COMBINATORIAL OPTIMIZATION AND SPARSE COMPUTATION FOR IMAGE SEGMENTATION AND LARGE SCALE DATA MINING

Dorit S. Hochbaum
University of California, Berkeley

LNMB, Jan 2019

OVERVIEW

- Motivation: Image segmentation and normalized cut
- Insights on how combinatorial optimization relates to spectral clustering
- Efficient polynomial time algorithm(s)
- The power of using pairwise similarities
- Lessons from experimental studies on effectiveness for image segmentation and for machine learning and data mining classification tasks.

NOTATIONS AND PRELIMINARIES

An undirected graph	$G = (V, E) \quad n = \lvert V \rvert \quad m = \lvert E \rvert$
Edges' weights	$w_{ij} \quad \forall [i, j] \in E$
Node weights	$q_i \quad \forall i \in V$
Capacity of (A, B)	$C(A, B) = \sum_{[i,j] \in E, i \in A, j \in B} w_{ij}$
Weighted degree	$d_i = \sum_{j [i,j] \in E} w_{ij}$
Degree volume	$d(A) = \sum_{i \in A} d_i = 2C(A, A) + C(A, \bar{A})$
Node volume	$q(A) = \sum_{i \in A} q_i$

AN INTUITIVE CLUSTERING CRITERION

Find a cluster that combines two objectives:
One, is to have large similarity within the cluster, and
to have small similarity between the cluster and its complement.

The combination of the two objectives can be expressed as:

HNC_1

$$\min_{S \subset V} \frac{C(S, \bar{S})}{C(S, S)} \quad \text{or}$$

HNC_2

$$\min_{S \subset V} C(S, \bar{S}) - \lambda C(S, S) \quad \text{or}$$

HNC_3

$$\min_{S \subset V} C_1(S, \bar{S}) - \lambda C_2(S, S)$$

We call this problem **H-normalized-cut (H for Hochbaum),** or **HNC**.

MOTIVATION FOR THE HNC_1 PROBLEM

NC, Shi and Malik 2001: $$\min_{S \subset V} \frac{C(S,\bar{S})}{d(S)} + \frac{C(S,\bar{S})}{d(\bar{S})}$$

Normalized cut (NC): NP-hard

Sharon et al. 2007: $$\min_{S \subset V} \frac{C(S,\bar{S})}{C(S,S)}$$

NP-hard?? **Same problem??**

HNC is poly time solvable: monotone IP3 (Hochbaum2010)

For "seed" nodes s and t, find a cluster S:

$$\min_{S \subset V} \frac{C(S, \bar{S})}{C(S, S)}$$

The formulation is (for $x_s = 1,\ x_t = 0$):

$$
\begin{array}{lll}
\text{(HNC)} & \min & \frac{\sum w_{ij} z_{ij}}{\sum w'_{ij} y_{ij}} \\
& \text{subject to} & x_i - x_j \leq z_{ij} \quad \text{for all } [i,j] \in E \\
& & x_j - x_i \leq z_{ji} \quad \text{for all } [i,j] \in E \\
& & y_{ij} \leq x_i \quad \text{for all } [i,j] \in E \\
& & y_{ij} \leq x_j \\
& & x_j, z_{ij}, y_{ij} \text{ binary.}
\end{array}
$$

This formulation is monotone

[H10,H13]

HOW DO NC AND HNC_1 COMPARE [H10]

Shi and Malik 2000:

$$\min_{S\subset V} \frac{C(S,\bar{S})}{d(S)} + \frac{C(S,\bar{S})}{d(\bar{S})}$$

HNC_1:

$$\min_{S\subset V} \frac{C(S,\bar{S})}{C(S,S)} = \frac{C(S,\bar{S})}{\frac{1}{2}[d(S) - C(S,\bar{S})]}$$

$$= \frac{1}{\frac{d(S)}{2C(S,\bar{S})} - \frac{1}{2}} \Rightarrow \max_{S\subset V} \frac{d(S)}{2C(S,\bar{S})} \Rightarrow \min_{S\subset V} \frac{C(S,\bar{S})}{d(S)}$$

Solving HNC_1 with the Spectral method [Sharon et al. 07] and optimally [H10]

- The {0,1} discrete values of *x* are relaxed. This continuous problem was shown to be solved approximately by an eigenvector.

$$\min_{x_i \in \{0,1\}} \frac{\sum w_{ij}\left(x_i - x_j\right)^2}{\sum w_{ij} x_i x_j} = \frac{x^T \mathcal{L} x}{x^T W x}$$

W is not a diagonal matrix.

However, using the formulation:

$$\min_{S \subset V} \frac{C\left(S, \bar{S}\right)}{d\left(S\right)} = \min_{x_i \in \{0,1\}} \frac{\sum w_{ij}\left(x_i - x_j\right)^2}{\sum d_i x_i^2} = \frac{x^T \mathrm{L} x}{x^T D x}$$

How does the HNC_1 solution relate to the spectral solution?

We will answer a more general question:

Consider q-normalized cut

TWO-TERMS FORMS OF THE PROBLEMS

Expander:

$$\min_{\emptyset \subset S \subset V, |S| \le |V|/2} \frac{C(S,\bar{S})}{|S|}$$

s-normalized:

$$\min_{S \subset V} \frac{C(S,\bar{S})}{|S|} + \frac{C(S,\bar{S})}{|\bar{S}|}$$

Cheeger constant:

$$\min_{\emptyset \subset S \subset V, |d(S)| \le |d(V)|/2} \frac{C(S,\bar{S})}{d(S)}$$

Normalized cut:

$$\min_{S \subset V} \frac{C(S,\bar{S})}{d(S)} + \frac{C(S,\bar{S})}{d(\bar{S})}$$

Half-q-normalized:

$$\min_{\emptyset \subset S \subset V, |q(S)| \le |q(V)|/2} \frac{C(S,\bar{S})}{q(S)}$$

q-normalized:

$$\min_{S \subset V} \frac{C(S,\bar{S})}{q(S)} + \frac{C(S,\bar{S})}{q(\bar{S})}$$

TWO TERMS EXPRESSIONS AND THE RAYLEIGH RATIO

Lemma 1 (cf. Hochbaum 2011):

$$\min_{S\subset V} \frac{C(S,\bar{S})}{q(S)} + \frac{C(S,\bar{S})}{q(\bar{S})} = \min_{\substack{y^T Q\vec{1}=0 \\ y_i \in \{-b,1\}}} \frac{y^T \overbrace{(D-W)}^{L} y}{y^T Q y}$$

A special case of this was shown by Shi and Malik.

THE COMBINATORIAL VS. THE SPECTRAL CONTINUOUS RELAXATIONS

$$\min_{\substack{y^T Q\vec{1}=0 \\ y_i \in \{-b,1\}}} \frac{y^T L y}{y^T Q y}$$

Raleigh ratio Problem (RRP)

$$\min_{\substack{y^T Q\vec{1}=0 \\ \cancel{y_i \in \{-b,1\}}}} \frac{y^T L y}{y^T Q y}$$

Spectral continuous relaxation

$$\min_{\substack{\cancel{y^T Q\vec{1}=0} \\ y_i \in \{-b,1\}}} \frac{y^T L y}{y^T Q y}$$

Combinatorial relaxation

THE SPECTRAL METHOD

$$Ly = \lambda Q y$$

- Where λ is the smallest non-zero eigenvalue (Fiedler Eigenvalue). We solve for the eigenvector z:

$$(Q^{-1/2} L Q^{-1/2}) z = \lambda z$$

- and set $y = Q^{-1/2} z$ which solves the continuous relaxation.

SOLVING THE COMBINATORIAL RELAXATION

$$y_i \quad \begin{cases} = 1, & i \in S \\ = -b, & \text{otherwise} \end{cases}$$

THE COMBINATORIAL RELAXATION RAYLEIGH PROBLEM

Lemma 2:

$$\min_{y\in\{-b,1\}} \frac{y^T(D-W)y}{y^TQy} = \min_{\emptyset\subset S\subset V} \frac{(1+b)^2C(S,\bar{S})}{q(S)+b^2q(\bar{S})}$$

