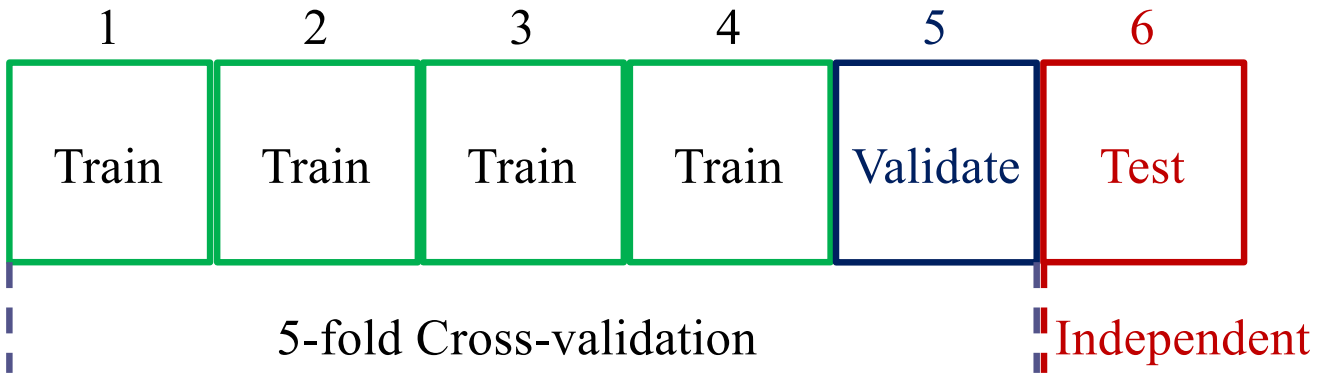
Cross-Validation

- Summary
  - *K*-fold cross-validation: Training set to fit the model, Validation set to validate the model
  - Independent test set
  - Commonly 5-fold/10-fold cross-validation



Cross-Validation

- Motivation
  - If we have enough data, we would set aside a validation set and use it to assess the performance of our prediction model.
  - *K*-fold cross-validation uses part of the available data to fit the model, and a different part to test it.

1	2	3	4	5
Train	Train	Validation	Train	Train

Cross-Validation

- Details:
  - The cross-validation estimate of prediction error:

$$\mathrm{CV}(\hat{f}) = \frac{1}{N}\sum_{i=1}^{N} L(y_i, \hat{f}^{-\kappa(i)}(x_i))$$

