A model of perceptual decision making in lateral intraparietal area

Andrew Saxe **Josh Herbach**

Advised by J. McClelland

1 Introduction

We model an oculomotor decision-making experiment in which monkeys are shown a cloud of moving dots and must decide which way the dots are moving on average. In each trial of the experiment a subset of the dots moves coherently to the left or right, while the remaining dots move randomly. The monkey indicates its choice by saccading to the left or right, and if correct it receives a juice reward. This standard paradigm has recently been extended to include differing reward conditions [1]: Before the onset of the motion stimulus, a reward stimulus is presented that indicates how many drops of juice the monkey will receive for a correct answer in each direction.

Analysis of the behavioral data from this experiment shows that the monkeys achieve a near-optimal reward rate on this task [1]. However, the neural circuits underlying this decision process remain unclear.

We attempt to explain the neural data using a computational neuroscience approach. We hypothesize the computation that we believe LIP performs, find the optimal solution to that computation, and see if the optimal solution predicts the experimental data. In particular, we hypothesize that LIP implements an optimal decision rule: it receives noisy observations of the motion direction as input, and produces the decision that will maximize the monkey's expected reward rate as output. In section 2 we formulate a probabilistic model of the task. In section 3 we find the optimal decision rule. In section 4 we set the parameters of our model to closely approximate previous modeling efforts. In section 5, we present a stochastic neural network representing LIP that we train to implement this optimal decision rule. Finally, section 6 makes comparisons to experimental data.

2 Modeling the Computation

Because we intend to train a recurrent neural network to implement the optimal decision rule, we first formulate the computational problem solved by the monkey in a discrete-time setting.

In a given trial of the experiment, a certain fraction of dots move coherently to the left or right and the rest move randomly. Define the signed coherence $c \in [-1, 1]$ of a trial to be the fraction of dots moving coherently with the sign indicating if the movement is to the left or right respectively. Thus the correct answer $y = \text{sign}(c)$. Correct responses are rewarded with either one or two drops of juice. Define $r_+, r_- \in \{1, 2\}$, where r_+ is the reward magnitude for a correct response with $y = 1$ and r_- is the reward magnitude for a correct response with $y = -1$.

During a trial, LIP receives input from lower-level visual processing areas sensitive to motion direction. We describe this input signal as a series of noisy inputs $X = \{x^{(t)}, t = 1, ..., T\}$, where LIP receives input $x^{(t)}$ at time t. Here the time period during which the monkey views the moving dots has been discretized into T instants. In the simplest model, let $x^{(t)} \in \{-1, 1\}$ represent the observed stimulus direction. We take the $x^{(t)}$ to be mutually independent given y, and will specify $p(x^{(t)})$ by defining a measurement model $p(x^{(t)}|c)$ and a prior over coherences for a trial $p(c)$. The experiment consists of blocks of trials with fixed coherence magnitude $|c| = c_{mag}$. For each trial the leftward or rightward direction is chosen randomly, i.e. $p(y = 1)$ is known and set by the experimenter. Thus $p(c) = \delta(c - c_{mag})p(y = 1) + \delta(c + c_{mag})p(y = -1)$.

To find $p(x^{(t)}|c) = p(x^{(t)}|y, c_{mag})$, we will require that the probability of error under the optimal decision function in the equal reward case $r_+ = r_-$ be identical to that of previous modeling approaches based on the continuous time

drift diffusion process [1,2]. Since previous approaches have treated the equal reward case, this requirement will make our formulation match previous work. To do so we must know the optimal decision rule.

3 The Optimal Decision Rule

We now derive the optimal decision rule as a function of $p(x^{(t)}|c)$ and the other parameters. Let $p' = p(x^{(t)} = 1|y = 1, c_{mag})$. We can now compute the expected reward R for a single trial given a response $D(X, r_+, r_-, c_{mag}) = 1$,

$$E[R|D(X, r_+, r_-, c_{mag}) = 1] \quad = \quad r_+ p(y = 1|X, c_{mag}) \tag{1}$$

$$= \quad r_+ \alpha p(y = 1) \prod_{t=1}^{T} p(x^{(t)}|y = 1, c_{mag}). \tag{2}$$

