Deconvolution

Problem 9.4

Contents

Setup

```
clear all;
load('blocks_deconv.mat');
L = length(h);
N = length(x);
M = N + L - 1;
```

Part a

```
A = conv_mat(h, N);
```

Part b

```
sigmas = svd(A);
sigma1 = sigmas(1);
x_clean = pinv(A)*y;
figure;
hold on;
xlabel('n');
ylabel('x, x_clean');
title('Reconstruction of x_clean without noise')
plot(x_clean)
plot(x, 'o')
```

Part c

```
x_n = pinv(A) * yn;
figure;
title('Plotting x reconstructed from noisy observations');
plot(x_n);
err_x = norm(x-x_n)^2
err_y = norm(y-yn)^2
```

Part d

```
[U, S, V] = svd(A);
q = 250
A_truninv = zeros(N,M);
A_trun = zeros(M, N);
for r = 1:512-q
    vr = V(:, r);
    ur = U(:, r);
    sr = S(r, r);
    A_trun = A_trun + (sr)*ur*vr';
end

A_truninv = pinv(A_trun);
x_trunc = A_truninv*yn;
figure; hold on;
plot(x);
plot(x_trunc);
```

Part e

```
[U, S, V] = svd(A);
U = U(:, 1:rank(A));
V = V(1:rank(A), :);
S = S(1:rank(A), 1:rank(A));

delta = 0.001;
err = zeros(length(delta), 1);
sigmat = S.^2 + delta*eye(rank(A));
x_tik = V * pinv(sigmat) * S * U' * yn;

figure;
hold on;
plot(x)
plot(x_tik);
```

Helper Function

```
function [A] = conv_mat(h, N)
    L = length(h);
    M = N + L - 1;
    h_new = zeros(2*N + L - 2, 1);
    h_new(1:N-1) = 0;
    h_new(N:N+L-1) = h;
    h_new(N+L:2*N+L-2) = 0;
    A = zeros(M, N);
    for i = [1:M]
        for j = [1:N]
            A(i, j) = h_new((i+N) - j);
        end
    end
end
```