AMERICAN INTERNATIONAL UNIVERSITY-BANGLADESH

Faculty of Science and Technology

Assignment Title:	Midterm Project		
Assignment No:	01	Date of Submission:	14 December 2024
Course Title:	Introduction to Data Science		
Course Code:	CSC4180	Section:	C
Semester:	Fall 2024-25	Course Teacher:	TOHEDUL ISLAM

Declaration and Statement of Authorship:

1. I/we hold a copy of this Assignment/Case-Study, which can be produced if the original is lost/damaged.
2. This Assignment/Case-Study is my/our original work and no part of it has been copied from any other student's work or from any other source except where due acknowledgement is made.
3. No part of this Assignment/Case-Study has been written for me/us by any other person except where such collaboration has been authorized by the concerned teacher and is clearly acknowledged in the assignment.
4. I/we have not previously submitted or currently submitting this work for any other course/unit.
5. This work may be reproduced, communicated, compared and archived for the purpose of detecting plagiarism.
6. I/we give permission for a copy of my/our marked work to be retained by the faculty for review and comparison, including review by external examiners.
7. I/we understand that Plagiarism is the presentation of the work, idea or creation of another person as though it is your own. It is a formofcheatingandisaveryseriousacademicoffencethatmayleadtoexpulsionfromtheUniversity. Plagiarized material can be drawn from, and presented in, written, graphic and visual form, including electronic data, and oral presentations. Plagiarism occurs when the origin of them arterial used is not appropriately cited.
8. I/we also understand that enabling plagiarism is the act of assisting or allowing another person to plagiarize or to copy my/our work.

* *Student(s) must complete all details except the faculty use part.*
** Please submit all assignments to your course teacher or the office of the concerned teacher.

Group Name/No.: 14

No	Name	ID	Program	Signature
01	AZMINUR RAHMAN	22-46459-1	BSc [CSE]	
02	MD. ABDUL MALEK RONY	20-43687-2	BSc [CSE]	
03	MD. TAMIM	22-46918-1	BSc [CSE]	
04	MD. FAHIM RAHMAN	21-45303-2	BSc [CSE]	

Faculty use only

FACULTY COMMENTS		
	Marks Obtained	
	Total Marks	

Dataset: **Introduction**

This dataset contains financial and personal information of 201 individuals, useful for analyzing loan applications and predicting loan repayment behavior. The dataset includes 14 attributes:

- **person_age**: Age of the individual.
- **person_gender**: Gender of the individual.
- **person_education**: Education level of the individual.
- **person_income**: Annual income of the individual.
- **person_emp_exp**: Employment experience in years.
- **person_home_ownership**: Home ownership status (e.g., RENT, OWN, MORTGAGE).
- **loan_amnt**: Loan amount requested.
- **loan_intent**: Purpose of the loan (e.g., PERSONAL, EDUCATION, MEDICAL).
- **loan_int_rate**: Interest rate on the loan.
- **loan_percent_income**: Percentage of income allocated to loan repayment.
- **cb_person_cred_hist_length**: Length of the individual's credit history in years.
- **credit_score**: Credit score of the individual.
- **previous_loan_defaults_on_file**: Indicates if there are previous loan defaults (Yes/No).
- **loan_status**: Outcome of the loan application (e.g., 1 for approved, 0 for denied).

While comprehensive, the dataset has some missing values in attributes like **person_age**, **person_income**, **person_education**, and **loan_status**. There are also potential inconsistencies, such as a typo in the **person_home_ownership** column (e.g., "RENTT"). These issues make the dataset an excellent candidate for preprocessing, data cleaning, and exploratory data analysis tasks.

Dataset: **About data**

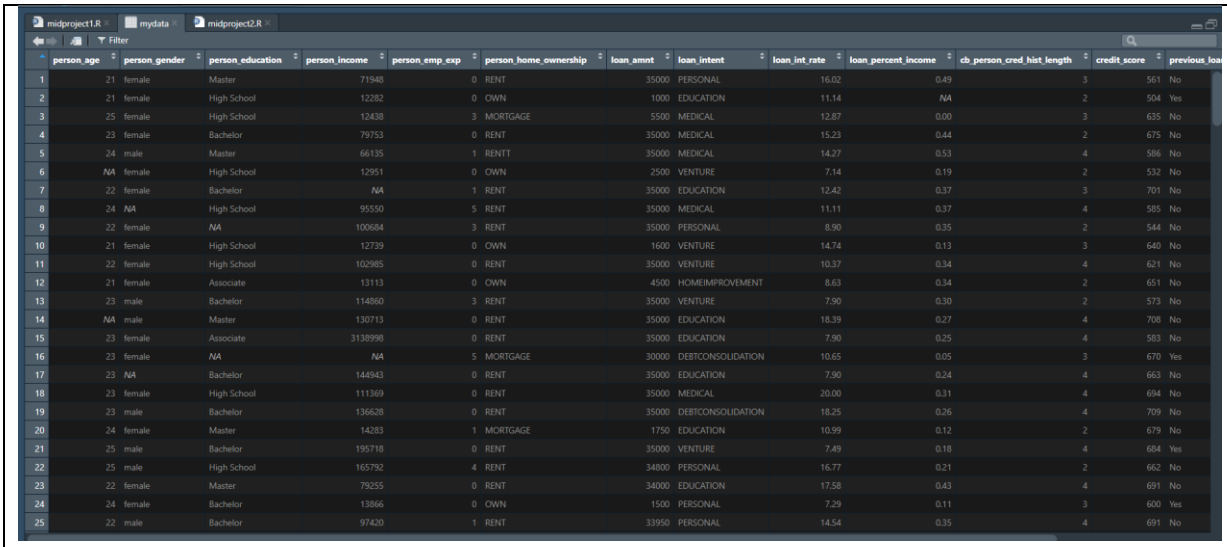
- **Library Use:**

```
library(readxl)

library(dplyr)
```

- **Read Data**

```
mydata <- read_excel("C:/Users/AZMINUR RAHMAN/OneDrive - American
International University-Bangladesh/2024-2025, Fall/INTRODUCTION TO DATA
SCIENCE [C]/Mid/Lab/Project/Midterm_Dataset_Section(C).xlsx", sheet =
"Sheet1")

View(mydata)
```

```
str(mydata)
summary(mydata)

num_instances <- nrow(mydata)
num_attributes <- ncol(mydata)
print(paste("Number of instances (rows):", num_instances))
print(paste("Number of columns:", num_attributes))

missing_values_indices <- lapply(mydata, function(x) {
  if (is.numeric(x) || is.character(x)) {
    return(which(is.na(x) | x == ""))
  } else {
    return(NULL)
  }
})
print(missing_values_indices)
```



	person_age	person_gender	person_education	person_income	person_emp_exp	person_home_ownership	loan_amnt	loan_intent	loan_int_rate	loan_percent_income	cb_person_cred_hist_length	credit_score	previous_loa
1	21	female	Master	71948	0	RENT	35000	PERSONAL	16.02	0.49	3	561	No
2	21	female	High School	12282	0	OWN	1000	EDUCATION	11.14	NA	2	504	Yes
3	25	female	High School	12438	3	MORTGAGE	5500	MEDICAL	12.87	0.00	3	635	No
4	23	female	Bachelor	79753	0	RENT	35000	MEDICAL	15.23	0.44	2	675	No
5	24	male	Master	66135	1	RENTT	35000	MEDICAL	14.27	0.53	4	586	No
6	NA	female	High School	12951	0	OWN	2500	VENTURE	7.14	0.19	2	532	No
7	22	female	Bachelor	NA	1	RENT	35000	EDUCATION	12.42	0.37	3	701	No
8	24	NA	High School	95550	5	RENT	35000	MEDICAL	11.11	0.37	4	585	No
9	22	female	NA	100684	3	RENT	35000	PERSONAL	8.90	0.35	2	544	No
10	21	female	High School	12739	0	OWN	1600	VENTURE	14.74	0.13	3	640	No
11	22	female	High School	102985	0	RENT	35000	VENTURE	10.37	0.34	4	621	No
12	21	female	Associate	13113	0	OWN	4500	HOMEIMPROVEMENT	8.63	0.34	2	651	No
13	23	male	Bachelor	114860	3	RENT	35000	VENTURE	7.90	0.30	2	573	No
14	NA	male	Master	130713	0	RENT	35000	EDUCATION	18.39	0.27	4	708	No
15	23	female	Associate	3138998	0	RENT	35000	EDUCATION	7.90	0.25	4	583	No
16	23	female	NA	NA	5	MORTGAGE	30000	DEBTCONSOLIDATION	10.65	0.05	3	670	Yes
17	23	NA	Bachelor	144943	0	RENT	35000	EDUCATION	7.90	0.24	4	663	No
18	23	female	High School	111369	0	RENT	35000	MEDICAL	20.00	0.31	4	694	No
19	23	male	Bachelor	136628	0	RENT	35000	DEBTCONSOLIDATION	18.25	0.26	4	709	No
20	24	female	Master	14283	1	MORTGAGE	1750	EDUCATION	10.99	0.12	2	679	No
21	25	male	Bachelor	195718	0	RENT	35000	VENTURE	7.49	0.18	4	684	Yes
22	25	male	High School	165792	4	RENT	34800	PERSONAL	16.77	0.21	2	662	No
23	22	female	Master	79255	0	RENT	34000	EDUCATION	17.58	0.43	4	691	No
24	24	female	Bachelor	13866	0	OWN	1500	PERSONAL	7.29	0.11	3	600	Yes
25	22	male	Bachelor	97420	1	RENT	33950	PERSONAL	14.54	0.35	4	691	No

```
> str(mydata)
tibble [201 × 14] (S3: tbl_df/tbl/data.frame)
 $ person_age                    : num [1:201] 21 21 25 23 24 NA 22 24 22 21 ...
 $ person_gender                 : chr [1:201] "female" "female" "female" "female" ...
 $ person_education              : chr [1:201] "Master" "High School" "High School" "Bachelor" ...
 $ person_income                 : num [1:201] 71948 12282 12438 79753 66135 ...
 $ person_emp_exp                : num [1:201] 0 0 3 0 1 0 1 5 3 0 ...
 $ person_home_ownership         : chr [1:201] "RENT" "OWN" "MORTGAGE" "RENT" ...
 $ loan_amnt                     : num [1:201] 35000 1000 5500 35000 35000 2500 35000 35000 35000 1600 ...
 $ loan_intent                   : chr [1:201] "PERSONAL" "EDUCATION" "MEDICAL" "MEDICAL" ...
 $ loan_int_rate                 : num [1:201] 16 11.1 12.9 15.2 14.3 ...
 $ loan_percent_income           : num [1:201] 0.49 NA 0 0.44 0.53 0.19 0.37 0.37 0.35 0.13 ...
 $ cb_person_cred_hist_length    : num [1:201] 3 2 3 2 4 2 3 4 2 3 ...
 $ credit_score                  : num [1:201] 561 504 635 675 586 532 701 585 544 640 ...
 $ previous_loan_defaults_on_file: chr [1:201] "No" "Yes" "No" "No" ...
 $ loan_status                   : num [1:201] 1 0 1 1 1 1 1 1 NA 1 ...
```

```
> summary(mydata)
   person_age     person_gender      person_education   person_income     person_emp_exp
 Min.   : 21.00   Length:201         Length:201         Min.   :  12282   Min.   :  0.000
 1st Qu.: 22.00   Class :character   Class :character   1st Qu.:  60501   1st Qu.:  0.000
 Median : 23.00   Mode  :character   Mode  :character   Median :  85284   Median :  1.000
 Mean   : 27.39                                        Mean   : 149875   Mean   :  2.761
 3rd Qu.: 25.00                                        3rd Qu.: 241060   3rd Qu.:  3.000
 Max.   :350.00                                        Max.   :3138998   Max.   :125.000
 NA's   :4                                             NA's   :4
 person_home_ownership   loan_amnt     loan_intent        loan_int_rate   loan_percent_income
 Length:201           Min.   : 1000   Length:201         Min.   : 5.42   Min.   :0.0000
 Class :character     1st Qu.:10000   Class :character   1st Qu.:10.65   1st Qu.:0.0900
 Mode  :character     Median :25000   Mode  :character   Median :11.83   Median :0.2350
                      Mean   :20553                      Mean   :12.29   Mean   :0.2293
                      3rd Qu.:28000                      3rd Qu.:14.42   3rd Qu.:0.3425
                      Max.   :35000                      Max.   :20.00   Max.   :0.5300
                                                                         NA's   :1
 cb_person_cred_hist_length  credit_score   previous_loan_defaults_on_file  loan_status
 Min.   :2.00                Min.   :484.0   Length:201                     Min.   :0.0000
 1st Qu.:2.00                1st Qu.:595.0   Class :character               1st Qu.:0.0000
 Median :3.00                Median :630.0   Mode  :character               Median :1.0000
 Mean   :2.99                Mean   :628.5                                  Mean   :0.6162
 3rd Qu.:4.00                3rd Qu.:665.0                                  3rd Qu.:1.0000
 Max.   :4.00                Max.   :807.0                                  Max.   :1.0000
                                                                            NA's   :3
> # Count rows and columns
> num_instances <- nrow(mydata)
> num_attributes <- ncol(mydata)
> print(paste("Number of instances (rows):", num_instances))
[1] "Number of instances (rows): 201"
> print(paste("Number of columns:", num_attributes))
[1] "Number of columns: 14"
>
```

```
> missing_values_indices <- lapply(mydata, function(x) {
+   if (is.numeric(x) || is.character(x)) {
+     return(which(is.na(x) | x == ""))
+   } else {
+     return(NULL)
+   }
+ })
> print(missing_values_indices)
$person_age
[1]  6 14 28 35

$person_gender
[1]   8  17 190 198

$person_education
[1]  9 16

$person_income
[1]  7 16 32 40

$person_emp_exp
integer(0)

$person_home_ownership
integer(0)

$loan_amnt
integer(0)

$loan_intent
integer(0)

$loan_int_rate
integer(0)

$loan_percent_income
[1] 2

$cb_person_cred_hist_length
integer(0)

$credit_score
integer(0)

$previous_loan_defaults_on_file
integer(0)

$loan_status
[1]  9 15 18

>
```

Description: Load Dataset to mydata. Found the total number of row and column using ncol() and nrow() function.Then we found all the missing values with the help of is.numeric() and is.character() function. Used print() functions to show the output in one line.

