TASK 3: CREDIT CARD FRAUD DETECTION

```r
# Load necessary libraries
library(dplyr)
library(ggplot2)
library(caTools)
library(ROSE)

# Read the dataset
creditcard <- read.csv("C:\\Users\\Tayyaba\\Documents\\internship tasks\\task 3\\credit
```

```
> # Data exploration and summary
> dim(creditcard) # Get dimensions of the dataset
[1] 284807     31
> head(creditcard)# Display the first few rows of the dataset
  Time         V1          V2        V3         V4          V5          V6          V7
1    0 -1.3598071 -0.07278117 2.5363467  1.3781552 -0.33832077  0.46238778  0.23959855
2    0  1.1918571  0.26615071 0.1664801  0.4481541  0.06001765 -0.08236081 -0.07880298
3    1 -1.3583541 -1.34016307 1.7732093  0.3797796 -0.50319813  1.80049938  0.79146096
4    1 -0.9662717 -0.18522601 1.7929933 -0.8632913 -0.01030888  1.24720317  0.23760894
5    2 -1.1582331  0.87773675 1.5487178  0.4030339 -0.40719338  0.09592146  0.59294075
6    2 -0.4259659  0.96052304 1.1411093 -0.1682521  0.42098688 -0.02972755  0.47620095
          V8         V9         V10        V11         V12        V13        V14
1  0.09869790  0.3637870  0.09079417 -0.5515995 -0.61780086 -0.9913898 -0.3111694
2  0.08510165 -0.2554251 -0.16697441  1.6127267  1.06523531  0.4890950 -0.1437723
3  0.24767579 -1.5146543  0.20764287  0.6245015  0.06608369  0.7172927 -0.1659459
4  0.37743587 -1.3870241 -0.05495192 -0.2264873  0.17822823  0.5077569 -0.2879237
5 -0.27053268  0.8177393  0.75307443 -0.8228429  0.53819555  1.3458516 -1.1196698
6  0.26031433 -0.5686714 -0.37140720  1.3412620  0.35989384 -0.3580907 -0.1371337
         V15        V16         V17         V18         V19         V20          V21
1  1.4681770 -0.4704005  0.20797124  0.02579058  0.40399296  0.25141210 -0.018306778
2  0.6355581  0.4639170 -0.11480466 -0.18336127 -0.14578304 -0.06908314 -0.225775248
3  2.3458649 -2.8900832  1.10996938 -0.12135931 -2.26185710  0.52497973  0.247998153
4 -0.6314181 -1.0596472 -0.68409279  1.96577500 -1.23262197 -0.20803778 -0.108300452
5  0.1751211 -0.4514492 -0.23703324 -0.03819479  0.80348692  0.40854236 -0.009430697
6  0.5176168  0.4017259 -0.05813282  0.06865315 -0.03319379  0.08496767 -0.208253515
          V22         V23         V24        V25        V26          V27         V28
1  0.277837576 -0.11047391  0.06692807  0.1285394 -0.1891148  0.133558377 -0.02105305
2 -0.638671953  0.10128802 -0.33984648  0.1671704  0.1258945 -0.008983099  0.01472417
3  0.771679402  0.90941226 -0.68928096 -0.3276418 -0.1390966 -0.055352794 -0.05975184
4  0.005273597 -0.19032052 -1.17557533  0.6473760 -0.2219288  0.062722849  0.06145763
5  0.798278495 -0.13745808  0.14126698 -0.2060096  0.5022922  0.219422230  0.21515315
6 -0.559824796 -0.02639767 -0.37142658 -0.2327938  0.1059148  0.253844225  0.08108026
  Amount Class
1 149.62     0
2   2.69     0
3 378.66     0
4 123.50     0
5  69.99     0
6   3.67     0
```

```
> tail(creditcard) # Display the last few rows of the dataset
         Time          V1          V2         V3         V4          V5         V6
284802 172785   0.1203164  0.93100513 -0.5460121 -0.7450968  1.13031398 -0.2359732
284803 172786 -11.8811179 10.07178497 -9.8347835 -2.0666557 -5.36447278 -2.6068373
284804 172787  -0.7327887 -0.05508049  2.0350297 -0.7385886  0.86822940  1.0584153
284805 172788   1.9195650 -0.30125385 -3.2496398 -0.5578281  2.63051512  3.0312601
284806 172788  -0.2404400  0.53048251  0.7025102  0.6897992 -0.37796113  0.6237077
284807 172792  -0.5334125 -0.18973334  0.7033374 -0.5062712 -0.01254568 -0.6496167
               V7         V8         V9        V10        V11         V12        V13
284802  0.8127221  0.1150929 -0.2040635 -0.6574221  0.6448373  0.19091623 -0.5463289
