MATH 5910

Cross Validation

Cross Validation

What is cross validation?

- Cross validation: Incrementally use entire data for validation (and training)
- Abbreviated as CV
- Standard and preferred method in practice
- Conceptually simple

Cross Validation

The K-fold CV

- Randomly divide observations n into (approximately) K equal sets (folds)
- First of the K sets set aside for validation, train on the remaining $K-1$ sets
- Repeat this for each of the K sets.
- Illustration

Cross Validation

Leave-one-out CV

- This is essentially an n-fold CV
- Validate on single data point, from the data trained on $n-1$ remaining observations
- Repeat this n times, for each observation
- No need for random permutation
- Very long history, also called jackknife

Cross Validation

Implementation

- Already implemented in R for many methods
- Packages
- Somewhat difficult to implement without package in R, but still doable

For regression

Mean squared error (MSE)

- Without CV

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

- Closely related to RSS

For regression

For leave-one-out CV:

- First, compute
$$\hat{Y}_{(i)} = \boldsymbol{x}_i' \hat{\beta}_{(i)}$$
- Where $\hat{\beta}_{(i)}$ is the estimated β without observation i (based on $n-1$ training observation)
- And $\boldsymbol{x}_i'$ is the ith row of the design matrix $\boldsymbol{X}$ (validating single data point)
- So that $\hat{Y}_{(i)}$ is predicted Y_i without observation i

For regression

Then, compute

- The CV error

$$\mathrm{CV}_n = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_{(i)})^2 = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \boldsymbol{x}_i'\hat{\beta}_{(i)})^2$$
$$= \frac{1}{n}\sum_{i=1}^{n}\left(\frac{Y_i - \hat{Y}_i}{1 - h_{ii}}\right)^2 = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\hat{e}_i}{1 - h_{ii}}\right)^2$$

  where h_{ii} is the ith diagonal element of the hat matrix $\boldsymbol{H}$

- The above quantity is also called the predicted sum of squares (PRESS) residual - see Problem 9.13, page 230 of ALR4
- Possible to do with K-fold

Classification

K-fold natural for classification:

- Compute misclassification rate, Err, for each of K folds
- K-fold CV error: Average misclassification rates for K folds

$$\mathrm{CV}_K = \frac{1}{K}\sum_{k=1}^{K} Err_k$$

CV Main Usage

In general

- Model selection
- Parameter selection

Demonstration

For regression, in R

- Implementing CV
- Both leave-one-out and K-fold
- Need `boot` package

Example

Load `boot` library, look at function `cv.glm()`

```
library(boot)

?cv.glm

data(mammals, package="MASS")
mammals
mammals.glm <- glm(log(brain) ~ log(body), data = mammals)
```

Example

Leave-one-out CV

```
(cv.err <- cv.glm(mammals, mammals.glm)$delta)
```

Example

6-fold CV

```
(cv.err.6 <- cv.glm(mammals, mammals.glm, K = 6)$delta)
```

You get different answers each time

Example

Can try to set seed (to get consistent answers)

```
set.seed(123)
(cv.err.6 <- cv.glm(mammals, mammals.glm, K = 6)$delta)
```

Example

Can do manually for regression (using the formula)

$$\mathrm{CV}_n = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_{(i)})^2 = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \boldsymbol{x}_i'\hat{\beta}_{(i)})^2$$

$$= \frac{1}{n}\sum_{i=1}^{n}\left(\frac{Y_i - \hat{Y}_i}{1 - h_{ii}}\right)^2 = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\hat{e}_i}{1 - h_{ii}}\right)^2$$

by

```
muhat <- fitted(mammals.glm)
mammals.diag <- glm.diag(mammals.glm)
?glm.diag
mean((mammals.glm$y - muhat)^2/(1 - mammals.diag$h)^2)
cv.err
```

Cross Validation

Challenge

- Create your own K-fold CV functions
- Be careful with each method

More later