Project 6

Marcus McKenzie

6.1

Using the crime data set uscrime.txt, apply Principal Component Analysis and then create a regression model using the first few principal components. Specify your new model in terms of the original variables (not the principal components), and compare its quality to that of your solution to project 5.2

Load and Organize data:

```
#data <- read.table("Documents/OMSCS/Analytics Modeling/Assignments/Assignment4/crime.txt", header= TRU

data <- read.table("uscrime.txt", header=TRUE, stringsAsFactors = FALSE)

head(data)
```

```
##      M So   Ed  Po1  Po2    LF   M.F Pop   NW    U1  U2 Wealth Ineq     Prob
## 1 15.1  1  9.1  5.8  5.6 0.510  95.0  33 30.1 0.108 4.1   3940 26.1 0.084602
## 2 14.3  0 11.3 10.3  9.5 0.583 101.2  13 10.2 0.096 3.6   5570 19.4 0.029599
## 3 14.2  1  8.9  4.5  4.4 0.533  96.9  18 21.9 0.094 3.3   3180 25.0 0.083401
## 4 13.6  0 12.1 14.9 14.1 0.577  99.4 157  8.0 0.102 3.9   6730 16.7 0.015801
## 5 14.1  0 12.1 10.9 10.1 0.591  98.5  18  3.0 0.091 2.0   5780 17.4 0.041399
## 6 12.1  0 11.0 11.8 11.5 0.547  96.4  25  4.4 0.084 2.9   6890 12.6 0.034201
##      Time Crime
## 1 26.2011   791
## 2 25.2999  1635
## 3 24.3006   578
## 4 29.9012  1969
## 5 21.2998  1234
## 6 20.9995   682
```

```
data <- data[-2]

head(data)
```

```
##      M   Ed  Po1  Po2    LF   M.F Pop   NW    U1  U2 Wealth Ineq     Prob
## 1 15.1  9.1  5.8  5.6 0.510  95.0  33 30.1 0.108 4.1   3940 26.1 0.084602
## 2 14.3 11.3 10.3  9.5 0.583 101.2  13 10.2 0.096 3.6   5570 19.4 0.029599
## 3 14.2  8.9  4.5  4.4 0.533  96.9  18 21.9 0.094 3.3   3180 25.0 0.083401
## 4 13.6 12.1 14.9 14.1 0.577  99.4 157  8.0 0.102 3.9   6730 16.7 0.015801
## 5 14.1 12.1 10.9 10.1 0.591  98.5  18  3.0 0.091 2.0   5780 17.4 0.041399
## 6 12.1 11.0 11.8 11.5 0.547  96.4  25  4.4 0.084 2.9   6890 12.6 0.034201
##      Time Crime
## 1 26.2011   791
```

```
## 2 25.2999  1635
## 3 24.3006   578
## 4 29.9012  1969
## 5 21.2998  1234
## 6 20.9995   682
```

