McGill Bellairs Research Institute

Tuesday, Dec. 14th 2021

# Robust Crowdsourcing

Panagiotis (Panos) Traganitis

**Acknowledgements:** Prof. G. B. Giannakis

NSF grants 1901134, 2126052, 2128593

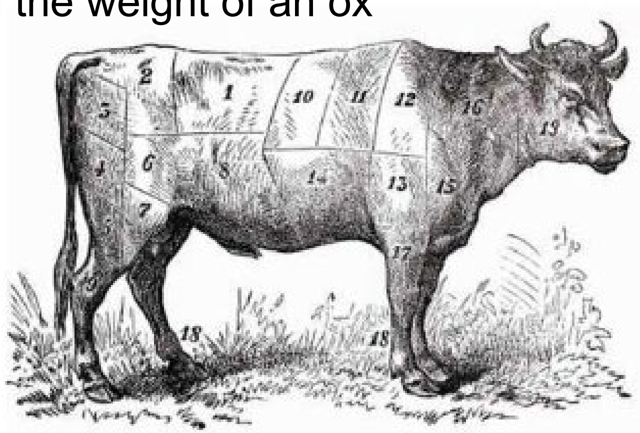
and ARO-STIR grant 00093896

University of Minnesota

Driven to Discover℠

# The Wisdom of Crowds

- **The parable of the ox** (Sir Francis Galton, 1906)
  - 787 people guessed the weight of an ox



  - Average crowd guess: **1,197 pounds** - True weight: **1,198 pounds!**

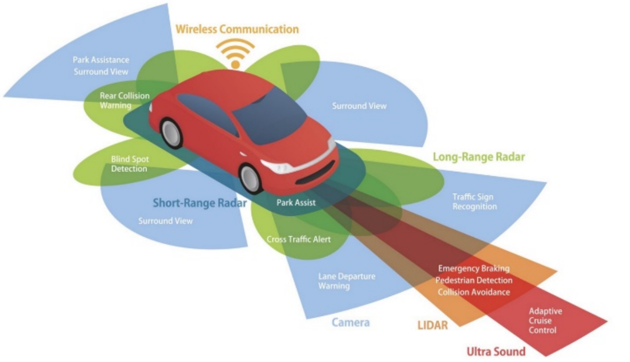
- **Who wants to be a millionaire –** Ask the audience

- Can we harness this wisdom in a principled way?

J. Surowiecki (2005). The Wisdom of Crowds. Anchor Books. pp. xv. ISBN 978-0-385-72170-7.

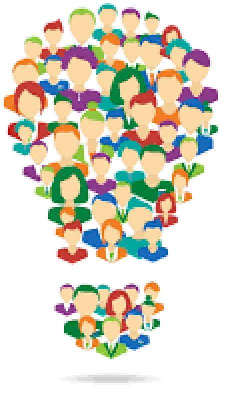
# Combining information/decisions

- ❑ Distributed detection/estimation [Tsitsiklis '89]




- ❑ Data fusion

- ❑ Ensemble learning
  - ➢ Combines results from multiple models
  - ➢ Can "boost" weak learners

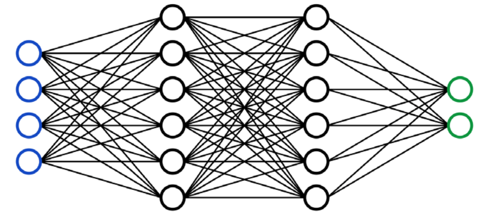


- ❑ **Crowdsourcing**
  - ➢ Provides labels for unlabeled datasets
  - ➢ Accomplish tasks w/o expert supervision
  - ➢ Cheap and efficient

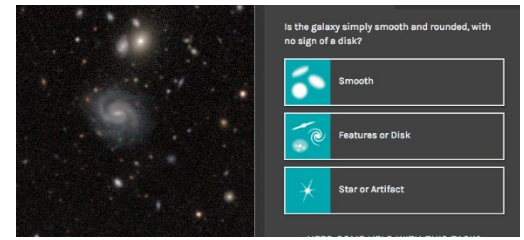

Google reCAPTCHA

- ❑ **Weak supervision / Data programming**

# Challenges and Impact

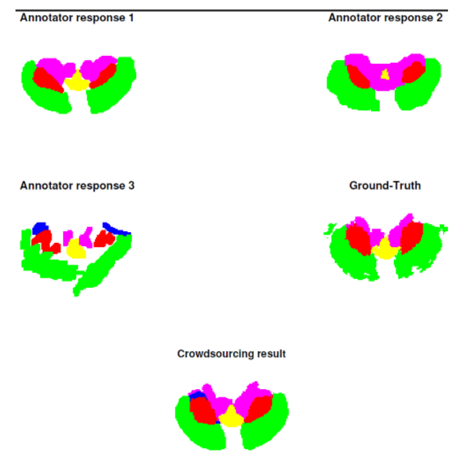


- ❑ Train and deploy complex models with limited supervision



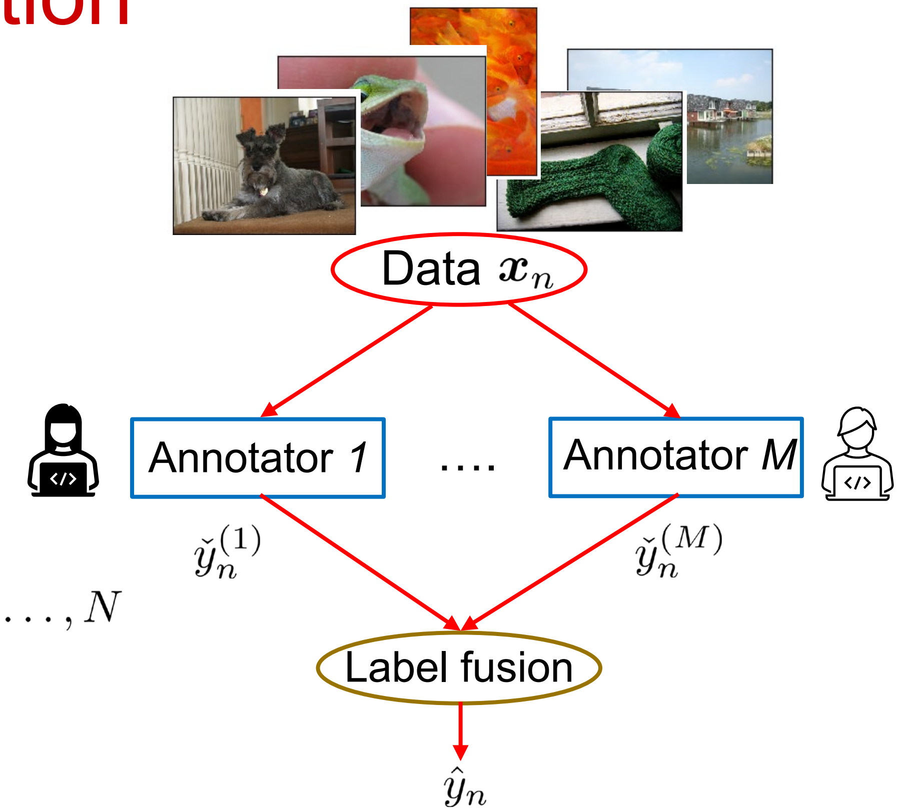

- ❑ Communication efficient distributed machine learning
- ❑ Citizen science


- ❑ Data sharing




- ❑ **Challenges**
  - ➢ Lack of ground-truth labels
  - ➢ Human annotators are not reliable
  - ➢ Sparsity of responses
  - ➢ Attacks by adversaries

# Crowdsourced classification




- $N$ data $\{(\boldsymbol{x}_n, y_n)\}_{n=1}^{N}$ , $K$ classes

- $M$ annotators – observe $\{\boldsymbol{x}_n\}_{n=1}^{N}$

- Annotator responses $\{\check{y}_n^{(m)}\}_{m=1}^{M}, \text{ for } n = 1, \ldots, N$
  "noisy" labels

- **Goal:** Given $\{\check{y}_n^{(m)}\}_{n=1,m=1}^{N,M}$ , find $\{\hat{y}_n\}_{n=1}^{N}$

