Machine Data and Learning

Assignment -2

Name :- Suyash Sethia

Roll No:- 2021114010

Task 1

The `LinearRegression().fit()` method trains a linear regression model on a given dataset.

This technique takes as input the independent variable(s) (let X) and the dependent variable (let y) and then optimizes the coefficients to minimize the difference between the expected and actual values to fit a linear equation to the observed data. As it establishes the coefficients that will be applied to create predictions on new data using the predict() function, the `fit()` method is a crucial stage in the linear regression modeling procedure.

It performs best when the relationship between the dependent and independent variable(s) is linear and the data meets the assumptions of linear regression, including normality, homoscedasticity, and independence of errors. Meeting these assumptions can ensure accurate predictions.

Task 2

Gradient descent is an optimization algorithm used to find the coefficients (slope and intercept) in a linear regression model. A measure of the error between the projected values and the actual values, the cost function, is what gradient descent aims to decrease.

For a model with one independent variable and one dependent variable, the mean squared error (MSE), which is the average of the squared discrepancies between the predicted values and the actual values, is commonly used as the cost function.

The MSE can be represented as

$$MSE = (1/n) * \sum(y_\mathrm{i} - (mx_\mathrm{i} + b))^2$$

y is the dependent variable, x is the independent variable, m is the slope (coefficient), b is the intercept (bias), n is the number of data points, and i is the index of the data point.

The gradient descent algorithm starts with an initial guess for the values of m and b and iteratively adjusts them based on the gradient of the cost function with respect to the coefficients. The gradient is the vector of partial derivatives of the cost function with respect to each coefficient.

In each iteration, the algorithm updates the values of m and b using the following equations:

$$m = m - \alpha * (\partial MSE/\partial m)$$

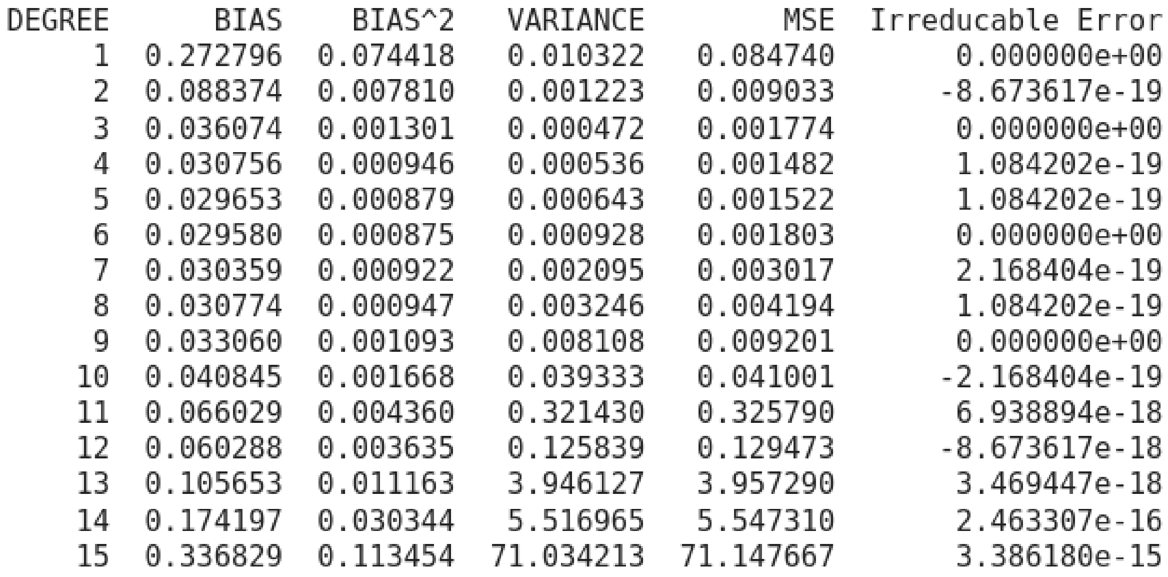
$$b = b - \alpha * (\partial MSE/\partial b)$$

where α is the learning rate, which controls the step size of the update. The partial derivatives are computed using the chain rule of calculus.

