Matching and maximizing are two ends of a spectrum of policy search algorithms

January 2, 2004

Abstract

According to the matching law, when an animal makes many repeated choices between alternatives, its preferences are in the ratio of the incomes derived from the alternatives. Because matching behavior does not maximize reward, it has been difficult to explain using optimal foraging theory or rational choice theory. Here I show that matching and maximizing can be regarded as two ends of a spectrum of policy search algorithms from reinforcement learning. The algorithms are parametrized by the time horizon within which past choices are correlated with present reward. Maximization corresponds to the case of a long time horizon, while matching corresponds to a short horizon. From this viewpoint, matching is an approximation to maximizing, with the advantage of faster learning and more robust performance in nonstationary environments. Between these two ends of the spectrum lie many strategies intermediate between matching and maximizing.

If an animal's relative preferences for alternatives are in the ratio of the incomes derived from them, then its behavior is said to be "matching." Matching behavior has been observed for certain types of reinforcement schedules, in particular those that randomize the interval between reward. The matching law was important because it gave the law of effect a quantitative formulation.

Given the matching law as an empirical observation about behavior, two questions immediately come to mind. The first question is functional: why is matching a good policy for animals to follow? The second is mechanistic: what neural mechanisms underlie the production of matching behavior? This note mainly addresses the first question, by elucidating the function of matching from the viewpoint of the mathematical theory of reinforcement learning. However, the second question is also peripherally addressed through mathematical developments that are shared by recent neural network models of matching behavior.

One of the most common ways to explain the function of a behavior is to argue that it has been adapted by evolution to be optimal. Such an explanation for matching has been elusive, because matching does not generally maximize the animal's overall rate of reward. In this respect, the matching law is suboptimal. This enables it to be used as an explanation for "irrational" human behaviors, such as addiction and other behaviors attributed to lack of "self-control." Nevertheless, it would be hasty to completely reject optimality as an explanation of matching. Often matching is close to optimal, even if it is not exactly so.

The purpose of this note is to point out that matching and maximizing are two ends of a spectrum of behaviors generated by the REINFORCE class of machine learning algorithms. Such algorithms learn by correlating past actions with reward, through an eligibility trace that maintains a memory of past actions. The time constant of the eligibility trace is an important parameter of the learner. If the time constant is sufficiently long, then the learner converges to maximizing behavior. On the other hand, if the time constant is extremely short, then the learner converges to matching behavior. This type of behavior is similar to the melioration model of Herrnstein, Prelec, and Vaughan [].

Why would a learner pursue a strategy that converges to matching rather than maximizing? In general, a bias-variance tradeoff is inherent to REINFORCE algorithms. They work by computing a stochastic approximation to the gradient of the expected reward. When the time constant of the eligibility trace is long, the bias of the gradient estimate is small, but the variance is large. Reducing the time constant of the eligibility trace increases the bias but decreases the variance, and can therefore be advantageous for fast learning and robust performance in nonstationary environments.

1 The matching law

Suppose that an animal is presented with repeated choices between two actions. For simplicity, it's assumed that the choices occur in discrete trials, though many matching experiments have been performed using continuous time. This section defines some notation, and uses it to express the matching law in mathematical form.

The action taken in trial t is indicated by the binary variables a_t and $\bar{a}_t$, which satisfy $a_t + \bar{a}_t = 1$. After each action, the animal receives reward h_t. The average income derived from action a is

$$H = \frac{1}{T} \sum_{t=1}^{T} h_t a_t$$

Similarly, the average income derived from action $\bar{a}$ is

$$\bar{H} = \frac{1}{T} \sum_{t=1}^{T} h_t \bar{a}_t$$

The frequencies of actions a and $\bar{a}$ are respectively

$$f = \frac{1}{T} \sum_{t=1}^{T} a_t \qquad \bar{f} = \frac{1}{T} \sum_{t=1}^{T} \bar{a}_t$$

and satisfy $f + \bar{f} = 1$.

According to the matching law, the frequencies are in the same ratio as the incomes,

$$\frac{H}{f} = \frac{\bar{H}}{\bar{f}}$$

Equivalently, the matching law can be stated as equality of the returns from the two actions, where the return from action a is defined as H/f, the reward averaged only over those trials in which action a was chosen.

2 Matching is not equivalent to maximizing

In the field of reinforcement learning, a method of choosing actions is called a policy. Is matching an optimal policy?

To address this question, we consider the class of policies in which actions a and $\bar{a}$ are chosen with probabilities p and $\bar{p}$ respectively, with $p+\bar{p}=1$. Each trial's action is assumed to be statistically independent of previous trials. This case is easy to analyze; generalization to more complex policies is left to the technical appendix.