Dataset: **Data Preparation & Exploration**

Column: person_age

- **Missing Value Imputation: Replacing NA Values with Median**

```
age_median <- round(median(mydata$person_age, na.rm = TRUE))
mydata$person_age[is.na(mydata$person_age)] <- age_median
```

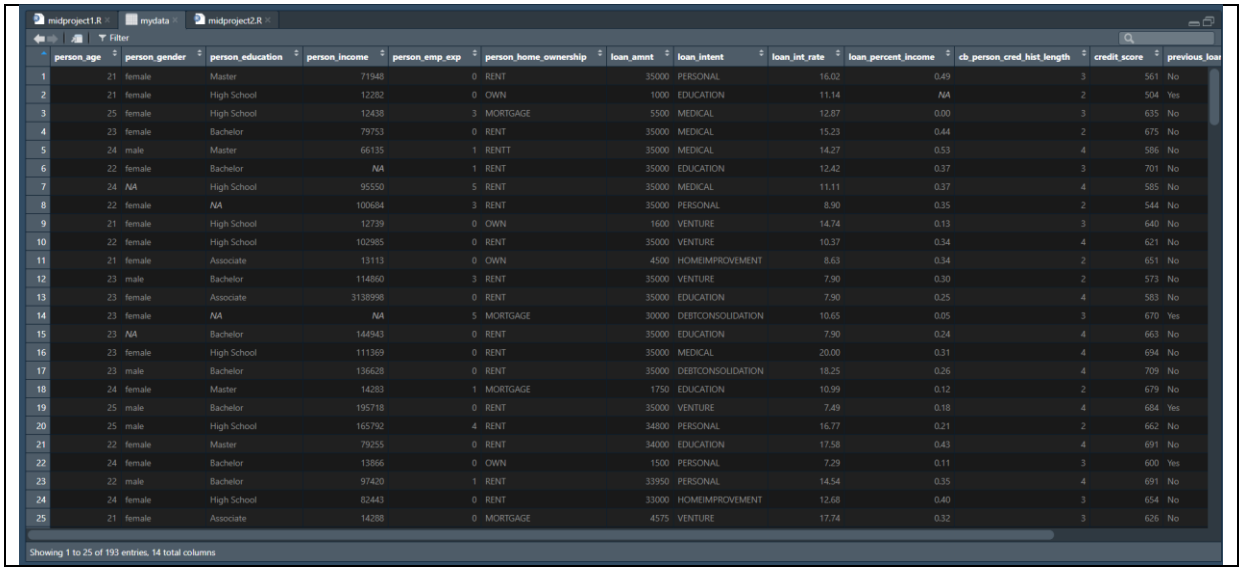
- **Outlier Detection and Removal with Interquartile Range (IQR) Method**

```
Q1 <- quantile(mydata$person_age, 0.25, na.rm = TRUE)
Q3 <- quantile(mydata$person_age, 0.75, na.rm = TRUE)
IQR_value <- Q3 - Q1
threshold <- 1.5
outlier_condition <- (mydata$person_age < (Q1 - threshold * IQR_value)) |
  (mydata$person_age > (Q3 + threshold * IQR_value))
```

- **Serial Update After Removing Outliers**

```
 mydata <- mydata %>%
  filter(!outlier_condition) %>%
  arrange(row_number())
View(mydata)
```

Output:



midproject1.R | mydata | midproject2.R

Filter

	person_age	person_gender	person_education	person_income	person_emp_exp	person_home_ownership	loan_amnt	loan_intent	loan_int_rate	loan_percent_income	cb_person_cred_hist_length	credit_score	previous_loa
1	21	female	Master	71948	0	RENT	35000	PERSONAL	16.02	0.49	3	561	No
2	21	female	High School	12282	0	OWN	1000	EDUCATION	11.14	NA	2	504	Yes
3	25	female	High School	12438	3	MORTGAGE	5500	MEDICAL	12.87	0.00	3	635	No
4	23	female	Bachelor	79753	0	RENT	35000	MEDICAL	15.23	0.44	2	675	No
5	24	male	Master	66135	1	RENTT	35000	MEDICAL	14.27	0.53	4	586	No
6	22	female	Bachelor	NA	1	RENT	35000	EDUCATION	12.42	0.37	3	701	No
7	24	NA	High School	95550	5	RENT	35000	MEDICAL	11.11	0.37	4	585	No
8	22	female	NA	100684	3	RENT	35000	PERSONAL	8.90	0.35	2	544	No
9	21	female	High School	12739	0	OWN	1600	VENTURE	14.74	0.13	3	640	No
10	22	female	High School	102985	0	RENT	35000	VENTURE	10.37	0.34	4	621	No
11	21	female	Associate	13113	0	OWN	4500	HOMEIMPROVEMENT	8.63	0.34	2	651	No
12	23	male	Bachelor	114860	3	RENT	35000	VENTURE	7.90	0.30	2	573	No
13	23	female	Associate	3138998	0	RENT	35000	EDUCATION	7.90	0.25	4	583	No
14	23	female	NA	NA	5	MORTGAGE	30000	DEBTCONSOLIDATION	10.65	0.05	3	670	Yes
15	23	NA	Bachelor	144943	0	RENT	35000	EDUCATION	7.90	0.24	4	663	No
16	23	female	High School	111369	0	RENT	35000	MEDICAL	20.00	0.31	4	694	No
17	23	male	Bachelor	136628	0	RENT	35000	DEBTCONSOLIDATION	18.25	0.26	4	709	No
18	24	female	Master	14283	1	MORTGAGE	1750	EDUCATION	10.99	0.12	2	679	No
19	25	male	Bachelor	195718	0	RENT	35000	VENTURE	7.49	0.18	4	684	Yes
20	25	male	High School	165792	4	RENT	34800	PERSONAL	16.77	0.21	2	662	No
21	22	female	Master	79255	0	RENT	34000	EDUCATION	17.58	0.43	4	691	No
22	24	female	Bachelor	13866	0	OWN	1500	PERSONAL	7.29	0.11	3	600	Yes
23	22	male	Bachelor	97420	1	RENT	33950	PERSONAL	14.54	0.35	4	691	No
24	24	female	High School	82443	0	RENT	33000	HOMEIMPROVEMENT	12.68	0.40	3	654	No
25	21	female	Associate	14288	0	MORTGAGE	4575	VENTURE	17.74	0.32	3	626	No

Showing 1 to 25 of 193 entries, 14 total columns

Description: First, we replaced the NA values with the median. Then with the help of IQR method we have found the Outliers then removed those Outliers. To update the rows after removing outliers we have used pipe operator (%>%)

Column: person_gender

- **Detecting and Recovering Noisy Values: There is no Noisy values**

```
unique_values_gender <- unique(mydata$person_gender)

print(unique_values_gender)
```

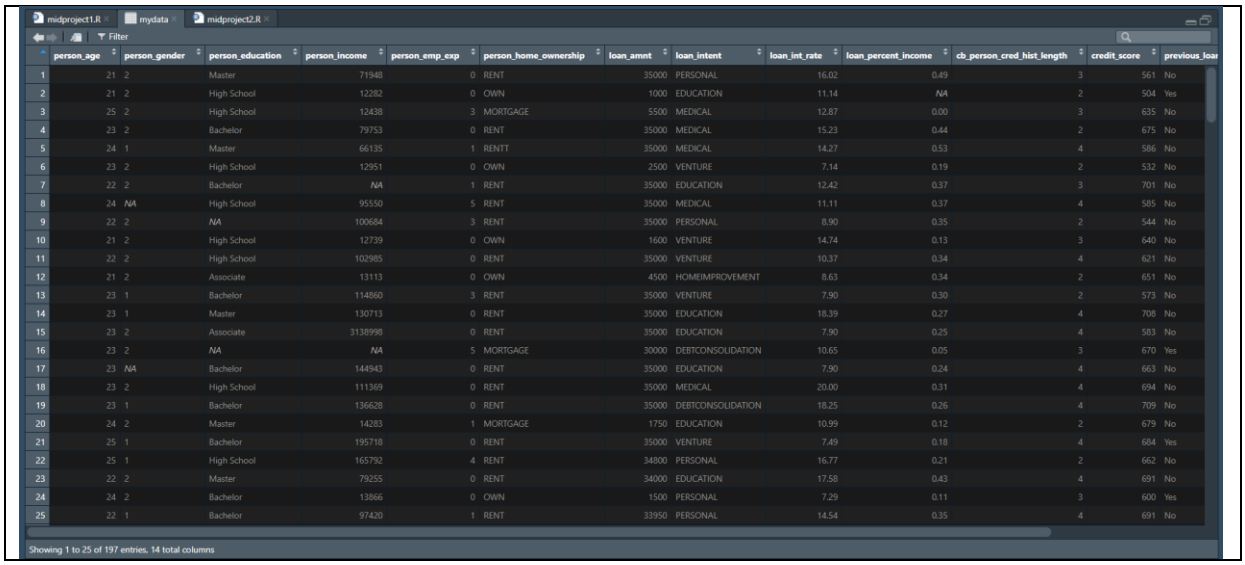
Output:

```
> unique_values_gender <- unique(mydata$person_gender)
> print(unique_values_gender)
[1] "female" "male"   NA
>
```

- **Data conversion: Converting categorical attributes to numeric (Gender is a categorical data)**

```
mydata$person_gender <- tolower(mydata$person_gender)

mydata$person_gender <- factor(mydata$person_gender,
                               levels = c("male", "female"),
                               labels = c(1, 2))

View(mydata)
```

Output:



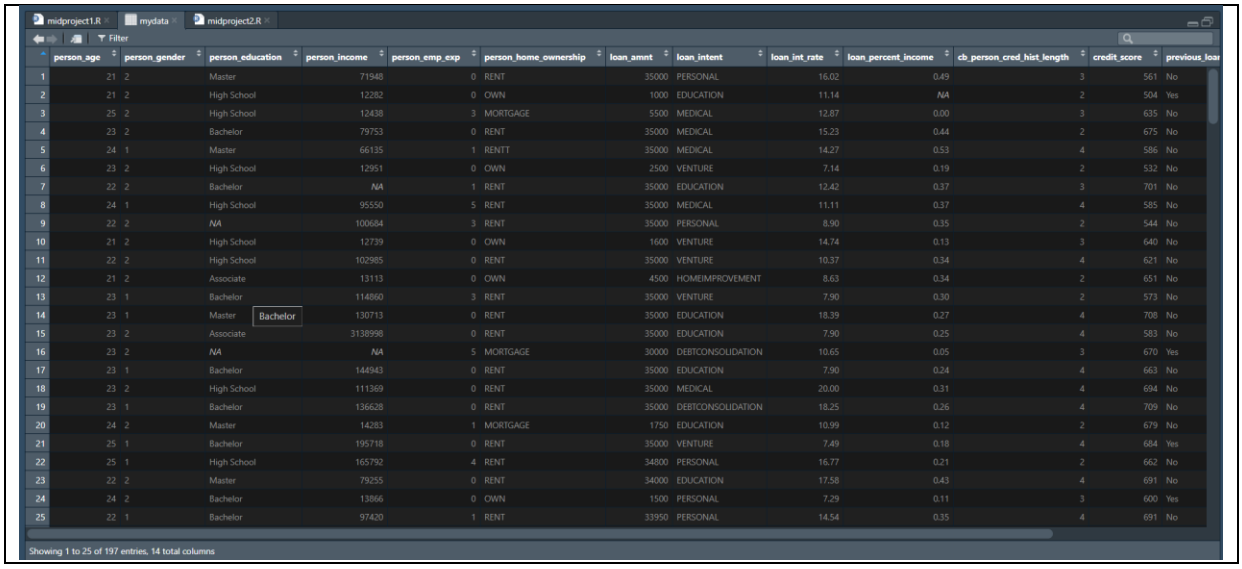
	person_age	person_gender	person_education	person_income	person_emp_exp	person_home_ownership	loan_amnt	loan_intent	loan_int_rate	loan_percent_income	cb_person_cred_hist_length	credit_score	previous_loan
1	21	2	Master	71948	0	RENT	35000	PERSONAL	16.02	0.49	3	561	No
2	21	2	High School	12282	0	OWN	1000	EDUCATION	11.14	NA	2	504	Yes
3	25	2	High School	12438	3	MORTGAGE	5500	MEDICAL	12.87	0.00	3	635	No
4	23	2	Bachelor	79753	0	RENT	35000	MEDICAL	15.23	0.44	2	675	No
5	24	1	Master	66135	1	RENTT	35000	MEDICAL	14.27	0.53	4	586	No
6	23	2	High School	12951	0	OWN	2500	VENTURE	7.14	0.19	2	532	No
7	22	2	Bachelor	NA	1	RENT	35000	EDUCATION	12.42	0.37	3	701	No
8	24	NA	High School	95550	5	RENT	35000	MEDICAL	11.11	0.37	4	585	No
9	22	2	NA	100684	3	RENT	35000	PERSONAL	8.90	0.35	2	544	No
10	21	2	High School	12739	0	OWN	1600	VENTURE	14.74	0.13	3	640	No
11	22	2	High School	102985	0	RENT	35000	VENTURE	10.37	0.34	4	621	No
12	21	2	Associate	13113	0	OWN	4500	HOMEIMPROVEMENT	8.63	0.34	2	651	No
13	23	1	Bachelor	114860	3	RENT	35000	VENTURE	7.90	0.30	2	573	No
14	23	1	Master	130713	0	RENT	35000	EDUCATION	18.39	0.27	4	708	No
15	23	2	Associate	3138998	0	RENT	35000	EDUCATION	7.90	0.25	4	583	No
16	23	2	NA	NA	5	MORTGAGE	30000	DEBTCONSOLIDATION	10.65	0.05	3	670	Yes
17	23	NA	Bachelor	144943	0	RENT	35000	EDUCATION	7.90	0.24	4	663	No
18	23	2	High School	111369	0	RENT	35000	MEDICAL	20.00	0.31	4	694	No
19	23	1	Bachelor	136628	0	RENT	35000	DEBTCONSOLIDATION	18.25	0.26	4	709	No
20	24	2	Master	14283	1	MORTGAGE	1750	EDUCATION	10.99	0.12	2	679	No
21	25	1	Bachelor	195718	0	RENT	35000	VENTURE	7.49	0.18	4	684	Yes
22	25	1	High School	165792	4	RENT	34800	PERSONAL	16.77	0.21	2	662	No
23	22	2	Master	79255	0	RENT	34000	EDUCATION	17.58	0.43	4	691	No
24	24	2	Bachelor	13866	0	OWN	1500	PERSONAL	7.29	0.11	3	600	Yes
25	22	1	Bachelor	97420	1	RENT	33950	PERSONAL	14.54	0.35	4	691	No

- **Missing Value Imputation: Replacing NA Values with Mode**

```
mode_gender <- as.numeric(names(sort(table(mydata$person_gender),
decreasing = TRUE)[1]))
```

```
mydata$person_gender[is.na(mydata$person_gender)] <- mode_gender


View(mydata)
```