The ideal values of m and b are identified after the cost function converges to a minimum. These numbers relate to the linear regression model's coefficients that best match the data.

Tasks 3 and 4

Here are the tabulated Model bias and variance for different degree of function classes



```
DEGREE      BIAS    BIAS^2   VARIANCE        MSE  Irreducable Error
     1  0.272796  0.074418   0.010322   0.084740       0.000000e+00
     2  0.088374  0.007810   0.001223   0.009033      -8.673617e-19
     3  0.036074  0.001301   0.000472   0.001774       0.000000e+00
     4  0.030756  0.000946   0.000536   0.001482       1.084202e-19
     5  0.029653  0.000879   0.000643   0.001522       1.084202e-19
     6  0.029580  0.000875   0.000928   0.001803       0.000000e+00
     7  0.030359  0.000922   0.002095   0.003017       2.168404e-19
     8  0.030774  0.000947   0.003246   0.004194       1.084202e-19
     9  0.033060  0.001093   0.008108   0.009201       0.000000e+00
    10  0.040845  0.001668   0.039333   0.041001      -2.168404e-19
    11  0.066029  0.004360   0.321430   0.325790       6.938894e-18
    12  0.060288  0.003635   0.125839   0.129473      -8.673617e-18
    13  0.105653  0.011163   3.946127   3.957290       3.469447e-18
    14  0.174197  0.030344   5.516965   5.547310       2.463307e-16
    15  0.336829  0.113454  71.034213  71.147667       3.386180e-15
```

Bias refers to the error that is introduced by approximating a real-world problem with a simplified model. This is often caused by the model's inability to capture the underlying patterns in the data. A high-bias model is typically too simple and tends to underfit the data.

