SVM Classification

I will be using data on credit card debt and income to try and predict if people will default on their credit card debt. Here is the link to the data.https://www.kaggle.com/datasets/mariosfish/default-of-credit-card-clients.

Loading data and divding

First we need to load in the data and divide it into train/test/validate. This file has about 30k entries so I will just take the first 10k so that the SVM algorithm does not take too long to run. I also removed the ID column.

```
library(readr)
df <- read.csv("credit.csv")
df <- df[1:10000,]
df <- subset(df, select = -c(ID))

df$dpnm <- as.factor(df$dpnm)

set.seed(100)
groups <- c(train=.6, test=.2, validate=.2)
i <- sample(cut(1:nrow(df),nrow(df)*cumsum(c(0,groups)), labels=names(groups)))
train <- df[i=="train",]
test <- df[i=="test",]
vald <- df[i=="validate",]
```

Lets take a look at some of the data.

```
library(ggplot2)
head(train)
```

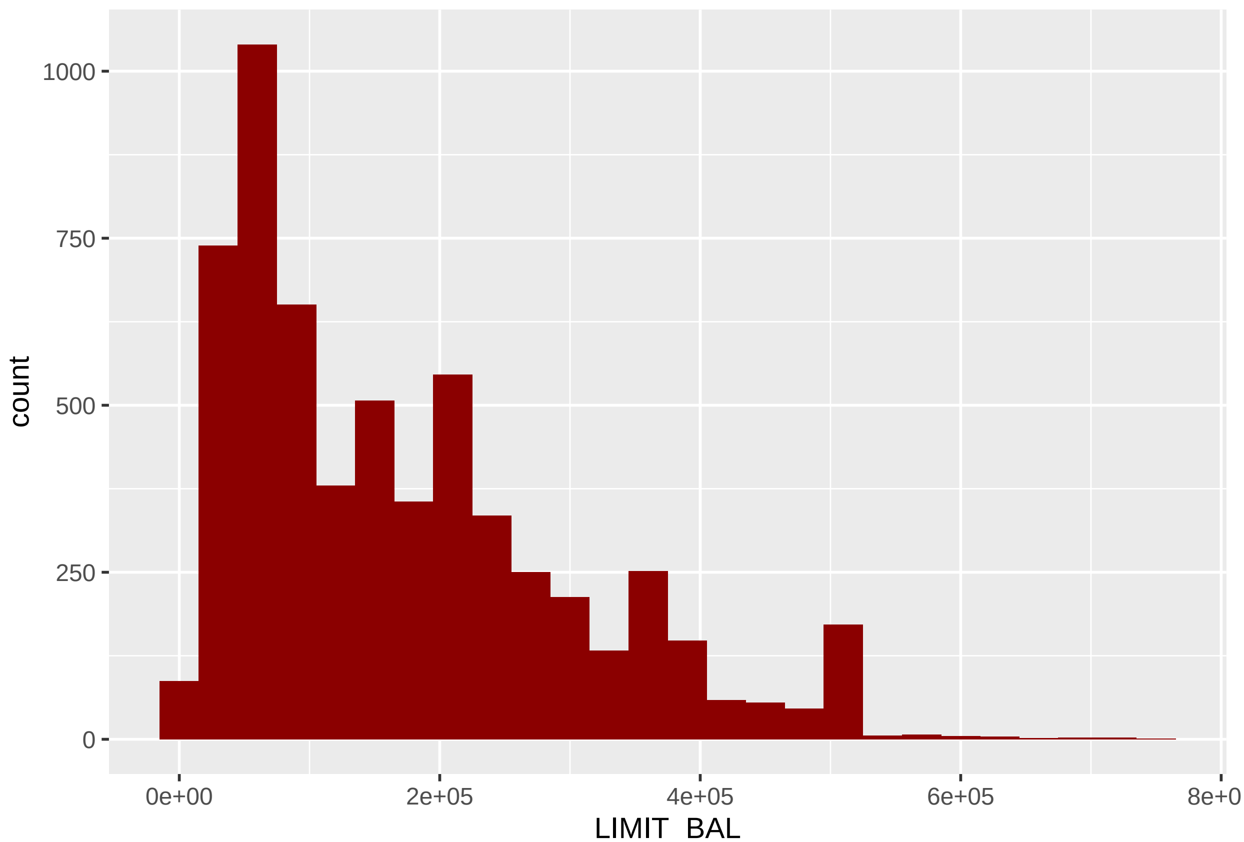
```
##   LIMIT_BAL SEX EDUCATION MARRIAGE AGE PAY_1 PAY_2 PAY_3 PAY_4 PAY_5 PAY_6
## 1     20000   2         2        1  24     2     2    -1    -1    -2    -2
## 2    120000   2         2        2  26    -1     2     0     0     0     2
## 3     90000   2         2        2  34     0     0     0     0     0     0
## 4     50000   2         2        1  37     0     0     0     0     0     0
## 5     50000   1         2        1  57    -1     0    -1     0     0     0
## 7    500000   1         1        2  29     0     0     0     0     0     0
##   BILL_AMT1 BILL_AMT2 BILL_AMT3 BILL_AMT4 BILL_AMT5 BILL_AMT6 PAY_AMT1 PAY_AMT2
## 1      3913      3102       689         0         0         0        0      689
## 2      2682      1725      2682      3272      3455      3261        0     1000
## 3     29239     14027     13559     14331     14948     15549     1518     1500
## 4     46990     48233     49291     28314     28959     29547     2000     2019
## 5      8617      5670     35835     20940     19146     19131     2000    36681
## 7    367965    412023    445007    542653    483003    473944    55000    40000
##   PAY_AMT3 PAY_AMT4 PAY_AMT5 PAY_AMT6 dpnm
## 1        0        0        0        0    1
## 2     1000     1000        0     2000    1
```

```
## 3      1000      1000      1000      5000     0
## 4      1200      1100      1069      1000     0
## 5     10000      9000       689       679     0
## 7     38000     20239     13750     13770     0
```