Output:



	person_age	person_gender	person_education	person_income	person_emp_exp	person_home_ownership	loan_amnt	loan_intent	loan_int_rate	loan_percent_income	cb_person_cred_hist_length	credit_score	previous_loa
1	21	2	Master	71948	0	RENT	35000	PERSONAL	16.02	0.49	3	561	No
2	21	2	High School	12282	0	OWN	1000	EDUCATION	11.14	NA	2	504	Yes
3	25	2	High School	12438	3	MORTGAGE	5500	MEDICAL	12.87	0.00	3	635	No
4	23	2	Bachelor	79753	0	RENT	35000	MEDICAL	15.23	0.44	2	675	No
5	24	1	Master	66135	1	RENTT	35000	MEDICAL	14.27	0.53	4	586	No
6	23	2	High School	12951	0	OWN	2500	VENTURE	7.14	0.19	2	532	No
7	22	2	Bachelor	NA	1	RENT	35000	EDUCATION	12.42	0.37	3	701	No
8	24	1	High School	95550	5	RENT	35000	MEDICAL	11.11	0.37	4	585	No
9	22	2	NA	100684	3	RENT	35000	PERSONAL	8.90	0.35	2	544	No
10	21	2	High School	12739	0	OWN	1600	VENTURE	14.74	0.13	3	640	No
11	22	2	High School	102985	0	RENT	35000	VENTURE	10.37	0.34	4	621	No
12	21	2	Associate	13113	0	OWN	4500	HOMEIMPROVEMENT	8.63	0.34	2	651	No
13	23	1	Bachelor	114860	3	RENT	35000	VENTURE	7.90	0.30	2	573	No
14	23	1	Master	130713	0	RENT	35000	EDUCATION	18.39	0.27	4	708	No
15	23	2	Associate	3138998	0	RENT	35000	EDUCATION	7.90	0.25	4	583	No
16	23	2	NA	NA	5	MORTGAGE	30000	DEBTCONSOLIDATION	10.65	0.05	3	670	Yes
17	23	1	Bachelor	144943	0	RENT	35000	EDUCATION	7.90	0.24	4	663	No
18	23	2	High School	111369	0	RENT	35000	MEDICAL	20.00	0.31	4	694	No
19	23	1	Bachelor	136628	0	RENT	35000	DEBTCONSOLIDATION	18.25	0.26	4	709	No
20	24	2	Master	14283	1	MORTGAGE	1750	EDUCATION	10.99	0.12	2	679	No
21	25	1	Bachelor	195718	0	RENT	35000	VENTURE	7.49	0.18	4	684	Yes
22	25	1	High School	165792	4	RENT	34800	PERSONAL	16.77	0.21	2	662	No
23	22	2	Master	79255	0	RENT	34000	EDUCATION	17.58	0.43	4	691	No
24	24	2	Bachelor	13866	0	OWN	1500	PERSONAL	7.29	0.11	3	600	Yes
25	22	1	Bachelor	97420	1	RENT	33950	PERSONAL	14.54	0.35	4	691	No

Description: At first we found the unique values using unique() then we have converted categorical attributes to numeric with the help of factor function.

Column: person_education

- **Detecting and Recovering Noisy Values: There is no Noisy values**

```
unique_education <- unique(mydata$person_education)

print(unique_education)
```

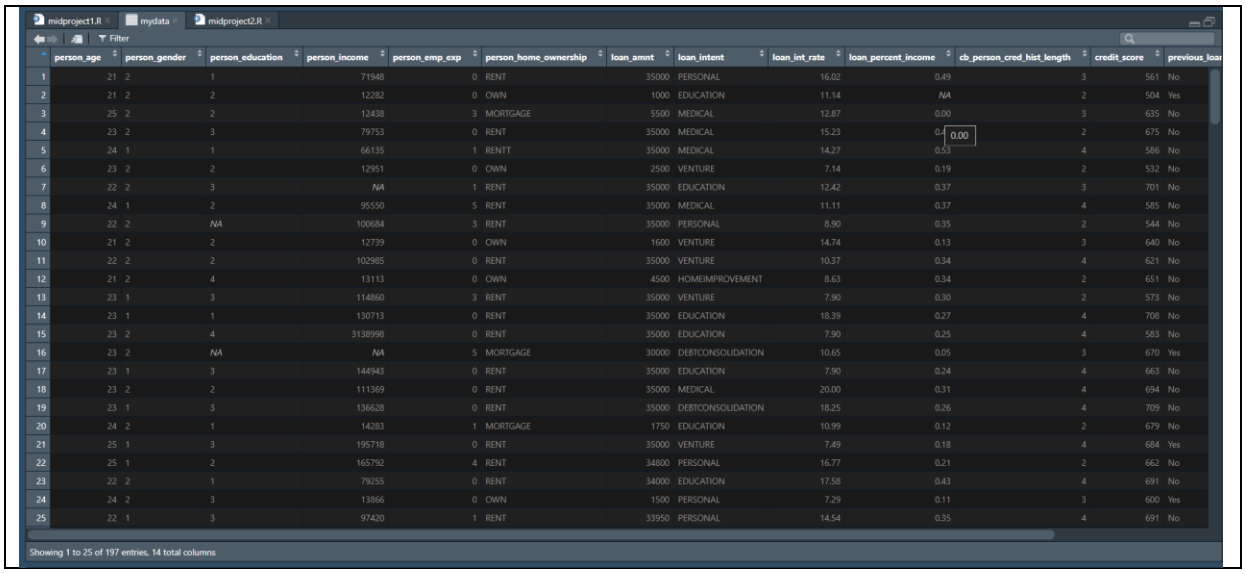
Output:

```
> unique_education <- unique(mydata$person_education)
> print(unique_education)
[1] "Master"      "High School" "Bachelor"    NA            "Associate"   "Doctorate"
>
```

- **Data conversion: Converting categorical attributes to numeric**

```
mydata$person_education <- tolower(mydata$person_education)

mydata$person_education <- factor(mydata$person_education,

                                  levels = c("master", "high school",
"bachelor", "associate", "doctorate"),

                                  labels = c(1, 2, 3, 4, 5))

View(mydata)
```

Output:

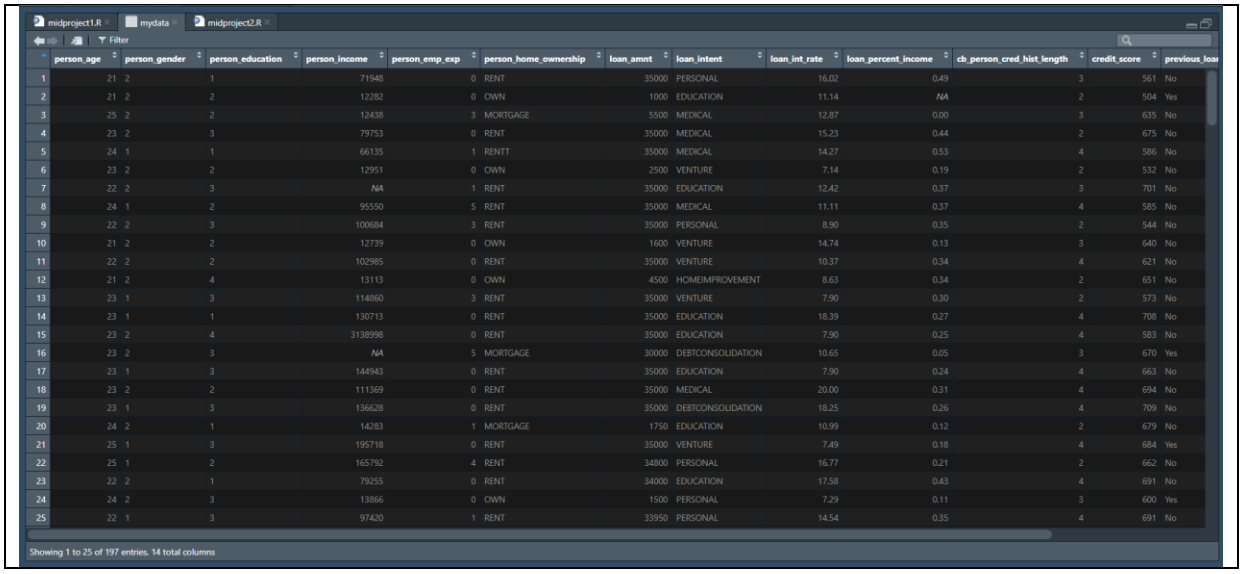


	person_age	person_gender	person_education	person_income	person_emp_exp	person_home_ownership	loan_amnt	loan_intent	loan_int_rate	loan_percent_income	cb_person_cred_hist_length	credit_score	previous_loan
1	21	2	1	71948	0	RENT	35000	PERSONAL	16.02	0.49	3	561	No
2	21	2	2	12282	0	OWN	1000	EDUCATION	11.14	NA	2	504	Yes
3	25	2	2	12438	3	MORTGAGE	5500	MEDICAL	12.87	0.00	3	635	No
4	23	2	3	79753	0	RENT	35000	MEDICAL	15.23	0.44	2	675	No
5	24	1	1	66135	1	RENTT	35000	MEDICAL	14.27	0.53	4	586	No
6	23	2	2	12951	0	OWN	2500	VENTURE	7.14	0.19	2	532	No
7	22	2	3	NA	1	RENT	35000	EDUCATION	12.42	0.37	3	701	No
8	24	1	2	95550	5	RENT	35000	MEDICAL	11.11	0.37	4	585	No
9	22	2	NA	100684	3	RENT	35000	PERSONAL	8.90	0.35	2	544	No
10	21	2	2	12739	0	OWN	1600	VENTURE	14.74	0.13	3	640	No
11	22	2	2	102985	0	RENT	35000	VENTURE	10.37	0.34	4	621	No
12	21	2	4	13113	0	OWN	4500	HOMEIMPROVEMENT	8.63	0.34	2	651	No
13	23	1	3	114860	3	RENT	35000	VENTURE	7.90	0.30	2	573	No
14	23	1	1	130713	0	RENT	35000	EDUCATION	18.39	0.27	4	708	No
15	23	2	4	3138998	0	RENT	35000	EDUCATION	7.90	0.25	4	583	No
16	23	2	NA	NA	5	MORTGAGE	30000	DEBTCONSOLIDATION	10.65	0.05	3	670	Yes
17	23	1	3	144943	0	RENT	35000	EDUCATION	7.90	0.24	4	663	No
18	23	2	2	111369	0	RENT	35000	MEDICAL	20.00	0.31	4	694	No
19	23	1	3	136628	0	RENT	35000	DEBTCONSOLIDATION	18.25	0.26	4	709	No
20	24	2	1	14283	1	MORTGAGE	1750	EDUCATION	10.99	0.12	2	679	No
21	25	1	3	195718	0	RENT	35000	VENTURE	7.49	0.18	4	684	Yes
22	25	1	2	165792	4	RENT	34800	PERSONAL	16.77	0.21	2	662	No
23	22	2	1	79255	0	RENT	34000	EDUCATION	17.58	0.43	4	691	No
24	24	2	3	13866	0	OWN	1500	PERSONAL	7.29	0.11	3	600	Yes
25	22	1	3	97420	1	RENT	33950	PERSONAL	14.54	0.35	4	691	No

- **Missing Value Imputation: Replacing NA Values with Mode**

```
mode_education <- names(which.max(table(mydata$person_education)))
mydata$person_education[is.na(mydata$person_education)] <- mode_education

View(mydata)
```

Output:



	person_age	person_gender	person_education	person_income	person_emp_exp	person_home_ownership	loan_amnt	loan_intent	loan_int_rate	loan_percent_income	cb_person_cred_hist_length	credit_score	previous_loan
1	21	2	1	71948	0	RENT	35000	PERSONAL	16.02	0.49	3	561	No
2	21	2	2	12282	0	OWN	1000	EDUCATION	11.14	NA	2	504	Yes
3	25	2	2	12438	3	MORTGAGE	5500	MEDICAL	12.87	0.00	3	635	No
4	23	2	3	79753	0	RENT	35000	MEDICAL	15.23	0.44	2	675	No
5	24	1	1	66135	1	RENTT	35000	MEDICAL	14.27	0.53	4	586	No
6	23	2	2	12951	0	OWN	2500	VENTURE	7.14	0.19	2	532	No
7	22	2	3	NA	1	RENT	35000	EDUCATION	12.42	0.37	3	701	No
8	24	1	2	95550	5	RENT	35000	MEDICAL	11.11	0.37	4	585	No
9	22	2	3	100684	3	RENT	35000	PERSONAL	8.90	0.35	2	544	No
10	21	2	2	12739	0	OWN	1600	VENTURE	14.74	0.13	3	640	No
11	22	2	2	102985	0	RENT	35000	VENTURE	10.37	0.34	4	621	No
12	21	2	4	13113	0	OWN	4500	HOMEIMPROVEMENT	8.63	0.34	2	651	No
13	23	1	3	114860	3	RENT	35000	VENTURE	7.90	0.30	2	573	No
14	23	1	1	130713	0	RENT	35000	EDUCATION	18.39	0.27	4	708	No
15	23	2	4	3138998	0	RENT	35000	EDUCATION	7.90	0.25	4	583	No
16	23	2	3	NA	5	MORTGAGE	30000	DEBTCONSOLIDATION	10.65	0.05	3	670	Yes
17	23	1	3	144943	0	RENT	35000	EDUCATION	7.90	0.24	4	663	No
18	23	2	2	111369	0	RENT	35000	MEDICAL	20.00	0.31	4	694	No
19	23	1	3	136628	0	RENT	35000	DEBTCONSOLIDATION	18.25	0.26	4	709	No
20	24	2	1	14283	1	MORTGAGE	1750	EDUCATION	10.99	0.12	2	679	No
21	25	1	3	195718	0	RENT	35000	VENTURE	7.49	0.18	4	684	Yes
22	25	1	2	165792	4	RENT	34800	PERSONAL	16.77	0.21	2	662	No
23	22	2	1	79255	0	RENT	34000	EDUCATION	17.58	0.43	4	691	No
24	24	2	3	13866	0	OWN	1500	PERSONAL	7.29	0.11	3	600	Yes
25	22	1	3	97420	1	RENT	33950	PERSONAL	14.54	0.35	4	691	No

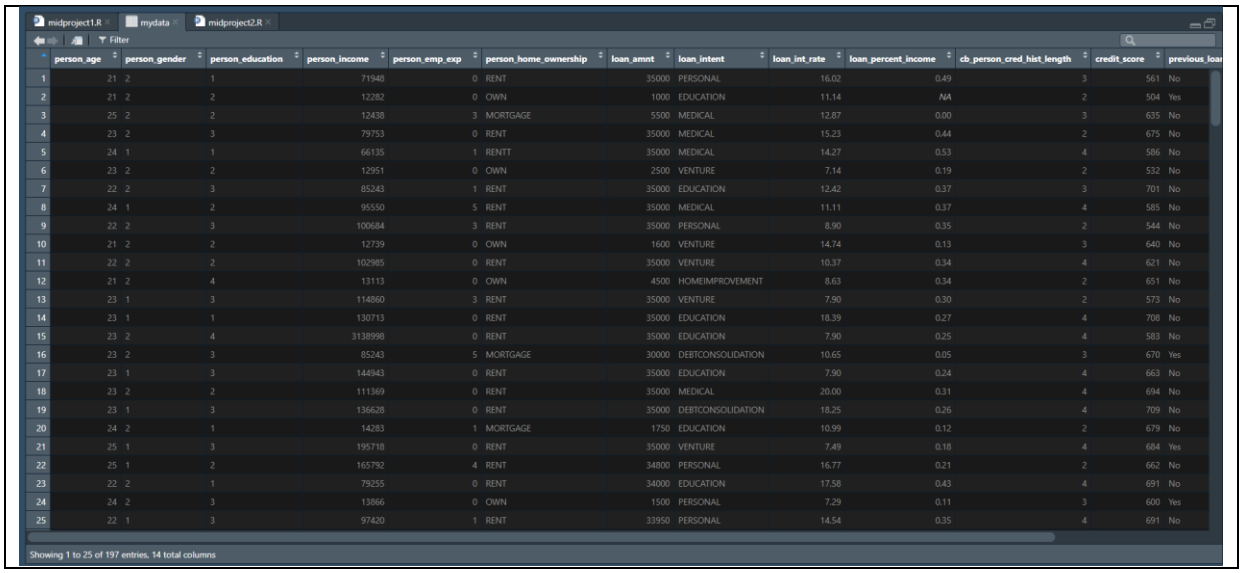
Description: At first we found the unique values using unique() then we have converted categorical attributes to numeric with the help of factor function.

