Machine learning tools for model fitting, comparison, and discovery

Peter Kvam

The Ohio State University

UIUC Quantitative Psychology

October 16, 2024

Outline

- Machine learning tools for **fitting and comparing models**
  - Issues with current approaches
  - Comparison with hierarchical Bayes
  - Illustration in intertemporal choice

- Machine learning tools for **discovering models**
  - Autoencoders, clustering
  - Reinforcement learning, explicit learning, and computational evolution
  - Application to procrastination behavior

- Future directions

Outline

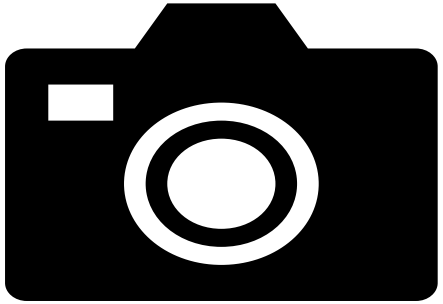
- Machine learning tools for **fitting and comparing models**
  - Issues with current approaches
  - Comparison with hierarchical Bayes
  - Illustration in intertemporal choice

- Machine learning tools for **discovering models**
  - Autoencoders, clustering
  - Reinforcement learning, explicit learning, and computational evolution
  - Application to procrastination behavior

- Future directions

Generative modeling

- Best way to understand something and predict what it does is to know how it works (**generative / process model**)
  - Goes a bit deeper than descriptive statistical models (regression, ANOVA, etc.) that summarize the patterns of observations



Description

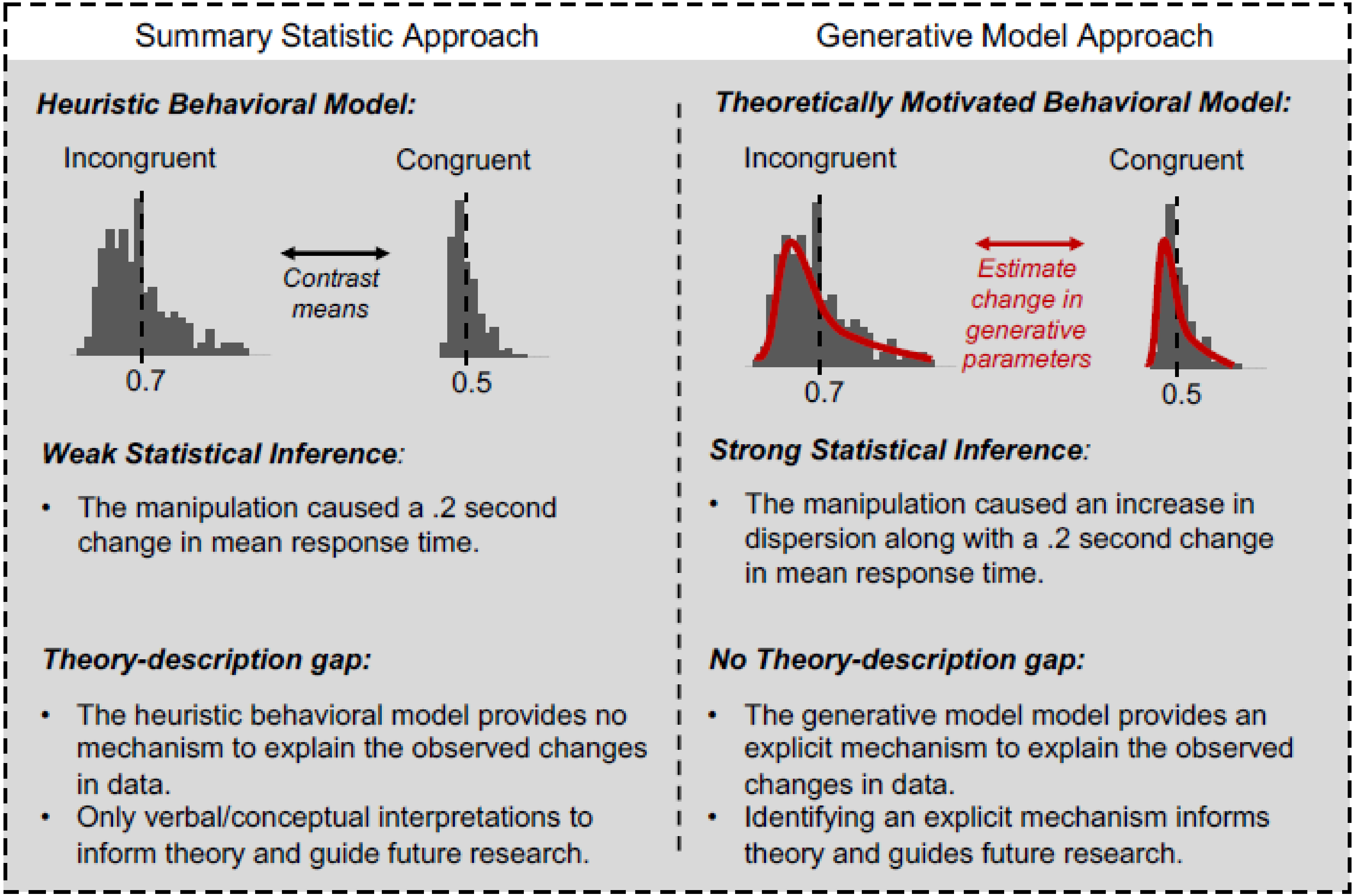
- Takes pictures
- Transmits pictures to computer
- Lightens photographs that were taken in low light conditions

Mechanism

- Focuses light onto internal mirrors
- Codes light & sounds as data
- Sensor evaluates lighting to adjust aperture, ISO, shutter speed

Better reliability and validity:

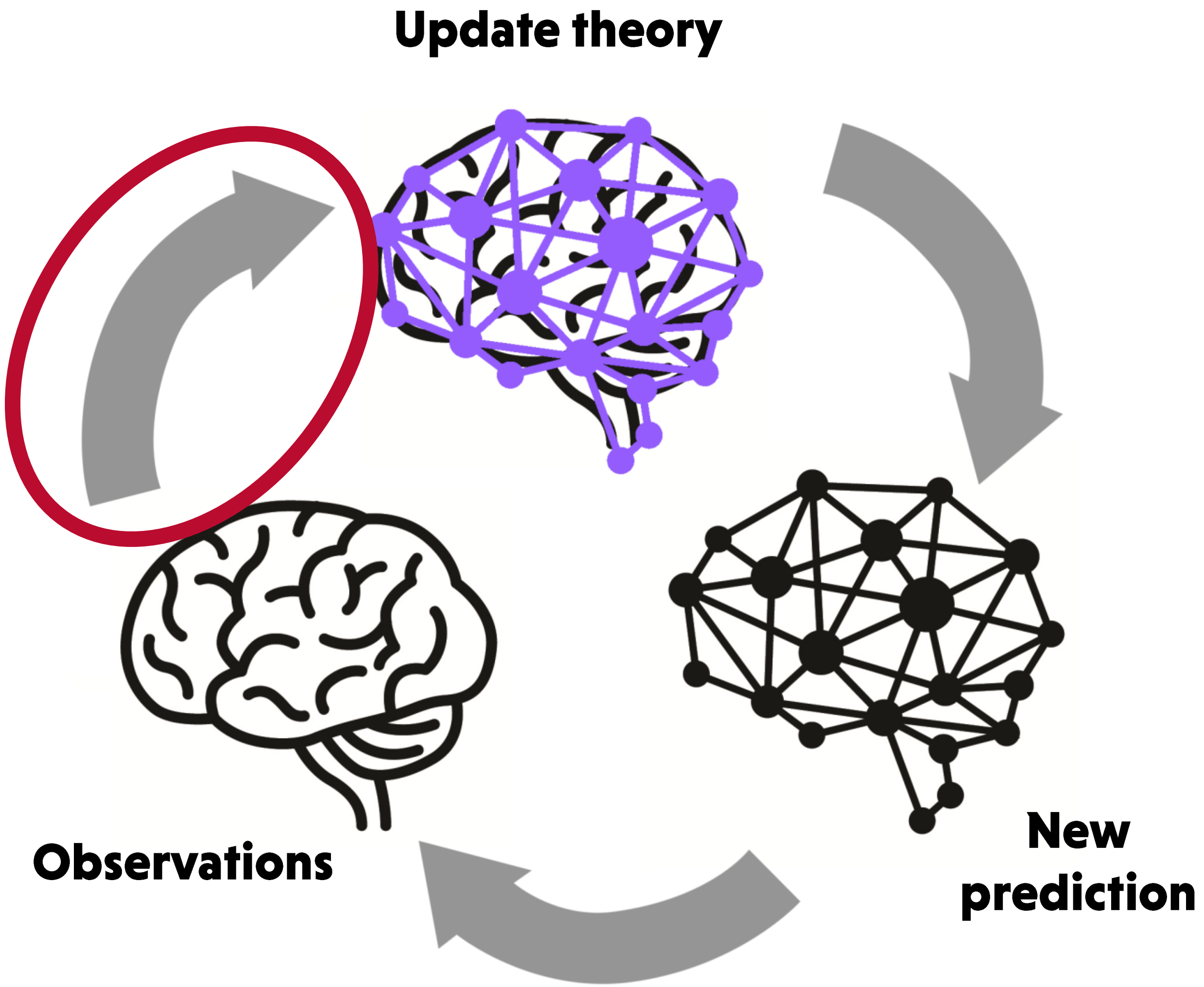
Haines, N., Kvam, P. D., Irving, L. H., Smith, C., Beauchaine, T. P., Pitt, M. A., Ahn, W.-Y., & Turner, B. M. (in press). A tutorial on using generative models to advance psychological science: Lessons from the reliability paradox. ***Psychological Methods.***


Summary Statistic Approach
Generative Model Approach
Heuristic Behavioral Model:
Incongruent
Congruent
0.7
0.5
Contrast means
Theoretically Motivated Behavioral Model:
Incongruent
Congruent
0.7
0.5
Estimate change in generative parameters
Weak Statistical Inference:
• The manipulation caused a .2 second change in mean response time.
Strong Statistical Inference:
• The manipulation caused an increase in dispersion along with a .2 second change in mean response time.
Theory-description gap:
• The heuristic behavioral model provides no mechanism to explain the observed changes in data.
• Only verbal/conceptual interpretations to inform theory and guide future research.
No Theory-description gap:
• The generative model model provides an explicit mechanism to explain the observed changes in data.
• Identifying an explicit mechanism informs theory and guides future research.

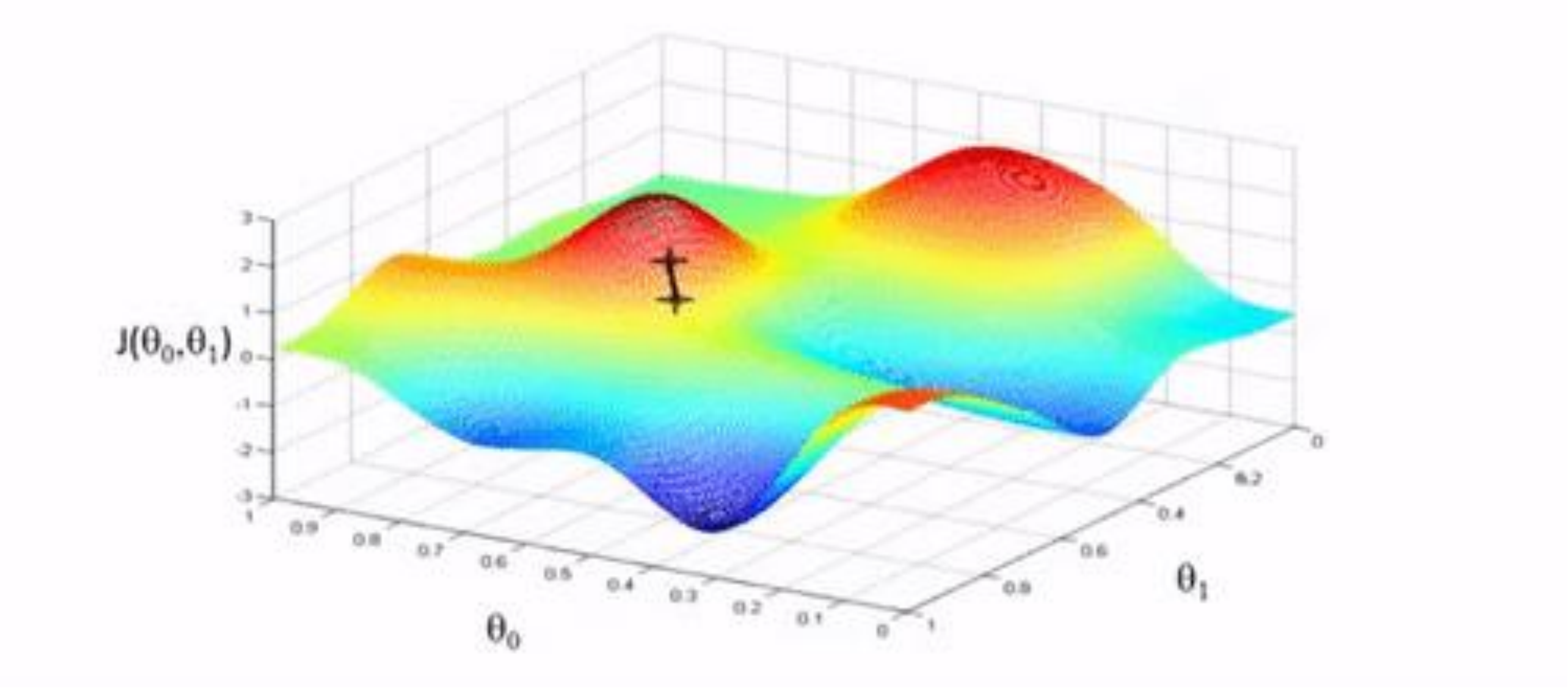
Modeling process

- Can be hypothesis-driven, exploratory, or abductive
- All require us to update our understanding of latent processes based on data
  - Model fitting
  - Model comparison
- Based on **likelihoods:**

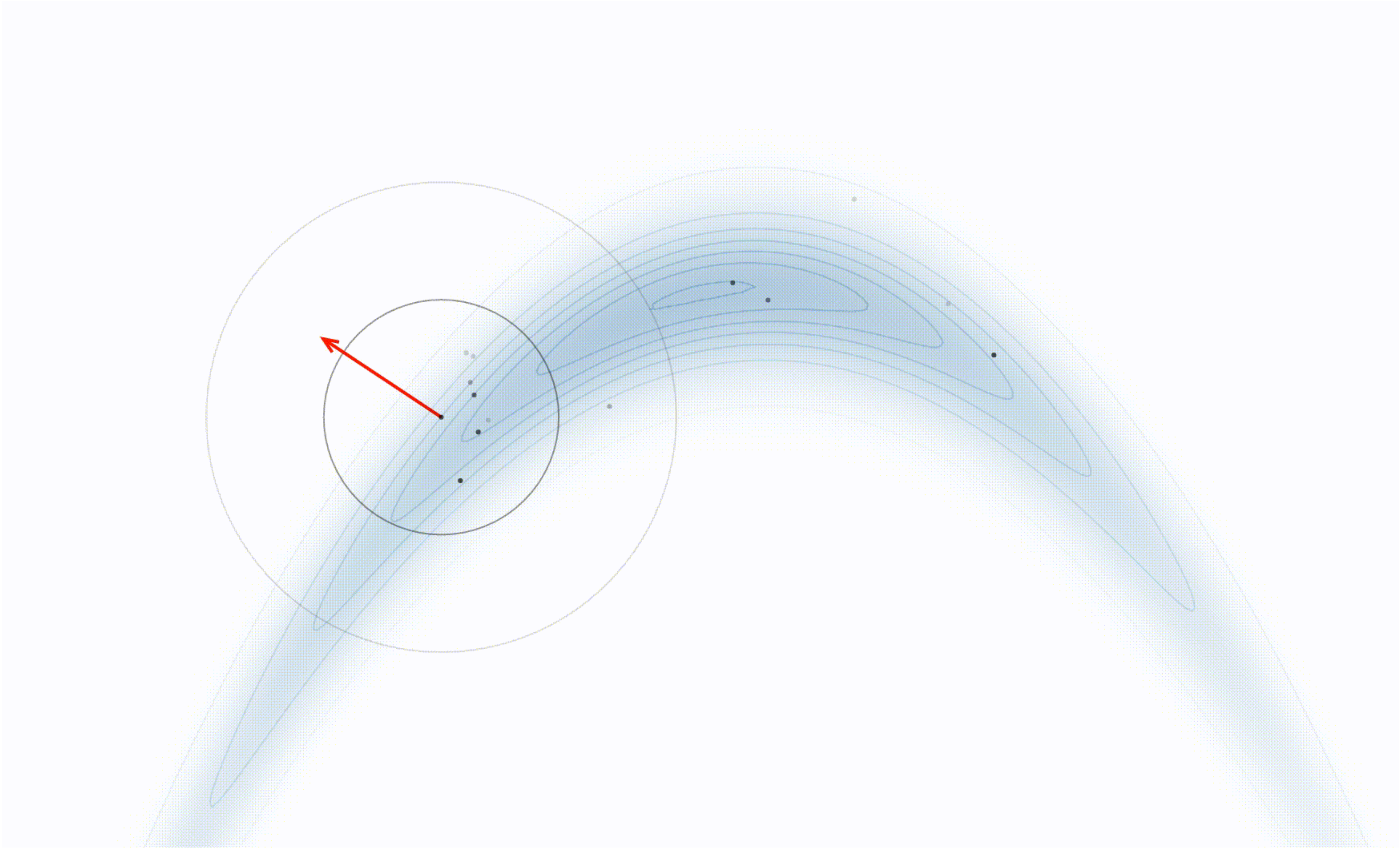
  Pr(Data | Model)



Maximum likelihood / gradient descent



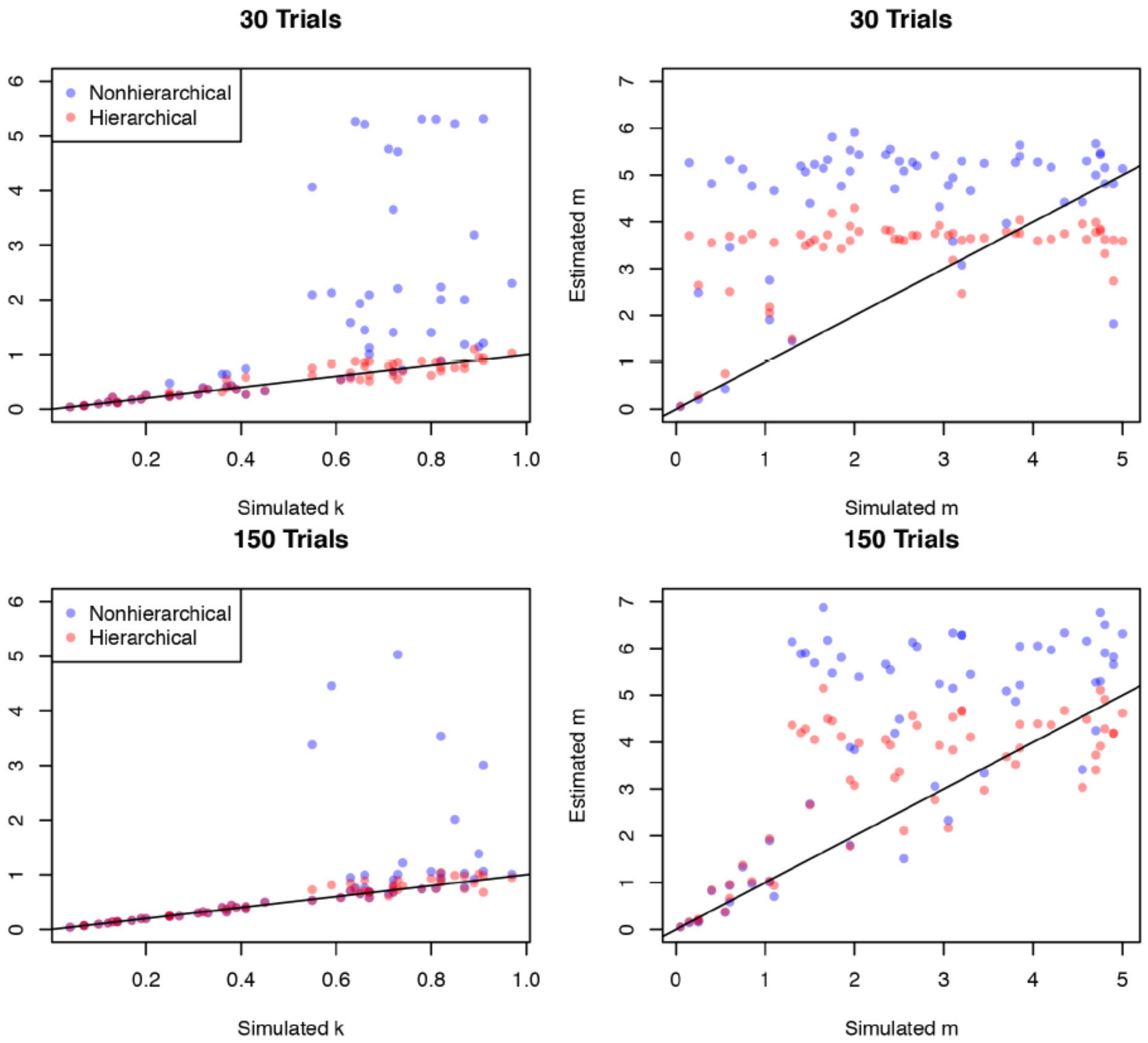
Bayesian / MCMC sampling



Hierarchical Bayes

- Better **estimation**

- As compared to maximum likelihood estimation

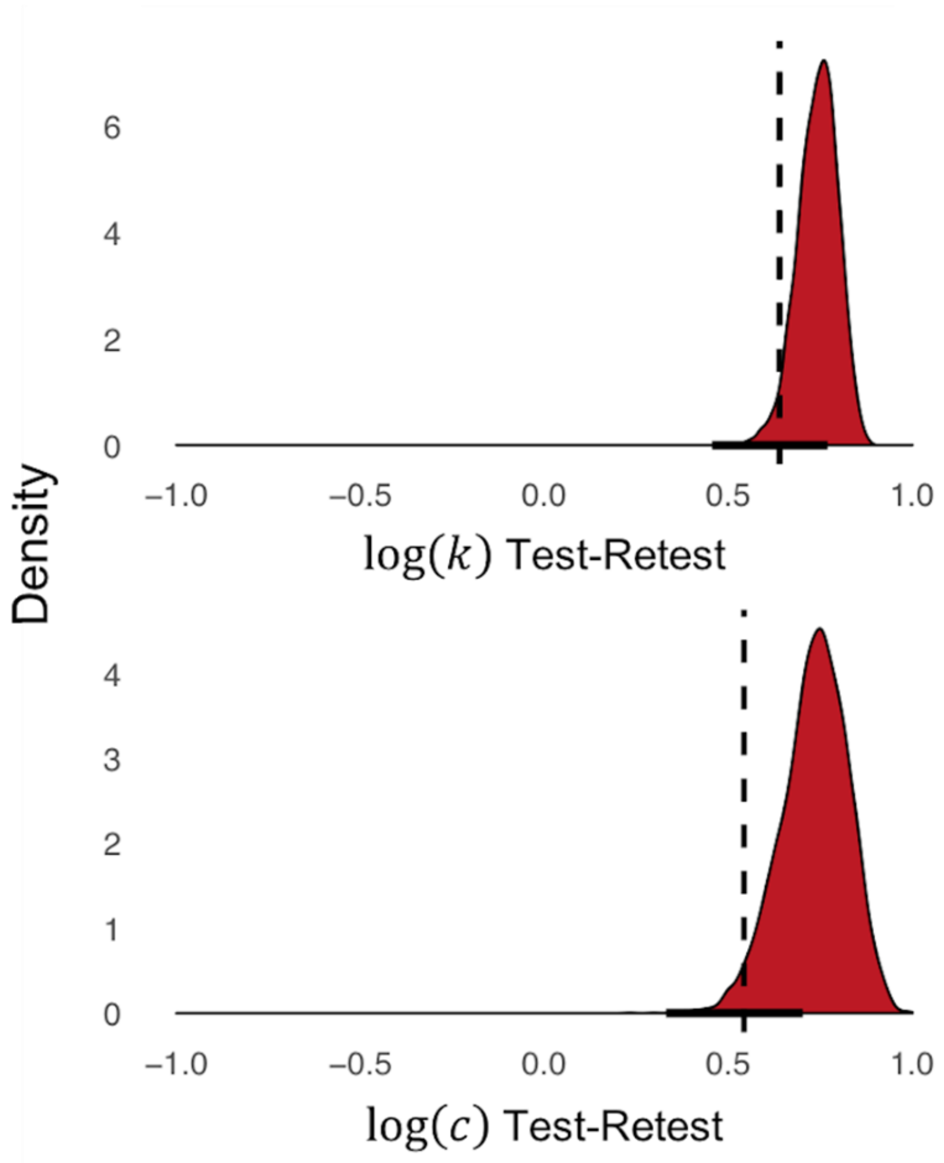


Molloy, M. F., Romeu, R. J., Kvam, P. D., Finn, P. R., Busemeyer, J., & Turner, B. M. (2020). Hierarchies improve individual assessment of temporal discounting behavior. *Decision*, *7*(3), 212-224.