**Q:** Which annotators are reliable?

**Q:** How to combine answers?

# Prior art

- **Model-free**
  - Majority voting (MV) - simplest method — Assumes all annotators are equally good
- **(Probabilistic) Model based**
  - Expectation Maximization (EM) [Dawid and Skene '79] — Guaranteed to converge only to a local optimum
  - Bayesian approaches [Kim and Ghahramani '12, Simpson et al '11] — Incorporates priors
  - "One-coin" model [Ghosh et al '11, Karger et al '13] — Restrictive assumptions
  - Moment-based methods — Can initialize the EM algorithm
    - One-coin model [Ma et al '18]
    - Binary classification [Jaffe et al '15]
    - Multi-class classification [Jain et al '14, Zhang et al '14, Traganitis et al '18, Ibrahim et al' 19]

# Outline

- Motivation
- Crowdsourcing 101 - Classification
    - Dawid and Skene (DS) model
    - The Expectation Maximization (EM) algorithm
    - Moment matching basics
- Crowdsourcing with spammers
- Crowdsourcing with cooperating adversaries
- Conclusion

# Probabilistic model for crowdsourcing

- Consider data: $\{(\boldsymbol{x}_n, y_n)\}_{n=1}^N \overset{\text{i.i.d.}}{\sim} \mathcal{P}$ $\qquad \boldsymbol{\pi} = [\Pr(y_n = 1), \ldots, \Pr(y_n = K)]^\top$

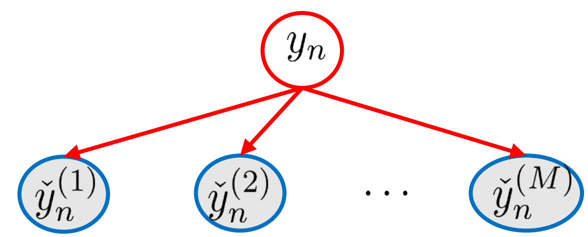
- Label fusion via MAP classifier

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \Pr(\boldsymbol{y}|\check{\mathbf{Y}}) = \arg\max_{\boldsymbol{y}} \Pr(\check{\mathbf{Y}}|\boldsymbol{y}) \Pr(\boldsymbol{y}) \overset{\text{i.i.d.}}{\Rightarrow} \boxed{\hat{y}_n = \arg\max_{k \in \{1,\ldots,K\}} \log \pi_k + \log \Pr(\{\check{y}_n^{(m)}\}_{m=1}^M | y_n = k)}$$

**(As1)**: Given ground-truth label $y_n$ annotator responses $\check{y}_n^{(1)}, \ldots, \check{y}_n^{(M)}$ are independent

$$\Pr\left(\check{y}_n^{(1)} = k_1, \ldots, \check{y}_n^{(M)} = k_M | y_n = k\right) = \prod_{m=1}^M \Pr\left(\check{y}_n^{(m)} = k_m | y_n = k\right)$$

$$\boxed{\hat{y}_n = \arg\max_{k \in \{1,\ldots,K\}} \log \pi_k + \sum_{m=1}^M \log \Pr\left(\check{y}_n^{(m)} | y_n = k\right)}$$




**Thm.** : Under **As1** there exist constants α,β > 0 such that the error probability of the MAP classifier satisfies

$$\mathcal{P}_e \le \alpha e^{-M\beta}$$


A. P. Dawid and A. M. Skene, "Maximum likelihood estimation of observer error-rates using the EM algorithm," Applied Statistics, pp. 20–28, 1979.

P. A. Traganitis, A. Pages-Zamora, and G. B. Giannakis, "Blind Multiclass Ensemble Classification," *IEEE Transactions on Signal Processing*, vol. 66, no. 18, pp. 4737-4752, September 2018.

# Dawid and Skene (DS) model

- Annotators characterized by confusion matrices $\{\mathbf{H}^{(m)}\}_{m=1}^{M}$

$$\mathbf{H}^{(m)} = \begin{bmatrix} \Pr(\check{y}_n^{(m)} = 1 | y_n = 1) & \Pr(\check{y}_n^{(m)} = 1 | y_n = 2) & \dots & \Pr(\check{y}_n^{(m)} = 1 | y_n = K) \\ \Pr(\check{y}_n^{(m)} = 2 | y_n = 1) & \Pr(\check{y}_n^{(m)} = 2 | y_n = 2) & & \vdots \\ \vdots & & \ddots & \\ \Pr(\check{y}_n^{(m)} = K | y_n = 1) & \dots & & \Pr(\check{y}_n^{(m)} = K | y_n = K) \end{bmatrix} = [\boldsymbol{h}_1^{(m)}, \dots, \boldsymbol{h}_K^{(m)}]$$

**(As2)**: Most annotators are better than random

$$\hat{y}_n = \underset{k \in \{1,\dots,K\}}{\arg\max} \; \log \pi_k + \sum_{m=1}^{M} \log(H^{(m)}(\check{y}_n^{(m)}, k))$$

- Simpler models realized by constraining $\{\mathbf{H}^{(m)}\}_{m=1}^{M}$
  - e.g. "One-coin" $\mathbf{H}^{(m)} = \left(\alpha^{(m)} - \frac{1-\alpha^{(m)}}{K-1}\right)\mathbf{I} + \frac{1-\alpha^{(m)}}{K-1}\mathbf{1}\mathbf{1}^\top, \quad 0 \le \alpha^{(m)} \le 1$

- Caveat: Parameters $\{\mathbf{H}^{(m)}\}_{m=1}^{M}, \boldsymbol{\pi}$ are unknown!
  - Can be estimated from $\check{\mathbf{Y}}$

A. P. Dawid and A. M. Skene, "Maximum likelihood estimation of observer error-rates using the EM algorithm," Applied Statistics, pp. 20–28, 1979.

# Expectation Maximization 101

- Popular tool for ML parameter estimation
  - Missing data problems
  - Mixture problems

Observed variables: $\check{\mathbf{Y}}$ Hidden variables: $\boldsymbol{y}$ parameters: $\boldsymbol{\theta}$

- EM seeks to maximize $L(\boldsymbol{\theta}) = \log \Pr(\check{\mathbf{Y}};\boldsymbol{\theta}) = \boxed{\log\left(\sum_{\boldsymbol{y}} \Pr(\boldsymbol{y},\check{\mathbf{Y}};\boldsymbol{\theta})\right)}$ Not available!

$$\mathbf{E}_{q(\boldsymbol{y})}\left[\log \Pr\left(\boldsymbol{y},\check{\mathbf{Y}};\boldsymbol{\theta}\right)\right] = \log \Pr(\check{\mathbf{Y}};\boldsymbol{\theta}) - D_{\mathrm{K}L}\left(q(\boldsymbol{y})||\Pr(\boldsymbol{y}|\check{\mathbf{Y}};\boldsymbol{\theta})\right)$$

- Two step iterative algorithm:
  - Expectation (E-)step $Q(\boldsymbol{\theta};\boldsymbol{\theta}[i]) = \mathbb{E}_{\boldsymbol{y}|\check{\mathbf{Y}};\boldsymbol{\theta}[i]}[\log \Pr(\boldsymbol{y},\check{\mathbf{Y}};\boldsymbol{\theta})]$ Missing data estimated using observed data and current parameters
  - Maximization (M-)step $\boldsymbol{\theta}[i+1] = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta};\boldsymbol{\theta}[i])$ Parameters are updated using estimated missing data
  - E- and M-steps repeated until convergence

- Basically Majorization-Minimization: M-step maximizes a lower bound of $L(\boldsymbol{\theta})$

- Nondecreasing sequence of L($\boldsymbol{\theta}$)'s - Converges to a stationary point

A.P. Dempster, N.M. Laird and D.B. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm". Journal of the Royal Statistical Society, Series B. 39 (1): 1–38, 1977.

