PQHS 471
Lecture 8: Resampling Methods
Cross-Validation, Bootstrap

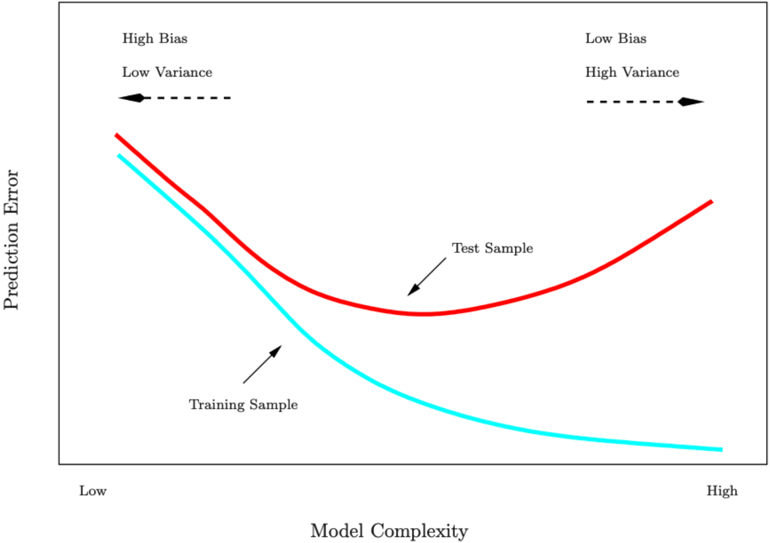
Cross-validation and the Bootstrap

- Two indispensable tools in modern machine learning and statistics.
- These methods refit a model of interest to samples formed from the training set, in order to obtain additional information about the fitted model.
- For example, they provide estimates of test-set prediction error, and the standard deviation and bias of our parameter estimates.

Training error vs test error

- Recall the distinction between the test error and the training error:
- The test error is the average error that results from using a statistical learning method to predict the response on a new observation, one that was not used in training the method.
- In contrast, the training error can be easily calculated by applying the statistical learning method to the observations used in its training.
- But the training error rate often is quite different from the test error rate, and in particular the former can dramatically underestimate the latter.

Training error vs test error



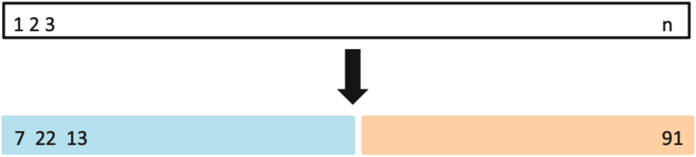
Prediction error estimate

- Best solution: a large designated test set. Often not available.
- Some methods make a mathematical adjustment to the training error rate. These include the Mallows's C_p, AIC and BIC.
- Here we instead consider a class of methods that estimate the test error by **holding out** a subset of the training observations from the fitting process, and then applying the statistical learning method to those held out observations.

Validation-set approach

- Here we randomly divide the available set of samples into two parts: a training set and a validation or hold-out set.
- The model is fit on the training set, and the fitted model is used to predict the responses for the observations in the validation set.
- The resulting validation-set error provides an estimate of the test error. This is typically assessed using MSE in the case of a quantitative response and misclassification rate in the case of a qualitative (discrete) response.

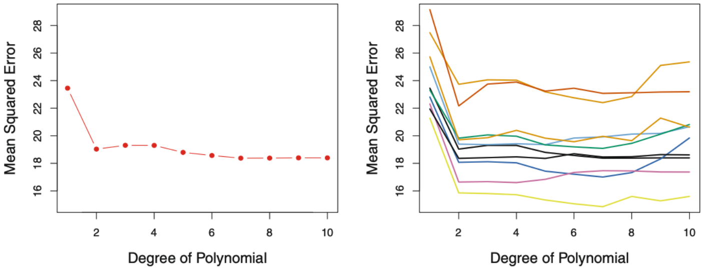
Validation-set approach



A random splitting into two halves: left part is training set, right part is validation set.

Example: automobile data

- Want to compare linear vs higher-order polynomial terms in a linear regression for predicting mpg.
- We randomly split the 392 observations into two sets, a training set containing 196 of the data points, and a validation set containing the remaining 196 observations.