284803 -4.9182154  7.3053340  1.9144283  4.3561704 -1.5931053  2.71194079 -0.6892556
284804  0.0243297  0.2948687  0.5848000 -0.9759261 -0.1501888  0.91580191  1.2147558
284805 -0.2968265  0.7084172  0.4324540 -0.4847818  0.4116137  0.06311886 -0.1836987
284806 -0.6861800  0.6791455  0.3920867 -0.3991257 -1.9338488 -0.96288614 -1.0420817
284807  1.5770063 -0.4146504  0.4861795 -0.9154266 -1.0404583 -0.03151305 -0.1880929
               V14         V15        V16         V17        V18        V19          V20
284802 -0.73170658 -0.80803553  0.5996281  0.07044075  0.3731103  0.1289038 0.0006758329
284803  4.62694203 -0.92445871  1.1076406  1.99169111  0.5106323 -0.6829197 1.4758291347
284804 -0.67514296  1.16493091 -0.7117573 -0.02569286 -1.2211789 -1.5455561 0.0596158999
284805 -0.51060184  1.32928351  0.1407160  0.31350179  0.3956525 -0.5772518 0.0013959703
284806  0.44962444  1.96256312 -0.6085771  0.50992846  1.1139806  2.8978488 0.1274335158
284807 -0.08431647  0.04133346 -0.3026201 -0.66037665  0.1674299 -0.2561169 0.3829481049
              V21        V22         V23          V24        V25        V26          V27
284802 -0.3142046 -0.8085204  0.05034266  0.102799590 -0.4358701  0.1240789  0.217939865
284803  0.2134541  0.1118637  1.01447990 -0.509348453  1.4368069  0.2500343  0.943651172
284804  0.2142053  0.9243836  0.01246304 -1.016225669 -0.6066240 -0.3952551  0.068472470
284805  0.2320450  0.5782290 -0.03750086  0.640133881  0.2657455 -0.0873706  0.004454772
284806  0.2652449  0.8000487 -0.16329794  0.123205244 -0.5691589  0.5466685  0.108820735
284807  0.2610573  0.6430784  0.37677701  0.008797379 -0.4736487 -0.8182671 -0.002415309
               V28 Amount Class
284802  0.06880333   2.69     0
284803  0.82373096   0.77     0
284804 -0.05352739  24.79     0
284805 -0.02656083  67.88     0
284806  0.10453282  10.00     0
284807  0.01364891 217.00     0
```

```
> # Check if there are any missing values in the dataset
> any(is.na(creditcard))
[1] FALSE
>
```

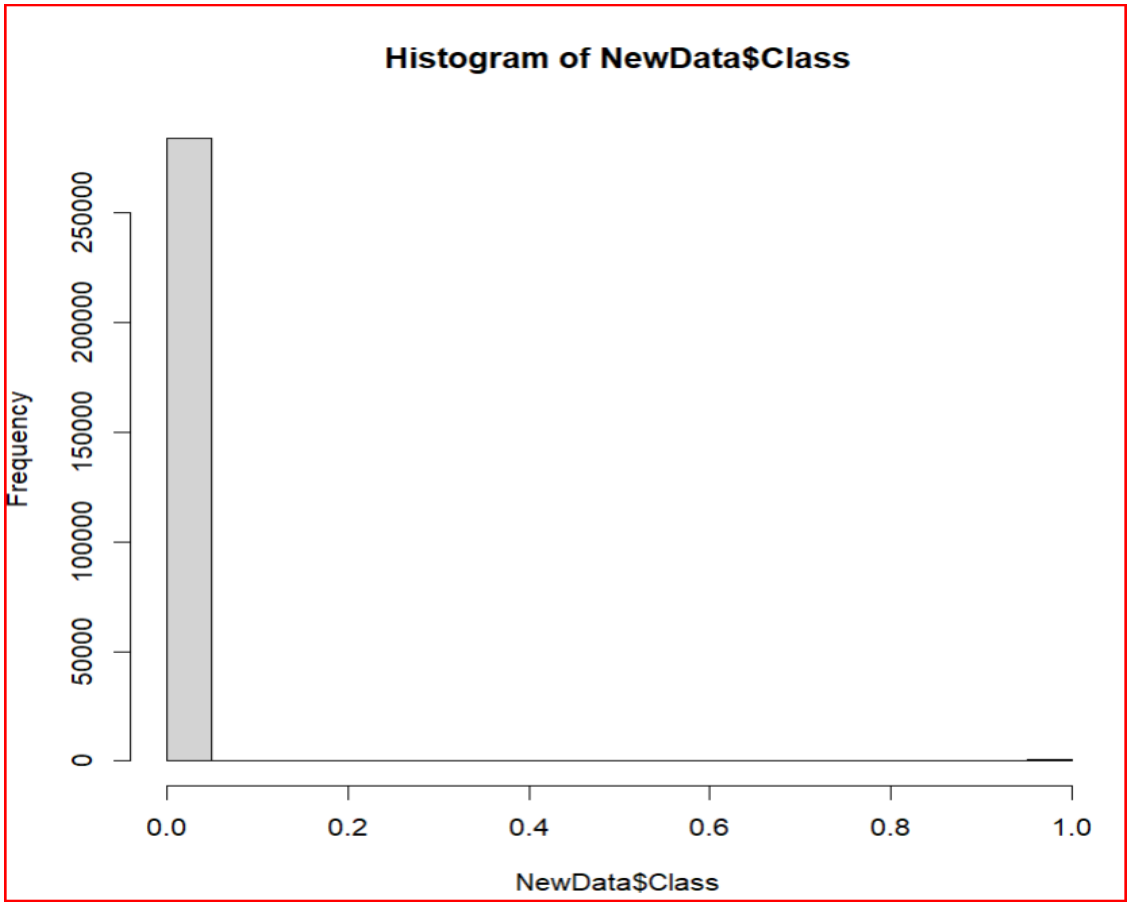
```
> # Display the structure of the dataset
> str(creditcard)
'data.frame':   284807 obs. of  31 variables:
 $ Time  : num  0 0 1 1 2 2 4 7 7 9 ...
 $ V1    : num  -1.36 1.192 -1.358 -0.966 -1.158 ...
 $ V2    : num  -0.0728 0.2662 -1.3402 -0.1852 0.8777 ...
 $ V3    : num  2.536 0.166 1.773 1.793 1.549 ...
 $ V4    : num  1.378 0.448 0.38 -0.863 0.403 ...
 $ V5    : num  -0.3383 0.06 -0.5032 -0.0103 -0.4072 ...
 $ V6    : num  0.4624 -0.0824 1.8005 1.2472 0.0959 ...
 $ V7    : num  0.2396 -0.0788 0.7915 0.2376 0.5929 ...
 $ V8    : num  0.0987 0.0851 0.2477 0.3774 -0.2705 ...
 $ V9    : num  0.364 -0.255 -1.515 -1.387 0.818 ...
 $ V10   : num  0.0908 -0.167 0.2076 -0.055 0.7531 ...
 $ V11   : num  -0.552 1.613 0.625 -0.226 -0.823 ...
 $ V12   : num  -0.6178 1.0652 0.0661 0.1782 0.5382 ...
 $ V13   : num  -0.991 0.489 0.717 0.508 1.346 ...
 $ V14   : num  -0.311 -0.144 -0.166 -0.288 -1.12 ...
 $ V15   : num  1.468 0.636 2.346 -0.631 0.175 ...
 $ V16   : num  -0.47 0.464 -2.89 -1.06 -0.451 ...
 $ V17   : num  0.208 -0.115 1.11 -0.684 -0.237 ...
 $ V18   : num  0.0258 -0.1834 -0.1214 1.9658 -0.0382 ...
 $ V19   : num  0.404 -0.146 -2.262 -1.233 0.803 ...
 $ V20   : num  0.2514 -0.0691 0.525 -0.208 0.4085 ...
 $ V21   : num  -0.01831 -0.22578 0.248 -0.1083 -0.00943 ...
 $ V22   : num  0.27784 -0.63867 0.77168 0.00527 0.79828 ...
 $ V23   : num  -0.11 0.101 0.909 -0.19 -0.137 ...
 $ V24   : num  0.0669 -0.3398 -0.6893 -1.1756 0.1413 ...
 $ V25   : num  0.129 0.167 -0.328 0.647 -0.206 ...
 $ V26   : num  -0.189 0.126 -0.139 -0.222 0.502 ...
 $ V27   : num  0.13356 -0.00898 -0.05535 0.06272 0.21942 ...
 $ V28   : num  -0.0211 0.0147 -0.0598 0.0615 0.2152 ...
