Modeling Human Strategic Behavior from a Machine Learning Perspective

Kevin Leyton-Brown

Professor of Computer Science

University of British Columbia

Canada CIFAR AI Chair, Alberta Machine Intelligence Institute

THE UNIVERSITY OF BRITISH COLUMBIA

CAIDA

amii

Thanks to my many collaborators!

- **James Wright** | *most of the projects in this talk*

- Greg d'Eon
- Sophie Greenwood

| *loss functions*


- Jason Hartford | *deep learning for behavioral GT*

If we didn't have game theory, we'd need to invent it

- A general mathematical approach for reasoning about **arbitrary strategic situations**
- Given predictions about counterfactual play, we can **design mechanisms** that optimize properties of interest

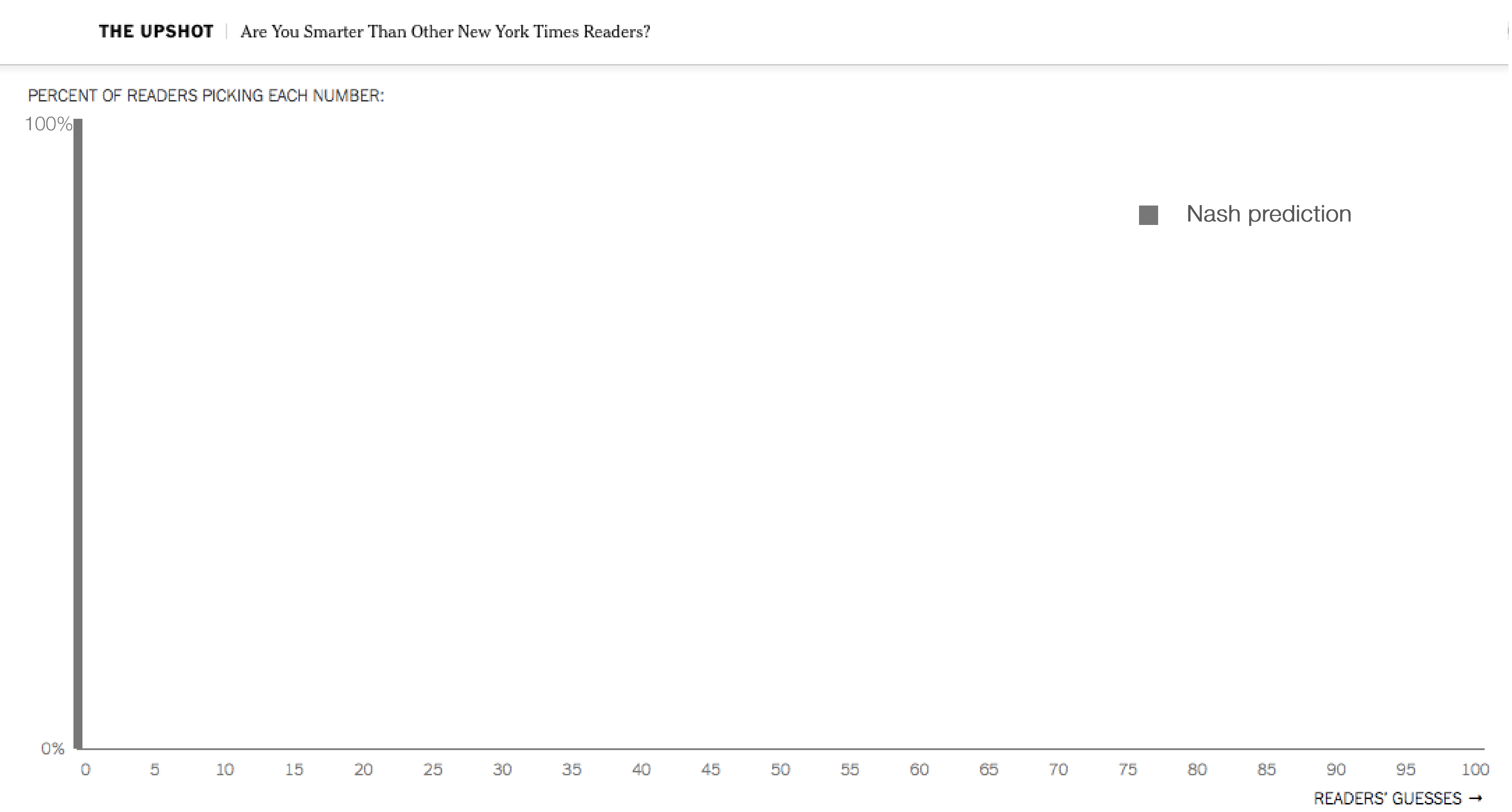
- The catch: design quality depends on **accuracy of the predictions**
- Let's consider a prediction that is among the strongest made by game theory: **unique, dominance-solvable Nash equilibrium**

Example: Beauty Contest Game

Pick a number from 0 to 100

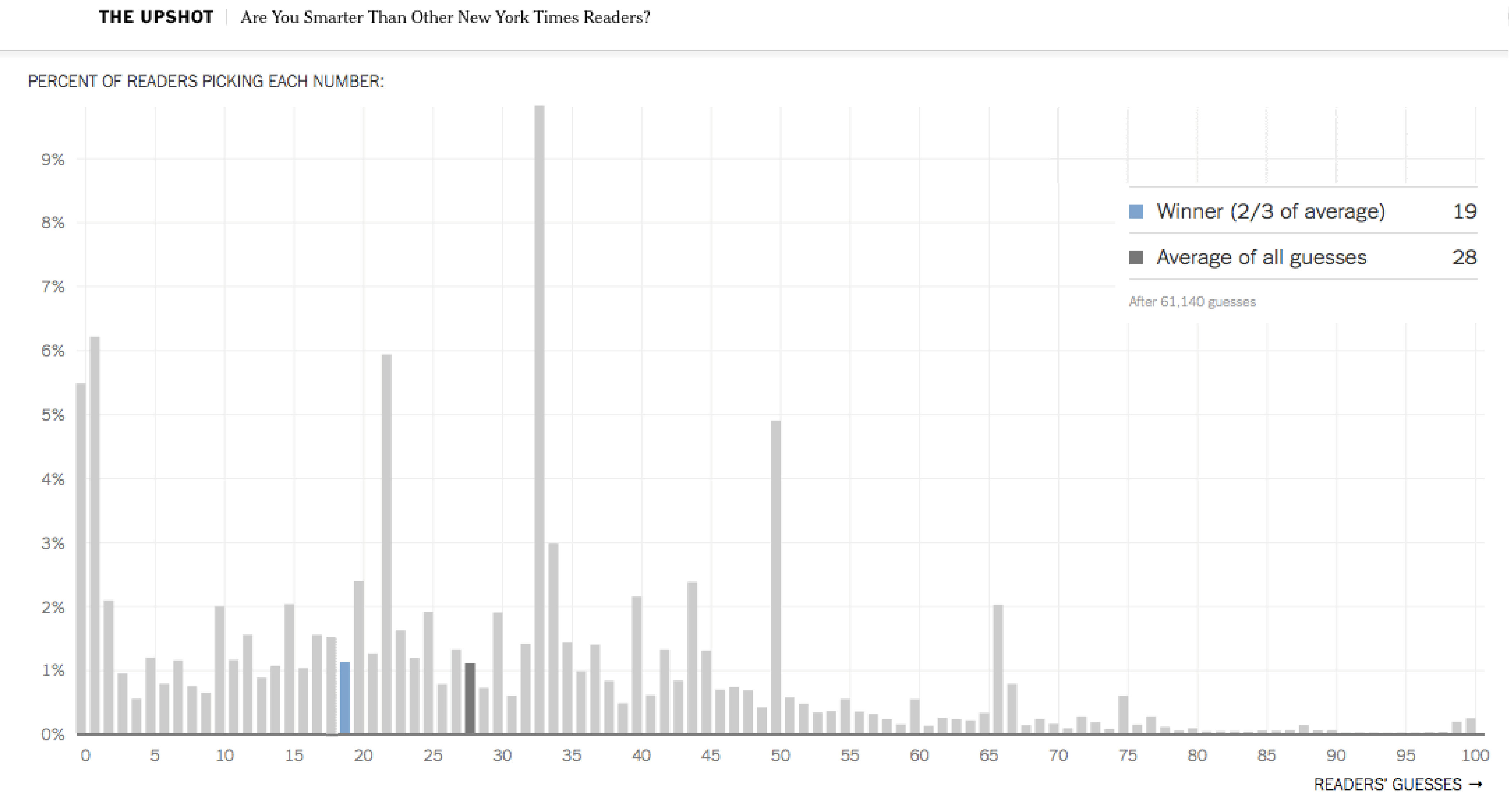
The integer closest to **two-thirds of the average of all numbers picked** wins

“Are You Smarter Than 61,140 Other New York Times Readers?”



Source: http://www.nytimes.com/interactive/2015/08/13/upshot/are-you-smarter-than-other-new-york-times-readers.html

“Are You Smarter Than 61,140 Other New York Times Readers?”