Hierarchical Bayes

- Better **estimation**
- Better **reliability**

- As compared to maximum likelihood estimation

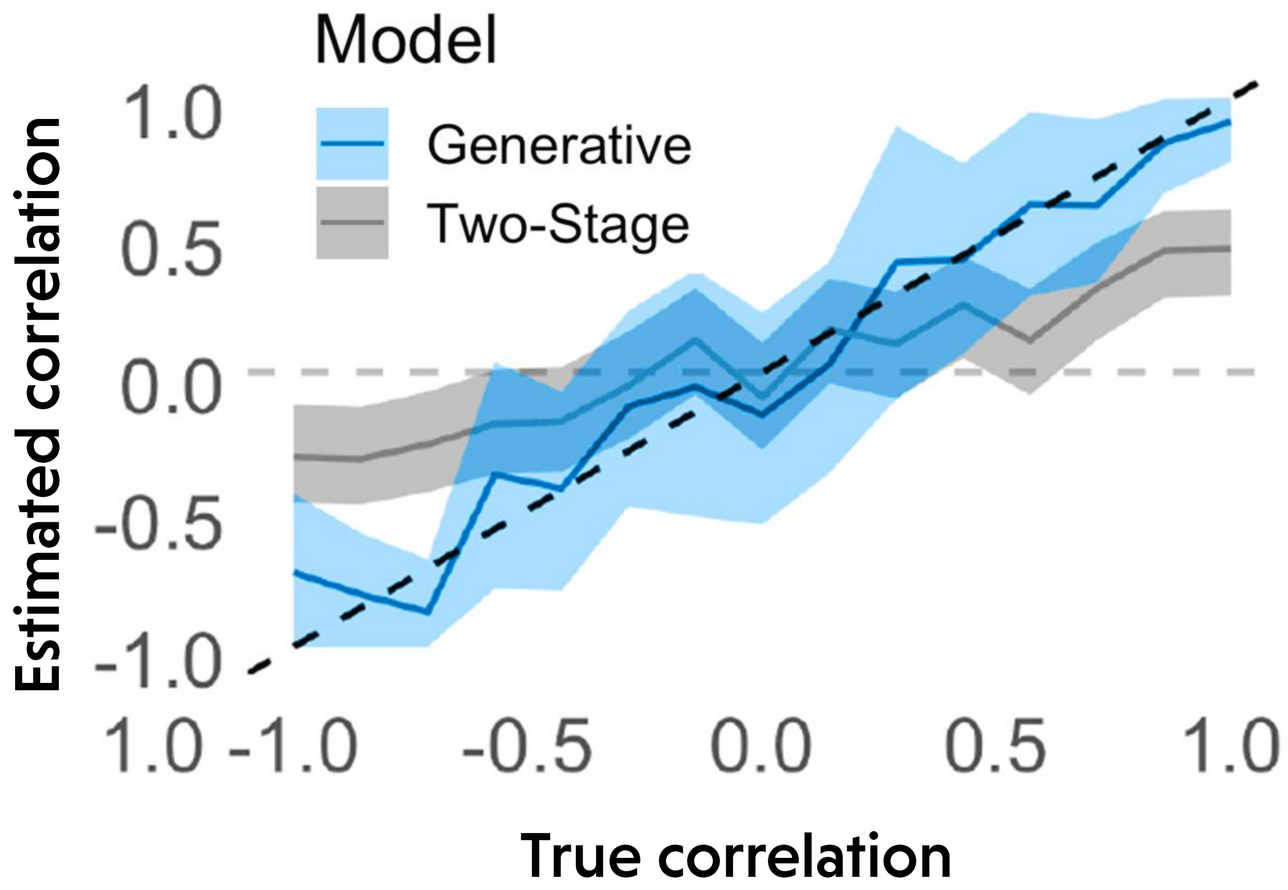


Molloy, M. F., Romeu, R. J., Kvam, P. D., Finn, P. R., Busemeyer, J., & Turner, B. M. (2020). Hierarchies improve individual assessment of temporal discounting behavior. *Decision*, *7*(3), 212-224.

Hierarchical Bayes

- Better estimation
- Better reliability
- Better prediction

- As compared to maximum likelihood estimation



Molloy, M. F., Romeu, R. J., Kvam, P. D., Finn, P. R., Busemeyer, J., & Turner, B. M. (2020). Hierarchies improve individual assessment of temporal discounting behavior. *Decision*, *7*(3), 212-224.

Drawbacks & Limitations

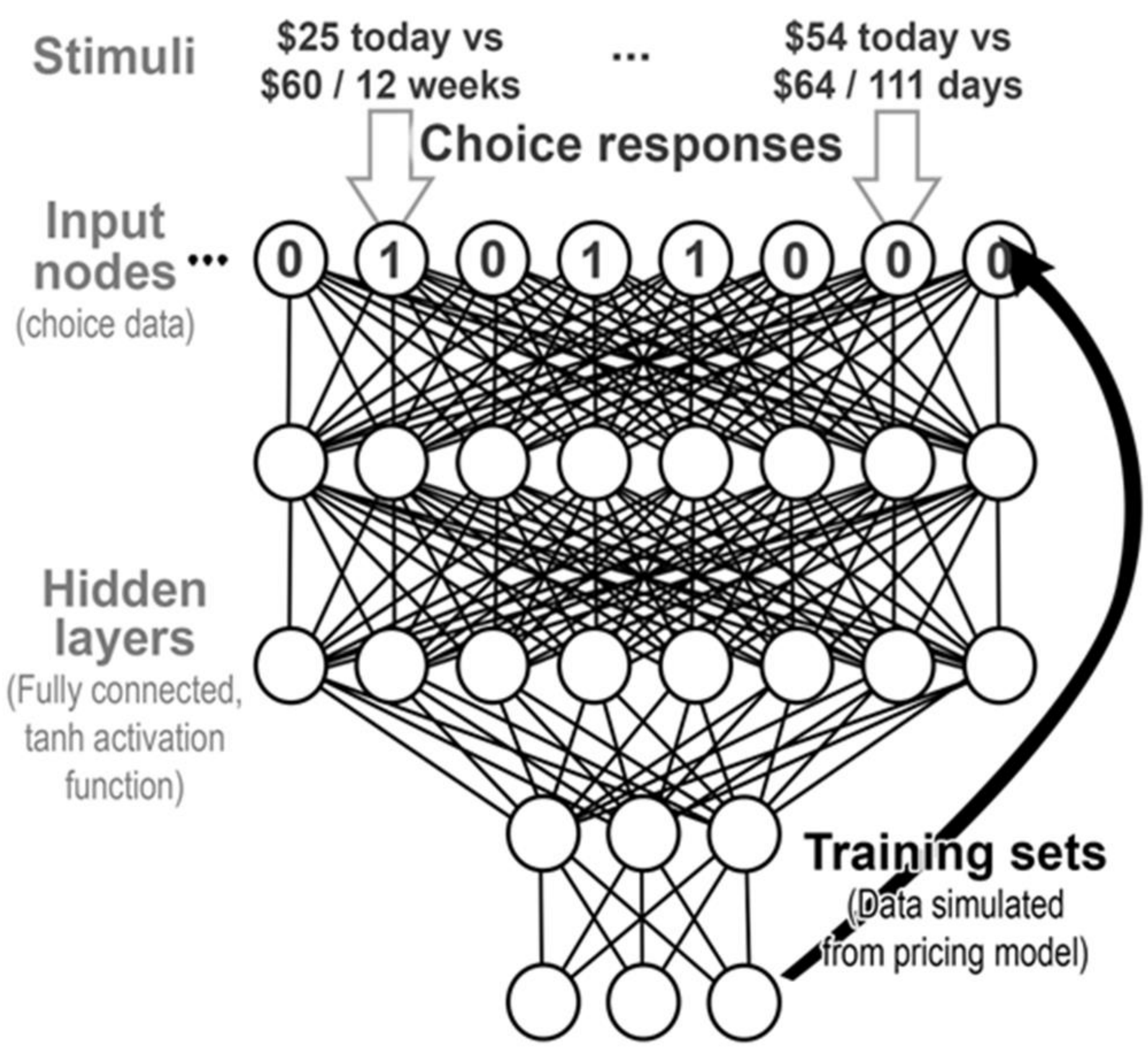
- Many models **don't have likelihoods** that make them easy to fit
  - Use PDA or some other approximation of the likelihood
  - Others have multimodal likelihoods that are difficult to explore
- **Slow:** many types of models can take several hours or even days / weeks to estimate (esp. if done hierarchically)
- **Computationally demanding:** need a good computer or even a supercomputer to fit
- **Inaccessible:** implementing & using models requires a high level of coding knowledge and quantitative training

AI for model fitting

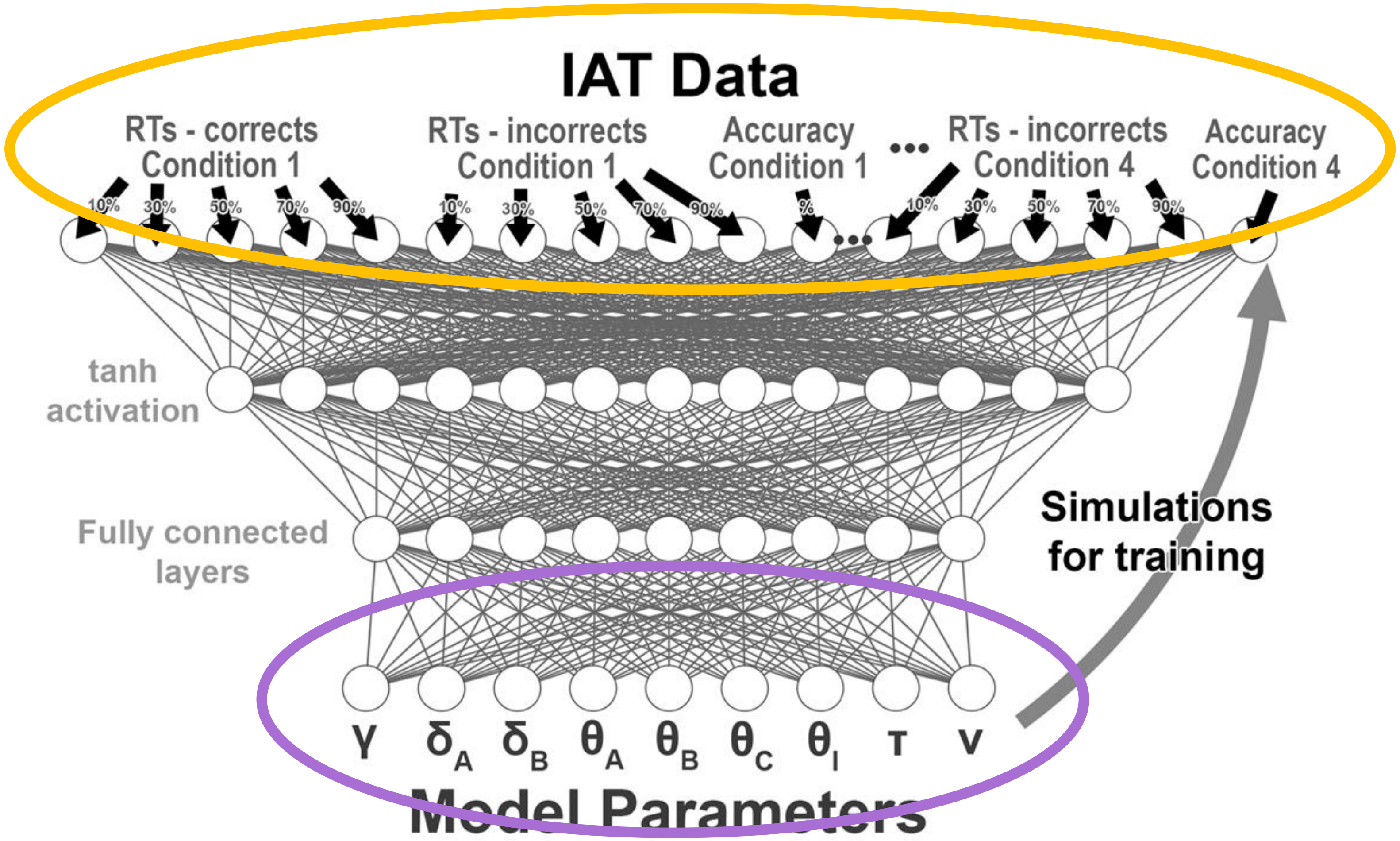
- Teach a machine to do the model fitting for us!
  - **Deep learning** for parameter estimation and model comparison

Simulation
Parameters → Data

Invert it!
Data → Parameters



Neural networks for model fitting



- **Inputs:** complete or simple statistical summary of the behavioral data
- **Outputs:** best-fit model parameters (what generated the data?)
  - Can be supplemented with additional networks that estimate error from input data, sample size, best-fit estimates: posterior variance
- Typically found that 3+ hidden layers, decreasing in size, with tanh activation works best

Example application: intertemporal choice

- Choices between a larger, later and a smaller, sooner option

A	B
$40	$55
today	30 weeks

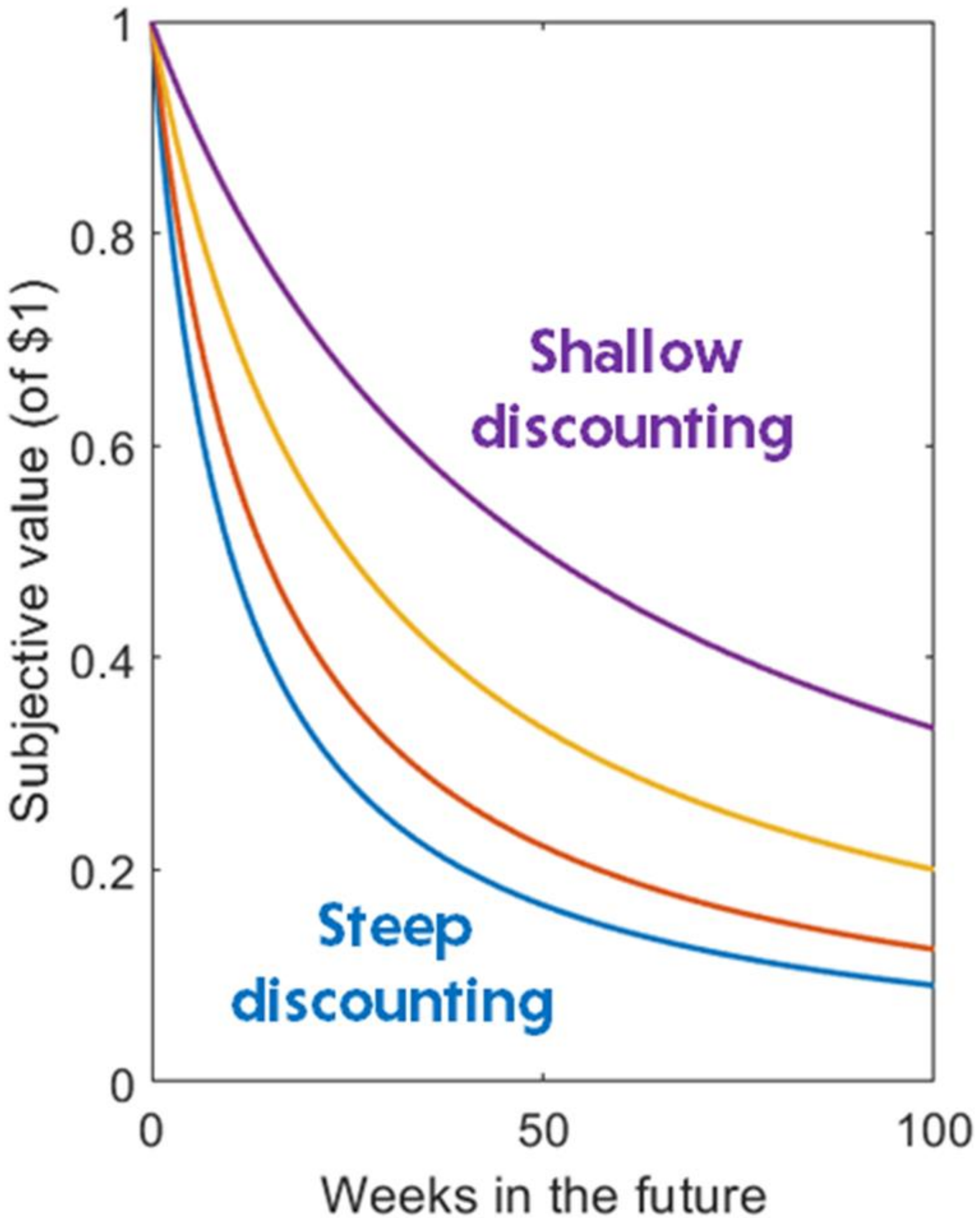
- Thought to be a **transdiagnostic process** in psychiatric disorders (Amlung et al, 2019)

Simple models

- Delaying an item in time (moving further away) decreases its subjective value
- Common model: **hyperbolic discounting**

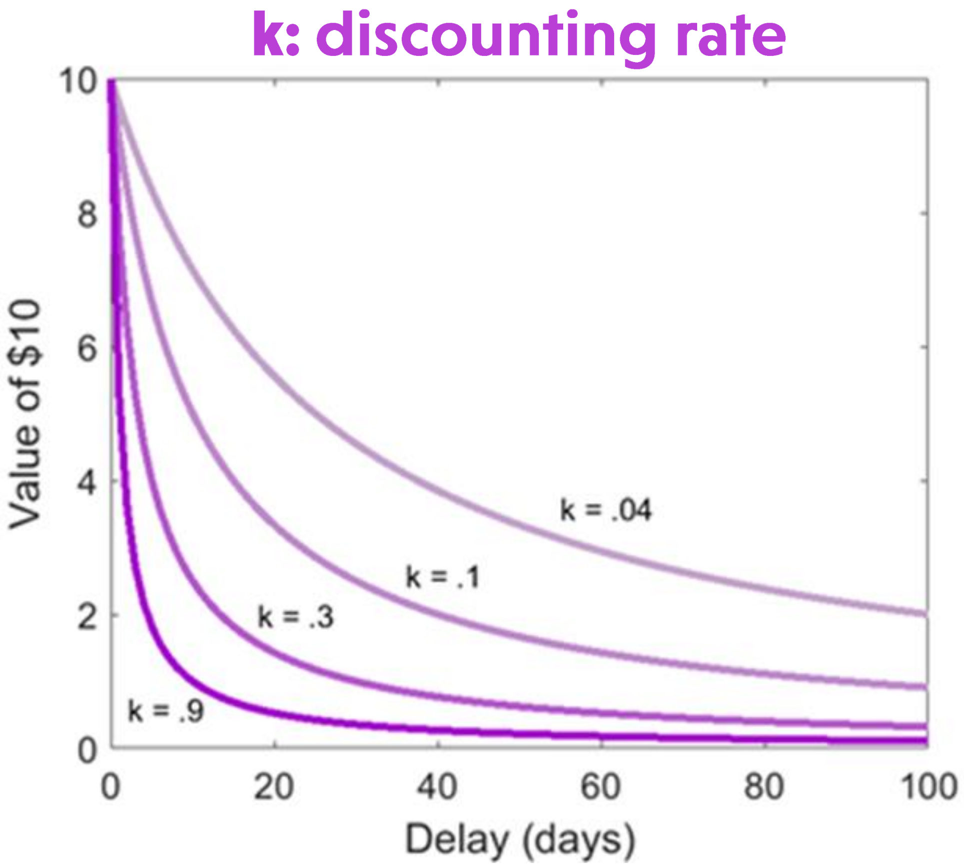
  V(t) = 1 / (1 + k*t)

