Week 8 Principal Component Analysis

Unit Co-ordinator: Dr. Liwan Liyanage

School of Computer, Data and Mathematical Sciences

WESTERN SYDNEY UNIVERSITY

Unsupervised learning

- Recall unsupervised learning
- We are not interested in prediction as there is no response/target variable
- The goal is to discover interesting things about the measurements
  - Is there an informtive way to visualize the data?
  - Can we identify sub-groups among variables or among observations?
- Hard to assess the results obtained from supervised learning - (no way to measure prediction accuracy like in supervised learning)

Unsupervised learning

- ***Dimension Reduction***
  easier to see patterns in lower (two?) dimensions
- ***Clustering***
  automatically seek groups in the data

Dimension Reduction

- Most datasets in Data Science are ***High dimensional*** (lots of measurements), have a large number of measurements on many observations.
- In this lecture, we want to look at how to find a smaller number of summary variables that still capture the nature of the data.
- This is known as ***dimension reduction***
- Dimension reduction is a form of unsupervised learning.

WESTERN SYDNEY UNIVERSITY

Dimension Reduction

In Dimension reduction, the goal is to seek a low dimensional representation of the data that in some sense matches the full, high dimensional data set.

- Iris data has 4 variables; Is there a 2-D picture that displays all the structure?
- Advertising data has 4 variables; Is there a 2D picture ...?
- NC160 data - more than 6000 measurements (variables)

What do we mean by structure? We will look at two dimension reduction techniques

- ***PCA*** - Principal Component Analysis
- ***MDS*** - Multi-Dimensional Scaling

Principal Component Analysis

Principal Component Analysis finds a new variable that is

- a linear combination of the original variables
- and has maximum variance

First some revision. . . .

Revision

A data set consists of multiple measurements (or variables) on several observations.

The terms measurement and observations have synonyms in different areas of Data Science

- Variable - measurement, field, attribute, feature or column
- Observation - case, recored, instance, subject example or row

In Statistics, *observation* and *variable* are common.

Maths - the value of the j^{th} variable measured on the i^{th} observation is denoted by x_{ij}.

Revision (Continued...)

The mean of a variable is the average value, and is a measure of the location or central tendency of a variable.

$$\bar{X}_j = (X_{1j} + X_{2j} + ... + X_{nj})/n = 1/n \sum_{i=1}^{n} X_{ij}$$

(Add up over the observations for that variable and divide by the number of observations)

Revision (Continued. . .)

The variance of a variable is a measure of the spread of the variable around the mean.

$$s_j^2 = (x_{1j} - \bar{X}_j)^2 + (x_{2j} - \bar{X}_j)^2 + ... + (x_{nj} - \bar{X}_j)^2/(n-1)$$

$$s_j^2 = 1/(n-1)\sum_{i=1}^{n}(x_{ij} - \bar{X}_j)^2$$

*(Add up the squared differences between the observation and the mean and divide it by **n-1**)*

s_j, the square root of the variance, is called the standard deviation.

Principal Component Analysis (PCA)

In PCA, a new variable y_1 is defined so that for each observation i,

$$y_{i1} = a_{11}x_{i1} + a_{21}x_{i2} + ... + a_{p1}x_{ip} = \sum_{j=1}^{p} a_{j1}X_{ij}$$

$\sum_{j=1}^{p}(a_{j1})^2 = 1$ and the variance of y_1 is maximised.

(PCs are only defined upto a sign change)

2^{nd} Principal Component

After the first principal cmponent is defined, the second and subsequent can be defined.

$$y_{i2} = a_{12}x_{i1} + a_{22}x_{i2} + ... + a_{p2}x_{ip} = \sum_{j=1}^{p} a_{j2}X_{ij}$$

$\sum_{j=1}^{p}(a_{j2})^2 = 1, \sum_{j=1}^{p} a_{j1}a_{j2} = 0$ and the variance of y_2 is maximised.

