Homework x **Hongyu Yan (516030910595)**
CS 259 Numerical Methods for Data Science ACM Class, Zhiyuan College, SJTU
Prof. **David Bindel** Due Date: June 20, 2018
TA. **Yurong You, Xinran Zhu** Submit Date: June 21, 2018

Problem 1

The lagrangian $f(x)$ is

$$f(x) = X^T MX + \lambda^T(AX - b)$$

So the KKT conditions are

$$\nabla f(x) = 2MX + A^T\lambda = 0 \tag{1}$$

$$AX - b = 0 \tag{2}$$

From equation(1) we get

$$X = -\frac{1}{2}M^{-1}A^T\lambda \tag{3}$$

Substituting the equation(3) into the equation (2) gives:

$$\frac{1}{2}AM^{-1}A^T\lambda = b \tag{4}$$

From the equation(4) we get

$$\lambda = 2bA^{-T}MA^{-1} \tag{5}$$

Substitute it to the equation(3) then get X.

Problem 2

Sometimes the curve become extremely strange (like figure1)

Test error: 4.335756e+04

After using Tikhonov inverse, Test error: 7.594892e+00

After using trucated SVD, Test error: 7.715455e+00

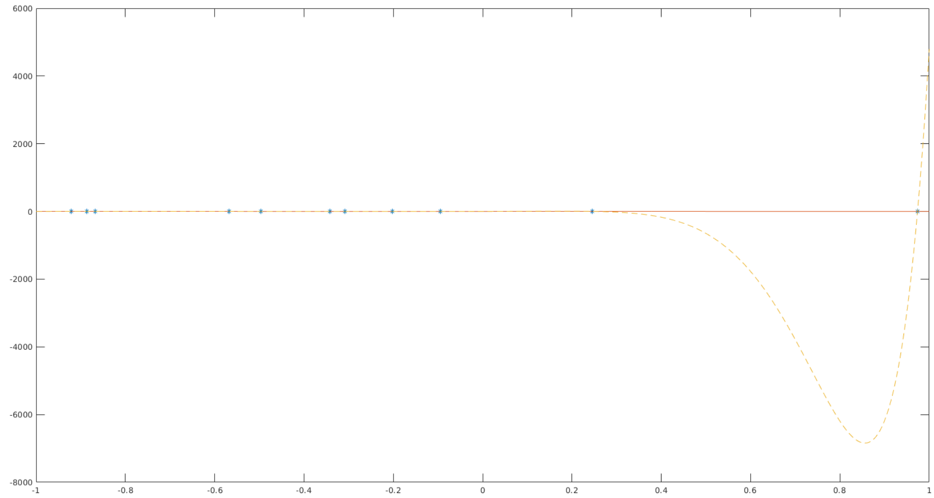

Figure 1: original method

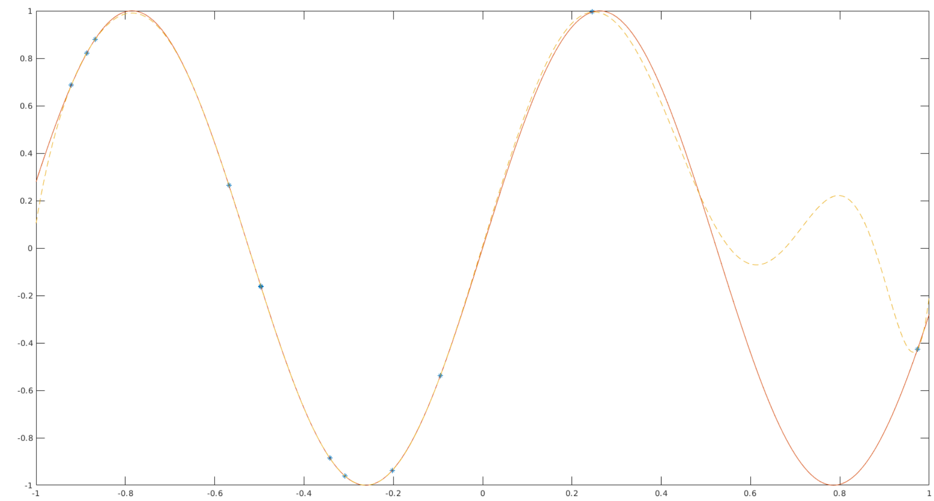

Figure 2: using Tikhonov inverse

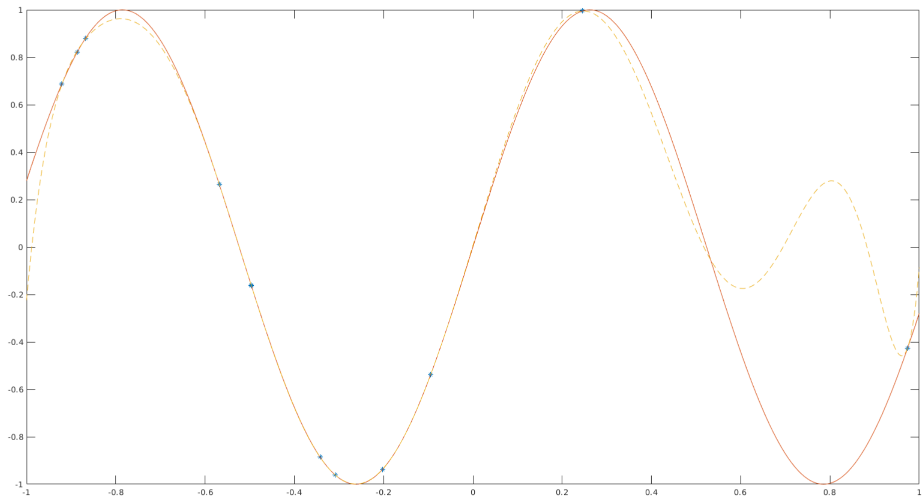

Figure 3: using trucated SVD

Code

```
% Demonstration of overfitting in a polynomial regression problem.
% We approximate sin(6x) by sum c_j T_j(x) where T_j(x) are the
% Chebyshev polynomials, and determine the coefficients by a least
