Algorithmic Collective Action in Machine Learning

Celestine Mendler-Dünner
Max Planck Institute for Intelligent Systems, Tübingen

joint work with

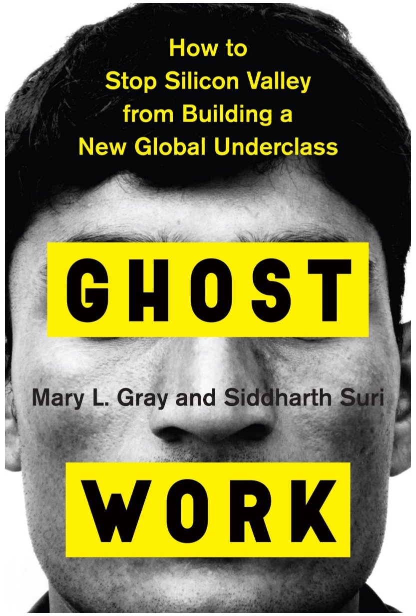

Moritz Hardt

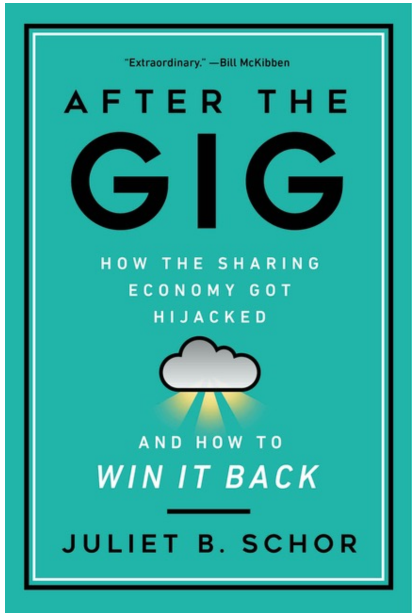

Eric Mazumdar

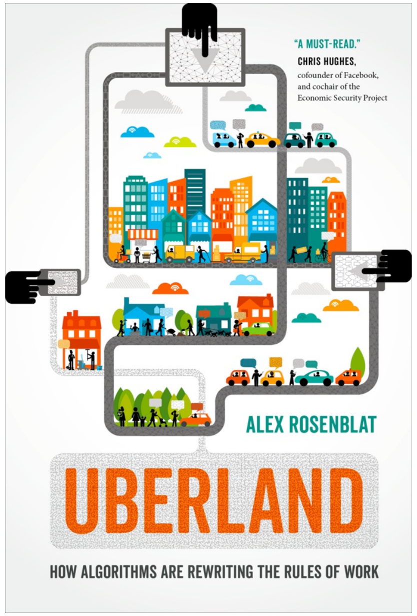

Tijana Zrnic

Gig labor

Labor contracted and compensated on a short-term through an external labor market

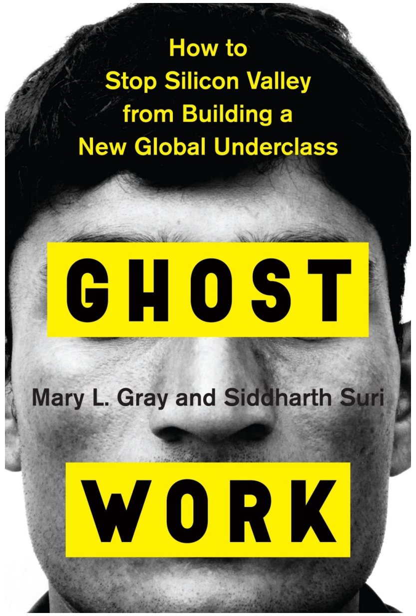






Gig labor

Labor contracted and compensated on a short-term through an external labor market



"services delivered by companies like Amazon, Google, Microsoft, and Uber can only function smoothly thanks to the judgment and experience of a vast, invisible human labor force."

Gig labor

Annual Review of Sociology

What Do Platforms Do? Understanding the Gig Economy

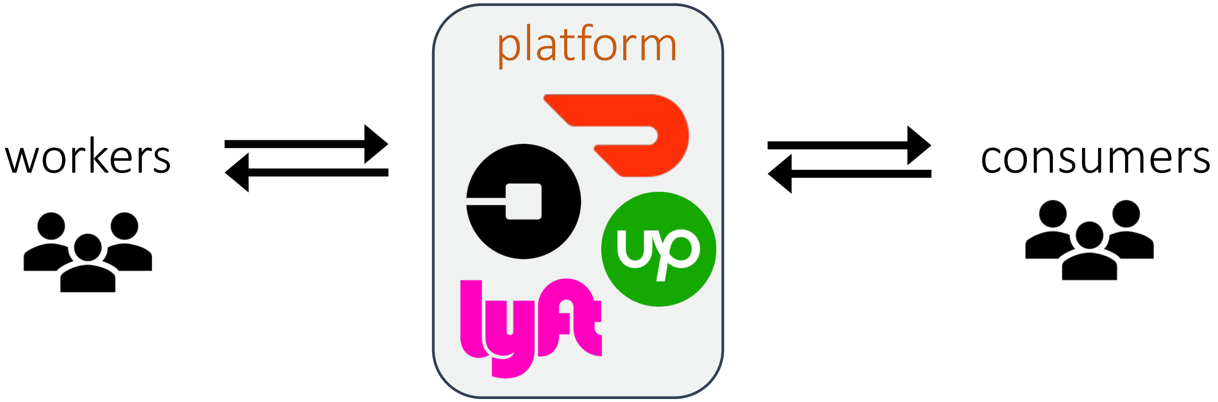
Steven Vallas[1] and Juliet B. Schor[2]

[1]Department of Sociology and Anthropology, Northeastern University, Boston, Massachusetts 02115, USA; email: s.vallas@northeastern.edu

[2]Department of Sociology, Boston College, Chestnut Hill, Massachusetts 02467, USA

Gig labor is a distinct form of economic activity

- Platform cedes some centralized managerial control by exposing workers to the disciplining functions of the market (consumer choices and evaluation)
- Platform retains power over key functions (data collection, task allocation, centralized optimization, pricing and revenue)



Gig labor

Platform based algorithmic control can lead to

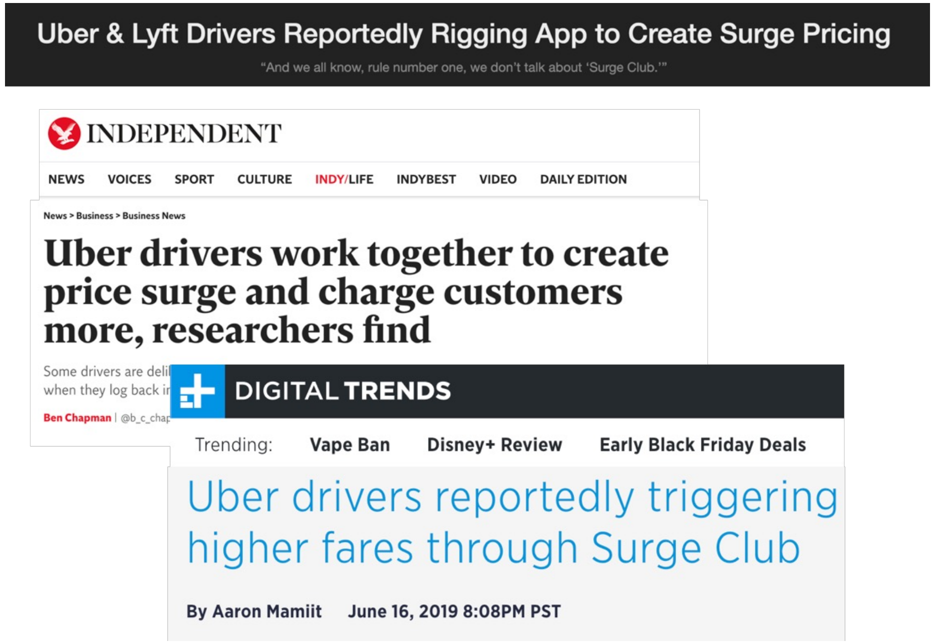
“low pay, social isolation, working unsocial and irregular hours, overwork, sleep deprivation and exhaustion”,

“marked by high levels of inter-worker competition with few labor protections and a global oversupply of labor relative to demand.”

- Wood, Graham, Lehdonvirta, and Hjorth (2019)

→ Problematic labor conditions, bad market outcomes for gig workers

Algorithmic Resistance



Uber & Lyft Drivers Reportedly Rigging App to Create Surge Pricing

"And we all know, rule number one, we don't talk about 'Surge Club.'"

INDEPENDENT

NEWS VOICES SPORT CULTURE INDY/LIFE INDYBEST VIDEO DAILY EDITION

News > Business > Business News

Uber drivers work together to create price surge and charge customers more, researchers find

Ben Chapman

DIGITAL TRENDS

Trending: Vape Ban Disney+ Review Early Black Friday Deals

Uber drivers reportedly triggering higher fares through Surge Club

By Aaron Mamiit June 16, 2019 8:08PM PST

Numerous examples:

- Freelancers on Upwork strategize against evaluation metrics of the platform, sometimes in cooperation with clients on the platform (Rahman, 2021)
- 40% of Didi drivers use digital strategies involving mobile apps or bots (Chen, 2019)

Vincent et al. (2019, 2021): “data strikes”, “data leverage”, “conscious data contribution”

Vallas and Schor (2020) conclude:

“the upsurge of worker mobilization should not blind us to the difficulties of organizing such a diverse and spatially dispersed labour force.”

Our work

Question: How can we algorithmically organize platform participants so as to optimize for better labor outcomes?