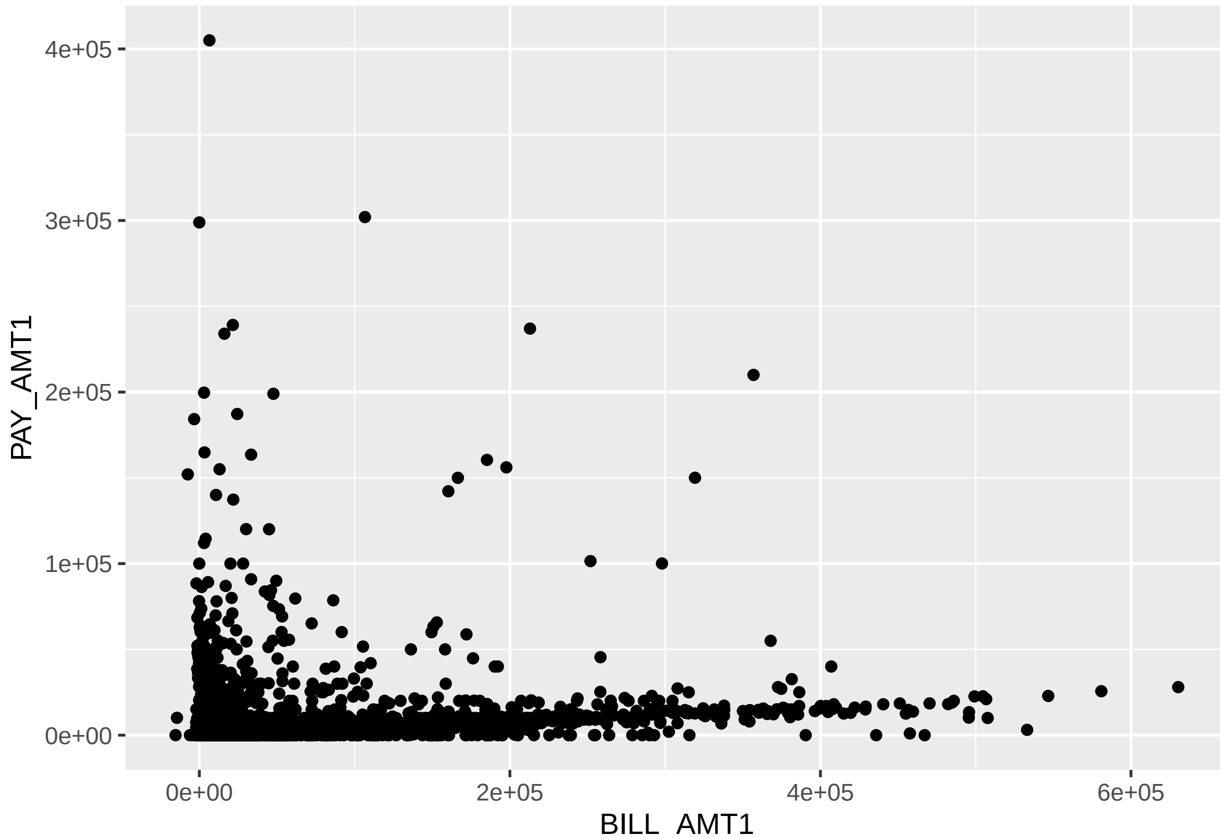
```r
mean(train$LIMIT_BAL)
```

```
## [1] 168442.7
```

