Risk Learning: A New Model of Learning in Games

Nathan A. Collins
Santa Fe Institute
nac@santafe.edu
November 26, 2008

Abstract
Experience-weighted attraction is the leading model of learning in games. However, it can not obviously be derived from any underlying principles, so that while EWA’s success is informative of what people do when they learn to play games, it is less informative of the precise mechanisms by which they do it. I present a model of learning in games, the key insights of which are that choosing between actions in a game is like choosing between risky lotteries and that utility is not a linear function of reward. Then, a measure of risk enters the utility function, and a model that incorporates some EWA-like intuitions follows. The model has three theoretically and empirically desirable features. First, it can be derived as an incremental, approximate method for solving a particular optimization problem (estimating a value function). Second, there is evidence that the brain keeps track of the quantities the model learns. Third, the model fits experimental data about as well as EWA and a bit better than the current version, self-tuning EWA.

1 Introduction
The most successful model of learning in games, experience-weighted attraction (Camerer and Ho 1999; Ho, Camerer, and Chong 2007), owes its success to the fact that it is a hybrid of reinforcement and belief learning models — simple reinforcement learning, fictitious-play belief learning, and Cournot belief learning are all special cases. However, the same reasons the model succeeds in explaining experimental data make it theoretically unsatisfying. As an essentially phenomenological model, there is no known way to derive EWA from elementary principles. This criticism holds true especially for the latest variation, self-tuning EWA (Ho et al. 2007), which replaces two parameters in EWA with adaptive functions, one that detects changes in opponents’ behavior and one that controls attention to different feasible actions. These modifications have sound justifications but lack any obvious underlying principle from which they could be mathematically derived.

*I thank Dilip Mookherjee and Barry Sopher for sharing their data, John Miller and Jon Wilkins for helpful suggestions, and Tanmoy Bhattacharya for technical advice on the estimation methods.
This paper develops a new model — risk learning — that is based on three key ideas. First, in contrast with an implicit assumption of much of the experimental economics literature, utility is not a linear function of reward. Second, choosing between actions in a game is like choosing between lotteries, except that one’s opponents generate the lottery. Because choosing an action is like choosing a lottery, risk attitudes become important when assessing the value of an action, and it then becomes important to pay attention to the curvature of the utility function when modeling value. Third, the model uses an adaptive approach similar to reinforcement learning, though more sophisticated, to construct approximately optimal estimates of each action’s utility.

Since the model incorporates risk and the possibility of risk-averse or risk-seeking behavior, the optimal estimate of expected utility depends at least on the mean payoff and the payoff variance. This observation is significant in part because there is considerable evidence of a neurological basis for learning about expected payoffs as well as some evidence of a neurological basis for learning about variances. Specifically, electrophysiology experiments in non-human primates indicate that the dopamine and possibly serotonin systems respond in proportion to reward prediction errors (see Schultz, Dayan, and Montague 1997; Daw and Touretsky 2002; Daw, Kakade, and Dayan 2002), results that are mirrored in fMRI studies in humans. There is similar evidence regarding reward-variance (i.e., risk) learning (e.g. Tobler, O’Doherty, Dolan, and Schultz 2007). (For a review of these and other issues in neural reward processing, see Rangel, Camerer, and Montague 2008, p.548-550.)

Using data from experiments conducted by Mookherjee and Sopher (1994, 1997), I show that the risk-learning model performs about as well as EWA models. I use a number of approaches — Bayesian information criteria, Vuong tests, out-of-sample predictions, and simulations — to compare the different models. Overall, risk learning performs better than self-tuning EWA and only marginally worse than the more complicated parametric EWA model. Thus, the model succeeds on multiple fronts. It is stronger theoretically than previous models because it can be derived from a clear principle — approximately optimizing a utility function that is itself derived in a principled way — and because it is simpler. It draws inspiration from recent advances regarding the biological
foundations of decision making. And it works about as well as the best models of learning in games.

The rest of this paper is organized as follows. In the next section, I discuss two models of empirical and theoretical significance, simple reinforcement learning and experience-weighted attraction. The latter is important theoretically because its derivation is closely related to risk learning’s derivation. The following section presents the derivation of risk learning. I then present a discussion of the experimental data and a comparison of EWA and this paper’s model using this data. Because I only have access to a relatively small data set, I next present a brief analysis that identifies what sort of games would be useful for future experimental tests of the risk-learning model. I conclude with a summary and a discussion of future theoretical research.

2 Previous Models of Learning in Games

In this section, I examine two earlier models of learning in games, simple reinforcement learning and experience-weighted attraction. Simple reinforcement learning is important theoretically because, while researchers do not often discuss this fact, it may derived as an adaptive approximation to optimizing behavior in which the goal is to estimate the value of one’s feasible actions. EWA is important because of its success in explaining behavior in experiments on repeated games and, while essentially phenomenological, for its insights regarding how people learn and react to new information.

In discussing both reinforcement learning and EWA, I will assume the following notation. A set of players plays a game $G$ in a series of rounds $t \in \{0, 1, 2, \ldots, T\}$. Each player $i$ has a set of strategies $S_i$ with elements $s_{ij}$ with which she associates values $v_{ij}(t)$ at time $t$. (Camerer and Ho (1999) label these $A_j^i(t)$ and call them “attractions,” hence “experience-weighted attraction.”) A player’s strategy at time $t$ is $s_i(t)$, and, following standard notation, the strategy profile is $s(t) = \{s_i(t), s_{-i}(t)\}$. Player $i$’s payoff at time $t$ is $\pi_i(s_i(t), s_{-i}(t))$. At round $t$, player $i$ chooses strategy $j$ with probability

$$P_{ij}(t) = \frac{e^{\beta v_{ij}(t)}}{\sum_{s_{ij} \in S_i} e^{\beta v_{ij}(t)}}.$$  \hspace{1cm} (1)

A natural way to interpret this choice probability, which is common to all the models I will study,
is in terms of random utilities (McFadden 1981) though it has a long history in psychology as well (Luce 1958).

2.1 Simple Reinforcement Learning

While often viewed as a single, extremely simple model of learning, reinforcement learning actually refers to a class of models that are intermediate between supervised and unsupervised learning. Supervised learning may be best thought of as learning that involves — indeed requires — explicit feedback regarding the correctness of an individual’s decisions. The conventional classroom examination is an example of supervised learning. Students make decisions about how to answer a teacher’s questions, and students learn primarily through feedback from the teacher. Furthermore, the feedback is binary: either the response was correct or incorrect. Unsupervised learning is at the opposite extreme, more akin to learning by reading a book outside of class than by taking an exam. Unsupervised learning is learning by observation alone.

Reinforcement learning is in the middle. There is no explicit feedback as to whether a response is correct or incorrect, but there is still feedback in the form of some kind of reward or punishment. That is, a person makes a decision, and following the decision, a reward or punishment arrives, either under the control of an experimenter or through a natural process. After this process repeats many times, a person comes to associate the reward or punishment with the corresponding choice.

