MLBAM Skills Assesment

Miguel Briones

January 21, 2017

Data Set 1

First, we load the data:

```
library(corrplot)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(cluster)
library(fpc)

Data1 = read.csv("ds1.csv", header = TRUE)
```

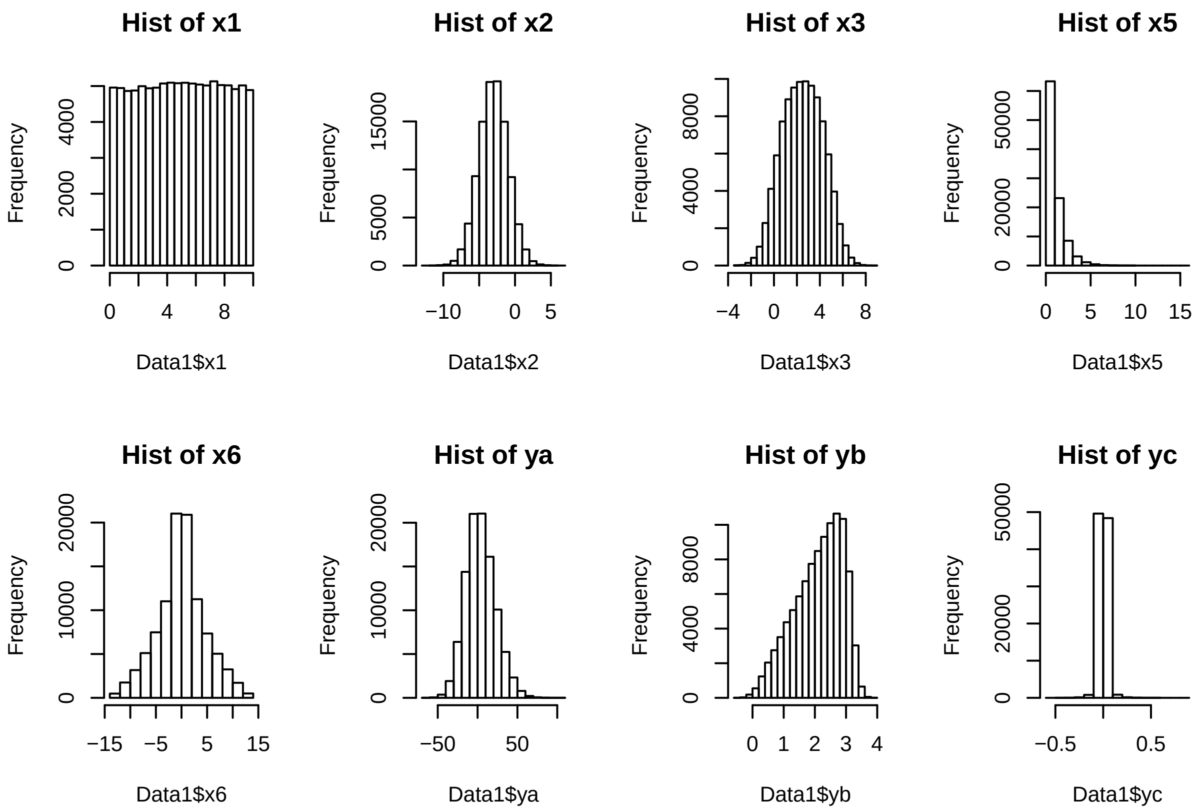
Then see if there are any missing values:

```
length(which(is.na(Data1)))
```

```
## [1] 0
```

Now we observe the distributions of each observation

```
par(mfrow=c(2,4))
hist(Data1$x1, main="Hist of x1")
hist(Data1$x2, main="Hist of x2")
hist(Data1$x3, main="Hist of x3")
hist(Data1$x5, main="Hist of x5")
hist(Data1$x6, main="Hist of x6")
hist(Data1$ya, main="Hist of ya")
hist(Data1$yb, main="Hist of yb")
hist(Data1$yc, main="Hist of yc")
```



We see that x1 has a uniform distribution, with all the data points appearing roughy at the same frequency. x2, x3, and x6 follow a normal distribution, while x5 follows a positively skewed distribution. For the Y data, ya and yc seem to follow a normal distribution, while yb shows a negatively skewed distribution.

