Dimensionality Reduction

Week 8- Principle Component Analysis

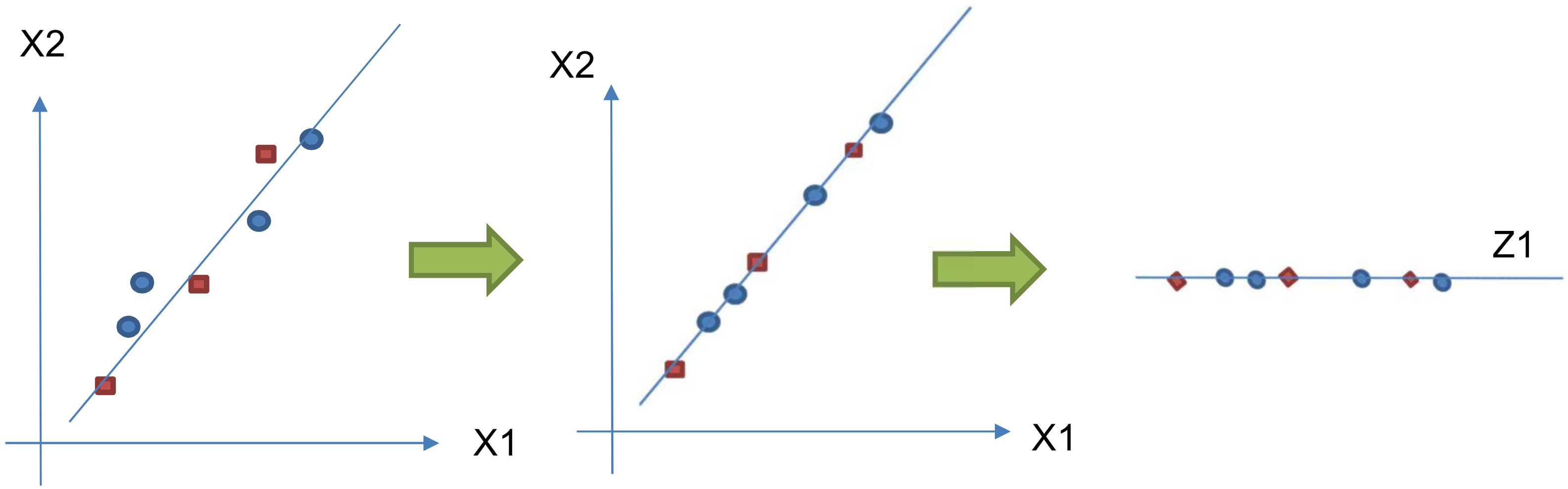
Mohammad Esmalifalak

McMaster University
CENTRE FOR CONTINUING EDUCATION

Why We Reduce Dimension?

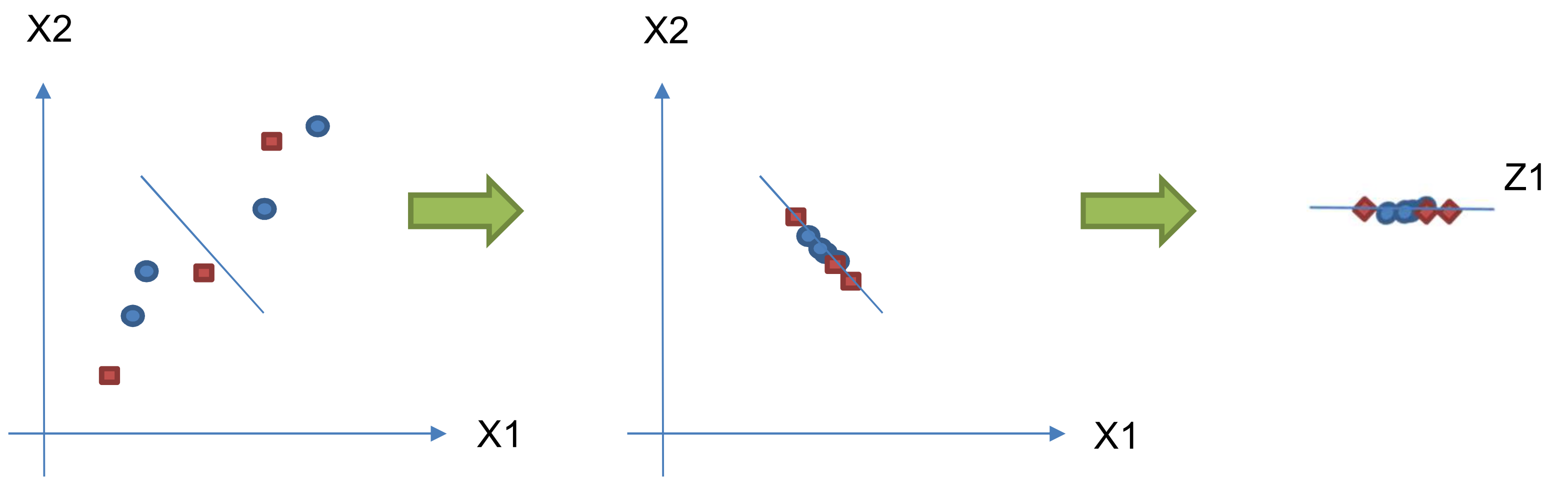
- Less Memory Usage
- Speeding up learning algorithms
- Visualization