SOLVING THE COMBINATORIAL RAYLEIGH PROBLEM OPTIMALLY

The problem is a ratio problem
General technique for ratio Problems: The λ-question

$$\min_{x \in F} \frac{f(x)}{g(x)} < \lambda ?$$

can be solved if one can solve the following λ-question:

$$f(x) - \lambda g(x) < 0 ?$$

*This λ is unrelated to an eigenvector –just a parameter

SOLVING THE LAMBDA QUESTION

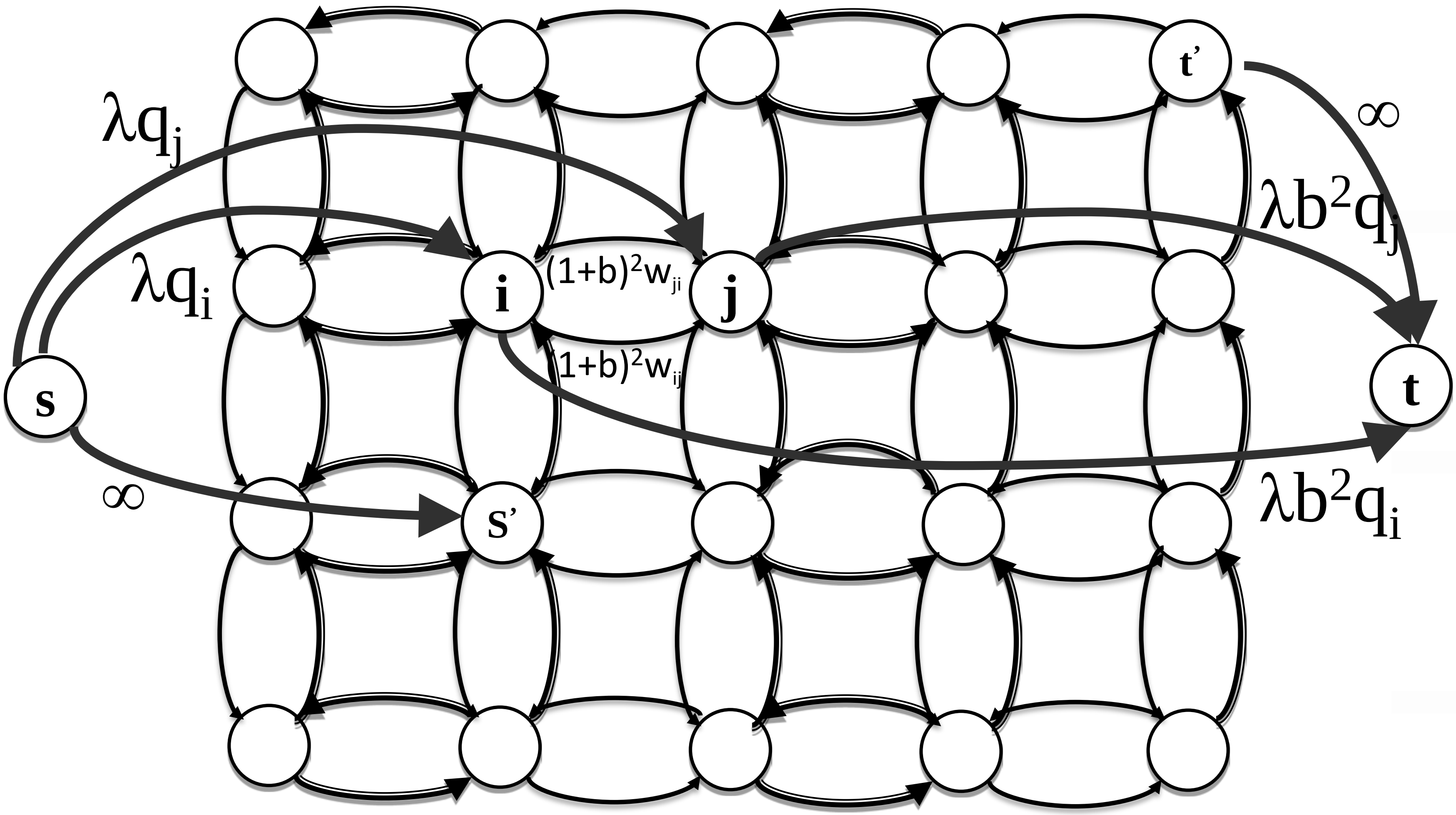
The λ-question of whether the value of RRP is less than λ is equivalent to determining whether:

$$\min_{y_i \in \{-b,1\}} y^T(D-W)y - \lambda y^T Q y < 0?$$

Or from Lemma 1, this is equivalent to:

$$\{\min_{S \subset V} (1+b)^2 C(S,\bar{S}) - \lambda[q(S) + b^2 q(\bar{S})]\} < 0?$$

THE GRAPH G_{ST} FOR TESTING THE LAMBDA-QUESTION: A CASE OF IP ON MONOTONE CONSTRAINTS [HOC02]



THEOREM

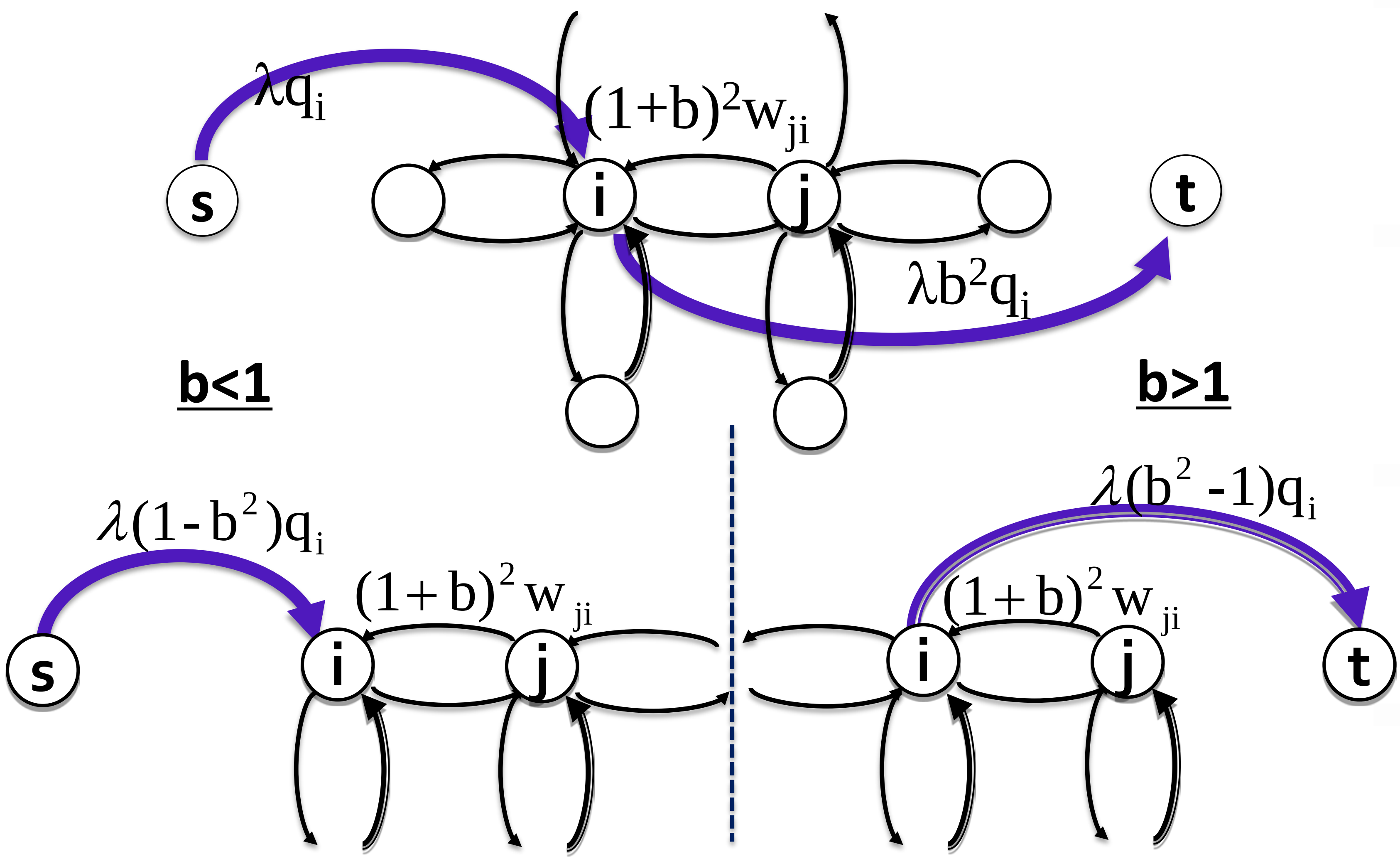
The source set of a minimum cut in the graph G_{st} is an optimal solution to the linearized (RRP)

Proof:

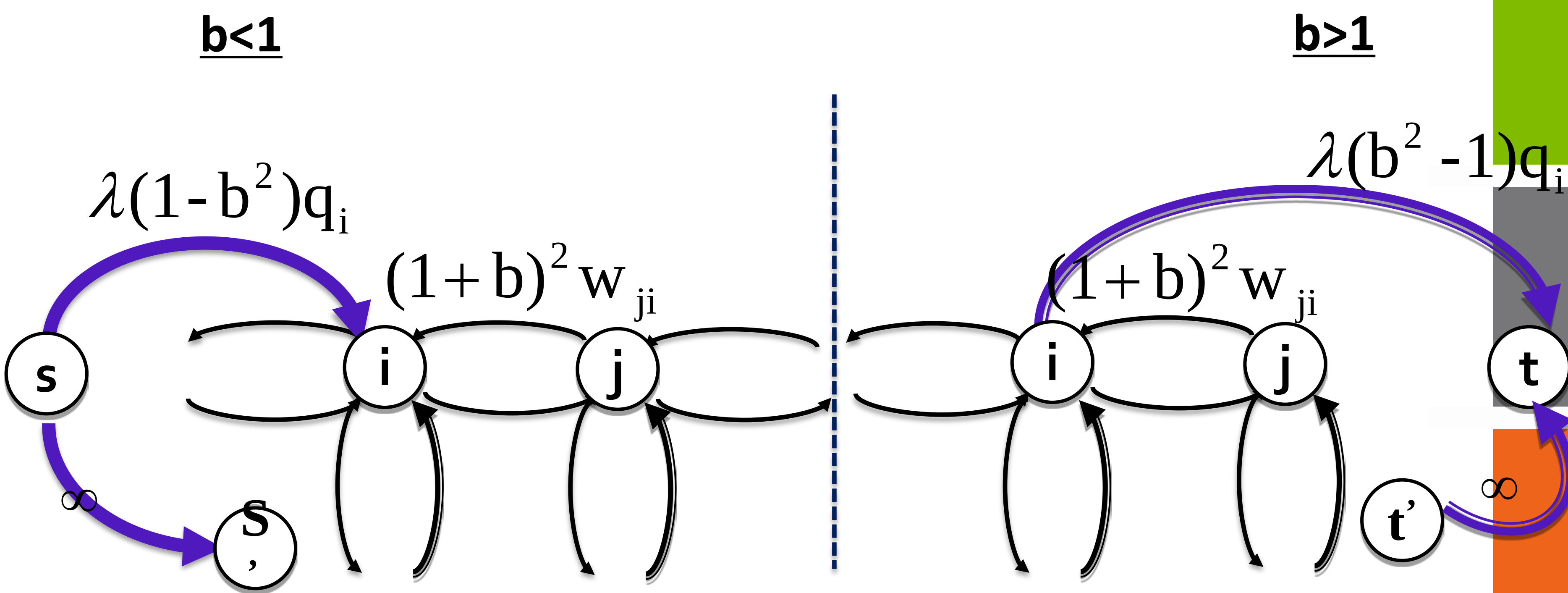
$$C(S \cup \{s\}, T \cup \{t\}) = \lambda q(T) + \lambda b^2 q(S) + C(S,T)$$
$$= \lambda(1+b^2)q(V) - \lambda q(S) - \lambda b^2 q(T) + C(S,T).$$

