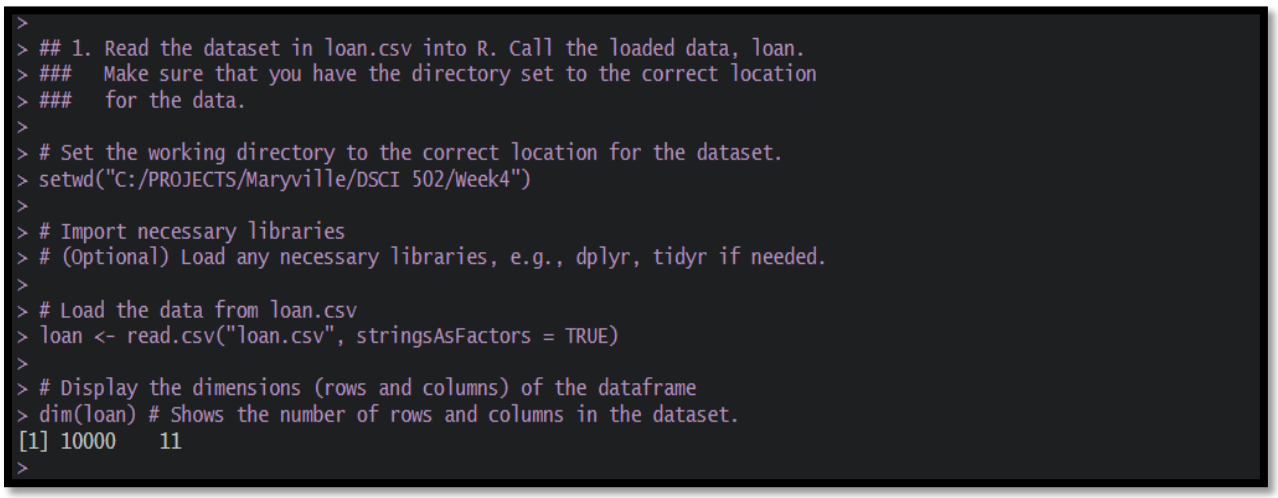
Student: **Seif Kungulio**
Date: **02/09/2025**
Subject: **Project 4**
Class: **DSCI 502**
Section: **01W**
Instructor: **Sean Yang**
File Name: **Project4_Kungulio_Seif.docx**

---

1. Read the dataset in loan.csv into R. Call the loaded data, loan. Make sure that you have the directory set to the correct location for the data.

```
>
> ## 1. Read the dataset in loan.csv into R. Call the loaded data, loan.
> ###   Make sure that you have the directory set to the correct location
> ###   for the data.
>
> # Set the working directory to the correct location for the dataset.
> setwd("C:/PROJECTS/Maryville/DSCI 502/Week4")
>
> # Import necessary libraries
> # (Optional) Load any necessary libraries, e.g., dplyr, tidyr if needed.
>
> # Load the data from loan.csv
> loan <- read.csv("loan.csv", stringsAsFactors = TRUE)
>
> # Display the dimensions (rows and columns) of the dataframe
> dim(loan) # Shows the number of rows and columns in the dataset.
[1] 10000    11
>
```



