ABSTRACT

MCMAHON, MALLORY ELISE. Unsupervised Learning Models for Dual-Domain Data with Proximal Geographic Clustering. (Under the direction of Mansoor Haider.)

A dual-domain data set is comprised of two distinct sets of features, with one set consisting of geographic or spatial data and the other consisting of attribute data. This type of data is ubiquitous across a wide spectrum of disciplines, yet identifying meaningful clusters within dual-domain data sets is often non-trivial. Conventional clustering methods often rely exclusively on similarity in the geographic domain when interpreting clusters, even though the algorithm was likely trained on the entire data set. While this approach may be effective for identifying well-delineated geographic groupings, it does not account for instances where data observations that are similar in the attribute domain are geographically dispersed. Some recent works have proposed multi-pass approaches that determine similarity in each domain individually, then iteratively merge and tune the clusters to resolve inter-domain discrepancies. In this work, we developed a tailored unsupervised learning methodology based on the Self-Organizing Map (SOM) algorithm that is specifically designed to address the challenges of dual-domain clustering in an automated way. Our framework incorporates domain-specific biasing parameters into the SOM training that are subsequently optimized via an objective function.

The tailored learning methodology in this work, referred to as the Tailored SOM, incorporated the use of a 2-dimensional, 3×3 rectangular SOM structure that was trained on dual-domain data via a reformulated Euclidean distance measure. This reformulation considers and measures similarity using domain-specific biasing parameters, which allow for feature weighting and better evaluation of inter-domain data relationships. To optimize the values of these parameters, we constructed an objective function defined with quantities derived from the clustering projection in the geographic domain as well as statistical measures of the attribute clusters. The choice of these quantities was motivated directly by the overall goals of dual-domain clustering and designed to be sufficiently versatile across applications. We prescribed an admissible space for the biasing parameters and proposed two global methods for its traversal: a systematic grid search and the stochastic technique known as Simulated Annealing.

To test the performance and robustness of the tailored learning methodology, we generated several cases of synthetic dual-domain data with varying structure and complexity. To ensure consistent and methodical algorithm assessment, we designed three metrics: the *Stable Score*, the *Inlier Score*, and the notion of *Geographic Feasibility*. We evaluated these metrics, along with the tailored objective function, to assess and compare the results obtained by the Tailored SOM and two standard clustering algorithms, the Standard SOM and the k-Means algorithm. The evaluation of these metrics demonstrated that the results produced by the Tailored SOM were, by and large, the most geographically desirable and feasible. Additionally, this analysis provided us with insight into the relative tendencies, strengths, and limitations of the Tailored SOM, the Standard SOM and the k-Means algorithm when applied to dual-domain data sets. Finally, we tested the capability of the Tailored SOM on a dual-domain data set consisting of geographically-ordered colorectal cancer incidence rates in the state of California. With this data set, we further investigated the applicability of a versatile objective function, the advantage of a tailored objective function, and the robustness of the Tailored SOM and its associated learning methodology in the context of higher-dimensional data. Our results demonstrated the unique capability of the Tailored SOM to identify both a well-delineated geographic clustering and significant groupings of high and low cancer incidence. These findings are significant in that they could provide the public health and epidemiology community with a more holistic understanding of the relationships between geographic location, demographic features, and the prevalence of certain cancers.

Unsupervised Learning Models for Dual-Domain Data with Proximal Geographic Clustering

by
Mallory Elise McMahon

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Applied Mathematics

Raleigh, North Carolina

2020

APPROVED BY:

Kevin Flores	Arvind Saibaba
Ralph Smith	Mansoor Haider Chair of Advisory Committee

DEDICATION

For my parents, who have always believed in the beauty of my dreams.

BIOGRAPHY

Mallory was born and raised in Southampton, New York. As an undergraduate, she studied Mathematics at High Point University in High Point, North Carolina and Spanish at the Spanish American Institute of International Education in Sevilla, Spain. In 2014, she graduated *summa cum laude* with All-University Honors and had hopes of soon pursuing a Ph.D. In 2015, she moved to Raleigh, North Carolina and began her graduate studies in Applied Mathematics at North Carolina State University. Aside from math, Mallory loves traveling, cooking, quoting The Office, and spending sunny Saturday afternoons with her friends and family.

ACKNOWLEDGEMENTS

I would like to sincerely thank my advisor, Dr. Mansoor Haider, for his guidance, support, and friendship throughout my graduate studies. His mentorship, enthusiasm, and encouragement were integral to the successful completion of this work. I would also like to thank the members of my committee, Dr. Kevin Flores, Dr. Arvind Saibaba, and Dr. Ralph Smith for their feedback and support. Additionally, I would like to acknowledge that this work was supported in part by a Statistical and Applied Mathematical Sciences Institute (SAMSI) Graduate Fellowship, and the National Science Foundation (NSF) Award DMS-1639521.

I would like to lovingly thank my family for their unwavering encouragement, support, and dependability. For instilling in me the confidence to dream big, and the work ethic and drive to transform my goals into reality. I could never have done this without you.

I am forever indebted to my truly incredible friends (and fellow *Modelers Anon*), Michael Lavigne, Tricity Andrew, Jared Cook, and Sangeeta Warrier for being by my side every step of the way these last five years. For sharing your brilliance, for finding laughter and fun among all the failures, for teaching me everything I know about MATLAB, and for always, always believing in me.

Finally, I owe an enormous thank-you to my partner, Rob, for his unyielding love and patience, and his unmatched ability to make me smile. For always taking an interest in my work, seeing me through the stressful days, and keeping me well-fed. I am grateful for you each and every day.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER

1

INTRODUCTION

1.1 Motivation

In this thesis we consider data from a dual-domain configuration. The configuration consists of an attribute domain A and a geographic domain G. The attribute domain may contain information related to quantitative or measurable qualitative features. The geographic domain G provides location information for each observation in A. We allow the definition of geographic to be flexible, thus G could provide data pertaining to physical locations, or "digital" locations, i.e., relative closeness on a social network map or similarity in web-based activity, as described in [12]. Each dual-domain data observation d_i is a concatenation of all features from both domains, i.e., $d_i = [a_i, g_i]$ where $a_i \in A$, $g_i \in G$. An example of this dual-domain configuration, projected into the geographic domain G is illustrated in Figure 1.1. In this example, each point is an observation from G, in the form of an (x, y) location pair, with the marker type indicating its true group in the attribute domain G. This example includes a geographic constraint in the form of a river running diagonally through the data.

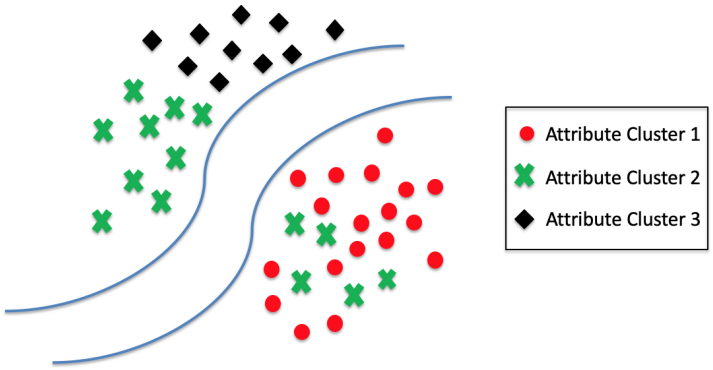


Figure 1.1 An example of a dual-domain data set, projected into the geographic domain G, featuring a geographic constraint in the form of a river. In this example, there are geographic data observations located on each side of the river that correspond to one of three attribute groups (groupings in the attribute domain A). We use the marker type/color (red circle, green x, black diamond) to indicate each geographic data observation's association with an attribute group.

We aim to find a clustering of a data set with observations of the form $d = [a, g]$ that is both telling of the intrinsic structure of the attribute domain A and is geographically useful. In other words, the goal is to identify clusters of "like-minded" individuals (data observations) that are geographically proximal. We loosely define a geographically useful cluster as one whose boundaries do not span across geographic barriers or constraints and whose coverage area has minimal overlap with other clusters. We view this problem as an unsupervised learning task, where neither the observations from the attribute domain A nor the geographic domain G are labeled, therefore we cannot train against a ground truth. Previous approaches to similar dual-domain (joint-domain) clustering problems have been more heuristic in nature, requiring multiple phases and explicit reliance on geographic constraints [24, 25, 27]. We, instead, view this as an optimization problem by introducing an objective function directly motivated by the modeling application.

1.2 Definition of a Successful Clustering

Unsupervised data clustering is an inherently uncertain task. That is to say, we often find ourselves relying on clustering algorithms to uncover structure and similarity within a data set without a widely accepted method for assessing the validity of the results. It would not be uncommon for two well-studied clustering algorithms to identify vastly different clusters within the same data set, with the likelihood of this outcome increasing dramatically as the dimension and complexity of the data

set grows. By definition, the observations of a data set undergoing a clustering regime are not labeled nor classified a priori, i.e., there is neither an underlying ground truth nor “right answer" to train the algorithm against [13]. With this type of task, we as researchers are responsible for designing a notion of correctness based on either domain-specific knowledge, measurable qualities of the data set itself, or desired cluster outcomes [1, 4, 44]. In the context of the dual-domain problem, we aim to uncover a clustering that is both telling of the intrinsic structure of the attribute domain A and is feasible in the geographic domain G. It is with this overarching goal in mind that we develop a definition for correctness and construct the desired properties of the ideal clustering.

The reliance on synthetic data sets when building and testing our dual-domain clustering approaches is especially advantageous when it comes to defining the desired properties of a right answer. With synthetic data comes knowledge of underlying generating distributions, various levels of noise within the data, and intended cluster arrangements in A and G. Based on this knowledge, we can test the performance of an algorithm with scoring methods that consider quantities such as:

- the number of known primary groups identified,
- the number of outlying/anomalous observations assigned to each geographic cluster, and
- the degree to which geographic constraints present in G (such as the river depicted in Figure 4.3 (b)) were adhered to, i.e., not violated when geographic cluster boundaries are imposed.

In contrast to working with real data, we can reliably use these type of metrics when assessing the clustering of synthetic data due to the level of control we have over its design. This control is rooted in how we choose to prescribe values for the many parameters that are used in the synthetic data set design process and allows for the creation of an assessment framework. Within this framework, we can investigate how varying levels of noise and cluster separation affect the accuracy and performance of our methods.

1.3 A Review of Literature on Dual-Domain Clustering Methods

Traditional clustering methods may approach a dual-domain problem by either: 1) only clustering over the data in the geographic domain G, thus overlooking any groupings that may exist in the

attribute domain and focusing exclusively on geographic proximity, or 2) using a standard algorithm to cluster over the entire data set, weighting the features from A and G equally when determining similarity. Figure 1.2 (a,b) demonstrates possible outcomes of implementing methods 1) and 2), respectively, on the sample dual-domain data set depicted in Figure 1.1. While the outcome presented in Figure 1.2 (a) is mindful of geographic proximity, it is blind to attribute differences coexisting within its clusters. The outcome presented in Figure 1.2 (b) is clearly problematic as it does not account for the geographic constraint in the data set and instead favors attribute similarity, thus producing a geographically unfeasible clustering.

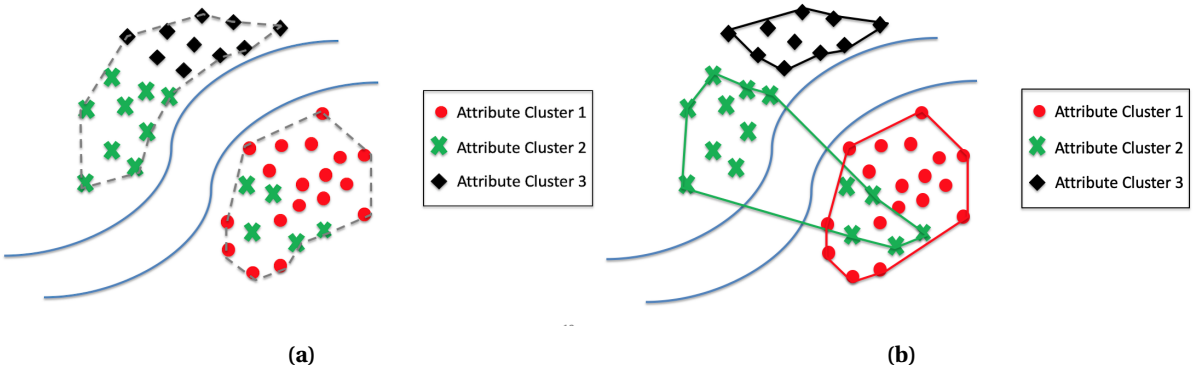


Figure 1.2 A depiction of possible geographic clusterings obtained via standard clustering algorithms that: (a) only consider the data in the geographic domain (G) during training, and (b) consider and weight all features in the data set equally (in both domains G, A) during training.

Recent works [24, 25, 27] have proposed iterative, multi-pass methodologies to efficiently cluster dual-domain data (also referred to as joint clustering). Such methods often require that the clustering algorithms have explicit knowledge of constraints present in the geographic domain prior to training, in order to produce a result that is geographically feasible (i.e., it does not violate said constraints). In [25], Lin et al. proposes the Interlaced Clustering-Classification (ICC) algorithm to cluster dual-domain data using an attribute-based objective function and specified constraints in the geographic domain. This methodology breaks the dual-domain problem into two separate tasks: (1) an unsupervised clustering problem over the optimization (attribute) domain and, (2) a supervised classification problem over the geographic domain based on constraint specifications. The ICC algorithm alternates between these two tasks to identify optimal clusters in the optimization

(attribute) domain that correspond to compact geographic regions. Another multi-step approach for clustering dual-domain data is proposed by Liao and Peng in [24]. Their method, called the Clustering with Local Search (CLS) algorithm, proposes the construction of a connected graph, based on given geographic constraints and requirements, which is subsequently used to determine the placement of initial cluster representatives. From there, distances to nearby, connected data observations are computed and “coarse" clusters are formed and then iteratively merged and tuned [24]. In [27], Lo and Peng build upon the clustering methodology presented in [24] with their proposal of the k-Means with Local Search (KLS) algorithm. The KLS algorithm consists of three phases which aim to (1) build a connected graph for representing constraints in the geographic domain, (2) apply a coarse clustering to the data via an initial placement of k or more cluster centroids, and (3) tune and merge the clusters, according to a cost function, until exactly k clusters are identified.

A few commonalities among the existing dual-domain clustering methods outlined above are:

- the use of multiple clustering and classification phases which focus exclusively on analyzing the data in one of the two domains,
- the requirement of explicit knowledge of geographic constraints or definition for geographically connected, and
- the specification of the number of desired clusters.

1.4 Tailored Clustering using Self-Organizing Maps

In this thesis, we propose a tailored unsupervised learning methodology to efficiently cluster dual-domain data sets that does not require explicit incorporation of the geographic constraints or a priori specification of the number of desired clusters. Our algorithm aims to simultaneously cluster data in both the geographic and attribute domain through the use of domain-specific biasing parameters, the optimization of a tailored objective function, and the evaluation of generic geometric and statistical quantities. In Chapter 2, we present an overview of unsupervised machine learning and a detailed description of two standard clustering approaches: the Self-Organizing Map [17, 18] and

the k-Means algorithm [30]. In Chapter 3, we propose and develop a novel tailored unsupervised learning algorithm called the Tailored SOM, which relies fundamentally on the Self-Organizing Map, incorporates a reformulated distance measure, and employs a tailored objective function. Alongside the development of this algorithm, we motivate the need for the tailored objective function, detail the process through which we design its terms, and outline two methods for optimizing this function within the clustering framework.

After fully developing the Tailored SOM, we build and consider three cases of synthetic dual-domain data with varying structural characteristics and levels of noise that were specifically designed to be useful dual-domain test cases. With each test case of synthetic data, we implement the tailored unsupervised learning methodology presented in Chapter 3 and evaluate and compare its performance against results obtained by the standard clustering algorithms presented in Chapter 2. Furthermore, we propose and develop four novel metrics for evaluating and comparing algorithm performance. We create two metrics that are based on properties of the synthetic-data set itself, a third metric that accounts for geographic feasibility, and a fourth metric that quantifies geographic desirability of a clustering. These metrics give structure and formality to our algorithm assessment framework. These metrics, as novel contributions, are critical in the context of unsupervised learning, where there is neither a notion of "ground truth" nor a widely-accepted method for assessing the validity of an algorithm's output.

Finally, in Chapter 5, we consider a real dual-domain data set consisting of latitude/longitude measurements, colorectal cancer incidence rates, and demographic information for counties in the state of California. This data was obtained from the United State Census Bureau [43] and the CDC's National Program of Cancer Registries (NPCR) and the National Cancer Institute's (NCI's) Surveillance, Epidemiology, and End Results (SEER) Program [37]. We use the cancer incidence data to further investigate the capabilities of the tailored unsupervised learning methodology developed in Chapters 3 and 4. With this investigation, we demonstrate the unique capability of our Tailored SOM and its adaptable learning framework to discover clusters that are geographically useful, information-dense in the attribute domain, and capable of providing critical insight to the areas of public health and epidemiology.

CHAPTER

2

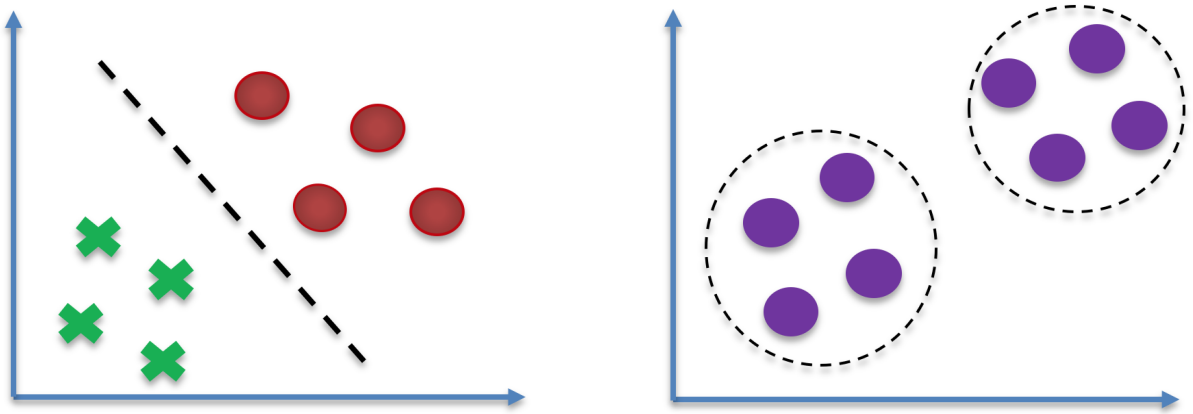
UNSUPERVISED LEARNING ALGORITHMS

2.1 Unsupervised Learning

Unsupervised learning refers to the class of machine learning techniques used for analyzing unlabeled data, i.e., a data set for which a known class or response for each observation is not provided. Unsupervised learning algorithms aim to identify patterns, groups, and similarities within a data set to ultimately cluster observations that have something in common [1, 32]. In the absence of explicit information regarding whether an input observation belongs to one class or another, unsupervised learning algorithms such as the Self-Organizing Map [18] and k-Means [30] rely on metrics such as distances between observations and the means of groups of observations to identify similarity and groupings in an automated fashion. In contrast, supervised learning techniques, such as regression, neural networks, and random forests, aim to learn the input-output relationship of labeled data for the purpose of building a prediction or classification model. Supervised learning algorithms

typically rely on an optimization regime to minimize an error function that uses the classes (outputs) and the data observations (inputs) to learn the relationship between the two. Once the algorithm has been trained and the input-output relationship has been learned, the supervised learning model can be used to predict the output of future observations or classify unlabeled observations [1].

Figure 2.1 depicts a fundamental difference between supervised and unsupervised learning. In an elementary supervised learning problem, as seen in Figure 2.1 (a), we have a standard two-class classification task. The two different classes of data are depicted by the red circles and green x's. For this example, a supervised learning algorithm, such as the traditional Support Vector Machine [7], would rely on the knowledge of the two labeled classes to learn, for instance, a linear separator between them. Then, when new data is presented, the algorithm would use the learned separator to classify the observations as belonging to one class or the other. This type of classification problem can also be extended to multi-class (more than 2 classes) data sets. In an elementary unsupervised learning task, as seen in Figure 2.1 (b), we are unaware of classes/labels, thus all data points are depicted as purple circles with no distinction among them. For this example, an algorithm, such as k-Means, would search for closeness among the data and decide on a clustering based on proximity. Depending on the application, new data can either be compared to the existing clusters and assigned to the cluster that it is most similar to, or a re-clustering of the entire data set may be appropriate.



(a) **(b)**

Figure 2.1 A depiction of a data set suited for: (a) an elementary supervised classification problem, and (b) an unsupervised clustering algorithm. The marker types/colors in (a) denote the *known* labels of each data observation, which are used to train a classifier. In contrast, observations in (b) have the same marker type/color, thus demonstrating the lack of available data labels in an unsupervised clustering problem.

2.2 Self-Organizing Map

The standard Self-Organizing Map (SOM) is a type of Artificial Neural Network developed by Teuvo Kohonen in the 1980s and was originally motivated by the human brain's spatial organization and how sensory signals are mapped into the cerebral cortex [17, 18]. The SOM functions similarly to a sensory map, which details how various parts of the brain respond to spatially organized sensory stimulation. The SOM learns an ordered, topology-preserving mapping from the input data to a low-dimensional (typically 1- or 2-dimensional) map structure via a similarity-based competition. This competitive learning algorithm uses a fully connected, low-dimensional map of nodes, depicted in Figure 2.2, to cluster and produce a representation of the higher-dimensional input data. Each node in the map is associated with a weight vector of dimension equal to that of the input data, and with a set of neighboring nodes.

During the SOM training, weight vectors compete against each other to be chosen as the "winner" based on the distance between each weight vector and the given data observation. Each winning weight vector, and the weight vectors of its node's neighbors, are then moved according to their proximity to data observations deemed closest. The goal of this iterative, competitive process is for each weight vector to become a "representative" for a cluster in the data. Additionally, the data ultimately clustered to nodes that are neighbors on the SOM will be more similar than data clustered to non-neighboring nodes. This property of neighboring nodes is unique to the SOM and results in a representation of clusters that preserves the spatial organization of the original input data space on the lower dimensional Self-Organizing Map.

Figure 2.2 depicts a two-dimensional, 3 x 3 rectangular lattice SOM consisting of nine nodes, three sample input data observations, and select weight vectors. The edges of the SOM are used to illustrate the neighborhood connections between nodes. Thus, any two nodes that are connected with an edge are considered neighbors and will be treated accordingly during the competitive learning. The weight vectors, a few of which are shown as dashed lines in Figure 2.2, are used to connect the two-dimensional map structure to the input data, shown as x's.

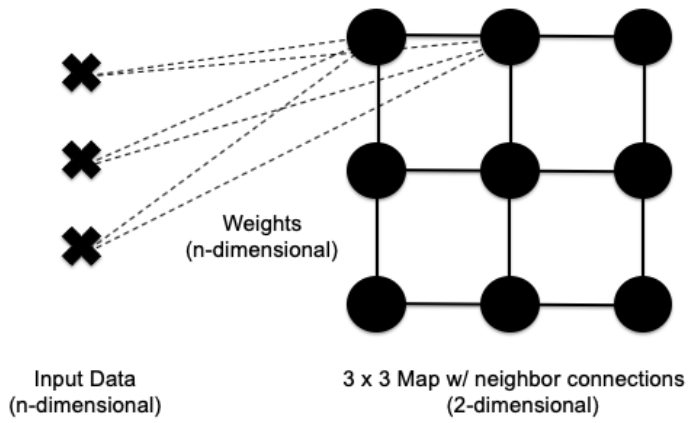


Figure 2.2 Illustration of a 3 x 3 rectangular lattice SOM with select weight vectors.

A step by step outline of the standard Self-Organizing Map algorithm [19] is given in Algorithm 1 . The key components of the algorithm are the total number of nodes (N) of the SOM, the arrangement of the nodes, and the learning rates z and . The learning rates dictate to what degree the position of the weight vectors are updated at each iteration of training. These rates decrease with time, thus the effects of these learning rates on the weight vectors' positions will gradually decline as the learning progresses. Exponential decay functions of the form

$$z(k) = a^{\frac{k}{k_{max}}} \tag{2.1}$$

and

$$(k) = b^{\frac{k}{k_{max}}} \tag{2.2}$$

where k is the iteration, k_{max} is the maximum number of iterations, $a \in (0,1)$, $b \in (0,1)$, and $a > b$, are typical choices for these learning rates.

Algorithm 1: Standard Self-Organizing Map

Choose topological arrangement, size (N), and neighboring relations of SOM;
Initialize learning rates $z(k)$ and (k);
Initialize weight vectors w_j, for $j = 1,..,N$ for each node of the SOM;
for $k = 1,..,k_{max}$ *(the number of batch training iterations)*
  for $i = 1,..,D$ *(each data vector in the data set* $\mathbf{S}$ *given by* d_i*)*
    Compute the index of the winning weight vector via the formulation:
    $c(i) = \text{argmin}_{j=1,..,N} \|d_i - w_j\|_2^2$.;
  end
  for $j = 1,..,N$
    $\mathbf{C}_j = \mathbf{S}(c == j,:)$ (a matrix of all data vectors with winning index j);
    Let $m_j = |\mathbf{C}_j|$ (the number of data vectors stored in C_j);
    Let N_j be the set of indices of the neighbors of w_j;
    Let $p_j = |N_j|$ (the number of neighbors associated with w_j) ;
  end
  for $j = 1,..,N$
    Let $w^* = w_j$;
    Update the j^{th} weight vector using $\mathbf{C}_j$;
    for $t = 1,..,m_j$
      $w_t = w^* + z(k)(\mathbf{C}_j(t,:) - w^*)$;
    end
    $w_j = \frac{1}{m_j} \sum_{t=1}^{m_j} w_t$;
    Update the j^{th} weight vector using the data associated with its neighbors;
    for $r \in N_j$
      for $s = 1,..,p_j$
        $w_s = w^* + (k)(\mathbf{C}_r(s,:) - w^*)$;
      end
      $w_j = w_j + \frac{1}{p_j} \sum_{s=1}^{p_j} w_s$;
    end
  end
  Update the learning rates $z(k)$ and (k);
end
for $i = 1,..,D$ *(each data vector in the data set* $\mathbf{S}$ *given by* d_i*)*
  Compute final winning cluster index $c(i) = \text{argmin}_{j=1,..,N} \|d_i - w_j\|_2^2$.;
end
return Final cluster indices c for $\mathbf{S}$ and final representative weight vectors w_j, for $j = 1,..,N$;

2.2.1 SOM Parameters

There are a number of parameters and algorithmic options that need to be prescribed when implementing a Standard SOM. Starting with line 1 of the algorithm, shown in Algorithm 1, we choose the total size N, the node arrangement (e.g., lattice, hexagonal), and the definition of a neighborhood. With a 2-dimensional rectangular lattice structure, depicted in Figure 2.2, the size N is given by $N = a \times b$, where $a = b = 3$. In this example, the edges of the rectangular lattice demonstrate which nodes in the map are defined as neighbors. Alternatively, a triangular lattice could be used, as shown in Figure 2.3. With a triangular lattice, the notion of neighbors is extended to include nodes that are adjacent diagonally, as well as those defined in the rectangular lattice.

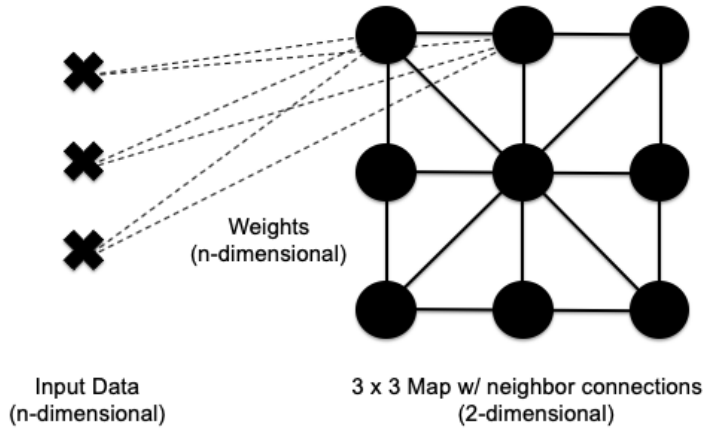


Figure 2.3 Illustration of a 3 x 3 triangular lattice SOM with select weight vectors.

In line 3 of Algorithm 1, we initialize the weight vector associated with each node in the SOM. There are several ways that weight vectors can be initialized, however, their dimension must always match that of the input data. We will further explore the various choices for how weight vectors can be initialized in Chapters 3 and 4 of this thesis. In line 6 of Algorithm 1, the competitive learning process begins with the computation of the distance between each weight vector and all observations in the input data. While this representation of the Standard Self-Organizing Map algorithm makes use of the Euclidean distance, it can be generalized further to make use of a different distance metric, when appropriate. Additionally, it is important to note that we implement a batch learning approach when training the SOM. Batch learning requires that the steps in which we update each weight vector and

its neighbors (lines 14 through 28) occur after winning weight vectors have been computed for the entire data set. This approach is in contrast to the online learning approach, which executes the weight vector updates immediately after each observation of the input data is presented. The batch competitive learning process is repeated until the user-prescribed maximum number of iterations, $k_{\max}$, is reached. Alternatively, the algorithm can terminate when changes to the weight vectors are within a prescribed tolerance.

2.3 k-Means

The k-Means algorithm [30] is arguably the most commonly used method for unsupervised data clustering. The algorithm aims to cluster D data observations into k clusters, based on each observation's proximity to the k cluster centroids. After k centroids are initialized, the algorithm proceeds by iterating back and forth between two steps: the *assignment step* and the *update step*. In the *assignment step*, each observation is assigned to the cluster with the nearest centroid, measured by Euclidean distance. Next, in the *update step*, each centroid is updated to equal the mean of the observations assigned to it. The algorithm terminates when there are no further changes made in the *assignment step*, or when a user-prescribed maximum number of iterations has been reached. A step by step outline of the standard k-Means algorithm [32], where termination depends on a user-prescribed maximum number of training iterations, is given in Algorithm 2.

