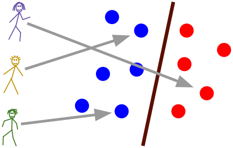
Strategic Classification from Revealed Preferences



Jinshuo Dong*
Aaron Roth*
Zachary Schutzman*
Bo Waggoner*'
Z. Steven Wu*'

**University of Pennsylvania*
'Microsoft Research

EC, June 2018

Strategic Classification from Revealed Preferences

Strategic Classification from Revealed Preferences

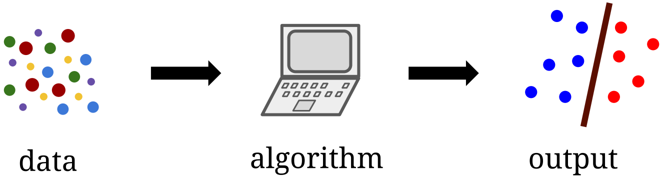
OR

Strategic Classification from Revealed Preferences

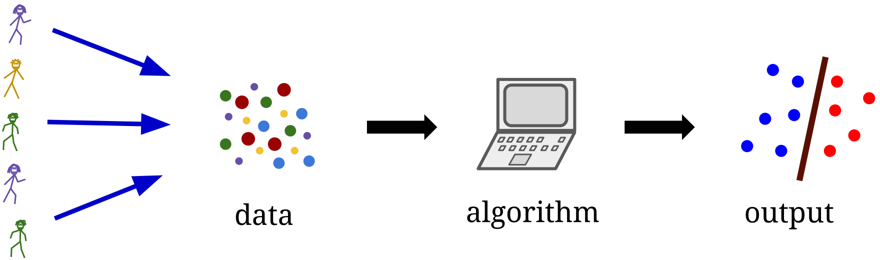
OR

When Data Goes Rogue

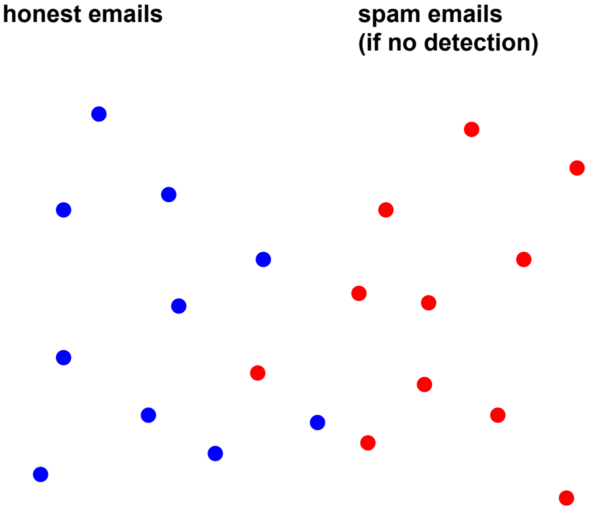
classification



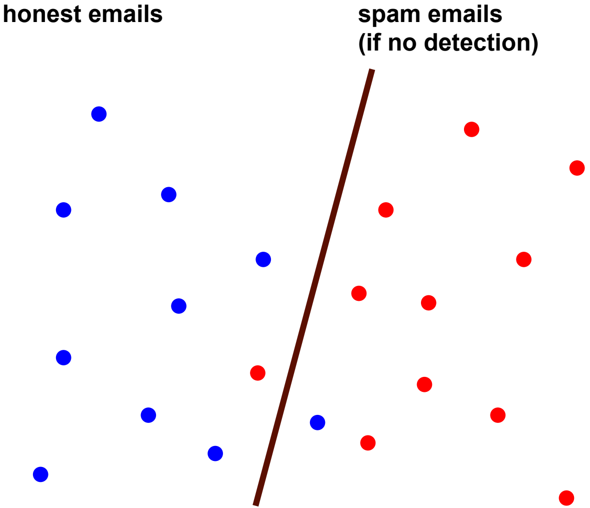
Strategic classification



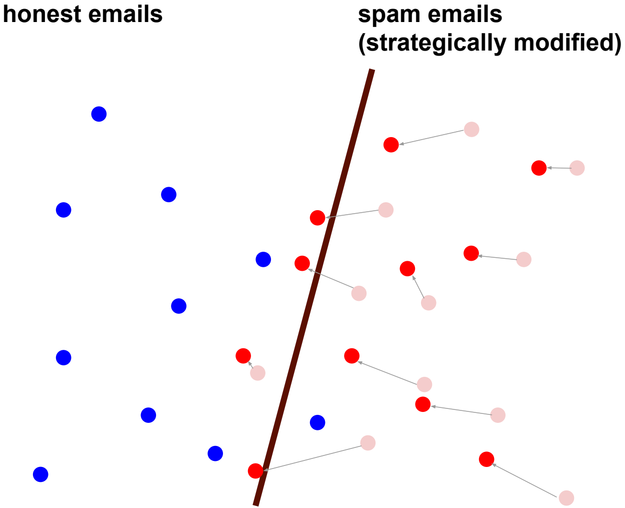
Strategic classification: pictorally



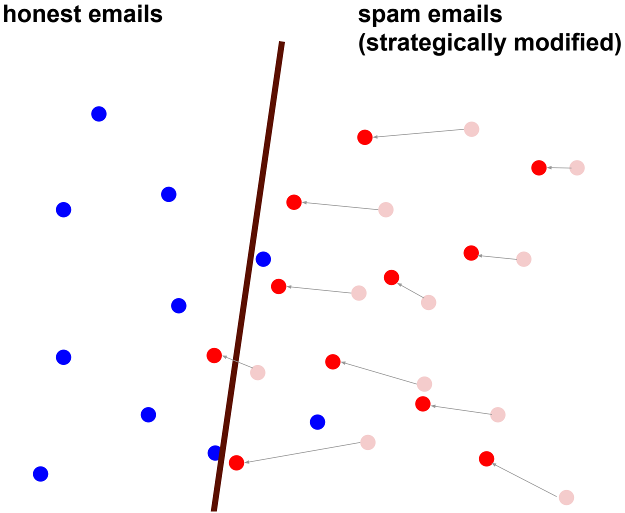
Strategic classification: pictorally



Strategic classification: pictorally



Strategic classification: pictorally



Online model of strategic classification

Over rounds $t = 1, \ldots, T$:

Online model of strategic classification

Over rounds $t = 1, \ldots, T$:

- An email arrives.

  $\hat{x}^t =$ *feature vector: what the system knows*

Online model of strategic classification

Over rounds $t = 1, \ldots, T$:

- An email arrives.
  $\hat{x}^t =$ *feature vector: what the system knows*
- The system classifies it as honest $(+1)$ or spam (-1)
  $\beta^t =$ *current classifier*

Online model of strategic classification

Over rounds $t = 1, \dots, T$:

- An email arrives.
  $\hat{x}^t$ = *feature vector: what the system knows*
- The system classifies it as honest $(+1)$ or spam (-1)
  β^t = *current classifier*
- The system observes *(eventually)* the true label $y^t \in \{\pm 1\}$.

Online model of strategic classification

Over rounds $t = 1, \ldots, T$:

- An email arrives.
  $\hat{x}^t$ = *feature vector: what the system knows*
- The system classifies it as honest $(+1)$ or spam (-1)
  β^t = *current classifier*
- The system observes *(eventually)* the true label $y^t \in \{\pm 1\}$.

Challenge: Spammers respond to the classifier!

Spam content $\hat{x}^t$ is strategically chosen depending on β^t.