# EM for crowdsourcing

- **Goal**: find $\boldsymbol{\theta} := [\boldsymbol{\pi}, \mathbf{H}^{(1)}, \ldots, \mathbf{H}^{(M)}]$ that maximizes: $\log \Pr(\check{\mathbf{Y}}; \boldsymbol{\theta})$
  - **S1**: Initialize $\boldsymbol{\theta}[0] := [\boldsymbol{\pi}[0], \mathbf{H}^{(1)}[0], \ldots, \mathbf{H}^{(M)}[0]]$
  - **S2**: E - step

$$Q(\boldsymbol{\theta}; \boldsymbol{\theta}[i]) = \mathbb{E}_{\boldsymbol{y}|\check{\mathbf{Y}};\boldsymbol{\theta}[i]}[\log \Pr(\boldsymbol{y}, \check{\mathbf{Y}}; \boldsymbol{\theta})] = \mathbb{E}_{\boldsymbol{y}|\check{\mathbf{Y}};\boldsymbol{\theta}[i]}[\log \Pr(\check{\mathbf{Y}}|\boldsymbol{y}; \boldsymbol{\theta})] + \mathbb{E}_{\boldsymbol{y}|\check{\mathbf{Y}};\boldsymbol{\theta}[i]}[\log \Pr(\boldsymbol{y}; \boldsymbol{\theta})]$$

$$= \sum_{n=1}^{N}\sum_{k=1}^{K} \log \Pr(\check{\mathbf{Y}}|y_n = k; \boldsymbol{\theta}) q_{nk} + \sum_{n=1}^{N}\sum_{k=1}^{K} \log \Pr(y_n = k; \boldsymbol{\theta}) q_{nk}$$

$$q_{nk} := \Pr(y_n = k|\check{\mathbf{Y}}; \boldsymbol{\theta}[i]) \propto \pi_k[i] \prod_{m=1}^{M}\prod_{k'=1}^{K} \left(H^{(m)}[i](k', k)\right)^{\mathbb{1}(\check{y}_n^{(m)}=k')}$$

Bayes rule

$$q_{nk}[i+1] = \frac{1}{Z}\exp\left(\log \pi_k[i] + \sum_{m=1}^{M}\sum_{k'=1}^{K} \mathbb{1}(\check{y}_n^{(m)} = k') \log(H^{(m)}[i](k', k))\right)$$

  - **S3**: M - step

$$\boldsymbol{\theta}[i+1] = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}; \boldsymbol{\theta}[i]) \Rightarrow$$

$$[\mathbf{H}^{(m)}[i+1]]_{k'k} = \frac{\sum_{n=1}^{N} q_{nk}[i+1]\mathbb{1}(\check{y}_n^{(m)} = k')}{\sum_{k''=1}^{K}\sum_{n=1}^{N} q_{nk}[i+1]\mathbb{1}(\check{y}_n^{(m)} = k'')}, \quad \forall m, k', k$$

$$\pi_k[i+1] = \frac{\sum_{n=1}^{N} q_{nk}[i+1]}{Z'}, \quad \forall k$$

- Steps 2 and 3 repeated until convergence

A. P. Dawid and A. M. Skene, "Maximum likelihood estimation of observer error-rates using the EM algorithm," Applied Statistics, pp. 20–28, 1979.

# Statistics of annotator responses

- **Q:** Can we estimate $\{\mathbf{H}^{(m)}\}_{m=1}^{M}, \boldsymbol{\pi}$ w/out EM?
- **A:** Moment matching
- Convert annotator responses to vector format, i.e. One-hot encoding $\check{y}_n^{(m)} = k \Rightarrow \check{\mathbf{y}}_n^{(m)} = \boldsymbol{e}_k$

$$\mathbb{E}[\check{\mathbf{y}}_n^{(m)} | y_n = k] = \sum_{k'=1}^{K} \boldsymbol{e}_{k'} \Pr\left(\check{y}_n^{(m)} = k' | y_n = k\right) = \boldsymbol{h}_k^{(m)} \quad \forall m, k$$

$$\mathbb{E}[\check{\mathbf{y}}_n^{(m)}] = \sum_{k=1}^{K} \mathbb{E}[\check{\mathbf{y}}_n^{(m)} | y_n = k] \Pr(y_n = k) = \mathbf{H}^{(m)} \boldsymbol{\pi} \quad \forall m$$

$$\mathbf{R}_{mm'} := \mathbb{E}[\check{\mathbf{y}}_n^{(m)} \check{\mathbf{y}}_n^{(m')^\top}] = \mathbf{H}^{(m)} \operatorname{diag}(\boldsymbol{\pi}) \mathbf{H}^{(m')^\top} \quad \forall m, m' \neq m$$

$$\underline{\Psi}_{mm'm''} := \mathbb{E}[\check{\mathbf{y}}_n^{(m)} \circ \check{\mathbf{y}}_n^{(m')} \circ \check{\mathbf{y}}_n^{(m'')}] = \sum_{k=1}^{K} \pi_k \boldsymbol{h}_k^{(m)} \circ \boldsymbol{h}_k^{(m')} \circ \boldsymbol{h}_k^{(m'')} = [[\mathbf{H}^{(m)} \operatorname{diag}(\boldsymbol{\pi}), \mathbf{H}^{(m')}, \mathbf{H}^{(m'')}]]_K \quad \forall m$$

PARAFAC/CPD tensor

$\{\mathbf{H}^{(m)}\}_{m=1}^{M}$, $\boldsymbol{\pi}$ Recoverable from annotator responses!

P. A. Traganitis, A. Pages-Zamora, and G. B. Giannakis, "Blind Multiclass Ensemble Classification," *IEEE Transactions on Signal Processing*, vol. 66, no. 18, pp. 4737-4752, September 2018.

# Moment matching

- Find $\{\mathbf{H}^{(m)}\}_{m=1}^{M}, \boldsymbol{\pi}$ s.t. ensemble moments match empirical moments

Empirical averages | Ensemble averages

$$\hat{\boldsymbol{\mu}}_m = \frac{1}{N}\sum_{n=1}^{N} \check{\mathbf{y}}_n^{(m)} \quad \xleftarrow{M} \quad \mathbb{E}[\check{\mathbf{y}}_n^{(m)}] = \mathbf{H}^{(m)}\boldsymbol{\pi}$$

$$\hat{\mathbf{R}}_{mm'} = \frac{1}{N}\sum_{n=1}^{N} \check{\mathbf{y}}_n^{(m)} \check{\mathbf{y}}_n^{(m')\top} \quad \xleftarrow{\binom{M}{2}} \quad \mathbf{R}_{mm'} = \mathbf{H}^{(m)}\mathrm{diag}(\boldsymbol{\pi})\mathbf{H}^{(m')\top}$$

$$\hat{\underline{\Psi}}_{mm'm''} = \frac{1}{N}\sum_{n=1}^{N} \check{\mathbf{y}}_n^{(m)} \circ \check{\mathbf{y}}_n^{(m')} \circ \check{\mathbf{y}}_n^{(m'')} \quad \xleftarrow{\binom{M}{3}} \quad \underline{\Psi}_{mm'm''} = [[\mathbf{H}^{(m)}\mathrm{diag}(\boldsymbol{\pi}), \mathbf{H}^{(m')}, \mathbf{H}^{(m'')}]]_K$$

- Third order moments provide identifiability of $\{\mathbf{H}^{(m)}\}_{m=1}^{M}, \boldsymbol{\pi}$
- At least 3 confusion matrices must be full rank
- Scales to datasets w/ large *N*

**Thm.**: Let $\mathcal{S}^*$ denote the solutions when ensemble statistics are available and $\mathcal{S}^N$ denote the solutions when the statistics are derived from *N* data. Then

$$\mathcal{D}(\mathcal{S}^*, \mathcal{S}^N) \rightarrow 0 \quad \text{as} \quad N \rightarrow \infty \quad \text{almost surely.}$$

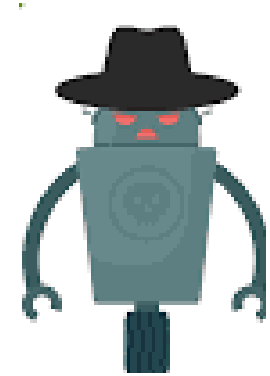
- Avoid third order moments using NMF [Ibrahim et al '19]

# Outline

- Motivation
- Crowdsourcing 101
- Crowdsourcing with spammers
    - Characterizing adversaries under DS model
    - A spectral algorithm for identifying adversaries
- Crowdsourcing with cooperating adversaries
- Conclusion

# Crowdsourcing under attack

- ❑ **Crowdsourcing is susceptible to adversarial attacks**
  - ❑ Adversaries may hide as annotators 😈
  - ❑ Adversaries manipulate results / reduce system performance / drain resources
  - ❑ Attacks in crowdsourcing can poison datasets



