CSE586 Project 1: Curve Fitting or Linear Regression

Sree Sai Teja Lanka

January 27, 2019

Abstract

The curve fitting problem motivates a number of important key concepts in the area of machine learning. The main idea of this project is to solve the linear regression problem by two different approaches: 1) direct error minimization and 2) Bayesian approach. This report enhances the comparison of the results for both the approaches. The purpose of this project is to have a good clench on the Bayesian modeling framework which is a building block for most of the machine learning concepts all over the frame. This project also helps the elementary level MATLAB user to have a good grip on how to use MATLAB for solving further problems in machine learning.

Contents

1 Introduction

The process to find the best suitable fit for the graph of set of data points is called the Curve Fitting, This is also called as Regression Analysis. This can be solved or achieved better to find the regression coefficients using the minimization factor to reduce the error of of the curve fitting for the following approaches
1.Direct error minimization
2. Bayesian approach

Direct error minimization: Sum of squared error method minimizes the error by taking considering the original data and the values predicted by the equation which we solved. This is the direct error minimization method to directly get the difference between the given and achieved values of test set. This is also widely known as Sum-of-least-squared error minimization method.

Bayesian approach minimization: It is an approach to linear regression in which the statistical analysis is undertaken within the context of Bayesian inference. The data are often supplemented with additional information in the form a prior probability distribution. The prior belief about the parameters is combined with datas likelihood function according to Bayes theorem yield a posterior belief about the parameters. Suppose we observe a real-valued input variable x and wish to use this observation to predict the value of a real-valued target variable t. A polynomial function is used to fit the data, which is given by:

$$y\left(x,\omega\right)=w_0+w_1x+\ldots+w_Mx^M=\sum_{j=0}^{M}w_jx^j \tag{1}$$

Here, M is the order of the polynomial which is states the count of weights or parameters. And the polynomial of those weights of size $M+1$ is denoted by the vector denoted by w. Furthermore, the polynomial function for has imbibed in it a nonlinear function of training points which is denoted by vector x and a linear function of the weights w. The weights will now be further derived or determined by fitting the polynomial to the training data by minimizing an error function. The purpose of this is to minimize the error between the function for any given value of w and the training set data points. Now that the widely used error function is given by the sum of the squares of the errors between the predictions for each data point and the their respective target values , so that we minimize the following error function

$$\tilde{E}(w)=\frac{1}{2}\sum_{n=1}^{N}\left\{y(x,w)-t\right\}^2 \tag{2}$$

2 Approach

2.1 Data Generation

The data is generated randomly with the help of the function $y=sin(x/2)$. The values of x are uniformly spaced in the range $[1,4\pi]$ and represented in the vector form as $x=[x_1,x_2,\ldots,x_n]^T$. Further, using a Gaussian distribution for the already generated y,

to obtain the target vector. Now the target vector is represented as $t = [t_1, t_2, \ldots, t_n]^T$. This is all given as a started code for the project and is a MATLAB code named as generateData which is a matlab file with .m extension. This generator code uses number of points as a base and generates the required values or pints based on the N value. In this project I have used the values of N as 10 and 50.

2.2 Methods for Error Minimization

The main purpose of these methods is to derive the values for weights or coefficients of the polynomial so as to enhance the error minimization when considered the target output and the predicted output. The number of coefficients revels the order of the polynomial.

2.2.1 Direct Error Minimization

To evaluate the error between the function y and the target function with any w and the training data points say w here, based on this the error function equation is as follows:

$$\tilde{E}(w) = \frac{1}{2}\sum_{n=1}^{N}\{y(x,w) - t\}^2 \tag{3}$$

The above equation 3 can now be written in matrix equation form as follows:

$$\tilde{E}(w) = \frac{1}{2}\left[(Xw - t)^T(Xw - t)\right] \tag{4}$$

where X is a matrix of order $N \times (M+1)$ and the matrix is of the form:

$$X = \begin{bmatrix} x_1^0 & x_1^1 & x_1^2 & \ldots & x_1^M \\ x_2^0 & x_2^1 & x_2^2 & \ldots & x_2^M \\ & & \ldots\ldots\ldots & & \\ x_N^0 & x_N^1 & x_N^2 & \ldots & x_N^M \end{bmatrix}$$

Given that the error function 3 can be minimised with an ideal values of weights or coefficients w. This set of coefficients can be obtained when the original error function is now differentiated with respect to w and then equating it to zero ti find the new values of w called w optimised or $w*$. The step wise derivation for obtaining optimal w values is as shown below.

Derivation:
Now, the error function in equation 3 is differentiated to minimize the error at $w*$. Therefore, $\tilde{E}'(w^*)$ would be equated to zero to find the optimal $w*$.

$$\tilde{E}'(w) = \frac{\mathrm{d}}{\mathrm{d}w}\left\{\frac{1}{2}\left[(Xw - t)^T(Xw - t)\right]\right\} = 0 \tag{5}$$

Solving equation 5, we get,

$$\frac{\mathrm{d}}{\mathrm{d}w}\left\{\frac{1}{2}\left[(Xw - t)^T(Xw - t)\right]\right\} = 0 \tag{6}$$

$$\left[X^T X w^* - X^T t\right] = 0 \tag{7}$$

$$(X^T X) w^* = X^T t \tag{8}$$

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

$$y(x, w^*) = X(X^T X)^{-1} X^T t \tag{10}$$

2.2.2 Direct Error Minimization with Regularisation

In the method of direct error minimization without regularisation there is a high chance of the curve over-fitting as the order of the polynomial increases and may result in larger values of weights when trying to find optimal values. Furthermore, to control the problem of over-fitting and to avoid the results of large coefficients there is an ideal way called Regularisation. Regularisation in general means it penalizes the error function to prevent the coefficients from reaching large values by adding some penalty value to the original error function. Hence, after adding the penalty value to the error function, the error function 1 comes to the form of the following equation:

$$\tilde{E}(w) = \frac{1}{2} \sum_{n=1}^{N} \{y(x, w) - t\}^2 + \frac{\lambda}{2} \|w\|^2 \tag{11}$$