Since $\lambda(1+b^2)q(V)$ is a constant, minimizing $C(S \cup \{s\}, T \cup \{t\})$ is equivalent to minimizing $(1+b)^2 C(S,\bar{S}) - \lambda[q(S) + b^2 q(\bar{S})]$.

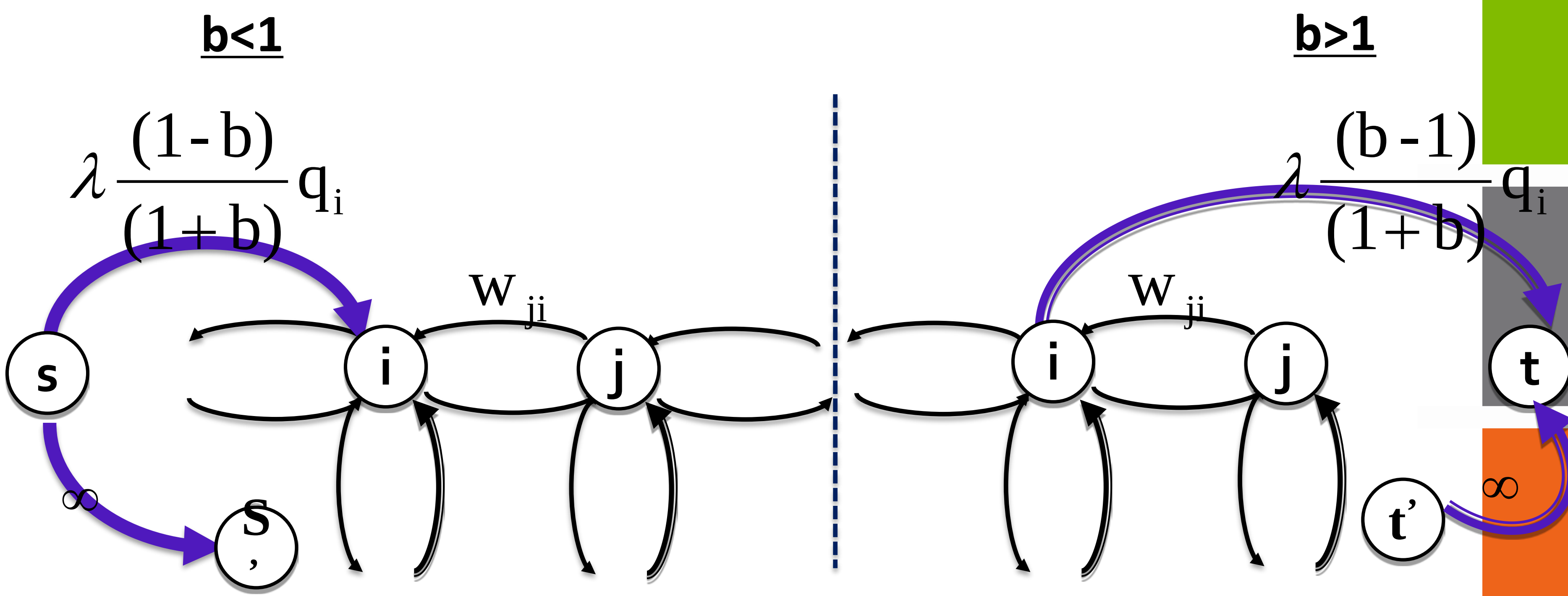
SIMPLIFYING THE GRAPH (TO MAKE IT PARAMETERIC)



SCALING ARCS WEIGHTS

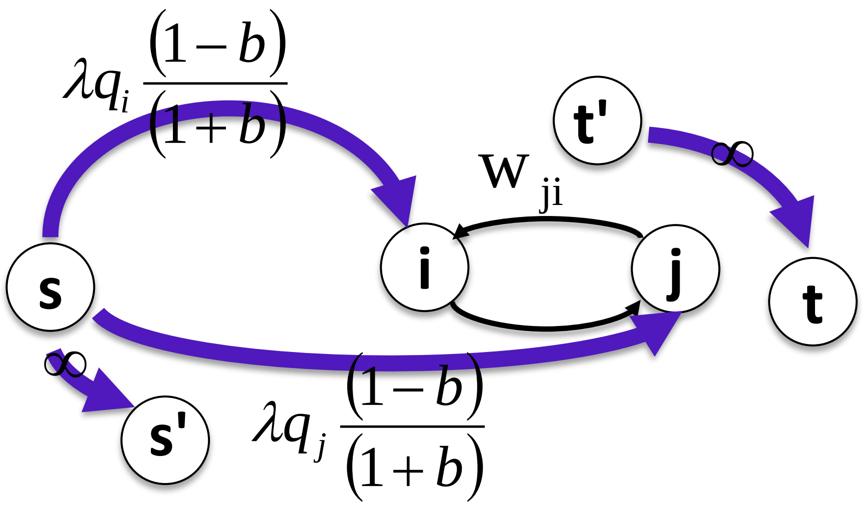


SCALING ARCS WEIGHTS

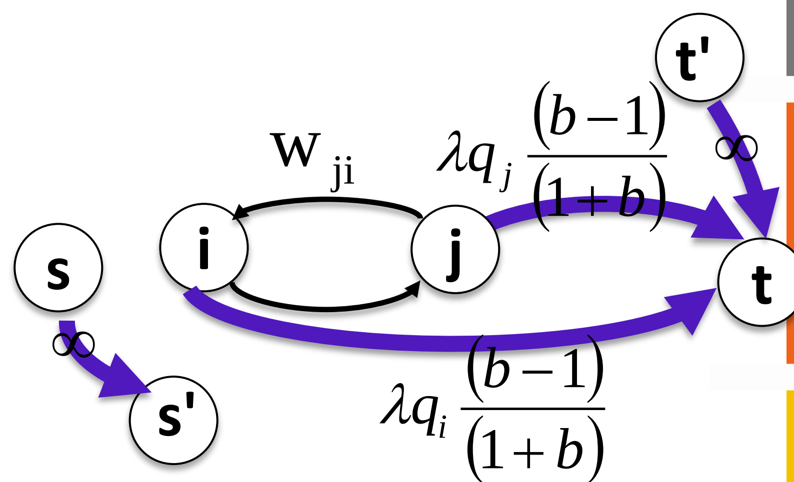


THE SIMPLIFIED EQUIVALENT GRAPH

b<1



b>1



SOLVING THE PROBLEM AS A PARAMETRIC MIN-CUT

The problem is a parametric cut problem: This is a graph setup when source adjacent arcs are monotone nondecreasing and sink adjacent are monotone nonincreasing (for b<1) with the parameter.

A parametric cut problem can be solved in the complexity of a single minimum cut (plus finding the zero of n monotone functions) [GGT89], [H08].

Here we let the parameter be β

$$\beta = \begin{cases} \lambda \dfrac{1-b}{1+b} & b < 1 \\ \lambda \dfrac{b-1}{1+b} & b \geq 1 \end{cases}$$

IN G_{ST}

The cut problem in the graph G_{st}, as a function of β is parametric (the capacities are linear in the parameter on one side and independent of it on the other).

In a parametric graph the sequence of source sets of cuts for increasing source-adjacent capacities is ***nested***.

There are no more than n breakpoints for β.

There are k≤n nested source sets of minimum cuts.

SOLVING FOR ALL VALUES OF B EFFICIENTLY

For

$$\beta = \begin{cases} \lambda \dfrac{1-b}{1+b} & b < 1 \\ \lambda \dfrac{b-1}{1+b} & b \geq 1 \end{cases}$$

Given the values of β at the breakpoints, we can generate, for each value of b, *all* the breakpoints.

Consequently, by solving once the parametric problem for β we obtain simultaneously, *all the breakpoint solutions for all b*, in the complexity of a single minimum cut.

For each b we find the last (largest value) breakpoint where the objective value <0.