WESTERN SYDNEY UNIVERSITY

k^{th} Principal Component

$$y_{ik} = \sum_{j=1}^{p} a_{jk} X_{ij}$$

$$\sum_{j=1}^{p} (a_{jk})^2 = 1$$

$$\sum_{j=1}^{p} a_{jk} a_{jm} = 0$$

for $m < k$, and the variance of y_k is maximised.

PCA in R . . .

Import and attach Iris data into R and see summary of data.

```r
library(ISLR)
```

```r
attach(iris)
summary(iris)
```

PCA in R . . .

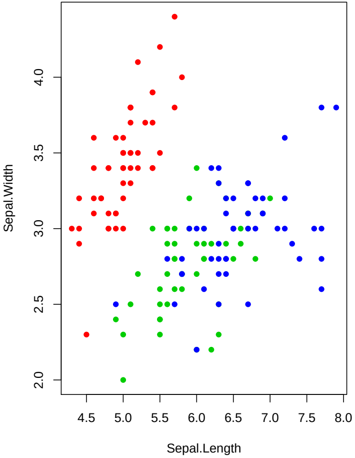
```
##   Sepal.Length    Sepal.Width     Petal.Length    Petal.Width
##  Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100
##  1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300
##  Median :5.800   Median :3.000   Median :4.350   Median :1.300
##  Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
##  3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
##  Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
##        Species
##  setosa    :50
##  versicolor:50
##  virginica :50
##
##
##
```

PCA in R ...

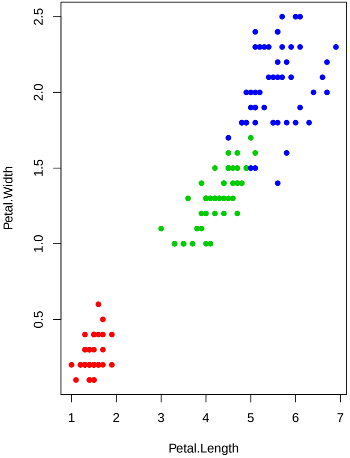
We can plot the original variables

```
par(mfrow=c(1,2))
plot(Sepal.Width~Sepal.Length, data=iris,
     col=unclass(Species)+1, pch=16)
plot(Petal.Width~Petal.Length, data=iris,
     col=unclass(Species)+1, pch=16)
```

PCA in R . . .



Sepal.Width
Sepal.Length
4.0
3.5
3.0
2.5
2.0
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0



Petal.Width
Petal.Length
2.5
2.0
1.5
1.0
0.5
1
2
3
4
5
6
7

PCA in R . . .

In R, there are (at least) two functions that do PCA

- *prcomp*
- *princomp*

They have slightly different interfaces.

Let's ignore the last column (Species) which is the target variable and consider this as an unsupervised problem. The first 4 columns of the iris data are numeric $iris[, 1 : 4]$.

```r
head(iris[,1:4])
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1          5.1         3.5          1.4         0.2
## 2          4.9         3.0          1.4         0.2
## 3          4.7         3.2          1.3         0.2
## 4          4.6         3.1          1.5         0.2
## 5          5.0         3.6          1.4         0.2
## 6          5.4         3.9          1.7         0.4
```

WESTERN SYDNEY UNIVERSITY

Mean and Variance of variables

```r
sapply(iris[,1:4],mean)
```

```
## Sepal.Length  Sepal.Width Petal.Length  Petal.Width
##     5.843333     3.057333     3.758000     1.199333
```

```r
sapply(iris[,1:4],var)
```

```
## Sepal.Length  Sepal.Width Petal.Length  Petal.Width
##    0.6856935    0.1899794    3.1162779    0.5810063
```

Scaling variables Sometimes, variables are scaled to have ***unit variance*** before PCA. This is usually done unless the original scale is meaningful in some way.

WESTERN SYDNEY UNIVERSITY

PCA in R ...