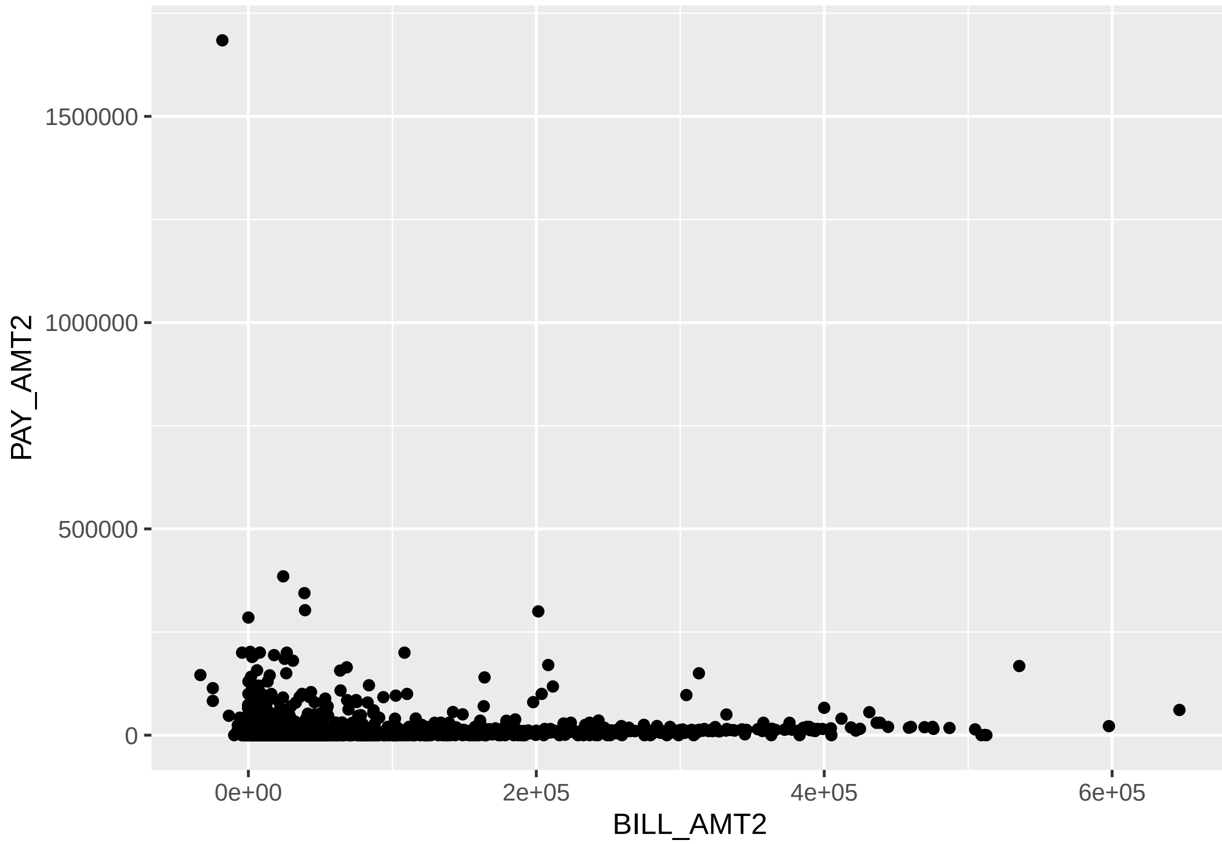
```r
ggplot(train, aes(x = LIMIT_BAL)) + geom_histogram(fill="red4", binwidth = 30000)
```



