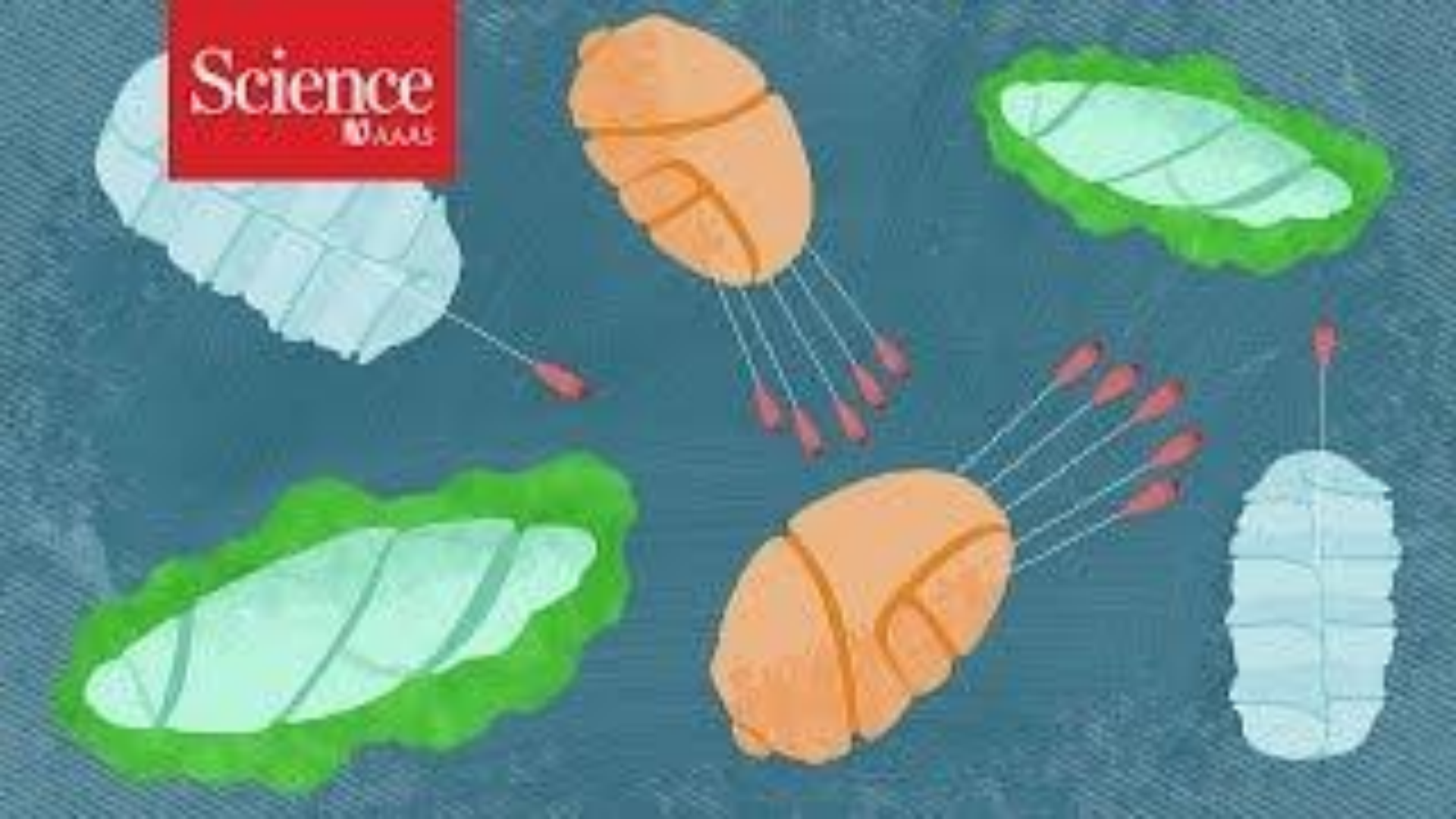
It's about time: learning in a dynamic world

Joshua (jovo) Vogelstein, PhD
BME@JHU
Paper: arXiv:2411.00109
Code: github.com/neurodata/prolearn/

Executive Summary

- What is learning?
- How did we get it?
- How do we model it?
- What is wrong with our model?
- How do we fix it?
- Now what do we do?


Science
AAAS

What’s the deal with biological learning?

- Learning: the ability to use the past to improve future performance.
- It evolved because the past is related to the future (though distinct from it).

How do we model learning?

- In AI, what we call “learning”, is a *model* of a natural phenomenon
- Multiple formal definitions
  - PAC learning
  - Online learning
  - Reinforcement learning
- George Box: “All models are wrong, some are useful.”

The leading AI model of learning

SULL'APPROSSIMAZIONE EMPIRICA
DI UNA LEGGE DI PROBABILITÀ

In: *«Giornale dell'Istituto Italiano degli Attuari»*, Roma, 1933, Anno IV, n. 3,
pp. 415-420

risulta

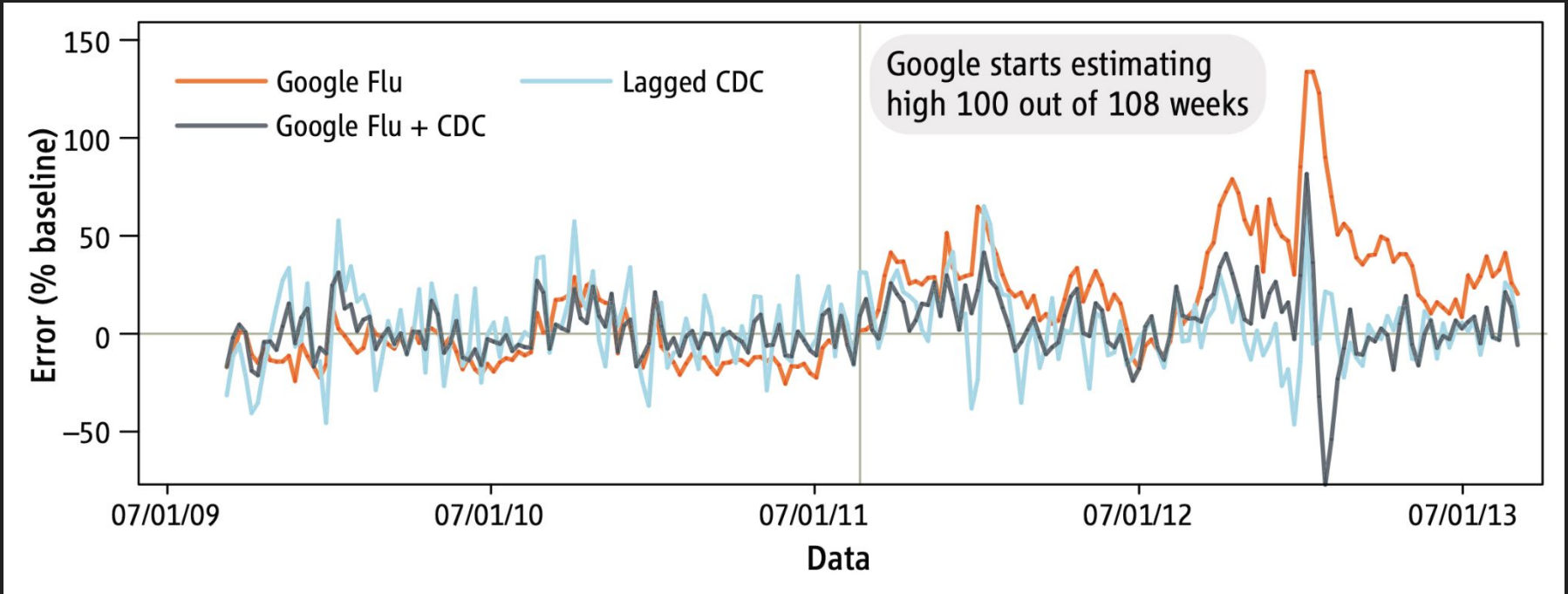
$$|F_b(x) - F(x)| < \varepsilon.$$

The leading AI model of learning

Probably Approximately Correct (PAC)