Figure 2.4 illustrates the first few steps of the k-Means algorithm on a small, two-dimensional data set. The green x's are the data, and the blue and red circles are the cluster centroids. This illustration begins with a random initialization of the two cluster centroids (seen in (a)). Next, the first *assignment step* is executed (seen in (b)), thus each observation of data is assigned to the nearest centroid according to Euclidean distance. In this step, we update the color of each data observation to indicate which cluster (red or blue) it has been assigned to. Next, the first *update step* is executed (seen in (c)). In this step the two centroids' locations are updated to equal the mean of the data that was assigned to it in the previous step. This step is analogous to those performed in lines 14 through 20 in the SOM algorithm (Algorithm 1) as it allows for the centroids to become better representatives of the clusters that exist within the data. It is important to note that the k-Means

Algorithm 2: k-Means Algorithm

1 Choose value of k;
2 Initialize cluster centroids $\vec{c}_j$, for $j = 1,..,k$;
3 **for** $t = 1,..,T_{max}$
4 **for** $i = 1,..,D$ *(each data vector in the data set* $\mathbf{S}$ *given by* $\vec{d}_i$*)*
5 Compute the index of the closest cluster centroid via: $b(i) = \text{argmin}_j \|\vec{d}_i - \vec{c}_j\|_2$;
6 **end**
7 **for** $j = 1,..,k$
8 $\mathbf{B}_j = \mathbf{S}(\vec{b} == j, :)$ (a matrix of all data vectors closest to cluster centroid j);
9 Let $m_j = |\mathbf{B}_j|$ (the number of data vectors stored in B_j);
10 Update cluster centroids: $\vec{c}_j = \frac{1}{m_j} \sum_{r=1}^{m_j} \mathbf{B}_j(r, :)$;
11 **end**
12 **end**
13 **return** Final cluster indices $\vec{b}$ for $\mathbf{S}$ and final cluster centroids $\vec{c}_j$, for $j = 1,..,k$;

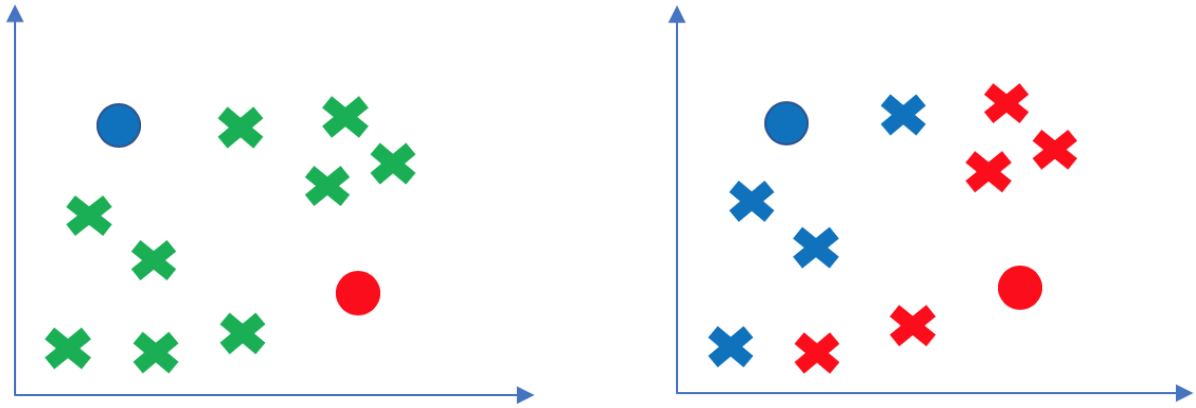
algorithm does not have an analog to the neighborhood update steps in the SOM algorithm (lines 22 through 27 in Algorithm 1) . The centroids in k-Means are neither aware of nor impacted by the location and training of each other, which is not the case in the SOM due to the implementation of a neighborhood among the nodes. Lastly, we illustrate the beginning of the next iteration of training, which begins again with an *assignment step* (seen in (d)). In this step, the data observations are compared to the locations of the updated centroids and reassigned to a cluster (red or blue) based on their Euclidean proximity. Even by only considering this small example, we can recognize two important features of the k-Means algorithm:

- the centroid initialization is highly influential in terms of how well the algorithm can perform and how quickly it terminates, and
- the choice of k, which is prescribed by the user, is nontrivial.

To address the first bullet, the k-Means algorithm is not alone in its sensitivity to initialization. The performance of the SOM algorithm will also vary, in some cases significantly, depending on the choice of initial weight vectors. In Chapter 3 of this thesis we will explore and experiment with various methods for SOM weight vector initialization. For the k-Means initialization, a widely used method is the $k++$ algorithm [5]. The $k++$ algorithm is a process that chooses the cluster centroids

iteratively, based on the distances between the input data and all previously chosen centroids. By considering the distances between a proposed centroid and the other centroids, the $k++$ algorithm creates an initialization that is sufficiently spread out in the input data space and less likely to lead to a poor clustering.

To address the second bullet, the choice of k in k-Means is somewhat analogous to the choice of the number of nodes (N) and their arrangement in the SOM algorithm. However, due to the lack of a neighborhood relation or connectivity among the centroids in k-Means, the algorithm will always produce a clustering with exactly k clusters. Whereas in the SOM, the nodes in the map are connected via a neighborhood relation. Thus the position of a node is affected by both the data assigned to it, and the data assigned to its neighbors. This feature allows for the algorithm to produce a clustering with less than N clusters. In other words, it is not always the case that each node in the SOM will be populated with data observations by the time the algorithm terminates.



(a) Initialization of 2 centroids

(b) First *assignment step*

(c) First *update step*

(d) Second *assignment step*

Figure 2.4 Illustration of implementing the first few steps of the k-Means algorithm on a two-dimensional data set.

2.3.1 Determining the Optimal k

When considering a two-dimensional data set for clustering, the natural first step would be to generate a scatter plot of the data in order to get a sense of how many clusters we expect to observe. However, as the dimension of the data set increases, visualization quickly becomes unfeasible. In the case of k-Means, it is the responsibility of the user to prescribe a value for k, i.e., we must first tell the algorithm how many clusters to look for. Without the ability to visualize the data, estimating k becomes a nontrivial, yet extremely important, task. To aid in this task, several methods have been developed. These methods rely on properties of the data set itself and a prescribed k to quantify the quality of a given clustering. Two of the most widely used methods are called the Elbow Method [31] and the Average Silhouette Method [39].

For the Elbow Method, we begin by choosing a range of possible integer values for k, for example $k = [2:10]$. For each choice of k, we randomly initialize k centroids, cluster the data, and compute the total within-cluster sum of squared distances (WSS) between the data observations and their cluster centroid. The WSS is defined as

$$\text{WSS} = \sum_{k=1}^{K} \sum_{i \in D_k} ||x_i - c_k||_2^2 \tag{2.3}$$

where D_k is the set of data assigned to the k-th cluster, x_i is the i-th data observation in D_k, and c_k is the k-th cluster centroid. As the value of k increases, the sum of squared distances will approach zero. However, this does not imply that a large k is inherently best. Consider the case where we let k equal the size of the data set, D. In this case, with careful centroid initialization, we expect the sum of squared distances to equal zero, since each data observation will form its own cluster. Clearly, this is not optimal nor useful. Instead, we plot the WSS as a function of k and look for the "elbow" in the graph to indicate an appropriate number of clusters. The "elbow" is defined as the point on the graph at which the WSS, as a function of k, begins to decrease linearly.

Similar to the Elbow Method, we begin the Average Silhouette Method by selecting a range of possible integer values for k. For each k, we randomly initialize k centroids, cluster the data, and compute the silhouette coefficient for each data observation. The silhouette coefficient S_i for an

observation x_i is defined as

$$S_i = \frac{b_i - a_i}{\max(a_i, b_i)}, \text{ for } i = 1, .., D \tag{2.4}$$

where a_i is the mean distance from x_i to the other observations in the same cluster, and b_i is the minimum mean distance from x_i to observations in a different cluster. The silhouette coefficient is designed to measure the similarity of points assigned to the same cluster, and the average of the silhouette coefficients across a clustered data set is used as the metric for assessing the quality of the clustering overall [39]. The range of S_i is $[-1, 1]$, with higher values indicating that a data observation is similar to other observations within its own cluster and dissimilar to observations in other clusters. While the silhouette coefficient can be computed using any distance metric, we implement the Euclidean distance throughout. We compare the average silhouette coefficients as a function of the number of clusters k and choose the optimal k to be the one that produces the largest average silhouette coefficient.

As a first illustrative example, we consider the two-dimensional data set depicted in Figure 2.5. The data set depicted in Figure 2.5 is generated from four known clusters. The observations in each cluster are generated from a circular region about a prescribed centroid, with radii drawn from the normal distribution $r \sim N(0, 0.2^2)$ and angular measure drawn from the uniform distribution $\sim U(0, 2\ )$. We apply both the Elbow Method and Average Silhouette Method to determine the optimal k for this data set. Based on the results depicted in Figure 2.6, we can determine that the "elbow" of the graph occurs at $k = 4$, since the total within-cluster sum of squared distances (WSS) appears to decrease linearly for $k > 4$. Similarly, the results from the Average Silhouette Method, depicted in Figure 2.7, suggest that $k = 4$ is optimal, since it corresponds with the maximum average silhouette coefficient. In this case, the two methods agree in their determination of the optimal k.

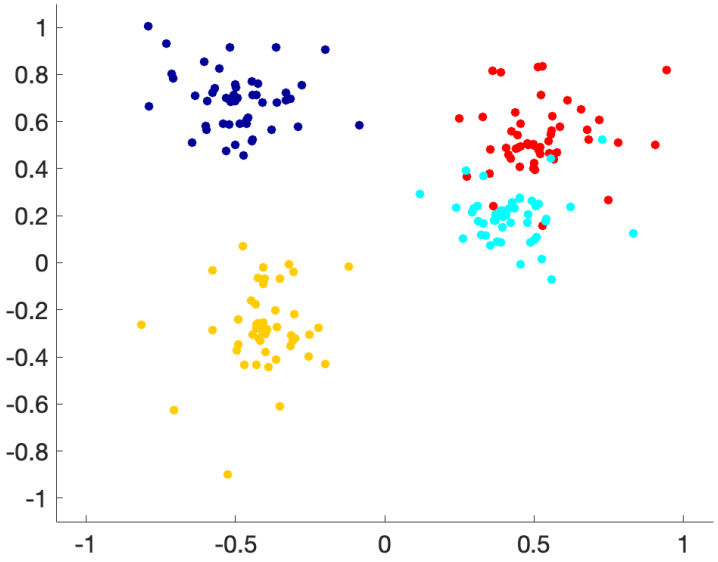


Figure 2.5 Two-dimensional data set generated from 4 cluster distributions with radial variance $= 0.2^2$.

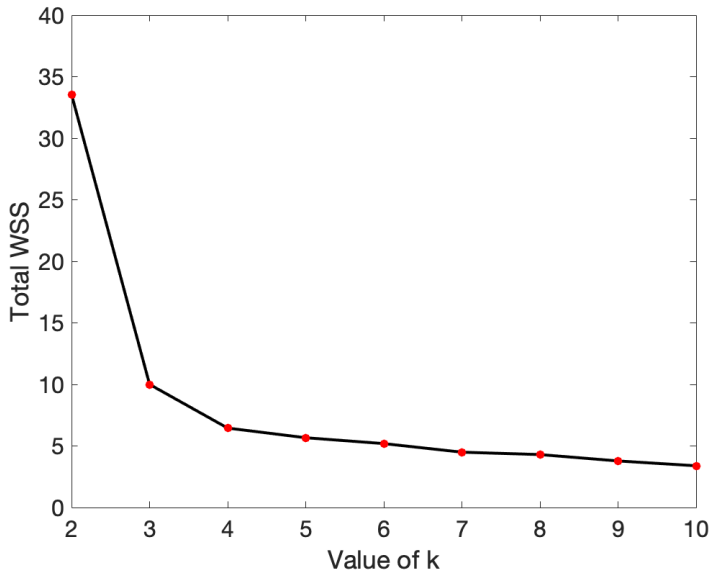


Figure 2.6 Elbow Method results for data depicted in Figure 2.5: total WSS as a function of k.

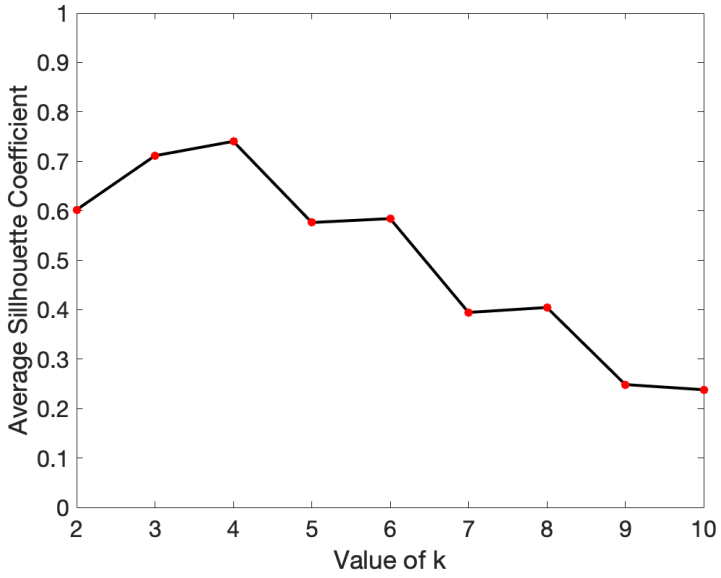


Figure 2.7 Average Silhouette Method results for data depicted in Figure 2.5: average S_i as a function of k.

As a second illustrative example, we consider the two-dimensional data set depicted in Figure 2.8. This data set was generated from the same four cluster centroids as the data set depicted in Figure 2.5, however the observations' radii were drawn from $N \sim (0, 0.25^2)$, i.e., the radial variance is 0.25^2 and larger than the previous example. We again implement the Elbow Method and the Average Silhouette Method to determine the optimal k. However, for this data set, the results of these methods are not as easily interpretable. The graph depicted in Figure 2.9 shows the results of implementing the Elbow Method. In this plot, it is not inherently clear where the "elbow" occurs. One could argue that possible contenders for the "elbow" may be $k = 3$, $k = 4$, or $k = 5$, given the decreasing nature of the graph at those points. Figure 2.10 depicts the results of implementing the Average Silhouette Method for the data set seen in Figure 2.8. Based on these results, we identify $k = 3$ as the optimal value as it yields the maximum average silhouette coefficient. However, it is important to note that $k = 4$ is a very close runner-up and may also be worth investigating further.

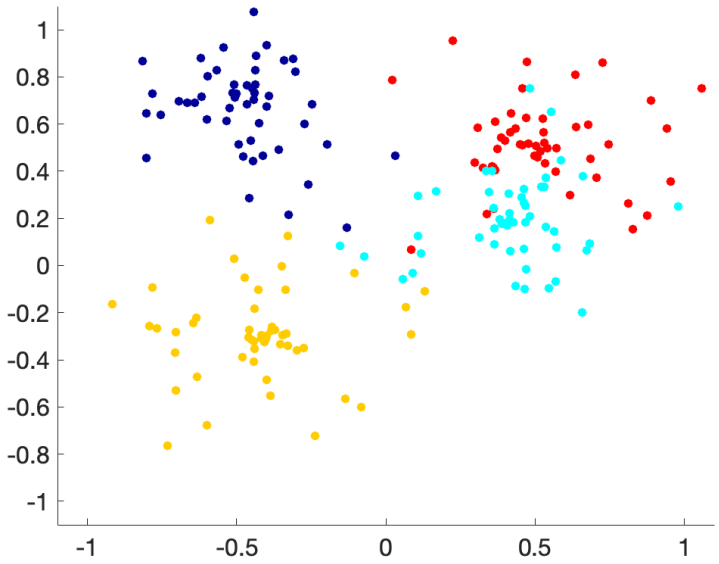


Figure 2.8 Two-dimensional data set generated from 4 cluster distributions with radial variance $= 0.25^2$.

In summary, these two illustrative examples further highlight one critical, yet nontrivial, component of implementing the k-Means algorithm: determining the best choice of k for a given data set. Even with a two-dimensional data set that can be visualized, the "optimal" k may not be obvious. The difficulty of this task compounds quickly as the dimension of the data set increases,

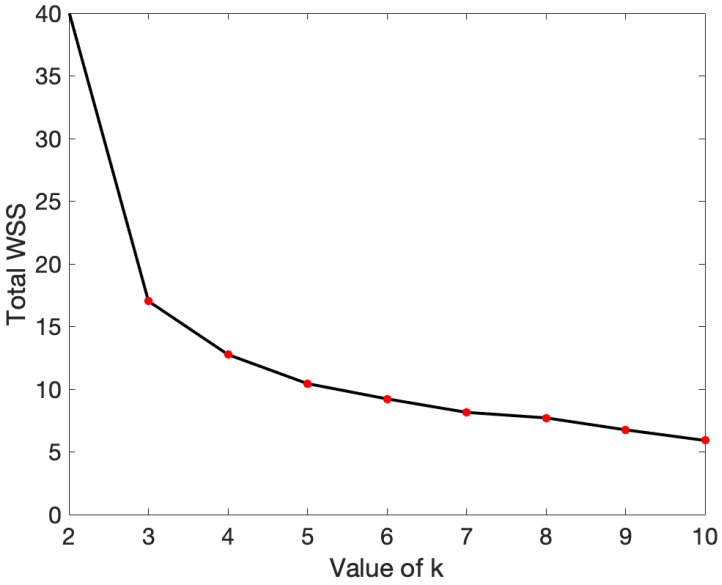


Figure 2.9 Elbow Method results for data depicted in Figure 2.8: total WSS as a function of k.

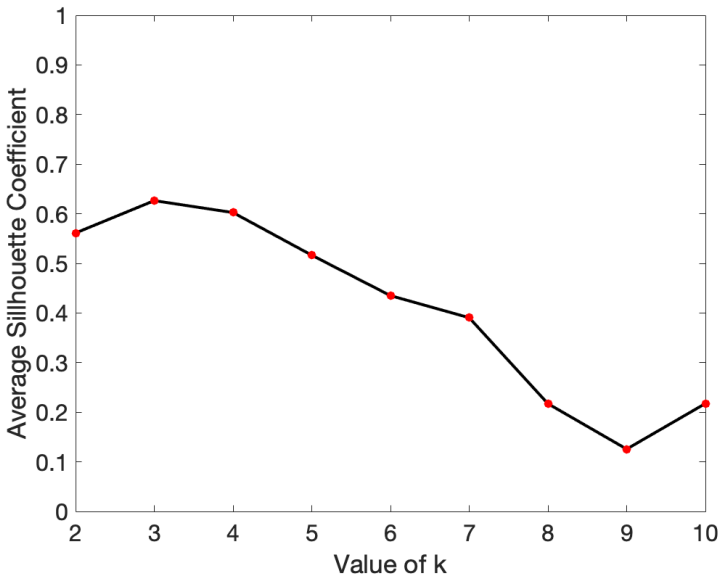


Figure 2.10 Average Silhouette Method results for data depicted in Figure 2.8: average S_i as a function of k.

thus complicating an otherwise straightforward algorithm. While the SOM algorithm also requires the user to prescribe the number of nodes (N) on the map before training can begin, its unique neighborhood feature provides more flexibility for how many clusters will actually be identified among the data.

2.4 Tailored Unsupervised Learning

A dual-domain clustering problem, defined in Chapter 1, presents unique challenges related to how information and similarity within each domain should be considered and prioritized. The Self-Organizing Map and k-Means algorithms, as outlined above, are two approaches designed to handle unsupervised learning tasks. However, these standard algorithms may fall short when applied to cluster a dual-domain data set. Especially in the case where the dimension of the attribute domain A is much larger than the dimension of the geographic domain G, these standard algorithms often fail to appropriately weight the distinct information obtained from each domain and result in either an over-simplified or geographically unfeasible clustering.

To overcome these challenges, we propose an unsupervised learning framework that highlights domain-specific similarity, while simultaneously considering global features of the data set. Our methodological framework, which still relies fundamentally on the principles of unsupervised learning, allows for the integration of an objective function that can be tailored to the geographic modeling application. With the introduction of an objective function, we are able to transform a traditional clustering task, for which a ground truth does not exist, into an optimization problem with a feasible solution set. In the next several chapters of this thesis, we outline the methodology, objective function construction, optimization approaches, and methods for algorithm assessment using synthetic data. Additionally, we will rely on two well-known, standard clustering algorithms, the Self-Organizing Map and the k-Means algorithm, to serve as comparisons for the tailored methodology developed in Chapter 3. These methods were chosen for their widely-accepted utility in clustering unlabeled data set and relatively few number of hyperparameters. Other common algorithms, such as the Fuzzy C-Means algorithm and an autoencoder, were also considered and examined for their use as potential comparisons.

The algorithmic approach of Fuzzy C-Means [6, 9] is very similar to that of the k-Means algorithm outlined above. However, in Fuzzy C-Means Clustering each data observation can be assigned to more than one of the C clusters. In other words, during training, the algorithm determines the probability of a given observation belonging to each of the C clusters. Therefore, the final clusters are referred to as "soft" clusters and often analyzed as probability distributions. Ultimately, given the many ways one could interpret the resulting cluster distributions determined by the Fuzzy C-Means algorithm, we felt it would be infeasible to use it as a frequent and repeated comparison. Similar to the Self-Organizing Map, an autoencoder (also referred to as Non-Linear Principal Component Analysis) is a type of artificial neural network that is designed to learn a low-dimensional (latent-space) representation of higher-dimensional data [20]. The autoencoder is not inherently a clustering algorithm, however, it does encode a representation of a data set into a lower-dimensional space. In short, an autoencoder aims to build an encoder of the input data, that compresses it into a latent-space representation, and a decoder, which reconstructs the input data from the latent-space, that minimize the difference between the true input and the reconstruction. Especially in the case of high-dimensional, noisy data, autoencoders are often used as a pre-processing tool for de-noising prior to the implementation of a standard clustering or classification algorithm, as demonstrated by [29]. To that end, we choose to reserve the use of autoencoders in our tailored framework for cases of dual-domain data not considered in this thesis, i.e., those with high-dimensional attribute spaces and an inhibitory level of noise.

CHAPTER

3

TAILORED CLUSTERING OF DUAL-DOMAIN DATA USING SELF-ORGANIZING MAPS

3.1 A Tailored Self-Organizing Map

In Chapter 2 we discussed in detail the algorithmic approach of the Self-Organizing Map, as well as the parameters associated with training the map. To better handle a clustering problem within the dual-domain configuration, we propose a novel, alternative approach called the Tailored Self-Organizing Map (TSOM). This new approach relies fundamentally on the standard Self-Organizing Map (SOM) algorithm and the incorporation of domain-specific biasing parameters and . These parameters are introduced into the competitive phase of the SOM algorithm (line 6 in Algorithm 1) and can be adjusted to result in a clustering that favors the features of one domain over the

other. A step by step outline of the Tailored Self-Organizing Map algorithm is depicted in Algorithm 3. The Tailored SOM algorithm is $O(k_{max} \cdot D \cdot N)$, where k_{max} is the number of batch training iterations, D is the number of observations in the data set, and N is the number of nodes in the map. The initialization of this algorithm still relies on the user-prescribed total number of nodes in the map (N), arrangement of the nodes (rectangular lattice, triangular lattice, etc.), and neighborhood relationships among the nodes. The weight vector, w_j, for each node is viewed as a concatenation of two subvectors. The first subvector, w_j^a, is associated with data in the attribute domain A and the second subvector, w_j^g, is associated with data in the geographic domain G. By considering each subvector as its own entity, we allow for more flexibility when determining domain-appropriate initializations, as well as interpretations of their final positions as cluster representatives.

The primary differences between the Tailored SOM algorithm (Algorithm 3) and the Standard SOM algorithm (Algorithm 1) can be seen in lines 4 and 7 of Algorithm 3, with the incorporation of the biasing parameters. These parameters, and , allow for more consideration to be given to the data in the attribute domain A or geographic domain G, respectively, when determining the winning weight vector at each iteration of training. In other words, by manipulating and we create a competitive learning scenario that favors one domain over the other. By increasing the relative magnitude of and , the similarity between the attribute (or geographic) components of each data observation and the weight vectors is weighed more heavily, thus producing a clustering that is more tailored to groups within the attribute domain A (or geographic domain G). When = = 1, the competitive phase of the Tailored SOM algorithm is equivalent to that of the Standard SOM algorithm. To maintain an appropriate balance between the domains, we institute a biasing threshold T, such that $\frac{\max(,)}{\min(,)} < T$ for all , . We will further discuss admissible parameter spaces and choice of an appropriate threshold in Section 3.3.

Algorithm 3: Tailored Self-Organizing Map

Choose topological arrangement, size (N), and neighboring relations of SOM;
Initialize learning rates $z(k)$ and $\quad(k)$;
Initialize weight vectors $w_j = [w_j^a, w_j^g]$, for $j = 1,..,N$ for each node of the SOM;
Initialize biasing parameter set $q = [\quad,\quad]$, such that $\quad > 0$, $\quad > 0$, and $\frac{\max(\ ,\ )}{\min(\ ,\ )} < T$;
for $k = 1,..,k_{max}$ *(the number of batch training iterations)*
  for $i = 1,..,D$ *(each data vector in the dual-domain data set* $\mathbf{S}$ *given by* $d_i = [a_i, g_i]$*)*
    Compute the index of the winning weight vector via the *biased* formulation:
    $c(i) = \text{argmin}_{j=1,..,N} \quad \|a_i - w_j^a\|_2^2 + \quad \|g_i - w_j^g\|_2^2$;
  end
  for $j = 1,..,N$
    $\mathbf{C}_j = \mathbf{S}(c == j,:)$ (a matrix of all data vectors with winning index j);
    Let $m_j = \ \mathbf{C}_j\ $ (the number of data vectors stored in C_j);
    Let N_j be the set of indices of the neighbors of w_j;
    Let $p_j = \ N_j\ $ (the number of neighbors associated with w_j) ;
  end
  for $j = 1,..,N$
    Let $w^* = w_j$;
    Update the j^{th} weight vector using $\mathbf{C}_j$;
    for $t = 1,..,m_j$
      $w_t = w^* + z(k)(\mathbf{C}_j(t,:) - w^*)$;
    end
    $w_j = \frac{1}{m_j} \sum_{t=1}^{m_j} w_t$;
    Update the j^{th} weight vector using the data associated with its neighbors;
    for $r \in N_j$
      for $s = 1,..,p_j$
        $w_s = w^* + \quad (k)(\mathbf{C}_r(s,:) - w^*)$;
      end
      $w_j = w_j + \frac{1}{p_j} \sum_{s=1}^{p_j} w_s$;
    end
  end
  Update the learning rates $z(k)$ and $\quad(k)$;
end
for $i = 1,..,D$ *(each data vector in the dual-domain data set* $\mathbf{S}$ *given by* $d_i = [a_i, g_i]$*)*
  Compute final winning cluster index $c(i) = \text{argmin}_{j=1,..,N} \quad \|a_i - w_j^a\|_2^2 + \quad \|g_i - w_j^g\|_2^2$;
end
return Final cluster indices c for $\mathbf{S}$ and final representative weight vectors w_j, for $j = 1,..,N$;

3.2 Designing an Objective Function

The introduction of and into our unsupervised learning algorithm allows for the construction of an *objective* function, J, that can be used to optimize their values in the context of the *objectives* of dual-domain clustering. We aim to design an objective function that is generic in the sense that it can be useful for many different data sets, yet also tailored to address the unique challenges present in a dual-domain clustering problem. Recall, these challenges include identifying clusters that adhere to geographic constraints and are geographically proximal, that simultaneously identify an information-dense representation of groupings in the attribute domain. To increase the likelihood that the Tailored SOM (Algorithm 3) will achieve a geographically feasible clustering of the data, we focus our attention on the geographic domain G when constructing the objective function J. Specifically, for each of the C clusters identified by the TSOM (with $C \leq N$) when trained with prescribed and , we consider its projection onto only the geographic domain G. For each cluster projection, we construct cluster boundaries using the convex hull of its geographic data. We do so via the native MATLAB function `boundary`, which creates a two- or three-dimensional boundary for a given set of points using a scaling parameter $s \in [0, 1]$. For $s = 0$, the boundary is the convex hull. For $s = 1$, the boundary is compact. Throughout this work, we will exclusively set the scaling parameter to $s = 0$ and rely on the properties of the convex hull. By imposing convex boundaries, we create a natural way of viewing the clusters as a contiguous geographic region, instead of as a discrete set of points [4]. Additionally, having the convex boundaries for each cluster allows for the incorporation of relevant two-dimensional quantities when specifying the objective function. For example, the average area of a cluster is a quantity that cannot be computed without the imposition of a boundary on a given cluster's data. Furthermore, we can consider instances of boundary intersections, total cluster region overlap, etc., to assess how well-separated, or unique, the identified geographic regions are.

When designing the objective function we recall the motivating goal: to find a clustering of a dual-domain data set that is both telling of the intrinsic structure of data in the attribute domain A and is geographically useful. In other words, we aim to identify clusters of "individuals" (observations) with similar attributes that are geographically proximal. To that end, we define the

objective, J_g , , to be a function of the geographic projection of the TSOM clusters, and hence denoted by the subscript g, for prescribed and . Thus, we consider the quantities defined in Table 3.1.

Table 3.1 Clustering-based quantities used in construction of objective function.

Name	Description
C	the total number of identified clusters (nodes in TSOM populated with data)
n_i	the number of data observations assigned to the i^{th} cluster, for $i = 1,..,C$
$A_{i,j}$	the overlapping area between the i^{th} and j^{th} geographic clusters, for $i, j = 1,..,C$ and $i \neq j$
$\bar{A}$	the average geographic cluster area
p_i	the perimeter of the i^{th} geographic cluster, for $i = 1,..,C$
$\bar{p}$	a prescribed “representative" cluster perimeter of the geographic data

To further illustrate the terms in Table 3.1 in the context of a clustering identified by the Tailored SOM, projected into the geographic domain G, we consider the example data depicted in Figure 3.1. The clusters seen in Figure 3.1 are the geographic components of the results obtained from implementing the TSOM algorithm for given values of and . As depicted by the marker types, the TSOM algorithm identified two clusters in the data, denoted by the red x's and green squares. Thus, for this example, we have $C = 2$, $n_1 = 11$ (red x's cluster), and $n_2 = 11$ (green squares cluster). It is important to note that in addition to the geographic observations seen in Figure 3.1, this fictitious data set also has data vector components in an associated attribute domain A (not pictured) that was considered during the training process. Post-clustering, we determine the convex hull of each geographic cluster and impose boundaries (Figure 3.1). The convex boundaries allow us to consider each geographic cluster as a region, instead of as just a discrete set of points.

