Predicting Average Cooperation in Repeated Prisoner's Dilemma: **Online Appendix**

Drew Fudenberg[1] and Gustav Karreskog[2]

[1]Department of Economics, MIT
[2]Department of Economics, Stockholm School of Economics

March 25, 2021

A Additional Depictions of the Data

In the paper we saw that initial round behavior differs between different values of Δ^{RD}, and that the differences increase over time as the participants play more supergames. In figure 1 and 2 we show the corresponding plots but for different memory-1 histories.

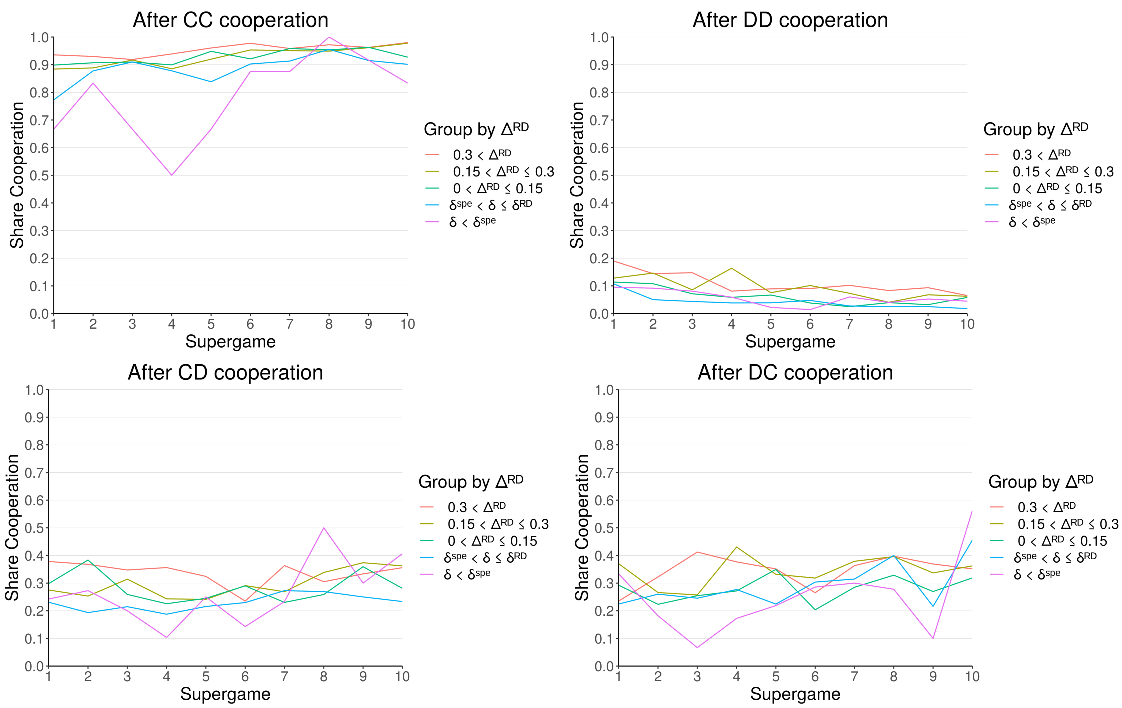

Figure 1: Average cooperation after different memory-1 histories for the first 10 supergames.

