Statistics In AI

There are certain terms we use in order to classify data.

Population - all possible data
Sample - a subset of the population

Usually, the population is theoretic, meaning that its existence is possible only in the sense that at one time it is true but practically unobtainable.

We typically aggregate data into a metric. This means that we take one or more records into a value to group or sort. How do we report on metrics?

- **Parameter**: a metric for some population
- **Statistic**: a metric for some sample

These distinctions are important for determining what data means. The statistic infers the parameter, meaning that the metric for a sample can generalize metrics in the population. It can be said that the parameter is theoretic since the population is theoretic.

Mean

In statistics, we denote:

$$Sample\ Mean = \bar{x}$$

Let x be some random variable. The top bar can be over any variable that represents a collection of data. More explicitly:

$$\bar{x} = \frac{\sum_i x_i}{n}$$

Where n = the size of x.
Consider:

$$\bar{x} \rightarrow sample\ mean$$

$$\mu = E(x) \rightarrow\ population\ mean$$

$E(x)$ is often referred to as the expected value.

For the population, it is essentially saying:

$$\mu = E(x) = \sum_{x \in X} x \cdot Pr(x)$$

Median

Also consider the median: m. We get the median by following these steps:

- Sort x
- Return $x[\frac{n}{2}]$

The median is some random variable is a robust statistic, meaning it is not affected by extreme data points.

It would make logical sense that is there are no outliers in a data set:

$$\bar{x} \approx m$$

Why do we square numbers (x^2)?

- Removes negative numbers

Why not use absolute values ($|-2|$)?

- Consider $\frac{d}{dx}[|x|]$. I don't remember it, do you? In computer science, we love $\frac{d}{dx}[x^2]$. We can find the min/max. Simply keeping it at positive x^2 lets us find derivatives easier.

An Algorithmic Approach

Construct an algorithm that will compute the average distance for each point form the mean.

For this algorithm $f(x)$:

- Compute $\bar{x}$
- For each entry, computer $x_i - \bar{x}$, add to total
- Divide: total/n

From this approach, we can take away the following:

- As long as we computer $\bar{x}$, we can place n-1 points anywhere. We then place the final point at a specific location to preserve $\bar{x}$. We call this the **degrees of freedom**.

Degrees of Freedom(DF)

The degrees of freedom represents that number of points that can be anything.

The *error* of our model is calculated as:

$$Error = x_i - \bar{x}$$

Where x_i is some data point and $\bar{x}$ is the model prediction.

A data set's total error is calculated as:

$$Total\ Error = Sum\ of\ Error = \sum_i (x_i - \bar{x})$$

For a total error of zero, you can subtract every point by the mean it it would come out to be zero. We want to remove negative values to understand how much error there is.

$$Sum\ of\ squared\ error = \sum_i (x_i - \bar{x})^2$$

This forms the basic idea behind neural networks.

To find the average error:

$$\frac{\sum_i (x_i - \bar{x})^2}{n-1}$$

Notice how we removed a degree of freedom to account for the mean metric. We call this an *unbiased* error.

This, sample variance, s^2 is

$$s^2 = \frac{\sum_i (x_i - \bar{x})^2}{n-1}$$

Standard Deviation

$$s = \sqrt{s^2}$$

$$\bar{x} \to \mu = E(x)$$

$$s^2 \to \sigma^2 = Var(x) = E(x - \mu)$$

$$s \to \sigma = \sqrt{Var(x)}$$