Strategic classification: prior work

Prior work:[1]

- Given dataset $\sim \mathcal{D}$ and spammer preferences, learn hypothesis β

[1] *Brückner, Scheffer 2011; Hardt, Megiddo, Papadimitriou, Wooters 2016.*

Strategic classification: prior work

Prior work:[1]

- Given dataset $\sim \mathcal{D}$ and spammer preferences, learn hypothesis β
- β should classify well on $\mathcal{D}$ in *Stackelberg equilibrium*
  spammers best-respond to β

[1] *Brückner, Scheffer 2011; Hardt, Megiddo, Papadimitriou, Wooters 2016.*

Strategic classification: prior work

Prior work:[1]

- Given dataset $\sim \mathcal{D}$ and spammer preferences, learn hypothesis β
- β should classify well on $\mathcal{D}$ in *Stackelberg equilibrium*
  spammers best-respond to β

This work (key goals):

- Agents arrive online; performance measured by **regret**
- Agents are **heterogeneous**
- System **never sees spammer preferences**!
  Must infer these from behavior.

[1] *Brückner, Scheffer 2011; Hardt, Megiddo, Papadimitriou, Wooters 2016.*

This work

Question

How should one **model** strategic classification with online arrivals and limited feedback?

What is the proper **benchmark** for this problem?

How do we design **algorithms** that perform well?

Model (1/2)

Each arrival t is defined by:

Model (1/2)

Each arrival t is defined by:

- $x^t =$ its "true" features

Model (1/2)

Each arrival t is defined by:

- x^t = its "true" features
- $y^t = 1$ if it is honest, -1 if spam

Model (1/2)

Each arrival t is defined by:

- x^t = its "true" features
- $y^t = 1$ if it is honest, -1 if spam
- u^t = utility function $u^t(\beta, x, \hat{x})$
  utility for modifying x to $\hat{x}$ when classifier is β

Model (1/2)

Each arrival t is defined by:

- x^t = its "true" features
- $y^t = 1$ if it is honest, -1 if spam
- u^t = utility function $u^t(\beta, x, \hat{x})$
  utility for modifying x to $\hat{x}$ when classifier is β

If $y^t = 1$ (honest): always set $\hat{x}^t = x^t$
Send desired email, nonstrategically.

If $y^t = -1$ (spam): choose $\hat{x}^t$ to maximize utility!
Strategically modify email in response to β^t.

Model (2/2)

Over rounds $t = 1, \ldots, T$:

- Classifier β^t is deployed

Model (2/2)

Over rounds $t = 1, \ldots, T$:

- Classifier β^t is deployed
- Data point $(\hat{x}^t, y^t)$ is observed

Model (2/2)

Over rounds $t = 1, \ldots, T$:

- Classifier β^t is deployed
- Data point $(\hat{x}^t, y^t)$ is observed
- System receives loss $\ell(\beta^t, \hat{x}^t, y^t)$
  Measures performance of classifier on observation

Model (2/2)

Over rounds $t = 1, \ldots, T$:

- Classifier β^t is deployed
- Data point $(\hat{x}^t, y^t)$ is observed
- System receives loss $\ell(\beta^t, \hat{x}^t, y^t)$
  Measures performance of classifier on observation
- System updates to β^{t+1}

Performance and benchmark

Best-response function:

If honest: $\hat{x}^t(\beta) = x^t$

If spam: $\hat{x}^t(\beta) = \arg\max_{\hat{x}} u^t(\beta, x^t, \hat{x})$.

Performance and benchmark

Best-response function:

If honest: $\hat{x}^t(\beta) = x^t$

If spam: $\hat{x}^t(\beta) = \arg\max_{\hat{x}} u^t(\beta, x^t, \hat{x})$.

$$\underbrace{\frac{1}{T}\sum_{t=1}^{T} \ell(\beta^t, \hat{x}^t(\beta^t), y^t)}_{\text{Performance}}$$

Performance and benchmark

Best-response function:
If honest: $\hat{x}^t(\beta) = x^t$
If spam: $\hat{x}^t(\beta) = \arg\max_{\hat{x}} u^t(\beta, x^t, \hat{x})$.