Principle Component Analysis (2D)



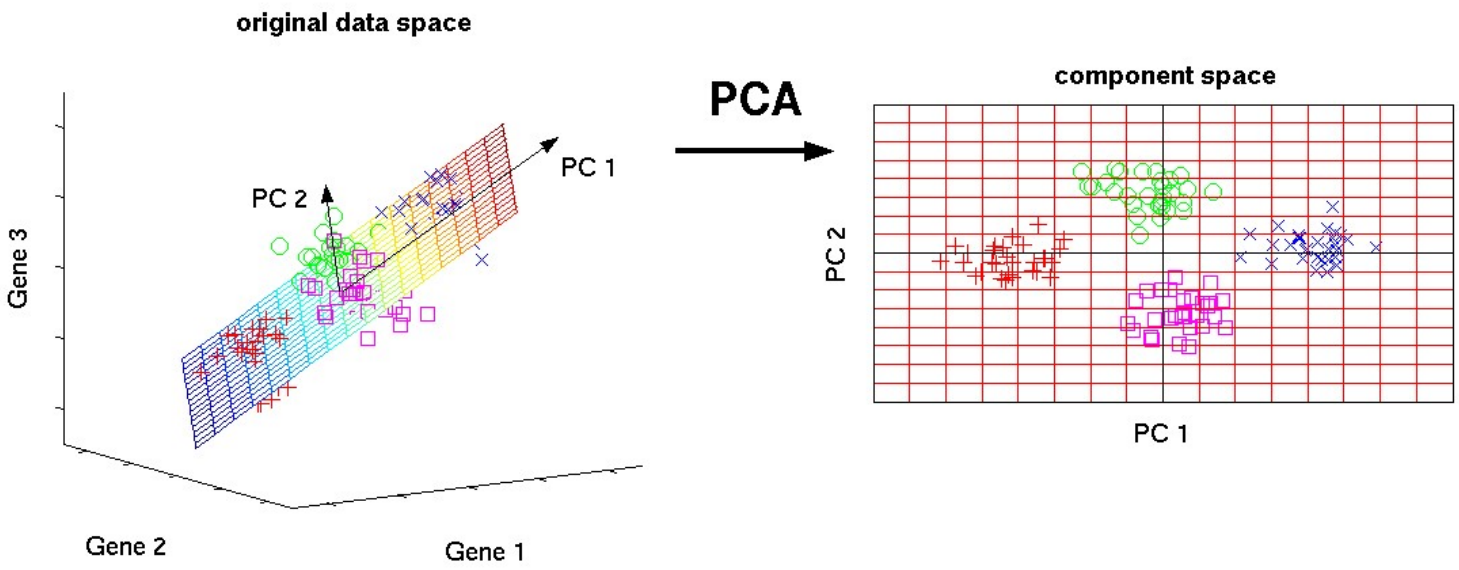
Reducing dimension from 2D to 1D will not cause two much of information loss

Principle Component Analysis (2D)



Reducing dimension from 2D to 1D could cause two much of information loss

Principle Component Analysis (3D)



PCA for Visualization

In [1]:

```
# Let's read in the dataset. I found this dataset on Kaggle
countries = read.csv('../input/countries of the world.csv', na.string = c("", "NA"))
```

