To approximate reality, learning algorithm use mathematical or statistical models whose “**error**” can be split into two main components: **reducible** and **irreducible** error. Irreducible error or inherent uncertainty is associated with a natural variability in a system. On the other hand, reducible error, as the name suggests, can be and should be minimized further to maximize accuracy.

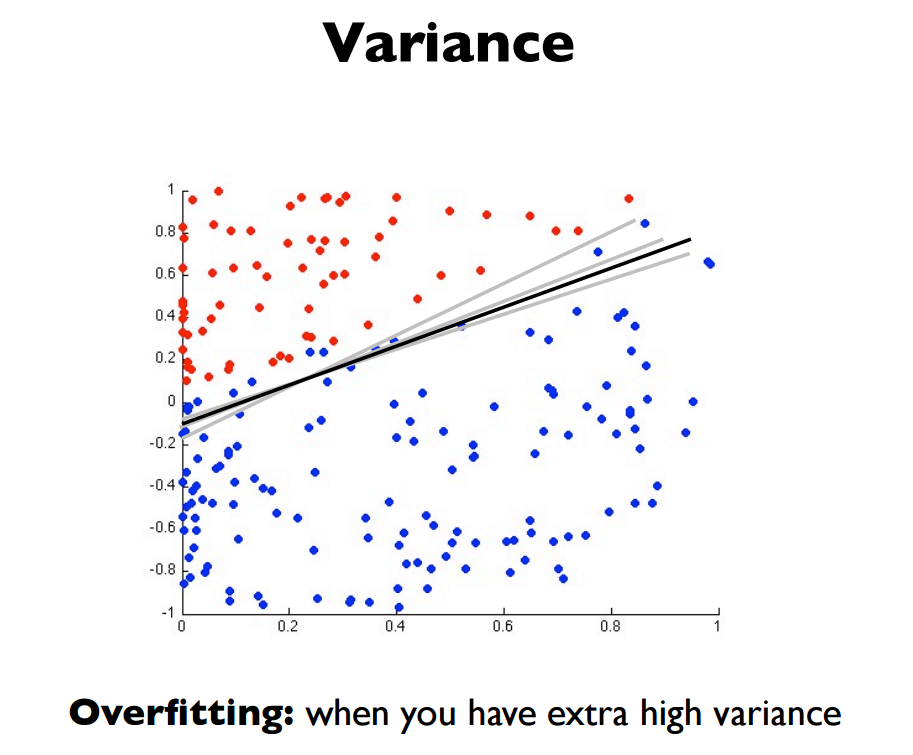
Reducible error can be further decomposed into “**error due to squared bias**” and “**error due to variance**.” The data scientist’s goal is to simultaneously reduce bias and variance as much as possible in order to obtain as accurate model as is feasible. However, there is a tradeoff to be made when selecting models of different flexibility or complexity and in selecting appropriate training sets to minimize these sources of error!

The **error due to squared bias** is the amount by which the expected model prediction differs from the true value or target, over the training data.

The **error due to variance** is the amount by which the prediction, over one training set, differs from the expected predicted value, over all the training sets. As with bias, you can repeat the entire model building process multiple times. To paraphrase Manning et al (2008), variance measures how inconsistent are the predictions from one another, over different training sets, not whether they are accurate or not.

Models that exhibit small variance and high bias underfit the truth target. Models that exhibit high variance and low bias overfit the truth target. Note that if your target truth is highly nonlinear, and you select a linear model to approximate it, then you’re introducing a bias resulting from the linear model’s inability to capture nonlinearity. In fact, your linear model is underfitting the nonlinear target function over the training set. Likewise, if your target truth is linear, and you select a nonlinear model to approximate it, then you’re introducing a bias resulting from the nonlinear model’s inability to be linear where it needs to be. In fact, the nonlinear model is overfitting the linear target function over the training set.