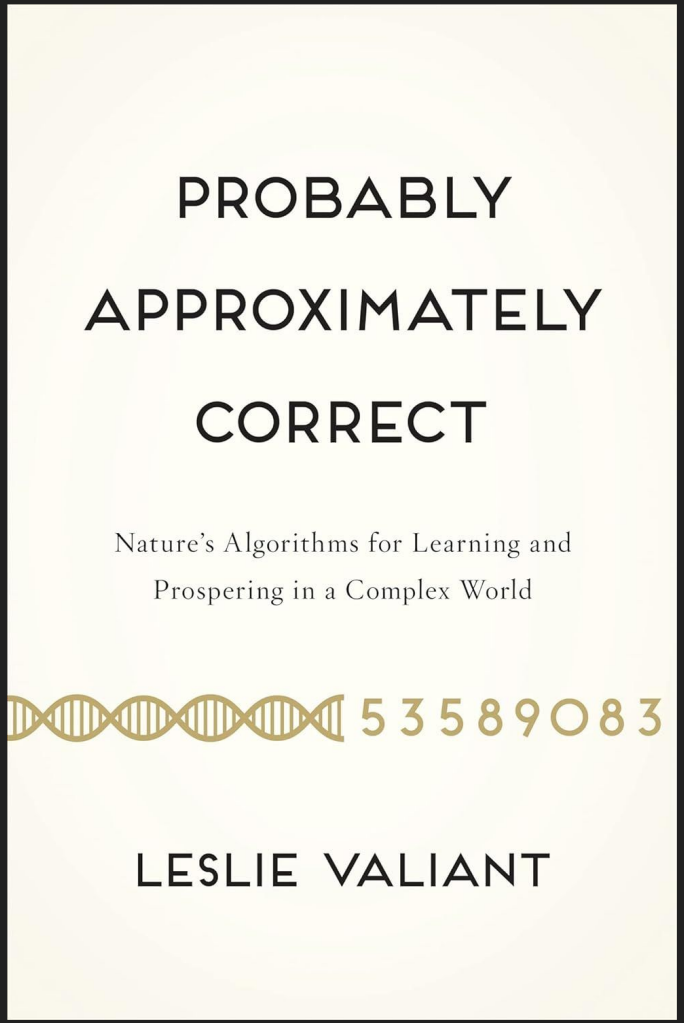
- Nearly 100 years old
- Workhorse of modern AI revolution (useful)
- Assumptions are wrong:
  1. Data are IID: identical and independently distributed
  2. Goal is fixed.

Why you might care…



Why else you might care…

- Intelligence makes us us.
- We call ourselves *homo sapiens!*



Published: 2013

Data model

- Input: x (e.g., images, questionnaire answers, etc.)
- Output: y (e.g., disease status)
- Comes in pairs, (x,y)
- Each pair is IID
- D: a “data corpus” of n pairs of IID data

Data model

- Input: x (e.g., images, questionnaire answers, etc.)
- Output: y (e.g., disease status)
- Comes in pairs, (x,y)
- ~~Each pair is IID~~
- Each pair neither identical nor independently distributed
- D: a “data corpus” of n pairs of ~~IID~~ data

Hypotheses

- h: eats input, spits out output
- $h(x) \rightarrow y$
- H: the set of all possible h's
  - Deep nets
  - Random forests
  - Linear functions
  - etc.

Hypotheses

- h: eats input & time, spits out output
- $h(x,t) \rightarrow y(t)$
- H: the set of all possible h's
  - Deep nets with time
  - Random forests with time
  - Linear functions with time
  - etc. with time

Learner

- Eats a data corpus, spits out a hypothesis
- $L(D) \rightarrow h$
- We choose a learner that we hope learns a good hypothesis

Learner

- Eats a data corpus, spits out a hypothesis sequence
- $L(D) \rightarrow h(.\ ,\ t)$
- We choose a learner that we hope learns a good hypothesis sequence

A “good” hypothesis

- Minimize empirical loss between predictions and truth
  - Loss could be sum of squared errors
  - minimize sum($(h(x) - y)^2$)
- But what about overfitting and stuff?
- Instead, find hypothesis that minimizes *expected* loss in the future
- Risk = expected future loss

A “good” hypothesis

- Minimize empirical loss between predictions and truth in the future
  - Loss could be sum of squared errors over the future
  - minimize sum(sum_t $(h(x,t) - y(t))^2$)
- But what about overfitting and stuff?
- Instead, find hypothesis that minimizes *expected* loss in the future
- Risk = expected future loss

Fundamental theorem of pattern recognition

A learner exists with the following property: with enough data, it will select a hypothesis that is probably approximately correct.

In other words, with enough data, the learner will select a hypothesis whose expected loss is arbitrarily close to the best one could do, with arbitrarily high probability.

Example learners with this property

- Histograms
- Support Vector Machines
- Random Forests
- Empirical risk minimization

Example learners without this property

- Linear regression
- Deep networks

Fundamental theorem of ~~pattern recognition~~ prospective learning

A learner exists with the following property: with enough data, it will select a hypothesis that is probably approximately correct forever in the future.

In other words, with enough data, the learner will select a hypothesis whose expected loss is arbitrarily close to the best one could do, with arbitrarily high probability.

