Fundamentals of Machine Learning - Final Exam

2022-11-27

```
library("stats")
library("factoextra")
```

```
## Loading required package: ggplot2
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

```
library("cluster")
library("fpc")
library("tidyverse")
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
```

```
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.3      v forcats 0.5.2
## v purrr   0.3.4
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library("caret")
```

```
## Loading required package: lattice
##
## Attaching package: 'caret'
##
## The following object is masked from 'package:purrr':
##
##     lift
```

```
library("corrplot")
```

```
## corrplot 0.92 loaded
```

```
library("ROSE")
```

```
## Loaded ROSE 0.0-4
```

```
library("Rtsne")
library("pROC")
```

```
## Type 'citation("pROC")' for a citation.
##
## Attaching package: 'pROC'
##
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
library("dplyr")
library("caTools")
library("rpart")
library("rpart.plot")
```

```
rm(list=ls())
creditcard <- read.csv("creditcard.csv", header = TRUE)
```

```
glimpse(creditcard)
```

```
## Rows: 284,807
## Columns: 31
## $ Time   <dbl> 0, 0, 1, 1, 2, 2, 4, 7, 7, 9, 10, 10, 10, 11, 12, 12, 12, 13, 1~
## $ V1     <dbl> -1.3598071, 1.1918571, -1.3583541, -0.9662717, -1.1582331, -0.4~
## $ V2     <dbl> -0.07278117, 0.26615071, -1.34016307, -0.18522601, 0.87773675, ~
## $ V3     <dbl> 2.53634674, 0.16648011, 1.77320934, 1.79299334, 1.54871785, 1.1~
## $ V4     <dbl> 1.37815522, 0.44815408, 0.37977959, -0.86329128, 0.40303393, -0~
## $ V5     <dbl> -0.33832077, 0.06001765, -0.50319813, -0.01030888, -0.40719338,~
## $ V6     <dbl> 0.46238778, -0.08236081, 1.80049938, 1.24720317, 0.09592146, -0~
## $ V7     <dbl> 0.239598554, -0.078802983, 0.791460956, 0.237608940, 0.59294074~
## $ V8     <dbl> 0.098697901, 0.085101655, 0.247675787, 0.377435875, -0.27053267~
## $ V9     <dbl> 0.3637870, -0.2554251, -1.5146543, -1.3870241, 0.8177393, -0.56~
## $ V10    <dbl> 0.09079417, -0.16697441, 0.20764287, -0.05495192, 0.75307443, -~
## $ V11    <dbl> -0.55159953, 1.61272666, 0.62450146, -0.22648726, -0.82284288, ~
## $ V12    <dbl> -0.61780086, 1.06523531, 0.06608369, 0.17822823, 0.53819555, 0.~
## $ V13    <dbl> -0.99138985, 0.48909502, 0.71729273, 0.50775687, 1.34585159, -0~
## $ V14    <dbl> -0.31116935, -0.14377230, -0.16594592, -0.28792375, -1.11966983~
## $ V15    <dbl> 1.468176972, 0.635558093, 2.345864949, -0.631418118, 0.17512113~
## $ V16    <dbl> -0.47040053, 0.46391704, -2.89008319, -1.05964725, -0.45144918,~
## $ V17    <dbl> 0.207971242, -0.114804663, 1.109969379, -0.684092786, -0.237033~
## $ V18    <dbl> 0.02579058, -0.18336127, -0.12135931, 1.96577500, -0.03819479, ~
## $ V19    <dbl> 0.40399296, -0.14578304, -2.26185710, -1.23262197, 0.80348692, ~
## $ V20    <dbl> 0.25141210, -0.06908314, 0.52497973, -0.20803778, 0.40854236, 0~
## $ V21    <dbl> -0.018306778, -0.225775248, 0.247998153, -0.108300452, -0.00943~
## $ V22    <dbl> 0.277837576, -0.638671953, 0.771679402, 0.005273597, 0.79827849~
## $ V23    <dbl> -0.110473910, 0.101288021, 0.909412262, -0.190320519, -0.137458~
## $ V24    <dbl> 0.06692807, -0.33984648, -0.68928096, -1.17557533, 0.14126698, ~
## $ V25    <dbl> 0.12853936, 0.16717040, -0.32764183, 0.64737603, -0.20600959, -~
## $ V26    <dbl> -0.18911484, 0.12589453, -0.13909657, -0.22192884, 0.50229222, ~
## $ V27    <dbl> 0.133558377, -0.008983099, -0.055352794, 0.062722849, 0.2194222~
## $ V28    <dbl> -0.021053053, 0.014724169, -0.059751841, 0.061457629, 0.2151531~
```

```
## $ Amount <dbl> 149.62, 2.69, 378.66, 123.50, 69.99, 3.67, 4.99, 40.80, 93.20, ~
## $ Class  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
```

```
summary(creditcard)
```

```
##       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.:139320   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  
##  Max.   :  7.8484   Max.   : 7.12688   Max.   : 10.5268   Max.   : 8.87774  
##       V16                 V17                 V18                
##  Min.   :-14.12985   Min.   :-25.16280   Min.   :-9.498746  
##  1st Qu.: -0.46804   1st Qu.: -0.48375   1st Qu.:-0.498850  
##  Median :  0.06641   Median : -0.06568   Median :-0.003636  
##  Mean   :  0.00000   Mean   :  0.00000   Mean   : 0.000000  
##  3rd Qu.:  0.52330   3rd Qu.:  0.39968   3rd Qu.: 0.500807  
##  Max.   : 17.31511   Max.   :  9.25353   Max.   : 5.041069  
##       V19                 V20                 V21                
##  Min.   :-7.213527   Min.   :-54.49772   Min.   :-34.83038  
##  1st Qu.:-0.456299   1st Qu.: -0.21172   1st Qu.: -0.22839  
##  Median : 0.003735   Median : -0.06248   Median : -0.02945  
##  Mean   : 0.000000   Mean   :  0.00000   Mean   :  0.00000  
##  3rd Qu.: 0.458949   3rd Qu.:  0.13304   3rd Qu.:  0.18638  
##  Max.   : 5.591971   Max.   : 39.42090   Max.   : 27.20284  
##       V22                 V23                 V24                
##  Min.   :-10.933144   Min.   :-44.80774   Min.   :-2.83663  
##  1st Qu.: -0.542350   1st Qu.: -0.16185   1st Qu.:-0.35459  
##  Median :  0.006782   Median : -0.01119   Median : 0.04098  
##  Mean   :  0.000000   Mean   :  0.00000   Mean   : 0.00000  
##  3rd Qu.:  0.528554   3rd Qu.:  0.14764   3rd Qu.: 0.43953  
##  Max.   : 10.503090   Max.   : 22.52841   Max.   : 4.58455
```

