Mistake, Manipulation and Margin Guarantees in Online Strategic Classification

Fatma Kılınç-Karzan

Carnegie Mellon University
Tepper School of Business

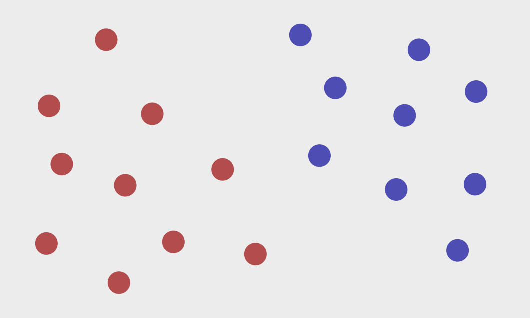
Joint work with **Lingqing Shen, Nam Ho-Nguyen, Hung Giang-Tran**

UPenn Optimization Seminar

Motivation

Classification

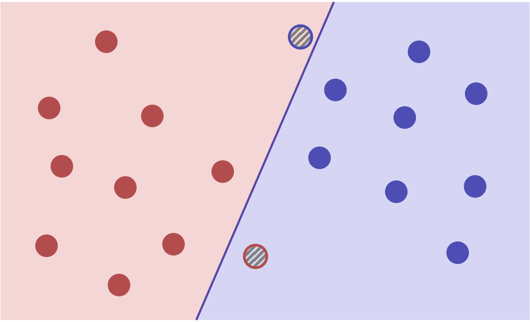
- Fundamental task in many domains: image classification, loan approval, ...



Motivation

Classification

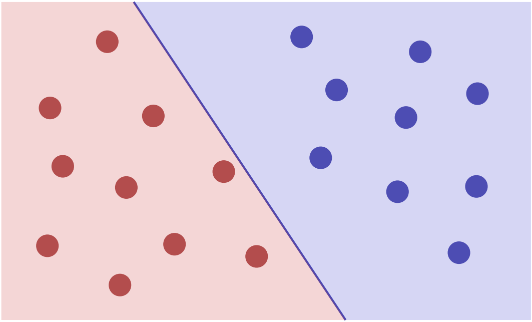
- Fundamental task in many domains: image classification, loan approval, . . .



Motivation

Classification

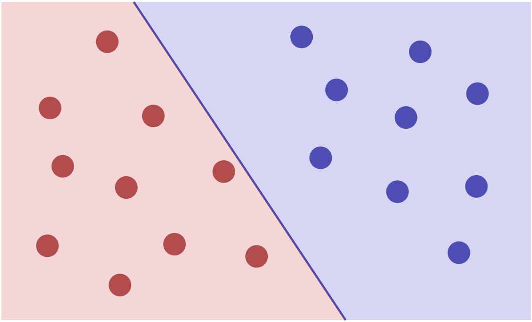
- Fundamental task in many domains: image classification, loan approval, . . .



Motivation

Classification

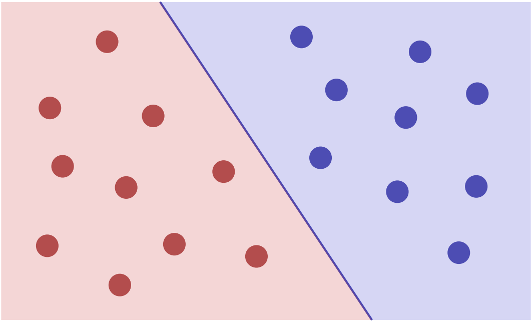
- What if the data are noisy?



Motivation

Classification

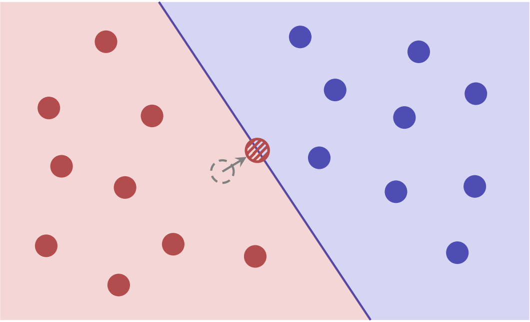
- What if the data are noisy?



Motivation

Classification

- What if the data are strategic?



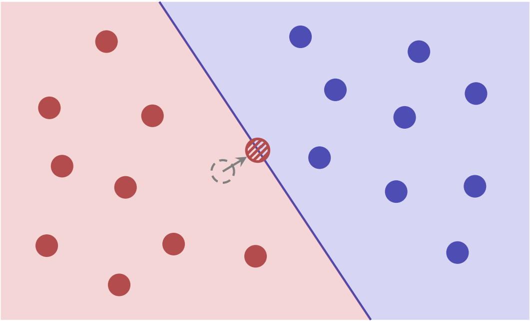
Motivation

Strategic behavior in classification

- binary classification as a game between an agent and a learner
- the agent manipulates their features to achieve a desired outcome

Strategic behavior in classification

- binary classification as a game between an agent and a learner
- the agent manipulates their features to achieve a desired outcome



Motivation

Strategic behavior in classification

- binary classification as a game between an agent and a learner
- the agent manipulates their features to achieve a desired outcome
  - e.g., graduate school admission, bank loan approval
  - true features and labels are not actually improved
  - manipulated features can be misleading

Motivation

Strategic behavior in classification

- binary classification as a game between an agent and a learner
- the agent manipulates their features to achieve a desired outcome
  - e.g., graduate school admission, bank loan approval
  - true features and labels are not actually improved
  - manipulated features can be misleading
- the learner aims at a classifier that effectively
  - predicts true labels,

Motivation

Strategic behavior in classification

- binary classification as a game between an agent and a learner
- the agent manipulates their features to achieve a desired outcome
  - e.g., graduate school admission, bank loan approval
  - true features and labels are not actually improved
  - manipulated features can be misleading
- the learner aims at a classifier that effectively
  - predicts true labels, and possibly discourages manipulation

Motivation

Strategic behavior in classification

- binary classification as a game between an agent and a learner
- the agent manipulates their features to achieve a desired outcome
  - e.g., graduate school admission, bank loan approval
  - true features and labels are not actually improved
  - manipulated features can be misleading
- the learner aims at a classifier that effectively
  - predicts true labels, and possibly discourages manipulation
- strategic agents $\neq$ adversarial agents

Motivation

Strategic behavior in classification

- challenge: as you learn and modify your decision rule, the agents will change how they respond to it

Motivation

Strategic behavior in classification

- challenge: as you learn and modify your decision rule, the agents will change how they respond to it
  - especially in online (non-distributional) settings, this leads to an informational problem in addition to computational problem

Motivation

Strategic behavior in classification

- challenge: as you learn and modify your decision rule, the agents will change how they respond to it
  - especially in online (non-distributional) settings, this leads to an informational problem in addition to computational problem
- similar to online learning of a Stackelberg leader strategy

Motivation

Strategic behavior in classification

- challenge: as you learn and modify your decision rule, the agents will change how they respond to it
  - especially in online (non-distributional) settings, this leads to an informational problem in addition to computational problem
- similar to online learning of a Stackelberg leader strategy
- challenge: as we measure performance (in this case agent's features), agents will manipulate without necessarily improving

Motivation

Strategic behavior in classification

- challenge: as you learn and modify your decision rule, the agents will change how they respond to it
  - especially in online (non-distributional) settings, this leads to an informational problem in addition to computational problem
- similar to online learning of a Stackelberg leader strategy
- challenge: as we measure performance (in this case agent's features), agents will manipulate without necessarily improving
- question: can we minimize mistakes and manipulations together?

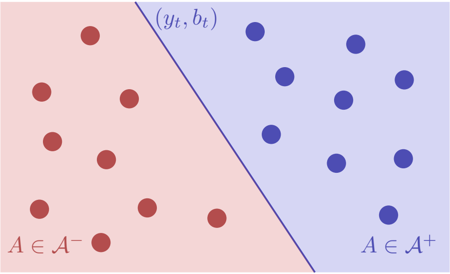
Problem Overview

Online setting: at each time step t, the agent and the learner take action alternately

Problem Overview

Online setting: at each time step t, the agent and the learner take action alternately

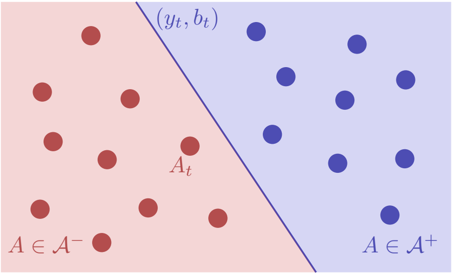
- agent
  - observes the current classifier (y_t, b_t) given by $x \mapsto \widetilde{\text{label}}(x, y_t, b_t)$



Problem Overview

Online setting: at each time step t, the agent and the learner take action alternately

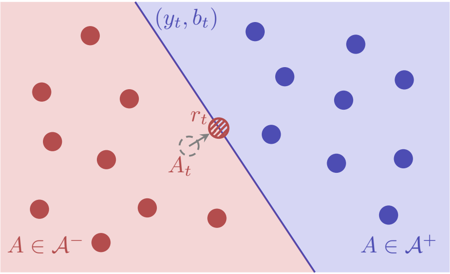
- agent
  - observes the current classifier (y_t, b_t) given by $x \mapsto \widetilde{\text{label}}(x, y_t, b_t)$
  - given their feature vector A_t,



Problem Overview

Online setting: at each time step t, the agent and the learner take action alternately

- agent
  - observes the current classifier (y_t, b_t) given by $x \mapsto \widetilde{\text{label}}(x, y_t, b_t)$
  - given their feature vector A_t, reports manipulated feature vector $r_t := r(A_t, y_t, b_t)$



Problem Overview

Online setting: at each time step t, the agent and the learner take action alternately

- agent
  - observes the current classifier (y_t, b_t) given by $x \mapsto \widetilde{\text{label}}(x, y_t, b_t)$
  - given their feature vector A_t, reports manipulated feature vector $r_t := r(A_t, y_t, b_t)$
- learner
  - observes the manipulated features $r_t = r(A_t, y_t, b_t)$

Problem Overview

Online setting: at each time step t, the agent and the learner take action alternately

