SKATER-CON: Unsupervised Regionalization via Stochastic Tree Partitioning within a Consensus Framework Using Random Spanning Trees

Research Paper

Orhun Aydin
Environmental Systems Research Institute
Redlands, California
OAydin@esri.com

Mark V. Janikas
Environmental Systems Research Institute
Redlands, California
mjanikas@esri.com

Renato Assunção
Departmento de Ciéncia da Computação
Belo Horizonte, Brazil
assuncao@dcc.ufmg.br

Ting-Hwan Lee
Environmental Systems Research Institute
Redlands, California
Ting_Lee@esri.com

ABSTRACT

Spatially constrained clustering, also known as regionalization, aims to group spatial objects into spatially contiguous clusters also known as regions. Among different approaches, tree-based partitioning is reported to define homogeneous regions rigorously, without ad-hoc adjustments, in a computationally efficient manner. One of the shortcomings of tree-based partitioning is the so-called chaining problem that results in sub-optimal regions. We propose a consensus-based regionalization approach to address the chaining problem associated with a single tree, in particular the minimum spanning tree, by exploring a wide range of partitions via a set of random spanning trees (RST). We propose an algorithm, namely SKATER-CON, that partitions spatial data via a consensus-based framework from an ensemble of regionalizations defined by its deterministic counter-part, the SKATER algorithm applied along stochastic search paths defined by RSTs. SKATER-CON utilizes evidence accumulation to represent an ensemble of regionalizations as a similarity graph. The similarity graph represents spatial objects as vertexes and frequency at which objects are assigned to the same region in the ensemble as edge weights. Proposed algorithm determines consensus among different regionalization by partitioning the similarity graph using a multi-level graph partitioning algorithm (METIS). Spatial constraints are imposed on the similarity graph prior to partitioning to ensure spatial constraints are reflected in the consensus result. We rigorously test the quality of regions produced by SKATER-CON on a large, synthetically generated dataset. The synthetic dataset is the result of full-factorial experiments designed on number, fuzziness, geometry and size of regions. Same dataset is also used compare our approach against state-of-the-art regionalization algorithms (SKATER and ARISEL). Lastly, we show the value added by SKATER-CON compared to SKATER on a real-world dataset based on Ecological Marine Units (EMU) dataset.

CCS CONCEPTS

• **Computing methodologies** → **Cluster analysis**; *Spatial and physical reasoning*; *Randomized search*; • **Mathematics of computing** → *Paths and connectivity problems*;

KEYWORDS

Clustering, consensus clustering, constrained optimization, graph partition, evidence accumulation

ACM Reference Format:
Orhun Aydin, Mark V. Janikas, Renato Assunção, and Ting-Hwan Lee. 2018. SKATER-CON: Unsupervised Regionalization via Stochastic Tree Partitioning within a Consensus Framework Using Random Spanning Trees: Research Paper. In *2nd ACM SIGSPATIAL International Workshop on AI for Geographic Knowledge Discovery (GeoAI'18), November 6, 2018, Seattle, WA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3281548.3281554

1 INTRODUCTION

Spatially constrained clustering, also known as regionalization, is an unsupervised machine learning technique in spatial analysis to discover spatially contiguous clusters, also known as regions. Regionalization is a type of constrained clustering problem [4] where clusters are based jointly on similarity in variable values and proximity in space. Regionalization has found a wide spectrum of uses from delineating distinct climate zones [29] to defining health-care regions [6].

Graph representation of spatial connectivity/neighborhood constraints have been integrated into widely used methodologies in numerous areas such as statistical image analysis [15], fuzzy clustering [44] and texture extraction [23]. Similar to methods pertaining to image analysis, methods for spatial data have also extensively

GeoAI'18, November 6, 2018, Seattle, WA, USA

ACM ISBN 978-1-4503-6036-4/18/11…$15.00
https://doi.org/10.1145/3281548.3281554

used graph-based representations [11]. Graph-based representation of spatial objects allows for posing regionalization as a graph partitioning problem [11][27][2]. One of the graph-based methods, the SKATER algorithm [2], reduces the full graph representation of spatial objects to a tree representation that can be easily pruned to create regions. SKATER uses the minimum spanning tree as the search path visit spatial objects to define regions. One of the main concerns with regionalization algorithms are artifacts due to spatial contiguity constraints, especially artifacts that impact the homogeneity of regions [19]. In the context of regionalization, homogeneity ensures neighboring locations with similar values are assigned to the same region.

We propose to use alternative trees in the SKATER solution to obtain a variety of weak (searched not along the optimal path) regionalizations of spatial data. Introducing weak regionalizations allows for searching a wide variety of region configurations that may not be discovered along the minimum spanning tree [37]. In the context of our work, every weak regionalization is treated as vote on the final spatially-constrained clusters. We apply a weighting scheme to weight regionalization votes with respect to the optimality of the path used to define regions within SKATER. Finally, a final regionalization is defined as the consensus-vote of all the weak regionalizations.

We use random spanning trees [1][43][35] to define alternative paths to SKATER algorithm. We sample different paths from a full-graph with via a loop-erased random walk on full-graph's vertexes [43][1]. Weights of random spanning trees are used to weigh weak regionalization votes. The weighting scheme allows incorporating optimality of the search path used to define regions in the final consensus. We define a final regionalization as a consensus regionalization of all votes from weak learners. We use an evidence accumulation (EA) framework [14] to represent the ensemble of regionalizations as a similarity matrix. The consensus regionalization is defined on the similarity matrix obtained from votes of weak regionalizations.

Different methods exist to define consensus among different votes from different clusterings of data [39][14][13]. In contrast to previous consensus methods, regionalization requires imposing spatial contiguity constraints on the final consensus vote. We represent different votes using graph-based approach [39], where edges of the graph are defined with similarity in votes from different weak regionalizations. Graph-based partitioning gives the ability to impose contiguity constraints on the final consensus of regionalization votes because contiguity constraints can be expressed as an adjacency graph. We use a multi-level graph partitioning method METIS, to partition the final graph to create spatially contiguous clusters for a desired number of regions.

2 RELATED WORK

Different types of approaches to regionalization include two-stage conventional clustering algorithms [33], explicit optimization on region characteristics [3][10], hierarchical clustering [28][12], agglomerative clustering [19], improving feasible initial solutions [10][9] and graph-based solutions [27] [2] [40].

Early methods define regions with a multi-step approach where non-spatial clustering results are further processed to impose spatial constraints [33] [31]. Methods that incorporate proximity as input to non-spatial clustering were also used to describe spatial clustering in data [30] [38]. Location-allocation formulation is modified for regionalization problems to assign regions to spatial units based on their similarity to defined region centers [42]. More recent methods were proposed that pose regionalization as an optimization problem where boundaries of regions were iteratively optimized with respect to an objective function, such as the AZP algorithm [32]. Improvements to the optimization scheme for AZP is proposed via simulated annealing and TABU search heuristic [34]. Following the success of optimization-based methods, ARISel is proposed to further the enhance the stability and performance of clustering methods via better initial seed for region optimization [9]. Convergence and run-time is sensitive to initial seeding, however ARISEL algorithm is shown to produce compact regions upon convergence. A detailed survey of early methods and optimization-based regionalization methods are provided by Duque [11].

