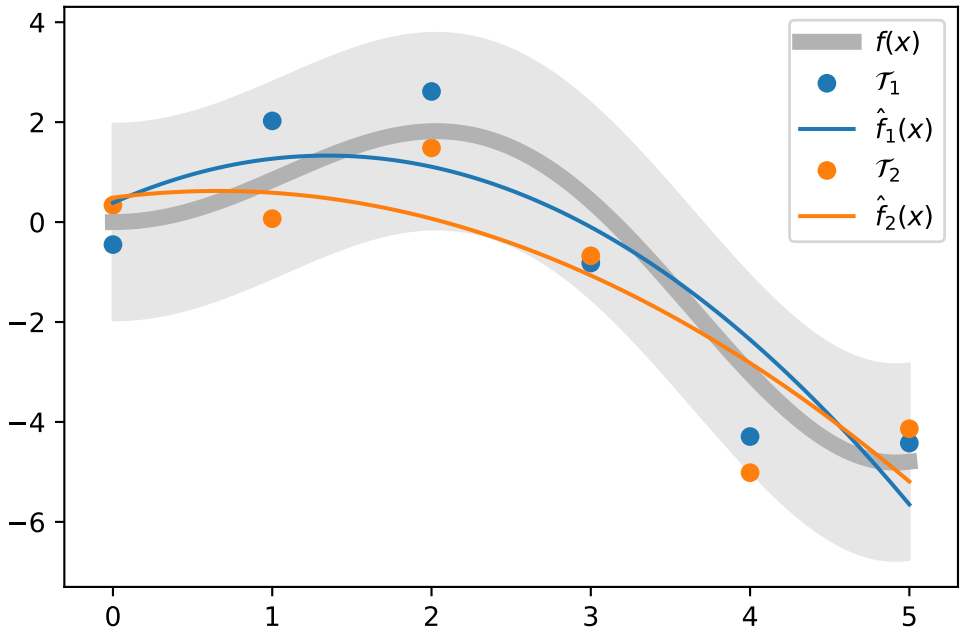
The Bias-Variance Decomposition

Posted on April 4, 2018 | 6 minutes | Chad Scherrer

Say there's some experiment that generates noisy data. You and I each go through the process independently, and model the results. Would the resulting models be exactly the same?

Well no, of course not. That's the whole problem with noise. Instead, we'll usually end up with something like this (for a quadratic fit):



The idea is that we'd like to find an approximation to $f(x)$, but we can never observe this function directly. Instead, we observe some *training set* $\mathcal{T}$ and use that to arrive at an approximation $\hat{f}_{\mathcal{T}}(x)$.

In general, the approximation won't be a perfect fit; there are two sources of error, which we'll soon connect.

- *Systematic error*, or *bias*, comes from the choice of model. In the current example, it's impossible for any quadratic function to exactly match the curve we're looking for.
- *Random error*, or *variance*, comes from randomness inherent in the training set. In the graph above, our two experiments gave different observations, leading to different model fits.

By the end of this post, we'll have a way to express the average error (over all training sets) in terms of these two sources.

Mean Squared Error

How should we measure "average error"? In regression, the most common approach (at least classically) is in terms of *squared loss*. After we use a training set $\mathcal{T}$ to find an approximation $\hat{f}_{\mathcal{T}}$, we hope that for a given x,

$$(\hat{f}_{\mathcal{T}}(x) - f(x))^2$$

is small. To quantify this, we can average over lots of possible training sets, which leads to the *mean squared error*,

$$\text{MSE}(x) = \mathbb{E}_{\mathcal{T}}[(\hat{f}_{\mathcal{T}}(x) - f(x))^2] \ .$$

The $\mathbb{E}_{\mathcal{T}}$ notation just means we're taking an average (or *expected value*) over different possibilities of the training set $\mathcal{T}$.

In order to decompose the MSE, it's helpful to think in terms of the average prediction (again, over all training sets). Let's call this $\mu(x)$. So we can define

$$\mu(x) = \mathbb{E}_{\mathcal{T}}[\hat{f}_{\mathcal{T}}(x)] \ .$$

Bias

Informally, we think of *bias* as an opinion that skews our interpretation of observations. The current context is no different, just more precise. As with people, **we can think of a biased model as one with a strong opinion**.

For example, in the graph above, the prediction function will have the shape of a parabola, no matter what training set is used. This shape is predetermined by the choice of model, and no data can possibly change its mind. In fact, even if we average over an infinite number of training sets, the result still won't be perfect.