2. Which variables (columns) are continuous/numerical variables? Which columns are factors (categorical variables)?

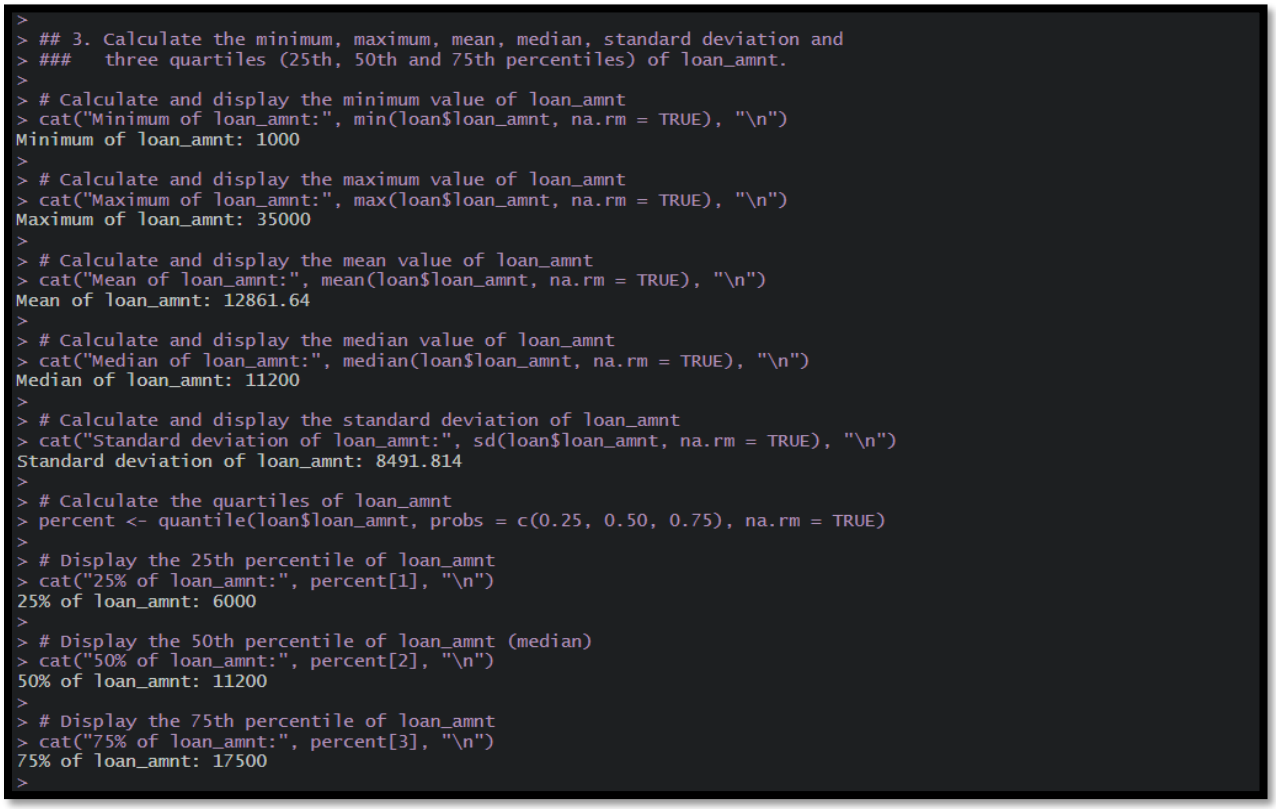
```
>
> ## 2. Which variables (columns) are continuous/numerical variables? Which
> ###   columns are factors (categorical variables)?
>
> # Identify variable types
> str(loan) # Displays the structure of the dataset including variable types.
'data.frame':   10000 obs. of  11 variables:
 $ id              : int  1077501 1077430 1077175 1076863 1075358 1075269 1069639
1072053 1071795 1071570 ...
 $ loan_amnt       : int  5000 2500 2400 10000 3000 5000 7000 3000 5600 5375 ...
 $ term            : Factor w/ 2 levels " 36 months"," 60 months": 1 2 1 1 2 1 2 1 2 2 ...
 $ int_rate        : num  10.6 15.3 16 13.5 12.7 ...
 $ installment     : num  162.9 59.8 84.3 339.3 67.8 ...
 $ grade           : Factor w/ 7 levels "A","B","C","D",..: 2 3 3 3 2 1 3 5 6 2 ...
 $ emp_length      : Factor w/ 12 levels "< 1 year","1 year",..: 3 1 3 3 2 5 10 11 6 1 ...
 $ home_ownership  : Factor w/ 3 levels "MORTGAGE","OWN",..: 3 3 3 3 3 3 3 3 2 3 ...
 $ annual_inc      : num  24000 30000 12252 49200 80000 ...
 $ verification_status: Factor w/ 3 levels "Not Verified",..: 3 2 1 2 2 2 1 2 2 3 ...
 $ loan_status     : Factor w/ 7 levels "Charged Off",..: 4 1 4 4 2 4 2 4 1 1 ...
>
> # Identify continuous (numerical) and categorical (factor) variables
> # Checks which variables are numerical.
> numerical_vars <- sapply(loan, is.numeric)
>
> # Checks which variables are categorical.
> categorical_vars <- sapply(loan, is.factor)
>
> # Extract the names of numerical variables
> numerical_columns <- names(numerical_vars[numerical_vars])
> cat("Numerical Variables:\n", numerical_columns, "\n\n")
Numerical Variables:
 id loan_amnt int_rate installment annual_inc

>
> # Extract the names of categorical variables
> categorical_columns <- names(categorical_vars[categorical_vars])
> cat("Categorical Variables:\n", categorical_columns, "\n\n")
Categorical Variables:
 term grade emp_length home_ownership verification_status loan_status

>
```

```
>
> ## 2. Which variables (columns) are continuous/numerical variables? Which
> ###   columns are factors (categorical variables)?
>
> # Identify variable types
> str(loan) # Displays the structure of the dataset including variable types.
'data.frame':   10000 obs. of  11 variables:
 $ id                  : int  1077501 1077430 1077175 1076863 1075358 1075269 1069639 1072053 1071795 1071570 ...
 $ loan_amnt           : int  5000 2500 2400 10000 3000 5000 7000 3000 5600 5375 ...
 $ term                : Factor w/ 2 levels " 36 months"," 60 months": 1 2 1 1 2 1 2 1 2 2 ...
 $ int_rate            : num  10.6 15.3 16 13.5 12.7 ...
 $ installment         : num  162.9 59.8 84.3 339.3 67.8 ...
 $ grade               : Factor w/ 7 levels "A","B","C","D",..: 2 3 3 3 2 1 3 5 6 2 ...
 $ emp_length          : Factor w/ 12 levels "< 1 year","1 year",..: 3 1 3 3 2 5 10 11 6 1 ...
 $ home_ownership      : Factor w/ 3 levels "MORTGAGE","OWN",..: 3 3 3 3 3 3 3 3 2 3 ...
 $ annual_inc          : num  24000 30000 12252 49200 80000 ...
 $ verification_status: Factor w/ 3 levels "Not Verified",..: 3 2 1 2 2 2 1 2 2 3 ...
 $ loan_status         : Factor w/ 7 levels "Charged Off",..: 4 1 4 4 2 4 2 4 1 1 ...
>
> # Identify continuous (numerical) and categorical (factor) variables
> # Checks which variables are numerical.
> numerical_vars <- sapply(loan, is.numeric)
>
> # Checks which variables are categorical.
> categorical_vars <- sapply(loan, is.factor)
>
> # Extract the names of numerical variables
> numerical_columns <- names(numerical_vars[numerical_vars])
> cat("Numerical Variables:\n", numerical_columns, "\n\n")
Numerical Variables:
 id loan_amnt int_rate installment annual_inc

>
> # Extract the names of categorical variables
> categorical_columns <- names(categorical_vars[categorical_vars])
> cat("Categorical Variables:\n", categorical_columns, "\n\n")
Categorical Variables:
 term grade emp_length home_ownership verification_status loan_status

>
```