 $ Amount: num  149.62 2.69 378.66 123.5 69.99 ...
 $ Class : int  0 0 0 0 0 0 0 0 0 0 ...
```

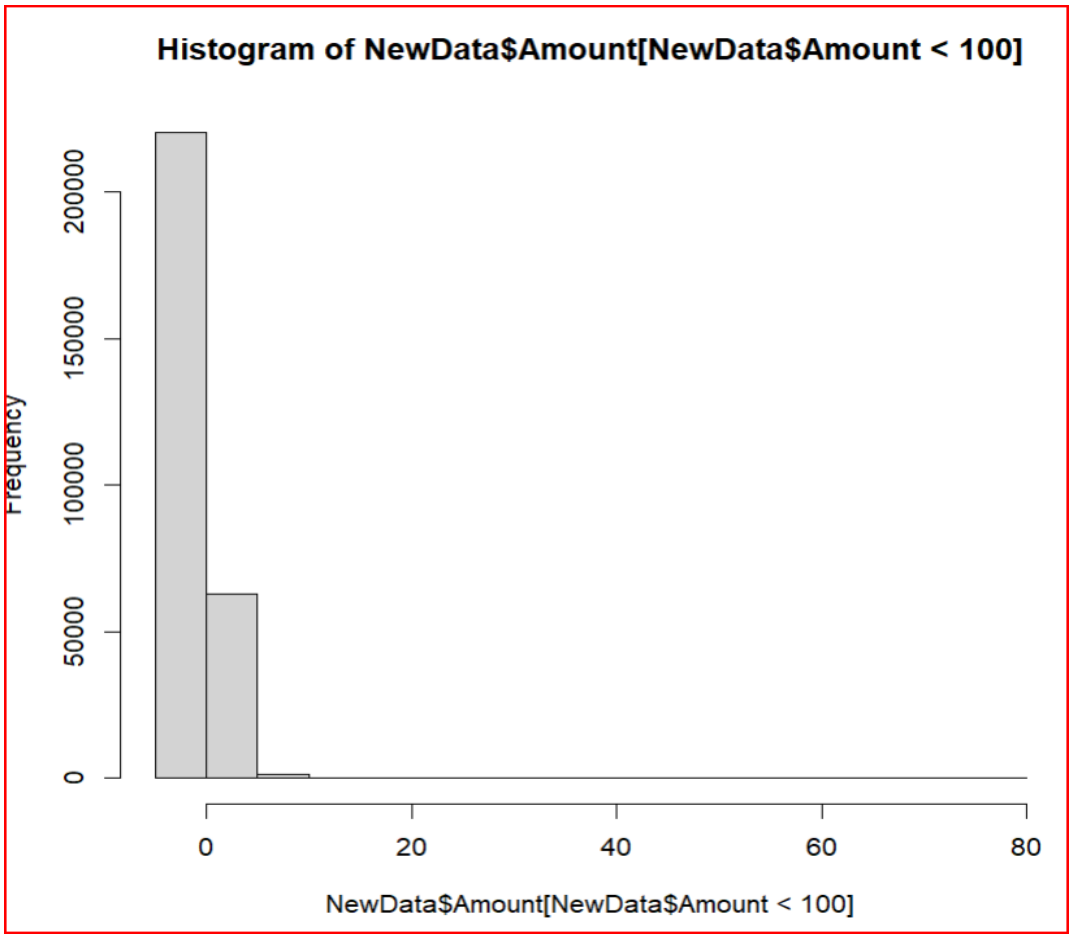
```
> #Summary
> table(creditcard$Class)           # Count the number of each class (fraudulent vs. non-fra
udulent)

     0      1
284315    492
> names(creditcard)                 # List column names
 [1] "Time"   "V1"     "V2"     "V3"     "V4"     "V5"     "V6"     "V7"     "V8"
[10] "V9"     "V10"    "V11"    "V12"    "V13"    "V14"    "V15"    "V16"    "V17"
[19] "V18"    "V19"    "V20"    "V21"    "V22"    "V23"    "V24"    "V25"    "V26"
[28] "V27"    "V28"    "Amount" "Class"
> summary(creditcard)               # Display summary statistics
      Time               V1                    V2                   V3
 Min.   :     0   Min.   :-56.40751   Min.   :-72.71573   Min.   :-48.3256
 1st Qu.: 54202   1st Qu.: -0.92037   1st Qu.: -0.59855   1st Qu.: -0.8904
 Median : 84692   Median :  0.01811   Median :  0.06549   Median :  0.1799
 Mean   : 94814   Mean   :  0.00000   Mean   :  0.00000   Mean   :  0.0000
 3rd Qu.:139321   3rd Qu.:  1.31564   3rd Qu.:  0.80372   3rd Qu.:  1.0272
 Max.   :172792   Max.   :  2.45493   Max.   : 22.05773   Max.   :  9.3826
       V4                   V5                    V6                   V7
 Min.   :-5.68317   Min.   :-113.74331   Min.   :-26.1605   Min.   :-43.5572
 1st Qu.:-0.84864   1st Qu.:  -0.69160   1st Qu.: -0.7683   1st Qu.: -0.5541
 Median :-0.01985   Median :  -0.05434   Median : -0.2742   Median :  0.0401
 Mean   : 0.00000   Mean   :   0.00000   Mean   :  0.0000   Mean   :  0.0000
 3rd Qu.: 0.74334   3rd Qu.:   0.61193   3rd Qu.:  0.3986   3rd Qu.:  0.5704
 Max.   :16.87534   Max.   :  34.80167   Max.   : 73.3016   Max.   :120.5895
       V8                   V9                   V10                  V11
 Min.   :-73.21672   Min.   :-13.43407   Min.   :-24.58826   Min.   :-4.79747
 1st Qu.: -0.20863   1st Qu.: -0.64310   1st Qu.: -0.53543   1st Qu.:-0.76249
 Median :  0.02236   Median : -0.05143   Median : -0.09292   Median :-0.03276
 Mean   :  0.00000   Mean   :  0.00000   Mean   :  0.00000   Mean   : 0.00000
 3rd Qu.:  0.32735   3rd Qu.:  0.59714   3rd Qu.:  0.45392   3rd Qu.: 0.73959
 Max.   : 20.00721   Max.   : 15.59500   Max.   : 23.74514   Max.   :12.01891
      V12                  V13                  V14                  V15
