# Computable, Robust Multivariate Location Using Integrated Univariate Ranks

by

Kelly Ramsay

A Thesis submitted to the Faculty of Graduate Studies of
The University of Manitoba
in partial fulfilment of the requirements of the degree of

Master of Science

Department of Statistics
University of Manitoba
Winnipeg

Copyright © 2017 by Kelly Ramsay

**Abstract**

This thesis concerns select methods related to multivariate nonparametric data description, especially multivariate location. It presents and provides implementations of algorithms for computing the projection median both exactly (in low dimensions) and approximately (for use in higher dimensions). The algorithms use techniques from computational geometry and Monte Carlo methods.

Further, an intuitive notion of data depth based on an average univariate ranking of points is introduced. This depth measure is shown to be quickly computable in low dimensions and easily approximated in high dimensions via Monte Carlo techniques. In addition, its theoretical properties are investigated.

Several applications of these methods are demonstrated, using both real and simulated data.

# Acknowledgment Page

I would like to thank my supervisors, Alexandre Leblanc and Stephane Durocher for their guidance, support and assistance with my research in this thesis, including the weekly brownies they bought me. I was very lucky to have had such such friendly, fun talented supervisors that made doing this work really enjoyable. I would also like to thank all of the staff that have helped me achieve so many things in my years at this university, including Mohammad Jafari-Jozani for giving me confidence in myself and excellent thesis comments and Saman Muthukumarana for explaining what a Dirichlet process was to me about twenty times. I would also like to thank Karen Gunderson for providing excellent suggestions and commentary on my thesis, especially with regard to the proofs.

I would also like to thank my friends and family for providing so much support and a nice break from school every once in a while. I would especially like to thank my parents, and boyfriend, Taylor Oxelgren, for listening to me talk about statistics all the time.

I would also like to thank the National Science and Engineering Research Council and the University of Manitoba for their generous financial assistance.

# Dedication Page

To my wonderful mother, Christine.

# Contents

# List of Figures

# Chapter 1

# Introduction

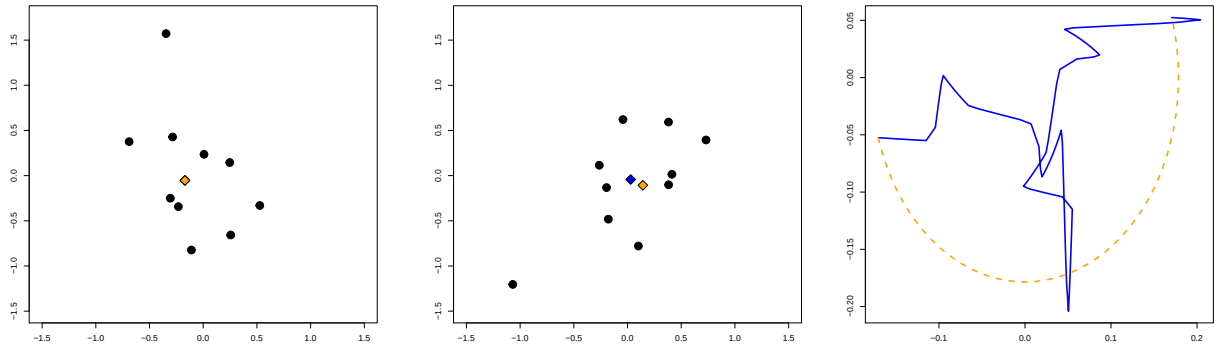
## 1.1 Motivation

Modern data analysis is becoming increasingly characterized by highly multivariate data. To draw insights from such data sets, it is necessary to either extend essential univariate tools to the multivariate setting or create new tools altogether. However, the appropriate extension of these tools to the multivariate setting is not always clear. Consider the following example of a robust (and nonparametric) measure of location.

**Example 1.**

How would you define a robust location estimator, such as the median, in the multivariate setting without making any parametric assumptions? You may say "That's easy! Take the median of each coordinate!" (This is known as the rectilinear median, vector of marginal medians or coordinate-wise median, e.g. see Section 1.3.2.) Now, examine Figure 1.1. In the left-most figure, a set of points is shown along with its rectilinear median. The subsequent figure shows the same set of points rotated by $\theta = 0.7\pi$ radians (about (0,0)) with its rectilinear median. To get the rotated point, $x_\theta$, multiply each point $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, on the left by a rotation matrix:

$$\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}.$$



(a) (b) (c)

Figure 1.1: (a) Point set with its rectilinear median in orange. (b) Point set rotated by $0.7\pi$ radians with its rectilinear median (blue) and the original rectilinear median rotated by $0.7\pi$ (orange). (c) Trajectory of the rectilinear median as the point set is rotated around 0 (blue) and trajectory of the original rectilinear median as it is rotated about 0 (orange). Notice how the trajectory of the rectilinear median jumps around.

It also shows the original median (from Figure 1.1a) rotated by $0.7\pi$ radians. Notice the median doesn't rotate with the points. Figure 1.1c shows the trajectory of the rectilinear median along $[0, \pi]$ (blue curve). The trajectory is the resulting curve from rotating the point set about every angle on $[0, \pi]$ and calculating its rectilinear median at every angle on $[0, \pi]$. Figure 1.1c also shows the path the original median follows as it is rotated along $[0, \pi]$ (orange curve). We expect the rectilinear median to rotate with the points; rotating the points does not change the shape or geometry of the cloud. This implies its trajectory should ideally follow the orange curve. Notice, however, this is not what occurs; the trajectory follows a jagged path, that can be far from the orange trajectory.

This behaviour jagged is due to the definition of the rectilinear median; it is only measuring centrality in the horizontal and vertical directions. When the point set is rotated, the pair of directions that the rectilinear median is based on changes, and thus, it can 'jump' around. By considering only two directions, we are ignoring the other (infinite) directions that could be used to measure centrality and so geometric features of the data are ignored. In fact, it has been shown that this median can fall outside the convex hull of points (Serfling, 2006).

A desirable property of an estimator is equivariance under certain transformations. This example shows that the rectilinear median is not equivariant under rotation.

Example 1 shows simple extensions of univariate data descriptors may not have desirable properties, or may not even be describing what the univariate estimator describes! This motivates the study of high quality analogues of univariate concepts, such as robust location, that accurately describe features which data analysts are interested in.

## 1.2 Desirable Characteristics of Multivariate Medians

For the remainder of the thesis let $d$ denote the dimension of a point set (number of features, number of covariates) and assume $\mathbf{X}_n$ is an i.i.d. random sample from a distribution over $\mathbb{R}^d$, where $n$ is the sample size. Let $X_{ij}$ be the $j^{th}$ coordinate of the $i^{th}$ observation from $\mathbf{X}_n$.

We have demonstrated the rectilinear median is not satisfactory in some respects. What is satisfactory? What is ideal? We of course want our median to be robust! How is robustness described in the multivariate setting? A common method is to arbitrarily corrupt points in a sample one at a time until the (multivariate) median becomes arbitrarily far from the median of the uncorrupted points. The proportion of points it takes for this to occur is known as the *finite sample breakdown point*. The limit of this proportion as $n \to \infty$ is known as the asymptotic breakdown point.

**Definition 1** (Huber and Ronchetti, 2009)**.**

The *finite sample breakdown point*, $\epsilon^*(T, n)$ of an estimator, $T$, can be defined as

$$\epsilon^*(T, n) = \frac{1}{n}\min\left\{m : \sup_{\mathbf{Y}_m} |T(\{\mathbf{X}_{n-m} \cup \mathbf{Y}_m\}) - T(\mathbf{X}_{n-m})| = \infty\right\}.$$

The *asymptotic breakdown point* is $\lim_{n\to\infty} \epsilon^*(T, n)$.

The rectilinear median (Example 1) has a finite sample breakdown of $\frac{n-1}{2n}$ (Lopuhaa and Rousseeuw, 1991). The breakdown point is the most common metric for measuring robustness

and is the one we will be concerned with in this thesis. It does not however, give the full picture so we describe some additional popular metrics. The breakdown point is a global measure of robustness in the sense that it quantifies the quality of the estimator in the case of any type of outlier. A local measure of robustness is concerned with quantifying the effect of small perturbations of the data. We need measures of local and global robustness.

We say $F^*(\epsilon, G)$ is $\epsilon$-contaminated by $G$ if $F^* = (1-\epsilon)F + \epsilon G$ where $G$ is an arbitrary distribution and $\epsilon \in (0, 1)$. The following metrics are all related to the effects of $\epsilon$-contaminated data. The *maximum bias* measures the maximum bias that can be caused by a fixed level $\epsilon$ of contamination over all $G$ (Hampel et al., 1986). In other words, it is the maximum distance, over all possible $G$, an estimator is pulled for fixed $\epsilon$. This is also a global measure of robustness. The *contamination sensitivity* is the maximum effect on the estimator of an infinitesimally small $\epsilon$-contamination, relative to $\epsilon$ (He and Simpson, 1993). This is both a local and global measure of robustness. The *influence function* sets $G = \delta_x$; a point mass at $x \in \mathbb{R}^d$, and measures the effect of an infinitesimally small perturbation toward $x$, relative to the size of the perturbation (Hampel et al., 1986). This is a measure of local robustness as it depends on the $x$ and concerns a small perturbation. The *gross error sensitivity* maximizes the norm of the influence function in $x$, and thus, measures the maximum relative effect of a small contamination toward a single point (Hampel et al., 1986). This is both a global and local measure of robustness. These metrics, including the breakdown point, are often hard to derive and so results on them are often incomplete for multivariate medians. These resources contain an overview of these metrics: Hampel et al. (1986); He and Simpson (1993) and these contain related results on multivariate medians: Chen and Tyler (2002); Zuo (2004).

Further, the field of computer science has properties for measures of location that also qualify as measures of robustness. These properties are not often investigated in the statistical literature but are useful and provide a different perspective. First consider the idea of an *$\epsilon$-perturbation*, which is similar to an $\epsilon$-contamination. Note, given $\epsilon > 0$ and a point set $\mathbf{X}_n$, a function $f : \mathbf{X}_n \to \mathbb{R}^d$ is an *$\epsilon$-perturbation* on $\mathbf{X}_n$ if, for all $X_i \in \mathbf{X}_n$, $||X_i - f(X_i)|| \leq \epsilon$

(Durocher and Kirkpatrick, 2009). Let $\mathbb{W}(\mathbf{X}_n, \epsilon)$ represent the set of all $\epsilon$-perturbations on $\mathbf{X}_n$.

**Definition 2** (Durocher and Kirkpatrick, 2009)**.**

An estimator, $T$, is *k-stable* if for all $\epsilon > 0$ and $f \in \mathbb{W}(\mathbf{X}_n, \epsilon)$, $k||T(\mathbf{X}_n) - T(f(\mathbf{X}_n))|| < \epsilon$.

In plain language, stability describes the relative magnitude of the perturbation of the estimator, relative to the magnitude of a perturbation of the points. For example, the mean vector (e.g. Section 1.3.1) is 1-stable (Durocher and Kirkpatrick, 2009). This means that if the any or all of the points move $\epsilon$ distance in any direction, the mean vector will not move more than $\epsilon$ distance. Note that this stability can be thought of as a measure of local robustness, as it bounds the effect of small perturbations. Another measure of robustness is *monotonicity* which says, colloquially, that if the points are moved in a certain direction, the estimator does not move in the opposite direction. This seems trivial but in fact, does not always hold; consider the following example.

**Example 2.**

Consider another measure of location, specifically the centre of the smallest enclosing circle (The centre of the smallest circle that contains all of the points). Consider the set of points in Figure 1.2, the red point is the centre of the smallest enclosing circle of the three black points. Now when we perturb two of the black points to the blue points (to the right and up or down), the centre of the smallest enclosing circle moves to the left!