Graph-based partitioning applied to regionalization starts with Maravelle's work [27]. They proposed a heuristic method, MIDAS, to perform graph partitioning via tree pruning. MIDAS is memory intensive and for large datasets or datasets that are well connected, the graph can get too dense for MIDAS to converge. Assuncao [2] proposed the SKATER algorithm which is one of the building blocks for the proposed methodology. SKATER iteratively prunes the minimum spanning tree to define regions. For every iteration, homogeneity metrics at iteration is calculated for all possible edges to prune. SKATER is a very efficient regionalization algorithm that can be performed faster than all the state-of-the art algorithms. However, it suffers from so-called chaining problem [20] due to its greedy approach. In the context of regionalization, chaining problem manifests itself as suboptimal groups created due to following the least cost path defined by the minimum spanning tree. Initial edges removed affect the final regions drastically. SKATER is a sequential algorithm and optimal initial cuts do not necessarily result in optimal final regions.

3 PROBLEM DEFINITION

Consider n locations with s observed variables at every location.

$$\boldsymbol{X} = \begin{bmatrix} | & | & & | \\ \boldsymbol{x_1} & \boldsymbol{x_2} & \dots & \boldsymbol{x_n} \\ | & | & & | \end{bmatrix} \tag{1}$$

In Eq. 1, any given location i has s different features where $\boldsymbol{x_i} = (x_{i,1}, x_{i,2}, \dots, x_{i,s}) \in \mathbb{R}^s$. Spatially constrained clustering aims to create k spatially contiguous regions that are homogeneous with respect to $\boldsymbol{X}$. In spatial-constrained clustering two objects can be assigned to the same cluster if they are similar in value and proximal in space. Thus, any regionalization algorithm needs to take spatial relationships into account. In spatial statistics, spatial relationships are represented using spatial weights matrices [1][2]. The general spatial weights matrix is given in Eq. 2

$$\boldsymbol{A} = \begin{bmatrix} 0 & & a_{1,n} \\ & \ddots & \\ a_{2,1} & & a_{2,n} \\ a_{n,1} & & 0 \end{bmatrix} \tag{2}$$

where $a_{i,j}$ is an indicator variable for location i and j's neighborhood with $a_{i,j} = a_{j,i}$. Neighborhood relationships between locations are defined as follows:

$$a_{i,j} = \begin{cases} 1, & d_{euc}(i,j) \leq \epsilon \\ 0, & d_{euc}(i,j) > \epsilon \end{cases} \tag{3}$$

Eq. 3 defines two locations i and j to be neighbors if they are within a certain distance ϵ with respect to a distance metric such as Euclidean distance d_{euc}. However, different conceptualization of spatial relationships exist such as k-neighbors and contiguity-based neighborhood [16][17].

Regionalization can be expressed as a constrained optimization problem of defining groups of objects without breaking spatial contiguity with respect to $\boldsymbol{A}$ per region $\{R_1, \ldots, R_k\} \in R$.

$$\begin{aligned} &\arg\min_{R} = \sum_{i=1}^{k} \sum_{j \in R_i} d\left(\boldsymbol{x_j}, \mu_{\boldsymbol{R_i}}\right) \\ &\text{subject to} \sum_{j \in R_q} \boldsymbol{A}[i,j] \geq 1 \qquad \forall i \in R_q, \forall q \in \{1, \ldots, k\} \end{aligned} \tag{4}$$

where $\mu_{\boldsymbol{R_i}}$ is the mean for region R_i and d is a distance measure, frequently used as the Euclidean distance. The Optimization problem in Eq. 4 requires finding regions R such that values within a region are homogeneous and the location of values are contiguous. We define a general operator $\mathcal{L}(\boldsymbol{X}, \boldsymbol{A}) : \rightarrow R$ where $R = \{R_1, \ldots, R_k\}$.

An efficient $\mathcal{L}(\boldsymbol{X}, \boldsymbol{A})$ does not require ad-hoc adjustments on aspatial clustering of data, $\mathcal{L}(\boldsymbol{X})$ based on spatial constraints, $\boldsymbol{A}$ [11]. One of the approaches to efficient regionalization is through representing $\boldsymbol{X}$ and $\boldsymbol{A}$ jointly and using efficient clustering algorithms on this new representation.

3.1 Graph-Based Approach to Spatially Constrained Clustering

One of the approaches to define an efficient $\mathcal{L}$ is through graph-partitioning. In our study, we represent spatial information using a weighted, undirected graph $G(V, E, l)$. Locations of spatial objects are represented with vertices, $V = V(G) = \{v_1, \ldots, v_n\}$ where $|V(G)| = n$ and neighborhood relationships between spatial objects are represented with edges, $E = E(G)$, where $|E(G)| = m$ is the number of neighboring pairs. Similarity of observed variables in each node is represented as pairwise edge weights where $w_{i,j}$. Edge weight $w_{i,j}$ for an edge $e_{i,j} \in E(G)$ is defined with a distance function $d\left(\boldsymbol{x_i}, \boldsymbol{x_j}\right)$ based on the vector of features $\boldsymbol{x_i}$ for object i. Graph representation of spatial objects allows partition operators to be used in clustering while preserving spatial contiguity constraints. The general regionalization operator acting on a graph G is denoted as

$$\mathcal{L}(G) = G^* = \{R_1, \ldots, R_k\} \tag{5}$$

Subgraph $G^* \subset G$, where $|V(G^*)| = n$, consists of spatially contiguous regions $R = \{R_1, \ldots, R_k\}$. Note that $G - G^* = E_{cut}$ where E_{cut} is the set of edges removed to partition G into spatially contiguous regions. Graph-based approach to spatially constrained clustering is depicted in Fig. 1.

Regionalization operator $\mathcal{L}$ defines spatially contiguous regions for the data in Fig. 1. Original graph G is depicted over the original dataset and partitioned graph G^* is over the regionalization result. The only difference between G and G^* in Fig. 1 is E_{cut}, edges removed to define R. Different methods define E_{cut} with respect to different criteria. The quality of regions created depends entirely on E_{cut}. The general quality metric for regionalization is a measure of homogeneity in values per region. General homogeneity metric $f_h(G^*)$ is expressed in Eq. 6

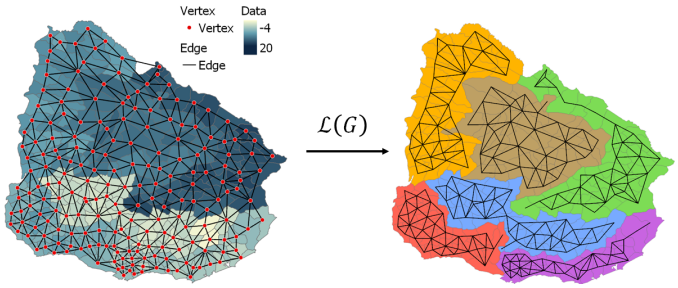

Figure 1: Graph Representation of Spatial Data (left) and region map from spatially-constrained clustering (right). Partitioned subgraph G^* is overlayed on regions (right).

$$f_h(G^*) = \sum_{i=1}^{k} \sum_{j \in R_i} d\left(\boldsymbol{x_j}, \mu(R_i)\right) \tag{6}$$

where, $d\left(\boldsymbol{x_j}, \mu(R_k)\right)$ is a measure of variation per region with respect to some centrality measure $\mu(R_k)$ associated with region R_k such as the mean or the median.

