Rethinking Robustness: From Classification to Contextual Bandits

Ankur Moitra (MIT)

UAI 2021 Keynote

Based on work with Sitan Chen, Frederic Koehler, Morris Yau

In this talk, we will explore models for corruption that **blend worst-case and average-case analysis**, in hopes of designing **more robust algorithms** for classic problems in learning

OUTLINE

Part I: Supervised Learning

- PAC Learning and Robustness
- Our Results and Framework
- Applications to Fairness

Part II: Online Learning

- Regression and Clean MSE
- Dynamic Range vs. Variance
- Our Results and Extensions to Contextual Bandits

OUTLINE

Part I: Supervised Learning

- **PAC Learning and Robustness**
- Our Results and Framework
- Applications to Fairness

Part II: Online Learning

- Regression and Clean MSE
- Dynamic Range vs. Variance
- Our Results and Extensions to Contextual Bandits

In 1984, Valiant introduced the **PAC Learning Model**:

(1) Given samples (X, Y) where the distribution on X is arbitrary and Y is a label that is +1 or -1

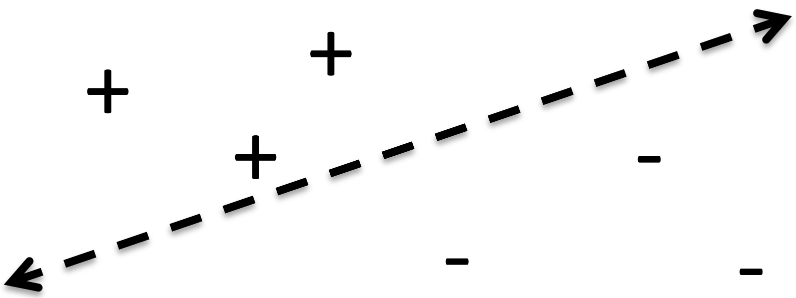
(2) Assume Y = h(X) for some unknown hypothesis h that is in a known class H

In 1984, Valiant introduced the **PAC Learning Model**:

(1) Given samples (X, Y) where the distribution on X is arbitrary and Y is a label that is +1 or -1

(2) Assume Y = h(X) for some unknown hypothesis h that is in a known class H

e.g. the class of **halfspaces** $Y = \mathrm{sgn}(\langle w^*, X \rangle + b)$

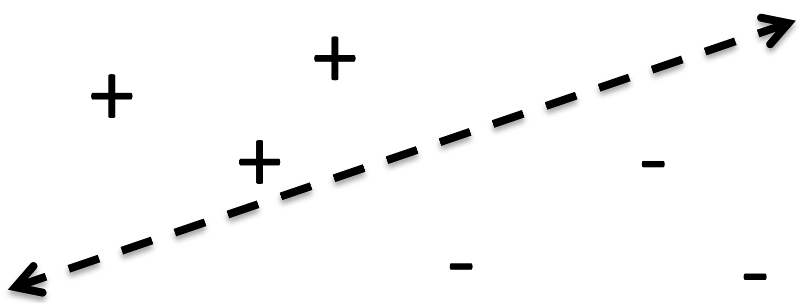


In 1984, Valiant introduced the **PAC Learning Model**:

(1) Given samples (X, Y) where the distribution on X is arbitrary and Y is a label that is +1 or -1

(2) Assume Y = h(X) for some unknown hypothesis h that is in a known class H

e.g. the class of **halfspaces** $Y = \mathrm{sgn}(\langle w^*, X\rangle + b)$



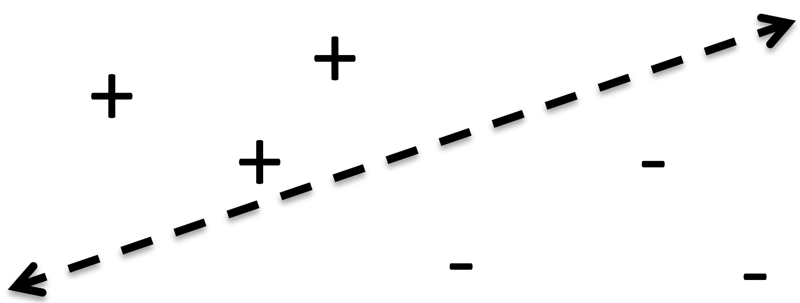
Goal: Estimate h approximately

In 1984, Valiant introduced the **PAC Learning Model**:

(1) Given samples (X, Y) where the distribution on X is arbitrary and Y is a label that is +1 or -1

(2) Assume Y = h(X) for some unknown hypothesis h that is in a known class H

e.g. the class of **halfspaces** $Y = \mathrm{sgn}(\langle w^*, X \rangle + b)$



Goal: Estimate h approximately

Probably **A**pproximately **C**orrect

MODELS FOR NOISE

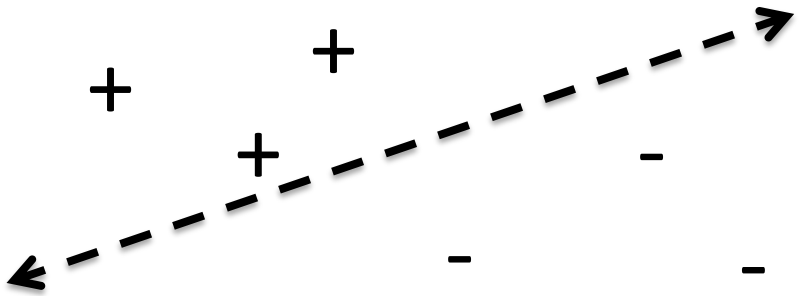
What if there is no simple hypothesis that fits the data *exactly*?

MODELS FOR NOISE

What if there is no simple hypothesis that fits the data *exactly*?

Standard frameworks:

Random Classification Noise: Each label is flipped with some fixed probability

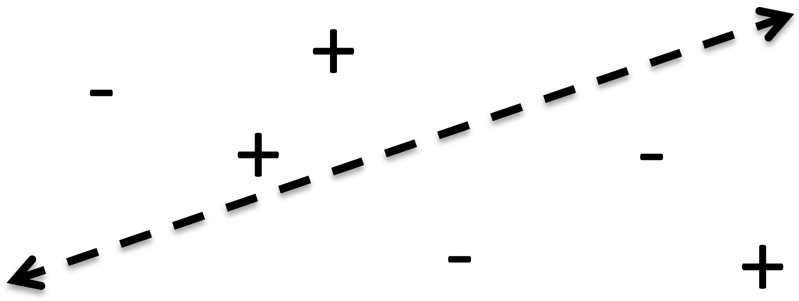


MODELS FOR NOISE

What if there is no simple hypothesis that fits the data *exactly*?

Standard frameworks:

Random Classification Noise: Each label is flipped with some fixed probability

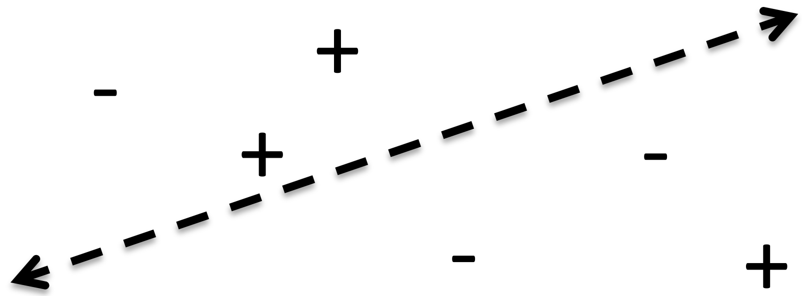


MODELS FOR NOISE

What if there is no simple hypothesis that fits the data *exactly*?

Standard frameworks:

Random Classification Noise: Each label is flipped with some fixed probability



[Blum et al.]: Efficient algorithm for halfspaces under RCN

MODELS FOR NOISE

What if there is no simple hypothesis that fits the data *exactly*?

Standard frameworks:

MODELS FOR NOISE

What if there is no simple hypothesis that fits the data *exactly*?

Standard frameworks:

Agnostic Noise: No assumption about the structure of the noise, still want to find approximately best agreement in the class

MODELS FOR NOISE

What if there is no simple hypothesis that fits the data *exactly*?

Standard frameworks:

Agnostic Noise: No assumption about the structure of the noise, still want to find approximately best agreement in the class

Unfortunately, agnostic learning is generally hard without further assumptions!

MODELS FOR NOISE

What if there is no simple hypothesis that fits the data *exactly*?

Standard frameworks:

Agnostic Noise: No assumption about the structure of the noise, still want to find approximately best agreement in the class

Unfortunately, agnostic learning is generally hard without further assumptions!

[Kalai et al.], [Awasthi, Balcan, Long], [Daniely]

Are there challenging noise models where we can learn without making distributional assumptions on X?

Are there challenging noise models where we can learn without making distributional assumptions on X?

In this talk, we'll be interested in:

Massart Noise: The label of each point x is flipped independently with some probability $\eta(x) \leq \eta < 1/2$

Are there challenging noise models where we can learn without making distributional assumptions on X?

In this talk, we'll be interested in:

Massart Noise: The label of each point x is flipped independently with some probability $\eta(x) \leq \eta < 1/2$

Interpretation #1: Each label is flipped independently with prob. η but an adversary can choose to **unflip** it

Are there challenging noise models where we can learn without making distributional assumptions on X?

In this talk, we'll be interested in:

Massart Noise: The label of each point x is flipped independently with some probability $\eta(x) \leq \eta < 1/2$

Interpretation #1: Each label is flipped independently with prob. η but an adversary can choose to **unflip** it

Interpretation #2 (sort of): An adversary can arbitrarily control a **random** η fraction of the data

Are there challenging noise models where we can learn without making distributional assumptions on X?

In this talk, we'll be interested in:

Massart Noise: The label of each point x is flipped independently with some probability $\eta(x) \leq \eta < 1/2$

Interpretation #1: Each label is flipped independently with prob. η but an adversary can choose to **unflip** it

Interpretation #2 (sort of): An adversary can arbitrarily control a **random** η fraction of the data

Are there distribution-independent algorithms for learning with Massart noise?