**Q:** Which are the worst adversarial attacks?

**Q:** Can we identify these adversarial attacks?

- ❑ **Prior art**
  - ➢ Spammer detection during aggregation [Raykar and Yu ’12] — Modified EM algorithm
  - ➢ Arbitrary adversaries under one-coin model [Jagabathula et al ’17, Kleindessner and Awasthi ’18, Ma and Olshevsky ‘20] — Can handle up to 50% adversaries

- ❑ This talk: Crowdsourcing w/ spammers & colluding adversaries

# Characterizing spammers

- Under the DS model: Annotators $\Rightarrow$ independent information bearing channels

Input | Channel $\mathbf{H}^{(m)}$ | Output

$y_n \rightarrow$ Annotator $m$ $\rightarrow \check{y}_n^{(m)}$

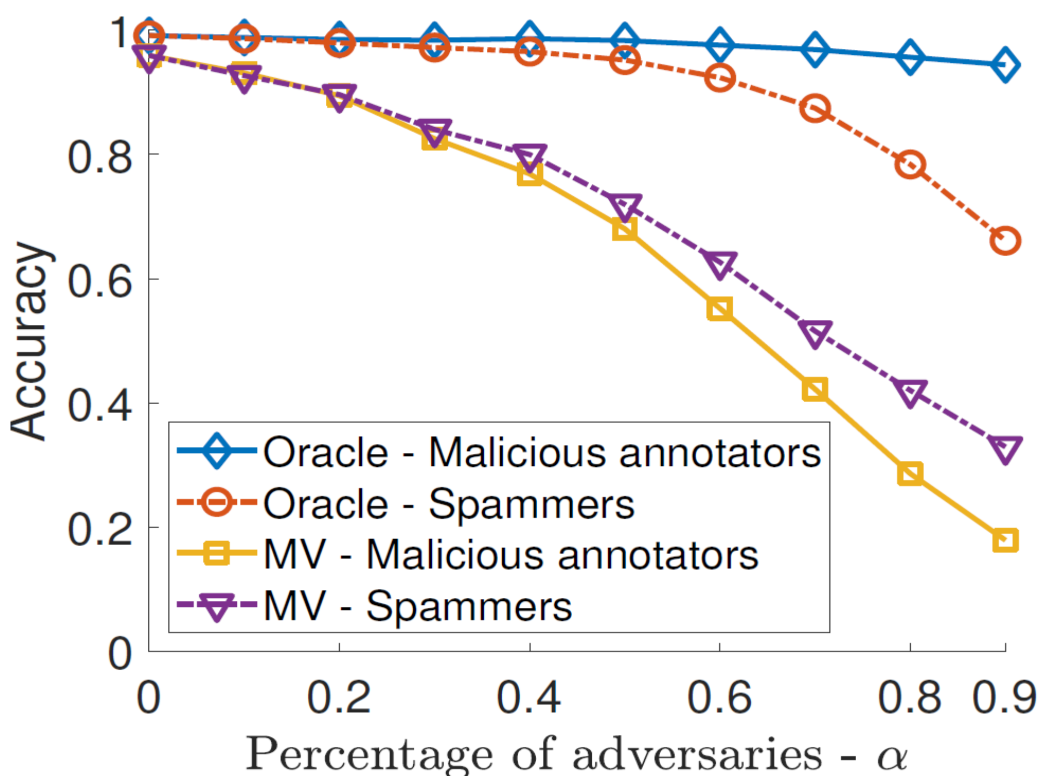
- Annotator performance indicated by channel capacity: $C^{(m)} := \max_{\boldsymbol{\pi}} I(y_n, \check{y}_n^{(m)}) \geq 0$
  - Overall capacity $C = \sum_{m=1}^{M} C^{(m)}$


- Worst annotator behavior: $C^{(m)} = 0$, i.e., output not related to input **Spammers**
  - Spammer confusion matrix $\mathbf{H}^{(m)} = \boldsymbol{s}^{(m)} \mathbf{1}^\top$ $\quad \boldsymbol{s}^{(m)} \geq \mathbf{0}, \mathbf{1}^\top \boldsymbol{s}^{(m)} = 1$

- Two groups of annotators:
  - Spammers $m \in \mathcal{S}$ Should be removed from dataset
  - Honest – follow DS model $m \in \mathcal{H}$

# Numerical test: effect of adversaries

- Synthetic dataset, $N$=10,000, $K$ = 4, $M$ = 20
  - $\alpha M$ annotators generated as adversaries, $(1-\alpha)M$ honest.
  - Malicious annotators provide wrong answer most of the time
  - Oracle classifier – MAP classifier with access to $\{\mathbf{H}^{(m)}\}_{m=1}^{M}, \boldsymbol{\pi}$



# Cross-covariance between annotators

- Cross-covariance $m \in \mathcal{H} \cup \mathcal{S}, m' \in \mathcal{S}$ Can be used to identify spammers

$$\tilde{r}_{m,m'} = \mathbf{E}[\check{y}_n^{(m)} \check{y}_n^{(m')}] - \tilde{\mu}_m \tilde{\mu}_{m'} = \mathbf{E}[\check{y}_n^{(m)}]\mathbf{E}[\check{y}_n^{(m')}] - \tilde{\mu}_m \tilde{\mu}_{m'} = 0$$

- Mean annotator response

$$\tilde{\mu}_m := \mathbf{E}[\check{y}_n^{(m)}] = \sum_{k'=1}^{K} k' \Pr(\check{y}_n^{(m)} = k') = \sum_{k'=1}^{K} k' \sum_{k=1}^{K} \Pr(\check{y}_n^{(m)} = k' | y_n = k) \Pr(y_n = k) = \mathbf{k}^\top \mathbf{H}^{(m)} \boldsymbol{\pi}$$

$$\mathbf{k} = [1, \ldots, K]^\top$$

- Cross-covariance between annotators *m, m'*

$$\mathbf{E}[\check{y}_n^{(m)} \check{y}_n^{(m')}] = \mathbf{k}^\top \mathbf{H}^{(m)} \mathrm{diag}(\boldsymbol{\pi}) \mathbf{H}^{(m')^\top} \mathbf{k}$$

$$\tilde{r}_{m,m'} := \mathbf{E}[(\check{y}_n^{(m)} - \mu_m)(\check{y}_n^{(m')} - \mu_{m'})] = \mathbf{k}^\top \mathbf{H}^{(m)} \mathbf{D}_\pi \mathbf{H}^{(m')^\top} \mathbf{k}$$

$$\mathbf{D}_\pi := \mathrm{diag}(\boldsymbol{\pi}) - \boldsymbol{\pi}\boldsymbol{\pi}^\top, \mathbf{D}_\pi \succeq \mathbf{0}, \mathrm{rank}(\mathbf{D}_\pi) = K - 1, \mathbf{D}_\pi \mathbf{1} = 0$$

- Structure of cross-covariance matrix

$$\left.\begin{array}{l} \tilde{r}_{m,m'} = \mathbf{k}^\top \mathbf{H}^{(m)} \mathbf{D}_\pi^{1/2} \mathbf{D}_\pi^{1/2} \mathbf{H}^{(m')^\top} \mathbf{k} = \mathbf{v}_m^\top \mathbf{v}_{m'} \\ \mathbf{v}_m := \mathbf{D}_\pi^{1/2} \mathbf{H}^{(m')^\top} \mathbf{k} \qquad \mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_M]^\top \end{array}\right\} \tilde{\mathbf{R}} = \mathbf{V}\mathbf{V}^\top + \mathbf{D}$$

  - Spammers: $\mathbf{v}_m \approx \mathbf{0}, m \in \mathcal{S}$

# An algorithm for identifying spammers

❖ **S1:** "Denoise" cross-covariance matrix

$$\{\hat{\mathbf{L}}, \hat{\mathbf{D}}\} = \arg\min_{\mathbf{L},\mathbf{D}} \|\mathbf{L}\|_* \quad \text{s. to } \boldsymbol{\Omega} * \hat{\mathbf{R}} = \boldsymbol{\Omega} * (\mathbf{D} + \mathbf{L}), \quad \mathbf{L} = \mathbf{L}^\top$$

