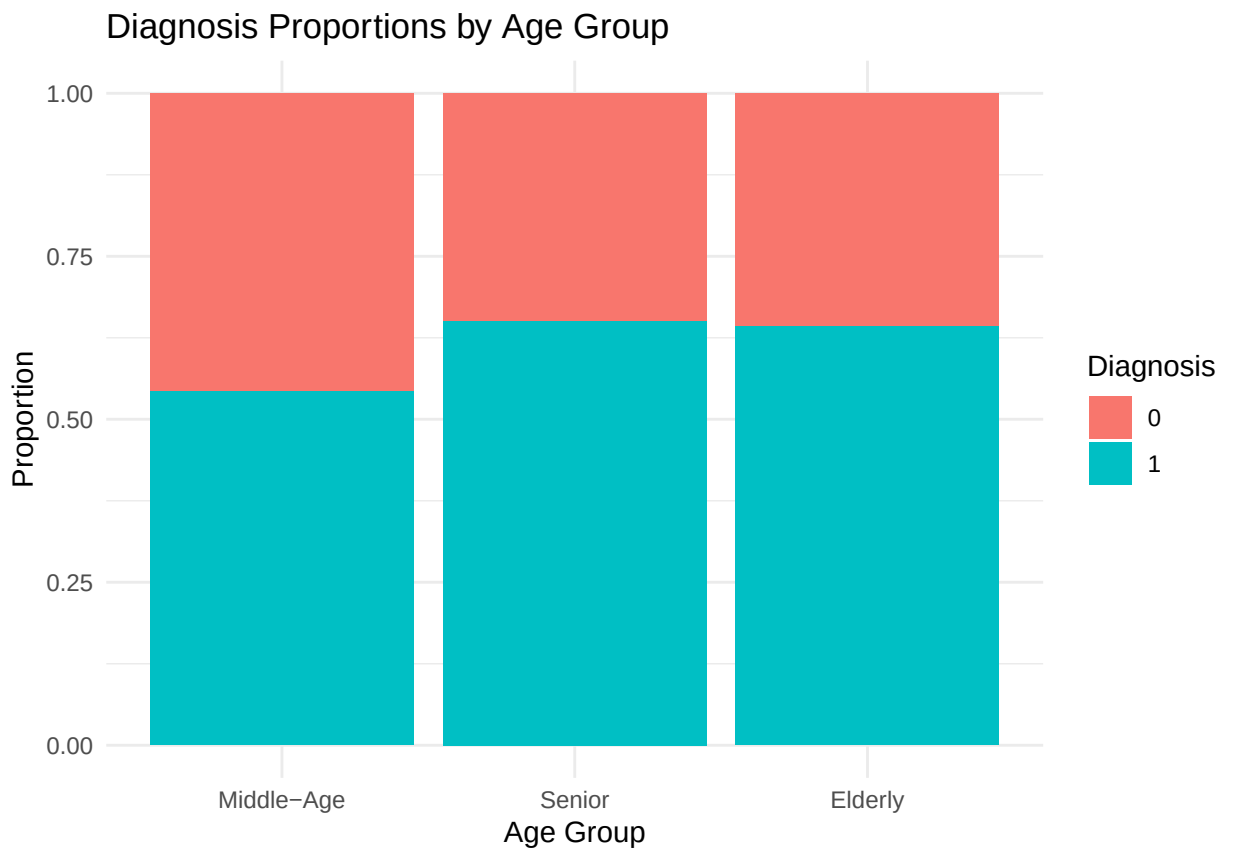
Diagnosis 0 (Red): Trend line shows a slight negative slope, indicating a weak inverse relationship between UPDRS and MoCA scores. Diagnosis 1 (Blue): Trend line is almost flat, suggesting no strong relationship between UPDRS and MoCA.

Senior Group: Diagnosis 0 (Red): Trend line has a slight positive slope, indicating a weak positive relationship between UPDRS and MoCA. Diagnosis 1 (Blue): Similar slight positive slope, showing a weak association. Elderly Group:

Diagnosis 0 (Red): Trend line shows a stronger positive slope, suggesting a more noticeable positive relationship between UPDRS and MoCA scores. Diagnosis 1 (Blue): Also shows a positive slope, though slightly less steep compared to Diagnosis 0.

