Details of Algorithm :

Linear Regression is a machine learning algorithm based on supervised learning. It performs a regression task. Regression models a target prediction value based on independent variables. It is mostly used for finding out the relationship between variables and forecasting.Linear regression performs the task to predict a 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).

Hypothesis Function for Linear Regression:

- $Y = \theta_1 + \theta_2 * X$

While training the model we are given :
x: input training data (univariate – one input variable(parameter))
y: labels to data (supervised learning)

When training the model – it fits the best line to predict the value of y for a given value of x. The model gets the best regression fit line by finding the best θ1 and θ2 values.
θ_1: intercept
θ_2: coefficient of x
Once we find the best θ_1 and θ_2 values, we get the best fit line. So when we are finally using our model for prediction, it will predict the value of y for the input value of x.

Cost Function (J):
By achieving the best-fit regression line, the model aims to predict y value such that the error difference between predicted value and true value is minimum. So, it is very important to update the θ1 and θ2 values, to reach the best value that minimize the error between predicted y value (pred) and true y value (y).
Cost function(J) of Linear Regression is the Root Mean Squared Error (RMSE) between predicted y value (pred) and true y value (y).

Gradient Descent:
To update θ_1 and θ_2 values in order to reduce Cost function (minimizing RMSE value) and achieving the best fit line the model uses Gradient Descent. The idea is to start with random θ_1 and θ_2 values and then iteratively updating the values, reaching minimum cost.

Implementation:

Following in built functions are used in code:

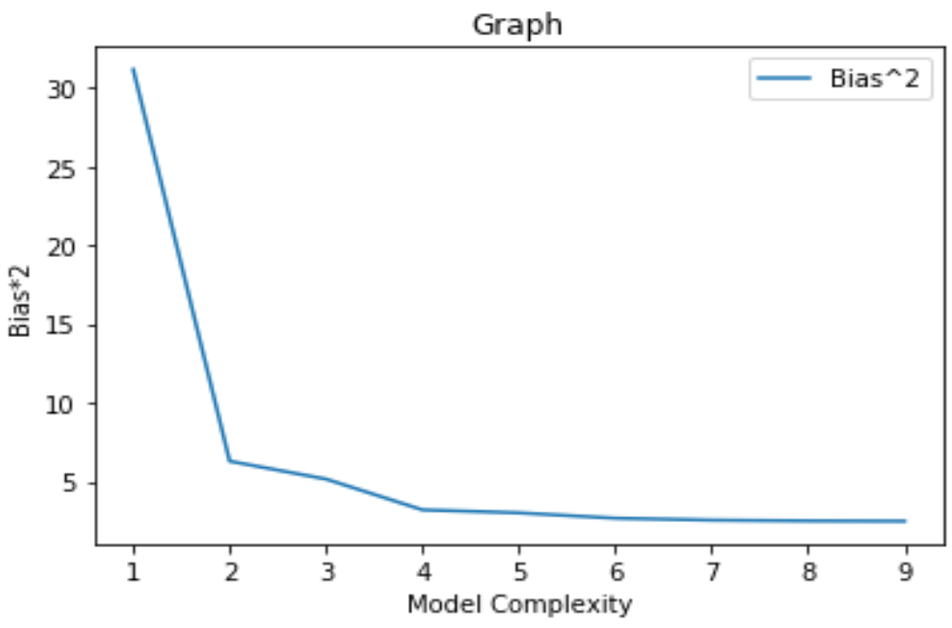
- sklearn’s linear_model.LinearRegression().fit() : Implementation of Linear Regression Algorithm.
- pickle.load() : For Loading of Data from Pickle file.
- sklearn’s train_test_split : For splitting of training and testing data.
- sklearn’s preprocessing.PolyomialFeatures : For converting data to different degrees.

