Neural Computing and Applications (2022) 34:25–38
https://doi.org/10.1007/s00521-021-05831-z

S.I.: WSOM 2019

DM-pruning CADJ graphs for SOM clustering

Josh Taylor[1] · Erzsébet Merényi[2]

Received: 22 May 2020 / Accepted: 17 February 2021 / Published online: 19 May 2021

Abstract

As topology representing networks, the Cumulative ADJacency graph CADJ and its symmetric version CONN $=$ CADJ $+$ CADJ^T Tasdemir and Merenyi (IEEE Trans Neural Netw 20(4): 549–562, 2009), can be utilized as inputs to graph-based clustering (GBC) paradigms for partitioning the learned prototypes of a vector quantizer. To express complex data faithfully, CADJ must typically be pruned (thresholded, or made sparse) to be most effective as an input to GBC routines, whether they be algorithmic or driven by human assessment. This work, given in two parts, develops a formal framework for CADJ pruning as a preprocessing (sparsifying) step to improve CADJ's use in *any* GBC routine. That is, rather than advocating a particular GBC method, our goal is development of sensible logic for creating sparse CADJ inputs to the entire family of GBC methods. Part 1 defines an overall quality measure for each CADJ edge by extending lines of reasoning used successfully in the past to prune CONN graphs. Part 2 introduces a Bayesian Dirichlet-multinomial (DM) model of CADJ edge weights with an intelligent prior constructed through analysis of the Voronoi tessellation generated by the vector quantization. The DM likelihood offers an internal assessment of information loss resulting from iterative CADJ edge removal, which is used to determine an optimal stopping criterion for the pruning process. We show that `DM-Pruned` CADJ graphs lead to GBCs comparable to the best previously achieved on highly structured real data.

Keywords SOM clustering · Topology representing graph · Graph sparsity

1 Introduction

CADJ was introduced in [22] as a weighted version of the Induced Delaunay Triangulation of Martinetz and Schulten [11] representing topological adjacencies of the M prototypes $\{\boldsymbol{w}_j\}_{j=1}^{M}$ of a vector quantizer. Given data $X = \{\boldsymbol{x}_i\}_{i=1}^{N}$ drawn from manifold $\mathcal{M} \subseteq \mathbb{R}^d$, the CADJ weight of the edge connecting prototypes $\boldsymbol{w}_j$ and $\boldsymbol{w}_k$ is $\text{CADJ}_{jk} = \#\{\boldsymbol{x} \in X : BMU1(\boldsymbol{x}) = j, BMU2(\boldsymbol{x}) = k\}$, where $BMU1(\cdot)$ and $BMU2(\cdot)$ return the index of the Best and second-Best Matching Unit (or prototype, respectively) of the datum $\boldsymbol{x}$, and $\#$ denotes set cardinality. Positive CADJ_{jk} values reflect the strength of topological connectivity of prototypes $\boldsymbol{w}_j$ and $\boldsymbol{w}_k$ (and, consequently, connectivity of the portion of the manifold $\mathcal{M}$ they represent), whereas values of 0 indicate disconnected portions of the manifold.

While the CADJ graph may originate from the learned prototypes of any vector quantizer (e.g., K-means or Neural Gas), in this work we focus solely on prototypes arising from Self-Organizing Map (SOM) learning [8], as its lattice both informs `DM-Prune` modeling and provides a way to visualize CADJ, which facilitates our discussion. Indeed, this CONNvis visualization ([22], which represents CADJ's symmetrized version CONN $=$ CADJ $+$ CADJ^T on the lattice), has been used as a successful tool for human-assisted GBC, via identification of strongly connected CONN prototype communities. However, the CONN graph representing complex, high-dimensional datasets typically does not contain readily identifiable communities due to a large number of weak connections induced by noise and deemed unimportant for the expression of manifold structure. [22] proposes a method for

✉ Josh Taylor
jtay@rice.edu

Erzsébet Merényi
erzsebet@rice.edu

[1] Department of Statistics, Rice University, Houston, TX 77005, USA

[2] Departments of Statistics and Electrical and Computer Engineering, Rice University, Houston, TX 77005, USA

pruning CONN edges by assessing their *global* and *local* importance relative to a thresholding grid generated from statistical summaries of the CONN edge weights. Selection of the "best" combination of thresholds has historically been performed ad hoc via human assessment of the clusterings resulting from each pruned graph. This process is tedious, as 1) there is no natural ordering to the thresholding grid (e.g., it is unknown in what circumstances an edge's global importance should outweigh its local importance), 2) it creates unnecessary work as many of the grid threshold combinations produce very similar pruned graphs and, most importantly, 3) it lacks guidance as to what thresholding levels might be most useful for the observed data. The `DM-Prune` strategy, introduced in [23] and extended here, develops a priori suggestions for thresholding CADJ for optimal cluster extraction. A scoring metric μ, which combines the spirit of grid thresholding discussed above with an additional probabilistic assessment, produces a quality metric for each edge which is used to rank connections for iterative removal over "time," stopping when a likelihood-based metric of pruning impact begins to deteriorate. Our goal is toward intelligent automation of CADJ pruning to improve its use in any GBC method.

While CADJ is technically an $M \times M$ matrix reporting the strength (or absence) of topological connections between prototypes, it is typically very sparse. For the rest of the discussion, we will only consider the subset of connections $JK = \{(j,k) : \text{CADJ}_{jk} > 0\}$ which index the nonzero entries in CADJ and let $|JK|$ denote the number of such connections. Obviously, $\sum_{jk} \text{CADJ}_{jk} = |X| = N$. To hopefully avoid later confusion, we pause to clarify some terminology we will consider interchangeable in what follows. Because CADJ is a weighted adjacency matrix, it defines a graph; we thus make no distinction between a "CADJ connection" and a "graph edge," as they refer to the same thing. A connection's "strength" (graph edge weight) refers to its corresponding CADJ_{jk} value. Additionally, this graph can simultaneously be considered to connect the SOM prototypes in data space ($\mathbb{R}^d$) or the SOM neurons on the lattice.

As a final introductory note, we reiterate that `DM-Prune` is not a clustering method on its own; rather, it is intended to be a sparsification step for CADJ inputs to GBC methods. We have previously shown [12] this combination can produce sophisticated clusterings of complicated real data *if the right CADJ graph is selected as input*. In this work, we are not advocating any particular clustering method to be paired with CADJ, but we *are* advocating the use of CADJ graphs as GBC inputs vs., e.g., spectral clustering methods, which also rely on topology representing networks. This conclusion is based on the increased sensitivity of all clusterings developed in this work vs. the spectral clustering of Fig. 1. Effectiveness of the use of the symmetric version of CADJ (CONN) vs. K-means has been demonstrated previously [20].

1.1 Example data: multispectral image from mars

The initial introduction of `DM-Prune` [23] exercised its nascent methodology on both synthetic and real hyperspectral images; for its expansion in this work, we will only consider a complex, real multispectral image (which we introduce immediately) for space considerations.

On January 4, 2004, NASA's Mars Exploration Rover Spirit landed on Mars to search for evidence of past water reserves in Gusev Crater. This search centered in part on compositional studies of the soil and rocks in the crater via remote sensing with a panoramic camera ("Pancam") capable of imaging in the 400–1100 nanometer wavelength range (near-UV to near-IR). We will experiment with a 700 x 450 pixel, seven-band multispectral image of Husband Hill, taken by Spirit in the Columbia Hills region of Gusev Crater on sol 608 of its mission, which we refer to as "MER Image" in what follows. Band one of this image, centered at 423 nm, is shown in Fig. 1.

The results of a previous SOM-based clustering of this image [14, 21] obtained interactively via CONNvis analysis (described below) are shown in the top row of Fig. 1. This study segmented 22 scientifically verified soil and rock compositions from the MER image, as listed in the provided table. The 40 x 40 clustered SOM lattice is shown here as well; white, unlabeled grid cells represent neurons which were previously unclustered by the human analyst. We will also ignore these in our analysis, as any new findings involving their inclusion would require additional scientific verification. The bottom row of Fig. 1 shows a Spectral Clustering (SC [10] using the automatic parameter tuning of [24], made available in the R package `Spectrum`), of the same learned SOM prototypes. While also derived from a topology representing graph, the SC exhibits a loss of scientific granularity; generally, it differentiates only rock from soil. This coarse clustering highlights the benefits of harnessing CADJ in GBC methods and motivates its further scrutiny in this study.