3.2 Tree-based Regionalization: SKATER Algorithm

Finding the optimal E_{cut} that maximizes Eq. 6 is a computationally intensive task, especially for large datasets that are spatially well-connected, in other words for large m. A branch of graph-based approaches to regionalization uses spanning trees to reduce the number of edges to search from m to n [2] [27]. Tree-based approaches gained popularity in graph-based regionalization area because for spatial problems $n \ll m$. Assuncao et al. [2] proposed to reduce the spatially-constrained clustering problem to a tree-partitioning problem through the use of a spanning tree T. The Spatial 'K'Luster Analysis by Tree Edge Removal (SKATER) algorithm (Assuncao) uses the minimum spanning tree $T_{MST}(V, E)$ where $V(T_{MST}) = (G)$ and $E(T_{MST}) \subset E(G)$ where $|E(T_{MST})| = n - 1$. SKATER algorithm uses T_{MST} as a path to visit all spatial locations to define E_{cut}. This approach reduces the number of neighbors from m to $n - 1$ [18]. Removing an edge from T_{MST} results in two sub-trees, T^+_{MST} and T^-_{MST}, on either side of the removed edge. These sub-trees represent two regions. SKATER removes edges from E_{MST}, iteratively, picking the edge that maximizes Eq. 6. Initially, T_{MST} represents one region and it is partitioned by SKATER into k subtrees, $T^* = \{T_1, \ldots, T_k\}$ that span regions $R = \{R_1, \ldots, R_k\}$, respectively.

SKATER removes an edge in T^* at every iteration that maximizes the objective function in Eq. 7.

$$f_{obj}\left(e_{i,j}\right) = f_h(T) - f_h\left(T^+\right) - f_h\left(T^-\right) \tag{7}$$

Eq. 7 quantifies the change in homogeneity by splitting the region represented with T into two regions T^+ and T^-. In their original work, Assuncao et al. [2] define f_h as the intra-cluster square deviation (SSD) where homogeneity is quantified with respect to deviation from region mean, $\mu(R_k)$.

SKATER algorithm is depicted in Fig. 2.

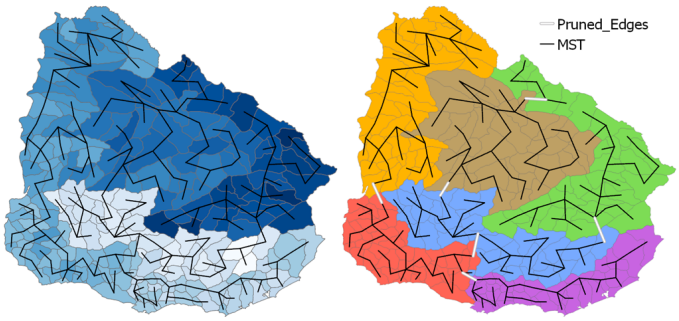

Figure 2: Spatial Data with T_{MST}, search path, overlaid(left). Region map defined by tree edge pruning (right). Pruned edges from T_{MST} are colored in white

Fig. 2 displays on left the spatial data and the corresponding minimum spanning tree overlaid. Edges from the minimum spanning tree are sequentially pruned (displayed with white lines) in order to define the underlying regionalization map in Fig. 2.

SKATER algorithm is presented in Algorithm 1.

Algorithm 1 Regionalization via Tree- Partitioning: SKATER

INPUT: Number of regions k, data matrix $\boldsymbol{X}$, spatial weights matrix $\boldsymbol{A}$
OUTPUT: R - regions

```
k_target ← k
subtrees ← T_MST
edges ← E(T_MST)
while k_current < k_target do
    f_obj ← 0
    edge_remove ← 0
    for all edge in edges do
        f_edge ← f_obj(edge)
        if f_edge > f_obj then
            f_obj ← f_edge
            edge_remove ← edge
        remove edge_remove from edges
        update subtrees
return subtrees
```

Homogeneity of the regions defined by SKATER strongly depends on the initial edges removed. SKATER prunes edges sequentially and every pruned edge limits the search space. In sequential tree-based partitioning algorithms an optimal split in early stages does not necessarily result in overall optimality of the regionalization. The memory effect in sequential tree splits is referred to as the chaining problem. Chaining is known to impact clustering algorithms that rely on iteratively defining clusters such as hierarchical clustering [14].

In the context of spatial analysis, SKATER visits spatial objects along T_{MST}, the path of highest similarity between neighbors. Removing an edge from $E(T_{MST})$ does not guarantee an optimal solution to Eq. 6 for a desired k. Thus with optimal initial cuts, a final optimal regionalization may not be achieved while honoring spatial contiguity constraints. In cases where small-scale variation exists chaining can result in defining singleton regions, in other words a region that consists of only one location. This shortcoming is amplified when a low number of regions is defined because a sub-optimal initial partitioning cannot be compensated for with future edge removals.

Chaining problem in tree-based regionalization is elaborated for a synthetically generated case in Fig. 3.

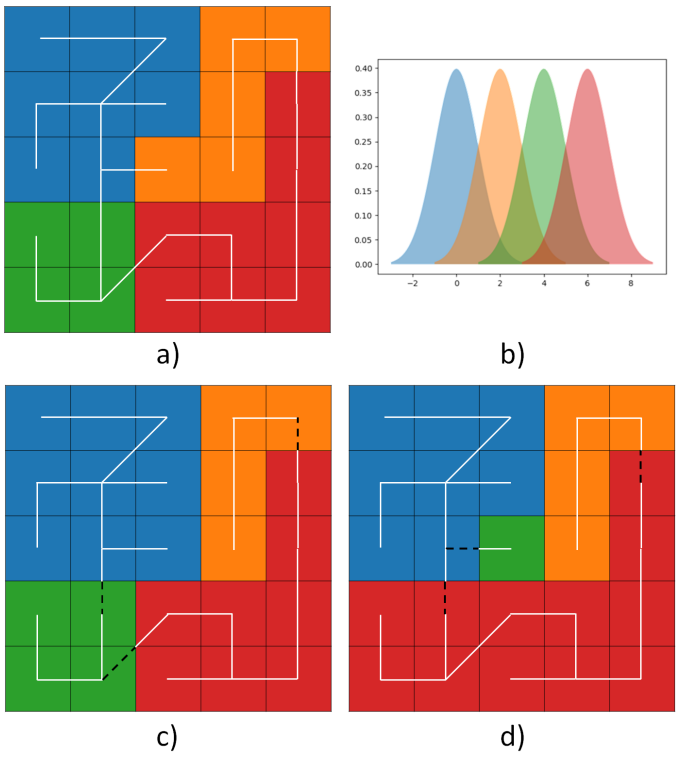

Figure 3: a) Underlying regions, different colors correspond to different regions. Minimum spanning tree is depicted in white b) Marginal probability distributions of values observed at different regions. Regions and their corresponding marginal probability distributions are color-coded c-d) Depiction of two possible tree-cuts to define spatially contiguous regions. Dashed lines indicate cut edges to define regions

Fig. 3 depicts the impact of chaining on regionalization result. Optimal cuts along the minimum spanning tree for a predefined k can yield a sub-optimal regionalization such as the one depicted in Fig. 3 (singleton region). Even though tree-based approach provides

computational efficiency in graph partitioning, the minimum spanning tree limits the number of edges that can be searched during the regionalization process.