3. Calculate the minimum, maximum, mean, median, standard deviation and three quartiles (25th, 50th and 75th percentiles) of loan_amnt.

```
>
> ## 3. Calculate the minimum, maximum, mean, median, standard deviation and
> ###   three quartiles (25th, 50th and 75th percentiles) of loan_amnt.
>
> # Calculate and display the minimum value of loan_amnt
> cat("Minimum of loan_amnt:", min(loan$loan_amnt, na.rm = TRUE), "\n")
Minimum of loan_amnt: 1000
>
> # Calculate and display the maximum value of loan_amnt
> cat("Maximum of loan_amnt:", max(loan$loan_amnt, na.rm = TRUE), "\n")
Maximum of loan_amnt: 35000
>
> # Calculate and display the mean value of loan_amnt
> cat("Mean of loan_amnt:", mean(loan$loan_amnt, na.rm = TRUE), "\n")
Mean of loan_amnt: 12861.64
>
```

```
> # Calculate and display the median value of loan_amnt
> cat("Median of loan_amnt:", median(loan$loan_amnt, na.rm = TRUE), "\n")
Median of loan_amnt: 11200
>
> # Calculate and display the standard deviation of loan_amnt
> cat("Standard deviation of loan_amnt:", sd(loan$loan_amnt, na.rm = TRUE), "\n")
Standard deviation of loan_amnt: 8491.814
>
> # Calculate the quartiles of loan_amnt
> percent <- quantile(loan$loan_amnt, probs = c(0.25, 0.50, 0.75), na.rm = TRUE)
>
> # Display the 25th percentile of loan_amnt
> cat("25% of loan_amnt:", percent[1], "\n")
25% of loan_amnt: 6000
>
> # Display the 50th percentile of loan_amnt (median)
> cat("50% of loan_amnt:", percent[2], "\n")
50% of loan_amnt: 11200
>
> # Display the 75th percentile of loan_amnt
> cat("75% of loan_amnt:", percent[3], "\n")
75% of loan_amnt: 17500
>
```



