slides available at:

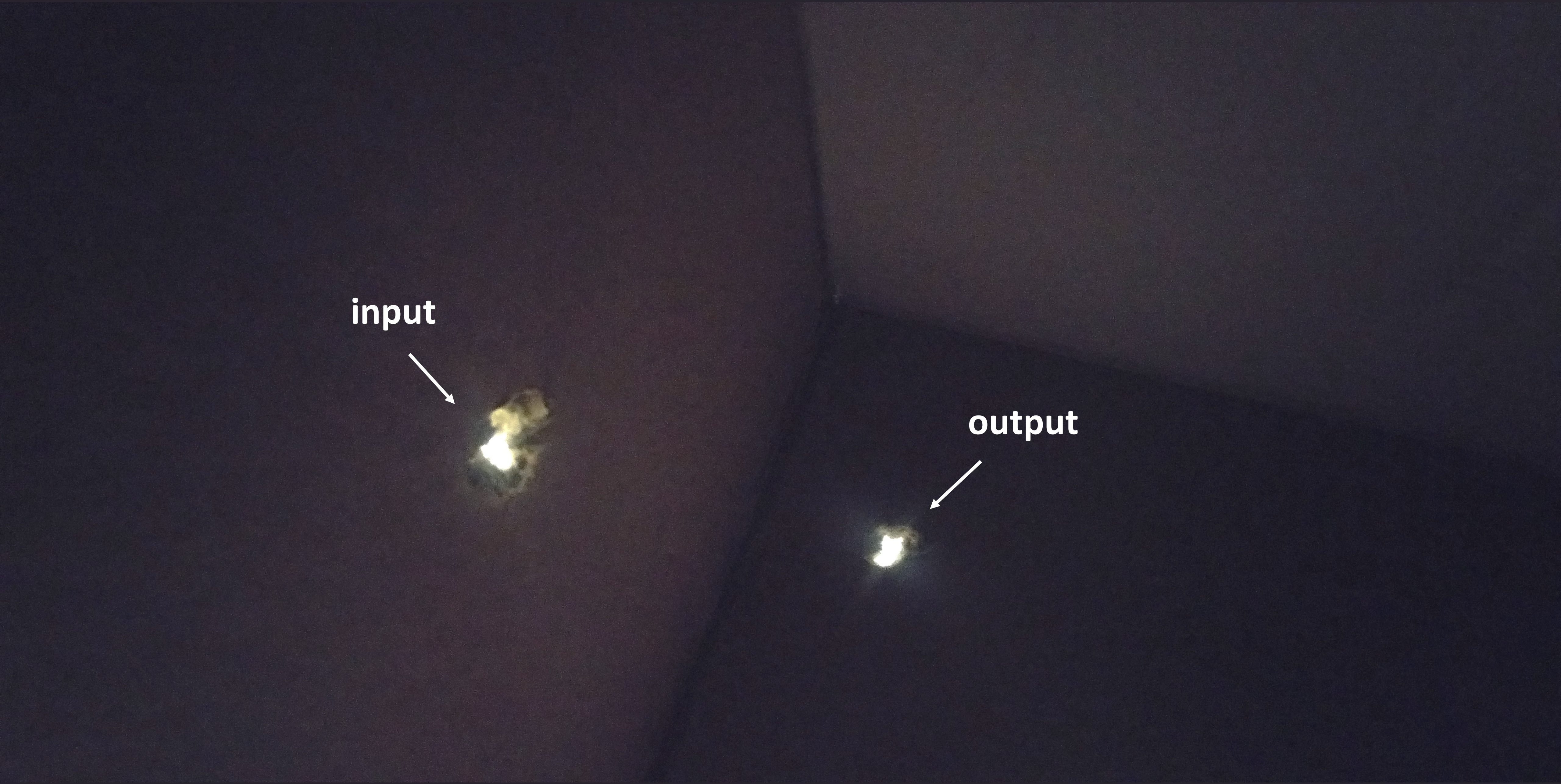
Strategic ML:
How to Learn With Data That 'Behaves'

Nir Rosenfeld

Technion CS

tutorial @ ICML 2024

outside → in:


input
output

this is machine learning:

this is machine learning on images:

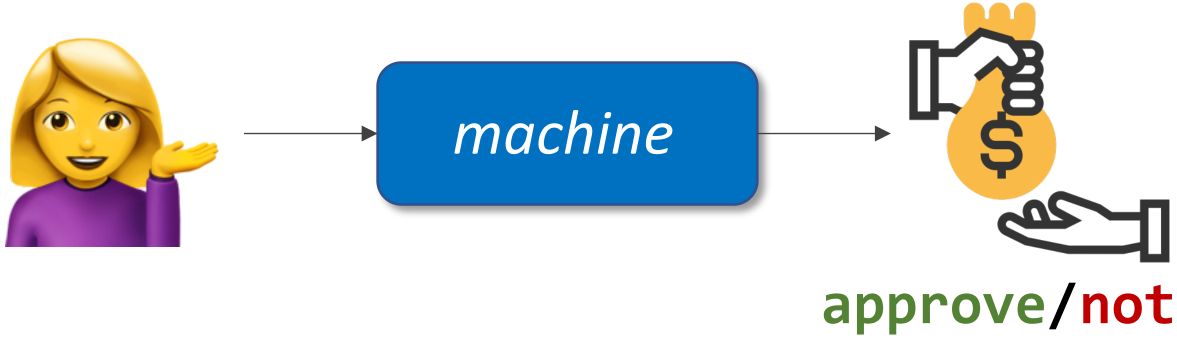
this is machine learning on text:

this is machine learning on... humans?

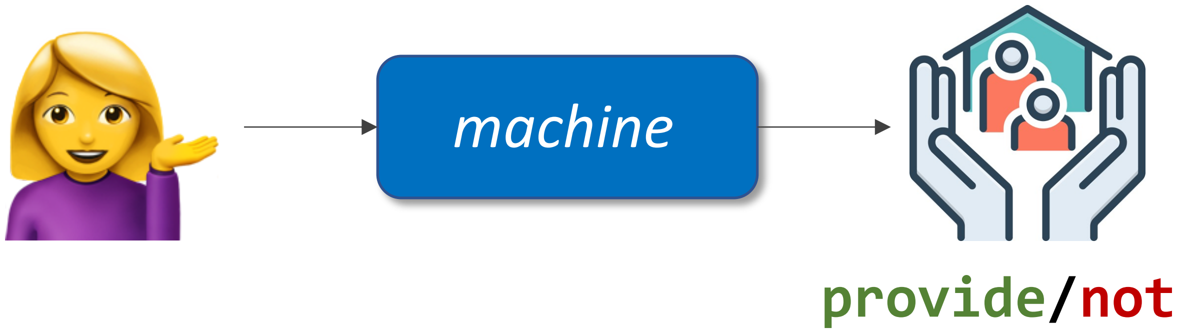
this is machine learning on... humans!

this is machine learning on… humans!

this is machine learning on... humans!



this is machine learning on... humans!



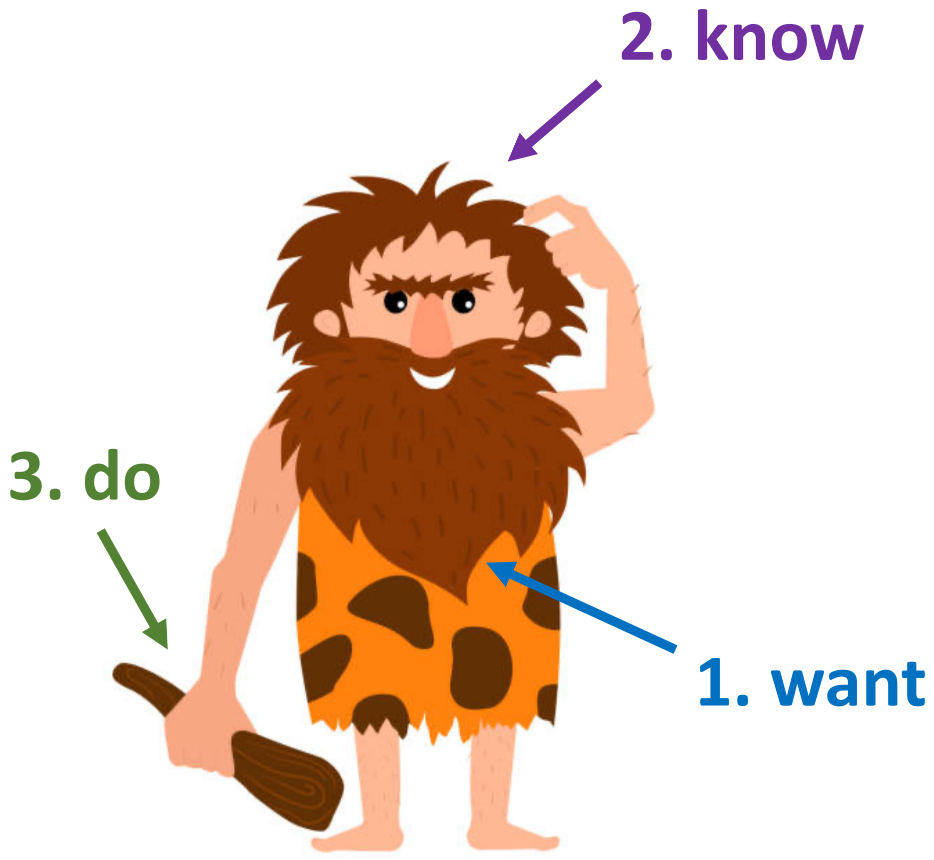
what could possibly go wrong?

(**or**: how does human behavior change learning and its outcomes?)

Strategic classification

- Builds on **conventional binary classification**
- **Augments** to account for **human behavior**
- Models humans as **inputs with agency**

- Allows (and requires!) to encode what humans:
- Basic elements of economic modeling
- Together, combine to determine **how humans behave**



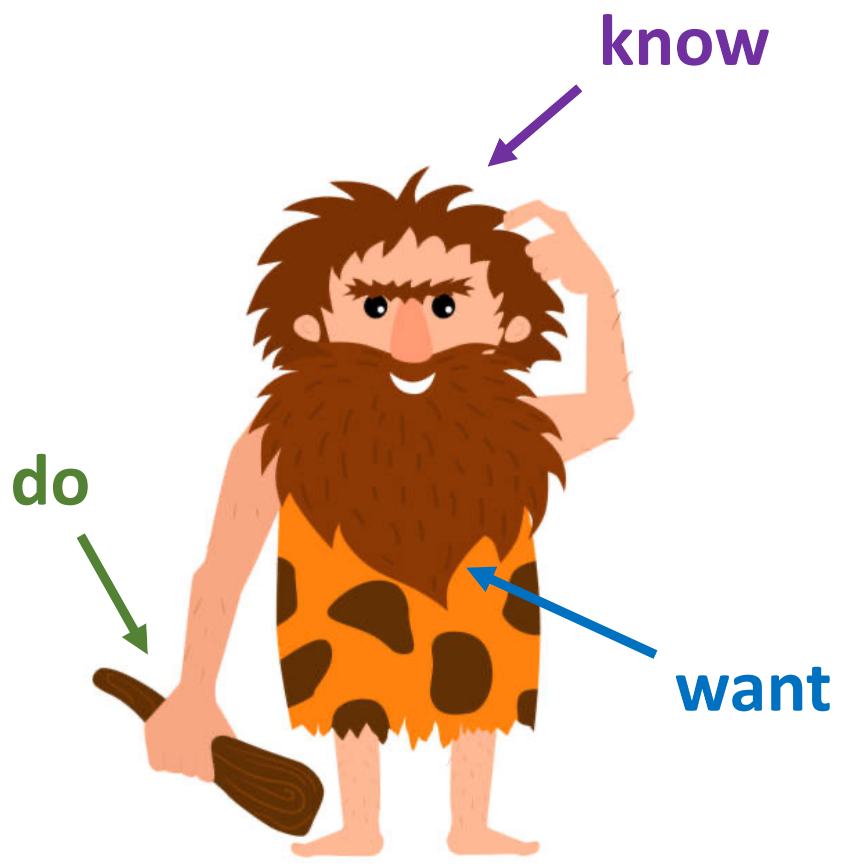
homo-sapiens

modeling challenge: weave these into learning setup

Strategic classification

- **SC is great** because it is:
  - **simple enough** to permit tractable analysis
  - **powerful enough** to introduce novel challenges
  - **meaningful enough** to have social implications
  - **flexible enough** to permit extensions, variations, and generalizations

- Start with **rigid assumptions** – e.g., rationaity:

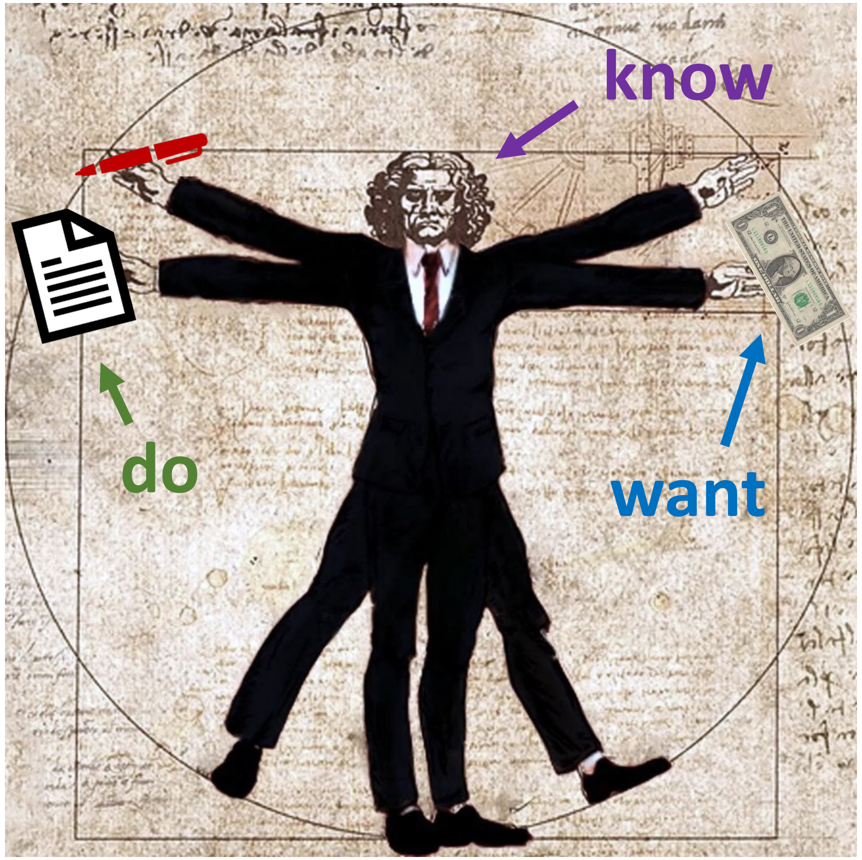


homo-sapiens

Strategic classification

- **SC is great** because it is:
  - **simple enough** to permit tractable analysis
  - **powerful enough** to introduce novel challenges
  - **meaningful enough** to have social implications
  - **flexible enough** to permit extensions, variations, and generalizations

- Start with **rigid assumptions** – e.g., rationaity:

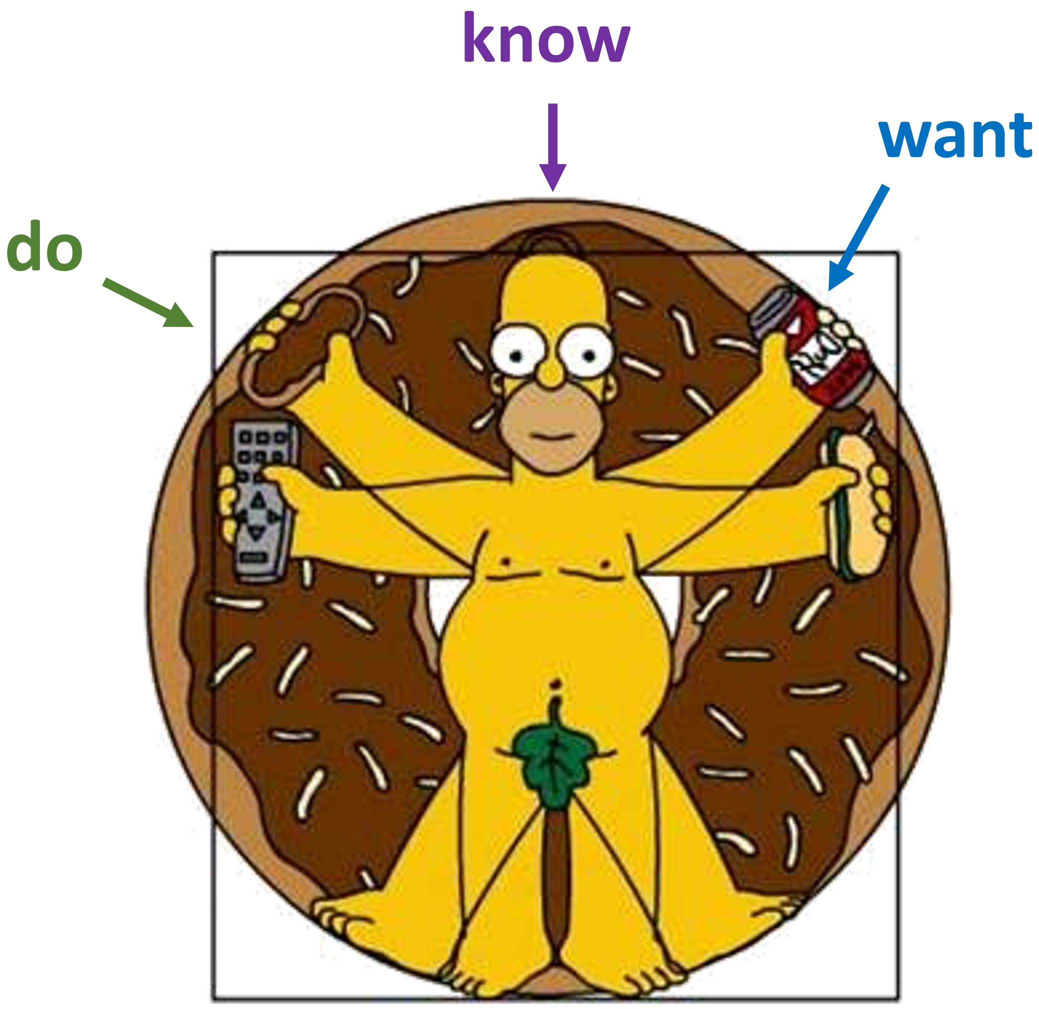


homo-economicus

Strategic classification

- **SC is great** because it is:
  - **simple enough** to permit tractable analysis
  - **powerful enough** to introduce novel challenges
  - **meaningful enough** to have social implications
  - **flexible enough** to permit extensions, variations, and generalizations

- Start with **rigid assumptions** – e.g., rationaity
- **Ultimate goal**: capture realistic behavior “in the wild”



homor-simpsonus

Outline

- **Introduction**
- Three main sections:

I) **ML aspects**: (~40 min)
- strategic learning – setup
- as learning vs. as a game
- optimization
- generalization (stats)
- modeling

II) **Econ/GT aspects**: (~60 min)
- incentives (=want)
- information (=know)
- actions (=do)
- limited resources
- social welfare

III) **Beyond**: (~20 min)
- causality
- dependencies
- over time

- **Challenges and opportunities**
- **Summary**

Tutorial theme and goals

- Introduction to **emerging new field**
- Many **open research questions**
- Much **potential for application**
- **Main theme**: transitioning from **theory** $\mapsto$ **practice**
- Focus on **supervised batch setting** (covers "half" of literature; other part being online)

- More **breadth** (less **depth**) $\to$ *see references*
- More **modeling** (less **results**)
- More **questions** (less **answers**)
- More **content** (less **time**) $\to$ *fast paced!*

Strategic classification

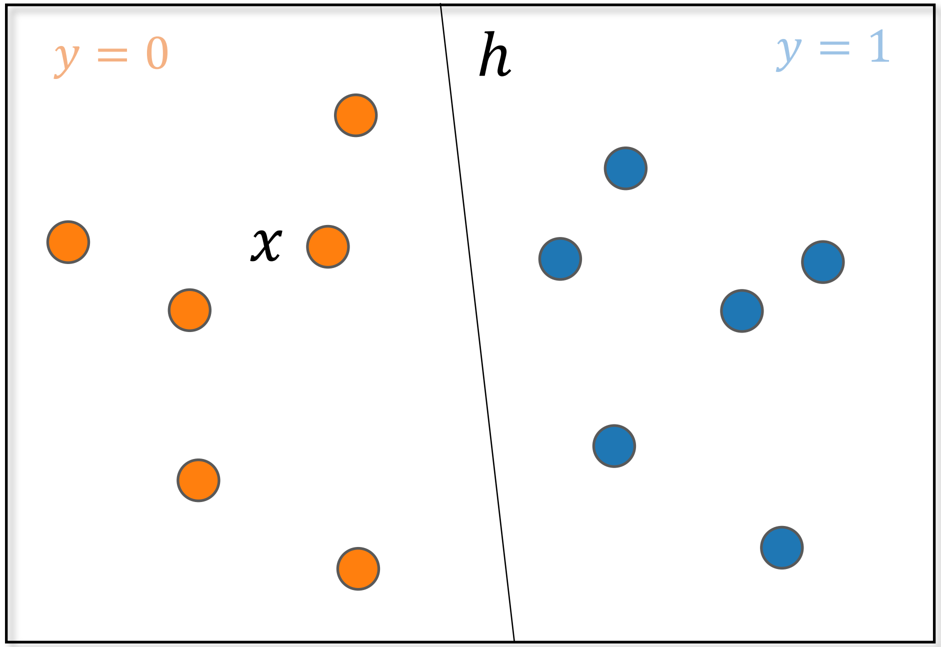
an introduction

standard classification:

learned model

train: $\underset{h}{\operatorname{argmin}}\ \mathbb{E}[\mathbb{1}\{y \neq h(x)\}]$

input features



standard classification:

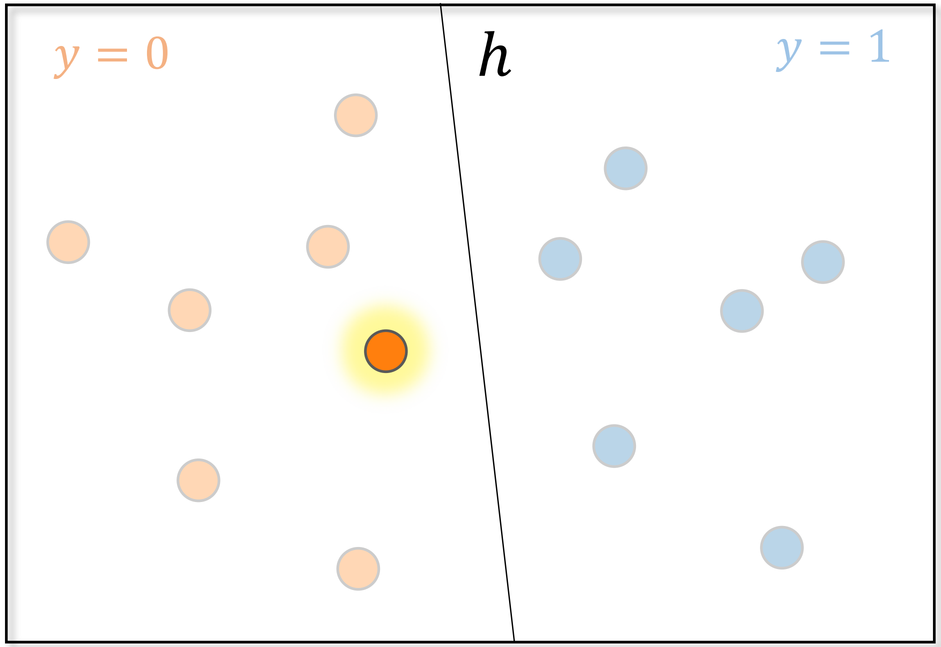
learned model

train: $$\underset{h}{\operatorname{argmin}}\ \mathbb{E}[\mathbb{1}\{y \neq h(x)\}]$$

input features

test: $$h(x) = \hat{y} \approx y$$

prediction *ground truth*



strategic

classification:

[BS'2011, HMPW'2016]

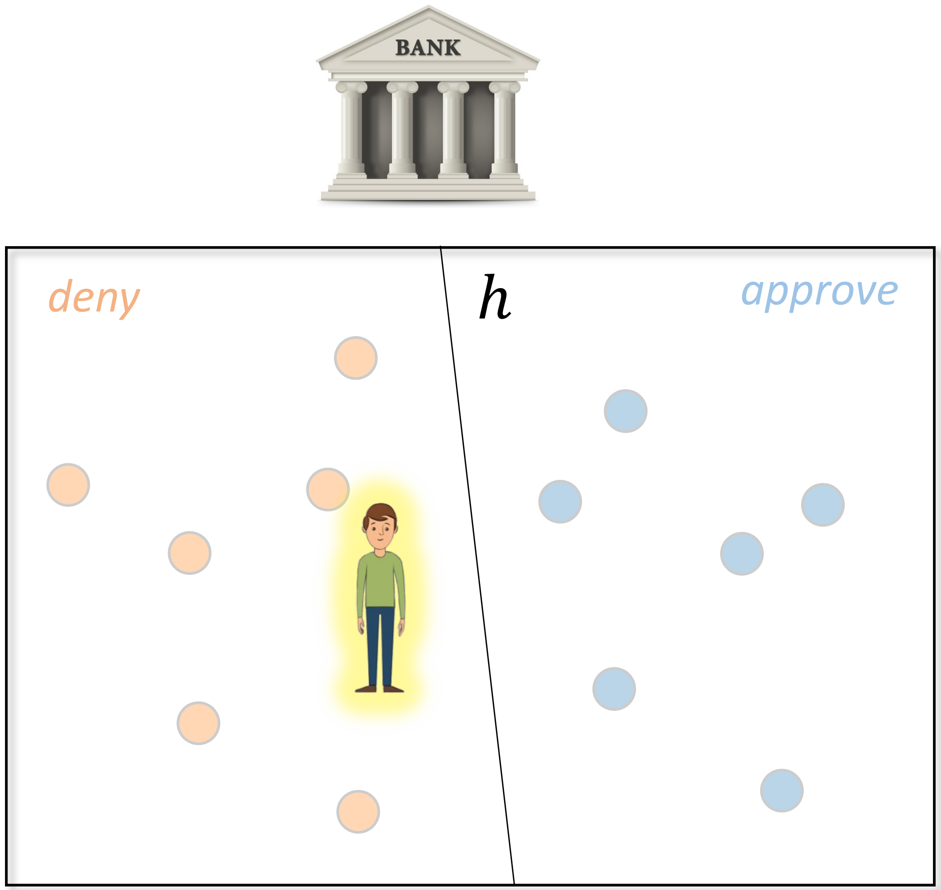
learned model

train: $\underset{h}{\operatorname{argmin}}\ \mathbb{E}[\mathbb{1}\{y \neq h(x)\}]$

representation of ***human agent***

test: $h(x) = \hat{y} \approx y$

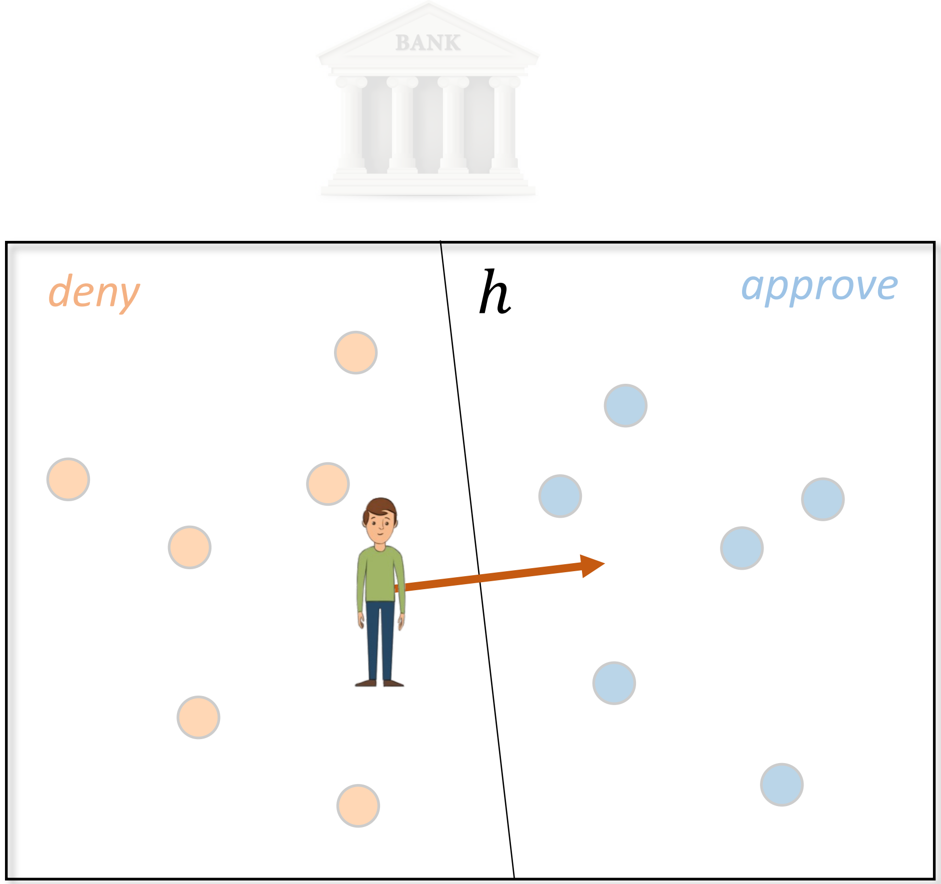
prediction *ground truth*



strategic
classification:

train: $\operatorname*{argmin}_{h} \mathbb{E}[\mathbb{1}\{y \neq h(x)\}]$

test: $h(x) = \hat{y} \approx y$

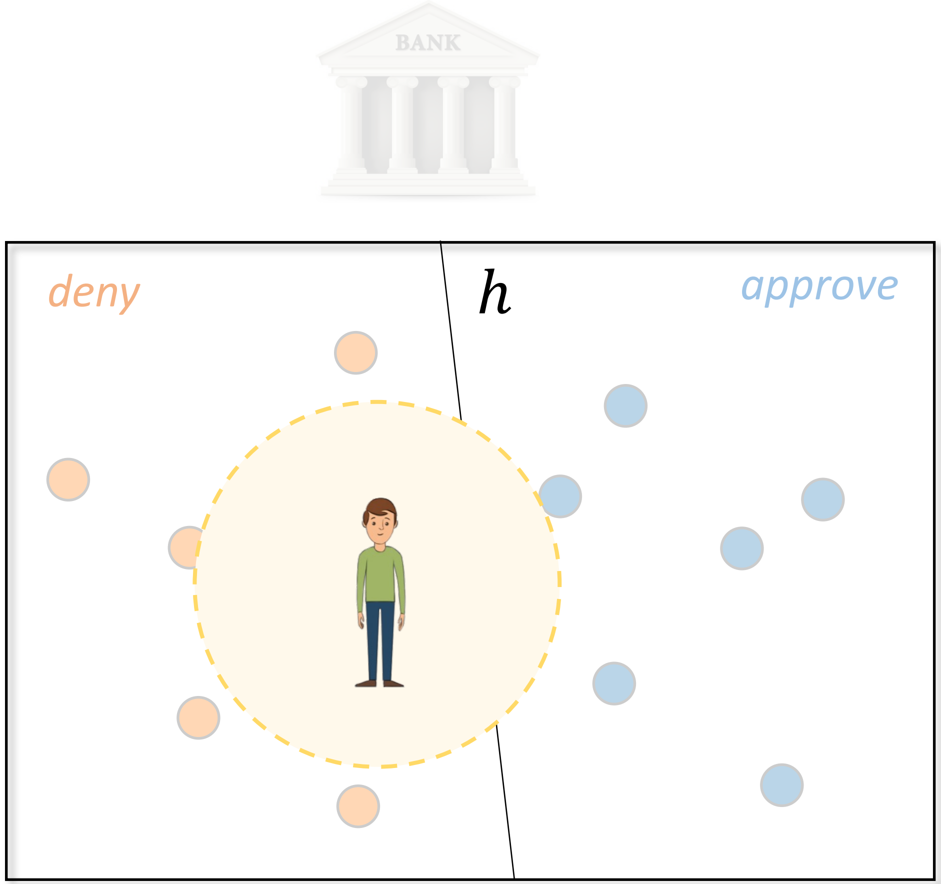


1. **want:** $\hat{y} = 1$ (get the loan)

strategic
classification:

train: $\operatorname*{argmin}_h \mathbb{E}[\mathbb{1}\{y \neq h(x)\}]$

test: $h(x) = \hat{y} \approx y$



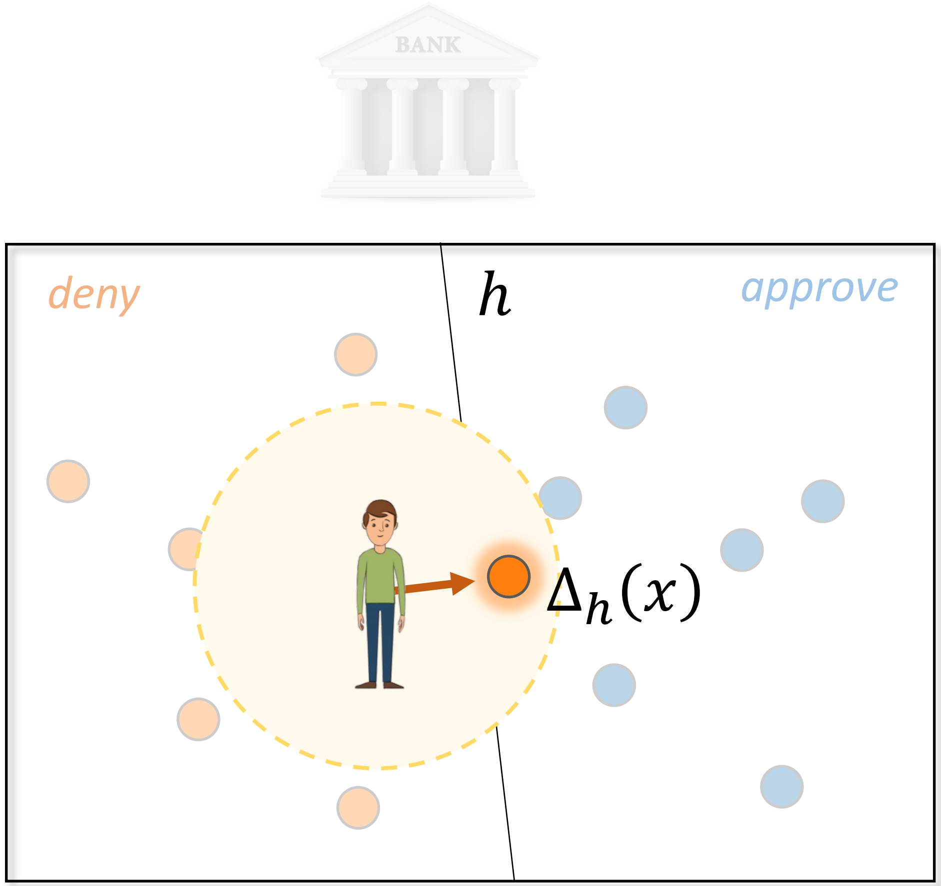
1. **want:** $\hat{y} = 1$ (get the loan)
2. **do:** modify features (at cost)

strategic

classification:

train: $\underset{h}{\operatorname{argmin}}\ \mathbb{E}[\mathbb{1}\{y \neq h(x)\}]$

test: $h(x) = \hat{y} \approx y$



1. **want:** $\hat{y} = 1$ (get the loan)
2. **do:** modify features (at cost)
3. **know:** h (and cost function)

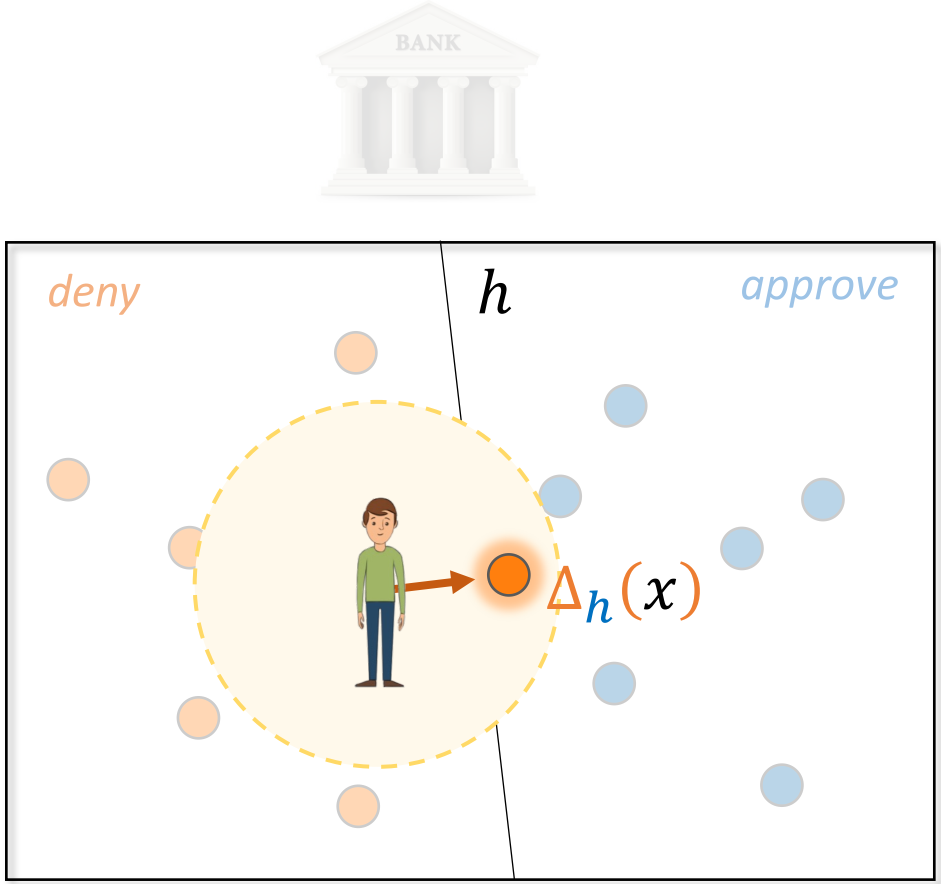
strategic
classification:

train: $\underset{h}{\operatorname{argmin}}\ \mathbb{E}[\mathbb{1}\{y \neq h(x)\}]$

test: $h(x) = \hat{y} \approx y$

response: $x \mapsto x^h \triangleq \Delta_h(x)$

behavior



1. **want:** $\hat{y} = 1$ (get the loan)
2. **do:** modify features (at cost)
3. **know:** h (and cost function)

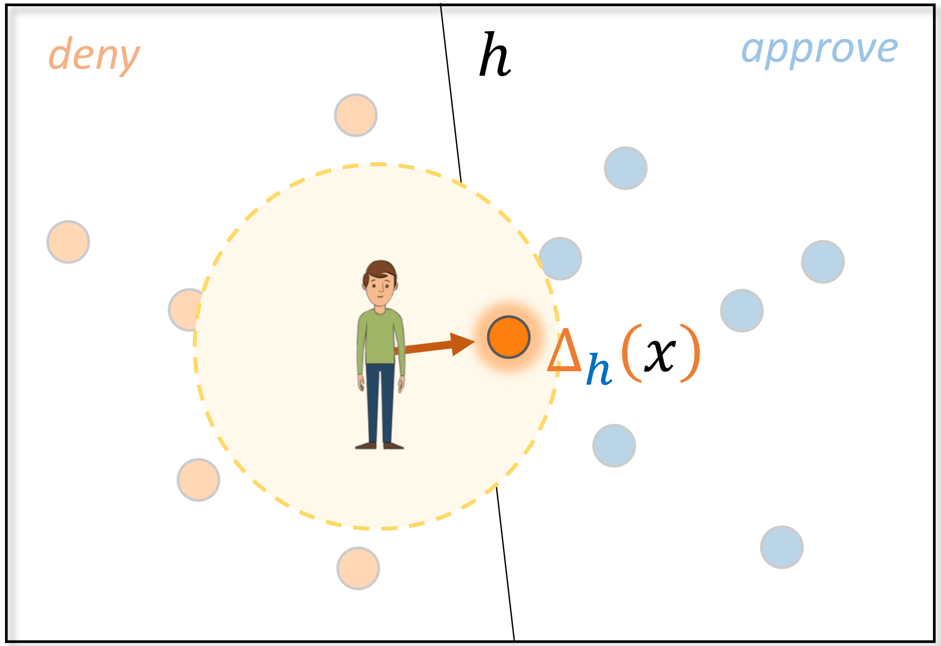
strategic
classification:

train: $\operatorname{argmin}_h \mathbb{E}[\mathbb{1}\{y \neq h(x)\}]$

test: $h(x) = \hat{y} \approx y$

response: $\Delta_h(x) = \operatorname{argmax}_{x'} h(x') - c(x, x')$

rational *utility = prediction* *cost (e.g., norm)*



rational ⇒ most cost-effective

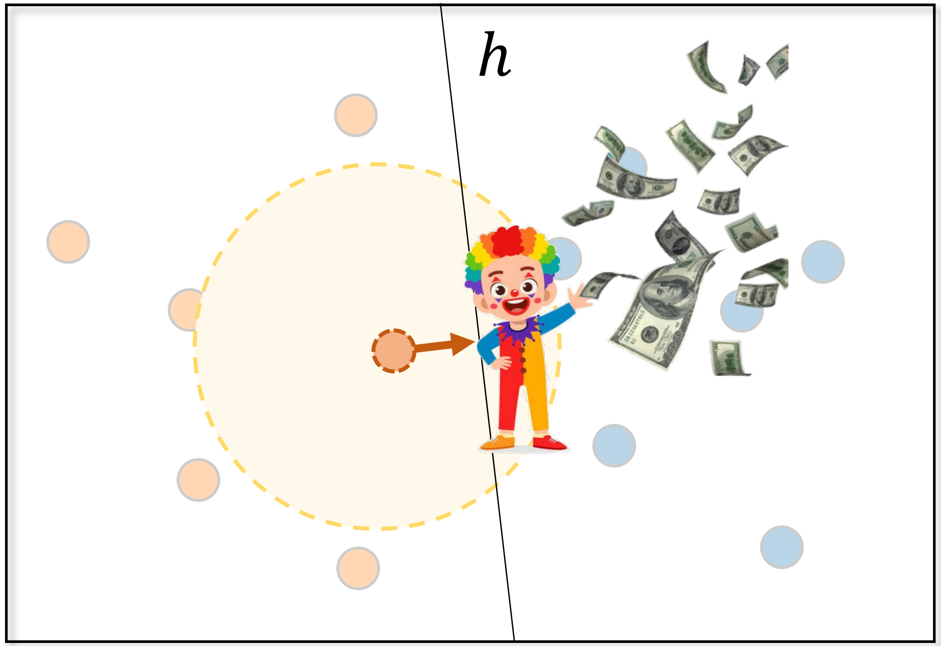
⇒ move on decision boundary

strategic classification:

train: $\underset{h}{\operatorname{argmin}}\ \mathbb{E}[\mathbb{1}\{y \neq h(x)\}]$

test: $h(\Delta_h(x)) = \hat{y} \not\approx y$

response: $\Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ h(x') - c(x, x')$



goal: learning that is robust to strategic "gaming" behavior

strategic classification:

train: $\underset{h}{\operatorname{argmin}}\ \mathbb{E}[\mathbb{1}\{y \neq h(x)\}]$

test: $h(\Delta_h(x)) = \hat{y} \not\approx y$

response: $\Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ h(x') - c(x, x')$

Goodhart's law:

"If a measure becomes the public's goal, it is no longer a good measure."

