# Hebbian learning for deciding optimally among many alternatives (almost)

**Patrick Simen (psimen@math.princeton.edu)**
Princeton Neuroscience Institute, Princeton University, Green Hall, Washington Rd., Princeton, NJ 08544

**Tyler McMillen (tmcmillen@fullerton.edu)**
Department of Mathematics, California State University at Fullerton, Fullerton, CA 92834

**Sam Behseta (sbehseta@fullerton.edu)**
Department of Mathematics, California State University at Fullerton, Fullerton, CA 92834

## Abstract

Reward-maximizing performance and neurally plausible mechanisms for achieving it have been completely characterized for a general class of two-alternative decision making tasks, and data suggest that humans can implement the optimal procedure. A greater number of alternatives complicates the analysis, but here too, analytical approximations to optimality that are physically and psychologically plausible have been analyzed. All of these analyses, however, leave critical open questions, two of which are the following: 1) How are near-optimal model parameterizations learned from experience? 2) How can sensory neurons' broad tuning curves be incorporated into the aforementioned optimal performance theory, which assumes decisions are based only on the most informative neurons? We present a possible answer to all of these questions in the form of an extremely simple, reward-modulated Hebbian learning rule for weight updates in a neural network that learns to approximate the multi-hypothesis sequential probability ratio test.

**Keywords:** Hebbian learning; diffusion model; neural network; multi-hypothesis sequential test; sequential probability ratio test; speed-accuracy tradeoff; response time

## Introduction

We examine the problem of maximizing earnings from a sequence of $N$-alternative decisions about the identity of noisy stimuli, with $N > 2$. Our goal is to parameterize a simple neural circuit model whose behavior approximates optimal performance in such tasks, while simultaneously accounting for the fundamental role of tuning curves in the neural representation of sensory stimuli. Throughout, we take 'optimal' to mean *reward maximizing*, and we assume that correct decisions earn rewards for the decider.

As we show, simple principles of neural computation are sufficient to approximate this form of optimality quite closely in a class of $N$-choice tasks involving response-terminated stimuli: that is, stimuli that provide information continuously until the time (the response time) at which participants decide for themselves when to stop observing and make a response. This is somewhat surprising, given that a general decision policy that guarantees truly optimal performance cannot even be explicitly formulated for such tasks, as we discuss below.

### $N$-choice, response-terminated decision tasks

We assume that participants earn rewards for correct responses, and earn less for errors (for simplicity, we assume errors earn nothing). In the simple tasks we consider, each stimulus type has a fixed prior probability within a block of trials, and the average signal-to-noise ratio of each stimulus is fixed. The duration, rather than the number of trials, is also held fixed, and the distribution of response-to-stimulus intervals (RSIs) that delay the onset of the next stimulus after a response is stationary. In this case, maximizing the rate of reward also maximizes the total reward.

Maximizing gains in this and a variety of similar tasks requires probabilistic inference. While the importance of a principled inference process is widely understood in psychology and neuroscience, the complexity of optimal decision policies in tasks with response-terminated stimuli (also known as 'free response' or 'response time' tasks) and $N > 2$ choices appears to be less well appreciated.

For 2-choice tasks of the type just described, reward-maximizing performance has been completely characterized (Bogacz et al., 2006): a sequential probability ratio test (SPRT) should be carried out in which the current likelihood ratio of the two hypotheses is multiplied by the probability of a given data sample under one hypothesis and divided by the probability of that data sample under the other hypothesis (equivalently, the logs of these probabilities can be added and subtracted, respectively — from now on, we will cast our discussion in terms of log-likelihoods). A response should be made when the resulting log-likelihood exceeds a fixed threshold (Wald & Wolfowitz, 1948). There exists an optimal starting point of the log-likelihood ratio (e.g., 0, for equally likely stimuli) and an optimal separation between the two response thresholds (one greater and one less than zero) that depends on the signal-to-noise ratio (SNR) and the RSI (Bogacz et al., 2006). Gold and Shadlen (2001) have demonstrated that for systems consisting of a neuron/anti-neuron pair, each of which is tuned for one of the two stimulus types in a 2-choice task, the log-likelihood ratio is approximately proportional simply to the difference between the activations of the two neurons, suggesting an extremely simple neural implementation of the SPRT.

