R Notebook

#loading library

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.4.4     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

#loading dataset

```
loan_approval <- read.csv("C:/Users/santo/OneDrive/Desktop/Fundamental of machinelearning/Assignment_1/l
head(loan_approval)
```

```
##   loan_id no_of_dependents     education self_employed income_annum loan_amount
## 1       1                2      Graduate            No      9600000    29900000
## 2       2                0  Not Graduate           Yes      4100000    12200000
## 3       3                3      Graduate            No      9100000    29700000
## 4       4                3      Graduate            No      8200000    30700000
## 5       5                5  Not Graduate           Yes      9800000    24200000
## 6       6                0      Graduate           Yes      4800000    13500000
##   loan_term cibil_score residential_assets_value commercial_assets_value
## 1        12         778                  2400000                17600000
## 2         8         417                  2700000                 2200000
## 3        20         506                  7100000                 4500000
## 4         8         467                 18200000                 3300000
## 5        20         382                 12400000                 8200000
## 6        10         319                  6800000                 8300000
##   luxury_assets_value bank_asset_value loan_status
## 1            22700000          8000000    Approved
## 2             8800000          3300000    Rejected
## 3            33300000         12800000    Rejected
## 4            23300000          7900000    Rejected
## 5            29400000          5000000    Rejected
## 6            13700000          5100000    Rejected
```

#Explanation #This code reads a CSV file containing loan approval data into an R data frame named "loan_approval" and then displays the first few rows of the dataset for a quick overview of the data.

```r
quantitative_vars <- c("loan_amount", "loan_term", "cibil_score")
quantitative_stats <- summary(loan_approval[, quantitative_vars])
print("Descriptive Statistics for Quantitative Variables:")
```

```
## [1] "Descriptive Statistics for Quantitative Variables:"
```

```r
print(quantitative_stats)
```

```
##   loan_amount         loan_term     cibil_score
##  Min.   :  300000   Min.   : 2.0   Min.   :300.0
##  1st Qu.: 7700000   1st Qu.: 6.0   1st Qu.:453.0
##  Median :14500000   Median :10.0   Median :600.0
##  Mean   :15133450   Mean   :10.9   Mean   :599.9
##  3rd Qu.:21500000   3rd Qu.:16.0   3rd Qu.:748.0
##  Max.   :39500000   Max.   :20.0   Max.   :900.0
```

#Explanation #This code provides descriptive statistics (summary statistics) for specific quantitative variables in the "loan_approval" data frame and quantitative variables in the dataset.

```r
categorical_var <- c("education","self_employed", "loan_status")
categorical_summary <- table(loan_approval[, categorical_var])
print("Summary for Categorical Variable:")
```

```
## [1] "Summary for Categorical Variable:"
```

```r
print(categorical_summary)
```

```
## , , loan_status =  Approved
##
##                self_employed
## education        No  Yes
##    Graduate     681  658
##    Not Graduate 637  680
##
## , , loan_status =  Rejected
##
##                self_employed
## education        No  Yes
##    Graduate     408  397
##    Not Graduate 393  415
```

#Explanation #This code provides categorical statistics (summary statistics) for specific categorical variables in the "loan_approval" data frame and categorical variables in the dataset.

```r
#Transform the "loan_amount" variable by adding a constant value
constant_value <- 1000000
loan_amount_transformed <- loan_approval$loan_amount + constant_value
#Display the first few rows of the dataset after transformation
print("Dataset after transforming 'loan_amount':")
```

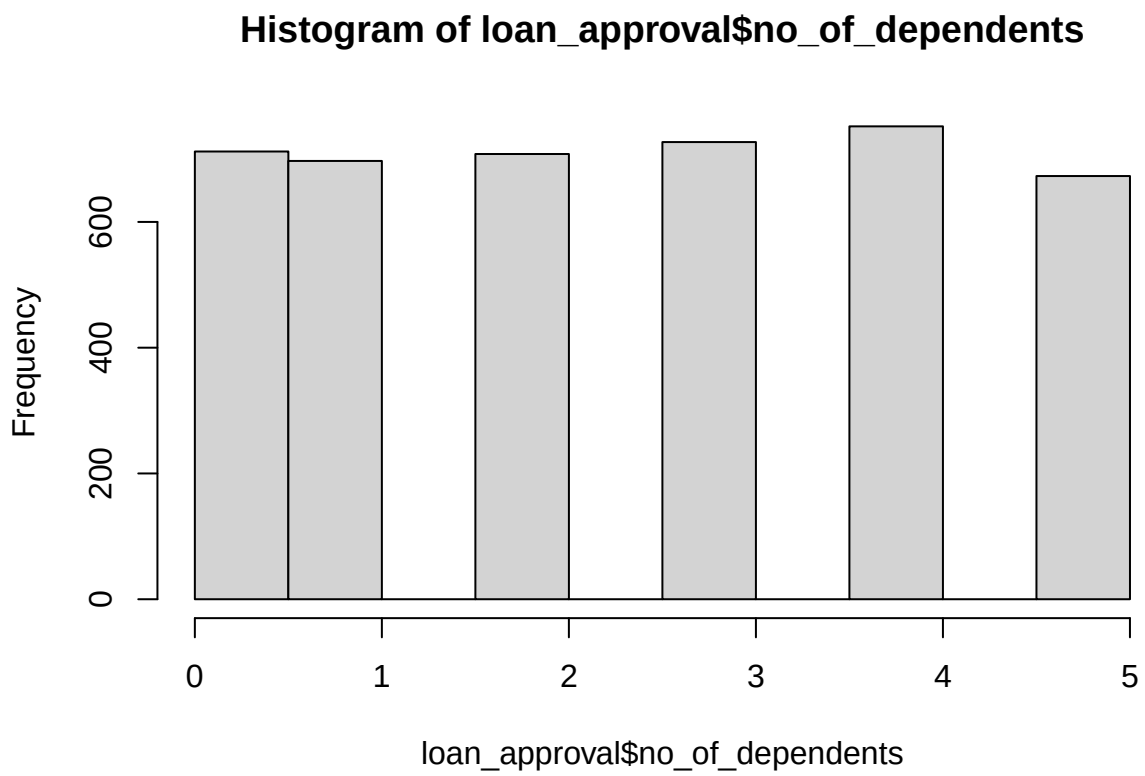
```
## [1] "Dataset after transforming 'loan_amount':"
```

```r
print(head(loan_amount_transformed))
```

```
## [1] 30900000 13200000 30700000 31700000 25200000 14500000
```

