

Unsupervised Learning

Principal Components Analysis

Maarten Cruyff

Morning session

Principal Components Analysis

Principal components analysis (PCA) is a data reduction technique. It's aim is to capture the information present in a large number of variables in a much smaller number of principal components. For example, a personality test may consist 5 groups of 10 questions each, with each group measuring a different personality trait. If the test is constructed well, the PCA will identify 5 principal components that each measure a different personality trait. By saving the individual scores on these principal components the dimension of the data is reduced from 50 to 5 variables, and the 5 variables can be used to make individual personality profiles.

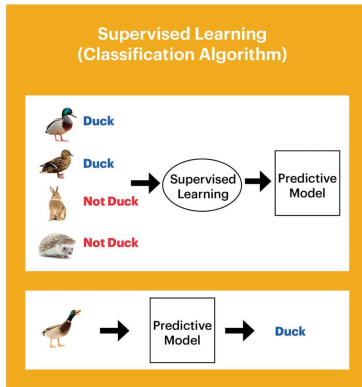
Course materials

- [Lecture sheets](#)
- [R lab](#)
- [R Markdown lab template](#)

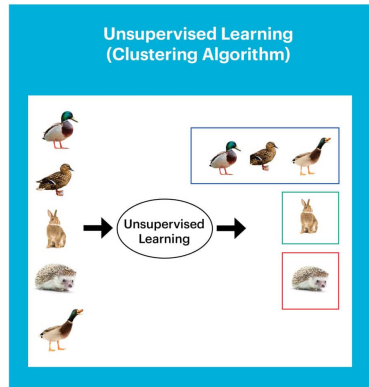
1. Unsupervised learning
2. Principal components analysis
3. Iris data

Supervised vs unsupervised

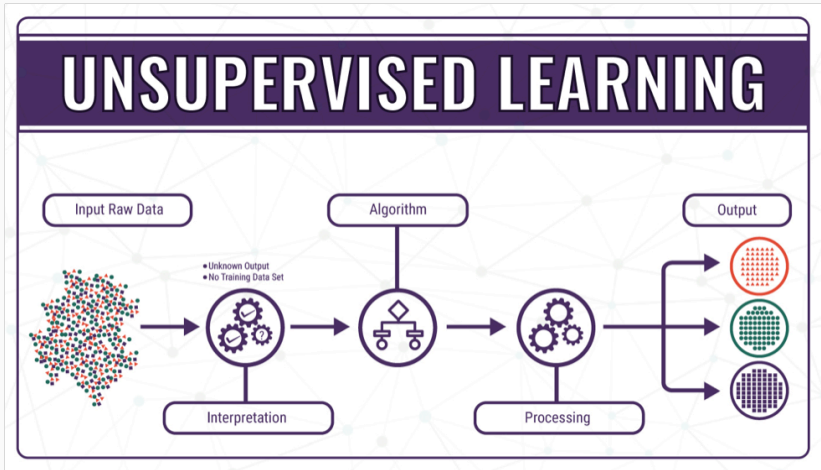
Known outcome



Unknown outcome



Western Digital.



Principal Components Analysis

How to make customer profiles?

Profiles of bought products

- many products but only a few profiles



What is Principal Components Analysis

Data of high dimension

- many variables containing different information
- high correlations within groups of variables
- low correlations between groups of variables

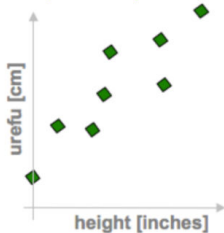
Dimension reduction

- groups of correlated variables form a single principal component
- different groups form different principal components

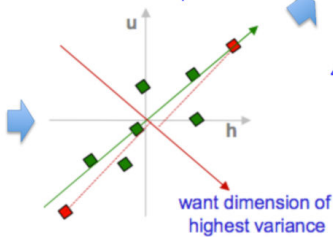
PCA in a nutshell

1. correlated hi-d data

("urefu" means "height" in Swahili)