In contrast, if the number of choices is greater than 2, the optimal policy for deciding based on accumulated information is nontrivial. In particular, a natural (but definitely suboptimal) approach to $N$-choice decision making is to compute the posterior probability of each of the $N$ hypotheses, and then select whichever one first exceeds a fixed threshold. In fact, the best decision is made when the entire set of posterior probabilities meets conditions that are nontrivial func-

tions of the posterior values. A thought experiment may help make clear why this is true. Consider the case of a 3-choice task in which one choice has attained an 80% posterior probability of being correct, while the other posteriors are 10% and 10%. A fixed 80% threshold will therefore not distinguish this case from a case in which the posteriors are 80%, 19% and 1%. These two cases are quite different, however, and dealing optimally with them requires taking account of all posterior probabilities in a more sophisticated way. Because of this, and because of the inherent difficulty in defining truly optimal decision policies to apply to the posteriors, multi-hypothesis sequential probability ratio tests (MSPRTs) were designed to approximate optimality with a decision policy consisting of fixed thresholds applied to posteriors or likelihood ratios (Dragalin, Tartakovsky, & Veeravalli, 1999).

### Tuning curves

Tuning curves are ubiquitous in neural responses to stimuli (Butts & Goldman, 2006). The relationship between tuning curve shape and decision making performance has intrigued researchers for several years (e.g., Pouget, Deneve, Ducom, & Latham, 1999). Naively, one may suppose that task participants improve their performance by sharpening the tuning curves of the neurons involved. However, wider tuning curves are in some cases more efficient in conveying information, and the most informative tuning curve shape depends strongly on the noise and correlations (Zhang & Sejnowski, 1999; Seriés, Latham, & Pouget, 2004). Moreover, in several tasks, participants may improve performance without significantly altering the shapes of the tuning curves in the neurons involved. For instance, in an angle discrimination task, monkeys are able to learn to discriminate between finer angles over time, while the tuning curves of primary sensory neurons are altered very little (Ghose, Yang, & Maunsell, 2002; Law & Gold, 2008). This suggests that improvements in performance take place in a learning process downstream from the receptor neurons.

In this paper we explore the ways in which a subject may improve performance in decision tasks, given tuning curve shapes in receptor neurons. We do not consider the alteration of receptor units' tuning curves, but rather how the information in tuning curves can be utilized more efficiently over the course of many trials.

## The leaky competing accumulator model for decision making

We propose a three layer neural model for decision making (defined in Table 1, and depicted in Fig. 1). The first layer acts simply as a sensory amplifier; the next layer integrates the information from the first layer, but also exhibits competitive dynamics that gradually build a commitment to one course of action over the alternatives; the last layer triggers a discrete motor response when commitment to one response is sufficiently strong. For convenience, we refer to these three layers, respectively, as the MT, LIP and SC layers. These labels reflect the fact that our model exhibits known properties of neurons in the monkey middle temporal area (MT), the lateral intraparietal area (LIP) and the superior colliculus (SC) in decision making tasks requiring eye movements in response to visual motion stimuli (i.e., random dot kinematograms; Shadlen & Newsome, 2001). The architecture of this circuitry is expected to apply without major modification to other stimulus and response types, however.

Table 1: Three layer model with weight learning rule.

| | |
|---|---|
| $S_i,\ i = 1, \ldots, n$ | input signals (MT) |
| $dx_i = \left( -k x_i - m \sum_{j \neq i} x_j + S_i \right) dt + \ldots$ $c\, dB_i$ | accumulators (LIP) |
| $z_i = H(y_i - \Theta), \quad y_i = \sum_{j=1}^{n} w_{ij} x_j$ | decision units (SC) |
| $w_{ij}^{\text{new}} = (1-\alpha) w_{ij}^{\text{old}} + \alpha \Delta w_{ij}$ $\Delta w_{ij} = r z_i x_j$ | LIP to SC weight-learning rule |

