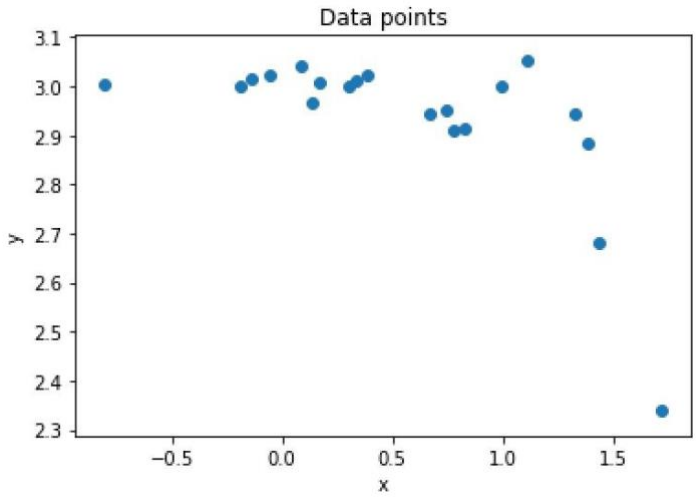
Report

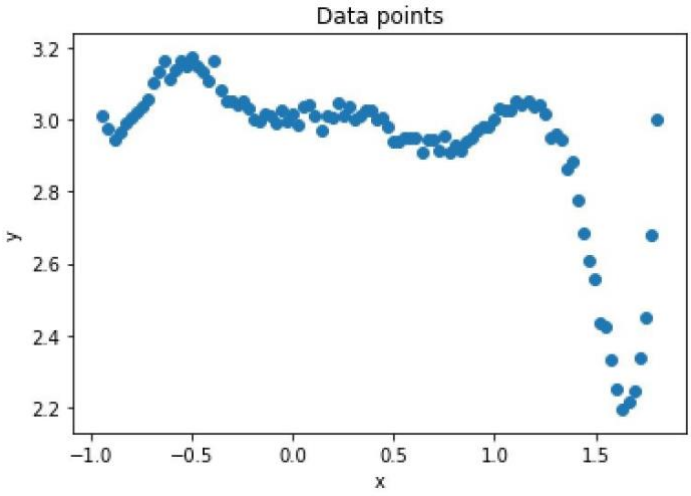
Ishank Pahwa

Finding the degree of polynomial-

Scatter Plot for 1st 20 points-



Scatter Plot for 100 points-



Moore-Penrose pseduoinverse- Without regularization-

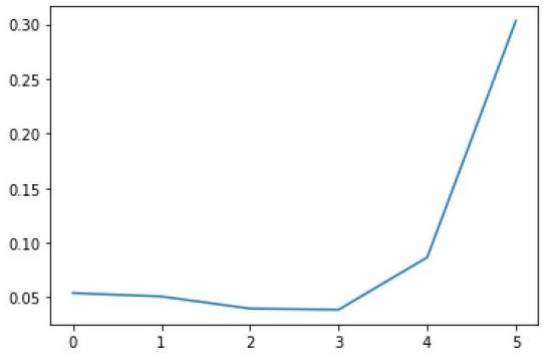
Calculate the hyper-parameter M (Degree of polynomial)

Used k fold cross validation, to calculate the mean squared error on validation set obtained after k fold cross validation.

Number of folds = 4

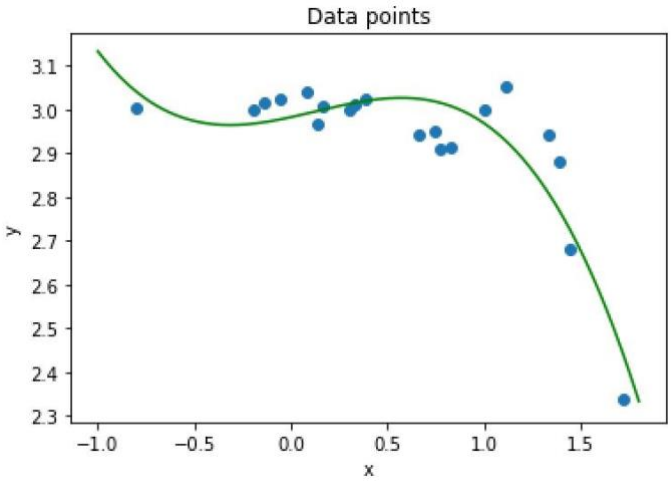
Degree of polynomial was varied from 1 to 5