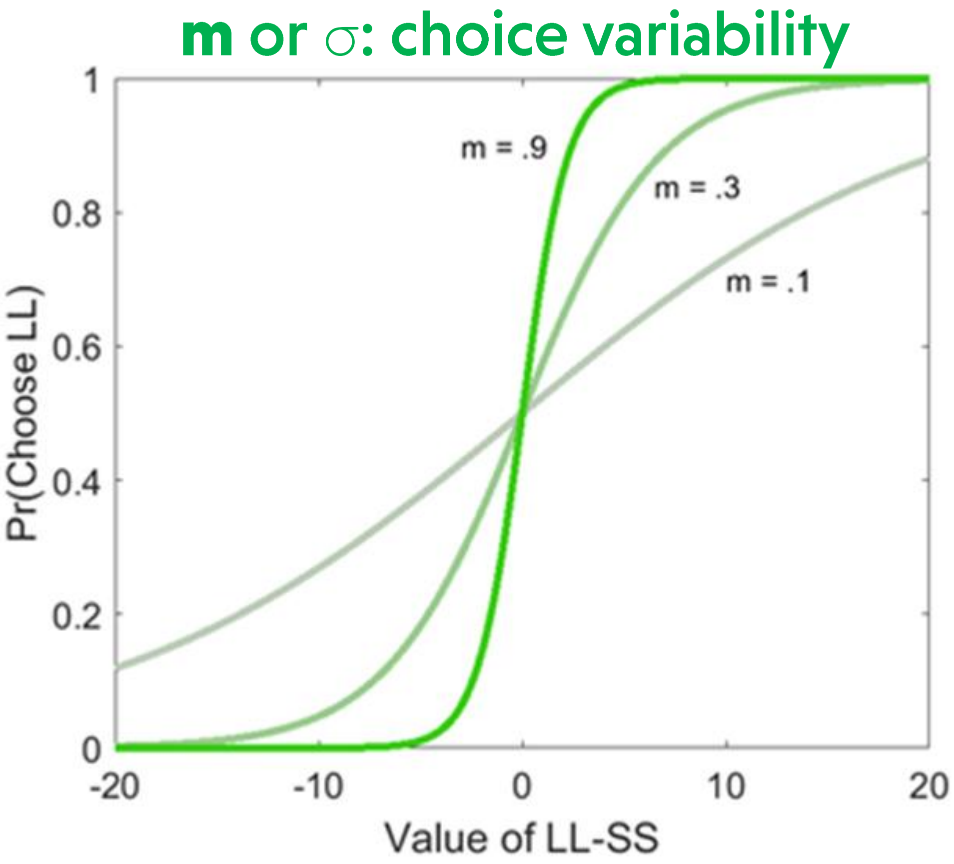
- Connected to choice proportions using logistic choice rule

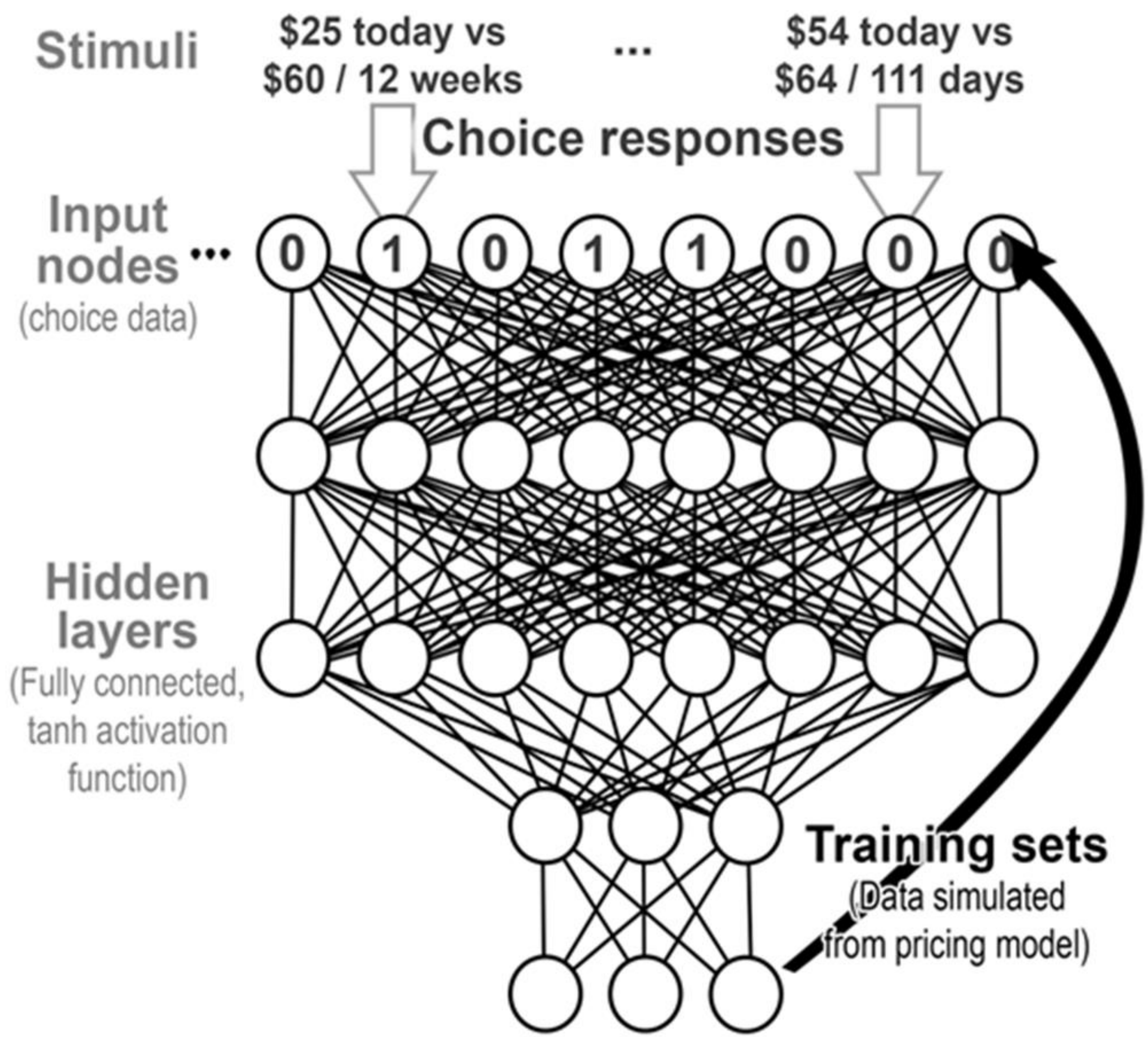


Delay discounting model

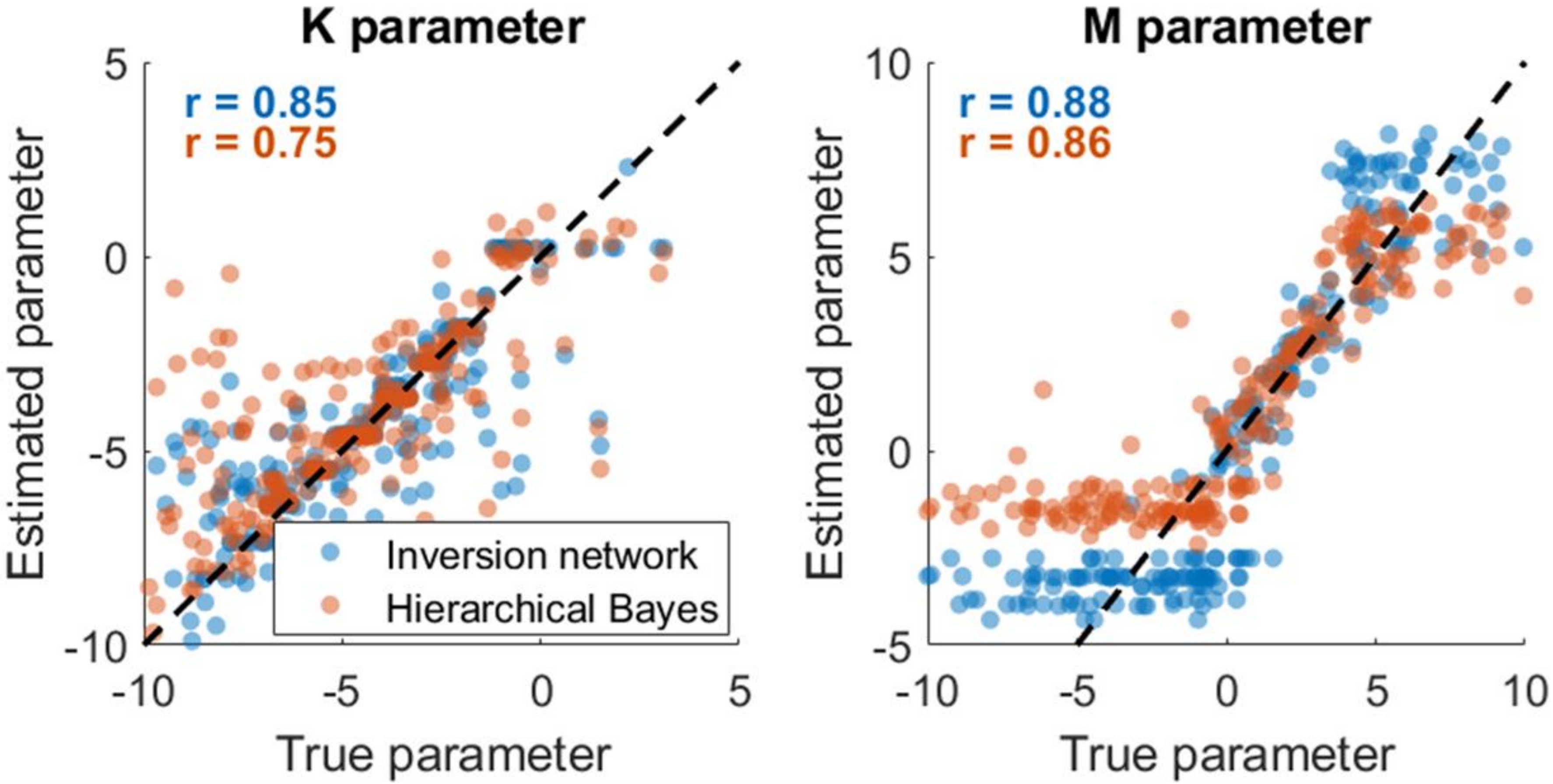
- **Hyperbolic discounting:**






Stimuli
$25 today vs
$60 / 12 weeks
...
$54 today vs
$64 / 111 days
Choice responses
Input nodes
(choice data)
...
0 1 0 1 1 0 0 0
Hidden layers
(Fully connected, tanh activation function)
Training sets
(Data simulated from pricing model)

How does this method compare?



Example: Fitting a hyperbolic discounting model to the data from the monetary choice questionnaire (MCQ, 27 intertemporal choice questions) using hierarchical Bayesian vs deep learning approaches

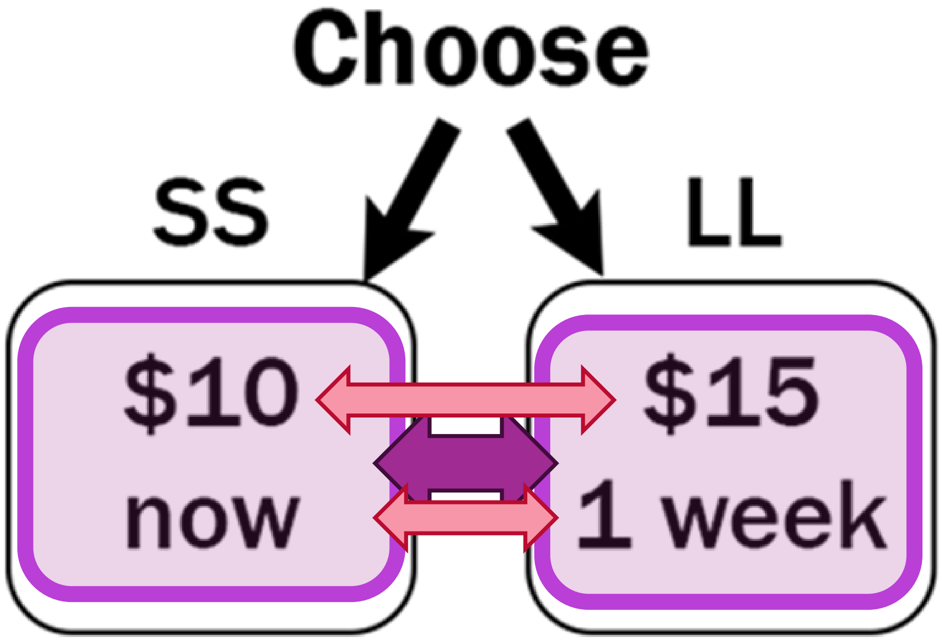
A couple other models

- **Hyperboloid discounting**

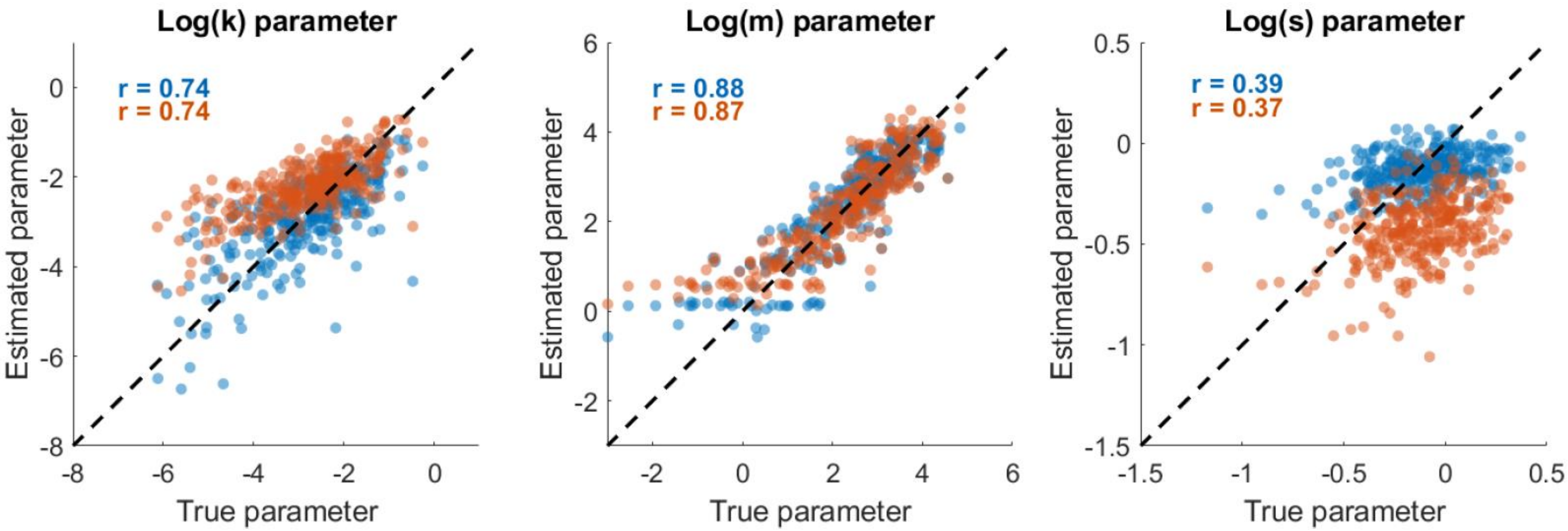
$$v(x, t) = \frac{x}{1 + (kt)^s}.$$

- **Direct difference**

$$v(x_1, x_2, t_1, t_2) = w \cdot (x_1^\alpha - x_2^\alpha) - (1 - w) \cdot (t_1^\nu - t_2^\nu).$$

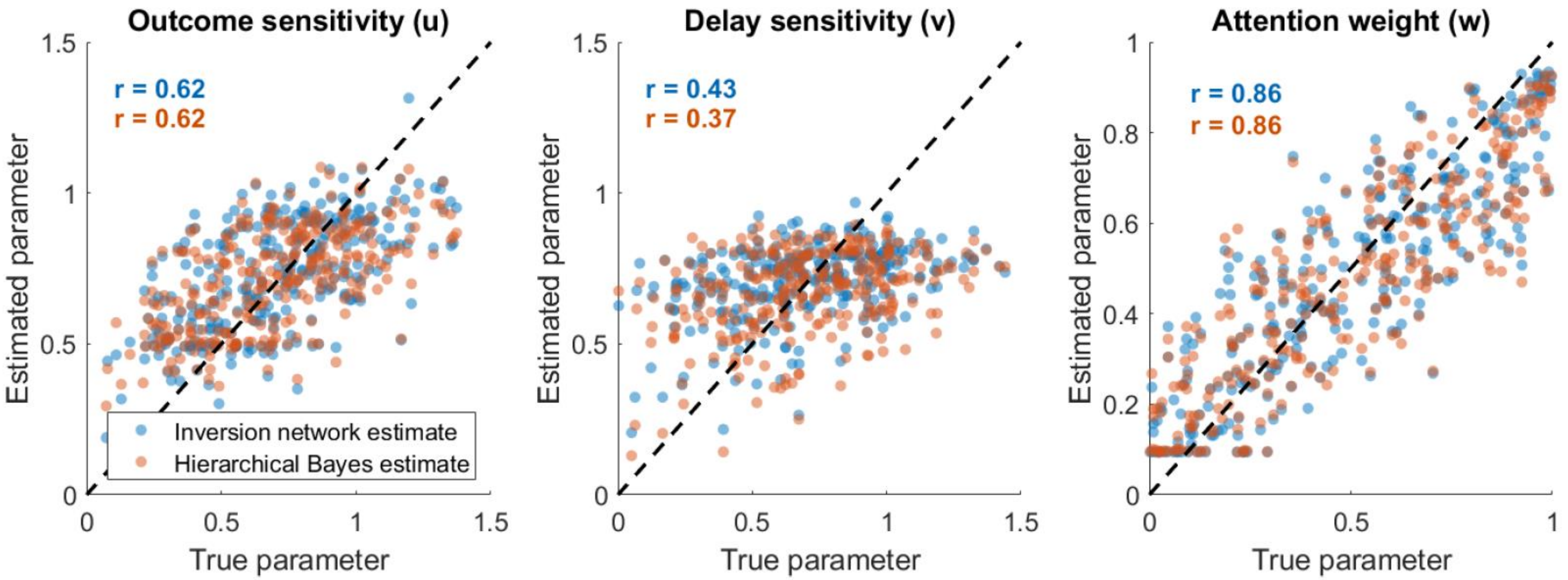


Hyperboloid parameter recovery



Kvam, P. D., Sokratous, K., Fitch, A., & Vassileva, J. (2024). Comparing likelihood-based and likelihood-free approaches to fitting and comparing models of intertemporal choice. PsyArXiv.

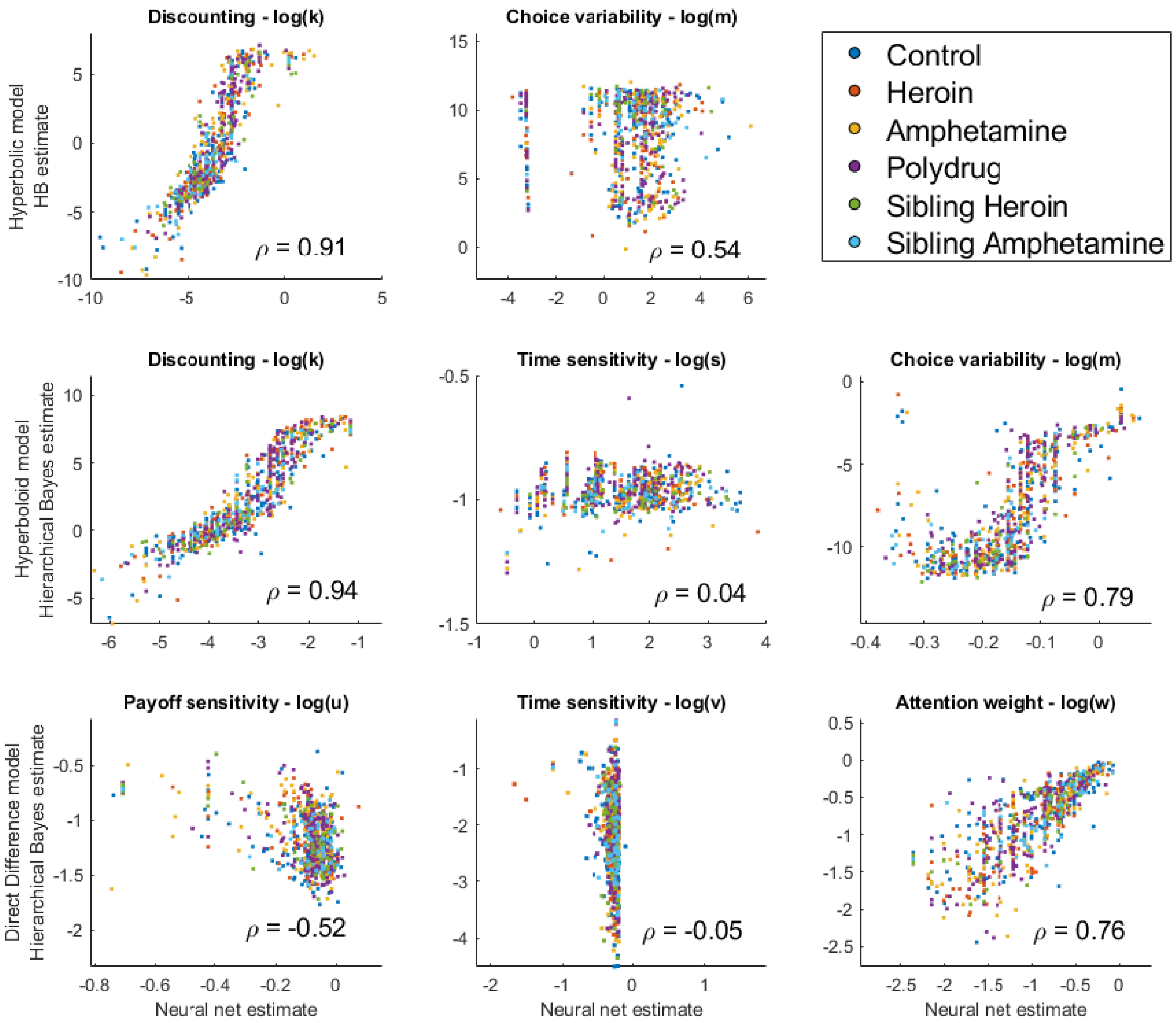
Direct difference parameter recovery



Kvam, P. D., Sokratous, K., Fitch, A., & Vassileva, J. (2024). Comparing likelihood-based and likelihood-free approaches to fitting and comparing models of intertemporal choice. PsyArXiv.