Compare to **best fixed classifier β^* in hindsight**.

Key point: If we had used a different classifier,
spammers **would have responded differently!**

Performance and benchmark

Best-response function:

If honest: $\hat{x}^t(\beta) = x^t$

If spam: $\hat{x}^t(\beta) = \arg\max_{\hat{x}} u^t(\beta, x^t, \hat{x})$.

$$\underbrace{\frac{1}{T}\sum_{t=1}^{T} \ell(\beta^t, \hat{x}^t(\beta^t), y^t)}_{\textbf{Performance}}$$

Performance and benchmark

Best-response function:

If honest: $\hat{x}^t(\beta) = x^t$

If spam: $\hat{x}^t(\beta) = \arg\max_{\hat{x}} u^t(\beta, x^t, \hat{x})$.

$$\underbrace{\frac{1}{T}\sum_{t=1}^{T} \ell(\beta^t, \hat{x}^t(\beta^t), y^t)}_{\textbf{Performance}} \qquad \underbrace{\frac{1}{T}\sum_{t=1}^{T} \ell(\beta^*, \hat{x}^t(\beta^*), y^t)}_{\textbf{OPT}}$$

Performance and benchmark

Best-response function:

If honest: $\hat{x}^t(\beta) = x^t$

If spam: $\hat{x}^t(\beta) = \arg\max_{\hat{x}} u^t(\beta, x^t, \hat{x})$.

$$\textbf{Avg Regret} = \underbrace{\frac{1}{T}\sum_{t=1}^{T} \ell(\beta^t, \hat{x}^t(\beta^t), y^t)}_{\textbf{Performance}} - \underbrace{\frac{1}{T}\sum_{t=1}^{T} \ell(\beta^*, \hat{x}^t(\beta^*), y^t)}_{\textbf{OPT}}.$$

Performance and benchmark

Best-response function:

If honest: $\hat{x}^t(\beta) = x^t$

If spam: $\hat{x}^t(\beta) = \arg\max_{\hat{x}} u^t(\beta, x^t, \hat{x})$.

$$\textbf{Avg Regret} \;=\; \underbrace{\frac{1}{T}\sum_{t=1}^{T} \ell(\beta^t, \hat{x}^t(\beta^t), y^t)}_{\textbf{Performance}} \;-\; \underbrace{\frac{1}{T}\sum_{t=1}^{T} \ell(\beta^*, \hat{x}^t(\beta^*), y^t)}_{\textbf{OPT}}.$$

Notice: Algorithm cannot know or compute OPT!

Performance and benchmark

Best-response function:

If honest: $\hat{x}^t(\beta) = x^t$

If spam: $\hat{x}^t(\beta) = \arg\max_{\hat{x}} u^t(\beta, x^t, \hat{x})$.

$$\textbf{Avg Regret} = \underbrace{\frac{1}{T}\sum_{t=1}^{T} \ell(\beta^t, \hat{x}^t(\beta^t), y^t)}_{\textbf{Performance}} - \underbrace{\frac{1}{T}\sum_{t=1}^{T} \ell(\beta^*, \hat{x}^t(\beta^*), y^t)}_{\textbf{OPT}}.$$

Notice: Algorithm cannot know or compute OPT!

Nevertheless: We will compete with it (under assumptions).

