Supplementary Information

Action Selection Model

We formulated a model based on a strategy vector **A**, with elements a_i representing weights associated with actions with i = {1,2,3} indicating rock, paper and scissors respectively. To derive a strategy $\mathbf{A}_t$ we use a general trial-by-trial update of the form:

$$\mathbf{A}_t = (1-\alpha)\mathbf{A}_{t-1} + \beta \mathrm{T} \qquad (1)$$

Where α is a parameter that decays existing evidence in the strategy vector **A**, and β is the weight assigned to new evidence; T is the function used for updating the strategy **A** with new evidence and is given by:

$$\mathrm{T}(\mathbf{S}_t, r_t) = (r_t - \mathbf{A}_t\mathbf{S}_t)\mathbf{S}_t \qquad (2)$$

Where payoff received at time t is r_t and $\mathbf{S}_t$ an indicator vector for what the subject played (for example, $\mathbf{S}_t$ = (0,1,0) indicates the participants played “scissors” at time t).

From equation (1), action selection proceeds by generating a distribution over **A** at time t using the softmax rule so that the participant’s probability of playing i is given by:

$$d_{i,t} = \frac{\exp(\tau a_{i,t})}{\sum_{k=1}^{3} \exp(\tau a_{k,t})} \qquad (3)$$

where $a_{i,t} \in \mathbf{A}$ and τ is the inverse temperature.

This model combines leaky-integrator and temporal difference models[1] so that the participant’s strategy is *updated* based on the difference between the *predicted* outcome (from the strategy **A**) and *actual* outcome obtained on a trial and this new evidence is combined with a “memory” for previous trials. This method *does not* assume that players explicitly use the payoff matrix rationally.

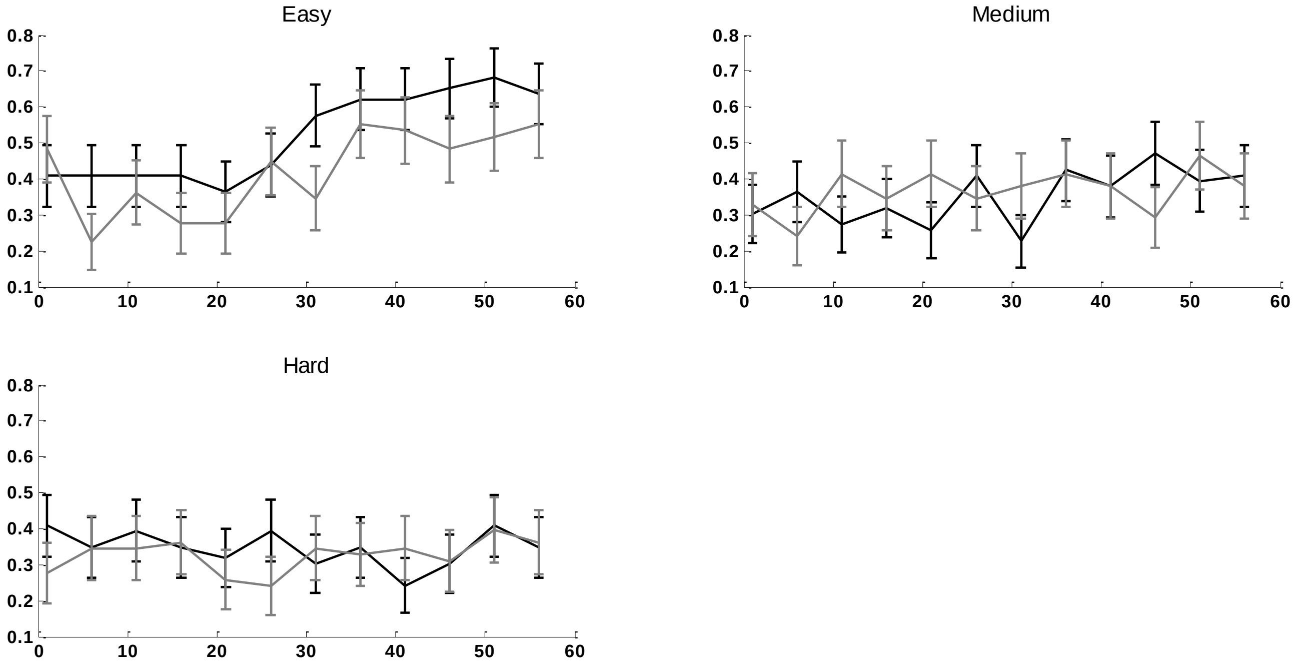
Maximum Likelihood Model Fitting

The model parameters (α, β) were fit by maximizing the likelihood of the data given the model parameters. The log of the likelihood function was computed on a per-game basis as follows:

$$g\left(\mathbf{S}|\alpha,\beta\right)=\sum_{t=1}^{60}\sum_{i=1}^{3}s_{i,t}\log\left(d_{i,t}\right) \qquad (4)$$