Application to real data set

- Do parameter estimates from the different approaches agree when we apply them?
- Generally, same conclusions about most important parameters
  - But, many parameters hard to estimate

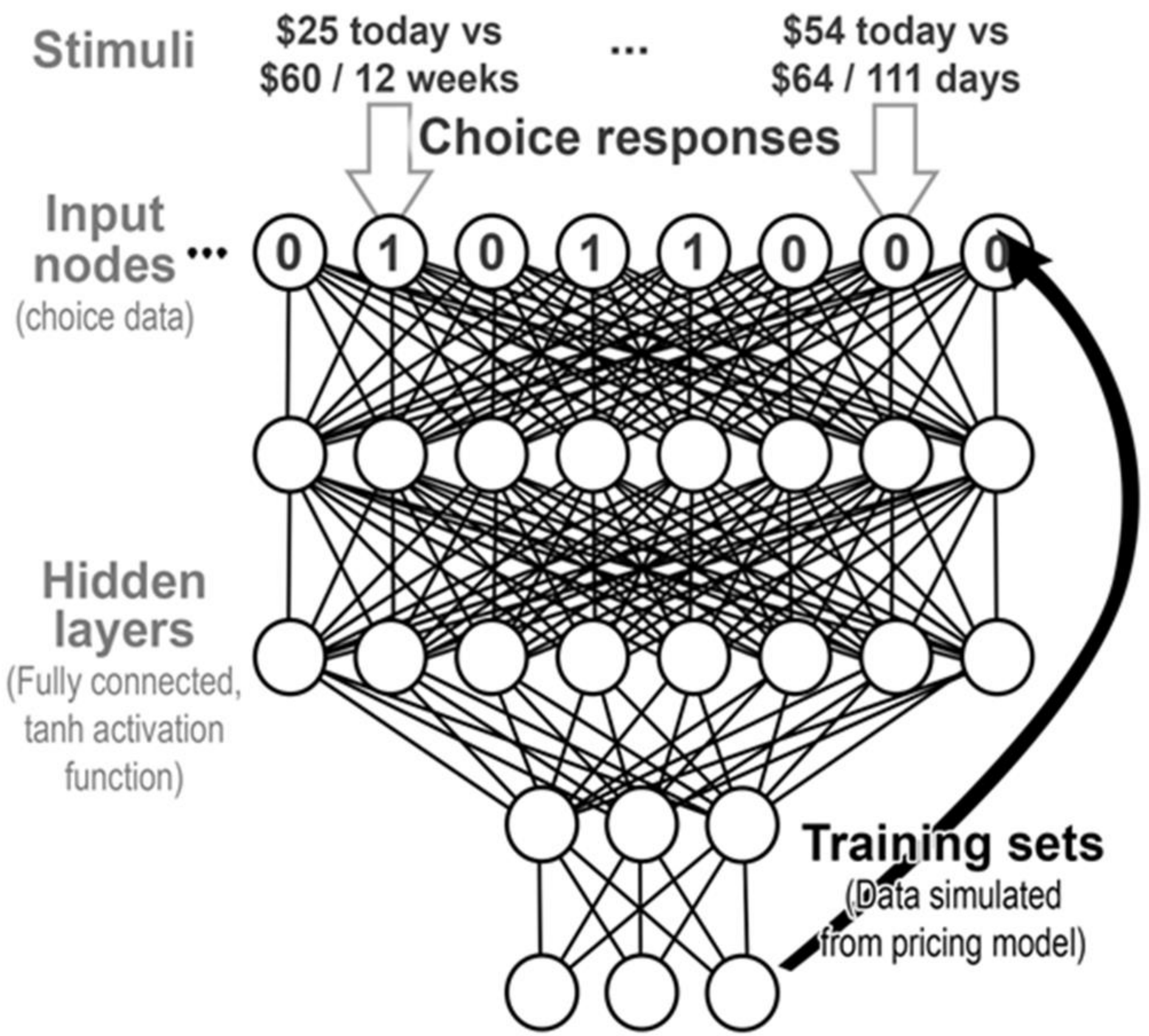


Take-home message #1

Neural network shows the **same level of performance** and reaches similar conclusions as hierarchical Bayesian methods for estimating model parameters

Model comparison

- Instead of parameters, the outputs are different models
  - Model comparison as a classification problem



Model comparison

Confusion matrix for neural network-based approach to model comparison.

		Inferred model		
		Direct difference	Hyperbolic	Hyperboloid
True model	Direct difference	[illegible]	[illegible]	22.16%
	Hyperbolic	[illegible]	[illegible]	32.82%
	Hyperboloid	[illegible]	1.66%	86.07%
	Total	100.83%	58.12%	141.05%

73.60%

Confusion matrix for DIC-based approach to model comparison.

		Inferred model		
		Direct difference	Hyperbolic	Hyperboloid
True model	Direct difference	[illegible]	[illegible]	28.20%
	Hyperbolic	[illegible]	[illegible]	5.39%
	Hyperboloid	[illegible]	36.31%	25.80%
	Total	95.64%	144.98%	59.38%

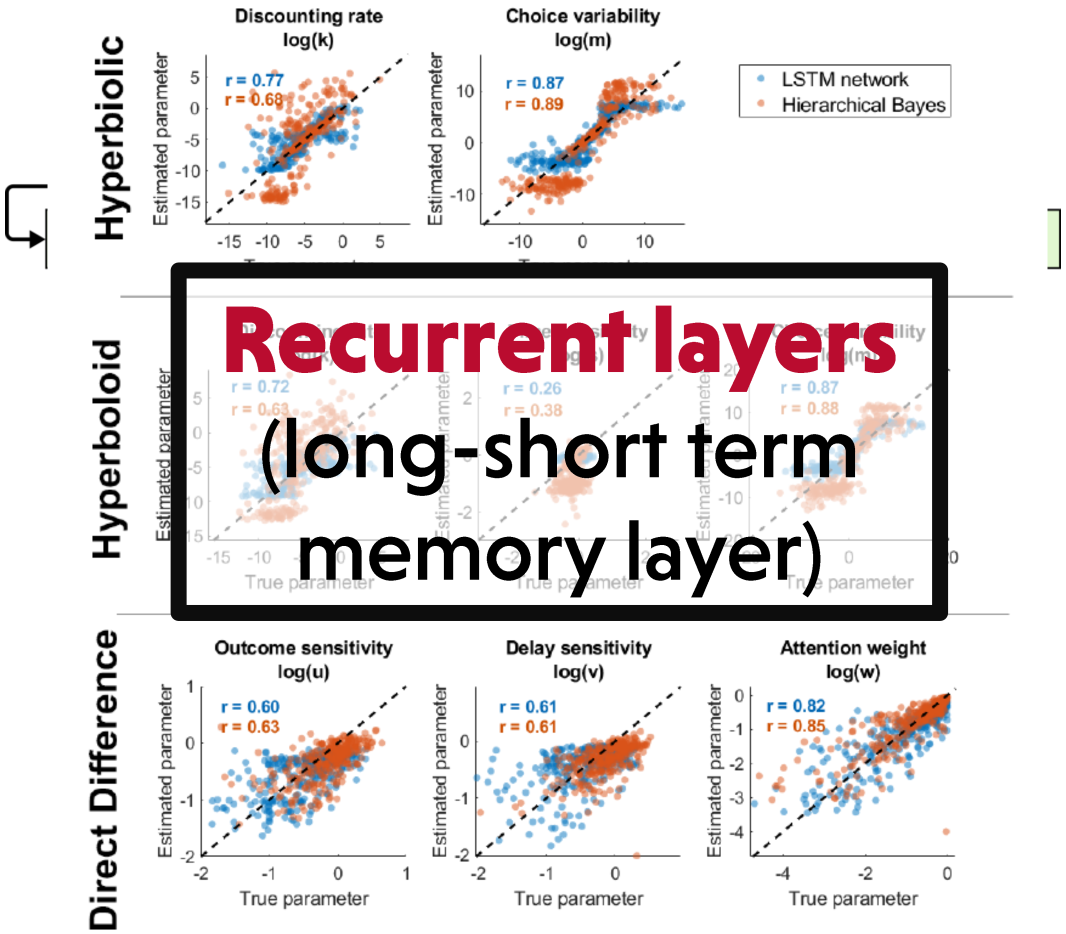
42.21%

Take-home message #2

Neural network shows the **much better** performance compared to model fit metrics when it comes to identifying the true underlying model!

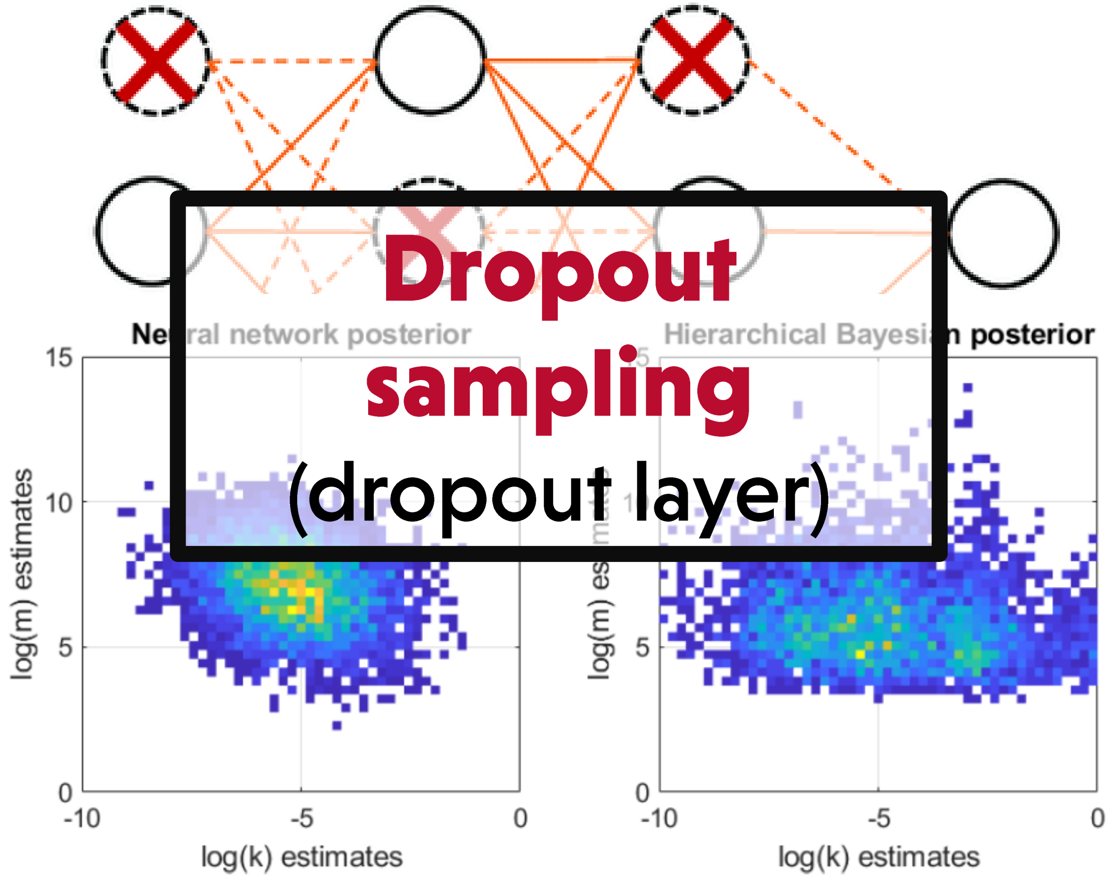
Two additional hurdles

- Fitting to data with varying designs / # of trials



Recurrent layers
(long-short term memory layer)

- Getting a measure of uncertainty (Bayesian posterior)



Dropout sampling
(dropout layer)

Outline

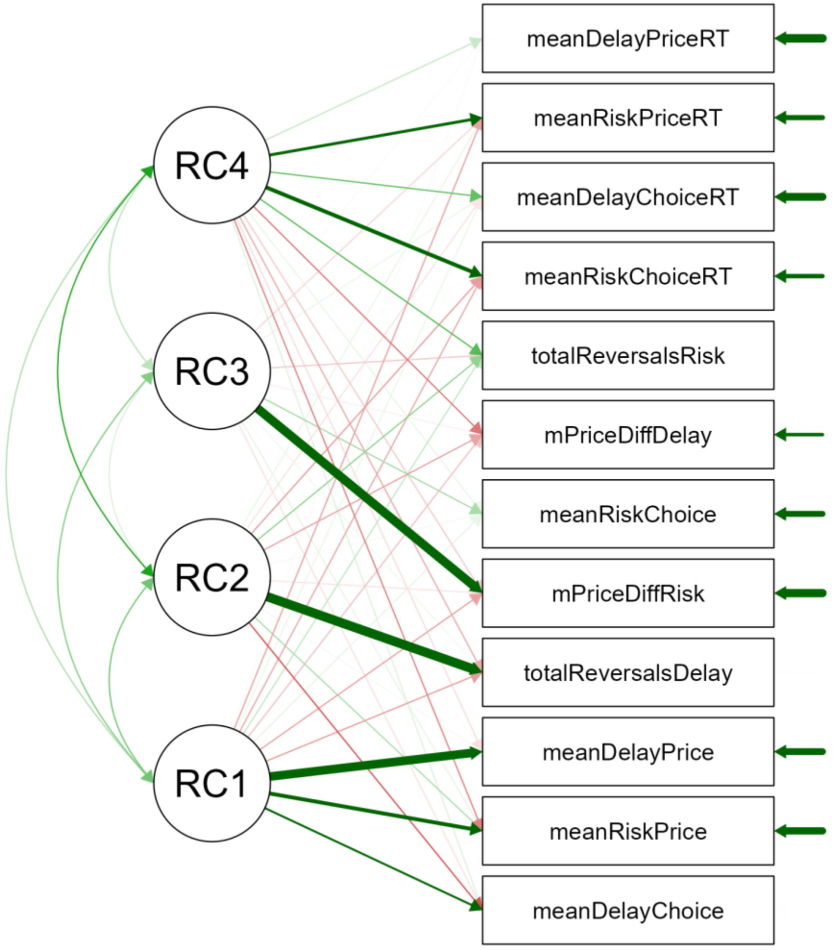
- Machine learning tools for **fitting and comparing models**
  - Issues with current approaches
  - Comparison with hierarchical Bayes
  - Illustration in intertemporal choice

- Machine learning tools for **discovering models**
  - Autoencoders for dimension reduction
  - Reinforcement learning, explicit learning, and computational evolution
  - Application to procrastination behavior

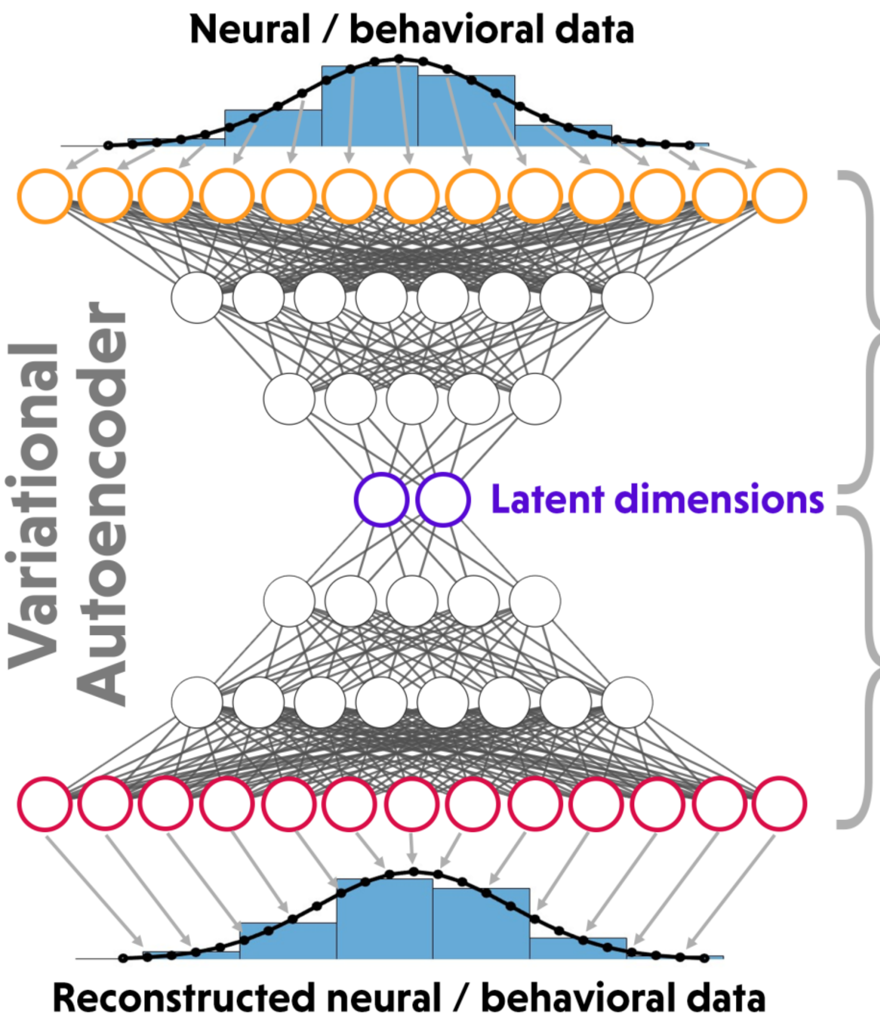
- Future directions

Exploratory data analysis

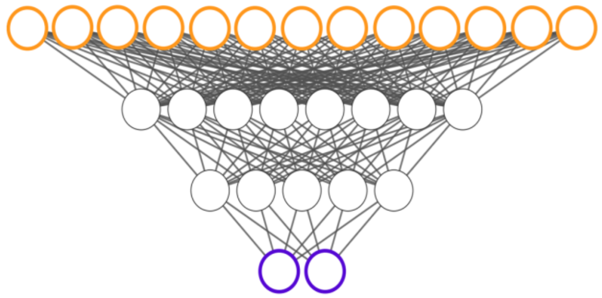
- Traditionally use approaches like **exploratory factor analysis** or **principal components analysis** to identify latent traits / processes
- However, these assume linear relationships between measures
  - Linear correlation / covariance matrix
- They can also create "phantom oscillations" (Shinn, 2023)



Autoencoders

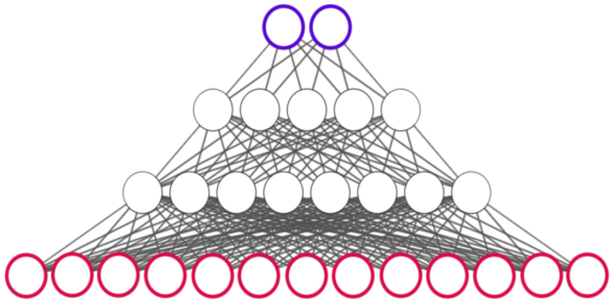


Encoder network



- Condense incoming data while preserving total information
- Identify latent constructs that predict / explain behavior
- Identify latent groups, new clusters (in combination with unsupervised learning)
- Exploratory data analysis

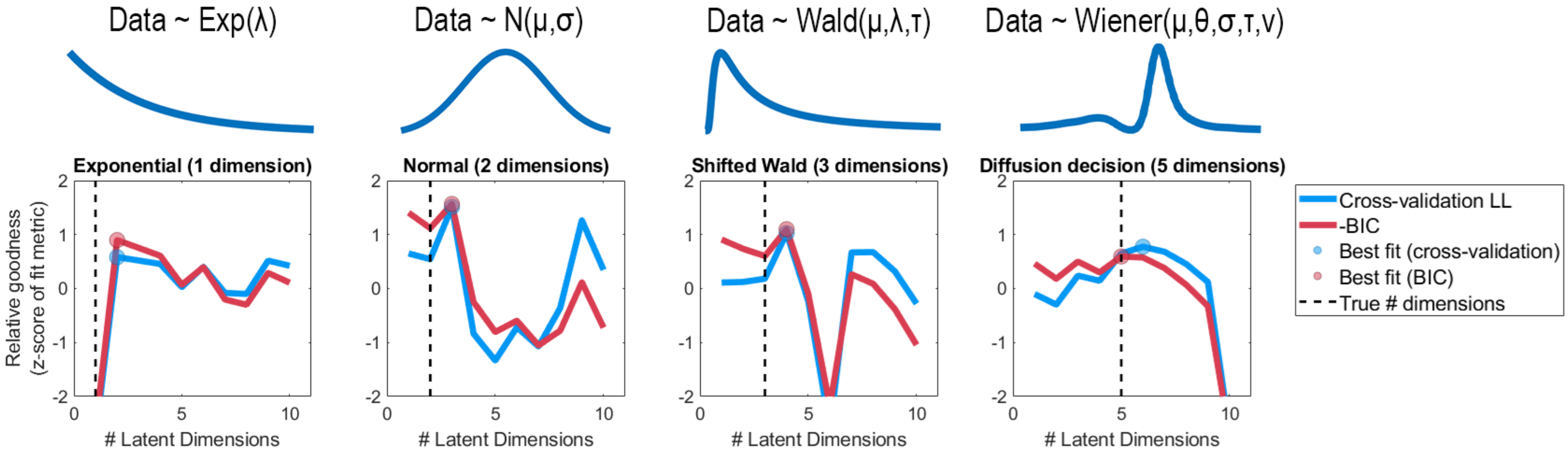
Decoder network



- Simulate expected data by varying latent dimensions
- Evaluate error in predictions
- Optimally impute / reconstruct missing data
- Explore nonlinear relationships between different outcome measures (via latent dimensions)

Does it work?

