Report Assignment 1 Pattern Recognition & Machine learning

Riccardo Guderzo GE24Z227
Joanna Kolaczek GE24Z229
Miguel Mauer GE24Z022

Assignment description

Dataset	Degree of Polynomial M	Regularization Coefficient λ
Dataset 1	3, 6, 9	0.001, 0.1, 1
Dataset 2	2, 4, 6	0.001, 0.1, 1
Dataset 3	2, 3	0.000001, 0.0001, 0.1

Regression Model: Linear model for regression using polynomial basis functions
Regularization method: Quadratic regularization
Dataset 1: 1-dimensional (Univariate) input data – Training Dataset 1(a): 10 examples, Training Dataset 1(b): 50 examples
Dataset 2: 2-dimensional (Bivariate) input data – Training Dataset 2(a): 25 examples, Training Dataset 2(b): 100 examples
Dataset 3: Multivariate data

Task 1

For Dataset 1: Plots of the approximated functions (curves) obtained using training datasets of different sizes (10 and 50), for different model complexities with no regularization, and for different values of λ with model complexity as 9. The training data points need to be superposed on the curve.

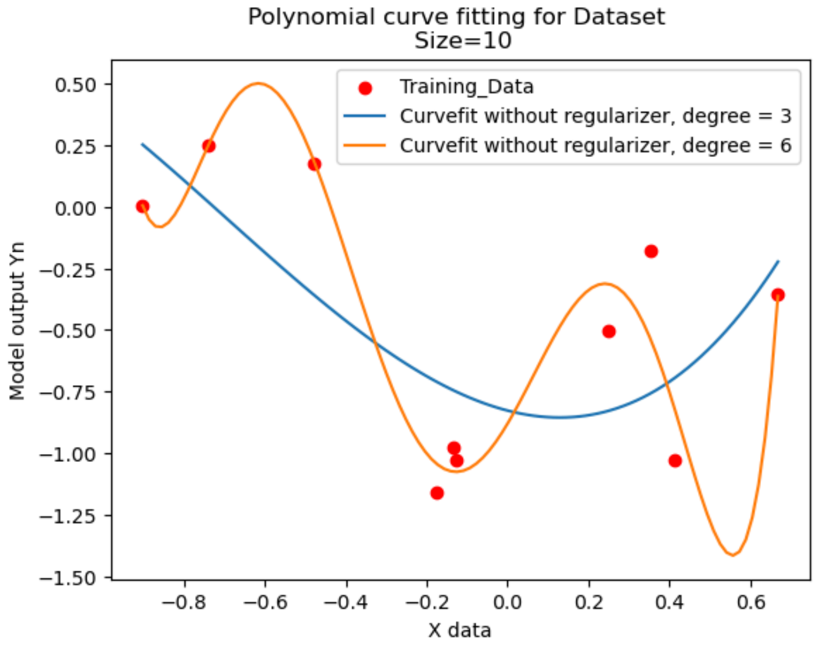


Fig 1.a

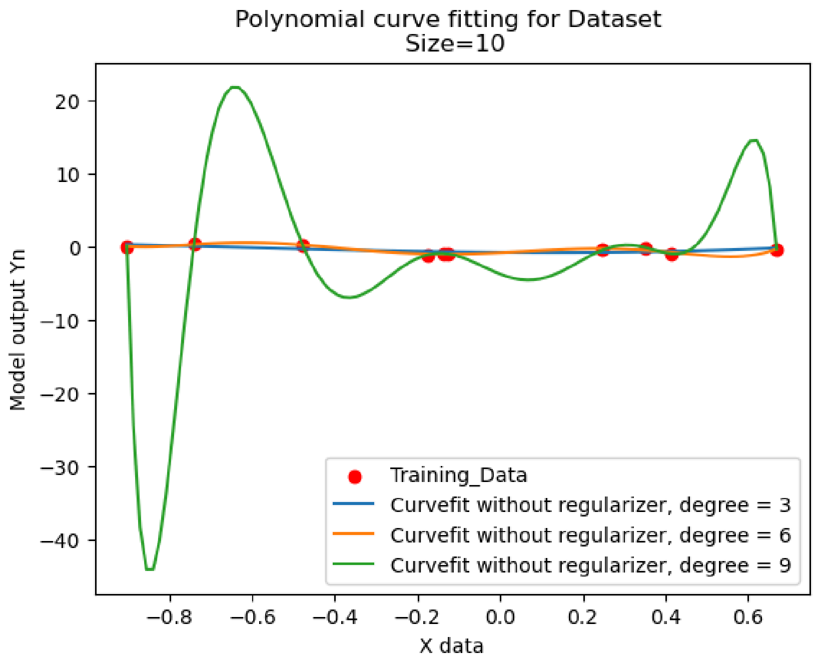


Fig 1.b

In order to better visualize the fit of the model to the data, the polynomials of grade 3 and 6 are reported alone (fig 1.a). Moreover, the fit of the three different grade models are reported all together in fig 1.b. To the naked eye, it appears that both grades 6 and 9 tend to overfit the data. The orange curve fits perfectly 4 points out of 10, while the green curve matches perfectly all the 10 points (which is reliable since the number of parameters is the same as N).

The models will be verified by using the test data below.

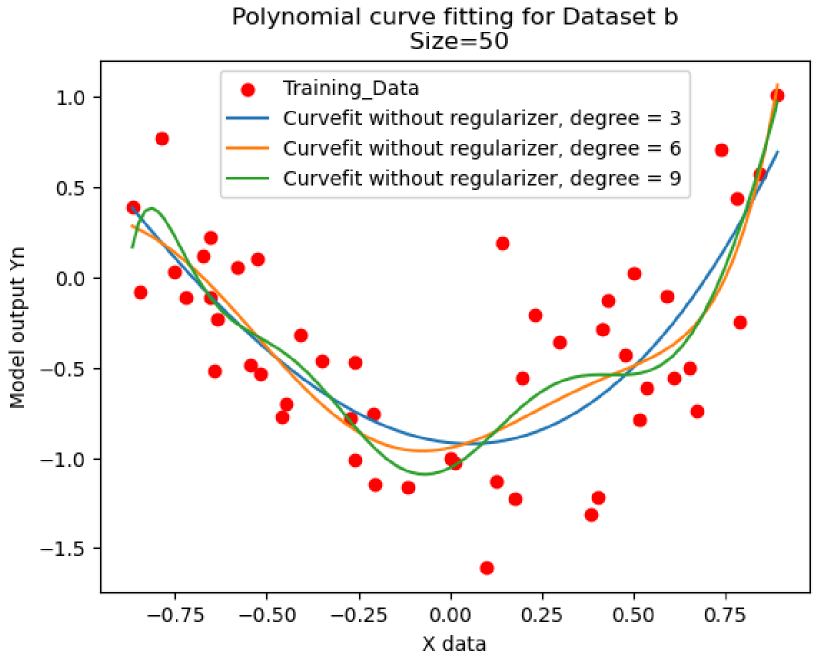


Fig 1.c

In Fig 1.c it is possible to see the fitting of the same polynomials as above but with a larger dataset of 50 samples. It is more difficult to understand if the models are actually overfitting since the training data are much more. The rule of thumb is that N should be around 10/15 times the number of parameters, therefore the orange curve should be a pretty good model. The models will be verified by using the test data below.

