A Probabilistic Method for Pruning CADJ Graphs with Applications to SOM Clustering

Josh Taylor[1(✉)] and Erzsébet Merényi[2]

[1] Department of Statistics, Rice University, Houston, TX 77005, USA
jtay@rice.edu
[2] Department of Statistics and Department of Electrical and Computer Engineering, Rice University, Houston, TX 77005, USA
erzsebet@rice.edu

Abstract. We introduce a Bayesian Dirichlet-Multinomial model of the edge weights of the Cumulative ADJacency ($CADJ$) graph [1] with the goal of intelligent graph pruning. As a topology representing graph, $CADJ$ is an effective tool for cluster extraction from the learned prototypes of SOMs, but for complex data the graph must typically be pruned to elicit meaningful cluster structure. This work is a first attempt to guide this pruning in a formal modeling framework. Our model, dubbed `DM-Prune`, earmarks edges for removal via comparisons to a *novel null model* and provides an internal assessment of information loss resulting from iterative removal of edges. We show that `DM-Pruned` $CADJ$ graphs lead to clusterings comparable to the best previously achieved on highly structured real data.

Keywords: SOM clustering · Topology representing network · Graph sparsity

1 Introduction: The $CADJ$ Graph

$CADJ$ was introduced in [1] as a weighted version of the Induced Delaunay Triangulation of Martinetz and Schulten [2]. As a subgraph of the Delaunay Triangulation, $CADJ$ represents topological adjacencies of the prototypes $\{\boldsymbol{w}_i\}_{i=1}^{W}$ of a vector quantizer. Given data $X = \{\boldsymbol{x}_s\}_{s=1}^{N}$ drawn from manifold $\mathcal{M} \subseteq \mathbb{R}^d$, the $CADJ$ weight of the edge connecting prototypes $\boldsymbol{w}_i$ and $\boldsymbol{w}_j$ is $CADJ_{ij} = \#\{\boldsymbol{x} \in X : BMU1(\boldsymbol{x}) = i, BMU2(\boldsymbol{x}) = j\}$, where $BMU1(\cdot)$ and $BMU2(\cdot)$ return the index of the Best and second-Best Matching Unit (or prototype, respectively) of the datum $\boldsymbol{x}$. Positive $CADJ_{ij}$ values reflect the strength of topological connectivity of prototypes $\boldsymbol{w}_i$ and $\boldsymbol{w}_j$ (and, consequently, connectivity of the portion of the manifold $\mathcal{M}$ they represent), whereas values of 0 indicate disconnected portions of the manifold. When used in conjunction with the learned prototypes of a Self-Organizing Map [3], $CADJ$–in particular its

A. Vellido et al. (Eds.): WSOM 2019, AISC 976, pp. 44–54, 2020.
https://doi.org/10.1007/978-3-030-19642-4_5

symmetrized version $CONN = CADJ + CADJ^T$ along with the *CONNvis* visualization ([1], an example of which is depicted in Fig. 1)–has been a successful tool for cluster discovery ([1,4] and references therein) via identification of its closed, connected communities. However, the *CONN* graph representing complex, high-dimensional datasets typically does not contain readily identifiable communities. In [1] a method for inducing sparsity in *CONN* is proposed to remedy this by evaluating *global* and *local* importance of edge weights relative to each other, removing edges by thresholding well-understood graph parameters on a grid, and judging the impact by post-evaluation of the quality of extracted clusters at the various levels of thresholding. This is tedious, and in the current framework the analyst has little formalized feedback regarding the effects of the thresholding on the representation of the data manifold as a whole. This work is a first attempt to provide formal, principled suggestions for thresholding, before extracting clusters. We will earmark connections for removal by comparing their strengths to those which would be expected if the data were generated under uniform noise; a metric afforded to us by `DM-Prune` will report the impact of their successive removal as a means of locating a stopping point in the pruning process. Our goal is toward intelligent automation of the pruning process.