Then we look at some of the summary statistics:

```r
summary(Data1)
```

```
##        X                x1                  x2                x3
##  Min.   :     1   Min.   : 0.000015   Min.   :-12.499   Min.   :-3.489
##  1st Qu.: 25001   1st Qu.: 2.536309   1st Qu.: -4.354   1st Qu.: 1.190
##  Median : 50000   Median : 5.022191   Median : -3.003   Median : 2.504
##  Mean   : 50000   Mean   : 5.011059   Mean   : -3.006   Mean   : 2.501
##  3rd Qu.: 75000   3rd Qu.: 7.486275   3rd Qu.: -1.649   3rd Qu.: 3.802
##  Max.   :100000   Max.   : 9.999887   Max.   :  6.090   Max.   : 8.679
##        x5                   x6                   ya
##  Min.   : 0.000003   Min.   :-13.885453   Min.   :-64.022
##  1st Qu.: 0.285629   1st Qu.: -2.611943   1st Qu.: -8.998
##  Median : 0.690903   Median : -0.000611   Median :  2.667
##  Mean   : 0.999136   Mean   :  0.000647   Mean   :  3.828
##  3rd Qu.: 1.386862   3rd Qu.:  2.621841   3rd Qu.: 15.580
##  Max.   :15.102966   Max.   : 13.924740   Max.   :107.714
##        yb                 yc
##  Min.   :-0.5237   Min.   :-0.5433613
##  1st Qu.: 1.5802   1st Qu.:-0.0024232
##  Median : 2.2311   Median : 0.0000000
##  Mean   : 2.1119   Mean   : 0.0001023
##  3rd Qu.: 2.7333   3rd Qu.: 0.0024767
##  Max.   : 3.8414   Max.   : 0.8183882
```

We see that x1, x2, x3, and x6 are evenly distributed, as you would expect from their histograms. x5 has a mean of ~ 1.0 but a maximum value of 15.10, so there are one or more outliers in this dataset. The ya dataset has a mean of ~4 but a min of -64 and a max of 107, so this dataset contains a large variance. The yb data set is contained between 0 and 4, and the yc dataset is even more contained, with a mean approaching 0 and most of the observaitions falling near 0.