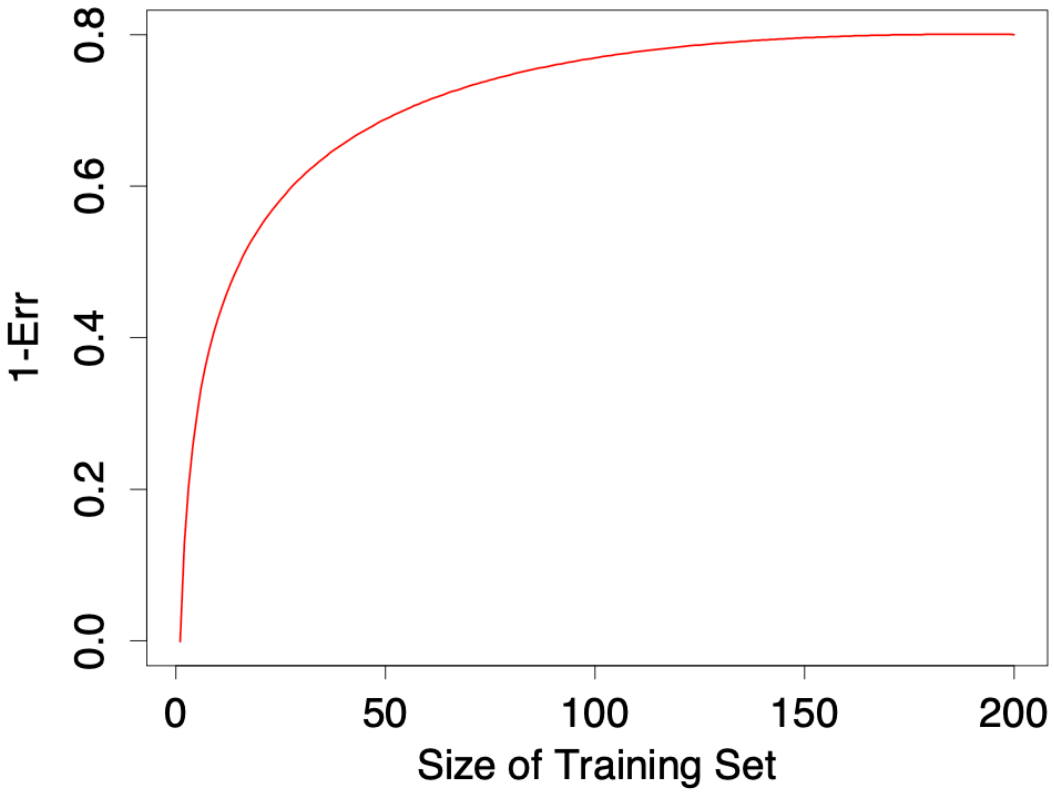
  - Given a set of models $f(x, \alpha)$ indexed by a tuning parameter α, denote by $f^{-k}(x, \alpha)$ the α-th model fit with the kth part of the data removed. Then for this set of models we define:

$$\mathrm{CV}(\hat{f}, \alpha) = \frac{1}{N}\sum_{i=1}^{N} L(y_i, \hat{f}^{-\kappa(i)}(x_i, \alpha)).$$

Cross-Validation

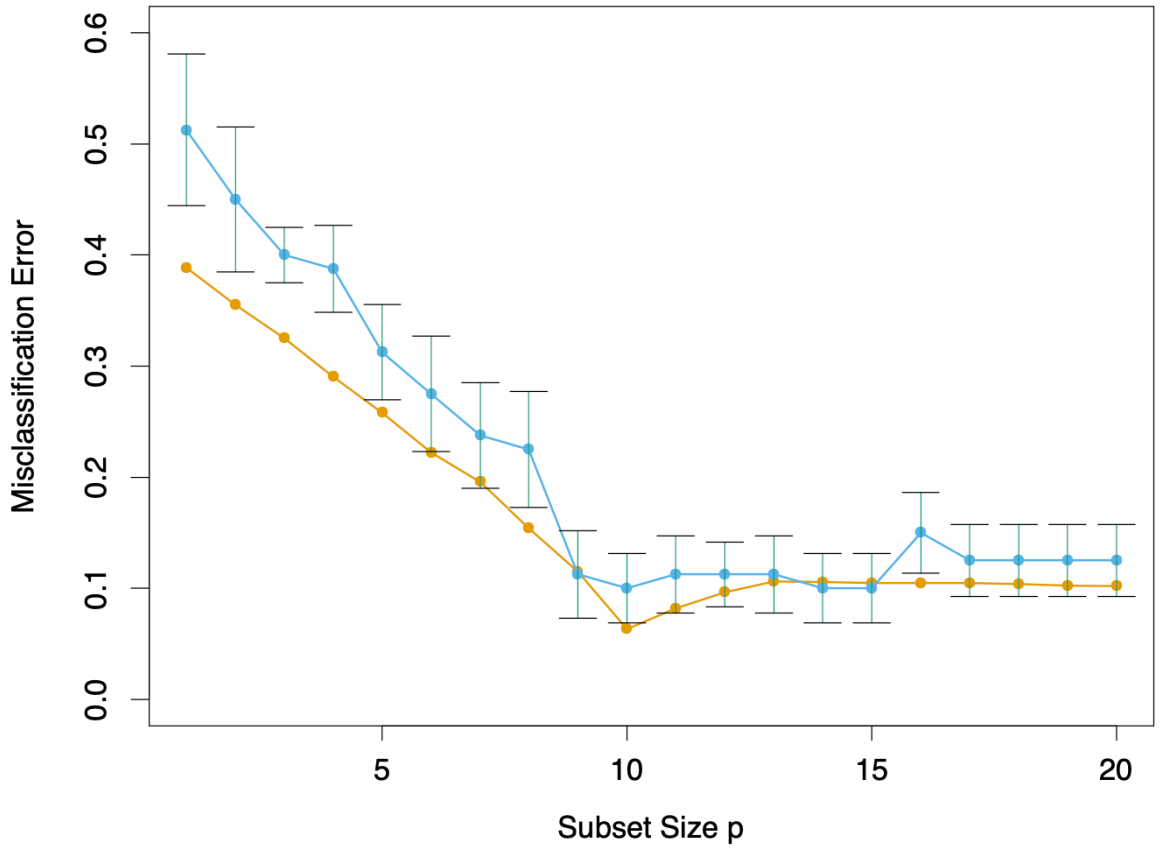
- What quantity *K*-fold cross-validation estimates?
  - It estimates the expected error or the conditional error.
  - *K*=5?
  - *K*=*N*?
- What value should we choose for K?
  - How to balance the variance and bias of Err estimates.
  - *K*=5?
  - *K*=*N*?