1.2 Grid-based CADJ thresholding

Along with the CADJ and CONN graphs, [22] also introduces the CONNvis visualization as a way to express manifold connectivity on the SOM lattice to support human-enabled GBC with SOMs. An example CONNvis for the MER CONN graph is found in Fig. 2. While CADJ/CONN are born sparse (with number of edges

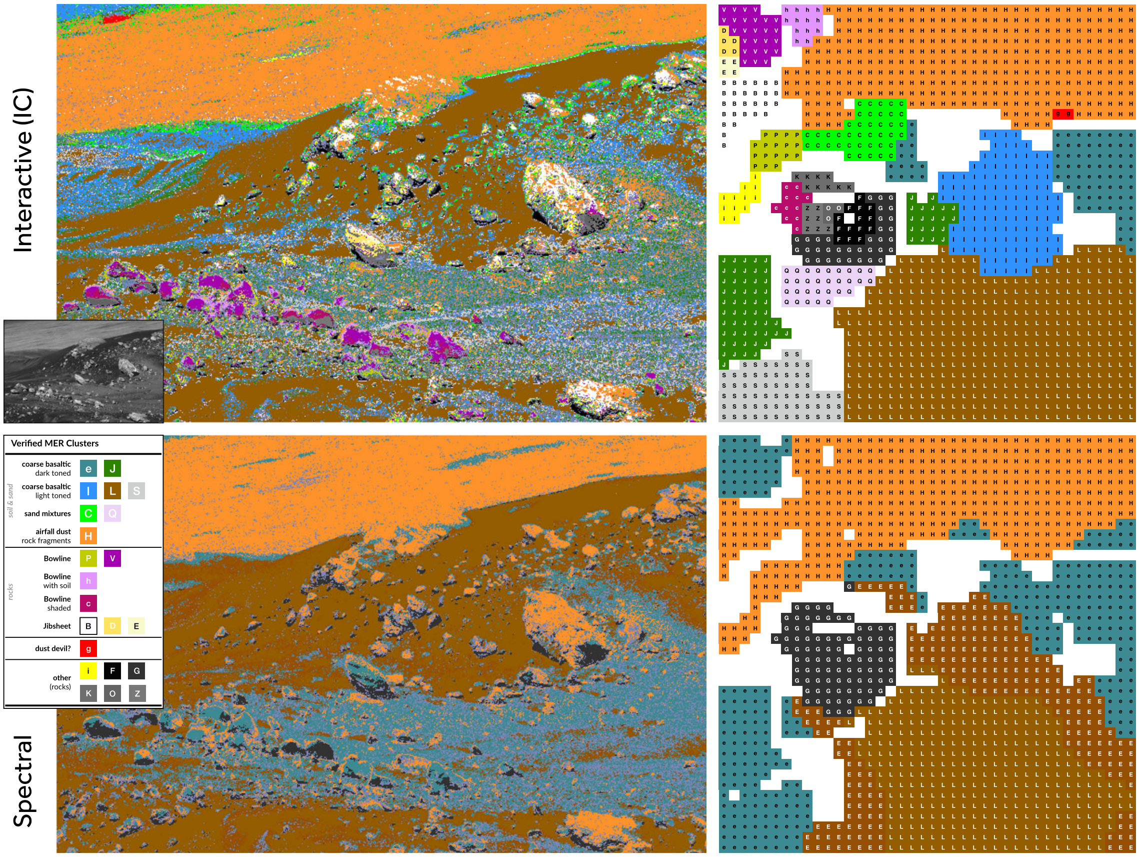


Fig. 1 Top Row: The Interactive Clustering (IC) of the MER image of Husband Hill from [14, 21], showing spatial cohesiveness of the 22 clusters verified by MER scientist WH Farrand. The 40 × 40 SOM clustering (top right) was obtained via human assessment of the CONNvis visualization [22]. **Bottom Row:** A Spectral Clustering of the learned SOM prototypes of the MER image, showing a marked loss of material discrimination compared to the IC. **Inset:** Band 1 (at 423 nm) of Spirit's image of Husband Hill, and the scientific interpretation of the 22 IC clusters comprising various compositions of rocks and soils visible in the clustered images

$<<\mathcal{O}(M^2 = 1600^2 = 2,540,000))$, the CADJ graph underlying the CONNvis displayed in Fig. 2 still contains 26,542 edges which, collectively, obscure any immediate cluster inference. To combat this, [22] proposes three criteria useful for CONN edge removal. We review these criteria below, applied directly to the CADJ graph (instead of CONN).

Global importance assessments, which are simple comparisons of CADJ_{jk} to all other CADJ_{rs}, reflect the intuitive idea that prototypes joined together by more data vectors are more likely to belong to the same cluster than not. We call thresholding by global importance *value* thresholding, with threshold limits denoted by tv (so that, e.g., thresholding at $tv = 30$ results in removing all $\mathrm{CADJ}_{jk} < 30$). The tv thresholds are set as the mean connection strengths when grouped by their local rank, which is the descending rank order of each connection strength CADJ_{jk} relative to the immediate CADJ graph-neighbors of node j (explained in detail in [22]). Because global assessments alone can overlook finer cluster structure that occurs in lower-density portions of the manifold, we must also incorporate local rank threshold limits, denoted by tn. Thresholding at, e.g., $tn = 2$, removes all connection whose local rank is > 2; the tn limits range in the integers from 1 to the maximum number of CADJ neighbors of any prototype j. Because the SOM is a topology-preserving mapping from $\mathbb{R}^d$ to its output lattice, [22] also suggests thresholding connections based on the lattice distance of the neurons they connect; due to the organization that occurs on the lattice, distant neurons are less likely to comprise the same cluster (and if they do, should be connected to intermediary lattice neurons). Letting $d^{\infty}(j,k)$ represent the Chebyshev distance between neurons j and k on the lattice, threshold limit $tl = 3$, e.g., removes all connections CADJ_{jk} such that $d^{\infty}(j,k) > 3$. The Cartesian product of the unique tv, tn and tl values constitutes the thresholding grid, which can grow quite large for complicated data.

The connection statistics underlying the CONNvis of the MER image are displayed at the right of Fig. 2. The maximum number of neighbors of any prototype for this data is 13, so there are $13 - 1 = 12$ possible tn

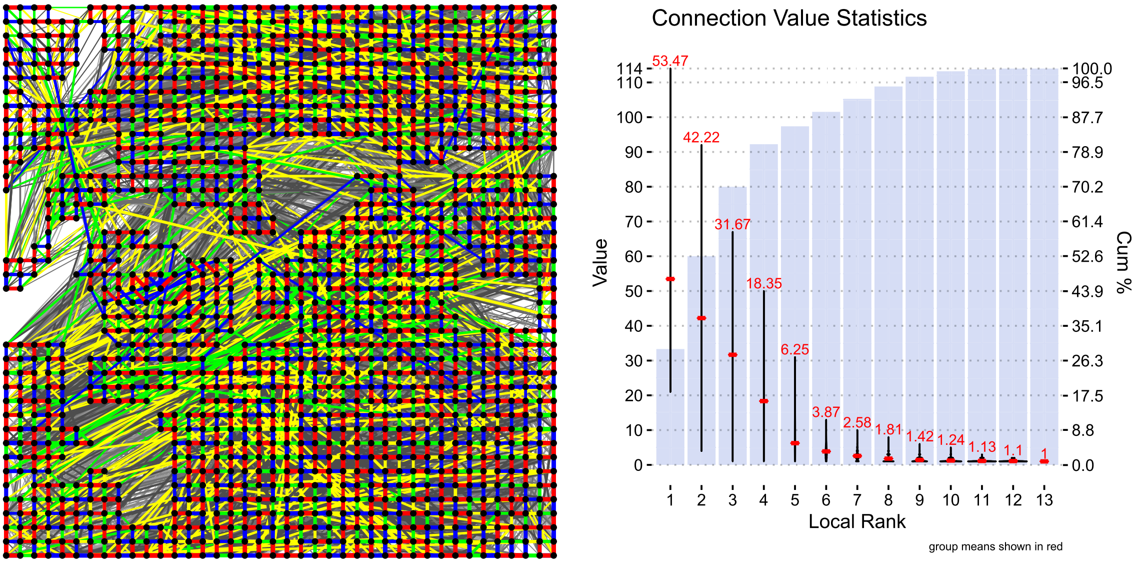


Fig. 2 Left: CONNvis (no thresholding) of the MER image data on the SOM lattice. Line widths correspond to global edge strength (in order, highest to lowest), while line colors correspond to the rank of local edge strengths (red, blue, green and yellow = highest through 4^{th} highest locally ranked connection, respectively, with the rest in grayscale from darkest to lightest. Right: The statistics of CADJ values computed by local rank, as suggested in [22]. Violin plots of the distribution of edge strengths within each rank are plotted, with the mean value per rank indicated in red. The blue bars show the cumulative percentage of data forming connections between prototypes in each rank