Furthermore, the matrix equation form of 1 is

$$\tilde{E}(w) = \frac{1}{2} \left[(Xw - t)^T (Xw - t)\right] + \frac{\lambda}{2} w^T w \tag{12}$$

Now, to find the ideal value for vector w, same as direct error minimization method the equation 12 is differentiated with respect to w and then equated to zero. **Derivation:** Let $\tilde{E}(w)$ be minimum at w^*. Thus $\tilde{E}'(w) = 0$.

$$\tilde{E}'(w) = \frac{\mathrm{d}}{\mathrm{d}w} \left\{ \frac{1}{2} \left[(Xw - t)^T (Xw - t)\right] + \frac{\lambda}{2} w^T w \right\} = 0 \tag{13}$$

Solving equation 13, we get

$$\frac{\mathrm{d}}{\mathrm{d}w} \left\{ \frac{1}{2} \left[(Xw - t)^T (Xw - t)\right] + \frac{\lambda}{2} w^T w \right\} = 0 \tag{14}$$

$$\left[X^T X w^* - X^T t\right] + \lambda w^* = 0 \tag{15}$$

$$(X^T X + \lambda I) w^* = X^T t \tag{16}$$

$$w^* = (X^T X + \lambda I)^{-1} X^T t \tag{17}$$

$$y(x, w^*) = X(X^T X + \lambda I)^{-1} X^T t \tag{18}$$

2.2.3 Maximum Likelihood Estimator (MLE) - A Bayesian Approach

The direct error minimization approach appears to solve the curve fitting problem but it has appealed largely to intuition and we need a more principled approach using probability theory. As we have discussed, there is an uncertainty in the given target vector used for training. Using probability theory, we can quantify this uncertainty.
For the curve fitting problem, we have an observed data, $D = \{t_1, t_2, \ldots, t_n\}$. We record the assumptions for the coefficients w before observing the data and call it the prior probability distribution $p(w)$. The effect of the observed data D is expressed through the conditional probability $p(D|w)$. From the Bayes theorem we know the formula of Bayes equation is

$$p(w|D) = \frac{p(D|w)p(w)}{p(D)} \tag{19}$$

which can be explained in the verbal notation as *posterior* $\propto$ *likelihood* $\times$ *prior*, where $p(D)$ can be ignored as a normalization term. We now have to find w to the values that that maximizes the likelihood because maximizing the likelihood will minimize the error as they are inversely proportionate to each other.
Now, assume the likelihood to be a Normal Distribution function. And for the problem of curve-fitting with N input points $x = [x_1, x_2, \ldots, x_n]^T$ and corresponding target variable, $t = [t_1, t_2, \ldots, t_n]^T$, the likelihood can be written as a Gaussian distribution with mean equal to $y(x, w)$ and standard deviation, β^{-1}.And the equation of Gaussian distribution is

$$p(t|x, w, \beta) = \mathcal{N}(t_n|y(x_n, w), \beta^{-1}) \tag{20}$$

We have to keep in mind that the data which we consider should be independent from the distribution. Now, the likelihood function for say N data point will be

$$p(\mathbf{t}|\mathbf{x}, w, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n|y(x_n, w), \beta^{-1}) \tag{21}$$

$$p(\mathbf{t}|\mathbf{x}, w, \beta) = \prod_{n=1}^{N} \frac{1}{(2\pi\beta^{-1})^{\frac{1}{2}}} exp\left\{\frac{-\beta}{2}(y(x_n, w) - t_n)^2\right\} \tag{22}$$

Applying logarithm on both the sides, we get

$$\ln p(\mathbf{t}|\mathbf{x}, w, \beta) = -\frac{\beta}{2}\sum_{n=1}^{N}(y(x_n, w) - t_n)^2 + \frac{N}{2}\ln\beta - \frac{N}{2}\ln 2\pi \tag{23}$$

By minimizing the negative logarithm of the likelihood, we can find the optimal w and β where the likelihood is maximum. And then differentiate of equation 9 with respect to w. We ignore the β terms and assume $\frac{\beta}{2}$ as $\frac{1}{2}$ as it does not affect the likelihood.

$$\frac{1}{2}\frac{\partial}{\partial w}\sum_{n=1}^{N}(y(x_n, w) - t_n)^2 = 0 \tag{24}$$

Using the matrix notation for X,

$$\frac{1}{2}\frac{\partial}{\partial w}\left[(Xw - t)^T(Xw - t)\right] = 0 \tag{25}$$

$$\frac{1}{2}\frac{\partial}{\partial w}\left[w^T X^T X w - 2w^T X^T t - t^T t\right] = 0 \tag{26}$$

$$\left[X^T X w^* - X^T t\right] = 0 \tag{27}$$

$$w^* = (X^T X)^{-1} X^T t \tag{28}$$

Thus, we obtain $w*$. Differentiating w.r.t. β,

$$\frac{\mathrm{d}}{\mathrm{d}\beta}\left\{\frac{\beta}{2}\sum_{n=1}^{N}(y(x_n, w) - t_n)^2 - \frac{N}{2}\ln\beta + \frac{N}{2}\ln 2\pi\right\} = 0 \tag{29}$$

$$\frac{1}{\beta_{ML}} = \frac{1}{N}\sum_{n=1}^{N}(y(x_n, w) - t_n)^2 \tag{30}$$

Finally, we obtain the standard deviation by $1/\beta = variance$ and the error for the curve would be of the form

$$\sqrt{((1/N)((Xt - t)^T)(XW * -t))} \tag{31}$$

2.2.4 Maximum Posterior Estimator (MAP) - A Bayesian Approach

In this method we assume prior distribution as a Gaussian distribution and now the distribution equation looks like

$$p(w|\alpha) = \mathcal{N}(w|0, \alpha^{-1}I) = \left(\frac{\alpha}{2\pi}\right)^{\frac{(M+1)}{2}} exp\left\{-\frac{\alpha}{2}w^T w\right\} \tag{32}$$