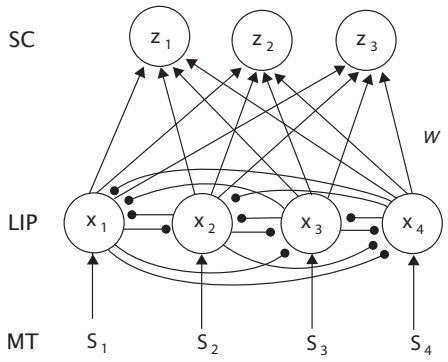


Figure 1: Neural network model with 4 accumulators and 3 alternatives. The weight matrix $W$ denotes the weights of the connections between the $x_i$'s and $z_j$'s. Arrows represent excitatory connections; circles represent inhibitory connections.

We suppose that MT neurons have tuning curves that are preferentially sensitive to a single, given direction of visual motion, and that another layer is stimulated by the activity in this input layer. By virtue of their excitatory connections to LIP, model MT units' tuning curves and their feedforward connections to LIP in turn define tuning curves for LIP units. Questions of major importance in computational neuroscience are: Through what sort of learning process do these tuning curves arise? Can we define an optimal connection scheme that maximizes some function, such as the rate of reward earned by the model? We attempt to make progress on these questions while making the simplifying assumption that the brain circuits in question are approximately

linear systems (at least over a limited range of inputs), and that they employ simple learning schemes (such as Hebbian learning, or error-updating rules such as the Widrow-Hoff, Rescorla-Wagner or delta rule). Recent work (e.g. McMillen & Holmes, 2006; Bogacz & Gurney, 2007) that avoids discussion of tuning curves and learning shows that these assumptions allow simple neural network models to map precisely onto one or another form of MSPRT. We now demonstrate that a similar model that learns connections strengths and accounts for tuning curves does remarkably well at approaching optimal (reward maximizing) performance in decision making tasks with multiple alternatives. The model's layers are represented mathematically by $S$, $x$, and $z$.

Upon presentation of a stimulus, the model's MT layer presents a vector of signals to accumulators in the LIP layer. The signal presented to the $i$th unit in the LIP layer is referred to as $S_i$, representing the total weighted sum of MT signals to the $i$th accumulator. Each stimulus corresponds to a unique signal, so that the set of signals to the LIP layer may be represented as a vector indexed by $\mu$:

$$S^{\mu} = \left(S_1^{\mu}, S_2^{\mu}, \ldots, S_n^{\mu}\right).$$

The task is to determine which of $N$ possible signal vectors this represents. Notice that the number of vectors can be different from the number of signals, i.e. in general $n > N$.

Although it is not required, we will generally take the $S^{\mu}$ signals to be Gaussian:

$$S_i^{\mu} = a \exp\left[-\frac{(i - dir_{\mu})^2}{2\phi^2}\right],\ i = 1, \ldots, n. \qquad (1)$$

Here $dir_{\mu}$ is the peak of the signal, $a$ is the height of the peak and $\phi$ is the width of the curve. As in McMillen and Behseta (2010), we interpret the $S^{\mu}$ in terms of approximately Gaussian MT tuning curves and weights from MT to LIP that preserve this Gaussian tuning in the LIP units. Notice that if $\phi = 0$, then

$$S_i^{\mu} = a\,\delta_{i,dir_{\mu}},$$

where $\delta_{i,j}$ is the Kronecker delta, so that the signal is concentrated in the channel $dir_{\mu}$. But, if $\phi > 0$, the signal will have a spread around the peak. For MT units associated with the dot-motion task, tuning curves have been measured to have a width of about 40° (Law & Gold, 2008). The situation is illustrated in Fig. 2. Angles far apart have very little overlap in the signals, but when the angles are close the overlap is substantial. For a two-alternative task in which dots travel on average either up or down, the signals have very little overlap. Signals for alternatives corresponding to more similar motion directions have more overlap.