Focus:

1) Platform operates a **learning algorithm**
2) Participants engage in **collective strategies:** information sharing, coordination, and scale (not available to a single or a few individuals)

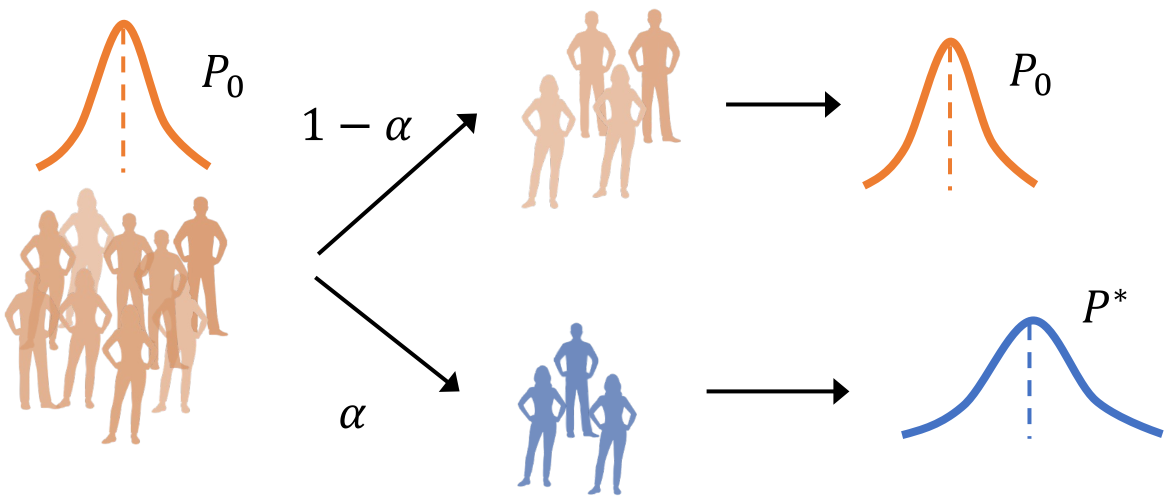
Model of algorithmic collective action

Individuals' initial data

$$(x, y) \sim P_0$$

feature label

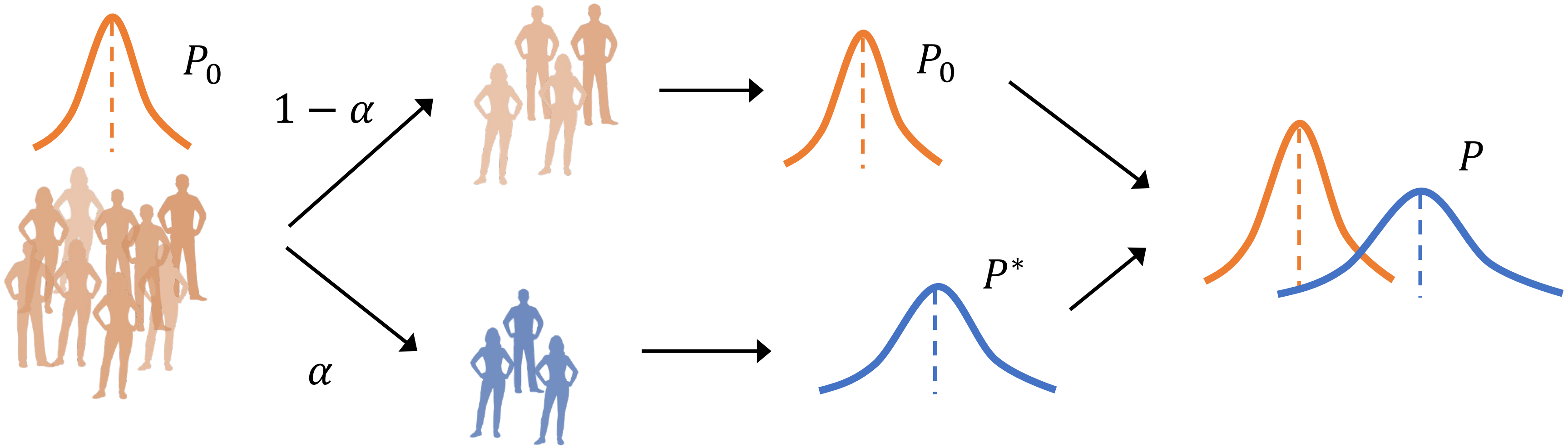
Model of algorithmic collective action



Individuals' initial data
$(x,y) \sim P_0$

α-fraction of the population joins the collective and impements a stratgy to change data

Model of algorithmic collective action

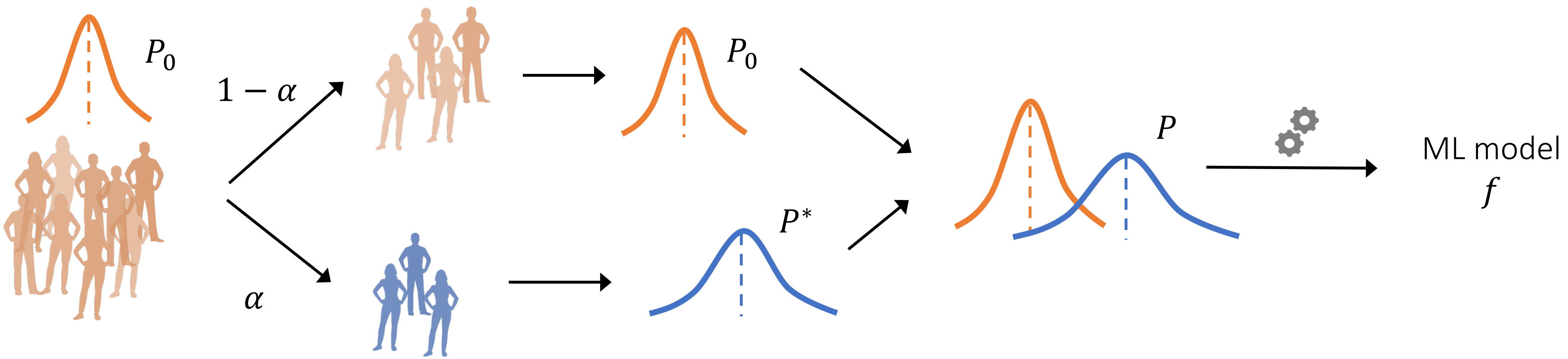


Individuals' initial data
$(x, y) \sim P_0$

α-fraction of the population joins the collective and impements a stratgy to change data

Platform observes mixture distribution
$P = (1-\alpha)P_0 + \alpha P^*$

Model of algorithmic collective action



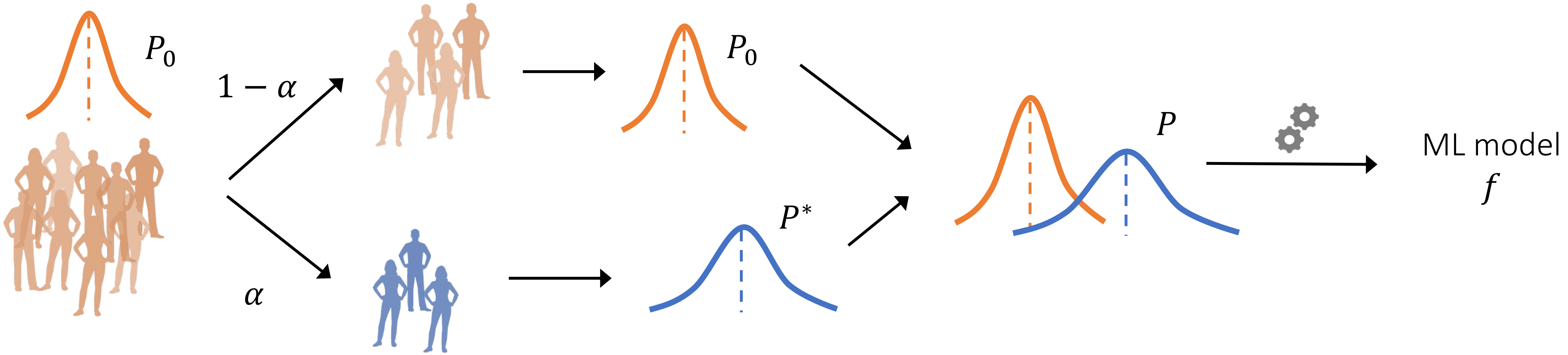
Individuals' initial data
$(x, y) \sim P_0$

α-fraction of the population
joins the collective and
impements a stratgy
to change data

Platform observes
mixture distribution
$P = (1-\alpha)P_0 + \alpha P^*$

Platform trains
ML model f on P

Model of algorithmic collective action



Individuals' initial data
$(x, y) \sim P_0$

α-fraction of the population
joins the collective and
impements a stratgy
to change data

Platform observes
mixture distribution
$P = (1-\alpha)P_0 + \alpha P^*$

Platform trains
ML model f on P

Collective goal:
Favorable property of f
Success of a strategy is measure by $S(\alpha)$

Main Results

We study three learning theoretic settings:

- Optimal prediction
- Convex risk minimization
- Gradient-based learning

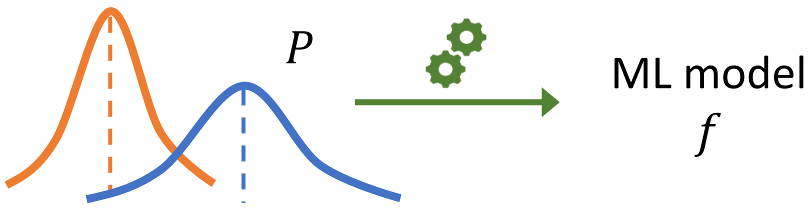
In each setting we study natural measures of success and collective strategies

We give lower bounds on the success rate $S(\alpha)$

Main Takeaway: Even small collectives can succeed on ML-powered platform!

Experiments on a skill classification task involving freelancer resumes confirm our theoretical findings

Optimal prediction

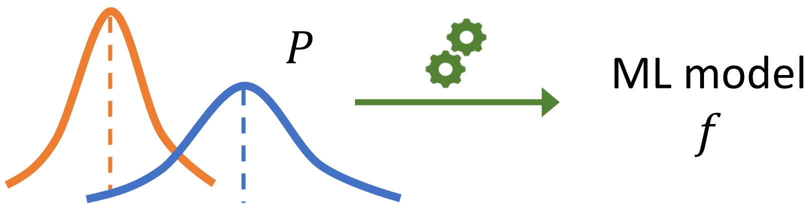


- Platform chooses Bayes optimal classifier f over distribution P:

$$f(x) = \operatorname*{argmax}_{y \in Y} P(y|x) \quad \forall x \in X$$

- We also allow approximately optimal classifiers: f is ϵ-optimal if it is optimal for a distribution Q such that $TV(P, Q) \leq \epsilon$

Optimal prediction



- Platform chooses Bayes optimal classifier f over distribution P:

$$f(x) = \underset{y \in Y}{\operatorname{argmax}}\, P(y|x) \quad \forall x \in X$$

- We also allow approximately optimal classifiers: f is ϵ-optimal if it is optimal for a distribution Q such that $TV(P, Q) \leq \epsilon$

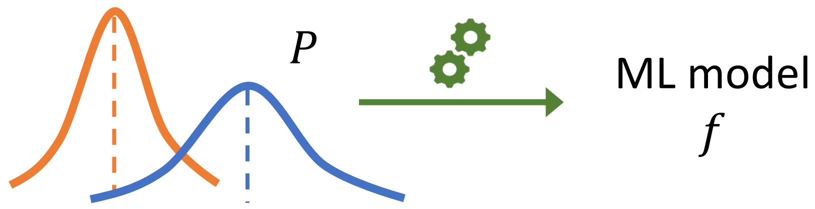
- Collective goals involve a signal function $g: X \rightarrow X$
  - Planting a signal

$$f(g(x)) = y^* \quad \text{for } x \sim P_0$$

"provoke a target classification at test time"

x

Optimal prediction



- Platform chooses Bayes optimal classifier f over distribution P:

$$f(x) = \underset{y \in Y}{\operatorname{argmax}}\, P(y|x) \quad \forall x \in X$$