Cross-Validation



Hypothetical learning curve for a classifier on a given task: a plot of 1 − Err versus the size of the training set N. With a dataset of 200 observations, 5-fold cross-validation would use training sets of size 160, which would behave much like the full set. However, with a dataset of 50 observations five-fold cross-validation would use training sets of size 40, and this would result in a considerable overestimate of prediction error.

Cross-Validation



Prediction error (orange) and ten-fold cross-validation curve (blue) estimated from a single training set, from the scenario in two-class classification with linear model.

Generalized Cross-Validation

- In certain special problems, CV can be done quickly.
- For linear fitting under squared-error loss:
  - A linear fitting method is one for which we can write:

$$\hat{\mathbf{y}} = \mathbf{S}\mathbf{y}.$$

  - Now for many linear fitting methods,

$$\frac{1}{N}\sum_{i=1}^{N}[y_i - \hat{f}^{-i}(x_i)]^2 \quad = \quad \frac{1}{N}\sum_{i=1}^{N}\left[\frac{y_i - \hat{f}(x_i)}{1 - S_{ii}}\right]^2,$$

  - where S_{ii} is the i-th diagonal element of $\mathbf{S}$

Generalized Cross-Validation

- For linear fitting under squared-error loss:
  - The GCV approximation is:

$$\mathrm{GCV}(\hat{f}) = \frac{1}{N}\sum_{i=1}^{N}\left[\frac{y_i - \hat{f}(x_i)}{1 - \mathrm{trace}(\mathbf{S})/N}\right]^2$$

  - The quantity $\mathrm{trace}(\mathbf{S})$ is the effective number of parameters
  - GCV can have a computational advantage in some settings, where the trace of $\mathbf{S}$ can be computed more easily than the individual elements S_{ii}.

The Wrong and Right Way to Do Cross-validation

- Consider a classification problem with a large number of predictors, a typical strategy for analysis might be as follows:
  - Screen the predictors: find a subset of "good" predictors that show fairly strong (univariate) correlation with the class labels
  - Using just this subset of predictors, build a multivariate classifier.
  - Use cross-validation to estimate the unknown tuning parameters and to estimate the prediction error of the final model.

The Wrong and Right Way to Do Cross-validation

- Consider a scenario:
  - N = 50 samples in two equal-sized classes, and p = 5000 quantitative predictors (standard Gaussian) that are independent of the class labels. The true (test) error rate of any classifier is 50%.
  - We carried out the above recipe, choosing in step(1) the 100 predictors having highest correlation with the class labels,
  - Then using a 1-nearest neighbor classifier, based on just these 100 predictors, in step (2).
  - Over 50 simulations from this setting, the average CV error rate was 3%. This is far lower than the true error rate of 50%.

