Crowdsourcing via Pairwise Co-occurrences: Identifiability and Algorithms

Shahana Ibrahim
Joint Work with Xiao Fu, Nikos Kargas, Kejun Huang

School of Electrical Engineering and Computer Science,
Oregon State University

Outline

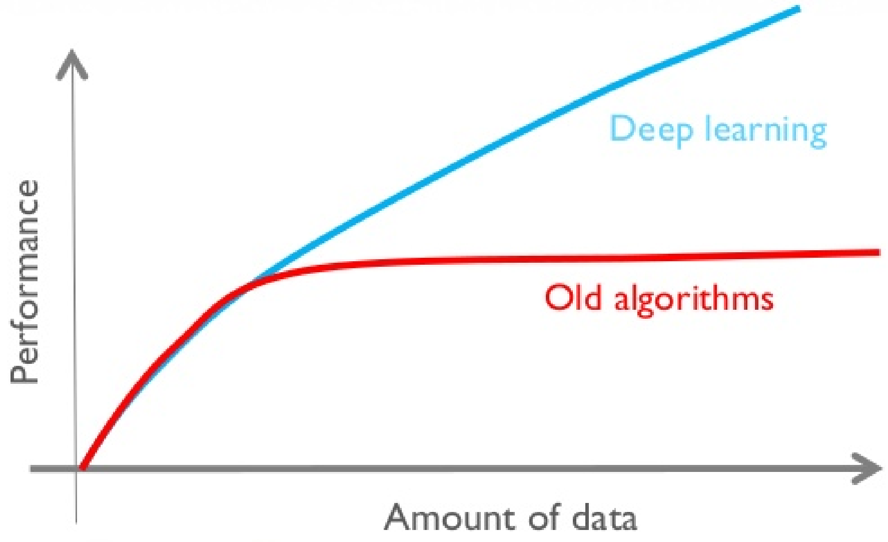
- What is Crowdsourcing?
- Problem modeling.
- Existing approaches.
- Proposed method and its implications.
- Experimental results.
- Conclusion.

The era of big data...

- Tremendous amount of data is being generated every day.
- Many supervised learning tasks, e.g. tasks in computer vision, natural language processing, speech processing heavily rely on labeled data.
- The volume of labeled data in deep learning datasets has grown to millions (e.g. ImageNet, MS.COCO).

The era of big data...

- One of the key performance boosters of the deep learning algorithms is labeled data.

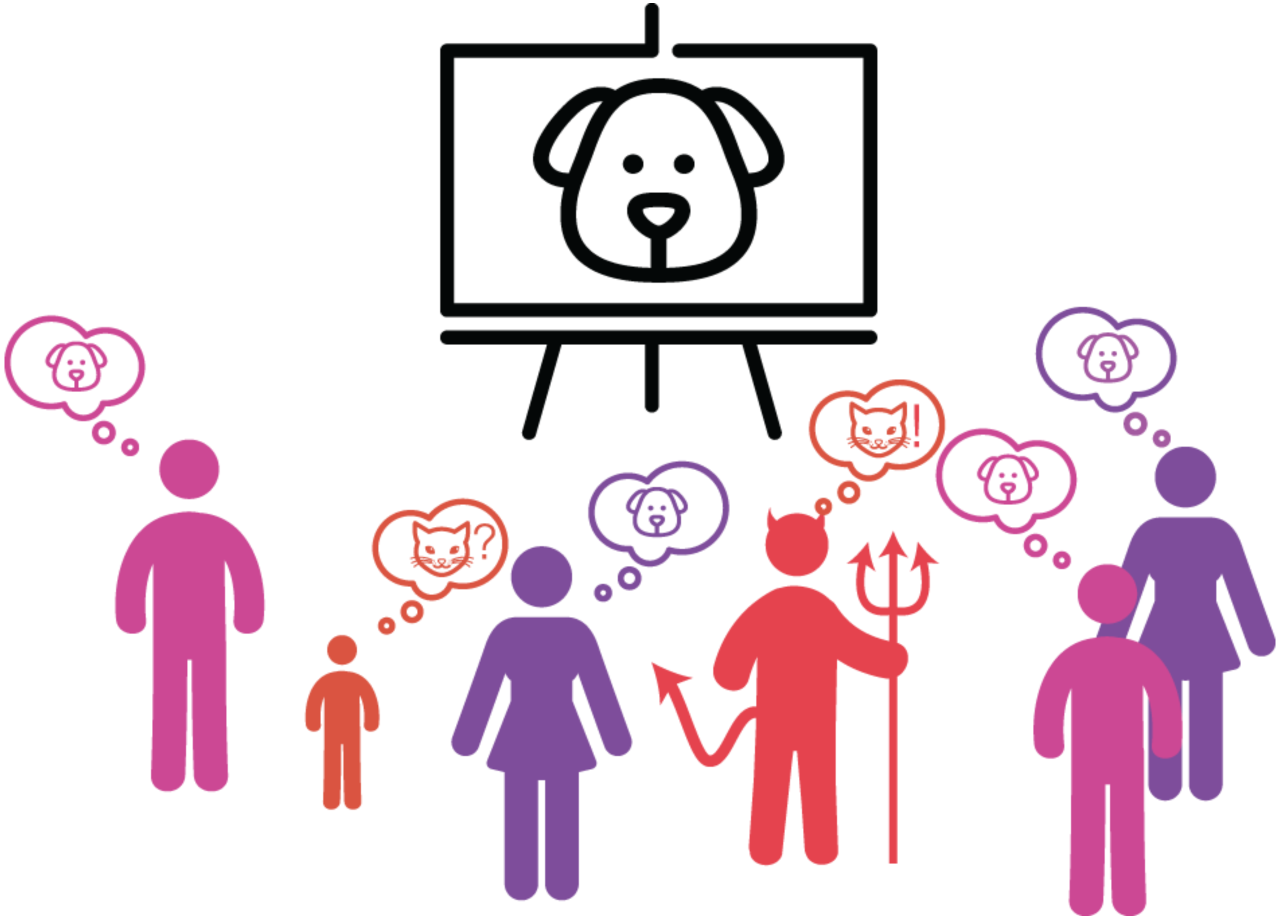


[1]

- **But labeling is not a trivial task!!**

[1]Source : https://www.normshield.com/machine-learning-in-cyber-security-domain-1-fundamentals/

