

Cross-Validation

Let $\kappa : \{1, \dots, N\} \rightarrow \{1, \dots, K\}$ and denote by \hat{f}^{-k} for $k = 1, \dots, K$ the estimator of f based on the data (x_i, y_i) with $\kappa(i) \neq k$.

The (x_i, y_i) with $\kappa(i) = k$ work as a test dataset for \hat{f}^{-k} and

$$\text{E}\hat{\text{P}}\text{E}(\hat{f}^{-k}) = \frac{1}{N_k} \sum_{i:\kappa(i)=k} L(y_i, \hat{f}^{-k}(x_i))$$

with $N_k = |\{i | \kappa(i) = k\}|$

The K -fold κ -cross-validation estimator of Err is the weighted average

$$\begin{aligned} \text{CV}_\kappa &= \sum_{k=1}^K \frac{N_k}{N} \text{E}\hat{\text{P}}\text{E}(\hat{f}^{-k}) \\ &= \frac{1}{N} \sum_{i=1}^N L(y_i, \hat{f}^{-\kappa(i)}(x_i)) \end{aligned}$$

Figure 7.8 – Err as a Function of N

We should write $\text{Err} = \text{Err}(N)$ as a function of the sample size. If $\hat{f}_N, f \in \mathcal{F}$ and f minimizes EPE then $\text{EPE}(\hat{f}_N) \geq \text{EPE}(f)$ and

$$\text{Err}(N) = E(L(Y, \hat{f}_N(X))) \geq \text{EPE}(f)$$

If we have a **consistent** estimator; $\hat{f}_N \rightarrow f$, then

$$\text{Err}(N) \rightarrow \text{EPE}(f).$$

Cross-Validation

Among several models we will choose the model with smallest CV_κ . How to choose κ ? How to choose K ?

We aim for $N_1 = \dots = N_K$ in which case

$$E(CV_\kappa) = \text{Err}(N - N_1).$$

With a **steep learning curve** at N we need N_1 to be small or we **underestimate** Err .

Extreme case; N -fold or **leave-one-out** cross-validation with $\kappa(i) = i$ leads to an almost unbiased estimator of $\text{Err}(N)$, but the strong correlation of the $\text{EPE}(\hat{f}^{-i})$'s works in the direction of given a larger variance. Recommendations are that 5- or 10-fold CV is a good compromise between bias and variance.

CV and Linear Smoothers

For many linear smoothers where $\hat{\mathbf{f}} = \mathbf{S}\mathbf{y}$ the **leave-one-out** CV estimator for squared error loss can be computed as

$$CV = \frac{1}{N} \sum_{i=1}^N \left[\frac{y_i - \hat{\mathbf{f}}_i}{1 - \mathbf{S}_{ii}} \right]^2$$

This is a computational gain as there is no need for N successive reestimations.

The **generalized cross-validation** estimator is defined as

$$GCV = \frac{1}{N} \sum_{i=1}^N \left[\frac{y_i - \hat{\mathbf{f}}_i}{1 - \text{trace}(\mathbf{S})/N} \right]^2$$

Details are in Theo.6.

The Wrong and The Right Way to Cross-Validate

- Mess with the data to find variables/methods that seem to be useful.
- Estimate parameters using the selected variables/methods and use cross-validation to choose tuning parameters.

WRONG

Don't mess with the data before the cross-validation.

Cross-Validation must be out side of all modeling steps, including filtering or variable selection steps.

Bootstrapping

With B bootstrap datasets and \hat{f}^{*b} the estimated predictor based on the b 'th bootstrap dataset with indices C_b we produce the estimator

$$\text{E}\hat{\text{PE}}(\hat{f}^{*b}) = \frac{1}{N_b} \sum_{i \notin C_b} L(y_i, \hat{f}^{*b}(x_i))$$

using (x_i, y_i) for $i \notin C_b$ as the test data for \hat{f}^{*b} and $N_b = N - |C_b|$

A natural estimator of Err is

$$\hat{\text{Err}} = \frac{1}{B} \sum_{b=1}^B \text{E}\hat{\text{PE}}(\hat{f}^{*b}) = \frac{1}{B} \sum_{b=1}^B \frac{1}{N_b} \sum_{i \notin C_b} L(y_i, \hat{f}^{*b}(x_i)).$$

Bootstrapping

Turning things inside-out; $\text{Err} = E(E(L(Y, \hat{f}(X))|X, Y))$ we can suggest the **leave-one-out** bootstrap estimator

$$\hat{\text{Err}}^{(1)} = \frac{1}{N} \sum_{i=1}^N \frac{1}{M_i} \sum_{b: i \notin C_b} L(y_i, \hat{f}^{*b}(x_i))$$

with $M_i = B - |\{b | i \in C_b\}|$.

As the average number of distinct variables in a bootstrap sample is $1 - (1 - N^{-1})^N \simeq 0.632$ both $\hat{\text{Err}}$ and $\hat{\text{Err}}^{(1)}$ are expected to estimate something like $\text{Err}(0.632N)$ rather than $\text{Err}(N)$.