Assumptions we make

To solve the problem, we assume:

- Linear prediction $\beta^t \cdot \hat{x}^t \in (-\infty, \infty)$ *larger $\leftrightarrow$ more honest*

Assumptions we make

To solve the problem, we assume:

- Linear prediction $\beta^t \cdot \hat{x}^t \in (-\infty, \infty)$ *larger $\leftrightarrow$ more honest*
- Loss $\ell(\beta, \hat{x}, y) = \log\left(1 + e^{-y\beta\cdot\hat{x}}\right)$ *results also hold for hinge*

Assumptions we make

To solve the problem, we assume:

- Linear prediction $\beta^t \cdot \hat{x}^t \in (-\infty, \infty)$ *larger $\leftrightarrow$ more honest*
- Loss $\ell(\beta, \hat{x}, y) = \log\left(1 + e^{-y\beta\cdot\hat{x}}\right)$ *results also hold for hinge*
- Spammer utility is of the form

$$u^t = \underbrace{\beta^t \cdot \hat{x}^t}_{\textbf{prediction}} - \underbrace{d^t(x^t, \hat{x}^t)}_{\textbf{cost}}$$

for a class of d^t = distance between truth and manipulation

Assumptions we make

To solve the problem, we assume:

- Linear prediction $\beta^t \cdot \hat{x}^t \in (-\infty, \infty)$ *larger $\leftrightarrow$ more honest*
- Loss $\ell(\beta, \hat{x}, y) = \log\left(1 + e^{-y\beta\cdot\hat{x}}\right)$ *results also hold for hinge*
- Spammer utility is of the form

$$u^t = \underbrace{\beta^t \cdot \hat{x}^t}_{\textbf{prediction}} - \underbrace{d^t(x^t, \hat{x}^t)}_{\textbf{cost}}$$

for a class of d^t = distance between truth and manipulation

Example: $d^t(x, \hat{x}) = \|Ax - A\hat{x}\|_p^r$ for $r > 1$ and A invertible.

Results

Main result: reduction to **online convex optimization**.

Results

Main result: reduction to **online convex optimization**.

Theorem

Let $\ell^t(\beta) = \ell(\beta, \hat{x}^t(\beta), y^t)$.
Then under our assumptions, ℓ^t *is* **convex***!*

Results

Main result: reduction to **online convex optimization**.

Theorem

Let $\ell^t(\beta) = \ell(\beta, \hat{x}^t(\beta), y^t)$.
Then under our assumptions, ℓ^t *is* **convex***!*

Main tool: convex analysis.

- $u^t = \hat{x} \cdot \beta - d^t(x^t, \hat{x})$.
- Best-response $\hat{x}^t(\beta)$ given by convex conjugate of d^t.
- d^t homogeneous of degree $k \implies \hat{x}^t(\beta) \cdot \beta$ is convex.
- $\implies \beta \mapsto \log\left(1 + e^{-y^t \hat{x}^t(\beta)\cdot\beta}\right)$ is convex.

Results

Main result: reduction to **online convex optimization**.

Theorem

Let $\ell^t(\beta) = \ell(\beta, \hat{x}^t(\beta), y^t)$.
Then under our assumptions, ℓ^t *is* **convex***!*

Corollary

By appropriate application of online convex optimization algorithms, we can achieve average Stackelberg regret $O\left(\frac{1}{\sqrt{T}}\right)$.

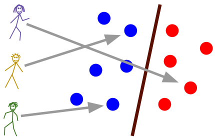
$T =$ number of arrivals

Despite not knowing the details of ℓ^t.

Extensions, future work

Extensions:

- Algorithm treats honest and spam updates differently
  can get full gradient feedback for honest data points
- Somewhat more general agent utilities; hinge loss



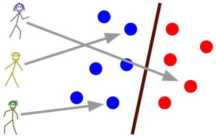
Extensions, future work

Extensions:

- Algorithm treats honest and spam updates differently
  can get full gradient feedback for honest data points
- Somewhat more general agent utilities; hinge loss

Future work:

- Other loss functions
- Other forms of agent utility
- Outside the convex optimization paradigm?



Extensions, future work

Extensions:

- Algorithm treats honest and spam updates differently
  can get full gradient feedback for honest data points
- Somewhat more general agent utilities; hinge loss

Future work:

- Other loss functions
- Other forms of agent utility
- Outside the convex optimization paradigm?

Thanks!