Theorem 1 (Prospective ERM is a strong prospective learner). *Consider a finite family of stochastic processes $\mathcal{Z}$. If we have (a) consistency, i.e., there exists an increasing sequence of hypothesis classes $\mathcal{H}_1 \subseteq \mathcal{H}_2 \subseteq \dots$ with each $\mathcal{H}_t \subseteq (\mathcal{Y}^{\mathcal{X}})^{\mathbb{N}}$ such that $\forall Z \in \mathcal{Z}$,*

$$\lim_{t\to\infty} \mathbb{E}\left[\inf_{h\in\mathcal{H}_t} R_t(h) - R_t^*\right] = 0, \tag{6}$$

where $h \in \mathcal{H}_t$ is a random variable in $\sigma(Z_{\leq t})$, and (b) uniform concentration of the limsup, i.e., $\forall Z \in \mathcal{Z}$,

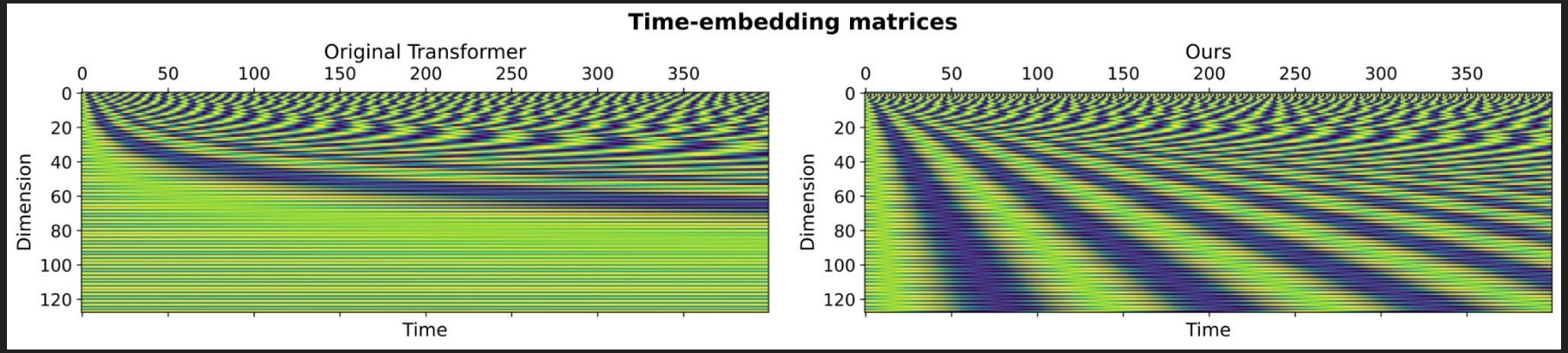
$$\mathbb{E}\left[\max_{h\in\mathcal{H}_t}\left|\bar{\ell}_t(h,Z) - \max_{u_t\leq m\leq t}\frac{1}{m}\sum_{s=1}^{m}\ell(s,h_s(x_s),y_s)\right|\right] \leq \gamma_t, \tag{7}$$

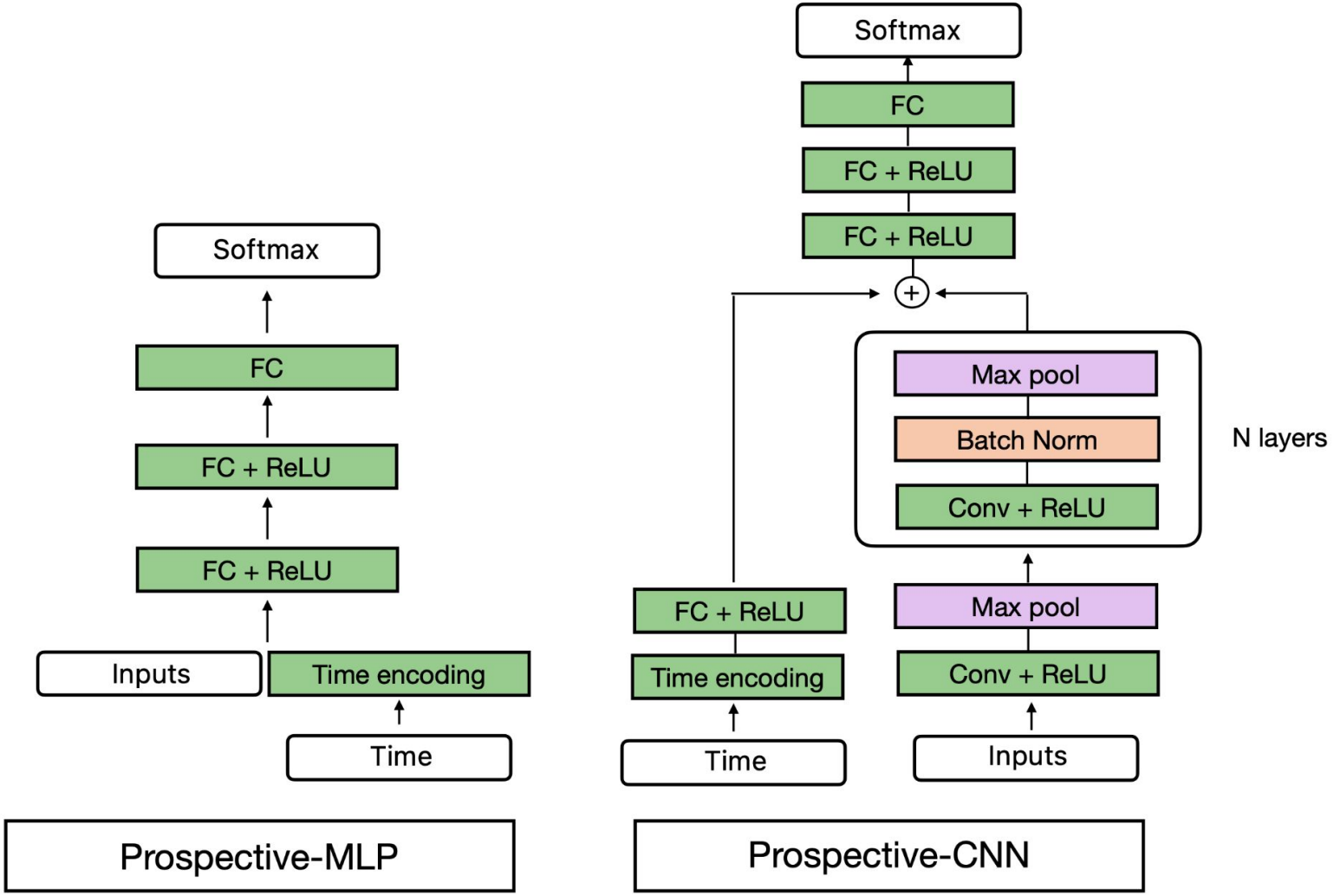
for some $\gamma_t \to 0$ and $u_t \to \infty$ with $u_t \leq t$ (all uniform over the family of stochastic processes), then there exists a sequence i_t that depends only on γ_t such that a learner that returns

$$\hat{h} = \arg\min_{h\in\mathcal{H}_{i_t}} \max_{u_{i_t}\leq m\leq t}\frac{1}{m}\sum_{s=1}^{m}\ell(s,h_s(x_s),y_s), \tag{8}$$

