Task: 1

The LinearRegression.fit() is a function from Python's scikit-learn library.
It is used to train a linear regression model on some given data values.

By using LinearRegression, we are creating an instance of LinearRegression.
It has 4 parameters:

1. fit_intercept : boolean, if True, decides to calculate the intercept of model
   Or, if False, considers the intercept to be 0 (False by default)
2. normalize: boolean, if True, decides to normalize the input variables
   Or, if False, don't normalize the input variables (False by default)
3. Copy_x: boolean, if True, decides whether to copy the input variables
   Or, if False, overwrite the input variables (True by default)
4. N_jobs: integer or None, represents the number of jobs in parallel
   computation, (defaults to None means one job).

By using .fit(), we are estimating the parameters of the linear regression model.
We can then use them to predict the target variable of the given test set.
It uses the least square method to find the best-fit line that minimises the sum of squared distance between the predicted values and the actual values.
It has 2 parameters:

1. x, this is the input data
2. y, this is the target variable

Task: 2

We use Gradient Descent in order to minimise the mean squared error (MSE) in our linear regression model. Thus, it helps to find the optimal values of the coefficients.

Now, let us consider the case when there is one independent variable and one dependent variable. Here, the linear regression model uses the equation of a straight line. This linear regression equation is given by the formula y = mx + c
where y is a dependent variable, x is an independent variable,
m is slope and y is intercept.

The algorithm begins with some initial values for m and c and iteratively update their values in order to minimise the cost function, which is the sum of mean squared error between the predicted and actual values of the dependent variable.

1. Let m = 0 and c = 0. Let t be the learning rate(small value like 0.01).
2. Calculate the partial derivative of cost function with respect to m. Let it be fm.
   Calculate the partial derivative of cost function with respect to c. Let it be fc.
   Now, update the values of m and c using the following equation:
   m = m - t * fm   &   c = c - t * fc
3. Repeat the step-2 until our cost function is as small as possible.

Since we reduce the values of m and c, we use the word descent.
Thus, the Gradient Descent gives optimum values of m and c of the linear regression equation. With these values of m and c, we will get the equation of the best-fit line.

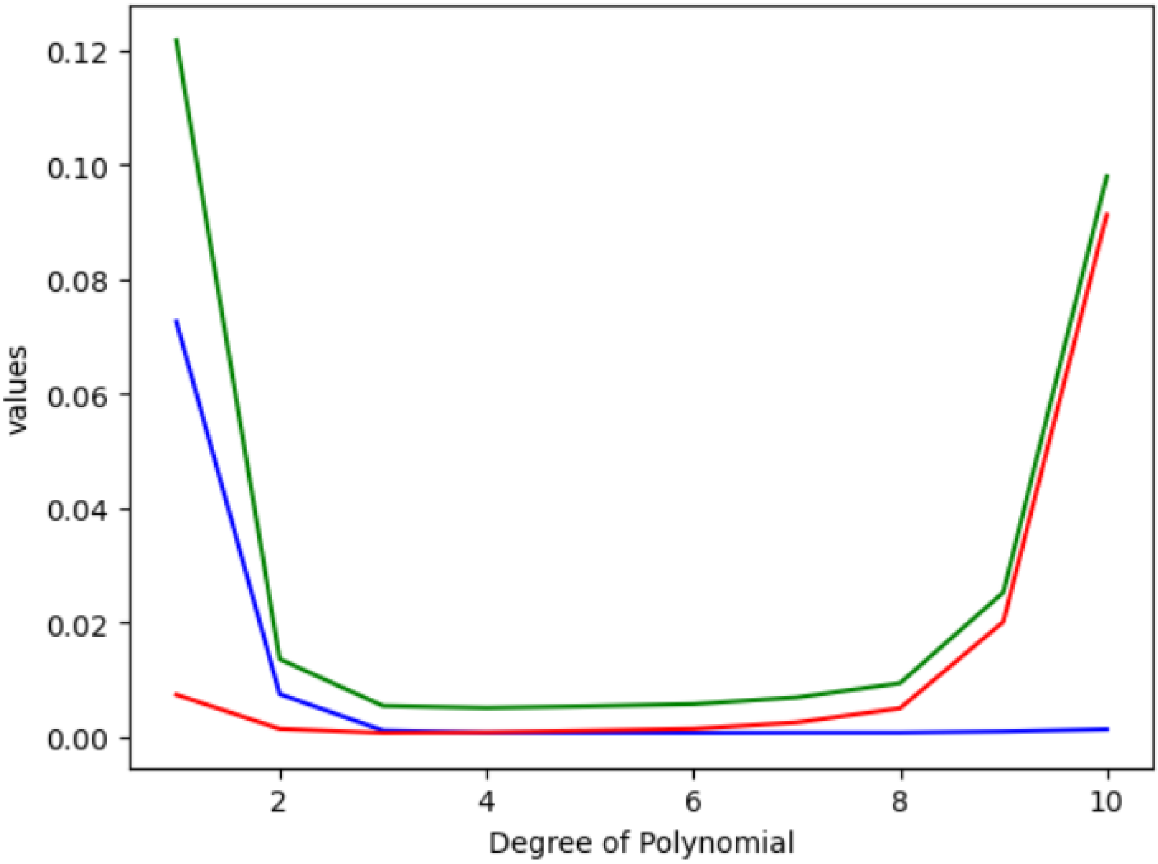
Task: 3

degree	Bias	Variance	Irreducible Error
1	0.269407	0.00736477	7.14229e-17
2	0.0863733	0.00139484	8.01253e-17
3	0.0332028	0.000661871	6.64357e-17
4	0.0262894	0.000764594	3.14429e-17
5	0.0258419	0.0010802	-1.68358e-16
6	0.0260188	0.00141167	6.11775e-17
7	0.0265709	0.0025186	9.6613e-17
8	0.0267022	0.00498205	3.59768e-17
9	0.0305461	0.0201098	3.09618e-17
10	0.0362998	0.0912507	-1.88441e-17
11	0.0514749	0.266064	-1.43113e-16
12	0.0636216	0.556785	4.5006e-17
13	0.0406749	1.51884	-1.81152e-16
14	0.125676	2.909	1.12434e-17
15	0.185591	7.37356	2.20694e-16

The above table consists of bias, variance and irreducible error of the linear regression model as the degree of the polynomial varies.

For lower degrees, the bias of the model is high. But as we increase the degree of the polynomial, we see that the bias of the model decreases.   Whereas, the variance of the model decreases (from its initial value at lower degree) to a minimum (around degree 4) and then gradually increases as the degree of the polynomial increases.

The irreducible error is caused by measurement errors, sampling errors or some other noises. It represents the portion of the dependent variable that cannot be explained by the independent variable and is outside the scope of the model. Its value cannot be reduced by changing the independent variables. Thus, the value of the irreducible error does not vary much as we vary your class function. The negative value of the irreducible error may be due to floating point error.



The blue line represents the Bias^2.
The red line represents the Variance.
The green line represents the Irreducible Error.

For lower degrees, we have high bias and low variance. Thus, the linear regression model is said to be underfitting. For higher degrees, we have low bias and high variance. Thus, the linear regression model is said to be overfitting. But we need to find a good balance without overfitting or underfitting the data. Thus, the linear regression model achieves the Optimal Model Complexity at degree 4.

Bonus

Capacitance(C) = 5e-05
Resistance(R) = 1e5