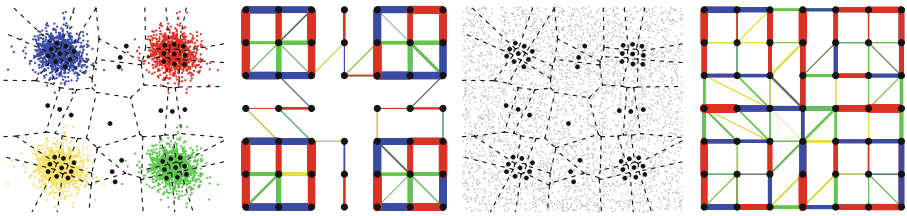


Fig. 1. (a) Data drawn from four Gaussian clusters (colored points, 1000 draws per cluster) and their learned SOM prototypes (black points). The Voronoi tessellation is outlined in dashed black lines. **(b)** The *CONNvis* visualization of the Gaussian data mapped to its SOM lattice, clearly outlining the four clusters (see [1] for an explanation of the line colorings and widths). **(c)** A sample of 4000 uniformly distributed points depicted against the same Voronoi tessellation as (a). **(d)** The *CONNvis* visualization of the uniform data recalled on the SOM learned by the Gaussian data, highlighting the structure expected from the Uniform Null Model $\mathcal{U}$

2 A Probabilistic Model for *CADJ*

By definition, the edge weights $CADJ_{ij}$ are counts of N observations falling into "bins," where the bins are the second-order Voronoi cells V_{ij} generated by the tessellation induced by $\{\boldsymbol{w}_i\}$ (see [5] for a more complete reference for higher-order Voronoi tessellations). To simplify the notation, let $k \in \{1, \ldots, K\}$ re-index the $(i, j) : CADJ_{ij} > 0$, so that K is the number of non-empty cells V_{ij}. Similarly, let $n_k = CADJ_{ij}$, $N = \sum_{\mathrm{k}=1}^{\mathrm{K}} n_k$; then the collection $\boldsymbol{n} = (n_1, n_2, \ldots, n_K)$

of counts in our K bins (categories) can be modeled via the Multinomial distribution with parameters $\boldsymbol{p} = (p_1, \ldots, p_K)$, which give the probabilities of observation $\boldsymbol{x}$ falling into bins $1, \ldots, K$.

For our data X, the true probability of bin k is $p_k = \int_{V_k} f_{\mathcal{M}}(\boldsymbol{x})$, where $f_{\mathcal{M}}(\boldsymbol{x})$ is the probability density of manifold $\mathcal{M}$ from which X was drawn. Since $f_{\mathcal{M}}(\boldsymbol{x})$ is unknown we must estimate $\boldsymbol{p}$ in order to parameterize the Multinomial distribution. This is easily accomplished in a Bayesian framework by appealing to the conjugacy of the Dirichlet distribution to the Multinomial. The Dirichlet distribution of dimension K, denoted $Dir(\boldsymbol{\alpha})$, is parameterized by vector $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_K)$, where all $\alpha_k > 0$, and describes probability distributions over the $K-1$ dimensional unit simplex (i.e., $\{\boldsymbol{p} : \sum_{k=1}^{K} p_k = 1\}$). We note for later use that each marginal p_k has a Beta distribution with parameters $\beta_1 = \alpha_k$ and $\beta_2 = \alpha_0 - \alpha_k$ (where $\alpha_0 = \sum_{\mathrm{k}=1}^{\mathrm{K}} \alpha_k$).

Invoking Dirichlet-Multinomial (DM) conjugacy, the posterior distribution of unknown $\boldsymbol{p}$ after observing multinomial count data $\boldsymbol{n}$ with prior distribution $Dir(\boldsymbol{\alpha})$ is also Dirichlet, but with modified posterior parameter $\bar{\boldsymbol{\alpha}} = \boldsymbol{\alpha} + \boldsymbol{n}$. Under the DM setup, the marginal distribution of the observed counts (i.e., the data likelihood, or Bayesian *evidence*) is given by [6]

$$f_{DM}(\boldsymbol{n}|\boldsymbol{\alpha}) = \int f_M(\boldsymbol{n}|\boldsymbol{p}, N) f_D(\boldsymbol{p}|\boldsymbol{\alpha})\, d\boldsymbol{p} = \frac{\Gamma(\alpha_0)\Gamma(N+1)}{\Gamma(N+\alpha_0)} \prod_{\mathrm{k}=1}^{\mathrm{K}} \frac{\Gamma(n_k+\alpha_k)}{\Gamma(\alpha_k)\Gamma(n_k+1)}. \quad (1)$$

