Machine, Data and Learning Assignment 1

Team 97: Arushi Mittal (2019101120), Meghna Mishra (2019111030)

Task 1:

LinearRegression().fit() is a function from the sklearn.linear_model module that is used in order to create a predictive machine learning model. In case of supervised machine learning models like the one used for this assignment, the function takes two parameters, the input values and the correct output values for which the model needs to be developed. The function then learns from this data and develops a linear regression model that can take input values and predict the output values. It returns the coefficients for each of the input features that would fit the model most accurately, depending on the curve that best fits the data. Linear regression is a machine learning algorithm that can be used to predict values over a continuous range using a constant slope. This constant slope is determined by using gradient descent to determine the model with the lowest mean squared error.

$$Y = b_0 + b_1 x^1 + b_2 x^2 + b_3 x^3 + ... + b_n x^n$$

where Y is the dependent variable or the output and x is the independent variable and $b_1, b_2, b_3, ..., b_n$ are the coefficients for each polynomial degree of x.

Task 2:

Initially, for the models with low complexity (referring to degrees 1 and 2), the values predicted by the model did not fit the test data accurately because the model was "underfit", and hence the bias was high. The models did not generalize the data well, and was oversimplified. Then as the order increases, the bias got reduced due to the model being better trained.
However, for very high complexities (referring to degrees greater than 15), the predicted values do not accurately fit the test data again and the bias begins increasing to higher values. This is because the model is now "overfit".
Now, looking at the variance. As we went from lowest to highest complexity (i.e. increased the degree), the variance of the model increased. This was because the models became more sensitive to fluctuations in the training data, and and so "overfitting" took place. Thus, with high variance, the predicted values again do not accurately fit the test data.

	Bias	Variance
1	819.85	41401.38
2	811.34	49153.33
3	70.09	59614.16
4	76.80	72772.60
5	77.02	95321.37
6	74.04	112955.60
7	76.01	148182.45
8	78.82	159901.90
9	82.73	189911.56
10	85.43	201069.77
11	82.99	229256.72
12	110.94	223353.33
13	82.72	253025.88
14	120.40	241767.61
15	162.17	259071.85
16	166.41	278154.85
17	240.46	290501.13
18	240.64	305427.05
19	309.69	310263.55
20	307.46	321608.26

Task 3:

The values of irreducible error recorded are extremely small across all models, and on varying the degree, the change in irreducible error is also extremely small (and essentially insignificant). This is the error that cannot be reduced by betterment of the model itself, since it is essentially a measure of the amount of noise in our data. "Noise" here refers to any unwanted distortion in the data, which is not intended to be present but still manages to exist due faults in the capturing process of the data. Hence, certain noise inherently exists in the given dataset which leads to irreducible error which cannot be removed. There is barely any change because it is a measure of the noise in the data set and it does not explicitly depend on the model itself.

	Irreduceable Error
1	-2.110000e-10
2	2.180000e-11
3	-1.460000e-11
4	0.000000e+00
5	0.000000e+00
6	-1.460000e-11
7	0.000000e+00
8	0.000000e+00
9	-2.910000e-11
10	0.000000e+00
11	5.820000e-11
12	0.000000e+00
13	0.000000e+00
14	2.910000e-11
15	5.820000e-11
16	0.000000e+00
17	0.000000e+00
18	5.820000e-11
19	5.820000e-11
20	0.000000e+00

Task 4:

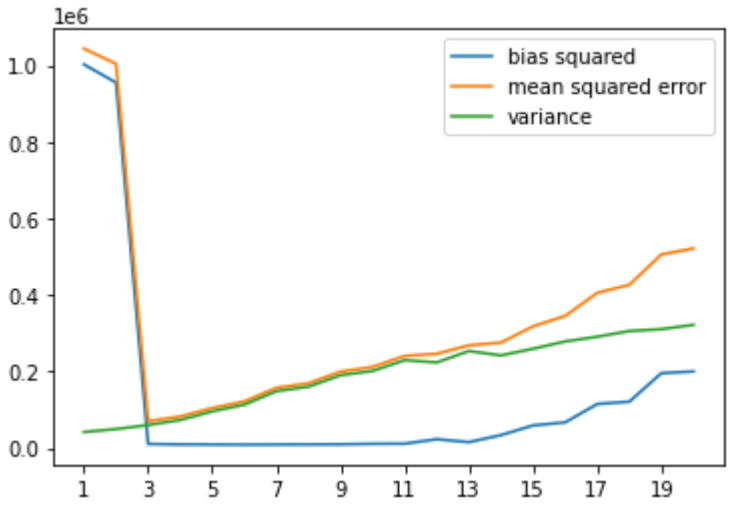
From the graph we observe that as complexity of the model increases, bias^2 first decreases and then for higher complexities begins increasing again. Meanwhile, variance increases with increasing complexity of the model.

Underfitting: When the complexity of the model was low (referring to degrees 1,2), the model was "underfit" as it was oversimplified and did not generalize well. We observe high bias and bias^2 as well as low variance. The reason for this is that due to the low complexity, it is not able to capture enough information about the training data, and hence the high bias^2. But the variance is low, because despite performing poorly the model is performing consistently.

Overfitting: When the complexity of the model was high (referring to degrees 15 and above), the model was "overfit". As a result of this, the variance is high, and the bias^2 also increases again after having decreased. It pays a lot of attention to the training data and does not generalize well on the testing data. Essentially, the model is so overfit on the training set, that it begins performing poorly on the testing set.

We observe that the total error is minimum at degree 3. Before degree 3, the bias is high and variance is low (underfit) and after degree 3, the variance and bias continuously increase (overfit). Thus, we can say that a model of degree 3 is best fitting, suggestive of a cubic function.

In the graph below, the x axis represents the degree of the polynomial (model cpmplexity) and the y axis represents the value of the given quantities (bias, variance and mean squared error) which is the model error.



	Mean Squared Error
1	1029734.16
2	1001154.17
3	74524.46
4	93209.22
5	127538.18
6	143660.30
7	174298.82
8	212292.08
9	267262.52
10	230385.22
11	300886.67
12	303649.40
13	305561.65
14	333365.10
15	367390.11
16	396502.96
17	456555.32
18	481318.32
19	578671.18
20	602543.13