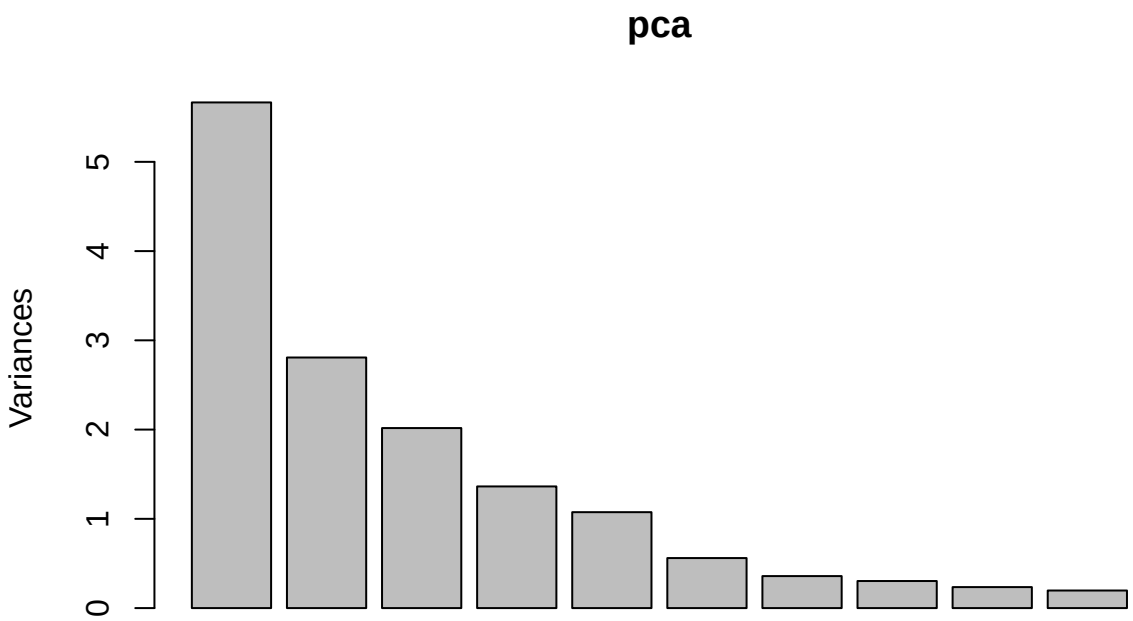
Create and plot PCA

```
pca <- prcomp(data[,1:15],scale =TRUE)

summary(pca)
```

```
## Importance of components:
##                            PC1    PC2    PC3     PC4     PC5     PC6    PC7
## Standard deviation     2.3802 1.6756 1.4202 1.16749 1.03667 0.74864 0.5988
## Proportion of Variance 0.3777 0.1872 0.1345 0.09087 0.07165 0.03736 0.0239
## Cumulative Proportion  0.3777 0.5649 0.6993 0.79020 0.86185 0.89921 0.9231
##                            PC8     PC9    PC10    PC11    PC12    PC13   PC14
## Standard deviation     0.55069 0.48478 0.44375 0.42652 0.32674 0.26644 0.2324
## Proportion of Variance 0.02022 0.01567 0.01313 0.01213 0.00712 0.00473 0.0036
## Cumulative Proportion  0.94334 0.95900 0.97213 0.98426 0.99138 0.99611 0.9997
##                           PC15
## Standard deviation     0.06595
## Proportion of Variance 0.00029
## Cumulative Proportion  1.00000
```