To make this precise, let's define $\text{Bias}(x)$ as

$$\begin{aligned}\text{Bias}(x) &= \mathbb{E}_{\mathcal{T}}[\hat{f}_{\mathcal{T}}(x) - f(x)] \\ &= \mu(x) - f(x) \ .\end{aligned}$$

We get the second line by distributing the expectation over the two terms; we can do this because expectations are linear.

Also, note that we're abusing the notation to keep it a bit simpler. To be really rigorous we should call it something like

$$\text{Bias}_{\mathcal{T}}[\hat{f}_{\mathcal{T}}(x), f(x)] \ .$$

This is too heavy to be useful, so we'll continue to write it as above, as a function of x.

Variance

Variance in this context is no different than the usual statistical concept. We just ask, "What's the average squared distance of these things from their average?".

For the current discussion, "these things" are the $\hat{f}_{\mathcal{T}}(x)$ values, and "their average" is $\mu(x)$. This leads us to

$$\text{Var}(x) = \mathbb{E}_{\mathcal{T}}[(\hat{f}_{\mathcal{T}}(x) - \mu(x))^2] \ .$$

The shorthand notation $\text{Var}(x)$ is a bit unusual (x is not a random variable or a distribution), but we'll stick with it to keep it parallel with the way we're writing the bias.

Note that just as with random variables, we can also write the variance as

$$\text{Var}(x) = \mathbb{E}_{\mathcal{T}}[\hat{f}_{\mathcal{T}}(x)^2] - \mu(x)^2 \ .$$

Decomposing the MSE

With all of the preliminaries out of the way, the decomposition becomes really easy. We can just expand the MSE, rearrange the terms and complete the square.

$$\begin{aligned}
\text{MSE}(x) &= \mathbb{E}_{\mathcal{T}}[(\hat{f}_{\mathcal{T}}(x) - f(x))^2] \\
&= \mathbb{E}_{\mathcal{T}}[\hat{f}_{\mathcal{T}}(x)^2 - 2\hat{f}_{\mathcal{T}}(x)f(x) + f(x)^2] \\
&= \mathbb{E}_{\mathcal{T}}[\hat{f}_{\mathcal{T}}(x)^2] - 2\mu(x)f(x) + f(x)^2 \\
&= \mathbb{E}_{\mathcal{T}}[\hat{f}_{\mathcal{T}}(x)^2] - \mu(x)^2 + \mu(x)^2 - 2\mu(x)f(x) + f(x)^2 \\
&= \mathbb{E}_{\mathcal{T}}[\hat{f}_{\mathcal{T}}(x)^2] - \mu(x)^2 + (\mu(x) - f(x))^2 \\
&= \text{Var}(x) + \text{Bias}(x)^2
\end{aligned}$$

Example

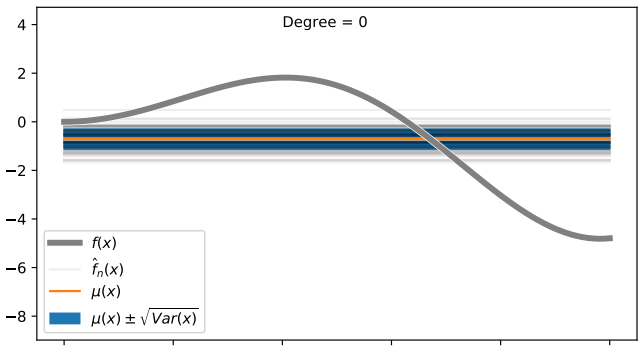
Let's consider our original example in a bit more detail. Starting with

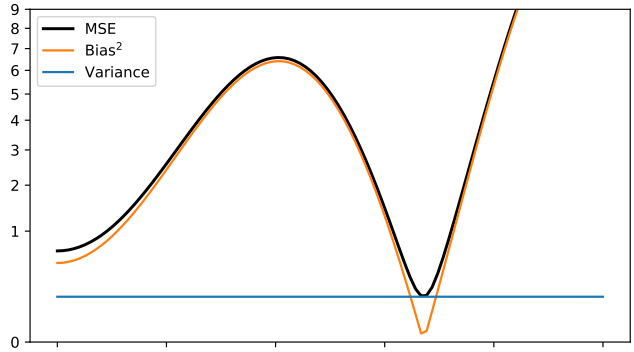
$$f(x) = x \sin x \ ,$$

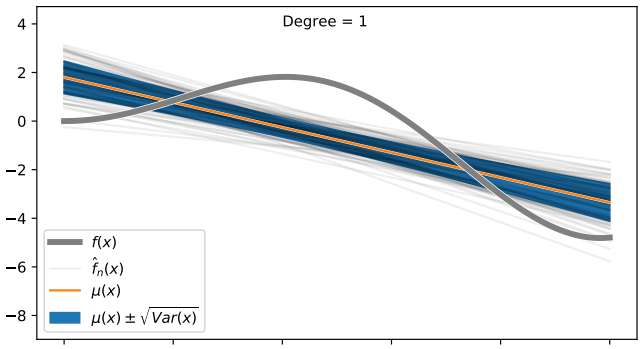
each training set consists of six (x_n, y_n) pairs, with