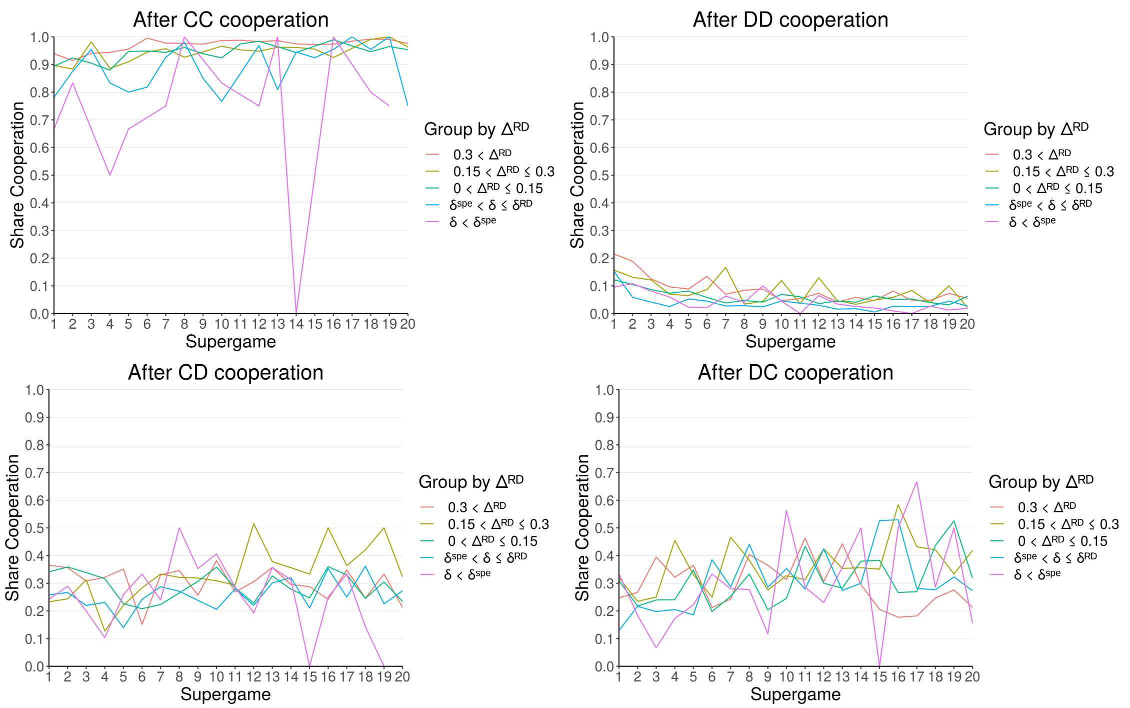

Figure 2: Average cooperation after different memory-1 histories for the first 20 supergames.

Behavior at non-initial memory-1 histories is less variable than behavior at initial histories, both between different values of Δ^{RD} and over the course of the experimental sessions.[1]

If the differences in average cooperation between different treatments is driven primarily by the initial round behavior, then average cooperation after the initial round should be should be primarily determined by the outcome of the initial round, and otherwise similar across treatments. To show this, we compare the following three regressions. The outcome variable is the average cooperation by a participant in a supergame in the rounds following the initial round, e.g., if 4 rounds were played in that particular supergame, we calculate the average cooperation by that participant

[1]In contrast to the initial round, different players face different distributions of the other memory-1 histories, and because these differences are not exogenous there may selection effects. Furthermore, some memory-1 histories are uncommon in certain treatments, e.g. there are few CC in games where the average cooperation rate is low.

in rounds 2, 3 and 4. The first regression, conditions only on the outcome of the initial round. The second adds game parameters (δ, g, l, and Δ^{RD}), and the last uses only the game parameters and not the initial round.

Table 1: Rest of supergame average cooperation conditional on initial round outcome.

	Dependent variable:		
	y		
	(1)	(2)	(3)
initial = CD	-0.635^{***} (0.004)	-0.619^{***} (0.004)	
initial = DC	-0.638^{***} (0.004)	-0.622^{***} (0.004)	
initial = DD	-0.833^{***} (0.004)	-0.788^{***} (0.004)	
g		0.013^{***} (0.005)	-0.011^{*} (0.006)
l		-0.013^{***} (0.003)	-0.039^{***} (0.004)
δ		-0.0001 (0.030)	-0.030 (0.042)
Δ^{RD}		0.170^{***} (0.030)	0.815^{***} (0.041)
Constant	0.906^{***} (0.003)	0.868^{***} (0.013)	0.392^{***} (0.018)
Observations	41,066	41,066	41,066
R^2	0.582	0.586	0.187
Adjusted R^2	0.582	0.586	0.187
Note:			*p<0.1; **p<0.05; ***p<0.01

Most game parameters are statistically significant, but they explain almost no extra variance: The difference in R^2 between the model with and without game parameters is less than 0.01. In contrast, removing the outcome of the initial round lowers the R^2 to 0.0172. This is also true for second- round cooperation instead of average cooperation in the rest of the supergame.

Table 2: Second round average cooperation conditional on initial round outcome.

