Project 1: Curve Fitting / Linear Regression

Due on Tuesday, February 4, 2020 at 11.59PM

Name: Ke LIANG ID: 926791183 email: kul660@psu.edu

contents

1 Introduction

The curve fitting problem motivates a number of important key concepts covered in the book. We can estimate the distribution of the data set is polynomial function below.

$$y(x,\omega) = \sum_{i=0}^{M}(\omega_0 + \omega_1 * x + \omega_2 * x^2 + \cdots + \omega_M * x^M) = \sum_{i=0}^{M}(\omega_i * x^i) \quad (1)$$

$$\omega=[\omega_0, \omega_1,\ldots,\omega_M]^T \quad (2)$$

$$T = [t_0 , t_1 ,\ldots, t_N]^T \quad (3)$$

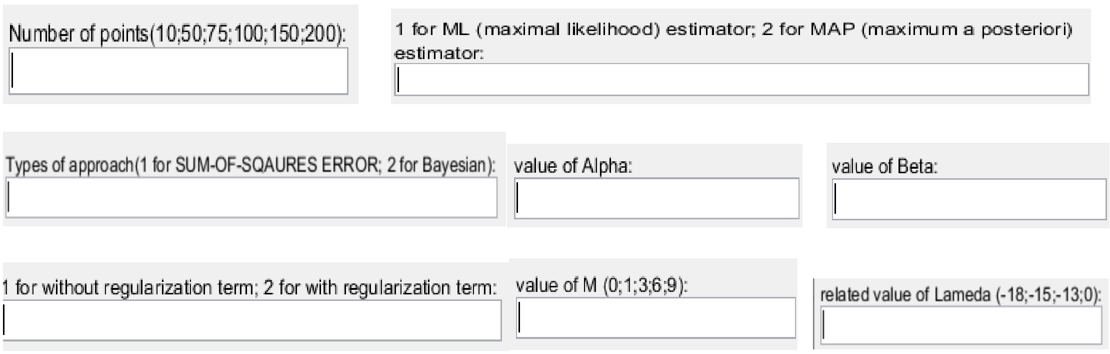
$$X = \begin{bmatrix} x_1^0 & \cdots & x_1^M \\ \vdots & \ddots & \vdots \\ x_N^0 & \cdots & x_N^M \end{bmatrix} \quad (4)$$

And for different points of sample, we use the approach below, we get different results. This project asks you to solve the linear regression problem by two different approaches: 1) direct error function (the sum-of-squares error) minimization and 2) Bayesian approach. For the direct error function, it is the sum-of-squares error in this project and we get the minimal error with or without the regularization term. And for Bayesian approach, the ML (maximal likelihood) estimator and the MAP (maximum a posteriori) estimator are used.

2 Methods

We first run the generateData.m and then we can get data_10.mat, data_50.mat, data_75.mat, data_100.mat, data_150.mat, and data_200.mat for the data set of 10 points, 50 points , 75 points, 100 points , 150 points and 200 points.

Then for the experiment part, we run curveFit.m with different input as shown below, and after this we can get different results for different task.



2.1 Error Minimization Without Regularization Term

For direct error function, we use sum-of-squares error without regularization term first. The choice of error function, which is widely used, is given by the sum of the squares of the errors between the predictions $y(x_n,w)$ for each data point x_n and the corresponding target values t_n, so that we minimize the E(ω) below.[1]

$$E(\omega) = \frac{1}{2}\sum_{n=1}^{N}\{y((x_n, \omega) - t_n\}^2 \tag{5}$$

$$E(\omega) = \frac{1}{2}(X\omega - T)^T(X\omega - T) \tag{6}$$

$$\frac{\partial E(\omega)}{\partial \omega} = X^T(X\omega - T) \tag{7}$$

$$\frac{\partial E(\omega)}{\partial \omega} = 0 \tag{8}$$

$$\omega^* = (X^T X)^{-1} X^T T \tag{9}$$

2.2 Error Minimization With Regularization Term

For direct error function, we can also use sum-of-squares error with regularization term first. One technique that is often used to control the over-fitting phenomenon in such cases is that of regularization, which involves adding a penalty term to the equation (5) in order to discourage the coefficients from reaching large values. The simplest such penalty term takes the form of a sum of squares of all of the coefficients, leading to a modified error function of the form.[1]