PRIOR WORK

Theorem [Diakonikolas, Goulekakis, Tzamos '19]: There is a polynomial time algorithm for **improperly** learning halfspaces under Massart noise with error $\eta + \epsilon$

PRIOR WORK

Theorem [Diakonikolas, Goulekakis, Tzamos ‘19]: There is a polynomial time algorithm for **improperly** learning halfspaces under Massart noise with error $\eta + \epsilon$

The algorithm outputs a partition of space into a polynomial number of regions, with a different halfspace on each

PRIOR WORK

Theorem [Diakonikolas, Goulekakis, Tzamos '19]: There is a polynomial time algorithm for **improperly** learning halfspaces under Massart noise with error $\eta + \epsilon$

The algorithm outputs a partition of space into a polynomial number of regions, with a different halfspace on each

Is there a proper learning algorithm?

OUTLINE

Part I: Supervised Learning

- PAC Learning and Robustness
- Our Results and Framework
- Applications to Fairness

Part II: Online Learning

- Regression and Clean MSE
- Dynamic Range vs. Variance
- Our Results and Extensions to Contextual Bandits

OUTLINE

Part I: Supervised Learning

- PAC Learning and Robustness
- **Our Results and Framework**
- Applications to Fairness

Part II: Online Learning

- Regression and Clean MSE
- Dynamic Range vs. Variance
- Our Results and Extensions to Contextual Bandits

OUR RESULTS

Theorem: There is a polynomial time algorithm for **properly** learning halfspaces under Massart noise with error $\eta + \epsilon$

OUR RESULTS

Theorem: There is a polynomial time algorithm for **properly** learning halfspaces under Massart noise with error $\eta + \epsilon$

We give a general framework based on zero-sum games

OUR RESULTS

Theorem: There is a polynomial time algorithm for **properly** learning halfspaces under Massart noise with error $\eta + \epsilon$

We give a general framework based on zero-sum games

Theorem: There is a polynomial time algorithm for learning **generalized linear models** under Massart noise

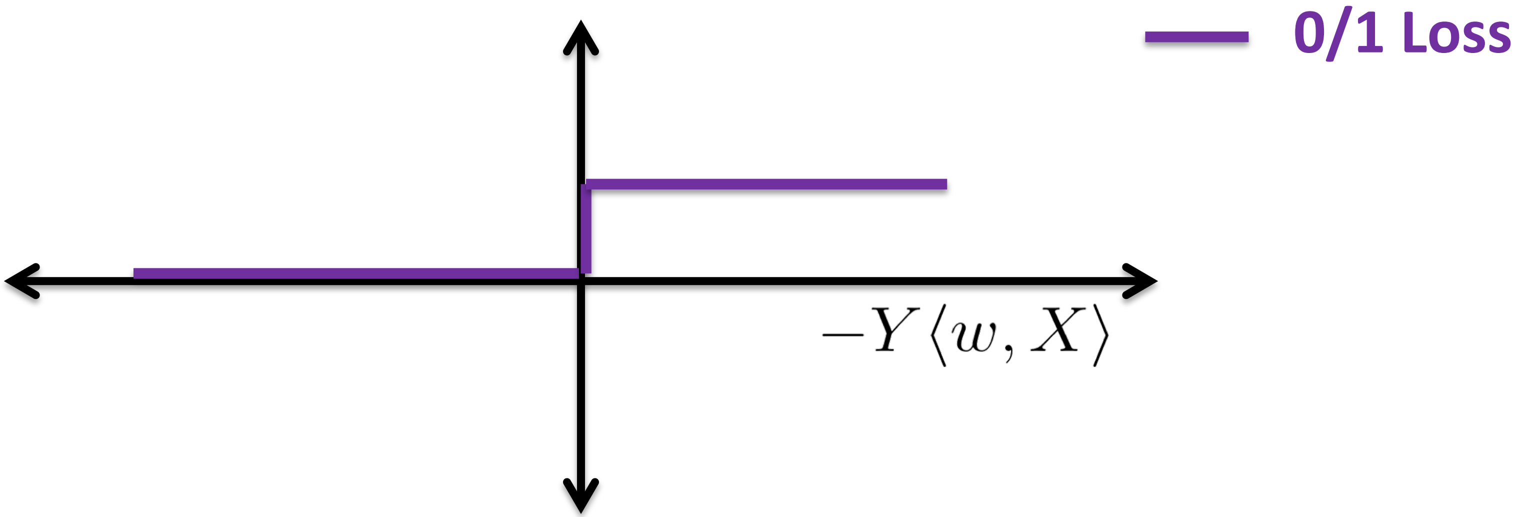
i.e $\mathbb{E}[Y|X] = \sigma(\langle w^*, X \rangle + b)$

