Classification

Here is the data used for classification.

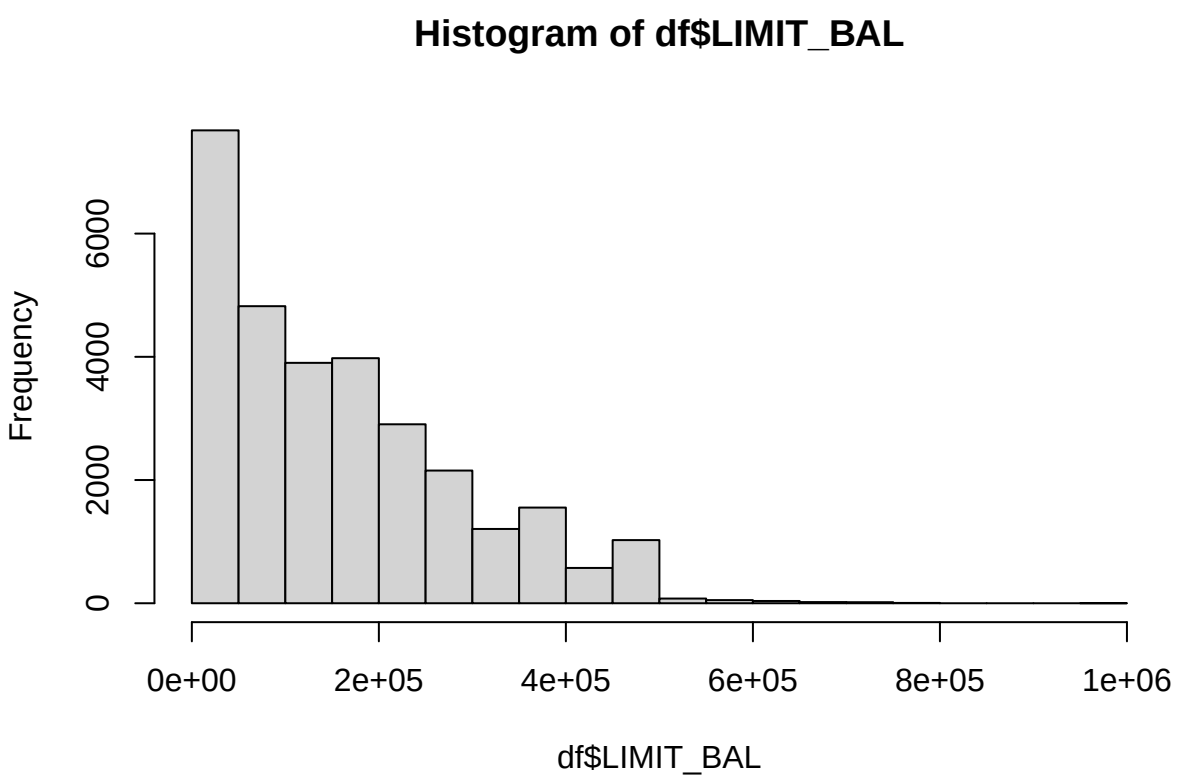
Dividing Data

```
library(readr)

df <- read.csv("credit.csv")

set.seed(1)

sample <- sample(c(TRUE,FALSE), nrow(df), replace = TRUE, prob=c(.8,.2))

train <- df[sample,]
test <-  df[!sample,]
```