$$E(\omega) = \frac{1}{2}\sum_{n=1}^{N}\{y((x_n, \omega) - t_n\}^2 + \frac{\lambda}{2}\|\omega\|^2 \text{ where } \|\omega\|^2 = \omega^T\omega \tag{10}$$

$$E(\omega) = \frac{1}{2}(X\omega - T)^T(X\omega - T) + \frac{\lambda}{2}\omega^T\omega \tag{11}$$

$$\frac{\partial E(\omega)}{\partial \omega} = X^T(X\omega - T) + \lambda\omega \tag{12}$$

$$\frac{\partial E(\omega)}{\partial \omega} = 0 \tag{13}$$

$$\omega^* = (X^TX + \lambda)^{-1} X^TT \tag{14}$$

2.3 ML (Maximal Likelihood) Estimator

For the Bayesian approach, we now use the training data {x,t} to determine the values of the unknown parameters w and β by maximum likelihood. If the data are assumed to β)be drawn independently from the distribution equation (15), then the likelihood function is given by equation (16). [1]

$$P(t|x, \omega, \beta) = N(t|y(x, \omega), \beta^{-1}) \tag{15}$$

$$P(t|x, \omega, \beta) = \prod_{n=1}^{N} N(t_n|y(x_n, \omega), \beta^{-1}) \tag{16}$$

$$\ln P(t|x, \omega, \beta) = \frac{\beta}{2}\sum_{n=1}^{N}\{y((x_n, \omega) - t_n\}^2 + \frac{N}{2}ln\beta - \frac{N}{2}\ln(2\pi) \tag{17}$$

We can also use maximum likelihood to determine the precision parameter β of the Gaussian conditional distribution. Maximizing (17) with respect to β gives the equation (18) below.

$$\frac{1}{\beta_{ML}} = \frac{1}{N}\sum_{n=1}^{N}\{y((x_n, \omega_{ML}) - t_n\}^2 \tag{18}$$

$$\beta_{ML} = \frac{N}{(X\omega - T)^T(X\omega - T)} \tag{19}$$

$$\sigma^2 = 1/\beta_{ML} \tag{20}$$

2.4 MAP (Maximum A Posteriori)timator

We can now determine w by finding the most probable value of w given the data, in other words by maximizing the posterior distribution. This technique is called maximum posterior, or simply MAP. We find that the maximum of the posterior is given by the minimum of the equation (20) bellow.[1]

$$\frac{\beta}{2}\sum_{n=1}^{N}\{y((x_n, \omega) - t_n\}^2 + \frac{\alpha}{2}\omega^T\omega \tag{20}$$

Thus we see that maximizing the posterior distribution is equivalent to minimizing the regularized sum-of-squares error function encountered earlier in the form (1.4), with a regularization parameter given by $\lambda = \alpha/\beta$[1]

3 Results

3.1 General Tasks

• error minimization (refer to Equation 1.2, page 5)

• error minimization with the regularization term (refer to Equation 1.4, page 10) (You can generate plots similar to Figure 1.7, page 10).

• the ML (maximal likelihood) estimator of the Bayesian approach (refer to Equation 1.62, page 29)

• the MAP (maximum a posteriori) estimator of the Bayesian approach (refer to Equation 1.67, page 30 and Equation 3.55, page 153, you can use beta = 11.1 and alpha = 0.005 as shown in textbook)

The ground truth of curve funtion (sinx/2) is the blue line in the Figure and the data together with the Guassian noises is shown as red circle in the Figure.

3.1.1 Error Minimization Without Regularization Term

We choose 10 for "Number of points", 1 for "Type of approach", 1 for "Without Regularization Term" and 0, 1, 3, 6 and 9 for M. We can get the Figure 1, Figure 2, Figure 3, Figure 4 and Figure 5 below. The green line is the polynomial function.

We can find that when $M = 0$ and $M = 1$, it fits poorly to the training data, and when $M = 9$, the polynomial function fits every training data, but it is overfitting since it cannot fit the function sin(x/2). When $M = 3$ and $M = 6$ fit well to the training data, and also fit well to the function sin(x/2), and it is easy to see that $M = 3$ is better, for example,when $0 < x < 4$ and $10 < x < 12$, the curve of $M = 3$ fits better than the curve of $M = 6$ to the function sin(x/2).

