Learning Parameterized Families of Games

Madelyn Gatchel[1,2] & Bryce Wiedenbeck[1]
[1]Davidson College, [2]University of Michigan

Background

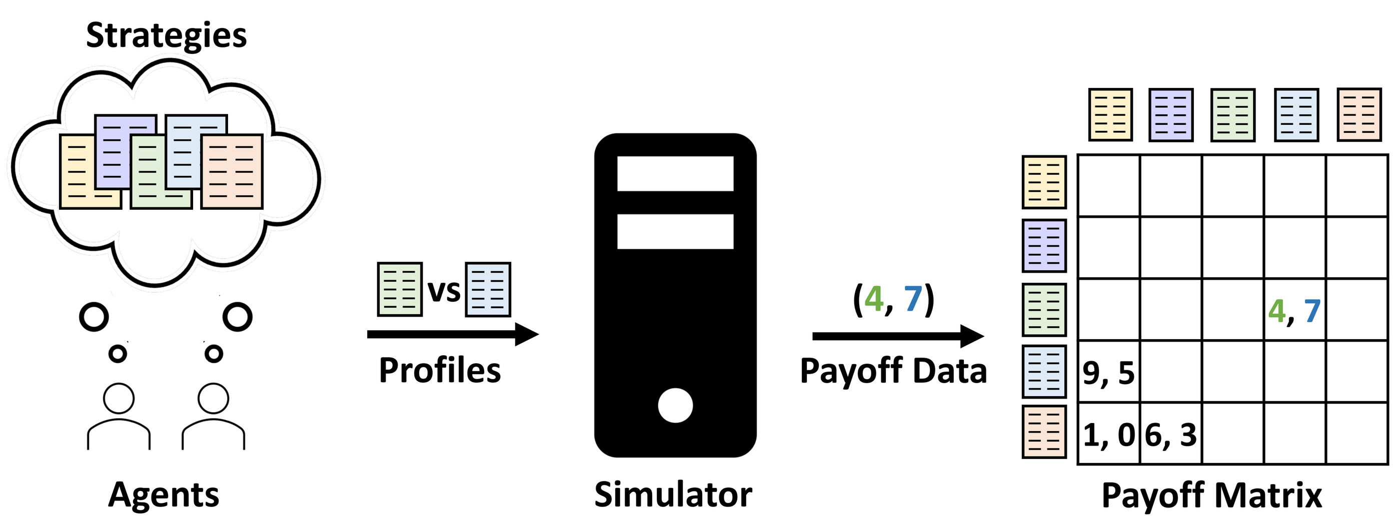
- **Normal-form game:** a mathematical model of incentives which includes
  - A fixed set of players
  - The possible actions (strategies) they can play
  - Each player's payoff function
- **Deviation payoff:** the expected payoff a player would receive by deviating or changing strategies, given the mixed strategies everyone else is playing
- **Regret:** the maximum payoff amount any player can gain by deviating to any other strategy
- **Nash equilibrium:** a set of strategies such that no player has an incentive to deviate

Example: Rock-Paper-Scissors

Player 1 \ Player 2	R	P	S
R	0, 0	-1, 1	1, -1
P	1, -1	0, 0	-1, 1
S	-1, 1	1, -1	0, 0

Empirical Game Theory

Our techniques are particularly relevant for analyzing **simulation-based games**, where the payoff matrix is not known in advance but can be filled through a series of multi-agent simulations.



Applications of simulation-based games:

- Latency arbitrage by high-frequency traders [4]
- Debt consolidation among financial institutions [2]
- Credit network liquidity [1]

Typically, the underlying agent-based model has **many free parameters** such as the number of background traders [4], recovery rate [2], and the probability of defaults [1]. Each parameter setting results in a distinct normal-form game. Existing techniques require **separate model construction and analysis for each parameter setting**, so in practice the environment parameter space is usually underexplored.

