Predicting and Understanding Initial Play

Drew Fudenberg
(MIT)

Annie Liang
(UPenn)

Machine Learning as a Tool for...

better predictions

- learn mapping from rich set of features into prediction of an outcome of interest
- methods often “black box”

better understanding

- can we use machine learning to identify new patterns?
- build on existing models? (e.g. add 1-2 parameters that substantially improve predictive accuracy)
- machine learning as a complement to model building

Machine Learning as a Tool for...

better predictions

- learn mapping from rich set of features into prediction of an outcome of interest
- methods often "black box"

better understanding

- can we use machine learning to identify new patterns?
- build on existing models? (e.g. add 1-2 parameters that substantially improve predictive accuracy)
- machine learning as a complement to model building

our domain: predicting initial play in games

Prediction Task

- prediction of play in 3×3 matrix games

	a_1	a_2	a_3
a_1	$25, 25$	$30, 40$	$100, 31$
a_2	$40, 30$	$45, 45$	$65, 0$
a_3	$31, 100$	$0, 65$	$40, 40$

- given a payoff matrix, can we predict the action most frequently chosen by the row player?
- assess accuracy using correct classification rate
  - in what fraction of games is the predicted modal action the actual modal action in observed play?
- always report tenfold cross-validated prediction accuracies (out-of-sample tests)

This Paper

can we use machine learning to improve understanding beyond these models, and to develop interpretable extensions?

This Paper

can we use machine learning to improve understanding beyond these models, and to develop interpretable extensions?

1. look where ML predicts well and existing models don't
   - opportunity for improvements of existing models
   - in our domain, identify from these instances a single parameter extension to best model

This Paper

can we use machine learning to improve understanding beyond these models, and to develop interpretable extensions?

1. look where ML predicts well and existing models don't
   - opportunity for improvements of existing models
   - in our domain, identify from these instances a single parameter extension to best model
2. "algorithmic game generation" to break our best model
   - generate new cases where best model performs poorly

This Paper

can we use machine learning to improve understanding beyond these models, and to develop interpretable extensions?

1. look where ML predicts well and existing models don't
   - opportunity for improvements of existing models
   - in our domain, identify from these instances a single parameter extension to best model
2. "algorithmic game generation" to break our best model
   - generate new cases where best model performs poorly
3. use ML to choose between models
   - build "hybrid model" that chooses which of two economic/behavioral models to use based on the game

Part I:

Use ML to Identify Parametric Extensions of Existing Models

Data Set

- 86 symmetric 3x3 games
- meta data-set aggregated over 6 lab experiments
  - original sources: Stahl-Wilson [94,95], Stahl-Haruvy [07,08], Haruvy-Stahl-Wilson [01], Rogers-Palfrey-Camerer [09]
- 40-147 observations per game
- (data set due to Kevin Leyton-Brown and James Wright)

Model-Based Predictions

- Uniform Nash: choose uniformly at random from actions consistent with NE.
- Poisson Cognitive Hierarchy Model (PCHM): (Camerer, Cho, and Hong [2004])
    - Level 0: $(1/3, 1/3, 1/3)$
    - Level 1 best responds to level 0
    - Level $k \geq 2$ plays best response to a weighted sum of play of levels 0 through $k-1$
    - Weights correspond to Poisson distribution with rate parameter τ, estimated from the data.

  Predict mode of distribution.

Models Improve on Predictive Accuracy

	Accuracy
Guess at random	0.33
Ideal prediction	1

Models Improve on Predictive Accuracy

	Accuracy
Guess at random	0.33
Uniform Nash	0.42
	(0.05)
Level-1/PCHM	0.72
	(0.04)
Ideal prediction	1

Machine Learning: Prediction Based on Game Features

1. identify each game with a feature vector

   e.g. for each action:

   - is it part of a pure-strategy NE?
   - is it part of a pure-strategy Pareto-dominant NE? (payoffs Pareto-dominate payoffs in all other pure-strategy NE).
   - is it level-k for $k \in \{1, 2, \ldots, 7\}$?
   - is it part of a profile that maximizes sum of player payoffs?