Data Size=20



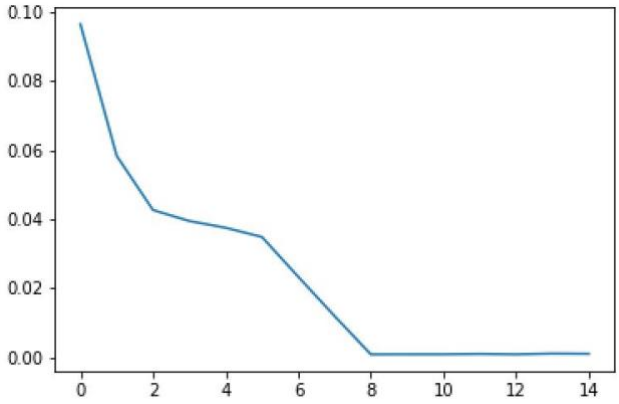
Degree of polynomial vs Error

Hence optimal value of M comes out to be 3



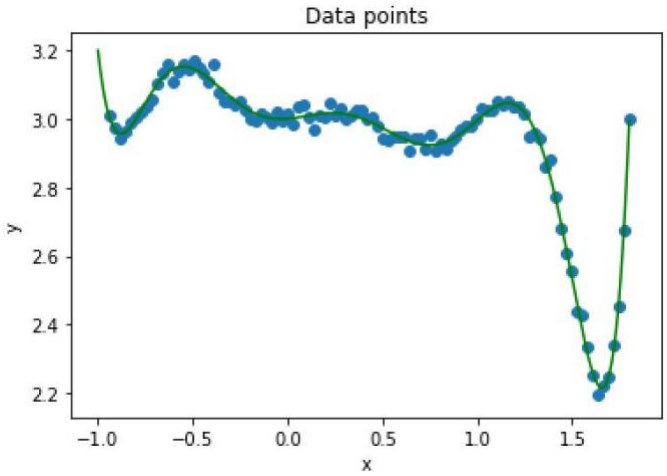
Curve after fitting the polynomial with the given parameters

Data Size =100



Degree of polynomial vs Error

Hence optimal value of M comes out to be 8



Curve after fitting the polynomial with the given parameters

With regularization-

Calculated 2 hyper-parameters

1. Degree of polynomial (M)
2. Regularization parameter

Varied the value of lambda(regularization parameter) and saw the possible values which could be the optimum values, to minimize the error.

Lambda varied from 0-0.1 (100 values)
Degree of polynomial was varied from 1 to 20

Performed a grid search by varying both lambda and M and found the minimum value of validation error.

Data Size = 20

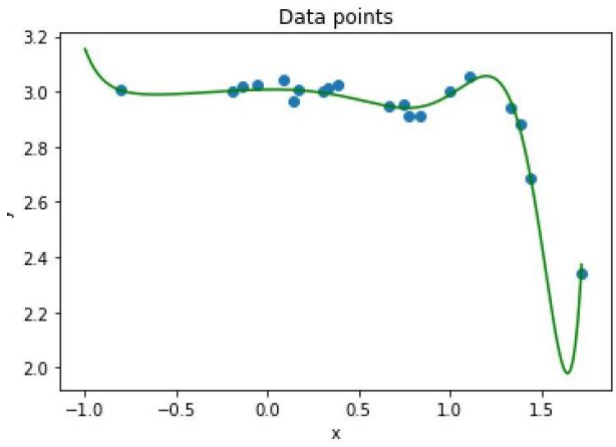
Optimal values-

1. lambda=0.027
2. M(degree of polynomial) = 12

Total error on cross validation set = 0.0019

	0
0	3.00725
1	0.00705778
2	−0.115387
3	−0.133067
4	−0.00477729
5	0.061978
6	0.126378
7	0.109724
8	0.0718796
9	−0.0180684
10	−0.0911635
11	−0.11023
12	0.0772965

Values of Parameters obtained with the given conditions



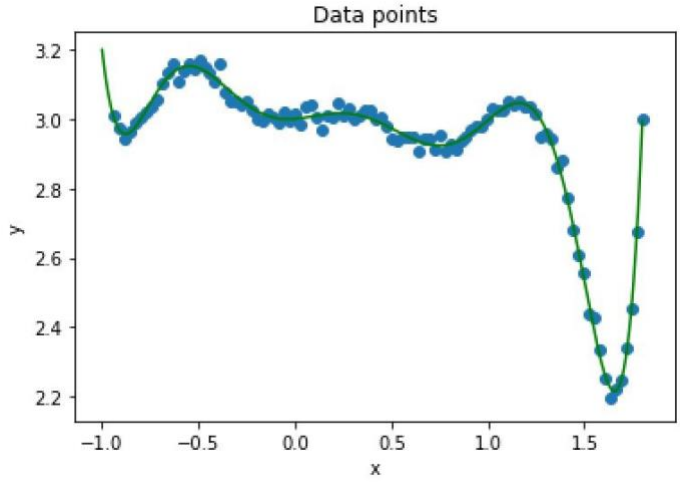
Curve after fitting the polynomial with the given parameters