From equation 19, we can see that $posterior \propto likelihood \times prior$.

$$p(w|x, t, \alpha, \beta) \propto p(t|x, w, \beta)p(w|\alpha) \tag{33}$$

The likelihood function is as in equation 22. Substitute equations 22 and 32 in equation 33:

$$p(w|x, t, \alpha, \beta) \propto \prod_{n=1}^{N}\frac{1}{(2\pi\beta^{-1})^{\frac{1}{2}}} exp\left\{\frac{-\beta}{2}(y(x_n, w) - t_n)^2\right\}\left(\frac{\alpha}{2\pi}\right)^{\frac{(M+1)}{2}} exp\{-\frac{\alpha}{2}w^T w\} \tag{34}$$

Applying logarithm on both sides.

$$-\ln p(w|x, t, \alpha, \beta) = \frac{\beta}{2}\sum_{n=1}^{N}(y(x_n, w) - t_n)^2 + \frac{\alpha}{2}w^T w \tag{35}$$

By minimizing the negative logarithm of the likelihood, we can find the optimal w.

$$\tilde{E}(w) = \frac{\beta}{2}\left[(Xw - t)^T(Xw - t)\right] + \frac{\alpha}{2}w^T w \tag{36}$$

$$\tilde{E}(w) = \frac{\beta}{2}\left[w^T X^T X w - w^T X^T t - t^T X w - t^T t\right] + \frac{\alpha}{2}w^T w \tag{37}$$

$$\tilde{E}(w) = \frac{\beta}{2}\left[w^T X^T X w - 2w^T X^T t - t^T t\right] + \frac{\alpha}{2} w^T w \tag{38}$$

Differentiating equation 38 w.r.t. w,

$$\frac{\mathrm{d}}{\mathrm{d}x}\tilde{E}(w) = \frac{\partial}{\partial w}\left\{\frac{\beta}{2}\left[w^T X^T X w - 2w^T X^T t - t^T t\right] + \frac{\alpha}{2} w^T w\right\} \tag{39}$$

$$\beta\left[X^T X w^* - X^T t\right] + \alpha w^* = 0 \tag{40}$$

$$\left[\beta X^T X + \alpha I\right] w^* = X^T t \beta \tag{41}$$

$$\left[X^T X + \frac{\alpha}{\beta} I\right] w^* = X^T t \tag{42}$$

$$w^* = \left(X^T X + \frac{\alpha}{\beta} I\right)^{-1} X^T t \tag{43}$$

$$y(x_n, w^*) = X w^* = X\left(X^T X + \frac{\alpha}{\beta} I\right)^{-1} X^T t \tag{44}$$

Thus, we obtain the optimal coefficients where the posterior is maximum.

3 Results with Comparisons

The results are obtained through the implementation of the four approaches discussed in MATLAB to experiment with sample data set. In this section, I shall discuss the results obtained using each of the approach and compare them following the comparison patterns from Reference book [1]. In each of the plot shown, the green line indicates the desired output, the red line indicates the predicted output and the blue dots indicate the noisy target values. And tables for the values of optimised w will help us know the fluctuation in w values if the order is increased and can be inhered from the tables that as we increase the order from 0, the coefficients have a huge difference of w values at m=0 and as we go on increase in value of m the values again get to the stage where we again visualise the huge fluctuations of values. Therefore, the value of order much not be too large or too small for the polynomial to the optimal values for w.

3.1 Basic Error Minimization

As discussed in section 2.2.1, the misfit between the polynomial function and the target values is minimized to find the optimal w. The plots in figure 1 show the predicted curves for 10 data samples with varying order of the predicted polynomial $M = \{0, 1, 3, 5, 7, 9\}$. It can be seen that, with increase in M, the predicted curve is trying to over-fit the data points. For M=9, there is perfect over-fitting. M=3 gives a descent output, again with the reference of the book [1]. Table **??** shows the variation in the values of w^* with increasing M. It can be seen that the values of $w*$ increase with the increase in M.

The table 1 shows us that the values of optimised values of w are varying with high fluctuation, as we can see that for the order 7 and 9 the fluctuation is not prominent and the error rate can be uncertain in this case. Based on the plots of figure 1 the over-fitting arises when there is increase in m value.
Similarly, the table 2 shows us that the values of optimised values of w are varying but not with high fluctuation as the 1 does. For instance we can see that for the order 7 and 9 the fluctuation 2 is not having a huge difference in w values as 1 and the and the error rate can be certain and the optimal values of w can be obtained with the order 5. But when figure 2 is compared to 1 it shows that figure 2 with increase in value for order will give over-fitting but not as much as the plots for hgher order value in figure 1. Furthermore, the experiment enhances the scenario that the value of λ involving as a penalty of Basic Error Minimization will reduce the effects of over-fitting and can be compared with the results based on plots of figure 4 and table 3.

	M=0	M=1	M=3	M=5	M=7	M=9
w_0^*	0.0396	1.1078	-0.1666	1.6712	8.4959	14.7798
w_1^*		-0.1575	1.0087	-1.7374	-15.0491	-29.7711
w_2^*			-0.2318	0.9990	9.9731	23.0394
w_3^*			0.0122	-0.2164	-3.0989	-9.0902
w_4^*				0.0186	0.5139	2.1168
w_5^*				-0.0005	-0.0473	-0.3113
w_6^*					0.0023	0.0294
w_7^*					-0.0000	-0.0017
w_8^*						0.0001
w_9^*						-0.0000

Table 1: Variation in the values of $w*$ in Basic Error Minimization approach for increasing values of $M = 0, 1, 3, 5, 7, 9,$. No of samples N = 10.