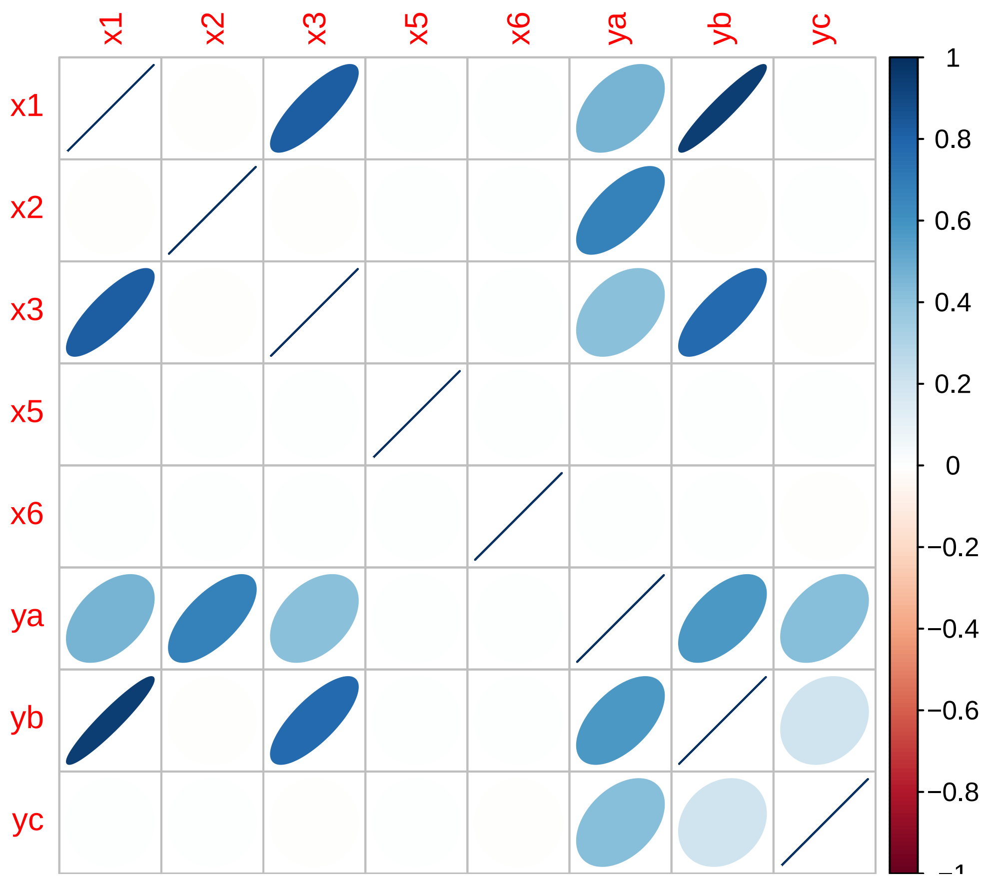
Next we want to see if there are any correlations between the data sets:

```
Data1obs = select(Data1, x1, x2, x3, x5, x6, ya, yb, yc)
cor(Data1obs)
```

```
##               x1            x2           x3           x5            x6
## x1  1.0000000000 -0.0017529136  0.821153567 0.0045435557  0.0007407111
## x2 -0.0017529136  1.0000000000 -0.002166799 0.0004927560  0.0007467917
## x3  0.8211535668 -0.0021667992  1.000000000 0.0024293829  0.0013688969
## x5  0.0045435557  0.0004927560  0.002429383 1.0000000000  0.0007464369
## x6  0.0007407111  0.0007467917  0.001368897 0.0007464369  1.0000000000
## ya  0.4654305850  0.6756484720  0.411148251 0.0024625640  0.0019637096
## yb  0.9459728690 -0.0014948070  0.775454994 0.0046656036  0.0011633502
## yc  0.0027456839  0.0004949225 -0.001077885 0.0028657434 -0.0001496716
##              ya           yb            yc
## x1 0.465430585  0.945972869  0.0027456839
## x2 0.675648472 -0.001494807  0.0004949225
## x3 0.411148251  0.775454994 -0.0010778852
## x5 0.002462564  0.004665604  0.0028657434
## x6 0.001963710  0.001163350 -0.0001496716
## ya 1.000000000  0.572271212  0.4215037584
## yb 0.572271212  1.000000000  0.2032748866
## yc 0.421503758  0.203274887  1.0000000000
```

And we plot them using the corrplot package, which allows us to visually see the strength of the correlations between each dataset:

```
corrplot(cor(Data1obs), method = "ellipse")
```



We can see the dataset ya is correlated with x1 (r = 0.4654), x2 (r = 0.6756), and x3 (r = 0.4111). The dataset yb is correlated with x1 (r = 0.9459) and x3 (r = 0.7754). The dataset y3 seems to not be correlated with any of the x data sets; every correlation is ~ 0.