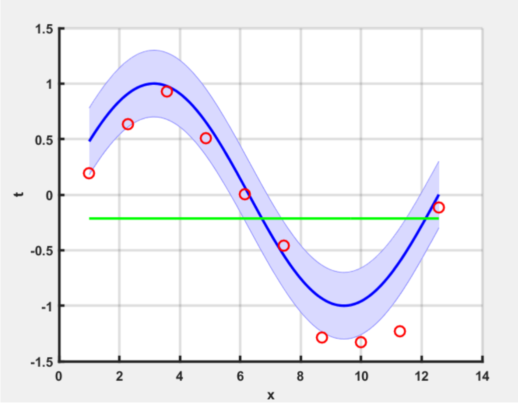


Figure 1 N=10 M=0 Error Minimization

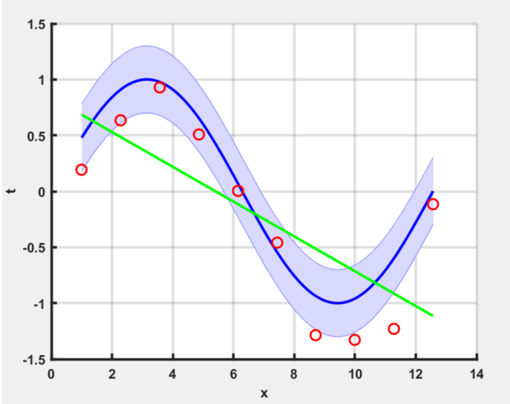


Figure 2 N=10 M=1 Error Minimization

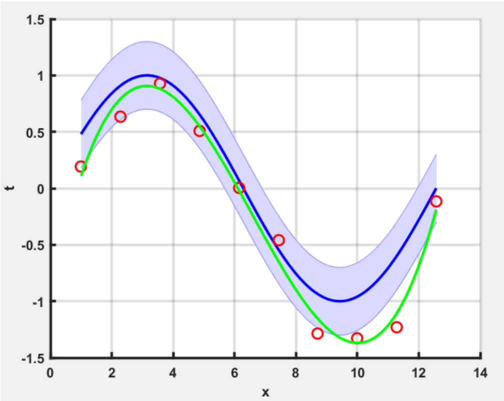


Figure 3 N=10 M=3 Error Minimization

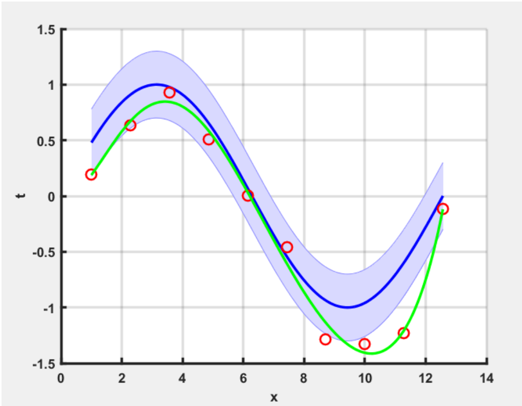


Figure 4 N=10 M=6 Error Minimization

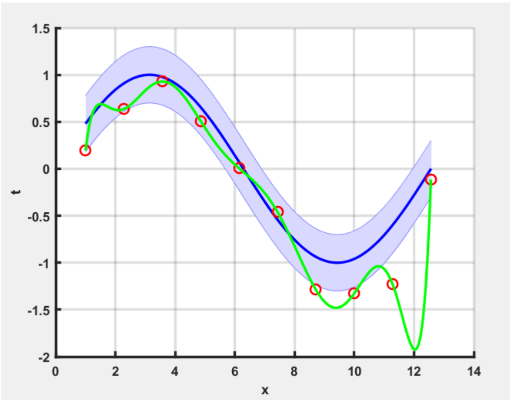


Figure 5 N=10 M=9 Error Minimization

3.1.2 Error Minimization WithRegularization Term

Regulartion is used to control the over-fitting phenomenon by adding a penalty term in order to contrain the value of ω^* not too large.

In this project, we choose 10 for “Number of points”, 1 for “Type of approach”, 2 for “Regularization Term” and 9 for M, since when M = 9 the overfitting situation will appear, and we have lnλ = -18, lnλ = -15, lnλ = -13, and lnλ = 0. Based on the parameters above, We can get the Figure 6, Figure 7, Figure 8 and Figure 9 below. The green line is the polynomial function without regularization term, and red line is the polynomial function with regularization term.