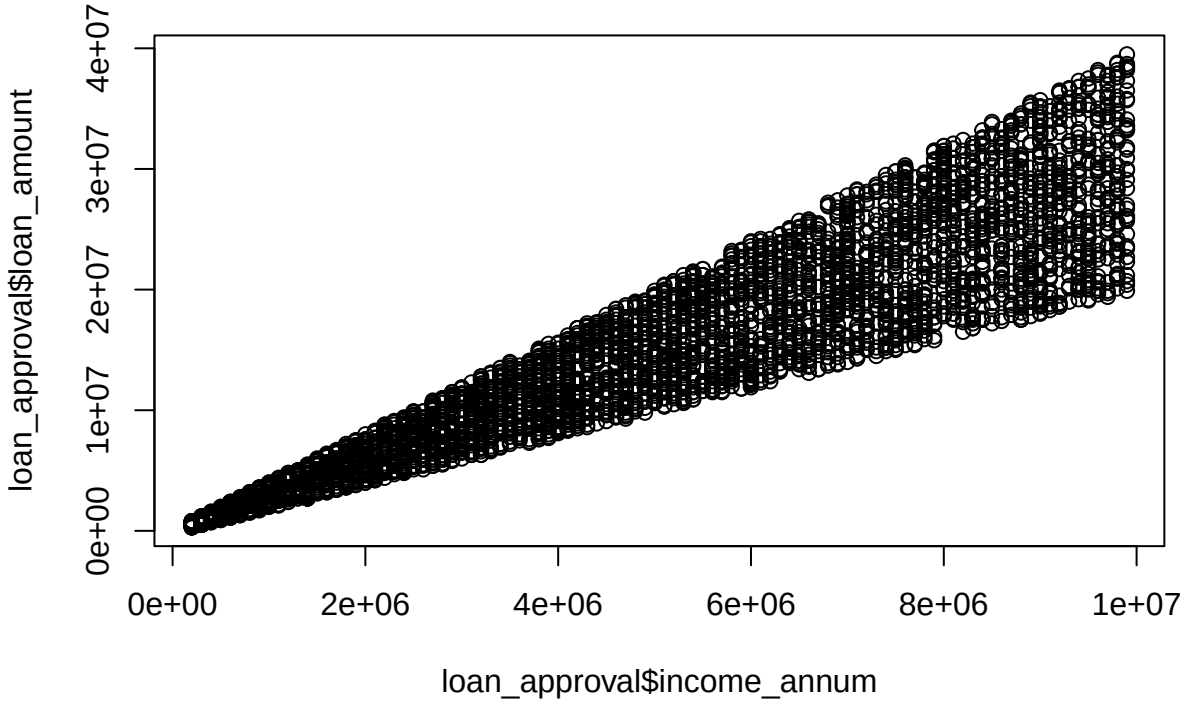
#Explanation #This code shows how to transform a variable in a dataset by adding a constant value. It adds 1,000,000 to each observation in the "loan_amount" variable and displays the first few rows of the dataset after this transformation.

```r
# Plot histogram for the distribution of "income_annum"
hist(loan_approval$no_of_dependents)
```



#Explanation #The resulting plot will show the distribution of the "no_of_dependents" variable, giving insights into the number of dependents for individuals in the dataset. Histograms are useful for visualizing the shape and spread of a distribution and identifying patterns in the data.

```r
# Plot scatterplot between "income_annum" and "loan_amount"
plot(loan_approval$income_annum,loan_approval$loan_amount)
```



#Explanation #The graph shows the relationship between “income_annum” and “loan_amount” in the “loan_approval” dataset. x-coordinate corresponding to the “income_annum” value and the y-coordinate corresponding to the “loan_amount” value.This type of plot is useful for identifying patterns or trends between two numeric variables.