	M=0	M=1	M=3	M=5	M=7	M=9
w_0^*	-0.0358	1.2690	-0.1048	0.6192	2.6140	1.6151
w_1^*		-0.1924	0.9589	0.0849	-3.7987	-1.5585
w_2^*			-0.2279	0.1119	2.8837	0.9527
w_3^*			0.0122	-0.0446	-1.0076	-0.1545
w_4^*				0.0043	0.1844	-0.0319
w_5^*				-0.0001	-0.0186	0.0144
w_6^*					0.0010	-0.0021
w_7^*					-0.0000	0.0001
w_8^*						-0.0000
w_9^*						0.0000

Table 2: Variation in the values of $w*$ in Basic Error Minimization approach for increasing values of $M = 0, 1, 3, 5, 7, 9,$. No of samples N = 10.

From the figure 1 the plots make us visualise that there is a bad curve fitting for less order like m=0,1 and there is over-fitting as the value of order get increased till starting from 9. The best fit lies for the order values of 3 till 7. As the value of number of points is fixed for all the orders which is 10 here shows more signs of over-fitting compared to the high values of N from the plots of figure 2.

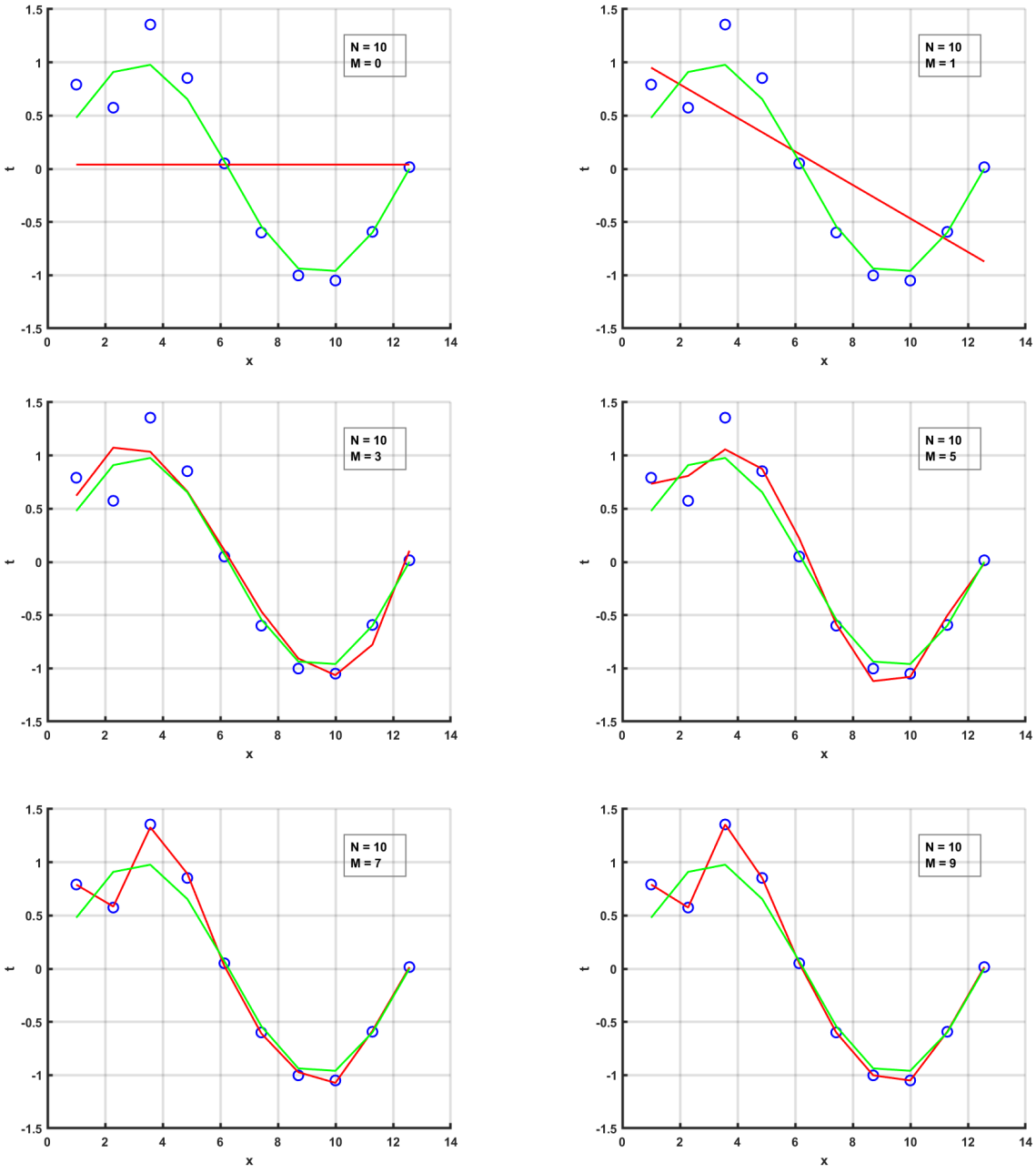

Figure 1: Plots for the predicted output(red) w.r.t. the desired output(green) and the given target output(blue) using the direct error minimization technique for 10 data points. The plots are shown for varying values of $M = \{0, 1, 3, 5, 7, 9\}$

From the figure 2 the plots make us visualise that there is a bad curve fitting for less order like m=0,1 and there is over-fitting as the value of order get increased till starting from 9. The best fit lies for the order values of 3 till 7. Compared to plots from figure 1 as the value of N increases from 10 to 50 there is a possibility for curve over-fitting but with less uncertainty of over-fitting compared to 1, the predicted curve is a way smooth when compared.

