Resampling Methods: Cross Validation for Classification and Introduction to Bootstrapping

Sections 5.1 & 5.2

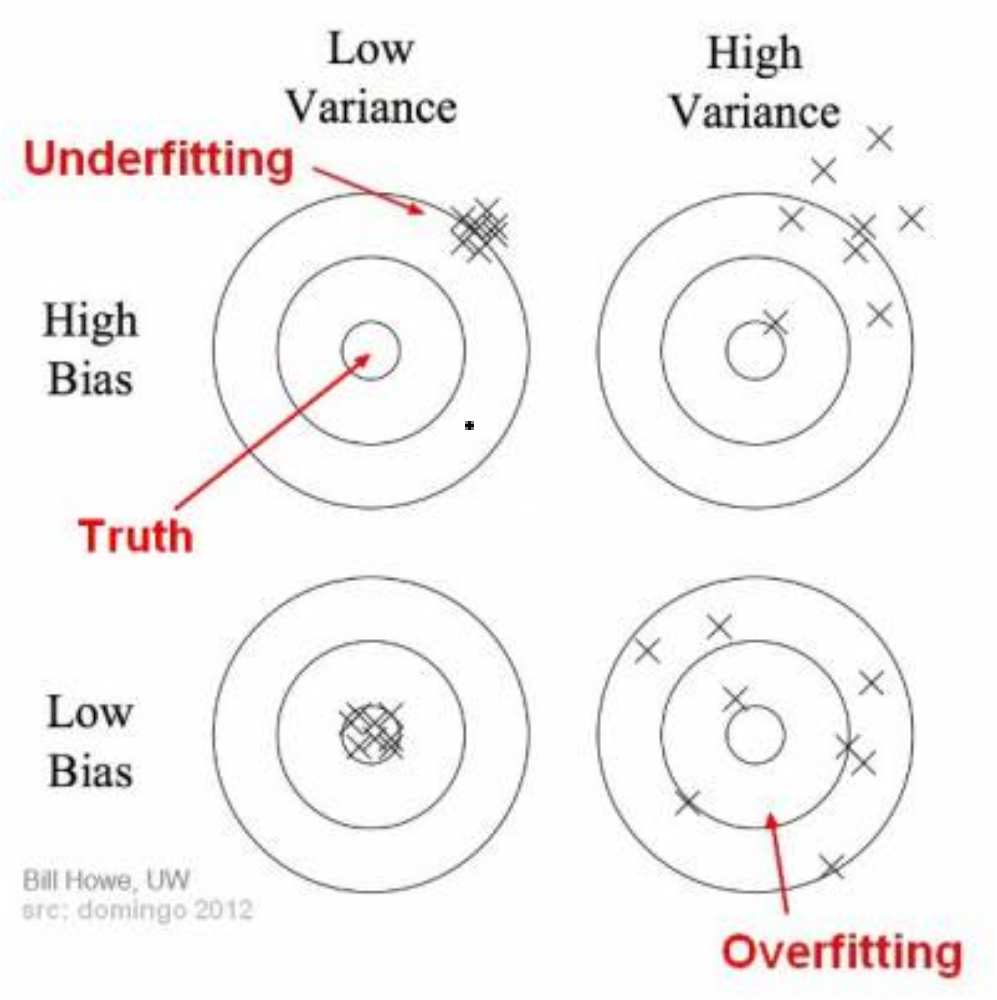
Cathy Poliak, Ph.D.
cpoliak@central.uh.edu

Department of Mathematics
University of Houston

Resampling Methods

- **Resampling methods** involve repeatedly drawing samples from a training set and refitting a model of interest on each sample in order to obtain additional information about the fitted model.
- Could potentially be computationally expensive, because they involve fitting the same statistical method multiple times using different subsets of the training data.
- Two most commonly used resampling methods:
  - Cross-validation - can be used to estimate the test error associated with a given statistical learning method in order to evaluate its performance.
  - Bootstrap - can be used to provide a measure of accuracy of a parameter estimate or of a given statistical learning method.

