Final project code

Group F

2023-12-07

Load data

```r
library(ggplot2)
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
library(tidyr)
library(pvclust)
library(cluster)
library(factoextra)
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

```r
# read data
data1 <- read.csv("data_1.csv", header=TRUE)
data1<- data1[,-1]

# Data source and structure description
cat("Data Structure Description:\n")
```

```
## Data Structure Description:
```

```r
str(data1)
```

```
## 'data.frame':    14728 obs. of  15 variables:
##  $ M2ID       : int  10005 10005 10005 10005 10005 10005 10005 10005 10015 10015 ...
##  $ B1SPWBU2   : num  48 48 48 48 48 48 48 48 38 38 ...
##  $ B1STINC1   : int  0 0 0 0 0 0 0 0 126250 126250 ...
##  $ B1PAGE_M2.x: int  80 80 80 80 80 80 80 80 53 53 ...
##  $ B1PGENDER  : int  2 2 2 2 2 2 2 2 2 2 ...
##  $ B2DN_STR   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ race       : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ marital    : int  0 0 0 0 0 0 0 0 1 1 ...
##  $ B1SQ2      : int  10 10 10 10 10 10 10 10 8 8 ...
##  $ B1SQ1      : int  10 10 10 10 10 10 10 10 7 7 ...
##  $ B1SQ3      : int  10 10 10 10 10 10 10 10 9 9 ...
##  $ B2DNEGAV   : num  0 0 0 0.07 0 0 0 0 0.14 0 ...
##  $ meanNA     : num  0.00875 0.00875 0.00875 0.00875 0.00875 ...
##  $ sdNA       : num  0.0247 0.0247 0.0247 0.0247 0.0247 ...
##  $ gender     : int  0 0 0 0 0 0 0 0 0 0 ...
```

EDA

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

# EDA
cat("Exploratory Data Analysis with Descriptive Statistics and Visualizations:\n")
```

The descriptive statistical analysis aims to understand sample characteristics

```
## Exploratory Data Analysis with Descriptive Statistics and Visualizations:
```

```r
# Age Distribution
cat("Age Distribution:\n")
```

```
## Age Distribution:
```

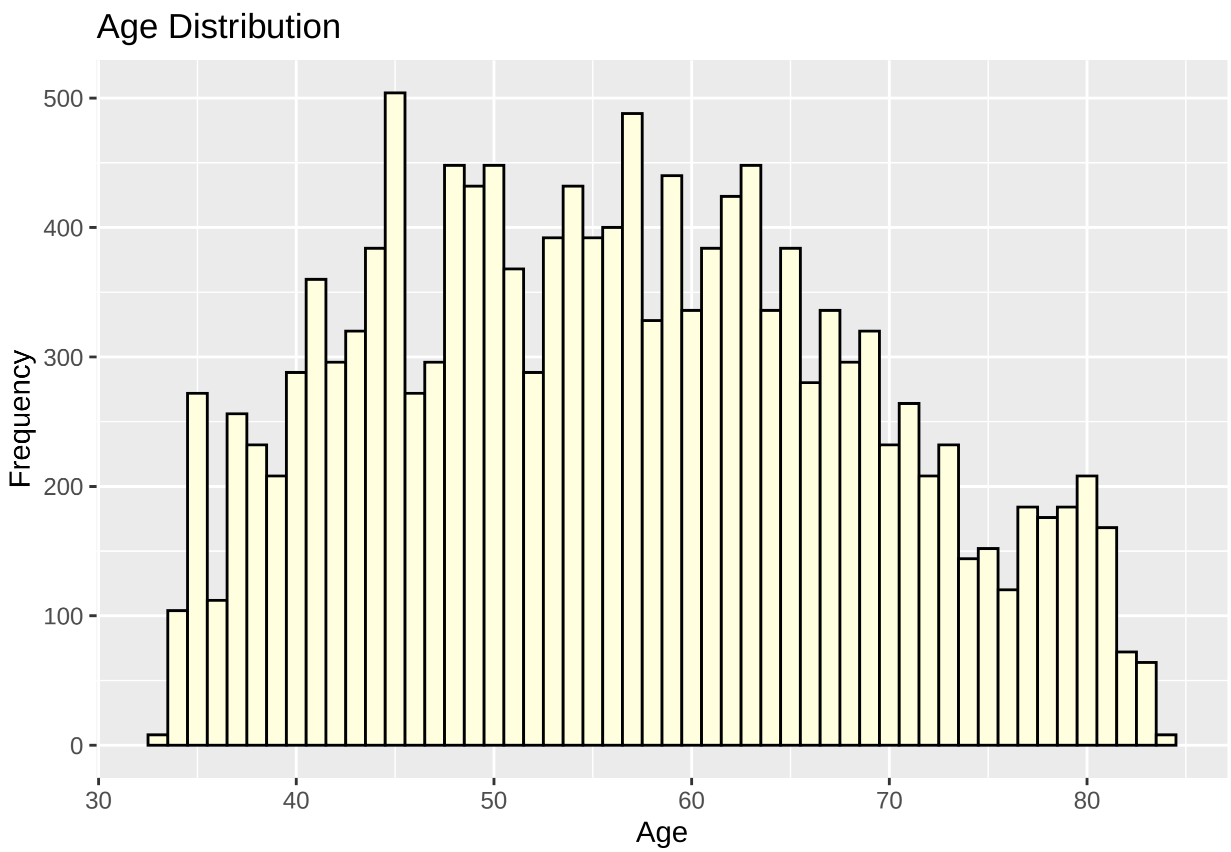
```r
summary(data1$B1PAGE_M2.x, na.rm = TRUE)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   33.00   47.00   56.00   56.51   65.00   84.00
```

```r
sd(data1$B1PAGE_M2.x, na.rm = TRUE)
```

```
## [1] 12.2322
```

```r
ggplot(data1, aes(x = B1PAGE_M2.x)) +
  geom_histogram(binwidth = 1, fill = "lightyellow", color = "black") +
  labs(title = "Age Distribution", x = "Age", y = "Frequency")
```