- Simulated data from a model (distribution) with a known # of parameters



- Used autoencoder to encode the data, and quantified fit using cross-validation & BIC

Take-home message #3

Autoencoders can **reduce dimensions of data** in a nonlinear way, while guaranteeing recoverability of the original inputs

AI as a model of behavior

- Many hypotheses about (e.g.) evolution are hard to test directly
  - We can't directly observe or manipulate human evolution!
  - Test hypotheses through simulation – **computational evolution**

- Others about development or learning over a lifetime are also hard to test without extensive tracking or longitudinal data
  - If a participant has done 10,000 instance of planning tasks in their life, how do we model that participant's history of learning?
  - **Reinforcement learning** models / simulations
  - Models of explicit / social learning: **supervised learning**

Procrastination

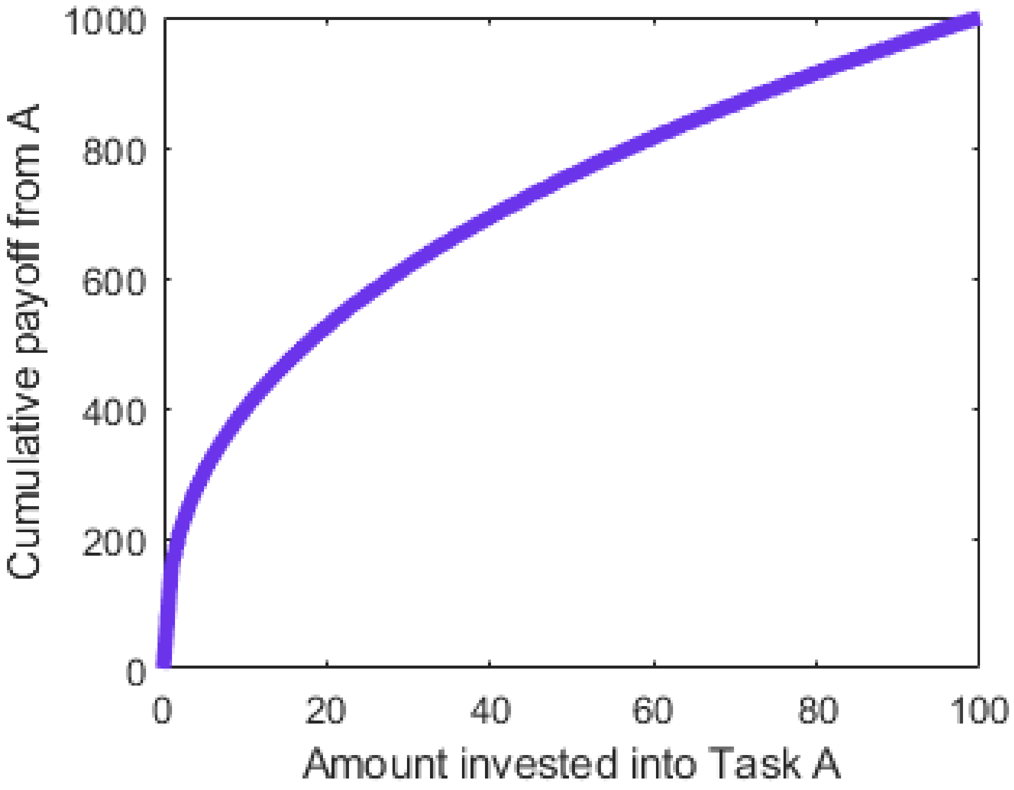
- Defined as investing in an easy task with **immediate payoffs** rather than another, **higher-payoff task** requiring more effort for payoffs to be realized
  - "Play" is immediately rewarding, but diminishing returns
  - "Work" is not immediately reward, requires protracted investment to complete and realize the benefits

- Economic definition as a resource allocation problem
- Allows for quantitative abstractions that make it easier to instantiate and analyze computationally

Task payoffs

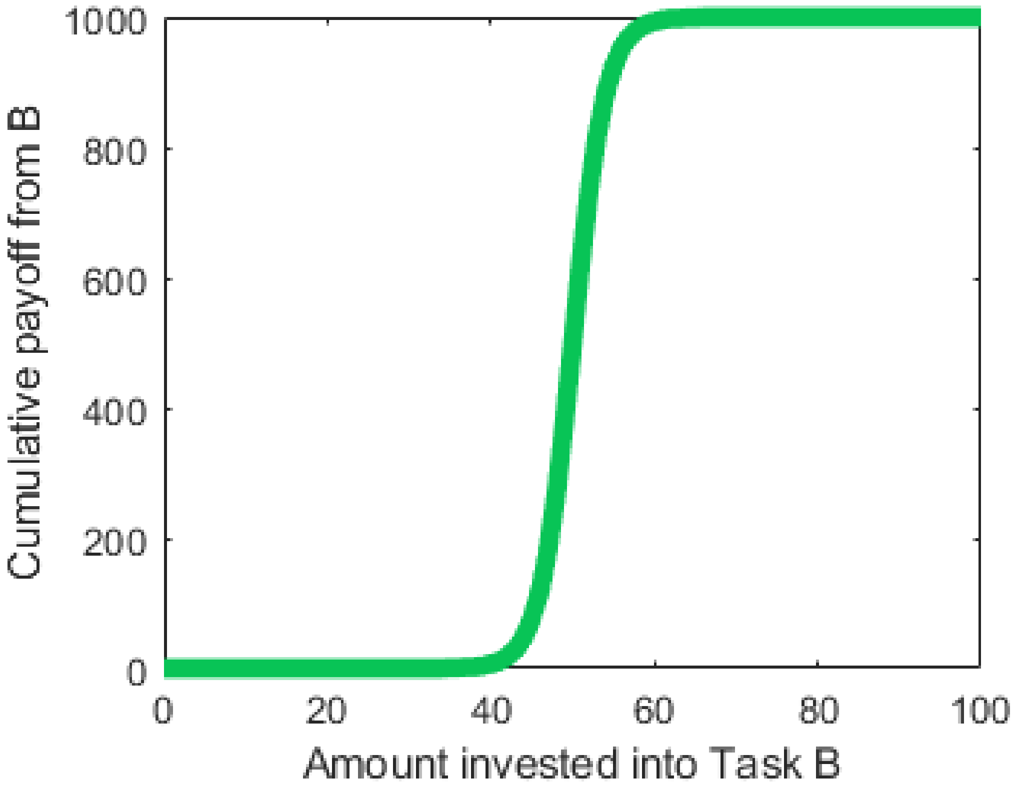
Task A

"Play"



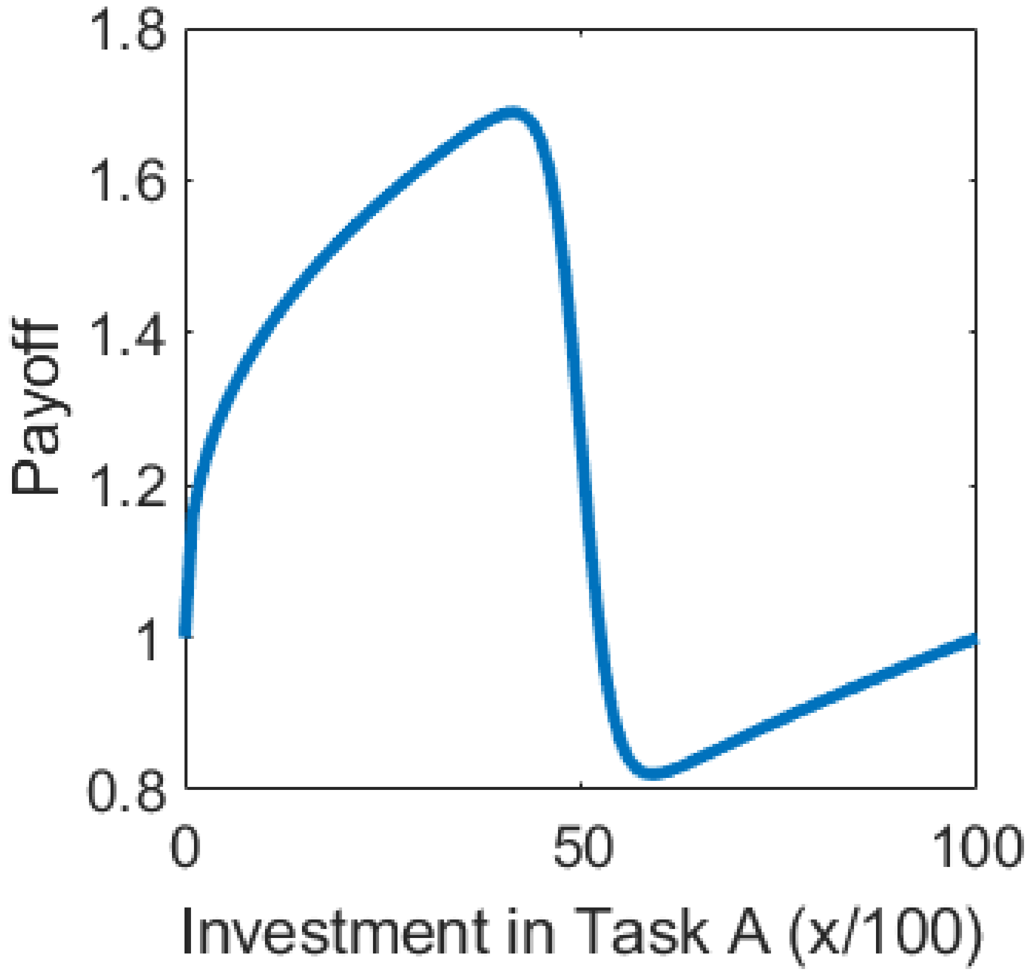
Task B

"Work"



Optimal solution

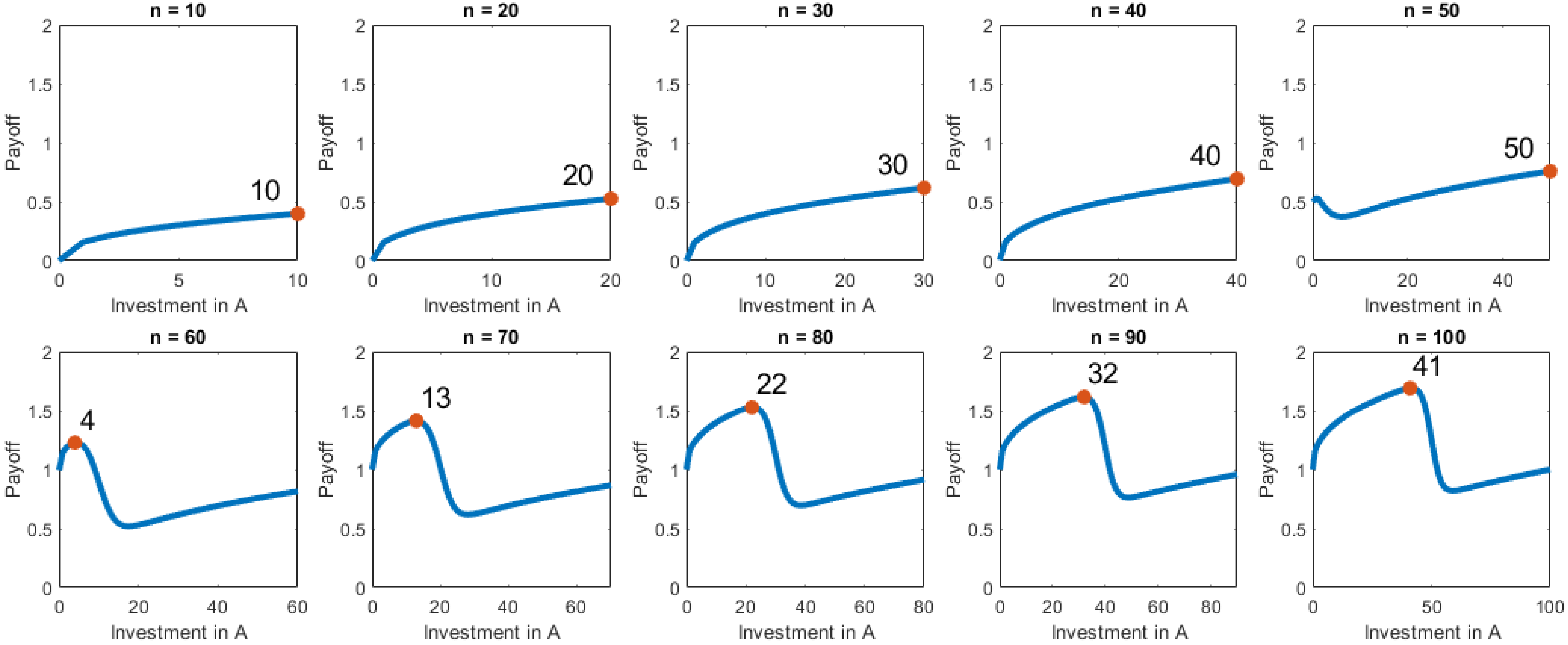
- Find the maximum value of f(#TaskA) + g(#TaskB)
- For a given total resource of time (e.g., t = 100), it is **simple to work out** the ideal balance of task investments
  - Here, a person should invest **41%** of their time into "play" task and 59% of their time into the "work" task
- So, why don't people do the optimal thing?



Where does procrastination come from?

- Incomplete optimization (learning / evolution)
  - Bias toward "safer" immediate payoff task
- Delay discounting
  - Task with later payoff assigned a lower value
- Unknown task payoff functions
- **Uncertainty about task duration and/or deadlines**

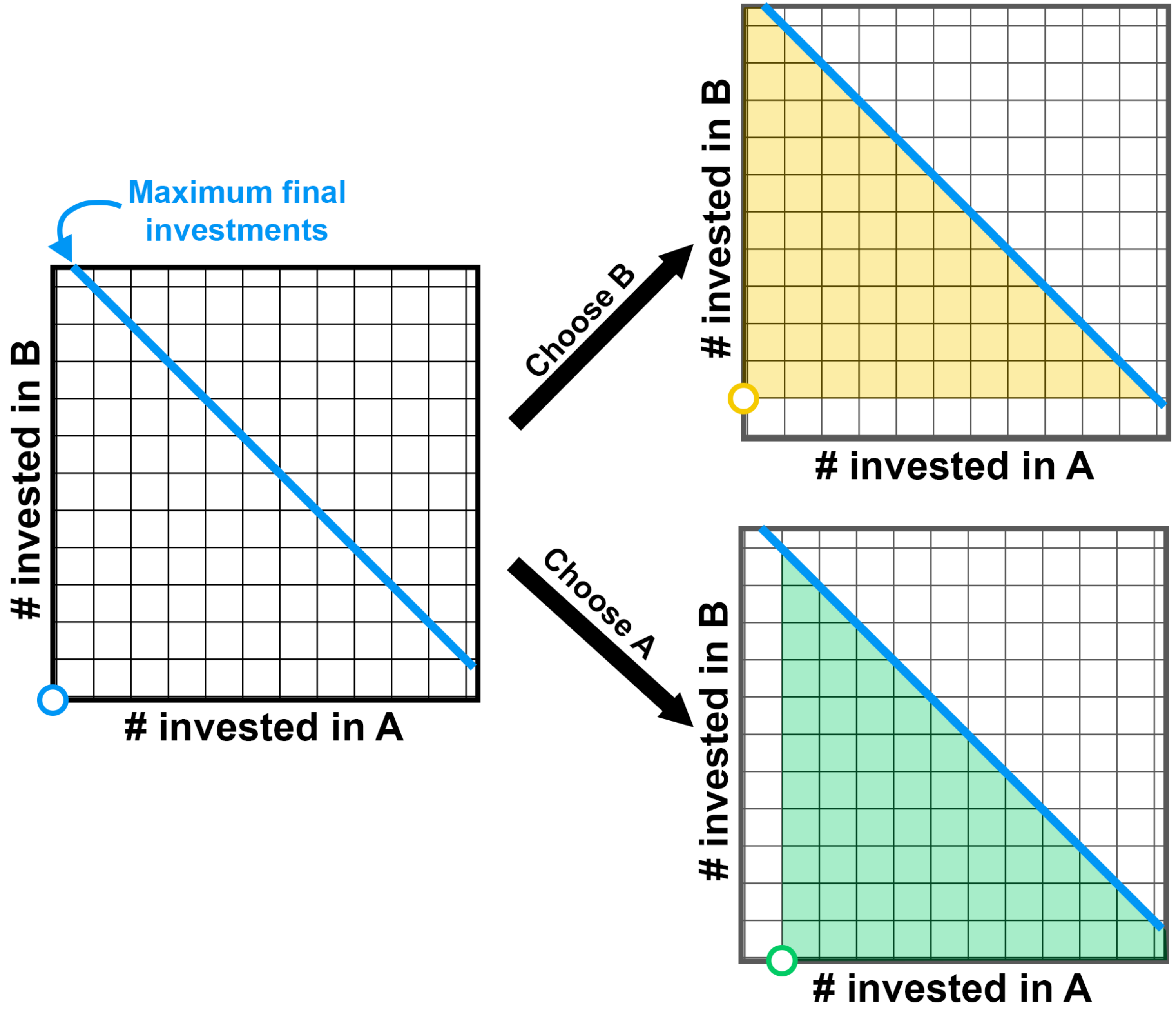
Optimality changes with time horizon



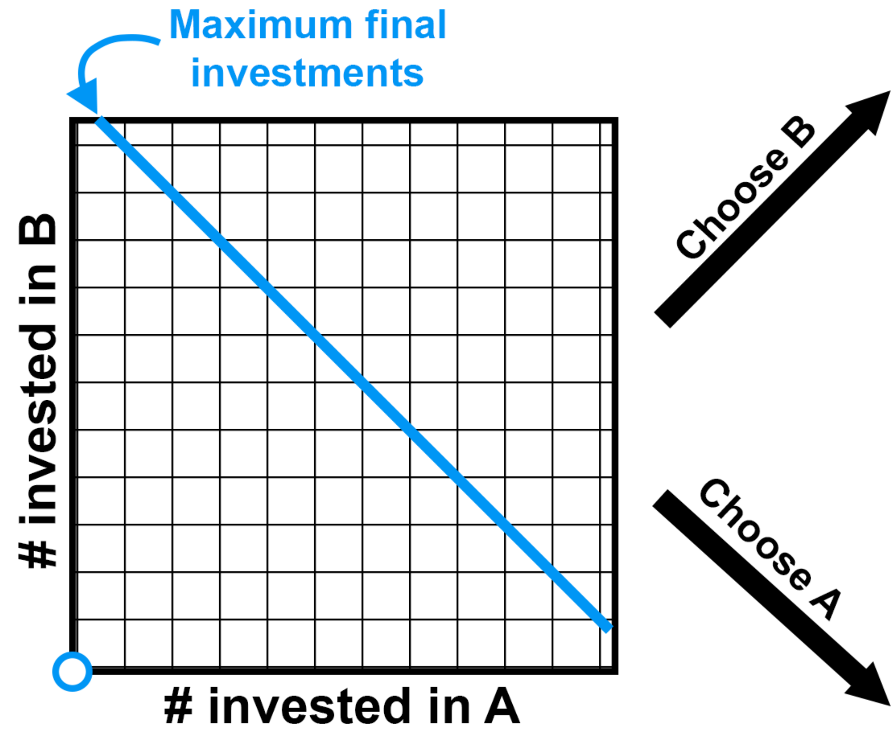
What if the amount of time is uncertain?

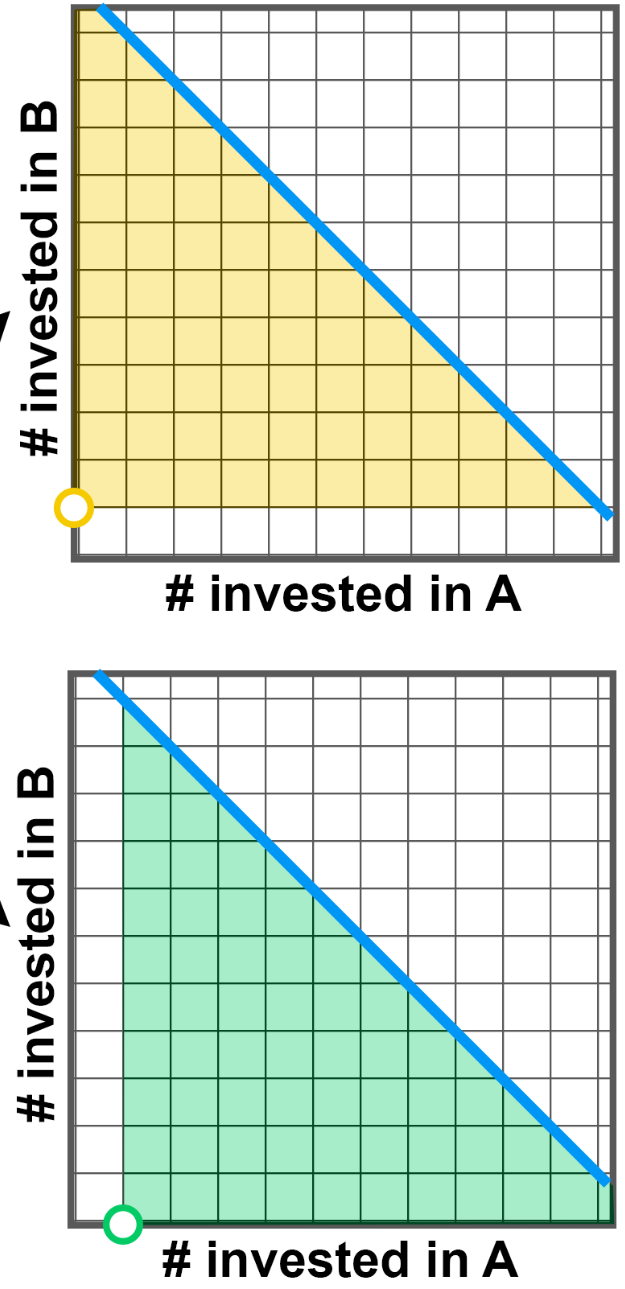
Working out the optimal solution

An experiment with real humans



Maximum final investments
invested in B
invested in A
Choose B
invested in B
invested in A
Choose A
invested in B
invested in A



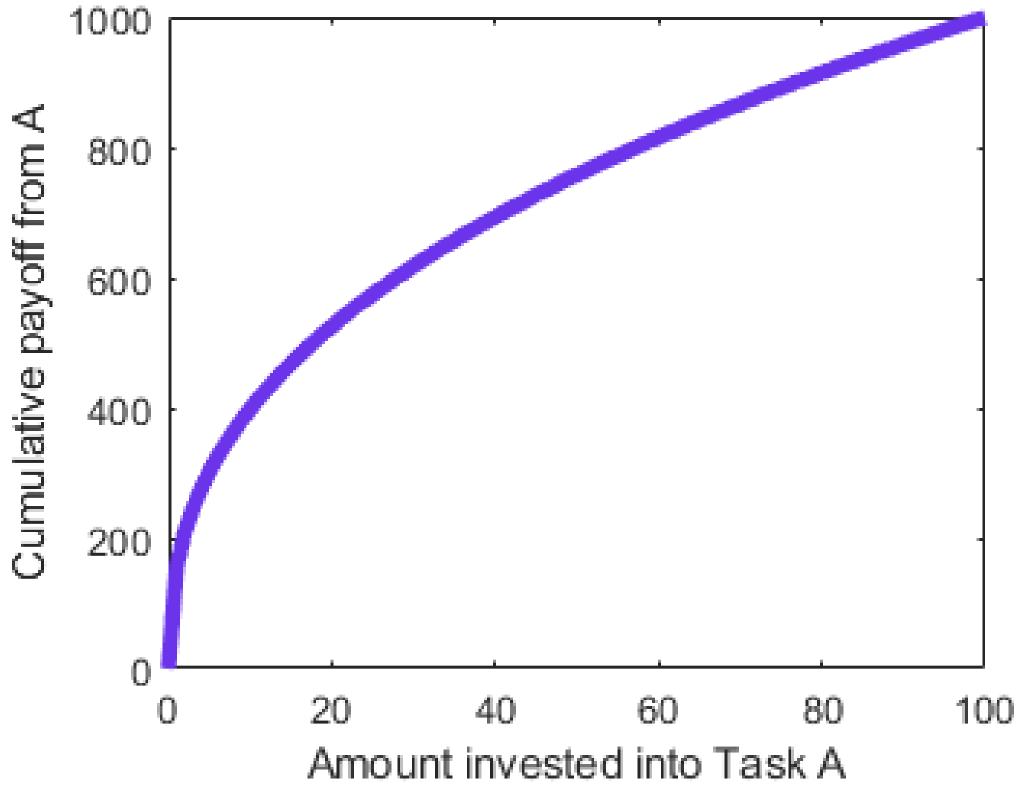


Exceptionally difficult to compute! Requires taking the sum / integral of reward function times achievability over all possible end states

