36-724 Spring 2006: Cross-Validation vs. Bootstrapping

Brian Junker

April 5a, 2006

- Quick Review of \(K\)-Fold Cross-Validation
- Simple Bootstrap Cross-Validation
- Leave-one-out Bootstrap Cross-Validation
- The .632 Bootstrap
Simple Bootstrap Cross-Validation

A simple bootstrap prediction error could be constructed as follows:

- Let the original data set be

\[
S = \begin{pmatrix} y_1 & x_{11} & \cdots & x_{1p} \\ \vdots & \vdots & \ddots & \vdots \\ y_N & x_{N1} & \cdots & x_{Np} \end{pmatrix}
\]

- Draw bootstrap samples \( S_b, b = 1, \ldots, B \), where

\[
S_b = \begin{pmatrix} y_{1b} & x_{11b} & \cdots & x_{1pb} \\ \vdots & \vdots & \ddots & \vdots \\ y_{Nb} & x_{N1b} & \cdots & x_{Npb} \end{pmatrix}
\]

- From each bootstrap sample \( S_b \) train our model \( \hat{f}^{sb}(x) \).
- Compute

\[
\hat{\text{Err}}_{\text{boot}} = \frac{1}{B N} \sum_{b=1}^{B} \sum_{i=1}^{N} L(y_i, \hat{f}^{sb}(x_i))
\]
**Problem:** The “full data set” act like the test set (generates $y_i$’s), and the “bootstrap samples” act like training sets (generate $\hat{f}^b(x_i)$’s).

- When $(y_i, x_i) \notin S_b$, the term $\sum_{i=1}^{N} L(y_i, \hat{f}^b(x_i))$ looks like cross-validation error;
- When $(y_i, x_i) \in S_b$, the term $\sum_{i=1}^{N} L(y_i, \hat{f}^b(x_i))$ looks like training-set error.

Since $S_b$’s are created by sampling with replacement from $S$

$$P[(y_i, x_i) \in S_b] = 1 - (1 - \frac{1}{N})^N \approx 1 - e^{-1} \approx 0.632,$$

$\hat{\text{Err}}_{\text{boot}}$ can be considerably biased downward.

---

**Leave-one-out Bootstrap Cross-Validation**

A bootstrap error estimate that tries to fix the problem is the “leave-one-out” bootstrap,

$$\hat{\text{Err}}^{(1)} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{|C^{-i}|} \sum_{b \in C^{-i}} L(y_i, \hat{f}^b(x_i))$$

where $C^{-i} = \{b : (y_i, x_i) \notin S_b\}$. Note that

- The average number of distinct elements in the $S_b$’s retained in $\hat{\text{Err}}^{(1)}$ is about $0.632 \cdot N$
- So, $\hat{\text{Err}}^{(1)}$ tends to have low-variance/high-bias for estimating $\text{Err} = E[L(Y, \hat{f}(X))]$ like 2-fold cross-validation.
The .632 Bootstrap

A compromise bootstrap error estimate is

$$\hat{\text{Err}}^{(0.632)} = (0.368) \cdot \overline{\text{err}} + (0.632) \cdot \hat{\text{Err}}^{(1)}$$

- HTF observe that
  - Derivation is complicated but basically it tries to reduce the bias of $\hat{\text{Err}}^{(1)}$ by pulling it toward the training-set error $\overline{\text{err}}$.
  - $\hat{\text{Err}}^{(0.632)}$ works well in light (under-) fitting situations, but can break down with overfit.
  - $\hat{\text{Err}}^{(0.632)}$ can be improved by adjusting the coefficients 0.368 and 0.632 for the “no-information” error rate obtained by training on a data sets in which all possible combinations $(y_i, x'_i)$ are considered.

Here is a comparison of these various prediction error estimates...