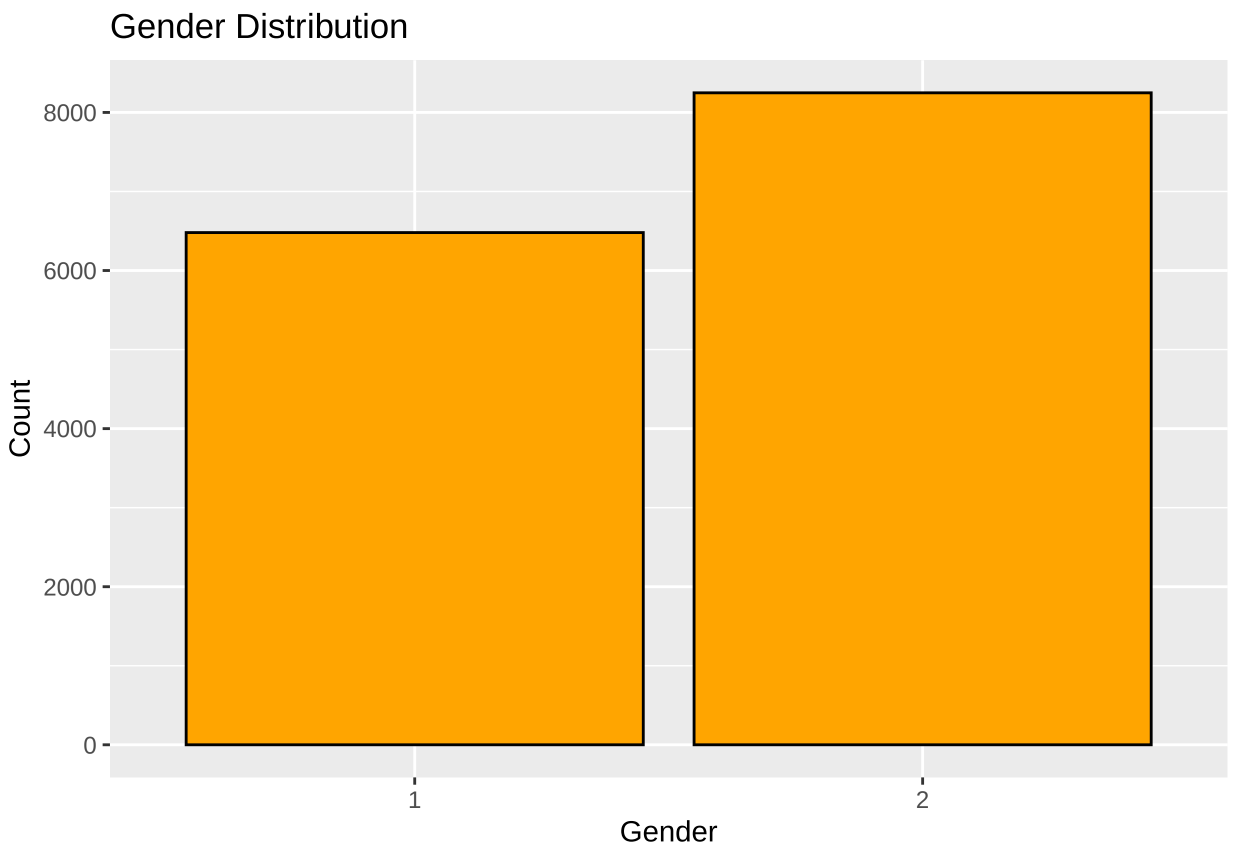
```r
# Gender Distribution
cat("Gender Distribution:\n")
```

```
## Gender Distribution:
```

```r
summary(data1$B1PGENDER, na.rm = TRUE)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00    1.00    2.00    1.56    2.00    2.00
```

```r
ggplot(data1, aes(x = factor(B1PGENDER))) +
  geom_bar(fill = "orange", color = "black") +
  labs(title = "Gender Distribution", x = "Gender", y = "Count")
```



```r
# Income Distribution
cat("Income Distribution:\n")
```

```
## Income Distribution:
```

```r
summary(data1$B1STINC1, na.rm = TRUE)
```

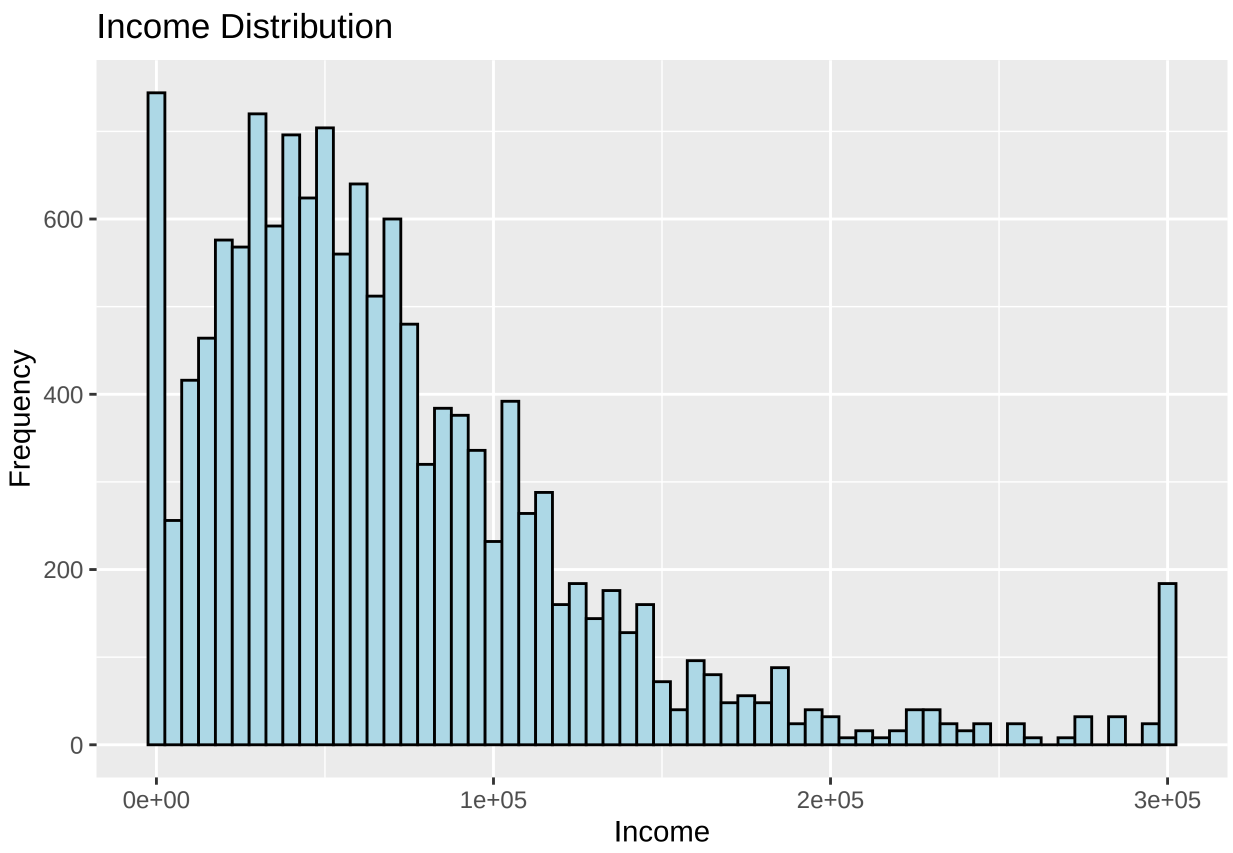
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##       0   30178   57500   70508   93750  300000     904
```

```r
sd(data1$B1STINC1, na.rm = TRUE)
```

```
## [1] 57837.37
```

```r
ggplot(data1, aes(x = B1STINC1)) +
  geom_histogram(binwidth = 5000, fill = "lightblue", color = "black") +
  labs(title = "Income Distribution", x = "Income", y = "Frequency")
```