Convex

$$\hat{\mu}_m = \frac{1}{N} \sum_{n=1}^{N} \check{y}_n^{(m)}$$

$$[\hat{\mathbf{R}}]_{m,m'} := \hat{r}_{m,m'} = \frac{1}{N-1} \sum_{n=1}^{N} (\check{y}_n^{(m)} - \hat{\mu}_m)(\check{y}_n^{(m')} - \hat{\mu}_{m'})$$

❖ **S2:** Recover $\mathbf{V}$ from the truncated SVD of $\hat{\mathbf{L}} = \mathbf{U}_{K-1} \boldsymbol{\Sigma}_{K-1} \mathbf{U}_{K-1}^\top$

$$\hat{\mathbf{V}} = \mathbf{U}_{K-1} \boldsymbol{\Sigma}_{K-1}^{1/2}$$

❖ **S3:** Cluster rows of $\hat{\mathbf{V}} = [\hat{\mathbf{v}}_1, \ldots, \hat{\mathbf{v}}_M]^\top$ in 2 clusters

- Using e.g. *K*-means
- Cluster indices collected in $\mathcal{C}_1, \mathcal{C}_2 \subseteq \{1, \ldots, M\}$

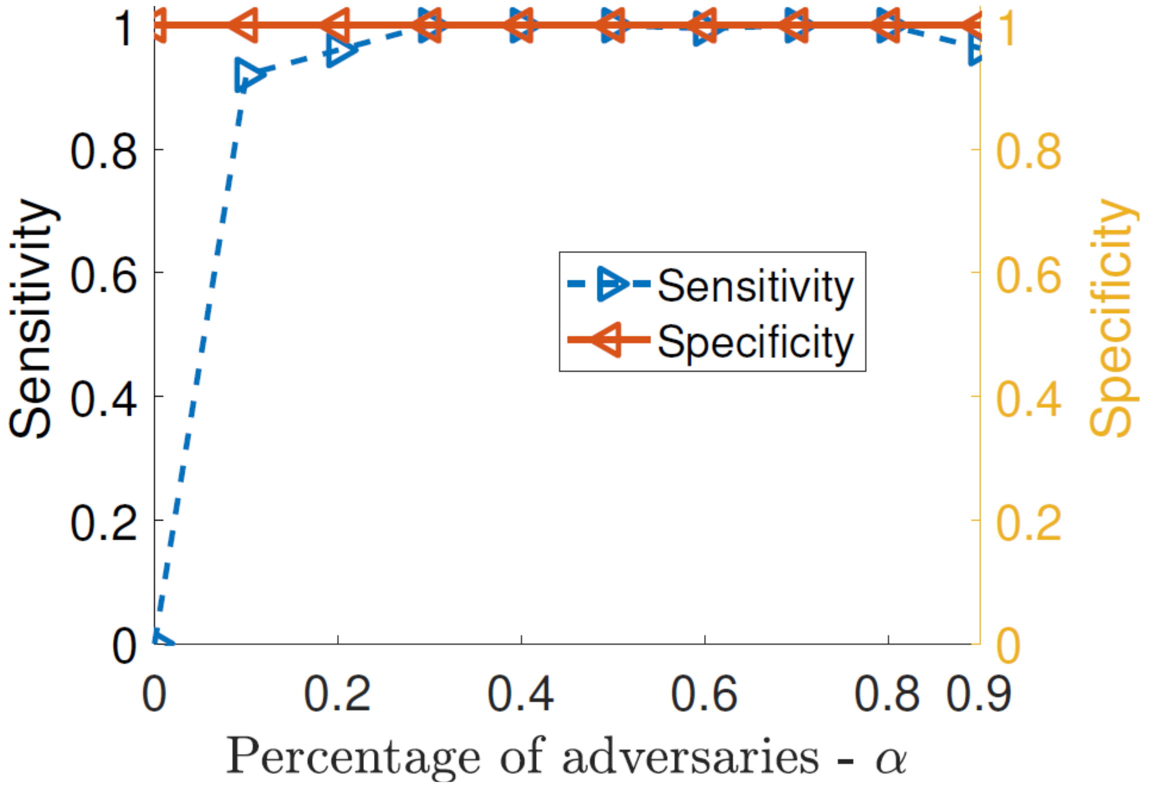
❖ **S4:** Identify spammer cluster $\hat{\mathcal{S}}$ as

$$\hat{\mathcal{S}} = \mathcal{C}_\ell \qquad \ell = \arg\min_i \frac{1}{|\mathcal{C}_i|} \sum_{m \in \mathcal{C}_i} \|\hat{\mathbf{v}}_m\|_2^2$$

P. A. Traganitis and G. B. Giannakis, "Identifying spammers to boost crowdsourced classification," in 46th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2021.

# Spammer detection performance

- Same synthetic dataset, *N=10,000*, *K = 4, M* = 20
  - Spammer detection evaluated using Sensitivity (true positive rate) and Specificity (true negative rate)



# Numerical tests: real crowdsourcing data

- Proposed algorithm (***Alg. 1***) tested on 3 crowdsourcing datasets
  - Annotators deemed spammers were removed from dataset

- Bluebird dataset *N=108, K = 2, M=39*
- Dog dataset *N=807, K = 4, M=109*
- Web dataset *N=2,655, K = 5, M=177*

Classification accuracy

| Dataset | MV | DS | Alg. 1 + MV | Alg. 1 + DS |
|---|---|---|---|---|
| Bluebird | 0.759 | 0.88 | 0.852(**22**) | 0.899(**22**) |
| Dog | 0.817 | 0.834 | 0.819(**12**) | 0.834(**12**) |
| Web | 0.776 | 0.871 | 0.841(**158**) | 0.91(**158**) |

Parentheses indicate number of pruned annotators

P. Welinder et al, "The multidimensional wisdom of crowds," in Advances in Neural Information Processing Systems 23, 2010, pp. 2424–2432.
J. Deng et al, "Imagenet: A large-scale hierarchical image database," in Proc. of IEEE Conf. on Computer Vision and Pattern Recognition, 2009, pp. 248–255.
D. Zhou et al, "Learning from the wisdom of crowds by minimax entropy," in Advances in Neural Information Processing Systems 25, 2012, pp. 2195–2203.

# Outline

- Motivation
- Crowdsourcing 101
- Crowdsourcing with spammers
- Crowdsourcing with cooperating adversaries
  - Properties of the inter-annotator agreement matrix
  - Yet another spectral algorithm
- Conclusion

# Cooperating / Colluding adversaries

- What if adversaries are allowed to cooperate?
  - Model misspecification – DS model no longer applicable!

- Two groups of annotators:
  - Adversaries – deviate from DS model $m \in \mathcal{A}$
  - Honest – follow DS model $m \in \mathcal{H}$

**(As3):** Adversaries are conditionally independent from honest workers

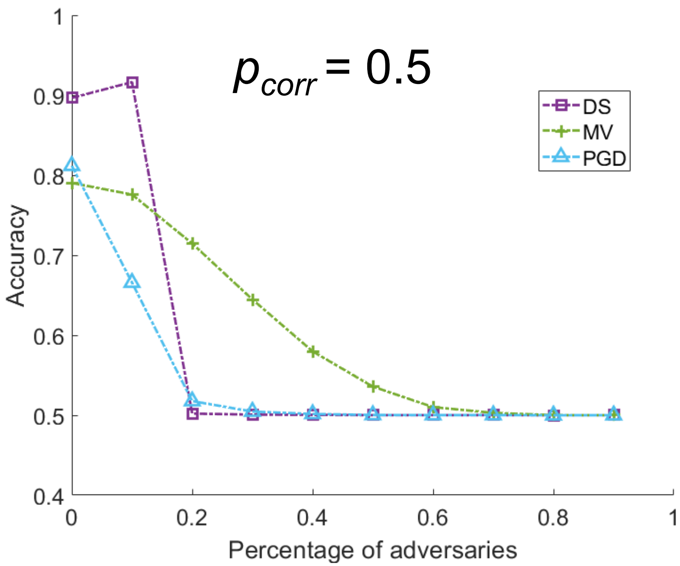
  - Adversaries don't have access to honest annotator responses, only the data

