Project Report of Assignment 2

ID: *14307130013*
E-mail: 14307130013@fudan.edu.cn

Oct. 20th

1 The least squares problem

1.1 Abstract

This report discusses the least squares problem. We approach the topic by dealing with a polynomial fitting problem $\min\limits_{c} ||Ac - y||$ at first. After that we take a step forward by adding a regularization parameter α. We draw a conclusion that a regularization term can well tackle ill-conditioned least squares problems.

1.2 Introduction

The method of least squares is often deployed in regression analysis to minimize the sum of squared residuals, i.e., the gap between a specific observed value and a fitted value given by the model. This is because a regression problem such as $Ac = y$, is usually overdetermined. We minimize the sum of squared residuals to create an optimal model instead of looking for an exact solution that does not exist.

1.3 Definitions and theories

1.3.1 Condition number[1]

As for a square matrix A, the condition number $cond(A) = ||A|| \cdot ||A^{-1}||$. If the condition number is not too much larger than 1, the matrix is well conditioned which means its inverse can be computed with good accuracy. Otherwise, the matrix is said to be ill-conditioned for it is almost singular and the computation of its reverse is prone to large numerical errors.

[1] From Wikipedia: Condition number. (https://en.wikipedia.org/wiki/Condition_number)

1.3.2 Matlab's *cond*[2]

In Matlab, the function *cond(A)* gives the 2-norm condition number by default regardless of whether A is square. The result is the ratio of the largest singular value of A to the smallest.

1.3.3 QR decomposition[3]

In linear algebra, a QR decomposition (also called a QR factorization) of a matrix is a decomposition of a matrix A into a product $A = QR$ of an orthogonal matrix Q and an upper triangular matrix R. QR decomposition is often used to solve the linear least squares problem, and is the basis for a particular eigenvalue algorithm, the QR algorithm.

1.4 Algorithms

1.4.1 Solving c in $\min\limits_c ||Ac - y||$ by normal equation

In the lecture, we have $c = (A^T A)^{-1} A^T y$ to minimize the loss function. Therefore we implement a function as below:

Matlab codes:

```
function c = cSolver_1()
% Solve c by normal equation.
t = linspace(-3, 3, 50);
f_t = 3 + 2 * t - 4 * t.^2 - t.^3;
y = f_t' + 0.5 * randn(50, 1);

A = ones(50, 1);
for i = 1: 20
    A = [A, t'.^i];
end

c = (A' * A) \ A' * y;
```

1.4.2 Solving c in $\min\limits_c ||Ac - y||$ by QR decomposition

Matlab codes:

```
function c = cSolver_1qr()
% Solve c by economical QR.
% Identical codes omitted.
```

[2] http://cn.mathworks.com/help/matlab/ref/cond.html

[3] From Wikipedia: QR decomposition. (https://en.wikipedia.org/wiki/QR_decomposition)

```
...
[Q1, R1] = qr(A, 0);

c = R1 \ Q1' * y;
```

1.4.3 Solving c in $\min_{c} ||Ac - y||^2 + \alpha||c||^2$ by normal equation

By computing the gradient of the loss function $||Ac - y||^2 + \alpha||c||^2$, we have $c = (A^T A + \alpha I)^{-1} A^T y$.

Matlab codes:

```
function c = cSolver_2()
% Solve c in the loss function with a regularization parameter.
% Identical codes omitted.
...

alpha = 1000;

I = eye(size(A' * A));

c = (A' * A + alpha * I) \ A' * y;
```

1.4.4 Testing codes

We run the following program n times to observe the mean, variance and distribution of each c_i from c:

Matlab codes:

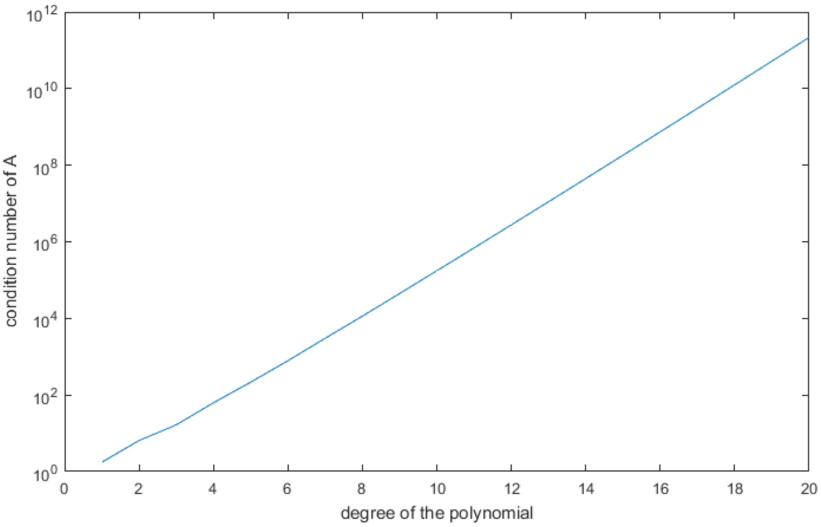
```
n = 1000;

C = zeros(21, n);
for i = 1: n
    C(:, i) = cSolver_1();
end

c_mean = mean(C, 2);
c_var = var(C, 0, 2);
c_normtest = zeros(21, 1);
for i = 1: 21
    c_normtest(i) = lillietest(C(i, :));
end
```