link function: monotone, Lipschitz

In particular, this includes noisy logistic regression as a special case

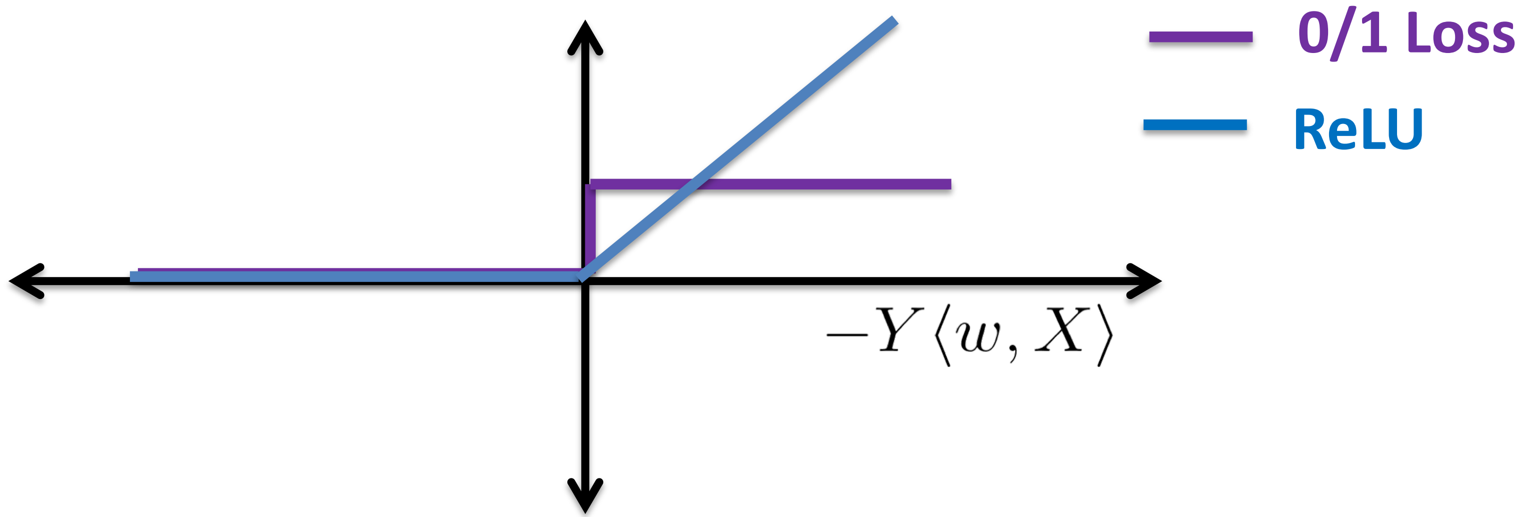
REVIEW: CONVEX SURROGATES

For RCN, a natural approach is to use the **Leaky ReLU**…



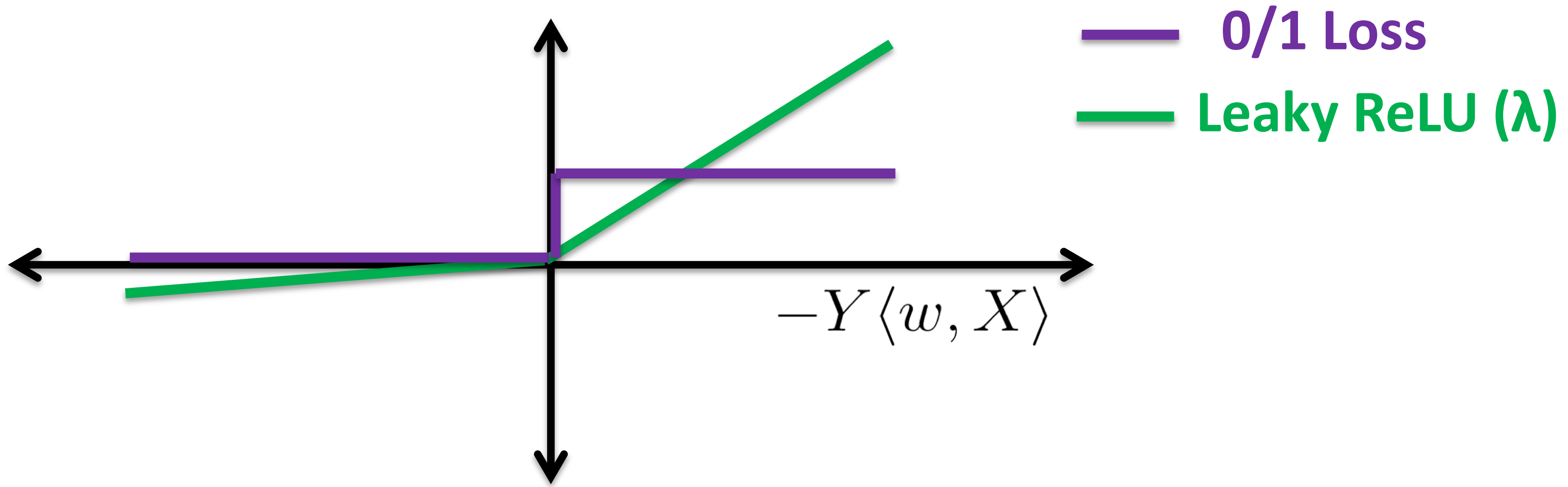
REVIEW: CONVEX SURROGATES

For RCN, a natural approach is to use the **Leaky ReLU**…



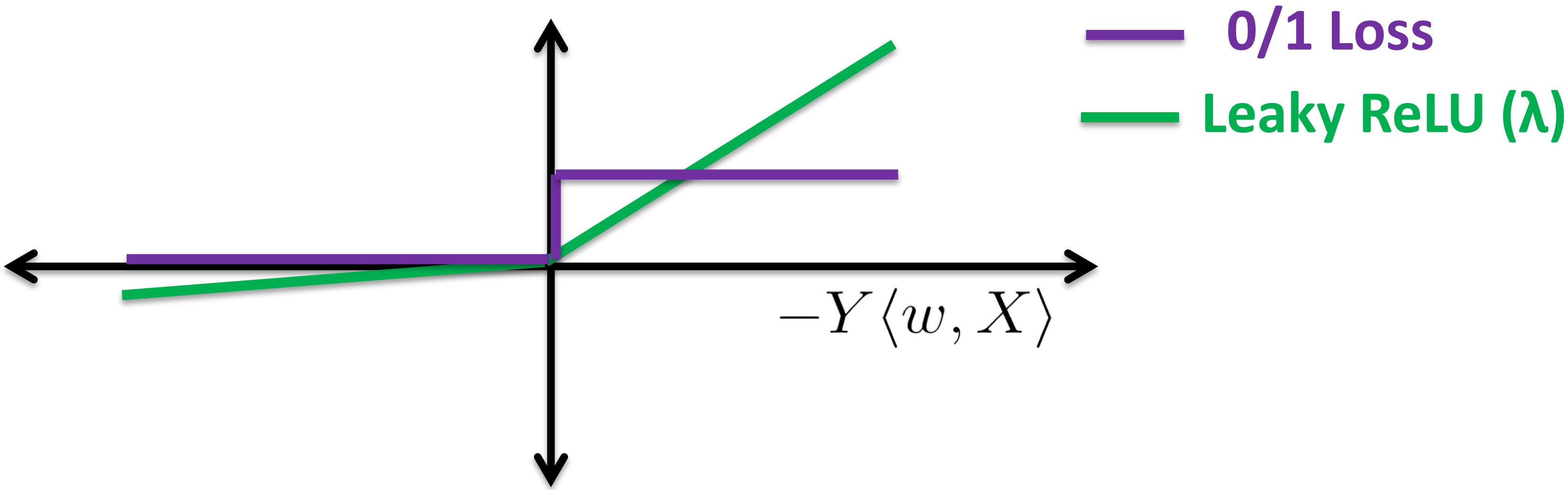
REVIEW: CONVEX SURROGATES

For RCN, a natural approach is to use the **Leaky ReLU**…



REVIEW: CONVEX SURROGATES

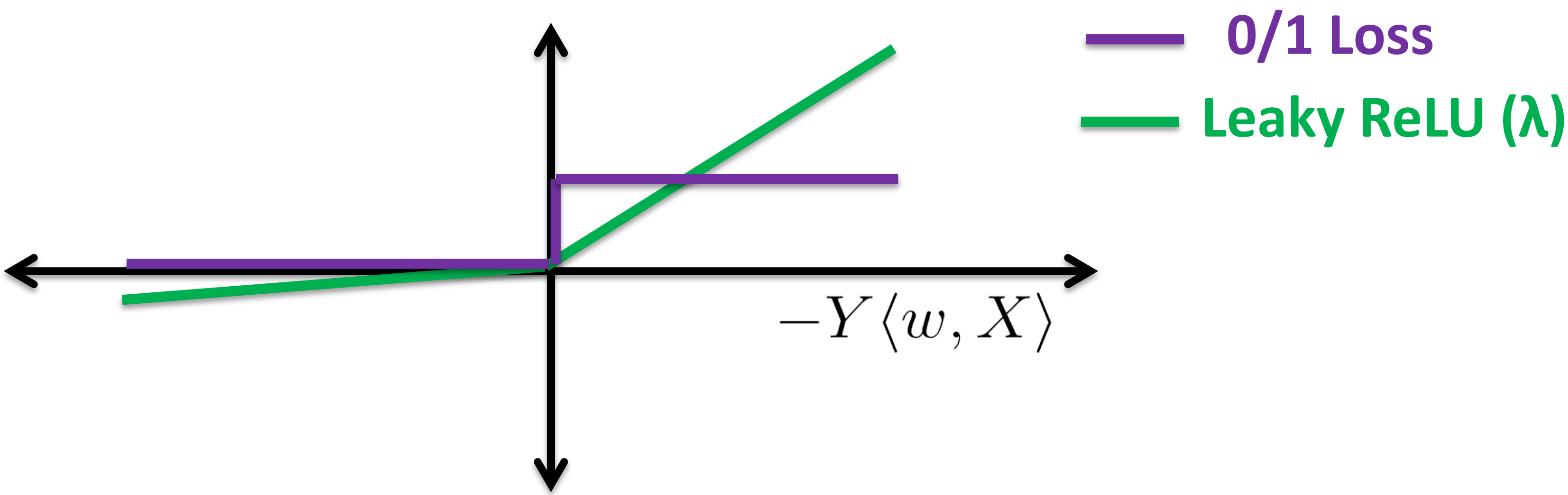
For RCN, a natural approach is to use the **Leaky ReLU**…



Proposition: The expected Leaky ReLU loss is convex and if you set λ appropriately, any minimum gets optimal error under RCN

REVIEW: CONVEX SURROGATES

For RCN, a natural approach is to use the **Leaky ReLU**…



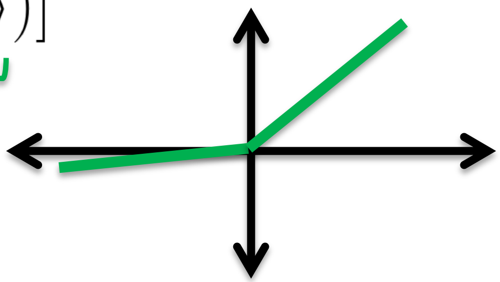
Proposition: The expected Leaky ReLU loss is convex and if you set λ appropriately, any minimum gets optimal error under RCN

How can we tolerate varying noise rates?

A GENERAL FRAMEWORK

Consider the following two-player game

$$\min_{\|w\| \leq 1} \max_{\mathbf{c}} \mathbb{E}[\mathbf{c}(\mathbf{X}) \underbrace{\ell_\lambda(-Y\langle w, X\rangle)}_{\text{Leaky ReLU}}]$$

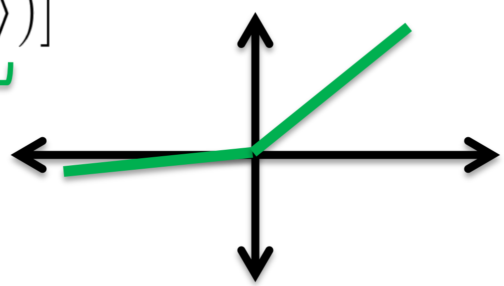


where c ranges over all reweightings

A GENERAL FRAMEWORK

Consider the following two-player game

$$\min_{\|w\| \leq 1} \max_{\mathbf{c}} \mathbb{E}[\mathbf{c}(\mathbf{X}) \underbrace{\ell_\lambda(-Y\langle w, X\rangle)}_{\text{Leaky ReLU}}]$$



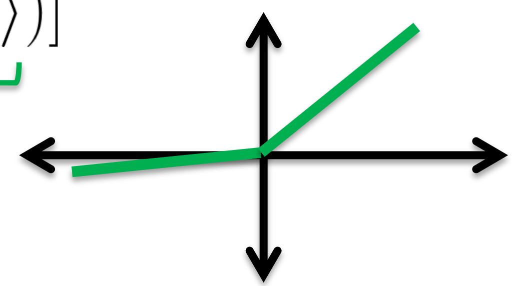
where c ranges over all reweightings

Intuition: The true hypothesis does well on any region of space, and the max-player looks for a region where the min-player is doing the worst

A GENERAL FRAMEWORK

Consider the following two-player game

$$\min_{\|w\| \leq 1} \max_{\mathbf{c}} \mathbb{E}[\mathbf{c}(\mathbf{X})\underbrace{\ell_\lambda(-Y\langle w, X\rangle)}_{\text{Leaky ReLU}}]$$



where c ranges over all reweightings

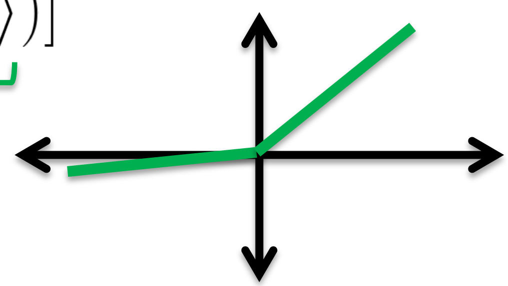
Intuition: The true hypothesis does well on any region of space, and the max-player looks for a region where the min-player is doing the worst

Claim: The optimal solution for the min-player is w^*

A GENERAL FRAMEWORK

Consider the following two-player game

$$\min_{\|w\| \leq 1} \max_{\mathbf{c}} \mathbb{E}[\mathbf{c}(\mathbf{X}) \underbrace{\ell_\lambda(-Y\langle w, X\rangle)}_{\text{Leaky ReLU}}]$$



where c ranges over all reweightings

Intuition: The true hypothesis does well on any region of space, and the max-player looks for a region where the min-player is doing the worst

Claim: The optimal solution for the min-player is w^*

Unfortunately, optimizing over the max-players strategies is both statistically and computationally hard

A GENERAL FRAMEWORK, CONTINUED

Instead we work with a relaxation where the max-player can only restrict the distribution to **slabs along the current w**

$$\min_{\|w\| \leq 1} \max_{r > 0} \mathbb{E}[\ell_\lambda(-Y\langle w, X\rangle) | -r \leq \langle w, X\rangle \leq r]$$

A GENERAL FRAMEWORK, CONTINUED

Instead we work with a relaxation where the max-player can only restrict the distribution to **slabs along the current w**

$$\min_{\|w\| \leq 1} \max_{r > 0} \mathbb{E}[\ell_\lambda(-Y\langle w, X\rangle) | -r \leq \langle w, X\rangle \leq r]$$

We show that any approximate equilibrium in this game necessarily corresponds to a hypothesis with low error

THE ALGORITHM

How do we find an approximate equilibrium?