4 CONSENSUS CLUSTERING UNDER SPATIAL CONTIGUITY CONSTRAINTS

We propose to address the SKATER algorithm's chaining problem by exploring a wider set of edges to cut in region definition. We propose to perform tree-cut for an ensemble of spanning trees. For aspatial problems, consensus-based methods are shown to outperform the individual cluster result in the ensemble [14][39]. In the context of the regionalization, we form consensus among different regionalizations defined via different paths (spanning trees). Defining a final regionalization as a consensus of regionalizations following sub-optimal paths results in regions obtained by stochastically relaxing the neighborhood relationships modeled with T_{MST} by the SKATER algorithm. We chose SKATER algorithm to partition the ensemble of spanning trees due to its efficiency in partitioning trees and ability to find optimal E_{cut} for a given tree. We represent different regionalizations within an evidence accumulation (EA) framework. EA allows representing regionalization ensemble as pairwise similarity between spatial objects. In the context of EA, similarity is defined as the frequency at which two objects are grouped in the same region among different regionalizations. We represent the similarity matrix defined by EA framework as a graph, where vertexes correspond to spatial objects and edge weights are define using EA similarity, also known as evidence. Graph-based representation of regionalization ensemble allows applying spatial contiguity constraints on the graph prior to defining the consensus. Thus, the proposed methodology allows spatial constraints to be reflected in the consensus result.

4.1 Alternate Search Paths Using Random Spanning Trees

We pose the question of alternate search paths for graph partitions as a sampling problem. We sample alternate spanning trees for G to define different search paths for SKATER. A finite graph G generally has an enormous number of spanning trees T, a number that may be obtained through the Kirchhoff's theorem [22]. However, efficient samplers that generate representative spanning trees exist [26] [1] [43] [25]. We propose to use random spanning trees (RST) [1][35] of G as alternate paths to search regions. RST is a spanning tree chosen randomly among set of all possible spanning trees of G.

Marginal probability distribution of a set of edges $\{e_1, \ldots, ej\}$ belonging to a spanning tree is defined in Eq. 8 [5].

$$\begin{aligned} P\left[e_1, \ldots, ej \in T\right] = P[e_1 \in T] P[e_2 \in T | e_1 \in T] \\ P[e_3 \in T | e_1, e_2 \in T] \ldots P[e_j \in T | e_1, \ldots, e_{j-1} \in T] \end{aligned} \tag{8}$$

Eq. 8 describes the probability of any subset of edges $E_{sub} \subset E\left(G\right)$ belonging to a spanning tree. Set of edges that violate rules for spanning trees such as forming cycles get a probability of 0. The probability of an edge belonging to a sampled tree T_{RST} depends on the probability of the path leading up to the edge. Neighborhood relationships are reflected in sampling spanning trees by using Markov Chains, where the a change in state corresponds to adding an edge and including a new vertex to the spanning tree. In our work, we sample spanning trees using loop-erased-random-walk (LERW) also known as Wilson's Algorithm [43]. LERW sampler for a random spanning tree is given in Algorithm 2.

Algorithm 2 Sampling Alternate Paths from a Weighted Graph

INPUT: Weighted Graph $G\left(V, E, l\right)$
OUTPUT: T_{RST} - Random Spanning Tree

```
node_root ~ U(1, n − 1)
T_RST ← []
for all node_current in V do
    path ← []
    while node_current ∉ T do
        node_next ← RandomWalk(node_current)
        node_current ← node_next
        Add node_current to path
    Add path to T_RST
return subtrees
```

Algorithm 2 performs a random walk on graph G and samples a random spanning tree T_{RST}. The random walk we perform makes use of edge weights and the transition kernel is defined in Eq. 9

$$\mathcal{K}\left(e_{i,j}\right) = \begin{cases} \frac{w_{i,j}^{-1}}{\sum w_i^{-1}}, & e_{i,j} \in E\left(G\right) \\ 0, & e_{i,j} \notin E\left(G\right) \end{cases} \tag{9}$$

The transition kernel in Eq. 9 uses the inverse weight to assign transition probabilities. Algorithm 2 samples spanning trees with following probability.

$$p\left(T_{RST}\right) = \prod_{e \in E(T_{RST})} \mathcal{K}\left(e\right) \tag{10}$$

Eq. 10 defines weighting scheme for regionalization votes obtained from different paths (T_{RST}). Examples of random spanning trees is depicted in Fig. 4.

Random spanning trees depicted in Fig. 4 allows SKATER-CON to search broad range of edges to define regions. Frequency map at the bottom of Fig. 4 shows T_{MST} (black lines at the bottom figure) does not cover all the high frequency edges. In the context of our sampler, edges sampled with high frequency are formed between spatial objects that are similar. Frequency plot shows that SKATER-CON searches some of the edges that may result in compact partitioning that would otherwise would not be visited with SKATER along T_{MST}. Similarly, edges with low weights are also visited by the ensemble of T_{RST}.

4.2 Evidence Accumulation for Spatially-Constrained Clustering

We define $\mathcal{L}^*$ to be a non-optimal regionalization operator in terms of the search path it follows. Every $\mathcal{L}^*$ defines regions along a T_{RST} using SKATER methodology. A regionalizations from $\mathcal{L}^*$ is treated as a vote [8] and the final regionalization is defined as the consensus among all votes. The ensemble of regionalizations $\mathcal{H} = \left\{\mathcal{L}_1^*\left(G\right), \ldots, \mathcal{L}_s^*\left(G\right)\right\}$ need to be combined in a way that reflects are represented with a measure of similarity, the evidence.

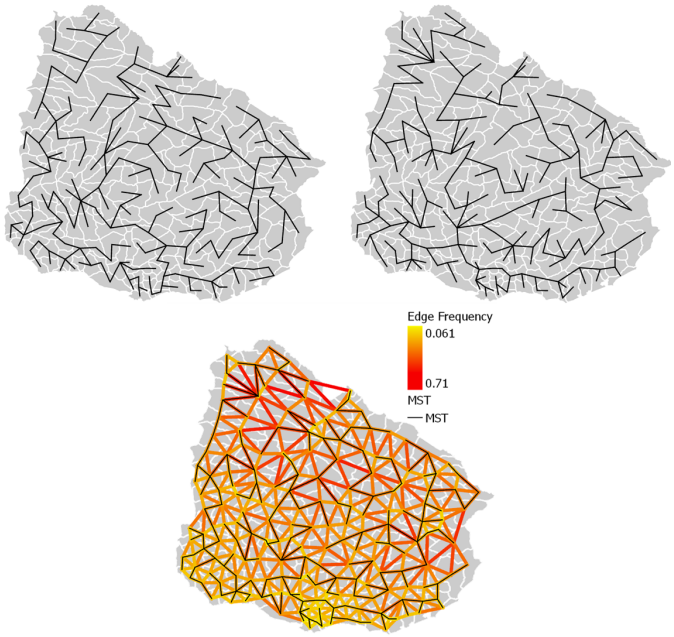

Figure 4: Examples of random spanning trees, T_{RST}, for the same dataset (top). Heat map for edges picked among 1000 random spanning trees with T_{MST} overlaid (bottom)

Final regionalization represents the consensus between different voters, different spatially-constrained clustering runs.

