Stable Reconstruction with the Truncated SVD

We have seen that if $\boldsymbol{A}$ has very small singular values and we apply the pseudo-inverse in the presence of noise, the results can be disastrous. But it doesn't have to be this way. There are several ways to stabilize the pseudo-inverse. We start be discussing the simplest one, where we simply "cut out" the part of the reconstruction which is causing the problems.

As before, we are given noisy indirect observations of a vector $\boldsymbol{x}$ through a $M \times N$ matrix $\boldsymbol{A}$:

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{e}. \tag{1}$$

The matrix $\boldsymbol{A}$ has SVD $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}}$, and pseudo-inverse $\boldsymbol{A}^{\dagger} = \boldsymbol{V}\boldsymbol{\Sigma}^{-1}\boldsymbol{U}^{\mathrm{T}}$. We can rewrite $\boldsymbol{A}$ as a sum of rank-1 matrices:

$$\boldsymbol{A} = \sum_{r=1}^{R} \sigma_r \boldsymbol{u}_r \boldsymbol{v}_r^{\mathrm{T}},$$

where R is the rank of $\boldsymbol{A}$, the σ_r are the singular values, and $\boldsymbol{u}_r \in \mathbb{R}^M$ and $\boldsymbol{v}_r \in \mathbb{R}^N$ are columns of $\boldsymbol{U}$ and $\boldsymbol{V}$, respectively. Similarly, we can write the pseudo-inverse as

$$\boldsymbol{A}^{\dagger} = \sum_{r=1}^{R} \frac{1}{\sigma_r} \boldsymbol{v}_r \boldsymbol{u}_r^{\mathrm{T}}.$$

Given $\boldsymbol{y}$ as above, we can write the least-squares estimate of $\boldsymbol{x}$ from the noisy measurements as

$$\hat{\boldsymbol{x}}_{\mathrm{ls}} = \boldsymbol{A}^{\dagger}\boldsymbol{y} = \sum_{r=1}^{R} \frac{1}{\sigma_r} \langle \boldsymbol{y}, \boldsymbol{u}_r \rangle \boldsymbol{v}_r. \tag{2}$$

As we can see (and have seen before) if any one of the σ_r are very small, the least-squares reconstruction can be a disaster.

A simple way to avoid this is to simply truncate the sum (2), leaving out the terms where σ_r is too small ($1/\sigma_r$ is too big). Exactly how many terms to keep depends a great deal on the application, as there are competing interests. On the one hand, we want to ensure that each of the σ_r we include has an inverse of reasonable size, on the other, we want the reconstruction to be accurate (i.e. does not deviate from the noiseless least-squares solution by too much).

We form an approximation $\boldsymbol{A}'$ to $\boldsymbol{A}$ by taking

$$\boldsymbol{A}' = \sum_{r=1}^{R'} \sigma_r \boldsymbol{u}_r \boldsymbol{v}_r^{\mathrm{T}},$$

for some $R' < R$. Again, our final answer will depend on which R' we use, and choosing R' is often times something of an art. It is clear that the approximation $\boldsymbol{A}'$ has rank R'. Note that the pseudo-inverse of $\boldsymbol{A}'$ is also a truncated sum

$$\boldsymbol{A}'^{\dagger} = \sum_{r=1}^{R'} \frac{1}{\sigma_r} \boldsymbol{v}_r \boldsymbol{u}_r^{\mathrm{T}}.$$

Given noisy data $\boldsymbol{y}$ as in (1), we reconstruct $\boldsymbol{x}$ by applying the truncated pseudo-inverse to $\boldsymbol{y}$:

$$\hat{\boldsymbol{x}}_{\text{trunc}} = \boldsymbol{A}'^{\dagger}\boldsymbol{y} = \sum_{r=1}^{R'} \frac{1}{\sigma_r} \langle \boldsymbol{y}, \boldsymbol{u}_r \rangle \boldsymbol{v}_r.$$

