CS7.301 Machine, Data and Learning

P Meena Raja Sree Naren Akash R J

2018101118 2018111020

International Institute of Information Technology Hyderabad

Keywords: Machine Learning, Data Sampling, Bias-Variance Tradeoff, Overfitting

Libraries: numpy, matplotlib, pretty-table, pickle, sklearn

Prediction Errors: Bias and Variance

Understanding how different sources of error lead to bias and variance helps us improve the data fitting process resulting in more accurate models.

The error due to bias is taken as the difference between the expected prediction of our model and the correct value which we are trying to predict. Each time we gather a new training data and run a new analysis creates a new model. Due to randomness in the underlying data sets, the resulting models will have a range of predictions. Bias measures how far off in general these models' predictions are from the correct value.

$$Bias = E[\hat{f}(x)] - f(x)$$

$$Bias^2 = (E[\hat{f}(x)] - f(x))^2$$

The error due to variance is taken as the variability of a model prediction for a given data point. Again, imagine you can repeat the entire model building process multiple times. The variance is how much the predictions for a given point vary between different realizations of the model.

$$Variance = E[(\hat{f}(x) - E[\hat{f}(x)])^2]$$

Regression

Regression analysis is a form of predictive modelling technique which investigates the relationship between a dependent and independent variable. There are many types of regressions – linear regression, polynomial regression, logistic regression, etc. In our analysis, we use only linear and polynomial regression.

Calculating Bias and Variance

The dataset is loaded into the Python program using pickle library function. The dataset is split into training and testing set in the ratio 90:10. Further dividing the training set into 10 equal parts, we get 10 different training datasets.

Train 01	Train 02	Train 03	Train 04	Train 05	Train 06	Train 07	Train 08	Train 09	Train 10	Test Set

For each of the training subsets we follow the following algorithmic implementation to get respective models using linear regression.

1. Create a linear regression object using sklearn's linear_model.LinearRegression()
2. Train the regression model using the fit() method of the object obtained above.
3. Make predictions using predict() method of the object after passing the test set.

For polynomial regression,

1. Setup polynomial transformation of the corresponding degree using sklearn's preprocessing.PolynomialFeatures() method.
2. Apply polynomial transformation to the test set using the above object's fit_transform() method to obtain the transformed test set.
3. Now, follow the same steps of the linear regression using the new test set.

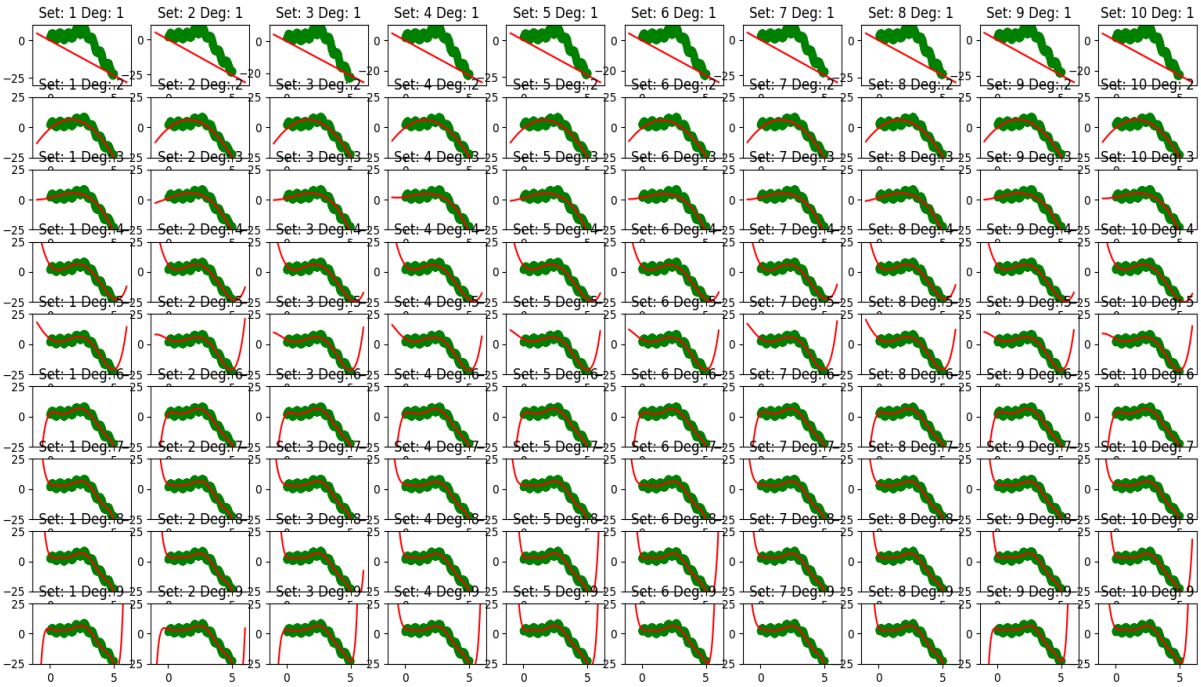
Bias, Bias^2 and Variance values are calculated as mentioned in the previous page. $Eff(x)]$ refers to the average predicted value of each datapoint for each of the model.