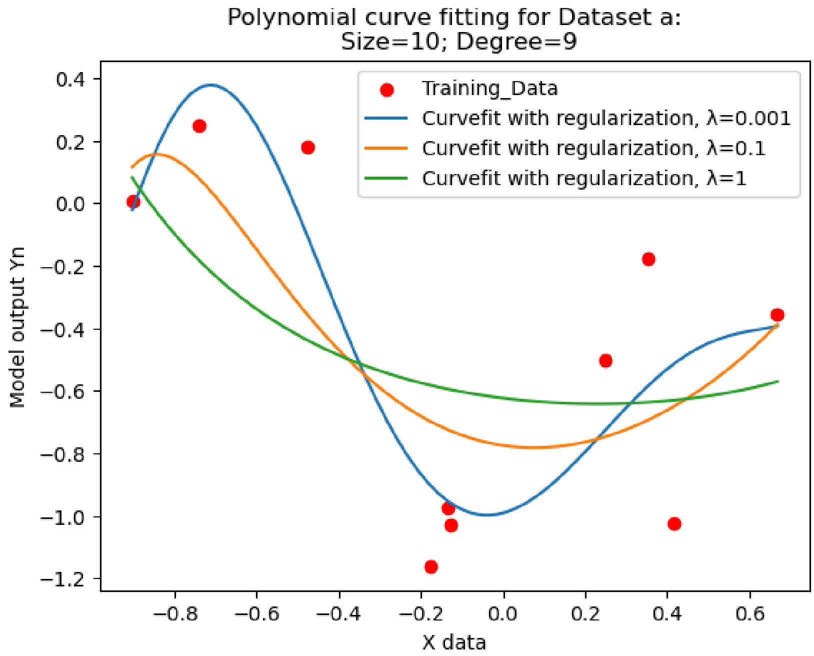


Fig 1.d

In Fig 1.d are reported the three polynomials of same grade but regularized with three different λ (Ridge coefficient). It is possible to spot the increasing shrinkage effect on the parameters: the bigger λ, the more the curve tends to become a straight line. Therefore, as expected, the bigger the Ridge coefficient, the bigger the bias: this is the price it is necessary to pay in order to lower the variance.

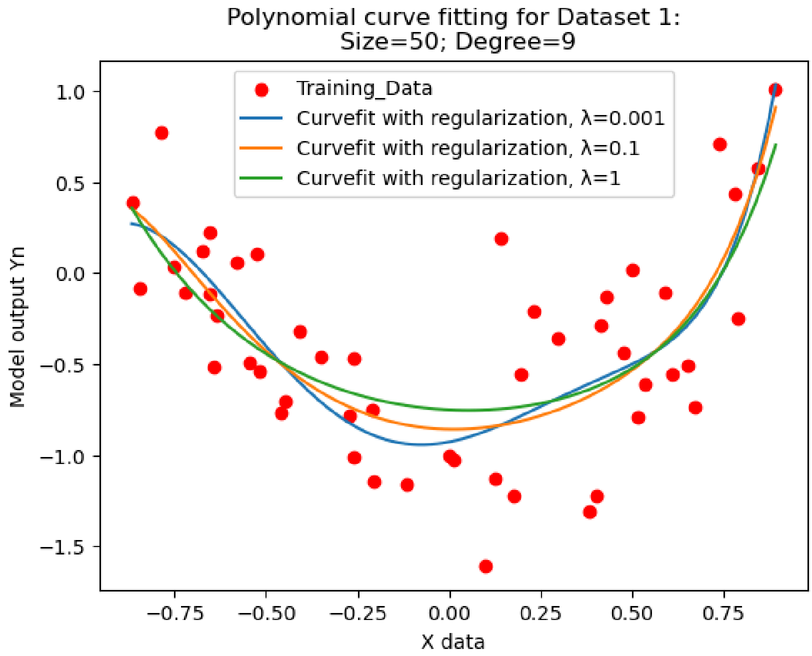


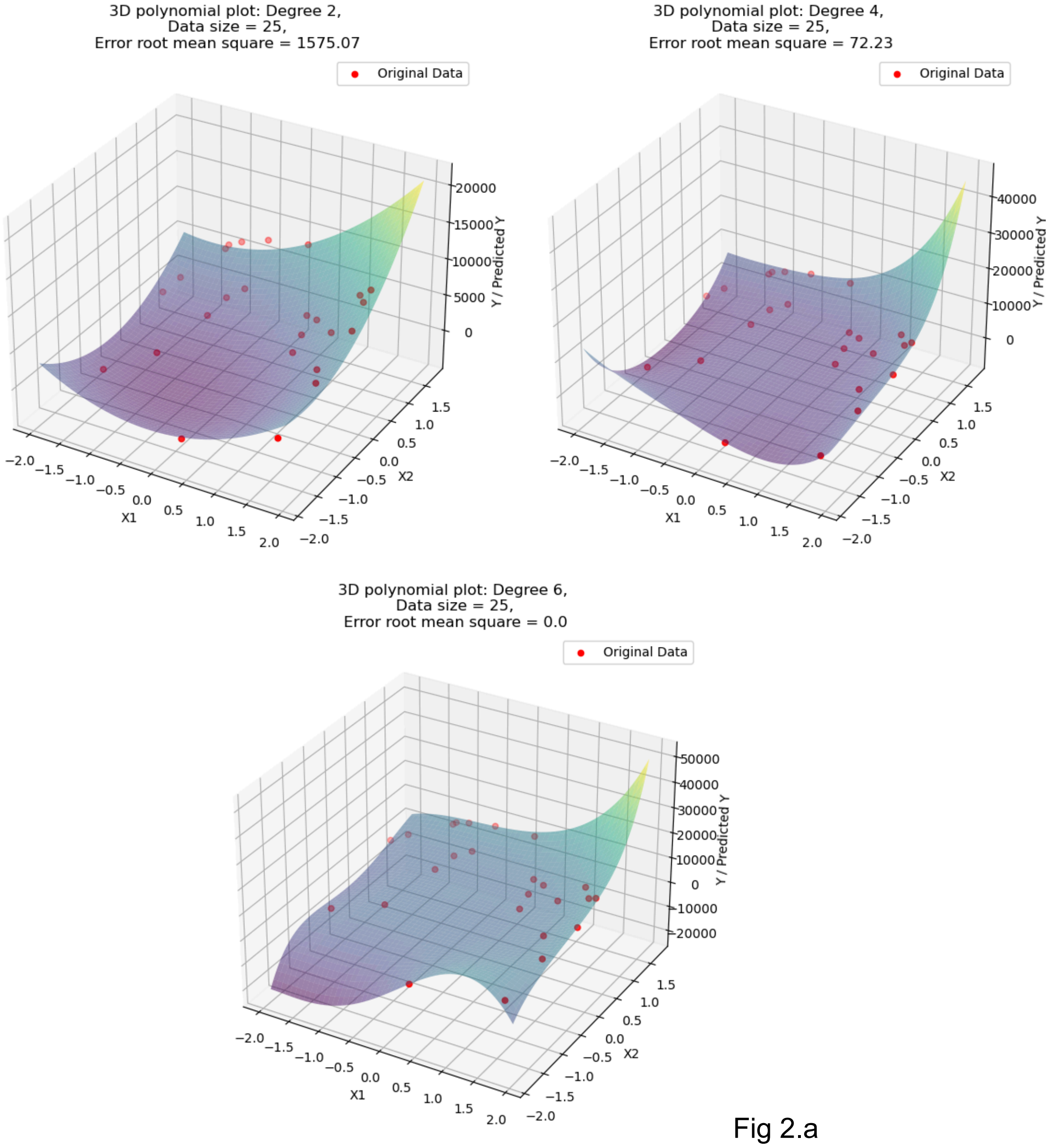
Fig 1.e

Also in Fig 1.e it is possible to spot the effect of the regularization as the Ridge coefficient increases. In this case the model has more information to learn the underlying patterns of the data because N=50, therefore the effect is a bit more difficult to spot.

This typically means that the model can achieve a good fit without heavily relying on the regularization term to prevent overfitting. On the other side, in Fig 1.d, the effect of the shrinkage was much more visible, since the risk of overfitting is higher because the model tries to fit the noise in the data rather than the underlying pattern.