Experimental task payoffs

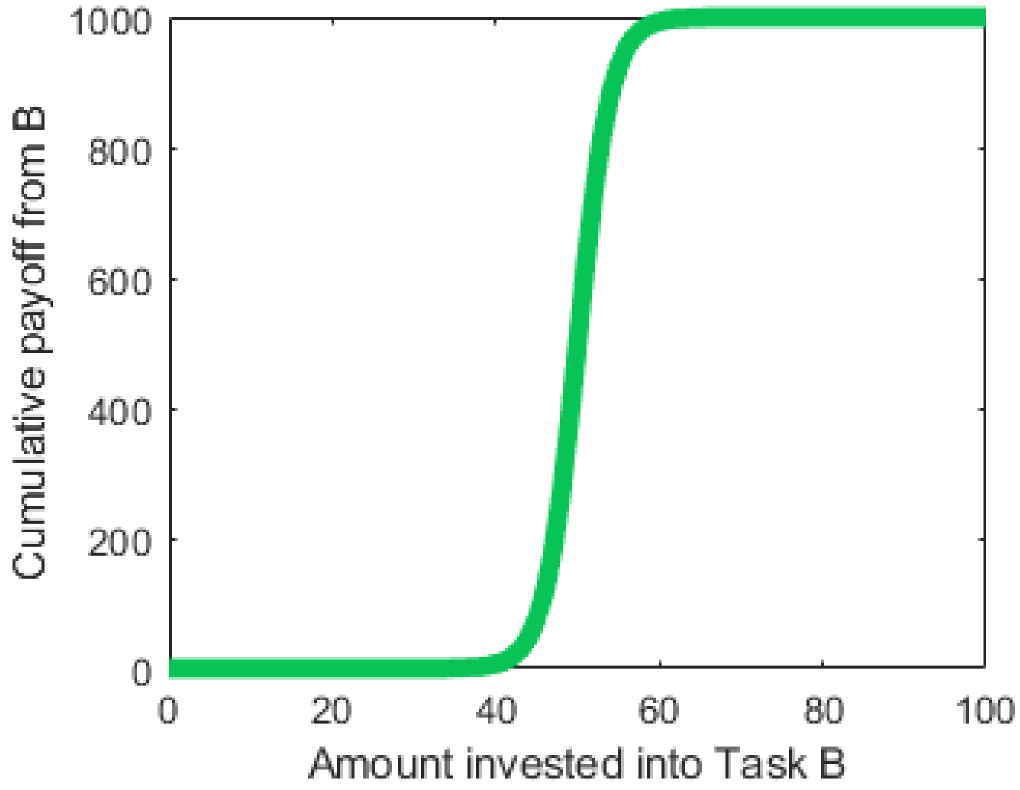
Task A

"Play"



Task B

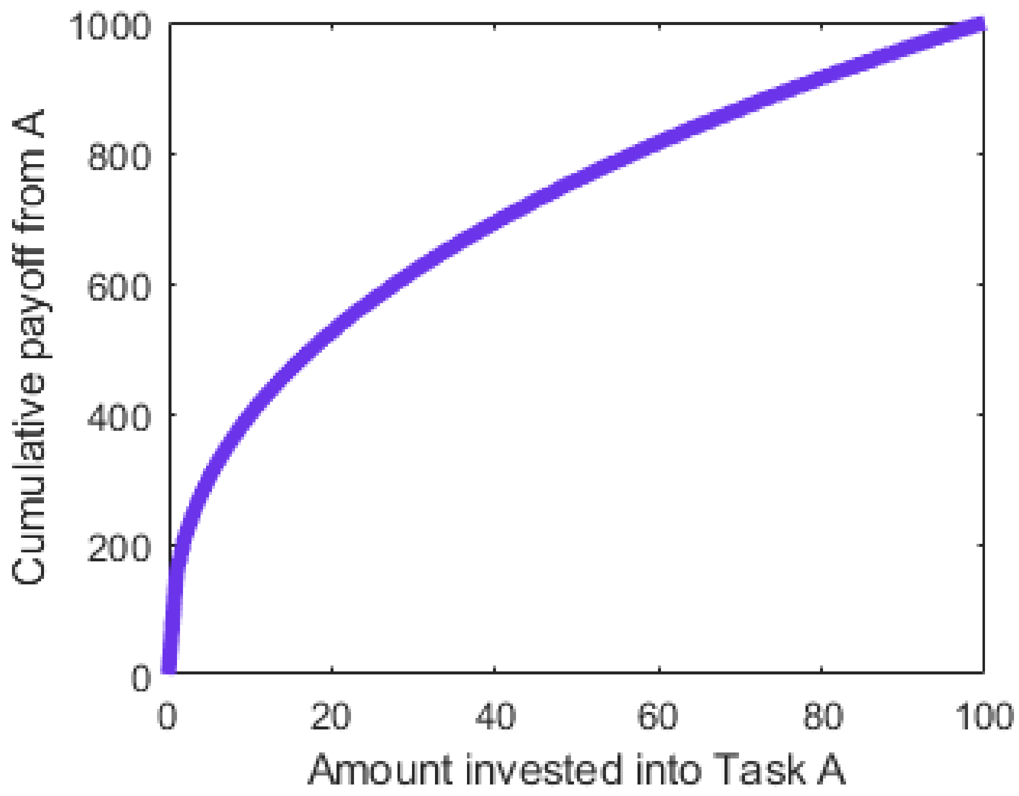
"Work"



Experimental task payoffs

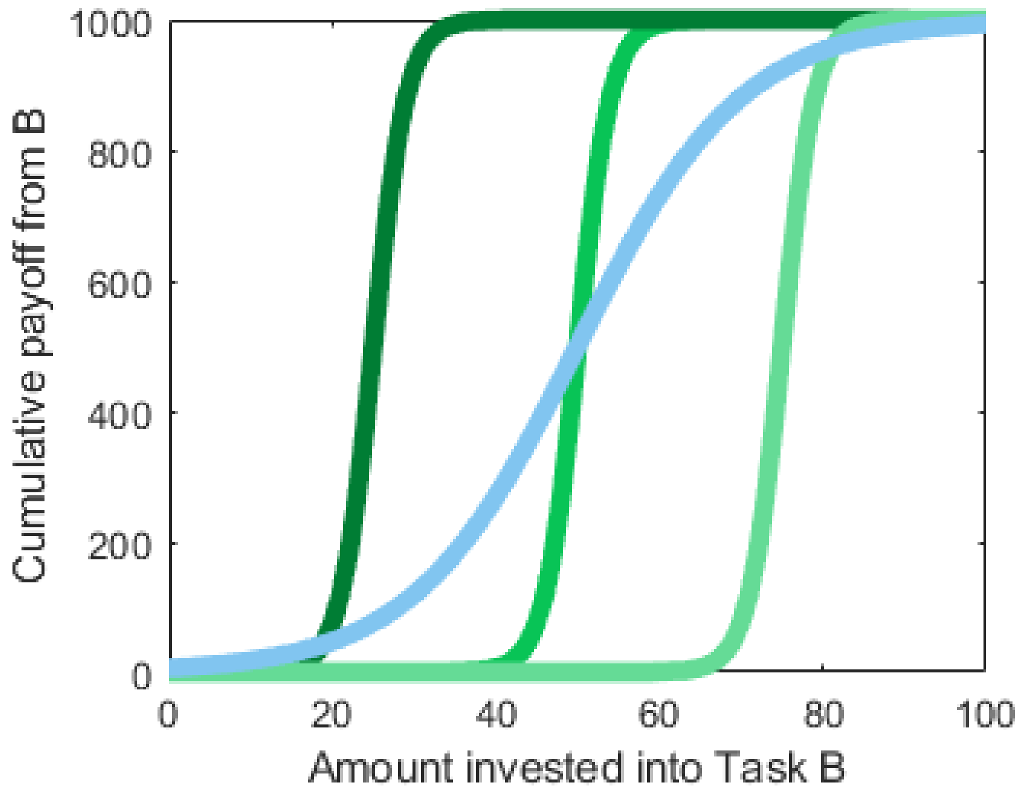
Task A

"Play"



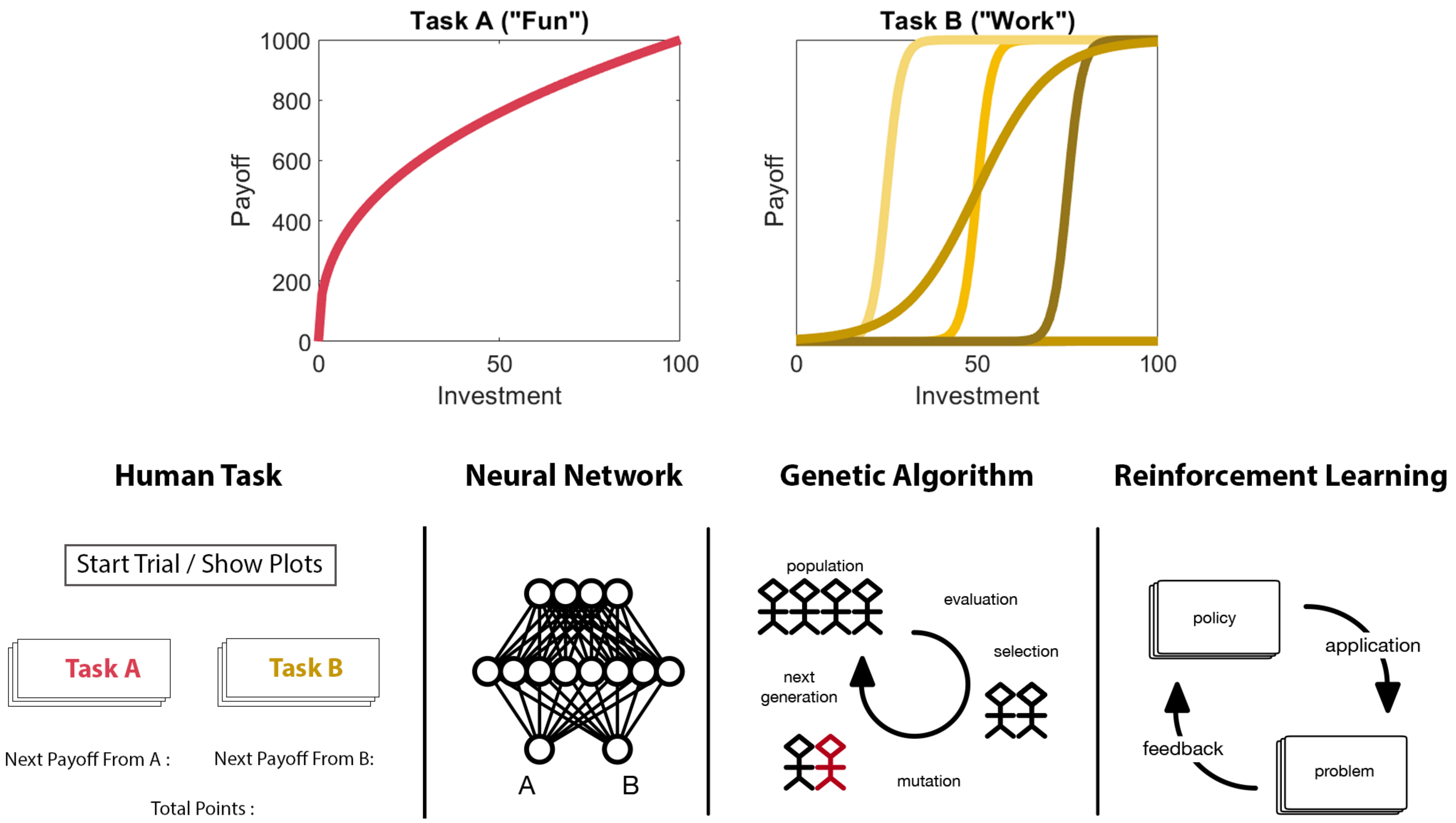
Task B

"Work"



Experimental task

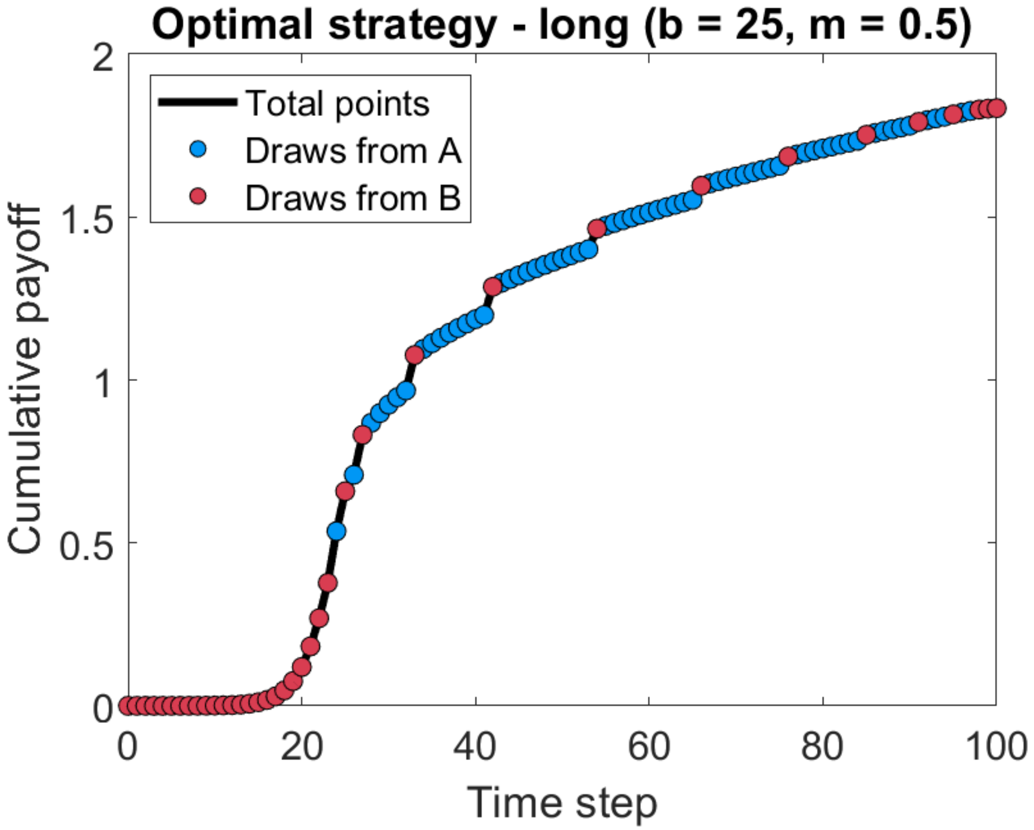
- **300 participants** completed the task for course credit
  - **261 retained** after applying exclusion criteria
- Task consisted of 24 trials
  - **4 different payoff functions** for Task B
  - **6 different time horizons:**
    - Exactly **30** time steps
    - Exactly **70** time steps
    - Exactly **100** time steps
    - **"Uniform":** # time steps draw from a uniform ranging from 1-100 time steps
    - **"Short":** # time steps drawn from a distribution favoring small numbers (1-100)
    - **"Long"**: # time steps drawn from a distribution favoring large numbers (1-100)

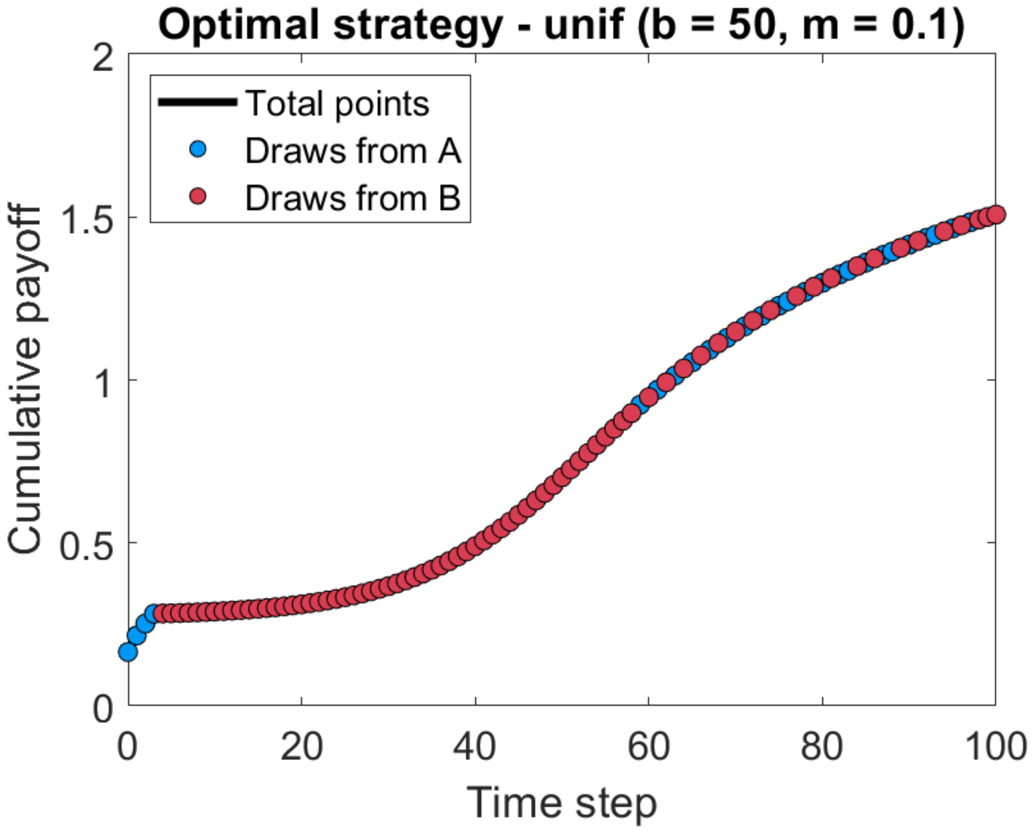

Task A ("Fun")
1000
800
600
400
200
0
Payoff
0
50
100
Investment
Task B ("Work")
Payoff
0
50
100
Investment
Human Task
Start Trial / Show Plots
Task A
Task B
Next Payoff From A :
Next Payoff From B:
Total Points :
Neural Network
A
B
Genetic Algorithm
population
evaluation
selection
next generation
mutation
Reinforcement Learning
policy
application
problem
feedback

Experimental task

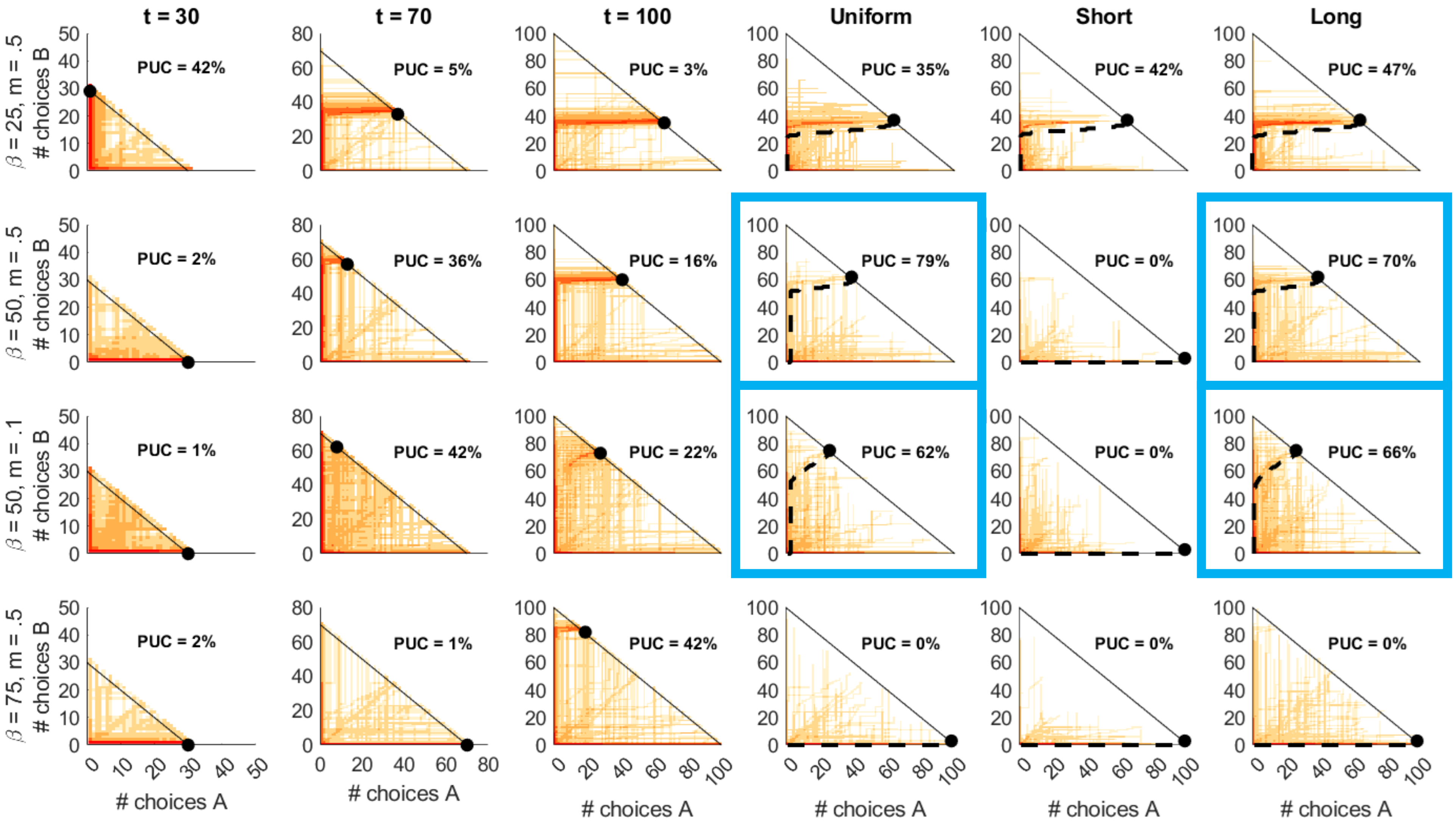
- **300 participants** completed the task for course credit
  - **261 retained** after applying exclusion criteria
- Task consisted of 24 trials
  - **4 different payoff functions** for Task B
  - **6 different time horizons:**
    - Exactly **30** time steps
    - Exactly **70** time steps
    - Exactly **100** time steps
    - **"Uniform":** # time steps draw from a uniform ranging from 1-100 time steps
    - **"Short":** # time steps drawn from a distribution favoring small numbers (1-100)
    - **"Long"**: # time steps drawn from a distribution favoring large numbers (1-100)