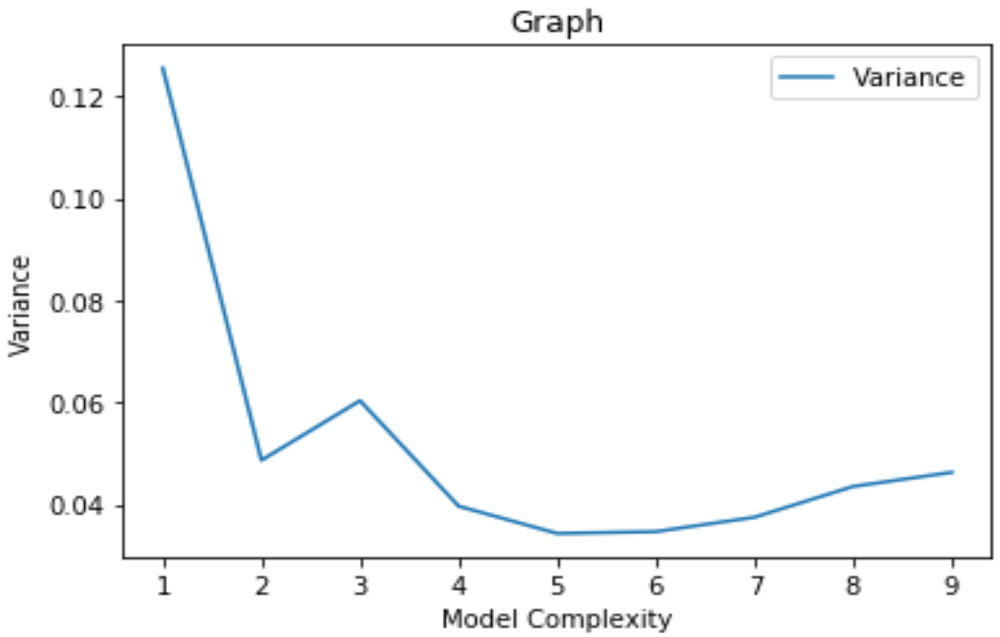
Results :

Q1: **Table of Bias and Variance with increasing Model Complexity.**

Bias	$Bias^2$	Variance
5.583	31.171	0.125
2.507	6.286	0.048
2.269	5.151	0.060
1.787	3.196	0.039
1.733	3.006	0.034
1.632	2.664	0.034
1.594	2.543	0.037
1.579	2.495	0.043
1.574	2.478	0.046

Graphs:





Explanation :

Bias : Bias is the simplifying assumptions made by the model to make the target function easier to approximate

Variance : Variance is the amount that the estimate of the target function will change given different training data

- Initially when the degree of polynomial is very less at that time our function is too simple and such a simple model can not fit our training dataset. So it will cause under-fitting. This will give very bad accuracy on training dataset therefore initially value is bias is too high. And at this time as model is not good therefore accuracy on test dataset is also very low i.e with simpler models accuracy is too low with both training and test dataset so variance is too low which in above graph is for degree 1,2,3.
- As we go on increasing the complexity of models , training accuracy goes on increasing till some limit as models fit more accurately to training dataset. So at that time training accuracy increases therefore bias decreases and also test accuracy also goes on increasing as model is good. So both training and test accuracy are good at this time therefore the value of variance is low which in the above graph is near 5.
- As we go on further complexity of model increases too much which gives very high accuracy on training dataset but this model is overfitted model because for this model training accuracy is too high but test accuracy is very bad. This is because as we increase the complexity of the model it tries to best fit the training data points. Therefore due to very large complexity it tries to exactly fit the training data points which results in very bad over-fitted model which give very bad test accuracy. Therefore at this point bias is too low but variance is too high because of the large difference between training and test accuracy which in above graph is 6 onwards.
- **Exception** : - If we remove the hills and valleys in the given polynomial of data points, we notice the given polynomial has roughly 3 inflection points. Therefore, it resembles a 5-degree polynomial with hills and valleys.Thus, variance decreases till degree 5. However, the later degrees start overfitting a lot more and therefore have high variance.

Type of Data : By looking at bias-variance curver we can say that our data points form a polynomial equation of degree 5 because at degree 5 both values of bias and variance are optimal. Neither model is overfitting and under-fitting.