Task 2

For Dataset 2: Plots of the surfaces of the approximated function obtained using training datasets of different sizes (25 and 100), for different model complexities with no regularization, and for different values of λ with model complexity as 6. The training data points need to be superposed on the surface.



3D polynomial plot: Degree 2,
Data size = 25,
Error root mean square = 1575.07

3D polynomial plot: Degree 4,
Data size = 25,
Error root mean square = 72.23

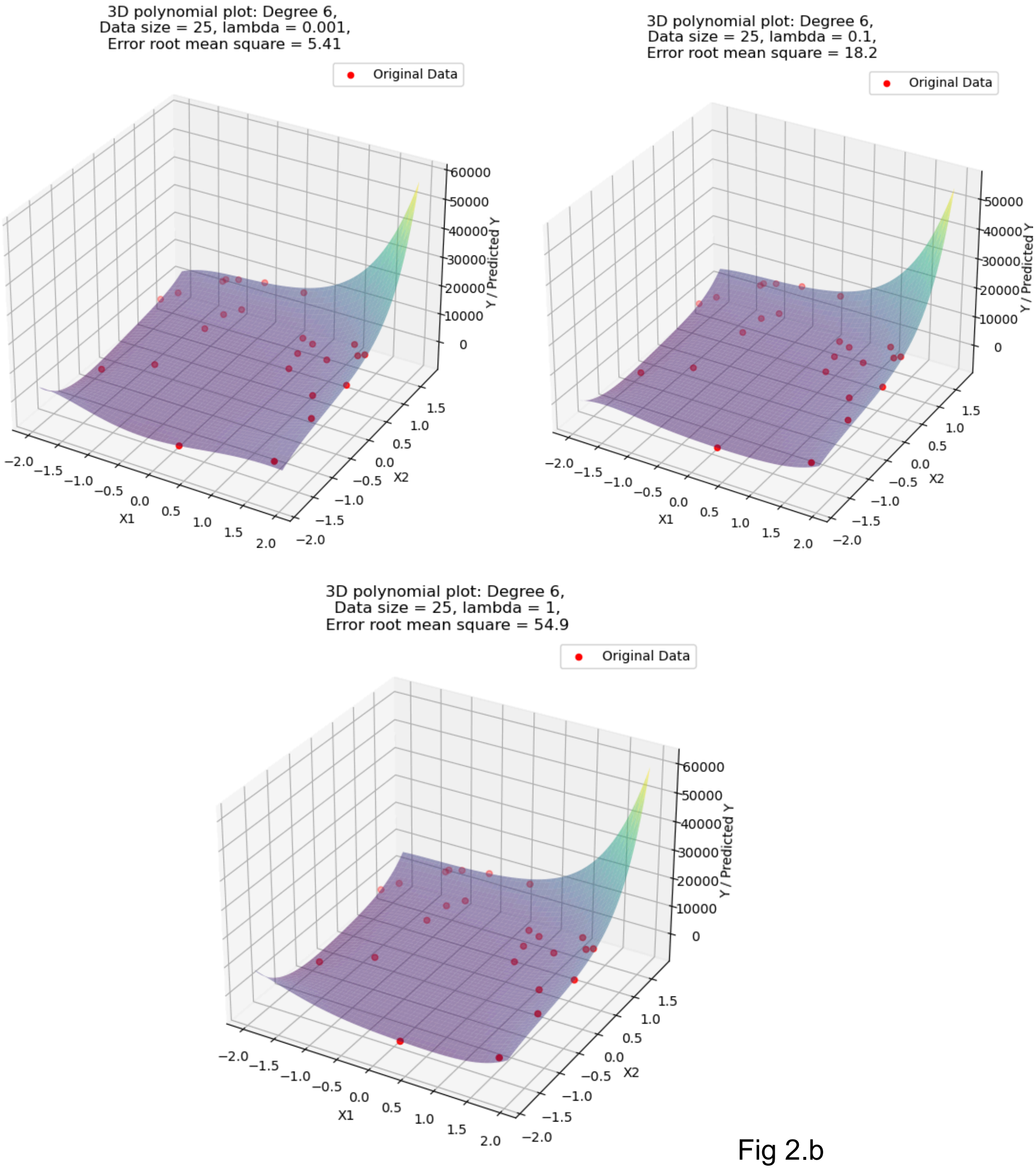
3D polynomial plot: Degree 6,
Data size = 25,
Error root mean square = 0.0

Fig 2.a

In this task the surfaces have been approximated from the training data. Fig 2.a visualizes surfaces obtained with polynomials of degree 2,4 and 6 fitted by minimizing loss function. Polynomial of degree 2 and dimension form 2 has general form as :

$$y = \alpha_0 + \alpha_1 x_1^2 + \alpha_2 x_2^2 + \alpha_3 x_1 x_2 + \alpha_4 x_1 + \alpha_5 x_2 ,$$

so it has 6 coefficients. For degree 6, the number of coefficients grows to 28, which is more than the number of datapoints in a given dataset, that's why error RMS can be reduced to 0.


3D polynomial plot: Degree 6,
Data size = 25, lambda = 0.001,
Error root mean square = 5.41

3D polynomial plot: Degree 6,
Data size = 25, lambda = 0.1,
Error root mean square = 18.2

3D polynomial plot: Degree 6,
Data size = 25, lambda = 1,
Error root mean square = 54.9

Fig 2.b

In Fig 2.b it is reported how a polynomial of degree 6 fitted on the same above datasets behaves with regularization. Error RMS is greater as respect to the case without regularization. This is because Ridge regression adds a penalty equal to the square of the coefficients. Now the model's coefficients are constrained, preventing it from fitting the data perfectly. As a result, the model might not capture every detail (including noise), leading to a higher RMS error on the training data.