Another important property is *transformation equivariance*, especially with respect to linear transformations. Equivariance with respect to linear transformations implies the estimator is independent of the coordinate system. Dependence on the coordinate system is not desirable for several reasons. One example is demonstrated in Figure 1.1; this type of dependence can cause the estimator to ignore certain geometric features of the data (Liu et al., 1999; Zuo and Serfling, 2000; Serfling, 2006). Secondly, if an estimator is not equivariant

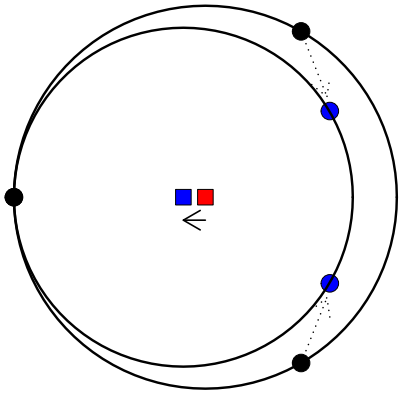


Figure 1.2: The centre of the smallest enclosing circle fails to have the monotonicity property. The red point is the centre of the smallest enclosing circle of the black points. When two of the black points move to the blue points (right and up or down), the centre of the smallest enclosing circle moves left.

under non-uniform scaling, then the unit of measurement could play a significant role in the statistical analysis. For instance, Zhang and Pan (2016) give an example where they use an estimator not equivariant under non-uniform scaling for the test statistic in a hypothesis test. They show by simply changing the scales of the axes they can make the p-value range from almost 0 to almost 1. Clearly this is not acceptable. Two types of equivariance are mainly discussed in the literature, *affine equivariance* and *similarity equivariance*. Affine transformations include all linear transformations whereas similarity transformations are a subset of linear transformations, including combinations of rotation, reflection, translation and uniform scaling. Another associated property is called *dimension consistency*: when the point set lies in a $d-1$ flat, the $(d-1)$-dimensional median definition should coincide with the $d$-dimensional definition (Durocher et al., 2017).

Further, we would like our estimator to be *computable in any dimension*. Computational difficulties can create a barrier between theoretical study of estimators and actual application. As will be seen, many existing extensions of nonparametric multivariate location are based

on depth measures whose associated medians are notoriously difficult to compute.

## 1.3 Basic Multivariate Location

### 1.3.1 The Mean Vector

The sample mean vector is generally the most popular location estimator.

**Definition 3.**

The *sample mean vector* of $\mathbf{X}_n$, denoted $\bar{\mathbf{X}}$, is defined as

$$\bar{\mathbf{X}} = \left( \frac{1}{n} \sum_{i=1}^{n} X_{i1}, ..., \frac{1}{n} \sum_{i=1}^{n} X_{id} \right)'.$$

The sample mean vector is equivariant under affine transformations. The sample mean vector, like the univariate sample mean, is not robust; it is easily influenced by corrupted data. It can be heavily influenced by a single observation which is reflected in its finite sample breakdown point of $\frac{1}{n}$. If we suspected our sample was corrupted, the sample mean vector is not a good estimator to choose. It is also 1-stable (Durocher and Kirkpatrick, 2009). Lastly, we can compute the sample mean very quickly in $O(nd)$ time (this is quite fast when it comes to multivariate medians).

### 1.3.2 The Rectilinear Median

The rectilinear median, described above and whose definition is given below, is another common location estimate.

**Definition 4.**

The *rectilinear median* of $\mathbf{X}_n$, denoted $R(\mathbf{X}_n)$, is the vector of component-wise medians; sample medians taken over each column of $\mathbf{X}_n$,

$$R(\mathbf{X}_n) = (\text{med}(X_{\cdot,1}), ..., \text{med}(X_{\cdot,d}))',$$

where $X_{\cdot,j}$ denotes the $j^{th}$ column of $\mathbf{X}_n$.

The rectilinear median is not similarity equivariant but it is equivariant under non-uniform scaling; its value relies on the chosen coordinate system. It is much more robust than the sample mean; it has the maximal finite sample breakdown of $\frac{n-1}{2n}$ which approaches $\frac{1}{2}$ when $n$ is large (Lopuhaa and Rousseeuw, 1991). This means that for a large enough sample it takes almost half the points to be corrupt to make the estimator 'bad'. It is also $\sqrt{d}$-stable (Durocher and Kirkpatrick, 2009). Like the sample mean we can compute the sample median very quickly in $O(nd)$ time.

## 1.4 Data Depth Medians

The above estimators do not fit the criterion as well as one would like. This has motivated the formulation of other multivariate medians. In order to better account for the geometry of the data many other extensions of the multivariate median are based on measures of *data depth*. Depth measures give every point in the plane a depth value, based on the points in a sample or a distribution. Large values refer to largely central points. The associated median for a depth measure is the point (in the plane) of maximal depth; the most central point.

We can actually recast the rectilinear median and sample mean as maximizers of depth measures. Let $y$ be an arbitrary point in $\mathbb{R}^d$. Consider the following function,

$$O_k(y; \mathbf{X}_n) = \sum_{i=1}^{n} ||X_i - y||_k,$$

where $||\cdot||_k$ is the $k$-norm. $O_k$ can be thought of as a measure of outlyingness, in the sense that a high value of $O_k$ can indicate $y$ is far from the data cloud. Depth is inversely related to outlyingness, and thus, we can define the following depth measures:

$$D_{O_k}(y; \mathbf{X}_n) = \frac{1}{1 + O_k(y; \mathbf{X}_n)},$$

$$D_{O^2{}_k}(y; \mathbf{X}_n) = \frac{1}{1 + (O_k(y; \mathbf{X}_n))^2}.$$

It can be shown that the rectilinear median is the maximizer of $D_{O_1}$ and the mean vector is the maximizer of $D_{O_2^2}$. In other words,

$$R(\mathbf{X}_n) = \underset{y \in \mathbb{R}^d}{\operatorname{argmax}} D_{O_1} \qquad \text{and} \qquad \bar{\mathbf{X}} = \underset{y \in \mathbb{R}^d}{\operatorname{argmax}} D_{O_2^2}.$$

It is also relevant to mention that a famous location estimate known as the *Weber point* minimizes $D_{O_2}$. The Weber point is unstable in the sense of Definition 2 (Durocher and Kirkpatrick, 2009).

Many other definitions of data depth have been proposed, based on a wide variety of concepts such as half-spaces (Tukey, 1974), random simplices (Liu, 1990), zonoid regions (Mosler and Hoberg, 2006), points represented as curves (Lopez-Pintado and Romo, 2009), and many more; see Liu et al. (2008); Aloupis (2006); Zuo and Serfling (2000); Serfling (2006) for an overview. Medians based on these depth measures often have very good properties. They often have high breakdown and are affine invariant. Additionally, depth measures provide multivariate analogues of outlyingness functions, quantiles, scale, rank and order statistics. Using a depth measure allows for a "cohesive" study of these properties (Serfling, 2006). For example, the Tukey median, based on Tukey depth, has many of the properties described above.

**Definition 5** (Tukey, 1974)**.**
Let $F_n$ be the empirical distribution function of a multivariate sample $\mathbf{X}_n$. The Tukey depth of a point $y \in \mathbb{R}^d$, denoted $D_t(y; F_n)$, is the minimum number of points (in $\mathbf{X}_n$) contained in a half-space $H$ such that $y \in H$;

$$D_t(y; F_n) = \inf_{H \in \mathbb{R}^d} \{\#\mathbf{X}_n \in H\}.$$

**Definition 6** (Tukey, 1974)**.**

The Tukey median of a point set $\mathbf{X}_n$, denoted $T(\mathbf{X}_n)$, is the point which maximizes Tukey depth. If this point is not unique it is the average of all such points that maximize Tukey Depth.

The Tukey median has asymptotic breakdown of $\frac{1}{3}$ under symmetric distributions, and the additional robustness metrics are relatively simple, see Donoho and Gasko (1992); Chen and Tyler (2002). It is also affine equivariant (Tukey, 1974). However, exact algorithms for the Tukey median take $\Theta(n^{d+1})$ time in the worst case and $\Theta(n^d)$ time on average (Dyckerhoff and Mozharovskyi, 2016). Compare this to computing the sample mean vector or the rectilinear median. The exponential dependence on $d$ makes exact computation impractical in high $d$ scenarios.

Even though they usually are highly robust and affine equivariant, many of these depth measures have complex definitions that are not intuitive at first and/or they are difficult or impossible to compute in high dimensions, which is precisely where they are needed the most (Liu et al., 2008). Thus, there is a need to find efficient algorithms for computing or approximating these medians in higher dimensions. Another alternative is to define a new depth measure (and by doing so, a new median) in such a way as to ensure it is computable in high dimensions. Chapter 2 takes the first approach and Chapter 3 takes the second.

Specifically, in this thesis we discuss a robust measure of location, the *projection median*, and walk through the efficient computation of it, which we have implemented in the R software. We then introduce a new depth measure, *integrated rank-weighted depth*, and its associated location estimator that satisfies the above criteria. Chapter 4 demonstrates an application of the resulting estimators by applying them to very high dimensional data. Many ideas in this thesis come from blending ideas from the field of computational geometry with statistics; the problem is approached from a geometric point of view.

# Chapter 2

# The Projection Median

## 2.1 The Projection Median

Though the projection median is defined for any dimension, it is helpful to start with simply defining it in two dimensions. Consider a 2-dimensional sample of size $n$, $\mathbf{X}_n$, and a unit vector, $u$; see Figure 2.1. Now, say we project the points onto the line passing through the origin and parallel to $u$ as in Figure 2.1, we will call the projected set $\mathbf{X}_{n,u}$. We can consider $\mathbf{X}_{n,u}$ to be a 1-dimensional point set with possible duplicates (a multiset) with ordering along $u$. In Figure 2.1 notice the green point's projection is the 1-dimensional median of $\mathbf{X}_{n,u}$. We denote this point $x_u^*$, the point whose projection is the median of $\mathbf{X}_{n,u}$. In the case where $n$ is even $x_u^*$ refers to the point of rank $\frac{n}{2}$.

**Definition 7** (Durocher et al., 2017)**.**

Let $m(x_i)$ be the multiplicity of $x_i \in \mathbf{X}_n$, $S^{d-1}$ be the $d-1$ unit hypersphere and let

$$S_{x_i}^{d-1} = \{u \ : \ u \, \epsilon \, S^{d-1} \ \& \ x_u^* = x_i\}.$$

The *projection median*, $M(\mathbf{X}_n)$, of a sample $\mathbf{X}_n$ whose elements are from $\mathbb{R}^d$ is

$$M(\mathbf{X}_n) = \sum_{x_i \in \mathbf{X}_n} w_i x_i, \qquad \text{with} \qquad w_i = \frac{\int_{S_{x_i}^{d-1}} m(x_i) du}{\sum_{i=1}^{n} \int_{S_{x_i}^{d-1}} m(x_i) du}. \tag{2.1}$$

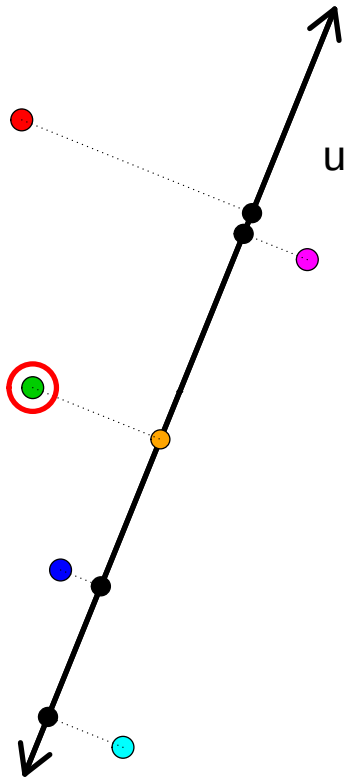



Figure 2.1: Data projected onto a line through the origin parallel to the vector $u = (\cos(1.236), \sin(1.236))$, note $x_u^* = x_{green}$.

