Optimal Permutation estimation in crowdsourcing problems

Alexandra Carpentier, Emmanuel Pilliat, and Nicolas Verzelen

Universität Potsdam, Université de Montpellier, and INRAE
Crowdsourcing Problems = Aggregation of Experts’ opinion

To calibrate the method: need to evaluate the reliability of the experts
Crowdsourcing Problems = Aggregation of Experts’ opinion

To calibrate the method: need to evaluate the reliability of the experts

Our Goal

Ranking $n$ experts according to their ability on $d$ questions
$n$ experts and $d$ questions

Observation Model

\[
Y = M + E \in \mathbb{R}^{n \times d}
\]

- $(E_{i,k})$ independent and Subgaussian (e.g. Bernoulli)
- $M_{i,k} \in [0,1]$ for all $i,k$
Statistical Model

$n$ experts and $d$ questions

<table>
<thead>
<tr>
<th>Observation Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Y = M + E \in \mathbb{R}^{n \times d}$</td>
</tr>
</tbody>
</table>

- $(E_{i,k})$ independent and Subgaussian (e.g. Bernoulli)
- $M_{i,k} \in [0,1]$ for all $i, k$

**Parametric Models for $M$:**

- *Questions equally difficult* $\sim M_{ij} = a_i$ $\approx$ [Dawid and Skene, 1979]
Statistical Model

$n$ experts and $d$ questions

**Observation Model**

\[ Y = M + E \in \mathbb{R}^{n \times d} \]

- $(E_{i,k})$ independent and Subgaussian (e.g. Bernoulli)
- $M_{i,k} \in [0,1]$ for all $i,k$

**Parametric Models for $M$**:

- **Questions equally difficult** \( \sim M_{ij} = a_i \) \( \approx [\text{Dawid and Skene, 1979}] \)
- **Ability/difficulty** \( \sim M_{ij} = \phi(\alpha_i - \beta_j) \) \( \approx [\text{Bradley and Terry, 1952}] \)
Statistical Model

$n$ experts and $d$ questions

Observation Model

\[ Y = M + E \in \mathbb{R}^{n \times d} \]

- $(E_{i,k})$ independent and Subgaussian (e.g. Bernoulli)
- $M_{i,k} \in [0,1]$ for all $i,k$

Parametric Models for $M$:
- Questions equally difficult $\sim M_{ij} = a_i \approx$ [Dawid and Skene, 1979]
- Ability/difficulty $\sim M_{ij} = \phi(\alpha_i - \beta_j) \approx$ [Bradley and Terry, 1952]

Non-Parametric Models for $M \approx$ [Mao et al., 2018]
- Increasing columns **up to permutation** $\pi^*$ of rows: $M_{\pi^{*-1}(i),k} \leq M_{\pi^{*-1}(i+1),k}$
Statistical Model

$n$ experts and $d$ questions

### Observation Model

\[ Y = M + E \quad \in \mathbb{R}^{n \times d} \]

- \((E_{i,k})\) independent and Subgaussian (e.g. Bernoulli)
- \(M_{i,k} \in [0,1]\) for all \(i,k\)

### Parametric Models for \(M\):

- **Questions equally difficult** \(\sim M_{ij} = a_i \quad \approx [\text{Dawid and Skene, 1979}]\)
- **Ability/difficulty** \(\sim M_{ij} = \phi(\alpha_i - \beta_j) \quad \approx [\text{Bradley and Terry, 1952}]\)

### Non-Parametric Models for \(M\) \(\approx [\text{Mao et al., 2018}]\)

- Increasing columns **up to permutation** \(\pi^*\) of rows: \(M_{\pi^*-1}(i),k \leq M_{\pi^*-1}(i+1),k\)
- Rows are increasing: \(M_{i,k} \leq M_{i,k+1}\)
$n$ experts and $d$ questions

**Observation Model**

$$Y = M + E \in \mathbb{R}^{n \times d}$$

- $(E_{i,k})$ independent and Subgaussian (e.g. Bernoulli)
- $M_{i,k} \in [0, 1]$ for all $i, k$

**Non-Parametric Models for** $M \approx$ [Mao et al., 2018]

- Increasing columns **up to permutation** $\pi^*$ of rows: $M_{\pi^* - 1}(i), k \leq M_{\pi^* - 1}(i + 1), k$
- Rows are increasing: $M_{i,k} \leq M_{i,k+1}$

**Aim**

Estimation of $\pi^*$.

Partial observation of $Y$ discussed later.
Statistical Model

$n$ experts and $d$ questions

**Observation Model**

\[ Y = M + E \quad \epsilon \mathbb{R}^{n \times d} \]

- $(E_{i,k})$ independent and Subgaussian (e.g. Bernoulli)
- $M_{i,k} \in [0,1]$ for all $i,k$

**Non-Parametric Models for $M$** $\approx$ [Mao et al., 2018]

- Increasing columns up to permutation $\pi^*$ of rows: $M_{\pi^*-1(i),k} \leq M_{\pi^*-1(i+1),k}$
- Rows are increasing: $M_{i,k} \leq M_{i,k+1}$

**Aim**

Estimation of $\pi^*$.

Partial observation of $Y$ discussed later.
Statistical Model

$n$ experts and $d$ questions

Observation Model

\[ Y = M + E \in \mathbb{R}^{n \times d} \]

- \((E_{i,k})\) independent and Subgaussian (e.g. Bernoulli)
- \(M_{i,k} \in [0,1]\) for all \(i,k\)

Non-Parametric Models for \(M \sim [\text{Mao et al., 2018}]\)

- Increasing columns up to permutation \(\pi^*\) of rows: 
  \[ M_{\pi^*-1}(i),k \leq M_{\pi^*-1}(i+1),k \]
- Rows are increasing: 
  \[ M_{i,k} \leq M_{i,k+1} \]

Aim

Estimation of \(\pi^*\).

