Exercises

SC2_10 – Principal Component Analysis.

1. Write a MATLAB function to implement the "*Incremental PCA algorithm*" (where the maximum eigenvalue and its eigenvector are computed by a function implementing the "power method"). Compare its results to those returned by the MATLAB `pca()` function. Compute also the function execution time for the "*Incremental PCA algorithm*" by means of: `tic` ⟨code to be evaluated⟩ `T=toc`, and compare it to the time required by the `pca()` function. The following code shows how to generate the random data (from a statistical point of view) and how to use the functions `tic` ... `T=toc` for the elapsed time.

```
N=50; % or more (number of samples)
% mean (mu) and covariance matrix (Sigma) of population
mu=[3 1 2 0]; Sigma=[1 .2 .7 .3; .2 1 0 .6; .7 0 1 .5; .3 .6 .5 1];
X=mvnrnd(mu,Sigma,N); % sample from a multivariate normal distribution N(μ,Σ)
 tic
     [basis,comp,lambda]=pca(X); % PCA of sample matrix
 T=toc;
fprintf("Elapsed time of pca() on %d samples: %g\n", N,T)
```

2. For the same samples as in the previous exercise, implement the *PCA algorithm* by computing eigenvalues and eigenvectors by means of the following MATLAB functions:
   - `pca()`: PCA applied to the data matrix, to the centered data matrix and to the standardized matrix;
   - `svd()`: SVD factorization of the centered matrix and of the standardized matrix;
   - `eig()`: eigenvalues/eigenvectors of the covariance matrix and of the correlation matrix;

   and compare their results.
   Also compare their execution times by means of: `tic` ⟨code to be evaluated⟩ `T=toc`.

3. Implement the "*Eigenfaces*" *algorithm* for face recognition using the MATLAB `pca()` function. For the exercise data (the face image database) download the `db_400faces_112x92_col_uint8.mat` file from the course page on the e-Learning platform.

4. Implement the "*Eigenfaces*" *algorithm* by computing eigenvalues and eigenvectors by means of the following MATLAB functions:
   - `pca()`: PCA applied to the data matrix, to the centered data matrix and to the standardized matrix;
   - `svd()`: SVD factorization of the centered matrix and of the standardized matrix;
   - `eig()`: eigenvalues/eigenvectors of the covariance matrix and of the correlation matrix;

   and compare their results.
   Also compare their execution times by means of: `tic` ⟨code to be evaluated⟩ `T=toc`.

5. Compute and display the regression line of y on x, and that of x on y for the following `X` data. Also compute and display the value of the functional $J_{LS}(a,b)$.

```
N=20; % or more (number of samples)
% mean (mu) and covariance matrix (Sigma) of population
mu=[1 2]; Sigma= [1 .2; .2 .7];
X=mvnrnd(mu,Sigma,N); % sample from a multivariate normal distribution N(μ,Σ)
```

6. Compute and display the regression plane of z on x,y, that of x on y,z and that of y on x,z for the following **X** data. Also compute and show the values of the three corresponding functionals $J_{LS}(a,b,c)$.

```
N=20; % or more (number of samples)
% mean (mu) and covariance matrix (Sigma) of population
mu=[3 1 2]; Sigma= [1 .2 .8; .2 1 0; .8 0 1];
X=mvnrnd(mu,Sigma,N); % sample from a multivariate normal distribution N(μ,Σ)
```

7. For the same data as in the previous exercise, compute and display the PCA plane (spanned by the first two principal directions) and the Least Squares plane of z on x,y. What optimal condition does each plan satisfy with respect to the samples? How to check it?