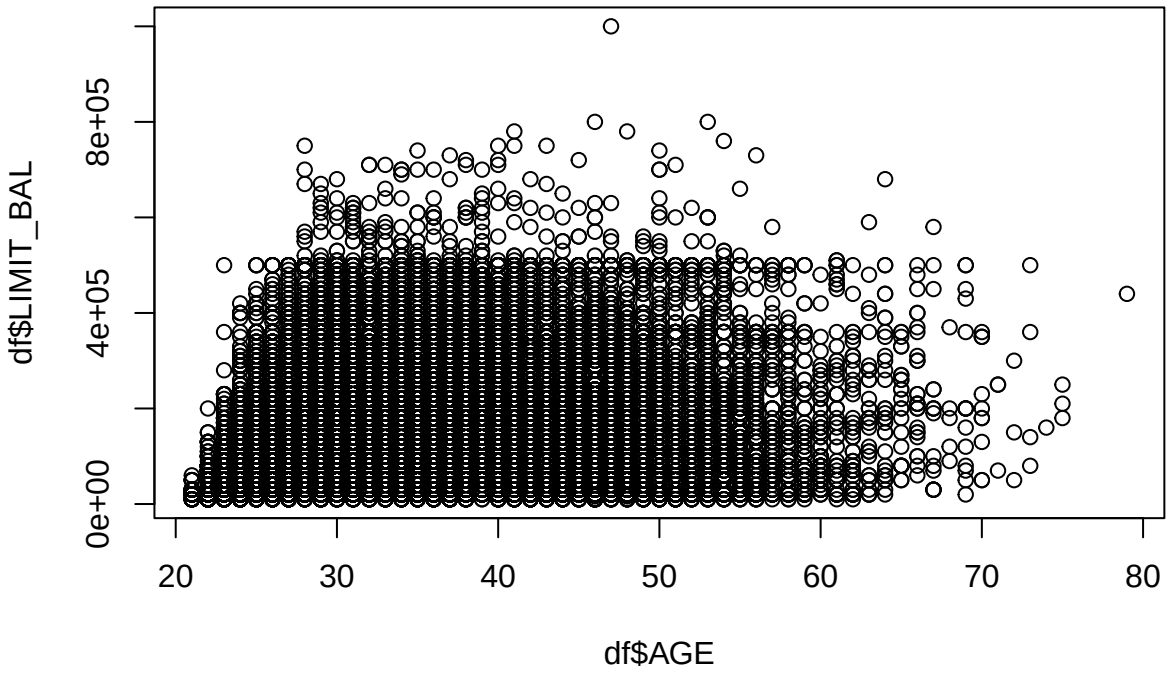
Data exploration

Lets take a look at the data graphically as well as some of the statistics.

```
hist(df$LIMIT_BAL)
```




```
plot(df$AGE, df$LIMIT_BAL)
```




```
cor(df[2:6], use = "complete")
```

```
##              LIMIT_BAL         SEX   EDUCATION    MARRIAGE         AGE
## LIMIT_BAL  1.00000000  0.02475524 -0.21916070 -0.10813941  0.14471280
## SEX        0.02475524  1.00000000  0.01423194 -0.03138884 -0.09087365
## EDUCATION -0.21916070  0.01423194  1.00000000 -0.14346434  0.17506066
## MARRIAGE  -0.10813941 -0.03138884 -0.14346434  1.00000000 -0.41416992
## AGE        0.14471280 -0.09087365  0.17506066 -0.41416992  1.00000000
```

Logistic Regression

```
model <- glm(dpnm~., data = train, family = binomial)
```

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

```
pred <- predict(model, newdata = test, type = "response")
p <- ifelse(pred>.5,1,0)
acc <- mean(p == test$dpnm)
print(paste("accuracy = ", acc))
```

```
## [1] "accuracy =  0.804723885562209"
```

---

kNN classification

```
library(class)
pred2 <- knn(train = train, test = test, cl = train$dpnm, k = 20)
acc2 <- length(which(pred2 == test$dpnm)) / length(pred2)
print(paste("accuracy = ", acc2))
```

```
## [1] "accuracy =  0.777611443779108"
```

Decision Tree

```
library(rpart)
tree <- rpart(dpnm~., data = train, method = "class")
pred3 <- predict(tree, test, type = "class")
table_mat <- table(test$dpnm, pred3)
acc3 <- sum(diag(table_mat)) / sum(table_mat)
print(paste("accuracy = ", acc3))
```

```
## [1] "accuracy =  0.810212907518297"
```

Results

We can see that the three algorithms got roughly the same accuracy, although in order for kNN to get a accuracy close to the other two the value of K needs to be somewhat large which leads to the algorithm running slower. Decision tree had the best accuracy with about a one percent improvement over logistic regression and a three percent improvement over kNN.

Analysis

Logistic regression makes sense at why the accuracy was high since I sort of expected that people default on their credit payments as income decreases and payments decrease as a result as well. I also was not surprised by the non parametric algorithms performing well as the people who default on their credit card debt most likely make similar incomes and pay similar amounts. I was somewhat surprised that DT was the best, I personally thought that Logistic regression would do better but as the improvement was so low it could probably be attributed to possible outliers in the data or maybe over fitting on the part of Logistic regression. kNN being last was not so much of a surprise, i've found that due to the simplicity of it that it often does the worst as it's just too niave.