The Dirichlet parameter $\boldsymbol{\alpha}$ controls both the mean and covariance of the resulting distribution, and consequently affects our estimation of $\boldsymbol{p}$. A typical choice for an uninformative prior has $\alpha_k = a = 1 \,\forall\, k$, meaning the resulting Dirichlet distribution considers all probability vectors equally likely. As $a \rightarrow \infty$ the Dirichlet more strongly favors uniform $\boldsymbol{p}$. Our choice of prior will be influenced by our choice of learning algorithm, where the prototypes $\{\boldsymbol{w}_i\}$ are placed via the *Conscience*-SOM (CSOM) algorithm of DeSieno [7], which aims to produce a maximum entropy mapping $\mathcal{M} \rightarrow \{\boldsymbol{w}_i\}$. Because of this, we have reason to expect equi-probable $\boldsymbol{p}$ and set $\alpha_k = a > 1 \,\forall\, k$. In the experiments which follow, $\alpha_k = 10$ was used as a default.

3 A Multi-focal View

The model in Sect. 2 was constructed by considering all $CADJ$ connections simultaneously, which we refer to as a *global* view of manifold connectivity (and denote by subscripts G in what follows). However, depending on characteristics of the manifold as represented by the prototypes $\{\boldsymbol{w}_i\}$, *local* connectivity might be more informative at certain points of the manifold than others, and vice-versa. In our framework, we have a separate local model of connectivity emanating from each (non-empty) Voronoi cell $i \in 1, \ldots, W$, with corresponding local versions of our quantities of interest: K_i = the number of bins in each local model, which is given by the number of $CADJ$ neighbors of prototype i; $\boldsymbol{n}_i = (n_{i,1}, \ldots, n_{i,K_i})$ are the observed counts of these K_i bins; $N_i = \sum_{j=1}^{K_i} n_{i,j}$; $\boldsymbol{\alpha}_i$ and $\bar{\boldsymbol{\alpha}}_i$ denote the vectors of prior and posterior parameters associated with

these local models, respectively. We stress that only the probability space (the number of bins K) changes between the global and local views, not the counts or parameters. In what follows, an expression $f_{DM}(\boldsymbol{n}|\boldsymbol{\alpha})$ should be interpreted as evaluating the DM density of the global model at observation $\boldsymbol{n}$ with parameter $\boldsymbol{\alpha}$, and $f_{DM}(\boldsymbol{n}_i|\boldsymbol{\alpha}_i)$ denotes evaluation of the DM density of local model i, at counts $\boldsymbol{n}_i$ with parameters $\boldsymbol{\alpha}_i$.

4 The Λ Metric

The evidence provided by the DM model (1) for our $CADJ$ values provides a natural starting point to monitor the degradation caused as we invoke sparsity into the prototype connectivity graph and, as discussed in the previous section, we would like to simultaneously monitor both the global (G) and overall local (L) nature of these impacts. In our setup, removing a $CADJ$ connection k, can be represented by sparsifying the $\boldsymbol{\alpha}$ parameters of our model (i.e., by setting α_k to some small value $\epsilon \approx 0$, we used $1e-20$). The impact of pruning can then be monitored via the Bayesian odds of the sparse prior model to the full, which forms the basis of our metric Λ.

The pruning process will be iterative over time t. Before beginning, we assign ranks r_k to each connection, which denote its order of removal (by convention, $r_k \in \{1, \ldots, R\}$, where $R \leq K$ is the number of distinct ranks and $r_k = 1$ means connection k is slated for removal at time $t = 1$). Let $\rho_t = \{k : r_k \leq t\}$, so that ρ_t returns the set of indices slated for removal at all times $\leq t$. Since, for us, removing connections corresponds to sparsifying $\boldsymbol{\alpha}$, we also define $\alpha_k(t) = \{\epsilon \text{ if } k \in \rho_t,\ \alpha_k \text{ else}\}$ and their vectorized versions $\boldsymbol{\alpha}(t)$ and $\boldsymbol{\alpha}_i(t)$. The impacts on our global and local models from pruning at step t are

$$\Lambda_G(t) = \left(\frac{f_{DM}(\boldsymbol{n}|\boldsymbol{\alpha}(t))}{f_{DM}(\boldsymbol{n}|\boldsymbol{\alpha}(0))}\right)^{1/N} \quad ; \quad \Lambda_L(t) = \prod_{i=1}^{W}\left(\frac{f_{DM}(\boldsymbol{n}_i|\boldsymbol{\alpha}_i(t))}{f_{DM}(\boldsymbol{n}_i|\boldsymbol{\alpha}_i(0)}\right)^{1/N_i} \tag{2}$$

and we aggregate these two views into a combined measure $\Lambda = \Lambda_G \times \Lambda_L$. Λ_G and Λ_L (and, consequently Λ) describe the evidence of the sparse model, relative to the full model. To account for the different natural scales of the data likelihoods and produce a comparable global and local view we have normalized each of the above odds by their effective sample size. We call the curve traced by $\Lambda(t)$ over time the Λ-Path and monitor its trajectory as we successively prune.

