Skills Class Signal & Image Processing – Principal Component Analysis

Pietro Bonizzi & Joël Karel

Look at the documentation of the Matlab function *eig*. This function produces a diagonal matrix `D` of eigenvalues and a full matrix `V` whose columns are the corresponding eigenvectors so that `X*V = V*D`.

Question 1

Build the bi-variate time series `X = [x1;x2]`, composed by 100 observation, as follows:

```
>> c1 = [2 8];
>> x1 = randn(1,100);
>> x2 = polyval(c1,x1)+randn(size(x1));
>> X = [x1;x2];
```

And center each time series with respect to its mean, as follows:

```
>> [m, n] = size(X);
>> mx = mean(X,2);
>> Xo = X - mx*ones(1, n);
```

Plot the original and the centered data on the same figure, so as to generate a figure like Fig. 1.

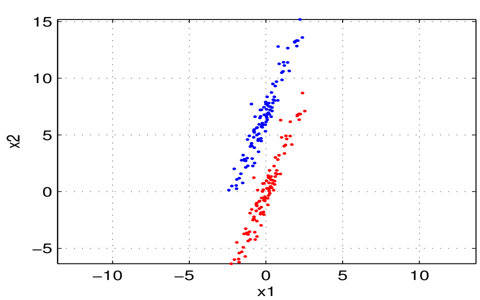

Figure 1: plot of the original and centered data.

Question 2

Apply function *eig* to the correlation matrix `Rx` of `Xo` so as to achieve its eigenvalue decomposition, as follows:

```
>> N = length(Xo);
>> Rx = Xo*Xo'/N ;
>> [U, D] = eig(Rx);
```

Verify that matrix `U` is orthogonal and look at matrix `D`: what do you observe and why?

Question 3

In the course we have seen that the PCA model can be described as: $X = AZ$. Generate the matrix of principal components `Z` and the transfer matrix `A`, collecting the corresponding eigenvectors, by exploiting the solution for the EVD model presented in the course, and assuming the covariance matrix of `Z` to be $Rz = \frac{ZZ^T}{N-1} = I$ (with I being the identity matrix, and N the number of samples in the multivariate signal Z).

Be careful: To solve the indeterminacy on the order of the components, reorganize the eigenvalues in matrix `D` and the corresponding eigenvectors in matrix `U` in decreasing order (use functions *fliplr* and *flipud* appropriately).

Question 4

Given the expressions for `A` and `Z` at the previous question, generate the following figures:

- Fig. 2: plot the centered data together with the first and second eigenvectors, using function `vectarrow`, which you can find in the folder `AssignmentPCA.rar` (e.g.: `vectarrow([0 0],A(:,1)')`). Notice the orthogonality of their directions!
- Fig. 3: plot the second principal component versus the first, what do you notice?
- Fig. 4: plot the projection of the first principal component only on the uncentered original subspace as showed in class.

What do you obtain? Why? Try to comment on the result of this projection.

Question 5

Repeat questions 2 to 4, but this time without centering the data with respect to their mean values. Try to explain the main differences which can be noticed in each figure.

A real application on ECG signal

Question 6

Load the signal in `data.mat`, which you can find in `AssignmentPCA.rar`. The signal `data` is a 12-lead (12 channels) ECG recording sampled at 1KHz (samples on rows and leads on columns). The signal has been acquired from a patient affected by atrial fibrillation.

- Consider `data`. Perform PCA on `data` by Singular Value Decomposition (look at the documentation of the Matlab function *svd*, and consider using `[U, S, V] = svd(data,'econ');`), to find out `A` and `Z` (based on the SVD model presented in class).
- How could you obtain an approximation of the cardiac beat (the peaks in the ECG signals) by exploiting the results at the previous point?

  Remember how PCA can be used to reduce the dimensionality of a dataset by compressing the information in few components describing most of the variance in the original dataset.

- It is known that atrial fibrillation is characterized by a dominant frequency in the range $[4, 10]$ Hz. With this knowledge, propose a way both to estimate the value of the dominant frequency of the atrial fibrillation in the ECG recordings under analysis, and to reconstruct the original ECG recordings only by using the PCs with a dominant peak in the range $[4, 10]$ Hz