CS 6210: Matrix Computations

Rank deficiency and regularization

David Bindel

2025-10-06

Bias-variance decomposition

The *bias-variance decomposition* is a standard result in statistics. Suppose we want to predict

$$y = f(x) + \epsilon$$

where $f(x)$ is some underlying "ground truth" function and ϵ is a noise term with mean zero and variance σ^2. Some algorithm takes in a data set D that encompasses our observations about the function (usually $D = \{(x_i, y_i)\}_{i=1}^n$) and produces a prediction function $s_D(x)$. The data set (and the algorithm) may include some randomness, which we assume is independent of the measurement noise ϵ at any test point. So we also define the mean and variance for the prediction function:

$$\bar{s}_D(x) = \mathbb{E}_D[s_D(x)], \quad \mathrm{Var}_D[s_D(x)] = \mathbb{E}_D[(s_D(x) - \bar{s}_D(x))^2].$$

The *bias* in the predictor is the difference between its mean and the true function $f(x)$:

$$\mathrm{Bias}_D[s_D(x)] = f(x) - \bar{s}_D(x).$$

The bias-variance decomposition says that the squared prediction error at a test point s can be written

$$\mathbb{E}_{D,\epsilon}[(y - s_D(x))^2] = \mathrm{Var}_D[s_D(x)] + \mathrm{Bias}_D[s_D(x)]^2 + \sigma^2.$$

The argument consists of two steps. First, we expand the quadratic and use independence of ϵ from D to get

$$\begin{aligned}
\mathbb{E}_{D,\epsilon}[(y - s_D(x))^2] &= \mathbb{E}_D[(f(x) - s_D(x))^2] + 2\mathbb{E}_{D,\epsilon}[\epsilon(f(x) - s_D(x))] + \mathbb{E}_\epsilon[\epsilon^2] \\
&= \mathbb{E}_D[(f(x) - s_D(x))^2] + 2\mathbb{E}_\epsilon[\epsilon]\mathbb{E}_D[(f(x) - s_D(x))] + \mathbb{E}_\epsilon[\epsilon^2] \\
&= \mathbb{E}_D[(f(x) - s_D(x))^2] + \sigma^2
\end{aligned}$$

since $\mathbb{E}_\epsilon[\epsilon] = 0$ and $\mathbb{E}_\epsilon[\epsilon^2] = \sigma^2$. Now write

$$f(x) - s_D(x) = b(x) - e_D(x)$$

where

$$b(x) := f(x) - \bar{s}_D(x)$$
$$e_D(x) := s_D(x) - \bar{s}_D(x).$$

Note that $b(x)$ the bias and that $\mathbb{E}_D[e_D(x)] = 0$ and $\mathbb{E}_D[e_D(x)^2] = \mathrm{Var}_D[s_D(x)]$, so that

$$\begin{aligned}\mathbb{E}_D[(b(x) - e_D(x))^2] &= b(x)^2 - 2b(x)\mathbb{E}_D(x) + \mathbb{E}_D[e_D(x)^2] \\ &= b(x)^2 + \mathrm{Var}_D[s_D(x)^2] \\ &= \mathrm{Bias}_D[s_D(x)]^2 + \mathrm{Var}_D[s_D(x)^2].\end{aligned}$$

Bias-variance tradeoffs in the matrix setting

Now consider linear least squares in the context of the bias-variance tradeoff. For simplicity, rather than a function $f(x)$, let us assume that our ground truth is a vector of measurements $f \in \mathbb{R}^M$. At test time, we are trying to predict $y = f + e$, where e is a vector of mean zero errors. Our model space will be $\{Ax : x \in \mathbb{R}^n\}$, and our fitting algorithm will involve

$$\hat{x} = A_1^\dagger y_1, \quad y_1 = f_1 + \tilde{e}_1$$

where $A_1 \in \mathbb{R}^{m \times n}$ is a subset of the rows of A and f_1 is the corresponding subset of rows of f, and $\tilde{e}_1$ is a vector of mean zero training-time errors. By linearity of expectation, the mean model is

$$\bar{x} = \mathbb{E}[\hat{x}] = A_1^\dagger f_1.$$

and

$$\hat{x} - \bar{x} = A_1^\dagger \tilde{e}_1$$

Then the bias-variance decomposition gives us

$$\mathbb{E}[\|y - A\hat{x}\|^2] = \|f - A\bar{x}\|^2 + \mathbb{E}[\|AA_1^\dagger \tilde{e}_1\|^2] + \mathbb{E}[\|e\|^2].$$

This is all as in the previous section, just with a concrete choice of algorithms for fitting the model.