Figure 3.2 serves to illustrate the terms in Table 3.1 in the context of the data seen in Figure 3.1. The third term in Table 3.1, $A_{i,j}$, is illustrated by the region shaded black (seen in (a)), which is the area common to both clusters. Recall the overarching goal of uncovering a clustering of a given data set that identifies, to the extent possible, unique geographic groupings of data. To that end, determining values of and that minimize the quantity $A_{i,j}$, for $i, j = 1,..,C$ and $i \neq j$, would

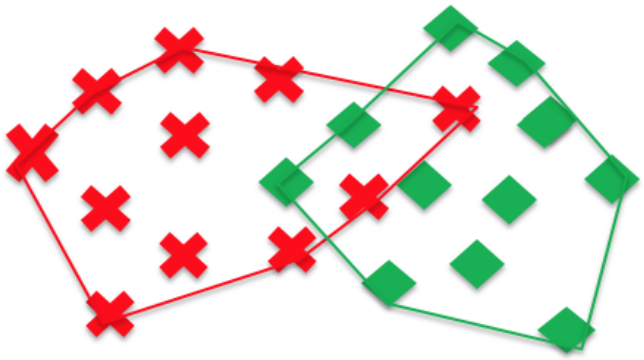
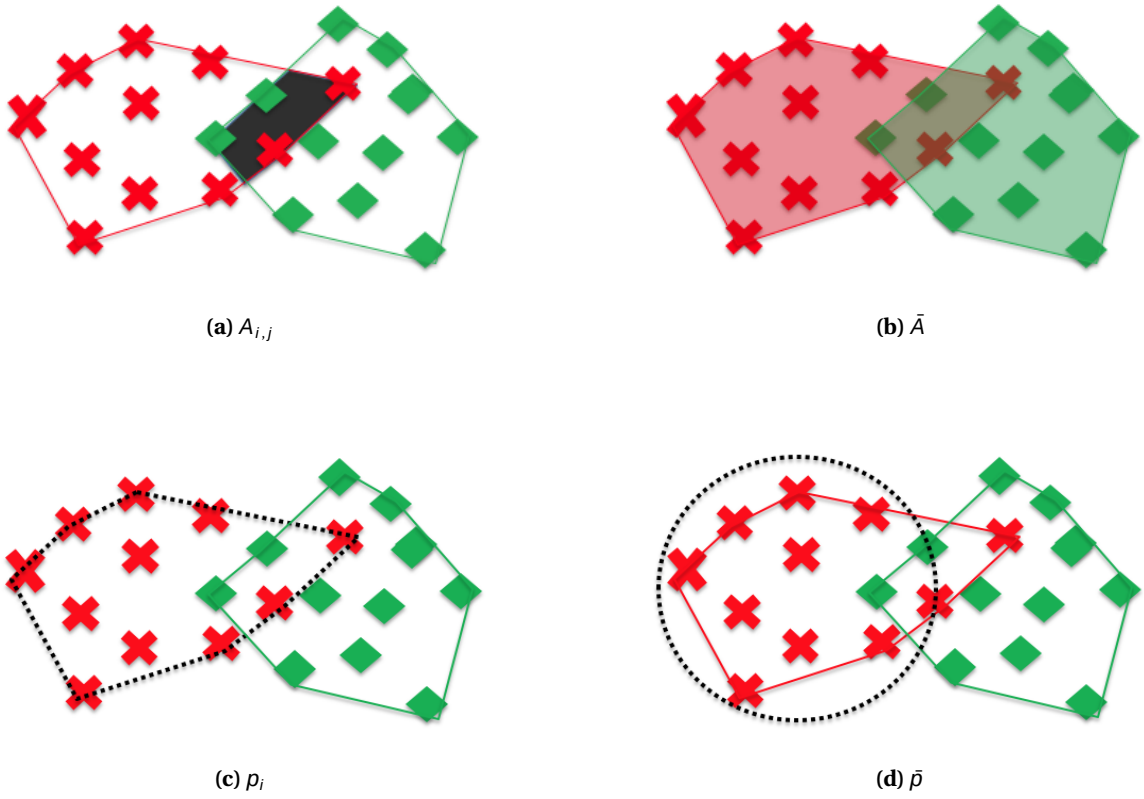


Figure 3.1 An example of a dual-domain data set, clustered by the Tailored SOM and projected into G. Once clustered and projected, we construct the convex hull of each geographic cluster. The convex hulls act as boundaries for the clusters and create a geographic region for each cluster.

be useful. The fourth term in Table 3.1, $\bar{A}$, is illustrated by the mean value of the areas of the red and green shaded regions (seen in (b)). In general, $\bar{A}$, is the mean value of C geographic cluster areas, where C is the number of populated nodes in the TSOM. This quantity, when incorporated into the objective function J_g, can be used to control the size of the identified geographic regions. For example, this term can help promote a clustering that amalgamates several smaller clusters to create a larger one. The fifth term in Table 3.1, p_i, for $i = 1,..,C$, is depicted by the black dotted boundary applied to the red cluster (seen in (c)). We compute each geographic cluster's perimeter, p_i, by summing the edge lengths of its convex hull. Along with each cluster's perimeter, we determine a representative perimeter, $\bar{p}$, that acts as an "ideal" perimeter and can be used to help smooth and tune geographic properties of the clustering. Segmenting and smoothing algorithms, such as the Mumford-Shah formulation [36], are prevalent in the study of image segmentation [35], where one seeks to partition an image into distinct sub-regions according to some measure of similarity or relevance. In our work, we aim to segment the geographic data observations into clusters (and via convex hulls, regions) that are optimal for a given application. Thus, one reasonable choice for a representative perimeter, $\bar{p}$, is the circumference of a circle (seen in (d)). For example, this choice would be appropriate in an application determining optimal sales territories, where it is beneficial to have approximately equivalent point-to-centroid distances within clusters in order to minimize cost, travel time, etc. However, other geometries may be more ideal depending on the data set or specific knowledge about the geographic domain, or application context.



(a) $A_{i,j}$ **(b)** $\bar{A}$

(c) p_i **(d)** $\bar{p}$

Figure 3.2 An illustration of various terms (defined in Table 3.1) to be included in the objective function $J_g(\ ,\ )$. (a) $A_{i,j}$: the overlapping area between two geographic clusters. (b) $\bar{A}$: the mean area of the two geographic clusters, shown as the red and green shaded regions. (c) p_i: the perimeter of a geographic cluster denoted by the black dotted line segments. (d) $\bar{p}$: an example of a circular representative perimeter denoted by the black dotted circle.

With the terms outlined in Table 3.1, we propose the objective function J_g , defined in equation (3.1).

$$J_g \quad , \quad = \sum_{\substack{i,j=1 \\ i \neq j}} \frac{A_{i,j}}{\bar{A}} + \frac{1}{C} \sum_{i=1} \bullet \frac{p_i}{\bar{p}} - 1 \quad ^{\prec 2} \tag{3.1}$$

We note that J_g is not explicitly defined in terms of , , rather it depends on quantities derived from a geographic clustering determined by the Tailored SOM for given values of , . Thus, we choose to refer to the objective function as J_g , as a reminder that the choice of , directly impacts the clustering that is determined by the Tailored SOM, and subsequently the evaluation of the terms in J_g , . When minimized, we aim for the formulation of J_g , seen in equation (3.1) to select an "optimal" parameter set $q^* = [\ ^*, \ ^*]$ such that

$$q^* = [\ ^*, \ ^*] = \underset{,}{\operatorname{argmin}} J_g \quad , \quad . \tag{3.2}$$

We note that J_g , (equation (3.1)) is constructed to be dimensionless, with each term in the function having equal weight. Given the weighting provided by implementing , in the TSOM training, we choose to not impose additional weighting parameters on the individuals terms in J_g , . However, it is possible that term-specific weights could be advantageous for certain formulations of J_g , . Based on the particular formulation of J_g , seen in equation (3.1), we define the "optimal" parameter set as one that produces a useful clustering via the Tailored SOM (Algorithm 3) with:

- minimal overlap among geographic cluster regions (first term in equation (3.1)),
- geographic clusters with boundaries that do not span/violate geographic constraints, such as a river (both terms in equation (3.1)), and
- geographic clusters of comparable size, in terms of their perimeters (second term in equation (3.1)).

Of course, this is just one possible formulation of an objective function that can be used to determine favorable values of and . Other applications may require additional terms to be

included, or existing terms to be deleted from the objective function defined in equation (3.1). For example, prior knowledge of specific features of the geographic domain, such as shape and size constraints, can be used when designing and tailoring terms in J_g , . Additionally, some applications may lend themselves better to an objective function that relies more heavily on a metric derived from, or inherent to, features in the attribute domain. In this case, one may choose to formulate an objective function that penalizes for a large quantity of observations falling outside n-standard deviations of their attribute cluster centroid, or dissimilar cluster population sizes.

3.3 Optimizing the Objective Function

To determine the optimal parameter set $q^* = [\ ^*, \ ^*]$ (equation (3.2)) that will minimize the prescribed objective function J_g , , we begin by defining admissible ranges of values. Recall the implementation of and in line 7 of the Tailored SOM algorithm (Algorithm 3), which states that for each data vector $\vec{d}_i = [\vec{a}_i, \vec{g}_i]$, for $i = 1, .., D$, we compute the index of the winning weight vector via the *biased* formulation

$$c(i) = \underset{j=1,..,N}{\operatorname{argmin}} \quad \|\vec{a}_i - \vec{w}_j^{\,a}\|_2^2 + \quad \|\vec{g}_i - \vec{w}_j^{\,g}\|_2^2. \tag{3.3}$$

This step is an expansion upon the competitive learning portion of the Standard SOM algorithm (Algorithm 1, line 6), which states that for each data vector $\vec{d}_i$, the index of winning weight vector will be determined via

$$c(i) = \underset{j=1,..,N}{\operatorname{argmin}} \|\vec{d}_i - \vec{w}_j\|_2^2.. \tag{3.4}$$

Based on the formulations of equations (3.3) and (3.4), we note that the two approaches will yield the same winning weight vector index $c(i)$ for a given data vector $\vec{d}_i$ for all parameter sets $q = [\ , \ ]$ such that = . Consequently, we anticipate the possibility of multiple (i.e., non-unique) local minima solutions arising for any ranges of admissible values that are not mutually exclusive. Additionally, the discrete nature of a clustering problem will inevitably lead to multiple (i.e., non-unique) local minima solutions. That is to say, since there are only finitely many ways a given set of data can be grouped into N clusters, it is likely that some perturbations to $q = [\ , \ ]$ during optimization may not yield changes to the clustering landscape, depending on the magnitude of the perturbations

relative to the parameter values. Hence, we anticipate cases where $J_g(\alpha_1, \beta_1) = J_g(\alpha_2, \beta_2)$ even though $[\alpha_1, \beta_1] \neq [\alpha_2, \beta_2]$. Thus, for the sake of simplicity, we choose the admissible parameter space to be of the form

$$[\alpha, \beta] \in [1-\gamma, 1+\gamma] \times [1-\gamma, 1+\gamma] \tag{3.5}$$

with $\gamma \geq 0$. This choice allows for the space of values where $\alpha > \beta$, and the space where $\beta > \alpha$, to be explored, while also maintaining a reasonable ratio, $\frac{\max(\alpha, \beta)}{\min(\alpha, \beta)}$, for γ appropriately small (choices for γ will be explored in Chapter 4). To that end, we introduce a threshold, T, to act as an upper bound on the parameter ratio, seen in equation (3.6), thus preventing the over-minimization (or over-emphasis) of one domain's information.

$$\frac{\max(\alpha, \beta)}{\min(\alpha, \beta)} < T \tag{3.6}$$

The choice of T will in turn dictate admissible values for γ. Additionally, for some bounded optimization regimes, this admissible space affords us the choice to initialize at $[\alpha_0, \beta_0] = [1, 1]$, which results in a first iteration of clustering equivalent to the Standard SOM algorithm (Algorithm 1).

Lastly, with an admissible parameter space in place, we seek to evaluate and compare appropriate and efficient optimization approaches. As previously mentioned, given the inclusion of biasing parameters and the discrete nature of a finite clustering problem, we fully anticipate the solution surface of $J_g(\alpha, \beta)$ to be one with many non-unique, local minima. Thus, it is in our best interest to explore methods for global optimization, rather than rely on methods designed for problems with a single unique solution and a smooth solution surface. For smaller problems, we can employ a global grid search to determine values of α and β that minimize $J_g(\alpha, \beta)$. For the grid search, we define a step size,

$$q = [\alpha, \beta] \tag{3.7}$$

with $\alpha = \beta$, that is used to traverse the admissible parameter space (equation (3.5)) in both dimensions. While this process can be slow for high-dimensional data sets, we can improve efficiency by taking advantage of the relationship between α and β in equation (3.3) to avoid evaluating $J_g(\alpha, \beta)$ unnecessarily. In other words, depending on the magnitude of q, we can determine pairs

of and values that put the parameters in equal ratio, and skip over all but one of these pairs when traversing the space and evaluating J_g , . For example, we can disregard iterations of the grid search where = , since it is equivalent to the result obtained for = = 1. Additionally, we can further improve efficiency by experimenting with the magnitude of q, as each data set will require a different level of granularity to achieve a sufficiently holistic view of the solution space. The illustration in Figure 3.3 depicts a high-level overview of this optimization approach as it relates to a dual-domain clustering via the Tailored SOM.

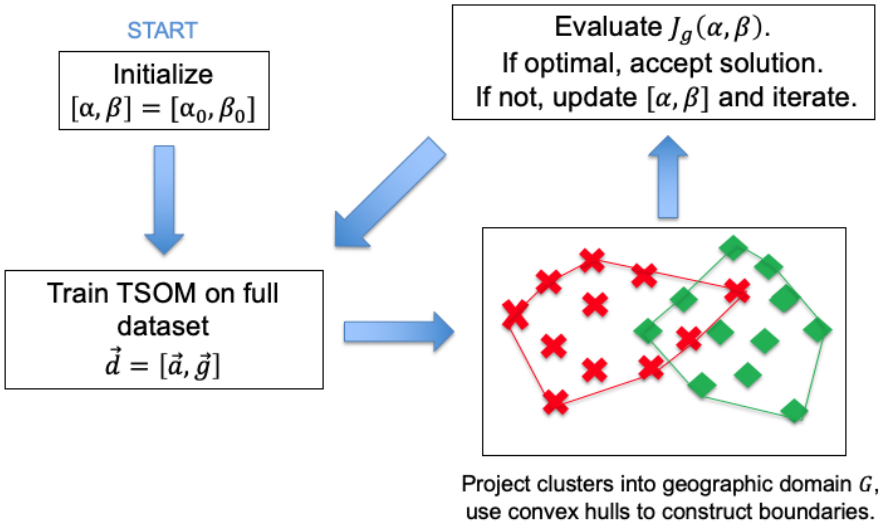


Figure 3.3 A depiction of the optimization framework for determining the optimal parameter set q^* = [*, *] for a given data set via the Tailored SOM (TSOM) algorithm and an objective function J_g , .

For larger data sets with a high-dimensional attribute space or a large number of observations, a global grid search may be unfeasible. In that case, to determine q^* = [*, *] we rely on a probabilistic global optimization method known as Simulated Annealing [8, 16, 22, 33, 41]. Other methods, such as Stochastic Gradient Descent, were also considered for this optimization task. However, given that J_g is not explicitly a function of the parameter set q = [,] and it fundamentally depends on the entire data set for evaluation (i.e., it depends on quantities derived from the full geographic clustering), we ultimately decided that a gradient-based method was not the best choice [41]. Alternatively, Simulated Annealing, which relies only on objective function evaluations, is a general method used

to determine the global minimum for a real-valued cost function even when its solution surface contains many local minima. This method is based on the thermodynamic principles of the cooling and annealing of metals. If a metal is held at a high temperature, its particles have a high probability of transitioning to a higher energy state. However, as the metal is cooled at a controlled rate (i.e., it is annealed) this probability decreases and particles are more likely to transition to a lower energy state. Annealing is complete when the temperature of the metal has reached equilibrium with its environment and the particles are at their lowest possible energy state. Simulated Annealing as an optimization algorithm mimics this physical process, relying on the length of annealing (total search duration) and a probability function to determine the global minimum of a cost function (an analog to the particles' lowest energy state). A step by step outline of the general Simulated Annealing algorithm is depicted in Algorithm 4 [34, 41].

In summary, the algorithm searches the admissible solution space for a minimum by randomly selecting candidates, accepting candidates that lower the cost function value with probability 1, and accepting candidates that do not lower the cost function (denoted by $c(x)$ in Algorithm 4, and analogous to J_g for our purposes) according to a decreasing probability function A. By probabilistically accepting candidate solutions that increase the value of the cost function, the solution space can be better explored and the algorithm is less likely to settle in a local minimum. A typical form for the acceptance probability function, $A(c(x), T)$ is

$$A(c(x), T) = \frac{1}{1 + \exp\ \frac{c}{T}} \tag{3.8}$$

where $c = c(x_i) - c(x_{i-1})$ and T is the current temperature. With c and T strictly positive, the range of A is (0, 1=2). Additionally, given the formulation of A, the algorithm is more likely to accept candidate solutions that increase the value of the cost function early on in the search, when the temperature is higher, rather than later in the search when it should be settling near the global minimum.

When considering a dual-domain data set with a high dimensional attribute space, we employ the Simulated Annealing algorithm to search the bounded admissible parameter space for q^*, as defined in equation (3.2), in lieu of using a global grid search. In Chapter 4 of this thesis, we will

Algorithm 4: Simulated Annealing Algorithm

Initialize the temperature T;
Define the ending temperature T_{end} and the decay constant ;
Define the acceptance probability function $A(c(x), T)$, where $c(x)$ is the cost function to be minimized;
Generate initial random solution x_0 and compute its cost $c(x_0)$;
Initialize acceptance counter $i = 1$;
while $T < T_{end}$ **do**
    Generate current solution x_i from the neighborhood of previous solution x_{i-1};
    Compute cost of current solution $c(x_i)$;
    if $c(x_i) < c(x_{i-1})$ **then**
        Accept x_i as solution;
    else
        Accept x_i as solution according to the acceptance probability $A(c(x), T)$;
    end
    if x_i *accepted* **then**
        Iterate i: $i = i + 1$;
    else
        Do not iterate i;
    end
    Update the temperature: $T = T$;
end
Compute best solution: $x^* = \text{argmin}_{x_i} c(x_i)$;
Compute minimized cost: $c(x^*)$;
return Best solution x^* and value of minimized cost $c(x^*)$;

use synthetic data to demonstrate and compare the performance of a global grid search versus the Simulated Annealing algorithm in optimizing J_g , and determining a geographically useful clustering.

CHAPTER

4

EVALUATING THE TAILORED SELF-ORGANIZING MAP WITH SYNTHETIC DATA

4.1 Approach

In this chapter, we will test and validate the tailored unsupervised learning methodology that we developed in Chapter 3 on three cases of synthetic dual-domain data sets. In summary, the methodology is comprised of:

1. Prescribing the map parameters and learning rates associated with the Tailored Self-Organizing Map algorithm (TSOM) (Algorithm 3) and initializing the weight vectors.

2. Prescribing an admissible space for and via the parameters and T (equations (3.5) and (3.6)).

3. Determining the optimal values for and that minimize the objective function $J_g(\ ,\ )$

(equation (3.1)) via: (1) a global grid search or (2) Simulated Annealing.

Once an optimal parameter set $q^* = [\ ^*, \ ^*]$ (equation (3.2)) has been determined, we will compare the corresponding Tailored SOM clustering to results obtained from the Standard Self-Organizing Map algorithm (Algorithm 1) and the k-Means algorithm (Algorithm 2). To compare the results of each algorithm, we will use the objective function J_g to evaluate each clustering in terms of the geographic characteristics used to define it. Additionally, we will propose the concept of *geographic feasibility* and the notion of a "desired answer" based on known groupings of data within the synthetic data set. With these metrics, we can validate the results of the Tailored SOM relative to the Standard SOM and k-Means algorithms in a consistent way that is specific to the goals of the dual-domain clustering problem.

4.2 Generation of Dual-Domain Synthetic Data

In the absence of a ground truth against which we can train a clustering algorithm, we begin by designing and generating experimental, synthetic data for which we can define a right answer. With synthetic data generation comes the freedom to customize; we can vary size, in terms of both the number of observations and features in the data set, noise level and prevalence of outliers in both domains, and type of constraint present in the geographic domain. By carefully varying each of these components, we can thoroughly test competing algorithms for efficiency and robustness when applied to synthetic data sets with measurable differences. Our methodology for generating data is similar to that of a stochastic block model [2], which is a generative random graph model often used in the study of community (cluster) detection within a network structure [14].

In the attribute domain, we create a data arrangement A^* containing p_a groups each with a centroid $M_a = [m^i_{a,x}, m^i_{a,y}]$, for $i = 1,..,p_a$, and D observations. The observations a^i_j, for $i = 1,..,p_a$ and $j = 1,..,D$, in each of the p_a groups are generated from a circular region about the centroid. The radii r^a_j, for $j = 1,..,D$, are drawn from the normal distribution $r^a \sim N(0, \ ^2_1)$ and the angular measures $\ ^a_j$, for $j = 1,..,D$, are drawn from the uniform distribution $\ ^a \sim U(0, 2\ )$. Thus, each

observation a_j^i, for $i = 1,..,p_a$ and $j = 1,..,D$, is of the form

$$a_j^i = [r_j^a \cos(\ _j^a), r_j^a \sin(\ _j^a)] + [m_{a,x}^i, m_{a,y}^i].$$

In the geographic domain, we generate a corresponding geographical arrangement G^* containing p_g groups each with a centroid $M_g = [m_{g,x}^i, m_{g,y}^i]$, for $i = 1,..,p_g$. Each observation $a_j^i \in A$, for $i = 1,..,p_a$ and $j = 1,..,D$, is associated with an observation in a corresponding geographic group with probability P and associated with an observation in a non-corresponding geographic group with probability $1-P$. Geographic data observations g_j^i, for $i = 1,..,p_g$ and $j = 1,..,D$, are generated from a circular region about the centroid. Here, each geographic group p_g has N_{p_g} observations, which is dependent on the prescribed association probability P. For $P \approx 1$, the mean value of N_{p_g} will approach $\frac{D}{p_g}$ over many realizations. The radii r_j^g, for $j = 1,..,D$, are drawn from the normal distribution $r^g \sim N(0,\ _2^2)$ and the angular measures $\ _j^g$ are drawn from the uniform distribution $^g \sim U(0, 2\ )$. Thus, each g_j^i, for $i = 1,..,p_g$ and $j = 1,..,D$, is of the form

$$g_j^i = [r_j^g \cos(\ _j^g), r_j^g \sin(\ _j^g)] + [m_{g,x}^i, m_{g,y}^i].$$

Figure 4.1 (a,b) depicts the first sample data arrangement, generated using the parameters outlined above. In generating this sample set, we prescribe the parameter set

$$Q_1 = [p_a, p_g, D,\ _1,\ _2, P] = [4, 5, 200, 0.15, 0.15, 1]$$

and choose the group centroids

$$M_{a,1} = \begin{bmatrix} 0.5 & 0.5 \\ -0.5 & 0.7 \\ -0.4 & -0.3 \\ 0.4 & -0.4 \end{bmatrix} \qquad M_{g,1} = \begin{bmatrix} 0.5 & 0.4 \\ -0.5 & 0.2 \\ -0.1 & -0.2 \\ 0.1 & -0.1 \\ 0.5 & -0.5 \end{bmatrix}.$$

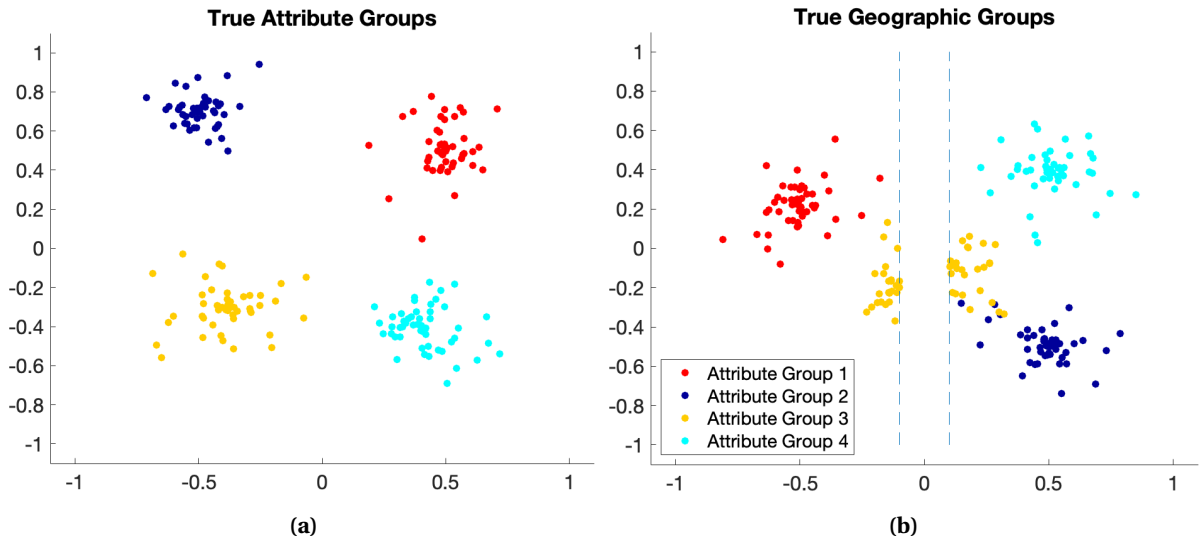


Figure 4.1 A sample dual-domain data arrangement generated from the parameter values specified by Q_1, $M_{a,1}$, and $M_{g,1}$ projected into the two-dimensional attribute domain (shown in (a)) and the two-dimensional geographic domain (shown in (b)). The marker colors in each subfigure are used to indicate the inter-domain correspondence/association between the geographic observations and the attribute observations.

In these figures, we use the marker colors to demonstrate inter-domain group connections, or correspondence. For example, the red data in Figure 4.1 (b) are the geographic locations corresponding to each of the red attribute observations in Figure 4.1 (a). We note that since the association probability $P = 1$, there are no misplaced geographic data observations, in terms of their color, i.e., they were all drawn from a corresponding/associated attribute group distribution.

By adjusting the values of $_1$, $_2$, and P used in data generation, we can create arrangements with varying levels of noise, and thus a clustering task with varying levels of difficulty. To contrast the example data set seen in Figure 4.1, we consider the parameter set

$$Q_2 = [p_a, p_g, D, \ _1, \ _2, P] = [4, 5, 200, 0.15, 0.15, 0.75]$$

and group centroids $M_{a,1}$, $M_{g,1}$ defined above. With these values, we generate a second sample dual-domain data set, depicted in Figure 4.2 (a,b). By holding $_1$ and $_2$ constant at 0.15 and lowering the association probability P to 0.75, we allow for each attribute data observation $\vec{a}$ to have a 25% chance of being associated with a geographic observation $\vec{g}$ from a non-corresponding group, thus introducing a substantial amount of noise in the geographic space (depicted in Figure 4.2 (b)).

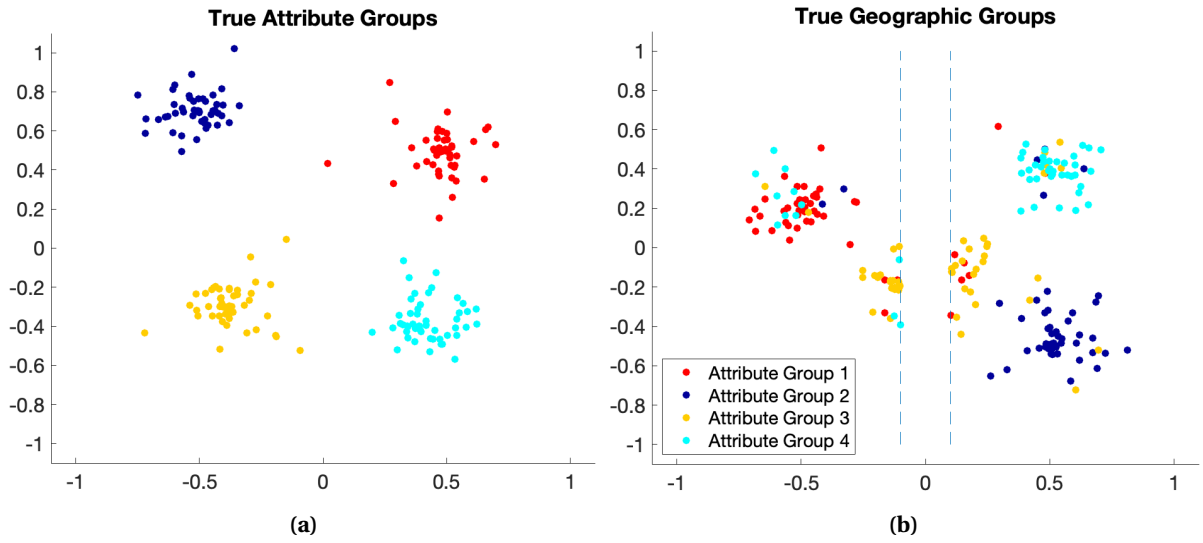


Figure 4.2 A second sample dual-domain data arrangement generated from the parameter values specified by Q_2, $M_{a,1}$, and $M_{g,1}$ projected into the two-dimensional attribute domain (shown in (a)) and the two-dimensional geographic domain (shown in (b)). The marker colors in each subfigure are used to indicate the inter-domain correspondence/association between the geographic observations and the attribute observations.

For a third example, we generate the sample dual-domain data arrangement depicted in Figure 4.3 (a,b). This third sample set was generated using the parameter set

$$Q_3 = [p_a, p_g, D, \ _1, \ _2, P] = [4, 5, 200, 0.25, 0.25, 0.95]$$

and cluster centroids $M_{a,1}$, $M_{g,1}$, as defined above. For this realization, we increase the association probability to 0.95, thus allowing for each attribute domain observation a to have a 5% chance of being associated with a geographic domain observation g from a non-corresponding group (thus $1-P$ reflects the probability of association with a non-corresponding group). By letting $_1 = \ _2 = 0.25$, we increase the radial spread of each attribute and geographic group, which produces a larger number of outliers per geographic and attribute group and increases the likelihood of group mixing. We summarize the parameters and definitions used for synthetic data generation throughout this thesis in Table 4.1.

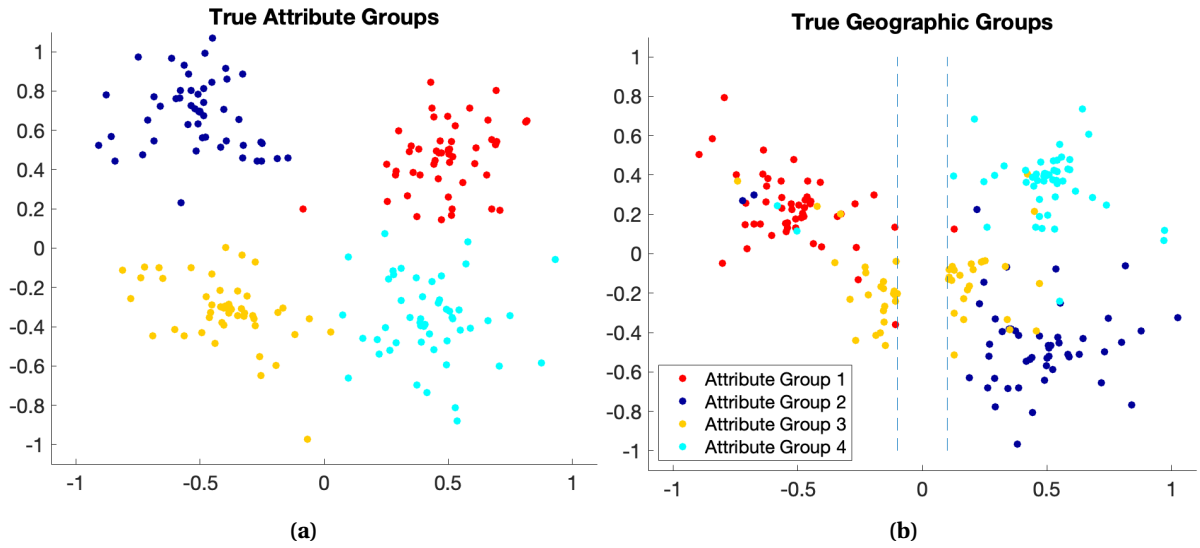


Figure 4.3 A third sample dual-domain data arrangement generated from the parameter values specified by Q_3, $M_{a,1}$, and $M_{g,1}$ projected into the two-dimensional attribute domain (shown in (a)) and the two-dimensional geographic domain (shown in (b)). The marker colors in each subfigure are used to indicate the inter-domain correspondence/association between the geographic observations and the attribute observations.