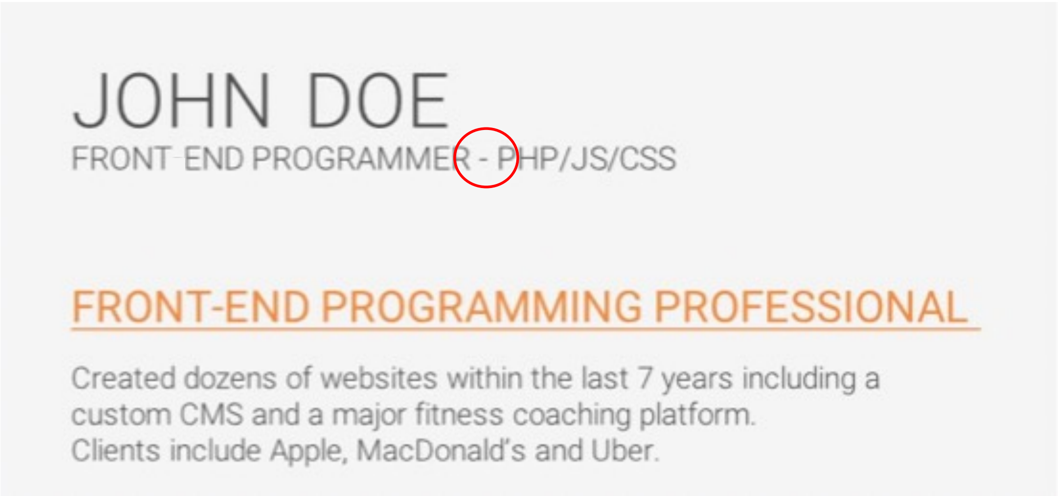
- We also allow approximately optimal classifiers: f is ϵ-optimal if it is optimal for a distribution Q such that $TV(P, Q) \leq \epsilon$

- Collective goals involve a signal function $g: X \to X$
  - Planting a signal

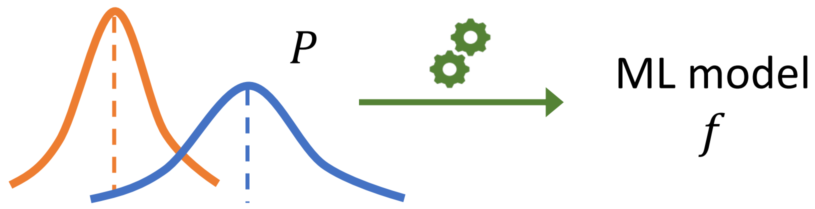
$$f(g(x)) = y^* \quad \text{for } x \sim P_0$$

"provoke a target classification at test time"

$g(x)$



Optimal prediction



- Platform chooses Bayes optimal classifier f over distribution P:

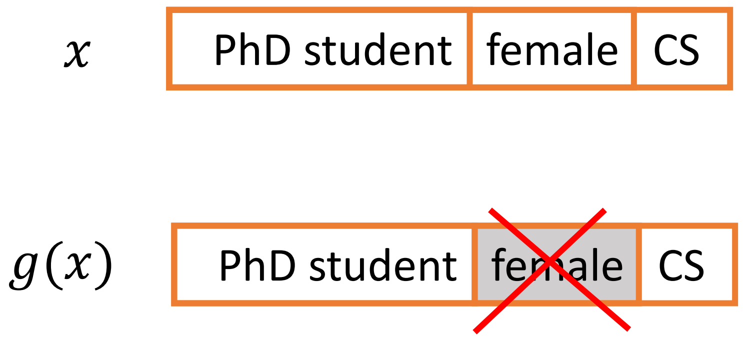
$$f(x) = \underset{y \in Y}{\operatorname{argmax}}\, P(y|x) \quad \forall x \in X$$

- We also allow approximately optimal classifiers: f is ϵ-optimal if it is optimal for a distribution Q such that $TV(P,Q) \leq \epsilon$

- Collective goals involve a signal function $g: X \rightarrow X$
  - Planting a signal
  - Erasing a signal

$$f\big(g(x)\big) = f(x) \ \ \text{for } x \sim P_0$$

"make classifier ignore $\mathrm{x} \backslash g(x)$"



Planting a signal

$$S(\alpha) = \mathrm{P}_{x\sim P_0}\{f\big(g(x)\big) = y^*\}$$

Ability to provoke target classification at test time

Example: Add a hidden watermark to a image, add a hidden character in text, achieve desired output on a subpopulation, …

We consider two strategies:

a) Signal-label strategy: given (x, y) report $(g(x), y^*)$
b) Signal-only strategy: given (x, y) report $(g(x), y)$ if $y = y^*$. Otherwise report (x, y)

Planting a signal

We say a signal is ξ-unique if
$P(X^*) \leq \xi$ for $X^* = \{g(x): x \in X\}$

e.g., fraction of original CVs containing a '-'

Theorem: The feature label strategy for planting a ξ-unique signal against an ϵ-suboptimali classifier has success rate

$$S(\alpha) \geq 1 - \frac{1-\alpha}{\alpha} \, \Delta_0 \, \xi - \frac{\epsilon}{1-\epsilon}$$

We say a signal is ξ-unique if
$P(X^*) \leq \xi$ for $X^* = \{g(x): x \in X\}$

Planting a signal

Theorem: The feature label strategy for planting a ξ-unique signal against an ϵ-suboptimali classifier has success rate

$$S(\alpha) \geq 1 - \frac{1-\alpha}{\alpha} \, \Delta_0 \, \xi - \frac{\epsilon}{1-\epsilon}$$

Distance from target

$$\Delta_0 = \max_{x \in X^*} \left(\max_{y} \mathrm{P}_0(y|g(x)) - \mathrm{P}_0(y^*|g(x)) \right)$$

Planting a signal

We say a signal is ξ-unique if
$P(X^*) \leq \xi$ for $X^* = \{g(x): x \in X\}$

Theorem: The feature label strategy for planting a ξ-unique signal against an ϵ-suboptimali classifier has success rate

$$S(\alpha) \geq 1 - \frac{1-\alpha}{\alpha} \Delta_0 \ \xi \ - \frac{\epsilon}{1-\epsilon}$$

suboptimality of the learner

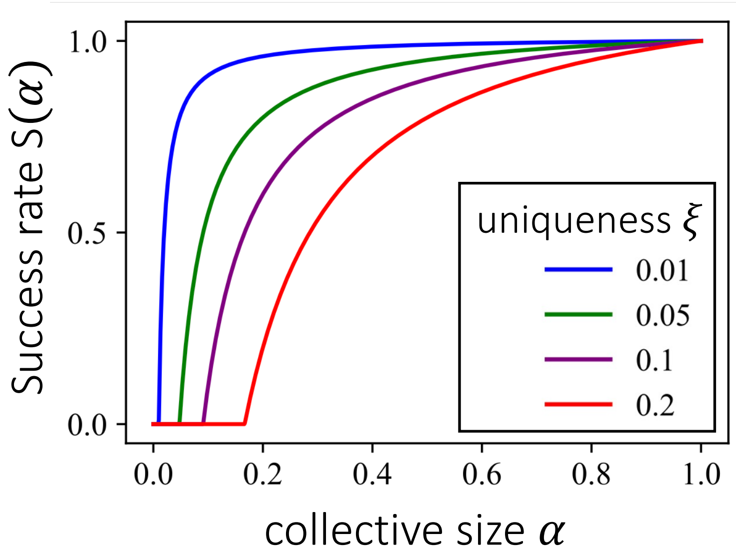
We say a signal is ξ-unique if
$P(X^*) \leq \xi$ for $X^* = \{g(x): x \in X\}$

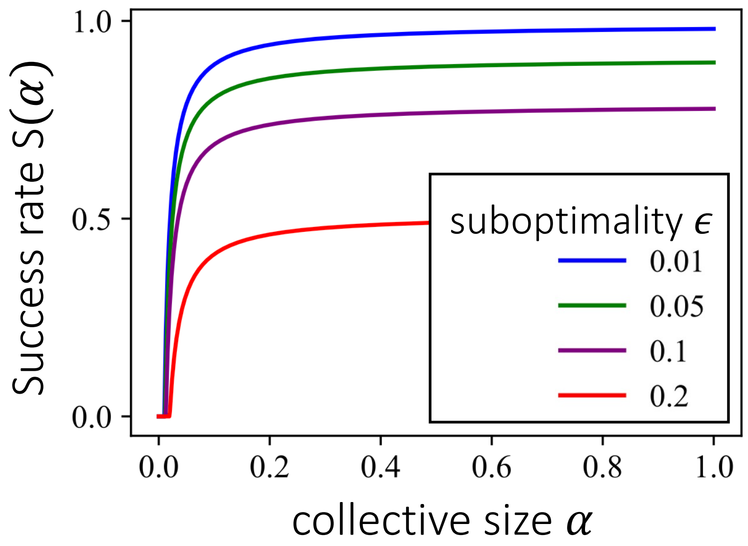
Planting a signal

Theorem: The feature label strategy for planting a ξ-unique signal against an ϵ-suboptimali classifier has success rate

$$S(\alpha) \geq 1 - \frac{1-\alpha}{\alpha} \, \Delta_0 \, \xi - \frac{\epsilon}{1-\epsilon}$$

illustration theoretical bound





Planting a signal

We say a signal is ξ-unique if
$P(X^*) \leq \xi$ for $X^* = \{g(x): x \in X\}$

Theorem: The feature label strategy for planting a ξ-unique signal against an ϵ-suboptimali classifier has success rate

$$S(\alpha) \geq 1 - \frac{1-\alpha}{\alpha} \Delta_0 \ \xi \ - \frac{\epsilon}{1-\epsilon}$$

Assume there is a $p > 0$ such that $P_0(y^*|x) \geq p$ for all x.