Column: person_income

- **Missing Value Imputation: Replacing NA Values with Median**

```
income_median <- median(mydata$person_income, na.rm = TRUE)
mydata$person_income[is.na(mydata$person_income)] <- income_median

View(mydata)
```

Output:



	person_age	person_gender	person_education	person_income	person_emp_exp	person_home_ownership	loan_amnt	loan_intent	loan_int_rate	loan_percent_income	cb_person_cred_hist_length	credit_score	previous_loan
1	21	2	1	71948	0	RENT	35000	PERSONAL	16.02	0.49	3	561	No
2	21	2	2	12282	0	OWN	1000	EDUCATION	11.14	NA	2	504	Yes
3	25	2	2	12438	3	MORTGAGE	5500	MEDICAL	12.87	0.00	3	635	No
4	23	2	3	79753	0	RENT	35000	MEDICAL	15.23	0.44	2	675	No
5	24	1	1	66135	1	RENTT	35000	MEDICAL	14.27	0.53	4	586	No
6	23	2	2	12951	0	OWN	2500	VENTURE	7.14	0.19	2	532	No
7	22	2	3	85243	1	RENT	35000	EDUCATION	12.42	0.37	3	701	No
8	24	1	2	95550	5	RENT	35000	MEDICAL	11.11	0.37	4	585	No
9	22	2	3	100684	3	RENT	35000	PERSONAL	8.90	0.35	2	544	No
10	21	2	2	12739	0	OWN	1600	VENTURE	14.74	0.13	3	640	No
11	22	2	2	102985	0	RENT	35000	VENTURE	10.37	0.34	4	621	No
12	21	2	4	13113	0	OWN	4500	HOMEIMPROVEMENT	8.63	0.34	2	651	No
13	23	1	3	114860	3	RENT	35000	VENTURE	7.90	0.30	2	573	No
14	23	1	1	130713	0	RENT	35000	EDUCATION	18.39	0.27	4	708	No
15	23	2	4	3138998	0	RENT	35000	EDUCATION	7.90	0.25	4	583	No
16	23	2	3	85243	5	MORTGAGE	30000	DEBTCONSOLIDATION	10.65	0.05	3	670	Yes
17	23	1	3	144943	0	RENT	35000	EDUCATION	7.90	0.24	4	663	No
18	23	2	2	111369	0	RENT	35000	MEDICAL	20.00	0.31	4	694	No
19	23	1	3	136628	0	RENT	35000	DEBTCONSOLIDATION	18.25	0.26	4	709	No
20	24	2	1	14283	1	MORTGAGE	1750	EDUCATION	10.99	0.12	2	679	No
21	25	1	3	195718	0	RENT	35000	VENTURE	7.49	0.18	4	684	Yes
22	25	1	2	165792	4	RENT	34800	PERSONAL	16.77	0.21	2	662	No
23	22	2	1	79255	0	RENT	34000	EDUCATION	17.58	0.43	4	691	No
24	24	2	3	13866	0	OWN	1500	PERSONAL	7.29	0.11	3	600	Yes
25	22	1	3	97420	1	RENT	33950	PERSONAL	14.54	0.35	4	691	No

Showing 1 to 25 of 197 entries, 14 total columns

Description: Using median() value for replacing NA.

Column: person_emp_exp

- **The 'person_emp_exp' column exhibits optimal data quality with no missing or invalid values.**

```
missing_values_emp_exp <- sum(is.na(mydata$person_emp_exp))
print(missing_values_emp_exp)
```

Output:

```
> missing_values_emp_exp <- sum(is.na(mydata$person_emp_exp))
> print(missing_values_emp_exp)
[1] 0
>
```

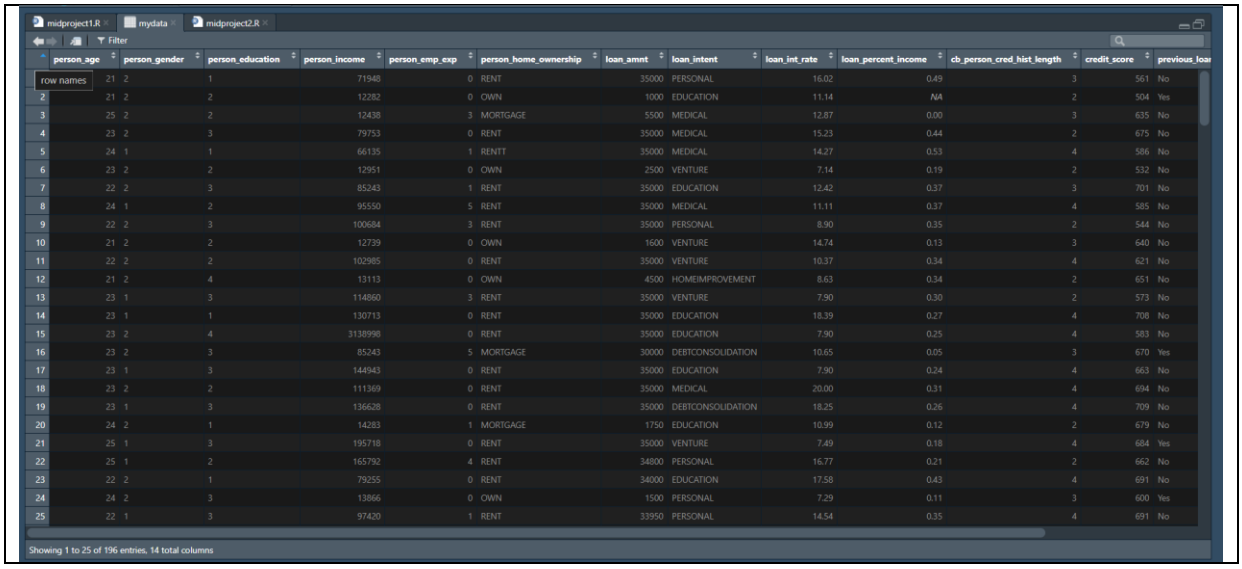
- **Outlier Detection and Removal with Interquartile Range (IQR) Method**

```
Q1 <- quantile(mydata$person_emp_exp, 0.25, na.rm = TRUE)
Q3 <- quantile(mydata$person_emp_exp, 0.75, na.rm = TRUE)
IQR_value <- Q3 - Q1
threshold <- 1.5
outlier_condition <- (mydata$person_emp_exp < (Q1 - threshold * IQR_value))|
  (mydata$person_emp_exp > (Q3 + threshold * IQR_value))
```

- **Serial Update After Removing Outliers**

```
 mydata <- mydata %>%
  filter(!outlier_condition) %>%
  arrange(row_number())
View(mydata)
```

Output:



row names	person_age	person_gender	person_education	person_income	person_emp_exp	person_home_ownership	loan_amnt	loan_intent	loan_int_rate	loan_percent_income	cb_person_cred_hist_length	credit_score	previous_loa
1	21	2	1	71948	0	RENT	35000	PERSONAL	16.02	0.49	3	561	No
2	21	2	2	12282	0	OWN	1000	EDUCATION	11.14	NA	2	504	Yes
3	25	2	2	12438	3	MORTGAGE	5500	MEDICAL	12.87	0.00	3	635	No
4	23	2	3	79753	0	RENT	35000	MEDICAL	15.23	0.44	2	675	No
5	24	1	1	66135	1	RENTT	35000	MEDICAL	14.27	0.53	4	586	No
6	23	2	2	12951	0	OWN	2500	VENTURE	7.14	0.19	2	532	No
7	22	2	3	85243	1	RENT	35000	EDUCATION	12.42	0.37	3	701	No
8	24	1	2	95550	5	RENT	35000	MEDICAL	11.11	0.37	4	585	No
9	22	2	3	100684	3	RENT	35000	PERSONAL	8.90	0.35	2	544	No
10	21	2	2	12739	0	OWN	1600	VENTURE	14.74	0.13	3	640	No
11	22	2	2	102985	0	RENT	35000	VENTURE	10.37	0.34	4	621	No
12	21	2	4	13113	0	OWN	4500	HOMEIMPROVEMENT	8.63	0.34	2	651	No
13	23	1	3	114860	3	RENT	35000	VENTURE	7.90	0.30	2	573	No
14	23	1	1	130713	0	RENT	35000	EDUCATION	18.39	0.27	4	708	No
15	23	2	4	3138998	0	RENT	35000	EDUCATION	7.90	0.25	4	583	No
16	23	2	3	85243	5	MORTGAGE	30000	DEBTCONSOLIDATION	10.65	0.05	3	670	Yes
17	23	1	3	144943	0	RENT	35000	EDUCATION	7.90	0.24	4	663	No
18	23	2	2	111369	0	RENT	35000	MEDICAL	20.00	0.31	4	694	No
19	23	1	3	136628	0	RENT	35000	DEBTCONSOLIDATION	18.25	0.26	4	709	No
20	24	2	1	14283	1	MORTGAGE	1750	EDUCATION	10.99	0.12	2	679	No
21	25	1	3	195718	0	RENT	35000	VENTURE	7.49	0.18	4	684	Yes
22	25	1	2	165792	4	RENT	34800	PERSONAL	16.77	0.21	2	662	No
23	22	2	1	79255	0	RENT	34000	EDUCATION	17.58	0.43	4	691	No
24	24	2	3	13866	0	OWN	1500	PERSONAL	7.29	0.11	3	600	Yes
25	22	1	3	97420	1	RENT	33950	PERSONAL	14.54	0.35	4	691	No

Showing 1 to 25 of 196 entries, 14 total columns

Description: By the help of IQR method we have found the Outliers then removed those Outliers. To update the rows after removing outliers we have used pipe operator (%>%)

Column: person_home_ownership

- **Detecting and Recovering Noisy Values:**

```
unique_home_ownership <- unique(mydata$person_home_ownership)

print(unique_home_ownership)
```

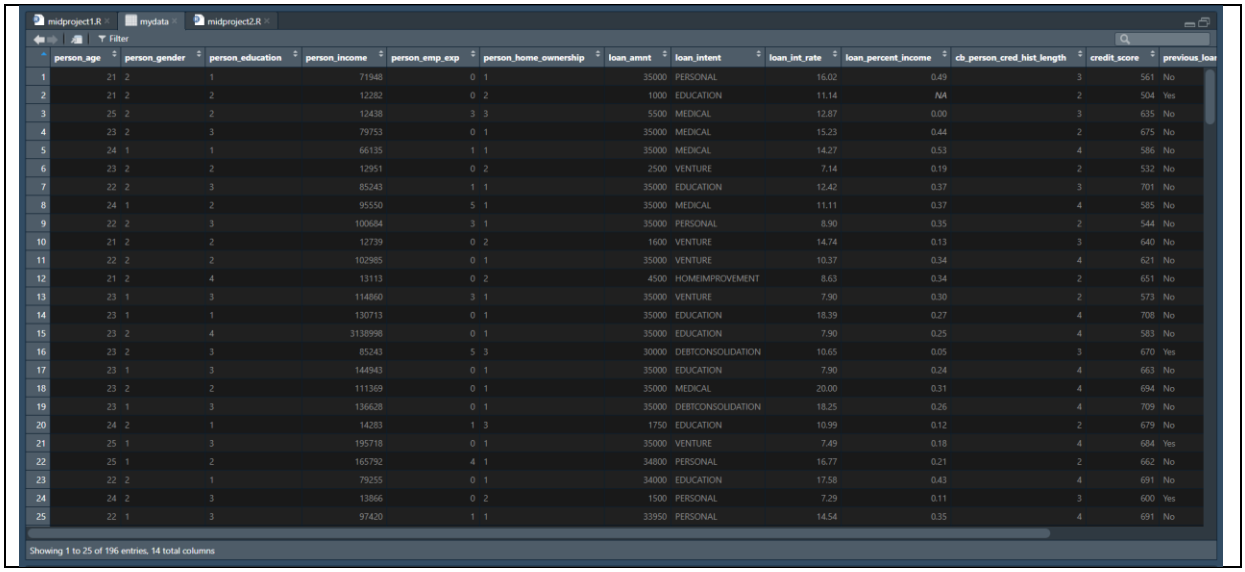
Output:

```
> unique_home_ownership <- unique(mydata$person_home_ownership)
> print(unique_home_ownership)
[1] "RENT"     "OWN"      "MORTGAGE" "RENTT"    "OOWN"     "OTHER"
>
```

- **Data conversion: Converting categorical attributes to numeric and replaces all instances of "rentt" with "rent" and "oown" with "own"**

```
mydata$person_home_ownership <- tolower(mydata$person_home_ownership)

mydata$person_home_ownership <- ifelse(mydata$person_home_ownership ==
"rentt", "rent", mydata$person_home_ownership)

mydata$person_home_ownership <- ifelse(mydata$person_home_ownership ==
"oown", "own", mydata$person_home_ownership)


mydata$person_home_ownership <- factor(mydata$person_home_ownership,

                                       levels = c("rent", "own", "mortgage",
"other"),

                                       labels = c(1, 2, 3, 4))

View(mydata)
```