Let's perform PCA

```
obj = prcomp(iris[,1:4], scale. = TRUE) # perform PCA
```

PCA in R ...

```r
names(obj)
```

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

```r
obj$rotation
```

```
##                     PC1         PC2        PC3        PC4
## Sepal.Length  0.5210659 -0.37741762  0.7195664  0.2612863
## Sepal.Width  -0.2693474 -0.92329566 -0.2443818 -0.1235096
## Petal.Length  0.5804131 -0.02449161 -0.1421264 -0.8014492
## Petal.Width   0.5648565 -0.06694199 -0.6342727  0.5235971
```

The rotation matrix provides the principal component loadings; each column contains the corresponding principal component loading vector

PCA in R . . .

principal components

```r
head(obj$x)
```

```
##            PC1        PC2         PC3          PC4
## [1,] -2.257141 -0.4784238  0.12727962  0.024087508
## [2,] -2.074013  0.6718827  0.23382552  0.102662845
## [3,] -2.356335  0.3407664 -0.04405390  0.028282305
## [4,] -2.291707  0.5953999 -0.09098530 -0.065735340
## [5,] -2.381863 -0.6446757 -0.01568565 -0.035802870
## [6,] -2.068701 -1.4842053 -0.02687825  0.006586116
```

"sdev" gives the **standard deviations of the principal components**,

"center" gives the **means of the original variables** and

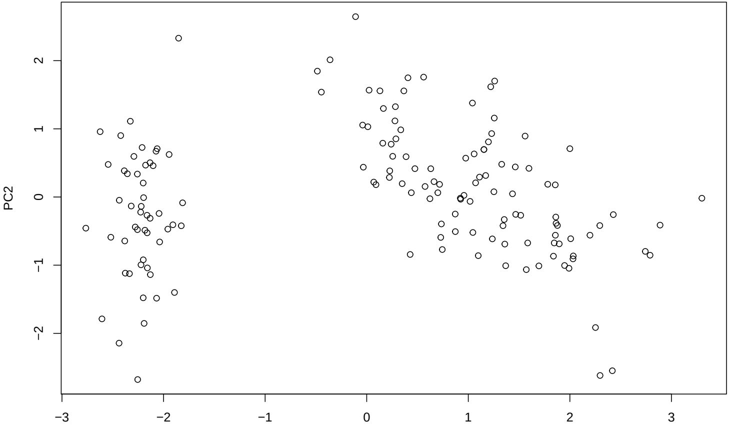
"scale" gives the **standard deviations of the original variables**.

WESTERN SYDNEY UNIVERSITY

PCA in R . . .

Plot the first two Principal Components.

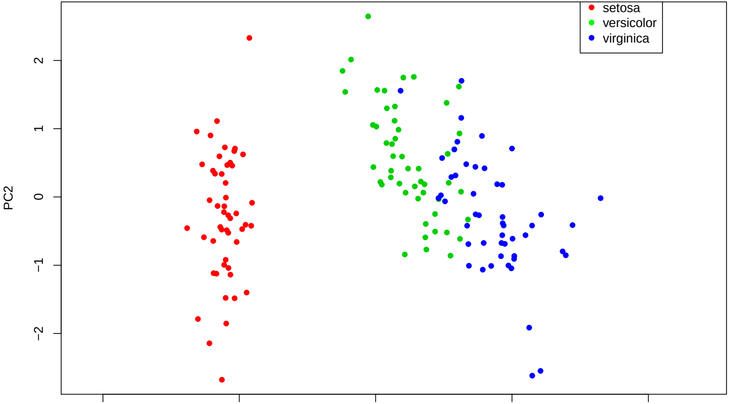
```
plot(obj$x[,1:2])
```



PCA in R . . .

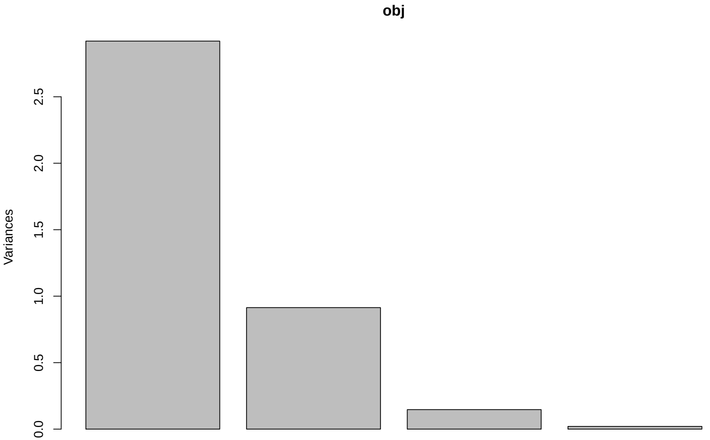
Since we have the species values, let's colour the observations according to the species category.

```
plot(obj$x[,1:2], col=unclass(iris$Species)+1, pch=16, asp=1)
legend(x=3, y = 3, c("setosa", "versicolor", "virginica"),
       col=c("red", "green", "blue"), pch = 16)
```