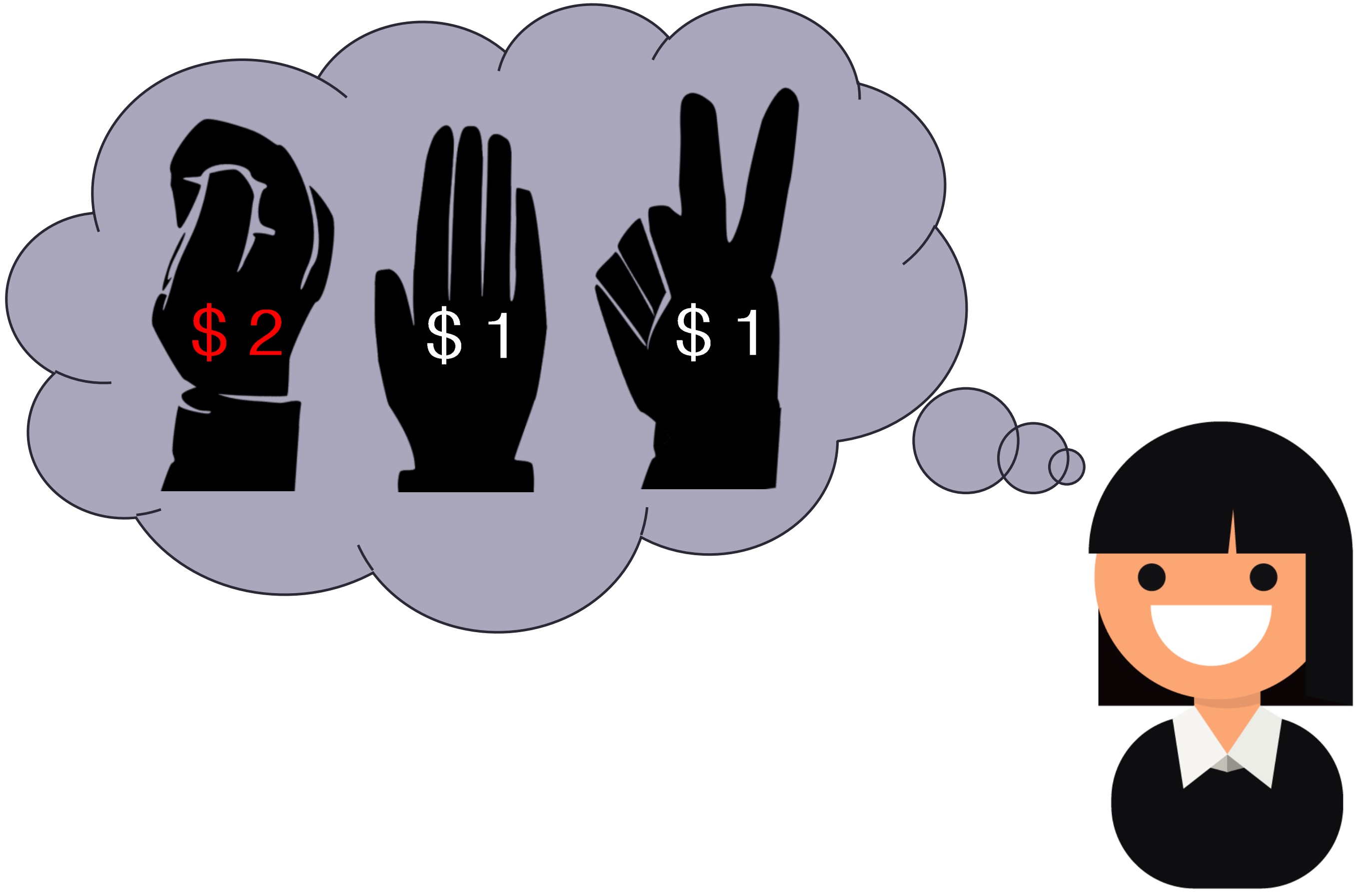


Source: http://www.nytimes.com/interactive/2015/08/13/upshot/are-you-smarter-than-other-new-york-times-readers.html

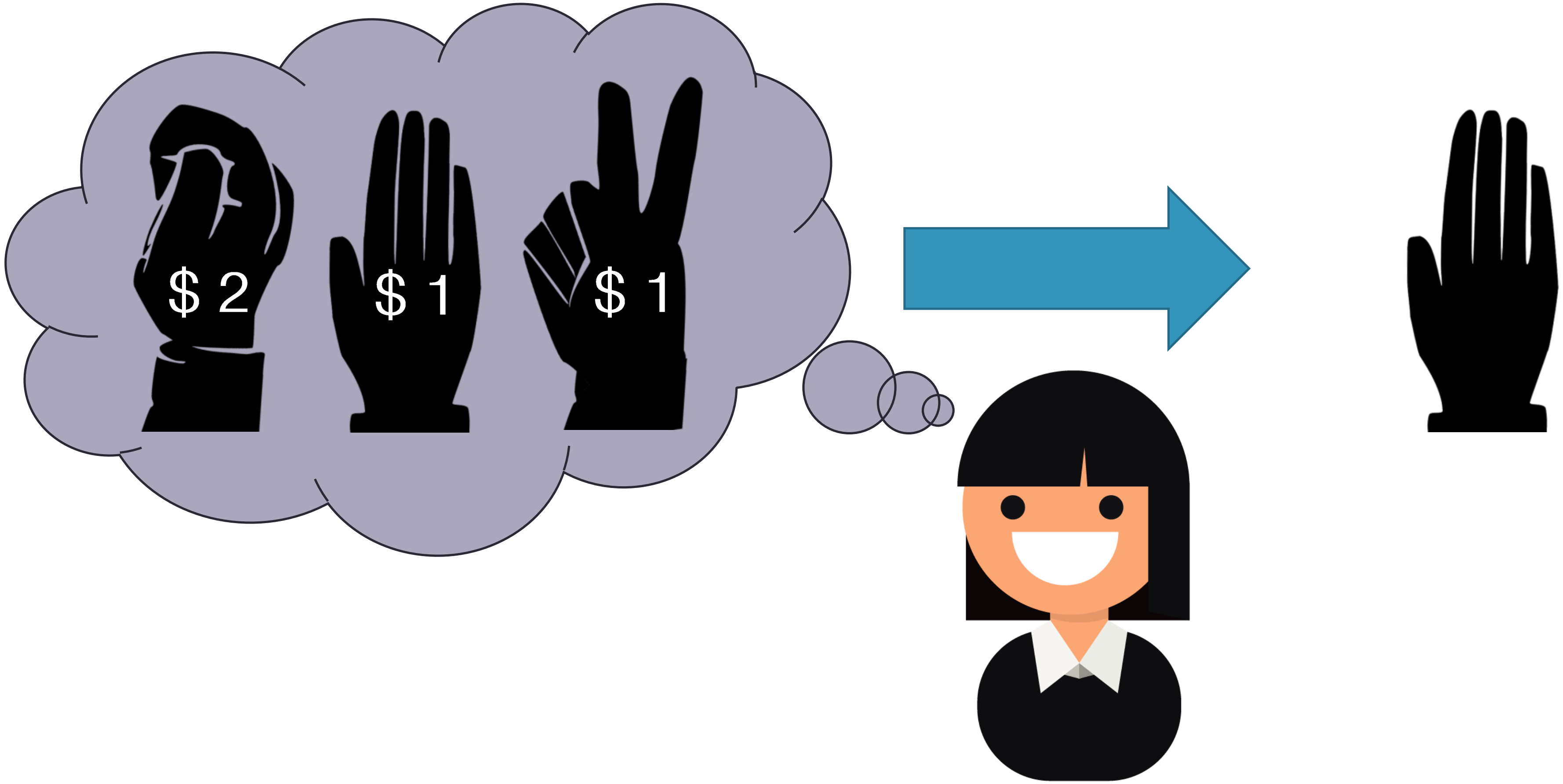
Limitations of perfect rationality

- Many of game theory's recommendations are **counterintuitive**
- Clearly the world is not populated only by **perfectly rational agents**
- To make good predictions about the play of unsophisticated humans (and hence, e.g., to design mechanisms they will use), we need a model of **human behavior**

Two player simultaneous-move games



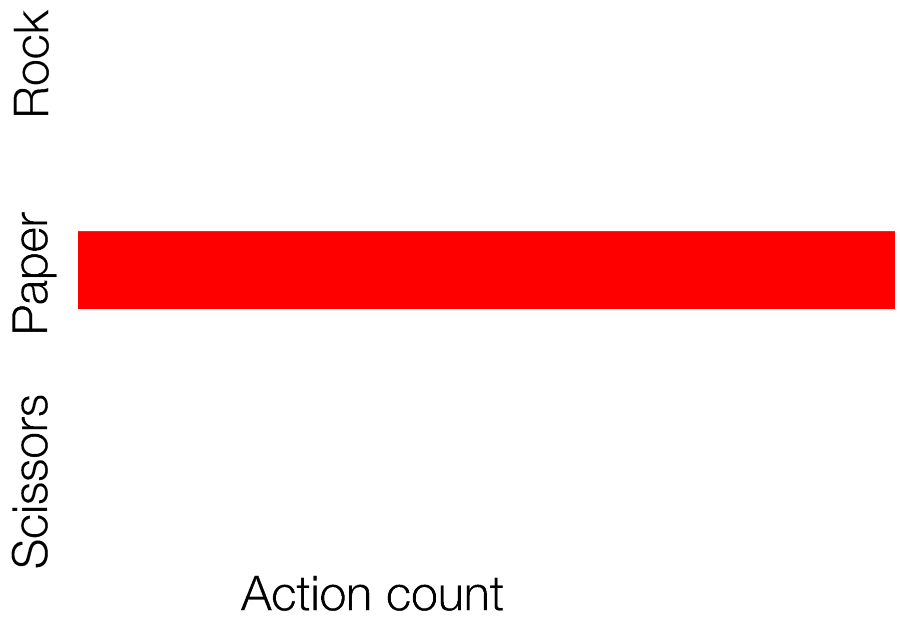
Two player simultaneous-move games



Two player simultaneous-move games

Column player's actions

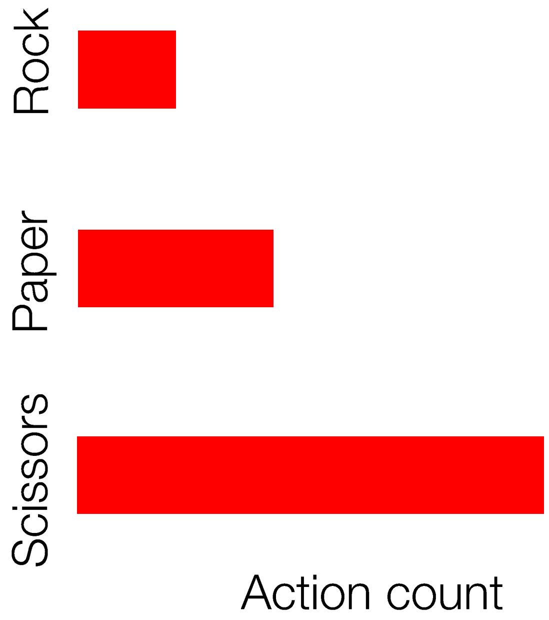
Row player's actions	Rock	Paper	Scissors
Rock	0, 0	-1, 1	2, -2
Paper	1, -1	0, 0	-1, 1
Scissors	-2, 2	1, -1	0, 0