We model the LIP layer as a set of $n$ leaky competing accumulators. The linearized model for their evolution is a stochastic differential equation (Usher & McClelland, 2001; Bogacz et al., 2006; McMillen & Holmes, 2006):

$$dx_i = \left(-kx_i - m\sum_{j \neq i} x_j + S_i\right) dt + c\,dB_i,\ \ i = 1, \cdots, n, \quad (2)$$

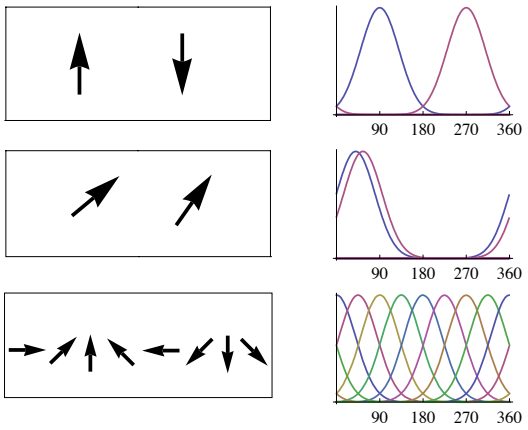


Figure 2: Possible directions of coordinated movement (left panels) and corresponding signal vectors (right panel).

where $k$ is the decay rate, $m$ is the mutual inhibition, and $B_i$ is a Wiener process (integrated white noise) representing the noise in the signal and from other sources. The signal-to-noise ratio is the ratio of the magnitude of the largest signal to the variance of the noise, i.e. $a/c$. We can thus model changes in the direction coherence by changing this ratio. The effect of decay and inhibition is to concentrate the values of the accumulators onto the signal vectors. Thus, moderate values of $w$ and $k$ tend to increase the accuracy. Best results are achieved when decay and inhibition are balanced, i.e. $w = k$ (McMillen & Holmes, 2006). For simplicity, and to be concrete, throughout the rest of this paper we will present results for $k = w = 0.5$, $a = 2$ and $c = 1$. Results are qualitatively insensitive to these choices.

The output from accumulator $j$ feeds into the $i$th unit of SC with weight $w_{ij}$. SC units apply step functions $H$ with thresholds $\Theta$ to their inputs. A response is made when SC unit $j$ transitions from 0 to 1 (i.e., when $y_j = \sum_{j=1}^{n} w_{ij} x_j > \Theta$).

The results in this paper are generally applicable, but to be precise we consider a motion direction task with 36 accumulators and interpret these as representing increments of 10°. If the direction $j \cdot 10°$ is presented, the signal vector takes the shape $S_i^{\mu}$ as in (1), with $dir_{\mu} = j$. For concreteness we consider four possible directions of motion: $30°, 60°, 140°, 220°$. Thus, if say, the direction of coordinated movement is 60°, the signal vector has a peak at the sixth accumulator. The four possibilities are represented by the four possible signal vectors with peaks at accumulators 3, 6, 14 and 22. In this paper we only consider the case when all the possibilities are equally likely, in which case the appropriate initial condition for the accumulators is $x_i(0) = 0$.

McMillen and Behseta (2010) showed that the optimal weights $w_{ij}$ in the above are achieved when the weights mimic the shape of the possible incoming signal vectors. That is to say, a threshold crossing test best approximates an MSPRT when $w_{ij} = S_j^{\mu_i}$. The magnitude of the weights

are not important in terms of optimality, as the magnitude may be incorporated into the thresholds. The performance of the threshold crossing tests is illustrated in Fig. 3. Here we consider a test with 36 accumulators and the four alternatives as described above. In Fig. 3 we plot the mean response time (MRT) for a fixed value of the error proportion (ER). For each value of the spread we compute the threshold such that ER $= 0.1$, and find the corresponding MRT. Each panel demonstrates an important fact, as we elucidate below.