 Min.   :-18.6837   Min.   :-5.79188   Min.   :-19.2143   Min.   :-4.49894
 1st Qu.: -0.4056   1st Qu.:-0.64854   1st Qu.: -0.4256   1st Qu.:-0.58288
 Median :  0.1400   Median :-0.01357   Median :  0.0506   Median : 0.04807
 Mean   :  0.0000   Mean   : 0.00000   Mean   :  0.0000   Mean   : 0.00000
 3rd Qu.:  0.6182   3rd Qu.: 0.66251   3rd Qu.:  0.4931   3rd Qu.: 0.64882
```

```
> sd(creditcard$Time)               # Calculate standard deviation of 'Time'
[1] 47488.15
> var(creditcard)                   # Calculate variance of all numeric columns
               Time             V1            V2            V3            V4
Time    2.255124e+09  1.091960e+04 -8.307031e+02 -3.021425e+04 -7.077378e+03
V1      1.091960e+04  3.836489e+00 -2.252859e-16 -1.689642e-15 -7.218405e-16
V2     -8.307031e+02 -2.252859e-16  2.726820e+00  1.303832e-16 -3.771750e-16
V3     -3.021425e+04 -1.689642e-15  1.303832e-16  2.299029e+00 -4.786833e-16
V4     -7.077378e+03 -7.218405e-16 -3.771750e-16 -4.786833e-16  2.004684e+00
V5      1.134407e+04  8.507676e-16  2.550723e-16 -1.258796e-15 -3.598732e-15
V6     -3.986868e+03  3.895661e-16  8.515163e-16  2.883053e-15 -8.012118e-16
V7      4.976739e+03  1.900133e-16 -2.671272e-16  4.309333e-16 -1.300357e-16
V8     -2.095683e+03 -1.274057e-16 -4.854199e-17 -1.332219e-16  1.083182e-15
V9     -4.518322e+02  8.205568e-17 -2.037673e-16  1.727645e-16  9.265182e-16
V10     1.583108e+03  1.135394e-16 -2.414319e-16  3.380706e-16 -1.610071e-16
V11    -1.200595e+04  6.002369e-16  5.796176e-16  1.759162e-16 -4.222618e-16
V12     5.900343e+03  3.559126e-16 -5.362190e-16  3.190443e-16 -2.743454e-16
V13    -3.114775e+03 -9.600057e-17 -6.214741e-17 -5.311266e-17  2.442103e-17
V14    -4.495601e+03  7.271512e-16 -6.025785e-16  9.736048e-16 -1.312263e-16
V15    -7.974101e+03 -1.637786e-16  9.759626e-17 -8.760794e-17  2.389862e-16
V16     4.952977e+02  5.749402e-16  5.886837e-17  7.591802e-16 -5.189596e-17
V17    -2.956329e+03 -3.948944e-17 -8.981140e-16  1.187564e-16 -4.483019e-16
V18     3.599748e+03  2.411640e-16  3.230786e-16  3.975738e-16 -1.797729e-17
V19     1.120106e+03  2.630740e-16  1.616504e-17  4.266434e-16 -3.324410e-16
V20    -1.862195e+03  2.163209e-16  1.043131e-16  8.188145e-17 -2.006220e-16
```

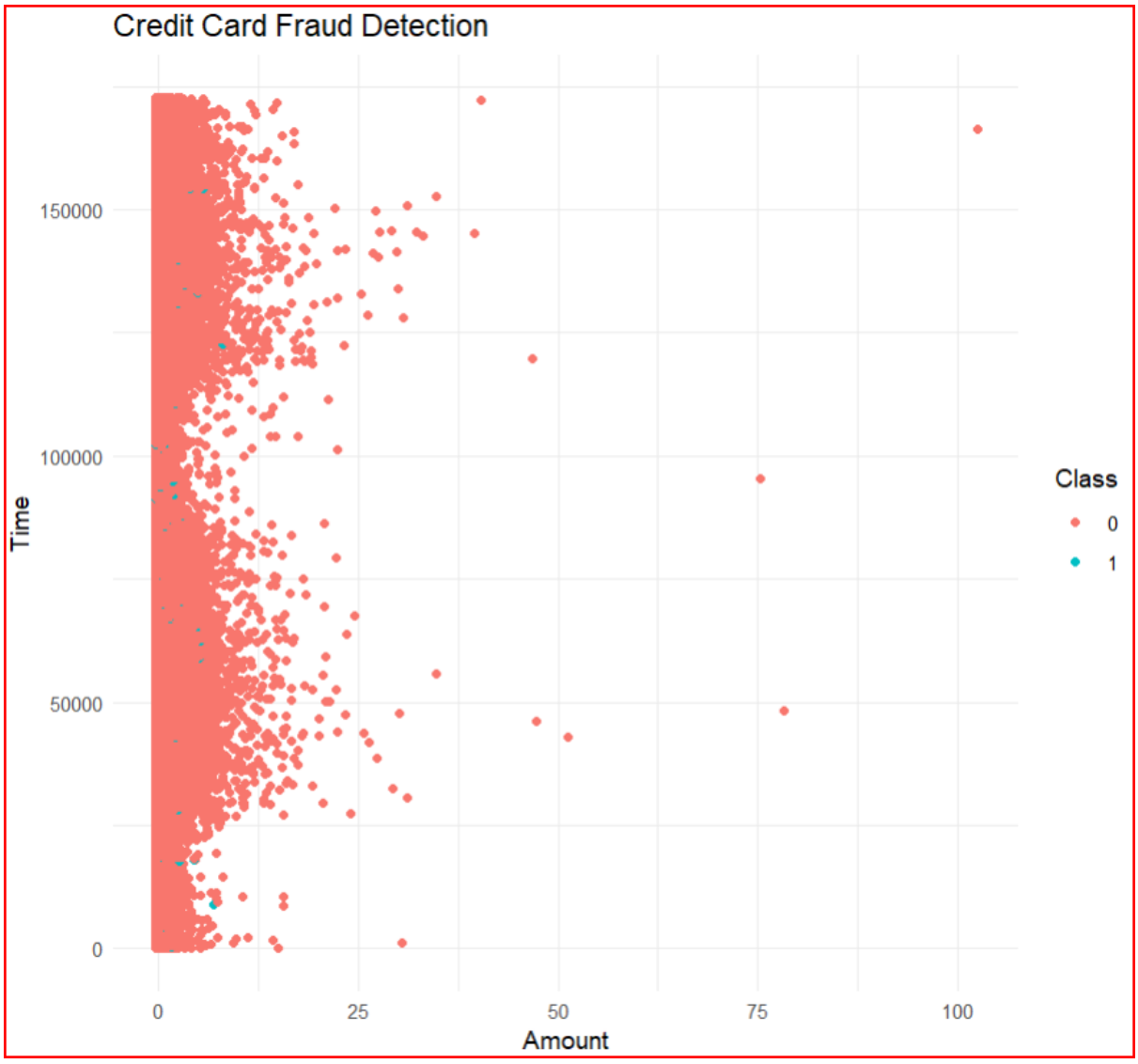
```
> creditcard$Amount=scale(creditcard$Amount)
> NewData=creditcard[,-1]
> head(NewData)