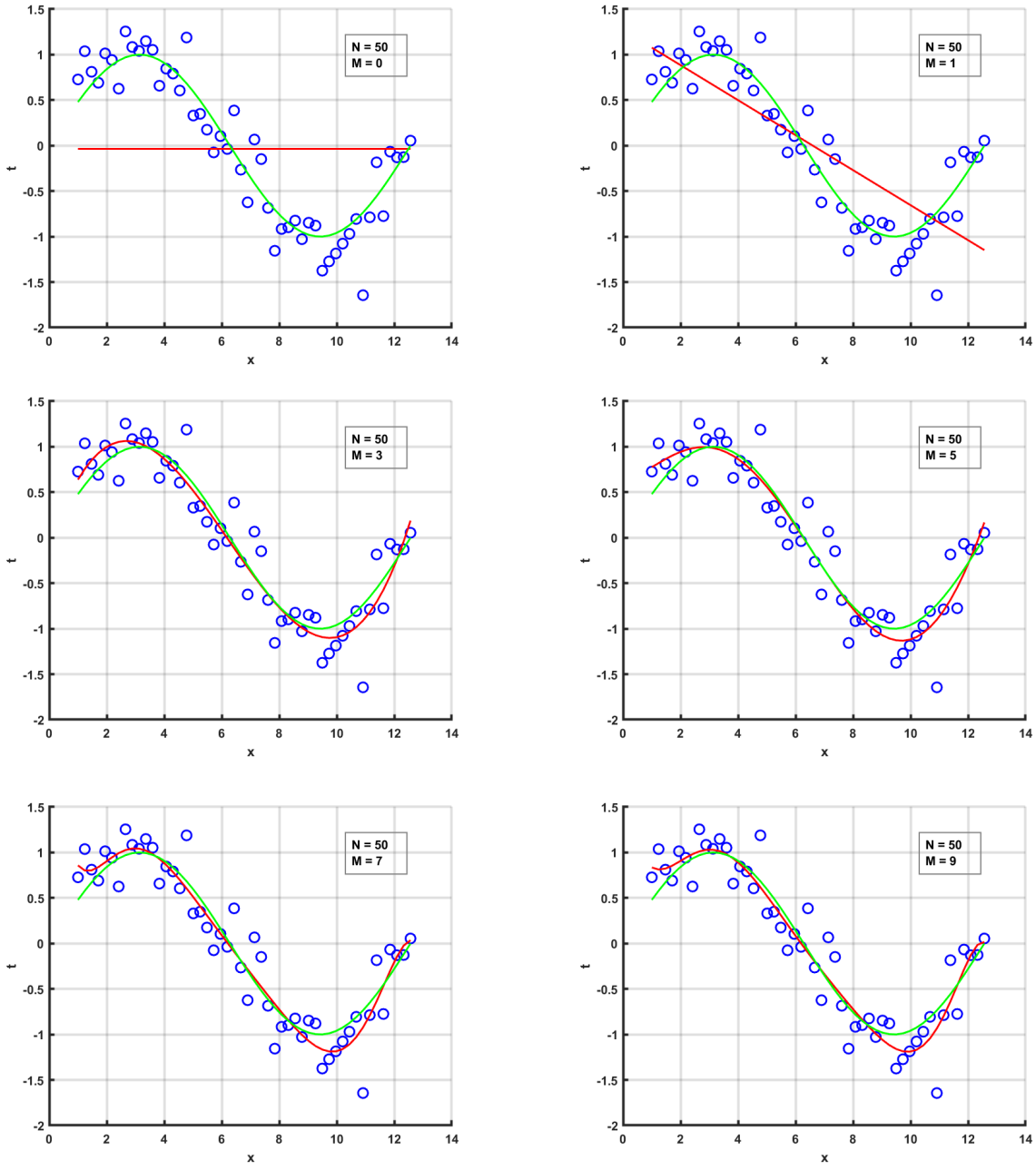

Figure 2: Plots for the predicted output(red) w.r.t. the desired output(green) and the given target output(blue) using the direct error minimization technique for 50 data points. The plots are shown for varying values of $M = \{0, 1, 3, 5, 7, 9\}$

Comparison of direct estimation for small and large N:

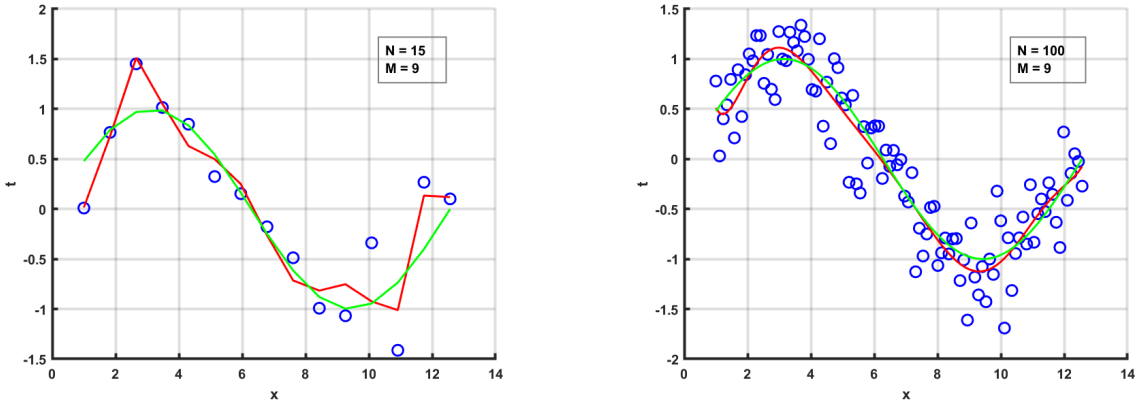

Figure 3: Comparison plots for the predicted output(red) w.r.t. the desired output(green) and the given target output(blue) using the direct error minimization technique for 15 and 100 data points which indicated small and large range of N comparison. The plots are shown for fixed value of $M = 9$

3.2 Basic Error Minimization with Regularization

As we have already discussed in section 2.2.2, the regularization term penalizes the error minimization technique to discourage over-fitting and larger values of the coefficients $w*$. The plots in figure **??** show the predicted output with different values of the regularization term, $ln\lambda = \{-18, -15, -13, ln3\}$ and $M = 9$. From my experiments, I find $\lambda = 3$ is an appropriate penalty term for error minimization as discussed on Piazza. It reduces the over-fitting effect for the higher-dimension M=9.
But for the given values of $ln\lambda = \{-18, -15, -13\}$, I could not visualize much variation in the plots. Table 3 shows the variation in the values of the coefficient for different values of $ln\lambda$. It can be seen that for $ln\lambda = ln3$, the coefficients have descent values and are better than compared to the reference book [1] values.