2. train a decision tree ensemble to predict play given features

Algorithm Improves Further

	Accuracy
Guess at random	0.33
Uniform Nash	0.42
	(0.05)
Level-1/PCHM	0.72
	(0.04)
Decision Tree Ensemble	0.77
	(0.02)
Ideal prediction	1

Identifying Predictable Structure Beyond Level 1

- look at games where the modal action is correctly predicted by our algorithm but not by level-1
- example game:

	a_1	a_2	a_3	payoff against uniform
a_1	$25, 25$	$30, 40$	$100, 31$	51.6
a_2	$40, 30$	$45, 45$	$65, 0$	50
a_3	$31, 100$	$0, 65$	$40, 40$	23.6

- level-1 action is a_1, but a_2 played most frequently

Identifying Predictable Structure Beyond Level 1

- look at games where the modal action is correctly predicted by our algorithm but not by level-1
- example game:

	a_1	a_2	a_3	payoff against uniform
a_1	$25, 25$	$30, 40$	$100, 31$	51.6
a_2	$40, 30$	$45, 45$	$65, 0$	50
a_3	$31, 100$	$0, 65$	$40, 40$	23.6

- level-1 action is a_1, but a_2 played most frequently
- modify level-1 by assuming participants have utility $u(x) = x^\alpha$; this adds one parameter to the level 1 model

Extension of Level-1 Achieves Performance of Algorithm

	Accuracy
Level-1	0.72 (0.04)
Decision Tree Ensemble	0.77 (0.02)

Extension of Level-1 Achieves Performance of Algorithm

	Accuracy
Level-1	0.72 (0.04)
Decision Tree Ensemble	0.77 (0.02)
Level-1(α)	0.79 (0.04)

- black box algorithm can lead to interpretable parametric models

Takeaways from Lab Games

- could stop here and conclude that level-1(α) is an almost complete model of play
- but set of lab games is very special

Takeaways from Lab Games

- could stop here and conclude that level-1(α) is an almost complete model of play
- but set of lab games is very special
- is the performance of level-1(α) due to idiosyncratic properties of the lab games?
- need observed play on new games. question is: which games?

Takeaways from Lab Games

- could stop here and conclude that level-1(α) is an almost complete model of play
- but set of lab games is very special
- is the performance of level-1(α) due to idiosyncratic properties of the lab games?
- need observed play on new games. question is: which games?
- first try generating payoffs uniformly at random, but level-1(α) performs even better on those games!
- need more sophisticated procedure

Part II:

Use ML to Find Cases That Break Our Best Model

Step 1: Train Algorithm to Predict Freq of Level-1 Play

	a_1	a_2	a_3
a_1	$40, 40$	$10, 20$	$70, 30$
a_2	$20, 10$	$80, 80$	$0, 100$
a_3	$30, 70$	$100, 0$	$60, 60$

freq. of level-1(α) action: 73%

	a_1	a_2	a_3
a_1	$20, 20$	$0, 60$	$100, 0$
a_2	$60, 0$	$20, 20$	$0, 60$
a_3	$0, 100$	$60, 0$	$40, 40$

freq. of level-1(α) action: 65%

	a_1	a_2	a_3
a_1	$20, 20$	$30, 40$	$100, 30$
a_2	$40, 30$	$40, 40$	$60, 0$
a_3	$30, 100$	$0, 60$	$40, 40$

freq. of level-1(α) action: 35%

learn a map from **payoff matrices** into **prediction of frequency of play of level-1(α) action**

Step 2: Generate New Games, Predict Freq of Level-1 Play

learn a map from **payoff matrices** into **prediction of frequency of play of level-1 action**

Step 2: Generate New Games, Predict Freq of Level-1 Play

	a_1	a_2	a_3
a_1	$90, 90$	$30, 80$	$45, 30$
a_2	$80, 30$	$55, 55$	$37, 5$
a_3	$30, 45$	$5, 37$	$70, 70$

learn a map from **payoff matrices** into **prediction of frequency of play of level-1 action**

Step 2: Generate New Games, Predict Freq of Level-1 Play

	a_1	a_2	a_3
a_1	$90, 90$	$30, 80$	$45, 30$
a_2	$80, 30$	$55, 55$	$37, 5$
a_3	$30, 45$	$5, 37$	$70, 70$

	a_1	a_2	a_3
a_1	$70, 70$	$45, 30$	$40, 35$
a_2	$30, 45$	$53, 53$	$93, 31$
a_3	$35, 40$	$31, 93$	$10, 10$

learn a map from **payoff matrices** into **prediction of frequency of play of level-1 action**