Research Questions

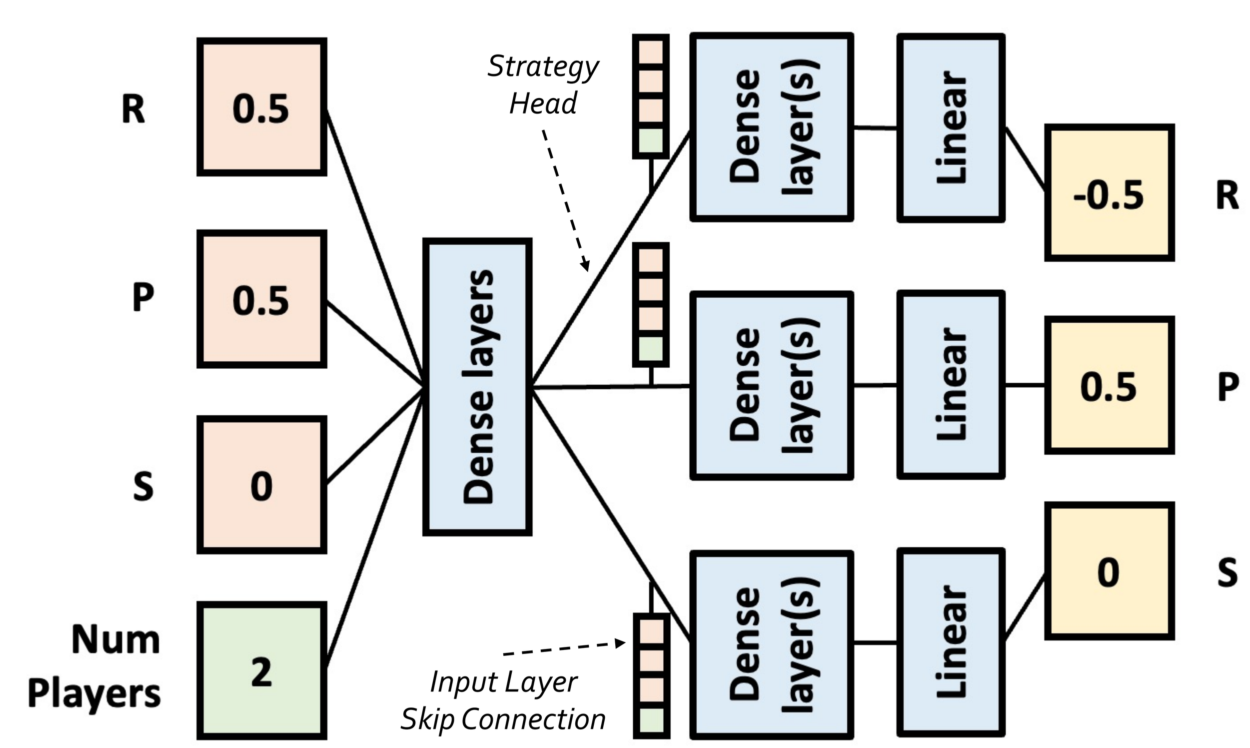
1. How can we use machine learning to construct a game-theoretic model that generalizes over variable environment parameters?
2. What types of game-theoretic analysis of the parameter space does the learned model enable?

Methodology

- **Game family:** a set of game instances that are related by one or more ordinal environment parameter(s)
- **Hypothesis:** game instances from the same game family likely have related payoff and deviation payoff functions
- We use a **multi-headed neural network** to learn a mapping from symmetric mixed-strategy profiles and environment parameters to deviation payoffs

VPL Neural Network Architecture

(shown with Rock-Paper-Scissors as example)



Proposed Analysis

Functions of Equilibrium:

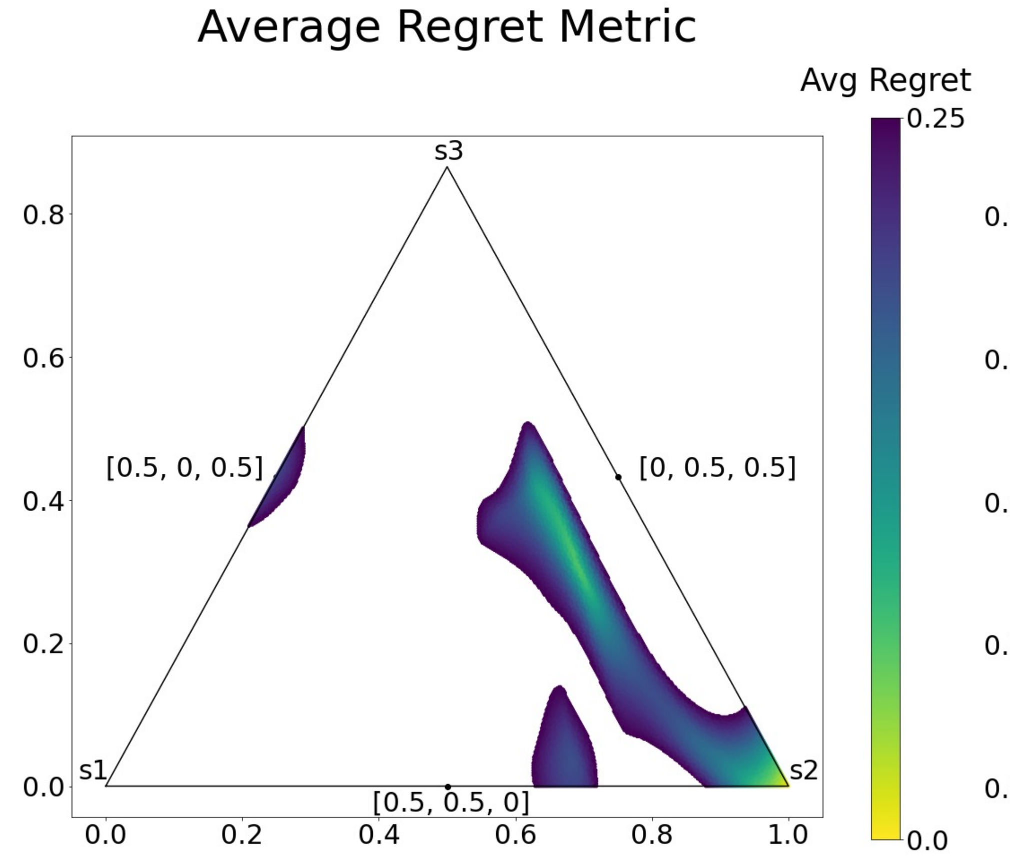
- Regret robustness metrics
- Social welfare, price of anarchy analysis
- Domain-specific equilibrium statistic