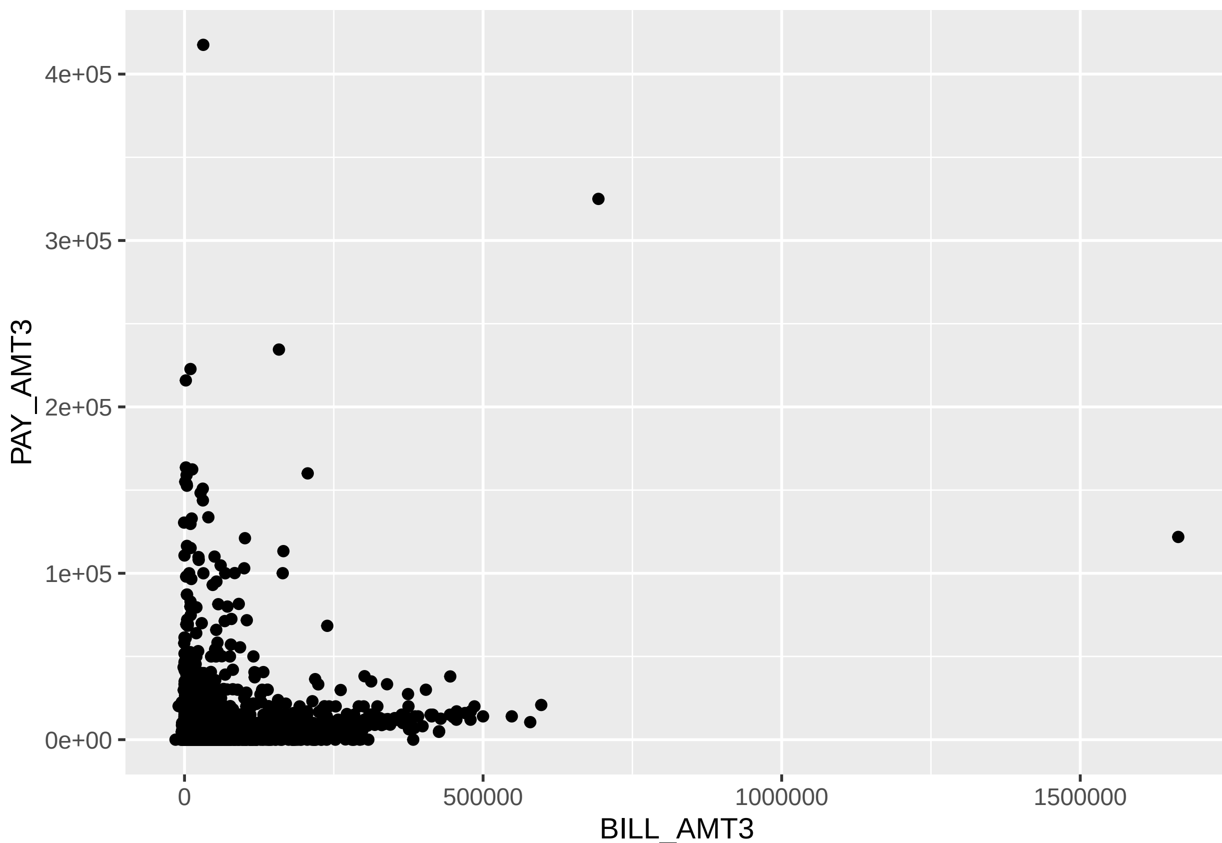

```r
ggplot(train, aes(x = BILL_AMT1, y = PAY_AMT1)) + geom_point()
```