```
##       V25                 V26                 V27
##  Min.   :-10.29540   Min.   :-2.60455   Min.   :-22.565679
##  1st Qu.: -0.31715   1st Qu.:-0.32698   1st Qu.: -0.070840
##  Median :  0.01659   Median :-0.05214   Median :  0.001342
##  Mean   :  0.00000   Mean   : 0.00000   Mean   :  0.000000
##  3rd Qu.:  0.35072   3rd Qu.: 0.24095   3rd Qu.:  0.091045
##  Max.   :  7.51959   Max.   : 3.51735   Max.   : 31.612198
##       V28                 Amount              Class
##  Min.   :-15.43008   Min.   :    0.00   Min.   :0.000000
##  1st Qu.: -0.05296   1st Qu.:    5.60   1st Qu.:0.000000
##  Median :  0.01124   Median :   22.00   Median :0.000000
##  Mean   :  0.00000   Mean   :   88.35   Mean   :0.001728
##  3rd Qu.:  0.07828   3rd Qu.:   77.17   3rd Qu.:0.000000
##  Max.   : 33.84781   Max.   :25691.16   Max.   :1.000000
```

```
length(creditcard$Class)
```

```
## [1] 284807
```

```
summary(as.factor(creditcard$Class))
```

```
##      0      1
## 284315    492
```

```
table(creditcard$Class)
```

```
##
##      0      1
## 284315    492
```

```
#V31 is the Class section
#0 is a legit transaction
#1 is a fraudulent transaction
492/284807
```

```
## [1] 0.001727486
```

```
names(creditcard)
```

```
##  [1] "Time"   "V1"     "V2"     "V3"     "V4"     "V5"     "V6"     "V7"
##  [9] "V8"     "V9"     "V10"    "V11"    "V12"    "V13"    "V14"    "V15"
## [17] "V16"    "V17"    "V18"    "V19"    "V20"    "V21"    "V22"    "V23"
## [25] "V24"    "V25"    "V26"    "V27"    "V28"    "Amount" "Class"
```

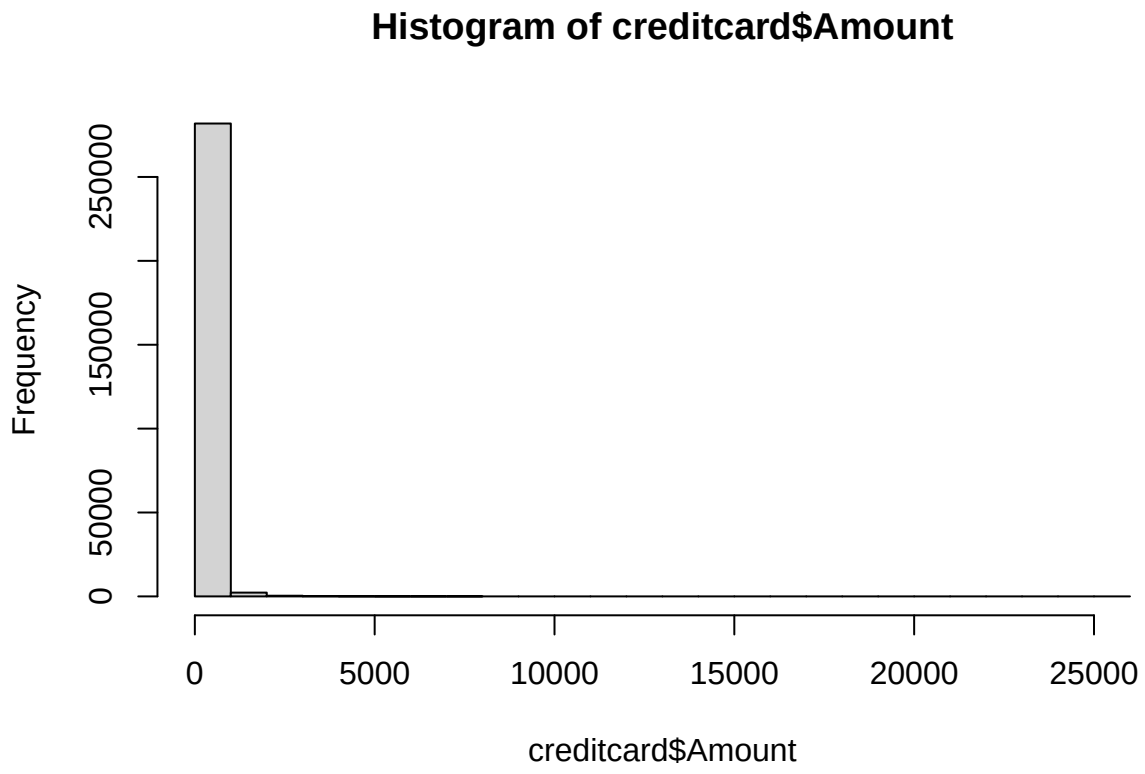
```
colSums(is.na(creditcard))
```