Output:



	person_age	person_gender	person_education	person_income	person_emp_exp	person_home_ownership	loan_amnt	loan_intent	loan_int_rate	loan_percent_income	cb_person_cred_hist_length	credit_score	previous_loan
1	21	2	1	71948	0	1	35000	PERSONAL	16.02	0.49	3	561	No
2	21	2	2	12282	0	2	1000	EDUCATION	11.14	NA	2	504	Yes
3	25	2	2	12438	3	3	5500	MEDICAL	12.87	0.00	3	635	No
4	23	2	3	79753	0	1	35000	MEDICAL	15.23	0.44	2	675	No
5	24	1	1	66135	1	1	35000	MEDICAL	14.27	0.53	4	586	No
6	23	2	2	12951	0	2	2500	VENTURE	7.14	0.19	2	532	No
7	22	2	3	85243	1	1	35000	EDUCATION	12.42	0.37	3	701	No
8	24	1	2	95550	5	1	35000	MEDICAL	11.11	0.37	4	585	No
9	22	2	3	100684	3	1	35000	PERSONAL	8.90	0.35	2	544	No
10	21	2	2	12739	0	2	1600	VENTURE	14.74	0.13	3	640	No
11	22	2	2	102985	0	1	35000	VENTURE	10.37	0.34	4	621	No
12	21	2	4	13113	0	2	4500	HOMEIMPROVEMENT	8.63	0.34	2	651	No
13	23	1	3	114860	3	1	35000	VENTURE	7.90	0.30	2	573	No
14	23	1	1	130713	0	1	35000	EDUCATION	18.39	0.27	4	708	No
15	23	2	4	3138998	0	1	35000	EDUCATION	7.90	0.25	4	583	No
16	23	2	3	85243	5	3	30000	DEBTCONSOLIDATION	10.65	0.05	3	670	Yes
17	23	1	3	144943	0	1	35000	EDUCATION	7.90	0.24	4	663	No
18	23	2	2	111369	0	1	35000	MEDICAL	20.00	0.31	4	694	No
19	23	1	3	136628	0	1	35000	DEBTCONSOLIDATION	18.25	0.26	4	709	No
20	24	2	1	14283	1	3	1750	EDUCATION	10.99	0.12	2	679	No
21	25	1	3	195718	0	1	35000	VENTURE	7.49	0.18	4	684	Yes
22	25	1	2	165792	4	1	34800	PERSONAL	16.77	0.21	2	662	No
23	22	2	1	79255	0	1	34000	EDUCATION	17.58	0.43	4	691	No
24	24	2	3	13866	0	2	1500	PERSONAL	7.29	0.11	3	600	Yes
25	22	1	3	97420	1	1	33950	PERSONAL	14.54	0.35	4	691	No

Description: At first we found the unique values using unique() then we replaces all instances of "rentt" with "rent" and “oown” with “own” by using `ifelse()`. At last we have converted categorical attributes to numeric with the help of factor function.

Column: loan_amnt

- **The ‘loan_amnt’ column exhibits optimal data quality with no missing or invalid values.**

```
missing_values_loan_amnt <- sum(is.na(mydata$loan_amnt))

print(missing_values_loan_amnt)
```

Output:

```
> missing_values_loan_amnt <- sum(is.na(mydata$loan_amnt))
> print(missing_values_loan_amnt)
[1] 0
>
```

Column: loan_intent

- **Detecting and Recovering Noisy Values: There is no Noisy values**

```
unique_loan_intent <- unique(mydata$loan_intent)

print(unique_loan_intent)
```

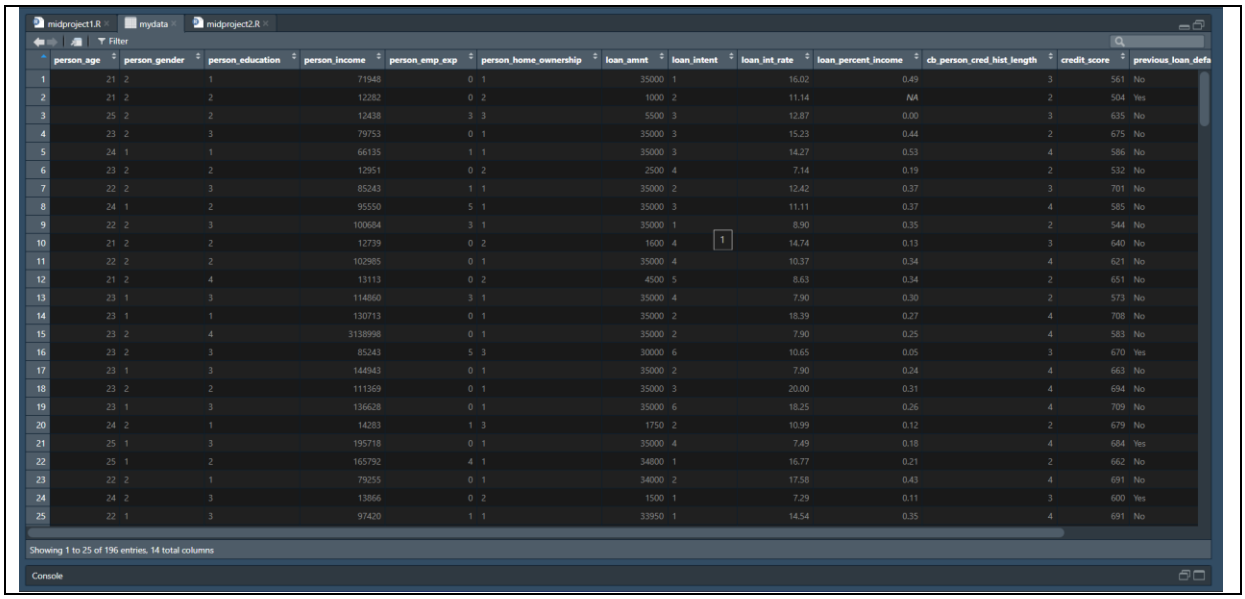
Output:

```
> unique_loan_intent <- unique(mydata$loan_intent)
> print(unique_loan_intent)
[1] "PERSONAL"          "EDUCATION"         "MEDICAL"           "VENTURE"           "HOMEIMPROVEMENT"   "DEBTCONSOLIDATION"
>
```

- **Data conversion: Converting categorical attributes to numeric**

```
mydata$loan_intent <- tolower(mydata$loan_intent)

mydata$loan_intent <- factor(mydata$loan_intent,

                                        levels = c("personal", "education",
"medical", "venture", "homeimprovement", "debtconsolidation"),

                                        labels = c(1, 2, 3, 4, 5, 6))

View(mydata)
```

Output:



	person_age	person_gender	person_education	person_income	person_emp_exp	person_home_ownership	loan_amnt	loan_intent	loan_int_rate	loan_percent_income	cb_person_cred_hist_length	credit_score	previous_loan_defa
1	21	2	1	71948	0	1	35000	1	16.02	0.49	3	561	No
2	21	2	2	12282	0	2	1000	2	11.14	NA	2	504	Yes
3	25	2	2	12438	3	3	5500	3	12.87	0.00	3	635	No
4	23	2	3	79753	0	1	35000	3	15.23	0.44	2	675	No
5	24	1	1	66135	1	1	35000	3	14.27	0.53	4	586	No
6	23	2	2	12951	0	2	2500	4	7.14	0.19	2	532	No
7	22	2	3	85243	1	1	35000	2	12.42	0.37	3	701	No
8	24	1	2	95550	5	1	35000	3	11.11	0.37	4	585	No
9	22	2	3	100684	3	1	35000	1	8.90	0.35	2	544	No
10	21	2	2	12739	0	2	1600	4	14.74	0.13	3	640	No
11	22	2	2	102985	0	1	35000	4	10.37	0.34	4	621	No
12	21	2	4	13113	0	2	4500	5	8.63	0.34	2	651	No
13	23	1	3	114860	3	1	35000	4	7.90	0.30	2	573	No
14	23	1	1	130713	0	1	35000	2	18.39	0.27	4	708	No
15	23	2	4	3138998	0	1	35000	2	7.90	0.25	4	583	No
16	23	2	3	85243	5	3	30000	6	10.65	0.05	3	670	Yes
17	23	1	3	144943	0	1	35000	2	7.90	0.24	4	663	No
18	23	2	2	111369	0	1	35000	3	20.00	0.31	4	694	No
19	23	1	3	136628	0	1	35000	6	18.25	0.26	4	709	No
20	24	2	1	14283	1	3	1750	2	10.99	0.12	2	679	No
21	25	1	3	195718	0	1	35000	4	7.49	0.18	4	684	Yes
22	25	1	2	165792	4	1	34800	1	16.77	0.21	2	662	No
23	22	2	1	79255	0	1	34000	2	17.58	0.43	4	691	No
24	24	2	3	13866	0	2	1500	1	7.29	0.11	3	600	Yes
25	22	1	3	97420	1	1	33950	1	14.54	0.35	4	691	No

Description: At first we found the unique values using unique() then we have converted categorical attributes to numeric with the help of factor function.

Column: loan_int_rate

- **The 'loan_int_rate' column exhibits optimal data quality with no missing or invalid values.**

```
missing_values_loan_amnt <- sum(is.na(mydata$loan_amnt))

print(missing_values_loan_amnt)
```

Output:

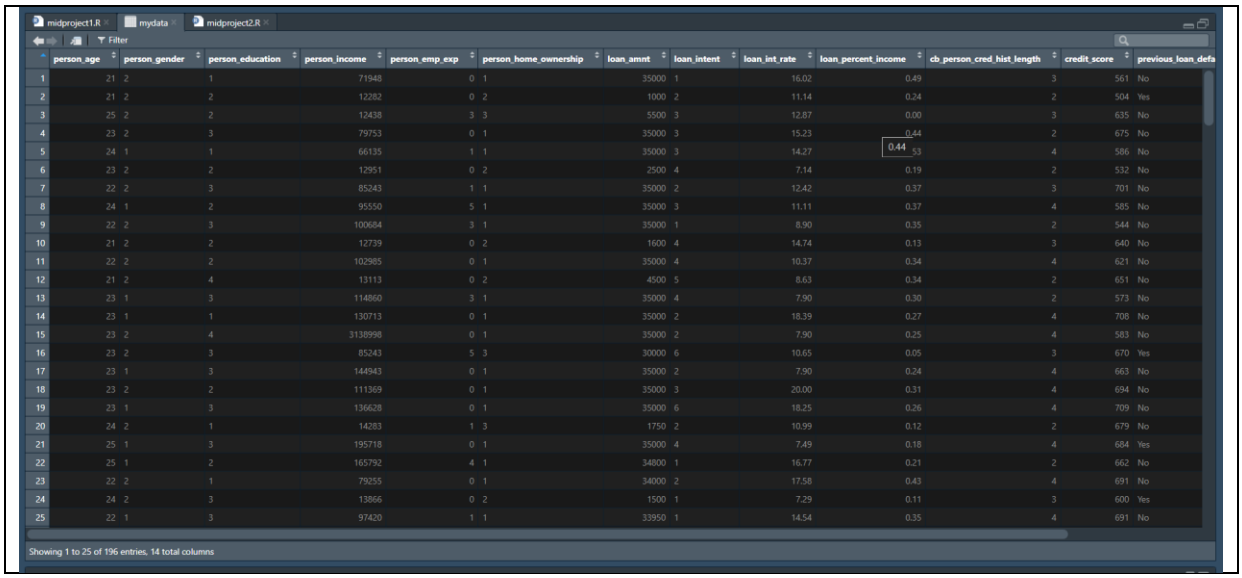
```
> missing_values_loan_int_rate <- sum(is.na(mydata$loan_int_rate))
> print(missing_values_loan_int_rate)
[1] 0
>
```

Column: loan_percent_income

- **Missing Value Imputation: Replacing NA Values with Median**

```
loan_percent_income_median <- median(mydata$loan_percent_income, na.rm =
TRUE)

mydata$loan_percent_income[is.na(mydata$loan_percent_income)] <-
loan_percent_income_median


View(mydata)
```

Output:



	person_age	person_gender	person_education	person_income	person_emp_exp	person_home_ownership	loan_amnt	loan_intent	loan_int_rate	loan_percent_income	cb_person_cred_hist_length	credit_score	previous_loan_defa
1	21	2	1	71948	0	1	35000	1	16.02	0.49	3	561	No
2	21	2	2	12282	0	2	1000	2	11.14	0.24	2	504	Yes
3	25	2	2	12438	3	3	5500	3	12.87	0.00	3	635	No
4	23	2	3	79753	0	1	35000	3	15.23	0.44	2	675	No
5	24	1	1	66135	1	1	35000	3	14.27	0.44	4	586	No
6	23	2	2	12951	0	2	2500	4	7.14	0.19	2	532	No
7	22	2	3	85243	1	1	35000	2	12.42	0.37	3	701	No
8	24	1	2	95550	5	1	35000	3	11.11	0.37	4	585	No
9	22	2	3	100684	3	1	35000	1	8.90	0.35	2	544	No
10	21	2	2	12739	0	2	1600	4	14.74	0.13	3	640	No
11	22	2	2	102985	0	1	35000	4	10.37	0.34	4	621	No
12	21	2	4	13113	0	2	4500	5	8.63	0.34	2	651	No
13	23	1	3	114860	3	1	35000	4	7.90	0.30	2	573	No
14	23	1	1	130713	0	1	35000	2	18.39	0.27	4	708	No
15	23	2	4	3138998	0	1	35000	2	7.90	0.25	4	583	No
16	23	2	3	85243	5	3	30000	6	10.65	0.05	3	670	Yes
17	23	1	3	144943	0	1	35000	2	7.90	0.24	4	663	No
18	23	2	2	111369	0	1	35000	3	20.00	0.31	4	694	No
19	23	1	3	136628	0	1	35000	6	18.25	0.26	4	709	No
20	24	2	1	14283	1	3	1750	2	10.99	0.12	2	679	No
21	25	1	3	195718	0	1	35000	4	7.49	0.18	4	684	Yes
22	25	1	2	165792	4	1	34800	1	16.77	0.21	2	662	No
23	22	2	1	79255	0	1	34000	2	17.58	0.43	4	691	No
24	24	2	3	13866	0	2	1500	1	7.29	0.11	3	600	Yes
25	22	1	3	97420	1	1	33950	1	14.54	0.35	4	691	No

Showing 1 to 25 of 196 entries, 14 total columns

Description: Using median() value for replacing NA.