         V1          V2        V3         V4          V5          V6          V7
1 -1.3598071 -0.07278117 2.5363467  1.3781552 -0.33832077  0.46238778  0.23959855
2  1.1918571  0.26615071 0.1664801  0.4481541  0.06001765 -0.08236081 -0.07880298
3 -1.3583541 -1.34016307 1.7732093  0.3797796 -0.50319813  1.80049938  0.79146096
4 -0.9662717 -0.18522601 1.7929933 -0.8632913 -0.01030888  1.24720317  0.23760894
5 -1.1582331  0.87773675 1.5487178  0.4030339 -0.40719338  0.09592146  0.59294075
6 -0.4259659  0.96052304 1.1411093 -0.1682521  0.42098688 -0.02972755  0.47620095
          V8         V9         V10        V11         V12        V13         V14
1  0.09869790  0.3637870  0.09079417 -0.5515995 -0.61780086 -0.9913898 -0.3111694
2  0.08510165 -0.2554251 -0.16697441  1.6127267  1.06523531  0.4890950 -0.1437723
3  0.24767579 -1.5146543  0.20764287  0.6245015  0.06608369  0.7172927 -0.1659459
4  0.37743587 -1.3870241 -0.05495192 -0.2264873  0.17822823  0.5077569 -0.2879237
5 -0.27053268  0.8177393  0.75307443 -0.8228429  0.53819555  1.3458516 -1.1196698
6  0.26031433 -0.5686714 -0.37140720  1.3412620  0.35989384 -0.3580907 -0.1371337
        V15        V16         V17         V18         V19         V20          V21
1  1.4681770 -0.4704005  0.20797124  0.02579058  0.40399296  0.25141210 -0.018306778
2  0.6355581  0.4639170 -0.11480466 -0.18336127 -0.14578304 -0.06908314 -0.225775248
3  2.3458649 -2.8900832  1.10996938 -0.12135931 -2.26185710  0.52497973  0.247998153
4 -0.6314181 -1.0596472 -0.68409279  1.96577500 -1.23262197 -0.20803778 -0.108300452
5  0.1751211 -0.4514492 -0.23703324 -0.03819479  0.80348692  0.40854236 -0.009430697
6  0.5176168  0.4017259 -0.05813282  0.06865315 -0.03319379  0.08496767 -0.208253515
          V22         V23         V24        V25         V26          V27         V28
1  0.277837576 -0.11047391  0.06692807  0.1285394 -0.1891148  0.133558377 -0.02105305
2 -0.638671953  0.10128802 -0.33984648  0.1671704  0.1258945 -0.008983099  0.01472417
3  0.771679402  0.90941226 -0.68928096 -0.3276418 -0.1390966 -0.055352794 -0.05975184
4  0.005273597 -0.19032052 -1.17557533  0.6473760 -0.2219288  0.062722849  0.06145763
5  0.798278495 -0.13745808  0.14126698 -0.2060096  0.5022922  0.219422230  0.21515315
6 -0.559824796 -0.02639767 -0.37142658 -0.2327938  0.1059148  0.253844225  0.08108026
       Amount Class
1  0.24496383     0
2 -0.34247394     0
3  1.16068389     0
