DEPTH-BASED VISUALIZATIONS FOR ENSEMBLE DATA AND GRAPHS

by

Mukund Raj

A dissertation submitted to the faculty of
The University of Utah
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Computing

School of Computing

The University of Utah

August 2018

The University of Utah Graduate School

STATEMENT OF DISSERTATION APPROVAL

The dissertation of **Mukund Raj**
has been approved by the following supervisory committee members:

Ross T. Whitaker ,	Chair(s)	**03 Apr 2018** Date Approved
Robert M. Kirby ,	Member	**03 Apr 2018** Date Approved
Suresh Venkatasubramanian ,	Member	**03 Apr 2018** Date Approved
P. Thomas Fletcher ,	Member	**03 Apr 2018** Date Approved
Alon Efrat ,	Member	**11 Apr 2018** Date Approved

by **Ross T. Whitaker** , Chair/Dean of
the Department/College/School of **Computing**
and by **David B. Kieda** , Dean of The Graduate School.

ABSTRACT

Ensemble data sets appear in many domains, often as a result of a collection of solutions arising from the use of different parameters or initial conditions in simulations, measurement uncertainty associated with repeated measurements of a natural phenomenon, and inherent variability in natural or human events. Studying ensembles in terms of the variability between ensemble members can provide valuable insight into the generating process, particularly when mathematically modeling the process is complex or infeasible.

Ensemble visualization is a way to understand the underlying generating model of data by studying ensembles of solutions or measurements. The objective of ensemble visualization is often to convey characteristics of the typical/central members, outliers, and variability among ensemble members. In the absence of any information about the generative model, a family of nonparametric methods, known as data depth, provides a quantitative notion of centrality for ensemble members. Data-depth methods also form the basis of several ensemble visualization techniques, including the popular Tukey boxplot.

This dissertation explores data depth as a basis for visualizing various types of data for which existing visualization methods are either not directly applicable or present significant limitations. Such data include ensembles of three-dimensional (3D) isocontours, ensembles of paths on a graph, ensemble data in high-dimensional and inner-product spaces, and graphs. The contributions of this dissertation span the following three aspects of data-depth based visualizations: first, development of new data-depth methods that address the limitations of existing methods for computing center-outward order statistics for various types of ensemble data; second, development of novel visualization strategies that use existing and proposed data depth methods; and third, demonstration of the effectiveness of the proposed methods in real motivating applications.

CONTENTS

LIST OF FIGURES

NOTATION AND SYMBOLS

$\mathcal{X}$	upper case script letters denote sets
$\mathbf{A}$	upper case bold letters denote matrices
$\mathbf{x}$	lower case bold letters denote column vectors
x_i	lower case letter with subscript denotes member of set/vector
x_{ij}	dual subscript indicates an element of a matrix
X	upper case letters denote a random variable
(x_1, x_2, x_3)	parenthesis denotes an ordered relationship

CHAPTER 1

INTRODUCTION

Recently, a combination of sensors, personal devices, and computational and communication infrastructure has resulted in large amounts of complex data. Analyzing such data to gain useful insights is important in a range of applications. Often the data are in the form of a group or collection of related entities called *ensembles*. These entities or *ensemble members* typically share an underlying generating process while also displaying variability among members. This variability can originate from various causes, such as variability in parameters or initial conditions in simulations, measurement uncertainty associated with repeated measurements of a natural phenomenon, and inherent variability in natural or human events. Ensembles that provide information about variability between members can therefore be valuable in understanding the generating process, particularly when mathematically modeling the process is complex or infeasible.

Data in applications spanning several domains originate from complex underlying processes. For example, meteorologists use complex models that are highly sensitive to parameters and initial conditions to generate ensembles of prediction data to understand the probabilities of specific weather and climate-related events [42, 101]. In the area of hurricane prediction, meteorologists analyze ensembles of potential hurricane trajectories obtained by a Monte Carlo simulation [60]. Researchers in the medical community use collections of magnetic resonance imaging (MRI) and functional MRI (fMRI) images of the brain to understand changes that occur in brain structure during neurological disorders, such as Alzheimer's disease [4]. Scientists study the process of protein folding by running multiple simulation runs of the folding process [8]. The proteins transition to a final stable state after passing through a set of stochastically determined intermediate states. In such applications, we can gain insights about the generating process by understanding the ensemble data.

Visualization can be a useful asset for understanding data. Scholars have been using visualization as a tool to understand and convey data for hundreds of years [72, 83]. Visualization brings humans into the process of data analysis, and allows them to take advantage of the high bandwidth of the human visual perception system to comprehend data [76]. The advantage of suitably chosen visualization is clearly seen in scatter plots of the well-known Anscombe's quartet [7], which consists of four multivariate data sets with identical mean, mode, and median (see Figure 1.1). The difference between the data sets is immediately apparent in scatter plot visualizations. Visualizations help humans use the data to quickly gain insights and form hypotheses. Benefits of visualization are particularly relevant when dealing with complex data such as ensemble data sets. Consequently, a significant amount of research has been carried out to develop effective ensemble visualization techniques.

Ensemble visualization techniques can be broadly classified into the following three classes: parametric methods that convey probabilities for features of interest, enumeration methods that directly display ensemble members, and nonparametric methods that convey summary statistics. Figure 1.1 shows the discrepancies that can occur between

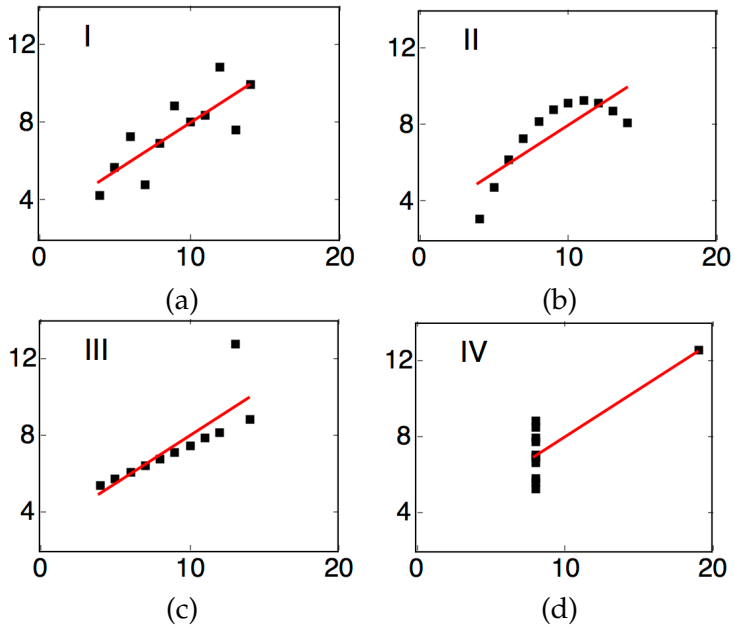


Figure 1.1. The Anscombe quartet. The four data sets in (a), (b), (c), and (d) have the same summary statistics: mean, standard deviation, and correlation. The red line shows the linear regression line for each data set.

parametric models and enumeration. Whereas parametric methods require some knowledge or assumptions regarding the generating process, enumeration can occlude details in the case of large ensembles and complex objects. In contrast, nonparametric summaries have proven to be effective for characterizing large and complex ensemble data without any assumption about the underlying generating processes [86]. This class of ensemble visualization techniques typically involves two steps: 1) identification of key features of interest in ensembles; and 2) effective visualization of those features.

In ensemble data, key features of interest often include summary statistics, such as most representative members (e.g., mean or median); least representative members, also known as outliers; and the variability in the ensemble. For an ensemble of univariate points, a Tukey *boxplot* highlights the key features, such as the median, outliers, 50% band, and 100% band [108] (see Figure 1.2). Given a univariate random variable X, a median is defined as the point in the distribution with the lowest expected sum of distances from all points in the distribution. For a set of points, $\mathcal{S}$, randomly sampled from X, a *sample* median is defined as the point in $\mathcal{S}$ with the lowest sum of distances to all other points in $\mathcal{S}$. For univariate points, the median is situated at the midpoint of the data set or the distribution, such that there is an equal probability for a random sample of falling above or below it (see Figure 1.3a and Figure 1.3b). Another important summary statistic is the *mean*, which is defined, with respect to a distribution, as the point that minimizes the expected sum of *squared* distances from all points in a distribution. The sample mean, also known as the *average*, minimizes the sum of squared distances from all points in a sample set (see Figure 1.3c). For univariate data, both L_1 and L_2 distance metrics lead to identical

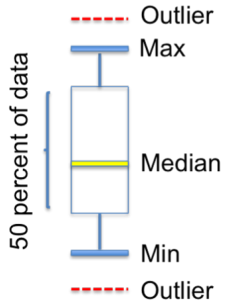


Figure 1.2. A Tukey boxplot summarizes 1D data by showing order-based statistics, such as the median, 50% band, and outliers.

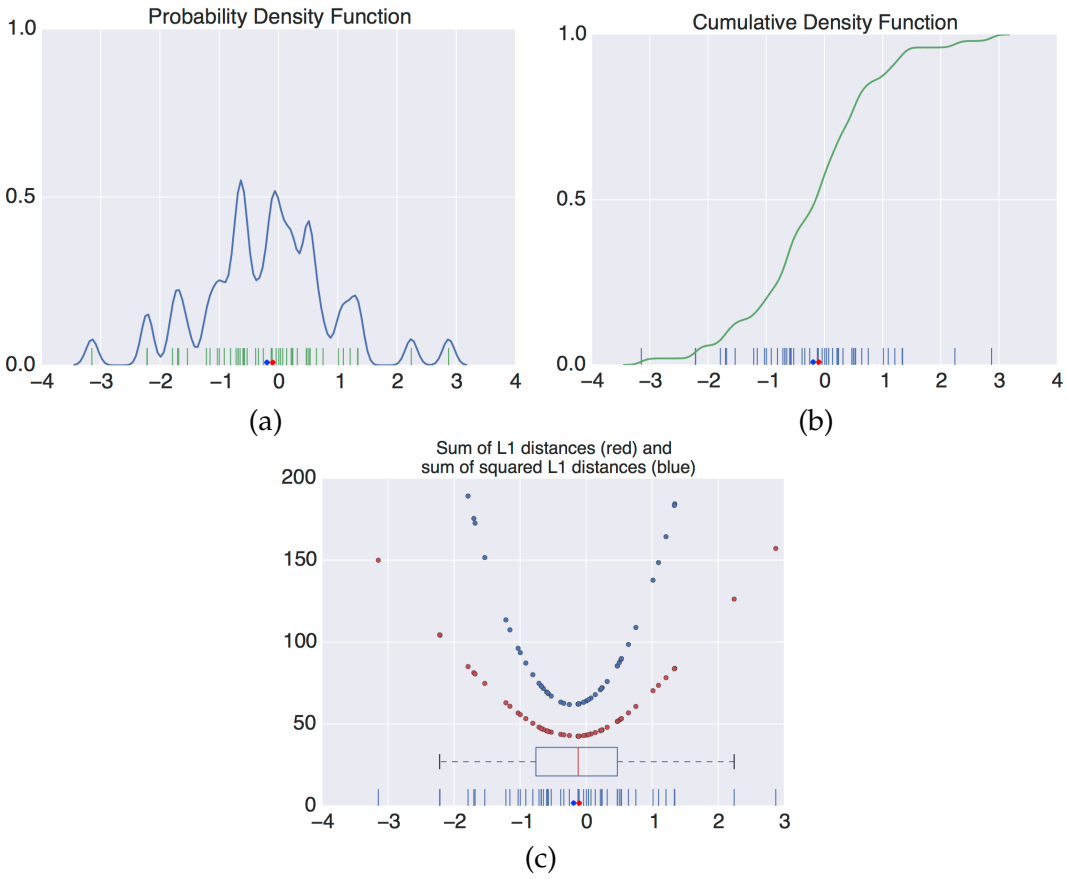


Figure 1.3. Summary statistics in univariate data. (a) A random sample of points from a univariate normal distribution and their probability density function (PDF). The median and mean are indicated using blue and red markers, respectively, along the x axis. (b) A cumulative density function (CDF) plot and (c) sum of L_1 distances and sum of squared L_1 distances for each point in the sample. We see that the minimas are located at the median and mean, respectively.

results. Given two points $\mathbf{x}$,$\mathbf{y} \in \mathbb{R}^d$, where $d \geq 1$, the L_1 and L_2 distances can be stated as:

$$d_{L1}(\mathbf{x}, \mathbf{y}) = \sum_i |x_i - y_i| \tag{1.1}$$

$$d_{L2}(\mathbf{x}, \mathbf{y}) = \left[\sum_i (x_i - y_i)^2 \right]^{\frac{1}{2}} \tag{1.2}$$

where $|\cdot|$ denotes the absolute value.

The sum of distances from all points in the data set also provides a way to order or sort the points in the data set. This approach provides a center-outward ordering of points starting with the median (lowest sum value). The Tukey boxplot uses such a center-outward ordering of points to determine other summary statistics, such as the 50%

and 100% bands. The points that have sum of l_1 distances in the lower 50 percentile are considered to be in the 50% band (see Figure 1.3d). The span of the 50% band is expanded about the median by a constant factor to obtain the 100% band. A center-outward order using sum of distances (L_1 or L_2), despite being useful for visualizing univariate data, is not as effective for visualizing more complex data types such as multivariate points. This lack of effectiveness is because distance-based approaches for determining center-outward order do not fully capture the structure of multivariate data since they fail to consider correlations in the data (see Section 2.2.1). There exists a more general measure, however, that is effective in determining center-outward order statistics for ensembles of complex, multivariate data.

Data depth is a measure of how central or outlying a given point is with respect to a multivariate data cloud or its underlying distribution [63]. The introduction of several notions of data depth has led to the development of a family of data-depth methods for quantifying the centrality of points in multivariate ensemble data. These methods provide a numeric, center-outward order statistic for multivariate data. In this dissertation, we use the term "*order structure*" to refer to center-outward order statistics, which describe the inward-outward relationship among data members with respect to some property, such as data depth. Data-depth methods have also been proposed for a variety of data types such as multivariate points [93] and curves [73], functions [66], and isocontours [115]. For each type of data, researchers have developed visualization strategies that use order statistics provided by data depth to convey key features in the data. Visualization practitioners over the past several years have been using data-depth-based visualizations to gain insights into and understand the general structure of ensemble data [61, 88]. However, there are several situations where existing data-depth-based-visualizations are either not applicable or could potentially fail to capture important features.

One shortcoming of state-of-the-art data-depth-based methods is their limited ability to tackle data types that are not in $\mathbb{R}^d$, where $d \in \mathbb{N}^+$; for example, path ensembles on graphs. A graph is a collection of entities (nodes) and relationships (edges). A path on a graph is a sequence of nodes on a graph that share an edge. Path ensembles on graphs are a type of ensemble data that appear in several domains. For example, packets traveling from a source node to a destination on the Internet often take different paths as determined by

routing algorithms [19]. Similarly, a variety of paths are typically possible for traveling to a destination on road networks, which are often modeled as paths on graphs [46]. Another domain where path ensembles appear as a mode of representing information is in molecular dynamics simulations of protein folding [8]. The energy landscape of protein structural configurations is often modeled as a graph. In such models, the path taken by a folding protein to reach a stable state involves stochastic transition probabilities along edges leading to an ensemble of paths over multiple simulation runs. In such applications, there is a need to understand the structure of path ensembles in terms of typical and outlier members.

In addition to paths, visualizing the *importance* of nodes in the context of relationships on a graph is also important in many applications, particularly in social networks [17]. A common approach to determine the importance of nodes is to consider more *central* nodes as more important [15]. The graph theory literature offers several methods to define centrality using the topology (and edge weights) of a graph, including quantifying the centrality of a node based on the number of incident edges, the sum of distances to other nodes, and the number of shortest paths passing through the node [37]. Researchers have proposed various methods to visually convey the centrality of nodes using node-link drawings [10, 16]. However, those methods often do not accurately represent internode distances, which are an important aspect of graphs in node-link drawings.

Another type of data that is challenging to handle with existing visualization methods is data in high-dimensional spaces ($d > 2$). Popular state-of-the-art methods for visualizing high-dimensional data determine a low-dimensional, typically two-dimensional (2D), embedding of the data by attempting to preserve distances between points/members as much as possible. Such methods often do not preserve the order structure of data; essentially, they fail to ensure that more central members in the data appear more central in the embedding. The inaccurate representation of order structure can be attributed to the *exclusive* focus of current dimensionality reduction methods on preserving the distances between ensemble members while ignoring aggregate statistical properties of the data, such as the order structure.

Dimensionality reduction methods are also used to visualize ensembles of data in implicit feature spaces that are defined using inner products. In such cases, the dimension-

ality as well as coordinate axes of the feature space are unknown, making it infeasible for existing data-depth methods to effectively determine the order structure of data. Other challenging data types include ensembles of three-dimensional (3D) shapes, which are often seen in the domain of medical imaging, and remain difficult to visualize due to the structural complexity of such data and the problem of occlusion.

1.1 Contributions

This dissertation introduces new visualization techniques for several kinds of ensemble data that address various shortcomings of state-of-the-art data-depth- or metric-based visualizations, including novel techniques to compute center-outward order statistics and novel visualization strategies based on such order statistics. Specifically, the contributions of this dissertation are as follows:

- *Evaluation of 3D shape alignment using ensemble visualization.* We show that ensemble visualization techniques can be effective for evaluating 3D shape alignment. We also extend the contour boxplot visualization technique for 3D shape ensembles.

- *A method for computing data depth and a visualization strategy for path ensembles on a graph.* Given an ensemble of paths on a graph, we introduce the path band depth method for determining data depth for paths by considering the global structure of the path ensemble. We also introduce path boxplot visualization for visualizing path ensembles using path band depth.

- *A method for visualizing node centrality while also preserving internode distances in node-link diagrams of graphs.* We determine node positions in a graph drawing by minimizing an energy function that contains two terms: the first term penalizes deviation between internode distances along edges and the embedding, and the second term penalizes drawings that inaccurately represent node centrality with regard to the graph structure.

- *A method for visualizing high-dimensional data using order statistics.* We determine a lower dimensional embedding for high-dimensional data by minimizing an energy function that contains two terms: the first term penalizes deviation between distances in the original or *intrinsic* space, and the second term penalizes embeddings

where points are projected to positions that inaccurately represent their order structure in the intrinsic space. We also introduce two visualization strategies, field overlay plot and projection boxplot, to visualize the lower dimensional embedding.

- *A method to compute data depth in high-dimensional inner product spaces.* Given an ensemble of points in any inner product space, the ellipse band depth method computes data depth by estimating the probability of points to be contained in a set of ellipses described by a randomly chosen pair of points.

1.2 Overview

Chapter 2 provides an overview of the technical details that are important to understand the work in this dissertation. The technical details include a discussion on order and rank statistics, the notion of data depth, useful properties of data-depth measures, and various classes of data-depth techniques.

Chapter 3 demonstrates the utility of using various ensemble visualization techniques for the task of evaluating 3D shape alignment in the context of a specific medical imaging application: constructing brain atlases, which are a representative image for an ensemble of brain MRI images. The visualization techniques include averaging, enumeration, and a 3D extension of the contour boxplot visualization [115]. The work described in this chapter is published in *IEEE Computer Graphics and Applications* [89].

Chapter 4 introduces a novel method to compute data depth for paths on a graph called *path band depth*, which considers the global structure of the paths. This chapter also provides a theoretical proof to show that the proposed method exhibits a desirable property with regard to data depth (Section 2.2). Next, this chapter introduces a visualization strategy for path ensembles based on path band depth called *path boxplot* and demonstrates its utility with synthetic data and real data sets. The work described in this chapter is published in the *Journal of Computational and Graphical Statistics* [90].

Chapter 5 introduces a novel technique to determine positions of nodes in a node-link diagram of a graph such that the internode distances as well as the importance or centrality of the nodes are conveyed correctly, as much as possible. This chapter also describes an algorithm for determining node positions, and a novel visualization strategy called *anisotropic radial layout* based on the resultant node positions. Finally, the chapter

includes results using both synthetic and real data sets. The work described in this chapter is published in the *Proceedings of the 25th International Symposium on Graph Drawing and Network Visualization (2017)* [91].

Chapter 6 introduces a technique for embedding high-dimensional data in lower dimensional spaces such that order structure as well as distances between points in the high-dimensional space is preserved, as much as possible. This chapter also describes a modification to the technique that focuses specifically on dealing with multimodal data. Finally, the chapter presents two visualization strategies that highlight the order structure and related features such as the median and outliers and demonstrates the utility of the proposed method using high-dimensional data from various application domains. The work described in this chapter is to appear in *Computer Graphics Forum (EuroVis) 2018*.

Chapter 7 describes a novel method for computing data depth for high - data, called *ellipse band depth*, that overcomes several shortcomings of existing methods for computing depth such as half-space depth and functional depth. This method is able to efficiently compute depth in very high-dimensional spaces using only inner product information while also capturing the correlations in the data. Data depth computed using this method can also be used in conjunction with the technique introduced in Chapter 6 for visualizing high-dimensional data.

Chapter 8 provides a concluding discussion for this dissertation. This chapter also includes a discussion of shortcomings of the proposed methods as well as several interesting avenues for further work.

CHAPTER 2

TECHNICAL BACKGROUND

This chapter provides a brief description of the technical background relevant to the work in this dissertation. First, the chapter discusses order and rank statistics in ensemble data. Next, it discusses the notion of data depth, various useful properties of data depth, and three general approaches for computing data depth.

2.1 Order and Rank Statistics

This dissertation deals with understanding the statistical structure of ensemble data. An important aspect of understanding the structure of ensembles is determining how central or typical various members are with respect to the ensemble. Order and rank statistics are descriptive statistics, which when selected appropriately, can help in determining the centrality of ensemble members. Methods proposed in this dissertation rely on center-outward order and rank statistics to identify and highlight key members of interest in ensemble data, such as median (most central) and outliers (least central).

2.1.1 Order Statistics

In the context of statistics, the k^{th} order statistic in a sample set is equal to the value of its k^{th} smallest value. For example, if $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ denotes a set of values that are sampled from a random variable X, and $\mathcal{X}_{(i)}$ is the i^{th} smallest value in $\mathcal{X}$, where $\{1 \leq i \leq n\}$, then $\mathcal{X}_{(i)}$ is called the i^{th} order statistic of $\mathcal{X}$. Determining order statistics requires a way to sort the members in the sample set based on a specified criterion. If X is a numeric random variable, ways to sort sample members include using the member value, member absolute value, or even the sum of distances from all other members in the sample set.

2.1.2 Rank Statistics

Given a set of members $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ sampled from a random variable X, the rank vector, $\mathbf{r} = (r_1, r_2, \ldots, r_n)$, is defined such that

$$r_i = \sum_{j=1}^{n} \delta(x_i - x_j), \tag{2.1}$$

where $\delta(\cdot)$ is the following indicator function:

$$\delta(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases}. \tag{2.2}$$

A rank statistic is any value that is a function of the rank vector. For example, the median of a set of numbers is the highest ranking member when the numbers are sorted based on a center-outward order statistic such as sum of distances from all numbers in the set.

2.1.3 Order and Rank Statistics in Visualization

Key features of interest in ensemble data are often central/outlier members and variability, which can be identified using center-outward order and rank statistics. Consequently, such statistics play an important role in visualizing many kinds of ensemble data. Generating ensemble visualizations using order and rank statistics typically involves two main steps: 1) analysis step: using an appropriate order/rank statistics to sort the ensemble members and identify key members and features; and 2) visualization step: highlighting the relevant information using visualization strategies specifically designed for the particular type of ensemble data. The Tukey boxplot, discussed in Chapter 1, is an example of ensemble visualization involving these two steps—analysis and visualization.

The Tukey boxplot conveys several interesting quantities derived from order and rank statistics, such as the median (rank statistic); 50% band, which comprises the innermost half of the members (rank statistic); and the 100% band and outliers (determined using order and rank statistics). The design of the Tukey boxplot has inspired several other ensemble visualization techniques for a range of different data types. These techniques include the bagplot for bivariate data [93], functional boxplot for ensembles of functions [102], curve boxplot for ensembles of multivariate curves [73], surface boxplot for ensembles of 2D functions [40], and contour boxplot for ensembles of sets/isocontours [115] as in Figure 2.1. These visualizations highlight key members such as the median and outliers as is

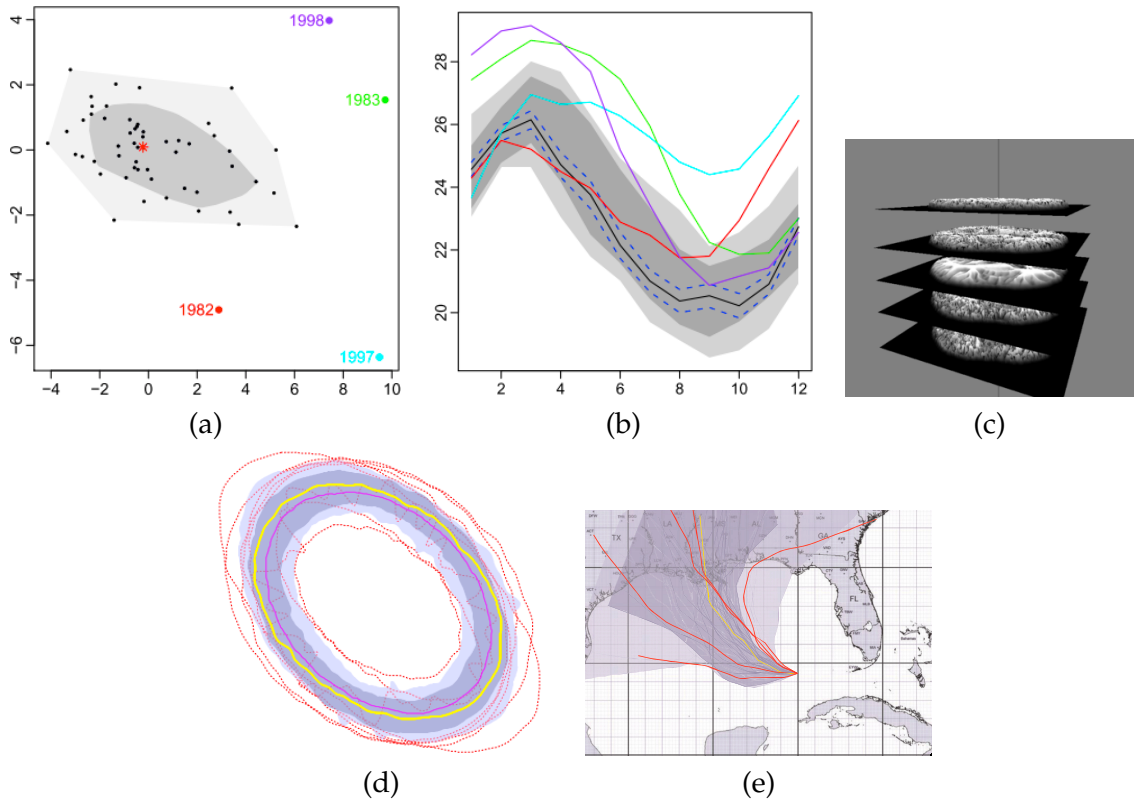


Figure 2.1. Examples of existing depth-based visualizations. (a) Bagplot for bivariate points (principal components of sea surface temperature data) [48] © 2010 Taylor and Francis Group, (b) fuctional boxplot for ensemble of functions (sea surface temperatures for 12 months) [48] © 2010 Taylor and Francis Group, (c) surface boxplot for ensembles of 2D fields [40] © 2014 John Wiley and Sons, (d) contour boxplot for curves (synthetically generated) [115] © 2013 IEEE, and (e) curve boxplot for ensemble for curves (hurricane paths) [73] © 2014 IEEE.

the case in the Tukey boxplot. The analysis step for these visualizations uses specialized methods designed for particular data types, such as functions, curves, or sets, to compute order and rank statistics.

2.2 Data Depth

Data-depth methods provide a quantitative notion of centrality for multivariate data. A *depth function* that evaluates the depth of a point, $\mathbf{x} \in \mathbb{R}^d$, with respect to a random variable $X \in \mathbb{R}^d$ has the general form: $D(\mathbf{x}; F_X) : \mathbb{R}^d \times \mathcal{F} \rightarrow [0, 1]$, where F_X is a probability distribution over X, $\mathcal{F}$ is the class of probability distributions on Borel sets of $\mathbb{R}^d$, and $F_X \in \mathcal{F}$. The notion of centrality provided by data-depth methods has been useful in several

applications, such as visualization [93, 102], classification [111], and outlier detection [21]. Researchers have proposed several data-depth methods for analyzing multivariate data, such as L_2 depth, zonoid depth [30], half-space depth [107], etc. Although the specific characteristics of the various data-depth methods may vary, they rely on a set of common desirable properties [124]: affine invariance, maximality at center, monotonicity relative to deepest point, and vanishing at infinity.

- P1: Affine Invariance

  Assumption: Let F_X be a probability distribution over a Borel set of $\mathbb{R}^d$.

  Property: The depth of any point $\mathbf{x} \in \mathbb{R}^d$ should be independent of the underlying coordinate system, particularly the scales of the measurement along each axis.

$$D(\mathbf{A}\mathbf{x} + \mathbf{b}; F_{\mathbf{A}X+\mathbf{b}}) = D(\mathbf{x}; F_X). \tag{2.3}$$

- P2: Maximality at Center.

  Assumption: Let F_X be an angularly symmetric distribution over a Borel set of $\mathbb{R}^d$ with a uniquely defined center, $\mathbf{c}$.

  Property: The depth at the center should be highest with respect to all other points in the distribution.

$$D(\mathbf{c}; F_X) = \sup_{x \in \mathbb{R}^d} D(\mathbf{x}; F_X). \tag{2.4}$$

- P3: Monotonicity Relative to Deepest Point. *Assumption:* Let F_X be an angularly symmetric distribution over a Borel set of $\mathbb{R}^d$ with a uniquely defined center, $\mathbf{c}$.

  Property: As a point $x \in \mathbb{R}^d$ moves away from center $\mathbf{c}$ in the direction of any fixed ray, the depth at x should decrease monotonically.

$$D(\mathbf{x}; F_X) \leq D(\mathbf{c} + \alpha(\mathbf{x} - \mathbf{c}); F_X) \ , \ \forall \alpha \in [0, 1]. \tag{2.5}$$

- P4: Vanishing at Infinity.

  Assumption: Let F_X be a probability distribution over a Borel set of $\mathbb{R}^d$.

  Property: The data depth of a point $\mathbf{x}$ should approach zero as $||\mathbf{x}||$ approaches infinity.

$$\lim_{||\mathbf{x}|| \to \infty} D(\mathbf{x}; F_X) = 0. \tag{2.6}$$

2.2.1 Data-Depth Approaches

Several notions of data depth exhibit the above properties. These notions can be broadly classified into three classes according to the approach of determining depth: distance metric, weighted mean, and space partition [77].

2.2.1.1 Distance metric

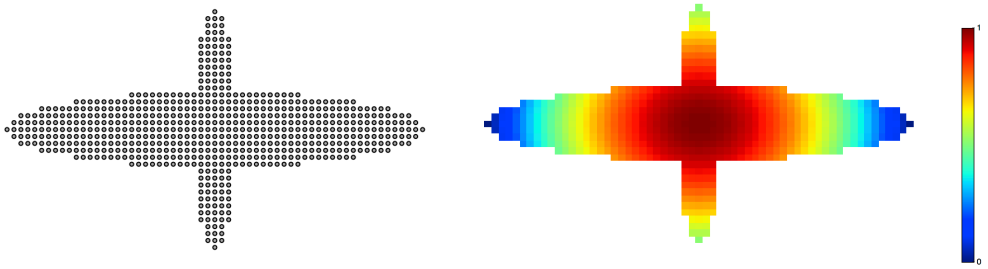
In depth functions based on a distance metric, depth of a point $\mathbf{x}$ is commonly defined as the inverse of the expected distance from other points in a distribution. Various distances metrics, such as L_2 or Mahalanobis distances, can be used to obtain depth functions with different properties. For example, the L_2 depth can be stated as:

$$D(\mathbf{x}; X) = (1 + E[||\mathbf{x} - X||_2])^{-1} \tag{2.7}$$

Although distance-based depth measures are simple and efficient to compute, they are often unable to properly account for the shape of anisotropic distributions. For example, in anisotropic distributions, points at extremes along a minor axis would tend to be assigned depth values similar to points along a major axis that are closer to the center (see Figure 2.2).

2.2.1.2 Weighted mean

Weighted-mean regions are nested convex regions that are centered around the geometric center of a distribution. These convex regions are composed of weighted means



(a) (b)

Figure 2.2. Distance-based depth for an anisotropic distribution. (a) A synthetic ensemble of points on a grid that comprises two crossing ellipses with horizontal and vertical major axes. (b) Normalized L_2 depth values for the point ensemble shown using a heatmap. We notice that points at the extremities of the vertical ellipse have depth values similar to the inner points in the horizontal ellipse.

of the data members, with a general set of restrictions on the weights that ensure their nested arrangement. This arrangement of nested convex weighted-mean regions is then used to determine the data-depth value of each data member. Various strategies of the assigning weights lead to different notions of weighted-mean depths [77]. An example of a weighted-mean depth is the Zonoid depth [30].

Let $x, x_1, \ldots, x_n \in \mathbb{R}^d$. Then the zonoid depth of x with respect to $x_1, \ldots, x_n$ is

$$D_{\text{zonoid}}(\mathbf{x}; X) = \sup\{\alpha : \mathbf{x} \in D_\alpha(x_1, \ldots, x_n)\},$$

where

$$D_\alpha(x_1, \ldots, x_n) = \left\{ \sum_{i=1}^{n} \lambda_i x_i : \sum_{i=1}^{n} \lambda_i = 1, 0 \leq \lambda_i, \alpha\lambda_i \leq \frac{1}{n} \text{ for all } i \right\}.$$

Here $D_\alpha(\cdot)$ denotes the weighted-mean region that indicates the region with a depth greater than α and is also known as the α-trimmed region. Note that when $\alpha = 1$, the weighted-mean region collapses to the mean of the data, whereas $0 \leq \alpha \leq \frac{1}{n}$ leads to a weighted-mean region that is the convex hull of data.

Weighted-mean-based formulations of depth, in comparison to the distance-based formulations, are more effective in capturing the shape of the distribution (see Figure 2.3). However, weighed-mean formulations are more susceptible to outliers in data as the shape of the weighted-mean regions, and consequently data depth, can be strongly influenced by pathological outliers. They are also more computationally expensive and often involve solving an optimization problem.

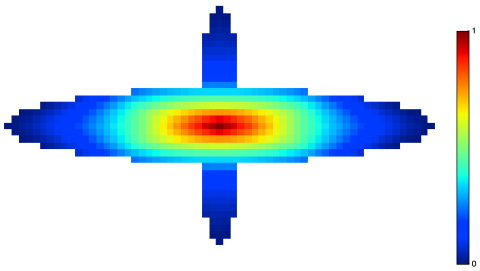


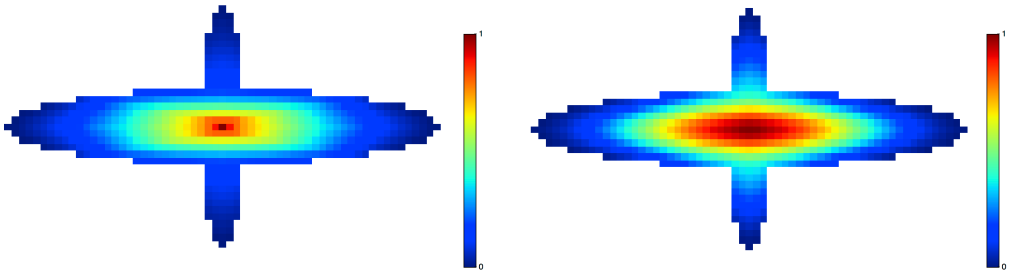
Figure 2.3. Normalized zonoid depth values for point ensemble in Figure 2.2a shown using a heatmap. Points at the extremities of both ellipses have been assigned the lowest depth values, unlike in the case of L_2 depth (see Figure 2.2b).

2.2.1.3 Space partition

Another class of data-depth techniques relies on partitions of the data space. These methods often involve two general steps: 1) a method to partition the space based on various combinatorial subsets of the ensemble data; and 2) determining whether the ensemble members are located within the partitioned subspace. Methods based on space partitioning can be further classified into the following two categories based on the kind of partitions: half-space and band partitions.

The half-space depth, introduced by Tukey, relies on half-space partitions and is a popular way to compute data depth for multivariate data in Euclidean space. Half-space depth of any point $\mathbf{x} \in \mathbb{R}^d$ with respect to $\mathcal{X} \subset \mathbb{R}^d$ is determined as the smallest number of data points from $\mathcal{X}$ that can be contained in a closed half space also containing $\mathbf{x}$. Unlike weighted-mean-based depths, half-space depth is robust to pathological outliers. However, the computing half-space depth can quickly get intractable as the dimension of the data spaces increases.

Several methods to compute data depth rely on the definition of a *band*, which is a partition of data space that is determined by a subset of data. For example, in *simplicial depth*, the band is defined as the convex hull, and the simplicial depth of any point $\mathbf{x} \in \mathbb{R}^d$ with respect to $\mathcal{X} \subset \mathbb{R}^d$ is determined as the probability of it being contained in a convex hull determined by $d + 1$ points randomly chosen from $\mathcal{X}$. Band-based methods or *band-depth* methods are also able to capture the shape of a distribution better than half-space depth, particularly, in the case of nonconvex-shaped multivariate distributions (see Figure 2.4).