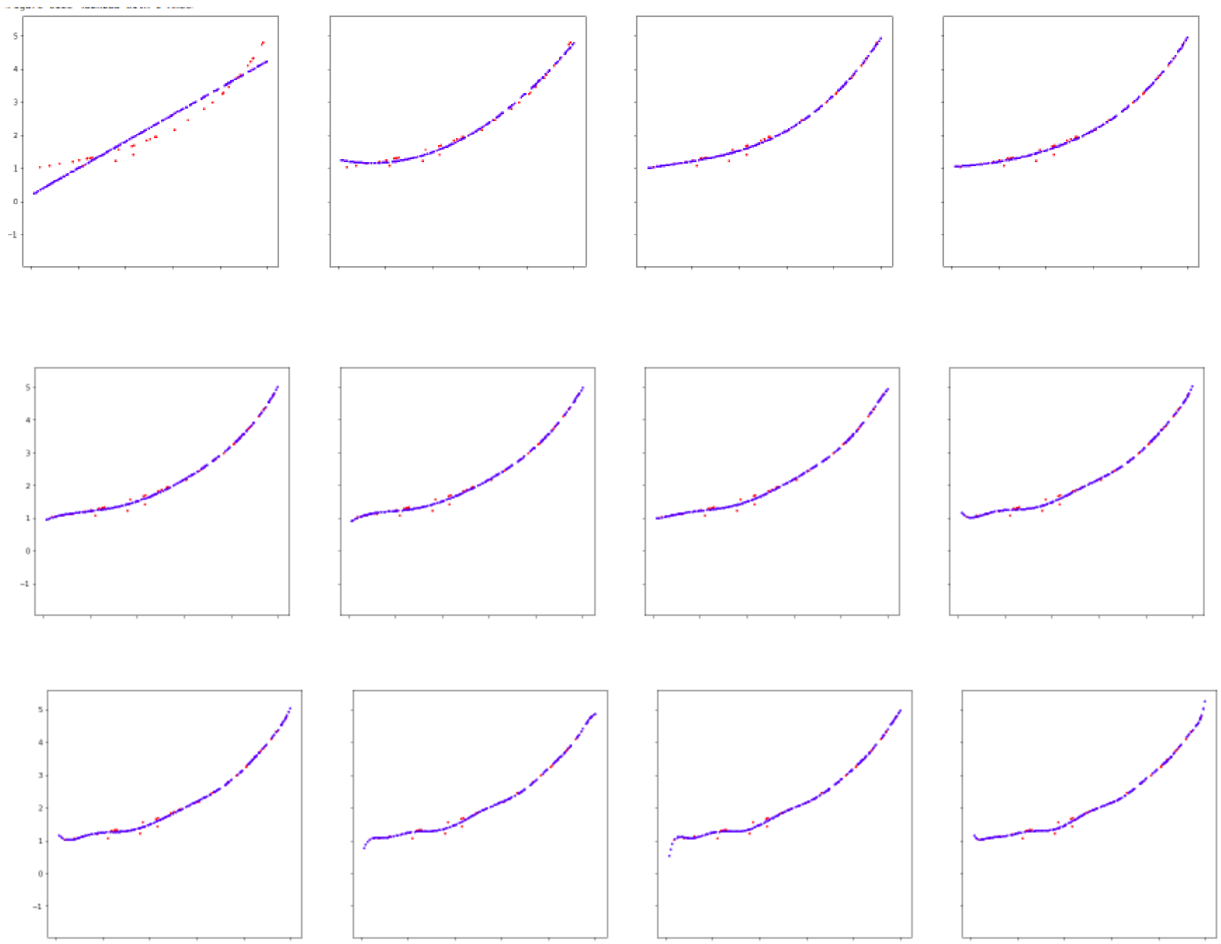
Variance refers to the error that is introduced by the model's sensitivity to small fluctuations in the training data. This is often caused by the model's complexity, where it learns noise as well as signals from the training data. A high-variance model is typically too complex and tends to overfit the data.

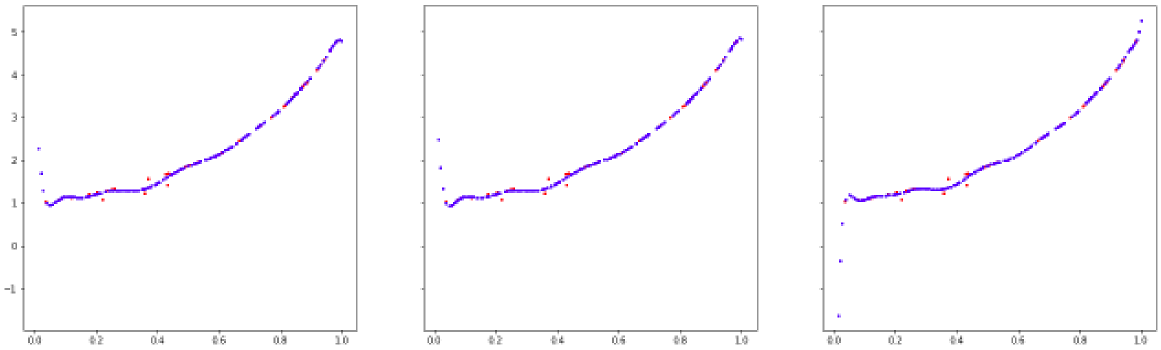
The **irreducible error** in a linear regression model is the part of the error that cannot be reduced, even if the perfect model were used. It is the noise that is inherent in the data and cannot be modeled accurately.