Partial observation of \(Y\) discussed later.
Loss functions

Permutation loss for $\hat{\pi}$

$$l(\hat{\pi}, \pi^*) := \| M_{\hat{\pi}-1} - M_{\pi^* - 1} \|_F^2 = \sum_{i=1}^{n} \sum_{k=1}^{d} (M_{\pi^*-1}(i),k - M_{\pi^*-1}(i),k)^2$$
**Loss functions**

**Permutation loss for $\hat{\pi}$**

$$l(\hat{\pi}, \pi^*) := \| M_{\hat{\pi}}^{-1} - M_{\pi^*}^{-1} \|_{F^2}^2 = \sum_{i=1}^{n} \sum_{k=1}^{d} (M_{\pi}(i,k) - M_{\pi^*}(i,k))^2$$

**Estimation loss for $\hat{M}$**

$$\| \hat{M} - M \|_{F^2}^2.$$
### Loss functions

#### Permutation loss for $\hat{\pi}$

$$ l(\hat{\pi}, \pi^*) := \| M_{\hat{\pi}}^{-1} - M_{\pi^*}^{-1} \|_F^2 = \sum_{i=1}^n \sum_{k=1}^d (M_{\pi}^{-1}(i,k) - M_{\pi^*}^{-1}(i,k))^2 $$

#### Estimation loss for $\hat{M}$

$$ \| \hat{M} - M \|_F^2. $$

**Remark:**

- Estimation of $\pi^*$ is "less demanding" than estimation of $M$.
- Estimating a bi-isotonic matrix computationally simple.
Interpretation of permutation Loss

\[ l(\hat{\pi}, \pi^*) := \| M_{\hat{\pi}^{-1}} - M_{\pi^*^{-1}} \|_F^2. \]

If green and red misclassified: Perm-Loss = \(2rh^2\).
Minimax Risk

\[ \mathcal{R}_{perm}^*[n,d] = \inf_{\hat{\pi}} \sup_{M,\pi^*} \mathbb{E}[\| M_{\hat{\pi}^{-1}} - M_{\pi^*^{-1}} \|_F^2] \]

\[ \mathcal{R}_{est}^*[n,d] = \inf_{\hat{M}} \sup_{M,\pi^*} \mathbb{E}[\| \hat{M} - M \|_F^2] \]
Minimax Risk

\[ \mathcal{R}_{perm}^*[n,d] = \inf_{\hat{\pi}} \sup_{M,\pi^*} \mathbb{E}[\|M_{\hat{\pi}^{-1}} - M_{\pi^*^{-1}}\|^2_F] \]

\[ \mathcal{R}_{est}^*[n,d] = \inf_{\hat{M}} \sup_{M,\pi^*} \mathbb{E}[\|\hat{M} - M\|^2_F] \]

Recovering \(\pi^*\) is easier than estimating \(M\)

\[ \mathcal{R}_{perm}^*[n,d] \preceq \mathcal{R}_{est}^*[n,d] \]
Related Rectangular Problems:

- **Two permutations** [Mao et al., 2018, Shah et al., 2019]:

  $M$ is bi-isotonic up to permutations $\pi_1^*$ and $\pi_2^*$ of rows and columns.

  **Objective**: ranking the experts and the questions.
Related Rectangular Problems:

- **Two permutations** [Mao et al., 2018, Shah et al., 2019]:
  \( M \) is bi-isotonic up to permutations \( \pi_1^* \) and \( \pi_2^* \) of rows and columns.
  Objective: ranking the experts and the questions.

- **Column isotony** [Flammarion et al., 2019]
Other ranking and permutations problems

Related Rectangular Problems:

- **Two permutations** [Mao et al., 2018, Shah et al., 2019]: 
  $M$ is bi-isotonic up to permutations $\pi_1^*$ and $\pi_2^*$ of rows and columns.
  **Objective**: ranking the experts and the questions.

- **Column isotony** [Flammarion et al., 2019]

**Ranking Players in a tournament**: $M$ is a $n \times n$ matrix with symmetries.

- **Non-parametric Models SST** [Shah et al., 2016]
Related Rectangular Problems:

- **Two permutations** [Mao et al., 2018, Shah et al., 2019]: $M$ is bi-isotonic up to permutations $\pi_1^*$ and $\pi_2^*$ of rows and columns. **Objective**: ranking the experts and the questions.

- **Column isotony** [Flammarion et al., 2019]

**Ranking Players in a tournament**: $M$ is a $n \times n$ matrix with symmetries.

- **Non-parametric Models** SST [Shah et al., 2016]

- **Parametric Models**:
  - Bradley-Luce-Terry (e.g. [Chen et al., 2019, Chen et al., 2020])
  - Noisy sorting [Braverman and Mossel, 2008]
Other ranking and permutations problems

Related Rectangular Problems:

- **Two permutations** [Mao et al., 2018, Shah et al., 2019]:
  
  \( M \) is bi-isotonic up to permutations \( \pi_1^* \) and \( \pi_2^* \) of rows and columns.

  **Objective**: ranking the experts and the questions.

- **Column isotony** [Flammarion et al., 2019]

Ranking Players in a tournament: \( M \) is a \( n \times n \) matrix with symmetries.

