AMERICAN INTERNATIONAL UNIVERSITY-BANGLADESH

Project Title: Midterm Project	
Project No: 01	Date of Submission: 14/12/2024
Course Title: INTRODUCTION TO DATA SCIENCE	
Course Code: 00489	Section: C
Semester: Fall	Course Teacher: TOHEDUL ISLAM

No	Name	ID	Program	Signature
1	**Sowad Hossain Nirob**	21-44759-1	BSc CSE	

Faculty use only		
FACULTYCOMMENTS	**Marks Obtained**	
	Total Marks	

Introduction:

In this project, we focused on preprocessing and analyzing a dataset containing information about individuals and their loan applications. The primary objective was to clean, transform, and explore the data to derive meaningful insights and prepare it for downstream machine learning or statistical analysis tasks.
The dataset consists of demographic, financial, and loan-related features, such as age, gender, income, education, homeownership status, credit score, loan amount, and loan status. Each row represents an individual and their respective loan application details.
The key goals of this project included:

1. Cleaning the data by handling missing values and outliers.
2. Filtering the data to focus on meaningful ranges of income and age.
3. Imputing missing values for both numeric and categorical columns.
4. Removing duplicates to ensure data integrity.
5. Normalizing income data to enable fair comparisons.
6. Generating visualizations to explore key patterns and insights.
7. Preparing a clean dataset for further modeling and analysis.

In the subsequent sections, we discuss the preprocessing steps, data cleaning methodologies, exploratory data analysis (EDA), and insights derived from the dataset. The cleaned dataset was also exported for future use in machine learning or predictive modeling tasks.

Data:

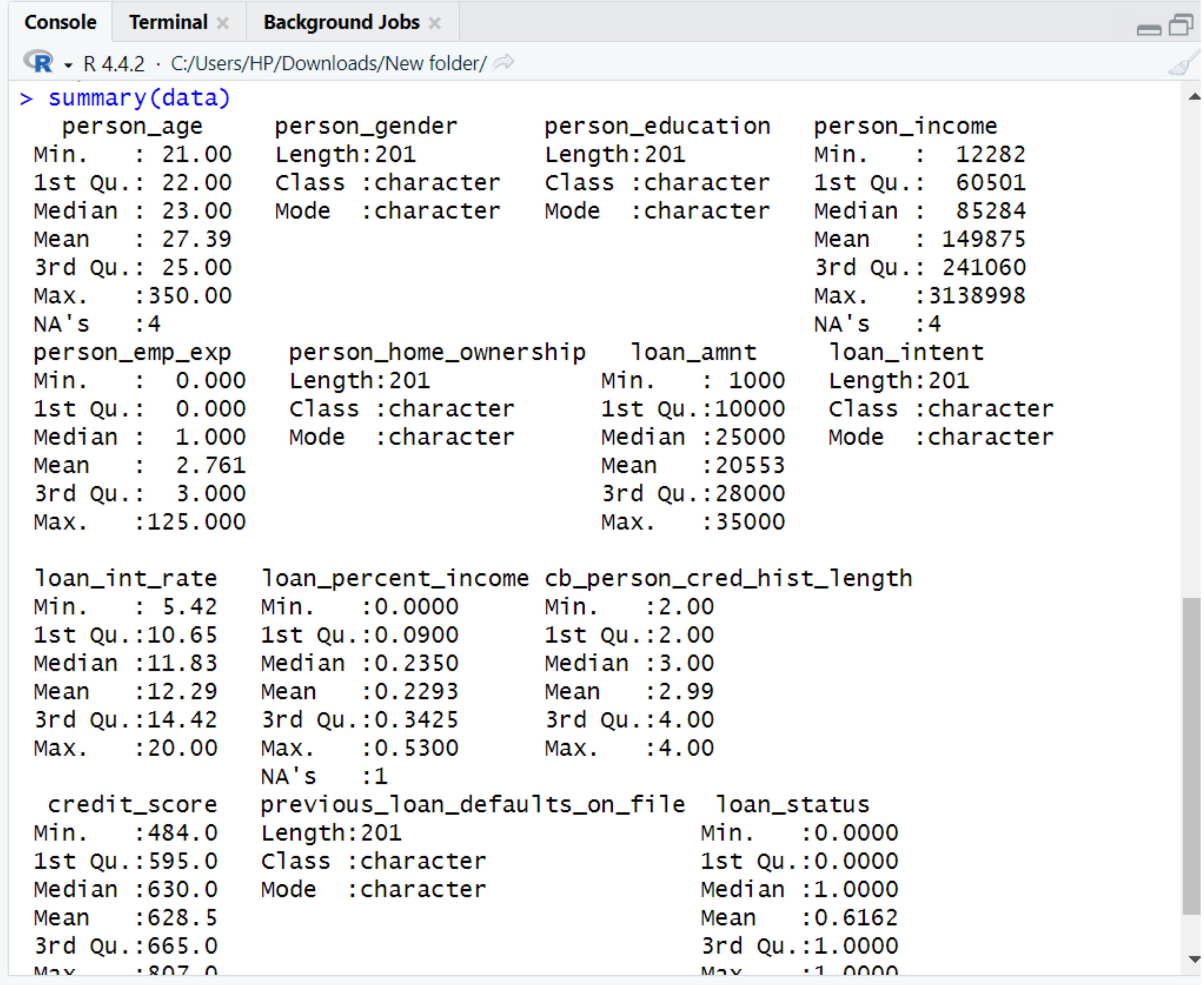
Library use:

```
library(readxl)
library(dplyr)
library(ggplot2)
library(naniar)
```