Intuitively, the projection median is a weighted average of the points in $\mathbf{X}_n$, with the weights chosen proportional to the multiplicity of $x_i$ and the proportion of unit vectors for which $x_i = x_u^*$ over all unit vectors $u$ in $\mathbb{R}^d$. The projection median was first introduced in $\mathbb{R}^2$ by Durocher and Kirkpatrick (2009), then extended to $\mathbb{R}^d$ by Basu et al. (2011) and the weighted mean representation, which is the definition presented above, was introduced by Durocher et al. (2017). We now introduce an equivalent definition of the projection median that is useful for proving certain properties.

**Definition 8.**

The *projection median*, $M(F)$, of a probability distribution $F$ on $\mathbb{R}^d$ satisfies

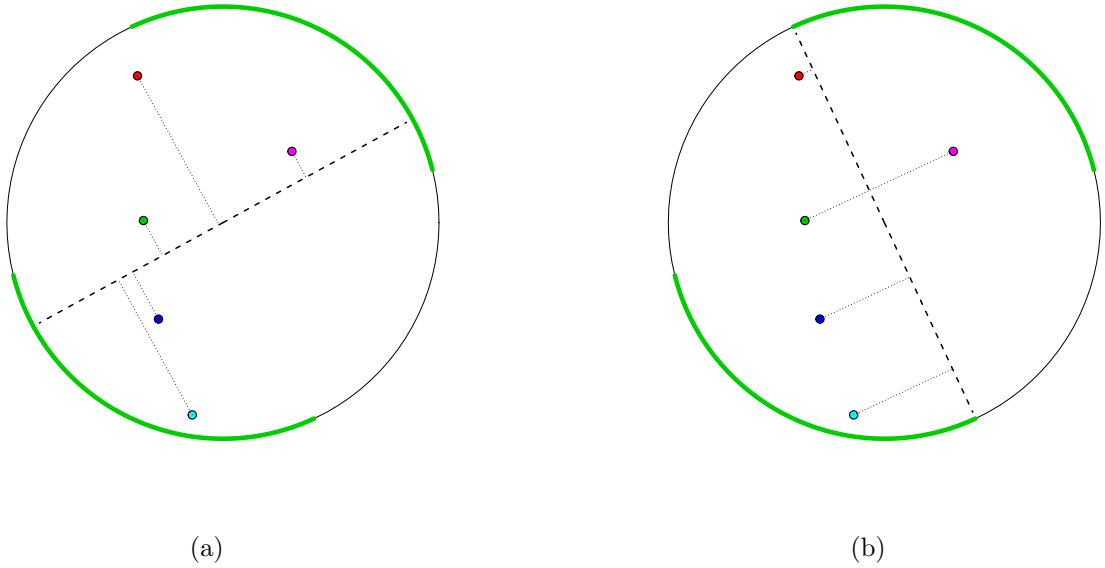
$$M(F) = \frac{d}{V_d} \int_{S^{d-1}} F_u^{-1}(\frac{1}{2}) u \, du \tag{2.2}$$

where $V_d = \int_{S^{d-1}} 1 du$ and $F_u^{-1}$ is the quantile function related to the distribution of $u'X$ if $X \sim F$.

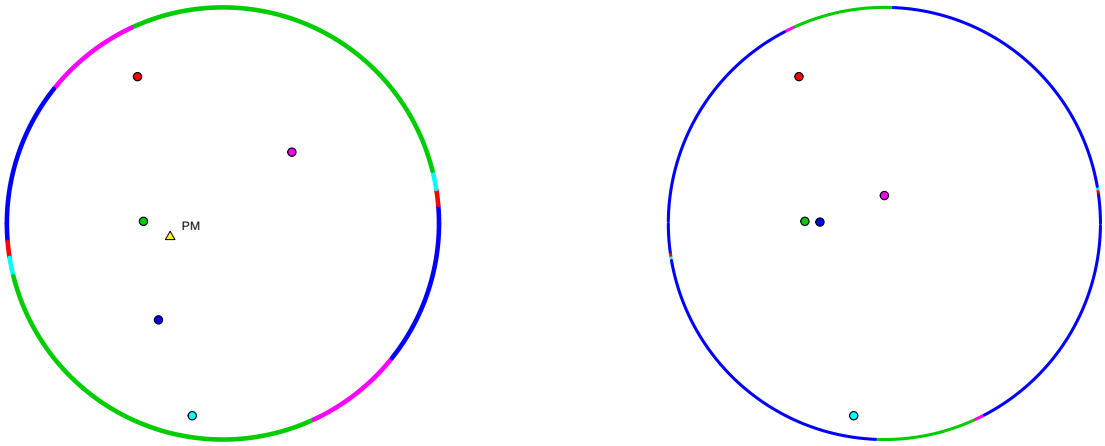
The above definition can be used in both continuous and discrete cases. The sample version of the above definition replaces $F$ with $F_n$; it can be thought of as a discrete distribution with equal probability placed on each point, and is equivalent to Definition 7.

### 2.1.1 Properties

The projection median has many desirable properties. Clearly it coincides with the one-dimensional median. It is also equivariant under similarity transformations (Durocher and Kirkpatrick, 2009). It is very robust; the following theorem shows that it has the highest possible asymptotic and finite sample breakdown.



(a) (b)




(c) (d)

Figure 2.2: (a) The projection median(yellow) of $n = 5$ points when $d = 2$. The unit circle represents all possible directions or unit vectors onto which we can project the points. The green section of the circle represents the set $S_{x_{green}}$ ($x_{green}$ is the green point), you can see when we project into the green section, $x_{green}$'s projection is indeed the univariate median. (b) When we project onto the boundary of $S_{x_{green}}$ and $S_{x_{pink}}$ the projections of these two points become equal. If we move $u$ into $S_{x_{pink}}$, the pink point will become the median. (c) The unit circle is color coded to show which of the sets $\{S_{x_{green}}, S_{x_{blue}}, ...\}$ each unit vector, $u$, belongs to. (d) Notice here that although the same 3 points are still central, not all have large weights; small weights do not indicate outliers.

**Theorem 1** (Basu et al., 2011)**.**

The projection median satisfies

$$\epsilon^*(M,n) = \frac{\lfloor \frac{n}{2} \rfloor - 1}{n},$$

and

$$\lim_{n\to\infty} \epsilon^*(M,n) = \frac{1}{2}.$$

Not only does it have high breakdown but outlying points also have small weights, this can be seen in Figure 2.2d. In fact, the weight of a point moving toward infinity along any ray approaches 0. This property is analogous to the vanishing at infinity property associated with medians based on statistical depth functions (Zuo and Serfling, 2000). Though the weights can be small, they are still positive for all points, thus, the projection median uses information from the whole data set (Basu et al., 2011). It is also a stable, in the sense of Definition 2, alternative to the Weber point (see Chapter 1).

**Theorem 2** (Durocher and Kirkpatrick, 2009, Basu et al., 2011)**.**

$M$ is at least $\frac{\pi}{(d)B(d/2,1/2)}$-stable, where $B(\alpha,\beta)$ denotes the Beta function.

On the topic of robustness, it is important to note that though outliers have small weights, the converse is not always true; small weights do not always indicate outlyingness. The situation can arise in which a point is deep, in the sense that its projection is often close to the projected median, but it is not often $x_u^*$ itself. See Figure 2.2c-2.2d for an example of this. This leads to the conclusion that weights cannot be interpreted as measures of depth. One could however use them to identify a subset of the data that contains the outliers and some additional points. That being said, when there is a focus on outlier detection, we recommend using a depth measure such as the one in Chapter 3.

Many depth based location estimators are difficult to compute. Unlike most estimators such as the Tukey Median and Stahel-Donoho estimator, see Donoho and Gasko (1992); Liu et al. (2008); Tukey (1974) computing the projection median is not an optimization problem. A weighted average is often less computationally expensive than a global optimum; it is also better suited to Monte Carlo approximation algorithms (Durocher et al., 2014; Durocher and Kirkpatrick, 2009; Basu et al., 2011). Techniques established in computational geometry and Monte Carlo algorithms make the projection median fairly straightforward to compute (exactly or approximately) in any dimension. It can be computed exactly in $O(n^{\frac{4}{3}+\epsilon})$ time in $\mathbb{R}^2$ and $O(n^{\frac{5}{2}+\epsilon})$ time in $\mathbb{R}^3$ (Basu et al., 2011). Algorithms that achieve these times are described in Sections 2.2 and 2.3. For arbitrary $d$, the projection median can be computed in $O(n^{d(1-\frac{\delta_d}{d+1})+\epsilon})$ time for some $\delta_d > 0$ (Basu et al., 2011). Implementation and computation in a reasonable time is impractical for these algorithms for $d > 3$, due to the complexity of the algorithms and the exponential times. However, approximation algorithms exist. There are two algorithms, outlined by Durocher et al. (2017), which are based on simple Monte Carlo techniques that run in $O(mnd)$ time, where $m$ denotes the number of replicates generated in the Monte Carlo experiment. These are described in Section 2.4.

## 2.2 Exact Algorithm in $\mathbb{R}^2$

This work seeks a practical implementation for efficient computation of the projection median which, at present, does not yet exist to the author's knowledge. The algorithm presented can be used to compute the projection median exactly and efficiently in $\mathbb{R}^2$, using a kinetic data structure called a kd-heap. An R function that uses this algorithm was implemented as part of the thesis work and is is available on Github (Ramsay, 2017).

Kinetic data structures, first described by Basch et al. (1999), were originally designed for 'moving' data. These are data that are changing (usually over a period of time) and thus,

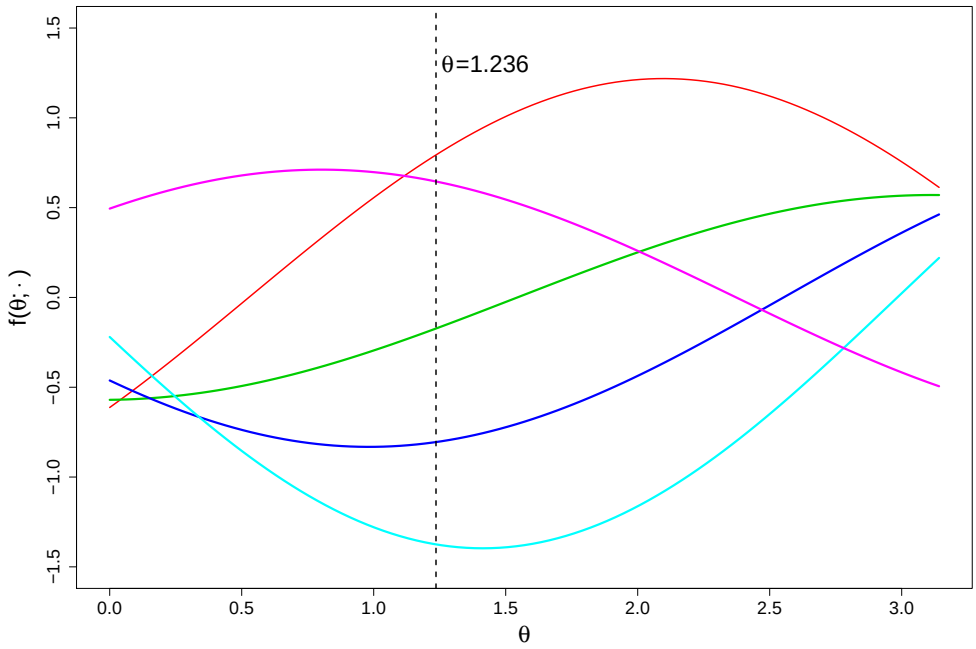



Figure 2.3: Trajectories of a point set in 2D, with the angle of the unit vector from Figure 2.1 labelled as a vertical line. Notice that the ordering and relative spacing of curves at this line is the same as that of the projections of the points in Figure 2.1. The median-level of the trajectories is highlighted with a dotted line.

can be expressed as continuous trajectories such as curves or functions. To use this structure we must transform our problem into a moving data problem. The continuous changes in the positions of points in $\mathbf{X}_{n,u}$ as $u$ rotates naturally lends itself to such an interpretation.