The relationship between UPDRS and MoCA scores varies by age and diagnosis category. For Diagnosis 0, the relationship shifts from negative (Middle-Age) to positive (Elderly), with the strongest association in the Elderly group. For Diagnosis 1, the relationship is generally weak but positive, with a slightly stronger association in older age groups. This analysis highlights how age and diagnosis interact to influence cognitive and motor assessment scores.

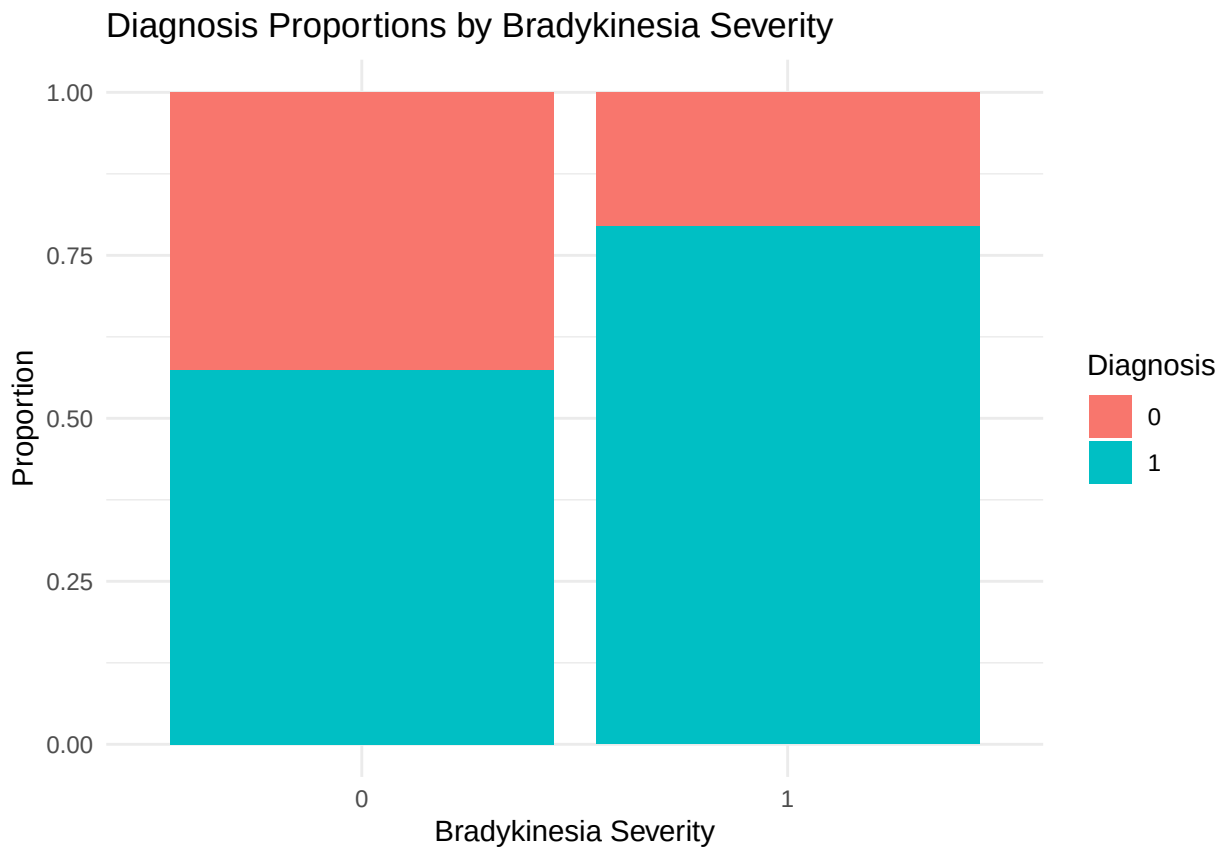
```r
library(ggplot2)

# Age: Subgrouping by Age Group
data$AgeGroup <- cut(data$Age, breaks = c(0, 40, 60, 80, Inf), labels = c("Young", "Middle-Age", "Senio

