Principal component analysis

What is Principal Component Analysis?

Principal component analysis : data and goals

- Data : n observations characterized by d quantitative variables. The raw data matrix is denoted $\mathbf{R}$
- Goal : dimension reduction.
- We want to summarize the observations using a small number k of synthetic variables called the principal components obtained as linear combinations of the initial variables
- Principal component analysis allows to
  - Compress the data set, keeping the initial structure of the data set
  - Visualize in low dimension how is organized the data set

A toy example

Let us consider a data set describing tree kinds of leafs coming from the website : `https://archive.ics.uci.edu/ml/datasets/Leaf`

An example

We focus on the two following variables

- Elongation : maximal normalized distance between a point of the leaf and its boundary
- Isoperimetric factor : ratio between the area and the square of the perimeter of the leaf

A toy example

Leaf Data Set

Download: Data Folder, Data Set Description

Abstract: This dataset consists in a collection of shape and texture features extracted from digital images of leaf specimens originating from a total of 40 different plant species.

Data Set Characteristics:	Multivariate	**Number of Instances:**	340	**Area:**	Computer
Attribute Characteristics:	Real	**Number of Attributes:**	16	**Date Donated**	2014-02-24
Associated Tasks:	Classification	**Missing Values?**	N/A	**Number of Web Hits:**	88647

Source:

This dataset was created by Pedro F. B. Silva and AndrÃ© R. S. MarÃ§al using leaf specimens collected by Rubim Almeida da Silva at the Faculty of Science, University of Porto, Portugal.

Data Set Information:

For further details on this dataset and/or its attributes, please read the 'ReadMe.pdf' file included and/or consult the Master's Thesis 'Development of a System for Automatic Plant Species Recognition' available at [Web Link].

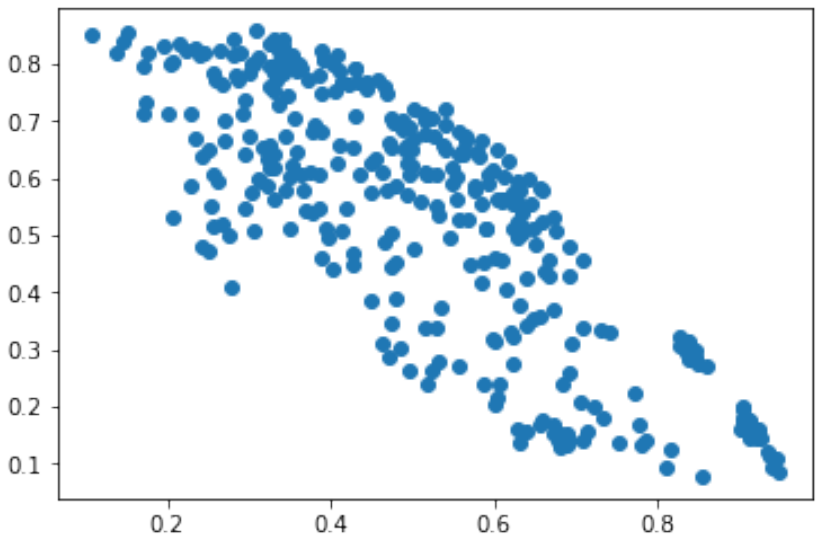
Attribute Information:

1. Class (Species)
2. Specimen Number
3. Eccentricity
4. Aspect Ratio
5. Elongation
6. Solidity
7. Stochastic Convexity
8. Isoperimetric Factor
9. Maximal Indentation Depth
10. Lobedness
11. Average Intensity
12. Average Contrast
13. Smoothness
14. Third moment
15. Uniformity
16. Entropy

A toy example
Vizulisation with Python

```
import numpy as np
leaf =
np.loadtxt('/home/marianne/Desktop/Enseignement
/2017-2018/S2/M1-AD/TP1/leaf.csv', delimiter=',')
import matplotlib.pyplot as plt
fig, ax = plt.subplots()
ax.scatter(leaf[:,4],leaf[:,7])
plt.show()
```