Table 4.1 Parameters used to generate synthetic dual-domain data.

Parameter	Definition
p_a	number of groups in A
p_g	number of groups in G
D	total number of observations in the data set
σ_1^2	radial variance for observations in A
σ_2^2	radial variance for observations in G
P	association probability
M_a	group centroids in A
M_g	group centroids in G

4.3 Synthetic Data: Case 1

For our first case, we manufacture a synthetic data set in which there are an equal number of attribute and geographic groups ($p_a = p_g$), as described in Section 4.2. We will use the term “group" when referring to a *known* grouping of data within a data set, i.e., a grouping that we specified, for which we know the generating distribution. We seek to identify these groups,without prior knowledge of them, when applying a clustering algorithm. We will reserve the use of the term “cluster" for describing the findings or output of an unsupervised algorithm. In addition to the number of attribute and geographic groups (p_a and p_g), we prescribe values for the total number of observations in the data set (D), the radial variances ($^2_1$ and $^2_2$) for the attribute and geographic groups relative to cluster centroids, the association probability (P), and the centroids of the attribute and geographic groups (M_a and M_g). These quantities were all defined in Section 4.2. We prescribe values for each of these parameters for Case 1 synthetic data, as indicated in Table 4.2.

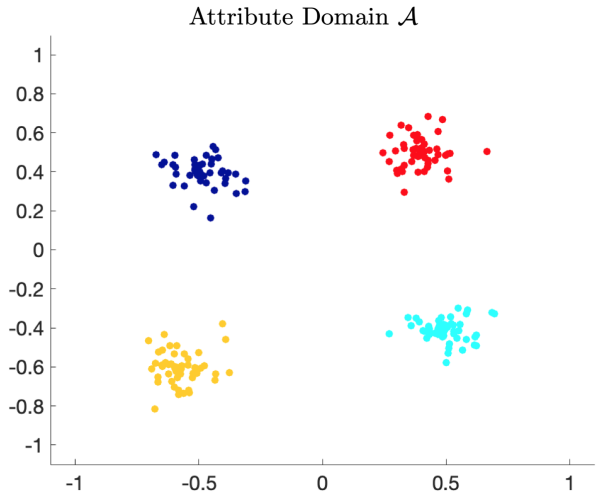
Recall that, in generating data sets, each attribute observation $a^i_j \in A$, for $i = 1,..,p_a$ and $j = 1,..,D$, is associated with an observation in a *corresponding* geographic group with probability P and associated with an observation in a *non-corresponding* geographic group with probability $1-P$. Therefore, while we can control the number of observations in each attribute group, N_i for $i = 1,..,p_a$, the number of observations in each corresponding geographic group, N_j for $j = 1,..,p_g$, is dependent upon the prescribed association probability P. In addition to these parameters (which are prescribed below in Table 4.2), we institute a vertical (north-south) geographic constraint in the form of a river. A depiction of this constraint, along with a sample data set generated from the parameter values depicted in Table 4.2, are shown in Figure 4.4.

First, we illustrate the projection of the sample data set into the attribute domain A (Figure 4.4 (a)). We note that we are not introducing the concept of noise nor observation misplacement among the groups in the attribute domain. This choice, paired with a low radial variance $^2_1$, results in four easily discernible attribute groups. Additionally, we illustrate the geographic data (Figure 4.4 (b)) that corresponds with the attribute domain projection (Figure 4.4 (a)). In Figure 4.4, we use the marker colors to indicate the inter-domain correspondence among groups. For example, the red geographic observations (Figure 4.4 (b)) are the geographic components of the red attribute

observations (4.4 (a)). Given that we instituted an association probability $P < 1$, we observe a low level of observation mismatch among the geographic groups. For Case 1, we intentionally keep the radial variances, σ_1^2 and σ_2^2, small, the group centroids, M_a and M_g, well-separated (within each domain), and the association probability, P, close to 1. In subsequent synthetic data experiments, we will systematically adjust and increase these values to test the robustness of various algorithms.

Table 4.2 Parameter values used to manufacture the first case of synthetic data.

Parameter	Value	Definition
p_a	4	number of groups in A
p_g	4	number of groups in G
D	200	total number of observations in the data set
σ_1^2	0.12^2	radial variance for observations in A
σ_2^2	0.15^2	radial variance for observations in G
P	0.92	association probability
M_a	$\begin{bmatrix} 0.4 & -0.5 & -0.6 & 0.5 \\ 0.4 & 0.4 & -0.6 & -0.3 \end{bmatrix}^T$	group centroids in A
M_g	$\begin{bmatrix} 0.5 & 0.5 & -0.6 & -0.6 \\ 0.4 & -0.5 & 0.35 & -0.35 \end{bmatrix}^T$	group centroids in G



(a) **(b)**

Figure 4.4 A sample Case 1 data set generated from the parameter values defined in Table 4.2 projected into the two-dimensional attribute domain (Subfigure (a)) and the two-dimensional geographic domain (Subfigure (b)). The marker colors in each subfigure are used to indicate the inter-domain correspondence between the geographic observations and the attribute observations.

4.3.1 Tailored versus Standard Clustering Results: Case 1

To begin training, we must define the various map parameters and learning rates associated with the Tailored SOM Algorithm (Algorithm 1). We choose to implement a 3×3 map, consisting of $N = 9$ nodes, arranged as a 2-dimensional rectangular lattice (as depicted by the example in Figure 2.2). We use the structure of the rectangular lattice to define the neighborhood relation among the nodes, i.e., nodes that are one edge length apart on the lattice are neighbors, while those diagonally adjacent are not. For Case 1, we initialize the weight vectors w_j, for $j = 1,..,N$, such that the components are evenly spaced throughout each sub domain ($[-1,1] \times [-1,1]$ for A and G). This choice of initialization does not afford the training algorithm any biased knowledge of the data landscape prior to the first iteration of training. In subsequent synthetic data experiments, we will investigate additional choices for weight vector initialization and discuss their effects on algorithm performance. For the learning rates $z(k)$ and (k), we use the forms defined in equations (2.1) and (2.2) with $a = \frac{1}{2}$, $b = \frac{1}{8}$. We summarize the parameters and design choices prescribed for this Tailored SOM in Table 4.3.

Table 4.3 Parameters and design choices prescribed for training the Tailored SOM on Case 1 synthetic data.

Parameter	Value	Definition
N	9	total number of nodes in SOM (given by 3×3 rectangular lattice structure)
initial w_j^a $(j = 1,..,N)$	evenly spaced in $[-1,1]\times[-1,1]$	weight vector initialization in A
initial w_j^g $(j = 1,..,N)$	evenly spaced in $[-1,1]\times[-1,1]$	weight vector initialization in G
w_n	1 edge away	neighborhood relation among nodes
$z(k)$	$\frac{1}{2}\,\frac{k}{k_{\max}}$	learning rate for winning weight vectors
(k)	$\frac{1}{8}\,\frac{k}{k_{\max}}$	learning rate for neighboring weight vectors
$k_{\max}$	12	number of training batches

For the optimization framework, as outlined in Section 3.3, we define the admissible parameter space (equation (3.5)) via = 0.25, i.e., we will optimize J_g , (equation (3.1)) over

$$[\ ,\ ] \in [0.75, 1.25]\times[0.75, 1.25]. \tag{4.1}$$

This choice of results in a parameter ratio threshold (equation (3.6)) of $T = \frac{5}{3} \approx 1.67$. Additionally, we choose to institute a circular representative perimeter, $\bar{p}$, with radius $R = 0.3$, in our formulation of J_g , (equation (3.1)). First, to determine the optimal $q^* = [\ ^*,\ ^*]$ (equation (3.2)) for the example Case 1 data set seen in Figure 4.4, we perform a global grid search of the parameter space using the step size $q = [0.05, 0.05]$. To depict the results of this grid search, we construct the heat map shown in Figure 4.5. Each square on the heat map represents the function value of J_g , when evaluated at the corresponding , values indicated on the x- and y-axis. We use the scaled colors seen in this heat map to illustrate the various peaks and valleys attained by J_g , during the search. The optimal parameter set for this data is $q^* = [0.8, 1.1]$, which yields the minimum value $J_g(0.8, 1.1) = 0.9192$, as depicted in Figure 4.5 by the deep blue color on the square corresponding to this parameter pair.

When used during training of the Tailored Self-Organizing Map, this optimal set, q^*, gen-

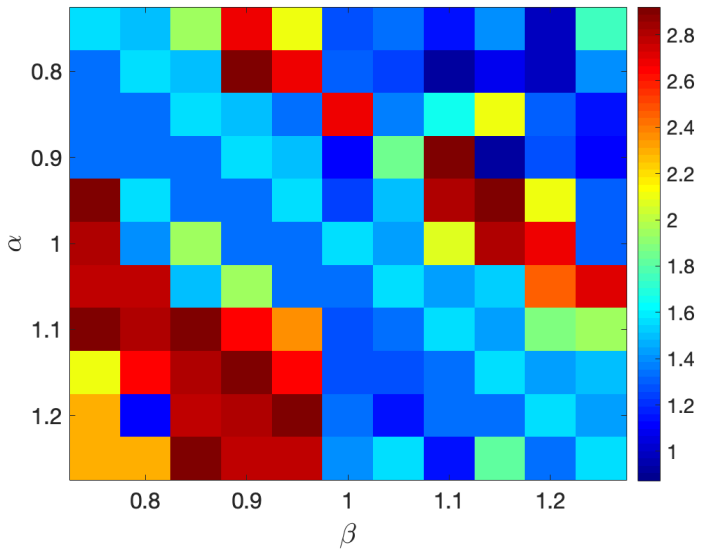


Figure 4.5 An illustration of the function values attained by J_g , (equation (3.1)) at each , pair (equation (4.1)) evaluated during the global grid search for the Case 1 data set depicted in Figure 4.4. Each function value is denoted by the color of the square.

erates the geographic clusters seen in Figure 4.6 (c). For a first comparison, we train a Standard Self-Organizing Map (Algorithm 1) on the full data set (containing both the attribute and geographic features) using the same map topology, neighborhood relation, weight vector initialization, parameters, and learning rates as defined in Table 4.3. Recall that, in the Standard SOM formulation, we have = = 1. The resulting geographic clustering is depicted in Figure 4.6 (d).

At first glance, we observe notable differences in the geographic clustering obtained by the Tailored SOM and the Standard SOM (Figure 4.6 (c,d)). First, the Tailored SOM was able to correctly identify the four primary geographic groups that we specified during generation of the synthetic data. The Standard SOM, in contrast, split the geographic observations occupying the lower left quadrant (Figure 4.6 (d)) into two clusters. Secondly, the Tailored SOM was able to allocate three nodes (shown in cyan, navy blue, and medium green in Figure 4.6 (c)) to capture some of the geographic noise caused by our choice of P (< 1). For example, among the red geographic cluster (Figure 4.6 (c)), a cyan cluster consisting of three non-corresponding observations was identified. This result is useful as it demonstrates the algorithm's ability to discover observations that are proximate geographically, but dissimilar in attribute space, rather than lump such observations in with a larger, nearby cluster.

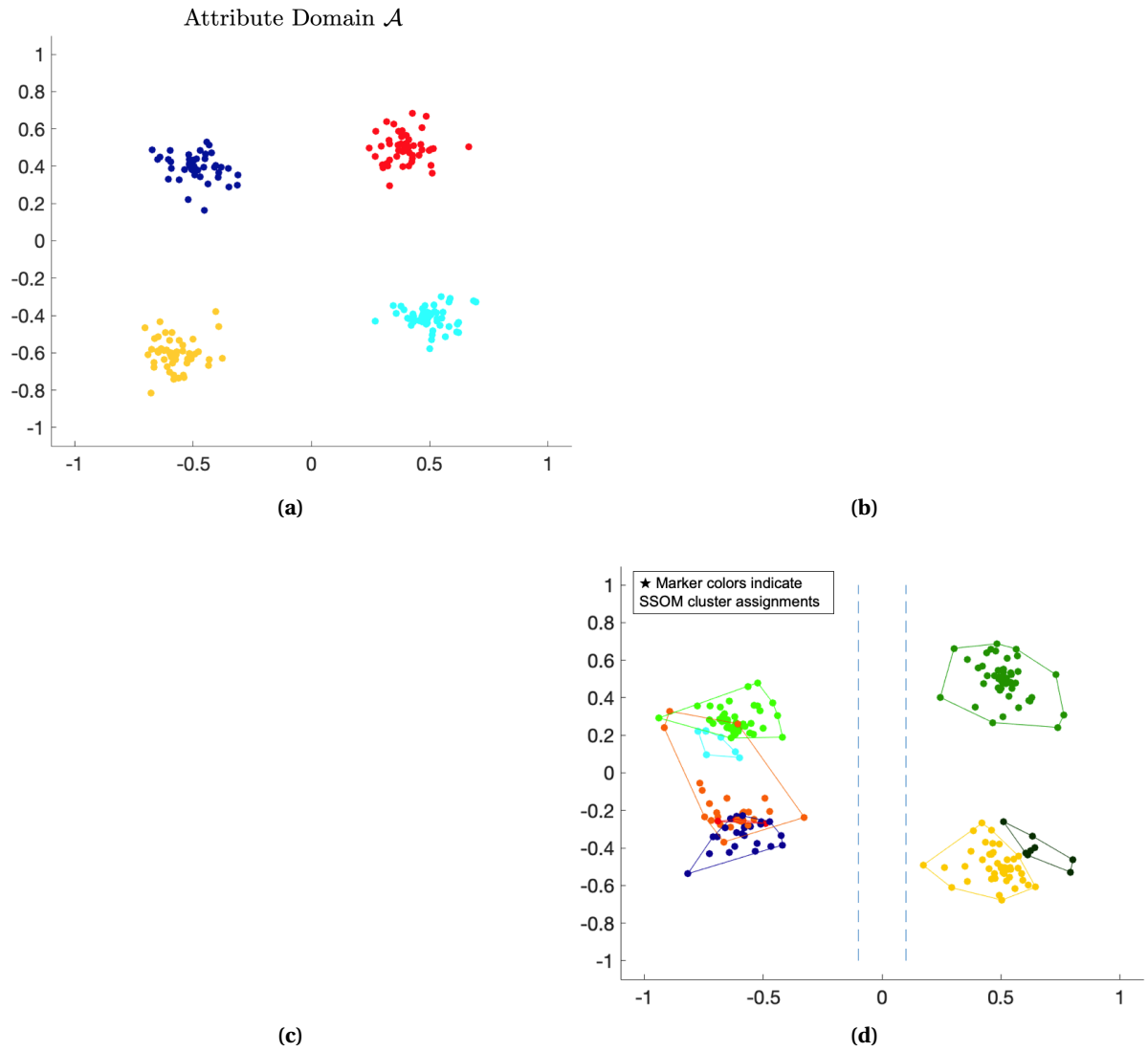
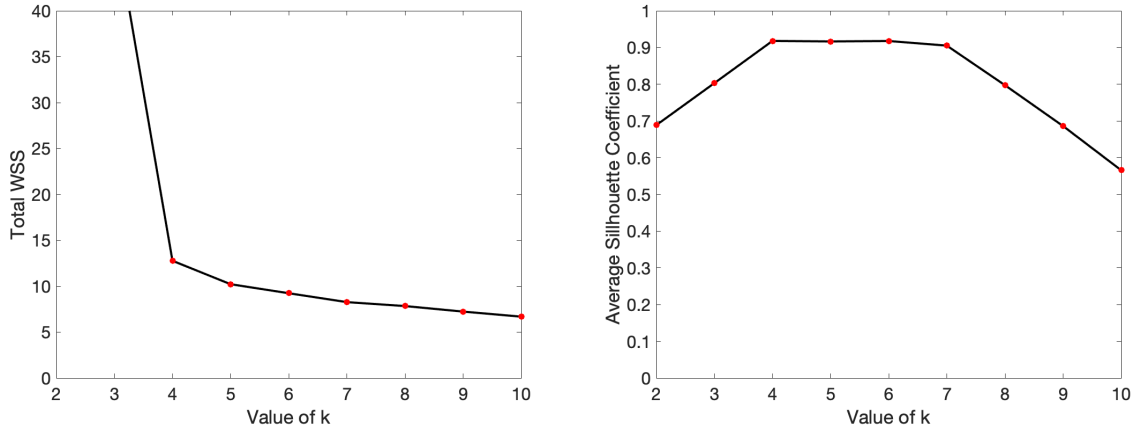


Figure 4.6 The geographic clusterings for the example Case 1 data set (shown in (a) and (b)) obtained from: (c) training the Tailored SOM algorithm with $q^* = [0.8, 1.1]$ and (d) training the Standard SOM algorithm. Post clustering, we apply the convex hull boundaries to each geographic cluster containing at least 2 data observations. For this data set, the Tailored SOM (Subfigure (c)) identified 7 geographic clusters, denoted by marker color, and the Standard SOM (Subfigure (d)) identified 8 geographic clusters, also denoted by marker color.

For a second clustering comparison, we implemented the k-Means algorithm initialized with the $k++$ approach. As detailed in Section 2.3, an important first step of this algorithm is deciding on an appropriate choice for k. To determine this, we implement both the Elbow Method [31] and the Average Silhouette Method [39] (Section 2.3.1) for the example Case 1 data set depicted in Figure 4.4. The results of these methods are illustrated in Figure 4.7. Recall that for the Elbow Method, we plot the within-cluster sum of squared distances (WSS) (equation (2.3)) as a function of k and

look for the “elbow", or the k-value for where the function begins to decrease linearly, for guidance on how to choose k. Determining the “elbow" is seldom a cut and dried task, as demonstrated by the results in Figure 4.7 (a). In this figure, one could argue that the elbow occurs at $k = 4$ or $k = 5$, depending on how strictly the "deceases linearly" property is enforced. The results obtained from the Average Silhouette Method, shown in Figure 4.7 (b), are similarly ambiguous. Recall that for the Average Silhouette Method, we seek to maximize the average silhouette coefficient (equation (2.4)) of a clustering, as a function of k. As illustrated in Figure 4.7 (b), $k = 4, 5, 6$, and 7 yield very similar average silhouette coefficients. Based on these results, we can conclude that there is no clear optimal choice for k.



(a) **(b)**

Figure 4.7 Results of the Elbow Method (shown in (a)) and the Average Silhouette Method (shown in (b)) for the Case 1 data set depicted in Figure 4.4. The Elbow Method relies on the total within-cluster sum of squared distances (WSS) (equation (2.3)) between data observations and their cluster centroid. The Elbow Method suggests choosing the value of k at which the WSS begins to decrease linearly, when plotted as a function of k. The Average Silhouette Method relies on the silhouette coefficient (S_i) (equation (2.4)) of each observation. The Average Silhouette Method suggests choosing the value of k that maximizes the average silhouette coefficient.

For comparative purposes, we train the k-Means algorithm for each k value recommended by either the Elbow or the Average Silhouette Method, i.e., $k = 4, 5, 6, 7$, and depict the geographic clustering results side-by-side in Figure 4.8. These results illustrate that the k-Means algorithm is less successful in identifying and separating the misplaced/non-corresponding observations in the geographic domain from nearby, well-clustered observations, even as we increase the number

of centroids (k). Speci cally, we observe this result as boundary intersections and substantial overlapping of geographic territory among the two primary clusters left of the river in each of the cases depicted in Figure 4.8. Additionally, Figure 4.8 illustrates the sensitivity of the clustering results to the prescribed value of k . A side-by-side comparison of the Tailored SOM geographic clustering and the k -Means algorithm geographic clustering for $k = 6$ is depicted in Figure 4.9. We make the note here that while the k -Means algorithm produced a geographically feasible result for this example data set (no cluster boundaries violate the vertical geographic constraint), over many realizations we notice that only a small percentage of k -Means results exhibit this quality. In the next section, we will provide a thorough discussion and systemic algorithm comparison based on results obtained by the Tailored SOM, Standard SOM, and k -Means algorithm.

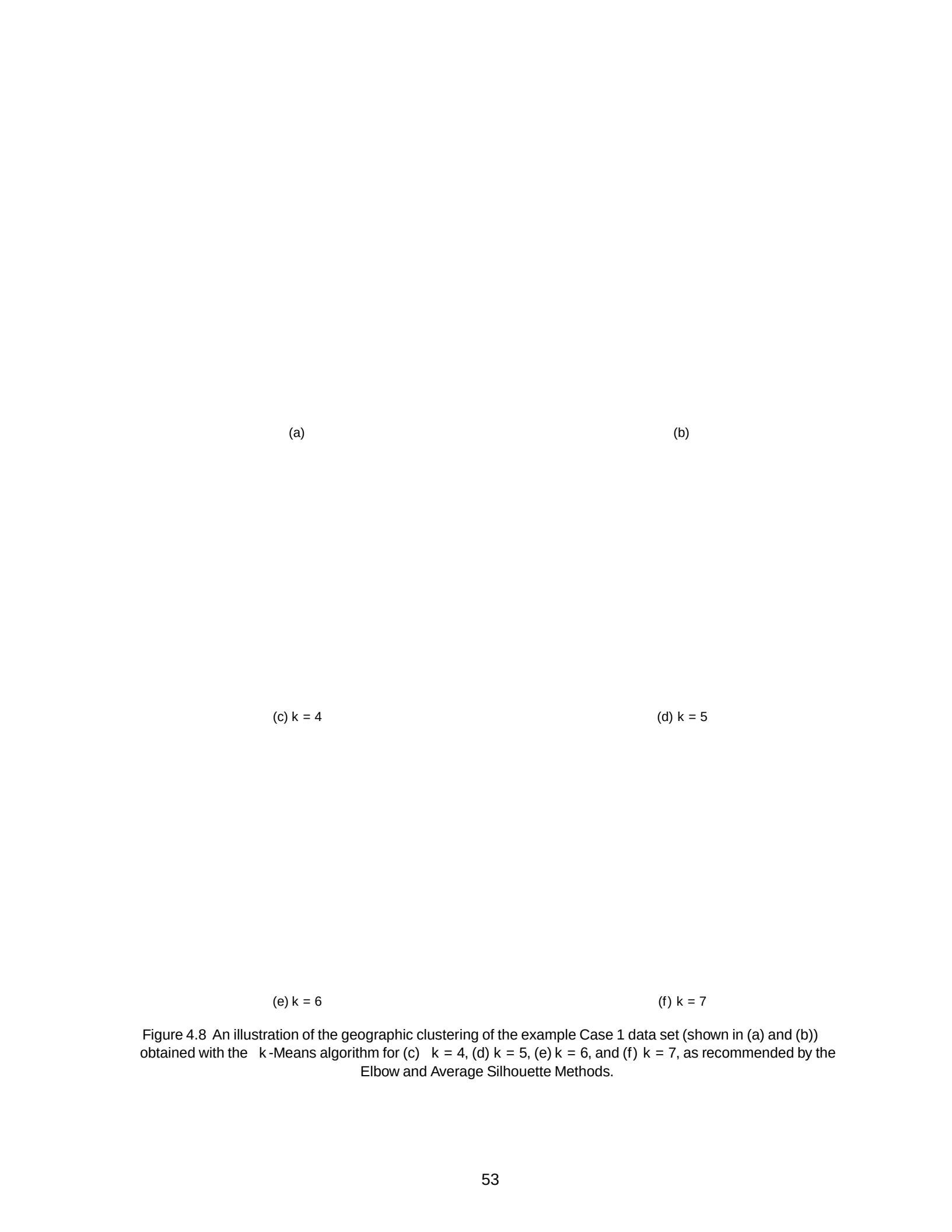


Figure 4.8 An illustration of the geographic clustering of the example Case 1 data set (shown in (a) and (b)) obtained with the k-Means algorithm for (c) $k = 4$, (d) $k = 5$, (e) $k = 6$, and (f) $k = 7$, as recommended by the Elbow and Average Silhouette Methods.

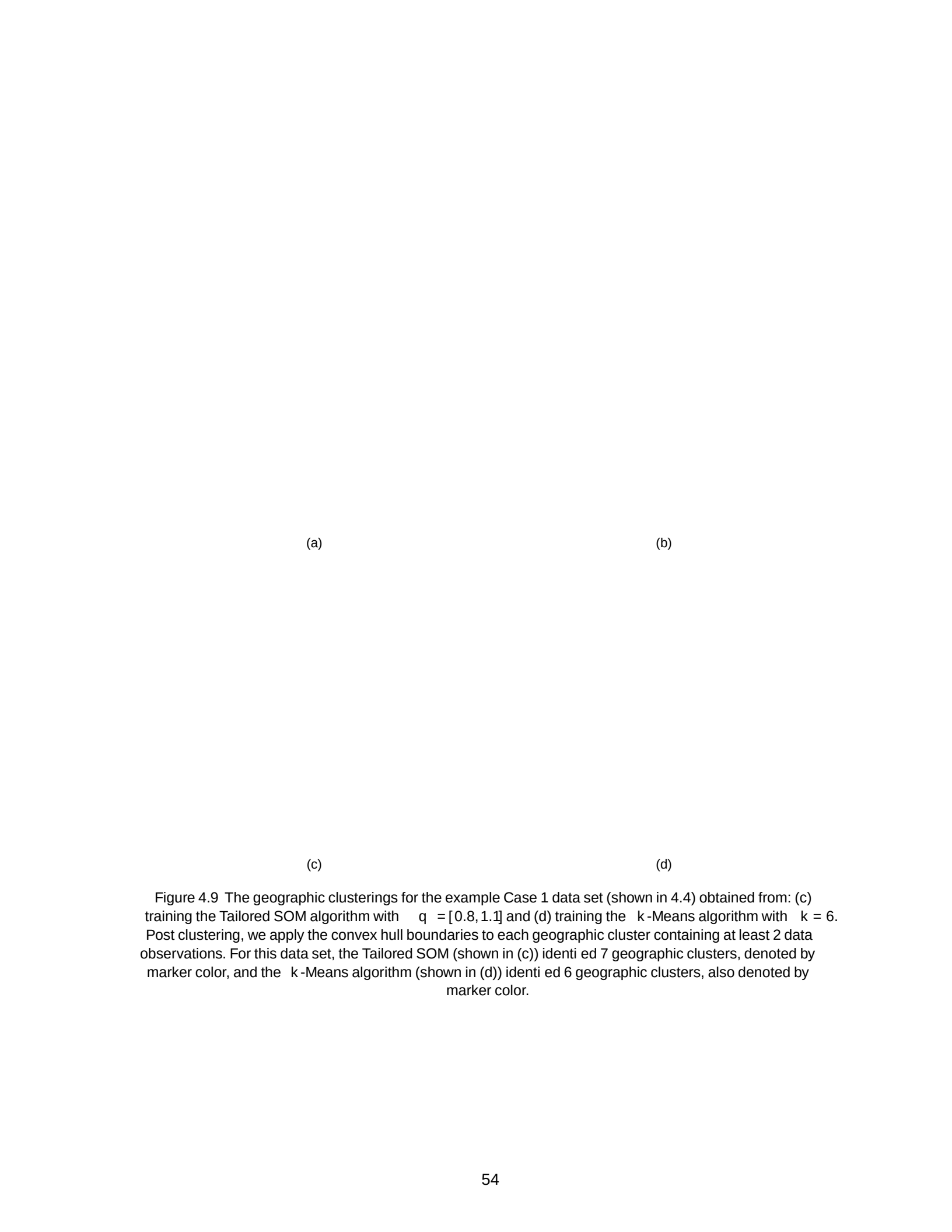


Figure 4.9 The geographic clusterings for the example Case 1 data set (shown in 4.4) obtained from: (c) training the Tailored SOM algorithm with $q = [0.8, 1.1]$ and (d) training the k-Means algorithm with $k = 6$. Post clustering, we apply the convex hull boundaries to each geographic cluster containing at least 2 data observations. For this data set, the Tailored SOM (shown in (c)) identified 7 geographic clusters, denoted by marker color, and the k-Means algorithm (shown in (d)) identified 6 geographic clusters, also denoted by marker color.

4.3.2 Algorithm Assessment: Case 1

While we can provide some discussion on the similarities and differences among the geographic clustering results achieved by the Tailored SOM (Figure 4.6 (a)), the Standard SOM (Figure 4.6 (b)), and the k-Means algorithms (Figure 4.8), it is important to have a consistent method for assessing and comparing the performance of each algorithm. In particular, we wish to assess the output of each algorithm with respect to the overall goals of the dual-domain clustering problem. First, we can evaluate algorithm performance and the quality of a clustering using the geographic objective function J_g (equation (3.1)). For this, we compute the value of J_g for clusterings obtained by each algorithm and deem the results that correspond with lower J_g values as more desirable. Additionally, as discussed in Section 1.2, we can also design performance metrics that rely on our knowledge of how the synthetic data sets were created. In other words, we can use our knowledge of the underlying distributions, various levels of noise within the data, and intended cluster arrangements in A and G that we prescribe when generating the synthetic data to design and score results against the notion of a "desired answer".

To that end, we propose two metrics for algorithm assessment that are based on the synthetic data generation: the *Stable Score* and the *Inlier Score*, defined in Table 4.4. The computation of these scores depends on a concept we call the "stable" groups. The stable groups are made up of the essential (non-outlying) observations of the geographic groups specified during synthetic data generation. As detailed in Step 1 of Table 4.4, the stable groups are constructed through an iterative process that aims to determine the core set of geographic observations in a known geographic group, and, ideally, are identified during clustering. An example of how we construct a stable group for a known geographic group from the example Case 1 data set (Figure 4.4) is depicted in Figure 4.10.

Based on the concept of stable groups, we define the Stable Score (Step 3, Table 4.4) as the percentage of the stable group's population represented by its best-matching cluster from the TSOM/SSOM/k-Means output. Here, we define the best-matching cluster for a stable group as the one with which it shares the largest intersection. We use the Stable Score as a way to quantify how well a clustering algorithm achieved the results intended during data generation. However, this formulation does not penalize a cluster for containing outlying observations, i.e., observations we

(a) (b)

Figure 4.10 A depiction of the first iteration of the stable group construction for a known geographic group. In the example Case 1 data set shown in (a), we consider all observations associated with Attribute Group 4 (cyan). In (b) we depict the centroid of the cyan observations with a black `x' and 2.5-standard deviations (radially) from the centroid with a black dashed circle. We identify 4 observations that fall outside this circle (each noted with a black arrow. Each of these outlying observations would be removed from the cyan stable group at this iteration of the construction process. This process repeats by updating the centroid and the 2.5-standard deviation boundary, and removing outliers until the population of the stable group no longer changes. Once complete, the stable group is considered the set of core observations for that particular geographic group and can be used in computing the Stable and Inlier Scores (defined in Table 4.4).

know to belong to a different geographic group. For example, the Stable Score would not penalize a cluster for capturing the red observations that are mixed in with the cyan geographic group seen in Figure 4.10 (b). To address this issue, we define the Inlier Score (Step 4, Table 4.4). The Inlier Score for a given cluster is the percentage of its population that is shared with its best-matching stable group. By considering each of these scores side by side, or their average, we are able to quantify how well a given clustering captures the intended observations / geographic groupings without over-simplifying its representation of the data by co-clustering multiple observations from distinct groupings.

