Method

Bayes factor gives relative prob. of failure vs success.
Predict Failure when B > 1, predict Success otherwise

$$B = \frac{p(F|\vec{x})}{p(S|\vec{x})} = \frac{p(\vec{x}|F)}{p(\vec{x}|S)} \frac{p(F)}{p(S)}$$

Prior ratio: measured from first 1 million events. Most events succeed, so data needs to give strong evidence to overcome the prior. Updated after every batch of 1 million events. “Knowing nothing about the data, how common are S/F?”

$$\frac{p(F)}{p(S)} \simeq \frac{0.7}{99.3} = 0.007$$

Likelihood ratio: what the data tells us about event in question. “How common are these parameters x among failures/successes?”

$$\mathcal{L} = \frac{p(\vec{x}|F)}{p(\vec{x}|S)} = \frac{N_F(\vec{x})/N_F}{N_S(\vec{x})/N_S}$$

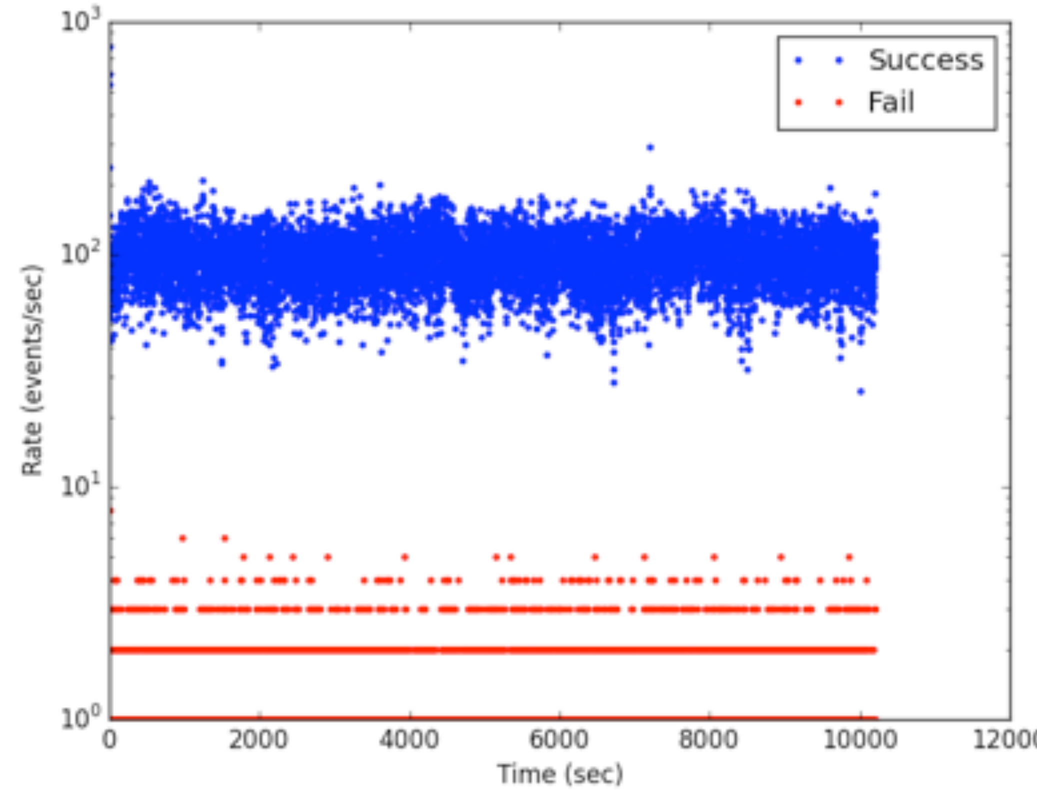
Tally $N_F(x)$, $N_S(x)$ for each value of x in each chunk. Estimate likelihood ratio from fraction of F/S for this key x versus all F/S (for any key) in current and previous chunks. Update tally after each event.

Results and Discussion

Time does not seem to correlate to outcome, at least in first chunk

Could imagine maintenance periods causing a correlation among time, computer and result

Likelihood has an implicit time dependence, could imagine giving more weight to recent outcomes in likelihood computation



Prediction

		Success	Fail
Actual Result	Success	98.75%	0.03%
	Fail	0.09%	1.13%

99.97% of Successes correctly classified
93% of Failures correctly classified

If a new value of x is encountered, merely predict via prior. A more informed guess would require modeling relations between x_i with machine learning and/or human knowledge of network