An example of a Λ-Path is given in Fig. 2 displaying the global, local and combined relative likelihoods of pruning over time. For viewing purposes, all three curves have been scaled from $[\Lambda(R), 1]$ to $[0, 1]$ (the likelihood of the fully pruned model at $t = R$ is not 0). Red dotted guidelines show selected % levels (e.g, every 5%) of the total likelihood retained, along with their corresponding steps t, at the bottom of the path plot. These guidelines suggest where to prune the graph in order to conserve a particular level of the likelihood. Note that a loss in data *likelihood* is not necessarily proportional to a total loss in data (or connections) from the model: certain connections are more important than

others, as will be reflected by their evidence (1). Of particular interest to us is the point of intersection where Λ_G and Λ_L attain equal relative evidence, since removing a connection harms its corresponding local likelihood more than it does Λ_G. At the point $t^* : \Lambda_G(t^*) = \Lambda_L(t^*)$, we have damaged our local model fit to the point where each of the component views contributes equal evidence; we postulate this might serve as an upper bound on t when implementing this pruning procedure.

5 Ranking Connections for Removal

The thresholding procedure described in [1] utilizes relative strengths of the observed $CADJ$ counts themselves as a guide for thresholding, with low-ranking connections (either globally or locally) removed first. This is essentially a rank analysis, where the justification for removing a certain connection is derived by comparing its strength (i.e., weight) to others, at different focal areas (globally or locally). Our basis for comparison will be informed by conditioning our notion of "strength" on a different set, which we call the Uniform Null Model $\mathcal{U}$. Specifically, we take $\mathcal{U}$ to be a fictitious, unobserved manifold that has Uniform measure $f_{\mathcal{U}}(\boldsymbol{x})$ over the support of $\mathcal{M}$. Under $\mathcal{U}$, the probabilities associated with each second-order Voronoi cell V_k are given by $q_k = \int_{V_k} f_{\mathcal{U}}(\boldsymbol{x})\, d\boldsymbol{x}$. Using the q_k we construct, for each connection k and its estimated probability distribution p_k, the quantity $Q_k = P_B(p_k > q_k)$, which serves as an indicator of strength relative to $\mathcal{U}$. The measure $P_B(\cdot)$ denotes probability with respect to k's marginal Beta distribution (see Sect. 2). Obviously, $0 \leq Q_k \leq 1$, where values ≈ 0 (or 1) indicate posterior bin probabilities much less (or greater) than would be expected under $\mathcal{U}$.

This choice of null model compares our observed $CADJ$ graph, which represents structured $\mathcal{M}$, to a $CADJ$ (with respect to the same set of prototypes $\{\boldsymbol{w}_i\}$) derived from data that has no structure at all. Namely, we are trying to distinguish observed structure in $CADJ$ from that which would result from uniform noise. Due completely to the *size* (volume) of the second-order Voronoi cells V_k, uniform noise (such as is displayed in Fig. 1c) can still produce the appearance of structure in the $CONNvis$ visualization (shown in Fig. 1d, which was generated by recalling the uniform noise using the SOM trained on the Gaussian clusters in Fig. 1a, and recording their $BMU1$ and $BMU2$). In our context, connections k whose Q_k values are the lowest are prime candidates for pruning (and are assigned rank $r_k = 1$, putting them first in line for removal). Under our multi-focal view we actually must consider $Q_{G,k}$ and $Q_{L,k}$ for each connection k, as the global and local models dictate separate marginal distributions. The ranking r_k is defined as the geometric mean of the two so $Q_k = \sqrt{Q_{G,k} \times Q_{L,k}}$ and $r_k = \text{rank}(Q_k)$.

As computing q_k is equivalent to determining the volume of convex polytopes (i.e., the second-order Voronoi cells) we must turn to estimation procedures for practical implementation. In low to medium dimension, rejection sampling can work; we have had some success in creating tight sampling bounds around the

second-order Voronoi cells via the solution to the extrema linear program of [8]. Alternatively, Markov-Chain Monte Carlo methods [9,10] can be employed. We believe even crude estimates may still be useful for our purposes as the q_k for the Ocean City data discussed in Sect. 6.2 were estimated by considering the relative proportions of their bounding hypercubes and the results there are promising.