	Dependent variable:		
	second_y		
	(1)	(2)	(3)
initial = CD	-0.688^{***} (0.005)	-0.677^{***} (0.005)	
initial = DC	-0.662^{***} (0.005)	-0.651^{***} (0.005)	
initial = DD	-0.877^{***} (0.004)	-0.843^{***} (0.005)	
g		0.006 (0.005)	-0.020^{***} (0.007)
l		-0.021^{***} (0.003)	-0.048^{***} (0.005)
δ		0.138^{***} (0.034)	0.106^{**} (0.046)
Δ^{RD}		0.047 (0.033)	0.736^{***} (0.045)
Constant	0.952^{***} (0.003)	0.852^{***} (0.015)	0.342^{***} (0.020)
Observations	41,066	41,066	41,066
R^2	0.551	0.554	0.166
Adjusted R^2	0.551	0.554	0.166
Note:			*p<0.1; **p<0.05; ***p<0.01

B The Pure Strategy Belief Learning Model

Here we outline the belief learning model from Dal Bó and Fréchette (2011), and our across-treatment generalization. Individuals are assumed to choose between TFT or AllD at the beginning of each supergame. The decision is made via a logit best reply based on the individuals beliefs about how likely a partner is to play TFT or AllD, and the implied expected payoffs.

The beliefs are tracked by the two values B_{is}^{C} and B_{is}^{D}, where i is the individual and s is the supergame. Since only two pure strategies are considered, and they prescribe different actions in the initial round of a supergame, the initial-round actions reveal the partner's strategy. The belief values are updated according to

$$B_{is+1}^{a} = \theta B_{is}^{a} + \mathbb{1}\{a_{-i}(s) = a\}$$

where $a_{-i}(s)$ denotes the initial round action taken by the partner of individual i in supergame s, and θ captures recency in the beliefs. Given those two belief values, the belief that the partner will play TFT in supergame s is given by $B^C_{is}/(B^C_{is}+B^D_{is})$.

Let $u^a(\text{TFT}), u^a(\text{AllD})$ denote the expected payoff from taking action a in the initial round if the partner is playing TFT and AllD respectively. Now given the beliefs and those values, the expected value of each choice is given by

$$U^a_{is} = \frac{B^C_{is}}{B^C_{is}+B^D_{is}} u^a(TFT) + \frac{B^D_{is}}{B^C_{is}+B^D_{is}} u^a(AllD) + \lambda_{is}\epsilon^a_{is}$$

where ϵ^a_{is} follows a type I extreme value distribution $\lambda_{is} = \lambda^F_i + (\phi_i)^s \lambda^V_i$. is a sensitivity parameter. This gives the following probability of subject i playing a in the initial round of supergame s, and thereafter following the according pure strategy,

$$p^a_{is} = \frac{\exp\left(\frac{1}{\lambda_{is}} U^a_{is}\right)}{\exp\left(\frac{1}{\lambda_{is}} U^C_{is}\right) + \exp\left(\frac{1}{\lambda_{is}} U^D_{is}\right)}.$$

B.1 Trembles

Here we modify the Dal Bó and Fréchette (2011) learning model by supposing that the individual takes the prescribed action with probability $1-\varepsilon_i$. Otherwise the model remains the same, including using the theoretical values for the value of TFT against TFT etc. We therefore include the model with added trembles as well.

C List of Pure Strategies Considered for Predicting the Next Action

Strategy	Description
Always Cooperate	Alwyas play C.
Always Defect	Always play D.
Tit-for-Tat	Play C unless partner played D the last round.
Tit-for-2-Tats	Play C unless partner played D the last 2 rounds.
Tit-for-3-Tats	Play C unless partner played D the last 3 rounds.
Exploitative Tit-for-Tat	Play D in first round, then play TFT.
2-Tits-for-1-Tat	Play C unless partner played D the last round and punish for 2 rounds.
2-Tits-for-2-Tat	Play C unless partner played D the last 2 rounds and punish for 2 rounds.
Grim	Play C until either player plays D, then defect forever.
Lenient Grim 2	Play C until two consecutive rounds occur in which either player played D, then play D forever.
Lenient Grim 3	Play C until three consecutive rounds occur in which either player played D, then play D forever.

Table 3: List of pure strategies considered for predicting the next action played.

D Evaluation of the procedure on simulated data

To test our estimation approach, we simulate the data using three different models: learning with semi-grim, learning with semi-grim with individual parameters drawn from a normal distribution, and the pure strategy reinforcement learning model.

From each of these models we generate a simulated data set that mimics the data we have. Each session is simulated with an actual sequence of supergame lengths, and with 16 participants in each session. On each of these different data sets we perform the estimations from the main text, and report averages and standard deviations across the 10 different folds. The parameters for each model are taken as the average parameter estimates we got from on the actual data.

Model	Learning with SG	+ Noise	Pure strategy reinf learning
Learning with semi-grim	0.0080	0.0099	0.0081
Standard Deviation	0.0023	0.0054	0.0036
Pure strategy reinf. learning	0.0134	0.0136	0.0031
Standard Deviation	0.0023	0.0054	0.0036

Table 4: Averages and standard deviations of a 10-fold cross validation on different simulated data sets.

