Robust Learning from Multiple Sources

Christoph H. Lampert

Mathematical Machine Learning Seminar MPI MiS + UCLA
April 14th, 2022
Institute of Science and Technology Austria (ISTA)

- public research institute, opened in 2009
- located in outskirts of Vienna

**Focus on curiosity-driven basic research**
- avoiding boundaries between disciplines
- current 70 research groups
  - Computer Science, Mathematics, Physics, Chemistry, Biology, Neuroscience, Earth and Climate Sciences
- ELLIS unit since 2019

**We’re hiring!** (on all levels)
- interns, PhD students, postdocs
- faculty (tenure-track or tenured), ...

More information: chl@ist.ac.at or https://cvml.ist.ac.at
### Topics in Our Research Group

#### Machine Learning Theory
- Transfer Learning
- Multi-task Learning
- Lifelong/Meta-Learning
- Multi-source/Federated Learning

#### Models/Algorithms
- Zero-shot Learning
- Continual Learning
- Weakly-supervised Learning
- Trustworthy/Robust Learning

#### Learning for Computer Vision
- Scene Understanding
- Generative Models
- Abstract Reasoning
- Semantic Representations
# Topics in Our Research Group

## Machine Learning Theory
- Transfer Learning
- Multi-task Learning
- Lifelong/Meta-Learning
- Multi-source/Federated Learning

## Models/Algorithms
- Zero-shot Learning
- Continual Learning
- Weakly-supervised Learning
- Trustworthy/Robust Learning

## Learning for Computer Vision
- Scene Understanding
- Generative Models
- Abstract Reasoning
- Semantic Representations
Training data from multiple sources
Training data from multiple sources

How much can be learned even if some data is corrupted or manipulated?
Schedule

Overview

Reminder: (Statistical) Learning Theory

Robust Learning From Untrusted Sources

Robust Fair Learning

Slides available at: http://cvml.ist.ac.at
Reminder: Supervised Learning
Setting:

- **Inputs:** $x \in \mathcal{X}$, e.g. strings, images, vectors, ...
- **Outputs:** $y \in \mathcal{Y}$. For simplicity, we use $\mathcal{Y} = \{\pm 1\}$ (binary classification)
- **Probability distribution:** $p(x, y)$ over $\mathcal{X} \times \mathcal{Y}$, unknown to the learner
- **Loss function:** $\ell : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$. For simplicity, we use 0/1-loss: $\ell(y, \bar{y}) = \mathbb{I}[y \neq \bar{y}]$

Abstract Goal:

- find a **prediction function**, $f : \mathcal{X} \rightarrow \mathcal{Y}$, such that the expected loss
  \[
  \text{er}(h) = E_{(x, y) \sim p}[\ell(y, f(x))] = Pr_{(x, y) \sim p}\{f(x) \neq y\}
  \]
  on future data is small.
Learning from data:

- training data: \( S = \{(x_1, y_1), \ldots, (x_m, y_m)\} \) i.i.d. \( p \)
- hypothesis class: \( \mathcal{H} = \{h : \mathcal{X} \to \mathcal{Y}\} \)
- learning algorithm \( \mathcal{L} : \mathcal{P}(\mathcal{X} \times \mathcal{Y}) \to \mathcal{H}, \quad \mathcal{P}(\cdot) = \text{power set} \)

- input: a training set, \( S \subset \mathcal{X} \times \mathcal{Y} \),
- output: a trained model \( \mathcal{L}(S) \in \mathcal{H} (= \text{prediction function}) \).
Learning from data:

- training data: $S = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ \text{i.i.d.} $p$
- hypothesis class: $\mathcal{H} = \{h : \mathcal{X} \rightarrow \mathcal{Y}\}$
- learning algorithm $\mathcal{L} : \mathcal{P}(\mathcal{X} \times \mathcal{Y}) \rightarrow \mathcal{H}$, $\mathcal{P}(\cdot) = \text{power set}$
  - input: a training set, $S \subset \mathcal{X} \times \mathcal{Y}$,
  - output: a trained model $\mathcal{L}(S) \in \mathcal{H}$ (= prediction function).

