Machine Intelligence II **Exercise Sheet 5**
SS 10, Prof. Obermayer due to 27.5.2010

Principal Component Analysis

In this exercise we will calculate the principal components of some toy and some real data in the original (input) space, as well as in high dimensional feature space by application of the *kernel trick*.

Additional Material: Download the data file archive `datafiles.zip` from the ISIS platform.

5.1 Toy Data

(3 points)

- Load the toy dataset `toypca/pca_data.dat` and plot the original (centered) data points.
- Calculate the principal components (PCs) and plot the data points using the PCs as coordinate system.
- Plot the reconstruction of the data in the original coordinate system. Make two additional plots, using only the first and the second PC for the reconstruction, respectively.

5.2 Image Data

(3 points)

The directory `imgpca` of the zip archive `datafiles.zip` contains two categories of training images: *nature* (prefix `n`) and *buildings* (prefix `b`). For each of both categories do the following steps:

- Sample image patches of size 8×8 pixels. To use them as training data rearrange them to a 8^2-dimensional vector. A several thousand samples are needed, sampled from all the images in the respective category.
- Calculate the principal components (PCs) of the image patches.
- Show the first 20 (or so) PCs as image patches by rearranging them into a 8×8 matrix.

  Which differences between both categories can be observed?

5.3 Kernel PCA

(4 points)

- Create a toy dataset consisting of 3×30 2-dimensional data points normally distributed around the means $(-0.5, -0.2)^T$, $(0, 0.6)^T$, $(0.5, 0)^T$ with standard deviation 0.1.
- Perform KPCA using the RBF kernel

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

Josef Ladenbauer: `jl@cs.tu-berlin.de`

that is, calculate the coefficients for the representation of the eigenvectors (principal components) in feature space (spanned by the transformed data points).

- Visualize the first 8 principal components in input space in the following way: Plot contour lines showing constant projections onto the principal components ($\hat{=}$ constant PC values). You may additionally use a pseudocolor plot (`pcolor` in Matlab) to distinguish the different regions.
  For the visualization use equally spaced "test" gridpoints and extract the features (PC values) of those points by projecting onto the first 8 eigenvectors in feature space. Interpret the results.

Total points: 10