(a) (b)

Figure 2.4. Normalized (a) half-space depth and (b) simplicial depth for point ensemble in Figure 2.2a shown using heatmaps. Simplicial depth shows a larger spread of depth values for points that lie along the vertical ellipse.

Recently introduced band-depth methods focus on computing data depth for ensembles of several non-Euclidean data types such as functions [66] and sets [115].

2.2.1.4 Extensions to graphs

The notion of centrality also plays an important role in the domain of graph analytics, particularly for social network analysis, where more central or *deeper* nodes are considered more important [37]. Researchers have proposed several methods to quantify the centrality of nodes on a graph, such as degree [37], closeness [94], and betweenness centrality [37] (see Figure 2.5). These methods determine node centrality by relying only on the graph structure, i.e., edge connectivity and weights, and can be considered extensions of data-depth methods to the graph domain. For example, the closeness centrality of a node is the reciprocal of the sum of its graph theoretical distance to all other nodes, and is analogous to the data-depth approach based on distance. Betweenness centrality counts the number of shortest paths that pass through a node, which in essence is similar to the data-depth approach based on band partition. In the case of betweenness centrality, the shortest paths can be considered to be bands described by pairs of nodes on the graph.

A significant part this dissertation deals with the visualization of graph-based data, such as nodes and paths on a graph. In Chapter 4, we introduce *graph-geodedic-hull-band depth*, which is an extension of simplicial depth to graphs. In Chapter 5, we use graph centrality to determine an order structure for nodes on the graphs, which is then used for drawing visualizations that correctly convey the order structure.

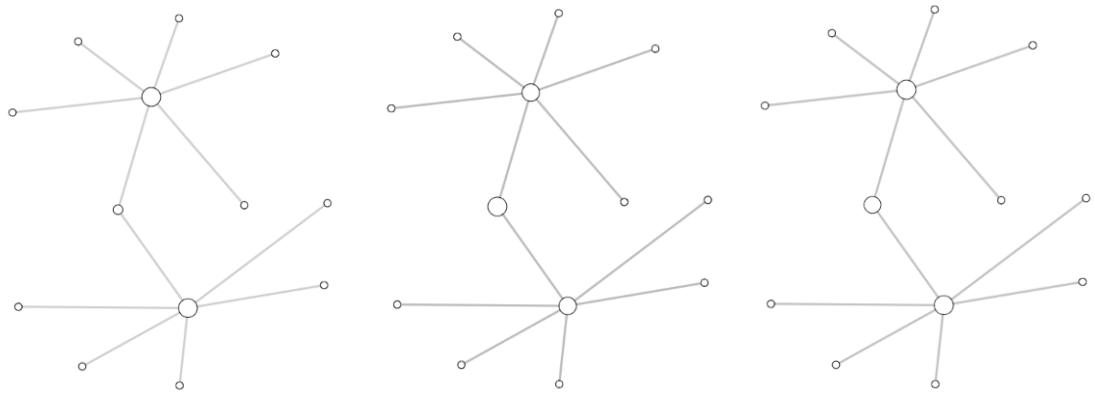


Figure 2.5. A node-link diagram of a synthetic graph where node sizes encode (a) degree, (b) closeness, and (c) betweenness centrality. Degree centrality is higher for nodes with more neighbors, closeness centrality is higher for nodes that have a lower sum of distances to other nodes, and betweenness centrality is higher for nodes that lie on more of the shortest paths between other nodes.

CHAPTER 3

EVALUATING SHAPE ALIGNMENT VIA ENSEMBLE VISUALIZATION

Portions of this chapter have been reproduced with permission from IEEE and is based on material published in CG&A, Evaluating Shape Alignment via Ensemble Visualization, M. Raj, M. Mirzargar, J.S. Preston, R.M. Kirby and R.T. Whitaker, vol. 36, 2016, pp. 60-71 [89].

3.1 Introduction

As computational tools for simulation and data analysis have matured, researchers, scientists, and analysts have become interested in understanding not only the *deterministic* output of these tools, but also the *uncertainty* associated with their computations and/or data collection. Consequently, there is an increasing interest in uncertainty quantification (UQ) as an integrated part of simulation and data science in a wide variety of science and engineering disciplines. UQ views the simulation and data science pipelines as a random process containing possibly both epistemic (i.e., reducible) and aleatoric (i.e., by chance) uncertainty. Quantification efforts in this random process are divided into roughly two categories: 1) efforts to understand the uncertainty and/or variability of the process through an examination of instances (samples) of the process; and 2) efforts to determine models (e.g., probability theory) that capture the nature of the process. The first of these categories, and the focus of this study, utilizes an *ensemble* of solutions meant to capture the inherent variability or uncertainty in a computational or data science pipeline. Although we assume that the variability seen in the ensemble can be attributed to some condition or property of the generating process, we do not assume that articulation of the process via a mathematical model is straightforward, and hence we have only the ensemble members themselves to gain insight into the originating process.

Studying an ensemble in terms of the *variability* or dispersion between ensemble mem-

bers can provide useful information and insight about the underlying distribution of possible outcomes. Correspondingly, ensemble visualization can be a powerful way to study this variability; however, a key challenge here is to be able to convey the variability among ensemble members while preserving the *main features* they share. Preservation of these features is particularly challenging in cases where the ensemble members are not *fields* over which statistical operations such as mean and variance are well defined, but instead are derived or extracted features such as isosurfaces.

In this chapter, we examine the effectiveness of the contour boxplot technique [115], a descriptive summary analysis and visualization methodology, in the context of a particular medical data science application: brain atlas construction and analysis. We conducted an expert-based evaluation of the visualization of ensembles generated through the alignment of shapes using the deformation of images in the construction of atlases (or templates) for brain image analysis. To accomplish this evaluation, we constructed a prototype system for visualizing and interacting with ensembles of 3D isosurfaces through a combination of 3D rendering (isocontouring) and cut-planes (slices through 3D volumetric fields). In addition, we generalized the algorithm in [115] to three dimensions as a direct extension of their analysis of isocontours to isosurfaces – that is, from co-dimension one objects embedded in 2D to co-dimension one objects embedded in 3D. This generalization allows us to compare contour boxplot summaries of an ensemble to both full enumeration of the ensemble as well as other traditional means of atlas evaluation (e.g., qualitative visual inspection of slices of the atlas image or individual volumetric images used for construction of the atlas). We employ this system to explore, in collaboration with domain experts, the efficacy of using ensemble visualization techniques for evaluating 3D shape alignment of brain MRI images.

The purpose of this chapter is to study and evaluate the use of contour boxplots in a real-world data science application, the alignment of 3D shapes or surfaces in a population-based ensemble. Our hypothesis is that the contour boxplot will allow users to summarize their data in a meaningful way that allows either better or more efficient (faster) assessment of the atlas construction as compared to explicit enumeration of the ensemble (i.e., looking at each member image individually) or through more coarse-grained characterizations such as examination of the average intensity image or label (segmentation) probability

maps. As our evaluation results will show, the contour boxplot methodology has the potential to significantly benefit the application under study by providing a visualization of the quantitative summaries of the ensemble. Although we have formulated our hypothesis in the context of a particular application, we believe that our evaluation may provide insight into other arenas where visualization and analysis of ensembles of shape are desired. Examples of such applications will be discussed in the conclusion section.

To begin, we give a brief introduction to the process of brain atlas construction and the evaluation process used by domain experts.

3.1.1 Brain Atlas Construction

Construction of an anatomical *atlas* for a collection of brain images is an important problem in medical image analysis. The goal of various atlas construction schemes is to construct a statistical *representative image* and associated set of *coordinate transformations* (i.e., deformations) from an ensemble of images [50]. Anatomical atlases provide a common coordinate system (atlas space) in which to define reference locations of brain structures. As part of the atlas construction process, nonlinear registration techniques generate deformations that can map the anatomies in an individual image to the atlas space (see Figure 3.1). The atlas construction process jointly estimates a representative image defining the atlas space (the *atlas image*) and the deformations aligning individual images to this atlas image (i.e., mapping the image individually to the atlas space). The atlas image generated by these techniques then represents the *average* (or normal) anatomy of this population. Such atlases help domain experts characterize the expected anatomical structure and variability of a population and compare different populations in terms of their group atlases (for example, healthy and unhealthy groups). Differences in the atlas anatomy can be identified both *qualitatively* by inspecting unaligned structures (when mapped to the atlas space) and *quantitatively* by analyzing the deformations, quantifying the amount of change necessary to bring individual ensemble members into alignment.

Atlas generation is an automated process, but it is not parameter free, and the choice of parameters can greatly influence the quality of the result. In particular, nonlinear deformations computed for medical image registration are a tradeoff between image matching and *plausible* deformations. For example, the deformation should not result in the elimination

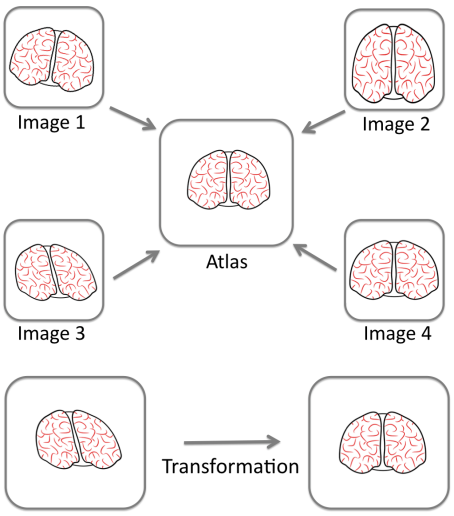


Figure 3.1. An atlas construction scheme involves deformation and registration of all ensemble members to the atlas. The process of deformation and registration of ensemble members is called transformation to the atlas coordinate system or the atlas space.

of anatomical features or noninvertible transformations. Hence, the deformation is often controlled by tuning parameters to find a compromise between the mismatch between images and the regularity (e.g., smoothness) of the transformation. Due to the regularization of the deformations and the inherent anatomical differences between ensemble members, not all features will be perfectly aligned. This imperfect alignment is manifested as *blurring* in the atlas image where there is disagreement regarding voxel intensity among ensemble members when mapped to the atlas space.

Correct tuning of the regularization parameters allows the deformations to account for as much anatomical variability as possible by correctly aligning the corresponding anatomy, and not simply matching similar intensities. This alignment of corresponding anatomy is essential for an atlas to be effective in later statistical analysis of the population. Convergence of the optimization can be easily checked, but the degree of alignment of particular structures is analyzed qualitatively by observing the amount of *blurring* in the atlas image and by checking the alignment of each ensemble member (deformed to atlas space) to the atlas image. The initial alignment is often unsatisfactory, which results in an iterative process of parameter tuning and rerunning the atlas generation process.

In addition, due to problems with image scans, extreme variability among the ensemble

members, or incorrect preprocessing, it may not be possible to achieve reasonable alignment to the atlas image for some set of *outlier* images. Identification and removal of such images is often another part of the atlas generation procedure. Automated measures of global image alignment are available, but they do not give insight into why or in which spatial regions particular ensemble members have poor alignment. Depending on the proposed application of the atlas, these insights may be pertinent to the decision to prune or keep particular images (ensemble members).

This manual iteration of parameter tuning/pruning and atlas generation eventually yields the final atlas to be used in further analysis. There are two important points to be noted about the final atlas image. The first is this *representative* image/segmentation is not a member of the ensemble itself, but rather an image/segmentation generated through statistical operations on the deformation fields. That is to say, it is not a member of the population that best represents the population, but rather an attempt at statistically characterizing a representative image. Second, as noted above, the iterative process does not guarantee that the resulting atlas image is crisp – that is, that there are no blurry regions in the image. The ensemble of images compared to the atlas image scenario is similar in spirit to the feature-space averaging issue highlighted in [115]; the analogy is that the isosurface (e.g., segmentation) of the average field is oftentimes not equivalent to a representative of a set chosen from isosurfaces of the individual fields. As per the rationale given in [115], the avoidance of feature-space averaging is why we believe the contour boxplot methodology provides a useful way to summarize the type of ensemble data where analyzing feature sets and their representatives is important. Since the manual, qualitative evaluation of shape alignment (as a result of image registration) is a challenging task, quantifying the variability of the shape alignment and visualizing this variability can facilitate the domain experts' ability to effectively validate the atlas construction scheme.

In Section 3.2, we introduce a prototype system that uses various uncertainty visualization schemes to enhance the study of variability in an ensemble of shapes. Before introducing our prototype system, we first provide an overview of the data used as well as a high-level description of our expert evaluation study.

3.1.2 Data Preprocessing for Atlases

The images analyzed in this chapter are 3D MRI images obtained from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database [49]. Each brain image in our ensemble was also provided with a corresponding label map volume with various anatomical structures segmented and marked, with each brain region having a unique integer value. In order to analyze a specific structure within the brain anatomy, we used the label assigned to that structure to select it and mask out the remaining region in all members of the ensemble. The atlas construction scheme we used is the unbiased diffeomorphic atlas proposed by Joshi et al. [50], implemented as part of an open-source medical image atlas construction package called AtlasWerks [97]. We constructed atlases from ensembles of MRI images using different choices of parameters and/or different ensembles (i.e., subject groups). In each case, after constructing the atlas using the MRI images, the corresponding label map images were transformed to the common (atlas) coordinate space using deformation fields calculated during the atlas construction process as described in Section 3.1.1. These transformed label maps were then passed as input to the preprocessing pipeline (described in Section 3.2) for visualization. For a well-constructed atlas, we can expect the anatomical structures in the brain to have a relatively small amount of variability after being transformed to the atlas space. We selected two anatomical structures in the brain expected to pose different levels of difficulty during atlas construction, namely the left ventricle and the cortex. The ventricle is often considered as a very distinct structure (i.e., high contrast) in the brain image and, therefore, can be expected to exhibit good alignment among ensemble members in the atlas space (if all goes well). The cortex was selected as an example of an anatomical structure with a complex shape (see Figure 3.2), a significant challenge for registration/alignment.

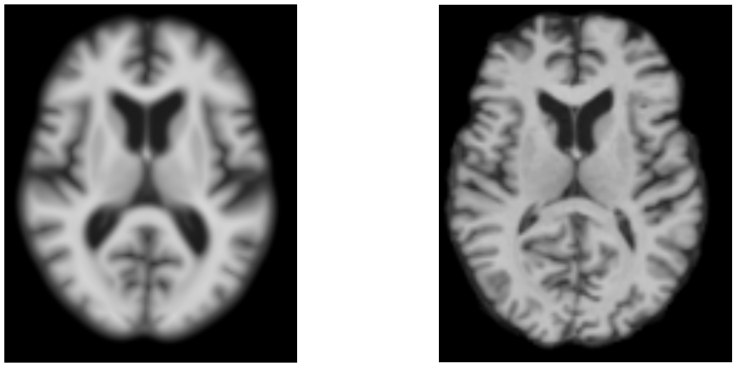
3.1.3 Expert Evaluation Study Details

Domain experts use various open-source or commercial packages to visualize slices from *individual* volumetric images or simply from the average of the aligned images, but to the best of our knowledge, ours is the first attempt to study the alignment of shapes in atlas construction using ensemble visualization techniques. For our evaluation study, we had access to a group of five domain experts who work with atlases on a regular basis and who

Figure 3.2. Illustration of the cortex (green) and the ventricle (red). This image shows the segmentation provided by the label map volume for a typical ensemble member. The coarseness of the segmentation seen in this label map is mitigated by smoothing for the final visualization.

volunteered to participate in our expert evaluation study. This group included graduate students, staff researchers, and faculty who use atlases and medical image ensembles in their research projects.

We asked the participants to explain their current methodology for evaluating the atlas construction scheme as well as the quality of the atlas in terms of being a representative of the ensemble. As mentioned earlier, we learned that this process is often performed *qualitatively*. A visual inspection is carried out to ascertain whether the shapes of the anatomical structures in the atlas space are realistic. Experts also mentioned that in order for an atlas to be helpful for different medical imaging applications such as segmentation of a specific structure in the brain, they need the atlas image and the anatomical structures therein to have sufficient contrast. For example, they expect to see a crisp boundary (in terms of the average combined image intensities) between gray and white matter in the brain. Therefore, the sharpness of the boundaries of the anatomical structures in the atlas image is another criterion examined qualitatively to evaluate the alignment of the ensemble. These qualitative evaluations are often performed on a subset of the ensemble of images (in the atlas coordinate system), because visualizing the entire ensemble results in too much clutter and blurriness. Figure 3.3 shows a snapshot of a slice of the brain atlas image used as a common (atlas) coordinate system to register individual label maps from the ensemble.



(a) Atlas image slice (b) An MRI image slice

Figure 3.3. Illustration of the atlas image slice constructed using AtlasWerks [97]. The anatomical structures in the atlas image usually have lower contrast and fuzzier edges as compared to an original MRI image. This fuzziness results from performing averaging while constructing the atlas.

3.2 Our Visualization Pipeline

In this section, we discuss the visualization pipeline of our prototype system. We first start with a brief summary of various ensemble visualization strategies that we have considered and incorporated into our prototype system. We then provide an overview of the pipeline and our design choices to mitigate the challenge of visualizing and rendering an ensemble of 3D isosurfaces.

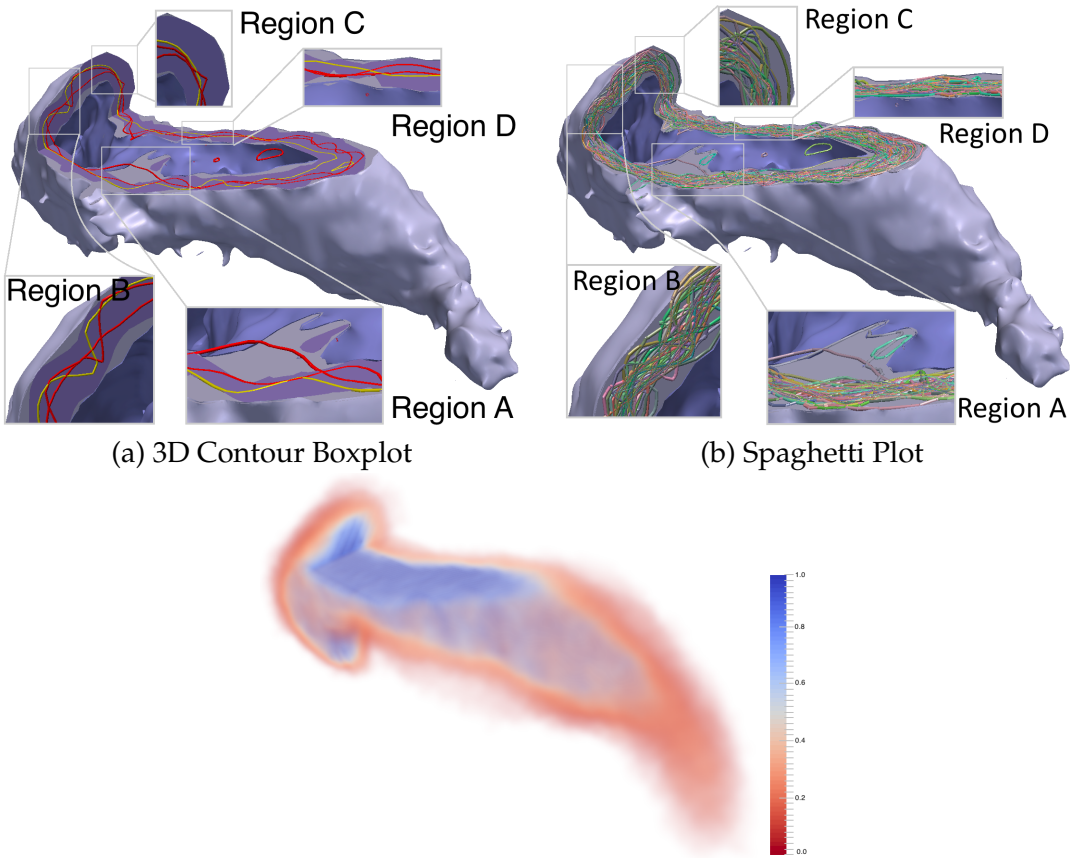
3.2.1 Ensemble Visualization Overview

Visualization is often data-driven, and therefore uncertainty visualization schemes are typically designed to deal with the type of data being visualized. For scientific data, users are often interested in visualizing derived *features* of their data, such as transition regions (or edges), critical points, or isosurfaces (of volumetric data) and the uncertainty associated with such feature sets. A thorough review of the rich literature on uncertainty visualization is beyond the scope of the current chapter. However, interested readers can consult [18, 85] for further details on recent advancements on this topic. The focus here is visualization of isosurfaces in the context of uncertain scalar fields, which has been studied somewhat extensively. Most relevant to the application under study (i.e., atlas construction) is the visualization of uncertain isosurface extracted from an *ensemble* of scalar fields.

Here we provide a brief summary of three classes of popular techniques for visualization of uncertain isosurfaces that are extracted from ensembles of scalar fields. These techniques were chosen to represent the range of strategies for representing an ensemble (as discussed in Chapter 1), namely, 1) enumeration of all ensemble members; 2) visualization of the statistical summaries induced from parametric uncertainty modeling; and 3) descriptive nonparametric summaries:

1. *Enumeration.* A widely used approach for ensemble visualization is the direct visualization of all ensemble members. Direct visualization of ensembles has gained significant interest in applications such as weather forecasting and hurricane prediction [23]. *Ensemble-vis* [86] is an example of the data analysis tools designed to visualize ensemble data. Ensemble-vis uses multiple views of fields of interest to enhance the visual analysis of ensembles. We incorporate direct visualization of 3D ensemble members (see second column in Figure 3.4) by rendering the curves formed by the region of intersection of the co-dimension one isosurface of each ensemble member with a cut plane. Note that as long as the isosurface embedded in 3D is closed, closed curves will be generated when the isosurface is sliced for visualization purposes. We refer to this visualization as a *spaghetti plot*. In order to facilitate the interpretation of the individual ensemble members, each of these curves has been rendered with distinct and random colors. There are a variety of options for rendering the enumeration of all 3D surfaces, including transparency, but clutter is a significant challenge [23]. For this work, we present the surfaces of the inner- and outermost volumetric bands formed by all ensemble members. User studies have suggested the effectiveness of direct ensemble visualization techniques [23]. However, direct visualization of the ensemble does not provide any quantitative information about the data uncertainty, and relies solely on the user for interpreting data.

2. *Parametric probabilistic summaries.* Many uncertainty visualization schemes use probabilistic modeling to convey *quantitative* information regarding data uncertainty. Such techniques often rely on a certain kind of statistical model such as multivariate normal distributions. As a representative of such techniques, we have chosen to consider



(a) 3D Contour Boxplot

(b) Spaghetti Plot

(c) 3D Average Intensity Image

Figure 3.4. Three visualizations of ventricles from an ensemble containing 34 images from the ADNI data set transformed to a common atlas space. Left: the contour boxplot visualization in 3D, with 50% volumetric band dark purple, 100% band volume in light purple, median in yellow, and outliers in red (on the cutting plane). Middle: direct visualization of the ensemble members (spaghetti plot). Right: 3D average intensity image.

the concept of *level-crossing probabilities* (LCP) [82]. For visualization, we implemented the 3D *probabilistic marching cubes* algorithms (proposed based on LCP) [84] as part of our initial visualization system. Probabilistic marching cubes rely on approximating and visualizing the probability map of the presence of the isosurface at each voxel location. However, the use of parametric modeling can limit the capability of this techniques. Approximating the underlying distribution giving rise to the ensemble and presenting the user with *only* aggregated quantities of the inferred distribution can be misleading in some applications. For instance, this approach can often hide or distort structures that are readily apparent in the ensemble.

3. *Nonparametric descriptive summaries.* An alternative strategy that relies on neither enumeration nor parametric modeling of the underlying distribution is to form *descriptive statistics* of an ensemble. Descriptive statistics offer an ensemble visualization paradigm for understanding or interpreting uncertainty from the structure of an ensemble. The notion of *centrality* is a natural approach to understanding the structure of an ensemble. Because an ensemble is an empirical description of its distribution, some instances from an ensemble are more central to the distribution, and therefore more typical within the distribution. The notion of *data depth* provides a formalism for characterizing how *central* a sample is within an ensemble. Data depth provides a natural generalization of rank statistics to multivariate data [66]. The univariate boxplot (or whisker plot) is a conventional approach to visualize order statistics. Boxplot visualizations provide a visual representation of the main features of an ensemble, such as the most representative member (i.e., the median), quartile intervals, and potential outliers. The notation of data depth has been generalized for ensembles of isocontours [115]. In [115], the authors propose *contour boxplot* as a visualization technique to summarize robust and descriptive statistics of ensembles of 2D isocontours [115]. In our system, we algorithmically extend and implement the *contour boxplot* analysis for isosurfaces embedded in 3D (see Figure 3.4, first column) as an example of visualization techniques based on nonparametric descriptive statistical summaries of an ensemble.

In order to analyze the alignment, or lack thereof, of shapes in an ensemble, we incorporated representative members of the aforementioned ensemble visualization technique categories as part of our prototype system.

3.2.2 Ensemble Visualization Prototype System

At a high level, our prototype system consists of two stages (see Figure 3.5):

1. *Data Preprocessing.* When visualizing isosurfaces of a binary 3D segmented image, it is often necessary to perform smoothing to reduce aliasing artifacts and facilitate 3D rendering/shading. We perform this smoothing in a two-step preprocessing stage. In the first step, the binary partitioned image is antialiased using an iterative relaxation process described in [114]. Next, a very small amount of mesh smoothing

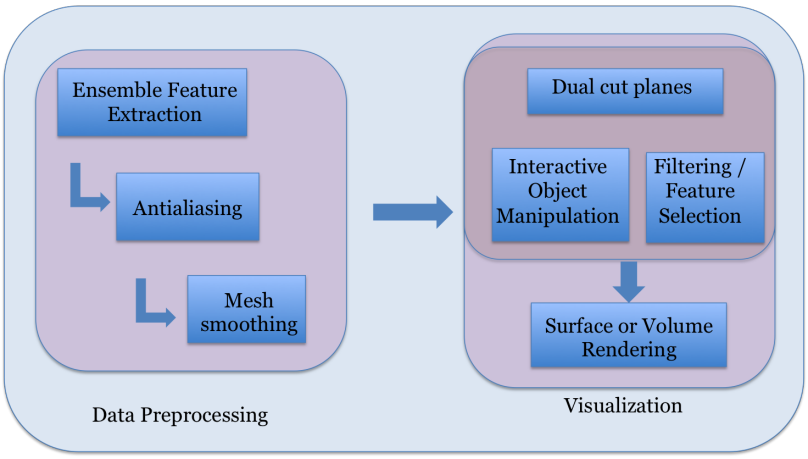


Figure 3.5. Overview of prototype system designed for shape alignment evaluation using ensemble visualization.

is performed on the isosurface mesh generated from the antialiased binary image. All visualization preprocessing operations occur on the 3D volume (and corresponding co-dimension one isosurfaces) prior to cut-plane extraction.

2. *Visualization.* This stage includes some visualization strategies to facilitate the perception and navigation of the rendered 3D objects. In order to improve the perception of shape in our application, we include interactivity with renderings of 3D objects as part of the visualization system. In our settings, the user is able to rotate the object displayed on the screen using the standard trackball interaction mechanism. The system allows the user to select cutting planes, which clip a portion of the volume displayed on the screen, to render cross-section views of surfaces embedded in 3D. The user can also interactively orient and translate the cutting plane. Additionally, the system provides the flexibility of having one or multiple cutting planes and interactively adjusting their position and orientation. The interface of the system allows the user to interactively select various features of interest for rendering in order to focus on any particular feature of interest. For example, the user can select specific ensemble members to be rendered individually.

In the case of 3D contour boxplots, the analysis is performed on the 3D binary segmented volumetric data (in the preprocessing stage), and the results are rendered interactively. While the analysis is performed on the volumetric data leading to volumetric 50-

and 100% bands, we render the visualization of the statistical summaries only on chosen cut planes to deal with the issue of occlusion. For instance, in the absence of a cut plane, the 100% band entirely occludes the median shape as well as the 50% band.

3.3 Evaluation

In this section, we demonstrate the efficacy of using ensemble visualization techniques to study the alignment of MRI brain images during brain atlas construction by gathering feedback as part of an expert evaluation study of the proposed prototype system. We refer to our expert evaluators as *participants*. All the visualizations presented were part of the prototype system introduced in Section 3.2. We described the prototype system to the participants after a walk-through presentation of the different ensemble visualization techniques. The participants were able to interact with the system and switch through the various visualization methods as explained in Section 3.2. For our study, we solicited their feedback on the visualization of the two anatomical structure presented below: the left ventricle and the cortical surface. We paid particular attention to the participants' comments concerning the suitability of ensemble visualization for this application. A summary of our interactions with the participants follows. We start by describing three examples where useful insights into the atlas data were gained by the participants when interacting with the system.

In our first example, we focus on analyzing the *variability within an ensemble* of different regions of brain ventricles transformed to a common atlas space using the unbiased, diffeomorphic approach in [50]. Ensemble visualization not only helps general users identify regions that are either well or poorly aligned, but also provides insight regarding whether the variability is due to differences in shape, position, or both.

Figure 3.4 shows the three approaches to visualizing the aligned ventricles for an ensemble of 34 brains. From the contour boxplot in Figure 3.4a, one can immediately identify regions of high variability such as Region A, which is highlighted in the figure. In this specific region, most of the variability is outside the 50% band, which means that less than half the ensemble members contributed to this variability. Looking at the spaghetti plot in Figure 3.4b, we see there are, in fact, only two ensemble members that significantly differ from the other members in Region A. These results show that the variability is due

to overall position as well as shape in this region. In Region B (Figure 3.4a), we notice that the variability can be attributed to significantly different shapes of the isocontours, and that these shapes would not easily be aligned through the smooth transformations in this atlas, and may require parameter tuning to achieve alignment. By observing Region C (Figures 3.4a–b), we see that the variability comes mostly from the positions of the isocontours. Results in Region C also show that no particular ensemble member is disproportionately responsible for the variability—the width of the 50% band is nearly that of the 100% band in this region, and outliers align well with the median contour. Finally, Region D (Figure 3.4a) demonstrates an area of very low variability across the ensemble and provides an example of good alignment of all the ventricles, which is confirmed by the spaghetti plot in Figure 3.4b. Figure 3.4c shows a volume-rendered 3D version of the average intensity image for comparison. The average intensity image is an essential part of the atlas, but it does not provide the same insights for *debugging* the atlas in a detailed way.

We also showed the participants' volume renderings of level-crossing probability values, as suggested in [84]. The participants noted that the level-crossing probability visualization shows almost the same information as the average intensity image (Figure 3.4c), which is already used extensively during atlas construction. They did not feel that further exploration of this form of ensemble uncertainty visualization for evaluating atlases would be useful, and therefore we did not include comprehensive results from level-crossing probability renderings in this study.

The second example was chosen to evaluate whether ensemble visualization can also provide insight into the *overall variability* between the members of an ensemble of aligned shapes. An understanding of the overall variability (as opposed to local variability) is useful not only to understand how well a particular atlas was constructed, but also to compare different atlases. For this example, we have constructed three atlases, each with an ensemble of size 30. The first atlas was constructed with a high value of regularization (transformation smoothing), $\lambda = 1.0$; a second atlas was constructed for the same ensemble while using a low regularization value, $\lambda = \frac{1}{9}$; and a third atlas was constructed from a different ensemble (i.e., subject group) with the regularization/smoothing at $\lambda = \frac{1}{9}$.

Figure 3.6 shows slices of intensity atlases and contour boxplot visualizations for each

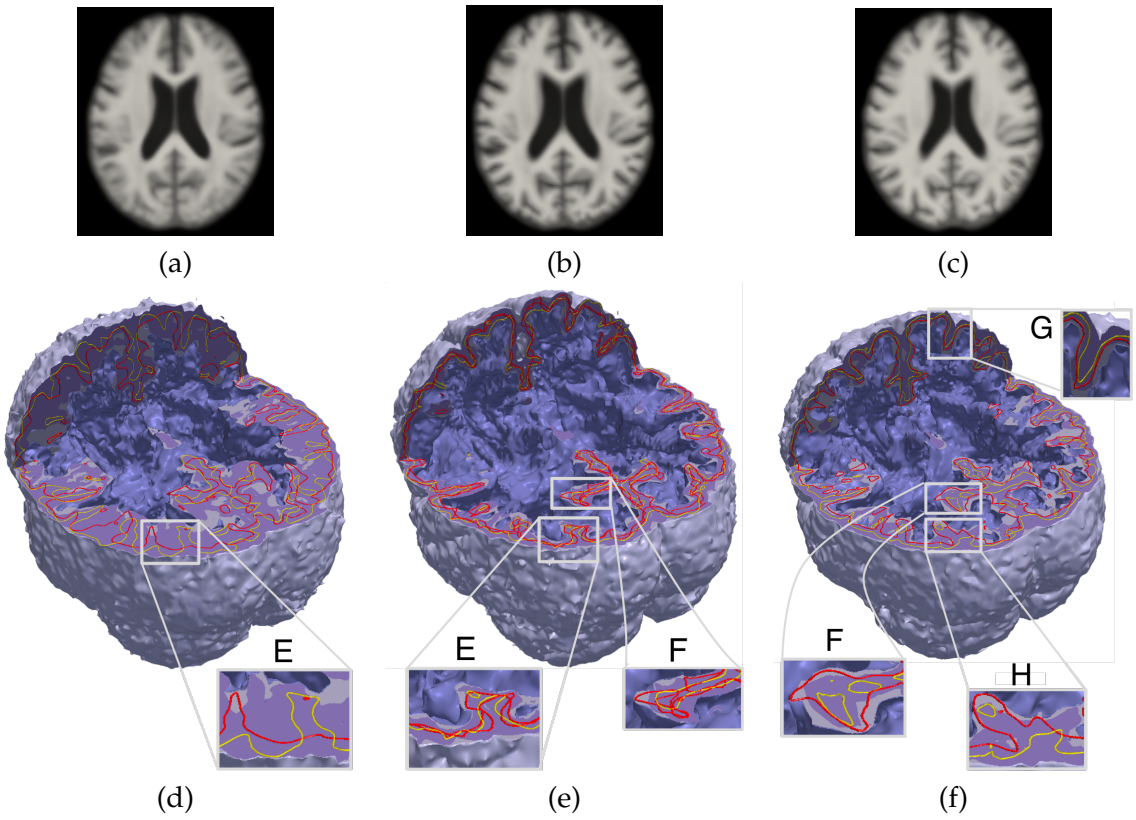


Figure 3.6. Brain atlases with different parameters and subject groups. Top: slices of average intensity atlases for ensembles of 30 brain images. Bottom: associated contour boxplot visualizations for cortical surfaces. Left: atlas constructed with high regularization of deformation. Middle: atlas constructed with low regularization. Right: atlas with low regularization using a different ensemble than in the other columns.

of the three cases (columns from left to right). The first row presents a slice of the intensity image for each atlas, and the second row demonstrates the 3D contour boxplot visualization of the cortical surfaces for atlases corresponding to the intensity image above.

Using a high value for the regularization parameter enforces high smoothness of the deformation fields, which in turn makes it harder to arrive at a set of deformations that would perfectly align all the individual images. The lack of alignment leads to high variability between isosurfaces in the ensemble. Such high variability is easily visible by looking at Region E in Figure 3.6d where the 50 and 100% bands are wider than in the corresponding region of the atlas with low regularization (Figure 3.6, middle column). Better image alignment when the atlas is constructed with low regularization is also evident in Region E by comparing contours of the median and outlier shapes rendered on the cut

plane in Figures 3.6c and d. We see that the median and the outlier shapes are poorly aligned for images aligned with an atlas constructed with high regularization (Figures 3.6, first column), but the alignment is much better when the atlas is constructed with low regularization.

Finally, the third atlas (right column of Figure 3.6) in this example demonstrates the effect of inherent variability between the ensemble members (i.e., brain images) on the atlas construction process. We see that in many regions of Figure 3.6f, for instance in Region F, the 100% band is significantly wider than the 50% band, indicating a significant spread in the distribution of surfaces, which is different from the variability seen in the corresponding region in Figure 3.6e, where both bands nearly overlap. Furthermore, in the third atlas we see that the outlier is well aligned with the median in some regions (see Region G), but poorly aligned in others (see Region H). This example demonstrates that shape/surface variability in atlases depends on, in addition to parameters of construction, the inherent variability of shapes in the ensemble. Thus, the contour boxplot, as part of the atlas construction process, can help users tease apart these different aspects of variability.