- **Non-parametric Models** SST [Shah et al., 2016]

- **Parametric Models**:
  
  Bradley-Luce-Terry (e.g. [Chen et al., 2019, Chen et al., 2020])

  Noisy sorting [Braverman and Mossel, 2008]

Short story:

- **No computational gap for parametric models** (BLT, noisy sorting)

- mostly unknown for non-parametric models **computational gaps are conjectured**
Main questions

1. Is Estimating $\pi^*$ much easier than estimating $M$?
Main questions

1. Is Estimating $\pi^*$ much easier than estimating $M$?
2. Is there a computational-statistical gap? (as in clustering problems with many groups)
Main questions

1. Is Estimating $\pi^*$ much easier than estimating $M$?
2. Is there a computational-statistical gap? (as in clustering problems with many groups)
3. Is the non-parametric problem intrinsically more challenging than the parametric one?
Main questions

1. Is Estimating $\pi^*$ much easier than estimating $M$?
2. Is there a computational-statistical gap? (as in clustering problems with many groups)
3. Is the non-parametric problem intrinsically more challenging than the parametric one?

Our Results

- Control of $R^*_{perm}(n, d)$
- A polynomial-time procedure achieves $R^*_{perm}(n, d)$
1 Setting and Questions

2 Simple ranking methods

3 Minimax risks and polynomial time algorithm
   • Ingredient 1: Localization of the differences
   • Ingredient 2: PCA and Hierarchical sorting
   • Ingredient 3: Hierarchical Sorting with memory
Analysis of least-squares estimator \[\text{Shah et al., 2016}\]

- \(\Pi_n\) collection of all permutations of \([n]\)
- \(\text{Biso}\) collection all bi-isotonic matrices in \([0, 1]\)

**Least-square estimator**

\[
(\hat{M}^{\text{LS}}, \hat{\pi}^{\text{LS}}) = \arg\min_{\hat{M} \in \text{Biso}, \hat{\pi} \in \Pi_n} (\|\hat{M}\hat{\pi} - Y\|_F^2)
\]
Analysis of least-squares estimator [Shah et al., 2016]

- $\Pi_n$ collection of all permutations of $[n]$
- Biso collection all bi-isotonic matrices in $[0, 1]$

Least-square estimator

$$(\hat{M}^{LS}, \hat{\pi}^{LS}) = \arg\min_{\widetilde{M} \in \text{Biso}, \widetilde{\pi} \in \Pi_n} (\|\widetilde{M}\widetilde{\pi} - Y\|_F^2)$$

Matrix $Y$. 
Analysis of least-squares estimator [Shah et al., 2016]

- $\Pi_n$ collection of all permutations of $[n]$
- $\text{Biso}$ collection all bi-isotonic matrices in $[0, 1]$

**Least-square estimator**

$$\left(\hat{M}^{\text{LS}}, \hat{\pi}^{\text{LS}}\right) = \arg\min_{\hat{M} \in \text{Biso}, \hat{\pi} \in \Pi_n} \left(\|\hat{M}_{\hat{\pi}} - Y\|_F^2\right)$$

Matrix $Y_{\hat{\pi}^{\text{LS}}}$.
Analysis of least-squares estimator [Shah et al., 2016]

- $\Pi_n$ collection of all permutations of $[n]$
- Biso collection all bi-isotonic matrices in $[0, 1]$

Least-square estimator

$$(\hat{M}^{LS}, \hat{\pi}^{LS}) = \arg \min_{\hat{M} \in \text{Biso}, \hat{\pi} \in \Pi_n} \left\{ \| \hat{M}_{\hat{\pi}} - Y \|_F^2 \right\}$$

Matrix $\hat{M}^{LS}_{\hat{\pi}^{LS}}$.

Proposition (e.g. [Shah et al., 2016])

$$\mathbb{E}[\| \hat{M} - M \|_F^2] \lesssim n + (\sqrt{nd} \land nd^{1/3})$$

In this presentation, $\asymp$, $\lesssim$, $\gtrsim$ is up to polylogarithms
Remarks:

- $\hat{M}^{LS}$ is minimax for the estimation loss

\[ R^*_{est} \asymp n \vee (\sqrt{nd} \wedge nd^{1/3}) \]
Remarks:

- $\hat{M}_S$ is minimax for the estimation loss

$$R^*_\text{est} \asymp n \vee (\sqrt{nd} \wedge nd^{1/3}) \,.$$ 

- We have $R^*_\text{perm} \gtrsim n$. 

But the algorithms are not polynomial time.
Remarks:

- \( \hat{M}^{LS} \) is minimax for the estimation loss

\[
\mathcal{R}_{est}^* \approx n \vee (\sqrt{nd} \wedge nd^{1/3})
\]

- We have \( \mathcal{R}_{perm}^* \propto n \).

\begin{tabular}{|c|c|c|c|}
\hline
 & \( n \lesssim d^{1/3} \) & \( d^{1/3} \lesssim n \lesssim d \) & \( d \lesssim n \) \\
\hline
\( \mathcal{R}_{perm}^* \) & \( ? \) & \( ? \) & \( n \) \\
\hline
\( \mathcal{R}_{est}^* \) & \( nd^{1/3} \) & \( \sqrt{nd} \) & \( n \) \\
\hline
\end{tabular}

But the algorithms are not polynomial time.
e.g. [Pananjady and Samworth, 2020, Shah et al., 2019]

**A simple ranking method** :

- For each expert $i$, average performances on all questions:

$$\bar{Y}_i = \frac{1}{d} \sum_{k=1}^{d} Y_{i,k}$$

- Rank experts according to their average : $\hat{\pi}^{av}$
Performances and failures of Global Average