RECALL PROBLEM HNC$_1$: IT IS A SPECIAL CASE

It is equal to the problem solved for b=0:

$$\min_{S \subset V} \frac{C(S, \bar{S})}{d(S)}$$

which has the same solution as:

$$\min_{S \subset V} \frac{C(S, \bar{S})}{C(S, S)}$$

IMAGE SEGMENTATION WITH HNC_1 VS SPECTRAL

Original Image

Shi & Malik

$$NC = 35 \cdot 10^{-4}$$

HNC_1

$$NC = 1.702 \cdot 10^{-4}$$

Original image

Eigenvector result

HNC_1 result

Another comparison

Original Image

Shi & Malik

$$NC = 127 \cdot 10^{-4}$$

NC’

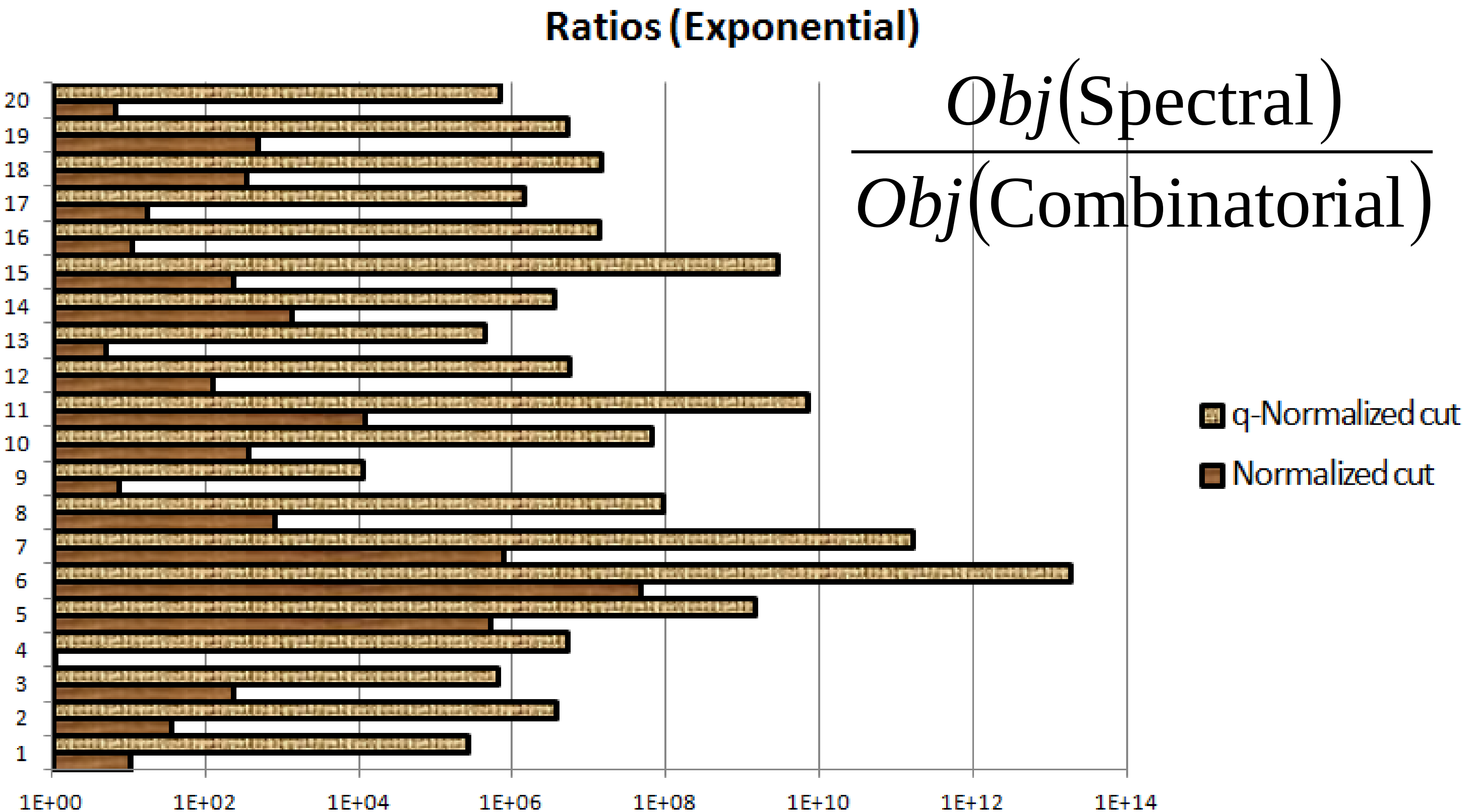
$$NC = 1.466 \cdot 10^{-4}$$

Original image

Eigenvector result

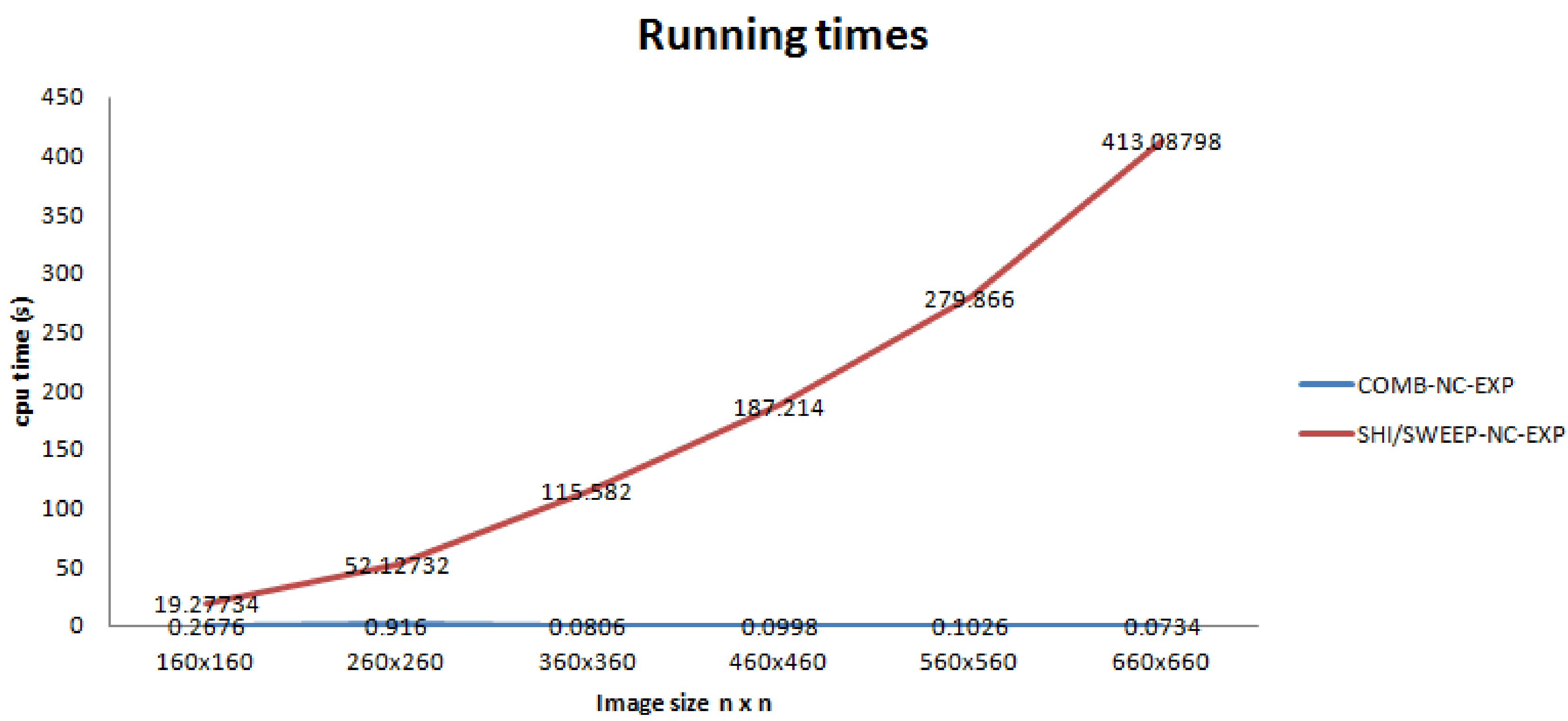
NC’ result

Spectral objective Vs. Combinatorial algorithm's objective [H,Lu,Bertelli13]



$$\frac{Obj(\text{Spectral})}{Obj(\text{Combinatorial})}$$

SCALABILITY OF THE ALGORITHM



HNC IN DATA MINING

HNC can be applied to binary classification problems

- Unsupervised:
  - Method finds a cluster distinct from the rest of the nodes and similar to itself
- Supervised (called SNC):
  - Training data is linked a-priori to either the source or the sink, based on the respective labels

HNC was successfuly used in data mining contexts (e.g., denoising spectra of nuclear detectors [YFHNS2014], ranking drugs according to their effectiveness, [HHY2012] and it has been a leading algorithm in the Neurofinder benchmark for cell identification in calcium imaging movies [SHA2017].)