- **Initialize** w to a vector in the unit ball
- **Repeat**
  - **Max-Player** finds the slab $S(w, r^*)$ that maximizes the loss $L_\lambda^{S(w,r^*)}$. If the loss is $\leq \epsilon$ then **return** w
  - **Min-Player** takes a step in the direction $-g$ where

$$g = \nabla L_\lambda^{S(w,r^*)}$$

and projects back into the unit ball

THE ALGORITHM

How do we find an approximate equilibrium?

- **Initialize** w to a vector in the unit ball
- **Repeat**
  - **Max-Player** finds the slab $S(w, r^*)$ that maximizes the loss $L_\lambda^{S(w,r^*)}$. If the loss is $\leq \epsilon$ then **return** w
  - **Min-Player** takes a step in the direction $-g$ where

$$g = \nabla L_\lambda^{S(w,r^*)}$$

    and projects back into the unit ball

Main Theorem: Converges in a polynomial number of iterations and provably solves the Massart learning problem

OUTLINE

Part I: Supervised Learning

- PAC Learning and Robustness
- Our Results and Framework
- Applications to Fairness

Part II: Online Learning

- Regression and Clean MSE
- Dynamic Range vs. Variance
- Our Results and Extensions to Contextual Bandits

OUTLINE

Part I: Supervised Learning

- PAC Learning and Robustness
- Our Results and Framework
- **Applications to Fairness**

Part II: Online Learning

- Regression and Clean MSE
- Dynamic Range vs. Variance
- Our Results and Extensions to Contextual Bandits

EXPERIMENTS

When is this noise model useful?

EXPERIMENTS

When is this noise model useful?

UCI Adults Dataset: 48.8k individuals, 14 attributes, goal is to predict whether income is above or below $50k

EXPERIMENTS

When is this noise model useful?

UCI Adults Dataset: 48.8k individuals, 14 attributes, goal is to predict whether income is above or below $50k

Motivation: Numerous empirical studies about how the level of noise various across demographic groups e.g. in surveys

EXPERIMENTS

When is this noise model useful?

UCI Adults Dataset: 48.8k individuals, 14 attributes, goal is to predict whether income is above or below $50k

Motivation: Numerous empirical studies about how the level of noise various across demographic groups e.g. in surveys

We added noise *outside* a target group, and ran off-the-shelf algorithms whose goal is to maximize overall accuracy

EXPERIMENTS

When is this noise model useful?

UCI Adults Dataset: 48.8k individuals, 14 attributes, goal is to predict whether income is above or below $50k

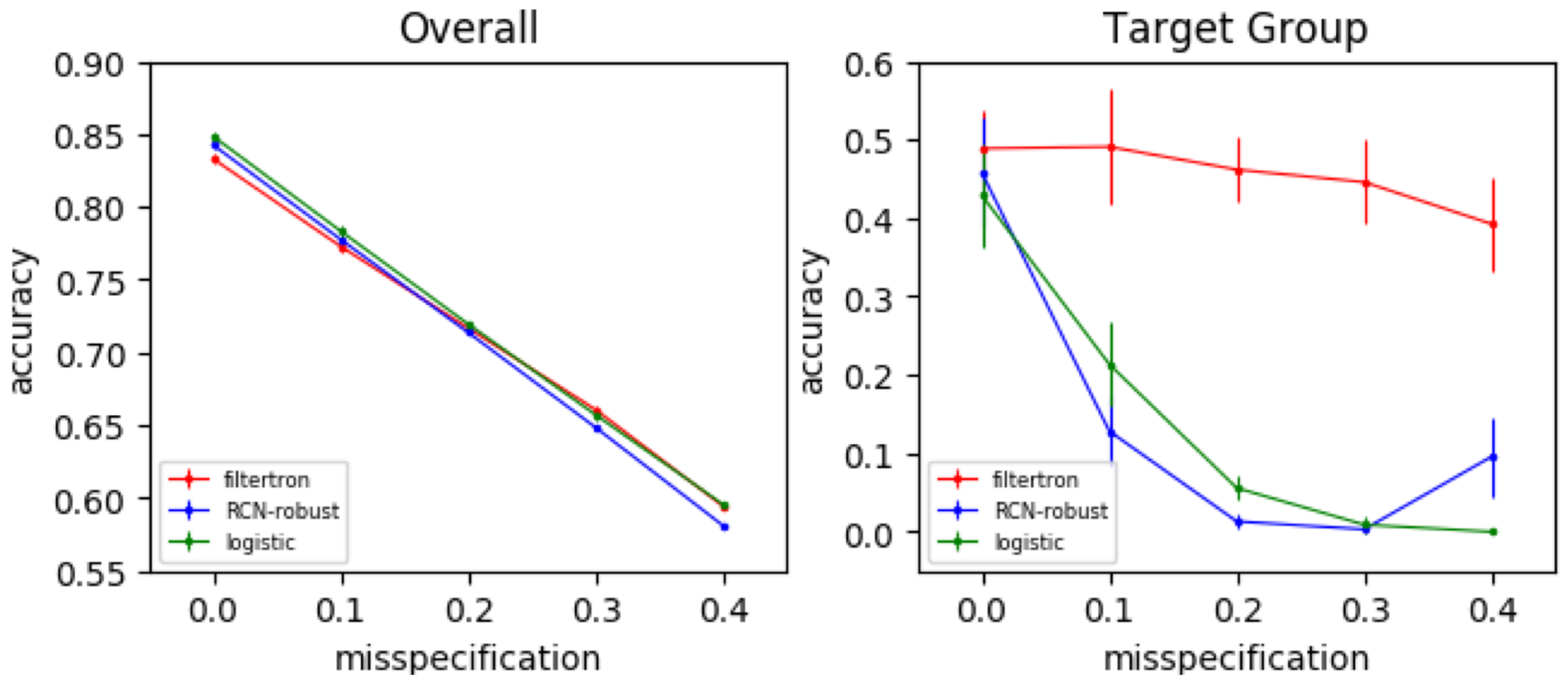
Motivation: Numerous empirical studies about how the level of noise various across demographic groups e.g. in surveys

We added noise *outside* a target group, and ran off-the-shelf algorithms whose goal is to maximize overall accuracy

We measure overall accuracy and accuracy on the part of the target group that is above $50k