	$\ln\lambda = -18$	$\ln\lambda = -15$	$\ln\lambda = -13$	$\ln\lambda = ln3$
w_0^*	14.6813	12.8174	7.9757	0.1208
w_1^*	-29.5317	-25.0020	-13.2386	0.0884
w_2^*	22.8155	18.5773	7.5742	0.0511
w_3^*	-8.9806	-6.9054	-1.5197	0.0263
w_4^*	2.0852	1.4863	-0.0675	0.0194
w_5^*	-0.3056	-0.1984	0.0798	-0.0174
w_6^*	0.0288	0.0167	-0.0145	0.0042
w_7^*	-0.0017	-0.0009	0.0013	-0.0005
w_8^*	0.0001	0.0000	-0.0001	0.0000
w_8^*	-0.0000	-0.0000	0.0000	-0.0000

Table 3: Variation in the values of $w*$ upon regularizing the above error minimization method. The results are shown for different values of $ln\lambda = \{-18, -15, -13, ln3\}$. M=9, N=10

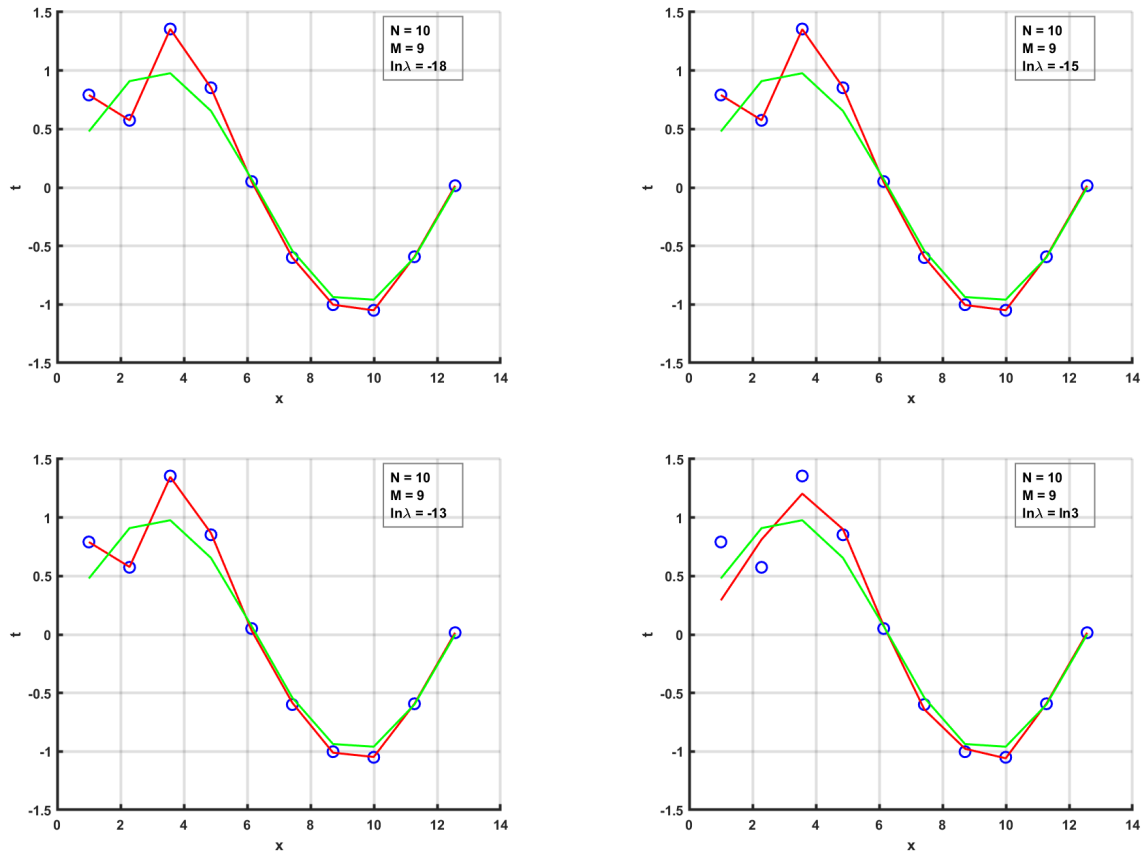

Figure 4: Plots of the predicted output (red) w.r.t the desired output (green) and the given target output (blue) upon regularizing the error minimization technique for N=10 data points. The plots are shown for varying values of the regularizing term $ln\lambda = \{-18, -15, -13, ln3\}$. for a fixed parameter of order 9 and number of data points 10

Plots from the figure 4 shows that for the value of $\ln\lambda = \ln 3$ the curve fits good. That means there is rival of original output curve matching the predicted curve. And for all the other values of $\ln\lambda$ except $\ln\lambda = \ln 3$, I visualised no difference as there is no change in a way the graph looks. Here, as the order is contact to 9 the difference in curve should solely depend on the value of $\ln\lambda$ and this can be seen from the values of w values which are optimised and can be seen from table 3. It is clear from the table 3 that for all the values of $\ln\lambda$ other than $\ln\lambda = \ln 3$, there is no certainty in the values of w and the range of uncertainty is very high compared to the uncertainty from the column of $\ln\lambda = \ln 3$. According to [1] the values for $\ln\lambda$ varies from $-$ till zero and it states clearly that the values around $\ln\lambda = -18$ will give the good fit for curve and from our experiment also it is clear that for values of $\ln\lambda = -13, -15, -18$ the curve is same and fits equally with minimal difference of w values. Furthermore, for the positive value of $\lambda = 3$ (value delivered in piazza) the curve fits very good than the value $\ln\lambda = -18$ from the reference book [1].

3.3 Maximum Likelihood Estimator (MLE)

From the section 2.2.3 the values of w which are optimised are derived and they are now used to calculate the beta value after differentiating the polynomial w.r.t. β. Using β as the main key to find the error, error is now again valuated same as equation 31 The plots in figure 5 show the predicted output curves using maximum likelihood estimation for increasing values of $M = \{0, 1, 3, 5, 7, 9\}$ for N=10 sample data points with the calculated error using the error plot for error, resulting in the good fit for not too small or large value of order. And it can be seen that with increasing M, the output tends to over-fit.