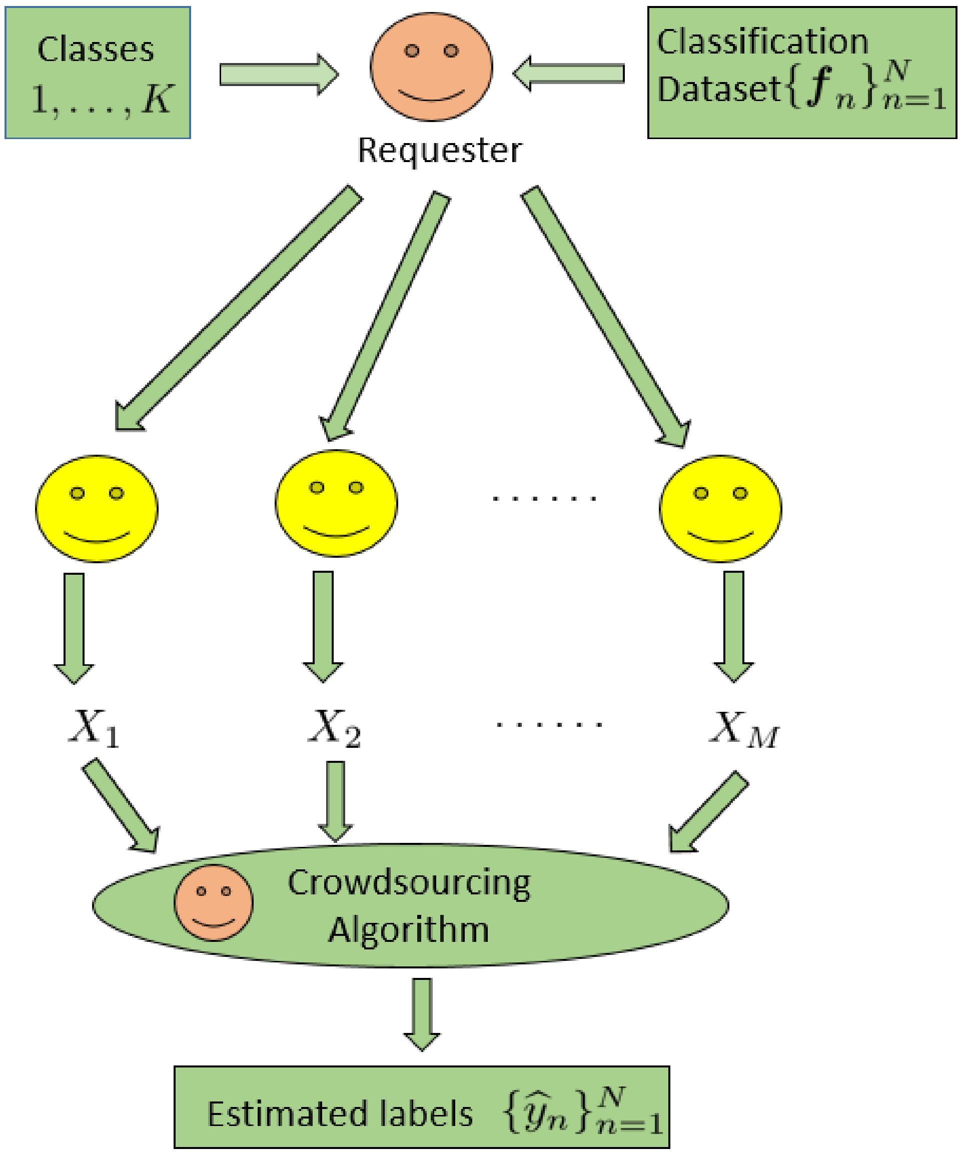
Crowdsourcing Paradigm



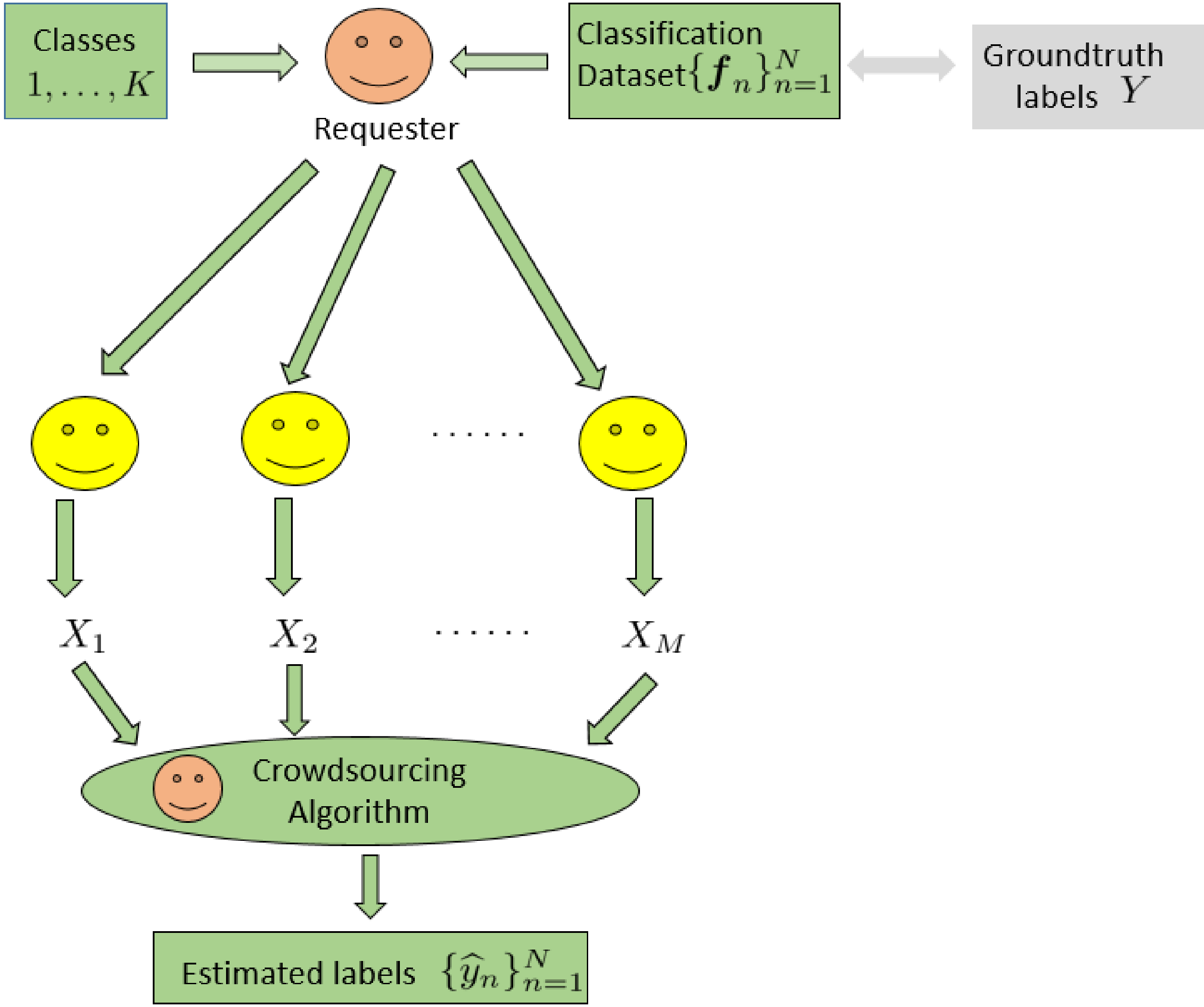
2

2 Source : http://wires.wiley.com/WileyCDA/WiresArticle/wisId-WIDM1288.html

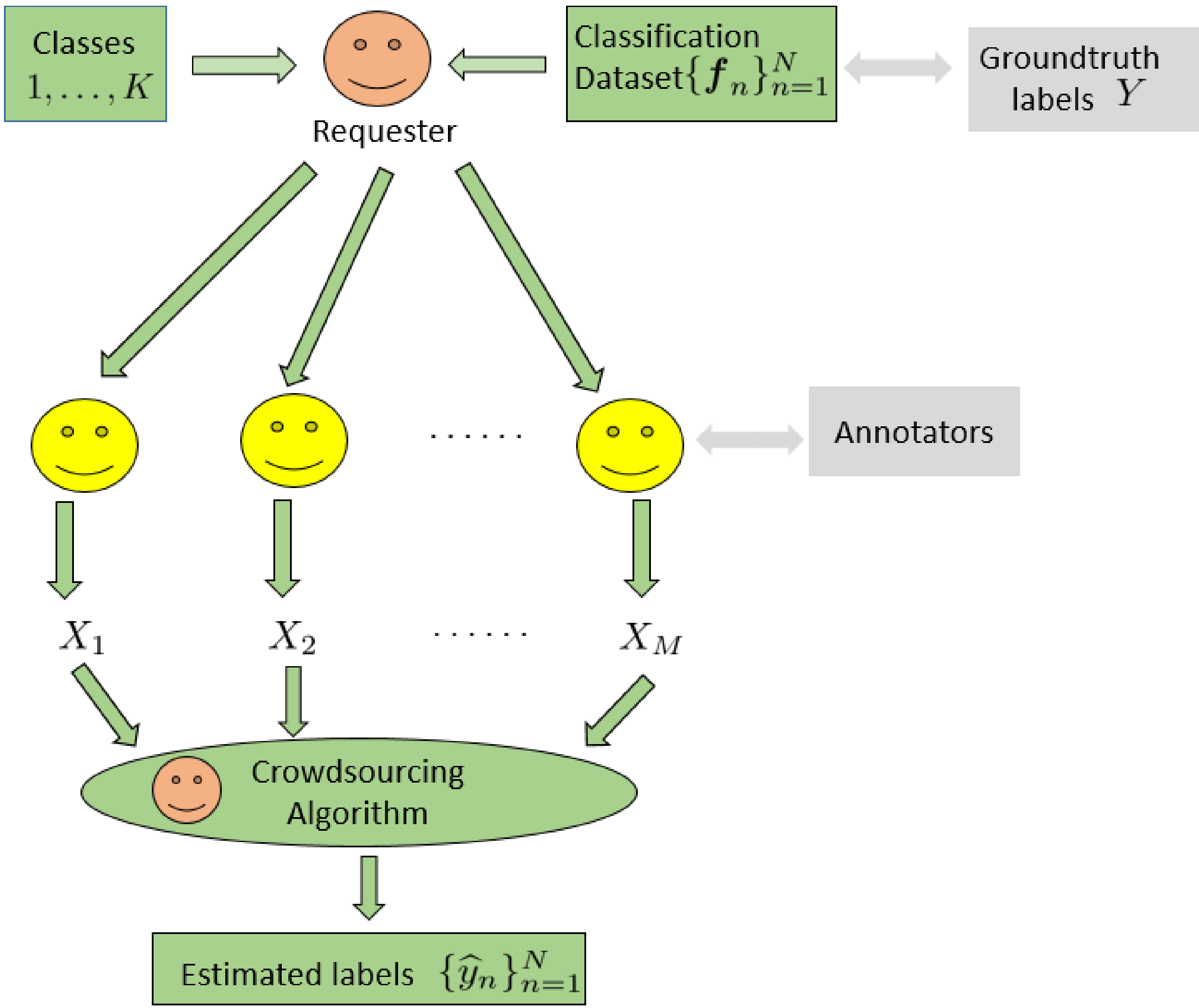
Crowdsourcing Dataflow



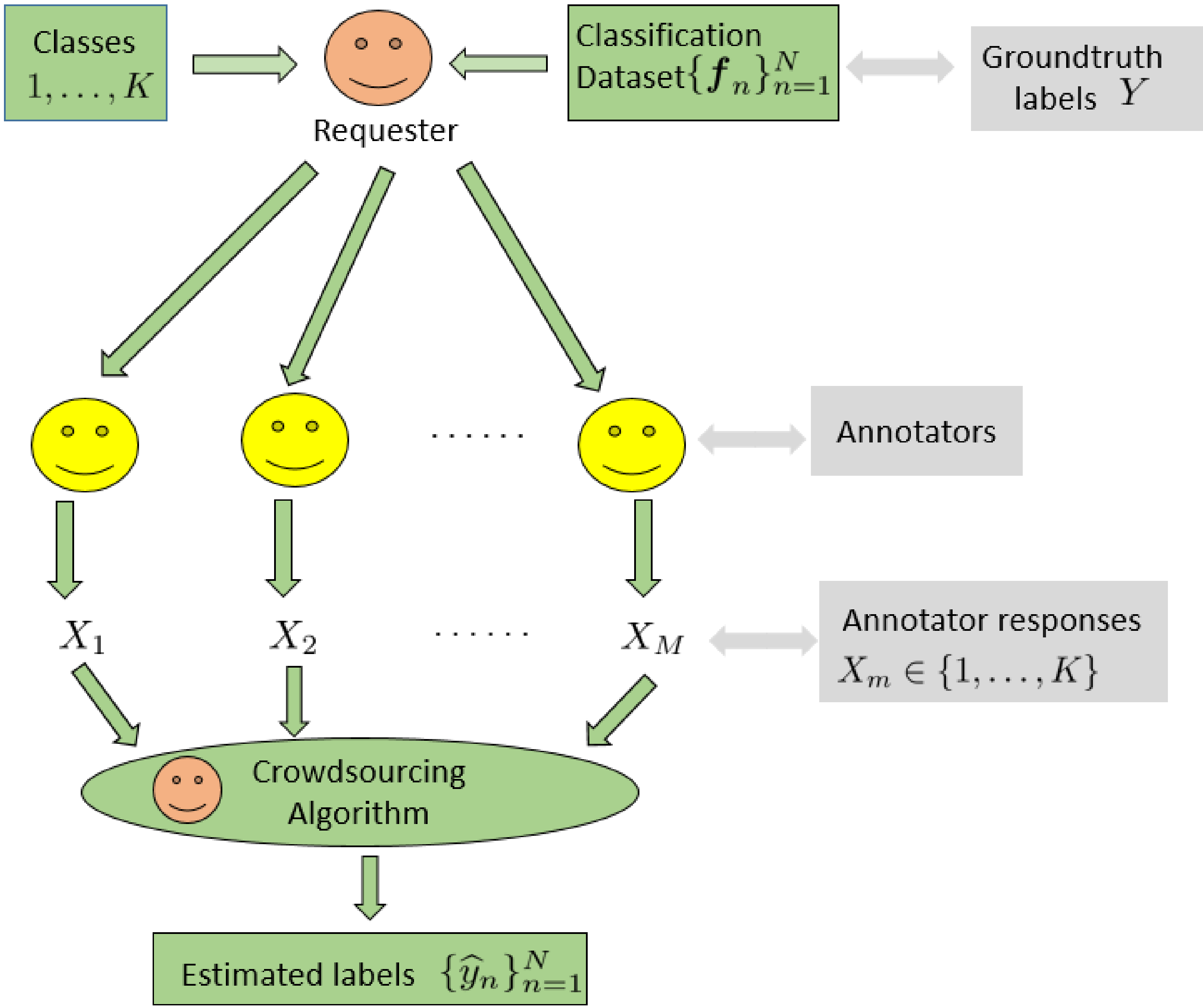
Crowdsourcing Dataflow



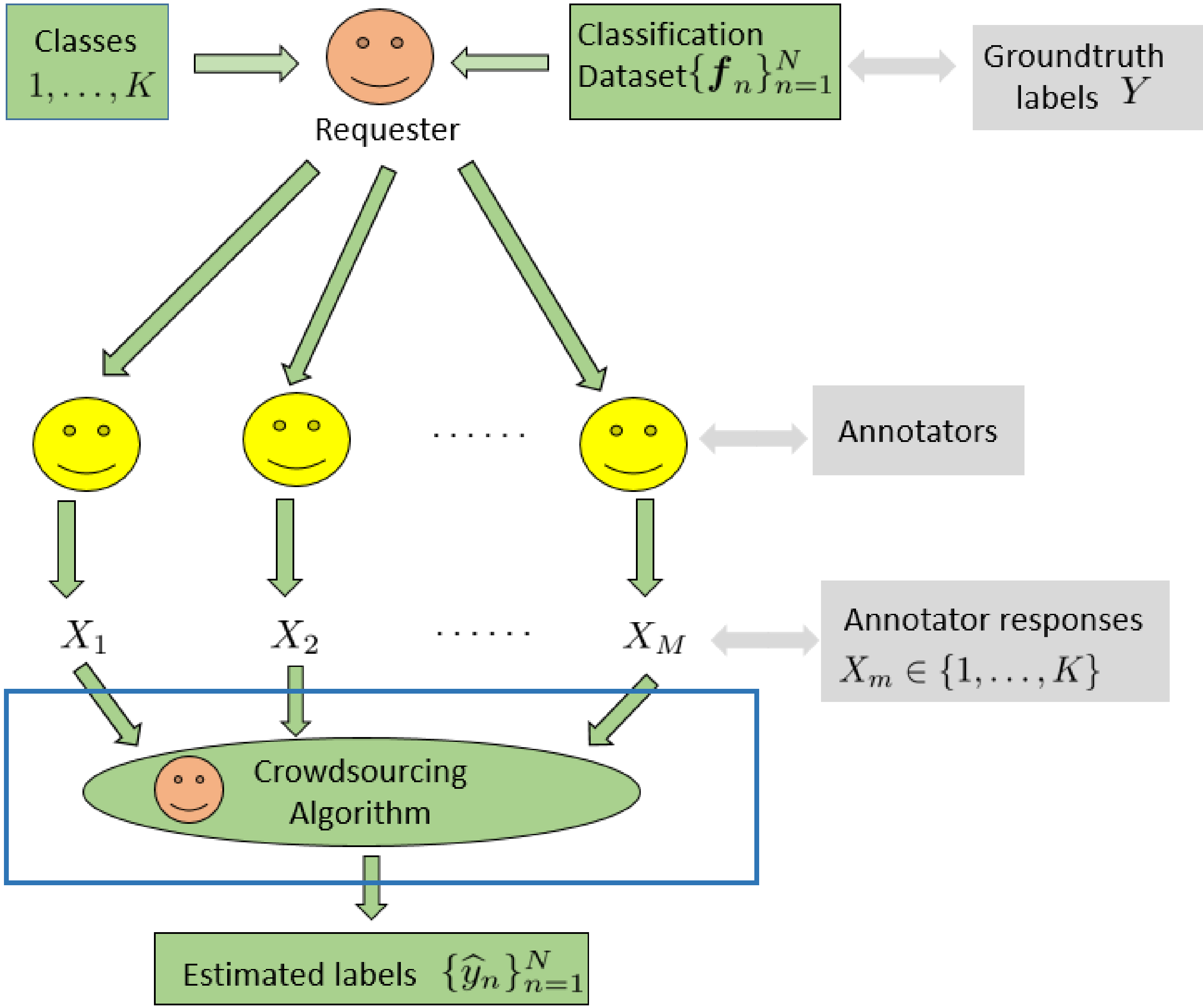
Crowdsourcing Dataflow



Crowdsourcing Dataflow



Crowdsourcing Dataflow



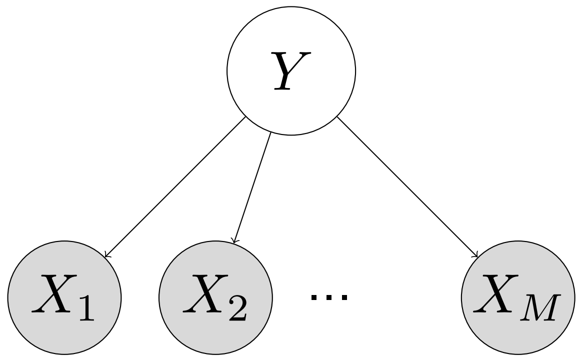
What are the challenges?

- A natural thought for a crowdsourcing algorithm is *majority voting*.
- *Majority voting* may not be always effective.
  - Not all annotators are equally reliable.
  - Each annotator may not be labeling all the data due to limited pay, time or lack of knowledge.

We need effective integrating algorithms for annotator responses.

Dawid-Skene Model in Crowdsourcing Problem

- One of the simplest model in crowdsourcing, but elegent and very effective.
- Crowdsourcing problem was associated to the Naive Bayes Model by [Dawid and Skene, 1979].



Dawid-Skene Model in Crowdsourcing Problem

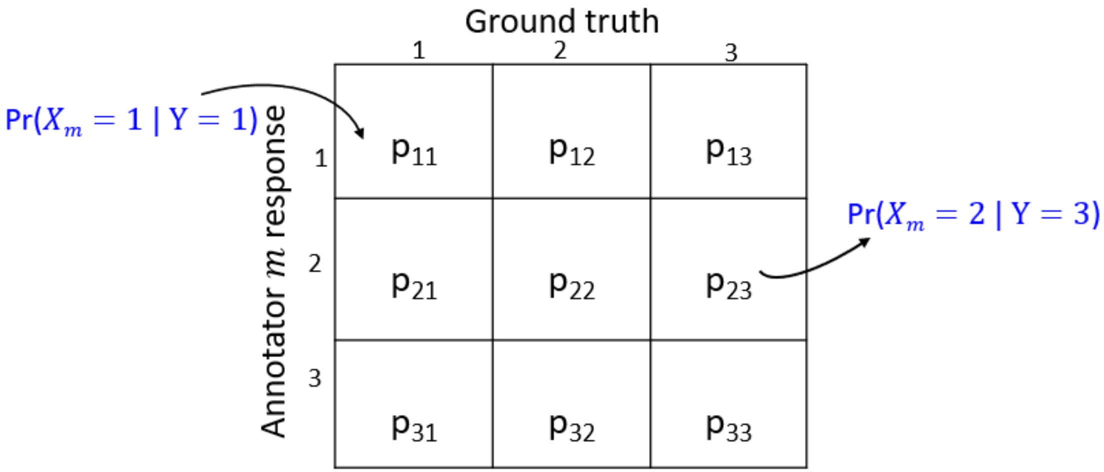
- Under Naive Bayes Model, the joint probability of annotator responses is given by,

$$\mathsf{Pr}(X_1 = k_1, \ldots, X_M = k_M) = \sum_{k=1}^{K} \mathsf{Pr}(Y = k) \prod_{m=1}^{M} \mathsf{Pr}(X_m = k_m | Y = k).$$

- We can define the **confusion matrix** $\boldsymbol{A}_m \in \mathbb{R}^{K \times K}$ for each annotator and the **prior probability vector** $\boldsymbol{d} \in \mathbb{R}^K$ such that,

$$\boldsymbol{A}_m(k_m, k) := \mathsf{Pr}(X_m = k_m | Y = k),$$
$$\boldsymbol{d}(k) := \mathsf{Pr}(Y = k)$$

Confusion Matrix



$\boldsymbol{A}_m$ of an annotator, K=3

- Note that columns of $\boldsymbol{A}_m$ and $\boldsymbol{d}$ are probability measures, so it should be nonnegetive and should sum to 1.
- **So the goal is to estimate $\boldsymbol{A}_m$ for $m = 1, \ldots, M$ and $\boldsymbol{d}$.**

Prior Art

- **Dawid-Skene Model** [Dawid and Skene, 1979] :
  - Proposed the Naive Bayesian model for crowd sourcing problem.
  - Based on ML estimation using expectation maximization (EM).
  - Widely used, but a non-convex optimization, model identification and convergence properties are unclear.
- **Spectral Method** [Zhang et al., 2014] :
  - Model identification using orthogonal and symmetric tensor decomposition.
  - Provides an initialization to Dawid-Skene estimator.
  - Provably effective, but using third-order co-occurrences of the annotator responses, thus suffers higher sample complexity.
- **Tensor CPD method** [Traganitis et al., 2018] :
  - Using $Canonical\ Polyadic\ Tensor\ decmposition$ (CPD) technique.
  - Established the identifiability, but not scalable since general tensor decomposition is a quite challenging.
  - Using third order co-occurrences of annotator responses.

