What Drives Cooperation? Extending the Initial-Round Learning with Semi-Grim Strategies Model to Repeated Public Goods Game

Kaiwen Hu

August 2, 2024

Abstract: I explore the application of learning models to infinitely repeated public goods games, expanding upon traditional analyses focused on the prisoner's dilemma. Utilizing the "initial-round learning with semi-Grim strategies" (IRL-SG) model, I examine the impact of game uncertainty and players' past payoffs on cooperative behavior. My findings reveal that as the action set cardinality increases, model performance declines significantly, stabilizing after a critical threshold. Evaluations also demonstrate IRL-SG's superior predictive accuracy compared to model free machine learning algorithms and greater optimism about players' long-term cooperation compared to a traditional learning model.

Keywords: Social Dilemma, Behavioral Game Theory, Repeated Games, Learning

1 Introduction

Recent literature has extensively focused on infinitely repeated games to understand the evolution of strategic behavior across multiple disciplines. Biologists employ evolutionary strategies to study subpopulation growth and decay dynamics, computer scientists utilize stochastic games to develop artificial intelligence capable of defeating top-tier Go and poker players, and economists analyze repeated social dilemmas to investigate the emergence of cooperation. In the field of economics, this research trajectory has led to a significant shift. Scholars have moved beyond the study of finite horizon games, increasingly exploring infinite horizon games in the context of social dilemmas. This transition reflects a growing recognition of the complexities inherent in real-world strategic interactions and the need for more sophisticated models to capture long-term behavioral dynamics.

Despite the advancements, many economic studies investigating adaptation mechanisms in games predominantly focus on the prisoner's dilemma due to its simplicity and the ease of evaluating limited possible states. This preference helps avoid sparsity issues in experimental design by ensuring sufficient observations across states. However, the applicability of learning models, empirically tested on the prisoner's dilemma, to more complex scenarios like the public goods game, remains underexplored. The central question of whether these learning models can be extended to the infinitely repeated public goods game is critical for understanding cooperative behavior in more intricate multi-agent environments.

This paper aims to bridge this gap by extending the application of learning models from the infinitely repeated prisoner's dilemma to the public goods game. Specifically, I investigate whether the "initial-round learning with semi-Grim strategy" (IRL-SG) model, as proposed by Fudenberg and Rehbinder [2024], can be applied to the infinitely repeated public goods game. The study seeks to identify key factors driving cooperation and assess the model's performance in predicting behavior in more complex settings.

Extending the work on the infinitely repeated prisoner's dilemma, I examine the impact of game uncertainty and past payoffs on cooperative behavior in the infinitely repeated public goods game. The economic model used is the IRL-SG model, which has shown that a player's initial action depends on their experiences, with subsequent actions following a first-order Markov chain. My findings demonstrate that the learning structures in IRL-SG are partially applicable to the public goods game, revealing a notable relationship between action set cardinality and model performance. As the number of action set partitions increases, the model's performance initially declines significantly but stabilizes after exceeding five partitions, suggesting a critical threshold in model complexity. The IRL-SG model also predicts unobserved public goods scenarios with moderate accuracy based on "cross-treatment validation scores." Overall, while IRL-SG performs best with lower action range partitions, traditional structural learning models generalize better as the number of partitions grows.

Methodologically, I start by obtaining the maximum likelihood estimates of the model using experimental data from a public goods game with a 26 cardinality action set. The data is divided into training and test sets to evaluate the model's predictive power on out-of-sample observations, considering model complexity. To test generalization across different game structures, I apply the estimated model to an 11 cardinality action set public goods game. This cross-treatment evaluation provides insights into the learning model's effectiveness in capturing decision-making processes across varying game scenarios, even with changing action spaces. Finally, simulations are conducted to examine the model's long-term behavior in hypothetical scenarios, testing its robustness and predictive capabilities beyond observed data.

This paper is structured to first review relevant literature on predicting actions in games, then present structural learning models, and finally, empirically evaluate the model using publicly available datasets from Lugovskyy et al. [2017] and Mengel and Peeters [2011]. My research contributes to the field by extending recent learning models to the more complex public goods game, potentially enhancing understanding of cooperation dynamics in diverse social dilemmas.

1.1 Related Literature

In any infinitely repeated scenario, a key determining factor of any outcome is the discount factor δ. For example, Dal Bó and Fréchette [2018] conduct a survey analysis on parameters that determine cooperation from multiple different experimental observations of repeated prisoner's dilemma, showing that the emergence of long run actions are much more predictable when there is a long run equilibrium, which is usually a function of δ. Schaefer [2023] analytically derives the determinant of emergence, also as a function of δ, of cooperation using replicator dynamics from Evolutionary Game Theory (EGT) across time of infinitely repeated prisoner's dilemma, and validates the analytical form using the same dataset as the former survey. δ plays a key determining factor in the player's decision in my economic model. The main idea draws from Blonski et al. [2011], where the uncertainty of an infinitely repeated social dilemma game can be parameterized as $\Delta^{RD} \equiv \delta - \delta^{RD} = \delta - \frac{g+l}{1+g+l}$.[1] g and l correspond to the standardized payoff matrix of a prisoner's dilemma laid out in Figure 4. This will be further discussed in Section 3.

Additionally, the intersection of machine learning algorithms and games has been a popular research topic in recent years. Typically, the intersections can be divided into two paradigms. One kind is trying to predict player's behavior through supervised learning. Wright and Leyton-Brown [2017] studied the distribution of player actions on games with arbitrary parameters, but this was limited to one-time frame games, where they develop a machine learning model that takes in game parameters as inputs and outputs the player's action distribution for any arbitrary games. Similarly, Fudenberg and Liang [2019] studied player initial play using machine learning for arbitrary matrix games. On the contrary will I not go as in depth as such literature in machine learning algorithms, but use vanilla

[1] Standard arguments show that players cooperate every round if and only if the game is a subgame-perfect equilibrium (SPE), where players use the Grim strategy: they cooperate in the first round and continue to cooperate only if no one has defected in the previous round. This condition is satisfied if and only if $\delta \geq \frac{g}{1+g} = \frac{1-c/N}{c(N-1)/N}$. Applications of repeated games often assume that players will cooperate whenever cooperation can be supported by an equilibrium (e.g., Rotemberg and Saloner [1986], Athey and Bagwell [2001] and Harrington [2017]). However, Fudenberg and Rehbinder [2024] argue that this hypothesis has little experimental support, so I stick with using Δ^{RD} in my analysis.

algorithms as benchmarks to compare against the proposed structural model.

Another kind of literature that studies how algorithms can learn to play games is using some extensions of reinforcement learning to study how agents learn in games (e.g., Calvano et al. [2020], Banchio and Mantegazza [2023], Possnig [2023], Deng et al. [2024]). However, I will stick to the simplest form of Reinforcement Learning as a counterfactual measure against IRL-SG, since Erev and Roth [1998] and Mertikopoulos and Sandholm [2016] show that it is a sufficient model to fit observations from experimental repeated games in infinite settings.

2 Data

Table 1 summarizes the dataset I use to empirically test the learning models, which are observations from infinitely repeated public goods game experiments conducted by Lugovskyy et al. [2017] and Mengel and Peeters [2011]. Lugovskyy et al. [2017] conduct their experiments over three independent sessions, each with different participants, while Mengel and Peeters [2011] conduct a single session. On a further note, Mengel and Peeters [2011] only has one sequence of infinitely repeated public goods game, but Lugovskyy et al. [2017] run multiple sequences. That is, for a treatment group with discount factor δ, participants play infinite rounds of the public goods game, where the sequence continues with probability δ. Once the sequences end the participants are randomly regrouped to form new groups and play another infinitely repeated sequence.[2] I will call the sequence of any infinitely repeated game a "super game".

The payoff function of the public goods game for player i is

$$u(a_i, a_{-i}) = \rho - a_i + \frac{c(a_i + \sum_{-i} a_{-i})}{N},$$

[2] Table 1 is the summary statistics of the observations that matched observations with probabilistic termination of δ, more than 2 action set cardinality, and with random group rematching only after the super game ends. Lugovskyy et al. [2017] also conducts experiment on finitely repeated public goods game, inifinitely repeated and finitely repeated prisoner's dilemma, and Mengel and Peeters [2011] conducts experiment where a super game consists of random regroup every round.

Table 1. Public Good Game Parameters from the Dataset

Study	$\boldsymbol{\delta}$	$\boldsymbol{c}$	$\boldsymbol{N}$	$\boldsymbol{\lvert\mathcal{A}\rvert}$	**Sessions**	**Super Games**	**Observations**
Lugovskyy et al.	0.8	2.4	4	26	3	15	$3,648$
	0.8	1.2	4	26	3	15	$3,968$
	0.8	1.2	2	26	3	15	$3,272$
Mengel and Peeters	0.9	2	4	11	1	1	336
Total							$11,224$

where ρ represents the initial endowment, a_i is player i's contribution, c is the return rate, and N is the number of players in the public goods game. The realized payoff of each round is the average contribution of all players multiplied by the return rate c and the left over endowment that each players have remaining, where $\frac{c}{N}$ is referred to as the marginal per capita return (MPCR) of the public goods game. For simplicity, both experiments use discrete action sets. Therefore, the action set for all players is $\mathcal{A} = \{0, 1, \ldots, \rho - 1, \rho\}$ with the cardinality $|\mathcal{A}| = \rho + 1$.

