PERFORMANCE EVALUATION II
TRAIN AND TEST

Dr. Lablanche Pierre-Yves
African Institute for Mathematical Sciences

EVALUATING AN ALGORITHM

Two phases : **training phase** and **testing phase**

Example : Training a Classifier

We have a data set of *N* samples.
We trained a classifier on the whole data set.

Does it make sense to evaluate it on the same data set ?

EVALUATING AN ALGORITHM

YES... :

It is important to know the Classifier is well trained.

... and NO :

Every trained model depends on the input data (cf. overfitting issue), we don't know yet how well the classifier respond to new samples.

Solution : Split the data set in two subsets
one for training and one for testing

TRAIN - VALIDATION - TEST

Most common practice : divide the data set in train set and test set … but sometimes also include a validation set.

- **Train set** : Subset of the initial data set used to train the algorithm.
- **Validation set** (optional): Subset of the original data set used during training phase to fine-tune the algorithm.
- **Test set** : Subset of the initial data set used to test the algorithm.

TRAIN - VALIDATION - TEST

" What is a good ratio between the train and the test sets ? "

It depends on the dataset and what you evaluate.

No formula or theoretical framework to choose the best split.
Recommended to keep the train set bigger than the test set.

ISSUE !

Any score evaluated using a single train/test split is still biased and depends on the split.

Solution n°1 : repeat the whole process

- split the dataset in train and test sets
- compute the score
- average the results

Solution n°2 : use the **k-fold cross-validation** technique

K-FOLD CROSS-VALIDATION

k-fold cross-validation technique in a nutshell :

- Split the dataset in k subsets

- Train the algorithm on k-1 subsets

- Test/compute the score on the remaining subset

- Average results

K-FOLD CROSS-VALIDATION

Choice for k (aka the number of *folds*) can be constrained by :

- The dataset size
- The dataset nature
- The computing time available
- …

LEAVE-ONE-OUT

Extreme case of k-fold, set $k=N$ (N number of samples the dataset),
Hence :

- Train the algorithm on $N-1$ samples

- Test the algorithm on a single sample

- Average results

CODING TIME

- Use two classifiers of your choice on the iris dataset.
- Use the different metrics on the trained classifiers;
- Do results make sense ?