Parameter Selection using Cross-Validation

WS 2014/2015

At the next lecture, you will learn several methods for estimation of classifiers' errors. *Cross-validation* is one such method which is especially useful when we do not have enough data to afford the luxury of splitting them to a big independent train set and test set. In this tutorial you will be using stratified cross-validation for selecting optimal parameters of a classification model.

Self-study...

1. Study the materials for the lecture on empirical assessment of learning methods. After that you should be able to understand the principles of cross-validation. **Briefly:** We have a training set[1] $\mathcal{T}$ containing examples which, in our case, belong to two classes (*positive* and *negative*). We split $\mathcal{T}$ to k disjoint sets with approximately the same cardinalities $\mathcal{T}_i$. All these sets should contain approximately the same proportion of positive and negative examples (i.e. if the training set $\mathcal{T}$ contained 90% positive examples and 10% negative examples then we would expect each $\mathcal{T}_i$ to contain also approximately 90% positive examples and 10% negative examples). Next, We repeat k-times the following procedure (for $i = 1, \ldots, k$). We take the i-th set (i.e. $\mathcal{T}_i$) and call it $Test_i$ and the union of the remaining sets $Train_i = \bigcup_{j=1,\ldots,k, j \neq i} \mathcal{T}_j$. Then we train a classifier on $Train_i$ and use it to predict class-labels of the examples from $Test_i$. Comparing the obtained predictions with the true class-labels, we obtain the error e_i. The final estimate of prediction error is then given as an average of these e_i's.

Data and Classifiers...

2. Training data will be generated from a mixture of k univariate Gaussians with the same σ (you will use the function `generate_examples`). The positive examples are generated from a mixture with means of the components being 2, 4, ..., $2q$. Similarly, the negative examples are generated from a

[1] More precisely: in this tutorial, when we speak of *sets*, we mean *multi-sets*.

mixture with means of the components being 1, 3, ..., $2q-1$ (where q is a parameter).

3. We will use Bayesion decision theory for which we need to know the probabilistic model for positive and negative examples. In our case, these models will be Gaussian mixture models and we will estimate their parameters using the EM algorithm. As you know from the first part of this course, it is necessary to specify the number of components when using the EM-algorithm for estimation of parameters of a Gaussian mixture. We will try to select this parameter using cross-validation. That is: we select such a number of components which minimizes the error estimated by cross-validation.

Implementation...

4. **Go through the source codes for this tutorial and implement the missing parts of function `cross_validation`.**

Testing...

5. Test your implementation on generated data (everything is prepared for you in `tutorial_13.m` file).