```
##   Time     V1     V2     V3     V4     V5     V6     V7     V8     V9    V10
##      0      0      0      0      0      0      0      0      0      0      0
##    V11    V12    V13    V14    V15    V16    V17    V18    V19    V20    V21
##      0      0      0      0      0      0      0      0      0      0      0
##    V22    V23    V24    V25    V26    V27    V28 Amount  Class
##      0      0      0      0      0      0      0      0      0
```

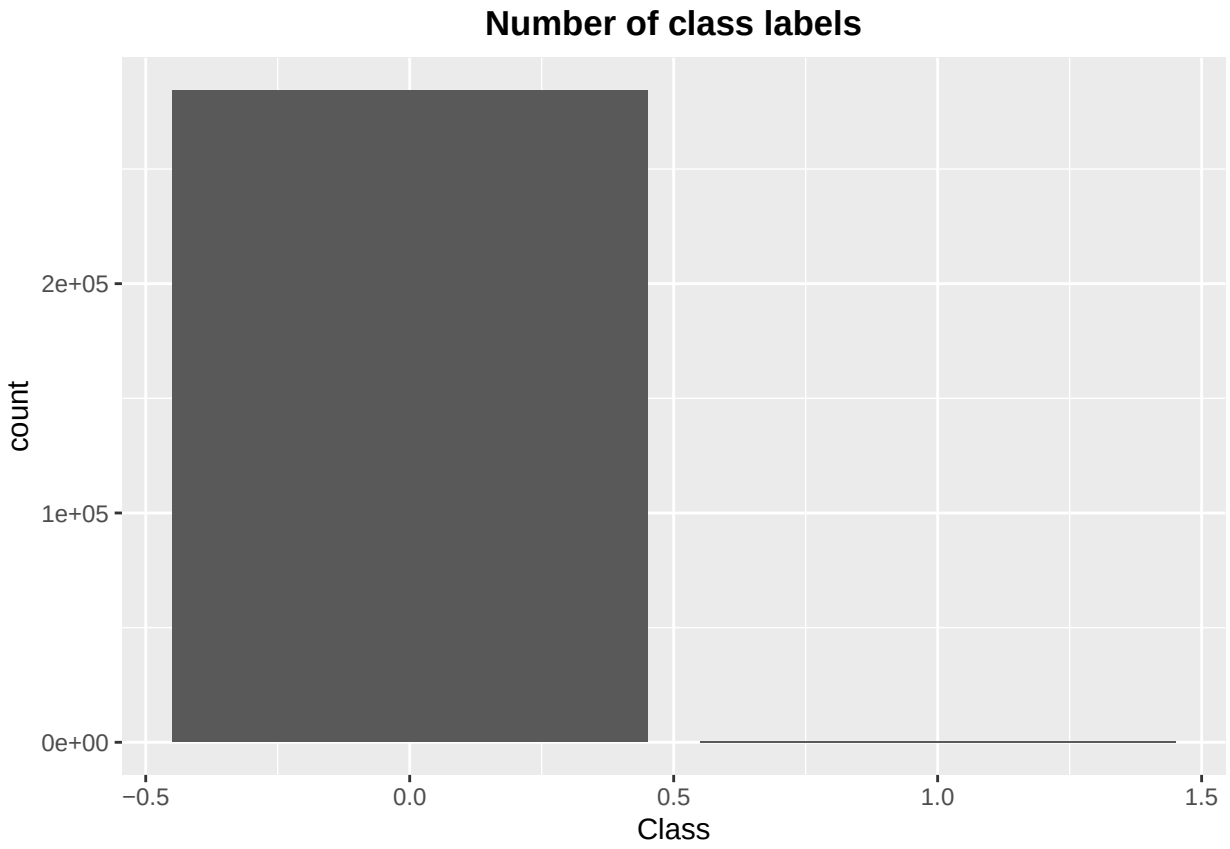
```
summary(creditcard$Amount)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##     0.00     5.60    22.00    88.35    77.17 25691.16
```

