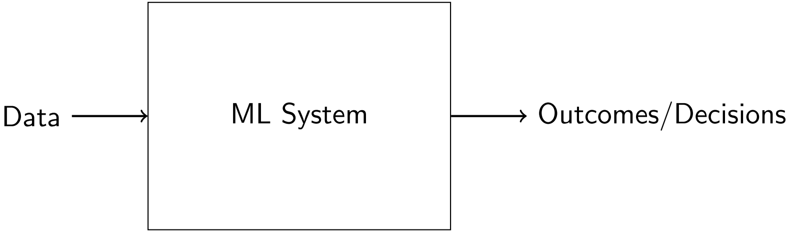
1. Introduction

2. Models of Strategic Learning

Classical ML



Classical ML

Historical Loans Data ⟶  ⟶ Loan decisions

Classical ML

Medical History data ⟶

⟶ Health Ris

Gaming

Gaming

Applications:

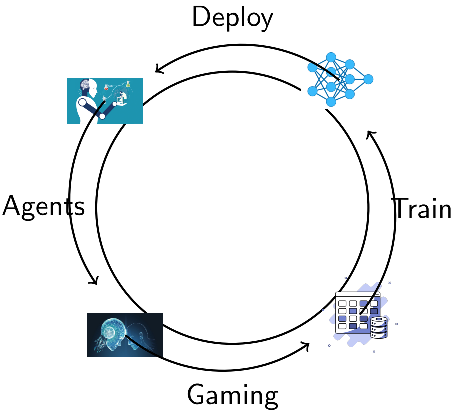
1. Health risk predictions
2. Bank loan approvals
3. Corporate hiring/promotions ...

Gaming

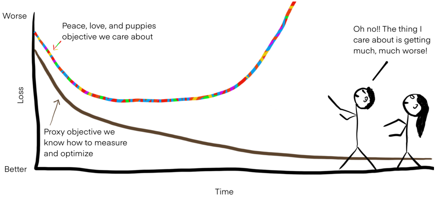
Applications:

1. Health risk predictions
2. Bank loan approvals
3. Corporate hiring/promotions ...

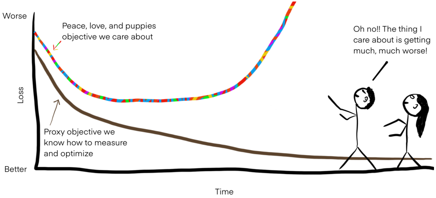
Gaming:



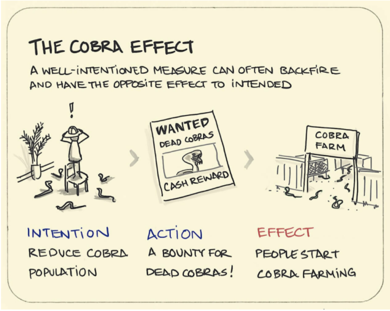
Gaming


Worse
Peace, love, and puppies
objective we care about
Oh no!! The thing I
care about is getting
much, much worse!
Loss
Proxy objective we
know how to measure
and optimize
Better
Time

Gaming



Worse
Peace, love, and puppies objective we care about
Oh no!! The thing I care about is getting much, much worse!
Loss
Proxy objective we know how to measure and optimize
Better
Time



THE COBRA EFFECT
A WELL-INTENTIONED MEASURE CAN OFTEN BACKFIRE AND HAVE THE OPPOSITE EFFECT TO INTENDED
WANTED
DEAD COBRAS
CASH REWARD
COBRA FARM
INTENTION
REDUCE COBRA POPULATION
ACTION
A BOUNTY FOR DEAD COBRAS!
EFFECT
PEOPLE START COBRA FARMING

Strategic Classification Setting [1]

- Classical vs strategic classification: $\mathcal{D}_{train} \neq \mathcal{D}_{test}$ in strategic classification

[1] Strategic Classification, Hardt, Maggiddo, Papadimitriou, Wooters, ITCS 2016.

Strategic Classification Setting [1]

- Classical vs strategic classification: $\mathcal{D}_{train} \neq \mathcal{D}_{test}$ in strategic classification
- $\mathcal{D}_{test}$ is a obtained from implemented classifier f and $\mathcal{D}_{train}$

[1] Strategic Classification, Hardt, Maggiddo, Papadimitriou, Wooters, ITCS 2016.