When we add noise to the learning model, we draw each individual's parameters from a normal distribution where $\alpha \sim N(-0.313, 0.5)$, $\beta \sim N(1.298, 1)$, $\lambda \sim N(0.196, 0.1)$, $p_{CC} \sim N(0.996, 0.1)$, $p_{CD/DC} \sim N(0.373, 0.1)$ and $p_{CC} \sim N(0.016, 0.1)$. The standard deviations were set ad-hoc to what we thought were reasonable sizes. The means are the estimated parameters from the main analysis. The sampled probabilities are then cut-off to be in the interval $(0, 1)$.

In table 4, we see that our estimation strategy can indeed distinguish between these different learning models.

E One Step Ahead Prediction and Maximum Likelihood Estimation

We here consider the question of how well we can predict the next action taken by a participant given their actions so far. If each participant uses a fixed strategy or learning rule, and the relative shares in the population are known, it should be possible to accurately predict the next action a given individual will take at a given history. Thus, following the literature, we assume that there are a finite number of different "strategic types" (i.e. strategies or learning rules) used in the population, and estimate the parameters and the shares of these strategies by maximum likelihood. We then see how well the different types match the individuals behavior up to that point, and then make the corresponding prediction.

We consider the the pure strategy mixture model. learning with semi-grim, learning with memory-1, and learning at all h, and compare their performance both to a naive benchmark that predicts the previous action taken by the individual, and to the predictions made by a gradient boosting tree.

We focus on out of sample predictions. This allows us to compare models of different complexities, because overly complex models may be penalized by cross-validation.

Several interesting conclusions arise from this exercise. First, in contrast to the problem of predicting the populations behavior, explicitly modeling heterogeneity does improve predictions here. Moreover, as above, learning allows us to make better out of sample predictions. Finally, as in past work we see no evidence of the participants using strategies of memory greater than 1.

E.1 The General Prediction Problem

Consider the complete data set of observations

$$D = \{(h_i(t), a_i(t)) | i \in I, t \in T(i)\},$$

each pair consisting of the history and the action taken for individual $i \in I$, in time period t, where $T(i)$ denotes all the rounds played by individual i, and we track the game parameters Γ_i as part of the history. The action taken $a_i(t)$ is 1 for cooperation and -1 for defection.

A predictive model is a function $m : \mathcal{H} \to [0, 1]$, where $\mathcal{H}$ is the space of all individual histories in an experimental session. This function predicts the probability that an individual with a given history cooperates. A model comes with a set of parameters θ and we write

$$m(h_i(t)|\theta) = \widehat{a}_i(t)$$

to the denote model m's predicted probability of cooperation given history $h_i(t)$.

Two different measures of predictive performance are used, prediction loss and accuracy. The prediction loss is based on the cross-entropy of the predicted probability of the taken action. For a data set $D' \subset D$, the average prediction loss is given by

$$\mathcal{L}(m|D',\theta) = \frac{-1}{|D'|} \sum_{(h_i(t),a_i(t))\in D'} \log(m(h_i(t)|\theta))\cdot\mathbb{1}\{a_i(t)=1\}+\log(1-m(h_i(t)|\theta))\cdot\mathbb{1}\{a_i(t)=-1\}.$$

or if we simplify the notation, by letting m and θ be implicit, with

$$\mathcal{L}(D') = \frac{-1}{|D'|} \sum_{(h_{isr},y_{isr})\in D'} \log(\widehat{y}_{isr})\cdot y_{isr} + \log(1-\widehat{y}_{isr})\cdot(1-y_{isr}).$$

The models are always optimized with respect to the prediction loss, however, it is also interesting to look at the accuracy of the predictions. The accuracy is the share of observations where the taken action was predicted to be the most likely, i.e.

$$Acc(m|D',\theta) = \frac{1}{|D'|} \sum_{(h_i(t),a_i(t))\in D'} \Big(\mathbb{1}\{a_i(t)=1\}\cdot\mathbb{1}\{m(h_i(t)|\theta)\geq 0.5\} \\ + \mathbb{1}\{a_i(t)=-1\}\cdot\mathbb{1}\{m(h_i(t)|\theta)<0.5\}\Big).$$

E.2 Finite Mixture models

When estimating the models, we assume that the population can be divided into different types, where individuals of the same type behave in the same way. Depending on the model, these types are parameterized in different ways. The learning models presented are the same ones used in the main text, with the difference that experience at a given supergame is calculated using the actual observed data upto that supergame, and not from simulation.