To draw an analogy with prisoner's dilemma, consider the extreme case where player i choose to either contribute all the endowment, ρ, or nothing, 0, with all other players following suit. The standardized payoff matrix is illustrated in Figure 5. In this scenario $1 + g = \frac{c - c/N}{c-1}$ and $-l = \frac{c/N - 1}{c-1}$. Consequently, the game uncertainly parameter becomes

$$\Delta^{RD} \equiv \delta - \delta^{RD} = \delta - \frac{\frac{2}{N}(N-c)}{1 + \frac{c}{N}(N-2)}. \tag{1}$$

This formulation highlights the strategic similarities between the public goods game and the prisoner's dilemma, while accounting for the multi-player nature and higher cardinality action space of the former. To make the observations comparable between different game parameters I will use the standardized payoffs and the standardized actions for the remainder of this paper.

3 Model

Before describing the model to be explored in infinitely repeated public goods game, it is crucial to first go through IRL-SG in the infinitely repeated prisoner's dilemma setting. The model is structured around two underlying principles. In an initial round of a super game, the player's strategy will depend on all past experiences. In non-initial round of a super game, the player's strategy will depend on the outcome of the previous round, which is referred to as a Memory One Strategy. Although this assumption, suggesting that infinitely repeated games follow a first order Markov chain, may seem quite strong, Fudenberg et al. [2012] supports this with finding that in a fully controlled lab environment players follow a Memory One Strategy.[3]

IRL-SG predicts that the probability of cooperation in the initial round for a player i in super game s of a repeated prisoner's dilemma as

$$\begin{aligned} P(a_i = C \mid s) &= \frac{1}{1 + \exp(-(\alpha + \beta \Delta^{\mathrm{RD}} + e_i(s)))} \\ e_i(s) &= \lambda a_i(s-1) V_i(s-1) + e_i(s-1), \end{aligned} \tag{2}$$

where the coefficients mean the following.

- $a_i(s)$ is the initial action for super game s. $a_i(s) = 1$ if the initial action in super game s was to cooperate. $a_i(s) = -1$ if otherwise.
- $V_i(s)$ is the total payoff in super game s.
- λ is the learning rate.

For the very first super game $s = 1$, the player will not have any experience so $e(1) = 0$. Learning rate λ is a crucial parameter, determining how strongly past experiences influence future decisions. A higher value of λ indicates that players give more weight to their past

[3] In natural setting or imperfect monitoring, players could follow strategies with longer memories Fudenberg et al. [2012].

experiences when making choices in subsequent games. An intuitive way to interpret the model is how past payoffs influence a player's expected utility for a given action. Imagine a player who chose to cooperate in the initial round of a previous super game. If that initial action of cooperating led to high total payoffs in the sequence, the player will likely view cooperating in the first round of the next super game more favorably. This is because their prior belief, i.e., expected utility, about the benefits of cooperation has been positively updated based on past experience.

Additionally, the model assumes that players follow a "semi-Grim" strategy in non-initial rounds, as proposed by Breitmoser [2015]. This strategy implies that the probability of cooperating, denoted as σ_h given the state in the preceding round h (for "history"), follows order $\sigma_{CC} > \sigma_{CD} = \sigma_{DC} > \sigma_{DD}$. Note that the state set $\mathcal{H}$ for prisoner's dilemma is $\{\varnothing, CC, CD, DC, DD\}$, where $\varnothing$ denotes the initial round of the super game. The parameter of interest in the model are scalers α, β, λ and vector $\boldsymbol{\sigma}$.

3.1 IRL-SG in Public Goods Game

The idea of extending IRL-SG in the context of infinitely repeated public goods game is relatively straight forward. Similar to prisoner's dilemma, the public goods game is also a social dilemma problem. However, the two main differences from the prisoner's dilemma is that it is typically analyzed with more than two players in mind and there are multiple possible actions compared to the binary action set of cooperate and defect. The introduction to higher cardinality in the action set of the public goods game compared to prisoner's dilemma implies that the player's strategy would follow a Categorical distribution rather than a Bernoulli distribution.

3.1.1 Initial Round

Given super game s, player i will choose action $a_i = k$ with the following probability.[4]

$$\begin{gathered}P(a_i = k \mid s) = \frac{\exp(\alpha_k + \beta_k \Delta^{RD} + e_{ik}(s))}{\sum_{k'} \exp(\alpha_{k'} + \beta_{k'} \Delta^{RD} + e_{ik'}(s))} \\ e_{ik}(s) = \lambda 1(a_i(s-1) = k) V_i(s-1) + e_{ik}(s-1)\end{gathered}$$

However, computing all probabilities for $a_i = k$ implies the observation must include enough samples of each possible actions at the initial round of a super game to avoid sparsity issues. Thus, the action set can be partitioned into K discrete sets and instead look at the probability a_i falls within a specific range instead where $\mathcal{A}_k$ denotes the kth range.

$$\begin{gathered}P(a_i \in \mathcal{A}_k \mid s) = \frac{\exp(\alpha_k + \beta_k \Delta^{RD} + e_{ik}(s))}{\sum_{k'} \exp(\alpha_{k'} + \beta_{k'} \Delta^{RD} + e_{ik'}(s))} \\ e_{ik}(s) = \lambda 1(a_i(s-1) \in \mathcal{A}_k) V_i(s-1) + e_{ik}(s-1)\end{gathered} \tag{3}$$

$e_{ik}(s)$ can be interpreted as cumulative payoffs for player i when the initial action was in $\mathcal{A}_k$ up until super game s, discounted by learning rate λ. Compared to the prisoner's dilemma scenario, the parameter of estimates for the initial rounds are vectors $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, and scalar λ.

3.1.2 Non-Initial Rounds

Recall that in any non-initial rounds of a super game players follow Memory One Strategy (i.e., the game is a first order Markov chain). In the dataset there are up to $N = 4$ players and $\rho = 25$, so the possible states would be $|\mathcal{A}|^4 = 456976$. This added complexity introduces potential sparsity issues, as a sufficient number of observations for each state is required to obtain unbiased inferences about the Markov chain. Additionally, this suggests I need to estimate different parameters for each state, introducing overfitting issues. Instead of looking at all possible states, I will define the states as the average relative contribution and

[4] The action set will be discrete because the experimental datasets has discrete choices. However, extending this model to a continuous action set can be done trivially.

discretize this into $[0, 0.25), [0.25, 0.5), [0.5, 0.75), [0.75, 1]$.[5]

Therefore, the set of all possible states $\mathcal{H}$ has $H < |\mathcal{A}|^N$ states. For example, if the model has $H = 4$ states then and I can denote this as $\mathcal{H}_1 = [0, 0.25), \mathcal{H}_2 = [0.25, 0.5), \mathcal{H}_3 = [0.5, 0.75), \mathcal{H}_4 = [0.75, 1]$. Conditional on the previous average contribution $h \in \mathcal{H}$, player i will choose action $a_i = k$ with

$$P(a_i = k \mid h) = \sigma_{hk}$$
$$\sum_k \sigma_{hk} = 1, \forall h.$$

Similar with the initial round probability, the model can be generalized such that the probability that the contribution a_i falls within a specific range $\mathcal{A}_k$ instead of a specific value. The parameter of estimate for non-initial rounds become a $H \times (K - 1)$ matrix $\boldsymbol{\sigma}$.

3.2 Reinforcement Learning

IRL-SG assumes that there is no learning in non-initial rounds of a super game. Consider the case where the players strategy changes over every round. That is every round t is a Markov Decision Process where player i will choose action a_i according to all the experiences up until that round

$$k^* \in \arg\max_{k \in \mathcal{A}} \{\alpha_k + \beta_k \Delta^{RD} + e_{ik}(t)\}$$
$$e_{ik}(t) = \lambda 1(a_i(t-1) = k) V_i(t-1) + e_{ik}(t-1).$$

Given this formulation, the probability that the player chooses $a_i = k$ at round t is given by the soft-max function.

$$P(a_i = k \mid t) = \frac{\exp(\alpha_k + \beta_k \Delta^{RD} + e_{ik}(t))}{\sum_{k'} \exp(\alpha_{k'} + \beta_{k'} \Delta^{RD} + e_{ik'}(t))}. \tag{4}$$

[5] I also look at IRL-SG with the average relative contribution discretized into 5 and 10 partitions.

Therefore, compared to IRL-SG, the players learn based on each round's payoffs rather than the each super game's cumulative payoffs. This results in the same parameters to be estimated as IRL-SG, except excluding $\boldsymbol{\sigma}$. The same sparsity issue may arise if the goal is to compute all probabilities for $a_i = k$, so I will partition the action set into K discrete sets just like with IRL-SG.

4 Estimation and Identification

For all models let the parameter of interest be packed in $\boldsymbol{\theta}$. Consider the estimation problem to be the following: the set of observations on the experimental sessions are

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

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.

The parameter of estimate is obtained by the maximum likelihood estimation

$$\hat{\boldsymbol{\theta}} \in \underset{\boldsymbol{\theta} \in \boldsymbol{\Theta}}{\arg\min} \{-\log P(\mathcal{D}' \mid \boldsymbol{\theta})\} \tag{5}$$

where $\mathcal{D}' \subset \mathcal{D}$ is the training set.[6] More detail on the estimation method is referenced in Appendix B.