1.5 Implementation issues discussion

1.5.1 A condition number-degree plot

We construct A of different degrees through a loop. Then the condition numbers can be computed easily by function *cond*. To demonstrate the relation between the condition number and degree, *plot* performs poorly due to the rapid increasement of condition number when degree is over 18. Thus, *semilogy* is used and works well. The output is shown below:

Figure 1: Relation between the condition number and degree

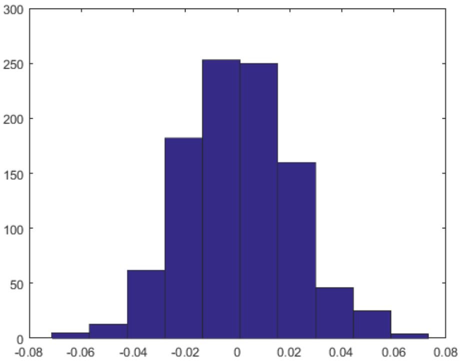


From Figure 1, we approximately have a curve of $cond(A) = x^{a \cdot degree + b}$, with $x \approx 3.6$.

1.5.2 Statistical properties for c in $\min_c ||Ac - y||$

Results given by both methods are similar. In regards to the mean, the first 4 components of c are close to (3, 2, -4, -1) as expected. Then the mean starts to decrease as the value of i in c_i goes up. As for the variance, those of c_4 to c_{11} are apparently larger than the others. In addition, the distribution of each component throughout n tests is fitted by *lillietest* to determine if there is a normal distribution. The output *c_normtest* is an all-zero vector. Then we use *hist* and *normplot* for more observation.

Taking c_{16} as example, we can tell 1000 results of each component conform to a normal distribution due to the normally distributed noise.



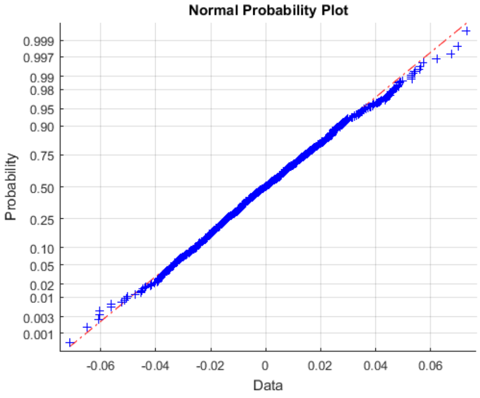


Figure 2: Histogram of c_{16}

Figure 3: Normality test of c_{16}

1.5.3 A least squares problem proof

Target: $\min\limits_c ||Ac - y||^2 + \alpha||c||^2$ is also a least square problem.

Proof:

$$\begin{aligned}\min_c ||Ac - y||^2 + \alpha||c||^2 &= \min_c c^T A^T Ac - 2c^T A^T y + y^T y + \alpha c^T c \\ &= \min_c c^T (A^T A + \alpha I)c - 2c^T A^T + y^T y \\ (B^T B = A^T A + \alpha I) &= \min_c c^T B^T Bc - 2c^T A^T y + y^T y \\ (z = (B^T)^{-1} A^T y) &= \min_c ||Bc - z||^2 - z^T z + y^T y\end{aligned}$$

Illustration:

1. Since $A^T A + \alpha I$ is positive definite, there exists an invertible matrix B so that $B^T B = A^T A + \alpha I$.

2. The term $-z^T z + y^T y$ is irrelative with c, so the initial problem is equal to $\min\limits_c ||Bc - z||^2$, which is a least squares problem.

1.5.4 Statistical properties for c in $\min\limits_c ||Ac - y||^2 + \alpha||c||^2$

Same tests are run upon *cSolver_2()*. The mean and variance (absolute value) of each c_i are way smaller than those delivered in **Section 1.5.2**. The comparison is shown in the next section. Yet the normality still occurs.