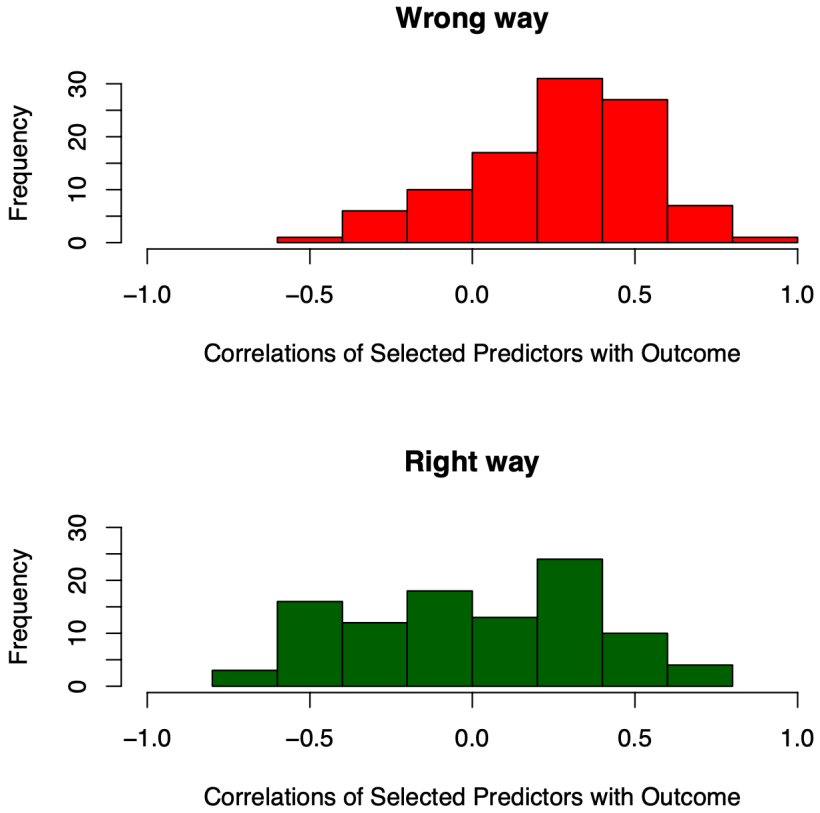
The Wrong and Right Way to Do Cross-validation

- What's the problem?
  - The predictors have an unfair advantage, as they were chosen in step (1) on the basis of all of the samples.
  - Leaving samples out after the variables have been selected does not correctly mimic the application of the classifier to a completely independent test set, since these predictors "have already seen" the left out samples.

The Wrong and Right Way to Do Cross-validation

- The correct way to carry out cross-validation in this example:
  - Divide the samples into *K* cross-validation folds (groups) at random.
  - For each fold $k = 1, 2, \ldots, K$
    - Find a subset of "good" predictors that show fairly strong (univariate) correlation with the class labels, using all of the samples except those in fold k.
    - Using just this subset of predictors, build a multivariate classifier, using all of the samples except those in fold k.
    - Use the classifier to predict the class labels for the samples in fold k.