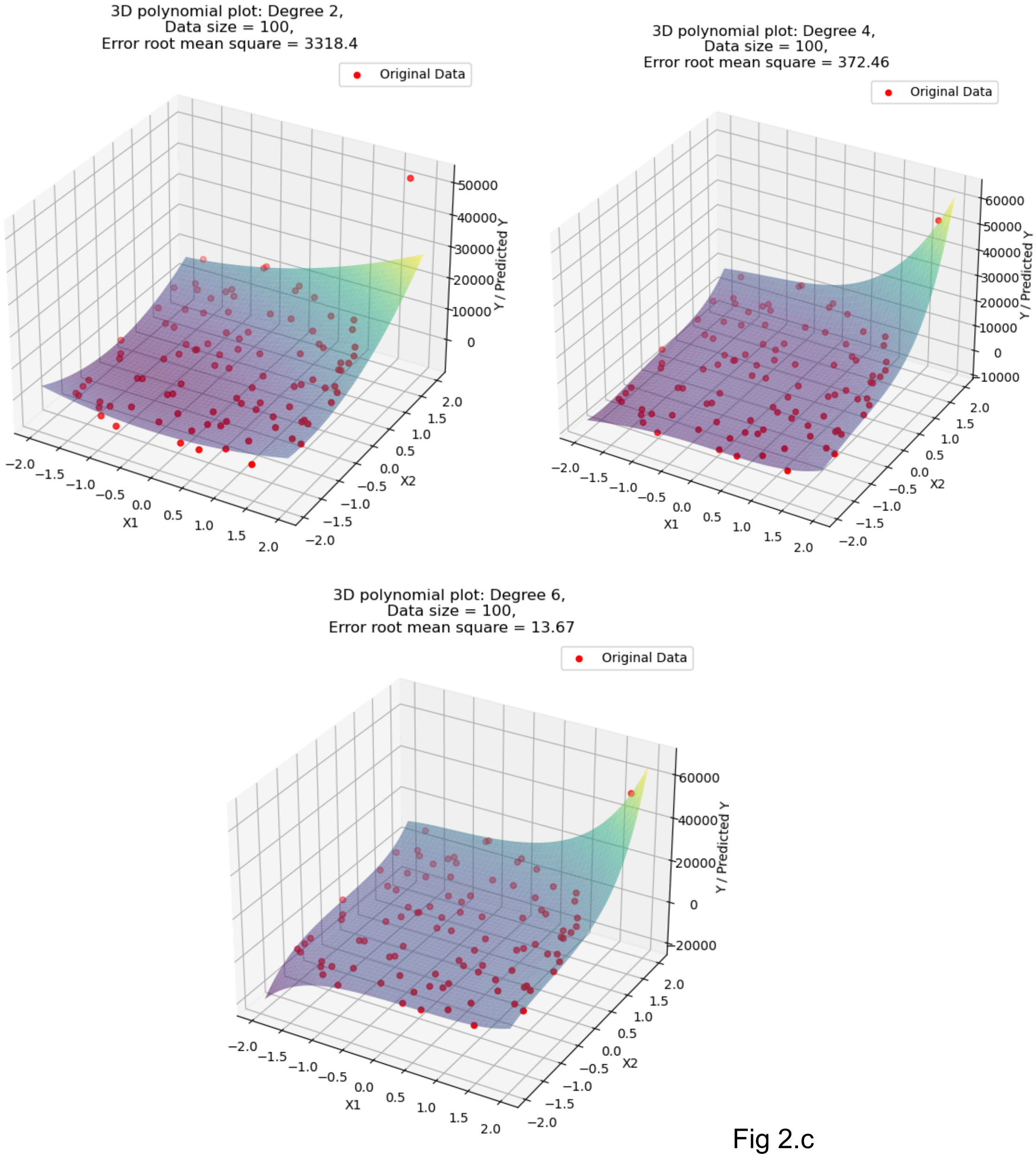
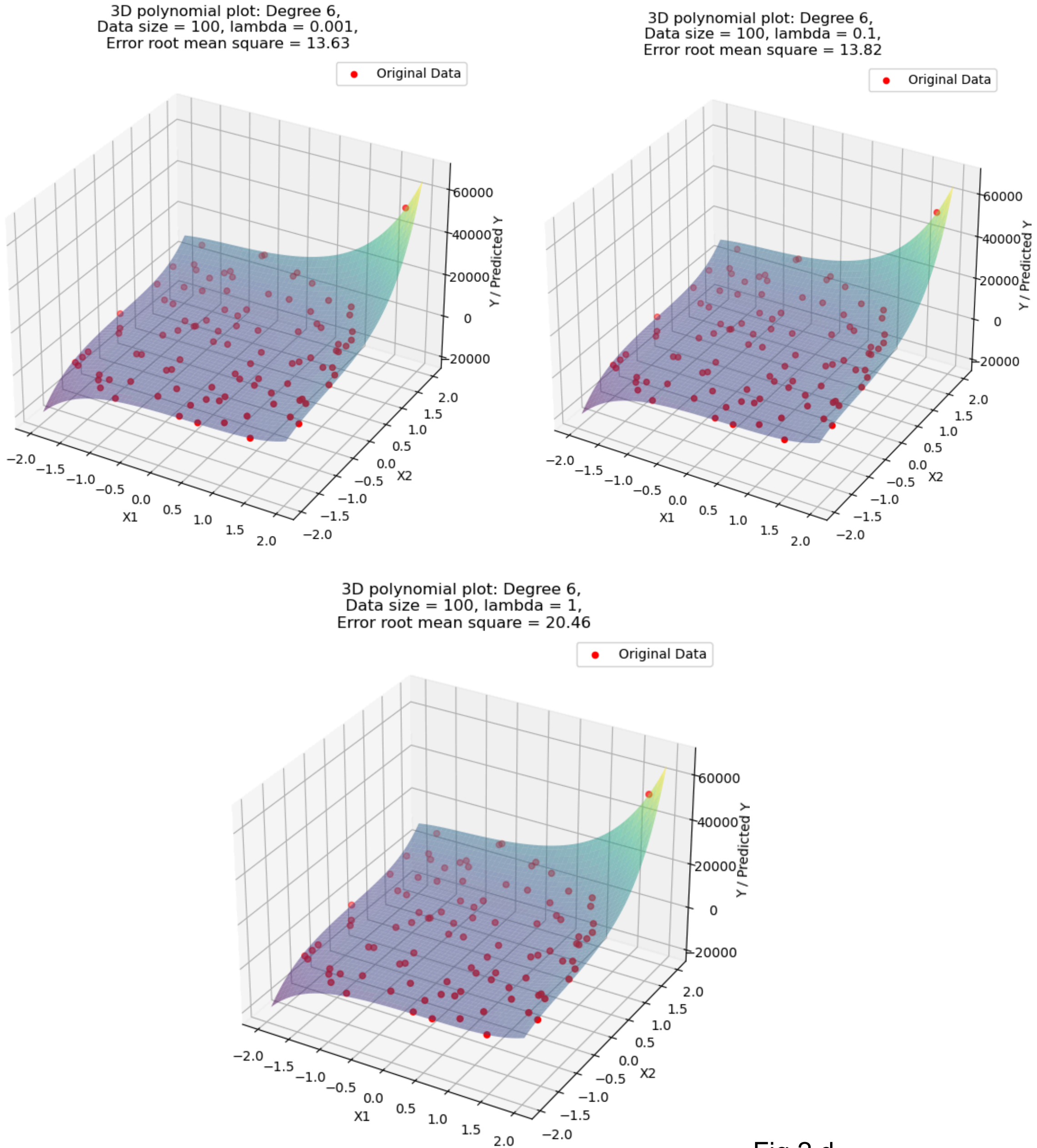


Fig 2.c

For a training dataset of size 100, results of polynomial surface fitting without regularization are plotted in Fig 2.c. It can be observed that, especially for degree 2, error RMS is high. It can also be seen that a surface of this degree, in contrast to the two other surfaces, can't reach a point that from visual inspection could be considered as an outlier.



3D polynomial plot: Degree 6,
Data size = 100, lambda = 0.001,
Error root mean square = 13.63

3D polynomial plot: Degree 6,
Data size = 100, lambda = 0.1,
Error root mean square = 13.82

3D polynomial plot: Degree 6,
Data size = 100, lambda = 1,
Error root mean square = 20.46

Fig 2.d

After performing regularization for the model of degree 6 (Fig 2.d) for the smallest lambda (regularization coefficient) we see negligible improvement of error RMS. For higher lambdas though, the error RMS tends to increase as the coefficients increase.

When RMS is close to 0 it may mean that model overfits as it was in case mentioned earlier, on the other hand it is possible to "over-regularize" a model, and doing so can negatively impact its performance. Overregularization occurs when the regularization penalty is too strong, excessively constraining the model.

In any case, the behavior analyzed in the 2D case is the same as the one described above for the univariate dataset. The error tends to decrease steadily as the complexity of the model increases. The regularization balances the effect by keeping the error more “steady”.

Task 3

For Datasets 1(a), 1(b), 2(a), 2(b), 3: Scatter plots with target output t_n on x-axis and model output $y(x_n,w)$ on y-axis for the best performing model, for training data and test data.

