writeup

Tian Tan

February 5, 2016

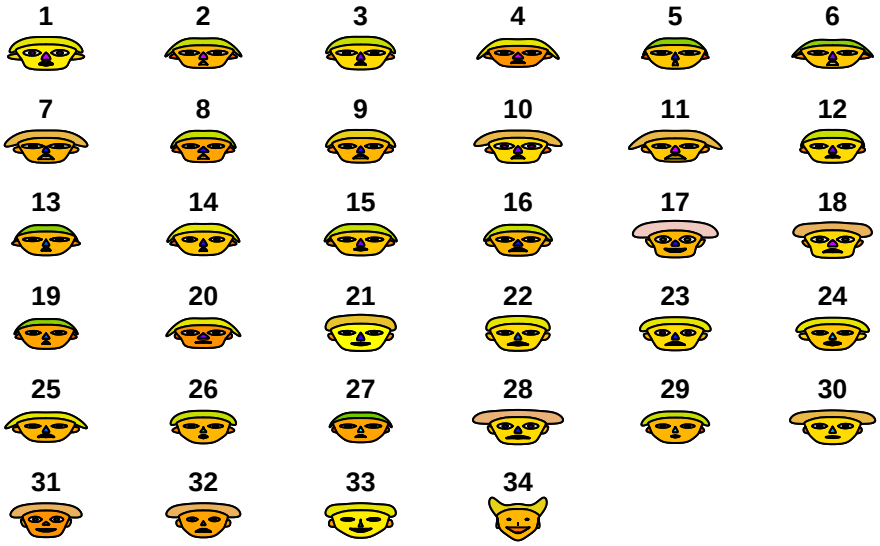
Chernoff Faces

1.

```
library(aplpack)
```

```
## Loading required package: tcltk
```

```
olympic=read.csv("olympic.csv")
faces(olympic[1:10])
```




```
## effect of variables:
##  modified item       Var
##  "height of face   " "m100"
##  "width of face    " "ljump"
##  "structure of face" "shot"
##  "height of mouth  " "hjump"
##  "width of mouth   " "m400"
##  "smiling          " "m110h"
##  "height of eyes   " "disc"
##  "width of eyes    " "pvault"
##  "height of hair   " "jav"
##  "width of hair    "  "m1500"
##  "style of hair    "  "m100"
##  "height of nose   "  "ljump"
##  "width of nose    "  "shot"
##  "width of ear     "  "hjump"
##  "height of ear    "  "m400"
```

2.

1) The 100-meter time running corresponds to the height of the face and the style of hair. The longer the time run, the longer the length the face, the higher the hair.
2) The long jump distance corresponds to the width of the face and the height of the nose. The further the distance jumped, the wider the the face and the higher the nose.
3) The shot distance corresponds to the structure of the face and the width of the nose. The shorter the distance, the more trangular the face becomes and the wider the nose is.
4) The high jump corresponds to the height of the mouth and the width of the ear. The higher the jump, the higher the mouth and the wider the ear.
5) The 400-meter running time corresponds to the width of the mouth and the height of the ear. The longer the time, the wider the mouth and higher the ear.
6) The 110 Meter and 120 Yard Hurdles corresponds to the smiling. The longer the time, the larger the smile.
7) The disc distance corresponds to the height of the eyes. The longer the distance, the higher the eyes.
8) The score of vault corresponds to the width of eyes. The higher the score, the wider the eyes.
9) The score of jav corresponds to the height of hair. The higher the score, the higher the hair.
10) The 1500-meter time corresponds to the width of the hair. The longer the time, the wider the hair.

3.All have similar height and width of face, similar height and width of mouth, and similar shape of smile, which means they should have similar m100,ljump, hjump,m400 and m110h value. And their corresponding data is shown pretty similar.

4. The outlier is the last one. Its face is close to triangle while others are not; its eye are similar to a point while the others are long and thin;its hair is upward while others are downward; it is laughing while others are either smiling or numb. So the 34th is the outlier.