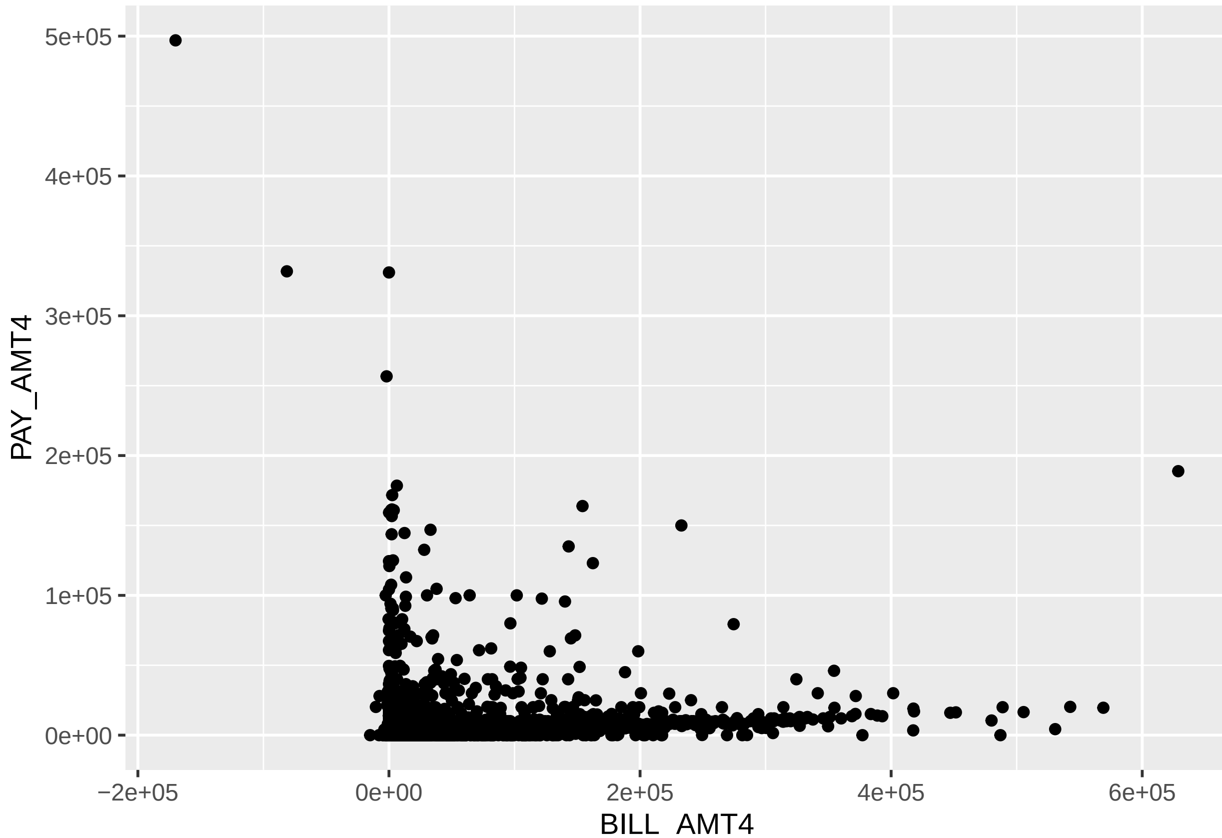

```
ggplot(train, aes(x = BILL_AMT2, y = PAY_AMT2)) + geom_point()
```