Load The Data:

```
data <- read_excel("Midterm_Dataset_Section(C).xlsx")
print("Data loaded:")
print(head(data))
```

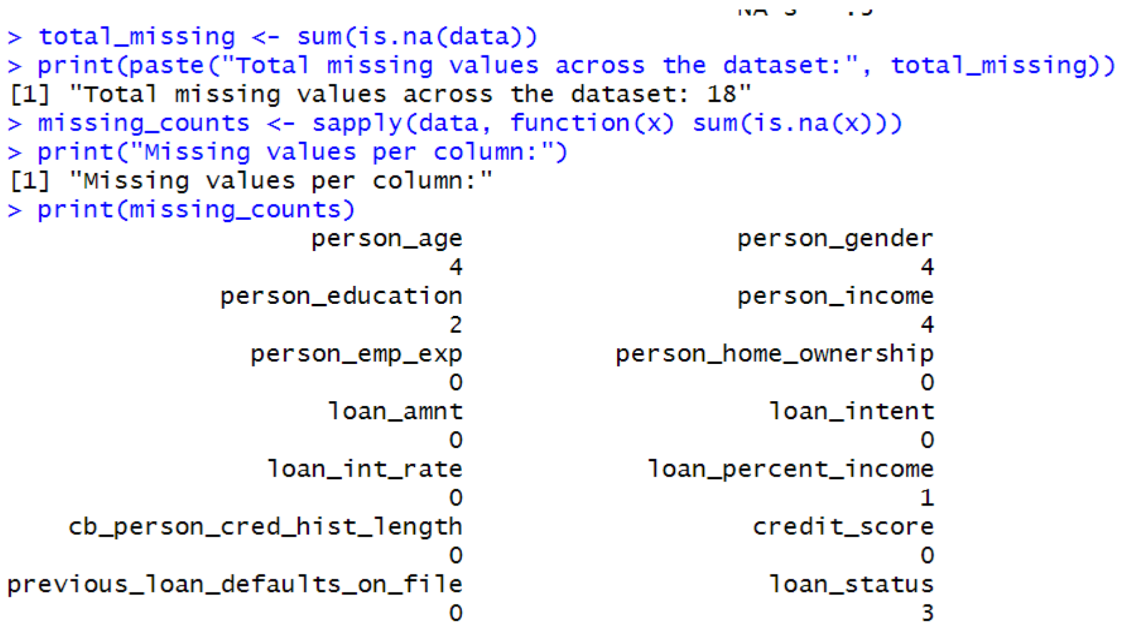
Data Set Summary:



```
> summary(data)
   person_age     person_gender      person_education   person_income    
 Min.   : 21.00   Length:201         Length:201         Min.   :  12282  
 1st Qu.: 22.00   Class :character   Class :character   1st Qu.:  60501  
 Median : 23.00   Mode  :character   Mode  :character   Median :  85284  
 Mean   : 27.39                                         Mean   : 149875  
 3rd Qu.: 25.00                                         3rd Qu.: 241060  
 Max.   :350.00                                         Max.   :3138998  
 NA's   :4                                              NA's   :4        
 person_emp_exp    person_home_ownership   loan_amnt      loan_intent       
 Min.   :  0.000   Length:201            Min.   : 1000   Length:201        
 1st Qu.:  0.000   Class :character      1st Qu.:10000   Class :character  
 Median :  1.000   Mode  :character      Median :25000   Mode  :character  
 Mean   :  2.761                         Mean   :20553                     
 3rd Qu.:  3.000                         3rd Qu.:28000                     
 Max.   :125.000                         Max.   :35000                     

 loan_int_rate   loan_percent_income cb_person_cred_hist_length
 Min.   : 5.42   Min.   :0.0000      Min.   :2.00              
 1st Qu.:10.65   1st Qu.:0.0900      1st Qu.:2.00              
 Median :11.83   Median :0.2350      Median :3.00              
 Mean   :12.29   Mean   :0.2293      Mean   :2.99              
 3rd Qu.:14.42   3rd Qu.:0.3425      3rd Qu.:4.00              
 Max.   :20.00   Max.   :0.5300      Max.   :4.00              
                 NA's   :1                                     
  credit_score   previous_loan_defaults_on_file  loan_status    
 Min.   :484.0   Length:201                     Min.   :0.0000  
 1st Qu.:595.0   Class :character               1st Qu.:0.0000  
 Median :630.0   Mode  :character               Median :1.0000  
 Mean   :628.5                                  Mean   :0.6162  
 3rd Qu.:665.0                                  3rd Qu.:1.0000  
 Max.   :807.0                                  Max.   :1.0000  
```