Consider a point, $x_i = (x_{i1}, x_{i2}) \in \mathbf{X}_n$, and a unit vector represented in spherical coordinates, $u_\theta = (\cos\theta, \sin\theta)$. We can define this point's *trajectory* as

$$f(\theta; x_i) = x_{i1}\cos\theta + x_{i2}\sin\theta, \tag{2.3}$$

which is the signed magnitude of its projection onto $u_\theta$. Each trajectory is 'moving' in $\theta$ and is periodic, starting at 0 with a period of $2\pi$. In fact, we need only consider $\theta$ between 0 and $\pi$, since $\pi$ to $2\pi$ is simply the reflection of 0 to $\pi$. Figure 2.3 shows the trajectories of the point set Figure 2.1. The ordering of curves at a vertical line drawn at any $\theta_0$ is the ordering of points along $u_{\theta_0}$. For example, Figure 2.1 shows the point's projection onto $u_{1.236}$. Drawing a vertical line in Figure 2.3 at $\theta = 1.236$ shows that the green point is the median.

The point whose trajectory at a fixed $\theta_0$ is the median corresponds to $x^*_{u_{\theta_0}}$. Further, since the magnitude of projections is preserved, the sum of the lengths of the intervals for which a point $x_i$'s trajectory is the median is directly proportional to $w_i$.

In summary, the set of median curve segments (or pair of near median curve segments if $n$ is even) on $[0, \pi)$ contains all relevant information needed to calculate $M(\mathbf{X}_n)$. This sequence of median curve segments is called the *median-level* of the trajectories. For every $\theta$, at most half of the trajectories lie above the median-level and at most half lie below it. Figure 2.3 shows the median-level of the trajectories associated with Figure 2.1. As $\theta$ increases from 0 to $\pi$, we record which point's trajectory is associated with the median-level. Imagine sweeping a vertical line from left to right and recording which curve is the median curve. We have transformed our problem into a moving data problem, even though our data are not actually moving at all!

We will make use of the fact that this set of curves are *pseudo-lines*. Pseudo-lines are curves that are similar to lines in the following sense: two pseudo-lines cross each other exactly once (in the range considered). It is easy to show that the set of trajectories are pseudo-lines for $\theta \in (0, \pi]$. For more on pseudo-lines and levels, see Edelsbrunner (1997).

As mentioned above we can use kd-heaps to keep track of the median-level, but first it is easier to understand how to use a kd-heap to keep track of the upper-envelope, or top-most level of curves, as seen in Figure 2.4. This algorithm is attributed to Basch et al. (1999). We start by describing how to initialize the kd-heap for the upper envelope. To avoid unnecessary details and confusion, the use of a kd-heap is discussed only in the context of computing the projection median; we do not give a large overview of kinetic data structures. Specifically, we use a *kd-heap* which a tree structure that efficiently keeps track of the maximum curve as $\theta$ increases. To initialize the kd-heap, we start by recursively splitting the set of curves in half until we have a set of disjoint sets or groups of curves, where each set has at most two curves. The middle column in Figure 2.5 shows the groupings at each recursion level for the example trajectories in Figure 2.3. Each member in this last set of groups is represented as a

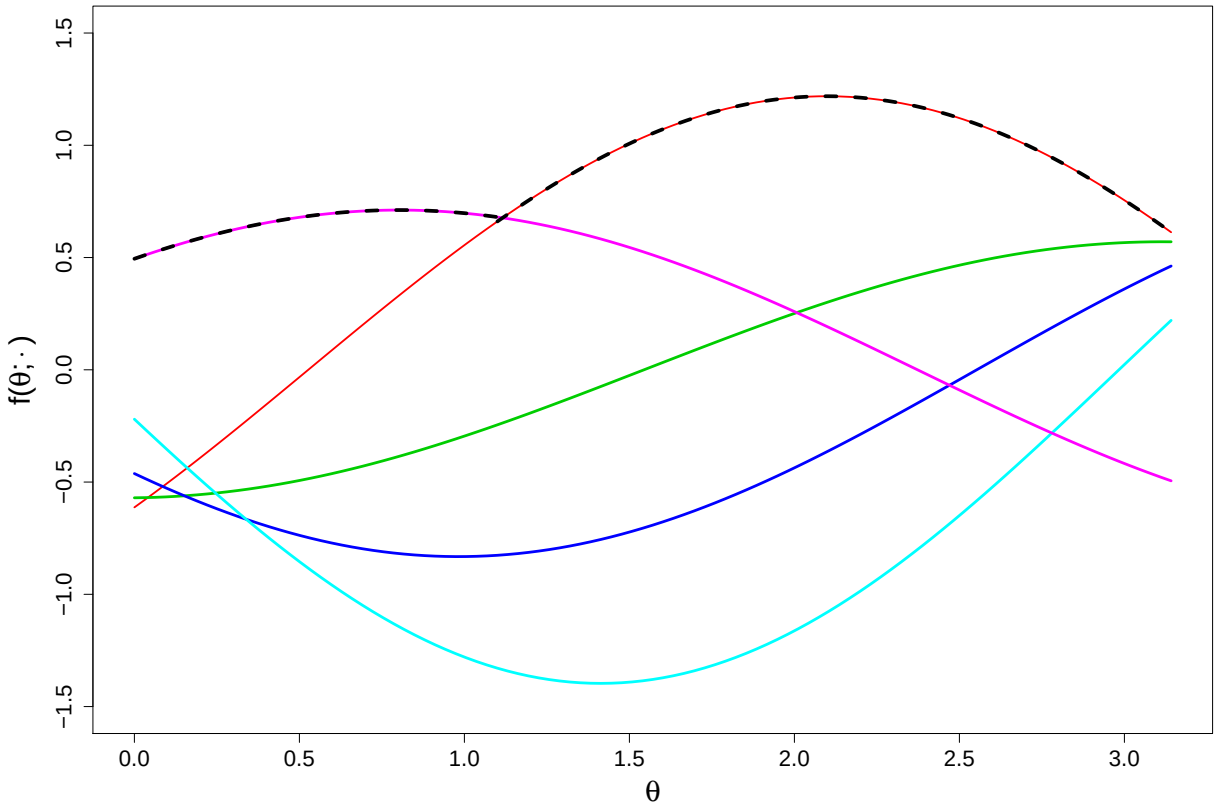



Figure 2.4: The upper envelope of the set of curves.

leaf node in the kd-heap (a leaf node is a node on the bottom of the heap) and members in a group are siblings (share the same parent; are connected to the same node above). Then, at the bottom level, each pair of siblings are 'pitted against each other', with whichever curve is above the other at $\theta = 0$ becoming the parent (node above the group). Recall there are only a maximum of two curves in each group at this recursion level, so each group has only one parent. For example, the bottom left pair of siblings in Figure 2.5 are $A$ and $C$, and $C$ is the parent because $C$ is above $A$ at $\theta = 0$ (can be seen on the left in Figure 2.5). In the kd-heap the parent is always the curve that is currently above the other curve in the pair of siblings. When the two curves in the group 'face off' (i.e. when the parent is identified) a new *certificate* is generated.

A *certificate* is associated with two curves. It is an object that contains the two curves'

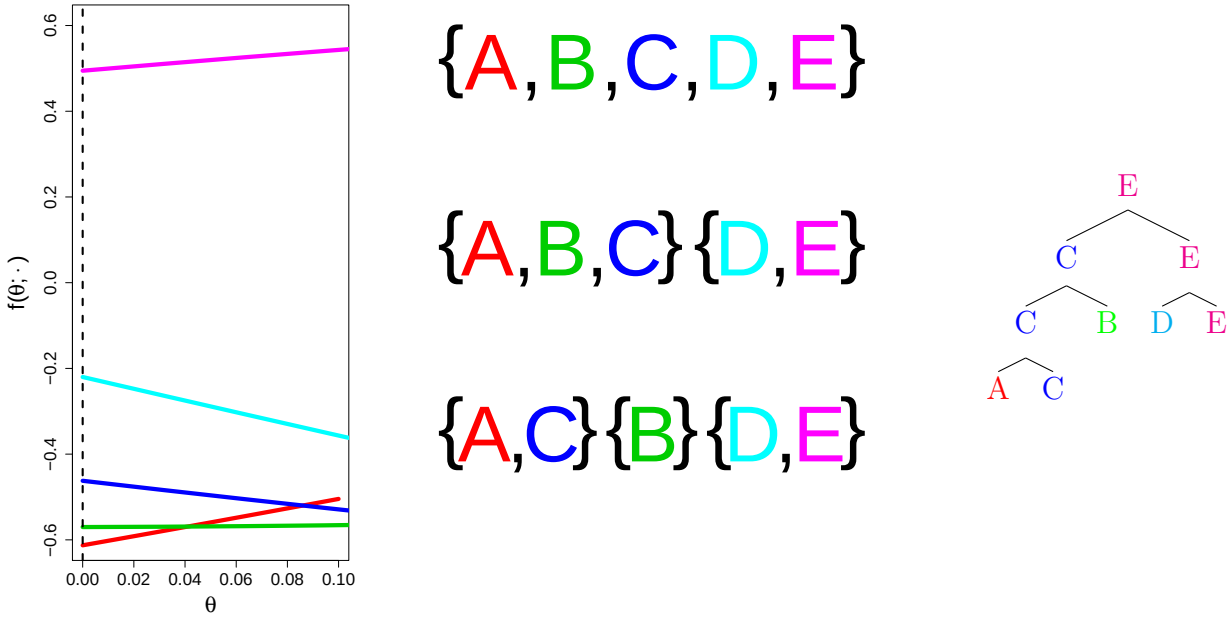



Figure 2.5: A kd-heap built from the curves in Figure 2.3, the trajectories are shown on the left for $\theta \in (0, 0.06)$. The groupings at each recursion level are shown in the middle, and the heap is shown on the right.

current relative position as well as when (at which angle, $\theta$) those two curves will switch relative position. The switching time is referred to as the certificate's expiry. Since we are dealing with pseudo-lines each certificate has a unique expiry. Continuing with the $A$ and $C$ example, the certificate would say '$C$ is above $A$' and would expire at $\theta = 0.086$. In other words, $A$ and $C$ intersect at $\theta = 0.086$, so at that time $C$ is no longer above $A$. If the node has no sibling, it is the automatic parent, see $B$ in Figure 2.5, and no certificate is generated. Since it is redundant to have a node in a kd-heap with one child that is the same as the parent, we can simply delete that child. Note that in Figure 2.5 $B$ has no child.

Now, for each level of the heap we recursively determine the parent for each pair of siblings in the same manner. As the recursion moves upward, at each recursion level the 2 champions from the previous level face off against one another again, the winner is the parent and a certificate is generated for those 2 curves. In Figure 2.5, $C$ would face off against $B$ and $D$ would also face off against $E$. Eventually there is just one pair of siblings and the winner is

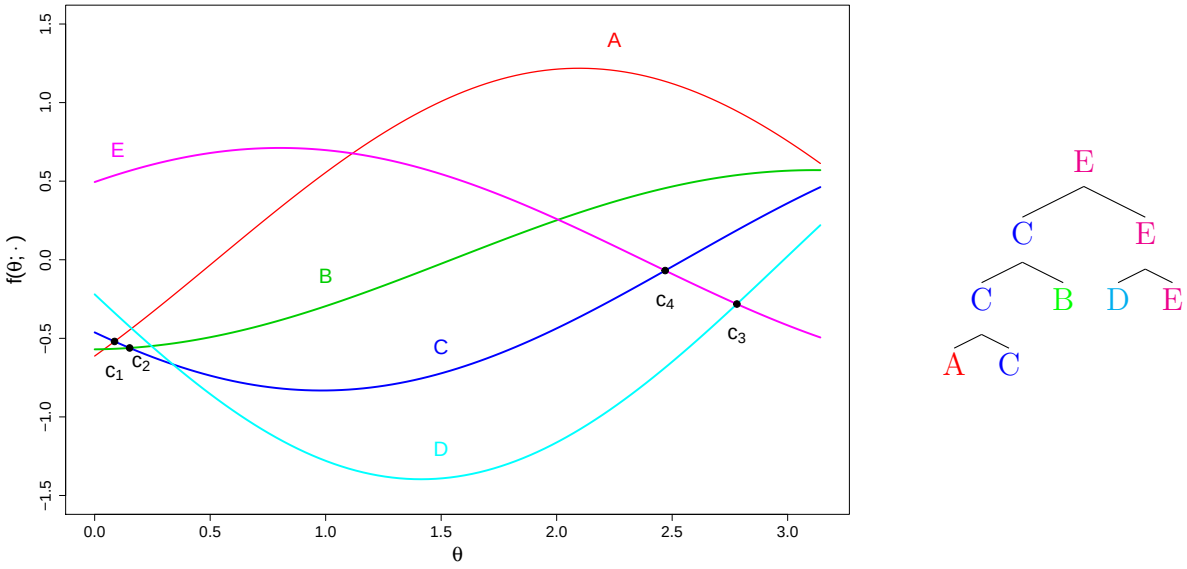



Figure 2.6: Trajectories with certificates generated by the heap to the right. The expiry of each certificate is labelled, $c_1$ will be the first to expire.

the root of the heap; see Figure 2.5. Initially, the root of the heap is the maximum curve at $\theta = 0$.