Step 2: Generate New Games, Predict Freq of Level-1 Play

	a_1	a_2	a_3
a_1	$90, 90$	$30, 80$	$45, 30$
a_2	$80, 30$	$55, 55$	$37, 5$
a_3	$30, 45$	$5, 37$	$70, 70$

	a_1	a_2	a_3
a_1	$70, 70$	$45, 30$	$40, 35$
a_2	$30, 45$	$53, 53$	$93, 31$
a_3	$35, 40$	$31, 93$	$10, 10$

	a_1	a_2	a_3
a_1	$60, 60$	$40, 40$	$51, 40$
a_2	$40, 40$	$80, 80$	$35, 10$
a_3	$40, 51$	$10, 35$	$100, 100$

learn a map from **payoff matrices** into **prediction of frequency of play of level-1 action**

Step 2: Generate New Games, Predict Freq of Level-1 Play

	a_1	a_2	a_3
a_1	$90, 90$	$30, 80$	$45, 30$
a_2	$80, 30$	$55, 55$	$37, 5$
a_3	$30, 45$	$5, 37$	$70, 70$

predicted frequency: 48%

	a_1	a_2	a_3
a_1	$70, 70$	$45, 30$	$40, 35$
a_2	$30, 45$	$53, 53$	$93, 31$
a_3	$35, 40$	$31, 93$	$10, 10$

predicted frequency: 56%

	a_1	a_2	a_3
a_1	$60, 60$	$40, 40$	$51, 40$
a_2	$40, 40$	$80, 80$	$35, 10$
a_3	$40, 51$	$10, 35$	$100, 100$

predicted frequency: 46%

learn a map from **payoff matrices** into **prediction of frequency of play of level-1 action**

Step 3: Redraw Games with High Predicted Frequencies

	a_1	a_2	a_3
a_1	$90, 90$	$30, 80$	$45, 30$
a_2	$80, 30$	$55, 55$	$37, 5$
a_3	$30, 45$	$5, 37$	$70, 70$

predicted frequency: 48%

	a_1	a_2	a_3
a_1	$60, 60$	$40, 40$	$51, 40$
a_2	$40, 40$	$80, 80$	$35, 10$
a_3	$40, 51$	$10, 35$	$100, 100$

predicted frequency: 46%

learn a map from **payoff matrices** into **prediction of frequency of play of level-1 action**

Step 3: Redraw Games with High Predicted Frequencies

$$\begin{array}{cccc} & a_1 & a_2 & a_3 \\ a_1 & 90,90 & 30,80 & 45,30 \\ a_2 & 80,30 & 55,55 & 37,5 \\ a_3 & 30,45 & 5,37 & 70,70 \end{array}$$

predicted frequency: 48%

$$\begin{array}{cccc} & a_1 & a_2 & a_3 \\ a_1 & 50,50 & 70,40 & 50,15 \\ a_2 & 40,70 & 15,15 & 88,100 \\ a_3 & 15,50 & 100,88 & 58,58 \end{array}$$

$$\begin{array}{cccc} & a_1 & a_2 & a_3 \\ a_1 & 60,60 & 40,40 & 51,40 \\ a_2 & 40,40 & 80,80 & 35,10 \\ a_3 & 40,51 & 10,35 & 100,100 \end{array}$$

predicted frequency: 46%

learn a map from **payoff matrices** into **prediction of frequency of play of level-1 action**

Step 3: Redraw Games with High Predicted Frequencies

	a_1	a_2	a_3	
a_1	$90, 90$	$30, 80$	$45, 30$	
a_2	$80, 30$	$55, 55$	$37, 5$	predicted frequency: 48%
a_3	$30, 45$	$5, 37$	$70, 70$	

	a_1	a_2	a_3	
a_1	$50, 50$	$70, 40$	$50, 15$	
a_2	$40, 70$	$15, 15$	$88, 100$	predicted frequency: 47%
a_3	$15, 50$	$100, 88$	$58, 58$	

	a_1	a_2	a_3	
a_1	$60, 60$	$40, 40$	$51, 40$	
a_2	$40, 40$	$80, 80$	$35, 10$	predicted frequency: 46%
a_3	$40, 51$	$10, 35$	$100, 100$	

learn a map from **payoff matrices** into **prediction of frequency of play of level-1 action**

Performance on New Games

	Accuracy
Guess at random	0.33
Level-1	0.36
	(0.01)
Level-1(α)	0.41
	(0.05)
Ideal prediction	1

- algorithmically designed games succeed in being poor matches for level-1.