It is found that if ln λ = -18, there is no change for the Figure which mean it can hardly impact on the overfitting, while with the smaller of the ln λ is, the more possitive impact on the situation of overfitting, and when ln λ = 0, the problem of overfitting is almost solved.

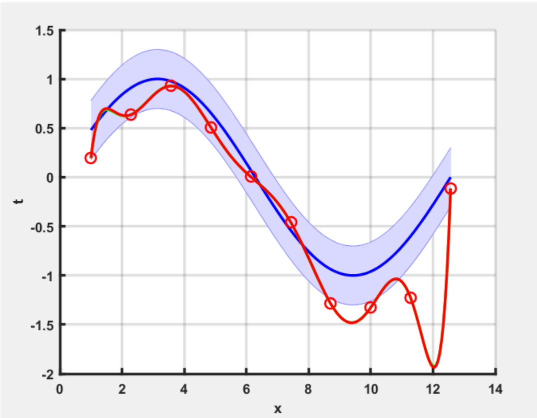


Figure 6 N=10 M=9 lnλ = -18 with regularization term

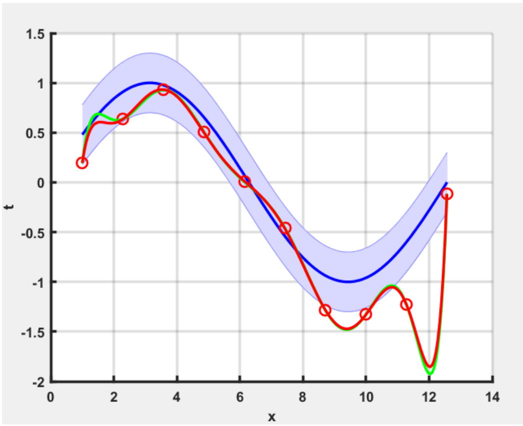


Figure 7 N=10 M=9 lnλ = -15 with regularization term

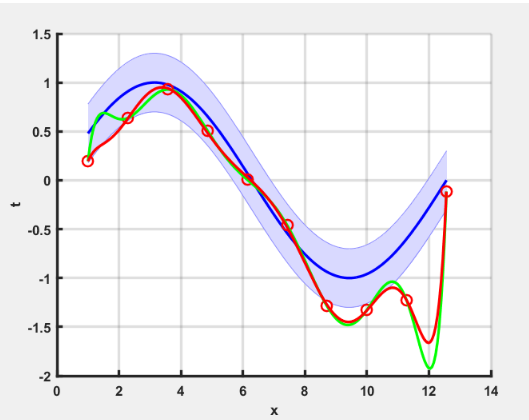


Figure 8 N=10 M=9 lnλ = -13 with regularization term

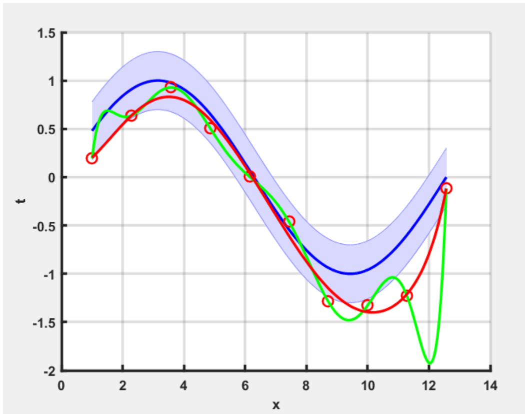


Figure 9 N=10 M=9 lnλ = 0 with regularization term

3.1.3 ML (Maximal Likelihood) Estimator

In this part, we choose 10 for "Number of points", 2 for "Type of approach", 1 for "Maximal Likelihood Estimator" and M = 3 since it is the best distribution performance for sin(x/2), then we can get BetaML = 90.4313 and σ = 0.1052. In the Figure 10, the red line is the ground truth for the sin(x/2), the green line is the polynomial function, and the red region represents the mean of the predictive distribution with σ.