In its most basic form — which I will derive in a moment — reinforcement learning is indeed quite unsophisticated, but in general it encompasses a variety of approaches to learning, from the fairly unsophisticated Bush-Mosteller rule (Bush and Mosteller 1955) to temporal-difference (TD) learning and actor-critic learning (see Sutton and Barto 1998; Dayan and Abbott 2001). I therefore refer to the model I present as “simple reinforcement learning,” because while economists tend to view reinforcement learning as unrealistically simple, the full range of reinforcement learning models include some of the most sophisticated approaches available.

A second, often unappreciated fact about reinforcement learning is that one can derive it from an optimization principle. Consider the following problem. A person confronts an $n$-armed ban-
dit without prior knowledge of the payoff distribution for each arm \( a \). Each arm has some true (expected) value \( v_a \), and if a person knew these values he could optimize his actions by selecting \( \arg\max_a v_a \). However, he does not know the true value of each action. Thus, he must estimate the values \( v_a \) and choose a policy that balances exploiting information he already has with enough exploration such that he does not stick with a suboptimal action that appears to be superior based on limited experience. A simply policy that achieves this goal is to select actions with probabilities given by Equation 1. (Actor-critic learning, mentioned earlier, is a generalization of this idea in which players directly optimize their expected payoffs in part by updating their policies.)

The problem we are left with is simply parameter estimation using a series of observations \( r_a(t) \) of the value of each action, and a simple way to estimate the value is to minimize the mean squared error:

\[
E_a = \frac{1}{T_a} \sum_{t=0}^{T_a} (r_a(t) - \hat{v}_a)^2,
\]

where \( \hat{v}_a \) is the estimate of \( v_a \) and \( T_a \) is the number of times the player chose action \( a \), so \( t \) indexes the trials on which the player chose action \( a \). Of course, the optimal estimate is the mean observed reward:

\[
\hat{v}_a = \frac{1}{T_a} \sum_{t=0}^{T_a} r_a(t).
\]

If a player had available a long series of observed payoffs \( r_a(t) \) for each action, then this problem would be very simple. However, the player would be wise to use what information he has at each time step rather than wait many periods before updating his estimates. In this way he can exploit any available information. One way to do this is to write the solution for \( \hat{v}_a \) with \( T_a + 1 \) observations as

\[
\hat{v}_a(T_a) = \frac{1}{T_a} \sum_{t=0}^{T_a-1} r_a(t) + \frac{1}{T_a} r_a(T_a)
\]

\[
= \frac{T_a - 1}{T_a} \hat{v}_a(T_a - 1) + \frac{1}{T_a} r_a(T_a)
\]

\[
= \hat{v}_a(T_a - 1) + \frac{1}{T_a} [r_a(T_a) - \hat{v}_a(T_a - 1)].
\]

This model is known as average reinforcement learning. From the second line, average reinforcement learning can be interpreted as decaying the current value (toward zero) and then reinforcing it with
the most recent observation at a time-dependent rate. From the third line, we can interpret the
approach in terms of prediction error: \( \hat{v}_a(T_a) \) is the player’s prediction of the value of action \( a \) after \( T_a \) observations, and he reinforces this value in proportion to the prediction error, i.e., the
difference between the expected and realized rewards.

In general it is probably not sensible to directly keep track of the number of periods \( T_a \) in which
a player chooses \( a \). For one thing, it is one more thing to keep track of. It also means that the model
is not stationary. More importantly, it implies that players become increasingly unresponsive over
time. Thus, even in nonstationary environments, a player’s estimated value \( \hat{v}_a \) will change very
little if \( T_a \) is large enough.

A way around these problems is to replace \( 1/T_a \) with a fixed learning rate \( \epsilon \):

\[
\hat{v}_a(T_a) = (1 - \epsilon) \hat{v}_a(T_a - 1) + \epsilon r_a(T_a)
\]

\[
= \hat{v}_a(T_a - 1) + \epsilon (r_a(T_a) - \hat{v}_a(T_a - 1)).
\]

In stationary environments, it is easy to see that \( \hat{v}_a \rightarrow v_a \) as long as the player samples each
action \( a \) often enough — something that the policy in Equation (1) guarantees. Equation (2) has
another nice property: it discounts the effect of older rewards, so that it can remain responsive in
nonstationary environments. (An important example of a nonstationary environments is a repeated
game, where \( v_a \) depends in general on others’ actions, which typically change over time.) If the
prediction at \( t = 0 \) is zero, then

\[
\hat{v}_a(T_a) = \sum_{t=0}^{T_a} (1 - \epsilon)^{T_a-t} r_a(t).
\]

Usually, simple reinforcement learning is stated such that only the value of the action a player
chooses updates, although there are variations (direct-actor learning, for example; see Dayan and
Abbott 2001) in which all values are updated.

### 2.2 Experience-Weighted Attraction

At its core, the experience-weighted attraction model is a hybrid of several simpler models, including
simple reinforcement learning as in Equation (2), average reinforcement learning, and belief learning
(e.g. Fudenberg and Levine 1998). Both kinds of reinforcement learning treat each action as simply
having some value that must be learned. Thus it does not explicitly incorporate any strategic information. In particular, it does not learn to anticipate an opponent’s actions — it is not even aware that an opponent exists. Belief learning, on the other hand, takes the payoffs as given and learns instead to anticipate an opponent’s strategy. One can show that belief learning of this kind is the same as reinforcing each action \( j \) with \( \pi_i(s_{ij}, s_{-i}(t)) \), i.e., the realized payoff if the player chose that action and the foregone payoff otherwise.

The original version of experience-weighted attraction, parametric EWA, is just a parameterized hybrid of these various models. The updating rule is

\[
v_{ij}(t) = \frac{\phi N(t-1)v_{ij}(t-1) + [\delta + (1-\delta)1(s_{ij}, s_i(t))]}{N(t-1)\phi(1-\kappa) + 1},
\]

where

\[
N(t) = (1-\kappa)N(t-1) + 1
\]

is the experience weight, \( \pi \) is the payoff matrix, \( 1 \) is an indicator function (one if its arguments are equal, zero otherwise) and \( \phi, \delta, \kappa, \) and \( N(0) \) are parameters. Simple reinforcement learning is the case \( \delta = 1 \) and \( \kappa = 1 \), so that \( N(t) \) is fixed at one, the denominator is one, and foregone payoffs are ignored. As Ho et al. (2007) note, this form of reinforcement learning is not particularly successful at predicting behavior in experimental repeated games. A better form uses average reinforcement \( (\delta = 1 \) and \( \kappa = 0) \) (e.g. Roth and Erev 1995). An even better way is to use information about foregone payoffs, thus incorporating the central features of two very different approaches to learning in games; this is exactly what EWA does.

