Lab-10

Ehshan Veerabangsa

12 December 2017

Principal Component Analysis

Explore Dataset

To begin, let us examine the dataset features.

```
# List all rows headers
states = row.names(USArrests)

states
```

```
##  [1] "Alabama"        "Alaska"         "Arizona"        "Arkansas"
##  [5] "California"     "Colorado"       "Connecticut"    "Delaware"
##  [9] "Florida"        "Georgia"        "Hawaii"         "Idaho"
## [13] "Illinois"       "Indiana"        "Iowa"           "Kansas"
## [17] "Kentucky"       "Louisiana"      "Maine"          "Maryland"
## [21] "Massachusetts"  "Michigan"       "Minnesota"      "Mississippi"
## [25] "Missouri"       "Montana"        "Nebraska"       "Nevada"
## [29] "New Hampshire"  "New Jersey"     "New Mexico"     "New York"
## [33] "North Carolina" "North Dakota"   "Ohio"           "Oklahoma"
## [37] "Oregon"         "Pennsylvania"   "Rhode Island"   "South Carolina"
## [41] "South Dakota"   "Tennessee"      "Texas"          "Utah"
## [45] "Vermont"        "Virginia"       "Washington"     "West Virginia"
## [49] "Wisconsin"      "Wyoming"
```

```
# List all column headers
names(USArrests)
```

```
## [1] "Murder"   "Assault"  "UrbanPop" "Rape"
```

Part 1: Calculate the mean and variance of each column

To compute the mean and variance of each column we can use the apply function, with the dataset and function as parameters. We will select 2 as our second parameter as we want to apply the function column-wise.

Calculate the mean.

```
apply(USArrests, 2, mean)
```

```
##   Murder  Assault UrbanPop     Rape
##    7.788  170.760   65.540   21.232
```

Calculate the variance.

```
apply(USArrests, 2, var)
```

```
##     Murder    Assault   UrbanPop       Rape
##   18.97047 6945.16571  209.51878   87.72916
```

Part 2: Dataset Analysis

If we look at the mean table, we can see that there is around 3 times more rapes than murders, & around 8 times more assaults than rapes.

When we consider the variance table, we notice that, for the 3 crimes, assault has by the largest variance, followed by rape. Generally the higher variance can be correlated higher mean crime rate.

We Should also note that the figures for Urban population have no real relationship with the crime averages as they are taken as a rate per 100000

As assault has by far the largest values for both tables, scaling will be required before PCA to avoid it dominating it.

Part 3: Principal Component Analysis

To perform Principal Component Analysis, we can call the prcomp function on the dataset. We has set the scale parameter to true, for the reasons given in part 2.

```
pca <- prcomp(USArrests, scale = TRUE)

pca
```

```
## Standard deviations:
## [1] 1.5748783 0.9948694 0.5971291 0.4164494
##
## Rotation:
##                 PC1        PC2        PC3         PC4
## Murder   -0.5358995  0.4181809 -0.3412327  0.64922780
## Assault  -0.5831836  0.1879856 -0.2681484 -0.74340748
## UrbanPop -0.2781909 -0.8728062 -0.3780158  0.13387773
## Rape     -0.5434321 -0.1673186  0.8177779  0.08902432
```

Part 4; Check the results, report the number of PCs and their center, scale, and rotation

We can pnow use our PCA model to extract some of the results of our analysis.

The Number of PCAs.

```
names(pca)
```

```
## [1] "sdev"     "rotation" "center"   "scale"    "x"
```

The PC Centers.

```
pca$center
```

```
##   Murder  Assault UrbanPop     Rape
##    7.788  170.760   65.540   21.232
```

The PC Scales.

```
pca$scale
```

```
##    Murder   Assault  UrbanPop      Rape
##  4.355510 83.337661 14.474763  9.366385
```

The PC Rotations.