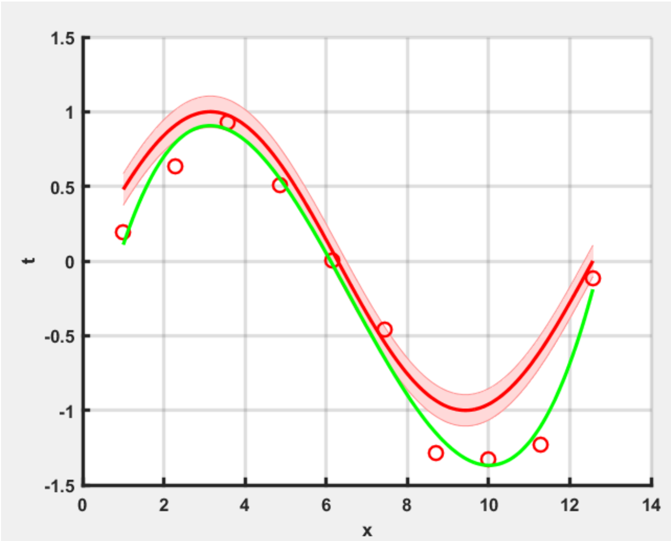


Figure 10 N=10 M=3 with ML

3.1.4 MAP (Maximum A Posteriori) Etimator

In this part, we choose 10 for "Number of points", 2 for "Type of approach", 2 for "Maximum A Posteriori Estimator", M = 9, β = 11.1 and α = 0.005 which are recommanded in the description. Then we can get the Figure 11 the green line is the predicted line using this method. But the Figure 11 shows that the result seems not well fit to the sin(x/2), so based on the Figure 9, we know when ln λ = 0, the problem of overfitting will be almost solved. Since λ = α/β, if we choose β = 11.1, we can choose α = 2.7 x 11.1 =29.97. Then we can get Figure 12, the green line is the predicted line and it almost solve the problem of overfitting.

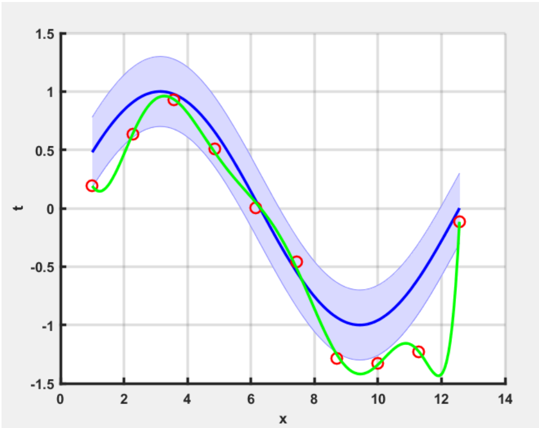


Figure 11 N=10 M=9 β=11.1 α=0.005 MAP

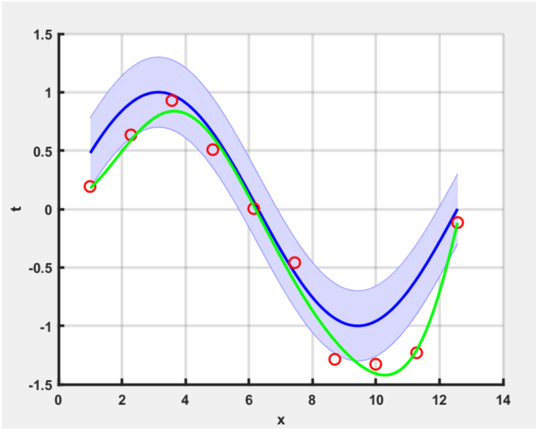


Figure 12 N=10 M=9 β=11.1 α=29.97 MAP

3.2 Extra Tasks

You may choose to include any the following for in your report for extra credits.

- Add to the plot of errors for the point ln λ = -18, -15 and 13 (Figure 1.8), and you are welcome to use more lambda values.
- For a fixed number of sample point (50 points), vary the order of polynomial M (M = 0,1,3,6,9). Generate a table similar to Table 1.1 (page 8).
- For a fixed degree of polynomial (M=9), vary the number of sample points N. Generate a plot similar to Figure 1.6 (page 9).