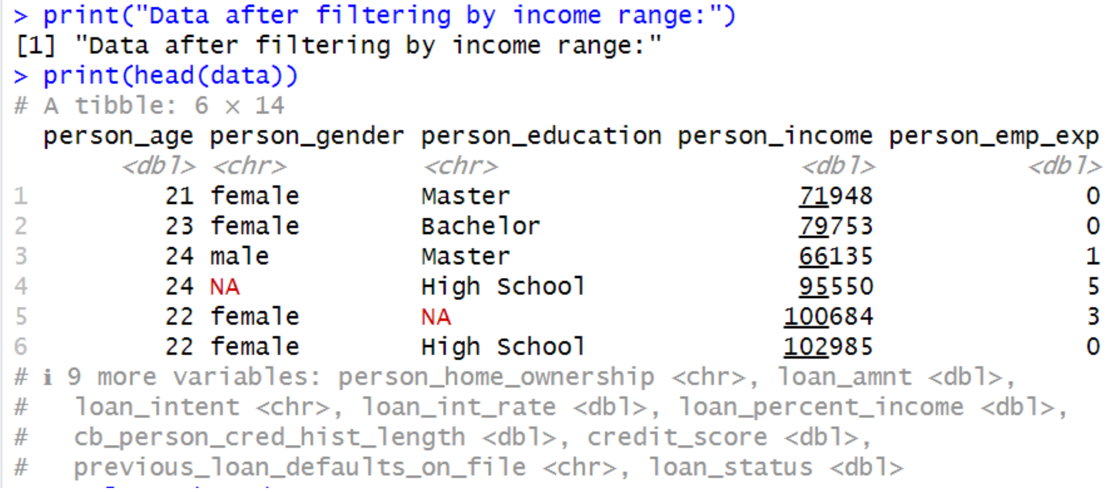
Data Preparation & Exploration:

```
total_missing <- sum(is.na(data))
print(paste("Total missing values across the dataset:", total_missing))

missing_counts <- sapply(data, function(x) sum(is.na(x)))
print("Missing values per column:")
print(missing_counts)
```



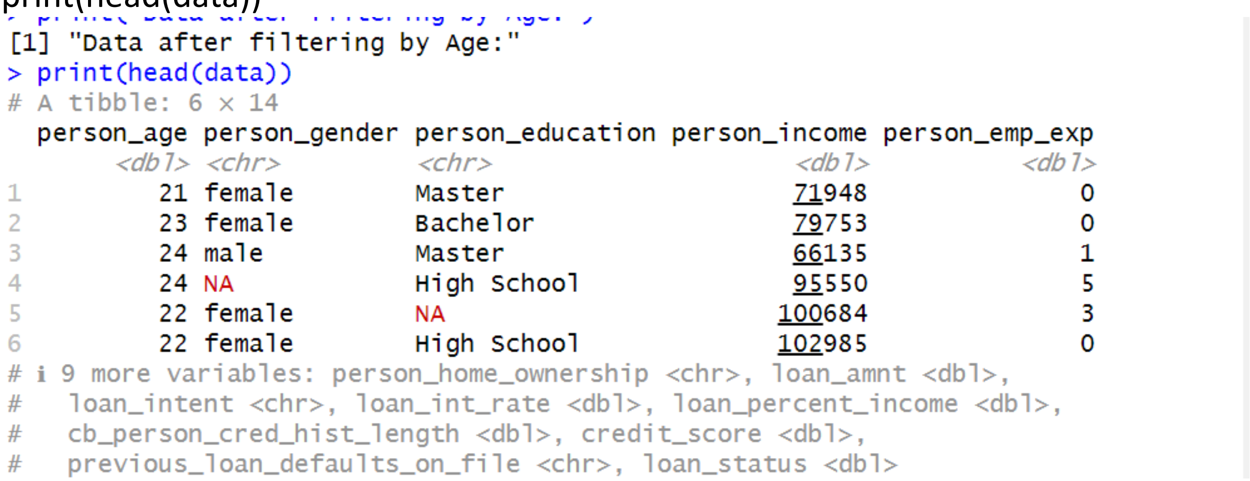
```
> total_missing <- sum(is.na(data))
> print(paste("Total missing values across the dataset:", total_missing))
[1] "Total missing values across the dataset: 18"
> missing_counts <- sapply(data, function(x) sum(is.na(x)))
> print("Missing values per column:")
[1] "Missing values per column:"
> print(missing_counts)
                    person_age                  person_gender 
                             4                              4 
              person_education                  person_income 
                             2                              4 
                person_emp_exp          person_home_ownership 
                             0                              0 
                     loan_amnt                    loan_intent 
                             0                              0 
                 loan_int_rate            loan_percent_income 
                             0                              1 
    cb_person_cred_hist_length                   credit_score 
                             0                              0 
previous_loan_defaults_on_file                    loan_status 
                             0                              3 
```

```
get_mode <- function(v) {
  uniqv <- unique(na.omit(v))
  uniqv[which.max(tabulate(match(v, uniqv)))]
}

income_lower_bound <- 50000
income_upper_bound <- 500000
data <- data %>%
  filter(person_income >= income_lower_bound & person_income <=
income_upper_bound)
print("Data after filtering by income range:")
print(head(data))
```