Now suppose that $x_* = A^\dagger f$ and $r = f - Ax_*$. The model associated with x_* is optimal: the squared bias term $\|r\|^2$ is as small as possible, and if a genie gives us x_*, there is no variance! How does the model associated with $\hat{x}$ compare? We can rewrite the squared bias term for the $\hat{x}$ model as

$$\|f - A\bar{x}\|^2 = \|r - A(\bar{x} - x_*)\|^2 = \|r\|^2 + \|A(\bar{x} - x_*)\|^2.$$

Note that $x_* = A_1^\dagger(b_1 + r_1)$, so $\bar{x} - x_* = -A_1^\dagger r_1$. Therefore

$$\|f - A\bar{x}\|^2 = \|r\|^2 + \|AA_1^\dagger r_1\|^2.$$

Putting things together, we have the decomposition

$$\mathbb{E}[\|y - A\hat{x}\|^2] = \left(\|r\|^2 + \|AA_1^\dagger r_1\|^2\right) + \left(\mathbb{E}[\|AA_1^\dagger \tilde{e}_1\|^2] + \mathbb{E}[\|e\|^2]\right),$$

where the first two terms come from the bias and the second two terms come from the variance of the training error $\tilde{e}_1$ and the test error e.

Note that if e and $\tilde{e}$ have independent entries with mean σ^2, then

$$\begin{aligned}\mathbb{E}[\|AA_1^\dagger \tilde{e}_1\|^2] &= \sigma^2 \|AA_1^\dagger\|_F^2 \leq \sigma^2 n \|AA_1^\dagger\|_2^2 \\ \mathbb{E}[\|e\|^2] &= \sigma^2 M\end{aligned}$$

In this case, we have

$$\mathbb{E}[\|y - A\hat{x}\|^2] \leq \left(1 + \|AA_1^\dagger\|^2 \frac{\|r_1\|^2}{\|r\|^2}\right)\|r\|^2 + \left(1 + \|AA_1^\dagger\|^2 \frac{n}{M}\right) M\sigma^2$$

We note that $\|r_1\|^2/\|r\|^2$ should be approximately n/M if the residual entries in r_1 are "typical" of those in r. We can be cruder in our bounds and say

$$\mathbb{E}[\|y - A\hat{x}\|^2] \leq \left(1 + \|AA_1^\dagger\|^2\right)\left(\|r\|^2 + M\sigma^2\right).$$

This can be interpreted as a *quasi-optimality* result: the expected squared prediction error is within a constant factor $(1 + \|AA_1^\dagger\|^2)$ of the best possible error given the model flexibility and the noise.

When $\|A_1^\dagger\|$ is large, the problem of fitting to training data is ill-posed, and the accuracy can be compromised. What can we do? As we discussed in the last section, the problem with ill-posed problems is that they admit many solutions of very similar quality. In order to distinguish between these possible solutions to find a model with good predictive power, we consider *regularization*: that is, we assume that the coefficient vector x is not too large in norm, or that it is sparse. Different statistical assumptions give rise to different regularization strategies; for the current discussion, we shall focus on the computational properties of a few of the more common regularization strategies without going into the details of the statistical assumptions. In particular, we consider four strategies in turn

1. *Factor selection* via *pivoted QR*.
2. *Tikhonov regularization* and its solution.
3. *Truncated SVD regularization*.
4. ℓ^1 *regularization* or the *lasso*.