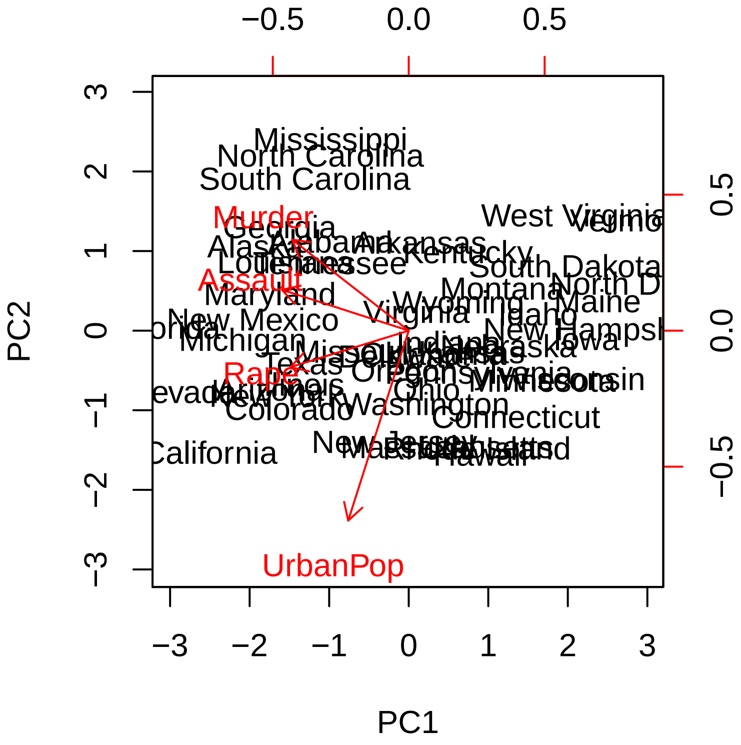
```r
pca$rotation
```

```
##                 PC1        PC2        PC3         PC4
## Murder   -0.5358995  0.4181809 -0.3412327  0.64922780
## Assault  -0.5831836  0.1879856 -0.2681484 -0.74340748
## UrbanPop -0.2781909 -0.8728062 -0.3780158  0.13387773
## Rape     -0.5434321 -0.1673186  0.8177779  0.08902432
```

Part 5: Plot of first 2 PCAs

To plot the first 2 PCs, we can use the biplot() function.

```r
biplot(pca, scale=0)
```



Part 6: Standard Deviation & Variance of components

We can also look at some statistical average from our PCA.

Standard Deviation.

```r
pca$sdev
```

```
## [1] 1.5748783 0.9948694 0.5971291 0.4164494
```

Variance.

```
pca_var = pca$sdev^2

pca_var
```

```
## [1] 2.4802416 0.9897652 0.3565632 0.1734301
```