Left panel shows single split; right panel shows 10 splits

Drawbacks of validation set approach

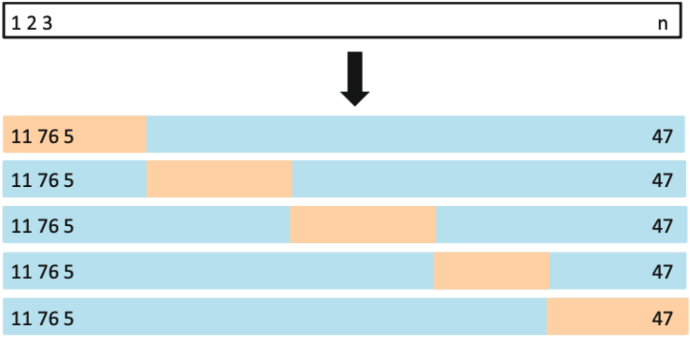
- Estimate of the test error can be highly variable, depending on precisely which observations are included in the training set and which observations are included in the validation set.
- Only a subset of the observations — those that are included in the training set rather than in the validation set — are used to fit the model.
- This suggests that the validation set error may tend to overestimate the test error for the model fit on the entire data set. Why?

K-fold Cross-validation

- Widely used approach for estimating test error.
- Estimates can be used to select best model, and to give an idea of the test error of the final chosen model.
- Idea is to randomly divide the data into K equal-sized parts. We leave out part k, fit the model to the other $K - 1$ parts (combined), and then obtain predictions for the left-out kth part.
- This is done in turn for each part $k = 1, 2, ...K$, and then the results are combined.

K-fold Cross-validation: example

Divide data into K roughly equal-sized parts ($K = 5$ here)



K-fold CV: details

- Let the K parts be $C_1, C_2, \ldots C_K$, where C_k denotes the indices of the observations in part k. There are n_k observations in part k: if N is a multiple of K, then $n_k = n/K$.
- Compute

$$CV_{(K)} = \sum_{k=1}^{K} \frac{n_k}{n} MSE_k$$

where $MSE_k = \sum_{i \in C_k} (y_i - \hat{y}_i)^2 / n_k$, and $\hat{y}_i$ is the fit for observation i, obtained from the data with part k removed.

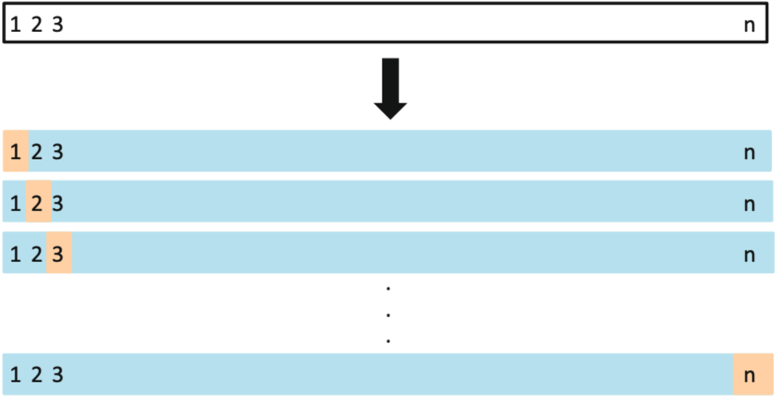
K-fold CV: details

- Let the K parts be $C_1, C_2, ...C_K$, where C_k denotes the indices of the observations in part k. There are n_k observations in part k: if N is a multiple of K, then $n_k = n/K$.
- Compute

$$CV_{(K)} = \sum_{k=1}^{K} \frac{n_k}{n} MSE_k$$

  where $MSE_k = \sum_{i \in C_k} (y_i - \hat{y}_i)^2 / n_k$, and $\hat{y}_i$ is the fit for observation i, obtained from the data with part k removed.
- Setting $K = n$ yields leave-one out cross-validation (LOOCV).

