# Bias-Variance Trade-Off

**Bias:**

Bias is the difference between the Predicted Value and the Expected Value.

Being high in biasing gives a large error in training as well as testing data. Its recommended that an algorithm should always be low biased to avoid the problem of **underfitting**.  
By high bias, the data predicted is in a straight-line format, thus not fitting accurately in the data in the data set. Such fitting is known as **Underfitting of Data**. This happens when the hypothesis is **too simple or linear in nature**

Bias is inherent to the algorithm we choose to make the Model. A biased model is one that makes incorrect assumptions about the dataset to make the target function easier to learn.

For example, suppose that we use a linear regression model on a data that has cubic relationship. This model will be biased because we have taken wrong assumption about the data and model would be trained using this wrong assumption about data.

**Low Bias is desired: High Bias means Underfitting of the model on training data.**

**Model with high bias is very simple** because of the assumptions made about the data (which may or may not be true) **pays very little attention to the training data and oversimplifies the model. High Bias model always leads to high error on training as well as on test data.**

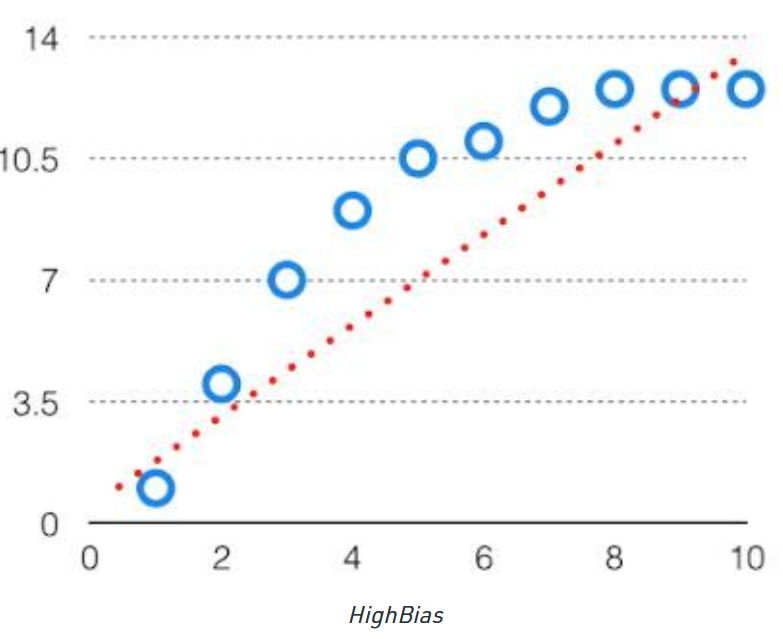
In such a problem, a hypothesis looks like

**Y = f(X) + e**

Given points x, a true value f(x), and a model f̂ (x), bias can be expressed mathematically as:

Bias [f̂ (x)]=E[f̂ (x)−f(x)]

Where E[⋅] is the [expected value](https://en.wikipedia.org/wiki/Expected_value) function (e.g. the mean value).

****

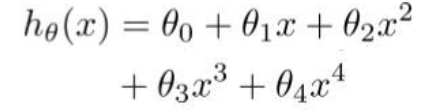
**Variance:**

The variability of model prediction for a given data point which tells us spread of our data is called the variance of the model. The model with high variance has a very complex fit to the training data and thus is not able to fit accurately on the data which it hasn’t seen before. As a result, such models perform very well on training data but has high error rates on test data.  
When a model is high on variance, it is then said to as **Overfitting of Data.**

**Model with high variance pays a lot of attention to training data and fail to generalize on the unseen data. As a result, such models perform very well on training data but has high prediction error on unseen/test data.**

**Low Variance is desired: High Variance means OverFitting of the model on training data***.*

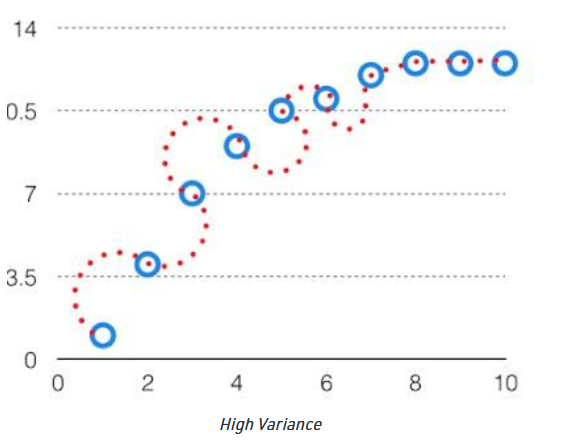
In such a problem, a hypothesis looks like:

**

It can expressed mathematically as:

Var (f̂ (x)) =E [f̂ (x)²] – E [f̂ (x)]²

Where E[⋅] is the [expected value](https://en.wikipedia.org/wiki/Expected_value) function (e.g. the mean value).



**Bias-Variance Trade-Off**

**Ultimate goel of any machine learning algorithm is to build a prediction model with Low Bias and Low Variance** that is measure of reduced error and correct prediction power of any model.

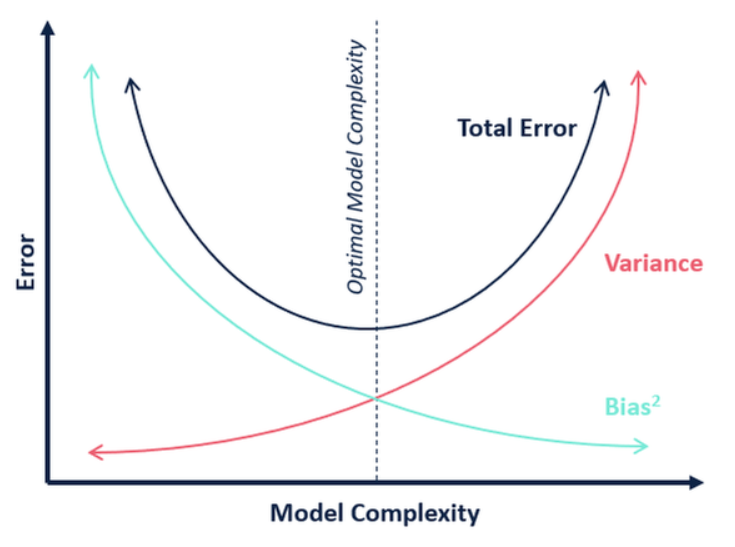
**Finding the right balance between the bias and variance of the model is called the Bias-Variance trade-off.**

***There is inverse relationship between bias and variance in machine learning.***

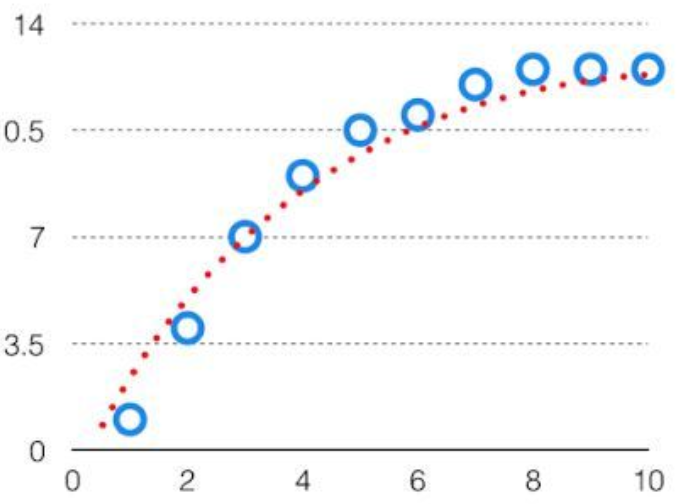
* I*ncreasing the bias will decrease the variance.*
* *Increasing the variance will decrease the bias*