- agent
  - observes the current classifier (y_t, b_t) given by $x \mapsto \widetilde{\text{label}}(x, y_t, b_t)$
  - given their feature vector A_t, reports manipulated feature vector $r_t := r(A_t, y_t, b_t)$
- learner
  - observes the manipulated features $r_t = r(A_t, y_t, b_t)$
  - makes a prediction $\widetilde{\text{label}}(r_t, y_t, b_t)$ using the current classifier (y_t, b_t)

Problem Overview

Online setting: at each time step t, the agent and the learner take action alternately

- agent
  - observes the current classifier (y_t, b_t) given by $x \mapsto \widetilde{\text{label}}(x, y_t, b_t)$
  - given their feature vector A_t, reports manipulated feature vector $r_t := r(A_t, y_t, b_t)$
- learner
  - observes the manipulated features $r_t = r(A_t, y_t, b_t)$
  - makes a prediction $\widetilde{\text{label}}(r_t, y_t, b_t)$ using the current classifier (y_t, b_t)
  - receives the true $\ell_t := \text{label}(A_t)$

Problem Overview

Online setting: at each time step t, the agent and the learner take action alternately

- agent
  - observes the current classifier (y_t, b_t) given by $x \mapsto \widetilde{\text{label}}(x, y_t, b_t)$
  - given their feature vector A_t, reports manipulated feature vector $r_t := r(A_t, y_t, b_t)$
- learner
  - observes the manipulated features $r_t = r(A_t, y_t, b_t)$
  - makes a prediction $\widetilde{\text{label}}(r_t, y_t, b_t)$ using the current classifier (y_t, b_t)
  - receives the true $\ell_t := \text{label}(A_t)$
  - updates the classifier to (y_{t+1}, b_{t+1}) based on historical data $\{(r_\tau, \ell_\tau, y_\tau, b_\tau)\}_{\tau \in [t]}$

    (without knowledge of true features $\{A_\tau\}_{\tau \in [t]}$)

Literature

How does the agent manipulate?

Various manipulation models:

- utility maximization:[1,2,3,4] $\max_x \{\text{gain}(x, y_t, b_t) - \text{cost}(A_t, x)\}$
- discrete features via a manipulation graph[5,6]

[1][Hardt el al., 2016], [2][Dong et al., 2018], [3][Chen et al., 2020], [4][Ahmadi et al., 2021], [5][Lechner and Urner, 2022], [6][Ahmadi et al., 2023]

Literature

How does the agent manipulate?

Various manipulation models:

- utility maximization:[1,2,3,4] $\max_x \{\text{gain}(x, y_t, b_t) - \text{cost}(A_t, x)\}$
- discrete features via a manipulation graph[5,6]

How to evaluate the classifier's effectiveness in the strategic setting?

- mistake bound[1,4,6]
- Stackelberg regret[3,6,2] w.r.t. various loss functions

[1] [Hardt el al., 2016], [2] [Dong et al., 2018], [3] [Chen et al., 2020], [4] [Ahmadi et al., 2021], [5] [Lechner and Urner, 2022], [6] [Ahmadi et al., 2023]

Our Model

We consider the following model:

- online scenario, $t = 1, 2, \ldots$
- binary classification, $\text{label}(A_t) \in \{-1, +1\}$
- linear classifier, $x \mapsto \widetilde{\text{label}}(x, y_t, b_t) = \text{sign}(y_t^\top x + b_t')$

Our Model

We consider the following model:

- online scenario, $t = 1, 2, \dots$
- binary classification, $\text{label}(A_t) \in \{-1, +1\}$
- linear classifier, $x \mapsto \widetilde{\text{label}}(x, y_t, b_t) = \text{sign}(y_t^\top x + b_t')$
- agent's utility function

$$r(A_t, y_t, b_t) \in \arg\max_{x \in \mathbb{R}^d} \left\{ \widetilde{\text{label}}(x, y_t, b_t) - \text{cost}(A_t, x) \right\}$$

- tradeoff between desired prediction outcome and manipulation cost

Our Model

We consider the following model:

- online scenario, $t = 1, 2, \ldots$
- binary classification, $\text{label}(A_t) \in \{-1, +1\}$
- linear classifier, $x \mapsto \widetilde{\text{label}}(x, y_t, b_t) = \text{sign}(y_t^\top x + b_t')$
- agent's utility function

$$r(A_t, y_t, b_t) \in \arg\max_{x \in \mathbb{R}^d} \left\{ \widetilde{\text{label}}(x, y_t, b_t) - c\|x - A_t\| \right\}$$

- tradeoff between desired prediction outcome and manipulation cost
- **assumption:** $\text{cost}(A_t, x)$ resembles a distance metric $\Rightarrow$ $\text{cost}(A_t, x) = c\|x - A_t\|$

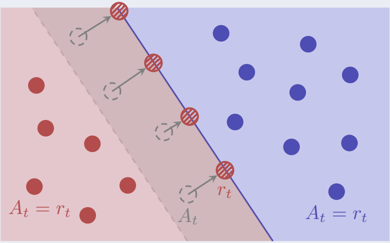
Preliminaries: Agent's response

Assumption

The agent's manipulation cost is $c\|x - A_t\|$, where c & $\|\cdot\|$ are known to the learner.

Lemma

*Given a classifier $x \mapsto \text{sign}\left(y^\top x + b - \frac{2\|y\|_*}{c}\right)$, the agent's response (i.e., manipulated feature) is given by**



*

Preliminaries: Agent's response

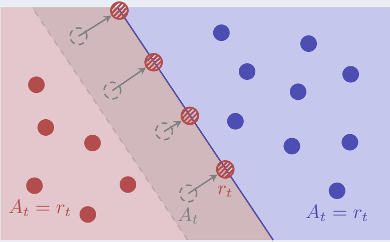
Assumption

The agent's manipulation cost is $c\|x - A_t\|$, where c & $\|\cdot\|$ are known to the learner.

Lemma

*Given a classifier $x \mapsto \operatorname{sign}\left(y^\top x + b - \frac{2\|y\|_*}{c}\right)$, the agent's response (i.e., manipulated feature) is given by**

$$r(A,y,b) = \begin{cases} A + \left(\frac{2}{c} - \frac{y^\top A + b}{\|y\|_*}\right) v(y), & \text{if } 0 \leq \frac{y^\top A+b}{\|y\|_*} < \frac{2}{c} \\ A, & \text{otherwise} \end{cases}$$



where $v(y) \in \partial\|y\|_*$.

*Learner and agent use the same common tie-breaking rule whenever the optimal response is not unique.

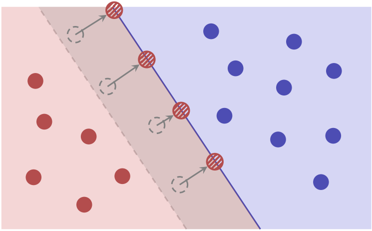
Preliminaries: Prediction

In the strategic setting, what is an ideal classifier?

Preliminaries: Prediction

In the strategic setting, what is an ideal classifier?

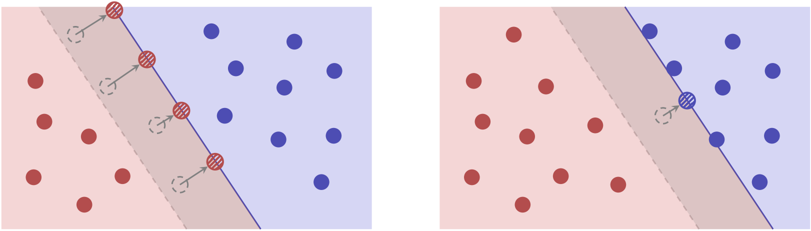
- a correct classifier on unmanipulated data may be incorrect on manipulated data



Preliminaries: Prediction

In the strategic setting, what is an ideal classifier?

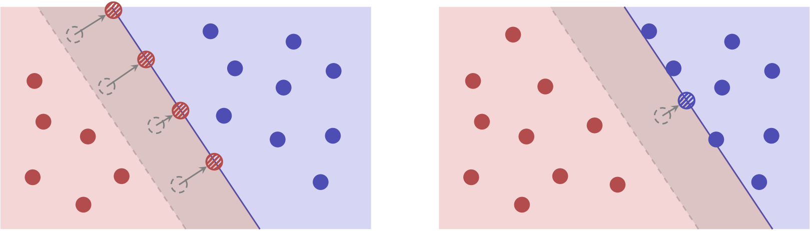
- a correct classifier on unmanipulated data may be incorrect on manipulated data
- an incorrect classifier on unmanipulated data may become correct



Preliminaries: Prediction

In the strategic setting, what is an ideal classifier?

- a correct classifier on unmanipulated data may be incorrect on manipulated data
- an incorrect classifier on unmanipulated data may become correct

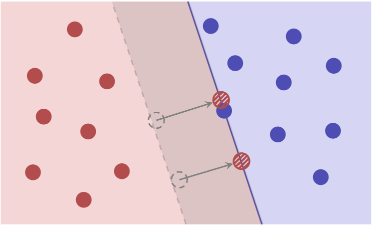


- key idea 1: shift the decision hyperplane so that $\widetilde{\text{label}}(A, y, b) = \text{sign}\left(\frac{y^\top A + b}{\|y\|_*} - \frac{2}{c}\right)$
- lemma: If $x \mapsto \text{sign}(\frac{y^\top x + b}{\|y\|_*})$ classifies all unmanipulated data correctly, then $x \mapsto \text{sign}(\frac{y^\top x + b}{\|y\|_*} - \frac{2}{c})$ classifies all *manipulated* features correctly

Preliminaries: Proxy data

What else could go wrong with manipulated data?

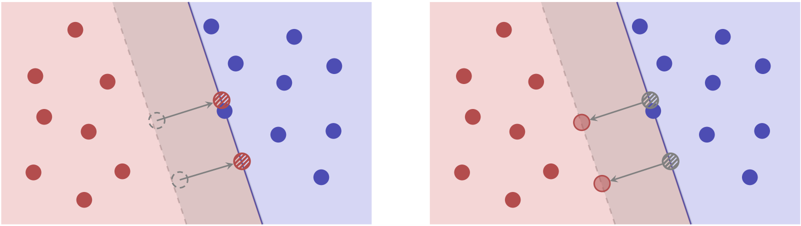
- agent's responses can be inseparable even if unmanipulated data are separable



Preliminaries: Proxy data

What else could go wrong with manipulated data?