We use an evidence accumulation framework to represent the result of different regionalizations as a similarity matrix. The evidence corresponds to the frequency at which pairs of spatial objects are assigned to the same region throughout different regionalizations. The resulting evidence is represented with a similarity matrix called the co-association matrix, C defined in Eq. 11.

$$C = \begin{bmatrix} s & \dots & s_{1,n} \\ \vdots & \ddots & \vdots \\ s_{n,1} & \dots & s \end{bmatrix} \tag{11}$$

Every off-diagonal element of C is the frequency of co-regionalization, number of times two spatial objects i and j are assigned to the same region.Off-diagonal elements of C is defined in Eq. 12.

$$c_{i,j} = \sum_{\mathcal{L}^*(G) \in \mathcal{H}} \alpha\left(\mathcal{L}^*\right) 1_{(i,j) \in R_l} \tag{12}$$

Eq. 12 defines the sum of weighted votes with respect to the probability of the path followed as defined in Eq. 10.

4.3 Finding Consensus Among Alternate Regionalizations

The general consensus clustering problem is defined as reducing an ensemble of clusterings into a final set of clusters. We have access to a limited number of regionalizations $\mathcal{H} \subset \Lambda$ where Λ is the set of all possible regionalizations of $\boldsymbol{X}$ subject to spatial constraints in $\boldsymbol{A}$. In terms of relaxing spatial constraints to define different regionalizations, Λ will consist of all regionalizations for any $k \in [1, n]$, defined from an ensemble containing all possible spanning trees of G. In our work, we focus on all possible regionalizations for a given k and do not search for different regionalizations of data for changing number of regions.

A consensus function, Γ, combines votes from a limited set of regionalizations $\mathcal{H}$ to a final regionalization that reflects the majority vote $\mathcal{L}_{con} \in \Lambda$. We formulate the consensus problem as follows:

$$\arg\min_{\mathcal{L}_{con}} \Gamma\left(\mathcal{L}_{con}\right) = \sum_{\mathcal{L}^* \in \mathcal{H}} d\left(L_{con}, \mathcal{L}^*\right) \tag{13}$$

Minimizing the consensus function, Γ, in Eq. 13, minimizes the discrepancy between the final regionalization $\mathcal{L}_{con}$ and the regionalization ensemble $\mathcal{H}$. For regionalization, $\mathcal{L}_{con}$ needs to honor the spatial constraints defined in $\boldsymbol{A}$. We propose to represent the co-association matrix C as a similarity graph and we prune edges that does not honor contiguity constrains with following.

$$c_{i,j} = \begin{cases} c_{i,j}, & a_{i,j} = 1 \\ 0, & a_{i,j} = 0 \end{cases} \tag{14}$$

Resulting C is a similarity graph that represents spatial contiguity constraints. A spatially-constrained similarity graph is depicted in Fig. 5.

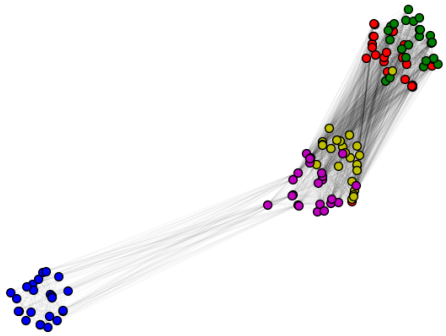

Figure 5: Depiction of similarity graph. Colors are associated with the consensus regions defined by $\mathcal{L}_{con}$. Vertexes are spatial objects edges are $c_{i,j}$

Vertexes in Fig. 5 are placed with respect to their similarity, the frequency at which they were assigned to the same region. The graph in Fig. 5 also reflects spatial contiguity constraints defined by $\boldsymbol{A}$. Colors of vertexes in Fig. 5 represent the result of final regionalizations. Final regionalization is achieved by partitioning the similarity graph. We use the METIS algorithm [21] to partition the similarity graph (similar to work by [39]). Application of METIS to a similarity graph is also called the CSPA algorithm described in-extenso in [39].

4.4 Overall Algorithm and Implementation

SKATER-CON algorithm and all of its dependencies such as SKATER, Wilson's algorithm and evidence accumulation are implemented in Python. METIS package is integrated into the implementation in the version that is made available by the authors to perform the final consensus partitioning.

The overall proposed algorithm is elaborated in Algorithm 3.

Algorithm 3 Consensus-based SKATER:SKATER-CON

INPUT: Number of regions, k, number of random spanning trees n_{sim}, adjacency matrix $\boldsymbol{A}$, observed values $\boldsymbol{X}$
OUTPUT: R_{con} - Consensus Regions

```
vertex, edge ← BuildWeightedGraph (X, A)
coAssociationMatrix ← []
for iteration in 1, ..., n_sim do
    T_RST, α(T_RST) ← Wilson (vertex, edge) ← []
    regionVotes ← SKATER(T_RST, X)
    coAssociationMatrix ← Accumulate(regionVotes.α(T_RST))
    Add node_current to path
    Add path to T_RST
R_con ← METIS(coAssociationMatrix, A)
```

5 EXPERIMENTS

5.1 Synthetic Test Dataset

We compare our method to deterministic SKATER and ARISEL algorithms. Clustering quality of all algorithms are evaluated on synthetically generated datasets. We generated ground truth spatial regions we call the region mask and stochastically simulated values for every location from distributions defined per region. For every region we sample values from the normal distribution $N(\mu(R_i), 1)$ where $\mu(R_i)$ is the mean for region i. We control region fuzziness with $\Delta\mu(R)$, mean separation between regions. For $\Delta\mu(R) \gg 1$, we have distinct regions without any overlapping values. For small $\Delta\mu(x)$, the different regions get fuzzy and the effects of underlying region mask may not be observed on the resulting simulations. An example of a region mask for a given k and associated simulations with different mean separation between regions are illustrated in Fig. 6.

For an effective regionalization algorithm, we would like to retrieve the mask in Fig. 6, especially for spatially distinct simulations such as $\Delta\mu(R) = 4$. However, for fuzzy regions we will be relying on unsupervised metrics of quality. Our simulation deck contains fuzzy cases where simulations do not reflect underlying regions. In these cases, quality metrics that compare regionalization results to underlying regions will underestimate the power of clustering methods. For this reason, we present a quality metric which quantifies quality with respect to ground truth. We also present another metric to quantify internal cluster quality (similarity of region members and dissimilarity between regions).

We designed our simulated dataset using a full-factorial design on the following simulation variables and associated levels.

In addition to variables above, the pattern and shape of regions is another experimental design variable. The mask in top-left corner of Fig. 6 is an example of a rectangular, stacked region. We create masks with non-rectangular regions (elliptic) that are either stacked or contained within another region. We performed a total of 5100 simulations with 20 realizations per variable combination. The only difference between two realizations in the same variable combination is the random number used to generate values per region stochastically.