As certificates are generated, they are stored in an event queue (a priority queue) in order of expiry. Following the line sweeping analogy, the first event in the queue is the first one hit by the line, where a certificate is placed on the graph at the point where the 2 curves it concerns cross. This can be seen in Figure 2.6, the certificate for each pair of siblings are labelled on the trajectory graph. The event queue would have the following order: $\{c_1, c_2, c_4, c_3\}$.

After the heap is initialized, the algorithm works by processing the events in the queue and the heap is maintained as events occur. In terms of the line sweeping analogy, this is when the line starts moving. The line pauses when a certificate expires. At that point, the event is processed by making necessary changes to the heap and queue. These changes are summarized in the flow chart in Figure 2.7. The algorithm ends when the line reaches $\pi$. The first event in the example in Figure 2.6 would be the expiry of $c_1$.

The algorithm would complete the following tasks. $c_1$ is dequeued. $C$ is no longer above

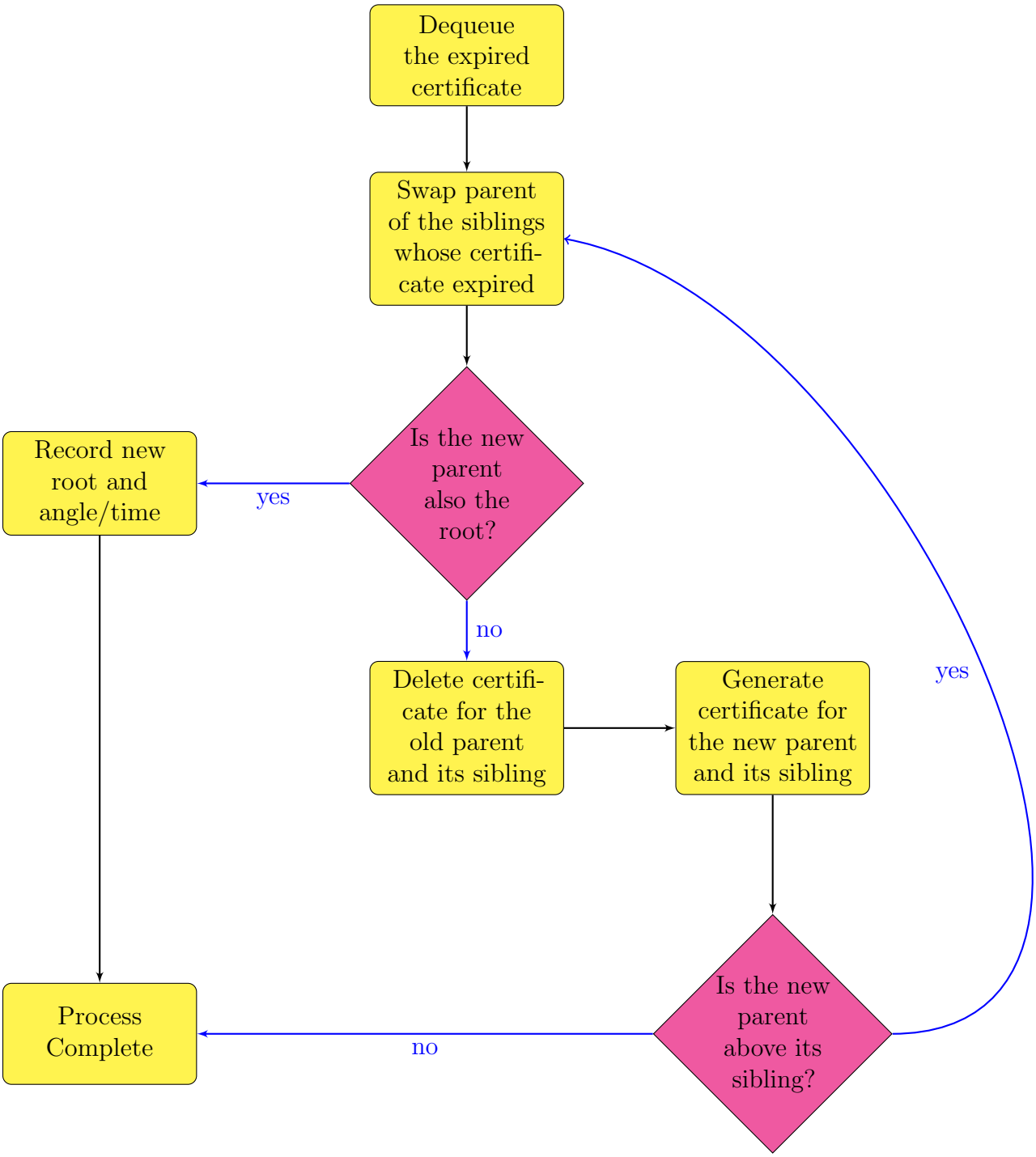



Figure 2.7: Flow chart for processing an event.

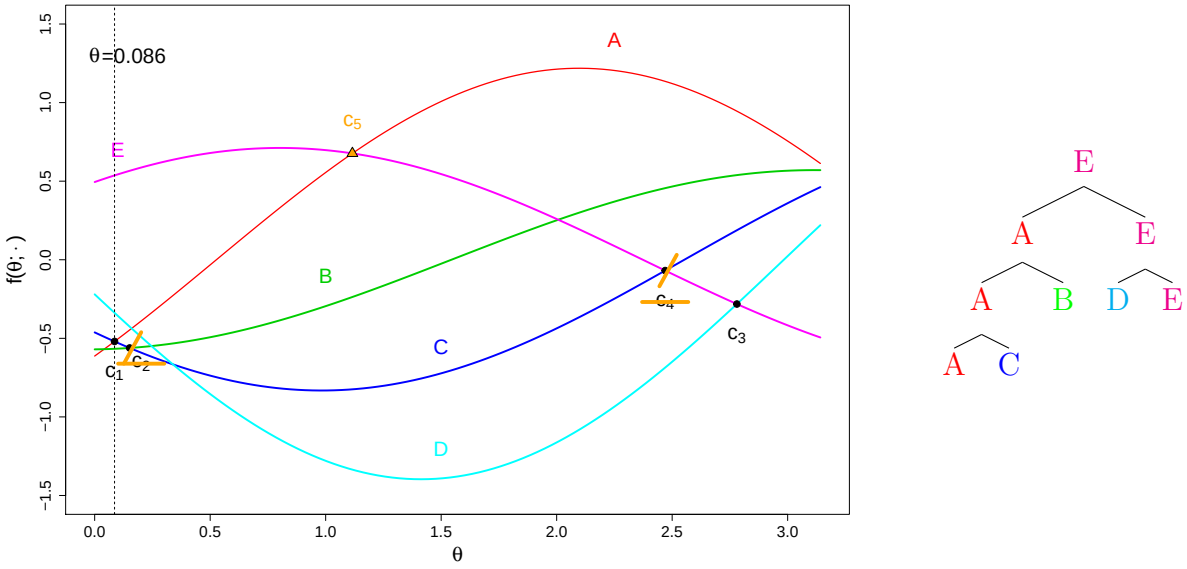



Figure 2.8: First event processed, the vertical line is at the first event $c_1$. $A$ replaces $C$ on two levels, since $A$ is now above $C$ and thus, is also above $B$. $E$ is still above $C$ and is thus above $A$ and so $A$ does not replace $E$. $c_5$ is generated for $A$ and $E$. No certificate is generated for $A$ and $B$ since they have already crossed each other. $c_4$ is removed with the elimination of the pair of children $B$ and $C$.

$A$, so $A$ swaps with $C$. $A$ has not replaced the root so we move to the right of the flow chart. We delete the certificate for $C$ and $B$, $c_2$, since they are no longer siblings. $A$ swapping with $C$ has produced a new set of siblings: $A$ and $B$. We must generate a certificate for this pair. In this case the vertical line has already swept past the intersection of $A$ and $B$, so the relative position of these curves will not change for the remainder of the algorithm; the certificate does not expire so we can discard it. Now, since $C$ is still above $B$, so is $A$. Thus, we go back to the top of the flow chart and swap $A$ with $C$ again. $C$ and $E$ are no longer siblings so their certificate, $c_4$, is deleted. $c_5$ is generated for $A$ and $E$ and is placed in the queue. The heap maintenance is now complete and the next event is then processed. All the changes to the heap after $c_1$ is processed can be seen in Figure 2.8. Events are processed until there are no events left in the queue, at this point we are left with the upper envelope of the curves. Specifically we are left with a list of $N$ intervals, $I_j = (\theta_{j-1}, \theta_j)$, $j \in \{1, \ldots, N\}$, where $N$ is the number of segments in the upper-envelope. For each interval, we also have the

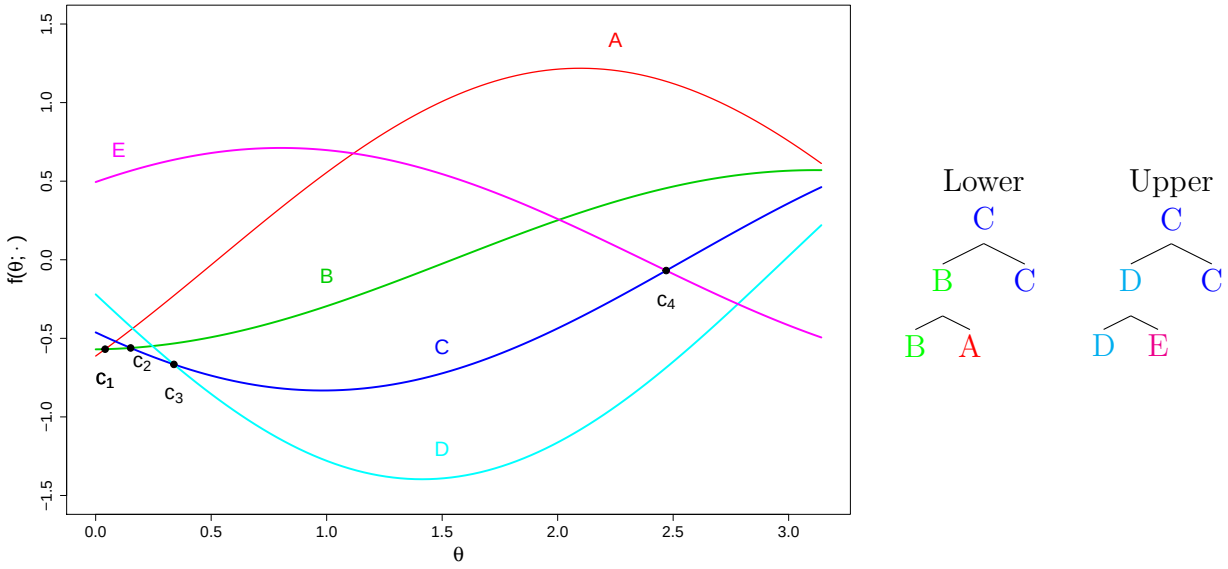



Figure 2.9: Trajectories of each point with certificates labelled, as generated from the heaps on the right. $C$ is the starting median curve and is at the top of the heap.

point whose trajectory is associated with the upper envelope in that interval. This algorithm can be easily modified to record the lower envelope or the bottom-most set of curve segments as well; the champions are simply the opposite curves.