1(a) - 1(b)

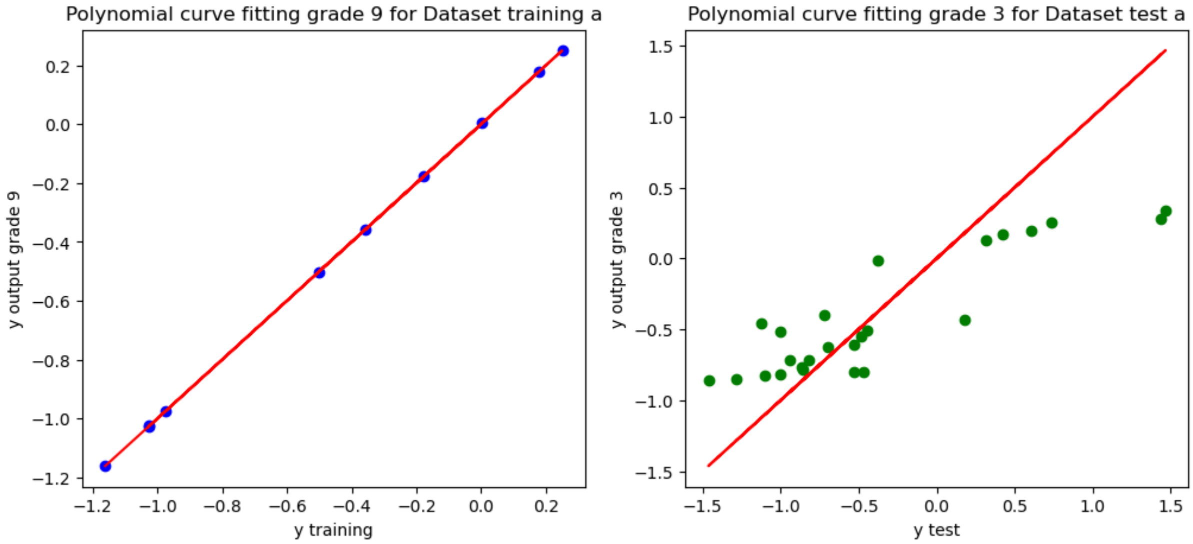


Fig 3.a

In the first plot of Fig 3.a, it is possible to see the target y vs the fitted values. The training data are from dataset a. Of course in the left side plot the chosen model is the polynomial with the greatest grade.This is because it fits perfectly the data (also the underlying noise) and the relative MSE is almost 0. On the right side, for the testing data, the best model is the polynomial of grade 3 since the MSE was 0.2139, which is the smallest as compared to the others (1.499 for grade 6, 10339 for grade 9). This result was expected since the model of grade 9, for example, excessively overfits the data, the bias is therefore 0 but with the price of a very big variance of the model.

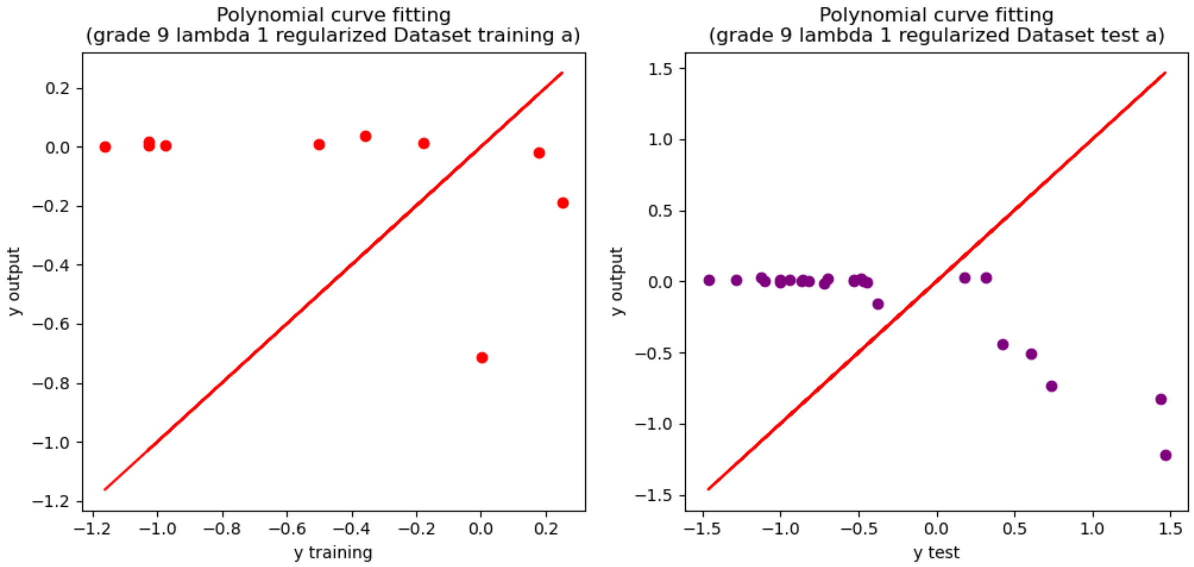


Fig 3.b

In both the plots of Fig 3.b, it can be seen that the fit of the model is not as good as above, even though the grade of the polynomial is 9. This is because the mode is balancing between bias and variance. Besides, N is really small, therefore the model has not been trained properly.

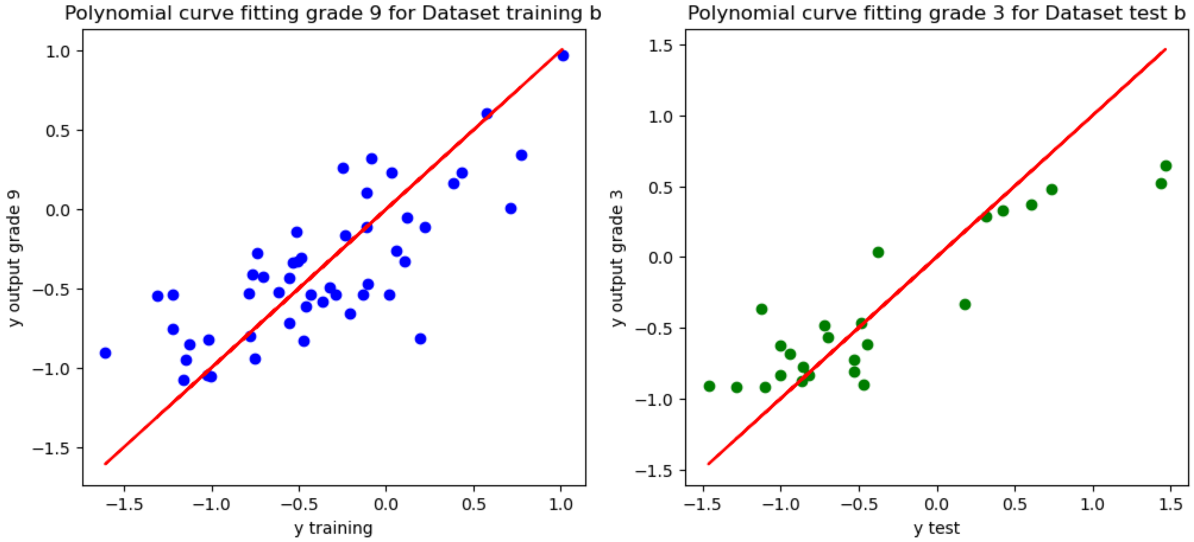
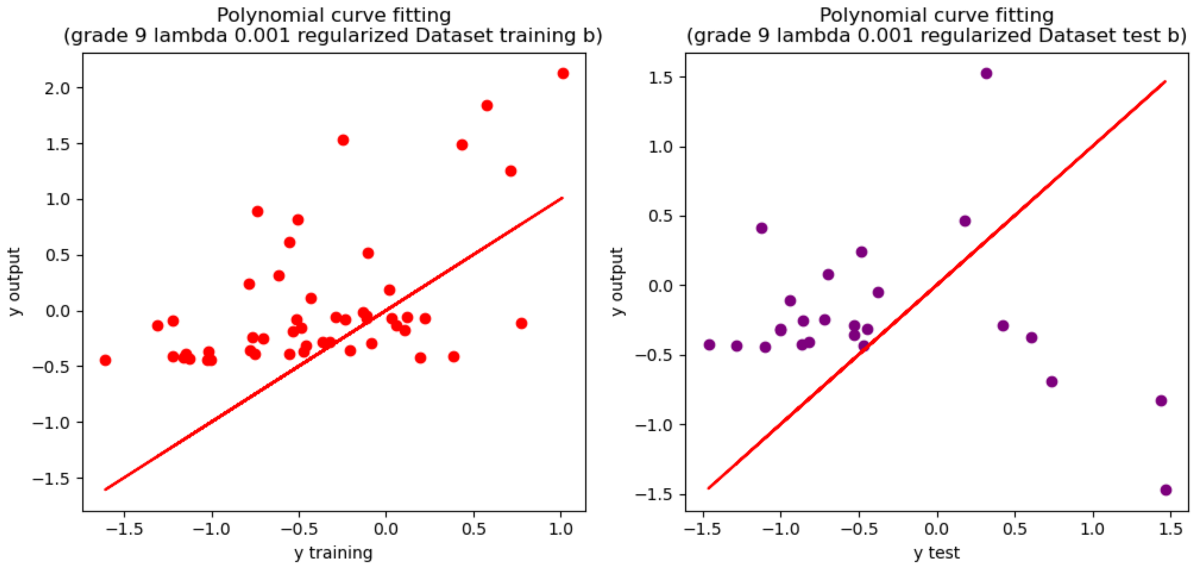