3.2.1 plot of errors for the point with different ln λ

In this part, we already get the Figures before which are Figure 6, Figure 7, Figure 8, and Figure 9.We run the code as 3.1.2 shows, and get the value of ω* (Wstar = [ω_9^*, ω_8^*, ω_7^*, ω_6^*, ω_5^*, ω_4^*, ω_3^*, ω_2^*, ω_1^*, ω_0^*] in the code), which is a vector including ω_9^* to ω_0^*. Then we can build the Table 1 with the value we get for different ln λ. And we can observe that showing that regularization has the desired effect of reducing the magnitude of the coefficients.

	ln λ = -18	ln λ = -15	ln λ = -13	ln λ = 0
ω_0^*	8.002E-06	7.131E-06	5.059E-06	1.291E-07
ω_1^*	-4.690E-04	-4.156E-04	-2.885E-04	-7.089E-06
ω_2^*	0.012	0.010	0.007	1.644E-04
ω_3^*	-0.160	-0.140	-0.091	-0.002
ω_4^*	1.334	1.153	0.721	0.015
ω_5^*	-6.905	-5.893	-3.480	-0.058
ω_6^*	21.997	18.490	10.130	0.074
ω_7^*	-41.208	-34.050	-16.975	0.096
ω_8^*	41.034	33.386	15.138	0.057
ω_9^*	-15.909	-12.763	-5.254	0.022

Table 1 Table of the coefficients ω* for M = 9 polynomials with various values for the regularization parameter λ

3.2.2 Table for a fixed number of 50 points and M = 0,1,3,6,9

We choose 50 for “Number of points”, 1 for “Type of approach”, 1 for “Without Regularization Term” and 0, 1, 3, 6 and 9 for M. We can get the Figure 13, Figure 14, Figure 15, Figure 16 and Figure 17 below.

The ground truth of curve funtion (sinx/2) is the blue line in the Figure, the data together with the Guassian noises is shown as red circle and the green line is the polynomial function in the Figure.