No overwhelmingly strong signal for competing label

Theorem: The feature-only strategy for planting a ξ-unique signal against an ϵ-suboptimali classifier has success rate

$$S(\alpha) \geq 1 - \frac{1-p}{p\alpha} \xi \ - \frac{\epsilon}{1-\epsilon}$$

Takeaway: as long as the signal is chosen to be unique, small collectives can succeed

Experiments on a resume classification task

Data: 30,000 resumes scraped from a freelancer gig platform

Multiclass, multilabel classification problem with 10 skills from IT sector

Model: BERT-like text transformer model (DistilBERT), fine-tuned for 5 epochs

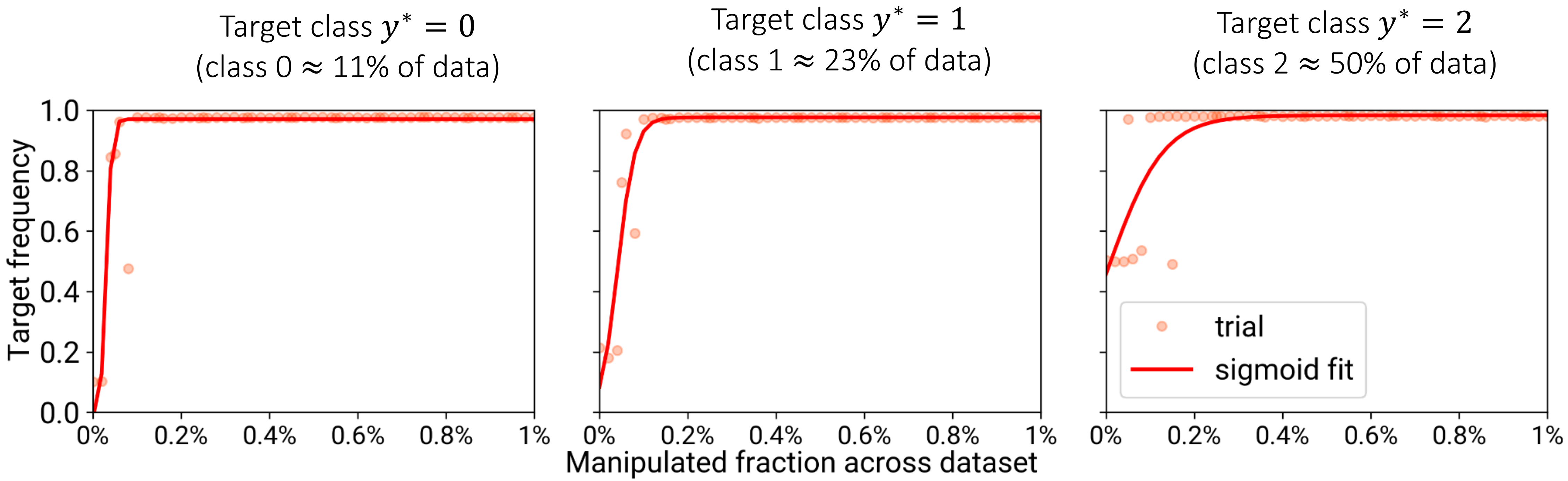
Strategy: Insert *unique* formatting symbol '-' every 20 words

Evaluation: Frequency of target label prediction on test set

(a) Target frequency: any position (typically 2-4 tags)
(b) Top-1 frequency: top 1 position

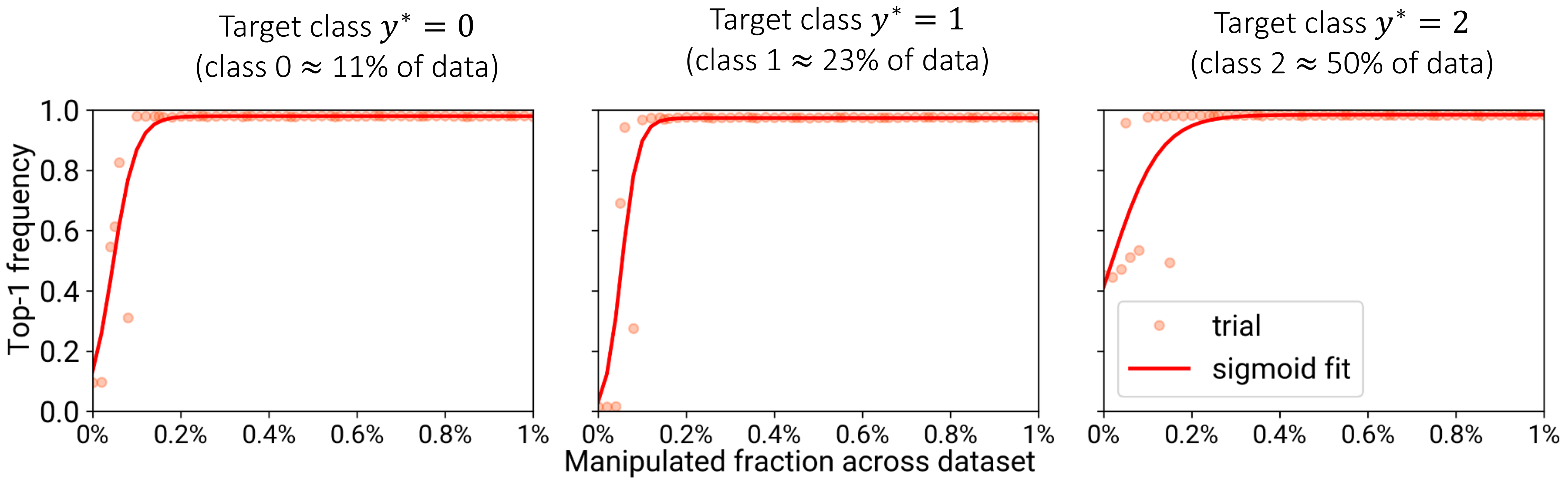
Findings from more than two thousand model training runs

Feature-label strategy



Success at **0.1%** of the data! **That's ~25 resumes.**

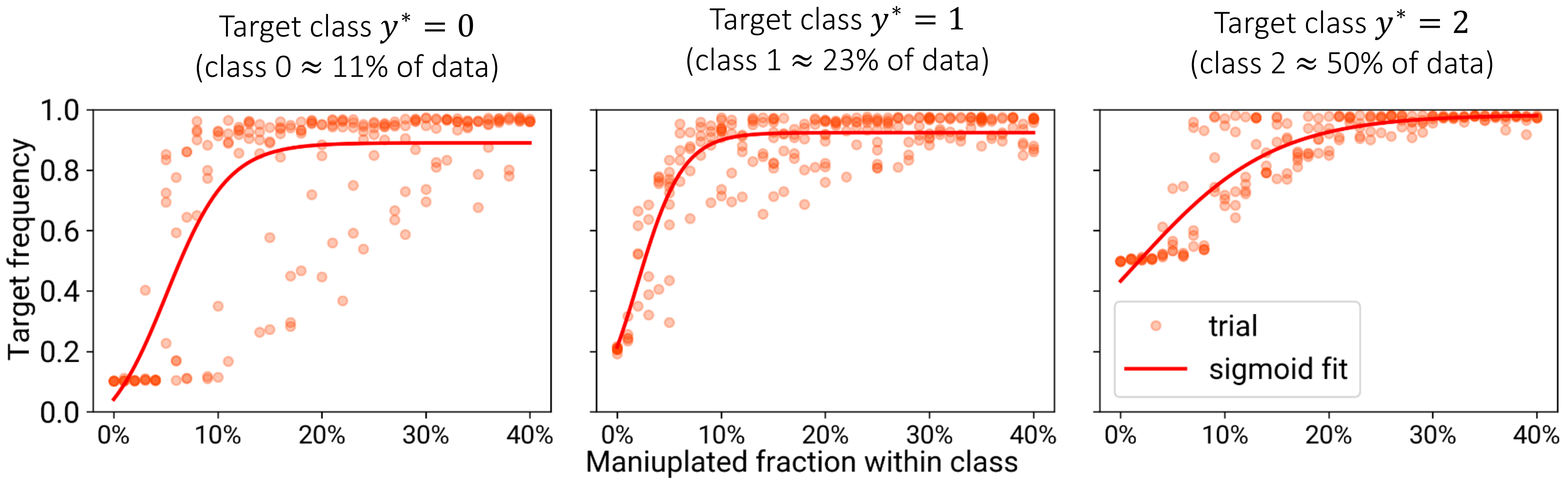
Feature-label strategy



Success at **0.1%** of the data! **That's ~25 resumes.**

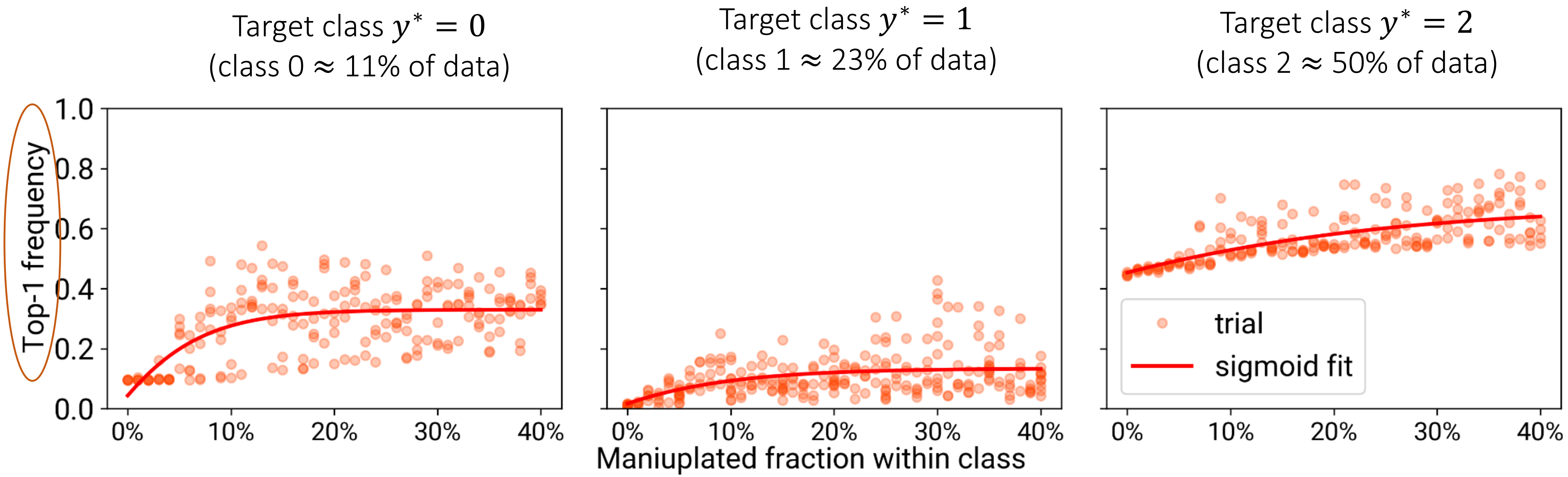
Aligned with theory for unique trigger

Feature-only strategy



Success at **1%** to **5%** of the dataset
depending on target class

Feature-only strategy



why does it not work well?

Strength of competing signal

According to our bound the feature-only strategy fails if $P_0(y^*|x)$ gets too small

This happens if features x contain overwhelmingly strong signal about the label

Can we empirically confirm that success rate goes up as the strength of competing signals diminishes?