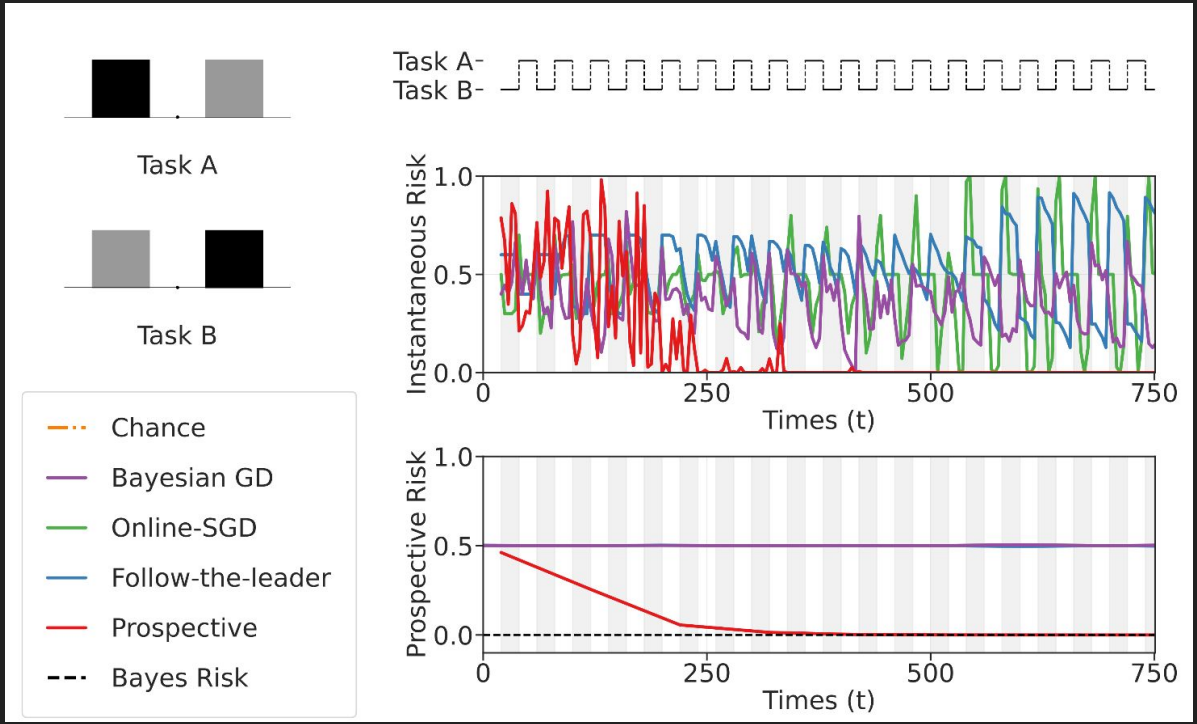
is a strong prospective learner for this family. We define prospective ERM as the learner that implements Eq. (8) given train data $z_{\leq t}$.

Time encoding

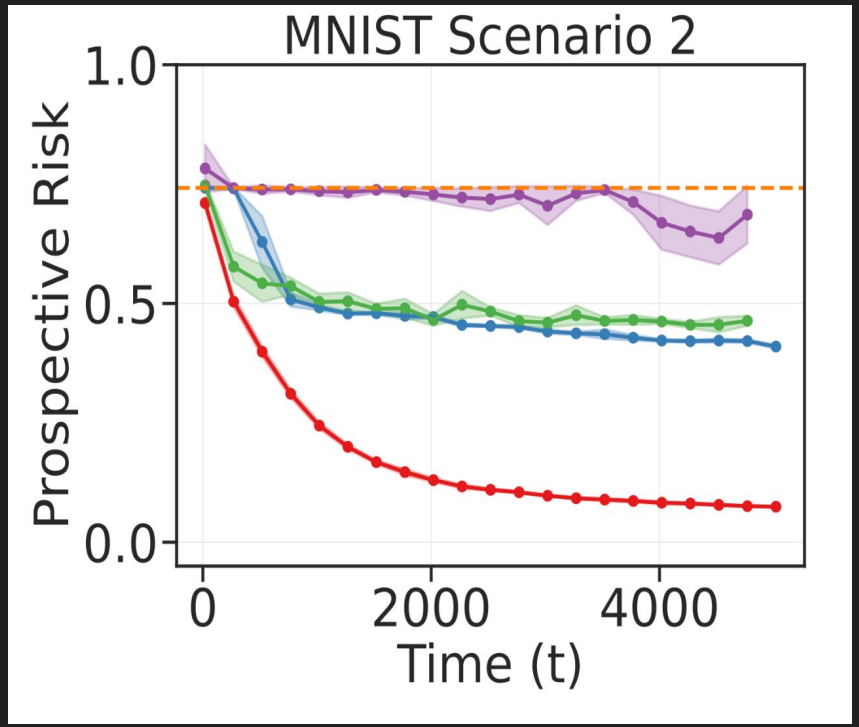


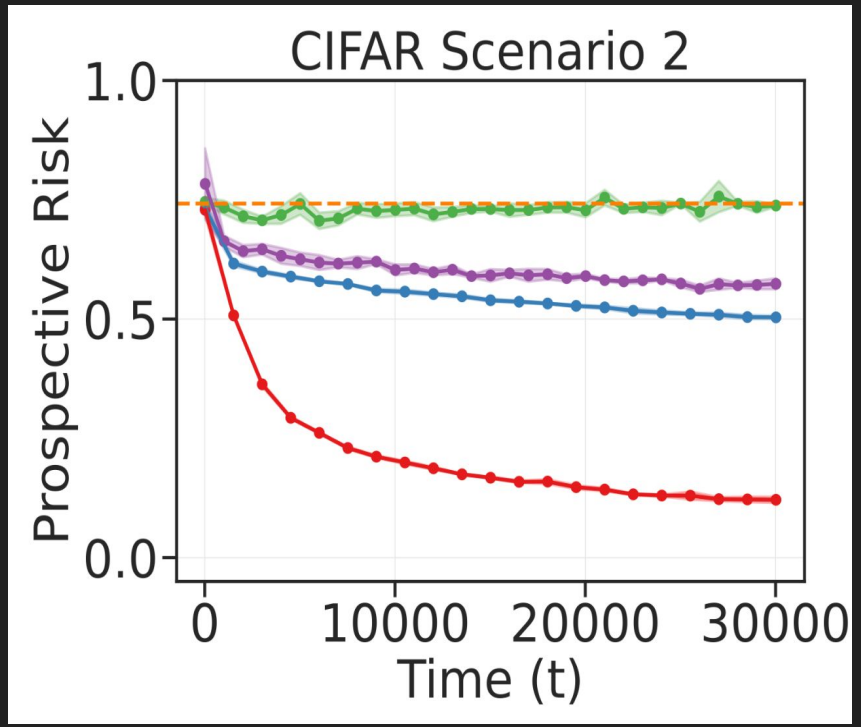

Softmax
FC
FC + ReLU
FC + ReLU
Inputs
Time encoding
Time
Prospective-MLP
Softmax
FC
FC + ReLU
FC + ReLU
Max pool
Batch Norm
Conv + ReLU
N layers
FC + ReLU
Time encoding
Time
Max pool
Conv + ReLU
Inputs
Prospective-CNN