As noted before, we are interested in the median-level and not the upper or lower envelopes. To uncover the median-level we use two kd-heaps, one for the curves above the median curve and one for the curves below. Let $L_\theta$ be the set comprising of the bottom $\lfloor \frac{n+1}{2} \rfloor$ trajectories at $\theta$ and $U_\theta$ be the set comprising of the top $\lfloor \frac{n+1}{2} \rfloor$ trajectories at $\theta$. The heap for the curves below keeps track of the maximum curve in $L_\theta$ and the heap for the curves above keeps track of the minimum curve in $U_\theta$. Note that membership in $U$ or $L$ depends on the angle $\theta$, the root of each heap at a given $\theta$ being the median curve at that angle when $n$ is odd. In the even case the two roots form the set of near median curves. In this case a certificate must be generated for the two different roots; if they swap relative positions we swap the roots of the heaps. In the odd case both heaps have the same root, so if the root of one heap changes the root of the other heap changes and must be replaced by the new root.

To initialise the algorithm we determine $L_0$ and $U_0$ and then initialize the two heaps in

the same manner as for the upper envelope. The event queues from each heap are combined into one queue. Let's continue with the example trajectories from Figure 2.3. Figure 2.9 features the initial heaps for $L_0 = \{A, B, C\}$ and $U_0 = \{C, D, E\}$ as well as the certificates generated.

As the events are processed the heaps are maintained as above, however, as indicated above, the roots must be changed in some cases. Figures 2.10 and 2.11 show the modified flow charts that includes the process for changing the root of a heap in the odd and even cases respectively. When the root of the other heap is replaced the new curve must be percolated down that heap.

Figure 2.12 shows the heaps after the first event is processed. Notice that first event is the expiry of certificate $c_1$, where $A$ switches with $B$. After $A$ swaps with $B$, $c_2$ becomes irrelevant to maintaining the heap and is removed from the queue. A new certificate, $c_5$ is generated to represent the switching of $A$ and $C$. Figure 2.13 shows the heaps after the next event (expiry of $c_5$) is processed. Here there is a root replacement. $A$ replaces $C$ at the root of the 'Lower' heap. The time is recorded as the end point for $I_1$ and starting point of $I_2$. The red point (associated with $A$) is noted to be associated with the median-level in $I_1$. $A$ then replaces $C$ on top of the upper heap. The next instance of $C$ in the next level down is replaced by $A$ and a new certificate is generated for $D$ and $A$. If $C$ appeared in lower levels it would be replaced by $A$ and new certificates generated as necessary. As usual, certificates are created if they have an expiry; no certificate is placed in the queue if the pair has already crossed. The algorithm again finishes when the queue is empty. We are again left with a list of $N$ intervals, $I_j = (\theta_{j-1}, \theta_j)$, $j \in \{1, \ldots, N\}$, where $N$ is the number segments in the median-level, or pair of near median-levels. We also have, for each interval, the point whose trajectory is associated with the median-level in that interval $x^*_j$. In the even case we have two points, $x^*_{j1}, x^*_{j2}$ associated with the near median-levels for each interval. This algorithm takes $O(n^{4/3} \log n)$ time, since propagation of a new root takes $O(\log n)$ (Basch et al., 1999) time and this happens $O(n^{4/3})$ times in the worst case (Basu et al., 2011). Below

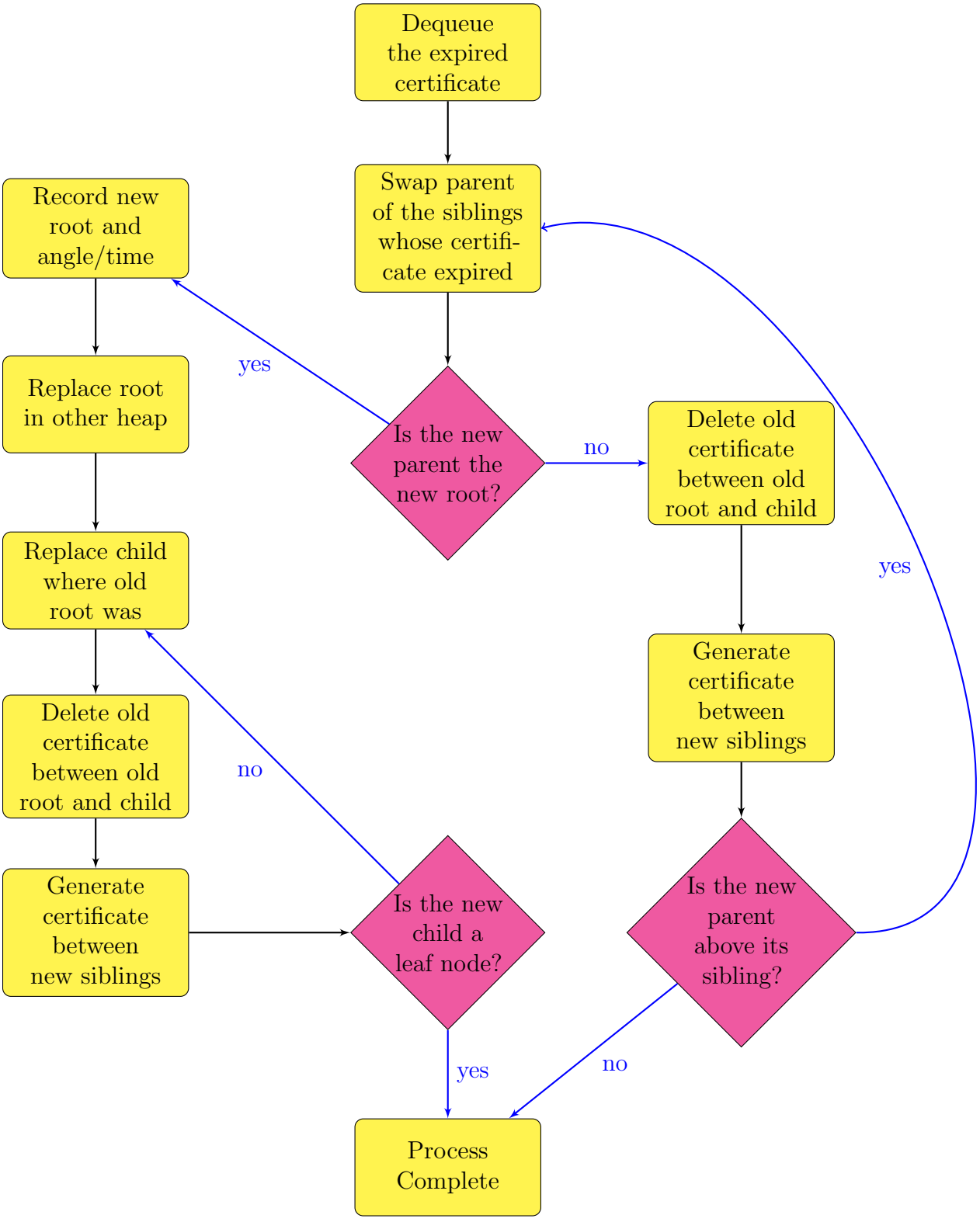



Figure 2.10: Flow chart for processing an event in the odd case of the median-level algorithm.

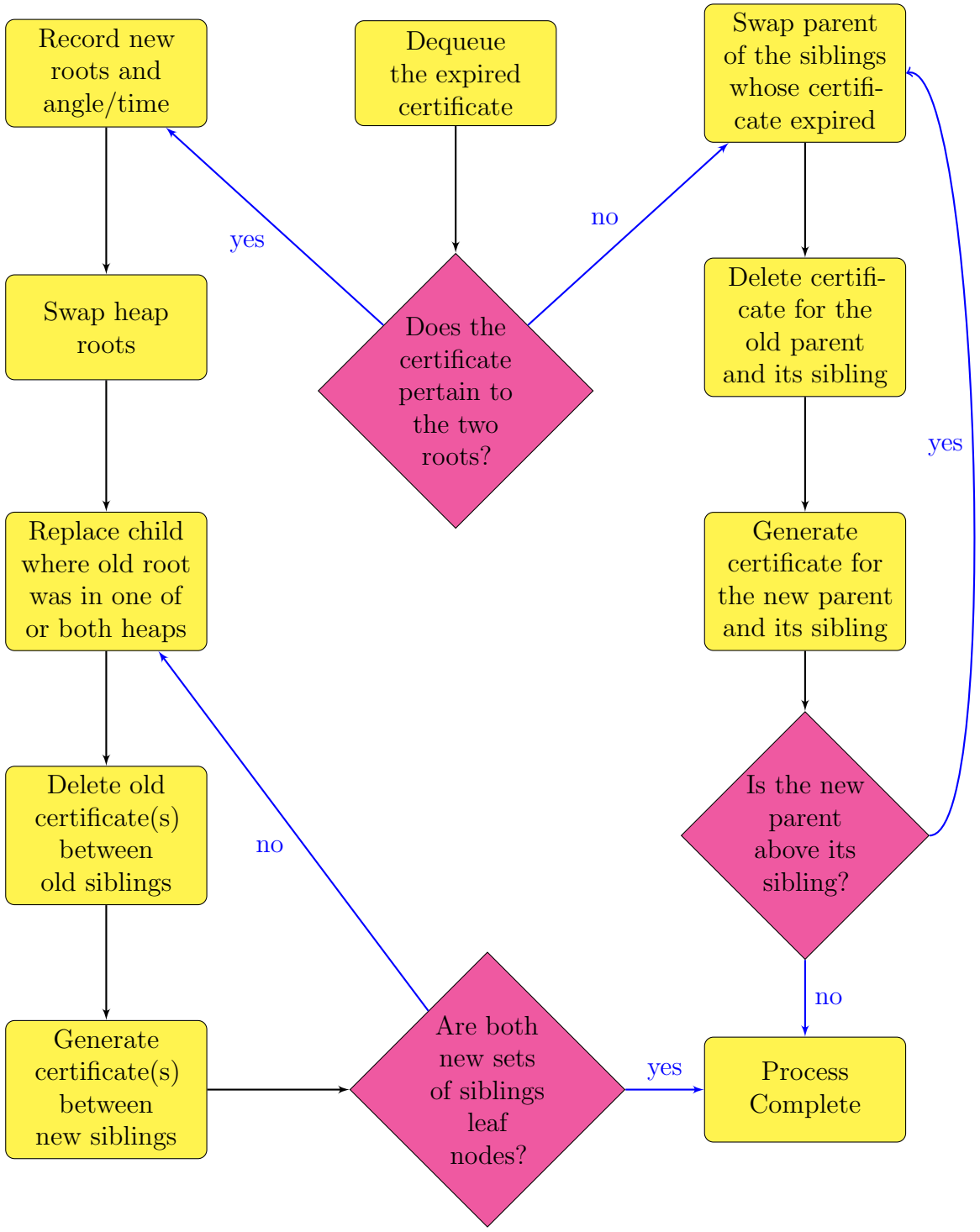



Figure 2.11: Flow chart for processing an event in the even case of the median-level algorithm.