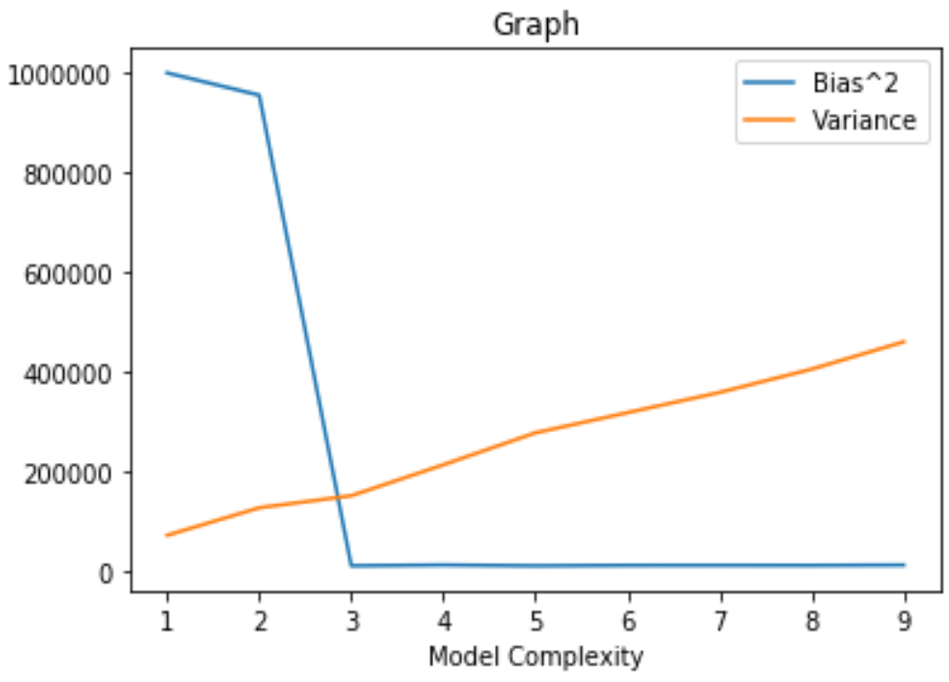
Q2.

Table of Bias and Variance with increasing ModelComplexity.

Bias	$Bias^2$	Variance
999.614	999228.396	70545.489
977.046	954619.273	125870.855
96.900	9389.730	150073.739
104.438	10907.348	212235.708
96.639	9339.194	276388.480
101.235	10248.585	316863.498
101.662	10335.275	357510.984

100.744	10149.419	404286.670
103.997	10815.487	459132.378

Graph:



Explanation :

Bias : Bias is the simplifying assumptions made by the model to make the target function easier to approximate
Variance : Variance is the amount that the estimate of the target function will change given different training data

- Initially when the degree of polynomial is very less at that time our function is too simple and such a simple model can not fit our training dataset. So it will cause under-fitting. This will give very bad accuracy on training dataset therefore initially value is bias is too high. And at this time as model is not good therefore accuracy on test dataset is also very low i.e with simpler models accuracy is too low with both training and test dataset so variance is too low which in above graph is for degree 1 and 2.
- As we go on increasing the complexity of models , training accuracy goes on increasing till some limit as models fit more accurately to training dataset. So at that time training accuracy increases therefore bias decreases and also test accuracy also goes on increasing as model is good. So both training and test accuracy are good at this time therefore the value of variance is low which in the above graph is near 3.
- As we go on further complexity of model increases too much which gives very high accuracy on training dataset but this model is overfitted model because for this model training accuracy is too high but test accuracy is very bad. This is because as we increase the complexity of the model it tries to best fit the training data points. Therefore due to very large complexity it tries to exactly fit the training

data points which results in very bad over-fitted model which give very bad test accuracy. Therefore at this point bias is too low but variance is too high because of the large difference between training and test accuracy which in above graph is 4 onwards.

Type of Data : By looking at bias-variance curver we can say that our data points form a polynomial equation of degree 3 because at degree 3 both values of bias and variance are optimal. Neither model is overfitting and under-fitting.