TESTING THE EFFECTIVENESS OF HNC AS A DATA MINING PROCEDURE [BAUMANN, H, YANG,17]

Two variants of HNC were tested:

1. The node weights are d_i **SNC** (supervised HNC)
2. The node weights are the average label of k nearest neighbors **SNCK**

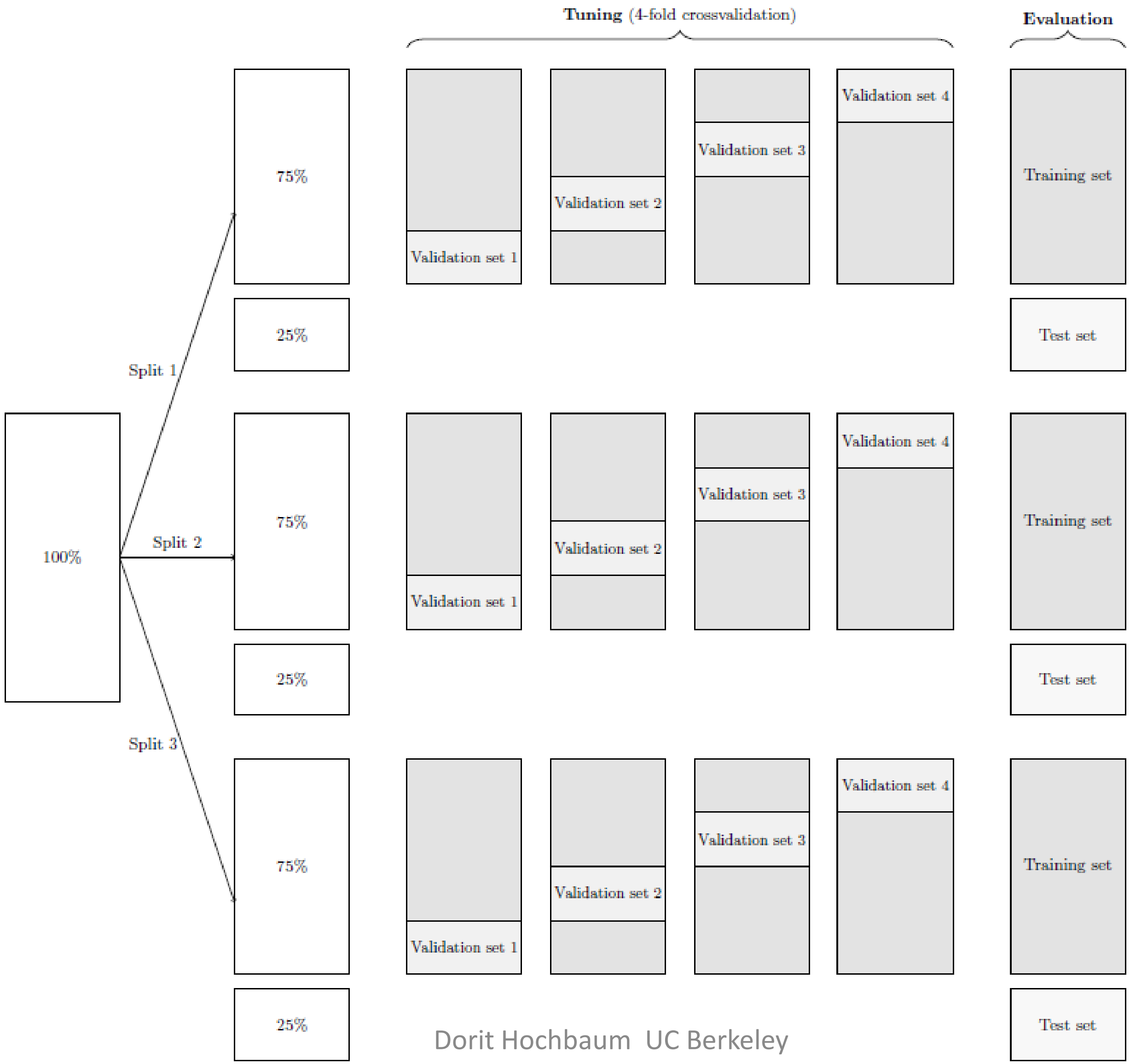
DATA SETS FROM UCI AND LIBSVM REPOSITORY

Abbr	Downloaded from	# Objects	# Attributes	# Positives	# Negatives	$\frac{\text{\# Positives}}{\text{\# Negatives}}$
IRS	LIBSVM	150	4	50	100	0.50
WIN	LIBSVM	178	13	59	119	0.50
PAR	UCI	195	22	147	48	3.06
SON	UCI	208	60	111	97	1.14
GLA	LIBSVM	214	9	70	144	0.49
HEA	LIBSVM	270	13	120	150	0.80
HAB	UCI	306	3	81	225	0.36
VER	UCI	310	6	210	100	2.10
ION	UCI	351	34	225	126	1.79
DIA	UCI	392	8	130	262	0.50
BCW	UCI	683	10	239	444	0.54
AUS	LIBSVM	690	14	307	383	0.80
BLD	UCI	748	4	178	570	0.31
FOU	LIBSVM	862	2	307	555	0.55
TIC	UCI	958	27	626	332	1.89
GER	UCI	1,000	24	300	700	0.43
CAR	UCI	2,126	21	1,655	471	3.51
SPL	LIBSVM	3,175	60	1,648	1,527	1.08
LE1	UCI	20,000	16	753	19,247	0.04
LE2	UCI	20,000	16	9,940	10,060	0.99

LOWER AND UPPER BOUNDS OF TUNING PARAM. VALUES

Abbr	Tuning parameter name	LB	UB	Type
ANN	Units in hidden layer	1	200	Integer
CART	Minimum leaf size	1	50	Integer
	Minimum parent size	2	25	Integer
ENSEM	Number of decision trees	2	1,000	Integer
LASSO	Regularization parameter λ_L	0.00	1.00	Real
LIN	Threshold	-0.50	0.50	Real
LOG	Threshold	0.25	0.75	Real
SVM	Polynomial (1) or radial basis function kernel (2)	1	2	Integer
	Degree of polynomial kernel	1	5	Integer
	Derivative param. of RBF kernel	0.01	1.00	Real
SVMR	Radial basis function kernel (2)	2	2	Integer
	Derivative param. of RBF kernel	0.01	1.00	Real
KNN	Parameter K	1	80	Integer
KSNC	Parameter K	1	3	Integer
	Weighting parameter λ	0.00	10.00	Real
	Scaling parameter ϵ	0.01	1.00	Real
SNC	Weighting parameter λ	0.00	1.00	Real
	Scaling parameter ϵ	0.01	1.00	Real

PARTITIONING OF DATA SETS



LOWER AND UPPER BOUNDS OF TUNING PARAM. VALUES

Abbr	Tuning parameter name	LB	UB	Type
ANN	Units in hidden layer	1	200	Integer
CART	Minimum leaf size	1	50	Integer
	Minimum parent size	2	25	Integer
ENSEM	Number of decision trees	2	1,000	Integer
LASSO	Regularization parameter λ_L	0.00	1.00	Real
LIN	Threshold	-0.50	0.50	Real
LOG	Threshold	0.25	0.75	Real
SVM	Polynomial (1) or radial basis function kernel (2)	1	2	Integer
	Degree of polynomial kernel	1	5	Integer
	Derivative param. of RBF kernel	0.01	1.00	Real
SVMR	Radial basis function kernel (2)	2	2	Integer
	Derivative param. of RBF kernel	0.01	1.00	Real
KNN	Parameter K	1	80	Integer
KSNC	Parameter K	1	3	Integer
	Weighting parameter λ	0.00	10.00	Real
	Scaling parameter ϵ	0.01	1.00	Real
SNC	Weighting parameter λ	0.00	1.00	Real
	Scaling parameter ϵ	0.01	1.00	Real

F1-SCORE, PRECISION, RECALL, ACCURACY

Let TP, TN, FP, and FN denote the number of true positives, true negatives, false positives, and false negatives, respectively. F_1-score, precision, recall, and accuracy are then defined as follows:

$$F_1\text{-score} = \frac{2TP}{2TP + FP + FN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN}$$

Note that the F_1-score is the harmonic mean of precision and recall. As some of the techniques do not have a probabilistic output, we do not report here the performance measure AUC (area under the curve).

NORMALIZED F1-SCORE (TESTED FOR SAME TUNING TIME)

	ANN	CART	ENSEM	LASSO	LIN	LOG	SVM	SVMR	KNN	KSNC	SNC	Avg
IRS	**100.0**	**100.0**	0.0*	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	90.9
WIN	80.1	9.2	0.0	64.0	82.5	**100.0**	78.7	78.7	82.5	62.2	23.0	60.1
PAR	33.6	20.3	79.7	27.2	48.9	24.3	0.0	50.2	**96.5**	**95.2**	**100.0**	52.4
SON	79.5	0.0	55.6	15.4	30.7	31.6	**100.0**	**100.0**	94.7	91.4	72.0	61.0
GLA	0.0	26.1	84.1	47.0	0.1	26.0	**100.0**	**100.0**	50.4	5.6	27.1	42.4
HEA	**100.0**	61.0	79.8	94.2	93.2	85.7	0.0	79.1	86.4	87.5	88.9	77.8
HAB	51.7	23.1	56.5	0.0	94.9	**100.0**	71.5	29.7	42.0	81.3	76.2	57.0
VER	**100.0**	73.7	90.1	**96.4**	61.4	94.8	0.0	51.4	84.0	49.5	42.6	67.6
ION	54.6	0.0	79.3	12.9	16.4	18.0	**100.0**	**100.0**	37.7	36.2	85.5	49.2
DIA	73.7	49.2	50.8	78.5	**100.0**	92.3	0.0	70.1	59.4	78.9	88.9	67.5
BCW	26.9	**100.0**	16.2	25.3	61.9	78.6	0.0	18.0	81.2	39.5	78.6	47.8
AUS	94.7	94.8	89.4	98.3	98.3	**99.1**	0.0	**100.0**	94.7	88.5	**98.9**	87.0
BLD	77.9	68.5	67.1	26.6	**100.0**	**97.1**	0.0	46.6	58.6	83.3	**95.6**	65.6
FOU	99.6	96.9	78.3	30.8	31.8	31.1	0.0	**100.0**	**100.0**	**100.0**	**100.0**	69.9
TIC	49.5	0.0	69.0	73.2	73.2	69.0	**100.0**	**100.0**	76.9	76.9	92.6	70.9
GER	39.9	0.0	65.8	86.8	**100.0**	90.8	2.8	19.8	19.7	27.2	62.3	46.8
CAR	69.8	84.3	**100.0**	74.4	58.2	69.4	0.0	88.1	82.9	83.6	78.5	71.7
SPL	48.8	**100.0**	90.0	32.5	22.6	21.7	68.4	71.2	0.0	28.8	46.6	48.3
LE1	44.6	87.9	71.2	0.0	0.0	0.0	97.5	**98.7**	**99.6**	**100.0**	94.1	63.1
LE2	72.3	75.2	36.5	0.0	5.6	7.0	82.5	**100.0**	98.0	**98.3**	**98.1**	61.2
Avg	64.9	53.5	66.3*	49.2	59.0	61.8	45.1	**75.1**	72.3	70.7	**77.5**	
Min	0.0	0.0	0.0*	0.0	0.0	0.0	0.0	18.0	0.0	5.6	**23.0**	