In addition to aiding in the understanding of the general alignment of shapes in an ensemble, the contour boxplot is also useful in conveying to the general user how well a particular shape is aligned with respect to the rest of the ensemble. Such knowledge is particularly useful in the case of *outlier shapes*. Atlas construction is often an iterative process, and identification of outlier images that do not align sufficiently with the atlas is an important intermediate step in the process. In the contour boxplot shown in Figure 3.7c, we see a single outlier shape and its alignment relative to the ensemble. In comparing this visualization with an average intensity image of the left ventricle region Figure 3.7a, we see that an anomaly in Region I (Figure3.7c) shows as a barely perceivable increase in intensity in Figure 3.7a. A similar observation can be made from the intensity image slice of the outlier member shown in Figure 3.7b. However, the anomaly shows up clearly in the contour boxplot, and because it is outside the 100% band, we know that the degree of misalignment of this shape is rare within the ensemble of ventricles. Region I also demonstrates the challenges of assessing geometry in 3D, because distances between surfaces can be exaggerated when viewing them on a single cut. However, interacting with the visualization by moving and rotating the cut plane can help verify the 3D shapes of rank

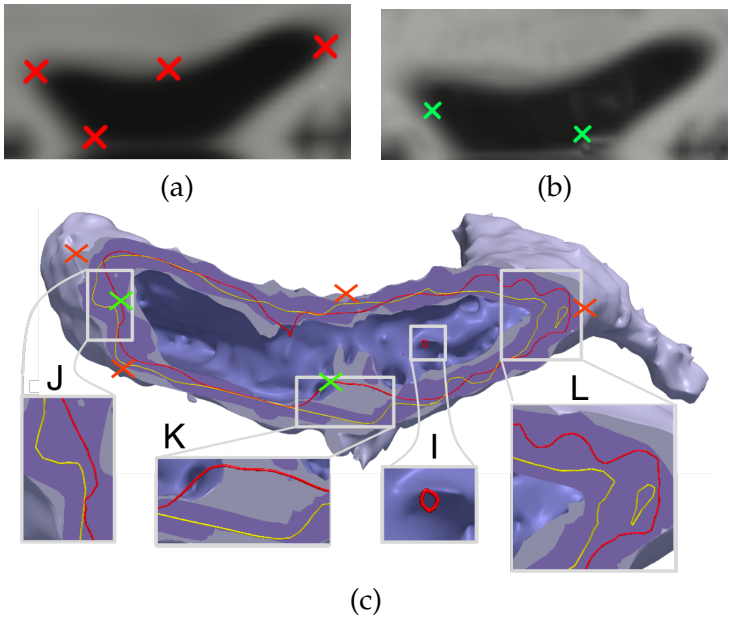


Figure 3.7. Visualizations of left ventricles. Crosses mark the correspondence between the images. (a) Left ventricle slice from an intensity image of the atlas. (b) Left ventricle slice of an ensemble member identified as an outlier by data depth analysis. (c) Contour boxplot visualization of an ensemble of 34 ventricles in atlas space.

statistics and the surface geometries and separation distances.

In some cases, aligned shapes can differ in *size* from the rest of the ensemble. For instance, Figure 3.7c shows that the outlier ventricle is noticeably smaller than the median ventricle in Regions J and K, which is not the case in the Region L. This observation is not possible in the corresponding intensity images. These size differences occur for several reasons. In this example, for instance, the outlier ventricle may have been different and irregular to begin with. Another reason could be mislabeling of the ventricular region during the segmentation process to generate the labels for that image. Finally, the process of generating deformations during the atlas construction might fail, leading to irregularities for an ensemble member when mapped onto the atlas space. The contour boxplot can provide information that can help the user decide whether or not any particular outliers need to be removed from the ensemble or if further investigation is necessary to identify causes of possible misalignment.

At the conclusion of our study, we asked the participants to comment about their experience with the system, including the applicability of such a system if integrated

into an atlas construction software. They were also asked to compare the ensemble visualizations to the evaluation techniques they currently use. The two main techniques currently used for atlas evaluation are inspecting unaligned structures (when mapped to atlas space) or analyzing the deformations, quantifying the amount of change necessary to bring individual ensemble members into alignment. Here we summarize the observations of the participants in this study:

- The participants pointed out that being able to visualize the *extent* of the variation among the ensemble of aligned shapes in terms of quantitative percentile information using the contour boxplot visualization was helpful for comparing various atlas construction schemes (or comparing atlases that were constructed from different ensembles or parameter settings). They also mentioned that the contour boxplot has the potential to help reduce the time needed for the user of the atlas construction software to gain insights regarding the quality of the atlas.

- The participants noted that state-of-the-art techniques for evaluation/visualization of atlases provided limited information about the variability that remained within an ensemble after transforming it to atlas space. Deformation and image match energies (quantities that are optimized during registration of images in atlas construction) are not able to provide insight into the geometric discrepancies that are crucial to understanding atlas quality.

- The participants noted that the capability of the contour boxplot to effectively locate and characterize different types of variability was valuable in atlas construction.

- The participants pointed out that an automated and statistically robust way of identifying and visualizing outliers in an ensemble can play a major role in construction of an atlas.

- The spaghetti plot was found to be helpful to view the contours of specific ensemble members other than the median or outliers.

- The participants noted that both the contour boxplot and the spaghetti plot were able to convey important details pertaining to the variability in an ensemble, whereas the average intensities had limited utility because of their general fuzziness.

We conclude this section by summarizing our findings from this study and the interview process. The goal of the application described in this chapter is to evaluate the alignment of 3D shapes, in particular the alignment of 3D MRI images that have been transformed to a common atlas space, using various ensemble visualization methods. It is observed that the ensemble visualization methods are helpful in characterizing the alignment of shapes, and furthermore, provide insights that are useful in understanding the variability in alignment. An understanding of the type or location of the variability can be helpful in tuning parameters used in atlas construction and/or removal of outliers to achieve better alignment. We observed that the contour boxplot emerged as a clear favorite of our participants. One of the salient features of the contour boxplot that makes it distinct from the other ensemble visualization approaches is its ability to convey an aggregated result from the analysis of all regions of shapes in the ensemble on any arbitrary cut plane. For example, visualizing a slice of the intensity image, or contours on a cut plane using the spaghetti plot, conveys the variability for *only* the region intersecting the cut plane, whereas a contour boxplot visualization using the same cut plane also provides information about the median and outlier contours that are calculated from a global analysis of contours. The contour boxplot, however, has a drawback in that it does not give the user much information about specific ensemble members, other than the median or the outliers. For such cases, the spaghetti plot with interactivity that allows highlighting of specific ensemble members can augment the contour boxplot by providing more detail if the general user wishes to focus on very specific anatomical areas or members of the ensemble.

3.4 Conclusions

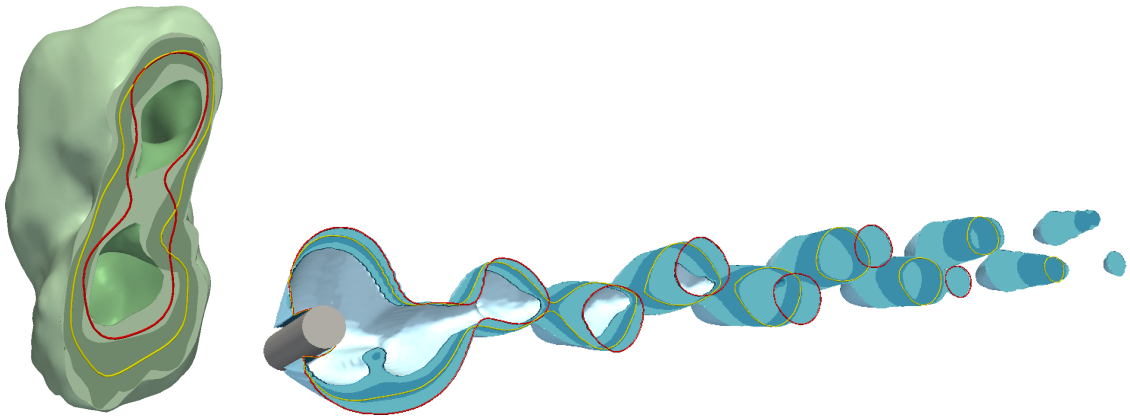
In this chapter, we introduce a new approach to study alignment of shapes. We demonstrate the efficacy of using the 3D contour boxplot ensemble visualization technique to analyze shape alignment and variability in atlas construction and analysis as a real-world application using a prototype system. The system was evaluated by medical imaging experts and researchers working with medical image atlases in an expert evaluation study that was conducted to examine the applicability of ensemble visualization for studying shape alignment and variability. We find that providing the user with both quantitative

and qualitative visualization of variability can yield better understanding of the main features of the ensemble and the atlas construction quality.

Future work for our system in the context of the current application includes refining the system in order to address the suggestions provided by the participants, such as viewing the specific structures in the context of the whole brain and more interaction options. Furthermore, ensemble visualization approaches discussed in this chapter can be integrated into an atlas construction package in order to provide the users the capability of interactively inspecting the shape alignments and the variability among ensemble members after atlas construction. Motivated by the feedback from the participants, a more comprehensive study is required to examine the applicability of ensemble visualization to *compare* different atlas construction schemes.

In addition, studying shape variability has applications in various branches of science. In molecular dynamics, researchers study different types of molecular structures and the shapes of their potential fields in solutions (which vary stochastically) in order to understand, for instance, their biochemical properties [123]. Scientists are also interested in the evolution of the shape of molecules. For example, the surfaces of 3D molecular chains are of significant interest for comparison of various types of protein structures [123]. In Figure 3.8a, the contour boxplot visualization of the surface of an ensemble of simulated HIV molecules is shown. The ensemble members underwent a Procrustes alignment (translation, rotation, scale) using the positions of the underlying molecules. The potential fields that form these contours are inherently smooth, and thus there was no need for preprocessing of these volume data.

Another application where the study of shape variability and alignment is of significant interest is fluid mechanics. In fluid mechanics, when developing models of vortex behavior, scientists oftentimes study the variability of the shape of vortex structures among different simulations (e.g., using slightly different parameter settings or boundary conditions) to confirm that their observations are repeatable [117], rather than a numerical artifact of a particular simulation. The center of an eddy corresponds to low pressure values in the flow, and hence studying the pressure field of a fluid flow can help detect the position of the eddies and regions of high vortices. We have used the 2D incompressible Navier-Stokes solver as part of the open-source package Nektar++ [1] to generate an ensemble of 28 fluid



(a) (b)

Figure 3.8. Contour boxplot visualizations for simulated ensemble data sets. (a) Contour boxplot visualization for an ensemble of size 100 simulated HIV protein. Here, we see the median contour in yellow and the outlier contours in red. (b) Contour boxplot visualization of the isosurface of the pressure field of a fluid flow. The pressure is considered as a function of depth to generate a 3D pressure volume. The median contour is drawn in yellow and the outlier contours are drawn in red.

flow simulation runs. These simulations have been designed for a steady fluid flowing past a cylindrical obstacle. For each of the ensemble members, we randomly perturbed the initial conditions such as inlet velocity and Reynolds number. For this example, the pressure dependence in the third dimension was computed analytically. The contour boxplot visualization of the isosurfaces of the pressure volume is shown in Figure 3.8b. Many possible applications beyond the ones showcased could benefit from the contour boxplot summary and visualization technique.

CHAPTER 4

PATH BOXPLOTS FOR CHARACTERIZING UNCERTAINTY IN PATH ENSEMBLES ON A GRAPH

Portions of this chapter have been reproduced with permission from Taylor and Francis Group and is based on material published in JCGS, Path Boxplots for Characterizing Uncertainty in Path Ensembles on a Graph, M. Raj, M. Mirzargar, R. Ricci, R.M. Kirby, R. Whitaker, vol. 9, 2017, pp. 243-252 [90].

4.1 Introduction

Making sense of sets of information defined over graphs can often be a challenging because graphs are typically used to represent abstract data that may not be easily representable in a flat, or Euclidean, space. Here, we define a graph $G(V, E, W)$ as a set of vertices (or nodes) V, a set of edges $E \subseteq V \times V$, and a set of edge weights, $W : E \mapsto \mathbb{R}^+$, assigned to each edge. In this chapter, we describe a method to gain insight into a particular type of data represented on graphs, namely collections or *ensembles* of paths on graphs, henceforth referred to as *path ensembles*. We define a path (a special type of subgraph) as a sequence of vertices $p = (v_i : 1 \leq i \leq m)$, where $v_i \in V$ and each consecutive pair of vertices in the sequence have an associated edge, $(v_i, v_{i+1}) \in E \; \forall \; i = \{1, \ldots, m-1\}$. We define a path ensemble as a collection of paths on a particular graph.

Paths on a graph are natural structures used to describe and analyze data in a range of applications. For instance, in transportation urban planners study ensembles of paths of commuters (e.g., from recorded GPS data) to identify important travel corridors to plan new routes [34]. Analysis is performed on a graph whose vertices are usually transition points (road intersections, airports). These vertices have a geographical location and an abstract, logical meaning. The edges in the graph represent direct transportation connections between vertices (segments of roads, routes of airplanes), and they often encode, as

weights, information about transit time or cost. A path on this graph is an abstraction of a commuter's path.

In computer networks, system administrators try to detect anomalies or attacks by keeping track of the paths taken by the network traffic over a period of time [19]. Analysis is performed on a graph whose vertices are Internet subdomains known as *autonomous systems* (ASes), and edges represent a direct data link between ASes, which can encode, as weights, transfer capacity. A path on this *AS graph* represents the path of a packet on the Internet.

In molecular dynamics where scientists are interested in studying the protein folding process, various possible configurations (also known as *states*) of a specific protein structure are known but the sequence of discrete intermediate states in the process of protein folding is not. Analysis is performed on a *configuration graph* whose vertices represent the possible protein configurations, and weights on edges denote the respective transition probabilities between the associated pair of configurations. In this case, a path is a sequence of potential discrete intermediate states and may be identified by carrying out simulations that incorporate stochastic transitions. These simulations result in an ensemble of possible paths for a folding process on the graph associated with a molecule [8]. In path analysis [119], graphs are used to model dependencies (encoded as edges) among a set of variables (encoded as vertices). Direct and indirect dependencies between variables can be represented as edges and paths, respectively, in a model (graph).

Recently, researchers have begun considering the problem of systematically analyzing and visualizing path ensembles. One of the first challenges is how to summarize or aggregate the information in path ensembles. One approach of aggregation relies on specialized heuristics that often incorporate statistics of low-dimensional descriptors of paths. In road networks, the average travel time between two nodes becomes a salient feature [46]. In the analysis of computer networks, one might quantify the amount of traffic passing through a node in a computer network [19]. In molecular dynamics, the product of transition probabilities along folding paths is considered [8].

Another aggregation approach proposed by researchers is to compare paths directly, rather than using low-dimensional descriptors. Aggregate operations on path ensembles often rely on a definition of the distance between two paths such as Hausdorff [113] or

Fréchet [32] metrics, which are, in turn, based on distances between individual vertices. From these distances one can generalize the classical notions of statistical summaries such as median and mean [2, 33], as well as clustering [54]. Such aggregate characteristics for a path ensemble can help in understanding the structure of the ensemble.

To fully understand the structure of path ensembles by evaluating relationships between paths, applications need to consider not only distances between vertices in a path, but also patterns or differences in the (global) structure or *shapes* of paths. For instance, some paths may deviate from a central or most representative path, but in either typical or atypical ways. State-of-the-art aggregation techniques for path ensembles typically ignore the relationships that may exist between patterns of vertices in a path.

A growing body of research in analysis methods based on the notion of *data depth* robustly accounts for nonlocal relationships (correlation) among variables in multidimensional data, in essence capturing their global structure faithfully. Data depth is a method from descriptive statistics that provides a way to quantify *centrality* of multivariate points in an ensemble and derive a center-outward ordering, with few assumptions about the underlying distribution. Data depth has been shown to generalize to multidimensional data, and data-depth formulations, which account for relationships among variables, have been developed for specialized data types such as functions [66, 103], isocontours [115], and curves [67, 73]. Motivated by formulations of data depth for ensembles of multidimensional data, we propose a generalization of data depth for path ensembles on graphs, which we call *path band depth*. At a high level, our generalization comprises the following two parts, which it shares with earlier formulations for functions and curves: 1) a definition of *band* formed by a set of ensemble members; and 2) a definition of path band depth. We also propose a visualization strategy for path ensembles, which we call *path boxplots*, based on the order statistics induced by the depth assigned to the paths.

This chapter is organized as follows. In Section 2, we briefly discuss distance metrics that are currently used to analyze path ensembles, followed by the notion of data depth and *band depth*, a type of data depth, and its existing formulations to specialized data types such as functions and curves. In Section 3, we develop our generalization of band depth for paths. In Section 4, we develop our proposed path boxplot visualization strategy. In Section 5, we compare our generalization to distance-metric-based alternatives using

synthetic data and present two real applications: transportation and computer networks.

4.2 Background and Related Work

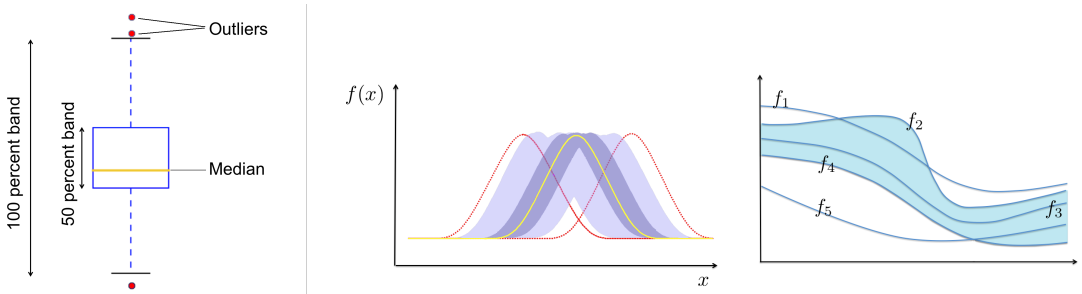
We begin with a brief discussion of current methods for analysis of path ensembles. In order to select a representative path, Evans et al. [33, 34] have proposed a generalization of Hausdorff distance for sets of vertices on graphs, which they call network Hausdorff distance (NHD). The classical Hausdorff distance is a measure of dissimilarity between sets and is defined as the maximum of distances from a set of points to their respective nearest neighbor in another set. For paths, we let p_a and p_b denote the sets of vertices for two paths within a weighted graph, and then the network Hausdorff distance is defined as [34]

$$d_{\mathrm{H}}(p_a, p_b) = \max_{v_a \in p_a} \min_{v_b \in p_b} d_g(v_a, v_b), \tag{4.1}$$

where $d_g(v_a, v_b)$ is the geodesic (or shortest path) distance between vertices v_a and v_b. The path minimizing the sum of distances from all other paths in an ensemble is the most representative path, a natural generalization of the median.

Alternatively, Eiter and Mannila [32] use the *discrete Fréchet distance* (DFD) between paths, as an approximation of the classical Fréchet distance. It relies on the set monotonic orderings of the vertices (correspondences or parameterization between paths). The length associated with a correspondence between two paths is defined as the maximum geodesic distance between corresponding vertices, and the DFD distance is defined as the minimum length over all possible correspondences. As with functions, point-based metrics of geometric distances, such as NHD and DFD, generally do not account for the overall, global structure of objects (paths in this case). Therefore, although such metric account for worst-case, vertex distances, they do not capture what is generally referred to as *shape differences* in the geometric setting.

This chapter introduces a method for exploratory analysis or *visualization* of path ensembles on graph, with consideration of their global structure. The proposed approach is motivated by the univariate boxplot (see Figure 4.1a) introduced by Tukey [108] as an exploratory data analysis tool, based on data depth to summarize the descriptive statistical summaries of an ensemble, based on rank statistics, such as median, first, and third quartile; nonoutlying minimum and maximum values; and identified outliers.



(a) (b) (c)

Figure 4.1. Band and boxplot for univariate points and functions. (a) A classic boxplot for univariate data. (b) A functional boxplot for an ensemble of functions. The median function is drawn in yellow, outlier functions in red. The 50% and the 100% data envelope are shown in dark and light purple, respectively. (c) An ensemble of five functions and a sample band formed by three member functions (f_2,f_3 and f_4) from the ensemble.

A widely adopted strategy for evaluating the depth of a data sample with respect to a data ensemble is *band depth*. Band depth is a formulation of data depth that relies on the probability that a data point lies *between* a random selection of other points from the distribution. For multivariate data, the *simplicial depth* of a n-dimensional point is the probability of a data point lying in the simplex formed by $n+1$ (distinct) randomly chosen points from the distribution [62]. Lopez-Pintado and Romo propose a concept of band depth for functions [66], in a way that goes beyond point-wise analysis of functions and provides an analysis that accounts for nonlocal correlations that span the function domain. Sun and Genton [103] use this data ordering to construct *functional boxplots*, a generalization of the conventional whisker plot for visualization of ensembles of functions (see Figure 4.1b). Several authors have proposed extensions of functional band depth to curves in n dimensions and associated boxplots [67, 73].

The proposed method generalizes the method of function/curve band depth for paths, and therefore we give a brief overview of methodology for band depth on functions/curves [66, 67, 73]. First, we consider an ensemble of n functions:

$$\mathcal{E} = \{f_1(t), f_2(t), \cdots, f_n(t)\} \subset \mathbb{F}, \quad f_i \in \mathbb{F}, \tag{4.2}$$

where $\mathbb{F} = \{f | f : \mathbb{R} \mapsto \mathbb{R}\}$ denotes the space of continuous functions on a compact interval. A function g falls within the band $B[\cdot]$ formed by a set of j functions if it lies within their min/max envelope (see Figure 4.1c). That is,

$$
\begin{aligned}
g \subset \mathrm{B}\left[\{f_{i_1}, \cdots, f_{i_j}\}\right] \quad & \text{iff} \\
& \min\left(f_{i_1}(t), \cdots, f_{i_j}(t)\right) \leq g(t) \leq \max\left(f_{i_1}(t), \cdots, f_{i_j}(t)\right) \quad \forall t.
\end{aligned}
\tag{4.3}
$$

Note that the *band* associated with a random set of functions is the min/max envelope, and the inclusion in the band forms a binary *test* that provides evidence of centrality—not to be confused with other statistical summaries, such as confidence interval or variance on functions.

The band depth of each ensemble member, g, is defined as the probability of its inclusion within the band formed by a random selection of j other functions from the ensemble:

$$
\mathrm{BD_j}\left(\mathrm{g}\right) = \mathrm{Prob}\left(\mathrm{g} \subset \mathrm{B}\left[\{\mathrm{f_{i_1}}, \cdots, \mathrm{f_{i_j}}\}\right]\right). \tag{4.4}
$$

For computation, the probability in (4.4) is expressed as the expectation of the characteristic function on $g \subset B[\{f_{i_1}, \cdots, f_{i_j}\}]$, and approximated by a sample mean using all choices of j samples from the ensemble (or a random subset, if the ensemble is large):

$$
\begin{aligned}
\mathrm{Prob}\left(g \subset B\left[\{f_{i_1}, \cdots, f_{i_j}\}\right]\right) &= E\left[\chi\left(g \subset B\left[\{f_{i_1}, \cdots, f_{i_j}\}\right]\right)\right] \\
&\approx \frac{1}{\binom{n}{j}} \sum_{\{f_{i_1}, \ldots f_{i_j}\} \subset \mathcal{E}} \chi\left(g \subset B\left[\{f_{i_1}, \cdots, f_{i_j})\}\right]\right),
\end{aligned}
\tag{4.5}
$$

where $\chi(\cdot)$ denotes the characteristic function.

Several practical issues are worth noting. The choice of the number of samples j used to form the band is not specified by the formulation, and may depend on the nature of the data (e.g., variability, number of samples). For larger ensembles, the total number of j-sized subsets may be too large, in which case random subsets may be chosen. Alternatively, the number of j-sized subsets of $\mathcal{E}$ may not be large enough to produce reliable probability estimates and properly order the samples. To address this issue, [66] proposed *modified functional band depth*, which replaces the characteristic function χ in (4.5) with the measure over the domain of $f \in \mathbb{F}$ for which the pointwise inclusion within the band holds. This relaxation can undermine the shape discrimination properties of the depth formulation. Alternatively, Whitaker et al. [115] propose an ϵ-modified band depth (for sets and contours) that relaxes χ to allow a certain amount (e.g., percentage) of the domain to fall outside of the band.

4.3 Band Depth for Paths on Graphs

In this chapter we propose a formulation of band depth for vertices of a graph, and extend that formulation to band depth for paths on graphs. The strategy for building a band for paths mirrors the development of the band depth for curves (i.e., functions $c : \mathbb{R} \mapsto \mathbb{R}^n$) [67, 73], which is to establish a definition of a band for points in the range of the function in $\mathbb{R}^n$ and then apply that band definition for all points in the domain.

In $\mathbb{R}^n$, the band formed by a set of j points has been formulated as the convex hull of $\mathcal{X} = \{x_1, \ldots, x_j\}$ where $\mathrm{x_i} \in \mathbb{R}^n \ \forall \mathrm{i} \in \{1, \ldots, \mathrm{j}\}$[62]. The convex hull of $\mathcal{X}$, $H[\mathcal{X}]$ is the smallest convex region that contains $\mathcal{X}$. $H[\mathcal{X}]$ is a simplex for $j = n + 1$ (and points in general position), and $H[\mathcal{X}]$ has measure zero for $j \leq n$. For $n = 1$, the convex hull is the subset of the real numbers bounded by the minimum and maximum of the points in $\mathcal{X}$. Lopez-Pintado et al. [66], as well as Mirzargar et al. [73], generalize the function-band-depth formulation to curves, $\mathcal{C}_j(t) = \{c_1(t), \ldots, c_j(t)\}$ where $c_i : \mathbb{R} \mapsto \mathbb{R}^n$, using the parameterized set of convex hulls for points in $\mathbb{R}^n$. That is $B[\mathcal{C}_j](t) = H[\{c_1(t), \ldots, c_j(t)\}]$. Here we use a similar generalization strategy for paths on graphs, namely a parameterized convex hull on the vertices.

We define the *length* of a path p as the sum of weights along its edges, denoted $\|p\|$, and its *cardinality* $|p|$ is the number of constituting vertices. A *geodesic* between two vertices (u, v) is the path between them with the shortest length, and we denote this geodesic distance as $d_g(u, v)$. Geodesic (shortest) paths are not necessarily unique in a graph. In this chapter, to clarify the discussion, we will generally assume there exists some consistent way to decide among multiple geodesics (in our implementation we use the first geodesic found by Dijkstra's algorithm), and the theory and formulation can be extended to the possibility of multiple geodesics.

We begin with a definition of the band formed by vertices on a graph. Let us define subsets of vertices of size j as follows: $S_j = \{\mathcal{V} \subset \mathcal{P}(V) : |\mathcal{V}| = j\}$ where $\mathcal{P}(V)$ is the power set of V. A vertex v is said to lie in the *band* formed by $\mathcal{V}_j \in S_j$ if and only if it lies in the *convex hull* [81] of $\mathcal{V}_j$ on G. There are several formulations of convex hulls of a subset of vertices $\mathcal{V}_j$ on G; here we propose to use the *geodesic-convex hull* on G, because of its natural relationship to the simplex and convex hull band depth in $\mathbb{R}^n$. The geodesic-convex set of vertices on a graph is a set of vertices that is closed under geodesic paths (all geodesic

paths between all vertices in the set are contained in the set). The convex hull of a set $\mathcal{V}_j$, referred to as a $j-$simplex, is the smallest geodesic-convex set that contains $\mathcal{V}_j$ (and hence can be thought of as the *geodesic closure* of $\mathcal{V}_j$). We denote the convex hull of $\mathcal{V}_j$ by $H[\mathcal{V}_j]$.

In order to define band depth, we consider selecting j vertices independently from a probability distribution over the vertex set V given by $\text{Prob}_V(v)$ where $v \in V$. From these vertices we form $\mathcal{V}_j \in \mathcal{S}_j$. We can now ask if a vertex v falls inside the convex hull formed by our random selection of vertices, where the probability of this event is the product of the aforementioned vertex probabilities (by the independence assumption). Once in place, we can define the *graph-geodesic-hull band depth* of a vertex with respect to the j-simplex to be $vBD(v) = \text{Prob}\left(v \in H[\mathcal{V}_j]\right)$, where $\mathcal{V}_j$ is a set of j independent samples taken from the probability distribution we have defined for vertices.

If the graph is finite, the depth of a vertex can be computed in closed form. The band depth of v can be expressed as the expected value of the characteristic function χ for v falling within (or belonging to) a random j-simplex. That is,

$$vBD(v) = E_{\mathcal{V}_j \in \mathcal{S}_j}\left[\chi\left(v \in H[\mathcal{V}_j]\right)\right] = \sum_{\mathcal{V}_j \in \mathcal{S}_j} \chi\left(v \in H[\mathcal{V}_j]\right) \prod_{v_m \in \mathcal{V}_k} \text{Prob}_V(v_m). \tag{4.6}$$

This form also reveals that the proposed *graph-geodesic-hull band depth* is a more general formulation of *graph centrality* from graph theory[36]. That is, the *centrality* of a vertex in a graph has been quantified as the number of geodesic paths that pass through that vertex [36], which corresponds to $j = 2$ and $\text{Prob}_V(v) = 1/|V|$ in (4.6). Thus, graph-geodesic-hull band depth characterizes both the structure of the graph itself (and the centrality of points), as well as the probability distribution on the vertices.

The extension from vertices to paths proceeds as in the case of curves, with some additional technicalities. For this, we formulate a path on a graph as a mapping $p : \mathcal{I} \mapsto V$ over an index set $\mathcal{I} = [1, 2, \ldots, m]$ onto the vertex set V, and we use the notation $p(l)$ to denote the vertex of path p that is mapped from index $l \in \mathcal{I}$. The band formed by j paths sharing a common index set is the parameterized set of j-simplex bands formed by their corresponding vertices. Thus, we can index a set of j paths, $\mathcal{P}_j$, such that $\mathcal{P}_j(l) \in \mathcal{S}_j$ for all $l \in \mathcal{I}$.

The formulation for testing a path p against the band formed by a set of paths $\mathcal{P}_j$ that are parameterized over $\mathcal{I}$ is

$$p \in B[\mathcal{P}_j] \quad \text{iff} \quad p(l) \in H[\{p_1(l)), \ldots, p_j(l)\}] \quad \forall l \in \mathcal{I}. \tag{4.7}$$

The *band depth* of a path p is $\text{Prob}(p \in B[\mathcal{P}_j])$ where $\mathcal{P}_j$ is a set of j, independently drawn paths from the distribution $\text{Prob}(P = p)$. Similar to other notions of band depth, the path band depth can be computed as the expectation of the characteristic function of p being in the band of a randomly chosen set from the distribution of paths:

$$\text{pBD}(p) = E\left[\chi\left(p \in B(\mathcal{P}_j)\right)\right], \tag{4.8}$$

where $\mathcal{P}_j$ again represents a set of j, independently drawn paths from the distribution $\text{Prob}_{\mathcal{P}}(p)$ over all possible paths $\mathcal{P}$.

The expectation over the bands is approximated as a sample mean, from a random collection of j-sized subsets of an ensemble. In some cases, small sample sizes may interfere with the ability to estimate this expectation with sufficient accuracy to resolve differences in samples with low band depth. Thus, modified versions can either use a measure over an index set rather than a binary characteristic function [66] or relax the "for all" condition in Equation 4.7 to allow a certain number of vertices to fall outside the simplex band, as proposed in [115].

The proposed formulation for band depth on paths requires $\mathcal{P}_j$ and p to share a common index $\mathcal{I}$, which is effectively a discrete parameterization. However, in most applications, paths are specified as sequences of vertices, without a corresponding index set. Thus, one of the contributions of this work is a strategy for forming these common index sets as part of the construction of bands for paths.

A common index set between a collection of paths establishes a *correspondence* between vertices on a path such that for each vertex on each path there is a (nonempty) set of corresponding vertexes on every other path. Because the paths may be of different lengths, the correspondences are not unique. However, we propose that the mapping from the index set to a path should be monotonic with respect to the sequence of the vertices on the path (order of the vertices in paths is respected), and thus, the correspondences are monotonic between every pair of paths.

The correspondence between a collection of paths is computed using an optimal matching strategy, similar to what is used for string matching in computer science and sequence alignment in biological protein analysis [79]. The intuition behind this method is to assign

correspondences such that the correspondences are *monotonic*, and the overall sum of geodesic lengths between *corresponding* vertices along the paths is minimized. We first describe the method for finding correspondences between two paths. Given two paths p_l and p_m, an optimal correspondence is established by a pair of monotonic mappings from a common index set $\mathcal{I}$ to the paths, such that the distances between vertices are minimized. Thus, we are trying to find two mappings that minimize

$$\left(\sum_{k \in \mathcal{I}} d_g \left(p_l(k), p_m(k) \right) \right), \tag{4.9}$$

where $p_l(k)$ is the vertex on path p_l that is mapped from the index $k \in \mathcal{I}$, and $d_g(,)$ denotes the geodesic distance between two vertices. This formulation generalizes to collections of paths (> 2) by minimizing the sums of all pairs of distances among corresponding vertices in the collection of paths.

To find the correspondences among a set of paths, we use the classical method of dynamic programming (DP) on the matrix/tensor consisting of all possible correspondences—e.g., the Needleman-Wunsch algorithm [20]. All pairwise distances are organized in a tensor with an order that is the number of paths to be aligned. Thus, the number of distances considered in the optimization is $\prod_{l=1}^{j+1} |p_l|$, which grows exponentially with the number of paths forming the band (generally, the problem is NP-Hard [51]). There are existing efficient, approximate algorithms for large numbers of paths [20], but that issue is beyond the scope of this chapter. For the results presented here, we use $j \leq 3$ and rely on the basic (full enumeration of tensor) approach for optimization.

In Figure 4.2 we see a band formed by three paths— p_a, p_b and p_c. Here, the elements from common index $\mathcal{I} = [1, 2, 3]$ are mapped to vertices on the graph from each path. Path p_x is completely contained within the band as all of its vertices are part of a j-simplex formed by corresponding vertices that are mapped from the same element in $\mathcal{I}$ to p_a, p_b and p_c. Similarly, we observe that no vertex from p_y is contained in any j-simplex. Also, two elements from $\mathcal{I}$ are mapped to a single vertex in p_c as it is shorter than the other paths. Once we are able to describe a band formed by a set of paths, we can generate order statistics on an ensemble of paths by calculating the path band depth of each member within the ensemble.

An ordering of the data based on *path band depth* readily yields a set of rank statistics.

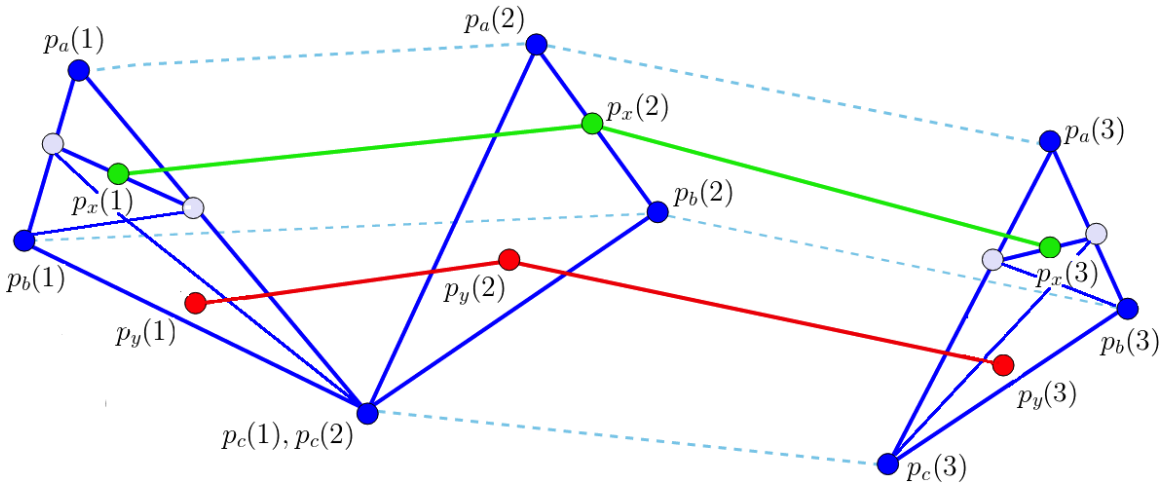


Figure 4.2. Band formed by three dashed paths on a complete graph whose edge weights are equal to the Euclidean distance between vertices (only selected edges are drawn). The green path is completely contained within the band according to definition in Equation 4.7, whereas the red path falls completely outside the band. Solid blue edges constitute the geodesics connecting vertices within graph simplices.

The median is the path with the highest probability of falling within a random band—(i.e., the *deepest* ensemble member). The 50% band consists of paths whose probabilities are in upper half percentile of all probabilities. The 100% envelop is formed by excluding the outliers. We define outliers (as in [103]): $pBD(p) < pBD(p_{median}) - \alpha \times \big(pBD(p_{median}) - pBD(p_{50\%})\big)$ where $p_{50\%}$ is the band depth value that splits the ensemble into equal parts, and $\alpha = 1.5$ is a typical value as found in the literature [103]. For the results shown in this chapter, we used values of α in the range 2.4 to 3.7 in order to flag only the most nonrepresentative paths as outliers. Furthermore, we used the modified formulation of band depth [66], in order to resolve depth with sufficient accuracy to avoid ties.

By convention, data-depth formulations in flat spaces (e.g., simplex depth in $\mathbb{R}^n$) are considered desirable if they demonstrate a set of properties that are consistent with classical methods on certain classes of distributions. For instance, [124] have proposed affine invariance, maximal depth around a point of symmetry, monotonic fall off with distance from a central point, and zero depth for points at infinity. Although some of these properties have yet to be developed for general graph structures, in the appendix we prove the asymptotic depth property for points at infinity for vertices and paths.