Prior Art

- **Dawid-Skene Model** [Dawid and Skene, 1979] :
  - Proposed the Naive Bayesian model for crowd sourcing problem.
  - Based on ML estimation using expectation maximization (EM).
  - Widely used, but a non-convex optimization, model identification and convergence properties are unclear.
- **Spectral Method** [Zhang et al., 2014] :
  - Model identification using orthogonal and symmetric tensor decomposition.
  - Provides an initialization to Dawid-Skene estimator.
  - Provably effective, but using third-order co-occurrences of the annotator responses, thus suffers higher sample complexity.
- **Tensor CPD method** [Traganitis et al., 2018] :
  - Using *Canonical Polyadic Tensor decmposition* (CPD) technique.
  - Established the identifiability, but not scalable since general tensor decomposition is a quite challenging.
  - Using third order co-occurrences of annotator responses.

Pairwise-cooccureces of the response

- Second order statistics has lower sample complexity compared to any higher order statistics (by basic concentration theorems).

- Consider the pairwise joint PMF of any two annoator responses,

$$
\begin{aligned}
\boldsymbol{R}_{m,\ell}(k_m, k_\ell) &= \mathsf{Pr}(X_m = k_m, X_\ell = k_\ell) \\
&= \sum_{k=1}^{K} \underbrace{\mathsf{Pr}(Y = k)}_{\boldsymbol{d}(k)} \underbrace{\mathsf{Pr}(X_m = k_m | Y = k)}_{\boldsymbol{A}_m(k_m, k)} \underbrace{\mathsf{Pr}(X_\ell = k_\ell | Y = k)}_{\boldsymbol{A}_\ell(k_\ell, k)}.
\end{aligned}
$$

- In matrix form, $\boldsymbol{R}_{m,\ell} := \boldsymbol{A}_m \boldsymbol{D} \boldsymbol{A}_\ell^\top$, where $\boldsymbol{D} = \mathrm{Diag}(\boldsymbol{d})$, $\boldsymbol{R}_{m,\ell} \in \mathbb{R}^{K \times K}$.

- In practice, if we are given with the annotator responses $X_m(\boldsymbol{f}_n)$, $\boldsymbol{R}_{m,\ell}$'s can be estimated via sample averaging.

Our Approach

- Consider an annotator m who co-labels the datasamples with annotators $m_1, \ldots, m_{T(m)}$, where $T(m)$ is # of annotators who co-label with m.

- It means, we can construct a matrix $\boldsymbol{Z}_m$ as

$$\boldsymbol{Z}_m = \left[\boldsymbol{R}_{m,m_1}, \boldsymbol{R}_{m,m_2}, \ldots, \boldsymbol{R}_{m,m_{T(m)}}\right].$$

- We can reformulate this as

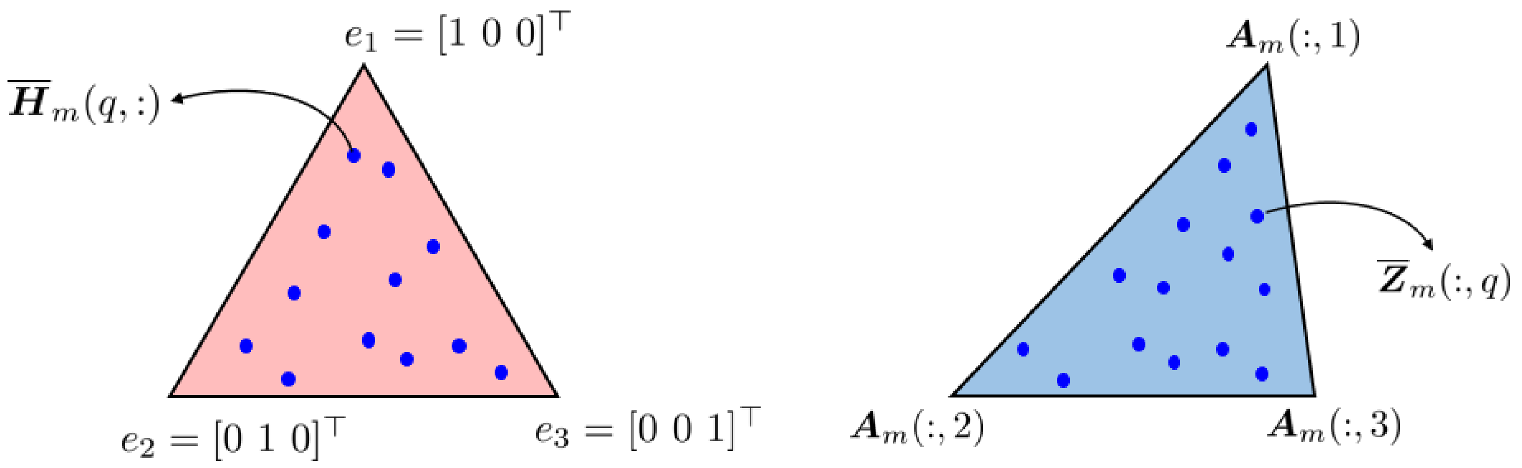
$$\begin{aligned}\boldsymbol{Z}_m &= \left[\boldsymbol{A}_m \boldsymbol{D} \boldsymbol{A}_{m_1}^{\top}, \ldots, \boldsymbol{A}_m \boldsymbol{D} \boldsymbol{A}_{T(m)}^{\top}\right] \\ &= \boldsymbol{A}_m \big[\underbrace{\boldsymbol{D} \boldsymbol{A}_{m_1}^{\top}, \ldots, \boldsymbol{D} \boldsymbol{A}_{T(m)}^{\top}}_{\boldsymbol{H}_m^{\top}}\big] \in \mathbb{R}^{K \times KT(m)}.\end{aligned}$$

- In short, we have to estimate $\boldsymbol{A}_m$ from the formulation $\boldsymbol{Z}_m = \boldsymbol{A}_m \boldsymbol{H}_m^{\top}$, $\forall m$.