thresholding limits. Studying the statistics per rank, we see there are only seven unique *tv* thresholds (as CADJ values are integers, all of the mean connection strengths of ranks 9-13 are redundant). The lattice lengths of MER connections occupy the entire possible range for a 40 x 40 SOM ($\{1, \ldots, 39\}$), so that the number of *tl* thresholds is $39 - 1 = 38$. Thus, the MER image thresholding grid is of total cardinality $12 \times 7 \times 38 = 3,192$. In practice, many of these (*tv*, *tn*, *tl*) combinations are redundant, and our experience with CONNvis circumvents the need to construct and analyze the full grid but, even with a small number of candidate thresholding combinations, producing and analyzing a clustering for each is tedious.

2 Scoring and ranking edge quality

To move toward a more unified framework for edge pruning, we propose scoring each connection CADJ_{jk} with a measure

$$\mu_{jk} = \left[\mathtt{G}_{jk}^{\gamma_G} \times \mathtt{N}_{jk}^{\gamma_N} \times \mathtt{L}_{jk}^{\gamma_L} \times \mathtt{S}_{jk}^{\gamma_S}\right]^{1/(\gamma_\mathtt{G}+\gamma_\mathtt{N}+\gamma_\mathtt{L}+\gamma_\mathtt{S})} \tag{1}$$

where $*_{jk}$ are individual components contributing to the overall quality score, defined in Table 1 and discussed below. The γ_* are weightings controlling the contribution of each component to the overall score. We have not experimented with changing these weightings (have set them all $= 1$ in the following) but include them in the definition of μ_{jk} for potential future optimization. The definitions of each component in Table 1 reflect our adopted convention that each $*_{jk}$ is in the range [0, 1] with 1 representing the highest quality according to the individual measure.

Because of the great success resulting from the grid thresholding scheme in Sect. 1.2, we define the first three components to simulate its effects: $\mathtt{G}_{jk}$ (inspired by *tv* thresholding) measures the global strength of each connection, $\mathtt{N}_{jk}$ (inspired by *tn* thresholding) measures the local strength of each connection relative to all neighbors of prototype w_j and $\mathtt{L}_{jk}$ (inspired by *tl* thresholding) measures the plausibility of each connection based on the lattice distance of neurons j and k. We describe this as a measure of plausibility due primarily to the local lattice influence of the prototype update step of the SOM learning rule. While connections of long lattice length do occur, the prototypes joined by a long lattice connection have not been directly influenced by each other during learning; consequently, we view them as less reliable than connections spanning small distances on the lattice.

In addition to the above, we have added to our quality score μ_{jk} the effects of component $\mathtt{S}_{jk}$, which attempts to correct for sampling noise contributing to our observed counts CADJ_{jk}. Using a kernel density estimator $\hat{f}(y)$ fit to data X, we draw B (new) samples $Y^1, \ldots, Y^B$ (of the same size, N, as X) and construct a CADJ matrix CADJ^b for each. $\mathtt{S}_{jk}$ is the proportion of times the edge CADJ_{jk}

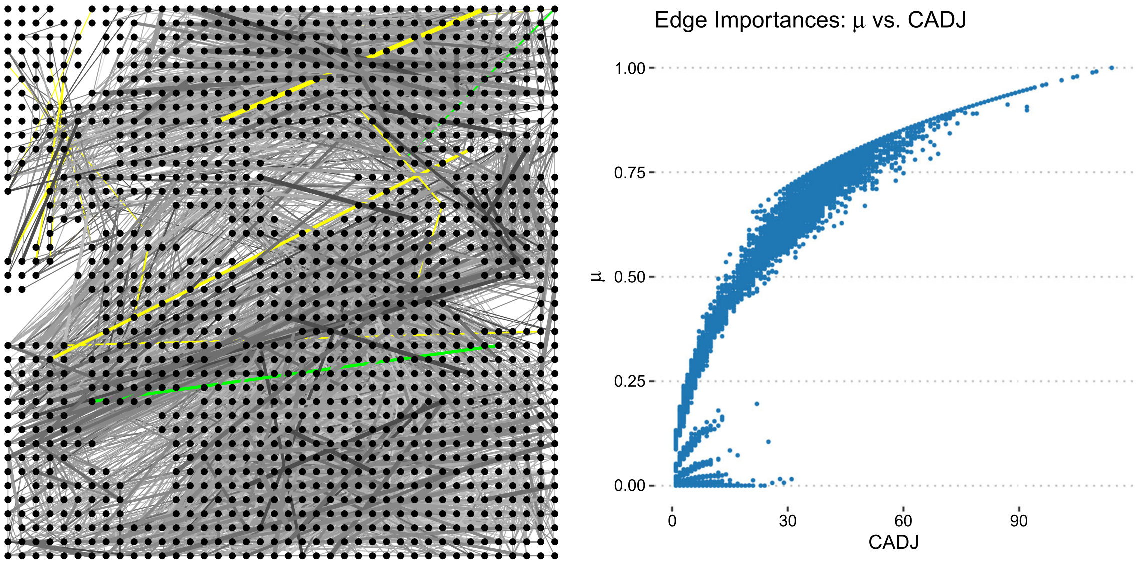
Table 1 Components comprising the quality score μ_{jk} of each connection CADJ_{jk}

Global importance $\mathtt{G}_{jk} = \dfrac{\mathrm{CADJ}_{jk}}{\max\limits_{r,s} \mathrm{CADJ}_{rs}}$ Measures the strength of each CADJ connection, relative to all others	*Neighborhood (local) importance* $\mathtt{N}_{jk} = \dfrac{\mathrm{CADJ}_{jk}}{\max\limits_{s} \mathrm{CADJ}_{js}}$ Measures the strength of each CADJ connection, relative to all neighbors of its source neuron
Length plausibility $\mathtt{L}_{jk} = \dfrac{D^* - d^{\infty}_{SOM}(j,k) + 1}{D^*}$ where $d^{\infty}_{SOM}(j,k)$ is the Chebyshev distance between neurons j and k on the SOM lattice and $D^* = \max_{r,s} d^{\infty}_{SOM}(r,s)$. Measures the plausibility of each CADJ connection based on its length on the lattice	*Sampling plausibility* $\mathtt{S}_{jk} = \dfrac{1}{B}\sum\limits_{b=1}^{B} \mathbb{1}\left[\mathrm{CADJ}^{b}_{jk} > 0\right]$ where B is the number of bootstrap samples and CADJ^{b} is the CADJ matrix constructed from the b^{th} sample. Measures the plausibility of each connection, as a proportion of times it appears during bootstrap resampling
Overall score $\mu_{jk} = \left[\mathtt{G}^{\gamma_G}_{jk} \times \mathtt{N}^{\gamma_N}_{jk} \times \mathtt{L}^{\gamma_L}_{jk} \times \mathtt{S}^{\gamma_S}_{jk}\right]^{1/(\gamma_{\mathtt{G}}+\gamma_{\mathtt{N}}+\gamma_{\mathtt{L}}+\gamma_{\mathtt{S}})}$ where each $\gamma_* > 0$ is a component weight. For this study all $\gamma_* = 1$.	

appears in the set of bootstrapped CADJ matrices. Connections formed spuriously through sampling variability of lower density portions of the manifold can be easily identified by low $\mathtt{S}_{jk}$ values, as depicted in the example in Fig. 3a, where connections of the MER image CONNvis with $\mathtt{S}_{jk} < 0.5$ are shown. The connections deemed least plausible due to sampling variability are generally of longer length and weaker strength (indicated by their thin lines and gray color). A few thicker green and yellow connections exist, indicating some edges of stronger global or local strength are not reliably persistent over repeated sampling. Overall, 6,695 of the 26,542 edges in this CADJ matrix have an S-score < 0.5.