(the formula for the expected reward is similar when $D(X, r_+, r_-, c_{mag}) = -1$). Since we have defined D^* to be a function maximizing the expected reward rate, D^* should choose 1 when $E[R|D(X, r_+, r_-, c_{mag}) = -1] < E[R|D(X, r_+, r_-, c_{mag}) = 1]$ or alternatively (since $E[R|D(X, r_+, r_-, c_{mag}) = i] \geq 0$), it chooses 1 when

$$1 \quad < \quad \frac{E[R|D(X, r_+, r_-, c_{mag}) = 1]}{E[R|D(X, r_+, r_-, c_{mag}) = -1]}. \tag{3}$$

Manipulating this eventually yields

$$1 \quad < \quad \frac{r_+}{r_-} \frac{p(y = 1)}{p(y = -1)} p'^{\sum_t x^{(t)}} (1 - p')^{-\sum_t x^{(t)}}. \tag{4}$$

Now we take the log of both sides to get the condition $D^*(X, r_+, r_-, c_{mag}) = 1$ if

$$0 \quad < \quad \log\left(\frac{r_+}{r_-}\right) + \log\left(\frac{p(y = 1)}{p(y = -1)}\right) + \log\left(\frac{p'}{1 - p'}\right) \sum_t x^{(t)}, \tag{5}$$

and -1 otherwise.

4 A Measurement Model in Agreement with the Drift Diffusion Process

In the equal reward, $p(y = 1) = 1/2$ case (5) becomes

$$D^*(X, r_+, r_-, c_{mag}) = \mathbf{1}\left\{\sum_t x^{(t)} > 0\right\} * 2 - 1.$$

Let $N = \sum_{t=1}^{T} \mathbf{1}\{x^{(t)} = 1\}$. Then $N \sim \text{Binomial}(T, p')$. Assuming $y = 1$, the probability of error is the probability that the sum of T Bernoulli trials – each with probability of success $p(x^{(t)} = 1|y = 1; r_+, r_-, c_{mag})$ – is less than $T/2$. To find p' we can require that as $T \to \infty$, the error matches the error of the drift diffusion model. By applying the de Moivre-Laplace theorem and setting the result equal to the error of the drift diffusion model, we find that

$$p' \quad = \quad \frac{1}{2} + \frac{1}{2\sqrt{\frac{k^2 T}{A^2 t_f} + 1}}. \tag{6}$$

A is a drift rate parameter representing the strength of the motion signal, and k is a parameter controlling the amount of noise. Unsurprisingly, as the influence of noise (k) increases, the signal portion of p' decreases and as the meaningful drift increases (A) the signal has a larger impact. In [1,2] the drift rate A is assumed to be proportional to coherence, $A = ac$ where a is a parameter. Substituting this into (6) yields

$$p' \quad = \quad \frac{1}{2} + \frac{1}{2\sqrt{\frac{k^2 T}{a^2 c^2 t_f} + 1}}, \tag{7}$$

from which $p(x^{(t)}|y, c_{mag})$ can be recovered. Thus our discrete time model and the continuous models in [1,2] will be close approximations of each other for large T. Figure 1a shows the quality of approximation for $T = 15$, the number of samples used subsequently.
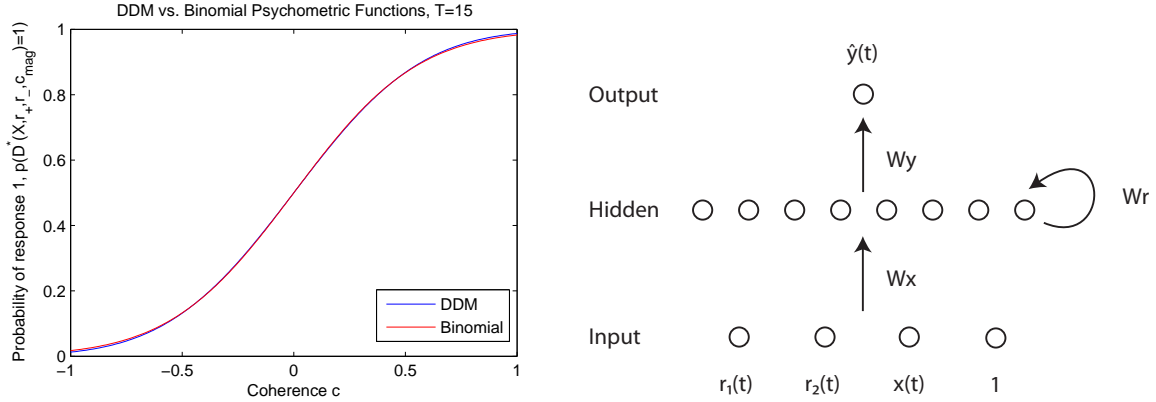
Figure 1: (a) Comparison of psychometric functions under a DDM and the discrete time optimal decision function (5), with $p(x(t)|c)$ as given in (7) and $T = 15$, $t_f = 5$, $a = 1$, $k = 1$; (b) Recurrent network architecture.