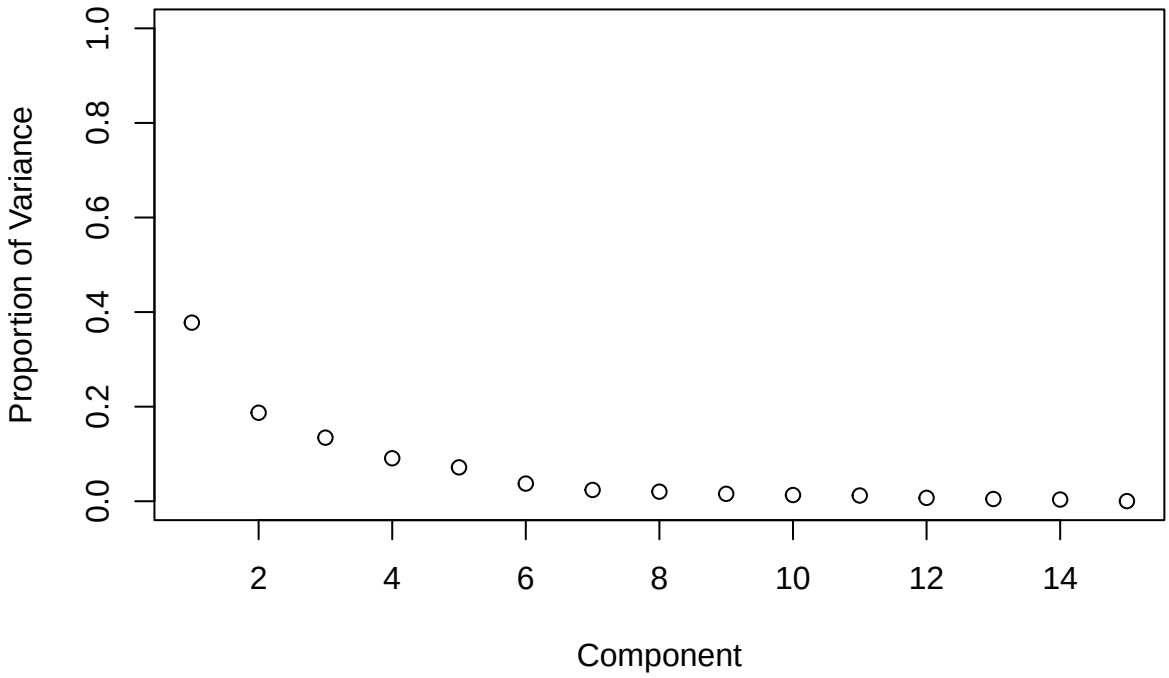
```
plot(pca)
```



Variance proportions for each component:

```
variance <- pca$sdev^2

proportional_variance <- variance/sum(variance)
plot(proportional_variance,
     xlab = "Component",
     ylab = "Proportion of Variance",
     ylim = c(0,1))
```



Testing pca model:

```
pca_test <- cbind(pca$x[,1:6],data[,15])

linear <- lm(V7~., data = as.data.frame(pca_test))

summary(linear)
```

```
##
## Call:
## lm(formula = V7 ~ ., data = as.data.frame(pca_test))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -289.78  -86.76   14.92   81.89  260.15
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  905.085     20.300  44.584  < 2e-16 ***
## PC1           89.117      8.621  10.337 7.36e-13 ***
## PC2           75.018     12.247   6.126 3.15e-07 ***
## PC3           38.075     14.449   2.635   0.0119 *
## PC4          222.730     17.576  12.672 1.38e-15 ***
## PC5           -2.104     19.794  -0.106   0.9159
## PC6          -50.000     27.410  -1.824   0.0756 .
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 139.2 on 40 degrees of freedom
## Multiple R-squared:  0.8874, Adjusted R-squared:  0.8705
## F-statistic: 52.54 on 6 and 40 DF,  p-value: < 2.2e-16
```

Calculate alpha and beta:

```
b <- linear$coefficients[1]

b2 <- linear$coefficients[2:7]

a <- pca$rotation[,1:6] %*% b2

a
```

```
##              [,1]
## M       79.657458
## Ed      12.112178
## Po1     89.280683
## Po2     85.467044
## LF      18.667639
## M.F     80.210336
## Pop      3.290012
## NW      71.168881
## U1      -3.283511
## U2      23.949950
## Wealth  22.734482
## Ineq    20.067746
## Prob   -39.276481
## Time    18.214699
## Crime  173.574985
```

Determine estimates:

```
mu <- sapply(data[,1:15],mean)
s <- sapply(data[,1:15],sd)
alpha <- a/s
beta <- b - sum(a*mu /s)

est <- as.matrix(data[,1:15]) %*% alpha + beta

est
```

```
##            [,1]
##  [1,]  740.6536
##  [2,] 1495.3752
##  [3,]  468.0246
##  [4,] 1890.6364
##  [5,] 1148.3222
##  [6,]  805.0207
##  [7,]  865.6597
##  [8,] 1344.7579
```

```
##  [9,]  863.6125
## [10,]  801.4037
## [11,] 1462.9800
## [12,]  795.4357
## [13,]  496.0775
## [14,]  601.7158
## [15,]  712.5756
## [16,]  979.5527
## [17,]  474.7708
## [18,]  998.4964
## [19,]  911.8258
## [20,] 1223.8850
## [21,]  715.0817
## [22,]  650.1627
## [23,]  955.8504
## [24,]  956.3487
## [25,]  448.2506
## [26,] 2061.6070
## [27,]  402.5166
## [28,] 1205.0591
## [29,] 1263.6466
## [30,]  794.9837
## [31,]  581.4470
## [32,]  772.3960
## [33,]  852.8947
## [34,]  888.0621
## [35,]  730.1218
## [36,] 1255.3792
## [37,] 1120.7829
## [38,]  530.3556
## [39,]  709.7502
## [40,] 1065.1025
## [41,]  887.9094
## [42,]  340.4742
## [43,] 1032.5556
## [44,] 1132.7131
## [45,]  391.5945
## [46,]  697.8823
## [47,] 1015.2899
```

Calculate accuracy of model:

```
error_sum = sum((est - data[,15])^2)
total_sum = sum((data[,15] - mean(data[,15]))^2)
acc <- 1 - error_sum/total_sum
acc
```

```
## [1] 0.8874037
```

Conclusion

Compared to the previous accuracy that was calculated in 8.2, we can see that the accuracy using a regression model with principal component analysis provides better results.