Principal Component Analysis

1.Getting the Principal Components

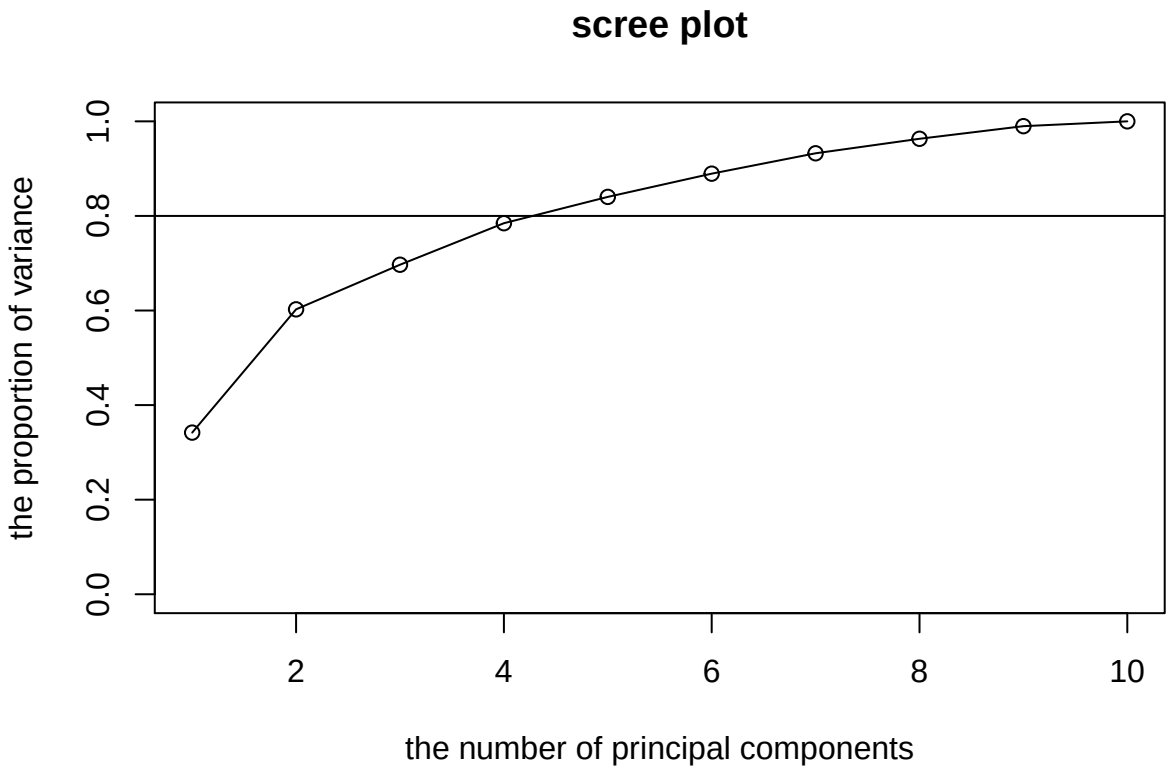
```
X=read.csv("olympic.csv")
X=X[-34,-11]
standardizedX=scale(X)
corX=cor(standardizedX)
eigenX=eigen(corX)
print(eigenX)
```

```
## $values
##  [1] 3.4182381 2.6063931 0.9432964 0.8780212 0.5566267 0.4912275 0.4305952
##  [8] 0.3067981 0.2669494 0.1018542
##
## $vectors
##             [,1]       [,2]        [,3]        [,4]         [,5]
##  [1,]  0.4158823  0.1488081 -0.26747198  0.08833244 -0.442314456
##  [2,] -0.3940515 -0.1520815 -0.16894945  0.24424963  0.368913901
##  [3,] -0.2691057  0.4835374  0.09853273  0.10776276 -0.009754680
##  [4,] -0.2122818  0.0278985 -0.85498656 -0.38794393 -0.001876311
##  [5,]  0.3558474  0.3521598 -0.18949642 -0.08057457  0.146965351
##  [6,]  0.4334816  0.0695682 -0.12616012  0.38229029 -0.088802794
##  [7,] -0.1757923  0.5033347  0.04609969 -0.02558404  0.019358607
##  [8,] -0.3840821  0.1495820  0.13687235 -0.14396548 -0.716743474
```

```
##  [9,] -0.1799436  0.3719570 -0.19232803  0.60046566  0.095582043
## [10,]  0.1701426  0.4209653  0.22255233 -0.48564231  0.339772188
##              [,6]       [,7]         [,8]        [,9]       [,10]
##  [1,]  0.03071237  0.2543985  0.663712826 -0.10839531 -0.10948045
##  [2,] -0.09378242  0.7505343  0.141264141  0.04613910 -0.05580431
##  [3,]  0.23002054 -0.1106637  0.072505560  0.42247611 -0.65073655
##  [4,]  0.07454380 -0.1351242 -0.155435871 -0.10206505 -0.11941181
##  [5,] -0.32692886  0.1413388 -0.146839303  0.65076229  0.33681395
##  [6,]  0.21049130  0.2725296 -0.639003579 -0.20723854 -0.25971800
##  [7,]  0.61491241  0.1439726  0.009400445 -0.16724055  0.53450315
##  [8,] -0.34776037  0.2732665 -0.276873049 -0.01766443  0.06589572
##  [9,] -0.43744387 -0.3419099  0.058519366 -0.30619617  0.13093187
## [10,] -0.30032419  0.1868704  0.007310045 -0.45688227 -0.24311846
```