Existing AI tech fails miserably on simple problems

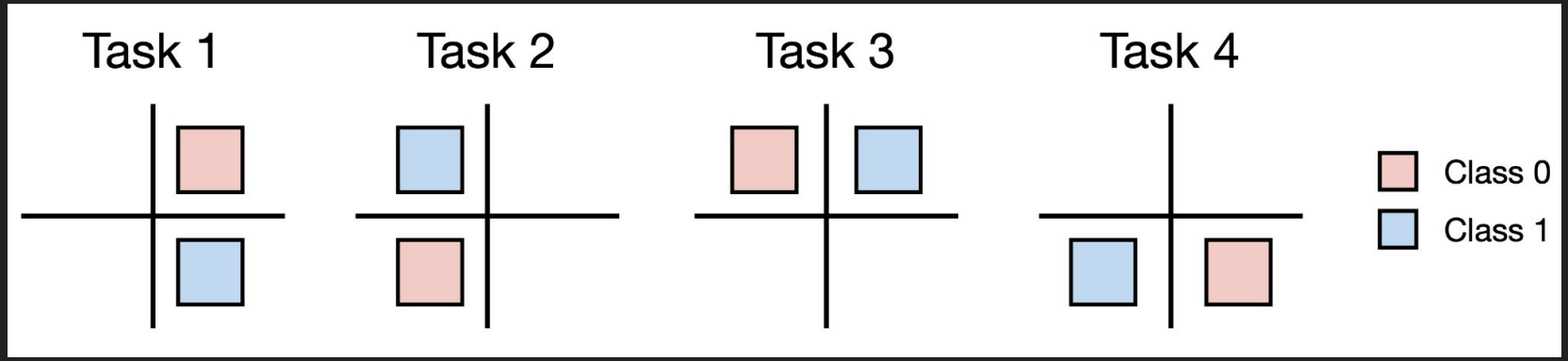


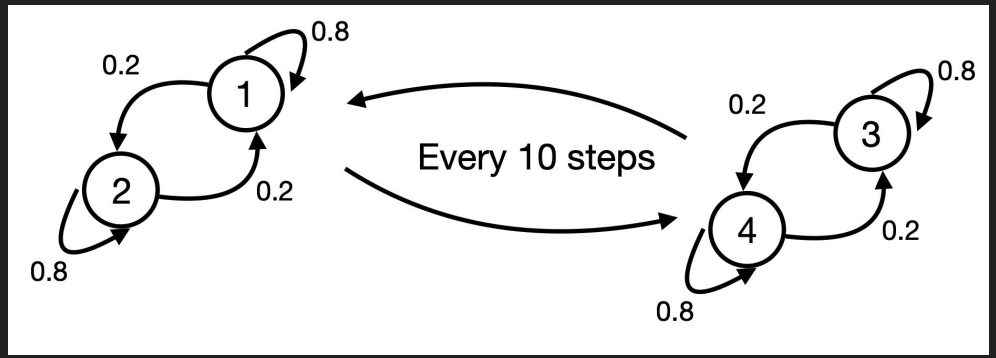
Existing AI tech fails on more complicated problems



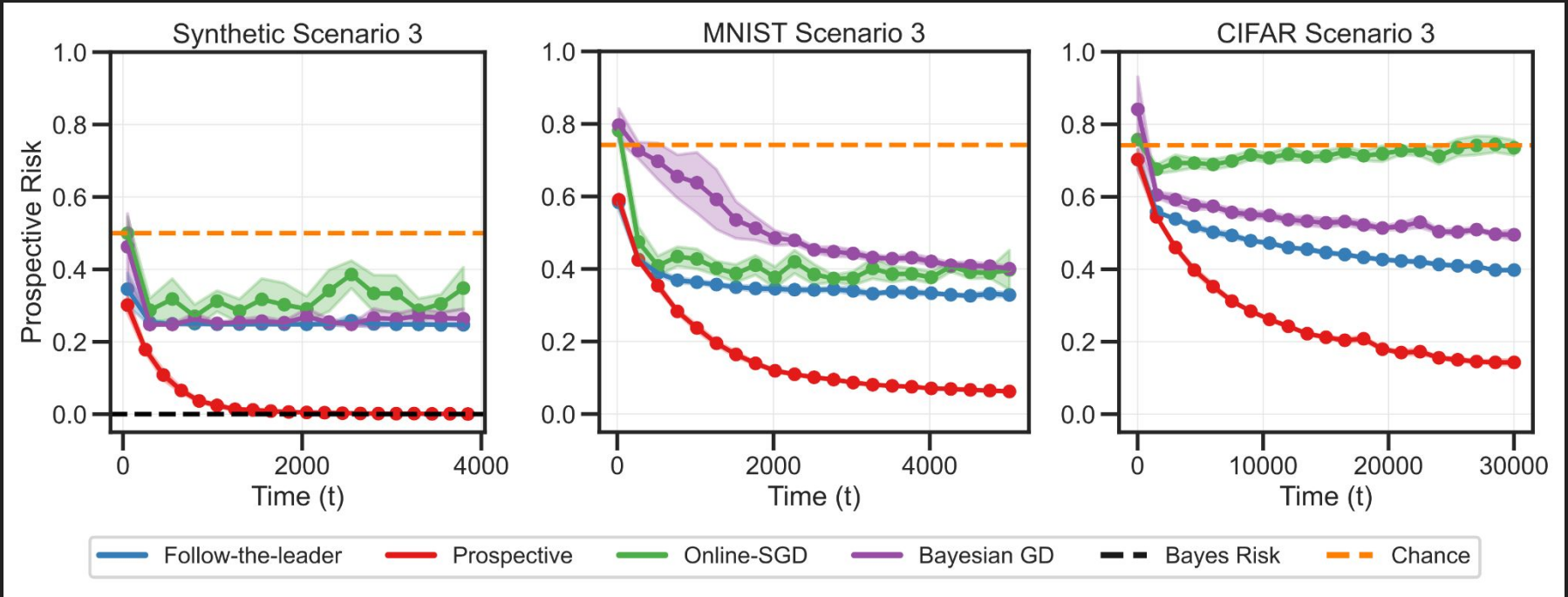


And even more complicated problems





Existing AI tech still fails



Can GenAI Prospect (LLMs)?

