HUMBOLDT-UNIVERSITÄT ZU BERLIN

BERNSTEIN CENTRE FOR COMPUTATIONAL NEUROSCIENCE

Bernstein Center for Computational Neuroscience
Humboldt-Universität zu Berlin
Philippstr. 13 House 6

Phone: 030/2093-9110
Fax: 030/2093-6771
webpage: http://www.bccn-berlin.de/

Models of Neural Systems I, WS 2008/09
Computer Practical 4

Unsupervised Learning

In the exercise we will study *unsupervised* learning algorithms. The purpose of unsupervised learning is to discover significant patterns or features in the input data *without the teacher*. The process of such self-organization is fundamental to the organization of the brain and therefore the modelling of network structures used for unsupervised algorithms tends to follow neurobiology to greater extent than for supervised learning. We will focus here on a wide-used feature extraction method called *principal component analysis* (PCA). We will first study its mathematical formulation and then its possible neural implementations. Finally, we will use it for the analysis of real neurophysiological data.

Exercises

1. **Principal Component Analysis**

   (a) Bivariate dataset: We want to generate a 2D dataset which consists of two clusters each containing 100 observations of two random variables from the following distributions:

   - 1st cluster – both variables are normally distributed with a mean of zero and a variance of one,
   - 2nd cluster – both variables are normally distributed with a mean of ten and a variance of one.

   Concatenate both clusters of data points into one array $\mathbf{Z}$.

   (b) Subtract the mean over all observations in each dimension of array $\mathbf{Z}$ and calculate its covariance matrix:

   $$\mathrm{Cov}(\mathbf{Z}) = \mathbf{Z}\mathbf{Z}^T/(n-1),$$

   where n is the number of observations.
   Hint: Make sure you don't mix up rows and columns (observations and variables).

(c) Now determine the eigenvectors and eigenvalues of the covariance matrix. Draw the eigenvectors (principal components) on the data plot. How much of the variance can be explained by each of the principal components?

(d) Project the data on principal components and plot the results on XY plane. How are the data transformed? Which dimension contains the most information about the data structure?

(e) Perform PCA with `pylab.prepca` function. Compare the results to your implementation.

2. **Oja's rule**

(a) Implement unsupervised Hebbian learning of a linear unit with weight normalization. The update rule (Oja's rule) is given by:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \eta y(n)[\mathbf{x}(n) - y(n)\mathbf{w}(n)]$$

Note: Here the activation function is an identity function.

(b) Train the neuron on the bivariate dataset from Exercise 1. Show that the neuron learns to extract the first principal component (with the largest variance).

(c) (Optional) Implement neural network to extract all of the principal components (using Sanger's rule). Test the results. What can be a function of such a network in the brain?

Contact

Robert Schmidt (ITB, r. 2316) Phone: 2093-8926 Email: r.schmidt@biologie.hu-berlin.de

Bartosz Telenczuk (ITB, r. 1309) Phone: 2093-8838 Email: b.telenczuk@biologie.hu-berlin.de