6 Clustering Applications

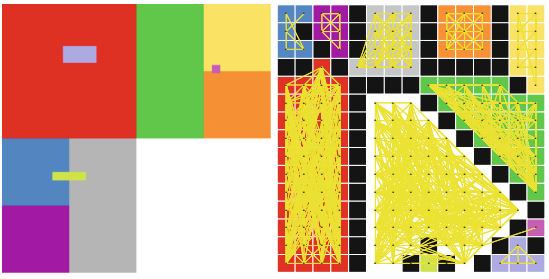
To determine the success of our method we compare the cluster structure recovered from the `DM-Pruned` $CADJ$ graph to known cluster structure in both synthetic and real-world data. Cluster extraction from an SOM has typically been of the highest quality when interactively performed from $CONNvis$. This is time-consuming, requires expertise, and is somewhat subjective. To mitigate these bottlenecks we employ a leading graph segmentation algorithm (Walktrap [11], available in [12]) to segment the pruned $CADJ$, which we have shown produces clusterings of similar quality to those of the human analyst [13]. Results are judged via Unweighted and Weighted Overall Accuracies (UOA and WOA) as compared to reference images. UOA is the average of class-wise %-accuracies whereas WOA is the average pixel-wise accuracy. Utilizing Walktrap as a clustering oracle has its own effects on the quality of the recovered partitioning as it requires further parameterization (which we do not optimize here), but the *relative* accuracies between different pruning levels should still indicate their comparative suitability for cluster discovery.

For each of our data applications we compute the Λ-path and select as candidates for pruning those connections which, when removed, result in 95% and 90% of the "full" model likelihood (denoted Λ-95 and Λ-90, respectively). We also consider the graph resulting from pruning at an additional t-step of interest in each case, as explained below. For reference we compare to a more naive scheme of simply keeping the top locally-ranked connections which contain 100, 95 and 90 percent of the data (denoted $tn-100, 95, 90$, respectively).

6.1 6d Synthetic Spectral Image

Our synthetic data is a 128×128 pixel synthetic spectral image (6d synthetic, 11-class, [14]) of 6 bands containing 11 clusters of various sizes. The spatial layout of the class labels, distinguished by 11 colors, is shown in Fig. 2a. The 15×15 SOM trained on this data is shown in Fig. 2b, where each lattice cell is painted with the color code of the majority pixel labels which were mapped to the cell's neuron. Overlain on the SOM is a visualization called TopoView [4], which depicts the $CONN$ graph without showing the edge weightings, for clarity. The islands which are readily visible indicate that this $CONN$ graph contains clear closed communities, even without pruning. The one exception is the connection from the single neurons representing clusters R (pink) and Y (fluorescent yellow); our goal with this experiment is to ensure that `DM-Pruneing` does not destroy the clearly delineated structure visible in Fig. 2b.

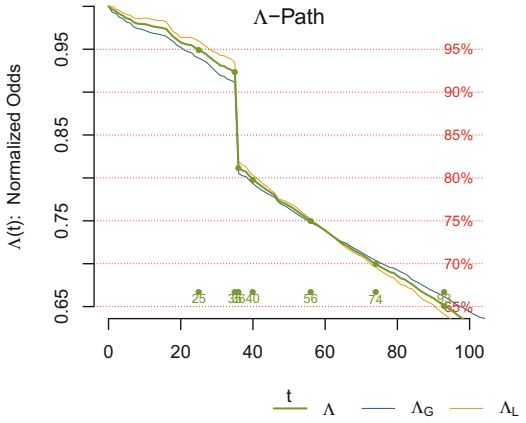
The Λ-path resulting from a Dirichlet-Multinomial fit of these data is shown in Fig. 2d. In it, we see that a steep drop in likelihood occurs between $t = 35$ and $t_{\text{drop}} = 36$. Pruning at $t = 25$ (Λ-95), $t = 35$ (Λ-90), and $t_{\text{drop}} = 36$ produces the accuracies reported in Fig. 2c, which shows perfect cluster capture at both the Λ-95 and Λ-90 levels (since each of their resulting pruned graphs removed the connection between R and Y). Pruning at $t_{\text{drop}} = 36$ also removed connection R-Y, but destroyed the local connectivity holding the large white cluster (B) together (not shown here), resulting in its lower accuracies.