Our Approach

- We first normalize the columns of $\boldsymbol{Z}_m$ to get $\overline{\boldsymbol{Z}}_m = \boldsymbol{A}_m \overline{\boldsymbol{H}}_m^\top$ where $\overline{\boldsymbol{H}}_m^\top$ is row normalized and $\boldsymbol{A}_m$ is column normalized by definition.

- So after normalization,
$$\overline{\boldsymbol{H}}_m \mathbf{1} = \mathbf{1},\ \overline{\boldsymbol{H}}_m \geq \mathbf{0},$$
i.e, the rows of $\overline{\boldsymbol{H}}_m$ lies in the probability simplex Δ_K and $\overline{\boldsymbol{Z}}_m \in \text{conv}(\boldsymbol{A}_m)$



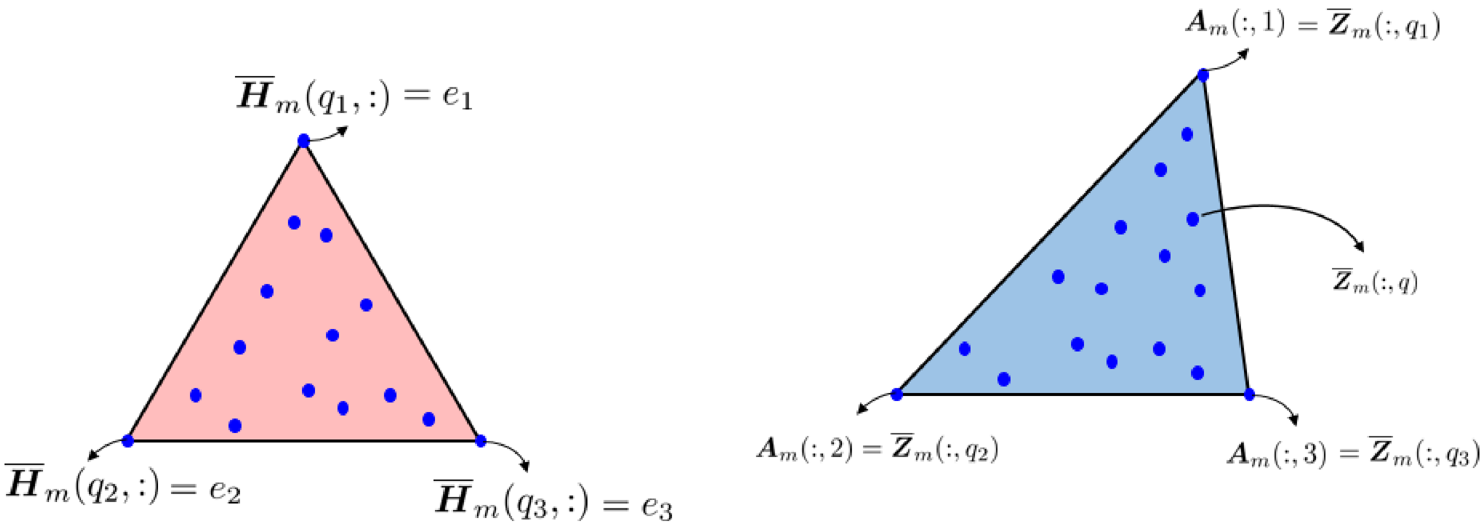
Our Approach

- Let us assume there exits some rows in $\overline{\boldsymbol{H}}_m$ such that,

$$\overline{\boldsymbol{H}}_m(\Lambda_q,:) = \boldsymbol{I}_K,\ \Lambda_q = \{q_1,\ldots,q_K\}.$$

  - known as *seperability* in *Nonnegetive Matrix Factorization* [Donoho and Stodden, 2003].

- Since $\overline{\boldsymbol{Z}}_m = \boldsymbol{A}_m\overline{\boldsymbol{H}}_m^\top$, then $\overline{\boldsymbol{Z}}_m(:,\Lambda_q) = \boldsymbol{A}_m\overline{\boldsymbol{H}}_m(\Lambda_q,:) \Rightarrow \boldsymbol{A}_m = \overline{\boldsymbol{Z}}_m(:,\Lambda_q)$.



Successive Projection Algorithm (SPA)

- **Under the seperability assumption, our task boils down to identifyting Λ_q, an index selection problem!**

- An algebraic algorithm exists which handles index identification known as *Successive Projection Approximation*(SPA) [Arora et al., 2013].

- SPA is a Gram-Schmitt-like algorithm, which only consists of norm comparisons and orthogonal projections.

- We repeat this index identification procedure via SPA for every m and thus all, $\boldsymbol{A}_m$'s are identified and name our approach *MultiSPA*.

Successive Projection Algorithm (SPA)

- **Under the seperability assumption, our task boils down to identifyting Λ_q, an index selection problem!**

- An algebraic algorithm exists which handles index identification known as *Successive Projection Approximation*(SPA) [Arora et al., 2013].

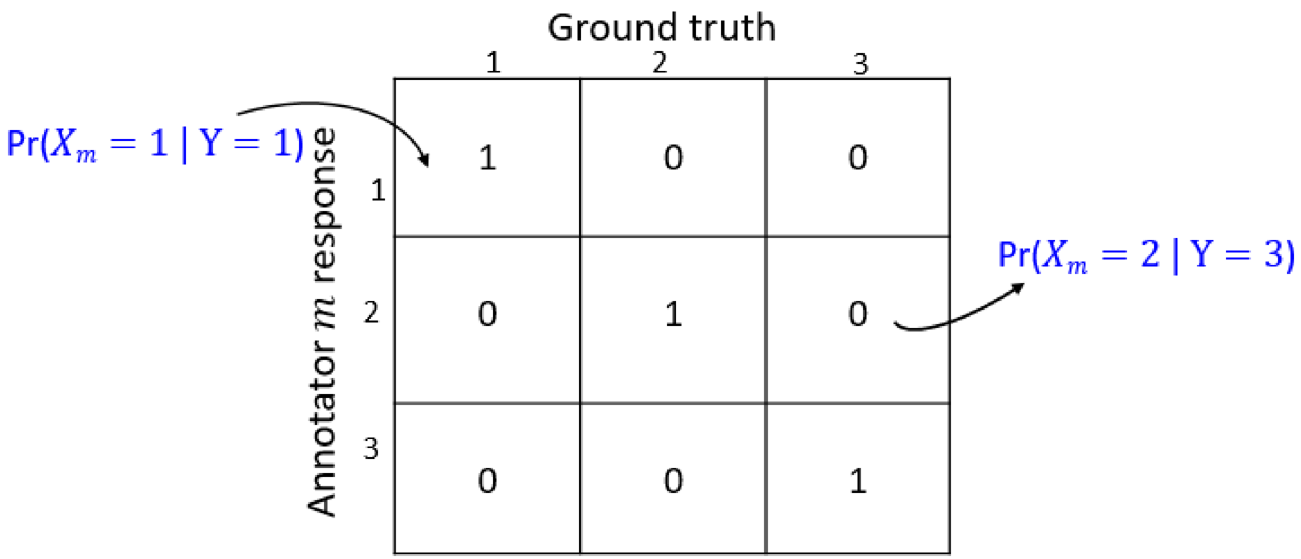
- SPA is a Gram-Schmitt-like algorithm, which only consists of norm comparisons and orthogonal projections.

- We repeat this index identification procedure via SPA for every m and thus all, $\boldsymbol{A}_m$'s are identified and name our approach *MultiSPA*.

The algorithm works under the assumption $\overline{\boldsymbol{H}}_m(\Lambda_q,:) = \boldsymbol{I}_K$.
But what does this mean in crowdsourcing?

A closer look at the assumptions

- Our assumption is $\overline{\boldsymbol{H}}_m(\Lambda_q, :) = \boldsymbol{I}_K$ where $\boldsymbol{H}_m^\top = \boldsymbol{D}[\boldsymbol{A}_{m_1}^\top, \dots, \boldsymbol{A}_{T(m)}^\top]$
- For $K = 3$, an ideal annotator looks as below



$\boldsymbol{A}_m$ of an ideal annotator

A closer look at the assumptions

- If there exists an ideal annotator $\boldsymbol{A}_{m*}$, $m^* \in \{m_1, \ldots, m_{T(m)}\}$, such that

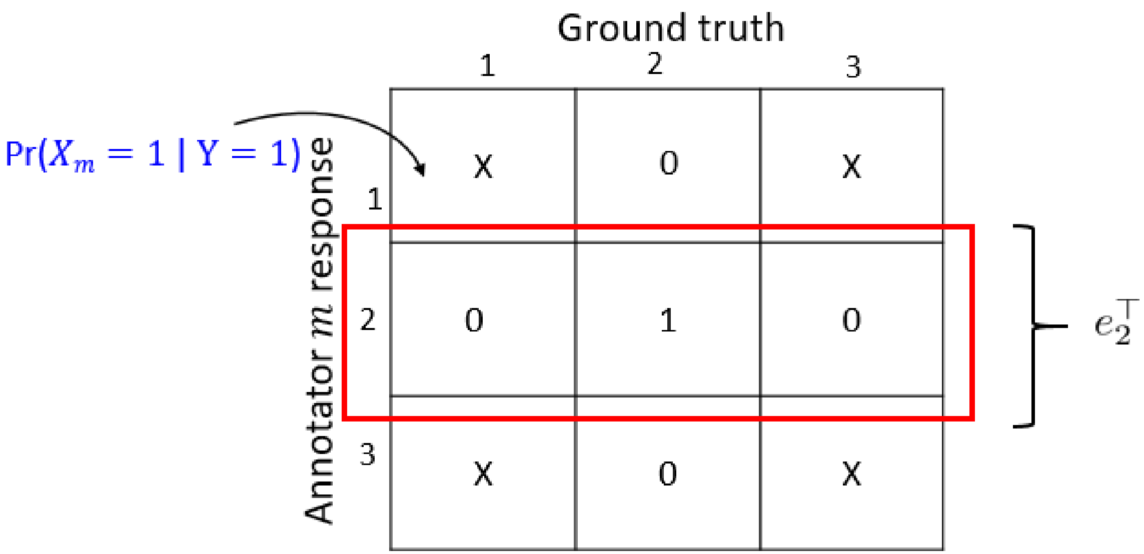
$$\boldsymbol{Z}_m = \boldsymbol{A}_m \underbrace{\boldsymbol{D}\big[\boldsymbol{A}_{m_1}^\top, \ldots, \overbrace{\boldsymbol{A}_{m*}^\top}^{\boldsymbol{I}_K}, \ldots, \boldsymbol{A}_{T(m)}^\top\big]}_{\boldsymbol{H}_m^\top}$$

- This satisfies the condition $\overline{\boldsymbol{H}}_m(\Lambda_q, :) = \boldsymbol{I}_K$ and thus $\boldsymbol{A}_m$ can be identified.

A closer look at the assumptions

- **Another scenario**...