SNC achieves best and most robust performance across data sets

RANK OF TECHNIQUES BASED ON F1-SCORE

	ANN	CART	ENSEM	LASSO	LIN	LOG	SVM	SVMR	KNN	KSNC	SNC
IRS	**5.5**	**5.5**	11.0	**5.5**	**5.5**	**5.5**	**5.5**	**5.5**	**5.5**	**5.5**	**5.5**
WIN	4.0	10.0	11.0	7.0	2.5	**1.0**	5.5	5.5	2.5	8.0	9.0
PAR	7.0	10.0	4.0	8.0	6.0	9.0	11.0	5.0	2.0	3.0	**1.0**
SON	5.0	11.0	7.0	10.0	9.0	8.0	**1.5**	**1.5**	3.0	4.0	6.0
GLA	11.0	7.0	3.0	5.0	10.0	8.0	**1.5**	**1.5**	4.0	9.0	6.0
HEA	**1.0**	10.0	8.0	2.0	3.0	7.0	11.0	9.0	6.0	5.0	4.0
HAB	7.0	10.0	6.0	11.0	2.0	**1.0**	5.0	9.0	8.0	3.0	4.0
VER	**1.0**	6.0	4.0	2.0	7.0	3.0	11.0	8.0	5.0	9.0	10.0
ION	5.0	11.0	4.0	10.0	9.0	8.0	**1.5**	**1.5**	6.0	7.0	3.0
DIA	6.0	10.0	9.0	5.0	**1.0**	2.0	11.0	7.0	8.0	4.0	3.0
BCW	7.0	**1.0**	10.0	8.0	5.0	3.5	11.0	9.0	2.0	6.0	3.5
AUS	7.0	6.0	9.0	5.0	4.0	2.0	11.0	**1.0**	8.0	10.0	3.0
BLD	5.0	6.0	7.0	10.0	**1.0**	2.0	11.0	9.0	8.0	4.0	3.0
FOU	5.0	6.0	7.0	10.0	8.0	9.0	11.0	**2.5**	**2.5**	**2.5**	**2.5**
TIC	10.0	11.0	8.5	6.5	6.5	8.5	**1.5**	**1.5**	5.0	4.0	3.0
GER	6.0	11.0	4.0	3.0	**1.0**	2.0	10.0	8.0	9.0	7.0	5.0
CAR	8.0	3.0	**1.0**	7.0	10.0	9.0	11.0	2.0	5.0	4.0	6.0
SPL	5.0	**1.0**	2.0	7.0	9.0	10.0	4.0	3.0	11.0	8.0	6.0
LE1	8.0	6.0	7.0	10.0	10.0	10.0	4.0	3.0	2.0	**1.0**	5.0
LE2	7.0	6.0	8.0	11.0	10.0	9.0	5.0	**1.0**	4.0	2.0	3.0
Avg	6.03	7.38	6.53	7.15	5.97	5.88	7.20	**4.67**	5.33	5.30	**4.58**

STANDARD DEVIATION OF F1-SCORE ACROSS SPLITS

	ANN	CART	ENSEM	LASSO	LIN	LOG	SVM	SVMR	KNN	KSNC	SNC	Avg
IRS	**0.00**	**0.00**	**0.00***	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	0.00
WIN	1.99	3.32	5.57	3.61	**1.75**	**0.00**	2.14	2.14	**1.75**	1.94	5.32	2.68
PAR	3.32	**2.64**	4.43	2.97	3.93	**2.31**	4.08	3.02	**0.50**	3.46	4.05	3.16
SON	4.80	12.01	4.76	**1.44**	**1.62**	**3.06**	3.83	3.83	5.87	6.77	5.33	4.85
GLA	16.93	**6.22**	**3.89**	12.21	6.76	7.61	10.63	10.63	6.52	8.41	**6.37**	8.74
HEA	4.74	10.76	**3.11**	5.61	5.18	5.90	11.45	**3.86**	6.49	**4.73**	5.46	6.12
HAB	8.87	9.39	**3.48**	8.12	7.55	**3.59**	5.01	16.56	5.22	**2.71**	5.95	6.95
VER	2.22	**1.02**	**0.49**	2.23	4.53	2.35	2.46	1.83	1.90	**1.48**	3.66	2.20
ION	**0.18**	2.66	1.23	2.54	1.19	1.90	1.96	1.96	2.50	**1.12**	**1.14**	1.67
DIA	9.61	5.38	12.22	3.63	**2.71**	**3.46**	6.66	5.39	5.39	8.70	**1.65**	5.89
BCW	0.91	**0.26**	2.05	**0.18**	1.96	**0.60**	1.61	2.39	1.28	0.99	**0.60**	1.17
AUS	3.55	3.99	**1.30**	3.59	3.71	3.64	6.95	4.40	**2.90**	6.72	**3.24**	4.00
BLD	9.82	**2.32**	**1.97**	**2.66**	9.07	6.72	6.07	7.19	2.86	3.71	5.13	5.23
FOU	0.40	0.83	3.97	3.92	3.04	2.91	4.31	**0.00**	**0.00**	**0.00**	**0.00**	1.76
TIC	**0.81**	1.82	0.88	**0.69**	**0.69**	0.88	1.04	1.04	0.82	1.58	0.86	1.01
GER	9.17	9.00	**3.50**	**2.44**	3.56	3.58	3.99	5.78	6.96	4.54	**1.96**	4.95
CAR	**0.18**	0.66	0.43	0.68	0.57	0.51	3.82	**0.29**	0.52	0.33	**0.20**	0.74
SPL	2.49	0.78	1.02	0.88	0.94	1.00	**0.45**	**0.32**	1.57	**0.61**	2.19	1.11
LE1	37.77	2.19	2.21	**0.00**	**0.00**	**0.00**	1.07	0.32	0.53	2.49	1.86	4.40
LE2	0.83	0.29	0.52	0.61	0.18	0.14	**0.04**	0.17	**0.11**	**0.08**	0.13	0.28
Avg	5.93	3.78	3.00*	2.90	2.95	**2.51**	3.88	3.56	**2.68**	3.02	**2.75**	
Max	37.77	12.01	12.22*	12.21	9.07	**7.61**	11.45	16.56	**6.96**	8.70	**6.37**	

EVALUATION TIME [SEC]

	ANN	CART	ENSEM	LASSO	LIN	LOG	SVM	SVMR	KNN	KSNC	SNC
IRS	0.707	0.034	0.080	0.087	0.040	0.056	**0.008**	**0.008**	0.009	0.009	**0.008**
WIN	0.789	0.035	10.852	0.050	0.042	0.068	0.010	**0.008**	**0.009**	0.010	**0.008**
PAR	0.738	0.034	10.983	2.084	0.045	0.013	0.011	**0.010**	0.011	**0.011**	**0.010**
SON	0.729	0.034	11.355	0.028	0.041	**0.012**	0.019	0.016	0.012	**0.011**	**0.010**
GLA	0.678	0.033	10.761	0.028	0.041	**0.010**	0.011	**0.010**	0.011	0.011	**0.010**
HEA	0.633	0.033	10.805	0.028	0.042	**0.012**	23.942	**0.012**	0.012	0.013	**0.011**
HAB	0.759	0.032	10.547	0.024	0.040	**0.011**	23.322	**0.012**	**0.012**	0.014	0.014
VER	0.754	0.034	10.778	0.043	0.041	0.013	**0.011**	0.015	**0.013**	0.013	**0.012**
ION	0.694	0.035	11.130	0.369	0.059	**0.012**	0.022	0.022	**0.014**	0.016	**0.014**
DIA	0.784	0.032	11.021	0.030	0.042	**0.010**	40.735	**0.013**	**0.014**	0.020	0.018
BCW	0.719	0.033	11.330	0.029	0.044	**0.011**	**0.014**	**0.012**	0.023	0.042	0.039
AUS	0.698	0.037	11.474	**0.028**	0.045	**0.012**	45.930	**0.025**	0.032	0.044	0.041
BLD	0.915	0.035	11.261	**0.027**	0.042	**0.012**	105.442	0.050	**0.027**	0.047	0.046
FOU	1.238	0.035	11.383	**0.027**	0.040	**0.010**	48.607	0.043	**0.029**	0.059	0.063
TIC	1.416	**0.042**	11.587	0.193	**0.057**	0.139	0.093	0.092	**0.053**	0.087	0.080
GER	0.811	**0.046**	11.723	0.066	0.051	**0.019**	75.986	0.105	**0.049**	0.100	0.091
CAR	1.665	**0.055**	22.062	0.247	**0.072**	**0.138**	15.725	0.172	0.371	0.402	0.376
SPL	1.824	**0.081**	17.138	0.244	**0.140**	**0.143**	1.629	4.969	1.938	0.969	0.896
LE1	98.316	**0.141**	69.126	1.981	**0.170**	**0.213**	46.099	9.807	21.128	44.440	44.316
LE2	35.185	**0.323**	67.267	1.051	**0.171**	**0.177**	1,589.243	93.411	27.063	54.123	45.665
Sum	150.052	**1.164**	342.663	6.663	**1.266**	**1.089**	2,016.858	108.811	50.831	100.440	91.727