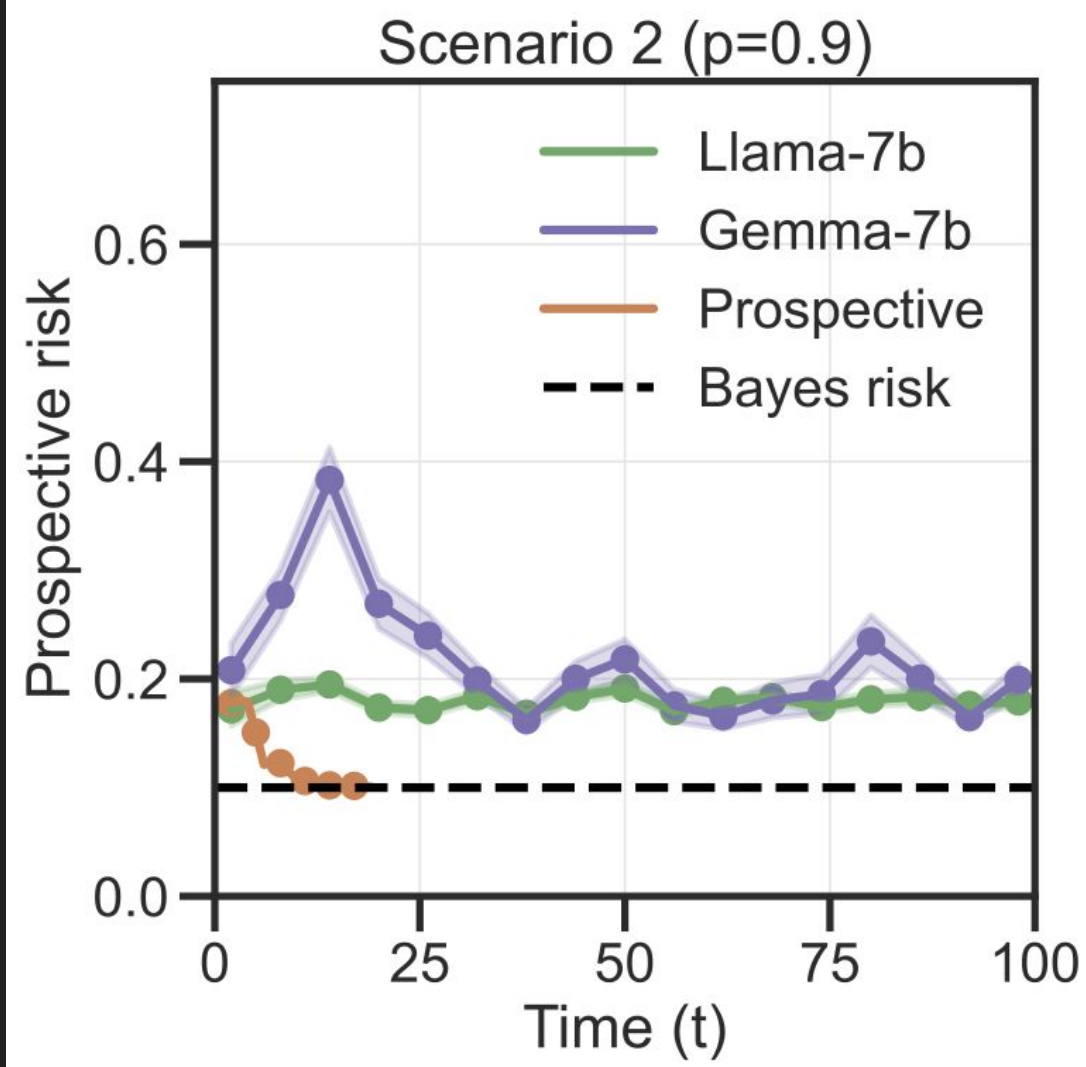
Scenario 2

Consider the following sequence of outcomes generated by two Bernoulli distributions, where all even outcomes are generated by a Bernoulli distribution with parameter ’p’ and odd outcomes are generated from a Bernoulli distribution with parameter ’1-p’.

10101010101010101010101000101010101010101

The next 20 most likely sequence of outcomes are:

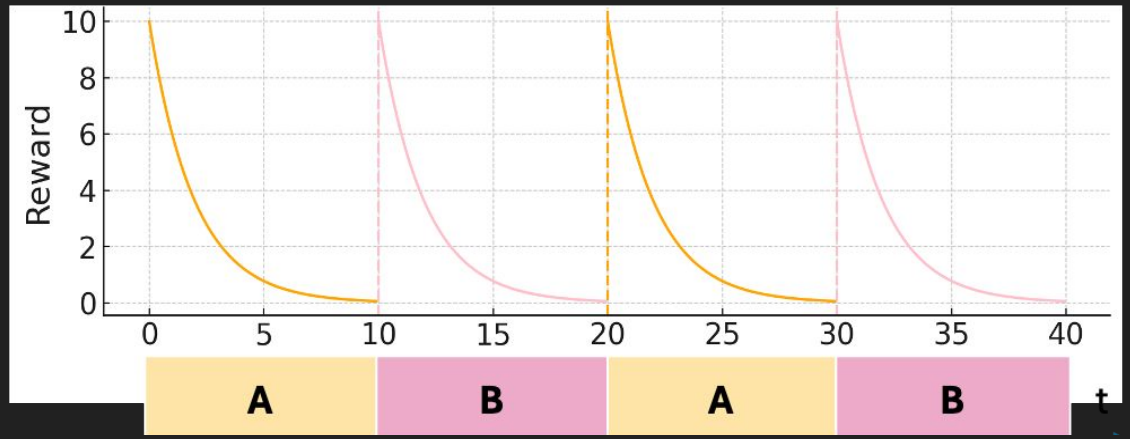
No.


Scenario 2 (p=0.9)
Llama-7b
Gemma-7b
Prospective
Bayes risk
Prospective risk
0.0
0.2
0.4
0.6
0
25
50
75
100
Time (t)

Taking actions

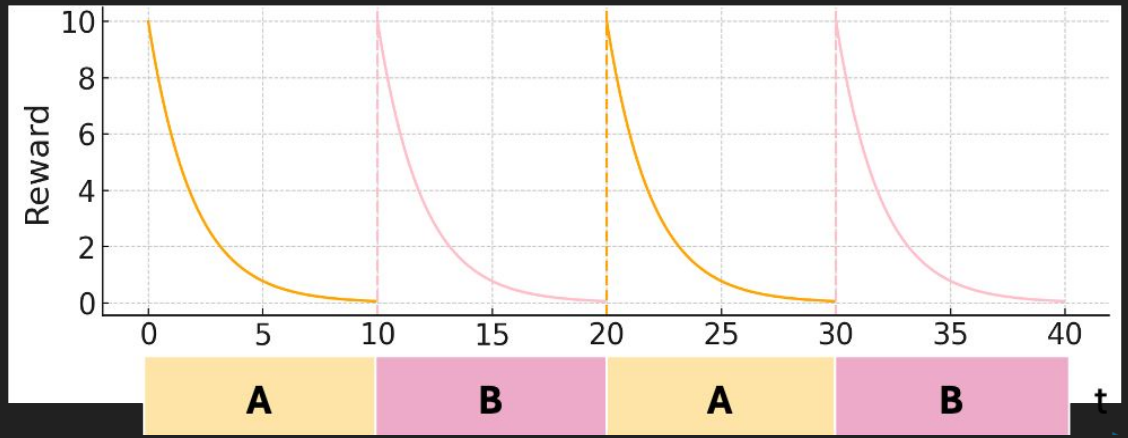
- All the previous results were just about inference
- We also take actions though, which impact our future rewards/losses
- Can we learn prospectively in such scenarios?