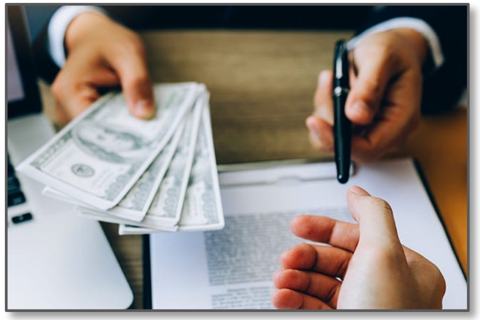
strategic classification:

train: $$\underset{h}{\operatorname{argmin}}\ \mathbb{E}[\mathbb{1}\{y \neq h(x)\}]$$

test: $$h(\Delta_h(x)) = \hat{y} \not\approx y$$

response: $$\Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ h(x') - c(x, x')$$

- **Common examples:**



...

strategic
classificat

train:

test:

respons $) - c(x,x')$

- SC is great, by frustrating
- **Culprit** – lots (and lots) of **assumptions**:
  - *outcomes are binary*
  - *users always want positive outcomes*
  - *costs are fixed, uniform, and known to all*
  - *classifier is made public*
  - *modifying x does not affect y*
  - *changes to x are real (no mis-reporting)*
  - *user actions = modify features*
  - *users are rational (best-respond)*
  - *users respond independently*
  - *input data are `clean' (=unmodified)*
  - *playing order is fixed*
  - *only single playing round*
  - *system cares only for accuracy*
  - ….
- ongoing community effort to relax, extend, scrutinize, and generalize

Holy grail: a realistically practical, well-understood, plug-n-play framework for strategic learning

strategic
classification:

train: $\underset{h}{\operatorname{argmin}}\ \mathbb{E}[\mathbb{1}\{y \neq h(x)\}]$

discrepant

test: $h(\Delta_h(x)) = \hat{y} \approx y$

response: $\Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ h(x') - c(x, x')$

- standard setup has **lots** (and lots) **of assumptions:** (implicit/explicit)
  - *modifying x does not affect y*
  - *outcomes are binary*
  - *input data are `clean' (=unmodified)*
  - *changes to x are real (no mis-reporting)*
  - *users always want positive outcomes*
  - *costs are fixed, uniform, and known to all*
  - *classifier is made public*
  - *user actions = modify features*
  - *users are rational (best-respond)*
  - *users respond independently*
  - *playing order is fixed*
  - *only single playing round*
  - *system cares only for accuracy*
- ongoing community effort to relax, extend, scrutinize, and generalize

strategic classification:

train: $\underset{h}{\operatorname{argmin}}\ \mathbb{E}[\mathbb{1}\{y \neq h(\Delta_h(x))\}]$

consistent

test: $h(\Delta_h(x)) = \hat{y} \approx y$

response: $\Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ h(x') - c(x, x')$

key point:

far from trivial! minor change ⇒ major implications

- standard setup has **lots** (and lots) **of assumptions:** (implicit/explicit)
  - *modifying x does not affect y*
  - *outcomes are binary*
  - *input data are `clean' (=unmodified)*
  - *changes to x are real (no mis-reporting)*
  - *users always want positive outcomes*
  - *costs are fixed, uniform, and known to all*
  - *classifier is made public*
  - *user actions = modify features*
  - *users are rational (best-respond)*
  - *users respond independently*
  - *playing order is fixed*
  - *only single playing round*
  - *system cares only for accuracy*

ongoing community effort to relax, extend, scrutinize, and generalize strategic

classification as a Stackelberg game: [HMPW '16]

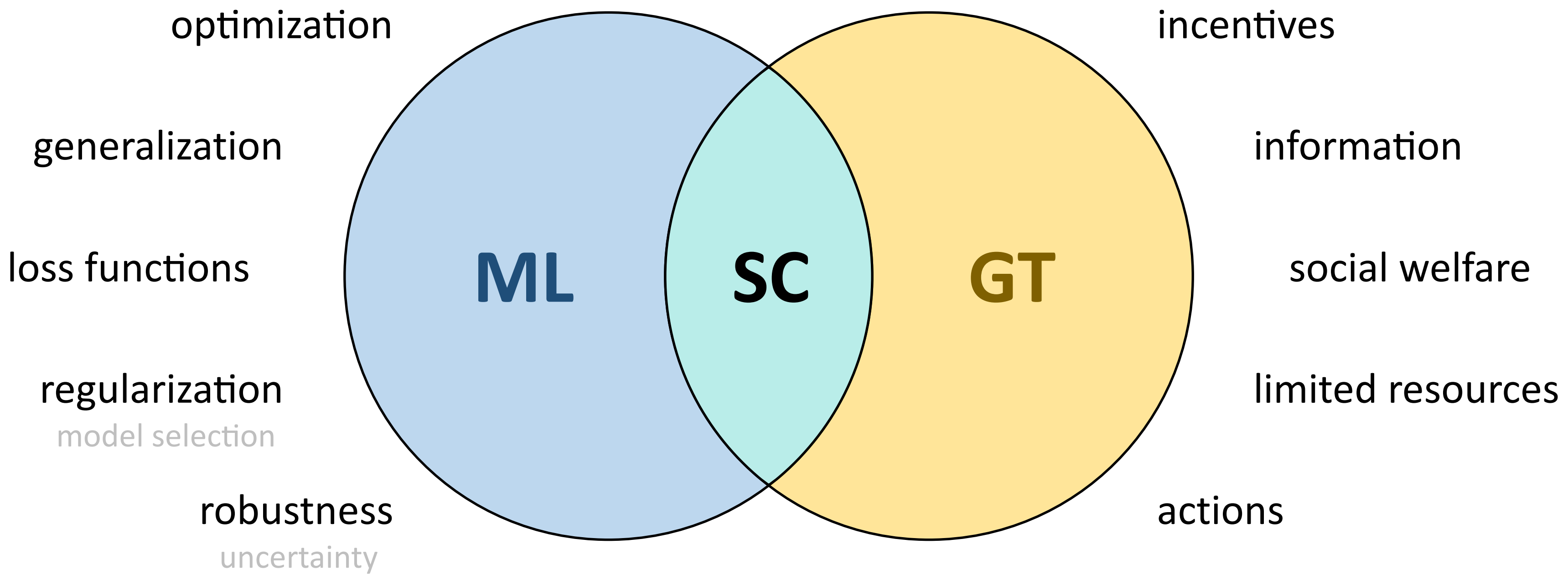
- Players: [1st] Learner [2nd] Users (dist.)
- Actions: classifier h modify $x \mapsto x^h$
- Payoffs: $\mathbb{E}[\mathbb{1}\{h(x^h) = y\}]$ $\mathbb{E}[\mathbb{1}\{h(x^h) = 1\}]$

- Best response: $x^h = \Delta_h(x) = \underset{x'}{\text{argmax}}\ h(x') - c(x, x')$

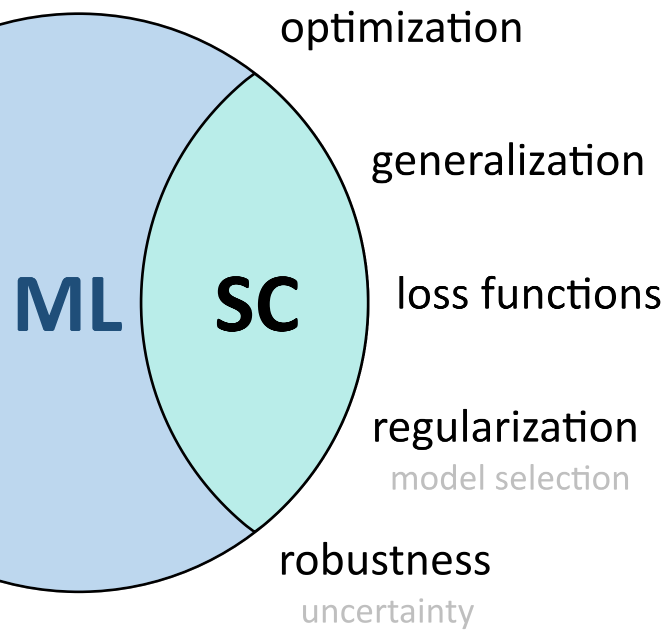
- **Solve equilibrium $\Leftrightarrow$ solve learning**
- Holds in idealized setting; trickier as becomes more realistic (finite data, partial information, weaker assumptions, …)
- **Still**: SC = fundamental ML task + basic economic questions

play order is **crucial modeling choice** – choose with care! [NGTR'21, [ZMSJ'21]

strategic classification as an **interface**
between **machine learning** and **game theory**:

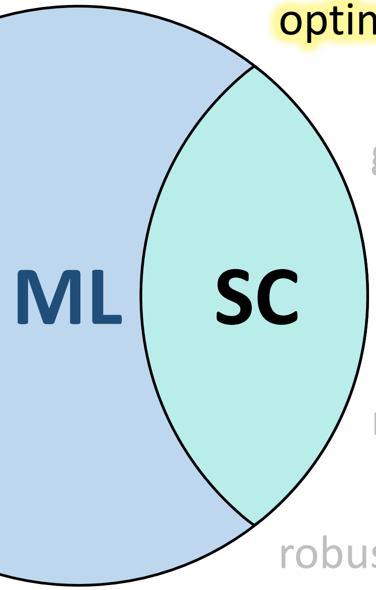


revisit old questions + tackle new ones



Learning aspects

of strategic classification



optimization

generalization

loss functions

regularization

model selection

robustness

uncertainty

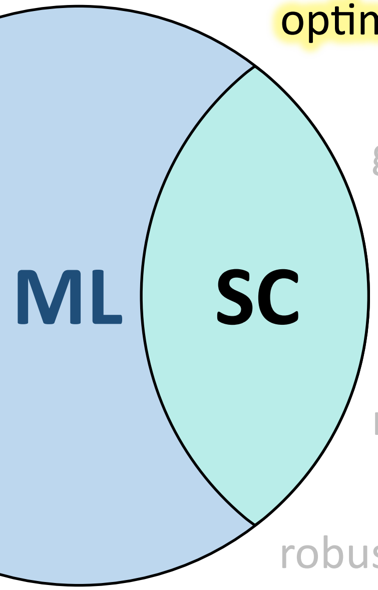
learning objective:

$$\underset{h}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h\left(\tilde{\Delta}_h(x_i)\right)\right)$$

$$\text{s.t. } \Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ h(x') - c(x, x')$$

nasty nested min-argmax problem!

ask: how to optimize objective?



optimization

generalization

loss functions

regularization

model selection

robustness

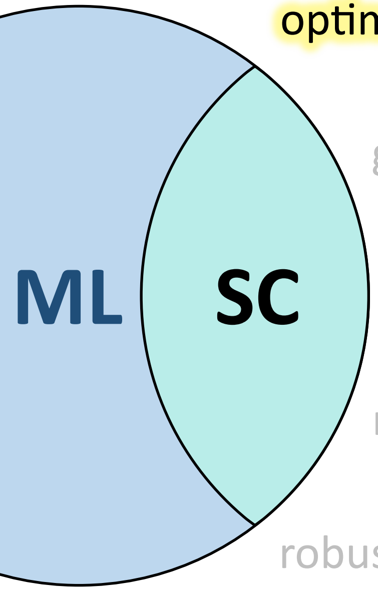
uncertainty

learning objective:

$$\underset{h}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h(\tilde{\Delta}_h(x_i))\right)$$

s.t. $\tilde{\Delta}_h(x) \approx \Delta_h(x)$ and is *differentiable*

ask: how to optimize objective?



optimization

generalization

loss functions

regularization

model selection

robustness

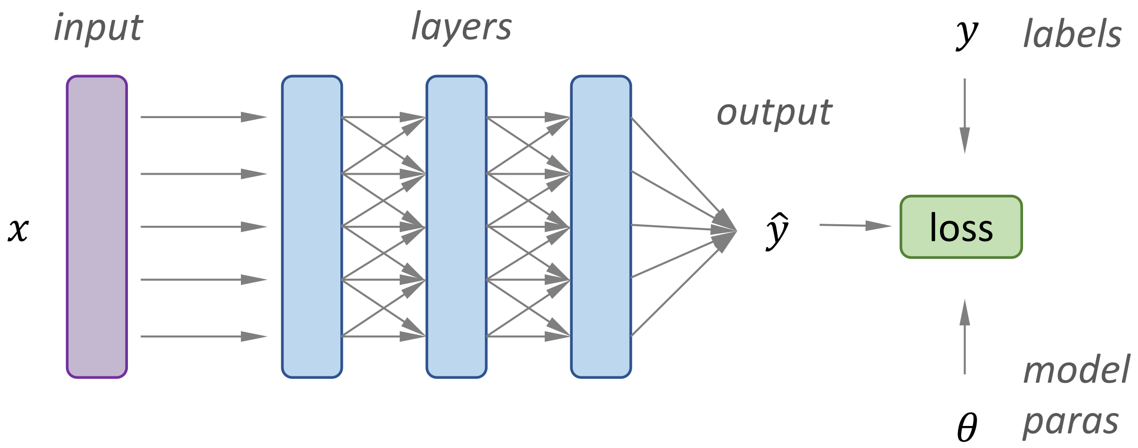
uncertainty

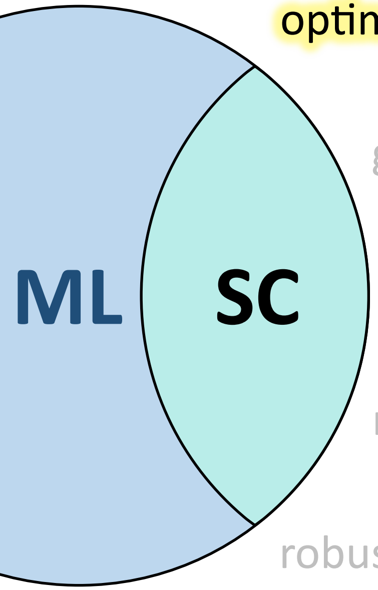
learning objective:

$$\underset{h}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h\left(\tilde{\Delta}_h(x_i)\right)\right)$$

s.t. $\tilde{\Delta}_h(x) \approx \Delta_h(x)$ and is *differentiable*

ask: how to optimize objective?





optimization

generalization

loss functions

regularization
model selection

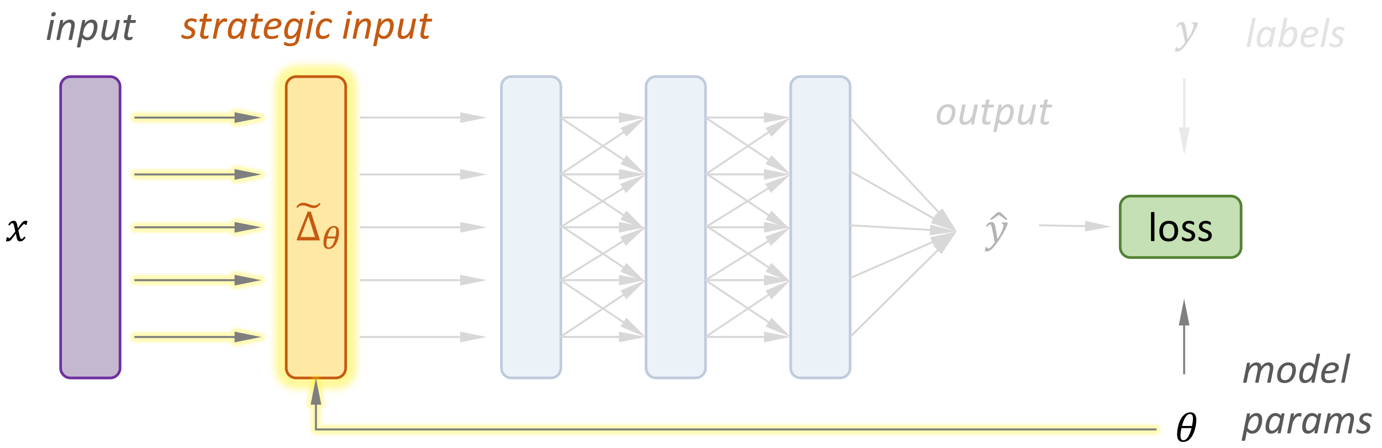
robustness
uncertainty

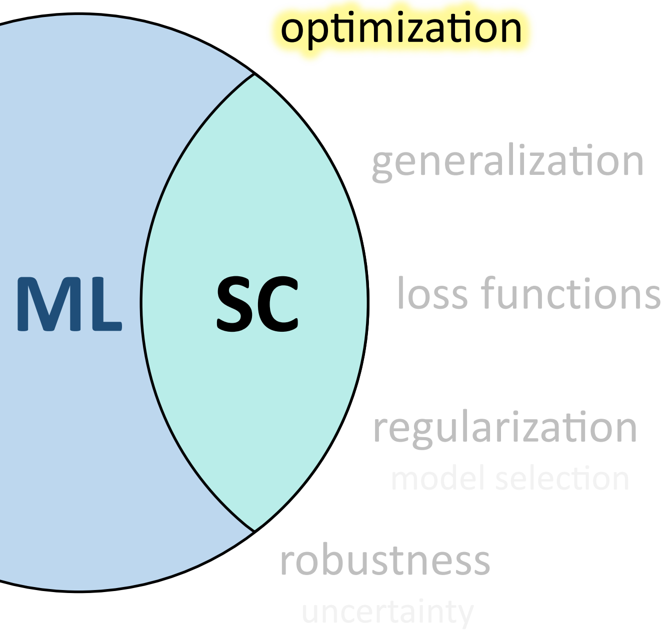
ask: how to optimize objective?

learning objective:

$$\underset{h}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h\left(\tilde{\Delta}_h(x_i)\right)\right)$$

s.t. $\tilde{\Delta}_h(x) \approx \Delta_h(x)$ and is *differentiable*

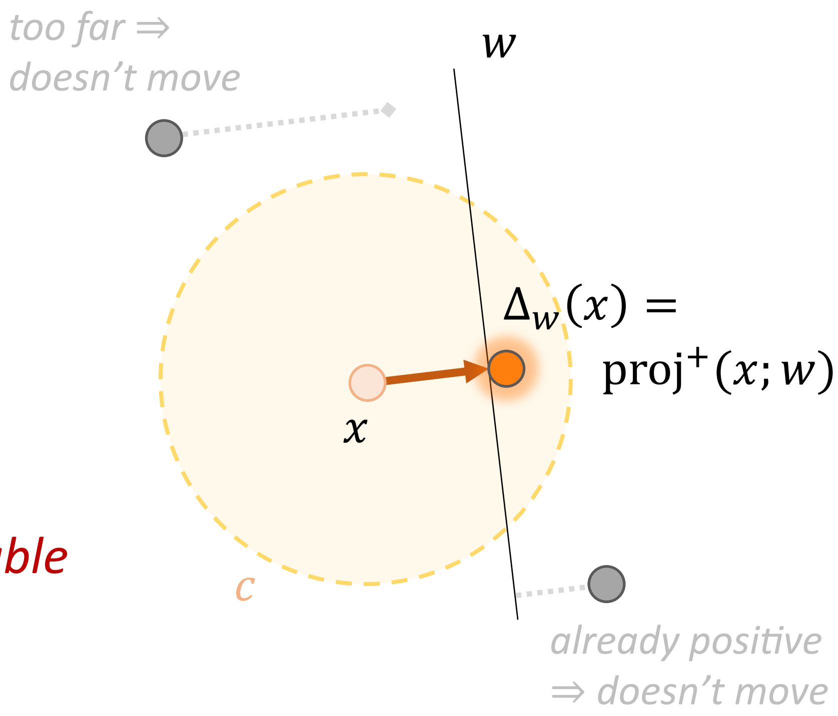




learning objective:

$$\operatorname*{argmin}_{h} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h(\tilde{\Delta}_h(x_i))\right)$$

s.t. $\tilde{\Delta}_h(x) \approx \Delta_h(x)$ and is *differentiable*

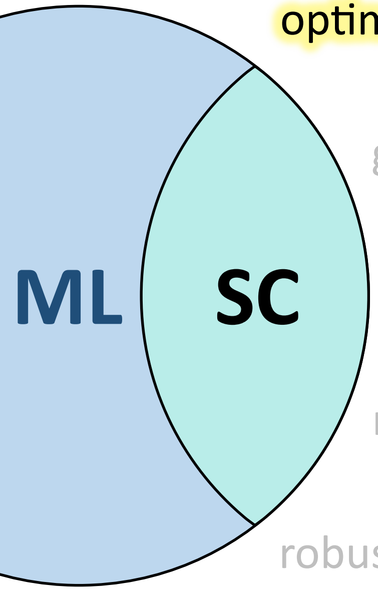


- For **common case** where:
  - $h(x) = w^\top x$
  - $c(x, x') = \|x' - x\|_2$ (or squared, or PSD)
- Admits **simple closed-form solution**:

$$\Delta_w(x) = \begin{cases} x & w^\top x \geq 0 \;\text{ or }\; \mathrm{dist}(x; w) > 2 \\ \mathrm{proj}^+(x; w) & \text{o.w.} \end{cases}$$

$$= x - \min\left\{0, \frac{w^\top x + b}{\|w\|_2^2}\right\}$$ *differentiable!*

- Just replace hard-if with soft-if (e.g., sigmoid)



optimization

generalization

loss functions

regularization

model selection

robustness

uncertainty

learning objective:

$$\underset{h}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h(\tilde{\Delta}_h(x_i))\right)$$

s.t. $\tilde{\Delta}_h(x) \approx \Delta_h(x)$ and is *differentiable*

- **Otherwise, when**:
  - ➢ $h(x) = w^\top \phi(x) + \psi(x)$
    (for some non-linear ϕ, ψ)
  - ➢ Δ applies to $z = \phi(x)$
  - ➢ c is convex (in z)
- Then can use **`plato:`** [LR ICML21]
  implements Δ as **concave optimization layer** [AABBDK'19]
- **Code**: https://plato.codes/

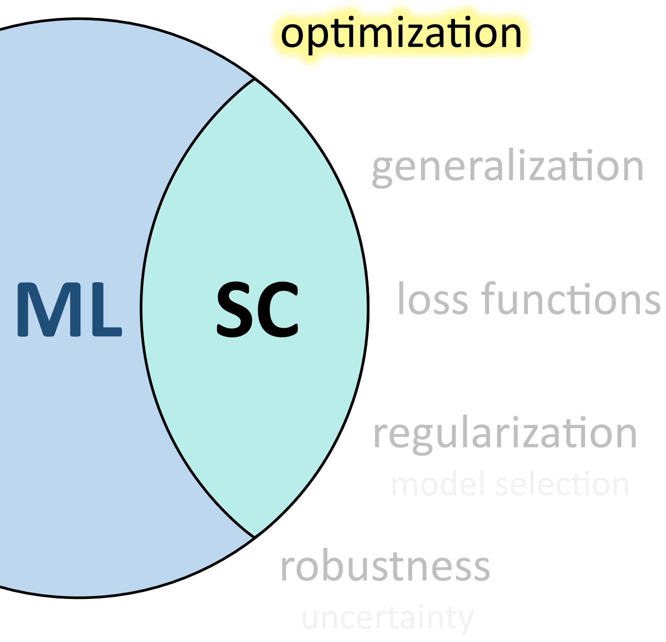
e.g., if Δ is LP:

$$z^* = \operatorname{argmax}_z x^\top A z \quad \text{s.t. } Bz \leq 0$$

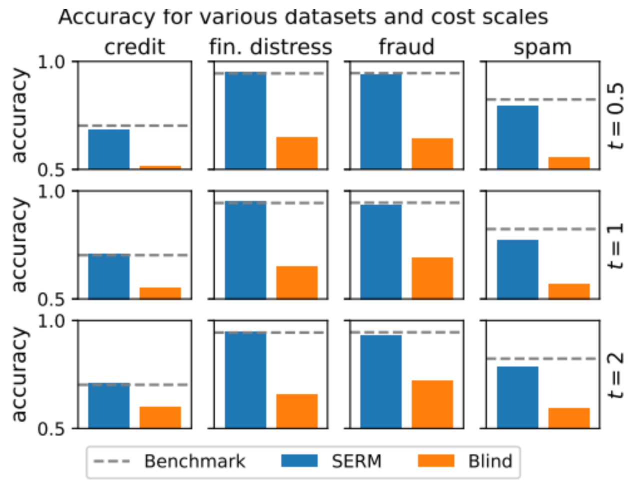
$$\updownarrow$$

$$z^* = h(x; \underbrace{A, B}_{=\theta})$$

can differentiate!



varied costs:
(concave in x, z)



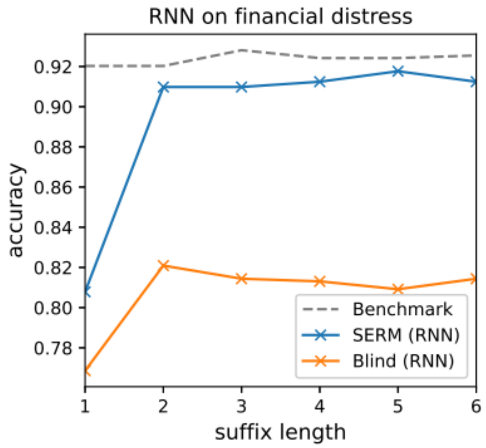
learning objective:

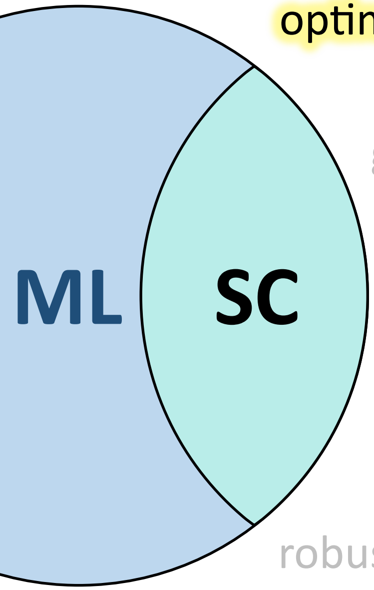
$$\underset{h}{\operatorname{argmin}} \frac{1}{m}\sum_{i=1}^{m} \ell\left(y_i, h\left(\widetilde{\Delta}_h(x_i)\right)\right)$$

s.t. $\widetilde{\Delta}_h(x) \approx \Delta_h(x)$ and is *differentiable*

- **Otherwise, when**:
  - $h(x) = w^\top \phi(x) + \psi(x)$
    (for some non-linear ϕ, ψ)
  - Δ applies to $z = \phi(x)$
  - c is convex (in z)
- Then can use **`plato:`** [LR ICML21]
  implements Δ as **concave optimization layer** [AABBDK'19]
- **Code**: https://plato.codes/

flexible models:
(concave in z)
e.g., certain RNNs





optimization

generalization

loss functions

regularization
model selection

robustness
uncertainty

varied costs:
(concave in x, z)

learning objective:

$$\underset{h}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h(\widetilde{\Delta}_h(x_i))\right)$$

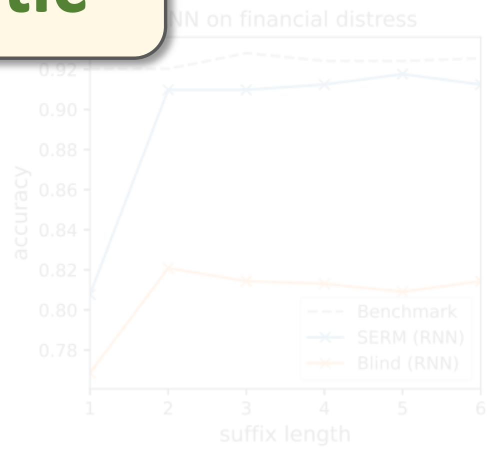
current state of affairs:

s.t. $\widetilde{\Delta}_h(x) \approx \Delta_h(x)$ and is *differentiable*

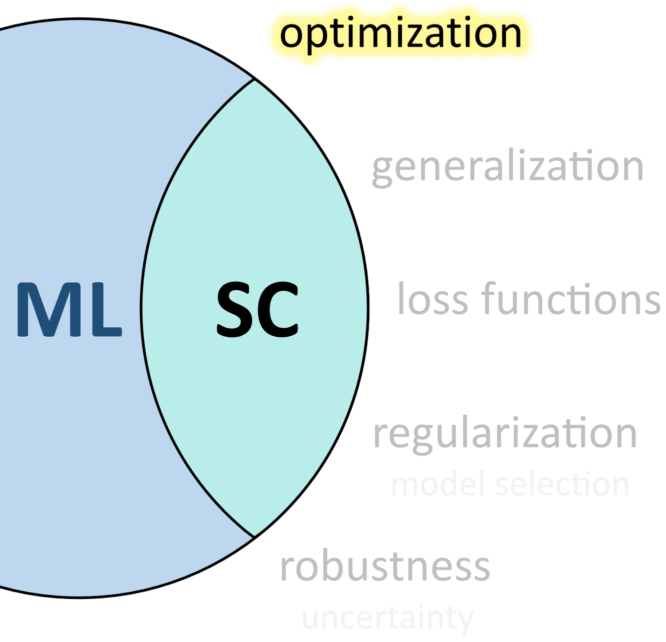
experiments = semi-syntetic

- **Otherwise, when:**
  - ➢ $h(x) = w^\top \phi(x) + \psi(x)$
    (for some non-linear ϕ, ψ)
  - ➢ Δ applies to $z = \phi(x)$
  - ➢ c is convex (in z)
- Then can use **plato:** [LR ICML21]
  implements Δ as **concave optimization layer** [AABBDK'19]
- **Code**: https://plato.codes/

flexible models:
(concave in z)
e.g., certain RNNs



LR ICML21



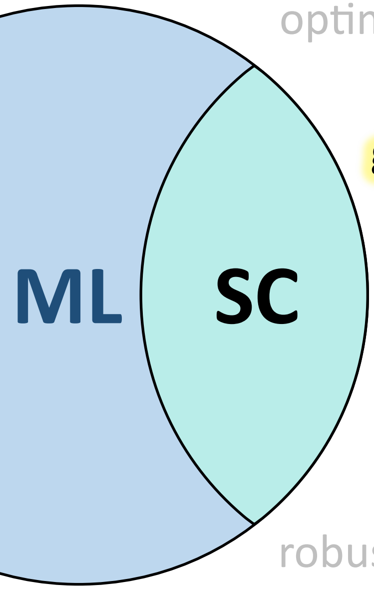
learning objective:

$$\underset{h}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h(\tilde{\Delta}_h(x_i))\right)$$

s.t. $\tilde{\Delta}_h(x) \approx \Delta_h(x)$ and is *differentiable*

- **Otherwise – uncharted territory**
- **Idea**: borrow methods from **adversarial learning** literature (e.g., FGSM [GSS'15] or PGD [MMSTV'18])
- Essentially, optimize objective by alternating between:
  - fixing features x^h and updating θ
  - fixing parameters θ and updating x^h
- Technically possible – but hasn't been done yet in strategic learning

➢ More on strategic ↔ adversarial connection to follow!



optimization

generalization

loss functions

regularization

model selection

robustness

uncertainty

empirical loss:

$$\operatorname*{argmin}_{h} \frac{1}{m}\sum_{i=1}^{m} \ell\left(y_i, h(\Delta_h(x_i))\right)$$

generalization

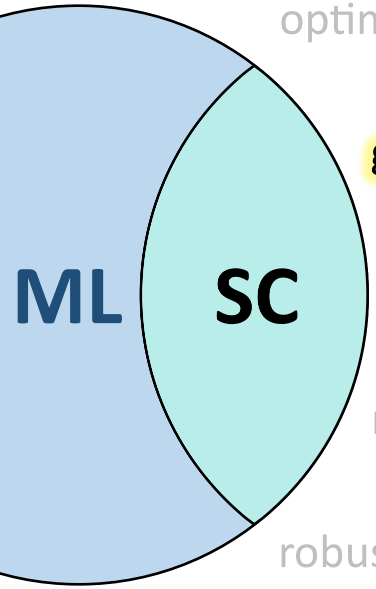
expected loss:

$$\operatorname*{argmin}_{h} \mathbb{E}\left[\ell\left(y, h(\Delta_h(x))\right)\right]$$

ask: how does behavior affect generalization?

empirical loss:

$$\underset{h}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h\left(\Delta_h(x_i)\right)\right)$$



optimization

generalization

loss functions

regularization

model selection

robustness

uncertainty

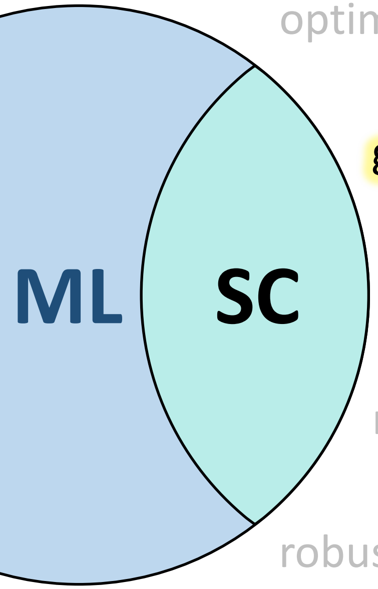
- **Q –** will strategic behavior:
  1. *increase* overfitting?
  2. *reduce* overfitting?
  3. make no difference?

