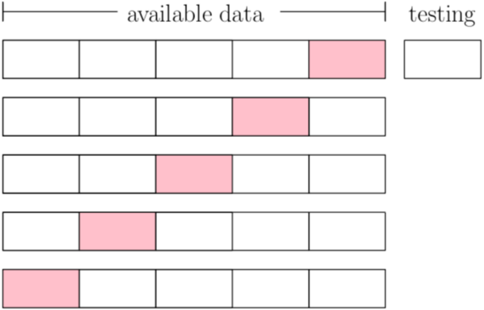
Machine learning I, supervised learning: scoring



Scoring

Many possibilities are available to evaluate the quality of an estimator.

Regression

- Input space $\mathcal{X}$
- Output space $\mathcal{Y} = \mathbb{R}$.

Until now we used the squared error or the mean squared error (MSE) :

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_{pred,i} - y_{label,i})^2 \tag{1}$$

Without normalisation, it is also called **residual sum of squares** (RSS)

$$RSS = \sum_{i=1}^{n}(y_{pred,i} - y_{label,i})^2 \tag{2}$$

Coefficient of determination

Also called $R2$. $R2 \leq 1$. We introduce the Total sum of squares (TSS)

$$TSS = \sum_{i=1}^{n} (y_{label,i} - \bar{y})^2 \tag{3}$$

where $\bar{y}$ is the mean of the observed data

$$\bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_{label,i} \tag{4}$$

Then

$$R2 = 1 - \frac{RSS}{TSS} \tag{5}$$

Coefficient of determination

$$TSS = \sum_{i=1}^{n} (y_{label,i} - \bar{y})^2 \tag{6}$$

$$RSS = \sum_{i=1}^{n} (y_{pred,i} - y_{label,i})^2 \tag{7}$$

Finally we define the Explained sum of squares ESS :

$$ESS = \sum_{i=1}^{n} (y_{pred,i} - \bar{y})^2 \tag{8}$$

Then if the predicitons are linear, then

$$TSS = ESS + RSS \tag{9}$$

Scikit metrics

https://scikit-learn.org/stable/modules/model_evaluation.html

Binary classification

We now review some metrics for binary classification problems.

Accuracy

Most simple scoring : accuracy.

$$\frac{\text{number of correct predictions}}{\text{number of samples}} \tag{10}$$

Precision and recall

Precision : "Quand tu dis que c'est positif, c'est positif".

$$\frac{\text{TP}}{\text{TP} + \text{FP}} \tag{11}$$

Recall / sensitivity (rappel) : "Quand c'est positif, tu dis que c'est positif".

$$\frac{\text{TP}}{\text{TP} + \text{FN}} \tag{12}$$

See also : specificity and 1-specificity.

F score

Also called $F1$. It quantifies the tradeoff between precision and recall.

$$F1 = 2 \times \frac{\text{sensitivity} \times \text{precision}}{\text{sensitivity} + \text{precision}} \tag{13}$$

Exercice 1 : What are the extremal values of F ?

`https://en.wikipedia.org/wiki/F-score`

Translation to french

In french, accuracy is sometimes translated as "précision", and precision as "exactitute".

Binary classification

Standard classifiers such as logistic regression and support vector machines are **binary**.
However, sometimes $|\mathcal{Y}| > 2$.

Number of classifiers for multi-class classification

We assume $|\mathcal{Y}| = p$.

Exercice 2 : How many classifiers need to be built :

- with the one-vs-rest scheme ?
- with the one-vs-one scheme ?

`https://machinelearningmastery.com/one-vs-rest-and-one-vs-one-for-multi-class-classification/`

Number of classifiers

- one-vs-rest is the standard approach.
- one-vs-one need to build a number of classifiers that is quadratic in p, $(\mathcal{O}(p^2))$.
- However one-vs-one might still be useful since less samples are used by each binary classifiers, since we only need the samples from the two selected classes. If the complexity of the classifier scales badly with the umber of samples n (like for kernel methods), one-vs-one might be faster.

Scikit

Scikit has a builtin implementation of both schemes.

`https://scikit-learn.org/stable/modules/generated/sklearn.multiclass.OneVsRestClassifier.html`

Softmax

It is also possible to directly learn p real outputs and convert the vector of outputs to a probability by normalizing (it is what is done with softmax regression in neural networks).
`https://fr.wikipedia.org/wiki/Fonction_softmax`

Confusion matrix

https://en.wikipedia.org/wiki/Confusion_matrix
https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion_matrix.html

Classification report

It is also possible to define precision, recall (and thus F1) for each class. In scikit, classification report prints these quantities for each class

`https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification_report.html`

Multi-class vs multi-label

Don't mix multi-class problems and multi-label problems.

- multi-class : several output classes are possible
- mutli-label : we have to make several predictions for each input

A problem can be both multi-class and multi-label.
https://scikit-learn.org/stable/modules/generated/sklearn.multioutput.MultiOutputClassifier.html
Multioutput regression is also possible.
https://scikit-learn.org/stable/modules/generated/sklearn.multioutput.MultiOutputRegressor.html

Hyperparameters

All learning algorithms have hyperparameters. Examples :

- regularization parameter
- learning rate schedule
- kernel widths
- tree depth for cart (trees)
- number of trees for random forest

Hyperparameter tuning

Sometimes, we have theoretical results that guarantee that a hyperparameter value is a good choice (e.g. the learning rates for GD, SGD, SAG).

However :

- often, these parameters values depend on constants that are problem-dependent and sometimes not available :
  - variance σ^2
  - smoothness constant L
- we may not have a theoretical result at all (true for some aspects of deep learning)
- some values of the hyperparameter might work **better** than the theoretical value.

Hyperparameter tuning

Conclusion : it is most often necessary to experiment and test in order to find relevant hyperparameters.

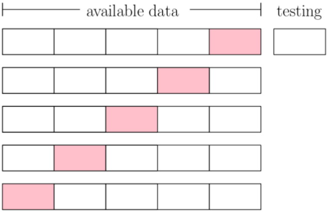
Common practice is to test several hyperparameters, which means training several models. The dataset can be split into 3 parts :

- **train set** : used to optimize each model
- **validation set** : used to compute a validation error of each model (error on this dataset). The model with lowest validation error can be chosen.
- **test set** : used to test the final model. We can not use it to choose the hyperparameters, otherwise the hyperparameters could "overfit this set".
- sometimes people use the opposite convention between "test" and "validation".

Problem : there might be a high variability in the validation procedure. The found hyperparameters might depend a lot on the initial choice of the validation set.

Cross-validation

Cross-validation is another method that allows the use of more training data. The train set is split in k folds (often 5 of 10), and k validation errors are computed (one for each fold). The model with the lowest average validation error is chosen, **and then** trained on the whole train set.



Cross-validation

Due the the higher number of computations, cross-validation might be slower than standard train/validation/test split.

Scikit

Grid search is a method for testing hyper parameters (exhaustive search among a fiven list of values).
Grid search and cross validation are builtin in scikit.
`https://scikit-learn.org/stable/modules/cross_validation.html`
`https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html`

Learning curves

https://scikit-learn.org/stable/auto_examples/model_selection/plot_learning_curve.html

Many model selection methods exist :

https://scikit-learn.org/stable/model_selection.html