Central question in Statistical Learning Theory:

Is there a universal learning algorithm, such that: $\text{er}(\mathcal{L}(S)) \xrightarrow{|S| \rightarrow \infty} \min_{h \in \mathcal{H}} \text{er}(h)$?
Learning from data:

▶ training data: $S = \{ (x_1, y_1), \ldots, (x_m, y_m) \}$ \( \sim \) \( p \)

▶ hypothesis class: $\mathcal{H} = \{ h : \mathcal{X} \rightarrow \mathcal{Y} \}$

▶ learning algorithm $L : \mathcal{P}(\mathcal{X} \times \mathcal{Y}) \rightarrow \mathcal{H}$, $\mathcal{P}(\cdot) =$ power set

- input: a training set, $S \subset \mathcal{X} \times \mathcal{Y}$,
- output: a trained model $L(S) \in \mathcal{H}$ (= prediction function).

Central question in Statistical Learning Theory:

Is there a universal learning algorithm, such that: $\lim_{|S| \rightarrow \infty} \min_{h \in \mathcal{H}} \text{er}(h)$?

Classic result: [Vapnik & Chervonenkis, 1971], [Blumer, Ehrenfeucht, Hassler, Warmuth, 1989]

If and only if $\text{VC}(\mathcal{H}) < \infty$, empirical risk minimization (ERM) does the job:

$L(S) \leftarrow \arg \min_{h \in \mathcal{H}} \text{er}_S(h)$ for $\text{er}_S(h) := \frac{1}{|S|} \sum_{(x, y) \in S} [f(x) \neq y]$.


Learning from unreliable/malicious data:

- training set: \( S = \{(x_1, y_1), \ldots, (x_m, y_m)\} \)
- but: data has issues: some data points might not really be samples from \( p \)

Formally: malicious adversary \( A \) can manipulate a fraction \( \alpha \) of the dataset.

- input: dataset \( S \)
- output: dataset \( S' \) with \( \lceil (1 - \alpha) m \rceil \) points are unchanged and \( \lfloor \alpha m \rfloor \) are arbitrary.

\( A \) can depend on the learning algorithms, etc.

Question: Is ERM still be a universally good learning strategy?

Classic Result: no! [Kearns and Li, 1993]

No learning algorithm can guarantee an error less than \( \alpha 1 - \alpha \) on future data! [L. G. Valiant. “Learning disjunctions of conjunctions”. IJCAI 1985]

Learning from unreliable/malicious data:

- **training set:** \( S = \{(x_1, y_1), \ldots, (x_m, y_m)\} \)
- but: data has issues: some data points might not really be samples from \( p \)
- formally: malicious adversary \( \mathcal{A} \) [Valiant, 1985]
  - \( \mathcal{A} \) can manipulate a fraction \( \alpha \) of the dataset
  - input: dataset \( S \)
  - output: dataset \( S' = \mathcal{A}(S) \) with \( \lceil(1 - \alpha)m\rceil \) points are unchanged and \( \lfloor \alpha m \rfloor \) are arbitrary
  - \( \mathcal{A} \) can depend on the learning algorithms, etc.

**Question:** Is ERM still be a universally good learning strategy?
Learning from unreliable/malicious data:

- **training set:** \( S = \{(x_1, y_1), \ldots, (x_m, y_m)\} \)
- **but:** data has issues: some data points might not really be samples from \( p \)
- **formally:** malicious adversary \( A \) [Valiant, 1985]
  - \( A \) can manipulate a fraction \( \alpha \) of the dataset
  - input: dataset \( S \)
  - output: dataset \( S' = A(S) \) with \([ (1 - \alpha) m ]\) points are unchanged and \([ \alpha m ]\) are arbitrary
  - \( A \) can depend on the learning algorithms, etc.

**Question:** Is ERM still be a universally good learning strategy?

**Classic Result:** no! [Kearns and Li, 1993]

No learning algorithm can guarantee an error less than \( \frac{\alpha}{1 - \alpha} \) on future data!