Two player simultaneous-move games

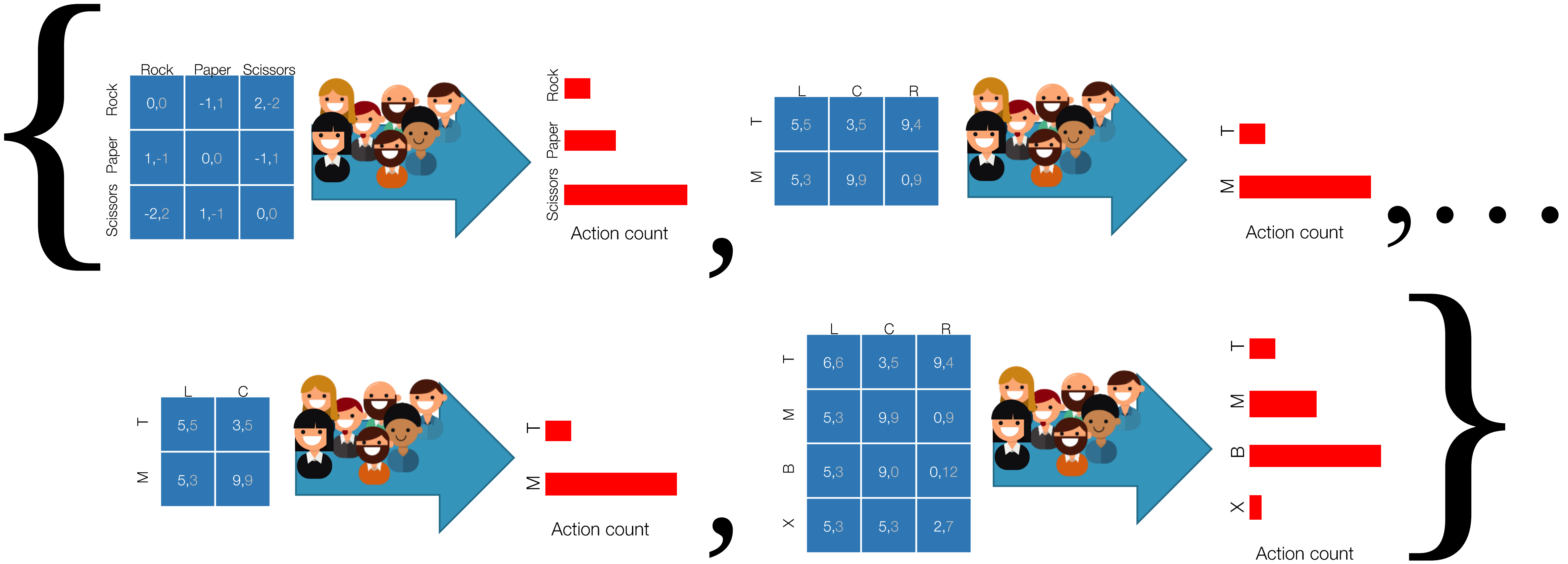
Column player's actions

Row player's actions	Rock	Paper	Scissors
Rock	0, 0	-1, 1	2, -2
Paper	1, -1	0, 0	-1, 1
Scissors	-2, 2	1, -1	0, 0



Learning problem

Given a dataset of **games**, each with observed **action counts**:

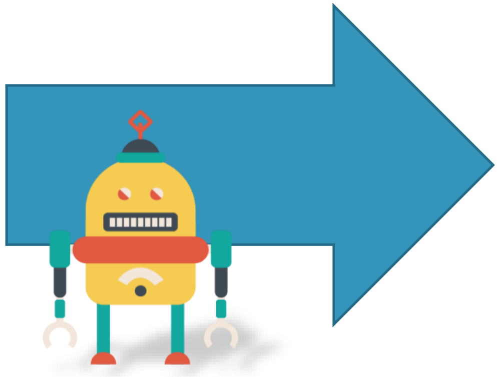


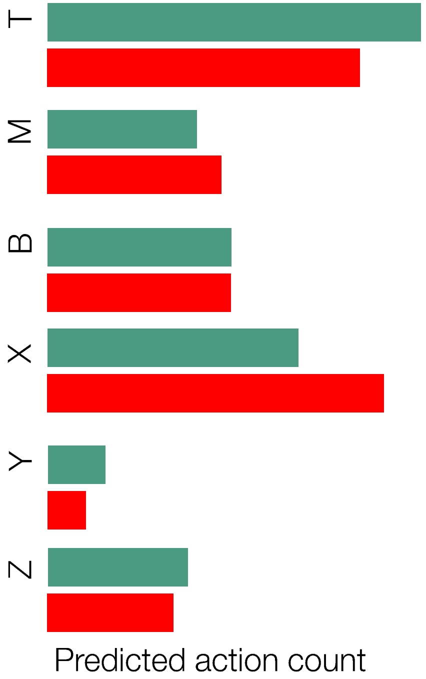
…learn a model that predicts players' **distribution** over actions

Learning problem

We will evaluate a learned model by assessing how well it **predicts the distribution of play** across human players from the same population **on arbitrary games not previously seen** when fitting the model

	L	C	R
T	6,6	3,5	9,4
M	5,3	9,9	0,9
B	5,3	9,0	0,12
X	5,3	5,3	2,7
Y	0,-1	10,-8	0,0
Z	7,12	9,-8	0,0





Data

Name	Source	Games	n
SW94	[Stahl and Wilson, 1994]	10	400
SW95	[Stahl and Wilson, 1995]	12	576
CGCB98	[Costa-Gomes et al., 1998]	18	1296
GH01	[Goeree and Holt, 2001]	10	500
CVH03	[Cooper and Van Huyck, 2003]	8	2992
RPC09	[Rogers et al., 2009]	17	1210
HSW01	[Haruvy et al., 2001]	15	869
HS07	[Haruvy and Stahl, 2007]	20	2940
SH08	[Stahl and Haruvy, 2008]	18	1288
Combo9	400 samples from each	128	3600

Evaluating models

- We randomly partition our data into **two different data sets**:

$$\mathcal{D} = \mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{test}}$$

- We choose parameter value(s) that **minimize loss** on the training data:

$$\theta^* = \text{argmin}_\theta \, L(\mathcal{D}_{\text{train}} | \mathcal{M}, \theta)$$

- We score the performance of a model by its loss on the **test data**:

$$L(\mathcal{D}_{\text{test}} | \mathcal{M}, \theta^*)$$

- To reduce variance, we **repeat this process multiple times** with different random partitions, averaging the results

Which loss function should we use?

- Many loss functions have been used in the literature:
  - negative log likelihood [e.g., McKelvey and Palfrey, 1992; **Wright and Leyton-Brown, 2010+**]
  - error rate [e.g., Fudenberg and Liang, 2019]
  - Brier score [e.g., Camerer, Ho, and Chong, 2004; Golman, Bhatia, and Kane, 2019]
  - squared L2 error (mean squared error) [e.g., Plonsky et al., 2019]
- Today: I'll follow our prior work and report negative log-likelihood. Some drawbacks:
  - units are **uninterpretable**: scales with the number of samples and actions/game
  - no measure of **how close** we are to perfect prediction
- Other losses can be problematic, too
  - example: error rate is minimized by predicting probability 1 on the modal action

Axioms for loss functions

[d'Eon, Greenwood, Leyton-Brown, Wright: AAAI 2024]

Can we make a **principled argument** for which loss function to use?

We argue that BGT loss functions should satisfy five axioms, falling into two categories:

- Alignment: the loss should induce correct **preferences** over predictions
  - SPA: closer to empirical distribution ⇒ lower loss
  - DPA: closer to true distribution ⇒ lower expected loss

  (both: stronger variants of propriety axioms that work for misspecified functions)
- Interpretability: the loss should represent the **quality** of a prediction
  - EDS: loss independent of number or order of observations
  - CPR: empirical distribution closer to prediction ⇒ lower loss
  - ZM: a perfect prediction gets 0 loss

Revisiting common loss functions

	Alignment				Interpretability		
Loss	SP	SPA	DP	DPA	EDS	CPR	ZM
Negative log-likelihood	✓	⚠	✓	⚠	×	×	×
Error rate	×	×	×	×	✓	×	×
L1 error (MAE)	✓	✓	×	×	✓	✓	✓
Cross-entropy	✓	⚠	✓	⚠	✓	×	×
KL divergence	✓	⚠	✓	⚠	✓	⚠	✓
Brier score	✓	✓	✓	✓	✓	×	×
Squared L2 error (MSE)	✓	✓	✓	✓	✓	✓	✓

- NLL satisfies alignment axioms (since our models put positive probability on every action)
  - but if we were starting our project today, we'd use **squared L2 error**

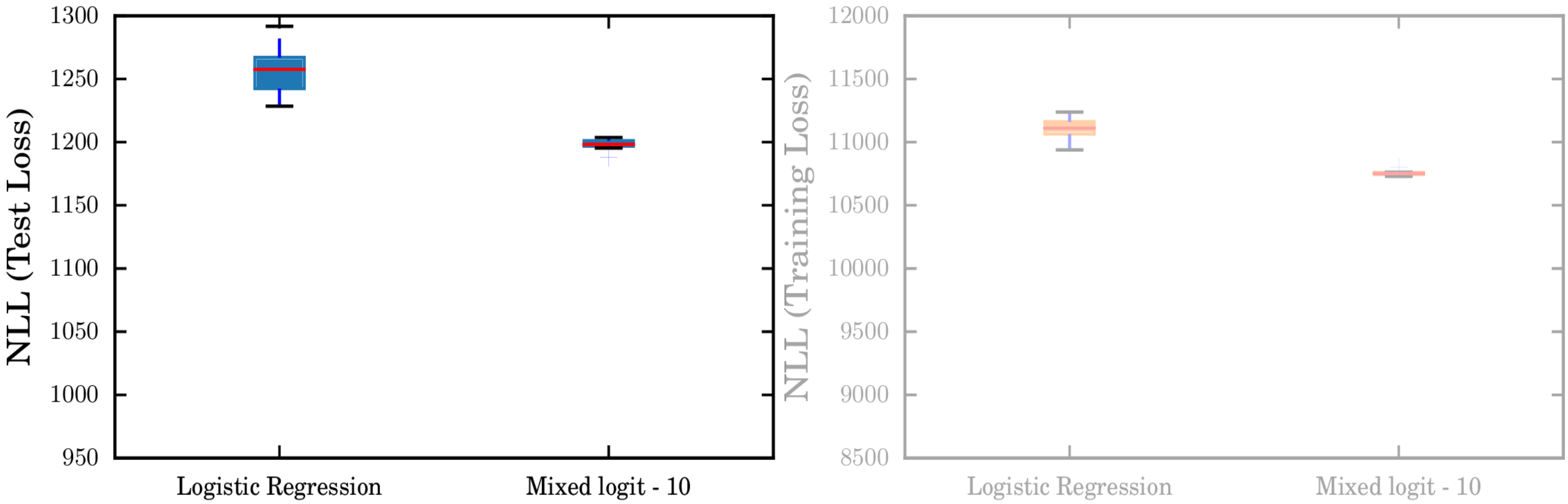
Research Question #1: choice of loss function
- Are there additional axioms that an ideal loss function should satisfy?
- How would our empirical results change if we used a different loss function?
- What more could we learn from our models by using a more interpretable loss?

A Standard Supervised Learning Problem?