(a) (b)

	% Acc. UOA	% Acc. WOA	% Data	# Clus	% Conn
Λ-95	100.0	100.0	96.0	11	98.3
Λ-90	100.0	100.0	93.8	11	97.7
Λ-t_{drop}	95.6	87.8	86.0	11	95.1
tn-100	90.9	99.9	100.0	10	100.0
tn-95	81.7	75.1	95.2	10	50.6
tn-90	87.6	90.9	90.1	10	35.2

(c)

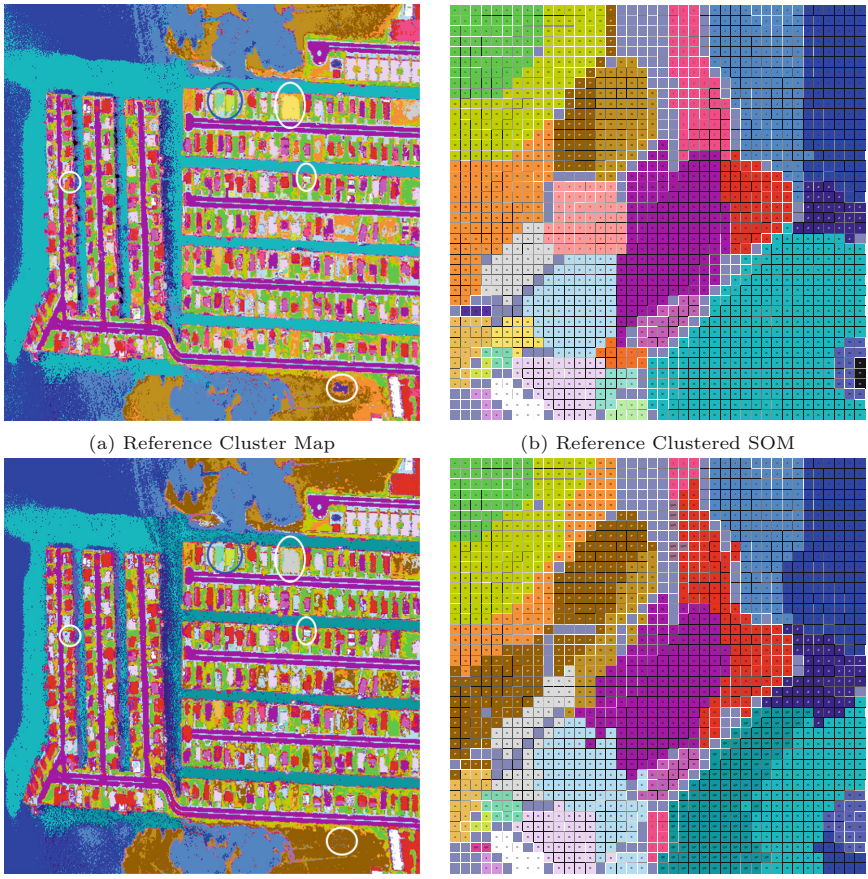


(d)

Fig. 2. (a) The spatial layout of the 11 classes of our 6d synthetic spectral cube. **(b)** The undirected TopoView visualization of $CADJ$ overlain on the SOM lattice which learned the data visualized in (a). Each lattice neuron is connected to the neurons which represent adjacent prototypes in the Induced Delaunay Triangulation of [2]. **(c)** The accuracies achieved after clustering the SOM depicted in (b) via pruning by both `DM-Prune` and a simpler scheme involving local connection rankings. Along with accuracy, the table also reports the % of data and connections which remain represented in the resulting pruned graph, as well as the number of true clusters which were identified by each method. **(d)** The Λ-Path (Sect. 4) depicting degradation of the Dirichlet-Multinomial relative likelihood as the $CADJ$ graph is progressively pruned.