The main evaluation metric for the models will be the respective out of sample loss function. That is, after obtaining $\hat{\boldsymbol{\theta}}$ from training samples $\mathcal{D}'$, the models can be evaluated by using the objective functions that were used for estimating the parameters. For example, OLS on Δ^{RD} would be evaluated using the mean squared error and IRL-SG would be the

[6] As the number of parameters increase with K the optimization problem becomes computationally expensive. Therefore, the negative log likelihood will be minimized with Stochastic Gradient Descent for scalability Kingma and Ba [2017].

cross entropy.[7] Given some out of sample observations $\tilde{\mathcal{D}} \subset \mathcal{D}$ where $\mathcal{D}' \cap \tilde{\mathcal{D}} = \varnothing$, the output of the objective function would be the metric used to compare models. This allows for least squares based models like OLS and Lasso to be comparable with multinomial based models like black box machine learning model such as Multi Layered Perception Classifiers (MLP) and Gradient Boosted Trees (GBT).[8] A complementary evaluation metric would be to look at the accuracy of the models. However, this would only be an appropriate metric for classification models, so models with continuous outputs such as OLS and Lasso will not be included in the comparison when I use accuracy.[9] Regarding the machine learning models such as Lasso, MLP, and GBT, I include variables such as subgame perfect equilibrium (SPE), defined as $\delta^{SP} = \frac{g}{1+g}$, the conditions $\delta \geq \delta^{RD}$ and $\delta \geq \delta^{SP}$, and the difference between expected and realized super game lengths, in addition to the variables required to estimate IRL-SG and Reinforcement Learning.[10]

5 Results

5.1 In-Treatment Analysis

First, my analysis will only look at observations from Lugovskyy et al. [2017], which had a bigger cardinality for the participants action set. The main objective is to show the trade

[7] IRL-SG model and Reinforcement Learning follow the same form as multinomial logistic regression, so the negative log likelihood in Equation 5 is the same as the cross entropy function.

[8] Comparing the fit of least squares models like OLS and Lasso against a multinomial logit model is challenging. There's no straightforward way to make this comparison, particularly when the categorical outcome in the multinomial model can be expressed as a continuous variable. In the public goods game scenario, if I choose the dependent variable of the least squares model to be the standardized contribution (i.e., the percentage of endowment contributed), the loss value (e.g., mean squared error) will remain constant regardless of how many categories (K) I use in the multinomial model. In other words, for a model with K partitions, where the dependent variable in the multinomial model represents the order of the partition, the OLS model's performance metric won't change. For example, if the multinomial model has 10 partitions, with the dependent variable taking discrete values from 1 to 10, the OLS model treating this as a continuous variable from 0 to 1 would yield the same MSE, making direct comparison problematic. Therefore, to make things comparable, I will make the dependent variable take values from 1 to K for least squares model as well.

[9] Details on how I evaluate the models are referenced in Appendix C.

[10] Mengel et al. [2022] argue that the realized length in the first third of the super games has an oversized impact on cooperation in later super games.

off between the action set partition size K and the model performances. The parameters are estimated on observations from sessions 1 and 2, and the model will be evaluated on observations from session 3.

5.1.1 In-Treatment Parameter Estimate

The parameter estimates for IRL-SG are presented in Table 2, when the action state is partitioned to 2 ranges (i.e., whether player contributed above or below 50 percent of the endowment ρ), and the history state is partitioned into 4 ranges are presented in Table 2 column 1.

To interpret the parameters it's important to recall that from Equation 2, the probability for player i to contribute more than half the endowment is given by the sigmoid function $\frac{1}{1+\exp(-(\alpha+\beta\Delta^{RD}+e_i(s))}$, where $e_i(s) = \lambda a_i(s-1)V_i(s-1) + e_i(s-1)$ represents the experience up to the sth super game, with $e(1) = 0$ for the initial super game.

The estimated $\hat{\alpha} = -0.454$ implies that when $\Delta^{RD} = 0$ and there is no prior experience $s = 1$, there is approximately a 38.8 percent likelihood that an average player will contribute more than 50 percent of the endowment in the initial round of the first public goods game. This suggests that without any prior experience of playing the public goods game, players are relative more reluctant to cooperate.

The estimated learning rate $\hat{\lambda} = 0.046$ is relatively small. To illustrate the impact of this learning rate, consider the public goods game with $N = 2$ players, return rate $c = 1.2$ and discount rate $\delta = 0.8$, resulting in $\Delta^{RD} = 0$. If the first super game an individual i plays terminates on the fifth round and both players choose to cooperate all their endowments in every round, then i's probability to contribute more than half the endowment in the initial round of the next super game would only increase to 44.4 percent. This modest increase demonstrates the gradual nature of the learning process in this model.

The positive $\hat{\beta}$ estimate indicates that as Δ^{RD} increases, the cooperation rate tends to increase. This aligns with the theoretical prediction that δ^{RD} increases as the return rate c

increases, suggesting that when the discount rate δ is sufficiently high and the return rate c is high (or the gain from cooperating is substantial), players are more inclined to contribute more than half of their endowments.[11] With the $\hat{\beta} = 0.758$, one unit increase in Δ^{RD} leads to the likelihood of contributing more than 50 percent of the endowment approximately 2.138 times higher, holding all other variables constant.

5.1.2 Action Set Partition and Accuracy Trade Off

Figure 1 illustrates the out-of-sample loss and accuracy specifically for Lugovskyy et al. [2017]. Analysis reveals that the out of sample loss for OLS and Lasso (mean squared error) is substantially better compared to loss for classification models (cross entropy loss) when the model's action set is partitioned to few ranges.[12] However, out of sample loss for classification models are more stable compared to the continuous models as the number of partition K increases. Additionally, the loss for the IRL-SG and Reinforcement Learning models are approximately the same as the best model free machine learning classification algorithms in terms of out of sample loss with lower loss as the action range increases. Complementary analysis on out of sample accuracy reveals that classification model's performance begins to plateau when $K = 13$ for models outlined in Section 3 and black box machine learning algorithms. A more detailed result with actual numbers is in Appendix D Table 3.

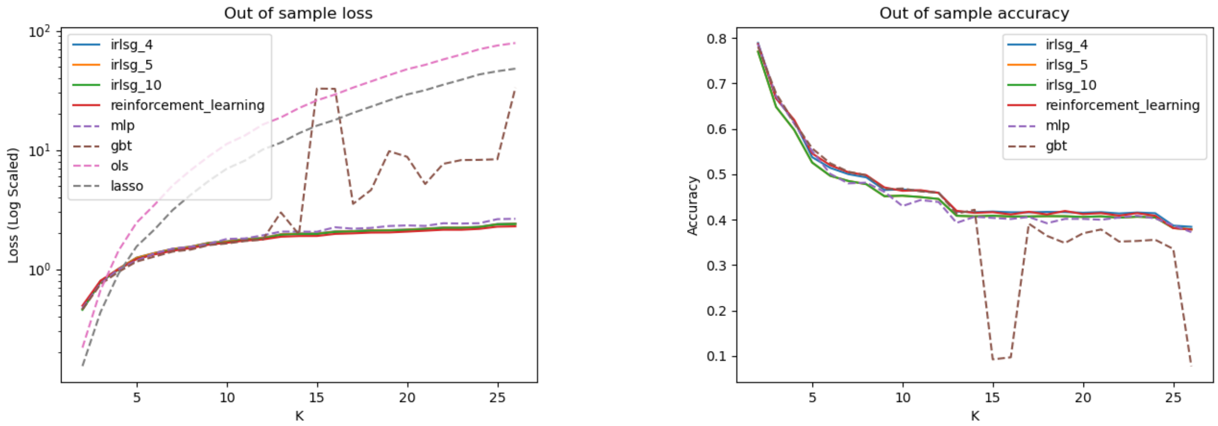
Models described in Section 3 all seem to have similar out of sample accuracy, with Reinforcement Learning and IRL-SG with 4 states performing the best. Additionally, initially GBT's accuracy is marginally higher than any of the structural model in Section 3, but later gets outperformed as the action set gets partitioned to more ranges. This plateau effect suggests that increasing the complexity of the action set partitions beyond $K = 13$ does not significantly affect model performance, indicating a potential optimal balance between model complexity and accuracy.

The graph also demonstrates a sharp initial decline in accuracy as K increases, followed

[11] Recall from Equation 1 that $\Delta^{RD} \equiv \delta - \delta^{RD}$ where $\delta^{RD} = \frac{\frac{2}{N}(N-c)}{1+\frac{c}{N}(N-2)}$ for a public goods game.

[12] The classification models include MLP, GBT, IRL-SG and Reinforcement Learning.

Figure 1. Out of Sample Loss and Accuracy for In Treatment Analysis



(A) Loss Value on out of sample observations (log scaled)

(B) Accuracy on out of sample observations

Note: These figures plot the out of sample loss and accuracy when the respective models are estimated using all observations from sessions 1 and 2 in Lugovskyy et al. [2017] and evaluated on session 3 in the same study.

by a more gradual decrease after $K = 5$. Notably, the last biggest trade off between the partition size and accuracy is $K = 13$ for both IRL-SG and Reinforcement Learning. Despite this loss of information, the stability observed beyond $K = 13$ combined with the fact that the overall diminishing decline in performance after $K = 5$ justifies combining this analysis with data from Mengel and Peeters [2011]. Their experiment uses an action set with cardinality of 11, allowing for a comprehensive analysis of action sets partitioned up to $K = 11$.

5.2 Across Treatment Analysis

The main analysis of this study considers out-of-sample fit evaluated on game parameter that was not observed in training set $\mathcal{D}'$. This approach offers robust insights into the generalizability of various learning models across different public goods game parameters. By estimating $\boldsymbol{\theta}$ from Lugovskyy et al. [2017] data and testing on Mengel and Peeters [2011] observations, I assess how well these models capture fundamental aspects of human behavior in diverse public goods scenarios.