- **Rephrase:** how does behavior affect sample complexity?

ask: how does behavior affect generalization?

empirical loss:

$$\underset{h}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h(\Delta_h(x_i))\right)$$



optimization

generalization

loss functions

regularization

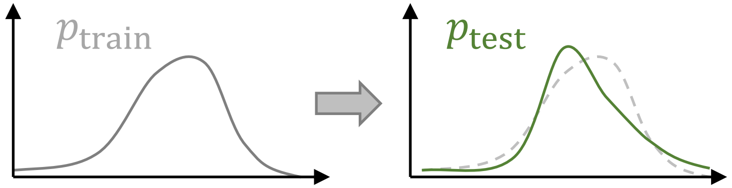
model selection

robustness

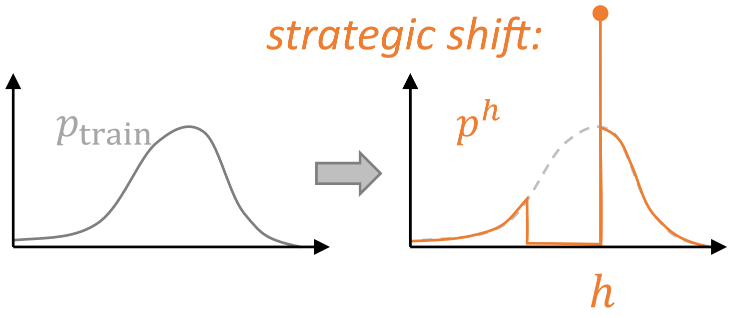
uncertainty

- SC = **model-dependend distribution shift**
- In **typical distriubtion shift**, p_{test} is assumed to be "close" to p_{train} (e.g., in ball)
- Contrarily, in **strategic shift**:
  1. only points in "band" before h move
  2. entire region moves <u>on</u> decision boundary
  3. moving region determined by choice of h

typical shift:



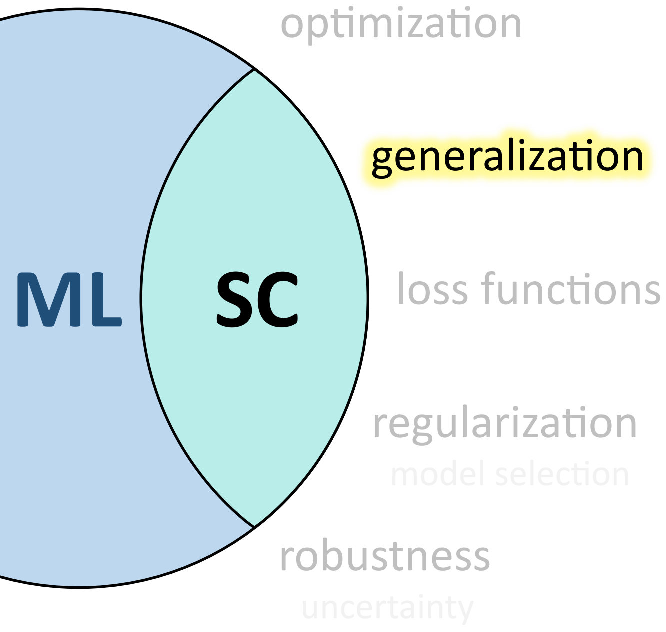
strategic shift:



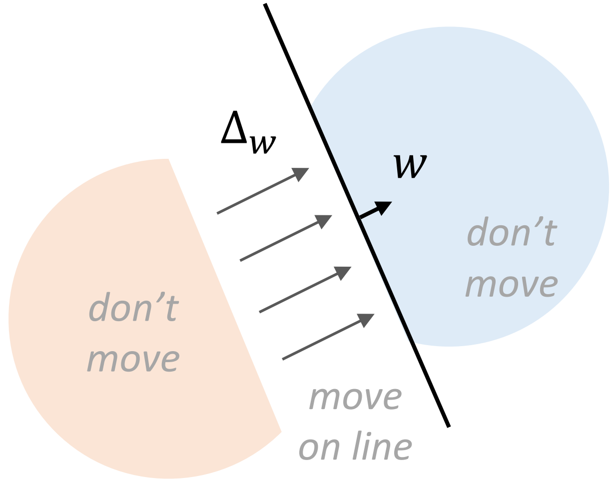
ask: how does behavior affect generalization?

empirical loss:

$$\underset{h}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h(\Delta_h(x_i))\right)$$



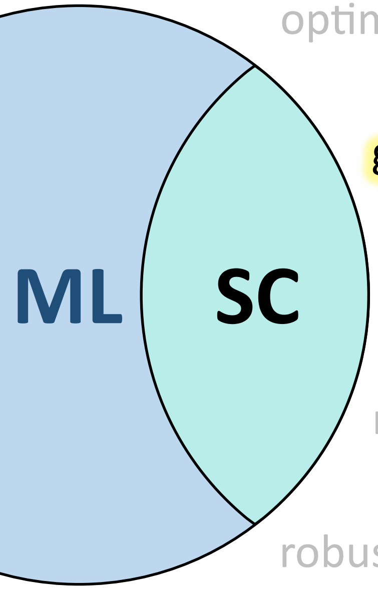
- SC = **model-dependend distribution shift**
- In **typical distriubtion shift**, p_{test} is assumed to be "close" to p_{train} (e.g., in ball)
- Contrarily, in **strategic shift**:
  1. only points in "band" before h move
  2. entire region moves on decision boundary
  3. moving region determined by choice of h



ask: how does behavior affect generalization?

empirical loss:

$$\underset{h}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h(\Delta_h(x_i))\right)$$



optimization

generalization

loss functions

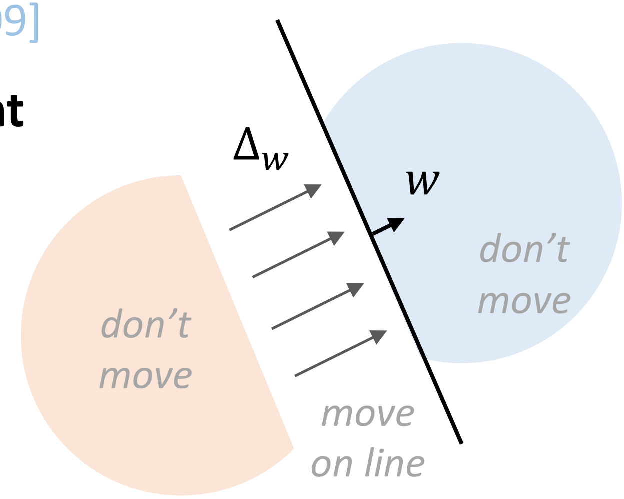
regularization

model selection

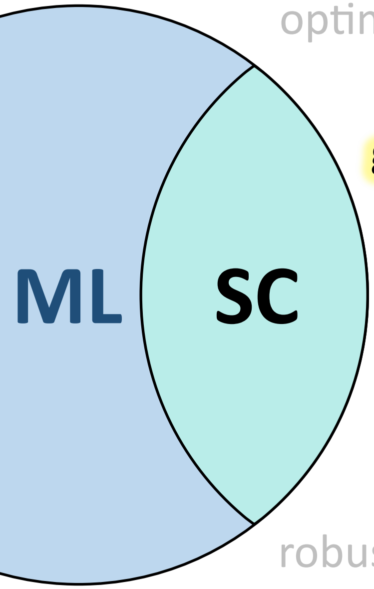
robustness

uncertainty

- Generalization theory typically relies on **discrepency measures** $d(p_{\text{train}}, p_{\text{test}})$ [MMR ICML09]
- $\Rightarrow$ bounds are **shift** (and so **dsitribution**) **dependent**
- Interestingly, strategic shifts admit **distribution-in̲dependent** generalization bounds



ask: how does behavior affect generalization?



optimization

generalization

loss functions

regularization
model selection

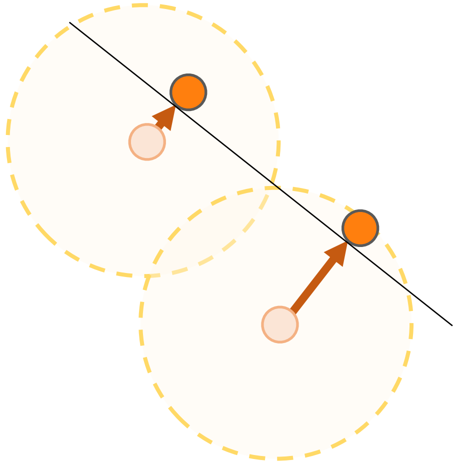
robustness
uncertainty

empirical loss:

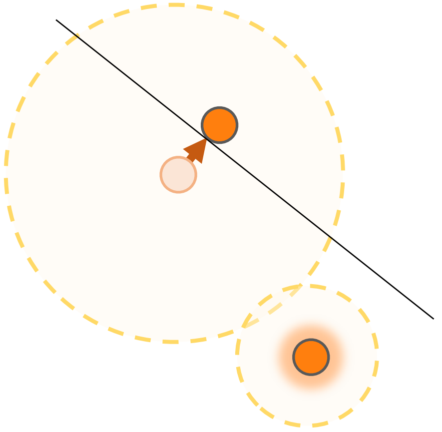
$$\operatorname*{argmin}_{h} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h\left(\Delta_h(x_i)\right)\right)$$

- **Induced class**: $H_\Delta = \{h(\Delta_h(x)) : h \in H\}$
- **Strategic VC**: $SVC(H) = VC(H_\Delta)$
- **Result**: for standard setting, recover non-strategic bounds (almost!)
- But – **cost form matters!** [SVXY'23] show:
  - **instance-invariant costs**:
    $c(x - x') \Rightarrow \; SVC \approx VC$ (for linear h)
    i.e., learning is **not harder**
  - **instance-wise costs:** *=individualized*
    $c_x(x') \Rightarrow$ unlearnable! (in the worst case)
    i.e., learning is **impossible**

instance-invariant:

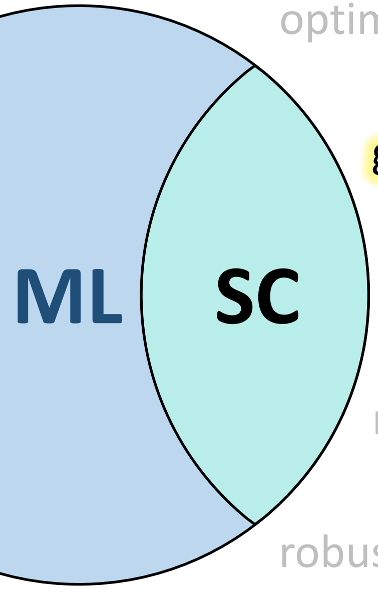


instance-wise:



empirical loss:

$$\operatorname*{argmin}_{h} \frac{1}{m}\sum_{i=1}^{m} \ell\left(y_i, h\left(\Delta_h(x_i)\right)\right)$$



optimization

generalization

loss functions

regularization

model selection

robustness

uncertainty

- Also have data-dependent Rademacher bounds:

$$\mathcal{L}_{0/1} \leq \mathcal{L}_{\text{s-hinge}} + \frac{4r\|\hat{w}\|}{\sqrt{m}} + (1 + 2(\overset{=\max\|x\|}{r} + 2)\|w\|)\sqrt{\frac{2\ln(4\|\hat{w}\|/\delta)}{m}}$$

only difference from non-strategic

- Notice behabior **just adds constant** to scale
- Applies to more general settings (to follow)

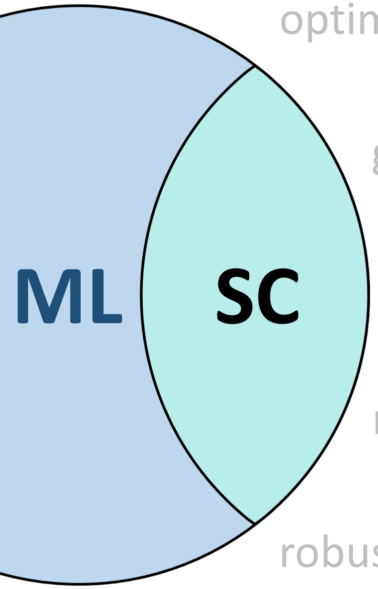
- **Also**: regret analys for online strategic classification *tractable proxy loss:* *(e.g., hinge, log-loss, …)*

$$\operatorname*{argmin}_h \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h\left(\Delta_h(x_i)\right)\right)$$

surrogate

true 0-1 loss:

$$\operatorname*{argmin}_h \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}\{y_i \neq h\left(\Delta_h(x_i)\right)\}$$



optimization

generalization

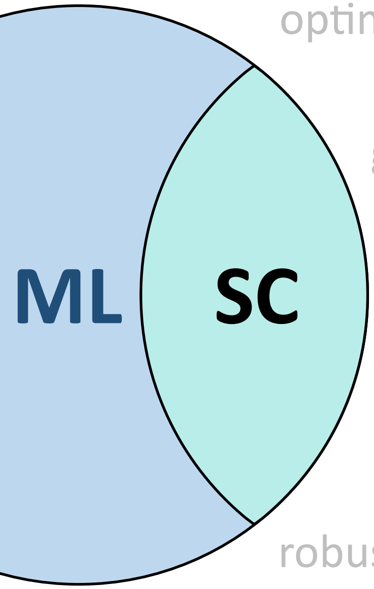
loss functions

regularization
model selection

robustness
uncertainty

ask: can we just use conventional proxies?

standard hinge: $\max\{0,1-yw^\top x\}$



optimization

generalization

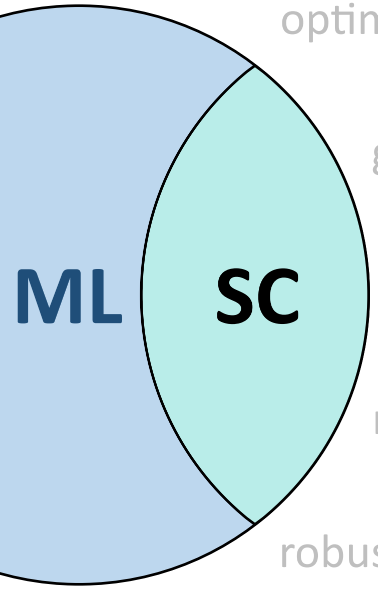
loss functions

regularization
model selection

robustness
uncertainty

ask: can we just use conventional proxies?

standard hinge: $\max\{0, 1 - y w^\top x\}$



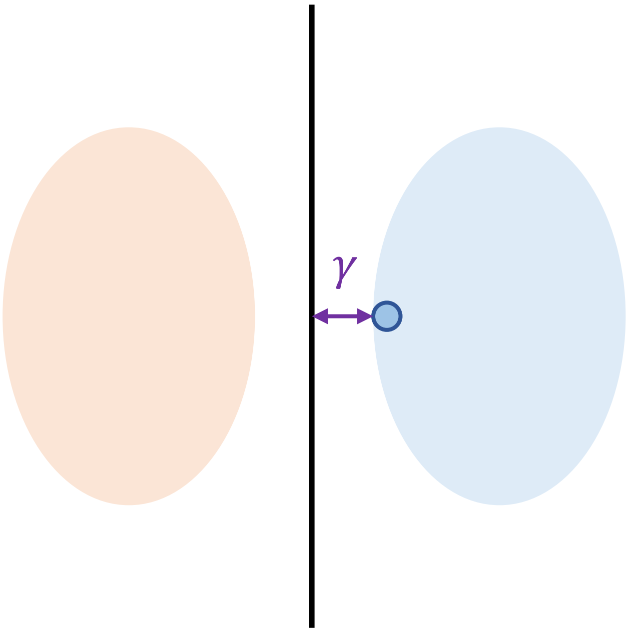
optimization

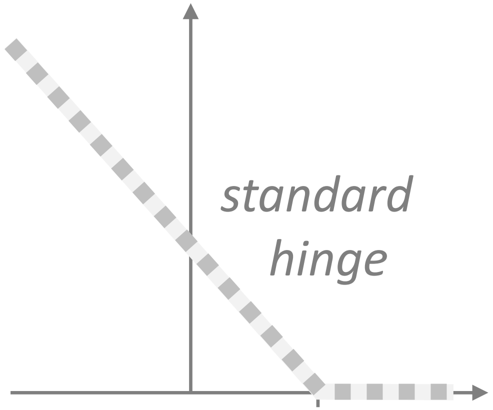
generalization

loss functions

regularization
model selection

robustness
uncertainty





max-margin classifier

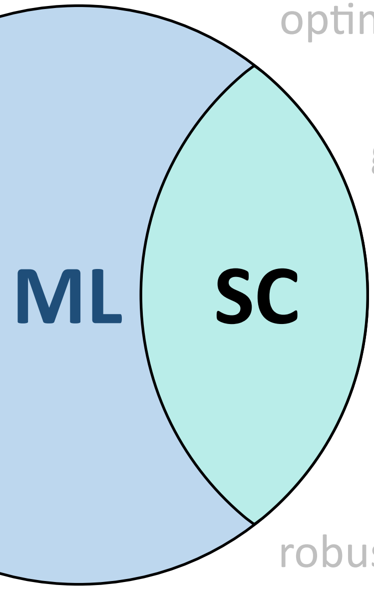
- selection criterion

- good generalization

- tractable

ask: can we just use conventional proxies?

naïve hinge: $\max\{0, 1 - y w^{\top} \Delta_h(x)\}$



optimization

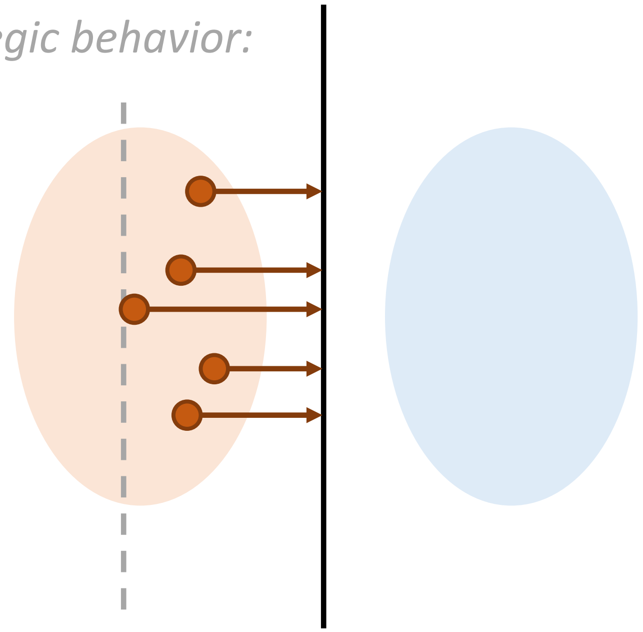
generalization

loss functions

regularization
model selection

robustness
uncertainty

- *strategic behavior:*

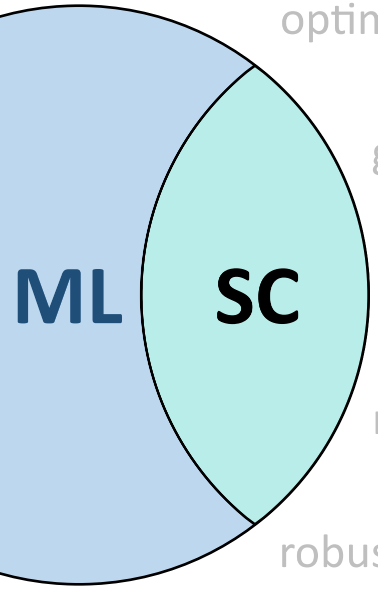


max-margin classifier

- *selection criterion*
- *good generalization*

ask: can we just use conventional proxies?

naïve hinge: $\max\{0,1 - yw^{\top}\Delta_h(x)\}$



optimization

generalization

loss functions

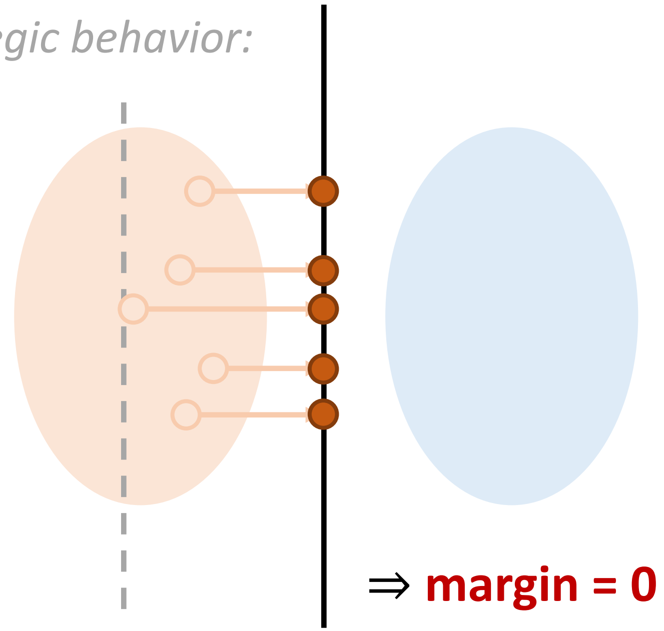
regularization

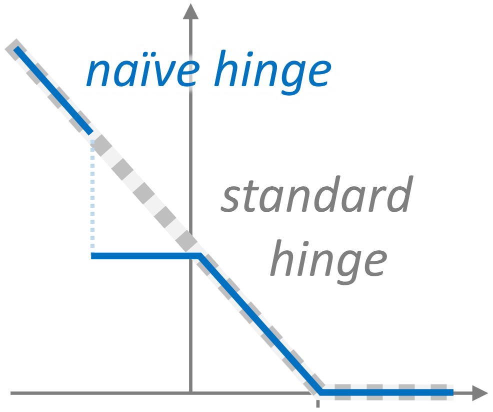
model selection

robustness

uncertainty

- strategic behavior:





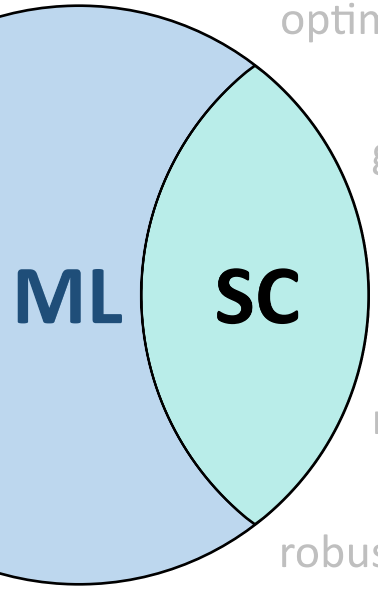
***naive* max-margin classifier**

- vacous criterion

- unclear if generalizes

ask: can we just use conventional proxies?

strategic hinge: $\max\{0, 1 - yw^\top x - 2\|w\|\}$



optimization

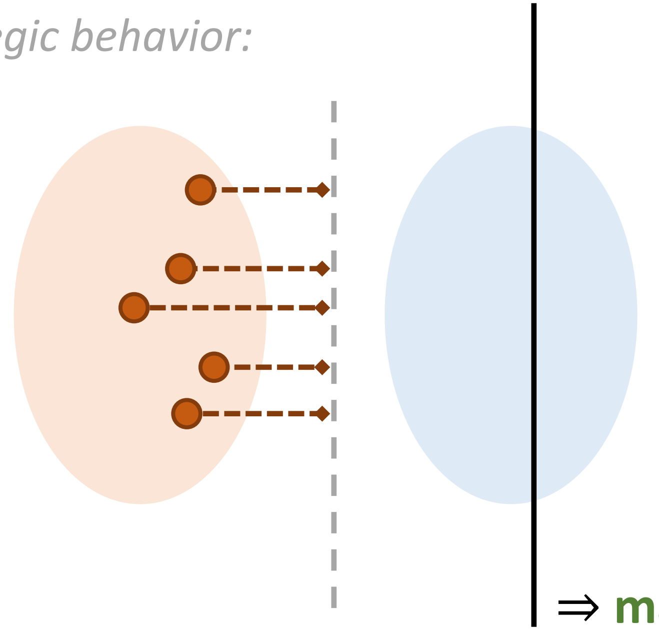
generalization

loss functions

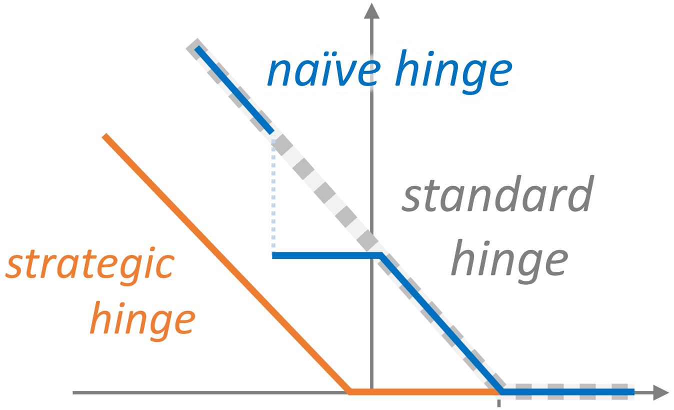
regularization
model selection

robustness
uncertainty

- *strategic behavior:*



$\Rightarrow$ **margin recovered!**

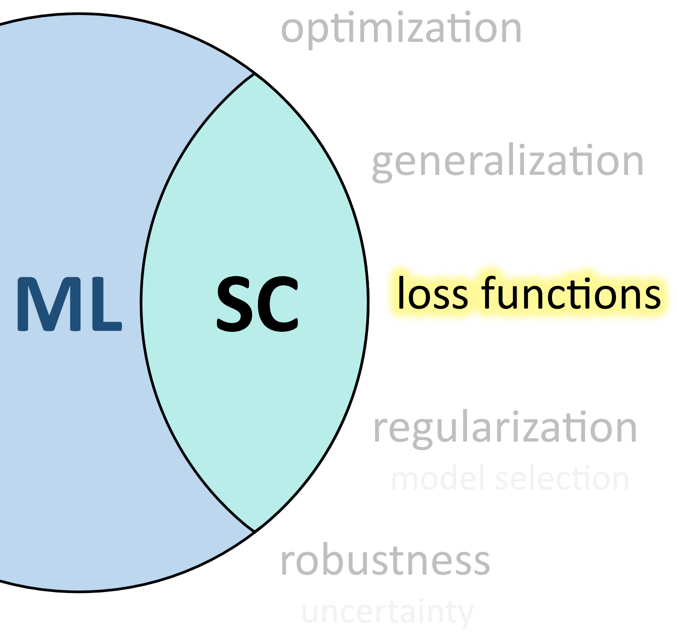


strategic max-margin classifier

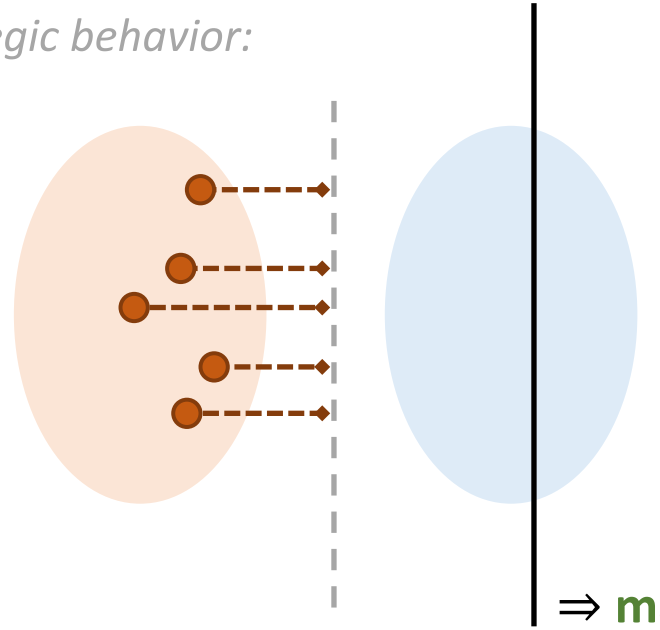
- *regain selection criterion*
- *comparable generalization*
- *reasonably tractable*

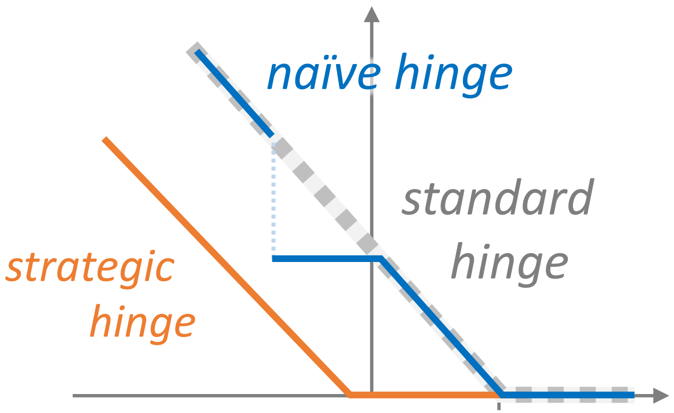
ask: can we just use conventional proxies?

strategic hinge: $\max\{0,1 - yw^{\top}x - 2\|w\|\}$



- *strategic behavior:*



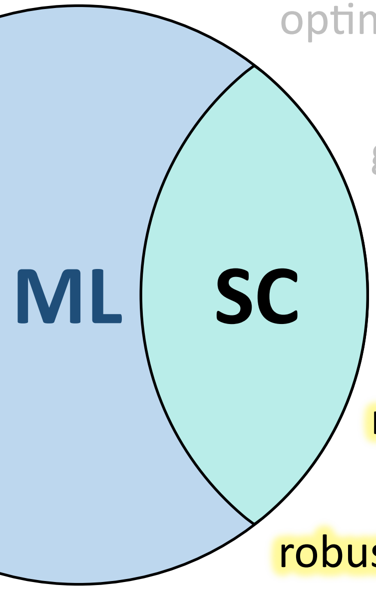


⇒ **margin recovered!**

strategic max-margin classifier

- *regain selection criterion*
- *comparable generalization*
- *reasonably tractable*

conclusion: strategic robustness requires
rethinking fundamental learning concepts



optimization

generalization

loss functions

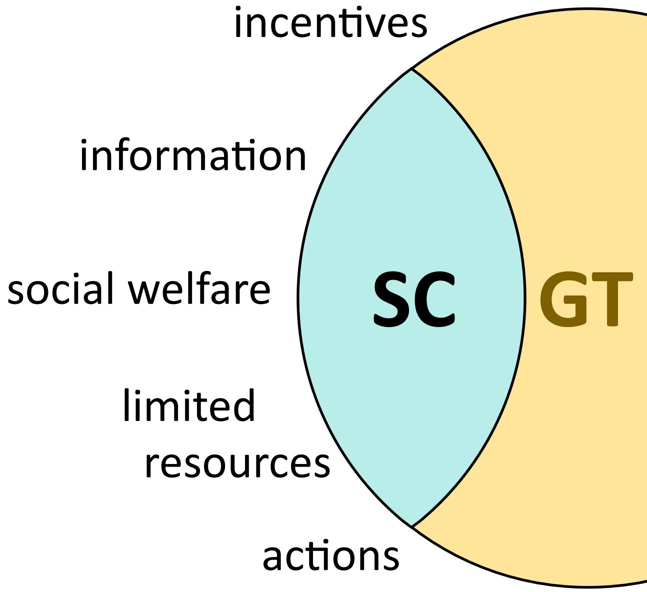
regularization
model selection

robustness
uncertainty

to follow!

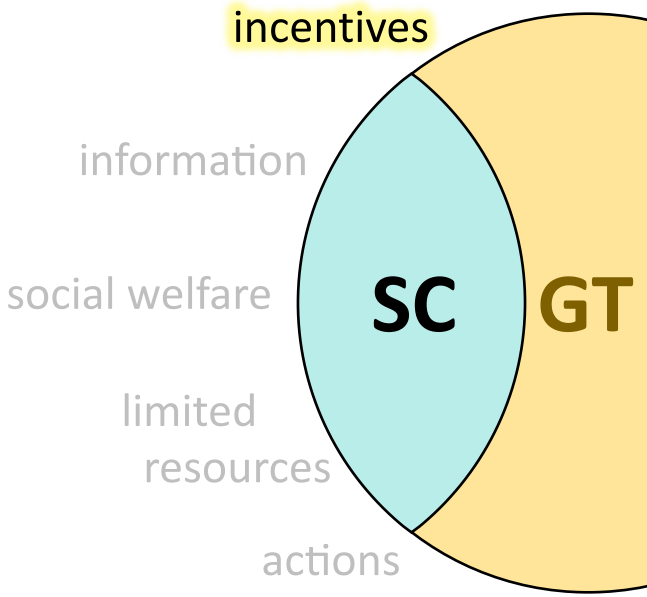
Economic aspects

of strategic classification



standard SC:

$$\Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ h(x') - c(x,x')$$

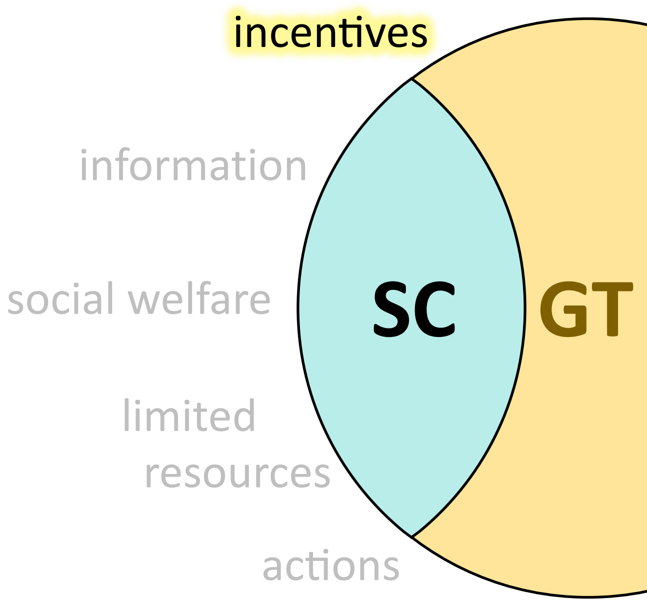


ask: what else could users want?

standard SC:

$$\Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ u(x') - c(x, x')$$

$$= \begin{cases} +1 & \hat{y} = +1 \\ -1 & \hat{y} = -1 \end{cases}$$



ask: what else could users want?

standard SC:

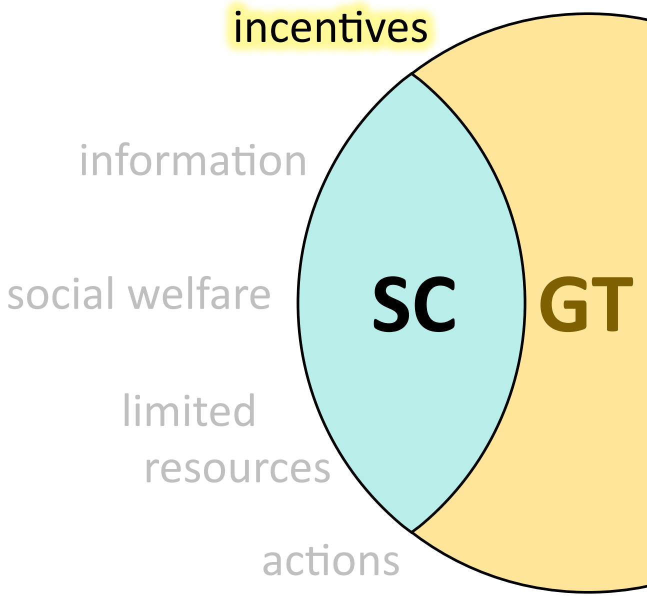
$$\Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ u(x') - c(x, x')$$

$$= \begin{cases} +1 & \hat{y} = +1 \\ -1 & \hat{y} = -1 \end{cases}$$

generalized SC:

$$\boxed{\Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ u(x') - c(x, x')}$$

1. arbitrary **utility function**



ask: what else could users want?

standard SC:

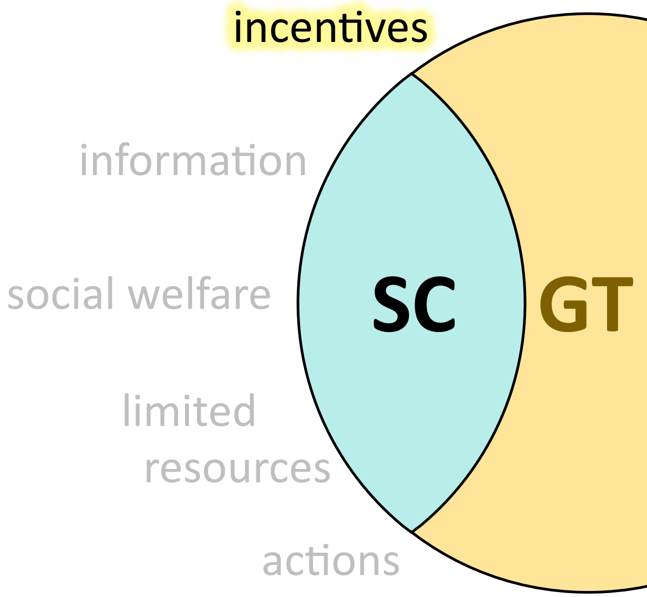
$$\Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ u(x') - c(x, x')$$

$$= \begin{cases} +1 & \hat{y} = +1 \\ -1 & \hat{y} = -1 \end{cases}$$

generalized SC:

$$\boxed{\Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ u(x'; z) - c(x, x')}$$

1. arbitrary **utility function**
2. can depend on **private information**



ask: what else could users want?

standard SC:

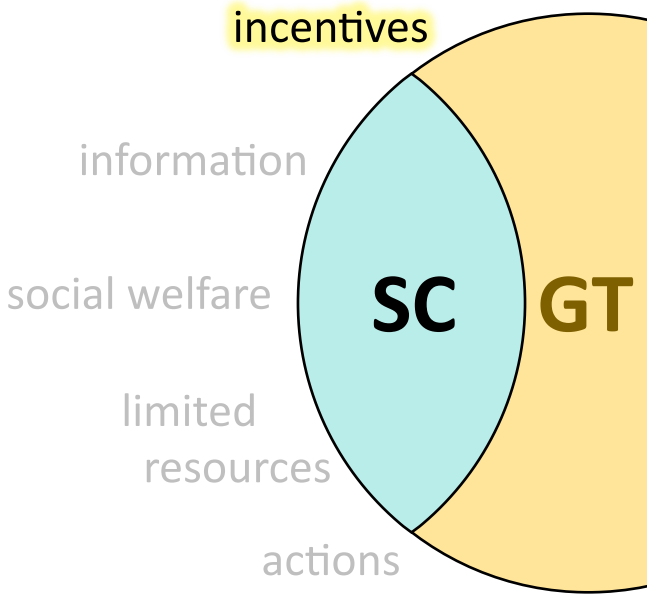
$$\Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ u(x') - c(x, x')$$

$$= \begin{cases} +1 & \hat{y} = +1 \\ -1 & \hat{y} = -1 \end{cases}$$

generalized SC:

$$\boxed{\Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ \tilde{u}(x'; z) - c(x, x')}$$

1. arbitrary **utility function**
2. can depend on **private information**
3. act on **perceived utility** ($\neq$ true utility)



ask: what else could users want?

generalized SC:

$$\Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ \tilde{u}(x'; z) - c(x, x')$$

- **Q**: how to learn?
- **A**: generalize strategic margins and hinge!

➢ **standard hinge:**

$$\max\{0, 1 - y w^\top x\}$$

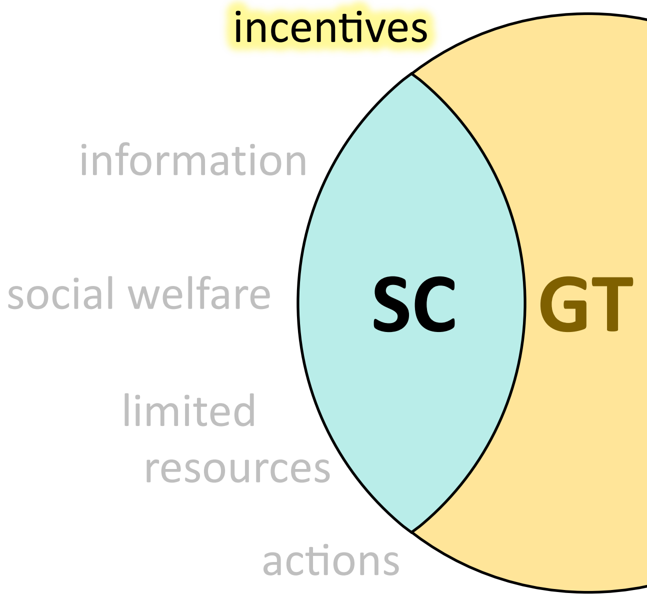
$$= \max\{0, 1 - \underbrace{\operatorname{sign}(y w^\top x)}_{\textit{correctness}} \underbrace{|w^\top x|}_{\textit{distance}}\}$$

➢ **naïve hinge:**

$$\max\left\{0, 1 - \operatorname{sign}\left(y w^\top \Delta_h(x, z)\right) |w^\top \Delta_h(x, z)|\right\}$$

➢ **generalized strategic hinge:** (gs-hinge)

$$\max\left\{0, 1 - \operatorname{sign}\left(y w^\top \Delta_h(x, z)\right) d_\Delta(x, z; w)\right\}$$



ask: what else could users want?

generalized SC:

$$\Delta_h(x) = \operatorname*{argmax}_{x'} \tilde{u}(x'; z) - c(x, x')$$

reinterpretation of "margin":

➢ distance to nearest x' that *flips label*:

$$d_\Delta(x, z; w) = \min_{x'} \frac{\|x - x'\|}{\|w\|}$$

minimal distance between points *(normalized)*

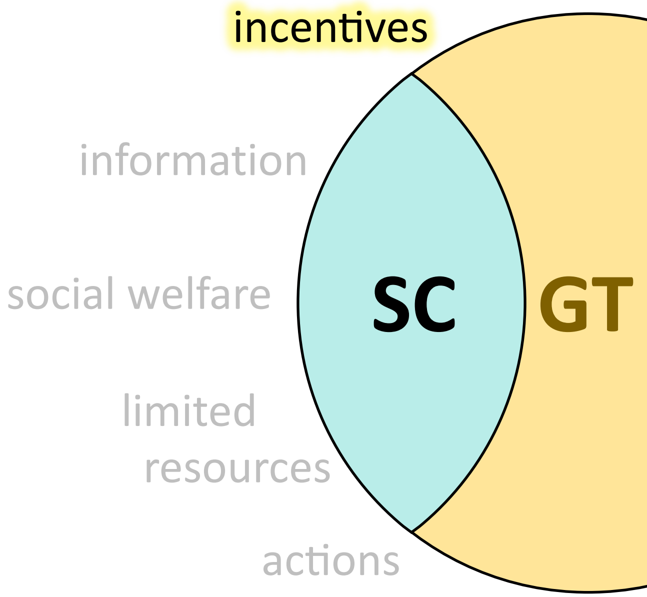
flip label (after movement) s.t. $h(\Delta_h(x,z)) \neq h(\Delta_h(x',z))$

subsumes non-strategic case

- admits convenient tractable form for several known special cases

➢ **generalized strategic hinge:** (gs-hinge)

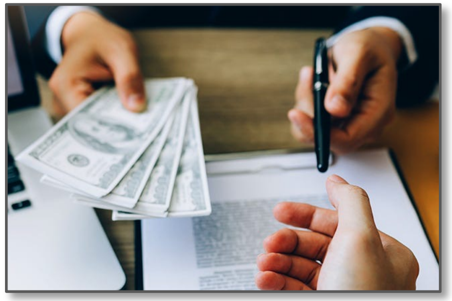
$$\max\{0, 1 - \operatorname{sign}(y w^\top \Delta_h(x,z)) d_\Delta(x, z; w)\}$$



ask: what else could users want?