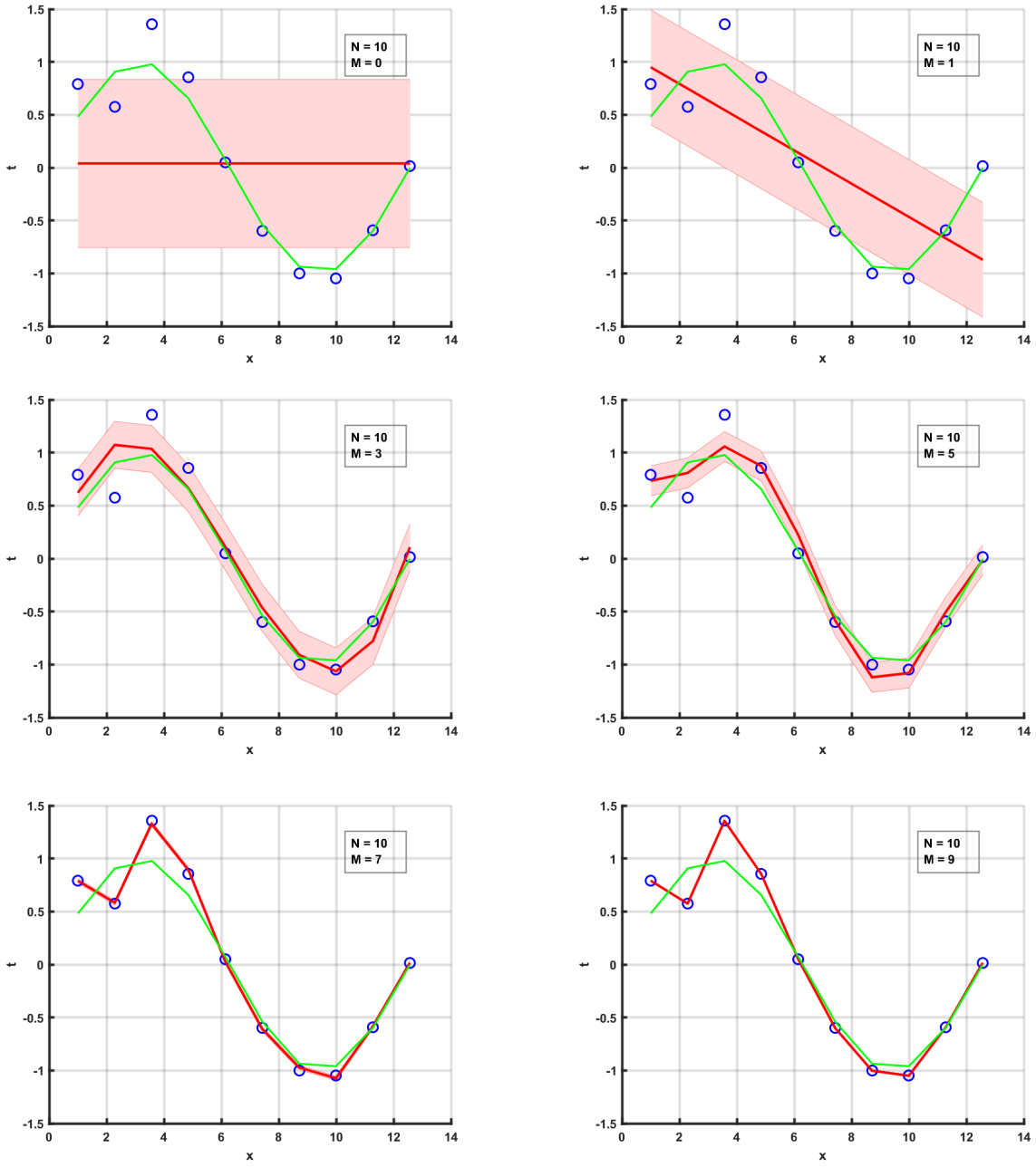

Figure 5: Plots for the predicted output(red) w.r.t. the desired output(green) and the given target output(blue) using the direct error minimization technique for 10 data points. The plots are shown for varying values of $M = \{0, 1, 3, 5, 7, 9\}$

3.4 Maximum Posterior Estimator (MAP)

From the section 2.2.4 the values of w which are optimised are derived and they are now used to calculate the beta value after differentiating the polynomial w.r.t. β. Using β as the main key to find the error, error is now again valuated same as equation 31. The plots in figure 6 show the predicted output curves using maximum likelihood estimation for increasing values of $M = \{0, 1, 3, 5, 7, 9\}$ for N=10 sample data points for the calculated error and keeping $\alpha = 0.005$ and $\beta = 11.1$ as fixed values. From the 5 it is clear that the curve fits better for not too large or small values of order. And it can be seen that with increasing M, the output tends to over-fit.