5.2.1 Across Treatment Parameter Estimate

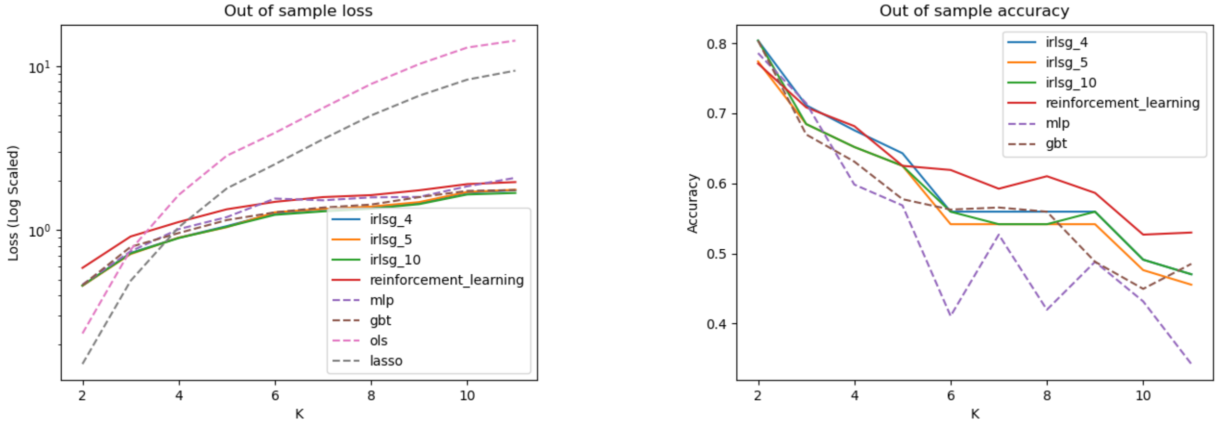
The parameter estimates and standard errors for IRL-SG when the action state is partitioned to above or below 50 percent contribution and the history state is partitioned into 4 ranges using all observations from Lugovskyy et al. [2017] are presented in Table 2 column 2. As expected, the estimates are fairly similar to column 1, which was discussed in Section 5.1.1, with slightly lower standard errors. The only estimated coefficient that is significantly different is $\hat{\beta}$. This is most likely due to the limited variability of the Δ^{RD} in the observations, contributing to higher standard deviation in the estimated coefficient relative to the other parameters.

Figure 7 demonstrates the fit of the IRL-SG (with 4 non-initial states) and Reinforcement Learning with action set partitioned to $K = 10$ ranges to the data. The simulation is based on 1000 i.i.d. samples from the parameter estimates $\hat{\boldsymbol{\theta}}$ and its covariance matrix, which are randomly matched to form groups of N players. These groups engage in infinitely repeated rounds of a public goods game, with the probability δ determining whether each super game continues. To align with the experimental observations, the simulation uses $\delta = 0.8$ and runs 15 super games for each treatment group. While the original study employed an action set with 26 discrete options, the simulation simplifies this to a model with 10 contribution ranges. To address this discrepancy, when the simulation predicts the kth range for a player's action, the player then selects a specific action within that range using a uniform distribution. For instance, in the original study the standardized contribution is $a_i \in \{0, 0.04, 0.08, 0.12, \ldots, 0.96, 1\}$, and if the simulation draws $a_i \in \mathcal{A}_1 = [0, 0.1)$, the final action will be randomly chosen from $\{0, 0.04, 0.08\}$ with equal probability. This approach maintains the granularity of the original study while working within the simplified model structure. Even with the simplified simulation, the models seem to fit the distribution of the overall contribution in the data with moderate accuracy.

5.2.2 Across Treatment Performance

Figure 2 shows the out of sample loss and accuracy of the model, which illustrates a consistent pattern across all models: a sharp decline in accuracy as K (the number of action set partitions) initially increases, followed by a plateau effect around $K = 5$. This suggests an optimal balance between model complexity and predictive power, indicating that finer partitioning beyond this point may not capture additional meaningful patterns in player behavior. Notably, IRL-SG with the state partitioned to 4 ranges yields the highest accuracy initially. However, as K increases, Reinforcement Learning demonstrates superior accuracy. This shift in relative performance as K increases could reflect different aspects of player decision-making being captured at various levels of model granularity.

Figure 2. Out of Sample Loss and Accuracy for Cross Treatment Analysis



(A) Loss Value on out of sample observations (log scaled)

(B) Accuracy on out of sample observations

Note: These figures plot the out of sample loss and accuracy when the respective models are estimated using all observations from Lugovskyy et al. [2017] and evaluated on Mengel and Peeters [2011].

A key finding is the consistent under-performance of black box machine learning algorithms in terms of accuracy compared to the models described in Section 3, regardless of K. This suggests that the explicitly structured models capture essential features of human strategic thinking in public goods games that are not easily learned by more general machine learning approaches. However, the loss value for Reinforcement Learning is worse than

MLP or GBT across all K while IRL-SG consistently better for out of sample loss, except for least squares model like OLS and Lasso for very low K. Finally, partitioning the state space to more ranges for IRL-SG yields marginal improvement at most for both accuracy and loss. This suggests the possible cognitive capacity that players takes into account of when choosing their action.

The ability for these models to generalize relatively well across unobserved variations of the public goods game is particularly noteworthy. This generalizability strengthens the case for these models as valuable tools for understanding and predicting behavior in a wide range of public goods scenarios.

5.3 Extrapolating Experiments

Practical experimental constraints limits researchers from running infinite super games of experimental games while simulations from the estimated model parameters provide a general overview of the counterfactual observations under a scenario where the experiment is conducted in infinite horizons. Here, instead of partitioning the observations, the model parameters will be obtained from all observations in the dataset. These estimated parameters will then be used to simulate hypothetical infinitely repeated public goods games.

5.3.1 Extrapolating to Hypothetical Session Lengths

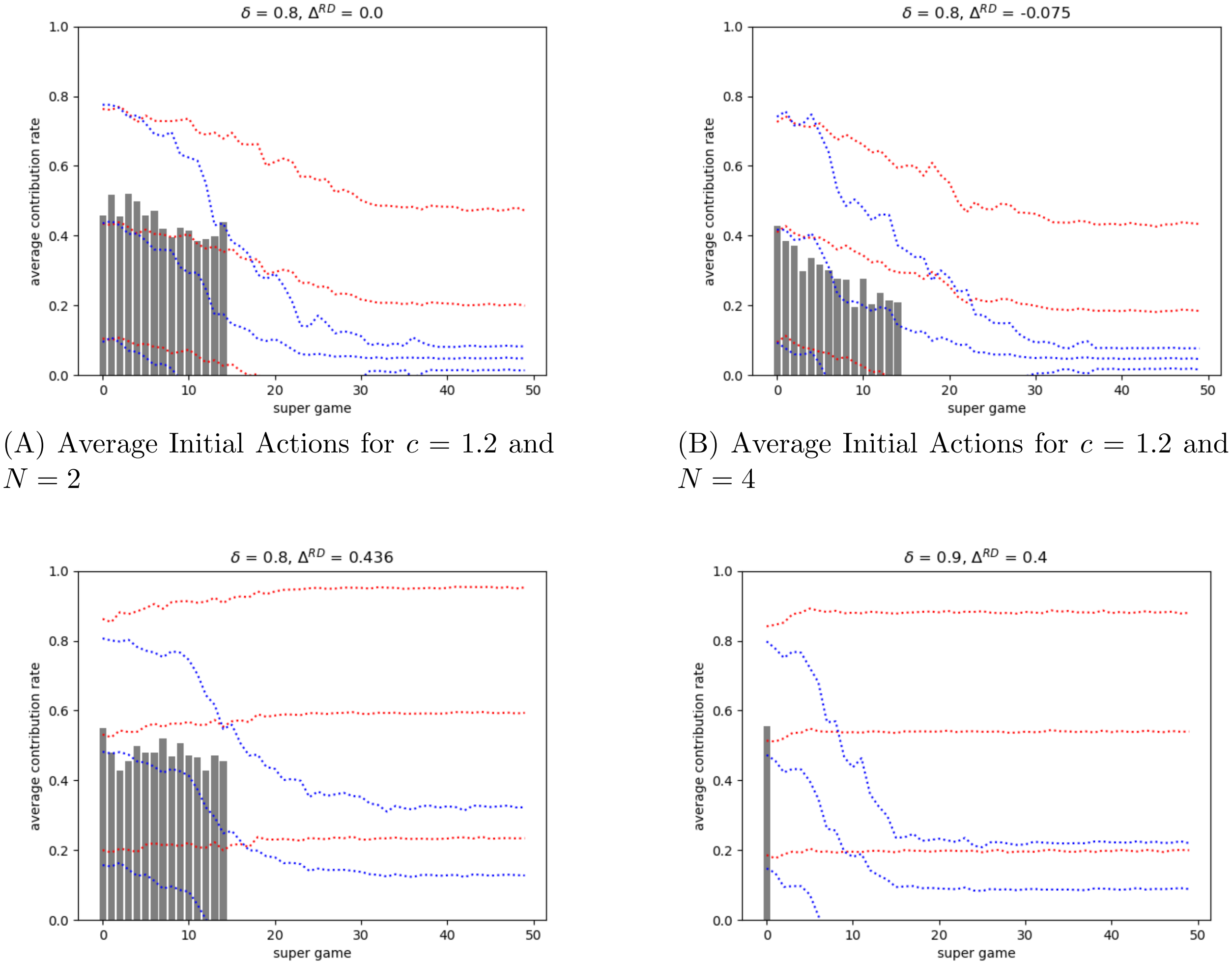
Figure 3 presents a comparison between the actual average initial actions observed in experimental data and simulated results extended to 50 super games. The analysis, based on 1000 i.i.d. samples from the parameter estimates on $K = 10$ model, demonstrates that the simulation achieves moderate accuracy in capturing the trends observed in the actual data.