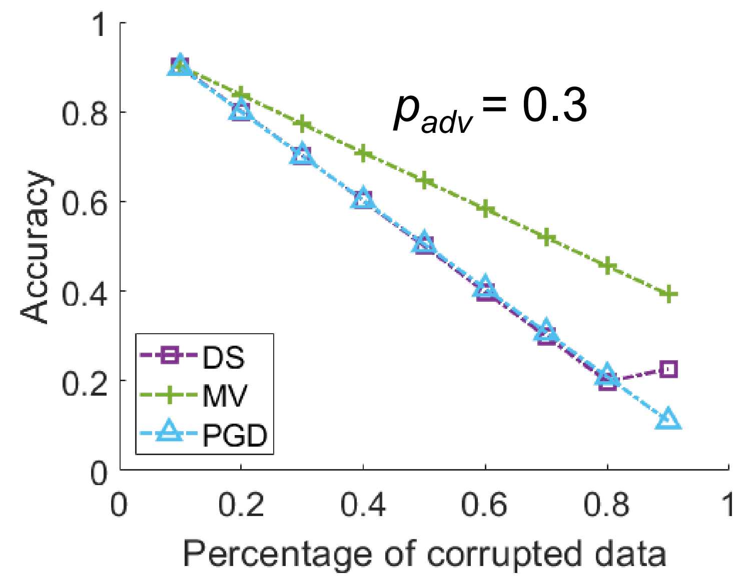
$$p_{\mathcal{A}} := \prod_{n=1}^{N} \Pr\left( \{\check{y}_n^{(m)} = k_{m,n}\}_{m \in \mathcal{A}} \middle| \{y_{n'} = k_{n'}\}_{n'=1}^{N} \right)$$

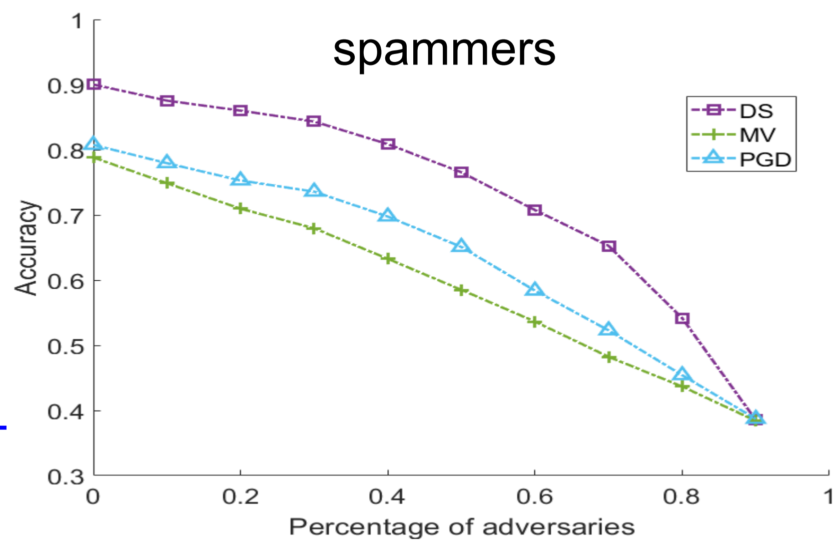
- Additional side information required:
  - 50% of annotators are honest
  - Knowledge of (at least) one trusted annotator

# Numerical test: Effect of colluding adversaries

- Synthetic dataset: $N$ = 5,000, $M$=60, $K$=3. Probability an annotator is adversarial = $p_{adv}$
- Adversaries provide wrong response w.p. $p_{corr}$, and ground-truth label for remaining data.



# Annotator agreement matrix – Honest annotators

- Recall *KxK* co-occurrence matrix for annotators *m,m'*:

$$\mathbf{R}_{m,m'} := \mathbf{E}[\mathbf{y}_n^{(m)} \mathbf{y}_n^{(m')^\top}] = \mathbf{H}^{(m)} \mathrm{diag}(\pi) \mathbf{H}^{(m')^\top} =$$

$$\begin{bmatrix} \Pr(\check{y}_n^{(m)} = 1, \check{y}_n^{(m')} = 1) & \Pr(\check{y}_n^{(m)} = 1, \check{y}_n^{(m')} = 2) & \dots & \Pr(\check{y}_n^{(m)} = 1, \check{y}_n^{(m')} = K) \\ \Pr(\check{y}_n^{(m)} = 2, \check{y}_n^{(m')} = 1) & \Pr(\check{y}_n^{(m)} = 2, \check{y}_n^{(m')} = 2) & & \vdots \\ \vdots & & \ddots & \vdots \\ \Pr(\check{y}_n^{(m)} = K, \check{y}_n^{(m')} = 1) & \dots & \dots & \Pr(\check{y}_n^{(m)} = K, \check{y}_n^{(m')} = K) \end{bmatrix}$$

- Annotator agreement rate: $\sigma_{m,m'} := \mathrm{trace}\left(\mathbf{R}_{m,m'}\right) = \mathrm{trace}\left(\mathbf{H}^{(m)} \mathrm{diag}(\boldsymbol{\pi}) \mathbf{H}^{(m')^\top}\right)$

$$\mathrm{tr}(\mathbf{H}^{(m)} \mathrm{diag}(\boldsymbol{\pi})^{1/2} \mathrm{diag}(\boldsymbol{\pi})^{1/2} \mathbf{H}^{(m')^\top}) = \mathrm{vec}(\mathrm{diag}(\boldsymbol{\pi})^{1/2} \mathbf{H}^{(m)^\top})^\top \mathrm{vec}(\mathrm{diag}(\boldsymbol{\pi})^{1/2} \mathbf{H}^{(m')^\top})$$

$$\boldsymbol{u}^{(m)} := \mathrm{vec}(\mathrm{diag}(\boldsymbol{\pi})^{1/2} \mathbf{H}^{(m)'^\top}) : K^2 \times 1$$

- Agreement matrix: $\boldsymbol{\Sigma}_{\mathcal{H}} = \mathbf{C}_{\mathcal{H}} + \mathbf{I}_{\mathcal{H}} = \mathbf{U}\mathbf{U}^\top + \mathbf{I}_{\mathcal{H}}$ **Low rank + Diagonal**

$$\mathbf{U} := [\boldsymbol{u}^{(1)}, \dots, \boldsymbol{u}^{(M_{\mathcal{H}})}]^\top$$

**(As4):** $M_{\mathcal{H}} > K^2$

# Agreement between honest and adversarial annotators

- Consider $m \in \mathcal{H}, \quad m' \in \mathcal{A}$

$$\mathbf{R}_{m,m'} := \mathbf{E}[\mathbf{y}_n^{(m)} \mathbf{y}_n^{(m')^\top}] = \mathbf{H}^{(m)} \mathrm{diag}(\pi) \mathbf{G}^{(m')^\top}$$

$$[\mathbf{G}^{(m)}]_{k,c_n} = \sum_{\mathbf{c}_{-n}} \Pr(\check{y}_n^{(m')} = k | \mathbf{y} = \mathbf{c}) \prod_{j \neq n} \Pr(y_j = c_j)$$

- Annotator agreement rate:

$$\sigma_{m,m'} := \mathrm{trace}\left(\mathbf{R}_{m,m'}\right) = \mathrm{trace}\left(\mathbf{H}^{(m)} \mathrm{diag}(\boldsymbol{\pi}) \mathbf{G}^{(m')^\top}\right) = \boldsymbol{u}^{(m)^\top} \tilde{\boldsymbol{u}}^{(m')}$$

$$\tilde{\boldsymbol{u}}^{(m)} := \mathrm{vec}(\mathrm{diag}(\boldsymbol{\pi})^{1/2} \mathbf{G}^{(m)'^\top}) : K^2 \times 1$$

- Inter-group agreement matrix: $\mathbf{C}_{\mathcal{H},\mathcal{A}} = \mathbf{C}_{\mathcal{A},\mathcal{H}}^\top = \mathbf{U}\tilde{\mathbf{U}}^\top : M_{\mathcal{H}} \times M_{\mathcal{A}}$