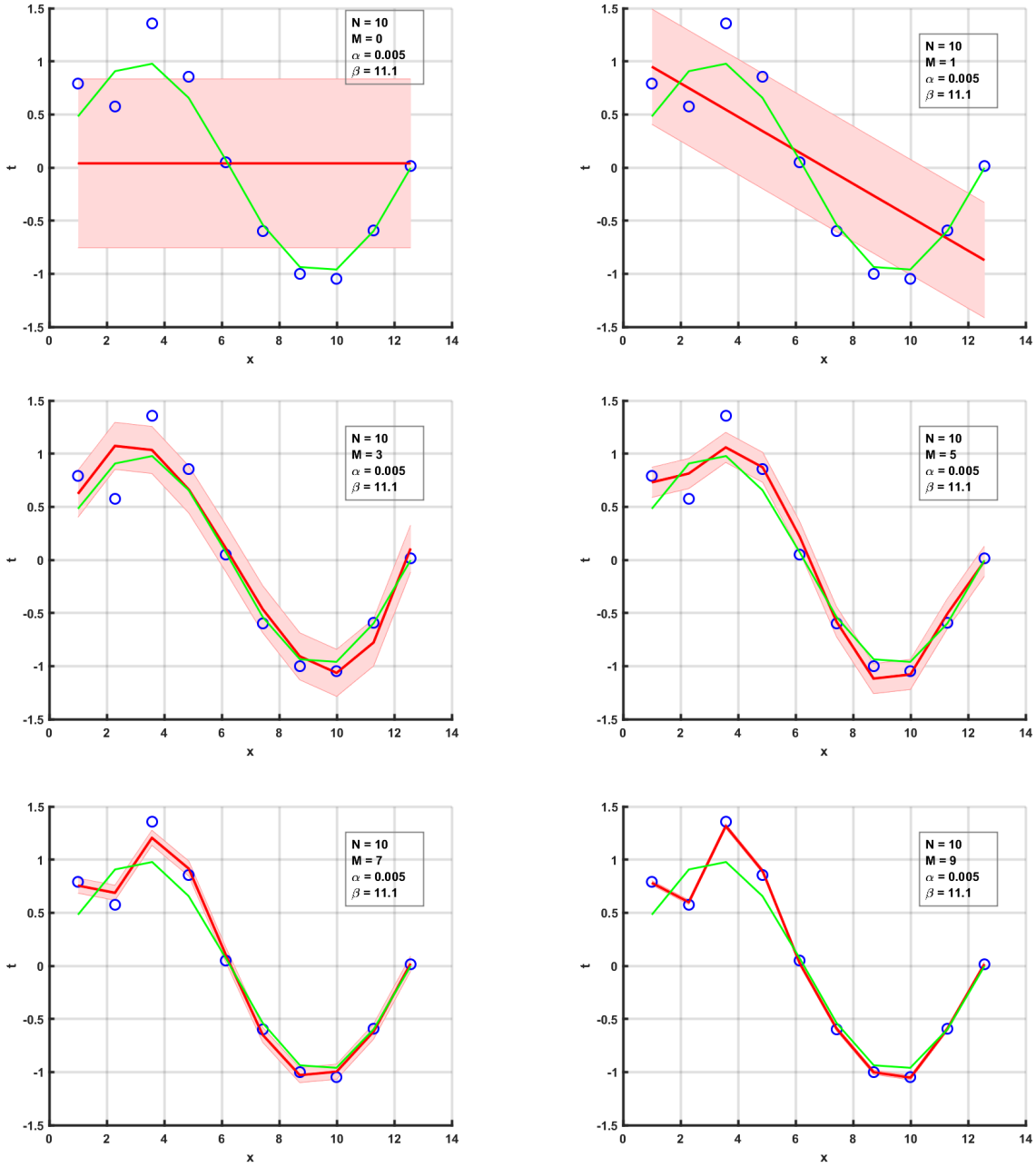

Figure 6: Plots for the predicted output(red) w.r.t. the desired output(green) and the given target output(blue) using the direct error minimization technique for 10 data points. The plots are shown for varying values of $M = \{0, 1, 3, 5, 7, 9\}$

Comparison of MLE and MAP for a fixed N = 50:
From the figure 7, I can visualise that the range of shaded error region around the predicted output curve is same for both graphs and I could enhance that predicted curve for MLE is as smooth as the predicted curve for MAP (here, value of N is fixed to be 50 in both the approaches).

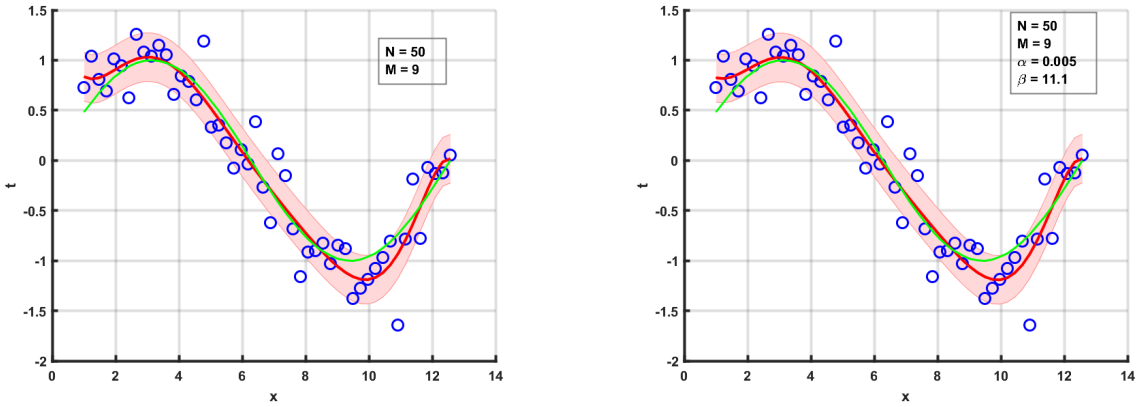

Figure 7: Comparison plots for the predicted output(red) w.r.t. the desired output(green) and the given target output(blue) using the direct error minimization technique for fixed data points say N = 50 for comparison. The plots are shown for fixed value of $M = 9$

4 Conclusion

- From all the four approaches of the error minimization, it is clear from the experiments that as we penalise the Basic Error function with some penalty factors the values obtained for the w are optimised more. Furthermore, the values of optimised values for w are very uncertain for high values of order and too low values of order for polynomial.
- For Basic Error Minimization approach the curve fits good for order 3 in both N=10 and N=50 cases. Where as when regularisation term is used the $\lambda = 3$ value gives the good fit and over comes the over-fitting problem.
- Coming to Bayesian approaches, MLE and MAP, the bayesian approaches holds the positivity to know the error range and comparing MLE and MAP, the MAP gives as soft as MLE accuracy of error around the predicted output curve. The target data used Gaussian Distribution to give better performance while curve fitting.

References

[1] C. Bishop, *Pattern Recognition and Machine Learning*. Springer-Verlag New York, 2006. 7, 11, 12