How good is this reconstruction? To answer this question, we will compare it to the noiseless least-squares reconstruction $\boldsymbol{x}_{\text{pinv}} = \boldsymbol{A}^\dagger \boldsymbol{y}_{\text{clean}}$, where $\boldsymbol{y}_{\text{clean}} = \boldsymbol{A}\boldsymbol{x}$ are "noiseless" measurements of $\boldsymbol{x}$. The difference between these two is the reconstruction error (relative to $\boldsymbol{x}_{\text{pinv}}$) as

$$\begin{aligned}\hat{\boldsymbol{x}}_{\text{trunc}} - \boldsymbol{x}_{\text{pinv}} &= \boldsymbol{A}'^\dagger \boldsymbol{y} - \boldsymbol{A}^\dagger \boldsymbol{A}\boldsymbol{x} \\ &= \boldsymbol{A}'^\dagger \boldsymbol{A}\boldsymbol{x} + \boldsymbol{A}'^\dagger \boldsymbol{e} - \boldsymbol{A}^\dagger \boldsymbol{A}\boldsymbol{x} \\ &= (\boldsymbol{A}'^\dagger - \boldsymbol{A}^\dagger)\boldsymbol{A}\boldsymbol{x} + \boldsymbol{A}'^\dagger \boldsymbol{e}.\end{aligned}$$

Proceeding further, we can write the matrix $\boldsymbol{A}'^\dagger - \boldsymbol{A}^\dagger$ as

$$\boldsymbol{A}'^\dagger - \boldsymbol{A}^\dagger = \sum_{r=R'+1}^{R} -\frac{1}{\sigma_r} \boldsymbol{v}_r \boldsymbol{u}_r^{\mathrm{T}},$$

and so the first term in the reconstruction error can be written as

$$\begin{aligned}(\boldsymbol{A}'^\dagger - \boldsymbol{A}^\dagger)\boldsymbol{A}\boldsymbol{x} &= \sum_{r=R'+1}^{R} -\frac{1}{\sigma_r} \langle \boldsymbol{A}\boldsymbol{x}, \boldsymbol{u}_r \rangle \boldsymbol{v}_r \\ &= \sum_{r=R'+1}^{R} -\frac{1}{\sigma_r} \left\langle \sum_{j=1}^{R} \sigma_j \langle \boldsymbol{x}, \boldsymbol{v}_j \rangle \boldsymbol{u}_j, \boldsymbol{u}_r \right\rangle \boldsymbol{v}_k \\ &= \sum_{r=R'+1}^{R} -\frac{1}{\sigma_r} \sum_{j=1}^{R} \sigma_j \langle \boldsymbol{x}, \boldsymbol{v}_j \rangle \langle \boldsymbol{u}_j, \boldsymbol{u}_r \rangle \boldsymbol{v}_r \\ &= \sum_{r=R'+1}^{R} -\langle \boldsymbol{x}, \boldsymbol{v}_r \rangle \boldsymbol{v}_r \quad (\text{since } \langle \boldsymbol{u}_r, \boldsymbol{u}_j \rangle = 0 \text{ unless } j = r).\end{aligned}$$

The second term in the reconstruction error can also be expanded against the $\boldsymbol{v}_r$:

$$\boldsymbol{A}'^\dagger \boldsymbol{e} = \sum_{r=1}^{R'} \frac{1}{\sigma_r} \langle \boldsymbol{e}, \boldsymbol{u}_r \rangle \boldsymbol{v}_r.$$

Combining these expressions, the reconstruction error can be written

$$\begin{aligned}\hat{\boldsymbol{x}}_{\text{trunc}} - \boldsymbol{x}_{\text{pinv}} &= \underbrace{\sum_{r=1}^{R'} \frac{1}{\sigma_r} \langle \boldsymbol{e}, \boldsymbol{u}_r \rangle \boldsymbol{v}_r}_{} + \underbrace{\sum_{k=R'+1}^{R} -\langle \boldsymbol{x}, \boldsymbol{v}_k \rangle \boldsymbol{v}_k}_{} \\ &= \text{Noise error} + \text{Approximation error.}\end{aligned}$$

