CS 504 – Programming Languages for Data Analysis

Assignment 4: Cross-validation in Machine Learning

I. Principle

Cross-validation is in machine learning, a method of estimating the reliability of a model based on a sampling technique. Suppose we have a statistical model with one or more unknown parameters, and a training dataset on which to learn (or "train") the model. The training process optimizes the parameters of the model so that it matches the training data as closely as possible. If we then take an independent validation sample, supposedly from the same population as the training sample, it will generally turn out that the model does not model the validation data as well as the training data: we talk about *overfitting*. However, an independent validation sample is not always available. In addition, from one validation sample to another, the model validation performance may vary. Cross-validation makes it possible to derive several validations sets from the same database and thus to obtain a more robust estimate, with bias and variance, of the validation performance of the model.

There are many variants of validation, but we can first distinguish:

- Non-cross validation, "testset validation" or "holdout method": we divide the sample of size n into two sub-samples, the first known as learning (commonly greater than 60% of the sample) and the second says validation or test. The model is built on the training sample and validated on the test sample with a performance score of our choice.
- Cross validation at k blocks, "*k-fold cross-validation*": we divide the original sample into k samples (or "blocks"), then we select one of the k samples as the validation set while the $k-1$ other samples constitute the training set. After learning, a validation performance can be calculated. Then the operation is repeated by selecting another validation sample from among the predefined blocks. At the end of the procedure, we thus obtain k performance scores, one per block. The mean and standard deviation of k performance scores can be calculated to estimate the bias and variance of validation performance. Table 1 illustrates an example of Cross-validation at $k = 3$.

- The cross validation of one against all, "leave-one-out cross-validation" (LOOCV): this is a special case of the cross validation at k blocks where $k = n$. That is to say that at each learning-validation iteration, the learning is done on $n - 1$ observations and the validation on the only remaining observation.

Table 1: Data distribution for cross validation at k = 3 blocks

k	**Block 1**	**Block 2**	**Block 3**
1	Validation	Learning	Learning
2	Learning	Validation	Learning
3	Learning	Learning	Validation

Your task in this assignment is to a pptx presentation about the tools that are offered by Julia, Python, R, and Octave to perform cross-validation. Your presentation must contain illustration with examples.

In your conclusion you must answer to the following questions:

1. Which one of the programming languages that you studied does support best the three variants of cross-validation? Justify your answer.
2. Do all the programming languages that you studied support the three variants of cross-validation?

II. Submission

You have to submit a pptx or ppt file of your presentation.

Good luck :)