**High Model Complexity: High Variance, Low Bias**

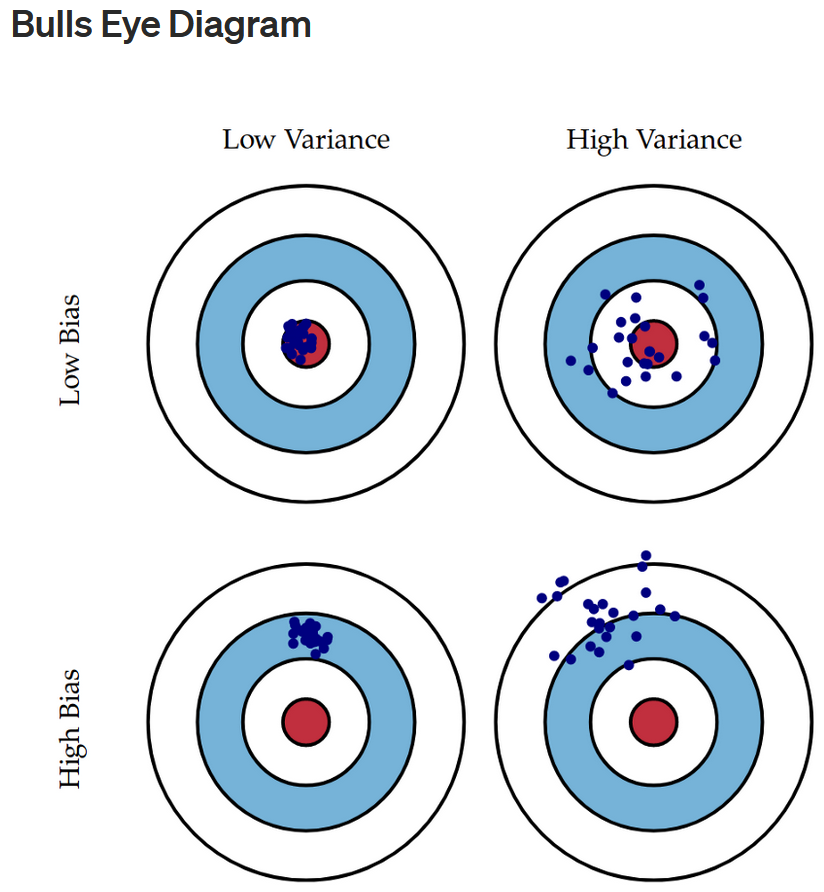
**Low Model Complexity: High Bias, Low Variance**



Tradeoff in complexity is why there is a tradeoff between bias and variance. An algorithm can’t be more complex and less complex at the same time. For the graph, the perfect tradeoff will be like.



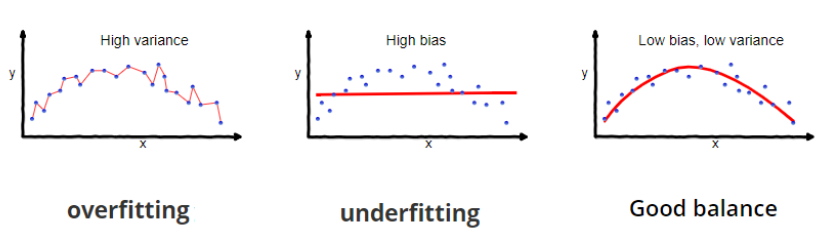
If the model is too simple and has very few parameters, it will suffer from high bias and low variance. On the other hand, if the model has many parameters, it will have high variance and low bias. This trade-off should result in a perfectly balanced relationship between the two.



# How to find the Right Balance?

Ideally, low bias and low variance is the target for any Machine Learning model.

Finding the right balance with Low Bias and Low Variance is an iterative process where model is trained with different combination of features, Hyperparameters, different set of data set for training and test to find the right combination. We stop when we reach the point where Low Bias and Low variance is achieved, and model is neither underfit and nor overfit i.e. Prediction accuracy is same on train and test data.



**Lowering high Bias or Underfitting:**

1. Use non Parameterised Algorithms

2. Make model more complex with more features

3. Use Non Linear Algorithms Example( Polynomial Regression, Kernel Function in SVM

**Lowering high Variance or Overfitting:**

1. Use More Data for training to make model learn maximum hidden pattern from the training data and model becomes generalised.

2. Use Regularization Techniques Example: L1 , L2, Drop Out, Early Stopping( in case of Neural Networks)etc.

3. Hyper Parameter Tuning to avoid Overfitting Example: Higher value of K in KNN, Tuning of C and Gama for SVM, Depth of Tree in Decision Tree

4. Use less number of features — Manual or Feature Selection Algorithms or automated using L1, L2 Regularization

5. Reduce complexity of Model — Reduce polynomial degree in case of Polynomial regression and Logistic regression

6. Use Advance techniques like Cross Validation, Stratified Cross Validation etc.