Clustering with Hypergraphs: The Case for Large Hyperedges

Pulak Purkait[a] Tat-Jun Chin[a] Hanno Ackermann[b] David Suter[a]
[a]The University of Adelaide, [b] Leibniz Universität Hannover
{pulak.purkait, tjchin, dsuter}@adelaide.edu.au
hanno.ackermann@tnt.uni-hannover.de

THE UNIVERSITY of ADELAIDE

Contribution

In this work, we show that large hyperedges are better from both theoretical and empirical standpoints. We then propose a novel guided sampling strategy for large hyperedges, based on the concept of *random cluster models*. Our method can generate pure large hyperedges that significantly improve grouping accuracy without exponential increases in sampling costs.

Problem

The *normalized cut* criterion for partitioning the set of vertices of a hypergraph $\mathcal{H} = (V, E)$ into (S, S^c) is

$$\text{ncut}(S, S^c) = \text{vol}(S, S^c)\left(\frac{1}{\text{vol}(S)} + \frac{1}{\text{vol}(S^c)}\right).$$

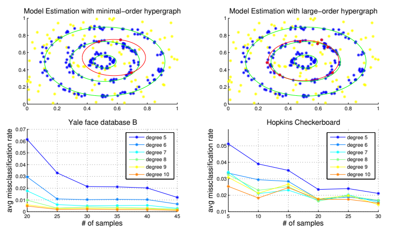
where the volume or cost of the cut $\text{vol}(S, S^c)$ is

$$\text{vol}(S, S^c) = \sum_{e \in c(S,S^c)} w(e)\frac{|e \cap S||e \cap S^c|}{\delta(e)}.$$

An overwhelming majority of the previous works that utilized the hypergraph formalism limited the hyperedges of the smallest possible size.

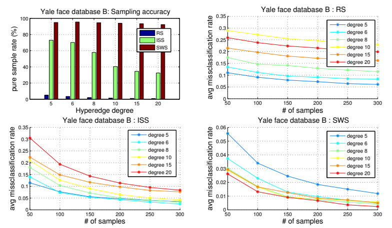
Why use large hyperedges?

1. The cut value $\text{vol}(S, S^c)$ is *always higher for larger hyperedges*, since the numerator increases quadratically while the denominator increases linearly. Hence, given two hyperedges of the same weight $w(e)$ and the same size ratio, *NCut will inherently favour preserving the larger hyperedge and cutting the smaller hyperedge.*
2. The preference can be rationalized since larger hyperedges convey more evidence on the existence of a cluster than smaller hyperedges.



(a)-(b) A typical sample for a circle fitting problem of sizes 3 and 8 respectively. (c)-(d) missclassification rate with number of pure samples.

The effects of large hyperedges



(a) Sampling accuracy for different degree D, (b)–(d) Misclassification rate of three methods on Yale Face Database B. For RS and ISS in the Figs.(b)–(c) the miss-classification rate increases with D; this reflects the inability of these methods to sample large pure hyperedges, and *not* because large hyperedges are ineffective.

Random Cluster Model(RCM)

The Swendsen-Wang method introduces the binary "bond" variables $d = \{d_e\}$ for each edge e of the auxiliary graph $G^{(0)}$ to yield the extended Potts model

$$P(f, d) = \frac{1}{Z'} \prod_{e=<i,j>\in E^{(0)}} g(f_i, f_j, d_e),$$

where the factor g is defined as

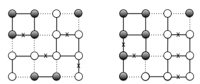
$$g(f_i, f_j, d_e) = \begin{cases} 1 - p_e & \text{if } d_e = 0, \\ p_e & \text{if } d_e = 1 \text{ and } f_i = f_j, \\ 0 & \text{if } d_e = 1 \text{ and } f_i \neq f_j. \end{cases}$$

A realization of (f, d) effectively partitions the vertices into a set of connected components. Marginalizing d in returns the Potts model, while marginalizing f in yields the RCM $P(d)$.

Proposed Framework: Swendsen-Wang sampling

Input: Data V, number of clusters K, hyperedge degree D, iteration count T and M, threshold ϵ.
Output: A set of hyperedges E of degree D for hypergraph $\mathcal{H} = (V, E)$.

1. Obtain auxiliary graph $G^{(0)} = (V, E^{(0)})$.
2. Initialize $f^{(1)}$ to a constant value.
3. For $t = 1, 2, \ldots, T$
   (a) From $f^{(t)}$, obtain clustering $\mathcal{C}^{(t)}$ and corresponding graph $G^{(t)} = (V, E^{(t)})$.
   (b) Repeat M times
      i. For all $e \in E^{(t)}$, turn off d_e with probability $1 - p_e$. This divides each $\mathcal{C}_k^{(t)}$ into a set of sub-clusters.
      ii. Collect all sub-clusters from $\{\mathcal{C}_k^{(t)}\}_{k=1}^K$ into $\mathcal{CP}$.
      iii. Remove from $\mathcal{CP}$ all components of size less than $D - 1$.
      iv. Select a component $\mathcal{P}_s$ from $\mathcal{CP}$ with probability $q(\mathcal{P}_s|\mathcal{CP})$, then randomly select a $(D-1)$-subset s from $\mathcal{P}_s$.
      v. Fit the model onto s and evaluate it against all data in V. Add the newly generated hyperedges to the set of all hyperedges.
   (c) Apply NCut on the current hypergraph $\mathcal{H}$ to obtain labels $f^{(t+1)}$.
   (d) If the difference between $f^{(t)}$ and $f^{(t+1)}$ is smaller than ϵ, terminate sampling.



