Dangers of designing for an average user:

Two theories - a thought experiment – simulation in numpy – gaussian correlation say whaaaa – extension to n dimensions – One size never fits all – Indian mom’s, the best optimizers

I recently came across two theories in probability, one from a friend and another from a brilliant podcast of 99pi. I try to explain these two theories using an idea borrowed from Allen Downey, the author of the books in the Think X series. The premise of the Think X series which I highly recommend is that If you know how to program, you can use that skill to learn other topics. These two theories share a common ground and hence a single post to cover them both. Unlike most articles that start with a story and end with the implementation, here I start with a concrete implementation, building a strong foundation to commit to the memory and then end with the history.

Let’s begin with a thought experiment. You are a lead designer for a new line of winter cloths for a leading retailer. You have the data of body dimensions from existing and probable customers taken at several branches across the country. The body dimensions include height, weight, hip size, knee width etc. Your task is to come up with the optimal standard design that will fit most people. After verifying that all the body dimensions roughly follow normal distribution, you conclude that the best way would be to use the average value in all these dimensions to arrive at your optimal design, after all average minimizes total error.

Let’s denote,

S1 – Set of all the people within +- K standard deviation away from the mean height,

S2 – Set of all the people within +- K standard deviation away from the mean weight

.

.

Sn. - Set of all the people within +- K standard deviation away from the mean of entity n. You get the idea.

The above sets represent people who are very close to average that they can get away wearing the dress designed for the average person without much ruckus. The root of the fallacy lies in the fact that, as you increase dimensions the argument falls apart. Even though you maximize the people that can wear your design by creating it for the average user, when you look across multiple dimensions the number of people who are very close to the average, rapidly approaches zero. Average User fallacy though intriguing, looks naïve at hindsight when considering the curse of dimensionality.

Talk is cheap. Show me the Code. – Linus Torvalds.

While theories are fascinating, they have little retain value. To drive home the point, I will try Downey’s method of trying to understand it through code.

Theories are like stories that are reductionisms of actual complex facts. In the above example, one key thing that the Average User Fallacy doesn’t address is the correlation between the dimensions. For example, there is a strong correlation between height and weight. Taller the person, the heavier he is considering all things equal. Similarly, taller the person, longer the knee length.

Let’s assume that the average height is 175 cm and average weight is 190 lbs.

Assuming they are correlated with the correlation factor of 0.3, lets try to simulate data from the gaussian distribution. In order to simulate multivariate correlated data, we do the following,

Correlated Data = Mean + Covariance \* Uncorrelated Gaussian Data.

We need covariance for generating correlated data from uncorrelated data. Covariance is the square root of the correlation matrix. We can use Cholesky’s decomposition to arrive at it, although the method is not exact.

Making sure that the Cholesky’s decomposition works, by recovering correlation matrix from the covariance matrix by taking square.

<Correlation of both the variables.>

<Histogram and scatter plots.>

Weight and Height have different mean but same standard deviation. Loop through the data set and find points within K=1 STD from mean.

normal\_height is a binary vector indicating all values within K standard deviation from mean height and the same for normal\_weight. Normal\_height\_weight is the intersection of both these conditions.

This provides a nice Segway into our second theory Gaussian Correlation Inequality. Before switching topics remember that we are trying to show how increasing the dimensions reduces the people who are closer to the average.

<insert image>

The image looks confusing at first glance, and the name is a mouthful. Let’s try to break it down. Let’s start with the basic rule of multiplication,

P(A intersection B) = P(A) \* P(B), given the events are independent.

But in our case, we know that A and B are not independent, and they are correlated given by the correlation matrix.

In this case,

P(A intersection B) = P(A) \* P(B|A).

Gaussian correlation inequality is given by,

P(A intersect B) >= P(A) \* P(B)

Replacing A and B with S1 and S2 in our scenario, we know that P(S1) and P(S2) ~ 0.68, because in a gaussian random variable 68% of the data falls within +-1 STD of the bell curve mean.

Plotting P(S1 intersect S2) and P(S1) \* P(S2) against various correlation values clearly shows that when there is no correlation they are identical but when there is a correlation, P(S1 intersect S2) steadily starts to increase and P(S1) \* P(S2) remains fairly constant.

<plot of CF vs probability)

Gaussian correlation inequality proved that,

<Equation.>

Now circling back to the average user fallacy, we know that P(S1 intersect S2) is < P(S1), which translates to Probability of a user falling +- K STD from the average, decreases as you increase the dimension in spite of the correlation among dimensions.

Let’s simulate correlated data in n dimensions and see the effects of curse of dimensionality on Average User fallacy. We will randomize correlation factor between dimensions.

<correlation matrix code>

<graph ndimension>

From the above it is clear that irrespective of correlation, as dimension increases Probability of occurring close to the average decreases rapidly.

You can find the entire code in this jupyter notebook. <add link)

One size never fits all:

For the tolerant readers who have stayed so far, let’s look at a real-world example of designing for the average user going horribly wrong.

99 pi -

During the World War II, the US Air Force expanded rapidly accompanied with a decline in performance and a rash of deaths even during training. The high death rate in the Air Force was a mystery for many years, but after blaming the pilots and their training programs, the military finally realized that the cockpit itself was to blame, that it didn’t fit most pilots. At first, they assumed it was just too small and that the average man had grown since the 1920s, so in 1950, they asked researchers at [Wright Air Force](https://en.wikipedia.org/wiki/Wright-Patterson_Air_Force_Base) base in Ohio to calculate the new average.

One of these researches was a young Harvard graduate named Gilbert S. Daniels. In his research measuring thousands of airmen on a set of ten critical physical dimensions, Daniels realized that none of the pilots he measured was average on all ten dimensions. Not a single one. When he looked at just three dimensions, less than five percent were average. Daniels realized that by designing something for an average pilot, it was literally designed to fit nobody.

Revisiting the thought experiment that we had at the start, being a lead designer and now knowing the Average User Fallacy, how should you go about optimizing your design? In an ideal world, not constrained by resources, the solution would be to custom fit every person with their own set of measurements. But the real world is never that simple. You are always bound by constraints and need to strike a balance. On the one end, you have witnessed the horrors of designing for the average and on the other hand custom fitting everyone is unattainable.

Indian Mom’s, The Best optimizers:

Average user fallacy is not ubiquitous. For example, if you are designing a chair, a car seat, a modern cockpit, you are not restricted to fixed dimensions. You use levers and other mechanisms to allow multitudes of body dimensions to access them comfortably. These dynamic systems let users navigate it seamlessly though its continuous design space. But there are certain designs that are inherently subjected to average user fallacy. These systems have certain rigidity within them that render them incapable of dynamic adjustments. One perfect example is the dress. Once bought, the size remains fixed. So the dress manufacturers based on their motive to maximize the profit tries to reach as many people as possible while still reducing the different varieties in sizes that they have to produce, leading to the system of making clothes in distinct discrete intervals marked by S, M , L , XL etc. Worst case, the dress start to shrink, and you start to gain weight testing the integrity of your dress. This seems like an impossible problem to solve, but not for an Indian Mom. How she solves this conundrum is quite ingenious. She single handedly decides that you will get one size above your current size so as to support her adage, “You never buy for your current self, but for the one that you will grow into”. She proudly then strolls past you, perfectly knowing that she got you what you wanted, while optimizing to increase the longevity of the product thereby decreasing the long-term cost. If this isn’t a perfect optimization, I don’t know what is.