Pure Strategy Model In this model we assume that each type σ^j follows a pure strategy with a fixed mistake probability ε_j. If we let $\omega^j : \mathcal{H} \to \{0,1\}$ denote a pure strategy, e.g., Tit for Tat or Grim, a type can be described by a tuple $(\omega^j, \varepsilon_j)$. We start with an exogenous list of 11 different pure strategies, taken from the pure strategies estimated to have positive share in Fudenberg, Rand and Dreber (2012) [2],

[2] While Fudenberg, Rand and Dreber (2012) studies interactions with exogenous noise, these 11 strategies contain those strategies often found to be used in games without noise e.g. Dal Bó and

and estimate the share ϕ_j and mistake probability ε_j and for each such pure strategy. The mistake probabilities ε_j and the shares ϕ_j are explicitly estimated, while the 11 available pure strategies remain fixed. In the standard SFEM approach it is commonly assumed that a common error rate ε is used, we relax this assumption in order to give the pure strategy model a better chance of performing well.

Estimating any finite mixture model gives us a set of types and their relative shares. To make a prediction of $a_i(t)$ based on $h_i(t)$, we first calculate the probability of $h_i(t)$ under the different types. For simplicity, we represent the different types with σ^j for each type j.

$$\Pr(h_i(t)|\sigma^j) = \prod_{\tau<t} \sigma^j(h_i(t))^{\mathbb{1}\{a_i(t)=1\}} \cdot \Big(1 - \sigma^j(h_i(t))\Big)^{\mathbb{1}\{a_i(t)=-1\}}.$$

Given the estimated shares ϕ the conditional probability of individual i being of type j at time t is given by

$$\Pr(\sigma^j|h_i(t)) = \frac{\phi^j \Pr(h_i(t)|\sigma^j)}{\sum_l \phi^l \Pr(h_i(t)|\sigma^l)}.$$

Given these estimated probabilities, the prediction of model m is given by

$$m(h_i(t)) = \sum_j \sigma^j(h_i(t)) \Pr(\sigma^j|h_i(t)).$$

E.3 Evaluating the Models

To evaluate out of sample performance we again use 10-fold cross-validation. Because we are now predicting individual and not aggregate play, here the partitions are at the level of individuals, so that each individual is in exactly one test set.

Furthermore, the splits are balanced over the treatments so that roughly 10% of the participants from each treatment are in each fold.

For each such partition k, we find the parameters θ_k^{train} with the smallest prediction loss on the training set,

Fréchette (2018)

$$\theta_k^{train} = \arg\min_{\theta \in \Theta} \mathcal{L}(m|D_k^{train}, \theta)$$

and calculate the prediction loss on the test set, $\mathcal{L}(m|D_k^{test}, \theta_k^{train})$. The prediction loss from the 10-fold cross-validation that will be reported, and used to compare the models, is given by averaging over all such splits,

$$\text{PredictionLoss}(m|D, K) = \frac{1}{K} \sum_{k=1}^{K} \mathcal{L}(m|D_k^{test}, \theta_k^{train})$$

and the accuracy is similarly given by

$$\text{Accuracy}(m|D, K) = \frac{1}{K} \sum_{k=1}^{K} Acc(m|D_k^{test}, \theta_k^{train}).$$

Depending on the model, we have to estimate it in slightly different ways. There is no canonical why to capture across treatment differences in the pure strategy model the way we do in the memory-1 mixed and learning model. Instead, we follow the literature and make a separate estimation for each treatment. For the memory-1 mixed and learning models however, we estimate one single finite mixture model for all treatments.

E.3.1 Results

In table 5 we see the prediction errors of the different models.

Model	N	AllD	Loss	Accuracy	Relative Accuracy
Naive			0.429	84.6%	
Pure			0.311	88.2%	23.0%
Initial round learning with semi-grim	1		0.321	87.6%	18.9%
	1	Yes	0.307	87.2%	16.8%
	2		0.297	88.0%	22.1%
	3		0.288	88.3%	23.9%
Initial round learning with memory-1	1		0.321	87.6%	18.9%
	2		0.292	88.5%	25.1%
	3		0.282	88.4%	24.5%
Initial round learning with flexible memory-1	1		0.319	87.6%	18.9%
	2		0.293	88.0%	22.0%
	3		0.28	88.4%	24.3%
Full learning	1		0.322	87.5%	18.6%
	2		0.283	88.3%	23.5%
	3		0.28	88.5%	25.2%
GBT with memory-1			0.225	90.8%	40.2%
GBT with memory-3			0.222	90.9%	41.0%

Table 5: Out of sample prediction errors for predicting the next action taken by an individual.