- Challenges:
  - not simple classification: must return a **probability distribution**
  - not straightforward density estimation: **distribution size** varies with input
  - ...models are **mappings** from games to probability distributions
- One off-the-shelf idea: **discrete choice**
  - set of choices = row player's actions
  - features = payoffs
  - **logistic regression:** $P(a_i) = \dfrac{e^{\alpha + \sum_j \beta x_{i,j}}}{\sum_i e^{\alpha + \sum_j \beta x_{i,j}}}$
  - **mixed logit model:** $P(a_i) = \sum_{c=1}^{10} s^{(c)} \dfrac{e^{\alpha^{(c)} + \sum_j \beta^{(c)} x_{i,j}}}{\sum_i e^{\alpha^{(c)} + \sum_j \beta^{(c)} x_{i,j}}}, \qquad \sum_{c=1}^{10} s^{(c)} = 1$

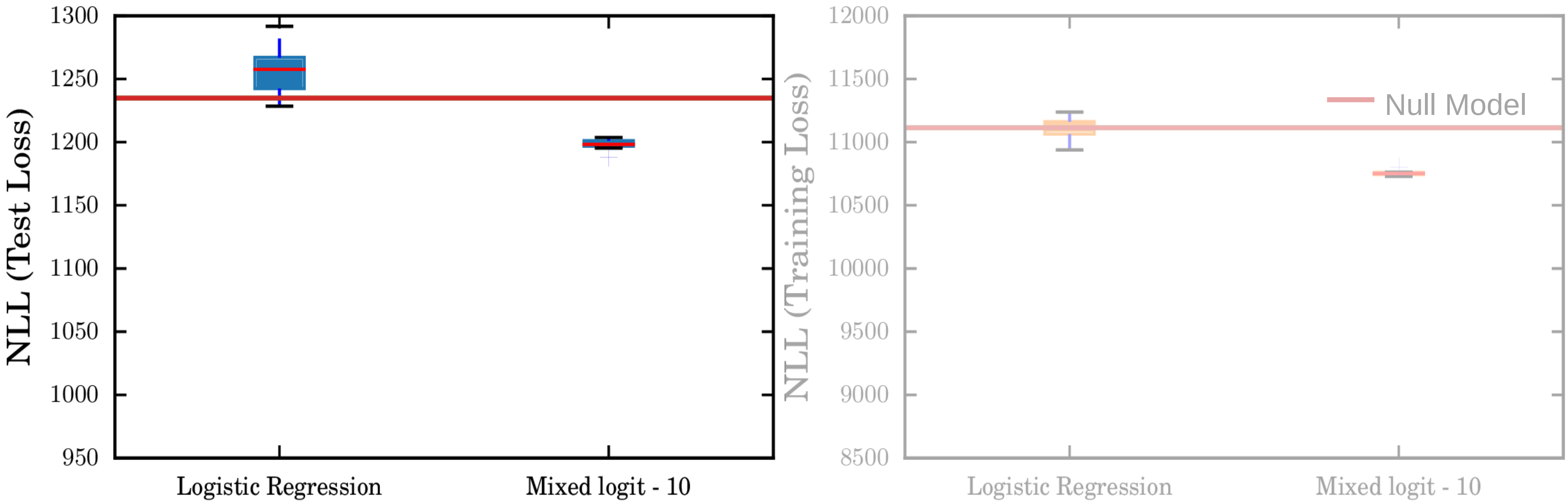
    (e.g., 10 latent classes)

Mixed-logit performance



Is this any good?

Mixed-logit performance



Logistic regression applied to raw payoffs is **worse** than always predicting the **uniform** distribution. **Mixed logit** is not much better…

Lessons from behavioral economics

Behavioral Game Theory has proposed hand-tuned models based on psychological insights:

- Quantal Response Equilibrium [McKelvey & Palfrey 1995]
- Level-k [Costa-Gomes et al. 2001]
- Cognitive Hierarchy [Camerer et al. 2004]
- Noisy introspection [Goeree & Holt 2004]
- Quantal Lk, Quantal CH [Stahl & Wilson 1994; Camerer et al.]

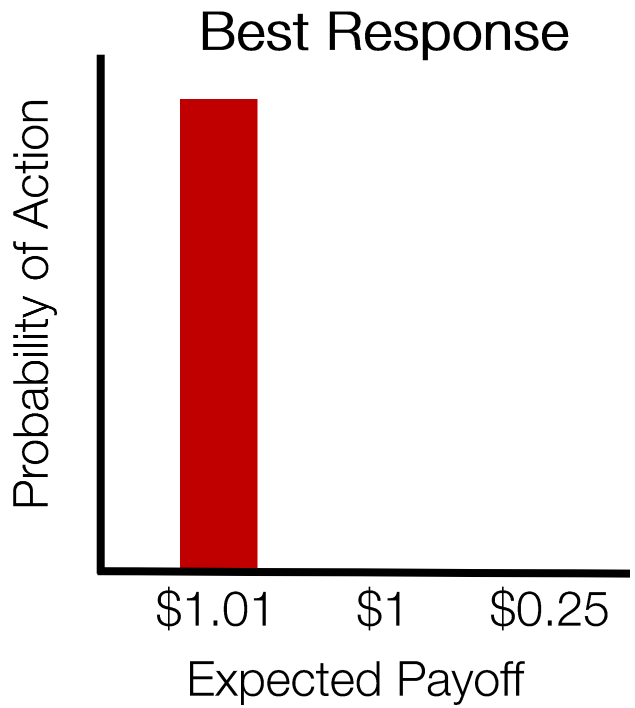
Two key ideas underlie the best performing models [Wright, Leyton-Brown: AAAI 2010; GEB 2017]:

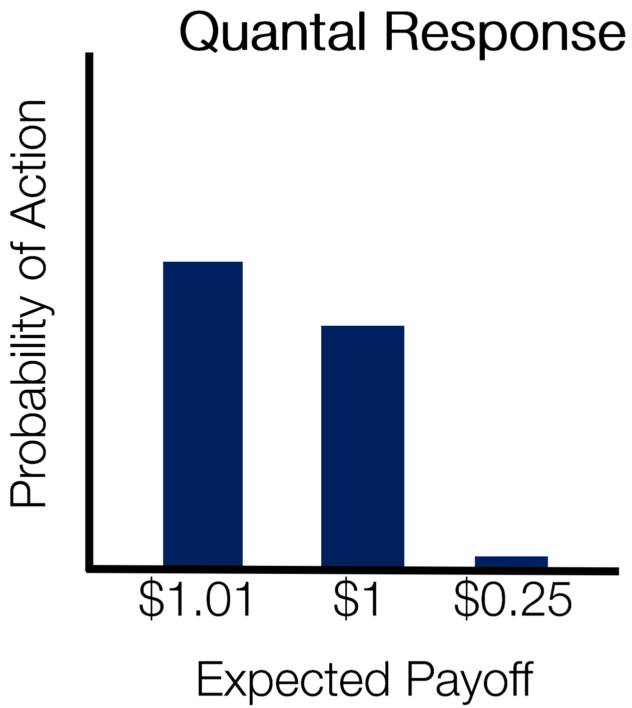
- **Quantal** utility maximization instead of utility maximization
- **Iterative strategic reasoning** instead of equilibrium

Research Question #2: Other Phenomena

- Are there other general psychological insights we should explore?

Quantal utility maximization





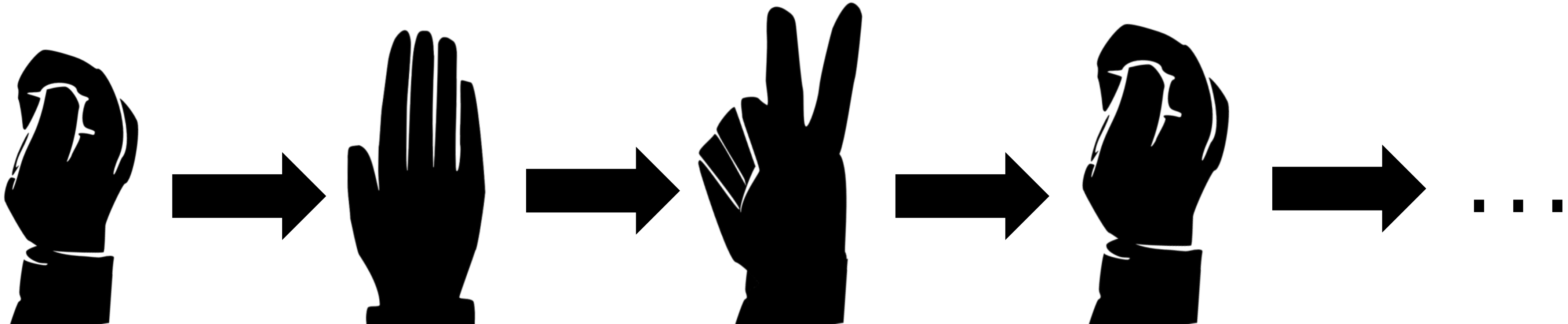
- **Best response**: Maximum utility action is always played
- **Quantal** (“softmax”) **response**: High-utility actions played often, low-utility actions played rarely