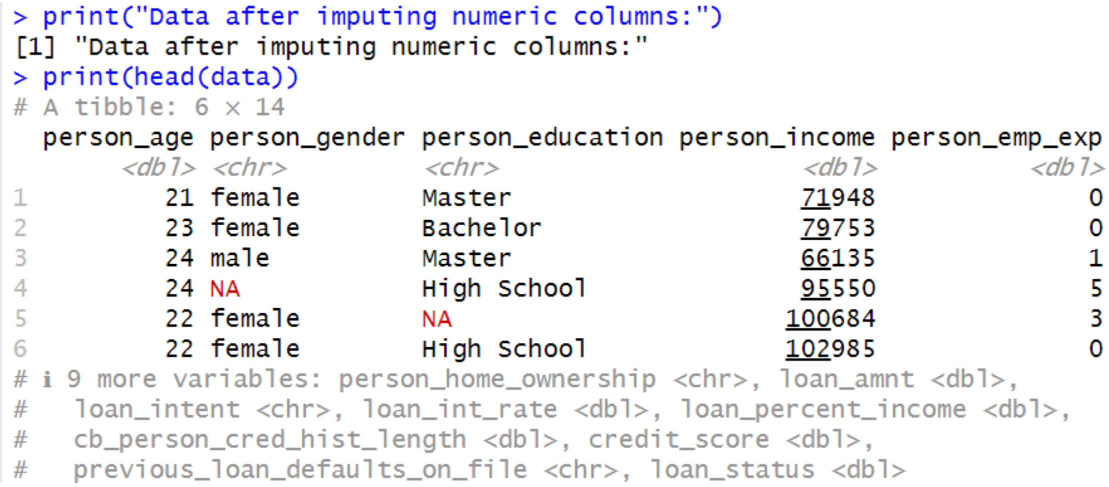
```
> print("Data after filtering by income range:")
[1] "Data after filtering by income range:"
> print(head(data))
# A tibble: 6 × 14
  person_age person_gender person_education person_income person_emp_exp
       <dbl> <chr>         <chr>                    <dbl>          <dbl>
1         21 female        Master                   71948              0
2         23 female        Bachelor                 79753              0
3         24 male          Master                   66135              1
4         24 NA            High School              95550              5
5         22 female        NA                      100684              3
6         22 female        High School             102985              0
# ℹ 9 more variables: person_home_ownership <chr>, loan_amnt <dbl>,
#   loan_intent <chr>, loan_int_rate <dbl>, loan_percent_income <dbl>,
#   cb_person_cred_hist_length <dbl>, credit_score <dbl>,
#   previous_loan_defaults_on_file <chr>, loan_status <dbl>
```

```
age_lower_bound <- 20
age_upper_bound <- 80
data <- data %>%
  filter(person_age >= age_lower_bound & person_age <= age_upper_bound)
print("Data after filtering by Age:")
print(head(data))
```