TUNING TIME [SEC]

	ANN	CART	ENSEM	LASSO	LIN	LOG	SVM	SVMR	KNN	KSNC	SNC
IRS	11	6	7	6	6	6	6	6	6	6	6
WIN	12	6	172	6	6	6	6	6	6	6	6
PAR	11	9	174	27	10	9	34	9	9	9	9
SON	11	9	179	42	9	10	9	9	9	9	9
GLA	11	9	173	9	9	9	9	9	9	9	9
HEA	21	12	174	12	12	12	64	12	12	12	12
HAB	22	15	179	15	15	15	65	15	15	15	15
VER	22	15	176	15	15	15	70	15	15	15	15
ION	22	21	177	21	21	22	21	21	21	21	21
DIA	35	24	175	24	24	24	104	24	24	24	24
BCW	58	54	178	54	54	54	132	54	54	54	54
AUS	59	54	180	54	54	54	128	54	54	54	54
BLD	62	60	177	60	60	60	323	60	60	60	60
FOU	92	75	180	75	75	75	231	75	75	75	75
TIC	106	90	184	91	90	91	90	91	90	90	90
GER	102	96	184	96	96	96	227	96	96	96	96
CAR	306	294	529	294	294	295	416	295	295	295	295
SPL	553	537	632	537	538	538	543	551	542	539	539
LE1	8,571	8,484	8,773	8,484	8,485	8,485	8,570	8,524	8,537	8,571	8,551
LE2	8,667	8,485	8,924	8,485	8,485	8,485	10,480	8,866	8,583	8,660	8,557
Sum	18,754	18,359	21,527	18,410	18,362	18,360	21,528	18,793	18,514	18,621	18,498

TAKE AWAYS

All pairwise comparisons' classification algorithms perform better than other methods

Challenge:

SNC and other data mining and clustering algorithms that perform well (e.g., KNN and SVM with kernels methods) require as input a similarity matrix

The number of pairwise similarities grows quadratically in the size of the data sets

SPARSE COMPUTATION FOR LARGE-SCALE DATA MINING WITH PAIRWISE COMPARISONS

Known Literature:

Existing sparsification approaches require complete matrix as input

-> not applicable for massively large datasets

Proposed methodolgy:

Sidesteps the computationally expensive task of constructing the complete similarity matrix

Generating only the relevant entries in the similarity matrix without performing pairwise comparisons

SPARSE COMPUTATION WITH APPROXIMATE PCA

Input: Data set as an *n x d* matrix *A* containing *n* objects with *d* attributes:

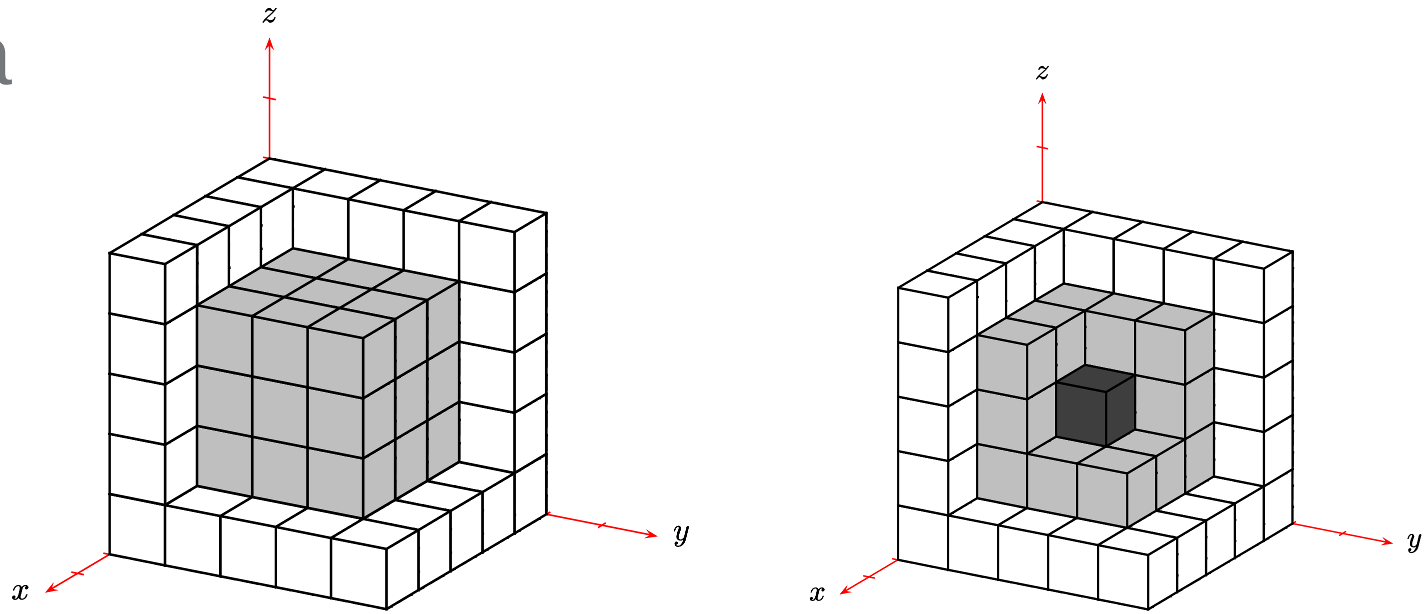
Output: Sparse *n x n* similarity matrix

Procedure:

1. Embed *d*-dimensional space in a *p*-dimensional space for *p*<<*d* with the use of approximate Principal Component Analysis (PCA) – based on ConstantTimeSVD of Drineas, Kannan, and Mahoney (2006). Pick *r* rows/objects of the matrix/dataset
2. Subdivide the range of values in each dimension into *k* intervals of equal length (can use a different number of intervals in each dimension
3. Assign each object to a single block based on its *p* entries. O(1) work per object
4. Compute distances between objects that are assigned to the same block in original *d*-dimensional space
5. Identify neighboring blocks and compute similarities between objects in those blocks.

BLOCK DATA STRUCTURE FOR SPARSIFICATION

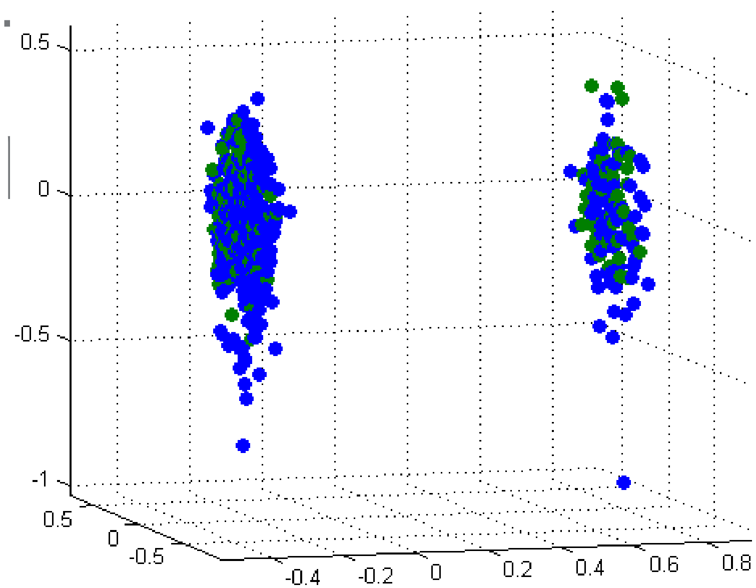
Example of block data structure in the space of the *p* = 3 leading principal components. Here the grid resolution *k* = 5 and the length of the intervals is the same for a



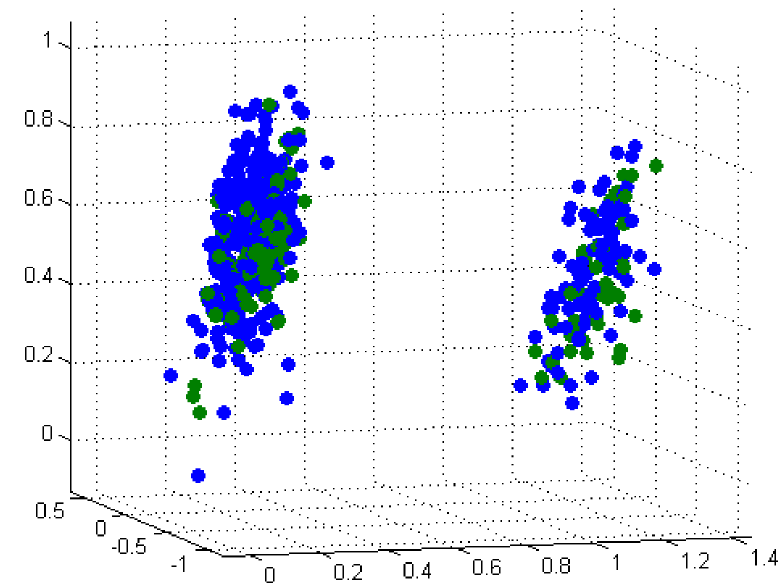
EFFECTIVENESS OF APPROXIMATE-PCA

Data set with 583 objects and 10 attributes

Blue dots represent 416 liver patients and green dots represent 167 non-liver patients