```
## Warning: Removed 904 rows containing non-finite values (`stat_bin()`).
```



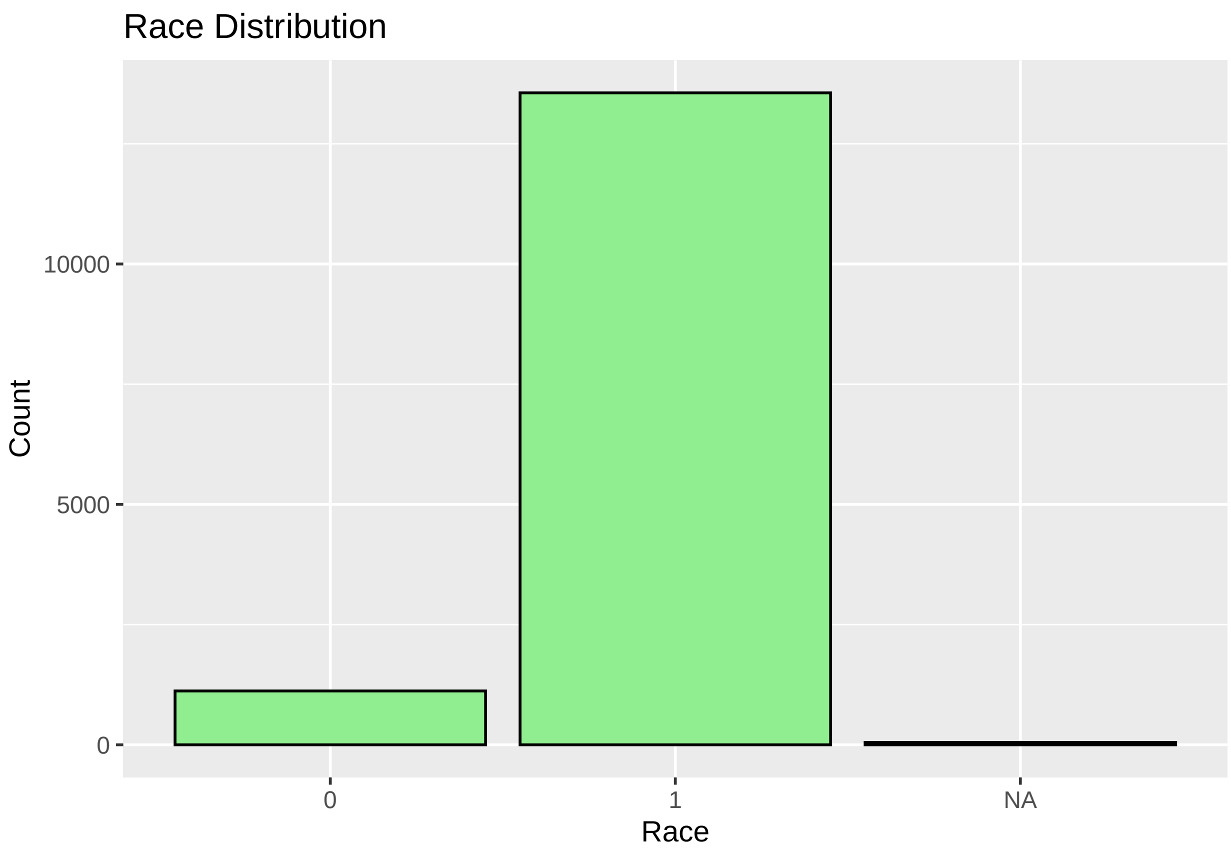
```r
# Race Distribution
cat("Race Distribution:\n")
```

```
## Race Distribution:
```

```r
summary(data1$race, na.rm = TRUE)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##  0.0000  1.0000  1.0000  0.9237  1.0000  1.0000      48
```

```r
ggplot(data1, aes(x = factor(race))) +
  geom_bar(fill = "lightgreen", color = "black") +
  labs(title = "Race Distribution", x = "Race", y = "Count")
```



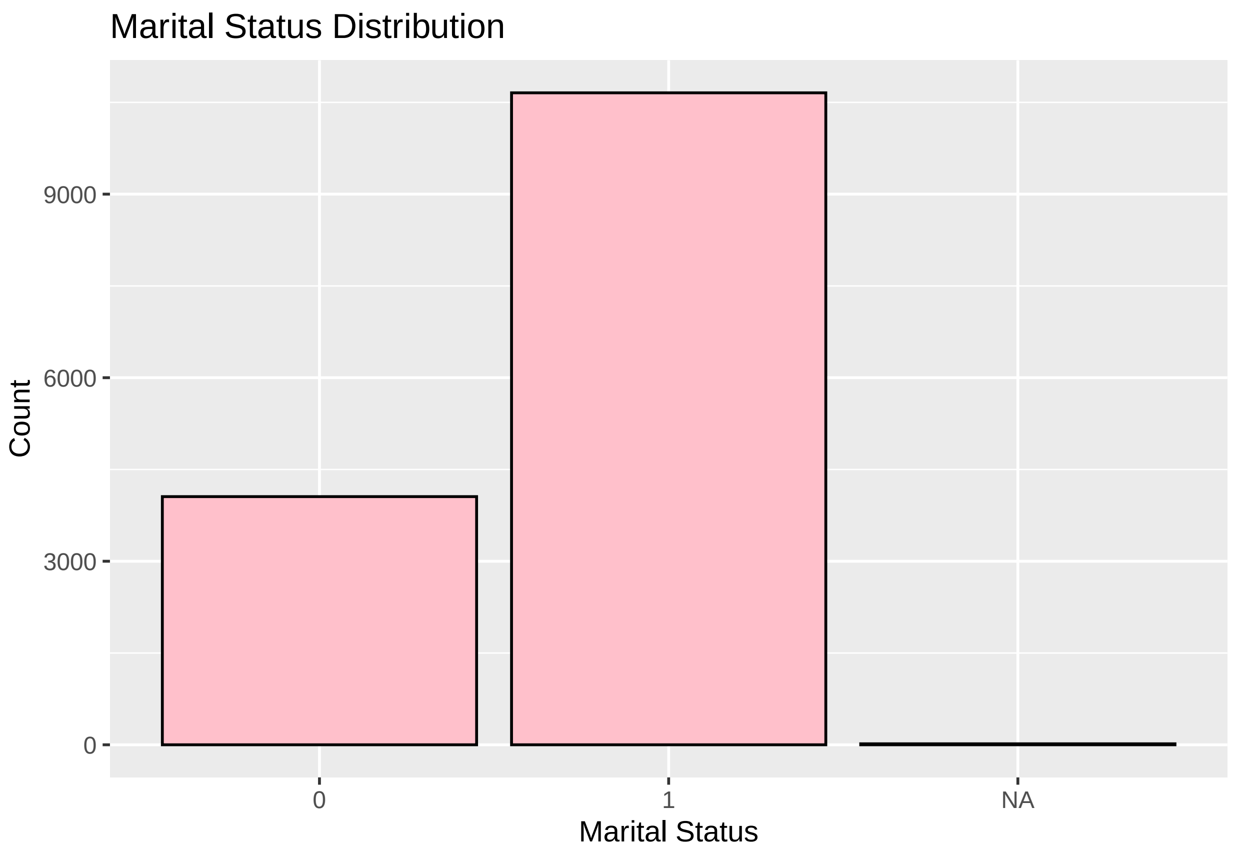
```
# Marital Status
cat("Marital Status Distribution:\n")
```

```
## Marital Status Distribution:
```

```
summary(data1$marital, na.rm = TRUE)
```

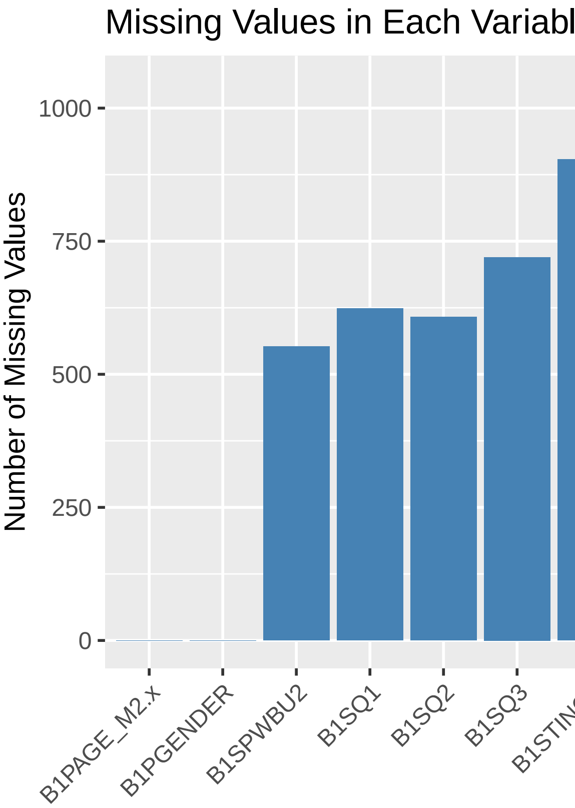
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##  0.0000  0.0000  1.0000  0.7243  1.0000  1.0000      16
```

```
ggplot(data1, aes(x = factor(marital))) +
  geom_bar(fill = "pink", color = "black") +
  labs(title = "Marital Status Distribution", x = "Marital Status", y = "Count")
```



Missing value Handling

```
# Create a data frame with the number of missing values for each variable
na_counts <- data1 %>% summarise_all(~sum(is.na(.))) %>% gather(key = "Variable", value = "NA_Count")

# Draw a bar chart of missing values
ggplot(na_counts, aes(x = Variable, y = NA_Count)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(title = "Missing Values in Each Variable", x = "Variable", y = "Number of Missing Values")
```