4. Calculate the minimum, maximum, mean, median, standard deviation and three quartiles (25th, 50th and 75th percentiles) of int_rate.

```
>
> ## 4. Calculate the minimum, maximum, mean, median, standard deviation and
> ###   three quartiles (25th, 50th and 75th percentiles) of int_rate.
>
> # Summary statistics for int_rate
> summary(loan$int_rate)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   5.42    8.90   12.42   12.43   15.27   24.11
>
> # Extract minimum value of int_rate
> cat("Minimum of int_rate:", summary(loan$int_rate)["Min."], "\n")
Minimum of int_rate: 5.42
>
> # Extract maximum value of int_rate
> cat("Maximum of int_rate:", summary(loan$int_rate)["Max."], "\n")
Maximum of int_rate: 24.11
>
> # Extract mean value of int_rate
> cat("Mean of int_rate:", summary(loan$int_rate)["Mean"], "\n")
Mean of int_rate: 12.42855
>
> # Extract median value of int_rate
> cat("Median of int_rate:", summary(loan$int_rate)["Median"], "\n")
Median of int_rate: 12.42
>
> # Calculate and display the standard deviation of int_rate
> cat("Standard deviation of int_rate:", sd(loan$int_rate, na.rm = TRUE), "\n")
Standard deviation of int_rate: 4.239117
>
> # Calculate the quartiles of int_rate
> percentile <- quantile(loan$int_rate, probs = c(0.25, 0.50, 0.75), na.rm = TRUE)
>
> # Display the 25th percentile of int_rate
> cat("25% of int_rate:", percentile["25%"], "\n")
25% of int_rate: 8.9
>
> # Display the 50th percentile of int_rate (median)
> cat("50% of int_rate:", percentile["50%"], "\n")
50% of int_rate: 12.42
>
> # Display the 75th percentile of int_rate
> cat("75% of int_rate:", percentile["75%"], "\n")
```