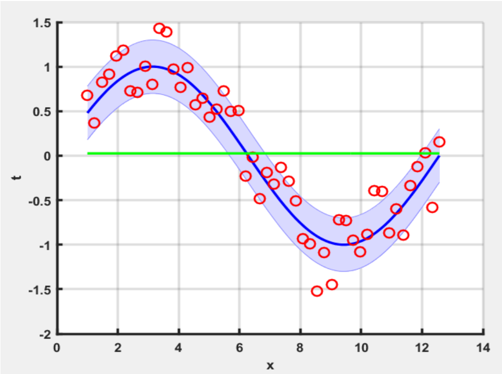


Figure 13 N=50 M=0 Error Minimization

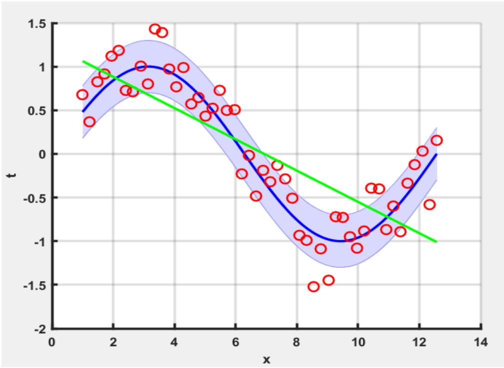


Figure 14 N=50 M=1 Error Minimization

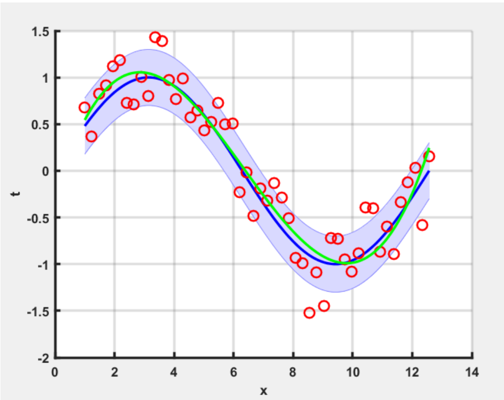


Figure 15 N=10 M=3 Error Minimization

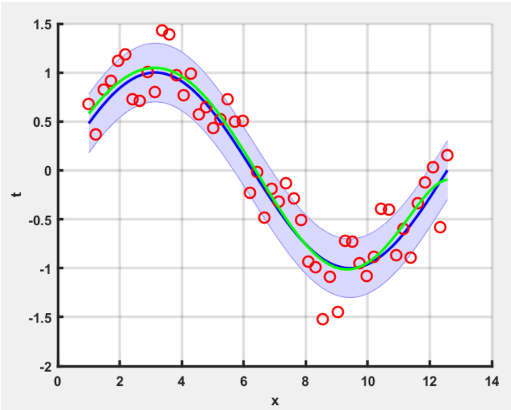


Figure 16 N=10 M=6 Error Minimization

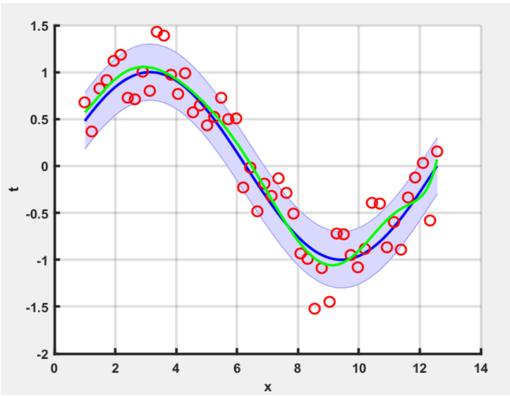


Figure 17 N=10 M=9 Error Minimization

In this situation, we can find that when M = 0 and M = 1, it fits poorly to the training data, and when M = 9, the polynomial function fits every training data, but it is overfitting since it cannot fit the function sin(x/2). When M = 3 and M = 6 fit well to the training data, and also fit well to the function sin(x/2), while it seems M = 6 fits better. And we run the code, and get the value of ω* (Wstar = [ω_9^*, ω_8^*, ω_7^*, ω_6^*, ω_5^*, ω_4^*, ω_3^*, ω_2^*, ω_1^*, ω_0^*], Wstar = [ω_9^*, ω_8^*, ω_7^*, ω_6^*, ω_5^*, ω_4^*, ω_3^*, ω_2^*, ω_1^*, ω_0^*], Wstar = [ω_9^*, ω_8^*, ω_7^*, ω_6^*, ω_5^*, ω_4^*, ω_3^*, ω_2^*, ω_1^*, ω_0^*], Wstar = [ω_9^*, ω_8^*, ω_7^*, ω_6^*, ω_5^*, ω_4^*, ω_3^*, ω_2^*, ω_1^*, ω_0^*], Wstar = [ω_9^*, ω_8^*, ω_7^*, ω_6^*, ω_5^*, ω_4^*, ω_3^*, ω_2^*, ω_1^*, ω_0^*] in the code), which is a vector including ω_9^* to ω_0^*. Then we can build the Table 2 with the value we get for different M. We can observe that We see that, as M increases, the magnitude of the coefficients typically gets larger.

	M = 0	*M = 1*	*M = 3*	*M = 6*	*M = 9*
ω_0^*	0.026	-0.179	0.012	-1.825E-05	4.075E-07
ω_1^*		1.241	-0.231	4.613E-04	-1.917E-05
ω_2^*			1.021	-0.003	3.490E-04
ω_3^*			-0.258	-0.002	-0.003
ω_4^*				-0.053	0.009
ω_5^*				0.529	0.036
ω_6^*				0.107	-0.365
ω_7^*					0.985
ω_8^*					-0.715
ω_9^*					0.625

Table 2 Table of the coefficients ω* for polynomials of various order

3.2.3 Figures for M=9, N = 10, 50, 75, 100, 150, 200

We choose 10, 50, 75, 100, 150, 200 for “Number of points”, 1 for “Type of approach”, 1 for “Without Regularization Term” and 9 for M. Then we can get Figure 18, Figure 19, Figure 20, Figure 21, Figure 22, and Figure 23.

The ground truth of curve funtion (sinx/2) is the blue line in the Figure, the data together with the Guassian noises is shown as red circle and the green line is the polynomial function in the Figure. And we can find that increasing the size of the data set reduces the over-fitting problem.