Since the $\boldsymbol{v}_r$ are mutually orthogonal, and the two sums run over disjoint index sets, the noise error and the approximation error will be orthogonal. Also

$$\begin{aligned}\|\hat{\boldsymbol{x}}_{\text{trunc}} - \boldsymbol{x}_{\text{pinv}}\|_2^2 &= \|\text{Noise error}\|_2^2 + \|\text{Approximation error}\|_2^2 \\ &= \sum_{r=1}^{R'} \frac{1}{\sigma_r^2} |\langle \boldsymbol{e}, \boldsymbol{u}_r \rangle|^2 + \sum_{r=R'+1}^{R} |\langle \boldsymbol{x}, \boldsymbol{v}_r \rangle|^2.\end{aligned}$$

The reconstruction error, then, is signal dependent and will depend on how much of the vector $\boldsymbol{x}$ is concentrated in the subspace spanned by $\boldsymbol{v}_{R'+1}, \ldots, \boldsymbol{v}_R$. We will lose everything in this subspace; if it contains a significant part of $\boldsymbol{x}$, then there is not much least-squares can do for you.

The worst-case noise error occurs when $\boldsymbol{e}$ is completely aligned with $\boldsymbol{u}_{R'}$:

$$\|\text{Noise error}\|_2^2 = \sum_{r=1}^{R'} \frac{1}{\sigma_r^2} |\langle \boldsymbol{e}, \boldsymbol{u}_r \rangle|^2 \leq \frac{1}{\sigma_{R'}^2} \cdot \|\boldsymbol{e}\|_2^2.$$

If the error $\boldsymbol{e}$ is random, this bound is a bit pessimistic. Suppose that each entry of $\boldsymbol{e}$ is an independent identically distributed random variable

$$e[m] \sim \text{Normal}(0, \nu^2).$$

The expected size of the error is then

$$\mathrm{E}[\|\boldsymbol{e}\|_2^2] = M\nu^2,$$

and the noise error in the reconstruction will be

$$\begin{aligned}
\mathrm{E}[\|\text{Noise error}\|_2^2] &= \sum_{r=1}^{R'} \frac{1}{\sigma_r^2}\ \mathrm{E}[|\langle \boldsymbol{e}, \boldsymbol{u}_r\rangle|^2] \\
&= \nu^2 \left(\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2} + \cdots + \frac{1}{\sigma_{R'}^2}\right) \\
&= \frac{1}{M}\left(\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2} + \cdots + \frac{1}{\sigma_{R'}^2}\right) \cdot \mathrm{E}[\|\boldsymbol{e}\|_2^2].
\end{aligned}$$

So when the error is random, the noise error in the reconstruction depends on the *average* of the $1/\sigma_r^2$ (with zeros in place of the truncated eigenvalues), rather than the largest one.

Stable Reconstruction using Tikhonov Regularization

Tikhonov[1] regularization is another way to stabilize the least-squares recovery. It has the nice features that: 1) it can be interpreted using optimization, and 2) it can be computed without direct knowledge of the SVD of $\boldsymbol{A}$.

Recall that we motivated the pseudo-inverse by showing that $\hat{\boldsymbol{x}}_{LS} = \boldsymbol{A}^\dagger \boldsymbol{y}$ is a solution to

$$\underset{\boldsymbol{x}\in\mathbb{R}^N}{\text{minimize}} \;\; \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2. \tag{3}$$

When $\boldsymbol{A}$ has full column rank, $\hat{\boldsymbol{x}}_{LS}$ is the unique solution, otherwise it is the solution with smallest energy. When $\boldsymbol{A}$ has full column rank but has singular values which are very small, huge variations in $\boldsymbol{x}$ (in directions of the singular vectors $\boldsymbol{v}_k$ corresponding to the tiny σ_k) can have very little effect on the residual $\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2$. As such, the solution to (3) can have wildly inaccurate components in the presence of even mild noise.

One way to counteract this problem is to modify (3) with a **regularization** term that penalizes the size of the solution $\|\boldsymbol{x}\|_2^2$ as well as the residual error $\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2$:

$$\underset{\boldsymbol{x}\in\mathbb{R}^N}{\text{minimize}} \;\; \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \delta\|\boldsymbol{x}\|_2^2. \tag{4}$$

The parameter $\delta > 0$ gives us a trade-off between accuracy and regularization; we want to choose δ small enough so that the residual

[1] Andrey Tikhonov (1906-1993) was a 20th century Russian mathematician.

for the solution of (4) is close to that of (3), and large enough so that the problem is well-conditioned.