As defined, μ_{jk} attempts to incorporate the assessments a human analyst would make about the importance of connection CADJ_{jk} to cluster discovery via CONNvis, in addition to incorporating new information from $\mathtt{S}_{jk}$ not previously considered. Ranking by (increasing) μ provides a natural ordering of the set of connections by their combined importance. This ordering deviates from that implied by the CADJ value alone, as is visible in Fig. 3b.

For the rest of this discussion, we will view the act of pruning as a process occurring over "time" steps t, whereby at each t we remove edges from the CADJ graph in increasing rank of μ_{jk}. The total time horizon T is equal



(a) **(b)**

Fig. 3 **a** An example of the MER image SOM lattice shown with connections whose $\mathtt{S}_{jk}$ values are < 0.5. **b** A view of the μ score vs. the respective CADJ value for each connection, showing how μ_{jk} can consider CADJ connections of the same strength very differently to max rank(μ_{jk}); because the μ_{jk} values are not necessarily distinct (edges can share the same score), T can be less than the number of nonzero CADJ edges. At a given time step t, we produce a pruned CADJ graph by zeroing out edges in the original CADJ matrix, which we capture with the following notation for CADJ as a function of time:

$$\mathrm{CADJ}_{jk}(t) = \left\{ \begin{matrix} 0 \\ \mathrm{CADJ}_{jk} \end{matrix} \right\}, \quad N(t) = \sum_{jk \in JK} \mathrm{CADJ}_{jk}(t) \tag{2}$$

As removing edges is equivalent to removing data, the effective sample size N also changes with t, which is captured in the notation for $N(t)$ above.

3 A probability model for CADJ

3.1 From graph to model

By definition, CADJ$_{jk}$ are random counts of N observations falling into "bins," where the bins are the second-order Voronoi cells V_{jk} [18] of the tessellation induced by prototypes $\{w_j\}$; recall that we have $|JK|$ such nonempty bins (nonzero edge weights). The standard probability model for the counts of N objects in $|JK|$ bins is the multinomial. For our data X, the true (unknown) probability of bin V_{jk} is $p_{jk} = \int_{V_{jk}} f(x)$, where $f(x)$ is the probability density of manifold $\mathcal{M}$ from which X was drawn. Because we have an intuitive prior, we will jointly model the CADJ counts and unknown probabilities p in a Bayesian setting using the compound Dirichlet-multinomial (DM) distribution [17]

$$\begin{aligned} f_{DM}(\mathrm{CADJ}|\alpha, N) &= \int f_M(\mathrm{CADJ}|p, N) f_D(p|\alpha)\,dp \\ &= \frac{\Gamma(\alpha_0)\Gamma(N+1)}{\Gamma(N+\alpha_0)} \prod_{jk \in JK} \frac{\Gamma(\mathrm{CADJ}_{jk} + \alpha_{jk})}{\Gamma(\alpha_{jk})\Gamma(\mathrm{CADJ}_{jk}+1)} \end{aligned} \tag{3}$$

where f_M is the multinomial pmf and f_D is the Dirichlet pdf. In words, this model asserts that the observed CADJ values were generated by a two-stage compound probability model: bin probabilities p are drawn from a Dirichlet distribution with parameter α (a vector of pseudo-counts), and then CADJ counts are drawn from a multinomial distribution with parameters p and N. Under DM conjugacy, the posterior distribution of unknown p, after observing multinomial count data, with prior distribution $Dir(\alpha)$, is also Dirichlet with modified posterior parameter $\bar{\alpha} = \{\alpha_{jk} + \mathrm{CADJ}_{jk}\}$.

3.2 Selecting the Dirichlet prior

Absent a particular reason to bias the resulting estimation, standard practice in Bayesian statistics is to select an uninformative prior α. In this case, $\alpha_{jk} = 1, \forall jk$ dictates all probability vectors p to be equally likely. However, because our bins V_{jk} are polytopes in $\mathbb{R}^d$, we are not completely uninformed: each possesses a distinct geometry, and hence a distinct volume, which we incorporate into the prior

$$\mathcal{U}_{jk} = N \times \frac{\int_{V_{jk}} 1\,dx}{\sum_{rs \in JK} \int_{V_{rs}} 1\,dx} = N \times \frac{\mathrm{volume}(V_{jk})}{\sum_{rs \in JK} \mathrm{volume}(V_{rs})}; \quad \mathcal{U} = \{\mathcal{U}_{jk}\} \tag{4}$$

Using $\mathcal{U}$ as prior information in (3) specifies a type of geometric "null model" for CADJ counts (capturing our expectation that larger Voronoi cells likely contain more observations, and vice-versa). That is, we compare observed CADJ counts to those which might arise if uniform noise were recalled through the SOM trained on X, which has the effect of normalizing each bin count relative to its volume. As each V_{jk} is highly irregular, computing its volume in $\mathbb{R}^d$ for general d is not trivial; we have utilized the volume of the Maximum Volume Inscribed Ellipsoid (MVIE, [25]) of each polytope V_{jk} as an approximator.

4 The Λ-path

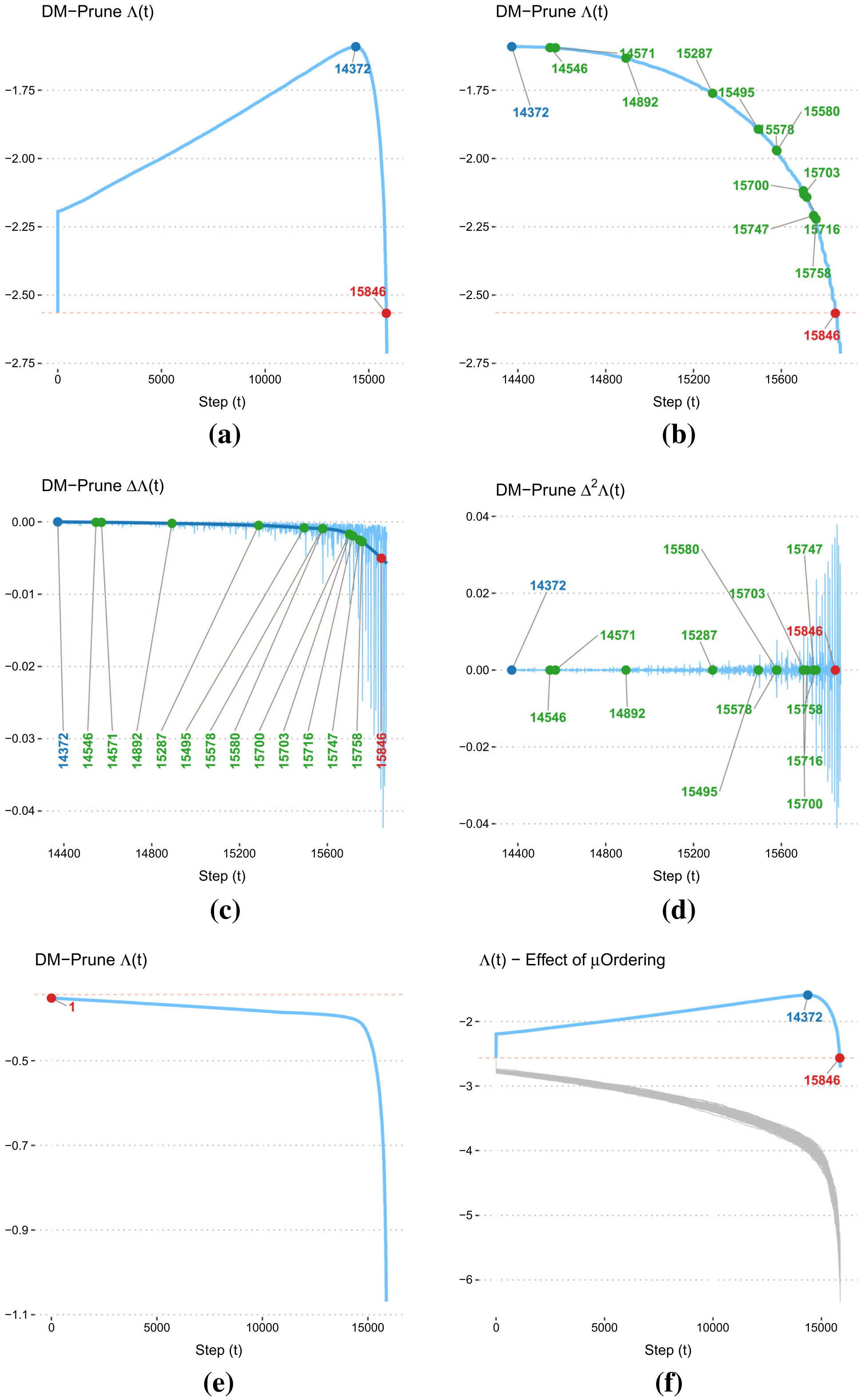
The DM likelihood afforded by (3) for each of our pruned CADJ graphs CADJ(t) offers a natural tool for monitoring the impacts of pruning over time, which we incorporate into the time-dependent metric $\Lambda(t)$:

$$\Lambda(t) = \frac{1}{N(t)} \log[f_{DM}(\mathrm{CADJ}(t)|\mathcal{U}, N(t))] \tag{5}$$