Example optimal trajectories

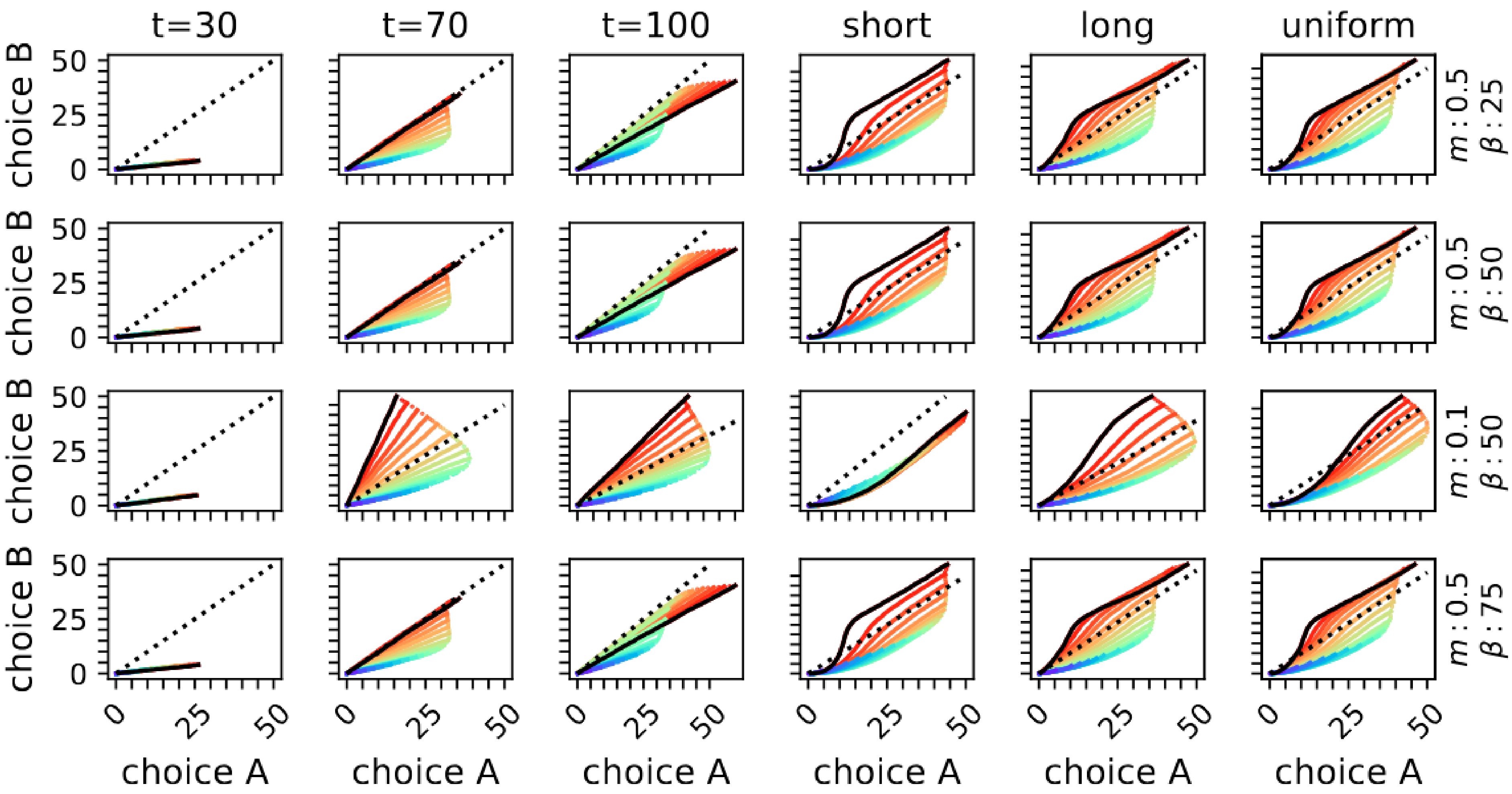




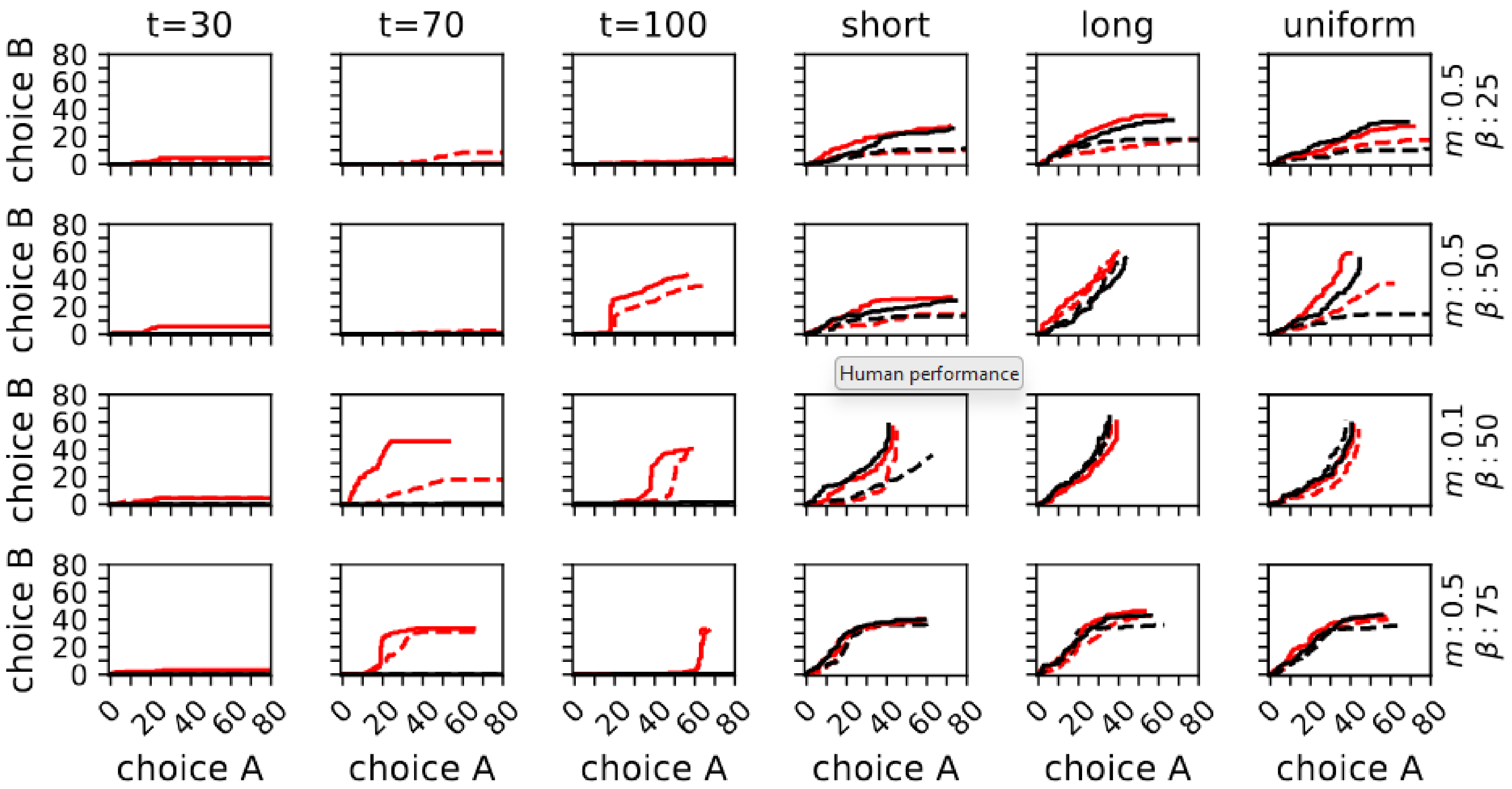
Human performance



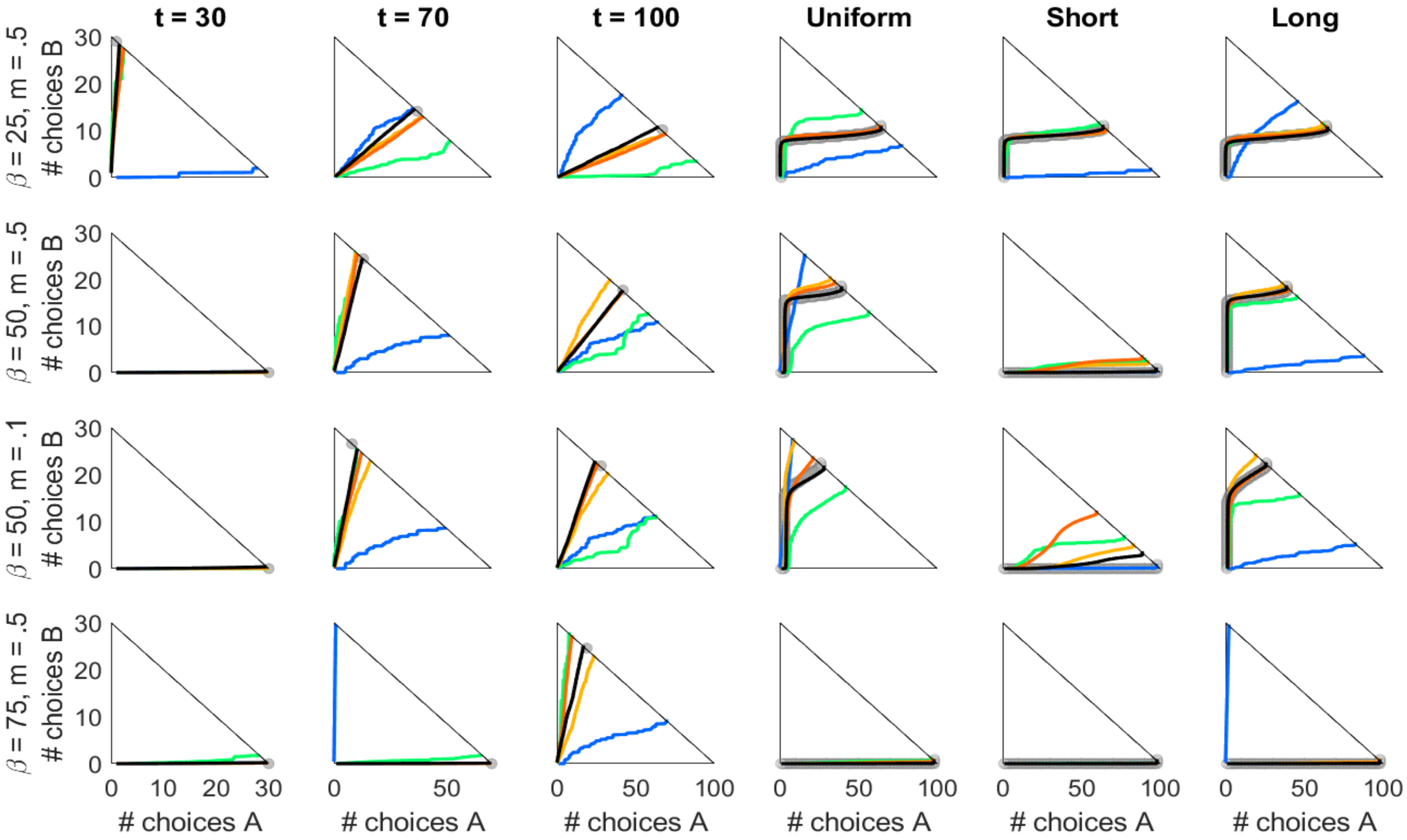
Machine performance - evolution



Machine performance – reinforcement learning



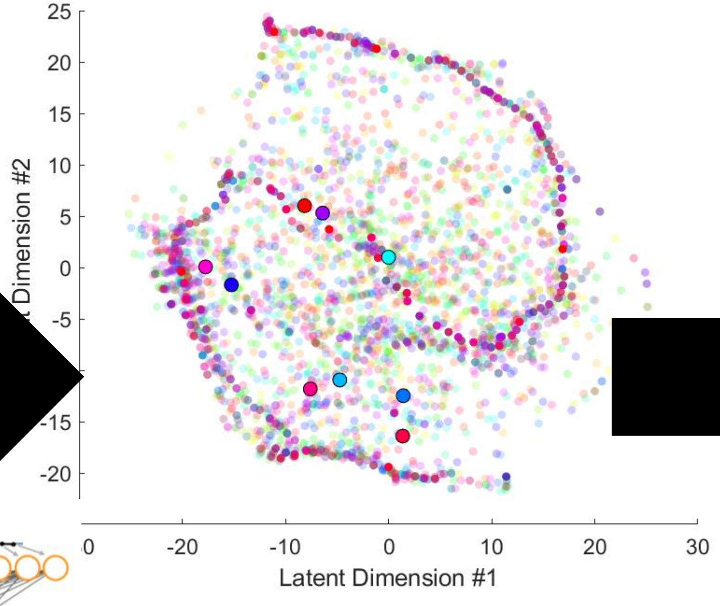
Machine performance – explicit learning



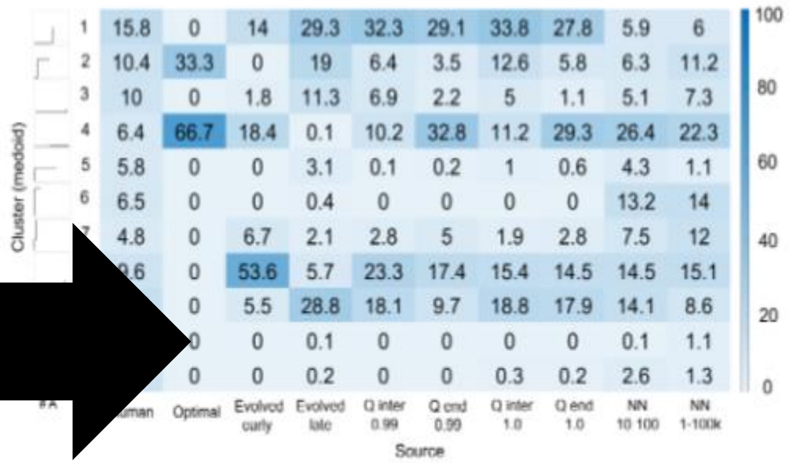
Encoding & clustering

Raw behavioral data

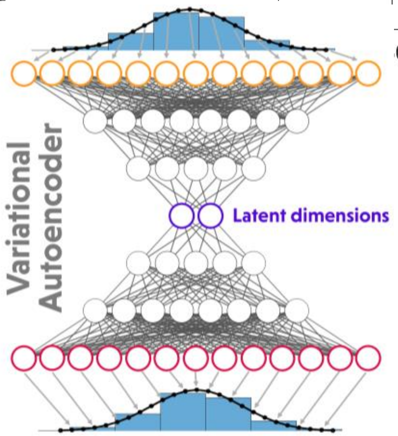
Encoded behavioral data

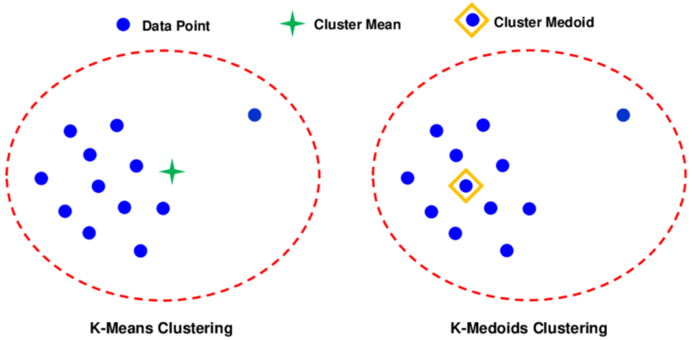


Clusters of behavioral data



Autoencoder to encode into latent dimensions



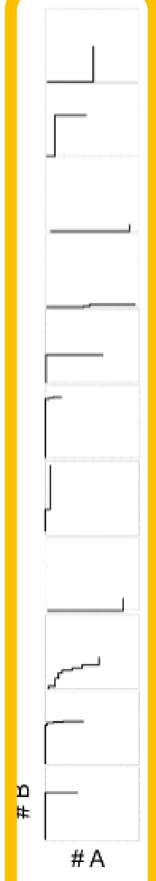


Cluster into latent groups using k-medoids

Clusters of human data

Procrastination behavior

Medoids



Cluster (medoid)

Cluster	t = 30, m = .5, β 25	t = 30, m = .5, β 50	t = 30, m = .5, β 75	t = 30, m = .1, β 50	t = 70, m = .5, β 25	t = 70, m = .5, β 50	t = 70, m = .5, β 75	t = 70, m = .1, β 50	t = 100, m = .5, β 25	t = 100, m = .5, β 50	t = 100, m = .5, β 75	t = 100, m = .1, β 50
1	10	72	78	37	3	2	4	6	1	5	8	8
2	4	5	2	11	7	11	5	14	11	9	2	13
3	4	5	6	10	1	6	4	11	1	9	10	11
4	0	0	0	0	1	1	1	0	8	20	26	10
5	14	2	2	6	21	0	0	2	15	0	1	3
6	0	0	0	0	2	4	11	26	3	4	34	27
7	3	2	1	4	6	2	2	2	4	2	2	1
8	0	0	0	1	6	26	62	10	2	1	4	2
9	8	11	9	16	8	11	13	15	9	9	13	14
10	0	0	0	0	3	36	0	14	3	41	0	10
11	56	3	2	15	41	0	0	0	42	0	0	1

Cluster	Uniform, m = .5, β 25	Uniform, m = .5, β 50	Uniform, m = .5, β 75	Uniform, m = .1, β 50	Short, m = .5, β 25	Short, m = .5, β 50	Short, m = .5, β 75	Short, m = .1, β 50	Long, m = .5, β 25	Long, m = .5, β 50	Long, m = .5, β 75	Long, m = .1, β 50
1	4	15	17	9	3	22	26	14	3	8	16	7
2	11	14	11	18	12	17	17	19	8	13	5	13
3	6	17	19	12	10	20	21	16	3	11	12	14
4	3	13	14	4	1	4	4	0	4	13	21	4
5	14	1	0	4	13	3	2	6	16	5	2	5
6	0	1	2	9	0	0	1	3	2	3	6	17
7	17	3	2	10	20	2	1	10	12	1	2	4
8	2	13	18	5	4	11	15	6	4	12	20	7
9	16	14	16	19	18	16	12	19	9	12	13	15
10	5	7	0	7	2	3	1	3	3	21	2	11
11	22	1	0	3	16	2	1	4	35	1	1	4

Condition

Colour scale: 0, 20, 40, 60, 80, 100

Clusters of machine data

Procrastination behavior

Cluster (medoid)	Human	Optimal	Evolved early	Evolved late	Q inter 0.99	Q end 0.99	Q inter 1.0	Q end 1.0	NN 10-100	NN 1-100k
1	15.8	0	14	29.3	32.3	29.1	33.8	27.8	5.9	6
2	10.4	33.3	0	19	6.4	3.5	12.6	5.8	6.3	11.2
3	10	0	1.8	11.3	6.9	2.2	5	1.1	5.1	7.3
4	6.4	66.7	18.4	0.1	10.2	32.8	11.2	29.3	26.4	22.3
5	5.8	0	0	3.1	0.1	0.2	1	0.6	4.3	1.1
6	6.5	0	0	0.4	0	0	0	0	13.2	14
7	4.8	0	6.7	2.1	2.8	5	1.9	2.8	7.5	12
8	9.6	0	53.6	5.7	23.3	17.4	15.4	14.5	14.5	15.1
9	13.1	0	5.5	28.8	18.1	9.7	18.8	17.9	14.1	8.6
10	7.2	0	0	0.1	0	0	0	0	0.1	1.1
11	10.5	0	0	0.2	0	0	0.3	0.2	2.6	1.3

#B #A

Source

Take-home messages

#4: People procrastinate when they are **uncertain about time** - and so do machines!