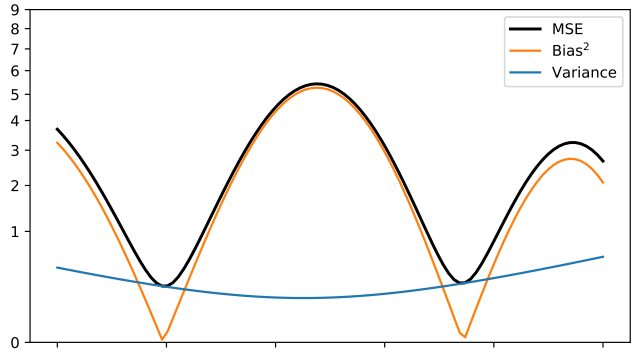
$$\begin{aligned}
x_n &= n + 1 \\
y_n &\sim \text{Normal}(f(x_n), 1) \ .
\end{aligned}$$

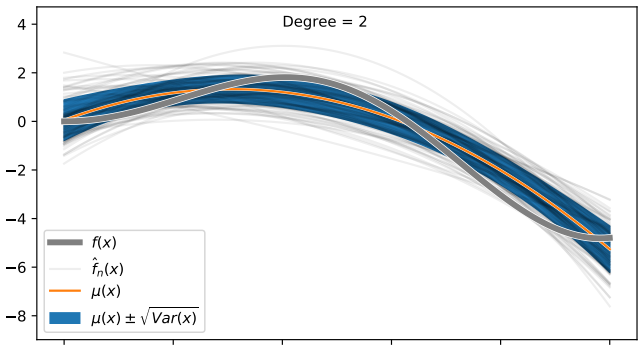
As we fit increasingly complex models, we can compare the bias, variance, and MSE. Note that to make the scale visually reasonable, the second column of graphs has a square-root scale for the y-axis. Because of this, the MSE, bias and variance are visusally related to the RMSE (*root* mean squared error), absolute bias, and standard deviation.

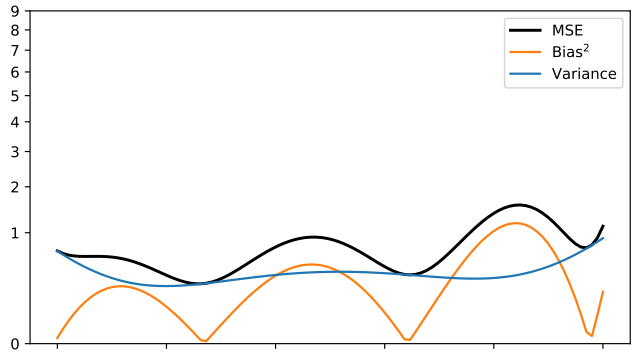

Degree = 0
f(x)
$\hat{f}_n(x)$
μ(x)
μ(x) ± √Var(x)

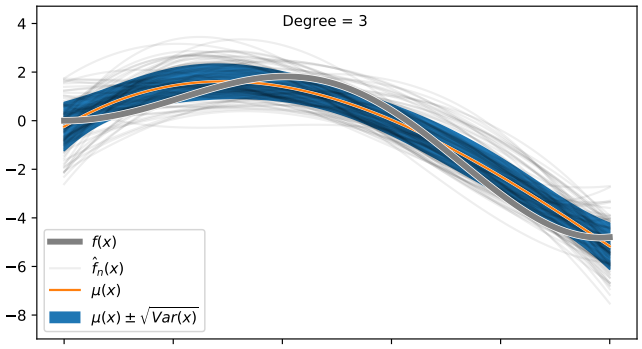

MSE
Bias²
Variance

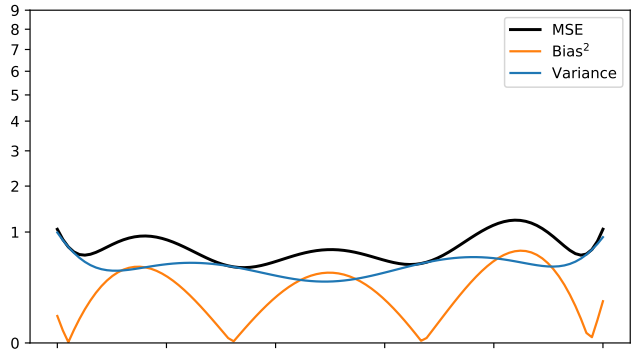

Degree = 1
f(x)
$\hat{f}_n(x)$
μ(x)
μ(x) ± √Var(x)