Perfect expert on easy questions VS random expert:

\[ M_{1,.} = (0.5, 0.5 \ldots 0.5, 0.9, 0.9, 0.9, 0.9) ; \quad M_{2,.} = (0.5, 0.5, \ldots, 0.5, 0.5) \]

\[ \sim \sqrt{d} \]
Performances and failures of Global Average

Perfect expert on easy questions VS random expert:

\[ M_{1,.} = (0.5, 0.5 \ldots 0.5, 0.5, 0.9, 0.9, 0.9, 0.9) \quad ; \quad M_{2,.} = (0.5, 0.5, \ldots, 0.5, 0.5) \]

Experts 1 and 2 are not distinguished by \( \overline{Y}_i \sim \text{Risk}(\hat{\pi}^{av}) \asymp \sqrt{d} \)
Performances and failures of Global Average

Perfect expert on easy questions VS random expert:

\[ M_{1,.} = (0.5, 0.5 \ldots 0.5, 0.9, 0.9, 0.9, 0.9) \ ; \ M_{2,.} = (0.5, 0.5, \ldots, 0.5, 0.5) \]

Experts 1 and 2 are not distinguished by \( \overline{Y}_i \sim \text{Risk}(\hat{\pi}^{av}) \sim \sqrt{d} \)

Guarantees on \( \hat{\pi}^{av} \)

\[
\sup_M \mathbb{E}[l(\hat{\pi}^{av}, \pi^*)] \sim n\sqrt{d}
\]
Performances and failures of Global Average

Perfect expert on easy questions VS random expert:

\[ M_{1,.} = (0.5, 0.5 \ldots 0.5, 0.9, 0.9, 0.9, 0.9) \approx \sqrt{d} \]
\[ M_{2,.} = (0.5, 0.5 \ldots, 0.5, 0.5) \]

Experts 1 and 2 are not distinguished by \( \bar{Y}_i \sim \text{Risk}(\hat{\pi}^{av}) \approx \sqrt{d} \)

Guarantees on \( \hat{\pi}^{av} \):

\[
\sup_{M} \mathbb{E}[l(\hat{\pi}^{av}, \pi^{*})] \approx n \sqrt{d}
\]

Sup-optimality of Global average:

- comparisons are not localized (similar phenomenon in tournament problems)
- Furthermore, one-to-one comparisons are not sufficient...

Improvements in [Mao et al., 2018] using local averages on bins.
Localization and Hierarchical clustering \cite{Liu and Moitra, 2020}

\cite{Liu and Moitra, 2020} consider only $d = n$, and provide a poly. time estimator $\hat{\pi}(LM)$

$$\mathbb{E}[l(\hat{\pi}(LM), \pi^*)] \leq n^{1+o(1)}.$$ 

**Optimal** for $d = n$
Localization and Hierarchical clustering [Liu and Moitra, 2020] consider only $d = n$, and provide a poly. time estimator $\hat{\pi}(LM)$. 

$$\mathbb{E} \left[ l(\hat{\pi}(LM), \pi^*) \right] \lesssim n^{1+o(1)}.$$ 

**Optimal** for $d = n$

Localization through change-point detection.

Hierarchical sorting.
### Summary

<table>
<thead>
<tr>
<th>Metric</th>
<th>( n \leq d^{1/3} )</th>
<th>( d^{1/3} \leq n \leq d )</th>
<th>( d \leq n )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( R^*_{\text{perm}} )</td>
<td>???</td>
<td>???</td>
<td>( n )</td>
</tr>
<tr>
<td>( R^*_{\text{est}} )</td>
<td>( nd^{1/3} )</td>
<td>( \sqrt{nd} )</td>
<td>( n )</td>
</tr>
</tbody>
</table>

- **Global average (UB)**: \( n \sqrt{d} \), \( n \sqrt{d} \), \( n \sqrt{d} \)
- **Ext. of [Liu and Moitra, 2020] (UB)**: \( d \), \( d \), \( n \)

**Remarks:**
- Polynomial method of [Liu and Moitra, 2020] minimax for \( d = n \)
- Known UB for rates in \( R^*_{\text{perm}} \) and \( R^*_{\text{est}} \) not in polynomial time.
Summary

<table>
<thead>
<tr>
<th></th>
<th>$n \leq d^{1/3}$</th>
<th>$d^{1/3} \leq n \leq d$</th>
<th>$d \leq n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R^*_{perm}$</td>
<td>??</td>
<td>??</td>
<td>$n$</td>
</tr>
<tr>
<td>$R^*_{est}$</td>
<td>$nd^{1/3}$</td>
<td>$\sqrt{nd}$</td>
<td>$n$</td>
</tr>
<tr>
<td>Global average (UB)</td>
<td>$n\sqrt{d}$</td>
<td>$n\sqrt{d}$</td>
<td>$n\sqrt{d}$</td>
</tr>
<tr>
<td>Ext. of [Liu and Moitra, 2020] (UB)</td>
<td>$d$</td>
<td>$d$</td>
<td>$n$</td>
</tr>
</tbody>
</table>

Remarks:
- Poly. time method of [Liu and Moitra, 2020] minimax for $d = n$
- Known UB for rates in $R^*_{est}$ and $R^*_{perm}$ not in polynomial time.
1 Setting and Questions

2 Simple ranking methods

3 Minimax risks and polynomial time algorithm
   - Ingredient 1: Localization of the differences
   - Ingredient 2: PCA and Hierarchical sorting
   - Ingredient 3: Hierarchical Sorting with memory
Idealized setting: (as in [Liu and Moitra, 2020])
polylog independent full samples $Y^{(1)} = M + E^{(1)}$, $Y^{(2)} = M + E^{(2)}$, \ldots
Minimax risks and polynomial time algorithm

**Idealized setting**: (as in [Liu and Moitra, 2020]) polylog independent full samples $Y^{(1)} = M + E^{(1)}$, $Y^{(2)} = M + E^{(2)}$, ...