The comparison in Figure 3B suggests that both IRL-SG (with 4 non-initial states) and Reinforcement Learning captures the trend of the average initial actions for low Δ^{RD}. However, the simulations do not quite align with the actual behavior in Figure 3A. This could be attributed to simply extrapolating the uncertainty parameter from prisoner's dilemma to public goods game in Blonski et al. [2011]. Recall from Equation 2 that the gain from defecting and loss from cooperating, g and l, are both affected by the number of players N. Compared to Figure 3C, Figure 3A has the same MPCR, but with different N, which may contribute to the potential differences in the trend in simulation. This discrepancy could suggest a further refinement in the model, specifically accounting for the number of players in the game to better capture the dynamics in multi-player long run behavior.

Additionally, Reinforcement Learning tends to underestimate the long run average initial actions compared to IRL-SG. Even with the highest Δ^{RD} out of all the datasets in Figure 3C Reinforcement Learning estimates that the initial actions of an average player would converge to roughly just under 20 percent. This is a byproduct of the relatively lower $\hat{\beta}$ that Reinforcement Learning estimates compared to IRL-SG. For example in Table 2, for $K = 2$ in all cases $\hat{\beta}$ Reinforcement Learning is significantly lower than IRL-SG while $\hat{\alpha}$ and $\hat{\lambda}$ are relatively close.

There are two possible hypotheses for this result. First is a recurrent issue with of the limited variability in Δ^{RD} in the observations and the suboptimal definition of uncertainty parameter for public goods game. Another explanation could just be that this is a result of the difference in learning mechanism outlined in IRL-SG and Reinforcment Learning. Recall from Section 3.1 that IRL-SG assumes players only learn in the initial round, where the associated experience on the initial action is total payoff of the super game. On the other hand, from Section 3.2, Reinforcement Learning assumes that learning happens every round and the experience on the associated action is the total payoff a player received from playing that action. Therefore, IRL-SG tends to be more optimistic because it attributes the entire sequence's cumulative payoff to the initial action unlike Reinforcement Learning. This can lead to higher expected utilities for actions, especially if sequences tend to have increasing or sustained payoffs, contributing to a more optimistic long run behavior.

Figure 3. Extrapolating Initial Actions



(A) Average Initial Actions for $c = 1.2$ and $N = 2$

(B) Average Initial Actions for $c = 1.2$ and $N = 4$

(C) Average Initial Actions for $c = 2.4$ and $N = 4$

(D) Average Initial Actions for $c = 2$ and $N = 4$

Note: These figures illustrate the initial average contribution rate for each super game. The black bars are the values from the data and the red dotted lines are the values from 1000 simulated samples from the IRL-SG with model with $K = 10$ and 4 states. The top and bottom red lines correspond to 1 standard deviation distribution away. The blue dotted lines are the same but for from simulated samples from Reinforcement Learning with $K = 10$.

5.3.2 Extrapolating to Hypothetical Public Goods Game

With the limited number of infinitely repeated public goods game experimental data, it is useful to further leverage the model estimates to run simulations on versions of public goods games that were not in the dataset. That is, given some unobserved combination of parameters δ, c, N running infinitely many super games will provide an idea of the overall picture of how public goods game parameters affect the long run outcome of the games.

Analysis of initial actions of simulated super games are plotted in Figure 8 now with 2000 i.i.d. samples with 10000 super games.[13] The simulation results indicate that public goods games with lower Δ^{RD} tend to display lower long run average initial contributions with smaller standard deviation. This pattern suggests that in games with lower Δ^{RD} the potential benefits of free-riding outweigh the expected long-term gains associated with a relatively lower discount rate δ. In these public goods games, the dynamics create a scenario where free-riding becomes a more apparent strategy for players. Consequently, this leads to a reduction in the standard error of initial actions over the long run, as player behavior becomes more predictable and converges towards free-riding strategies. The same pattern is observed in Figure 3.

Comparing the simulations between the two models, there is again a pattern where Reinforcement Learning consistently underestimates the long run average initial action compared to IRL-SG. The only public goods game scenario that led to a long run average action of over 50 percent is in Figure 8B, where the $\Delta^{RD} = 0.7$ is relatively high. The two simulated models only have similar long run trends when Δ^{RD}, as observed in Figure 8D.

6 Conclusion

This study bolsters the validity of IRL-SG on slightly more complicated social dilemma setting. The model does a relatively good job predicting out of sample observations, and it

[13] I use more simulated samples for longer length of super games to reduce the likelihood of the simulation reforming groups that had already been made.

even does so without losing information compared to black box machine learning algorithms. A counterfactual model, Reinforcement Learning, tends to fit the model better overall as the action sets are partitioned into more number of discrete sets. However, as the simulation suggests, Reinforcement Learning is pessimistic about cooperation rate in initial rounds compared to IRL-SG. This could be attributed to the following possibilities. Despite the differences, both models suggest that higher Δ^{RD} will lead to long run cooperations, which is consistent with existing literature Fudenberg and Rehbinder [2024]. Games where the discount factor δ is high relative to the gains from trade and loss from defection δ^{RD} leads to long run cooperation.

Although the results I present show that IRL-SG generalizes well to a more complex social dilemma problem in infinitely repeated settings, there are limitations mainly due to the lack of observations I had in the dataset used to empirically estimate and evaluate the model. First is the lacked game parameter variety. The observation only had four different variation of public goods game payoff structures. A better way to account for this would to treat the Δ^{RD} terms as dummies instead. However, for the purpose of simulating the model the parameter estimate I kept it continuous. Second, I did not do any cross-fold validation to avoid overfitting. This is a slightly tricky problem with the dataset I was working with. A standard 10 fold validation might seem fine with the amount of observations I had in the dataset. However, the observation is sequential in nature so it would only make sense for me to partition the validation sets by players. With less than 150 players in the dataset, conducting a 10 fold validation would lead to overfitting th eestimations on the validation set. A natural alternative of regularizing would be to use Empirical Bayes and estimate the parameter using Maximum a posteriori estimation instead of maximum likelihood estimation. I discuss this in Section A.2. There are also some limitations economic models I propose. First, I don't account for heterogeneity in player profiles. Proto et al. [2019] indicates that groups comprising more intelligent individuals demonstrate a higher propensity for cooperation. To address this limitation, I can consider a finite mixture model that introduces heterogeneity among players. A comprehensive description of this model is provided in Section A.3.

Even with these limitations, the findings open up several avenues for future investigation. Researchers could explore why certain models perform better at different K values, potentially uncovering insights into the cognitive processes underlying decision-making in these games. Additionally, testing these models on public goods games with different structural features (e.g., varying group sizes, contribution mechanisms, or payoff structures) could further validate their robustness and identify any boundaries to their applicability. There is also the open-ended problem of what the best metric is to evaluate models. I simply looked at the out of sample loss of the models, but other criteria such as Akaike Information Criterion (AIC) or Bayesian Information Criterion (BIC) could be alternatives.

References

Susan Athey and Kyle Bagwell. Optimal collusion with private information. *RAND Journal of Economics*, 32(3), Autumn 2001.

Martino Banchio and Giacomo Mantegazza. Artificial intelligence and spontaneous collusion, 2023. URL https://arxiv.org/abs/2202.05946.

Matthias Blonski, Peter Ockenfels, and Giancarlo Spagnolo. Equilibrium selection in the repeated prisoner's dilemma: Axiomatic approach and experimental evidence. *American Economic Journal: Microeconomics*, 3(3), August 2011.

Yves Breitmoser. Cooperation, but no reciprocity: Individual strategies in the repeated prisoner's dilemma. *American Economic Review*, 105(9), September 2015.

Emilio Calvano, Giacomo Calzolari, Vincenzo Denicolò, and Sergio Pastorello. Artificial intelligence, algorithmic pricing, and collusion. *American Economic Review*, 110(10), October 2020.

Pedro Dal Bó and Guillaume R. Fréchette. On the determinants of cooperation in infinitely repeated games: A survey. *Journal of Economic Literature*, 56(1), March 2018.

Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1), September 1977.

Shidi Deng, Maximilian Schiffer, and Martin Bichler. Algorithmic collusion in dynamic pricing with deep reinforcement learning, 2024. URL https://arxiv.org/abs/2406.02437.

Ido Erev and Alvin E. Roth. Predicting how people play games: Reinforcement learning in experimental games with unique mixedstrategy equilibria. *American Economic Review*, 88(4), September 1998.

Drew Fudenberg and Annie Liang. Predicting and understanding initial play. *American Economic Review*, 109(12), November 2019.

Drew Fudenberg and Gustav Karreskog Rehbinder. Predicting cooperation with learning models. *American Economic Review*, 16(1), February 2024.

Drew Fudenberg, David G. Rand, and Anna Dreber. Slow to anger and fast to forgive: Cooperation in an uncertain world. *American Economic Review*, 102(2), April 2012.

Joseph E. Harrington. *The Theory of Collusion and Competition Policy.* MIT Press, 2017.

Valen E. Johnson and James H. Albert. *Ordinal Data Modeling.* Springer Science Business Media, 1999.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. URL `https://arxiv.org/abs/1412.6980`.

Volodymyr Lugovskyy, Daniela Puzzello, Andrea Sorensen, James Walker, and Arlington Williams. An experimental study of finitely and infinitely repeated linear public goods games. *Games and Economic Behavior*, 102, March 2017.

Friederike Mengel and Ronald Peeters. Strategic behavior in repeated voluntary contribution experiments. *Journal of Public Economics*, 95, February 2011.

Friederike Mengel, Ludovica Orlandi, and Simon Weidenholzer. Match length realization and cooperation in indefinitely repeated games. *Journal of Economic Theory*, 200, 2022.

Panayotis Mertikopoulos and William H. Sandholm. Learning in games via reinforcement and regularization. *Mathematics of Operations Research*, 41(4), August 2016.