2. center the points



3. compute covariance matrix

$$\begin{matrix} h & u \\ h & \begin{pmatrix} 2.0 & 0.8 \\ 0.8 & 0.6 \end{pmatrix} \end{matrix} \rightarrow \text{cov}(h, u) = \frac{1}{n} \sum_{i=1}^n h_i u_i$$

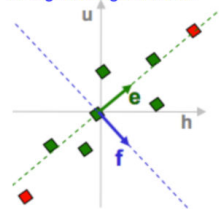
4. eigenvectors + eigenvalues

$$\begin{pmatrix} 2.0 & 0.8 \\ 0.8 & 0.6 \end{pmatrix} \begin{bmatrix} e_h \\ e_u \end{bmatrix} = \lambda_e \begin{bmatrix} e_h \\ e_u \end{bmatrix}$$

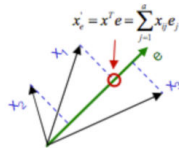
$$\begin{pmatrix} 2.0 & 0.8 \\ 0.8 & 0.6 \end{pmatrix} \begin{bmatrix} f_h \\ f_u \end{bmatrix} = \lambda_f \begin{bmatrix} f_h \\ f_u \end{bmatrix}$$

$\text{eig}(\text{cov}(\text{data}))$

5. pick $m < d$ eigenvectors w. highest eigenvalues

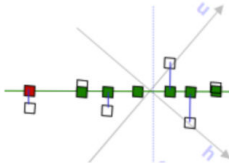


6. project data points to those eigenvectors



Copyright © 2011 Victor Lavrenko

7. uncorrelated low-d data



Principal components

The p principal component Z_j are computed as:

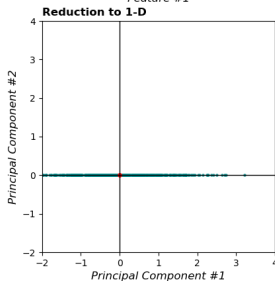
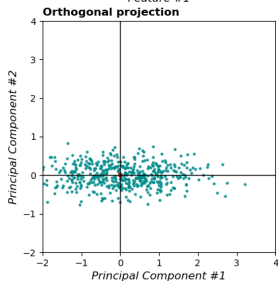
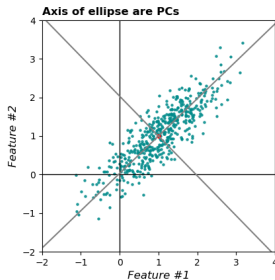
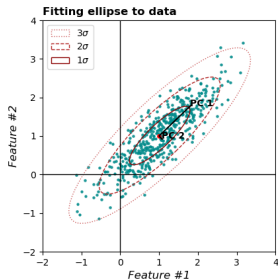
$$Z_j = \phi_{1j}X_1 + \dots + \phi_{pj}X_p, \quad j = 1, \dots, p$$

- Z_j is weighted sum of the variables
- ϕ_{1j} is loading of X_1 and Z_j (like a correlation)

The loadings are computed such that

- Z_1, \dots, Z_p are ordered in terms of explained variance

PCA in two-dimensional space



PCA in high-dimensional space

Properties of principal components (PCs)

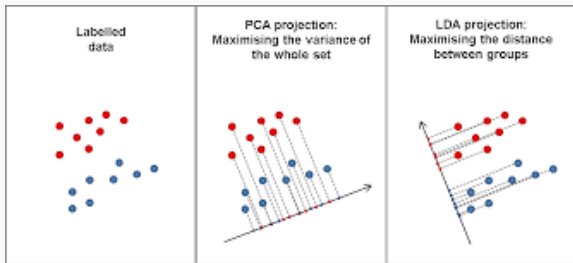
- explained variance is expressed in *eigenvalues*
- sum of eigenvalues is p (number of variables)
- PCs with eigenvalues > 1 are considered informative
- PCs with eigenvalues < 1 are considered noise