3.Which Principal Components are important?

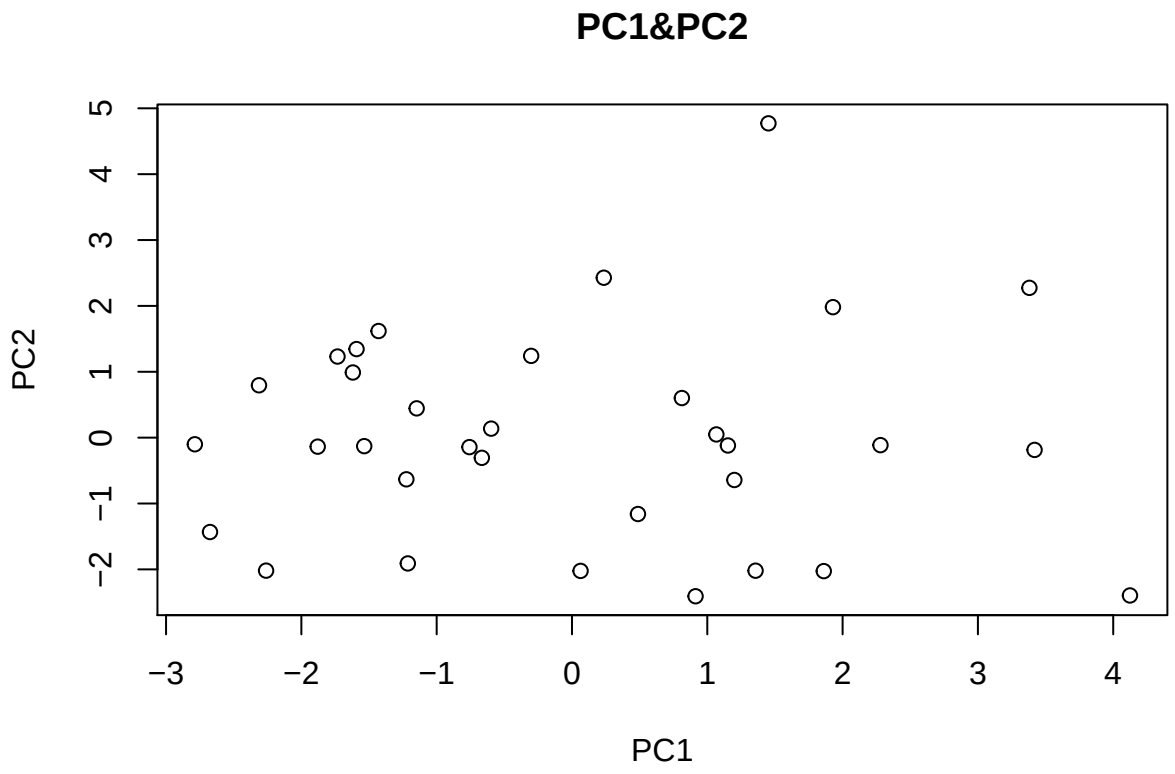
```
sum=0
for (k in c(eigenX$values)){
  sum=sum+k
}
result=rep(0,10)
sum2=0
for (i in 1:10){
  sum2=(sum2)+eigenX$values[i]
  result[i]=sum2/sum
}
matrix=cbind(c(1:10),result)
plot(matrix,xlab="the number of principal components",ylab="the proportion of variance",ylim=c(0,1),xli
abline(h=0.8)
lines(matrix)
```