First, visually display the missing values, and then process them.

```
# Function to calculate the mode
get_mode <- function(v) {
  uniqv <- unique(na.omit(v))
  uniqv[which.max(tabulate(match(v, uniqv)))]
}

# Missing value handling

# Use median to fill missing values for continuous variables
data1$B1SPWBU2[is.na(data1$B1SPWBU2)] <- median(data1$B1SPWBU2, na.rm = TRUE)
data1$B1STINC1[is.na(data1$B1STINC1)] <- median(data1$B1STINC1, na.rm = TRUE)
data1$B2DNEGAV[is.na(data1$B2DNEGAV)] <- median(data1$B2DNEGAV, na.rm = TRUE)

# Use mode to fill missing values for categorical variables
data1$B2DN_STR[is.na(data1$B2DN_STR)] <- get_mode(data1$B2DN_STR)
data1$race[is.na(data1$race)] <- get_mode(data1$race)
data1$marital[is.na(data1$marital)] <- get_mode(data1$marital)

# For rating variables (assuming a 1-10 scale), use median or mode
data1$B1SQ2[is.na(data1$B1SQ2)] <- median(data1$B1SQ2, na.rm = TRUE)
data1$B1SQ1[is.na(data1$B1SQ1)] <- median(data1$B1SQ1, na.rm = TRUE)
data1$B1SQ3[is.na(data1$B1SQ3)] <- median(data1$B1SQ3, na.rm = TRUE)

# Print the updated data to check
sum(is.na(data1))
```

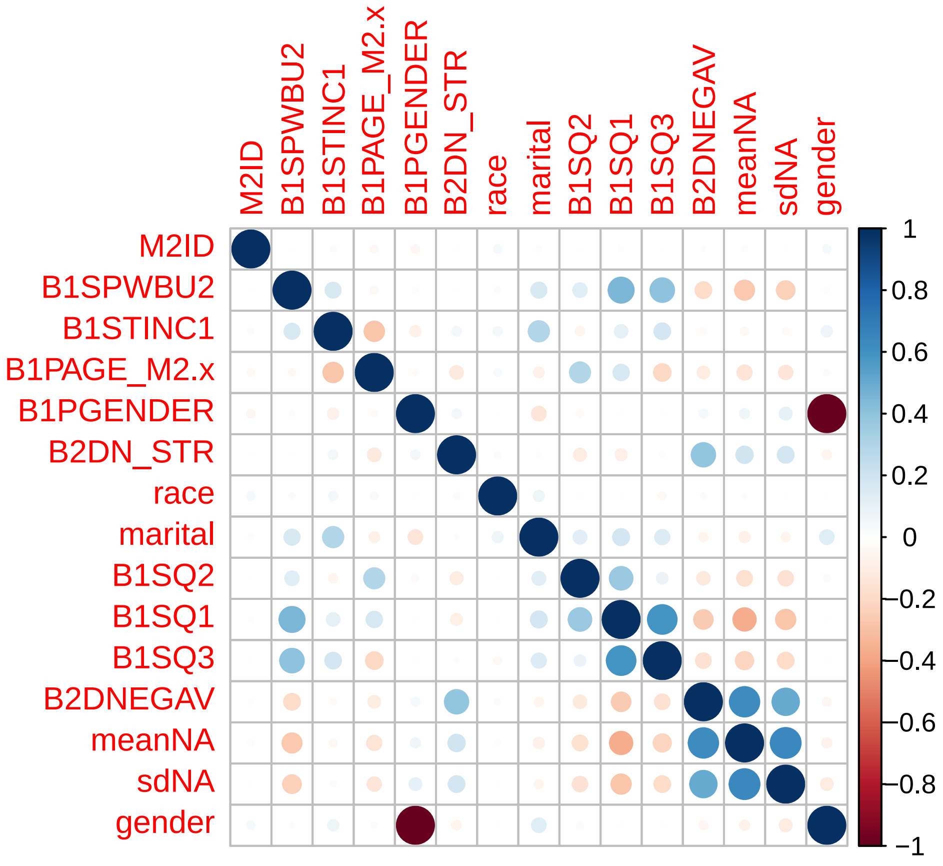
```
## [1] 0
```

More dimensions of data visualization

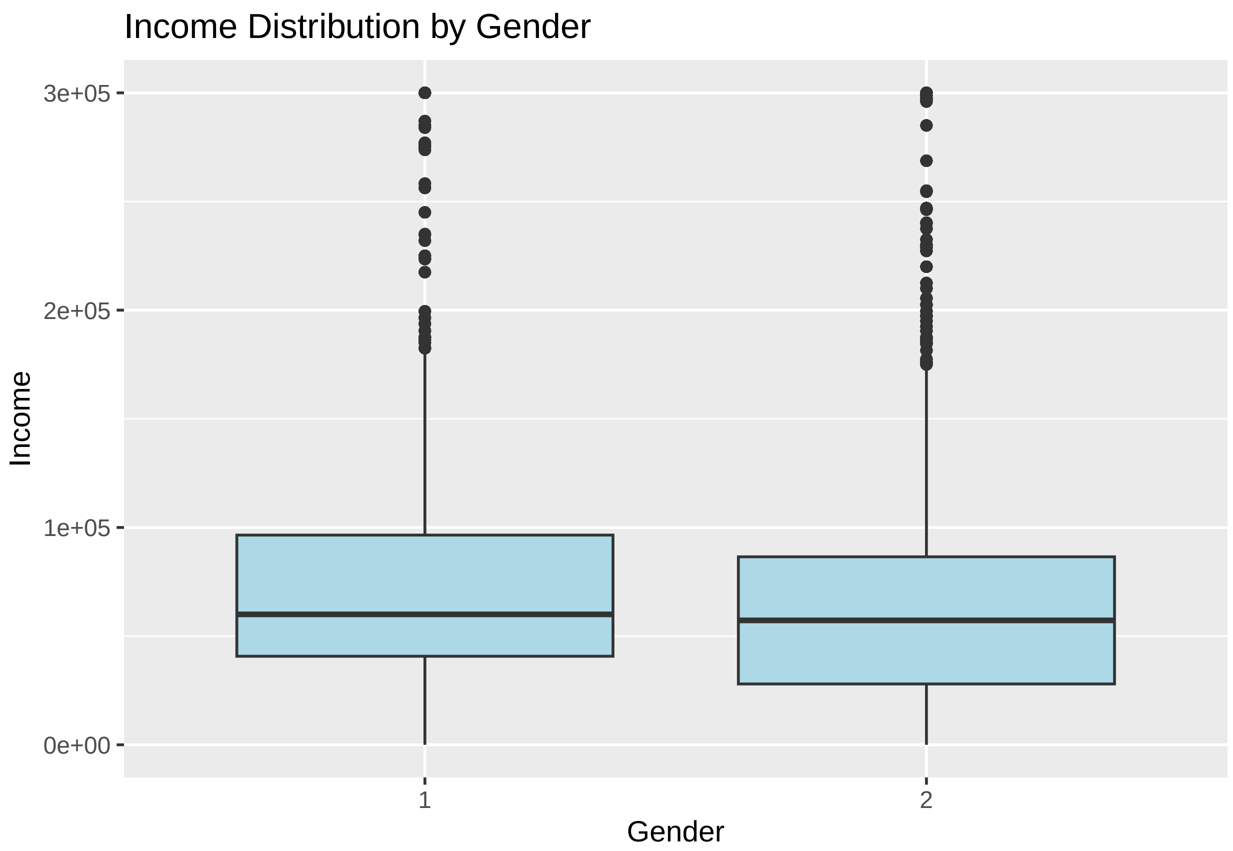
```
# correlation matrix - Correlation between variables
library(corrplot)
```

```
## corrplot 0.92 loaded
```