- Consider an annotator who can perfectly identify class k and never confuses with other classes,



$\boldsymbol{A}_m$ of a perfect annotator for class 2

A closer look at the assumptions

- In this way, if every class has a perfect annotator, then

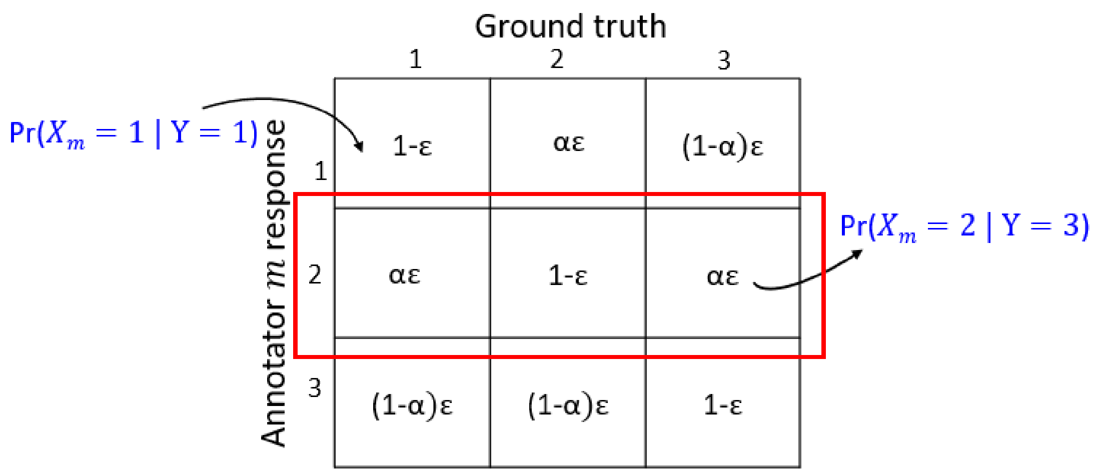
$$\boldsymbol{Z}_m = \boldsymbol{A}_m \underbrace{\boldsymbol{D}\left[\boldsymbol{A}_{m_1}^\top, \ldots, \overset{\overset{e_1}{\uparrow}}{\boldsymbol{A}_{m_{e1}}^\top}, \ldots, \overset{\overset{e_2}{\uparrow}}{\boldsymbol{A}_{m_{e2}}^\top}, \ldots, \overset{\overset{e_3}{\uparrow}}{\boldsymbol{A}_{m_{e3}}^\top}, \ldots, \boldsymbol{A}_{T(m)}^\top\right]}_{\boldsymbol{H}_m^\top}$$

$\overline{\boldsymbol{H}}_m(\Lambda_q, :) = \boldsymbol{I}_K$ can be satisfied

A closer look at the assumptions

- Satisfying $\overline{\boldsymbol{H}}_m(\Lambda_q, :) = \boldsymbol{I}_K$ may be too ideal.

- In practice, the annotators may not be perfect for any class, but can be reasonably good for some class. For example, reasonably good annotator for class 2,



- **Under such cases, does the proposed method work?**?

Identification Theorem

Theorem 1 : Assume that annotators m and t co-label at least S samples $\forall t \in \{m_1, \ldots, m_{T(m)}\}$, and that $\widehat{\boldsymbol{Z}}_m$ is constructed using $\widehat{\boldsymbol{R}}_{m,m_{T(m)}}$'s according to Eq. (). Also assume that the constructed $\widehat{\boldsymbol{Z}}_m$ satisfies $\|\widehat{\boldsymbol{Z}}_m(:,l)\|_1 \geq \eta, \forall l \in \{1, \ldots KT(m)\}$, where $\eta \in (0,1]$. Suppose that $\mathrm{rank}(\boldsymbol{A}_m) = \mathrm{rank}(\boldsymbol{D}) = K$ for $m = 1, \ldots, M$, and that for every class index $k \in \{1, \ldots, K\}$, there exists an annotator $m_{t(k)} \in \{m_1, \ldots, m_{T(m)}\}$ such that

$$\mathsf{Pr}(X_{m_{t(k)}} = k | Y = k) \geq (1 - \epsilon) \sum_{j=1}^{K} \mathsf{Pr}(X_{m_{t(k)}} = k | Y = j),$$

where $\epsilon \in [0,1]$. Then, if $\epsilon \leq \mathcal{O}\left(\max\left(K^{-1}\kappa^{-3}(\boldsymbol{A}_m), \sqrt{\ln(1/\delta)}(\sigma_{\max}(\boldsymbol{A}_m)\sqrt{S}\eta)^{-1}\right)\right)$, with probability greater than $1 - \delta$, the SPA algorithm can estimate an $\widehat{\boldsymbol{A}}_m$ such that

$$\left(\min_{\boldsymbol{\Pi}} \|\widehat{\boldsymbol{A}}_m \boldsymbol{\Pi} - \boldsymbol{A}_m\|_{2,\infty}\right) \leq \mathcal{O}\left(\sqrt{K}\kappa^2(\boldsymbol{A}_m) \max\left(\sigma_{\max}(\boldsymbol{A}_m)\epsilon, \sqrt{\ln(1/\delta)}(\sqrt{S}\eta)^{-1}\right)\right)$$

where $\boldsymbol{\Pi} \in \mathbb{R}^{K \times K}$ is a permutation matrix, $\|\boldsymbol{Y}\|_{2,\infty} = \max_\ell \|\boldsymbol{Y}(:,\ell)\|_2$, $\sigma_{\max}(\boldsymbol{A}_m)$ is the largest singular value of $\boldsymbol{A}_m$, and $\kappa(\boldsymbol{A}_m)$ is the condition number of $\boldsymbol{A}_m$.

Identification Theorem

Theorem 1 : Assume that annotators m and t co-label at least S samples $\forall t \in \{m_1, \dots, m_{T(m)}\}$, and that $\widehat{\boldsymbol{Z}}_m$ is constructed using $\widehat{\boldsymbol{R}}_{m,m_{T(m)}}$'s according to Eq. (). Also assume that the constructed $\widehat{\boldsymbol{Z}}_m$ satisfies $\|\widehat{\boldsymbol{Z}}_m(:,l)\|_1 \geq \eta, \forall l \in \{1, \dots KT(m)\}$, where $\eta \in (0,1]$. Suppose that $\mathrm{rank}(\boldsymbol{A}_m) = \mathrm{rank}(\boldsymbol{D}) = K$ for $m = 1, \dots, M$, and that for every class index $k \in \{1, \dots, K\}$, there exists an annotator $m_{t(k)} \in \{m_1, \dots, m_{T(m)}\}$ such that

$$\mathsf{Pr}(X_{m_{t(k)}} = k | Y = k) \geq (1-\epsilon) \sum_{j=1}^{K} \mathsf{Pr}(X_{m_{t(k)}} = k | Y = j),$$

where $\epsilon \in [0,1]$. Then, if $\epsilon \leq \mathcal{O}\left(\max\left(K^{-1}\kappa^{-3}(\boldsymbol{A}_m), \sqrt{\ln(1/\delta)}(\sigma_{\max}(\boldsymbol{A}_m)\sqrt{S}\eta)^{-1}\right)\right)$, with probability greater than $1-\delta$, the SPA algorithm can estimate an $\widehat{\boldsymbol{A}}_m$ such that

$$\left(\min_{\boldsymbol{\Pi}} \|\widehat{\boldsymbol{A}}_m \boldsymbol{\Pi} - \boldsymbol{A}_m\|_{2,\infty}\right) \leq \mathcal{O}\left(\sqrt{K}\kappa^2(\boldsymbol{A}_m) \max\left(\sigma_{\max}(\boldsymbol{A}_m)\epsilon, \sqrt{\ln(1/\delta)}(\sqrt{S}\eta)^{-1}\right)\right)$$

where $\boldsymbol{\Pi} \in \mathbb{R}^{K \times K}$ is a permutation matrix, $\|\boldsymbol{Y}\|_{2,\infty} = \max_\ell \|\boldsymbol{Y}(:,\ell)\|_2$, $\sigma_{\max}(\boldsymbol{A}_m)$ is the largest singular value of $\boldsymbol{A}_m$, and $\kappa(\boldsymbol{A}_m)$ is the condition number of $\boldsymbol{A}_m$.

Identification Theorem

Theorem 1 : Assume that annotators m and t co-label at least S samples $\forall t \in \{m_1, \dots, m_{T(m)}\}$, and that $\widehat{\boldsymbol{Z}}_m$ is constructed using $\widehat{\boldsymbol{R}}_{m,m_{T(m)}}$'s according to Eq. (). Also assume that the constructed $\widehat{\boldsymbol{Z}}_m$ satisfies $\|\widehat{\boldsymbol{Z}}_m(:,l)\|_1 \geq \eta, \forall l \in \{1, \dots KT(m)\}$, where $\eta \in (0,1]$. Suppose that $\mathrm{rank}(\boldsymbol{A}_m) = \mathrm{rank}(\boldsymbol{D}) = K$ for $m = 1, \dots, M$, and that for every class index $k \in \{1, \dots, K\}$, there exists an annotator $m_{t(k)} \in \{m_1, \dots, m_{T(m)}\}$ such that

$$\mathsf{Pr}(X_{m_{t(k)}} = k | Y = k) \geq (1-\epsilon) \sum_{j=1}^{K} \mathsf{Pr}(X_{m_{t(k)}} = k | Y = j),$$

where $\epsilon \in [0,1]$. Then, if $\epsilon \leq \mathcal{O}\left(\max\left(K^{-1}\kappa^{-3}(\boldsymbol{A}_m), \sqrt{\ln(1/\delta)}(\sigma_{\max}(\boldsymbol{A}_m)\sqrt{S}\eta)^{-1}\right)\right)$, with probability greater than $1-\delta$, the SPA algorithm can estimate an $\widehat{\boldsymbol{A}}_m$ such that

$$\left(\min_{\boldsymbol{\Pi}} \|\widehat{\boldsymbol{A}}_m \boldsymbol{\Pi} - \boldsymbol{A}_m\|_{2,\infty}\right) \leq \mathcal{O}\left(\sqrt{K}\kappa^2(\boldsymbol{A}_m) \max\left(\sigma_{\max}(\boldsymbol{A}_m)\epsilon, \sqrt{\ln(1/\delta)}(\sqrt{S}\eta)^{-1}\right)\right)$$

where $\boldsymbol{\Pi} \in \mathbb{R}^{K \times K}$ is a permutation matrix, $\|\boldsymbol{Y}\|_{2,\infty} = \max_\ell \|\boldsymbol{Y}(:,\ell)\|_2$, $\sigma_{\max}(\boldsymbol{A}_m)$ is the largest singular value of $\boldsymbol{A}_m$, and $\kappa(\boldsymbol{A}_m)$ is the condition number of $\boldsymbol{A}_m$.

Do we favour more annotators?

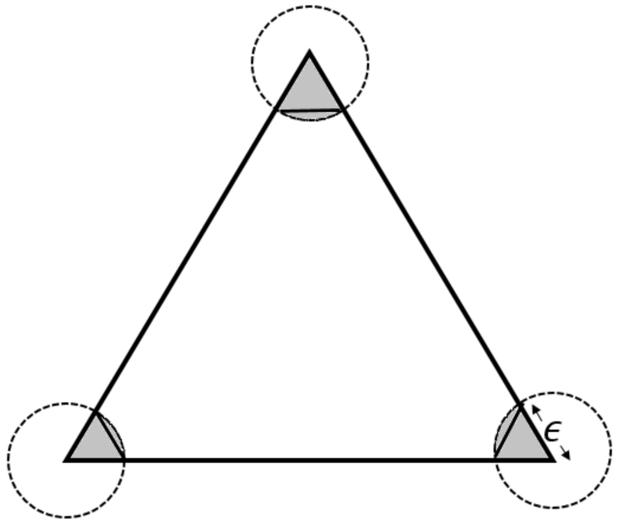
- Recall the construction of $\boldsymbol{Z}_m$,

$$\begin{aligned}\boldsymbol{Z}_m &= \left[\boldsymbol{R}_{m,m_1}, \boldsymbol{R}_{m,m_2}, \ldots, \boldsymbol{R}_{m,m_{T(m)}}\right]. \\ &= \left[\boldsymbol{A}_m \boldsymbol{D} \boldsymbol{A}_{m_1}^{\top}, \ldots, \boldsymbol{A}_m \boldsymbol{D} \boldsymbol{A}_{T(m)}^{\top}\right] \\ &= \boldsymbol{A}_m \underbrace{\boldsymbol{D}\left[\boldsymbol{A}_{m_1}^{\top}, \ldots, \boldsymbol{A}_{T(m)}^{\top}\right]}_{\boldsymbol{H}_m^{\top}} \in \mathbb{R}^{K \times KT(m)}.\end{aligned}$$

Do we favour more annotators?