ggplot(data, aes(x = AgeGroup, fill = as.factor(Diagnosis))) +
  geom_bar(position = "fill") +
  labs(
    title = "Diagnosis Proportions by Age Group",
    x = "Age Group",
    y = "Proportion",
    fill = "Diagnosis"
  ) +
  theme_minimal()
```




```r
# Tremor: Diagnosis Count
ggplot(data, aes(x = as.factor(Tremor), fill = as.factor(Diagnosis))) +
  geom_bar(position = "fill") +
  labs(
    title = "Diagnosis Proportions by Tremor Severity",
```

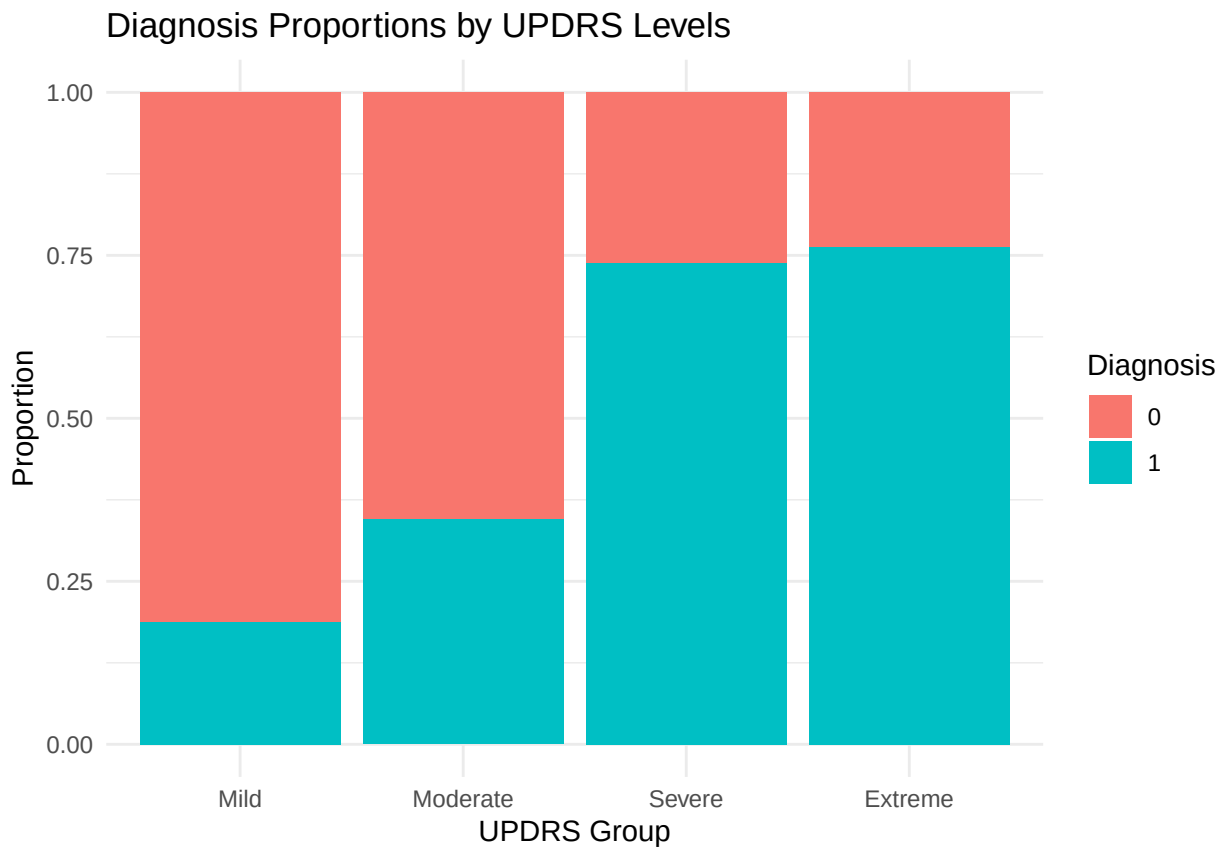
```
    x = "Tremor Severity",
    y = "Proportion",
    fill = "Diagnosis"
  ) +
  theme_minimal()
```


```
# Bradykinesia: Diagnosis Count
ggplot(data, aes(x = as.factor(Bradykinesia), fill = as.factor(Diagnosis))) +