Dimension reduction criteria, retain PC's

- with eigenvalues > 1
- above the elbow in the scree plot

PCA vs LDA

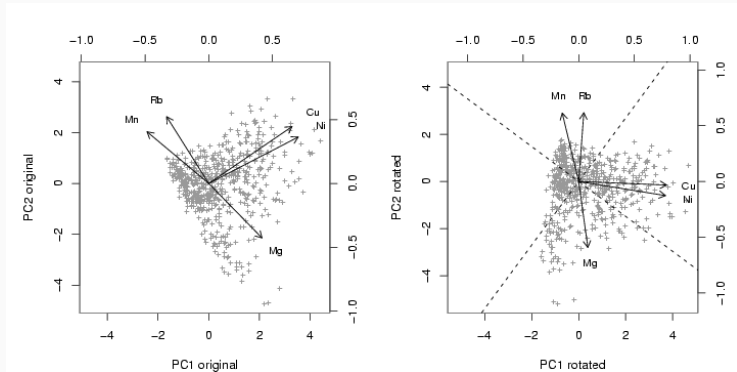
- PCA maximizes variance between data points
- LDA maximizes variance between groups



Rotation

Facilitates interpretation of PC's

- maximizes loading on one PC
- minimizes loadings on others



Iris data

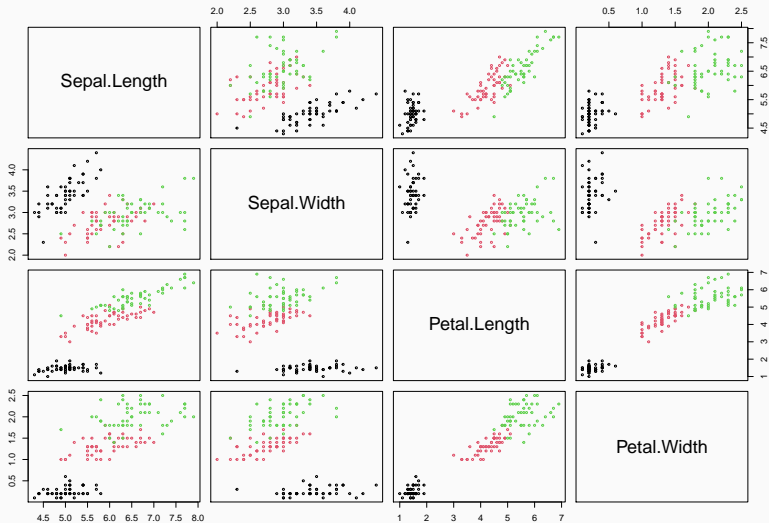
Example iris data

4-dimensional data:

- 150 Iris flowers (3 species)
- 4 features
 - petal length
 - petal width
 - sepal length
 - sepal width



Feature structure in 4 dimensions



PCA solution with 4 components

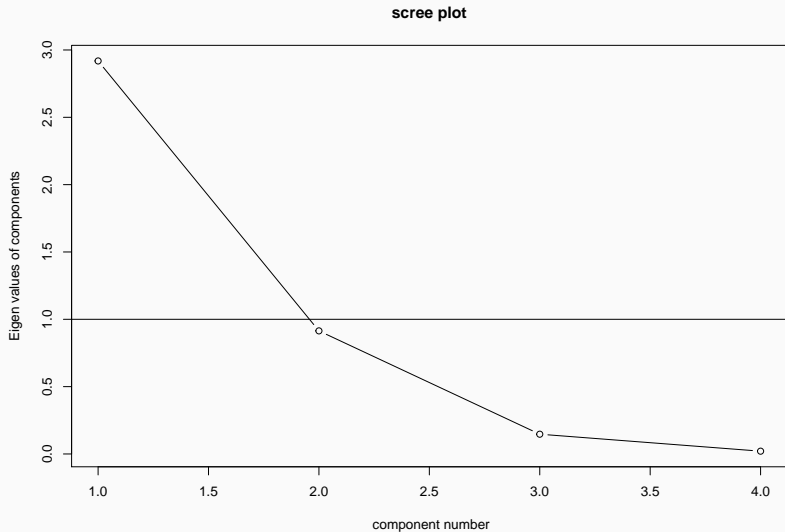
```
pc4 <- principal(iris[, 1:4], nfactors = 4, rotate = "none")  
  
print(loadings(pc4), cutoff = .1)
```