Clustering result on Yale face B dataset

Experimental setup: we use Govindu's [1] dense sample reuse approach. For our method (SWS followed by NCut), we chose degree $D = 20$ with 300 samples.

K	2	3	4	5	6	7	8	9	10	time(s)
GPCA	0.0	49.5	0.0	26.6	9.9	25.2	28.5	30.6	19.8	$\approx 10^6$
SCC	0.0	0.0	0.0	1.1	2.7	2.1	2.2	5.7	6.6	4.93
SSC	0.0	0.0	0.0	0.0	0.0	0.0	0.0	2.4	4.6	6.12
SLBF	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.2	0.9	1.72
ALC	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1878.56
Ours	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.74

Table. 1 Mean percentage of misclassification on clustering Yale face B dataset.

Motion segmentation with sparse trajectories

The dataset we used the Hopkins 155 dataset.
Types of motions : checkerboard, traffic and articulated.

Experiment setup We used D = 10 and 50 samples. We observe that on average 98.34% the hyperedges generated are clean.

Comparison against state-of-the-arts Unlike our method, the chosen state-of-the-art techniques are dedicated to motion segmentation.

	Two Motions						Three Motions							
Method	Chck.(78)		Trfc.(31)		Artc.(11)		Chck.(26)		Trfc.(7)		Artc.(2)		all(155)	
	MN	MD	MN	MD	MN	MD	MN	MD	MN	MD	MN	MD	MN	MD
GPCA	6.09	1.03	1.41	0.00	2.88	0.00	31.95	32.93	19.83	19.55	16.85	16.85	10.34	2.54
LSA	2.57	0.27	5.43	1.48	4.10	1.22	5.80	1.77	25.07	23.79	7.25	7.25	4.94	0.90
ALC	1.55	0.29	1.59	1.17	10.70	0.95	5.20	0.67	7.75	0.49	21.08	21.08	3.56	0.50
SCC	1.31	0.06	1.02	0.26	3.21	0.76	6.31	1.97	3.31	3.31	9.58	9.58	2.42	NA
SSC	1.12	0.00	0.02	0.00	0.62	0.00	2.97	0.27	0.58	0.00	1.42	1.42	1.24	0.00
DCT	0.71	0.00	0.05	0.00	0.96	0.00	2.44	1.29	0.05	0.00	1.60	1.60	0.87	NA
Ours	1.86	0.00	0.08	0.00	1.13	0.00	2.72	0.00	0.00	0.00	1.06	1.06	1.50	0.00
MN: mean, MD: median, NA: not available														

Table. 2 Mean percentage of misclassification error on Hopkins 155 dataset.

Motion segmentation with dense trajectories

The dataset Barkhley Motion segmentation dataset [2]:
Comparison against state-of-the-arts Brox and Malik [2] follows conventional clustering approach which constructs pairwise affinities, that restricts the motions to be 2D translations.
Ochs and Brox (OB) [3] used 2D similarity motion model $p = 2$. The bottleneck is for a typical `car1` sequence with 4850 trajectories $4850 \times 4850 \times (30 + 12)$ ($\approx 10^9$) hyperedges are used.
In our method, the hyperedges of degree of $D = 10$ were used. Based on SWS, we only generated 1000 samples and for each sample dense affinity is computed for all overlapping trajectories. Since SWS produces pure hyperedges accurately, this relatively few samples were sufficient to approximate the affinity matrix. This reduces run time enormously (10 seconds against 48 minutes for `car1` sequence).

Brox and Malik [2] Ochs and Brox [3] Our method Brox and Malik [2] Ochs and Brox [3] Our method
Fig. 1: Segmentation results of sequences *car5*, *marple8*, *marple13* and *duck*.

Conclusions

We have established theoretically and empirically the benefits of using large hyperedges in hypergraph clustering– this departs from previous methods that have exclusively used the smallest possible hyperedge.
Our message through this paper is for higher order clustering use large size samples and use a guided sampling technique to sample them.

References

[1] Govindu, V.M.: A tensor decomposition for geometric grouping and segmentation. In: CVPR 2005
[2] Brox, T., Malik, J.: Object segmentation by long term analysis of point trajectories. In: ECCV 2010
[3] Ochs, P., Brox, T.: Higher order motion models and spectral clustering. In: CVPR 2012
Acknowledgment: This work was supported by ARC grant DP130102524.