2022-10-04

1 Choice of regularization

All of the regularization methods we have discussed share a common trait: they define a parametric family of models. With more regularization, we restrict the range of models we can easily generate (adding bias), but we also reduce the sensitivity of the fit (reducing variance). The choice of the regularization parameter is a key aspect of these methods, and we now briefly discuss three different ways of systematically making that choice. In all cases, we rely on the assumption that the sample observations we use for the fit are representative of the population of observations where we might want to predict.

1.1 Morozov's discrepancy principle

Suppose that we want to fit $Ax \approx \hat{b}$ by regularized least squares, and the (noisy) observation vector $\hat{b}$ is known to be within some error bound $\|e\|$ of the true values b. The discrepancy principle says that we should choose the regularization parameter so the residual norm is approximately $\|e\|$. That is, we seek the most stable fitting problem we can get subject to the constraint that the residual error for the regularized solution (with the noisy vector $\hat{b}$) is not much bigger than we would get from unknown true solution.

One of the most obvious drawbacks of the discrepancy principle is that it requires that we have an estimate for the norm of the error in the data. Sadly, such estimates are not always available.

1.2 The L-curve

A second approach to the regularization parameter is the *L-curve*. If we draw a parametric curve of the residual error versus solution norm on a log-log plot, with $\log \|r_\lambda\|$ on the x axis and $\log \|x_\lambda\|$ on the y axis, we often see an "L" shape. In the top of the vertical bar (small λ), we find that increasing regularization decreases the solution norm significantly without significantly increasing the residual error. Along the end of the horizontal part, increasing regularization increases the residual error, but does not significantly help with the solution norm. We want the corner of the curve, where the regularization

is chosen to minimize the norm of the solution subject to the constraint that the residual is close to the smallest possible residual (which we would have without regularization).

Computing the inflection point on the L-curve is a neat calculus exercise which we will not attempt here.

1.3 Cross-validation

The idea with cross-validation is to choose the parameter by fitting the model on a subset of the data and testing on the remaining data. We may do this with multiple partitions into data used for fitting versus data reserved for checking predictions. We often choose regularization parameters to give the smallest error on the predictions in a cross-validation study.

One variant of cross-validation involves minimizing the *leave-one-out cross-validation* (LOOCV) statistic:

$$\text{LOOCV} = \frac{1}{m} \sum_{i=1}^{m} \left[Ax^{(-i)} - b \right]_i^2,$$

where $x^{(-i)}$ denotes the model coefficients fit using all but the ith data point.

To compute the LOOCV statistic in the most obvious way, we would delete each row a_i^T of A in turn, fit the model coefficients $x^{(-i)}$, and then evaluate $r^{(-i)} = b_i - a_i^T x^{(-i)}$. This involves m least squares problems, for a total cost of $O(m^2n^2)$ (as opposed to the usual $O(mn^2)$ cost for an ordinary least squares problem). Let us find a better way! For the sake of concreteness, we will focus on the Tikhonov-regularized version of the problem

The key is to write the equations for $x^{(-i)}$ as a small change to the equations for $(A^TA + \lambda^2 I)x^* = A^Tb$:

$$(A^TA + \lambda^2 I - a_i a_i^T)x^{(-i)} = A^Tb - a_i b_i.$$

This subtracts the influence of row i from both sides of the normal equations. By introducing the auxiliary variable $\gamma = -a_i^T x^{(-i)}$, we have

$$\begin{bmatrix} A^TA + \lambda^2 I & a_i \\ a_i^T & 1 \end{bmatrix} \begin{bmatrix} x^{(-i)} \\ \gamma \end{bmatrix} = \begin{bmatrix} A^Tb - a_i b_i \\ 0 \end{bmatrix}.$$

Eliminating $x^{(-i)}$ gives

$$(1 - \ell_i^2)\gamma = \ell_i^2 b_i - a_i^T x^*$$

where $\ell_i^2 = a_i^T (A^T A + \lambda^2 I)^{-1} a_i$ is called the *leverage score* for row i. Now, observe that if $r = b - Ax^*$ is the residual for the full problem, then

$$(1 - \ell_i^2) r^{(-i)} = (1 - \ell_i^2)(b_i + \gamma) = (1 - \ell_i^2) b_i + \ell_i^2 b_i - a_i^T x_* = r_i,$$

or, equivalently

$$r^{(-i)} = \frac{r_i}{1 - \ell_i^2}.$$

We finish the job by observing that ℓ_i^2 is the ith diagonal element of the orthogonal projector $\Pi = A(A^T A + \lambda I) A^{-1}$, which we can also write in terms of the economy QR decomposition

$$\begin{bmatrix} A \\ \lambda I \end{bmatrix} = \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} R$$

as $\Pi = Q_1 Q_1^T$. Hence, ℓ_i^2 is the squared row sum of Q_1 in the QR factorization.

2 Linearly constrained case

Consider the weighted least squares problem

$$\text{minimize } \sum_{i=1}^{m} w_i r_i^2$$

where w_1 is much larger than the others. If we let $w_1 \to \infty$ while the others are fixed, what happens? We essentially say that we care about enforcing the first equation above all others, and in the limit we are solving the *constrained* least squares problem

$$\text{minimize } \sum_{i=2}^{m} w_i r_i^2 \text{ s.t. } r_1 = 0.$$

Unfortunately, if we actually try to compute this way, we are dancing on dangerous ground; as w_1 goes to infinity, so does the condition number of the least squares problem. But this is only an issue with the weighted formulation; we can formulate the constrained problem in other ways that are perfectly well-behaved.

In the remainder of this section, we address two ways of handling the linearly constrained least squares problem

$$\text{minimize } \|Ax - b\|^2 \text{ s.t. } C^T x = d,$$

by either eliminating variables (the *null-space method*) or adding variables (the method of *Lagrange multipliers*).

2.1 Null space method

In the null space method, we write an explicit expression for the solutions to $C^T x = d$ in the form $x^p + Wz$ where x^p is a particular solution to $C^T x^p = d$ and W is a basis for the null space of C^T. Perhaps the simplest particular solution is $x^p = (C^T)^\dagger d$, the solution with minimal norm; we can compute both this particular solution and an orthogonormal null space basis quickly using a full QR decomposition of C:

$$C = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \begin{bmatrix} R_1 \\ 0 \end{bmatrix}, \quad x^p = Q_1 R_1^{-T} d, \quad W = Q_2.$$

Note that

$$C^T x^p = (R_1^T Q_1^T) x^p = d,$$

so this is indeed a particular solution. Having written an explicit parameterization for all solutions of the constraint equations, we can minimize the least squares objective with respect to the reduced set of variables

$$\text{minimize } \|A(x^p + Wz) - b\|^2 = \|(AW)z - (b - Ax^p)\|^2.$$

This new least squares problem involves a smaller set of variables (which is good); but in general, even if A is sparse, AW will not be. So it is appropriate to have a few more methods in our arsenal.

2.2 Lagrange multipliers

An alternate method is the method of *Lagrange multipliers*. This is an algebraic technique for adding equations to enforce constraints.

One way to approach the Lagrange multiplier method is to look at the equations for a constrained minimum. In order not to have a downhill direction, we require that the directional derivatives be zero in any direction

consistent with the constraint; that is, we require $Cx = d$ and

$$\delta x^T A^T r = 0 \text{ when } C^T \delta x = 0.$$

The constraint says that admissible δx are orthogonal to the columns of C; the objective tells us the admissible δx should be orthogonal to the residual. So we need that $A^T r$ should lie in the column span of C; that is,

$$A^T r = -C\lambda$$

for some λ, and $Cx = d$. Putting this together, we have the KKT equations

$$\begin{bmatrix} A^T A & C \\ C^T & 0 \end{bmatrix} \begin{bmatrix} x \\ \lambda \end{bmatrix} = \begin{bmatrix} A^T b \\ d \end{bmatrix}.$$

These bordered normal equations are not the end point for constrained least squares with Lagrange multipliers, any more than the normal equations are the end point for unconstrained least squares. Rather, we can use this as a starting point for clever manipulations involving our favorite factorizations (QR and SVD) that reduce the bordered system to a more computationally convenient form.

3 Quadratically constrained least squares

We end the lecture by commenting on the *quadratically constrained* least squares problem

$$\text{minimize } \frac{1}{2}\|Ax - b\|^2 \text{ s.t. } \|x\|_M^2 \leq \rho^2$$

for some positive definite matrix M. Again applying the method of Lagrange multipliers, we have that either $\|A^\dagger b\|_M^2 \leq \rho^2$ (i.e. the constraint is inactive) or we seek a stationary point of

$$\mathcal{L}(x, \lambda) = \frac{1}{2}\|Ax - b\|^2 + \frac{\lambda}{2}(x^T M x - \rho^2),$$

and taking variations with respect to x gives us

$$\nabla_x \mathcal{L} = A^T(Ax - b) + \lambda M x = (A^T A + \lambda M)x - A^T b = 0.$$

That is, if the constrained problem is active, we are actually solving a Tikhonov-regularized least squares problem, with the Lagrange multiplier serving the role of the regularization parameter.

4 Iteratively reweighted least squares

We conclude with a brief example of how least squares can be used as a building block for related functions. As an example, consider replacing the least squares loss with an alternate loss function:

$$\text{minimize } \sum_i \phi(r_i) \text{ s.t. } r = Ax - b$$

where ϕ is a continuous symmetric function such that $\phi(0) = 0$. A common example is the *Huber* loss function

$$\phi_\delta(r) = \begin{cases} \frac{1}{2}r^2, & |r| \leq \delta \\ \delta\left(|r| - \frac{1}{2}\delta\right), & \text{otherwise.} \end{cases}$$

Optimizing the Huber loss is much less sensitive to outliers than the least squares loss. Other loss functions, such as the Tukey biweight, are even less sensitive to outliers (but are nonconvex, and may lead to a non-unique optimization problem).

How do we minimize the Huber loss? There are several options, but one of the most popular is the *iteratively reweighted least squares* (IRLS) algorithm. To derive the algorithm, we write the stationary conditions as

$$\delta r^T(\psi(r) \odot r) = 0$$

where $\psi(r_i) = \phi_\delta'(r_i)/r_i$ is a weight. In terms of x, we have $\delta r = A\delta x$, and so the stationary conditions are

$$\delta x^T A^T W(r)(Ax - b) = 0$$

where $W(r)$ is a diagonal matrix with entries $W_{ii}(r) = \psi(r_i)$. That is, the stationary conditions correspond to a set of normal equations for a weighted least squares problem! Unfortunately, we don't know what the weights are; but we can guess them based on previous iterates. That is, we repeatedly solve problems of the form

$$\text{minimize } \|Ax^{(k+1)} - b\|^2_{W(r^{(k)})}$$

until convergence.