4.4 Path Boxplot Visualization

Here we develop a visualization for the proposed analysis in a manner similar to what has been proposed for functions, contours, and curves [40, 73, 103, 115]. The proposed visualization approach is motivated by the classical whisker plot or boxplot, and relays a display of the median, 50% band, 100% band, and outliers for graph-based path ensembles. Figure 4.3a shows a synthetically generated path ensemble with each path drawn using a random color. Figures 4.3c and 4.3d show two variations of our proposed visualization described next.

We render the visualizations in a way that describes the rank statistics of the distribution or ensemble. We first establish the placement of vertices and edges either intrinsically or via a layout algorithm [41]. Next, we use color and width/thickness on edges and vertices to represent their rank. The paths in the 100% band are drawn thickest in light

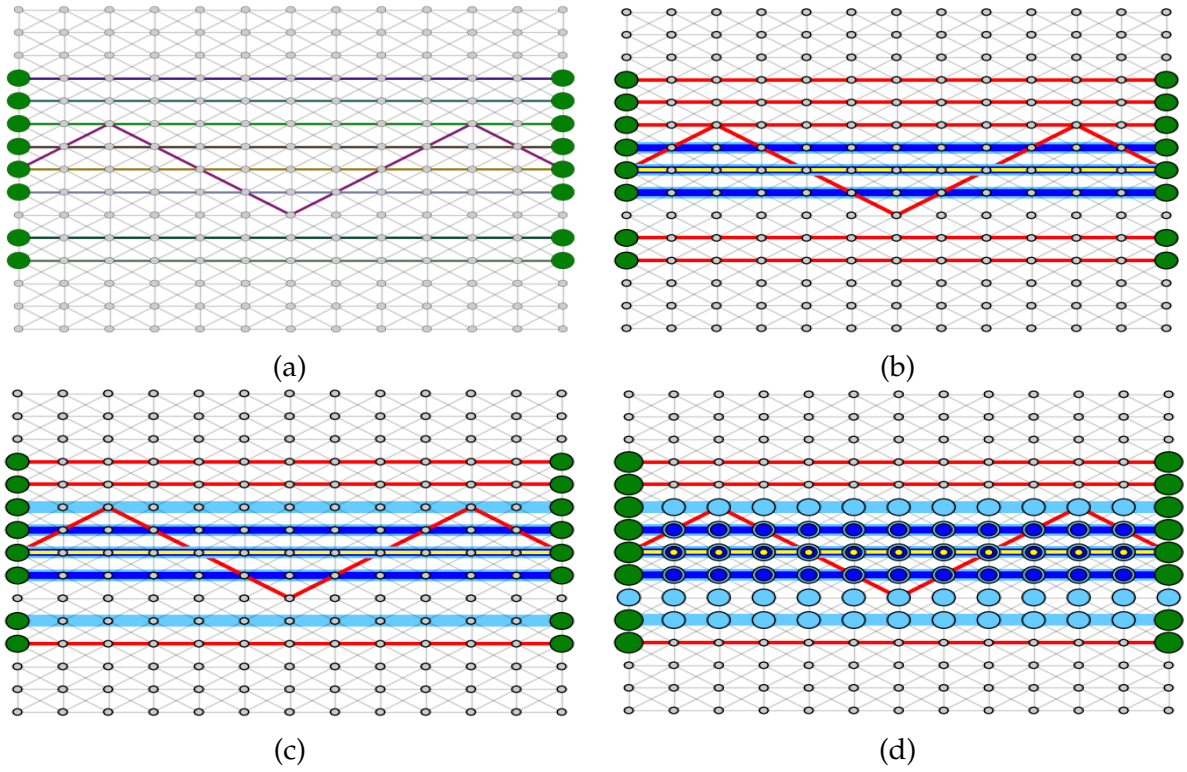


Figure 4.3. Synthetic example 1. (a) A path ensemble with each path rendered with a random color. (b) Path boxplot using rank statistics based on the sum of Fréchet distances. (c) Path boxplot based on path band depth (visualization *without* vertex encoding). (d) Path boxplot based on path band depth (visualization *with* vertex encoding).

blue. The paths in the 50% band are drawn using a thinner dark blue stroke on top of the thicker light blue band. This drawing of the thinner dark blue stroke *over* the thicker light blue stroke is done to indicate that the path in the 50% band is contained within the 100% band as well. Continuing this strategy, the *median* path is drawn using a thin yellow stroke drawn over a thicker dark blue stroke, which in turn, is drawn over the thickest light blue stroke. To signify that the outlier paths lie outside even the 100% envelop, they are drawn using only a thin red stroke. Figure 4.3c shows a version of the path boxplot that uses the described encoding for paths. A variation of this approach is seen in Figure 4.3d where the vertices are also encoded, based on their position, in addition to the edges in the graph. Vertices that are not part of the convex hull formed by any set of corresponding vertices between paths are drawn as small gray circles. Vertices that are in the convex hulls formed by paths in the 50% band are drawn using a light blue circle. Analogous to the encoding for the paths, vertices in the convex hulls formed by paths in the 50% band are drawn using a deep blue circle contained within a larger light blue circle, whereas vertices lying on the median path are marked with an additional yellow circle drawn within the deep blue circle, which is itself contained within a light blue circle.

The sections that follow demonstrate applications of the proposed method on synthetic examples and data sets from applications in transportation and computer networks. We use the visualization approach *with* vertex encoding (as seen Figure 4.3d and Figure 4.4c) for all further path boxplot visualizations based on path band depth in this chapter except when vertices on the graph are not rendered.

4.5 Results

We begin by showing results for two synthetically generated path ensembles on a graph. For these ensembles, we show path boxplot visualizations generated using rank statistics obtained by path band depth analysis, as well as, the Fréchet distance metric analysis [32]. For these path ensembles, the results were identical upon replacing the Fréchet metric with the Hausdorff metric, and therefore we show only one of these methods. When using a distance metric, we rank each path using the sum of its distances from all the other paths in the ensemble. Hence the path that *minimizes* this sum is identified as the median. Note that this is different from path band depth where the median path

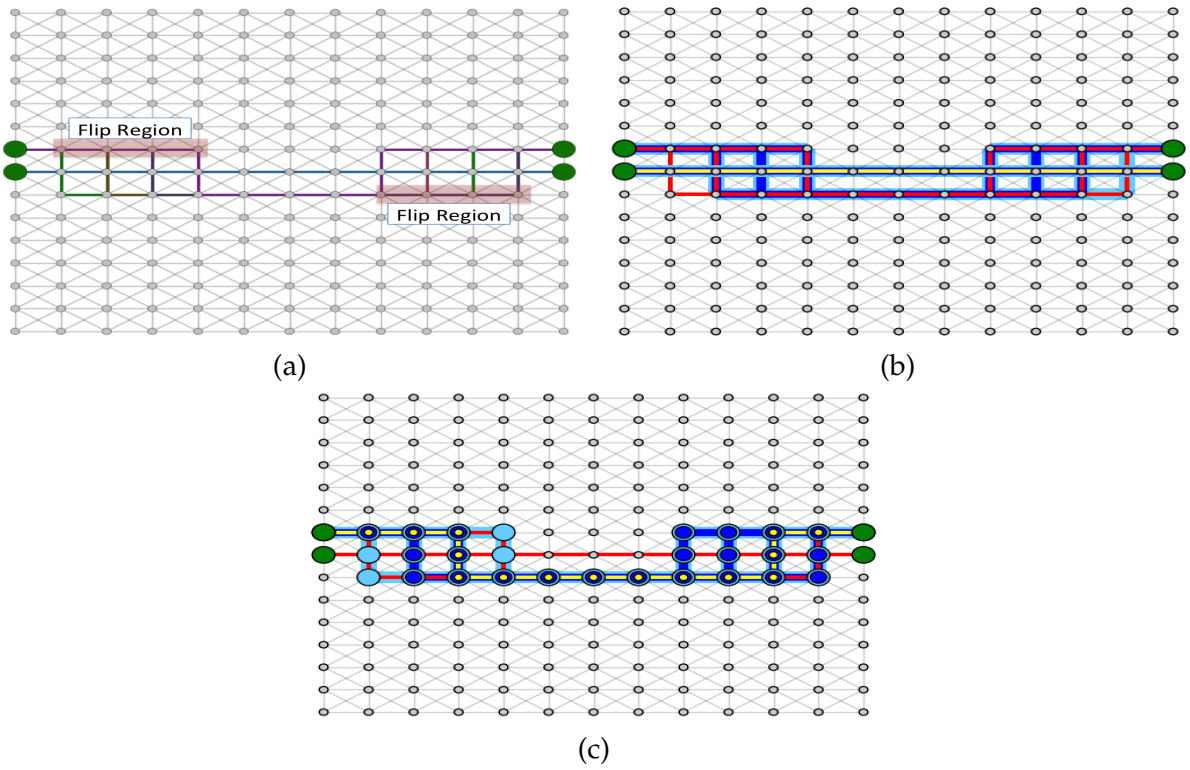


Figure 4.4. Synthetic example 2. (a) A path ensemble with each path rendered with a random color. (b) Path boxplot using order statistics based on the sum of Fréchet distances. (c) Path boxplot based on path band depth.

has *maximum* depth. The underlying graph in both our examples is associated with a regular, *diagonal grid* (constructed from including diagonals in a conventional, structured quadrilateral grid).

For the first of these examples (see Figure 4.3), we generate an ensemble of 20 paths by sampling with replacement from a set of straight paths (all vertices in the path have the same ordinate) spanning the horizontal extent of the grid. The ordinate of each path comes from a random variable associated with a normal distribution centered at the central ordinate of the grid. We complete the ensemble by adding a simulated outlier in the form of a zigzag path (see Figure 4.3a). In Figure 4.3b we see the path boxplot visualization of Fréchet distance-based depth. Figures. 4.3c and 4.3d show two versions of the path boxplot of the path band depth analysis. In this simple example we see that the result from path band depth analysis is very similar to distance-metric-based analysis with both approaches identifying the zigzag and peripheral paths as outliers.

We now present an example where distance-metric-based methods fail to detect the general structure (median) and anomalous path (outlier) in an ensemble. Further, we see that path band depth analysis is able to correctly make this determination by capturing the nonlocal correlations in the path ensemble. Here, we produce an ensemble of 20 straight paths spanning the grid's horizontal extent, starting and ending at vertices with the same ordinate (see Figure 4.4). In this case, however, each path is required to undergo flips when traversing the flip regions as seen in Figure 4.4a. The vertex within each zone where the flip occurs is chosen uniformly from among the vertices in each zone. We add a simulated outlier to this ensemble in the form of a path with no flips (Figure 4.4a). In this case we see that the distance-based metrics (Figure 4.4b) identify the simulated outlier as the median (most representative) whereas the path band depth method (Figure 4.4c) selects one of the randomly sampled paths as the median. The simulated outlier is *closest* to other paths with regard to the distance metrics, but it is identified as an outlier by the path band depth analysis.

4.5.1 Transportation Networks

We used publicly available road data from OpenStreetMaps (OSM) [43] for a randomly chosen region in Los Angeles, California. Figure 4.5a shows a part of the road graph overlaid on a map. We used *expected travel time* between the two adjacent vertices, obtained by querying the open-source routing engine Gosmore, as the weight of each edge. Travel time along a short road segments can be modeled using a normal distribution [45]. We obtained an ensemble of 20 paths between two random vertices by repeatedly finding the lowest cost path on the graph whose edge weights were picked, after each iteration, from a normal distribution centered at the expected travel time for that edge.

For visualizing the paths, we used the geographical coordinates of the vertices on the road graph for layout. A map, also based on OSM data, is provided in the background for context in accordance with the common practice for viewing geographical routes (see Figures 4.5a and 4.5b). In order to have the overlaid paths align with underlying roads on the map and also be feasible with regard to traffic restrictions, we used the Gosmore routing service to obtain the geographic coordinates of the spatial path drawn between every pair of adjacent vertices along each path in our ensemble. Such alignment is necessary for drawing road segments that are curved or where the direct connection between two

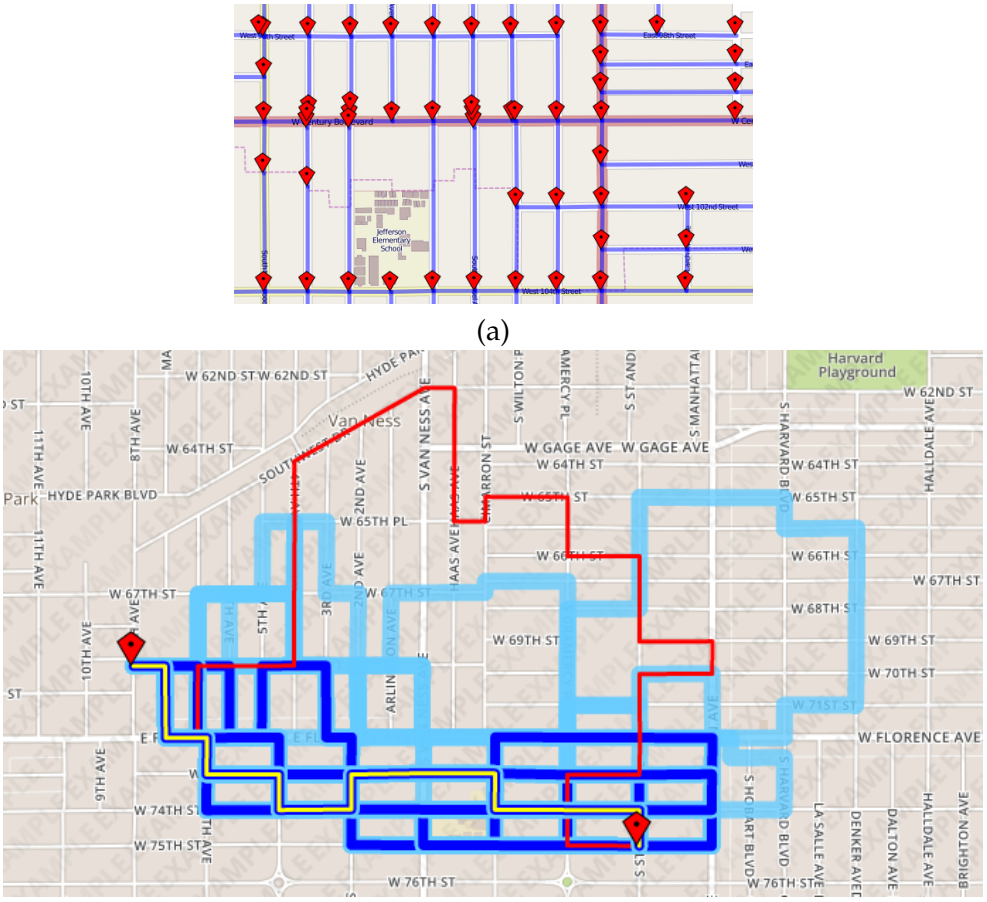


Figure 4.5. Road network: (a) A section of the road graph overlaid on a map representing actual spatial embedding of vertices and edges. (b) Path boxplot for an ensemble of paths on a road network.

vertices is illegal according to local traffic rules.

A path boxplot of a path ensemble on a road graph is shown in Figure 4.5b. The most representative path or the median path seen here can be useful when the requirement is to select a particular path from a collection of paths on a road graph. For instance, a median path would be a good choice of a path that affords quick access to a number of alternate paths, which would be useful in situations involving- high traffic conditions or blockages. The path boxplot would also find utility for planning bicycle corridors [33, 34].

4.5.2 Computer Networks (Autonomous Systems)

We used a subset of the AS graph as well as path ensembles of packets traveling between ASes on that graph from a set of path snapshots seen from the Oregon Routeviews

server. For clarity, we filtered out vertices in the graph that did not lie on a geodesic between any pair of vertices in the path ensemble. Additionally, in the visualization we included only a single geodesic between all pairs of vertices in the ensemble. For the graph layout in 2D, we modified the force directed model in [38] by including an extra repulsion between the vertices at the two endpoints, so that they were placed at nearly opposite ends of the layout. Also, the charge/repulsion on each vertex was made proportional to its degree for avoiding congestion near high-degree vertices.

We looked at several destinations that had significant variations in their paths throughout the year. Visualizations of a few of these ensembles can be seen in Figure 4.6, Figure 4.7, and Figure 4.8. Looking at a selection of these ensembles, there are some special cases identified as outliers. Figure 4.6 shows an outlier where the outlying path is of the same cardinality as the median path and does not contain any unique vertices or edges causing it to be undetectable by common heuristic methods used to analyze network traffic. Other cases include Figure 4.7 and Figure 4.8 , where the outlier path bypasses other paths

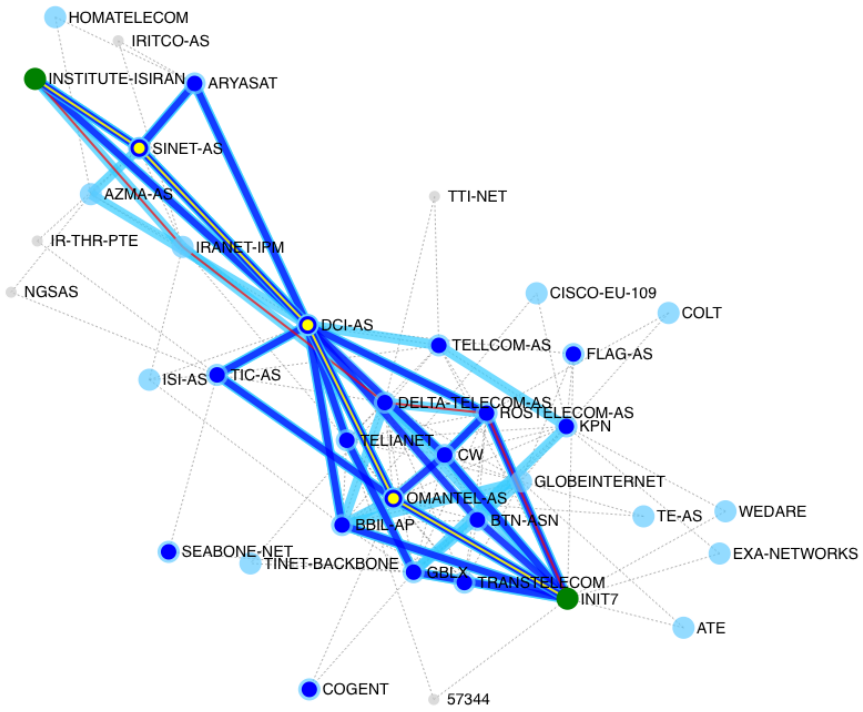
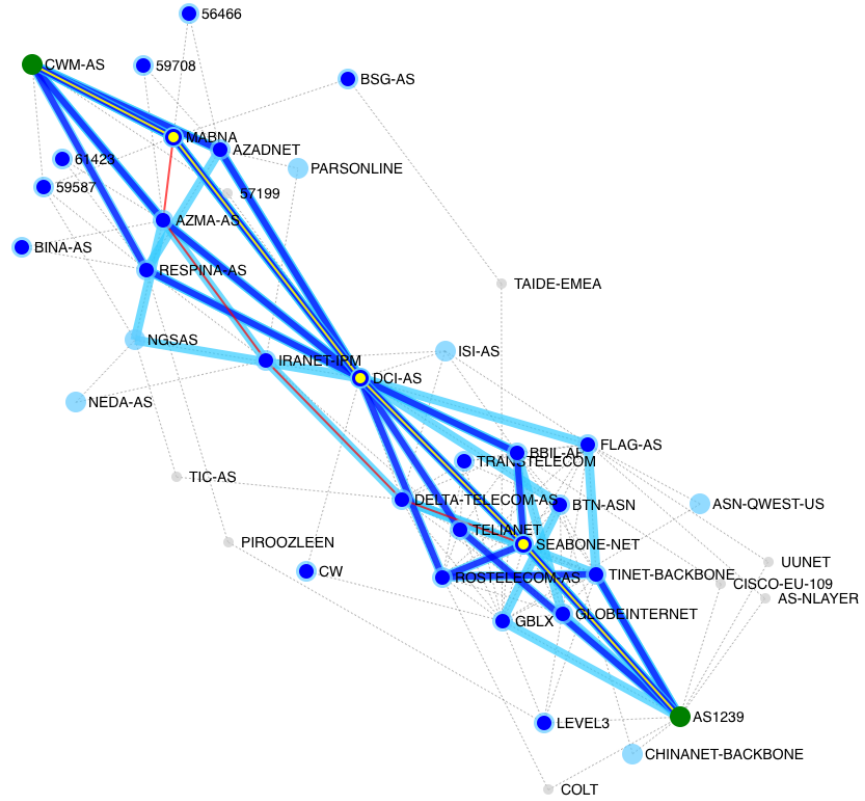
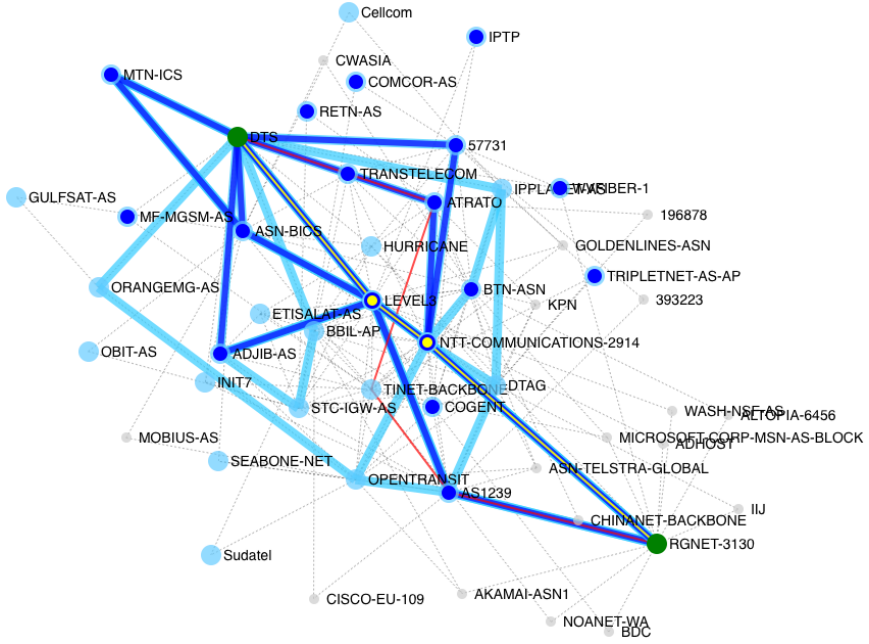


Figure 4.6. Outlier paths on AS graph: A class 1 outlier with no unique vertices/edges in the outlier path.

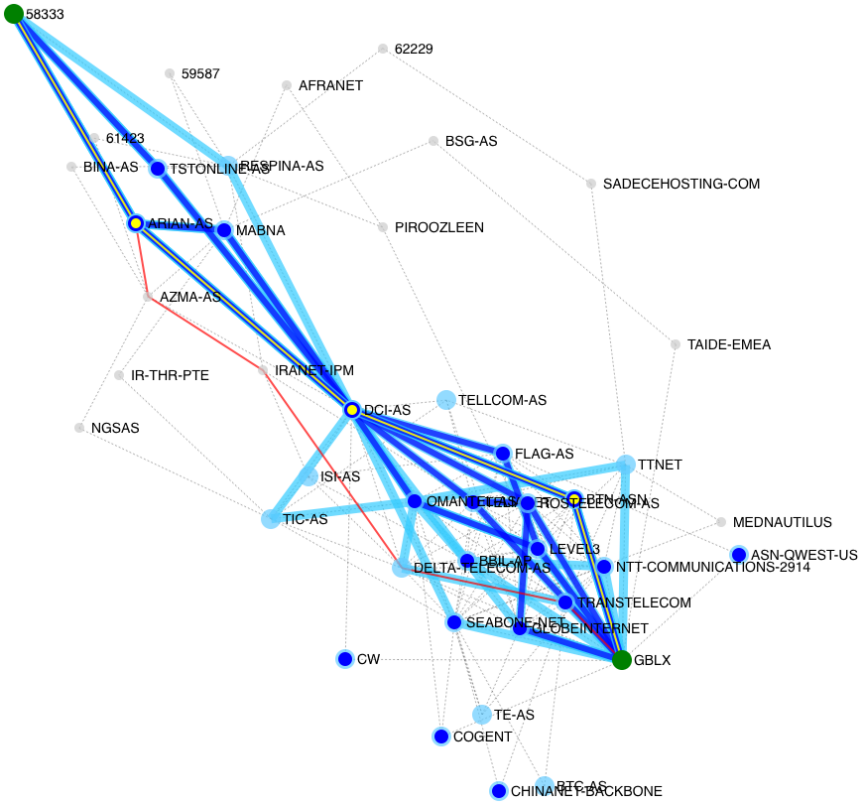


(a)

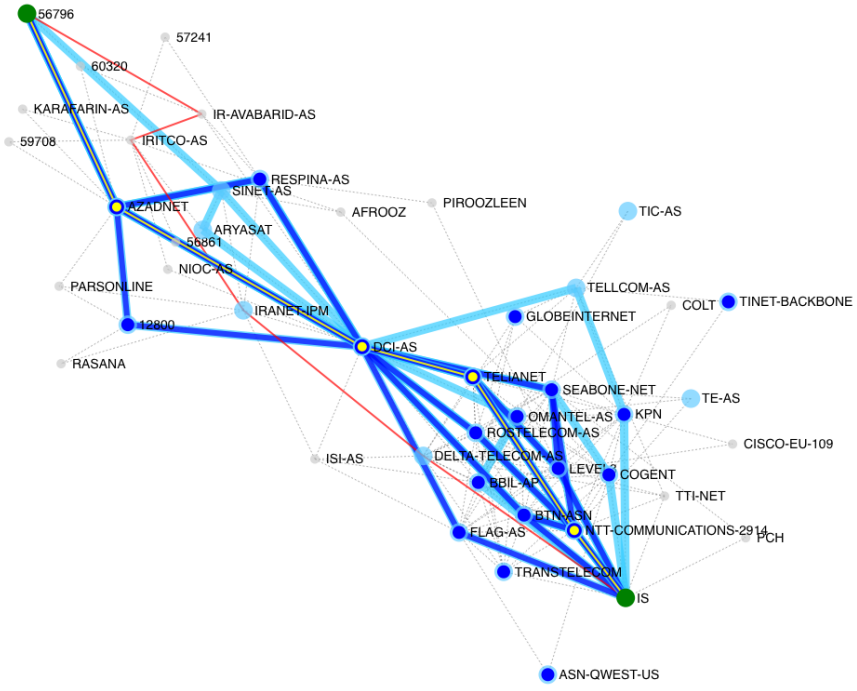


(b)

Figure 4.7. Outlier paths on AS graph: (a) Class 2 outlier: One unique edge, no unique vertices in the outlier path. (b) Class 3 outlier: Outlier appears to be one hop, bypassing a more normal route



(a)



(b)

Figure 4.8. Outlier paths on AS graph: (a) Class 4 outlier: Outlier is two hops around a more normal route; and (b) Class 5 outlier: Outlier takes several hops around the usual path.

through unique edges or vertices. Cases where they depart near one of the endpoints, such as 4.8b, may be relatively straightforward for the operators of those edge networks to detect, as their own routers will directly see the change in where traffic enters or exits their networks. Cases such as Figure 4.7b, however, exhibit changes in networks that may not be directly visible from the endpoints, and yet affect the overall behavior of traffic to/from these endpoints. These are cases that can be particularly difficult to discover and diagnose; a path boxplot can aid operators in assessing such cases.

4.6 Conclusion and Future Work

Assigning centrality-based ordering for an ensemble of paths is useful in many applications. Although robust band-depth-based methods for calculating order statistics have been recently introduced for various kinds of ensembles on a continuous domain, they cannot be employed in cases where the ensemble members are described on a graph. We identify the challenges in extending this approach to paths on a graph and present a solution in the form of a novel notion of depth denoted as path band depth. A visualization scheme based on this new notion of depth called path boxplot is also introduced. This chapter demonstrates the utility of the path boxplot for helping users understand the overall structure of the ensemble using synthetic data as well as data from two real application areas, path ensembles on autonomous system (AS) graphs and on road graphs.

Although a robust method for generating order statistics for path ensembles, the proposed analysis is computationally intensive due to its combinatorial nature. The topology of the underlying graph as well as the density of its edges also affects the computation time by a constant factor. A practical approach to deal with larger ensembles (with a large number of paths) is to trade running time for an approximate solution by randomly selecting a subset from the set of all possible bands as suggested in [66]. In the case of ensembles with long paths, skipping vertices in the description of the paths may also provide an acceptable compromise between accuracy and performance. Developing a heuristic for skipping vertices in large ensembles of long paths to achieve an optimum trade-off between running time of analysis and the quality of the *solution* would be an interesting avenue for future work. It would also be interesting to explore the application of path boxplot in other areas such as in mobile ad hoc networks, which can be modeled

as a graph with dynamic topology [75] and in molecular dynamics, to identify a most representative path as an alternative to computing the mean statistics for the ensemble.

CHAPTER 5

ANISOTROPIC RADIAL LAYOUT FOR VISUALIZING CENTRALITY AND STRUCTURE IN GRAPHS

Portions of this chapter have been reproduced with permission from Springer and is based on material published in Proc. GDNV, Anisotropic Radial Layout for Visualizing Centrality and Structure in Graphs, M. Raj and R. T. Whitaker, 2018, pp. 351-364 [91].

5.1 Introduction

Graphs are an important data structure that are used to represent relationships between entities in a wide range of domains. An interesting aspect in graph analysis is the notion of (structural) *centrality*, which pertains to quantifying the importance of entities (or vertices, nodes) within the context of the graph structure as defined by its relationships (or edges). The need to compute centrality and convey it through visualization is seen in many areas, for example, in biology [96], transportation [16], and social sciences [15]. In this work, we propose a method to visualize node centrality information in the context of overall graph structure, which we capture through intervertex (graph theoretical) distances. The proposed method determines a *layout* (positions of nodes on a 2D drawing) that meet the following two, often competing, criteria:

- *Preservation of distances:* The Euclidean (geometrical) distances in the layout should approximate, to the extent possible, the graph theoretical distances between the respective nodes.
- *Anisotropic radial monotonicity:* Along any ray traveling away from the position of the most central node, nodes with a lower centrality should be placed geometrically further along the ray.

We also introduce a visualization strategy for the proposed layout that further highlights

the centrality and structure in the graph by using additional encoding channels, and demonstrate the benefits of our approach with real data sets (see Figure 5.1 as an example).

Visualization methods for gaining insights from graph-structured data are an important and active area of research. Significant efforts in this area are targeted toward developing effective layouts. Layout methods can have various goals that range from trying to reduce clutter and edge crossings [17] to faithfully representing the structure by preserving the distances between nodes and topological features [39]. As positions are the best way to graphically convey numbers [22], layouts are also used to convey numerically encoded measures of hierarchy or the importance associated with nodes [15, 28].

Radial layouts have been shown to be an effective method to visually convey the relative *importance* of nodes, where importance may be defined, for instance, by a node's *centrality* [15]. The centrality of a node is a quantification of its importance in a graph by considering its various structural properties, such as connectedness, closeness to others,

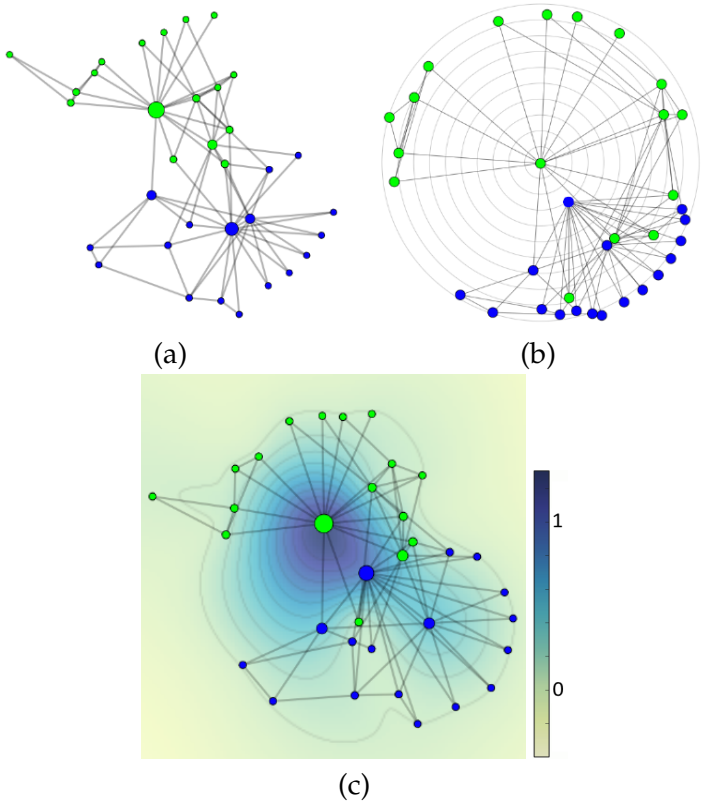


Figure 5.1. Visualization of Zachary's karate club social network using (a) MDS, (b) radial layout, and (c) anisotropic radial layout. Node sizes encode betweenness centrality.

and role as an intermediary [37, 125]. In conventional radial layouts, the distance of nodes from the geometric center (origin) of the layout depends *only* on the node's centrality, and nodes with a higher centrality value are placed closer to the origin in the layout, oftentimes forming rings or concentric circles.

Given a graph and centrality values associated with its nodes, several approaches have been proposed to determine a radial layout. One line of work, which deals with discrete centrality values, attempts to minimize edge crossings [9]. Another approach, which also tackles continuous centrality values, involves optimizing a *stress* energy (5.2.2) by including a penalty for representation error (of graph distances) as well as deviation from radial constraints [15, 16]. The penalty acts as a *soft* constraint wherein the solution is allowed to deviate from the constraint at the expense of increased local stress. The literature shows that radial constraints may also be included as a *hard* constraint by allowing only those solutions that satisfy the constraints [10, 27, 29].

Although state-of-the-art methods for radial graph layout do effectively convey node centrality, the associate circular centrality constraints make it difficult to preserve other important, structural graph characteristics such as distances, which, in turn, makes it difficult to preserve the holistic structure of the graph. On the other hand, despite being effective in preserving the overall structure, general layout methods such as multidimensional scaling often fail to readily convey centrality (e.g., by failing to ensure that structurally central nodes in the graph-theoretical sense appear near the center of the layout and vice versa). In this chapter, we propose a method that simultaneously tackles both the above issues.

The underlying idea for the proposed layout algorithm is that we can relax the constraint that requires nodes with similar centrality to lie on a circle, and instead, allow for such nodes to be constrained by a more general shape: a simple closed curve or *centrality contour*. Centrality contours are nested isolevel curves on a smooth, radially decreasing estimate of node centrality values over a 2D field. We demonstrate that the additional flexibility in placing the nodes afforded by the centrality contours over circles, in conjunction with some additional visual cues in the background, lets us achieve a better trade-off than existing methods in conveying centrality and general structure together.

5.2 Background

In this section, we describe the various underlying technicalities that are relevant to the proposed method, and begin with some notation/definitions.

We define a weighted, undirected graph $G(V, E, W)$ as a set of vertices (or nodes) V, a set of edges $E \subseteq V \times V$, and a set of edge weights, $W : E \mapsto \mathbb{R}^+$, assigned to each edge. We define n to be cardinality of node set, i.e., $n = |V|$. The graph-theoretical distance (shortest-path along edges) between two nodes u and v is denoted by d_{uv}. We denote a general position in a 2D layout as $\mathbf{x} = \{x, y\}$ and the Euclidean distance between two nodes u and v as $\delta(\mathbf{x}_i, \mathbf{x}_j) = ||\mathbf{x}_u - \mathbf{y}_v||_2$.

5.2.1 Centrality and Depth

The need to measure, and quantify, the importance of individual entities within the context of a group occurs in many domains. In graph analytics, this need is addressed by *centrality* indices, which are typically real-valued functions over the nodes of a graph [125]. The specific properties that qualify the importance of nodes may depend on the application or data type, and several methods to compute centrality have been proposed, such as degree centrality [37], closeness centrality [94], and betweenness centrality [37]. Although the emphasis of the various centrality definitions can be different, they share a common characteristic of depending only on the *structure* of the graph rather than parameters associated with the nodes [125]. For the examples in this chapter, we use betweenness centrality due to its relevance to the data sets (Section 5.4).

The *betweenness centrality* of a node, $v \in G$, is defined as the percentage (or number) of shortest paths in the entire graph G that pass through the node v. As shown in the work of Raj et al. [90], barring instances of multiple geodesics, betweenness centrality is a special case of a more general notion of *vertex depth* on graphs—a generalization of data depth to vertices on graphs. Data depth is a family of methods from descriptive statistics that attempts to quantify the idea of centrality for ensemble data without any assumption of the underlying distribution. Data-depth methods often rely on the formation of *bands* from convex sets and the probability of a point lying within a randomly chosen band. The extension of band depth to graphs [90] relies on the convex closure of a set of points (via shortest paths), and thereby generalizes betweenness centrality by considering bands

formed by sets of nodes, rather than only the shortest paths between pairs of nodes, and allows for a nonuniform probability distribution over the nodes of the graph.