Column: cb_person_cred_hist_length

- **The 'cb_person_cred_hist_length' column exhibits optimal data quality with no missing or invalid values.**

```
missing_values_cb_person_cred_hist_length <- sum(is.na(mydata$cb_person_cred_hist_length))

print(missing_values_cb_person_cred_hist_length)
```

Output:

```
> missing_values_cb_person_cred_hist_length <- sum(is.na(mydata$cb_person_cred_hist_length))
> print(missing_values_cb_person_cred_hist_length)
[1] 0
>
```

Column: credit_score

- **The 'credit_score' column exhibits optimal data quality with no missing or invalid values.**

```
missing_values_credit_score <- sum(is.na(mydata$credit_score))

print(missing_values_credit_score)
```

Output:

```
> missing_values_credit_score <- sum(is.na(mydata$credit_score))
> print(missing_values_credit_score)
[1] 0
>
```

Column: previous_loan_defaults_on_file

- **Detecting and Recovering Noisy Values: There is no Noisy values**

```
unique_previous_loan_defaults_on_file <-
unique(mydata$previous_loan_defaults_on_file)

print(unique_previous_loan_defaults_on_file)
```

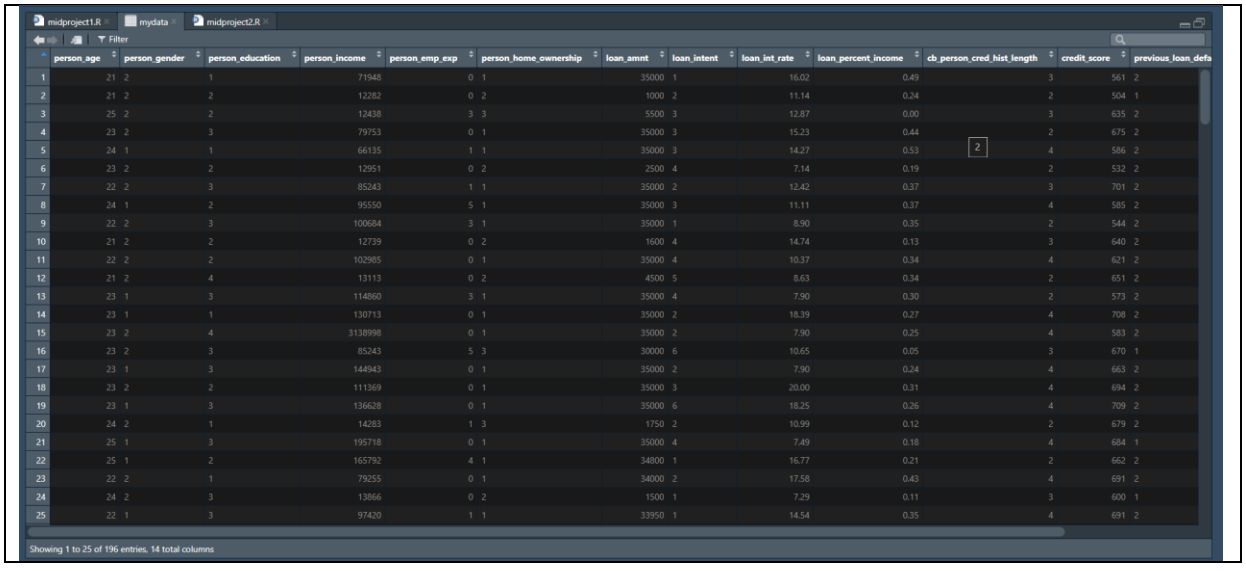
Output:

```
> unique_previous_loan_defaults_on_file <- unique(mydata$previous_loan_defaults_on_file)
> print(unique_previous_loan_defaults_on_file)
[1] "No"  "Yes"
>
```

- **Data conversion: Converting categorical attributes to numeric**

```
mydata$previous_loan_defaults_on_file <-
tolower(mydata$previous_loan_defaults_on_file)

mydata$previous_loan_defaults_on_file <-
factor(mydata$previous_loan_defaults_on_file,
                                levels = c("yes", "no"),
                                labels = c(1, 2))


View(mydata)
```

Output:



	person_age	person_gender	person_education	person_income	person_emp_exp	person_home_ownership	loan_amnt	loan_intent	loan_int_rate	loan_percent_income	cb_person_cred_hist_length	credit_score	previous_loan_defa
1	21	2	1	71948	0	1	35000	1	16.02	0.49	3	561	2
2	21	2	2	12282	0	2	1000	2	11.14	0.24	2	504	1
3	25	2	2	12438	3	3	5500	3	12.87	0.00	3	635	2
4	23	2	3	79753	0	1	35000	3	15.23	0.44	2	675	2
5	24	1	1	66135	1	1	35000	3	14.27	0.53	4	586	2
6	23	2	2	12951	0	2	2500	4	7.14	0.19	2	532	2
7	22	2	3	85243	1	1	35000	2	12.42	0.37	3	701	2
8	24	1	2	95550	5	1	35000	3	11.11	0.37	4	585	2
9	22	2	3	100684	3	1	35000	1	8.90	0.35	2	544	2
10	21	2	2	12739	0	2	1600	4	14.74	0.13	3	640	2
11	22	2	2	102985	0	1	35000	4	10.37	0.34	4	621	2
12	21	2	4	13113	0	2	4500	5	8.63	0.34	2	651	2
13	23	1	3	114860	3	1	35000	4	7.90	0.30	2	573	2
14	23	1	1	130713	0	1	35000	2	18.39	0.27	4	708	2
15	23	2	4	3138998	0	1	35000	2	7.90	0.25	4	583	2
16	23	2	3	85243	5	3	30000	6	10.65	0.05	3	670	1
17	23	1	3	144943	0	1	35000	2	7.90	0.24	4	663	2
18	23	2	2	111369	0	1	35000	3	20.00	0.31	4	694	2
19	23	1	3	136628	0	1	35000	6	18.25	0.26	4	709	2
20	24	2	1	14283	1	3	1750	2	10.99	0.12	2	679	2
21	25	1	3	195718	0	1	35000	4	7.49	0.18	4	684	1
22	25	1	2	165792	4	1	34800	1	16.77	0.21	2	662	2
23	22	2	1	79255	0	1	34000	2	17.58	0.43	4	691	2
24	24	2	3	13866	0	2	1500	1	7.29	0.11	3	600	1
25	22	1	3	97420	1	1	33950	1	14.54	0.35	4	691	2

Description: At first we found the unique values using unique() then we have converted categorical attributes to numeric with the help of factor function.

Column: loan_status

- **Checking missing value.**

```
missing_values_loan_status <- sum(is.na(mydata$loan_status))

print(missing_values_loan_status)
```

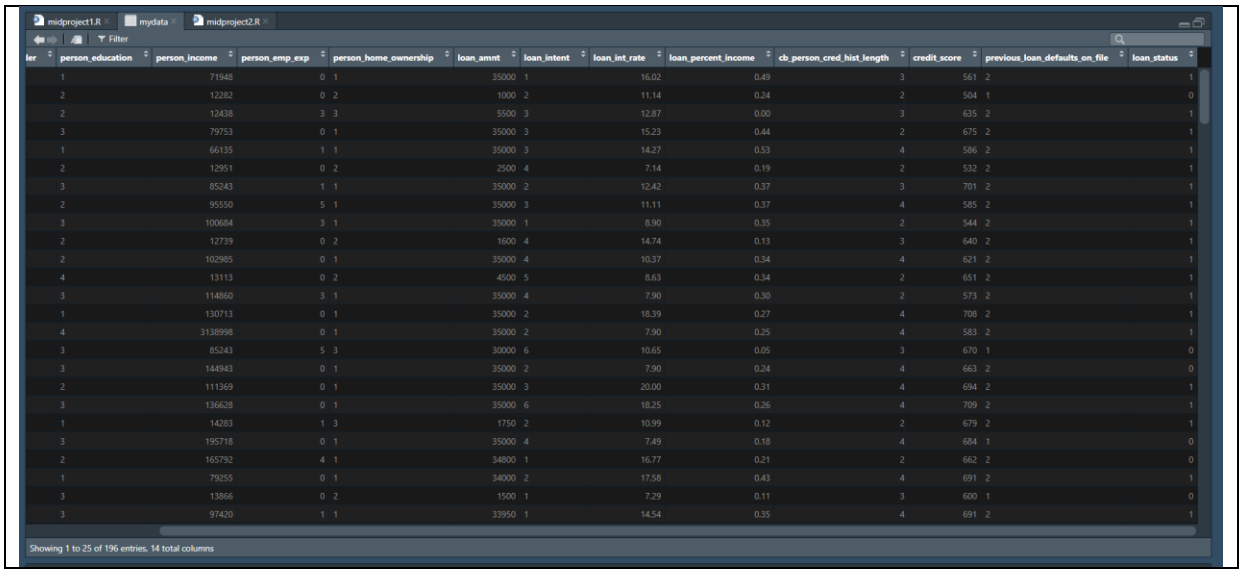
Output:

```
> missing_values_loan_status <- sum(is.na(mydata$loan_status))
> print(missing_values_loan_status)
[1] 3
>
```

- **Missing Value Imputation: Replacing NA Values with Mode**

```
mode_loan_status <- as.numeric(names(sort(table(mydata$loan_status),
decreasing = TRUE)[1]))

mydata$loan_status[is.na(mydata$loan_status)] <- mode_loan_status


View(mydata)
```

Output:



Description: At first we found the missing values quantity by using sum() & is.na() then we replace the missing values by mode.

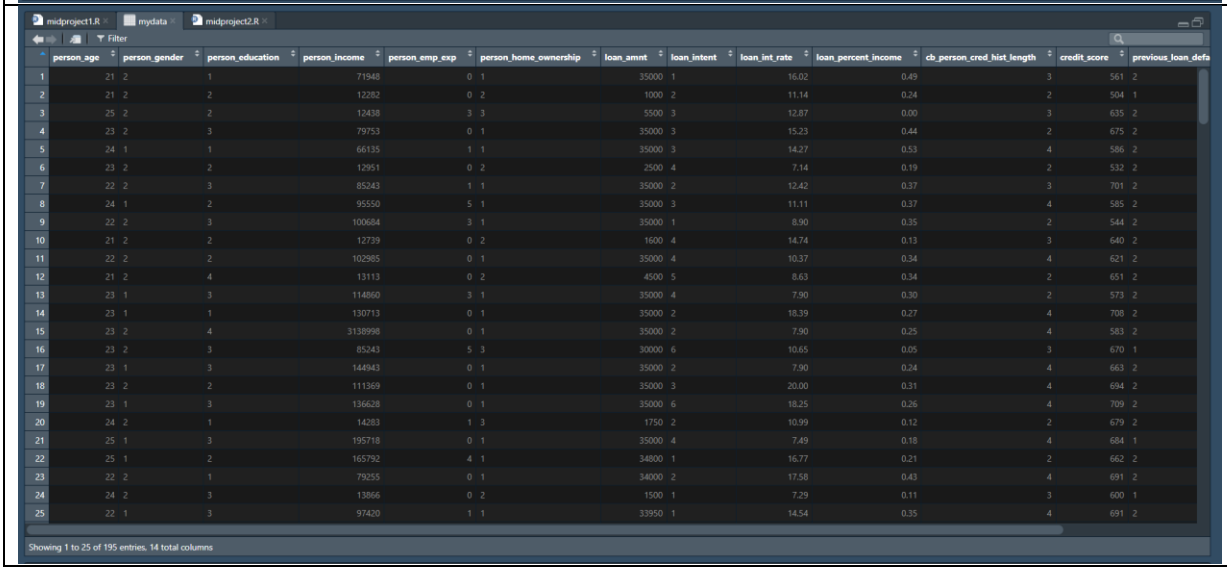
Remove duplicate rows:

Code:

```
duplicate_rows <- mydata[duplicated(mydata), ]
print(paste("Number of duplicate rows:", nrow(duplicate_rows)))
mydata <- mydata[!duplicated(mydata), ]
View(mydata)
```

Output:

```
> duplicate_rows <- mydata[duplicated(mydata), ]
> print(paste("Number of duplicate rows:", nrow(duplicate_rows)))
[1] "Number of duplicate rows: 1"
```



	person_age	person_gender	person_education	person_income	person_emp_exp	person_home_ownership	loan_amnt	loan_intent	loan_int_rate	loan_percent_income	cb_person_cred_hist_length	credit_score	previous_loan_defa
1	21	2	1	71948	0	1	35000	1	16.02	0.49	3	561	2
2	21	2	2	12282	0	2	1000	2	11.14	0.24	2	504	1
3	25	2	2	12438	3	3	5500	3	12.87	0.00	3	635	2
4	23	2	3	79753	0	1	35000	3	15.23	0.44	2	675	2
5	24	1	1	66135	1	1	35000	3	14.27	0.53	4	586	2
6	23	2	2	12951	0	2	2500	4	7.14	0.19	2	532	2
7	22	2	3	85243	1	1	35000	2	12.42	0.37	3	701	2
8	24	1	2	95550	5	1	35000	3	11.11	0.37	4	585	2
9	22	2	3	100684	3	1	35000	1	8.90	0.35	2	544	2
10	21	2	2	12739	0	2	1600	4	14.74	0.13	3	640	2
11	22	2	2	102985	0	1	35000	4	10.37	0.34	4	621	2
12	21	2	4	13113	0	2	4500	5	8.63	0.34	2	651	2
13	23	1	3	114860	3	1	35000	4	7.90	0.30	2	573	2
14	23	1	1	130713	0	1	35000	2	18.39	0.27	4	708	2
15	23	2	4	3138998	0	1	35000	2	7.90	0.25	4	583	2
16	23	2	3	85243	5	3	30000	6	10.65	0.05	3	670	1
17	23	1	3	144943	0	1	35000	2	7.90	0.24	4	663	2
18	23	2	2	111369	0	1	35000	3	20.00	0.31	4	694	2
19	23	1	3	136628	0	1	35000	6	18.25	0.26	4	709	2
20	24	2	1	14283	1	3	1750	2	10.99	0.12	2	679	2
21	25	1	3	195718	0	1	35000	4	7.49	0.18	4	684	1
22	25	1	2	165792	4	1	34800	1	16.77	0.21	2	662	2
23	22	2	1	79255	0	1	34000	2	17.58	0.43	4	691	2
24	24	2	3	13866	0	2	1500	1	7.29	0.11	3	600	1
25	22	1	3	97420	1	1	33950	1	14.54	0.35	4	691	2

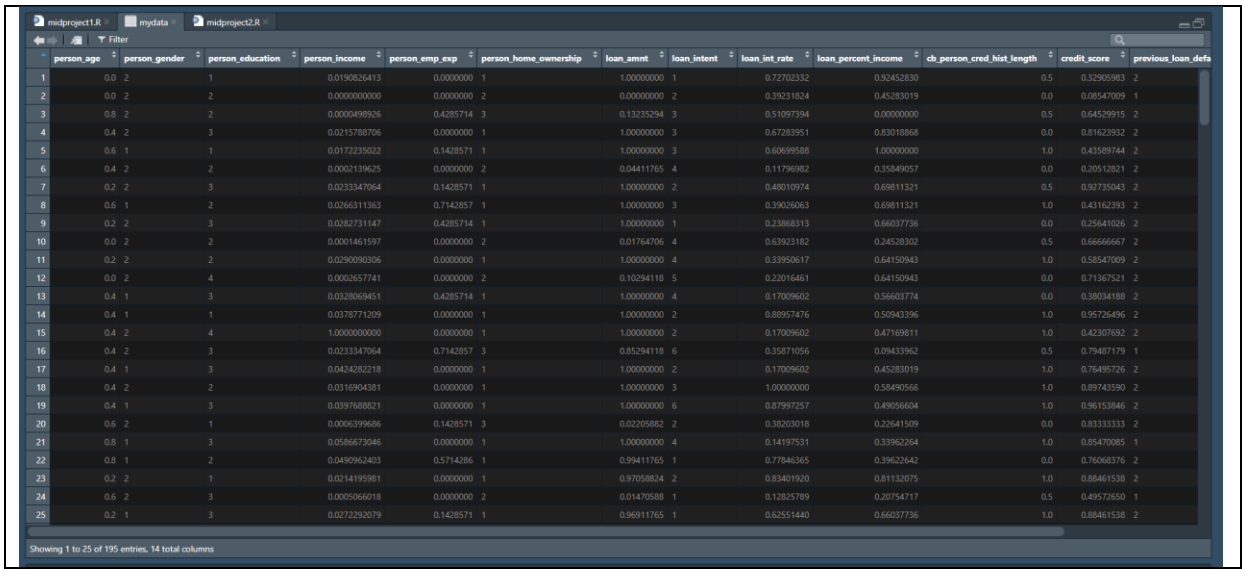
Description: Finding duplicated rows by using duplicated() function and drop those rows.

