MDL — Assignment 1 Report

Team 55

Aum Khatlawala — 2020113008

Mulukutla Krishna Praneet — 2020113010

Task 1: Write a brief about what function the method $LinearRegression().fit()$ performs.

$LinearRegression().fit()$ takes in X, Y values, where X is the training data, and Y are target values (i.e the actual outputs for the corresponding inputs in X).

Linear regression performs the task of training/creating a model to predict the dependent variable value (y) based on a given independent variable (x). So, this regression technique finds out a linear relationship between x (input) and y (output).

In more concrete words, single variable linear regression aims to fit the training data X into the equation $y = mx + c$. The algorithm uses the target values Y and corresponding values in X to determine the values of the parameters m and c. As more data points (x_i, y_i) in the training set come in, the values of these parameters (m and c) are fine-tuned further, to make a model that can predict the values within some margin of error.

In the case of multivariable regression models, the algorithm uses the equation $y = m_1x_1 + m_2x_2 + \cdots + m_nx_n + c$, and determines the parameters m_i's and c in a similar fashion.

So, as a result of using this function (taking a single variable model as an example), we have an equation (which we can now call a model) $y = m_{determined}x + c_{deteremined}$, which can then be used to predict the value of y for a given input x.

Task 2: Tabulate the values of bias and variance and also write a detailed report explaining how bias and variance changes as you vary your function classes.

```
degree      bias values         variance values
  1    [161.05627914824498]   [34194.23263561963]
  2    [155.21422143892323]   [67373.94524558309]
  3   [-12.968798272738363]   [67829.79406663417]
  4   [-11.322505725508396]   [99258.35242469373]
  5    [-4.071417574543349]  [113511.57904588434]
  6   [-2.3292653816293383]   [137784.3824536346]
  7     [9.069010875726459]  [157096.80643703646]
  8    [10.950404970094946]  [182770.01727468256]
  9    [12.774473256096893]  [192415.93490720395]
 10    [12.964531888883787]  [213769.35210284073]
 11    [16.239688678147257]   [287497.7817828112]
 12     [18.49914864257475]  [297591.95888557757]
 13     [22.25148173292434]    [345627.231327559]
 14    [17.824507417273214]   [311241.7577996013]
 15    [21.898508150607093]   [333536.3184387727]
```

The values for bias and variance have been tabulated and shown as the output of cells in the code itself. The values tabulated above are for one of the runs of the program.

The trend in bias and variance are very clearly seen in the observations.

The bias decreases as the model complexity increases till a certain point and then fluctuates a bit around that value for the remaining complexities.

The variance increases as the model complexity increases.

These observations are in line with what is expected as per the theory.

Why do we see these trends though? Let's have a look at what bias and variance actually calculate.

The bias term measures the error of estimations (made by the model) and the variance term describes how much the estimation (made by the model) is distributed around its mean. Bias is basically the error stemming from incorrect assumptions in the learning algorithm. Variance measures how sensitive the model prediction is to variations in the data set.

Trends in Bias:

For lower order polynomials, since the model is too simple, the model cannot account for the complexity of the data and thus, we see higher bias.

Conversely, for higher-order polynomials, since the model too accurately accounts for the complexity of the data, we see lower bias. Once a particular polynomial degree model reaches the least MSE, the bias fluctuates around the bias value for that polynomial degree model.

Trends in Variance:

For lower order polynomials, the model does not overfit on the noise in the split data. As a result, the predictions of each of the 16 different models of the same polynomial degree are not too different from each other (close to the mean of the predictions). This results in lower variance.