Theorem 2 :Let $\rho > 0, \varepsilon > 0$, and assume that the rows of $\overline{\boldsymbol{H}}_m$ are generated within the $(K-1)$-probability simplex uniformly at random. If the number of annotators satisfies $M \geq \Omega\left(\frac{\varepsilon^{-2(K-1)}}{K}\log\left(\frac{K}{\rho}\right)\right)$, then, with probability greater than or equal to $1-\rho$, there exist rows of $\overline{\boldsymbol{H}}_m$ indexed by $q_1, \ldots q_K$ such that

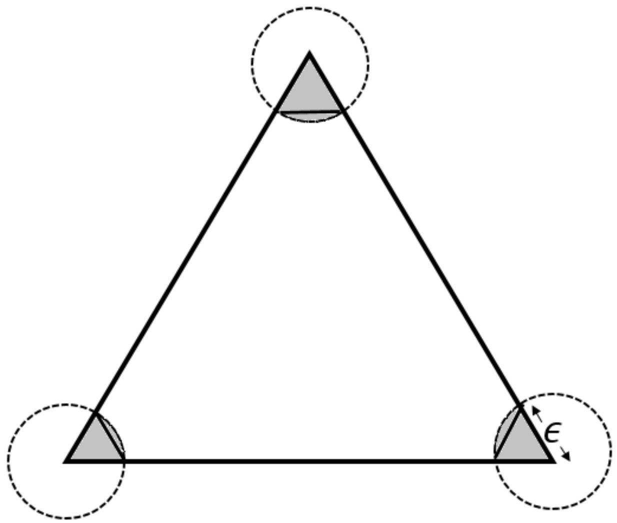
$$\|\overline{\boldsymbol{H}}_m(q_k,:) - \boldsymbol{e}_k^\top\|_2 \leq \varepsilon,\ \ k = 1, \ldots, K.$$



Do we favour more annotators?

Theorem 2 :Let $\rho > 0, \varepsilon > 0$, and assume that the rows of $\overline{\boldsymbol{H}}_m$ are generated within the $(K-1)$-probability simplex uniformly at random. If the number of annotators satisfies $M \geq \Omega\left(\frac{\varepsilon^{-2(K-1)}}{K}\log\left(\frac{K}{\rho}\right)\right)$, then, with probability greater than or equal to $1-\rho$, there exist rows of $\overline{\boldsymbol{H}}_m$ indexed by $q_1, \ldots q_K$ such that

$$\|\overline{\boldsymbol{H}}_m(q_k, :) - \boldsymbol{e}_k^\top\|_2 \leq \varepsilon, \; k = 1, \ldots, K.$$



MultiSPA - In a Nutshell

- Based on Dawid-Skene model which is simple, yet useful.
- Simple, scalable algorithm, like Gram-Schmidt procedure.
- Enjoys lower sample complexity compared to tensor based methods.
- Model parameters can be provably identified under certain assumptions which has practical implications in crowdsourcing.

MultiSPA - In a Nutshell

- Based on Dawid-Skene model which is simple, yet useful.
- Simple, scalable algorithm, like Gram-Schmidt procedure.
- Enjoys lower sample complexity compared to tensor based methods.
- Model parameters can be provably identified under certain assumptions which has practical implications in crowdsourcing.

Can we offer stronger identifiability guarantees?

Identifiability Enhanced Theorem

Theorem 3 : Assume that $\text{rank}(\boldsymbol{D}) = \text{rank}(\boldsymbol{A}_m) = K$ for all $m = 1, \ldots, M$, and that there exist two subsets of the annotator, indexed by $\mathcal{P}_1$ and $\mathcal{P}_2$, where $\mathcal{P}_1 \cap \mathcal{P}_2 = \emptyset$ and $\mathcal{P}_1 \cup \mathcal{P}_2 \subseteq \{1, \ldots, M\}$. Suppose that from $\mathcal{P}_1$ and $\mathcal{P}_2$ the following two matrices can be constructed: $\boldsymbol{H}^{(1)} = [\boldsymbol{A}_{m_1}^\top, \ldots, \boldsymbol{A}_{m_{|\mathcal{P}_1|}}^\top]^\top$, $\boldsymbol{H}^{(2)} = [\boldsymbol{A}_{\ell_1}^\top, \ldots, \boldsymbol{A}_{\ell_{|\mathcal{P}_2|}}^\top]^\top$, where $m_t \in \mathcal{P}_1$ and $\ell_j \in \mathcal{P}_2$. Furthermore, assume that

i) both $\boldsymbol{H}^{(1)}$ and $\boldsymbol{H}^{(2)}$ are *sufficiently scattered*;

ii) all $\boldsymbol{R}_{m_t,\ell_j}$'s for $m_t \in \mathcal{P}_1$ and $\ell_j \in \mathcal{P}_2$ are available; and

iii) for every $m \notin \mathcal{P}_1 \cup \mathcal{P}_2$ there exists a $\boldsymbol{R}_{m,r}$ available, where $r \in \mathcal{P}_1 \cup \mathcal{P}_2$.

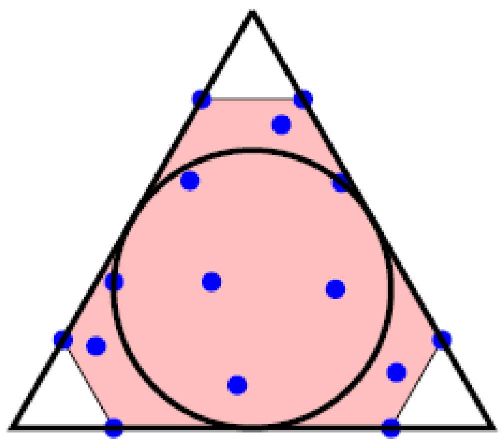
Then, solving the coupled decomposition problem recovers $\boldsymbol{A}_m$ for $m = 1, \ldots, M$ up to a unified permutation ambiguity matrix.

Theorem 3 says...

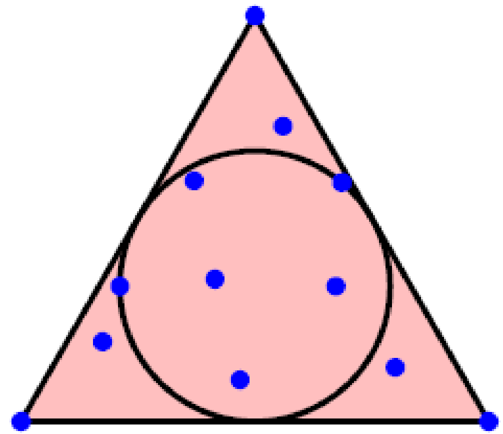
- If we have two annotator groups $\mathcal{P}_1$ and $\mathcal{P}_2$ such that all the pairwise statistics across the group are available, then there may exist a construction as below

$$\boldsymbol{R} = \begin{bmatrix} \boldsymbol{R}_{m_1,\ell_1} & \boldsymbol{R}_{m_1,\ell_2} & \dots & \boldsymbol{R}_{m_1,\ell_{|\mathcal{P}_2|}} \\ \vdots & \vdots & \dots & \vdots \\ \boldsymbol{R}_{m_{|\mathcal{P}_1|},\ell_1} & \boldsymbol{R}_{m_1,\ell_2} & \dots & \boldsymbol{R}_{m_{|\mathcal{P}_2|},\ell_{|\mathcal{P}_2|}} \end{bmatrix}$$
$$= \begin{bmatrix} \boldsymbol{A}_{m_1} \\ \vdots \\ \boldsymbol{A}_{m_{|\mathcal{P}_1|}}. \end{bmatrix} \boldsymbol{D}[\boldsymbol{A}_{\ell_1}^\top, \dots, \boldsymbol{A}_{\ell_{|\mathcal{P}_2|}}^\top] = \underbrace{\boldsymbol{H}^{(1)}\boldsymbol{D}}_{\boldsymbol{W}}(\underbrace{\boldsymbol{H}^{(2)}}_{\boldsymbol{H}})^\top.$$

- If the rows of $\boldsymbol{W}$ and $\boldsymbol{H}$ satisfies a certain geometrical property called sufficiently scattered (SS) condition, then $\boldsymbol{W}$ and $\boldsymbol{H}$ are identifiable upto trivial ambiguity [Huang et al., 2014].



Sufficiently scattered $\boldsymbol{W}$



Seperable $\boldsymbol{W}$

- SS is much easier to satisfy relative to seperability.
  - We do not need extremely well trained annotators for each class to satisfy SS.

Theorem 3 says...

- By solving the below proposed coupled decomposition problem, $\boldsymbol{A}_m$ for $m = 1, \ldots, M$ can be estimated

$$
\begin{aligned}
\text{find} \quad & \{\boldsymbol{A}_m\}_{m=1}^{M},\ \boldsymbol{D} \\
\text{subject to} \quad & \boldsymbol{R}_{m,\ell} = \boldsymbol{A}_m \boldsymbol{D} \boldsymbol{A}_\ell^\top,\ \forall m, \ell \in \{1, \ldots, M\} \\
& \boldsymbol{1}^\top \boldsymbol{A}_m = \boldsymbol{1}^\top,\ \boldsymbol{A}_m \geq \boldsymbol{0},\ \forall m \\
& \boldsymbol{1}^\top \boldsymbol{d} = 1,\ \boldsymbol{d} \geq \boldsymbol{0}.
\end{aligned}
$$

Does SS condition favour more annotators?

Theorem 4: Let $\rho > 0, \frac{\alpha}{2} > \varepsilon > 0,$, and assume that the rows of $\boldsymbol{H}^{(1)}$ and $\boldsymbol{H}^{(2)}$ are generated from $\mathbb{R}^K$ uniformly at random. If the number of annotators satisfies $M \geq \Omega\left(\frac{(K-1)^2}{K\alpha^{2(K-2)}\varepsilon^2}\log\left(\frac{K(K-1)}{\rho}\right)\right)$, where $\alpha = 1$ for $K = 2$, $\alpha = 2/3$ for $K = 3$ and $\alpha = 1/2$ for $K > 3$, then with probability greater than or equal to $1-\rho$, $\boldsymbol{H}^{(1)}$ and $\boldsymbol{H}^{(2)}$ are ε-sufficiently scattered.

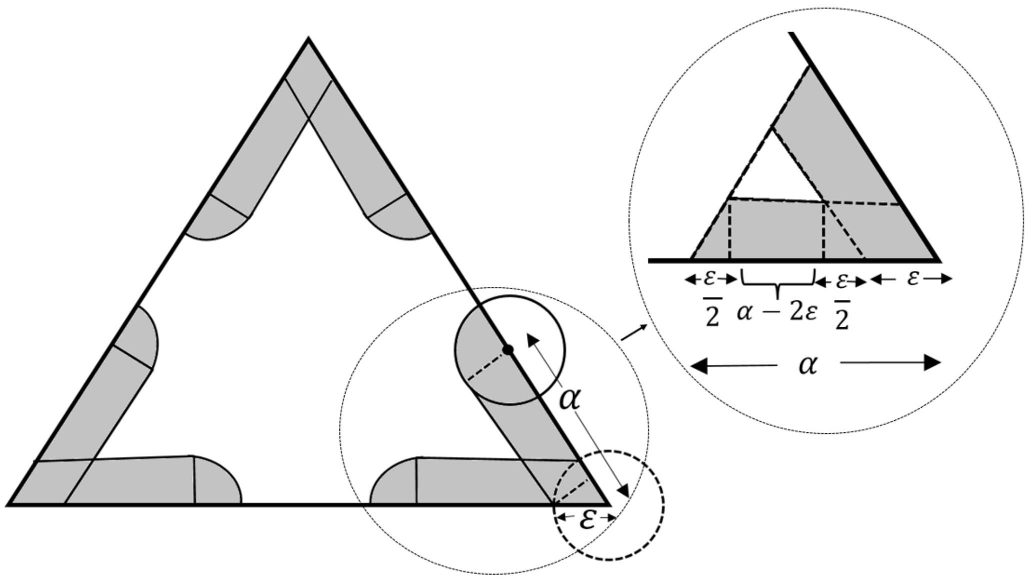


Figure 1: (ε-sufficiently scatterd)

Does SS condition favour more annotators?