Clemens Possnig. *Reinforcement Learning and Collusion.* PhD thesis, University of British Columbia, 2023.

Eugenio Proto, Aldo Rustichini, and Andis Sofianos. Intelligence, personality, and gains from cooperation in repeated interactions. *Journal of Political Economy*, 127(3), June 2019.

Julio J. Rotemberg and Garth Saloner. A supergame-theoretic model of price wars during booms. *American Economic Review*, 76(3), June 1986.

Maximilian Schaefer. On the emergence of cooperation in the repeated prisoner's dilemma, 2023. URL `https://arxiv.org/abs/2211.15331`.

James R. Wright and Kevin Leyton-Brown. Predicting human behavior in unrepeated, simultaneous-move games. *Games and Economic Behavior*, 106, November 2017.

Appendix

A Model Solution

A.1 Multi Colinearity in Multi-Range Model

Due to multicolinearity when estimating the multi-ranged model explained in Section 3.1.1 and Section 3.2, calculating the variance of the estimates will have numerical stability issues when computing the information matrix. This is a problem since a primary part of the finding in my study comes from drawing samples from the parameter estimates, which relies on the ability to compute the covariance matrix. Therefore, instead of directly estimating the parameters laid out in Equation 3, the maximum likelihood estimate will estimate the following coefficients where for the model with up to K finite action ranges, the probability that player i contributes within the range $\mathcal{A}_K$ is

$$\begin{aligned} P(a_i \in \mathcal{A}_K \mid s) &= \frac{1}{1+\exp(-(\sum_{k' \neq K} \alpha_{k'} + \beta_{k'}\Delta^{RD} + e_{ik'}(s)))} \\ e_{ik}(s) &= \lambda a_{ik}(s-1)V_i(s-1) + e_{ik}(s-1), \end{aligned}$$

where $a_{ik}(s) = 1$ if the initial action in super game s was in the range $\mathcal{A}_K$. $a_{ik}(s) = -1$ if the initial action was in the range $\mathcal{A}_k$, and 0 otherwise. For all $k \neq K$, the probability of the contribution lying within range $\mathcal{A}_k$ is

$$P(a_i \in \mathcal{A}_k \mid s) = \frac{\exp(-(\alpha_k + \beta_k\Delta^{RD} + e_{ik}(s)))}{1+\exp(-(\sum_{k' \neq K} \alpha_{k'} + \beta_{k'}\Delta^{RD} + e_{ik'}(s)))}$$

The same will be done to estimate the parameters in Reinforcement Learning.

A.2 Regularization

Recall that for IRL-SG $\boldsymbol{\sigma}$ is a matrix, where the hth row of the matrix denotes the vector $\boldsymbol{\sigma}_h$ that denotes the parameter of the categorical distribution of the action given the state h. A natural way to regularize when estimating for Categorical random variable is using Dirichlet distribution $D(\boldsymbol{\zeta}_h)$ as the prior and maximize $P(\boldsymbol{\sigma}_h \mid \mathcal{D}', \boldsymbol{\zeta}_h)$, where $\boldsymbol{\zeta}_h$ is the vector of hyper parameters for the categorical distribution at state h. A naive way to choose this would be to use cross validation, but with the risk of overfitting the validation set. An alternative approach would be to use Empirical Bayes to pick $\boldsymbol{\zeta}_h$.

$$\begin{aligned} \Psi(\boldsymbol{\zeta}_h) = \int_{\boldsymbol{\theta}} P(\mathcal{D}' \mid \boldsymbol{\sigma}_h) P(\boldsymbol{\sigma}_h \mid \boldsymbol{\zeta}) d\boldsymbol{\theta} \\ \hat{\boldsymbol{\zeta}}_h \in \arg\max_{\boldsymbol{\zeta}_h} \{\Psi(\boldsymbol{\zeta}_h)\} \end{aligned} \tag{6}$$

Equation 5 is simply maximizing the marginal probability with respect to the hyper parameter $\boldsymbol{\zeta}_h$. For example, for state h the vector $_h$ would correspond to Equation 7 where n_{kh} is the number of samples in the observation that the contribution was in the kth action range, given state h.

$$\hat{\boldsymbol{\zeta}}_h \in \arg\max_{\boldsymbol{\zeta}_h} \left\{ \frac{D(n_{1h} + \zeta_{1h}, \ldots, n_{Kh} + \zeta_{Kh})}{D(\zeta_{1h}, \ldots, \zeta_{Kh})} \right\}. \tag{7}$$

Instead of maximizing with respect to a K-dimensional space, an alternative approach can maximize the objective function Equation 8, where n_h is the total number of observations at state h.[14]

$$\hat{\eta} \in \arg\max_{\eta} \left\{ \frac{D(n_{1h} + \eta \frac{n_{1h}}{n_h}, \ldots, n_{Kh} + \eta \frac{n_{Kh}}{n_h})}{D(\eta \frac{n_{1h}}{n_h}, \ldots, \eta \frac{n_{Kh}}{n_h})} \right\} \tag{8}$$

For the initial round, where $h = \varnothing$, the estimator $\hat{\boldsymbol{\alpha}}$, $\hat{\boldsymbol{\beta}}$ and $\hat{\lambda}$ can be regularized using a similar approach. However, the integral in Equation 5 does not yield a closed form solution

[14] This approach is how Johnson and Albert [1999] suggest for hyper parameter tuning.

for a multinomial logistic models since the posterior distribution is a normal distribution. Thus, estimating the marginal distribution with some approximation techniques such as Markov chain Monte Carlo (MCMC) or minimizing the KL divergence would be a natural solution.[15]

A.3 Mixture Model

Assume there are Z classes of players, and each player i belongs to a latent class $j \in Z$. Each latent class will have different $\boldsymbol{\theta}_j$, and let $\boldsymbol{\Theta}$ encompass all the parameters for the latent classes. Then, given the observations on player i, $\mathcal{D}_i = \{h_i(t), a_i(t)\} \subseteq \mathcal{D}$, the probability that i belongs to latent class j, $\phi_j(i, \boldsymbol{\Theta})$, is

$$\phi_j(i, \boldsymbol{\Theta}) = \frac{\pi_j P(\mathcal{D}_i \mid \boldsymbol{\theta}_j)}{\sum_{j'} \pi_{j'} P(\mathcal{D}_i \mid \boldsymbol{\theta}_{j'})},$$

where π_j is the probability that player i belongs to latent class j. Note that this is not the same as $\phi_j(i, \boldsymbol{\Theta})$, which is the conditional probability.

The likelihood of the mixture model is maximized by Expectation Maximization, which is an iterative algorithm to maximize the expected log likelihood Dempster et al. [1977]. The algorithm starts with initializing $\boldsymbol{\Theta}$ and all π_j, then conditional on these parameters optimize the expected log likelihood to obtain an updated.

$$\underset{Z|\mathcal{D},\boldsymbol{\Theta}_\tau}{E} \log P(\mathcal{D}, Z \mid \boldsymbol{\Theta}_\tau) = \sum_{i=1}^{I} \sum_{j \in Z} \phi_j(i, \boldsymbol{\Theta}_\tau) \left[\log \pi_j + \sum_{t=1}^{T_i} \log P(a_i(t) \mid h_i(t), \boldsymbol{\theta}_j) \right]$$

$$\boldsymbol{\Theta}_{\tau+1} \in \underset{\boldsymbol{\Theta}_{\tau+1}}{\arg\max} \left\{ \underset{Z|\mathcal{D},\boldsymbol{\Theta}_\tau}{E} \log P(\mathcal{D}, Z \mid \boldsymbol{\Theta}_\tau) \right\}$$

After each iteration, the algorithm updates π_j and $\phi_j(i, \boldsymbol{\Theta}_{\tau+1})$, and it terminates when the expected log likelihood doesn't improve.

[15] Regularizing for the estimators Reinforcement Learning would be the same for regularizing for the intial round parameter in IRL-SG.

B Estimation

Consider a function $m_K(i, t; \boldsymbol{\theta}) : \{a_i(t), h_i(t)\} \rightarrow [0, 1]^K$, where m_K stands for the model m with K action set partitions, that maps the sequential history of observations for player i to a K dimensional vector that sums to 1, and each kth entry corresponds to the probability that the action falls in the $\mathcal{A}_k$. Then the objective function that needs to be minimized to estimate $\hat{\boldsymbol{\theta}}$ is the cross entropy function in Equation 9.

$$-\frac{1}{|\mathcal{D}'|} \sum_{i \in \mathcal{D}'} \sum_{t \in T_i} e(i, t)^\top \log(m_K(i, t; \boldsymbol{\theta})) \tag{9}$$

$e(i, t)$ is simply the indicator vector where $e(i, t)_k = 1$ if $a_i(t) \in \mathcal{A}$, and $e(i, t)_k = 0$ otherwise.