For the policy indexed by p, we define $H(p)$ and $\bar{H}(p)$ as the incomes derived from the two actions, averaged over time. It is assumed that the reward process has some sort of statistical stationarity, so that these time averages are well-defined.

The total average income is given by the sum $H(p)+\bar{H}(p)$, and the optimal policy is found by maximizing with respect to p. Assuming smoothness, the optimal p should be a stationary point of the sum, assuming that it is strictly between 0 and 1. This means that

$$\frac{dH}{dp} = -\frac{d\bar{H}}{dp} = \frac{d\bar{H}}{d\bar{p}}$$

In other words, the optimal p is given by the condition that the marginal incomes dH/dp and $d\bar{H}/d\bar{p}$ are equal. In general, this is not the same as the matching law, which is the condition that the returns H/p and $\bar{H}/\bar{p}$ are equal. In short, matching is not typically the same as maximizing.

3 REINFORCE learning

Suppose that the learner does not know the functions $H(p)$ and $\bar{H}(p)$, so it cannot compute the optimal p directly. Based on observations of its actions and rewards, how can the learner find the optimal p? One method is provided by the REINFORCE class of learning rules, which are an important class of policy search algorithms.

Define the *eligibility* at trial t to be

$$e_t = \bar{p}_t a_t - p_t \bar{a}_t \tag{1}$$

This is positive for action a and negative for action $\bar{a}$, with zero mean.

The *eligibility trace* is a short-term memory of recent eligibilities, and can be defined in various ways. For example, it could be the sum of recent eligibilities,

$$\hat{e}_t = \sum_{\tau=0}^{\tau_c} e_{t-\tau}$$

This will be called "sharp discounting," because of the sharp cutoff at $\tau = \tau_c$. This parameter will be called the consequential horizon, because it determines the temporal range over which the learner correlates actions with their consequences.

Or the eligibility trace could be the infinite series

$$\hat{e}_t = \sum_{\tau=0}^{\infty} \beta^\tau e_{t-\tau}$$

where the discount factor $0 \leq \beta < 1$ makes past eligibilities count less. This exponentially discounted eligibility trace can be computed in an “online” way by

$$\hat{e}_t = \hat{e}_{t-1} + \beta e_t$$

For either type of discounting, the learning rule is

$$\Delta q_t = \eta h_t \hat{e}_t$$

where $\eta > 0$ is the learning rate, and the log odds $q = \log(p/\bar{p})$ is the learned parameter. This is related to p by the monotone increasing function, $p = 1/(1 + \exp(-q))$. The choice of q as the learned parameter is advantageous for a number of reasons. One is that it implements the constraint that p must lie between 0 and 1.

4 How it works

For an intuitive explanation of how the learning rule works, consider the case of sharp discounting. If the probability p is fixed, then the eligibility trace $\hat{e}$ is equal to $(\tau_c + 1)(f - p)$, where f is the frequency with which action a was chosen in the last $\tau_c + 1$ trials. This is the deviation between the actual and expected number of a actions. Suppose that positive reward is received. If action a was chosen more times than expected, then p is increased. If action a was chosen fewer times than expected, then p is decreased. Intuitively, by monitoring how reward is correlated to fluctuations in the frequency of action a, the learner is sensitive to the marginal incomes, and such sensitivity is necessary for finding the optimal policy.

5 Maximizing behavior

There are some theoretical assurances that REINFORCE learning rules do indeed maximize average reward, though these assurances are not unconditional guarantees. For example, Baxter and Bartlett have proved the following result []. Suppose that the world plus learner forms an ergodic Markov chain. If q is held fixed. then the time average of $h_t \hat{e}_t$ is approximately equal to the derivative of the average income $H + \bar{H}$ with respect to p. This implies that the learning rule is driven by a stochastic approximation to the gradient, and therefore tends to change q in the direction that increases the average income.

The gradient approximation contains two types of error, systematic bias and random variance. The bias vanishes in the limit of $\beta \to 1$ or $\tau_c \to \infty$, which corresponds to an eligibility trace with infinite time scale. However, the variance diverges in this limit. Therefore it is best to use a finite time scale, to reduce the variance at the expense of increasing the bias. The bias will be small, as long as the time scale is much longer than the mixing time of the Markov chain. Reducing variance has the effect of speeding up the initial stages of learning, but makes the final policy suboptimal.