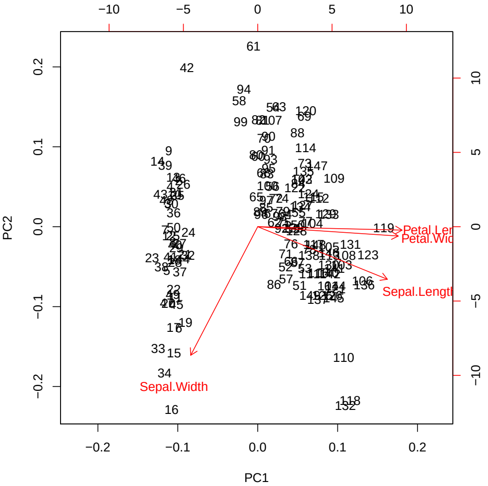
Scree plot

```r
screeplot(obj)
```



Biplot

```r
biplot(obj)
```



PCA - Propotion of Variance Explained (PVE)

Each PC has a variance, displayed in the *scree plot*. The percentage of the total variance for each PC, is called the **Propotion of Variance Explained**.

```
obj$sdev
```

```
## [1] 1.7083611 0.9560494 0.3830886 0.1439265
```

```
pr.var <- obj$sdev^2
pr.var
```

```
## [1] 2.91849782 0.91403047 0.14675688 0.02071484
```

```
pve <- pr.var/sum(pr.var)
pve
```

```
## [1] 0.729624454 0.228507618 0.036689219 0.005178709
```

```
cumsum(pve)
```

```
## [1] 0.7296245 0.9581321 0.9948213 1.0000000
```

Statistic	PC1	PC2	PC3	PC4
Standard deviation	1.708	0.956	0.383	0.144
Proportion of Variance	0.729	0.228	0.037	0.005
Cumulative Proportion	0.729	0.958	0.994	1.000

PCA example - US Arrests

```r
attach(USArrests)
head(USArrests)
```

```
##            Murder Assault UrbanPop Rape
## Alabama      13.2     236       58 21.2
## Alaska       10.0     263       48 44.5
## Arizona       8.1     294       80 31.0
## Arkansas      8.8     190       50 19.5
## California    9.0     276       91 40.6
## Colorado      7.9     204       78 38.7
```

```r
states <- row.names(USArrests)
head(states)
```

```
## [1] "Alabama"    "Alaska"     "Arizona"    "Arkansas"   "California"
## [6] "Colorado"
```

PCA example - US Arrests

```r
sapply(USArrests,mean)
```

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

```r
sapply(USArrests,var)
```

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

PCA example - US Arrests

```r
pr.out <- prcomp(USArrests, scale. = TRUE)
head(pr.out$x)
```

```
##                   PC1        PC2         PC3          PC4
## Alabama    -0.9756604  1.1220012 -0.43980366  0.154696581
## Alaska     -1.9305379  1.0624269  2.01950027 -0.434175454
## Arizona    -1.7454429 -0.7384595  0.05423025 -0.826264240
## Arkansas    0.1399989  1.1085423  0.11342217 -0.180973554
## California -2.4986128 -1.5274267  0.59254100 -0.338559240
## Colorado   -1.4993407 -0.9776297  1.08400162  0.001450164
```

PCA example - US Arrests

```r
pr.out$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
```

PCA example - US Arrests

```r
pr.var <- pr.out$sdev^2
pr.var
```

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

```r
pve <- pr.var/sum(pr.var)
pve
```