To demonstrate the usage of these synthetic data based scoring techniques and how we can evaluate J_g to assess algorithm performance, we refer back to the Case 1 data set depicted in Figure 4.4. Recall that for this data set we have obtained clustering results via the Tailored SOM (Figure 4.6 (c)), the Standard SOM (Figure 4.6 (d)) (abbreviated as SSOM in subsequent figures and tables), and k-Means (Figure 4.8 (d)). For comparative purposes, we will focus on the k-Means result for $k = 6$ (Figure 4.8 (c)). For each result, we determine cluster boundaries using the convex hulls (Section 3.2)

Table 4.4 Stable Score and Inlier Score Algorithm

Algorithm:

1. After generation of a synthetic data set, determine “stable" groups in the geographic domain G.
   (a) For each of the known geographic groups, compute the centroid and standard deviation of its observations.
   (b) Remove all observations that fall outside n-standard deviations of the centroid; these observations are labeled as outliers.
   (c) Update centroids and standard deviations using remaining observations.
   (d) Repeat 1(b) and (c) until the group population is stable.
2. For each stable group identified, determine its best-matching cluster from the TSOM/ SOM/ k-Means Algorithm output.
3. For each best-matching pair, the cluster's Stable Score is defined as

$$\bullet \frac{|\text{best-matching cluster} \setminus \text{stable group}|}{|\text{stable group}|} \quad 100$$

   where $|\cdot|$ is the cardinality of the enclosed set.
4. For each best-matching pair, the cluster's Inlier Score is defined as

$$\bullet \frac{|\text{best-matching cluster} \setminus \text{stable group}|}{|\text{best-matching cluster}|} \quad 100$$

   where $|\cdot|$ is the cardinality of the enclosed set.

and compute the value of J_g (equation (3.1)) given a circular representative perimeter ($\bar{p}$) with radius R = 0.3. Recall that we constructed J_g such that a lower value reflects a more desirable geographic clustering. Hence, when we compare J_g values for the clusterings produced by each algorithm, we consider the “best" result to be the one that corresponds with the lowest value. Based on the results depicted in Table 4.5, the clustering identified by Tailored SOM is more desirable than those produced by the Standard SOM and the k-Means algorithms (for $k = 6$) in terms of the geographic qualities addressed by J_g (Section 3.2). Given that the clustering obtained by the Tailored SOM fundamentally relied on the minimization of J_g during training, this result is somewhat expected. Nonetheless, J_g is a valuable metric for algorithm evaluation and comparison as it highlights the important differences among results obtained by a standard approach versus one that was tailored to the challenges of dual-domain data.

Table 4.5 Values of the geographic objective function J_g (equation (3.1)) for the clusterings determined by the Tailored SOM, the Standard SOM, and the k-Means algorithm ($k = 6$) on the example Case 1 data set depicted in Figure 4.4.

	Tailored SOM	Standard SOM	k-Means
Value of J_g	0.9192	1.5661	2.0000

Given the data generation parameters defined in Table 4.2, we know there to be four primary geographic groups within the data set. Hence, we can score the results obtained from each clustering algorithm based on four stable groups determined with $n = 2.5$ standard deviations (Step 1(b), Table 4.4). For each of these stable groups, we compute the Stable Score and the Inlier Score for each clustering algorithm, along with the mean Stable Score and Inlier Score across all four stable groups (Table 4.6). While in some cases it may be beneficial to compare the Stable and Inlier Scores of individual clusters across algorithms, considering their means and medians (last four columns of Table 4.6) is a useful way of assessing the clustering as a whole. Based on the summary statistics reported in the last four columns of Table 4.6, we can conclude that the Tailored SOM outperformed the Standard SOM and the k-Means algorithms in both metrics (Stable and Inlier Scores). In terms of the individual clusters, the lowest Stable Score obtained by the Tailored SOM was 97.9%, the Standard SOM was 51.2%, and the k-Means algorithm was 50.0%. Furthermore, the lowest Inlier

Score for an individual cluster obtained by the Tailored SOM was 86.0%, the Standard SOM was 75.9%, and the k-Means algorithms was 81.2%. These results suggest that, due to the additional signi cance placed on the geographic domain (since > for this example), the Tailored SOM is able to identify the intended geographic groups to a higher degree of accuracy than the Standard SOM and the k-Means algorithm.

Table 4.6 Stable Scores (SS) and Inlier Scores (IS) for the clustering of the example Case 1 data set depicted in Figure 4.4 obtained by the Tailored SOM (Figure 4.6 (c)), the Standard SOM (Figure 4.6 (d)), and k-Means for $k = 6$ (Figure 4.9 (d)). Each of the “Cluster" columns contains the scores for the best-matching cluster to one of the 4 stable groups determined for this example data. The “Mean" column contains the mean Stable Score and mean Inlier Score across all four stable groups for each algorithm. The “Median" column contains the median Stable Score and median Inlier Score across all four stable groups for each algorithm.

	Cluster 1		Cluster 2		Cluster 3		Cluster 4		Mean		Median	
	SS	IS	SS	IS	SS	IS	SS	IS	SS	IS	SS	IS
TSOM (%)	100	87.0	97.9	97.9	100	95.8	100	86.0	99.5	91.7	100	91.4
SSOM (%)	100	87.0	83.3	95.2	84.8	95.1	51.2	75.9	79.8	88.3	84.1	91.0
k-Means (%)	100	87.0	100	96.0	50.0	85.2	100	81.2	87.5	87.3	100	86.1

To validate these results, we repeat the experiment outlined by the above example by generating 20 realizations of Case 1 synthetic data sets according to the parameter de nitions in Table 4.2. For each realization we optimize the Tailored SOM using the parameters de ned in Table 4.3 and the objective function J_g , (equation (3.1)) over the admissible parameter space (de ned by equation (4.1)) to determine an optimal q (equation (3.2)). Over all realizations, the mean optimal parameter set is $\bar{q} = [\bar{\ }, \bar{\ }] = [0.793, 1.180]$ and the median parameter set is $\tilde{q} = [\tilde{\ }, \tilde{\ }] = [0.750, 1.20]$. Furthermore, in each of the 20 optimal parameter sets determined for the Case 1 data realizations, we have < . Additionally, we train a Standard SOM on the full data set using the parameters de ned in Table 4.3 and the k-Means algorithm with optimal k as identi ed by the Average Silhouette Method.

First, for each realization and clustering algorithm, we compute the value of J_g (given a circular representative perimeter with radius $R = 0.3$) and, by inspection, determine whether each resulting clustering is geographically feasible. We de ne a geographically feasible clustering as one that does

not result in any of the convex boundaries crossing over the vertical geographic constraint (river) depicted in Figure 4.4 (b). For each algorithm tested, we compute the percentage of realizations for which a geographically feasible clustering was achieved and record the results in Table 4.7. An example of a geographically unfeasible clustering produced by the k-Means algorithm is depicted in Figure 4.11.

Table 4.7 The percentage of Case 1 realizations for which a geographically feasible clustering was achieved by the Tailored SOM, the Standard SOM, and the k-Means algorithm.

	Tailored SOM	Standard SOM	k-Means
Geographically Feasible (%)	85%	45%	30%

Figure 4.11 An example of a geographically unfeasible clustering produced by the k-Means algorithm (with $k = 5$) on a Case 1 synthetic data set. We make this determination based on how the convex boundaries of several geographic clusters span across the vertical constraint.

We depict the box plots of J_g values for each algorithm over all Case 1 data realizations in Figure 4.12. We implement the Kruskal-Wallis test to determine the validity of the null hypothesis that the J_g values obtained by each algorithm (TSOM, SSOM, k-Means) come from the same distribution [21]. This test returns a p-value of $8.16 \quad 10^{-5}$, indicating that we should reject the null hypothesis and accept the alternative hypothesis that not all J_g values come from the same distribution. Based

on this result, we perform Dunn's multiple comparison test [10]. This test investigates which pairwise comparisons are signi cantly different (in the statistical sense), i.e., which pair(s) of algorithms produce clusterings that have J_g values that are unlikely to have come from the same distribution. The results of Dunn's test are depicted in Table 4.8. Based on these results, we can conclude that the J_g values obtained by the Tailored SOM are signi cantly different and lower than those obtained by the Standard SOM and the k-Means algorithm.

Figure 4.12 The box plots of the J_g values obtained by each algorithm: the Tailored SOM (TSOM), the Standard SOM (SSOM), and k-Means for all realizations of Case 1 data.

Table 4.8 The results of performing Dunn's multiple comparison test (as a follow-up to the Kruskal-Wallis test) to test for statistically signi cant differences among the distributions of J_g values obtained by the Tailored SOM, the Standard SOM, and the k algorithm.

	p-value
TSOM vs. SSOM	0.0001
TSOM vs. k-Means	0.0003
SSOM vs. k-Means	0.9859

For the purpose of further comparison and visualization, we sort and plot the values of J_g

obtained by each algorithm in ascending order and use the marker colors to indicate the geographic feasibility of the clustering. These plots, which compare the results for the TSOM versus SSOM and for the TSOM versus k-Means, are depicted in Figure 4.13 (a) and (b), respectively. In each comparison, the solid plotted line corresponds to the results obtained from the Tailored SOM and the dashed line corresponds to the results obtained by the Standard SOM (Figure 4.13(a)) or the k-Means algorithm (Figure 4.13 (b)). A blue marker indicates a geographically feasible result and a red marker indicates a geographically unfeasible result. The solid plotted line corresponds to the results obtained from the Tailored SOM and the dashed line corresponds to the results obtained by the Standard SOM (Figure 4.13(a)) or the k-Means algorithm (Figure 4.13 (b)). Based on these results (depicted in Figure 4.13 (a,b)), we can conclude that the Tailored SOM clustering outperforms those obtained by the Standard SOM and the k-Means algorithm in all realizations of the Case 1 data. Additionally, we note the correlation between realizations that have the largest values of J_g and are geographically unfeasible.

(a) (b)

Figure 4.13 A depiction of the values of J_g obtained for each realization of the Case 1 data by (a): the Tailored SOM versus the Standard SOM and (b): the Tailored SOM versus the k-Means algorithm. For each comparison, we sort the function values in ascending order and use the marker colors to indicate the geographic feasibility of each realization. A blue marker indicates a geographically feasible result and a red marker indicates a geographically unfeasible result. In each plot, the solid plotted line corresponds to the results obtained from the Tailored SOM and the dashed line corresponds to the results obtained by the Standard SOM (a) or the k-Means algorithm (b).

Secondly, we compute the mean Stable Score and the mean Inlier Score for each clustering

across all 20 realizations. We record these scoring results, organized by realization number and scoring metric, along with geographic feasibility (Table 4.9). We illustrate these results and compare the performance of each algorithm with a depiction of the difference in mean Stable and Inlier Scores for the Tailored SOM versus the Standard SOM (Figure 4.14), and the k-Means algorithm (Figure 4.15). In each of these gures, the solid blue line indicates no difference between mean scores, the solid red line indicates the mean score difference across all realizations, and the dashed red lines represent 1 standard deviation of the mean score differences.

From the results depicted in Figure 4.14 we conclude that, on average, the Tailored SOM (TSOM) achieves a higher mean Stable Score (Figure 4.14 (a)) and an approximately equal Inlier Score (Figure 4.14 (b)) as compared to the Standard SOM (SSOM). This suggests that, for Case 1 data, the Tailored SOM is better suited to identify the intended geographic groups than the Standard SOM, without over simplifying the clustering and grouping together dissimilar (outlying) observations. It is somewhat expected that these two algorithms have an approximately equal mean Inlier score, given that each was trained on the same map structure with an equivalent number of nodes. From the results depicted in Figure 4.15, we can conclude that, while on average, the mean Stable Score for the Tailored SOM is lower than that of k-Means, the Tailored SOM achieves a Stable Score approximately equal to, or greater than, the k-Means algorithm in 55% of all realizations (Figure 4.15 (a)). Additionally, from the Inlier Score comparison (Figure 4.15 (b)), we can conclude that the Tailored SOM outperforms the k-Means algorithm in its ability to isolate the intended geographic groups, rather than simply capture them within a larger grouping, alongside outliers. This result is signi cant in that it shows the Tailored SOM is better suited to identify a clustering which can shed light on the more nuanced groupings within a data set.

Table 4.9 Mean Stable Score, Mean Inlier Score, and Geographic Feasibility results for the Tailored SOM (TSOM), Standard SOM (SSOM), and k-Means Algorithm for 20 realizations of Case 1 synthetic data. Mean Stable Scores and Mean Inlier Scores were computed based on four stable groups determined with $n = 2.5$ standard deviations (Step 1(b), Table 4.4). In the "Geographically Feasible?" columns, 'Y' indicates the result is geographically feasible and 'N' indicates the result is not geographically feasible.

	Stable Score			Inlier Score			Geographically Feasible?		
Realization	TSOM	SSOM	k-Means	TSOM	SSOM	k-Means	TSOM	SSOM	k-Means
1	100%	91.4%	91.7%	92.9%	96.0%	94.3%	Y	N	Y
2	100%	95.1%	88.8%	92.7%	92.9%	89.4%	Y	Y	Y
3	98.5%	98.2%	100%	92.1%	92.0%	89.4%	Y	N	N
4	92.4%	97.2%	100%	93.9%	93.6%	70.6%	Y	N	Y
5	100%	95.7%	100%	95.8%	95.4%	69.6%	Y	Y	Y
6	89.5%	96.4%	100%	93.8%	91.0%	86.5%	N	N	N
7	98.4%	80.8%	91.5%	93.4%	95.9%	93.9%	Y	Y	Y
8	99.5%	93.5%	89.7%	93.9%	95.5%	95.5%	Y	N	N
9	97.5%	96.9%	94.4%	96.3%	97.9%	94.1%	Y	Y	N
10	91.6%	95.2%	100%	94.1%	95.0%	93.3%	Y	N	N
11	96.5%	90.5%	100%	90.8%	88.0%	85.1%	Y	N	N
12	100%	89.1%	92.9%	91.0%	92.1%	86.4%	Y	Y	Y
13	100%	98.8%	100%	90.1%	88.6%	85.6%	N	N	N
14	89.2%	96.7%	99.4%	90.9%	93.0%	91.1%	Y	Y	N
15	83.1%	64.7%	82.3%	91.4%	89.8%	89.3%	Y	N	N
16	81.1%	96.9%	100%	93.5%	92.6%	92.7%	N	Y	N
17	91.9%	86.4%	100%	93.0%	92.6%	89.5%	Y	N	N
18	100%	96.1%	100%	88.1%	90.8%	89.7%	Y	Y	N
19	100%	97.3%	100%	95.8%	95.8%	92.8%	Y	Y	N
20	97.1%	94.8%	100%	91.7%	88.2%	85.9%	Y	N	N

(a) TSOM versus SSOM mean Stable Score differences

(b) TSOM versus SSOM mean Inlier Score differences

Figure 4.14 For each realization of Case 1 data, we compute the difference in mean Stable Score (shown in (a)) and mean Inlier Score (shown in (b)) achieved by the optimized Tailored SOM (TSOM) versus the Standard SOM (SSOM). A positive score difference indicates a realization for which the TSOM achieved a higher score than the SSOM. Additionally, we plot the mean score difference across all realizations (solid red line) and 1 standard deviation of the mean score differences (dashed red lines). The solid blue line indicates no difference in the mean scores.

(a) TSOM versus k-Means mean Stable Score differences

(b) TSOM versus k-Means mean Inlier Score differences

Figure 4.15 For each realization of Case 1 data, we compute the difference in mean Stable Score (shown in (a)) and mean Inlier Score (shown in (b)) achieved by the optimized Tailored SOM (TSOM) versus the k-Means algorithm. A positive score difference indicates a realization for which the TSOM achieved a higher score than the k-Means algorithm. Additionally, we plot the mean score difference across all realizations (solid red line) and 1 standard deviation of the mean score differences (dashed red lines). The solid blue line indicates no difference in the mean scores.

4.4 Synthetic Data: Case 2

For a second case, we manufacture a synthetic data set similar to that created for Case 1, with an equal number of attribute and geographic groups ($p_a = p_g$), as described in Section 4.2. However, we choose to increase the attribute domain radial variance ($^2_1$) and decrease the association probability P. For this second case, we are interested in investigating how these levels of noise will affect or impede the performance of each clustering algorithm. By holding the attribute and geographic group centroids (M_a and M_g) constant and increasing the attribute domain radial variance ($^2_1$), we create attribute groups that are less well-separated and may contain outlying observations that mix in with a nearby group. Additionally, by decreasing the association probability P, it is more likely that a larger number of attribute observations will be associated with a non-corresponding geographic group. We de ne values for these parameters, along with the other parameters related to synthetic data generation, in Table 4.10. Furthermore, we continue to implement a vertical geographic constraint in the geographic domain G. We depict a sample Case 2 data set, generated from the parameters de ned in Table 4.10, in Figure 4.16. As previously done, the marker colors of the geographic observations (Figure 4.16 (b)) indicate their correspondence with an attribute group (Figure 4.16 (a)).

Table 4.10 Parameter values used to manufacture the second case of synthetic data.

Parameter	Value	De nition
p_a	4	number of groups in A
p_g	4	number of groups in G
D	200	total number of observations in the data set
σ_1^2	0.15^2	radial variance for observations in A
σ_2^2	0.15^2	radial variance for observations in G
P	0.85	association probability
M_a	$\begin{bmatrix} 2 & 3 \\ 0.4 & 0.5 \\ 0.4 & 0.4 \\ 0.6 & 0.5 \\ 0.6 & 0.3 \end{bmatrix}^T$	group centroids in A
M_g	$\begin{bmatrix} 2 & 3 \\ 0.5 & 0.5 \\ 0.4 & 0.5 \\ 0.6 & 0.6 \\ 0.35 & 0.35 \end{bmatrix}^T$	group centroids in G

(a) (b)

Figure 4.16 A sample Case 2 data set generated from the parameter values de ned in Table 4.10 projected into the two-dimensional attribute domain (shown in (a)) and the two-dimensional geographic domain (shown in (b)). The marker colors in each sub gure are used to indicate the inter-domain correspondence between the geographic observations and the attribute observations.

4.4.1 Tailored versus Standard Clustering Results: Case 2

For the Case 2 data, we maintain many of the same Tailored SOM design choices instituted for the Case 1 data. We implement a $3 \quad 3$ map, consisting of $N = 9$ nodes, with a neighborhood relation de ned by the structure of the rectangular lattice. However, one notable difference is our method of weight vector initialization. In Section 4.3.1, we evenly spaced the initial weight vectors for each domain in $[\ 1,1] \quad [\ 1,1]$. For Case 2 data, we will experiment with a pseudo-random initialization of the weight vectors. For this initialization, we randomly generate $N \ 4$ dimensional weight vectors from the continuous uniform distribution on $(\ 1,1)$. For each of the N initial weight vectors, we compute the closest data observation across the full feature set according to Euclidean distance, and move the weight vector to be positioned at its location in data space. In contrast with the weight vector initialization regime discussed for Case 1, this approach provides the clustering algorithm with some preliminary knowledge of the data landscape prior to the start of training. For example, by updating each random weight vector initialization to share the same location as an observation in the data set, we are preventing the situation where weight vectors begin in an unpopulated region of one or both domains. We summarize all parameters and design choices used for the Tailored SOM used for Case 2 synthetic data in Table 4.11.

Table 4.11 Parameters and design choices prescribed for training the Tailored SOM on Case 2 synthetic data.

Parameter	Value	De nition
N	9	total number of nodes in SOM (given by 3 3 rectangular lattice structure)
initial $\vec{w}_j^a$ (j = 1,..,N)	pseudo-random	weight vector initialization in A
initial $\vec{w}_j^g$ (j = 1,..,N)	pseudo-random	weight vector initialization in G
$\vec{w}_n$	1 edge away	neighborhood relation among nodes
z(k)	$\frac{1}{2}^{\frac{k}{k_{max}}}$	learning rate for winning weight vectors
(k)	$\frac{1}{8}^{\frac{k}{k_{max}}}$	learning rate for neighboring weight vectors
k_{max}	9	number of training batches

For the optimization of J_g , , we maintain the same admissible parameter space (equation (4.1)) and representative perimeter, $\bar{p}$, that were de ned for Case 1. To determine an optimal q = [,] (equation (3.2)), we begin with the global grid search of the parameter space via the step size $q = [0.05, 0.05]$. For the sample Case 2 data set depicted in Figure 4.16, we determine the optimal $q = [0.75, 1.1]$ via the global grid search. We implement this optimal parameter set in the Tailored Self-Organizing Map (speci ed by the parameters in Table 4.11), and this result generates the geographic clustering seen in Figure 4.17 (c).

To evaluate and compare, we repeat the same procedures as outlined for Case 1. First, we train a Standard Self-Organizing Map (Algorithm 1) with the map topology, neighborhood relation, pseudo-random weight vector initialization, and parameters de ned in Table 4.11. The resulting geographic clustering is depicted in Figure 4.17 (d). Evident from this comparison is the difference in the geographic features of each clustering. While the Tailored SOM clustering is capable of identifying the four primary groupings, the Standard SOM splits the group in the upper left portion of the geographic domain into two clusters. Additionally, the Standard SOM fails to prioritize geographic proximity and identi es a cluster (shown in orange in Figure 4.17 (d)) that violates the vertical constraint. Additionally, the Tailored SOM more appropriately clusters non-corresponding geographic observations within larger primary groups based on their true attribute group (true attribute groups shown in Figure 4.16 (b)). In contrast, the Standard SOM assigns many non-corresponding

geographic observations from across different regions of the domain into a single cluster, focusing too much on their attribute proximity.

(a) (b)

(c) (d)

Figure 4.17 The geographic clusterings obtained for the example Case 2 data set (shown in (a), (b)) from training the Tailored SOM algorithm with $q = [0.75, 1.1]$ (shown in (c)) and the Standard SOM algorithm (shown in (d)). Post clustering, we apply the convex hull boundaries to each geographic cluster containing at least 2 data observations. For this data set, the Tailored SOM (shown in (c)) identified 9 geographic clusters, denoted by marker color, and the Standard SOM (shown in (d)) identified 9 geographic clusters, also denoted by marker color.

Secondly, we implemented the k-Means algorithm initialized with the $k++$ approach and with an optimal value of $k = 7$, determined via the Average Silhouette Method (Figure 4.18). Similar

to the results depicted in Figure 4.7, the average silhouette coef cients are quite close for $k = 7, 8, 9$. However, for the sake of automation, we will continue with the value of k for which the function attains its true maximum, $k = 7$. A side-by-side comparison of the Tailored SOM geographic clustering and the k-Means algorithm geographic clustering for $k = 7$ is depicted in Figure 4.19. The results obtained from k-Means are most similar to those obtained from the Standard SOM, with the exception of geographic feasibility, i.e., no cluster boundaries cross the vertical geographic constraint. Based on these results, we note that the k-Means algorithm has dif culty managing the additional geographic noise generated from our choice of the association probability, $P = 0.85$, as demonstrated by the overlapping of several large clusters on each side of the geographic constraint, i.e., east and west of the "river".

Figure 4.18 Average Silhouette Method results for the Case 2 data set depicted in Figure 4.16. This method relies on the silhouette coef cient (S_i) (equation (2.4)) of each observation. The Average Silhouette Method suggests choosing the value of k that maximizes the average silhouette coef cient. In this example, the optimal value is $k = 7$.

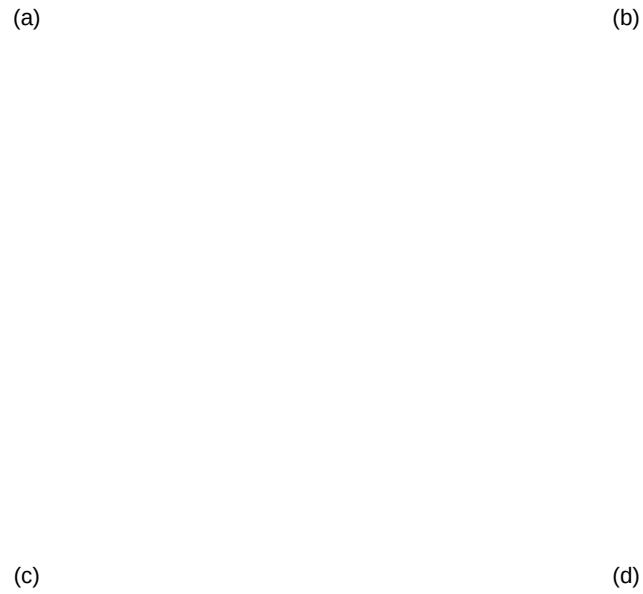


Figure 4.19 The geographic clusterings obtained for the example Case 2 data set (shown in (a,b)) from training the Tailored SOM algorithm with $q = [0.75, 1.1]$ (shown in (c)) and the k-Means algorithm with $k = 7$ (shown in (d)). Post clustering, we apply the convex hull boundaries to each geographic cluster containing at least 2 data observations.

4.4.2 Algorithm Assessment: Case 2

To compare the geographic clusterings obtained from the Tailored Self-Organizing Map, Standard Self-Organizing Map, and k-Means, depicted in Figures 4.17 and 4.19, we refer back to the geographic objective function J_g (equation (3.1)), and the Stable and Inlier Scores (Table 4.4). For this single example of a Case 2 data set (Figure 4.16), we compute the values of J_g (Table 4.12), and the mean and median Stable and Inlier Score for each algorithm along with the Geographic Feasibility (Table

4.13). From the J_g evaluation results (Table 4.12), we can observe a signi cant difference between the performance of the tailored algorithm and the two standard approaches. Speci cally, the two standard approaches achieve a very similar J_g value that is nearly twice that achieved by the Tailored SOM.

Table 4.12 Values of the geographic objective function J_g (equation (3.1)) for the clusterings determined by the Tailored SOM, the Standard SOM, and the k-Means algorithm ($k = 7$) on the example Case 2 data set depicted in Figure 4.16.

	Tailored SOM	Standard SOM	k-Means
Value of J_g	2.1112	4.1290	4.0966

The Stable and Inlier Score results (Table 4.13) suggest that the Tailored SOM outperformed the Standard SOM considerably, and slightly outperformed the k-Means algorithm in one or more of the synthetic data based metrics (in terms of the summary statistics). Not noted in Table 4.13, but easily observable in Figure 4.19, is the difference in the degree of delineation, or segregation, among the primary clusters identi ed by the Tailored SOM and k-Means algorithm, respectively. This difference can be quanti ed by considering the total amount of overlapping area among the geographic clusters produced by each algorithm. This is worth noting as it highlights the concept of geographic "usefulness" that we have previously discussed. Hence, while the TSOM and k-Means' performances appear to be quite close (according to results in Table 4.13), it is reasonable to conclude that the result produced by the TSOM is favorable, from a holistic standpoint, as demonstrated by the comparison of J_g values (Table 4.12).

To validate these results, we repeat the experiment with an additional 20 realizations of Case 2 synthetic data sets according to the parameter de nitions in Table 4.10. For each realization, we optimize the Tailored SOM with the parameters speci ed in Table 4.11 and the objective function J_g , (equation (3.1)) over the admissible parameter space (equation (4.1)) to determine the optimal q (equation (3.2)). Over all realizations, the mean optimal parameter set is $\bar{q} = [\bar{\ }, \bar{\ }] = [0.793, 1.180]$ and the median parameter set is $\tilde{q} = [\tilde{\ }, \tilde{\ }] = [0.750, 1.225]$. Furthermore, in each of the 20 optimal parameter sets determined for the Case 2 data realizations, we have $<$. We

Table 4.13 Stable Scores (SS) and Inlier Scores (IS) for the clustering of the example Case 2 data set depicted in Figure 4.16 obtained by the Tailored SOM (Figure 4.17 (c)), the Standard SOM (Figure 4.17 (d)), and k-Means for $k = 7$ (Figure 4.19 (d)). Each of the “Cluster" columns contains the scores for the best-matching cluster to one of the 4 stable groups determined for this example data. The “Mean" column contains the mean Stable Score and mean Inlier Score across all four stable groups for each algorithm. The “Median" column contains the median Stable Score and median Inlier Score across all four stable groups for each algorithm. The red text color is used to denote an algorithm that produced a geographically unfeasible clustering for this example Case 2 data set.

	Cluster 1		Cluster 2		Cluster 3		Cluster 4		Mean		Median	
	SS	IS	SS	IS	SS	IS	SS	IS	SS	IS	SS	IS
TSOM (%)	100	95.6	100	82.9	100	87.5	100	93.0	100	89.8	100	90.3
SSOM (%)	100	95.6	97.1	83.5	57.1	71.4	95.0	97.4	87.3	86.7	96.0	89.0
k-Means (%)	100	95.6	100	87.2	100	83.3	100	88.9	100	88.7	100	88.0

maintain the same training parameters for the Standard SOM as used for the Tailored SOM and, for k-Means, we rely on the optimal k determined by the Average Silhouette Method.

We begin the discussion of comparative algorithm performance with the concept of geographic feasibility. For the Case 2 data, a geographically feasible clustering was achieved by the Tailored Self-Organizing in 80% of realizations, by the Standard SOM in 50% of realizations, and by the k-Means algorithm in 45% of realizations (Table 4.14). The relative ranking of algorithm performance with respect to this assessment method for the Case 2 data is consistent with that of the Case 1 data. In both cases, the Tailored SOM signi cantly outperformed the other two algorithms in producing clusterings that are mindful of geographic constraints and dense in the sense that the clusters do not span large sections of the geographic domain.

Table 4.14 The percentage of Case 2 realizations for which a geographically feasible clustering was achieved by the Tailored SOM, the Standard SOM, and the k-Means algorithm.

	Tailored SOM	Standard SOM	k-Means
Geographically Feasible (%)	80%	50%	45%

Next, for each realization and clustering algorithm, we compute the value of the geographic objective function J_g. We depict the box plots of these values for each algorithm over all 20 Case 2

realizations in Figure 4.20. Given the normality and homogeneity of the variances of these results, we implement a one-way Analysis of Variance (ANOVA) [11] test to determine the validity of the null hypothesis that the J_g values obtained by each algorithm (TSOM, SSOM, k-Means) come from the same distribution. This test returns a p-value of 0.0024, indicating that we should reject the null hypothesis and accept the alternative hypothesis that not all J_g values come from the same distribution. Based on this result, we perform Tukey's multiple comparison test [42]. The results of Tukey's test (Table 4.15) suggest that the J_g values obtained by the Tailored SOM are significantly different (in the statistical sense) and lower than those obtained by the Standard SOM and the k-Means algorithm.

Figure 4.20 The box plots of the J_g values obtained by each algorithm: the Tailored SOM (TSOM), the Standard SOM (SSOM), and k-Means for all realizations of Case 2 data.