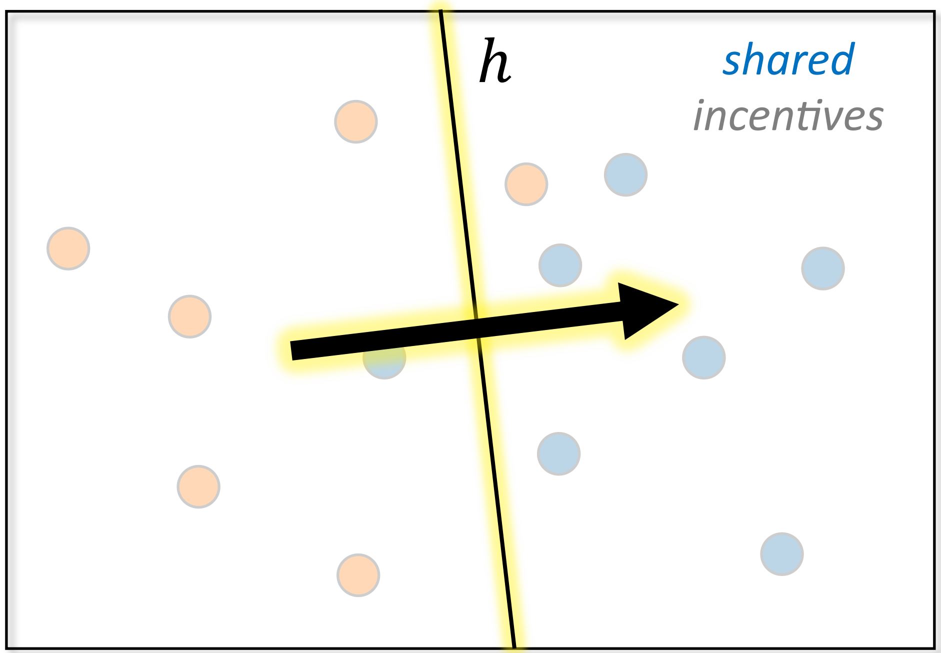
standard SC:

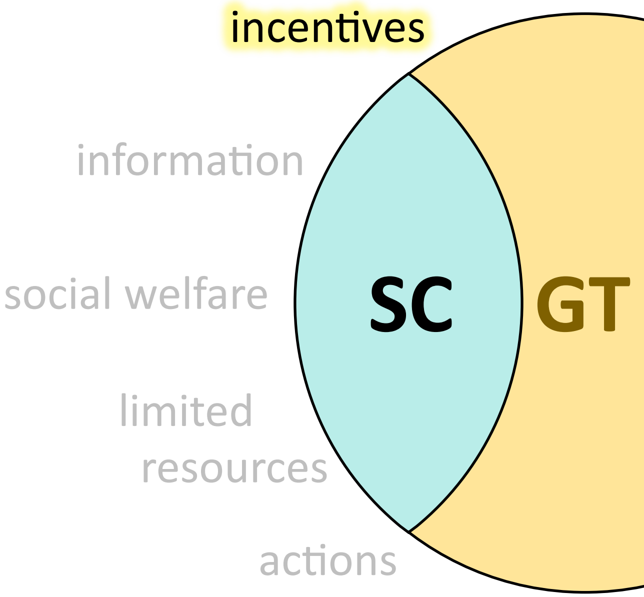
classification ***about*** humans



system wants: *correct predictions*

users want: *positive predictions*

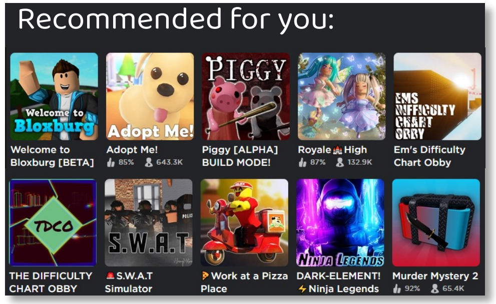




ask: what else could users want?

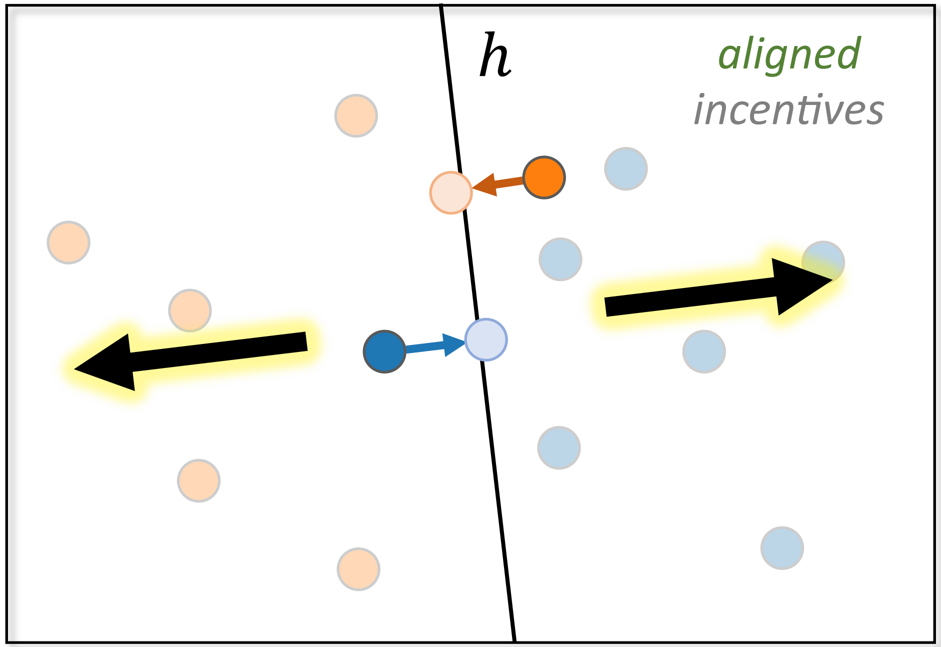
incentive-aligned:

classification ***for*** humans (*as a service*)



system wants: *correct predictions*

users want: *correct predictions*



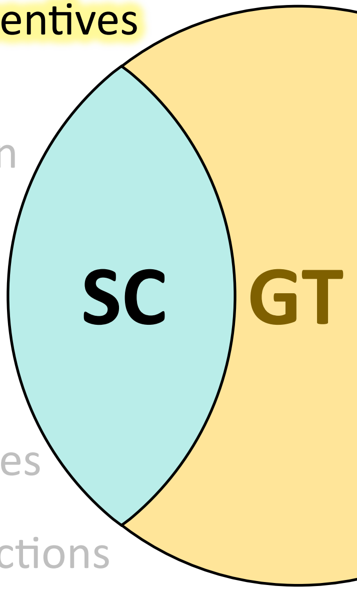
incentives

information

social welfare

limited resources

actions



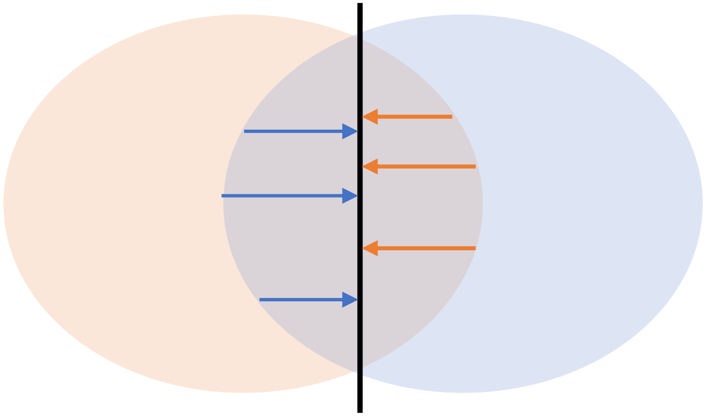
ask: can learning (implicitly) coordinate cooperation?

incentive-aligned:

classification *for* humans (as a *service*)

system wants: *correct predictions*

users want: *correct predictions*



not linearly separable

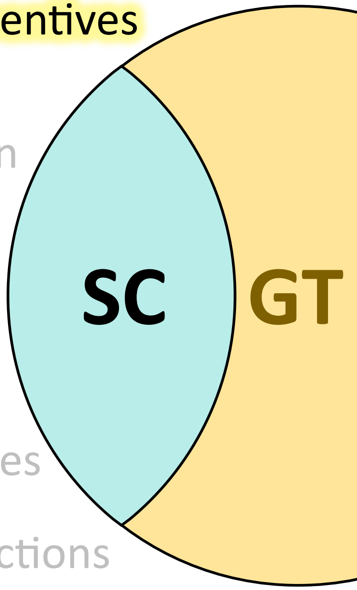
incentives

information

social welfare

limited resources

actions



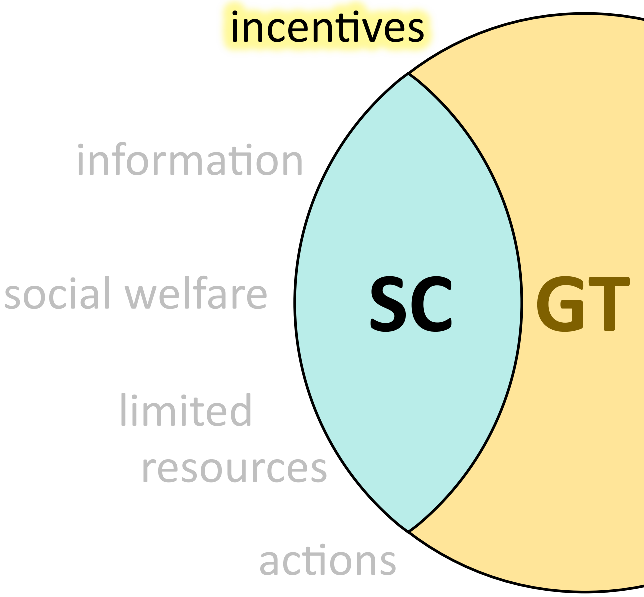
incentive-aligned:

classification ***for*** humans (as a *service*)

system wants: *correct predictions*

users want: *correct predictions*

strategically linearly separable

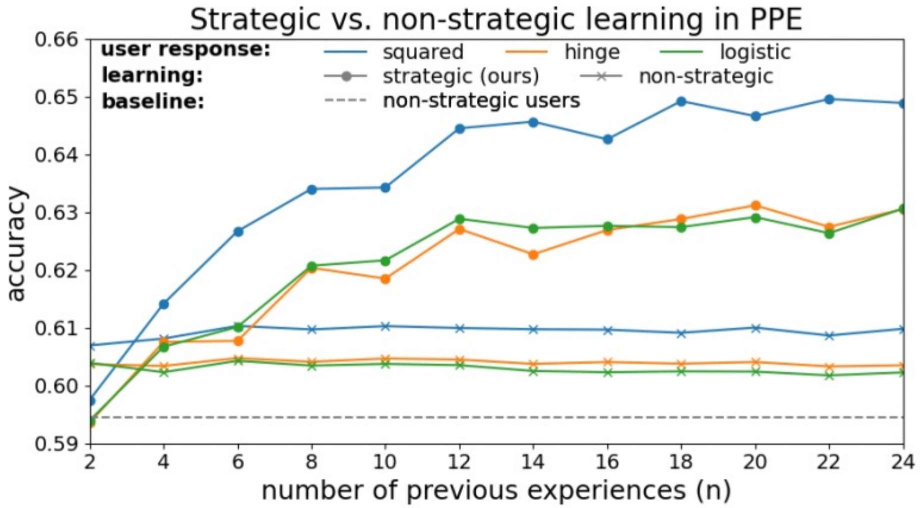


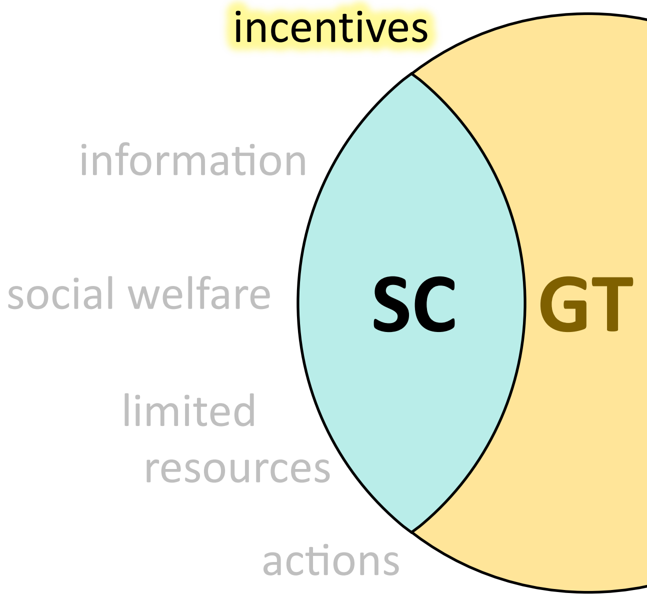
incentive-aligned:

classification ***for*** humans (as a *service*)

system wants: *correct predictions*

users want: *correct predictions*



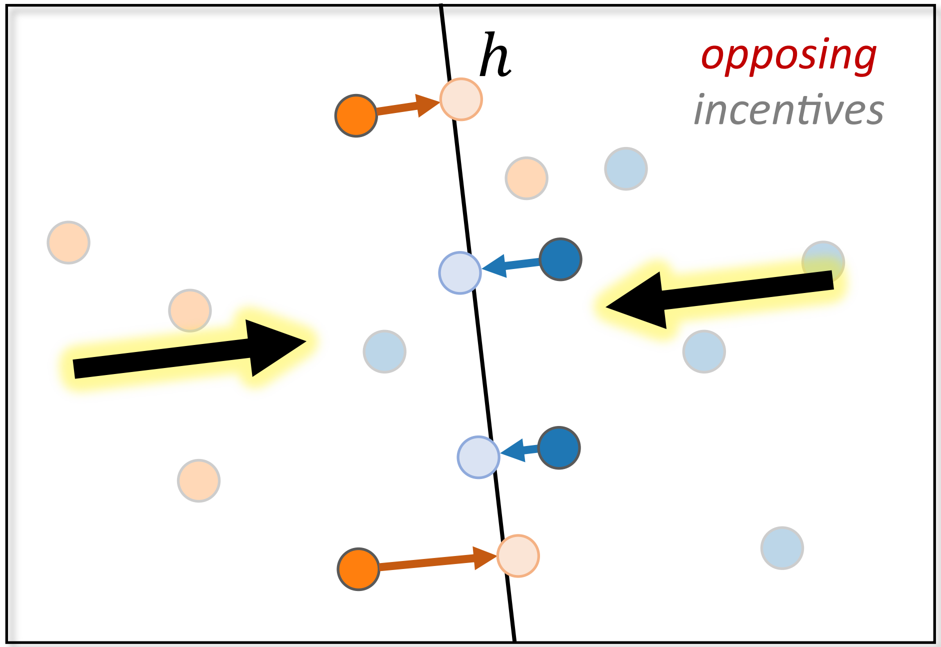


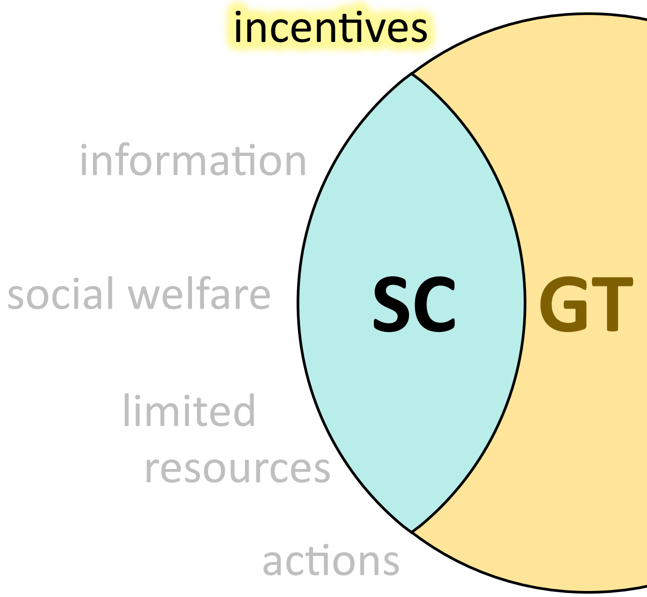
adversarial:

classification ***against*** humans (?)

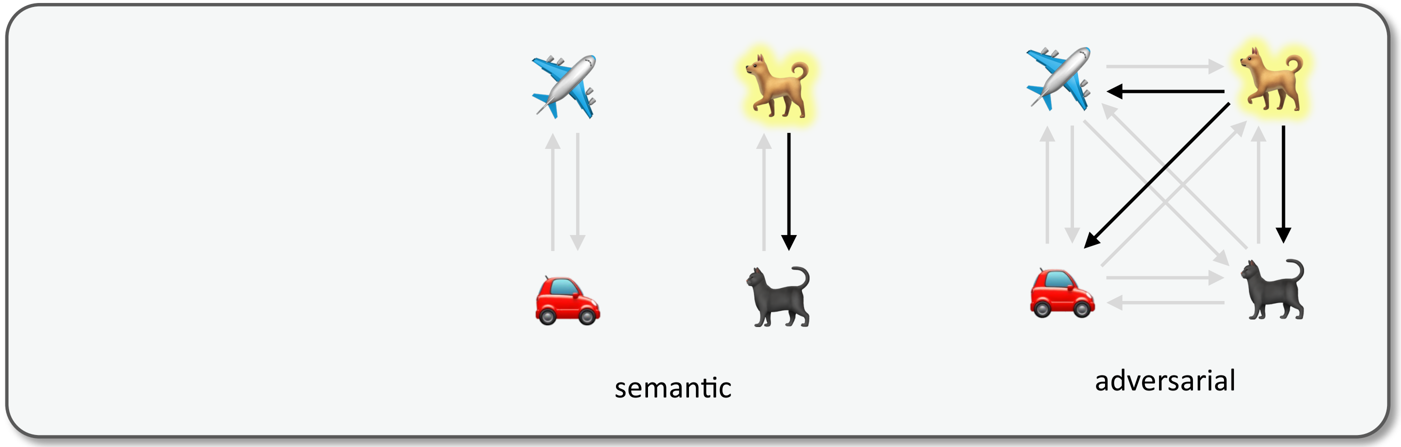
system wants: correct predictions

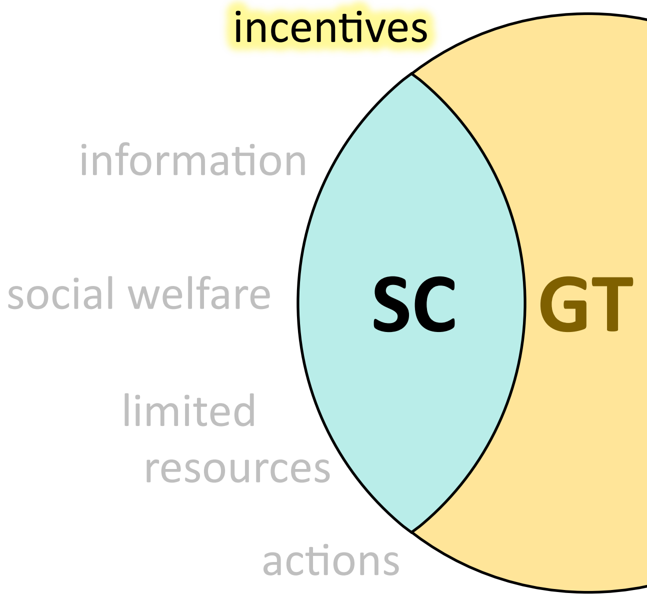
users want: wrong predictions

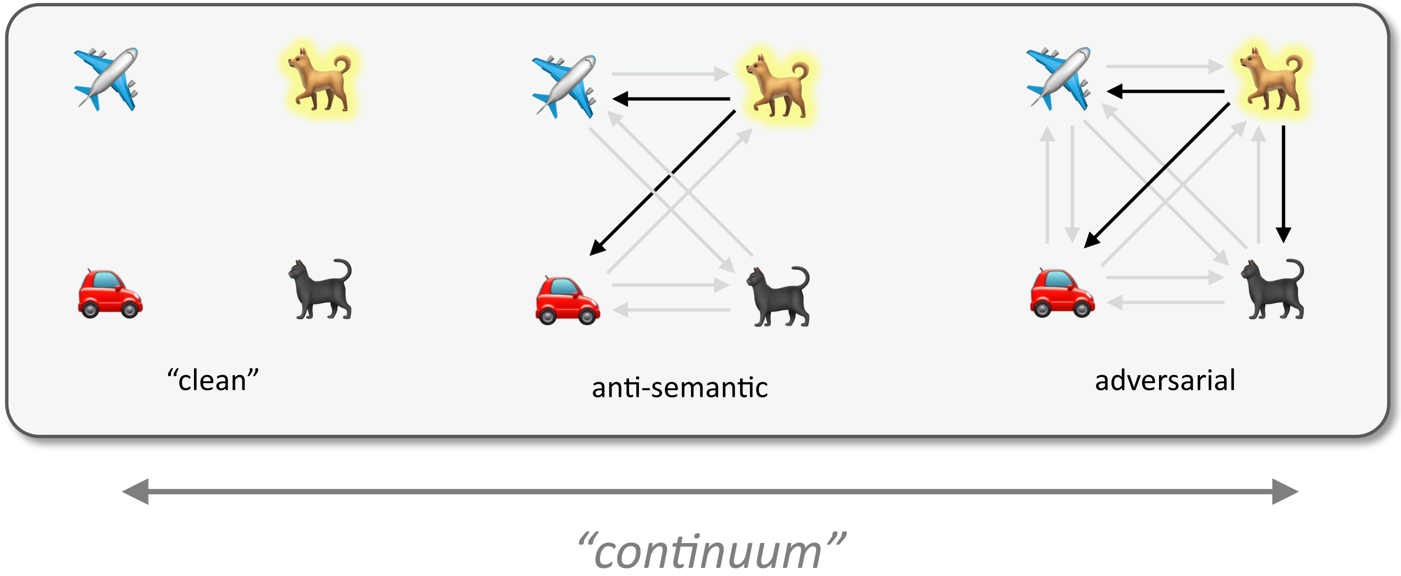


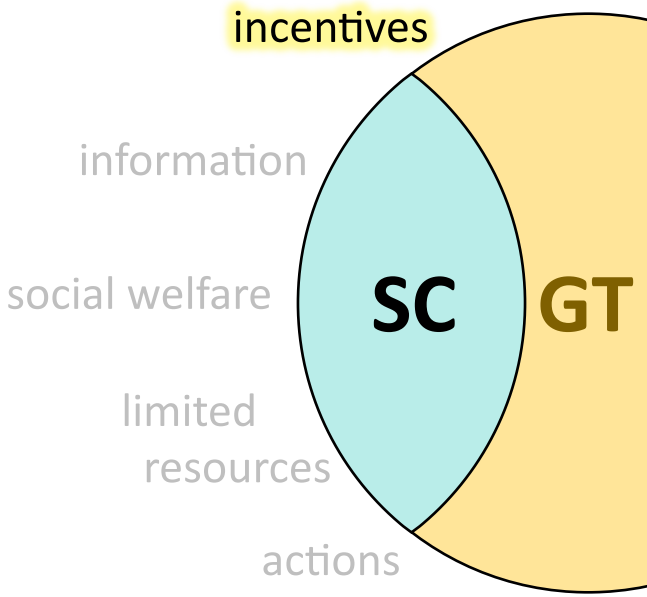


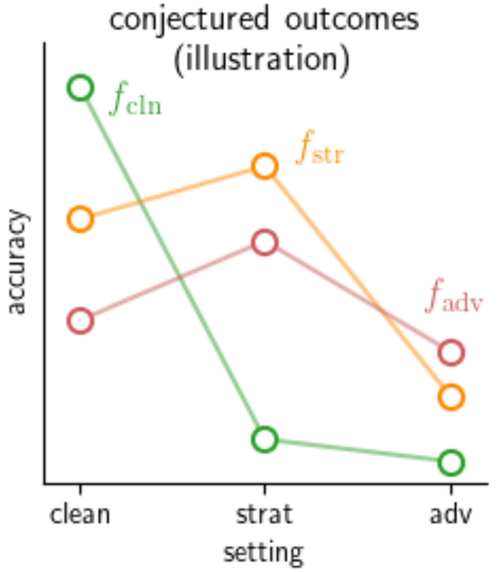
ask: can strategic modeling help make adversarial training *less conservative*?


semantic
adversarial


incentives
information
social welfare
limited resources
actions
SC
GT







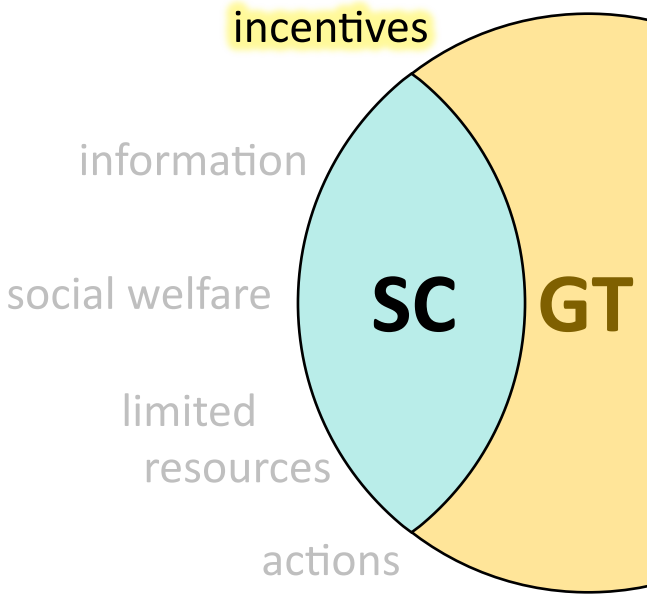
	train \ test	CIFAR-10 VGG clean	strat	adv	CIFAR-10 ResNet18 clean	strat	adv	CIFAR-10 ViT clean	strat	adv	GTSRB VGG clean	strat	adv	GTSRB ResNet18 clean	strat	adv	GTSRB ViT clean	strat	adv
sem.	clean	**89.2**	0.1	0.6	**93.1**	0.0	0.0	**77.5**	0.1	0.0	**95.1**	5.8	15.4	**96.6**	1.8	1.6	**90.7**	2.8	1.8
	strategic	72.2	**50.9**	33.7	78.9	**52.5**	40.7	58.1	**44.0**	23.6	82.4	**58.3**	**47.0**	92.0	**64.9**	**61.7**	80.2	**49.9**	42.4
	adversarial	67.7	46.4	**39.3**	79.8	49.6	**45.5**	53.1	39.8	**33.3**	80.5	47.2	43.8	90.9	55.4	53.9	79.4	49.6	**47.1**
a-sem.	clean	**89.2**	0.2	0.6	**93.1**	0.0	0.0	**77.5**	0.2	0.0	**95.1**	14.6	15.4	**96.6**	4.3	1.6	**90.7**	6.4	1.8
	strategic	80.3	**69.8**	25.0	85.1	**73.4**	33.4	65.3	**58.7**	13.1	88.2	**71.2**	36.7	92.0	**79.1**	49.0	83.5	**74.0**	34.0
	adversarial	67.7	56.8	**39.3**	79.8	67.4	**45.5**	53.1	45.8	**33.3**	80.5	63.2	**43.8**	90.9	73.0	**53.9**	79.4	64.6	**47.1**

EGR '22

much potential for synergy!
will return to this

A note on strategic vs. adversarial learning:

- From SC perspective, adversarial is "**special case**"
- But only in a narrow sense – many distinctions in practice
- E.g., in **adversarial learning** (vs. **strategic learning**):
  - attack proxy loss (e.g. log-loss) → vs. 0-1
  - focus on non-linear models → vs. mostly linear
  - focus on complex modalities (e.g. images) → vs. mostly tabular
  - ⇒ best-responses are approximate → vs. exact best-responses
  - vulnerabilities mostly in latent space → vs. no latent space
  - maximize utility under budget constraints → vs. minimize cost
  - ⇒ features always modified and to the max → vs. modify minimally – and only if needed
  - ⇒ optimize minimax objective → vs. nested min-argmax



vanilla SC

$\downarrow$

h

$\downarrow$

Δ_h

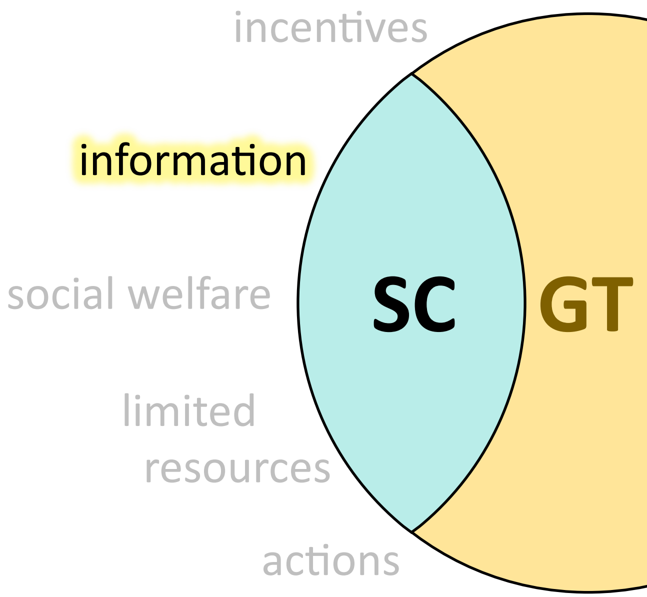
partial

$\downarrow$

$?$

$\downarrow$

$\Delta_?$



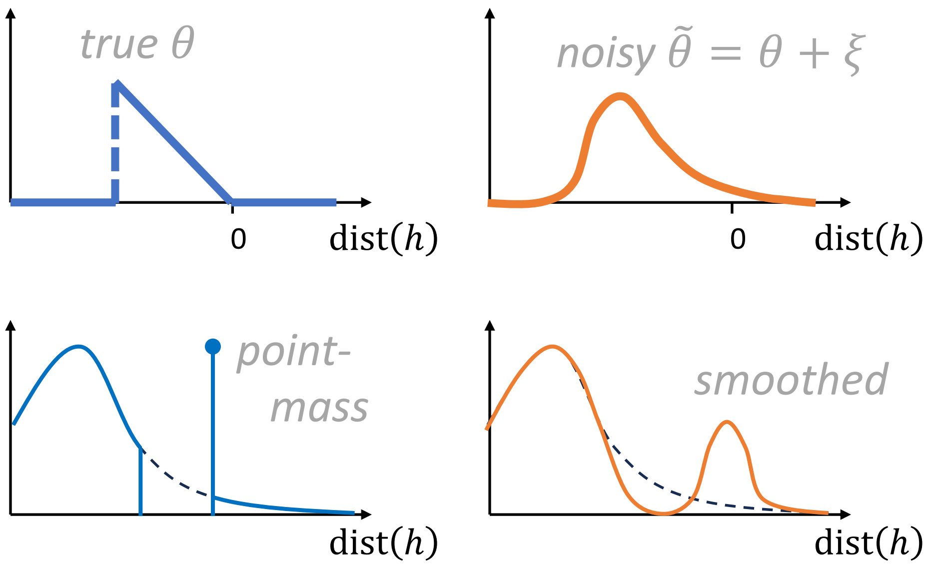
ask: what happens when users have partial knowledge of h?

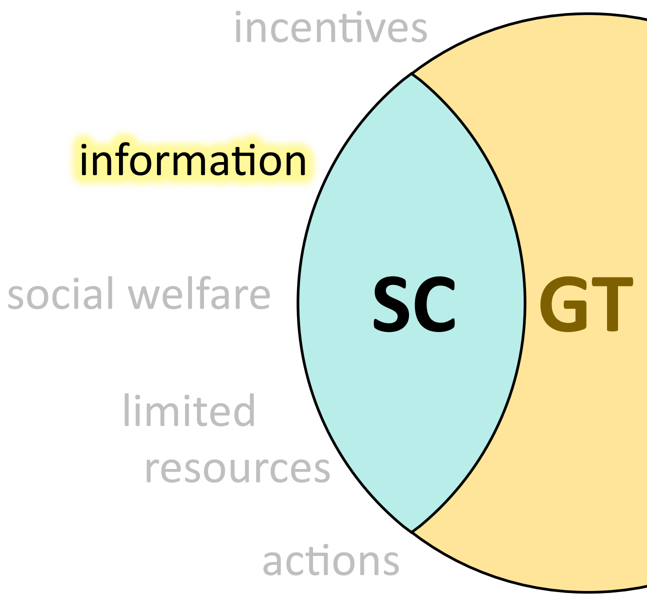
[JMH ICML21]

vanilla SC	**noisy**
$\downarrow$	$\downarrow$
h	$h' \approx h$
$\downarrow$	$\downarrow$
Δ_h	$\Delta_{h'}$

response curve:

induced distribution:





ask: what happens when users have partial knowledge of h?

vanilla SC

$\downarrow$

h

$\downarrow$

Δ_h

in the dark

$\downarrow$

(h)

$\downarrow$

$\Delta_?$

transparent

$\Downarrow$

informed users:

- have more power
- easy to anticipate

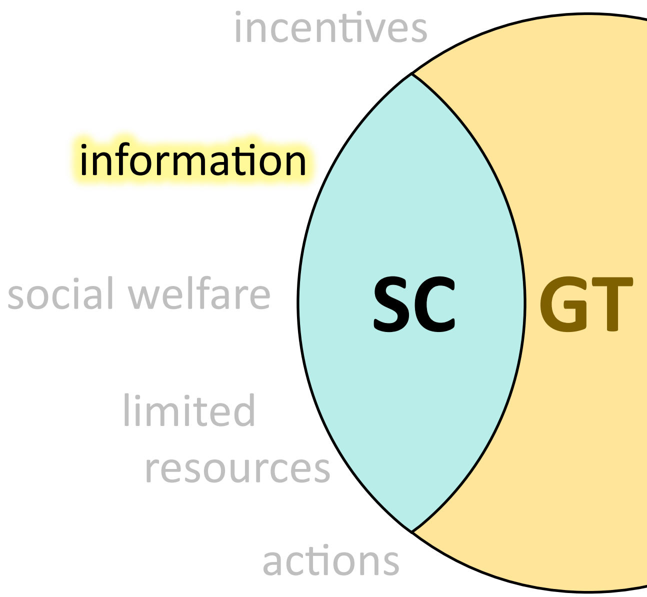
opaque

$\Downarrow$

uninformed users:

- have less power
- harder to anticipate

tradeoff



ask: what happens when users have partial knowledge of h?

vanilla SC

$\downarrow$

h

$\downarrow$

Δ_h

"optimistic"

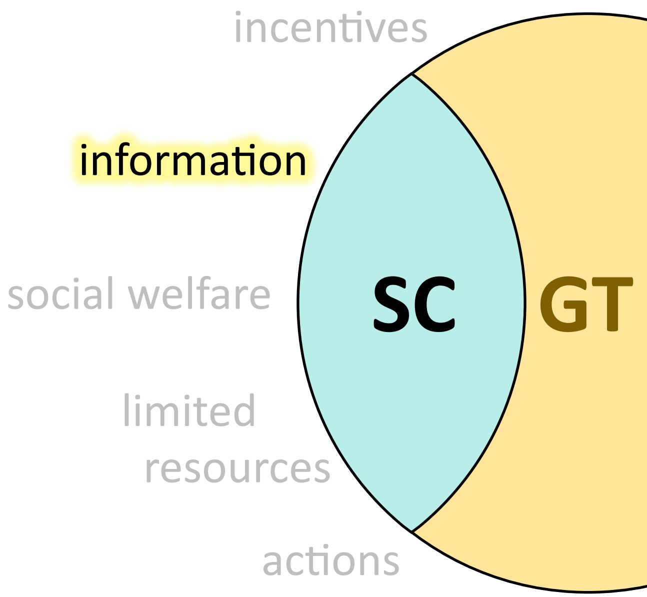
info leaks

$h \qquad \hat{h}$

$\downarrow \qquad \downarrow$

$\Delta_h \approx \Delta_{\hat{h}}$

SCHUFA



ask: what happens when users have partial knowledge of h?

vanilla SC

$\downarrow$

h

$\downarrow$

Δ_h

"optimistic"

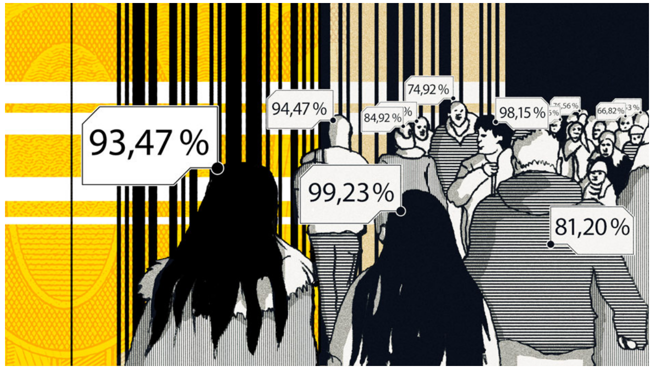
$\swarrow$ $\searrow$ *info leaks*

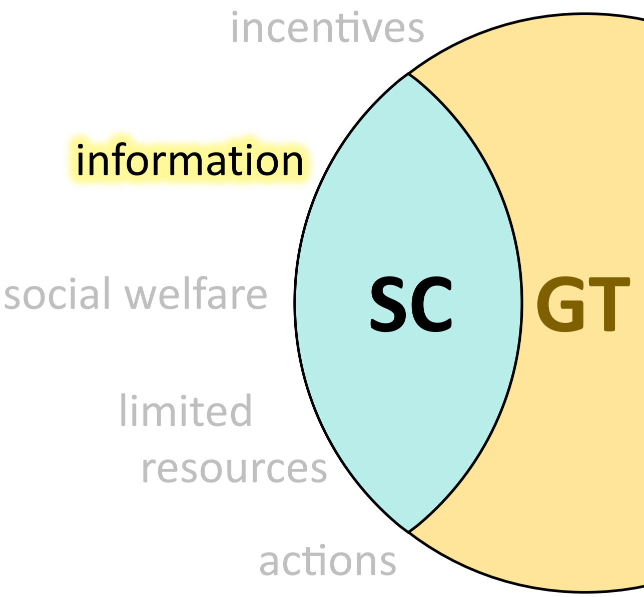
h $\quad$ $\hat{h}$

$\downarrow$ $\quad$ $\downarrow$

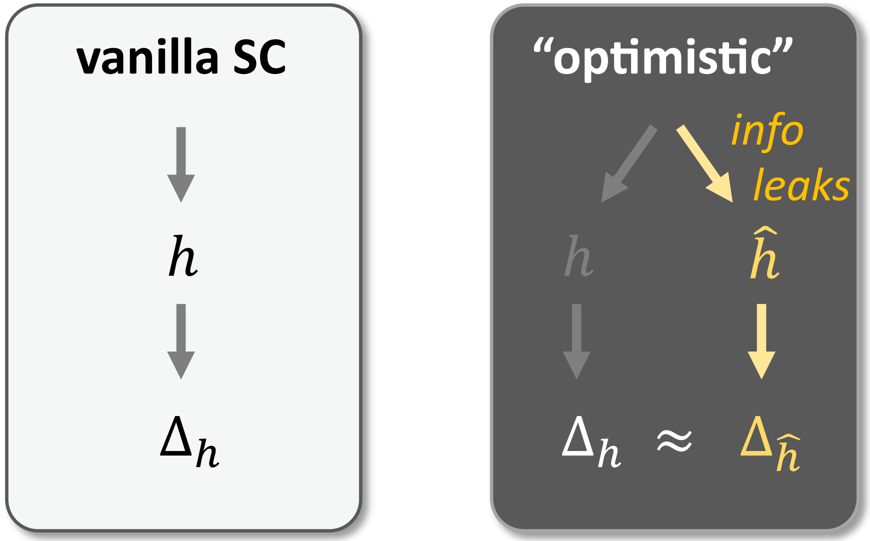
$\Delta_h \approx \Delta_{\hat{h}}$

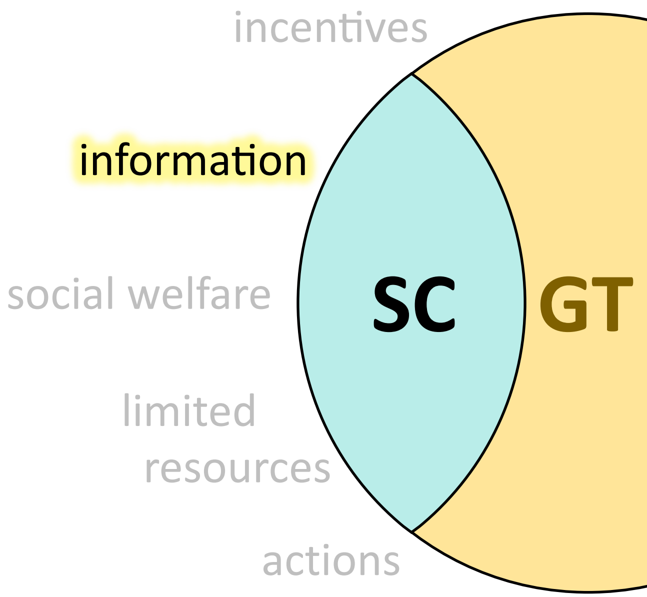
OPEN SCHUFA





ask: what happens when users have partial knowledge of h?



