Linear-Nonlinear Regression

Ethan Ligon

April 25, 2022

Linear-in-Parameter Models

We've brushed over models which are non-linear in parameters (with the notable exception of the logit). Why?

- Problems linear in parameters generally much easier to estimate; criterion functions often quadratic; normal equations linear.
- Perhaps more important: linear models can nevertheless be very effective for estimating non-linear phenomena.

Estimating non-linear phenomena with linear models

Our basic linear regression model is:

$$y = X\beta + u.$$

But the linearity that's important for estimation here is the linearity in parameters. We can just as well have

$$y = f(X) + u, \qquad \text{with } f(X) = \hat{f}(X;\alpha) + \epsilon;$$

where

$$\hat{f}(X;\alpha) = \sum_{k=1}^{K} \alpha_k \phi_k(X);$$

The (ϕ_k) are *basis* functions with which we can try to approximate f. Note linearity in parameters α!

Stepwise Basis Functions

For a function f defined over an interval $(0, 1)$ define:

K	$\phi_1(x)$	$\phi_2(x)$	$\phi_3(x)$	$\phi_4(x)$
2	$\mathbb{1}\{x \leq \frac{1}{2}\}$	$\mathbb{1}\{x > \frac{1}{2}\}$		
3	$\mathbb{1}\{x \leq \frac{1}{3}\}$	$\mathbb{1}\{\frac{1}{3} > x \leq \frac{2}{3}\}$	$\mathbb{1}\{x > \frac{2}{3}\}$	
4	$\mathbb{1}\{x \leq \frac{1}{4}\}$	$\mathbb{1}\{\frac{1}{4} > x \leq \frac{1}{2}\}$	$\mathbb{1}\{\frac{1}{2} < x \leq \frac{3}{4}\}$	$\mathbb{1}\{x > \frac{3}{4}\}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

Radial Basis Functions

If $x \in \mathbb{R}^n$, define a set of *centers* $c_k \in \mathbb{R}^n$, and let

$$\phi_k(x) = K(x, c_k) = e^{-\frac{1}{2}\|x-c_k\|^2}.$$

Gram Matrix

This may seem as though we then need to choose a bunch of non-linear parameters, but consider letting $c_k = x_k$, where x_k is the kth observation in a dataset; then we have:

$$\boldsymbol{K} = \begin{bmatrix} K(x_1,x_1) & K(x_1,x_2) & \cdots & K(x_1,x_N) \\ K(x_2,x_1) & K(x_2,x_2) & \cdots & K(x_2,x_N) \\ \vdots & \vdots & \ddots & \vdots \\ K(x_N,x_1) & K(x_N,x_2) & \cdots & K(x_N,x_N) \end{bmatrix}$$

Other Basis Functions

- Polynomials
- Splines
- Piecewise linear functions
- Periodic functions. . .

“Over-fitting” & MSE

We can fit any given dataset better by increasing the number of functions in the basis. However, at some point improving the fit for a *given* sample makes the fit worse for a *different* sample.

Mean Squared Error

In a given sample, large deviations from true f are evidence of either a large *bias* or large *variance*.
In this case we can compute the squared bias of this particular estimated function $\hat{f}$ by

$$\text{MSE}(\hat{f}) = \int \left(f(x) - \hat{f}(x)\right)^2 dF(x),$$

where $F(x)$ is the CDF of X.

Leave-one-out estimators

Whatever estimator we have of $\hat{y}(X)$, we presume that can be estimated using observations $j = 1, \ldots, N$.

Estimator

An old idea for improving out-of-sample predictive power is the "leave-one-out" estimator. In this case for each observation j we construct a prediction function using every observation *except* j; i.e.,

$$\hat{y}_{-j}(X) = \hat{f}(X|y_{-j}, X_{-j}) \qquad j = 1, \ldots, N.$$

That is, $\hat{y}_{-j}$ is estimated using data (y_{-j}, X_{-j}), but can then be evaluated at any X. Then the usual leave-one-out estimator is

$$\hat{y}(X) = \frac{1}{N} \sum_{j=1}^{N} \hat{y}_{-j}(X).$$

Note that *every single* $\hat{y}_{-j}(X_j)$ is an out-of-sample prediction, since (y_j, X_j) weren't used to construct $\hat{y}_{-j}$.

Cross-validation criterion

Let $e_{-j} = y_j - \hat{y}_{-j}(X_j)$. Call this the "leave-one-out error".