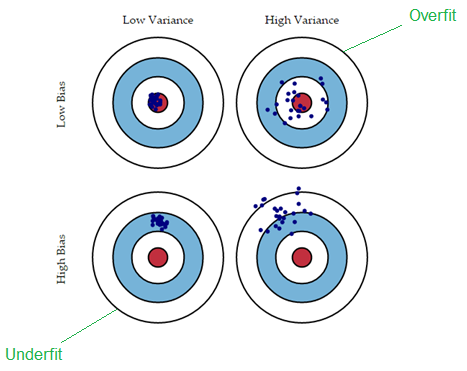
The “tradeoff” between bias and variance can be viewed in this manner – a learning algorithm with low bias must be “flexible” so that it can fit the data well. But if the learning algorithm is too flexible (for instance, too linear), it will fit each training data set differently, and hence have high variance. A key characteristic of many supervised learning methods is a built-in way to control the bias-variance tradeoff either automatically or by providing a special parameter that the data scientist can adjust.

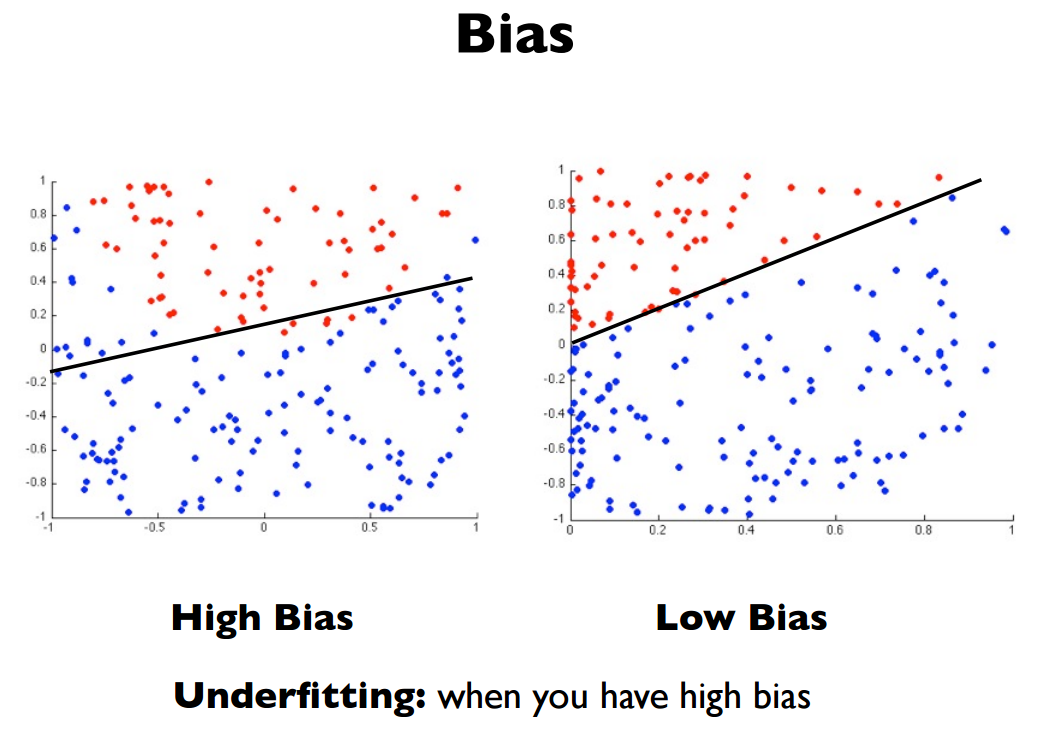
When working to characterize the bias-variance tradeoff, you need to develop metrics for determining the accuracy of your model. There are two common metrics used in machine learning: training error and test error. As an example, for linear regression models you can calculate the Mean Square Error (MSE) for the different data sets – training set to train the model (60-80% of the available data), and test set to check the accuracy of the model (40-20% of the available data). For completeness, there is an additional validation step after training. So a typical split is 50% training, 25% validation, and 25% test. The training set is for model fitting. The validation set is for estimating the prediction error so you can choose the appropriate model. And the test set is used to assess the model (and its error) once the model is chosen. 