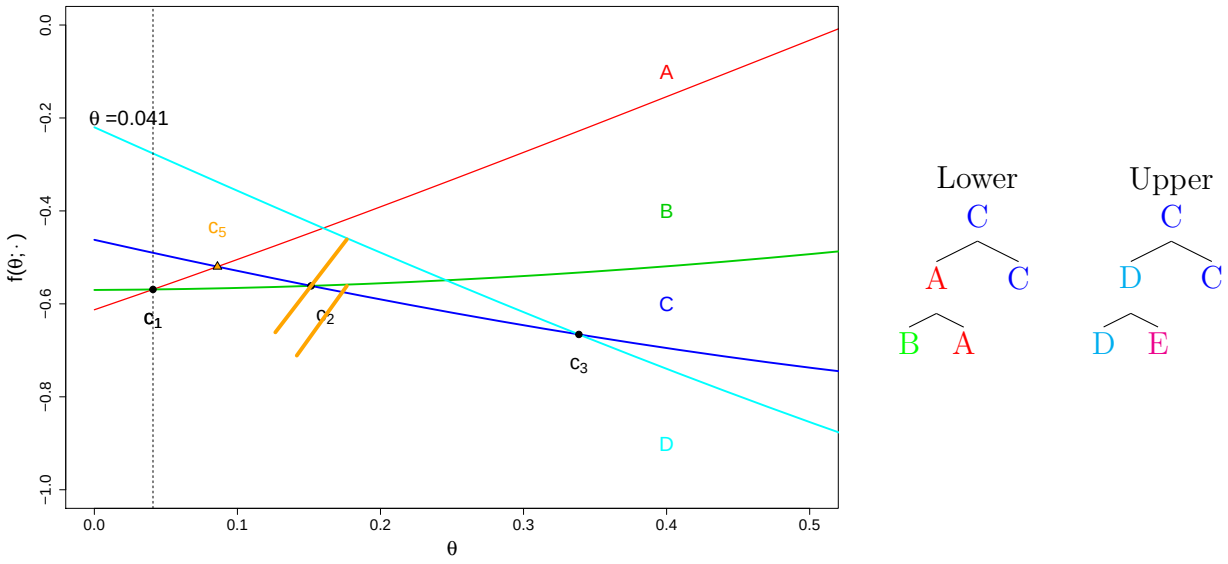



Figure 2.12: Zoomed in trajectories from Figure 2.9. Shows the certificates after first event is processed. $c_2$ is removed and $c_5$ is added. $A$ and $B$ swap, but $C$ is currently above $A$ so $C$ stays at the root.

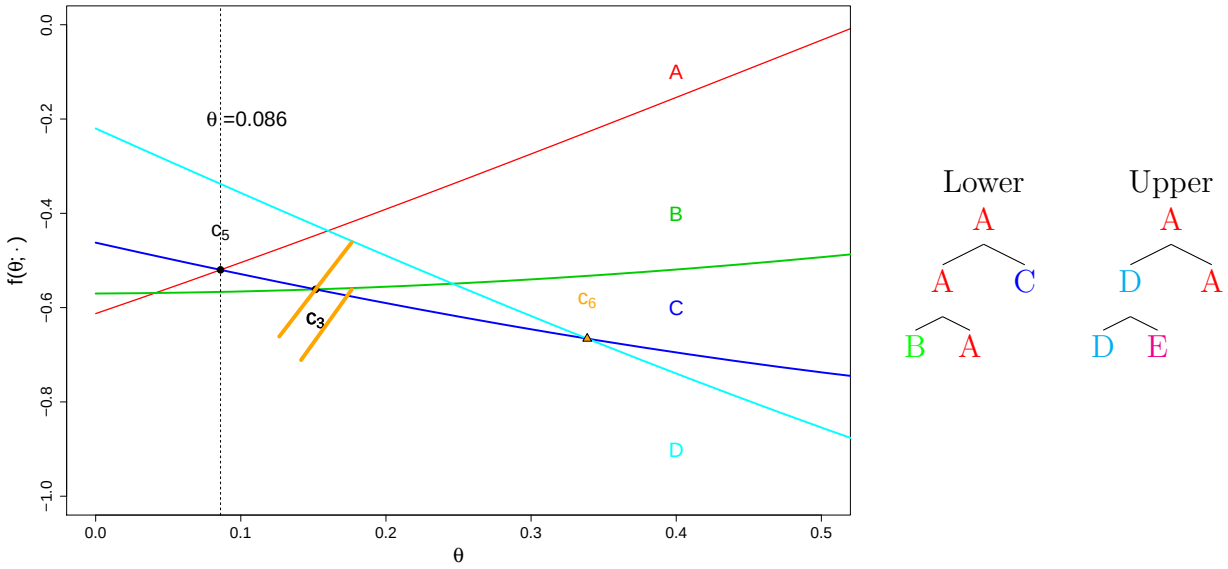



Figure 2.13: Zoomed in trajectories from Figure 2.9. Shows the certificates after second event is processed. $A$ and $C$ swap and $A$ becomes the median curve. $A$ replaces $C$ as the root of the Upper heap and replaces $C$ everywhere in the Upper heap. A certificate, $c_6$, is generated for $A$ and $D$. $c_3$ is removed since $D$ and $C$ are no longer siblings.

is a summary of the algorithm, which can be described as a 'sweep-line' algorithm. The line

actually stops at discrete points (events).

**Algorithm 1** (Projection Median in $\mathbb{R}^2$)**.**

1. Represent the points as trajectories.
2. Begin with the line at $\theta = 0$ and initialize the kd-heaps.
3. As trajectories change relative positions, maintain the kd-heaps.
4. Stop sweeping the line at $\theta = \pi$. We now have the median-level.
5. Calculate the length of the interval for each segment in the median-level. For each $i$, in the odd (even) case sum the lengths for which $x_i = x_j^*$ ($x_i$ is one of $x_{j1}^*, x_{j2}^*$) and divide by $\pi$ ($2\pi$), this is $w_i$ in (2.1).
6. Use (2.1) to calculate $M(\mathbf{X}_n)$.

## 2.3 Exact Algorithm in $\mathbb{R}^3$

We will now examine the problem of computing $M(\mathbf{X}_n)$ when $d = 3$. The following underlying technique of transforming the point set $\mathbf{X}_n$ into a dual arrangement of planes, whose median-level (Definition 13) is computed, is due to Basu et al. (2011). If we start by looking at (2.1) the computational difficulty is evident; it seems to involve computing an infinite amount of univariate medians. We have seen in Section 2.2 that this is not actually the case in $\mathbb{R}^2$, where it suffices to examine only discrete points at which the median-level changes. The same is true in $\mathbb{R}^3$. The permutation of projected points along $u$ only changes when $u$ varies past a discrete set of $\binom{n}{2}$ planes intersecting the unit sphere. It is obvious that the projections of two points, say $x_i$ and $x_j$, are the same if and only if $u \cdot x_i = u \cdot x_j$. Geometrically this is when $u$ is contained in the plane containing the origin and whose normal is parallel to the

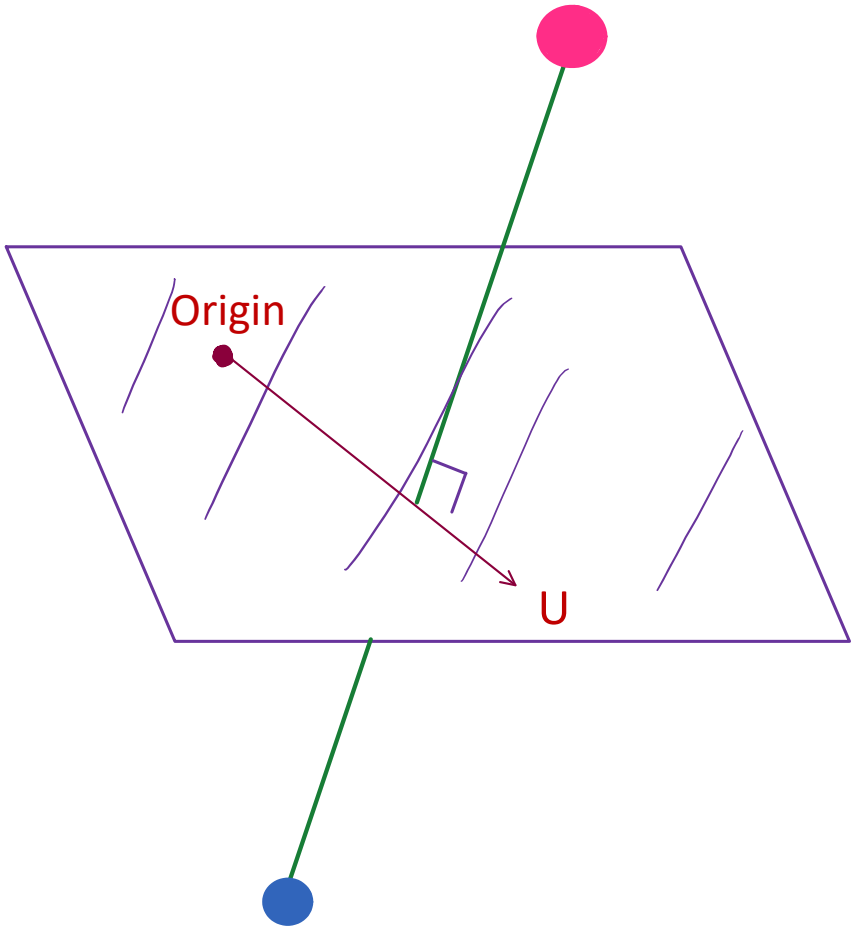



Figure 2.14: Two points have the same projection onto unit vectors within the plane whose normal is parallel to the line running through the two points.

line running through the two points, see Figure 2.14. For each pair of points $x_i, x_j \in \mathbf{X}_n$ there is one plane, so there are $\binom{n}{2}$ planes, each passing through the origin, with a normal determined by the vector $x_i - x_j$.

This creates a partition of the sphere into regions, $\mathbb{G} = \{Q_1, .., Q_{k_1}\}$, such that permutations of the projected points only differ when the directions, $u_1$ and $u_2$, belong to different regions $Q_i$ and $Q_j$. In fact, we can consider a coarser partition than $\mathbb{G}$, say $\mathbb{G}^* = \{Q_1^*, .., Q_{k_2}^*\}$, $k_2 < k_1$, such that each member of $\mathbb{G}^*$ constitutes a set of directions where the median point of the permutation remains the same, but the ordering of other points may change. Figure

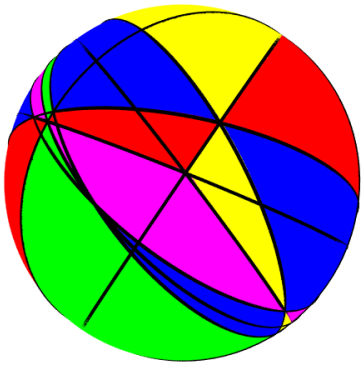


Figure 2.15: Partition $\mathbb{G}$ vs. $\mathbb{G}^*$. Each region is is a member of $\mathbb{G}$, whereas each union of same coloured, neighbouring regions constitutes a member of $\mathbb{G}^*$. For example the sum of the surface areas of all of the green regions divided by the total surface area of the sphere will be $w_{green}$, the weight for the green point.

2.15 gives a visual interpretation of the regions while also demonstrating the computational redundancy of computing $\mathbb{G}$ versus $\mathbb{G}^*$. Although it is less straightforward, it is much more efficient to only compute $\mathbb{G}^*$; the upper bounds on the number of vertices in each partition are $O(n^{5/2})$ and $O(n^4)$ respectively, see Basu et al. (2011). It is left to determine an efficient algorithm to compute $\mathbb{G}^*$.

To find $\mathbb{G}^*$, the problem of partitioning the unit sphere is transformed into finding the median-level of an arrangement of planes. This is extremely similar to the median-level described in the previous section, however the method of computing is very different. First, note the following definition.

**Definition 9** (Edelsbrunner, 1997, Andrzejak and Welzl, 1997)**.**
A *halving-facet* of a set of points $\mathbf{X}_n$ in $\mathbb{R}^3$ is an oriented hyperplane that contains 3 points of $\mathbf{X}_n$ and has $\frac{n-3}{2}$ points on its positive side, i.e. the side containing $(0, 0, \infty)$.