[L. G. Valiant. "Learning disjunctions of conjunctions". IJCAI 1985]
Learning from Multiple Sources
If all sources are i.i.d. samples from the correct data distribution

→ naive strategy "merge all datasets and train a classifier" works perfectly
If some sources are not reliable, naive strategy can fail miserably!
Robust Learning from Unreliable or Malicious Sources

Disclaimer: "These results have been modified from their original form. They have been edited to fit the screen and the allotted time slot."

[N. Konstantinov, CHL. "Robust Learning from Untrusted Sources", ICML 2019]
Learning from Multiple Sources

- multiple training sets: $S_1, S_2, \ldots, S_N$
  - each $S_i = \{(x_{i1}^i, y_{1i}^i), \ldots, (x_{mi}^i, y_{mi}^i)\}$ i.i.d. $\sim p$
- multi-source learning algorithm: $\mathcal{L}: (\mathcal{X} \times \mathcal{Y})^{N \times m} \rightarrow \mathcal{H}$
  - input: training sets, $S_1, S_2, \ldots, S_N$
  - output: one hypothesis $\mathcal{L}(S_1, \ldots, S_N) \in \mathcal{H}$ (= a trained model).
Learning from Multiple Unreliable/Malicious Sources

- multiple training sets: \( S_1, S_2, \ldots, S_N \)
  - each \( S_i = \{(x^i_1, y^i_1), \ldots, (x^i_m, y^i_m)\} \sim \text{i.i.d. } p \)

- multi-source learning algorithm: \( \mathcal{L} : (\mathcal{X} \times \mathcal{Y})^{N \times m} \rightarrow \mathcal{H} \)
  - input: training sets, \( S'_1, S'_2, \ldots, S'_N = \mathcal{A}(S_1, \ldots, S_N) \)
  - output: one hypothesis \( \mathcal{L}(S'_1, S'_2, \ldots, S'_N) \in \mathcal{H} (= \text{a trained model}) \)

- adversary \( \mathcal{A} \)
  - input: data sets \( S_1, \ldots, S_N \)
  - output: data sets \( S'_1, \ldots, S'_N \),
    - of which \( \lceil(1 - \alpha)N\rceil \) are identical to before and \( \lfloor\alpha N\rfloor \) are arbitrary
  - the adversary might know the training algorithm
Learning from Multiple Unreliable/Malicious Sources

- multiple training sets: $S_1, S_2, \ldots, S_N$
  - each $S_i = \{(x^i_1, y^i_1), \ldots, (x^i_m, y^i_m)\}$ i.i.d. $p$

- multi-source learning algorithm: $L : (X \times Y)^{N \times m} \rightarrow \mathcal{H}$
  - input: training sets, $S'_1, S'_2, \ldots, S'_N = \mathcal{A}(S_1, \ldots, S_N)$
  - output: one hypothesis $L(S'_1, S'_2, \ldots, S'_N) \in \mathcal{H}$ (= a trained model).

- adversary $\mathcal{A}$
  - input: data sets $S_1, \ldots, S_N$
  - output: data sets $S'_1, \ldots, S'_N$, of which $\lceil (1 - \alpha)N \rceil$ are identical to before and $\lfloor \alpha N \rfloor$ are arbitrary
  - the adversary might know the training algorithm

Is there a universal learning algorithm, i.e. $\lim_{m \to \infty} \min_{h \in \mathcal{H}} \text{er}(L(S'_1, \ldots, S'_N)) = \text{er}(h)$?
Related Work

Robust learning from a single dataset

- no universal algorithm: minimum guaranteeable error is $\alpha_1 - \alpha$ [Kearns and Li, 1993]

- identical to our situation when each dataset consists of a single point, $m = 1$ → only $N \to \infty$ will probably not suffice to learn arbitrarily well

Collaborative learning

- universal learning algorithm exists [Blum et al., 2017], [Qiao, 2018]

Density estimation from untrusted batches

- possible, but not applicable to supervised learning [Qiao and Valiant, 2018], [Jain and Orlitsky, 2020]