LOOCV



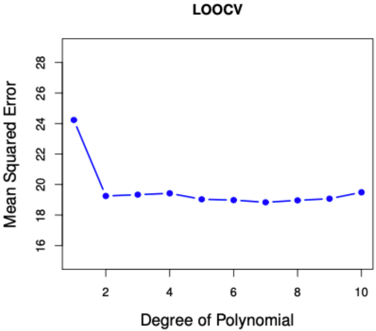
LOOCV: a nice special case

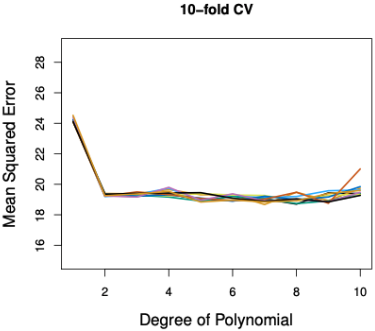
- With least-squares linear or polynomial regression, an amazing shortcut makes the cost of LOOCV the same as that of a single model fit! The following formula holds:

  $$CV_{(n)} = \frac{1}{n}\sum_{i=1}^{n}(\frac{y_i - \hat{y}_i}{1 - h_i})^2$$

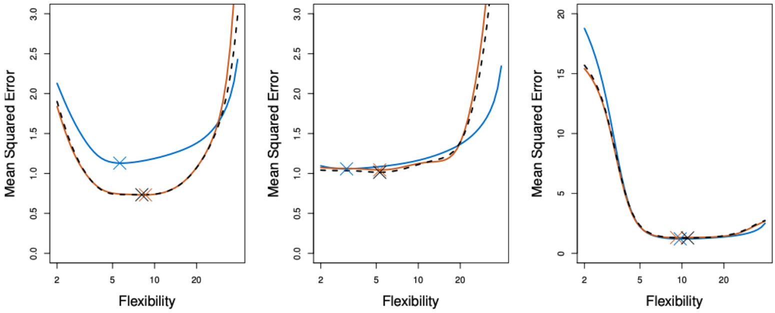
  where y_i is the ith fitted value from the original least squares fit, and h_i is the leverage (diagonal of the "hat" matrix) This is like the ordinary MSE, except the ith residual is divided by $1 - h_i$.
- LOOCV sometimes useful, but typically doesn't shake up the data enough. The estimates from each fold are highly correlated.

Auto data revisit





True and est. test MSE in simulation



Blue: true test error in simulation.
Black dashed: LOOCV. Orange: 10-fold CV.

Considerations in cross-validation

- Since each training set is only $(K-1)/K$ as big as the original training set, the estimates of prediction error will typically be biased upward. Why?
- This bias is minimized when $K = n$ (LOOCV), but this estimate has high variance.
- $K = 5$ or 10 provides a good compromise for this **bias-variance tradeoff**.

CV for classification problems

- Let the K parts be $C_1, C_2, ...C_K$, where C_k denotes the indices of the observations in part k. There are n_k observations in part k: if N is a multiple of K, then $n_k = n/K$.
- Compute:

$$CV_K = \sum_{k=1}^{K} \frac{n_k}{n} Err_k$$

where $Err_k = \sum_{i \in C_k} I(y_i \neq \hat{y}_i)/n_k$

CV: right and wrong

Consider a simple classifier applied to some two-class data:

1. Starting with 5000 predictors and 50 samples, find the 100 predictors having the largest correlation with the class labels.
2. We then apply a classifier such as logistic regression, using only these 100 predictors.

How do we estimate the test set performance of this classifier?
Can we apply cross-validation in step 2, forgetting about step 1?

CV: right and wrong

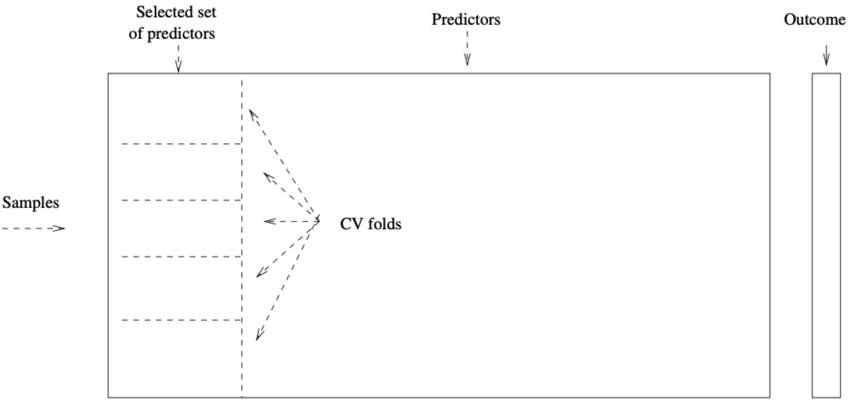
The answer is NO.

