Cross-validation

Matthew S. Shotwell, Ph.D.

Department of Biostatistics
Vanderbilt University School of Medicine
Nashville, TN, USA

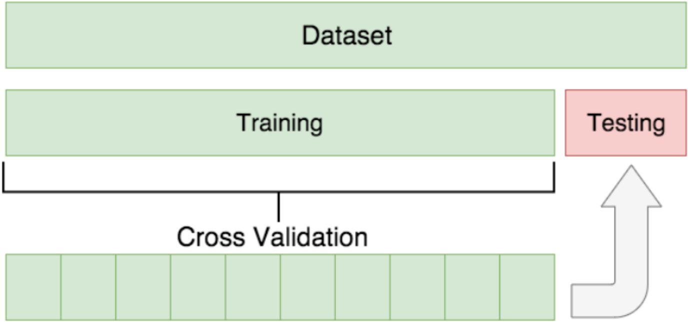
March 16, 2021

Cross-validation

- cross-validation is a method to estimate average test error:
- average test error - test error, averaged over training samples
- $\{\tau_1, \ldots, \tau_B\}$ - multiple training samples

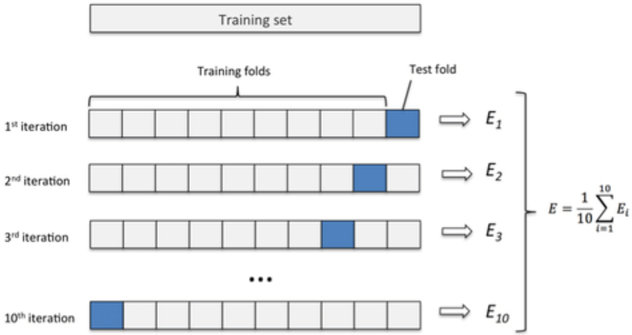
$$\mathrm{Err} = \frac{1}{B}\sum_{b=1}^{B} \mathrm{Err}_{\tau_b}$$

- use to select tuning parameters
- mimics training/test sample pairs


Dataset
Training
Testing
Cross Validation

K-fold cross-validation

1. randomly shuffle the data
2. split data into K equal parts
3. for $k = 1 \dots K$:
   3.1 fit model to $K - 1$ parts not including part k
   3.2 calculate prediction error using part k as test data


Training set
Training folds
Test fold
1st iteration
2nd iteration
3rd iteration
...
10th iteration
E_1
E_2
E_3
E_{10}
$E = \frac{1}{10}\sum_{i=1}^{10} E_i$

K-fold cross-validation

- let $\mathcal{K}_i \in \{1 \dots K\}$ be the split containing obs i
- let $\hat{f}^{-k}(X)$ be the predictor fitted without part k
- the K-fold cross-validation estimate of Err is:

$$\widehat{\mathrm{Err}} = \mathrm{CV}(\hat{f}) = \frac{1}{N} \sum_{i=1}^{N} L(y_i, \hat{f}^{-\mathcal{K}_i}(x_i))$$

K-fold cross-validation

- can also write this way

$$\widehat{\mathrm{Err}}_k = \frac{1}{N_k} \sum_{i \in \mathrm{part}\ k}^{N_k} L(y_i, \hat{f}^{-k}(x_i))$$

$$\widehat{\mathrm{Err}} = \frac{1}{K} \sum_{k=1}^{K} \widehat{\mathrm{Err}}_k$$

K-fold cross-validation

- if there is a tuning parameter α, then

$$\widehat{\mathrm{Err}}(\alpha) = \mathrm{CV}(\hat{f}_\alpha) = \frac{1}{N}\sum_{i=1}^{N} L(y_i, \hat{f}_\alpha^{-\mathcal{K}_i}(x_i))$$

- std. err. of $\widehat{\mathrm{Err}}(\alpha)$ is sample std. dev. of $\widehat{\mathrm{Err}}_k(\alpha)$
- use std. err. in "one std. err. rule": "choose the smallest model whose test error is no more than one std. err. above the test error of the best model"

K-fold cross-validation

- in code, k-fold CV often computed as follows

$$\widehat{\mathrm{Err}}_k = \frac{1}{N_k} \sum_{i \in \mathrm{part}\ k}^{N_k} L(y_i, \hat{f}^{-k}(x_i))$$

$$\widehat{\mathrm{Err}} = \frac{1}{K} \sum_{k=1}^{K} \widehat{\mathrm{Err}}_k$$

$$\mathrm{sd}(\widehat{\mathrm{Err}}) = \sqrt{\frac{1}{K} \sum_{k=1}^{K} (\widehat{\mathrm{Err}}_k - \widehat{\mathrm{Err}})^2}$$