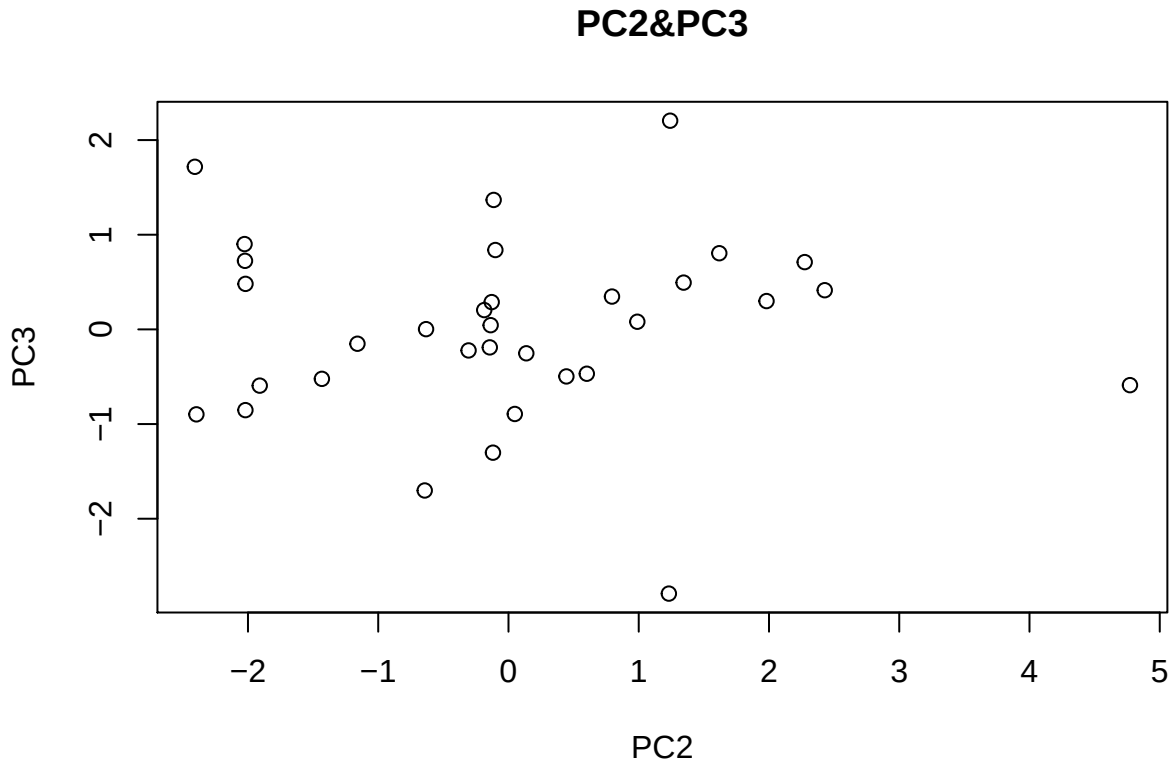
4. Interpret these principal components The second principal component: From the magnitude and the signs of each entry, we can tell that only ljump has a negative sign. Ljump, hjump,m110h and pvault are extremely small in magnitude. This would roughly mean that the value for jumping are less than others(except m110h). The third principal component: There's no interpretation for it. Shot, disc, pvault and m1500 have a negative sign, while shot and disc have a smaller magnitude than others. Analysis can't be given since there is no obvious evidence show correlation between events. The forth principal component: It's noise as well. The magnitude and the signs of each entry seems randomly listed.