6.2 Real Data: Ocean City Spectral Image

For experiments with real data, we use a 512×512 pixel, 8-band spectral image of Ocean City, Maryland, with 1.5 m/pixel resolution. References to data collection, description of pre-processing and mean signatures of verified land-cover classes are given in [15] along with an earlier interactive clustering as in [1]. This image contains many clusters with widely varying statistical properties



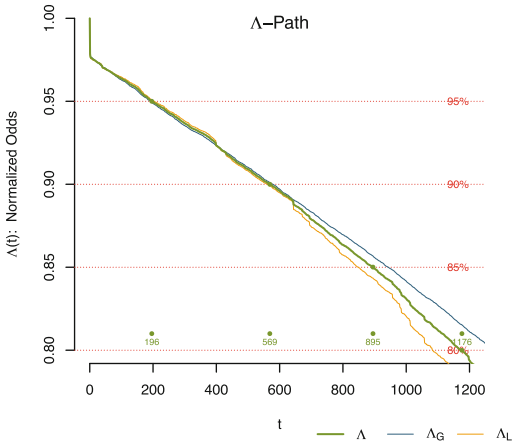
(a) Reference Cluster Map (b) Reference Clustered SOM

(c) $\Lambda(t^*)$ Cluster Map (d) $\Lambda(t^*)$ Clustered SOM

Fig. 3. (a) Interactive clustering of the Ocean City spectral image from CONNvis visualization of SOM prototypes, from [15]. The 28 clusters include ocean, bay, canal, pool water, (medium to dark blue colors); roofing materials (red, white, light pink, hot pink, magenta); grass, shrubs around houses (green, yellow), other vegetation (orange, brown), and several rare clusters: roofs a, m (in blue circle), roofs c, V, shrub g, dry grass M (in white ovals). Asphalt (magenta, G) and reflective paint (neon blue, X) occur on roads as well as on roofs. **(b)** The 40×40 clustered SOM for Ocean City, with the labels and colors of each reference cluster depicted in the lattice cells. **(c)** The clustered image resulting from pruning the Ocean City $CADJ$ at step $t^* = 398$. **(d)** The clustered SOM which generated (c).

(see Table 1 of [4]), and it is very noisy. We compare `DM-Prune` segmentations with the segmentation of this image from [15], shown in Figs. 3a, b. The $CONN$ representation of this data can be seen in [1].

The `DM-Prune` clustering from step $t^* = 398$ (where Λ_L dips below Λ_G in Fig. 4a) in Fig. 3c is very similar to that in Fig. 3a by visual inspection. All major landmarks, water components, all houses, roads, and vegetation are very well delineated. Major differences include the absence of cluster T (salmon color); the absence of small clusters c, g, M, V, X; and the presence of a few extra clusters in Fig. 3c. The locations of c, g, M, V are in white ovals; X was merged to E. These diferences are easiest to find in the SOMs in Figs. 3b and d. Cluster T (a particular roof type) has been divided among clusters G, P, Q, and S, which is reasonable given their neighboring locations in the SOM and their similar cluster signatures in the reference clustering (as seen in Fig. 3 of [15]). In the spatial image cluster T pixels are mostly merged to cluster G, a similar type of roof. Given that in [16] cluster T was detected by Walktrap using the unthresholded $CONN$ graph, the absence of T here could be the result of `DM-Prune`ing, as its null model probabilities q_k were very crudely estimated. The missing small clusters can be ascribed to Walktrap performance since in [16] the same are also absent when Walktrap (with the same default number of steps, 4) is applied to the unthresholded $CONN$ graph. The small clusters a and m (circled in blue) are found, probably due to their more unique signatures. Class-wise and pixel-wise accuracies in Fig. 4b also indicate good match between Fig. 3c and a. While they are relatively low (best case just below 70%) in absolute terms, we need to consider that, in an image with 1.5 m/pixel footprint, many pixels contain multiple materials at object boundaries. These mixed pixels may get assigned to



(a) Λ-Path

	% Acc. UOA	% Acc. WOA	% Data	# Clus	% Conn
Λ-95	57.4	67.6	98.1	22	97.7
Λ-90	58.1	62.7	96.4	23	95.4
Λ-t^*	59.8	67.6	97.3	22	96.5
tn-100	56.1	62.2	100.0	22	100.0
tn-95	58.2	60.7	94.8	22	56.0
tn-90	68.9	69.1	88.6	25	40.5

(b) Clustering Accuracies

Fig. 4. (a) 1200 steps of the Λ-Path of pruning the Ocean City $CADJ$ with `DM-Prune`. **(b)** Accuracies of resulting clustering from `DM-Prune` and simpler thresholding schemes, as compared to Fig. 3a.

different clusters based on different pruning/thresholding of the $CADJ$ graph. The differences can cause fairly large accuracy loss even though they are not distracting to the eye. The accuracy measures ignore the spatial coherence of the errors in areas where mixing is expected. I.e., if the non-matching pixels were scattered throughout the image in a salt-and-pepper fashion, we would consider the match much poorer while the accuracies would remain the same. Overall, we can say the `DM-Prune` clustering is approaching the quality of the reference clustering.