Loadings:

	PC1	PC2	PC3	PC4
Sepal.Length	0.890	0.361	-0.276	
Sepal.Width	-0.460	0.883		
Petal.Length	0.992			0.115
Petal.Width	0.965		0.243	

	PC1	PC2	PC3	PC4
SS loadings	2.918	0.914	0.147	0.021
Proportion Var	0.730	0.229	0.037	0.005
Cumulative Var	0.730	0.958	0.995	1.000

The four eigenvalues

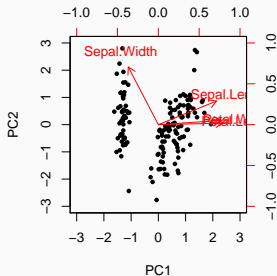


Conclusion: One PC needed, but let's select two

Unrotated loadings

```
pc2 <- principal(iris[, -5], nfactors = 2, rotate = "none")  
round(loadings(pc2)[1:4, 1:2], 2)
```

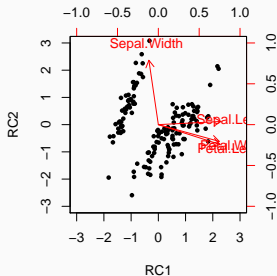
	PC1	PC2
Sepal.Length	0.89	0.36
Sepal.Width	-0.46	0.88
Petal.Length	0.99	0.02
Petal.Width	0.96	0.06



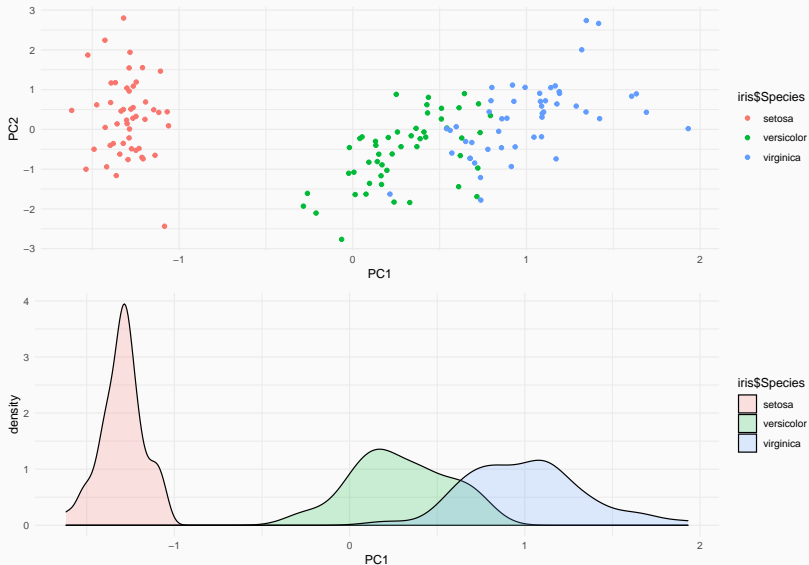
Rotated loadings

```
pc2rot <- principal(iris[, -5], nfactors = 2, rotate = "varimax")  
round(loadings(pc2rot)[1:4, 1:2], 2)
```

	RC1	RC2
Sepal.Length	0.96	0.05
Sepal.Width	-0.14	0.98
Petal.Length	0.94	-0.30
Petal.Width	0.93	-0.26



PC scores with species in 1 and 2 dimensions



Large data sets

Function `prcomp()` handles data with $p > n$

```
pc2 <- prcomp(x = iris[, -5])
```