5. What else can I do with the principal componets?

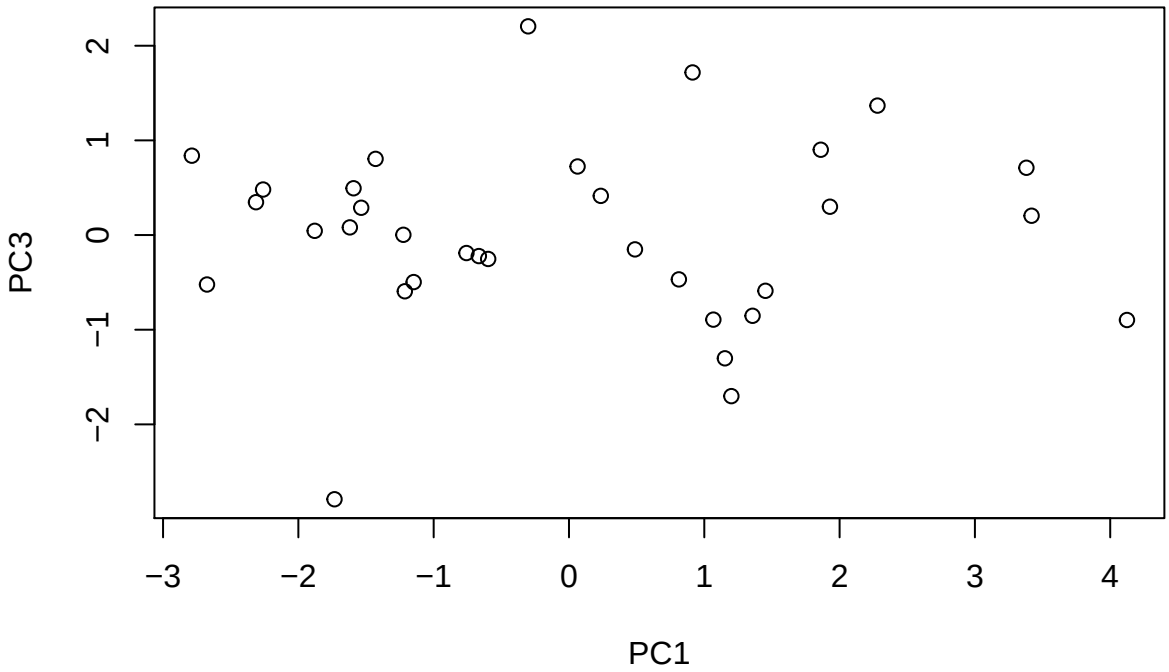
```
PC1=matrix(eigenX$vectors[,1],nrow=10)
XP1=standardizedX%*%(PC1)
PC2=matrix(eigenX$vectors[,2],nrow=10)
XP2=standardizedX%*%(PC2)
PC3=matrix(eigenX$vectors[,3],nrow=10)
XP3=standardizedX%*%(PC3)
plot(XP1,XP2,main="PC1&PC2",xlab="PC1",ylab="PC2")
```




```
plot(XP2,XP3,main="PC2&PC3",xlab="PC2",ylab="PC3")
```




```
plot(XP1,XP3,main="PC1&PC3",xlab="PC1",ylab="PC3")
```



Interpretation: The data from PC1&PC2 plot seems randomly distributed. There might be some correlation between PC2&PC3, because some of its data points fit a line with a positive slope. The data distribution in PC1&PC3 doesn't have support their correlation either.It is somehow symmetric though.

6. A brief comparison Yes, there are similarities between PCA and Chernoff Faces. Both methods use multivariate statistics to identify unreliable data in a large dataset, which may contain large amount of parameters.Though it is easier to identify the outlier in Chernoff Faces method.