Iterative Strategic Reasoning

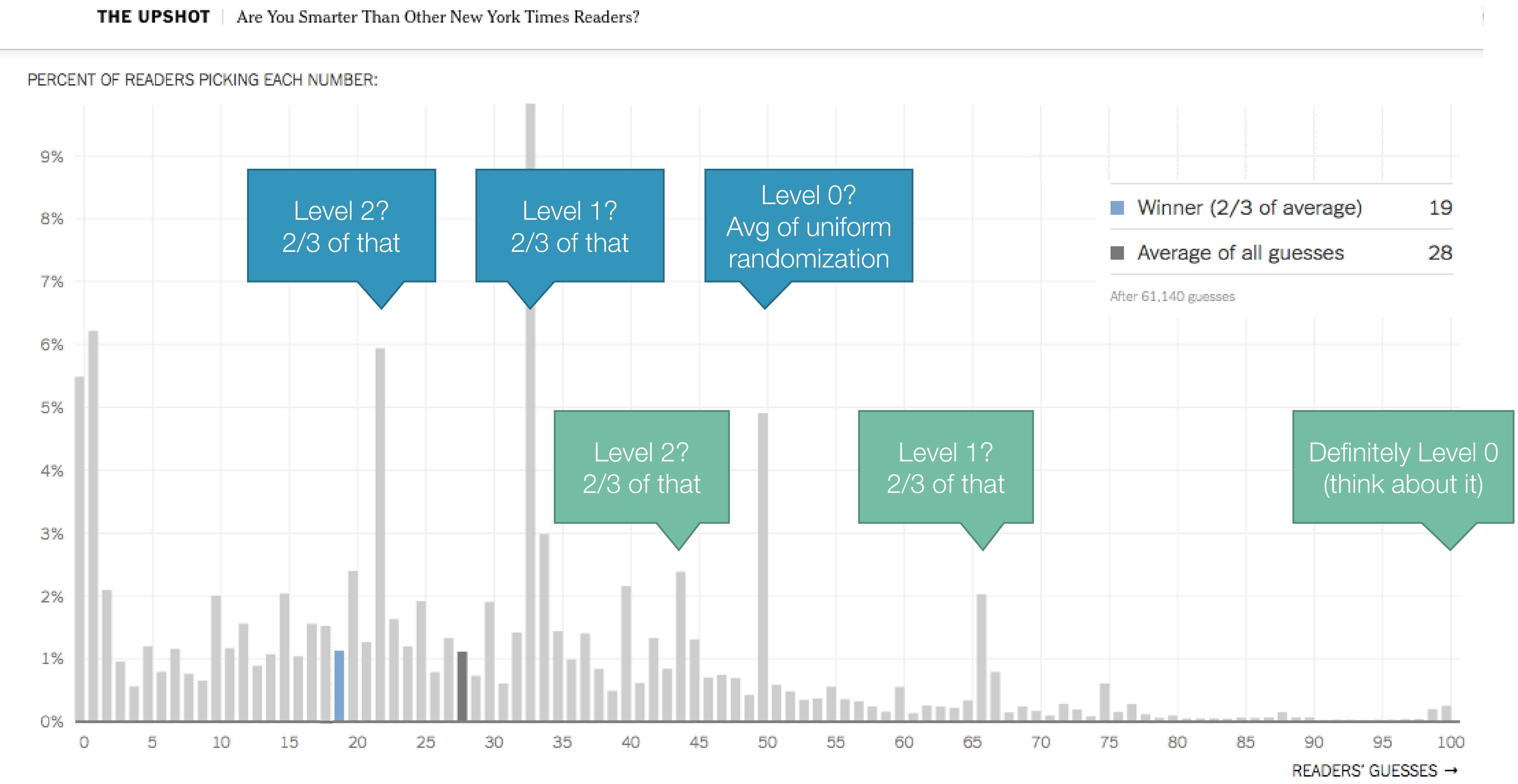
- **Level-0:** Some **nonstrategic** distribution of play (often uniform distribution)
- **Level-1:** Respond to level-0 players
- **Level-2:** Respond to level-1, or levels 0, 1

⋮

- **Level-k:** Respond to level $k-1$, or levels $\{0, \ldots, k-1\}$

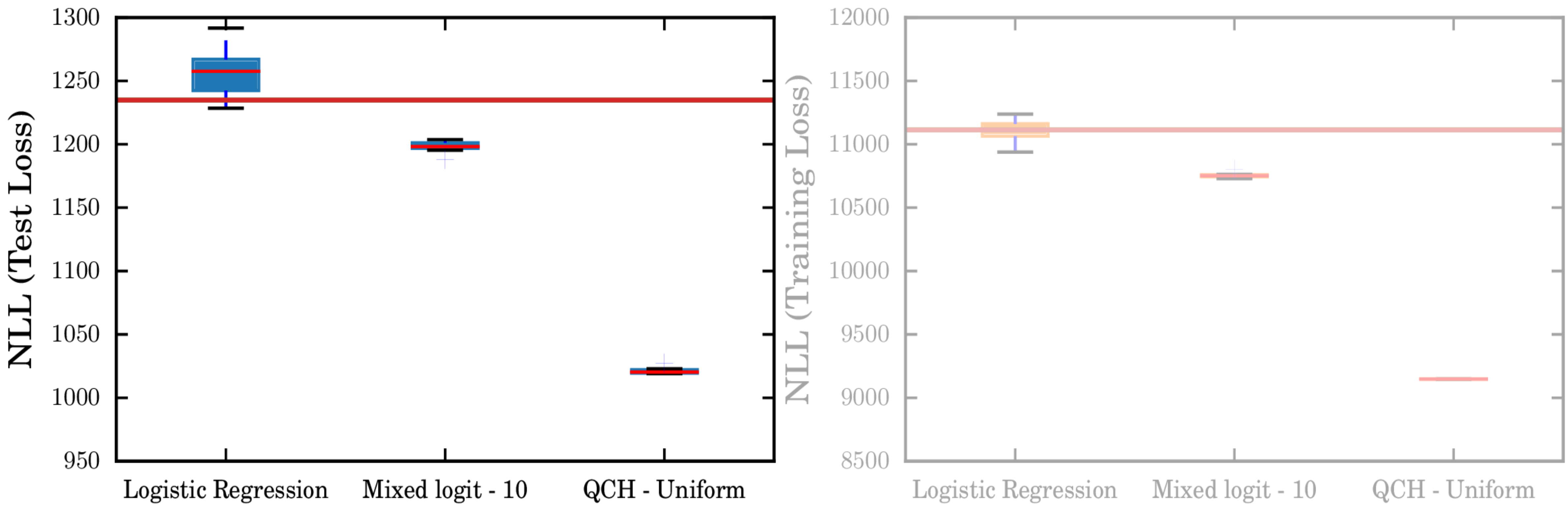


"Are You Smarter Than 61,140 Other New York Times Readers?"



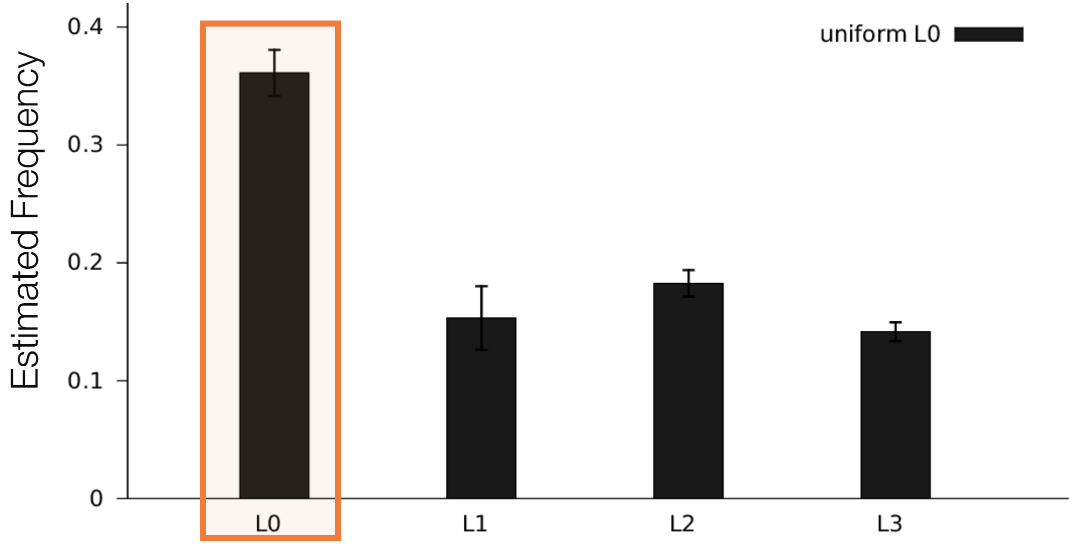
Source: http://www.nytimes.com/interactive/2015/08/13/upshot/are-you-smarter-than-other-new-york-times-readers.html

Behavioral model performance



Level-0 agents

- **Bayesian analysis of parameters** [Wright, Leyton-Brown: AAMAS 2012] shows something strange:



- Best performing models quite certain that many players **randomize uniformly**
  - evidence of a misspecified model?

- **Research Question #3:** Can we fit models in a way that better constrains parameters to their **intended interpretations**?

Let’s model Level-0 behavior explicitly

[Wright, Leyton-Brown: EC 2012; JAIR 2019]

Four binary features:

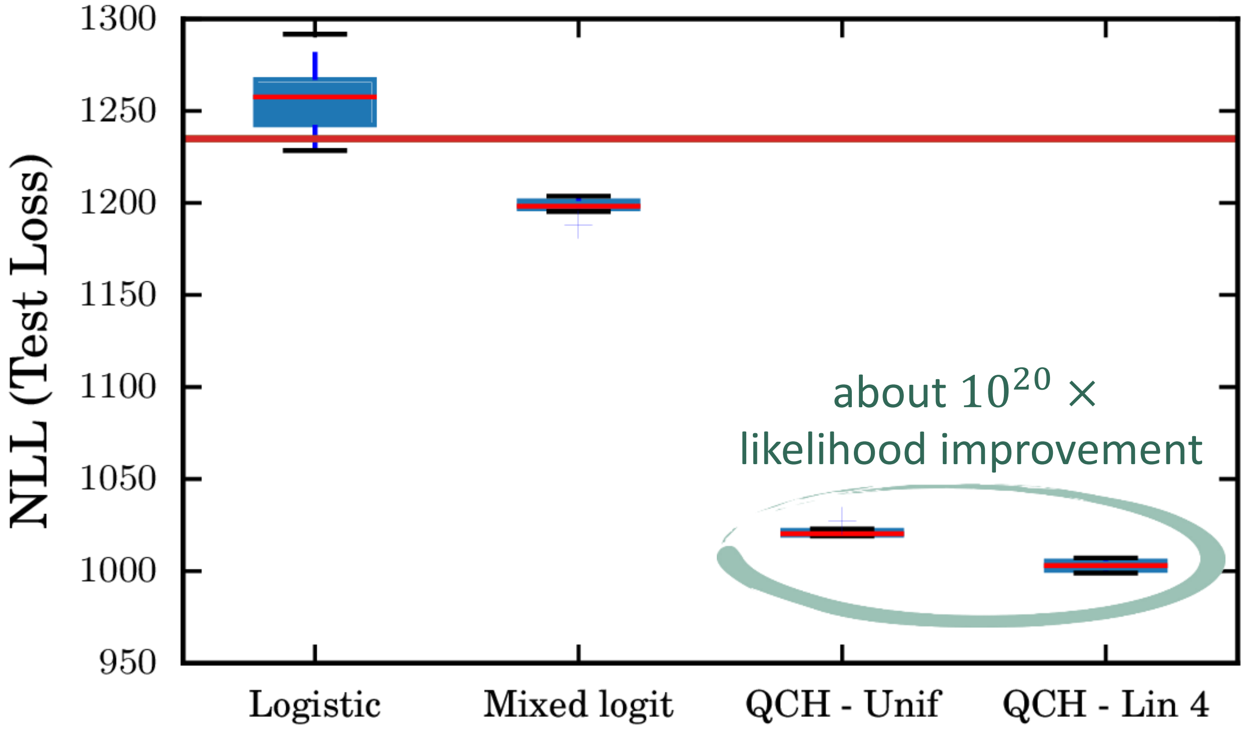
- **Maxmin payoff (“Pessimistic”):** Is this action best in the (deterministic) worst case?
- **Maxmax payoff (“Optimistic”):** Does this action contribute to my own highest-payoff outcome?
- **Fairness:** Does this action contribute to the least unfair outcome?
- **Symmetry:** In symmetric games, would this action be best if my opponents copied my strategy?