Parametric EWA was criticized in part for having too many parameters. There are a total of five — \( \phi, \delta, \kappa, \beta \) (for which Ho et al. (2007) use \( \lambda \)), and \( N(0) \), the initial experience weight. (Allowing \( N(0) \) to be a parameter allowed EWA to include, in addition to the models already mentioned, a form of Bayesian learning.) In response to this criticism, EWA’s creators developed self-tuning EWA, which fixed \( N(0) = 1 \) and \( \kappa = 1 \) and replaced \( \delta \) and \( \phi \) with functions of experience. To model the idea that people would respond differently when opponents chose unexpected actions,
Ho et al. set
\[ \phi(t) = 1 - \frac{1}{2} \sum_{k=1}^{m_{-i}} (h_k^i(t) - 1(s-i(t), s_{-i,k}))^2, \]
where \( m_{-i} \) is the number of action profiles for \( i \)'s opponents, \( h_k^i(t) \) is the realized frequency of these profiles up to time \( t \), \( s_{-i}(t) \) is the action profile at time \( t \), excluding \( i \)'s action, and \( s_{-i,k} \) is the \( k \)th feasible action profile. Ho et al. (2007) call \( \phi(t) \) the change-detector function. When opponents take unexpected actions, \( \phi \) is smaller and the model learns more quickly. Likewise, they replace \( \delta \) with the attention function,
\[ \delta_{ij}(t) = \begin{cases} 1 & \text{if } \pi_i(s_i, s_{-i}(t)) \geq \pi_i(s_i(t), s_{-i}(t)) \\ 0 & \text{otherwise.} \end{cases} \]
This definition implies that in addition to the realized action, any action that would have yielded a higher payoff is reinforced, while the remaining actions’ attractions decay. Thus, self-tuning EWA is a one-parameter model; one must only estimate \( \beta \), the parameter that tunes choice probabilities.

Self-tuning EWA is only somewhat less successful at predicting behavior than parametric EWA, and it is in fact better than parametric EWA when performing out-of-sample prediction across games. That is, if one uses a portion of the data for several different games to estimate parameters and then computes the likelihood of the remaining data using these parameters, self-tuning EWA performs better.

Although parametric and self-tuning EWA are empirically successful, they share a theoretical drawback. As hybrids of other, simpler models, it is not clear that one can derive them from any underlying principles. Parametric EWA is literally a parameterized hybrid of several other models. Self-tuning EWA gets rid of these parameters but replaces them with functions that, while they have sensible intuitions, lack theoretical foundation. Introducing these functions also increases EWA’s computational requirements. In particular, one must either keep track of complete action histories or keep track of the number of rounds, and either way one must recompute the action frequencies at each time step. In simple games this may be plausible, but it becomes less so in many of the games Ho et al. (2007) study, some of which have more than 100 pure strategies.

None of this is intended to suggest that EWA is an example of poor modeling. On the con-
trary, phenomenology is in an important stage in the development of good theory, and EWA has
provided valuable insights. The success of EWA indicates that more is going on than simple action
reinforcement; people respond to more than just the average payoff of an action. They also re-
spond to unexpected results: if opponents choose actions they have not often selected in the past,
players learn more quickly. The challenge is to find a way to derive a model in a fashion similar to
reinforcement learning that still incorporates something like this intuition. That is the subject of
the next section.

3 The Risk-Learning Model

The central conceptual insight of this paper is this: when a person chooses an action in a game, she
is really choosing a lottery. The feasible payoffs of the lottery are determined by the feasible payoffs
associated with the action, and the probabilities are determined by the players’ action probabilities.
This is not so different from many modern treatments of game theory. In EWA as well as random
utility models and quantal response equilibrium (McKelvey and Palfrey 1995), players always have
some non-zero probability of choosing a given strategy. However, the present approach deviates
from others by making an explicit distinction between rewards and utilities. If the payoffs in games
are utilities, then in a sense risk is irrelevant — the expected value of an action is just its mean
utility. If, however, the payoffs are rewards — dollars, rupees, ice cream, or whatever — then the
fact that utility is not linear in rewards makes a difference and risk matters. In real situations, of
course, people earn money or some equivalent, and choosing an action in a game involves risk, so
that learning how risky an action is becomes important for assessing its value.

Risk and surprise — one of the things that self-tuning EWA uses — are conceptually similar.
If an opponent takes a surprising action, then one’s own action must be at least somewhat risky.
However, exactly how risky it is depends on how much the opponent’s action affects one’s payoff.
For example, if the surprising action does not affect the payoff taking an action, then the value of
the action should not be significantly affected. If the surprising action does affect the payoff, then
a player should respond to the surprise. EWA has these properties, but learning about risk itself
is a simpler way to incorporate similar properties.

Learning about risk would not be important if utility were a linear function of payoffs. However, it is well-known that utility is not a linear function of payoffs. Perhaps the best known version of this idea is prospect theory (e.g., Kahneman and Tversky 2000), though even classical economics acknowledges the importance of risk aversion in utility calculations. When we incorporate the possibility of a nonlinear utility function, both the expected value of an action and the payoff variance become important. Therefore, thinking in terms of the basic reinforcement learning problem defined above, it is important to estimate both expected value and variance.

3.1 Deriving a New Learning Rule: The Risk Learning Model

The problem we want to solve is as follows. Individuals play a simultaneous-form game, and each individual has some set of actions. We will treat the payoff matrix as unknown to both players. Begin by postulating the existence of a utility function $u(r)$ defined on the real line $\mathbb{R}$. Suppose that $du/dr > 0$ and $d^2u/dr^2 < 0$. Further suppose that each player $i$ treats her payoff from choosing action $j$ as if it is generated by a lottery; that is, there are $L$ possible outcomes for each action with rewards $r_{ijl}$ and probabilities $p_{ijl}$, where $l$ indexes the feasible outcomes for player $i$’s action $j$. Let player $i$’s expected utility from action $j$ be $E[v_{ij}]$. Then,

$$E[v_{ij}] = \sum_{l=1}^{L} p_{ijl} u(r_{ijl}).$$

Now, expand $u(r_{ijl})$ around the mean payoff $\bar{r}_{ij} = \sum_{l=1}^{L} p_{ijl} r_{ijl}$:

$$E[v_{ij}] \approx \sum_{l=1}^{L} p_{ijl} \left[ u(\bar{r}_{ij}) + (r_{ijl} - \bar{r}_{ij}) \frac{du}{dr} \bigg|_{r=\bar{r}_{ij}} + (r_{ijl} - \bar{r}_{ij})^2 \frac{d^2u}{dr^2} \bigg|_{r=\bar{r}_{ij}} \right]$$

$$= \sum_{l=1}^{L} p_{ijl} \left[ u(\bar{r}_{ij}) + (r_{ijl} - \bar{r}_{ij})^2 \frac{d^2u}{dr^2} \bigg|_{r=\bar{r}_{ij}} \right]$$

$$= u(\bar{r}_{ij}) - \kappa \sigma_{ij}^2,$$

where $\sigma_{ij}^2 = \sum_{l=1}^{L} p_{ijl} (r_{ijl} - \bar{r}_{ij})^2$ and $\kappa$ is a positive constant that depends on the curvature or $u$. (This $\kappa$ should not be confused with the $\kappa$ in EWA, which has a very different meaning.) Finally, expand $u(\bar{r}_{ij})$ about $r = 0$ and choose $u(0) = 0$, so that to lowest order

$$E[v_{ij}] \propto \bar{r}_{ij} - \kappa \sigma_{ij}^2,$$

(4)
where $\kappa$ now depends on both the slope and curvature of $u$.