Strategic Classification Setting [1]

- Classical vs strategic classification: $\mathcal{D}_{train} \neq \mathcal{D}_{test}$ in strategic classification
- $\mathcal{D}_{test}$ is a obtained from implemented classifier f and $\mathcal{D}_{train}$
- Game Theory interpretation: Two players, System and User(s) play following Stackelberg game

[1] Strategic Classification, Hardt, Maggiddo, Papadimitriou, Wooters, ITCS 2016.

Strategic Classification Setting [1]

- Classical vs strategic classification: $\mathcal{D}_{train} \neq \mathcal{D}_{test}$ in strategic classification
- $\mathcal{D}_{test}$ is a obtained from implemented classifier f and $\mathcal{D}_{train}$
- Game Theory interpretation: Two players, System and User(s) play following Stackelberg game
  - System learns a classifier f from training data $\mathcal{D}_{train}$

[1] Strategic Classification, Hardt, Maggiddo, Papadimitriou, Wooters, ITCS 2016.

Strategic Classification Setting [1]

- Classical vs strategic classification: $\mathcal{D}_{train} \neq \mathcal{D}_{test}$ in strategic classification
- $\mathcal{D}_{test}$ is a obtained from implemented classifier f and $\mathcal{D}_{train}$
- Game Theory interpretation: Two players, System and User(s) play following Stackelberg game
  - System learns a classifier f from training data $\mathcal{D}_{train}$
  - System makes f public

[1] Strategic Classification, Hardt, Maggiddo, Papadimitriou, Wooters, ITCS 2016.

Strategic Classification Setting [1]

- Classical vs strategic classification: $\mathcal{D}_{train} \neq \mathcal{D}_{test}$ in strategic classification
- $\mathcal{D}_{test}$ is a obtained from implemented classifier f and $\mathcal{D}_{train}$
- Game Theory interpretation: Two players, System and User(s) play following Stackelberg game
  - System learns a classifier f from training data $\mathcal{D}_{train}$
  - System makes f public
  - User, on observing f, misreport (at cost) her features to obtain the desired outcome from f

Goal: To minimize risk under strategic data distribution shift (strategic error).

[1] Strategic Classification, Hardt, Maggiddo, Papadimitriou, Wooters, ITCS 2016.

Utility Model, Cost ...

Strategies and Utilities:

- Users want favourable outcome; Users utility is 1 if classified positively and 0 otherwise.

Utility Model, Cost ...

Strategies and Utilities:

- Users want favourable outcome; Users utility is 1 if classified positively and 0 otherwise.
- System wants to predict true label accurately;
- Users optimal response to f

$$\Delta_f(x) \in \arg\min_{x' \in \mathcal{X}} \Big(\underbrace{f(x')}_{\text{classfier}} - \underbrace{c(x, x')}_{\text{cost}} \Big)$$

Utility Model, Cost ...

Strategies and Utilities:

- Users want favourable outcome; Users utility is 1 if classified positively and 0 otherwise.
- System wants to predict true label accurately;
- Users optimal response to f

$$\Delta_f(x) \in \arg\min_{x' \in \mathcal{X}} \Big(\underbrace{f(x')}_{\text{classfier}} - \underbrace{c(x, x')}_{\text{cost}} \Big)$$

- $c(x, x')$: cost of reporting x as x'.

Utility Model, Cost ...

Strategies and Utilities:

- Users want favourable outcome; Users utility is 1 if classified positively and 0 otherwise.
- System wants to predict true label accurately;
- Users optimal response to f

$$\Delta_f(x) \in \arg\min_{x' \in \mathcal{X}} \left(\underbrace{f(x')}_{\text{classfier}} - \underbrace{c(x, x')}_{\text{cost}} \right)$$

- $c(x, x')$: cost of reporting x as x'.
- cost is non-negative, truthful reports incur zero cost

Utility Model, Cost ...