Criterion

Define the *cross-validation* criterion as

$$\mathsf{CV} = \frac{1}{N}\sum_{j=1}^{N} e_{-j}^2.$$

Given the iid sampling assumption it's easy to see that this is an unbiased estimator of the expected squared out-of-sample prediction error.

General Approach

If we have multiple possible models or estimators of the relationship between y and X, say $(\hat{y}^1(X), \ldots, \hat{y}^q(X))$ we *select* a model by choosing the one that minimizes the CV criterion.

Parameter Estimation

Rather than selecting among a finite number of discrete “models”, we can also use the CV criterion as the basis for selecting a vector of continuous parameters.

K-fold Cross-Validation

A practical problem is that as N grows large our proposal to choose models by minimizing CV become impractical; typically the computational cost of implementing the estimator becomes $O(N^2)$ or worse. Enter K-fold cross-validation.

1. Divide sample of N observations into K different "folds" $d_{(k)}$, each of roughly size N/K.
2. Estimate a "leave N/K out" estimator that uses data from $K-1$ folds to estimate $\hat{y}^{(-k)}(X)$.
3. Let
$$e_j^{(k)} = y_j - \hat{y}^{(-k)}(X_j) \qquad \text{for } j \in d_{(k)}.$$
4. The criteria
$$\mathsf{CV}^{(k)} = \sum_{k=1}^{K} \sum_{j \in d_{(k)}} (e_j^{(k)})^2$$
then approximates the CV criterion. (If $K = N$ the two are identical).

Expected MSE

We can think of the *expected* MSE as the limit we'd reach taking averages in repeated samples. (This can be estimated in a given finite sample by our Cross-Validation measure). If $\hat{f}_m$ is estimated using a sample $m = 1, \ldots, M$, then

$$\mathsf{CV}_M = \frac{1}{M}\sum_{m=1}^{M} \mathsf{MSE}(\hat{f}_m) \xrightarrow{p} \mathsf{EMSE}.$$

Various Penalizations

A variety of approaches to trying to encourage models with fewer parameters: goal is to achieve balance between bias and variance, say by minimizing EMSE.

- Adjusted R^2: $1-(1-R^2)\frac{N-1}{N-k-1}$
- Akaike Information Criterion: $N(1+\log 2\pi\hat{\sigma}^2)+2k$
- Bayesian Information Criterion: $N(1+\log 2\pi\hat{\sigma}^2)+k\log N$

Loss-Penalty Form

A really wide variety of estimators can be written in so called "loss-penalty" form, where we try to choose a vector of parameters b to solve

$$\min_{b\in B} L(b) + \lambda\|b\|.$$

The first term is something like (minus) a log-likelihood, or the GMM criterion, or some other loss function. The second term is a "penalty" function, which induces a bias toward making the parameters b small (perhaps zero). The parameter $\lambda > 0$ is a "tuning" parameter; larger values "penalize" large b more, increasing bias so as to reduce variance.

Effective Degrees of Freedom

Consider a regression linear in K parameters; then the model can be represented as

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{a} + \boldsymbol{e}.$$

Let $\boldsymbol{V} = \boldsymbol{X}^\top\boldsymbol{X} - \bar{\boldsymbol{X}}\bar{\boldsymbol{X}}^\top$ be the covariance matrix of $\boldsymbol{X}$, and let d_k^2 be the eigenvalues of this matrix. Then the *effective degrees of freedom* for the regression in Loss-Penalty form is

$$df(\lambda) = \sum_{k=1}^{K} \frac{d_k^2}{d_k^2 + \lambda}.$$

The Lasso (Least Absolute Shrinkage and Selection Operator)

The Lasso takes the form

$$\min_{b \in B} \sum_{j=1}^{N} (y_j - X_j b)^2 + \lambda \sum_{k=1}^{K} |b_k|$$

The absolute value penalty (L_1 norm) means that the method will try to set coefficients to zero where doing so doesn't compromise the fit too much. Thus, the larger λ the fewer non-zero coefficients we expect to see (think, the more parsimonious the regression specification).

Tuning

So, how should we choose λ? Too big, and we increase bias; too small we increase variance. Note that in the Lasso case choosing *one* parameter can the the effect of introducing or eliminating *lots* of parameters.

Cross-Validation

The cross-validation tools we discussed last time have many uses, but one very simple and effective use case involves tuning just a single parameter to try and minimize MSE. Let

$$\mathsf{CV}(\lambda) = \frac{1}{N}\sum_{j=1}^{N} e_{-j}(\lambda)^2;$$

Then choose

$$\lambda^* = \operatorname*{arg\,min}_{\lambda\in\mathbb{R}_+} \mathsf{CV}(\lambda).$$