Strength of competing signal

According to our bound the feature-only strategy fails if $P_0(y^*|x)$ gets too small

This happens if features x contain overwhelmingly strong signal about the label

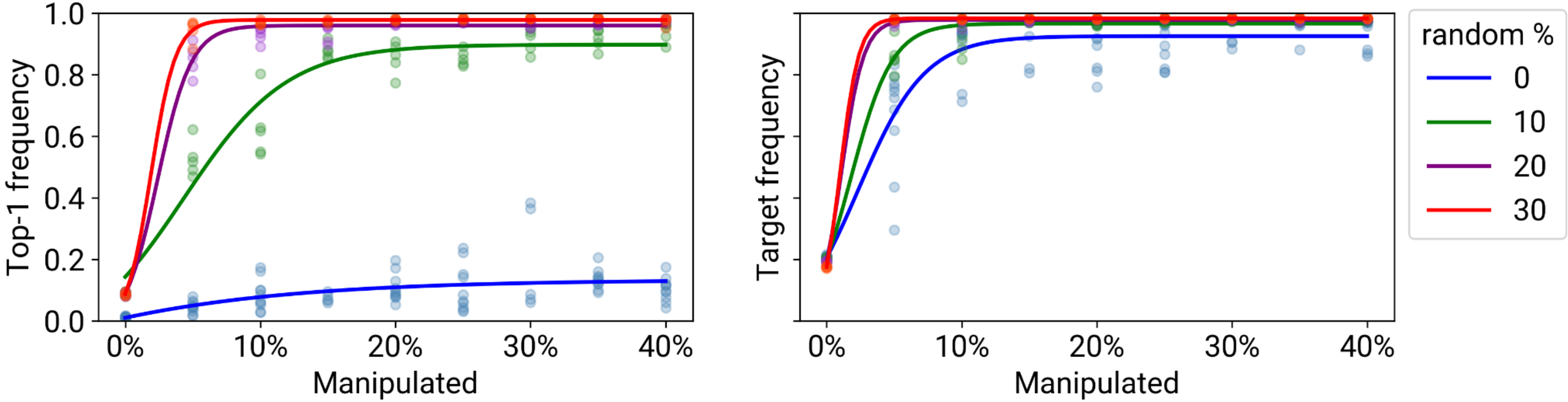
Can we empirically confirm that success rate goes up as the strength of competing signals diminishes?

Test: Randomize a fraction of the labels in the original data.

$\rightarrow$ Random labels diminish strength of competing signals.

Strength of competing signal

Target class $y^* = 1$
(class 1 ≈ 23% of data)



Test confirms:

Small label uncertainty greatly increases success of signal-only strategy

"Blessing of dimensionality"

Two other predictions our theory makes

1. Suboptimality of the predictor diminishes success rate

    Test: Vary number of epochs in training

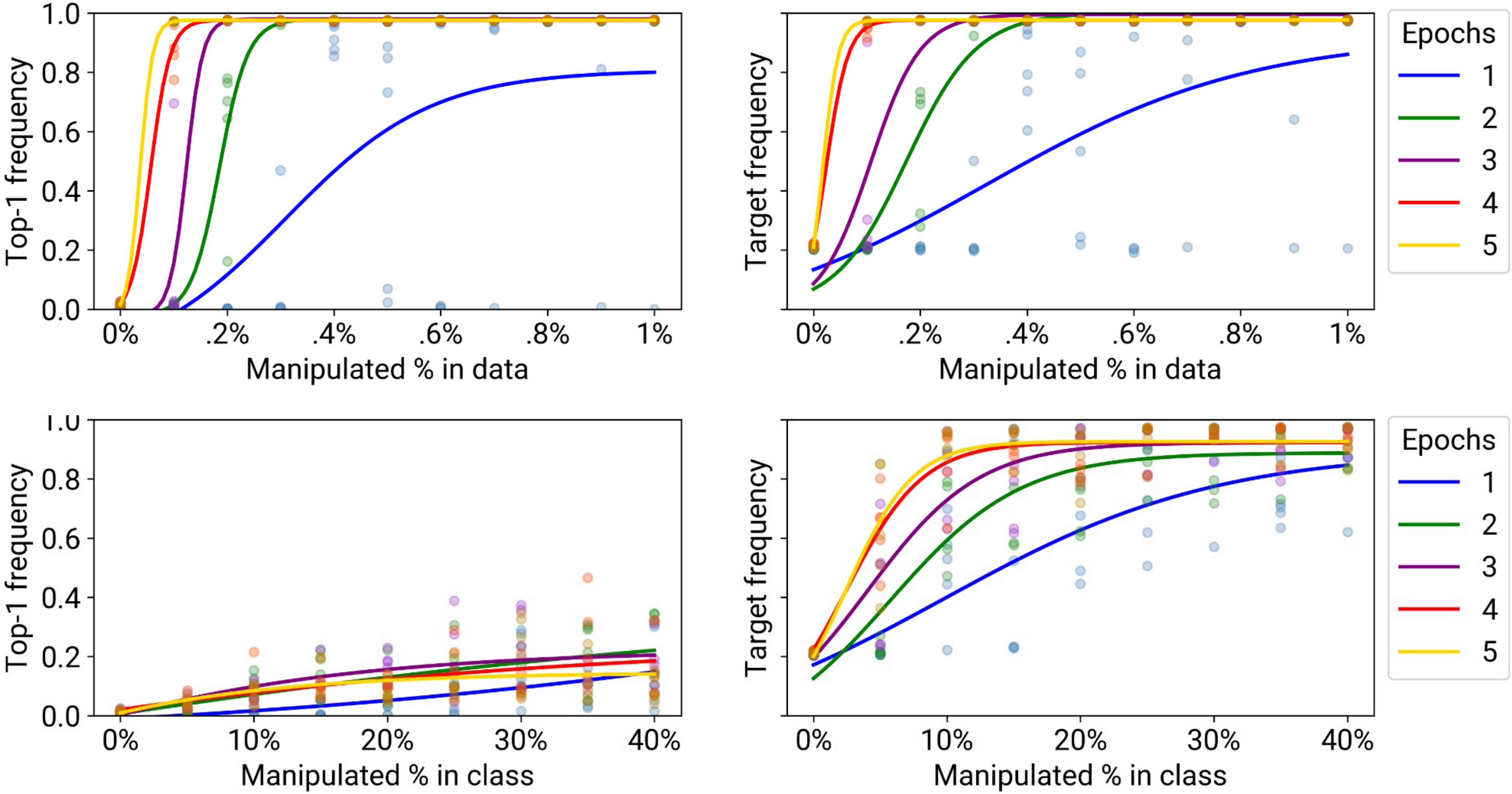
2. Uniqueness of signal set matters, not how the signal is placed

    Test: Vary spacing of signal placement

Variying number of epochs f is trained

Feature-label strategy
$(y^* = 1)$

Feature-only strategy
$(y^* = 1)$

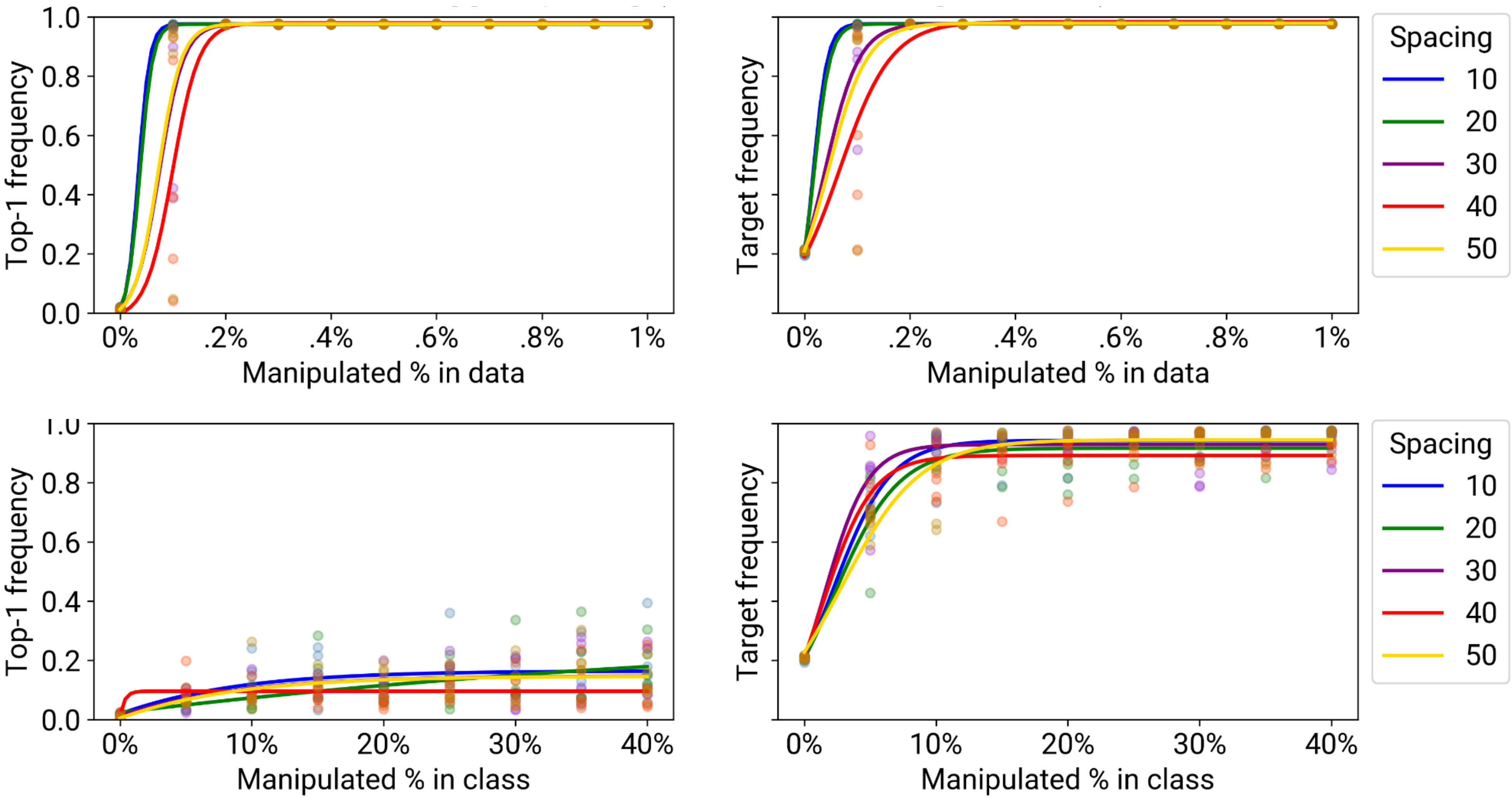


→ Suboptimality of predictor diminishes success rate

Varying trigger spacing

Feature-label strategy ($y^* = 1$)

Feature-only strategy ($y^* = 1$)



→ Uniqueness of signal set matters, not how the signal is placed

In praise of Bayes optimality

Simple theory for Bayes optimal predictor turns out to be surprisingly predictive

Perhaps an indication that the language model approximates likelihood well

As an aside, not the only case where Bayes optimal comes in handy in ML

Tabular data (large *n*, small *d*) generally admits models close to Bayes optimality.