Table 1: Quantitative Experiment Variables for Cluster Quality Tests

Variable	Low	Medium	High
$\Delta\mu(R)$	2	3	4
k	5	10	15
n	120	300	1200

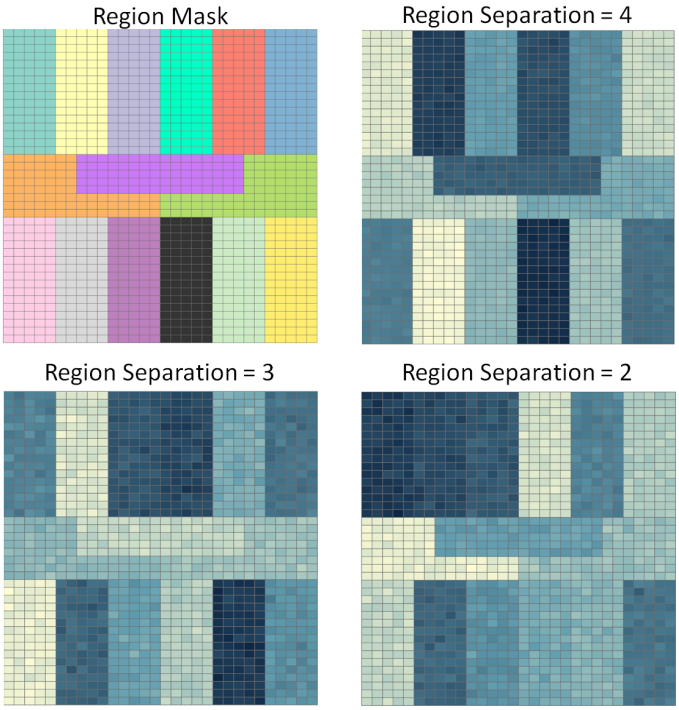

Figure 6: Region mask for k=15 (top left). Simulated maps with varying region separations.

5.2 Region Quality Metrics

We assess the quality of the methods based on their ability to retrieve inherent regions in spatial data and in terms of the resulting region homogeneity. We borrow metrics from the clustering literature and use the following to compare our proposed methodology to state-of-the-art methods:

(1) Normalized Mutual Information (NMI) [24]
(2) Calinski-Harabasz Index [7]

We use NMI to quantify the similarity between regions detected by the algorithms and the underlying regions in the data. In our quality metric analysis we stress NMI for cases where distinct regions exist (such as $\Delta\mu(R) = 4$). NMI is defined in Eq. 15.

$$NMI\left(R^{(1)}, R^{(2)}\right) = \frac{2I\left(R^{(1)}, R^{(2)}\right)}{H\left(R^{(1)}\right) + H\left(R^{(2)}\right)} \in [0, 1] \quad (15)$$

In Eq. 15 $R^{(1)}$ and $R^{(2)}$ are two regionalizations of the same dataset, $I\left(R^{(1)}, R^{(2)}\right)$ is the mutual information between $R^{(1)}$ and $R^{(2)}$ and $H\left(R^{(1)}\right)$ is the entropy for $R^{(1)}$.NMI of 1 implies regionalizations $R^{(1)}$ and $R^{(2)}$ are identical and a value of 0 is the largest dissimilarity between two regionalizations possible for the NMI metric.

Calinski-Harabasz (CH) index is used as an internal measure of regionalization quality. For cases with fuzziness CH is a useful metric to judge the quality of clustering results. CH index is defined as the ratio of between and within cluster deviation.

$$CH(R) = \frac{n\sum_{i=1}^{k} d(R_i, R)^2}{\sum_{i=1}^{k}\sum_{x\in R_i} d(x, R_i)^2} \quad (16)$$

CH index provides a compactness metric for regions because it incorporates within region deviation $d(x, R_i)^2$ and dissimilarity between regions $d(R_i, R)^2$, jointly.

5.3 Evaluation of Region Quality Metrics

We first evaluate our metrics for cases where regions are distinct and there is no fuzziness between regions. We present performance metrics as box plots to display average performance and the stability of the different algorithms. We first compare the results of SKATER-CON, SKATER and ARISEL using NMI.

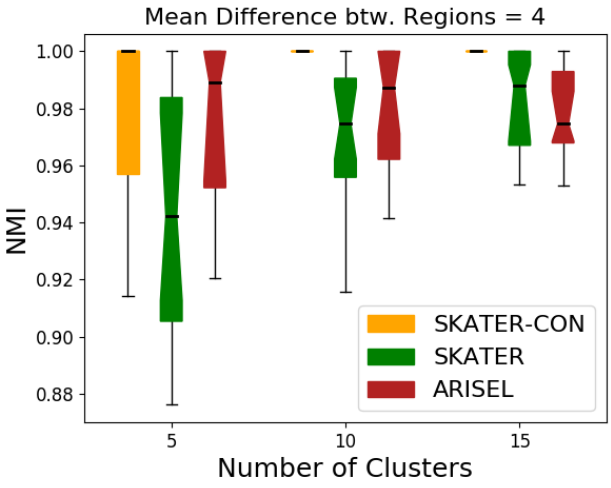

Figure 7: Normalized Mutual Information (NMI) calculated for simulations with varying number of underlying regions

Fig. 7 shows that SKATER-CON is outperforming both of the algorithms. For large number of regions ($k = 10, 15$), SKATER-CON retrieved underlying regions perfectly for all simulations. Fig. 7 is also indicative of SKATER's performance for small number of clusters. SKATER under-performs both methods for small number of underlying regions and outperforms ARISEL for large number of clusters.

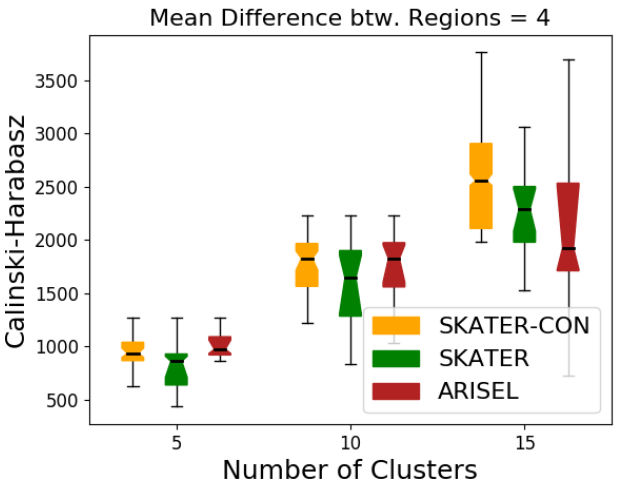

Figure 8: Calinski-Harabasz Index calculated for simulations with varying number of underlying regions

For the same set of simulations, we investigate the CH index. Fig. 8 shows that SKATER-CON finds compact regions when regions are distinct. Note that CH index shows a larger variation in returned regions for large number of groups. This is a triviality in the metric with respect to our data generating process because marginal distribution has a wider variance for large number of clusters. It is important to note that proposed method outperforms other algorithms for all k.

We evaluate the change in CH for regionalization results with respect to different fuzziness of the clusters. Note that in all of our experiments we return regionalization results for the underlying k used in the design of the region mask. Thus, for fuzzy cases low compactness is due to both the quality of the regionalization algorithm and/or using an inappropriate k for the fuzzy map. Generally for a fuzzy case such as $\Delta\mu(R) = 2$ actual clusters in the simulated map is less than k.

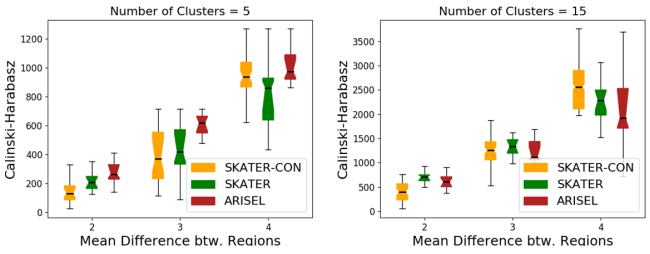

Figure 9: CH index for simulations with varying fuzziness for high and low number of underlying regions

Fig. 9 shows that CH index for SKATER-CON outperforms both SKATER and ARISEL when there are distinct regions. For high fuzziness, the CH indexes are low as k defined for all three algorithms

is not reflected in the original data. For instance, the bottom-right map in Fig. 9 shows 10 distinct regions although the underlying region mask has 15 regions.