As we see, a single type of the main learning model performs only slightly worse than fitting 11 different pure strategy types on each treatment. Allowing for heterogeneity in the learning model makes it slightly better than the pure strategies. The further improvement from allowing any memory-1 strategy is small, and little is gained by extending to flexible memory-1 behavior that adjusts to Δ^{RD} or extending learning to all h.

E.4 Maximum Likelihoods

Our main analysis of OSAP focus on the prediction errors of the estimated models of the next action taken by individuals, since this allows for straightforward comparisons between models of different complexities. However, since it is more common in the literature to use to consider the likelihoods instead of predictive abilities, we report these likelihoods for completeness. For every history $h_i(t)$ the behavior of type j is captured by a function $\sigma^j : \mathcal{H} \rightarrow [0,1]$ that takes a history and assigns a probability to cooperate. Each model comes with set of parameters. We will go through the different models in the following subsections, but first present the general estimation procedure.

If we let $a_i(t) \in \{-1, 1\}$ denote the action taken by individual i at time t, the likelihood of the observed behavior for participant i if she was of type σ^j with parameters is given by

$$\Pr_i(\sigma^j|\theta_j) = \prod_{t \in T(i)} \sigma^j(h_i(t))^{\mathbb{1}\{a_i(t)=1\}}(1 - \sigma^j(h_i(t)))^{\mathbb{1}\{a_i(t)=-1\}}.$$

Let $\theta = (\theta^j)_{j=1}^J$ denote the parameters of the different types, and let $\phi \in \Delta(J)$ denote their relative share. A is then a pair $m = (\theta, \phi)$, and its likelihood is

$$\mathcal{L}\left(m|\theta, \phi, I\right) = \sum_{i \in I} \log \left(\sum_{j=1}^{J} \phi^j \Pr_i(\sigma^j|\theta_j) \right).$$

The model is then estimated by maximum likelihood.

Our main learning model only has six parameters per type, and these six parameters are the same across treatments. In comparison, the pure strategy model incorporates 11 different pure strategies, each with a different mistake probability, and these are estimated separately for each of the 28 treatments. If we were to directly compare the pure strategy model's loglikelihoods with the initial round learning model's loglikelihoods, we would be comparing a model with 736 parameters and one with 6.

To make the comparison more meaningful, here we consider the models estimated separately on each treatment as well as on the overall data, and we include BIC values to compensate for model complexity.

We consider three versions of each model (except the 11-type pure strategy model): A single type, a single type plus an AllD type, and three types. Since the first period learning model does not include AllD as a subset, we also consider a version with three first period learning types and one AllD.

In table 6 we see the loglikelihoods, estimated using E.4, of the different models, estimated and evaluated on the full supergames, and in table 7 evaluated on the last third of the supergames in each session.

Model	N	AllD	Estimated on	Loglikelihood	BIC
Pure	11	No	Each Treat	-72272	149394
Learning with semi-grim	1	No	Each Treat	-72676	146738
Learning with semi-grim	1	Yes	Each Treat	-68476	138800
Learning with semi-grim	3	No	Each Treat	-65481	135581
Learning with semi-grim	3	Yes	Each Treat	-65688	135995
Learning with semi-grim	1	No	All Treat	-74559	149193
Learning with semi-grim	1	Yes	All Treat	-71814	143727
Learning with semi-grim	3	No	All Treat	-66919	134061
Learning with semi-grim	3	Yes	All Treat	-65030	130307
Learning with memory-1	1	No	Each Treat	-72341	146300
Learning with memory-1	3	No	Each Treat	-63437	131725
Learning with memory-1	1	No	All Treat	-74121	148328
Learning with memory-1	3	No	All Treat	-65080	130420
Learning at all h	1	No	Each Treat	-72068	146677
Learning at all h	3	No	Each Treat	-64129	135879
Learning at all h	1	No	All Treat	-74774	149685
Learning at all h	3	No	All Treat	-64995	130399

Table 6: Maximum likelihood log-likelihoods evaluated on the complete set of supergames.

In the literature, it is common to focus on the latter part of the experiment, under the assumption that behavior then has become more stable.