While derived from a probability mass function, $\Lambda(t)$ is not a log-likelihood in the typical sense because of division by $N(t)$, which is a decreasing function of t due to the sparsity invoked by removing CADJ edges over time. Its presence normalizes the marginal likelihood of the data at each pruning step, since f_{DM} is a decreasing function of N (equivalently, an increasing function of $N(t)$). For each t, $\Lambda(t)$ can be thought of as the log-probability of a typical categorical trial (with $|JK|$ categories) in a sequence of $N(t)$ trials. Note that the prior $\mathcal{U}$ is constant over t. We refer to a plot of $\Lambda(t)$ as a function of t as the Λ-Path. An example for this MER image is given in Fig. 4a.

The genesis of the Λ-Path is based on intuition that there exists a relationship between data topology and data likelihood (resulting from an appropriate probability model). That is, we expect connections with lower importance to

Fig. 4 **a** DM-Prune Λ-Path, **b** in the pruning range beyond $t_{\max}$, along with its **c** first- and **d** second-order differences. **e** Depicts a Λ-path resulting from using the prior $\alpha = 1$ in the DM model instead of (4), while **f** shows 100 Additional Λ-Paths (in gray) resulting from pruning CADJ edges in random order. The green markers along each curve indicate variance change points in the time series of $\Delta^2\Lambda(t)$ and are used as suggestions for halting the pruning process



(a)

(b)

(c)

(d)

(e)

(f)

our topological inference from the SOM to be considered more or less expendable by Λ. If this intuition is true, the Λ-Path should exhibit generally concave behavior over time, with a (hopefully visible, and informative) maximum at some time $t_{\max}$. We propose that for $t \in [0, t_{\max})$ the curve represents a denoising (or "cleaning") phase of the pruning process, whereby successive removal of spurious CADJ edges improves Λ. Figure 4a confirms the existence of such a $t_{\max} = 14,372$ for the MER image. Recall that edges do not necessarily have distinct μ scores; the large upswing in the curve at $t = 1$ results from initial, collective removal of all lowest-strength edges ($\mu \approx 0$). Overall, Λ increases in the range $[0, t_{\max})$, but not monotonically (which is not obvious at the scale at which Fig. 4a is shown).

From $t_{\max}$ onward, Λ exhibits decreasing behavior, which we would expect if Λ is a faithful barometer of edge importance. We call this second phase the actual pruning phase of the process, during which Λ considers additional edge removal deleteriously. The red marker in Fig. 4a at step 15,846 denotes the point $t_{\text{end}} = t : \Lambda(t) = \Lambda(0) \wedge t > t_{\max}$. At t_{end}, we have lost all Λ gains achieved during pruning and suggest this point as an upper bound (not tight) for the `DM-Pruning` process.

One might wonder whether Λ is guaranteed to exhibit the behavior visible in Fig. 4a in all situations, regardless of the order in which edges are removed. The answer is no, an example of which is shown in Fig. 4f. In it, we have visualized the Λ-Path of 4a (which results from pruning over time according to μ-ordering) alongside 100 different Λ-Paths (gray lines) resulting from pruning edges in a random order. The difference in behavior is stark; all random pruning orders produce (generally) monotonically decreasing Λ-Paths. This leads to the un-insightful conclusion that removing edges randomly from the graph is not recommended. What we do gain from Fig. 4f, however, is evidence that Λ *does* appear to encapsulate our somewhat loose starting intuition that some edges in the CADJ graph are expendable while others are not, assuming we specify the pruning order accordingly.

4.1 Halting the pruning process

We will consult the Λ-Path to help decide on a point in time t^* which produces a recommended pruning strategy. To be clear, it will be used as a guide for sensible pruning, instead of an oracle. At first glance, $t_{\max}$ appears an obvious candidate for t^* but we suggest it as a lower bound in the pruning process. Recall from above that as the cleaning stage ($t \in [0, t_{\max}]$) progresses, we gain confidence that the remaining CADJ edges are topologically important, meaning they are nonredundant, and relevant for topological inference from the graph. If we merely wanted the "best" view of the graph in this sense, we would stop pruning at $t_{\max}$. Recall, however, that our goal for this procedure is clustering the CADJ vertices (i.e., the SOM prototypes), which is strongly facilitated by starting with a graph with closed (or nearly closed) graph communities. With this goal in mind, it is likely necessary to remove some CADJ edges beyond the cleaning stage, particularly for the discovery of small or rare clusters. Put more strongly, we advocate for intentionally damaging (vis-à-vis Λ) parts of the denoised CADJ graph to help elicit its most intricate community structure. The artful part of this is, of course, determining how much damage to inflict to reveal such structure without completely degrading the cohesiveness of the communities themselves.

To help guide us to a suitable t^*, we will additionally analyze the first and second derivatives of $\Lambda(t)$ *in the pruning phase*. Considering the time series nature of the Λ-Path this amounts to considering its first- and second-order differences $\Delta\Lambda(t)$ and $\Delta^2\Lambda(t)$, respectively. By definition, $\Delta\Lambda(t_{\max})$ should be close to zero. As time moves onward from here, $\Lambda(t)$ should exhibit increasing orders of degradation as more important edges are removed from the graph. Using this intuition, we suggest the point t^* be set where this degradation begins to accelerate appreciably, which we formalize below.

Figure 4b provides a zoomed-in view of the Λ-Path for the MER image in the pruning phase. A human would likely identify some time around $t = 14,892$ as a natural change point along the curve, which is somewhat justified when viewing $\Delta\Lambda(t)$ (light blue) and its trend (dark blue) in Fig. 4c. Consulting the plot of $\Delta^2\Lambda(t)$ in Fig. 4d confirms accelerated change of $\Lambda(t)$ at this point as well. At small time resolutions, $\Lambda(t)$ is not a smooth curve, and the (forward)-differencing filters utilized in Fig. 4c, d amplify this local noise. Recalling that our goal is to strike a balance between under- and over- pruning, we recommend setting t^* at the point where this noise appears to increase, which is most amplified by studying $\Delta^2\Lambda(t)$. Due to the noise amplification that occurs in the second-order difference, this strategy would be considered conservative.

To help automatically identify such points, we turn to more formalized changepoint analysis, which in general outlines a framework for determining when the statistical properties of an ordered sequence of random variables changes. Early work [5, 6] focuses on formal hypothesis tests of distributional changes in the sequence (particularly changes in the mean) based on asymptotic likelihood-ratio test statistics. From the discussion above, we are most concerned with changes in variance of the sequence constructed by $\Delta^2\Lambda(t)$ and will thus rely on the variance changepoint detection outlined in [1].

The second-order variance change points for the MER image, calculated via the `changepoint` package [7] in R, are visible as green markers in the panels of Fig. 4. Analyzing these points in the context of all three paths, we see the first two (at $t \approx 14,500$) do not contain much visible change in either $\Delta^2\Lambda(t_{\max})$ or $\Lambda(t_{\max})$ itself. The third identified point at $t = 14,892$ does appear to signal a perceptual change in both of these graphs, so we have set t^* at this point and show the CADJvis of the resulting

CADJ(t^*) in Fig. 5 along with that of CADJ(t_{end}) for comparison (CADJvis is the directed version of CONNvis, producing half-lines to highlight the asymmetric strengths among prototype connections). The CADJ(t^*) graph appears much more amenable to cluster extraction than its unpruned peer in Fig. 2, while CADJ(t_{end}) has been pruned to the point that no cohesive prototype communities remain. Earlier we suggested t_{end} as an upper bound to the pruning process, and we see evidence of that here.