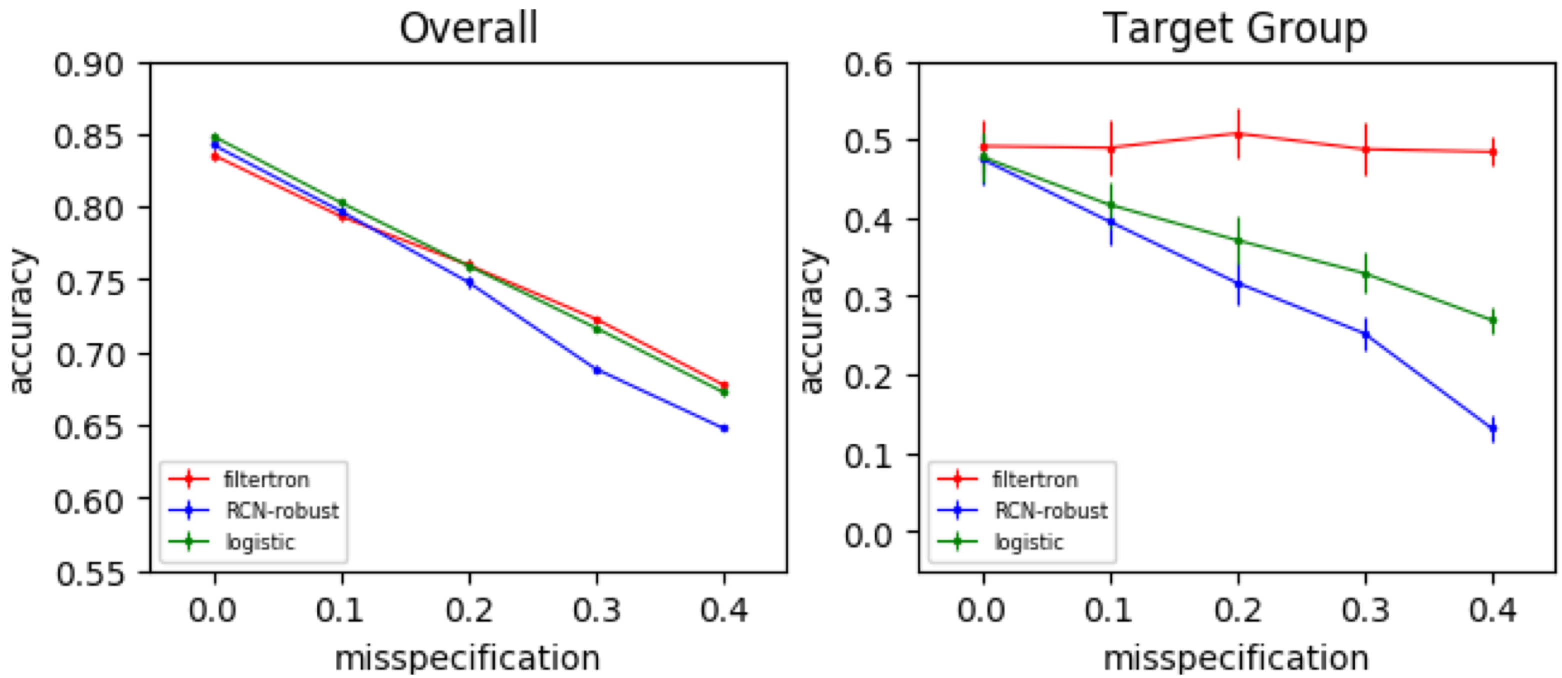
EXPERIMENTS

Target group: African Americans



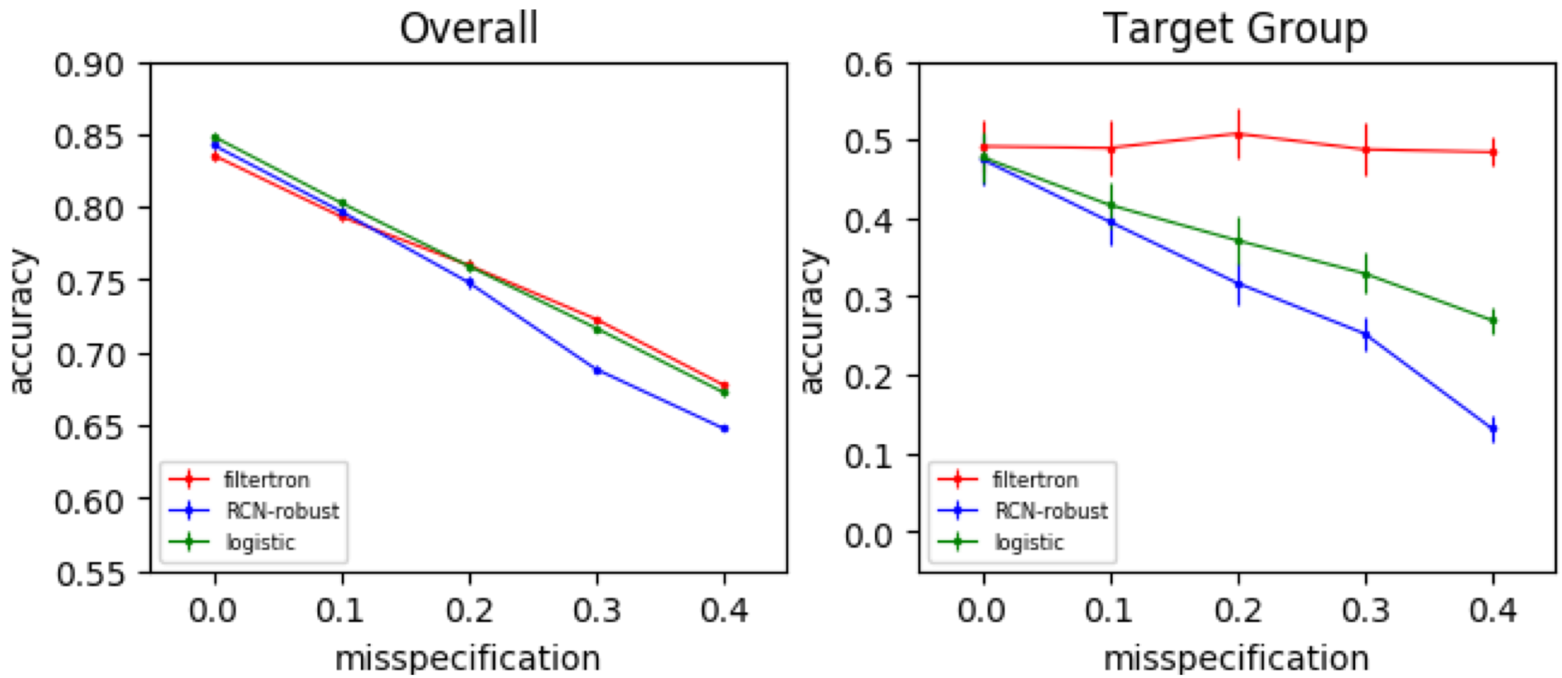
EXPERIMENTS

Target group: Female



EXPERIMENTS

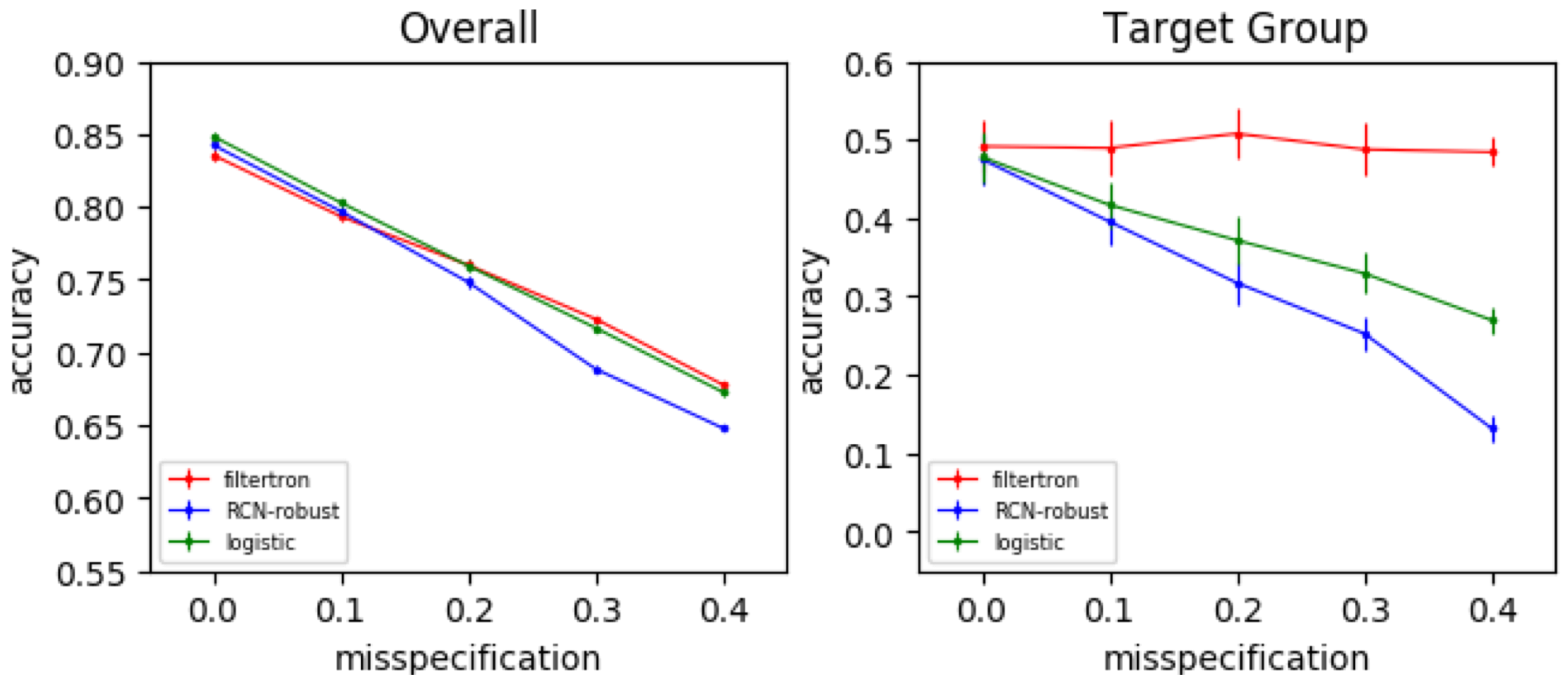
Target group: Female



Many natural algorithms (e.g. logistic) amplify bias in the data

EXPERIMENTS

Target group: Female



Many natural algorithms (e.g. logistic) amplify bias in the data

Is ours more fair *because* it can tolerate heterogenous noise?

OUTLINE

Part I: Supervised Learning

- PAC Learning and Robustness
- Our Results and Framework
- Applications to Fairness

Part II: Online Learning

- Regression and Clean MSE
- Dynamic Range vs. Variance
- Our Results and Extensions to Contextual Bandits

OUTLINE

Part I: Supervised Learning

- PAC Learning and Robustness
- Our Results and Framework
- Applications to Fairness

Part II: Online Learning

- **Regression and Clean MSE**
- Dynamic Range vs. Variance
- Our Results and Extensions to Contextual Bandits

ONLINE LINEAR REGRESSION

Goal: Solve a sequence of linear prediction problems

ONLINE LINEAR REGRESSION

Goal: Solve a sequence of linear prediction problems

In each time step, we

(1) **Observe a covariate** x_t

ONLINE LINEAR REGRESSION

Goal: Solve a sequence of linear prediction problems

In each time step, we

(1) **Observe a covariate** x_t

(2) **Predict the response** $y_t = \langle w^*, x_t \rangle + \sigma_t$

true regressor **additive noise**

and incur loss based on the squared error

ONLINE LINEAR REGRESSION

Goal: Solve a sequence of linear prediction problems

In each time step, we

(1) **Observe a covariate** x_t

(2) **Predict the response** $y_t = \langle w^*, x_t \rangle + \sigma_t$

true regressor **additive noise**

and incur loss based on the squared error

Classic Solution: Online Gradient Descent, see e.g. **[Hazan, '19]**

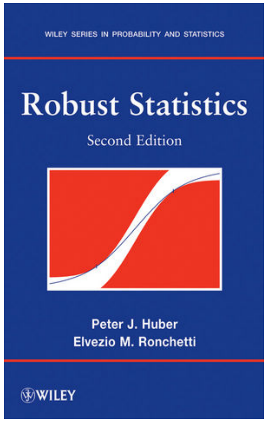
MODELS FOR NOISE

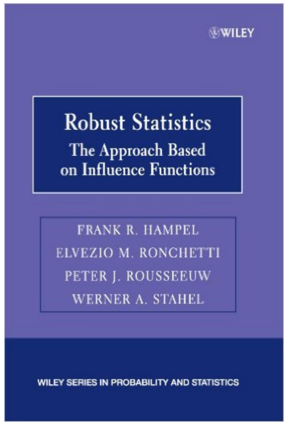
What if some of the responses are corrupted?

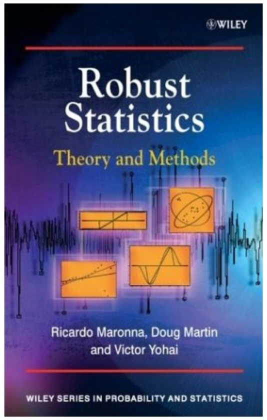
MODELS FOR NOISE

What if some of the responses are corrupted?