- agent's responses can be inseparable even if unmanipulated data are separable



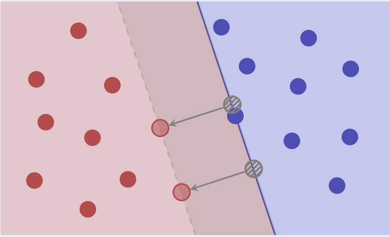
- key idea 2: construct a proxy $s(A_t, y_t, b_t)$ that approximates A_t using *only* the information we have, i.e., r_t, ℓ_t, y_t, b_t

Preliminaries: Proxy data

Lemma

Given a classifier $x \mapsto \text{sign}\left(y^\top x + b - \frac{2\|y\|_*}{c}\right)$, *and agent's response* $r(A, y, b)$, *the* *proxy data* *is computed as*

$$s(A,y,b) = \begin{cases} r(A, y, b) - \frac{2}{c} v(y), & \text{if } \frac{y^\top r(A,y,b)+b}{\|y\|_*} = \frac{2}{c} \\ & \quad \text{and } \text{label}(A) = -1, \\ A, & \text{otherwise.} \end{cases}$$

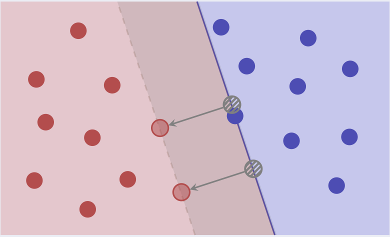


Preliminaries: Proxy data

Lemma

Given a classifier $x \mapsto \mathrm{sign}\left(y^\top x + b - \frac{2\|y\|_*}{c}\right)$, *and agent's response* $r(A, y, b)$, *the* ***proxy data*** *is computed as*

$$s(A,y,b) = \begin{cases} r(A,y,b) - \frac{2}{c} v(y), & \text{if } \frac{y^\top r(A,y,b)+b}{\|y\|_*} = \frac{2}{c} \\ & \text{and } \mathrm{label}(A) = -1, \\ A, & \text{otherwise.} \end{cases}$$



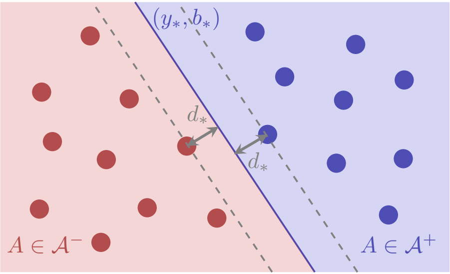
Lemma (correctness)

A response $r(A, y, b)$ *is misclassified by* $x \mapsto \mathrm{sign}(y^\top x + b - 2\|y\|_*/c) = \widetilde{\mathrm{label}}(x, y, b)$
$\iff$ *its proxy* $s(A, y, b)$ *is misclassified by* $x \mapsto \mathrm{sign}(y^\top x + b)$.

Preliminaries: Margin

Assumption (separability)

Unmanipulated data $\{(A_t, \text{label}(A_t))\}$ are separable, with a max margin classifier (y_*, b_*) achieving a margin of $d_* > 0$.



Preliminaries: Margin

Assumption (separability)

Unmanipulated data $\{(A_t, \text{label}(A_t))\}$ *are separable, with a max margin classifier* (y_*, b_*) *achieving a margin of* $d_* > 0$.

Question

Proxy data $s(A, y, b)$ depends on classifier (y, b). As we learn and revise classifiers (y_t, b_t), how can we ensure that proxy data remains separable?

Preliminaries: Margin

Assumption (separability)

Unmanipulated data $\{(A_t, \text{label}(A_t))\}$ *are separable, with a max margin classifier* (y_*, b_*) *achieving a margin of* $d_* > 0$.

Lemma (classifier alignment)

Suppose $(y, b), (\bar{y}, \bar{b}) \in \mathbb{R}^d \setminus \{0\} \times \mathbb{R}$ *are such that* $\bar{y}^\top v(y) \geq 0$. *Then,*

- $\text{label}(A) \cdot (\bar{y}^\top s(A, y, b) + \bar{b}) \geq \text{label}(A) \cdot (\bar{y}^\top A + \bar{b})$ *for all* A;

Preliminaries: Margin

Assumption (separability)

Unmanipulated data $\{(A_t, \text{label}(A_t))\}$ *are separable, with a max margin classifier* (y_*, b_*) *achieving a margin of* $d_* > 0$.

Lemma (classifier alignment)

Suppose $(y, b), (\bar{y}, \bar{b}) \in \mathbb{R}^d \setminus \{0\} \times \mathbb{R}$ *are such that* $\bar{y}^\top v(y) \geq 0$. *Then,*

- $\text{label}(A) \cdot (\bar{y}^\top s(A, y, b) + \bar{b}) \geq \text{label}(A) \cdot (\bar{y}^\top A + \bar{b})$ *for all* A;
- *thus,* $\min_{A \in \mathcal{A}} \left\{ \text{label}(A) \cdot \frac{\bar{y}^\top s(A,y,b) + \bar{b}}{\|\bar{y}\|_*} \right\} \geq \min_{A \in \mathcal{A}} \left\{ \text{label}(A) \cdot \frac{\bar{y}^\top A + \bar{b}}{\|\bar{y}\|_*} \right\}$;

Preliminaries: Margin

Assumption (separability)

Unmanipulated data $\{(A_t, \text{label}(A_t))\}$ *are separable, with a max margin classifier* (y_*, b_*) *achieving a margin of* $d_* > 0$.

Lemma (classifier alignment)

Suppose $(y, b), (\bar{y}, \bar{b}) \in \mathbb{R}^d \setminus \{0\} \times \mathbb{R}$ *are such that* $\bar{y}^\top v(y) \geq 0$. *Then,*

- $\text{label}(A) \cdot (\bar{y}^\top s(A, y, b) + \bar{b}) \geq \text{label}(A) \cdot (\bar{y}^\top A + \bar{b})$ *for all* A;
- *thus,* $\min_{A \in \mathcal{A}} \left\{ \text{label}(A) \cdot \frac{\bar{y}^\top s(A,y,b) + \bar{b}}{\|\bar{y}\|_*} \right\} \geq \min_{A \in \mathcal{A}} \left\{ \text{label}(A) \cdot \frac{\bar{y}^\top A + \bar{b}}{\|\bar{y}\|_*} \right\}$;

That is, under separability assumption on unmanipulated data, for every $y \in \mathbb{R}^d \setminus \{0\}$ *satisfying* $y_*^\top v(y) \geq 0$, *we have proxy data* $s(A, y, b)$ *are separable with margin at least* d_*.

Main Idea

Generate and use classifiers (y_t, b_t) that ensure separability of the proxy data $s(A_t, y_t, b_t)$ and work with the proxy data

Algorithms

Main Idea

Generate and use classifiers (y_t, b_t) that ensure separability of the proxy data $s(A_t, y_t, b_t)$ and work with the proxy data

What works in the non-strategic setting?

- *perceptron*
  - update by $y_{t+1} \leftarrow y_t + \text{label}(A_t) \cdot A_t$ whenever A_t is misclassified
  - finite mistake bound, but no margin guarantee
  - computationally cheap

Algorithms

Main Idea

Generate and use classifiers (y_t, b_t) that ensure separability of the proxy data $s(A_t, y_t, b_t)$ and work with the proxy data

What works in the non-strategic setting?

- *perceptron*
  - update by $y_{t+1} \leftarrow y_t + \text{label}(A_t) \cdot A_t$ whenever A_t is misclassified
  - finite mistake bound, but no margin guarantee
  - computationally cheap
- margin maximization
  - $$\max_{\|y\|_* \leq 1, b \in \mathbb{R}} \min_t \left\{ \text{label}(A_t) \cdot (y^\top A_t + b) \right\}$$
  - maximal margin classifier
  - computationally expensive

Strategic Perceptron: Algorithm

Projected strategic perceptron (S-perceptron)†

Select a closed convex cone $\mathbb{L} \subset \mathbb{R}^d \times \mathbb{R}$. Initialize by $(y_0, b_0) = 0$.

At iteration $t = 1, 2, \ldots$

Step 1. Receive manipulated data r_t and predict by $\widetilde{\text{label}}(r_t, y_t, b_t)$.

Step 2. Receive $\text{label}(A_t)$ and compute the proxy $s(A_t, y_t, b_t)$

Step 3. Update by $(y_{t+1}, b_{t+1}) = \text{Proj}_{\mathbb{L}}(z_{t+1})$ where

$$z_{t+1} = \begin{cases} (y_t, b_t) + \text{label}(A_t) \cdot (s(A_t, y_t, b_t), 1), & \text{if } A_t \text{ is misclassified,} \\ (y_t, b_t), & \text{otherwise.} \end{cases}$$

- Why projection onto a cone?
- To capture a priori information on y_* or b_*, e.g., $b_* = 0$ or $y_* \in \mathbb{R}^d_+$, etc.

---

†

Strategic Perceptron: Algorithm

Projected strategic perceptron (S-perceptron)[†]

Select a closed convex cone $\mathbb{L} \subset \mathbb{R}^d \times \mathbb{R}$. Initialize by $(y_0, b_0) = 0$.

At iteration $t = 1, 2, \ldots$

Step 1. Receive manipulated data r_t and predict by $\widetilde{\text{label}}(r_t, y_t, b_t)$.

Step 2. Receive $\text{label}(A_t)$ and compute the proxy $s(A_t, y_t, b_t)$

Step 3. Update by $(y_{t+1}, b_{t+1}) = \text{Proj}_{\mathbb{L}}(z_{t+1})$ where

$$z_{t+1} = \begin{cases} (y_t, b_t) + \text{label}(A_t) \cdot (s(A_t, y_t, b_t), 1), & \text{if } A_t \text{ is misclassified,} \\ (y_t, b_t), & \text{otherwise.} \end{cases}$$

- Why projection onto a cone?
- To capture a priori information on y_* or b_*, e.g., $b_* = 0$ or $y_* \in \mathbb{R}^d_+$, etc.