With a pruning step t^* specified and a pruned CADJ obtained, we are now ready to extract clusters from the SOM of the MER image. The following section details CADJ(t^*)'s role in automated cluster discovery.

4.2 Automated cluster extraction from a DM-Pruned graph

One could cluster a pruned CADJ(t) interactively (i.e., through visual inspection), which is how cluster extraction from CONNvis has traditionally been performed. As we believe this step to be the largest bottleneck to wider adoption of SOM-based clustering, we have recently employed modern graph segmentation (also called community detection) algorithms to do this heavy lifting, with promising results [12, 13, 15, 16]. In this work, we will utilize clusterings obtained from this procedure as a proxy oracle to provide feedback for the suitability of DM-Pruned CADJ graphs.

Modern graph segmentation methods (of which there are many, see, e.g., [4] for a thorough overview of classes and methods) attempt to find communities of vertices in a graph (such as CADJ) using properties of the graph's adjacency matrix. The large and varied methodologies on offer for this task differ in how they incorporate this adjacency information. Typically, for graph-based clustering, individual data vectors would comprise the set of graph vertices; the analyst then specifies a pairwise (dis-)similarity measure as the graph's adjacency matrix to serve as input for the segmentation algorithms. In modern settings, this can be computationally infeasible. The MER image, for example, is 700 x 450 pixels which necessitates 315,000 graph vertices and $\mathcal{O}(10^{10})$ pairwise adjacencies, typically specified as Euclidean distance.

The studies cited above conclude that SOM representation of the data can make these algorithms a feasible clustering tool by clustering the prototypes instead of individual data vectors (there are far fewer prototypes than data vectors; the MER image SOM has only 1600), making the clustering not only automated, but very fast (with computation time of just a few seconds for most SOMs). Additionally, we have found that using CADJ as the graph adjacency, instead of the typical Euclidean distance, results in markedly better clusterings that are in line with those produced by interactive human assessment. This success is most pronounced when the right CADJ thresholding scheme is chosen, which motivated the advent of DM-Prune. Of the many graph segmentation algorithms available, we have had most success in the studies cited above with the Walktrap method [19], which we will use

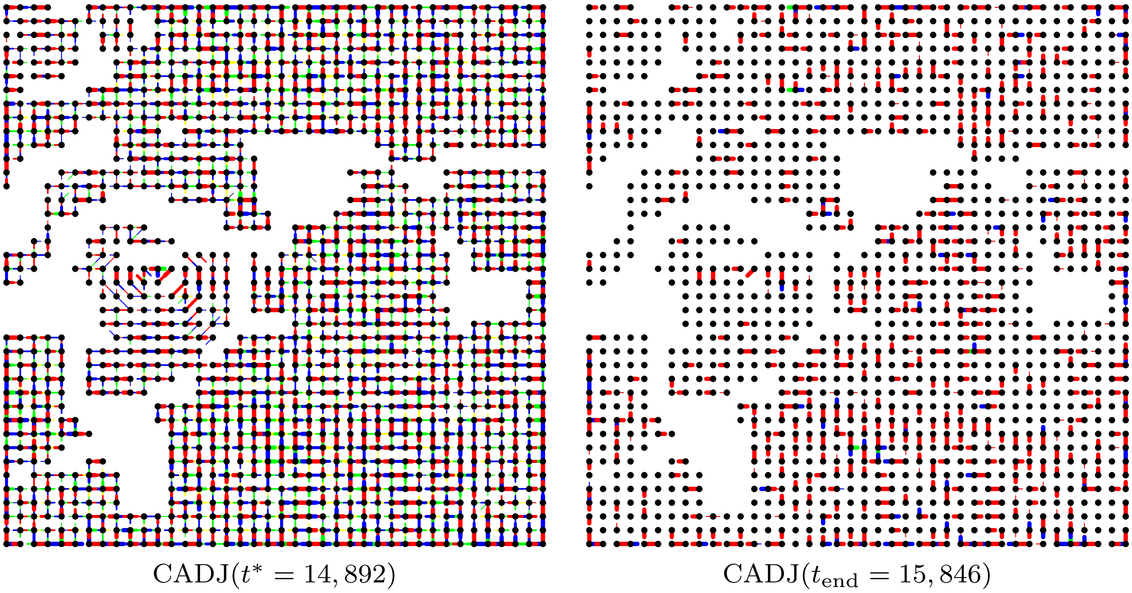


Fig. 5 CADJvis resulting from pruning at $t^* = 14,892$ (left) and $t_{\text{end}} = 15,846$ (right). CADJ(t^*) has much cleaner structure than the unpruned graph visible in Fig. 2, while CADJ(t_{end}) has been pruned so heavily little coherent structure remains. CADJvis is the directed version of CONNvis, showing the asymmetries between connections

(again) in this study, as implemented in the `igraph` package [2].

In the next section, we will evaluate the quality of the `DM-Prune` procedure by allowing Walktrap to cluster its pruned graphs and performing subsequent assessment on the resulting clusterings. Before doing so, however, we make one final contribution to parameterizing the automated graph segmentation process. Invoking the edge-scoring metric μ_{jk} (1), we define a new prototype similarity matrix with entries

$$\mu\mathrm{ADJ}_{jk}(t) = \mu_{jk} * \mathrm{CADJ}_{jk}(t) \tag{6}$$

to pass off to Walktrap. μADJ now contains additional information unavailable in CADJ alone (the SOM length measure L_{jk} and the sampling noise measure S_{jk}). Experiments in the next section will show the addition of μ valuable to Walktrap-based cluster discovery.

4.3 Computational considerations

Once its inputs (CADJ and prior $\mathcal{U}$) are available, constructing the Λ-path is relatively lightweight, involving only repeated evaluation of the DM pmf (3) to build the path itself and a univariate normal pdf [1] for its changepoint analysis. For example, the path presented in Fig. 4a required $\approx$ 10 seconds of computation on a 2.4 GHz Intel Core i9 processor. This time excludes both the SOM learning (which can be very fast in parallel hardware [9], seconds for the data in this paper) and subsequent computation of the max. volume ellipsoids which underly the prior $\mathcal{U}$; the latter is the most demanding, requiring $\approx$ 20 minutes on the same CPU. All steps in this analysis are amenable to parallel computation, although we feel more substantial reductions in computation time can be achieved by exploring alternative estimators of $\mathcal{U}$, via either early termination of the algorithm of [25] or appealing to other polytope approximators such as the Dikin ellipsoid [3]. Such optimization aspects are planned for future work.

5 Clustering the MER image with `DM-Prune`

.