I assume that the learning algorithm will attempt to optimize estimates of $\hat{r}_{ij}$ and $\hat{\sigma}_{ij}^2$, just as simple reinforcement learning optimizes $\hat{v}_{ij}$. Let $\hat{r}_{ij}$ be the estimate of $r_{ij}$ and $\hat{\sigma}_{ij}^2$ be the estimate of $\sigma_{ij}^2$. The optimal estimate of $\hat{r}_{ij}$ is

$$
\frac{1}{T} \sum_{t=0}^{T} 1(s_i(t), s_{ij}) r(t),
$$

and the optimal estimate of $\hat{\sigma}_{ij}^2$ is

$$
\frac{1}{T} \sum_{t=0}^{T} 1(s_i(t), s_{ij}) (r(t) - \hat{r}_{ij})^2.
$$

Here, I have explicitly incorporated a factor $1(s_i(t), s_{ij})$ that selects only the periods when $s_i(t) = s_{ij}$, so that only these periods affect $\hat{r}_{ij}$ and $\hat{\sigma}_{ij}^2$. Then as with simple reinforcement learning we can define incremental updating rules that approximate the optimal estimates:

$$
\hat{r}_{ij} \rightarrow \hat{r}_{ij} + \epsilon (r(t) - \hat{r}_{ij})
$$

$$
\hat{\sigma}_{ij}^2 \rightarrow \hat{\sigma}_{ij}^2 + \epsilon \left[ (r(t) - \hat{r}_{ij})^2 - \hat{\sigma}_{ij}^2 \right].
$$

These rules apply to the realized strategy $s_i(t)$; the predicted reward and value for other actions do not update at $t$.

From Equation (4), the estimate of $E[v_{ij}]$ is $\hat{v}_{ij} = \hat{r}_{ij} - \kappa \hat{\sigma}_{ij}^2$, which I refer to as player $i$’s action value for strategy $s_{ij}$. As above, I suppose that players choose actions with probabilities given by Equation (1). Now, using Equation (5), we can define an action-value update rule:

$$
\hat{v}_{ij} \rightarrow v_{ij} + \epsilon \left[ r(t) - \kappa (r(t) - \hat{r}_{ij})^2 - \hat{v}_{ij} \right].
$$

Note that in this form we must still keep track of two values per action, the action value and the predicted reward $\hat{r}_{ij}$. In summary, the risk-learning update rule is

$$
\hat{r}_{ij} \rightarrow \hat{r}_{ij} + \epsilon (r(t) - \hat{r}_{ij})
$$

$$
\hat{v}_{ij} \rightarrow \hat{v}_{ij} + \epsilon \left[ r(t) - \kappa (r(t) - \hat{r}_{ij})^2 - \hat{v}_{ij} \right],
$$

with choice probabilities

$$
P(s_i(t) = s_{ij}) = \frac{e^{\beta \hat{v}_{ij}}}{\sum_k e^{\beta \hat{v}_{ik}}}.\tag{7}
$$
After absorbing all of this, one may wonder whether one needs to incorporate risk through the parameter $\kappa$ when in the context of random utility models $\beta$ itself parameterizes utility variability. The answer is that $\kappa$ and $\beta$ parameterize different kinds of variability with different consequences. Recall that $\beta$ parameterizes the variability of a utility measurement (McFadden 1981). In contrast, $\kappa$ parameterizes the weight one puts on reward variability — not utility variability — and hence on risk. Only reward variability has an impact on the mean utility of an action. This is another instance in which it is vital to distinguish between an outcome’s reward and its utility.

3.2 Discussion

One of the key features of self-tuning EWA that makes it successful is that in addition to reinforcing actions with their realized payoffs, it responds to outcomes that are surprising. EWA does so by keeping track of opponents’ previous actions and changing its learning rate when opponents take surprising actions. Examining Equation (3), one can see that this choice is most important when the difference between an action’s attraction and its realized payoff is large.

The risk learning model responds similarly to surprising actions, but it does so entirely via payoffs. The important insight is to view choosing between actions when playing a game as choosing between lotteries. Then it becomes important to learn about how risky each lottery is. If an action’s potential payoffs do not vary much as a function of opponents’ actions, then surprise should not figure in very much. Likewise, risk should not figure heavily in the value of the action if the payoffs do not vary much, since in fact there is little risk involved in taking such an action. These are both features of EWA, but the present model incorporates similar intuitions in a simpler and more principled way. Of course, risk and surprise are not the same thing. They do, however, go together. If one is never in the slightest bit surprised, it is unlikely that her actions are risky, and vice-versa.

Finally, there is some evidence that the brain actually processes expected value and variance signals. The results concerning expected value processing are the most clear (e.g., Schultz et al. 1997), but there is also evidence that the brain processes reward variances as well (e.g. Tobler et al. 2007). As Rangel et al. (2008) note, it is difficult to clearly distinguish an expected-value-plus-risk
model from a prospect model that combines the values of individual rewards and their probabilities into a single measure. In either case, however, there appears to be a biological underpinning for learning something about an action’s payoff variability, either directly by learning about its risk or indirectly by learning about the probabilities of individual outcomes.

These considerations are important theoretically, but the model would not be very important if it did not do at least approximately as well as EWA in explaining experimental data. Fortunately, the model does about as well as parametric and self-tuning EWA, particularly in out-of-sample prediction. This holds true even though I correct for the fact the model has more parameters than self-tuning EWA — three (β, ε, and κ) in the present model as opposed to one (β) in self-tuning EWA. I turn to an empirical analysis of the model in the next section.

4 Empirical Analysis

In this section, I use data from Mookherjee and Sopher’s 1994 experiments on learning in the “matching pennies” game and 1997 experiments on learning in constant-sum games1 to estimate the model and compare it with the results of Ho et al. (2007), who used these same data as part of their empirical study. I begin with the constant-sum games, since these are the games that Ho et al. (2007) studied (among others). Although the data from the matching-pennies games have received less attention, they are interesting in part because there are two groups, one which knew only realized payoffs and a second which knew the payoff matrix and could infer opponents’ actions. Studying this data will therefore improve our understanding of the mechanisms that make EWA or risk-learning function better in different environments.