There's more theory we can do

- Signal **erasure** strategies:
  "success scales with unique information contained in the signal to be removed"
- **Regression** under squared loss: Platform chooses $f(x) = \mathrm{E}[y|x]$

Parametric risk minimization

Platform learns parametric model f_θ by minimizing a risk function

$$\theta = \mathrm{argmin}_{\theta'} \mathrm{E}_{z\sim P}\, \ell(\theta'; z)$$

Collective wants to reach target model θ^*.

Parametric risk minimization

Platform learns parametric model f_θ by minimizing a risk function

$$\theta = \mathrm{argmin}_{\theta'} \, \mathrm{E}_{z\sim P} \, \ell(\theta'; z)$$

strictly convex loss

Collective wants to reach target model θ^*.

Gradient canceling strategy exists for GLMs where $\nabla\ell\big(\theta; (x,y)\big) = \gamma x$

Convex risk minimizer.

- *Gradient canceling strategy*: Choose distribution P^* such that for some $t > 0$:

$$\mathrm{E}_{z\sim P^*} \, [\nabla\ell(\theta^*; z)] = -t \, \mathrm{E}_{z\sim P_0}[\ell \, \nabla(\theta^*, z)]$$

Proposition: The collective can reach the target θ^* for some $\alpha \le 1/(1+t)$

→ target models θ^* that look more optimal on the base distribution are easier to achieve

Parametric risk minimization

Platform learns parametric model f_θ by minimizing a risk function

$$\theta = \mathrm{argmin}_{\theta'} \mathrm{E}_{z\sim P}\, \ell(\theta'; z)$$

non-convex

Collective wants to reach target model θ^*.

Gradient learner:

- Collective gets to modify distribution in every step (e.g., federated learning)

model update: $\theta_{t+1} = \theta_t - \eta\, \mathrm{E}_{z\sim P_t}\, \nabla\ell(\theta_t; z)$

Informal Result:
- Collective size related to the magnitude of the largest gradient encountered along the path $\theta_0 \to \theta^*$ measured on P_0
- Convergence occurs at convex rate despite non-convex loss

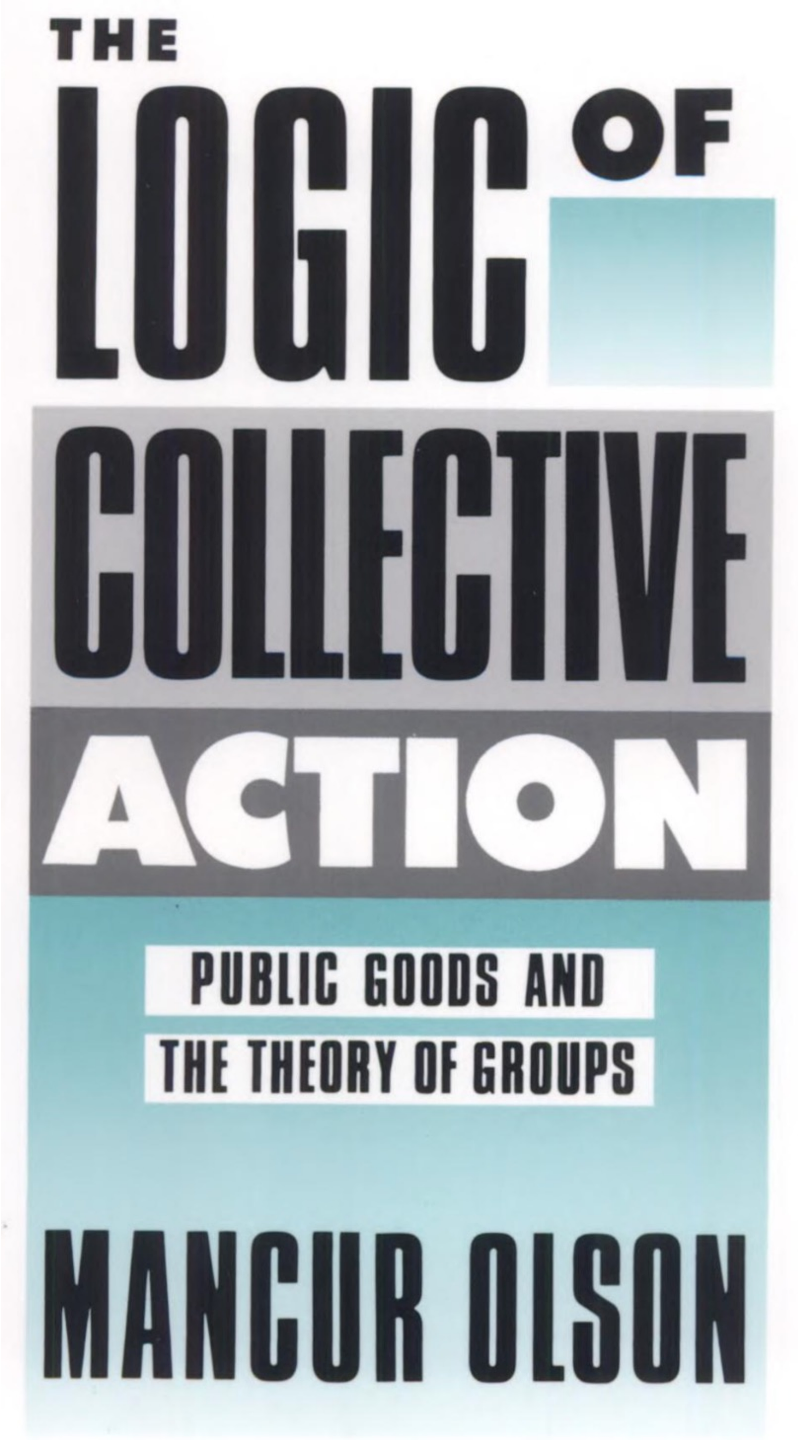
What about incentives?

Free riding (Olson, 1965)

- Collective can share signal function only with participants
- Technology exists for that

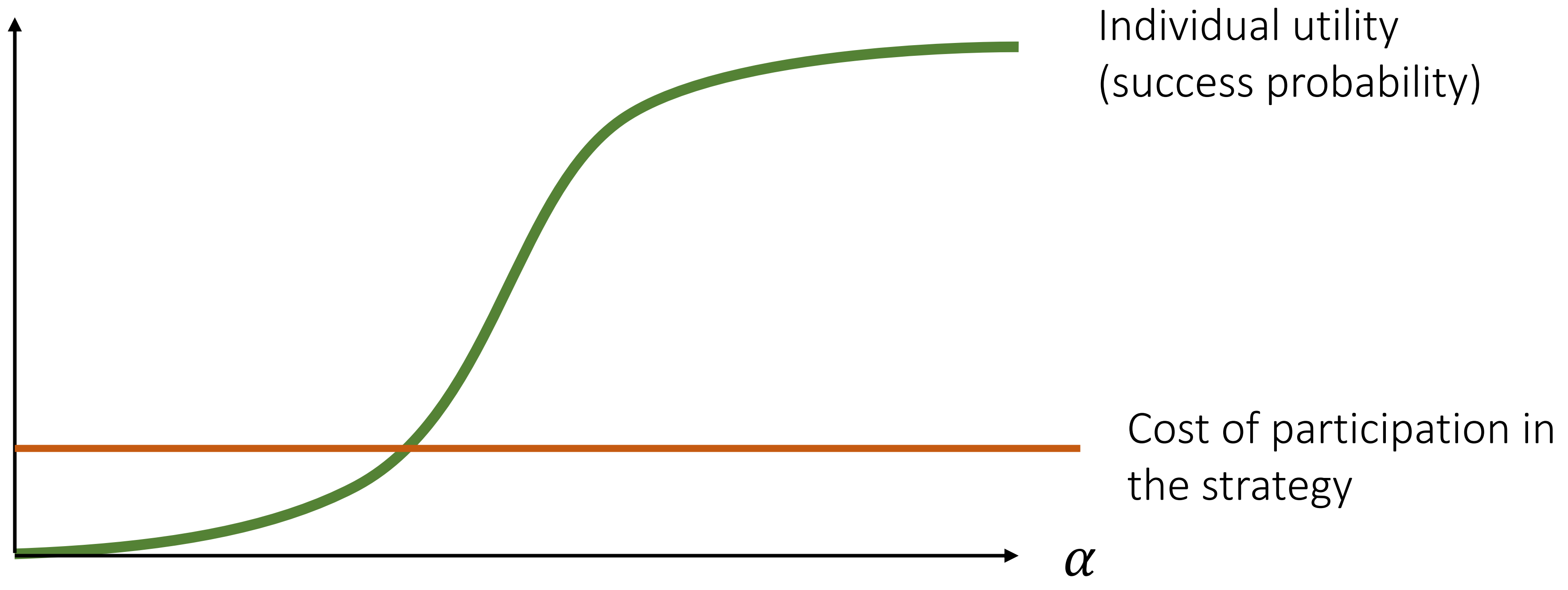
Early adoption

- Initially no inherent pay off to first participants
- This is where critical threshold comes in