Definition: In the **Huber Contamination Model**, a random η fraction of the responses are arbitrarily corrupted







ROBUSTNESS GUARANTEES

Proposition [folklore]: Online gradient descent achieves

clean average mean squared error

error on uncorrupted responses

$$\frac{1}{T}\sum_{t=1}^{T}\left(y_t - \widehat{y}_t\right)^2$$

of $O(\eta R^2)$, where $\|w^*\| \leq 1$ and $\|x_t\| \leq R$

ROBUSTNESS GUARANTEES

Proposition [folklore]: Online gradient descent achieves

clean average mean squared error

error on uncorrupted responses

$$\frac{1}{T}\sum_{t=1}^{T}\left(y_t - \widehat{y}_t\right)^2$$

of $O(\eta R^2)$, where $||w^*|| \leq 1$ and $||x_t|| \leq R$

This can be quite far from optimal

ROBUSTNESS GUARANTEES

Proposition [folklore]: Online gradient descent achieves

clean average mean squared error

error on uncorrupted responses

$$\frac{1}{T}\sum_{t=1}^{T}\left(y_t - \widehat{y}_t\right)^2$$

of $O(\eta R^2)$, where $\|w^*\| \leq 1$ and $\|x_t\| \leq R$

This can be quite far from optimal

Lower Bound: CAMSE must be at least $\Omega(\eta^2\sigma^2)$

variance of stochastic noise

OUTLINE

Part I: Supervised Learning

- PAC Learning and Robustness
- Our Results and Framework
- Applications to Fairness

Part II: Online Learning

- Regression and Clean MSE
- Dynamic Range vs. Variance
- Our Results and Extensions to Contextual Bandits

OUTLINE

Part I: Supervised Learning

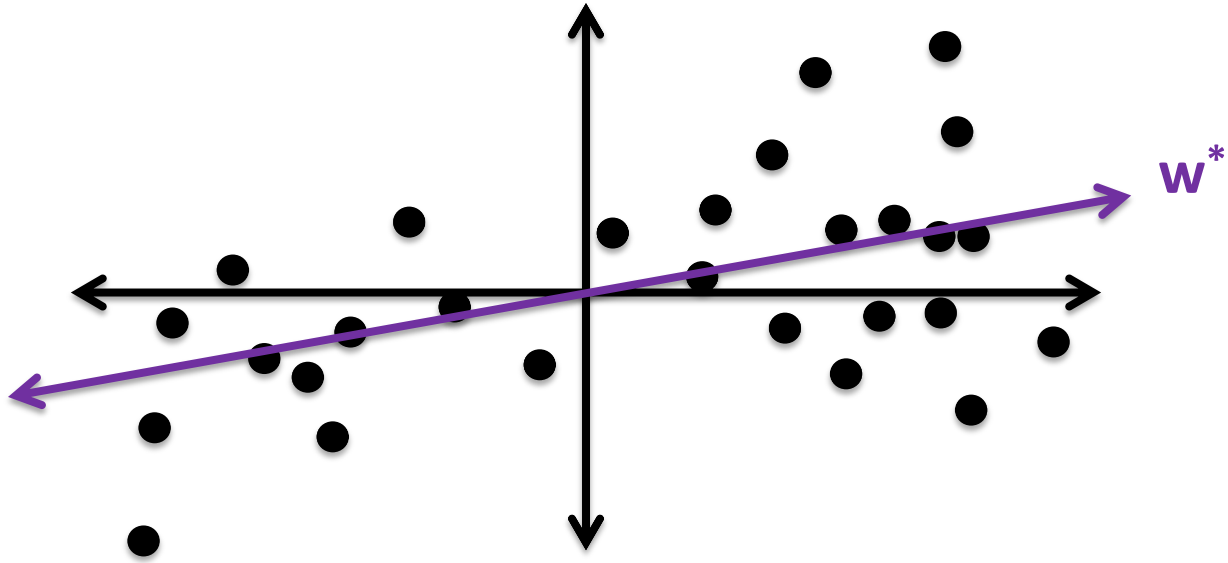
- PAC Learning and Robustness
- Our Results and Framework
- Applications to Fairness

Part II: Online Learning

- Regression and Clean MSE
- **Dynamic Range vs. Variance**
- Our Results and Extensions to Contextual Bandits

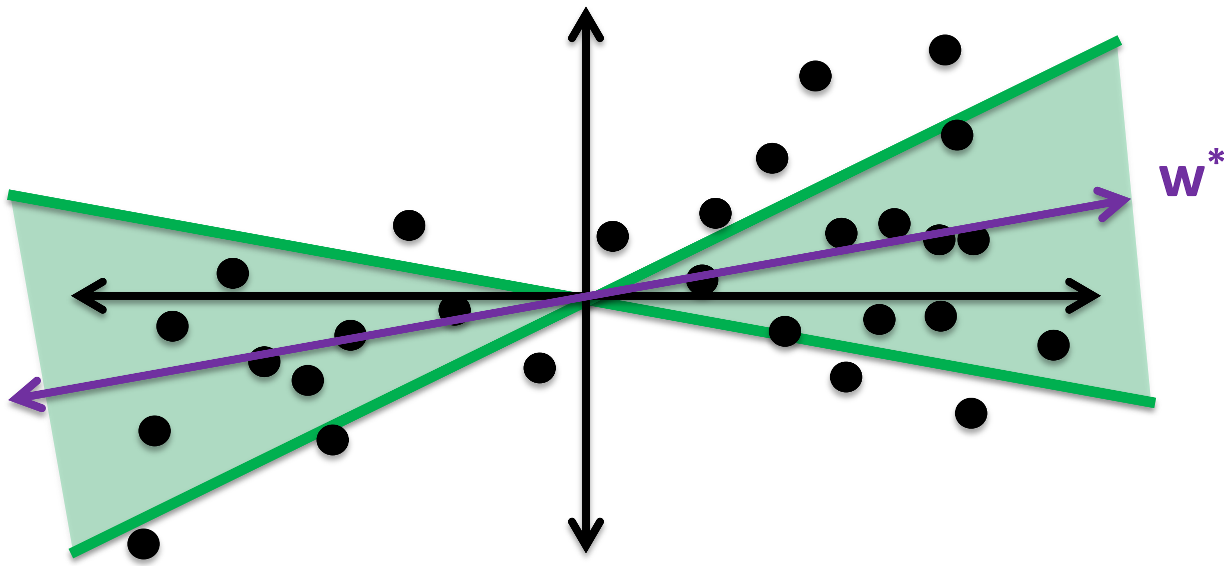
CORRUPTIONS AND THE DYNAMIC RANGE

Easy Case: The range^2 and variance are on the same order



CORRUPTIONS AND THE DYNAMIC RANGE

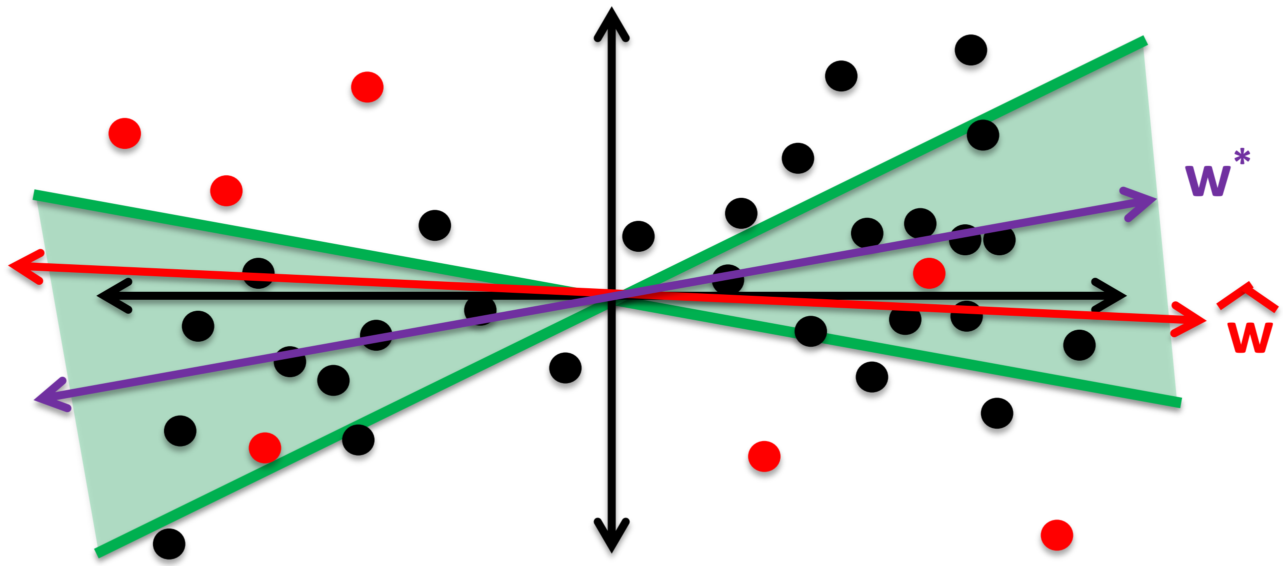
Easy Case: The range^2 and variance are on the same order



The cone of lines achieving nearly optimal CAMSE is **wide**

CORRUPTIONS AND THE DYNAMIC RANGE

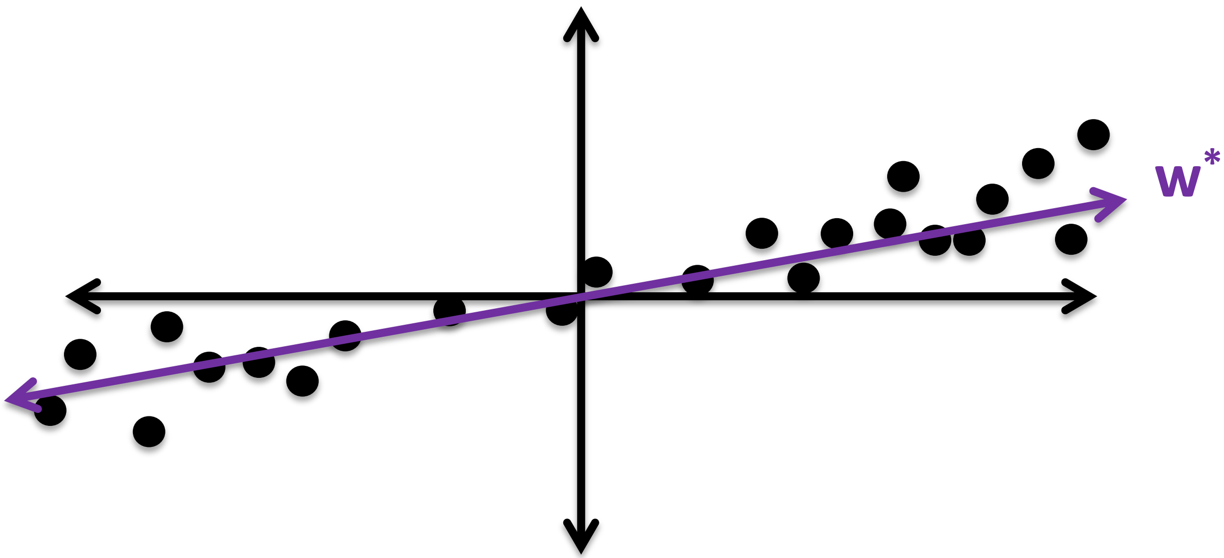
Easy Case: The range^2 and variance are on the same order



The cone of lines achieving nearly optimal CAMSE is **wide**, and corruptions cannot mess things up too much!