4.1 Constant-Sum Games

The constant-sum games experiments were conducted in India at the Delhi School of Economics. 80 experimental subjects played one of four similar games (20 played each game), each of which involved two players, one designated the row player and one the column player. The players were matched and played a stage game for 40 periods. The players did not know who their opponents

---

1 Again, my thanks to Dilip Mookherjee and Barry Sopher for sharing their data with me.
were. In two of the games (Games 1 and 3), each player had four feasible actions, while in the other
two (Games 2 and 4) each player had six feasible actions. In all games, each outcome led to one
player winning and the other losing, though in some cases a lottery determined which player won.
Winners received a payoff of five rupees in Games 1 and 2 and ten rupees in Games 3 and 4, while
losers received nothing. Table 1 presents the row player’s probability of winning for each outcome.

As Ho et al. (2007) do, I set the initial values $v_{ij}(0)$ using a version of the cognitive hierarchy
framework (Camerer, Ho, and Chong 2003, 2004). The intuition of this approach is that, while
people probably do not necessarily have accurate beliefs about others’ strategies or about others’
beliefs, they probably do take into account the possibility that opponents have beliefs and respond
to these in choosing their actions. To formulate this precisely, suppose there are two players $A$ and
$B$. I assume that $A$ believes that $B$ has beliefs about how $A$ will play on the first round, namely,
that $A$ will randomize uniformly over her actions. Furthermore, $A$ believes that $B$ will best-respond
to these beliefs, and $A$ conditions her choices on this belief.

An example using Mookherjee and Sopher’s Game 1 will help illustrate this idea. Suppose a
player simply randomizes and chooses each action with probability $1/4$. This is $k = 0$ or zero-step
thinker. Now suppose that a player thinks that all others will randomize — i.e., are $k = 0$ players
— and computes the expected utilities of her actions accordingly. This is a $k = 1$ or one-step
thinker. A two-step thinker supposes that there is some mixture of zero and one-step thinkers
and computes expected utilities accordingly. All players have a common belief about the relative
probabilities step-$k$ thinkers, except that each believes that every other player uses fewer steps
than he does. Thus, if the actual distribution is $f(k)$, a step-$L$ thinker believes the distribution is
$g(k) = f(k)/\sum_{l=0}^{L-1} f(l)$ for $0 < k < L$ and $g(k) = 0$ otherwise. A convenient distribution is the
Poisson distribution with mean $\tau$. Empirically, this distribution with $\tau = 1.5$ fits best across games
(Camerer et al. 2003).
In the cognitive hierarchy model, players best-respond to beliefs constructed in this fashion. Self-tuning EWA uses Poisson cognitive hierarchy beliefs with $\tau = 1.5$ and sets the initial attractions for action $i$ at its expected value under these beliefs. I used the same initial attractions for parametric EWA, and I used the same values for both the initial reward predictions $\hat{r}_{ij}(0)$. I then computed the reward variances $\hat{\sigma}_{ij}(0)$ under cognitive hierarchy beliefs and set $\hat{v}_{ij}(0) = \hat{r}_{ij}(0) - \kappa \hat{\sigma}_{ij}(0)$. This choice again reflects the key idea that differentiates risk learning from past work: since utility is not the same as reward, we should take risk attitudes into account. For the choice of initial action values, this means taking predicted reward variance based on cognitive-hierarchy beliefs into account.

To compare the risk-learning model with self-tuning and parametric EWA, I used two different measures. First, I computed Bayesian Information Criteria for each of the models using all of the data. (Ho et al. report BICs using 70 percent of the data; they use the remaining thirty percent to compute out-of-sample predictions. I report similar results below.) The likelihood is

$$L(\beta, \epsilon, \kappa) = \prod_{i=1}^{N} \prod_{t=1}^{T} P(s_i(t)|\beta, \epsilon, \kappa; \{s_i(t-i), r_i(t-i) : 1 \leq i < t\})$$

where $N$ is the number of players and $T$ is the number of periods. Note that the probability of action $s_i(t)$ depends on the realized history of player $i$’s actions and rewards, since this history determines $\hat{r}_{ij}(t)$ and $\hat{v}_{ij}(t)$ and hence the action probabilities. The BIC correction is $-K \log N/2$, where $K$ is the number of parameters.

Second, I computed Vuong statistics (Vuong 1989), again using all of the data. To compute the variances for the Vuong statistics, I treated each player as an observation, so that there were a total of 80 observations. I did not correct the Vuong statistics for the number of parameters. This choice has the greatest (though still small) impact on the comparison between risk learning and parametric EWA: including a BIC correction changes the Vuong statistic from $-1.52$ to $-1.42$. I report the BICs and Vuong statistics in Table 2. Notice that $\kappa$ in the risk-learning model is negative, indicating risk-seeking behavior. I expand on this observation in the discussion below.

[Table 2 about here.]
By the BIC standard, risk learning fares much better than self-tuning EWA and almost as well as parametric EWA. By the Vuong statistics, self-tuning EWA is clearly inferior to both risk learning and parametric EWA. Depending on where one sets the standard of statistical significance, parametric EWA is statistically indistinguishable from or slightly superior to risk learning. Overall, the pattern of results shows that risk learning is superior to self-tuning EWA and nearly as good as parametric EWA as a model of the Mookherjee and Sopher games.

The results above indicate statistically how the different models fare, but they do not provide any intuition about how the models respond in particular situations or to what their relative success should be attributed. For these purposes, a useful approach is to compare the evolution of simulated and experimentally-observed action frequencies. I simulated each model using the parameter estimates reported in Table 2 I pooled data for both row and column players. In Figures 1-4, I present experimental, simulated self-tuning EWA, and simulated risk-learning frequency trajectories for each action and for each game.

From Figures 1-4 and Table 2, it is clear that parametric EWA and risk learning produce qualitatively similar results. By the mean squared error measure, risk learning performs better than self-tuning EWA and about the same as parametric EWA. From the figures, risk-learning and parametric EWA appear to do well by better matching action frequencies in extreme cases, e.g., action 4 in games 1 and 3. Generally speaking, both EWA and risk learning track overall trends in action frequencies fairly well, though neither of them exhibits the rapid fluctuations that the data do. Self-tuning EWA’s primary advantage is that it exhibits precisely this volatility, but this
appears to come at the cost of relatively poor prediction for low-frequency actions (e.g. action 4 in Game 1).

It is worth pointing out that the EWA parameters I estimated differ from those Ho et al. (2007) report, even accounting for the fact that they converted rupees to U.S. dollars (i.e., \( \beta \) has units of inverse currency, so the choice of currency affects its numerical value). Comparing the simulations I performed with those described by Ho et al. and with simulations using their reported parameters, this discrepancy does not seem to affect the parametric EWA simulations very much. For self-tuning EWA, the parameter change does not affect the simulations very much, but the simulations I performed differ from those Ho et al. reported in an earlier version of their paper. (As of this draft, these data are available at http://www.bschool.nus.edu.sg/Staff/bizcjk/fewa.htm under “Mixed Games.”) These latter simulations, however, appear to be based on a very different approach to determining initial EWA attractions, and overall the simulations appear to be somewhat sensitive to initial conditions.

