**Why is MSE = Bias² + Variance?**

**Introduction to “good” statistical estimators and their properties**

**The bias-variance tradeoff**

*“The bias-variance tradeoff”* is a popular concept you’ll encounter in the context of [ML/AI](http://bit.ly/quaesita_emperor). In building up to making it intuitive, I figured I’d give the formula-lovers among you a chatty explanation of where this key equation comes from:

MSE = Bias² + Variance

Well, this article isn’t only about proving this formula — that’s just a [mean](http://bit.ly/quaesita_gistlist) (heh) to an end. I’m using it as an excuse to give you a behind-the-scenes look into how and why [statisticians](http://bit.ly/quaesita_pointofstats) manipulate some core building blocks and how we think about what makes some estimators better than others, but be warned: it’s about to get technical around here.

Forays into formulas and generalized nitty gritty are out of character for my blog, so many readers might like to take this opportunity to rush for the exit. If the idea of a proof fills you with existential dread, here’s a [fun article](http://bit.ly/quaesita_noeqns) for you to enjoy instead. Never fear, you’ll still be able to follow the upcoming bias-variance tradeoff article, but you’ll have to take it on faith that this formula is accurate. This article is for those who demand proof! (And a discussion about festooned Greek letters.)

**Prerequisites**

Still here? Nice. This stuff will go down smoother if you’re somewhat familiar with a few core concepts, so here’s a quick checklist:

**Bias; Distribution; Estimand; Estimate; Estimator; Expected value E(X); Loss function; Mean; Model; Observation; Parameter; Population; Probability; Random variable; Sample; Statistic; Variance V(X)**

If you’re missing a concept, I’ve got you covered in my [statistical glossary](http://bit.ly/quaesita_gistlist).

**E(X) and V(X)**

To make sure you’re comfy with manipulating the building blocks for our discussion, let’s grab an excerpt out of my [field guide to a distribution’s parameters](http://bit.ly/quaesita_lemur):

**Expected value E(*X*)**

An expected value, written as *E(X)* or *E(X = x)*, is the theoretical probability-weighted **mean** (this word is pronounced “average”) of the [random variable *X*](http://bit.ly/quaesita_distributions)*.*

You find it by weighting (multiplying) each potential value [*x* that *X* can take](http://bit.ly/quaesita_distributions) by its corresponding probability [*P(X = x)*](http://bit.ly/quaesita_distributions)and then combining them (with an integral ∫ for [continuous](http://bit.ly/quaesita_datatypes) variables like height or a sum for [discrete](http://bit.ly/quaesita_datatypes) variables like height-rounded-to-the-nearest-inch): ***E(X) =* ∑ *x P(X = x)***

If we’re dealing with a fair six-sided die, *X* can take each value in {1, 2, 3, 4, 5, 6} with equal probability 1/6, so:

E(*X*) = (1)(1/6) + (2)(1/6) + (3)(1/6) + (4)(1/6) + (5)(1/6) + (6)(1/6) = 3.5

In other words, 3.5 is the probability-weighted average for *X* and nobody cares that 3.5 isn’t even an allowable outcome of the die roll.

**Variance V(X)**

Replacing *X* with *(X - E(X))²* in the E(*X*) formula above gives you the variance of a [distribution](http://bit.ly/quaesita_distributions). Let me empower you to calculate it whenever the urge strikes you:

V(*X*) = E[*(X - E(X))²*] = **∑[*x -* E(X)]² *P(X = x)***

That’s a definition, so there’s no proof for this part. Let’s take it for a spin to get the variance for a fair die: V(X) = **∑[*x -* E(X)]² *P(X=x)*** = **∑(*x - 3.5)*² *P(X=x) =*** (1–3.5)² (1/6) + (2–3.5)² (1/6) + (3–3.5)² (1/6) + (4–3.5)² (1/6) + (5–3.5)² (1/6) + (6–3.5)² (1/6) = 2.916666…

If you’re dealing with continuous data, you’ll use an integral instead of a sum, but it’s the same idea.