In addition to graphs, data-depth methods have been proposed for several other data types such as points in Euclidean space [107], functions [66], and curves [67, 73]. Despite their distinct formulations, data-depth methods are expected to share a few common desirable properties [124] such as: 1) maximum at geometric center; 2) zero at infinity; and 3) radial monotonicity, which make data depth an attractive basis for ensemble visualization methods [73, 93, 102]. Graph centrality is a type of data depth on the nodes of a graph, and here we pursue layout methods that convey these depth properties.

5.2.2 Stress and Multidimensional Scaling (MDS)

Our proposed method is based on a modification to the MDS objective function, and therefore we give a brief summary of MDS. MDS is family of methods that help visualize the similarity (or dissimilarity) between members in a data set [13]. Over the years, MDS has been the foundation for a range of graph drawing algorithms that aim to achieve an isometry between graph-theoretical and Euclidian distances between nodes [16, 52]. From among various types of MDS methods that exist, here we consider *metric MDS with distance scaling*, which is popular in the graph drawing literature [39] (see Figure 5.2 for an example).

In the context of graph drawing, given a distance matrix based on the graph-theoretical

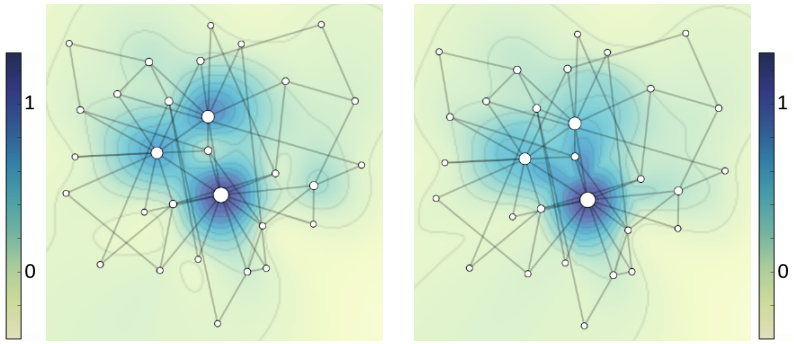


Figure 5.2. Interpolation and monotonic fields for a sample graph. An (a) *interpolation field* for node centrality values and (b) the associated (radially) *monotonic field* for a 30-node random graph generated using the Barabasi-Albert model. Node positions are determined using MDS and node sizes encode betweenness centrality.

distance, the goal is to find node positions $\mathbf{X} = \{\mathbf{x}_i : 1 \leq i \leq n\}$ that minimize the following sum of squared residuals—also known as *stress*:

$$\sigma(X) = \sum_{u,v} w_{uv}\big(d_{uv} - ||\mathbf{x}_u - \mathbf{x}_v||_2\big)^2, \tag{5.1}$$

where $w_{uv} \geq 0$ is the weighting term for residual associated with pair u, v. In the proposed work we employ a standard weighting scheme for graphs, known as *elastic scaling* [70], by setting $w_{uv} = d_{uv}^{-2}$. Elastic scaling gives preference to local distances by minimizing *relative* error rather than *absolute* error during the optimization.

Node positions that minimize the objective (5.1) have been shown to be visually pleasing and convey the general structure of the graph [52]. Although the state-of-the-art approach for optimizing the objective function is *stress majorization* [39], we employ standard *gradient descent* because of its compatibility with the proposed modification to the objective (Section 5.3). The gradient of the standard MDS objective is as follows [13]:

$$\nabla\sigma(\mathbf{X}) = 2\mathbf{V}\mathbf{X} - \mathbf{B}(\mathbf{X})\mathbf{X} \tag{5.2}$$

where matrices $\mathbf{V} = (v_{ij})$ and $\mathbf{B} = (b_{ij})$, with $1 \leq i, j \leq n$, can be compactly represented as

$$v_{ij} = \begin{cases} -w_{ij} & \text{for } i \neq j \\ \sum_{j=1, j\neq i}^{n} w_{ij} & \text{for } i = j \end{cases} \qquad b_{ij} = \begin{cases} -\frac{w_{ij} d_{ij}}{\delta(\mathbf{x}_i, \mathbf{x}_j)} & \text{for } i \neq j \text{ and } \delta(\mathbf{x}_i, \mathbf{x}_j) \neq 0 \\ 0 & \text{for } i \neq j \text{ and } \delta(\mathbf{x}_i, \mathbf{x}_j) = 0 \end{cases}$$

$$b_{ii} = -\sum_{j=1, j\neq i}^{n} b_{ij}.$$

5.2.3 Strictly Monotone and Smooth Regression

The proposed method also relies on the construction of a smooth and radially decreasing approximation of centrality values over a 2D field, which we call the *monotonic field* (Figure 5.2). The first part of this construction is an interpolation of centrality values of sparsely located nodes on the layout to obtain a dense 2D field, which we call the *interpolation field* (Figure 5.2a). We use *thin plate splines* [12] interpolation, a standard technique for interpolating unstructured data that produces optimally smooth fields.

The next part is to construct a radially monotonic approximation of the interpolation field. We devote the rest of this section to a brief description of the method that we use for constructing this approximation (monotonic field), which is adapted from Dette et al. [25, 26].

For a 1D function [25], $m(t) : [0,1] \rightarrow \mathbb{R}$, an elegant algorithm for computing its monotonic approximation $\hat{m}_A(t)$ proceeds as follows in *two* steps [25]:

- *Step 1 (Monotonization)*: Construct a density estimate from sampled values of input function m and use it as input to compute an estimate of the inverse of the regression function $\hat{m}_A^{-1}$.

$$\hat{m}_A^{-1}(t) = \frac{1}{Q\omega} \sum_{i=1}^{Q} \int_{\infty}^{t} K\left(\frac{m\left(\frac{i}{Q}\right) - u}{\omega} \right) du, \tag{5.3}$$

  where Q is the parameter controlling the sampling density, K is a continuously differentiable and symmetric kernel, and ω is the bandwidth. Here, $\hat{m}_A^{-1}$ is a strictly *increasing* estimate of m^{-1}; however, we can easily obtain a strictly *decreasing* estimate by reversing the limits on the integral in (5.3).

- *Step 2 (Inversion)*: Obtain the final estimate of $\hat{m}_A$ by numerically inverting $\hat{m}_A^{-1}$.

In order to obtain an approximation to a 2D function that is monotonic along radial lines emanating from the deepest or most central node, we use a polar coordinate representation of the field. We build the polar representation by sampling the interpolation field along 360 evenly spaced, center-outward rays. The idea is to repeatedly monotonize the interpolation field with respect to a single variable, i.e., for a fixed value of the angular coordinate, obtain a (1D) estimate that is strictly decreasing along the radial coordinate. We then repeat this process, successively monotonizing 1D functions that correspond to each value of the angular coordinate in its (discrete) domain; see Figure 5.2b for an example of the resulting monotonic field. The spline interpolation is smooth, and by the properties of the monotonic approximation (see [26]), the resulting monotonic field is smooth (except at origin, where polar the coordinates maybe nonsmooth).

5.3 Method

Here we describe our method in two parts. First is the layout algorithm (Section 5.3.1), and second is a visualization strategy (Section 5.3.2) that complements the layout to simultaneously convey graph structure and node centrality.

5.3.1 Anisotropic Radial Layout

In addition to preserving the graph-theoretical distances, we also aim to place every node on a radially monotonic approximation of a centrality field—called the *monotonic field* (Section 5.2.3)—such that the value of the field at the location of the node is equal to the centrality value of the node. We accomplish this by modifying the (distance preserving) MDS objective or *stress* (Section 5.2.2) to incorporate the following penalty term, which penalizes the deviation of monotonic field values from the node centrality values

$$\rho(\mathbf{X}) = \big(M_{\mathbf{X},\mathbf{c}}(\mathbf{X}) - \mathbf{c}\big)^2, \tag{5.4}$$

where $\mathbf{c} \in \mathbb{R}^n$ is a vector of node centrality values and $\mathbf{X} \in \mathbb{R}^{n\times 2} = \{\mathbf{x}_i : 1 \leq i \leq n\}$ denotes associated node positions. $M_{\mathbf{X},\mathbf{c}}(\mathbf{X}) \in \mathbb{R}^n$ denotes a vector of values of the 2D monotonic field at locations $\mathbf{X}$. The symbols in the subscript ($\mathbf{X}$ and $\mathbf{c}$) denote the use of node positions and centrality values in the construction of the monotonic field. In the limiting case where the *interpolation field* (Section 5.2.3) itself is monotonic, the value of this penalty term drops to zero. Our final objective is a sum of the MDS stress and the above penalty term, and can be stated as follows:

$$\gamma\big(\mathbf{X}\big) = \underbrace{\sigma(\mathbf{X})}_{\text{MDS stress}} + \; w_\rho \, \rho(\mathbf{X}), \tag{5.5}$$

where w_ρ is a weighting factor that controls the influence of the penalty, with respect to the MDS stress. The gradient of the modified objective above is obtained as

$$\nabla\gamma\big(\mathbf{X}\big) = \nabla\sigma(\mathbf{X}) + w_\rho \times \underbrace{2\big(M_{\mathbf{X},\mathbf{c}}(\mathbf{X}) - \mathbf{c}\big) \odot \nabla M_{\mathbf{X},\mathbf{c}}(\mathbf{X})}_{\nabla\rho(\mathbf{X})}, \tag{5.6}$$

where $\odot$ denotes element-wise product. It is difficult to compute the gradient of $M_{\mathbf{X},\mathbf{c}}(\mathbf{X})$ because of the dependence of M on $\mathbf{X}$ and the associated process for monotonic approximation. Therefore, we let the field lag, and treat $\mathbf{X}$ (in subscript) as a constant when numerically approximating the gradient of M. We deal with the resulting accumulation of error by recomputing the depth field after a fixed number of iterations, or *lag*, denoted by ℓ.

The parameters w_ρ and ℓ need to be chosen carefully. w_ρ needs to be set to find a balance between preserving the intrinsic graph structure and ensuring that the centrality of nodes matches the field value at their position. Figures 5.3a-c show, respectively, results of a

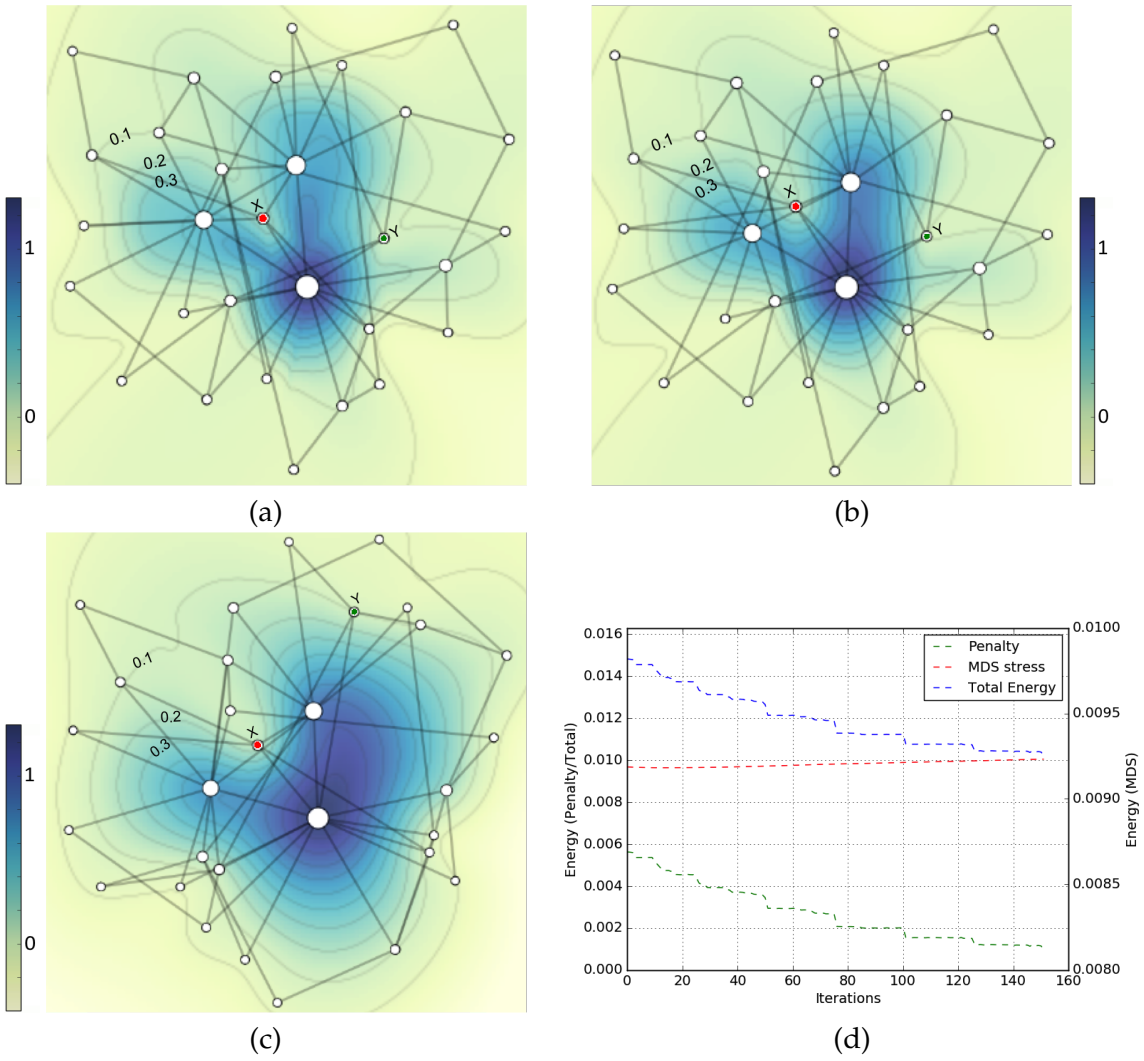


Figure 5.3. Sensitivity of anisotropic radial layout to penalty weights for the graph in Figure 5.2: (a) $w_\rho = 0.1$, (b) $w_\rho = 1$, (c) $w_\rho = 10$; centrality contours with isovalues 0.1, 0.2, and 0.3 as well as nodes X (red) and Y (green) with centrality values 0.2 and 0.1 are identified, and (d) a typical plot of objective energy during the optimization process ($w_\rho = 1$).

small w_ρ unable to move nodes to appropriate positions with regard to the field (observe nodes X,Y), an *intermediate* w_ρ, and a *large* w_ρ resulting in unnecessary structural distortion with regard to initial positions (observe node Y). The parameter ℓ controls the lag of the monotonic field; if ℓ is too small, the frequent updates can lead to instabilities, and values that are too large can cause slow convergence. A typical energy profile during optimization is shown in Figure 5.3d, where the sharp changes in the total energy correspond to the updates of the monotonic field. We encourage the layout to be as similar as possible to

Algorithm 1: Layout with anisotropic radial constraints

Input: Graph $G = \{V, E, W\}$, maximum number of iterations $k \in \mathbb{N}$, depth field lag ℓ, step size α, weighing factor w_ρ
Output: Positions $\mathbf{X} = \{\mathbf{x}_i : 1 \leq i \leq n\}$ for all $v_i \in V$
$n \leftarrow |V|$
$\mathbf{X}_0 \leftarrow$ initialize node positions using MDS ; `/* (Section 5.2.2) */`
$\mathbf{c} \in \mathbb{R}^n \leftarrow$ compute graph centrality values for $v_i \in V$
$j \leftarrow -1$; `/* index to keep track of field updates */`
for $t = 1, \ldots, k$ **do**
  if $t \mod \ell = 0$ **then**
    $j \leftarrow j + 1$
    $\mathbf{X}_j \leftarrow \mathbf{X}_t$
    $M_{\mathbf{X}_j,\mathbf{c}}(\mathbf{X}_t) \leftarrow$ compute monotonic field ; `/* (Section 5.2.3) */`
  end
  $\mathbf{X}_{t+1} \leftarrow \mathbf{X}_t - \alpha\Big(\nabla\sigma(\mathbf{X}_t) + w_\rho \times 2\big(M_{\mathbf{X}_j,\mathbf{c}}(\mathbf{X}_t) - \mathbf{c}\big) \odot \nabla M_{\mathbf{X}_j,\mathbf{c}}(\mathbf{X}_t)\Big)$; `/* gradient update step (Section 5.3.1) */`
end

the MDS layout by initializing the node positions as determined by an *unmodified* MDS objective [39]. The entire process, as summarized in Algorithm 1, iterates until updates no longer result in significant changes to node positions.

The computational complexity of a single iteration is $\mathcal{O}(n^3)$ due to the step of computing the monotonic field, which involves interpolation using thin plate spline. However, we update the field only once every ℓ iterations, which leads to a complexity of $\mathcal{O}(n^2)$ (the same as MDS) for a large majority of iterations.

5.3.2 Visualization

In this layout, nodes are constrained to lie on level sets of centrality, which are *general* closed curves, rather than circles, and the shapes of these curves depend on the structure of the graph. Therefore, we can improve the interpretability of the layout and reduce cognitive load for the user by providing additional cues for shapes of these curves. We provide cues in the form of faded renderings of centrality contours (isolines on the monotonic field) and a monotonic field colormap in the background. The radial monotonicity described in Section 5.3.1 ensures that the contours are nested curves that enclose a *common* maxima (at origin); leading to a bijective mapping between contours and centrality values, and pushing nodes to lie on the *unique* contour that corresponds to their centrality. In

this chapter, we normalize node centrality to fall between 0 and 1, and show 10 contour curves that evenly span this range. We also use node size as an extra encoding channel for centrality —in addition to location—to further highlight the order structure. We can, of course, use the size channel to encode centrality even with the standard MDS layout; however, that approach can lead to the issue of conflicting centrality cues from size and location channels (see Figure 5.1a).

5.4 Results

In this section we demonstrate anisotropic radial layout using three real world data sets.

5.4.1 Zachary's Karate Club

The Zachary's karate club graph is a well-known data set that is a social network of friendships in a karate club at a US university, as recorded during a study [122]. This graph contains 34 nodes, each representing an individual, and 78 unweighted edges that represent a friendship between the associated individuals (Figure 5.1). During the period of observation, a conflict between two key members, identified as the "administrator" and "instructor," leads to a split in the club, giving it an interesting two-cluster structure. In Figure 5.1, nodes representing members who are part of the instructor's and administrator's groups are drawn in green and blue, respectively.

Figure 5.1 shows three different visualizations of the karate club network: MDS, radial layout (from [16]), and anisotropic radial layout (ARL). We can make a few observations from the visualizations. Although MDS does a good job of preserving the two clusters, it does not unambiguously convey centrality. On the other hand, radial layout clearly showcases the centrality at the expense of dispersing the clusters by distorting distances among their nodes, thereby obscuring their internal structure. We see that ARL is able to largely preserve the structure seen in MDS with clearly distinguishable clusters, and also clearly convey the centrality information. Although radial layout pushes the instructor's group far away due to low betweenness centrality, ARL lets them remain close by *bringing in* the outermost contour toward to the group instead. Similarly, the administrator is also allowed to remain closer to their group by the protrusion of the inner contours, which

enclose the most central nodes, toward the administrator.

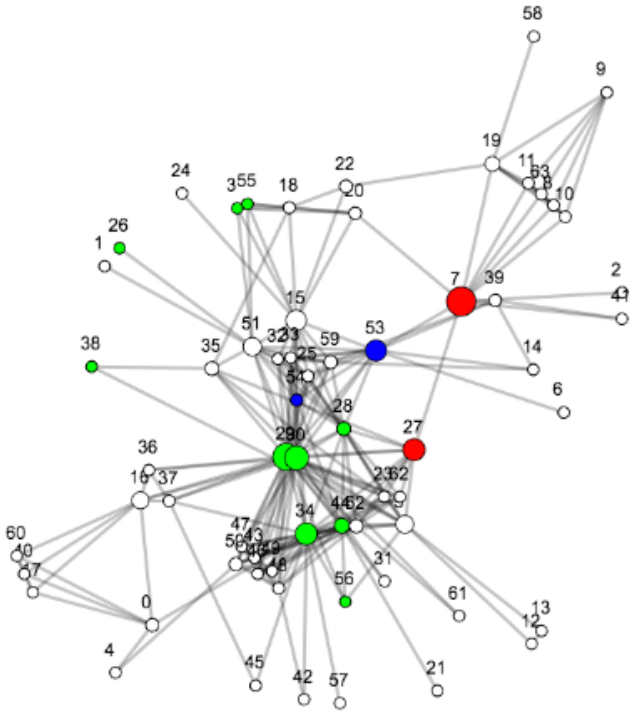
5.4.2 Terrorist Network From 2004 Madrid Train Bombing

Figure 5.4 and Figure 5.5 shows visualizations of a network of individuals connected to the bombing of trains in Madrid on March 11, 2004. These data were originally compiled by Rodriguez [92] from newspaper articles that reported on the subsequent police investigation. Sixty-four nodes represent suspects and their relatives, and 243 edges have weights ranging from 1 to 4, which represent an aggregated strength of connection based on various parameters such as contact, kinship, ties to Al Qaeda, etc. [44]. In Figure 5.4, (as well as Figures 5.5-5.7), distances between nodes are related inversely to edge weights. In the visualization, we identify nodes using numbers to avoid text clutter; however, we include a mapping to the names of individuals represented by the nodes in the Appendix.

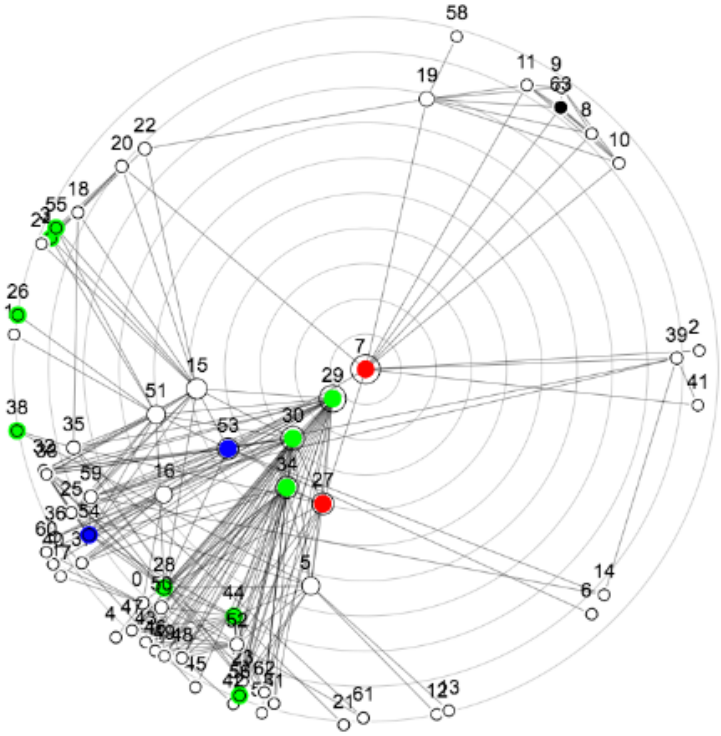
Rodriguez [92] identifies several key suspects as follows: ring leaders (marked in blue in Figure 5.4), members of a field operating group, who were closely involved with the actual carrying out of the attack (green); intermediaries (red); and suspects with local roots, ties to foreign Al Queda, and those who supplied explosives. We see that ARL (Figure 5.5) is able to better preserve the structure and cohesiveness of the core members of the field operating group in comparison to the radial layout (Figure 5.4b). Critically, a key mastermind in this event, despite having a low centrality (due to communicating often through an intermediary), is allowed to be close to the center in the ARL. This arrangement, which is possible due to the ability of centrality contours to adapt to the circumstance, preserves the close association between the masterminds that is lost in the radial layout. We also see that the flexibility of contours in ARL preserves the locality of various groups, which allows us to see the role of intermediaries with high centrality in acting as a bridge between various groups.

5.4.3 Coappearance Network for Characters in *Les Miserables*

The third data set is a graph of character associations in the famous French novel *Les Miserables* [55]. This graph consists of 77 nodes, each representing a character in the novel, and 254 weighted edges where the weights represent the number of chapters that feature both characters associated with an edge.



(a)



(b)

Figure 5.4. Network of terrorists and affiliates connected to the 2004 Madrid train bombing using (a) MDS and (b) radial layout.

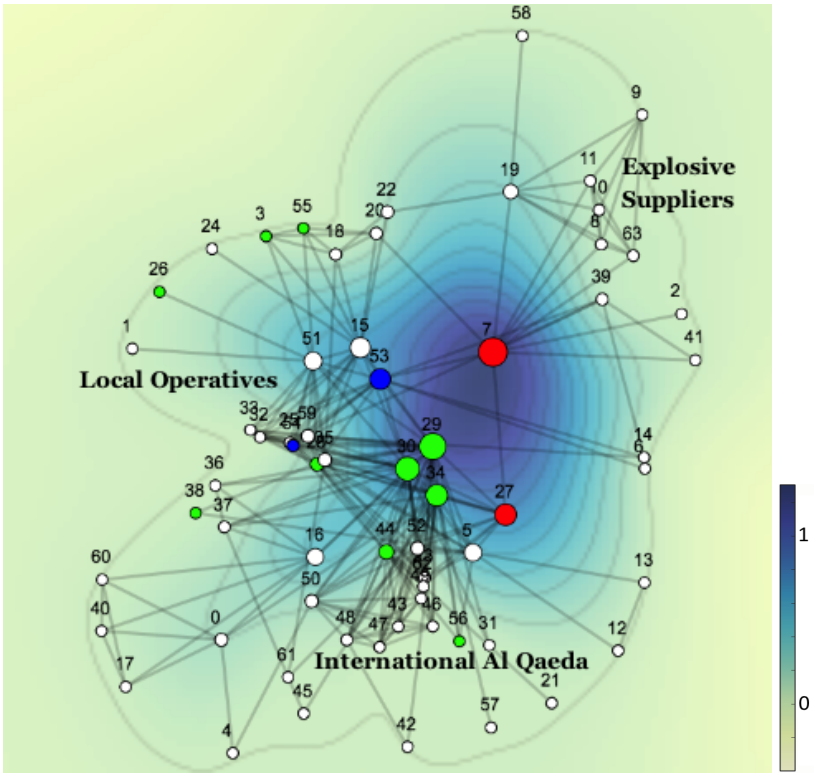
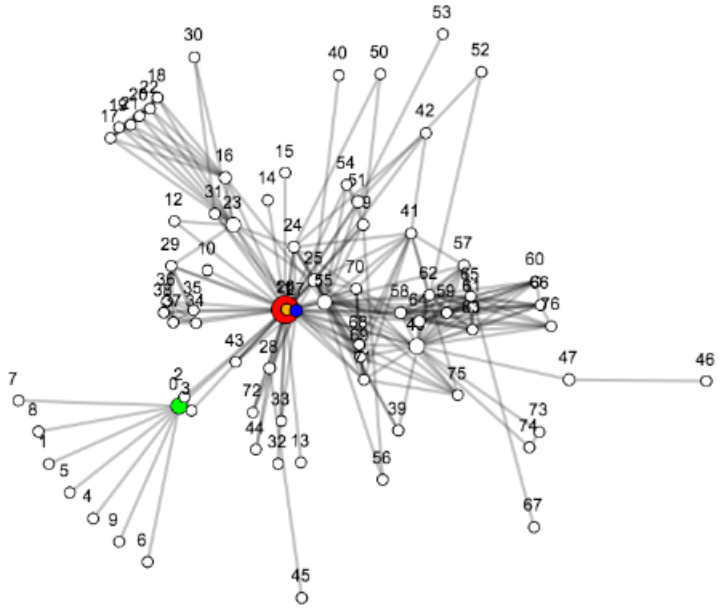
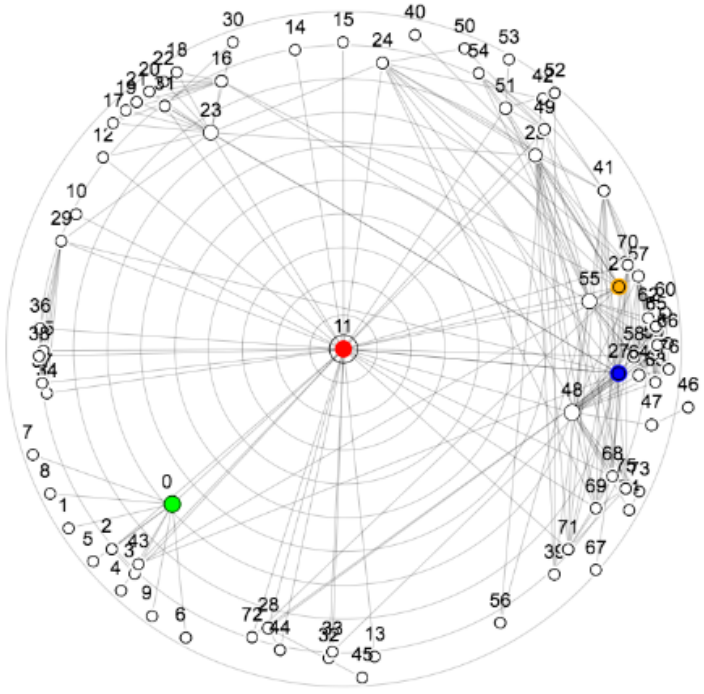


Figure 5.5. Network of terrorists and affiliates connected to the 2004 Madrid train bombing using anisotropic radial layout.



(a)



(b)

Figure 5.6. Coappearance network for characters in the novel *Les Miserables* using (a) MDS and (b) radial layout.

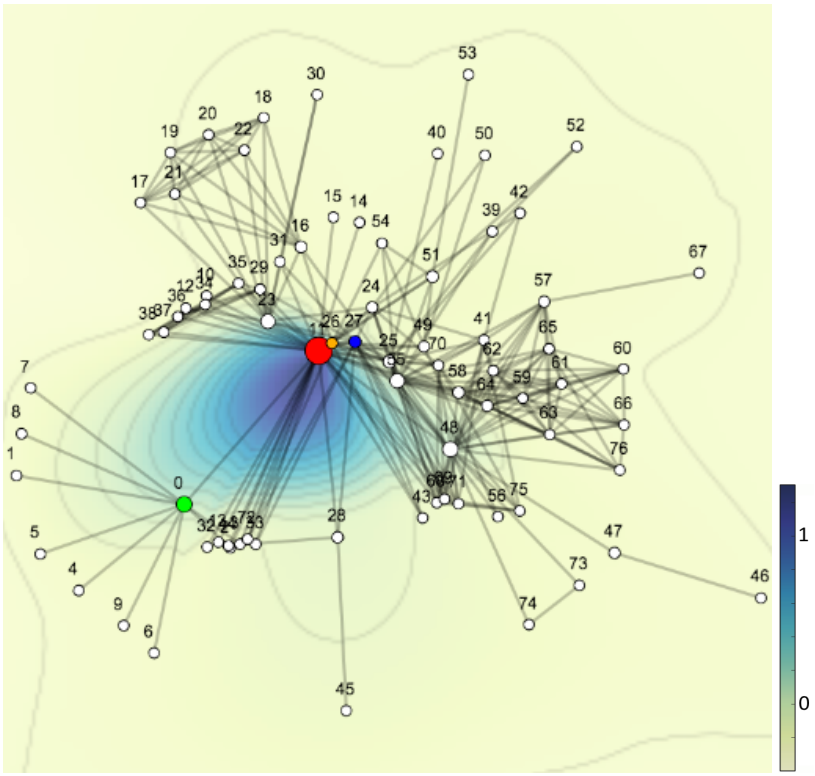


Figure 5.7. Coappearance network for characters in the novel *Les Miserables* using anisotropic radial layout.

We see the that the main protagonist *Valjean* (marked in red) is placed prominently in all three visualizations (Figure 5.6 and Figure 5.7). However, other key characters in the plot such as *Inspector Javert* (blue) and *Cosett* (orange), who do not appear often with characters other than the protagonist (and thus have low betweenness centrality), are treated differently. Although the radial layout relegates them to the periphery (far from Valjean) (Figure 5.6b), MDS (Figure 5.6a) paints a conflicting picture with regard to their centrality, e.g., Cosett's node almost overlaps with Valjean despite its low centrality. In contrast, the proposed ARL (Figure 5.7) is able to coherently convey the low centrality of Inspector Javert and Cosett, as well as their closeness to Valjean. The above issue of distance distortion appears to be a frequent occurrence in the radial layout due to the many characters who have a low centrality value, causing them to end up being packed in the outer periphery. A case of contrast is that of the character *Bishop Myriel* (green), who despite being associated with several characters, is seen with Valjean only once.

5.5 Discussion

This chapter describes an energy-based layout algorithm for graphs, called *anisotropic radial layout*, which conveys structural centrality using *anisotropic*, radial constraints, that also preserve approximate distances (or structure) in the graph. In contrast to existing methods for conveying node centrality that employ an *isotropic* centrality field [10, 16], the proposed method determines an *anisotropic* centrality field on which to project nodes. Although the energy minimization strategy described in this chapter allows the solution to deviate from constraints, one can enforce hard constraints by adding a postprocessing step that projects nodes onto the closest position on their associated isocontour.

The key implication of the anisotropic centrality field in our method is that more central nodes are allowed to be placed further from origin than less central nodes—without an energy penalty—if they do not lie on a common ray, which aids our objective of achieving a better balance between visual representations of centrality and structure than possible with existing methods. Our objective differs from other prior works that use centrality or continuous fields to visualize the structure of dense graphs [109, 110].

CHAPTER 6

VISUALIZING HIGH-DIMENSIONAL DATA USING ORDER STATISTICS

Portions of this chapter have been reproduced with permission from John Wiley and Sons, and based on material that is to appear in an article titled "Visualizing High-Dimensional Data using Order Statistics" by Raj, M and Whitaker, R. in the Computer Graphics Forum Volume 37(2018), Number 3.

6.1 Introduction

Multidimentional data appear frequently in a wide range of domains and applications. For example, data from domains such as healthcare, engineering, and social sciences often contain a large number of dimensions [59]. The various dimensions in such data can contain either numerical or categorical values. Multidimensional data can also have complex structures, for example, the data can be multimodal with several clusters or lie on a lower dimensional manifold in a high-dimensional space. A wide range of visualization methods has been developed to help visualize and understand such complex, high-dimensional data sets [64].

Among the various methods for analyzing high-dimensional data, dimensionality reduction methods that project data onto lower dimensional spaces are often useful for getting a quick and general overview of the data. These methods include various linear and nonlinear methods such as principal component analysis (PCA), multidimensional scaling (MDS), and t-distributed stochastic neighbor embedding (t-SNE). The objective of these methods is often to convey the structure of data by preserving approximate pairwise distances from the original or intrinsic space in the lower dimensional embedding space. Methods such as PCA and MDS can be formulated to work with *only* inner product information, which is useful for visualizing data in kernel spaces, which may lack an explicit vector representation [95]. Dimensionality reduction techniques are also used in

conjunction with other visualization methods [69, 93].

Despite the usefulness of dimensionality reduction methods for visualizing multidimensional data, there are a few critical limitations associated with those methods. The PCA and related subspace-based approaches may not be suitable if the data are not well approximated by a linear subspace. Although MDS and nonlinear methods such as t-SNE are able to highlight geometric relationships, even in the presence of a nonlinear structure, they are susceptible to misrepresenting the statistical structure in the data. For example, points that are rare and on the outer periphery of a distribution in a high-dimensional space may be projected close to a more typical point near the center of the distribution. Such instances are common, and unsurprising if we consider that the objective of those methods is typically to preserve the relative distances between points with no mechanism to correctly convey how central or typical points are in a data set or distribution. Although the focus on preserving relative distances to reveal high-level structure can be useful, doing so at the expense of centrality information can hinder a true understanding of the data set as a whole, and be particularly detrimental for the purpose of analyzing outliers [116].

In this chapter, we propose a novel method to project multidimensional data onto a lower dimensional space while preserving order structure as well as relative distances in the data. The focus of this work is different from prior work in robust multidimensional scaling that aims to mitigate the undesirable effects resulting from inconsistencies in data (pairwise distances in the original space) [35, 100]. In contrast, the proposed method is relevant even when there are no inconsistencies in the data. The proposed method does share the ideology with a family of methods from the domain of graph drawing, where the goal is to determine node positions in a drawing that simultaneously conveys graph-theoretic, internode distances (distances along edges) as well as node centrality or importance based on complementary, graph-theoretical measures [10, 15, 16, 91]. The internode distances in the drawing approximate the graph-theoretical distances under the constraint that the distance of each node from the drawing's geometric center be proportional to its graph centrality value.

In this work, we aim to preserve *order statistics* of the data in the original space by ensuring that less central or outlying points do not end up appearing to be more central in low-dimensional embeddings, or vice versa. We also want to preserve relative pairwise

distances in the data as much as possible. An overview of the proposed projection method for satisfying the above objectives is as follows. We first quantify the centrality of each member in the original space by employing data-depth methods (see Section 6.2.1). Next, we design a penalty term to be added to the MDS optimization objective that penalizes low-dimensional embeddings, where along any ray traveling away from the position of the most central member, *less central* points are situated further from the center than more central points. Although we demonstrate the proposed method with the help of the MDS objective, the general approach can be used to similar effect with any other dimensionality reduction method that involves iterative optimization.

The goal of visualizations, in general, is to highlight features of interest in the data. These feature often include summary statistics such as most central or typical member (also known as the median), least central or outlier members, as well as the shape and the spread of the bulk of data. In case of one-dimensional (1D) and two-dimensional (2D) data, visualizations such as the Tukey boxplot [107] and the bivariate bagplot [93] convey a visual summary of the data by displaying summary statistics. In this chapter, we exploit the coherent order structure in the embedding space afforded by the proposed projection method to develop visualization strategies, along the lines of the bivariate bagplot, for multidimensional data (i.e., where $d > 2$).