75% of int_rate: 15.27
>

```
>
> ## 4. Calculate the minimum, maximum, mean, median, standard deviation and
> ###   three quartiles (25th, 50th and 75th percentiles) of int_rate.
>
> # Summary statistics for int_rate
> summary(loan$int_rate)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   5.42    8.90   12.42   12.43   15.27   24.11
>
> # Extract minimum value of int_rate
> cat("Minimum of int_rate:", summary(loan$int_rate)["Min."], "\n")
Minimum of int_rate: 5.42
>
> # Extract maximum value of int_rate
> cat("Maximum of int_rate:", summary(loan$int_rate)["Max."], "\n")
Maximum of int_rate: 24.11
>
> # Extract mean value of int_rate
> cat("Mean of int_rate:", summary(loan$int_rate)["Mean"], "\n")
Mean of int_rate: 12.42855
>
> # Extract median value of int_rate
> cat("Median of int_rate:", summary(loan$int_rate)["Median"], "\n")
Median of int_rate: 12.42
>
> # Calculate and display the standard deviation of int_rate
> cat("Standard deviation of int_rate:", sd(loan$int_rate, na.rm = TRUE), "\n")
Standard deviation of int_rate: 4.239117
>
> # Calculate the quartiles of int_rate
> percentile <- quantile(loan$int_rate, probs = c(0.25, 0.50, 0.75), na.rm = TRUE)
>
> # Display the 25th percentile of int_rate
> cat("25% of int_rate:", percentile["25%"], "\n")
25% of int_rate: 8.9
>
> # Display the 50th percentile of int_rate (median)
> cat("50% of int_rate:", percentile["50%"], "\n")
50% of int_rate: 12.42
>
> # Display the 75th percentile of int_rate
> cat("75% of int_rate:", percentile["75%"], "\n")
75% of int_rate: 15.27
>
```

5. Calculate the correlation coefficient of the two variables: int_rate and installment. Do they have a strong relationship?

```
>
> ## 5. Calculate the correlation coefficient of the two variables: int_rate
> ###   and installment. Do they have a strong relationship?
>
> # Compute the correlation coefficient between int_rate and installment
> correlation_value <- cor(loan$int_rate, loan$installment, use = "complete.obs")
>
> # Display the correlation coefficient
> cat("Correlation between int_rate and installment:", correlation_value, "\n")
Correlation between int_rate and installment: 0.2819849
>
```

```
>
> ## 5. Calculate the correlation coefficient of the two variables: int_rate
> ###   and installment. Do they have a strong relationship?
>
> # Compute the correlation coefficient between int_rate and installment
> correlation_value <- cor(loan$int_rate, loan$installment, use = "complete.obs")
>
> # Display the correlation coefficient
> cat("Correlation between int_rate and installment:", correlation_value, "\n")
Correlation between int_rate and installment: 0.2819849
>
```

The correlation coefficient between int_rate and installment is 0.2819849, indicating a weak to moderate positive relationship. This suggests that while higher interest rates may lead to higher installment amounts, the connection is not strong. Since a strong correlation typically exceeds 0.7, this low value implies that other factors, such as loan amount and term length, likely have a greater influence on installment amounts.

6. Calculate the frequency table of term? What's the mode of term variable?

```
>
> ## 6. Calculate the frequency table of term? What's the mode of term variable?
>
> # Create a frequency table for the term variable
> term_table <- table(loan$term)
>
> # Identify the mode of the term variable
> mode_term <- names(term_table[term_table == max(term_table)])
>
> # Print the frequency table
> print(term_table)

 36 months  60 months
    6649      3351
>
> # Display the mode of the term variable
> cat("Mode of term:", mode_term, "\n")
Mode of term:  36 months
>
```

```
>
> ## 6. Calculate the frequency table of term? What's the mode of term variable?
>
> # Create a frequency table for the term variable
> term_table <- table(loan$term)
>
> # Identify the mode of the term variable
> mode_term <- names(term_table[term_table == max(term_table)])
>
> # Print the frequency table
> print(term_table)

 36 months  60 months
      6649       3351
>
> # Display the mode of the term variable
> cat("Mode of term:", mode_term, "\n")
Mode of term:  36 months
>
```