Data Points =100

Optimal Values- 1.
lambda=0
2. M(degree of polynomial) = 8
Total error on cross validation set = 0.00076

	0
0	3.00155
1	0.0448884
2	0.531627
3	−1.75683
4	−1.18102
5	4.15129
6	−0.205374
7	−2.54125
8	0.952432

Values of Parameters obtained with the given conditions



Curve after fitting the polynomial with the given parameters

Gradient Descent

Data points=20

Calculated 2 hyper-parameters

1. Degree of polynomial (M)
2. Regularization parameter

Varied the value of lambda(regularization parameter) and saw the possible values which could be the optimum values, to minimize the error.

Lambda varied from 0-0.05(5 values)
Degree of polynomial was varied from 1 to 15

Performed a grid search by varying both lambda and M and found the minimum value of validation error.

Exit criteria was varied by taking different conditions such as

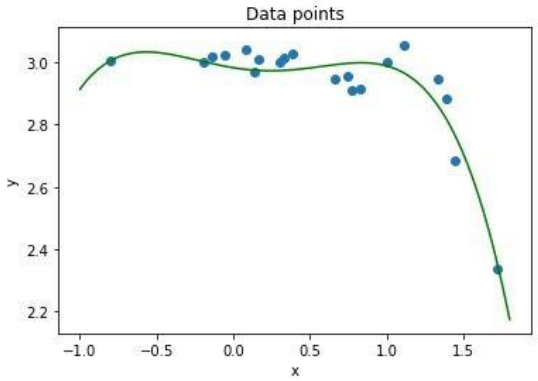
1. Number of epochs used should be less than 5,00,000
2. Relative change in error should be less than 10^(-9) to exit
3. Change in value of parameters should be less than 10^(-9) to exit

Alpha(Learning Rate) can be made time dependent by changing its value to half when there is an increase in error in a given epoch

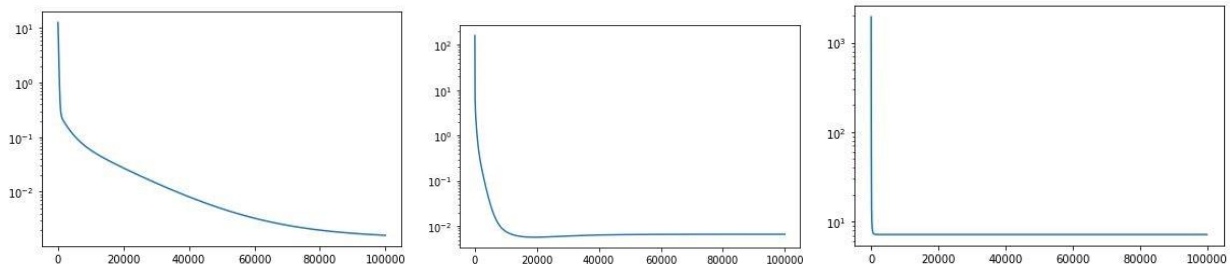
Gradient descent was used (batch size=20)(with time dependent alpha(learning rate)) to calculate the hyperparameters which came out as follows
Optimal Values- 1.
lambda=0.05
2. M(degree of polynomial) = 4
Total error on cross validation set = 0.00188

	0	1	2	3	4
0	0.0102855	0.0068088	0.021017	0.0271073	0.167046
1	0.0154526	0.0219268	0.0269412	0.00635715	0.029805
2	0.0101486	0.0109695	0.0119706	0.00191364	0.00326945
3	0.0128328	0.0143044	0.00559864	0.00203472	0.00389296
4	0.0111156	0.0106062	0.0133662	0.00188513	0.00372922

Grid search



Curve after fitting the polynomial with the given parameters
Batch size=20



Batch size=20 Batch size=10 Batch size=1

Error vs epochs for 500000 epochs with optimal hyper-parameters

Gradient descent was used with batch size=10(with time dependent alpha(learning rate)) to calculate the hyperparameters which came out as follows