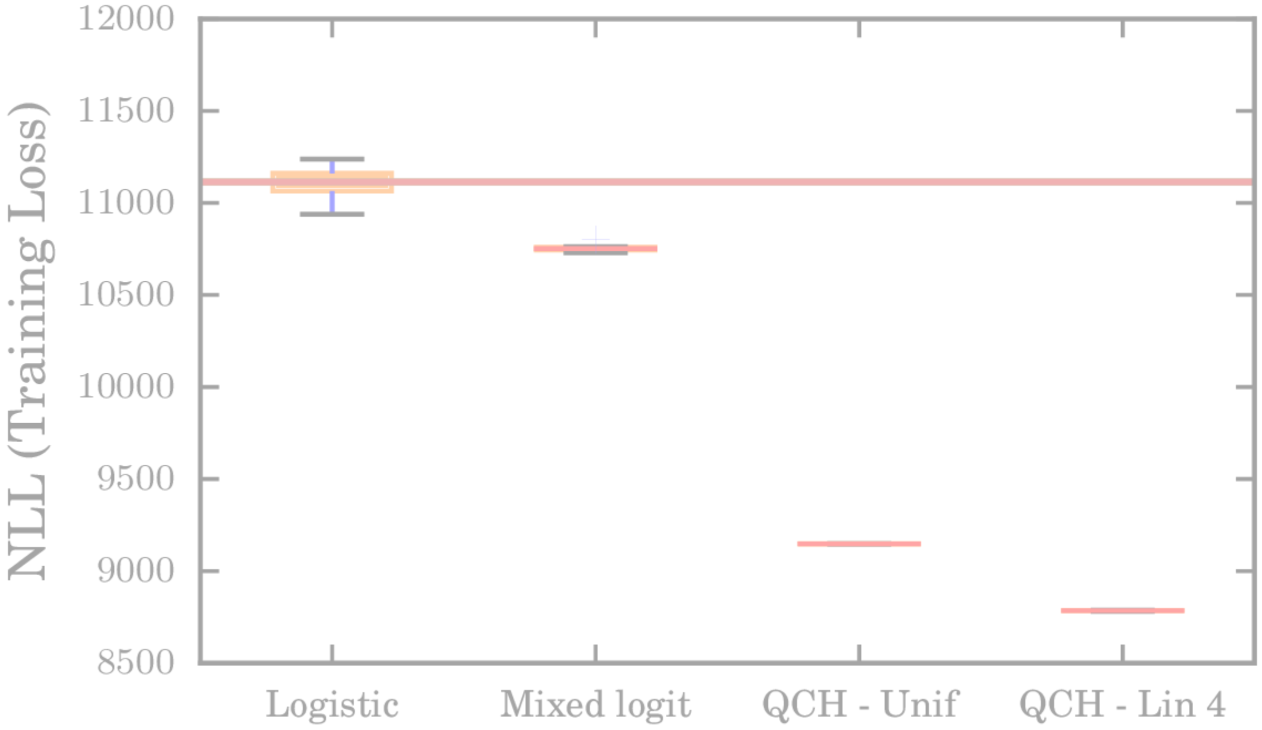
Weighted linear model

- A feature f is **informative for game G** if f can distinguish at least one pair of actions in G
- For each action, compute a **sum of weights for features that are both informative and that "fire"**, plus a noise weight

$$\text{prediction for } a_i \propto w_0 + \sum_{f \in F} \mathbb{I}[f \text{ is informative}] \cdot \mathbb{I}[f(a_i) = 1] \cdot w_f$$

Effect of modeling nonstrategic play





Beyond Feature Engineering

- A better model of **nonstrategic play** made a big difference
- But, it's hard to know if we've got the model right:
  - have we included the right **features**?
  - do our models have the right **functional form**?

Research Question #4: We proposed a pretty arbitrary level-0 model. Is there a **more principled way** to find good level-0 models?

- Deep learning has demonstrated the possibility of stunning predictive performance via **learning features**
- Could we **automatically search** for behavioral models?

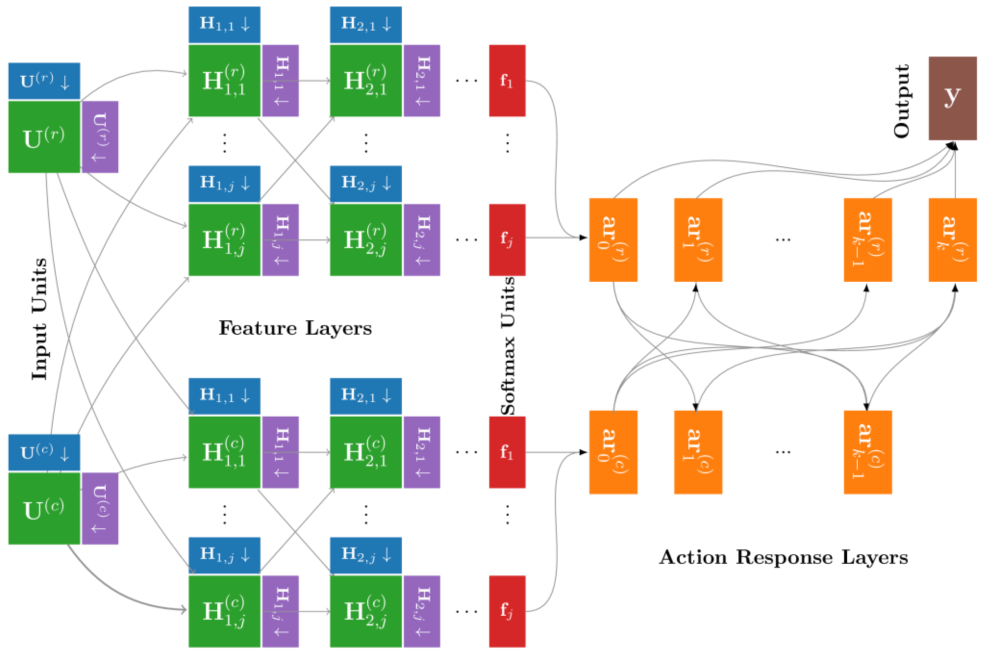
Game-Theoretic Wish List

1. Invariance to game **permutations**
2. **Variable-size output**: a probability distribution over player 1's action space
3. Rich enough to model **iterative strategic reasoning**

