Task 1 (Linear Regression)

LinearRegression() minimizes the residual sum of squares between the observed targets in the dataset and the targets predicted by the linear approximation by fitting a linear model with coefficients w = (w1,..., wp). The LinearRegression().fit() function implements the Ordinary Least Squares (OLS) method to estimate the unknown parameters in the linear regression model.

LinearRegression().fit() returns a trained model on the dataset sent to this method as an argument. When given the training set as input, this method calculates the model's output. The output is then compared to the actual values in the dataset.

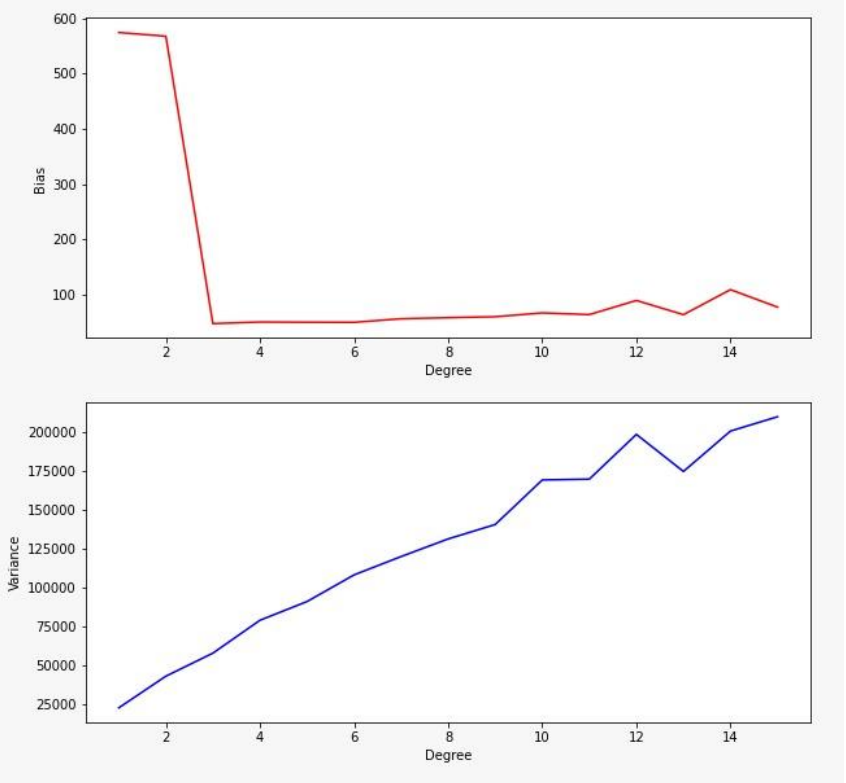
It tries to fit the linear equation $y = \mathbf{\Sigma} a_i x_i + b$ with the best value of a_i and b such that the sum of squares of the difference between predicted value (y) and real value is minimum with the input value (x_i) .

Task 2 (Bias and Variance)

Bias is a measure of how much the value predicted by our model differs from the actual value; a high bias describes a huge variation of the predicted values from the actual value of the data set. It is defined as the difference between the average prediction of our model and the correct value that we are trying to predict.

Variance, on the other hand, is a metric for determining the dispersion of data. A larger value of variance indicates a large fluctuation of the anticipated value around the dataset's actual value. This is supported by the fact that as the degree increases, the number of complex polynomial terms increases, resulting in more volatility.

After spending hours trying to train his model on several classifiers and at various degrees, Bajaj plotted the following... well, cool little graphs.



He had a eureka moment when he observed:

1. *The biases of his models had a sharp decline after the 3rd-degree polynomial. He noted that the model was underfitting the data for 1st and 2nd-degree polynomials. This led to high bias while training on lower degree polynomials. The real data set values varied a lot from the mean which really worried Bajaj.*
2. *The variances of his models kept increasing with the complexity of the higher degree of polynomials. The more parameters we add to the equations, the more spread out the data set*

seems to be! Bajaj used his realtor's experience to note that a high variance also brought high risk to the table, but here's a twist - it also comes with high reward and low bias. So, he was a little clueless here! Should Bajaj be glad or sad?!