Therefore, we can begin by trying to predict the ya dataset. We can use the three x datasets that are most closely correlated with ya, and see which model is best. First, we start with the best correlation, which is ya ~ x2, to create the first model. We then add the next best correlated dataset, x1, to create the second model. Finally, we add the third best correlated dataset, x3, to create the third model:

```
yamodel = lm(ya ~ x2, data = Data1)
summary(yamodel)
```

```
##
## Call:
## lm(formula = ya ~ x2, data = Data1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -60.009  -9.314  -0.117   9.274  61.106
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 22.54199    0.07757   290.6   <2e-16 ***
## x2           6.22649    0.02148   289.8   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 13.59 on 99998 degrees of freedom
## Multiple R-squared:  0.4565, Adjusted R-squared:  0.4565
## F-statistic: 8.399e+04 on 1 and 99998 DF,  p-value: < 2.2e-16
```

```r
yamodel2 = lm(ya ~ x1 + x2, data = Data1)
summary(yamodel2)
```

```
##
## Call:
## lm(formula = ya ~ x1 + x2, data = Data1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -46.711  -7.102  -0.052   7.051  56.380
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.56109    0.08346    90.6   <2e-16 ***
## x1           2.99409    0.01158   258.5   <2e-16 ***
## x2           6.23403    0.01663   374.8   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.52 on 99997 degrees of freedom
## Multiple R-squared:  0.6742, Adjusted R-squared:  0.6742
## F-statistic: 1.035e+05 on 2 and 99997 DF,  p-value: < 2.2e-16
```

```r
yamodel3 = lm(ya ~ x1 + x2 + x3, data = Data1)
summary(yamodel3)
```

```
##
## Call:
## lm(formula = ya ~ x1 + x2 + x3, data = Data1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -45.852  -7.058  -0.054   7.043  56.898
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.57195    0.08312   91.10   <2e-16 ***
## x1           2.51766    0.02021  124.57   <2e-16 ***
## x2           6.23463    0.01657  376.36   <2e-16 ***
## x3           0.95111    0.03313   28.71   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.48 on 99996 degrees of freedom
## Multiple R-squared:  0.6769, Adjusted R-squared:  0.6769
## F-statistic: 6.983e+04 on 3 and 99996 DF,  p-value: < 2.2e-16
```

Then we run an ANOVA to see which model is the best predictor:

```r
anova(yamodel, yamodel2, yamodel3)
```

```
## Analysis of Variance Table
##
## Model 1: ya ~ x2
## Model 2: ya ~ x1 + x2
## Model 3: ya ~ x1 + x2 + x3
##   Res.Df      RSS Df Sum of Sq        F    Pr(>F)
## 1  99998 18477633
## 2  99997 11075343  1   7402290 67383.90 < 2.2e-16 ***
## 3  99996 10984811  1     90532   824.12 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We see that Model 2 and 3 are significantly different from Model 1, and Model 2 has the bigger F statistic than Model 3, so we can conclude that Model 2 is the best predictor for the ya dataset.

Next, we do the same for the yb model, and chose the datasets that bet correlate with yb, which in this case is x1 and x3:

```r
ybmodel = lm(yb ~ x1, data = Data1)
summary(ybmodel)
```

```
##
## Call:
## lm(formula = yb ~ x1, data = Data1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.38507 -0.15637  0.01141  0.17047  0.96795
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.8427828  0.0015857   531.5   <2e-16 ***
## x1          0.2532541  0.0002745   922.6   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2494 on 99998 degrees of freedom
## Multiple R-squared:  0.8949, Adjusted R-squared:  0.8949
## F-statistic: 8.511e+05 on 1 and 99998 DF,  p-value: < 2.2e-16
```

```r
ybmodel2 = lm(yb ~ x1 + x3, data = Data1)
summary(ybmodel2)
```

```
##
## Call:
## lm(formula = yb ~ x1 + x3, data = Data1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.38435 -0.15631  0.01134  0.17042  0.97007
```

```
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.8427657  0.0015857  531.48   <2e-16 ***
## x1           0.2541545  0.0004810  528.40   <2e-16 ***
## x3          -0.0017975  0.0007885   -2.28   0.0226 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2494 on 99997 degrees of freedom
## Multiple R-squared:  0.8949, Adjusted R-squared:  0.8949
## F-statistic: 4.256e+05 on 2 and 99997 DF,  p-value: < 2.2e-16
```

```
anova(ybmodel, ybmodel2)
```

```
## Analysis of Variance Table
##
## Model 1: yb ~ x1
## Model 2: yb ~ x1 + x3
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1  99998 6222.2
## 2  99997 6221.9  1   0.32336 5.1969 0.02263 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Looking at the ANOVA, we see that the comparison between Model 1 and Model 2 was statistically significant, thereby telling us that Model 2, with the added dataset (x3) is a better predictor than Model 1.

Because there was no strong correlation between any of the x datasets with yc, I did not run a linear regression.

