Contents

Model an adaptation experiment

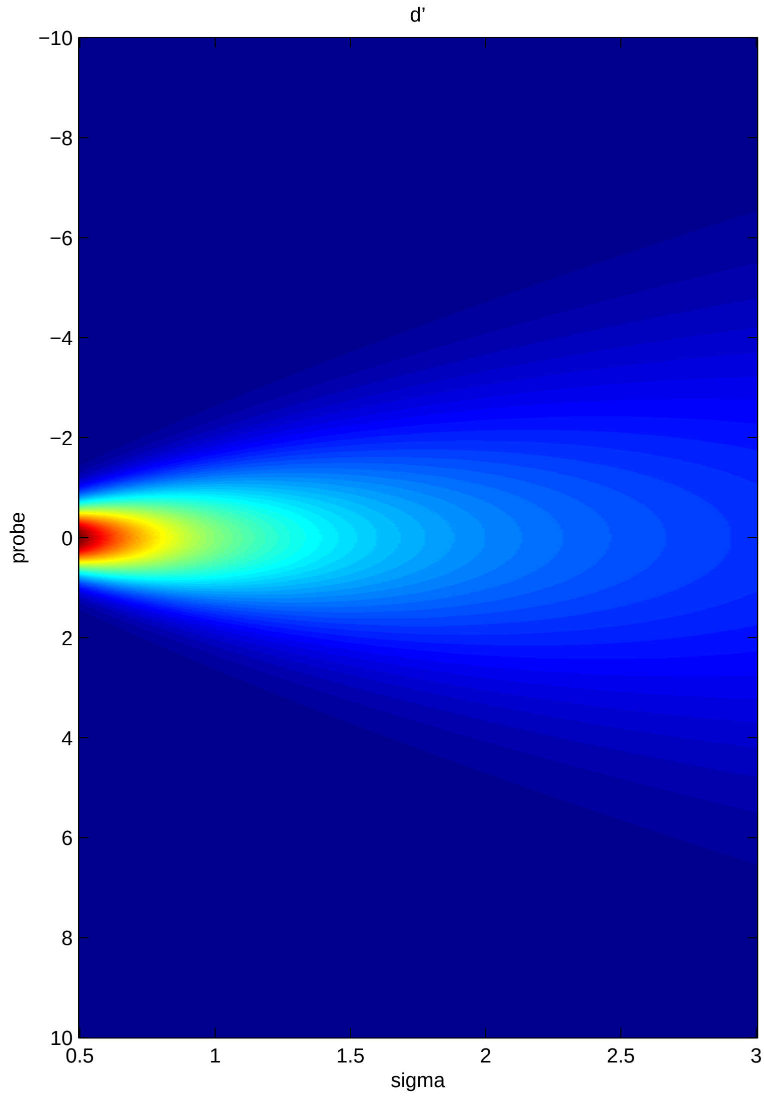
Imagine an experiment where the goal of the observer is to discrimate two stimuli. These stimuli are embedded in a stream of distractors. The variance of the distractors are modulated in time. This is a psychophysical version of the experiment in Fairhall et al. 2001.

Let's assume that the observer adapts to optimally represent the distribution of the distractors (from an information-theoretic perspective). Whether we're thinking of one neuron or several neurons representing the input, the amount of "mental juice" given to one stimulus should be proportional to p(s), and d' is proportional to 1/p(s).

This is easiest to see when thinking of a single "neuron" with equal variance for each of its possible outputs. In that case, the optimal tuning function should be proportional to the cdf of p(s). Then a stimulus s_1 is represented by $cdf(s_1)$, and a stimulus $s_1 + \delta$ is given by $cdf(s_1 + \delta) \approx cdf(s_1) + \delta p(s_1)$. Since these two outputs will be corrupted by additive gaussian noise $N(0, \sigma^2)$, d' will be given by $\delta p(s_1)/\sigma$. Thus $d'(s) \propto p(s)$. See also Ganguli & Simoncelli (2010) for a version of this argument with population codes.