---

†

Strategic Perceptron: Algorithm

Projected strategic perceptron (S-perceptron)†

Select a closed convex cone $\mathbb{L} \subset \mathbb{R}^d \times \mathbb{R}$. Initialize by $(y_0, b_0) = 0$.

At iteration $t = 1, 2, \ldots$

Step 1. Receive manipulated data r_t and predict by $\widetilde{\text{label}}(r_t, y_t, b_t)$.

Step 2. Receive $\text{label}(A_t)$ and compute the proxy $s(A_t, y_t, b_t)$

Step 3. Update by $(y_{t+1}, b_{t+1}) = \text{Proj}_{\mathbb{L}}(z_{t+1})$ where

$$z_{t+1} = \begin{cases} (y_t, b_t) + \text{label}(A_t) \cdot (s(A_t, y_t, b_t), 1), & \text{if } A_t \text{ is misclassified}, \\ (y_t, b_t), & \text{otherwise}. \end{cases}$$

- Why projection onto a cone?
- To capture a priori information on y_* or b_*, e.g., $b_* = 0$ or $y_* \in \mathbb{R}^d_+$, etc.

---

†

Strategic Perceptron: Algorithm

Projected strategic perceptron (S-perceptron)[†]

Select a closed convex cone $\mathbb{L} \subset \mathbb{R}^d \times \mathbb{R}$. Initialize by $(y_0, b_0) = 0$.

At iteration $t = 1, 2, \ldots$

Step 1. Receive manipulated data r_t and predict by $\widetilde{\text{label}}(r_t, y_t, b_t)$.

Step 2. Receive $\text{label}(A_t)$ and compute the proxy $s(A_t, y_t, b_t)$

Step 3. Update by $(y_{t+1}, b_{t+1}) = \text{Proj}_{\mathbb{L}}(z_{t+1})$ where

$$z_{t+1} = \begin{cases} (y_t, b_t) + \text{label}(A_t) \cdot (\, s(A_t, y_t, b_t)\, , 1), & \text{if } A_t \text{ is misclassified,} \\ (y_t, b_t), & \text{otherwise.} \end{cases}$$

- Why projection onto a cone?
- To capture a priori information on y_* or b_*, e.g., $b_* = 0$ or $y_* \in \mathbb{R}^d_+$, etc.

†

Strategic Perceptron: Algorithm

Projected strategic perceptron (S-perceptron)[†]

Select a closed convex cone $\mathbb{L} \subset \mathbb{R}^d \times \mathbb{R}$. Initialize by $(y_0, b_0) = 0$.
At iteration $t = 1, 2, \ldots$

Step 1. Receive manipulated data r_t and predict by $\widetilde{\text{label}}(r_t, y_t, b_t)$.

Step 2. Receive $\text{label}(A_t)$ and compute the proxy $s(A_t, y_t, b_t)$

Step 3. Update by $(y_{t+1}, b_{t+1}) = \text{Proj}_{\mathbb{L}}(z_{t+1})$ where

$$z_{t+1} = \begin{cases} (y_t, b_t) + \text{label}(A_t) \cdot (s(A_t, y_t, b_t), 1), & \text{if } A_t \text{ is misclassified,} \\ (y_t, b_t), & \text{otherwise.} \end{cases}$$

- Why projection onto a cone?
- To capture a priori information on y_* or b_*, e.g., $b_* = 0$ or $y_* \in \mathbb{R}^d_+$, etc.

[†] Captures the strategic perceptron algorithm of [Ahmadi et al., 2021] for ℓ_2-based manipulation costs.

Strategic Perceptron: Results

Let $\mathcal{M} = \#$ of mistakes throughout the algorithm.

Theorem (informal)

S-perceptron algorithm is guaranteed to have a **finite mistake bound** ...

- *whenever* $d_* > \frac{2}{c}$, *but no prior knowledge on* (y^*, b^*) *exists:*

  select $\mathbb{L} = \mathbb{R}^d \times \mathbb{R}$ *to get* $|\mathcal{M}| \leq \frac{\|y_*\|_2^2 + b_*^2}{\|y_*\|_*^2} \frac{\tilde{D}^2 + 1}{(d_* - 2/c)^2}$;

*

Strategic Perceptron: Results

Let $\mathcal{M} = \#$ of mistakes throughout the algorithm.

Theorem (informal)

S-perceptron algorithm is guaranteed to have a **finite mistake bound** ...

- *whenever* $d_* > \frac{2}{c}$*, but no prior knowledge on* (y^*, b^*) *exists:*

$$\text{select } \mathbb{L} = \mathbb{R}^d \times \mathbb{R} \text{ to get } |\mathcal{M}| \leq \frac{\|y_*\|_2^2 + b_*^2}{\|y_*\|_*^2} \frac{\tilde{D}^2+1}{(d_* - 2/c)^2};$$

- *whenever* $b_* = 0$ *is known a priori and* $\|\cdot\|$ *is* ℓ_2 *norm***:*

$$\text{select } \mathbb{L} = \mathbb{R}^d \times \{0\} \text{ to get } |\mathcal{M}| \leq \frac{\tilde{D}^2+1}{d_*^2};$$

*Recovers mistake bounds from [Ahmadi et al., 2021] given for this case.

Strategic Perceptron: Results

Let $\mathcal{M} = \#$ of mistakes throughout the algorithm.

Theorem (informal)

S-perceptron algorithm is guaranteed to have a **finite mistake bound** ...

- *whenever* $d_* > \frac{2}{c}$, *but no prior knowledge on* (y^*, b^*) *exists:*
  select $\mathbb{L} = \mathbb{R}^d \times \mathbb{R}$ *to get* $|\mathcal{M}| \leq \frac{\|y_*\|_2^2 + b_*^2}{\|y_*\|_*^2} \frac{\tilde{D}^2+1}{(d_* - 2/c)^2}$;
- *whenever* $b_* = 0$ *is known a priori and* $\|\cdot\|$ *is* ℓ_2 *norm**:
  select $\mathbb{L} = \mathbb{R}^d \times \{0\}$ *to get* $|\mathcal{M}| \leq \frac{\tilde{D}^2+1}{d_*^2}$;
- *whenever* $y^* \in \mathbb{R}^d_+$ *is known a priori and* $\|\cdot\|$ *is any* ℓ_p *norm:*
  select $\mathbb{L} = \mathbb{R}^d_+ \times \mathbb{R}$ *to get* $|\mathcal{M}| \leq \frac{\|y_*\|_2^2 + b_*^2}{\|y_*\|_*^2} \frac{\tilde{D}^2+1}{d_*^2}$.

*Recovers mistake bounds from [Ahmadi et al., 2021] given for this case.

Strategic Perceptron: Summary

Projected strategic perceptron

(+) computationally cheap

(+) finite mistake bound

Strategic Perceptron: Summary

Projected strategic perceptron

(+) computationally cheap

(+) finite mistake bound

(−) update only when making a mistake

(−) not guaranteed to converge to (y_*, b_*); no margin guarantee

Strategic Perceptron: Summary

Projected strategic perceptron

(+) computationally cheap

(+) finite mistake bound

(−) update only when making a mistake

(−) not guaranteed to converge to (y_*, b_*); no margin guarantee

Question: How can we improve?

Strategic Perceptron: Summary

Projected strategic perceptron

(+) computationally cheap

(+) finite mistake bound

(−) update only when making a mistake

(−) not guaranteed to converge to (y_*, b_*); no margin guarantee

Question: How can we improve?

- strategic perceptron uses only information from current iteration in its update
- idea: make use of all historical data: $\{(r_\tau, \ell_\tau, y_\tau, b_\tau)\}_{\tau\in[t]}$

Strategic Max Margin: Algorithm

Strategic max-margin (SMM) algorithm

Call initialization subroutine. At iteration $t = 1, 2, \ldots$

Step 1. Receive manipulated data r_t and predict by $\widetilde{\text{label}}(r_t, y_t, b_t)$.

Step 2. Receive $\text{label}(A_t)$ and compute the proxy $s(A_t, y_t, b_t)$.

Step 3. Update to (y_{t+1}, b_{t+1}) by solving the proxy margin maximization problem

$$(y_{t+1}, b_{t+1}) \in \underset{\|y\|_* \leq 1, b \in \mathbb{R}}{\arg\max} \; \min_{\tau \in [t]} \left\{ \text{label}(A_\tau) \cdot \left(y^\top s(A_\tau, y_\tau, b_\tau) + b \right) \right\}. \qquad (\mathrm{P}_t)$$

Strategic Max Margin: Algorithm

Strategic max-margin (SMM) algorithm

Call initialization subroutine. At iteration $t = 1, 2, \ldots$

Step 1. Receive manipulated data r_t and predict by $\widetilde{\text{label}}(r_t, y_t, b_t)$.

Step 2. Receive label(A_t) and compute the proxy $s(A_t, y_t, b_t)$.

Step 3. Update to (y_{t+1}, b_{t+1}) by solving the proxy margin maximization problem

$$(y_{t+1}, b_{t+1}) \in \underset{\|y\|_* \le 1, b \in \mathbb{R}}{\arg\max} \min_{\tau \in [t]} \left\{ \text{label}(A_\tau) \cdot \left(y^\top s(A_\tau, y_\tau, b_\tau) + b \right) \right\}. \qquad (\mathrm{P}_t)$$

Strategic Max Margin: Results

Theorem (informal)

SMM algorithm is guaranteed to have

- *a* **finite mistake bound***; and*
- *a* **finite manipulation bound** *whenever* $d_* > \frac{2}{c}$.

Assumption (distributional separability)

$\{A_t\}_{t\in\mathbb{N}}$ *are i.i.d. samples from a probability distribution with support* $\mathcal{A}$*, and the max margin classifier on* $\{(A, \text{label}(A)) : A \in \mathcal{A}\}$ *is* (y_*, b_*) *achieving a margin of* $d_* > 0$.