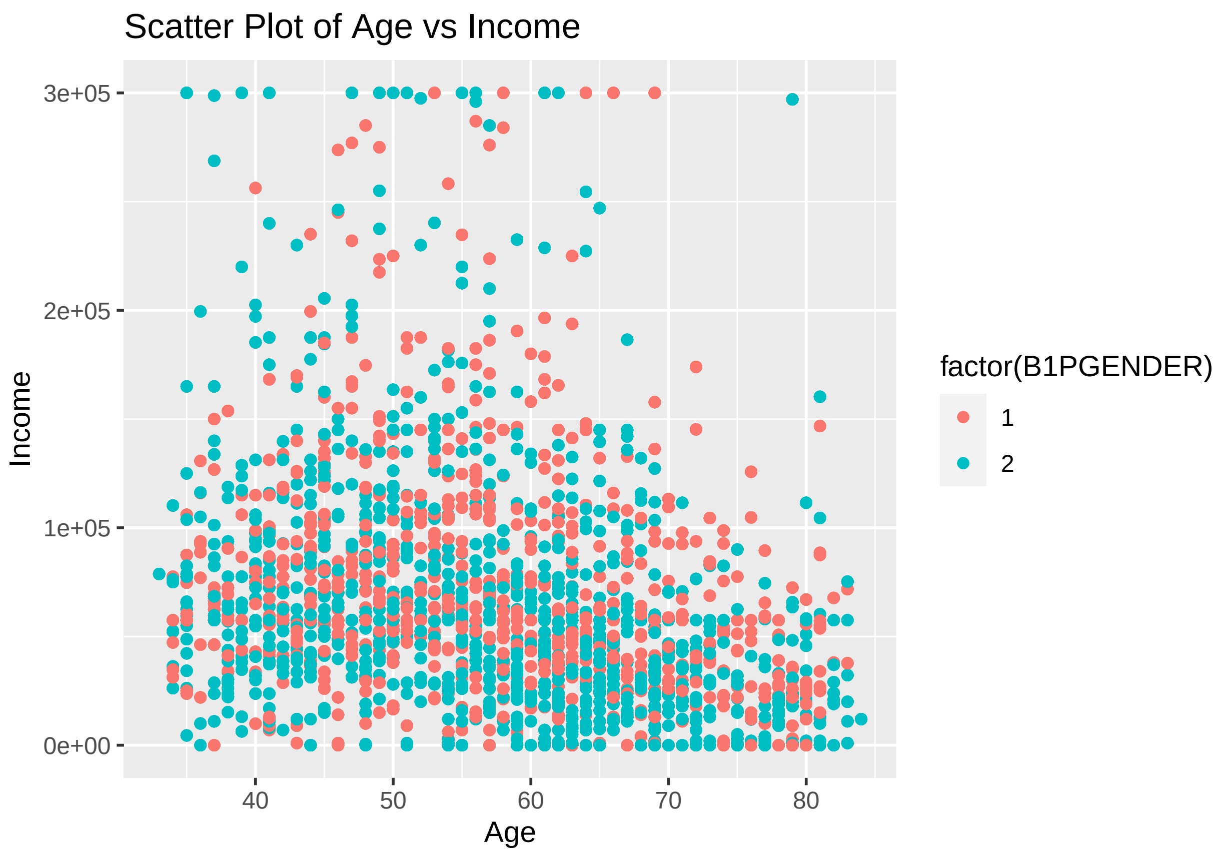
```
M <- cor(data1, use = "complete.obs")
corrplot(M, method = "circle")
```



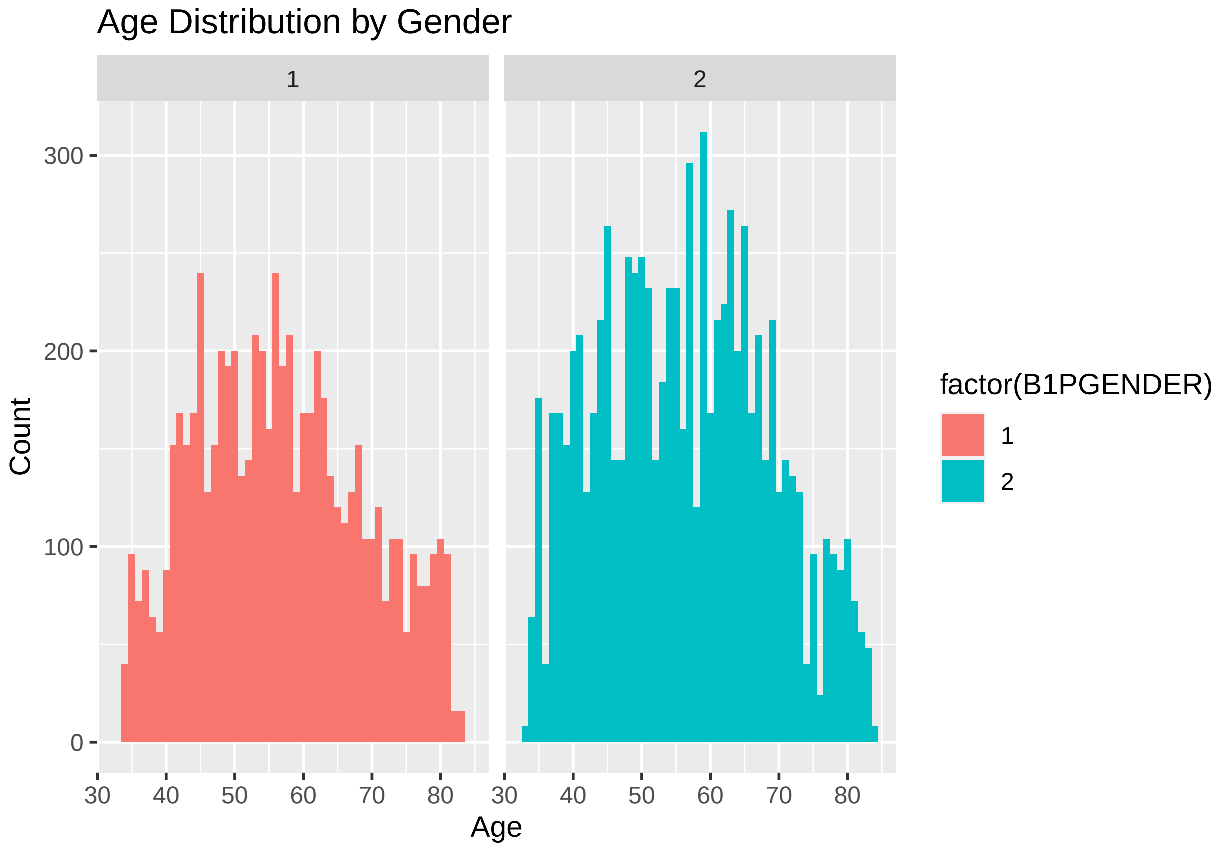
```
# Comparative analysis - Income distribution by sex
ggplot(data1, aes(x = factor(B1PGENDER), y = B1STINC1)) +
  geom_boxplot(fill = "lightblue") +
  labs(title = "Income Distribution by Gender", x = "Gender", y = "Income")
```



```
# Correlation analysis - Scatter plot of age and income
ggplot(data1, aes(x = B1PAGE_M2.x, y = B1STINC1)) +
  geom_point(aes(color = factor(B1PGENDER))) +
  labs(title = "Scatter Plot of Age vs Income", x = "Age", y = "Income")
```