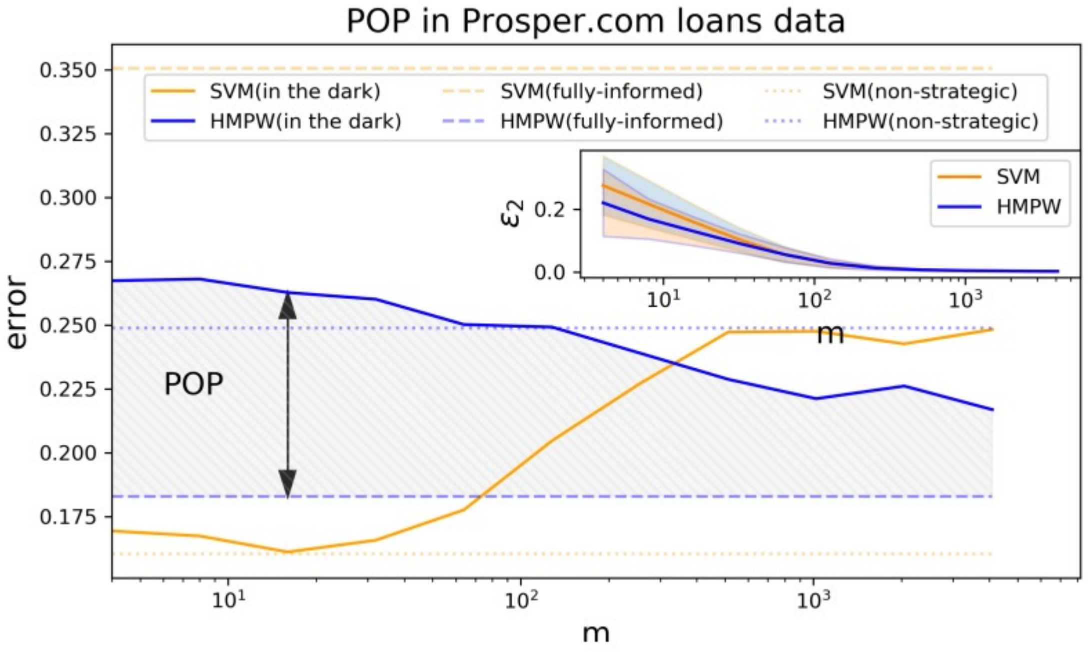


- **Price of OPacity:** (POP)

$$err(h, \hat{h}) - err(h, h)$$

- **Main result:** can be arbitrarily bad
  $\Rightarrow$ **transparency** is often in best interest of system!

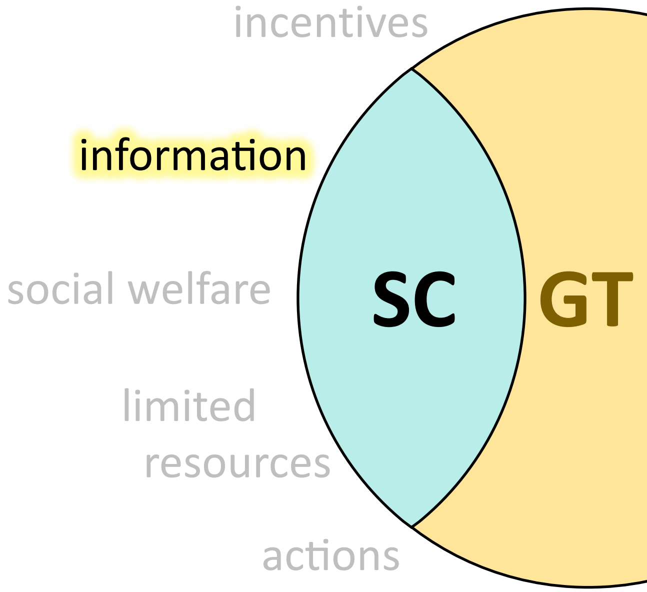
ask: what happens when users have partial knowledge of h?



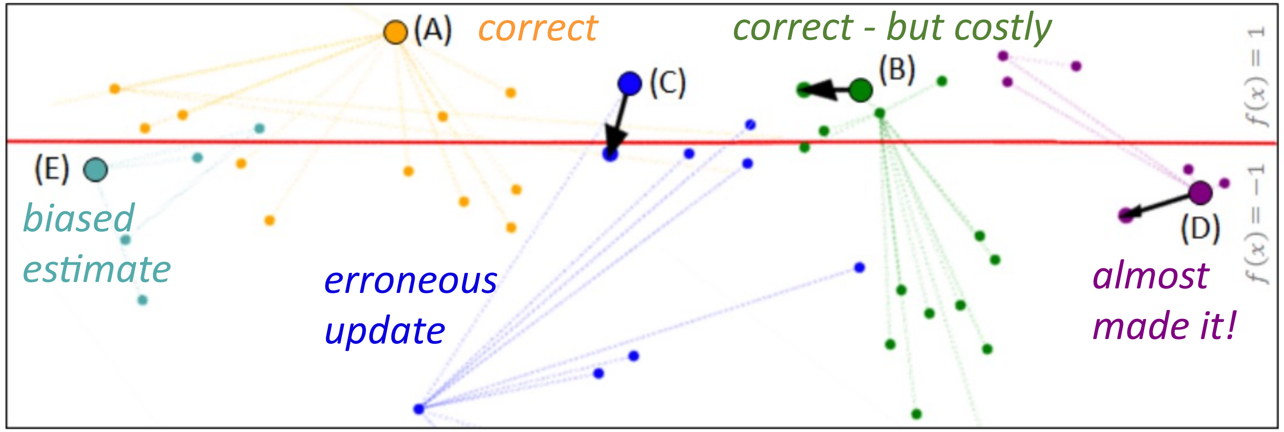
- **Price of OPacity:** (POP)

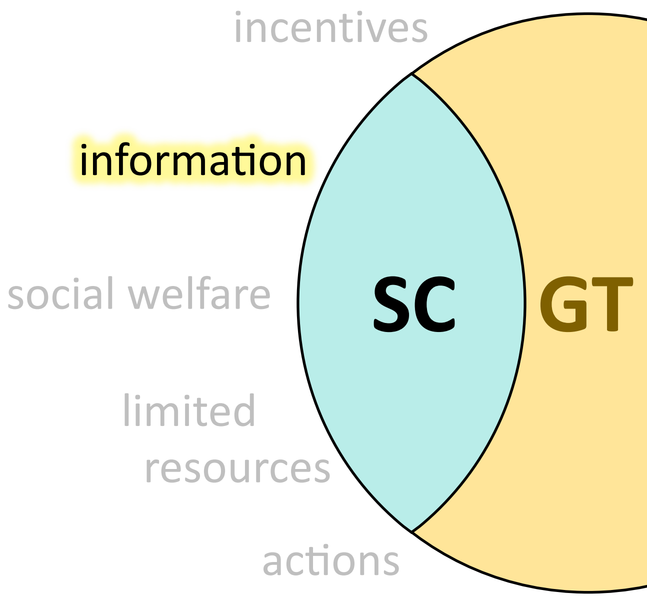
$$err(h, \hat{h}) - err(h, h)$$

- **Main result:** can be arbitrarily bad
  ⇒ **transparency** is often in best interest of system!



ask: what happens when users have partial knowledge of h?





ask: what happens when users have partial knowledge of h?

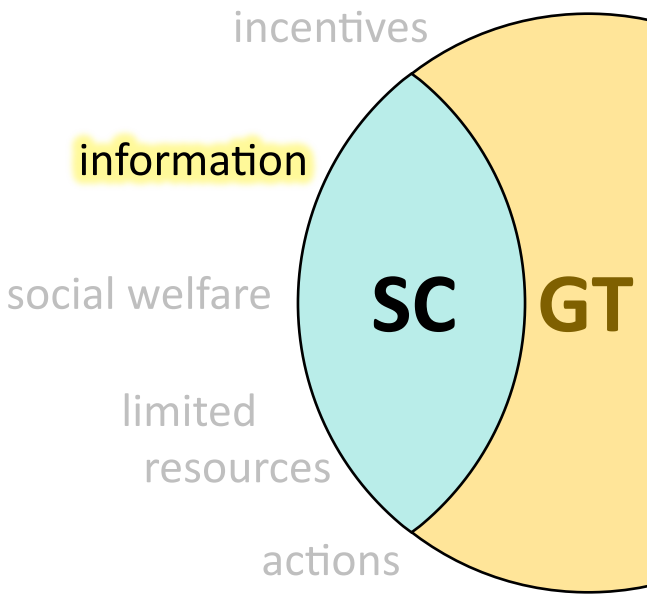
known user response:

$$\operatorname*{argmin}_h \frac{1}{m}\sum_{i=1}^{m} \ell\left(y_i, h\left(\Delta_h(x_i)\right)\right)$$

uncertainty

***un**known user response:*

$$\operatorname*{argmin}_h \frac{1}{m}\sum_{i=1}^{m} \ell\left(y_i, h\left(\Delta_?(x_i)\right)\right)$$



ask: how can learning contend with uncertain user behavior?

known user response:

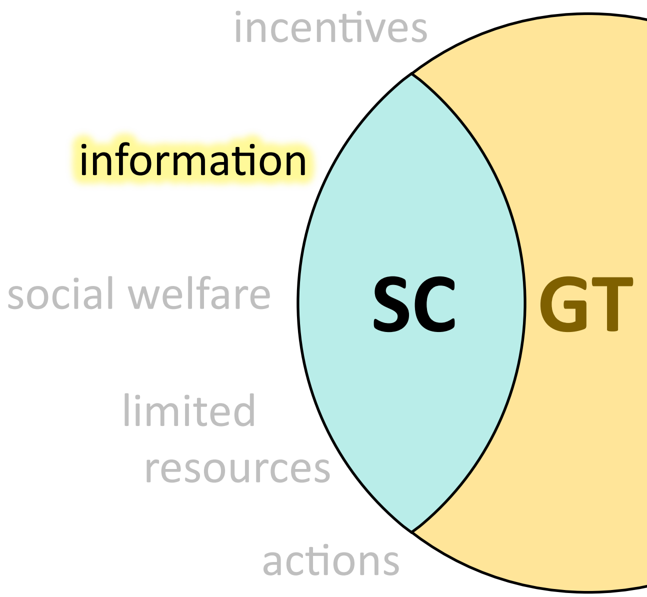
$$\underset{h}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h(\Delta_h(x_i))\right)$$

uncertainty

unknown user response:

$$\underset{h}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h(\Delta_?(x_i))\right)$$

1) infer Δ over time (more on this later)



ask: how can learning contend with uncertain user behavior?

known user response:

$$\operatorname*{argmin}_h \frac{1}{m}\sum_{i=1}^{m} \ell\left(y_i, h(\Delta_h(x_i))\right)$$

uncertainty

unknown user response:

$$\operatorname*{argmin}_h \frac{1}{m}\sum_{i=1}^{m} \ell\left(y_i, h(\Delta_?(x_i))\right)$$

1) infer Δ over time (more on this later)

2) ***robust learning:***

$$\operatorname*{argmin}_h \max_{\Delta \in \mathbb{U}} \frac{1}{m}\sum_{i=1}^{m} \ell\left(y_i, h(\Delta_h(x_i))\right)$$

uncertainty set

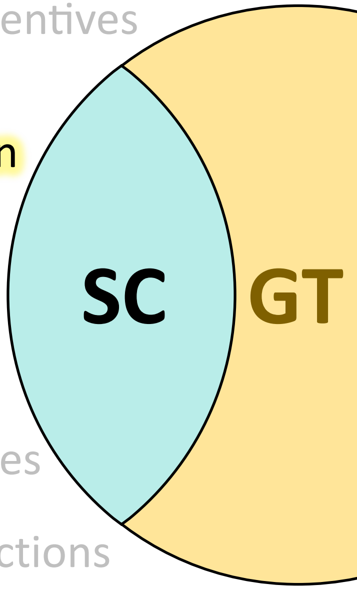
incentives

information

social welfare

limited resources

actions



ask: how can learning contend with uncertain user behavior?

known user response:

$$\operatorname*{argmin}_h \frac{1}{m}\sum_{i=1}^{m} \ell\left(y_i, h(\Delta_h(x_i))\right)$$

uncertainty

unknown user response:

$$\operatorname*{argmin}_h \frac{1}{m}\sum_{i=1}^{m} \ell\left(y_i, h(\Delta_?(x_i))\right)$$

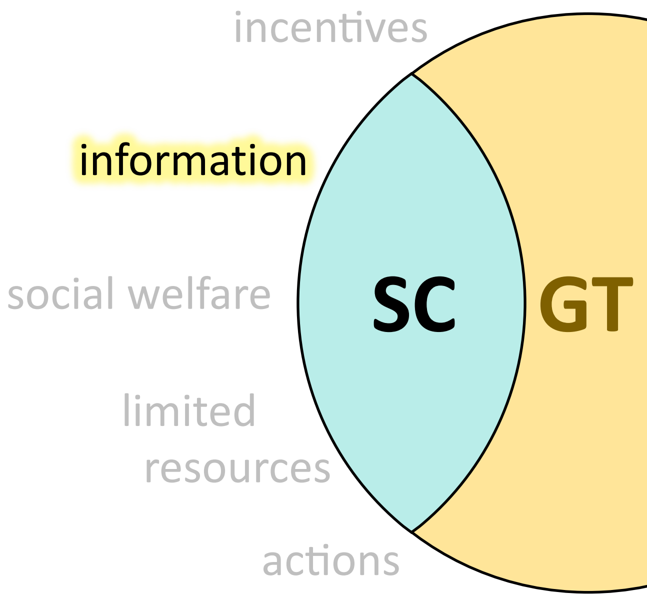
1) infer Δ over time (more on this later)

2) ***robust learning*** *– unkonwn costs:*

$$\operatorname*{argmin}_h \max_{c \in C} \frac{1}{m}\sum_{i=1}^{m} \ell\left(y_i, h(\Delta_h^c(x_i))\right)$$

uncertainty set

check out our poster!



ask: how can learning contend with uncertain user behavior?

known user response:

$$\underset{h}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h(\Delta_h(x_i))\right)$$

unknown user response:

$$\underset{h}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h(\Delta_?(x_i))\right)$$

1) infer Δ over time (more on this later)

2) ***robust learning*** – *unkonwn costs:*

$$\underset{h}{\operatorname{argmin}} \max_{c \in C} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h(\Delta_h^c(x_i))\right)$$

uncertainty set

(public) policy problems:

- **"One shot"** – can deploy <u>only once</u>
- **Goal**: learn to be doubly-robust:
  - vs. strategic behavior
  - vs. worst-case cost $c \in C$
- **Hardness:** not knowing c can be **catastrophic**
- **Convexification:** updated ad-hoc s-hinge
- **Algorithm:** effective, converge to opt. min-max

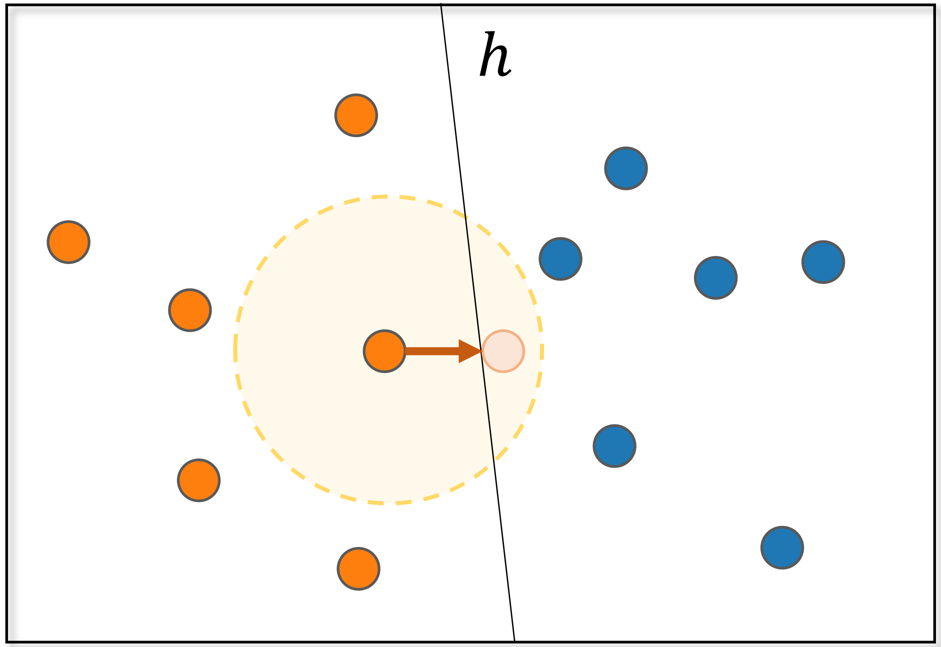
incentives

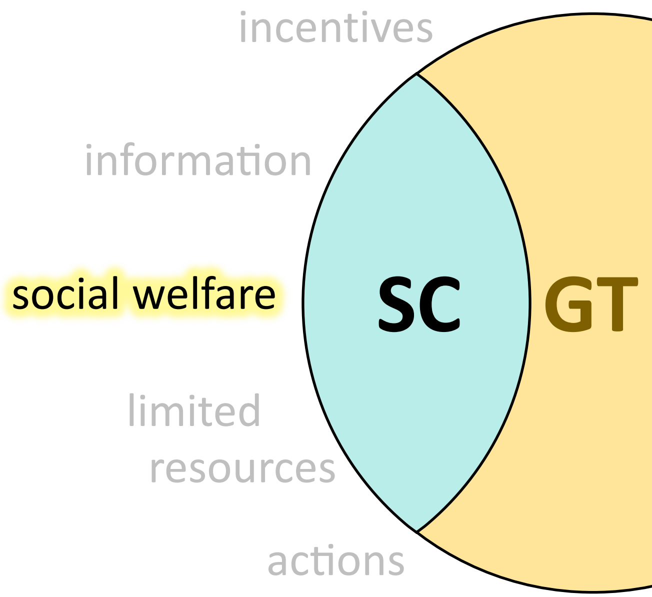
information

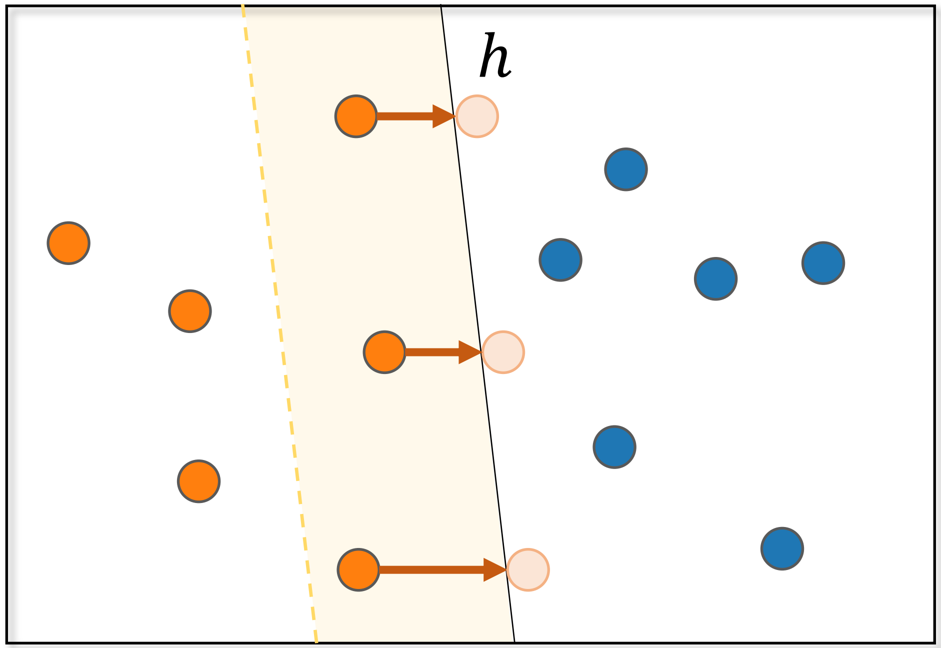
social welfare

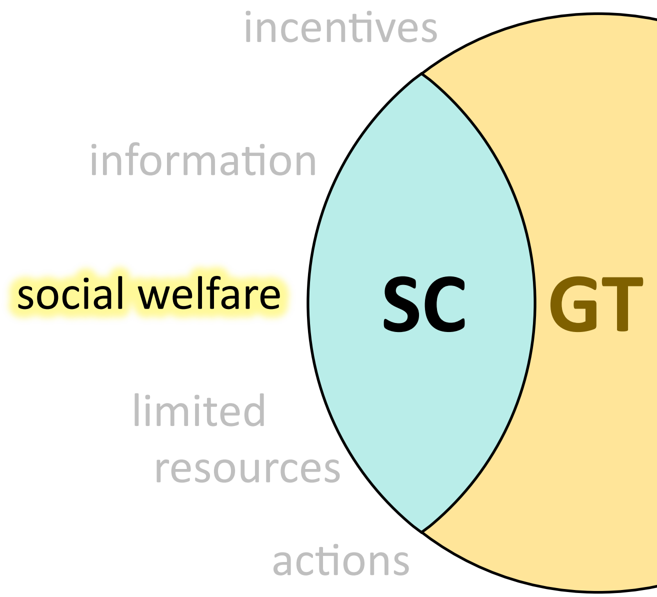
limited resources

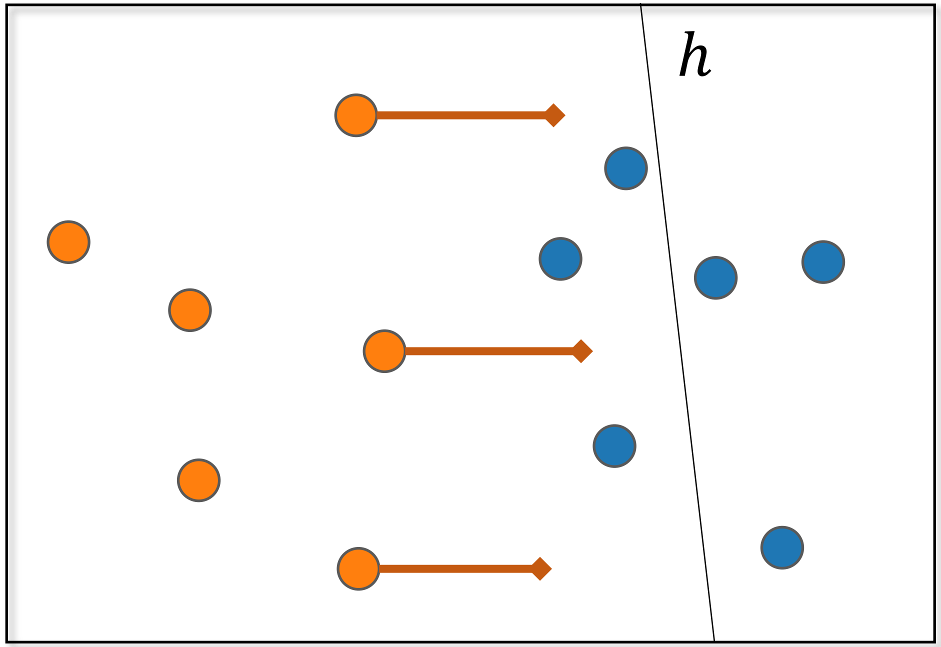
actions


h

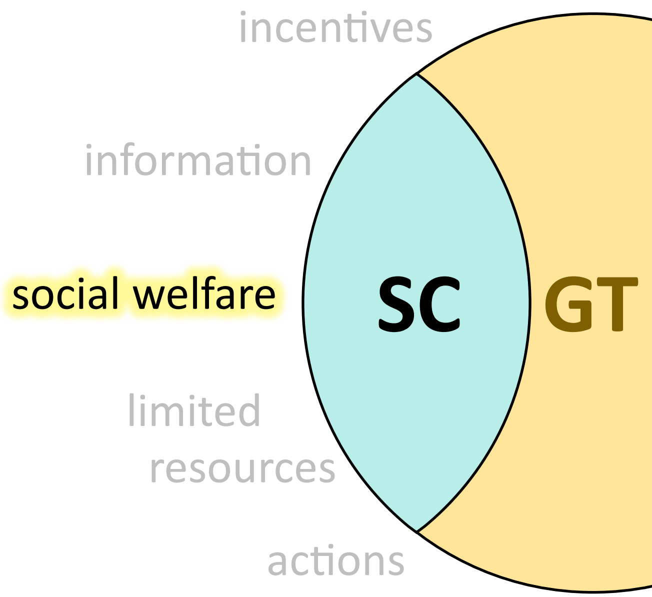

incentives
information
social welfare
limited resources
actions
SC
GT


h


incentives
information
social welfare
limited resources
actions
SC
GT



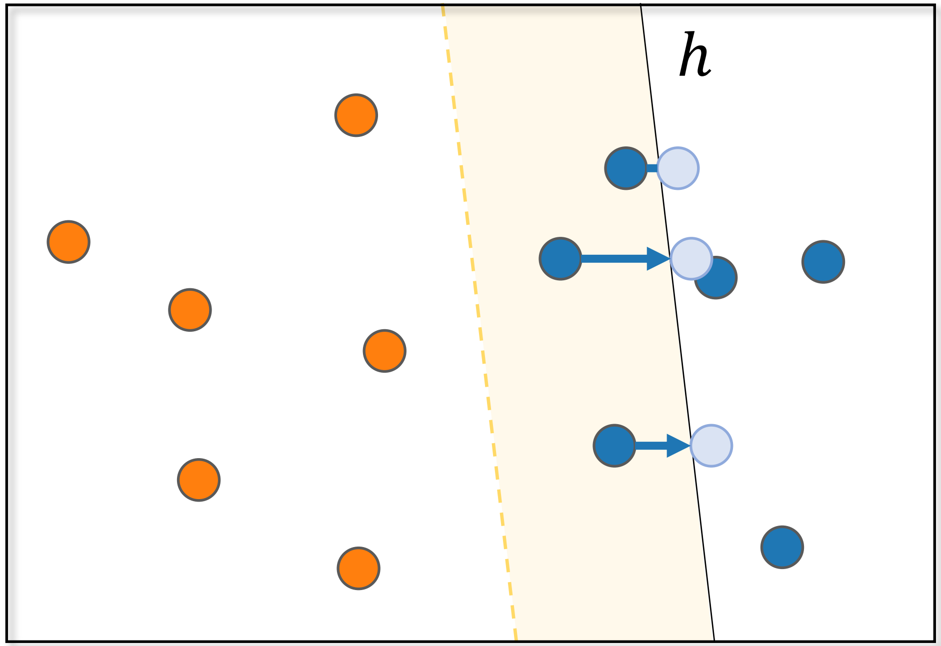
h



incentives
information
social welfare
limited resources
actions
SC
GT

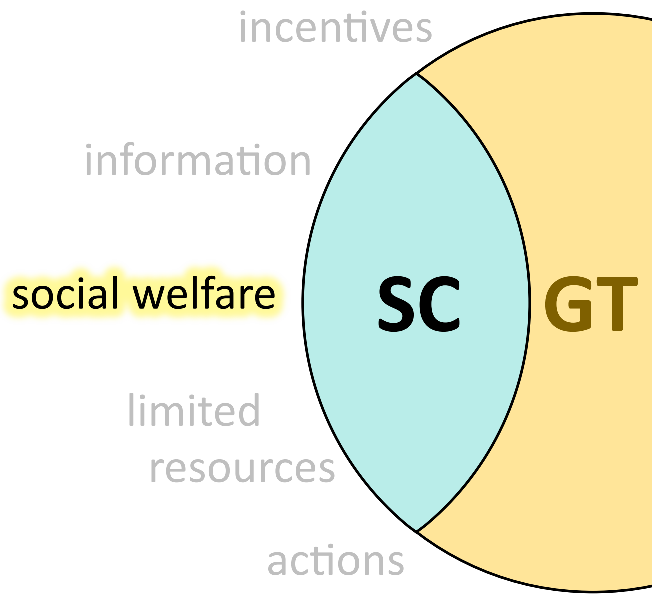
- Robustness via penalizing deserving sub-population
- Main result is **negative**: increased accuracy ⇒ increased social burden
- However, results apply to certain **monotone** setting
- In more general settings, there is **reason for optimism**!

ask: when and how can we reduce social harm?



= **"social burden"** [MMDH'19]

$$\text{burden}(h) = \mathbb{E}\left[\min_{x':h(x')=1} c(x,x') \mid y=1\right]$$



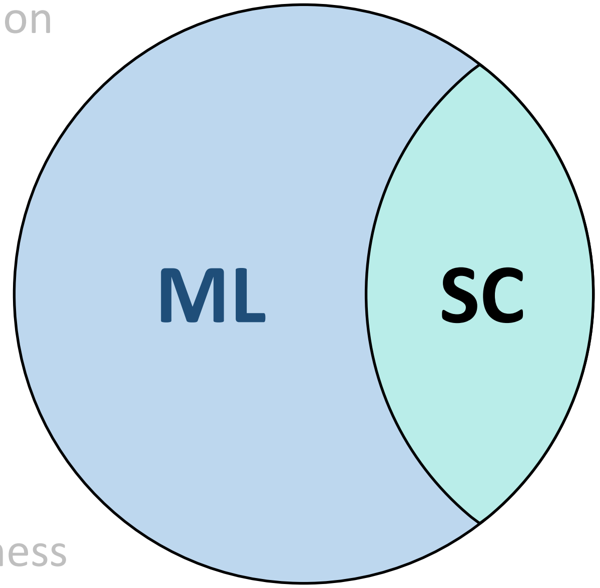
optimization

generalization

loss functions

regularization
model selection

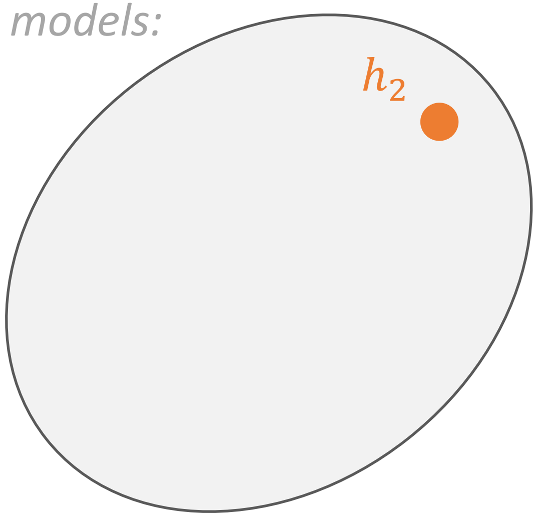
robustness
uncertainty



- **Conjecture**: many good models, vary in burden
- Learning objective underspecified – **can exploit!**
- Regularize for **generalization**:

$$\underset{h \in H}{\operatorname{argmin}} \sum_{i=1}^{m} \ell\left(y, h(\Delta_h(x))\right) + \lambda \|h\|_2$$

set of accurate models:



optimization

generalization

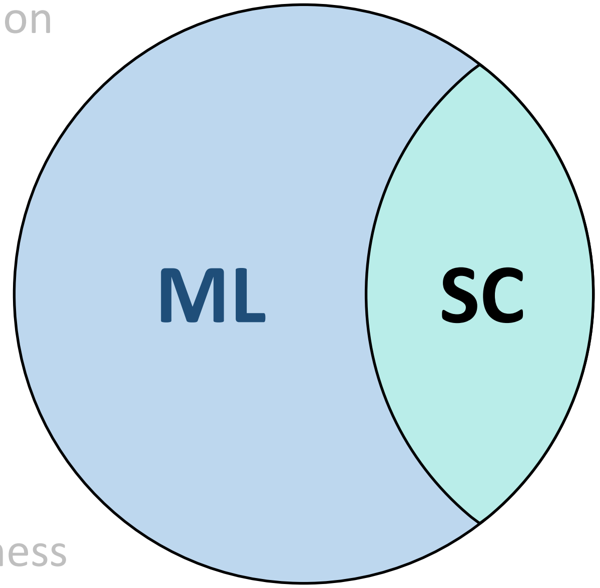
loss functions

regularization

model selection

robustness

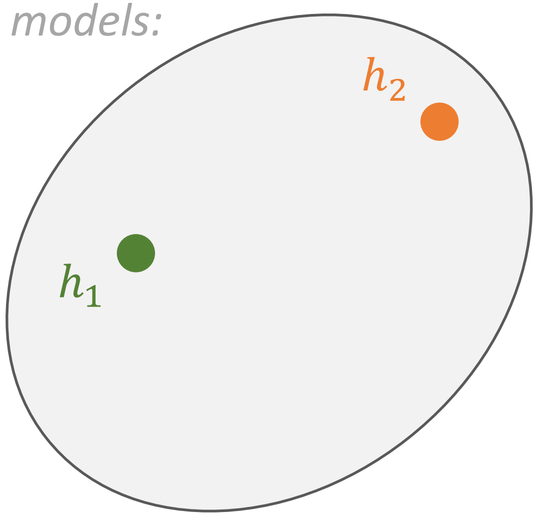
uncertainty



- **Conjecture**: many good models, vary in induced burden
- Learning objective underspecified – **can exploit!**
- Regularize for **sparsity**:

$$\underset{h \in H}{\operatorname{argmin}} \sum_{i=1}^{m} \ell\left(y, h(\Delta_h(x))\right) + \lambda \, \|h\|_1$$

set of accurate models:



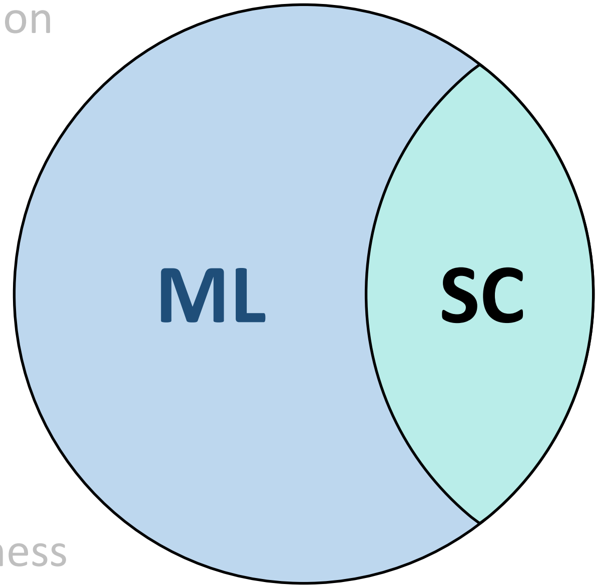
optimization

generalization

loss functions

regularization
model selection

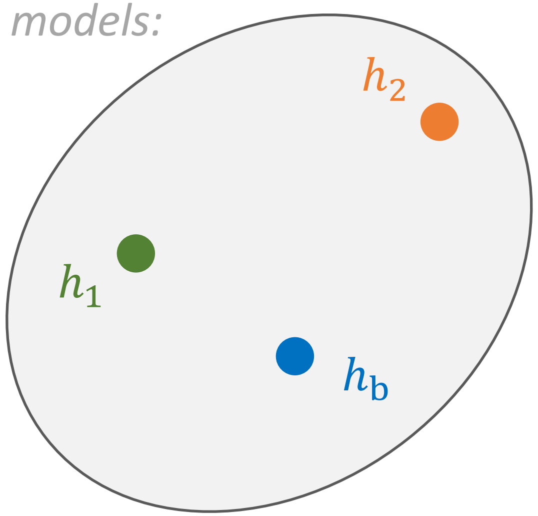
robustness
uncertainty

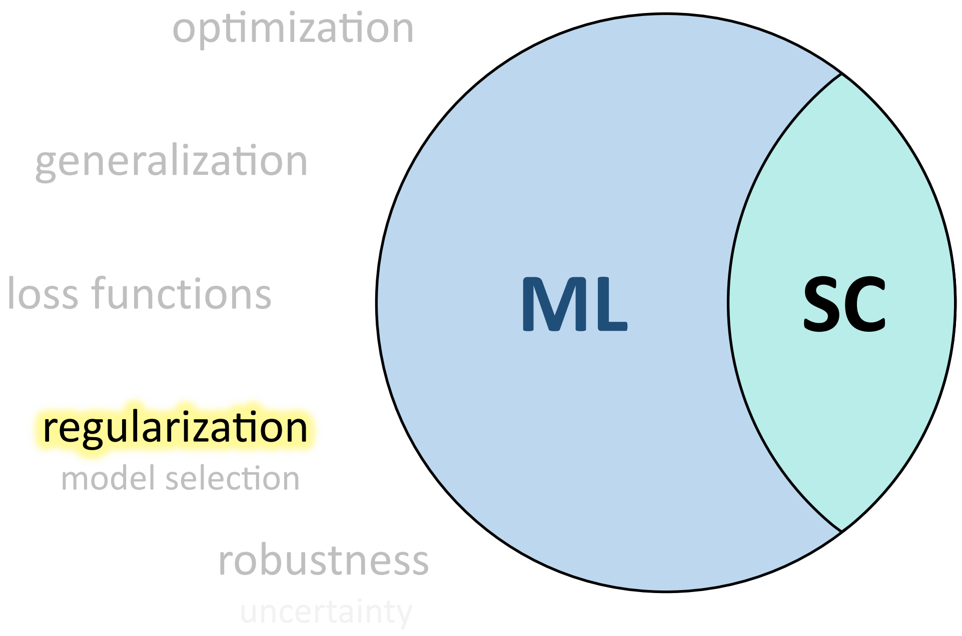


- **Conjecture**: many good models, vary in induced burden
- Learning objective underspecified – **can exploit!**
- Regularize for… **social good?**

$$\underset{h\in H}{\operatorname{argmin}} \sum_{i=1}^{m} \ell\left(y, h\left(\Delta_h(x)\right)\right) + \lambda\, R_{\text{burden}}(h)$$