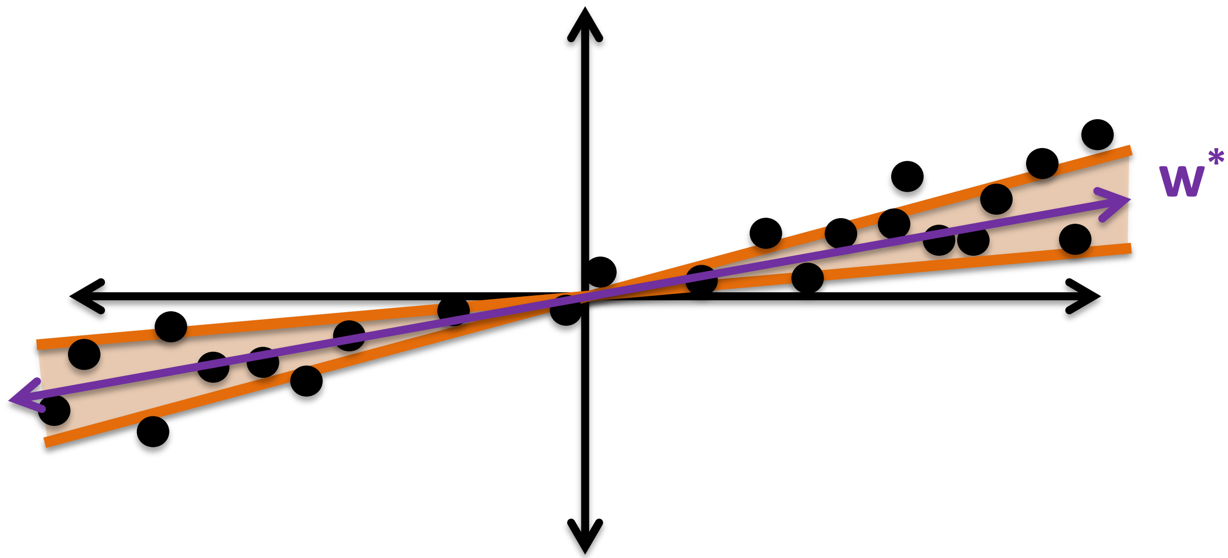
CORRUPTIONS AND THE DYNAMIC RANGE

Hard Case: The range^2 is much larger than the variance



CORRUPTIONS AND THE DYNAMIC RANGE

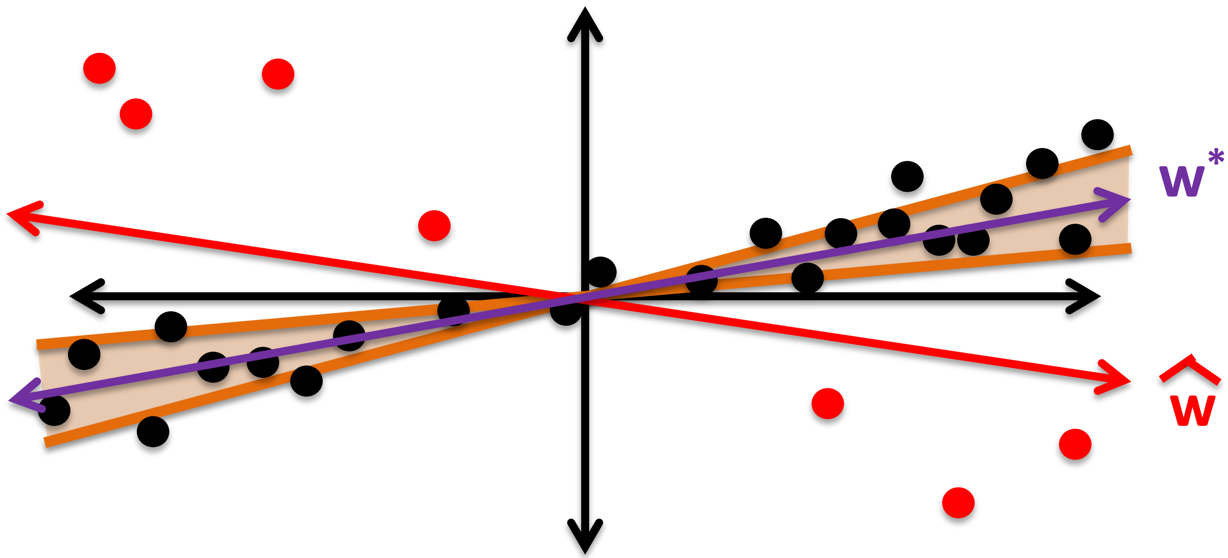
Hard Case: The range^2 is much larger than the variance



The cone of lines achieving nearly optimal CAMSE is **narrow**

CORRUPTIONS AND THE DYNAMIC RANGE

Hard Case: The range^2 is much larger than the variance



The cone of lines achieving nearly optimal CAMSE is **narrow**, and corruptions **can** mess things up badly

OUTLINE

Part I: Supervised Learning

- PAC Learning and Robustness
- Our Results and Framework
- Applications to Fairness

Part II: Online Learning

- Regression and Clean MSE
- Dynamic Range vs. Variance
- Our Results and Extensions to Contextual Bandits

OUTLINE

Part I: Supervised Learning

- PAC Learning and Robustness
- Our Results and Framework
- Applications to Fairness

Part II: Online Learning

- Regression and Clean MSE
- Dynamic Range vs. Variance
- **Our Results and Extensions to Contextual Bandits**

OUR RESULTS

Theorem: There is a simple and practical algorithm that achieves CAMSE $\widetilde{O}(\eta^2\sigma^2)$ for online linear regression with Huber contamination

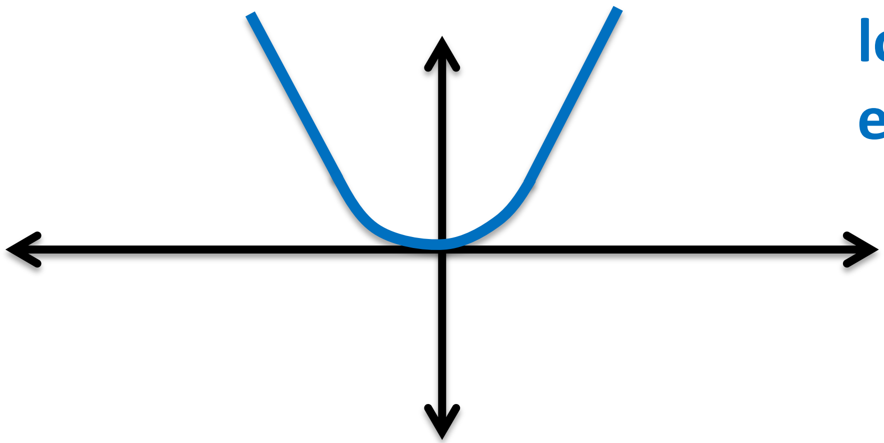
OUR RESULTS

Theorem: There is a simple and practical algorithm that achieves CAMSE $\widetilde{O}(\eta^2\sigma^2)$ for online linear regression with Huber contamination

Our algorithm is a simple twist on least trimmed squares

NOTES

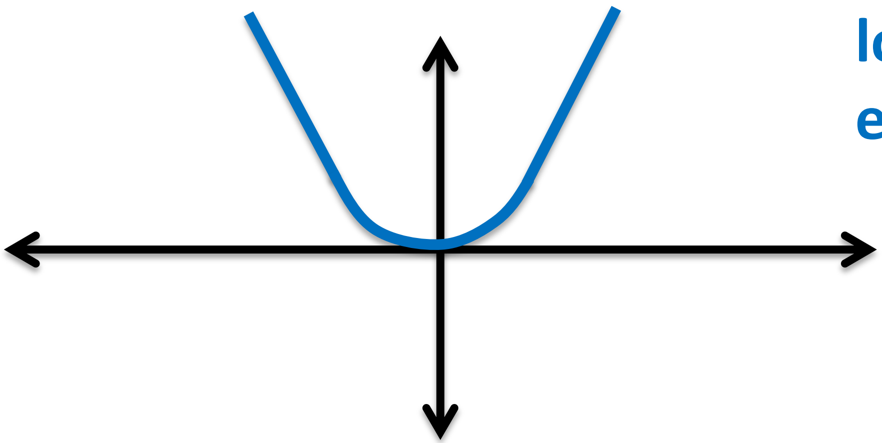
Classic heuristics, like **Huber regression** provably fail



locally quadratic, eventually linear

NOTES

Classic heuristics, like **Huber regression** provably fail



locally quadratic,
eventually linear

Many works in stronger contamination models, but work in offline setting and make distributional assumptions

[Klivans, Kothari, Meka], **[Prasad et al.]**, **[Diakonikolas et al.]**, **[Bakshi, Prasad]**, **[Zhu et al.]**, **[Cherapnamjeri et al.]**, …