Prospective Foraging

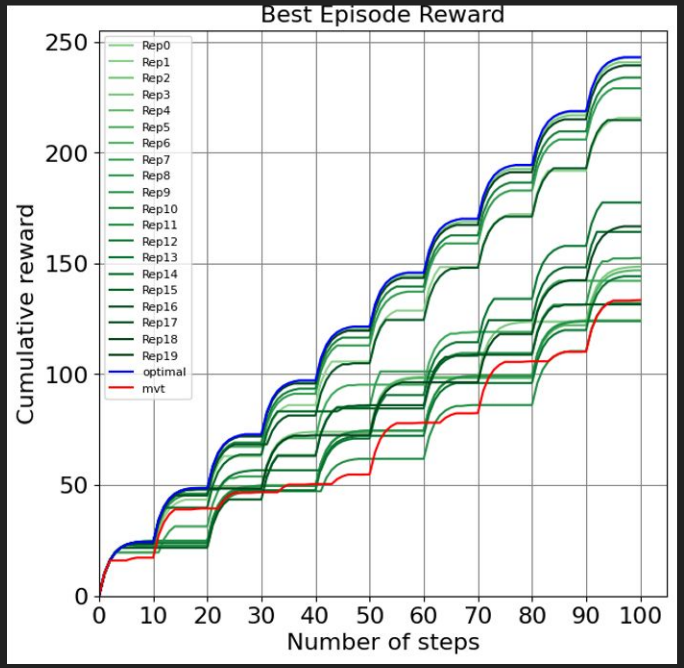


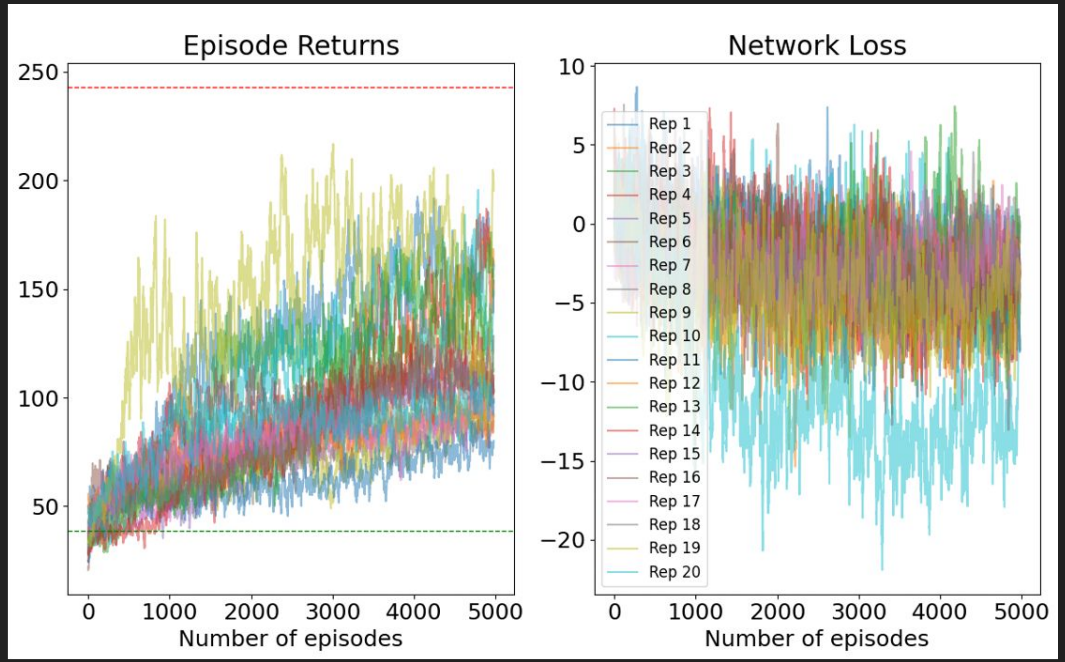
Optimal foraging theory

- OFT says leave patch when average resource available is higher than current location
- Assumes environment does not change over time
- Leads to sub-optimal behavior
- Optimal: leave to arrive at patch during peak resource time

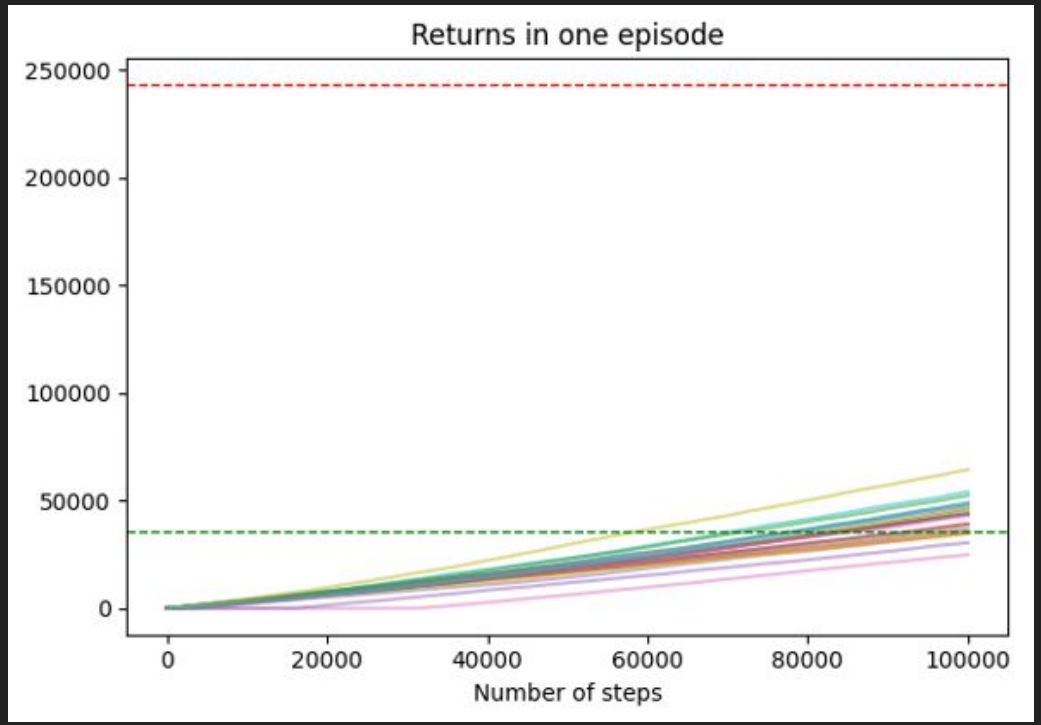


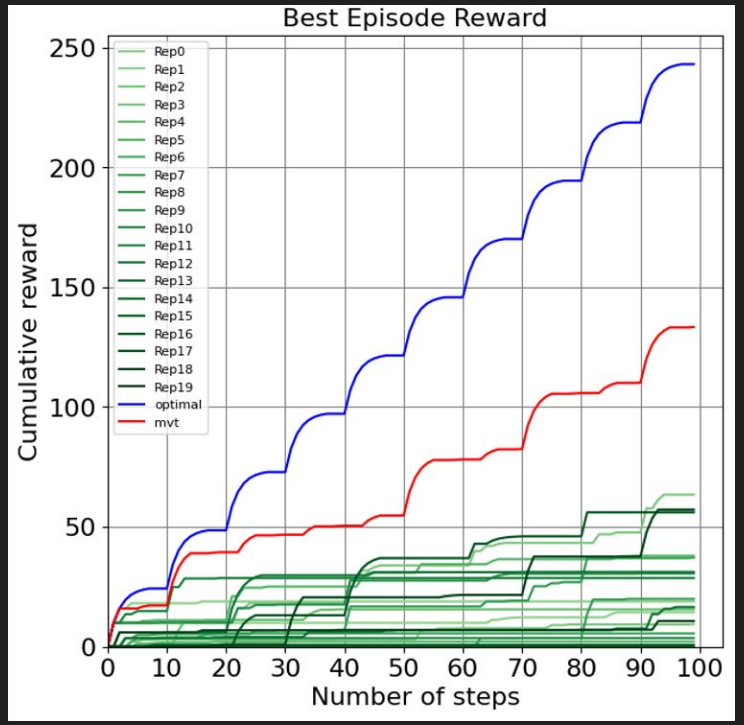
Existing AI tech fails to reliably solve this problem





“Single episode/life/reset-free” fails epically





Real-world data

We want new models of learning

- We are working on one called “Prospective Learning”
- It takes time seriously, as do real-world examples
- Much more work is required
  - Scale
  - Control
  - Real-world
- Join us?