5 Implementing the decision rule in a recurrent neural network

The optimal decision rule given in (5) can only predict behavioral data. To make the connection to neural data, we train a recurrent neural network to implement the optimal decision rule and see if the firing rates of the network match those recorded in LIP. Our network consists of three layers (Figure 1b). The input layer presents the vector $u^{(t)}$ where $u_1^{(t)} = \mathbf{1}\left\{x^{(t)} = 1\right\}$, $u_2^{(t)} = \mathbf{1}\left\{x^{(t)} = -1\right\}$, $u_3^{(t)} = r_+$, $u_4^{(t)} = r_-$, and $u_5^{(t)} = 1$. The components $u_1^{(t)}$ and $u_2^{(t)}$ split the input $x^{(t)}$ into two processes so that neurons in the network can become responsive to only the leftward or rightward directions (or to a mixture of both). It projects to a hidden layer with weights W^u. Each neuron in the hidden layer receives input from all other hidden neurons through recurrent connections W^r, and sends output to the output layer through weights W^y. The output layer consists of a single neuron which outputs the decision $\hat{y}^{(t)}$.

In other contexts noise has been found to be crucial to reproducing observed properties of experimental data [3] and so we have trained networks in both a noiseless, deterministic setting, and a noisy, stochastic one.

Let $o^{(t)}$ be the hidden neuron activation at time t. This activation is updated according to

$$o_i^{(t+1)} = f\left(\sum_j W_{ij}^r(o_j^{(t)} + \zeta_{ij}^{(t)}) + \sum_j W_j^u(u_j^{(t)} + \eta_i^{(t)} + \nu_i)\right),$$

where $f(x) = \frac{1}{1+\exp{-x}}$, $\zeta_{ij}^{(t)} \sim \text{Gaussian}\left(0, \alpha\frac{o_j^{(t)}}{\left(1+ao_j^{(t)}\right)^2}\right)$, $\eta_i^{(t)} \sim \text{Gaussian}\left(0, \sigma_\eta^2\right)$, $\nu_i \sim \text{Gaussian}\left(0, \sigma_\nu^2\right)$, and all random variables are mutually independent. The noise terms were proposed by Todorov in [3]. The random variable $\zeta_{ij}^{(t)}$ models the poisson firing characteristics of neurons in vivo and synaptic depression effects. The random variables $\eta_i^{(t)}$ and ν_i model input noise, with $\eta_i^{(t)}$ representing fluctuating noise and ν_i representing slowly varying noise that remains constant over the course of a trial. We took $\alpha = .1$, $a = .1$, and $\sigma_\eta^2 = \sigma_\nu^2 = .001$.

The output of the network $\hat{y}^{(t)}$ is calculated as

$$\hat{y}^{(t)} = 2f\left(\sum_j W_j^y o_j^{(t)}\right) - 1.$$

To train the network we minimize the expected error of the network over a set of training data. We use the back propagation through time algorithm [4] to calculate the gradient of the error with respect to the network parameters, and make use of noise freezing tricks described in [3] to speed the computations.