Just as with (3), which is solved by applying the pseudo-inverse to $\boldsymbol{y}$, we can write the solution to (4) in closed form. To see this, recall that we can decompose any $\boldsymbol{x} \in \mathbb{R}^N$ as

$$\boldsymbol{x} = \boldsymbol{V}\boldsymbol{\alpha} + \boldsymbol{V}_0\boldsymbol{\alpha}_0,$$

where $\boldsymbol{V}$ is the $N \times R$ matrix (with orthonormal columns) used in the SVD of $\boldsymbol{A}$, and $\boldsymbol{V}_0$ is a $N \times N - R$ matrix whose columns are an orthogonal basis for the null space of $\boldsymbol{A}$. This means that the columns of $\boldsymbol{V}_0$ are orthogonal to each other and all of the columns of $\boldsymbol{V}$. Similarly, we can decompose $\boldsymbol{y}$ as

$$\boldsymbol{y} = \boldsymbol{U}\boldsymbol{\beta} + \boldsymbol{U}_0\boldsymbol{\beta}_0,$$

where $\boldsymbol{U}$ is the $M \times R$ matrix used in the SVD of $\boldsymbol{A}$, and the columns of $\boldsymbol{U}_0$ are an orthogonal basis for the left null space of $\boldsymbol{A}$ (everything in $\mathbb{R}^M$ that is not in the range of $\boldsymbol{A}$).

For any $\boldsymbol{x}$, we can write

$$\begin{aligned}\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x} &= \boldsymbol{U}\boldsymbol{\beta} + \boldsymbol{U}_0\boldsymbol{\beta} - \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}}(\boldsymbol{V}\boldsymbol{\alpha} + \boldsymbol{V}_0\boldsymbol{\alpha}_0) \\ &= \boldsymbol{U}(\boldsymbol{\beta} - \boldsymbol{\Sigma}\boldsymbol{\alpha}) + \boldsymbol{U}_0\boldsymbol{\beta}.\end{aligned}$$

Since the columns of $\boldsymbol{U}$ are orthonormal, $\boldsymbol{U}^{\mathrm{T}}\boldsymbol{U} = \mathbf{I}$, and also $\boldsymbol{U}_0^{\mathrm{T}}\boldsymbol{U}_0 = \mathbf{I}$, and $\boldsymbol{U}^{\mathrm{T}}\boldsymbol{U}_0 = \mathbf{0}$, we have

$$\begin{aligned}\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 &= \langle \boldsymbol{U}(\boldsymbol{\beta} - \boldsymbol{\Sigma}\boldsymbol{\alpha}) + \boldsymbol{U}_0\boldsymbol{\beta}_0, \boldsymbol{U}(\boldsymbol{\beta} - \boldsymbol{\Sigma}\boldsymbol{\alpha}) + \boldsymbol{U}_0\boldsymbol{\beta}_0 \rangle \\ &= \|\boldsymbol{\beta} - \boldsymbol{\Sigma}\boldsymbol{\alpha}\|_2^2 + \|\boldsymbol{\beta}_0\|_2^2,\end{aligned}$$

and

$$\|\boldsymbol{x}\|_2^2 = \|\boldsymbol{\alpha}\|_2^2 + \|\boldsymbol{\alpha}_0\|_2^2.$$

Using these facts, we can write the functional in (4) as

$$\|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\|_2^2+\delta\|\boldsymbol{x}\|^2=\|\boldsymbol{\beta}-\boldsymbol{\Sigma}\boldsymbol{\alpha}\|_2^2+\delta\|\boldsymbol{\alpha}\|_2^2+\delta\|\boldsymbol{\alpha}_0\|_2^2. \tag{5}$$