I would argue that the chosen model for ya is the strongest model for predicting the ya dataset, due to the strong correlations between the chosen variables. The chosen model for yb is a good model, but I would argue that it is not as strong as the chosen model for ya, again because of the correlations between the chosen variables.

Dataset 2

First we import the second dataset:

```
Data2 = read.csv("ds2.csv", header = TRUE)
```

Then we check whether there are any missing values in the dataset:

```
length(which(is.na(Data2)) == TRUE)
```

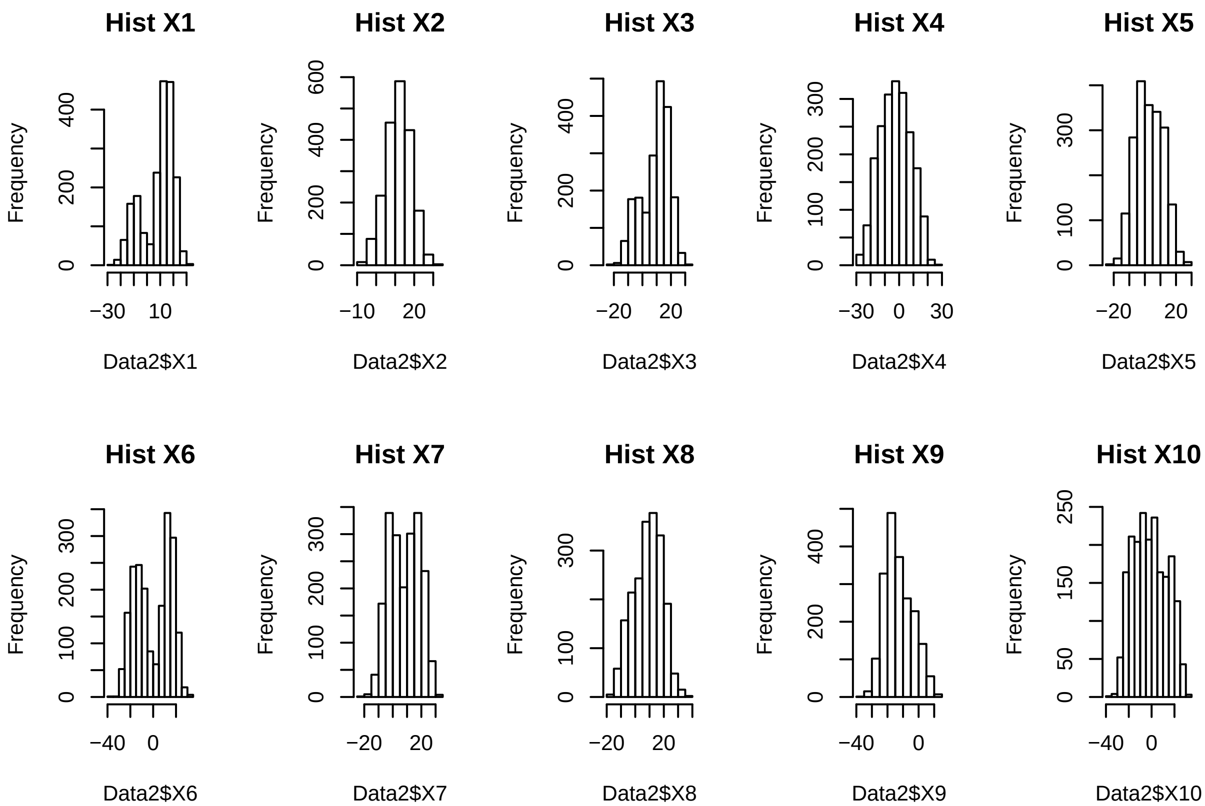
```
## [1] 0
```

Then we look at the summary of the data and create histograms for each of the variables in the dataset:

```r
summary(Data2)
```

```
##        X                 X1                 X2                X3
##  Min.   :   1.0   Min.   :-25.8242   Min.   :-8.498   Min.   :-23.666
##  1st Qu.: 500.8   1st Qu.:  0.2313   1st Qu.: 7.162   1st Qu.:  2.649
##  Median :1000.5   Median : 12.7543   Median :11.896   Median : 11.422
##  Mean   :1000.5   Mean   :  8.6778   Mean   :11.717   Mean   :  9.253
##  3rd Qu.:1500.2   3rd Qu.: 17.3643   3rd Qu.:16.279   3rd Qu.: 16.504
##  Max.   :2000.0   Max.   : 32.2686   Max.   :32.910   Max.   : 31.231
##        X4                X5                X6
##  Min.   :-29.430   Min.   :-22.033   Min.   :-35.26402
##  1st Qu.:-10.653   1st Qu.: -4.098   1st Qu.:-14.00367
##  Median : -2.631   Median :  2.484   Median :  1.50084
##  Mean   : -2.680   Mean   :  2.775   Mean   :  0.07763
##  3rd Qu.:  5.340   3rd Qu.:  9.661   3rd Qu.: 14.05051
##  Max.   : 26.423   Max.   : 29.312   Max.   : 31.72704
##        X7                 X8                X9                X10
##  Min.   :-21.4285   Min.   :-16.811   Min.   :-36.065   Min.   :-36.468
##  1st Qu.: -0.8129   1st Qu.:  1.481   1st Qu.:-19.431   1st Qu.:-13.216
##  Median :  8.5325   Median :  9.628   Median :-14.418   Median : -2.094
##  Mean   :  8.2009   Mean   :  8.713   Mean   :-12.860   Mean   : -1.339
##  3rd Qu.: 17.1389   3rd Qu.: 16.081   3rd Qu.: -6.534   3rd Qu.: 10.562
##  Max.   : 32.0843   Max.   : 36.848   Max.   : 13.554   Max.   : 32.642
```

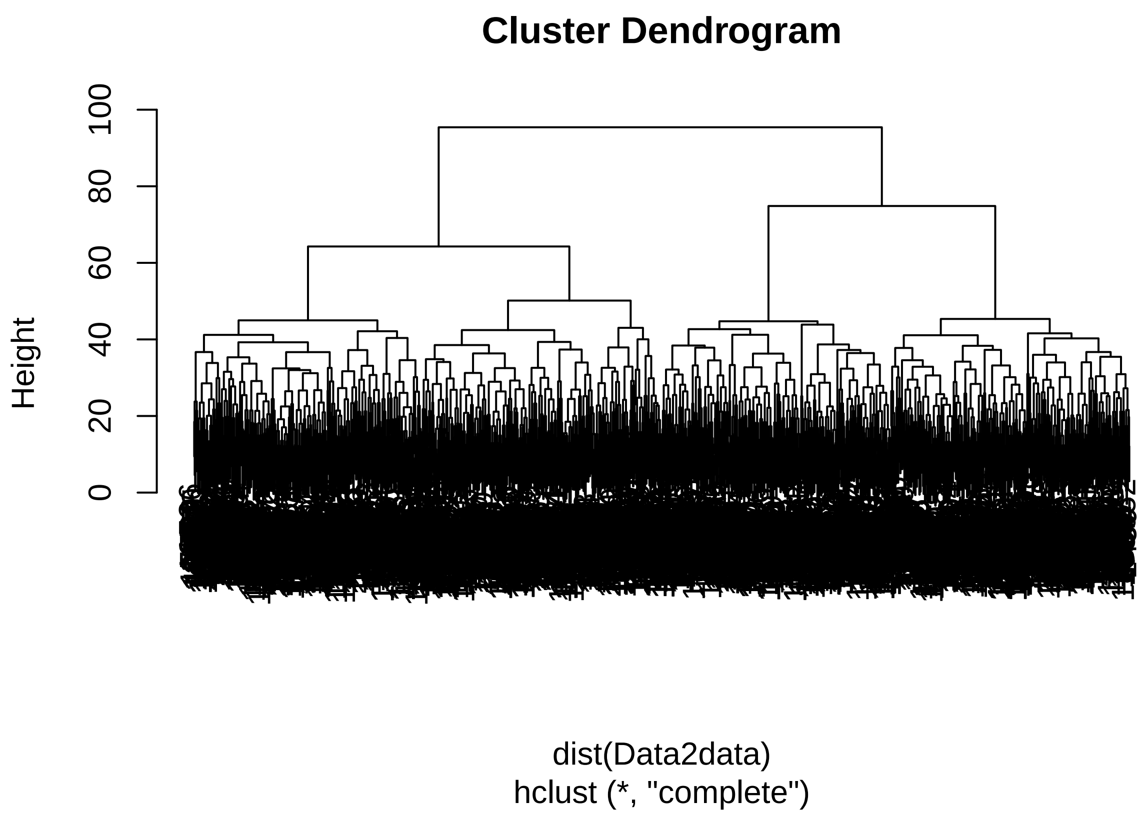
```r
par(mfrow=c(2,5))
hist(Data2$X1, main = "Hist X1")
hist(Data2$X2, main = "Hist X2")
hist(Data2$X3, main = "Hist X3")
hist(Data2$X4, main = "Hist X4")
hist(Data2$X5, main = "Hist X5")
hist(Data2$X6, main = "Hist X6")
hist(Data2$X7, main = "Hist X7")
hist(Data2$X8, main = "Hist X8")
hist(Data2$X9, main = "Hist X9")
hist(Data2$X10, main = "Hist X10")
```