Theorem (informal)

If $d_* > \frac{2}{c}$*, SMM algorithm guarantees* (y_t, b_t) **converges** *to* $(y_*, b_*)/\|y_*\|_*$ *almost surely.*

Strategic Max Margin: Proof Highlights

- Recall the proxy margin maximization problem

$$\max_{\|y\|_* \leq 1, b \in \mathbb{R}} \min_{\tau \in [t]} \left\{ \text{label}(A_\tau) \cdot \left(y^\top s(A_\tau, y_\tau, b_\tau) + b \right) \right\}. \qquad (\mathrm{P}_t)$$

Strategic Max Margin: Proof Highlights

- Recall the proxy margin maximization problem

$$\max_{\|y\|_* \le 1, b \in \mathbb{R}} \min_{\tau \in [t]} \left\{ \text{label}(A_\tau) \cdot \left(y^\top s(A_\tau, y_\tau, b_\tau) + b \right) \right\}. \qquad (\mathrm{P}_t)$$

- Define $\widetilde{\mathcal{A}}_t^+ := \{ s(A_\tau, y_\tau, b_\tau) : \tau \in [t] \text{ s.t. } \text{label}(A_\tau) = +1 \}$ and also $\widetilde{\mathcal{A}}_t^-$.

Strategic Max Margin: Proof Highlights

- Recall the proxy margin maximization problem

$$\max_{\|y\|_* \le 1, b \in \mathbb{R}} \min_{\tau \in [t]} \left\{ \text{label}(A_\tau) \cdot \left(y^\top s(A_\tau, y_\tau, b_\tau) + b \right) \right\}. \qquad (\text{P}_t)$$

- Define $\widetilde{\mathcal{A}}_t^+ := \{ s(A_\tau, y_\tau, b_\tau) : \tau \in [t] \text{ s.t. } \text{label}(A_\tau) = +1 \}$ and also $\widetilde{\mathcal{A}}_t^-$.
- Then (P_t) is

$$\max_{\|y\|_* \le 1, b \in \mathbb{R}} h(y, b; \widetilde{\mathcal{A}}_t^+, \widetilde{\mathcal{A}}_t^-)$$

where $$h(y, b; \widetilde{\mathcal{A}}_t^+, \widetilde{\mathcal{A}}_t^-) := \min \left\{ \min_{x \in \widetilde{\mathcal{A}}_t^+} \left\{ y^\top x + b \right\}, \min_{x \in \widetilde{\mathcal{A}}_t^-} \left\{ -y^\top x - b \right\} \right\}.$$

Strategic Max Margin: Proof Highlights

$$h(y, b; \widetilde{\mathcal{A}}^+, \widetilde{\mathcal{A}}^-) = \min \left\{ \min_{x \in \widetilde{\mathcal{A}}^+} \{y^\top x + b\}, \min_{x \in \widetilde{\mathcal{A}}^-} \{-y^\top x - b\} \right\}$$

Lemma (witness points, classifier alignment)

Suppose $\widetilde{\mathcal{A}}^+, \widetilde{\mathcal{A}}^- \subset \mathbb{R}^d$ *separable with positive margin. Then,*

- $(\tilde{y}, \tilde{b}) \in \arg\max_{\|y\|_* \leq 1, b \in \mathbb{R}} h(y, b; \widetilde{\mathcal{A}}^+, \widetilde{\mathcal{A}}^-)$ *satisfy* $\|\tilde{y}\|_* = 1$*;*

Strategic Max Margin: Proof Highlights

$$h(y, b; \widetilde{\mathcal{A}}^+, \widetilde{\mathcal{A}}^-) = \min\left\{ \min_{x \in \widetilde{\mathcal{A}}^+} \{y^\top x + b\}, \min_{x \in \widetilde{\mathcal{A}}^-} \{-y^\top x - b\} \right\}$$

Lemma (witness points, classifier alignment)

Suppose $\widetilde{\mathcal{A}}^+, \widetilde{\mathcal{A}}^- \subset \mathbb{R}^d$ *separable with positive margin. Then,*

- $(\tilde{y}, \tilde{b}) \in \arg\max_{\|y\|_* \le 1, b \in \mathbb{R}} h(y, b; \widetilde{\mathcal{A}}^+, \widetilde{\mathcal{A}}^-)$ *satisfy* $\|\tilde{y}\|_* = 1$;
- $\exists$ *witness points* $\tilde{x}^+ \in \mathrm{conv}(\widetilde{\mathcal{A}}^+)$ *and* $\tilde{x}^- \in \mathrm{conv}(\widetilde{\mathcal{A}}^-)$ *s.t.*

$$\tilde{y}^\top(\tilde{x}^+ - \tilde{x}^-) = \|\tilde{x}^+ - \tilde{x}^-\| \cdot \|\tilde{y}\|_*, \text{ and}$$

$$\tilde{d}\|\tilde{y}\|_* = \tilde{y}^\top \tilde{x}^+ + \tilde{b} = -\tilde{y}^\top \tilde{x}^- - \tilde{b};$$

Strategic Max Margin: Proof Highlights

$$h(y, b; \widetilde{\mathcal{A}}^+, \widetilde{\mathcal{A}}^-) = \min\left\{ \min_{x \in \widetilde{\mathcal{A}}^+} \{y^\top x + b\}, \min_{x \in \widetilde{\mathcal{A}}^-} \{-y^\top x - b\} \right\}$$

Lemma (witness points, classifier alignment)

Suppose $\widetilde{\mathcal{A}}^+, \widetilde{\mathcal{A}}^- \subset \mathbb{R}^d$ *separable with positive margin. Then,*

- $(\tilde{y}, \tilde{b}) \in \arg\max_{\|y\|_* \le 1, b \in \mathbb{R}} h(y, b; \widetilde{\mathcal{A}}^+, \widetilde{\mathcal{A}}^-)$ *satisfy* $\|\tilde{y}\|_* = 1$;
- $\exists$ *witness points* $\tilde{x}^+ \in \operatorname{conv}(\widetilde{\mathcal{A}}^+)$ *and* $\tilde{x}^- \in \operatorname{conv}(\widetilde{\mathcal{A}}^-)$ *s.t.*
$$\tilde{y}^\top(\tilde{x}^+ - \tilde{x}^-) = \|\tilde{x}^+ - \tilde{x}^-\| \cdot \|\tilde{y}\|_*, \text{ and}$$
$$\tilde{d}\|\tilde{y}\|_* = \tilde{y}^\top \tilde{x}^+ + \tilde{b} = -\tilde{y}^\top \tilde{x}^- - \tilde{b};$$
- *whenever* $\|\cdot\|$ *and its dual norm* $\|\cdot\|_*$ *are strictly convex,*

Strategic Max Margin: Proof Highlights

$$h(y, b; \widetilde{\mathcal{A}}^+, \widetilde{\mathcal{A}}^-) = \min \left\{ \min_{x \in \widetilde{\mathcal{A}}^+} \{y^\top x + b\}, \min_{x \in \widetilde{\mathcal{A}}^-} \{-y^\top x - b\} \right\}$$

Lemma (witness points, classifier alignment)

Suppose $\widetilde{\mathcal{A}}^+, \widetilde{\mathcal{A}}^- \subset \mathbb{R}^d$ *separable with positive margin. Then,*

- $(\tilde{y}, \tilde{b}) \in \arg\max_{\|y\|_* \leq 1, b \in \mathbb{R}} h(y, b; \widetilde{\mathcal{A}}^+, \widetilde{\mathcal{A}}^-)$ *satisfy* $\|\tilde{y}\|_* = 1$;
- $\exists$ *witness points* $\tilde{x}^+ \in \mathrm{conv}(\widetilde{\mathcal{A}}^+)$ *and* $\tilde{x}^- \in \mathrm{conv}(\widetilde{\mathcal{A}}^-)$ *s.t.*
$$\tilde{y}^\top(\tilde{x}^+ - \tilde{x}^-) = \|\tilde{x}^+ - \tilde{x}^-\| \cdot \|\tilde{y}\|_*, \text{ and}$$
$$\tilde{d}\|\tilde{y}\|_* = \tilde{y}^\top \tilde{x}^+ + \tilde{b} = -\tilde{y}^\top \tilde{x}^- - \tilde{b};$$
- *whenever* $\|\cdot\|$ *and its dual norm* $\|\cdot\|_*$ *are strictly convex,*
  - $(\tilde{y}, \tilde{b})$ *is unique; and*

Strategic Max Margin: Proof Highlights

$$h(y, b; \widetilde{\mathcal{A}}^+, \widetilde{\mathcal{A}}^-) = \min\left\{ \min_{x \in \widetilde{\mathcal{A}}^+} \{y^\top x + b\}, \min_{x \in \widetilde{\mathcal{A}}^-} \{-y^\top x - b\} \right\}$$

Lemma (witness points, classifier alignment)

Suppose $\widetilde{\mathcal{A}}^+, \widetilde{\mathcal{A}}^- \subset \mathbb{R}^d$ *separable with positive margin. Then,*

- $(\tilde{y}, \tilde{b}) \in \arg\max_{\|y\|_* \leq 1, b \in \mathbb{R}} h(y, b; \widetilde{\mathcal{A}}^+, \widetilde{\mathcal{A}}^-)$ *satisfy* $\|\tilde{y}\|_* = 1$;
- $\exists$ *witness points* $\tilde{x}^+ \in \mathrm{conv}(\widetilde{\mathcal{A}}^+)$ *and* $\tilde{x}^- \in \mathrm{conv}(\widetilde{\mathcal{A}}^-)$ *s.t.*
$$\tilde{y}^\top(\tilde{x}^+ - \tilde{x}^-) = \|\tilde{x}^+ - \tilde{x}^-\| \cdot \|\tilde{y}\|_*, \text{ and}$$
$$\tilde{d}\|\tilde{y}\|_* = \tilde{y}^\top \tilde{x}^+ + \tilde{b} = -\tilde{y}^\top \tilde{x}^- - \tilde{b};$$
- *whenever* $\|\cdot\|$ *and its dual norm* $\|\cdot\|_*$ *are strictly convex,*
  - $(\tilde{y}, \tilde{b})$ *is unique; and*
  - *any* $(\bar{y}, \bar{b})$ *satisfying* $h\left(\bar{y}, \bar{b}; \widetilde{\mathcal{A}}^+, \widetilde{\mathcal{A}}^-\right) \geq \bar{d} > 0$ *satisfies* $\bar{y}^\top v(\tilde{y}) \geq (\bar{d}/\tilde{d}) > 0.$

Strategic Max Margin: Proof Highlights

Suppose separability and strict convexity of the norms hold.