Below are the Predicted values graph for each degree

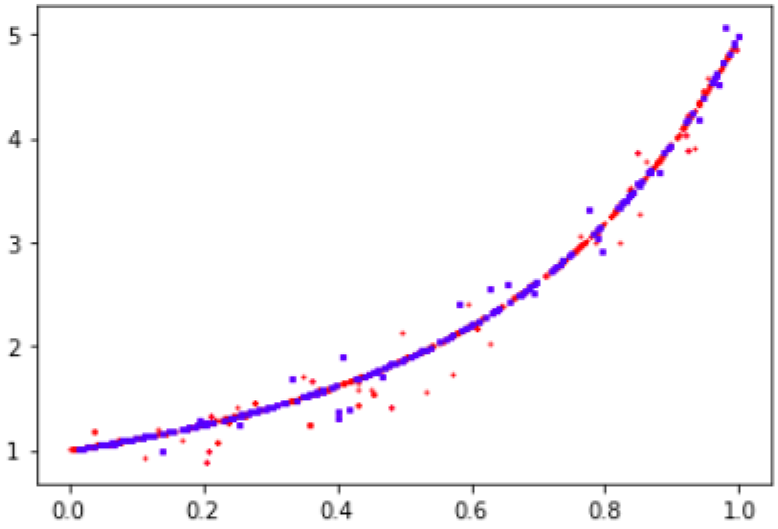
Blue = predicted

red = test set





The graph for Test and Training Data



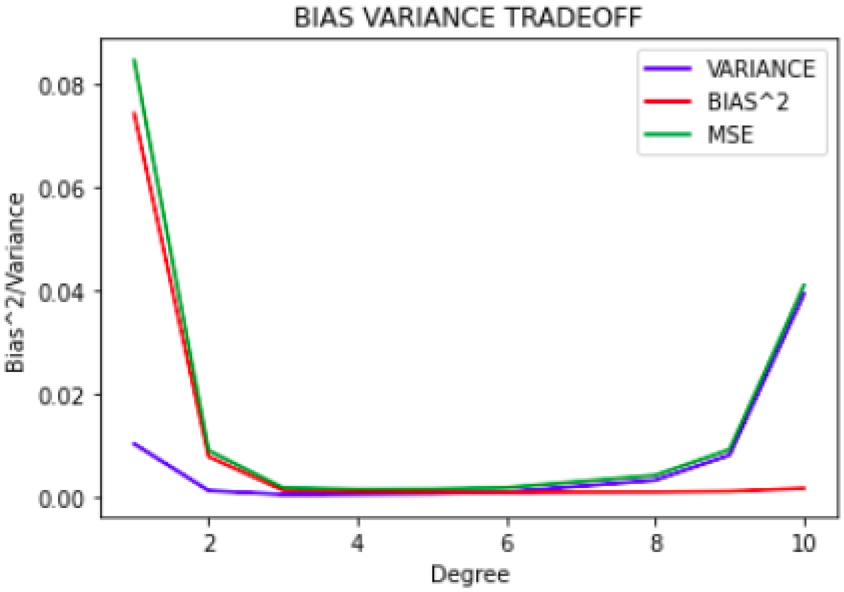
OBSERVATIONS

- We observe from the table for degree 15 the variance increases drastically so the model is Overfit on the training data given , and may not give accurate results on new data .
- We observe that the bias decreases till degree 9 and then increases again till 15

- we observe that Bias value is relatively high at degree 1 , 14 and 15 which makes the model underfit and it is consistently poor performing
- the irreducible error shows no significant pattern , at degree 3 , 6 and 9 the error value is 0 .

Task 5

Bias - variance tradeoff



Task 6 (Bonus)

given is the formula for charge

$$Q = CV_o e^{-t/RC}$$

we take Log both sides , and the equation looks like

$$logQ = logC + logV_o - t/RC$$

We observe that we can generalise this equation as a straight line

$$Y = wX + I$$

where Y = logQ , w = -1/RC , I = logC +logV

Using the Above formulae

Intercept I = LogC + Log5
Weight w = -1/RC

Values of R and C are

$$R = 100000.00000001$$

$$C = 5.0 * 10^{-5}$$