Apply min max for all numeric column:

Code:

```
 numeric_columns <- sapply(mydata, is.numeric)
mydata[numeric_columns] <- lapply(mydata[numeric_columns], function(x) {
  (x - min(x, na.rm = TRUE)) / (max(x, na.rm = TRUE) - min(x, na.rm = TRUE))
})
```

Output:



midproject1.R | mydata | midproject2.R

Filter

	person_age	person_gender	person_education	person_income	person_emp_exp	person_home_ownership	loan_amnt	loan_intent	loan_int_rate	loan_percent_income	cb_person_cred_hist_length	credit_score	previous_loan_defa
1	0.0	2	1	0.0190826413	0.0000000	1	1.00000000	1	0.72702332	0.92452830	0.5	0.32905983	2
2	0.0	2	2	0.0000000000	0.0000000	2	0.00000000	2	0.39231824	0.45283019	0.0	0.08547009	1
3	0.8	2	2	0.0000498926	0.4285714	3	0.13235294	3	0.51097394	0.00000000	0.5	0.64529915	2
4	0.4	2	3	0.0215788706	0.0000000	1	1.00000000	3	0.67283951	0.83018868	0.0	0.81623932	2
5	0.6	1	1	0.0172235022	0.1428571	1	1.00000000	3	0.60699588	1.00000000	1.0	0.43589744	2
6	0.4	2	2	0.0002139625	0.0000000	2	0.04411765	4	0.11796982	0.35849057	0.0	0.20512821	2
7	0.2	2	3	0.0233347064	0.1428571	1	1.00000000	2	0.48010974	0.69811321	0.5	0.92735043	2
8	0.6	1	2	0.0266311363	0.7142857	1	1.00000000	3	0.39026063	0.69811321	1.0	0.43162393	2
9	0.2	2	3	0.0282731147	0.4285714	1	1.00000000	1	0.23868313	0.66037736	0.0	0.25641026	2
10	0.0	2	2	0.0001461597	0.0000000	2	0.01764706	4	0.63923182	0.24528302	0.5	0.66666667	2
11	0.2	2	2	0.0290090306	0.0000000	1	1.00000000	4	0.33950617	0.64150943	1.0	0.58547009	2
12	0.0	2	4	0.0002657741	0.0000000	2	0.10294118	5	0.22016461	0.64150943	0.0	0.71367521	2
13	0.4	1	3	0.0328069451	0.4285714	1	1.00000000	4	0.17009602	0.56603774	0.0	0.38034188	2
14	0.4	1	1	0.0378771209	0.0000000	1	1.00000000	2	0.88957476	0.50943396	1.0	0.95726496	2
15	0.4	2	4	1.0000000000	0.0000000	1	1.00000000	2	0.17009602	0.47169811	1.0	0.42307692	2
16	0.4	2	3	0.0233347064	0.7142857	3	0.85294118	6	0.35871056	0.09433962	0.5	0.79487179	1
17	0.4	1	3	0.0424282218	0.0000000	1	1.00000000	2	0.17009602	0.45283019	1.0	0.76495726	2
18	0.4	2	2	0.0316904381	0.0000000	1	1.00000000	3	1.00000000	0.58490566	1.0	0.89743590	2
19	0.4	1	3	0.0397688821	0.0000000	1	1.00000000	6	0.87997257	0.49056604	1.0	0.96153846	2
20	0.6	2	1	0.0006399686	0.1428571	3	0.02205882	2	0.38203018	0.22641509	0.0	0.83333333	2
21	0.8	1	3	0.0586673046	0.0000000	1	1.00000000	4	0.14197531	0.33962264	1.0	0.85470085	1
22	0.8	1	2	0.0490962403	0.5714286	1	0.99411765	1	0.77846365	0.39622642	0.0	0.76068376	2
23	0.2	2	1	0.0214195981	0.0000000	1	0.97058824	2	0.83401920	0.81132075	1.0	0.88461538	2
24	0.6	2	3	0.0005066018	0.0000000	2	0.01470588	1	0.12825789	0.20754717	0.5	0.49572650	1
25	0.2	1	3	0.0272292079	0.1428571	1	0.96911765	1	0.62551440	0.66037736	1.0	0.88461538	2

Showing 1 to 25 of 195 entries, 14 total columns

Description: Finding duplicated rows by using duplicated() function and drop those rows.

Date Visualization

- **Find Missing Values:**

Code:

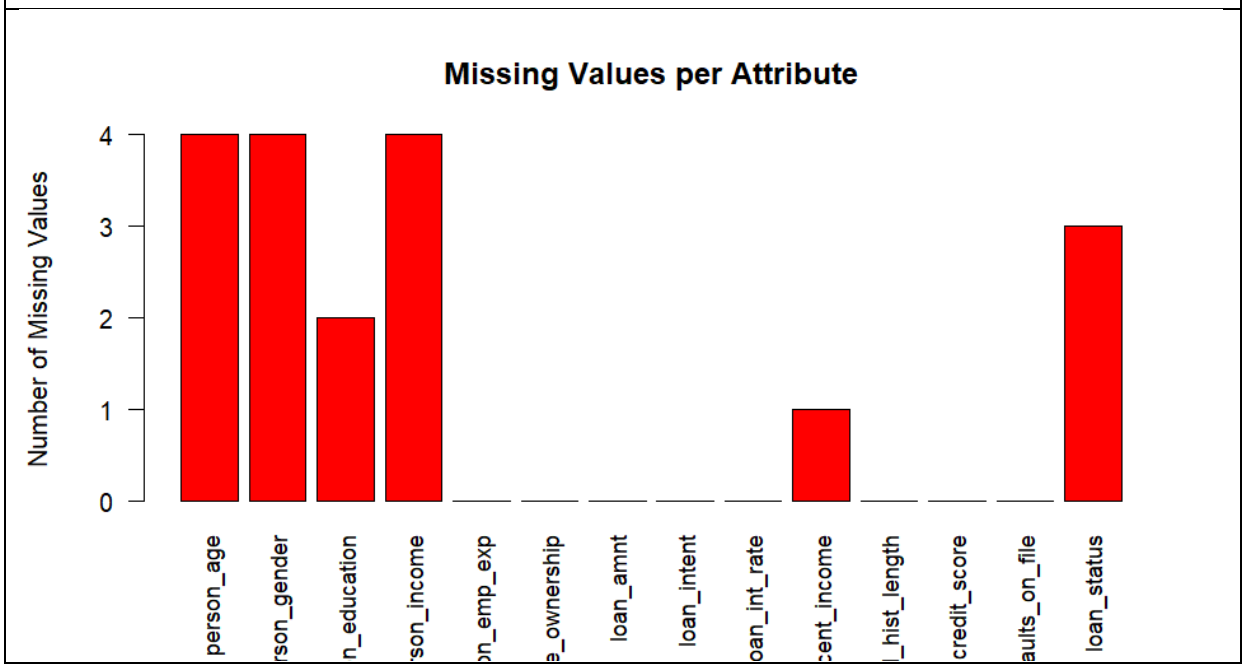
```
na_counts <- colSums(is.na(mydata))
print(na_counts)
```

Visualization:

```
barplot(na_counts, names.arg = names(na_counts),
        ylab = "Number of Missing Values", col = "red", cex.names = 0.9,
        main = "Missing Values per Attribute", las = 2)
```

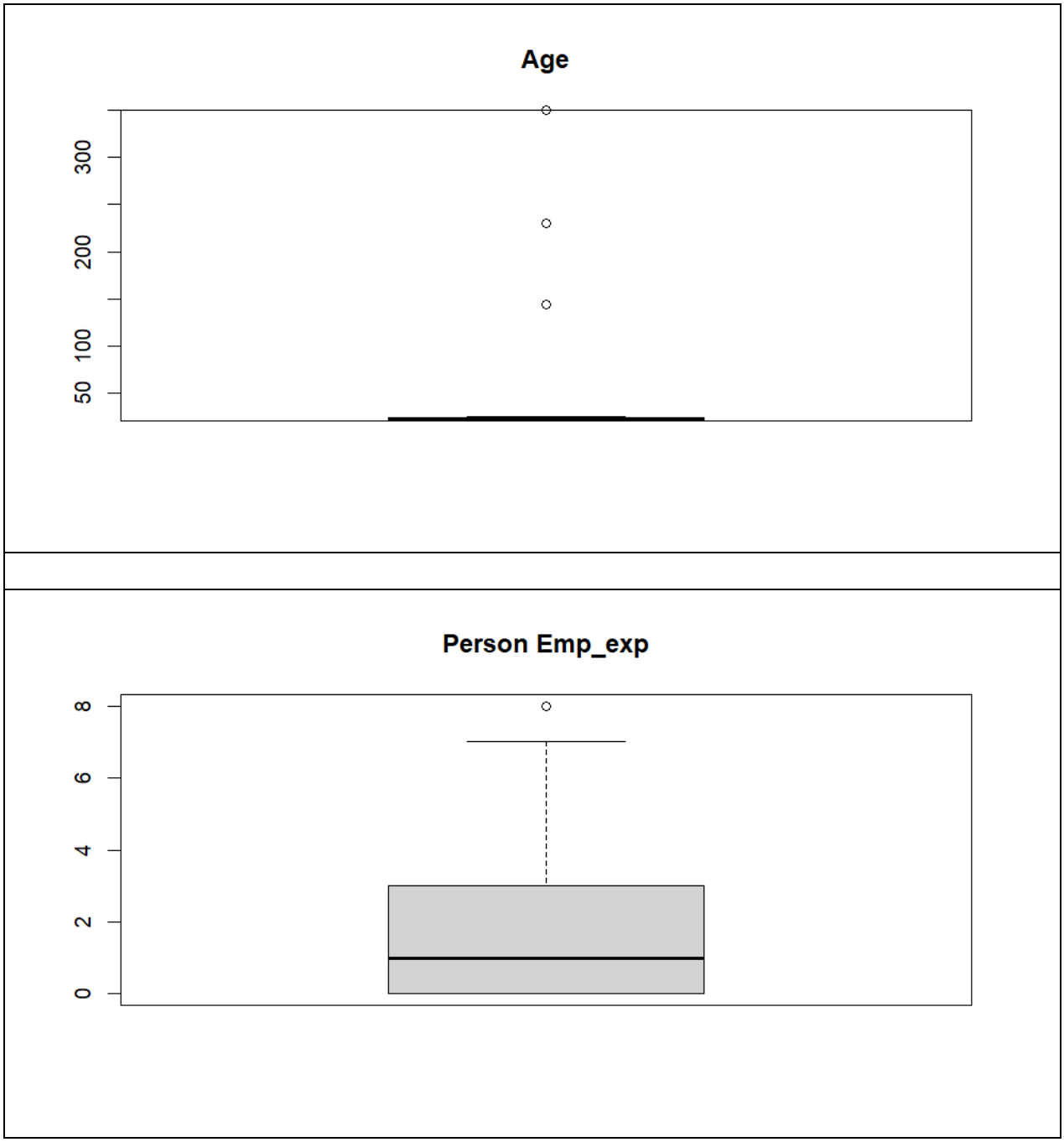
Output:

```
> na_counts <- colSums(is.na(mydata))
> print(na_counts)
                    person_age                   person_gender                person_education
                             4                               4                               2
                 person_income                  person_emp_exp            person_home_ownership
                             4                               0                               0
                     loan_amnt                     loan_intent                   loan_int_rate
                             0                               0                               0
            loan_percent_income       cb_person_cred_hist_length                    credit_score
                             1                               0                               0
previous_loan_defaults_on_file                     loan_status
                             0                               3
```