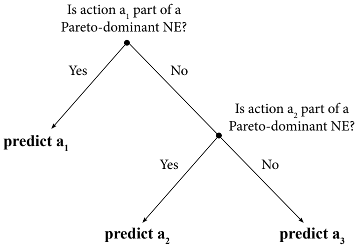
Performance on New Games

	Accuracy
Guess at random	0.33
Level-1	0.36
	(0.01)
Level-1(α)	0.41
	(0.05)
Decision Tree Ensemble	0.73
	(0.02)
Ideal prediction	1

- algorithmically designed games succeed in being poor matches for level-1.

Best 2-split Decision Tree

decision tree ensemble is hard to interpret, but best 2-split decision tree is not:



motivates:

Pareto-Dominant NE (PDNE): predict at random from actions consistent with PDNE, otherwise predict at random.

Performance of PDNE

	Accuracy
Guess at random	0.33
Level-1	0.36
	(0.01)
Level-1(α)	0.41
	(0.05)
Decision Tree Ensemble	0.73
	(0.02)
Ideal prediction	1

Performance of PDNE

	Accuracy
Guess at random	0.33
Level-1	0.36
	(0.01)
Level-1(α)	0.41
	(0.05)
PDNE	0.65
	(0.02)
Decision Tree Ensemble	0.73
	(0.02)
Ideal prediction	1

- PDNE performs very well on this data set (substantially outperforms level-1(α))

Performance of PDNE

	Accuracy
Guess at random	0.33
Level-1	0.36
	(0.01)
Level-1(α)	0.41
	(0.05)
PDNE	0.65
	(0.02)
Decision Tree Ensemble	0.73
	(0.02)
Ideal prediction	1

- PDNE performs very well on this data set (substantially outperforms level-1(α))
- but don't want to rank PDNE and level-1(α)

Aggregation of All Games (Lab, Randomly-Generated, Algorithmically-Generated)

	Accuracy
Guess at random	0.33
Level-1(α)	0.68
	(0.02)
PDNE	0.56
	(0.02)
Ideal prediction	1

- neither level-1(α) nor PDNE is the "full story"
- can we improve upon them by predicting when PDNE is a good model of play, and when level-1(α) is better?

Part III:

Use ML to Choose Which Model to Use for Prediction

Hybrid Models

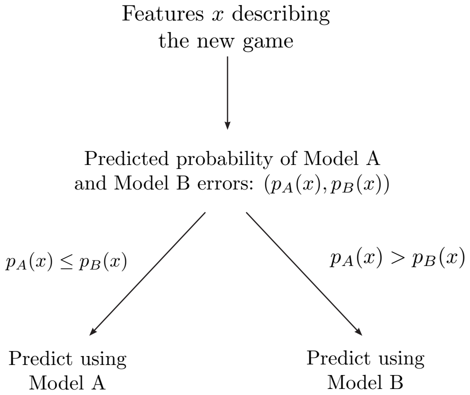
approach: use machine learning to choose between level-1(α) and PDNE game-by-game

- related: mixture of experts (Masoudnia and Ebrahimpour, 2014), model trees (Quinlan,1992), forecast combinations (Timmerman, 2006)

on the training data:

- use each model to predict the modal action in each game $\longrightarrow$ binary error vectors
- fit a decision tree for predicting the probability that each model predicts correctly

Hybrid Model: Illustration



Level-1(α) and PDNE hybrid model

Performance accuracies below are shown for the set of all games:

	Accuracy
Guess at random	0.33
Level-1(α)	0.68 (0.02)
PDNE	0.56 (0.02)
Ideal prediction	1