```
hist(creditcard$Amount)
```

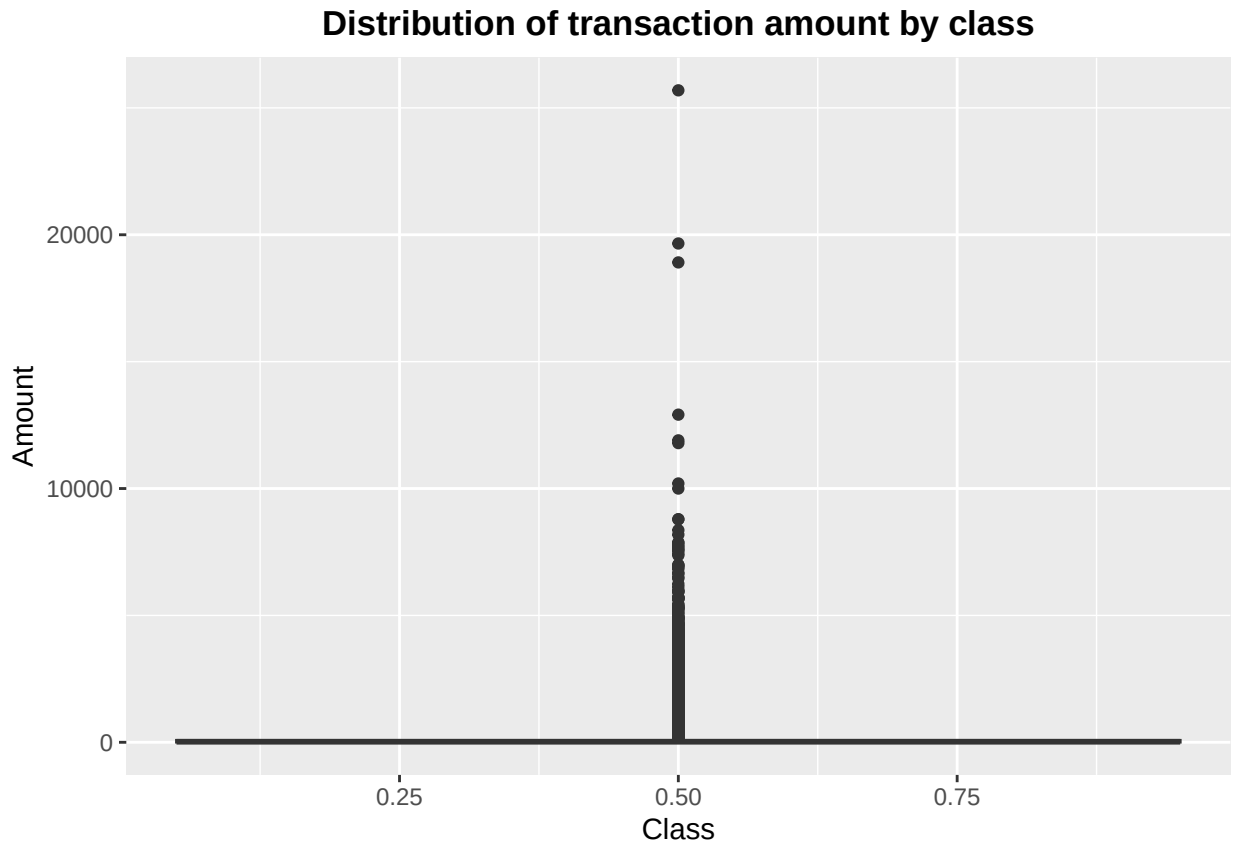



```
GGPlotTheme <- theme(plot.title = element_text(hjust = 0.5, face = "bold"))
p <- ggplot(creditcard, aes(x = Class)) + geom_bar() + ggtitle("Number of class labels") + GGPlotTheme
print(p)
```

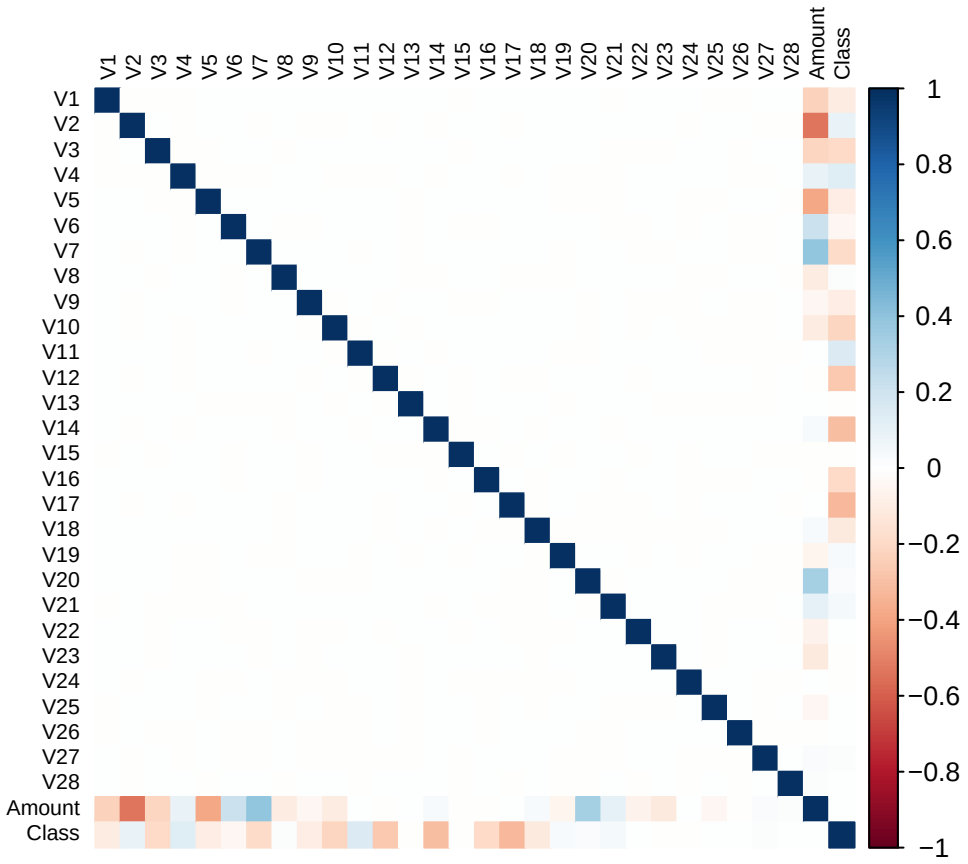



```
p <- ggplot(creditcard, aes(x = Class, y = Amount)) + geom_boxplot() + ggtitle("Distribution of transac
print(p)
```

```
## Warning: Continuous x aesthetic
## i did you forget `aes(group = ...)`?
```




```
correlation <- cor(creditcard[, -1], method = "pearson")
corrplot(correlation, number.cex = 1, method = "color", type = "full", tl.cex=0.7, tl.col="black")
```




```
creditcard$Amount <- scale(creditcard$Amount)
new_data <- creditcard[, -c(1)]
head(new_data)
```

```
##           V1          V2        V3         V4          V5          V6
## 1 -1.3598071 -0.07278117 2.5363467  1.3781552 -0.33832077  0.46238778
## 2  1.1918571  0.26615071 0.1664801  0.4481541  0.06001765 -0.08236081
## 3 -1.3583541 -1.34016307 1.7732093  0.3797796 -0.50319813  1.80049938
## 4 -0.9662717 -0.18522601 1.7929933 -0.8632913 -0.01030888  1.24720317
## 5 -1.1582331  0.87773675 1.5487178  0.4030339 -0.40719338  0.09592146
## 6 -0.4259659  0.96052304 1.1411093 -0.1682521  0.42098688 -0.02972755
##            V7          V8         V9         V10        V11         V12
## 1  0.23959855  0.09869790  0.3637870  0.09079417 -0.5515995 -0.61780086
## 2 -0.07880298  0.08510165 -0.2554251 -0.16697441  1.6127267  1.06523531
## 3  0.79146096  0.24767579 -1.5146543  0.20764287  0.6245015  0.06608369
## 4  0.23760894  0.37743587 -1.3870241 -0.05495192 -0.2264873  0.17822823
## 5  0.59294075 -0.27053268  0.8177393  0.75307443 -0.8228429  0.53819555
## 6  0.47620095  0.26031433 -0.5686714 -0.37140720  1.3412620  0.35989384
##           V13        V14        V15        V16         V17         V18
## 1 -0.9913898 -0.3111694  1.4681770 -0.4704005  0.20797124  0.02579058
## 2  0.4890950 -0.1437723  0.6355581  0.4639170 -0.11480466 -0.18336127
## 3  0.7172927 -0.1659459  2.3458649 -2.8900832  1.10996938 -0.12135931
## 4  0.5077569 -0.2879237 -0.6314181 -1.0596472 -0.68409279  1.96577500
## 5  1.3458516 -1.1196698  0.1751211 -0.4514492 -0.23703324 -0.03819479
## 6 -0.3580907 -0.1371337  0.5176168  0.4017259 -0.05813282  0.06865315
```