**Theorem [Pilliat, Carpentier, V., 2022]**

There exists a estimator $\hat{\pi}$ of $\pi^*$ which is **poly. time** and **minimax optimal**

$$
\mathbb{E}[l(\hat{\pi}, \pi^*)] \lesssim n + \left(n^{3/4} d^{1/4} \wedge nd^{1/6}\right) \asymp R_{perm}^*.
$$
Idealized setting: (as in [Liu and Moitra, 2020])

Polylog independent full samples $Y^{(1)} = M + E^{(1)}$, $Y^{(2)} = M + E^{(2)}$, ...

**Theorem [Pilliat, Carpentier, V., 2022]**

There exists an estimator $\hat{\pi}$ of $\pi^*$ which is **poly. time** and **minimax optimal**

$$
\mathbb{E}[l(\hat{\pi}, \pi^*)] \lesssim n + (n^{3/4} d^{1/4} \wedge n d^{1/6}) \times R_{\text{perm}}^* .
$$

<table>
<thead>
<tr>
<th></th>
<th>$n \leq d^{1/3}$</th>
<th>$d^{1/3} \leq n \leq d$</th>
<th>$d \leq n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_{\text{perm}}^*$</td>
<td>$n d^{1/6}$</td>
<td>$n^{3/4} d^{1/4}$</td>
<td>$n$</td>
</tr>
<tr>
<td>$R_{\text{est}}^*$</td>
<td>$n d^{1/3}$</td>
<td>$\sqrt{nd}$</td>
<td>$n$</td>
</tr>
</tbody>
</table>
Idealized setting: (as in [Liu and Moitra, 2020])

Polylog independent full samples \( Y^{(1)} = M + E^{(1)}, Y^{(2)} = M + E^{(2)}, \ldots \)

Theorem [Pilliat, Carpentier, V., 2022]

There exists a estimator \( \hat{\pi} \) of \( \pi^* \) which is **poly. time** and **minimax optimal**

\[
\mathbb{E}[l(\hat{\pi}, \pi^*)] \preceq n + (n^{3/4}d^{1/4} \wedge nd^{1/6}) \approx \mathcal{R}_{perm}^* .
\]

<table>
<thead>
<tr>
<th>( \mathcal{R}_{perm}^* )</th>
<th>( n \preceq d^{1/3} )</th>
<th>( d^{1/3} \preceq n \preceq d )</th>
<th>( d \preceq n )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \mathcal{R}_{est}^* )</td>
<td>( nd^{1/6} )</td>
<td>( n^{3/4}d^{1/4} )</td>
<td>( n )</td>
</tr>
</tbody>
</table>

Consequence: Optimal estimation rate of \( M \) achievable in polynomial time.
From global to local averages

Global average good.

If $M_1, M_2$ not isotonic or unbounded undistinguishable if $\| M_{1,.} - M_{2,.} \|_2^2 \lesssim \sqrt{d}$.

Global average bad $\rightarrow$ localize.
From global to local averages

Global average good.

Global average bad $\rightarrow$ **localize**.

If $M_1, M_2$ not isotonic or unbounded undistinguishable if $\| M_1, - M_2, \|_2^{\frac{2}{2}} \lesssim \sqrt{d}$.

**Idea:**
- Local difference between experts $\sim$ a high-variation signature
- Variation signatures detectable at larger scale
From global to local averages

Global average good.

Global average bad $\Rightarrow$ localize.

If $M_1$, $M_2$ not isotonic or unbounded undistinguishable if $\|M_1, - M_2,\|_2^2 \lesssim \sqrt{d}$.

**Idea:**
- Local difference between experts $\sim$ a high-variation signature
- Variation signatures detectable at larger scale

**Procedure**
- Localize areas where any of the two experts varies by more than $h$...
- ... and compute local averages.
Algorithm

CUSUM Statistic:
\[ C_{l,r} = \frac{1}{r} \left( \sum_{k=l}^{l+r-1} Y_{1,k} - \sum_{k=l-r}^{l-1} Y_{1,k} \right) \]

Pick height \( h > 0 \) and scale \( r > 0 \):

**Step 1** High-Variation Detection \( C_{l,r} \geq h \)

\[ S_{r,h} = \bigcup \{[l - r, l + r) : C_{l,r} \geq h \} \]

**Step 2** Localized comparison

\[ \Psi(S_{r,h}) = \frac{1}{\sqrt{|S_{r,h}|}} \sum_{k \in S_{r,h}} (Y_{2,k} - Y_{1,k}) \]
Algorithm

CUSUM Statistic:
\[ C_{l,r} = \frac{1}{r} \left( \sum_{k=l}^{l+r-1} Y_{1,k} - \sum_{k=l-r}^{l-1} Y_{1,k} \right) \]

Pick height \( h > 0 \) and scale \( r > 0 \):

**Step 1** High-Variation Detection \( C_{l,r} \geq h \)

\[ S_{r,h} = \bigcup \{ [l-r, l+r) : C_{l,r} \geq h \} \]

**Step 2** Localized comparison
\[ \Psi(S_{r,h}) = \frac{1}{\sqrt{|S_{r,h}|}} \sum_{k \in S_{r,h}} (Y_{2,k} - Y_{1,k}) \]

Proposition

*Whp valid comparison if* \[ \| M_{1,.} - M_{2,.} \|_2^2 \geq d^{1/6} \]