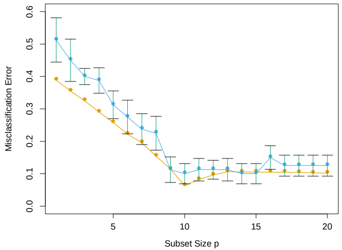
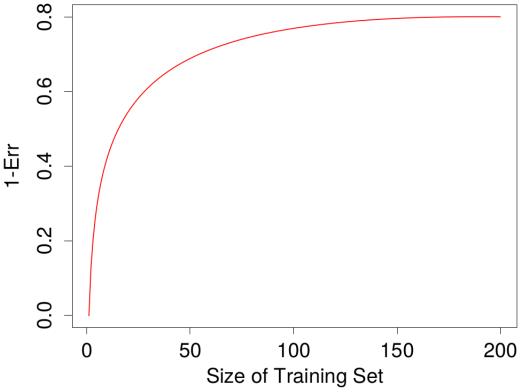


FIGURE 7.9. *Prediction error (orange) and tenfold cross-validation curve (blue) estimated from a single training set, from the scenario in the bottom right panel of Figure 7.3.*

K-fold cross-validation

- K should be selected so that each train/test split is "representative" of the overall sample
- increasing K - increasing variance, decreasing bias
- typically $K = 5$ or 10 (performs well empirically)
- $K = N$ is "leave-one-out CV"

If $\widehat{\mathrm{Err}}$ curve has big slope at training sample size, then CV estimate will be biased upward; worse for smaller K



Leave-one-out-CV

- Leave-one-out-CV:

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

- if $\hat{y} = Sy$ and $L(y, \hat{f}(x)) = (y - \hat{f}(x))^2$ then

$$\begin{aligned}\mathrm{CV}_{\mathrm{loo}}(\hat{f}) &= \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{f}^{-i}(x_i))^2 \\ &= \frac{1}{N}\sum_{i=1}^{N}\left(\frac{y_i - \hat{f}(x_i)}{1 - S_{ii}}\right)^2\end{aligned}$$

Bootstrap

- denote training data $z = \{z_1, \ldots, z_N\}$ where $z_i = (x_i, y_i)$
- purpose of bootstrap is to simulate the sampling process and summarize its effects on statistical procedures
- Bootstrap:
  1. randomly draw N items from z with replacement
  2. implement a statistical procedure
  3. repeat 1 and 2 B times
  4. summarize sampling properties of statistical procedure

Bootstrap

- consider a sample statistic $S(z)$
- denote b^{th} bootstrap sample z^{*b}
- by computing $S(z^{*b})$ for each of B bootstrap samples, we can approximate the sampling distribution of the statistic S.

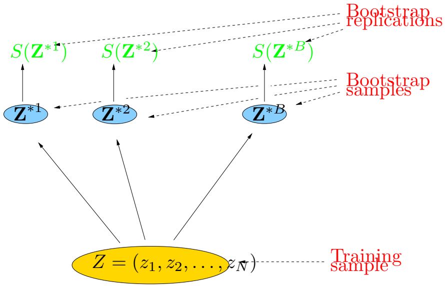


FIGURE 7.12. *Schematic of the bootstrap process. We wish to assess the statistical accuracy of a quantity $S(\mathbf{Z})$ computed from our dataset. B training sets $\mathbf{Z}^{*b}$, $b = 1, \ldots, B$ each of size N are drawn with replacement from the original dataset. The quantity of interest $S(\mathbf{Z})$ is computed from each bootstrap training set, and the values $S(\mathbf{Z}^{*1}), \ldots, S(\mathbf{Z}^{*B})$ are used to assess the statistical accuracy of $S(\mathbf{Z})$.*

Bootstrap

- bootstrap estimate of the sample variance of $S(z)$ is

$$\widehat{\mathrm{var}}[S(z)] \approx \frac{1}{B-1}\sum_{b=1}^{B}[S(z^{*b}) - \bar{S}^*]^2$$

Bootstrap validation

- approx. avg. test error by simulating the train/test process
- training data $z = \{z_1, \dots, z_N\}$ where $z_i = (x_i, y_i)$
- resampled training data $z^* = \{z_1^*, \dots, z_N^*\}$
- A boostrap estimate of average test error:

$$\widehat{\mathrm{Err}}_{\mathrm{boot}} = \frac{1}{B}\frac{1}{N}\sum_{b=1}^{B}\sum_{i=1}^{N} L(y_i, \hat{f}^{*b}(x_i))$$

- $\hat{f}^{*b}$ is fitted using z^{*b}
- overlap in data used to fit $\hat{f}^{*b}$ and to compute $\widehat{\mathrm{Err}}_{\mathrm{boot}}$
- can be too optimistic

Bootstrap validation

- A leave-one-out boostrap estimate of EPE:

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

- $\mathcal{C}^{-i}$ are the bootstrap replicates that do not contain obs i.
- can be a bit too conservative

Pros and Cons

- k-fold CV
  - k=10 or k=5 gives good tradeoff of bias and variance in $\widehat{\mathrm{Err}}$
  - sensitive to how data are split into k-folds (can be fixed)
  - less computationally intensive
- LOO CV
  - low bias but high variance in $\widehat{\mathrm{Err}}$
  - not sensitive to how data are split
  - computationally intensive
- Bootstrap validation
  - good balance of bias and variance in $\widehat{\mathrm{Err}}$
  - not sensitive to how data are split
  - computationally intensive

Code example

`kNN-CV.R`