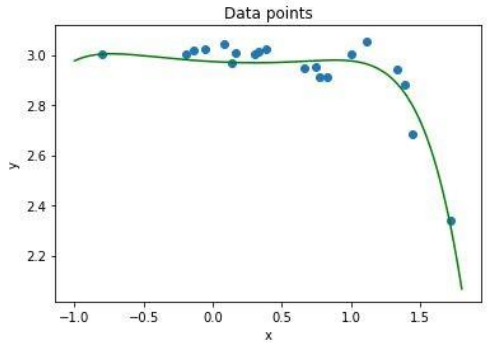
Optimal Values-

1. lambda=0.044
2. M(degree of polynomial) = 6

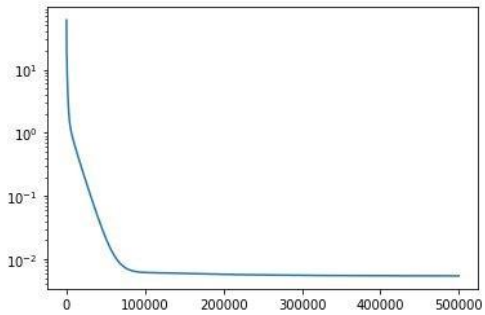
Total error on cross validation set = 0.00599

1	2	3	4	5	6	7
0.027745	0.0096697	0.0425553	0.192101	0.632999	4.9416	0.262898
0.0214982	0.0308947	0.0233776	0.342078	0.125622	0.345286	0.0403857
0.0332018	0.0145412	0.0485905	0.00927636	0.0303888	0.133958	0.28563
0.026611	0.0106272	0.00695985	0.00741967	0.0536278	0.0839989	0.291958
0.0168934	0.0227257	0.00645468	0.00697397	0.00599372	0.0752643	0.0337513
0.0302838	0.012119	0.00614482	0.0151402	0.0198998	0.0352913	0.0653272

Grid search



Curve after fitting the polynomial with the given parameters
Batch size=10



Epochs vs error for batch size=10

Comparing results for different batch sizes

Batch Size	Degree of polynomial	Lambda
20	4	0.05
10	6	0.044

Data points=100

Calculated 2 hyper-parameters

3. Degree of polynomial (M)
4. Regularization parameter

Batch size=100 with time dependent alpha(learning rate)

	0	1	2	3	4	5	6	7
0	0.0147106	0.00993901	0.0100027	0.0087515	0.00891851	0.00844023	0.00498189	0.00142318
1	0.0139186	0.00980866	0.00944102	0.00854752	0.00827005	0.0109156	0.00648898	0.00255698
2	0.0146787	0.00995014	0.0108414	0.0100393	0.00825721	0.00631955	0.00740368	0.00346673
3	0.0137177	0.0101418	0.00945097	0.00873661	0.00978487	0.00846535	0.00367839	0.0022807
4	0.0140662	0.00945256	0.00942817	0.0100946	0.0139941	0.00656466	0.00361248	0.00742279

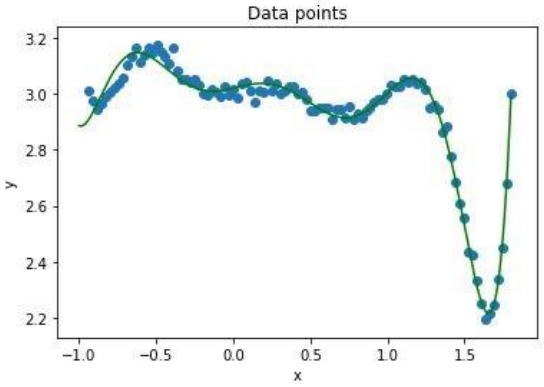
	0
0	2.98699
1	−0.0117481
2	0.802579
3	−1.62184
4	−1.96427
5	4.40108
6	0.318814
7	−2.95256
8	1.03682

Grid search and values of parameters

Optimal Values- 1.