6 CASE STUDY

We apply SKATER-CON to a real-world problem to highlight the capabilities of the methodology and the importance of defining compact regions. We analyze the publicly-available Ecological Marine Unit (EMU) dataset [36] using SKATER-CON to determine oceanic regions with respect to ocean temperature, salinity, dissolved oxygen, phosphate concentration, nitrate concentration and silicate concentration. Understanding oceanic regions is important for building predictive models locally rather than trying to fit models to global ocean data that contain numerous trends. The correlation matrix for aforementioned ocean measurements are in Fig. 10

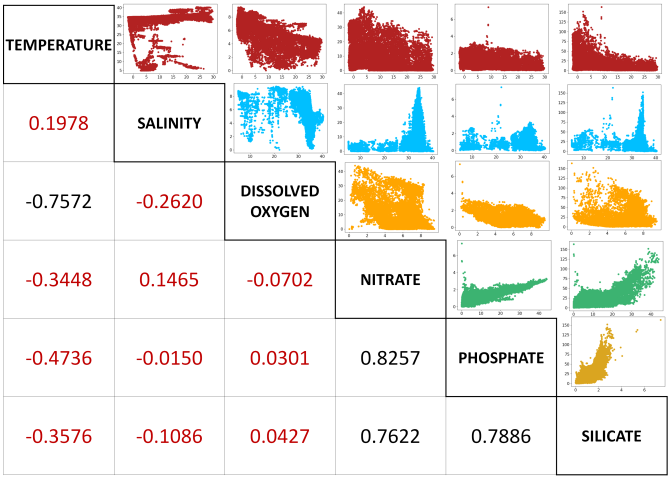

Figure 10: Correlation matrix for 6 different ocean variables

Fig. 10 illustrates the complexity of the relationships between different ocean variables at a global scale. Note that different trends between relationships of ocean variables are observed on scatterplots. For instance, temperature and salinity have two distinct relationships. One is a linear relationship for high ocean temperature and other is a non-linear relationship for low ocean temperature. While a complex predictive model can be fit to this relationship, distinct and simpler relationships in the data can be exposed through regionalization. Thus, a seemingly complex global problem can be subdivided into simpler local problems. We apply SKATER-CON to the EMU dataset and used the Calinski-Harabasz pseudo-F statistic [41] to determine the number of clusters.

In Fig. 11, distinct clusters are observed and the result of regionalization is mapped out for visualization. Some of the complex trends in the data is due to the difference in relationships between ocean variables in ocean coasts versus sea coasts. Due to the definition of the coordinate system, ocean measurements around the Ring of Fire is split into two (green and parts of brown). However, colder seas are detected as separate regions (yellow and brown), in addition East Coast of United States is regionalized with warmer Central

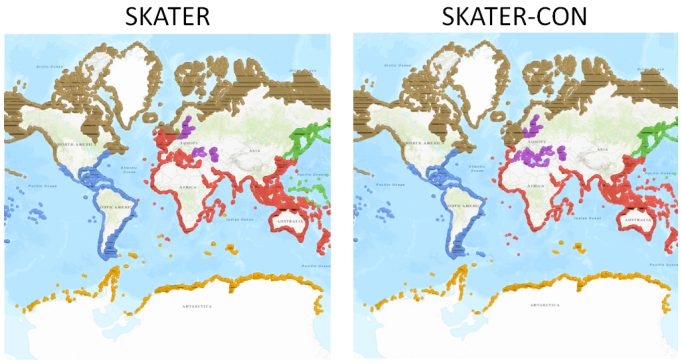

Figure 11: Oceanic regions defined for temperature, salinity, dissolved oxygen, nitrate, phosphate and silicate concentration jointly using SKATER (left) and SKATER-CON (right)

and South American coasts whereas West Coast is grouped in with colder seas. Lastly, temperate coasts are regionalized as the red region in Fig. 11.

We evaluate the performance of proposed method over SKATER with respect to Calinski-Harabasz index. The reason behind the choice of this metric is due to lack of true region labels for this problem.

Table 2: Quantitative Comparison of SKATER and SKATER-CON

Variable	SKATER	SKATER-CON	Improvement (%)
Joint	15269.2	13335.3	12.67
Salinity	36270.6	36840.8	-1.57
Temperature	22862.0	6563.6	71.29
Dissolved O_2	12242.5	14100.8	-15.18
Nitrate	9341.6	8485.0	9.17
Phosphate	6198.1	5295.4	14.56
Silicate	12971.7	11513.6	11.24

Table 2 shows that SKATER-CON provides more homogeneous groups compared to SKATER. Even though some variables are less homogeneous, overall clustering improvement with respect to Calinski-Harabasz is 12.7 %.

6.1 Value Added with SKATER-CON

Extracting meaningful oceanic regions is important in the context of modeling any system that relies on ocean variables. In this case, breaking down complex global relationships between ocean variables into simpler local ones allows scientists to build simpler, local models. One of the biggest difference in regions detected by SKATER and SKATER-CON are pertaining to sea coasts. Landlocked seas and seas that have access to oceans via straits are detected as a distinct group (purple in Fig. 11) by SKATER-CON. Note that SKATER-CON was able to resolve the Mediterranean Sea as a member of the purple region whereas SKATER lumped it in with a large region. In addition, UK is placed in the brown region by SKATER-CON rather than the red region suggested by

SKATER. SKATER result lumps UK together with the North African and Australian coasts.

7 CONCLUSIONS

In this paper, we defined a consensus-based regionalization algorithm to create spatially-contiguous and compact regions. Proposed method, SKATER-CON has outperformed state-of-the-art methods SKATER and ARISEL in terms of accuracy of the regions detected and overall compactness of regions. We solve the chaining problem of tree based graph partitioning by proposing many partitions using alternate trees defined by uniform spanning trees. We used an evidence accumulation framework to represent the ensemble of different regionalizations as a similarity matrix. We defined consensus regionalization as the partitioning of the similarity graph under spatial constraints. Proposed work allows imposing spatial contiguity relationships on regionalization votes. Empirical and synthetic studies show that SKATER-CON improves on pre-existing regionalization approaches. In our synthetic experiments, SKATER-CON returns regions that are more compact compared to state-of-the art methods. In addition, in terms of its performance, SKATER-CON is the most stable among other regionalization algorithms. For extremely fuzzy cases, the proposed method slightly underperformed. This is due to searching for the number of clusters in designed region-mask that is not reflected in simulation due to high level of fuzziness.

8 ACKNOWLEDGEMENTS

Authors acknowledge the ESRI Spatial Statistics team. In particular, Jenora D'Acosta for her feedback on the manuscript.

