The Grouped Two-Sided Orthogonal Procrustes Problem

Bryan Conroy, Peter Ramadge
Presented: by Alex Lorbert
Dept. of Electrical Engineering, Princeton University
ICASSP 2011
May 25, 2011
Introduction

- Given matrices $A, B$ the **orthogonal Procrustes problem** seeks an orthogonal matrix that satisfies

$$R^* = \arg \min_{R \in O(n)} \|AR - B\|^2_f$$

- Useful as a geometric transformation to correct for rotational distortions between two datasets

- This paper considers a constrained version of the **Two-sided orthogonal Procrustes problem**: Given symmetric matrices $C, C_r$, find an orthogonal matrix that satisfies:

$$R^* = \arg \min_{R \in O(n)} \| R^T C R - C_r \|^2_f$$
Applications

- Covariance (or correlation) matrix matching

- Graph matching problems
  - $C, C_r$ are adjacency matrices of weighted graphs
  - Goal: identify permutation of the vertices of one graph that best matches the adjacency matrices:
    \[
    P^* = \arg \min_P ||P^T C P - C_r||_f^2
    \]
    Hard problem: combinatorial

- Umeyama’s method:
  1. Relax permutation constraint to an orthogonal transformation constraint (two-sided Procrustes)
  2. Find closest permutation matrix to the orthogonal matrix by the Hungarian algorithm
Orthogonal Procrustes Solutions

• For orthogonal Procrustes problem,
  \[ R^* = \arg \min_{R \in O(n)} \|AR - B\|_f^2 \]
  ▫ Solution obtained by SVD of \( A^TB \)

• For two-sided orthogonal Procrustes problem,
  \[ R^* = \arg \min_{R \in O(n)} \|R^TCR - C_r\|_f^2 \]
  ▫ Family of solutions may be expressed in terms of the eigendecompositions:
    \[ C = V \Lambda V^T \]
    \[ C_r = V_r \Lambda_r V_r^T \]
    \[ R^* = V D V_r^T \]
  ▫ where D is a any diagonal matrix with diagonal entries in \( \{+1, -1\} \)
Grouped Two-Sided Procrustes

- Graph vertices are sometimes attributed with side information that facilitate solving the two-sided Procrustes problem
- *This paper* considers the case when the vertices can be parcellated into groups

Examples:
- When the graph represents an object, vertices can be grouped based on part label
- When there is a spatial configuration to the vertices, a grouping may be derived based on a region-based segmentation
Problem Formulation

- Assume each dataset is parcellated into $M$ groups
  - For each point $i$, let $g(i) \in \{1, \ldots, M\}$ denote the group it belongs to

- Goal: estimate $M$ orthogonal transformations (one for each group) that satisfy:

$$R^* = \arg \min_{R=\text{diag}(R_1, \ldots, R_M)} \|R^T C R - C_r\|_F^2$$

- Denote this as the **grouped** two-sided orthogonal Procrustes problem
  - Reduces to the standard two-sided problem when $M=1$
Problem Formulation

- Grouped version is a constrained two-sided problem
  - Subject to an additional set of linear constraints:
    \[ [R]_{pq} = 0 \text{ if } g(p) \neq g(q) \]

- Benefits of the grouping structure:
  - Constraints serve to regularize the optimization
  - The grouping structure imposes prior knowledge on the problem
  - Reduction in the number of estimation parameters
Solution Method Initialization

- For $M > 1$, an iterative solution must be pursued.
- Let: $C_{ii}, C_{ii}^r$ denote the within-region similarity structures.
- $C_{ij}, C_{ij}^r$ denote the across-region similarity structures.

- A logical initialization is to only consider within-region similarities.
  - Problem decouples into $M$ two-sided Procrustes problems:
    \[ R_m = V_{mm} D_m V_{mm}^r T \]
  - Where $C_{mm} = V_{mm} \Sigma_{mm} V_{mm}^T$ and $C_{mm}^r = V_{mm}^r \Sigma_{mm}^r V_{mm}^r T$.
Solution Method

• Goal: Improve on initial estimates by incorporating across-region similarities

• Our proposed algorithm is a two-step process:
  1. Solve for \((D_1, \ldots, D_M)\) by matching across-region similarities
     - Diagonal entries \((\{+1, -1\})\) can be approximated by MAXCUT
  2. Fine-tune initial orthogonal estimates by a greedy selection strategy
     - Updates are progressively built up as a product of Givens rotations
     - Can be seen as 2x2 block coordinate descent
Solution Method (Step 2)

• **Model assumption:** observed $C$ is a perturbed version of the template:

$$C = RC_r R^T + E$$

• Due to noise, the eigenvectors of the observed within-region similarities $C_{mm}$ are perturbed versions of the true eigenvectors

• Modify the estimate for group $m$ by:

$$R_m = V_{mm} U_m D_m V_{mm}^r T$$

• where $V_{mm} U_m$ serves as an improved estimate for the eigenvectors of $C_{mm}$
Solution Method (Step 2)

- Let $U = \text{diag}(U_1, \ldots, U_M)$
- $U$ is estimated by matching both the within-region and the across-region similarities
- **Greedy approach**: model $U$ at iteration $t$ as:
  \[ U_t = U_{t-1} U_{p_t q_t} (\theta_{p_t q_t}) \]

  Givens rotation matrix that rotates $p_t, q_t$ through an angle $\theta_t$

- For $t = 1, 2, ...$
  - Choose $p_t, q_t$ (such that $g(p_t) = g(q_t)$) by a greedy selection strategy
  - Select the rotation angle $\theta_t$ by a line search that maximizes decrease in objective
Results

• Algorithm was tested on 2 datasets:
  ▫ MNIST handwritten digits (parcellated by digit)
  ▫ YaleB face database (parcellated by identity)
• Each group $m$ was transformed by a random orthogonal matrix $A_m$ and corrupted by additive noise
Results

- Accuracy measure between true and estimated transformations:

\[
\text{Accuracy} = \frac{1}{N} \sum_{m=1}^{M} \text{trace}(\hat{A}_m^T A_m) \quad (N=\text{total # datapoints})
\]

Improvement over iterations shows the usefulness of across-region similarities

Results on MNIST

Results on YaleB
Results (Graph matching problem)

- Each group was transformed by a random permutation and perturbed with noise
- Hungarian algorithm was run to estimate optimal permutation matrix from estimated orthogonal transformation

Results on MNIST

Results on YaleB
Conclusion

- We posed the grouped version of the two-sided orthogonal Procrustes problem.
- Proposed an algorithm that is both computationally efficient and simple to implement.
- Results of the algorithm on simulations show that the algorithm effectively utilizes the grouping structure to greatly improve estimation accuracy in the presence of noise.
Thanks!
(Extra slides)
## Results (Compared with Umeyama)

<table>
<thead>
<tr>
<th>SNR</th>
<th>Permutation Accuracy (%) - MNIST</th>
<th>Permutation Accuracy (%) - YaleB</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Umeyama's Method</td>
<td>Proposed Algorithm</td>
</tr>
<tr>
<td>10dB</td>
<td>32.8%</td>
<td>97.6%</td>
</tr>
<tr>
<td>6dB</td>
<td>13.5%</td>
<td>94.5%</td>
</tr>
<tr>
<td>2dB</td>
<td>7.2%</td>
<td>83.4%</td>
</tr>
<tr>
<td>-2dB</td>
<td>4.1%</td>
<td>46.5%</td>
</tr>
</tbody>
</table>