Model	N	AllD	Estimated on	Loglikelihood	BIC
Pure	11	No	Each Treat	-17142	38427
Learning with semi-grim	1	No	Each Treat	-19442	40068
	1	Yes	Each Treat	-17838	37255
	3	No	Each Treat	-16381	36710
	3	Yes	Each Treat	-15951	35848
	1	No	All Treat	-19804	39675
	1	Yes	All Treat	-20261	40611
	3	No	All Treat	-17343	34889
	3	Yes	All Treat	-16589	33402
Learning with memory-1	1	No	Each Treat	-18916	39213
	3	No	Each Treat	-15746	35636
	1	No	All Treat	-19697	39473
	3	No	All Treat	-16320	32876
Learning at all h	1	No	Each Treat	-18775	39722
	3	No	Each Treat	-15612	37736
	1	No	All Treat	-20005	40134
	3	No	All Treat	-15922	32216

Table 7: Maximum likelihood log-likelihoods evaluated on the last third of the supergames.

As shown in the tables above, both the maximum likelihood results on all supergames and on the last third are consistent with the primary analysis. According to the BIC, the best model is the learning model that extends to all memory-1 histories, while the pure strategies model is one of the worst. We also see that the difference between the model with learning and semi-grim, and the possible extensions, is very small. Furthermore, we achieve relatively good performance with a single learning model that keeps behavior after the initial round constant across treatments and individuals, especially if we include AllD.

We also see that we accurately capture the between treatment variation within our models. The loglikelihood is often similar for the models estimated jointly for all treatment, with logistic functions of Δ^{RD} capturing the variation between treatments, and the ones estimated separately for each treatment. And the lowest BIC is given by such joint estimation.

F Between Session Variance and Δ^{RD}

For intermediate values of Δ^{RD}, our model predicts that small differences in behavior and realized supergame lengths can have large implications for the resulting average cooperation in a session. An intuitive hypothesis is thus that between session variation would be larger for intermediate values of Δ^{RD} than extremer values. We here show some evidence of that relationship.

We consider both the actual between session variation and our model's predicted between session variation. For the actual data we are restricted to amount of data collected, which only consists of 161 sessions. For the simulated data, we have no such restriction. We therefore simulate 10 different populations, with randomly generated supergame lengths, when making the predictions. Since cooperation within a supergame is not independent of the realized length, δ is in itself likely to give rise to some variation. We therefore consider only initial round cooperation in the plots below.

In Figure 3 we see the actual and simulated between session variance for each Δ^{RD}-group. Note that each such group does contain different treatments, so the comparison of variance is not perfect.

In Figure 4 we instead look at the actual and predicted variance in Δ^{RD} for each value of Δ^{RD} for which we have more than 1 observation. For most of these values, we have quite few observations, so the variance is quite poorly identified.

We see in both these figures that the predicted variance is slightly lower than the actual. We also see that for especially for lower values of Δ^{RD}, more extreme values of Δ^{RD} lead to lower between session variance both in the actual data and our simulated data.