```
##           V19         V20          V21          V22         V23         V24
## 1  0.40399296  0.25141210 -0.018306778  0.277837576 -0.11047391  0.06692807
## 2 -0.14578304 -0.06908314 -0.225775248 -0.638671953  0.10128802 -0.33984648
## 3 -2.26185710  0.52497973  0.247998153  0.771679402  0.90941226 -0.68928096
## 4 -1.23262197 -0.20803778 -0.108300452  0.005273597 -0.19032052 -1.17557533
## 5  0.80348692  0.40854236 -0.009430697  0.798278495 -0.13745808  0.14126698
## 6 -0.03319379  0.08496767 -0.208253515 -0.559824796 -0.02639767 -0.37142658
##          V25        V26          V27         V28      Amount Class
## 1  0.1285394 -0.1891148  0.133558377 -0.02105305  0.24496383     0
## 2  0.1671704  0.1258945 -0.008983099  0.01472417 -0.34247394     0
## 3 -0.3276418 -0.1390966 -0.055352794 -0.05975184  1.16068389     0
## 4  0.6473760 -0.2219288  0.062722849  0.06145763  0.14053401     0
## 5 -0.2060096  0.5022922  0.219422230  0.21515315 -0.07340321     0
## 6 -0.2327938  0.1059148  0.253844225  0.08108026 -0.33855582     0
```

```
new_data$Class <- as.factor(new_data$Class)
levels(new_data$Class) <- c("Not Fraudulent", "Fraudulent")
```

```
set.seed(123)
split <- sample.split(new_data$Class, SplitRatio = 0.8)
train_data <- subset(new_data, split == TRUE)
test_data <- subset(new_data, split == FALSE)
dim(train_data)
```

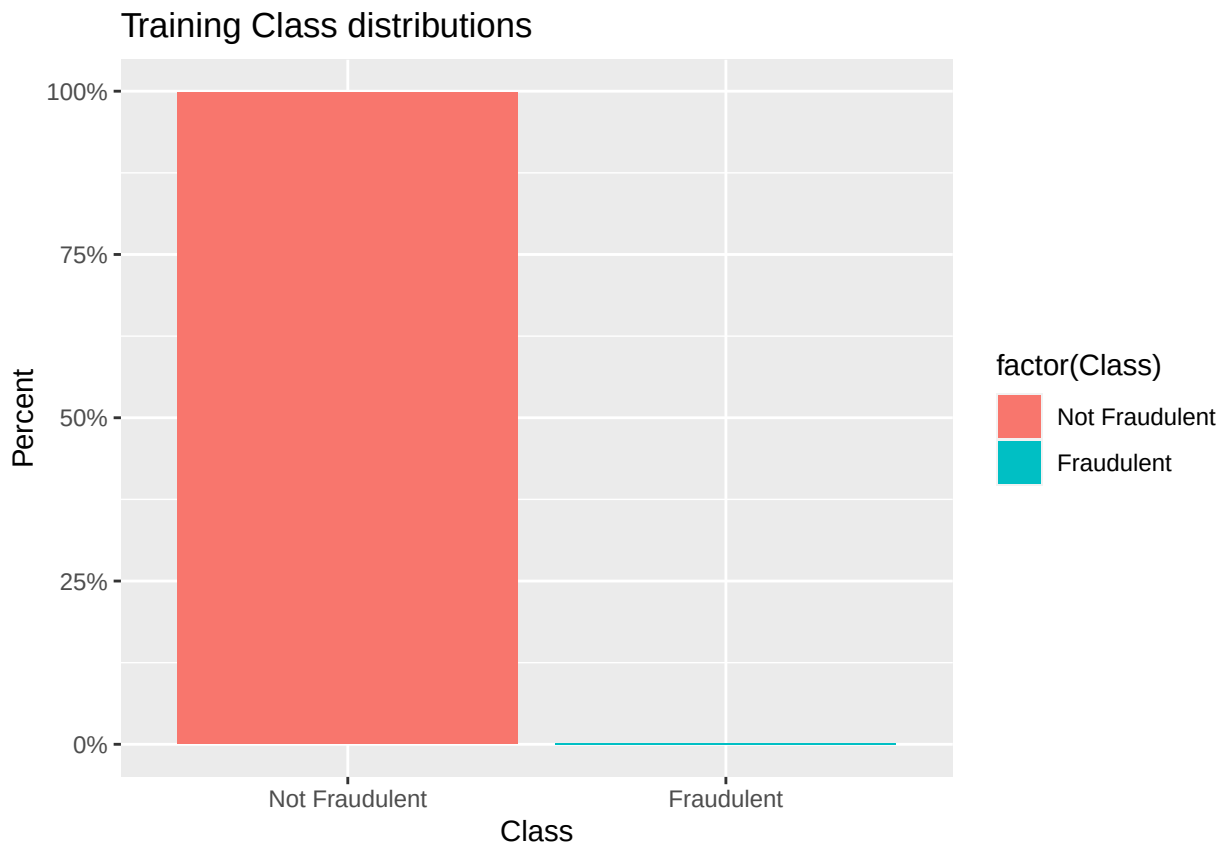
```
## [1] 227846     30
```

```
dim(test_data)
```

```
## [1] 56961    30
```

```
train_data %>% ggplot(aes(x = factor(Class), y = prop.table(stat(count)), fill = factor(Class))) +
  geom_bar(position = "dodge") +
  scale_y_continuous(labels = scales::percent) +
  labs(x = 'Class', y = 'Percent', title = 'Training Class distributions') +
  theme_grey()
```

```
## Warning: `stat(count)` was deprecated in ggplot2 3.4.0.
## i Please use `after_stat(count)` instead.
```