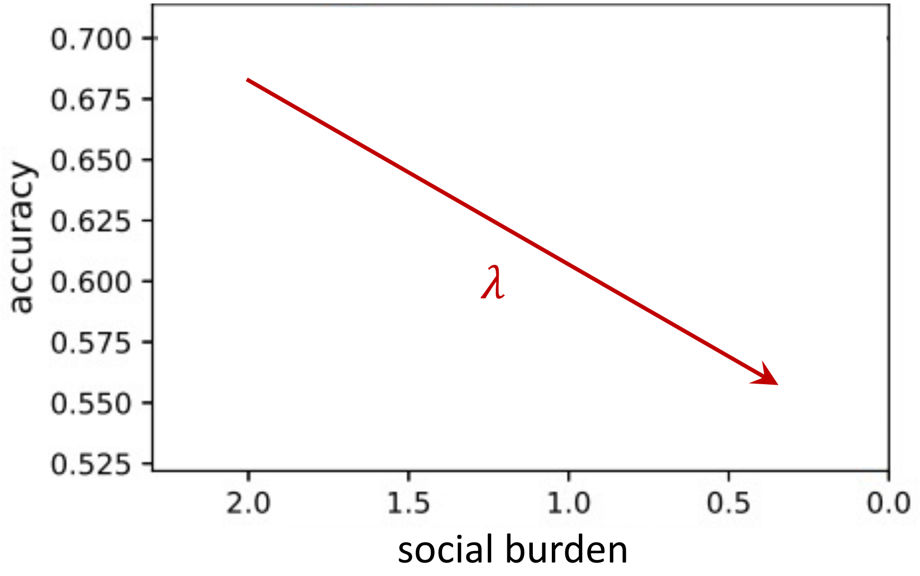
set of accurate models:





- **Conjecture**: many good models, vary in induced burden
- Learning objective underspecified – **can exploit!**
- Regularize for… **social good?**

$$\underset{h \in H}{\operatorname{argmin}} \sum_{i=1}^{m} \ell\left(y, h(\Delta_h(x))\right) + \lambda\, R_{\text{burden}}(h)$$



optimization

generalization

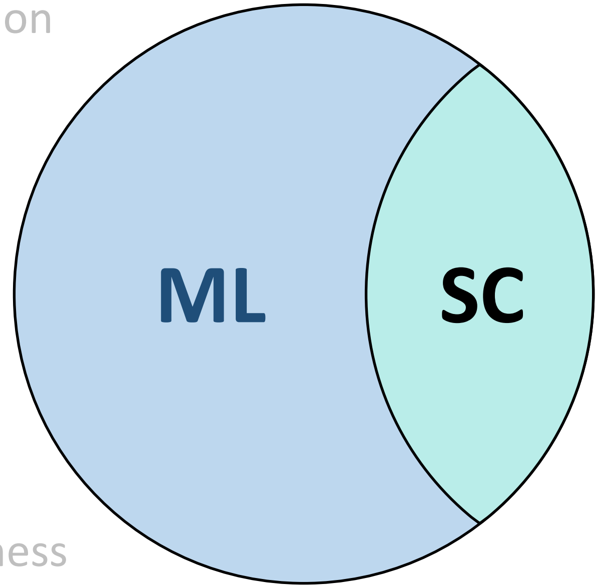
loss functions

regularization

model selection

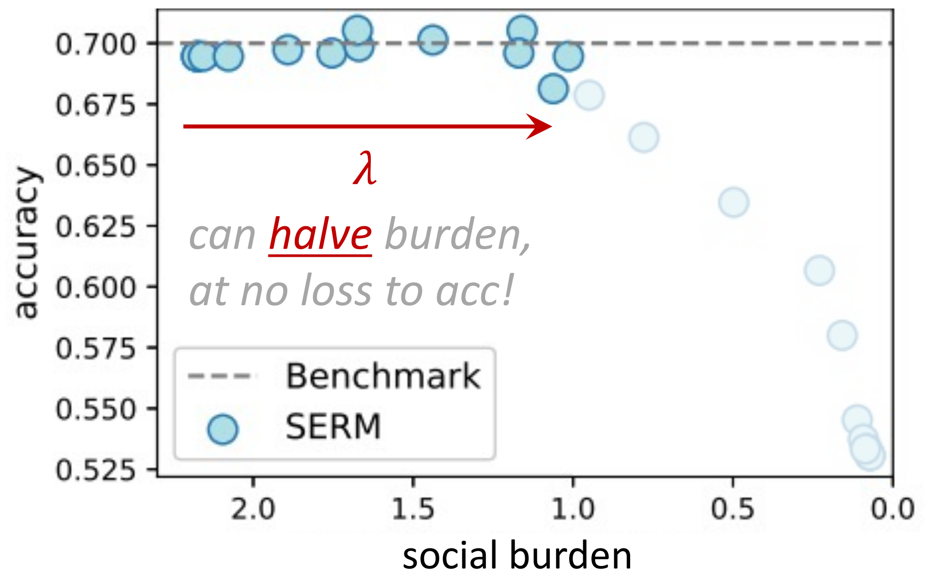
robustness

uncertainty



- **Conjecture**: many good models, vary in induced burden
- Learning objective underspecified – **can exploit!**
- Regularize for… **social good!**

$$\underset{h \in H}{\operatorname{argmin}} \sum_{i=1}^{m} \ell\left(y, h(\Delta_h(x))\right) + \lambda \, R_{\text{burden}}(h)$$



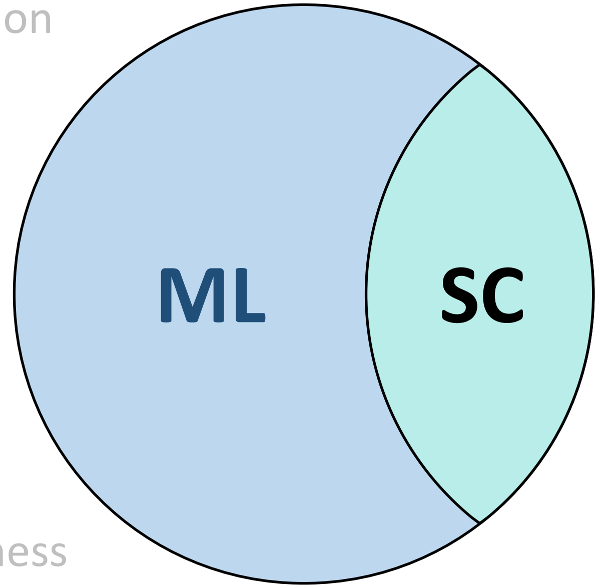
optimization

generalization

loss functions

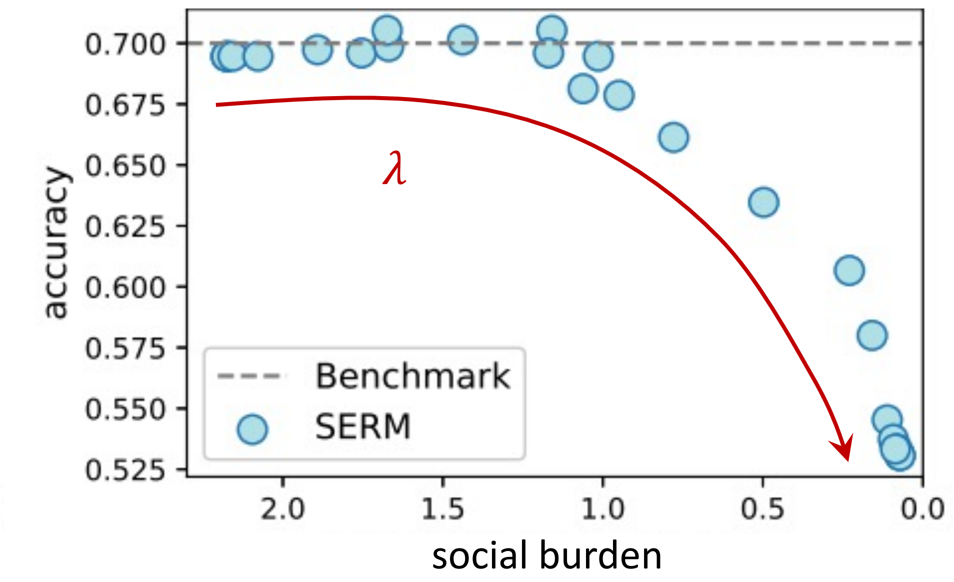
regularization
model selection

robustness
uncertainty

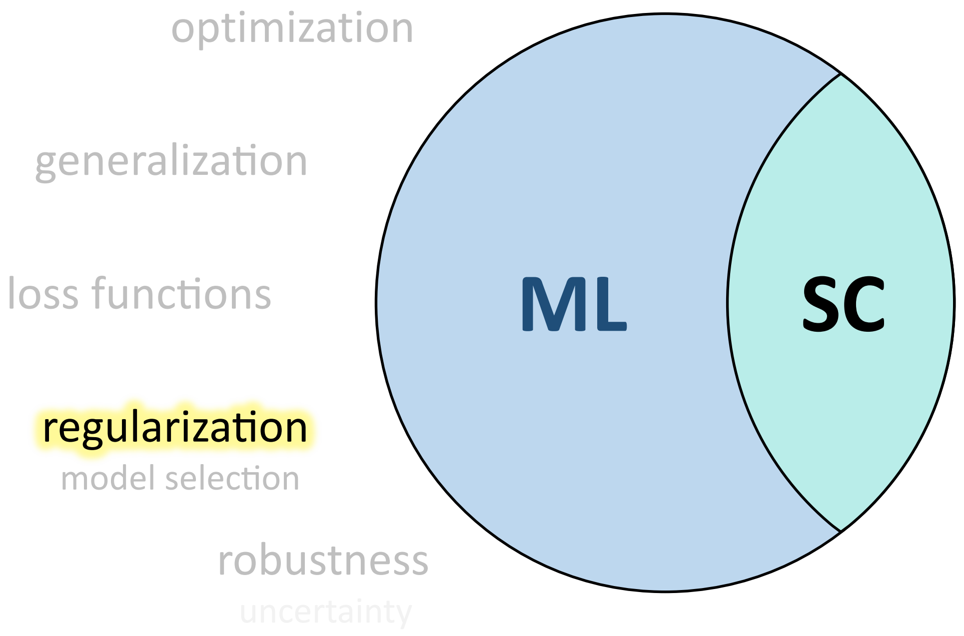


- **Conjecture**: many good models, vary in induced burden
- Learning objective underspecified – **can exploit!**
- Regularize for… **social good!**

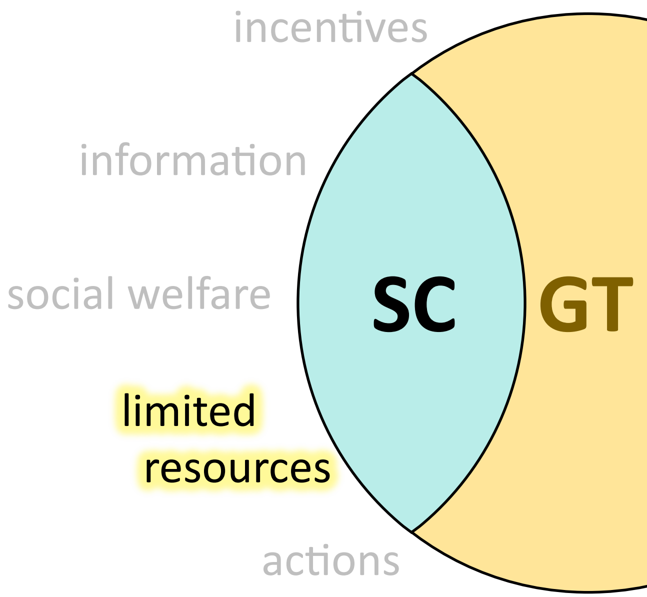
$$\operatorname*{argmin}_{h \in H} \sum_{i=1}^{m} \ell\left(y, h(\Delta_h(x))\right) + \lambda \, R_{\text{burden}}(h)$$

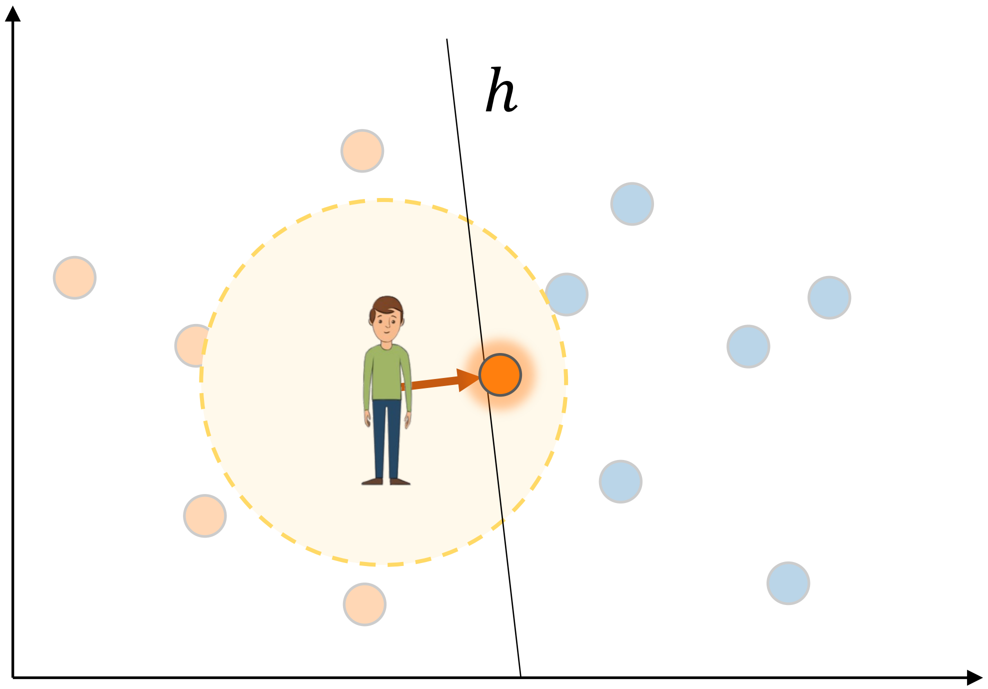


additional social good metrics:

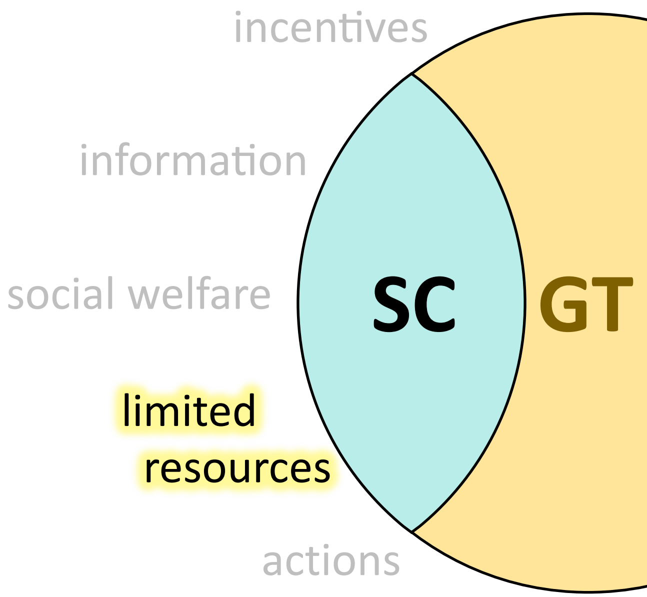


- Applies to other social good metrics (utility, recourse, …)
- Similarly underspecified – similar pareto fronts!


incentives
information
social welfare
limited resources
actions
SC
GT

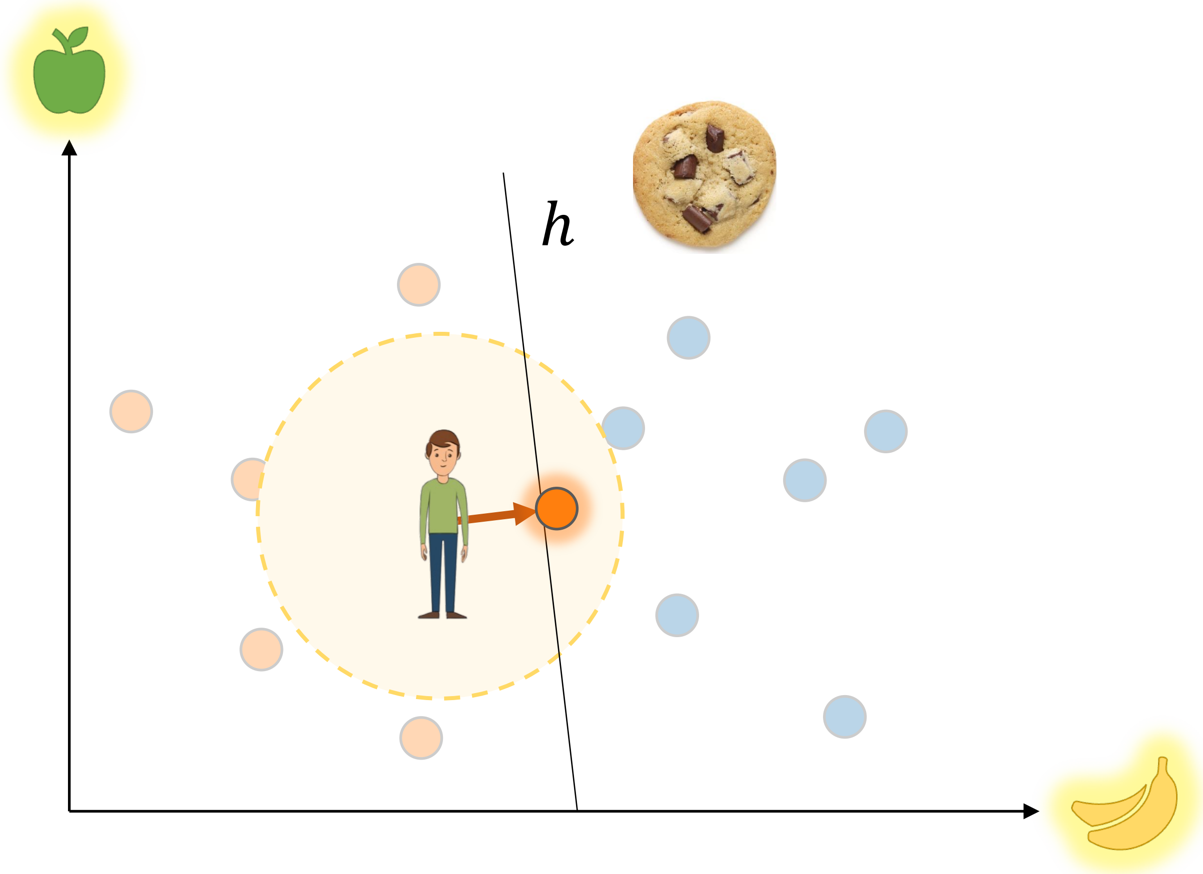


$$\Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ \underbrace{h(x')}_{\text{utility}} - \underbrace{c(x,x')}_{\text{cost}}$$



ask: where do costs come from?

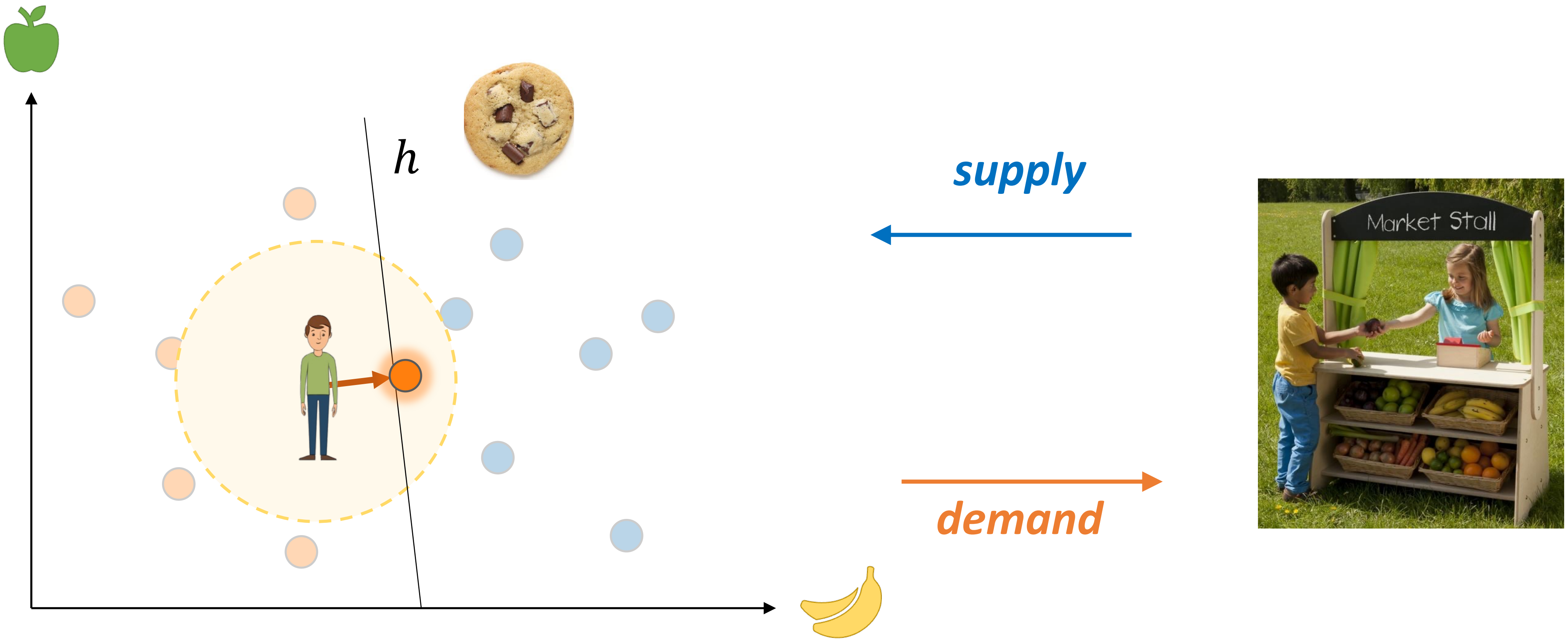
(ask first: what are features?)



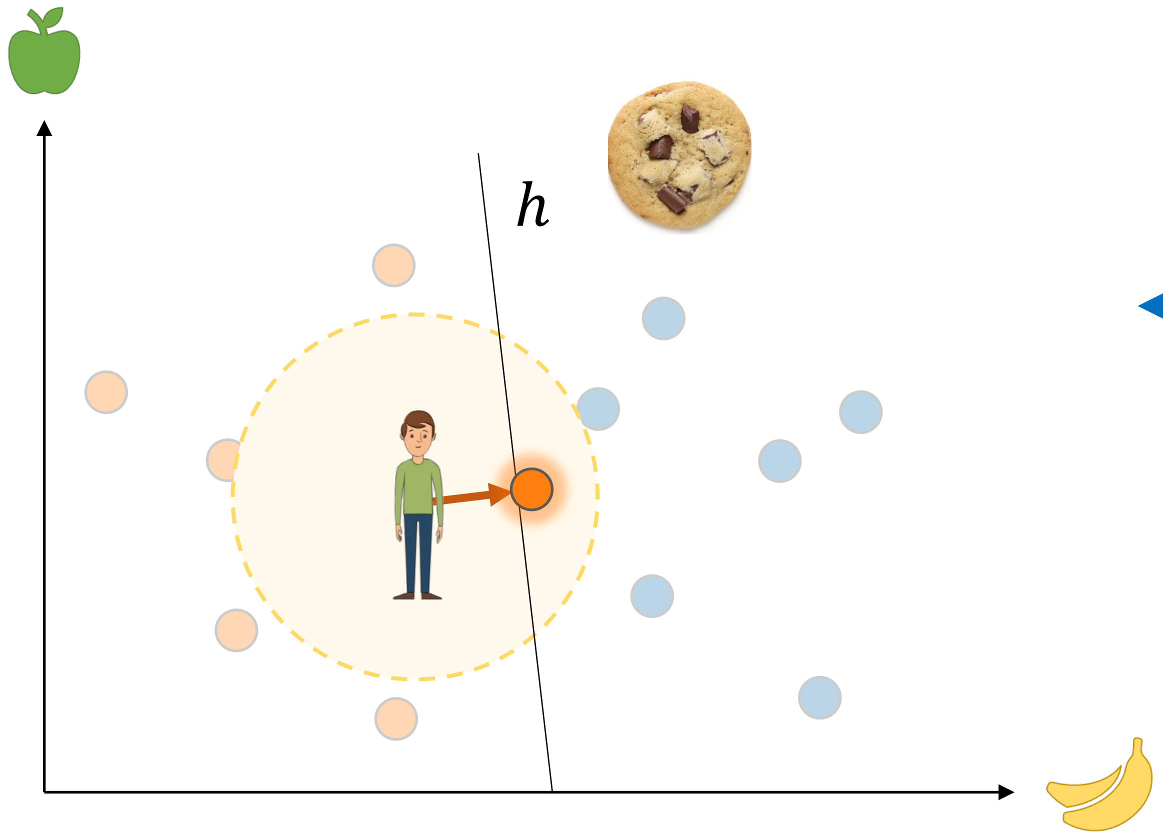
$$\Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ h(x') - c(x, x')$$

utility *cost of **apples** and **bananas***

ask: where do fruits come from?



$$\Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ \underbrace{h(x')}_{\text{utility}} - \underbrace{c(x,x')}_{\text{cost of apples and bananas}}$$



$$\Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ h(x') - c(x, x')$$

utility

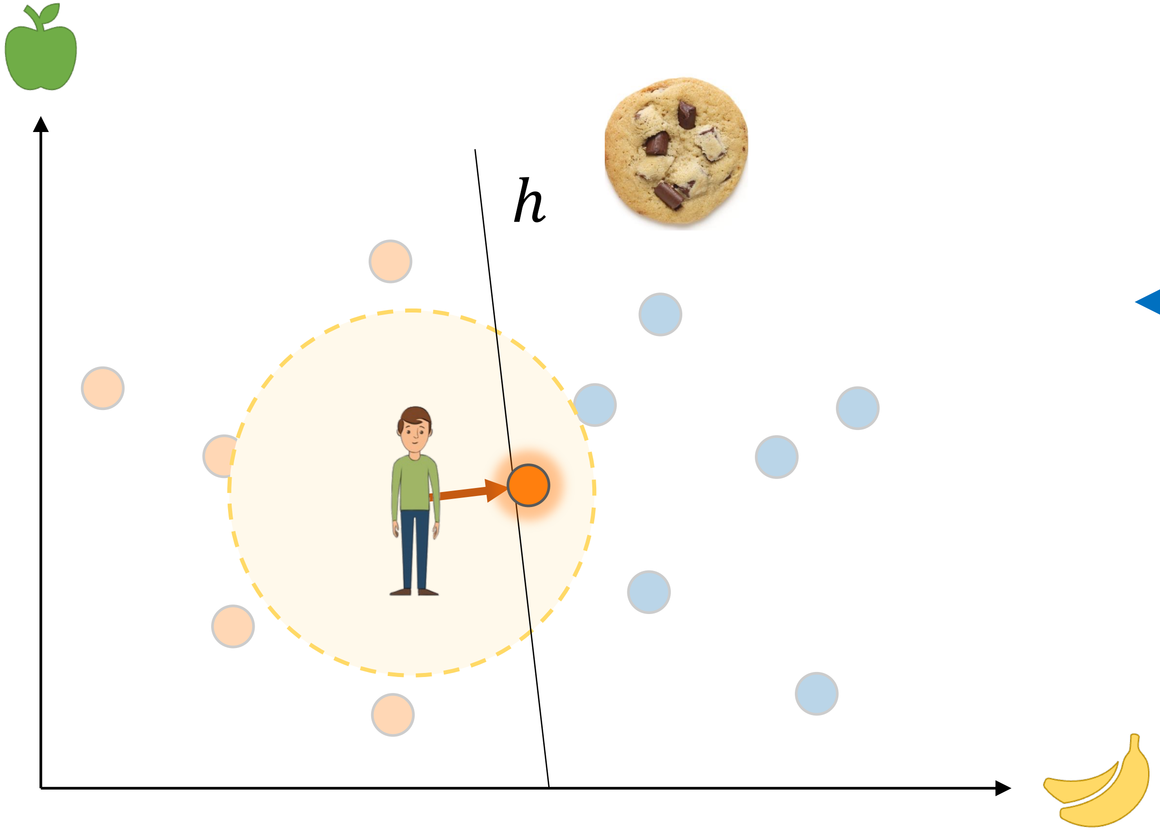
price of **apples** and **bananas**

supply

$

price

demand



supply

$

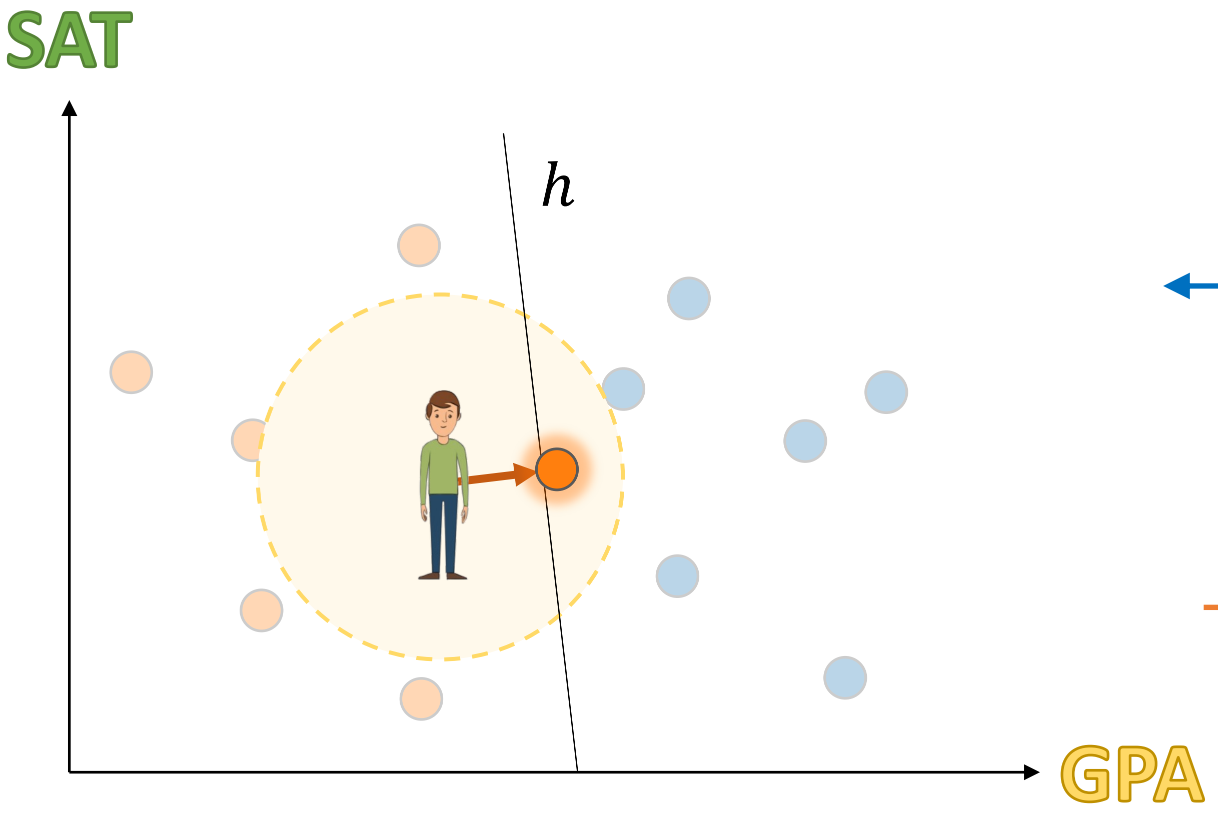
price

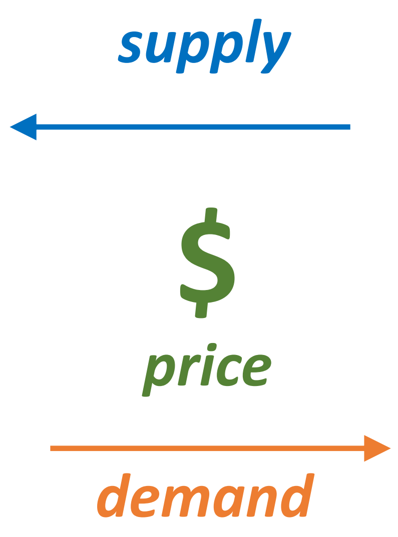
demand

$$\Delta_h(x) = \underset{x'}{\operatorname{argmax}} \; h(x') - c(x, x')$$

utility

price of apples and bananas

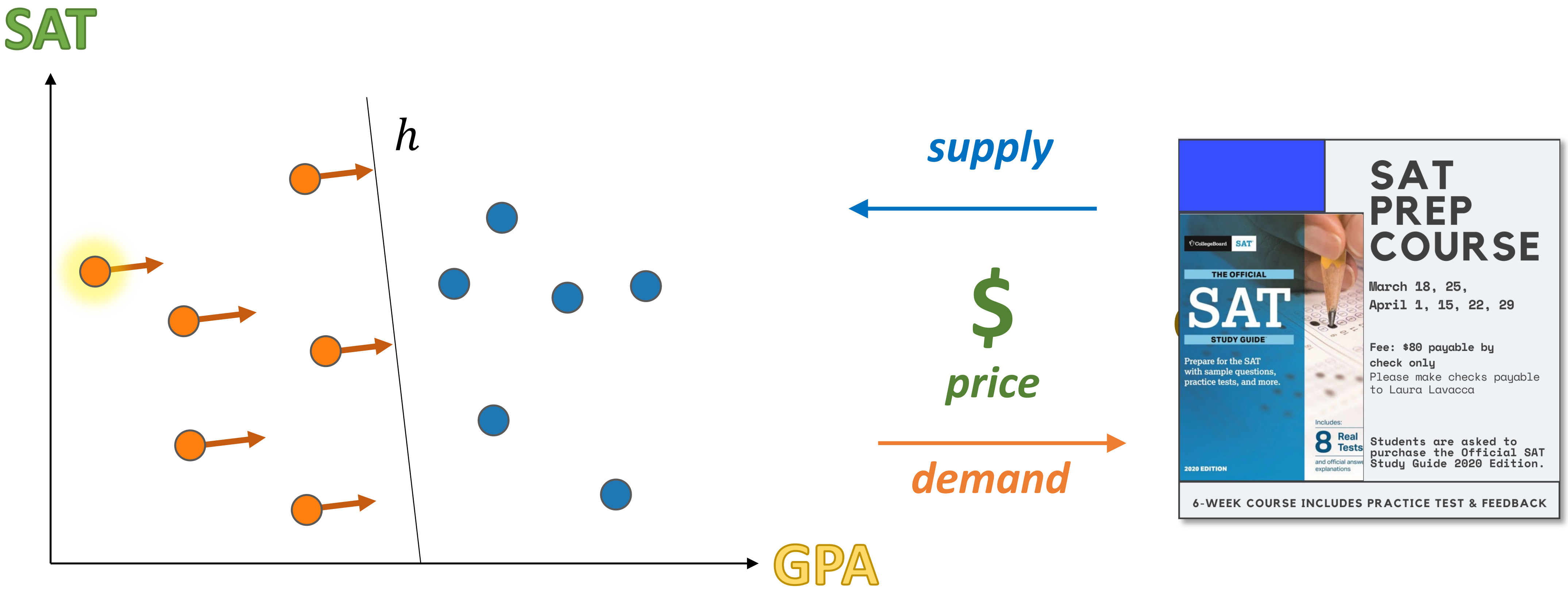


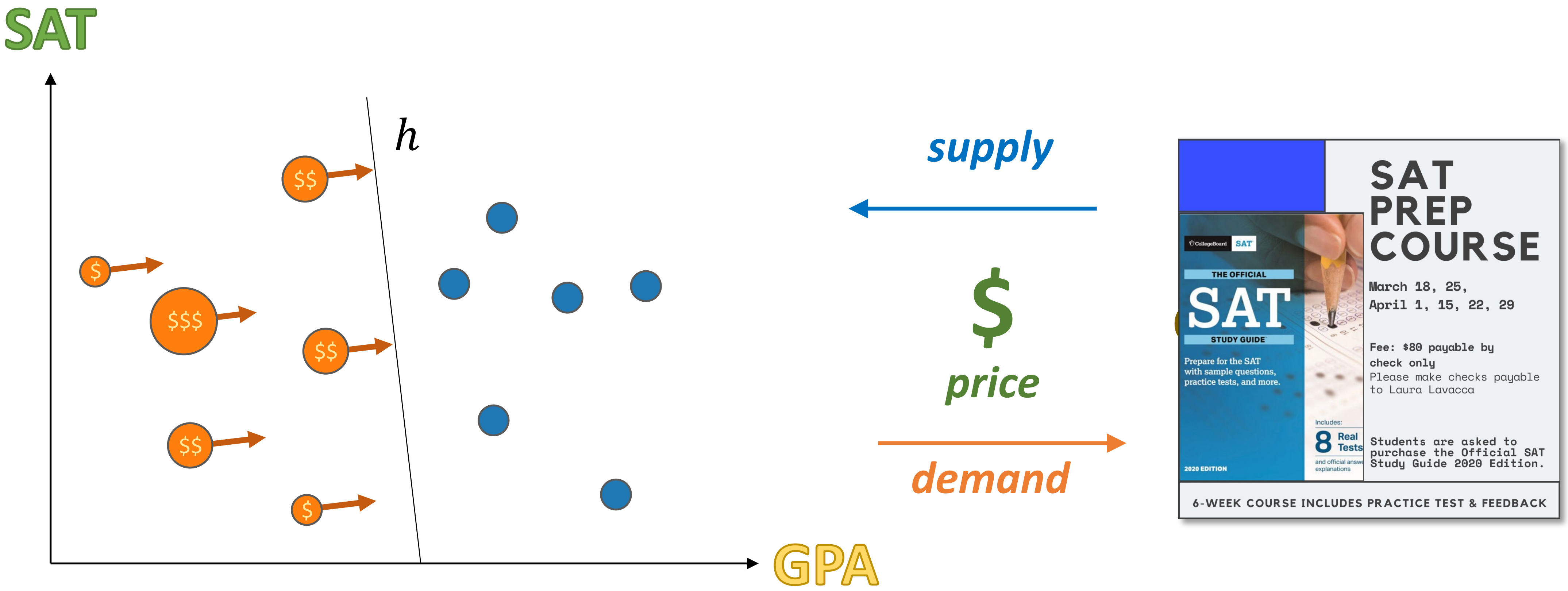


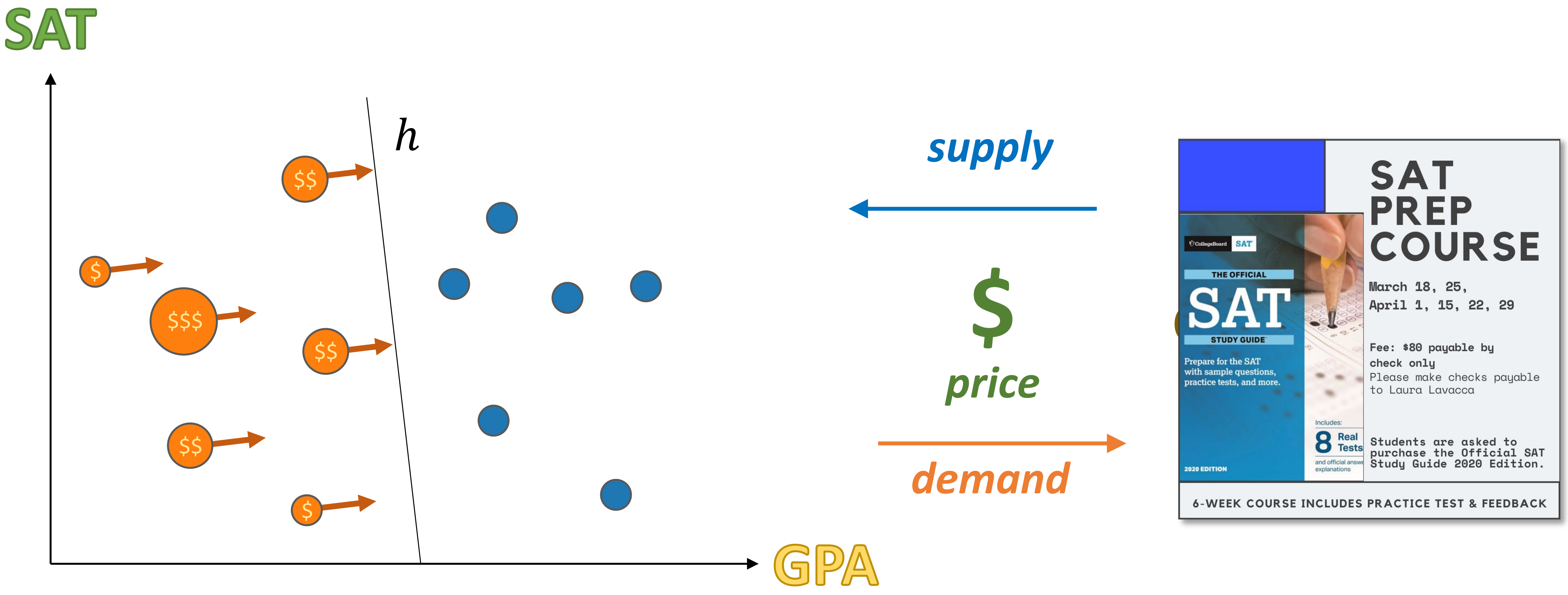
$$\Delta_h(x) = \underset{x'}{\operatorname{argmax}}\ h(x') - c(x, x')$$

utility   price of prep courses

claim: classifier induce markets!