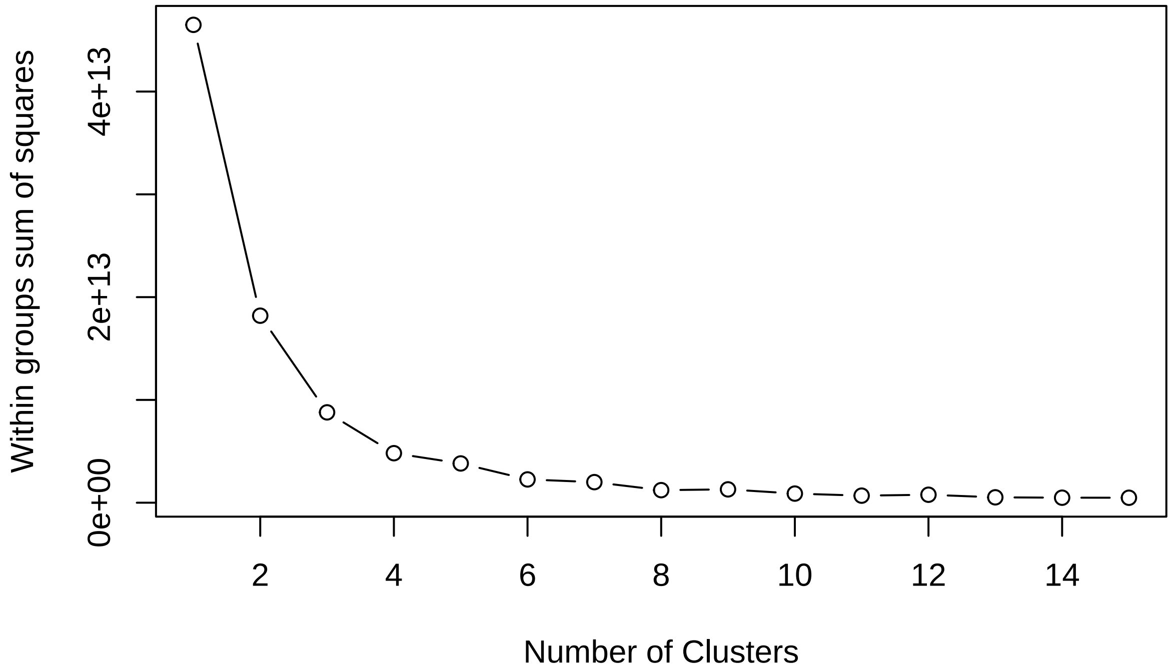
```
# multivariate analysis - gender difference of income
ggplot(data1, aes(x = B1PAGE_M2.x, fill = factor(B1PGENDER))) +
  geom_histogram(binwidth = 1) +
  facet_wrap(~ B1PGENDER) +
  labs(title = "Age Distribution by Gender", x = "Age", y = "Count")
```



```
# save data into original path
write.csv(data1, "cleaned_data.csv", row.names = FALSE)
```

Cluster Analysis

```
library(cluster)
life_satisfaction <- na.exclude(data1[, c("B1SQ2", "B1SQ1", "B1SQ3")])
d <- dist(life_satisfaction, method = "euclidean")
fit_hc <- hclust(d, method="ward.D2")
clusters <- cutree(fit_hc, k=3)
data1$cluster <- clusters

# Perform K-means clustering

# Determine the optimal number of clusters
set.seed(123)
wss <- (nrow(data1)-1)*sum(apply(data1,2,var))
for (i in 2:15) wss[i] <- sum(kmeans(data1, centers=i)$withinss)

# Plot elbow method
plot(1:15, wss, type="b", xlab="Number of Clusters", ylab="Within groups sum of squares")
```



```
# Perform K-means with an appropriate number of clusters
set.seed(123)
kmeans_result <- kmeans(data1, centers=3)
data1$cluster <- as.factor(kmeans_result$cluster)

# Analyze the cluster results
table(data1$cluster)
```

```
##
##    1    2    3
## 4560 9360  808
```

Multilevel Analysis

```
library(lme4)
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

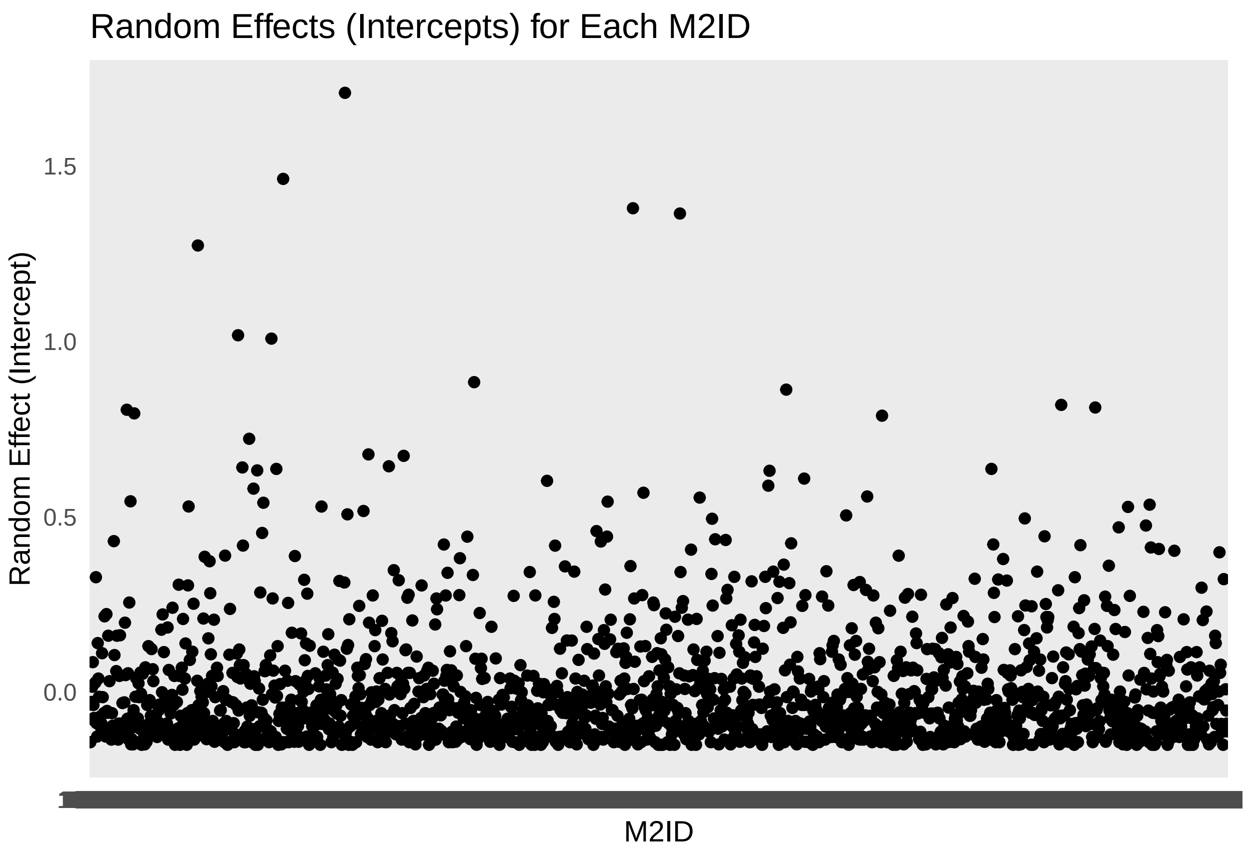
```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
library(ggplot2)

model_mcrm <- lmer(B2DNEGAV ~ cluster + (1|M2ID), data=data1)
```

```r
summary(model_mcrm)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: B2DNEGAV ~ cluster + (1 | M2ID)
##    Data: data1
##
## REML criterion at convergence: 1480.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -6.5149 -0.4007 -0.1748  0.1698 10.5657
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  M2ID     (Intercept) 0.03598  0.1897
##  Residual             0.05103  0.2259
## Number of obs: 14728, groups:  M2ID, 1841
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)  0.17564    0.00862  20.375
## cluster2     0.00143    0.01051   0.136
## cluster3    -0.02502    0.02222  -1.126
##
## Correlation of Fixed Effects:
##          (Intr) clstr2
## cluster2 -0.820
## cluster3 -0.388  0.318
```