```
Logit_Model=glm(Class~.,test_data,family=binomial())
```

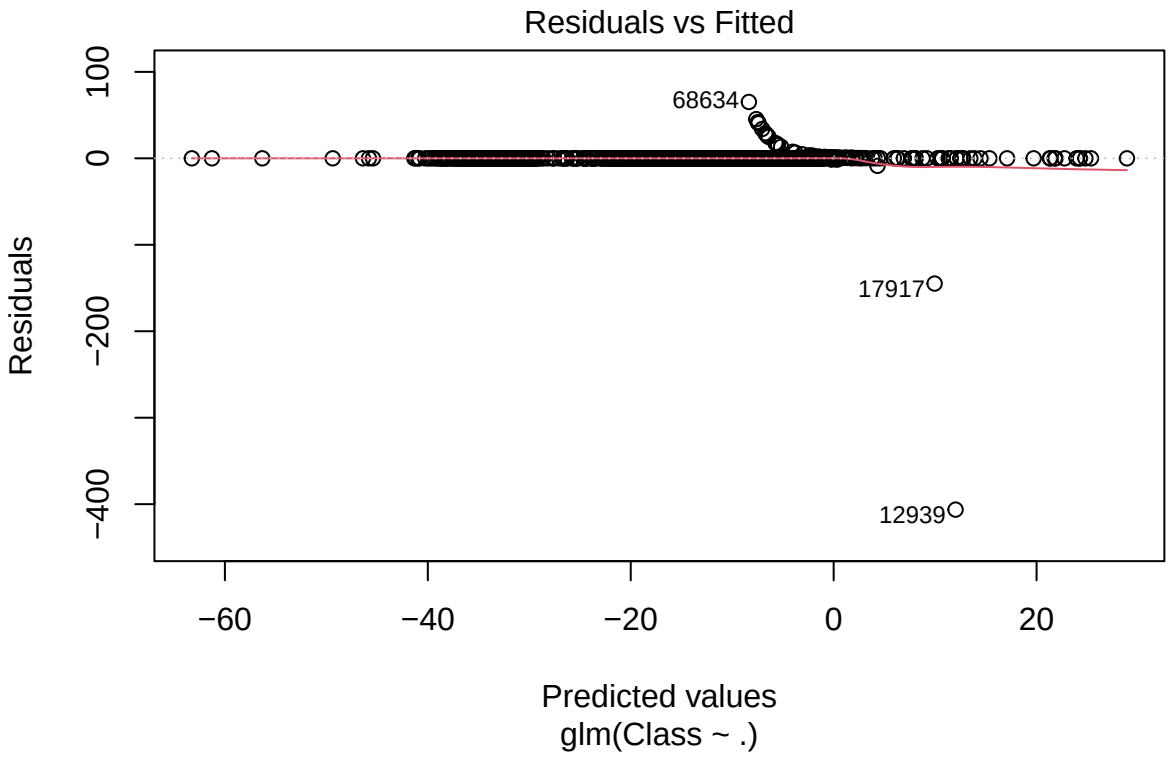
```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

