

TenFoldCrossValidation

10-Fold Cross Validation

10-fold cross validation (abbreviated "10FCV") is a system for testing trained classifiers. We use it in [SpamAssassin](#) development and QA.

The comp.ai.neural-nets FAQ covers it well, in <http://www.faqs.org/faqs/ai-faq/neural-nets/part3/section-12.html> :

Cross-validation

+++++

In k-fold cross-validation, you divide the data into k subsets of (approximately) equal size. You train the net k times, each time leaving out one of the subsets from training, but using only the omitted subset to compute whatever error criterion interests you. If k equals the sample size, this is called "leave-one-out" cross-validation. "Leave-v-out" is a more elaborate and expensive version of cross-validation that involves leaving out all possible subsets of v cases.

In other words, take a testing corpus, divided into ham and spam; each message has previously been hand-verified as being of the correct type (e.g. ham if it's in the ham corpus, spam if in the other one). Divide each corpus into k folds. (In [SpamAssassin](#), we generally use k=10 – which is what pretty much everyone else does anyway, it just seems to work well 😊). Then run these 10 tests:

```
Train classifier on folds: 2 3 4 5 6 7 8 9 10; Test against fold: 1
Train classifier on folds: 1 3 4 5 6 7 8 9 10; Test against fold: 2
Train classifier on folds: 1 2 4 5 6 7 8 9 10; Test against fold: 3
Train classifier on folds: 1 2 3 5 6 7 8 9 10; Test against fold: 4
Train classifier on folds: 1 2 3 4 6 7 8 9 10; Test against fold: 5
Train classifier on folds: 1 2 3 4 5 7 8 9 10; Test against fold: 6
Train classifier on folds: 1 2 3 4 5 6 8 9 10; Test against fold: 7
Train classifier on folds: 1 2 3 4 5 6 7 9 10; Test against fold: 8
Train classifier on folds: 1 2 3 4 5 6 7 8 10; Test against fold: 9
Train classifier on folds: 1 2 3 4 5 6 7 8 9; Test against fold: 10
```

We use 10FCV to test:

- new tweaks to the "Bayesian" learning classifier (the BAYES_* rules)
- new tweaks to the rescoring system (which is also a learning classifier, just at a higher level).

Traditionally, k-fold cross-validation uses a "train on k-1 folds, test on 1 fold"; we use that for testing our rescoring system. However, for the BAYES rules, we use "train on 1 fold, test on k-1 folds", as otherwise it can be hard to get a meaningful number of false positives and false negatives to be able to distinguish improvements in accuracy, because that classifier is very accurate when sufficiently trained.

So, for example,

See [RescoreTenFcv](#) for a log of a sample 10-fold CV run against two [SpamAssassin](#) rescoring systems (the GA and the perceptron).