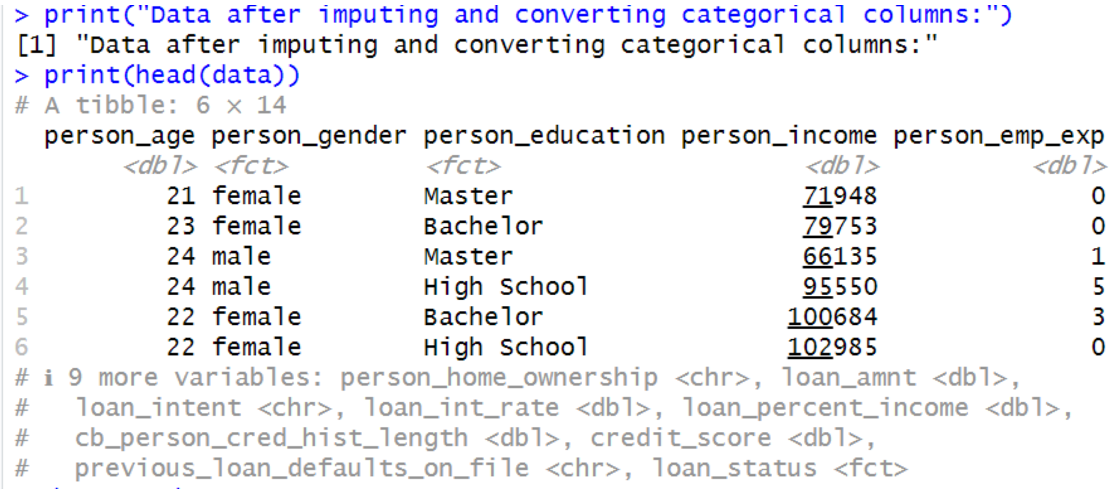
```
> print("Data after filtering by Age:")
[1] "Data after filtering by Age:"
> print(head(data))
# A tibble: 6 × 14
  person_age person_gender person_education person_income person_emp_exp
       <dbl> <chr>         <chr>                    <dbl>          <dbl>
1         21 female        Master                   71948              0
2         23 female        Bachelor                 79753              0
3         24 male          Master                   66135              1
4         24 NA            High School              95550              5
5         22 female        NA                      100684              3
6         22 female        High School             102985              0
# ℹ 9 more variables: person_home_ownership <chr>, loan_amnt <dbl>,
#   loan_intent <chr>, loan_int_rate <dbl>, loan_percent_income <dbl>,
#   cb_person_cred_hist_length <dbl>, credit_score <dbl>,
#   previous_loan_defaults_on_file <chr>, loan_status <dbl>
```

```
data$person_age[is.na(data$person_age)] <- mean(data$person_age, na.rm =
TRUE)
data$person_income[is.na(data$person_income)] <-
median(data$person_income, na.rm = TRUE)
print("Data after imputing numeric columns:")
print(head(data))
```



```
> print("Data after imputing numeric columns:")
[1] "Data after imputing numeric columns:"
> print(head(data))
# A tibble: 6 × 14
  person_age person_gender person_education person_income person_emp_exp
       <dbl> <chr>         <chr>                    <dbl>          <dbl>
1         21 female        Master                   71948              0
2         23 female        Bachelor                 79753              0
3         24 male          Master                   66135              1
4         24 NA            High School              95550              5
5         22 female        NA                      100684              3
6         22 female        High School             102985              0
# ℹ 9 more variables: person_home_ownership <chr>, loan_amnt <dbl>,
#   loan_intent <chr>, loan_int_rate <dbl>, loan_percent_income <dbl>,
#   cb_person_cred_hist_length <dbl>, credit_score <dbl>,
#   previous_loan_defaults_on_file <chr>, loan_status <dbl>
```

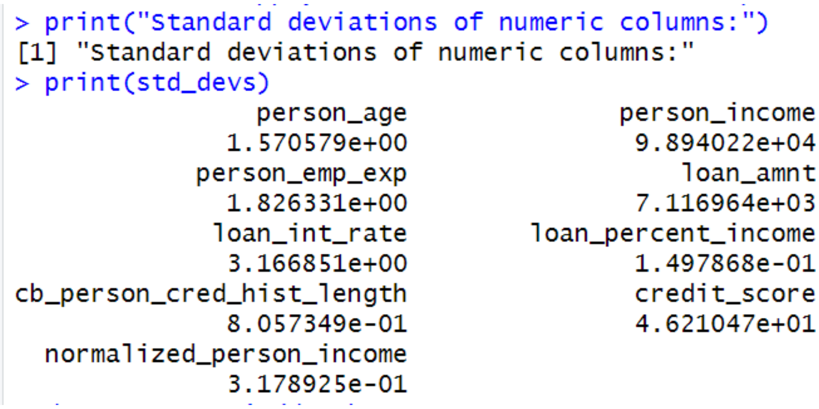
```
mode_education <- get_mode(data$person_education)
data$person_education[is.na(data$person_education)] <- mode_education
data$person_education <- as.factor(data$person_education)

mode_loan_status <- get_mode(data$loan_status)
data$loan_status[is.na(data$loan_status)] <- mode_loan_status
data$loan_status <- as.factor(data$loan_status)

mode_gender <- get_mode(data$person_gender)
data$person_gender[is.na(data$person_gender)] <- mode_gender
data$person_gender <- as.factor(data$person_gender)

print("Data after imputing and converting categorical columns:")
print(head(data))
```



```
> print("Data after imputing and converting categorical columns:")
[1] "Data after imputing and converting categorical columns:"
> print(head(data))
# A tibble: 6 × 14
  person_age person_gender person_education person_income person_emp_exp
       <dbl> <fct>         <fct>                    <dbl>          <dbl>
1         21 female        Master                   71948              0
2         23 female        Bachelor                 79753              0
3         24 male          Master                   66135              1
4         24 male          High School              95550              5
5         22 female        Bachelor                100684              3
6         22 female        High School             102985              0
# ℹ 9 more variables: person_home_ownership <chr>, loan_amnt <dbl>,
#   loan_intent <chr>, loan_int_rate <dbl>, loan_percent_income <dbl>,
#   cb_person_cred_hist_length <dbl>, credit_score <dbl>,
#   previous_loan_defaults_on_file <chr>, loan_status <fct>
```

```
data <- data %>%
  distinct()
print("Data after removing duplicates:")
print(head(data))

data$normalized_person_income <- (data$person_income -
min(data$person_income)) / (max(data$person_income) -
min(data$person_income))
print("Data after normalization of person_income:")
print(head(data))
```

```
summary_stats <- summary(data)
print("Summary statistics of dataset:")
print(summary_stats)

numeric_columns <- sapply(data, is.numeric)
std_devs <- sapply(data[, numeric_columns, drop = FALSE], sd, na.rm = TRUE)
print("Standard deviations of numeric columns:")
print(std_devs)
```



```
> print("Standard deviations of numeric columns:")
[1] "Standard deviations of numeric columns:"
> print(std_devs)
                person_age              person_income
              1.570579e+00               9.894022e+04
            person_emp_exp                  loan_amnt
              1.826331e+00               7.116964e+03
             loan_int_rate        loan_percent_income
              3.166851e+00               1.497868e-01
cb_person_cred_hist_length               credit_score
              8.057349e-01               4.621047e+01
  normalized_person_income
              3.178925e-01
```