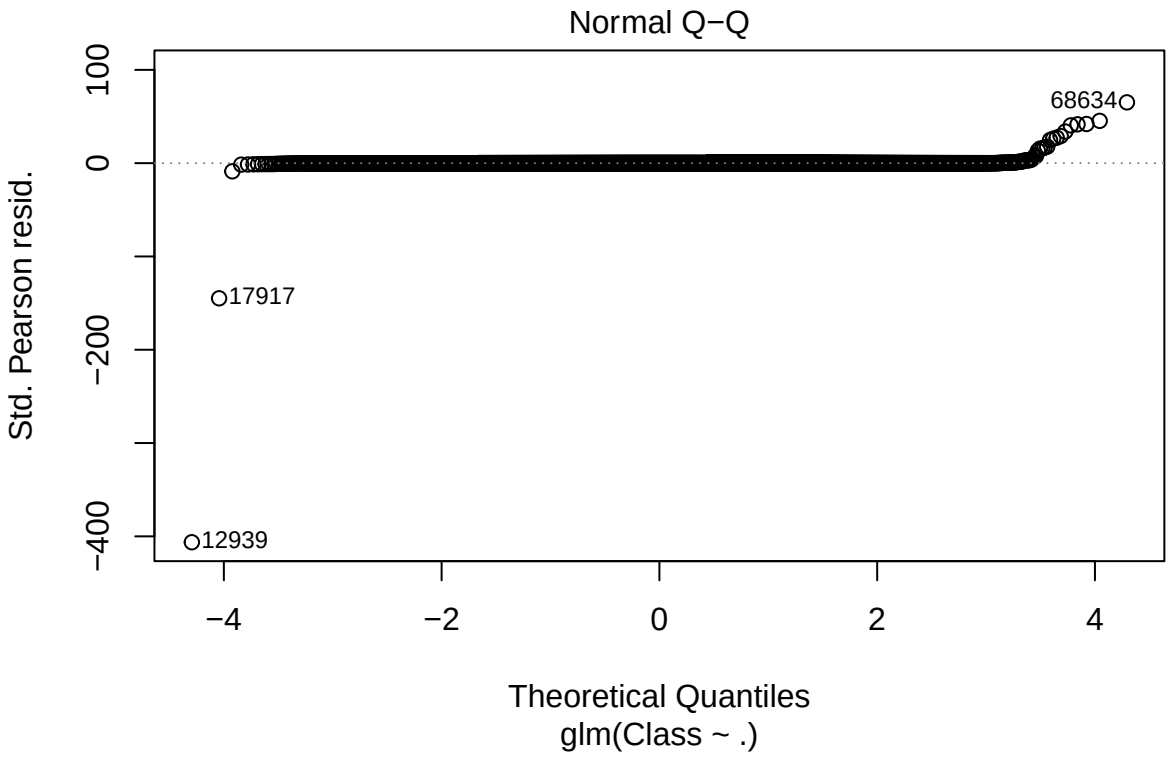
```
summary(Logit_Model)
```

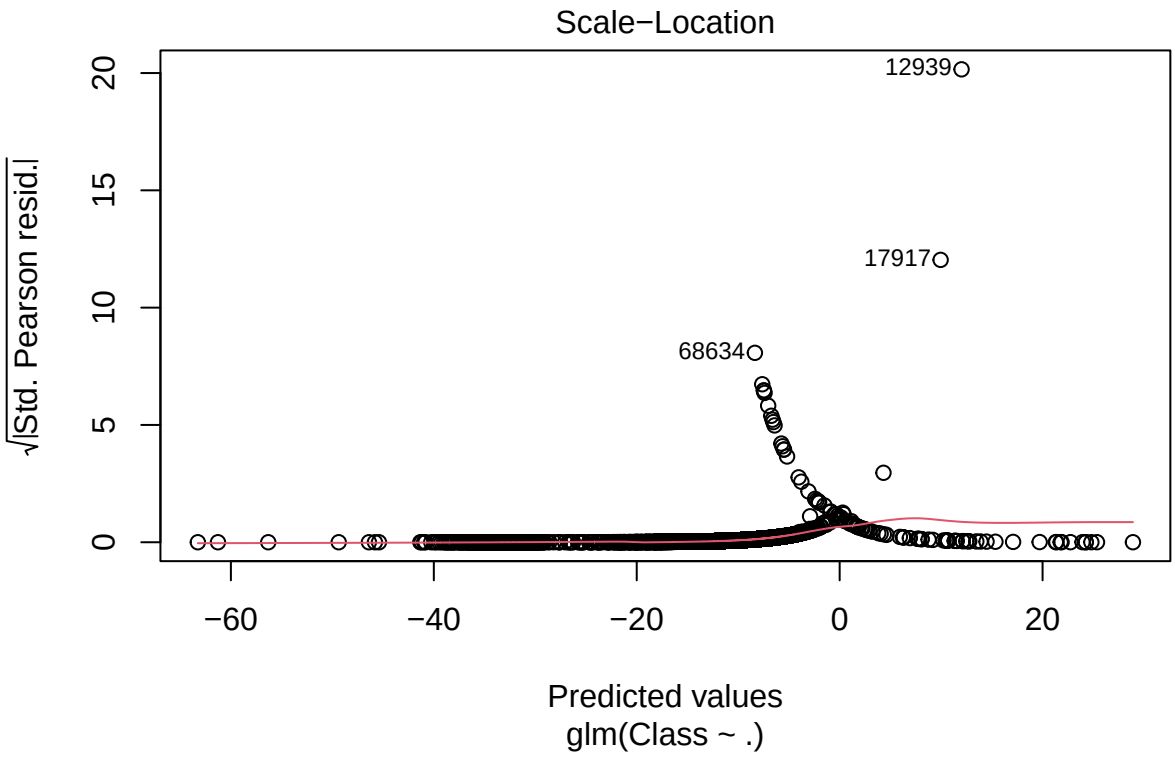
```
##
## Call:
## glm(formula = Class ~ ., family = binomial(), data = test_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -4.9019  -0.0254  -0.0156  -0.0078   4.0877
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -12.52800   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
```

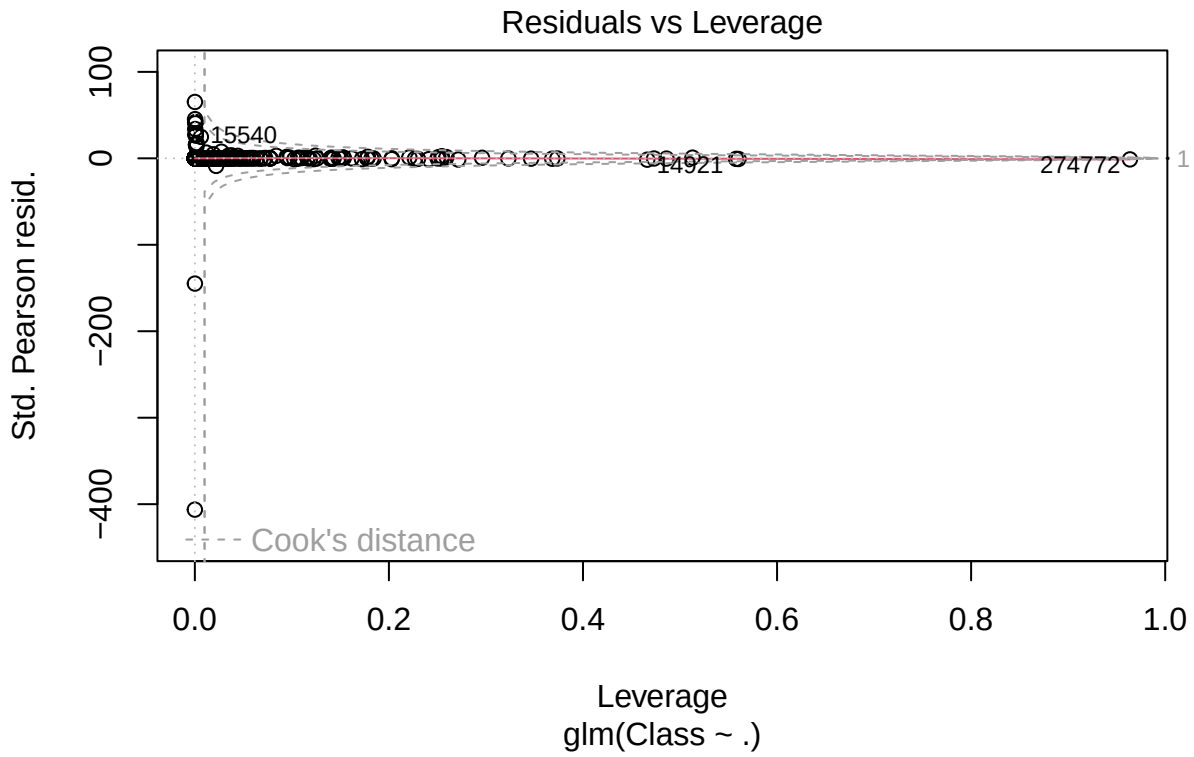
```
plot(Logit_Model)
```


Residuals vs Fitted
Residuals
100
0
−200
−400
68634
17917
12939
−60
−40
−20
0
20
Predicted values
glm(Class ~ .)


Normal Q−Q
Std. Pearson resid.
100
0
−200
−400
68634
17917
12939
−4
−2
0
2
4
Theoretical Quantiles
glm(Class ~ .)