LEAST TRIMMED SQUARES

In 1984, Rousseeuw introduced a powerful methodology

- **Initialize** S to be the set of all points
- **Repeat**
  - Set $\widehat{w}$ to be the output of running ordinary least squares on S
  - Set S to be points with smallest residuals under $\widehat{w}$

LEAST TRIMMED SQUARES, REVISITED

In 1984, Rousseeuw introduced a powerful methodology

- **Initialize** S to be the set of all points
- **Repeat**
  - Set $\widehat{w}$ to be the output of running ordinary least squares on S
  - ~~Set S to be points with smallest residuals under~~ $\cancel{\widehat{w}}$

Our twist: Set $S \leftarrow \arg\min_{S \in \mathcal{F}} \sum_{i \in S} \Big(y_i - \langle \widehat{w}, x_i \rangle \Big)^2$

where $\mathcal{F}$ is the set of all subsets whose covariance is approx. the same as covariance of all points

ONLINE SETTING

Finally, we can build an online algorithm from the offline one using cutting planes methods

SYNTHETIC EXPERIMENTS

Set N = 10000, d = 500, R = 1, σ = 0.1 and true regressor

$$w = [1, 0, 0, \cdots, 0]$$

SYNTHETIC EXPERIMENTS

Set N = 10000, d = 500, R = 1, σ = 0.1 and true regressor

$$w = [1, 0, 0, \cdots, 0]$$

And to model rare, but predictive features set

$$x_t = \begin{cases} \text{approximately 0, with probability 0.8} \\ [1, 0, 0, \cdots, 0] \text{, with probability 0.1} \\ [-1/2, 0, 0, \cdots, 0] \text{, else} \end{cases}$$

SYNTHETIC EXPERIMENTS

Set N = 10000, d = 500, R = 1, σ = 0.1 and true regressor

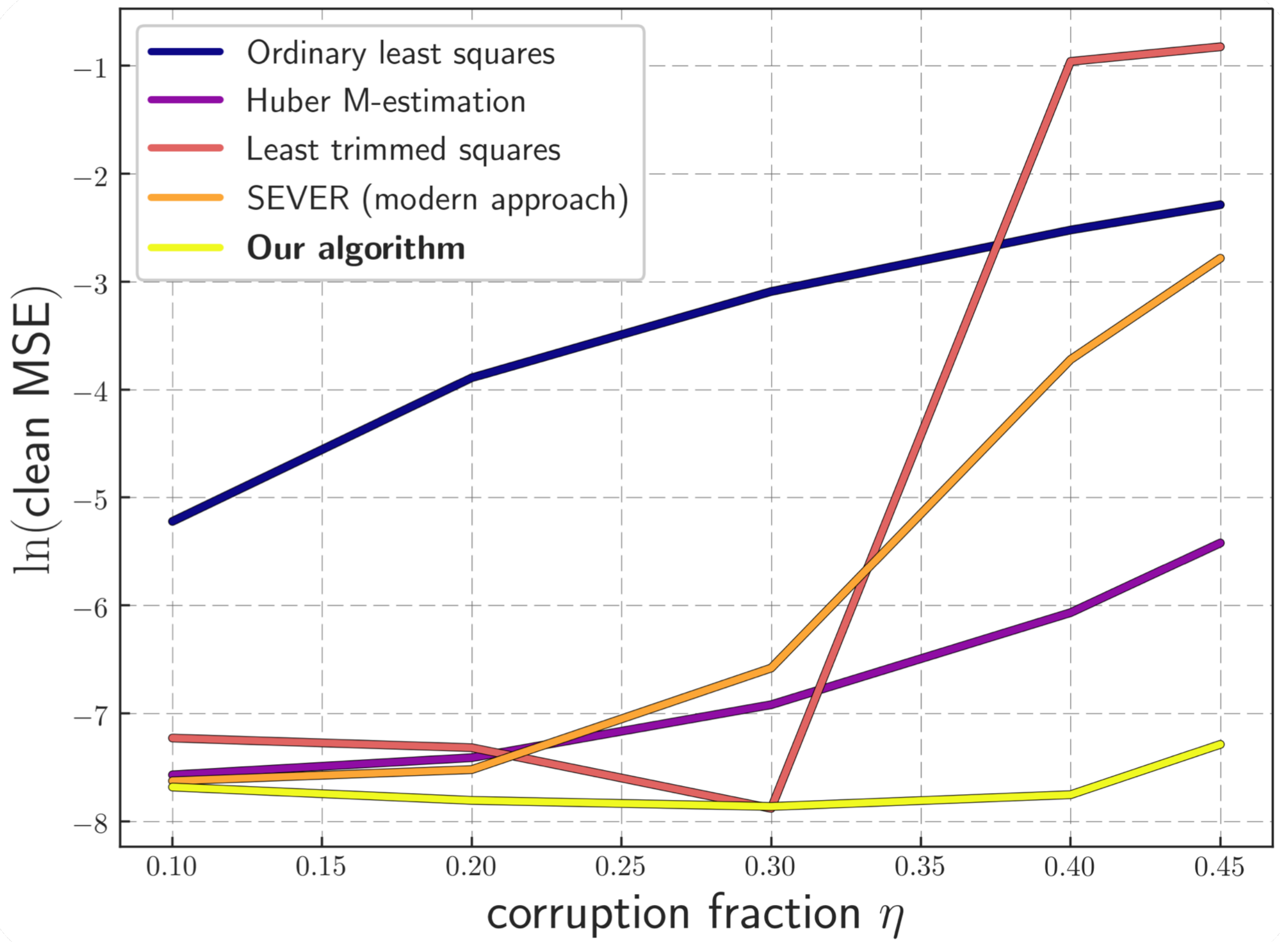
$$w = [1, 0, 0, \cdots, 0]$$

And to model rare, but predictive features set

$$x_t = \begin{cases} \text{approximately 0, with probability 0.8} \\ [1, 0, 0, \cdots, 0]\text{, with probability 0.1} \\ [-1/2, 0, 0, \cdots, 0]\text{, else} \end{cases}$$

Adversary: Zero out a random fraction of the responses

SYNTHETIC EXPERIMENTS



LINEAR CONTEXTUAL BANDITS

What about sequential decision making?

LINEAR CONTEXTUAL BANDITS

What about sequential decision making?

Goal: Use available information to make better decisions

LINEAR CONTEXTUAL BANDITS

What about sequential decision making?

Goal: Use available information to make better decisions

In each round, we

(1) **Observe a context** $x_t = (x_{ta})_{a \in \mathcal{A}}$, which is a collection of high-dimensional vectors

LINEAR CONTEXTUAL BANDITS

What about sequential decision making?

Goal: Use available information to make better decisions

In each round, we

(1) **Observe a context** $x_t = (x_{ta})_{a \in \mathcal{A}}$, which is a collection of high-dimensional vectors

(2) **Play an action** $a_t \in \mathcal{A}$ and incur loss

$$\ell_t(a_t) = \langle w^*, x_{t,a} \rangle + \sigma_t$$

LINEAR CONTEXTUAL BANDITS

What about sequential decision making?

Goal: Use available information to make better decisions

In each round, we

(1) **Observe a context** $x_t = (x_{ta})_{a \in \mathcal{A}}$, which is a collection of high-dimensional vectors

(2) **Play an action** $a_t \in \mathcal{A}$ and incur loss

$$\ell_t(a_t) = \langle w^*, x_{t,a} \rangle + \sigma_t$$

Note: Can extend to infinite dimensional spaces, using kernels

Many applications:

E-commerce: Selecting ads to display, based on user history

e.g. **[Abe, Nakamura]**

Many applications:

E-commerce: Selecting ads to display, based on user history

e.g. **[Abe, Nakamura]**

Collaborative Filtering: Personalizing news recommendations

e.g. **[Li, Chu, Langford, Schapire]**

Many applications:

E-commerce: Selecting ads to display, based on user history

e.g. [Abe, Nakamura]

Collaborative Filtering: Personalizing news recommendations

e.g. [Li, Chu, Langford, Schapire]

Mobile Health: Just-in-time interventions to modify behavior, adapted to the user e.g. [Nahum-Shani et al.]

Can we learn good policies in spite of corrupted responses?

Can we learn good policies in spite of corrupted responses?

e.g. bots clicking on ads, to **manipulate prices**

Can we learn good policies in spite of corrupted responses?

e.g. bots clicking on ads, to **manipulate prices**

e.g. **connectivity issues** in mobile health

Can we learn good policies in spite of corrupted responses?

e.g. bots clicking on ads, to **manipulate prices**

e.g. **connectivity issues** in mobile health

e.g. using **proxy variables** instead of the actual losses

Can we learn good policies in spite of corrupted responses?

e.g. bots clicking on ads, to **manipulate prices**

e.g. **connectivity issues** in mobile health

e.g. using **proxy variables** instead of the actual losses

Yes! Standard approach uses linear regression as a subroutine

Can we learn good policies in spite of corrupted responses?

e.g. bots clicking on ads, to **manipulate prices**

e.g. **connectivity issues** in mobile health

e.g. using **proxy variables** instead of the actual losses

Yes! Standard approach uses linear regression as a subroutine

[Foster, Rakhlin]

make accurate predictions about the loss of an action → **choose a good action**

Can we learn good policies in spite of corrupted responses?

e.g. bots clicking on ads, to **manipulate prices**

e.g. **connectivity issues** in mobile health

e.g. using **proxy variables** instead of the actual losses

Yes! Standard approach uses linear regression as a subroutine

[Foster, Rakhlin]

make accurate predictions about the loss of an action → **choose a good action**

Thus we get new algorithms for linear contextual bandits that are **provably resistant to adversarial corruptions**

CONCLUDING REMARKS

Robustness is a spectrum

CONCLUDING REMARKS

Robustness is a spectrum

It's not just about handling more powerful adversaries, but also finding the right compromises that avoid computational hardness

CONCLUDING REMARKS

Robustness is a spectrum

It's not just about handling more powerful adversaries, but also finding the right compromises that avoid computational hardness

Are there real-world applications where provably robust estimators can replace their non-robust counterparts?

Thanks!

Any Questions?