Variance-Bias Tradeoff



https://towardsdatascience.com/understanding-the-bias-variance-tradeoff-165e6942b229

Cross-Validation for Classification

CV can easily be used for classification when the response variable is categorical.

The biggest difference of CV for classification as opposed to regression: to estimate a test error, we use

Err = # of misclassified observations,

instead of squared error $\sum_i (y_i - \hat{y}_i)^2$

E.g. LOOCV error rate for classification

$$CV_{(n)} = \frac{1}{n} \sum_{i=1}^{n} Err_i, \quad \text{where } Err_i = \boldsymbol{I}(y_i \neq \hat{y}_i)$$

Same story for validation set approach and K-fold CV.

Example

- A data set called nodal in the `boot` library.
- Title: Nodal Involvement in Prostate Cancer
- The following variables in this data frame are:
  - m A column of ones.
  - r An indicator of nodal involvement.
  - aged The patients age dichotomized into less than 60 (0) and 60 or over 1.
  - stage A measurement of the size and position of the tumor observed by palpitation with the fingers via the rectum. A value of 1 indicates a more serious case of the cancer.
  - grade Another indicator of the seriousness of the cancer, this one is determined by a pathology reading of a biopsy taken by needle before surgery. A value of 1 indicates a more serious case of the cancer.
  - xray A third measure of the seriousness of the cancer taken from an X-ray reading. A value of 1 indicates a more serious case of the cancer.
  - acid The level of acid phosphate in the blood serum.

Description

The `nodal` data frame has 53 rows and 7 columns.

The treatment strategy for a patient diagnosed with cancer of the prostate depend highly on whether the cancer has spread to the surrounding lymph nodes. It is common to operate on the patient to get samples from the nodes which can then be analyzed under a microscope but clearly it would be preferable if an accurate assessment of nodal involvement could be made without surgery.

For a sample of 53 prostate cancer patients, a number of possible predictor variables were measured before surgery. The patients then had surgery to determine nodal involvement. It was required to see if nodal involvement could be accurately predicted from the predictor variables and which ones were most important.

Which Variables

```
> library(boot)
> nodal.glm = glm(r ~ aged + stage + grade + xray + acid,
+                 family = binomial, data = nodal)
> summary(nodal.glm)
Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -3.0794     0.9868  -3.121   0.0018 **
aged         -0.2917     0.7540  -0.387   0.6988
stage         1.3729     0.7838   1.752   0.0799 .
grade         0.8720     0.8156   1.069   0.2850
xray          1.8008     0.8104   2.222   0.0263 *
acid          1.6839     0.7915   2.128   0.0334 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Lab Question:

1. Which variables are not significant in predicting nodal involvement?

   a) All of them are needed
   b) Aged and stage
   c) Aged and grade
   d) Xray and acid

How Accurate Is This Prediction?

To answer this question we will use the LOOCV using the predictors, stage, xray, and acid and find the validation error rate. The following is the R code.

```
glm.pred = rep(0,nrow(nodal))
glm.probs = rep(0,nrow(nodal))
for (i in 1:nrow(nodal)) {
     glm.probs[i] = predict(glm(r ~ stage + xray + acid,
                                family = binomial,
                                 data = nodal[-i,]),
                            nodal[i,],type = "response")
}
glm.pred[glm.probs>0.5] = 1
mse = mean(abs(nodal$r - glm.pred))
```

2. Run the code above, what is the validation error rate?

a) 0.1887
b) 0.8113
c) 1
d) 0.5

Using the `cv.glm` Function

- First, you have set a cut-off as 0.5 (this is typically what we use). If the predicted probability is above 0.50 then we predict the observation 1.
- We use a cost function in `R`