4  0.14053401     0
5 -0.07340321     0
```

```
> hist(NewData$Class)                  # Plot histogram of the 'Class' variable
> hist(NewData$Amount[NewData$Amount < 100])  # Plot histogram of 'Amount' with a filter
>
```

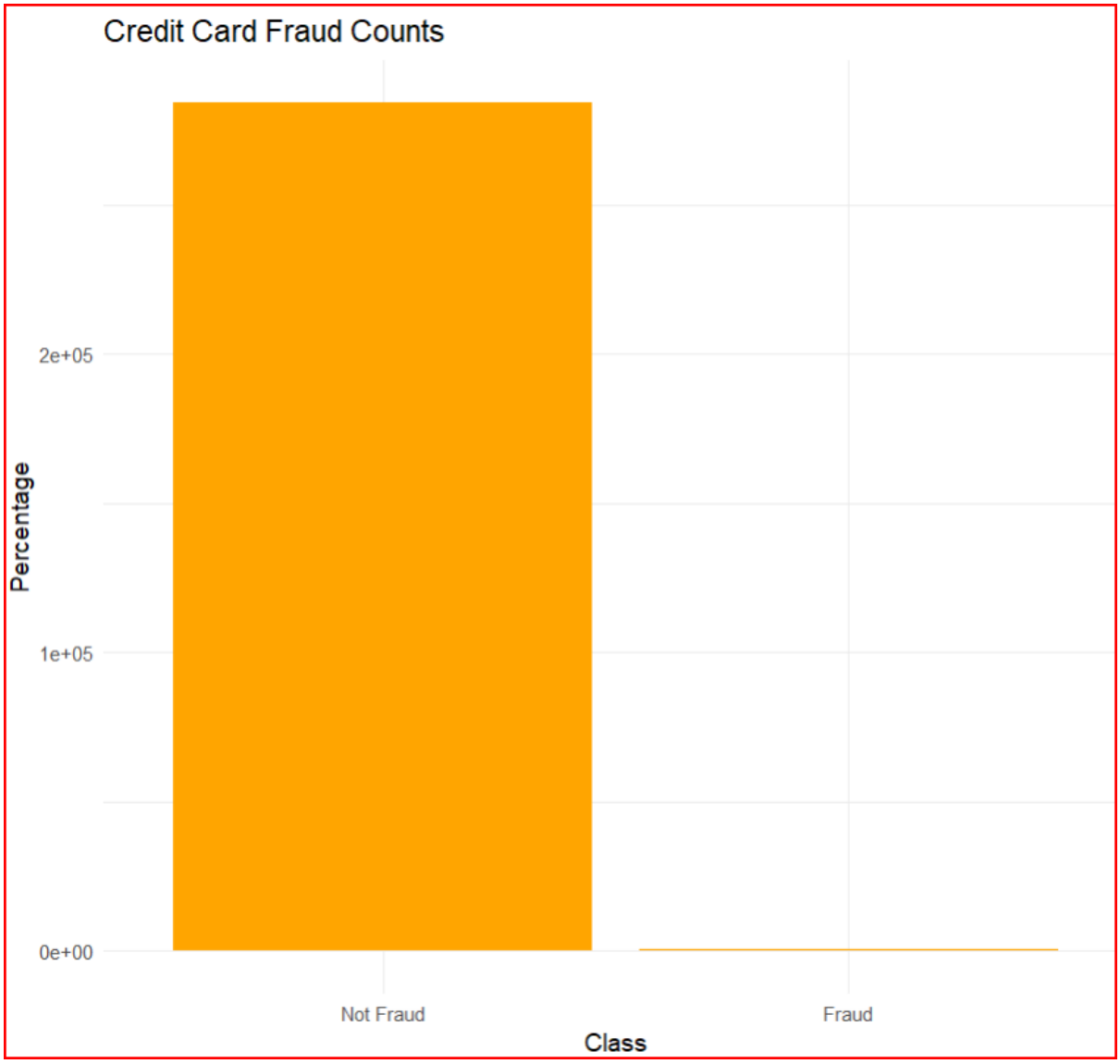

Histogram of NewData$Class
Frequency
0
50000
100000
150000
200000
250000
0.0
0.2
0.4
0.6
0.8
1.0
NewData$Class


Histogram of NewData$Amount[NewData$Amount < 100]
Frequency
0
50000
100000
150000
200000
0
20
40
60
80
NewData$Amount[NewData$Amount < 100]

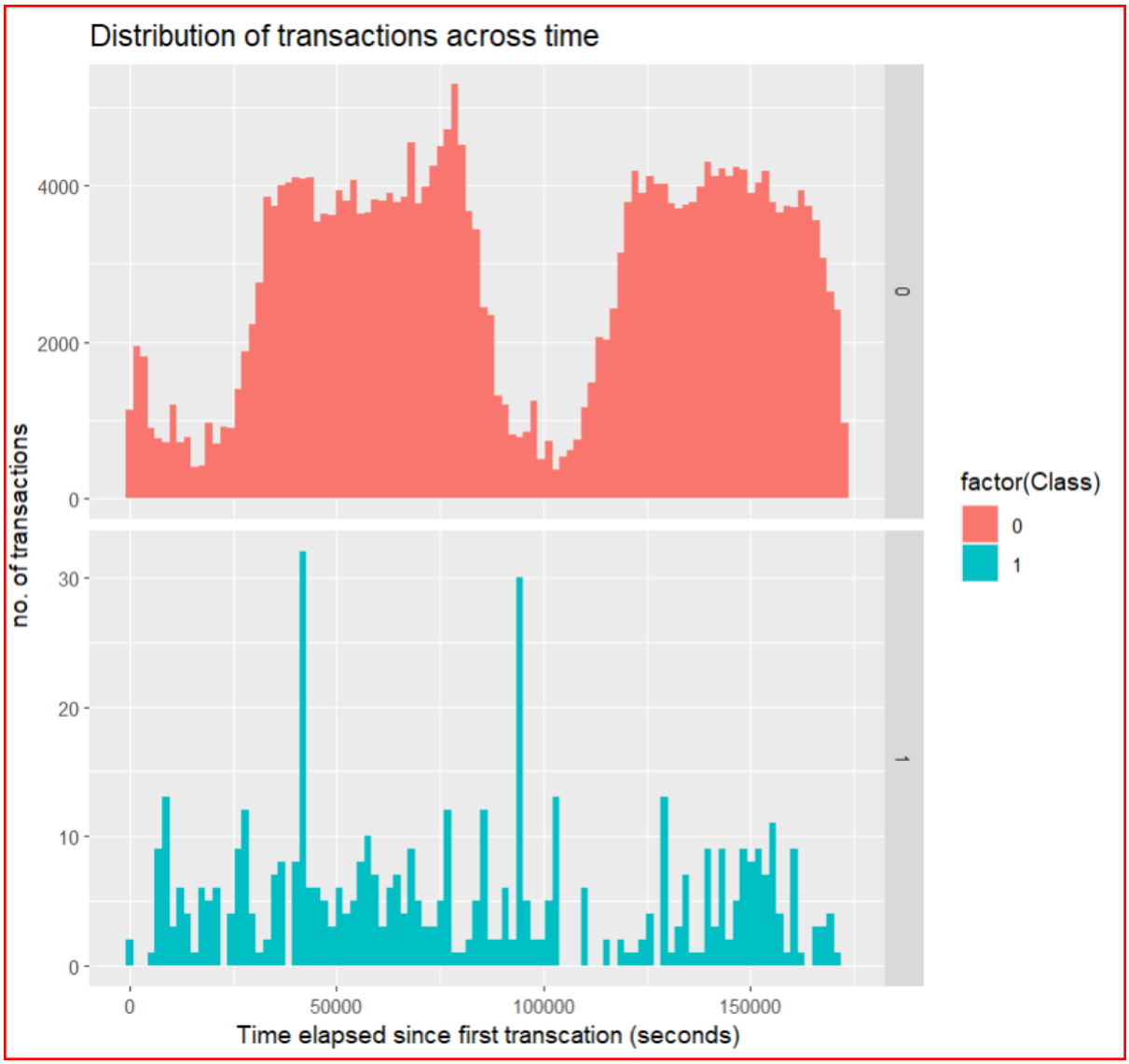
```
> # Create scatter plot with color differentiation by class
> ggplot(creditcard, aes(x = Amount, y = Time, color = factor(Class))) +
+   geom_point() +
+   labs(title = "Credit Card Fraud Detection",
+        x = "Amount",
+        y = "Time",
+        color = "Class") +
+   theme_minimal()
```



```
> # Create bar plot to show counts of fraud and non-fraud transactions
> ggplot(NewData, aes(x = factor(Class))) +
+   geom_bar(position = "dodge", fill = "orange") +
+   labs(title = "Credit Card Fraud Counts",
+        x = "Class",
+        y = "Percentage") +
+   scale_x_discrete(labels = c("Not Fraud", "Fraud")) +
+   theme_minimal()
>
```

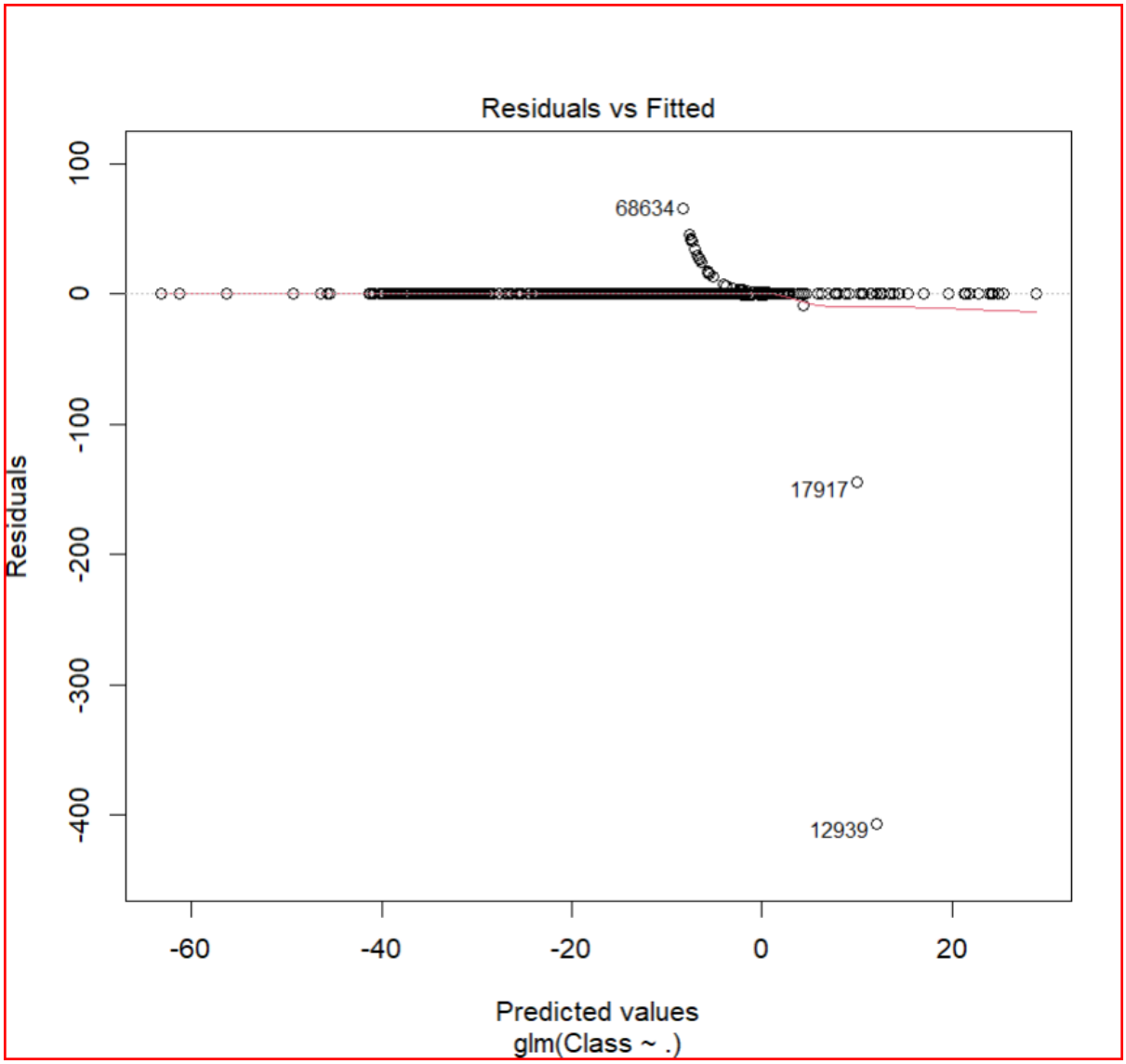


```
> # More data visualization - histogram of 'Time' for each class
> ggplot(creditcard, aes(x = Time , fill = factor(Class))) +
+   geom_histogram(bins = 100) +
+   labs(x = "Time elapsed since first transcation (seconds)",
+        y = "no. of transactions",
+        title = "Distribution of transactions across time") +
+   facet_grid(Class ~ ., scales = 'free_y') + theme()
>
```

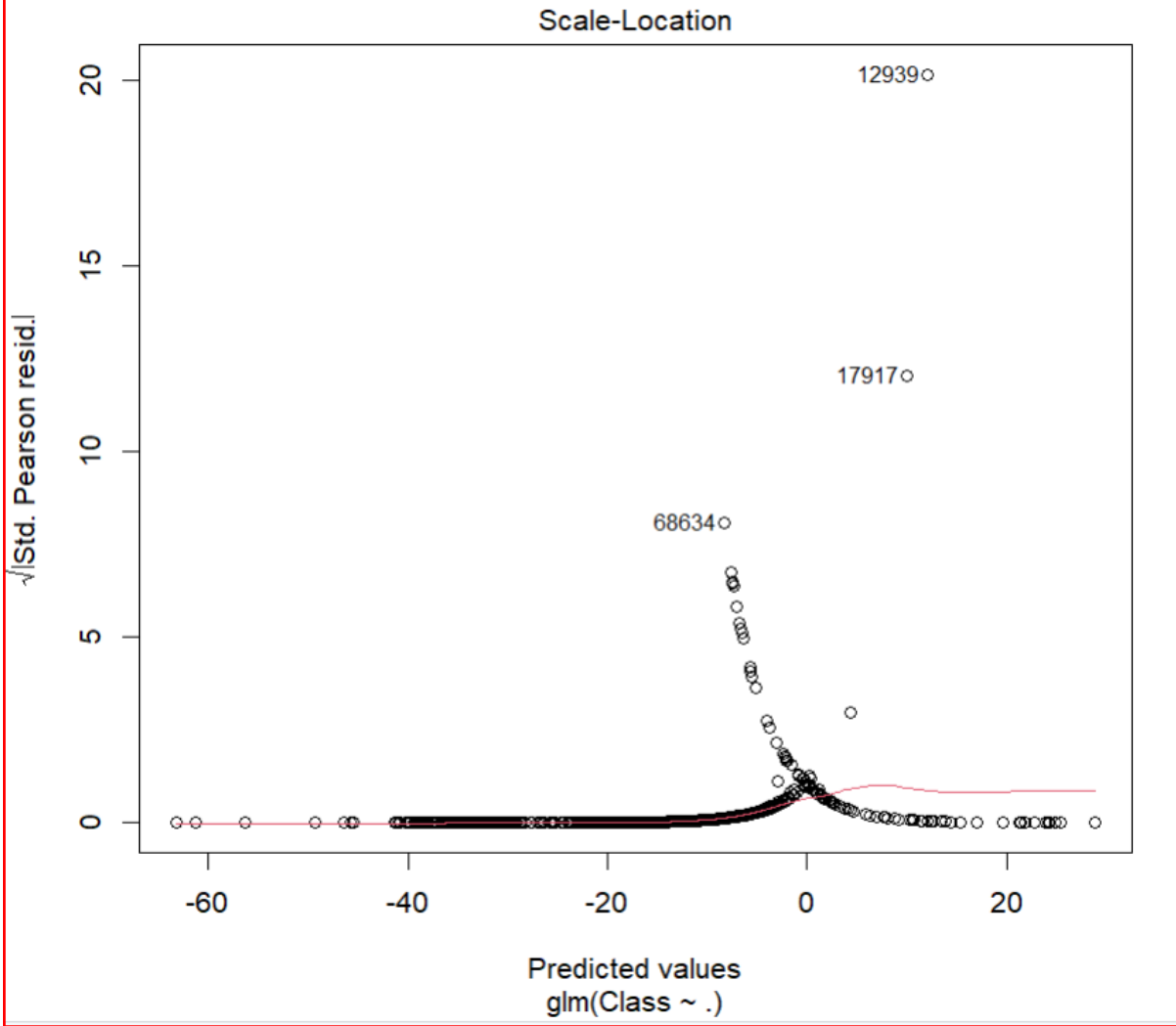


```
> # Model the data to train data and test data
> set.seed(123)
> data_sample = sample.split(NewData$Class,SplitRatio=0.80)
> train_data = subset(NewData,data_sample==TRUE)
> test_data = subset(NewData,data_sample==FALSE)
> dim(train_data)
[1] 227846     30
> dim(test_data)
[1] 56961    30
> #Fit the model using logistic regression, with family as binomial
> Logistic_Model = glm(Class~., test_data,family = binomial())
Warning message:
glm.fit: fitted probabilities numerically 0 or 1 occurred
> summary(Logistic_Model)

Call:
glm(formula = Class ~ ., family = binomial(), data = test_data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-4.9019  -0.0254  -0.0156  -0.0078   4.0877