The main contributions of this chapter are:

- A novel method for projecting multidimensional data using order statistics called *order aware projection* (OAP).
- Two visualization strategies based on the proposed projection method, namely, *field overlay plot* and *projection bagplot*.
- An interactive prototype tool to explore data.
- A demonstration of the effectiveness of the method with four real data sets.

The rest of the chapter is organized as follows. Section 6.2 provides an overview of the technical background related to the proposed methods. Section 6.3 presents a description of the proposed dimensionality reduction method and visualization strategy. We demon-

strate proposed methods using real data in Section 6.4, which is followed by a general discussion in Section 6.5.

6.2 Background

Here we provide an overview of necessary technical background and related work.

6.2.1 Order Statistics and Data Depth

Order statistics for a data set are members from the data set placed in an ascending order based on some criteria. For our purpose, we are interested in *center-outward* order statistics that help quantify how central or outlying a member is with respect to a data set. In the case of 1D numeric data, sorting numbers based on distance from the median provides an easy way to obtain order statistics. When the data are multidimensional, a family of methods from descriptive statistics known as *data depth* can be used to quantify center-outwardness. Data depth methods exhibit several useful properties, which make it an attractive basis for analyzing data. These properties include robustness, maximum at center, monotonicity, and zero at infinity [124].

Data-depth methods have been proposed for tackling several types of multidimensional and multivariate data, for example, high-dimensional points [107], functions [66], sets [115], multivariate curves [73], and paths on a graph [90]. In this chapter, we use different formulations of data depth based on the type of data. We use half-space depth for numerical multidimensional data with relatively few dimensions (Section 6.4.2). We use functional depth for dealing with higher dimensional data because it can be efficiently computed for such data (Section 6.4.1). Finally, we use set depth for categorical data sets (Sections 6.4.3 and 6.4.4).

A brief overview of half-space depth, functional depth, and set depth follows. Half-space depth of any point $\mathbf{x} \in \mathbb{R}^d$ with respect to a set of points $X \in \mathbb{R}^d$ is defined as the smallest number of data points from S that can be contained in a closed half space also containing $\mathbf{x}$ [31, 107], which can be stated as

$$d_{\text{halfspace}}(\mathbf{x}|X) = \min_{\mathbf{a} \in \mathbb{R}^d \setminus 0} |\{\mathbf{p} \in X \, : \langle \mathbf{a}, \mathbf{p} \rangle \geq \langle \mathbf{a}, \mathbf{x} \rangle\}|. \tag{6.1}$$

Functional depth of any function $g(t)$ with respect to a set of functions $F = \{f_i(t) : 1 \leq i \leq n\}$, $f_i : \mathcal{D} \to \mathcal{R}$, where $\mathcal{D}$ and $\mathcal{R}$ are intervals in $\mathbb{R}$, is given by the probability

of $g(t)$ being contained in a functional band, where functional band is the region between the min/max envelope formed by a set of j randomly chosen functions $\{f_1(t), \ldots, f_j(t)\} \in F$ [66], which can be stated as

$$d_{\text{functional}}\left(g(t)|F\right) = \text{Prob}\left(g(t) \subset \text{fB}\left[\{f_1(t), \cdots, f_j(t)\}\right]\right), \tag{6.2}$$

where $\text{fB}[\,\cdot\,]$ denotes the functional band. A function $g(t)$ is contained in the functional band formed by $\{f_1(t), \ldots, f_j(t)\}$ if it satisfies the following:

$$\begin{aligned} g(t) \subset\ & \text{fB}\left[\{f_1(t), \cdots, f_j(t)\}\right] \quad \text{iff} \\ & \min\left(f_1(t), \cdots, f_j(t)\right) \leq g(t) \leq \max\left(f_1(t), \cdots, f_j(t)\right) \quad \forall t. \end{aligned} \tag{6.3}$$

Set depth of any set s with respect to a set of sets $S = \{s_i : 1 \leq i \leq n\}$ is given by the probability of s being contained in a *set band*, where set band is the set bounded by the union and intersection of j randomly chosen sets $\{s_1, \ldots, s_j\} \in S$ [115], which can be stated as

$$d_{\text{set}}\left(s|S\right) = \text{Prob}\left(s \subset \text{sB}\left[\{s_1, \ldots, s_j\}\right]\right), \tag{6.4}$$

where $\text{sB}[\,\cdot\,]$ denotes the *set band*. A set s is contained in the set band formed by $\{s_1, \ldots, s_j\}$ if it satisfies the following:

$$s \subset \text{sB}\left[\{s_1, \ldots, s_j\}\right] \quad \text{iff} \quad \bigcup_{k=1}^{j} s_k \subset s \subset \bigcap_{k=1}^{j} s_k.$$

Function depth and set depth are stable with respect to choice of j where $2 \leq j \leq n$ [66, 115].

6.2.2 Data-Depth-Based Visualizations

A common area of application for data depth methods is ensemble visualization where the order statistics obtained using data depth are used to design summary visualizations for ensembles of various kinds of data. The perhaps most well-known example is the Tukey boxplot [107]. Other depth-based visualizations have been proposed for bivariate data [93], high-dimensional data [47], ensembles of functions [102], surfaces [40], sets or isocontours [115], curves [67, 73], and paths on graphs [90]. Our work relates closely to the visualizations for multidimensional data, particularly the bivariate bagplot [93] and the high-density region (HDR) boxplot [48] (see Figure 6.1).

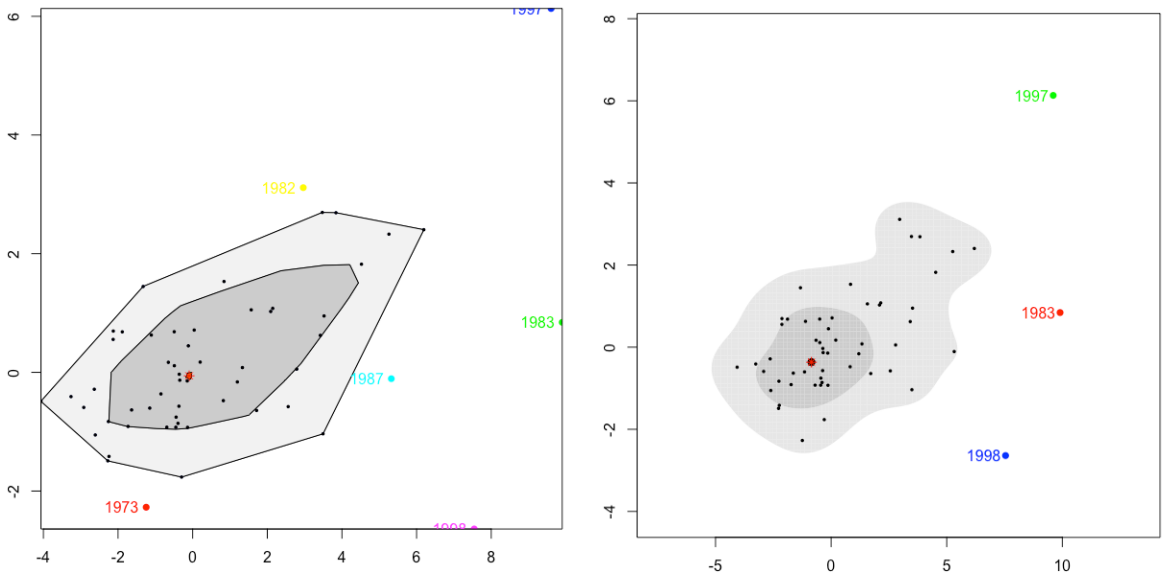


Figure 6.1. Existing visualizations for multivariate data. (a) Bivariate bagplot and (b) high-density region (HDR) boxplot visualizations of El Niño data set (12-dimensional temperature data for each year from 1951 to 2007) generated using R Rainbow package [99].

For 2D data, the bivariate bagplot (Figure 6.1a) is a visualization technique that highlights the median, spread, skewness, and outliers in the data. The first step for drawing a bagplot is to determine order statistics using half-space depth. This step is followed by drawing the inner and outer convex polygons or *bands*. The inner band highlights the most central half of the data as determined by the order statistics, and the outer band is constructed by inflating the inner band by a constant factor α. Points outside the outer band are considered to be outliers. The HDR boxplot uses bivariate kernel density estimation to identify regions of interest. The bivariate bagplot as well as the HDR boxplot use dimensionality reduction methods, typically PCA, for dealing with higher dimensional data ($d > 2$) by projecting the data to 2D as a preprocessing step [47, 48].

6.2.3 Multidimensional Scaling (MDS)

Since the proposed projection method uses the MDS objective function, here we give a brief overview of MDS and its usage in dimensionality reduction. MDS refers to a popular class of techniques for visualizing similarities between members of a data set. Given a collection of high-dimensional points $Y = [\mathbf{y}_1, \ldots, \mathbf{y}_n]^T \in \mathbb{R}^{n \times d}$, the goal of MDS is to

find a low-dimensional embedding $X = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^T \in \mathbb{R}^{n \times k}$, where $k < d$, such that the discrepancy between the pairwise distances in the original space $\mathbb{R}^d$ and corresponding distances in the embedding space $\mathbb{R}^k$ is minimal.

Although there are several variants of MDS, in this chapter we use a variant known as metric MDS with distance scaling; without loss of generality. Distance scaling makes this variant of MDS nonlinear with more emphasis on conveying smaller distances. The objective function of metric MDS is also known as *stress*, and after incorporating distance scaling, it can be written as follows [13, 70]:

$$\sigma(X) = \sum_{i<j} w_{ij} \big(\delta_{ij} - d(\mathbf{x}_i, \mathbf{x}_j)\big)^2, \tag{6.5}$$

where $\delta_{ij} = ||\mathbf{y}_i - \mathbf{y}_j||_2$, $d(\mathbf{x}_i, \mathbf{x}_j) = ||\mathbf{x}_i - \mathbf{x}_j||_2$, and $w_{ij} = \delta_{ij}^{-2}$. The gradient of the above MDS objective can be written as follows [13]:

$$\nabla\sigma(X) = 2VX - B(X)X \tag{6.6}$$

where matrices $V = (v_{ij})$ and $B = (b_{ij})$, with $1 \leq i, j \leq n$, can be represented as

$$v_{ij} = \begin{cases} -w_{ij} & \text{for } i \neq j \\ \sum_{j=1, j\neq i}^{n} w_{ij} & \text{for } i = j \end{cases} \qquad b_{ii} = -\sum_{j=1, j\neq i}^{n} b_{ij}$$

$$b_{ij} = \begin{cases} -\frac{w_{ij}\delta_{ij}}{d(\mathbf{x}_i, \mathbf{x}_j)} & \text{for } i \neq j \text{ and } d(\mathbf{x}_i, \mathbf{x}_j) \neq 0 \\ 0 & \text{for } i \neq j \text{ and } d(\mathbf{x}_i, \mathbf{x}_j) = 0. \end{cases}$$

6.2.4 Monotone Regression Along One Variable for Multivariate Data

The proposed projection method also involves computing a continuous and smooth, radially decreasing approximation of depth of members in a 2D embedding. We call this approximation the *monotonic field* (Figure 6.2c). Note that data-depth values are computed for points in the original space and not after they are projected onto the embedding space. To construct a monotonic depth field from a sparse set of depth values arranged in a 2D embedding plane, we start by computing a smooth *interpolated field* (Figure 6.2b) of depth values using the thin plate spline technique [12]. In what follows, we briefly describe our approach for computing the monotonic field by radially monotonizing the interpolation

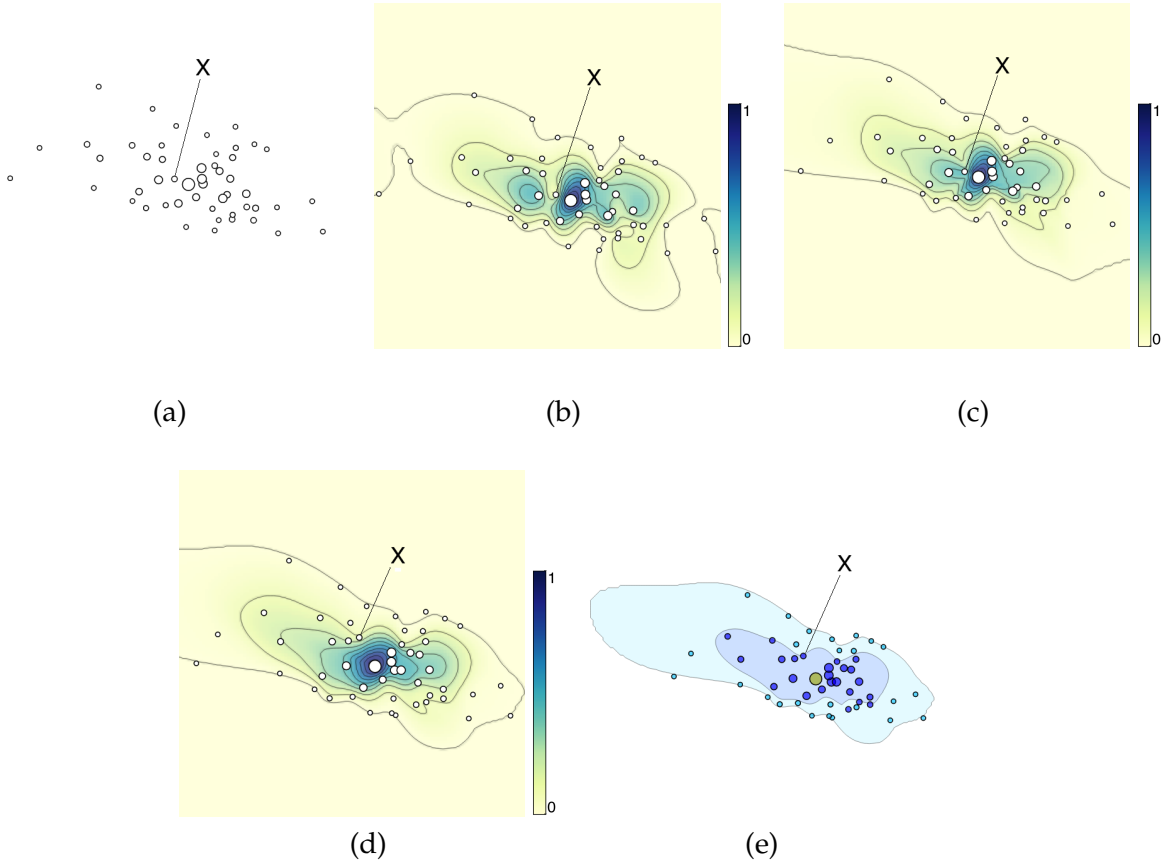


Figure 6.2. Various stages during the proposed methods. a) Points from an anisotropic, 3D normal distribution projected on a 2D plane using MDS. Circle sizes indicate half-space depth of points in the original 3D space. b) The initial interpolated field in the background of the MDS projection. c) The initial monotonic field in background obtained from initial interpolated field. d) Field overlay plot using order aware projection (OAP) after optimization is complete. The final monotonic field shown in the background. e) Projection bagplot visualization. Median is shown in yellow. Deep blue indicates 50% band and light blue indicates 100% band.

field. This approach is adapted from a technique for performing monotone regression for multivariate data [26].

The process of computing radially monotonic approximations of a smooth 2D field depends on a method to find monotonic approximations of univariate data. Given a smooth 1D function $m(t) : [0, 1] \rightarrow \mathbb{R}$, the following two steps provide a monotonic approximation, $\hat{m}_A(t)$, that is smooth and first-order asymptotically equivalent to $m(t)$ [25]:

- Step 1 (monotonization): Sample input function at regular intervals, compute a density estimate of the samples, and then compute a cdf of the density estimate to arrive

at the inverse of the monotonic approximation.

$$\hat{m}_A^{-1}(z) = \frac{1}{N\omega} \sum_{i=1}^{N} \int_z^{\infty} K\left(\frac{m\left(\frac{i}{N}\right) - u}{\omega} \right) du, \tag{6.7}$$

where N controls the sampling resolution, and K is a smooth, symmetric kernel with bandwidth ω.

- Step 2 (inversion): Calculate the inverse of $\hat{m}_A^{-1}$, which is the desired monotonically decreasing approximation of the 1D function $m(t)$.

For computing a radially monotonic approximation of the interpolated field, we proceed by resampling the field onto a polar grid centered at the median (deepest member as per data depth computed in the original space). We then treat values on the field along each of the evenly spaced angular coordinates as 1D functions, which can then be monotonized using the procedure described above. On monotonizing those 1D functions along each direction, we arrive at the monotonic field, which we then resample back to the Cartesian coordinates. The resolution of the polar grid, both radial and angular, determines the quality (smoothness) of monotonic field (we use 360 radial divisions for results in this chapter). The smoothness of the interpolation field is preserved through this process, meaning that field values along adjacent directions vary smoothly and remain coherent, due to the properties of the monotonization process (except at origin due to the intermediate polar coordinate representation) [26].

6.3 Method

Here we describe the proposed projection method, which preserves the centrality structures using order statistics (Section 6.3.1) and visualization strategies, which use the resultant embedding (Section 6.3.2).

6.3.1 Projecting Multidimensional Data Using Order Statistics (Order Aware Projection)

The high-level goal of our projection method is to preserve both the relative distances between individual members as well as the order statistics from the original multidimensional space when computing a lower dimensional embedding. To achieve this, we design an objective function that comprises two terms. The first term is identical to the MDS

stress (Section 6.2.3), which penalizes discord in pairwise distances between intrinsic space and the embedding. The second term levies an energy penalty for discord in the center-outward order statistics. Since the order statistics are determined using data depth, we call this term the *depth penalty*.

The data-depth values computed in the original space (Section 6.2.1) and the monotonic field computed in the embedding space (Section 5.2.3) are used to quantify the discord in order structure between the original and embedding spaces. The isocontours of the monotonic field mimic the monotonic, center-outward decrease of depth values in the original space. The depth penalty at each point is proportional to the difference between its depth value in the original space and the depth values of the monotonic field sampled at the location of its projection in the embedding space, and can be expressed as follows:

$$p(X) \propto \big(M_{X,\mathbf{h}}(X) - \mathbf{h}\big)^2, \tag{6.8}$$

where $X = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^T \in \mathbb{R}^{n\times k}$, $\mathbf{h} \in \mathbb{R}^{n\times 1}$ contains depth values associated with X computed in the original space $\mathbb{R}^d$, where $k < d$, and $M_{X,\mathbf{h}}(X) \in \mathbb{R}^{n\times 1}$ denotes values of the 2D monotonic field at positions in X. The mention of X and $\mathbf{h}$ in the subscript indicates their use in the construction of the monotonic field, and X in parenthesis indicates positions where field values are sampled. If the interpolated field (Section 6.2.4) is also itself radially monotonic, the value of depth penalty term approaches zero. The complete objective function, which includes both MDS stress and depth penalty, can be stated as follows:

$$\gamma\big(X\big) = \underbrace{\sigma(X)}_{\text{MDS stress}} + \; w_p \; p(X), \tag{6.9}$$

where w_p is a constant of proportionality controlling the relative importance of the depth penalty with respect to MDS stress. The gradient of the above objective can be derived as

$$\nabla\gamma\big(X\big) = \nabla\sigma(X) + w_p \times \underbrace{2\big(M_{X,\mathbf{h}}(X) - \mathbf{h}\big) \odot \nabla M_{X,\mathbf{h}}(X)}_{\nabla p(X)}, \tag{6.10}$$

where $\odot$ denotes the element-wise product. We perform an optimization of the above objective using gradient descent until X converges or the maximum number of allowed iterations is reached. The proposed projection method is summarized in Algorithm 2.

Optimization of the above objective requires a few considerations in practice. First, computing the gradient of the monotonic field M at positions X is nontrivial due to the

Algorithm 2: Order Aware Projection (OAP)

Input: $Y = [\mathbf{y}_1, \ldots, \mathbf{y}_n]^T \in \mathbb{R}^{n \times d}$, maximum number of iterations $i_{\max} \in \mathbb{N}$, depth field lag ℓ, step size τ, depth weight w_p

Output: Positions $X = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^T \in \mathbb{R}^{n \times k}$ where $k < d$

$X_0 \leftarrow$ compute initial embedding using MDS ; `/* (6.2.3) */`

$\mathbf{h} \in \mathbb{R}^{n \times 1} \leftarrow$ compute order statistics for $\{\mathbf{y}_1, \ldots, \mathbf{y}_n\} \in \mathbb{R}^d$

$j \leftarrow -1$; `/* counter for field updates */`

for $i = 1, \ldots, i_{max}$ **do**

  if $i \mod \ell = 0$ **then**

    $j \leftarrow j + 1$

    $X_j \leftarrow X_i$

    $M_{X_j,\mathbf{h}}(X_i) \leftarrow$ compute monotonic field ; `/* (6.2.4) */`

  end

  $X_{i+1} \leftarrow X_i - \tau\Big(\nabla\sigma(X_i) + w_p \times 2\big(M_{X_j,\mathbf{h}}(X_i) - \mathbf{h}\big) \odot \nabla M_{X_j,\mathbf{h}}(X_i)\Big)$; `/* perform gradient update (6.3.1) */`

end

dependence of M itself on X. We deal with this issue by letting the field lag, which means to recompute M only after a fixed number of iterations, ℓ, have passed since the previous update and treat it as a constant during all intervening iterations (see Figure 6.3). If field M is held constant ($\ell = \infty$), convergence at a local minima can be guaranteed due to properties of gradient descent. On allowing the field to lag suitably ($1 \leq \ell < \infty$, see Section 6.5), in practice we observe convergence to a lower energy state, although a theoretical guarantee remains a topic for future work. This approach is also used for minimizing similar energies for computing graph layouts [91]. Second, for stability with regard to the median, the proposed method relies on known robustness of data-depth methods in situations of data contamination [102, 107, 115]. In cases where there are multiple members identified as a median in the original space, we choose the member with the highest depth value among those in the embedding space. This approach helps reduce overall energy if different medians are projected far apart, as is often observed with categorical data (Sections 6.4.3 and 6.4.4).

In the case of multimodal data where class membership information is known in advance, we construct a separate monotonic field for each class centered at the median of that class, which leads to separate, exclusive depth penalty terms that apply only to the members of the associated class (Sections 6.4.2 and 6.4.4). The MDS term is the same as in

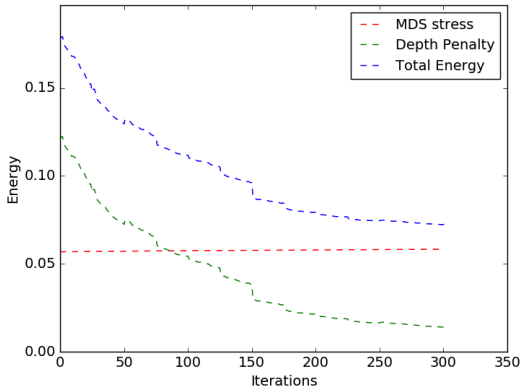


Figure 6.3. The typical profile for MDS stress and depth penalty during the optimization process. MDS stress increases slightly. The depth penalty undergoes sharp drops periodically at iterations with monotonic field updates.

the general case (which assumes a unimodal distribution) and considers pairwise distance relationships across the entire data set. This approach preserves the order structure within each class while allowing MDS forces to determine the relative placement of different classes.

6.3.2 Field Overlay and Projection Bagplot Visualizations

At the end of the optimization process, all members in the data set are aligned with their corresponding isocontours (whose isovalue matches member depth) on the underlying monotonic field. The shape of the isocontours depends on the data and can often provide useful insights into the structure of the data in the original space. Our first visualization strategy, called *field overlay plot*, is to present the OAP embedding overlaid on the associated monotonic field (Figure 6.2d). We show the monotonic field as a color heatmap with isocontour lines for 10 equidistant values spanning the range of depth values. This approach helps with the interpretability of the OAP embedding by highlighting the depth associated with each member as well as the regions/directions of fast and slow depth changes.

Due to the radially, monotonically decreasing property of the monotonic field, all isocontours divide the embedding space into inner and outer regions, which exclusively contain members with higher and and lower depth in the original space. We propose

the *projection bagplot* visualization (Figure 6.2e), which uses this arrangement of members in the embedding space to convey the median, inner, and outer bands analogous to those seen in the Tukey boxplot [107]. The depth value of the isocontour corresponding to the 50% band, $h_{50\%}$, is chosen to be the value of the member at 50th percentile by ranking the members' depth values. The 100% band is formed by inflating the 50% band by a constant factor α. We therefore have $h_{100\%} = h_{\text{median}} - \alpha \times (h_{\text{median}} - h_{50\%})$, where h_{median} is the depth value of the median (highest depth value by definition) and $\alpha = 1.5$ typically [102, 107]. We use higher and lower color saturation for indicating band/members in the 50% and 100% bands, respectively. For multimodal data with known class membership information, a separate set of 50% and 100% bands is computed and displayed for each class as demonstrated in later figures.

The projection bagplot visualization shares some similarities with both the bagplot and the HDR bagplot. The interpretation of the bands in the proposed method is similar to that for the bagplot, whereas the shape of bands is smooth and star shaped like in the HDR bagplot. Despite the similarities, the proposed method is notability different in its handling of multidimensional data projected to lower dimensions due to the emphasis on maintaining the center-outward order of members during the order-preserving projection process. Although glyph sizes or color can be modulated to convey order statistics, the reliance of bagplot and HDR bagplot visualization on existing dimensionality reduction techniques [48] can lead to a conflict between glyph size/color and location cues (e.g., members appearing as outliers due to smaller glyphs also appearing closer to the center). Furthermore, when displaying large data sets, available display space may place an upper bound on the glyph sizes, thereby restricting the usable range of glyph sizes.

6.4 Results

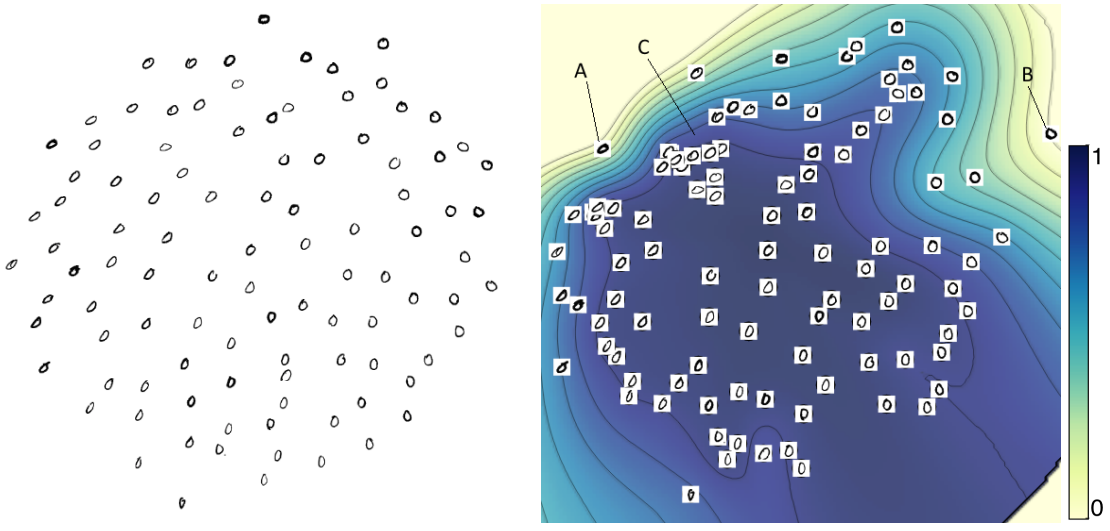
We now present some example visualizations of real data sets with existing and proposed methods.

6.4.1 MNIST Data

The MNIST data set is popular in the machine learning community and is comprised of thousands of samples of handwritten digits [56]. The samples are formatted as 28×28

pixel gray-scale images, resulting in each sample being comprised of 784 dimensions. Figure 6.4 shows two visualizations of a random subset of 100 samples of digit 0, although Figure 6.5 shows digits 0, 1, and 7 with 100 samples each. We consider each sample to be an instance of a 784-dimensional function and use functional depth to compute order statistics. In Figure 6.5, we use order statistics computed for each digit separately. These order statistics are used to obtain an OAP embedding, which is used to draw a field overlay plot (Section 6.3.2). In Figure 6.5, we use the proposed visualization strategy for multimodal data with a separate monotonic field for each digit (Section 6.3).

On comparing the MDS embedding Figure 6.4a and the proposed field overlay plot Figure 6.4b, we can make a few interesting observations. First, we notice that the underlying depth contours make it easy to spot outliers in the field overlay plot. Since the contours adapt to the data, we also notice different outlier characteristics, such as sharing some similarity with other members (see member A) or being more peculiar (see member B). We also notice that the proposed projection (OAP) presents more clique-like structures, often with similar members in a tighter cluster than in the MDS (see cliques around region C). The formation of cliques can be understood by considering that members in a relatively



(a) (b)

Figure 6.4. MNIST data sample visualizations: (a) MDS and (b) field overlay plot using order aware projection (OAP). Outliers and cliques appear more prominent in the field overlay plot.

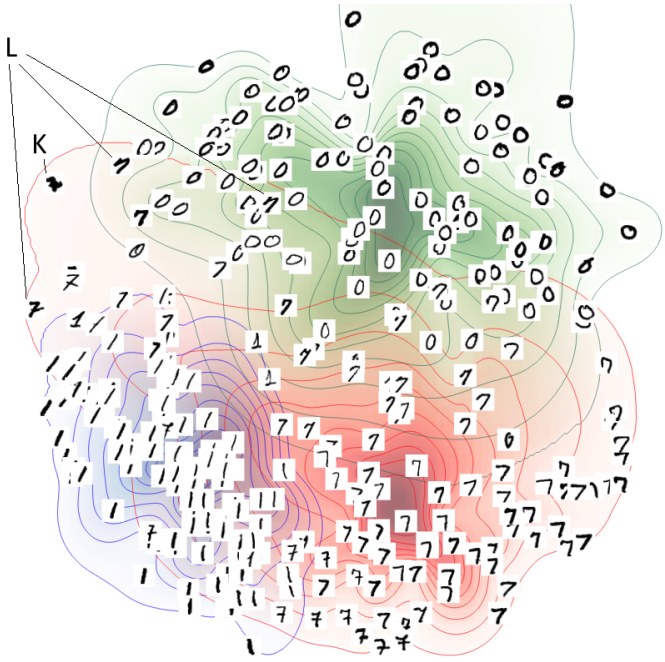


Figure 6.5. MNIST data sample visualization for multiple digits (0, 1, and 7) with field overlay plot using OAP. Monotonic fields corresponding to 0, 1, and 7 are shown using heatmaps and isocontour lines drawn in green, blue, and red, respectively. Higher saturation of colors in the heatmaps indicates a higher value of the monotonic field. Unusual members are apparent on tracing outermost isocontours.

local region of the original space would tend to have similar depth values and low pairwise distances, and would be encouraged to be placed similarly in the embedding by both depth and MDS energies. In Figure 6.5, we can observe the different monotonic fields for digits 0, 1, and 7. The higher overlap of digit 7 with other digits, particularly with digit 1, is immediately clear. Furthermore, on tracing the outermost isocontours of fields, we are immediately drawn toward outlying members that have been placed far from other members, or share similarities with other digits. For example, see instances marked by K and L, respectively, in Figure 6.5.

6.4.2 Iris Flower Data

We obtained the well-known Iris data set from the UCI machine learning repository [59]. The data set contains flower sepal and petal measurements from three related species of Iris flowers, and includes 50 instances of each species with four numeric measurements per instance. In Figure 6.6, we use the proposed visualization strategy for multimodal

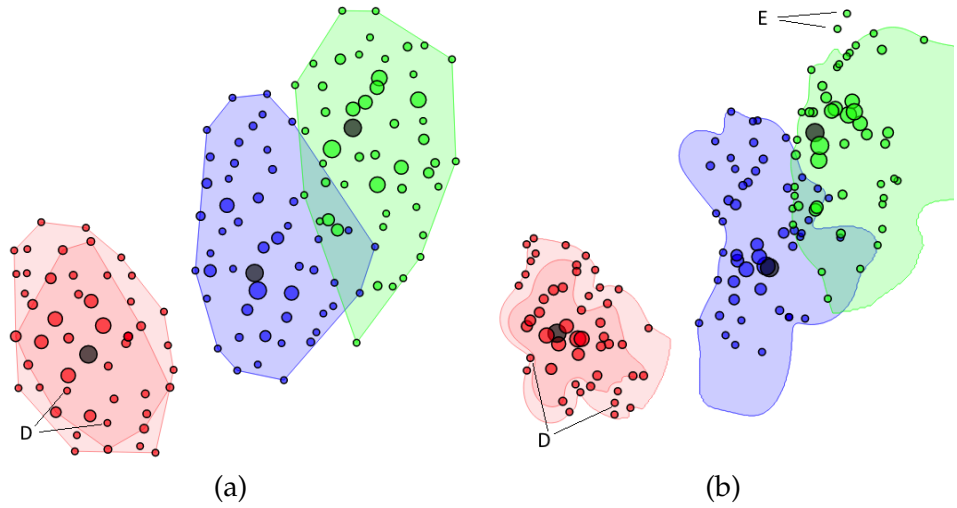


Figure 6.6. Iris flower data visualization: (a) bivariate bagplot using MDS and (b) projection bagplot using OAP. There are three species of flowers, each represented by a color, and each circle represents an individual flower. For the blue and green classes, 50% and 100% bands overlap due to a large proportion of members with identical, lowest value of depth. For the red class, only the projection bagplot conveys band associations of the flowers correctly.

data with a separate monotonic field for each of the three species classes (Section 6.3). The order statistics are computed using half-space depth for each class separately. The median of each class is colored dark gray, and the size of the circular glyphs encodes the depth of the members with respect to their respective classes.

Figure 6.6a shows a bivariate bagplot [48] and Figure 6.6b shows the proposed projection bagplot. In both figures, we immediately notice a difference in the structure of the classes based on the overlap of the 50% and 100% bands. In the red (Setosa) class, we observe a partial overlap as opposed to a full overlap in the blue (Versicolor) and green (Virginica) classes. This observation indicates a more even spread of members in the red class and more members at the class boundaries for the blue and green classes. We also see that the order structures within classes are preserved in the projection bagplot; along all outward directions from the median, the depth of the members falls monotonically. The preservation of centrality structures prevents cases as in region D where members in the 100% band are projected to fall inside the 50% bands in the embedding in the bivariate bagplot. Another interesting area is region E where two members are pushed out of the 100% band despite being of similar depth as other nearby points. Such layout of members

is due to the distance-preserving aspect of the proposed objective (Equation 6.9) trying to convey differences among members that are all on the boundary of the green class. Such cases as highlighted by the projection bagplot are good instances for further exploration.

6.4.3 Unidentified Flying Object (UFO) Encounters Data

We now look at a data set related to UFO encounters that was compiled by Winner from information available in the public domain [118]. The data set contains the following six attributes (one numeric and five categorical): year of sighting, location, presence/absence of physical effects, multimedia, extraterrestrial contact, and involvement of abduction. To understand the typical/atypical characteristics of recorded UFO encounters across the years, we exclude the year information, and include only the categorical dimensions in our analysis. The distances between members needed for the MDS term are obtained through the inner products computed using the "k0" kernel for categorical data [11]. We compute order statistics for categorical data by using set band depth (Sec 6.2.1) and treating each member as a set of its attribute values from all dimensions [74].

Figure 6.7 shows two visualizations for the UFO data set. An interesting feature of this

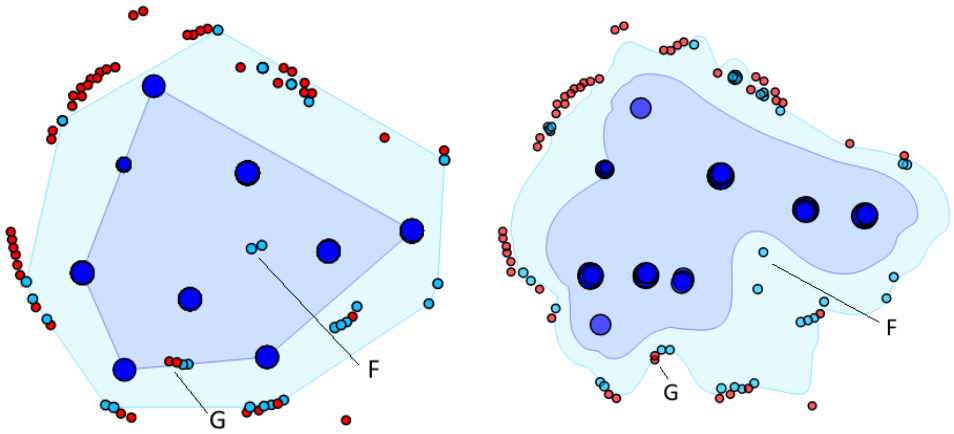


Figure 6.7. Unidentified flying object (UFO) encounters data visualizations: (a) bivariate bagplot using MDS and (b) projection bagplot using OAP. Each circle represents an encounter. Deep blue, light blue, and red circle colors indicate association to the 50% band, 100% band, and outliers. The projection bagplot is able to show correct band associations for members as opposed to the bivariate bagplot, which misplaces some encounter instances with respect to bands.

data set is the presence of several members with the highest depth value that are placed relatively far from each other. On inspection, we find that they are all sightings in the USA, which leads to the conclusion that a large number and variety of UFO sightings are recorded in the USA. Some sightings at other locations share many attributes of US sightings, but still cannot be representative of the data, as indicated by their low depth values, because of being at a different location (see region F). The projection bagplot (Figure 6.7b) is able to convey this well by adjusting the shape of the 50% band to exclude those points without a significant change in their positions, whereas the bivariate bagplot (Figure 6.7a) shows a contradiction where members supposed to be in the 100% band are seen within the 50% band. Another such contradiction is seen in region G, where outliers (shown in red) appear to be inside the 100% band.