```
cost = function(r, pi = 0) mean(abs(r-pi) > 0.5)
```

  The r is 0/1 from the original data, pi is predicted probability. So individual cost is 1 if absolute error is greater than 0.5, otherwise 0. Then, this function calculates the average error rate.

- The cut-off of 0.5 has been set before you define your cost function.
- The default is for the linear regression MSE.
- Creating the validation error rate in `R`:

```
nodal.glm = glm(r ~ stage + xray + acid, family = binomial,
                data = nodal)
(cv.err = cv.glm(nodal, nodal.glm, cost)$delta)
```

K-fold Cross Validation For Classification

Use $K = 11$

```
(cv.11.err <- cv.glm(nodal, nodal.glm, cost, K = 11)$delta)
[1] 0.2641509 0.2527590
```

3. Run the code above. Do you get the same values?
   a) Yes
   b) No

The Bootstrap Methods

- The **bootstrap** is a widely applicable and extremely powerful statistical tool that can be used to quantify the uncertainty associated with a given estimator or statistical learning method.
- The machine learning techniques that use the bootstrap is the *tree*-based models: Bagging, Random Forest, ect.
- The power of the bootstrap lies in the fact that it can be easily applied to a wide range of statistical learning methods, including some for which a measure of variability is otherwise difficult to obtain and is not automatically output by statistical software.

Idea of The Bootstrap

- Resample from the original data - either directly or via a fitted model - to create data sets, from which the variability of the quantities of interest can be assessed with out long-winded and error-prone analytical calculations.
- This approach involves repeating the original data analysis procedure with many replicate sets of data.
- The central goal is to obtain reliable standard errors, confidence intervals, and other measures of uncertainty for a wide range of problems.
- This approach can be applied in simple problems to check the adequacy of standard measures of uncertainty, to relax assumptions, and to give quick approximate solutions.
- The basic idea of bootstrap is to make inference about an estimate (such as the sample mean or sample coefficients $\hat{\beta}_j$) for a population parameter θ (such as the population mean or coefficients β_j) on sample data.

Steps for Bootstrap

1. Get a sample from a population with sample size n.

Steps for Bootstrap

1. Get a sample from a population with sample size n.
2. Draw a sample from the original sample data **with replacement** with size n, and replicate B times, each re-sampled sample is a called a **Bootstrap Sample**, and there will be totally B Bootstrap Samples.

Steps for Bootstrap

1. Get a sample from a population with sample size n.
2. Draw a sample from the original sample data **with replacement** with size n, and replicate B times, each re-sampled sample is a called a **Bootstrap Sample**, and there will be totally B Bootstrap Samples.
3. Evaluate the statistic of θ for each Bootstrap Sample, and there will be a total of B estimates of θ.

Steps for Bootstrap

1. Get a sample from a population with sample size n.
2. Draw a sample from the original sample data **with replacement** with size n, and replicate B times, each re-sampled sample is a called a **Bootstrap Sample**, and there will be totally B Bootstrap Samples.
3. Evaluate the statistic of θ for each Bootstrap Sample, and there will be a total of B estimates of θ.
4. Construct a **sampling distribution** with these B Bootstrap statistics and use it to make further statistical inference, such as:
   - Estimating the standard error of the statistic for θ.
   - Obtaining a confidence interval for θ.

Example

A thermostat used in an electrical devise is to be checked for the accuracy of its design setting of 200°F. Ten thermostats were tested to determine their actual settings, resulting in the following data:

202.2	203.4	200.5	202.5	206.3	198.0	203.7	200.8	201.3	199.0

- We wish to estimate the true mean of this thermostat.
- To understand the estimate we want to determine also the **standard error**.
- The standard error may be used to judge the precision of the statistic and/or calculate a confidence interval for the parameter that the statistic is estimating.

Bootstrap for Estimating Standard Error

- Let $x_1, x_2, \ldots, x_n$ be a random sample from a probability distribution F with mean μ and standard deviation σ.
- Consider a very simplistic statistic, the sample mean $\bar{x}$. We know the **estimated standard error** of the mean is:

$$SE(\bar{x}) = \frac{s}{\sqrt{n}} = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n(n-1)}}$$

- So SE($\bar{x}$) can be readily calculated and there is not need to estimate.
- However, there are no such simple formulas for more complicated sample statistics, as in trimmed mean or sample median.
- To explain more we will try to estimate SE($\bar{x}$).
- For our example: $\bar{x} = 201.77$ and $SE(\bar{x}) = 0.762168$.

Example of Resampling In R