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -12.52800   10.30537  -1.216   0.2241
V1           -0.17299    1.27381  -0.136   0.8920
V2            1.44512    4.23062   0.342   0.7327
V3            0.17897    0.24058   0.744   0.4569
V4            3.13593    7.17768   0.437   0.6622
V5            1.49014    3.80369   0.392   0.6952
V6           -0.12428    0.22202  -0.560   0.5756
V7            1.40903    4.22644   0.333   0.7388
V8           -0.35254    0.17462  -2.019   0.0435 *
V9            3.02176    8.67262   0.348   0.7275
V10          -2.89571    6.62383  -0.437   0.6620
V11          -0.09769    0.28270  -0.346   0.7297
V12           1.97992    6.56699   0.301   0.7630
V13          -0.71674    1.25649  -0.570   0.5684
V14           0.19316    3.28868   0.059   0.9532
V15           1.03868    2.89256   0.359   0.7195
V16          -2.98194    7.11391  -0.419   0.6751
V17          -1.81809    4.99764  -0.364   0.7160
V18           2.74772    8.13188   0.338   0.7354
V19          -1.63246    4.77228  -0.342   0.7323
V20          -0.69925    1.15114  -0.607   0.5436
V21          -0.45082    1.99182  -0.226   0.8209
V22          -1.40395    5.18980  -0.271   0.7868
V23           0.19026    0.61195   0.311   0.7559
V24          -0.12889    0.44701  -0.288   0.7731
V25          -0.57835    1.94988  -0.297   0.7668
V26           2.65938    9.34957   0.284   0.7761
V27          -0.45396    0.81502  -0.557   0.5775
V28          -0.06639    0.35730  -0.186   0.8526
Amount        0.22576    0.71892   0.314   0.7535
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1443.40  on 56960  degrees of freedom
Residual deviance:  378.59  on 56931  degrees of freedom
AIC: 438.59

Number of Fisher Scoring iterations: 17
```

```
> plot(Logistic_Model)
```



```
# Make predictions on the test data
lr.predict <- predict(Logistic_Model,train_data, probability = TRUE)
```




```
# Load required libraries for modeling
library(glmnet)
library(ROCR)

Logistic_Model = glm(Class~., train_data,family = binomial())
summary(Logistic_Model)
```

```
> Logistic_Model = glm(Class~., train_data,family = binomial())
> summary(Logistic_Model)

Call:
glm(formula = Class ~ ., family = binomial(), data = train_data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-4.6108  -0.0292  -0.0194  -0.0125   4.6021

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -8.651305   0.160212 -53.999  < 2e-16 ***