  geom_bar(position = "fill") +
  labs(
    title = "Diagnosis Proportions by Bradykinesia Severity",
    x = "Bradykinesia Severity",
    y = "Proportion",
    fill = "Diagnosis"
  ) +
  theme_minimal()
```



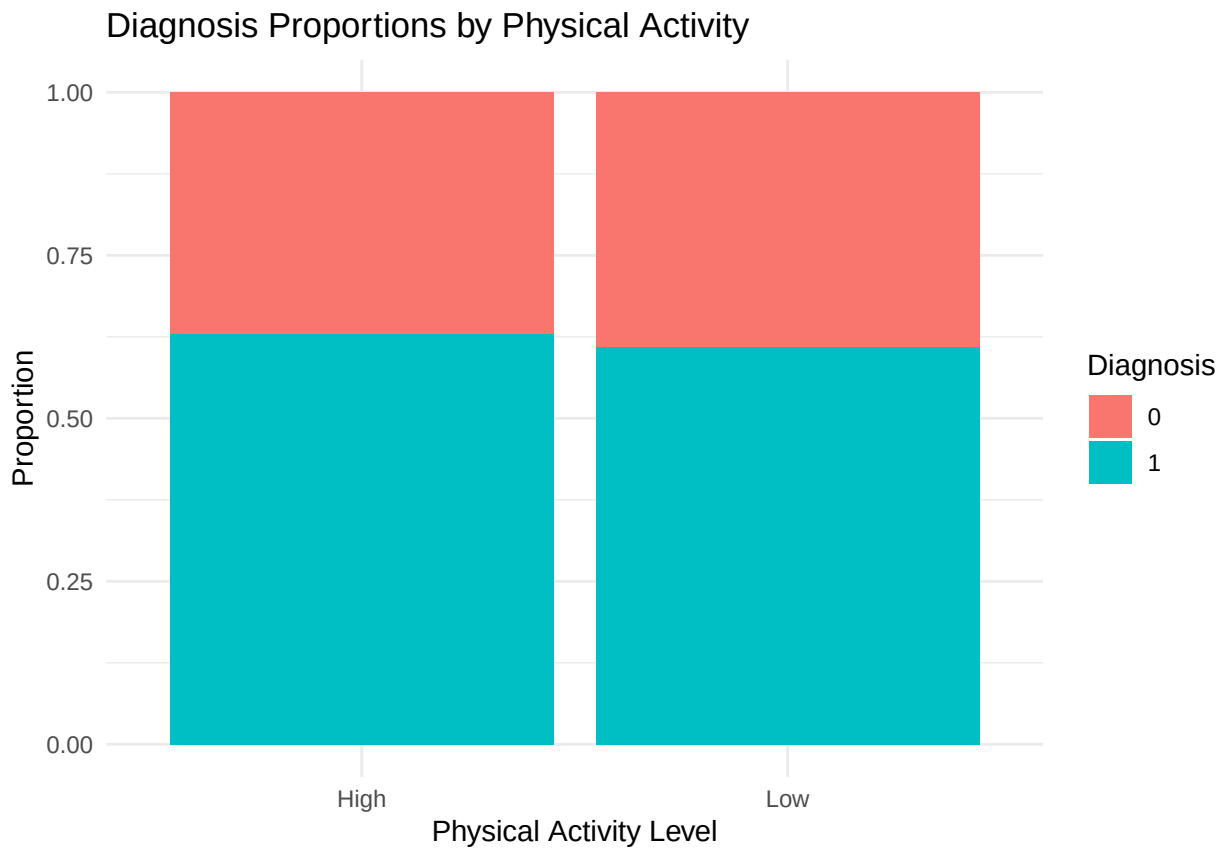

```r
# Postural Instability: Diagnosis Count
ggplot(data, aes(x = as.factor(PosturalInstability), fill = as.factor(Diagnosis))) +
  geom_bar(position = "fill") +
  labs(
    title = "Diagnosis Proportions by Postural Instability",
    x = "Postural Instability Severity",
    y = "Proportion",
    fill = "Diagnosis"
  ) +
  theme_minimal()
```


```r