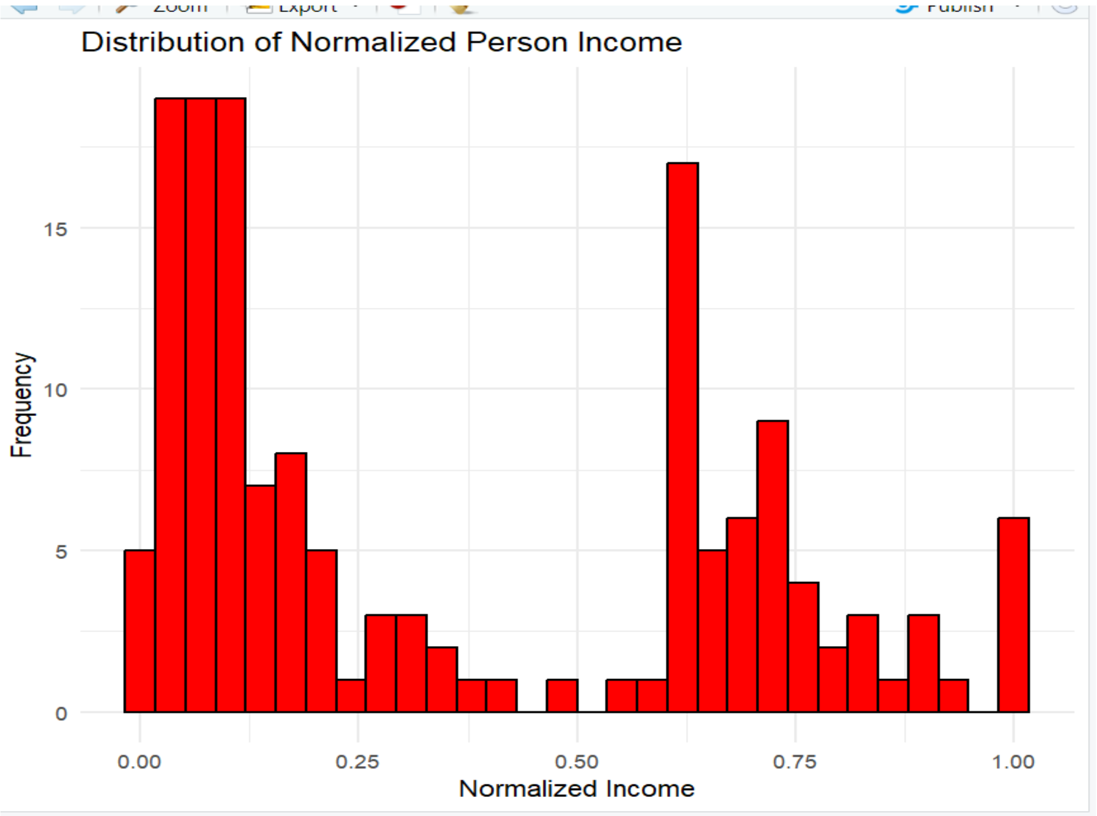
```
data <- na.omit(data)
print(head(data))
```

