2021101076
Anuhya Nallapati

MDL Assignment-1 Report

Task 1:

Write a brief about what function the method LinearRegression().fit() performs.

LinearRegression fits a linear model with coefficients w = (w1, …, wp) to minimize the residual sum of squares between the observed targets in the dataset, and the targets predicted by the linear approximation.

The LinearRegression().fit() is from the SciKit.linear_model library.
It is used as LinearRegression().fit(X, y) where X represents training data and Y represents target values. It returns a fitted estimator.

LinearRegression().fit() is an algorithm to find the coefficients of the linear regression model that best fit the training data.

Task 2:

Explain how gradient descent works to find the coefficients. For simplicity, take the case where there is one independent variable and one dependent variable.

Gradient Descent is used to find the coefficients of a linear regression model that best fits the training data.

When there is one independent variable and one dependent variable, the linear regression model is as follows:

$y=ax+b$

where y is the dependent variable
x is the independent variable
a is the slope of the line
b is the y-intercept

Now, we need to find the values of a and b that minimize the sum of squared errors between the predicted values and the actual values of the dependent variable in the training data.

We initially take random values for a and b and then update these values iteratively based on the cost function. The cost function refers to MSE(the average of the squared differences between the predicted and actual values of the dependent variable in the training data). At each iteration, we update a and b as follows

b=b-alpha*(1/N)*sum(y_pred-y_actual)
a=a-alpha*(1/N)*sum((y_pred-y_actual)*x)
where
alpha is the learning rate (typically set to a small value)
N is the number of training examples
y_pred is the predicted value of y
y_actual is the actual value of y

This algorithm continues to update the values of a and b until a maximum number of iterations are reached or until the change in the cost function is below a predefined threshold.

Task 3:

Tabulate the values of bias and variance.

For unshuffled data, we get the values of bias and variance as follows

```
TASK 3

Values of Bias and Variance

degree 1  bias->0.26939817862780674  variance->0.00868095161648306
degree 2  bias->0.08625653626329852  variance->0.001224357844645493
degree 3  bias->0.03327182734458063  variance->0.00033733942997527894
degree 4  bias->0.024282630552664487  variance->0.0003669994779255686
degree 5  bias->0.02387931868714452  variance->0.0004619378654759439
degree 6  bias->0.02395536905119095  variance->0.0005815200868688619
degree 7  bias->0.024830004382679502  variance->0.000916796100938208
degree 8  bias->0.024887389385694993  variance->0.0017609193269420595
degree 9  bias->0.030418442333700037  variance->0.00827682989110048
degree 10  bias->0.028663868274853135  variance->0.006500850640588358
degree 11  bias->0.03659031702793742  variance->0.03269275421182073
degree 12  bias->0.07091716143450878  variance->0.8844942809310076
degree 13  bias->0.04224905611828066  variance->0.05431697879461794
degree 14  bias->0.15642832764743492  variance->8.04334545927474
degree 15  bias->0.0891286756264066  variance->2.6695306813686837
```

Write a detailed report explaining how bias and variance change as you vary your function classes.

As the degree of the polynomial increases, bias and variance change in different ways.
We observe that the bias decreases from degree 1 to degree 4. When the degree of the polynomial is small, the models are underfit and have a high bias. But, as the degree increases from 1 to 4 they perform more stably so the bias decreases. Then the bias doesn't change much till degree 10 because the models fit the training data well. Then the bias increases from degree 10 to 15 as the models get more complex
We observe that the variance values are mostly consistent till degree 12 after which the variance increases. As the size of the training data is small even though the value of variance usually increases with degree the variance remains mostly consistent. From degree 13 onwards due to the random noise there is an increase in variance.

Task 4:

Tabulate the values of irreducible error for the models.

For unshuffled data, we get the values of irreducible errors as follows

```
TASK 4

Values of Irreducible Error for the models

degree 1  irreducible error->[-1.50053581e-17]
degree 2  irreducible error->[-5.78584489e-18]
degree 3  irreducible error->[6.46306468e-18]
degree 4  irreducible error->[1.75207071e-18]
degree 5  irreducible error->[-3.14296657e-18]
degree 6  irreducible error->[-3.18755439e-19]
degree 7  irreducible error->[8.45440525e-18]
degree 8  irreducible error->[-3.332973e-18]
degree 9  irreducible error->[-2.66171633e-18]
degree 10  irreducible error->[3.02146817e-18]
degree 11  irreducible error->[-6.92053023e-18]
degree 12  irreducible error->[-1.62694023e-16]
degree 13  irreducible error->[-5.20904934e-18]
degree 14  irreducible error->[1.6300031e-15]
degree 15  irreducible error->[5.46661105e-16]
```

Write a detailed report explaining why or why not the value of irreducible error changes as you vary your class function.