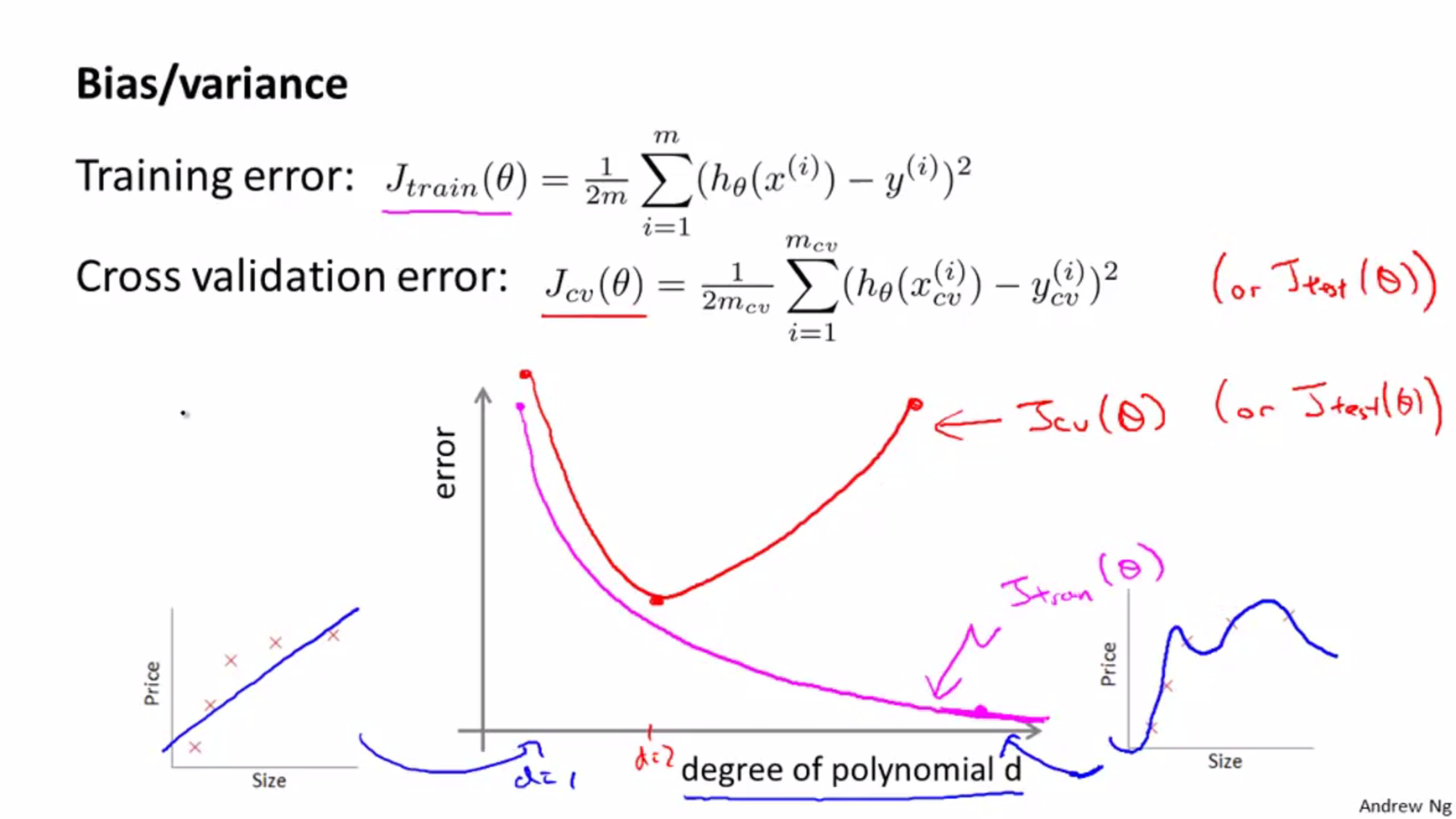
Models that are too complex tend to have high variance and low bias, while models that are too simple will tend to have high bias and low variance. The best model will have both low bias and low variance.