In the left panel of Fig. 3 we take the signal vectors to be as (1), and allow $\phi$ to vary. Thus, $\phi = 0$ corresponds to the case when the signal is concentrated in a single channel. Positive values of $\phi$ correspond to signals that are spread about a peak. In these computations, the weights are as desired for MSPRT approximation, i.e. $w_{ij} \propto S_j^{\mu_i}$. This panel thus shows the minimal MRT that can be achieved by a threshold crossing test for an ER of 0.1. We see that there is an advantage to a moderate spread in the signals if this information can be utilized by the decision mechanism. In fact, the optimal spread is near $\phi = 3$. It is interesting to note that this corresponds to a width in the shape of the signal vectors of about $30^\circ$, while the width of tuning curves in MT associated with the direction task as measured in Law and Gold (2008) are approximately $40^\circ$.

In the right panel we fix the spread in the signal vectors at $\phi = 4$, and compute MRT for various spreads in the weights. In order to get an idea of how the spread in the shape of the weights affects performance when the signal shape is fixed, in these simulations we suppose that the weights also have a Gaussian shape:

$$w_{ij} = w_0 \exp\left[-\frac{(i-j)^2}{2\phi_W^2}\right],\ j = 1,\ldots,n,$$

where $w_0$ is a normalizing factor chosen so that $\sum_{j=1}^n w_{ij}^2 = 1$ (this normalization step is not in fact required). The spread $\phi_W$ controls how the values of the accumulators are weighted before making a decision. In the case $\phi_W = 0$, we have $y_i = x_i$, so that the accumulator values are not weighted. When $\phi_W = \infty$, each $y_i$ is the same, i.e. the sum of all accumulators. The right panel of Fig. 3 shows that MRT is minimized when $\phi_W = \phi$. That is, the optimal weights occur when the width of the weight shape is the same as that in the signal vector.

To reiterate, a moderate spread in the signals is a significant advantage, but only if the LIP-to-SC weights can be tuned to take on the same shape as the possible signal vectors defined by MT activity. In the following section we consider how the weights may be modified over the course of trials.

## An algorithm for learning the LIP to SC weights

We propose a simple Hebbian weight learning algorithm for the weights $w_{ij}$. The learning algorithm is a modification of a classical Widrow-Hoff rule (see, e.g., Hertz, Krogh, & Palmer, 1991). In rules of this type, the connection strength being modified acts as a filter that tracks an input signal. At

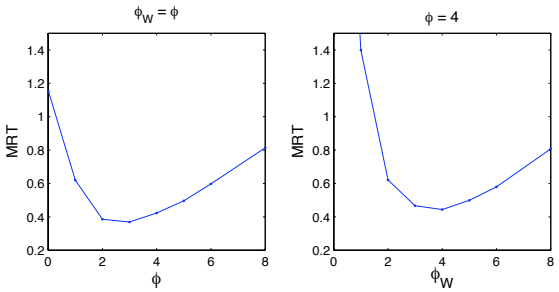


Figure 3: Effects of signal spread and weight shape. Left panel: Simulated MRT vs. spread in the signal vectors, where the weights have the same shape. Right panel: MRT vs. spread in shape of weights with signal vector fixed with $\phi = 4$. In all cases the threshold is such that ER $= 0.1$.

any point, its value is an approximately exponentially decaying, weighted average of past input values. High frequency changes in this signal (representing noise) are filtered out by the algorithm, producing little change in the updated weight. In contrast, low frequency signal changes (representing, hopefully, the uncorrupted input signal) produce significant changes in the weight. If the signal is constant and noise is absent, the weight will converge approximately exponentially on the value of the signal. If what is being tracked is a signal that depends on the product of activations in a sending unit and a receiving unit, then this rule is simply a Hebbian update rule with a decay term for forgetting old co-activation levels — a useful feature in a noisy neural system.

After each trial the subject responds with a choice among alternatives, say $i$. At this time the weights to the output unit $z_i$ corresponding to the choice made are updated, according to whether a reward is received or not. Then, if the choice corresponding to $z_i$ is chosen, the weights are updated by the rule

$$w_{ij}^{\text{new}} = (1-\alpha) w_{ij}^{\text{old}} + \alpha \Delta w_{ij}, \quad (3)$$

$$\Delta w_{ij} = r z_i x_j, \quad (4)$$

where $r$ is the magnitude of the reward, and $\alpha$ is the learning rate. Notice that only the weights to the unit corresponding to the choice made are updated, and this is the sense in which the rule is Hebbian. For simplicity, we assume here that a reward is either earned or not, so that $r$ is either 1 or 0 depending on whether a correct decision is made.