Theorem 4: Let $\rho > 0, \frac{\alpha}{2} > \varepsilon > 0,$, and assume that the rows of $\boldsymbol{H}^{(1)}$ and $\boldsymbol{H}^{(2)}$ are generated from $\mathbb{R}^K$ uniformly at random. If the number of annotators satisfies $M \geq \Omega\left(\frac{(K-1)^2}{K\alpha^{2(K-2)}\varepsilon^2}\log\left(\frac{K(K-1)}{\rho}\right)\right)$, where $\alpha = 1$ for $K = 2$, $\alpha = 2/3$ for $K = 3$ and $\alpha = 1/2$ for $K > 3$, then with probability greater than or equal to $1 - \rho$, $\boldsymbol{H}^{(1)}$ and $\boldsymbol{H}^{(2)}$ are ε-sufficiently scattered.

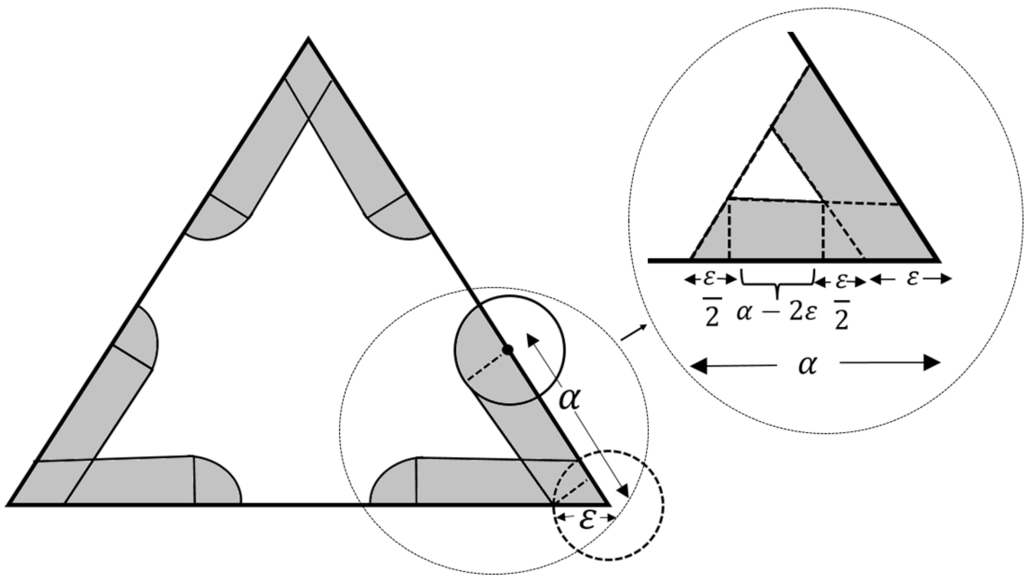


Figure 2: (ε-sufficiently scatterd)

Alternating Optimization KL Algorithm

- Kullback-Leibler (KL) divergence is the natural distance measure under proaba-bility measures. Let $\boldsymbol{R}$ and $\hat{\boldsymbol{R}}$ are two probability distribution matrices, then

$$D_{KL}(\boldsymbol{R}||\hat{\boldsymbol{R}}) = -\sum_{i,j} \boldsymbol{R}_{ij} \text{log} \frac{\boldsymbol{R}_{ij}}{\hat{\boldsymbol{R}}_{ij}} \tag{1}$$

- So we use KL divergence in our fitting probelm

$$\begin{aligned} \underset{\{\boldsymbol{A}_m\}_{m=1}^{M},\ \boldsymbol{D}}{\text{minimize}} \quad & \sum_{m,\ell} D_{KL}\left(\boldsymbol{R}_{m,\ell}||\boldsymbol{A}_m \boldsymbol{D} \boldsymbol{A}_\ell^\top\right) \\ \text{subject to} \quad & \mathbf{1}^\top \boldsymbol{A}_m = \mathbf{1}^\top,\ \boldsymbol{A}_m \geq \mathbf{0},\ \forall m \\ & \mathbf{1}^\top \boldsymbol{d} = 1,\ \boldsymbol{d} \geq \mathbf{0}, \end{aligned}$$

- It is a non-convex optimization. So, we use alternating optimization (AO) approach, by cyclically updating each parameters, solving the convex subproblems using mirror descent.

Experiment setup & Results

- UCI datasets (https://archive.ics.uci.edu/ml/datasets.html) are considered

- For each dataset, we use different MATLAB classifiers to annotate the data samples

Table 1: Details of UCI Datasets.

UCI dataset name	# classes	# items	# annotators
Adult	2	7017	10
Mushroom	2	6358	10
Nursery	4	3575	10

Experiment setup & Results

- For training, we use 20% of the samples to act as training data
- In practice, not all samples are labeled by an annotator
- To simulate such a scenario, each of the trained algorithms is allowed to label a data sample with probability $p < 1$. A smaller p means a more challenging scenario
- We use MAP estimator to predict the labels, e.g,

$$\hat{y}_{\text{MAP}} = \arg\max_{k \in [K]} \left[\log(\boldsymbol{d}(k)) + \sum_{m=1}^{M} \log\big(\boldsymbol{A}_m(k_m, k)\big) \right]$$

- We compare the performance with different baselines.

Results

Table 2: Classification Error (%) on UCI Datasets

	Nursery			Mushroom			Adult		
Algorithms	$p=1$	$p=0.5$	$p=0.2$	$p=1$	$p=0.5$	$p=0.2$	$p=1$	$p=0.5$	$p=0.2$
`MultiSPA`	2.83	4.54	17.96	0.02	0.293	6.35	**15.71**	**16.05**	17.66
`MultiSPA-KL`	**2.72**	**4.26**	**13.06**	**0.00**	**0.152**	**5.89**	**15.66**	**15.98**	**17.63**
`MultiSPA-D&S`	**2.82**	**4.44**	13.39	**0.00**	0.194	6.17	15.74	16.29	23.88
`Spectral-D&S`	3.14	37.2	44.29	**0.00**	0.198	6.17	15.72	16.31	23.97
`TensorADMM`	17.97	7.26	19.78	0.06	0.237	6.18	15.72	**16.05**	25.08
`MV-D&S`	2.92	66.48	66.61	**0.00**	47.99	48.63	15.76	75.21	75.13
`Minmax-entropy`	3.63	26.31	**11.09**	**0.00**	**0.163**	8.14	16.11	16.92	**15.64**
`EigenRatio`	N/A	N/A	N/A	0.06	0.329	**5.97**	15.84	16.28	17.69
`KOS`	4.21	6.07	13.48	0.06	0.576	6.42	17.19	24.97	38.29
`Ghosh-SVD`	N/A	N/A	N/A	0.06	0.329	**5.97**	15.84	16.28	17.71
`Majority Voting`	2.94	4.83	19.75	0.14	0.566	6.57	15.75	16.21	20.57
Single Best	3.94	N/A	N/A	0.00	N/A	N/A	16.23	N/A	N/A
Single Worst	15.65	N/A	N/A	7.22	N/A	N/A	19.27	N/A	N/A

Experiment setup and Results

- The datasets annotated by Amazon Mechanical Turk (https://www.mturk.com) (AMT) workers are used here

Table 3: AMT Dataset description.

Dataset	# classes	# items	# annotators	# annotator labels
Bird	2	108	30	3240
RTE	2	800	164	8,000
TREC	2	19,033	762	88,385
Dog	4	807	52	7,354
Web	5	2,665	177	15,567

- We use MAP estimator to predict the labels, e.g,

$$\hat{y}_{\mathrm{MAP}} = \underset{k \in [K]}{\arg\max} \left[\log(\boldsymbol{d}(k)) + \sum_{m=1}^{M} \log\big(\boldsymbol{A}_m(k_m, k)\big) \right]$$

Experiment setup and Results

Table 4: Classification Error (%) and Run-time (sec) : AMT Datasets

Algorithms	TREC		Bluebird		RTE	
	(%) Error	(sec) Time	(%) Error	(sec) Time	(%) Error	(sec) Time
MultiSPA	31.47	50.68	13.88	0.07	8.75	0.28
MultiSPA-KL	**29.23**	536.89	**11.11**	1.94	**7.12**	17.06
MultiSPA-D&S	29.84	53.14	12.03	0.09	**7.12**	0.32
Spectral-D&S	**29.58**	919.98	12.03	1.97	**7.12**	6.40
TensorADMM	N/A	N/A	12.03	2.74	N/A	N/A
MV-D&S	30.02	3.20	12.03	0.02	7.25	0.07
Minmax-entropy	91.61	352.36	**8.33**	3.43	7.50	9.10
EigenRatio	43.95	1.48	27.77	0.02	9.01	0.03
KOS	51.95	9.98	**11.11**	0.01	39.75	0.03
GhoshSVD	43.03	11.62	27.77	0.01	49.12	0.03
Majority Voting	34.85	N/A	21.29	N/A	10.31	N/A

Experiment setup and Results

Table 5: Classification Error (%) and Run-time (sec) : AMT Datasets

Algorithms	**Web**		**Dog**	
	(%) Error	**(sec) Time**	**(%) Error**	**(sec) Time**
MultiSPA	15.22	0.54	17.09	0.07
MultiSPA-KL	**14.58**	12.34	**15.48**	15.88
MultiSPA-D&S	15.11	0.84	**16.11**	0.12
Spectral-D&S	16.88	179.92	17.84	51.16
TensorADMM	N/A	N/A	17.96	603.93
MV-D&S	16.02	0.28	15.86	0.04
Minmax-entropy	**11.51**	26.61	16.23	7.22
EigenRatio	N/A	N/A	N/A	N/A
KOS	42.93	0.31	31.84	0.13
GhoshSVD	N/A	N/A	N/A	N/A
Majority Voting	26.93	N/A	17.91	N/A

Conclusion & Future direction

- We proposed a **second order statistics** based approach for identifiability to the Dawid-Skene model for crowdsourcing
- The proposed $multiSPA$ algorithm is simple, **light weight and need lower sample complexity** compared to existing tensor based methods
- We also proposed an approach with **enhanced identifiabity** and provided an alternating optimization based algorithm
- We supported our theoretical analysis using **experimental results**.
- Potential future works:
  - Analyze the dependent annotator and dependent data scenario.
  - Faster algorithm for the proposed coupled decomposition problem.

*

References

S. Arora, R. Ge, Y. Halpern, D. Mimno, A. Moitra, D. Sontag, Y. Wu, and M. Zhu. A practical algorithm for topic modeling with provable guarantees. In *Proceedings of ICML*, 2013.

Alexander Philip Dawid and Allan M Skene. Maximum likelihood estimation of observer error-rates using the em algorithm. *Applied statistics*, pages 20–28, 1979.

D. Donoho and V. Stodden. When does non-negative matrix factorization give a correct decomposition into parts? In *Advances in neural information processing systems*, volume 16, 2003.

K. Huang, N. D. Sidiropoulos, and A. Swami. Non-negative matrix factorization revisited: New uniqueness results and algorithms. *IEEE Trans. Signal Process.*, 62 (1):211–224, Jan. 2014.