1.5.5 Comparison among 3 algorithms

The testing program is modified in order to run the 3 algorithms over the same set of data. We plot the means and variances together:

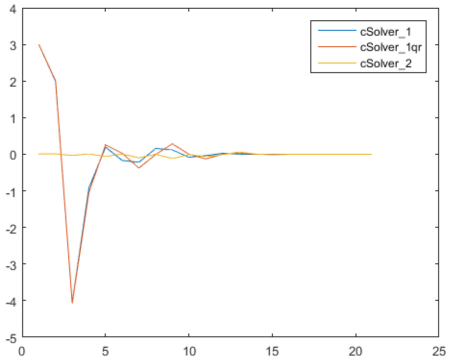


Figure 4: Mean comparison

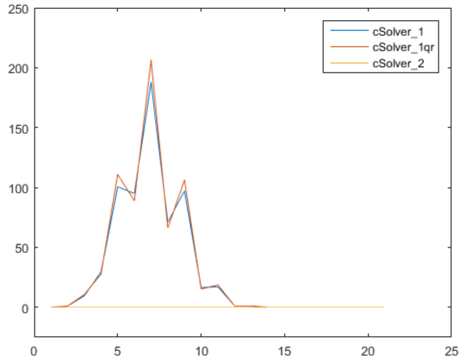


Figure 5: Variance comparison

1.6 Experiment results

1.6.1 Discussion on Figure 1

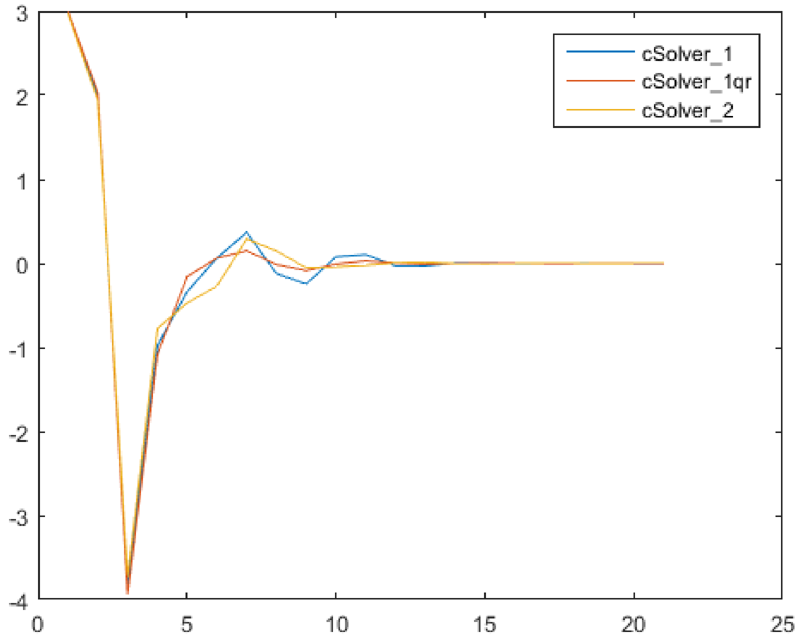
Walter Gautschi et al. (1988) gave the lower bounds for the condition number of Vandermonde Matrices. According to Victor Y. Pan (2015), it has been proven that the condition number $\kappa(V_s)$ of a Vandermonde Matrix V_s, is exponential in n if all knots are real.

1.6.2 Discussion on Figure 4 and Figure 5

The expectations of the first 4 components converge to true values as n increases. The larger i is, the less significant c_i is. However, the large variances mentioned before indicate the unstablity of the components after c_4. This is because the unstablity brought by large condition number when degree is 20. To illustrate this, a theorem in numerical analysis can be presented here. Let $\hat{c}$ be the approximate solution to the least squares problem we focus on. We have $\frac{||c-\hat{c}||}{||c||} \leq cond(A) \cdot \frac{||y-A\hat{c}||}{||y||}$. Hence, even we minimize the sum of squared residuals, an A with a large condition number will still provides enormous error in solution. Besides, overfitting can be a reason too.

The variances are substantially reduced as a regularization parameter α is introduced. In machine learning, $\alpha||c||^2$ is known as regularization term or penalty term (of L2 norm). It is added into loss functions so as to avoid overfitting as well as deal with ill-conditioned problems when α is relatively small. In this case, we set $\alpha = 1000$, which is much larger than coefficients that are already known. This results in the reduction of mean of every c_i in c, for the regularization term “takes over” and derives inaccurate c. A small α is usually selected to overcome the drawback. If we have $\alpha = 0.01$ here and plot Figure 4 again, we will acquire a better solution c:

Figure 6: Mean comparison with $\alpha = 0.01$



1.7 Conclusion

The condition number of a Vandermonde Matrix is exponential in its degree. As the degree grows, the matrix will become ill-conditioned instantly, which may cause unstablity in the solution to a least squares problem.
A regularization term is introduced to optimize least squares problems. It takes a number of trials to choose the proper parameter α.

1.8 Acknowledgement

Great gratitude to TA Lu for giving the definition of statistical properties.

1.9 References

1. https://en.wikipedia.org/wiki/Condition_number
2. http://cn.mathworks.com/help/matlab/ref/cond.html
3. https://en.wikipedia.org/wiki/QR_decomposition
4. Walter Gautschi et al. *Lower Bounds for the Condition Number of Vandermonde Matrices*
5. Victor Y. Pan. *How Bad Are Vandermonde Matrices?*