- Overall agreement matrix: $\boldsymbol{\Sigma} = \mathbf{C} + \mathbf{I} = \left[\begin{array}{c|c} \mathbf{C}_{\mathcal{H}} & \mathbf{C}_{\mathcal{H},\mathcal{A}} \\ \hline \underbrace{\mathbf{C}_{\mathcal{A},\mathcal{H}}}_{rank \leq K^2} & \mathbf{C}_{\mathcal{A}} \end{array}\right] + \begin{bmatrix} \mathbf{I}_{\mathcal{H}} & \\ & \mathbf{I}_{\mathcal{A}} \end{bmatrix}$

unknown

# A spectral algorithm for identifying cooperating adversaries

- **S1:** Estimate **C** from **Σ** RPCA/Robust Matrix Completion

$$\{\hat{\mathbf{C}}, \hat{\mathbf{S}}\} = \arg\min_{\mathbf{C},\mathbf{S}} \|\mathbf{C}\|_* + \lambda \|\mathrm{vec}(\mathbf{S})\|_1$$
$$\text{subject to } \boldsymbol{\Omega} * \hat{\boldsymbol{\Sigma}} = \boldsymbol{\Omega} * (\mathbf{C} + \mathbf{S})$$

$$\lambda = \frac{1}{\sqrt{\mathrm{nnz}(\boldsymbol{\Omega})/M}}$$

- **S2:** Cluster rows/columns of **C** Subspace clustering
  - Solve $\min_{\mathbf{Z}} \|\hat{\mathbf{C}} - \hat{\mathbf{C}}\mathbf{Z}\|_F^2 + \rho r(\mathbf{Z})$
  - Apply Spectral Clustering to $|\mathbf{Z}| + |\mathbf{Z}^\top|$, obtain two clusters of annotators $\mathcal{C}_1, \mathcal{C}_2$

Elementwise absolute value

- **S3:** Using side-information decide $\hat{\mathcal{H}}, \hat{\mathcal{A}} := \{1, \ldots, M\}/\hat{\mathcal{H}}$
  - Honest annotators > 50% : $\hat{\mathcal{H}} = \arg\max_i \mathrm{cardinality}(\mathcal{C}_i)$
  - Knowledge of one trusted annotator $m_T$ : $\hat{\mathcal{H}} = \begin{cases} \mathcal{C}_1 & \text{if } m_T \in \mathcal{C}_1 \\ \mathcal{C}_2 & \text{if } m_T \in \mathcal{C}_2 \end{cases}$

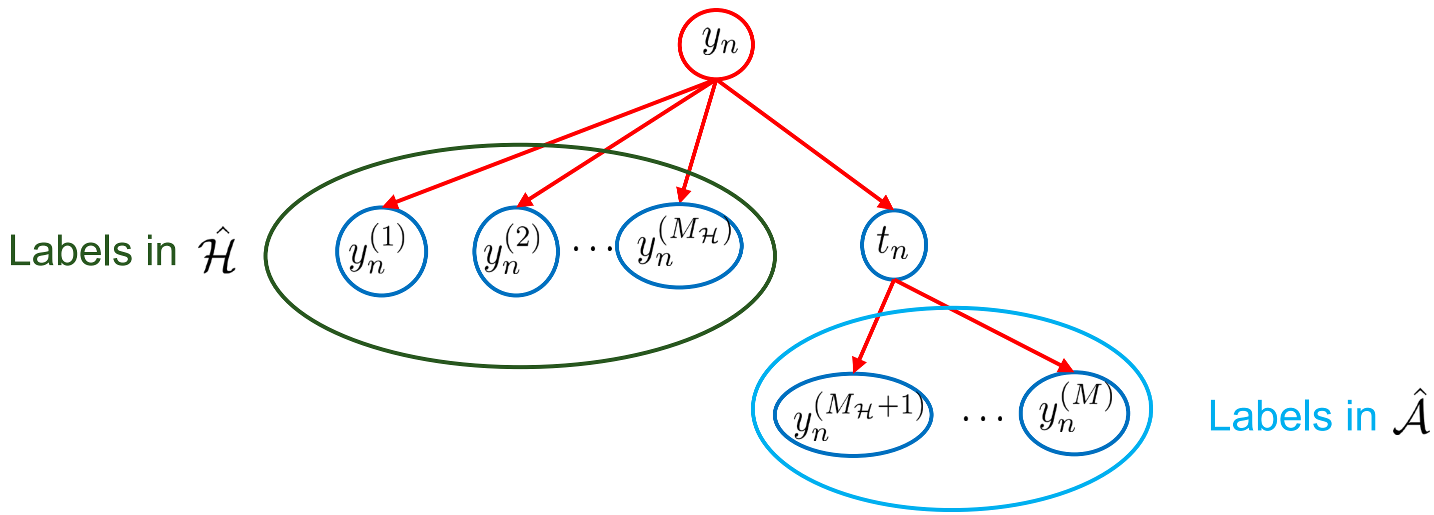
P. A. Traganitis and G. B. Giannakis, "Detecting Adversaries in Crowdsourcing," in 21st IEEE International Conference on Data Mining (ICDM), 2021.

# Aggregating labels in the presence of adversaries

- **Q:** How to fuse $\check{\mathbf{Y}}$ w/ $\hat{\mathcal{H}}, \hat{\mathcal{A}}$ available?

- **A1:** Prune annotators in $\hat{\mathcal{A}}$
  - Treats adversaries as spammers
  - Useful information may be lost

- **A2:** Aggregate labels in $\hat{\mathcal{A}}$ - fuse result with labels in $\hat{\mathcal{H}}$