A correction is suggested

$$\hat{\text{Err}}^{(.632)} = 0.368\bar{\text{err}} + 0.632\hat{\text{Err}}^{(1)}$$

Estimates of Expected Prediction Error

If \hat{f} is estimated based on a data set, we can only get an estimate of $\text{EPE}(\hat{f})$ by an **independent** test set $(x_1, y_1), \dots, (x_B, y_B)$ as

$$\text{EPE}(\hat{f}) = \frac{1}{B} \sum_{b=1}^B L(y_b, \hat{f}(x_b)).$$

Bootstrap and cross-validation provide estimates $\hat{\text{Err}}$ of the generalization error.

- $\text{EPE}(\hat{f})$ is a random variable with mean Err .
- $\hat{\text{Err}}$ is a random variable with mean Err .

Can $\hat{\text{Err}}$ be regarded as an approximation/estimate of $\text{EPE}(\hat{f})$?

Figure 7.15 – The Relation Between $\hat{E}rr$ and $EPE(\hat{f})$

Classification and The Confusion Matrix

For a classifier with two groups we can decompose the errors:

Observed y	Predicted y	
	1	0
1	$\Pr(Y = 1, f(X) = 1)$	$\Pr(Y = 1, f(X) = 0)$
0	$\Pr(Y = 0, f(X) = 1)$	$\Pr(Y = 0, f(X) = 0)$

This is the **confusion matrix** and

$$\text{EPE}(f) = \Pr(Y = 0, f(X) = 1) + \Pr(Y = 1, f(X) = 0).$$

As with $\text{EPE}(\hat{f})$ the confusion matrix can only be estimated using an independent test dataset. “Estimates” based on e.g. cross-validation are estimates of $E(\Pr(Y = k, \hat{f}(X) = l))$.

Generalized Additive Models

A generalized **additive** model of Y given X is given by a **link function** g such that the **mean** $\mu(X)$ of Y given X is

$$g(\mu(X)) = \alpha + f_1(X_1) + \dots + f_p(X_p).$$

This is an extension from general linear models by allowing for non-linear but univariate effects given by the f_i -functions.

The functions are not in general identifiable – and we can face a problem similar to **collinearity**, which is known as **concurvity**.

Recall Naive Bayes

If the X -coordinates are independent given the Y , then

$$g_k(x) = \prod_{i=1}^p g_{k,i}(x_i)$$

with $g_{k,i}$ univariate densities.

$$\begin{aligned} \log \frac{\Pr(Y = k|X = x)}{\Pr(Y = K|X = x)} &= \log \frac{\pi_k}{\pi_K} + \log \frac{g_k(x)}{g_K(x)} \\ &= \log \frac{\pi_k}{\pi_K} + \underbrace{\sum_{i=1}^p \log \frac{g_{k,i}(x_i)}{g_{K,i}(x_i)}}_{h_{k,i}(x_i)} \\ &= \log \frac{\pi_k}{\pi_K} + \sum_{i=1}^p h_{k,i}(x_i) \end{aligned}$$

Generalized Additive Logistic Regression

An important example arise with $Y \in \{0, 1\}$ with the **logit link**

$$g(\mu) = \log \left(\frac{\mu}{1 - \mu} \right), \quad \mu \in (0, 1)$$

Then

$$\mu(X) = \Pr(Y = 1|X) = \frac{\exp(\alpha + f_1(X_1) + \dots + f_p(X_p))}{1 + \exp(\alpha + f_1(X_1) + \dots + f_p(X_p))}$$

Like logistic regression we can use other link functions like the **probit link**

$$g(\mu) = \Phi^{-1}(\mu)$$

where Φ is the distribution function for the normal distribution.

Penalized Estimation

The general, penalized minus-log-likelihood function is

$$l_N(\alpha, f_1, \dots, f_p) + \sum_{j=1}^p \lambda_j \int_a^b f_j''(x) dx$$

with **tuning parameters** $\lambda_1, \dots, \lambda_p$. The minimizer, if it exists, consists of natural cubic splines. For identification purposes we assume

$$\sum_{i=1}^N f_j(x_{ij}) = 0, \quad j = 1, \dots, p.$$

This is equivalent to $\mathbf{f}_j = (f_j(x_{1j}), \dots, f_j(x_{Nj}))^T$ being perpendicular to the column vector $\mathbf{1}$ for $j = 1, \dots, p$. The penalization resolves overparameterization problems for the non-linear part **but not the linear part** of the fit.

Informal Tests for Non-Linear Effects

Using smoothing splines there is to each estimated function \hat{f}_j an associated **linear smoother matrix** \mathbf{S}_j – where the linear fit has been removed.

The **effective degrees of freedom** for the non-linear part of the fit is

$$\text{df}_j = \text{trace}(\mathbf{S}_j) - 1$$

Many implementations perform ad hoc χ^2 -tests for the non-linear part using χ^2 -distributions with df_j degrees of freedom – these tests are at best justified by some simulation studies, and can be used as guidelines only.

Spam Email Classification

Whether an email is a spam email or a regular email is a great example of a problem where **prediction** is central and **interpretation** is secondary.

The (simplistic) example in the book deals with 4601 emails to an employee at Hewlett-Packard.

Each email is **dimension reduced** to a 57-dimensional vector containing

- Quantitative variables of **word or special character percentages**.
- Quantitative variables describing the occurrence of **capital letters**.

Figure 9.1 – Non-linear Email Spam Predictor Effects