Ordering Objects

What Heuristic Should We Use?

Michael Hahsler
Dept. of Engineering Management, Information, and Systems
SMU

November 4, 2015
INFORMS Annual Meeting 2015

Introduction and Motivation

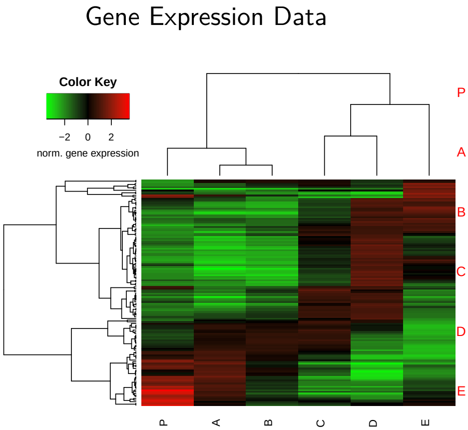
Introduction

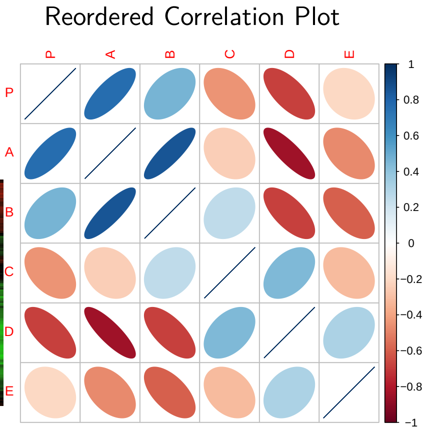
- Part of **Combinatorial Data Analysis** (P. Arabie and Hubert 1996)
- Ordering objects to reveal structural information
- Related to ranking

Some Applications

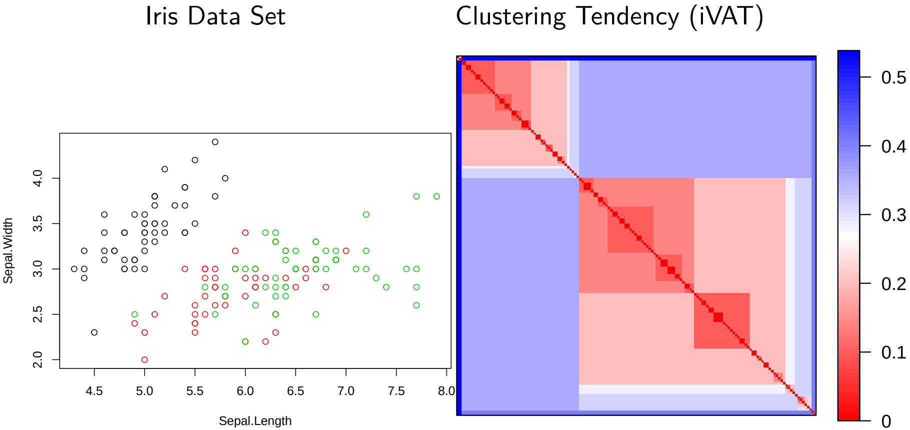
- **Sociology:** Find group structure in sociograms
- **Psychology:** Order subject-by-item response matrix
- **Ecology:** Analyze plant associations
- **Manufacturing:** Product flow analysis
- **Biology:**
  - Arrange gene expression data (heat maps)
  - Gene sequencing, read assembly using the consecutive ones problem (C1P)
- **Visualization:**
  - Reorder large data tables
  - Cluster tendency (visual assessment of cluster tendency, VAT)
  - Evaluation of clustering quality (dissimilarity plot)

Examples


Gene Expression Data
Color Key
−2 0 2
norm. gene expression
P
A
B
C
D
E


Reordered Correlation Plot
P
A
B
C
D
E
1
0.8
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−0.8
−1

Examples


Iris Data Set
Clustering Tendency (iVAT)
Sepal.Width
Sepal.Length
4.0
3.5
3.0
2.5
2.0
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
0.5
0.4
0.3
0.2
0.1
0

Motivation

Many methods to order objects have been proposed.

How should we do it?

Seriation

Problem Definition

- Arrange a set of n objects

$$\mathcal{O} = \{O_1, O_2, \ldots, O_n\}$$

  in a **linear order** given available data and some loss function in order to **reveal structural information.**
- **Data:** A symmetric dissimilarity matrix $\mathbf{D} = [d_{ij}]_{n \times n}$, where d_{ij} represents the dissimilarity between O_i and O_j, and $d_{ii} = 0$ for all i.
- **Linear order:** A permutation function $\Psi(\mathbf{D}) = P_\pi \mathbf{D} P_\pi^T$, where π is the permutation vector and P_π is the permutation matrix for Ψ.

Problem Definition

- Arrange a set of n objects

$$\mathcal{O} = \{O_1, O_2, \dots, O_n\}$$

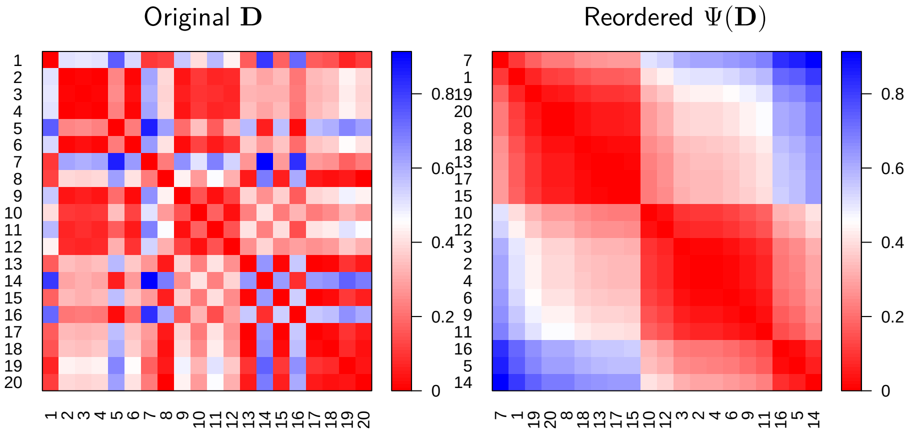
  in a **linear order** given available data and some loss function in order to **reveal structural information.**
- **Data:** A symmetric dissimilarity matrix $\mathbf{D} = [d_{ij}]_{n \times n}$, where d_{ij} represents the dissimilarity between O_i and O_j, and $d_{ii} = 0$ for all i.
- **Linear order:** A permutation function $\Psi(\mathbf{D}) = P_\pi \mathbf{D} P_\pi^T$, where π is the permutation vector and P_π is the permutation matrix for Ψ.

Optimization problem

$$\Psi^* = \operatorname*{argmin}_{\Psi \in \mathcal{P}_n} L(\Psi(\mathbf{D})),$$

where L is a loss function to evaluate how well a given permutation reveals structural information. $\mathcal{P}_n$ is the set of all possible permutation functions.

Example of Seriation



Typical assumption: Structural information is revealed if more similar objects are presented closer together.

Issues and Techniques

Issues

- How do we define the loss function L?
- Requires to solve a discrete optimization problem with solution space size of $O(n!)$.

Issues and Techniques

Issues

- How do we define the loss function L?
- Requires to solve a discrete optimization problem with solution space size of $O(n!)$.

Techniques

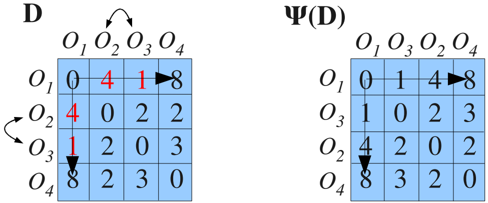
- Partial enumeration methods (currently solve problems with $n \leq 40$):
  - dynamic programming
  - branch-and-bound
- Other discrete optimization methods
  - QAP formulation
  - spectral methods
- Heuristics for larger problems

Loss Functions

Column/Row Gradient Measures

Perfect anti-Robinson matrix (Robinson 1951): A symmetric matrix where the values in all rows and columns only increase when moving away from the main diagonal. Gradient conditions (L. Hubert, Arabie, and Meulman 1987):

$$\text{within rows:} \quad d_{ik} \leq d_{ij} \quad \text{for} \quad 1 \leq i < k < j \leq n;$$
$$\text{within columns:} \quad d_{kj} \leq d_{ij} \quad \text{for} \quad 1 \leq i < k < j \leq n.$$



The closer objects are together in the order of the matrix, the higher their similarity.

Note: Most matrices can only be brought into a near anti-Robinson form.

Column/Row Gradient Measures (cont.)

Gradient measure (quantifies the divergence from anti-Robinson form)

$$L(\mathbf{D}) = \sum_{i<k<j} f(d_{ik}, d_{ij}) + \sum_{i<k<j} f(d_{kj}, d_{ij})$$

where $f(\cdot,\cdot)$ is a function which defines how a violation or satisfaction of a gradient condition for an object triple (O_i, O_k and O_j) is counted.

Column/Row Gradient Measures (cont.)

Gradient measure (quantifies the divergence from anti-Robinson form)

$$L(\mathbf{D}) = \sum_{i<k<j} f(d_{ik}, d_{ij}) + \sum_{i<k<j} f(d_{kj}, d_{ij})$$

where $f(\cdot, \cdot)$ is a function which defines how a violation or satisfaction of a gradient condition for an object triple (O_i, O_k and O_j) is counted.

Raw number of violations minus satisfactions:

$$f(z,y) = \mathrm{sign}(y-z) = \begin{cases} -1 & \text{if} \quad z > y; \\ 0 & \text{if} \quad z = y; \\ +1 & \text{if} \quad z < y. \end{cases}$$

Weight each satisfaction or violation by its magnitude (absolute difference between the values):

$$f(z,y) = |y-z| \, \mathrm{sign}(y-z) = y - z$$

Anti-Robinson Events/Deviation

An even simpler loss function can be created in the same way as the gradient measures above by concentrating on violations only.

AR Events

$$L(\mathbf{D}) = \sum_{i<k<j} f(d_{ik}, d_{ij}) + \sum_{i<k<j} f(d_{kj}, d_{ij})$$

To only count the violations (called events) we use

$$f(z,y) = I(z,y) = \begin{cases} 1 & \text{if} \quad z < y \quad \text{and} \\ 0 & \text{otherwise.} \end{cases}$$

$I(\cdot)$ is an indicator function returning 1 only for violations.

(Chen 2002) also introduced a weighted versions of this loss function by using the absolute deviations as weights:

$$f(z,y) = |y - z|\, I(z,y)$$

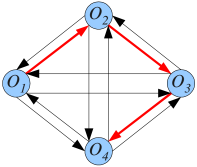
Hamiltonian Path Length

- $\mathbf{D}$ is seen as a complete weighted graph $G = (\Omega, E)$ with $\Omega = \{O_1, O_2, \ldots, O_n\}$ and the weight w_{ij} for edge $e_{ij} \in E$ represents d_{ij}.
- An order Ψ can be seen as a *Hamiltonian path* through the graph.
- **Minimizing the path length** results in a seriation optimal with respect to dissimilarities between neighboring objects (Hubert 1974, Caraux and Pinloche (2005)).

Loss function:

$$L(\mathbf{D}) = \sum_{i=1}^{n-1} d_{i,i+1}$$

$\mathbf{D}$	O_1	O_2	O_3	O_4
O_1	0	**4**	1	8
O_2	4	0	**2**	2
O_3	1	2	0	**3**
O_4	8	2	3	0



This optimization problem is related to the *traveling salesperson problem* (Gutin and Punnen 2002) for which good solvers and efficient heuristics exist.

Other Measures

- Inertia criterion (Caraux and Pinloche 2005)
- Least squares criterion (Caraux and Pinloche 2005)
- Measure of Effectiveness (McCormick, Schweitzer, and White 1972)
- Moore and Neumann stress (Niermann 2005)
- Relative generalized Anti-Robinson events (RGAR) (Tien et al. 2008)

Optimization Techniques

Partial Enumeration

Directly optimize the column/row gradient measures using:

- Dynamic programming (L. Hubert, Arabie, and Meulman 1987)
- Branch-and-bound (Michael Brusco and Stahl 2005)

solves problems with $n \leq 40$

Heuristics for larger problems:

- Simulated annealing (M. Brusco, Köhn, and Stahl 2007)

QAP Formulations

Quadratic Assignment problem

The QAP is in general NP-hard. Methods include QIP, linearization, branch and bound and cutting planes as well as heuristics including Tabu search, simulated annealing, genetic algorithms, and ant systems (Burkhard 1998)

$$\text{QAP}(\mathbf{A},\mathbf{B}) : \min_{\Psi \in \mathcal{P}_n} \sum_{i,j=1}^{n} \mathbf{A}_{ij} \Psi(\mathbf{B})_{ij}$$

QAP Formulations

Quadratic Assignment problem

The QAP is in general NP-hard. Methods include QIP, linearization, branch and bound and cutting planes as well as heuristics including Tabu search, simulated annealing, genetic algorithms, and ant systems (Burkhard 1998)

$$\mathrm{QAP}(\mathbf{A}, \mathbf{B}) : \min_{\Psi \in \mathcal{P}_n} \sum_{i,j=1}^{n} \mathbf{A}_{ij} \Psi(\mathbf{B})_{ij}$$

Formulate as **2-Sum Problem** (Barnard, Pothen, and Simon 1993)

$$\min_{\Psi} \sum_{i,j=1}^{n} \Psi(\mathbf{S})_{ij} (i-j)^2) \rightarrow \mathrm{QAP}([(i-j)^2]_{n \times n}, \mathbf{S})$$

where similarity matrix $\mathbf{S} = \frac{1}{1+\mathbf{D}}$

QAP Formulations

Quadratic Assignment problem

The QAP is in general NP-hard. Methods include QIP, linearization, branch and bound and cutting planes as well as heuristics including Tabu search, simulated annealing, genetic algorithms, and ant systems (Burkhard 1998)

$$\mathrm{QAP}(\mathbf{A}, \mathbf{B}) : \min_{\Psi \in \mathcal{P}_n} \sum_{i,j=1}^{n} \mathbf{A}_{ij} \Psi(\mathbf{B})_{ij}$$

Formulate as **2-Sum Problem** (Barnard, Pothen, and Simon 1993)

$$\min_{\Psi} \sum_{i,j=1}^{n} \Psi(\mathbf{S})_{ij} (i-j)^2) \rightarrow \mathrm{QAP}([(i-j)^2]_{n \times n}, \mathbf{S})$$

where similarity matrix $\mathbf{S} = \frac{1}{1+\mathbf{D}}$

Formulate as **Linear Seriation Problem** (L. Hubert and Schultz 1976)

$$\min_{\Psi} \sum_{i,j=1}^{n} \Psi(\mathbf{D})_{ij} (-|i-j|) \rightarrow \mathrm{QAP}([-|i-j|]_{n \times n}, \mathbf{D})$$

Spectral Seriation

Minimizes the **2-Sum Problem** formulation:

$$\min_{\Psi} \sum_{i,j=1}^{n} \Psi(\mathbf{S})_{ij}(i-j)^2)$$

Rewriting the minimization problem using a permutation vector π, its inverse, rescaling to q and using a Lagrangian multiplier for the constraint on the permutation yields (Ding and He 2004) the following equivalent optimization problem:

$$\min_{\mathbf{q}} \frac{\mathbf{q}^T L_{\mathbf{S}} \mathbf{q}}{\mathbf{q}^T \mathbf{q}}$$

where $L_{\mathbf{S}}$ is the Laplacian of $\mathbf{S}$.

The optimal order can be recovered by the order of the Fielder vector (second smallest eigenvector of the Laplacian).

Hamiltonian Path

TSP Solver

- Concorde (fast cutting planes and tour separation)

Heuristic

- TSP Heuristics (Lin–Kernighan, Nearest neighbors, insertion heuristics, etc.)
- Hierarchical clustering with reordering:
  - Gruvaeus-Wainer heuristic (Gruvaeus and Wainer 1972),
  - Optimal leaf ordering (Bar-Joseph et al. 2001)

Other Heuristics

- Hierarchical clustering
- Visual Assessment of cluster Tendency (VAT based on a MST) (Bezdek and Hathaway 2002)
- Multidimensional scaling (metric, non-metric, angle)
- Rank-two ellipse seriation (series of correlation matrices) (Chen 2002).
- Sorting Points Into Neighborhoods (SPIN) (D. Tsafrir et al. 2005)

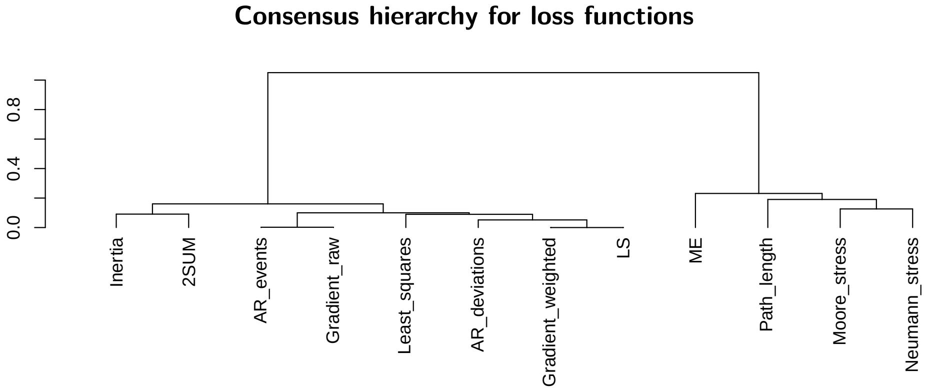
Experimental Comparison

Data Sets

- **Pre-Robinson:** A pre-Anti-Robinson matrix with 100 objects.
- **Random:** 200 objects with 2 features independently and uniformly drawn from $[0, 1]$ (Euclidean dist.)
- **Iris:** 150 flowers with 4 features (scaled, Euclidean dist.)
- **Zoo:** 101 animals with 17 (mostly 0-1) features (Euclidean dist.)
- **Votes:** 1984 Congressional votes for 435 congress men on 16 key votes: yes, no, abstain (Jaccard index on 32 binary features)
- **Wood:** 136 poplar trees with normalized gene expression for 6 locations (Euclidean dist.)
- **Elutriation:** Ratios of gene expression levels for a sample of 100 genes of Saccharomyces cerevisiae with 14 eigengenes (from SVD) as features (Euclidean dist.)

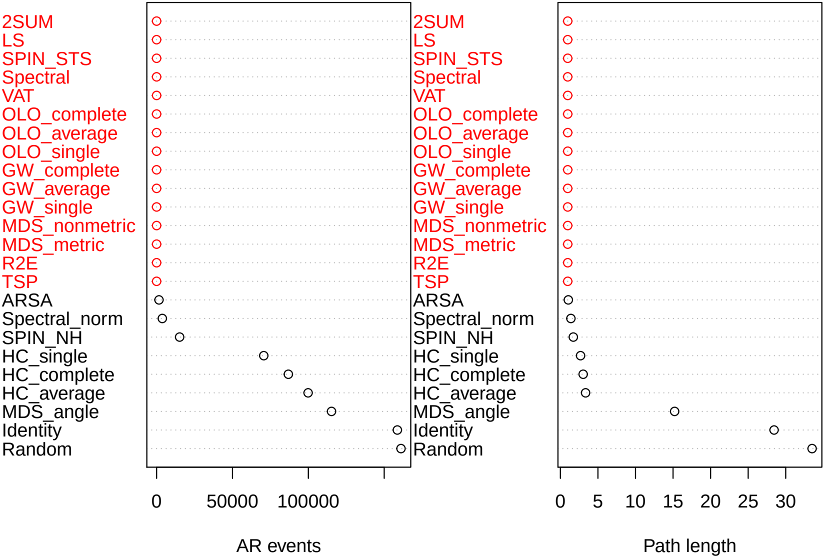
Comparison: Loss Functions

Compare how different loss functions rank the results of different seriation methods.

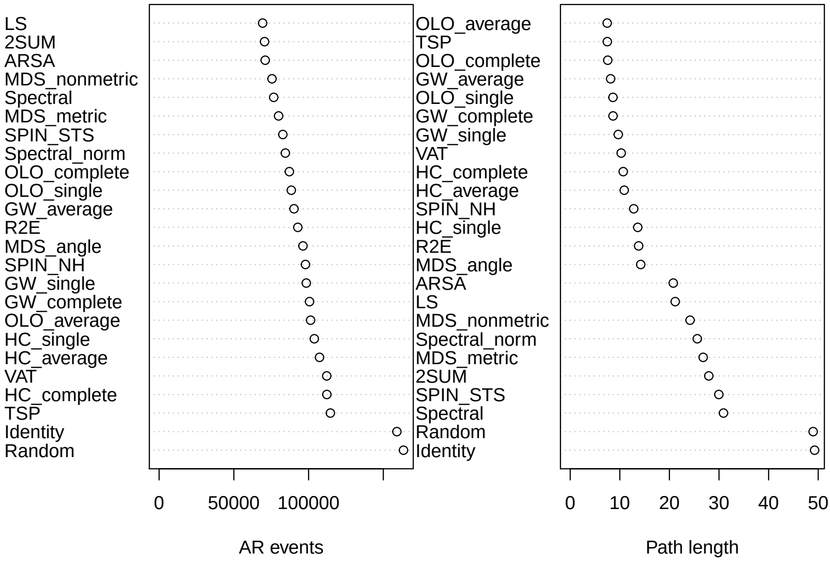


Best least square ultrametric approximation over all methods and data sets.

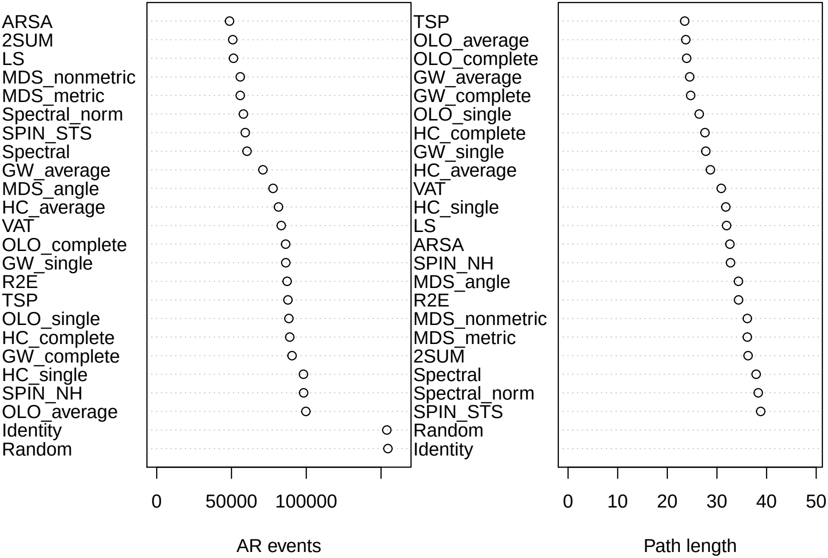
Method Comparison: Pre-Robinson Data



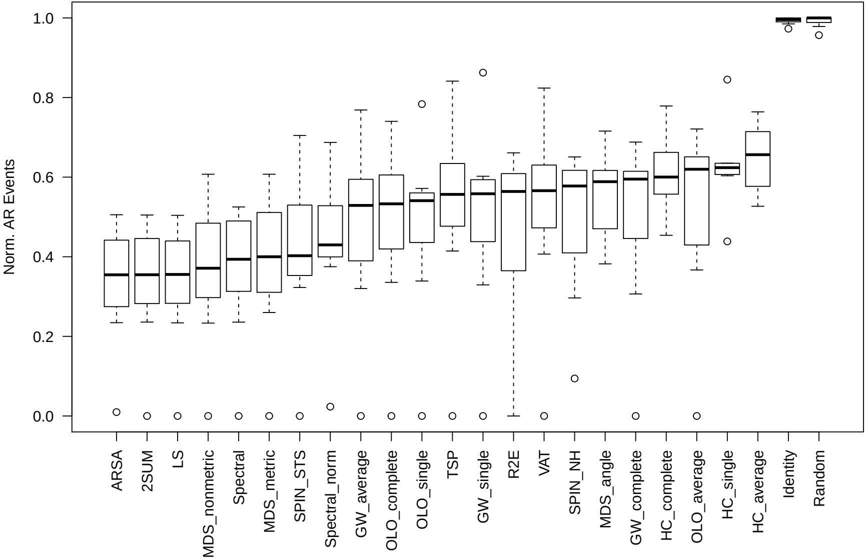
Method Comparison: Random Data



Method Comparison: Votes

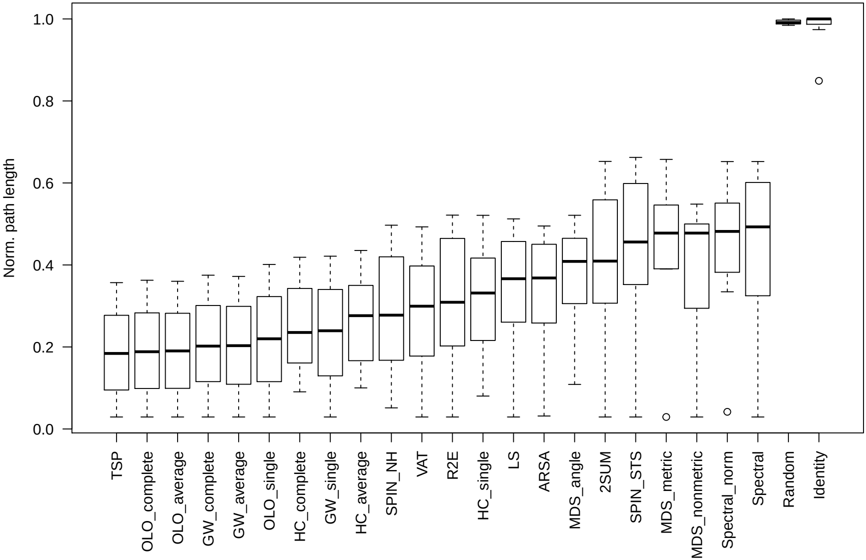


Method Comparison: Anti-Robinson Events



Distribution over all data sets; normalized by largest AR.

Method Comparison: Path length



Distribution over all data sets; normalized by longest path.

Compare Resulting Orders

	Identity	Random	ARSA	TSP	R2E	MDS_metric	MDS_nonmetric	MDS_angle	HC_single	HC_complete
1	47	2	1	36	36	1	1	60	12	1
2	5	56	2	71	71	2	2	47	84	58
3	25	55	3	41	41	3	3	74	82	90
4	69	85	4	31	31	33	33	72	28	62
5	29	26	5	43	43	5	5	88	6	6
6	26	54	6	68	68	6	6	31	5	5
7	97	32	7	64	64	7	7	97	9	68
8	98	59	8	73	73	8	8	4	65	52
9	94	11	9	90	90	9	9	46	96	78
10	65	24	10	60	60	10	10	9	95	77
11	1	43	11	26	26	11	11	2	11	3
12	53	28	12	55	55	12	12	12	55	95
13	32	86	13	33	33	31	31	16	30	31
14	87	5	14	82	82	62	62	15	86	51
15	64	69	15	44	44	15	15	86	70	42
16	10	7	16	42	42	20	20	91	91	82
17	13	79	17	14	14	17	17	70	19	57
18	19	89	18	34	34	18	18	1	39	84
19	22	95	19	76	76	22	22	20	57	20
20	8	46	20	65	65	16	16	84	3	39
21	63	58	21	93	93	21	21	71	27	65
22	21	53	22	23	23	19	19	83	79	83
23	80	63	23	19	19	23	23	43	25	69
24	6	51	24	96	96	24	24	90	24	19
25	72	42	25	5	5	43	43	76	62	30
26	100	35	26	11	11	26	26	37	77	43
27	45	60	27	35	35	27	27	17	36	56
28	85	10	28	59	59	28	28	8	41	34
29	83	82	29	32	32	29	29	28	76	99
30	67	66	30	50	50	30	30	26	53	87

Shown are the object ranks for the pre-Robinson data (first 30 objects sorted by ARSA and first 10 methods).

Compare Orders (cont.)

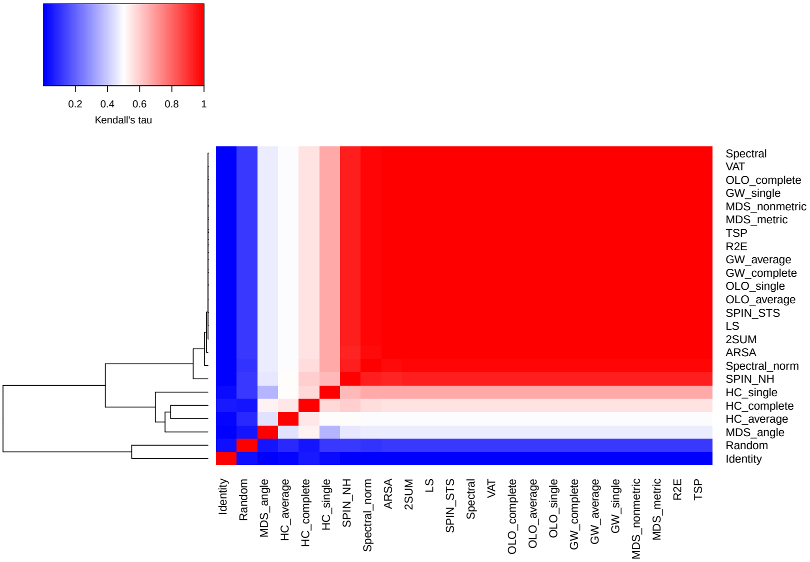
Compare the seriation order produced by two methods using

- **Kendall's rank-order correlation**

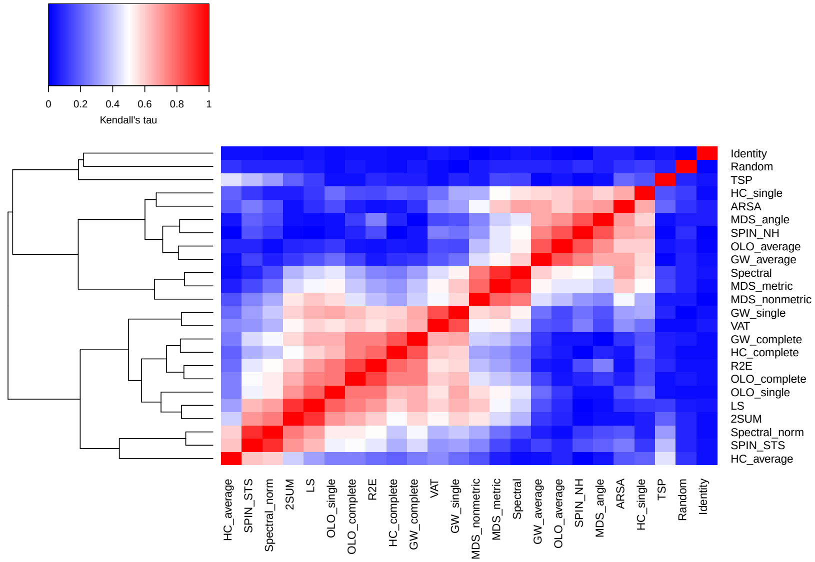
Other options would be

- Spearman's rank-order correlation
- Spearman's footrule metric (i.e., Manhattan distance between ranks) (Diaconis 1988)
- Positional proximity coefficient (Goulermas et al 2015)

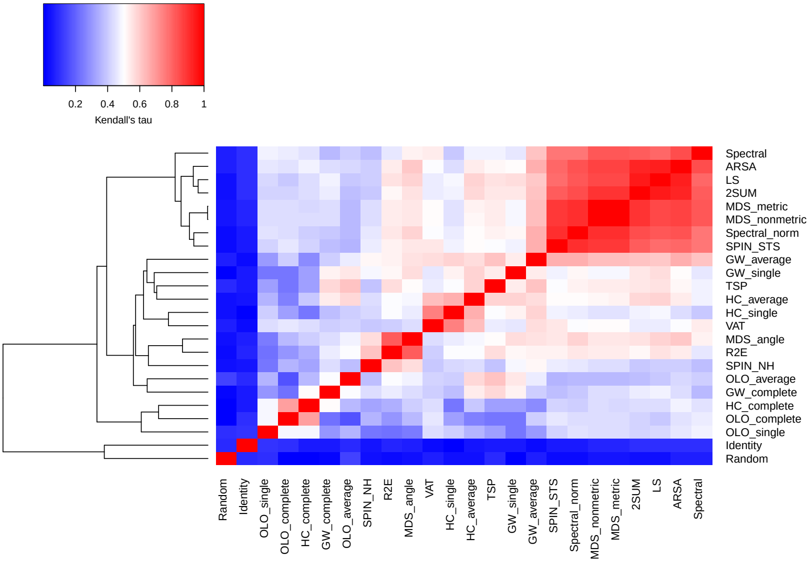
Comparison: Pre-Robinson



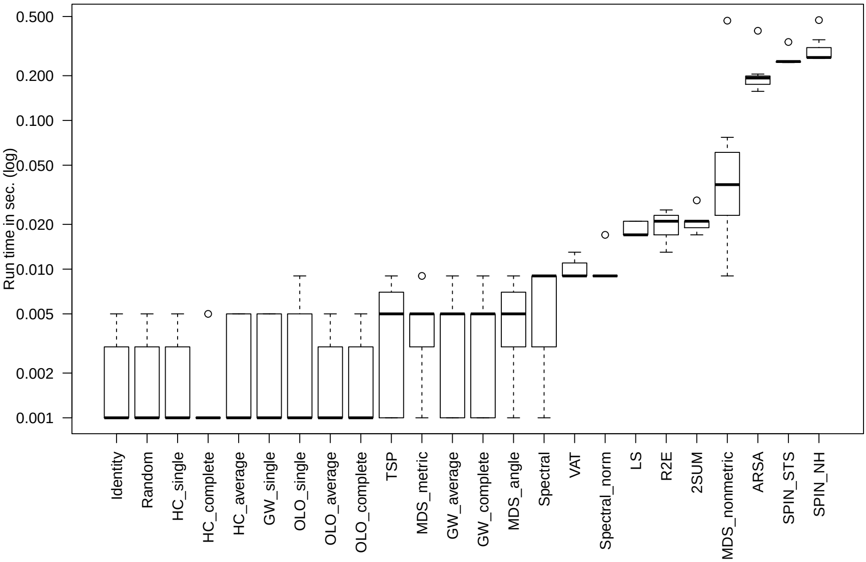
Comparison: Random Data



Comparison: Votes

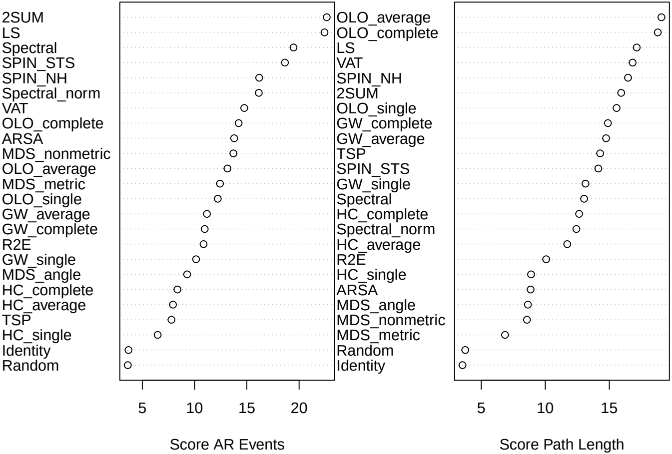


Computational Time



Comparison: Quality and Speed

1. Rank each algorithm for quality and speed (1 point for the worse, 2 for the next, etc.)
2. Average points over all data sets.



Conclusion

In this limited study we empirically found:

- There are two big groups of objective functions
  1. **Anti Robinson Events** (AR) / **Gradient Measure**
  2. **Path length**
- For group 1 (AR) **QAP formulations** provide a good tradeoff between quality and speed.
- For group 2 (path length) **Hierachical Clustering with Optimal Leaf Ordering** is fast and also provides quality comparable to TSP solvers.

Future work

- Even the heuristics (except for the TSP) are poorly scalable to applications with thousands of objects. Faster seriation heuristics are needed.

Thank you!

All seriation methods are available in the R package **seriation**.

You can contact me at `mhahsler@lyle.smu.edu`

References I

Arabie, P., and L. J. Hubert. 1996. “An Overview of Combinatorial Data Analysis.” In *Clustering and Classification*, edited by P. Arabie, L. J. Hubert, and G. De Soete, 5–63. River Edge, NJ: World Scientific.

Bar-Joseph, Z., E. D. Demaine, D. K. Gifford, and T. Jaakkola. 2001. “Fast Optimal Leaf Ordering for Hierarchical Clustering.” *Bioinformatics* 17 (1): 22–29.

Barnard, S. T., A. Pothen, and H. D. Simon. 1993. “A Spectral Algorithm for Envelope Reduction of Sparse Matrices.” In *Proceedings of the 1993 ACM/IEEE Conference on Supercomputing*, 493–502. Supercomputing ’93. New York, NY, USA: ACM.

Bezdek, J.C., and R.J. Hathaway. 2002. “VAT: A Tool for Visual Assessment of (Cluster) Tendency.” In *Proceedings of the 2002 International Joint Conference on Neural Networks (IJCNN ’02)*, 3:2225–30.

Brusco, M., H. F. Köhn, and S. Stahl. 2007. “Heuristic Implementation ofDynamic Programming for Matrix Permutation Problems in CombinatorialData Analysis.” *Psychometrika*.

Brusco, Michael, and Stephanie Stahl. 2005. *Branch-and-Bound Applications in Combinatorial Data Analysis*. Springer.

Caraux, G., and S. Pinloche. 2005. “Permutmatrix: A Graphical Environment to Arrange Gene Expression Profiles in Optimal Linear Order.” *Bioinformatics* 21 (7): 1280–81.

Chen, Chun-Houh. 2002. “Generalized Association Plots: Information Visualization via Iteratively Generated Correlation Matrices.” *Statistica Sinica* 12 (1): 7–29.

Ding, Chris, and Xiaofeng He. 2004. “Linearized Cluster Assignment via Spectral Ordering.” In *In ICML ’04: Proceedings of the Twenty-First International Conference on Machine Learning*, 30. ACM Press.

Gruvaeus, G., and H. Wainer. 1972. “Two Additions to Hierarchical Cluster Analysis.” *British Journal of Mathematical and Statistical Psychology* 25: 200–206.

Gutin, G., and A. P. Punnen, eds. 2002. *The Traveling Salesman Problem and Its Variations*. Vol. 12. Combinatorial Optimization. Dordrecht: Kluwer.

Hubert, L. J. 1974. “Some Applications of Graph Theory and Related Nonmetric Techniques to Problems of Approximate Seriation: The Case of Symmetric Proximity Measures.” *British Journal of Mathematical Statistics and Psychology* 27: 133–53.

Hubert, Lawrence, and James Schultz. 1976. “Quadratic Assignment as a General Data Analysis Strategy.” *British Journal of Mathematical and Statistical Psychology* 29 (2). Blackwell Publishing Ltd: 190–241.

References II

Hubert, Lawrence, Phipps Arabie, and Jacqueline Meulman. 1987. *Combinatorial Data Analysis: Optimization by Dynamic Programming*. Society for Industrial Mathematics.

McCormick, William T., Paul J. Schweitzer, and Thomas W. White. 1972. "Problem Decomposition and Data Reorganization by a Clustering Technique." *Operations Research* 20 (5): 993–1009.

Niermann, Stefan. 2005. "Optimizing the Ordering of Tables with Evolutionary Computation." *The American Statistician* 59 (1): 41–46.

Robinson, W. S. 1951. "A Method for Chronologically Ordering Archaeological Deposits." *American Antiquity* 16: 293–301.

Tien, Yin-Jing, Yun-Shien Lee, Han-Ming Wu, and Chun-Houh Chen. 2008. "Methods for Simultaneously Identifying Coherent Local Clusters with Smooth Global Patterns in Gene Expression Profiles." *BMC Bioinformatics* 9 (155): 1–6.

Tsafrir, Dan, Ilan Tsafrir, Liat Ein-Dor, Or Zuk, Daniel A. Notterman, and Eytan Domany. 2005. *Bioinformatics* 21 (10): 2301–8.