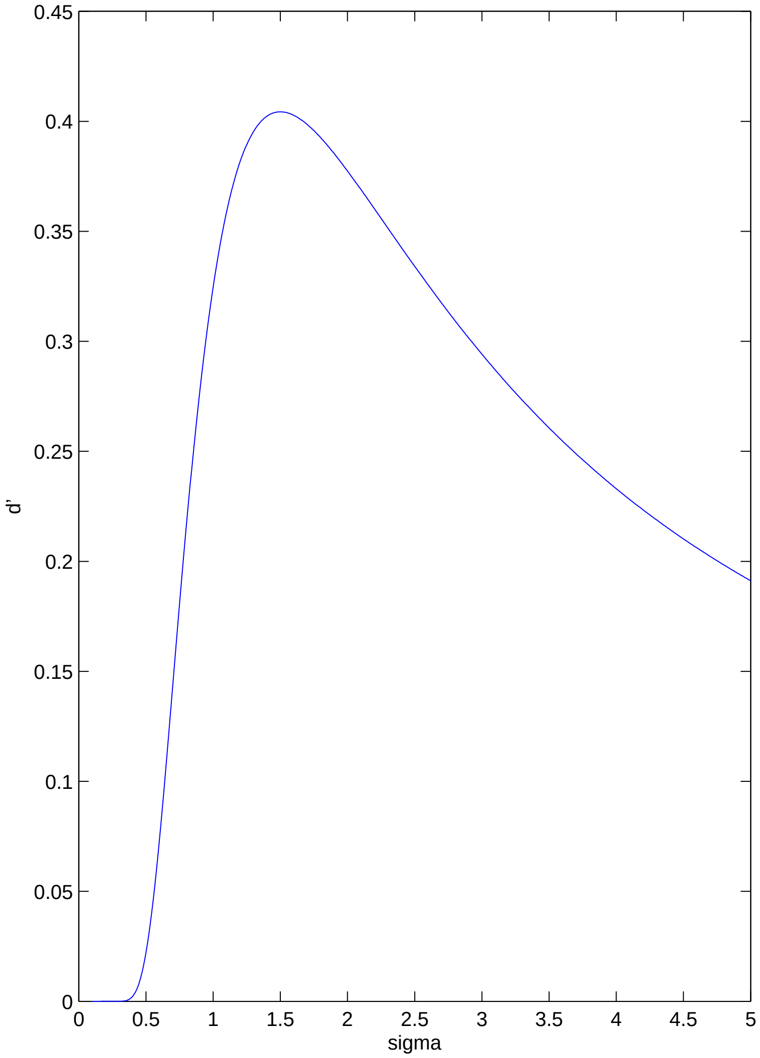
Let's plot d' as a function of the stimulus for different normal distributions with increasing sigma:

```
clf;
sigmas = .5:.01:3;
xs     = (-10:.01:10)';
dprime = bsxfun(@times,1./sigmas,exp(-.5*bsxfun(@times,xs.^2,1./sigmas.^2)));

imagesc(sigmas,xs,dprime);
xlabel('sigma');
ylabel('probe');
title('d''');
```



For a fixed probe value, d' can both increase and decrease as a function of the value of σ. Here's an example:

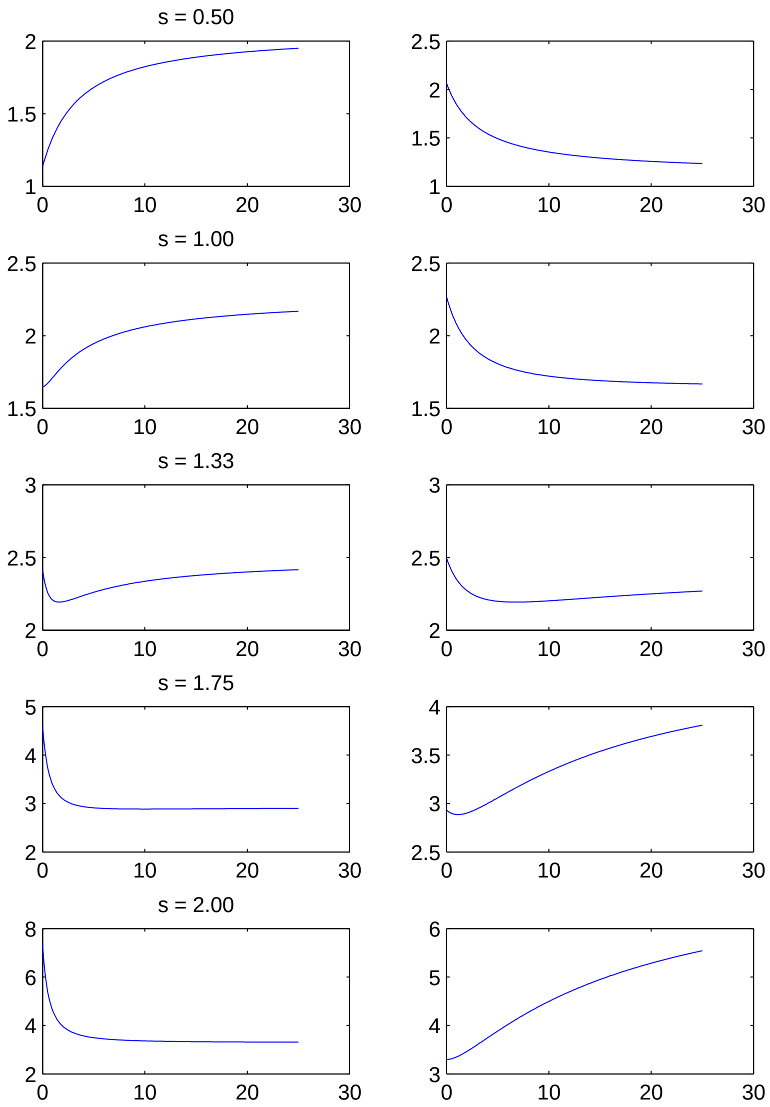
```
sigmas = .1:.01:5;
xs     = 1.5';
dprime = bsxfun(@times,1./sigmas,exp(-.5*bsxfun(@times,xs.^2,1./sigmas.^2)));

clf;
plot(sigmas,dprime);
xlabel('sigma');
ylabel('d''');
```



In the hypothetical experiment above, let's say that sigma jumps from 1 to 2. That means that sigma_hat will sweep slowly from 1 to 2. Thus, using a probe stimulus s = 1.5 presented at time t after the jump, d'(t) will vary from its initial value first increasing, then decreasing. Let's see this using the 1/(1+x) behaviour of the sigma_hat estimate for different values of s. LHS: increase from 1 -> 2 RHS: decrease from 2 -> 1

```
sigma1 = 1;
sigma2 = 2;

t = 0:.1:25;
alpha = .3;
```

```
ss = [.5,1,1.33,1.75,2];

for ii = 1:length(ss)
    s = ss(ii);
    subplot(length(ss),2,ii*2-1);
    sigmat = (sigma1-sigma2)./(1+alpha*t) + sigma2;
    dprime = sigmat.*exp(.5*s^2./sigmat.^2);
    plot(t,dprime);
    title(sprintf('s = %.2f',s))

    subplot(length(ss),2,ii*2);
    sigmat = -(sigma1-sigma2)./(1+alpha*t) + sigma1;
    dprime = sigmat.*exp(.5*s^2./sigmat.^2);

    plot(t,dprime);
end
```



Analysis

Remarkably, d' generally stabilises more slowly for decreases than increases, and can be nonmonotonic as a function of time for a given probe.