Parameter Sensitivity Analysis:

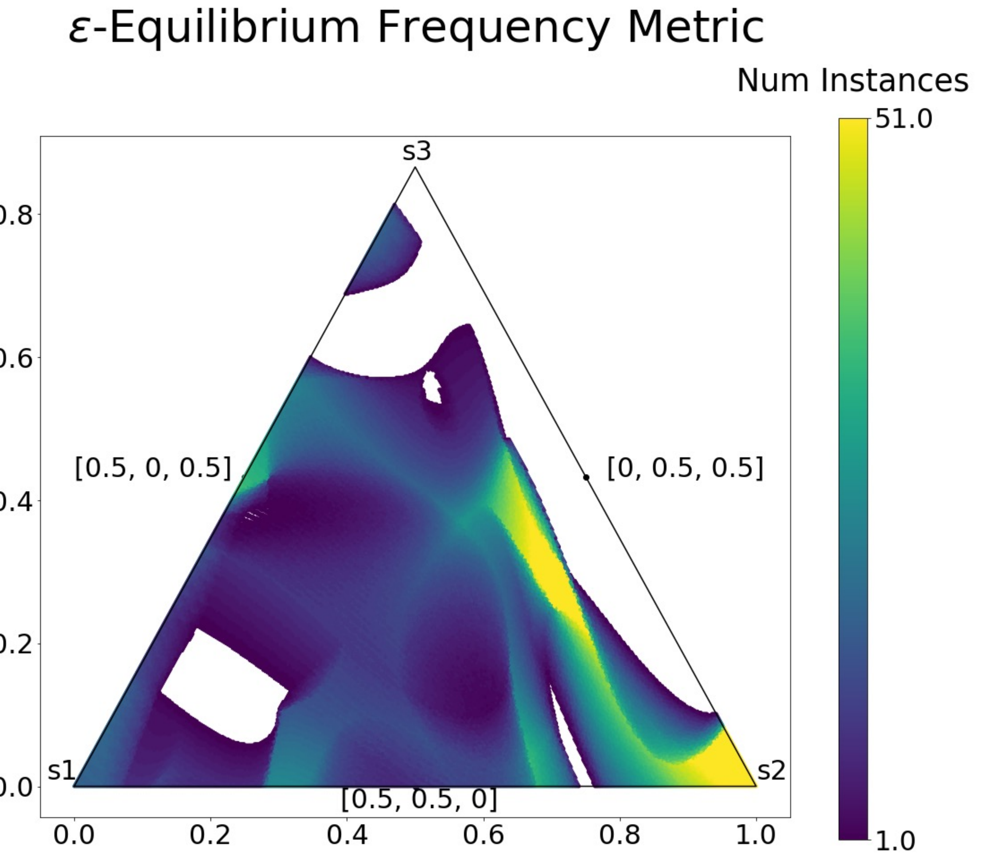
- Feature analysis
- Game instance clustering
- Equilibrium basins of attraction