Strategies and Utilities:

- Users want favourable outcome; Users utility is 1 if classified positively and 0 otherwise.
- System wants to predict true label accurately;
- Users optimal response to f

$$\Delta_f(x) \in \arg\min_{x' \in \mathcal{X}} \Big(\underbrace{f(x')}_{\text{classfier}} - \underbrace{c(x,x')}_{\text{cost}} \Big)$$

- $c(x,x')$: cost of reporting x as x'.
- cost is non-negative, truthful reports incur zero cost

- System's payoff:$\mathbb{P}_{x \in \mathcal{D}}(y = f(\Delta_f(x)))$. Throughout this talk we will consider strategic error.

$$f^* \in \arg\min_{f \in \mathcal{F}} \mathbb{P}_{x \in \mathcal{D}}(y \neq f(\Delta_f(x)))$$

Systems goal: Find f^* that adjusts to distribution shift in test data.

Separable Cost Functions

Definition (Separable costs)

A cost function $c(x, y)$ is called separable if it can be written as

$$c(x, y) = \max(0, c_2(y) - c_1(x)) \tag{1}$$

$c_1, c_2 : \mathcal{X} \to \mathbb{R}$ and, $c_2(X) \subseteq c_1(X)$.

Separable Cost: Example

Example

$c(x, y) = \langle \alpha, y - x \rangle_+$.

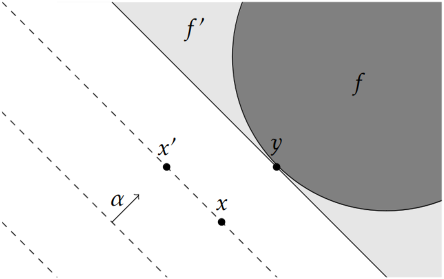


Figure: Let f be an optimal classifier. Then since moving perpendicular to α is cost-free for agent, Systems payoff from f' is equivalent that from f.

General Setting

Definition (Cost threshold classifier)

$$c_i[t](x) = \begin{cases} +1 & c_i(x) \geq t \\ -1 & \text{otherwise} \end{cases}$$

Definition (Rademacher Complexity)

Let $\mathcal{F}$ be a function class and $m > 0$ be a number of i.i.d. samples from $\mathcal{D}$. Define σ_i as i.i.d. Rademacher random variables then

$$\mathcal{R}_m(\mathcal{F}) = \mathbb{E}_{x_1, x_2, \cdots, x_m \sim \mathcal{D}} \mathbb{E}_{\sigma_1, \sigma_2, \cdots \sigma_n} \left[\sup \left\{ \frac{1}{m} \sum_{i=1}^{m} \sigma_i f(x_i) : f \in \mathcal{F} \right\} \right] \tag{2}$$

Algorithm for SC

Algorithm 1 Strategic ERM

Require: Data: $(x_i, y_i)_{i\in[m]}$, $c(x,y) = \max(0, c_2(y) - c_1(x))$.

for $i = 1$ to m **do**

  $t_i := c_1(x_i)$

  $s_i = \begin{cases} \max(c_2(X \cap [t_i, t_i + 2]) & c_2(X) \cap [t_i, t_i + 2] \neq \emptyset \\ \infty & \text{otherwise} \end{cases}$

  set $s_{m+1} = \infty$

end for

Compute:

$$\widehat{\mathrm{ERR}}(s_i) = \frac{1}{m}\sum_{j=1}^{m} \mathbb{1}\{h(x_j) \neq c_1[s_i - 2](x_j)\}. \tag{3}$$

Find $i^*, 1 \leq i^* \leq m+1$ that minimizes $\widehat{\mathrm{ERR}}(s_i)$.

return $f := c_2[s_i^*]$

Important Results

Theorem

Let $\mathcal{H}$ be a concept class, $\mathcal{D}$ be a distribution and c be a separable cost function. Further, let m denote the number of samples and suppose

$$\mathcal{R}_m(\mathcal{H}) + 2\sqrt{\frac{\log(m+1)}{m}} + \sqrt{\frac{\log(2/\delta)}{8m}} \leq \frac{\varepsilon}{8}. \tag{4}$$

Then with probability atleast $1-\delta$,

$$\mathbb{P}_{x \in \mathcal{D}}(h(x) = f(\Delta(x))) \geq \textsc{Opt}_h(\mathcal{D}, c) - \varepsilon.$$

Variation 1: SC in the Dark [2]

1. Agent(s) may not have complete access to f;

[2] Ghalme et al. Strategic Classification in the Dark, ICML 2021.

Variation 1: SC in the Dark [2]

1. Agent(s) may not have complete access to f;
2. Agents may have access to decisions by f; Example: OpenShufa

[2] Ghalme et al. Strategic Classification in the Dark, ICML 2021.