Table 4.15 The results of performing Tukey's multiple comparison test (as a follow-up to the one-way ANOVA test) to test for statistically significant differences among the distributions of J_g values obtained by the Tailored SOM, the Standard SOM, and the k algorithm for the Case 2 data.

	p-value
TSOM vs. SSOM	0.0026
TSOM vs. k-Means	0.0240
SSOM vs. k-Means	0.7098

We continue with our evaluation and comparison by sorting and plotting the values of J_g obtained by each algorithm, shown in Figure 4.21 (a,b). These plots compare the J_g results for the Tailored SOM versus the Standard SOM (a) and the Tailored SOM versus the k-Means algorithm (b). As previously done, the marker colors are used to indicate the geographic feasibility of the results obtained by each algorithm for each realization. These plots demonstrate a result similar to the one discussed for the Case 1 data; the Tailored SOM outperforms both the Standard SOM and the k-Means algorithm in terms of the geographic qualities quanti ed by the terms in J_g.

(a) (b)

Figure 4.21 A depiction of the values of J_g obtained for each realization of the Case 2 data by (a): the Tailored SOM versus the Standard SOM and (b): the Tailored SOM versus the k-Means algorithm. For each comparison, we sort the function values in ascending order and use the marker colors to indicate the geographic feasibility of each realization. A blue marker indicates a geographically feasible result and a red marker indicates a geographically unfeasible result. In each plot, the solid plotted line corresponds to the results obtained from the Tailored SOM and the dashed line corresponds to the results obtained by the Standard SOM (a) or the k-Means algorithm (b).

Lastly, we compute and consider the mean Stable Score and mean Inlier Score alongside geographic feasibility. We compare the performance of each algorithm by plotting the difference in mean scores for the Tailored SOM versus the Standard SOM (Figure 4.22), and versus the k-Means algorithm (Figure 4.15). From the results depicted in Figure 4.22, we can conclude that, on average, the Tailored SOM achieves a higher mean Stable Score (shown in (a)) than the Standard SOM and an approximately equal Inlier Score (shown in (b)). These results are very similar to those discussed in Case 1, suggesting that the relative performance of the Tailored versus Standard SOM persists in the

presence of increased variance σ_1^2 and lower association probability P. From the results depicted in Figure 4.23, we can conclude that Tailored SOM and the k-Means algorithm are, more or less, on par with one another with regards to the Stable Score (shown in (a)). However, the Tailored SOM does have a small advantage over the k-Means algorithm with regards to the Inlier Score (shown in (b)). We note that the side by side comparison of the mean Stable and Inlier Score for the Tailored SOM and k-Means does not suggest a significant difference in algorithm performance. However, these results (Figure 4.23) are more meaningful when considered in conjunction with the stark contrast between the geographic feasibility of the results obtained by each algorithm and their respective J_g values (Figure 4.21 (b)).

(a) TSOM versus SSOM mean Stable Score differences

(b) TSOM versus SSOM mean Inlier Score differences

Figure 4.22 For each realization of Case 2 data, we compute the difference in mean Stable Score (Sub gure (a)) and mean Inlier Score (Sub gure (b)) achieved by the optimized Tailored SOM (TSOM) versus the Standard SOM (SSOM). A positive score difference indicates a realization for which the TSOM achieved a higher score than the SSOM. Additionally, we plot the mean score difference across all realizations (solid red line) and 1 standard deviation of the mean score differences (dashed red lines). The solid blue line indicates no difference in the mean scores.

(a) TSOM versus k -Means mean Stable Score differences

(b) TSOM versus k -Means mean Inlier Score differences

Figure 4.23 For each realization of Case 2 data, we compute the difference in mean Stable Score (Sub gure (a)) and mean Inlier Score (Sub gure (b)) achieved by the optimized Tailored SOM (TSOM) versus the k -Means algorithm. A positive score difference indicates a realization for which the TSOM achieved a higher score than the k -Means algorithm. Additionally, we plot the mean score difference across all realizations (solid red line) and 1 standard deviation of the mean score differences (dashed red lines). The solid blue line indicates no difference in the mean scores.

4.5 Synthetic Data: Case 3

For a third case, we manufacture a synthetic data set with an unequal number of attribute and geographic groups (p_a 6= p_g). In other words, we consider the scenario where observations within a single attribute group are associated with observations within one of two possible geographic groups. To build this, we let $p_a = 4$ and $p_g = 5$. To ensure that our analysis is able to adequately capture the effects of this structural change in the data on each algorithm's performance, we keep the attribute and geographic radial variances ($^2_1$ and $^2_2$) constant at 0.15^2, and implement an association probability (P) close to 1. The full set of parameters used in data generation are detailed in Table 4.16 and a sample Case 3 data set is depicted in Figure 4.24. The property of unequal attribute and geographic groups in the data set is clearly illustrated by the two distinct geographic groups, one on each side of the vertical constraint, both populated primarily by yellow markers (Figure 4.24 (b)). These two geographic groups both correspond to the yellow attribute group (Figure 4.24 (a)). In this section, we will investigate the performance of the Tailored SOM, the Standard SOM, and the k-Means algorithm with respect to this new property of the data.

(a) (b)

Figure 4.24 A sample Case 3 data set generated from the parameter values de ned in Table 4.2 projected into the two-dimensional attribute domain (shown in (a)) and the two-dimensional geographic domain (shown in (b)). The marker colors in each sub gure are used to indicate the inter-domain correspondence between the geographic observations and the attribute observations. For this case, there are two distinct geographic groups (shown in (b)) that correspond to the same attribute group (yellow group in (a)).

Table 4.16 Parameter values used to manufacture the third case of synthetic data.

Parameter	Value	Definition
p_a	4	number of groups in A
p_g	5	number of groups in G
D	200	total number of observations in the data set
σ_1^2	0.15^2	radial variance for observations in A
σ_2^2	0.15^2	radial variance for observations in G
P	0.95	association probability
M_a	$\begin{bmatrix} 2 & & & & 3 \\ 0.5 & 0.5 & 0.4 & 0.4 & 7 \\ 4 & & & & 5 \\ 0.5 & 0.7 & 0.3 & 0.2 & \end{bmatrix}^T$	group centroids in A
M_g	$\begin{bmatrix} 2 & & & & & 3 \\ 0.5 & 0.5 & 0.4 & 0.1 & 0.5 & 7 \\ 4 & & & & & 5 \\ 0.4 & 0.2 & 0.3 & 0.1 & 0.5 & \end{bmatrix}^T$	group centroids in G

4.5.1 Tailored versus Standard Clustering Results: Case 3

For the Case 3 data, we use the same Tailored SOM design as was used for Case 2, with the parameters and learning rates shown in Table 4.11. This consists of a 3 3 map with $N = 9$ nodes and a neighborhood relation de ned by the structure of the rectangular lattice. For weight vector initialization, we will continue to use the pseudo-random initialization in both the geographic and attribute domains, as described in Section 4.2.1. We continue to optimize the biasing parameter set $q = [\ ,\ ]$ by minimizing the geographic objective function $J_g\ ,$ (equation (3.1)) over the admissible parameter space de ned in equation (4.1), with a circular representative perimeter $\bar{p}$ with radius $R = 0.3$.

For the sample Case 3 data set depicted in Figure 4.24, we obtain the optimal parameter set $q = [0.75, 1.2]$ (equation (3.2)) via a global grid search of the admissible parameter space. With this q, the Tailored SOM produces the geographic clustering depicted in Figure 4.25 (c). We compare these results to the geographic clustering, seen in Figure 4.25 (d), obtained from training a Standard SOM with the same map topology, neighborhood relation, pseudo-random weight vector initialization and learning parameters de ned in Table 4.11. With this comparison, we can observe how implementing an unequal number of attribute and geographic groups (p_a ~~6~~ p_g) has increased the overall level of dif culty of the dual-domain clustering problem. While the Tailored SOM is able to identify dense, geographically feasible clusters throughout G (Figure 4.25 (c)), the results of the Standard SOM (Figure 4.25 (d)) exhibit the disadvantageous effects of the additional geographic group. In particular, these results demonstrate how the imbalance of attribute and geographic groups can lead to a clustering that is both an oversimpli cation of the data and geographically unfeasible. The large dark green cluster spanning the geographic constraint (seen in Figure 4.25 (d)) is an example of the Standard SOM oversimplifying its representation of the data set. By failing to assign the observations in these two geographic regions to distinct clusters, the Standard SOM has created a representation that overlooks the manufactured property p_a ~~6~~ p_g, in favor of clusters that emphasize similarity in A.

Next, to compare against the k-Means algorithm, we implement the Elbow Method and the Average Silhouette Method (Section 2.3.1) to determine an optimal value for k, and depict the results

Figure 4.25 The geographic clusterings obtained for the example Case 3 data set (shown in (a), (b)) from training the Tailored SOM algorithm with $q = [0.75, 1.20]$ (shown in (c)) and the Standard SOM algorithm (shown in (d)). Post clustering, we apply the convex hull boundaries to each geographic cluster containing at least 2 data observations. For this data set, the Tailored SOM (shown in (c)) identified 8 geographic clusters, denoted by marker color, and the Standard SOM (shown in (d)) identified 9 geographic clusters, also denoted by marker color.

in Figure 4.26. The Elbow Method (Figure 4.26 (a)) returns inconclusive results, as there is no clear value of k for which the WSS begins to decrease linearly. Thus, we rely on the Average Silhouette Coefficient (Figure 4.26 (b)), which, as a function of k, is maximized at $k = 4$. We suspect that the imbalance among the number of geographic and attribute groups in this case of synthetic data may be leading to the ambiguous results obtained from the Elbow Method, as it has created a dual-domain clustering problem that is less straightforward than those presented in Case 1 and 2. The

geographic clustering produced by the k-Means algorithm initialized via the $k++$ approach with $k = 4$ is depicted in Figure 4.27. Clearly, these results are problematic. Given the parameters used to generate this data set (Table 4.16), we know there to be five primary geographic groups and a number of smaller geographic groups stemming from noise, yet $k = 4$ clusters was determined to be optimal by standard approaches. As an experiment and an opportunity to further investigate the effects of the property $p_a \ll p_g$ on algorithm performance, we implement the k-Means algorithm for $k = 5$ and $k = 6$, and depict the geographic clustering results in Figure 4.28. These results demonstrate that even with the availability of additional clusters to populate with observations of data, the k-Means algorithm does not produce geographically feasible results that appropriately handle the imbalance among the number of geographic and attribute groups.

(a) (b)

Figure 4.26 Results of the Elbow Method (shown in (a)) and the Average Silhouette Method (shown in (b)) for the Case 3 data set depicted in Figure 4.24. While the results of the Elbow Method are somewhat inconclusive, the Average Silhouette Method suggests the the optimal value for this data set is $k = 4$.

(a) (b)

(c) (d)

Figure 4.27 The geographic clusterings obtained for the example Case 3 data set (shown in (a,b)) from training the Tailored SOM algorithm with $q = [0.75, 1.20]$ (shown in (c)) and the k-Means algorithm with $k = 4$ (shown in (d)). Post clustering, we apply the convex hull boundaries to each geographic cluster containing at least 2 data observations.

(a) (b)

(c) (d)

Figure 4.28 An illustration of the geographic clustering of the example Case 3 data set (shown in (a,b)) obtained with the k-Means algorithm for $k = 5$ (shown in (c)) and $k = 6$ (shown in (d)).

4.5.2 Algorithm Assessment: Case 3

As outlined in previous cases, we compute the value of the geographic objective function J_g (equation (3.1)), the Stable Score and the Inlier Score (Table 4.4) and evaluate the geographic feasibility of the geographic clusterings produced by the Tailored SOM (Figure 4.25 (c)), the Standard SOM (Figure 4.25 (d)) and the k-Means algorithm for $k = 4$ (Figure 4.27 (d)). These results are shown in Tables 4.17 and 4.18. Based on these results, we conclude that in addition to being the only algorithm to achieve a geographically useful clustering of the sample Case 3 data set, the Tailored SOM also outperformed the other two algorithms in terms of J_g, and the mean and median Inlier Score. Furthermore, the Tailored SOM outperformed the Standard SOM in both the mean and median Stable Score.

Table 4.17 Values of the geographic objective function J_g (equation (3.1)) for the clusterings determined by the Tailored SOM, the Standard SOM, and the k-Means algorithm ($k = 4$) on the example Case 3 data set depicted in Figure 4.24.

	Tailored SOM	Standard SOM	k-Means
Value of J_g	0.8626	1.9412	2.6711

Table 4.18 Stable Scores (SS) and Inlier Scores (IS) for the clustering of the example Case 3 data set depicted in Figure 4.24 obtained by the Tailored SOM (Figure 4.25 (c)), the Standard SOM (Figure 4.25 (d)), and k-Means for $k = 4$ (Figure 4.27 (d)). Each of the "Cluster" columns contains the scores for the best-matching cluster to one of the 5 stable groups determined for this example data. The "Mean" column contains the mean Stable Score and mean Inlier Score across all five stable groups for each algorithm. The "Median" column contains the median Stable Score and median Inlier Score across all five stable groups for each algorithm. The red text color is used to denote the algorithms that produced a geographically unfeasible clustering for this example Case 3 data set.

	Cluster 1		Cluster 2		Cluster 3		Cluster 4		Cluster 5		Mean		Median	
	SS	IS	SS	IS	SS	IS	SS	IS	SS	IS	SS	IS	SS	IS
TSOM (%)	100	91.7	100	93.6	100	91.2	100	95.2	100	82.0	100	90.8	100	91.7
SSOM (%)	100	86.3	100	89.8	95.5	53.9	70.0	35.9	83.0	91.9	89.7	71.5	95.5	86.3
k-Means (%)	100	84.6	100	89.8	100	44.0	100	40.0	100	83.7	100	68.4	100	83.7

To investigate whether this pattern of relative algorithm performance persists, we repeat the experiment with an additional 20 realizations of Case 3 synthetic data according to the parameter

values described in Table 4.16. For each realization, we train a Tailored SOM and a Standard SOM according to the parameters in Table 4.11. For the Tailored SOM, determine the optimal $q = [\ ,\ ]$ with $J_g\ ,$ over the admissible parameter space de ned in equation (4.1). Over all realizations, the mean optimal parameter set is $\bar{q} = [\bar{\ },\bar{\ }] = [0.816, 1.158]$ and the median parameter set is $\tilde{q} = [\tilde{\ },\tilde{\ }] = [0.750, 1.20]$. Furthermore, in 85% (17 out of 20) of the optimal parameter sets determined for the Case 3 data realizations, we have $<$. For the k-Means algorithm, we rely on the optimal value of k as identi ed by the Average Silhouette Method. With the results of this experiment, we begin with a comparison of geographic feasibility. Across the 20 realizations of the Case 3 data, a geographically feasible clustering was achieved by the Tailored SOM in 80% of realizations, the Standard SOM in 40% of realizations, and the k-Means algorithm in only 15% of realizations (Table 4.19). Thus, even more so than in the previous two cases of synthetic data, the Tailored SOM far outperformed the Standard SOM and the k-Means algorithm in determining a geographically useful clustering, despite the additional challenge associated with the property $p_a \not\leftarrow p_g$.

Table 4.19 The percentage of Case 3 realizations for which a geographically feasible clustering was achieved by the Tailored SOM, the Standard SOM, and the k-Means algorithm.

	Tailored SOM	Standard SOM	k-Means
Geographically Feasible (%)	80%	40%	15%

To complement the geographic feasibility comparison, we compute the value of J_g for each algorithm and realization and depict the box plots in Figure 4.29. As previously described, we implement the Kruskal-Wallis test [21], which returns a p-value of $2.19 \quad 10^{-8}$, indicating that the J_g values for each algorithm do not come from the same distribution. Thus, we perform Dunn's multiple comparison test and record the pairwise p-values in Table 4.20 [10]. Based on these results, we can conclude that the J_g values obtained by the Tailored SOM are signi cantly different and lower than those obtained by the Standard SOM and the k-Means algorithm. Additionally, the distributions of the J_g values obtained by the Standard SOM and the k-Means algorithm are not signi cantly different.

Figure 4.29 The box plots of the J_g values obtained by each algorithm: the Tailored SOM (TSOM), the Standard SOM (SSOM), and k-Means for all realizations of Case 3 data.

Table 4.20 The results of performing Dunn's multiple comparison test (as a follow-up to the Kruskal-Wallis test) to test for statistically significant differences among the distributions of J_g values obtained by the Tailored SOM, the Standard SOM, and the k-Means algorithm for Case 3 data.

	p-value
TSOM vs. SSOM	$9.35 \quad 10^{5}$
TSOM vs. k-Means	$2.69 \quad 10^{8}$
SSOM vs. k-Means	0.3024

Next, we sort and plot the values of J_g obtained by the Tailored SOM, the Standard SOM, and the k-Means algorithm. The comparison between the Tailored SOM and the Standard SOM is depicted in Figure 4.30 (a) and the comparison between the Tailored SOM and the k-Means algorithm is depicted in Figured 4.30 (b). In both comparisons, J_g values for each algorithm are sorted in ascending order and the marker colors are used to indicate the geographic feasibility of the results obtained in each case. These results indicate an overall larger gap between the J_g value obtained by the Tailored SOM and the standard approaches (Standard SOM and k-Means) than was observed in Case 1 and 2. This observation suggests that the imbalance in the number of attribute versus geographic groups (p_a 6= p_g) is resulting in a larger discrepancy between the performance of the tailored algorithm and the standard algorithms. Thus, these findings further support the claim

that standard approaches, such as the Standard SOM and the k-Means algorithms, are not optimal in a dual-domain clustering setting.

(a) (b)

Figure 4.30 A depiction of the values of J_g obtained for each realization of the Case 3 data by (a): the Tailored SOM versus the Standard SOM and (b): the Tailored SOM versus the k-Means algorithm. For each comparison, we sort the function values in ascending order and use the marker colors to indicate the geographic feasibility of each realization. A blue marker indicates a geographically feasible result and a red marker indicates a geographically unfeasible result. In each plot, the solid plotted line corresponds to the results obtained from the Tailored SOM and the dashed line corresponds to the results obtained by the Standard SOM (a) or the k-Means algorithm (b).

We continue the algorithm evaluation with a comparison of the mean Stable Score and mean Inlier Score (Table 4.4). As previously done, we plot the difference in mean scores for the Tailored SOM versus the Standard SOM (Figure 4.31), and versus the k-Means algorithm (Figure 4.32). Keeping in mind the signi cant difference in geographic feasibility among the three algorithms, the results depicted in Figures 4.31 and 4.32 further suggest that the Tailored SOM is best suited to handle this case of dual-domain data. In particular, we note the considerable difference in mean Inlier Score for the Tailored SOM versus the k-Means algorithm (Figure 4.32 (b)). This nding is consistent with the previous discussion regarding the dif culty of determining an optimal value of k for a data set with an unbalanced number of true geographic and attribute groups. As a result, the Average Silhouette Method frequently determined an optimal k that was too small to effectively identify the nuanced groupings within the data set. Thus, for many of the 20 realizations, k-Means produced a clustering containing many large, far-reaching geographic clusters with low Inlier Scores. The relatively high

mean Stable Scores achieved by k -Means can also be explained by the frequent choice of a low k value, since large spanning clusters will inevitably capture a large percentage of a stable group's population.

(a) TSOM versus SSOM mean Stable Score differences

(b) TSOM versus SSOM mean Inlier Score differences

Figure 4.31 For each realization of Case 3 data, we compute the difference in mean Stable Score (Sub gure (a)) and mean Inlier Score (Sub gure (b)) achieved by the optimized Tailored SOM (TSOM) versus the Standard SOM (SSOM). A positive score difference indicates a realization for which the TSOM achieved a higher score than the SSOM. Additionally, we plot the mean score difference across all realizations (solid red line) and 1 standard deviation of the mean score differences (dashed red lines). The solid blue line indicates no difference in the mean scores.

(a) TSOM versus k-Means mean Stable Score differences

(b) TSOM versus k-Means mean Inlier Score differences

Figure 4.32 For each realization of Case 3 data, we compute the difference in mean Stable Score (Sub gure (a)) and mean Inlier Score (Sub gure (b)) achieved by the optimized Tailored SOM (TSOM) versus the k-Means algorithm. A positive score difference indicates a realization for which the TSOM achieved a higher score than the k-Means algorithm. Additionally, we plot the mean score difference across all realizations (solid red line) and 1 standard deviation of the mean score differences (dashed red lines). The solid blue line indicates no difference in the mean scores.

4.5.3 Optimization via Simulated Annealing

We use the Case 3 data to demonstrate a proof of concept for using the global optimization technique Simulated Annealing (de ned in Algorithm 4) [33], in lieu of the global grid search, to determine the optimal $q = [\ ,\ ]$ (equation (3.2)). For this bounded optimization regime, we maintain the

same admissible parameter space (4.1) that was explored by the global grid search. Additionally, as de ned in Algorithm 4, we prescribe an initial temperature $T = 100$, an initial solution $q_0 = [1, 1]$, and implement the acceptance function $A(c(x), T)$ de ned by equation (3.8). We terminate the global optimization by imposing a termination tolerance of $1 \quad 10^{-6}$ on the objective function J_g (referred to as the "cost function" in Algorithm 4). This termination criterion will cause the algorithm to terminate when the change in the best value of J_g is smaller than the tolerance.

First, we refer back to the example Case 3 data set that we explored in Section 4.5.1. Recall that for this data set we determined the optimal set, $q_1 = [0.75, 1.20]$, via the global grid search. As a comparison, we also use the con guration of the Simulated Annealing algorithm described above to determine the optimal parameter set $q_2 = [0.7519, 1.265]$ for this example data set. With q_2, we train a Tailored Self-Organizing map using the parameters in Table 4.11 and the same pseudo-random initial weight vectors as we did for the global grid search in Section 4.5.1. We illustrate this comparison, along with projections of the example data set, in Figure 4.33. Based on this illustration, we can conclude that these two geographic clusterings (shown in Figure 4.33 (c,d)) are nearly equivalent, with the cluster assignments of three data observations west of the geographic constraint (denoted with black arrows in Figure 4.33 (c)) being the only exceptions. Given the very similar elements of q_1 and q_2, this result is not surprising. Futhermore, we can use J_g to evaluate the similarity between these two geographic clusterings (Table 4.21). For context, we also include the J_g values obtained by the Standard SOM and the k-Means algorithms (as outlined in Section 4.5.1) in Table 4.21.

Table 4.21 Values of the geographic objective function J_g (equation (3.1)) for the clusterings determined by: the Tailored SOM with optimal q_1 determined by the global grid search (denoted as GSin the table), the Tailored SOM with optimal q_2 determined by the Simulated Annealing algorithm (denoted as SAin the table), the Standard SOM, and the k-Means algorithm on the example Case 3 data set depicted in Figure 4.33 (a,b).

	Tailored SOM (GS)	Tailored SOM (SA)	Standard SOM	k-Means
Value of J_g	0.8626	0.6484	1.9412	2.6711

Figure 4.33 The geographic clusterings obtained for the example Case 3 data set (shown in (a), (b)) from training the Tailored SOM algorithm with the optimal parameter set identified by the global grid search, $q_1 = [0.75, 1.20]$, (shown in (c)) and the Tailored SOM algorithm with the optimal parameter set identified by the Simulated Annealing algorithm, $q_2 = [0.7519, 1.2465]$ (shown in (d)). The cluster assignment differences between these two results are denoted with black arrows in (c).

Next, we extend our comparison to include all 20 realizations of the Case 3 data that were examined in Section 4.5.2. For each realization, we determine the optimal parameter set $q = [\ ,\ ]$ using the Simulated Annealing algorithm and train a Tailored SOM with the parameters in Table 4.11 and the same pseudo-random initial weight vectors used for the global grid search. Over all realizations, the mean optimal parameter set is $\bar{q} = [\bar{\ },\ \bar{\ }] = [0.839, 1.193]$ and the median parameter set is $\tilde{q} = [\tilde{\ },\ \tilde{\ }] = [0.780, 1.219]$. Furthermore, in 90% (18 out of 20) of the optimal

parameter sets determined by Simulated Annealing for the Case 3 data realizations, we have < . To evaluate and compare the results obtained by optimizing with the Simulated Annealing algorithm versus the global grid search we generate four visualizations. The rst, shown in Figure 4.34 (a), depicts the box plots for the J_g values achieved by each optimization approach (global grid search and Simulated Annealing). Additionally, we implement the one-way ANOVA test [11], which returns a p -value of 0.1349, indicating that we should accept the hypothesis that the J_g values for each approach come from the same distribution and are not signi cantly different (in the statistical sense). For the second visualization, depicted in Figure 4.34 (b), we sort and plot the values of J_g obtained for each realization by each optimization approach.

The third visualization, depicted in Figure 4.34 (c), illustrates the difference in mean Stable Scores for the geographic clustering determined by each optimization approach. Similarly, for the fourth visualization in Figure 4.34 (d), we depict the difference in mean Inlier Scores for the geographic clusterings determined by each optimization approach over the 20 realizations. In Figure 4.34 (c,d), a positive score difference indicates a realization for which the Taiored SOM optimized via the global grid search obtained a higher mean score than the Tailored SOM optimized via the Simulated Annealing algorithm. Based on these results, we can conclude that the choice in optimization regime does not signi cantly impact the geographic clusterings achieved by the Tailored SOM. Thus, it is reasonable to let the size of the dual-domain data set, or the range of the admissible parameter space for and , dictate which optimization approach is most appropriate and ef cient.

(a)

(b)

(c)

(d)

Figure 4.34 To compare the clusterings of the Case 3 data realizations achieved by the Tailored SOM via a global grid search (GS) and the Tailored SOM via the Simulated Annealing algorithm (SA), we depict the box plots of the J_g values obtained by each optimization approach (shown in (a)). Additionally, we depict of the values of J_g obtained for each realization by the two optimization approaches (shown in (b)). For this comparison, we sort the function values in ascending order and use the marker colors to indicate the geographic feasibility of each realization. A blue marker indicates a geographically feasible result and a red marker indicates a geographically unfeasible result. The solid plotted line corresponds to the results obtained from the Tailored SOM via the global grid search and the dashed line corresponds to the results obtained by the Tailored SOM via the Simulated Annealing algorithm. Lastly, we depict the difference in mean Stable Scores and mean Inlier Scores achieved by the Tailored SOM optimized with the global grid search versus the Tailored SOM optimized via the Simulated Annealing Algorithm (shown in (c,d)).

4.6 Discussion

The geographic objective function J_g developed in Chapter 3, de ned in equation (3.1), and implemented and studied extensively in this chapter is the product of much experimentation. The success demonstrated by the Tailored SOM, as documented in this chapter, is largely a result of a comprehensive and lengthy re nement process. This process consisted of:

- hypothesizing terms and quantities that could be potentially valuable in an objective function designed to tailor and tune the unsupervised clustering of a dual-domain data set,
- testing the capability and robustness of a proposed objective function with synthetic data sets of varying dimension, structure, and levels of noise, and
- probing the results of such tests to better understand the utility in each term of the proposed objective function.

At the core of this process was a requirement for an objective function that was versatile and unbiased. Enough so that it could lend itself to enhancing the capabilities of the Standard Self-Organizing Map algorithm in a number of settings; whether it be a particularly noisy data set, a data set too large to feasibly apply a supervised learning approach, or a real-world application of critical importance. To that end, we carefully considered several general concepts and quantities that could be of use. A few examples of these quantities are: the overlapping area between geographic clusters, the perimeter of a geographic cluster (both of which are outlined in Section 3.2), the average point-to-centroid Euclidean distance of a geographic cluster, and the density of a geographic cluster. With each new proposed term, we examined whether there was a need for an additional weighting parameter to balance its magnitude relative to other terms in the function. Furthermore, we studied whether two or more distinct terms relied too heavily on the same properties of the clustering, thus leading to an overemphasis of particular features of the data set during optimization. Ultimately, we arrived at the version of J_g de ned in equation (3.1) and implemented throughout this chapter. The advantages of this J_g are its dimensionless character, lack of apparent need for additional weighting parameters, and ef cacy in balancing the size and scope of geographic clusters even in the presence of increased levels of noise and geographic observation misplacement.

In the sections of this chapter devoted to algorithm assessment, we detailed how J_g can also serve as an effective method for evaluating and comparing the performance of various clustering algorithms. Given the design process outlined above, J_g is inherently built to favor a clustering with qualities that are desirable in the context of the dual-domain problem. Such qualities include:

- clusters with minimal shared geographic territory, i.e., clusters that identify distinct, non-overlapping regions of interest, and
- geographic feasibility with respect to constraints that are known to the user but unknown to the clustering algorithm.

Therefore, by relying on J_g as a metric for comparing the results of several algorithms, we have not only a method for evaluating each clustering with respect to the overall goals of the dual-domain problem, but also a tool for highlighting the need for and value of a tailored approach. The results of implementing J_g as an algorithm assessment metric (such as those depicted in Figures 4.12, 4.13) demonstrate the signi cant advantage of using the Tailored SOM, as compared to off-the-shelf methods such as the Standard SOM and k-Means algorithm, to achieve dual-domain clusterings that are more desirable and feasible.

An interesting observation worth discussing further is the apparent utility of the Stable Score and the Inlier Score as evaluation metrics for results obtained by the Standard SOM and the k-Means algorithm, respectively. Based on the mean score difference visualizations depicted in Figures 4.14, 4.22, 4.31, 4.15, 4.23, and 4.32, we can observe that, overall, the Stable Score was more useful for quantifying the advantage of the results obtained by the Tailored SOM when compared to the results obtained by the Standard SOM. Whereas the Inlier Score was more useful for quantifying the advantage of the results obtained by the Tailored SOM versus the results obtained by the k-Means algorithm. This pattern can be largely explained by the relationship between the features used to construct each score and the underlying principles of each algorithm.

Recall that the Stable Score (de ned in Step 3, Table 4.4) is de ned as the percentage of the stable group's population represented by its best-matching cluster from the TSOM / SSOM/ k-Means output. Therefore, high Stable Scores are inevitably obtained by large clusters, which likely wholly

envelop a stable group, but also contain outlying observations (observations that are nearby in the geographic domain but dissimilar in the attribute domain). To address this issue, we de ned the Inlier Score (de ned in Step 4, Table 4.4). The Inlier Score for a given cluster is the percentage of its population that is shared with its best-matching stable group. Hence, given that the Elbow Method and the Average Silhouette Method often identi ed low values of k as optimal, the k-Means algorithm frequently produced results with large, far-reaching clusters (such as those in Figure 4.27 (d)). These types of clusters ultimately score highly in terms of the Stable Score, but low in terms of the Inlier Score. Furthermore, given the fundamental similarities between the Standard SOM and the Tailored SOM, we observe more structural similarities between the results achieved by these two algorithms. However, given its equal consideration of data similarity in the attribute and the geographic domain, the Standard SOM often struggled to appropriately handle the misplaced observations / noise. This frequently resulted in the segmentation of geographic data observations belonging to the same stable group into two or more clusters (as demonstrated by the yellow and dark blue clusters in Figure 4.17 (d)). Clusters of this form achieve high Inlier Scores, as they tend to contain few outliers, but low Stable Scores due to the partitioning of a stable group. Ultimately, it may be advantageous to consider an algorithm-speci c approach for evaluating clusters from a synthetic data perspective, though it may sacri ce on the consistent, universal nature of the algorithm assessment regime outlined in this work.