Byzantine-robust distributed optimization

- specific solutions for gradient-based optimization [Yin et al., 2018], [Alistarh et al., 2018]

- results focus on convergence analysis
Related Work

Robust learning from a single dataset

- no universal algorithm: minimum guaranteeable error is $\frac{\alpha}{1-\alpha}$ [Kearns and Li, 1993]
- identical to our situation when each dataset consists of a single point, $m = 1$
  $\rightarrow$ only $N \to \infty$ will probably not suffice to learn arbitrarily well
Related Work

Robust learning from a single dataset

- no universal algorithm: minimum guaranteable error is $\frac{\alpha}{1-\alpha}$ [Kearns and Li, 1993]
- identical to our situation when each dataset consists of a single point, $m = 1$ → only $N \to \infty$ will probably not suffice to learn arbitrarily well

Collaborative learning (multiple parties together learn individual models)

- universal learning algorithm exists [Blum et al., 2017], [Qiao, 2018]
Related Work

**Robust learning from a single dataset**

- no universal algorithm: minimum guaranteable error is $\frac{\alpha}{1-\alpha}$ [Kearns and Li, 1993]
- identical to our situation when each dataset consists of a single point, $m = 1$ → only $N \to \infty$ will probably not suffice to learn arbitrarily well

**Collaborative learning** (multiple parties together learn *individual models*)

- universal learning algorithm exists [Blum et al., 2017], [Qiao, 2018]

**Density estimation from untrusted batches**

- possible, but not applicable to supervised learning [Qiao and Valiant, 2018], [Jain and Orlitsky, 2020]
Related Work

Robust learning from a single dataset

- no universal algorithm: minimum guaranteable error is $\frac{c}{1-c}$ [Kearns and Li, 1993]
- identical to our situation when each dataset consists of a single point, $m = 1$ → only $N \to \infty$ will probably not suffice to learn arbitrarily well

Collaborative learning (multiple parties together learn individual models)

- universal learning algorithm exists [Blum et al., 2017], [Qiao, 2018]

Density estimation from untrusted batches

- possible, but not applicable to supervised learning [Qiao and Valiant, 2018], [Jain and Orlitsky, 2020]

Byzantine-robust distributed optimization

- specific solutions for gradient-based optimization [Yin et al., 2018], [Alistarh et al., 2018]
- results focus on convergence analysis
Our Result

**Theorem** [Konstantinov et al., 2020]

There exists a learning algorithm, $\mathcal{L}$, such that with high probability:

$$\text{er}(\mathcal{L}(S'_1, \ldots, S'_N)) \leq \min_{h \in \mathcal{H}} \text{er}(h) + \tilde{O}\left(\frac{1}{\sqrt{(1 - \alpha)Nm}} + \alpha \frac{1}{\sqrt{m}}\right),$$

with $S'_1, \ldots, S'_N = \mathcal{A}(S_1, \ldots, S_N)$ for any adversary $\mathcal{A}$ with $\alpha < \frac{1}{2}$.

($\tilde{O}$-notation hides constant and logarithmic factors)

**Question:** why is learning easier from multiple sources than from a single one?

**Answer:** it’s not. But the task for the adversary is harder!
- single source: no restrictions how to manipulate the data
- multi-source: manipulation must adhere to the source structure

**Algorithm idea:** exploit law of large numbers
1. majority of datasets are unperturbed
2. for $m \to \infty$ these start to look more and more similar
3. we can identify (at least) the unperturbed datasets
4. we perform ERM on the union of only those
Robust multi-source learning algorithm:

- **Input:** datasets $S'_1, \ldots, S'_N$
- **Input:** suitable distance measure $d$ between datasets
- **Input:** suitable threshold value $\theta$

**Step 1)** identify which sources to trust

- compute all pairwise distance $d_{ij}$ between datasets $S'_1, \ldots, S'_N$
- for any $i$: if $d_{ij} < \theta$ for at least $\left\lfloor \frac{N}{2} \right\rfloor$ values of $j \neq i$, then $T \leftarrow T \cup \{i\}$