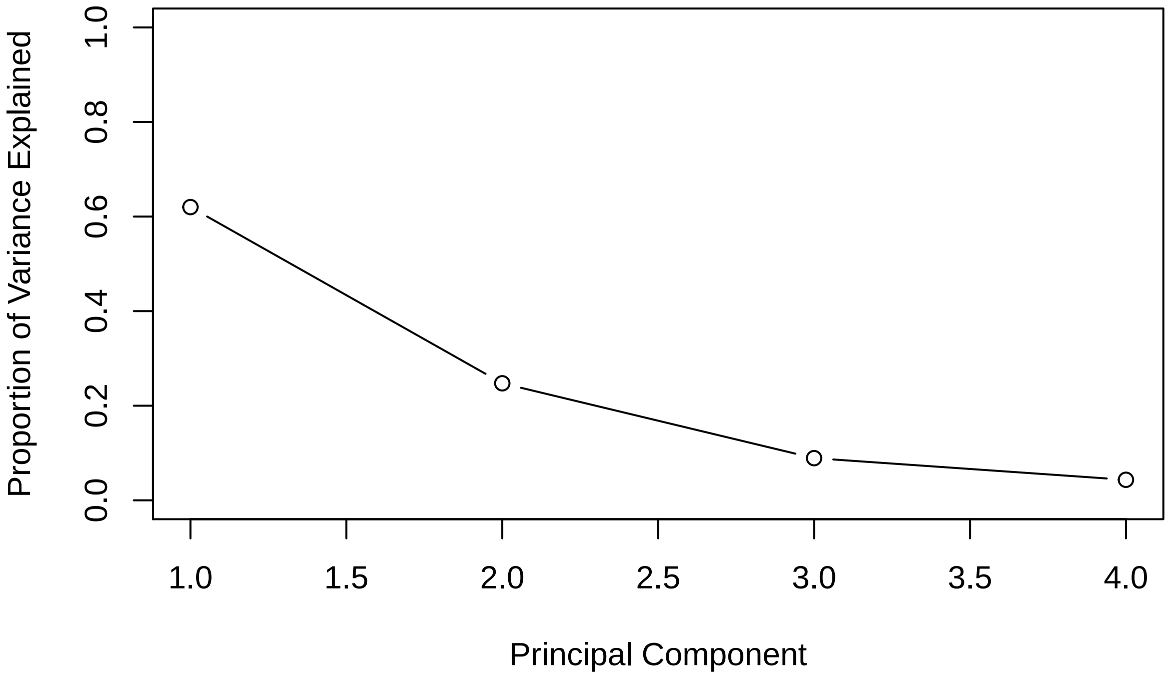
Part 7: PVE Plots

To decipher how much information from a dataset is lost by removing principal components, we can calculate the proportion of variance explained (PVE) for each component, and visualise the results in a scree plot.

```
# each component / total variance
pve = pca_var /sum(pca_var)
pve
```

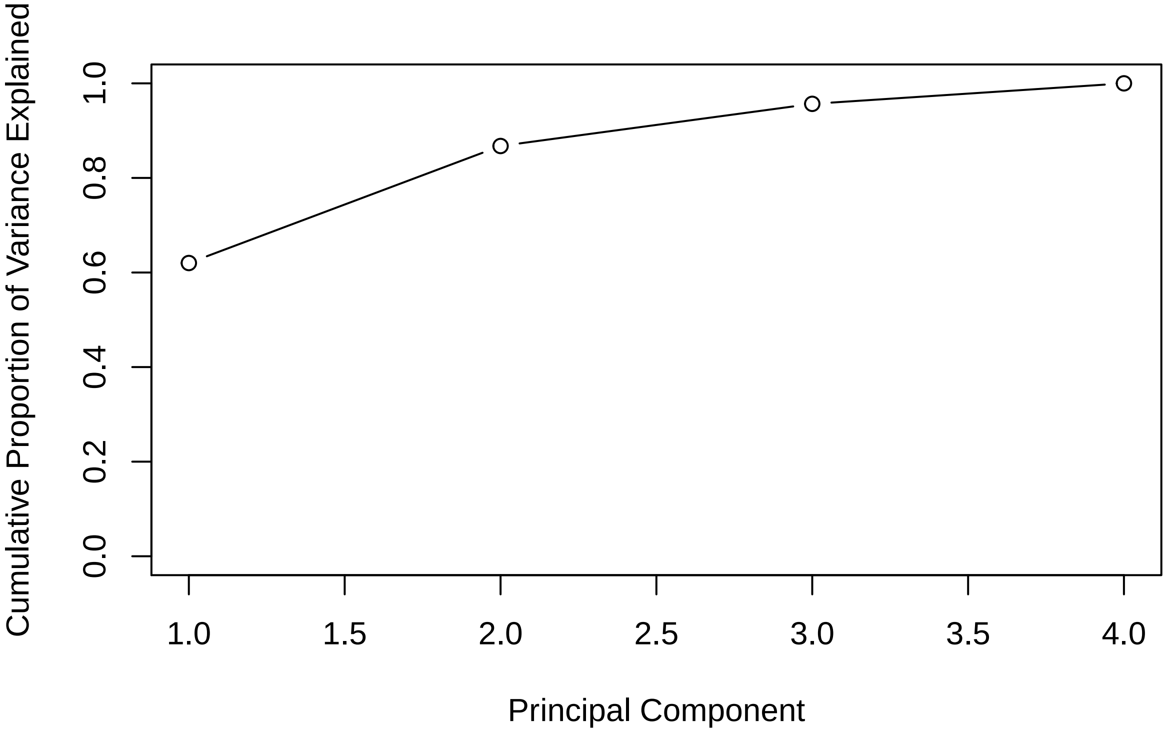
```
## [1] 0.62006039 0.24744129 0.08914080 0.04335752
```

```
plot(pve, xlab = " Principal Component ", ylab =" Proportion of Variance Explained ", ylim = c(0,1) ,typ
```



We can also plot the cumulative PVE of each principal component.

```
plot(cumsum (pve), xlab = " Principal Component ", ylab =" Cumulative Proportion of Variance Explained
```


Cumulative Proportion of Variance Explained
0.0
0.2
0.4
0.6
0.8
1.0
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Principal Component