- At time t, SMM generates (y_t, b_t) with margin d_t. Then, for all t

Strategic Max Margin: Proof Highlights

Suppose separability and strict convexity of the norms hold.

- At time t, SMM generates (y_t, b_t) with margin d_t. Then, for all t
  - y_t will be well-aligned with y_*, i.e., $y_*^\top v(y_t) \geq \|y_*\|_* \frac{d_*}{d_t} > 0$;

Strategic Max Margin: Proof Highlights

Suppose separability and strict convexity of the norms hold.

- At time t, SMM generates (y_t, b_t) with margin d_t. Then, for all t
  - y_t will be well-aligned with y_*, i.e., $y_*^\top v(y_t) \geq \|y_*\|_* \frac{d_*}{d_t} > 0$;
  - $d_{t+1} \geq d_*$;

Strategic Max Margin: Proof Highlights

Suppose separability and strict convexity of the norms hold.

- At time t, SMM generates (y_t, b_t) with margin d_t. Then, for all t
  - y_t will be well-aligned with y_*, i.e., $y_*^\top v(y_t) \geq \|y_*\|_* \frac{d_*}{d_t} > 0$;
  - $d_{t+1} \geq d_*$;
  - if $\text{label}(A_t)[y_t^\top s(A_t, y_t, b_t) + b_t] \leq a\|y_t\|_*$ holds for $a < d_*$, then $d_{t+1} \leq \kappa(a, d_*, \tilde{D})d_t$.
    $\big(\kappa(a, d_*, \tilde{D}) \in (0, 1)$ is a parameter based on the geometry of the problem, margin and size of data$\big)$

Strategic Max Margin: Proof Highlights

Suppose separability and strict convexity of the norms hold.

- At time t, SMM generates (y_t, b_t) with margin d_t. Then, for all t
  - y_t will be well-aligned with y_*, i.e., $y_*^\top v(y_t) \geq \|y_*\|_* \frac{d_*}{d_t} > 0$;
  - $d_{t+1} \geq d_*$;
  - if $\text{label}(A_t)[y_t^\top s(A_t, y_t, b_t) + b_t] \leq a\|y_t\|_*$ holds for $a < d_*$, then $d_{t+1} \leq \kappa(a, d_*, \tilde{D})d_t$.
    $\big(\kappa(a, d_*, \tilde{D}) \in (0,1)$ is a parameter based on the geometry of the problem, margin and size of data$\big)$

$\Longrightarrow$ # of mistakes $\mathcal{M}$ satisfies $|\mathcal{M}| \leq \frac{\log(d_1/d_*)}{\log\big(1/\kappa(0,d_*,\tilde{D})\big)} < \infty$;

Strategic Max Margin: Proof Highlights

Suppose separability and strict convexity of the norms hold.

- At time t, SMM generates (y_t, b_t) with margin d_t. Then, for all t
  - y_t will be well-aligned with y_*, i.e., $y_*^\top v(y_t) \geq \|y_*\|_* \frac{d_*}{d_t} > 0$;
  - $d_{t+1} \geq d_*$;
  - if $\text{label}(A_t)[y_t^\top s(A_t, y_t, b_t) + b_t] \leq a\|y_t\|_*$ holds for $a < d_*$, then $d_{t+1} \leq \kappa(a, d_*, \tilde{D}) d_t$.
    $\big(\kappa(a, d_*, \tilde{D}) \in (0,1)$ is a parameter based on the geometry of the problem, margin and size of data$\big)$

$\Longrightarrow$ # of mistakes $\mathcal{M}$ satisfies $|\mathcal{M}| \leq \frac{\log(d_1/d_*)}{\log\left(1/\kappa(0, d_*, \tilde{D})\right)} < \infty$;

$\Longrightarrow$ # of manipulations of negative data $\mathcal{N}^-$, (as well as $\mathcal{N}^+$ whenever $d_* > 2/c$) satisfy

$$|\mathcal{N}^-| \leq \frac{\log(d_1/d_*)}{\log\left(1/\kappa\left(0, d_*, \tilde{D}\right)\right)} < \infty, \qquad |\mathcal{N}^+| \leq \frac{\log(d_1/d_*)}{\log\left(1/\kappa\left(2/c, d_*, \tilde{D}\right)\right)} < \infty;$$

Strategic Max Margin: Proof Highlights

Lemma (uniform convergence)

Let

$$\widetilde{\mathcal{A}}_1^+ \subseteq \widetilde{\mathcal{A}}_2^+ \subseteq \ldots \subseteq \widetilde{\mathcal{A}}_\infty^+ \subset \mathbb{R}^d$$
$$\widetilde{\mathcal{A}}_1^- \subseteq \widetilde{\mathcal{A}}_2^- \subseteq \ldots \subseteq \widetilde{\mathcal{A}}_\infty^- \subset \mathbb{R}^d.$$

If both sets $\widetilde{\mathcal{A}}_\infty^+$ *and* $\widetilde{\mathcal{A}}_\infty^-$ *are bounded, then* $h_t(y, b) := h\left(y, b; \widetilde{\mathcal{A}}_t^+, \widetilde{\mathcal{A}}_t^-\right)$ *converge uniformly to* $h_\infty(y, b) := h\left(y, b; \widetilde{\mathcal{A}}_\infty^+, \widetilde{\mathcal{A}}_\infty^-\right)$ *over any compact domain* $\mathcal{D} \subset \mathbb{R}^d \times \mathbb{R}$.

Strategic Max Margin: Proof Highlights

Lemma (uniform convergence)

Let

$$\widetilde{\mathcal{A}}_1^+ \subseteq \widetilde{\mathcal{A}}_2^+ \subseteq \ldots \subseteq \widetilde{\mathcal{A}}_\infty^+ \subset \mathbb{R}^d$$
$$\widetilde{\mathcal{A}}_1^- \subseteq \widetilde{\mathcal{A}}_2^- \subseteq \ldots \subseteq \widetilde{\mathcal{A}}_\infty^- \subset \mathbb{R}^d.$$

If both sets $\widetilde{\mathcal{A}}_\infty^+$ *and* $\widetilde{\mathcal{A}}_\infty^-$ *are bounded, then* $h_t(y,b) := h\left(y, b; \widetilde{\mathcal{A}}_t^+, \widetilde{\mathcal{A}}_t^-\right)$ *converge uniformly to* $h_\infty(y,b) := h\left(y, b; \widetilde{\mathcal{A}}_\infty^+, \widetilde{\mathcal{A}}_\infty^-\right)$ *over any compact domain* $\mathcal{D} \subset \mathbb{R}^d \times \mathbb{R}$.

- When data A_t is bounded, i.e., $\|A_t\| \leq D$, we get uniform conv. to $h_\infty(y,b)$.

Strategic Max Margin: Proof Highlights

Lemma (uniform convergence)

Let
$$\widetilde{\mathcal{A}}_1^+ \subseteq \widetilde{\mathcal{A}}_2^+ \subseteq \ldots \subseteq \widetilde{\mathcal{A}}_\infty^+ \subset \mathbb{R}^d$$
$$\widetilde{\mathcal{A}}_1^- \subseteq \widetilde{\mathcal{A}}_2^- \subseteq \ldots \subseteq \widetilde{\mathcal{A}}_\infty^- \subset \mathbb{R}^d.$$

If both sets $\widetilde{\mathcal{A}}_\infty^+$ and $\widetilde{\mathcal{A}}_\infty^-$ are bounded, then $h_t(y,b) := h\left(y, b; \widetilde{\mathcal{A}}_t^+, \widetilde{\mathcal{A}}_t^-\right)$ converge uniformly to $h_\infty(y,b) := h\left(y, b; \widetilde{\mathcal{A}}_\infty^+, \widetilde{\mathcal{A}}_\infty^-\right)$ over any compact domain $\mathcal{D} \subset \mathbb{R}^d \times \mathbb{R}$.

- When data A_t is bounded, i.e., $\|A_t\| \leq D$, we get uniform conv. to $h_\infty(y,b)$.
- When $d_* > 2/c$, $\implies$ finitely many mistakes and manipulations $\implies$ $\exists t_0 \in \mathbb{N}$ s.t. $r(A_t, y_t, b_t) = s(A_t, y_t, b_t) = A_t$ for all $t \geq t_0$ a.s.

Strategic Max Margin: Proof Highlights

Lemma (uniform convergence)

Let

$$\widetilde{\mathcal{A}}_1^+ \subseteq \widetilde{\mathcal{A}}_2^+ \subseteq \ldots \subseteq \widetilde{\mathcal{A}}_\infty^+ \subset \mathbb{R}^d$$
$$\widetilde{\mathcal{A}}_1^- \subseteq \widetilde{\mathcal{A}}_2^- \subseteq \ldots \subseteq \widetilde{\mathcal{A}}_\infty^- \subset \mathbb{R}^d.$$

If both sets $\widetilde{\mathcal{A}}_\infty^+$ and $\widetilde{\mathcal{A}}_\infty^-$ are bounded, then $h_t(y,b) := h\left(y, b; \widetilde{\mathcal{A}}_t^+, \widetilde{\mathcal{A}}_t^-\right)$ converge uniformly to $h_\infty(y,b) := h\left(y, b; \widetilde{\mathcal{A}}_\infty^+, \widetilde{\mathcal{A}}_\infty^-\right)$ over any compact domain $\mathcal{D} \subset \mathbb{R}^d \times \mathbb{R}$.

- When data A_t is bounded, i.e., $\|A_t\| \leq D$, we get uniform conv. to $h_\infty(y,b)$.
- When $d_* > 2/c$, $\implies$ finitely many mistakes and manipulations $\implies$ $\exists t_0 \in \mathbb{N}$ s.t. $r(A_t, y_t, b_t) = s(A_t, y_t, b_t) = A_t$ for all $t \geq t_0$ a.s.
- Distributional separability will ensure $\{A_t : t \geq t_0\}$ is dense in $\mathcal{A}$ a.s.
- (y_*, b_*) maximizes h_∞ a.s. (recall also that h_∞ has a unique maximizer)
- Then, uniform conv. of $h_t \to h_\infty$ implies $(y_t, b_t) \to (y_*, b_*)$ almost surely.