**Step 2)** merge data from all sources $S'_i$ with $i \in T$ into a new dataset $\tilde{S}$

**Step 3)** minimize training error on $\tilde{S}$

Open choices:

- distance measure $d$ (discussed later), threshold $\theta$ (see paper)
All datasets clean
All datasets clean
All datasets clean
All datasets clean
All datasets clean
All datasets clean
All datasets clean
All datasets clean
All datasets clean $\rightarrow$ all datasets included $\rightarrow$ same as (optimal) naive algorithm
Some datasets manipulated
Some datasets manipulated $\rightarrow$ manipulated datasets excluded
Consistent manipulations
Consistent manipulations $\rightarrow$ manipulated datasets excluded
Some datasets manipulated to look like originals
Some datasets manipulated to look like originals → all datasets included.
What properties does the distance measure $d$ need?

1) ‘clean’ datasets should get grouped together: $S, \hat{S} \sim p \Rightarrow d(S, \hat{S}) \rightarrow \infty \rightarrow 0$

2) if manipulated datasets are grouped with the clean ones, they should not hurt the learning step $d(S, \hat{S})$ is small $\Rightarrow L(\hat{S}) \approx L(S)$

Observation:
▶ many candidate distances do not fulfill both conditions simultaneously:
▶ geometric: average Euclidean distance, Chamfer distance, Haussdorf distance, ...
▶ probabilistic: Wasserstein distance, total variation, KL-divergence, ...
▶ discrepancy distance does fulfill the conditions!
What properties does the distance measure $d$ need?

1) ‘clean’ datasets should get grouped together:

$$S, \hat{S} \sim p \implies d(S, \hat{S}) \xrightarrow{m \to \infty} 0$$
What properties does the distance measure \( d \) need?

1) ‘clean’ datasets should get grouped together:

\[
S, \hat{S} \sim p \quad \Rightarrow \quad d(S, \hat{S}) \underset{m \to \infty}{\to} 0
\]

2) if manipulated datasets are grouped with the clean ones, they should not hurt the learning step

\[
d(S, \hat{S}) \text{ is small} \quad \Rightarrow \quad \mathcal{L}(\hat{S}) \approx \mathcal{L}(S)
\]

Observation:

- many candidate distances do not fulfill both conditions simultaneously:
  - geometric: average Euclidean distance, Chamfer distance, Haussdorf distance, ...
  - probabilistic: Wasserstein distance, total variation, KL-divergence, ...
- discrepancy distance does fulfill the conditions!
What properties does the distance measure $d$ need?

1) ‘clean’ datasets should get grouped together:

\[ S, \hat{S} \sim p \quad \Rightarrow \quad d(S, \hat{S}) \xrightarrow{m \to \infty} 0 \]

2) if manipulated datasets are grouped with the clean ones, they should not hurt the learning step

\[ d(S, \hat{S}) \text{ is small} \quad \Rightarrow \quad \mathcal{L}(\hat{S}) \approx \mathcal{L}(S) \]

Observation:

- many candidate distances do not fulfill both conditions simultaneously:
  - geometric: average Euclidean distance, Chamfer distance, Haussdorf distance, ...  
  - probabilistic: Wasserstein distance, total variation, KL-divergence, ...
- discrepancy distance does fulfill the conditions!
Discrepancy Distance [Mansour et al. 2009]

For a set of classifiers $\mathcal{H}$ and datasets $S, \hat{S}$, define

$$\text{disc}(S, \hat{S}) = \max_{h \in \mathcal{H}} \left| \text{er}_S(h) - \text{er}_{\hat{S}}(h) \right|.$$ 