6.4.4 Breast Cancer Data

Figure 6.8 displays our final data set, which consists of a collection of breast cancer patient attributes compiled at the University Medical Center at Ljubljana and made avail-

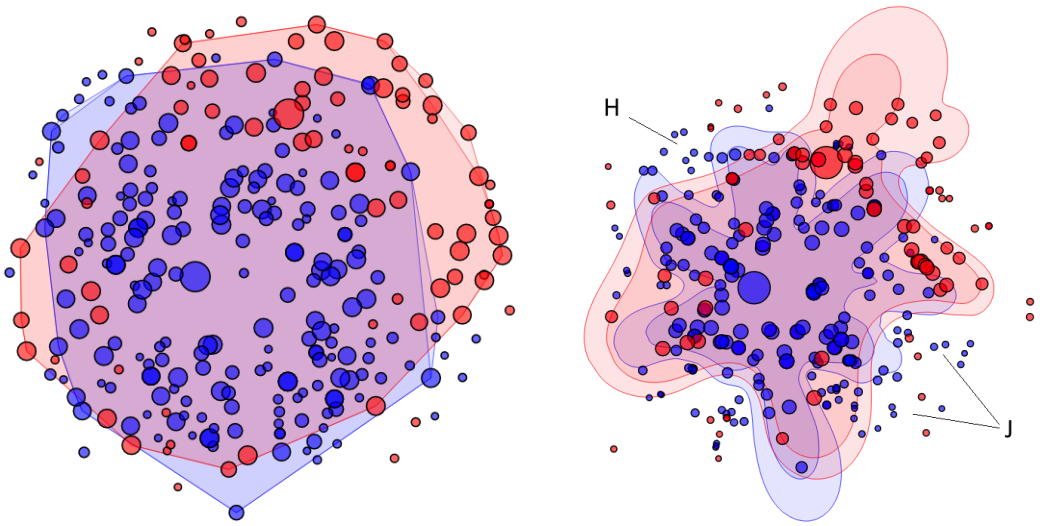


Figure 6.8. Breast cancer data visualizations: (a) bivariate bagplot using MDS and (b) projection bagplot using OAP. Each circle represents a set of patient attributes. The data contain two classes based on patient outcomes: recurrence (red) and nonrecurrence (blue). The recurrence class is seen to deviate from normal, whereas the nonrecurrence class presents a more coherent distribution based on recorded attributes.

able by the UCI machine learning repository [59, 126]. This data set contains two patient classes, recurrence and nonrecurrence, with 85 and 201 instances per class, respectively. There are nine attributes per instance such age range, tumor size, degree of malignancy, etc. Analogous to the approach in Section 6.4.3, we use a categorical kernel to compute distances in the original space [11]. Since the data are bimodal with known class membership information, we use the proposed projection and visualization strategy for multimodal data with a separate monotonic field for each class (Section 6.3). The two medians are drawn as larger circles in the color of their respective class.

This is a case of bimodal data where the classes are not clearly separated. We notice from Figure 6.8 that the nonrecurrence class is somewhat coherent while the recurrence class is more spread out in a ring-like distribution. Such a distribution is a case that highlights the distinction between data depth and data density. Although depth would be high at the geometric center of such a distribution, density would be low due to the absence of members near the center. Form this distribution, we can infer that there must be a large variation among the member attributes of the recurrence class, with no good options among members to be considered typical or most representative.

As expected, the projection bagplot visualization has members in both classes arranged radially, in order of decreasing depth from their respective class medians. The resulting structure makes it easier to spot several interesting outlying cliques. For example, instances near region H correspond to relatively younger individuals with moderate to high tumor size and malignancy and appear to be outliers with respect to both recurrence and nonrecurrence classes. Another interesting region of interest is J where there are instances of older individuals with large tumor size and varying malignancy also appearing as outliers with regard to both classes.

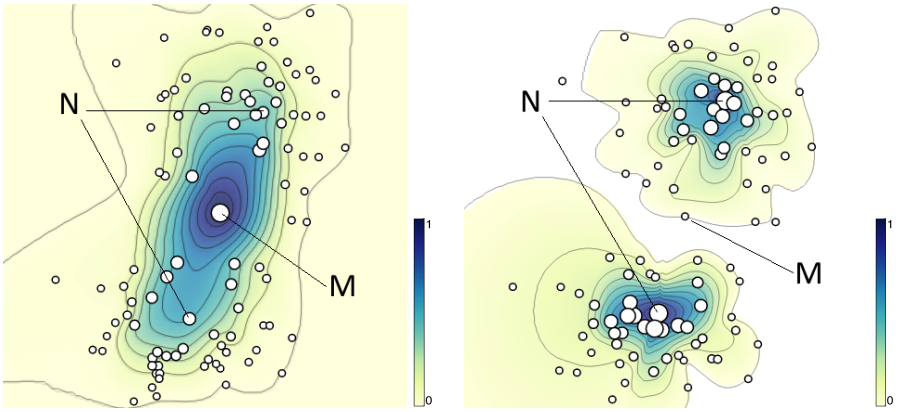
6.5 Discussion

This chapter provides a solution to visualize high-dimensional data ($d \geq 3$) with order statistics in a manner that is popular for visualizing lower dimensional data sets. Similar members are positioned close in the embedding, and central or typical members appear to be more toward the center than outlying or atypical members. To achieve such an embedding, one might consider simply augmenting the data with an additional dimension

containing data-depth values. However, such an approach fails to take into account the anisotropic structure of data, and pushes for members with similar depth to be placed at similar distance from center regardless of direction from center, with members having higher depth value being placed closer to center. On the other hand, OAP allows more flexibility by allowing the rate of fall of depth values to vary smoothly along across adjacent directions, as long as depth values drop monotonically along each direction. This flexibility is helpful to better preserve pairwise distances, particularly in frequently seen cases where points near the boundary along the minor axis are projected close to the median. For example, the proposed methods allow the point X in Figure 6.2 to remain close to the median while also indicating that it is more outlying than it appears in the MDS projection (compare Figure 6.2a to Figures 6.2d and 6.2e).

Oftentimes, multidimensional data to be analyzed are heterogeneous, which means that they include both numerical and categorical dimensions. Our approach is able to handle such data since the method's only requirement is a way to compute distances and center-outward order statistics, which can be computed for such data [74]. Such data are often seen as input for machine learning tasks (e.g., classification or clustering), and the proposed methods can be valuable to understand the structure of kernel spaces where those tasks are performed. A key feature of the proposed projection method (OAP) is its ability to integrate distances and order statistics from different spaces as shown in Sections 6.4.3 and 6.4.4. We may also use order statistics with any other (possibly non-data-depth) method that may be appropriate for the application at hand [116].

In the case of multimodal data *without* known class membership information, OAP can lead to significant misrepresentation of distances if we compute data depth with respect to all points. This misrepresentation of distances is because standard data depth methods, which measure geometric centrality, could assign high depth values for points in region between the clusters, even if the region is sparsely populated; low density does not imply low centrality (see point M in Figure 6.9a). Furthermore, geometric centers of various clusters would be assigned low depth values if they do not lie near the geometric center of the entire distribution (see points N in Figure 6.9a). Such an assignment of depth, although technically correct, can lead to an embedding where cluster centers seem less prominent than surrounding points. One way to make cluster centers more prominent, which may



(a) (b)

Figure 6.9. Field overlay plots using OAP for a synthetically generated 3D multimodal data set with unknown class membership. Order statistics are computed using half-space depth (a) for all points in the data set together and (b) for each cluster separately after a clustering step using the k-means method.

be important in multimodal data, is to first cluster the data and then compute data depth, and monotonic fields, for each cluster separately (see Figure 6.9b).

The proposed projection method (OAP) requires manual adjustment of two parameters: ω_p, which controls the relative emphasis on the order structure with respect to preserving pairwise distances, and ℓ, which controls the lag between updates of the monotonic field. Too small values of ω_p will converge to the MDS layout, whereas too large values of ω_p can cause unnecessary distortion. We find that values between 1 and 3 for ω_p provide a good balance. In the case of ℓ, values that are too small can cause an instability that prevents convergence. The instability arises due to the possibility of (a typically small) increase in overall energy accompanying the computation of the monotonic field. With sufficiently large ℓ, this increase is more than compensated for after points adjust to the new field. On the other hand, too large values of ℓ can delay convergence due to delayed spline updates. We use $\ell = 25$ for all examples in this chapter. During the iterative optimization process, the computational cost of iterations involving an update of the monotonic field is $\mathcal{O}(n^3)$—arising from computation of the thin plate spline (Sec 6.2.4). However, the majority of iterations do not involve field updates and incur a lower cost of $\mathcal{O}(n^2)$ operations.

6.6 Future Work

An important area of application for our method is the visualization of data in kernel spaces (Sections 6.4.3 and 6.4.4). Although we use set band depth for kernel-based examples in this chapter to obtain order statistics, often the only option is to compute depth directly in the kernel space, for example, in the case of ensembles of structured data such as chemical compound graphs [24]. Since existing methods for computing depth are not suitable for *high-dimensional kernel spaces*, which is often the case with graph kernels [112], a method to compute depth in such spaces would expand the scope of data that could be visualized using the proposed method. Such a method to compute depth would need to address the limitations of existing methods by being efficiently computable in high-dimensional spaces as well as having an inner-product-based formulation for operating in kernel spaces.

Another exciting avenue for future work would be to extend the proposed approach to work with manifold-based dimensionality reduction techniques such as Isomap and tSNE, which motivates the need to develop data-depth methods that are also able to operate with respect to manifolds. Automatic estimation of parameter values based on the data to achieve an optimum balance between conveying distances and centrality would also be useful. Finally, projection bagplot visualization could complement other methods for set visualization such as tabplot [106] and parallel coordinates [120] as part of an integrated, interactive system with linked views.

CHAPTER 7

ELLIPSE BAND DEPTH

Visualizing high-dimensional data is important in many domains. Consequently, high-dimensional data visualization continues to be an active area of research. Chapter 6 introduced a technique to visualize high-dimensional data in a way that preserved the order structure and distances in high-dimensional data. This technique relies on quantifying the centrality of data members using data depth (see Section 2.2). Although several data-depth methods have been introduced for computing depth in high-dimensional spaces, current methods face challenges when dealing with data in high-dimensional spaces that are implicitly defined using inner product functions (also known as *kernel* functions). This chapter introduces a novel method, called *ellipse band depth*, to compute data depth in high-dimensional kernel spaces, which are implicitly defined using inner product functions or *kernels*.

A kernel is a function that corresponds to an inner product in a kernel space (also referred to as a feature space). Kernel spaces are often high-dimensional and are related to the original data space by a nonlinear transformation, which does not necessarily have an explicit form. Kernel spaces are a popular concept in the area of machine learning where they are used for data classification tasks that would be difficult to perform in the original data space [71]. Kernel functions, being inner products, can be used to determine similarity or distances between data members in kernel spaces. This property of kernel functions is particularly important in the analysis of both structured and unstructured data, such as graphs and text, and has resulted in the development of a variety of specialized kernel functions for such data [65, 112]. Kernel functions are also used for visualizing data in kernel spaces using methods such as multidimensional scaling (MDS) and kernel principal component analysis (KPCA) [3, 57, 78]. These visualization approaches also suffer from the issue of inaccurate representation of ordering in low-dimensional embeddings of data,

which was discussed in Chapter 6.

Although the technique introduced in Chapter 6 is able to correctly convey order structure of data in high-dimensional spaces using data depth, correctly visualizing data in kernel spaces remains challenging due to the lack of an effective method to compute data depth in such spaces. Kernel spaces are often very high-dimensional spaces, and may not necessarily specify information regarding coordinate axes; data in kernel spaces are typically expressed implicitly using kernel functions. These properties make it infeasible to compute depth for data in such spaces using state-of-the-art methods such as spatial depth [98], half-space depth [108], simplicial depth [62], and functional band depth [66] (see Chapter 6).

Half-space depth can be formulated in terms of inner products. However, half-space depth is expensive to compute in high dimensions. The computational complexity of half-space depth is $O(n^{(d-1)}log(n))$, where n is the number of points in the ensemble and d is the dimension of the space of points. In very high-dimensional spaces, half-space depth is also challenged by the high separability of data leading to a lowest possible depth value for most, if not all, data members. Although simplicial depth can also be formulated in terms of inner products, it is also challenged in high dimensions due to large the number of points $(d + 1)$ needed to form simplices in high dimensions. The large number of data points required for computing a sufficient number of simplices and the associated computations for effectively determining depth make simplicial depth difficult for data in very high-dimensional spaces.

Additionally, simplicial depth in high-dimensional spaces also suffers due to the curse of dimensionality [53]. As the dimension of the space increases, the proportion of volume of space contained within a simplex with respect to the bounding rectangle drops rapidly. This reduction in volume proportion inside a simplex leads to an increased probability for a randomly chosen simplex band to not contain any other point, resulting in a loss of discriminative ability of simplicial depth in such high-dimensional spaces. This issue can be mitigated if the data are intrinsically limited to a lower dimensional subspace; however, that may not always be the case.

On the other hand, although functional depth can handle data in very high-dimensional spaces, its reliance on axis-aligned partitions of space (axis-aligned rectangles) reduces

its effectiveness in capturing the anisotropic structure of data. Furthermore, it can be challenging to compute functional depth in kernel spaces with no explicit information regarding data coordinates and coordinate axes. Although we can obtain obtain coordinate information with respect to a subspace using kernel principal component analysis [95], such coordinates are highly sensitive to outliers in the data, in turn affecting the robustness of functional depth with regard to outliers. In the case of distance-based methods such as L_2 depth, although they can be computed efficiently in kernel spaces, they are limited in their ability to capture the structure of data (see Section 2.2.1).

This chapter presents a novel method to compute center-outward order statistics in high-dimensional kernel spaces. The method overcomes the limitations of existing data depth methods in kernel spaces. In particular, it effectively captures the structure of the data, and can be computed using only inner products for data in kernel spaces. The proposed method, called *ellipse band depth*, is a type of *band*-based method to compute data depth and relies on ellipse-shaped bands, called *ellipse band*, to capture the structure of data (see Section 2.2.1). The rest of this chapter includes a description of the method, results using synthetic and real data sets, and a discussion about the properties and limitations of the proposed method.

7.1 Ellipse Band Depth

This section introduces notions of ellipse band and ellipse band depth, and describes a method to compute ellipse band depth for points in high-dimensional spaces using only inner product information.

An ellipse in $\mathbb{R}^2$ can be defined as a curve surrounding two focal points such that the sum of distances to the two focal points is constant for every point on the curve. To define the ellipse band in $\mathbb{R}^2$, we consider two points, $\{\mathbf{x}_a, \mathbf{x}_b\} \in \mathbb{R}^2$, and a scalar parameter ϵ such that $\epsilon > 1$. We determine a point, $\mathbf{x} \in \mathbb{R}^2$, to be within an ellipse band, $\mathrm{eB}[\cdot]$, formed by $\{\mathbf{x}_a, \mathbf{x}_b\}$ if the following condition holds:

$$\mathbf{x} \in \mathrm{eB}[\mathbf{x}_a, \mathbf{x}_b] \iff d(\mathbf{x}, \mathbf{x}_a) + d(\mathbf{x}, \mathbf{x}_b) \leq \epsilon d(\mathbf{x}_a, \mathbf{x}_b), \tag{7.1}$$

where $d(\cdot)$ denotes the Euclidean distance between two points. Here $\{x_a, x_b\}$ are the foci of an ellipse and ϵ determines its eccentricity. For higher dimensions ($d > 2$), this definition of the ellipse band remains the same. In such cases, a point x is considered to be inside a

band if it falls within an ellipse described by (7.1) on any 2D subspace containing $\{\mathbf{x}_a, \mathbf{x}_b\}$. In order to compute ellipse band depth using only inner product information, we simply need to compute the distances in (7.1) from inner product information using the following equation:

$$d(\mathbf{p}, \mathbf{q}) = \left(\langle \mathbf{p}, \mathbf{p} \rangle + \langle \mathbf{q}, \mathbf{q} \rangle - 2 \times \langle \mathbf{p}, \mathbf{q} \rangle\right)^{\frac{1}{2}}, \tag{7.2}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product.

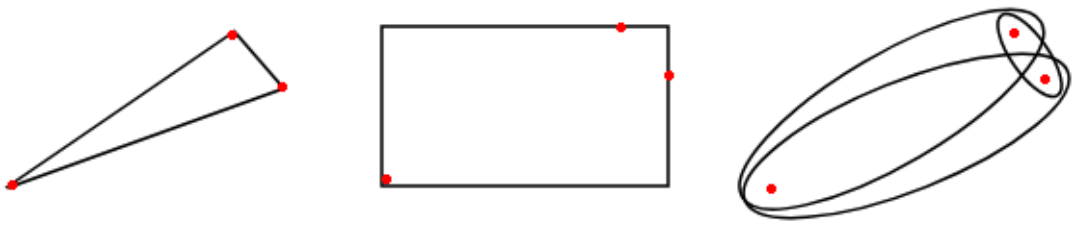
The formulation of ellipse band depth from ellipse band proceeds in a manner that is similar to the formulation of simplicial depth and functional band depth from simplicial and functional bands, respectively (see Figure 7.1). Let X be a probability distribution over points in $\mathbb{R}^2$. Let $\{\mathbf{x}_a, \mathbf{x}_b\}$ be two data points chosen independently from X, and ϵ any value such that $\epsilon > 1$. Then, the ellipse band depth of any point $\mathbf{x} \in \mathbb{R}^d$ with respect to X is the probability of $\mathbf{x}$ falling inside the ellipse band described by $\{\mathbf{x}_a, \mathbf{x}_b\}$ and ϵ, which can be stated as

$$\text{eBD}(\mathbf{x}; X) = \text{Prob}\left[\mathbf{x} \in \text{eB}[\mathbf{x}_a, \mathbf{x}_b]\right]. \tag{7.3}$$

Given a set of points $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \in \mathbb{R}^d$ that are sampled from a random variable $X \in \mathbb{R}^d$, ellipse band depth can be empirically computed as follows:

$$\text{eB}[\mathbf{x}; \mathcal{X}] = \frac{1}{n} \sum_{\mathbf{x}_a, \mathbf{x}_b \in \mathcal{X}} \Lambda(\mathbf{x}, \mathbf{x}_a, \mathbf{x}_b), \tag{7.4}$$

where Λ is an indicator variable such that



(a) (b) (c)

Figure 7.1. Bands used by three different notions of data depth. (a) Simplicial, (b) functional, and (c) ellipse bands are formed by a set of three points in $\mathbb{R}^2$ that are marked in red. Note that three ellipse bands are shown, one for each pair of points in the set.

$$\Lambda = \begin{cases} 1 & \text{if } \mathbf{x} \in \text{eB}[\mathbf{x}_a, \mathbf{x}_b] \\ 0 & \text{if } \mathbf{x} \notin \text{eB}[\mathbf{x}_a, \mathbf{x}_b] \end{cases}. \tag{7.5}$$

7.2 Results

This section contains results of ellipse band depth for synthetic and real data sets. This section also includes results using simplicial depth and L_2 depth for comparison. The L_2 distance in kernel space is computed using (7.2). All data-depth values are normalized to lie in the range $[0, 1]$.

7.2.1 Synthetic 2D Data

The first result uses an angularly symmetric bimodal distribution. This distribution consists of two anisotropic normal distributions with means that are separated vertically (see Figure 7.2a). Figure 7.2b and Figure 7.2c show heatmaps of simplicial and ellipse band depth, respectively, for points on a grid with respect to points sampled from the bimodal distribution. We notice that both notions of depth provide visually similar results for this data set, with the highest depth values being evaluated between the two modes.

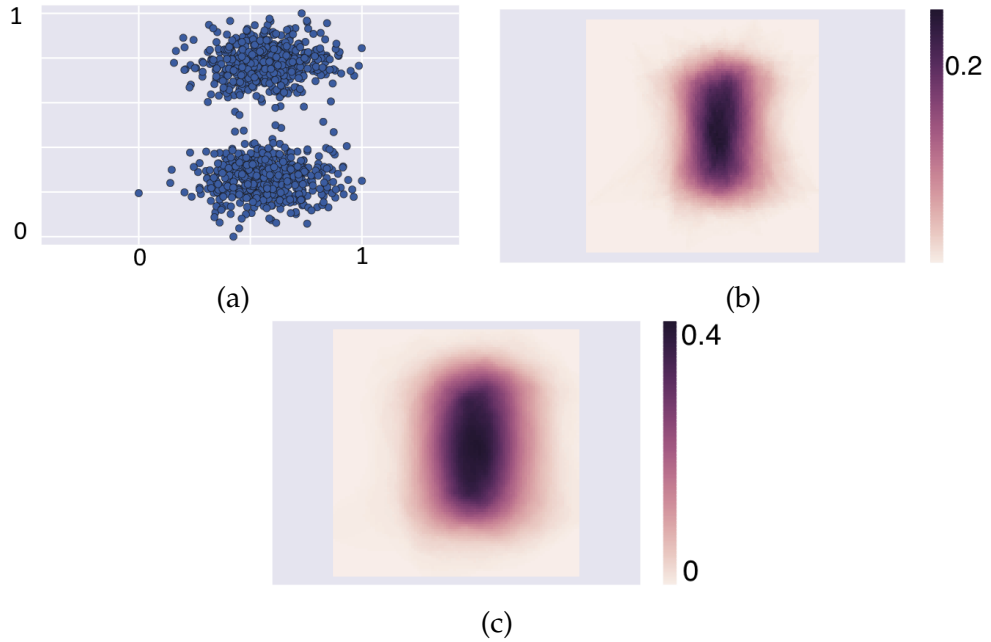


Figure 7.2. Comparison of simplicial depth and ellipse band-depth with synthetic 2D data. (a) An angularly symmetric point distribution. (b) Simplicial depth heatmap. (c) Ellipse band depth heatmap.

7.2.2 Synthetic 3D Data

The second result comes from projecting points sampled from a 3D, anisotropic normal distribution onto a 2D plane. Figure 7.3a and Figure 7.3b show the 2D projections of the data using MDS. Circle sizes indicate the simplicial depth and ellipse depth, respectively, of points in the original 3D space. We can make a few observations. First, we see that the MDS layout does not preserve the order structure with respect to either simplicial depth or ellipse band depth. For example, in Figure 7.3a and Figure 7.3b, point 18 (marked by red arrow) seems to be surrounded by points that are more central in the original space. Second, we observe a similarity between the data-depth values assigned to points by simplicial depth and ellipse band depth with respect to the relative magnitude of centrality values.

Figure 7.3c and Figure 7.3d show the field overlay plots (described in Chapter 6) for the data using simplicial and ellipse band depth, respectively. Here we notice a similarity in the monotonic fields arising from the similarity in the corresponding data-depth values. This similarity indicates that ellipse band depth could be used in place of the simplicial band depth for depth-based based visualizations such as field overlay plots, which makes ellipse band depth particularly attractive for visualizing data in kernel spaces, where computing simplicial band depth could be problematic due to reasons discussed earlier in this chapter.

Figure 7.3c, Figure 7.3d, and Figure 7.3e show field overlay plots for the data using ellipse band depth with different values of parameter ϵ. A lower ϵ value in Figure 7.3e causes a loss in discriminative ability that can be inferred by observing the similar depth values of the most central members. On the other hand, a higher ϵ value in Figure 7.3f also causes a loss in discriminative ability that is evident from generally more similar depth values and the smoother monotonic field in the background.

7.2.3 Chemicals in Kernel Space

The final result displays the similarity between a set of chemical compounds from the MUTAG data set [24]. This set contains 63 compounds that have been identified as *nonmutagenic*. We use a graph kernel, called a propagation kernel, to determine the similarity between the molecules based on their chemical structure [80]. The similarity

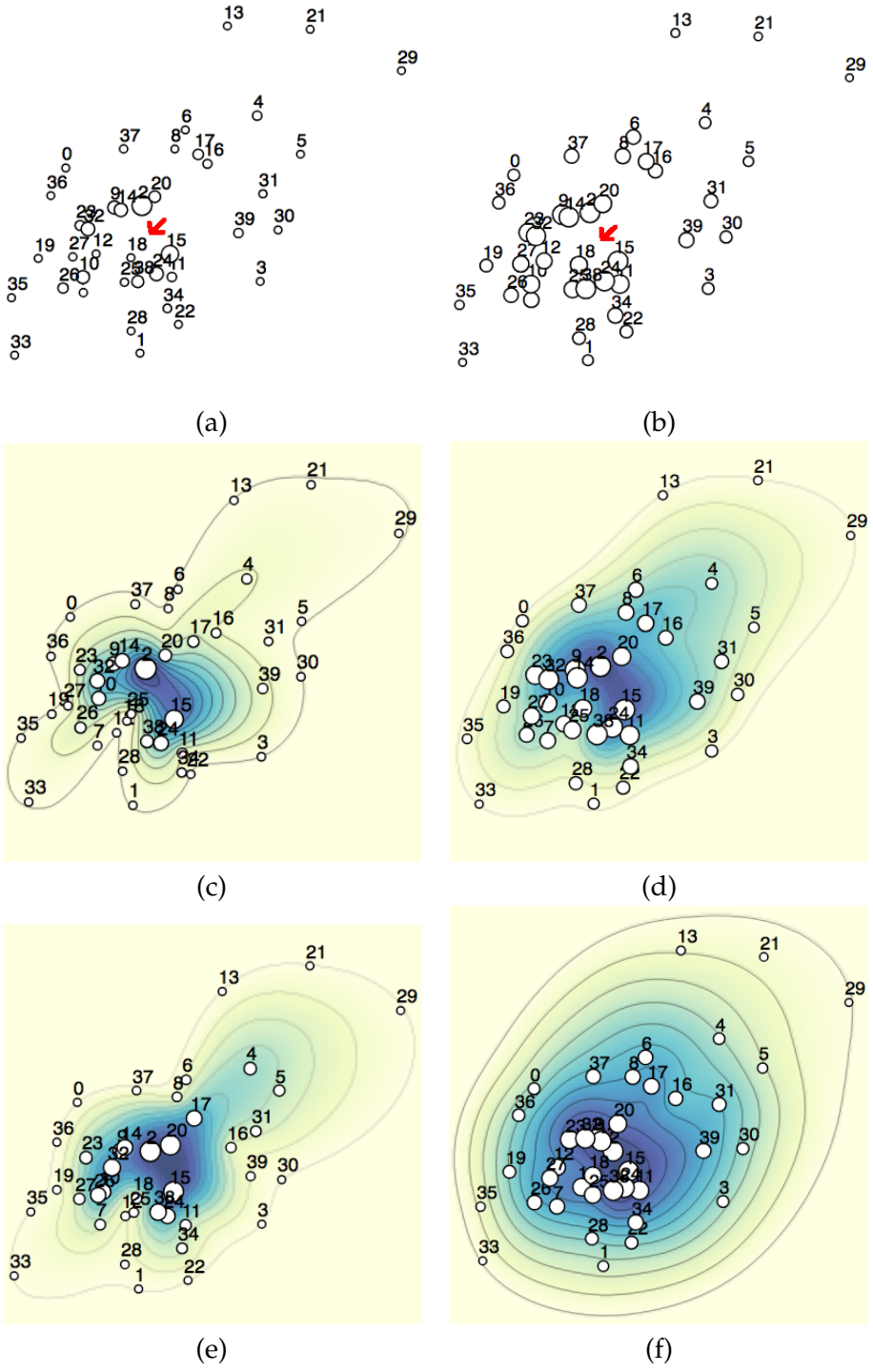


Figure 7.3. Visualizations of a 3D anisotropic multivariate normal distribution. (a) MDS projection with circle size indicating simplicial depth, (b) MDS projection with circle size indicating ellipse band depth, (c) field overlay plot from Chapter 6 using simplicial depth, and field overlay plots using ellipse band depth with (d) $\epsilon = 1.1$, (e) $\epsilon = 1.01$, and (f) $\epsilon = 1.5$.

values between chemicals are used to obtain a 2D MDS embedding of the data set as seen in Figure 7.4a and Figure 7.4b. Circle sizes indicate l_2 distance depth and ellipse band depth, respectively, in the kernel space. Figure 7.4c and Figure 7.4d show the field overly plots for the chemical data set using l_2 distance depth and ellipse band depth values, respectively. We notice that the contours of the monotonic field with ellipse band depth have a more irregular structure when compared to contours with l_2 distance depth, which are smooth and coherent with the MDS embedding. Such a difference in regularity of monotonic fields indicates that the ellipse band depth is able to capture information about the structure of

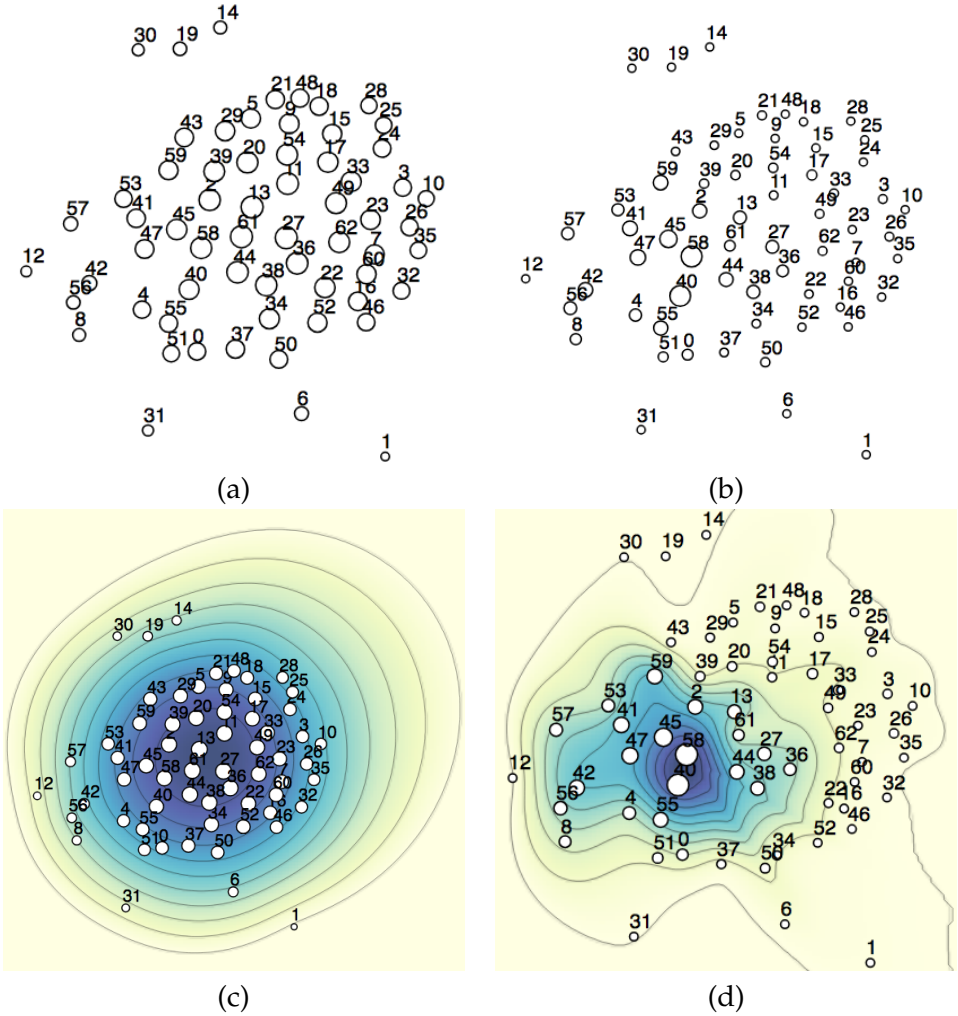


Figure 7.4. Similarity of chemical molecules in the MUTAG dataset. (a) MDS visualization with circle size encoding distance depth, (b) MDS visualization with circle size encoding ellipse band depth, (c) field overlay plot using distance depth, and (d) field overlay plot using ellipse band depth.

the data in the kernel space that is missed by l_2 distance depth.

7.3 Discussion

The proposed ellipse-band-depth method has several properties that make it an attractive addition to the repertoire of data-depth methods. First, it is easily computed using inner products. Second, it is effective in very high-dimensional spaces since the ellipse band always spans volume in all dimensions. Third, the ellipse band can better capture the correlation in the data than the rectangle band, which is used for computing functional depth. This enhanced ability to capture shape is because the ellipse band can tightly fit the data regardless of the positions of the points with regard to the coordinate axes. Finally, the ellipse band depth is simple and fast to compute.

Despite the advantages of the proposed method, areas remain that require further investigation. The desirable properties of data depth have not yet been established for ellipse band depth and remain an active area of research (see Section 2.2). Although a rigorous proof for the monotonicity property has not been established yet, empirical results do indicate that the property holds in the case of ellipse band depth (see Figure 7.5).

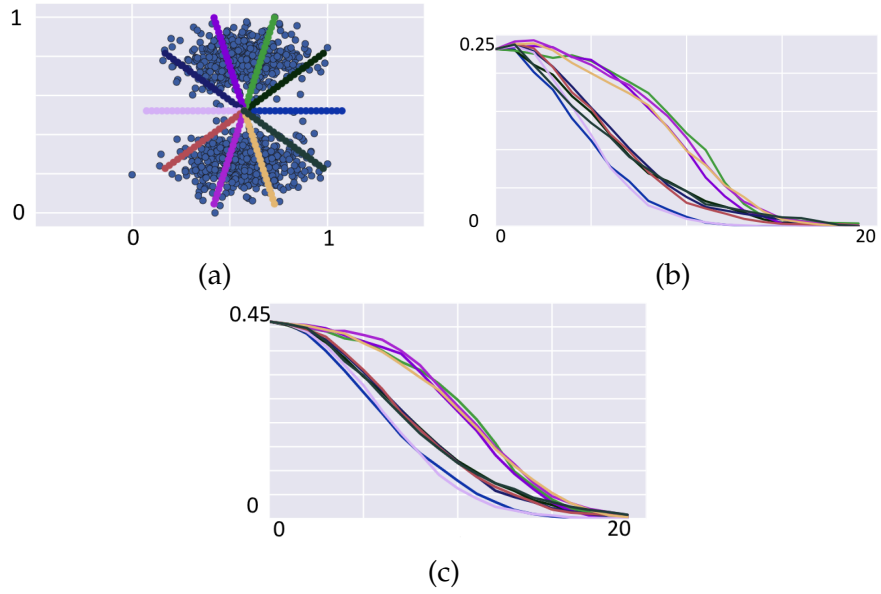


Figure 7.5. Simplicial and ellipse band depth along center-outward rays. Empirical results for both methods indicate monotonic drop in depth value, barring some sample noise in case of simplicial depth. (a) Direction of rays traveling away from the center, (b) simplicial depth, and (c) ellipse band depth along the rays.

Another issue that is still unresolved is estimation of the parameter ϵ. This parameter determines the eccentricity or shape of the ellipse band. Although too small a value would lead to thin ellipses that could fail to contain any points in high dimensions, too large a value would lead to a loss of structure of the ellipse band; both cases leading to a loss in the ability of ellipse band depth to discriminate between points with regard to their centrality in the data set.

There are a variety of possible solutions for determining an appropriate ϵ with regard to a particular data set. For example, one such approach would be to use statistics such as average nearest neighbor distance in the data set to ensure that the ellipse bands are wide enough to have a high probability of containing nearby points. Another approach to set ϵ involves sampling an additional point for each ellipse band and adjusting the ϵ such that this point touches the boundary of the ellipse band. Although these methods to determine ϵ would could possibly affect the behavior of the proposed epsilon band depth method, analysis of such effects is a topic for future work, and out of the scope of the present chapter.

CHAPTER 8

DISCUSSION AND FUTURE WORK

This dissertation tackles the problem of visualizing several different kinds of data types that can be broadly classified into either ensemble data or graphs. The common theme underlying the methods introduced in this dissertation is quantification and visualization of order structure in data. In both kinds of data, ensembles and graphs, we consider data to be composed of a collection of individual members, which can be complex entities such as 3D shapes in the case of ensemble data, or nodes in the case of a graph. The methods in this dissertation rely on a family of descriptive statistical methods, known as *data depth*, to determine the centrality of members in the data. Based on data depth, the most central members are considered to be most representative of the data whereas least central members are considered to be most atypical or outlying with respect to the data set. Typical and outlying members as well as the variability among all members are considered important features in many applications. The visualization methods introduced in this dissertation highlight such important features for different types of data. These data types include ensembles of 3D shapes and paths on a graph, which are relevant for applications such as understanding the typical structure of brain anatomy across populations in medical imaging (Chapter 3) and identifying anomalous packet paths in Internet routing (Chapter 4).