Strategic Max Margin: Summary

Strategic max-margin algorithm

(+) finite mistake and manipulation bounds

(+) convergence to the max margin classifier (y_*, b_*)

Strategic Max Margin: Summary

Strategic max-margin algorithm

(+) finite mistake and manipulation bounds

(+) convergence to the max margin classifier (y_*, b_*)

(−) requires solving an optimization problem at each iteration

Strategic Max Margin: Summary

Strategic max-margin algorithm

(+) finite mistake and manipulation bounds

(+) convergence to the max margin classifier (y_*, b_*)

(−) requires solving an optimization problem at each iteration

Question: Can we reduce the computation cost?

Strategic Max Margin: Summary

Strategic max-margin algorithm

(+) finite mistake and manipulation bounds

(+) convergence to the max margin classifier (y_*, b_*)

(−) requires solving an optimization problem at each iteration

Question: Can we reduce the computation cost?

- idea: Joint estimation-optimization[7]
  - given a sequence of optimization problems that converges to a target problem
  - perform one update (e.g., one step of gradient descent) based on the problem defined by the current data

Gradient-based SMM: Algorithm

Gradient-based strategic max-margin algorithm (Gradient SMM)

Call initialization subroutine. Select stepsizes $\{\gamma_t\}$. At iteration $t = 1, 2, \ldots$

Step 1. Receive manipulated data r_t and predict by $\widetilde{\text{label}}(r_t, y_t, b_t)$.

Step 2. Receive label(A_t) and compute the proxy $s(A_t, y_t, b_t)$.

Step 3. Update to (y_{t+1}, b_{t+1}) by

$$s_t^+ \in \operatorname*{arg\,max}_{s \in \widetilde{\mathcal{A}}_t^+} z_t^\top s, \quad s_t^- \in \operatorname*{arg\,min}_{s \in \widetilde{\mathcal{A}}_t^-} z_t^\top s, \quad z_{t+1} = \text{Proj}_{B_{\|\cdot\|_2}}\left(z_t + \gamma_t(s_t^+ - s_t^-)\right)$$

$$\text{and} \quad y_{t+1} = \frac{\sum_{\tau \in [t+1]} \gamma_\tau z_\tau}{\sum_{\tau \in [t+1]} \gamma_\tau}, \quad b_{t+1} = -\frac{1}{2}\left(\min_{s \in \widetilde{\mathcal{A}}_t^+} y_{t+1}^\top s + \max_{s \in \widetilde{\mathcal{A}}_t^-} y_{t+1}^\top s\right).$$

- key idea:

$$h(y, b; \widetilde{\mathcal{A}}^+, \widetilde{\mathcal{A}}^-) = \tfrac{1}{2}\left(\min_{x \in \widetilde{\mathcal{A}}^+} y^\top x - \max_{x \in \widetilde{\mathcal{A}}^-} y^\top x\right) - \left| b + \tfrac{1}{2}\left(\min_{x \in \widetilde{\mathcal{A}}^+} y^\top x + \max_{x \in \widetilde{\mathcal{A}}^-} y^\top x\right)\right|.$$

Gradient-based SMM: Algorithm

Gradient-based strategic max-margin algorithm (Gradient SMM)

Call initialization subroutine. Select stepsizes $\{\gamma_t\}$. At iteration $t = 1, 2, \ldots$

Step 1. Receive manipulated data r_t and predict by $\text{label}(r_t, y_t, b_t)$.

Step 2. Receive $\text{label}(A_t)$ and compute the proxy $s(A_t, y_t, b_t)$.

Step 3. Update to (y_{t+1}, b_{t+1}) by

$$s_t^+ \in \arg\max_{s \in \widetilde{\mathcal{A}}_t^+} z_t^\top s, \quad s_t^- \in \arg\min_{s \in \widetilde{\mathcal{A}}_t^-} z_t^\top s, \quad z_{t+1} = \text{Proj}_{B_{\|\cdot\|_2}}\left(z_t + \gamma_t(s_t^+ - s_t^-)\right)$$

$$\text{and} \quad y_{t+1} = \frac{\sum_{\tau \in [t+1]} \gamma_\tau z_\tau}{\sum_{\tau \in [t+1]} \gamma_\tau}, \quad b_{t+1} = -\frac{1}{2}\left(\min_{s \in \widetilde{\mathcal{A}}_t^+} y_{t+1}^\top s + \max_{s \in \widetilde{\mathcal{A}}_t^-} y_{t+1}^\top s\right).$$

- key idea:
$h(y, b; \widetilde{\mathcal{A}}^+, \widetilde{\mathcal{A}}^-) = \frac{1}{2}\left(\min_{x \in \widetilde{\mathcal{A}}^+} y^\top x - \max_{x \in \widetilde{\mathcal{A}}^-} y^\top x\right) - \left|b + \frac{1}{2}\left(\min_{x \in \widetilde{\mathcal{A}}^+} y^\top x + \max_{x \in \widetilde{\mathcal{A}}^-} y^\top x\right)\right|.$

Gradient-based SMM: Results

Assumption (distributional separability)

$\{A_t\}_{t\in\mathbb{N}}$ are i.i.d. samples from a probability distribution with support $\mathcal{A}$, and the max margin classifier on $\{(A, \text{label}(A)) : A \in \mathcal{A}\}$ is (y_*, b_*) achieving a margin of $d_* > 0$.

Theorem (informal)

Suppose $\|\cdot\|$ is the ℓ_2 norm and $\gamma_t = \gamma_0/\sqrt{t}$. Then, gradient SMM algorithm is guaranteed to

- *make* **finitely many mistakes** *almost surely,*
- *induce* **finite manipulations** *whenever $d_* > \frac{2}{c}$, and*
- **converge** *to $(y_*, b_*)/\|y_*\|_2$ almost surely whenever $d_* > \frac{2}{c}$.*

Theoretical Guarantees: Summary

- Suppose $\|\cdot\|$ is the ℓ_2 norm. Then,

---

*under a priori assumption of $b_* = 0$

†under the assumption $d_* > 2/c$

‡under distributional separability assumption

Theoretical Guarantees: Summary

- Suppose $\|\cdot\|$ is the ℓ_2 norm. Then,

Algorithm	**Mistake**	**Manipulation**	**Margin**
S-perceptron	finite bound*	–	–

*under a priori assumption of $b_* = 0$
†under the assumption $d_* > 2/c$
‡under distributional separability assumption

Theoretical Guarantees: Summary

- Suppose $\|\cdot\|$ is the ℓ_2 norm. Then,

Algorithm	Mistake	Manipulation	Margin
S-perceptron	finite bound*	–	–
SMM	finite bound	finite bound†	convergence†‡

*under a priori assumption of $b_* = 0$
†under the assumption $d_* > 2/c$
‡under distributional separability assumption

Theoretical Guarantees: Summary

- Suppose $\|\cdot\|$ is the ℓ_2 norm. Then,

Algorithm	Mistake	Manipulation	Margin
S-perceptron	finite bound*	–	–
SMM	finite bound	finite bound†	convergence†‡
Gradient SMM	finite‡	finite†‡	convergence†‡

*under a priori assumption of $b_* = 0$
†under the assumption $d_* > 2/c$
‡under distributional separability assumption

Computational Study - Setting

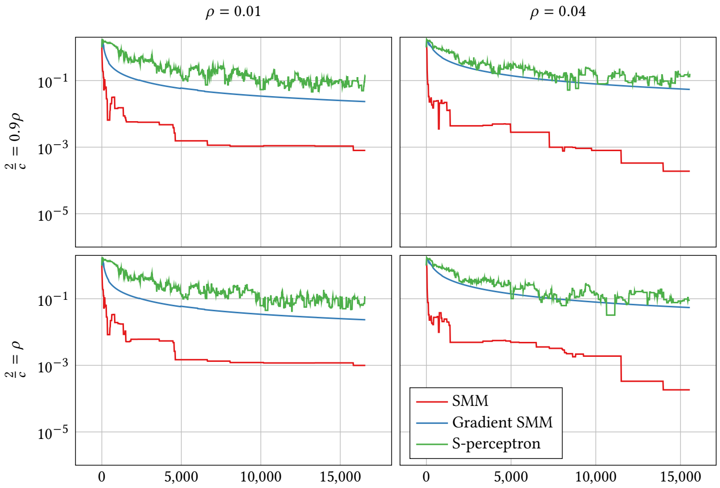
- Bank loan application data from [8] (collected by an online platform `Prosper`):
  - $d = 6$ continuous features (bank card utilization, credit history length, etc.)
  - $20,222$ data points (41.70% have $+1$ labels)
  - Preprocessed to ensure separability and a margin of at least $\rho > 0$

Computational Study - Setting

- Bank loan application data from [8] (collected by an online platform `Prosper`):
  - $d = 6$ continuous features (bank card utilization, credit history length, etc.)
  - $20,222$ data points (41.70% have $+1$ labels)
  - Preprocessed to ensure separability and a margin of at least $\rho > 0$
- Tested the impact of
  - Margin $\rho \in \{0.01, 0.02, 0.04\}$,
  - Cost of manipulation $2/c \in \{0.9,\ 1.0,\ 1.1\} \cdot \rho$, and
  - Noise in agent responses: learner observes $r(A_t, y_t, b_t) + \varepsilon_t$, where $\varepsilon_t \sim \mathcal{N}(0, \sigma^2 I_d)$ is i.i.d. Gaussian noise with $\sigma \in \left\{0, 10^{-3}, 10^{-2}\right\}$.