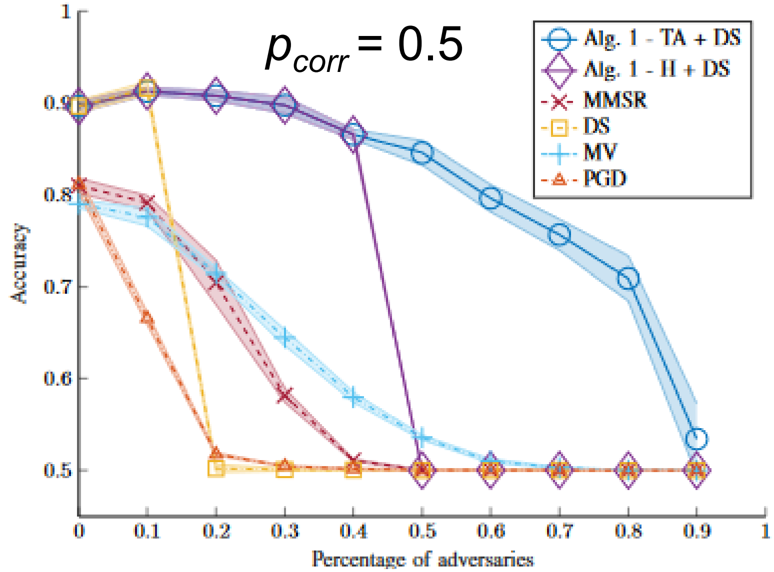


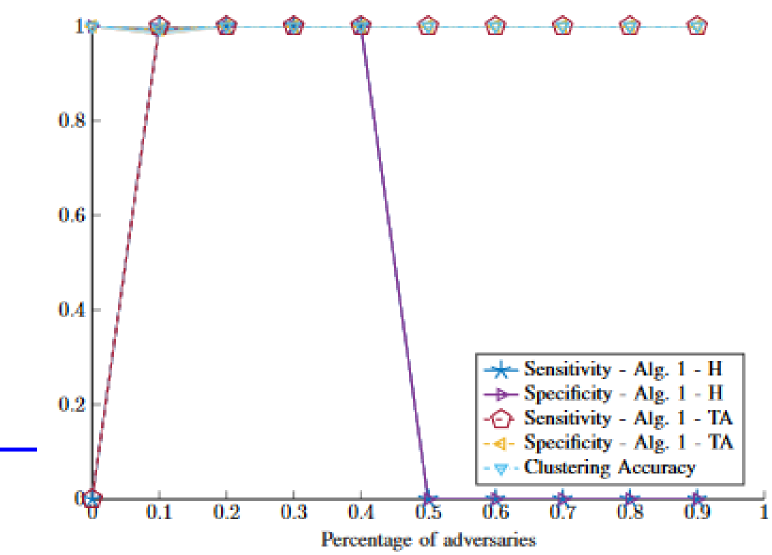


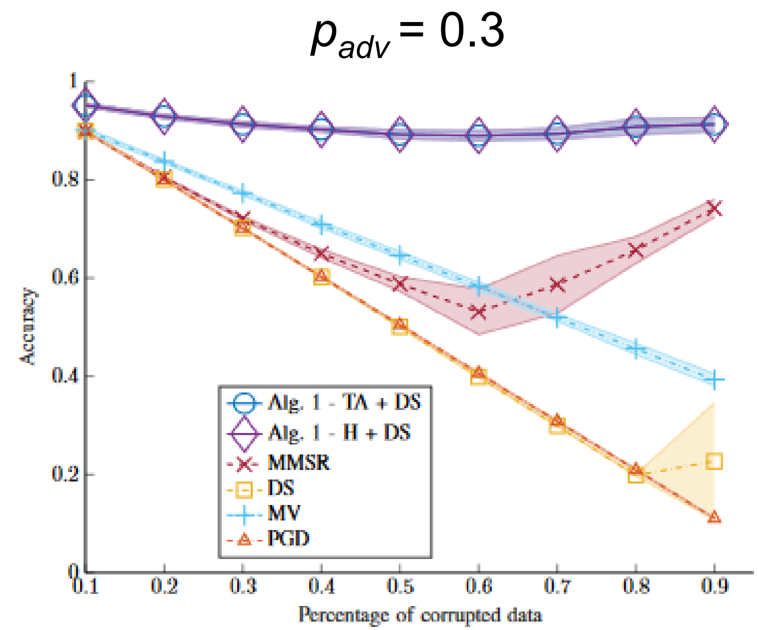
P. A. Traganitis and G. B. Giannakis, "Blind multi-class Ensemble Learning with Dependent Classifiers," Proc. of EUSIPCO, Rome, Italy, Sep 3-7, 2018.
A. Jaffe, E. Fetaya, B. Nadler, T. Jiang, and Y. Kluger, "Unsupervised ensemble learning with dependent classifiers," in Artificial Intelligence and Statistics, 2016, pp. 351–360.

# Numerical tests: Synthetic data

- Synthetic dataset: $N = 5{,}000$, $M=60$, $K=3$

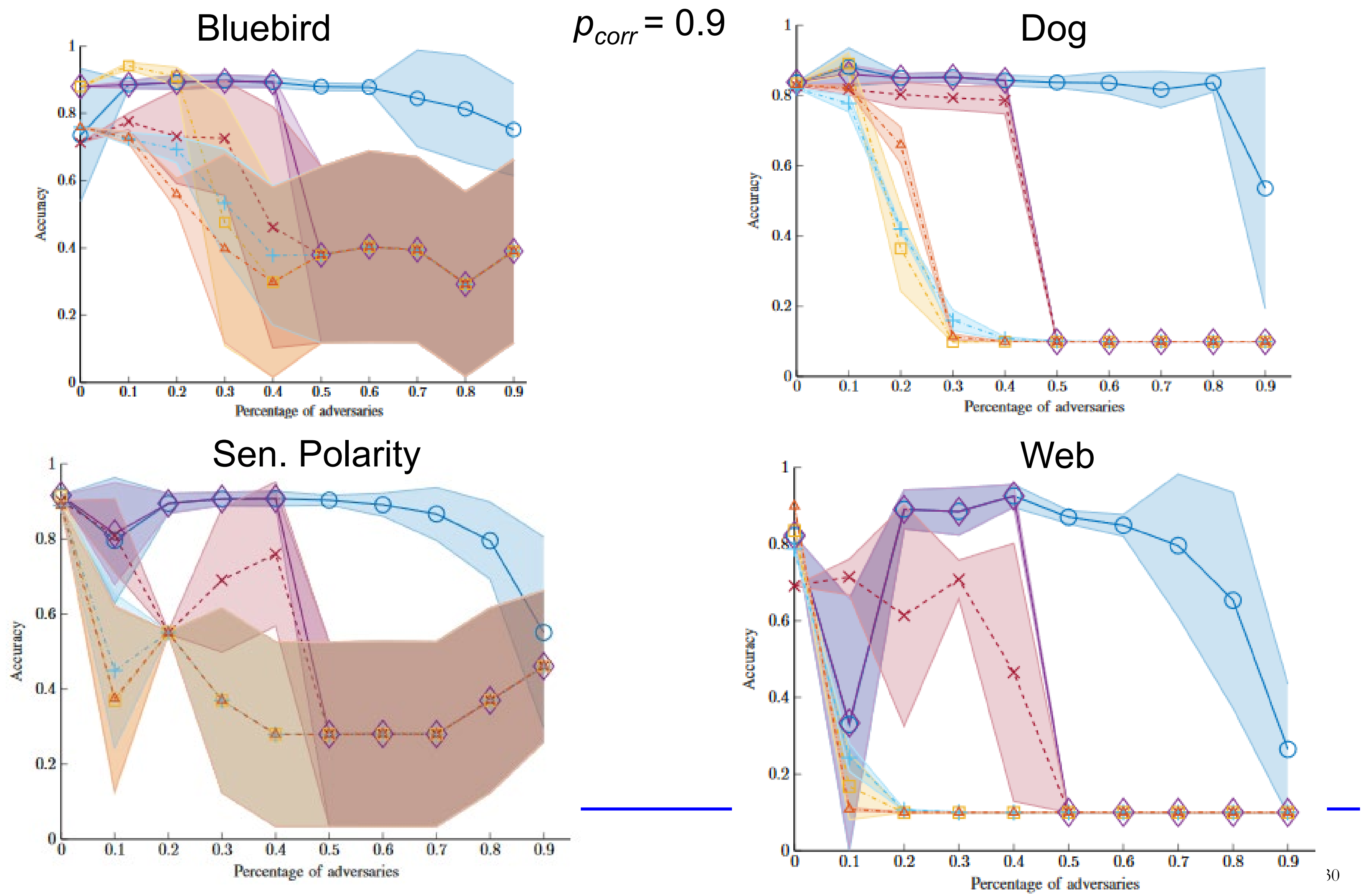


# Numerical tests: Real data

| Dataset | Bluebird | Sentence Polarity | Dog | Web |
|---|---|---|---|---|
| $N$ | 108 | 5,000 | 807 | 2,665 |
| $M$ | 39 | 203 | 109 | 177 |
| $K$ | 2 | 2 | 5 | 5 |
| $\delta$ | 108 | 136.68 | 74.03 | 87.94 |

# What we did not cover

- Regression, clustering
  - Regression [Raykar et al ’10, Zhou et al ’15, Ok et al ‘19]
  - Clustering [Gomes et al ’10, Yi et al ’12, Chen et al ’18]
- Dependent annotators
  - Annotator groups [Venanzi et al ’14, Jaffe et al ’16, Traganitis and Giannakis ’18, Imamura et al ‘18]
- Non-i.i.d. data
  - Sequential data [Nguyen et al ’17, Traganitis and Giannakis ’20, Lu and Chow ’21, Simpson et al ’19, Sabetpour et al ‘21]
  - Networked data [Traganitis and Giannakis ‘20]
- Semi-supervised / Constrained Crowdsourcing
  - Label propagation [Yan et al ’10]
  - Label constraints [Tang and Lease ’11, Liu et al ‘17]
  - Pairwise constraints [Traganitis and Giannakis ‘21]
- Parametric and Neural Network approaches
  - Logistic regression [Raykar et al ‘10], Gaussian Processes [Rodrigues et al ‘14]
  - Deep learning [Shaham et al ’16, Rodrigues and Pereira ’18, Shi et al ‘20]
  - Autoencoders [Yin et al ’17]

# Conclusions

- **Take home:** Crowdsourcing can combine labels from multiple annotators
  - Harnesses wisdom of crowds
  - Workhorse under DS model: EM algorithm
  - Moment based methods can initialize EM

- Crowdsourcing is vulnerable to adversarial attacks
  - Spectral methods can uncover adversaries
  - Structure of (modified) cross-covariance matrix reveals spammers
  - Structure of agreement matrix can reveal colluding adversaries
  - Proposed algorithms can detect large number of adversaries

# Open Issues - Future directions

- Crowdsourcing
  - Constraints for regression/clustering
  - Alternative constraints (Triplet, label proportion etc.)
  - Uncertain annotations
  - Alternative annotations (pairwise, triplet, label proportions, multiple instance etc.)
  - Connections w/ Meta-learning & Weak supervision

- Crowdsourcing with adversaries
  - Can we relax **As4**? (Probably yes)
  - Advanced adversaries
  - Do constraints help us identify adversaries?
  - Optimal label fusion under adversarial attacks?
  - Theoretical analysis
  - Robust EM

***Thank you!***