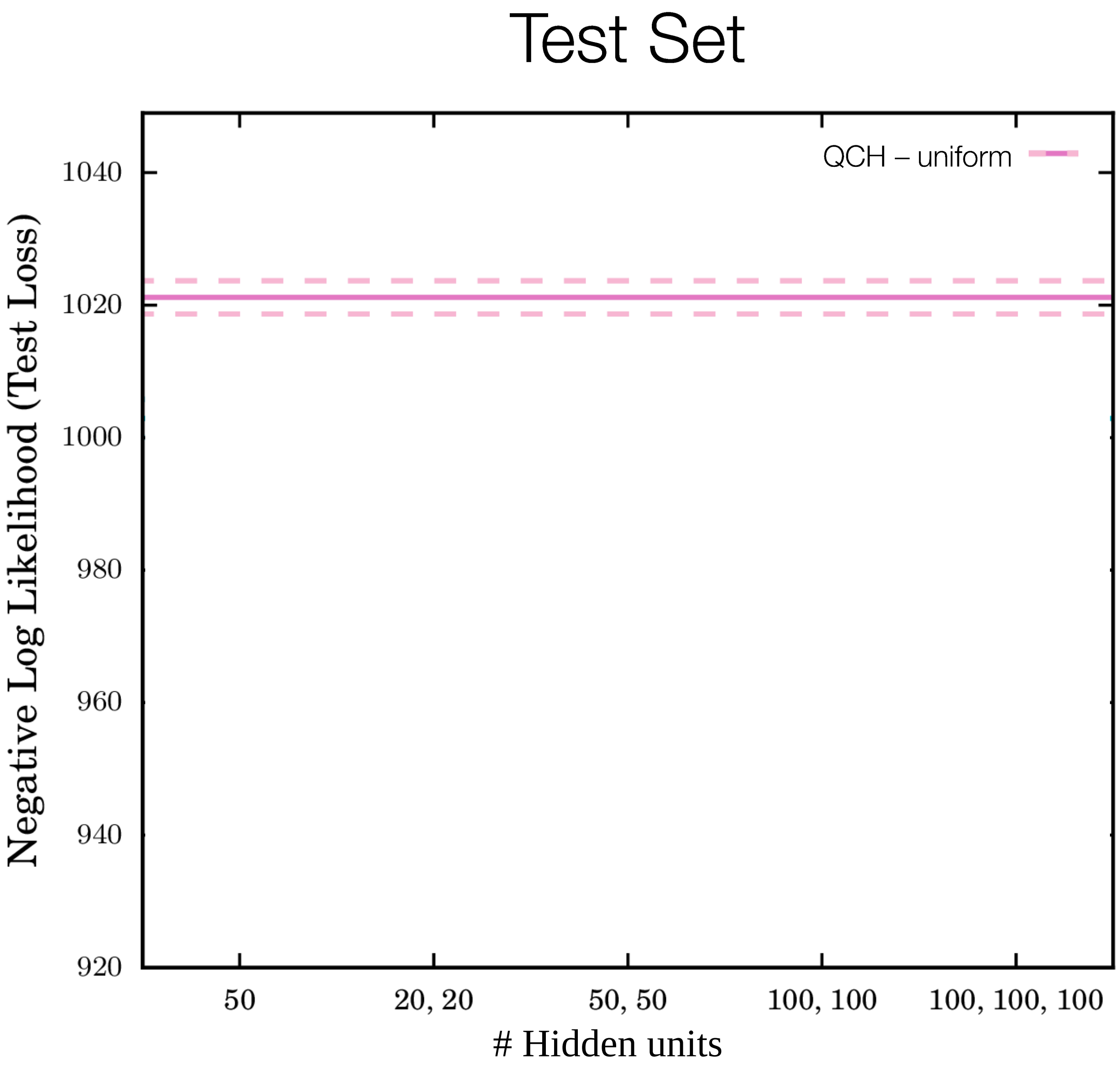
Deep Learning for BGT

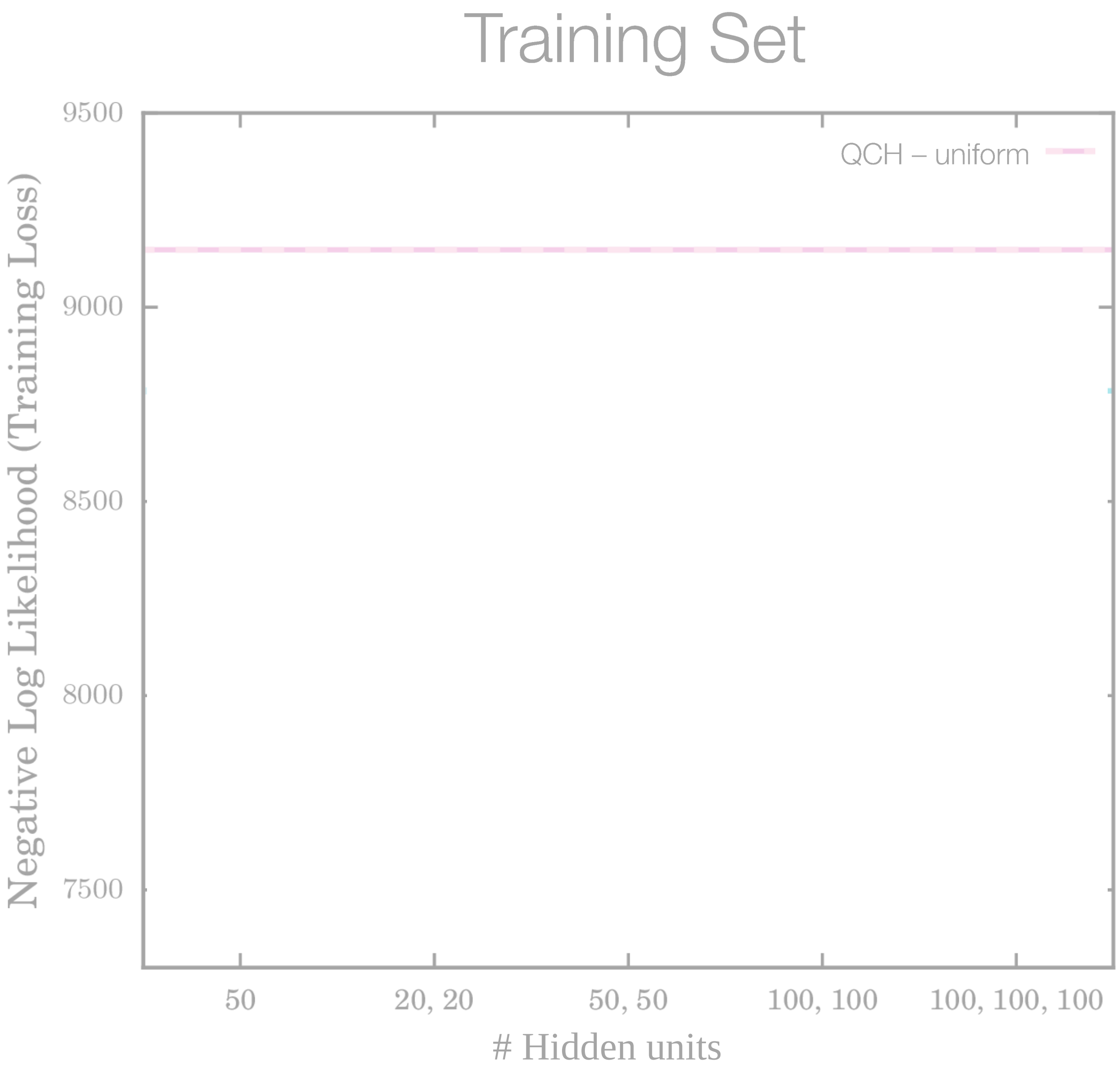
[Hartford, Wright, Leyton-Brown: NeurIPS 2016]

- Our solution: a novel neural network architecture
  - nodes compute relu of element-wise weighted sum of input matrices, output new matrices
  - interaction across elements via "max pooling" across rows and columns
  - permutation-equivariant
  - today, the same ideas could be implemented off-the-shelf using graph convolution
- explicit action-response layers to capture QCH