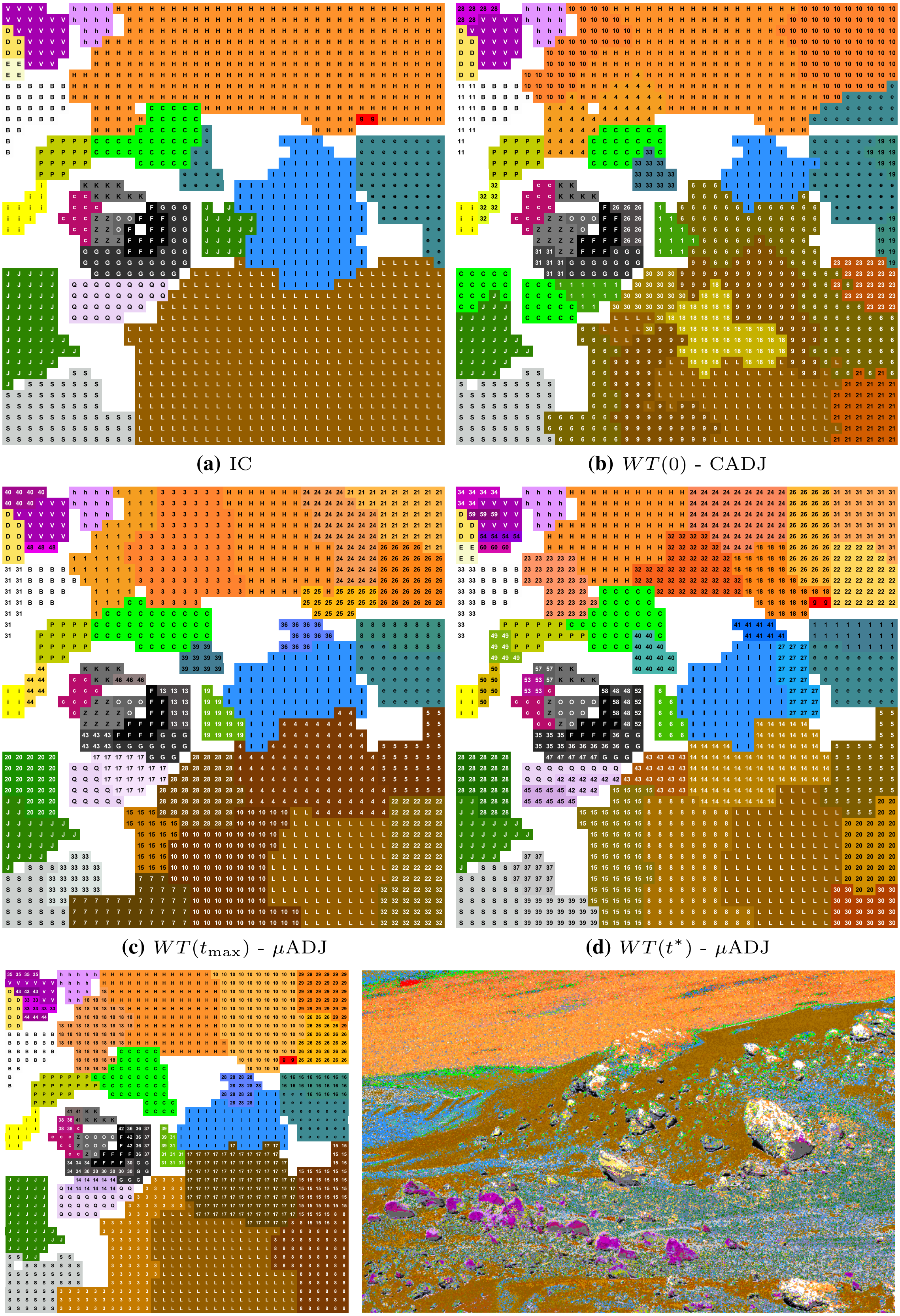
The following clustering results for MER were all produced in an automated fashion by the Walktrap algorithm utilizing the $\mu\mathrm{ADJ}(t)$ similarity defined above, unless otherwise noted. For exposition, we present different clusterings resulting from pruning at various times t along MER's Λ-Path, which we denote by $WT(t)$. Walktrap does have one tuning parameter ("steps," in [19]) which has been set at its recommended default (= 4) in this study.

Fig. 6 MER clustered SOMs via **a** human CONNvis analysis and **b–e** ▶ the `DM-Prune`-ing strategy using the indicated prune step and prototype similarity. **f** The MER image clustered via $WT(t^*)$ showing good spatial agreement to the IC clustering of Fig. 1

The experimental clusterings discussed in this section have been *reconciled* to the previously cited interactive clustering (shown in Fig. 1, abbreviated by IC below), which we consider a baseline for this comparative study. The reconciliation process matches clusters from a $WT(t)$ clustering to the IC by a plurality pixel vote (i.e., by the number of shared image pixels between clusters of each clustering). During reconciliation, the best matching $WT(t)$ cluster inherits its corresponding IC cluster's label and color; second, third, and so on best matching clusters will keep their label, but inherit a color similar to their closest IC cluster. Labels of the IC clustering are all alphabetical, while Walktrap clusters received integer labels. This process is done to aid the human eye during comparative clustering assessments.

Figure 6 shows the clustered SOMs used for comparison. The IC clustering is repeated at the top left (6a) for convenient reference. The top right panel (6b) shows the clustering $WT(0)$ using the original CADJ (i.e., without any `DM-Prune` involvement) as graph adjacency. We include this clustering here to give the reader a baseline to gauge pruning performance. In 6b we see substantial bleeding between cluster boundaries: the separation between e and H, I / L / S, and C / J is not well formed. As these are all sand or soil mixtures, this boundary bleeding is not as scientifically dissonant as the SOM might make it appear. Of greater concern, Walktrap using the unpruned CADJ has failed to identify clusters E (pale yellow, northwest quadrant of SOM), g (red, northeast) and Q (lilac, southwest). Clusters E (541 total pixels) and g (262 pixels) are more rare, but Q is larger (4,758 pixels); its omission poses more concern. In all, panel 6b highlights the need to prune the CADJ graph to facilitate clean and complete cluster discovery.

The $WT(t_{\max})$ clustering of panel 6c (using the modified adjacency μADJ, instead of CADJ) has improved much of the cluster boundary bleeding discussed above. Further, at $t_{\max}$ cluster Q has been separated from C. We are, however, still missing the more rare clusters E and g from panel 6c, suggesting further CADJ pruning would be beneficial. As the Λ-Paths of Fig. 4 collectively show no noticeable degradation for pruning in a small extended window beyond $t_{\max}$ we selected $t^* = 14,892$ (as identified by the changepoint analysis) to halt the pruning process. The resulting Walktrap clustering in panel 6d has retained much of the cleanup observed in 6c and finally separates the small clusters E and g from their parents. Figure 7 (left



(a) IC

(b) $WT(0)$ - CADJ

(c) $WT(t_{\max})$ - μADJ

(d) $WT(t^{*})$ - μADJ

(e) $WT(t^{*})$ - CADJ

(f) MER Clustered Image of $WT(t^{*})$

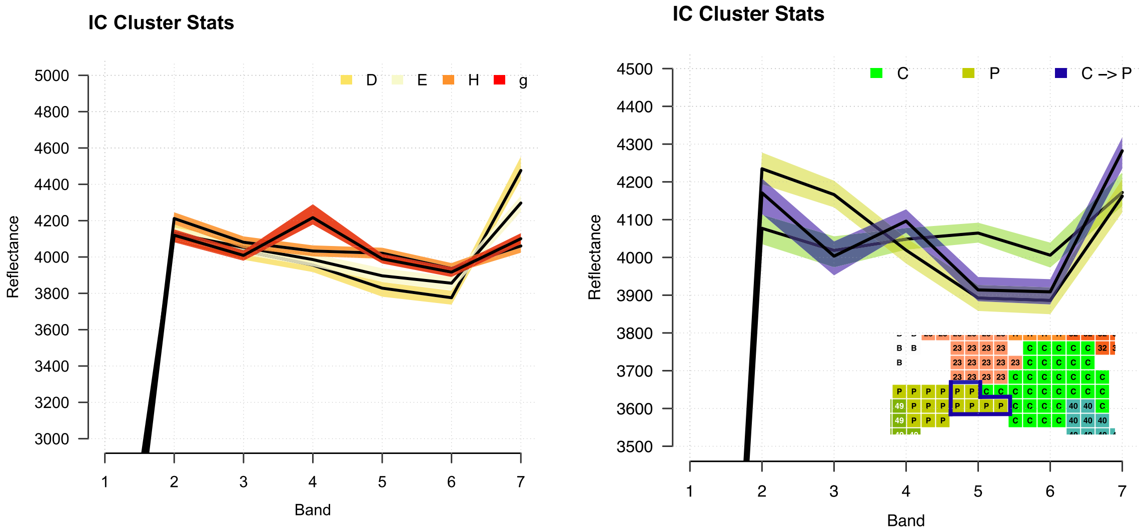


Fig. 7 Band-wise statistics of the spectra of Left: IC clusters D,E,H and g and Right: C, P and the C⟶P merged region of the $WT(t^*)$ clustering. Black lines represent median cluster spectra; shaded regions are the IQR of each cluster

panel) displays the median spectral signatures (black lines) of IC clusters D, E, H and g, along with the band-wise interquartile range of each cluster (shaded according to cluster color). In it, we see that clusters D and E have very similar spectra. As noted in [14], these clusters likely represent the same material (Jibsheet rock) with slight spectral differences due to heterogeneous textures on the surfaces of the rocks. Because these spectra are so similar, it is not surprising that further pruning (at t^* vs. $t_{\max}$) is needed to distinguish them. The spectral differences between g and H, however, are much more pronounced. It is suggested in [14] that g could be a dust devil (whirling cloud of dust) in Spirit's frame at the time of imaging; if so, it is not unexpected that cluster g remained hidden within H (which is a large heterogeneous cluster collectively representing airfall dust) until further pruning at t^*. While the stark spectral distinction between g and H would likely have flagged a human's attention at less severe pruning (we can easily identify g in the CONNvis of CADJ$(t_{\max})$, not shown), our goal for `DM-Prune` is to arrive at a pruned graph that produces sensitive AND automated clusterings. The experienced human eye can detect nuance in CONNvis but an algorithm needs more explicit direction, hence our suggestion to examine graph prunings beyond $t_{\max}$.