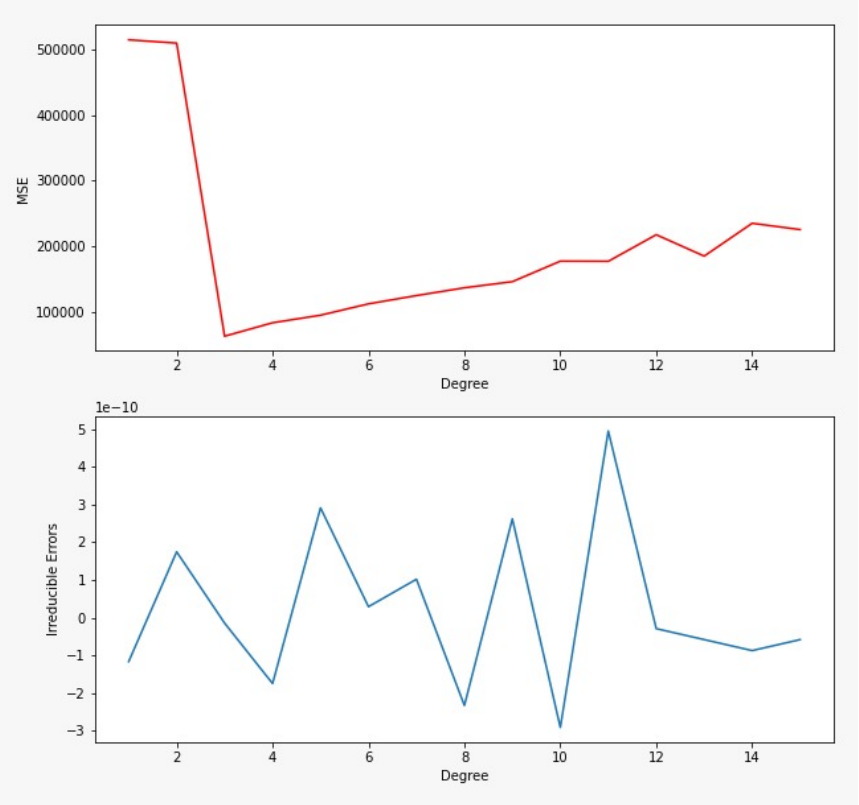
Task 3 (Irreducible Error)

Regardless of how thoroughly the model is trained, there will always be some inaccuracy owing to data noise that cannot be avoided (that is, the extrinsic factors that impact the output and consequently generate mistakes, in addition to existing parameters in the data set). As we change the class function, tinker around with the degree, the biases, and the variances tend to follow an uncanny pattern. This, in turn, causes the irreducible error to vary.

$$\text{Irreducible Error} = \text{MSE} - (\text{Bias}^2 + \text{Variance})$$

Here, Bias $\propto$ 1/Variance. So, finding a good balance between them is crucial.

Bajaj stole the Irreducible Error's formula from DuckDuckGo and substituted his values in them. He found that his irreducible error values were around 1e-11 which he really wished he could overcome. But he was very tired after a long day, so he decided to call it a night.



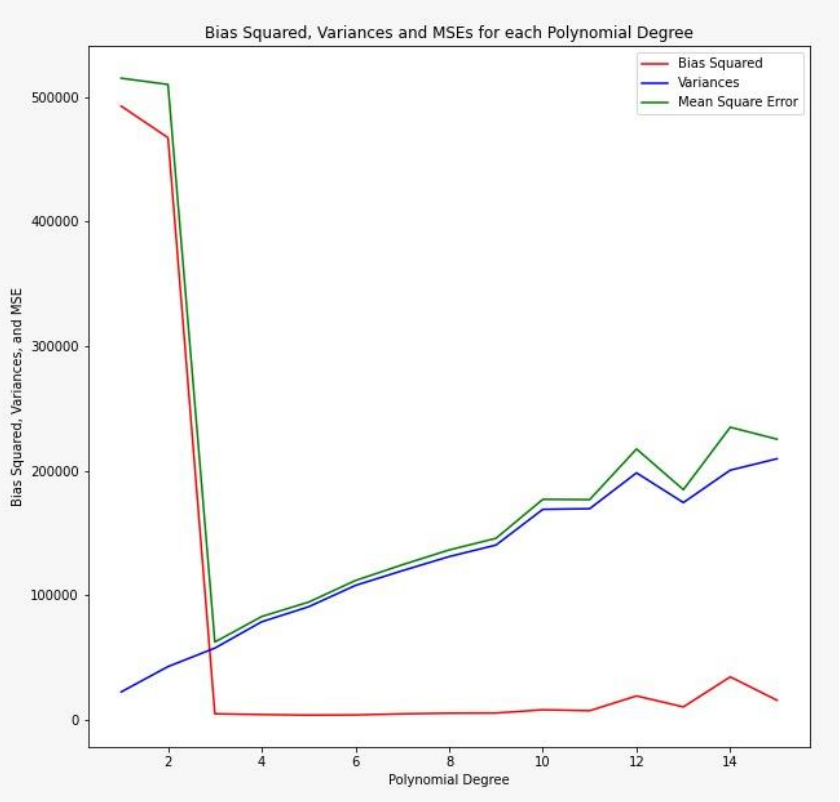
Task 4 (Bias^2-Variance Tradeoff)

Bias is the model's assumptions that simplify the target function, making it easier to estimate whereas variance refers to how much the target function's estimate will fluctuate as a result of varied training data. The tension between the bias-induced error and the variance is known as a trade-off.

We need such a value that the model neither learns from the noise (overfit on data) nor makes broad assumptions on the data to achieve a balance between the Bias and Variance errors (underfit on data).

Because of underfitting, the bias for the given test data set is initially high, then reduces to the best value, then rises again when the model adheres too closely to the test data and loses its generality, causing it to perform badly.

Because the best fit curve overfits the training data, the variance continues to rise (beyond degree 3), resulting in an erroneous representation of the test data while also lowering the model's precision.



Presented by
MAYHEM 53

Starring
Radwait *played by* Sanyam Shah
Radwait's Girlfriend *played by* Freyam Mehta