We want to choose $\boldsymbol{\alpha}$ and $\boldsymbol{\alpha}_0$ that minimize (5). It is clear that, just as in the standard least-squares problem, we need $\boldsymbol{\alpha}_0=\mathbf{0}$. The part of the functional that depends on $\boldsymbol{\alpha}$ can be rewritten as

$$\|\boldsymbol{\beta}-\boldsymbol{\Sigma}\boldsymbol{\alpha}\|_2^2+\delta\|\boldsymbol{\alpha}\|_2^2=\sum_{k=1}^{R}(\beta[k]-\sigma_k\alpha[k])^2+\delta\alpha[k]^2. \tag{6}$$

We can minimize this sum simply by minimizing each term independently. Since

$$\frac{d}{d\alpha[k]}\left[(\beta[k]-\sigma_k\alpha[k])^2+\delta\alpha[k]^2\right]=-2\beta[k]\sigma_k+2\sigma_k^2\alpha[k]+2\delta\alpha[k],$$

we need

$$\alpha[k]=\frac{\sigma_k}{\sigma_k^2+\delta}\beta[k].$$

Putting this back in vector form, (6) is minimized by

$$\hat{\boldsymbol{\alpha}}_{\text{tik}}=(\boldsymbol{\Sigma}^2+\delta\mathbf{I})^{-1}\boldsymbol{\Sigma}\boldsymbol{\beta},$$

and so the minimizer to (4) is

$$\begin{aligned}\hat{\boldsymbol{x}}_{\text{tik}}&=\boldsymbol{V}\hat{\boldsymbol{\alpha}}_{\text{tik}}\\&=\boldsymbol{V}(\boldsymbol{\Sigma}^2+\delta\mathbf{I})^{-1}\boldsymbol{\Sigma}\boldsymbol{U}^{\mathrm{T}}\boldsymbol{y}.\end{aligned} \tag{7}$$

We can get a better feel for what Tikhonov regularization is doing by comparing it directly to the pseudo-inverse. The least-squares reconstruction $\hat{\boldsymbol{x}}_{\text{ls}}$ can be written as

$$\begin{aligned}\hat{\boldsymbol{x}}_{\text{ls}} &= \boldsymbol{V}\boldsymbol{\Sigma}^{-1}\boldsymbol{U}^{\mathrm{T}}\boldsymbol{y} \\ &= \sum_{r=1}^{R} \frac{1}{\sigma_r}\langle \boldsymbol{y}, \boldsymbol{u}_r\rangle \boldsymbol{v}_r,\end{aligned}$$

while the Tikhonov reconstruction $\hat{\boldsymbol{x}}_{\text{tik}}$ derived above is

$$\hat{\boldsymbol{x}}_{\text{tik}} = \sum_{r=1}^{R} \frac{\sigma_r}{\sigma_r^2 + \delta}\langle \boldsymbol{y}, \boldsymbol{u}_r\rangle \boldsymbol{v}_r. \tag{8}$$

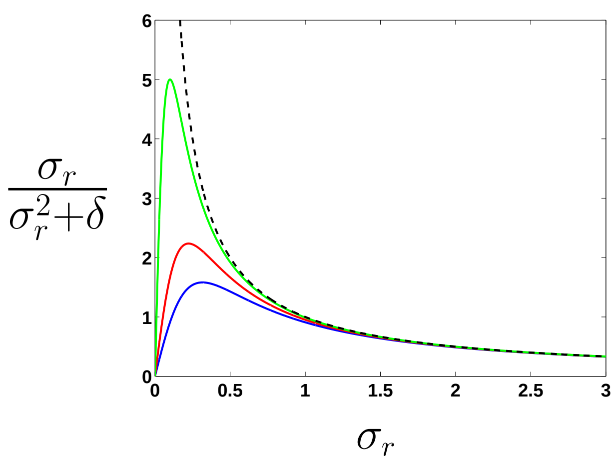
Notice that when σ_r is much larger than δ,

$$\frac{\sigma_r}{\sigma_r^2 + \delta} \approx \frac{1}{\sigma_r}, \quad \sigma_r \gg \delta,$$

but when σ_r is small

$$\frac{\sigma_r}{\sigma_r^2 + \delta} \approx 0, \quad \sigma_r \ll \delta.$$

Thus the Tikhonov reconstruction modifies the important parts (components where the σ_r are large) of the pseudo-inverse very little, while ensuring that the unimportant parts (components where the σ_r are small) affect the solution only by a very small amount. This **damping** of the singular values, is illustrated below.



