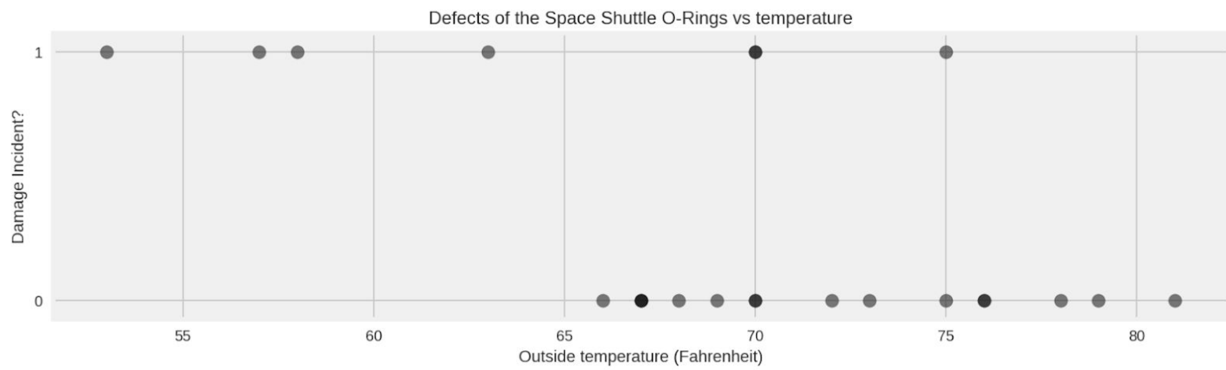
Bayesian inference real life example

Challenger example:-



$$p(t) = \frac{1}{1+e^{\beta t+\alpha}},$$

$$\beta = Normal\ (\mu, \sigma^2)$$

$$\alpha = Normal\ (\mu, \sigma^2)$$

```
def challenger_joint_log_prob(D, temperature_, alpha, beta):
       rv_alpha = tfd.Normal(loc=0., scale=1000.)
       rv_beta = tfd.Normal(loc=0., scale=1000.)
       logistic_p = 1.0/(1. + tf.exp(beta * tf.to_float(temperature_) + alpha))
       rv_observed = tfd.Bernoulli(probs=logistic_p)

       return (rv_alpha.log_prob(alpha)+
       rv_beta.log_prob(beta)+tf.reduce_sum(rv_observed.log_prob(D)))
```

Also, note that *rv_alpha* and *rv_beta* represent the random variables for our prior distributions for α and β. By contrast, `rv_observed`represents the conditional distribution for the likelihood of observations in temperature and O-ring outcome, given a logistic distribution parameterized by α and β. tf.reduce_sum(rv_observed.log_prob(D)) represents the likelihood.

Next, we take our `joint_log_prob` function, and send it to the [`tfp.mcmc`] module. Markov chain Monte Carlo (MCMC) algorithms make educated guesses about the unknown input values, computing the likelihood of the set of arguments in the `joint_log_prob` function. By repeating this process many times, MCMC builds a distribution of likely parameters. Constructing this distribution is the goal of probabilistic inference.