Thus, after each trial, if a correct decision is made the weights to the correct output unit are increased in proportion to the values of the accumulators $\mathbf{x}$. There is no need to estimate the probability of making a correct decision or an expected value of the reward, as for example in reinforcement learning methods, since only the values of the units are used in the update rule. With this rule the weights track the shape of the vectors being passed from the LIP layer. The weights thus tend to oscillate around the means of the accumulator

values, $\langle x_j(t) \rangle$.

The accumulator values on average take on the shape of the signal vector from the MT layer. This can be proved analytically, but here we show only simulation results. The update rule (3-4) thus causes the weights to track values whose means take on the shape of the MT-to-LIP signal vectors. Therefore the weights tend, on average, to mimic the shape of the signal vector, with oscillations about this shape that depend on the learning rate.

## Results of simulations

Figs. 4 and 5 show results of simulations using the update rule (3 - 4). The weights are initially chosen randomly, with a peak added at $w_{ii}$. Fig. 4 shows how the weights evolve over time, and how this affects the performance of the subject. The reward rate continually increases on average, and the ER continually decreases. The bottom panels show the weights to SC corresponding to $i = 14$, or to angle 140°. The weights for the other alternatives behave similarly. Simulations in which the weights are chosen differently show similar improvements in performance and similar matching of the weight profiles to the signal vector shapes. Cases in which the weights are all chosen randomly show a more dramatic improvement in reward rate (RR) since then the accuracy will initially be very low. Fig. 4 shows that even when the weight has a peak at the right position, a dramatic improvement occurs: for example, the RR more than doubles and the RT and ER both decrease over time.

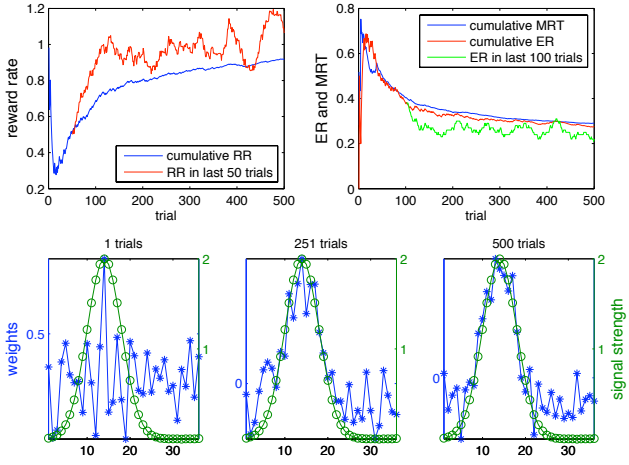


Figure 4: Effects of weight learning rule. The threshold is fixed at $z = 1$. There are four alternatives (3, 6, 14, 22), and the learning rate is $\alpha = .05$. In the bottom panel the signal strength is plotted on the right axis (circles), and the weights are shown on the left axis (stars). The inter-trial delay used in the calculation of reward rate (RR) here is 500 msec.

Figure 4 shows one block of 500 trials. In order to see how the weight update rule behaves on average, we carried out the same simulation for a number of blocks and averaged

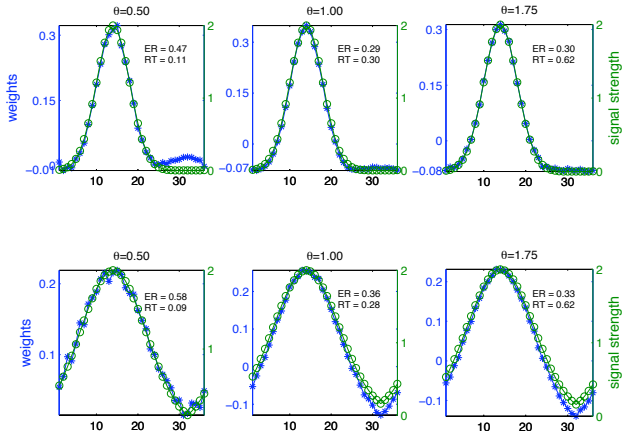


Figure 5: Averaged weights over 150 blocks of 500 trials. In the top row $\phi = 4$; in the bottom row $\phi = 8$.