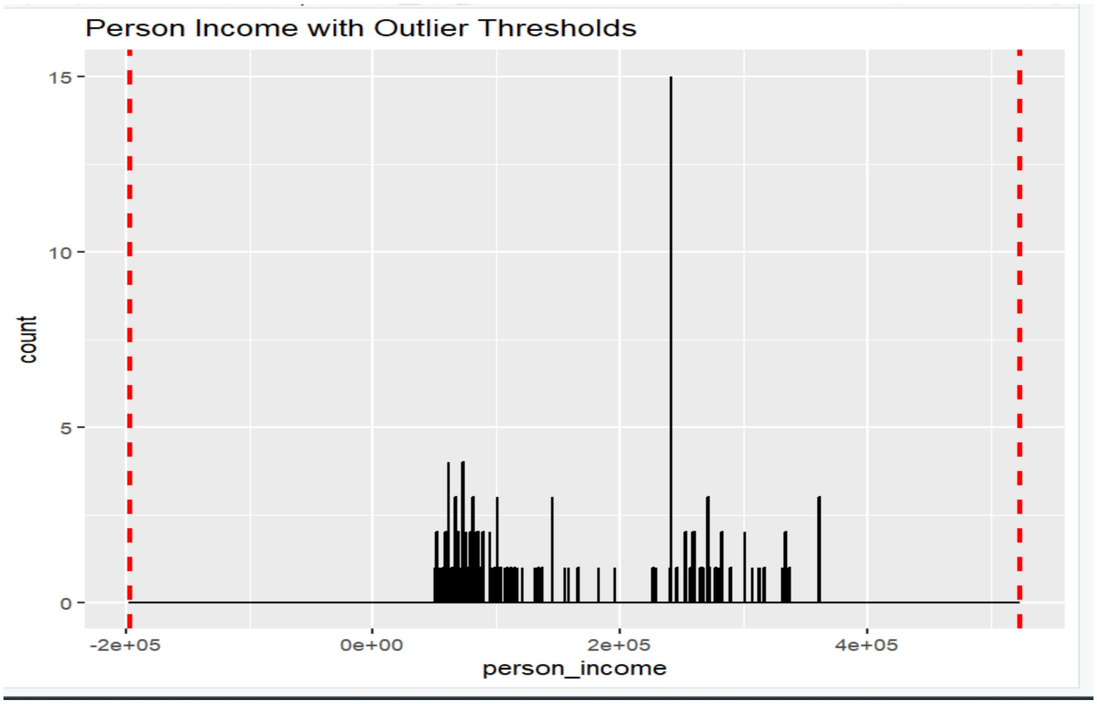
Histogram, Box plot and bar Plot:

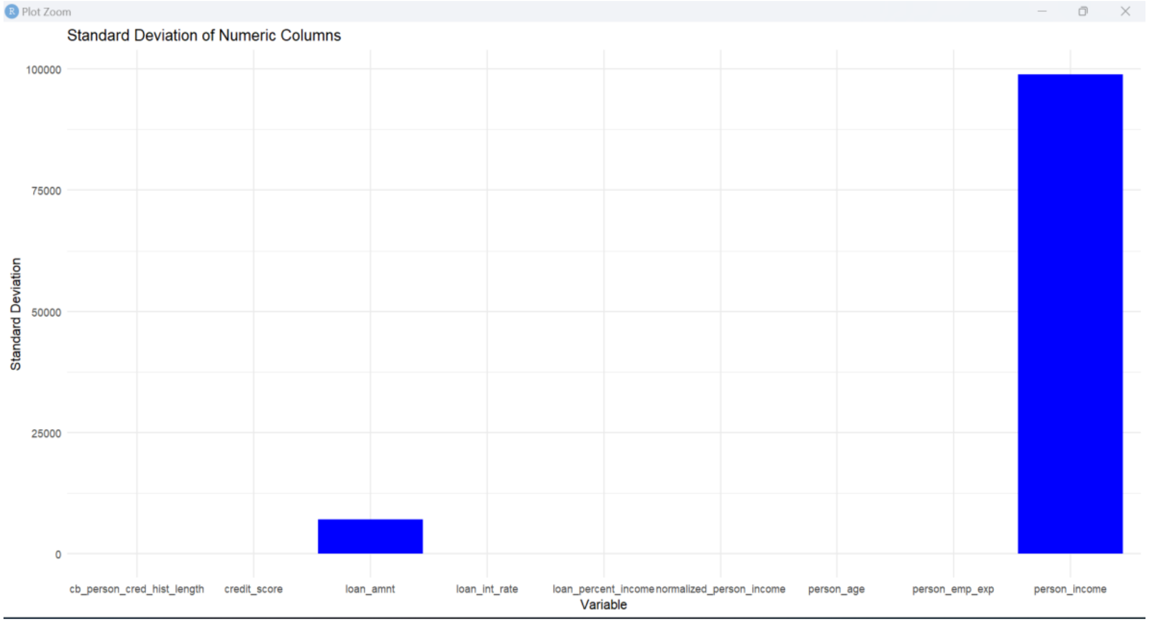
```
std_data <- data.frame(Variable = names(std_devs), StdDev = std_devs)
ggplot(std_data, aes(x = Variable, y = StdDev)) +
  geom_col(fill = "blue") +
  theme_minimal() +
  labs(title = "Standard Deviation of Numeric Columns", x = "Variable", y =
"Standard Deviation")

Q1 <- quantile(data$person_income, 0.25, na.rm = TRUE)
Q3 <- quantile(data$person_income, 0.75, na.rm = TRUE)
IQR <- Q3 - Q1
```

```
lower_bound <- Q1 - 1.5 * IQR
upper_bound <- Q3 + 1.5 * IQR
outliers <- data$person_income < lower_bound | data$person_income >
upper_bound

print("Visualizing outliers in person_income:")
ggplot(data, aes(x = person_income)) +
  geom_histogram(fill = "blue", color = "black", binwidth = 500) +
  geom_vline(xintercept = c(lower_bound, upper_bound), color = "red", linetype =
"dashed", size = 1) +
  labs(title = "Person Income with Outlier Thresholds")

data <- data[!outliers, ]
print("Data after removing outliers:")
print(head(data))

ggplot(data, aes(x = normalized_person_income)) +
  geom_histogram(bins = 30, fill = "red", color = "black") +
  theme_minimal() +
  labs(title = "Distribution of Normalized Person Income", x = "Normalized
Income", y = "Frequency")
```