Hybrid model improves predictive performance over both component models. more

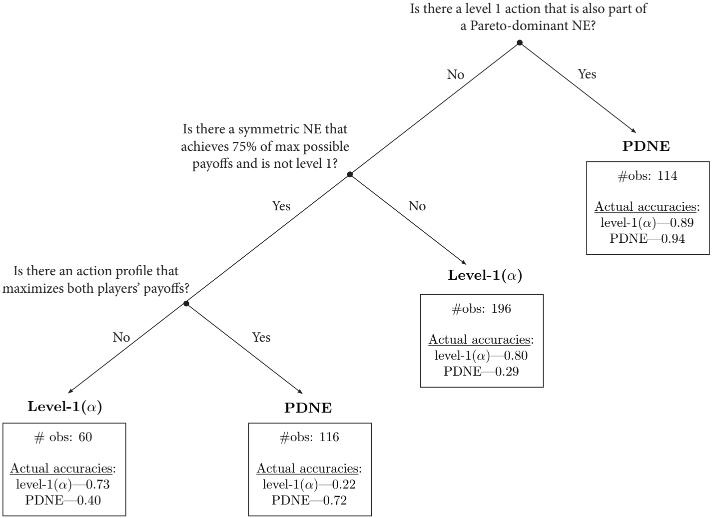
Level-1(α) and PDNE hybrid model

Performance accuracies below are shown for the set of all games:

	Accuracy
Guess at random	0.33
Level-1(α)	0.68 (0.02)
PDNE	0.56 (0.02)
Level-1(α)+PDNE	0.79 (0.03)
Ideal prediction	1

Hybrid model improves predictive performance over both component models. more

Assignment of Models



Example Game Assigned to PDNE

one class of games assigned to PDNE:

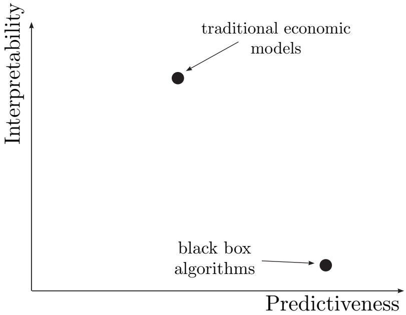
- there is a very good Nash equilibrium (Pareto-dominant, symmetric, yields maximal payoffs for both players), and
- level-1 action is not part of that NE

for example:

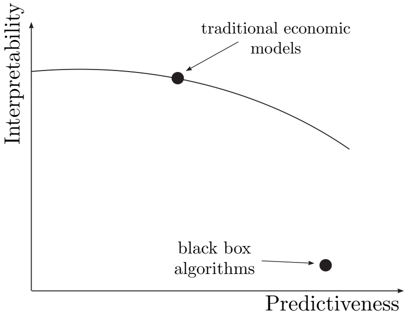
	a_1	a_2	a_3	frequency of play
a_1	$90, 90$	$30, 80$	$45, 30$	72%
a_2	$80, 30$	$55, 55$	$37, 5$	28%
a_3	$30, 45$	$5, 37$	$70, 70$	0%

action a_2 is level-1(α), but action profile (a_1, a_1) is appealing

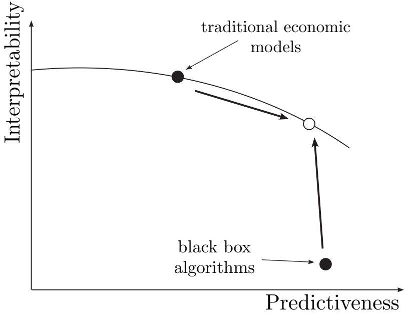
Conclusion


Interpretability
traditional economic models
black box algorithms
Predictiveness

Conclusion

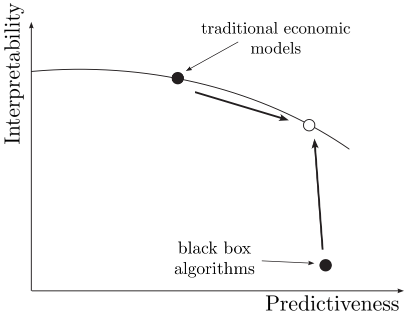

Interpretability
traditional economic
models
black box
algorithms
Predictiveness

Conclusion



Interpretability
traditional economic models
black box algorithms
Predictiveness

Conclusion



can potentially use approaches from ML

- not only to improve predictions in a problem domain
- but also to improve our understanding of it, and to develop simple and portable improvements on existing models