V1           0.072540   0.044144   1.643 0.100332
V2           0.014818   0.059777   0.248 0.804220
V3           0.026109   0.049776   0.525 0.599906
V4           0.681286   0.078071   8.726  < 2e-16 ***
V5           0.087938   0.071553   1.229 0.219079
V6          -0.148083   0.085192  -1.738 0.082170 .
V7          -0.117344   0.068940  -1.702 0.088731 .
V8          -0.146045   0.035667  -4.095 4.23e-05 ***
V9          -0.339828   0.117595  -2.890 0.003855 **
V10         -0.785462   0.098486  -7.975 1.52e-15 ***
V11          0.001492   0.085147   0.018 0.986018
V12          0.087106   0.094869   0.918 0.358532
V13         -0.343792   0.092381  -3.721 0.000198 ***
V14         -0.526828   0.067084  -7.853 4.05e-15 ***
V15         -0.095471   0.094037  -1.015 0.309991
V16         -0.130225   0.138629  -0.939 0.347537
V17          0.032463   0.074471   0.436 0.662900
V18         -0.100964   0.140985  -0.716 0.473909
V19          0.083711   0.105134   0.796 0.425897
V20         -0.463946   0.081871  -5.667 1.46e-08 ***
V21          0.381206   0.065880   5.786 7.19e-09 ***
V22          0.610874   0.142086   4.299 1.71e-05 ***
V23         -0.071406   0.058799  -1.214 0.224589
V24          0.255791   0.170568   1.500 0.133706
V25         -0.073955   0.142634  -0.519 0.604109
V26          0.120841   0.202553   0.597 0.550783
V27         -0.852018   0.118391  -7.197 6.17e-13 ***
V28         -0.323854   0.090075  -3.595 0.000324 ***
Amount       0.292477   0.092075   3.177 0.001491 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 5799.1  on 227845  degrees of freedom
Residual deviance: 1790.9  on 227816  degrees of freedom
AIC: 1850.9

Number of Fisher Scoring iterations: 12
```

```
> # Convert probabilities to class labels (0 or 1)
> predicted_class <- ifelse(lr.predict > 0.5, 1, 0)
> # Calculate the confusion matrix
> confusion_matrix <- table(predicted_class, test_data$Class)
> # Calculate precision, recall, and F1-score
> precision <- confusion_matrix[2, 2] / sum(confusion_matrix[, 2])
> recall <- confusion_matrix[2, 2] / sum(confusion_matrix[2, ])
> f1_score <- 2 * (precision * recall) / (precision + recall)
> # Print the results of evaluation metrics
> cat("Precision:", precision, "\n")
Precision: 0.7040816
> cat("Recall:", recall, "\n")
Recall: 0.92
> cat("F1-score:", f1_score, "\n")
F1-score: 0.7976879
>
```