Ho et al. (2007) use a different approach to estimation and prediction, and for the sake of comparison I present similar results here. As I mentioned above, the authors estimated parameters using 70 percent of the experimental subjects. These subjects were selected at random. With the resulting parameters, Ho et al. computed the likelihood of the behavior of the remaining 30 percent of the subjects. Such out-of-sample prediction is one intuitive measure of the usefulness of a model as well as a straightforward way of comparing non-nested models (see Ho et al. 2007, p.178). Because Ho et al. do not report the realized 70 percent sample they used for parameter estimation, I report average BICs for 100 random samples of 70 percent of the experimental subjects. As Ho et al. did, I compute the BIC-adjusted log-likelihood of the remaining 30 percent of the data using the parameters estimated with the first 70 percent. Table 3 presents the results, which are generally very similar to those based on the other measures I used above.

[Table 3 about here.]
4.2 Matching Pennies

The second set of games I study are the matching pennies games from Mookherjee and Sopher (1994). As with the constant-sum games, these experiments were conducted in India at the Delhi School of Economics. A total of 40 experimental subjects played the game in 4, a constant-sum game in which a player either received zero or four rupees at the end of each round. Each pair of players played the game for 40 rounds. The 40 subjects were divided into two groups. Group 1 subjects were told the basic structure of the game — two players with two actions each — but did not know the monetary payoffs and were not told their opponents’ actions. Group 2 subjects, on the other hand, were told the complete payoff matrix. They could, therefore, infer their opponents’ actions with relative ease.

I performed much the same analysis for the matching pennies game as I did for the constant-sum games, except that I separately studied Groups 1 and 2. In Group 1, by the BIC standard the ordering of the models is reversed: self-tuning EWA does best, followed closely by risk learning and then parametric EWA. By the Vuong statistic measure, risk learning does better than either model, although only marginally so. Statistically, the difference is comparable to that between risk learning and parametric EWA in the constant-sum games, except that here risk learning does better than parametric EWA.

Group 2 is a different story entirely. First, all the models do less well than they did in Group 1. Second, by the BIC measure self-tuning EWA does best, followed by risk learning and then parametric EWA. Mookherjee and Sopher (1994) note that Group 2 players appeared to be making choices at random throughout the 40 periods they played. This suggests a possible explanation for why self-tuning EWA performs better in the matching pennies game. As illustrated in the simulations of the constant-sum games, self-tuning EWA is better at capturing a certain amount of instability or fluctuation in choices, while the other two models tend to predict relatively stable choice frequencies that react largely to longer-term trends rather than short-term fluctuations. If
the long-term trend is fairly stable but there are short-term fluctuations, self-tuning EWA ought to do well, while the other two models will have no real advantage, since the average choice is fairly stable.

One caution regarding the Group 2 data and results is in order. Several of the parameter estimates for the EWA models were negative, including $\beta$ for both EWA models as well as $\phi$ in parametric EWA. An investigation of the parametric EWA likelihood function using randomly sampled parameter values suggests that it is quite flat and that in turn the estimated parameters’ variances are very high. These oddities probably result from the random character of the choices made by Group 2 subjects. Furthermore, this random behavior makes a fair amount of sense. If one knows the payoff matrix in such a simple game, it should be fairly obvious that random choices are optimal, in which case one should not expect any of the models to be particularly useful.

[Table 5 about here.]

5 Identifying Discriminating Games

The empirical results above indicate that the risk-learning model is a strong candidate for a general model of learning in games. However, the analysis so far is limited to a somewhat narrow class of games, and it is not clear how well the model will compare to EWA in other contexts. For that reason, I now use simulation techniques to identify games that will be useful for discriminating between risk learning and EWA. I analyze several two-player games: a patent-race game (Rapaport and Amaldoss 2000), the traveller’s dilemma (Capra, Goeree, Gomez, and Holt 1999), several of the $2 \times 2$ games enumerated by Rapaport and Guyer (1978), and, for the sake of comparison with games for which I have data, two of the constant-sum games described above (Mookherjee and Sopher 1997).

In each case, I simulated play for 100 pairs of players over 40 rounds using the risk-learning (RL) model, self-tuning EWA (sEWA), and parametric EWA (pEWA). I set the initial value functions using the cognitive-hierarchy approach described earlier and used parameters estimated from the constant-sum games (see Table 2). This results in three action frequencies $f_{ij}^{RL}(t)$, $f_{ij}^{sEWA}(t)$,
\( f_{ij}^{pEWA}(t) \), where \( i \) labels the player, \( j \) labels \( i \)'s actions, and \( t \) is the round. The goal is to identify games in which these frequency trajectories differ. One useful measure of difference is

\[
D_{AB} = \frac{1}{T} \max_{ij} |f_{ij}^{A}(t) - f_{ij}^{B}(t)|,
\]

(8)

where \( T = 40 \) is the number of rounds, \( A, B \in \{RL, sEWA, pEWA\} \), and \( A \neq B \). The quantity in Equation (8) is similar to the Kolmogorov-Smirnov statistic used for testing whether an empirical distribution could have been generated by some hypothesized distribution. Thus, \( D_{AB} \) is a way of gauging whether, say, simulated risk-learning data could have been generated by self-tuning EWA. Larger values of \( D_{AB} \) indicate that data consistent with model \( A \)'s predictions are less likely to have been generated by model \( B \). Games with larger values of \( D_{AB} \) are therefore more likely to prove useful in discriminating between models.

**The Games**

In the patent race game I describe here, there are two players, one strong and one weak. The strong player begins with 5 resource units and the weak player begins with 4. Each simultaneously invests an integer amount up to their initial endowment. Whichever player invests more earns 10 minus her investment, and whichever invests less earns her initial endowment minus her investment. If the two players invest the same amount, both earn their endowments minus their investments. Both players have several strategies that are (iteratively) dominated. The game has a unique mixed-strategy equilibrium: the strong player invests five 60 percent of the time and one and three each 20 percent of the time, while the weak player invests nothing 60 percent of the time and two and four each 20 percent of the time.

In the traveller’s dilemma, a price-matching game, we imagine two travellers lose identical pieces of luggage, and an airline must decide how much to compensate them. The airline asks the travellers to submit claims \( x_1 \) and \( x_2 \) dollars, where \( x_1 \) and \( x_2 \) are integers in the present version, and will give both travellers \( \min(x_1, x_2) \) plus or minus two dollars — plus two as a bonus to the traveller who makes the smaller claim and minus two as a punishment to other traveller. If \( x_1 = x_2 \), both travellers receive this amount. The game has a unique, pure-strategy equilibrium — both players
propose the smallest feasible claim — from which experimental subjects deviate greatly. Here, I
assume that the feasible claims are integers between 8 and 20, inclusive.