*Conversely, optimal for* \( n = 2 \).
## Summary

<table>
<thead>
<tr>
<th></th>
<th>$n \leq d^{1/3}$</th>
<th>$d^{1/3} \leq n \leq d$</th>
<th>$d \leq n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_{\text{perm}}^*$</td>
<td>$nd^{1/6}$</td>
<td>$n^{3/4}d^{1/4}$</td>
<td>$n$</td>
</tr>
<tr>
<td>$R_{\text{est}}^*$</td>
<td>$nd^{1/3}$</td>
<td>$\sqrt{nd}$</td>
<td>$n$</td>
</tr>
<tr>
<td><strong>Global average (UB)</strong></td>
<td>$n\sqrt{d}$</td>
<td>$n\sqrt{d}$</td>
<td>$n\sqrt{d}$</td>
</tr>
<tr>
<td><strong>Ext. of [Liu and Moitra, 2020] (UB)</strong></td>
<td>$d$</td>
<td>$d$</td>
<td>$n$</td>
</tr>
</tbody>
</table>
1 Setting and Questions

2 Simple ranking methods

3 Minimax risks and polynomial time algorithm
   - Ingredient 1: Localization of the differences
   - Ingredient 2: PCA and Hierarchical sorting
   - Ingredient 3: Hierarchical Sorting with memory
Hierarchical Sorting Tree

Start from the complete set \([n]\) of experts
Hierarchical Sorting Tree

Start from the complete set \([n]\) of experts

Build a **Trisection** \((O, P, I)\) of this set where:

1. Experts in \(O\) are whp among the \(n/2\) worst
2. Experts in \(I\) are whp among the \(n/2\) best
3. Experts in \(P\) are undecided
Start from the complete set $[n]$ of experts

Build a **Trisection** $(O, P, I)$ of this set where:

1. Experts in $O$ are whp among the $\frac{n}{2}$ worst
2. Experts in $I$ are whp among the $\frac{n}{2}$ best
3. Experts in $P$ are undecided

\ldots iterate on $O$, $P$, $I$ with fresh samples

$\sim$ ordered partition of $[n]$
$\sim$ Random partition $\tilde{\pi}$
Hierarchical Sorting Tree

Start from the complete set \([n]\) of experts

Build a **Trisection** \((O, P, I)\) of this set where:

1. Experts in \(O\) are whp among the \(n/2\) worst
2. Experts in \(I\) are whp among the \(n/2\) best
3. Experts in \(P\) are undecided

... Iterate on \(O, P, I\) with fresh samples

\[ \leadsto \text{ordered partition of } [n] \]
\[ \leadsto \text{Random partition } \widehat{\pi} \]

**Lemma**

\[ l(\widehat{\pi}, \pi^*) \leq \sum_{\overline{P}} \| M(\overline{P}) - \overline{M}(\overline{P}) \|_F^2, \]

where \( \overline{P} \supset P \) (slightly larger set)
Partitioning a group $G$ into three blocks

General Strategy:

$O = \emptyset$; $I = \emptyset$

For all heights $h$, scales $r$.

1. **Dimension Reduction**

   $\sim$ high-variation regions $h$ of mean expert at scale $r$
Partitioning a group $G$ into three blocks

**General Strategy:**

$O = \emptyset; I = \emptyset$

For all heights $h$, scales $r$.

1. **Dimension Reduction**
   - $\sim$ high-variation regions $h$ of mean expert at scale $r$

2. **Estimate a direction** $\omega \in \mathbb{R}^d$
Partitioning a group $G$ into three blocks

**General Strategy**:

1. **Dimension Reduction**
   - high-variation regions $h$ of mean expert at scale $r$

2. **Estimate a direction** $\omega \in \mathbb{R}^d$

3. **Expert comparison** by weighted average $\sum_k Y_{i,k} \omega_k$
   - $(L, U) \subset G$

   - $G \leftarrow G \setminus (U \cup L)$; $O \leftarrow O \cup L$; $I \leftarrow I \cup U$

$O = \emptyset$; $I = \emptyset$

For all heights $h$, scales $r$. 

$M_{i,k}$

1

0

question $k$

$d$
Partitionning a group \( G \) into three blocks

**General Strategy:**

1. \( O = \emptyset; \ I = \emptyset \)
   
   For all heights \( h \), scales \( r \).

2. **Dimension Reduction**
   
   \( \sim \) high-variation regions \( h \) of mean expert at scale \( r \)

3. **Estimate a direction** \( \omega \in \mathbb{R}^d \)

4. **Expert comparison by weighted average** \( \sum_k Y_{i,k} \omega_k \)

5. \( (L, U) \subset G \)

6. \( G \leftarrow G \setminus (U \cup L); \ O \leftarrow O \cup L; \ I \leftarrow I \cup U \)

**Iterate Polylog times**
Partitioning a group $G$ into three blocks

**General Strategy:**

$O = \emptyset; \ I = \emptyset$

For all heights $h$, scales $r$.

1. Estimate a direction $\omega \in \mathbb{R}^d$

Iterate $\text{Polylog times}$
Toy example (with two pure subgroups)

Variation detection:

Aggregation:

Remark: For these two groups of experts Ranking = Clustering

Direction $\omega$ selection: right singular vector + left singular vector + image thresholding + correction (≠ [Liu and Moitra, 2020])
Toy example (with two pure subgroups)

**Variation detection**:  
\[ \sim \text{keeping windows of size } r \text{ with variation } h \]
Toy example (with two pure subgroups)

Variation detection:
\[ \sim \text{keeping windows of size } r \text{ with variation } h \]

Aggregation:
\[ \sim \text{rescaled sum of observations on each window} : \]

\[
\frac{1}{2} \sqrt{r} h \times \begin{pmatrix}
0 & -1 & -1 & 0 & 0 & -1 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & -1 & -1 & 0 & 0 & -1 & 0 & 0 \\
0 & -1 & -1 & 0 & 0 & -1 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 1 & 0 & 0
\end{pmatrix}
\]