- maximal amount any classifier, $h \in \mathcal{H}$, can disagree between $S, \hat{S}$
- discrepancy can be estimated by training a classifier itself:
  - $S^{\pm} \leftarrow S$ with all $\pm 1$ labels flipped to their opposites
  - $\tilde{S} \leftarrow S^{\pm} \cup \hat{S}$
  - $\text{disc}(S, \hat{S}) \leftarrow 1 - 2 \min_{h \in \mathcal{H}} \text{er}_{\tilde{S}}(h)$ (minimal training error of any $h \in \mathcal{H}$ on $\tilde{S}$)
Two datasets, $S, \hat{S}$
Flip signs of $S$
Merge both datasets
Classifier with small training error $\rightarrow$ large discrepancy
Two datasets, $S, \hat{S}$
Flip signs of S
Merge both datasets
No classifier with small training error $\rightarrow$ small discrepancy
**Observation:** discrepancy distance has both property we need

1) Datasets from the same distribution (eventually) get grouped together
   ▶ for \( \text{VC}(\mathcal{H}) < \infty \), if \( S \) and \( \hat{S} \) are sampled from the same distribution, then
   \[
   \text{disc}(S, \hat{S}) \to 0 \quad \text{for} \quad |S|, |\hat{S}| \to \infty
   \]

2) Datasets that are grouped together cannot hurt the learning much

Consider:
   ▶ training set \( S_{\text{trn}} \), \( i.i.d. \) \( p \)
   ▶ arbitrary set \( \hat{S} \), potentially manipulated but with \( \text{disc}(S_{\text{trn}}, \hat{S}) \leq \theta \)
   ▶ test set \( S_{\text{tst}} \), \( i.i.d. \) \( p \)

Then, for every \( h \in \mathcal{H} \):
\[
\text{er}_{S_{\text{tst}}}(h) \leq \text{er}_{\hat{S}}(h) + \text{disc}(S_{\text{trn}}, \hat{S}) + \text{disc}(S_{\text{trn}}, S_{\text{tst}}) \leq \theta + \text{small by prop. 1)}
\]
Robust Fair Learning
Fairness-Aware Learning from Unreliable or Malicious Data

Disclaimer: "These results have been modified from their original form. They have been edited to fit the screen and the allotted time slot."

How to ensure that a classifier does not discriminate against certain groups?
Setting:

- Inputs: $x \in \mathcal{X}$, e.g. strings, images, vectors, ...
- Protected attribute: $a \in \mathcal{A}$, e.g. gender, age, race, ...
- Outputs: $y \in \mathcal{Y}$ (for simplicity: $\mathcal{Y} = \{0, 1\}$)
- Probability distribution: $p(x, a, y)$ over $\mathcal{X} \times \mathcal{A} \times \mathcal{Y}$
- Loss function: $\ell : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$ (for simplicity: 0/1-loss)

Abstract Goal:

- find a prediction function, $f : \mathcal{X} \rightarrow \mathcal{Y}$ low expected loss

$$\text{er}(h) = \mathbb{E}_{(x,y) \sim p}([f(x) \neq y]) = \Pr_{(x,y) \sim p}\{f(x) \neq y\}$$

that in addition fulfills some condition of (group) fairness.
Group Fairness:

- **demographic parity**: "all groups have the same success rate"

\[
\forall a, b \in A \quad p(f(X) = 1 | A = a) = p(f(X) = 1 | A = b)
\]

- **equality of opportunity**: "all groups have the true positive rate"

\[
\forall a, b \in A \quad p(f(X) = 1 | A = a, Y = 1) = p(f(X) = 1 | A = b, Y = 1)
\]

and many others [Barocas et al., 2019]

Several fairness-aware learning methods exist to enforce these criteria.

---

Fair Learning from unreliable/malicious data:

- **original training set:** \( S = \{(x_1, a_1, y_1), \ldots, (x_m, a_m, y_m)\} \)
- **adversary** \( \mathcal{A} \) can manipulate a fraction \( \alpha \) of the dataset
- **actual training set:** \( \mathcal{A}(S) \)

**Question:** Can a fairness-aware learner overcome the manipulation?
Fair Learning from unreliable/malicious data:

- **original training set:** \( S = \{(x_1, a_1, y_1), \ldots, (x_m, a_m, y_m)\} \)
- adversary \( \mathcal{A} \) can manipulate a fraction \( \alpha \) of the dataset
- actual training set: \( \mathcal{A}(S) \)

**Question:** Can a fairness-aware learner overcome the manipulation?

**Theorem** [Konstantinov and Lampert, 2021]

There is even for finite-sized hypothesis classes, \( \mathcal{H} \), for which:

- No learning algorithm can guarantee optimal fairness.
- This effect is independent of whether accuracy is also affected or not.
- The smaller the minority group, the stronger the bias.