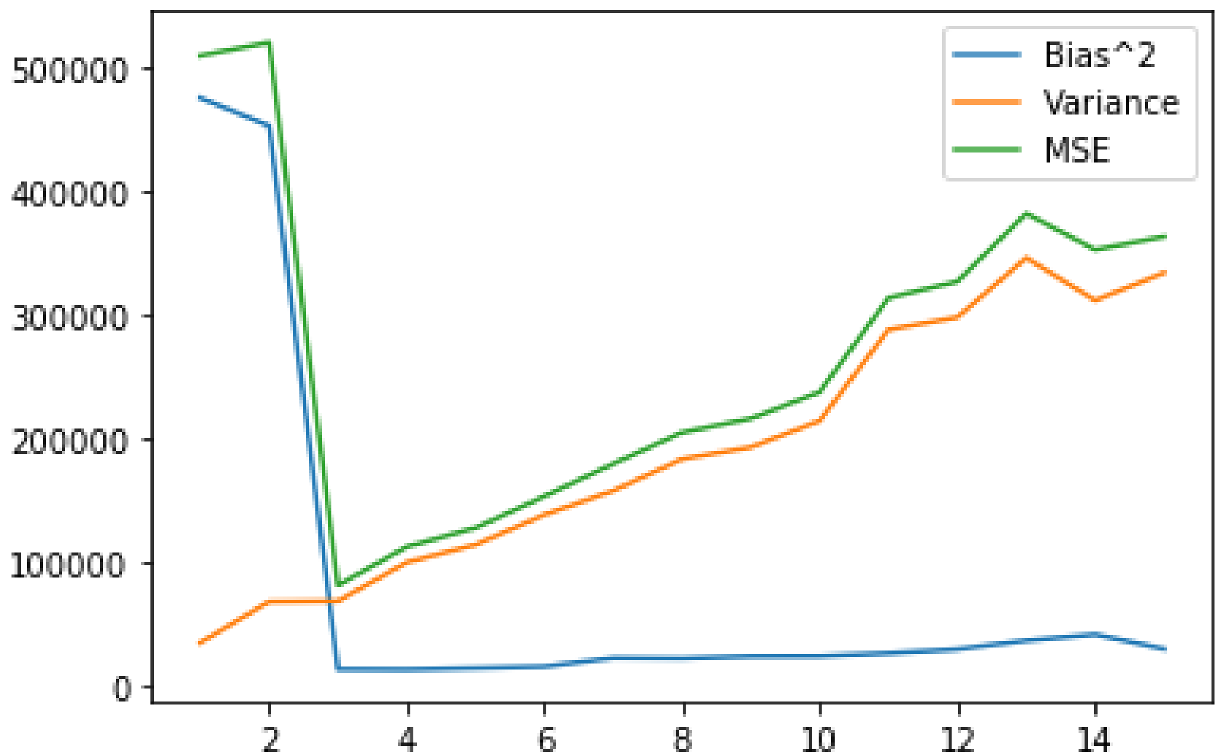
Conversely, for higher-order polynomials, the model overfits on the noise in the split data. As a result, the predictions of each of the 16 different models of the same polynomial degree are very different from each other (not close to the mean of the predictions). This results in higher variance.

Task 3: Tabulate the values of irreducible error for the models in Task 2 and also write a detailed report explaining why or why not the value of irreducible error changes as you vary your class function.

```
degree           irreducible error
  0          [5.820766091346741e-11]
  1        [-4.0745362639427185e-10]
  2         [1.4551915228366852e-11]
  3         [2.9103830456733704e-11]
  4          [5.820766091346741e-11]
  5         [2.9103830456733704e-11]
  6          [8.731149137020111e-11]
  7         [-5.820766091346741e-11]
  8        [-2.9103830456733704e-11]
  9        [-1.1641532182693481e-10]
 10         [1.1641532182693481e-10]
 11        [-1.1641532182693481e-10]
 12          [5.820766091346741e-11]
 13                            [0.0]
 14         [1.7462298274040222e-10]
```

In theory, the irreducible error shouldn't change along with the model complexity as it is an inherent error in the data that doesn't depend on how good our model is. Irreducible error is a measure of the amount of noise in the data. As per the calculations made in this Assignment, we can conclude that the irreducible error changes by such a small margin (in the order of 10^{-11} units) that it can be safely inferred to be **constant** (i.e it follows the theory). The individual values for irreducible errors for each degree polynomial is also close to 0 as the values are of the order 10^{-10} units.

Task 4: Based on the variance, bias and total error calculated in earlier tasks, plot the $Bias^2 - Variance$ tradeoff graph and write your observations in the report with respect to underfitting, overfitting and also comment on the type of data just by analyzing the $Bias^2 - Variance$ plot.



As we can see from the above graph, the total error (MSE) is minimum at degree 3, i.e, the sum of irreducible error, the square of bias and variance is minimum for a degree 3 model for this data set.

As mentioned earlier, the general trend is that the value of the square of bias decreases with increase in the complexity of the model and the value of variance keeps increasing with increase in the complexity of the model.

Underfitting:

The models with degree less than 3 (i.e degree 1 and 2) underfit the data since they are too simple to model the complexity of the data. Thus, we observe that they have high bias and low variance, which are indicators of underfitting when tested on the test data. This means that these models fail to capture the underlying trend of data.

Overfitting:

The models with degree more than 3 (i.e degrees 4-15) overfit the data since they are complex models that accommodate even the noise present in the data set into the model. When such models are tested on test data, we observe low bias and high variance, which are indicators of overfitting. Overfitting is akin to memorizing all data points on the training set, leading it to perform worse on an unseen dataset like the test data set.

Type of data:

We observe that the data is a polynomial function of degree 3, of the form $a_3x^3 + a_2x^2 + a_1x + a_0$. (we conclude this, as the MSE value is least for the degree 3 model).