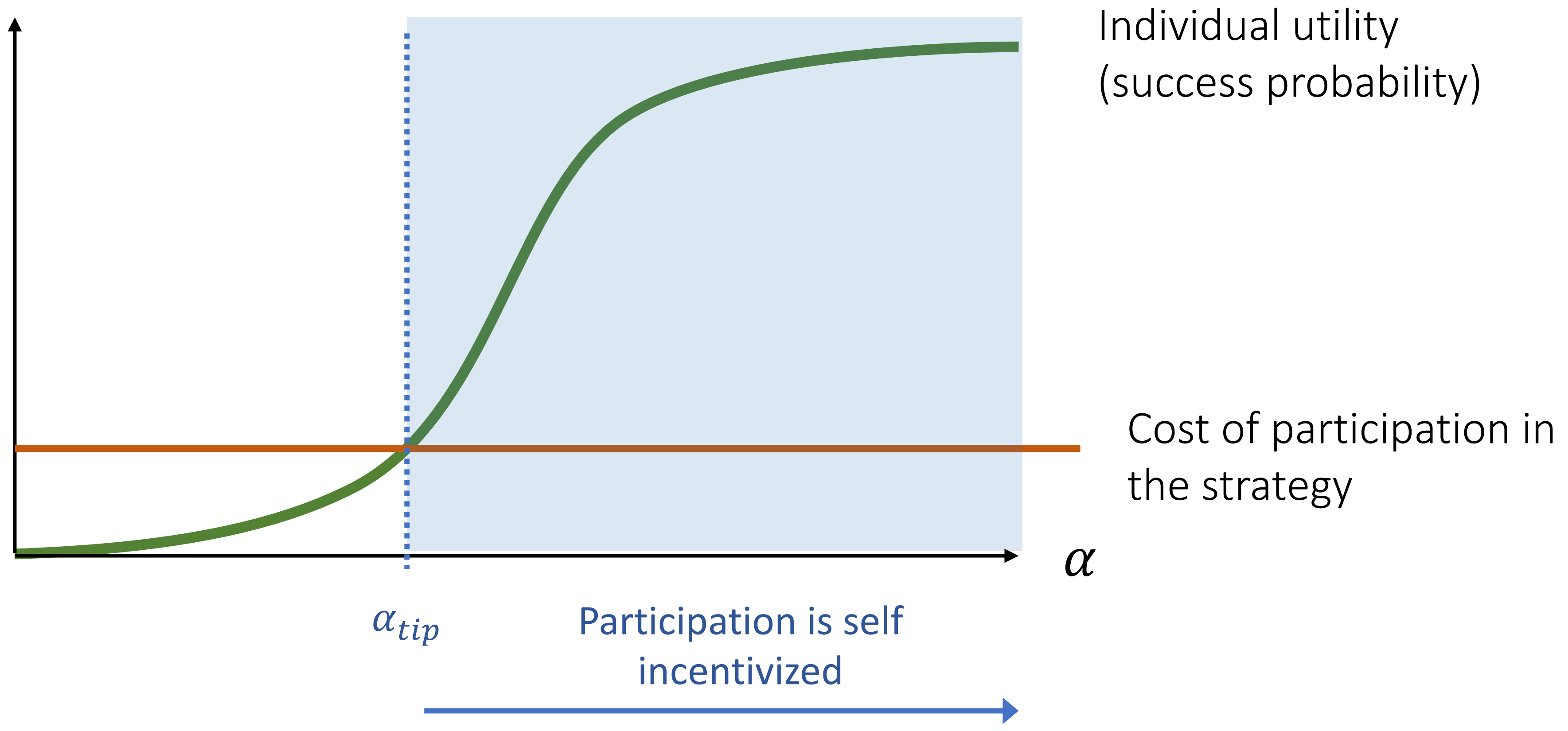
Critical threshold for algorithmic collective action

Assumption: ‘exclusive good’, e.g., signal function is kept secret



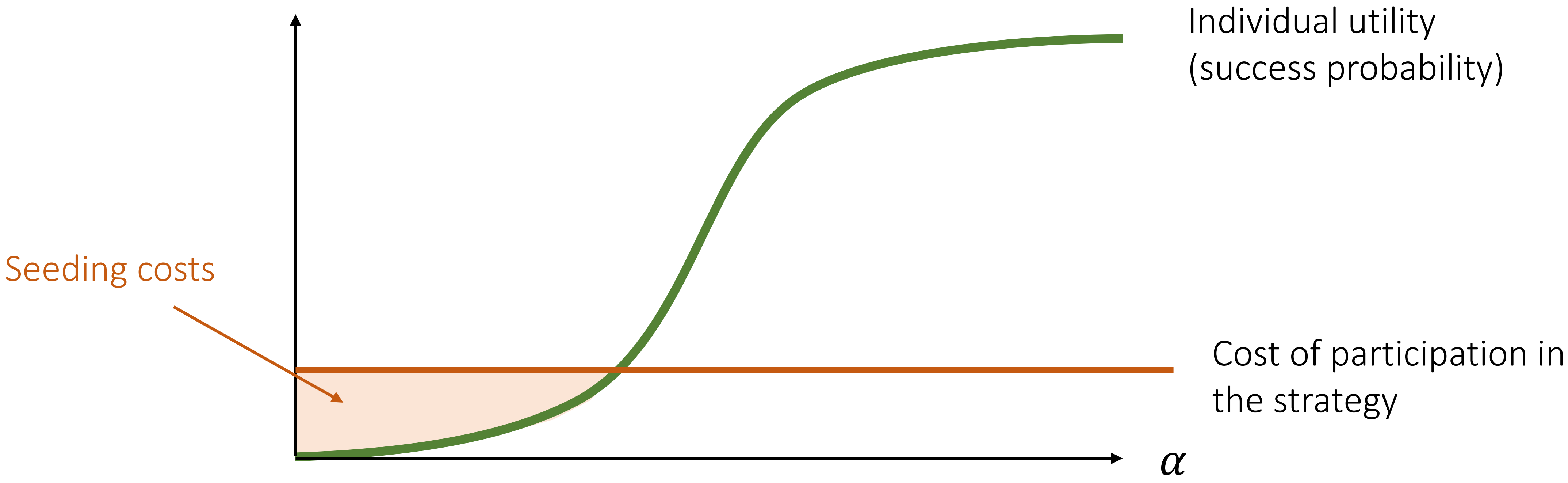
Critical threshold for algorithmic collective action

Assumption: ‘exclusive good’, e.g., signal function is kept secret



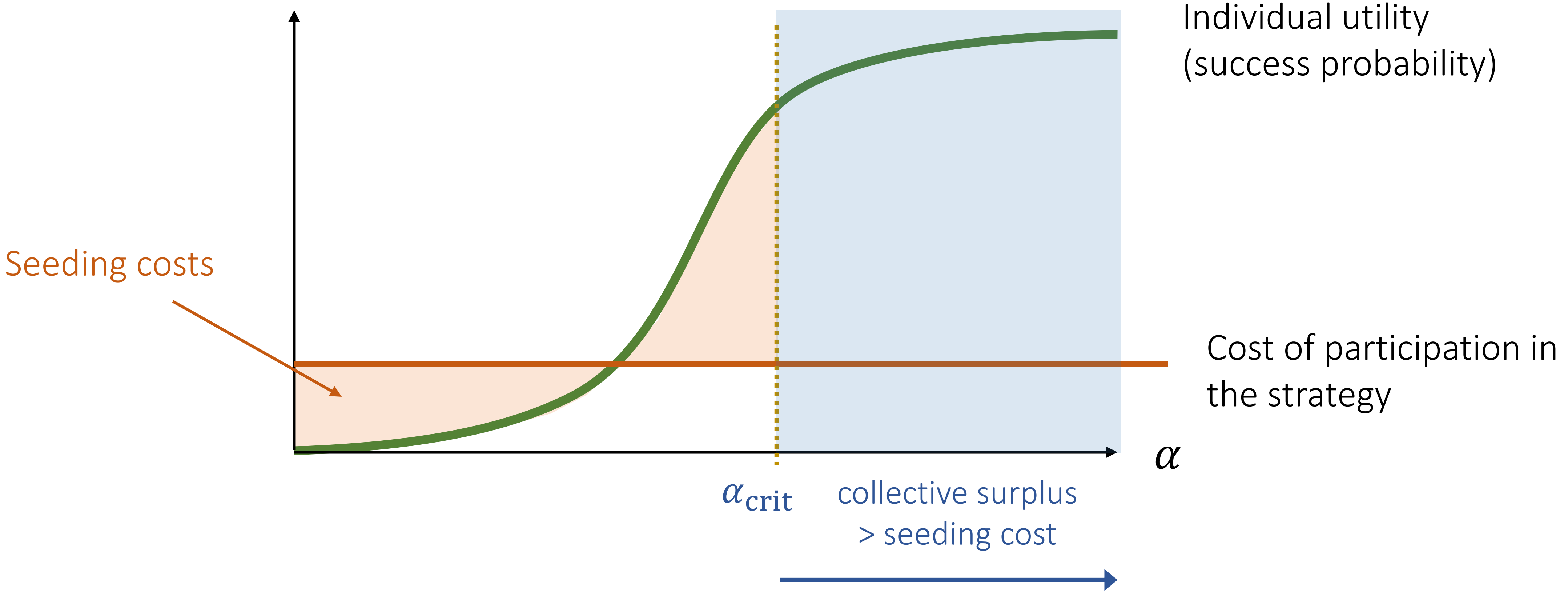
Critical threshold for algorithmic collective action