Fig 3.c

The above figure depicts the fitted values with a polynomial of grade 9. The fitting is not as perfect as in Figure 3.a: the dataset has more data values as compared to the number of parameters.


Polynomial curve fitting
(grade 9 lambda 0.001 regularized Dataset training b)

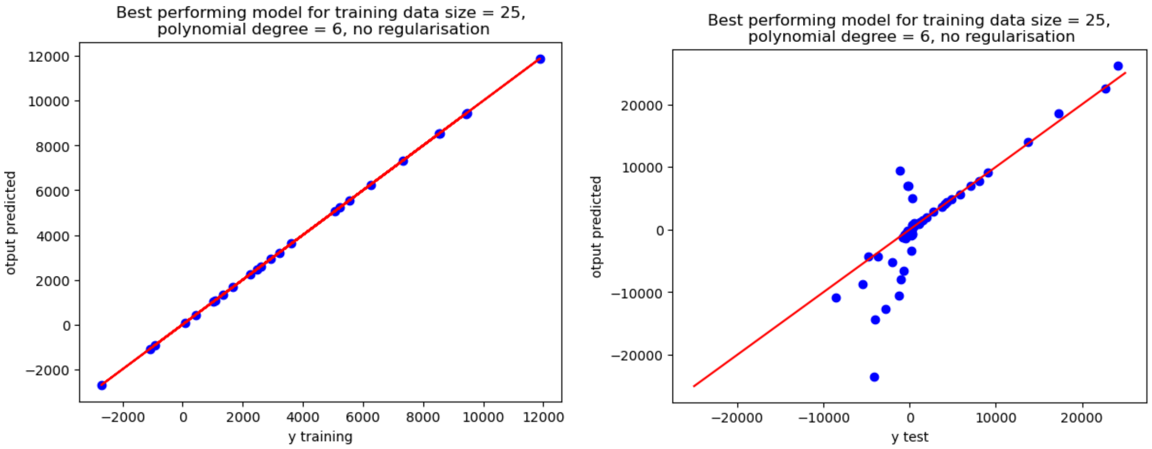
Polynomial curve fitting
(grade 9 lambda 0.001 regularized Dataset test b)

Fig 3.d

In Fig 3.d it is possible to see that, even though the magnitude of λ is not high, the regularization tends to decrease the weights too much. Therefore the data plotted tends to be more sparse.

2(a) - 2(b)

The lowest ERMS was used as an indicator to choose the degree and regularization coefficient for comparing the model trained on the training data with the test data. For dataset 2a, this was a degree 6 polynomial without regularization. The ERMS was equal to 0 for the training data—as shown in Figure 3.e, the training data and model output are identical. However, this is not the case for the test data: the plot shows significant differences between the model and the test data.


Best performing model for training data size = 25,
polynomial degree = 6, no regularisation

Best performing model for training data size = 25,
polynomial degree = 6, no regularisation

Fig 3.e

For the model trained on the larger dataset (2b), a degree 6 polynomial and a regularization coefficient of lambda 0.001 were chosen as the best. In Figure 3.f it is possible to see that both the training and test results match the model very well.

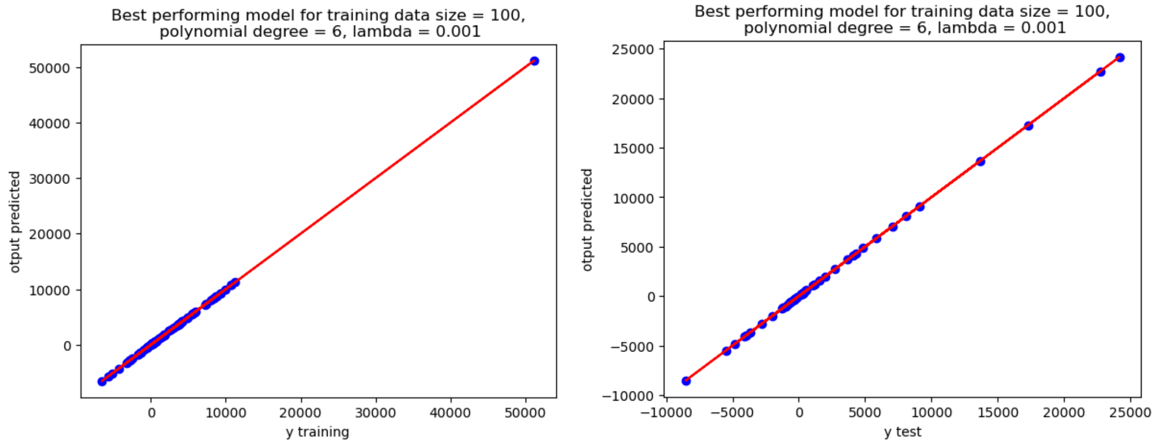


Fig 3.f

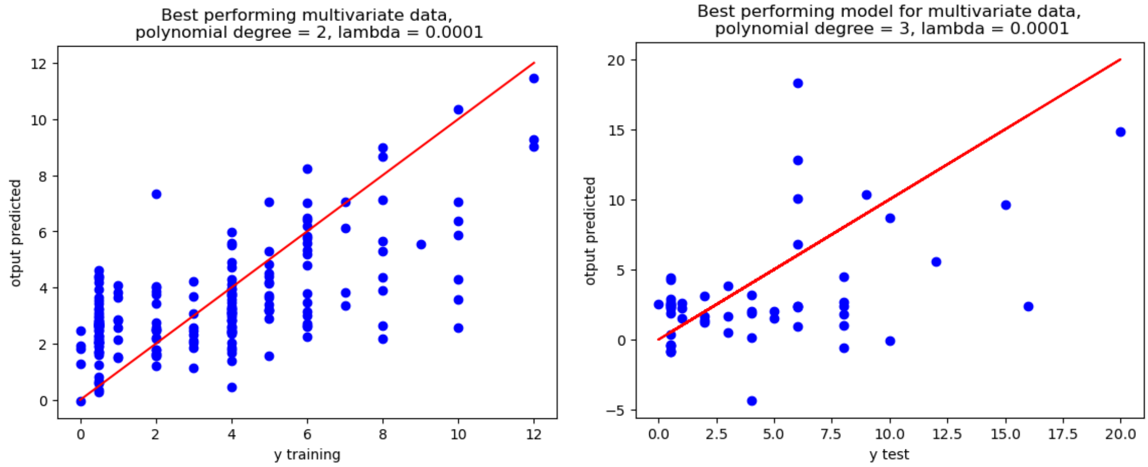


Fig 3.g

As shown in Fig 3.g for both the models (degree 2 and 3) the best regularization's results have been achieved with a λ = 0.0001.