The Wrong and Right Way to Do Cross-validation



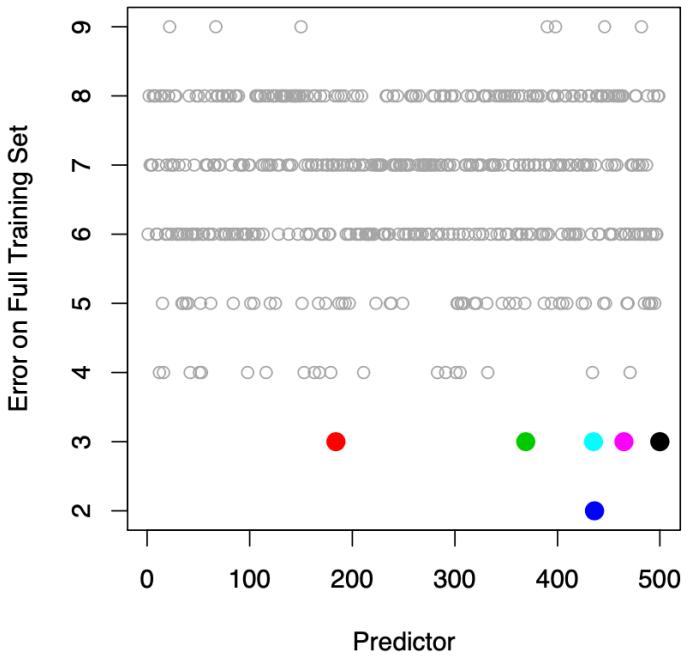
Cross-validation the wrong and right way: histograms shows the correlation of class labels, in 10 randomly chosen samples, with the 100 predictors chosen using the incorrect (upper red) and correct (lower green) versions of cross-validation.

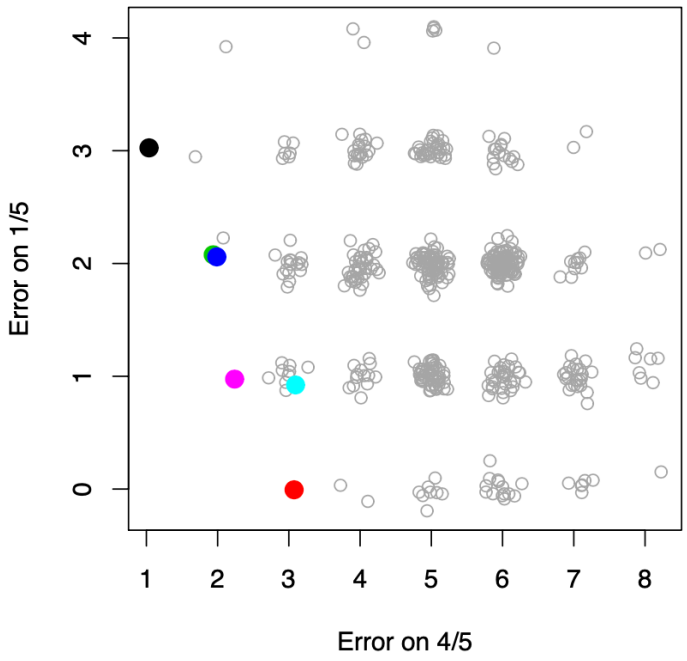
Does Cross-Validation Really Work?

- Consider a scenario with N = 20 samples in two equal-sized classes, and p = 500 quantitative predictors that are independent of the class labels.
- Consider a simple univariate classifier: a single split that minimizes the misclassification error (a “stump”) .
- A simple argument suggests that cross-validation will not work properly in this setting:
  - Fitting to the entire training set, we will find a predictor that splits the data very well. If we do 5-fold cross-validation, this same predictor should split any 4/5ths and 1/5th of the data well too, and hence its cross-validation error will be small (much less than 50%.) Thus CV does not give an accurate estimate of error.

Does Cross-Validation Really Work?

Left: The number of errors made by individual stump classifiers on the full training set.
Right: The errors made by individual stumps trained on a random split of the dataset into 4/5ths and tested on the remaining 1/5. The best performers are depicted by colored dots in each panel.





Does Cross-Validation Really Work?

Left: The effect of re-estimating the split point in each fold: the colored points correspond to the four samples in the 1/5th validation set. The split point derived from the full dataset classifies all four samples correctly, but when the split point is re-estimated on the 4/5ths data (as it should be), it commits two errors on the four validation samples. Right: The overall result of five-fold cross-validation applied to 50 simulated datasets. The average error rate is about 50%, as it should be.