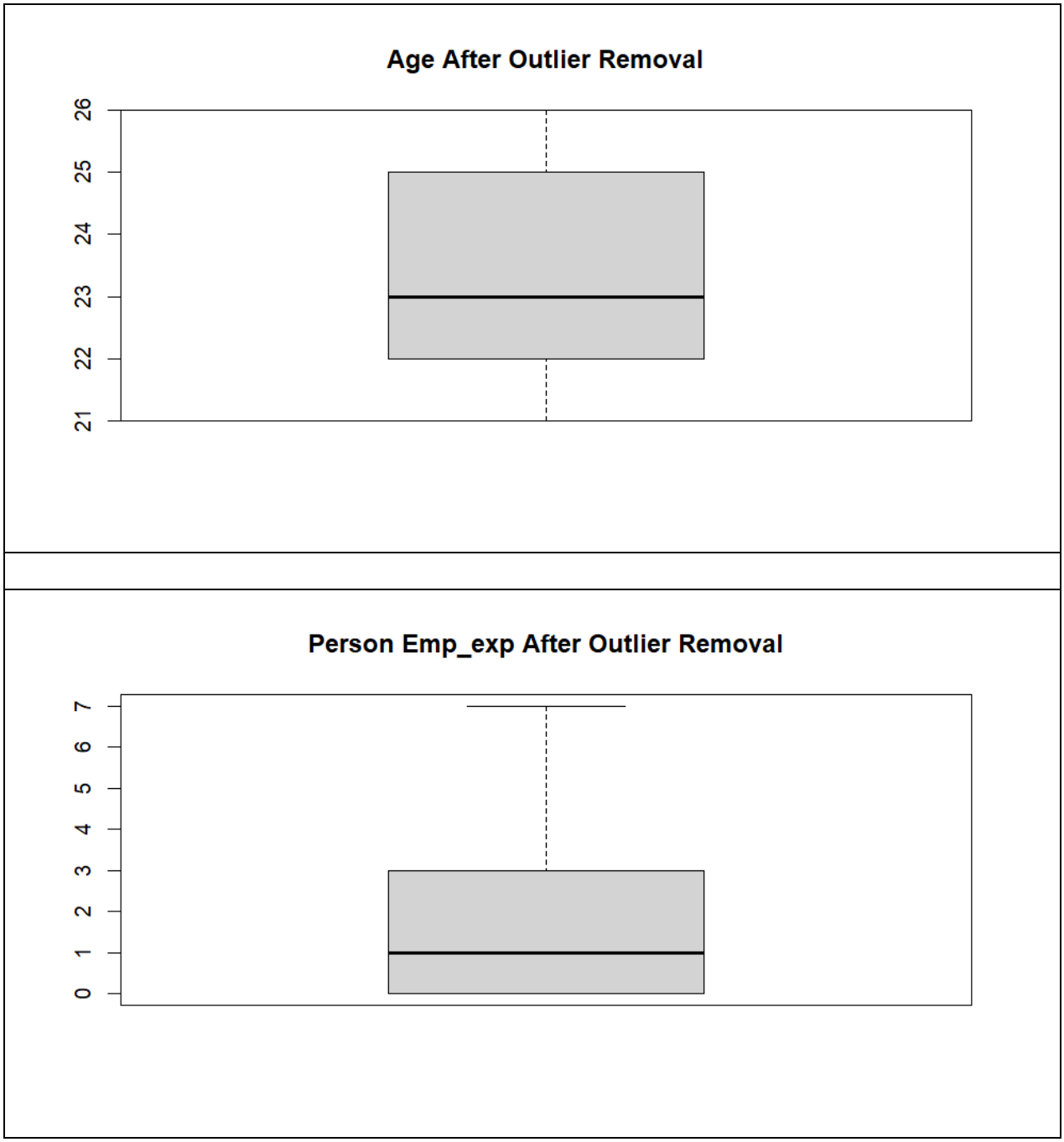
Description: Using colSums() and is.na() we found the number of missing values for each attribute.

- **Find Outliers:**



Description: Using boxplot, we found outliers on Age and Person_emp_exp we no need to find other attributes outliers.

After removing outliers:



- **Summary:**

```
> summary(mydata)
   person_age     person_gender person_education person_income     person_emp_exp
 Min.   :0.0000   1:118         1:23             Min.   :0.00000   Min.   :0.0000
 1st Qu.:0.2000   2: 77         2:58             1st Qu.:0.01550   1st Qu.:0.0000
 Median :0.4000                 3:69             Median :0.02333   Median :0.1429
 Mean   :0.4892                 4:44             Mean   :0.04368   Mean   :0.2183
 3rd Qu.:0.8000                 5: 1             3rd Qu.:0.07316   3rd Qu.:0.4286
 Max.   :1.0000                                  Max.   :1.00000   Max.   :1.0000
 person_home_ownership   loan_amnt      loan_intent loan_int_rate
 1:183                 Min.   :0.0000   1:30        Min.   :0.0000
 2:  7                 1st Qu.:0.3221   2:54        1st Qu.:0.3587
 3:  4                 Median :0.7059   3:26        Median :0.4396
 4:  1                 Mean   :0.5776   4:31        Mean   :0.4732
                       3rd Qu.:0.7941   5:23        3rd Qu.:0.6214
                       Max.   :1.0000   6:31        Max.   :1.0000
 loan_percent_income cb_person_cred_hist_length  credit_score
 Min.   :0.0000      Min.   :0.0000             Min.   :0.0000
 1st Qu.:0.1698      1st Qu.:0.0000             1st Qu.:0.4722
 Median :0.4528      Median :0.5000             Median :0.6239
 Mean   :0.4316      Mean   :0.5026             Mean   :0.6109
 3rd Qu.:0.6415      3rd Qu.:1.0000             3rd Qu.:0.7671
 Max.   :1.0000      Max.   :1.0000             Max.   :1.0000
 previous_loan_defaults_on_file  loan_status
 1: 50                          Min.   :0.0000
 2:145                          1st Qu.:0.0000
                                Median :1.0000
                                Mean   :0.6256
                                3rd Qu.:1.0000
                                Max.   :1.0000
> |
```

- **Mean-Median-Mode:**

Code:

```
 descriptive_stats <- function(column) {

 mean_value <- mean(column, na.rm = TRUE)

 median_value <- median(column, na.rm = TRUE)

 mode_value <- as.numeric(names(sort(table(column), decreasing =
TRUE)[1]))

  return(c(Mean = mean_value, Median = median_value, Mode = mode_value))

}


descriptive_summary <- lapply(mydata[numeric_columns], descriptive_stats)

descriptive_summary <- do.call(rbind, descriptive_summary)


print("Descriptive Statistics for Numeric Columns:")

print(descriptive_summary)
```

Vizualization:

```
> print("Descriptive Statistics for Numeric Columns:")
[1] "Descriptive Statistics for Numeric Columns:"
> print(descriptive_summary)
                               Mean     Median       Mode
person_age                0.48923077 0.40000000 0.20000000
person_income             0.04367904 0.02333471 0.02333471
person_emp_exp            0.21831502 0.14285714 0.00000000
loan_amnt                 0.57763198 0.70588235 0.70588235
loan_int_rate             0.47319827 0.43964335 0.38340192
loan_percent_income       0.43164006 0.45283019 0.64150943
cb_person_cred_hist_length 0.50256410 0.50000000 0.50000000
credit_score              0.61089196 0.62393162 0.64102564
loan_status               0.62564103 1.00000000 1.00000000
> 
```

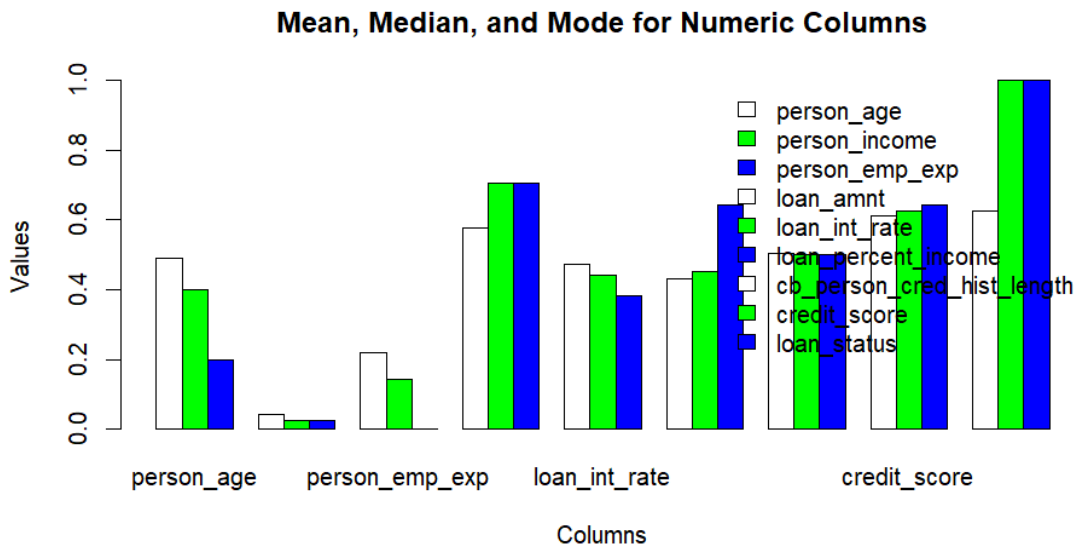
Description: A function, descriptive_stats, is defined to calculate the mean, median, and mode of a given numeric column, with missing values handled using na.rm = TRUE. The mean(), median(), and table() functions are used to compute these statistics, and the results are returned as a named vector. The function is applied to all numeric columns of a dataset (mydata) using lapply(), and the outputs are combined into a single data frame using do.call(rbind, ...). Finally, the descriptive statistics for the numeric columns are displayed.

- ## Mean-Median-Mode Graph:

Code:

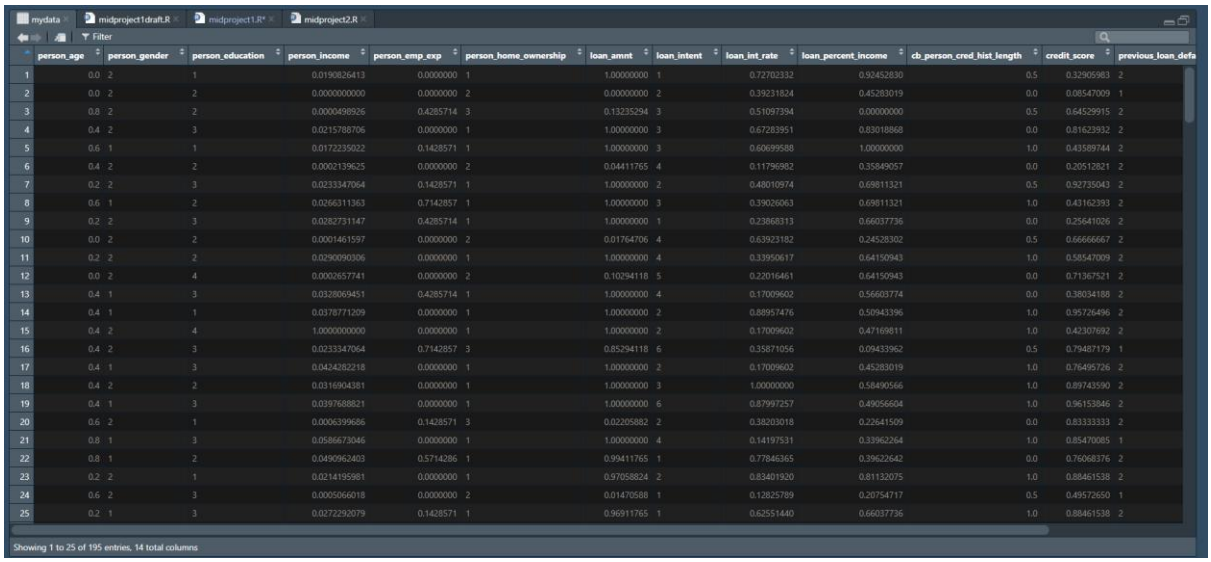
```
 barplot(t(descriptive_summary), beside = TRUE, col = c("white", "green", "blue"),
        legend.text = rownames(descriptive_summary), args.legend = list(x = "topright", bty = "n"),
        main = "Mean, Median, and Mode for Numeric Columns", ylab = "Values", xlab = "Columns")
```

Vizualization:



Description: Barplot has been drawn by using barplot() function to visualize.

Final Data Set:



	person_age	person_gender	person_education	person_income	person_emp_exp	person_home_ownership	loan_amnt	loan_intent	loan_int_rate	loan_percent_income	cb_person_cred_hist_length	credit_score	previous_loan_defa
1	0.0	2	1	0.0190826413	0.0000000	1	1.00000000	1	0.72702332	0.92452830	0.5	0.32905983	2
2	0.0	2	2	0.0000000000	0.0000000	2	0.00000000	2	0.39231824	0.45283019	0.0	0.08547009	1
3	0.8	2	2	0.0000498926	0.4285714	3	0.13235294	3	0.51097394	0.00000000	0.5	0.64529915	2
4	0.4	2	3	0.0215788706	0.0000000	1	1.00000000	3	0.67283951	0.83018868	0.0	0.81623932	2
5	0.6	1	1	0.0172235022	0.1428571	1	1.00000000	3	0.60699588	1.00000000	1.0	0.43589744	2
6	0.4	2	2	0.0002139625	0.0000000	2	0.04411765	4	0.11796982	0.35849057	0.0	0.20512821	2
7	0.2	2	3	0.0233347064	0.1428571	1	1.00000000	2	0.48010974	0.69811321	0.5	0.92735043	2
8	0.6	1	2	0.0266311363	0.7142857	1	1.00000000	3	0.39026063	0.69811321	1.0	0.43162393	2
9	0.2	2	3	0.0282731147	0.4285714	1	1.00000000	1	0.23868313	0.66037736	0.0	0.25641026	2
10	0.0	2	2	0.0001461597	0.0000000	2	0.01764706	4	0.63923182	0.24528302	0.5	0.66666667	2
11	0.2	2	2	0.0290090306	0.0000000	1	1.00000000	4	0.33950617	0.64150943	1.0	0.58547009	2
12	0.0	2	4	0.0002657741	0.0000000	2	0.10294118	5	0.22016461	0.64150943	0.0	0.71367521	2
13	0.4	1	3	0.0328069451	0.4285714	1	1.00000000	4	0.17009602	0.56603774	0.0	0.38034188	2
14	0.4	1	1	0.0378771209	0.0000000	1	1.00000000	2	0.88957476	0.50943396	1.0	0.95726496	2
15	0.4	2	4	1.0000000000	0.0000000	1	1.00000000	2	0.17009602	0.47169811	1.0	0.42307692	2
16	0.4	2	3	0.0233347064	0.7142857	3	0.85294118	6	0.35871056	0.09433962	0.5	0.79487179	1
17	0.4	1	3	0.0424282218	0.0000000	1	1.00000000	2	0.17009602	0.45283019	1.0	0.76495726	2
18	0.4	2	2	0.0316904381	0.0000000	1	1.00000000	3	1.00000000	0.58490566	1.0	0.89743590	2
19	0.4	1	3	0.0397688821	0.0000000	1	1.00000000	6	0.87997257	0.49056604	1.0	0.96153846	2
20	0.6	2	1	0.0006399686	0.1428571	3	0.02205882	2	0.38203018	0.22641509	0.0	0.83333333	2
21	0.8	1	3	0.0586673046	0.0000000	1	1.00000000	4	0.14197531	0.33962264	1.0	0.85470085	1
22	0.8	1	2	0.0490962403	0.5714286	1	0.99411765	1	0.77846365	0.39622642	0.0	0.76068376	2
23	0.2	2	1	0.0214195981	0.0000000	1	0.97058824	2	0.83401920	0.81132075	1.0	0.88461538	2
24	0.6	2	3	0.0005066018	0.0000000	2	0.01470588	1	0.12825789	0.20754717	0.5	0.49572650	1
25	0.2	1	3	0.0272292079	0.1428571	1	0.96911765	1	0.62551440	0.66037736	1.0	0.88461538	2

This is the outcome of the data set after cleaning all the data.

THE END