6 Comparison to data

The firing rates of hidden units in the trained network show two typical patterns. In the first, the unit begins each trial with a low activation, and its activation increases as it receives information favoring one direction. We interpret this
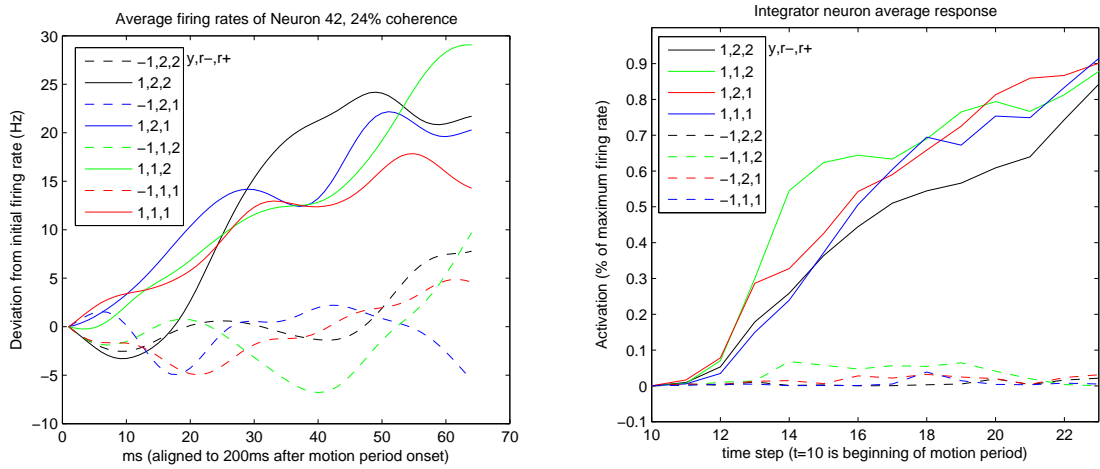
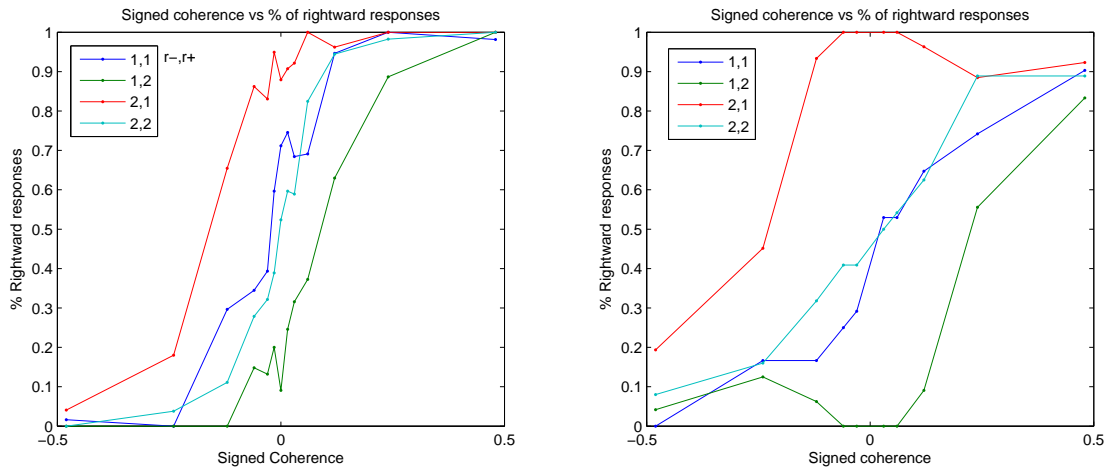Figure 2: Firing rates. (a) Monkey data; (b) Model

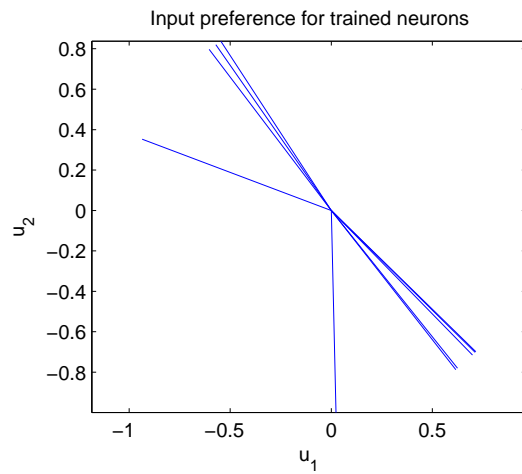Figure 3: Psychometric functions. (a) Monkey data; (b) Model

Figure 4

neuron as tracking the sum of $x^{(t)}$, and hence call it an integrator neuron. The average activity of a typical integrating neuron is shown in Figure 2b. In the second pattern, the unit begins near its maximum activation, and its activation decreases as it receives information favoring one direction. We call this type of unit an reverse integrator neuron.