The one apparent degradation that is visible between the $WT(t^*)$ and IC clusterings is the destruction of the clean boundary between clusters P (Bowline type rock) and C (sand mixture), where $WT(t^*)$ has merged more of C into P. In the right panel of Fig. 7, we provide the cluster statistics of the IC cluster P, the part of C that was not merged with P, and the $WT(t^*)$ induced overlap (which we call C⟶P and color blue temporarily for this discussion). Examining the plot at right, it is clear that the spectra of C⟶P more closely resemble those of P, except at the third image band where they match C well. This indicates that the spectral delineation in the IC between C and P is not as clean as it could be. Because of this, we consider this instance of boundary bleeding to be potentially valid and put it forth as an example of the types of discoveries that can be made from automating the SOM clustering process. Unlike humans, computers never tire and their judgment remains constant over time.

5.1 Efficacy of μADJ

We finally address the impact of incorporating the μ scores into the graph adjacency weights prior to invoking Walktrap. Intuitively, this makes sense, as CADJ alone is blind to both its lattice representation and any sampling variability inherent in our observed data. The clustered SOM in Fig. 6e reports the Walktrap clustering obtained from using CADJ(t^*) instead of μADJ(t^*). Again we see bleeding between clusters I, L and S, although that is not alarming as discussed above. Also, the rare cluster g is retained in this clustering, which could have been predicted as it forms a well-separated island in the CADJvis of Fig. 5. However, CADJ alone pruned at this level fails to signal the presence of cluster E strongly enough for Walktrap to isolate it. The well-formed cluster attached to the tail of C is also omitted. Interestingly, the boundary bleeding between clusters C and P discussed at length above is also present here, lending further support to the hypothesis that the original C–P boundary could use re-examination. Overall, additional weighting of CADJ edges by their respective μ score appears valuable for Walktrap-based cluster discovery. This will be further explored in follow-up studies.

6 Conclusions and outlook

The key to obtaining the most sensitive clusterings from CADJ/CONN graphs is to determine the subset of edges relevant for the representation of the most salient parts of the underlying manifold topology, which is accomplished by intelligent graph pruning. To achieve this, `DM-Prune` invokes the Λ-path, based on a Dirichlet-multinomial likelihood of CADJ weights, to signal an appropriate level of pruning. As examined in Sect. 4, Λ is only able to convey such information when the size (volume) of the second-order Voronoi cells underlying the CADJ values is properly incorporated into the model. Ultimately, this provides a type of effect-size analysis for CADJ, which helps ensure edges representing the most effective topological connectivities are preserved during edge pruning. Variance changepoint analysis of the Λ-path appears suitable for identifying candidate levels of pruning which retain vital local connectivities without destroying meaningful cluster structure. For the time being, we have relegated the selection of the best candidate pruning level among those identified via changepoint analysis to the human analyst. In future work, we will explore more formal analysis of the variance likelihoods underlying the changepoint analysis in hopes of informing this decision via proper statistical significance tests. Collectively, these steps move us closer to fully automated, high-quality cluster extraction from the SOM.

Acknowledgements Many thanks to MER scientist Dr. William Farrand from the Space Science Institute, Boulder, CO, USA, for the use of the MER imagery and verification of the IC clustering.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Chen J, Gupta AK (1997) Testing and locating variance changepoints with application to stock prices. J Am Stat Assoc 92(438):739–747
2. Csardi G, Nepusz T (2006) The igraph software package for complex network research. InterJournal Complex Systems 1695
3. Dikin I (1967) Iterative solution of problems of linear and quadratic programming 174(4):747–748
4. Fortunato S (2010) Community detection in graphs. Phys Rep 486(3–5):75–174
5. Hinkley DV, Hinkley EA (1970) Inference about the change-point in a sequence of binomial variables. Biometrika 57(3):477–488
6. Horvath L (1993) The maximum likelihood method for testing changes in the parameters of normal observations. Ann Stat 21(2):671–680
7. Killick R, Eckley IA (2014) changepoint: an R package for changepoint analysis. J Stat Softw 58(3):1–19
8. Kohonen T (2000) Self-organizing maps. Springer, Berlin
9. Lachmair J, Merényi E, Porrmann M, Rückert U (2013) A reconfigurable neuroprocessor for self-organizing feature maps. Neurocomputing 112:189–199
10. Luxburg U (2007) A tutorial on spectral clustering. Stat Comput 17(4):395–416. https://doi.org/10.1007/s11222-007-9033-z
11. Martinetz T, Schulten K (1994) Topology representing networks. Neural Netw 7(3):507–522
12. Merényi, E., Taylor, J.: Empowering graph segmentation methods with SOMs and CONN similarity for clustering large and complex data. Neural Comput Appl pp 1–18 (2019)
13. Merényi E, Taylor J, Isella A (2016) Deep data: discovery and visualization application to hyperspectral ALMA imagery. Proc Int Astron Union 12(S325):281–290
14. Merényi E, Tasdemir K, Farrand WH (2008) Intelligent information extraction to aid science decision making in autonomous space exploration. In: Fink W (ed) Space Exploration Technologies, vol 6960. International Society for Optics and Photonics, SPIE, pp 155–169
15. Merényi E, Taylor J (2017) SOM-empowered graph segmentation for fast automatic clustering of large and complex data. In: 2017 12th International workshop on self-organizing maps and learning vector quantization, clustering and data visualization (WSOM), pp 1–9
16. Merényi E, Taylor J, Isella A (2016) Mining complex hyperspectral ALMA cubes for structure with neural machine learning. In: 2016 IEEE symposium series on computational intelligence (SSCI), pp 1–9
17. Mosimann JE (1962) On the compound multinomial distribution, the multivariate beta-distribution, and correlations among proportions. Biometrika 49(1–2):65–82
18. Okabe A, Boots B, Sugihara K, Chiu SN (2000) Spatial tessellations: concepts and applications of voronoi diagrams, 2nd edn. Series in probability and statistics. John Wiley and Sons Inc, New Jersy
19. Pons P, Latapy, M (2005) Computing communities in large networks using random walks. In: Proceedings of the 20th international conference on computer and information sciences, ISCIS'05, pp 284–293. Springer-Verlag, Berlin
20. Taşdemir K, Merényi E (2011) A validity index for prototype based clustering of data sets with complex structures. IEEE Trans Syst Man Cybern Part B 41(4):1039–1053. https://doi.org/10.1109/TSMCB.2010.2104319
21. Taşdemir K, Merényi E (2008) Cluster analysis in remote sensing spectral imagery through graph representation and advanced som visualization. In: Jean-Fran JF, Berthold MR, Horváth T (eds) Discovery Science. Springer, Berlin, pp 259–271
22. Taşdemir K, Merényi E (2009) Exploiting data topology in visualization and clustering of self-organizing maps. IEEE Trans Neural Netw 20(4):549–562
23. Taylor J, Merényi E (2020) A probabilistic method for pruning CADJ graphs with applications to SOM clustering. In: Vellido A, Gibert K, Angulo C, Guerrero JDM (eds) Advances in self-organizing maps, learning vector quantization, clustering and data visualization. Springer International Publishing, Cham, pp 44–54
24. Zelnik-Manor L, Perona P (2004) Self-tuning spectral clustering. In: Proceedings of the 17th International Conference on Neural

Information Processing Systems, NIPS’04, p. 1601–1608. MIT Press, Cambridge
25. Zhang Y, Gao L (2003) On numerical solution of the maximum volume ellipsoid problem. SIAM J Optim 14(1):53–76

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.