TRUNCATED SVD AND DECONVOLUTION

Let $f : [0, 1] \to \mathbb{R}$ be a signal to be estimated from noisy samples of the convolution integral,

$$g(s) = \int_0^1 a(s - t)f(t)dt + e(s),$$

where $a$ is a known convolution kernel. Discretization:

$$y_j = g(s_j) \approx \frac{1}{N} \sum_k a(s_j - t_k)f(t_k) + e(s_j).$$

Denote $x_k = f(t_k), k = 1, 2, \ldots, N.$
Example: Optical blur

\[ y_j \]

\[ y_{j+n} \]

\[ x_{j-k} \]

\[ x_j \]

\[ x_{j+n} \]
MATRICES MODEL

\[ x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}. \]

Symmetric kernel:

\[ a = \begin{bmatrix} a_{-L} \\ a_{-L+1} \\ \vdots \\ a_{L-1} \\ a_L \end{bmatrix} \in \mathbb{R}^{2L+1}. \]
Matrix model

Write the matrix equation

\[ y = Ax + e, \]

where \( A \in \mathbb{R}^{N \times N} \) is the Toeplitz matrix,

\[
A = \begin{bmatrix}
a_0 & a_{-1} & \cdots & a_{-L} \\
a_{-1} & a_0 & & \\
& \ddots & \ddots & \ddots \\
& & a_{-L} \\
a_L & & & \ddots \\
& a_L & & \ddots \\
& & a_0 & a_{-1} \\
& & a_L & \cdots & a_1 & a_0
\end{bmatrix}.
\]

The parameter \( L \) defines the bandwidth of the matrix.
Defining the blurring kernel an true signal

Gaussian blurring kernel,

\[ a(t) = \frac{1}{\sqrt{2\pi w^2}} \exp\left(-\frac{1}{2w^2} t^2\right). \]

The true signal is a boxcar function,

\[ x_j = \begin{cases} 
0, & \text{if } j < n_1 \text{ or } j > n_2, \\
1, & \text{if } n_1 \leq j \leq n_2.
\end{cases} \]
In Matlab

```
N = 60;
t = linspace(0,1,N);
width = 0.1;
a = 1/sqrt(2*pi*width^2)*exp(-((1/(2*width^2))*t.^2));
A = (1/N)*toeplitz(a);

n1 = 25;
n2 = 40;
xtrue = zeros(N,1);
xtrue(n1:n2) = ones(n2-n1+1,1);
```
Adding noise

Gaussian additive noise, the standard deviation (STD) 2% of the maximum of the noiseless signal:

\[
b0 = A*xtrue;
\]
\[
\text{noiselevel} = 0.02*\text{max}(b0);
\]
\[
\text{noise} = \text{noiselevel}*\text{randn}(N,1);
\]
\[
b = b0 + \text{noise};
\]
plot(t,xtrue,'b-','LineWidth',1.2);
hold on
plot(t,b,'r-','LineWidth',1.2);
hold off
SVD: plot singular values

\[ A = UDV^T, \]

where

\[ D = \text{diag}(d_1, \ldots, d_n). \]

Machine epsilon: smallest non-negative number that the machine recognizes to be non-zero. Below that level, values are cluttered under the roundoff errors. In Matlab \( \text{eps} = 2.2204e-016. \)

Singular values below \( \text{eps} \) can be treated as zeros.
SVD in Matlab and logarithmic plot

```matlab
[U,D,V] = svd(A);
d = diag(D);
r = max(find(d>eps));
semilogy(d,'b.','MarkerSize',8);
hold on
semilogy([0,N],[eps,eps],'r-');
text(r+2,1e-14,['r = ',num2str(r)]);
hold off
```
Calculating TSVD\((k)\)–estimates

\[
\hat{x}(k) = \sum_{j=1}^{k} \frac{1}{d_j} (u_j^T b)v_j.
\]

Xk = zeros(N,r);
normX = zeros(r,1);
discr = zeros(r,1);
for k = 1:r
    Xk(:,k) = V(:,1:k)*diag(1./d(1:k))*U(:,1:k)'*b;
    normX(k) = norm(Xk(:,k));
    discr(k) = norm(b - A*Xk(:,k));
end
Plotting the discrepancy curve

Estimate of the noise level: Here, we set

$$\delta = 1.2 \|e\|.$$

Notice, that in reality, $\|e\|$ is not known and has to be estimated.

```matlab
plot([1:r],discr,'b.'],'MarkerSize',8); hold on plot([1:r],discr,'k-','LineWidth',0.8) plot([0,r],[delta,delta],'r-') hold off
```
Optimal value seems to be $k = 6$ or $k = 7$. 
Plotting the L–curve

Plot in log–log scale the points

\[ (\|\hat{x}^{(k)}\|, \|A\hat{x}^{(k)} - y\|), \quad k = 1, 2, \ldots, r. \]

```
loglog(normX,discr,'b.','MarkerSize',8);
hold on
loglog(normX,discr,'k-','LineWidth',0.8)
hold off
```
Optimal $k$ again around 5–7.
Plotting the solutions

Solutions corresponding to values

\[ k = 5, 6, 7. \]
Is it important to cut off the singular values?

Discrepancy curve with all singular values retained
L–curve with all singular values retained.