Example: Regret Robustness Metrics

Average Regret Metric

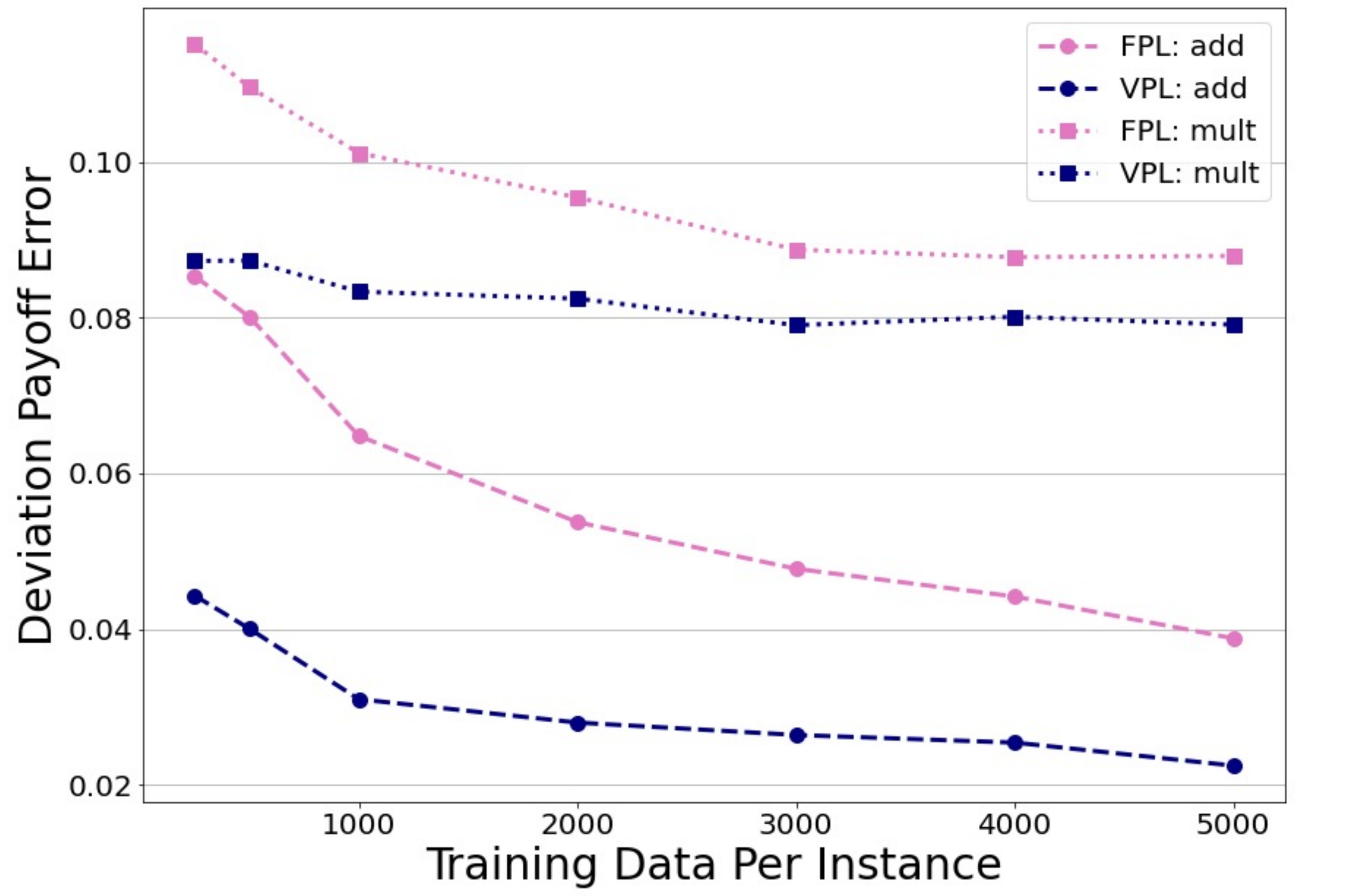


ϵ-Equilibrium Frequency Metric



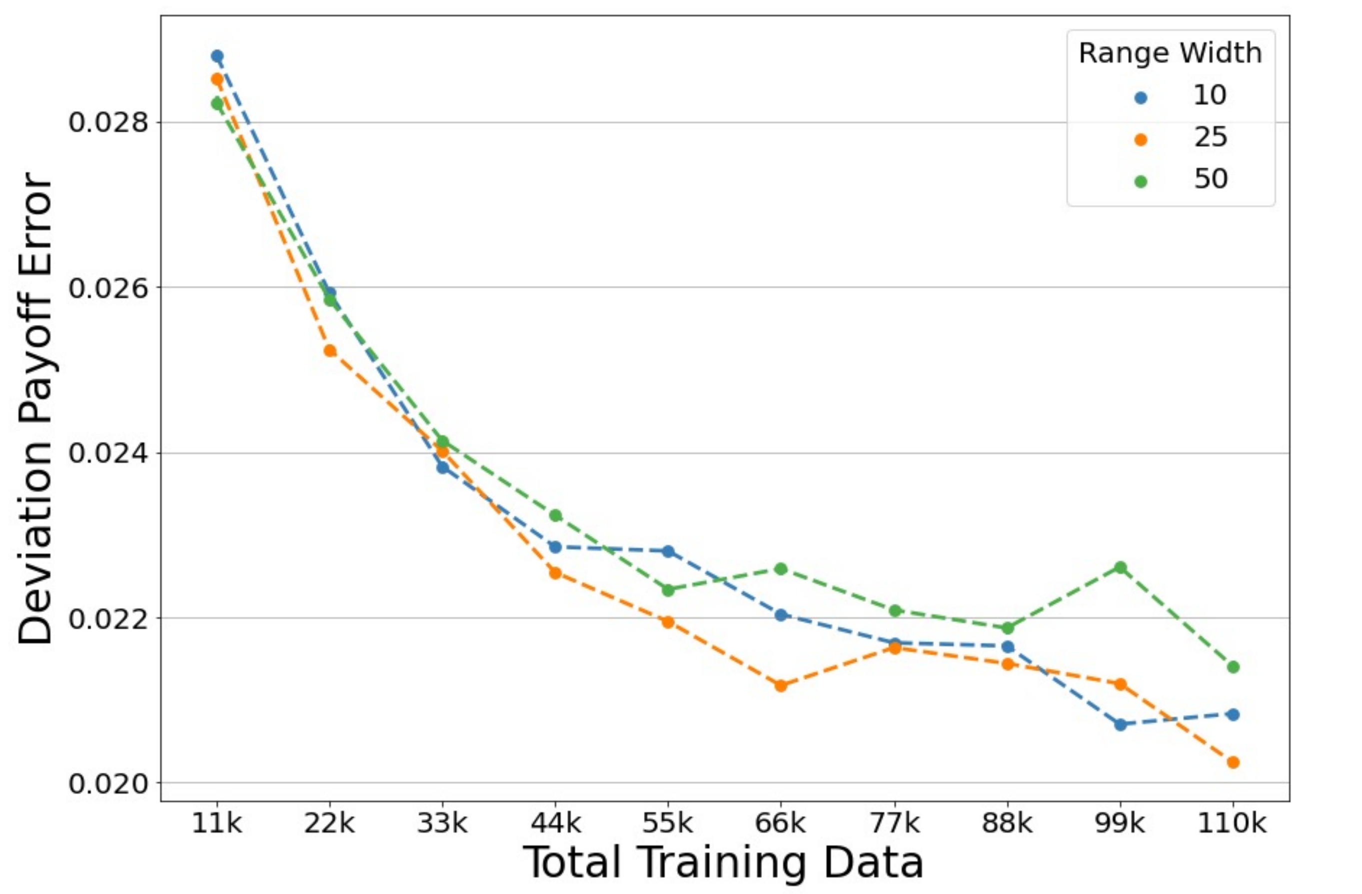
Comparison to Existing Work

Variable-Parameter Learning (VPL) outperforms Fixed-Parameter Learning (FPL) [3] on random additive and multiplicative sine games with 90 to 100 players, given the same training data per instance.



Scalability Experiment

Given the same amount of total training data, the deviation payoff errors are roughly the same for the three models with different player-count range widths. This suggests our model is still the better approach, even as the width of the player-count range increases.



References

[1] Pranav Dandekar, Ashish Goel, Michael P. Wellman, and Bryce Wiedenbeck. 2015. Strategic Formation of Credit Networks. *ACM Trans. Internet Techn.* 15, 1 (2015), 3:1-3:41.
[2] Katherine Mayo and Michael P. Wellman. 2021. A Strategic Analysis of Portfolio Compression. In *Proceedings of the 2nd ACM International Conference on AI in Finance.* Association for Computing Machinery, New York, NY, USA, 8 pages.
[3] Sam Sokota, Caleb Ho, and Bryce Wiedenbeck. 2019. Learning Deviation Payoffs in Simulation-Based Games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 2173-2180.
[4] Elaine Wah and Michael P. Wellman. 2016. Latency arbitrage in fragmented financial markets: A strategic agent-based analysis. *Algorithmic Finance* 5, 3-4, 69-93.