In Fig 4.f below the ERMS for different combinations have been reported.

Graphically it is possible to see an increasing trend in the plotted points, even though as the y increases the behavior tends to be more random.

Task 4

For Datasets 1(a), 1(b), 2(a), 2(b), 3: Tables showing the $ERMS$ on the training data, the validation data and the test data, for models without and with regularization

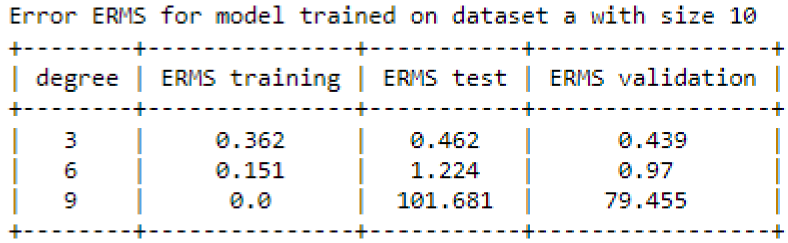
1(a) - 1(b)

Error ERMS is defined as follow:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} \|y(i) - \hat{y}(i)\|^2}{N}},$$

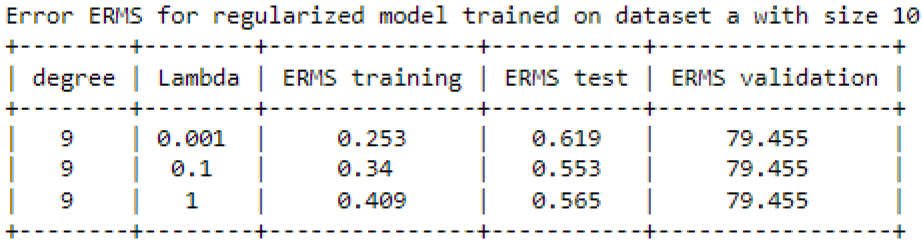
The above index has been defined for different degree polynomials without regularization (Fig 4.a and Fig 4.c below) and for a polynomial of degree 9 with regularization for different values of λ (Fig 4.b and Fig 4.d below).

In the models without regularization it is possible how the bias/variance affect the error. In a polynomial of degree 9, for example, the ERMS training is 0 (or close to 0 when the dataset has 50 samples), but the same is very high when the validation/test dataset has been used.
In the case of regularized models, the ERMS are more “balanced” between the training and the test dataset (ie: with the test dataset the ERMS is not rocketing as expected without regularization). In the case of validation dataset though, the ERMS remains constant for all the different values of λ. This behavior is unusual, it might be the result of a too small λ, so that its impact on the model is negligible. In this case, the model would behave similarly to an unregularized model, and the validation error might not change much with different regularization strengths.



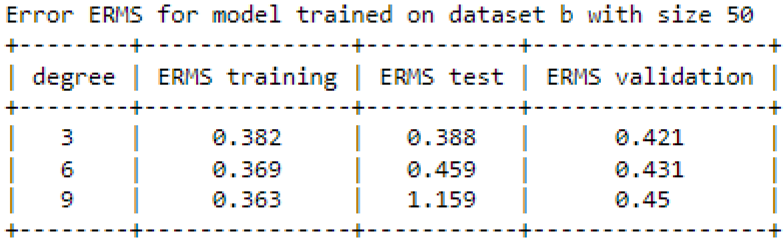
```
Error ERMS for model trained on dataset a with size 10
+--------+---------------+-----------+-----------------+
| degree | ERMS training | ERMS test | ERMS validation |
+--------+---------------+-----------+-----------------+
|   3    |     0.362     |   0.462   |      0.439      |
|   6    |     0.151     |   1.224   |      0.97       |
|   9    |      0.0      |  101.681  |     79.455      |
+--------+---------------+-----------+-----------------+
```

Fig 4.a



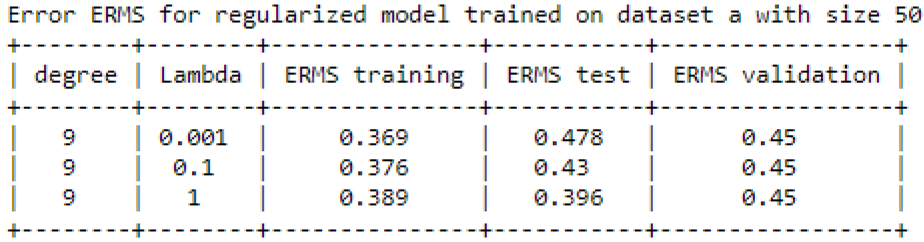
```
Error ERMS for regularized model trained on dataset a with size 10
+--------+--------+---------------+-----------+-----------------+
| degree | Lambda | ERMS training | ERMS test | ERMS validation |
+--------+--------+---------------+-----------+-----------------+
|   9    | 0.001  |     0.253     |   0.619   |     79.455      |
|   9    |  0.1   |     0.34      |   0.553   |     79.455      |
|   9    |   1    |     0.409     |   0.565   |     79.455      |
+--------+--------+---------------+-----------+-----------------+
```

Fig 4.b



```
Error ERMS for model trained on dataset b with size 50
+--------+---------------+-----------+-----------------+
| degree | ERMS training | ERMS test | ERMS validation |
+--------+---------------+-----------+-----------------+
|   3    |     0.382     |   0.388   |      0.421      |
|   6    |     0.369     |   0.459   |      0.431      |
|   9    |     0.363     |   1.159   |      0.45       |
+--------+---------------+-----------+-----------------+
```

Fig 4.c

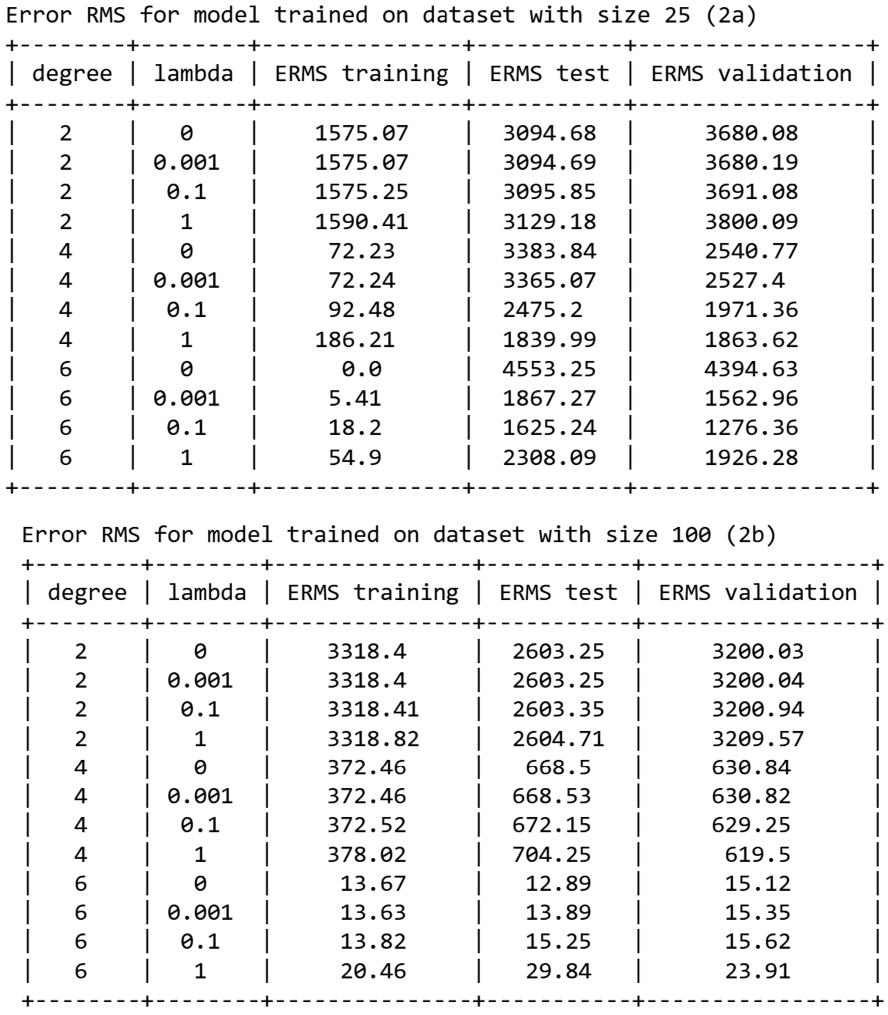


```
Error ERMS for regularized model trained on dataset a with size 50
+--------+--------+---------------+-----------+-----------------+
| degree | Lambda | ERMS training | ERMS test | ERMS validation |
+--------+--------+---------------+-----------+-----------------+
|   9    | 0.001  |     0.369     |   0.478   |      0.45       |
|   9    |  0.1   |     0.376     |   0.43    |      0.45       |
|   9    |   1    |     0.389     |   0.396   |      0.45       |
+--------+--------+---------------+-----------+-----------------+
```

