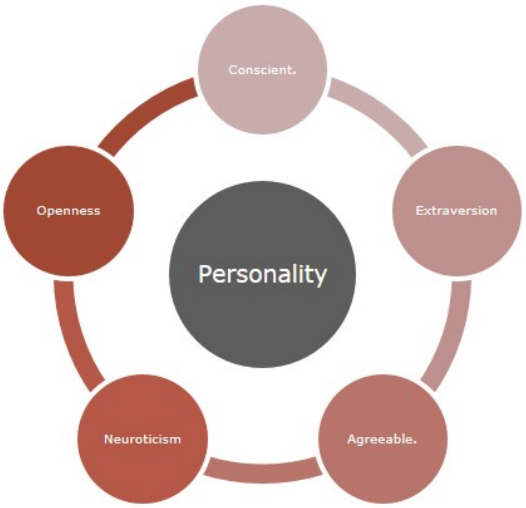
Feature Extraction: Principal Component Analysis Exercise #1

Copenhagen Summer University on Big Data

April 29, 2019



Introduction

We shall in the following exercise perform basic feature extration from personality questionnaire data. The features will be used as a predictor for gender using any of the statistical models you have learned, e.g. basic classification using logistic regression or later in the week using support vector machines or the deep learning interface Keras. We shall use the following libraries

```
import pandas as pd
from sklearn import preprocessing
import numpy as np
import matplotlib.pylab as plt
from sklearn.decomposition import PCA as PCA
```

The data file you will be using is saved as an R data frame and can be accessed by the command

```
# Load the data
dat1=pd.read_table("http://www.nbi.dk/~mathies/RGender.dat", delimiter=' ')
```

Note that "read_table()" will work for most basic text files arranged in rows/columns. For help or more information about a command in R, you can in the console type. The data file "RGender.dat" contains anynomenized answers from 940 individuals to 43 items in the Big Five Inventory together with information about gender. Basic information about the data file can be achieved by the commands

```
# shape of the data
dat1.shape
# The row names (features)
dat1.index
```

The data for individuals are given along the columns whereas the rows contain answers to items in the inventory. The answers are given by single digits ranging from 0 to 4 indicating to which extent a person agrees with a certain statement. For example, items could be "*I am quick to understand things*" or "*I pay attention to details*". Broadly speaking, the Big Five theory suggests that personality is spanned by a five dimensional space where the dimensions are defined as **1)** Openness to experience (curiosity), **2)** Conscientiousness (organized), **3)** Extraversion (outgoing), **4)** Agreeableness (cooperative, friendly) and **5)** Neuroticism (anxiety).

Principal Component Analysis

We shall first try to see if the answers to the 43 items "live" in a five dimensional space. For that purpose, we will use principal component analysis. The eigenvalues or variances associated to each direction in the rotated space will provide a rough indication of the dimensionality of the space. We first split the data file in two, containing respectively the 43 items and the gender,

```
### Make a feature and target dataframe
bfi = dat1.iloc[1::,:]
gender  = dat1.iloc[0:1,:]
```

The principal component analysis is performed after some basic rescaling of the data

```
### Scaling!
p1 = preprocessing.scale(bfi.transpose(),
                         axis=0,
                         with_mean=True,
                         with_std=True,
                         copy=True)
np.shape(p1)
# Perform the PCA
pca = PCA()
transformed = pca.fit_transform(p1)
```

The standard deviation of each component can be plotted according to size by

```
# The standard deviation of each component
# can be plotted according to size by
plt.figure()
plt.plot(np.std(transformed, axis = 0), '.')
```

Dimensionality of feature space

From the principal component analysis of the data, does it seem reasonable to claim that personality is five dimensional?

Gender prediction

From a reduced feature space, we now test whether it is possible to predict gender based on the Big Five variables. For this exercise you are free to use whichever method you like and also to test across various dimensions of feature space. For example, we can make a classification based on linear regression from the generalized linear model (glm) in R

```
# generate a data frame with gender as target and "k" leading PCs
k=5
d1 = transformed[:,0:k]
# for a linear model, we do not have to split
# in training and validation, look up the command cv.glm
dat_learn=d1
dat_valid=d1
# generalized linear model

from sklearn import linear_model
g1 = linear_model.LinearRegression()
g1.fit(d1, gender.transpose())
```

You can now make a prediction on the validation set with the command

```
probabilities = g1.predict(d1)
prediction = g1.predict(d1) > 0.5
prediction[0:5]

is_prediction_equal_to_actual_gender =
  np.array(gender == 1)[0] == prediction.transpose()
percent_right =
  np.sum(Is_prediction_equal_to_actual_gender) / len(prediction)
```

Gender prediction

Use the principal components to learn a statistical model and use the model to assess how well one can predict gender using as input the Big Five personality items. Try with different models, e.g. GLM and SVM and compute the corresponding ROC curves.