Degree 8

2. lambda=0



Curve after fitting the polynomial with the given parameters
Batch size=100

Batch Size	Training Error
100	0.000407
50	0.00096
1	0.00194

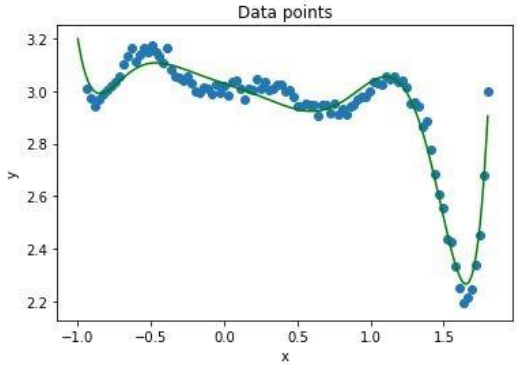
Batch size=50

		4	5	6	7	8
0	2	0.0373685	0.0369941	0.0107451	0.0106681	0.00815158
1	9	0.0304551	0.0253601	0.0157779	0.0526988	0.0222273
2	3	0.0258056	0.0339025	0.03014	0.0142145	0.0382413
3	4	0.0209302	0.0201768	0.0130319	0.0174542	0.0149424
4	6	0.0216318	0.0211332	0.0248071	0.0124808	0.0101372
5	1	0.0203396	0.0186346	0.016477	0.00652279	0.0207664
6	3	0.0194452	0.0324334	0.0128978	0.00468749	0.0135429
7	8	0.0205493	0.0122066	0.0116155	0.0100031	0.00586057
8	9	0.0186143	0.0118892	0.0151054	0.0232505	0.00764387
9	8	0.0200306	0.0206391	0.00878173	0.0105848	0.00497071

Grid Search

Optimal Values

1. Degree 8
2. Lambda=0.066



Curve after fitting the polynomial with the given parameters
Batch size=50

Comparing results of different batch size

Batch Size	Degree of polynomial	Lambda
100	8	0
50	8	0.066

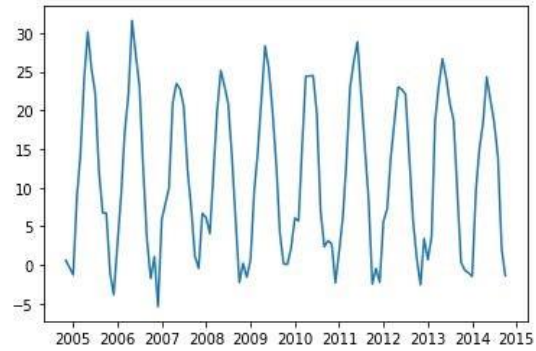
Conclusion

When using only 20 data points the polynomial degree comes out different in different case (in analytical and gradient descent) as data is too less to find the hyperparameter using validation set. It is difficult to predict the degree of a polynomial with only these many data points. Using 100 data points degrees comes out as 8 which should be the answer

Optimal polynomial comes out to be 8 as both analytical solution and gradient descent for 100 data points comes out to be 8 which when plotted seems good. We do not see similar results coming from 20 data points as data is too less to predict correct degree of polynomial.

Predicting Time Series Data-

Data points-



Formed a design matrix taking sin(ix)*b and cos(ix)*b by varying the values of i and b and calculating the hyper-parameter by using k fold cross validation using gradient descent. Number of folds used = 10

Gradient Descent-

Performed a grid search to find the optimal values of i and b

```
8.0 8 126.7834464523611
12.0 2 7.119578177885932
12.0 3 7.1095669915828745
12.0 4 5.873376719807216
12.0 5 5.687413971369962
12.0 6 5.623339127012022
12.0 7 5.560382696408817
12.0 8 5.489748543898168
16.0 2 100.57711702774972
16.0 3 96.54813183358766
16.0 4 96.50006490354019
16.0 5 96.18055487949665
16.0 6 96.12041636078092
16.0 7 95.87887371077395
16.0 8 95.81414498187965
20.0 2 101.47101370320178
20.0 3 101.44776902002619
20.0 4 101.44383327180077
20.0 5 101.36159929506394
20.0 6 101.24632764677827
20.0 7 100.98091633055257
20.0 8 100.50221012057834
24.0 2 101.20700541720005
24.0 3 100.20772530454836
24.0 4 6.973553628804488
24.0 5 6.717953421908216
24.0 6 6.704294227250528
24.0 7 6.349117667340624
24.0 8 5.116474580144971
```

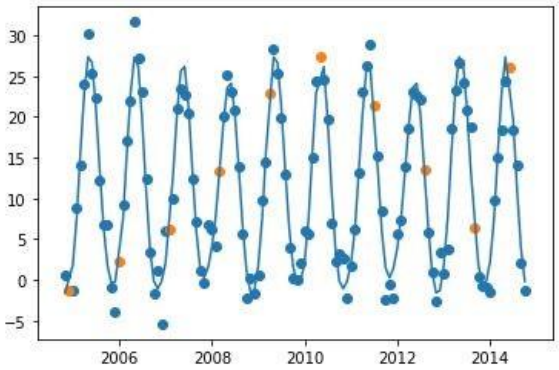
Grid search - 1st column is value of b, 2nd is i and 3rd is the value of error

Optimal values-

1. i=8
2. b=24

Hence design matrix has values from Sinx , Cox to Sin8x, Cos8x

After training with the given values test set could be predicted as follows



Blue points- Training data
Orange points- Test data

After submitting to Kaggle got an MSE of 3.44461