7 Conclusions and Outlook

We have introduced `DM-Prune` as a Bayesian Dirichlet-Multinomial model of the edge weights of the $CADJ$ graph for the purpose of pruning its edges to facilitate cluster discovery from SOM prototypes. The modeling framework provides a natural mechanism for both selecting edges ripe for pruning and assessing the overall impact from doing so. Experiments with both synthetic and real-world spectral image data confirm that graph sparsification governed by `DM-Prune` is capable of retaining vital local connectivities, which if removed would destroy our view of meaningful cluster structure, while simultaneously shedding spurious connections which cloud this view. Future work involves integrating more sophisticated estimation of the null model probabilities q_k and further experiments with the methodology to gain confidence for setting the DM $\boldsymbol{\alpha}$ parameter and, most importantly, to provide more concrete recommendations for identifying optimal pruning levels from the Λ-Path metric.

Acknowledgment. We thank Dr. Beáta Csathó, University of Buffalo, for the Ocean City spectral image and accompanying truth. This project was partially supported by a North American ALMA Development Cycle 5 Study Program, administered by the National Radio Astronomy Observatory, with the consent of the U.S. National Science Foundation.

References

1. Taşdemir K, Merényi E (2009) Exploiting data topology in visualization and clustering of Self-Organizing Maps. IEEE Trans. Neur. Netw. 20(4):549–562. ISSN 1045-9227
2. Martinetz T, Schulten K (1994) Topology representing networks. Neural Netw. 7(3):507–522. ISSN 0893-6080
3. Kohonen T (1997) Self-Organizing Maps, 2nd edn. Springer, Heidelberg
4. Merényi E, Taşdemir K, Zhang L (2009) Similarity-based clustering. chapter Learning Highly Structured Manifolds: Harnessing the Power of SOMs. Springer, Heidelberg, pp 138–168. ISBN 978-3-642-01804-6
5. Okabe A, Boots B, Sugihara K (1992) Spatial tessellations: concepts and applications of voronoi diagrams. John Wiley & Sons Inc., New York
6. Mosimann J (1962) On the compound multinomial distribution, the multivariate β-distribution, and correlations among proportions. Biometrika 49(1/2):65–82. ISSN 00063444

7. DeSieno D (March 1988) Adding a conscience to competitive learning. In: Proceedings of the international conference neural network (ICNN), New York, vol. I, pp I–117–124
8. Agrell E (January 1993) A method for examining vector quantizer structures. In: Proceedings of the IEEE international symposium information theory. IEEE, pp 394–394
9. Dyer M, Frieze A (1991) Computing the volume of convex bodies: a case where randomness probably helps. Probab. Comb. Appl. 44:123–170
10. Lovász L, Vempala S (2006) Simulated annealing in convex bodies and an o(n4) volume algorithm. J. Comput. Syst. Sci. 72(2):392–417
11. Pons P, Latapy M (2005) Computing communities in large networks using random walks. In: Proceedings of the 20th international conference on computer information science, ISCIS 2005, Springer, Heidelberg, pp 284–293
12. Nepusz T, Csardi G (2006) The igraph software package for complex network research. Int J Complex Syst 1695(5):1–9
13. Merényi E, Taylor J (April 2018) Empowering graph segmentation methods with SOMs and CONN similarity for clustering large and complex data. Neural Comput. Appl. (forthcoming)
14. Jain A (2004) Issues Related to Data Mining with Self-Organizing Maps. Rice University, 2004. M.Sc. thesis
15. Merényi E, Csató B, Taşdemir K (2007) Knowledge discovery in urban environments from fused multi-dimensional imagery. In: Gamba P, Crawford M (eds) Proceedings of the IEEE GRSS/ISPRS joint workshop on remote sensing and data fusion over urban areas (URBAN 2007), Paris, France, 11–13 April 2007, pp 1–13
16. Merényi E, Taylor J (June 2017) SOM-empowered graph segmentation for fast automatic clustering of large and complex data. In: 12th international workshop on self-organizing maps and learning vector quantization, clustering and data visualization (WSOM+2017), pp 1–9