CHAPTER

5

APPLICATION OF THE TAILORED SELF-ORGANIZING MAP TO CANCER INCIDENCE DATA

5.1 Cancer Incidence Data

In this chapter we will further investigate the capabilities of the Tailored Self-Organizing Map in the context of a real data set pertaining to colorectal cancer incidence in the state of California. This data set is an amalgamation of two smaller data sets. The first data set contains county-level colorectal cancer incidence rates and demographic information collected from 2001-2016. This data set was obtained from the CDC's National Program of Cancer Registries (NPCR) and the National Cancer Institute's (NCI's) Surveillance, Epidemiology, and End Results (SEER) Program [37]. The second data set contains representative geographic points, known as “internal points", in the form of the latitude

and longitude measurements for each county represented in the first data set. The internal points were obtained from a United States Gazetteer File [43]. The Gazetteer Files are publicly available and contain information such as geographic identifier codes, area measurements, and representative latitude and longitude coordinates.

The augmented data set described above matches well with the requirements for a dual-domain clustering problem. As stated in earlier chapters, a dual-domain data set is comprised of features that belong to either a geographic domain or an attribute domain. For the work presented in this chapter, we define a two-dimensional geographic domain as containing a representative latitude and longitude pair for each county. For the attribute domain, we will focus on a selection of four colorectal incidence rates and four types of demographic information provided in the SEER data set. Each of the colorectal incidence rates was collected on a population group categorized by sex and race, e.g., white males. These eight features, along with a description of each feature, is shown in Table 5.1. All data considered was collected over the period 2001-2016 and, when applicable, relied on demographic and population data collected during the 2010 U.S. Census. Thus, the dual-domain data set considered in this chapter is 10-dimensional. With this data set, we will investigate and evaluate the performance of the Tailored Self-Organizing Map when clustered on the geographic features (latitude, longitude) augmented with various subsets of the 8-dimensional attribute domain. In other words, we will hold back certain attribute domain features from the clustering algorithm and subsequently analyze the results in the context of the full data set.

We aim to evaluate the capability of the Tailored SOM algorithm in identifying clusters of geographically proximal counties with similar levels of colorectal cancer among the various aforementioned population groups. Additionally, we seek to find clusters with attribute features that are significantly dissimilar from one another, i.e., clusters that uniquely identify groups of counties with high or low incidence rates, median ages, etc. The purpose of this investigation is to better understand how other factors, such as demography, population density, and environmental factors, are correlated to trends of incidence of colorectal cancer. From a public health viewpoint, identifying groups of counties, which posses distinct levels of cancer incidence and a uniquely characterized population, would be useful for assessing county-level need for resources such as government funding, educational initiatives, or early cancer screenings. From an epidemiology viewpoint, it

is critical that we understand better the impact that geographic and population factors, such as urban versus rural community structure, proximity to possible sources of water / air contamination, population density, etc., may have on the prevalence of certain types of cancer. The insight that can be gleaned from this type of analysis is integral to developing effective preventative measures and disease intervention protocols.

Table 5.1 Colorectal cancer incidence and demographic features selected for clustering and / or analysis obtained from the 2001-2016 SEER database.

Feature	Description
Male/ White Rate (MW)	Colorectal cancer incidence rate per 100,000 white males in California
Male/ Black Rate (MB)	Colorectal cancer incidence rate per 100,000 black males in California
Female/ White Rate (FW)	Colorectal cancer incidence rate per 100,000 white females in California
Female/ Black Rate (FB)	Colorectal cancer incidence rate per 100,000 black females in California
Population Density (Pop Dens)	Population (based on 2010 U.S. Census) per square mile in county
Percent Male (Perc M)	Percentage of males in county population (based on 2010 U.S. Census)
Median Male Age (Age M)	Median male age in county (based on 2010 U.S. Census)
Median Female Age (Age F)	Median female age in county (based on 2010 U.S. Census)

To visualize the geographic domain, we create a scatter plot of the representative longitude, latitude pairs (where longitude is on the horizontal axis and latitude is on the vertical axis) for each county, and label the points with the county name, shown in Figure 5.1. Given the dimensionality of the attribute domain, it is not feasible to visualize all of the features simultaneously. Hence, to gain some awareness of the distributions of the data belonging to each feature, we depict the corresponding box plots in Figure 5.2. Based on the box plots, we note that the median, range, and number of points considered to be outliers varies from one feature to the next. Thus, before we can apply a clustering algorithm, we must normalize the data so that some features are not dominant when measuring similarity / distance during the training phase . We compute the feature-wise (column-wise) z-score and use it normalize the data set such that each feature has a mean value of 0 and a standard deviation of 1 but retains the original shape of its distribution [38].

Figure 5.1 A scatter plot of the representative longitude, latitude pair for each county, labeled with the county name. To maintain consistency with standard map views, we have longitude on the horizontal axis and latitude on the vertical axis.

(a) Cancer Incidence Rates

(b) Demographic Information

(c) Population Density ($\log_{10}$ scale)

Figure 5.2 A depiction of the box plots of the data from various features in the attribute domain. In (a), we depict the box plots for the colorectal cancer incidence rates by group: Male / White Rate (MW Rate), Male / Black Rate (MB Rate), Female/ White Rate (FW Rate), Female/ Black Rate (FB Rate). In (b), we depict the box plots for the features related to demographic information: Percent Male (Perc M), Median Male Age (Age M), Median Female Age (Age F). In (c), we depict the box plot for the Population Density.

5.2 Case 1: An R^4 Subset of Cancer Incidence Data

We begin our investigation of the colorectal cancer incidence data by applying the same Tailored Self-Organizing Map framework that we developed in Chapter 3 and thoroughly assessed in Chapter 4 on an R^4 dual-domain subset of the colorectal cancer incidence data. This first case (subset) will be similar to the synthetic data sets previously analyzed in Chapter 4 in that it will consist of a two-dimensional geographic domain (G) and a two-dimensional attribute domain (A). By commencing our analysis with an implementation of the same Tailored SOM and geographic objective function J_g (equation (3.1)) that we have previously examined on a data set with a familiar structure, we provide ourselves the opportunity to evaluate how the methodology and existing parameters and design choices perform in the context of real, non-synthetic data. The two-dimensional geographic domain contains the representative latitude and longitude measurements for each county. The two-dimensional attribute domain will consist of the two colorectal cancer incidence rates pertaining to the male population: *Male/White Rate* and *Male/Black Rate*. While only these four features will be considered during algorithm training, we will use the results and extend cluster assignments to the full data set (consisting of all 10 features) in order to identify and analyze any patterns, groupings, or trends that exist among all features.

5.2.1 Tailored versus Standard Clustering Results: Original J_g

First, for the Tailored Self-Organizing Map, we consider a 3×3 map, consisting of $N = 9$ nodes, arranged as a 2-dimensional rectangular lattice, where the lattice structure defines the neighborhood relation among the nodes. To mirror the evenly-spaced initialization of weight vectors implemented for the Case 1 data in Chapter 3, for this experiment we initialize the weight vectors evenly throughout data space based on the feature *Median Age Male*. That is to say, we identify the California counties with the minimum male median age, the maximum male median age, and determine the seven values evenly spaced between these end points (with the endpoints included, this yields a total of 9 candidate values). Then, for each candidate median age value, we compute its closest data observation within the *Median Age Male* feature and initialize one of the weight vectors to be positioned at its location in data space. We conclude that this method of weight vector initialization is reasonably unbiased given that we are not considering the *Median Age Male* feature during the

clustering regime. The remaining parameters and design choices for this implementation of the Tailored SOM are summarized in in Table 5.2.

Table 5.2 Parameters and design choices prescribed for training the Tailored SOM on the R^4 subset of the colorectal cancer data consisting of latitude, longitude, MW Rate, and MB Rate features.

Parameter	Value	De nition
N	9	total number of nodes in SOM (given by 3 3 rectangular lattice structure)
initial w_j^a ($j = 1,..,N$)	evenly spaced in Male Median Age	weight vector initialization in A
initial w_j^g ($j = 1,..,N$)	evenly spaced in Male Median Age	weight vector initialization in G
w_n	1 edge away	neighborhood relation among nodes
$z(k)$	$\frac{1}{2}^{\frac{k}{k_{max}}}$	learning rate for winning weight vectors
(k)	$\frac{1}{6}^{\frac{k}{k_{max}}}$	learning rate for neighboring weight vectors
k_{max}	7	number of training batches

For the optimization of J_g , (as de ned in equation (3.1)), we maintain the admissible parameter space used throughout Chapter 4 (equation (4.1)). To account for the difference in the range of the normalized latitude / longitude values versus the range of the synthetic geographic data considered in Chapter 4, we prescribe a circular representative perimeter, $\bar{p}$, with a radius of $r = 0.5425$ in J_g. This radius was computed by determining a total normalized area of California (via the convex hull of all geographic data observations), dividing this area into 9 circular representative areas (equal to the total number of nodes in the TSOM), and solving for the radius. To determine an optimal $q = [\ ,\ ]$ (equation (3.2)), we use the global grid search to traverse the admissible parameter space via the step size $q = [0.05, 0.05]$. With this approach, we determine that the optimal parameter set for this 4-dimensional case of the colorectal cancer data is $q = [0.75, 1.25]$. When trained with this q , the Tailored SOM identi es the geographic clustering depicted in Figure 5.3 (a).

For comparisons, we train a Standard Self-Organizing Map and the k-Means algorithm. For the Standard Self-Organizing Map, we use the same structure, parameters, and weight vector initialization as we did for the Tailored SOM described above. For the k-Means algorithm, we train two formulations, both initialized with the $k++$ approach. The first k-Means formulation uses $k = 5$, which was identified as the optimal value for k by the Average Silhouette Method. The second k-Means formulation uses $k = 9$, which mirrors the size (number of nodes) of the Tailored SOM. The geographic clusterings produced by these algorithms are depicted in Figure 5.3 (b,c,d). As previously done, we use the convex hull of each cluster to define geographic boundaries throughout. Additionally, to streamline the way in which we evaluate and analyze the results obtained by each clustering algorithm, we identify and focus on the top four largest clusters (in terms of the number of counties assigned to the cluster), when appropriate. We denote the top four most populated clusters for each result in the legend of each plot (shown in Figure 5.3).

Immediately evident from the results depicted in Figure 5.3 is the difference in geographic delineation and separation achieved by the tailored approach versus the three standard approaches. In particular, we observe a high level of geographic cluster overlap in the k-Means result for $k = 9$ (Figure 5.3 (d)), with the purple, medium blue, and red clusters all occupying much of the same territory. Additionally, we note that the k-Means result for $k = 5$, which was the value identified as optimal by the Average Silhouette Method, appears to be somewhat oversimplified. We base this assessment on the fact that all but 5 of the counties (shown as the red and purple clusters in Figure 5.3 (c)) were assigned to one of only three clusters. Upon examination of the Standard SOM results, we note some general structural similarities between its clusters and the clusters identified by the Tailored SOM. For example, there are consistencies in how some groups of counties in the northernmost and southernmost regions were clustered.

To give a more thorough evaluation and comparison of the results depicted in Figure 5.3 we rely on two metrics. First, we evaluate the geographic objective function J_g, as described earlier in this section, for each clustering. This will serve as a way to assess algorithm performance from a geographic-similarity viewpoint. Recall that J_g was designed such that the clustering with the lowest score is considered most desirable. Secondly, we use statistical testing to compare the number of statistically significant differences (at the significance level $= 0.05$) among the data components in

(a) Tailored SOM Result

(b) Standard SOM Result

(c) k-Means Result: k = 5

(d) k-Means Result: k = 9

Figure 5.3 The geographic clusterings obtained for 4-dimensional case of the colorectal cancer data with features: latitude, longitude, MW Rate, and MB Rate. These results are from training the Tailored SOM algorithm with $q = [0.75, 1.25]$ obtained via the geographic objective function J_g (shown in (a)), the Standard SOM algorithm (shown in (b)), the k-Means algorithm with $k = 5$ (shown in (c)), and the k-Means algorithm with $k = 9$. Post clustering, we apply the convex hull boundaries to each geographic cluster containing at least 2 data observations.

the attribute domain in the top 4 largest clusters in the clusterings produced by the Tailored SOM, the Standard SOM, and the k-Means algorithm. Depending on the distribution of each cluster's data and its adherence to normality requirements, we utilize either: (1) the one-way Analysis of Variance (ANOVA) test [11] followed by Tukey's multiple comparison test [42], or (2) the Kruskal-Wallis test [21] followed by Dunn's multiple comparison test [10]. For this, we will use the cluster assignments identified by each algorithm and extend them to include the data from all 8 attribute features of the

cancer incidence data set (listed in Table 5.1).

Given that we are considering the top 4 largest clusters, there are six pairwise comparisons to evaluate for statistically signi cant differences: Cluster 1 versus Cluster 2, Cluster 1 versus Cluster 3, Cluster 1 versus Cluster 4, etc. As an example, we depict the full pairwise cluster comparison statistical testing results for the Tailored SOM in Table 5.3. In terms of a single aggregate metric, we report these results as the percentage of the table entries that indicate a statistically signi cant difference among the attribute features for the six pairwise comparisons, denoted as $_A$ (i.e., the percentage of entries in Tables 5.3, 5.4, 5.5 that are highlighted, indicating a p-value < 0.05). This will serve as a way to assess algorithm performance from an attribute-similarity viewpoint, where a higher $_A$ is considered more desirable. We collate the J_g and $_A$ results for each algorithm in Table 5.6. Given that the k-Means result for $k = 5$ (Figure 5.3 (c)) assigned approximately 81% of the counties to only 3 clusters, we exclude it from this analysis.

Table 5.3 A depiction of the results of performing pairwise statistical testing (either the ANOVA test followed by the Tukey multiple comparison test, or the Kruskal-Wallis test followed by Dunn's test) on the top 4 largest clusters identi ed by the Tailored SOM (Figure 5.3 (a)). The top 4 largest clusters are denoted as "C_1^T","C_2^T","C_3^T","C_4^T" and correspond to the clusters referenced in the legend of Figure 5.3 (a). To aid in the visualization of these results, we have highlighted in yellow each pairwise comparison that is statistically signi cant at the $= 0.05$ signi cance level.

	Pop Dens	Perc M	Age M	Age F	MW	MB	FW	FB
C_1^T v. C_2^T	0.0003	0.0252	0.0224	0.0019	0.0065	0.0009	0.0026	0.0006
C_1^T v. C_3^T	0.9337	0.8691	0.7837	0.6673	0.9748	0.9999	0.3368	0.9889
C_1^T v. C_4^T	0.0124	0.5247	0.4700	0.1374	0.9921	0.9999	0.9964	0.6305
C_2^T v. C_3^T	0.0433	0.5630	0.0018	0.0005	0.0027	0.0124	0.6561	0.0300
C_2^T v. C_4^T	0.9791	0.8706	0.8866	0.4705	0.0045	0.0124	0.0554	0.2050
C_3^T v. C_4^T	0.2964	0.9984	0.0713	0.0286	1.0000	1.0000	0.8154	0.9812

Table 5.4 A depiction of the results of performing pairwise statistical testing (either the ANOVA test followed by the Tukey multiple comparison test, or the Kruskal-Wallis test followed by Dunn's test) on the top 4 largest clusters identi ed by the Standard SOM (Figure 5.3 (b)). The top 4 largest clusters are denoted as "C_1^S","C_2^S","C_3^S","C_4^S" and correspond to the clusters referenced in the legend of Figure 5.3 (b). To aid in the visualization of these results, we have highlighted in yellow each pairwise comparison that is statistically signi cant at the = 0.05 signi cance level.

	Pop Dens	Perc M	Age M	Age F	MW	MB	FW	FB
C_1^S v. C_2^S	0.0002	0.0354	0.7357	0.6119	0.0129	0.0010	0.1313	0.0160
C_1^S v. C_3^S	0.0700	0.0164	0.4369	0.5098	0.9093	0.9935	0.2544	0.9675
C_1^S v. C_4^S	0.9981	0.9563	0.6129	0.8002	0.5060	0.9998	0.4493	1.0000
C_2^S v. C_3^S	0.8315	0.9934	0.0331	0.0286	0.1537	0.0008	0.9997	0.3456
C_2^S v. C_4^S	0.0096	0.5347	0.0650	0.0890	0.0010	0.0022	0.9659	0.0876
C_3^S v. C_4^S	0.3433	0.2838	1.0000	0.9995	0.2729	1.0000	0.9860	0.9948

Table 5.5 A depiction of the results of performing pairwise statistical testing (either the ANOVA test followed by the Tukey multiple comparison test, or the Kruskal-Wallis test followed by Dunn's test) on the top 4 largest clusters identi ed by the k-Means algorithm for $k = 9$ (Figure 5.3 (d)). The top 4 largest clusters are denoted as "C_1^{k9}","C_2^{k9}","C_3^{k9}","C_4^{k9}" and correspond to the clusters referenced in the legend of Figure 5.3 (d). To aid in the visualization of these results, we have highlighted in yellow each pairwise comparison that is statistically signi cant at the = 0.05 signi cance level.

	Pop Dens	Perc M	Age M	Age F	MW	MB	FW	FB
C_1^{k9} v. C_2^{k9}	0.0979	0.2252	0.2862	0.3653	0.8738	0.7541	0.0191	0.8955
C_1^{k9} v. C_3^{k9}	0.4778	0.9708	1.0000	1.0000	0.0014	0.0000	0.0176	0.1181
C_1^{k9} v. C_4^{k9}	1.0000	0.9950	0.4472	0.4955	0.6133	0.2969	0.0789	1.0000
C_2^{k9} v. C_3^{k9}	0.9972	0.9091	0.4928	0.6303	0.0263	0.0018	0.9939	0.7738
C_2^{k9} v. C_4^{k9}	0.2973	0.8551	1.0000	1.0000	0.3468	0.8411	0.9982	0.9793
C_3^{k9} v. C_4^{k9}	0.6834	1.0000	0.6008	0.6951	0.0007	0.0380	0.9789	0.3689

Table 5.6 Values of the geographic objective function J_g (as de ned in Section 5.2.1) and the percentage of pairwise cluster comparisons (of the 4 largest clusters, across the 8 attribute features) that are signi cantly different ($_A$) for the clusterings determined by the Tailored SOM, the Standard SOM, and the k-Means algorithm ($k = 9$) on the R^4 subset of the cancer incidence data.

	Tailored SOM	Standard SOM	k-Means ($k = 9$)
Value of J_g	0.6846	2.0080	4.7696
Value of $_A$	37.5%	25%	16.7%

Based on the results in Table 5.6, we can conclude that the Tailored SOM achieved the most desirable results from both a geographic-similarity and attribute-similarity point of view. This is signi cant as it demonstrates the advantage of applying a tailored approach over a standard, off-the-shelf algorithm on a real world, non-synthetic data set. Our analysis found the highest number of statistically signi cant differences among the attribute features existed between:

- Cluster 1 and Cluster 2 in the Tailored SOM result (cyan and dark green clusters in Figure 5.3 (a)), with all 8 features having signi cant differences (Table 5.3),
- Cluster 1 and Cluster 2 in the Standard SOM result (orange and medium blue clusters in Figure 5.3 (b)), with 5 features having signi cant differences (Table 5.4) and,
- Cluster 1 and Cluster 3 in the k-Means for $k = 9$ result (purple and medium blue clusters in Figure 5.3 (d)), with 3 features having signi cant differences (Table 5.5).

Next, to expand upon the rst bulleted item, we illustrate some of the differences that exist among the largest (Cluster 1) and second largest (Cluster 2) clusters (in terms of the number of counties) in the Tailored SOM result. To do so, we depict the data pertaining to select features (Population Density, MW Rate, MB Rate, and FW Rate) assigned to Cluster 1, assigned to Cluster 2, and all data pertaining to that feature. We present this depiction in Figure 5.4, organized by feature: Population Density (shown in (a)), MW Rate (shown in (b)), MB Rate (shown in (c)), and FW Rate (shown in (d)). From this depiction, we can conclude that Cluster 1 contains more counties with higher population densities, higher incidences of colorectal cancer among the Male / White population, the Male / Black population, and the Female / White population, than Cluster 2. Probing the Tailored SOM results in this manner is advantageous for two reasons. First, it allows us to better understand the groupings and patterns that were identi ed among the attribute features considered during training (MW Rate, MB Rate). Secondly, it affords us the opportunity to infer correlations between attribute features that were not considered during training (Population Density, FW Rate). This second result is especially useful as it suggests it is not always necessary to implement a high-dimensional clustering problem to uncover a comprehensive representation of patterns that exist within a data set. Instead, using strategic feature selection, such as experimenting with holding back certain attributes during algorithm training, has proven to be a useful approach.

(a) Population Density

(b) MW Rate

(c) MB Rate

(d) FW Rate

Figure 5.4 A depiction of some of the results obtained by the Tailored SOM clustering (shown in Figure 5.3 (a)). Each plot shows the data pertaining to select features (Population Density, MW Rate, MB Rate, and FW Rate) that was assigned to rst largest cluster (Cluster 1), the second largest cluster (Cluster 2), and all data pertaining to that feature.

5.2.2 Tailored versus Standard Clustering Results: Modi ed Objective Function

Next, to build upon the results presented in the previous section, we propose a modi ed objective function, J_c, to aid in the tailored clustering of the cancer incidence data. We will refer to this new formulation as a combination objective function, denoted by the subscript c, as it does not exclusively rely on quantities derived only from the geographic projection of the clustering. With the implementation of J_c, we aim to identify clusters with signi cantly different levels of colorec-

tal cancer incidence, population density, and demographic features that are also geographically proximal. Thus, we propose the function de ned in equation (5.1), which consists of one term that relies on geographic properties of the clustering, and one term that relies on a quantity based in the attribute domain features. This objective function was built by modifying the geographic objective function J_g, shown here in equation (5.2), thus making it more appropriately tailored to the speci c characteristics of the cancer incidence data set.

$$J_c(\ ,\ ) = \left(1 - \sum_{\substack{i,j=1 \\ i \neq j}}^{C} \frac{A_{i,j}}{\tilde{A}}\right) + \ _A \tag{5.1}$$

$$J_g(\ ,\ ) = \sum_{\substack{i,j=1 \\ i \neq j}}^{C} \frac{A_{i,j}}{\tilde{A}} + \frac{1}{C}\sum_{i=1}^{C}\left(\frac{p_i}{\bar{p}} - 1\right)^2 \tag{5.2}$$

We design the combination objective function J_c using the quantities listed below.

- $A_{i,j}$: the overlapping geographic area between the i^{th} and j^{th} clusters, for $i, j = 1,..,C$ and $i \neq j$, where C is the total number of identi ed clusters.
- $\tilde{A}$: a total representative area of California, obtained by determining the area contained within the convex hull of all counties.
- $_A$: the percentage of statistically signi cant differences among the 8 attribute features across the six pairwise comparisons of the four most populated clusters, as shown by Table 5.3.

Given these terms, and in particular the quantity $_A$, we will search for the parameter set $q = [\ ,\ ]$ that maximizes J_c, i.e.,

$$q = [\ ,\ ] = \underset{,}{\text{argmax}}\ J_c(\ ,\ ). \tag{5.3}$$

Though it would have been feasible to incorporate $_A$ such that we would seek to minimize J_c, we felt it was more natural to leave $_A$ as is and, instead, seek to maximize the objective function. With the decision to frame the optimization of J_c as a maximization problem came the need to modify the overlapping area term. Thus, we incorporate the term $\left(1 - \sum_{i,j} \frac{A_{i,j}}{\tilde{A}}\right)$ which aims to maximize the

geographic area not shared between clusters as a percentage of the total representative California area.

One important difference between J_c and J_g is the term in J_g involving the cluster perimeters, p_i for $i = 1,..,C$, and a representative perimeter $\bar{p}$. Although we investigated the utility of including a perimeter-based term in the modified objective function, ultimately, we felt that it was not appropriate given the geography of California. Specifically, we determined that there was not an obvious best choice for the shape or size of a representative perimeter, given the uneven distribution of counties throughout the state. Instead, we chose to exclusively rely on the overlapping area term $\left(1 - \sum \frac{A_{i,j}}{\tilde{A}}\right)$ for tuning the geographic properties of the optimal clustering. Lastly, to balance with the geo-centric quality of $\left(1 - \sum \frac{A_{i,j}}{\tilde{A}}\right)$, we incorporate α_A into the new function J_c. As demonstrated by the discussion related to Table 5.3, this term aims to promote a clustering within which there exists numerous statistically significant differences, in terms of the attribute features, across primary clusters. With the incorporation of α_A into the objective function, we ensure that adequate weight is being placed on the features from the attribute domain, so as to not diminish their distinct groupings in favor of geographic proximity. By defining J_c in terms of α_A and $\left(1 - \sum \frac{A_{i,j}}{\tilde{A}}\right)$, we are more likely to identify a clustering that is well-balanced in terms of both its geographic clusters' uniqueness and ability to represent the complex groupings with the attribute domain. Lastly, by designing J_c to consist of only one geographic-based term and one attribute-based term, we limit the need for additional term-specific weighting parameters.

To begin our implementation of J_c, we return to the R^4 subset of the colorectal cancer incidence data consisting of a two-dimensional geographic domain (latitude, longitude) and a two-dimensional attribute domain (MW Rate, MB Rate). We continue to use the Tailored SOM described in the previous section and defined by the parameters in Table 5.2. For the optimization of J_c, we use the global grid search to traverse the admissible parameter space defined in Table 4.1 with the step size $\eta = [0.05, 0.05]$. With this approach, we determine the optimal parameter set for this 4-dimensional case of the colorectal cancer data is $\eta = [0.75, 1.10]$. When trained with this η, the Tailored SOM identifies the geographic clustering depicted in Figure 5.5 (a). Recall that the Tailored SOM obtained via the geographic objective function J_g (Section 5.2.1) outperformed the Standard

SOM and the k-Means algorithm based on the metrics depicted in Table 5.6. Hence, we will limit our comparisons in this section to only those results achieved by the Tailored SOM via J_g and J_c. For visualization purposes, we depict these two results side by side in Figure 5.5, with the previous results achieved via the geographic objective function J_g (discussed in Section 5.2.1) shown in (b).

(a) Tailored SOM Result via J_c (b) Tailored SOM Result via J_g

Figure 5.5 The geographic clusterings obtained for 4-dimensional case of the colorectal cancer data (with features: latitude, longitude, MW Rate, MB Rate) from training the Tailored SOM algorithm with $q = [0.75, 1.1]$ obtained via the combination objective function J_c (shown in (a)) and the Tailored SOM algorithm with $q = [0.75, 1.25]$ obtained via the geographic objective function J_g (shown in (b)). Post clustering, we apply the convex hull boundaries to each geographic cluster containing at least 2 data observations.

To compare these results, we evaluate one geographic metric and one attribute metric, shown in Table 5.7. For the geographic evaluation, we compute $\sum \frac{A_{i,j}}{\tilde{A}}$, for $i, j = 1, .., C$ and $i \neq j$, which is the fraction of the total California area occupied by overlapping cluster regions (reported as a percentage). For the attribute evaluation, we compute $_A$, which is the percentage of pairwise cluster comparisons (of the 4 largest clusters) that are signi cantly different in terms of the 8 attribute features. To illustrate the computation of $_A$, we depict the full pairwise cluster comparison statistical testing results for the tailored clustering obtained via J_g and J_c in Table 5.8 and Table 5.9, respectively. In these tables, the yellow highlighted entries indicate each pairwise cluster comparison that was found to be statistically signi cant at the $= 0.05$ signi cance level.

Based on the results depicted in Figure 5.5 and Tables 5.7, 5.8, and 5.9, we can conclude

that the two objective functions yield rather similar results when implemented with the Tailored SOM. The tailored clustering obtained via the combination objective function, J_c (equation (5.1)), identi ed clusters with a higher number of signi cant differences in the attribute domain data. Whereas, the tailored clustering obtained via the geographic objective function, J_g (equation (3.1)), achieved a clustering with a slightly lower amount of geographic cluster overlap. Thus, for this R^4 subset of the colorectal cancer data, the preferred choice in objective function will likely depend on the speci c research question.

Table 5.7 The fraction (reported as a percentage) of the total California area occupied by overlapping cluster regions $\sum \frac{A_{i,j}}{\tilde{A}}$, for $i, j = 1, .., C$ and $i \neq j$ and the percentage of pairwise cluster comparisons (of the 4 largest clusters, across the 8 attribute features) that are signi cantly different ($_A$) for the clusterings determined by the Tailored SOM via J_c and via J_g on the R^4 subset of the cancer incidence data.

	Tailored SOM via J_c	Tailored SOM via J_g
$\sum \frac{A_{i,j}}{\tilde{A}}$ 100	4.44%	3.27%
$_A$	39.6%	37.5%

Table 5.8 A depiction of the results of performing pairwise statistical testing (either the ANOVA test followed by the Tukey multiple comparison test, or the Kruskal-Wallis test followed by Dunn's test) on the top 4 largest clusters identi ed by the Tailored SOM via the combination objective function J_c (Figure 5.5 (a)). The top 4 largest clusters are denoted as " $C_1^{T,c}$," " $C_2^{T,c}$," " $C_3^{T,c}$," " $C_4^{T,c}$ " and correspond to the clusters referenced in the legend of Figure 5.5 (a). To aid in the visualization of these results, we have highlighted in yellow each pairwise comparison that is statistically signi cant at the = 0.05 signi cance level.

	Pop Dens	Perc M	Age M	Age F	MW	MB	FW	FB
$C_1^{T,c}$ v. $C_2^{T,c}$	0.7719	0.5879	0.5395	0.3889	0.9513	0.9993	0.1769	0.9857
$C_1^{T,c}$ v. $C_3^{T,c}$	0.3634	1.0000	0.0136	0.0028	1.0000	0.9994	0.5039	0.8836
$C_1^{T,c}$ v. $C_4^{T,c}$	0.0224	0.7175	0.0002	0.0000	0.0013	0.0032	0.5339	0.0044
$C_2^{T,c}$ v. $C_3^{T,c}$	0.0185	0.7851	0.5083	0.1107	0.9453	0.9755	0.9987	0.4583
$C_2^{T,c}$ v. $C_4^{T,c}$	0.0003	0.0595	0.0451	0.0015	0.0217	0.0008	0.0021	0.0004
$C_3^{T,c}$ v. $C_4^{T,c}$	0.8787	0.6347	0.8897	0.3669	0.0020	0.0205	0.0165	0.1468

Table 5.9 A depiction of the results of performing pairwise statistical testing (either the ANOVA test followed by the Tukey multiple comparison test, or the Kruskal-Wallis test followed by Dunn's test) on the top 4 largest clusters identified by the Tailored SOM via the geographic objective function J_g (Figure 5.5 (b)). The top 4 largest clusters are denoted as "$C_1^{T,g}$", "$C_2^{T,g}$", "$C_3^{T,g}$", "$C_4^{T,g}$" and correspond to the clusters referenced in the legend of Figure 5.5 (b). To aid in the visualization of these results, we have highlighted in yellow each pairwise comparison that is statistically significant at the $= 0.05$ significance level.