Plots:

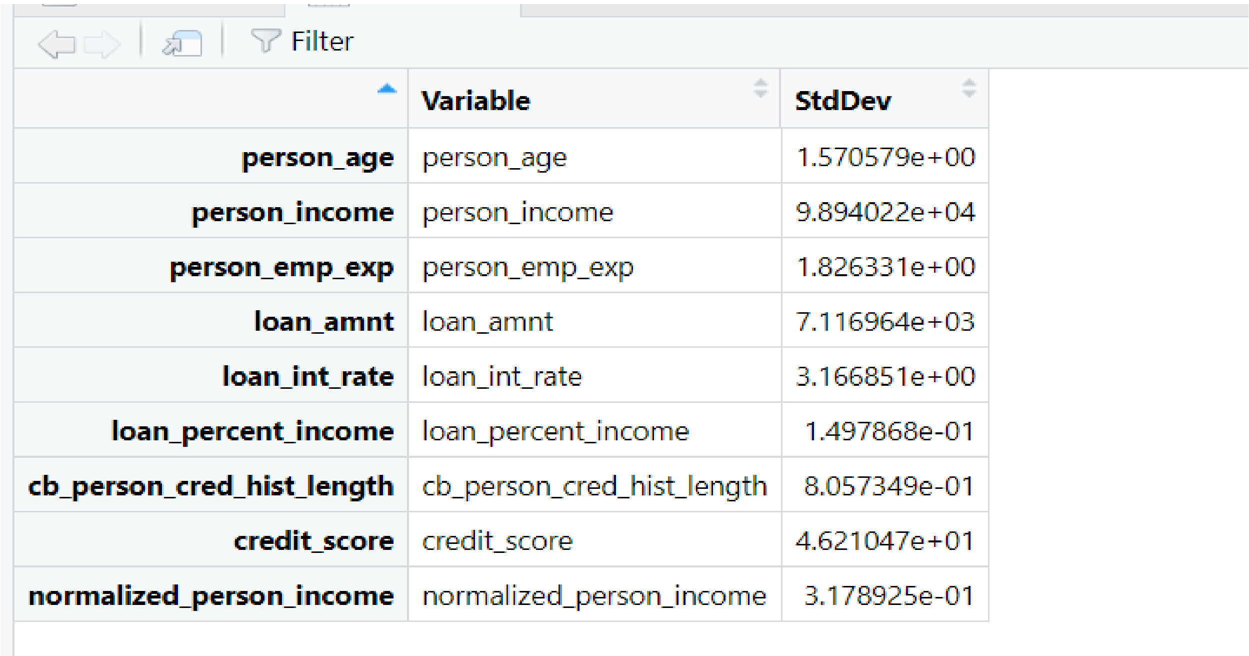

Distribution of Normalized Person Income


Person Income with Outlier Thresholds



Standard Deviation:

```
numeric_columns <- sapply(data, is.numeric)
std_devs <- sapply(data[, numeric_columns, drop = FALSE], sd, na.rm = TRUE)
print("Standard deviations of numeric columns:")
print(std_devs)
```



	Variable	StdDev
person_age	person_age	1.570579e+00
person_income	person_income	9.894022e+04
person_emp_exp	person_emp_exp	1.826331e+00
loan_amnt	loan_amnt	7.116964e+03
loan_int_rate	loan_int_rate	3.166851e+00
loan_percent_income	loan_percent_income	1.497868e-01
cb_person_cred_hist_length	cb_person_cred_hist_length	8.057349e-01
credit_score	credit_score	4.621047e+01
normalized_person_income	normalized_person_income	3.178925e-01