A short digression into priors

How should we choose our priors? Isn't this biasing the inference?

We will **set** priors using statistical probability distributions (in R type `?Distributions`). These distributions cover a range of widely understood generative processes and include:

- Gaussian/Normal distribution
- Binomial
- Uniform
- Student's T
- Poisson
- InvGamma
- Weibull

And many many others.

In R, these distributions are included by default, and can be used typically with the following prefixes:

`d`, for "density", `r` for "random generation" (ie. draw one number from the distribution at random), `p` for the distribution, `q` for quantile. To do a simulation of the grades of 100 students at random assuming the grades are normally distributed with mean 75 and SD 10, we could say

`grades = rnorm(100, 75, 10)` and get a vector of grades, again randomly distributed according to the normal we specified. (Hmm, new grading idea!)

One thing to note: although libraries like Seaborn allow you to fit distributions to your data, in a principled Bayesian workflow we would try to choose the prior based on our domain understanding.

In other words, taking the data and fitting a distribution implies we think the data perfectly determines the distribution. This may not be (and often isn't) the case.

One of the bigger challenges for non-statisticians like us is understanding the underlying assumptions and applicability of probability distributions. In a lot of cases the Normal, Poisson, logNormal distributions (in addition to Uniform) give us most of what we need for this course.

- The Poisson is good for simulating discrete event arrivals, like bugs in code or cars at a stoplight.
- The logNormal simulates the possibility that there is a fat tail (skewness) in the data, which is often the case in software data (especially for networks). Of course we could take the log of the measured data that is logNormally distributed and then model the result using the simple Normal distribution.
- The Uniform is an equal probability distribution across all outcomes, and often models the situation when we don't know any better.

Dealing with Zeros

A further consideration is *zero-inflation* which means distributions are padded with more zero values. This reflects the scenario in which a lot of results are empty, e.g., if you are looking at defect data and some group of files have no bugs reported.

Again, base your choice on the **domain process as you see it**. And then **compare the models** using info criterion like AIC to see which model is best suited.