Factor selection and pivoted QR

In ill-conditioned problems, the columns of A are nearly linearly dependent; we can effectively predict some columns as linear combinations of other columns. The goal of the column pivoted QR algorithm is to find a set of columns that are "as linearly independent as possible." This is not such a simple task, and so we settle for a greedy strategy: at each step, we select the column that is least well predicted (in the sense of residual norm) by columns already selected. This leads to the *pivoted QR factorization*

$$A\Pi = QR$$

where Π is a permutation and the diagonal entries of R appear in descending order (i.e. $r_{11} \geq r_{22} \geq ...$). To decide on how many factors to keep in the factorization, we either automatically take the first k or we dynamically choose to take k factors where r_{kk} is greater than some tolerance and $r_{k+1,k+1}$ is not.

The pivoted QR approach has a few advantages. It yields *parsimonious* models that predict from a subset of the columns of A – that is, we need to measure fewer than n factors to produce an entry of b in a new column. It can also be computed relatively cheaply, even for large matrices that may be sparse. However, pivoted QR is not the only approach! A related approach due to Golub, Klema, and Stewart computes $A = U\Sigma V^T$ and chooses a subset of the factors based on pivoted QR of V^T. More generally, approaches such as the lasso yield an automatic factor selection.

Tikhonov regularization (ridge regression)

Another approach is to say that we want a model in which the coefficients are not too large. To accomplish this, we add a penalty term to the usual least squares problem:

$$\text{minimize } \|Ax - b\|^2 + \lambda^2 \|x\|^2.$$

Equivalently, we can write

$$\text{minimize } \left\| \begin{bmatrix} A \\ \lambda I \end{bmatrix} x - \begin{bmatrix} b \\ 0 \end{bmatrix} \right\|^2,$$

which leads to the regularized version of the normal equations

$$(A^T A + \lambda^2 I)x = A^T b.$$

In some cases, we may want to regularize with a more general norm $\|x\|_M^2 = x^T M x$ where M is symmetric and positive definite, which leads to the regularized equations

$$(A^T A + \lambda^2 M)x = A^T b.$$

If we want to incorporate prior information that pushes x toward some initial guess x_0, we may pose the least squares problem in terms of $z = x - x_0$ and use some form of Tikhonov regularization. If we know of no particular problem structure in advance, the standard choice of $M = I$ is a good default.

It is useful to compare the usual least squares solution to the regularized solution via the SVD. If $A = U\Sigma V^T$ is the economy SVD, then

$$x_{LS} = V\Sigma^{-1}U^T b$$
$$x_{Tik} = Vf(\Sigma)^{-1}U^T b$$

where

$$f(\sigma) = \frac{1}{\sqrt{\sigma^{-1} + \lambda^2}}.$$

This *filter* of the inverse singular values affects the larger singular values only slightly, but damps the effect of very small singular values.

Truncated SVD

The Tikhonov filter reduces the effect of small singular values on the solution, but it does not eliminate that effect. By contrast, the *truncated SVD* approach uses the filter

$$f(z) = \begin{cases} z, & z > \sigma_{\min} \\ \infty, & \text{otherwise.} \end{cases}$$

In other words, in the truncated SVD approach, we use

$$x = V_k \Sigma_k^{-1} U_k^T b$$

where U_k and V_k represent the leading k columns of U and V, respectively, while Σ_k is the diagonal matrix consisting of the k largest singular values.

ℓ^1 and the lasso

An alternative to Tikhonov regularization (based on a Euclidean norm of the coefficient vector) is an ℓ^1 regularized problem

$$\text{minimize } \|Ax - b\|^2 + \lambda \|x\|_1.$$

This is sometimes known as the "lasso" approach. The ℓ^1 regularized problem has the property that the solutions tend to become sparse as λ becomes larger. That is, the ℓ^1 regularization effectively imposes a factor selection process like that we saw in the pivoted QR approach.

Unlike the pivoted QR approach, however, the ℓ^1 regularized solution cannot be computed by one of the standard factorizations of numerical linear algebra. Instead, one treats it as a more general *convex optimization* problem. We will discuss some approaches to the solution of such problems later in the semester.