7. Calculate the proportion table of loan_status? What's the mode of loan_status variable?

```
>
> ## 7. Calculate the proportion table of loan_status? What's the mode of
> ###   loan_status variable?
>
> # Compute the proportion table for loan_status
> loan_status_table <- prop.table(table(loan$loan_status))
>
> # Identify the mode of the loan_status variable
> mode_loan_status <- names(loan_status_table[loan_status_table ==
+                                 max(loan_status_table)])
>
> # Print the proportion table
> print(loan_status_table)

     Charged Off          Current          Default       Fully Paid
          0.1517           0.0956           0.0002           0.7487
In Grace Period Late (16-30 days) Late (31-120 days)
          0.0008           0.0006           0.0024
>
> # Display the mode of the loan_status variable
> cat("Mode of loan_status:", mode_loan_status, "\n")
Mode of loan_status: Fully Paid
>
```

```
>
> ## 7. Calculate the proportion table of loan_status? What's the mode of
> ###   loan_status variable?
>
> # Compute the proportion table for loan_status
> loan_status_table <- prop.table(table(loan$loan_status))
>
> # Identify the mode of the loan_status variable
> mode_loan_status <- names(loan_status_table[loan_status_table ==
+                                               max(loan_status_table)])
>
> # Print the proportion table
> print(loan_status_table)

       Charged Off            Current            Default         Fully Paid
            0.1517             0.0956             0.0002             0.7487
   In Grace Period  Late (16-30 days) Late (31-120 days)
            0.0008             0.0006             0.0024
>
> # Display the mode of the loan_status variable
> cat("Mode of loan_status:", mode_loan_status, "\n")
Mode of loan_status: Fully Paid
>
```

8. Calculate the cross table of term and loan_status. Then produce proportions by row and column respectively.

```
>
> ## 8. Calculate the cross table of term and loan_status. Then produce
> ###   proportions by row and column respectively.
>
> # Compute the cross table of term and loan_status
> table_term_status <- table(loan$term, loan$loan_status)
>
> # Compute and print row proportions
> cross_table_row <- prop.table(table_term_status, margin = 1)
> cat("Row proportions of term and loan_status:\n")
Row proportions of term and loan_status:
> print(cross_table_row)

            Charged Off     Current     Default   Fully Paid In Grace Period
  36 months 0.1134005114 0.0000000000 0.0000000000 0.8865994886
0.0000000000
  60 months 0.2276932259 0.2852879737 0.0005968368 0.4750820651
0.0023873471

          Late (16-30 days) Late (31-120 days)
  36 months     0.0000000000     0.0000000000
  60 months     0.0017905103     0.0071620412
>
> # Compute and print column proportions
> cross_table_col <- prop.table(table_term_status, margin = 2)
> cat("Column proportions of term and loan_status:\n")
```

Column proportions of term and loan_status:
> print(cross_table_col)

	Charged Off	Current	Default	Fully Paid	In Grace Period
36 months	0.4970336	0.0000000	0.0000000	0.7873648	0.0000000
60 months	0.5029664	1.0000000	1.0000000	0.2126352	1.0000000

	Late (16-30 days)	Late (31-120 days)
36 months	0.0000000	0.0000000
60 months	1.0000000	1.0000000

>

```
>
> ## 8. Calculate the cross table of term and loan_status. Then produce
> ###   proportions by row and column respectively.
>
> # Compute the cross table of term and loan_status
> table_term_status <- table(loan$term, loan$loan_status)
>
> # Compute and print row proportions
> cross_table_row <- prop.table(table_term_status, margin = 1)
> cat("Row proportions of term and loan_status:\n")
Row proportions of term and loan_status:
> print(cross_table_row)

             Charged Off      Current      Default   Fully Paid In Grace Period
   36 months 0.1134005114 0.0000000000 0.0000000000 0.8865994886    0.0000000000
   60 months 0.2276932259 0.2852879737 0.0005968368 0.4750820651    0.0023873471

            Late (16-30 days) Late (31-120 days)
   36 months      0.0000000000       0.0000000000
   60 months      0.0017905103       0.0071620412
>
> # Compute and print column proportions
> cross_table_col <- prop.table(table_term_status, margin = 2)
> cat("Column proportions of term and loan_status:\n")
Column proportions of term and loan_status:
> print(cross_table_col)

            Charged Off   Current   Default Fully Paid In Grace Period
   36 months   0.4970336 0.0000000 0.0000000  0.7873648       0.0000000
   60 months   0.5029664 1.0000000 1.0000000  0.2126352       1.0000000

            Late (16-30 days) Late (31-120 days)
   36 months         0.0000000          0.0000000
   60 months         1.0000000          1.0000000
>
```