the weights over each block, and then took the average over 150 blocks of trials. Fig. 5 shows the averaged weights for different values of the threshold, as well as different values of the spread in the signals. We see that on average, the weight profile shape is very close to the signal shape. Also indicated in these figures are the ERs and MRTs for these blocks of trials. Notice that in the lower left panel, the ER = .58 is not much smaller than would be achieved by random guessing (.75). In this case the threshold is very small, as is the corresponding MRT of .09. In this situation it will take the weights much longer to learn the shape of the signal vectors, since most of the time the decision will be incorrect. This is why the weights appear more erratic in this frame than in the others. However, even in this case, the average values of the weights take the same shape as the signal vector. Similar comments apply, *mutatis mutandis*, to the upper left panel.

Generally, the model is insensitive to changes in the parameters $a, c, k, m$, in the sense that the weights tend on average toward the optimal weight shape mimicking the shape of the signal vectors. If the learning rate $\alpha$ is made smaller, the weights take longer to track to the shape of the signals, but there is less variation around these mean values.

Fig. 6 shows the dynamics of evidence accumulation within trials, demonstrating that Gaussian bumps of activation arise on the LIP layer (preserving the Gaussian input signal profiles, and therefore producing Gaussian LIP-to-SC weights through Hebbian learning).

## Discussion

The simple rule (3-4) works remarkably well at learning the shapes of the signal vectors from MT to LIP. This leads to a dramatic improvement in performance, and occurs without any direct connection to the MT layer. The three layer model incorporates integration of information, a rule for making the decision, as well as a simple algorithm for learning to optimize reward rates by learning the shapes of the vectors of