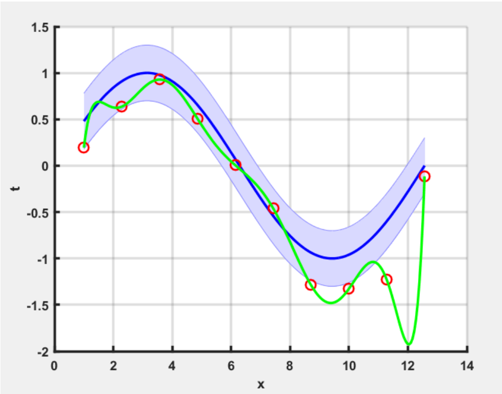


Figure 18 N=10 M=9 Error Minimization

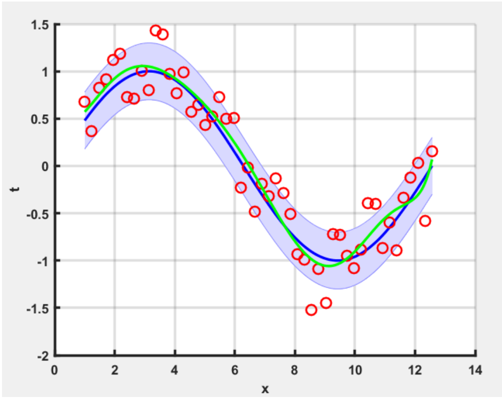


Figure 19 N=50 M=9 Error Minimization

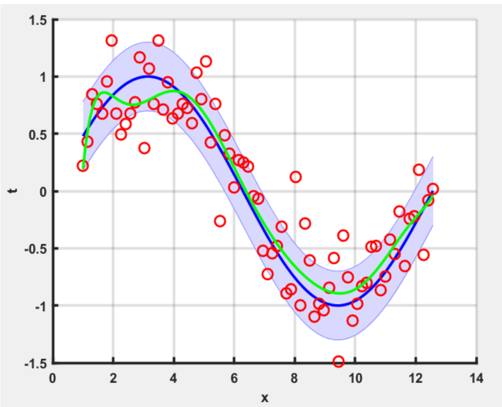


Figure 20 N=75 M=9 Error Minimization

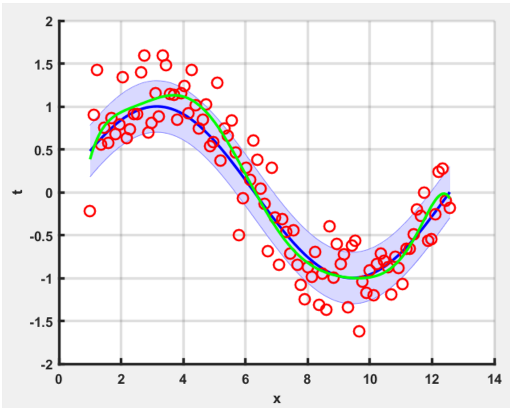


Figure 21 N=100 M=9 Error Minimization

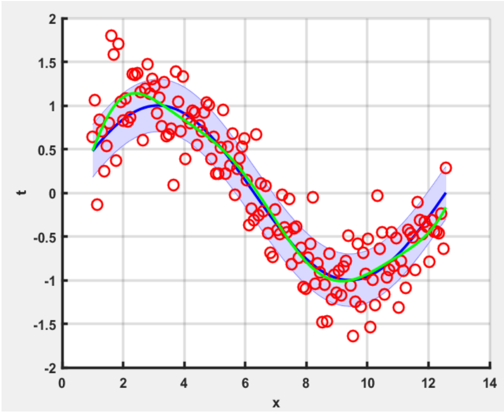


Figure 22 N=150 M=9 Error Minimization

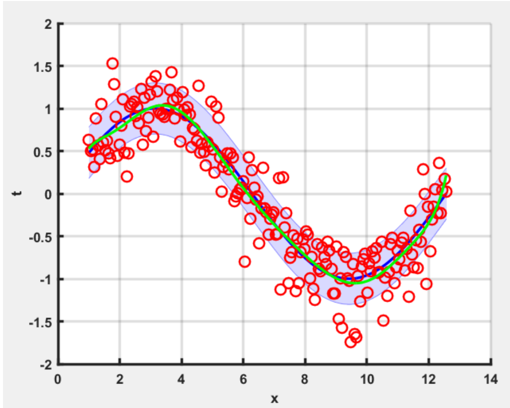


Figure 23 N=200 M=9 Error Minimization

4 Conclusions

The project is aimed to learn Error Minimization with or without regularization and Bayesian methods by solving curving fitting / linear regression.
It is easy to realize the code of error minimization method and when we are training the model, if M is too large, there may appear the prblem of overfitting, and regularization is helpful to this problem by controling the value of λ. And we can find that increasing the size of the data set can also reduce the over-fitting problem.
When we realize the method of Bayesian, we need to get the β based on the real data and get the standard error which can lead to the predictive distribution, and then we combine β with α to solve the problem of overfitting since $\lambda = \alpha/\beta$.

Reference

1 Christopher, M. Bishop. PATTERN RECOGNITION AND MACHINE LEARNING. Springer-Verlag New York, 2016.