```r
# Extracting random effects for M2ID
rand_eff <- ranef(model_mcrm)$M2ID
rand_eff_df <- as.data.frame(rand_eff)
rand_eff_df$M2ID <- rownames(rand_eff_df)

# Plotting random effects
ggplot(rand_eff_df, aes(x=M2ID, y=`(Intercept)`)) +
  geom_point() +
  theme_minimal() +
  labs(title="Random Effects (Intercepts) for Each M2ID",
       x="M2ID",
       y="Random Effect (Intercept)")
```

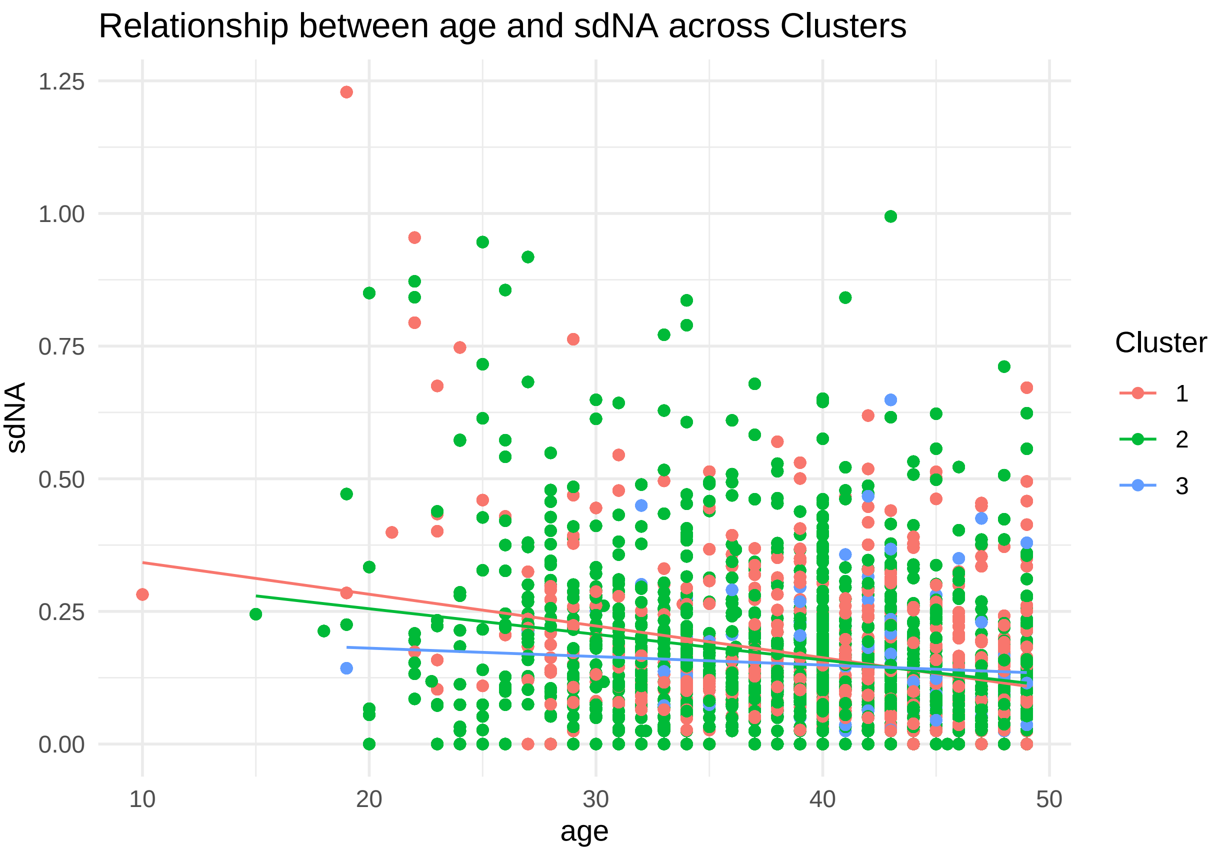


Regression Analysis 1

```r
model_ols <- lm(sdNA ~ cluster * B1SPWBU2, data=data1)
summary(model_ols)
```

```
## 
## Call:
## lm(formula = sdNA ~ cluster * B1SPWBU2, data = data1)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -0.25501 -0.09272 -0.03306  0.05492  0.94055 
## 
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)    
## (Intercept)        0.4018484  0.0130192  30.866  < 2e-16 ***
## cluster2          -0.0502626  0.0155249  -3.238  0.00121 ** 
## cluster3          -0.1894491  0.0373259  -5.076 3.91e-07 ***
## B1SPWBU2          -0.0059778  0.0003213 -18.603  < 2e-16 ***
## cluster2:B1SPWBU2  0.0011490  0.0003882   2.959  0.00309 ** 
## cluster3:B1SPWBU2  0.0043922  0.0008990   4.886 1.04e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.1422 on 14722 degrees of freedom
## Multiple R-squared:  0.05498,    Adjusted R-squared:  0.05466 
## F-statistic: 171.3 on 5 and 14722 DF,  p-value: < 2.2e-16
```

```
# Create a new data frame for plotting
plot_data <- data1
plot_data$predicted_sdNA <- predict(model_ols, newdata = data1)

# Plotting
ggplot(plot_data, aes(x=B1SPWBU2, y=sdNA, color=factor(cluster))) +
  geom_point() +  # Actual data points
  geom_line(aes(y=predicted_sdNA)) +  # Regression lines
  theme_minimal() +
  labs(title="Relationship between age and sdNA across Clusters",
       x="age",
       y="sdNA",
       color="Cluster")
```

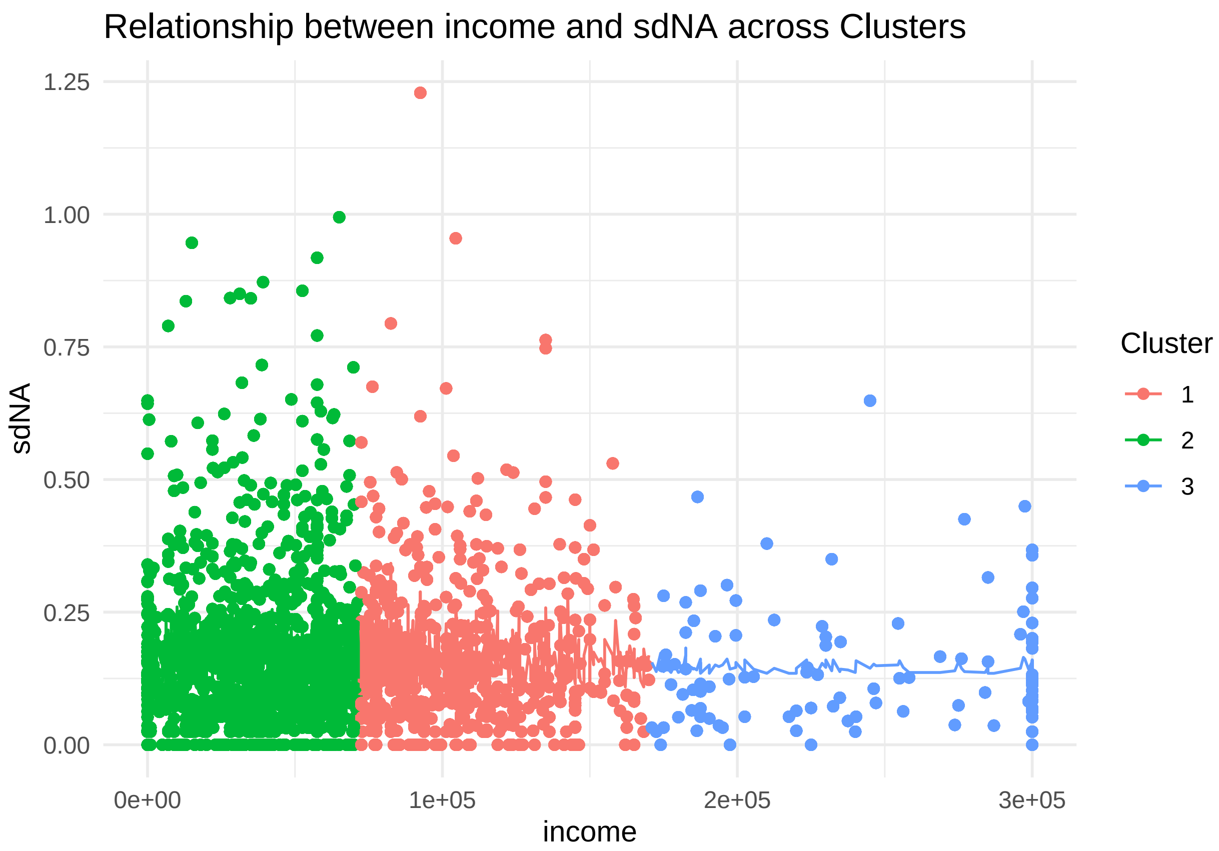


Regression Analysis 2