	Pop Dens	Perc M	Age M	Age F	MW	MB	FW	FB
$C_1^{T,g}$ v. $C_2^{T,g}$	0.0003	0.0252	0.0224	0.0019	0.0065	0.0009	0.0026	0.0006
$C_1^{T,g}$ v. $C_3^{T,g}$	0.9337	0.8691	0.7837	0.6673	0.9748	0.9999	0.3368	0.9889
$C_1^{T,g}$ v. $C_4^{T,g}$	0.0124	0.5247	0.4700	0.1374	0.9921	0.9999	0.9964	0.6305
$C_2^{T,g}$ v. $C_3^{T,g}$	0.0433	0.5630	0.0018	0.0005	0.0027	0.0124	0.6561	0.0300
$C_2^{T,g}$ v. $C_4^{T,g}$	0.9791	0.8706	0.8866	0.4705	0.0045	0.0124	0.0554	0.2050
$C_3^{T,g}$ v. $C_4^{T,g}$	0.2964	0.9984	0.0713	0.0286	1.0000	1.0000	0.8154	0.9812

5.3 Case 2: An R^6 Subset of Cancer Incidence Data

We continue our investigation of the colorectal cancer incidence data by considering a larger, R^6 dual-domain subset. This subset of data will maintain the geographic domain consisting of the latitude and longitude measurements for each county (plotted in Figure 5.1). We increase the dimension of the attribute domain to 4-dimensional, consisting of the features Male / White Rate, Male / Black Rate, Female/ White Rate, and Female / Black Rate (defined in Table 5.1). With this subset, we can test and evaluate the capabilities and robustness of the Tailored SOM clustering approach, compared to standard approaches, in the presence of a higher-dimensional attribute space (and thus an unequal number of geographic and attribute features).

5.3.1 Tailored versus Standard Clustering Results: J_c

With this larger data subset, we choose to experiment further with the modified, combination objective function J_c (equation (5.1)) defined in Section 5.2.2. For the Tailored SOM, we use the weight vector initialization based on even spacing in the Median Age Male feature and the map parameters and design choices shown in Table 5.2. For the optimization of J_c, we continue our use of the global grid search to explore the admissible parameter space defined in equation (4.1) with

step size $q = [0.05, 0.05]$. With this approach, we determine the optimal parameter set for this 6-dimensional case of the colorectal cancer data is $q = [0.75, 0.95]$. When trained with this q, the Tailored SOM identifies the geographic clustering depicted in Figure 5.6 (a).

As outlined in Section 5.2.1, we generate three additional clusterings to compare against the Tailored SOM results. For the first comparison, we train a Standard SOM with the same structure, parameters, and weight vector initialization as we prescribed for the Tailored SOM. The geographic clustering results of this algorithm are shown in Figure 5.6 (b). For the k-Means algorithm, we train two formulations, both initialized with the $k++$ approach. The first k-Means formulation uses $k = 6$, which was identified as the optimal value for k by the Average Silhouette Method. The second k-Means formulation uses $k = 9$, which mirrors the size (number of nodes) of the Tailored SOM. The geographic clustering results of these two implementations of the k-Means algorithm are depicted in Figure 5.6 (c,d). For each result, we first compute the fraction of the total California area occupied by overlapping cluster regions $\sum \frac{A_{i,j}}{\tilde{A}}$, for $i, j = 1, .., C$ and $i \neq j$. Additionally, we compute the percentage of pairwise cluster comparisons (of the 4 largest clusters) that are significantly different in terms of the 8 attribute features ($_A$). The results of these evaluation metrics for each clustering are shown in Table 5.10. Furthermore, we depict the full pairwise cluster comparison statistical testing results for the each result in Tables 5.11, 5.12, 5.13, 5.14.

From the results depicted Figure 5.6 we observe significant similarities between the clusterings achieved by the k-Means algorithm for $k = 6$ and $k = 9$. In particular, we note that in both cases a significant majority of the counties are accounted for by only 3 large clusters. Whereas the clusterings generated by the Tailored SOM and Standard SOM reveal smaller geographic clusters, especially in the more densely populated northern half of California. With the identification of smaller geographic clusters, we are able to learn information about the distributions of each incidence rate and demography at a more granular level. This result is reflected in the values of $_A$ (Table 5.10) computed for each clustering. Specifically, the Tailored SOM outperforms the three other algorithms in its ability to identify clusters that are significantly distinct from one another (in the statistical sense) in the features of the attribute domain. Furthermore, the Tailored SOM is able to achieve clusters with well-defined attribute similarity without compromising on geographic uniqueness, as demonstrated by the overlapping region results $\sum \frac{A_{i,j}}{\tilde{A}}$ shown in Table 5.10.

(a) Tailored SOM Result

(b) Standard SOM Result

(c) k-Means Result: k = 6

(d) k-Means Result: k = 9

Figure 5.6 The geographic clusterings obtained for 6-dimensional case of the colorectal cancer data with features: latitude, longitude, MW Rate, MB Rate, FW Rate, and FB Rate. These results are from training the Tailored SOM algorithm with $q = [0.75, 0.95]$ obtained via the combination objective function J_c (shown in (a)), the Standard SOM algorithm (shown in (b)), the k-Means algorithm with $k = 6$ (shown in (c)), and the k-Means algorithm with $k = 9$ (shown in (d)). Post clustering, we apply the convex hull boundaries to each geographic cluster containing at least 2 data observations.

Table 5.10 The fraction (reported as a percentage) of the total California area occupied by overlapping cluster regions $\sum \frac{A_{i,j}}{\tilde{A}}$, for $i, j = 1, .., C$ and $i \neq j$ and the percentage of pairwise cluster comparisons (of the 4 largest clusters, across the 8 attribute features) that are significantly different (α_A) for the clusterings determined by the Tailored SOM, the Standard SOM, and the k-Means algorithm (for $k = 6, 9$) on the R^6 subset of the cancer incidence data.

	Tailored SOM	Standard SOM	k-Means ($k = 6$)	k-Means ($k = 9$)
$\sum \frac{A_{i,j}}{\tilde{A}}$ 100	11.96%	12.06%	13.38%	11.1%
α_A	39.6%	31.3%	21.3%	20.1%

Table 5.11 A depiction of the results of performing pairwise statistical testing (either the ANOVA test followed by the Tukey multiple comparison test, or the Kruskal-Wallis test followed by Dunn's test) on the top 4 largest clusters identified by the Tailored SOM via the combination objective function J_c for the R^6 data subset (Figure 5.6 (a)). The top 4 largest clusters are denoted as "C_1^T","C_2^T","C_3^T","C_4^T" and correspond to the clusters referenced in the legend of Figure 5.6 (a). To aid in the visualization of these results, we have highlighted in yellow each pairwise comparison that is statistically significant at the $\alpha = 0.05$ significance level.

	Pop Dens	Perc M	Age M	Age F	MW	MB	FW	FB
C_1^T v. C_2^T	0.7286	0.5270	0.4304	0.5068	0.4601	0.9989	0.0000	0.6685
C_1^T v. C_3^T	0.0010	0.0491	0.2986	0.0116	0.0015	0.0000	0.0000	0.0000
C_1^T v. C_4^T	1.0000	0.7050	0.9909	0.8624	0.0327	0.0253	0.0001	0.5564
C_2^T v. C_3^T	0.0486	0.7304	0.0058	0.0004	0.0463	0.0003	0.9734	0.0047
C_2^T v. C_4^T	0.9398	0.0594	0.2624	0.2457	0.3683	0.0735	0.9920	0.9990
C_3^T v. C_4^T	0.0133	0.0039	0.8663	0.1802	0.8270	0.7578	0.9995	0.0698

Table 5.12 A depiction of the results of performing pairwise statistical testing (either the ANOVA test followed by the Tukey multiple comparison test, or the Kruskal-Wallis test followed by Dunn's test) on the top 4 largest clusters identified by the Standard SOM for the R^6 data subset (Figure 5.6 (b)). The top 4 largest clusters are denoted as "C_1^S","C_2^S","C_3^S","C_4^S" and correspond to the clusters referenced in the legend of Figure 5.6 (b). To aid in the visualization of these results, we have highlighted in yellow each pairwise comparison that is statistically significant at the $\alpha = 0.05$ significance level.

	Pop Dens	Perc M	Age M	Age F	MW	MB	FW	FB
C_1^S v. C_2^S	0.6648	0.4084	0.5070	0.5672	0.6105	0.9998	0.0002	0.7269
C_1^S v. C_3^S	0.0014	0.0548	0.1526	0.0018	0.0003	0.0073	0.0014	0.0001
C_1^S v. C_4^S	0.0749	0.9991	0.7377	0.1559	0.2477	0.0647	0.1305	0.0486
C_2^S v. C_3^S	0.0813	0.8310	0.0040	0.0001	0.0087	0.0046	0.9995	0.0092
C_2^S v. C_4^S	0.5561	0.9756	0.1240	0.0268	0.7223	0.1267	0.7885	0.4045
C_3^S v. C_4^S	0.9984	0.5614	0.9982	0.7765	0.3993	0.0000	0.7878	0.9665

Table 5.13 A depiction of the results of performing pairwise statistical testing (either the ANOVA test followed by the Tukey multiple comparison test, or the Kruskal-Wallis test followed by Dunn's test) on the top 4 largest clusters identified by the k-Means algorithm with $k = 6$ for the R^6 data subset (Figure 5.6 (c)). The top 4 largest clusters are denoted as "C_1^{k6}","C_2^{k6}","C_3^{k6}","C_4^{k6}" and correspond to the clusters referenced in the legend of Figure 5.6 (c). To aid in the visualization of these results, we have highlighted in yellow each pairwise comparison that is statistically significant at the = 0.05 significance level.

	Pop Dens	Perc M	Age M	Age F	MW	MB	FW	FB
C_1^{k6} v. C_2^{k6}	0.0116	0.3133	0.4069	0.3331	0.0001	0.0016	0.0017	0.0001
C_1^{k6} v. C_3^{k6}	0.4533	0.2688	0.1677	0.2030	0.1205	0.9780	0.0001	0.9438
C_1^{k6} v. C_4^{k6}	0.9954	1.0000	0.2984	0.3647	0.9986	0.9999	0.0149	1.0000
C_2^{k6} v. C_3^{k6}	0.8487	1.0000	0.0033	0.0031	0.1733	0.0871	0.6486	0.0196
C_2^{k6} v. C_4^{k6}	0.0439	0.8071	0.0157	0.0169	0.0042	0.1255	0.9951	0.0141
C_3^{k6} v. C_4^{k6}	0.4308	0.7338	1.0000	1.0000	0.2902	0.9999	0.8970	0.9923

Table 5.14 A depiction of the results of performing pairwise statistical testing (either the ANOVA test followed by the Tukey multiple comparison test, or the Kruskal-Wallis test followed by Dunn's test) on the top 4 largest clusters identified by the k-Means algorithm with $k = 9$ for the R^6 data subset (Figure 5.6 (c)). The top 4 largest clusters are denoted as "C_1^{k9}","C_2^{k9}","C_3^{k9}","C_4^{k9}" and correspond to the clusters referenced in the legend of Figure 5.6 (d). To aid in the visualization of these results, we have highlighted in yellow each pairwise comparison that is statistically significant at the = 0.05 significance level.

	Pop Dens	Perc M	Age M	Age F	MW	MB	FW	FB
C_1^{k9} v. C_2^{k9}	0.0066	0.2089	0.6338	0.5422	0.0009	0.0002	0.0004	0.0001
C_1^{k9} v. C_3^{k9}	0.5996	0.3454	0.2409	0.3083	0.1101	0.9999	0.0001	0.7674
C_1^{k9} v. C_4^{k9}	0.9999	0.9789	0.6313	0.7512	0.9712	0.9763	0.0022	0.8257
C_2^{k9} v. C_3^{k9}	0.6129	1.0000	0.0115	0.0119	0.4938	0.0008	0.8680	0.0615
C_2^{k9} v. C_4^{k9}	0.1371	0.9928	0.1361	0.1673	0.1637	0.3467	0.7550	0.5525
C_3^{k9} v. C_4^{k9}	0.8012	0.9968	1.0000	1.0000	0.7066	0.9457	0.9727	0.9999

5.4 Discussion

The dual-domain colorectal cancer incidence data set investigated in this chapter served as a real-world case study for the tailored unsupervised learning methodology developed in Chapter 3. The use of this data set enabled us to expand upon the algorithm assessments performed using synthetic data sets in Chapter 4, and further demonstrate the need for a tailored approach when clustering dual-domain data. By commencing this chapter with the analysis of an R^4 subset of the cancer data, analogous to those considered in Chapter 4, we were able to evaluate the robustness and utility of the overall methodology and Tailored SOM design choices outside the con nes of a synthetic data set. The results of this analysis, depicted in Figure 5.3 and Tables 5.3, 5.4, and 5.5, exhibit the advantage of implementing the Tailored SOM as compared to the Standard SOM and the k-Means algorithm, despite the fact that the underlying objective function was generic (i.e., not tailored to characteristics of the California geography). Not only was the Tailored SOM more successful than the Standard SOM and k-Means algorithm in achieving a geographically useful clustering, it also identi ed the most information-dense attribute clustering.

This nding led us to hypothesize about the effect that a modi ed, California-speci c objective function could have on the performance of the Tailored SOM. We explored this idea through the use of J_c (equation (5.1)), a combination objective function de ned with geometric quantities from the geographic domain and statistical measures of features from the attribute domain. We tested and evaluated the clustering capabilities of the Tailored SOM when paired with the combination objective function J_c on the R^4 subset and an R^6 subset of the cancer incidence data. The results of these experiments, described in Sections 5.2.2 and 5.3.1, suggest that incorporating both geographic- and attribute-based metrics in the tailored objective function is advantageous for determining an optimal clustering, particularly in the higher dimensional (R^6) case. Ultimately, the comparisons and evaluations of the Tailored SOM, the Standard SOM, and the k-Means algorithm depicted in this chapter support and further solidify the necessity for unsupervised clustering algorithms that are speci c to the goals of a dual-domain problem.

CHAPTER

6

CONCLUSIONS

6.1 Conclusions and Summary of Contributions

Dual-domain data sets comprised of both geographic and attribute features whose observations are associated, yet distinct in nature, are ubiquitous. There is a plethora of applications for which dual-domain data sets exist and could be used to better understand proximal groupings within the geographic domain, and how said groupings are related to feature similarity in the attribute domain. However, given the unique structure of a dual-domain data set, standard unsupervised clustering algorithms often fail to appropriately partition geographic groups in such a way that maintains meaningful similarity in their corresponding attribute clusters. Therefore, to effectively and ef ciently mine useful clusters from a dual-domain data set, one must employ an algorithm that allows for domain-speci c assessment of feature similarity. Existing dual-domain clustering methodologies [24, 25, 27] rely on multi-pass, iterative algorithms that consider each domain individually and require a priori speci cation of geographic connectedness / constraints and the target number of clusters.

In this thesis, we developed a tailored, unsupervised clustering methodology, called the Tailored Self-Organizing Map (Algorithm 3), specifically designed to achieve the goals of a dual-domain clustering problem in an automated and model-driven manner. Our methodology relies fundamentally on the algorithmic approach of the Standard Self-Organizing Map [17, 18], a well-known clustering algorithm that creates a low-dimensional representation of a higher-dimensional data set through the training of a fully-connected map of nodes and representative weight vectors. We extend the applicability of the Standard SOM to a dual-domain clustering problem by reformulating the underlying Euclidean distance measure used to assess data similarity to include domain-specific biasing parameters capable of better evaluating inter-domain data relationships. The introduction of biasing parameters allows for the construction of an objective function, J, that is used to optimize their values in the context of the broader application.

In Chapter 2, we provided a detailed overview of the algorithmic approaches and example implementations of two standard clustering algorithms: the Standard SOM and the k-Means algorithms [30]. We highlighted the various parameters, design choices, strengths, and limitations of these algorithms in order to provide sufficient context for their subsequent use as comparative approaches in later chapters. Furthermore, we drew comparisons between these two standard algorithms and their underlying clustering regimes.

In Chapters 3 and 4, we formally proposed and developed the Tailored Self-Organizing Map algorithm and the broader tailored unsupervised learning framework designed specifically for dual-domain clustering. At the core of our tailored algorithm is the reformulation of the Euclidean distance measure used by the Standard SOM to identify similarity in data space. Our reformulation, which enables the integration of domain-specific biasing parameters directly into map training, allows for a better evaluation of inter-domain data relationships. The entirety of Chapters 3 and 4 was devoted to testing and experimenting with the Tailored SOM algorithm, and broader learning framework, on three cases of synthetic, dual-domain data sets of varying complexity. We designed these synthetic data sets to rely on parameters (defined in Table 4.1) whose values can be easily manipulated and systemically varied to produce test sets ideal for investigating the robustness of each clustering algorithm of interest. We gave careful consideration to the construction of each case of synthetic data set to ensure that its properties and characteristics were reasonably analogous to

those that one might observe in a real data set.

Next, to assess and compare algorithm performance, we evaluated the tailored objective function J and designed three additional metrics: the Stable Score, the Inlier Score, and the notion of Geographic Feasibility. With these metrics, we were able to consistently evaluate and compare the output of each algorithm in a manner that was both appropriate and meaningful for a dual-domain clustering problem. Without these metrics, we would have relied solely on observational evaluations, which are ultimately of little use when aiming to fully understand the relative strengths and limitations of each clustering algorithm. Even in the absence of a "ground truth" to use as a basis for quantifying the quality of an algorithm's performance, our novel metrics, i.e., the objective function J, the notion of Geographic Feasibility, the Stable Score, and the Inlier Score, each provided structure and formality to our algorithm assessment framework. With these metrics in hand, we considered 20 realizations of each case of synthetic data and trained and compared the results of the following.

- A two-dimensional, 3 3 Tailored SOM consisting of nine nodes, with biasing parameters and that were globally optimized with the geographic objective function J_g
- A two-dimensional, 3 3 Standard SOM consisting of nine nodes
- The k-Means algorithm

Across a total of 60 realizations of three cases of synthetic data sets we observed that the Tailored SOM generally outperforms the Standard SOM and the k-Means algorithms. Most notably, the results produced by the Tailored SOM were, by and large, the most geographically desirable (quanti ed by the evaluation of J_g) and geographically feasible. We also observed the apparent utility of the Stable Score and the Inlier Score as evaluation metrics for results obtained by the Standard SOM and the k-Means algorithm, respectively. This nding provided us with insight into the relative tendencies, strengths, and limitations of the Tailored SOM, the Standard SOM and the k-Means algorithm when applied to a dual-domain data set. For example, in the presence of increased levels of noise within the geographic domain, we noted a tendency of the k-Means algorithm to frequently underestimate the number of true clusters, thus identifying larger, more far-reaching clusters of

data. Whereas under the same data conditions, the Tailored SOM and Standard SOM were more likely to over-partition the geographic space, thus producing smaller clusters with fewer outliers.

In Chapter 5, we continued the investigation of the Tailored SOM in the context of a real data set pertaining to colorectal cancer incidence in the state of California. We amalgamated the cancer incidence data set [37], which provides county-level colorectal incidence rates per 100,000 people in various population groups, with additional county-level demographic information and representative latitude / longitude measurements [43] to create a dual-domain data set well-suited for clustering. For this 10-dimensional data set, we designed and tested a modified, combination objective function J_c. We tailored this function to the geography of California with the goal of identifying information-dense clusters in the attribute domain. We assessed the performance of the Tailored SOM optimized with J_g and J_c on varying subsets of the 10-dimensional data set using both geographic quantities, such as overlapping cluster area, and statistical measures of features from the attribute domain. Similar to the procedure outlined in Chapter 3, we also trained a Standard SOM and the k-Means algorithm as comparisons. The results of these experiments demonstrated the advantage of employing a Tailored SOM as compared to a Standard SOM or the k-Means algorithm, especially in the case where the dimension of the attribute domain exceeds that of the geographic domain. In particular, when compared to the k-Means algorithm, the Tailored SOM achieved significantly better results in terms of uniqueness and delineation of geographic cluster regions, with numerous statistically significant differences in multiple, important attribute domain features, i.e., cancer incidence rates.

6.2 Future Work

In this thesis we developed, tested, and systemically evaluated the Tailored Self-Organizing Map algorithm in the context of 4-dimensional synthetic data and 10-dimensional cancer incidence data. While a considerable amount of groundwork and a number of fundamental aspects related to the proposed methodology were established, there still exist a number of interesting questions that could not be feasibly addressed. First, we were unable to do more than hypothesize possible ways of integrating the neighborhood relation among the SOM nodes into the tailored clustering regime. Given that the neighborhood component of the Self-Organizing Map is inherent to its structure

and training procedure, it would be readily available for use within an objective function, a post-clustering procedure / visualization, or when probing the clusters for information, as demonstrated by the application in [23]. We hypothesize that this feature could be exploited to provide a criteria for merging or tuning clusters after the SOM has been trained. This addition could help remedy the scenario of the SOM populating more nodes with observations of data than necessary to accurately represent true groupings (which may result from training too large of an SOM).

A second interesting feature that future studies could consider is the use of an adaptive, or growing, Self-Organizing Map, like the one presented by Alahakoon et al. in [3] or by Jin et al, in [15]. For example, the growing SOM presented in [3] is able to adapt the map dimension (the total number and arrangement of the nodes) during the competitive training phase, according to a “spread factor" parameter. The implementation of this type of adaptive SOM within the Tailored SOM framework may help to extend the work presented in this thesis to larger, more complex, or noisy dual-domain data sets without requiring pre-training experimentation related to map size and node arrangement.

Finally, there exist ample opportunities for further research in areas related to scalability and further automation of the components within our tailored learning framework. In this thesis, we focused our efforts primarily on building the methodological approach, therefore we limited our experiments to relatively small data sets with no more than 10 dimensions. However, as both the size and dimension of a dual-domain data set grow, several additional opportunities for optimization within our tailored framework arise. For example, a more thorough study on an appropriate choice of global optimization technique would be worthwhile. While the results presented in Section 4.5.3 showed promise for the use of Simulated Annealing in determining the optimal parameter set q, other global optimization methods may prove to be more ef cient for larger data sets and certain formulations of J. Additionally, a closer, more systematic investigation of the sensitivity of the SOM hyperparameters, such as learning rates (k) and $z(k)$ and arrangement of nodes (rectangular lattice, hexagonal lattice, etc.) could be performed. To do so, one may consider employing the Bayesian hyperparameter optimization method [40], which constructs a probabilistic model that is used to determine the optimal values for the hyperparamters of a given machine learning algorithm, based on a validation data set. Implementing this type of hyperparameter optimization method would add additional customization and further streamline the model training portion of our tailored

framework.

Lastly, in Chapter 5 we demonstrated the advantage of incorporating application-specific terms into the modified, tailored objective function J_c. The selection of the terms ultimately used in J_c was a result of much hypothesizing, experimentation, and discussion. Given a large data set for which one possesses little background knowledge or subject matter expertise, the term selection task could prove to be quite arduous. Alternatively, we propose a two-level machine learning architecture, where the upper level aims to learn, through data analysis, a set of candidate terms to be included in the objective function J. Then, the lower level would consist of the model training and parameter optimization regime developed in this thesis. Though considerable research would be required to construct a library of such terms (based in both the geographic and attribute domain), this additional feature would greatly improve the transferability of our tailored learning framework to many dual-domain data sets and applications.

Overall, we believe the area of dual-domain unsupervised clustering to be one rich with intriguing questions and meaningful applications. While in this thesis we focused exclusively on the implementation of domain-specific biasing parameters into the Self-Organizing Map, this concept could be extended to any number of clustering algorithms with an embedded distance metric, such as the k-Means algorithm. Moreover, we hypothesize that the data exploration and visualization technique t-SNE [26, 28], which uses a similarity measure to create a low-dimensional data representation of high-dimensional data, may be amenable to the incorporation of biasing parameters. Nonetheless, the methodology developed throughout this thesis demonstrates the need for, and advantage of, employing a tailored algorithm when seeking to identify geographically feasible and information-dense clusters within a dual-domain data set.

BIBLIOGRAPHY

[1] A.K. Jain M.N. Nurty, P.J. Flynn. “Data clustering: a review”. ACM Computing Surveys 31.3 (1999), pp. 264–323.

[2] Abbe, Emmanuel & Sandon, Colin. Community detection in general stochastic block models: fundamental limits and ef cient recovery algorithms . 2015. arXiv: 1503.00609 [math.PR].

[3] Alahakoon, D., Halgamuge, S.K. & Srinivasan, B. “Dynamic self-organizing maps with controlled growth for knowledge discovery”. IEEE Transactions on Neural Networks 11.3 (2000), pp. 601–614.

[4] Anderburg, M.R. Cluster Analysis for Applications . Academic Press, 1973.

[5] Arthur, David & Vassilvitskii, Sergei. “K-means ++ : the advantages of careful seeding”. Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms (2007), pp. 1027–1035.

[6] Bezdek, James C.Pattern Recognition with Fuzzy Objective Function Algorithms . Springer, 1981.

[7] Boser, Bernhard E., Guyon, Isabelle M. & Vapnik, Vladimir N. “A training algorithm for optimal margin classi ers”. Proceedings of the Fifth Annual Workshop on Computational Learning Theory (1992), pp. 144–152.

[8] Dreo, J. et al. Metaheuristics for Hard Optimization . Springer, 2006.

[9] Dunn, J.C. “A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters”. Journal of Cybernetics 3.3 (1973), pp. 32–57.

[10] Dunn, Olive Jean. “Multiple comparisons using rank sums”. Technometrics 6 (1964), pp. 241–252.

[11] Fisher, R.A. Statistical Methods for Research Workers . Stechert, 1928.

[12] Girvan, M. & Newman, M.E.J. “Community structure in social and biological networks”. Proceedings of the National Academy of Sciences 99.12 (2002), pp. 7821–7826.

[13] Hastie, Trevor, Tibshirani, Robert & Friedman, Jerome. The elements of statistical learning: data mining, inference, and prediction . Springer, 2001.

[14] Holland, P.W., Laskey, K. & Leinhardt, S. “Stochastic blockmodels: First steps”. Social Networks 5.2 (1983), pp. 109–137.

[15] Jin, H. et al. “Expanding self-organizing map for data visualization and cluster analysis”. Information Sciences Journal 163 (2004), pp. 157–173.

[16] Kirkpatrick, S., Jr., C.D. Gelatt & Vecchi, M.P. “Optimization by simulated annealing”. *Science* **220** (1983), pp. 671–680.

[17] Kohonen, T. “Analysis of a simple self-organizing process”. *Biological Cybernetics* **44** (1982), pp. 135–140.

[18] Kohonen, T. “Self-organized formation of topologically correct feature maps”. *Biological Cybernetics* **43** (1982), pp. 59–69.

[19] Kohonon, T. *MATLAB®Implementations and Applications of the Self-Organizing Map*. Unigrafia, 2014.

[20] Kramer, Mark A. “Nonliner principal component analysis using autoassociative neural networks”. *AIChE Journal* **37**.2 (1991), pp. 233–243.

[21] Kruskal, William H. & Wallis, W. Allen. “Use of ranks in one-criterion variance analysis”. *Journal of the American Statistical Association* **47** (1952), pp. 538–621.

[22] Laarhoven, P.J.M. van & Aarts, E.H.L. *Simulated Annealing: Theory and Applications*. Springer, 1987.

[23] Lagus, Krista et al. “WEBSOM for textual data mining”. *Artifical Intelligence Review* **13** (1999), pp. 345–364.

[24] Liao, Z.X & Peng, W.C. “Clustering spatial data with a geographic constraint: exploring local search”. *Knowledge and Information Systems* **31** (2012), pp. 153–170.

[25] Lin, C. R., Liu, K. H. & Chen, M. S. “Dual clustering: integrating data clustering over optimzation and constraint domains”. *IEEE Transactions on Knowledge and Data Engineering* **17** (2005), pp. 628–637.

[26] Linderman, George C. & Steinerberger, Stefan. “Clustering with t-SNE, provably”. *SIAM Journal on Mathematics of Data Science* **1**.2 (2019), pp. 313–332.

[27] Lo, Chia-Hao & Peng, Wen-Chih. “Efficient joint clustering algorithms in optimization and geography domains”. *PAKDD Advances in Knowledge Discovery and Data Mining* **5012** (2008), pp. 945–950.

[28] Maaten, Laurens van der & Hinton, Geoffrey. “Visualizing data using t-SNE”. *Journal of Machine Learning Research* **9** (2008), pp. 2579–2605.

[29] Macías-García, Laura et al. “A study of the suitability of autoencoders for preprocessing data in breast cancer experimentation”. *Journal of Biomedical Informatics* **72** (2017), pp. 33–44. DOI: 10.1016/j.jbi.2017.06.020.

[30] MacQueen, J. “Some methods for classification and analysis of multivariate observations”. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability* **1** (1967), pp. 281–297.

[31] Madhulatha, T. Soni. “An overview on clustering methods”. *IOSR Journal on Engineering* **2** (2012), pp. 719–725.

[32] Marsland, S. *Machine Learning: An Algorithmic Perspective*. CRC Press, 2009.

[33] MathWorks®. *Global Optimization Toolbox Documentation (R2020a)*. Retrieved March 4, 2019 from https://www.mathworks.com/help/gads/.

[34] MathWorks®. *Simulated Annealing Documentation (R2020a)*. Retrieved March 4, 2019 from https://www.mathworks.com/help/gads/simulated-annealing.html.

[35] Mitiche, Amar & Ayed, Ismail Ben. *Variational and level set methods in image segmentation*. Springer, 2010.

[36] Mumford, David & Shah, Jayant. “Optimal approximations by piecewise smooth functions and associated variational problems”. *Communications on Pure and Applied Mathematics* **42** (1989), pp. 577–685.

[37] *National Program of Cancer Registries and Surveillance, Epidemiology, and End Results SEER*Stat Database: NPCR and SEER Incidence – U.S. Cancer Statistics 2001–2016 Public Use Research Database*. November 2018 submission (2001-2016), United States Department of Health and Human Services, Centers for Disease Control and Prevention and National Cancer Institute. Accessed at `www.cdc.gov/cancer/uscs/public-use`. Released June 2019, based on the November 2018 submission.

[38] Romesburg, H. Charles. *Cluster Analysis for Researchers*. Reprint of 2984 edition, with minor revisions. Lulu Press, 2004.

[39] Rousseeuw, Peter J. “Silhouettes: A graphical aid to the interpretation and validation of cluster analysis”. *Journal of Computational and Applied Mathematics* **20** (1987), pp. 53–65.

[40] Snoek, Jasper, Larochelle, Hugo & Adams, Ryan P. “Practical Bayesian optimization of machine learning algorithms”. *Advances in Neural Information Processing Systems 25*. Ed. by Pereira, F. et al. Curran Associates, Inc., 2012, pp. 2951–2959. URL: `http://papers.nips.cc/paper/4522-practical-bayesian-optimization-of-machine-learning-algorithms.pdf`.

[41] Spall, J.C. *Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control*. Wiley, 2003.

[42] Tukey, John W. “Comparing individual means in the analysis of variance”. *Biometrics* **5** (1949), pp. 99–114.

[43] *United States Gazetteer Files*. United States Census Bureau. Accessed at `https://www.census.gov/geographies/reference-files/time-series/geo/gazetteer-files.html`. 2019.

[44] Zhong, Shi & Ghosh, Joydeep. “A unified framework for model-based clustering”. *The Journal of Machine Learning Research* **4** (2003), pp. 1001–1037.