The data set seems to be composed of observations that range from -37 to 37. The histogram data show that variables x1, x6, and x7 show a slight binomial distribution. Variables x2, x4, x5, x8 and x10 show a normal distribution. X3 follows a slightly negatively skewed distribution and X9 a slightly positively skewed distribution. Overall, the data set seems to have observations that are distributed in four different ways.

In order to visually see how the data is clustered, I plotted the data using the hierarchical clustering method:

```
par(mfrow=c(1,1))
Data2data = select(Data2, X1, X2, X3, X4, X5, X6, X7, X8, X9, X10)
Data2Cluster = hclust(dist(Data2data))
plot(Data2Cluster)
```



We can see that at height 70, the data is split into three clusters. If we cut the hierarchical cluster by 3, we see the following:

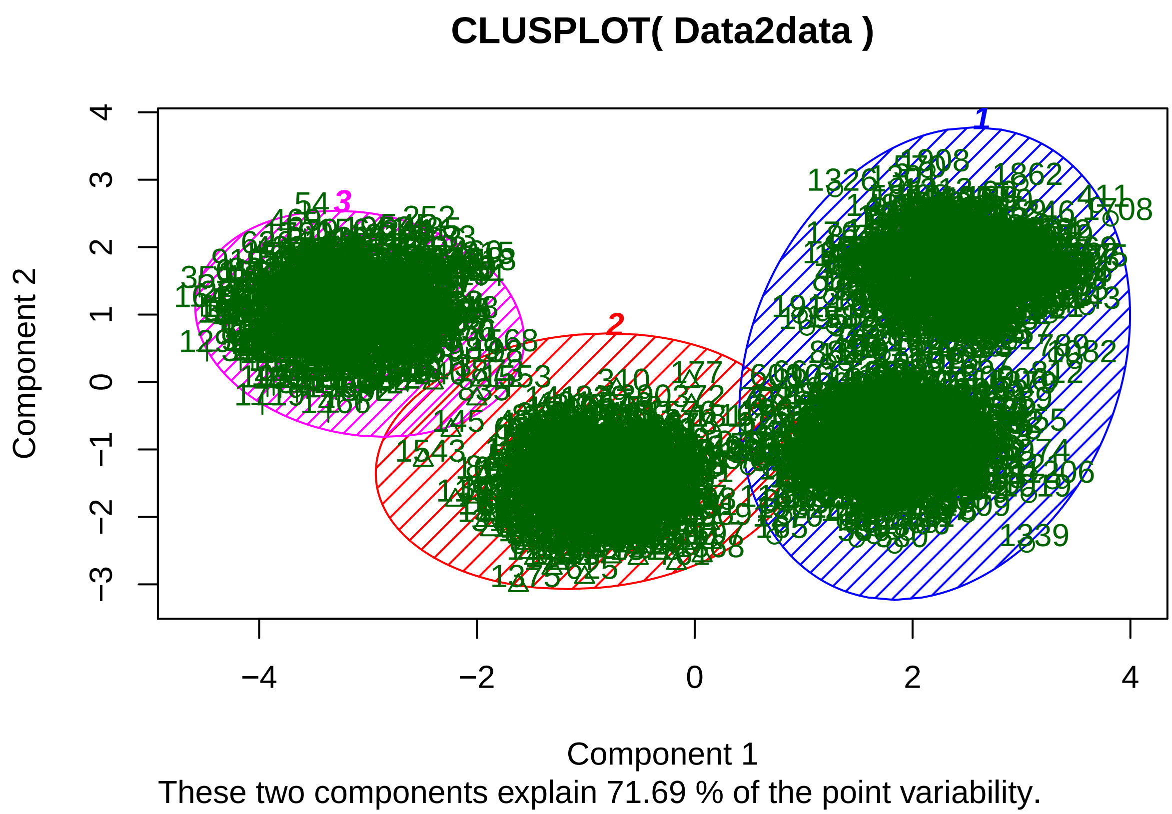
```
Data2ClusterCut = cutree(Data2Cluster, 3)
table(Data2ClusterCut)
```

```
## Data2ClusterCut
##    1    2    3
## 1001  489  510
```