```
model_ols2 <- lm(sdNA ~ cluster * B1STINC1, data=data1)
summary(model_ols)
```

```
##
## Call:
## lm(formula = sdNA ~ cluster * B1SPWBU2, data = data1)
##
## Residuals:
##       Min       1Q   Median       3Q      Max
## -0.25501 -0.09272 -0.03306  0.05492  0.94055
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.4018484  0.0130192  30.866  < 2e-16 ***
## cluster2          -0.0502626  0.0155249  -3.238  0.00121 **
## cluster3          -0.1894491  0.0373259  -5.076 3.91e-07 ***
## B1SPWBU2          -0.0059778  0.0003213 -18.603  < 2e-16 ***
## cluster2:B1SPWBU2  0.0011490  0.0003882   2.959  0.00309 **
## cluster3:B1SPWBU2  0.0043922  0.0008990   4.886 1.04e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1422 on 14722 degrees of freedom
## Multiple R-squared:  0.05498,    Adjusted R-squared:  0.05466
## F-statistic: 171.3 on 5 and 14722 DF,  p-value: < 2.2e-16
```

```
# Create a new data frame for plotting
plot_data <- data1
plot_data$predicted_sdNA <- predict(model_ols, newdata = data1)

# Plotting
ggplot(plot_data, aes(x=B1STINC1, y=sdNA, color=factor(cluster))) +
  geom_point() +  # Actual data points
  geom_line(aes(y=predicted_sdNA)) +  # Regression lines
  theme_minimal() +
  labs(title="Relationship between income and sdNA across Clusters",
       x="income",
       y="sdNA",
       color="Cluster")
```

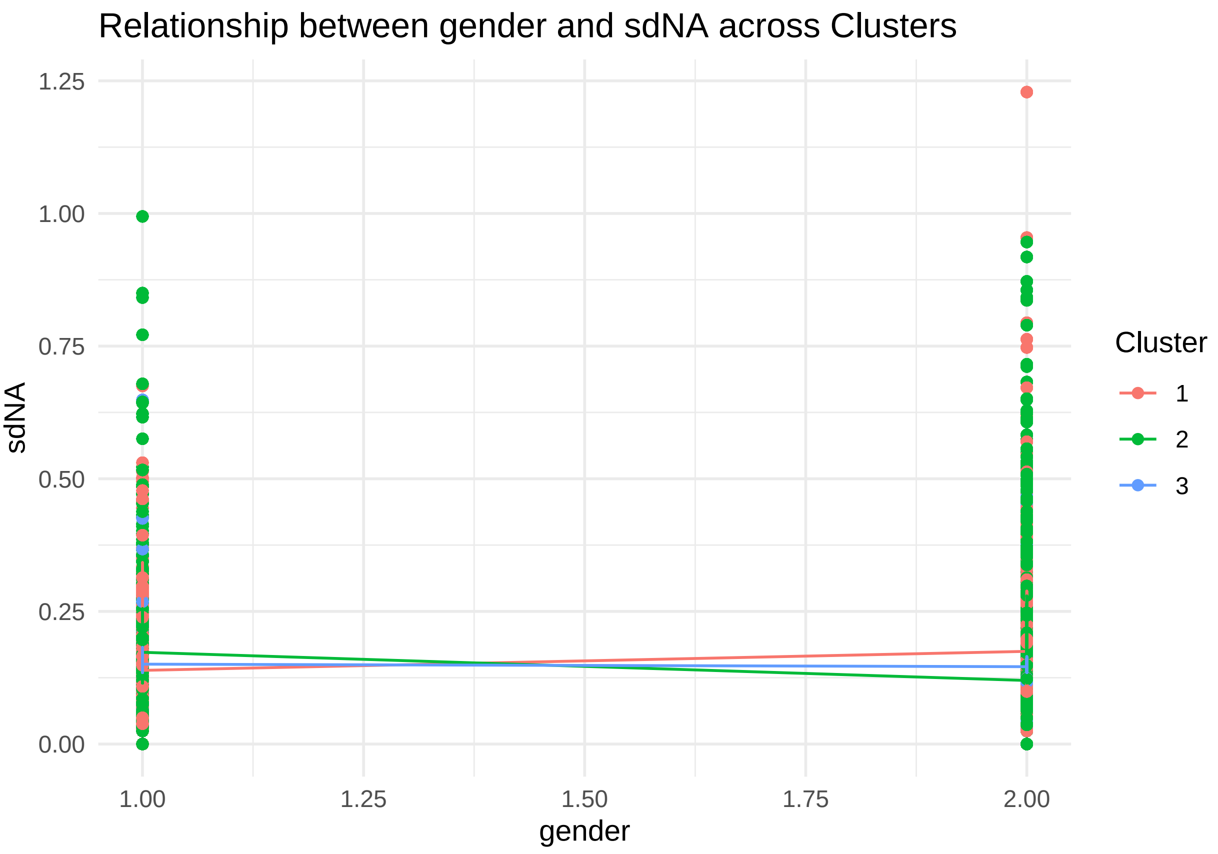


Regression Analysis 3

```
model_ols3 <- lm(sdNA ~ cluster * B1PGENDER, data=data1)
summary(model_ols)
```

```
##
## Call:
## lm(formula = sdNA ~ cluster * B1SPWBU2, data = data1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.25501 -0.09272 -0.03306  0.05492  0.94055
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.4018484  0.0130192  30.866  < 2e-16 ***
## cluster2          -0.0502626  0.0155249  -3.238  0.00121 **
## cluster3          -0.1894491  0.0373259  -5.076 3.91e-07 ***
## B1SPWBU2          -0.0059778  0.0003213 -18.603  < 2e-16 ***
## cluster2:B1SPWBU2  0.0011490  0.0003882   2.959  0.00309 **
## cluster3:B1SPWBU2  0.0043922  0.0008990   4.886 1.04e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1422 on 14722 degrees of freedom
## Multiple R-squared:  0.05498,    Adjusted R-squared:  0.05466
## F-statistic: 171.3 on 5 and 14722 DF,  p-value: < 2.2e-16
```

```r
# Create a new data frame for plotting
plot_data <- data1
plot_data$predicted_sdNA <- predict(model_ols, newdata = data1)

# Plotting
ggplot(plot_data, aes(x=B1PGENDER, y=sdNA, color=factor(cluster))) +
  geom_point() +  # Actual data points
  geom_line(aes(y=predicted_sdNA)) +  # Regression lines
  theme_minimal() +
  labs(title="Relationship between gender and sdNA across Clusters",
       x="gender",
       y="sdNA",
       color="Cluster")
```


Relationship between gender and sdNA across Clusters
sdNA
gender
Cluster