**Alternative V(X) formula**

In our proof below, we’re going to use a little switcheroo with that variance formula, replacing the middle bit with the rightmost bit:

V(*X*) = E[*(X - E(X))²*] = E[(*X* )²] *-* [E(*X*)]²

I owe you an explanation of where it comes from, so let’s cover that quickly:

V(*X*) = E[*(X - E(X))²*]   
= E[*X*² *-* 2 *X* E(*X*) + E(*X*)²]   
= E(*X*²) *-* 2 E(*X*) E(*X*) + [E(*X*)]²  
= E[(*X* )²] *-* [E(*X*)]²

How and why did this happen? The key bit is going from line 2 to line 3… the reason we can do this with the brackets is that expected values are sums/integrals, so whatever we’re allowed to do with constants and brackets for sums and integrals we’re also allowed to do with expected values. That’s why if **a** and **b** are constants, then E[**a***X* + **b**] = **a**E(*X*) + **b**. Oh, and E(X) itself is also a constant — it’s not random after it’s calculated — so E(E(*X*)) = E(*X*). Glad that’s sorted.

**Notes on notation and bling**

[**Estimands**](http://bit.ly/quaesita_vocab) (the things you want to [*estimate*](http://bit.ly/quaesita_vocab)) are often indicated with unadorned Greek letters, most often θ. (This is the letter “theta” which we’d have in English if we felt that “th” deserved its own letter; “th” is close enough to “pffft” to make θ a truly excellent choice for the standard placeholder in statistics.)

Estimands θ are [parameters](http://bit.ly/quaesita_vocab), so they’re (unknown) constants: E(θ) = θ and V(θ) = 0.

[**Estimators**](http://bit.ly/quaesita_vocab) (the formulas you’re using in order to [*estimate*](http://bit.ly/quaesita_vocab) the [*estimand*](http://bit.ly/quaesita_vocab)) are often indicated by putting bling on Greek letters, such as a little hat on θ, like so:

https://miro.medium.com/max/642/1*kbb6HlbZNRvaNfXQ_k8H1A.png

Since it’s a pain to get this blog post to render a θ with a hat nicely in a Medium post, I’ll ask you to use your imagination and see this neat little guy whenever I type “θhat”. Also, you’re going through this with pen-and-paper anyways — you’re not trying to study formulas just by reading, like some kind of maniac, right? — so you won’t get confused by my notation. You’ll copy down the formulas formatted with the pretty hat above and then read your own notes, glancing at my chatty explanations to help if you get lost.

**Deriving the MSE formula**

Estimators are [random variables](http://bit.ly/quaesita_distributions) until you plug your [data](http://bit.ly/quaesita_hist) in to get an **estimate** (“best guess”). An estimate is a constant, so you’ll treat it as a plain ol’ number. Again, so we don’t get confused:

* **Estimand**, θ, the thing we’re trying to estimate, a constant.
* **Estimator**, θhat, the formula we’re using to get the estimate, a random variable that depends on the data you get. Luck of the draw!
* **Estimate**, some number that comes out at the end once we plug data into the estimator.

Now in order to know if our estimator θhat is dumb as bricks, we’re going to want to check if we can ***expect*** it to be close to the estimand θ. So E() of the random variable X = (θhat - θ) is the first one we’ll be playing with.

*E(X) = E((θhat - θ)) = E(θhat ) - E(θ) = E(θhat) - E(θ) = E(θhat) - θ*

This quantity has a special name in statistics: bias.