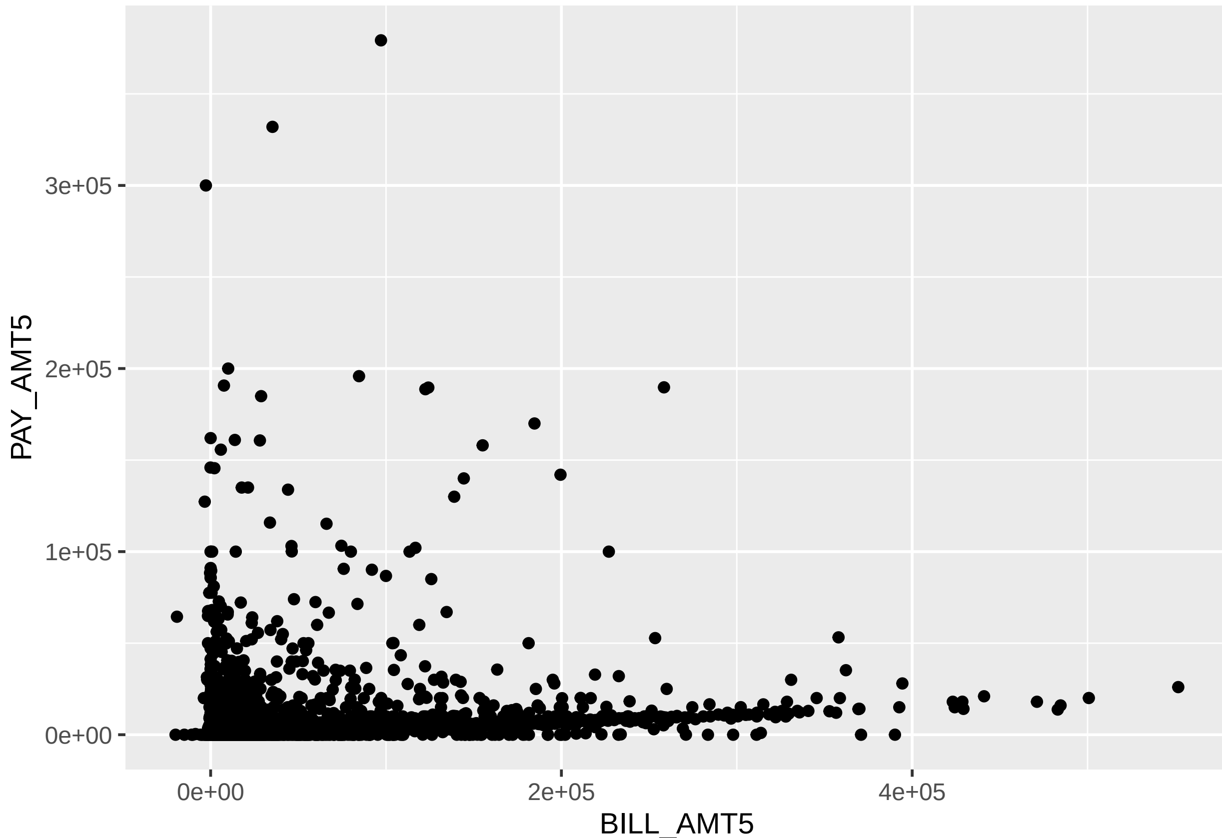

```
ggplot(train, aes(x = BILL_AMT3, y = PAY_AMT3)) + geom_point()
```



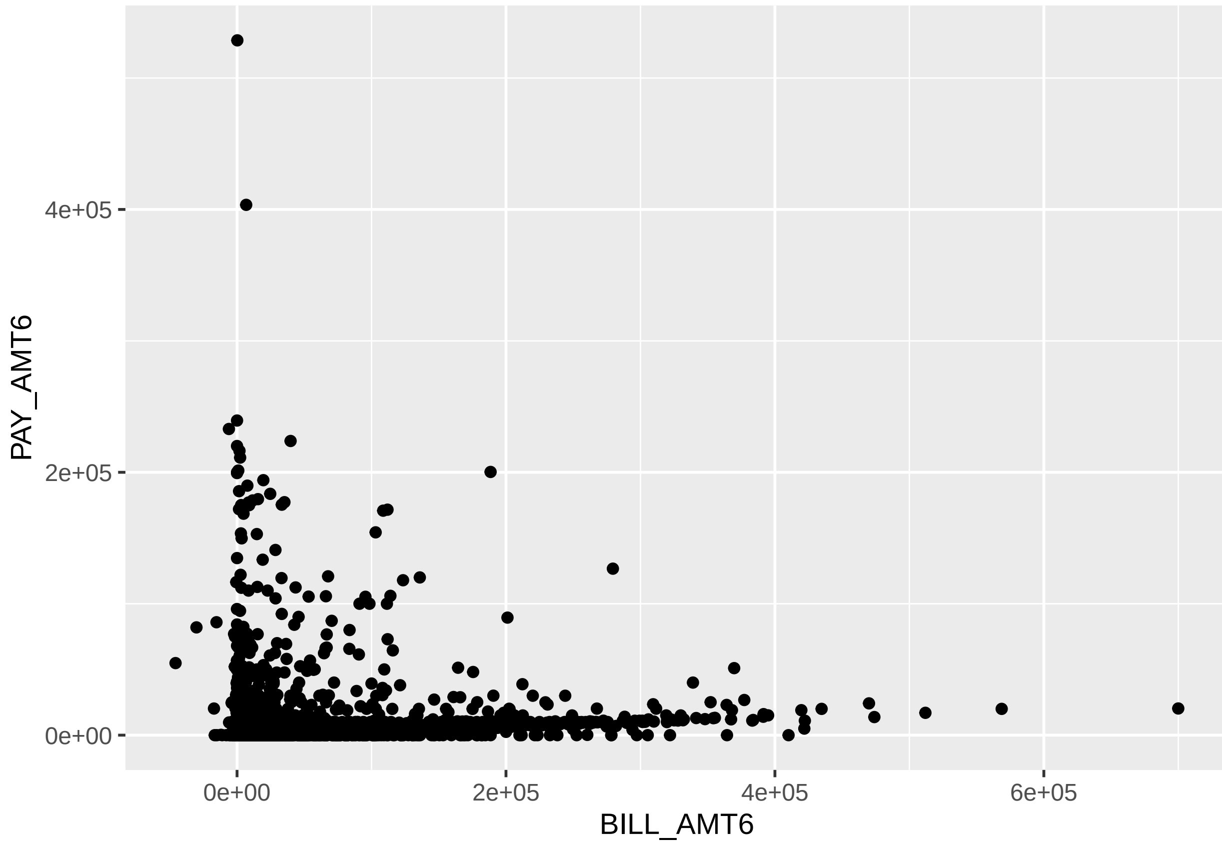

```
ggplot(train, aes(x = BILL_AMT4, y = PAY_AMT4)) + geom_point()
```




```
ggplot(train, aes(x = BILL_AMT5, y = PAY_AMT5)) + geom_point()
```




```
ggplot(train, aes(x = BILL_AMT6, y = PAY_AMT6)) + geom_point()
```