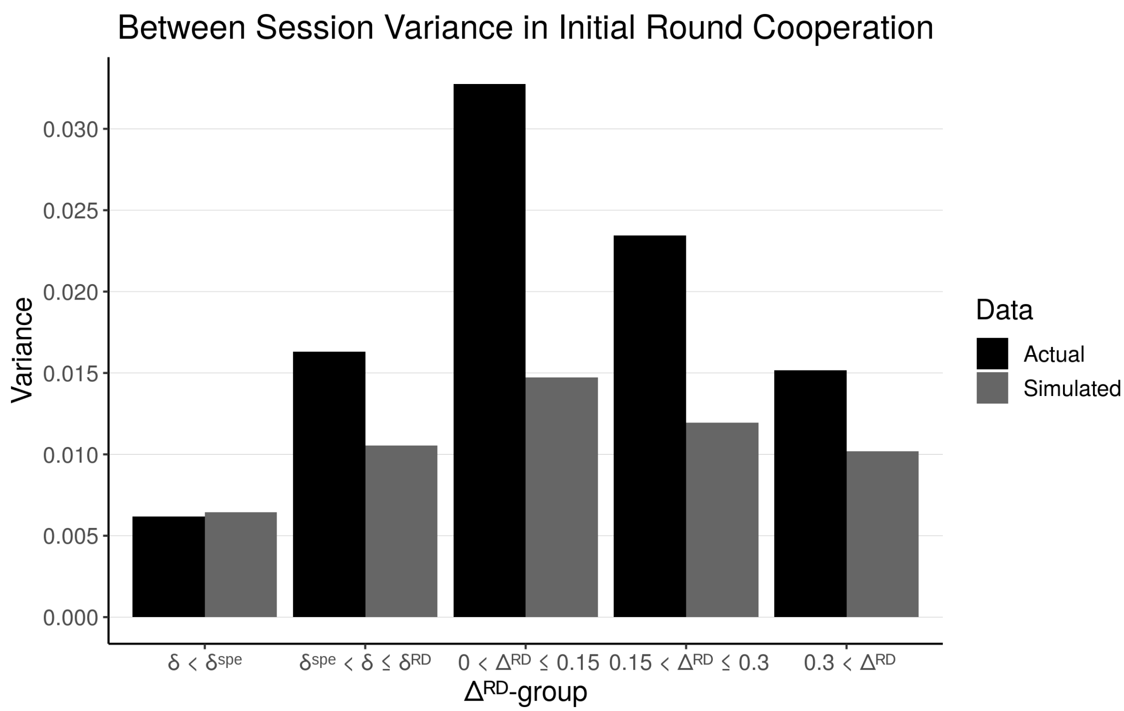

Figure 3: Actual and predicted variance in initial round cooperation for each Δ^{RD}-group

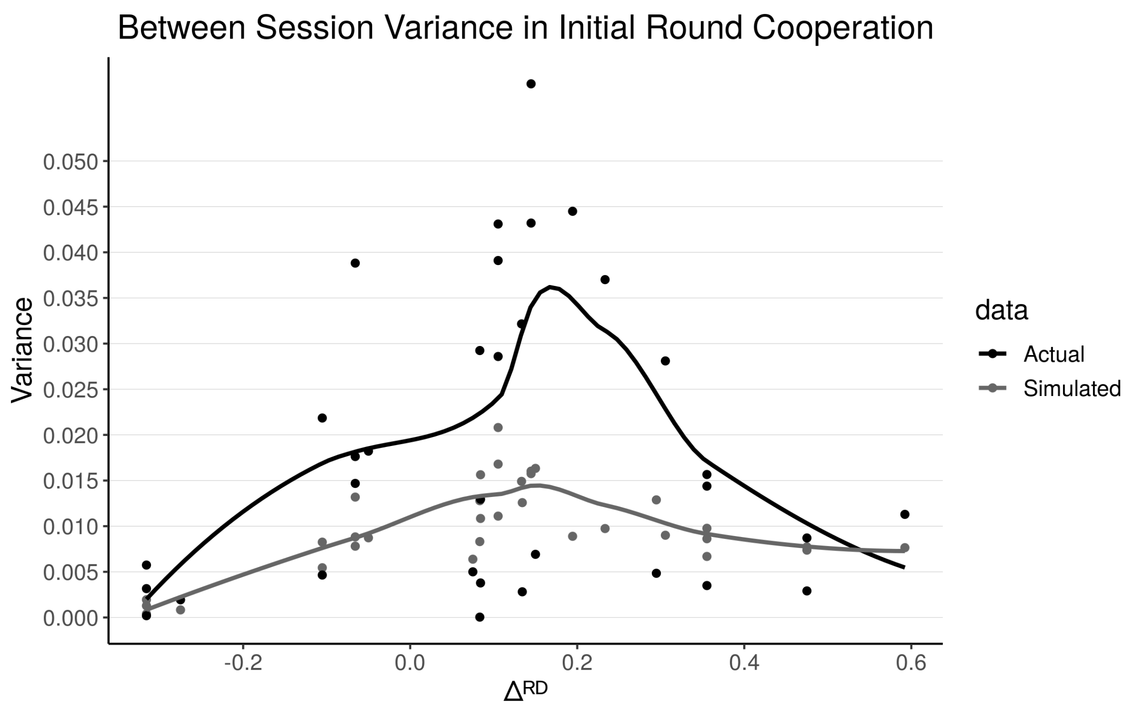

Figure 4: Actual and predicted variance in initial round cooperation for each Δ^{RD}-group

References

Dal Bó, P., and G. R. Fréchette. 2011. “The Evolution of Cooperation in Infinitely Repeated Games: Experimental Evidence.” *American Economic Review*, 101: 411–429.

Dal Bó, P., and G. R. Fréchette. 2018. “On the Determinants of Cooperation in Infinitely Repeated Games: A Survey.” *Journal of Economic Literature*, 56: 60–114.

Fudenberg, D., D. G. Rand, and A. Dreber. 2012. “Slow to Anger and Fast to Forgive: Cooperation in an Uncertain World.” *American Economic Review*, 102: 720–749.