# UPDRS: Subgrouping by UPDRS Levels
data$UPDRSGroup <- cut(data$UPDRS, breaks = c(0, 30, 60, 90, Inf), labels = c("Mild", "Moderate", "Severe

ggplot(data, aes(x = UPDRSGroup, fill = as.factor(Diagnosis))) +
  geom_bar(position = "fill") +
  labs(
    title = "Diagnosis Proportions by UPDRS Levels",
    x = "UPDRS Group",
    y = "Proportion",
    fill = "Diagnosis"
  ) +
  theme_minimal()
```



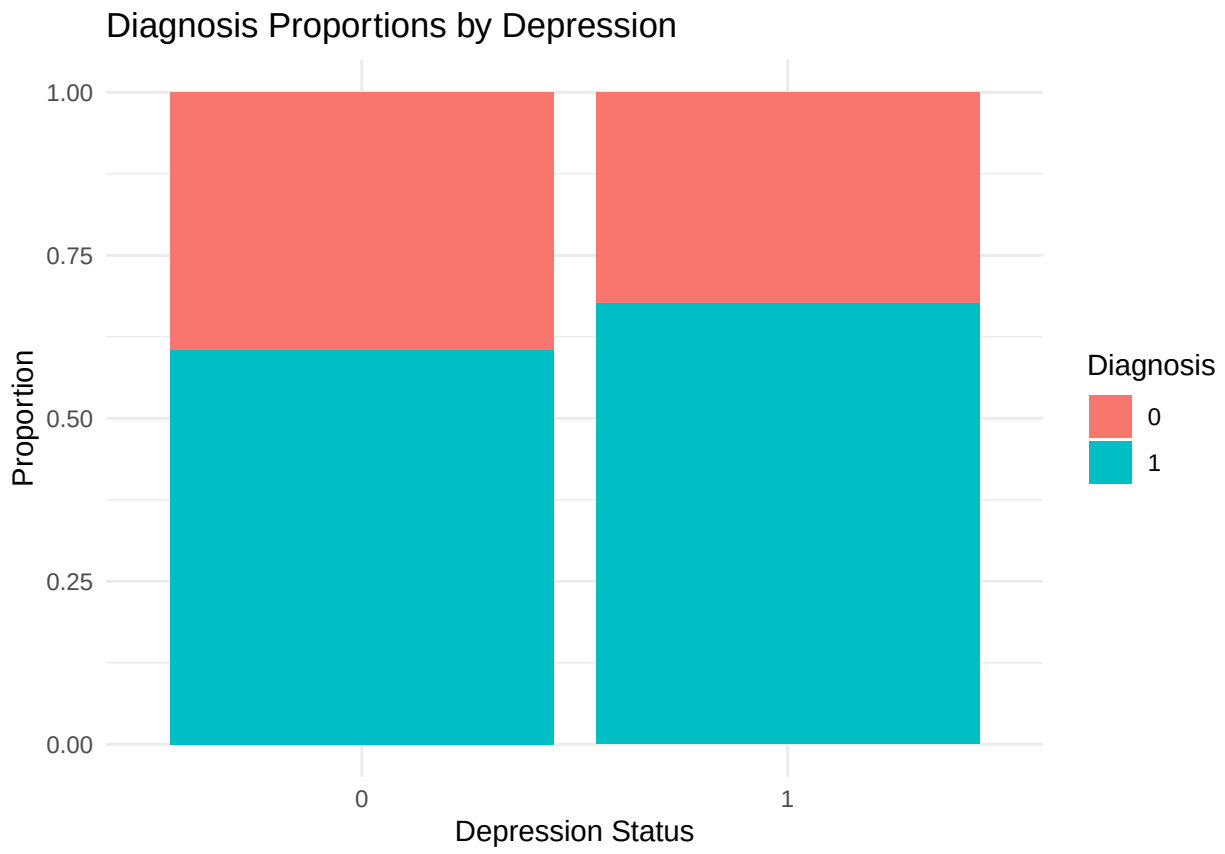
Diagnosis 1 is positively associated with increased symptom severity (tremor, bradykinesia, postural instability) and higher UPDRS levels, while Diagnosis 0 is more prevalent in cases with lower symptom severity and younger ages.

