LELEC2870: Exercises session 5
Feature Selection and Model Selection

1 Objectives

As you may have noticed during the previous sessions, two practices are commonly used when solving a machine learning problem. The first one, introduced in Session 1, is feature selection. Indeed, some models are very sensitive to the presence of irrelevant features which affect, for instance, distance measures. The second practice, introduced in Session 3 and 4, is model selection. Indeed, most prediction models have one or several meta-parameters whose values have to be carefully chosen in order to get good model performances. Both the feature selection and the model selection procedures will be investigated more in depth in this session.

2 Creating a dataset for our experiments

For this session, you have to build an artificial dataset containing 1000 observations. For each observation $x_i = (x_{i,1}, \ldots, x_{i,6})$, the values of the 6 features are randomly chosen in the interval $[0, 1]$ (you can use *numpy.random.random_sample* for this). The target is then computed as:

$$f(x_i) = 2\sin(2 * x_{i,1})\, x_{i,2} + 4\,(x_{i,3} - .5)^2 + x_{i,4} + \epsilon_i \tag{1}$$

where ϵ_i is a noise component following a normal distribution $\mathcal{N}(0, 0.01)$ (you can use *numpy.random.normal* for this). Before going ahead, take a look at Equation 1. What can you say about the six features of our problem? Are they all equally useful?

3 Feature Selection

Feature selection can be performed with a criterion that quantifies the pertinence of features for predicting the target. We will investigate the two following criteria: the correlation coefficient (using *numpy.corrcoef*) and the mutual information (using *sklearn.feature_selection.mutual_info_regression*). A simple feature selection strategy consists in selecting the features achieving a sufficiently high score for a given criterion. Implement this strategy and apply it on your dataset.

Analyse the results. Did you expect these results ? Are they coherent with Equation (1)? Are both criteria appropriate for our task?

Before moving to the next step, build a reduced training set only containing the features you have selected. Keep also a copy of the complete training set in order to make comparisons and to assess the interest of feature selection.

4 Model selection

RBFNs that you have implemented in session 3 have two meta-parameters (the number of centers and the smoothing factor). These meta-parameters cannot be optimized directly using a training set since their role is to control the model complexity and to prevent under/overfitting. In the remaining of this session, you will implement a simple validation procedure which allows to select good values of the meta-parameters for a specific training set. To train RBFNs, you can either use your own code or the one we provide on Moodle (*rbfn.py*).

Divide the dataset you just created in a training set (70%) and a validation set (30%). Then, build a grid (of reasonable size) of the values you will test for the meta-parameters. For each pair of values (number of centers, smoothing factor), train a RBFN using the training set and measure the error made on the validation set. Use the results to select the appropriate number of centers and the smoothing factor for your problem. Eventually, train a RBFN on the whole dataset using the chosen meta-parameters.

Build a test set containing 10000 samples. Measure the error made on these data using your model. Repeat the whole procedure detailed in this section using all the features. How do the results compare to the case where feature selection has been made ?