Panagiotis A Traganitis, Alba Pages-Zamora, and Georgios B Giannakis. Blind multiclass ensemble classification. *IEEE Trans. Signal Process.*, 66(18):4737–4752, 2018.

Yuchen Zhang, Xi Chen, Dengyong Zhou, and Michael I Jordan. Spectral methods meet em: A provably optimal algorithm for crowdsourcing. In *Advances in neural information processing systems*, pages 1260–1268, 2014.

Synthetic Data Experiments

- No of annotators $M = 25$, no of classes $K = 3$, no of items $N = 10000$.
- Case 1: A randomly chosen annotator is assigned identity matrix as confusion matrix.

Table 6: Average MSE of the confusion matrices $\boldsymbol{A}_m$ for case 1.

Algorithms	$p = 0.2$	$p = 0.3$	$p = 0.5$	$p = 1$
`MutliSPA`	0.0184	0.0083	0.0063	0.0034
`MultiSPA-KL`	**0.0019**	**0.0009**	**0.0004**	**1.73E-04**
`Spectral D&S`	0.0320	0.0112	0.0448	1.74E-04
`TensorADMM`	0.0026	0.0011	0.0005	1.88E-04
`MV-D&S`	–	–	0.0173	1.84E-04

Synthetic Data Experiments

- No of annotators $M = 25$, no of classes $K = 3$, no of items $N = 10000$.
- Case 2: A randomly chosen annotator is assigned a diagonally dominant confusion matrix.

Table 7: Average MSE of the confusion matrices $\boldsymbol{A}_m$ for case 2.

Algorithms	$p = 0.2$	$p = 0.3$	$p = 0.5$	$p = 1$
`MutliSPA`	0.0229	0.0188	0.0115	0.0102
`MultiSPA-KL`	**0.0029**	**0.0014**	**0.0005**	**1.67E-04**
`Spectral D&S`	0.0348	0.0265	0.0391	**1.67E-04**
`TensorADMM`	0.0031	0.0016	0.0006	1.93E-04
`MV-D&S`	–	–	0.0028	5.88E-04

Synthetic Data Experiments

Table 8: Classification Error(%) & Averge run-time when $\boldsymbol{d} = [\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]^\top$

Algorithms	$p = 0.2$	$p = 0.3$	$p = 0.5$	**Run-time(sec)**
`MultiSPA`	37.24	26.39	19.21	0.049
`MultiSPA-KL`	**31.71**	**21.10**	**12.79**	18.07
`MultiSPA-D&S`	**31.95**	**21.11**	**12.80**	0.069
`Spectral-D&S`	46.37	23.92	12.89	27.17
`TensorADMM`	32.16	21.34	12.91	56.09
`MV-D&S`	66.91	57.92	13.09	0.096
`Minmax-entropy`	62.83	65.50	67.31	200.91
`KOS`	71.47	61.05	13.12	5.653
`Majority Voting`	67.57	68.37	71.39	–

Synthetic Data Experiments

Table 9: Classification Error(%) & Averge run-time when $\boldsymbol{d} = [\frac{1}{6}, \frac{2}{3}, \frac{1}{6}]^\top$

Algorithms	$p = 0.2$	$p = 0.3$	$p = 0.5$	**Run-time(sec)**
`MultiSPA`	**30.75**	**21.29**	**13.67**	0.105
`MultiSPA-KL`	**23.19**	**16.62**	**10.13**	18.93
`MultiSPA-D&S`	40.12	32.1	21.46	0.122
`Spectral-D&S`	56.17	49.41	39.17	28.01
`TensorADMM`	34.17	25.53	11.97	152.76
`MV-D&S`	83.14	83.15	32.98	0.090
`Minmax-entropy`	83.04	63.08	74.29	232.82
`KOS`	70.79	67.55	78.00	6.19
`Majority Voting`	65.37	65.57	66.06	–

Synthetic Data Experiments

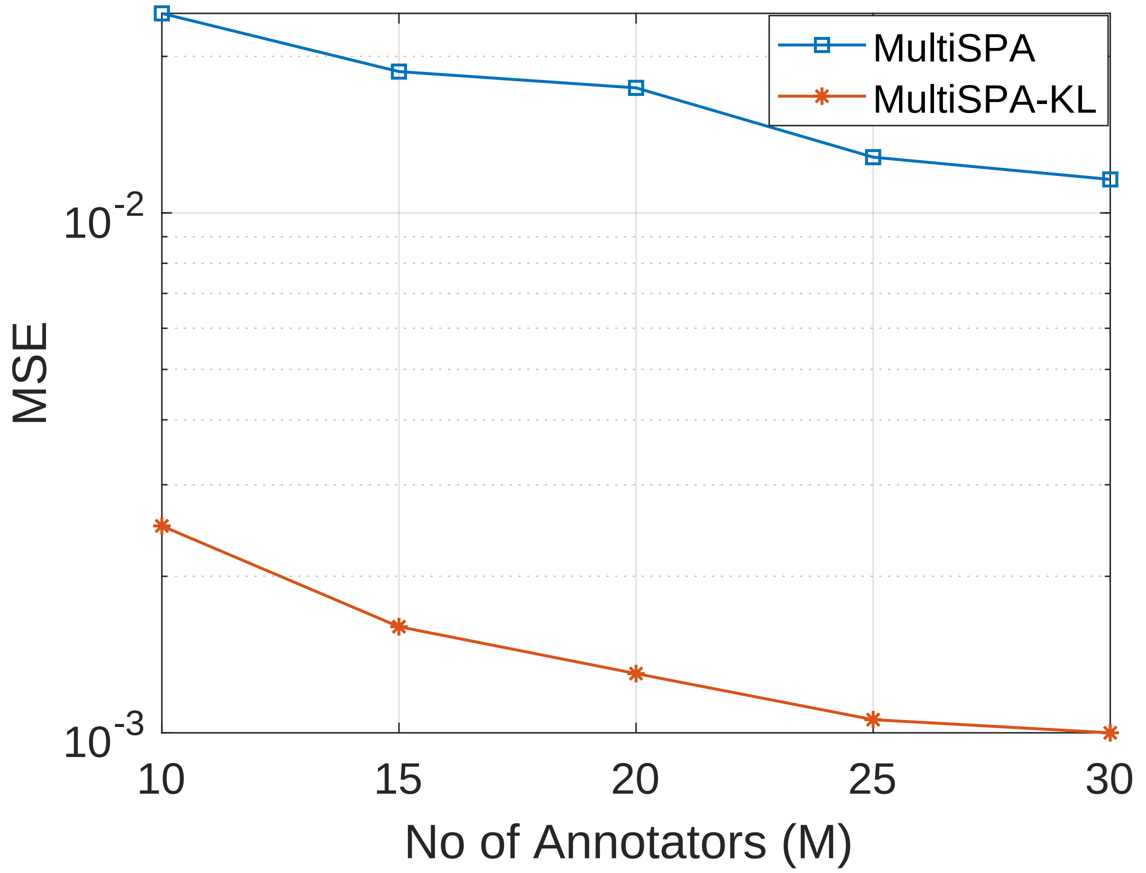


Figure 3: MSE of the confusion matrices for various values of M

UCI Dataset Experiments - Run-time performance

Table 10: Average runtime (sec) for UCI datset experiments.

Algorithms	Nursery	Mushroom	Adult
MultiSPA	0.021	0.012	0.018
MultiSPA-KL	1.112	0.663	0.948
MultiSPA-D&S	0.035	0.027	0.027
Spectral-D&S	10.09	0.496	0.512
TensorADMM	5.811	0.743	4.234
MV-D&S	0.009	0.007	0.008
Minmax-entropy	19.94	2.304	6.959
EigenRatio	–	0.005	0.007
KOS	0.768	0.085	0.118
Ghosh-SVD	–	0.081	0.115

Resolving Permutation ambiguity

- SPA-estimated $\hat{\boldsymbol{A}}_m$ is up to column permutation, even if there is no noise, i.e., $\hat{\boldsymbol{A}}_m = \boldsymbol{A}_m \boldsymbol{\Pi}_m$, Π_m is the permutation matrix.

- A very practical heuristic can be used to resolve permutation ambiguity - if one believes that all the annotators are reasonably trained, then we can rearrange the columns of $\hat{\boldsymbol{A}}_m$ so that it is diagonal dominant

- Once $\boldsymbol{A}_m$ are identified, $\boldsymbol{d}$ can be estimated by $\boldsymbol{D} = \boldsymbol{A}_m^{-1} \boldsymbol{R}_{m,\ell} (\boldsymbol{A}_\ell^\top)^{-1}$ using any $m, l \in \{1, \ldots, M\}$

Experiment setup and Results

- The datasets annotated by Amazon Mechanical Turk (https://www.mturk.com) (AMT) workers are used here

Table 11: AMT Dataset Description

Dataset name	# classes	# items	# annotators
Bluebird	2	108	39
RTE	2	800	20
Dog	4	807	20

Experiment setup and Results

Classification Error (%) : AMT Datasets

Algorithms	RTE	Dog	Bluebird
MultiSPA	17.87	24.9	12.96
MultiSPA-KL	**17.37**	**24.89**	11.11
Spectral-D&S	17.75	25.52	**10.19**
TensorADMM	**17.50**	40.64	**10.19**
MV-D&S	18.75	**21.32**	11.11
Majority Voting	33.62	26.59	24.07
KOS	40.12	41.38	11.11

Successive Projection Algorithm (SPA)

- An algebraic algorithm exists which handles index identification known as *Successive Projection Approximation*(SPA) [Arora et al., 2013].
- Consider a column in $\overline{\boldsymbol{Z}}_m$, then,

$$
\begin{aligned}
\left\|\overline{\boldsymbol{Z}}_m(:,q)\right\|_2 &= \left\|\sum_{k=1}^{K} \boldsymbol{A}_m(:,k)\overline{\boldsymbol{H}}_m(q,k)\right\|_2, && \text{(data model)}\\
&\leq \sum_{k=1}^{K} \left\|\boldsymbol{A}_m(:,k)\overline{\boldsymbol{H}}_m(q,k)\right\|_2, && \text{(triangular inequality)}\\
&= \sum_{k=1}^{K} \overline{\boldsymbol{H}}_m(q,k)\left\|\boldsymbol{A}_m(:,k)\right\|_2, && \text{(non-negetivity of } \overline{\boldsymbol{H}}_m)\\
&\leq \max_{k=1,\ldots,K} \left\|\boldsymbol{A}_m(:,k)\right\|_2, && \text{(rows of } \overline{\boldsymbol{H}}_m \text{ sum to one)}
\end{aligned}
$$

Successive Projection Algorithm (SPA)

- By this inequality, the column index corresponding to first vertex, $\hat{q}_1$ is identified as,

$$\hat{q}_1 = \arg\max_{q} \left\| \overline{\boldsymbol{Z}}_m(:, q) \right\|_2^2 . \tag{2}$$

- Then all the remaining columns of $\overline{\boldsymbol{Z}}_m$ are projected to the orthogonal complement of the selected column, we repeat the vertex identification for $K-1$ times.

- We repeat this index identification procedure for every m and thus all, $\boldsymbol{A}_m$'s are identified and name our approach $MultiSPA$.

Successive Projection Algorithm (SPA)

- By this inequality, the column index corresponding to first vertex, $\hat{q}_1$ is identified as,

$$\hat{q}_1 = \arg\max_{q} \ \left\| \overline{\boldsymbol{Z}}_m(:, q) \right\|_2^2 . \tag{3}$$

- Then all the remaining columns of $\overline{\boldsymbol{Z}}_m$ are projected to the orthogonal complement of the selected column, we repeat the vertex identification for $K-1$ times.

- We repeat this index identification procedure for every m, thus all $\boldsymbol{A}_m$'s are identified and name our approach $MultiSPA$.

The algorithm works under the assumption $\overline{\boldsymbol{H}}_m(\Lambda_q, :) = \boldsymbol{I}_K$.
But what does this mean in crowdsourcing?