#5: Procrastination occurs in **evolved agents and RL agents with temporal discounting**

Outline

- Machine learning tools for **fitting and comparing models**
  - Issues with current approaches
  - Comparison with hierarchical Bayes
  - Illustration in intertemporal choice

- Machine learning tools for **discovering models**
  - Autoencoders for dimension reduction
  - Reinforcement learning, explicit learning, and computational evolution
  - Application to procrastination behavior

- **Current / Future directions**

Current / future directions

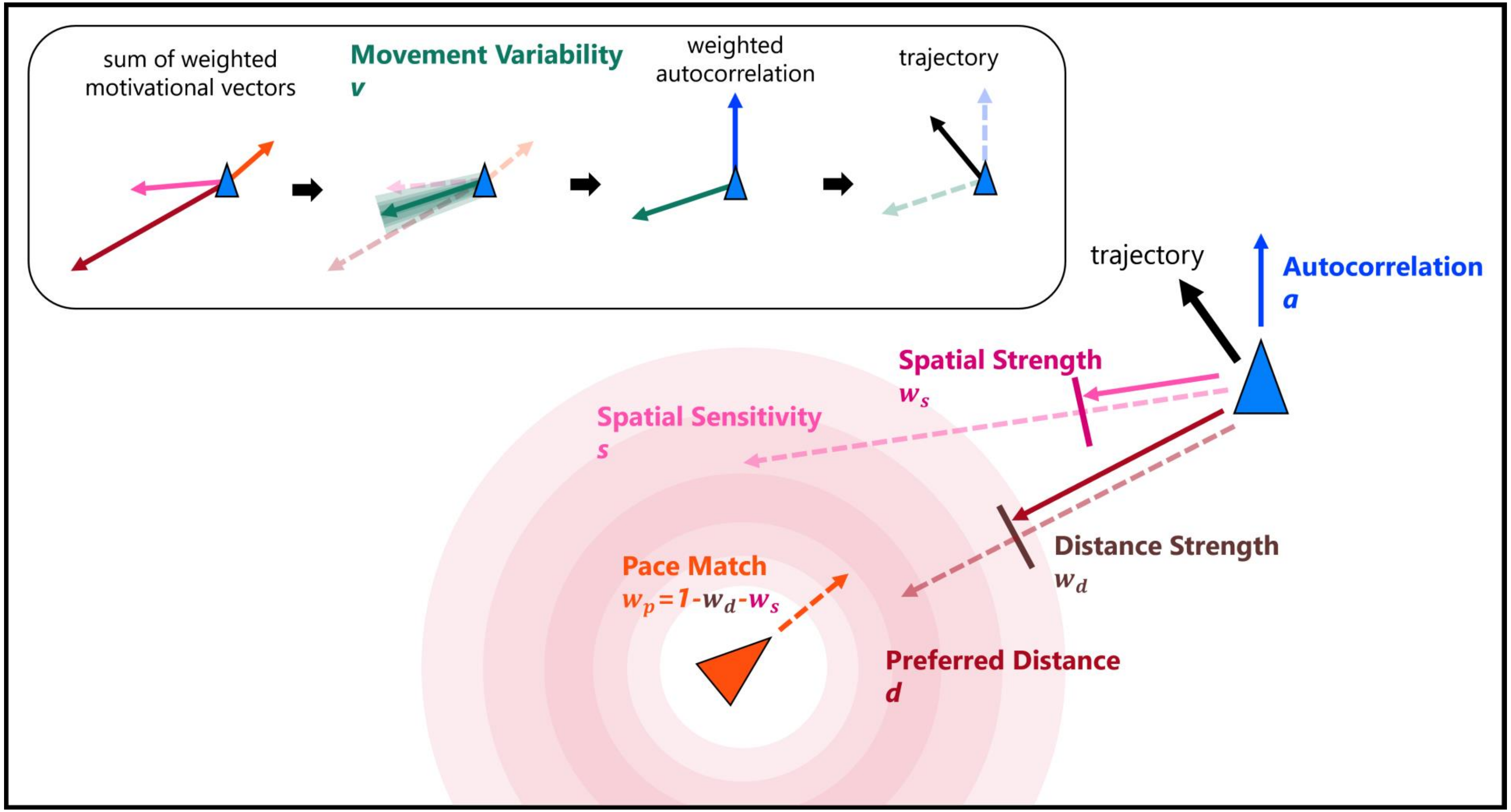
- **Real-time intent inference** using cognitive models
  - AI is socially incompetent – typically doesn't understand **what you want**
  - Equipping AI with efficient cognitive models allows them to make inferences about latent psychological processes (goals, intents)

Current / future directions

- Real-time intent inference using cognitive models
  - AI is socially incompetent – typically doesn't understand what you want
  - Equipping AI with efficient cognitive models allows them to make inferences about latent psychological processes (goals, intents)

- Uses networks to fit approach-avoidance models in real time
  - Predicts latent goals from a continuous stream of task behavior
    - Player vs Computer (75 participants + eye tracking)
    - Player vs Player (LAN connection, currently collecting)
    - Third-person assessment of players' goals

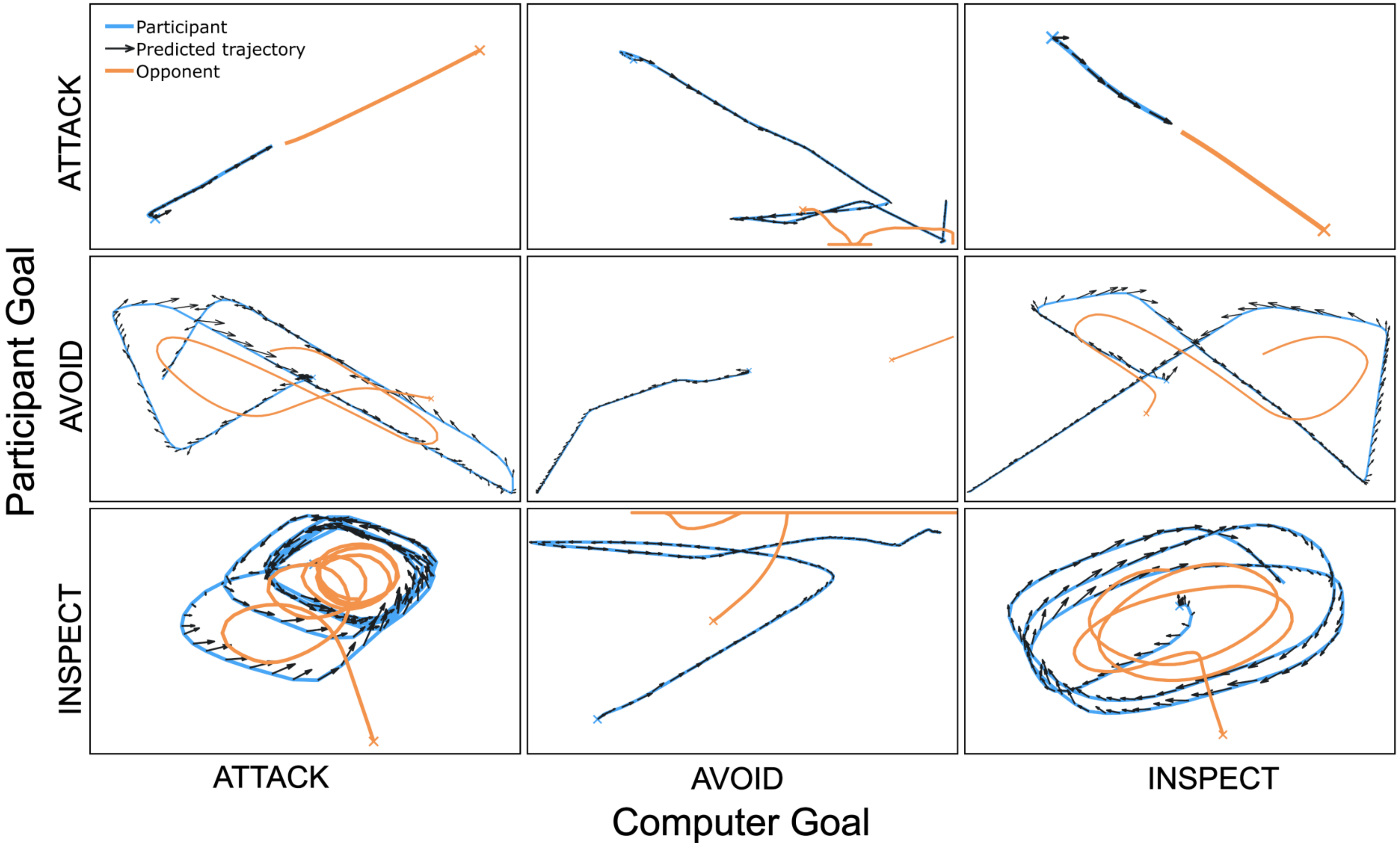
Global-local objective pursuit model



Intent inference model fitting

- Time series data from a single trial (10 seconds)
  - Typically ~130-160 pairs of positions [player, opponent]
  - Used a neural network to estimate GLOP parameters based on data from individual trials

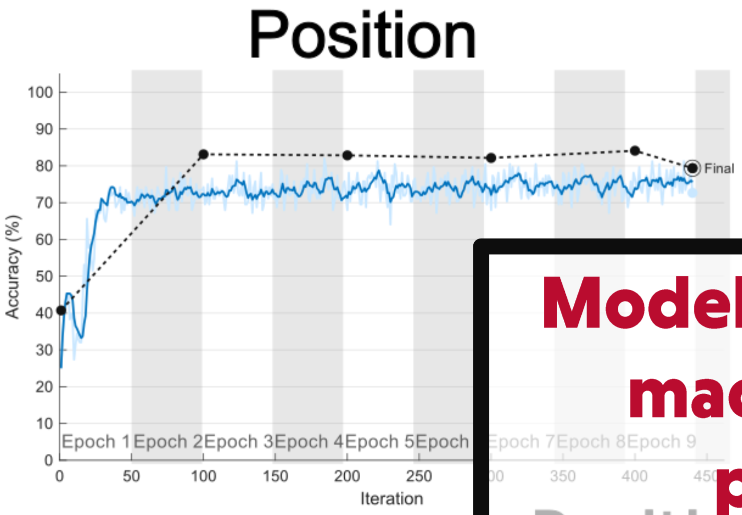
Real-time intent inference

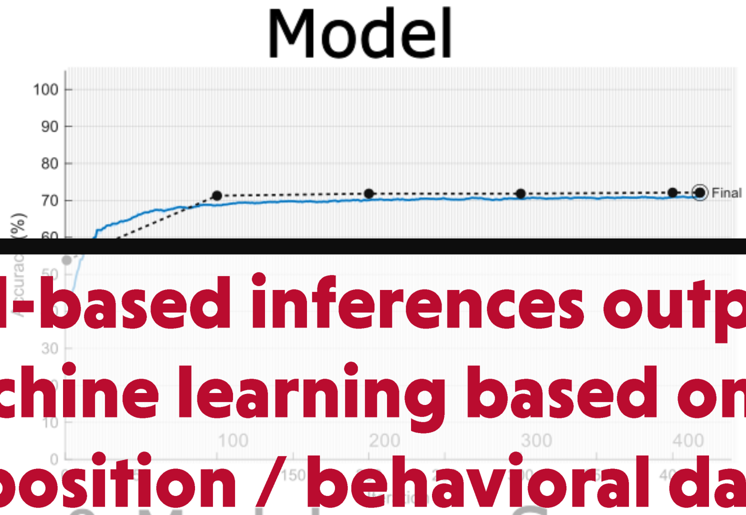


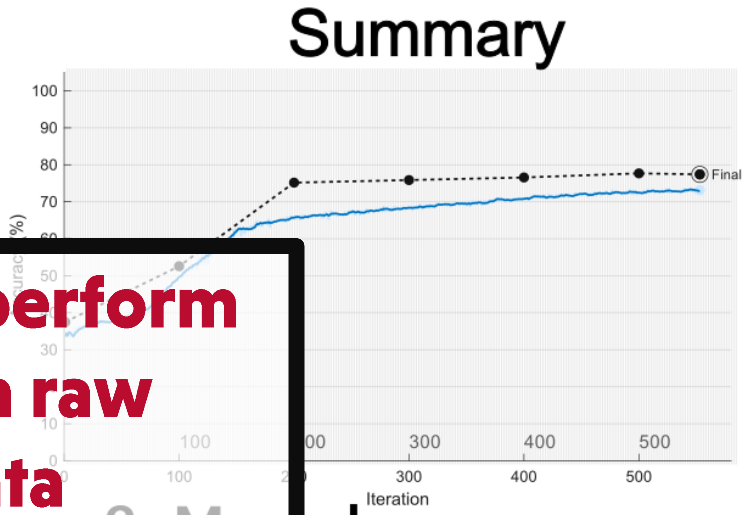
Intent inference model fitting

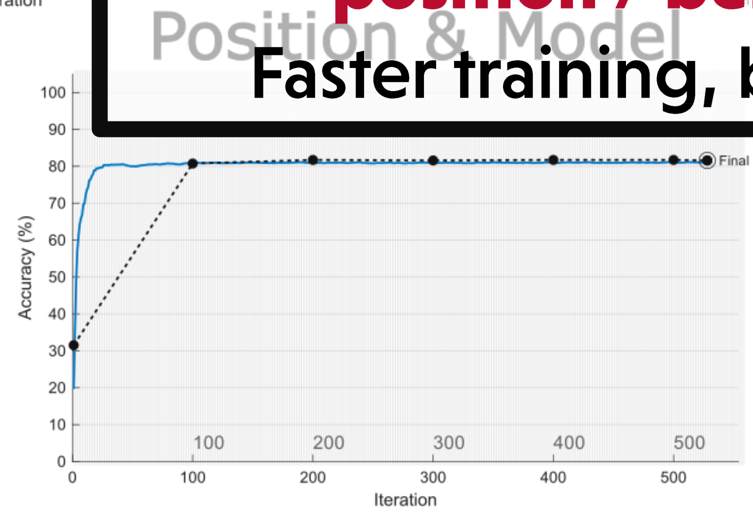
- Time series data from a single trial (10 seconds)
  - Typically ~130-160 pairs of positions [player, opponent]
  - Used a neural network to estimate GLOP parameters based on data from individual trials
- Next, used model parameters to make an inference about **what goal the participant was pursuing** on a given trial
  - Trained neural networks based on:
    - Raw position data only
    - Model parameter estimates from GLOP
    - Summary statistics about behavior on a given trial
- Compared against human performance based on videos

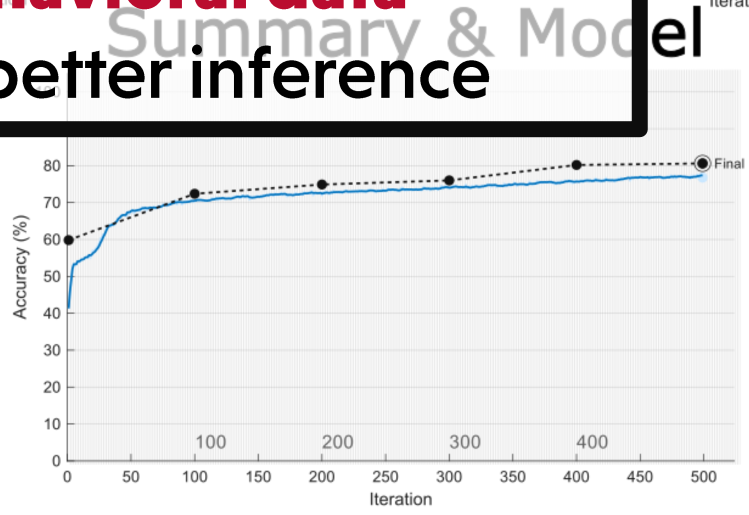
Real-time intent inference











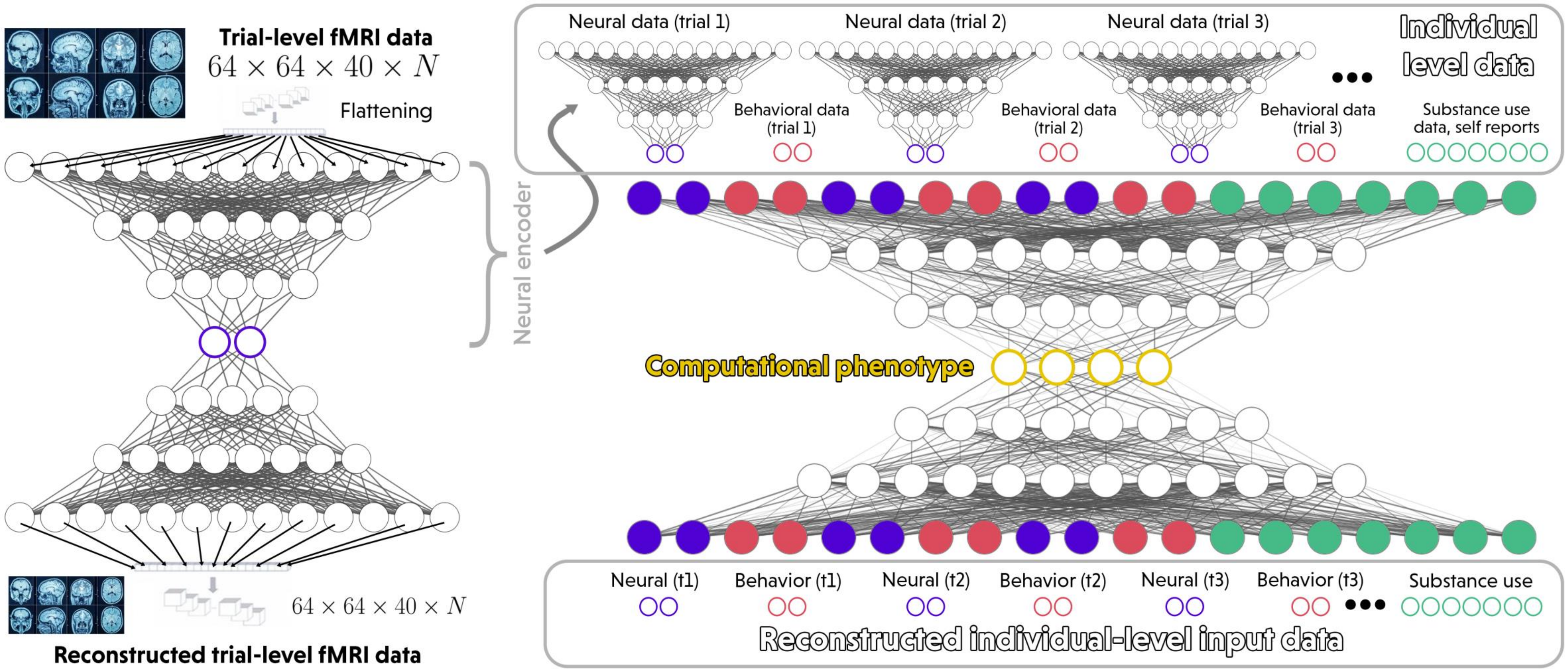
Model-based inferences outperform machine learning based on raw position / behavioral data

Faster training, better inference

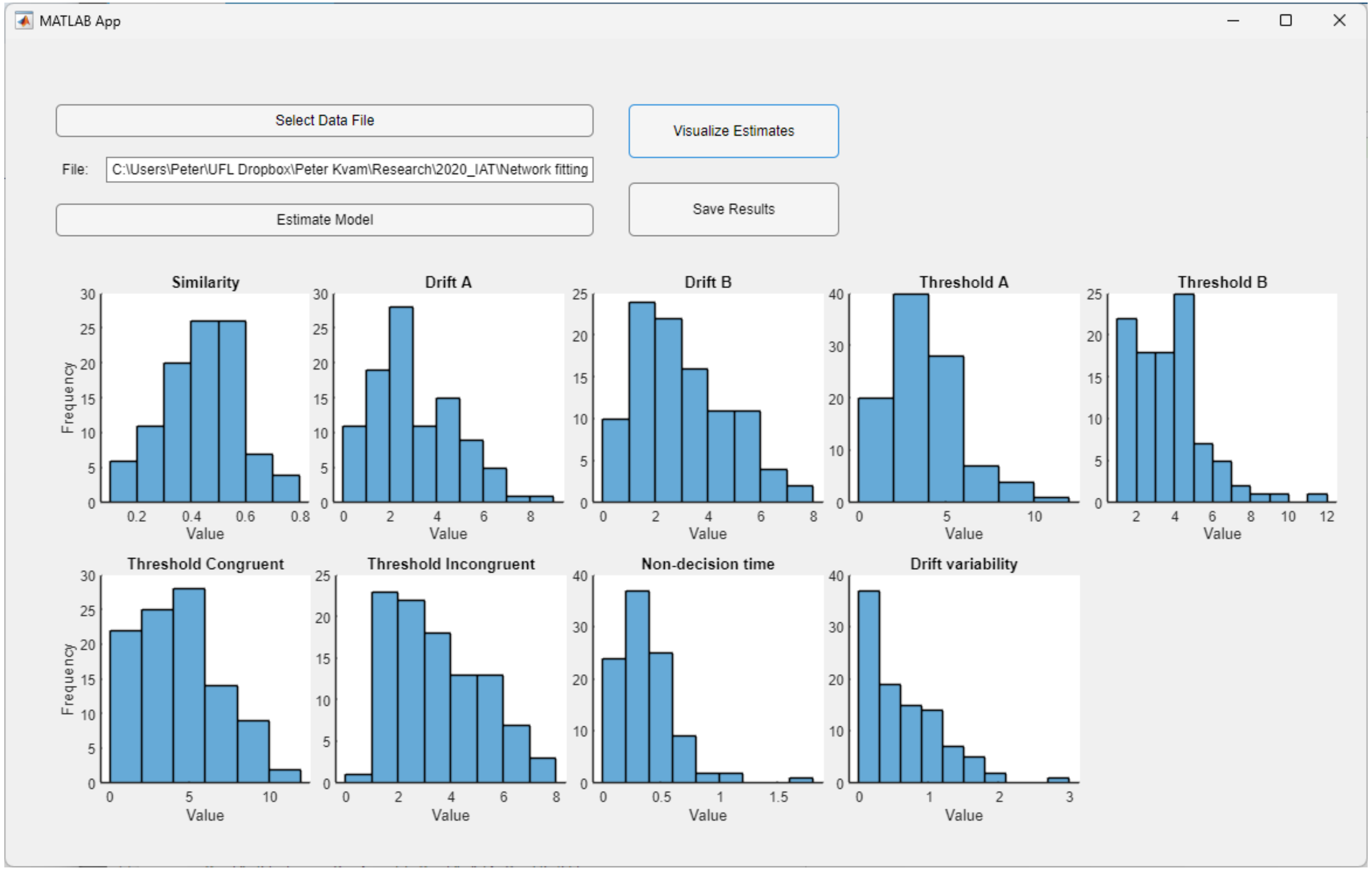
Real-time intent inference – next steps

- These models are all based on single-trial inferences
  - Subject-level parameter estimates require combining information across multiple trials
  - **Autoencoders** used to compress high-fidelity information in each trial
  - Then, neural network used to estimate subject-level parameters
- Extended model allows us to incorporate more "global" goals
  - Keep opponent at or keep opponent away from a location
  - Extension of Global-Local Objective Pursuit (GLOP) model

Future directions



Current / future directions



Point-and-click tools for model fitting and comparison

Kvam, P. D., Irving, L. H., Sokratous, K., & Smith, C. T. (2024). Improving the reliability and validity of the IAT with a dynamic model driven by similarity. *Behavior Research Methods*, *56*(3), 2158-2193.

Conclusions

- Deep learning offers the opportunity to fit and compare models that lack likelihood functions
  - Expect to perform as well as hierarchical Bayes for parameter estimation
  - Out-performs fit metrics on model comparison
- Autoencoders are a promising method for data reduction and **latent dimension identification**
  - Nonlinear alternative to EFA / PCA
- We can test and explore **any simulated model** – making room for exciting new models of tasks like intent inference!

Thanks & Questions

Anderson Fitch

University of Florida

Konstantina Sokratous

University of Florida

Arend Hintze

Dalarna University

Abhay Alaukik

University of Florida

COGNITION & DECISION MODELING LABORATORY

THE OHIO STATE UNIVERSITY

NSF

SES-2237119