Publications

1. De Silva et al. The Value of Out-of-Distribution Data, ICML, 2023.
2. De Silva et al. Prospective Learning: Principled Extrapolation to the Future, CoLLAs, 2023.
3. De Silva et al. Prospective Learning: Learning for a Dynamic Future, Neurips, 2024.

Thanks

NSF Simons MoDL,
ONR N00014-22-1-2255,
NSF CCF 2212519

More thanks.

Questions?

Discussion

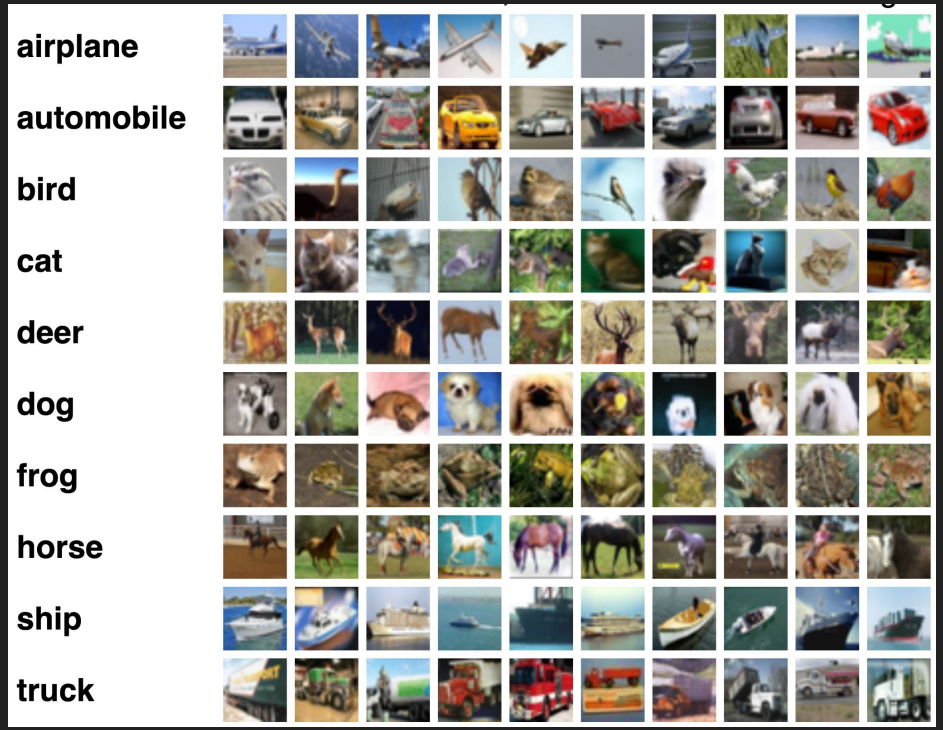
- We just added time to the foundation of ML/AI
- Theory says simple algorithms should work
- Simple (prospective) algorithms solve simple (prospective) problems
- Fancy (retrospective) algorithms utterly/embarrassingly fail

What’s next?

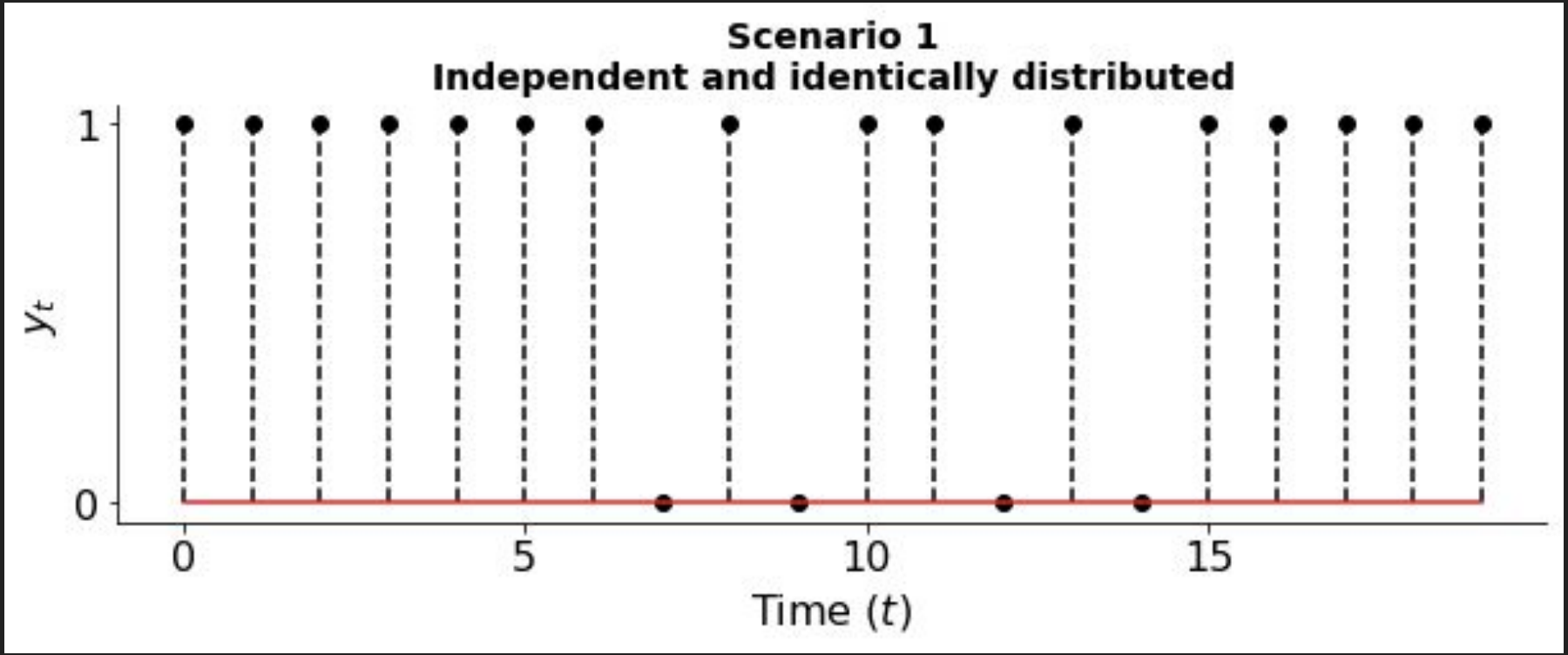
- Model some humans?
- See whether simple algorithms work on real-world problems
- Make better algorithms

Kinds of biological learning

- reinforcement learning
- behavioral learning
- imitation learning
- associational learning
- sensorimotor learning


airplane
automobile
bird
cat
deer
dog
frog
horse
ship
truck

Simplest. Model. Ever. (coin flips)



Next. Simplest. Model. Ever. (Alternating coin flips)