- This would ignore the fact that in Step 1, the procedure has already seen the labels of the training data, and made use of them. This is a form of training and must be included in the validation process.
- It is easy to simulate realistic data with the class labels independent of the outcome, so that true test error =50%, but the CV error estimate that ignores Step 1 is zero! Try to do this yourself
- This is a common error in genomics research.

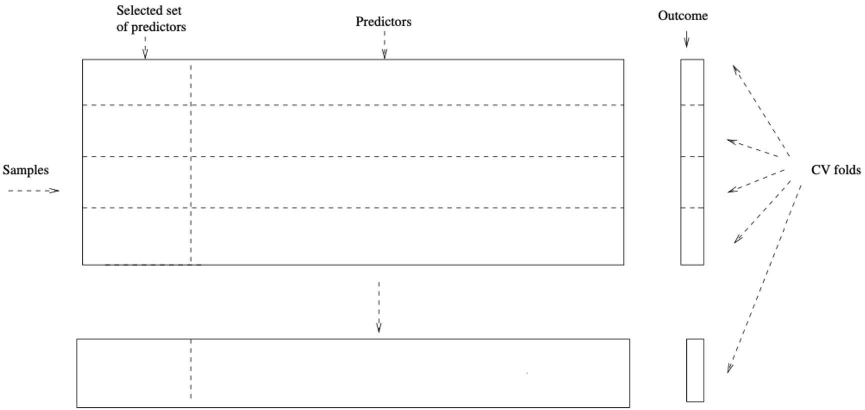
The Wrong and Right Way

- Wrong: Apply cross-validation in step 2.
- Right: Apply cross-validation to steps 1 and 2.

Wrong way


Selected set of predictors
Predictors
Outcome
Samples
CV folds

Right way


Selected set
of predictors
Predictors
Outcome
Samples
CV folds

The Bootstrap

The Bootstrap

- The bootstrap is a flexible and powerful statistical tool that can be used to quantify the uncertainty associated with a given estimator or statistical learning method.
- For example, it can provide an estimate of the standard error of a coefficient, or a confidence interval for that coefficient.

The name

- The use of the term bootstrap derives from the phrase to pull oneself up by one's bootstraps, widely thought to be based on one of the eighteenth century "The Surprising Adventures of Baron Munchausen" by Rudolph Erich Raspe:
  The Baron had fallen to the bottom of a deep lake. Just when it looked like all was lost, he thought to pick himself up by his own bootstraps.

An example

- Suppose that we wish to invest a fixed sum of money in two financial assets that yield returns of X and Y, respectively, where X and Y are random quantities.
- We will invest a fraction α of our money in X, and will invest the remaining $1 - \alpha$ in Y .
- We wish to choose α to minimize the total risk, or variance, of our investment. In other words, we want to minimize $Var(\alpha X + (1 - \alpha)Y)$.
- One can show that the value that minimizes the risk is given by:

$$\alpha = \frac{\sigma_Y^2 - \sigma_{XY}}{\sigma_X^2 + \sigma_Y^2 - 2\sigma_{XY}}$$

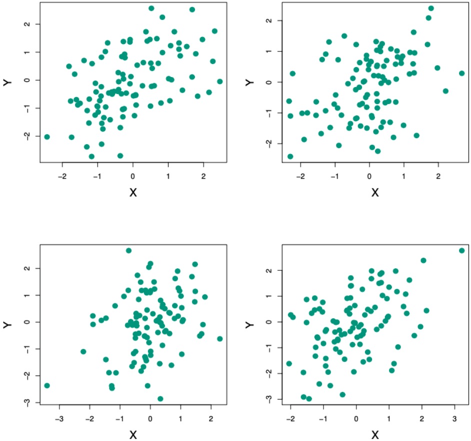
where $\sigma_X^2 = Var(X)$, $\sigma_Y^2 = Var(Y)$, and $\sigma_{XY} = Cov(X, Y)$.