% squares fit with noise.

m = 12;       % Number of data points
n = 12;       % Number of expansion terms
sig = 1e-4;   % Noise level

% Set up interpolation and test points

x  = 2*rand(m,1)-1;
xt = linspace(-1,1,400)';

% Function values
fx  = sin(6*x);
fxt = sin(6*xt);
fxe = fx + sig*randn(m,1);

% Evaluate Chebyshev polynomials at points x and fit
A = chebmatrix(x, n);
c = A\fxe;

% Predict at test grid xt
At = chebmatrix(xt,n);
fxt_pred = At*c;

% Tikhonov inverse
```

```
[u z v] = svd(A, 'econ');
lambda = sqrt(z(1,1)) / 100;
d = (A'*A + lambda^2*eye(n,n)) \ (A' * fxe);

% truncated_SVD
threshold = 0.5
infinity = 1e7
% [u z v] = svd(A, 'econ');
for i=1:length(z)
        if z(i,i) < threshold
                z(i,i) = infinity;
        end
end
newA = u*z*v';
e = newA\fxe;

% Show the norm of the test error
test_err = fxt-fxt_pred;
fprintf('Test error: %e\n', norm(test_err));
fprintf('After using Tikhonov inverse, Test error: %e\n', norm(fxt-At*d));
fprintf('After using trucated_SVD, Test error: %e\n', norm(fxt-At*e));

% Plot data
figure(1); plot(x, fxe, '*', xt, fxt, '-', xt, At*c, '--');
figure(2); plot(x, fxe, '*', xt, fxt, '-', xt, At*d, '--');
figure(3); plot(x, fxe, '*', xt, fxt, '-', xt, At*e, '--');
figure(4); semilogy(x, abs(fxe-fx), '*', xt, abs(fxt-At*c));
```