Above, we see the damped multipliers $\sigma_r/(\sigma_r^2 + \delta)$ versus σ_r for $\delta = 0.1$ (blue), $\delta = 0.05$ (red), and $\delta = 0.01$ (green). The black dotted line is $1/\sigma_r$, the least-squares multiplier. Notice that for large σ_r ($\sigma_r > 2\sqrt{\delta}$, say), the damping has almost no effect.

This damping makes the Tikhonov reconstruction exceptionally stable; large multipliers never appear in the reconstruction (8). In fact it is easy to check that

$$\frac{\sigma_r}{\sigma_r^2 + \delta} \leq \frac{1}{2\sqrt{\delta}}$$

no matter the value of σ_r.

Tikhonov Error Analysis

Given noisy observations $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}_0 + \boldsymbol{e}$, how well will Tikhonov regularization work? The answer to this questions depends on multiple factors including the choice of δ, the nature of the perturbation $\boldsymbol{e}$, and how well $\boldsymbol{x}_0$ can be approximated using a linear combination of the singular vectors $\boldsymbol{v}_r$ corresponding to the large (relative to δ) singular values. Since a closed-form expression for the solution to (4) exists, we can quantify these trade-offs precisely.

We compare the Tikhonov reconstruction to the reconstruction we would obtain if we used standard least-squares on perfectly noise-free observations $\boldsymbol{y}_{\text{clean}} = \boldsymbol{A}\boldsymbol{x}_0$. This noise-free reconstruction can be written as

$$
\begin{aligned}
\boldsymbol{x}_{\mathrm{pinv}} = \boldsymbol{A}^{\dagger}\boldsymbol{y}_{\mathrm{clean}} &= \boldsymbol{A}^{\dagger}\boldsymbol{A}\boldsymbol{x}_0 \\
&= \boldsymbol{V}\boldsymbol{\Sigma}^{-1}\boldsymbol{U}^{\mathrm{T}}\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}}\boldsymbol{x}_0 \\
&= \boldsymbol{V}\boldsymbol{V}^{\mathrm{T}}\boldsymbol{x}_0 \\
&= \sum_{r=1}^{R} \langle \boldsymbol{x}_0, \boldsymbol{v}_r \rangle \boldsymbol{v}_r.
\end{aligned}
$$

The vector $\boldsymbol{x}_{\mathrm{pinv}}$ is the orthogonal projection of $\boldsymbol{x}_0$ onto the row space (everything orthogonal to the null space) of $\boldsymbol{A}$. If $\boldsymbol{A}$ has full column rank, then $\boldsymbol{x}_{\mathrm{pinv}} = \boldsymbol{x}_0$. If not, then the application of $\boldsymbol{A}$ destroys the part of $\boldsymbol{x}_0$ that is not in $\boldsymbol{x}_{\mathrm{pinv}}$, and so we only attempt to recover the "visible" components. In some sense, $\boldsymbol{x}_{\mathrm{pinv}}$ contains all of the components of $\boldsymbol{x}_0$ that we could ever hope to recover, and has them preserved perfectly.

The Tikhonov regularized solution is given by

$$
\begin{aligned}
\hat{\boldsymbol{x}}_{\mathrm{tik}} &= \sum_{r=1}^{R} \frac{\sigma_r}{\sigma_r^2 + \delta} \langle \boldsymbol{y}, \boldsymbol{u}_r \rangle \boldsymbol{v}_r \\
&= \sum_{r=1}^{R} \frac{\sigma_r}{\sigma_r^2 + \delta} \langle \boldsymbol{e}, \boldsymbol{u}_r \rangle \boldsymbol{v}_r \;+\; \sum_{r=1}^{R} \frac{\sigma_r}{\sigma_r^2 + \delta} \langle \boldsymbol{A}\boldsymbol{x}_0, \boldsymbol{u}_r \rangle \boldsymbol{v}_r \\
&= \sum_{r=1}^{R} \frac{\sigma_r}{\sigma_r^2 + \delta} \langle \boldsymbol{e}, \boldsymbol{u}_r \rangle \boldsymbol{v}_r \;+\; \sum_{r=1}^{R} \frac{\sigma_r^2}{\sigma_r^2 + \delta} \langle \boldsymbol{x}_0, \boldsymbol{v}_r \rangle \boldsymbol{v}_r,
\end{aligned}
$$