C Evaluation

For some estimated $\hat{\boldsymbol{\theta}}$, consider a estimation problem such that given $\{a_i(t), h_i(t)\}$, the model wants to predict which range of values the $a_i(t)$ lies in. Thus, for classification models like IRL-SG and Reinforcement Learning, this would be the entry that corresponds to the highest value from m_K. Consequently, the accuracy is defined as

$$\frac{1}{|\tilde{\mathcal{D}}|} \sum_{i \in \tilde{\mathcal{D}}} \sum_{t \in T_i} 1(a_i(t) = \hat{k}(i, t; \hat{\boldsymbol{\theta}}))$$

$$\hat{k}(i, t; \hat{\boldsymbol{\theta}}) \in \arg\max_k \{m_K(i, t; \hat{\boldsymbol{\theta}})_k\}.$$

D Tables

Table 2. Parameter Estimates for Contribution Over or Under 50 Percent

	IRL-SG with 4 states			Reinforcement Learning		
	(1)	(2)	(3)	(4)	(5)	(6)
$\hat{\alpha}$	-0.454^{***} (0.080)	-0.469^{***} (0.066)	-0.466^{***} (0.066)	-0.378^{***} (0.039)	-0.427^{***} (0.031)	-0.427^{***} (0.031)
$\hat{\beta}$	0.758^{**} (0.328)	0.943^{***} (0.269)	0.994^{***} (0.264)	0.526^{***} (0.139)	0.727^{***} (0.109)	0.623^{***} (0.107)
$\hat{\lambda}$	0.046^{***} (0.003)	0.045^{***} (0.002)	0.045^{***} (0.002)	0.047^{***} (0.002)	0.041^{***} (0.001)	0.042^{***} (0.001)
$\hat{\sigma}_1$	0.046^{***} (0.004)	0.053^{***} (0.003)	0.054^{***} (0.003)			
$\hat{\sigma}_2$	0.414^{***} (0.011)	0.380^{***} (0.009)	0.377^{***} (0.009)			
$\hat{\sigma}_3$	0.637^{***} (0.017)	0.629^{***} (0.013)	0.630^{***} (0.013)			
$\hat{\sigma}_4$	0.775^{***} (0.042)	0.838^{***} (0.021)	0.849^{***} (0.020)			
Observations	7104	10888	11224	7104	10888	11224
Num. Players	120	132	156	120	132	156

Note: *p<0.1; **p<0.05; ***p<0.01

Table 3. Maximum Likelihood Estimates from Lugovskyy et al. [2017] Session 1 and 2 with Out-of-Sample Loss and Accuracy Evaluated on Session 3

Model	**K**	**Loss**	**Accuracy**	**ML Model**
OLS	2	0.219		
	3	0.665		
	4	1.429		
	5	2.454		
	6	3.470		
	7	5.039		
	8	6.766		
	9	8.828		
	10	11.221		
	11	13.203		
IRL-SG with 4 states	2	0.465	78.9%	
	3	0.794	66.7%	
	4	1.006	61.8%	
	5	1.248	53.8%	
	6	1.367	51.4%	
	7	1.480	50.0%	
	8	1.546	49.3%	
	9	1.664	46.5%	
	10	1.712	46.6%	
	11	1.790	46.4%	
Reinforcement Learning	2	0.493	78.7%	
	3	0.800	66.7%	
	4	0.996	62.0%	
	5	1.215	54.6%	
	6	1.330	52.0%	
	7	1.426	50.4%	
	8	1.501	49.8%	
	9	1.604	47.1%	
	10	1.664	46.4%	
	11	1.736	46.5%	
Machine Learning Model with Lowest Loss	2	0.154		Lasso
	3	0.435		Lasso
	4	0.917		Lasso
	5	1.158	55.7%	GBT
	6	1.275	52.4%	GBT
	7	1.403	50.6%	GBT
	8	1.459	49.8%	GBT
	9	1.641	46.7%	GBT
	10	1.719	46.9%	GBT
	11	1.759	46.2%	GBT

Table 4. Maximum Likelihood Estimates from Lugovskyy et al. [2017] with Out-of-Sample Loss and Accuracy Evaluated on Mengel and Peeters [2011]

Model	**K**	**Loss**	**Accuracy**	**ML Model**
OLS	2	0.234		
	3	0.757		
	4	1.637		
	5	2.840		
	6	3.914		
	7	5.549		
	8	7.767		
	9	10.278		
	10	12.965		
	11	14.293		
IRL-SG with 4 states	2	0.457	80.4%	
	3	0.724	71.1%	
	4	0.898	67.6%	
	5	1.056	64.3%	
	6	1.266	56.0%	
	7	1.339	56.0%	
	8	1.380	56.0%	
	9	1.463	56.0%	
	10	1.691	49.1%	
	11	1.748	47.0%	
Reinforcement Learning	2	0.588	77.1%	
	3	0.913	70.8%	
	4	1.119	68.2%	
	5	1.338	62.5%	
	6	1.486	61.9%	
	7	1.588	59.2%	
	8	1.633	61.0%	
	9	1.748	58.6%	
	10	1.905	52.7%	
	11	1.960	53.0%	
Machine Learning Model with Lowest Loss	2	0.153		Lasso
	3	0.490		Lasso
	4	0.958	63.1%	GBT
	5	1.150	57.7%	GBT
	6	1.283	56.3%	GBT
	7	1.371	56.5%	GBT
	8	1.430	56.0%	GBT
	9	1.587	48.8%	GBT
	10	1.733	44.9%	GBT
	11	1.755	48.5%	GBT

E Figures

Figure 4. Prisoner's Dilemma Payoff Matrix

Prisoner's Dilemma Payoffs

	C	D
C	R, R	P, W
D	W, P	L, L

Standardized Prisoner's Dilemma Payoffs

	C	D
C	$1, 1$	$-l, 1+g$
D	$1+g, -l$	$0, 0$

Note: Here C denotes cooperation between the players and D denotes defection. The reward R is attained when both players cooperate, the loser's reward L is obtained when both players defect. When one player cooperates and the other defects, the cooperator receives P (for punishment) and the defector receives W (for winning). It is assumed that $P < L < R < W$. Note that the game is symmetric, both players have the same payoff matrix. The payoff matrix is usually standardized to enable comparison between different prisoner's dilemma payoff parameters.

Figure 5. Public Goods Game Payoff Matrix

Public Goods Game Payoffs

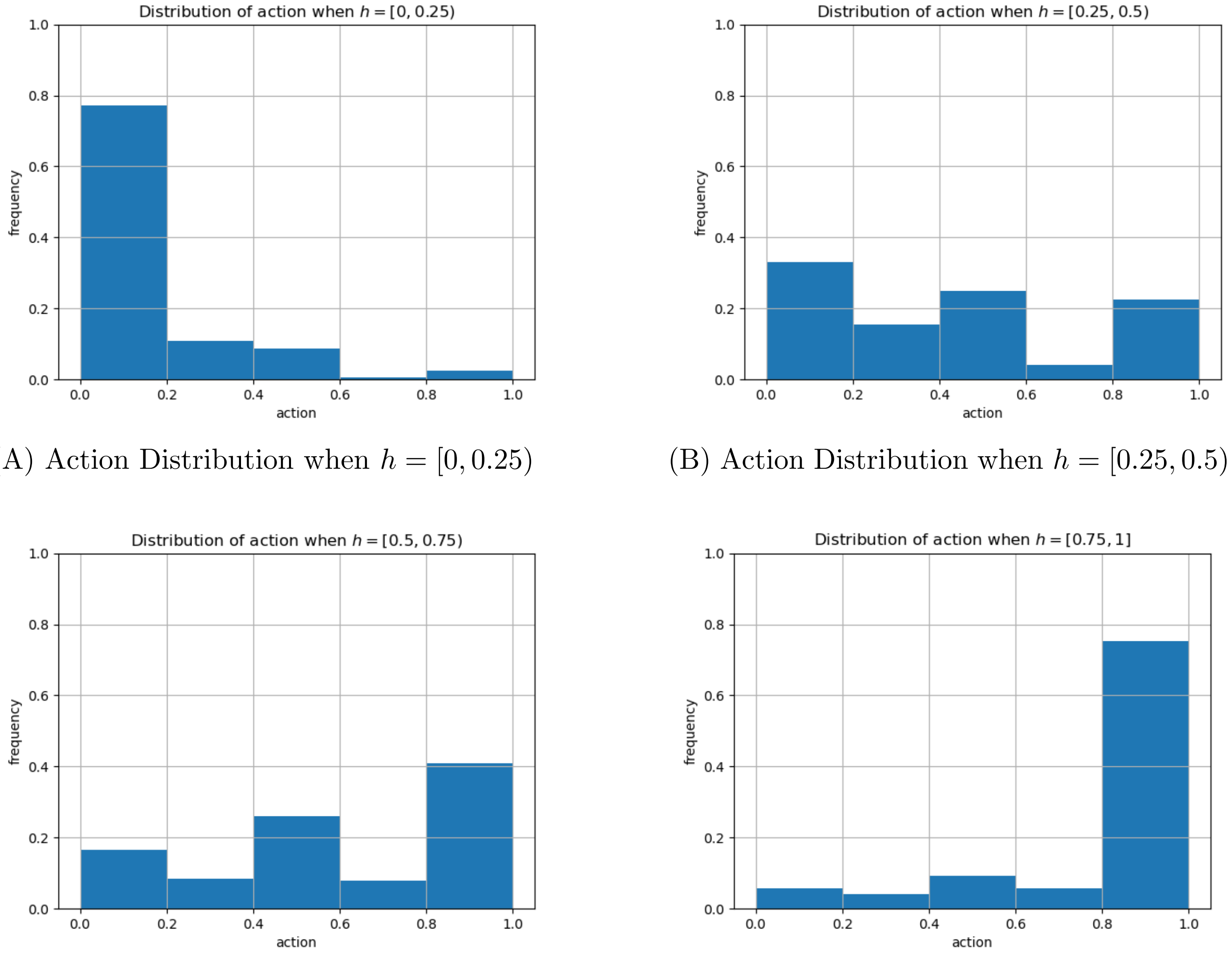
	$a_{-i} = (\rho, \rho, \dots)$	$a_{-i} = (0, 0, \dots)$
$a_i = \rho$	$c\rho$	$\frac{c\rho}{N}$
$a_i = 0$	$\rho + \frac{c(N-1)\rho}{N}$	ρ

Standardized Public Goods Game Payoffs

	$a_{-i} = (\rho, \rho, \dots)$	$a_{-i} = (0, 0, \dots)$
$a_i = \rho$	1	$\frac{c/N-1}{c-1}$
$a_i = 0$	$\frac{c-c/N}{c-1}$	0

Note: Here are the payoffs for player i in the extreme case that all players contribute all their endowments ρ or nothing at all. This follows a social dilemma structure if $c \in (1, N)$ then the payoff maintains the relationship of $\frac{c\rho}{N} < \rho < c\rho < \rho + \frac{c(N-1)\rho}{N}$. To make public goods games comparable across different parameters the analysis will look at the standardized payoffs.