9. The data is stored in the data frame, loan. Please summarize all the variables using one command.

```
>
> ## 9. The data is stored in the data frame, loan. Please summarize all the
> ###   variables using one command.
>
> # Generate summary statistics for all variables in the dataset
> summary(loan)
       id          loan_amnt          term        int_rate
 Min.   : 458165   Min.   : 1000   36 months:6649   Min.   : 5.42
```

```
1st Qu.: 878178   1st Qu.: 6000    60 months:3351   1st Qu.: 8.90
Median : 987925   Median :11200                     Median :12.42
Mean   : 963545   Mean   :12862                     Mean   :12.43
3rd Qu.:1033696   3rd Qu.:17500                     3rd Qu.:15.27
Max.   :1077501   Max.   :35000                     Max.   :24.11

 installment     grade      emp_length    home_ownership  annual_inc
Min.   :  22.24   A:2765   10+ years:2548   MORTGAGE:4612   Min.   :  6000
1st Qu.: 193.58   B:3113   2 years  : 987   OWN     : 748   1st Qu.:  42000
Median : 322.25   C:1825   3 years  : 904   RENT    :4640   Median :  60000
Mean   : 363.82   D:1220   < 1 year : 900                   Mean   :  70267
3rd Qu.: 480.33   E: 718   4 years  : 861                   3rd Qu.:  84500
Max.   :1288.10   F: 292   5 years  : 855                   Max.   :1782000
                  G:  67   (Other)  :2945
   verification_status         loan_status
Not Verified   :3050    Charged Off     :1517
Source Verified:3069    Current         : 956
Verified       :3881    Default         :   2
                        Fully Paid      :7487
                        In Grace Period :   8
                        Late (16-30 days):   6
                        Late (31-120 days):  24
>
```

```
>
> ## 9. The data is stored in the data frame, loan. Please summarize all the
> ###   variables using one command.
>
> # Generate summary statistics for all variables in the dataset
> summary(loan)
       id            loan_amnt            term         int_rate
 Min.   : 458165   Min.   : 1000    36 months:6649   Min.   : 5.42
 1st Qu.: 878178   1st Qu.: 6000    60 months:3351   1st Qu.: 8.90
 Median : 987925   Median :11200                     Median :12.42
 Mean   : 963545   Mean   :12862                     Mean   :12.43
 3rd Qu.:1033696   3rd Qu.:17500                     3rd Qu.:15.27
 Max.   :1077501   Max.   :35000                     Max.   :24.11

  installment      grade        emp_length    home_ownership   annual_inc
 Min.   :  22.24   A:2765   10+ years:2548   MORTGAGE:4612   Min.   :  6000
 1st Qu.: 193.58   B:3113   2 years  : 987   OWN     : 748   1st Qu.:  42000
 Median : 322.25   C:1825   3 years  : 904   RENT    :4640   Median :  60000
 Mean   : 363.82   D:1220   < 1 year : 900                   Mean   :  70267
 3rd Qu.: 480.33   E: 718   4 years  : 861                   3rd Qu.:  84500
 Max.   :1288.10   F: 292   5 years  : 855                   Max.   :1782000
                   G:  67   (Other)  :2945
      verification_status            loan_status
 Not Verified   :3050     Charged Off     :1517
 Source Verified:3069     Current         : 956
 Verified       :3881     Default         :   2
                          Fully Paid      :7487
                          In Grace Period :   8
                          Late (16-30 days):   6
                          Late (31-120 days):  24
>
```