Apart from the games described in the empirical analysis above, the remaining games I consider
are the symmetric games enumerated by Rapaport and Guyer (1978). Suppose that each player has
actions $a$ and $b$; the outcomes are the ordered pairs $(a, a)$, $(a, b)$, $(b, a)$, and $(b, b)$. In Rapaport and
Guyer’s framework, the utilities for each player are 1, 2, 3, and 4, and the symmetric games may
be identified by the 12 different ways Player 1 assigns the utilities to the outcomes. (Actually, the
authors considered ordinal preferences, but for present purposes we require cardinal preferences.)
In six of these games, each player has a dominant strategy, and in the other six no player does. In
Rapaport and Guyer’s enumeration, the dominance-solvable games are Games 3, 5, 6, 7, 9, and 12,
and the remaining symmetric games are Games 60, 61, 63, 66, 68, and 69.

Results

Table 6 presents the results. There are a number of interesting observations to make. First,
according to the $D_{AB}$ measure defined above, Mookherjee and Sopher’s 1997 Game 4 is one of the
better games with which to distinguish the different models: there are substantial differences in the
predictions the three models make for this game. Second, the differences between risk learning and
parametric EWA tend to be smaller than those between self-tuning EWA and the other models,
suggesting — as does the data above — that risk learning captures some of the phenomenological
properties of parametric EWA while remaining a simpler model.

Third, with respect to the Rapaport and Guyer games, those with dominant strategies tend
to discriminate better between models. For example, while $D_{RL,sEWA}$ averages about 0.14 for
dominance-solvable games, it averages 0.11 for non-dominance-solvable games, each of which has
a mixed-strategy equilibrium. The major exceptions are Games 6, 12, and 60. Games 6 and 12,
which are dominance solvable, have the property that switching actions has the minimum possible
impact on payoffs — one reward unit under any circumstances. Game 60, which is not dominance
solvable, has two pure-strategy equilibria and one mixed-strategy equilibrium, a property it shares
with Games 61 and 63. Of these three games, Game 60 is the only one with the property that for each player the payoff variance for either action is the same in mixed-strategy equilibrium. Interestingly, Game 69 also has this property, except that the mixed-strategy equilibrium is the only equilibrium. It is therefore worth pointing out that Game 69’s $D_{AB}$ values are the slightly larger on average than the other games without dominant strategies.

The analysis of the Rapaport and Guyer games suggests that a confluence of several equilibria and equal payoff variances across actions may be a discriminating factor — a result that is somewhat intuitive given the special role that payoff variance has in the risk-learning model. The difference between the two Mookherjee and Sopher games I analyzed also suggests that payoff variance plays a role: the mixed-strategy equilibria in their Games 2 and 4 is the same, but the higher payoffs in Game 4 imply higher payoff variances in equilibrium (or out of it, for that matter). This observation may provide useful guidance for designing future tests of the risk-learning model.

6 Discussion

Experience-weighted attraction is a successful phenomenological model of learning in games, but as a theory of learning in games it is somewhat less satisfying because it has no clear derivation from basic principles and it is quite complicated, especially in its self-tuning form. However, the same features that make it complicated make it empirically successful. It incorporates reinforcement but also responds to beliefs about others’ actions and to surprising actions. The challenge for this paper was to derive a model from simple principles while maintaining at some level the conceptual insights and empirical success of EWA.

The model began with two key observations — first, that a player chooses between lotteries that the player’s opponents generate, and second, that payoffs are not the same as utilities, as models of learning in games often implicitly assume. It then becomes important to study the consequences of risk-aversion, or, more generally, utility functions that are not risk-neutral. A simple way to model a lottery is to approximate its utility with a mean value and a variance that are weighted
according to the curvature and average slope of the utility function. A few approximations then lead to a reinforcement-learning rule that learns both an expected payoff and a payoff variance and incorporates both into a value function. This is an entirely natural thing to do if we treat actions in games as lotteries and payoffs as rewards rather than utilities. The empirical value of the model is born out by comparisons with EWA. Furthermore, there is evidence that the brain processes information related to learning an expected value and its variance. To summarize, the model is elegant and empirically reasonable at both neurological and behavioral levels.

One of the more surprising results is that people appear to be risk seeking (conditional on the risk-learning model being correct, of course). That is, the parameter \( \kappa \), which is a measure of risk aversion, is negative. Exactly why this occurs is not obvious. In the context of prospect theory (e.g. Kahneman and Tversky 2000), if the reference point is set fairly high, then prospect theory’s prediction that people are risk seeking in losses may explain the observation. It remains unclear how the reference point is set and why it would be set high in this case, though recent research may provide insight into that issue (e.g. Koszegi and Rabin 2007).

I conclude by discussing some possible extensions. First, there are extensions motivated by more sophisticated approaches to reinforcement learning. For example, the direct actor method optimizes expected utility assuming the policy in Equation (1). This approach has the potential advantage that all of the action values update at each step, and they do so in a way that may capture similar intuitions as EWA’s attention function. Specifically, if an action’s payoff exceeds the mean expected payoff, then the corresponding action value increases and other action values decrease. If the action’s payoff is smaller than the mean expected payoff, the opposite happens, so that something loosely akin to EWA’s attention function is present in direct-actor learning. A related model, actor-critic learning, may be useful for studying extensive form games. (See Dayan and Abbott 2001, Chapter 9 more for information about these models.)

Second, it may be useful to incorporate in a more detailed way what we know about the shape of the utility function. From prospect theory, we know that the second derivative is not always negative, and we know that computing the value of a prospect requires a reference point with
which to distinguish gains from losses. The first issue may mean that the approximations leading up to Equation (5) are less appropriate, particularly in games in which there are obviously losses. A trickier issue is how to define the reference point that separates gains from losses. Perhaps it should be set at the mean expected reward — something that would be natural in a direct-actor learning model — or perhaps it should be set separately for each action.

Finally, the model does not take into account so-called fictive learning, in which individuals playing a game take into account foregone payoffs, and could potentially benefit from incorporating such information in some way. Experience-weighted attraction takes such information into account, and a recent fMRI study (Lohrenz, McCabe, Camerer, and Montague 2007) suggests that the brain processes fictive-learning signals in a region associated with valuation and choice. The direct-actor model or a sophisticated form of reinforcement learning called Q-learning (see Sutton and Barto 1998; Lohrenz et al. 2007) may provide an appropriate pathway for generalizing the risk-learning model to include fictive learning. Either way, more work is necessary to address these issues in a theoretically sound way.