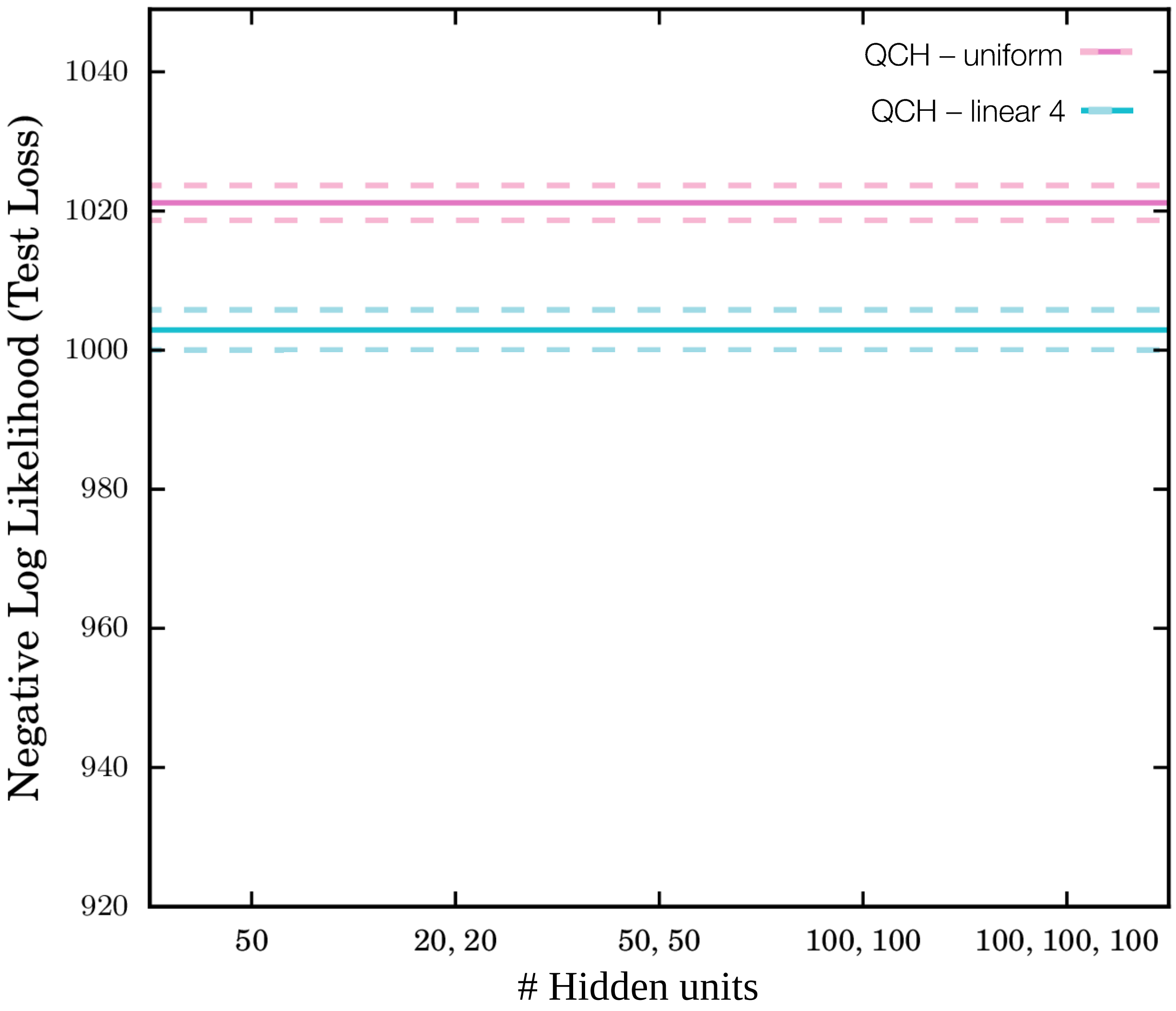
Performance



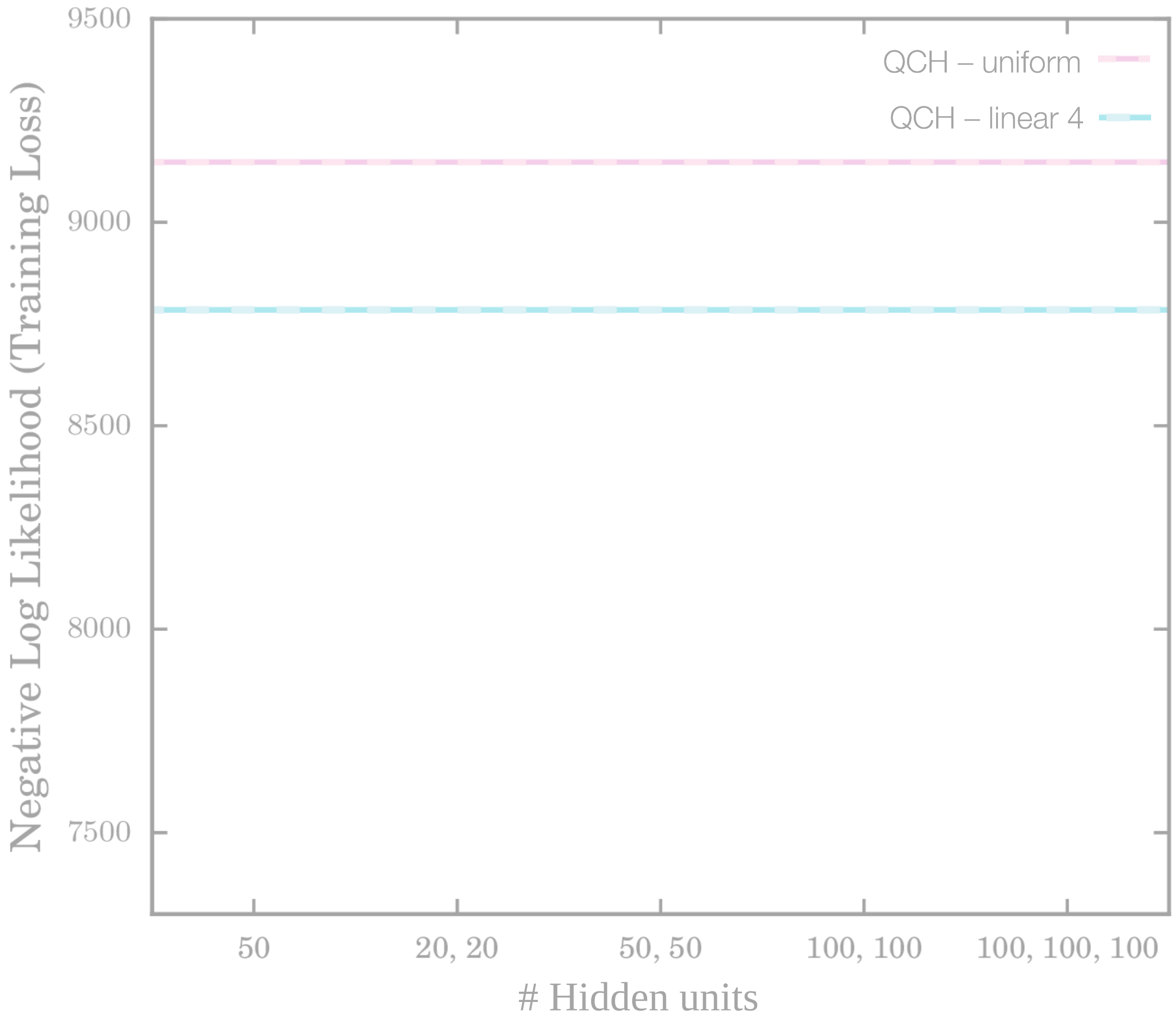


Performance

Test Set

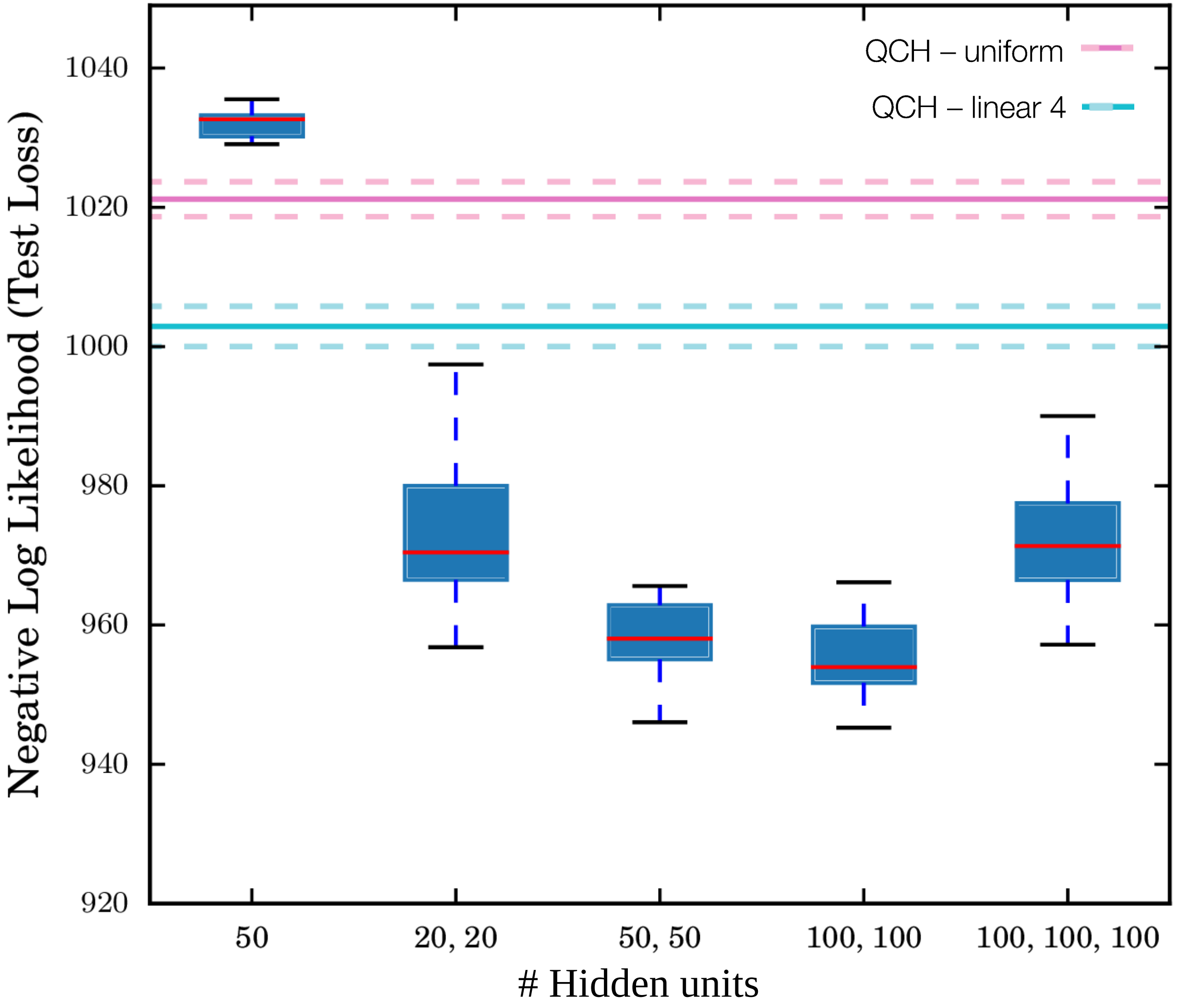


Training Set

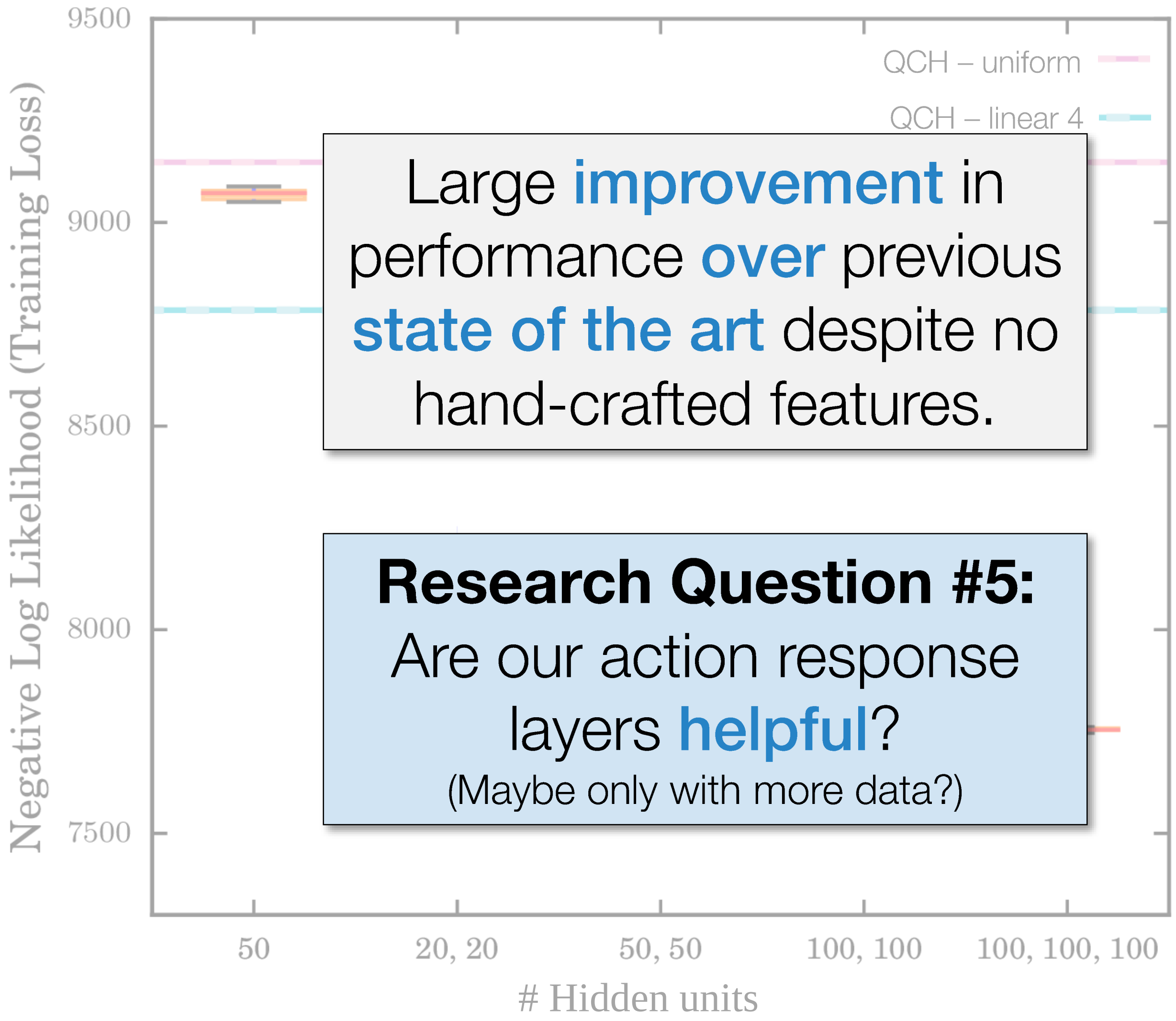


Overall performance

Test Set



Training Set



Large **improvement** in performance **over** previous **state of the art** despite no hand-crafted features.

Research Question #5:
Are our action response layers **helpful**?
(Maybe only with more data?)

Limits of Nonstrategic Behavior

- **Research Question #6:** At what point does L0 behavior get so complex that it ought to be **considered strategic**?
  - Typical answer: if the behavior involves **modeling other agents**
  - But, hard to know if apparently nonstrategic behavior can be **rephrased in strategic terms**
    - weighted linear combinations of our four hand-crafted L0 features?
    - the L0 model learned by a deep network?
- A new, formal characterization of nonstrategic behavior [Wright, Leyton-Brown: EC 2020; R&R@JET] that satisfies two properties:
  1. general enough to capture all **existing "nonstrategic" decision rules**
  2. behavior we characterize is **distinct from strategic** in a precise sense
- Permits e.g. optimizing over the space of nonstrategic behaviors

Elementary Behavioral Models

- How an elementary model works:
  - Given a game $G = (N, A, u)$, for each action profile $a \in A$, **apply the same "no-smuggling" function φ** to the $|N|$-tuple of real values $\langle u_1(a), \ldots, u_{|N|}(a) \rangle$, producing a real-valued "potential matrix" Φ
  - Apply **any arbitrary h** to Φ, producing a probability distribution over A_i
- We prove that
  - no existing **strategic decision rule** (Nash, QRE, QCH, etc) is elementary
  - **no elementary model is strategic** (i.e., both "other responsive" and "dominance responsive")
  - neither is any function of the predictions of **finitely many elementary models**
    - Linear4 is nonstrategic
  - GameNet is *not* nonstrategic (perhaps why action response layers didn't help us)

- **Research Question #7:** identify **nonstrategic deep learning architectures**

Conclusions and Future Directions

- **Behavioral game theory** does a much better job than traditional game theory for modeling human behavior
- The best models (e.g., quantal cognitive hierarchy) depend on a specification of **nonstrategic “level-0” behavior**
  - performance can be improved by modeling this richly
  - and can be even further improved with fancy deep learning
- Directions for further research (in many cases, with preliminary answers) :
  1. What difference does it make to use a **better motivated loss function** than log likelihood?
  2. Should we explicitly model **further behavioral phenomena**?
  3. Can we fit models to better conform to their **intended interpretations**?
  4. Is there a more principled way to model **level-0 behavior**?
  5. Does it help to combine **deep learning with cognitive hierarchy**?
  6. How should we define **(non)strategic behavior**?
  7. Can we find **nonstrategic deep learning architectures**?