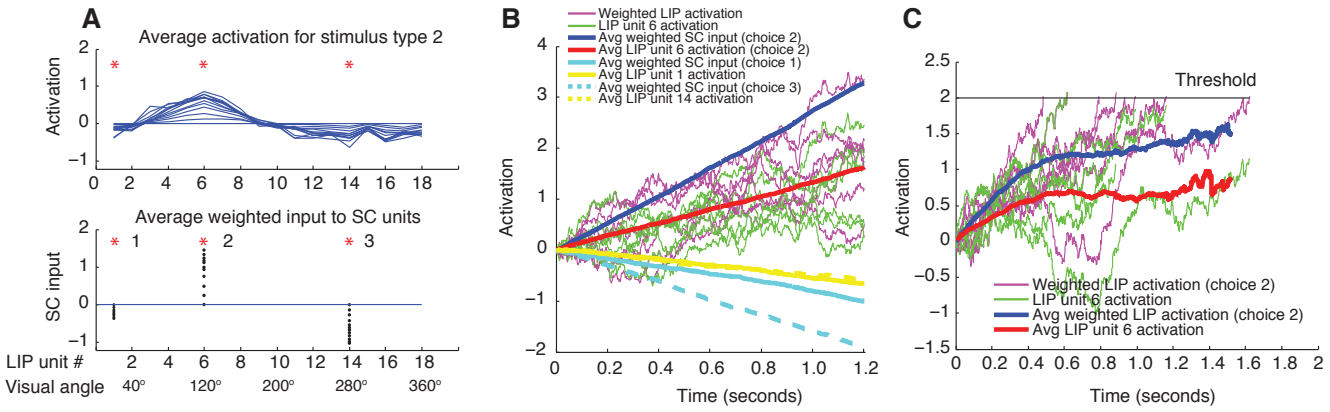


Figure 6: Panel A, top, shows LIP unit activations at several time points within a decision. Activations are averaged over many instances of stimulus type 2, which produces maximal activation in LIP unit 6 (visual angle 120°; here we arbitrarily quantized visual angle into 18 levels). Panel A, bottom, shows the weighted values of these activations feeding into each of 3 SC units. Panel B shows the average state of evidence accumulation for choice 2 (input to SC unit 2; blue) and average LIP unit 6 activity (red) within fixed-viewing time trials, without thresholds applied to the evidence (the interrogation protocol). Panel C shows the average state of weighted evidence accumulation for choice 2 (blue) and average unit 6 activity (red) within free response trials (black line indicates threshold). The weighted sum produces a higher signal-to-noise ratio, and therefore better performance than evidence from unit 6 alone. Red and blue traces fall off over time because the average is based on fewer and fewer trials as time progresses (more and more decisions have already taken place by the end of the plot).

neural signals coming from an input layer. These features are essential elements of a complete decision-theoretic model.

## Acknowledgments

P. Simen was supported by a postdoctoral training fellowship from the National Institute of Mental Health (MH080524).

## References

Bogacz, R., Brown, E., Moehlis, J., Hu, P., Holmes, P., & Cohen, J. (2006). The physics of optimal decision making: A formal analysis of performance in two-alternative forced choice tasks. *Psych. Rev.*, *113*(4), 700-765.

Bogacz, R., & Gurney, K. (2007). The basal ganglia and cortex implement optimal decision making between alternative actions. *Neural Comput.*, *19*(2), 442–477.

Butts, D. A., & Goldman, M. S. (2006). Tuning curves, neuronal variability, and sensory coding. *PLoS Biol*, *4*(4), e92.

Dragalin, V., Tartakovsky, A., & Veeravalli, V. (1999). Multihypothesis sequential probability ratio tests, part I: Asymptotic optimality. *IEEE Trans. Inform. Theory*, *45*, 2448-2461.

Ghose, G., Yang, T., & Maunsell, J. (2002). Physiological correlates of perceptual learning in monkey V1 and V2. *J. Neurophysiol.*, *87*, 1867-1888.

Gold, J., & Shadlen, M. (2001). Neural computations that underlie decisions about sensory stimuli. *Trends in Cognitive Sciences*, *5*, 10-16.

Hertz, J., Krogh, A., & Palmer, R. (1991). *Introduction to the theory of neural computation*. Redwood City, CA: Addison-Wesley Publishing Company Advanced Book Program.

Law, C.-T., & Gold, J. (2008). Neural correlates of perceptual learning in a sensory-motor, but not a sensory cortical area. *Nat. Neurosci.*, *11*, 505-513.

McMillen, T., & Behseta, S. (2010). On the effects of signal acuity in a multi-alternative model of decision making. *Neural Comput.*, *22*(2), 539-580.

McMillen, T., & Holmes, P. (2006). The dynamics of choice among multiple alternatives. *J. Math. Psych.*, *50*(1), 30-57.

Pouget, A., Deneve, S., Ducom, J.-C., & Latham, P. (1999). Narrow vs. wide tuning curves: what's best for a population code? *Neural Comput.*, *11*, 85-90.

Seriés, P., Latham, P., & Pouget, A. (2004). Tuning curve sharpening for orientation selectivity: coding efficiency and the impact of correlations. *Nat. Neurosci.*, *7*(10), 1129-1135.

Shadlen, M., & Newsome, W. (2001). Neural basis of a perceptual decision in the parietal cortex (area LIP) of the rhesus monkey. *J. Neurophysiol.*, *86*, 1916-1936.

Usher, M., & McClelland, J. (2001). On the time course of perceptual choice: The leaky competing accumulator model. *Psych. Rev.*, *108*, 550-592.

Wald, A., & Wolfowitz, J. (1948). Optimal character of the sequential probability ratio test. *Ann. Math. Statist.*, *19*, 326-339.

Zhang, K., & Sejnowski, T. (1999). Neural tuning: to sharpen or broaden? *Neural Comput.*, *11*, 75-84.