Variation 1: SC in the Dark [2]

1. Agent(s) may not have complete access to f;
2. Agents may have access to decisions by f; Example: OpenShufa

Definition (Strategic error in the dark)

$$\mathrm{Err}(f, \widehat{f}) = \mathbb{P}_{x \sim \mathcal{D}}(y \neq f(\Delta_{\widehat{f}}(x))) \tag{5}$$

[2] Ghalme et al. Strategic Classification in the Dark, ICML 2021.

Variation 1: SC in the Dark [2]

1. Agent(s) may not have complete access to f;
2. Agents may have access to decisions by f; Example: OpenShufa

Definition (Strategic error in the dark)

$$\textsc{Err}(f, \widehat{f}) = \mathbb{P}_{x \sim \mathcal{D}}(y \neq f(\Delta_{\widehat{f}}(x))) \tag{5}$$

Who is in the dark?

[2] Ghalme et al. Strategic Classification in the Dark, ICML 2021.

Variation 1: SC in the Dark [2]

1. Agent(s) may not have complete access to f;
2. Agents may have access to decisions by f; Example: OpenShufa

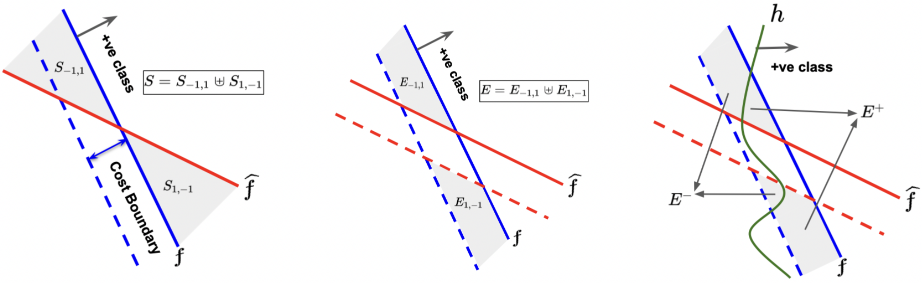
Definition (Strategic error in the dark)

$$\textsc{Err}(f, \widehat{f}) = \mathbb{P}_{x \sim \mathcal{D}}(y \neq f(\Delta_{\widehat{f}}(x))) \tag{5}$$

Who is in the dark? By making f public, System can anticipate agents' response better (and construct robust f). By keeping f private, System is also in the dark as uninformed (partially informed) users may lead to unpredictable response.

[2] Ghalme et al. Strategic Classification in the Dark, ICML 2021.

Price of Opacity



Main Results

Definition (Price of Opacity (POP))

$$POP(f, f') := \textsc{Err}(f, f') - \textsc{Err}(f, f).$$

Here f is the System's classifier and f' is the classifier Agents' classifier (Agent responds to f').

Main Results

Definition (Price of Opacity (POP))

$$POP(f, f') := \mathrm{Err}(f, f') - \mathrm{Err}(f, f).$$

Here f is the System's classifier and f' is the classifier Agents' classifier (Agent responds to f').

Theorem (POP characterization)

If $\mathbb{P}_{x \sim \mathcal{D}}(x \in E) > 2\mathrm{Err}(f^*, f^*) + 2\varepsilon$, then $POP > 0$, for a given $\varepsilon > 0$.

Results

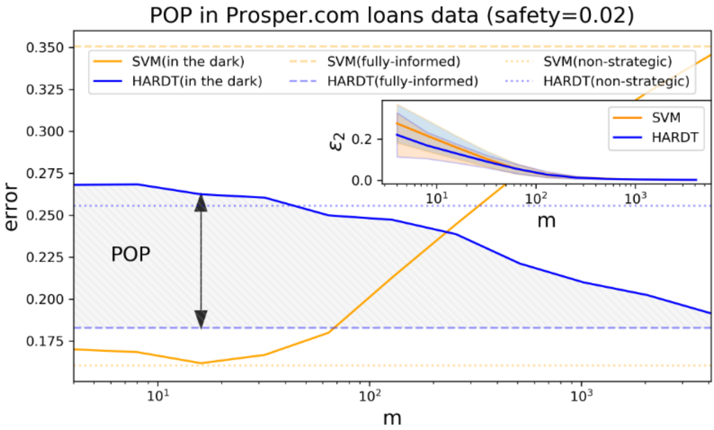


Figure: Price of Opacity is positive and decreases with the training samples m used to construct $\widehat{f}$.