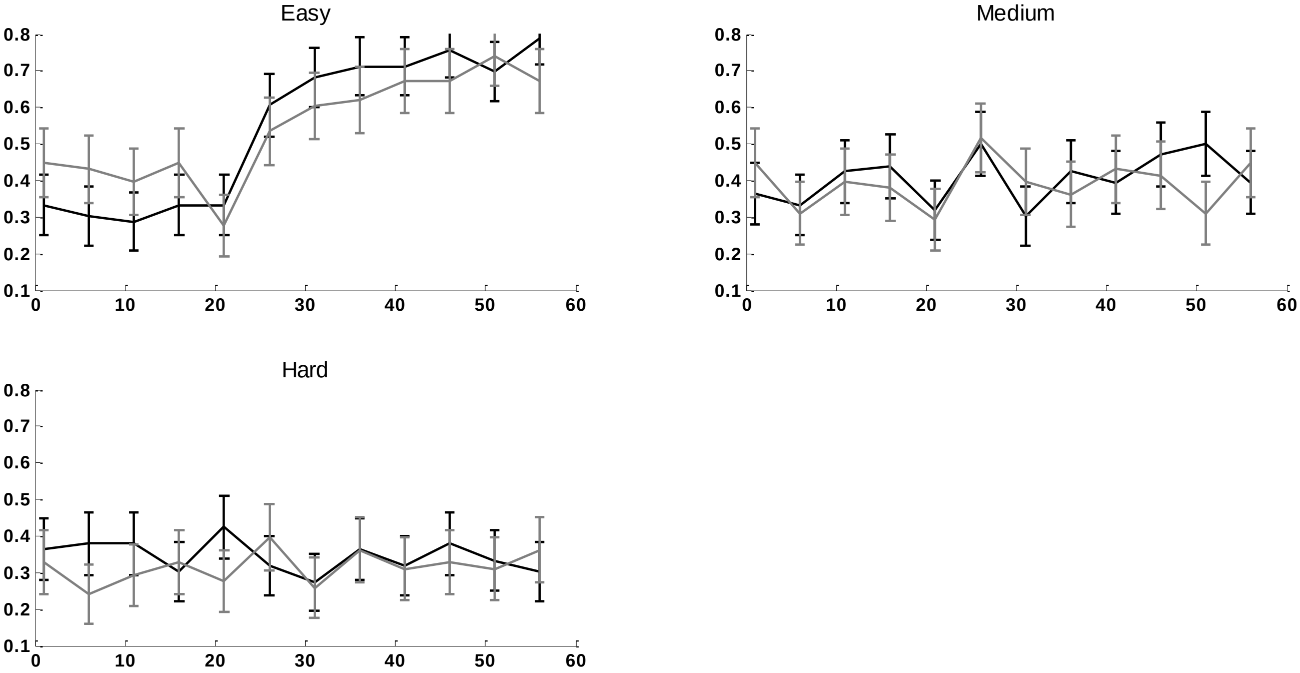
Where $d_{i,t}$, $s_{i,t}$ are the output of the model (equation 3) and the participant's choice on trial t respectively (i.e. the data, **S**). The models were fit using MATLAB's fminsearch implementation of the Nelder-Mead simplex algorithm over g.

The empirical learning behavior of participants is given in **Supplementary Figure 1** below, with the model's predicted performance analyzed similarly in **Supplementary Figure 2**. The model provides a good fit (both quantitatively and qualitatively).



Supplementary Figure 1. Participant's average learning

Plots for the participant's average learning in easy, medium and hard games expressed as probability of choosing the correct play to win on each trial. Controls: Black line, Patients: Grey line. Error bars are +/- 1 standard error. Using the mean of each participant's probability of choosing the correct play over the last 5 trials as an estimate of end-point learning, patients generally learn less well than the controls on easy (t-test, $p < 0.1-6$, controls mean probability correct = 0.705, patients = 0.596) but not in medium (t-test, $p = 0.261$) and hard games (t-test, $p = 0.304$).



Supplementary Figure 2. Model predicted learning

Plots for model predicted learning in easy, medium and hard games, expressed as probability of choosing the correct play to win on each trial. Controls: Black line, Patients:Gray line. Error bars are +/- 1 standard error. The goodness of fit of the model was measured by grouping the log likelihoods for all games (controls, patients), and a one-way ANOVA (controls, patients by average log likelihood) revealed no significant difference between groups, indicating the model fit patients and controls equally.