Remark: For these two groups of experts Ranking = Clustering

Direction \( \omega \) selection:
- right singular vector
- left singular vector + image thresholding + correction

\[ (\neq [\text{Liu and Moitra, 2020}]) \]
**Toy example (with two pure subgroups)**

**Variation detection:**
\[ \sim \text{keeping windows of size } r \text{ with variation } h \]

**Aggregation:**
\[ \sim \text{rescaled sum of observations on each window :} \]
\[
\frac{1}{2} \sqrt{r} h \times \left( \begin{array}{cccccccc}
0 & -1 & -1 & 0 & 0 & -1 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & -1 & -1 & 0 & 0 & -1 & 0 & 0 \\
0 & -1 & -1 & 0 & 0 & -1 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
\end{array} \right)
\]

**Remark:** For these two groups of experts

- Ranking = Clustering
- PCA outperforms row sums for large groups (to select active regions).
Toy example (with two pure subgroups)

Variation detection:
~ keeping windows of size $r$ with variation $h$

Aggregation:
~ rescaled sum of observations on each window:

$$\frac{1}{2} \sqrt{r} h \times \begin{pmatrix} 0 & -1 & -1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & -1 & 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & -1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Remark: For these two groups of experts

- Ranking = Clustering
- PCA outperforms row sums for large groups (to select active regions).

Direction $\omega$ selection: right singular vector
Toy example (with two pure subgroups)

**Variation detection**:  
\( \sim \) keeping windows of size \( r \) with variation \( h \)

**Aggregation**:  
\( \sim \) rescaled sum of observations on each window:

\[
\frac{1}{2} \sqrt{rh} \times \begin{pmatrix}
0 & -1 & -1 & 0 & 0 & -1 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & -1 & -1 & 0 & 0 & -1 & 0 & 0 \\
0 & -1 & -1 & 0 & 0 & -1 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 1 & 0 & 0
\end{pmatrix}
\]

**Remark**: For these two groups of experts

- Ranking = Clustering
- PCA outperforms row sums for large groups (to select active regions).

**Direction \( \omega \) selection**: right singular vector  
left singular vector + image thresholding + correction  
\((\neq [\text{Liu and Moitra, 2020}])\)
Suboptimality of the procedure

<table>
<thead>
<tr>
<th>Procedure</th>
<th>( n \leq d^{1/3} )</th>
<th>( d^{1/3} \leq n \leq d )</th>
<th>( d \leq n )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( R_{perm}^* )</td>
<td>( nd^{1/6} )</td>
<td>( n^{3/4}d^{1/4} )</td>
<td>( n )</td>
</tr>
<tr>
<td>( R_{est}^* )</td>
<td>( nd^{1/3} )</td>
<td>( \sqrt{nd} )</td>
<td>( n )</td>
</tr>
</tbody>
</table>

Ext. of [Liu and Moitra, 2020] (UB)

\[(\text{modified}) \text{ PCA+ Hierarchy}\]

<table>
<thead>
<tr>
<th></th>
<th>( d )</th>
<th>( d )</th>
<th>( n )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( nd^{1/6} )</td>
<td>( n^{2/3}d^{1/3} )</td>
<td>( n )</td>
<td></td>
</tr>
</tbody>
</table>
Suboptimality of the procedure

<table>
<thead>
<tr>
<th></th>
<th>$n \leq d^{1/3}$</th>
<th>$d^{1/3} \leq n \leq d$</th>
<th>$d \leq n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathcal{R}_{perm}^*$</td>
<td>$n d^{1/6}$</td>
<td>$n^{3/4} d^{1/4}$</td>
<td>$n$</td>
</tr>
<tr>
<td>$\mathcal{R}_{est}^*$</td>
<td>$n d^{1/3}$</td>
<td>$\sqrt{n d}$</td>
<td>$n$</td>
</tr>
</tbody>
</table>

**Ext. of [Liu and Moitra, 2020] (UB)**

<table>
<thead>
<tr>
<th></th>
<th>$d$</th>
<th>$d$</th>
<th>$n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>(modified) PCA+ Hierarchy</td>
<td>$n d^{1/6}$</td>
<td>$n^{2/3} d^{1/3}$</td>
<td>$n$</td>
</tr>
</tbody>
</table>

**Benefits of hierarchical Sorting:**

- Allows to localize the differences between subgroup of experts
- Builds upon large groups of close experts
## Suboptimality of the procedure

<table>
<thead>
<tr>
<th></th>
<th>$n \lesssim d^{1/3}$</th>
<th>$d^{1/3} \lesssim n \lesssim d$</th>
<th>$d \lesssim n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathcal{R}_{perm}$</td>
<td>$nd^{1/6}$</td>
<td>$n^{3/4}d^{1/4}$</td>
<td>$n$</td>
</tr>
<tr>
<td>$\mathcal{R}_{est}$</td>
<td>$nd^{1/3}$</td>
<td>$\sqrt{nd}$</td>
<td>$n$</td>
</tr>
</tbody>
</table>

Ext. of [Liu and Moitra, 2020] (UB)  
(modified) PCA+ Hierarchy

### Benefits of hierarchical Sorting:
- Allows to localize the differences between subgroup of experts
- Builds upon large groups of close experts
- ... but **oblivious** of previous structure found in the data

> Hierarchical Sorting with Memory which is optimal.
1 Setting and Questions

2 Simple ranking methods

3 Minimax risks and polynomial time algorithm
   - Ingredient 1 : Localization of the differences
   - Ingredient 2 : PCA and Hierarchical sorting
   - Ingredient 3 : Hierarchical Sorting with memory
Each line $M_{i,k}$ represents an expert $i$. 
Our vanilla dimension reduction techniques:
Detection of variations of the mean expert in $G$
Which information is brought by the tree?