An example

- But the values of σ_X^2, σ_Y^2 and σ_{XY} are unknown.
- We can compute estimates for these quantities, $\hat{\sigma}_X^2$, $\hat{\sigma}_Y^2$ and $\hat{\sigma}_{XY}$, using a data set that contains measurements for X and Y.
- We can then estimate the value of α that minimizes the variance of our investment using

$$\hat{\alpha} = \frac{\hat{\sigma}_Y^2 - \hat{\sigma}_{XY}}{\hat{\sigma}_X^2 + \hat{\sigma}_Y^2 - 2\hat{\sigma}_{XY}}$$

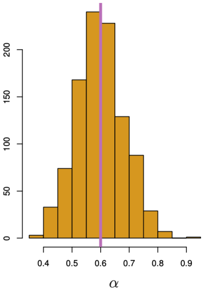
An example



Each panel displays 100 simulated returns for investments X and Y . From left to right and top to bottom, the resulting estimates for α are 0.576, 0.532, 0.657, and 0.651.

An example

- To estimate the standard deviation of $\hat{\alpha}$, we repeated the process of simulating 100 paired observations of X and Y , and estimating α 1,000 times.
- We thereby obtained 1,000 estimates for α, which we can call $\hat{\alpha}_1, \hat{\alpha}_2, ..., \hat{\alpha}_{1000}$.
- For these simulations the parameters were set to $\sigma_X^2 = 1$,$\sigma_Y^2 = 1.25$, and $\sigma_{XY} = 0.5$, and so we know that the true value of α is 0.6 (indicated by the red line).



An example

- The mean over all 1,000 estimates for α is

$$\bar{a} = \frac{1}{1000} \sum_{r=1}^{1000} \hat{\alpha}_r = 0.5996$$

very close to $\alpha = 0.6$, and the standard deviation of the estimates is

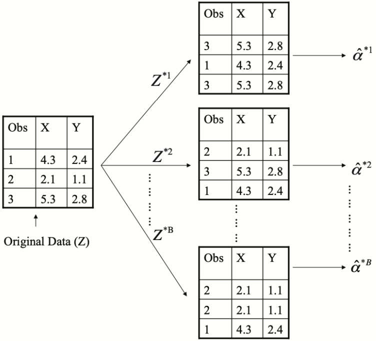
$$\sqrt{\frac{1}{1000-1} \sum_{r=1}^{1000} (\hat{\alpha}_r - \bar{\alpha})^2} = 0.083$$

- This gives us a very good idea of the accuracy of $\hat{\alpha}$: $SE(\hat{\alpha}) \approx 0.083$.

Now back to the real world

- The procedure outlined above cannot be applied, because for real data we cannot generate new samples from the original population.
- However, the bootstrap approach allows us to use a computer to mimic the process of obtaining new data sets, so that we can estimate the variability of our estimate without generating additional samples.
- Rather than repeatedly obtaining independent data sets from the population, we instead obtain distinct data sets by repeatedly sampling observations from the original data set with replacement.
- Each of these "bootstrap data sets" is created by sampling with replacement, and is the same size as our original dataset. As a result some observations may appear more than once in a given bootstrap data set and some not at all.

Example with just 3 observations



Obs	X	Y
1	4.3	2.4
2	2.1	1.1
3	5.3	2.8

Original Data (Z)