Variation 2: Performative Prediction

- SC assumption: Labels are immutable
- Performative Prediction: The distribution $\mathcal{D}$ changes (inclding true labels) to D_θ.

Performative Prediction

Performative Prediction

Definition (Performative Risk)

$$PR(\theta) = \mathbb{R}_{Z \sim \mathcal{D}(\theta)} \ell(Z; \theta)$$

Performative Prediction

Definition (Performative Risk)

$$PR(\theta) = \mathbb{R}_{Z \sim \mathcal{D}(\theta)} \ell(Z; \theta)$$

Definition (Iterative Version)

$$\theta_{t+1} = \arg\min_{\theta} \mathbb{E}_{Z \sim \mathcal{D}(\theta_t)} \ell(Z; \theta)$$

Performative Prediction

Definition (Performative Risk)

$$PR(\theta) = \mathbb{R}_{Z \sim \mathcal{D}(\theta)} \ell(Z; \theta)$$

Definition (Iterative Version)

$$\theta_{t+1} = \arg\min_{\theta} \mathbb{E}_{Z \sim \mathcal{D}(\theta_t)} \ell(Z; \theta)$$

Definition (Performative Stability)

A model $f_{\theta_{ps}}$ is called performatively stable if

$$\theta_{PS} = \arg\min_{\theta} \mathbb{E}_{Z \sim \mathcal{D}(\theta_{PS})} \ell(z; \theta)) \tag{6}$$

Results: Performative Predictions

Theorem (Informal)

If the loss is smooth, strongly convex, and the mapping $\mathcal{D}(.)$ is sufficiently Lipschitz, then repeated risk minimization converges to performative stability at a linear rate.

Results: Performative Predictions

Theorem (Informal)

If the loss is smooth, strongly convex, and the mapping $\mathcal{D}(.)$ is sufficiently Lipschitz, then repeated risk minimization converges to performative stability at a linear rate.

Theorem (Informal)

If the loss is Lipschitz and strongly convex, and the map $\mathcal{D}()$ is Lipschitz, all stable points and performative optima lie in a small neighbourhood around each other.

Takeaways

- Traditional ML algorithms perform poorly in a strategic setting

भारतीय प्रौद्योगिकी संस्थान हैदराबाद
Indian Institute of Technology Hyderabad

Takeaways

- Traditional ML algorithms perform poorly in a strategic setting
- The other extreme; overfit to strategic nature

Takeaways

- Traditional ML algorithms perform poorly in a strategic setting
- The other extreme; overfit to strategic nature
- Strategic classifiers are learnable under reasonable assumptions on cost functions

Takeaways

- Traditional ML algorithms perform poorly in a strategic setting
- The other extreme; overfit to strategic nature
- Strategic classifiers are learnable under reasonable assumptions on cost functions
- Many questions: Heterogeneous Users, Social Burden, Information disparity, Herd Behavior....

Takeaways

- Traditional ML algorithms perform poorly in a strategic setting
- The other extreme; overfit to strategic nature
- Strategic classifiers are learnable under reasonable assumptions on cost functions
- Many questions: Heterogeneous Users, Social Burden, Information disparity, Herd Behavior....
- Beyond SC: Ranking, clustering, Online learning...

Takeaways

- Traditional ML algorithms perform poorly in a strategic setting
- The other extreme; overfit to strategic nature
- Strategic classifiers are learnable under reasonable assumptions on cost functions
- Many questions: Heterogeneous Users, Social Burden, Information disparity, Herd Behavior....
- Beyond SC: Ranking, clustering, Online learning...
- System Manipulation: strategic representation, User targetting, Persuasion ...

Takeaways

- Traditional ML algorithms perform poorly in a strategic setting
- The other extreme; overfit to strategic nature
- Strategic classifiers are learnable under reasonable assumptions on cost functions
- Many questions: Heterogeneous Users, Social Burden, Information disparity, Herd Behavior....
- Beyond SC: Ranking, clustering, Online learning...
- System Manipulation: strategic representation, User targetting, Persuasion ...

Thank you!