REFERENCES

[1] David Aldous. 1990. A random tree model associated with random graphs. *Random Structures & Algorithms* 1, 4 (1990), 383–402.
[2] Renato M Assunção, Marcos Corrêa Neves, Gilberto Câmara, and Corina da Costa Freitas. 2006. Efficient regionalization techniques for socio-economic geographical units using minimum spanning trees. *International Journal of Geographical Information Science* 20, 7 (2006), 797–811.
[3] Fernando Bacao, Victor Lobo, and Marco Painho. 2005. Applying genetic algorithms to zone design. *Soft Computing* 9, 5 (2005), 341–348.
[4] Sugato Basu, Ian Davidson, and Kiri Wagstaff. 2008. *Constrained clustering: Advances in algorithms, theory, and applications.* CRC Press.
[5] Itai Benjamini, Russell Lyons, Yuval Peres, and Oded Schramm. 2001. Special invited paper: uniform spanning forests. *Annals of probability* (2001), 1–65.
[6] Nina Bullen, Graham Moon, and Kelvyn Jones. 1996. Defining localities for health planning: a GIS approach. *Social Science & Medicine* 42, 6 (1996), 801–816.
[7] Tadeusz Caliński and Jerzy Harabasz. 1974. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods* 3, 1 (1974), 1–27.
[8] Thomas G Dietterich. 2000. Ensemble methods in machine learning. In *International workshop on multiple classifier systems.* Springer, 1–15.
[9] JC Duque and RL Church. 2004. A new heuristic model for designing analytical regions. In *North American Meeting of the International Regional Science Association, Seattle.*
[10] Juan Carlos Duque. 2004. *Design of homogenous territorial units. A methodological proposal and applications.* Universitat de Barcelona.
[11] Juan Carlos Duque, Raúl Ramos, and Jordi Suriñach. 2007. Supervised regionalization methods: A survey. *International Regional Science Review* 30, 3 (2007), 195–220.
[12] Anuška Ferligoj and Vladimir Batagelj. 1982. Clustering with relational constraint. *Psychometrika* 47, 4 (1982), 413–426.
[13] Ana LN Fred and Anil K Jain. 2002. Data clustering using evidence accumulation. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, Vol. 4. IEEE, 276–280.
[14] Ana LN Fred and Anil K Jain. 2005. Combining multiple clusterings using evidence accumulation. *IEEE transactions on pattern analysis and machine intelligence* 27, 6 (2005), 835–850.
[15] Stuart Geman and Donald Geman. 1987. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. In *Readings in Computer Vision.* Elsevier, 564–584.
[16] Arthur Getis. 2009. Spatial weights matrices. *Geographical Analysis* 41, 4 (2009), 404–410.
[17] Arthur Getis and Jared Aldstadt. 2010. Constructing the spatial weights matrix using a local statistic. In *Perspectives on spatial data analysis.* Springer, 147–163.
[18] Ronald L Graham and Pavol Hell. 1985. On the history of the minimum spanning tree problem. *Annals of the History of Computing* 7, 1 (1985), 43–57.
[19] Diansheng Guo. 2008. Regionalization with dynamically constrained agglomerative clustering and partitioning (REDCAP). *International Journal of Geographical Information Science* 22, 7 (2008), 801–823.
[20] Anil K Jain and Richard C Dubes. 1988. Algorithms for clustering data. (1988).
[21] George Karypis and Vipin Kumar. 1995. METIS–unstructured graph partitioning and sparse matrix ordering system, version 2.0. (1995).
[22] Gustav Kirchhoff. 1847. Ueber die Auflösung der Gleichungen, auf welche man bei der Untersuchung der linearen Vertheilung galvanischer Ströme geführt wird. *Annalen der Physik* 148, 12 (1847), 497–508.
[23] Vivek Kwatra, Arno Schödl, Irfan Essa, Greg Turk, and Aaron Bobick. 2003. Graphcut textures: image and video synthesis using graph cuts. In *ACM Transactions on Graphics (ToG)*, Vol. 22. ACM, 277–286.
[24] Andrea Lancichinetti, Santo Fortunato, and János Kertész. 2009. Detecting the overlapping and hierarchical community structure in complex networks. *New Journal of Physics* 11, 3 (2009), 033015.
[25] Russell Lyons. 1998. A birdâĂŹs-eye view of uniform spanning trees and forests. *Microsurveys in discrete probability* 41 (1998), 135–162.
[26] Russell Lyons and Yuval Peres. 2016. *Probability on trees and networks.* Vol. 42. Cambridge University Press.
[27] Maurizio Maravalle and Bruno Simeone. 1995. A spanning tree heuristic for regional clustering. *Communications in statistics-theory and methods* 24, 3 (1995), 625–639.
[28] CR Margules, DP Faith, and L Belbin. 1985. An adjacency constraint in agglomerative hierarchical classifications of geographic data. *Environment and Planning A* 17, 3 (1985), 397–412.
[29] Val L Mitchell. 1976. The regionalization of climate in the western United States. *Journal of Applied Meteorology* 15, 9 (1976), 920–927.
[30] MA Oliver and R Webster. 1989. A geostatistical basis for spatial weighting in multivariate classification. *Mathematical Geology* 21, 1 (1989), 15–35.
[31] Stan Openshaw. 1973. A regionalisation program for large data sets. *Computer Applications* 3, 4 (1973), 136–147.
[32] Stan Openshaw. 1977. A geographical solution to scale and aggregation problems in region-building, partitioning and spatial modelling. *Transactions of the institute of british geographers* (1977), 459–472.
[33] Stan Openshaw. 1995. Classifying and regionalizing census data. *Census users' handbook* (1995), 239–270.
[34] Stan Openshaw and Liang Rao. 1995. Algorithms for reengineering 1991 Census geography. *Environment and planning A* 27, 3 (1995), 425–446.
[35] Robin Pemantle. 2004. Uniform random spanning trees. *arXiv preprint math/0404099* (2004).
[36] Roger G Sayre, Dawn J Wright, Sean P Breyer, Kevin A Butler, Keith Van Graafeiland, Mark J Costello, Peter T Harris, Kathleen L Goodin, John M Guinotte, Zeenatul Basher, et al. 2017. A three-dimensional mapping of the ocean based on environmental data. *Oceanography* 30, 1 (2017), 90–103.
[37] Robert E Schapire. 1990. The strength of weak learnability. *Machine learning* 5, 2 (1990), 197–227.
[38] Nigel A Spence. 1968. A multifactor uniform regionalization of British counties on the basis of employment data for 1961. *Regional Studies* 2, 1 (1968), 87–104.
[39] Alexander Strehl and Joydeep Ghosh. 2002. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of machine learning research* 3, Dec (2002), 583–617.
[40] Leonardo Vilela Teixeira, Renato Martins Assuncao, and Rosangela Helena Loschi. 2015. A generative spatial clustering model for random data through spanning trees. In *Data Mining (ICDM), 2015 IEEE International Conference on.* IEEE, 997–1002.
[41] Robert Tibshirani, Guenther Walther, and Trevor Hastie. 2001. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 63, 2 (2001), 411–423.
[42] James B Weaver and Sidney W Hess. 1963. A Procedure for Nonpartisan Districting: Development of Computer Techiques. *Yale LJ* 73 (1963), 288.
[43] David Bruce Wilson. 1996. Generating random spanning trees more quickly than the cover time. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing.* ACM, 296–303.
[44] Feng Zhao, Licheng Jiao, Hanqiang Liu, and Xinbo Gao. 2011. A novel fuzzy clustering algorithm with non local adaptive spatial constraint for image segmentation. *Signal Processing* 91, 4 (2011), 988–999.