SAT
h
GPA
supply
$
price
demand
SAT PREP COURSE
March 18, 25,
April 1, 15, 22, 29
Fee: $80 payable by check only
Please make checks payable to Laura Lavacca
Students are asked to purchase the Official SAT Study Guide 2020 Edition.
6-WEEK COURSE INCLUDES PRACTICE TEST & FEEDBACK
CollegeBoard SAT
THE OFFICIAL
SAT
STUDY GUIDE
Prepare for the SAT with sample questions, practice tests, and more.
Includes:
8 Real Tests
2020 EDITION


SAT
h
$$
$
$$$
$$
$$
$
GPA
supply
$
price
demand
SAT PREP COURSE
March 18, 25,
April 1, 15, 22, 29
Fee: $80 payable by check only
Please make checks payable to Laura Lavacca
Students are asked to purchase the Official SAT Study Guide 2020 Edition.
6-WEEK COURSE INCLUDES PRACTICE TEST & FEEDBACK
THE OFFICIAL
SAT
STUDY GUIDE
Prepare for the SAT with sample questions, practice tests, and more.
2020 EDITION

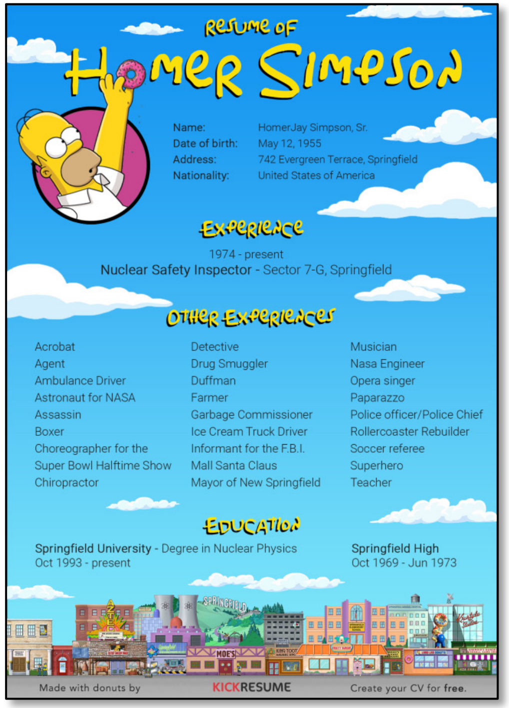


ask: can learning anticipate and account for the markets it induces?

*strategic **modification**:*

$$\underset{h}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h(\Delta_h(x_i))\right)$$

$$\text{s.t.} \;\; \Delta_h(x) = \underset{x'}{\operatorname{argmax}} \; h(x') - c(x, x')$$


Resume of
Homer Simpson

Name:	HomerJay Simpson, Sr.
Date of birth:	May 12, 1955
Address:	742 Evergreen Terrace, Springfield
Nationality:	United States of America

Experience

1974 - present
Nuclear Safety Inspector - Sector 7-G, Springfield

Other Experiences

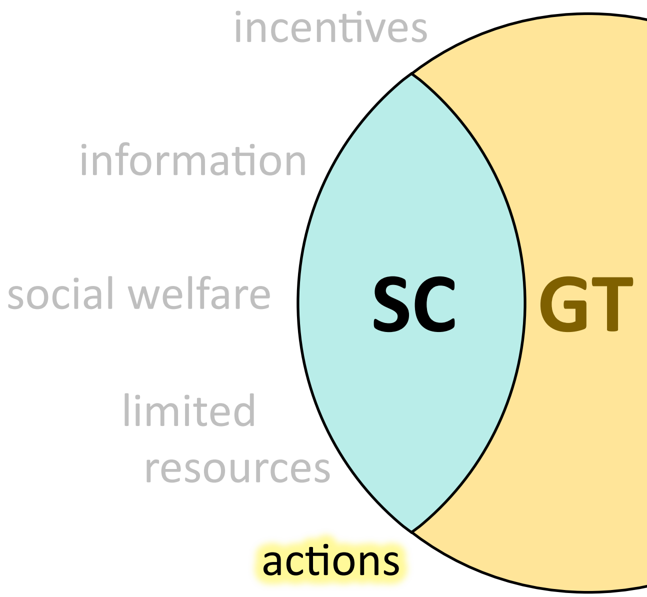
Acrobat	Detective	Musician
Agent	Drug Smuggler	Nasa Engineer
Ambulance Driver	Duffman	Opera singer
Astronaut for NASA	Farmer	Paparazzo
Assassin	Garbage Commissioner	Police officer/Police Chief
Boxer	Ice Cream Truck Driver	Rollercoaster Rebuilder
Choreographer for the Super Bowl Halftime Show	Informant for the F.B.I.	Soccer referee
	Mall Santa Claus	Superhero
Chiropractor	Mayor of New Springfield	Teacher

Education

Springfield University - Degree in Nuclear Physics
Oct 1993 - present

Springfield High
Oct 1969 - Jun 1973

Made with donuts by KICKRESUME Create your CV for free.



ask: what other actions can users take?

strategic ***modification:***

$$\underset{h}{\operatorname{argmin}} \frac{1}{m}\sum_{i=1}^{m} \ell\left(y_i, h(\Delta_h(x_i))\right)$$

strategic ***participation:***

$$\underset{h}{\operatorname{argmin}} \frac{1}{m}\sum_{i=1}^{m} a_h(x_i)\ell(y_i, h(x_i))$$

s.t. $a_h(x) = \mathbb{1}\{\text{worthwhile to apply}\}$

check out our poster!

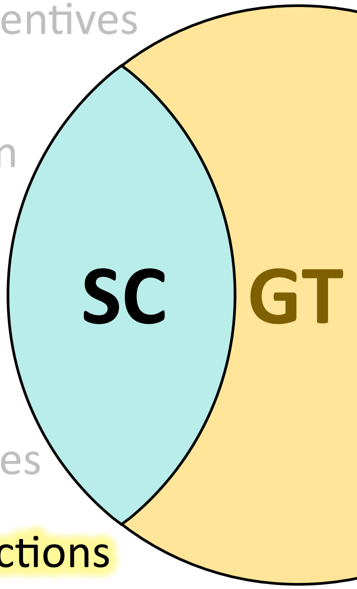
incentives

information

social welfare

limited resources

actions



ask: what other actions can users take?

*strategic **modification**:*

$$\underset{h}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} \ell\left(y_i, h(\Delta_h(x_i))\right)$$

*strategic **participation**:*

$$\underset{h}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} a_h(x_i)\ell\left(y_i, h(x_i)\right)$$

s.t. $a_h(x) = \mathbb{1}\{\text{worthwhile to apply}\}$

check out our poster!

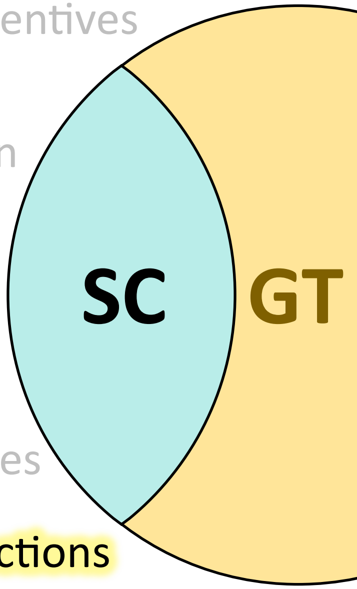
incentives

information

social welfare

limited resources

actions



ask: what other actions can users take?

strategic participation:

test

$y = 1$

$y = 0$

(interview, trial period, …)

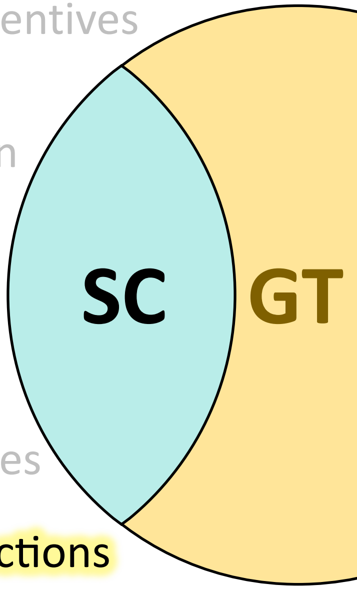
incentives

information

social welfare

limited resources

actions



strategic participation:

screen **test** **utility**

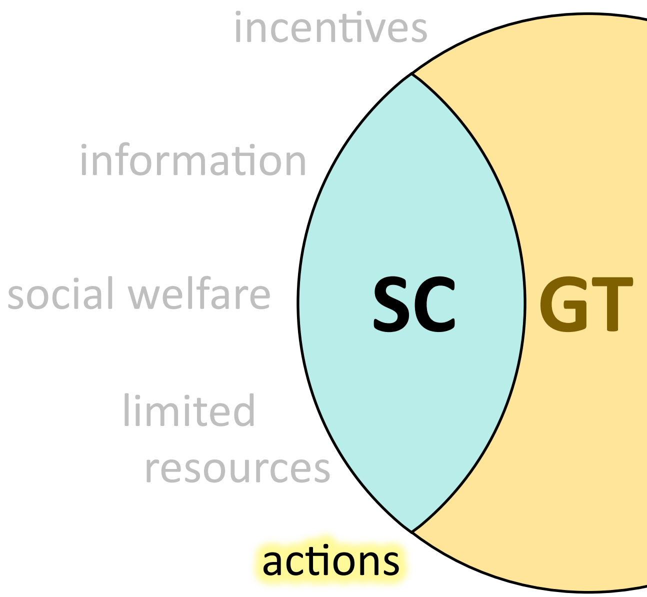
cost of test

$\hat{y} = 1$ → $y = 1$ → $1 - c$ *hired*

$\hat{y} = 0$ → $y = 0$ → $-c$ *not hired*

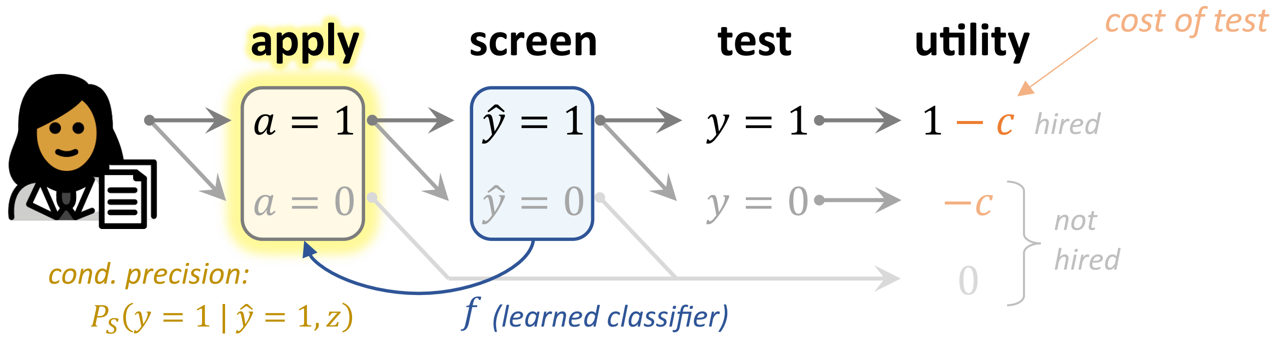
→ 0

f *(learned classifier)*

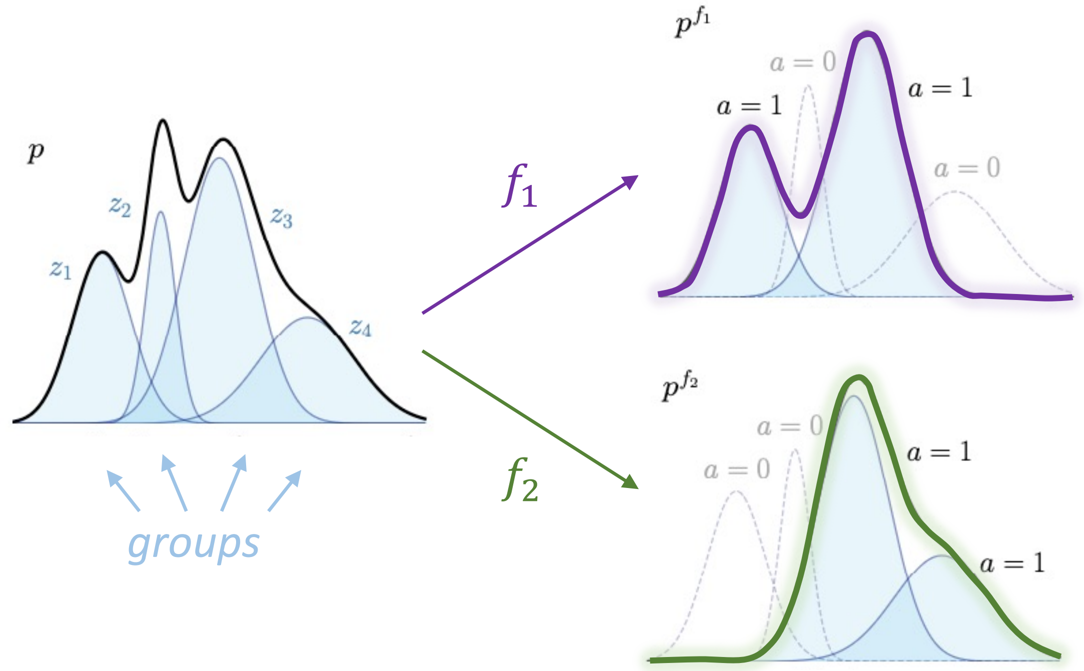


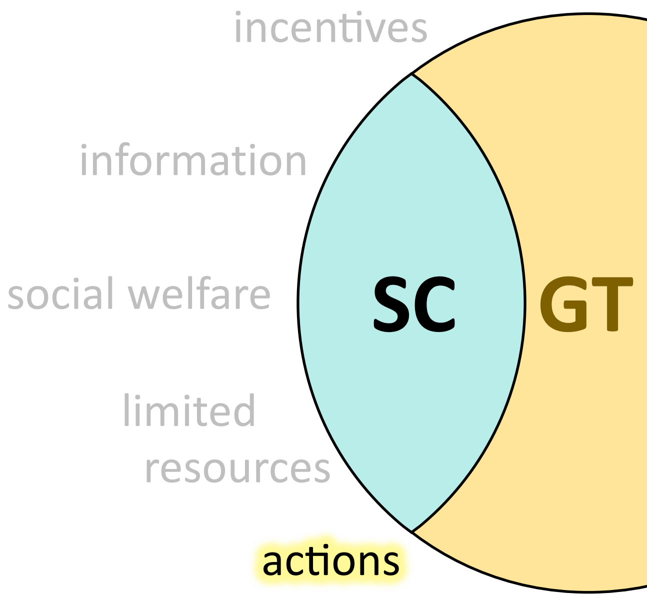
ask: how does learning affect applications?

strategic participation:



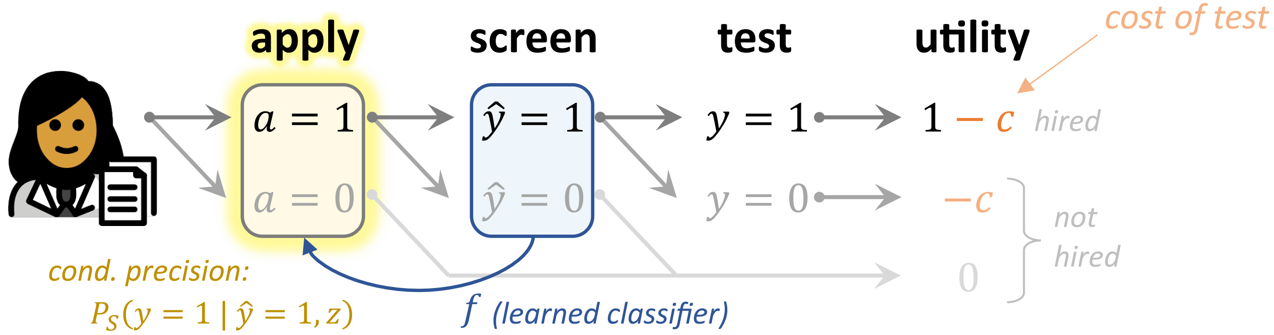
- **Observation:** learning rule determines **self-selection**



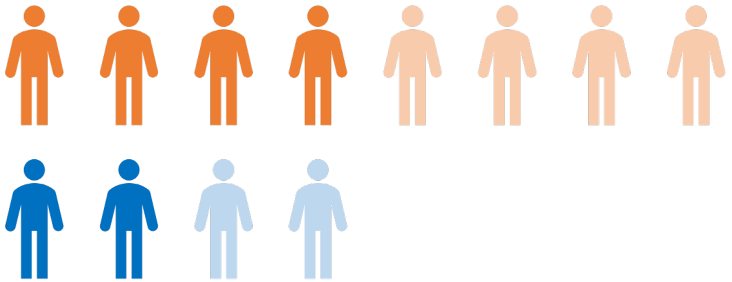


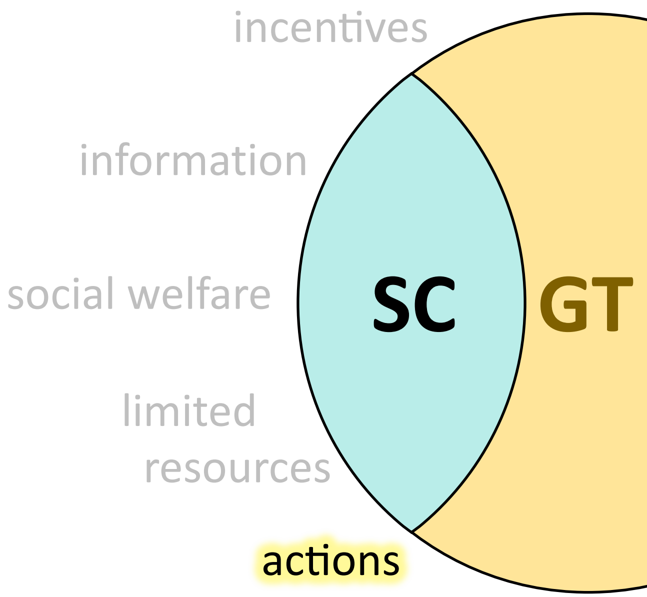
main result: learning has capacity to fully determine applications!

strategic participation:



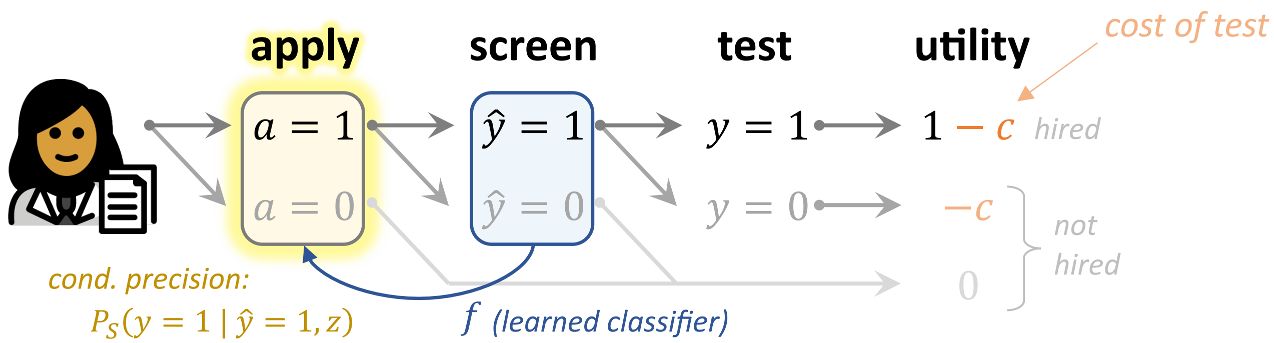
- **Observation:** learning rule determines **self-selection**
- **Implications**: can create *false appearance* of fairness, discriminates by making application too costly/risky



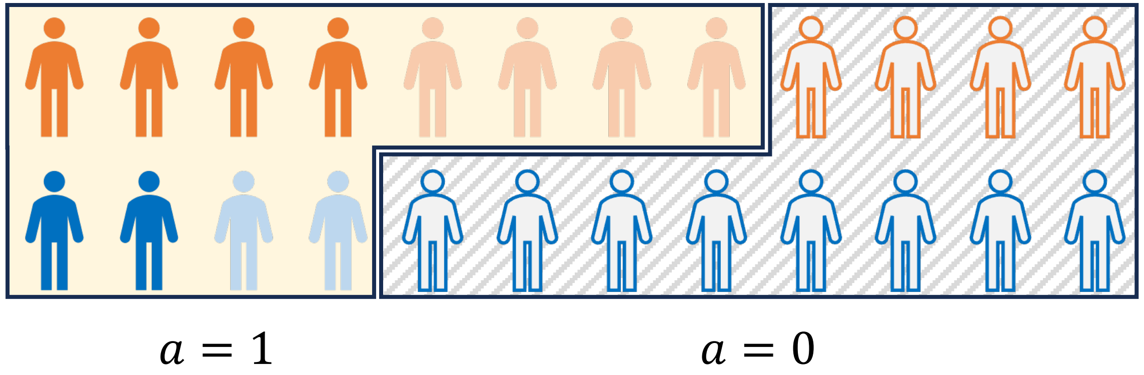


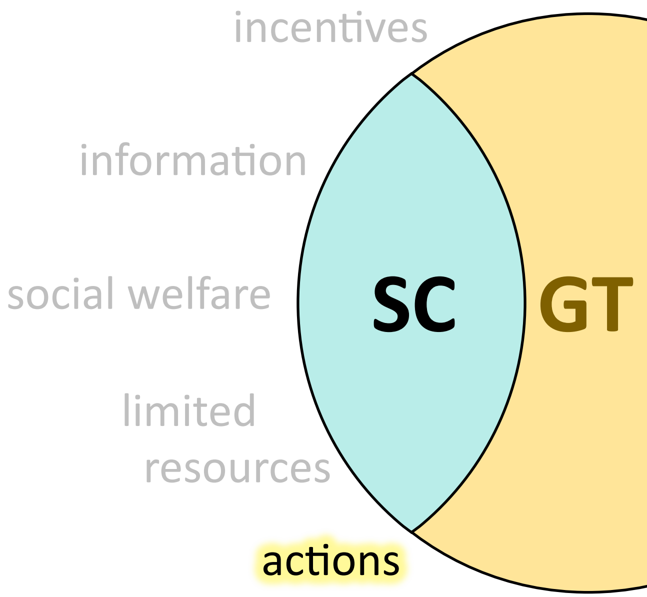
ask: how does learning affect applications?

strategic participation:



- **Observation:** learning rule determines **self-selection**
- **Implications**: can create *false appearance* of fairness, discriminates by making application too costly/risky

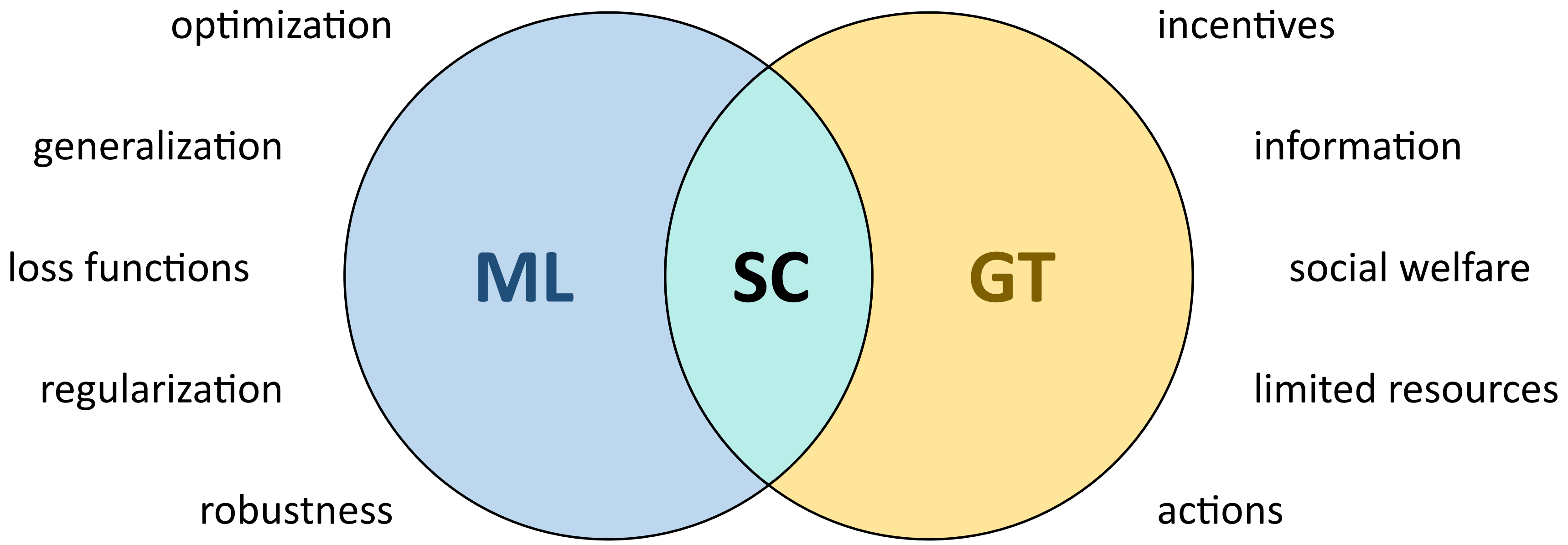


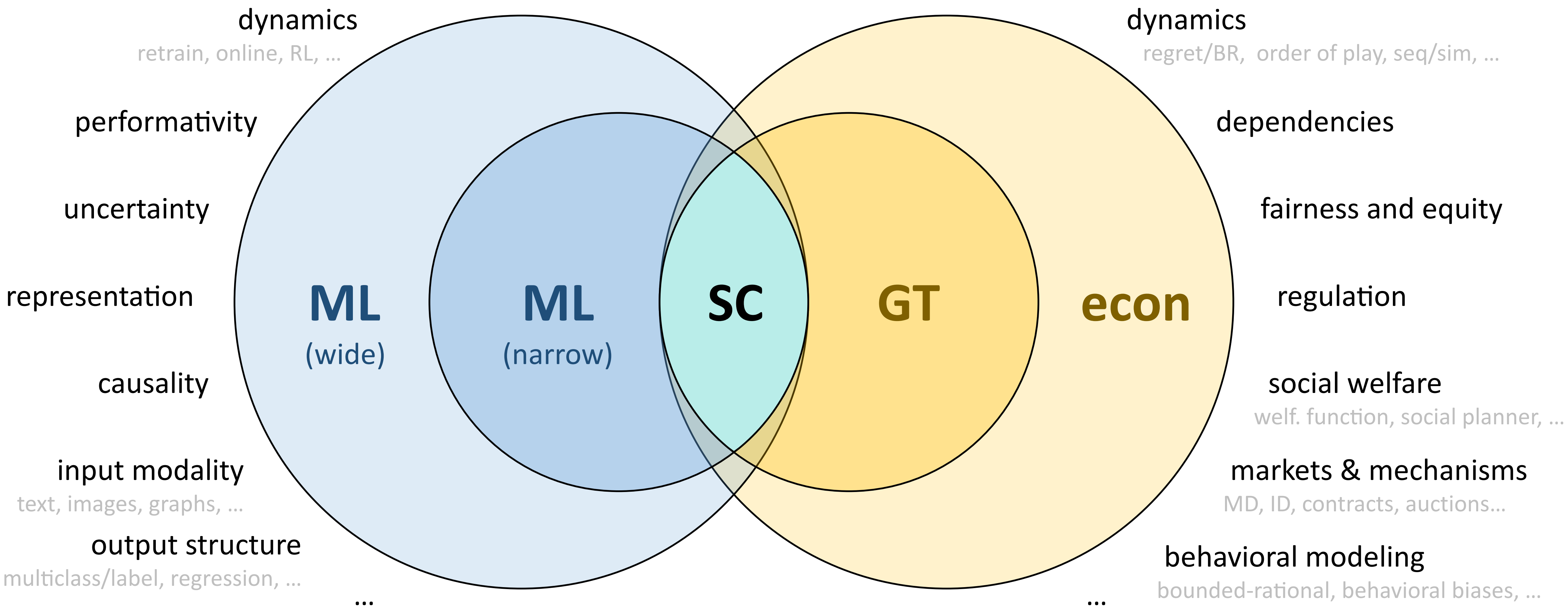


ask: how does learning affect applications?

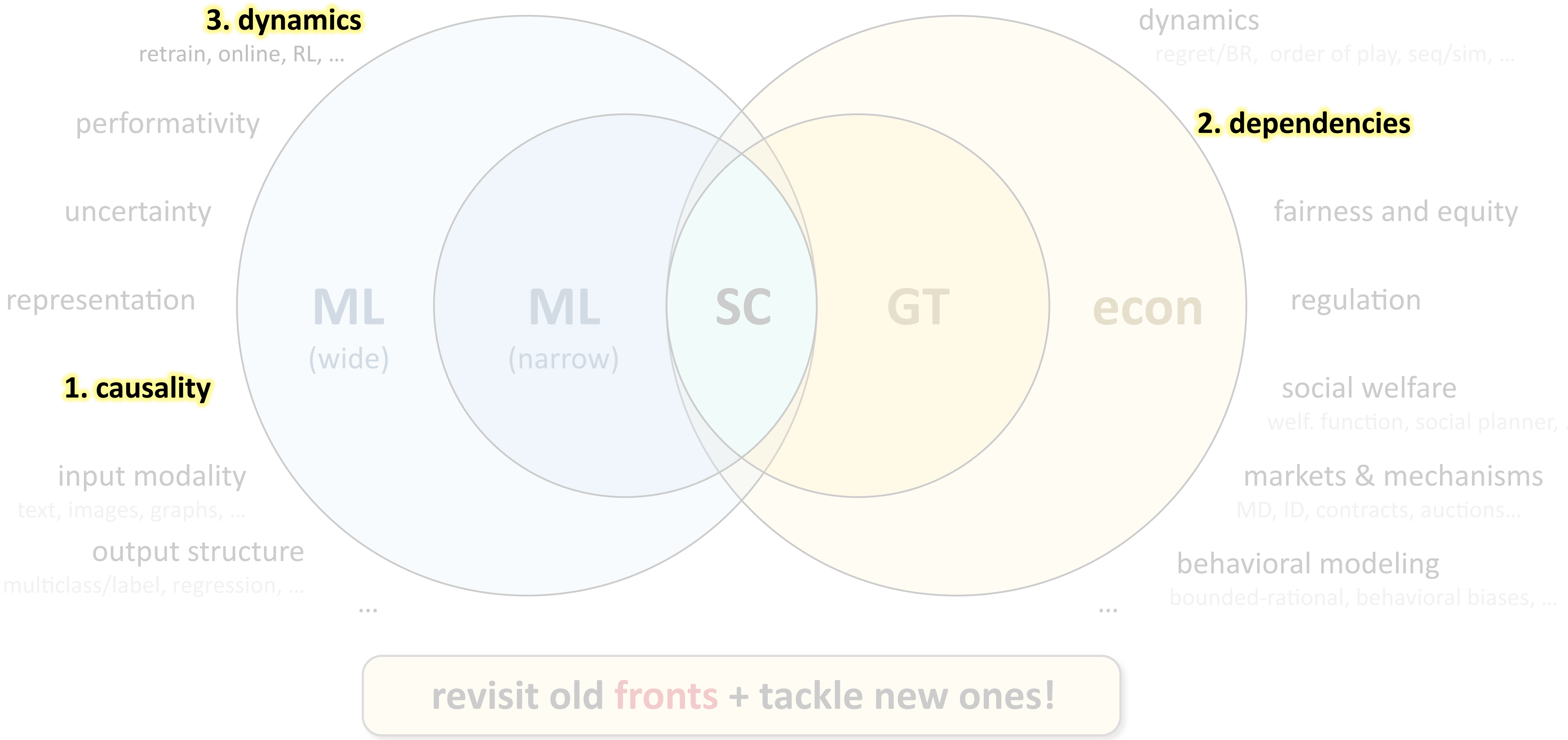
Beyond

the standard setup


optimization
generalization
loss functions
regularization
robustness
ML
SC
GT
incentives
information
social welfare
limited resources
actions



revisit old fronts + tackle new ones!

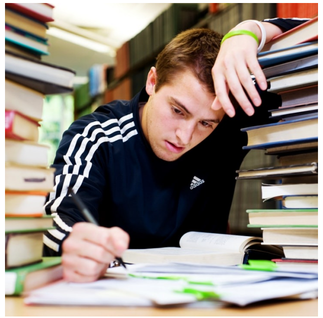

3. dynamics
retrain, online, RL, ...
performativity
uncertainty
representation
1. causality
input modality
text, images, graphs, ...
output structure
multiclass/label, regression, ...
...
ML
(wide)
ML
(narrow)
SC
GT
econ
dynamics
regret/BR, order of play, seq/sim, ...
2. dependencies
fairness and equity
regulation
social welfare
welf. function, social planner, ...
markets & mechanisms
MD, ID, contracts, auctions...
behavioral modeling
bounded-rational, behavioral biases, ...
...
revisit old fronts + tackle new ones!

1) Causality

vanilla SC

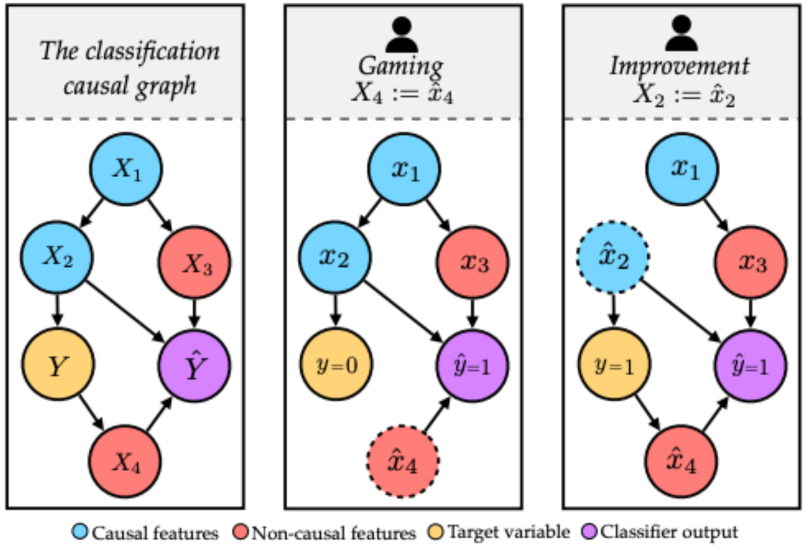
superficial changes
⇒ gaming

<u>causal</u> SC



1) Causality

- **Standard SC:** changing x does <u>not</u> affect y (=gaming)
- **More realistic**: changing x can also change y
- Assume exists underlying *causal graph* [Pearl 2009]:



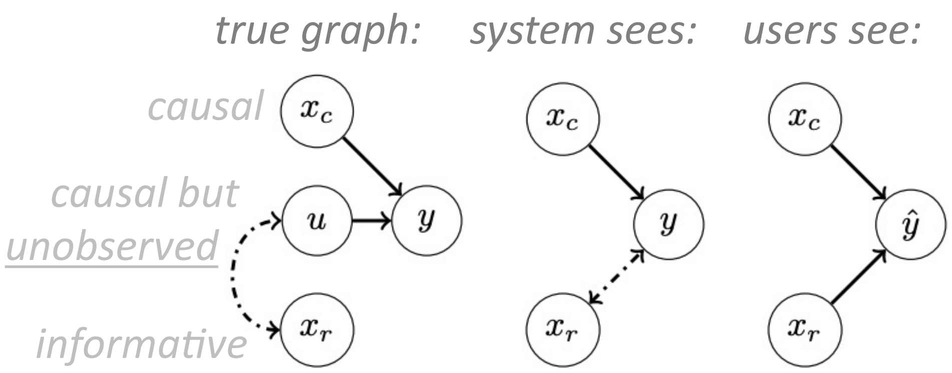
(taken from Miller et al. 2020)

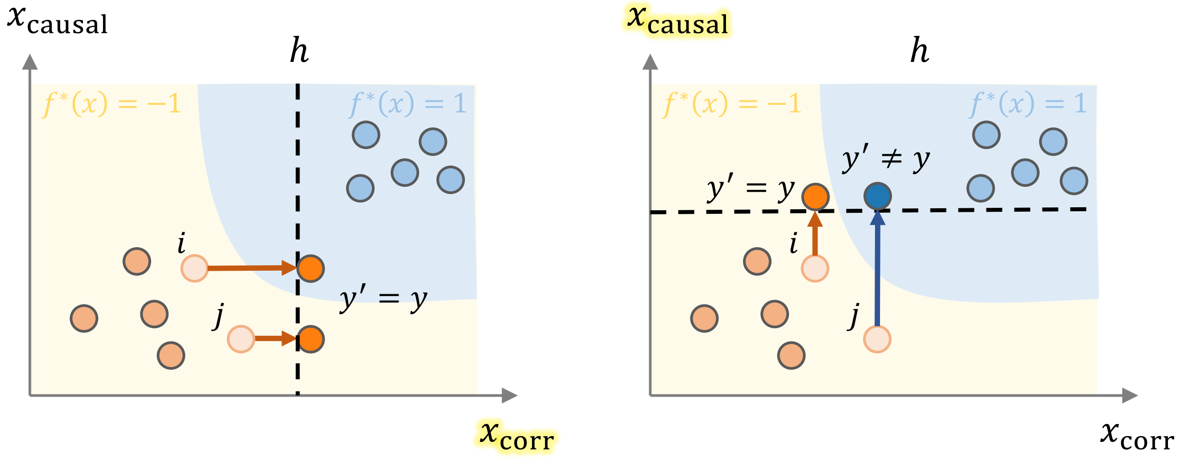
ask: can we learn in causal strategic settings?