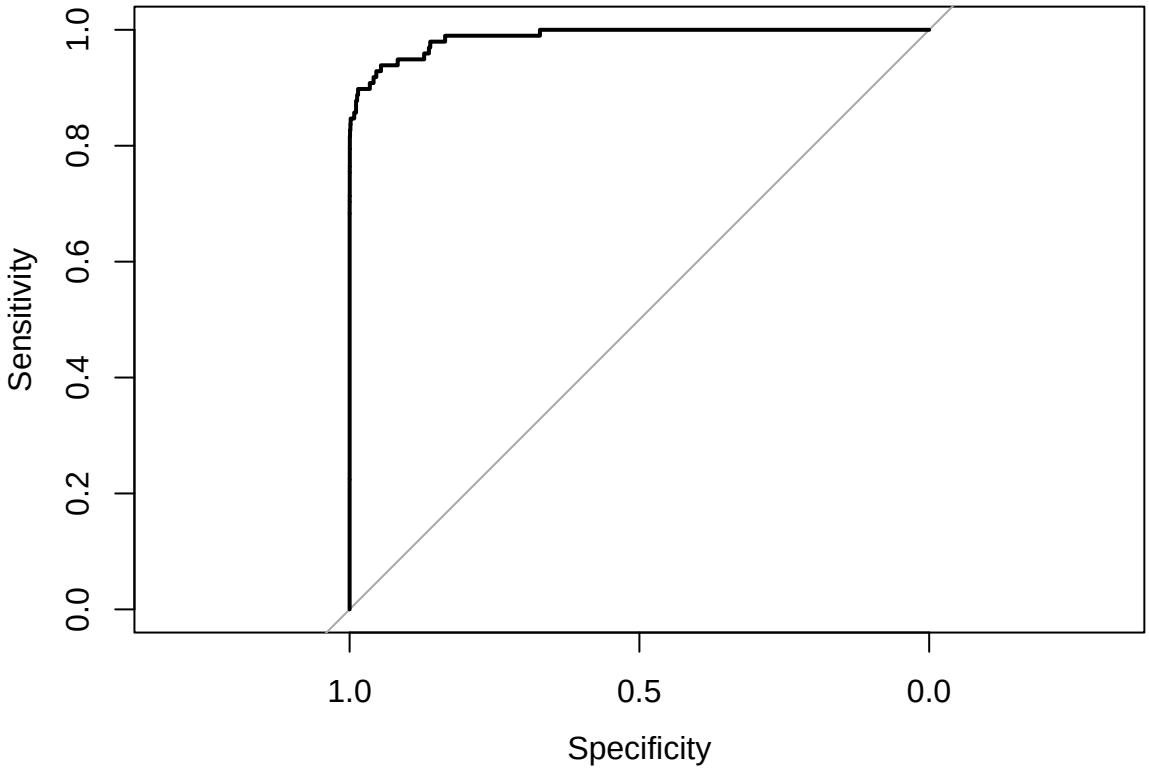

```
## Warning in sqrt(crit * p * (1 - hh)/hh): NaNs produced

## Warning in sqrt(crit * p * (1 - hh)/hh): NaNs produced
```



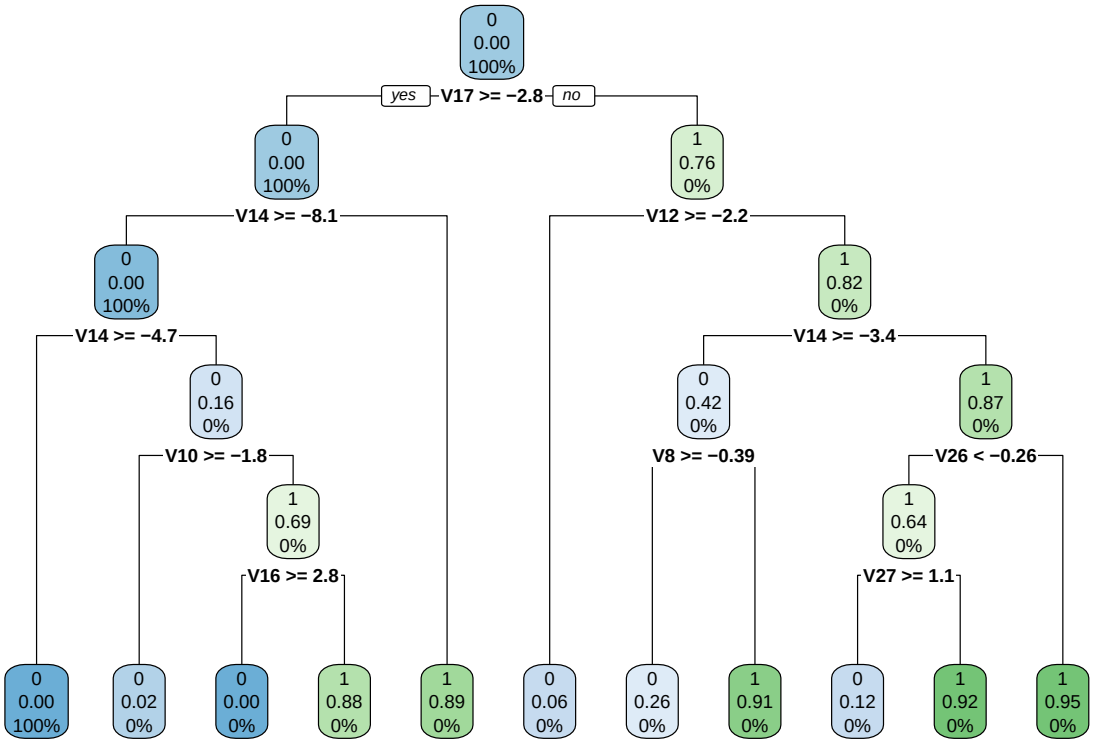

```
lr.predict <- predict(Logit_Model,test_data, probability = TRUE)
auc.gbm = roc(test_data$Class, lr.predict, plot = TRUE, col = "black")
```

```
## Setting levels: control = Not Fraudulent, case = Fraudulent
```

```
## Setting direction: controls < cases
```




```
decisionTree <- rpart(Class ~ . , creditcard, method = 'class')
prediction <- predict(decisionTree, creditcard, type = 'class')
probability <- predict(decisionTree, creditcard, type = 'prob')
rpart.plot(decisionTree)
```


0
0.00
100%
yes
V17 >= −2.8
no
0
0.00
100%
1
0.76
0%
V14 >= −8.1
V12 >= −2.2
0
0.00
100%
1
0.82
0%
V14 >= −4.7
V14 >= −3.4
0
0.16
0%
0
0.42
0%
1
0.87
0%
V10 >= −1.8
V8 >= −0.39
V26 < −0.26
1
0.69
0%
1
0.64
0%
V16 >= 2.8
V27 >= 1.1
0
0.00
100%
0
0.02
0%
0
0.00
0%
1
0.88
0%
1
0.89
0%
0
0.06
0%
0
0.26
0%
1
0.91
0%
0
0.12
0%
1
0.92
0%
1
0.95
0%