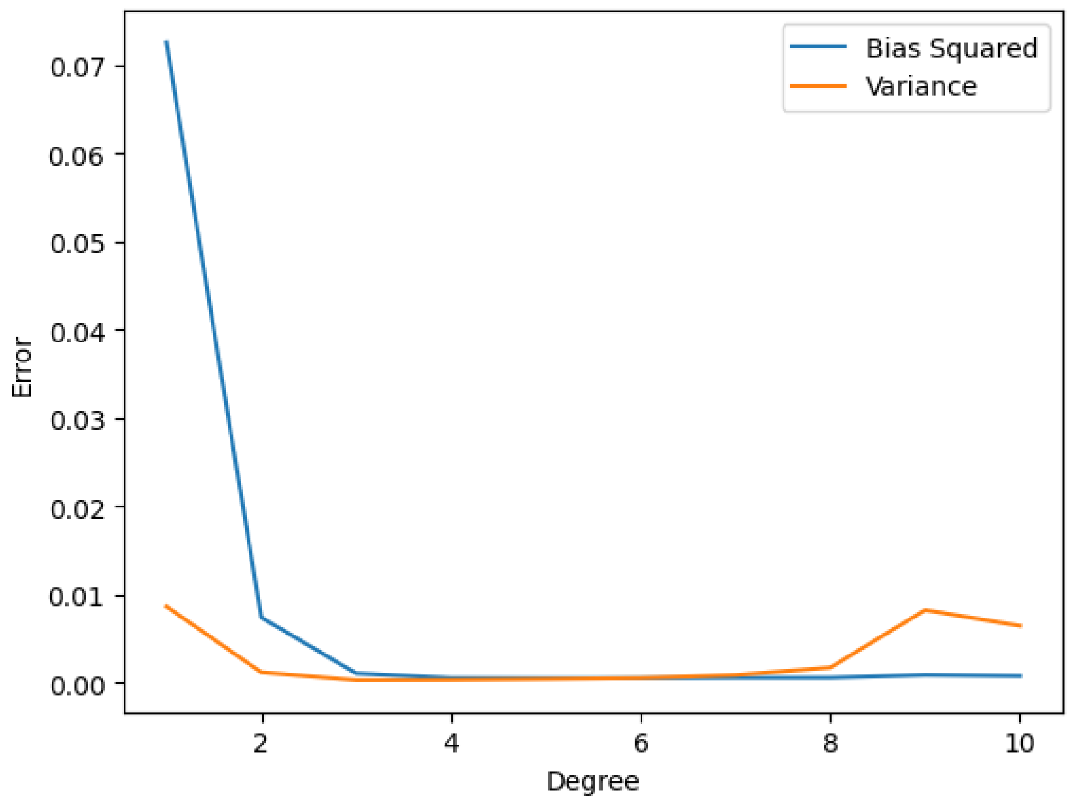
The irreducible error is mostly fixed and doesn't change with the degree. This is because irreducible error is related to the pre-present error and is therefore not affected by the degree of the polynomial. So, the value does not change as the degree changes regardless of how good our model is due to the noise always present.

Task 5:

Plot the Bias^2−Variance tradeoff graph

For unshuffled data, the Bias^2-Variance graph is as follows

```
Bias^2−Variance Tradeoff Graph
```



Write your observations with respect to underfitting, overfitting and also comment on the type of data just by analyzing the Bias^2−Variance plot.

Underfitting (higher bias, lower variance)
Here, the models have a low complexity and are not able to capture data accurately resulting in large total errors, a high bias^2 graph, and low variance graph.

Overfitting (lower bias, higher variance)
Here, the models have a high complexity and they have a low bias^2 graph, and a high variance graph.

From degrees 1 to 4 the bias decreases and variance increases and later on the bias increases and the variance decreases. So, the data becomes overfitting from being underfitting. Models with degrees 4 to 7 fit well because they have both low variance and low bias.

Plot variation of Bias^2, Variance and MSE against degree of polynomial in the same Graph. (You need to plot the graph for polynomials of up to degree 10 only)

The graph of variation of Bias^2, Variance and MSE against degree of polynomial for unshuffled data is as follows

```
Variation of Bias^2, Variance and MSE against degree of polynomial
```