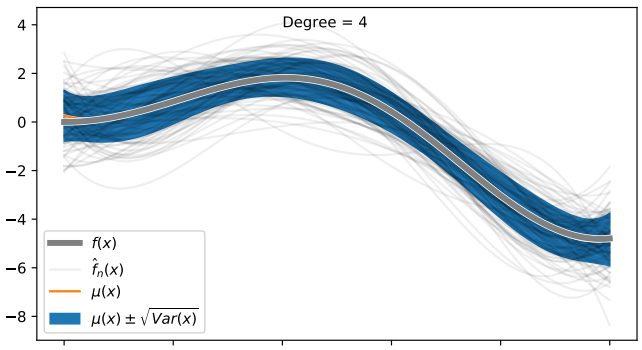

MSE
Bias²
Variance

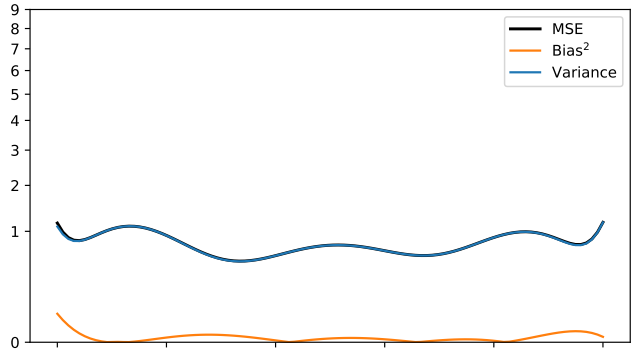

Degree = 2
f(x)
$\hat{f}_n(x)$
μ(x)
μ(x) ± √Var(x)


MSE
Bias²
Variance


Degree = 3
f(x)
$\hat{f}_n(x)$
μ(x)
μ(x) ± √Var(x)


MSE
Bias²
Variance


Degree = 4
f(x)
$\hat{f}_n(x)$
μ(x)
μ(x) ± √Var(x)


MSE
Bias²
Variance

As model complexity increases, more of the MSE can be attributed to variance. For complex models, this motivates the introduction of regularization, in which we artificially increase the bias in order to reduce variance.

Irreducible Error

Though we're mostly interested in approximating $f(x)$, most applied problems don't give us access to this directly. Instead, we train on a set $\mathcal{T}_0$, and compute the loss on a test set $\mathcal{T}_1$ from the same distribution.

Similarly to our approach to this point, we can average this over all possible train/test sets, to arrive at the *expected prediction error*. For squared loss this has the form

$$\mathrm{EPE}(x) = \mathbb{E}_{\mathcal{T}_0,\mathcal{T}_1}[(y - \hat{f}_{\mathcal{T}_0}(x))^2] \ ,$$

where (x, y) are taken from the test set $\mathcal{T}_1$.

Continuing as before, we can write the EPE as

$$\mathrm{EPE}(x) = \mathrm{Bias}(x)^2 + \mathrm{Var}(x) + \sigma^2 \ .$$

The new term σ^2 is called the *irreducible error*, and is the same as the empirical MSE of the test set.

Other Loss Functions

Though all terms to this point are defined in the context of squared loss, it's common to hear them used more generally. This use is usually less rigorous, and only by analogy. But it doesn't have to be. For some more detail on this, check out this paper:

Domingos, Pedro. "A unified bias-variance decomposition." Proceedings of 17th International Conference on Machine Learning. 2000. (https://homes.cs.washington.edu/~pedrod/papers/mlc00a.pdf)

Conclusion

An understanding of the bias-variance decomposition, and the corresponding tradeoff in modeling, are crucial for any data scientist to understand. This post has become much longer than I had originally planned, and I hope the discussion

has been useful to you. Thanks for your time!

← PREVIOUS POST (HTTPS://CSCHERRER.GITHUB.IO/POST/BAYESIAN-CHANGEPOINT/)

NEXT POST → (HTTPS://CSCHERRER.GITHUB.IO/POST/MAX-PROFIT/)


(mailto:chad.scherrer@gmail.com)

(https://plus.google.com/+chad.scherrer)

(https://github.com/cscherrer)

(https://twitter.com/@ChadScherrer)

(https://reddit.com/u/cscherrer)

(https://linkedin.com/in/chadscherrer)

(https://stackoverflow.com/users/488124/chad-scherrer)

()