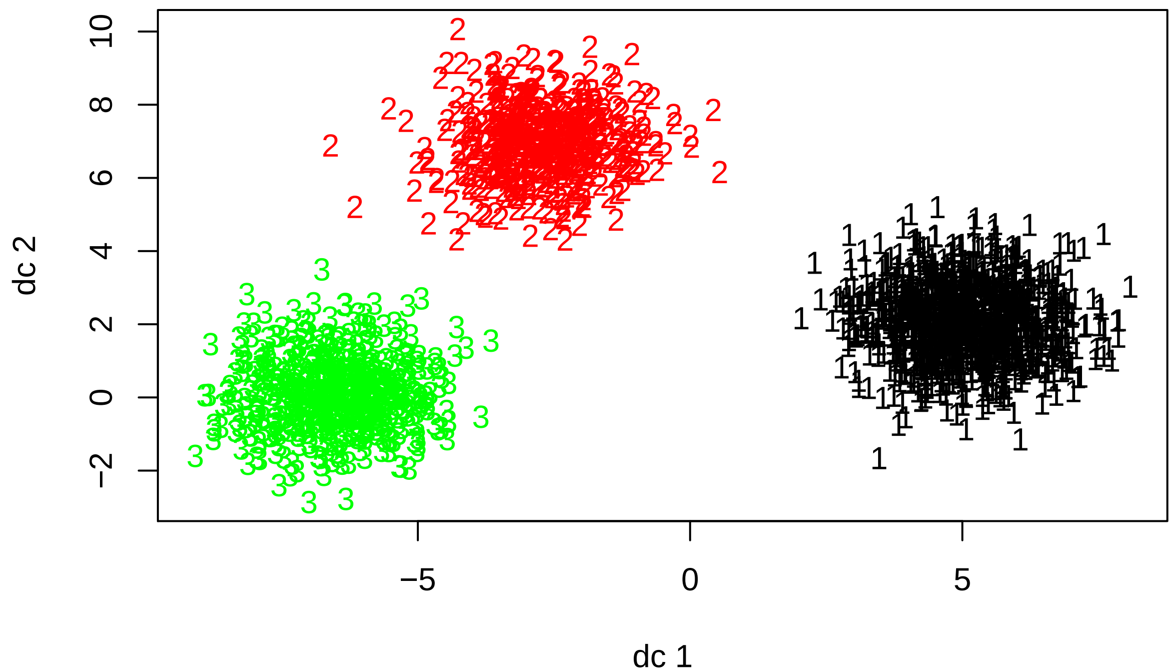
It seems that there are three 3 sources that the data could've came from.

If we graph the clusters, we see where there is overlap between the data:

```
clusplot(Data2data, Data2ClusterCut, color=TRUE, shade=TRUE,
         labels=2, lines=0)
```




```r
plotcluster(Data2data, Data2ClusterCut)
```



We see that there are three distinct clusters where the data fit.