and so the reconstruction error, relative to the best possible reconstruction $\boldsymbol{x}_{\text{pinv}}$, is

$$\begin{aligned}
\hat{\boldsymbol{x}}_{\text{tik}} - \boldsymbol{x}_{\text{pinv}} &= \sum_{r=1}^{R} \frac{\sigma_r}{\sigma_r^2 + \delta} \langle \boldsymbol{e}, \boldsymbol{u}_r \rangle \boldsymbol{v}_r \;+\; \sum_{r=1}^{R} \left(\frac{\sigma_r^2}{\sigma_r^2 + \delta} - 1 \right) \langle \boldsymbol{x}_0, \boldsymbol{v}_r \rangle \boldsymbol{v}_r \\
&= \underbrace{\sum_{r=1}^{R} \frac{\sigma_r}{\sigma_r^2 + \delta} \langle \boldsymbol{e}, \boldsymbol{u}_r \rangle \boldsymbol{v}_r} \;+\; \underbrace{\sum_{r=1}^{R} \frac{-\delta}{\sigma_r^2 + \delta} \langle \boldsymbol{x}_0, \boldsymbol{v}_r \rangle \boldsymbol{v}_r} \\
&= \qquad \text{Noise error} \qquad + \qquad \text{Approximation error.}
\end{aligned}$$

The approximation error is signal dependent, and depends on δ. Since the $\boldsymbol{v}_r$ are orthonormal,

$$\|\text{Approximation error}\|_2^2 \;=\; \sum_{r=1}^{R} \frac{\delta^2}{(\sigma_r^2 + \delta)^2} |\langle \boldsymbol{x}_0, \boldsymbol{v}_r \rangle|^2.$$

Note that for the components much smaller than δ,

$$\sigma_r^2 \ll \delta \quad \Rightarrow \quad \frac{\delta^2}{(\sigma_r^2 + \delta)^2} \approx 1,$$

so this portion of the approximation error will be about the same as if we had simply truncated these components.

For large components,

$$\sigma_r^2 \gg \delta \quad \Rightarrow \quad \frac{\delta^2}{(\sigma_r^2 + \delta)^2} \approx \frac{\delta^2}{\sigma_r^2}$$

and so this portion of the approximation error will be very small.

For the noise error energy, we have

$$\begin{aligned}\|\text{Noise error}\|_2^2 &= \sum_{r=1}^{R}\left(\frac{\sigma_r}{\sigma_r^2+\delta}\right)^2 |\langle \boldsymbol{e},\boldsymbol{u}_r\rangle|^2 \\ &\le \frac{1}{4\delta}\sum_{r=1}^{R}|\langle \boldsymbol{e},\boldsymbol{u}_r\rangle|^2 \\ &\le \frac{1}{4\delta}\|\boldsymbol{e}\|_2^2.\end{aligned}$$

The worst-case error is more or less determined by the choice of δ. The regularization makes the effective condition number of $\boldsymbol{A}$ about $1/(2\sqrt{\delta})$; no matter how small the smallest singular value is, the noise energy will not increase by more than a factor of $1/(4\delta)$ during the reconstruction process.

Less pessimistic is the average case error. Suppose that the entries of the error vector $\boldsymbol{e}$ are iid Gaussian random variables

$$e[m] \sim \text{Normal}(0,\nu^2).$$

Then the expected value of the noise error reconstruction will be

$$\begin{aligned}\mathrm{E}\left[\|\text{Noise error}\|_2^2\right] &= \sum_{r=1}^{R}\left(\frac{\sigma_r}{\sigma_r^2+\delta}\right)^2 \mathrm{E}\left[|\langle \boldsymbol{e},\boldsymbol{u}_r\rangle|^2\right] \\ &= \nu^2\cdot\sum_{r=1}^{R}\left(\frac{\sigma_r}{\sigma_r^2+\delta}\right)^2 \\ &= \frac{1}{M}\cdot\left(\sum_{r=1}^{R}\frac{\sigma_r^2}{(\sigma_r^2+\delta)^2}\right)\cdot \mathrm{E}\left[\|\boldsymbol{e}\|_2^2\right]. \qquad (9)\end{aligned}$$