An unbiased estimator is one where E(θhat) = θ, which is an excellent property. It means we can ***expect*** our estimator to be on the money (on average). In my [gentle intro blog post](http://bit.ly/quaesita_bivar), I explained that bias refers to *“results that are systematically off the mark.”* I should more properly have said that bias is the expected distance between the results our estimator (θhat) gives us and the thing we’re aiming at (θ), in other words:

**Bias = E(θhat) - θ**

**Picking the “best” estimator**

If you like unbiased estimators, then you’ll love you some UMVUEs. This acronym stands for uniformly minimum-variance unbiased estimator and what it refers to is a criterion for a best choice among unbiased estimators: if they’re all unbiased, pick the one with the lowest variance! (And now I’ve brought you to approximately [chapter 7 of a master’s level statistical inference textbook](https://mybiostats.files.wordpress.com/2015/03/casella-berger.pdf). You’re welcome.)

The fancy term for “you offered me two estimators with the same bias, so I chose the one with the smaller variance, duh” is ***efficiency***.

Of course, there are many different ways to pick a “best” estimator. Nice properties to look for include unbiasedness, relative efficiency, consistency, asymptotic unbiasedness, and asymptotic efficiency. The first two are ***small sample properties*** and the last three are ***large sample properties*** since they deal with how the estimator behaves as you increase the sample size. An estimator is ***consistent*** if it’s eventually on target as the sample size grows. (That’s right, it’s time for limits! Read [this](https://www.unil.ch/files/live/sites/issr/files/shared/8._Telechargement/Cours_MA_quantiBasel/Chapter_7_Point_Estimation.pdf) if *your* time -> infinity.)

Efficiency is a pretty solid property to care about, since no one wants their estimator to be all over the place. (Gross.) Since efficiency is about variance, let’s try plugging X = (θhat — θ) into our variance formula:

*Variance* V(X) = E[(X)²] - [E(X)]²  
*becomes* V(θhat -θ) = E[(θhat - θ)²] - [E(θhat - θ)]²

Variance measures the spread of a random variable, so subtracting a constant (you can treat the parameter θ as a constant) merely shifts everything over without changing spread, V(θhat - θ) = V(θhat), so:

V(θhat) = E[(θhat - θ)²] - [E(θhat) - E(θ)]²

Now we rearrange terms and remember that E(θ) = θ for constants:

E[(θhat - θ)²] = [E(θhat) - θ]² + V(θhat)

Now let’s take a look at this formula, because it has some special things with special names in it. Hint: remember bias?

Bias = E(θhat) — θ

Can we find that in our formula? Sure can!

E[(θhat - θ)²] = [Bias]² + V(θhat) = Bias² + Variance

So what the hell is the thing on the left? It’s a useful quantity, but we weren’t very creative in naming it. Since “error” is a decent way to describe the difference (often notated as ε) between where our shot landed (θhat) and where we were aiming (θ), E[(θhat - θ)²] = E(ε²).

E(ε²) is named, wait for it, mean squared error! That’s MSE for short. Yes, it’s literally named E(ε²): we take the mean (another word for expected value) of squared errors ε². Bonus points for creativity there, statisticians.

MSE is the most popular (and vanilla) choice for a model’s [loss function](http://bit.ly/quaesita_msefav) and it tends to be the first one you’re taught (here it is in my own [machine learning course](http://bit.ly/mfml_006)).

And so we have:

MSE = Bias² + Variance

Now that you’ve worked through the math, you’re ready to understand what the bias-variance tradeoff in machine learning is all about. We’ll cover that in my next article — stay tuned by hitting that *follow* button.

**[Stats Gist List: An irreverent statistician’s guide to jargon](https://towardsdatascience.com/stats-gist-list-an-irreverent-statisticians-guide-to-jargon-be8173df090d" \t "_blank)**

**[Plain-language bandaids to fix gaps in your statistics knowledge](https://towardsdatascience.com/stats-gist-list-an-irreverent-statisticians-guide-to-jargon-be8173df090d" \t "_blank)**

[towardsdatascience.com](https://towardsdatascience.com/stats-gist-list-an-irreverent-statisticians-guide-to-jargon-be8173df090d" \t "_blank)

**Thanks for reading! How about a YouTube course?**

If you had fun here and you’re looking for an entire applied AI course designed to be fun for beginners and experts alike, here’s the one I made for your amusement: