Attribute Clustering Based on Heuristic Tree Partition

Jorge Cordero Hernandez and Yifeng Zeng, Department of Computer Science, Aalborg University,
Selma Lagerlös Vej 300, DK-9220 Aalborg ø, Denmark

Background

Attribute clustering is the task of selecting subsets of highly dependent variables from data.

Motivations

- Widest use in bioinformatics.
- Feature selection, Grouping variables.
- Tree partitioning is easy to implement and fast to execute.

Objective

Given X, the objective is to find a set of disjoint clusters $C = \{C_i | (i = 1, \cdots, k) \wedge (\forall_{i \neq j} C_i \cap C_j = \emptyset)\}$ that maximizes:

$$W^C = \sum_{C_i} \sum_{x_j \in (C_i - \{o_i\})} w_{o_i, x_j} \quad (1)$$

where w_{o_i,x_j} denote the weight (measured by association functions) from the center o_i to other variables x_j in the cluster C_i.

Methods

k-modes

It identifies points in the space (variables) as centers or modes. Every variable is attached to a mode whose distance is minimal.

Maximum Spanning Tree (MAST) Clustering

- Standard Euclidean maximum spanning tree (SEMST): k - 1 inconsistent edges with minimal weights are removed.
- Maximum cost spanning tree (CESMT): k - 1 inconsistent edges with minimal costs are removed.
- Zahn's maximum spanning tree (ZEMST): Edges are removed if their attached weights are smaller than the average of weights in neighborhoods.

Star Discovery Algorithm

- The star discovery (SD) algorithm uses both weights and topology for deleting inconsistent edges.
- The MAST is divided into Spanning Stars (we encapsulate a center O, adjacent variables (Adj) and leaf nodes ($Leaf$)).
- A Spanning Star is a sub-tree over the MAST, $S = (V_S, E_S)$. It has a center and a set of adjacent nodes (extended to include leaf nodes).

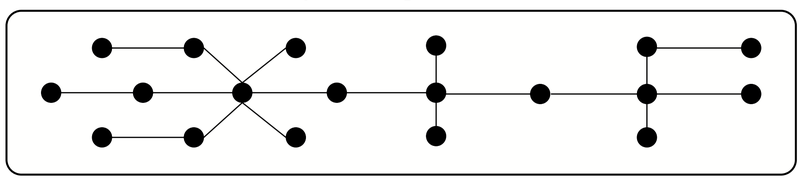
Illustration

We find the set of stars SS that maximizes:

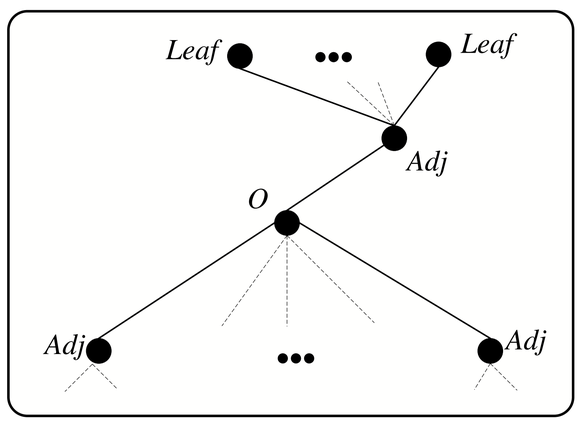
$$W = \sum_{S_l \in SS} \Big(\sum_{x_i \in Adj_l} (w_{x_i,o_l}) + \sum_{x_j \in Adj_l, x_h \in Leaf_l} (w_{x_j,x_h}) \Big) \quad (2)$$

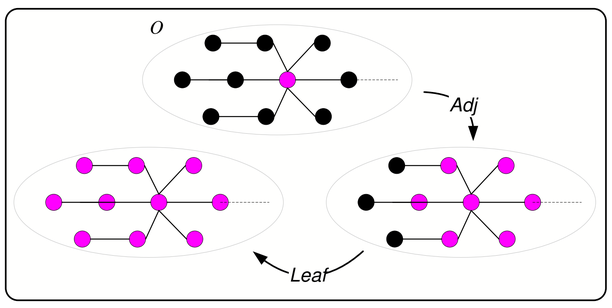
where o_l is the star(cluster) center, Adj_l is a set of adjacent nodes to the center node o_l, and $Leaf_l$ a set of leaf nodes that connect to either o_l or Adj_l.

- Find a maximum spanning tree

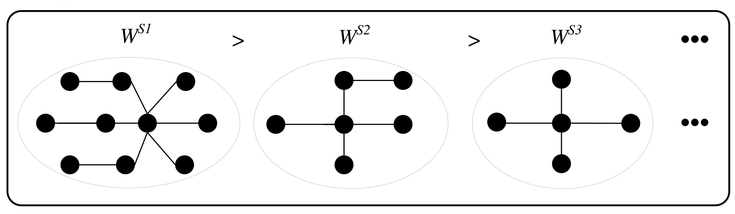


- Find a set of stars and calculate star weights

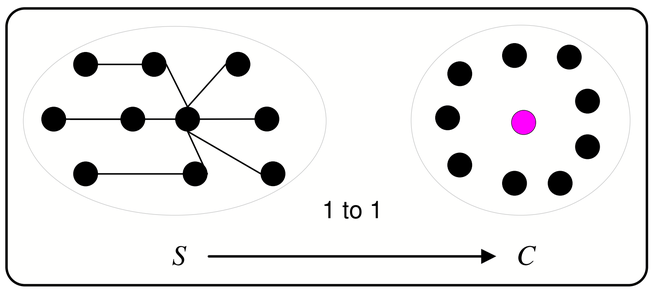




- Sort all star and select the stars having maximal weights



- Every resulted star becomes a cluster



Experimental Results

k-modes Reliability

Feed all possible combinations of initial modes given the setting: k=2 and Ω=10000.

Table 1: Number of local optima into which the k-modes algorithm falls.

Domains vs Local Optima			
Alarm	HeparII	Hailfinder	Pathfinder
17	130	91	117

Cluster Quality

Initial modes of k-modes are taken from the final modes in SD.

Table 2: Performance (W^C) of algorithms (Alg.) in five domains over different sample sizes Ω and k=8. The k-modes algorithm is optimal when fed with the right initial modes.

		Ω			
Domain	**Alg.**	**10000**	**8000**	**6000**	**4000**
Alarm	SEMST	4.13	18.41	21.61	22.99
	CESMT	5.4	18.78	22.07	23.52
	ZEMST	6.11	19.10	22.85	24.66
	SD	7.85	**21.30**	**23.95**	**25.38**
	k-modes	**8.35**	**21.30**	**23.95**	**25.38**
Barley	SEMST	2.33	14.67	19.03	22.23
	CESMT	2.55	14.85	19.24	22.48
	ZEMST	3.85	14.91	20.70	24.20
	SD	4.88	**15.39**	**21.02**	**25.41**
	k-modes	**5.61**	**15.39**	**21.02**	**25.41**
HeparII	SEMST	50.97	50.32	51.49	52.32
	CESMT	51.21	50.55	51.71	52.89
	ZEMST	51.27	51.43	52.55	53.54
	SD	**55.57**	**56.98**	**58.34**	**59.56**
	k-modes	**55.57**	**56.98**	**58.34**	**59.56**
Hailfinder	SEMST	30.26	31.33	32.42	33.65
	CESMT	31.02	32.00	33.01	34.16
	ZEMST	32.41	33.28	33.81	34.97
	SD	**32.48**	**33.58**	**34.69**	**35.96**
	k-modes	**32.48**	**33.58**	**34.69**	**35.96**
Pathfinder	SEMST	85.98	87.53	88.75	89.82
	CESMT	88.63	88.22	89.40	90.19
	ZEMST	88.315	88.75	89.64	90.61
	SD	86.61	89.31	89.71	91.03
	k-modes	**90.33**	**89.41**	**91.32**	**92.72**

Timing Issue

Measurements in seconds given Ω=10000.

Table 3: Elapsed times for algorithms in all domains.

	Domain				
Alg.	Alarm	Barley	HeparII	Hailfinder	Pathfinder
SEMST	**0.031**	**0.04**	**0.044**	**0.049**	**0.047**
CESMT	0.04	0.042	0.056	0.05	0.062
ZEMST	0.078	0.057	0.065	0.07	0.094
SD	0.047	**0.04**	0.046	0.061	0.062
k-modes	0.109	0.063	0.077	0.078	0.125

Discussions

- The SD outperforms other MAST-based clustering algorithms (quality).
- The SD is competitive (it obtains similar results as k-modes in a reduced time complexity).
- The SD is deterministic while the k-modes falls in local optima.
- The SD + k-modes produces optimal clusters.
- Complexity of learning a BN can be reduced by learning small BNs (for each cluster) and then combining them.