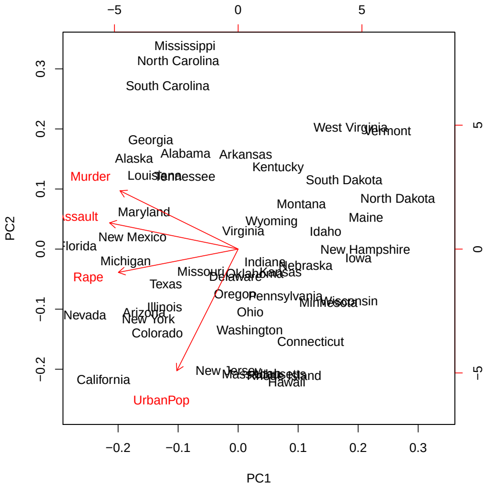
```
## [1] 0.62006039 0.24744129 0.08914080 0.04335752
```

```r
cumsum(pve)
```

```
## [1] 0.6200604 0.8675017 0.9566425 1.0000000
```

PCA example - US Arrests

```r
biplot(pr.out)
```



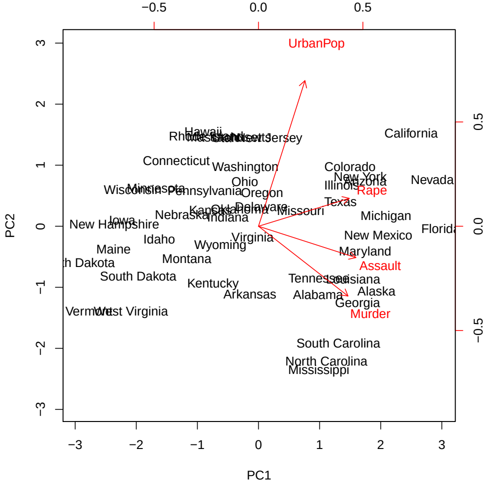
PCA example - US Arrests

Let's rotate the biplot for the convenience

```
pr.out$rotation <- -pr.out$rotation
pr.out$x <- -pr.out$x
biplot(pr.out)
```

WESTERN SYDNEY UNIVERSITY

PCA example - US Arrests

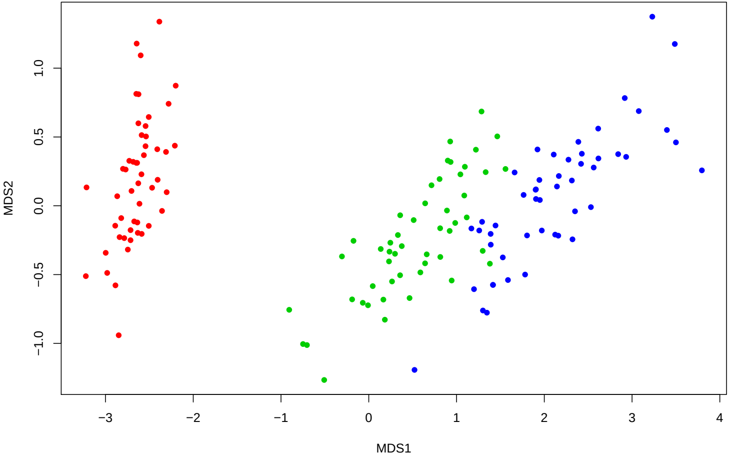


Multi-Dimensional Scaling - Example

For multidimensional scaling, the function "*cmdscale*" is used on the "*dist*" (distances).

```
mds = cmdscale(dist(iris[,1:4]), k=2)
plot(mds, col=unclass(iris$Species)+1,
     pch = 16, xlab = "MDS1", ylab = "MDS2")
```

Multi-Dimensional Scaling - Example (Continued. . .)



WESTERN SYDNEY UNIVERSITY

TEXT BOOK

Lecture notes are based on the textbook.

For further reference refer;

Prescribed Textbook - Chapter 10

- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An Introduction to Statistical Learning: with Applications in R Springer.