```
#Resample
B = 1000 #number of resamples
M = NA #a vector of the means
for(i in 1:B) {
x = sample(temp,length(temp),replace=T)
M[i] = mean(x)
}
x #last sample in the for loop
mean(x) #mean of the last sample
mean(M) #mean of the 1000 resampled means
sd(M) #The estimated standard error of the mean
```

The `boot` Function in `R`

Performing a bootstrap analysis in `R` entails only two steps:

1. Create a function that computes that statistic of interest.
2. Use the `boot()` function, which is part of the `boot` library, to perform the bootstrap by repeatedly sampling observations form the data set with replacement.

Example of Thermostat Temperature

```
#Bootstrap Function
#install.packages("boot")
library(boot)
mean.fun <- function(dat, idx) mean(dat[idx], na.rm = TRUE)
boot.out = boot(data = temp,statistic = mean.fun,R = 1000)
boot.out
mean(boot.out$t)
```

The Ideal and Reality in Statistics World

Ideal World

- A standard error of our sample mean can be easily estimated and can find the estimated standard error.
- We assume we know or can estimate about the estimator's population.

Real World

- Hard to know the information about the population or it's distribution.
- The standard error of an estimate is hard to evaluate in general.

When the assumptions are violated, or when no formula exists for estimating standard errors, bootstrap is the powerful choice.

Why Does the Simulation of the Bootstrap Work?

Let $X_1, X_2, \ldots, x_n$ be a random sample from a population P with cumulative distribution function F. And let $M = g(X_1, X_2, \ldots, X_n)$ be our statistic for the parameter of interest. What we desire to is to know $Var(M)$. We resample B times.

By the **Law of Large Numbers**:

$$\bar{m} = \frac{1}{B}\sum_{j=1}^{B} M_j \xrightarrow{P} E(M), \text{ as } B \to \infty$$

Where $E(M)$ is the true mean of the statistic M.

In addition, the sample variance of these B statistics converges to the true variance of statistic M as $B \to \infty$.

$$s^2 = \frac{\sum_{j=1}^{B}(M_j - \bar{m})^2}{B-1} \xrightarrow{P} Var(M), \text{ as } B \to \infty$$

Where $Var(M)$ is the true variance of the statistic M.

Empirical Cumulative Distribution Function

- The ECDF (empirical cumulative distribution function) F_n is a step function with jumps i/n at observation values, where i is the number of tied observations at that value.
- Missing values are ignored.
- For observations $x = (x_1, x_2, \ldots, x_n)$, F_n is the fraction of observations less or equal to t, i.e.,

$$F_n(t) = \frac{\text{number of elements in the sample} \leq t}{n}$$

Try the Bootstrap for the Median

4. From the temperature sample, what is the estimated median?

   a) 200
   b) 201.77
   c) 201.75
   d) 0.762

The following is the code to estimate the standard error:

```
median.fun = function(dat, idx) median(dat[idx],na.rm = TRUE)
boot.out.median = boot(data = temp, statistic = median.fun, R = 1000)
boot.out.median

ORDINARY NONPARAMETRIC BOOTSTRAP

Call:
boot(data = temp, statistic = median.fun, R = 1000)


Bootstrap Statistics :
original  bias    std. error
t1*   201.75 -0.0054   0.8682336
```

5. You run the code above. Do you get exactly the same standard error?

   a) Yes
   b) No