6 Matching behavior

The previous section discussed the case of a long time scale for the eligibility trace. The opposite extreme is to make this time scale as short as possible, $\beta = 0$ or $\tau_c = 0$. Then the eligibility trace is equal to the present eligibility, $\hat{e}_t = e_t$, and the learning rule takes the form

$$\begin{aligned} \Delta q &= \eta h_t e_t & (2) \\ &= \eta h_t (\bar{p}_t a_t - p_t \bar{a}_t) & (3) \\ &= \eta h_t (a_t - p_t) & (4) \end{aligned}$$

Suppose that the learning rule approaches some stationary probability density as time increases. In the limit of small η, this stationary density will be concentrated around a value of q satisfying $\langle h_t e_t \rangle = 0$. Substituting the expression (1) yields

$$\frac{\langle h_t a_t \rangle}{p} = \frac{\langle h_t \bar{a}_t \rangle}{\bar{p}}$$

The quantities $\langle h_t a_t \rangle$ and $\langle h_t \bar{a}_t \rangle$ are the average incomes H and $\bar{H}$. They are in the same ratio as the action probabilities, which is precisely the matching law.

There are some equivalent ways of writing the learning rule (2). For example, suppose that the learner maintains two numbers z and $\bar{z}$, which are unnormalized probabilities, so that the choice probabilities are

$$p = \frac{z}{z + \bar{z}} \qquad \bar{p} = \frac{\bar{z}}{z + \bar{z}}$$

Then the learning rule can be written as a multiplicative update

$$z_{t+1} = z_t \exp(\eta \bar{p}_t h_t a_t) \qquad \bar{z}_{t+1} = \bar{z}_t \exp(p_t h_t \bar{a}_t)$$

To prove this, simply compute the log odds $q = \log(z/\bar{z})$

Alternatively, suppose that the learner maintains two numbers u and $\bar{u}$, determining the choice probabilities via

$$p = \frac{e^u}{e^u + e^{\bar{u}}} \qquad \bar{p} = \frac{e^{\bar{u}}}{e^u + e^{\bar{u}}}$$

Then the learning rule can be written as an additive update

$$\Delta u_t = \eta \bar{p}_t h_t a_t \qquad \Delta \bar{u}_t = \eta p_t h_t \bar{a}_t$$

A modification of (2) is to make the update directly in p, rather than in the log odds q.

$$\Delta p_t = \eta h_t e_t = \eta h_t (a_t - p_t)$$

If $h_t, \eta \leq 1$, then this respects the constraint $0 < p < 1$. Then the above additive learning rule holds with

$$p = u - \bar{u}$$

7 Questions for further research

The two extreme cases $\tau_c = 0$ and $\tau_c \to \infty$ were considered above. What happens for intermediate τ_c? For example, suppose that $\tau_c = 1$. Then the learning rule is

$$\Delta\theta = \eta h_t(e_t + e_{t-1})$$

The stationary point satisfies $\langle h_t e_t \rangle + \langle h_t e_{t-1} \rangle = 0$, which implies that

$$\frac{\langle h_t a_t \rangle + \langle h_t a_{t-1} \rangle}{p} = \frac{\langle h_t \bar{a}_t \rangle + \langle h_t \bar{a}_{t-1} \rangle}{\bar{p}}$$

This deviates from matching. Can we characterize exactly how?

Suppose that the actions are generated by a Markov chain, instead of being statistically independent from previous trials. How does these results generalize?

A REINFORCE for n actions

For simplicity, the main text assumed that there are only two possible actions, and that the learned parameter was the log odds of the two actions. More generally, there could be n actions, and the probability vector might be parametrized in some other way.

Suppose that action a^i has probability p^i, where i runs from 1 to n. The probability vector is parametrized by the m-dimensional vector θ. Let ∇ denote the gradient with respect to θ. Then the eligibility is defined by

$$e_t = \sum_i \frac{a_t^i}{p^i} \nabla p^i$$

and the eligibility trace $\hat{e}_t$ is similar to before. The REINFORCE learning rule is

$$\Delta\theta = \eta h_t \hat{e}_t$$

where $\eta > 0$ is the learning rate and h_t is the reward in trial t.

B Matching behavior for n actions

Define the $m \times n$ matrix $A_{\alpha i} = \nabla_\alpha p^i$. We'll have to assume that the rank of this matrix is $n - 1$, in order to derive matching behavior. This assumption is important, because it guarantees that any vector v_i satisfying $\sum_i A_{\alpha i} v_i = 0$ is proportional to the vector of all ones. To see this, note that $\sum_i A_{\alpha i} = 0$ follows from differentiation of the identity $\sum_i p^i = 1$, and apply the fundamental theorem of linear algebra. The assumption should hold generically, provided that θ contains $n-1$ or more parameters.

Suppose that the learning rule depends only on the present eligibility,

$$\Delta\theta = \eta h_t e_t$$

Then a stationary point satisfies $\langle h_t e_t \rangle = 0$, or

$$\sum_i \frac{H_i}{p^i} \nabla p^i = 0$$

where $H_i = \langle h_t a_t^i \rangle$ is the average income derived from action i. Therefore the vector H_i/p_i should be proportional to the vector of all ones, which is the matching law.