The use of data depth in visualizations to summarize data is not new; in fact, the well-known Tukey boxplot is an example of such a visualization that was introduced decades ago and continues to be widely used. The novelty of the work in this dissertation is in introducing more effective methods to determine data depth, and expanding the scope of data types and applications that can be tackled using depth-based visualizations. A key contribution of this dissertation with regard to novel data-depth methods is the development of the *path-band-depth* method to determine center-outward order

statistics for path ensembles on a graph (Chapter 4). The path-band-depth method is able to capture correlation patterns that are missed by existing methods for determining center-outwardness for paths. Another novel data-depth method introduced is the *ellipse band depth* with several advantages over existing methods for determining the centrality of points in high-dimensional spaces (Chapter 7).

Apart from novel data-depth methods, another key contribution of this dissertation is the development of visualization techniques for various data types that convey key features of data by taking advantage of existing and proposed data-depth methods. Chapter 4, Chapter 5, and Chapter 6 introduce such visualization techniques for ensembles of paths on graphs, nodes on graphs, and point ensembles in high-dimensional spaces, respectively. These visualizations highlight the key members in the data while also providing additional context in terms of variability (paths and high-dimensional points) or relationships between members (between nodes on a graph).

Finally, this dissertation also demonstrates the utility of the data-depth-based visualizations in various real applications. Chapter 3 shows the advantages of depth-based visualization, specifically, a 3D extension of the contour boxplot, for the application of evaluating alignment of 3D brain images in the context of medical imaging. Chapter 4, Chapter 5, and Chapter 6 include examples from various real application domains such as transport planning, social networks, and health care. Although the methods proposed in this dissertation have several advantages, they also have certain limitations. The rest of this chapter discusses those limitations, possible options in order to overcome the limitations, and a few additional directions for future work.

8.1 Discussion

We now look at a few limitations that we need to be aware of while using the proposed methods and planning future directions of work. Although 3D contour boxplots are effective in summarizing ensemble of 3D shapes, they suffer from the known problem of occlusion in 3D, which may lead to misunderstanding about the global nature of analysis (determination of key members is done using 3D volume analysis, although the results are visible for only a single cut plane at any instance). One possible solution to mitigate this issue is to make use of the transparency/alpha channels for rendering the parts of contour

boxplots that do not intersect the cut plane, including the clipped portion, so that the entire structure of the inner members and bands is always visible. This additional context could be a cue that the key features are precomputed and not dependent on the orientation of the cut planes.

Occlusion is also an issue for 2D embedding methods introduced in this dissertation when dealing with large numbers of data members (Chapter 5 and Chapter 6). Furthermore, the computational cost of computing the thin plate spline for determining the embedding is n^3 where n is the number of data members. One option to address this issue is to compute the spline using a randomly sampled subset of data members. In the case of path boxplots, the computational cost of dealing with ensembles with a large number of paths is high. Specifically, the cost of the path alignment step for determining path band depth increases exponentially with the number of paths (see Chapter 4). In order to address this problem, we use an approximation of aligning subsets of paths rather than aligning all paths at once. A limitation of this method is that a theoretical error bound on the quality of approximate path band depth values is still unknown.

In the case of ellipse band depth, more work is needed in order to demonstrate the utility of the method in real-world applications. In particular, it remains to be shown whether ellipse band depth exhibits desirable properties of data-depth methods such as monotonicity, maximum at center, and zero at infinity (see Section 2.2). Also, the effectiveness of the ellipse band depth depends on careful selection of the parameter ϵ, which determines the eccentricity of the ellipse bands. Further work is required to understand the impact of various methods of automatically determining ϵ based on the data. Additionally, in the current formulation of epsilon band depth, a large value of ϵ would lead to ellipse bands that protrude much beyond the region between the points forming the band, particularly along the direction of the major axis. This protrusion of bands can lead to nonzero depth values for points outside the convex hull of the distribution, which is a characteristic seen in less discriminative measures of data depth such as distance-based and functional depth. A possible way to improve the discriminative ability of ellipse band depth would be to scale the ellipse band to have the points determining the ellipse band to fall on the boundary of ellipse (where the boundary intersects with the major axis) rather than inside the ellipse (at foci).

8.2 Future Work

There are many interesting avenues for further work that would complement the work in this dissertation. In Chapter 3, we saw the 3D contour boxplot is useful for evaluating alignment of brain MRI images in brain atlases. Integration of the 3D contour boxplot pipeline with existing atlas construction software could be valuable for researchers who work with atlases. In the case of anisotropic radial layout (Chapter 5) and order aware projection (OAP) (Chapter 6), theoretical guarantees with regard to the convergence of the optimization process as well as more computationally efficient approaches, on the lines of SMACOF algorithm [58], for arriving at a solution would be useful when handling real data sets. Even for path band depth (Chapter 4) and ellipse band depth (Chapter 7), more results on theoretical guarantees would be useful. For example, an interesting direction for theoretical work includes determining error bounds on approximate solutions and establishing desirable properties of data depth in the case of path band depth and ellipse band depth, respectively.

In addition to extensions to the proposed methods, a few new research directions also seem interesting in the context of this dissertation. One particularly important type of data that is not tackled in this dissertation is ensembles of *aligned* graphs. In such ensembles, all graphs share a common vertex set, and members of the ensemble are independent samples from some stochastic, generative process (e.g., probability distribution) on the edges and the edge weights. Although graph ensembles can be visualized in terms of similarity between graphs by using methods introduced in Chapter 6 and Chapter 7, oftentimes it is necessary to understand graph ensembles in the context of the connectivity structure of graphs. Such an understanding of graph ensembles is particularly important in the domain of neuroscience. Neuroscientists use resting-state functional MRI (fMRI) data to infer correlations in blood flow between brain regions that are represented as functional graphs. In functional graphs of the brain, vertices represent individual brain regions and edges connect regions with a high correlation in blood flow patterns [87, 104]. Understanding ensembles of such graphs is important for comparing the general brain structures across different populations groups such as healthy individuals and Alzheimer's patients [5].

Existing methods to visualize aligned graph ensembles include the heatmap [14] and the cell histogram [121], which fail to capture correlations across edges (see Figure 8.1).

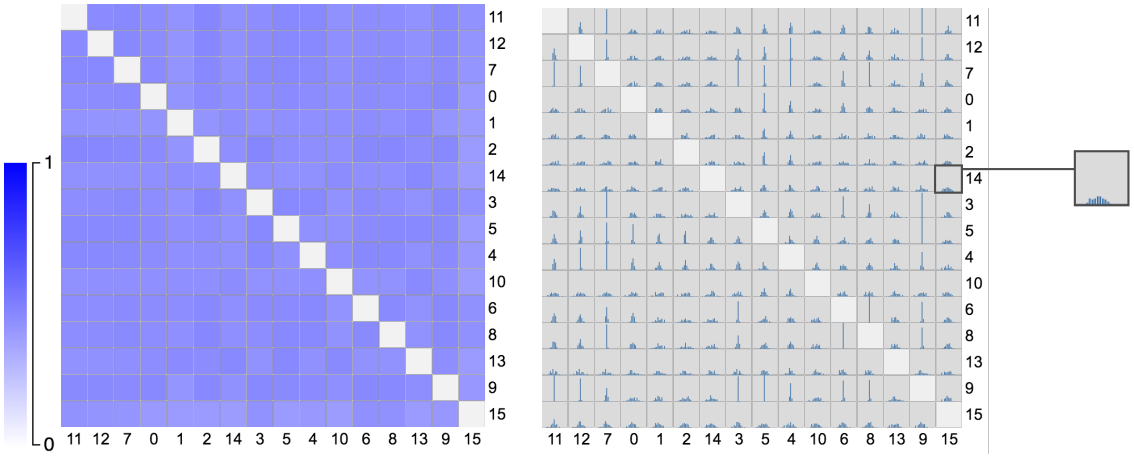


Figure 8.1. Existing methods to visualize aligned graph ensembles. (a) Adjacency matrix heatmaps and (b) cell histogram. In both visualizations, each cell summarizes the weights on an edge (between specific pair of nodes) across an entire ensemble of graphs. Furthermore, the encodings in each cell are determined independent of edges corresponding to other cells.

Some other data types that are of interest with regard to future work on data-depth-based visualization include ensembles of trees [6] and 3D scalar fields.

Another important direction for future work is to determine data depth on manifolds. The existing geometric data-depth measures that are used in Chapter 6 for preserving centrality lower dimensional embeddings do not consider any manifold structure and could cause significant distortion of distances in the embedding if combined with manifold-based dimensionality reduction methods. Manifold-based measures of data depth could complement manifold-based dimensionality reduction methods such as isomap [105] and t-SNE [68] to preserve order structure with regard to the manifold.

APPENDIX

NAME ASSOCIATIONS FOR NODES IN VISUALIZATIONS IN CHAPTER 5

A.1 Name Associations for Nodes in the Terrorist Network (Figure 5.4)

Node ID	Name	Node ID	Name
0	Jamal Zougam	35	Mohamed Oulad Akcha
1	Mohamed Bekkali	36	Rachid Oulad Akcha
2	Mohamed Chaoui	37	Mamoun Darkaanli
3	Vinay Kholy	38	Fouad El Morabit Anghar
4	Suresh Kumar	39	Abdeluahid Berrak
5	Mohamed Chedadi	40	Said Berrak
6	Imad Eddin Barakat	41	Waanid Altaraki Almasri
7	Abdelai Benyaich	42	Abddenabi Koujma
8	Abu Abderrahame	43	Otman El Gnaut
9	Omar Dhegayes	44	Abdelilah el Fouad
10	Amer Aii	45	Mohamad Bard Ddin Akkab
11	Abu Musad Alsakaoui	46	Abu Zubaidah
12	Mohamed Atta	47	Sanel Sjekirika
13	Rami Binalshibh	48	Parlindumgan Siregar
14	Mohamed Belfatmi	49	El Hemir
15	Said Bahaji	50	Anuar Asri Rifaat
16	Al Amrous	51	Rachid Adli
17	Galeb Kalaje	52	Ghasoub Al Albrash
18	Abderrahim Zbakh	53	Said Chedadi
19	Farid Oulad Ali	54	Mohamed Bahaiah
20	Jos Emilio Sure	55	Taysir Alouny
21	Khalid Ouled Akcha	56	OM Othman Abu Qutada
22	Rafa Zuher	57	Shakur
23	Naima Oulad Akcha	58	Driss Chebli
24	Abdelkarim el Mejjati	59	Abdul Fatal
25	Abdelhalak Bentasser	60	Mohamed El Egipcio
26	Anwar Adnan Ahmad	61	Nasredine Boushoa
27	Basel Ghayoun	62	Semaan Gaby Eid
28	Faisal Alluch	63	Emilio Llamo
29	S B Abdelmajid Fakhet	64	Ivan Granados
30	Jamal Ahmidan	65	Raul Gonales Pere
31	Said Ahmidan	66	El Gitanillo
32	Hamid Ahmidan	67	Mouta Almallah
33	Mustafa Ahmidan	68	Mohamed Almallah
34	Antonio Toro	69	Yousef Hichman

A.2 Name Associations for Nodes in *Les Miserables* Network (Figure 5.5)

Node ID	Name	Node ID	Name	Node ID	Name
0	Myriel	26	Cosette	52	MmePontmercy
1	Napoleon	27	Javert	53	MlleVaubois
2	MlleBaptistine	28	Fauchelevent	54	LtGillenormand
3	MmeMagloire	29	Bamatabois	55	Marius
4	CountessDeLo	30	Perpetue	56	BaronessT
5	Geborand	31	Simplice	57	Mabeuf
6	Champtercier	32	Scaufflaire	58	Enjolras
7	Cravatte	33	Woman	59	Combeferre
8	Count	34	Judge	60	Prouvaire
9	OldMan	35	Champmathieu	61	Feuilly
10	Labarre	36	Brevet	62	Courfeyrac
11	Valjean	37	Chenildieu	63	Bahorel
12	Marguerite	38	Cochepaille	64	Bossuet
13	MmeDeR	39	Pontmercy	65	Joly
14	Isabeau	40	Boulatruelle	66	Grantaire
15	Gervais	41	Eponine	67	MotherPlutarch
16	Tholomyes	42	Anelma	68	Gueulemer
17	Listolier	43	Woman	69	Babet
18	Fameuil	44	MotherInnocent	70	Claquesous
19	Blacheville	45	Gribier	71	Montparnasse
20	Favourite	46	Jondrette	72	Toussaint
21	Dahlia	47	MmeBurgon	73	Child
22	Zephine	48	Gavroche	74	Child
23	Fantine	49	Gillenormand	75	Brujon
24	MmeThenardier	50	Magnon	76	MmeHucheloup
25	Thenardier	51	MlleGillenormand		

REFERENCES

[1] *Nektar++*, 2015. `http://www.nektar.info`.

[2] P. K. Agarwal, R. B. Avraham, H. Kaplan, and M. Sharir, *Computing the discrete Fréchet distance in subquadratic time*, SIAM J. Comput., 2 (2014), pp. 156–167.

[3] D. K. Agrafiotis, D. N. Rassokhin, and V. S. Lobanov, *Multidimensional scaling and visualization of large molecular similarity tables*, J. Comput. Chem., 22 (2001), pp. 488–500.

[4] A. Alexander-Bloch, R. Lambiotte, B. Roberts, J. Giedd, N. Gogtay, and E. Bullmore, *The discovery of population differences in network community structure: New methods and applications to brain functional networks in schizophrenia*, Neuroimage, 59 (2012), pp. 3889–3900.

[5] B. Alper, B. Bach, N. Henry Riche, T. Isenberg, and J.-D. Fekete, *Weighted graph comparison techniques for brain connectivity analysis*, in Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, 2013, pp. 483–492.

[6] N. Amenta, M. Datar, A. Dirksen, M. de Bruijne, A. Feragen, X. Ge, J. H. Pedersen, M. Howard, M. Owen, J. Petersen, et al., *Quantification and visualization of variation in anatomical trees*, in Research in Shape Modeling, K. Leonard and S. Tari, eds., Springer, Cham, 2015, pp. 57–79.

[7] F. Anscombe, *Graphs in statistical analysis*, Amer. Statist., 27 (1973), pp. 17–21.

[8] M. S. Apaydin, D. L. Brutlag, C. Guestrin, D. Hsu, J.-C. Latombe, and C. Varma, *Stochastic roadmap simulation: An efficient representation and algorithm for analyzing molecular motion*, J. Comput. Biol., 10 (2003), pp. 257–281.

[9] C. Bachmaier, *A radial adaptation of the sugiyama framework for visualizing hierarchical information*, IEEE Trans. Vis. Comput. Graphics, 13 (2007), pp. 583–594.

[10] B. Baingana and G. B. Giannakis, *Embedding graphs under centrality constraints for network visualization*, arXiv preprint arXiv:1401.4408, (2014).

[11] L. A. Belanche Muñoz and M. Villegas, *Kernel functions for categorical variables with application to problems in the life sciences*, in Artificial Intelligence Research and Development: Proceedings of the 16 International Conference of the Catalan Association of Artificial Intelligence, 2013, pp. 171–180.

[12] F. L. Bookstein, *Principal warps: Thin-plate splines and the decomposition of deformations*, IEEE Trans. Pattern Anal. Mach. Intell., 11 (1989), pp. 567–585.

[13] I. Borg and P. J. Groenen, *Modern multidimensional scaling: Theory and applications*, Springer-Verlag, New York, 2005.

[14] G. Bounova and O. de Weck, *Overview of metrics and their correlation patterns for multiple-metric topology analysis on heterogeneous graph ensembles*, Phys. Rev. E, 85 (2012), p. 016117.

[15] U. Brandes, P. Kenis, and D. Wagner, *Communicating centrality in policy network drawings*, IEEE Trans. Vis. Comput. Graphics, 9 (2003), pp. 241–253.

[16] U. Brandes and C. Pich, *More flexible radial layout*, in International Symposium on Graph Drawing, Springer, 2009, pp. 107–118.

[17] U. Brandes and D. Wagner, *Analysis and visualization of social networks*, Graph drawing software, (2004), pp. 321–340.

[18] K. Brodlie, R. AllendesOsorio, e. J. Lopes, Adriano", R. Earnshaw, D. Kasik, J. Vince, and P. C. Wong, *A review of uncertainty in data visualization*, in Expanding the Frontiers of Visual Analytics and Visualization, Springer, London, 2012, pp. 81–109.

[19] K. Butler, T. Farley, P. Mcdaniel, and J. Rexford, *A survey of bgp security issues and solutions*, Proc. IEEE, 98 (2010), pp. 100–122.

[20] H. Carrillo and D. Lipman, *The multiple sequence alignment problem in biology*, SIAM J. Appl. Math., 48 (1988), pp. 1073–1082.

[21] Y. Chen, X. Dang, H. Peng, and H. L. Bart, *Outlier detection with the kernelized spatial depth function*, IEEE Trans. Pattern Anal. Mach. Intell., 31 (2009), pp. 288–305.

[22] W. S. Cleveland and R. McGill, *Graphical perception: Theory, experimentation, and application to the development of graphical methods*, J. Amer. Statist. Assoc., 79 (1984), pp. 531–554.

[23] J. Cox, D. House, and M. Lindell, *Visualizing uncertainty in predicted hurricane tracks*, Int. J. Uncertain. Quantif., 3 (2013).

[24] A. K. Debnath, R. L. Lopez de Compadre, G. Debnath, A. J. Shusterman, and C. Hansch, *Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds. correlation with molecular orbital energies and hydrophobicity*, J. Med. Chem., 34 (1991), pp. 786–797.

[25] H. Dette, N. Neumeyer, K. F. Pilz, et al., *A simple nonparametric estimator of a strictly monotone regression function*, Bernoulli, 12 (2006), pp. 469–490.

[26] H. Dette and R. Scheder, *Strictly monotone and smooth nonparametric regression for two or more variables*, Canad. J. Statist., 34 (2006), pp. 535–561.

[27] T. Dwyer, *Scalable, versatile and simple constrained graph layout*, Comput. Graphics Forum, 28 (2009), pp. 991–998.

[28] T. Dwyer, Y. Koren, and K. Marriott, *Ipsep-cola: An incremental procedure for separation constraint layout of graphs*, IEEE Trans. Vis. Comput. Graphics, 12 (2006), pp. 821–828.

[29] T. Dwyer, Y. Koren, and K. Marriott, *Constrained graph layout by stress majorization and gradient projection*, Discrete Math., 309 (2009), pp. 1895–1908.

[30] R. Dyckerhoff, K. Mosler, and G. Koshevoy, *Zonoid data depth: Theory and computation*, Physica-Verlag, Heidelberg, 1996, pp. 235–240.

[31] R. Dyckerhoff and P. Mozharovskyi, *Exact computation of the halfspace depth*, Comput. Statist. Data Anal., 98 (2016), pp. 19–30.

[32] T. Eiter and H. Mannila, *Computing discrete Fréchet distance*, Tech. Report CD-TR 94/64, Technical University of Vienna, 1994.

[33] M. R. Evans, D. Oliver, S. Shekhar, and F. Harvey, *Summarizing trajectories into k-primary corridors: A summary of results*, in Proceedings of the 20th International Conference on Advances in Geographic Information Systems, 2012, pp. 454–457.

[34] M. R. Evans, D. Oliver, S. Shekhar, and F. Harvey, *Fast and exact network trajectory similarity computation: A case-study on bicycle corridor planning*, in Proceedings of the 2nd ACM SIGKDD International Workshop on Urban Computing, ACM, 2013, p. 9.

[35] P. A. Forero and G. B. Giannakis, *Sparsity-exploiting robust multidimensional scaling*, IEEE Trans. Signal Process., 60 (2012), pp. 4118–4134.

[36] L. C. Freeman, *A set of measures of centrality based on betweenness*, Sociometry, 40 (1977), pp. 35–41.

[37] L. C. Freeman, *Centrality in social networks conceptual clarification*, Soc. Networks, 1 (1978), pp. 215–239.

[38] T. M. J. Fruchterman and E. M. Reingold, *Graph drawing by force-directed placement.*, Softw., Pract. Exper., 21 (1991), pp. 1129–1164.

[39] E. R. Gansner, Y. Koren, and S. North, *Graph drawing by stress majorization*, in Graph Drawing, J. Pach, ed., Springer, Berlin Heidelberg, 2005, pp. 239–250.

[40] M. G. Genton, C. Johnson, K. Potter, G. Stenchikov, and Y. Sun, *Surface boxplots*, Stat., 3 (2014), pp. 1–11.

[41] H. Gibson, J. Faith, and P. Vickers, *A survey of two-dimensional graph layout techniques for information visualisation*, Inform. Vis., 12 (2013), pp. 324–357.

[42] T. Gneiting and A. E. Raftery, *Weather forecasting with ensemble methods*, Sci., 310 (2005), pp. 248–249.

[43] M. Haklay and P. Weber, *Openstreetmap: User-generated street maps*, IEEE Pervasive Comput., 7 (2008), pp. 12–18.

[44] B. Hayes, *Connecting the dots can the tools of graph theory and social-network studies unravel the next big plot?*, Amer. Scientist, 94 (2006), pp. 400–404.

[45] R. R. He, H. X. Liu, and A. L. Kornhauser, *Temporal and spatial variability of travel time*, Center for Traffic Simulation Studies. Paper UCI-ITS-TS-02, 14 (2002).

[46] M. Hua and J. Pei, *Probabilistic path queries in road networks: Traffic uncertainty aware path selection*, in Proceedings of the 13th International Conference on Extending Database Technology, 2010, pp. 347–358.

[47] R. J. Hyndman, *Computing and graphing highest density regions*, Amer. Statist., 50 (1996), pp. 120–126.

[48] R. J. Hyndman and H. L. Shang, *Rainbow plots, bagplots, and boxplots for functional data*, J. Comput. Graph. Statist., 19 (2010), pp. 29–45.

[49] C. R. Jack, M. A. Bernstein, N. C. Fox, P. Thompson, G. Alexander, D. Harvey, B. Borowski, P. J. Britson, J. L Whitwell, C. Ward, A. M. Dale, J. P. Felmlee, J. L. Gunter, D. L. Hill, R. Killiany, N. Schuff, S. Fox-Bosetti, C. Lin, C. Studholme, C. S. DeCarli, G. Krueger, H. A. Ward, G. J. Metzger, K. T. Scott, R. Mallozzi, D. Blezek, J. Levy, J. P. Debbins, A. S. Fleisher, M. Albert, R. Green, G. Bartzokis, G. Glover, J. Mugler, and M. W. Weiner, *The Alzheimer's Disease Neuroimaging Initiative (ADNI): MRI methods*, JMRI, 27 (2008), pp. 685–691.

[50] S. Joshi, B. Davis, B. M. Jomier, and G. G. B, *Unbiased diffeomorphic atlas construction for computational anatomy*, Neuroimage, 23 (2004), pp. 151–160.

[51] W. Just, *Computational complexity of multiple sequence alignment with sp-score*, J. Comput. Biol., 8 (2001), pp. 615–623.

[52] T. Kamada and S. Kawai, *An algorithm for drawing general undirected graphs*, Inform. Process. Lett., 31 (1989), pp. 7–15.

[53] E. Keogh and A. Mueen, *Curse of Dimensionality*, Springer US, Boston, MA, 2017, pp. 314–315.

[54] A. Kharrat, I. S. Popa, K. Zeitouni, and S. Faiz, *Clustering algorithm for network constraint trajectories*, in Headway in Spatial Data Handling, A. Ruas and C. Gold, eds., Springer, Berlin Heidelberg, 2008, pp. 631–647.

[55] D. E. Knuth, *The Stanford GraphBase: A Platform for Combinatorial Computing*, ACM, New York, NY, USA, 1993.

[56] Y. LeCun and C. Cortes, *MNIST handwritten digit database*. `http://yann.lecun.com/exdb/mnist/`, 2010.

[57] Y.-J. Lee and S.-Y. Huang, *Reduced support vector machines: A statistical theory*, IEEE Trans. Neural Netw., 18 (2007), pp. 1–13.

[58] J. D. Leeuw, I. J. R. Barra, F. Brodeau, G. Romier, and B. V. C. (eds), *Applications of convex analysis to multidimensional scaling*, in Recent Developments in Statistics, North Holland Publishing Company, 1977, pp. 133–146.

[59] M. Lichman, *UCI machine learning repository*. `http://archive.ics.uci.edu/ml`, 2013.

[60] L. Liu, A. P. Boone, I. T. Ruginski, L. Padilla, M. Hegarty, S. H. Creem-Regehr, W. B. Thompson, C. Yuksel, and D. H. House, *Uncertainty visualization by representative sampling from prediction ensembles*, IEEE Trans. Vis. Comput. Graphics, 23 (2017), pp. 2165–2178.

[61] L. Liu, M. Mirzargar, R. M. Kirby, R. Whitaker, and D. H. House, *Visualizing time-specific hurricane predictions, with uncertainty, from storm path ensembles*, 34 (2015), pp. 371–380.

[62] R. Y. Liu, *On a notion of data depth based on random simplices*, Ann. Statist., 18 (1990), pp. 405–414.

[63] R. Y. Liu, J. M. Parelius, K. Singh, et al., *Multivariate analysis by data depth: Descriptive statistics, graphics and inference (with discussion and a rejoinder by Liu and Singh)*, Ann. Statist., 27 (1999), pp. 783–858.

[64] S. Liu, D. Maljovec, B. Wang, P.-T. Bremer, and V. Pascucci, *Visualizing high-dimensional data: Advances in the past decade*, IEEE Trans. Vis. Comput. Graphics, 23 (2017), pp. 1249–1268.

[65] H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins, *Text classification using string kernels*, J. Mach. Learn. Res., 2 (2002), pp. 419–444.

[66] S. López-Pintado and J. Romo, *On the concept of depth for functional data*, J. Amer. Statist. Assoc., 104 (2009), pp. 718–734.

[67] S. López-Pintado, Y. Sun, J. Lin, and M. Genton, *Simplicial band depth for multivariate functional data*, Adv. Data Anal. Classif., 8 (2014), pp. 1–18.

[68] L. V. D. Maaten and G. Hinton, *Visualizing data using t-sne*, J. Mach. Learn. Res., 9 (2008), pp. 2579–2605.

[69] S. M. Mahmoud, A. Lotfi, and C. Langensiepen, *User activities outlier detection system using principal component analysis and fuzzy rule-based system*, in Proceedings of the Fifth International Conference on Pervasive Technologies Related to Assistive Environments, ACM, 2012, p. 26.

[70] V. E. McGee, *The multidimensional analysis of elastic distances*, British J. Math. Statist. Psychol., 19 (1966), pp. 181–196.

[71] P. Milanfar, *A tour of modern image filtering: New insights and methods, both practical and theoretical*, IEEE Signal Process., 30 (2013), pp. 106–128.

[72] C.-J. Minard, *Des Tableaux graphiques et des cartes figuratives, par M. Minard*, Thunot, Paris, 1862.

[73] M. Mirzargar, R. Whitaker, and R. Kirby, *Curve boxplot: Generalization of boxplot for ensembles of curves*, IEEE Trans. Vis. Comput. Graphics, 20 (2014), pp. 2654–2663.

[74] M. Mirzargar, R. T. Whitaker, and R. M. Kirby, *Exploration of heterogeneous data using robust similarity*, arXiv preprint arXiv:1710.02862, (2017).

[75] M. Molnár and R. Marie, *Stability oriented routing in mobile ad hoc networks based on simple automatons*, in Mobile Ad-Hoc Networks: Protocol Design, X. Wang, ed., Communications, INTECH, Jan. 2011, pp. 363–390.

[76] R. Moorhead, C. Johnson, T. Munzner, H. Pfister, P. Rheingans, and T. S. Yoo, *Visualization research challenges: A report summary*, Comput. Sci. Eng., 8 (2006), pp. 66–73.

[77] K. Mosler, *Depth statistics*, in Robustness and Complex Data Structures, C. Becker, R. Fried, and S. Kuhnt, eds., Springer, Berlin, Heidelberg, 2013, pp. 17–34.

[78] A. Nasser, D. Hamad, and C. Nasr, *Kernel PCA as a visualization tools for clusters identifications*, in International Conference on Artificial Neural Networks, Springer, 2006, pp. 321–329.

[79] S. B. Needleman and C. D. Wunsch, *A general method applicable to the search for similarities in the amino acid sequence of two proteins*, J. Molecular Biol., 48 (1970), pp. 443 – 453.

[80] M. Neumann, R. Garnett, C. Bauckhage, and K. Kersting, *Propagation kernels: Efficient graph kernels from propagated information*, Machine Learning, 102 (2016), p. 209.

[81] I. M. Pelayo, *Geodesic Convexity in Graphs (Springerbriefs in Mathematics)*, Springer-Verlag, New York, 2014.

[82] T. Pfaffelmoser, M. Reitinger, and R. Westermann, *Visualizing the positional and geometrical variability of isosurfaces in uncertain scalar fields*, in Comput. Graphics Forum, vol. 30, Wiley Online Library, 2011, pp. 951–960.

[83] W. Playfair, *The Statistical Breviary; Shewing the Resources of Every State and Kingdom in Europe*, J. Wallis, London, 1801.

[84] K. Pöthkow, B. Weber, and H.-C. Hege, *Probabilistic marching cubes*, Comput. Graphics Forum, 30 (2011), pp. 931–940.

[85] K. Potter, P. Rosen, and C. R. Johnson, *From quantification to visualization: A taxonomy of uncertainty visualization approaches*, in Uncertainty Quantification in Scientific Computing, A. M. Dienstfrey and R. F. Boisvert, eds., Springer, Berlin Heidelberg, 2012, pp. 226–249.

[86] K. Potter, A. Wilson, V. Pascucci, D. Williams, C. Doutriaux, P.-T. Bremer, and C. Johnson, *Ensemble-vis: A framework for the statistical visualization of ensemble data*, in Data Mining Workshops, 2009. ICDMW '09. IEEE International Conference on, 2009, pp. 233–240.

[87] J. D. Power, A. L. Cohen, S. M. Nelson, G. S. Wig, K. A. Barnes, J. A. Church, A. C. Vogel, T. O. Laumann, F. M. Miezin, B. L. Schlaggar, et al., *Functional network organization of the human brain*, Neuron, 72 (2011), pp. 665–678.

[88] P. S. Quinan and M. Meyer, *Visually comparing weather features in forecasts*, IEEE Trans. Vis. Comput. Graphics, 22 (2016), pp. 389–398.

[89] M. Raj, M. Mirzargar, J. S. Preston, R. M. Kirby, and R. T. Whitaker, *Evaluating shape alignment via ensemble visualization*, IEEE Comput. Graphics Applications, 36 (2016), pp. 60–71.

[90] M. Raj, M. Mirzargar, R. Ricci, R. M. Kirby, and R. T. Whitaker, *Path boxplots: A method for characterizing uncertainty in path ensembles on a graph*, J. Comput. Graph. Statist., 26 (2017), pp. 243–252.

[91] M. Raj and R. T. Whitaker, *Anisotropic radial layout for visualizing centrality and structure in graphs*, in Graph Drawing and Network Visualization, F. Frati and K.-L. Ma, eds., Springer International Publishing, Cham, 2018, pp. 351–364.

[92] J. A. Rodríguez and J. A. Rodríguez, *The March 11th terrorist network: In its weakness lies its strength*, in 25th International Sunbelt Social Network Conference, Los Angeles, 2005, Citeseer.

[93] P. J. Rousseeuw, I. Ruts, and J. W. Tukey, *The bagplot: A bivariate boxplot*, Amer. Statist., 53 (1999), pp. 382–387.

[94] G. Sabidussi, *The centrality index of a graph*, Psychometrika, 31 (1966), pp. 581–603.

[95] B. Schölkopf, A. Smola, and K.-R. Müller, *Kernel principal component analysis*, in International Conference on Artificial Neural Networks, Springer, 1997, pp. 583–588.

[96] F. Schreiber, T. Dwyer, K. Marriott, and M. Wybrow, *A generic algorithm for layout of biological networks*, BMC Bioinform., 10 (2009), p. 375.

[97] SCI Institute, 2015. AtlasWerks: An open-source (BSD license) software package for medical image atlas generation. Scientific Computing and Imaging Institute (SCI), Download from: http://www.sci.utah.edu/software/atlaswerks.html.

[98] R. Serfling, *A depth function and a scale curve based on spatial quantiles*, in Statistical Data Analysis Based on the L1-Norm and Related Methods, Y. Dodge, ed., Birkhäuser, Basel, 2002, pp. 25–38.

[99] H. L. Shang, R. J. Hyndman, and M. H. L. Shang, *Package 'rainbow'*. `https://cran.r-project.org/web/packages/rainbow/index.html`, 2016.

[100] I. Spence and S. Lewandowsky, *Robust multidimensional scaling*, Psychometrika, 54 (1989), pp. 501–513.

[101] P. A. Stott and C. E. Forest, *Ensemble climate predictions using climate models and observational constraints*, Philo.Trans. A, 365 (2007), pp. 2029–2052.

[102] Y. Sun and M. G. Genton, *Functional boxplots*, J. Comput. Graph. Statist., 20 (2011), pp. 316–334.

[103] Y. Sun and M. G. Genton, *Adjusted functional boxplots for spatio-temporal data visualization and outlier detection*, Environmetrics, 23 (2012), pp. 54–64.

[104] S. J. Teipel, A. L. Bokde, T. Meindl, E. Amaro, J. Soldner, M. F. Reiser, S. C. Herpertz, H.-J. Möller, and H. Hampel, *White matter microstructure underlying default mode network connectivity in the human brain*, Neuroimage, 49 (2010), pp. 2021–2032.

[105] J. B. Tenenbaum, V. De Silva, and J. C. Langford, *A global geometric framework for nonlinear dimensionality reduction*, Science, 290 (2000), pp. 2319–2323.

[106] M. Tennekes, E. de Jonge, P. J. Daas, et al., *Visualizing and inspecting large datasets with tableplots*, J. Data Sci., 11 (2013), pp. 43–58.

[107] J. W. Tukey, *Mathematics and the picturing of data*, in Proceedings of the International Congress of Mathematicians, R. James, ed., Vancouver, 1975.

[108] J. W. Tukey, *Exploratory data analysis*, Behavioral Science: Quantitative Methods, Addison-Wesley, Reading, Massachusetts, 1977.

[109] F. Van Ham and M. Wattenberg, *Centrality based visualization of small world graphs*, Comput. Graphics Forum, 27 (2008), pp. 975–982.

[110] R. Van Liere and W. De Leeuw, *Graphsplatting: Visualizing graphs as continuous fields*, IEEE Trans. Vis. Comput. Graphics, 9 (2003), pp. 206–212.

[111] O. Vencálek, *Depth-based classification for multivariate data*, Austrian J. Statist., 46 (2017), pp. 117–128.

[112] S. V. N. Vishwanathan, N. N. Schraudolph, R. Kondor, and K. M. Borgwardt, *Graph kernels*, J. Mach. Learn. Res., 11 (2010), pp. 1201–1242.

[113] T. Von Landesberger, A. Kuijper, T. Schreck, J. Kohlhammer, J. Van Wijk, J.-D. Fekete, and W. Fellner, Dieter, *Visual analysis of large graphs: State-of-the-art and future research challenges*, Comput. Graphics Forum, 30 (2011), pp. 1719–1749.

[114] R. T. Whitaker, *Reducing aliasing artifacts in iso-surfaces of binary volumes*, in Proceedings of the 2000 IEEE Symposium on Volume Visualization, ACM, 2000, pp. 23–32.

[115] R. T. Whitaker, M. Mirzargar, and R. M. Kirby, *Contour boxplots: A method for characterizing uncertainty in feature sets from simulation ensembles*, IEEE Trans. Vis. Comput. Graphics, 19 (2013), pp. 2713–2722.

[116] L. Wilkinson, *Visualizing big data outliers through distributed aggregation*, IEEE Trans. Vis. Comput. Graphics, 24 (2017), pp. 256–266.

[117] C. H. K. Williamson, *Vortex dynamics in the cylinder wake*, Ann. Rev. Fluid Mech., 28 (1996), pp. 477–539.

[118] L. Winner, *UFO encounters*. `http://www.stat.ufl.edu/~winner/datasets.html`, 2004.

[119] S. Wright, *The method of path coefficients*, Ann. Math. Statist., 5 (1934), pp. 161–215.

[120] V. Yang, H. Nguyen, N. Matloff, and Y. Xie, *Top-frequency parallel coordinates plots*, CoRR, abs/1709.00665 (2017).

[121] J. S. Yi, N. Elmqvist, and S. Lee, *Timematrix: Analyzing temporal social networks using interactive matrix-based visualizations*, Int. J. Human-Comput. Interaction, 26 (2010), pp. 1031–1051.

[122] W. W. Zachary, *An information flow model for conflict and fission in small groups*, J. Anthro. Res., 33 (1977), pp. 452–473.

[123] X. Zhang, C. L. Bajaj, B. Kwon, T. J. Dolinsky, J. E. Nielsen, and N. A. Baker, *Application of new multiresolution methods for the comparison of biomolecular electrostatic properties in the absence of global structural similarity*, Multiscale Model. Simul., 5 (2006), pp. 1196–1213.

[124] Y. Zuo and R. Serfling, *General notions of statistical depth function*, Ann. Statist, 28 (2000), pp. 461–482.

[125] K. A. Zweig et al., *Network analysis literacy*, MMB & DFT 2014, (2014), p. 3.

[126] M. Zwitter and M. Soklic, *Breast cancer data*. `https://archive.ics.uci.edu/ml/datasets/Breast+Cancer`, 1988.