Model Fit as Correct Prediction of Participant's Play

The model fitting and corresponding goodness of fit measure (equation 4) uses the output distribution given by equation 3 and results in an average log likelihood fit to data (over each group; participants with schizophrenia, and controls).

Alternatively, we present a residual error measure based on the proportion of actual trials where the model made the *same* predicted play as the participant. Such an approach heavily penalizes distributions from equation 3 scoring a 'correct trial' as 1 *if and only if* the model predicts the actual play (rather than giving credit if the model produced a distribution close to the actual participants' play).

As before, let $d_{i,t}$, $s_{i,t}$ be the output of the model (equation 3) and the participant's choice on trial t respectively. We define the mean number of correct trials in T total trials as:

$$M = \frac{1}{T}\sum_{\forall t} c_t$$

$$c_t = \begin{cases} 1 \text{ iff } \operatorname{argmax}\,(d_{i,t}) = s_{i,t} \\ \quad 0 \text{ otherwise} \end{cases}$$

This residual error measure shows for the control group, a mean correct model prediction of 0.595 (SD = 0.17) and 0.544 (SD = 0.18) for patients (where a score of 1.0 would indicate each model correctly predicted every trial of every game).

Confidence / Double Payoff Decisions

A similar approach was taken for modeling the participant's decision to double the payoff. Let x_t be the accumulator favoring the decision to double the payoff matrix, and the parameters η and κ be the weights associated with decaying the previous payoff history and accumulating new payoffs. Then, the update rule for the accumulator is

$$x_t = (1-\eta)x_{t-1} + \kappa P \tag{5}$$

where P is defined as follows for absolute payoff (Figure 2A)

$$P_t^{abs} = r_t \tag{6}$$

and for the "internally derived" prediction error (i.e. the absolute payoff minus the predicted payoff given by each element of **A**; (Figure 2B)

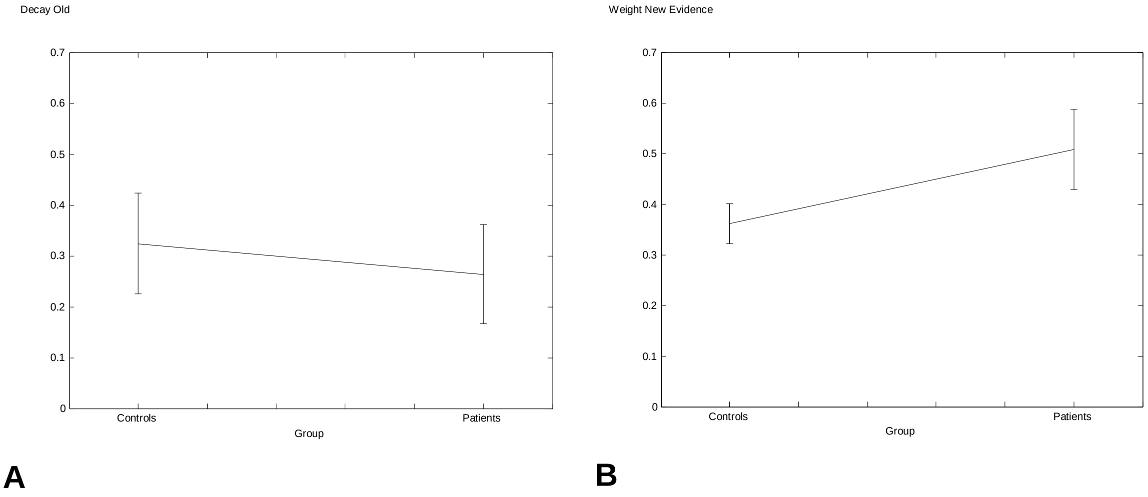
$$P_t^{pred} = (r_t - A_{i,t}) \tag{7}$$

The decision to double payoffs is the trial t_p at which $x_t > \gamma$ where γ is a threshold constant chosen arbitrarily at 0.8.

The model was fit to each game using the Nelder-Mead algorithm (fminsearch function in MATLAB) to minimize a quadratic objective function of the time the participant made the decision to double payoffs, t_d, and that predicted by the model, t_p:

$$O\left(t_d, t_p \middle| \eta, \kappa\right) = \left(t_p - t_d\right)^2 \tag{8}$$

The two parameters governing the decision (η and κ) are shown below in **Supplementary Figure 3**.



Supplementary Figure 3. Decision to double payoffs

Plots for the decision to double payoffs : (**a**) Decay old rewards parameter (η); (**b**) Weight new rewards parameter (κ). Patients mean $\kappa = 0.51$, Controls mean $\kappa = 0.36$. (one tailed t-test, patients > controls; $p<0.0008$)

References

1. Sutton, R., Barto, A. *Reinforcement Learning* (MIT Press, Cambridge, 1998).