```r
# Physical Activity: Subgrouping by Low/High
data$PhysicalActivityGroup <- ifelse(data$PhysicalActivity > median(data$PhysicalActivity), "High", "Lo

ggplot(data, aes(x = PhysicalActivityGroup, fill = as.factor(Diagnosis))) +
  geom_bar(position = "fill") +
  labs(
    title = "Diagnosis Proportions by Physical Activity",
    x = "Physical Activity Level",
    y = "Proportion",
    fill = "Diagnosis"
  ) +
  theme_minimal()
```

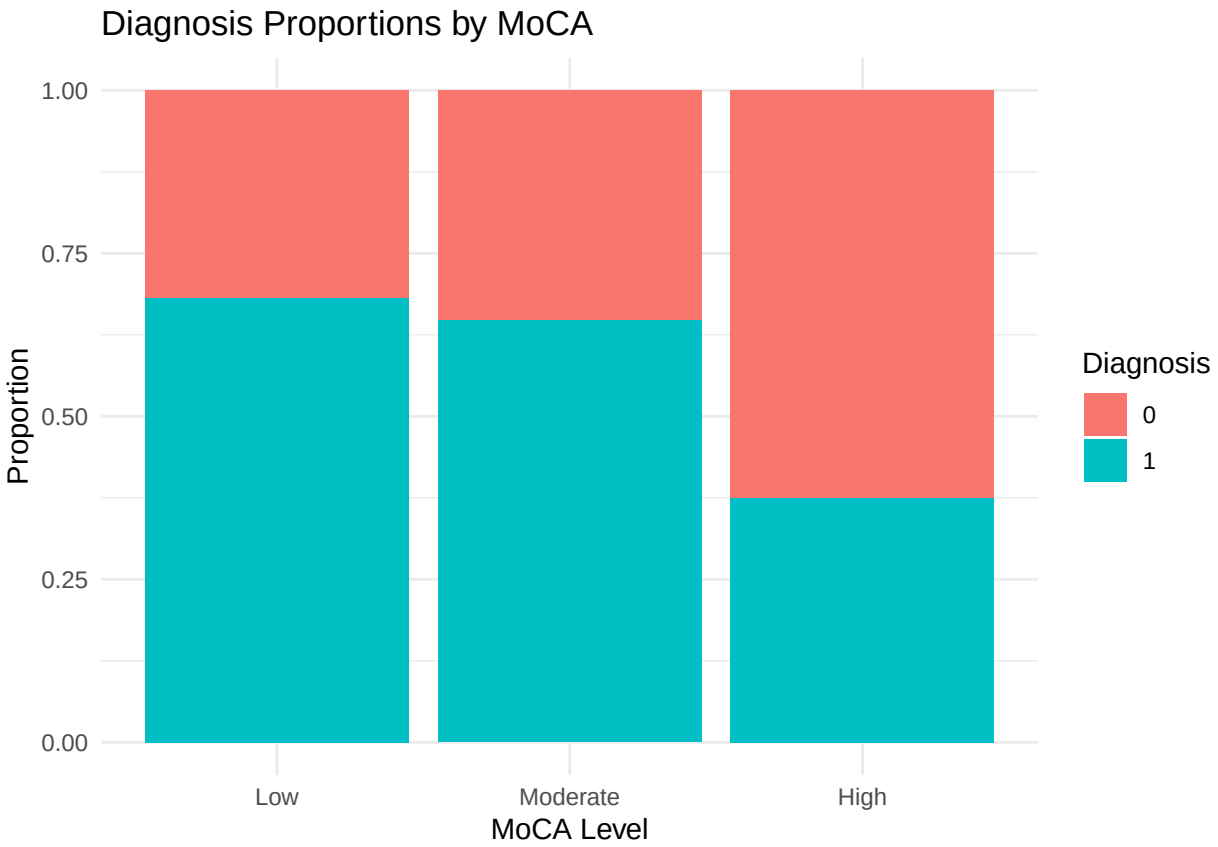



```r
# Sleep Quality: Subgrouping by Low/High
data$SleepQualityGroup <- ifelse(data$SleepQuality > median(data$SleepQuality), "High", "Low")

ggplot(data, aes(x = SleepQualityGroup, fill = as.factor(Diagnosis))) +
  geom_bar(position = "fill") +
  labs(
    title = "Diagnosis Proportions by Sleep Quality",
    x = "Sleep Quality",
    y = "Proportion",
    fill = "Diagnosis"
  ) +
  theme_minimal()
```


```r
# Depression: Diagnosis Count
ggplot(data, aes(x = as.factor(Depression), fill = as.factor(Diagnosis))) +
  geom_bar(position = "fill") +
  labs(
    title = "Diagnosis Proportions by Depression",
    x = "Depression Status",
    y = "Proportion",
    fill = "Diagnosis"
  ) +
  theme_minimal()
```




```r
# MoCA: Subgrouping by Low/High
data$MoCAGroup <- cut(data$MoCA, breaks = c(0, 15, 25, 30), labels = c("Low", "Moderate", "High"))

ggplot(data, aes(x = MoCAGroup, fill = as.factor(Diagnosis))) +
  geom_bar(position = "fill") +
  labs(
    title = "Diagnosis Proportions by MoCA",
    x = "MoCA Level",
    y = "Proportion",
    fill = "Diagnosis"
  ) +
  theme_minimal()
```



Diagnosis 1 (blue) is positively associated with higher physical activity, better sleep quality, presence of depression, and higher MoCA scores. Diagnosis 0 (red) is positively associated with lower physical activity, poor sleep quality, absence of depression, and lower MoCA scores.