Results

Degree	Average Bias	Average Bias^2	Average Variance
1	0.385	31.099	0.2437
2	-0.016	6.084	0.0423
3	0.067	5.498	0.0545
4	-0.024	3.178	0.0333
5	-0.022	3.09	0.0322
6	-0.013	2.724	0.029
7	-0.021	2.509	0.0322
8	-0.015	2.508	0.0408
9	-0.014	2.5	0.0467

Observations

As the table shows, a model with high complexity can achieve low training error but can fail to generalize to the test set because of its high model variance. On the other hand, a model with low complexity will have low model variance but can also fail to generalize because of its high model bias. To select a useful model, we must strike a balance between model bias and variance.



1. The average bias^2 value is steadily decreasing as the model complexity increases. The decrease from function class of degree 1 to degree 2 is very sharp.
2. The average variance sharply decreases from function class of degree 1 to 2 and increases from function class of degree 2 to 3. It falls from degree 3 to 4 and maintains approximately the same value from function class of deggree 4 to 7. Then, the value increases slowly for function class of degree 8 and 9.
3. The bias2 and variance of the dataset are generally small which makes it a good dataset.
4. Models of degree 1 to 6 and degree 8 to 9 have higher total error from a top-view. Model from function class of degree 7 seems to be good fit.

Model Complexity and Bias-Variance Tradeoff

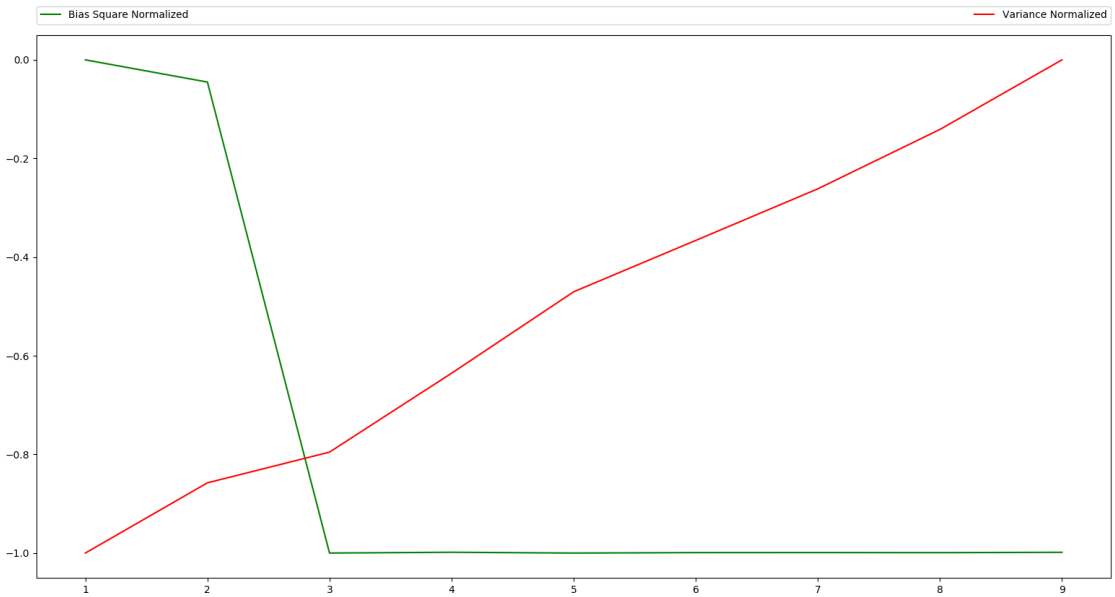
Here, we are given 20 different train datasets and a single test set. We run linear regression and polynomial regression for degrees from 2 to 9. We follow the same algorithmic implementation as mentioned for the previous dataset.