Assumption: ‘exclusive good’, e.g., signal function is kept secret



Critical threshold for algorithmic collective action

Assumption: ‘exclusive good’, e.g., signal function is kept secret



Imagine a future of platform labor

For each platform app

Uber

Making the market

There is a labor app

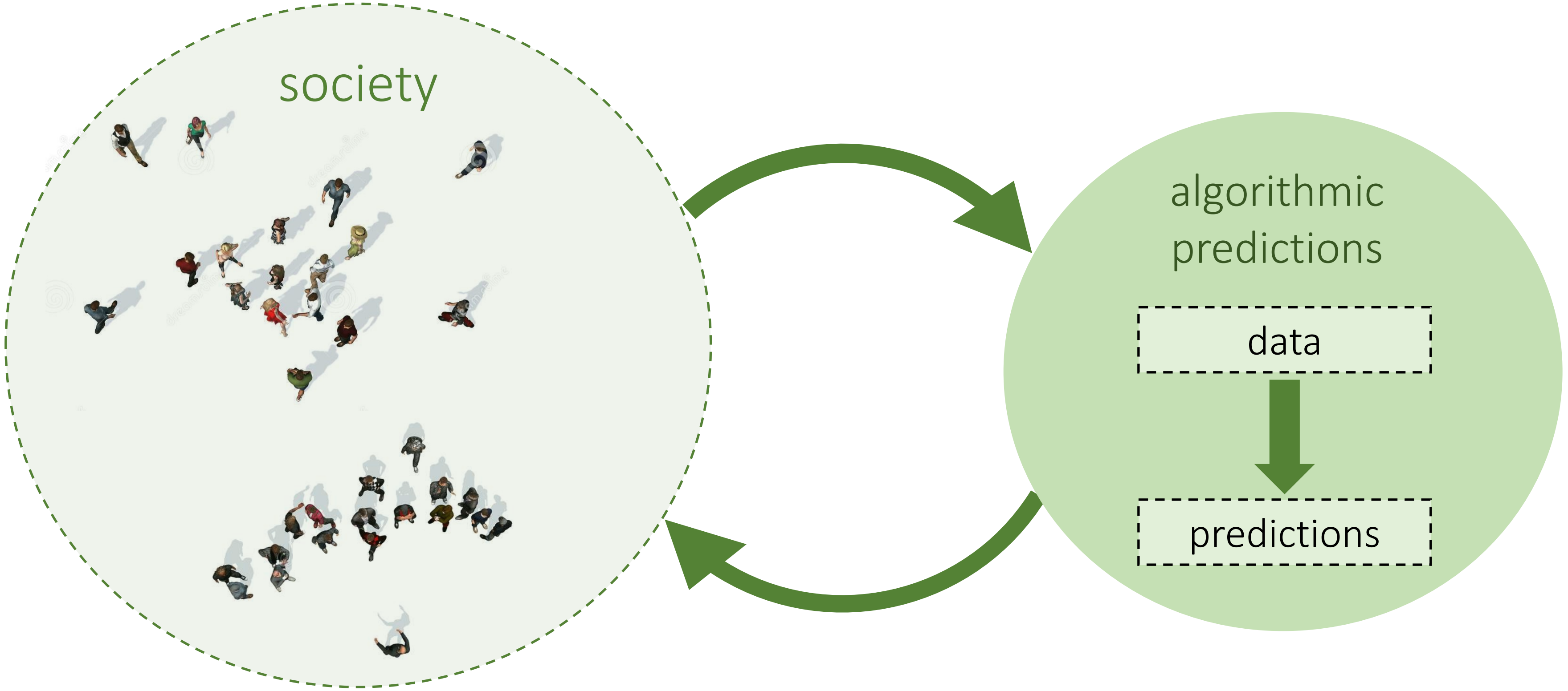
Under

Coordinating labor

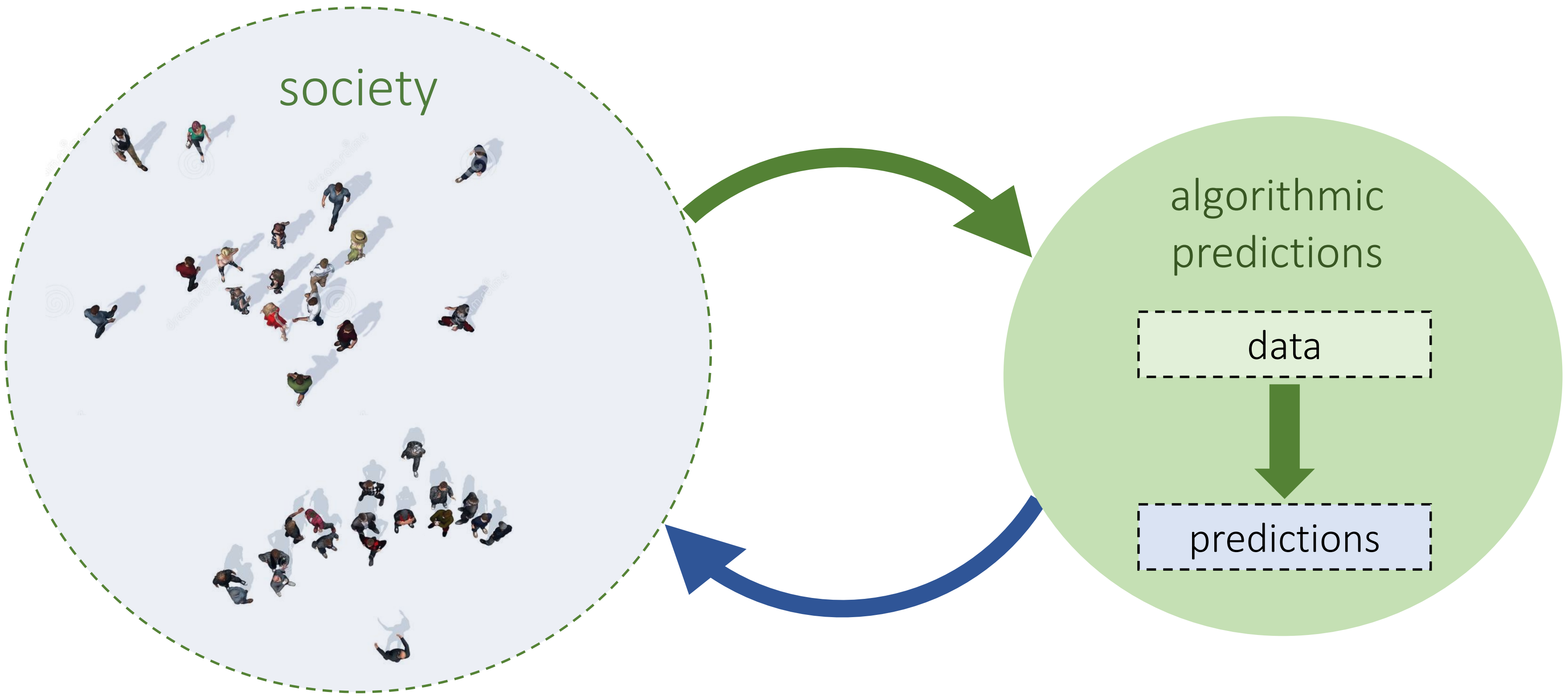
What new equilibria arise?

Potential: More favorable labor outcomes,
more competitive markets

Dynamics in predictive systems

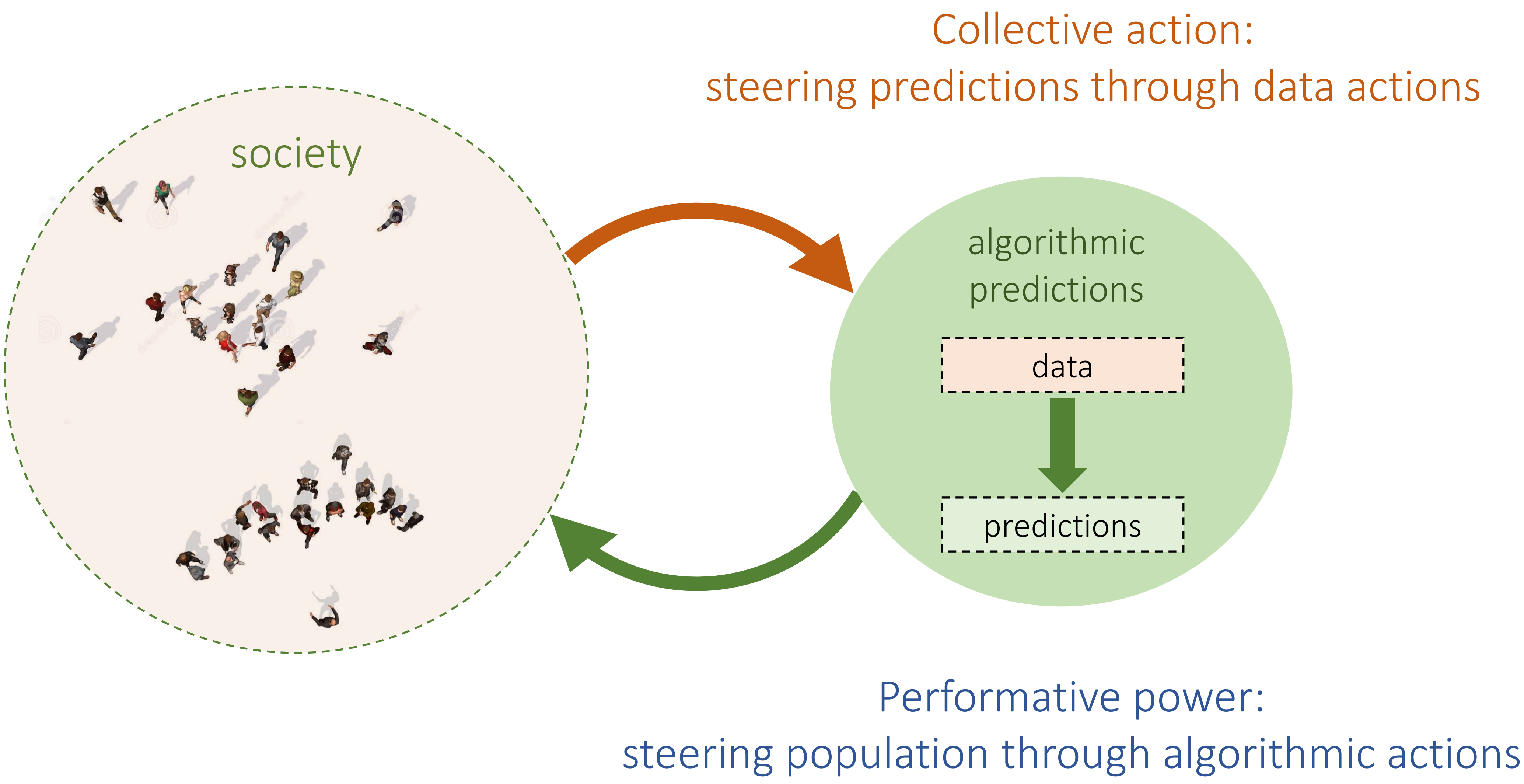


Dynamics in predictive systems



Performative power:
steering population through algorithmic actions

Dynamics in predictive systems



Going further

Finite sample analysis

- Connection between collective success and signal-to-noise ratio in data
- Connection between collective success and generalization ability of the learning algorithm (memorization capacity)

Other collective strategies apart from signal and erasure strategies?
Other collective goals?
More complex utility functions?

Empirical work: Other data domains (vision, speech, tabular), other problems

Going further

Game-theoretic and economic considerations

- Incentive design
- How do collectives form?
- Modeling existing collective action strategies
- Relationship to power and competition in digital economies (cf. Performative Power [HJM22])

How to use information advantage of collectives?

Mechanisms for organizing?

Potential negative results and lower bounds

Questions, thoughts, suggestions?

cmendler@tuebingen.mpg.de

More examples

- Waze jams neighborhood
  → People carry phones through streets
- Youtube extensively upvotes polls
  → Content creators excessively use it to fix this
- Uber has a high profit margin
  → Coordinated logoffs to trigger surge pricing
- Doordash pays low vages
  → Coordinated rejection of low price offers

...

DiDi

Thrown under the bus and outrunning it! The logic of Didi and taxi drivers' labour and activism in the on-demand economy

Julie Yujie Chen View all authors and affiliations

CNET

Your quote tweets make bad tweets worse. Do this instead

Waze to go: residents fight off crowdsourced traffic… for a while

Uber & Lyft Drivers Reportedly Rigging App to Create Surge Pricing

"And we all know, rule number one, we don't talk about 'Surge Club.'"

INDEPENDENT

NEWS VOICES SPORT CULTURE INDY/LIFE INDYBEST VIDEO DAILY EDITION

News > Business > Business News

Uber drivers work together to create price surge and charge customers more, researchers find

Some drivers are deli

when they log back i

Ben Chapman | @b_c_chap

DIGITAL TRENDS

Trending: Vape Ban Disney+ Review Early Black Friday Deals

Uber drivers reportedly triggering higher fares through Surge Club

By Aaron Mamiit June 16, 2019 8:08PM PST