References


Figure 1: Action frequencies as a function of time for Mookherjee and Sopher's Game 1.
Figure 2: Action frequencies as a function of time for Mookherjee and Sopher’s Game 2.
Figure 3: Action frequencies as a function of time for Mookherjee and Sopher's Game 3.
Figure 4: Action frequencies as a function of time for Mookherjee and Sopher’s Game 4.
Table 1: Row player’s probability of winning in Mookherjee and Sopher (1997).
### Estimation Summary Data: Constant-Sum Games

#### Bayesian Information Criteria

<table>
<thead>
<tr>
<th>Model</th>
<th>Risk Learning</th>
<th>Self-tuning EWA</th>
<th>Parametric EWA</th>
</tr>
</thead>
<tbody>
<tr>
<td>BIC</td>
<td>-4468</td>
<td>-4915</td>
<td>-4444</td>
</tr>
</tbody>
</table>

#### Vuong Statistics

<table>
<thead>
<tr>
<th>Model</th>
<th>Risk Learning</th>
<th>Self-tuning EWA</th>
<th>Parametric EWA</th>
</tr>
</thead>
<tbody>
<tr>
<td>vs. pEWA</td>
<td>-1.90 ($p = 0.971$)</td>
<td>-11.13 ($p &gt; 0.999$)</td>
<td>–</td>
</tr>
<tr>
<td>vs. sEWA</td>
<td>8.96 ($p &lt; 0.001$)</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>

#### Parameter Estimates

<table>
<thead>
<tr>
<th>Model</th>
<th>Risk Learning</th>
<th>Self-tuning EWA</th>
<th>Parametric EWA</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta$</td>
<td>0.619</td>
<td>0.203</td>
<td>0.526</td>
</tr>
<tr>
<td>$\epsilon$</td>
<td>0.055</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>$\kappa_{RL}$</td>
<td>-0.127</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>$\phi$</td>
<td>–</td>
<td>–</td>
<td>0.978</td>
</tr>
<tr>
<td>$\delta$</td>
<td>–</td>
<td>–</td>
<td>0.303</td>
</tr>
<tr>
<td>$\kappa_{pEWA}$</td>
<td>–</td>
<td>–</td>
<td>0.043</td>
</tr>
<tr>
<td>$N(0)$</td>
<td>–</td>
<td>–</td>
<td>18.352</td>
</tr>
</tbody>
</table>

Table 2: Fitted BIC for each of the three models, Vuong statistics comparing the three models, mean-squared error per action in each of the four games, and parameter estimates.
Table 3: Fitted in-sample and out-of-sample Bayesian information criteria for the model of this paper, self-tuning EWA, and parametric EWA. Results for EWA models are those reported in Ho et al. (2007).

<table>
<thead>
<tr>
<th>Model</th>
<th>In-sample BIC</th>
<th>Out-of-sample BIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Risk Learning</td>
<td>-3125</td>
<td>-1354</td>
</tr>
<tr>
<td>Self-tuning EWA</td>
<td>-3206</td>
<td>-1394</td>
</tr>
<tr>
<td>Parametric EWA</td>
<td>-3040</td>
<td>-1342</td>
</tr>
</tbody>
</table>
Table 4: Payoff matrix for the matching pennies game in Mookherjee and Sopher (1997).
**Estimation Summary Data: Matching Pennies**

<table>
<thead>
<tr>
<th>Model</th>
<th>Bayesian Information Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Risk Learning</td>
</tr>
<tr>
<td>Group 1</td>
<td>-544</td>
</tr>
<tr>
<td>Group 2</td>
<td>-564</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model</th>
<th>Vuong Statistics (for Group 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>vs. pEWA</td>
<td>1.31 (p = 0.095)</td>
</tr>
<tr>
<td>vs. sEWA</td>
<td>1.76 (p = 0.039)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model</th>
<th>Parameter Estimates (for Group 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Risk Learning</td>
</tr>
<tr>
<td>(\beta)</td>
<td>0.227</td>
</tr>
<tr>
<td>(\epsilon)</td>
<td>0.564</td>
</tr>
<tr>
<td>(\kappa_{RL})</td>
<td>-0.193</td>
</tr>
<tr>
<td>(\phi)</td>
<td>–</td>
</tr>
<tr>
<td>(\delta)</td>
<td>–</td>
</tr>
<tr>
<td>(\kappa_{PEWA})</td>
<td>–</td>
</tr>
<tr>
<td>(N(0))</td>
<td>–</td>
</tr>
</tbody>
</table>

Table 5: Fitted BIC for each of the three models, Vuong statistics comparing the three models, and parameter estimates for the matching pennies game.
### D-values for 2 Player Games

<table>
<thead>
<tr>
<th>Game</th>
<th>$D_{RL,SEWA}$</th>
<th>$D_{RL,EWA}$</th>
<th>$D_{sEWA,pEWA}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traveller’s Dilemma</td>
<td>0.152</td>
<td>0.101</td>
<td>0.140</td>
</tr>
<tr>
<td>Patent Race</td>
<td>0.124</td>
<td>0.118</td>
<td>0.113</td>
</tr>
<tr>
<td>MS Game 2</td>
<td>0.138</td>
<td>0.100</td>
<td>0.148</td>
</tr>
<tr>
<td>MS Game 4</td>
<td>0.298</td>
<td>0.204</td>
<td>0.392</td>
</tr>
</tbody>
</table>

### Rapaport and Guyer (1978) Games

#### Games with Dominant Strategies

<table>
<thead>
<tr>
<th>Game</th>
<th>$D_{RL,SEWA}$</th>
<th>$D_{RL,EWA}$</th>
<th>$D_{sEWA,pEWA}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Game 3 (4,3,2,1)</td>
<td>0.171</td>
<td>0.119</td>
<td>0.248</td>
</tr>
<tr>
<td>Game 5 (4,3,1,2)</td>
<td>0.209</td>
<td>0.111</td>
<td>0.264</td>
</tr>
<tr>
<td>Game 6 (4,2,3,1)</td>
<td>0.089</td>
<td>0.126</td>
<td>0.159</td>
</tr>
<tr>
<td>Game 7 (3,4,2,1)</td>
<td>0.120</td>
<td>0.121</td>
<td>0.206</td>
</tr>
<tr>
<td>Game 9 (3,4,1,2)</td>
<td>0.163</td>
<td>0.108</td>
<td>0.225</td>
</tr>
<tr>
<td>Game 12 (2,4,1,3)</td>
<td>0.102</td>
<td>0.096</td>
<td>0.131</td>
</tr>
</tbody>
</table>

#### Games without Dominant Strategies

<table>
<thead>
<tr>
<th>Game</th>
<th>$D_{RL,SEWA}$</th>
<th>$D_{RL,EWA}$</th>
<th>$D_{sEWA,pEWA}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Game 60 (4,2,1,3)</td>
<td>0.234</td>
<td>0.110</td>
<td>0.296</td>
</tr>
<tr>
<td>Game 61 (4,1,3,2)</td>
<td>0.087</td>
<td>0.103</td>
<td>0.096</td>
</tr>
<tr>
<td>Game 63 (4,1,2,3)</td>
<td>0.090</td>
<td>0.081</td>
<td>0.074</td>
</tr>
<tr>
<td>Game 66 (3,2,4,1)</td>
<td>0.080</td>
<td>0.083</td>
<td>0.070</td>
</tr>
<tr>
<td>Game 68 (2,3,4,1)</td>
<td>0.086</td>
<td>0.086</td>
<td>0.095</td>
</tr>
<tr>
<td>Game 69 (2,4,3,1)</td>
<td>0.097</td>
<td>0.090</td>
<td>0.131</td>
</tr>
</tbody>
</table>

Table 6: $D_{AB}$ values for several two-player games.