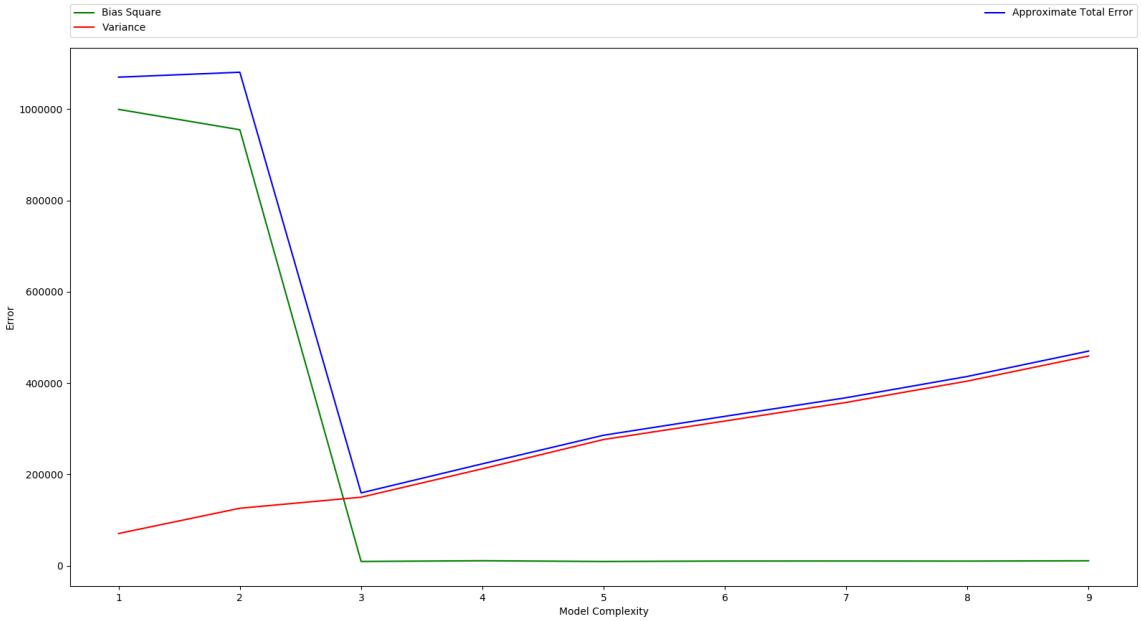
The values for Average Bias, Average Bias^2, and Average Variance for each class of functions are calculated. We use the formulae on the first page of the report for calculation.

The Average Bias^2 and Average Variance for each class of functions are normalized to the range from 0 to 1 using Min-Max Normalization (soley for visualization purpose) and plotted.

Results

Degree	Average Bias	Average Bias^2	Average Variance
1	228.639	999228.397	70545.4891
2	228.639	954619.274	125870.8555
3	-13.552	9389.73	150073.7395
4	-11.707	10907.348	212235.7083
5	-9.309	9339.194	276388.4803
6	-12.752	10248.586	316863.4984
7	-8.446	10335.276	357510.9848
8	-8.112	10149.419	404286.6707
9	-7.773	10815.487	459132.3784





Understanding bias and variance is critical for understanding the behavior of prediction models, but in general what you really care about is overall error, not the specific decomposition. The sweet spot for any model is the level of complexity at which the increase in bias is equivalent to the reduction in variance. If our model complexity exceeds this sweet spot, we are in effect over-fitting our model; while if our complexity falls short of the sweet spot, we are under-fitting the model.

Observations

1. Models of degree 1 and 2 have high variance even though they have low $bias^2$ values. This indicates under-fitting.
2. Models of degrees from 4 to 9 have low variance but the $bias^2$ value is high. This indicates over-fitting.
3. Model of degree 3 seems to be a good-fit from the plot as it has reasonably lower $bias^2$ and a reasonable variance. The approximate total error which is the sum of $bias^2$, variance and noise is the lowest at the model of degree 3.
4. The $bias^2$ and variance of the dataset are generally too high which indicates a not so good dataset.