PCA as a tool for preprocessing and Kernel PCA

This problem sheet explores applications and extensions of PCA. The first two exercises deal with PCA as a method for preprocessing and the third one illustrates how to find nonlinear structure via kernel PCA.

4.1 Preprocessing (2 points)

(a) Load the dataset `pca2.csv`. Compute the Principal Components PC1 and PC2 and plot the data in the coordinate system PC1 vs. PC2 – What do you observe?

(b) Remove Observations 17 and 157 and redo the first two steps. What is the difference?

4.2 Whitening (3 points)

(a) Load the dataset `pca4.csv` and check for outliers in the individual variables.

(b) Do PCA on a reasonable subset of this data. Use a scree plot to determine how many PCs represent the data well.

(c) "Whiten" the data, i.e. create a set of 4 *uncorrelated* variables with *mean 0* and *standard deviation equal to 1*. This can be done e.g. using the transformation

$$Z = \tilde{X} E D^{-1/2}$$

The new variables z_i form the columns of Z, E is a matrix containing the normalized eigenvectors of the covariance matrix Σ of the centered data $\tilde{X}$ and D is a diagonal matrix containing the corresponding eigenvalues.

(d) Make 3 heat plots of the (i) 4x4 covariance matrix Σ, (ii) the covariance matrix of the data projected onto PC1-PC4, and (iii) of the whitened variables.

4.3 Kernel PCA: Toy Data (5 points)

(a) Create a toy dataset of 2-dimensional data points $\boldsymbol{x}^{(\alpha)} = (x_1^{(\alpha)}, x_2^{(\alpha)})$, $\alpha = 1, \ldots, 90$. The points represent iid samples of 30 points from 3 different distributions with uncorrelated, normally distributed (sd=0.1) coordinate values differing only in their mean value. The first sample ($\alpha = 1, \ldots, 30$) should be centered on $\langle \boldsymbol{x}^{(\alpha)} \rangle_1 = (-0.5, -0.2)$, the second ($\alpha = 31, \ldots, 60$) on $\langle \boldsymbol{x}^{(\alpha)} \rangle_2 = (0, 0.6)$, and the third ($\alpha = 61, \ldots, 90$) on $\langle \boldsymbol{x}^{(\alpha)} \rangle_3 = (0.5, 0)$.

(b) Apply a Kernel PCA using the RBF kernel (see below) with a suitable parameter value for the width σ of the kernel and calculate the coefficients for the representation of the eigenvectors (PCs) in the space spanned by the transformed data points.

(c) Visualize the first 8 PCs in the 2-dimensional input space in the following way: Use equally spaced "test" gridpoints (in a rectangle $[a, b] \times [c, d] \subset \mathbb{R}^2$ containing all sampled data points) and determine their PC values by projecting onto the first 8 eigenvectors in feature space. For example, plot contour lines indicating points that yield the same projection onto the respective PC. You may also use a heat map or pseudo color plot (e.g. `pcolor`) to distinguish the different regions. Plot the 90 data points in the same plot (e.g, using small gray circles). How do you interpret the results?

Remark 1: Ensure to center the kernel matrix of the data points before projecting them onto the PCs. Furthermore, since most test points will differ from the sampled data points, you have to ensure that also when calculating the PC projections of these points centered feature vectors are considered.

Remark 2:

RBF kernel: $$k(x^{(\alpha)}, x^{(\beta)}) = \exp\left(-\frac{\|x^{(\alpha)} - x^{(\beta)}\|^2}{2\sigma^2}\right)$$

Total points: 10