**Our vanilla dimension reduction techniques:**
Detection of variations of the mean expert in \( G \) ... but ...

- A large scale \( r \) is needed if \(|G|\) is small.
- Spurious regions are detected (those where the width of \( G \) is small).
Which information is brought by the tree?

Our vanilla dimension reduction techniques:
Detection of variations of the mean expert in $G$ ...

- A large scale $r$ is needed if $|G|$ is small.
- Spurious regions are detected (those where the width of $G$ is small).

Idea:
Using the partial ordering to:
- decrease the variance of the CUSUM (with $\mathcal{V} \supset G$ experts)
- Estimate the width $\Delta$ of $G$

$$\Delta_k = \max_{i \in G} M_{i,k} - \max_{i \notin G} M_{i,k}$$
of $G$ by comparing mean experts in groups above and below $G$. 

\[ \begin{array}{c}
\text{subset } U^* \text{ above} \\
\text{subset } L^- \text{ below} \\
\text{mean } \overline{\mathcal{P}} \text{ of } \mathcal{P}
\end{array} \]
In practice

Fix a height $h$, and a scale $r$ (possibly too small for $G$). Consider expert sets $\mathcal{V}^+$ above $G$ and $\mathcal{V}^-$ below $G$.

Simultaneously check:

1. If variations at scale $r$ higher than $h$

$$\hat{C}_{k,r}^{(ext)} = \frac{1}{r} \sum_{l=k+1}^{k+r} \hat{y}_l (\mathcal{V}^+ \cup \mathcal{V}^-) - \sum_{l=k+1}^{k+r} \hat{y}_l (\mathcal{V}^+ \cup \mathcal{V}^-)$$
In practice

Fix a height $h$, and a scale $r$ (possibly too small for $G$). Consider expert sets $\mathcal{V}^+$ above $G$ and $\mathcal{V}^-$ below $G$

Simultaneously check:

1. If variations at scale $r$ higher than $h$
   \[ \hat{C}_{k,r}^{(ext)} = \frac{1}{r} \sum_{l=k+1}^{k+r} y_l (\mathcal{V}^+ \cup \mathcal{V}^-) - \sum_{l=k+1}^{k+r} \bar{y}_l (\mathcal{V}^+ \cup \mathcal{V}^-) \]

2. If the width of $G$ at scale $\frac{r}{2}$ higher than $h$.
   \[ \hat{\Delta}_{k,r}^{(ext)} = \frac{1}{r} \sum_{l=k-r}^{k+r} \bar{y}_l (\mathcal{V}^+) - \bar{y}_l (\mathcal{V}^-) \]
Main result

Estimator $\hat{\pi}^{WM}$ with this new **dimension reduction** step

**Theorem**

$$\text{Max-Perm}(\hat{\pi})^{WM} \leq \left[ nd^{1/6} \wedge (n^{3/4}d^{1/4}) \right] + n \asymp \text{MiniMax-Perm}$$
Main result

Estimator $\hat{\pi}^{WM}$ with this new **dimension reduction** step

**Theorem**

$$\text{Max-Perm}(\hat{\pi})^{WM} \lesssim \left[ nd^{1/6} \wedge (n^{3/4} d^{1/4}) \right] + n \asymp \text{MiniMax-Perm}$$

<table>
<thead>
<tr>
<th>$R^*_\text{perm}$</th>
<th>$n \lesssim d^{1/3}$</th>
<th>$d^{1/3} \lesssim n \lesssim d$</th>
<th>$d \lesssim n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{Ext. of [Liu and Moitra, 2020] (UB)}$</td>
<td>$nd^{1/6}$</td>
<td>$n^{3/4} d^{1/4}$</td>
<td>$n$</td>
</tr>
<tr>
<td>$R^*_\text{est}$</td>
<td>$nd^{1/3}$</td>
<td>$\sqrt{nd}$</td>
<td>$n$</td>
</tr>
</tbody>
</table>

(modified) PCA+ Hierarchy+Memory

$nd^{1/6}$ | $n^{3/4} d^{1/4}$ | $n$ |
Main result

Estimator $\hat{\pi}^{WM}$ with this new **dimension reduction** step

**Theorem**

$$\text{Max-Perm}(\hat{\pi})^{WM} \lesssim \left[ nd^{1/6} \land (n^{3/4}d^{1/4}) \right] + n \asymp \text{MiniMax-Perm}$$

<table>
<thead>
<tr>
<th></th>
<th>$n \lesssim d^{1/3}$</th>
<th>$d^{1/3} \lesssim n \lesssim d$</th>
<th>$d \lesssim n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R^*_{perm}$</td>
<td>$nd^{1/6}$</td>
<td>$n^{3/4}d^{1/4}$</td>
<td>$n$</td>
</tr>
<tr>
<td>$R^*_{est}$</td>
<td>$nd^{1/3}$</td>
<td>$\sqrt{nd}$</td>
<td>$n$</td>
</tr>
</tbody>
</table>

Ext. of [Liu and Moitra, 2020] (UB)  
(modified) PCA+ Hierarchy+Memory  

$\sim$ As a corollary, minimax polynomial-time estimator of $M$. 
No **computational gap** for this ranking (and estimation) problem.

In comparison to $n = d$, rectangular setting requires **new ideas**:
- side information from partial ranking.

Results extend to **partial observations** and **general noise** levels.
No computational gap for this ranking (and estimation) problem.

In comparison to $n = d$, rectangular setting requires new ideas:
- Use side information from partial ranking.

Results extend to partial observations and general noise levels.

For two permutations, existence of a computational gap is not clear.