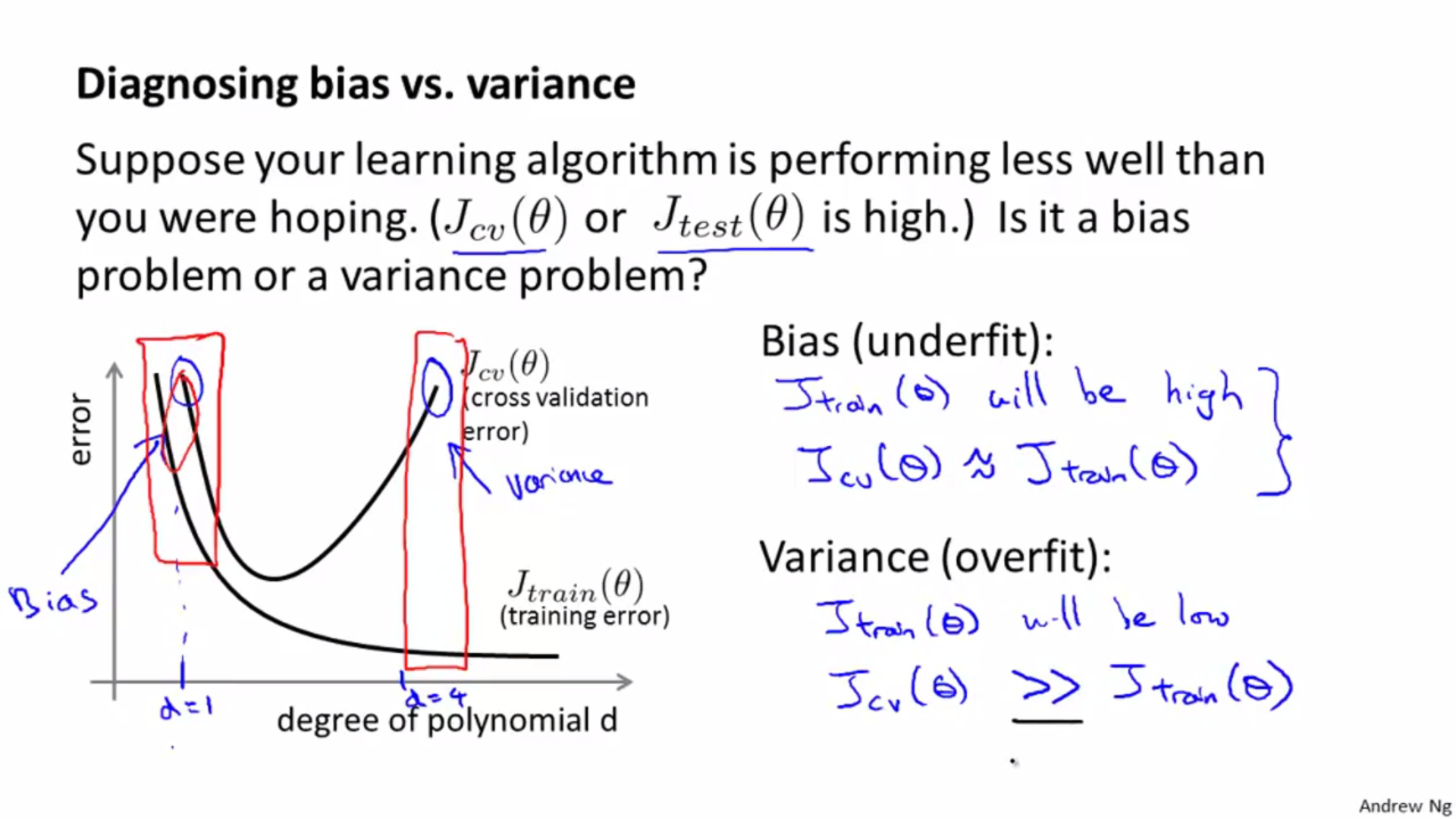
We can create a graphical visualization of bias and variance using a bulls-eye diagram. Imagine that the center of the target is a model that perfectly predicts the correct values. As we move away from the bulls-eye, our predictions get worse and worse. Imagine we can repeat our entire model building process to get a number of separate hits on the target. Each hit represents an individual realization of our model, given the chance variability in the training data we gather. Sometimes we will get a good distribution of training data so we predict very well and we are close to the bulls-eye, while sometimes our training data might be full of outliers or non-standard values resulting in poorer predictions. These different realizations result in a scatter of hits on the target.

We can plot four different cases representing combinations of both high and low bias and variance









Referrence: <http://insidebigdata.com/2014/10/22/ask-data-scientist-bias-vs-variance-tradeoff/>