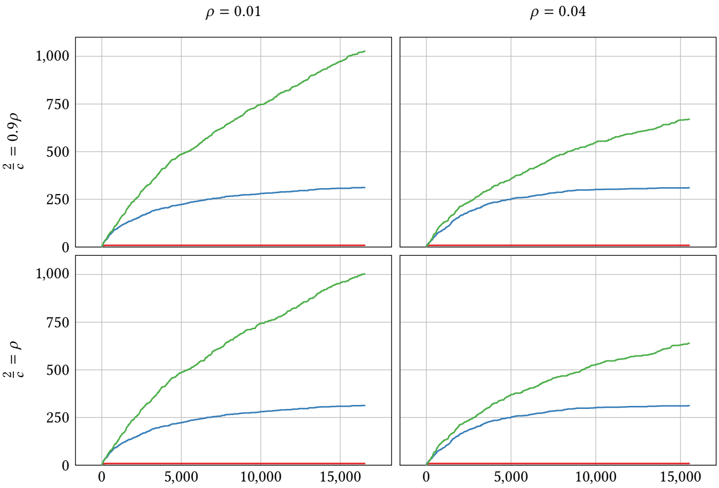
Performance Comparison

- No noise ($\sigma = 0$), performance metric: **convergence to max-margin classifier**



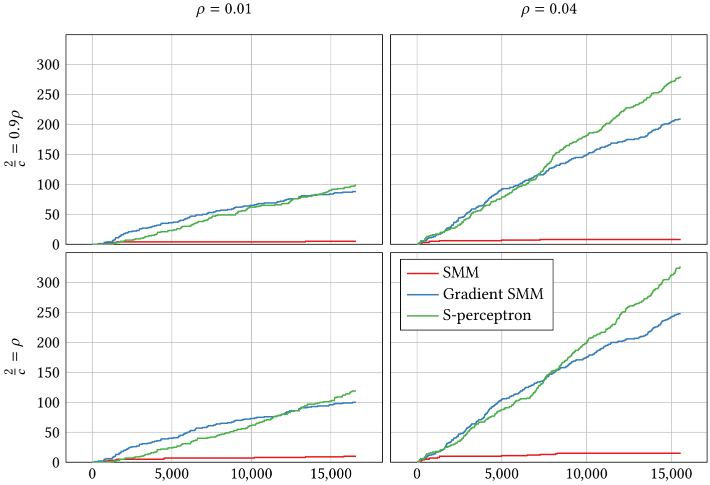
Performance Comparison

- No noise ($\sigma = 0$), performance metric: **# of mistakes**



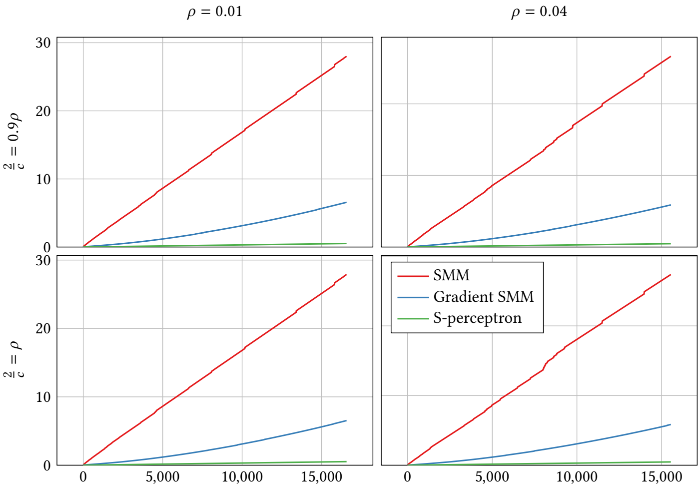
Performance Comparison

- No noise ($\sigma = 0$), performance metric: **# of manipulations**



Performance Comparison

- No noise ($\sigma = 0$), performance metric: **solution time (seconds)**

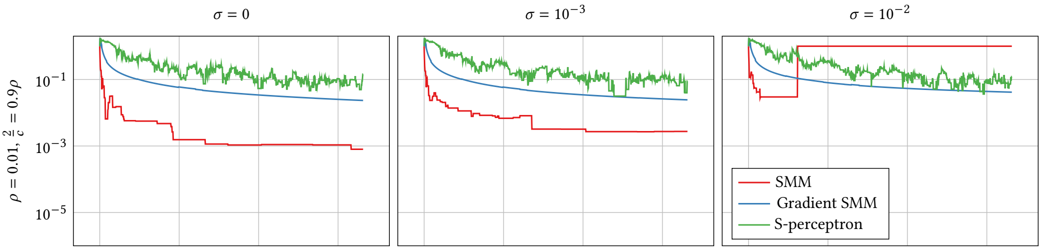


Performance Comparison: Noisy Response

- learner observes $r(A_t, y_t, b_t) + \varepsilon_t$, where $\varepsilon_t \sim \mathcal{N}(0, \sigma^2 I_d)$ is i.i.d. Gaussian noise with σ

Performance Comparison: Noisy Response

- learner observes $r(A_t, y_t, b_t) + \varepsilon_t$, where $\varepsilon_t \sim \mathcal{N}(0, \sigma^2 I_d)$ is i.i.d. Gaussian noise with σ
- varying noise level in agent responses: $\sigma \in \{0,\ 10^{-3},\ 10^{-2}\}$



Performance Comparison: Noisy Response

- learner observes $r(A_t, y_t, b_t) + \varepsilon_t$, where $\varepsilon_t \sim \mathcal{N}(0, \sigma^2 I_d)$ is i.i.d. Gaussian noise with σ
- varying noise level in agent responses: $\sigma \in \{0,\ 10^{-3},\ 10^{-2}\}$
- performance metric: **convergence to max-margin classifier**

Computational Study - Summary

- Summary of **numerical performance** (no noise):

Algorithm	Margin	Mistake	Manipulation	Time
S-perceptron	(−/+)	(−−)	(−−)	(++)
SMM	(++)	(++)	(++)	(−−)
Gradient SMM	(+−)	(+)	(−)	(+)

  - SMM performs the best in terms of all metrics except solution time.
  - Gradient-based SMM does better than strategic perceptron in terms of convergence and # of mistakes, and eventually in terms of # manipulations as well.

Computational Study - Summary

- Summary of **numerical performance** (no noise):

Algorithm	Margin	Mistake	Manipulation	Time
S-perceptron	(−/+)	(−−)	(−−)	(++)
SMM	(++)	(++)	(++)	(−−)
Gradient SMM	(+−)	(+)	(−)	(+)

  - SMM performs the best in terms of all metrics except solution time.
  - Gradient-based SMM does better than strategic perceptron in terms of convergence and # of mistakes, and eventually in terms of # manipulations as well.
- SMM is robust to low magnitude of noise, but not high noise.
- Gradient SMM and S-perceptron appear to be quite robust to noise.

Conclusion

Summary

- **New algorithms** for classification in strategic setting with theoretical guarantees on **# of mistakes, # of manipulations** and **margin**

Conclusion

Summary

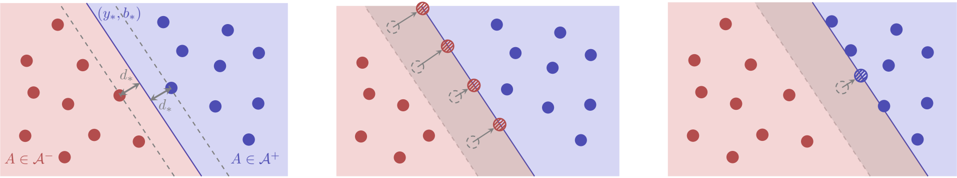
- **New algorithms** for classification in strategic setting with theoretical guarantees on **# of mistakes, # of manipulations** and **margin**

Future outlook

- model variants
  - alternative manipulation models (other cost structures, discrete features via manipulation graph, . . .)
  - unknown utility function
  - strategic classification for nonlinear classifiers
- connections with Stackelberg games more generally
- more tools to handle strategic behavior effectively

Thank you!

fkilinc@andrew.cmu.edu



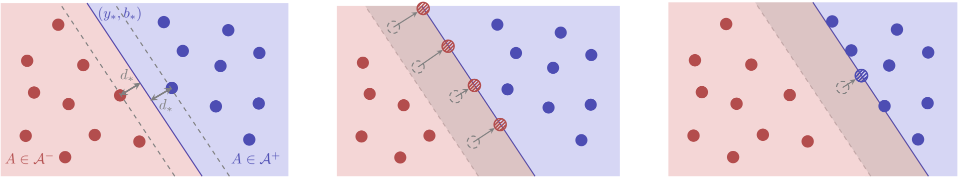
[Shen et al., 2024]

Mistake, Manipulation, and Margin Guarantees in Online Strategic Classification (March 2024).

arXiv:2403.18176.

Questions?

fkilinc@andrew.cmu.edu



[Shen et al., 2024]

Mistake, Manipulation, and Margin Guarantees in Online Strategic Classification (March 2024).

arXiv:2403.18176.

References I

[1] Moritz Hardt et al. "Strategic Classification". In: *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science*. Cambridge Massachusetts USA: ACM, Jan. 2016, pp. 111–122.

[2] Jinshuo Dong et al. "Strategic Classification from Revealed Preferences". In: *Proceedings of the 2018 ACM Conference on Economics and Computation*. Ithaca NY USA: ACM, June 2018, pp. 55–70.

[3] Yiling Chen, Yang Liu, and Chara Podimata. "Learning Strategy-Aware Linear Classifiers". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., 2020, pp. 15265–15276.

[4] Saba Ahmadi et al. "The Strategic Perceptron". In: *Proceedings of the 22nd ACM Conference on Economics and Computation*. Budapest Hungary: ACM, July 2021, pp. 6–25.

[5] Tosca Lechner and Ruth Urner. "Learning losses for strategic classification". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 36. 7. 2022, pp. 7337–7344.

References II

[6] Saba Ahmadi, Avrim Blum, and Kunhe Yang. "Fundamental Bounds on Online Strategic Classification". In: *Proceedings of the 24th ACM Conference on Economics and Computation*. London United Kingdom: ACM, July 2023, pp. 22–58.

[7] Nam Ho-Nguyen and Fatma Kılınç-Karzan. "Exploiting problem structure in optimization under uncertainty via online convex optimization". In: *Mathematical Programming* 177.1-2 (2018), pp. 113–147.

[8] Ganesh Ghalme et al. "Strategic Classification in the Dark". en. In: *Proceedings of the 38th International Conference on Machine Learning*. ISSN: 2640-3498. PMLR, July 2021, pp. 3672–3681.