We examined neural data collected and provided to us by A. Rorie, W.T. Newsome, and J. Guan for neurons exhibiting these patterns in monkeys performing the task. We found four neurons of fifty-two that robustly showed this integrating response pattern. One such neuron is shown in Figure 2a (compare to Figure 2b). A number of other neurons showed robust integrating processes in some but not all of the reward conditions. However, no reverse integrators were found in the monkey data. This observation is consistent with the general finding of sparse firing rates in the brain. Although we expected reverse integrators in our deterministic network, we had hoped that they would not be present in the stochastic network since the variance of the noise added to each hidden neuron scales linearly with that neuron's activation. We reasoned that this would push the network toward lower firing rates to reduce the impact of this noise, as reported by [3]. However this was not observed, and our stochastic network was very close in character to the deterministic one.

To compare the behavioral data to that of the model, we analyze the percentage of rightward responses as a function of coherence and reward condition. The resulting curve is called a psychometric function, and is shown in Figure 3a for the monkey data. The psychometric function shows intuitive behavior: the equal reward conditions yield indistinguishable curves, and the asymmetric reward conditions shift the curves in the direction of higher reward. Figure 3b shows the psychometric function calculated based on five trained networks, each one trained for a different coherence level. The curves show the same general pattern, except that the asymmetric reward conditions for the model go to zero or one at coherences near zero. This pattern is actually optimal for our discrete setting of the task, since the sum of the inputs is bounded by T, i.e. $\left|\sum_t x^{(t)}\right| \leq T$, so for $\left|\frac{log(r_+/r_-)}{log(p'/(1-p'))}\right| > T$ the optimal decision will be fixed regardless of the input sequence. Since these bumps are clearly not observed in the monkey data, our computational model of the task is too crude to properly capture monkey behavior near low coherences. To mitigate these problems in the future, the parameters could be chosen so that $\frac{log(r_+/r_-)}{log(p'/(1-p'))}$ is smaller than T.

Finally, we examine the structure of the learned solution. Models of the equal reward case often posit one group of neurons sensitive to motion in one direction and another group sensitive to motion in the other direction. Our model had the freedom to become selective to any mixture of the two input components u_1 and u_2. To see if neurons in the model become sensitive to one direction of motion or the other, we plot the unit vector in the direction of the weights to these input components in Figure 4. A weight vector aligned with one axis or the other would indicate a neuron selective only to one direction of motion. Instead, we see weight vectors aligned with the negative diagonal, indicating opposed input weights of equal magnitude. That is, our neurons become selective to both directions of motion.

7   Conclusion

The model presented in this paper exhibits a number of features of the neural and behavioral data for monkeys, yet the discrepancies between the model and data remain large. We believe that the successes of the model argue for continued investigation. If the basic hypothesis is correct, i.e., the neural responses are indeed the result of optimizing the neural system subject to constraints imposed by its architecture and biological substrate, then we can set conditions under which we would expect the model to reproduce the experimental data. First, we can ask if we have the proper model of the computation. The optimal solution to the proper model should match the behavioral data. Next, we can ask if we have successfully trained a neural network to implement the optimal decision rule. Since we know the optimal rule, it is easy to determine whether the problem lies in the training process. Finally, we can ask if we have successfully captured the constraints of the biological system.

References

[1] S. Feng, P. Holmes, A. Rorie, and W.T. Newsome. "Can monkeys choose optimally when faced with noisy stimuli and unequal rewards?" Submitted, *PLOS Computational Biology*, Aug 2008.

[2] R. Bogacz, E. Brown, J. Moehlis, P. Holmes, J.D. Cohen. "The physics of optimal decision making: A formal analysis of models of performance in two-alternative forced choice tasks." *Psychological Review*, Vol 113(4), Oct 2006, 700-765.

[3] E. Todorov. "Mixed muscle-movement representations emerge from optimization of stochastic sensorimotor transformations." Submitted 2007.

[4] P.J. Werbos. "Back Propagation Through Time: What it Does and How to Do it." *Proceedings of the IEEE*, Vol 78(10), 1990, 1550-1560.