Exact PCA



Approximate-PCA with *r*=5

EMPIRICAL ANALYSIS: LARGE SCALE DATASETS

Source: Machine Learning Repository of the University of California at Irvine

Selection criteria:

- Thousands of objects
- Data from different domains
- Balanced and unbalanced data sets

Abbr	Domain	Attribute types	# Objects	# Attributes	# 1-Labels	# 0-Labels
CAR	Cardiotocography	Real	2,126	21	471	1,655
LE1	Letter recognition	Integer	20,000	16	753	19,247
LE2	Letter recognition	Integer	20,000	16	9,940	10,060
BAN	Bank marketing	Binary, Real	45,211	51	5,289	39,922
ADU	Income prediction	Binary, Integer	45,222	88	11,208	34,014
CO1	Forest cover types	Binary, Integer	46,480	54	16,947	29,533
CO2	Forest cover types	Binary, Integer	581,012	54	211,840	369,172

EXPERIMENTAL DESIGN

Tuning:

- Grid search
- Exponential similarity
- Tuning parameters:
  - Epsilon = {1,...,30}
  - Lambda = $\{10^{-5},...,10^{-1}\}$
  - Normalization of input data
- Sub-sampling validation
- Complete similarity matrix

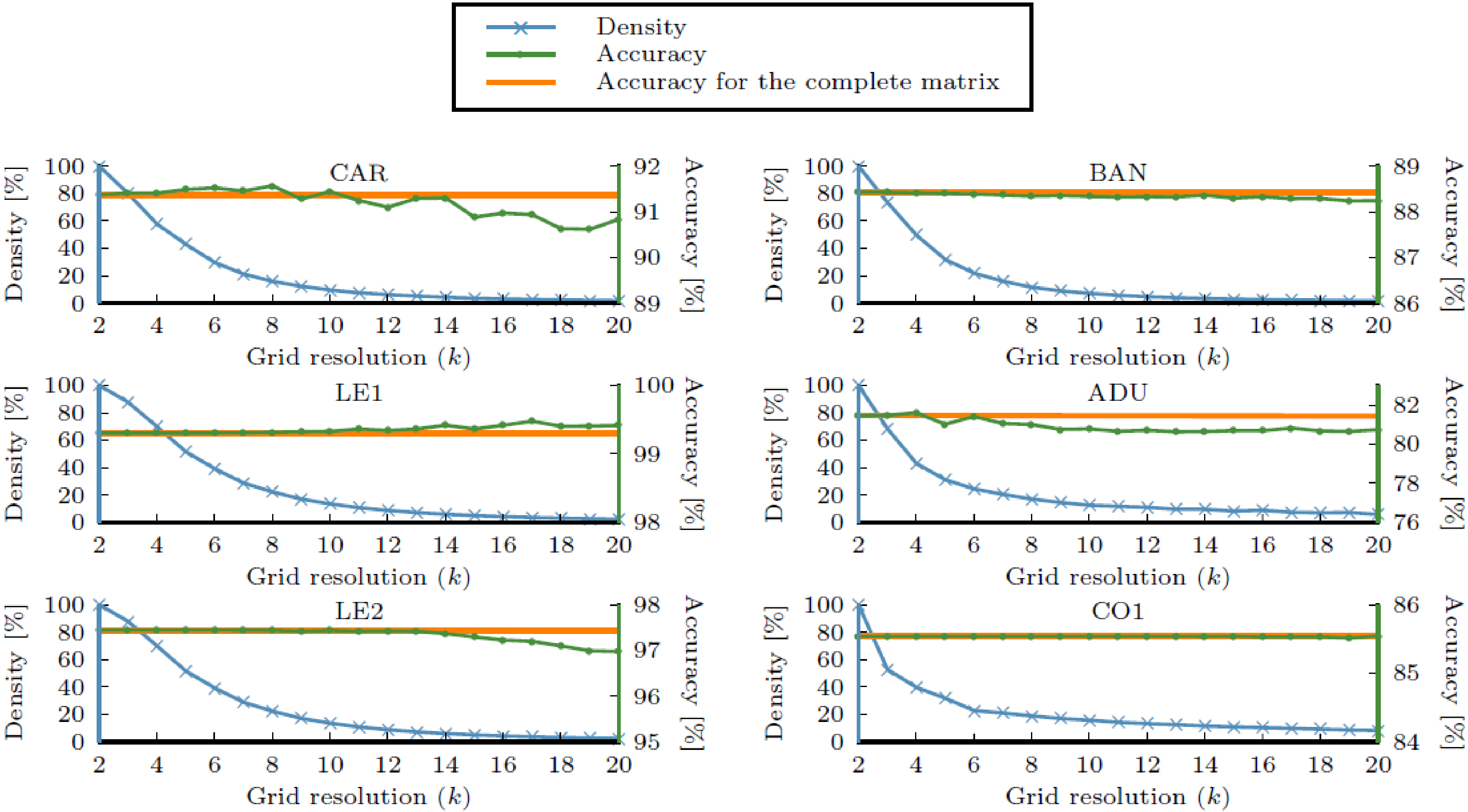
Testing:

- Number of rows for appr.-PCA
  - CO2: *r = 100*
  - Remaining sets: *r* = 30
- Value for grid resolution *k*
  - CAR, LE1, LE2: *k* = {2,...,20}
  - ADU, BAN, CO1: *k* = {3,...,30}
  - CO2: *k* = {100,...,500}
- *k*=2 corresponds to complete similarity matrix
- *k*>2 generates sparse similarity matrix

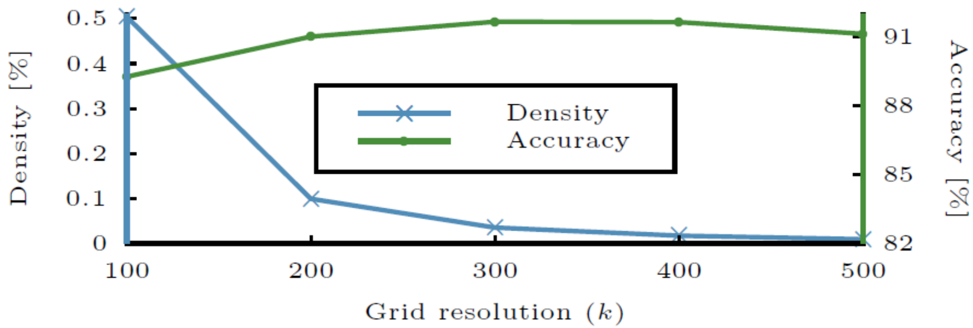
Implementation: Matlab and C

Machine: Workstation with two Intel E5-2687W (3.1 GHz) and 128 GB RAM

COMPUTATIONAL RESULTS FOR ALL DATA SETS EXCEPT CO2



RESULTS FOR DATA SET CO2:



	k	ACC [%]	DEN [%]	RAM [GBs]	CPU matrix [s]	CPU cut [s]
complete matrix	2	na	100.000	1728.39	na	na
sparse matrices	100	89.27	0.504	8.71	1221.90	23.05
	200	91.02	0.099	1.71	401.64	3.99
	300	91.67	0.035	0.60	726.72	1.29
	400	91.65	0.017	0.29	1121.40	0.56
	500	91.14	0.009	0.15	1546.08	0.27

- Accuracy achieved with the very sparse similarity matrices very similar to accuracies obtained based on complete similarity matrix
- Accuracy changes little with increasing grid resolution
- Running time decreases substantially (roughly proportional to density)
- CO2: Accuracy of 89.72% possible with density of 0.008%. Complete similarity matrix would contain over 54 billion entries

SUMMARY

- A clustering/classification optimization model and a combinatorial optimization algorithm that uses pairwise comparisons
- Effective for general classification tasks as well as for specific application contexts
- Efficient in theory and in practice
- The approach of sparse computation enable the use of the method for massively large data sets.

SELECTED REFERENCES

D.S. Hochbaum. [Polynomial time algorithms for ratio regions and a variant of normalized cut.](#) *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32 (5), 889-898, May 2010.

D.S. Hochbaum. A polynomial time algorithm for Rayleigh ratio on discrete variables: Replacing spectral techniques for expander ratio, normalized cut and Cheeger constant. *Operations Research* , Vol. 61:1, 2013, pp. 184-198

D.S. Hochbaum, C.N. Hsu, Y.T. Yang. [Ranking of Multidimensional Drug Profiling Data by Fractional Adjusted Bi-Partitional Scores.](#) *Bioinformatics* (2012) 28:12, pp. 106-114.

D.S. Hochbaum, C. Lu, E. Bertelli. ["Evaluating Performance of Image Segmentation Criteria and Techniques".](#) *EURO Journal on Computational Optimization*, Vol 1:1-2, May 2013, pp. 155-180.

D.S. Hochbaum, P. Baumann and Y. T. Yang. A comparative study of machine learning methodologies and two new effective optimization algorithms. EJOR, 2019.

D.S. Hochbaum and P. Baumann. [Sparse Computation for Large-Scale Data Mining.](#) *IEEE Transactions on Big Data*, Vol 2, Issue 2, 151-174, 2016.

QUESTIONS

Dorit S. Hochbaum

hochbaum@ieor.berkeley.edu

http://www.ieor.berkeley.edu/~hochbaum/