- **Lots of challenges:**
  - ➢ graph not necessarily known
  - ➢ key variables not necessarily observed (e.g., confounders)
  - ➢ structure determines interactions (i.e., what affects what)
- Causal SC is **inherently difficult** – as hard as *causal inference* [MMH ICML20]

Causal SC as distribution shift

- **Q1**: How does causality affect learning?
- Simplifying assumption: causal vs. correlative features
- **A1**: Entails different **types of distribution shift**:
    - correlative → *strategic* shift → gaming
    - only causal → *covariate* shift → missinformation
    - both → *mixture* shift → interactions
- **Corollary**: choose your battles!



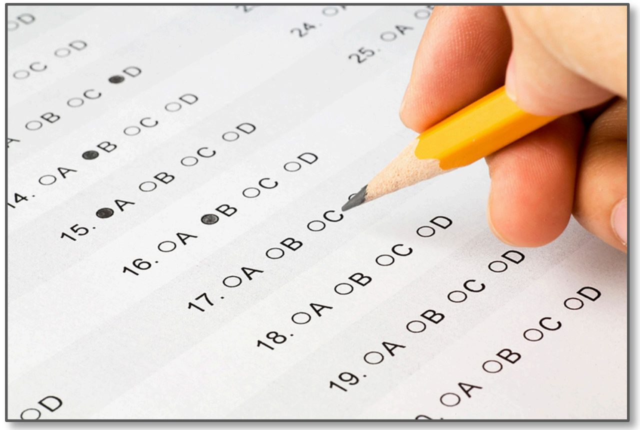


Incentivizing improvement

- **Q1**: How does causality affect learning?
- **Q2**: How does causality affect social outcomes?
- **A2**: Causal SC has **potential for *improvement*:**

$$\mathbb{E}_x\big[\mathbb{E}[p(y \mid \mathrm{do}(\Delta)) - p(y) \mid x]\big]$$

- **Goal**: learn h that (also) promotes improvement
- Has long and rich history in economics (e.g., see [KR 19])
- Also considered in (online) SC
  (e.g., [SEA'20, BLWZ'21, CWL'21, HNSHW'22, MDW'22])



Incentivizing improvement

- **Q1**: How does causality affect learning?
- **Q2**: How does causality affect social outcomes?
- **A2**: Causal SC has **potential for *improvement*:**

$$\mathbb{E}_x\left[\mathbb{E}[p(y \mid \mathrm{do}(\Delta)) - p(y) \mid x]\right]$$

- **Goal**: learn h that (also) promotes improvement
- Has long and rich history in economics (e.g., see [KR 19])
- Also considered in (online) SC
  (e.g., [SEA'20, BLWZ'21, CWL'21, HNSHW'22, MDW'22])

- But changing x can also **impair** outcomes!

Incentivizing improvement

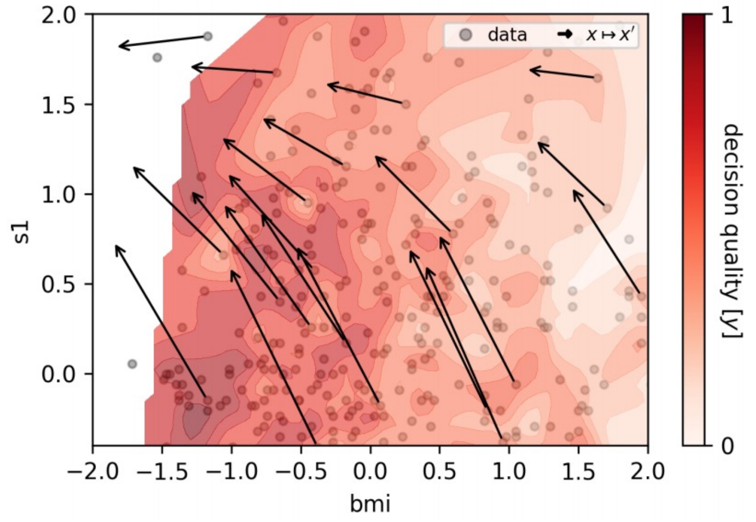
- **Q1**: How does causality affect learning?
- **Q2**: How does causality affect social outcomes?
- **A2**: Causal SC has **potential for *improvement*:**

$$\mathbb{E}_x\left[\mathbb{E}[p(y \mid \mathrm{do}(\Delta)) - p(y) \mid x]\right]$$

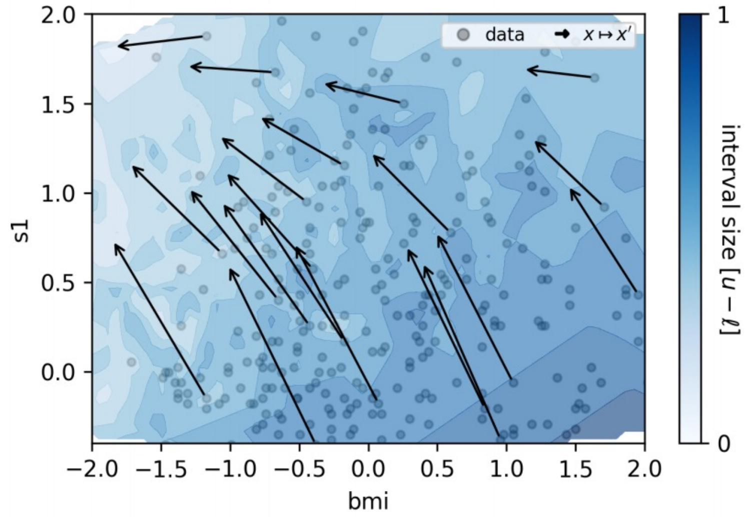
- **Goal**: learn h that (also) promotes improvement
- Has long and rich history in economics (e.g., see [KR 19])
- Also considered in (online) SC
  (e.g., [SEA'20, BLWZ'21, CWL'21, HNSHW'22, MDW'22])

- But changing x can also **impair** outcomes!
- **Solution**: learn safe models by "looking ahead"

causal effect:



uncertainty:



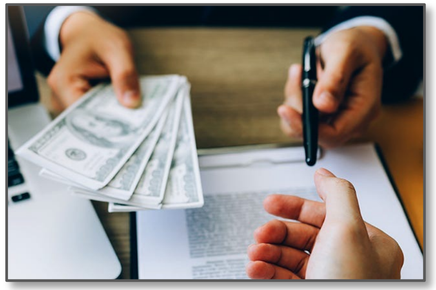
RHRP NeurIPS20

2) Dependencies: How users relate

- **Standard SC:** responses are **independent** ($\Delta_f(x)$ depends only on x)
- **More realistic**: responses are **interdependent**

- **Reason #1**: limited resources
  - Actually, all common examples have limit on # of $\hat{y} = 1$
  - This means that users **compete**

limited financial resources



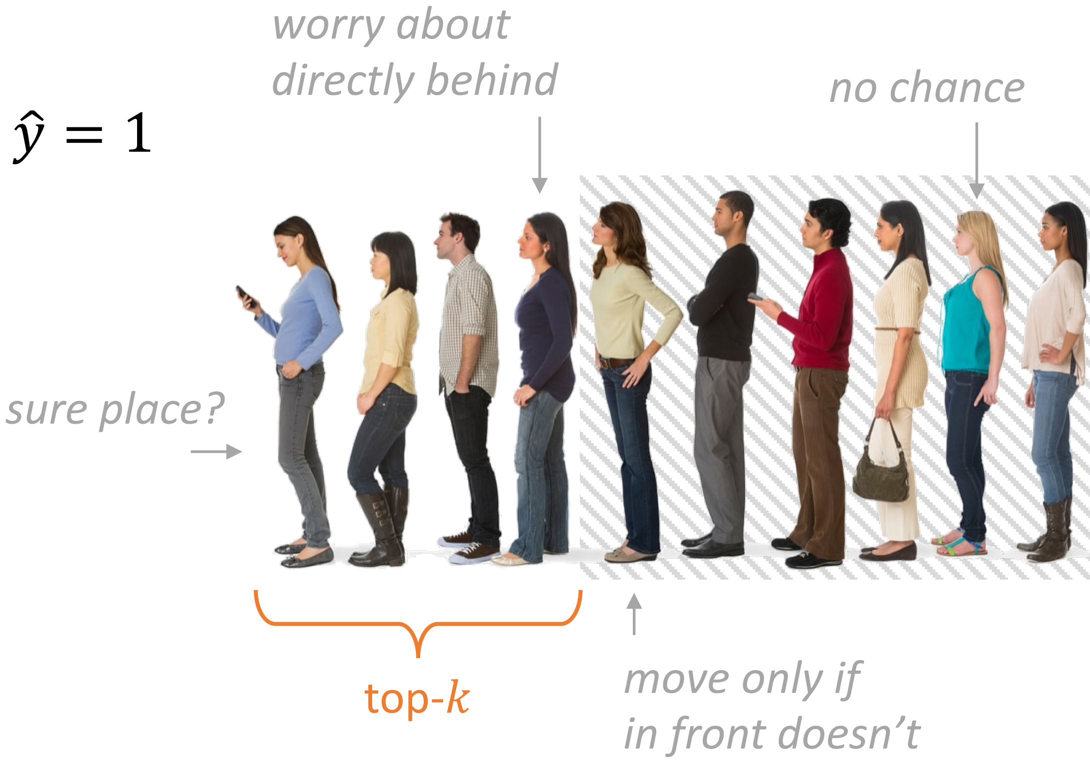
limited teaching capacity

limited qualified personell

2) Dependencies: How users relate

- **Standard SC:** responses are **independent** ($\Delta_f(x)$ depends only on x)
- **More realistic**: responses are **interdependent**

- **Reason #1:** limited resources
  - Actually, all common examples have limit on # of $\hat{y} = 1$
  - This means that users **compete**
  - **Reasonable approach**:
    learn to **rank**, then set $\hat{y} = 1$ only for **top-k**
  - Turns out to be *exceedingly hard* [LGB ICML22]
  - Still – major goal!

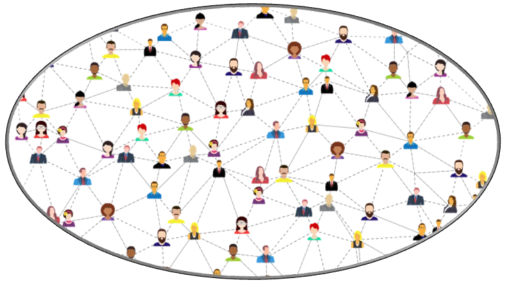


2) Dependencies: How users relate

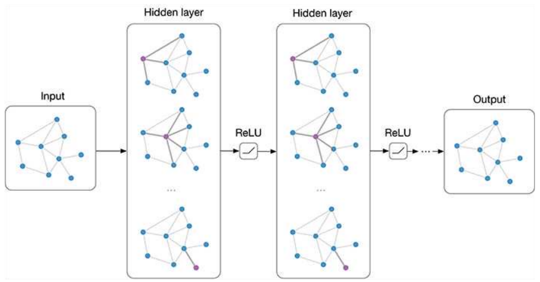
- **Standard SC:** responses are **independent**
- **More realistic**: responses are **interdependent**

- **Reason #1:** limited resources
- **Reason #2:** model-induced dependencies

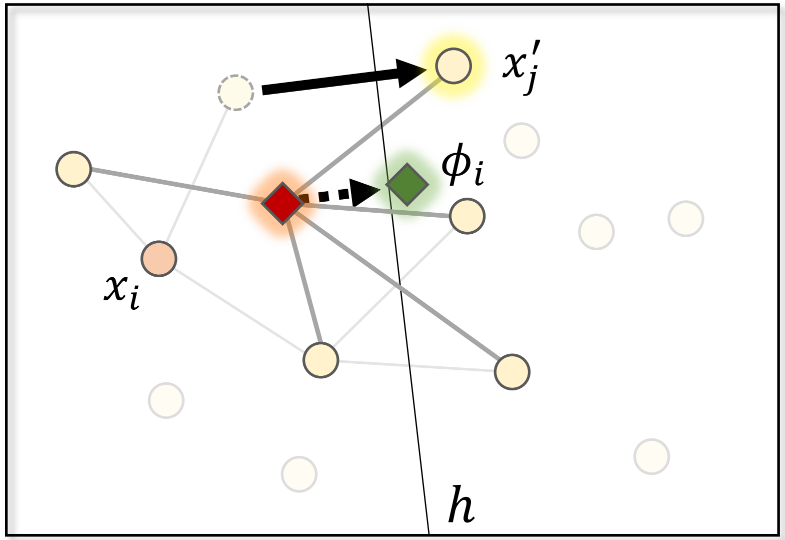
using graph in learning creates dependencies



social network



GNN



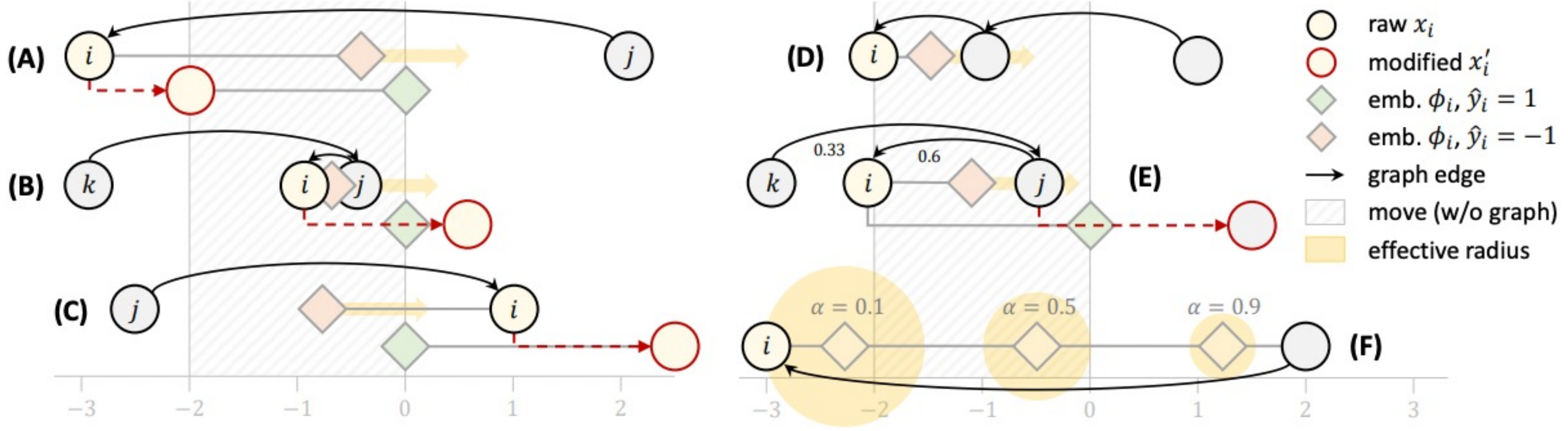
graph-dep. embedding

2) Dependencies: How users relate

- **Standard SC:** responses are **independent**
- **More realistic**: responses are **interdependent**

using graph in learning
creates surprising dependencies

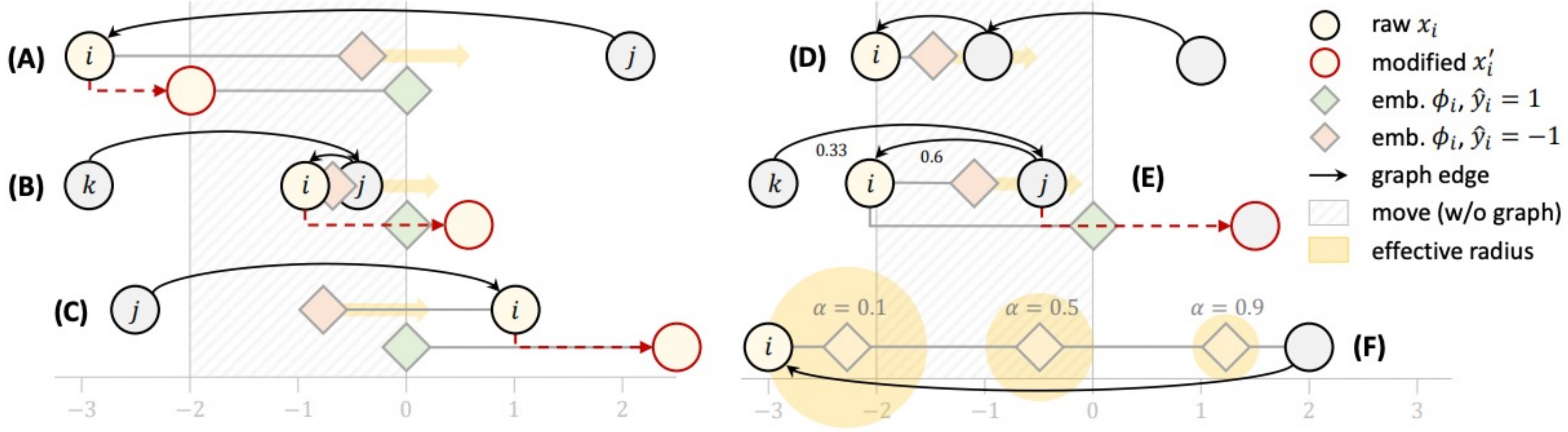
- **Reason #1:** limited resources
- **Reason #2:** model-induced dependencies



2) Dependencies: How users relate

- **Standard SC:** responses are **<u>ind</u>ependent**
- **More realistic**: responses are **<u>inter</u>dependent**

- **Reason #1:** limited resources
- **Reason #2:** model-induced dependencies

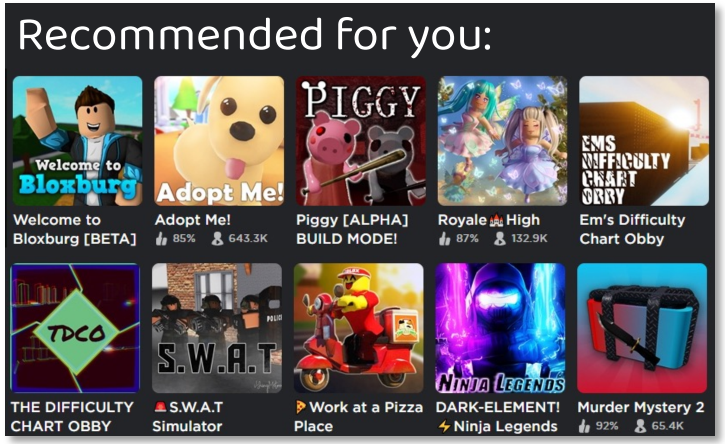


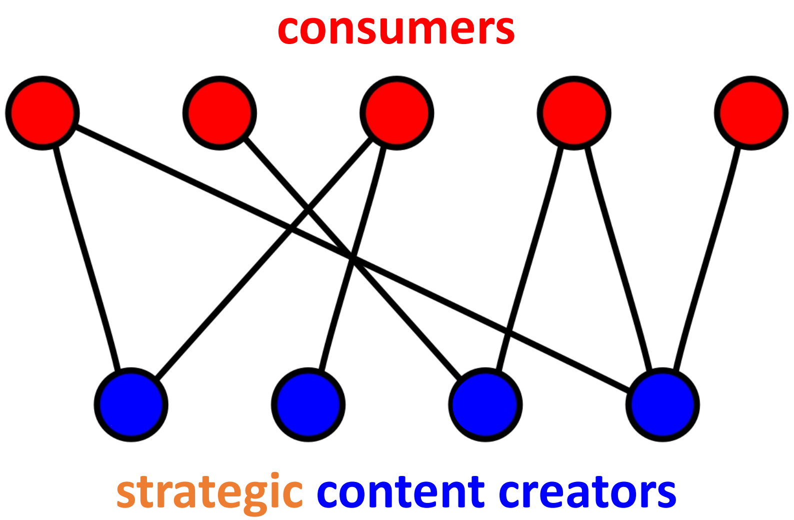
2) Dependencies: How users relate

- **Standard SC:** responses are **independent**
- **More realistic**: responses are **interdependent**

- **Reason #1:** limited resources
- **Reason #2:** model-induced dependencies
- **Reason #3:** economic graph structure

main result: can use graph to incentiveize **diversity**





3) Learning over time

- **Standard SC:** batch setting: train → deploy → test
- Assumes access to **clean data** (otherwise, chicken & egg!)
- **More realistic:** data is dirty (i.e., result of some behavior)

- **One solution:** iterated deployments over time: train → deploy → train → deploy → train → …
- **Three main aproaches:** → *lots of research; will present here only in brief*
  1. online learning (e.g., bandits)
     (e.g., [DRSWW NeurIPS17, CSSVZ ICML23, HPW NeurIPS23, SBM NeurIPS23, ABBN EC21, …])
  2. performative prediction (retraining revisited) [PZMH ICML20]
  3. dynamical systems

3) Learning over time

- **Standard SC:** batch setting: train → deploy → test
- Assumes access to **clean data** (otherwise, chicken & egg!)
- **More realistic:** data is dirty (i.e., result of some behavior)

- **One solution:** iterated deployments over time: train → deploy → train → deploy → train → …
- **Pros**: less restrictive
  1. does <u>not</u> require clean data
  2. does not assume known Δ_h (or even best-response)
  3. permits causal Δ_h (under additional assumptions)
- **Cons**: each deployment is social "experiment"
  - in some cases, exploration is reasonable
  - in other cases – it is very much <u>not</u>

Opportunities & challenges

open questions

Open questions

- Strategic learning is exiting new field with much potential for growth
- But it is also young – so that many challenges still lie ahead:

1. **Learning aspects**:

Open questions

- Strategic learning is exiting new field with much potential for growth
- But it is also young – so that many challenges still lie ahead:

1. **Learning aspects**:
   - labels beyond binary
     - regression
     - multiclass
     - multilabel
     - sequences
     - structured (e.g., graphs)
     - …

Open questions

- Strategic learning is exiting new field with much potential for growth
- But it is also young – so that many challenges still lie ahead:

1. **Learning aspects**:
   - labels beyond binary
   - inputs beyond vectors
     - images
     - text
     - graphs
     - …

Open questions

- Strategic learning is exiting new field with much potential for growth
- But it is also young – so that many challenges still lie ahead:

1. **Learning aspects**:
    - labels beyond binary
    - inputs beyond vectors
    - models beyond linear
        - neural nets (behavior in latent space)
        - tree-based
        - text-based (prompts)
        - …

Open questions

- Strategic learning is exiting new field with much potential for growth
- But it is also young – so that many challenges still lie ahead:

1. **Learning aspects**:
    - labels beyond binary
    - inputs beyond vectors
    - models beyond linear
    - settings beyond classification
        - unsupervised and semi-supervised
        - generative
        - RL and MARL
        - …

Open questions

- Strategic learning is exiting new field with much potential for growth
- But it is also young – so that many challenges still lie ahead:

1. **Learning aspects**
2. **Econ/GT aspects:**
    - information
        - power
        - control
        - selective release/withold
        - …

Open questions

- Strategic learning is exiting new field with much potential for growth
- But it is also young – so that many challenges still lie ahead:

1. **Learning aspects**
2. **Econ/GT aspects:**
    - information
    - other economic settings
        - markets, auctions, contracts, …
        - competition (between classifiers)
        - cooperation (between users)
        - monopolistic behavior
        - …

Open questions

- Strategic learning is exiting new field with much potential for growth
- But it is also young – so that many challenges still lie ahead:

1. **Learning aspects**
2. **Econ/GT aspects:**
   - information
   - other economic settings
   - behavior
     - Bayesian
     - non-rational “behavioral” (=biases)
     - …

loss aversion *confirmation bias* *causal fallacy* *future discounting* *decoy effect* *primacy/recency* *bounded rationality* *anchoring* *endowment effect* *risk aversion/seeking* *choice overload* *availability bias* *k-level reasoning* *quantal response* *base rate fallacy* *bandwagon effect* *framing/priming* …

Open questions

- Strategic learning is exiting new field with much potential for growth
- But it is also young – so that many challenges still lie ahead:

1. **Learning aspects**
2. **Econ/GT aspects**
3. **"In the wild":**
    - evaluation [BBK 20, HHP 23, CIALRM 23]
    - measuring utility/welfare
    - estimating costs
    - monitoring and regulation

Why supervised learning?

- Most human-centric tasks are **policy problems** (vs. prediction problems)
- So supervised learning is clearly the wrong tool to use
- But it is also *by far* the most prevelant, accessible, and easy to use

- **Vision for the future:**

```
strat_clf = LogisticRegression(penalty='l2', C=0.01, max_iter=500,
                               want='yhat=1', know='noisy_h', do='game')
strat_clf.fit(X, Y, cost=c)
pred = strat_clf.predict(X_test)
```

- **Goal**: make integrating human agency as seemless as possible
- **Not so easy!** And requires much caution and deliberation (c.f. fairness)

Summary

Summary

- SC captures natural tension between learning systems and their users
- Appealing interface between ML and GT – many open question!
- Original setup is clean and simple, but likely to narrow
- Nonetheless, flexible and modular: easy to extend, relax, and generalize

Summary

- SC captures natural tension between learning systems and their users
- Appealing interface between ML and GT – many open question!
- Original setup is clean and simple, but likely to narrow
- Nonetheless, flexible and modular: easy to extend, relax, and generalize

➢ A call to rethink the design of learning algorithms for social settings

➢ An opportunity to revise foundations using economic and behavioral modeling

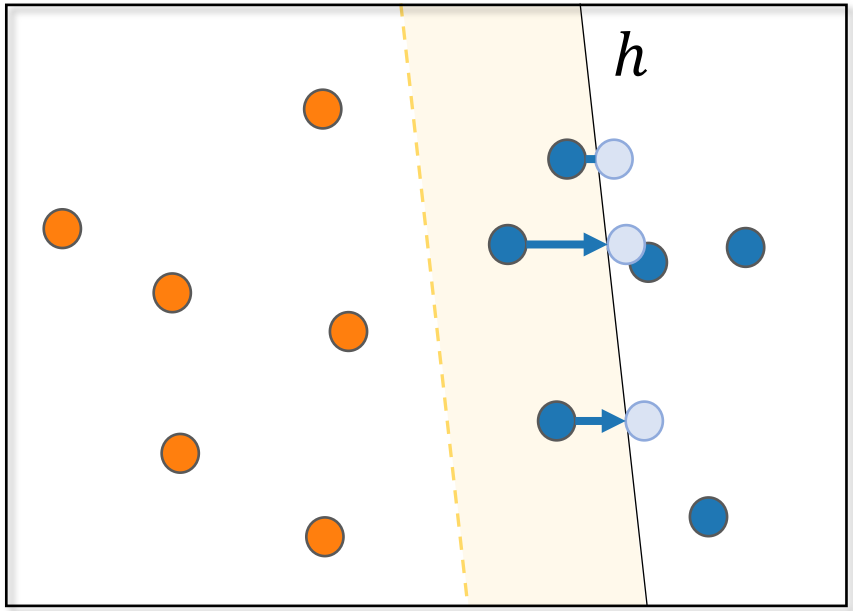
➢ High potential for real impact – much more work needed!

Narrative(s)

- “users game system”

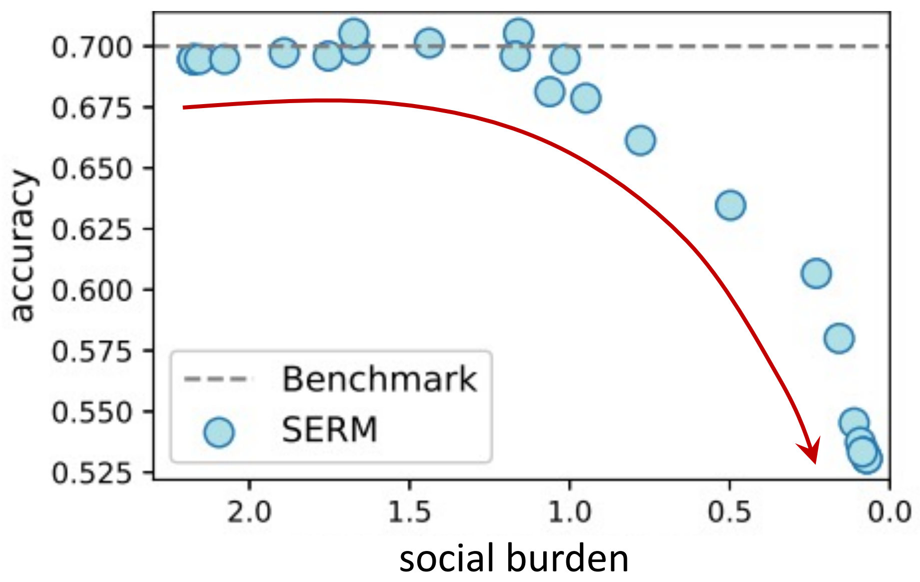
Narrative(s)

- “users game system”
- “system exploits users”



Narrative(s)

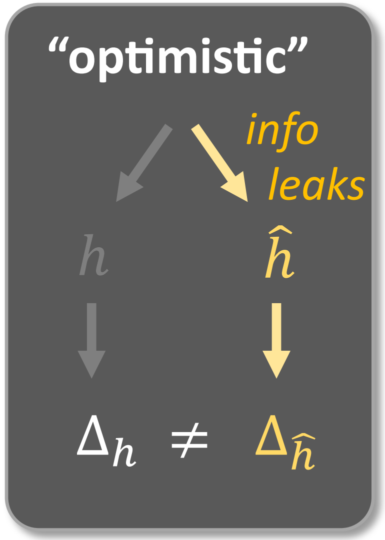
- “users game system”
- “system exploits users”
- “system exploits users *unintentionally*”



Narrative(s)

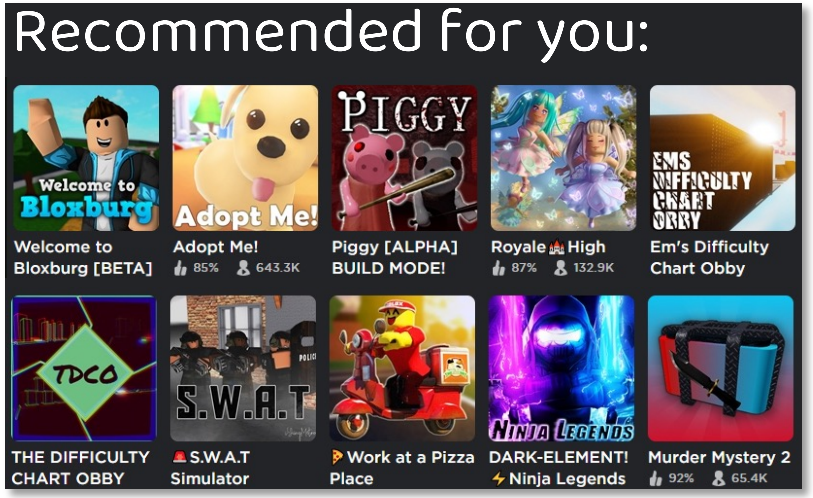
- “users game system”
- “system exploits users”
- “system exploits users *unintentionally*”
- “… as long as there is transparency”

SC in the dark



Narrative(s)

- “users game system”
- “system exploits users”
- “system exploits users *unintentionally*”
- “… as long as there is transparency”
- “potential for cooperation…”

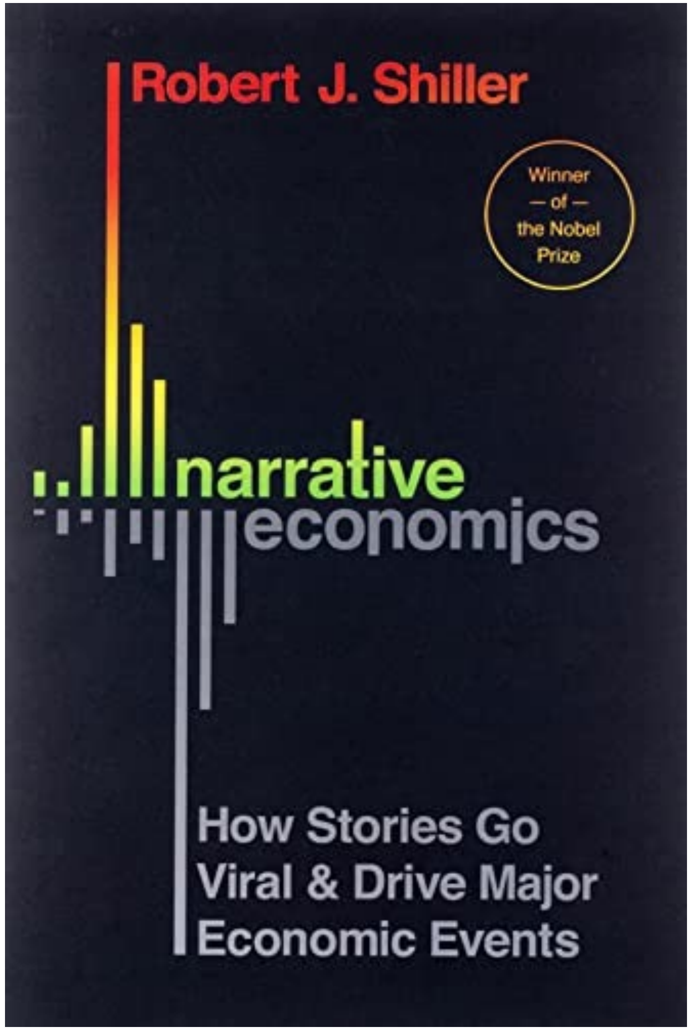


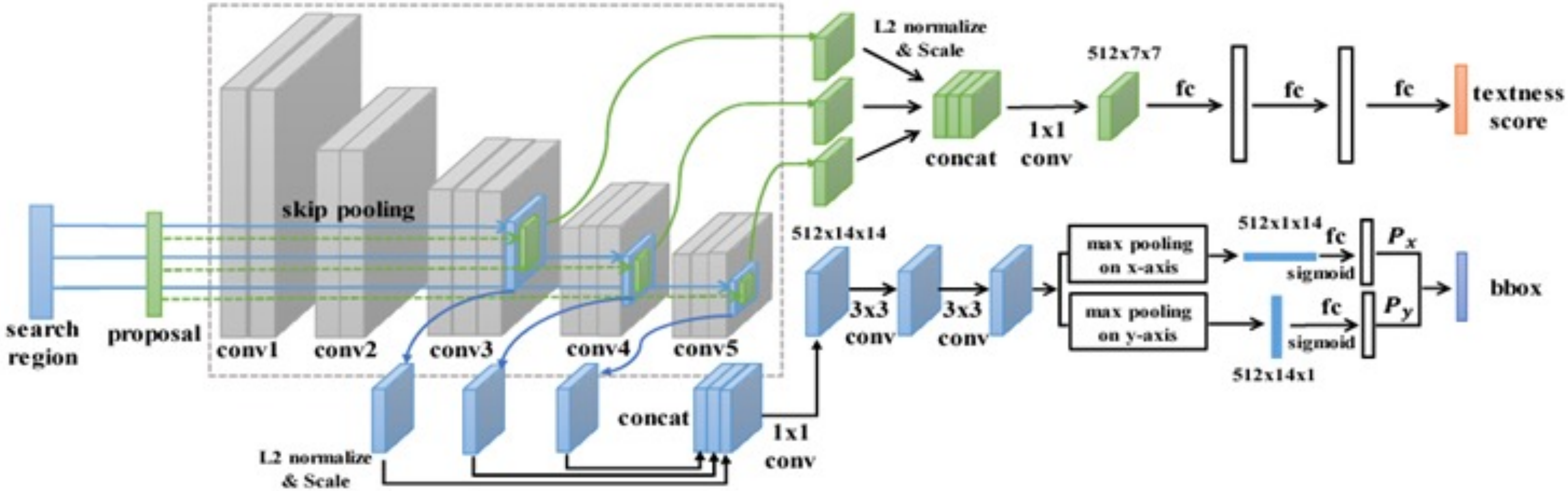
Narrative(s)

- “users game system”
- “system exploits users”
- “system exploits users *unintentionally*”
- “… as long as there is transparency”
- “potential for cooperation…”
- “its just a market”
- …

Narrative(s)

- “users game system”
- “system exploits users”
- “system exploits users *unintentionally*”
- “… as long as there is transparency”
- “potential for cooperation…”
- “its just a market”
- …





L2 normalize
& Scale
512x7x7
fc
fc
fc
textness
score
concat
1x1
conv
skip pooling
512x14x14
512x1x14
fc
P_x
max pooling
on x-axis
sigmoid
bbox
max pooling
on y-axis
fc
P_y
sigmoid
512x14x1
3x3
conv
3x3
conv
search
region
proposal
conv1
conv2
conv3
conv4
conv5
concat
1x1
conv
L2 normalize
& Scale

LES ARTISTES ASSOCIÉS S.A.B. présentent
PARLANT FRANÇAIS
105
ENFANTS ADMIS
CHARLIE CHAPLIN
DANS
UNITED 35 ARTISTS ANNIVERSARY
les TEMPS MODERNES
ECRIT, REALISÉ et PRODUIT par CHARLES CHAPLIN
MODERNE TIJDEN
UNITED ARTISTS
IMPRIMÉ EN BELGIQUE
ÉTUDES PUBLICITAIRES
IMP. J. LICHTERT & FILS - BRUXELLES


LES ARTISTES ASSOCIÉS S.A.B. présentent PARLANT FRANÇAIS ENFANTS ADMIS
CHARLIE
CHAPLIN
DANS
les TEMPS MODERNES
ECRIT, REALISÉ et PRODUIT par CHARLES CHAPLIN
MODERNE TIJDEN
UNITED ARTISTS

THANKS!