Figure 6. Action Distribution for Non-Initial Rounds



(A) Action Distribution when $h = [0, 0.25)$

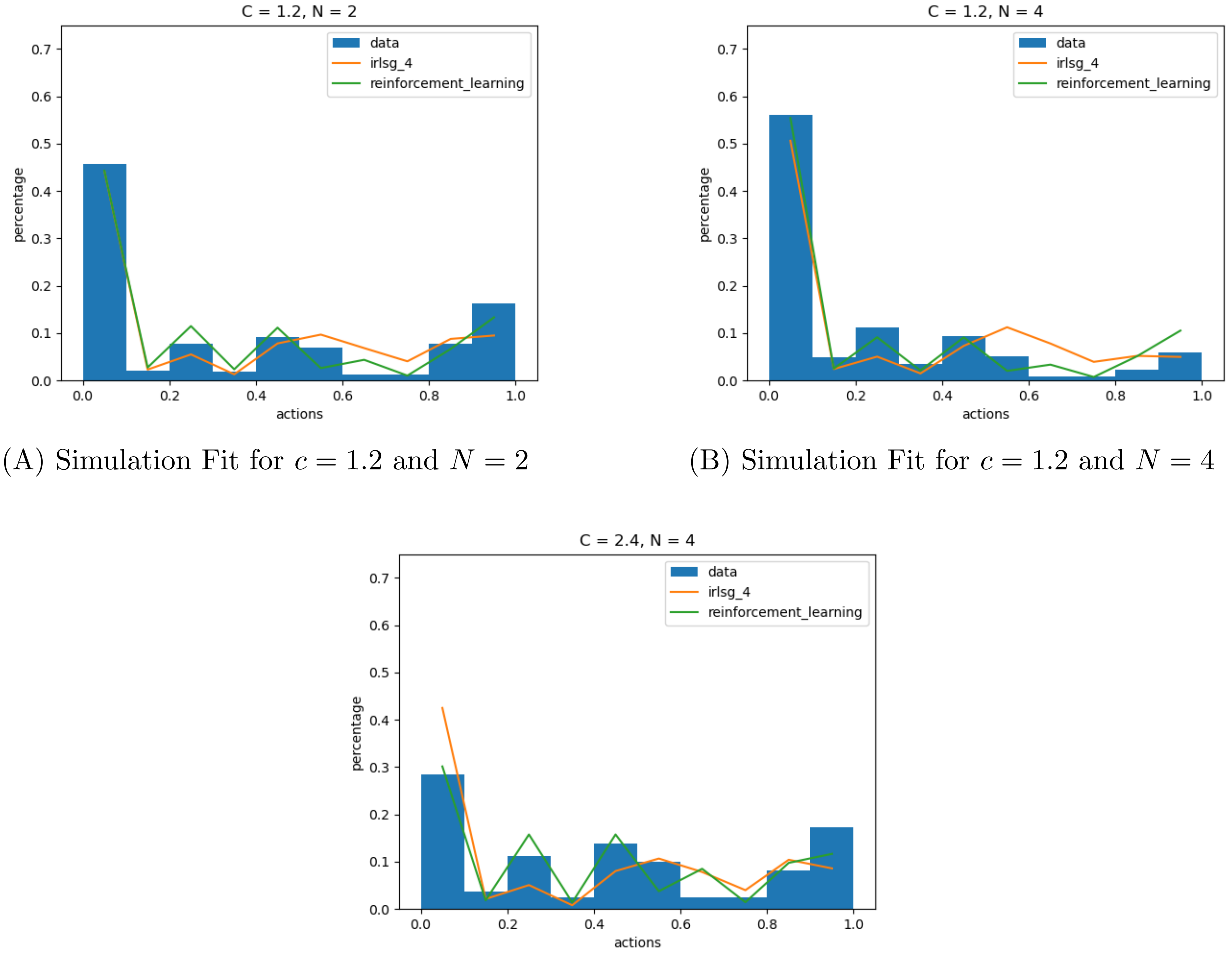
(B) Action Distribution when $h = [0.25, 0.5)$

(C) Action Distribution when $h = [0.5, 0.75)$

(D) Action Distribution when $h = [0.75, 1)$

Note: These figures illustrate the distribution of contribution percentage for non-initial rounds from Lugovskyy et al. [2017] and Mengel and Peeters [2011].

Figure 7. Model Fit



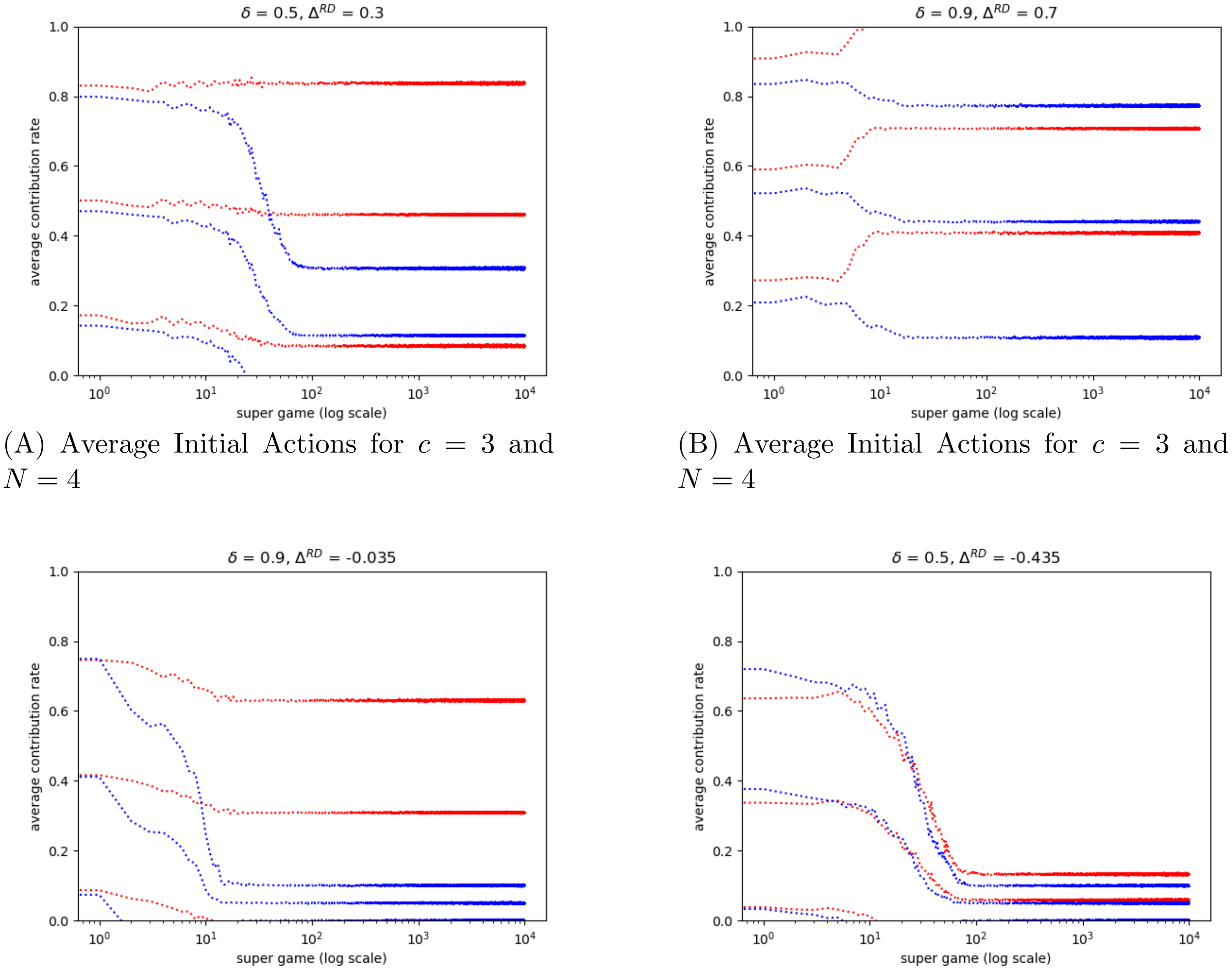
(A) Simulation Fit for $c = 1.2$ and $N = 2$

(B) Simulation Fit for $c = 1.2$ and $N = 4$

(C) Simulation Fit for $c = 2.4$ and $N = 4$

Note: These figures illustrate the distribution of contribution percentage from Lugovskyy et al. [2017] versus the simulated results, where lines corresponds to the shares of contribution from 1000 simulated samples.

Figure 8. Initial Actions of Hypothetical Infinitely Repeated public goods games



(A) Average Initial Actions for $c = 3$ and $N = 4$

(B) Average Initial Actions for $c = 3$ and $N = 4$

(C) Average Initial Actions for $c = 1.1$ and $N = 4$

(D) Average Initial Actions for $c = 1.1$ and $N = 4$

Note: These figures illustrate the initial average contribution rate for each super game. The red dotted lines are the values from 2000 simulated samples from the IRL-SG with model with $K = 10$ and 4 states. The top and bottom red lines correspond to 1 standard deviation distribution away. The blue dotted lines are the same but for from simulated samples from Reinforcement Learning with $K = 10$.