Fig 4.d

2(a) - 2(b)

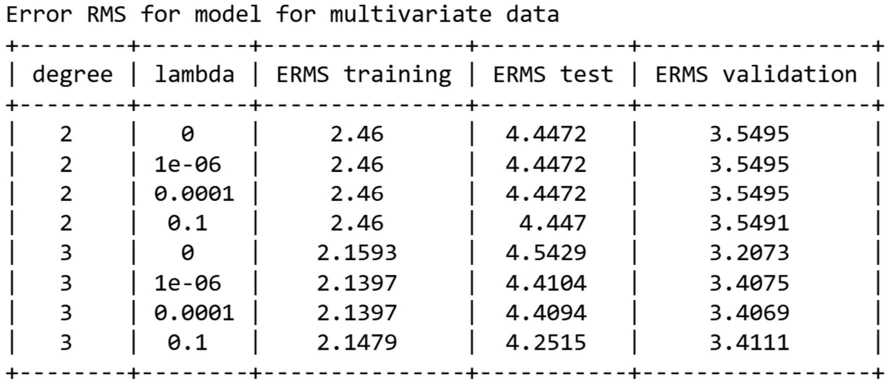
The left table in Figure 4.e shows the ERMS values for training, test, and validation data for the bivariate input of length 25 (2a), while the right table shows the ERMS values for input of length 100 (2b). Both models are compared using the test and validation data. Although the ERMS values for the training data of the smaller dataset (2a) are generally lower than those for the training data of 2b, it can be observed that the ERMS values for the test and validation data are much higher for the model based on 2a than for the model based on 2b. This indicates that overfitting occurred in the case of the 2a model. For the test and validation data, the best performance is given by the polynomial of degree 6, particularly the one without regularization, as regularization in this case may have caused underfitting.



```
Error RMS for model trained on dataset with size 25 (2a)
+--------+--------+--------------+----------+----------------+
| degree | lambda | ERMS training | ERMS test | ERMS validation |
+--------+--------+--------------+----------+----------------+
|   2    |   0    |   1575.07    | 3094.68  |    3680.08     |
|   2    | 0.001  |   1575.07    | 3094.69  |    3680.19     |
|   2    |  0.1   |   1575.25    | 3095.85  |    3691.08     |
|   2    |   1    |   1590.41    | 3129.18  |    3800.09     |
|   4    |   0    |    72.23     | 3383.84  |    2540.77     |
|   4    | 0.001  |    72.24     | 3365.07  |    2527.4      |
|   4    |  0.1   |    92.48     | 2475.2   |    1971.36     |
|   4    |   1    |   186.21     | 1839.99  |    1863.62     |
|   6    |   0    |     0.0      | 4553.25  |    4394.63     |
|   6    | 0.001  |    5.41      | 1867.27  |    1562.96     |
|   6    |  0.1   |    18.2      | 1625.24  |    1276.36     |
|   6    |   1    |    54.9      | 2308.09  |    1926.28     |
+--------+--------+--------------+----------+----------------+
```

```
Error RMS for model trained on dataset with size 100 (2b)
+--------+--------+--------------+----------+----------------+
| degree | lambda | ERMS training | ERMS test | ERMS validation |
+--------+--------+--------------+----------+----------------+
|   2    |   0    |    3318.4    | 2603.25  |    3200.03     |
|   2    | 0.001  |    3318.4    | 2603.25  |    3200.04     |
|   2    |  0.1   |   3318.41    | 2603.35  |    3200.94     |
|   2    |   1    |   3318.82    | 2604.71  |    3209.57     |
|   4    |   0    |   372.46     |  668.5   |    630.84      |
|   4    | 0.001  |   372.46     | 668.53   |    630.82      |
|   4    |  0.1   |   372.52     | 672.15   |    629.25      |
|   4    |   1    |   378.02     | 704.25   |     619.5      |
|   6    |   0    |    13.67     |  12.89   |     15.12      |
|   6    | 0.001  |    13.63     |  13.89   |     15.35      |
|   6    |  0.1   |    13.82     |  15.25   |     15.62      |
|   6    |   1    |    20.46     |  29.84   |     23.91      |
+--------+--------+--------------+----------+----------------+
```

Fig 4.e



```
Error RMS for model for multivariate data
+--------+--------+--------------+----------+----------------+
| degree | lambda | ERMS training | ERMS test | ERMS validation |
+--------+--------+--------------+----------+----------------+
|   2    |   0    |     2.46     |  4.4472  |     3.5495     |
|   2    | 1e-06  |     2.46     |  4.4472  |     3.5495     |
|   2    | 0.0001 |     2.46     |  4.4472  |     3.5495     |
|   2    |  0.1   |     2.46     |   4.447  |     3.5491     |
|   3    |   0    |    2.1593    |  4.5429  |     3.2073     |
|   3    | 1e-06  |    2.1397    |  4.4104  |     3.4075     |
|   3    | 0.0001 |    2.1397    |  4.4094  |     3.4069     |
|   3    |  0.1   |    2.1479    |  4.2515  |     3.4111     |
+--------+--------+--------------+----------+----------------+
```

Fig 4.f

The table in Fig. 4.f presents Error RMS (Root Mean Square) values for models with polynomial degrees of 2 and 3, trained with regularization parameters (lambda) of 0.000001, 0.0001, and 0.1. The ERMS values are shown for the training, test, and validation datasets.

As observed, for a polynomial degree of 2, increasing the regularization parameter slightly reduces the validation and test RMSE, indicating a marginal improvement in model performance. Specifically, the model with lambda = 0.1 achieves the lowest ERMS.

For a polynomial degree of 3, a different trend is seen, where a higher regularization parameter lambda = 0.1 leads to the higher ERMS values across validation and training datasets. This suggests that a higher degree polynomial combined with a moderate level of regularization provides the best balance between model complexity and generalization performance.

Conclusion

In this analysis, we explored the behavior of polynomial models with varying degrees and the impact of regularization across different datasets. As expected, higher-degree polynomials tend to overfit the training data, especially when the dataset is small, leading to a significant increase in test error. Regularization, particularly Ridge regression, effectively mitigates this overfitting by introducing bias, which reduces variance and prevents the model from fitting noise in the data. The results emphasize the importance of balancing model complexity and regularization to achieve optimal performance, particularly when working with limited data.