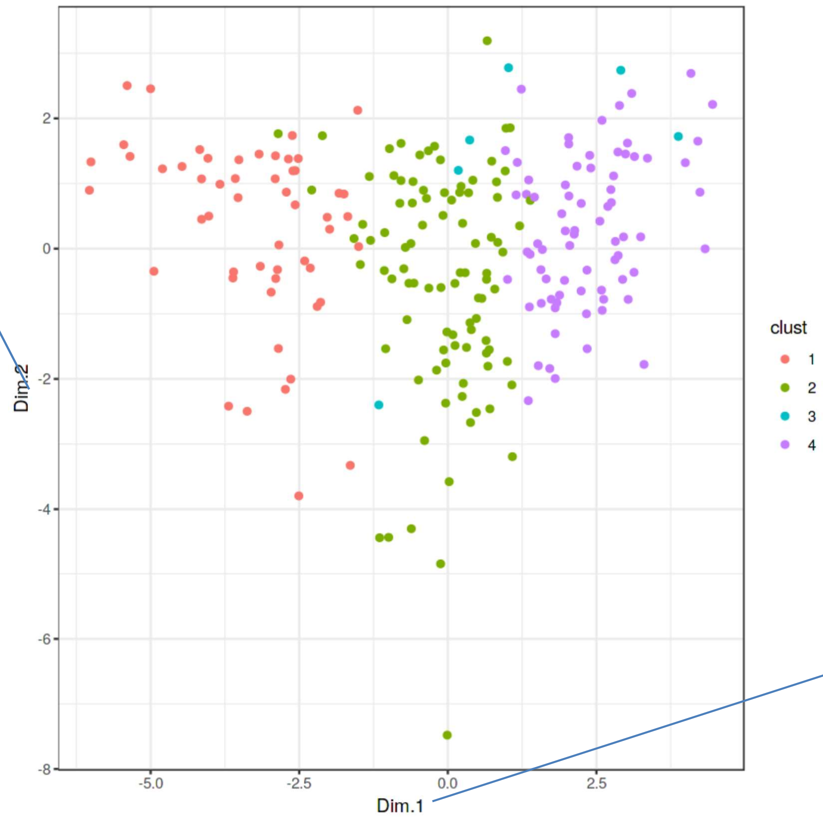
In [2]:

```
# Let's check the head of the dataframe
head(countries)
```

Country	Region	Population	Area..sq..mi..	Pop..Density..per.sq..mi..	Coastline..coast.area.ratio.	Net.migration	Infant.mortality..pe
Afghanistan	ASIA (EX. NEAR EAST)	31056997	647500	48,0	0,00	23,06	163,07
Albania	EASTERN EUROPE	3581655	28748	124,6	1,26	-4,93	21,52
Algeria	NORTHERN AFRICA	32930091	2381740	13,8	0,04	-0,39	31
American Samoa	OCEANIA	57794	199	290,4	58,29	-20,71	9,27
Andorra	WESTERN EUROPE	71201	468	152,1	0,00	6,6	4,05
Angola	SUB-SAHARAN AFRICA	12127071	1246700	9,7	0,13	0	191,19

PCA for Visualization

~ Per Person GDP



~ Country Size/GDP

PCA-Algorithm

1 Training set: $x^{(1)}, x^{(2)}, \ldots, x^{(m)}$

Preprocessing (feature scaling/mean normalization):

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_j^{(i)}$$

Replace each $x_j^{(i)}$ with $x_j - \mu_j$.

If different features on different scales (e.g., $x_1 =$ size of house, $x_2 =$ number of bedrooms), scale features to have comparable range of values.

2 `Sigma =` $\frac{1}{m}\sum_{i=1}^{m}(x^{(i)})(x^{(i)})^T$

3 `[U,S,V] = svd(Sigma);`

$x \in R^n \rightarrow z \in R^k$

U_k

$$U = \begin{bmatrix} | & | & & | \\ u^{(1)} & u^{(2)} & \ldots & u^{(n)} \\ | & | & & | \end{bmatrix} \in \mathbb{R}^{n \times n}$$

$$U_k^T \times X = Z$$

$$(k \times n) \times (n \times 1) = (k \times 1)$$

PCA-Choosing 'K'

```
Sigma =
```
$$\frac{1}{m}\sum_{i=1}^{m}(x^{(i)})(x^{(i)})^T$$

```
[U,S,V] = svd(Sigma);
```

$$\begin{pmatrix} S_{11} & & & 0 \\ & S_{22} & & \\ & & \ddots & \\ 0 & & & S_{nn} \end{pmatrix} \Rightarrow 1 - \frac{\sum_{i=1}^{k} S_{ii}}{\sum_{i=1}^{n} S_{ii}} < 0.01 \Rightarrow$$

99% of variance is retained

PCA-Reconstruction (X_{app})

$x \in R^n \rightarrow z \in R^k \quad \Rightarrow \quad z \in R^k \rightarrow x \in R^n$

$$\boxed{U_k^T \times X = Z}$$

$$\boxed{U_k \times Z = X_{app}}$$

$(k \times n) \times (n \times 1) = (k \times 1)$

$(n \times k) \times (k \times 1) = (n \times 1)$

References:

[1] Andrew Ng, Principal components analysis, CS229 Lecture notes

[2] http://www.nlpca.org

[3] SVD MIT OpenCourseWare

https://www.youtube.com/watch?v=EqSJLHkixSs