Obs	X	Y
3	5.3	2.8
1	4.3	2.4
3	5.3	2.8

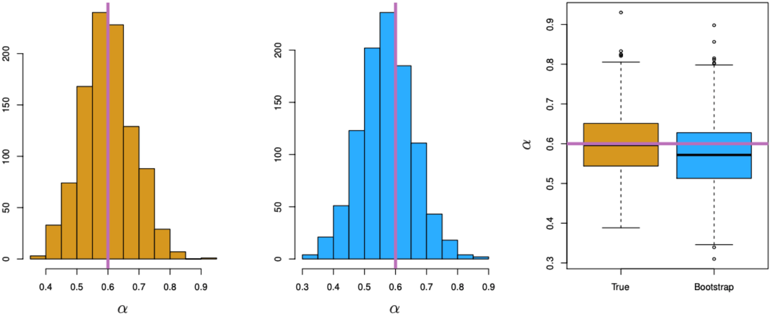
Obs	X	Y
2	2.1	1.1
3	5.3	2.8
1	4.3	2.4

Obs	X	Y
2	2.1	1.1
2	2.1	1.1
1	4.3	2.4

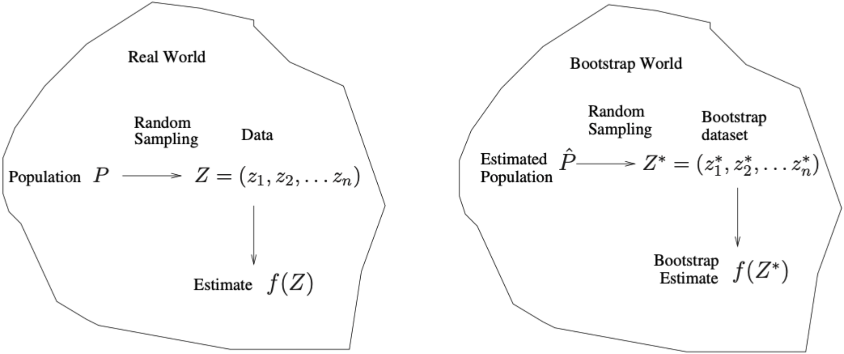
A graphical illustration of the bootstrap approach on a small sample containing $n = 3$ observations. Each bootstrap data set contains n observations, sampled with replacement from the original data set. Each bootstrap data set is used to obtain an estimate of α.

- Denoting the first bootstrap data set by Z^{*1}, we use Z^{*1} to produce a new bootstrap estimate for α, which we call α^{*1}.
- This procedure is repeated B times for some large value of B (say 100 or 1000), in order to produce B different bootstrap data sets, $Z^{*1}, Z^{*2}, ..., Z^{*B}$ and B corresponding α estimates, $\alpha^{*1}, \alpha^{*2}, ..., \alpha^{*B}$.
- We estimate the standard error of these bootstrap estimates using the formula:

$$\mathrm{SE}_B(\hat{\alpha}) = \sqrt{\frac{1}{B-1}\sum_{r=1}^{B}\left(\hat{\alpha}^{*r} - \bar{\hat{\alpha}}^{*}\right)^2}$$


0
50
100
150
200
0.4
0.5
0.6
0.7
0.8
0.9
α
0
50
100
150
200
0.3
0.4
0.5
0.6
0.7
0.8
0.9
α
0.3
0.4
0.5
0.6
0.7
0.8
0.9
α
True
Bootstrap

A general picture for the bootstrap


Real World
Random Sampling
Data
Population P
$Z = (z_1, z_2, \ldots z_n)$
Estimate $f(Z)$
Bootstrap World
Random Sampling
Bootstrap dataset
Estimated Population $\hat{P}$
$Z^* = (z_1^*, z_2^*, \ldots z_n^*)$
Bootstrap Estimate $f(Z^*)$

The bootstrap in general

- In more complex data situations, figuring out the appropriate way to generate bootstrap samples can require some thought.
- For example, if the data is a time series, we can't simply sample the observations with replacement.
- We can instead create blocks of consecutive observations, and sample those with replacements. Then we paste together sampled blocks to obtain a bootstrap dataset.

The bootstrap usage

- Primarily used to obtain standard errors of an estimate.
- Also provides approximate confidence intervals for a population parameter. For example, looking at the histogram in the middle panel of the Figure for α, the 5% and 95% quantiles of the 1000 values is (.43, .72).
- This represents an approximate 90% confidence interval for the true α.
- The above interval is called a Bootstrap Percentile confidence interval. It is the simplest method (among many approaches) for obtaining a confidence interval from the bootstrap.

Textbook chapters

- ISLR: chapter 5: 5.1 - 5.2