A toy example
Vizualisation with Python



The Leaf data set

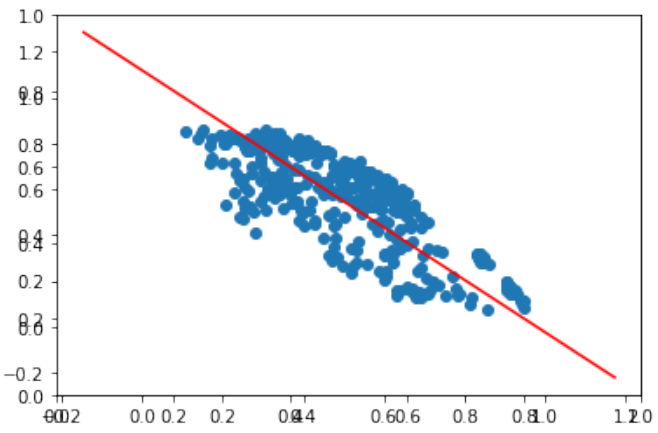
A toy example

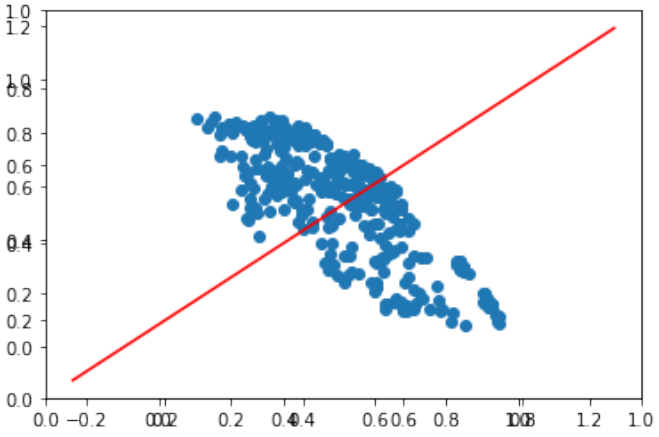
Some questions

- How can we summarize properly using only one variable the information in this data set?
- Which variable allows to separate in the best possible way the data?
- Can we find an orientation along which the variance of the data is much higher ?

A toy example

Several possibilities....
...the axis of the figure on the left seems to be the best one!



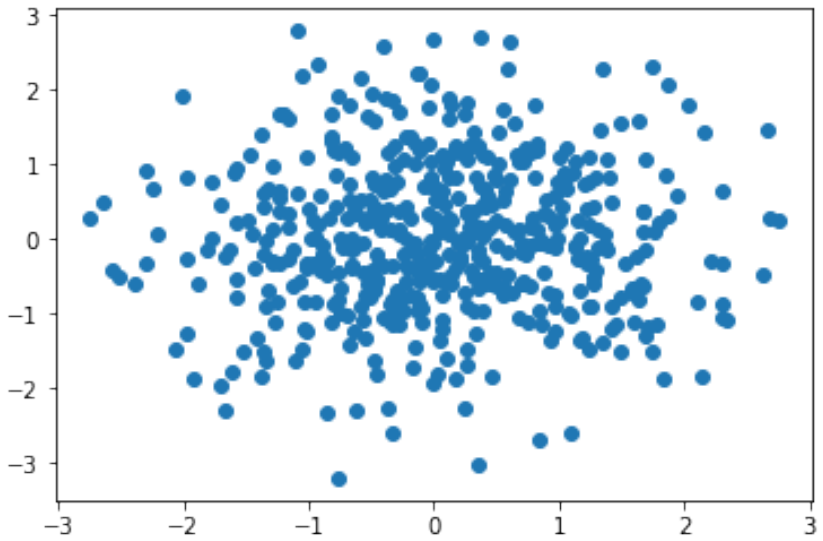


A toy example

We try with a synthetic dataset!

```
rndn = np.random.randn(500,2)
fig, ax = plt.subplots()
ax.scatter(rndn[:,0],rndn[:,1])
plt.show()
```

A toy example



Not always possible to find an axis separating properly the data!

Principal component analysis with Python

We import the data from the website :
https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data

```
import pandas as pd
url = "https://archive.ics.uci.edu/ml/
machine-learning-databases/ iris/iris.data"
df = pd.read_csv(url, names=['sepal length','sepal
width','petal length','petal width','target'])
```

Principal component analysis with Python

Preprocessing

```
features = ['sepal length', 'sepal width', 'petal
length', 'petal width']
x = df.loc[:, features].values
y = df.loc[:,['target']].values
from sklearn.preprocessing import StandardScaler
x = StandardScaler().fit_transform(x)
```

Principal component analysis with Python

We apply PCA

```
from sklearn.decomposition import PCA
pca = PCA(n_components=2)
principalComponents = pca.fit_transform(x)
principalDf = pd.DataFrame(data =
principalComponents, columns = ['principal component
1', 'principal component 2'])
```

Principal component analysis with Python

We use it for visualisation

```
fig, ax = plt.subplots()
ax.scatter(principalComponents[:,0],principalComponents[:,1
ax.set_xlabel('Principal Component 1', fontsize = 15)
ax.set_ylabel('Principal Component 2', fontsize = 15)
ax.set_title('Two-component PCA', fontsize = 20)
plt.show()
```

Principal component analysis with Python

Link with different species?

```
finalDf = pd.concat([principalDf, df[['target']]],
axis = 1)
targets = ['Iris-setosa', 'Iris-versicolor',
'Iris-virginica']
```

Principal component analysis with Python

Link with different species?

```
fig = plt.figure(figsize = (8,8))
ax = fig.add_subplot(1,1,1)
ax.set_xlabel('Principal Component 1', fontsize = 15)
ax.set_ylabel('Principal Component 2', fontsize = 15)
ax.set_title('Two-component PCA', fontsize = 20)
```

Principal component analysis with Python

Link with different species?

```
colors = ['r', 'g', 'b']
for target, color in zip(targets,colors):

   indicesToKeep = finalDf['target'] == target

   ax.scatter(finalDf.loc[indicesToKeep, 'principal
component 1']

      , finalDf.loc[indicesToKeep, 'principal
component 2']

      , c = color

      , s = 50)
ax.legend(targets)
ax.grid()
```