Fairness-Aware Learning from Multiple Unreliable Sources

- multiple training sets: $S_1, S_2, \ldots, S_N \subset \mathcal{X} \times \mathcal{A} \times \mathcal{Y}$
- adversary $\mathcal{A}$ can manipulate $K = \lfloor \alpha N \rfloor$ of the datasets for $\alpha < \frac{1}{2}$
- actual training sets: $\mathcal{A}(S_1, \ldots, S_N)$

Is there a fairness-aware learning algorithm that overcomes such manipulations?
Fairness-Aware Learning from Multiple Unreliable Sources

- multiple training sets: $S_1, S_2, \ldots, S_N \subset X \times A \times Y$
- adversary $\mathfrak{A}$ can manipulate $K = \lfloor \alpha N \rfloor$ of the datasets for $\alpha < \frac{1}{2}$
- actual training sets: $\mathfrak{A}(S_1, \ldots, S_N)$

Is there a fairness-aware learning algorithm that overcomes such manipulations?

**Theorem** [Iofinova et al., 2021]

There exists a learning algorithm, $\mathcal{L}$, such that for $h^* = \mathcal{L}(\mathfrak{A}(S_1, \ldots, S_N))$ with high probability

$$\text{er}(h^*) \leq \min_{h \in \mathcal{H}} \text{er}(h) + \tilde{O}(\frac{1}{\sqrt{m}}), \quad \Gamma(h^*) \leq \min_{h \in \mathcal{H}} \Gamma(h) + \tilde{O}(\frac{1}{\sqrt{m}})$$

where $\Gamma$ is a quantitative measure of demographic parity fairness.

FLEA (Fair LEarning against Adversaries):

▶ **Input:** datasets $S'_1, \ldots, S'_N$

▶ **Input:** $\beta \leq \frac{1}{2}$ upper bound on fraction of malignant sources

▶ **Define:** distance measure $d(S, \hat{S}) = \text{disc}(S, \hat{S}) + \text{disp}(S, \hat{S}) + \text{disb}(S, \hat{S})$
  
  ▶ $\text{disc}(S, \hat{S})$: discrepancy as before
  
  ▶ $\text{disp}(S, \hat{S})$: maximal fairness difference of any classifier between $S$ and $\hat{S}$
  
  ▶ $\text{disb}(S, \hat{S})$: difference in protected group proportions

▶ **Step 1)** identify which sources to trust
  
  ▶ compute all pairwise distance $d_{ij}$ between datasets $S'_1, \ldots, S'_N$
  
  ▶ for any $i = 1, \ldots, N$: $q_i \leftarrow \beta$-quantile($d_{i1}, \ldots, d_{iN}$)
  
  ▶ $T \leftarrow \{i : q_i \leq \beta$-quantile($q_1, \ldots, q_N$)$\}$

▶ **Step 2)** merge data from all sources $S'_i$ with $i \in T$ into a new dataset $\tilde{S}$

▶ **Step 3)** train fairness-aware learning algorithm on $\tilde{S}$
Experimental Results

- bars are different data manipulations, designed to hurt accuracy or fairness
- simply training on all data often suboptimal
- other baselines often fail to overcome problems
- FLEA reliably recovers fairness and accuracy

More results and ablation studies in the paper.

Summary

**Bad news:**
- Learning is not robust to bad data.
- This can affect accuracy as well as fairness.

**Good news:**
- Modern data sets are often not monolithic but collected from multiple sources.
- Multi-source learning can be made robust to bad data sources.
- This holds for accuracy as well as fairness.

Thank you!

**Thanks to:**
- Nikola Konstantinov
- Jen Iofinova
- Elias Frantar
- Dan Alistarh

**Funding sources:**
- ISTA
- ERC
References