Note that

$$\frac{\sigma_r^2}{(\sigma_r^2+\delta)^2} \;\le\; \min\left(\frac{1}{\sigma_r^2},\frac{1}{4\delta}\right),$$

so we can think of the error in (9) as an average of the $\frac{1}{\sigma_r^2}$, with the large values simply replaced by $1/(4\delta)$.

A Closed Form Expression

Tikhonov regularization is in some sense very similar to the truncated SVD, but with one significant advantage: we do not need to explicitly calculate the SVD to solve (4). Indeed, the solution to (4) can be written as

$$\hat{\boldsymbol{x}}_{\text{tik}} = (\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A} + \delta\mathbf{I})^{-1}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{y}. \tag{10}$$

To see that the expression above is equivalent to (7), note that we can write $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}$ as

$$\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A} = \boldsymbol{V}\boldsymbol{\Sigma}^2\boldsymbol{V}^{\mathrm{T}} = \boldsymbol{V}'\boldsymbol{\Sigma}'^2\boldsymbol{V}'^{\mathrm{T}},$$

where $\boldsymbol{V}'$ is $N \times N$,

$$\boldsymbol{V}' = \begin{bmatrix}\boldsymbol{V} & \boldsymbol{V}_0\end{bmatrix},$$

and the $N \times N$ diagonal matrix $\boldsymbol{\Sigma}'$ is simply $\boldsymbol{\Sigma}$ padded with zeros:

$$\boldsymbol{\Sigma}' = \begin{bmatrix}\boldsymbol{\Sigma} & \mathbf{0} \\ \mathbf{0} & \mathbf{0}\end{bmatrix}.$$

The verification of (10) is now straightforward:

$$\begin{aligned}
(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A} + \delta\mathbf{I})^{-1}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} &= (\boldsymbol{V}'\boldsymbol{\Sigma}'^2\boldsymbol{V}'^{\mathrm{T}} + \delta\mathbf{I})^{-1}\boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{U}^{\mathrm{T}}\boldsymbol{y} \\
&= \boldsymbol{V}'(\boldsymbol{\Sigma}'^2 + \delta\mathbf{I})^{-1}\boldsymbol{V}'^{\mathrm{T}}\boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{U}^{\mathrm{T}}\boldsymbol{y} \\
&= \boldsymbol{V}'(\boldsymbol{\Sigma}'^2 + \delta\mathbf{I})^{-1}\begin{bmatrix}\boldsymbol{\Sigma}\boldsymbol{U}^{\mathrm{T}}\boldsymbol{y} \\ \mathbf{0}\end{bmatrix} \\
&= \boldsymbol{V}(\boldsymbol{\Sigma}^2 + \delta\mathbf{I})^{-1}\boldsymbol{\Sigma}\boldsymbol{U}^{\mathrm{T}}\boldsymbol{y} \\
&= \hat{\boldsymbol{x}}_{\text{tik}}.
\end{aligned}$$

The expression (10) holds for all M, N, and R. We will leave it as an exercise to show that

$$(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A} + \delta\mathbf{I})^{-1}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} = \boldsymbol{A}^{\mathrm{T}}(\boldsymbol{A}\boldsymbol{A}^{\mathrm{T}} + \delta\mathbf{I})^{-1}\boldsymbol{y}.$$

The importance of not needing to explicitly compute the SVD is significant when we are solving large problems. When $\boldsymbol{A}$ is large ($M, N > 5000$, say) it is impossible to construct the matrix and compute with it explicitly. However, if it has special structure (if it is sparse, for example), then it may take many fewer than MN operations to compute a matrix vector product $\boldsymbol{A}\boldsymbol{x}$.

In these situations, a **matrix free** iterative algorithm can be used to perform the inverse required in (10). A prominent example of such an algorithm is **conjugate gradients**, which we will see later in this course.