It seems that people with higher bills tend to pay more, except for people who have particularly low bills as they probably just pay the entire bill off.

Lets start with linear classification

I considered tuning for the best C value but considering the number of columns this would take a while, so I decided to just do multiple SVM classifications at different C values and compare them.

```
library(e1071)
lm1 <- svm(dpnm~., data = train, kernel = "linear", cost = .01)
p1 <- predict(lm1, newdata = test)
mean(p1==test$dpnm)
```

```
## [1] 0.771
```

```
lm2 <- svm(dpnm~., data = train, kernel = "linear", cost = 1)
p2 <- predict(lm2, newdata = test)
mean(p2==test$dpnm)
```

```
## [1] 0.7725
```

```
lm3 <- svm(dpnm~., data = train, kernel = "linear", cost = 10)
p3 <- predict(lm3, newdata = test)
mean(p3==test$dpnm)
```

```
## [1] 0.7725
```

It seems that 1 and 10 have the same values but they are a very slight improvement over .01. Cost doesn't really seem to have a large affect here.

Now lets try out polynomial

Same thing here as tuning would take a while.

```
library(e1071)
pm1 <- svm(dpnm~., data = train, kernel = "polynomial", cost = .01)
pp1 <- predict(pm1, newdata = test)
mean(pp1==test$dpnm)
```

```
## [1] 0.77
```

```
pm2 <- svm(dpnm~., data = train, kernel = "polynomial", cost = 1)
pp2 <- predict(pm2, newdata = test)
mean(pp2==test$dpnm)
```

```
## [1] 0.7775
```

```
pm3 <- svm(dpnm~., data = train, kernel = "polynomial", cost = 10)
pp3 <- predict(pm3, newdata = test)
mean(pp3==test$dpnm)
```

```
## [1] 0.775
```

There is pretty much no difference between linear and polynomial models here. Same thing with the cost values, slight improvement for the larger ones but nothing notable.

Radial last

Here we will do the same cost values as before with different gamma values each time as well.

```
rm1 <- svm(dpnm~., data = train, kernel = "radial", cost = .01, gamma = .05)
rp1 <- predict(rm1, newdata = test)
mean(rp1==test$dpnm)
```

```
## [1] 0.7705
```

```
rm2 <- svm(dpnm~., data = train, kernel = "radial", cost = 1, gamma = 1)
rp2 <- predict(rm2, newdata = test)
mean(rp2==test$dpnm)
```

```
## [1] 0.776
```

```
rm3 <- svm(dpnm~., data = train, kernel = "radial", cost = 10, gamma = 5)
rp3 <- predict(rm3, newdata = test)
mean(rp3==test$dpnm)
```

```
## [1] 0.7595
```

Analysis

Differently from the regression notebook, whenever doing classification with this data set it seemed that all of the different kernels had pretty much the same accuracy. I think that this is because with this particular data set, the data can be split linearly to some degree of success (77% success) and it does not get better than that because the data set uses multiple of the same types of data. For example there is pay1, pay2, pay3, etc. I think that because the data is all similar, just different values for different people that theres no improvement in higher dimensions. The final value with radial that was lower than the rest is probably because the high gamma value caused a higher level of variance.