COURSE 02457

Non-Linear Signal Processing: Exercise 8

This exercise is based on C. M. Bishop: *Neural Networks for Pattern Recognition* sections 2.6, 5.9.4.

Print and comment on the figures produced by the software as outlined below at the **Checkpoints**.

Density Estimation

We observe a stochastic multi-channel signal $\mathbf{x}$ and our aim is model the density $p(\mathbf{x}) \sim p(\mathbf{x}|\mathbf{w})$ where the family $p(\mathbf{x}|\mathbf{w})$ is a given parametric density. The training set is $\chi = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, ..., \mathbf{x}_N\}$ and the likelihood function is given by $p(\chi|\mathbf{w}) = \prod_{n=1}^{N} p(\mathbf{x}_n|\mathbf{w})$ The costfunction which is minimized by the optimal parameters is,

$$E(\mathbf{w}) = \sum_{n=1}^{N} -\log p(\mathbf{x}_n|\mathbf{w}).$$

The *test error* of a density model can be estimated by evaluating the costfunction on a test set.

The gaussian mixture model family is defined

$$p(\mathbf{x}|\mathbf{w}) = \sum_{j=1}^{M} p(\mathbf{x}|j, \mathbf{w}_j)P(j).$$

Where each component density is a normal distribution. Here we will invoke a family of "isotropic" Gaussians, i.e., Gaussians with covariance matrices that are scaled unit matrices,

$$p(\mathbf{x}|\boldsymbol{\mu}_j, \sigma_j^2) = \frac{1}{(2\pi\sigma_j^2)^{d/2}} \exp\left(-\frac{(\mathbf{x}-\boldsymbol{\mu}_j)^2}{2\sigma_j^2}\right).$$

The Expectation-Maximization algorithm is a general scheme for maximum likelihood estimation and for the above mode it leads to the iterative procedure (Bishop page 67),

$$\widehat{\boldsymbol{\mu}_j^{\text{new}}} = \frac{\sum_{n=1}^{N} P^{\text{old}}(j|\mathbf{x}_n)\mathbf{x}_n}{\sum_{n=1}^{N} P^{\text{old}}(j|\mathbf{x}_n)}$$

$$\widehat{(\sigma_j^{\text{new}})^2} = \frac{1}{d}\frac{\sum_{n=1}^{N} P^{\text{old}}(j|\mathbf{x}_n)(\mathbf{x}_n - \boldsymbol{\mu}_j^{\text{new}})^2}{\sum_{n=1}^{N} P^{\text{old}}(j|\mathbf{x}_n)}$$

and

$$P^{\text{new}}(j) = \frac{1}{N}\sum_{n=1}^{N} P^{\text{old}}(j|\mathbf{x}_n).$$

The rules for updating $\boldsymbol{\mu}_j, \sigma_j^2$ are very similar to the wellknown rules for computing mean and variance of a normal distribution, but here weighted by the probability $P^{\text{old}}(j|\mathbf{x}_n)$ that a given datapoint $\mathbf{x}_n$ belongs to the given mixture component.

The *K-means* clustering algorithm is an important simplification of these rules obtained by using each datapoint only once, namely for updating the gaussian component which it is most associated with and by letting all the component variances be equal (Bishop page 190). In the context of the K-means algorithm a component is often referred to as a *cluster*.

Checkpoint 8.1

Use the program `main8a.m` to perform K-means analysis on synthetic two-dimensional data with three clusters. The program creates two plots. The first shows the training points and the current position of the cluster centers as time progress. You can zoom to inspect the details of the convergence. The second figure shows the assignment of points to clusters at the end of the procedure. Is the final configuration sensitive to how the initial cluster centers are located?. You can change the distribution of the initial clusters by changing the parameter `initial-width` and you can change the number of clusters `K`. Sometimes you will see that a cluster never moves away from its initial position, why?.

Checkpoint 8.2

The program `main8b.m` uses the Expectation-Maximization algorithm to adapt a gaussian mixture, as descibed above. The program shows three plots. The first plot is the training (blue) and test (yellow) points, the second shows the temporal evolution of the variance parameters and the training and test errors. Create a flow chart of the program `main8b.m` and its functions. The test error is the mean negative log likelihood (the costfunction) estimated on the test set. Change the number of clusters $K = 2, 3, 4, 5, 6$ and inspect the temporal evolution and final value of the test error. Explain how the Gaussian mixture can over-fit.

Checkpoint 8.3

In `main8b.n` you can choose three different strategies for initializing the cluster centers and initial variances. Set $K = 5$ and run the program with the three different schemes. Describe the strategies and their results. You will see a problem with the Gaussian mixture and the EM algorithm for `method 3`, where one component converge towards a very small variance and is centered on a specific data point: explain why this is a serious overfit.

DTU, 1999 Lars Kai Hansen