Now, note that when $u = \nu_{ijk}$, where $\nu_{ijk}$ is normal to the plane containing 3 points of $\mathbf{X}_n$; $x_i, x_j, x_k$, the projections onto $u$ of these 3 points are equal. Further if $x^*_{\nu_{ijk}} = x_i$ then this plane is a halving facet of $\mathbf{X}_n$. Such vectors that are normal to halving facets are called

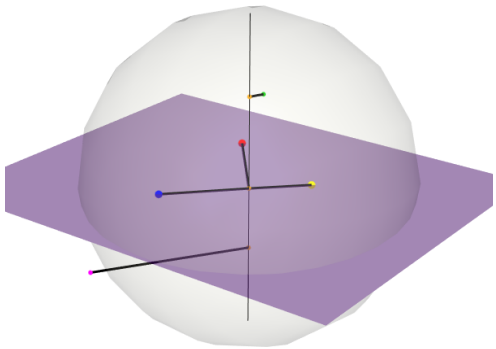


Figure 2.16: A halving facet in $\mathbb{R}^3$. The projections of the 3 points in the plane onto the normal are indeed equal and all 3 of these projections are the univariate median. There is an equal number of points on either side of the plane, confirming it is a halving facet. The normal to this plane is thus a critical vector and is in $\mathbf{C}$.

*critical vectors*, see Figure 2.16 for an example.

Denote the set of critical vectors by $\mathbf{C}$. Knowing $\mathbf{C}$ and the adjacency relations of the vectors in $\mathbf{C}$ we can trivially construct $\mathbb{G}^*$. These vectors represent the vertices of $\mathbb{G}^*$.

To find such $\mathbf{C}$ we can apply a dual, $\mathbb{D}$ to $\mathbf{X}_n$ to transform the set of points into an arrangement of hyperplanes, denoted $\mathbb{D}(\mathbf{X}_n)$. A *projective geometric dual*, denoted $\mathbb{D}(X)$ where $X$ is a set of objects, is an isomorphic mapping from one set of objects to another such that certain properties are preserved. Here we are concerned with a dual between a multiset and an arrangement of planes (Edelsbrunner, 1997; Andrzejak and Welzl, 1997). For a formal definitions of an arrangement see Definition 10. In plain language, we represent each point as a hyperplane and consider the cell complex created by those hyperplanes. Specifically, if we apply the paraboloidal transformation, denoted $\mathbb{D}_t$ and defined below, we can take advantage of a one-to-one correspondence between halving facets and *median-level vertices* in the arrangement $\mathbb{D}_t(\mathbf{X})$.

**Definition 10** (Edelsbrunner, 1997, Andrzejak and Welzl, 1997, Agarwal et al., 1998)**.** An *arrangement*, $A(\mathbf{H})$, of a set of hyperplanes, $\mathbf{H}$, is the cell complex determined by $\mathbf{H}$.

**Definition 11** (Andrzejak and Welzl, 1997)**.**

The *paraboloidal transformation*, denoted $\mathbb{D}_T$, is a transformation such that each point $x_i = (x_{i1}, x_{i2}, x_{i3})$ in a multiset $\mathbf{X}$ is mapped to the plane satisfying $z = 2x_{i1}x + 2x_{i2}y - x_{i3}$.

**Definition 12** (Andrzejak and Welzl, 1997)**.**

A *median-level vertex* in an arrangement $A(\mathbf{H})$ is an intersection point of 3 planes such that $\lfloor \frac{n-d}{2} \rfloor$ planes lie strictly above it and $\lfloor \frac{n-d}{2} \rfloor$ lie strictly below it.

The bijection generated from the paraboloidal transformation says that the adjacency graph of median-level vertices in the arrangement directly corresponds to the graph of $\mathbb{G}^*$. By 'graph' we refer to the graph whose nodes are labelled with three points in $\mathbf{X}_n$, where the normal to the plane containing them is in $\mathbf{C}$. Two nodes are connected if their normals are adjacent vertices with respect to the $\mathbb{G}^*$ partition of the sphere. If a group of 3 points' dual planes form a median-level vertex, the normal to the plane containing them will be in $\mathbf{C}$. Thus, the bijection implies that computing the median-level vertices and their adjacency relations is equivalent to computing the *median-level* in the arrangement. $\mathbb{D}_T$ has the properties known as *order* and *incidence preservation* from which the bijection comes, see Edelsbrunner (1997).

**Definition 13** (Edelsbrunner, 1997, Andrzejak and Welzl, 1997, Agarwal et al., 1998)**.**

The *level* of a point is the number of hyperplanes lying strictly above it. A cell's *level* is defined by the level of one of its interior points. The *median-level* in an arrangement of planes is the cell complex comprised of cells of level $\frac{n}{2}$.

Another description of the median-level could be the cell complex comprised of faces in the arrangement that contain only median-level vertices. To compute the median-level of an arrangement of hyperplanes, we apply the randomized algorithm given by Agarwal et al. (1998). This algorithm achieves an expected running time of $O(n^{5/2+\epsilon})$ (Agarwal et al., 1998). We next give an overview of the algorithm.

Consider a set of planes, $\mathbf{H}$. $\mathbf{H}$ is randomly permuted and planes are inserted one by one into a smaller arrangement $A(\mathbf{R})$. At step $r$, we have an arrangement $A(\mathbf{R})$ of the first $r$ planes $\mathbf{R}$. Throughout the computation we keep track of 3 things:

- a canonical, or bottom-vertex triangulation of a sub-complex of $A(\mathbf{R})$,
- the level of an interior point of each simplex,
- for each simplex, a list of the planes remaining to be inserted, $\mathbf{H}\backslash\mathbf{R}$, that intersect it as well as for each plane in $\mathbf{H}\backslash\mathbf{R}$ a list of each simplex that it intersects, called its *conflict list*.

Each time we insert a plane we efficiently update these structures and then use the updated levels and conflict lists to determine, for each simplex, if it is possible that it is part of the median-level. If not, we delete it from the data structure. In the end, we are left with a 3-dimensional cell complex containing the median-level and we simply trim it to the median-level. Below is a summary of the algorithm.

**Algorithm 2** (Projection Median in $\mathbb{R}^3$)**.**

1. Represent the points as planes using the paraboloidal transformation (Definition 11).
2. Find the median-level of this arrangment of planes via Agarwal et al. (1998).
3. Use the graph of the median-level and which 3 points make up each vertex to produce $\mathbb{G}^*$.
4. Calculate the surface area and $x_u^*$ (for one $u$) in each section of $\mathbb{G}^*$ .
5. Use step 4 and (2.1) to calculate $M(\mathbf{X}_n)$.

## 2.4 Approximations in $\mathbb{R}^d$

The following Monte Carlo algorithms are due to Durocher et al. (2017), both of which are in the R implementation available on Github (Ramsay, 2017). As described above, (2.1) is not an optimization problem but a weighted average. This fact provides two very straightforward Monte Carlo approximation algorithms for computing the projection median when $d > 3$. We start with the following algorithm.

**Algorithm 3** (Projection Median Approximation 1 (Durocher et al., 2017))**.**

1. Sample $m$ unit vectors uniformly distributed on $S^{d-1}$.
2. Calculate $x^*_u$ for each unit vector.
3. Calculate $\hat{M}_1(\mathbf{X}_n) = \frac{1}{m}\sum_{j=1}^{m} x^*_{u_j}$.

We can sample many, say $m$, unit vectors uniformly on the $d-1$-unit hypersphere. To sample a vector uniformly on the unit hypersphere one can sample $d$-dimensional standard multivariate normal vectors $Y_j$ and transform them to uniform on $S^{d-1}$ via $U_j = \frac{Y_j}{||Y_j||_2}$ (Muller, 1959). For each vector, it is easy to compute $x^*_{u_j}$. Averaging the $x^*$s is actually equivalent to estimation of the weights. The weights are approximated by the proportion of times $x_i$'s projection is the univariate projected median in the sample of unit vectors. We can define $\hat{w}_i = \frac{1}{m}\sum_{j=1}^{m} \mathbb{1}(x^* = u_j' x_i)$. Thus, we approximate $M(\mathbf{X}_n)$ with

$$\hat{M}_1(\mathbf{X}_n) = \sum_{i=1}^{n} \hat{w}_i x_i = \frac{1}{m}\sum_{j=1}^{m} x^*_{u_j}.$$

The second algorithm is based on the fact that the weights in (2.1) can be written as $P(U \in S^{d-1}_{x_i})$ if $u$ is replaced by $U$, which is taken to be uniformly random on the unit hypersphere. The intuition behind this estimator comes from the fact that all probabilities

can be written as expectations of Bernoulli random variables; $\hat{w}_i$ is simply a sample mean. We now summarize the second algorithm.

**Algorithm 4** (Projection Median Approximation 2 (Durocher et al., 2017))**.**

1. Sample $m$ unit vectors on $S^{d-1}$.
2. Calculate $\text{med}(\mathbf{X}_{n,u})$ for each unit vector.
3. Calculate $\hat{M}_2(\mathbf{X}) = \frac{d}{m}\sum_{j=1}^{m} \text{med}(\mathbf{X}_{n,u_j})\ u_j$.

To understand the intuition behind the second algorithm first recall the second definition of the projection median given in Section 2.1, Definition 8. Notice that we can write (2.2) as $d \cdot \mathbb{E}(\text{med}(\mathbf{X}_{n,U})U)$ where $U$ has the same meaning as above. This alternate form leads us to another sample mean based estimator,

$$\hat{M}_2(\mathbf{X}) = \frac{d}{m}\sum_{j=1}^{m} \text{med}(\mathbf{X}_{\mathbf{n},\mathbf{u_j}})\ u_j.$$

More colloquially, the projection median can be approximated by an average of all univariate projected medians, as opposed to a weighted average of sample points. Both of these algorithms take $O(mnd)$ time, and approximation factors, confidence ellipses and confidence intervals are given by Durocher et al. (2017). These algorithms converge to the exact projection median.

**Theorem 3** (Durocher et al., 2017)**.**

These algorithms converge to the exact projection median;

$$\hat{M}_1(\mathbf{X}_n) \overset{\text{a.s.}}{\to} M(\mathbf{X}_n)$$

and

$$\hat{M}_2(\mathbf{X}_n) \overset{\text{a.s.}}{\to} M(\mathbf{X}_n),$$

as $m \to \infty$.

It is natural to compare these algorithms in terms of their approximation factor. It has been shown by Durocher et al. (2017) that Algorithm 3 performs better when $d$ is very large, but the opposite is true when $n$ is large and $d$ is relatively small.

# Chapter 3

# Integrated Rank-Weighted Depth

## 3.1 Introduction

The study of the projection median has motivated the study of a new data depth measure, based on centre outward ranks. Among many other applications, data depth extends the concept of rank to the multivariate setting. As discussed in Chapter 1, measures of data depth provide a center outward ordering of points in any dimension. In this chapter we define and study *integrated rank-weighted depth*, or IRW depth for short, an intuitively defined depth measure that is easily approximated in high dimensions.

## 3.2 Integrated Rank-Weighted Depth in $\mathbb{R}^2$

Consider a set of univariate points. Informally, we say a point is 'deep' relative to a point set if it is surrounded by many other points; see Figure 3.1. A point's univariate depth is defined below.

**Definition 14.**

Let $F_n(y) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}(X_i \leq y)$ and $r = nF_n(y) + 1$ is the rank of $y$ as if it were in the sample. The univariate depth of a point $y \in \mathbb{R}$ with respect to a (univariate) sample $X_n$ of


Figure 3.1: Depth rankings for a set of univariate points

size $n$ whose empirical distribution is $F_n$ is

$$d(y; F_n) = \min(r - 1, n - r) = \frac{1}{2}(n - 1 - |n - 2r + 1|).$$

Integrated rank-weighted depth comes from asking the question: on average, how 'deep', in the univariate sense, is a point relative to the other points, when considering all possible univariate projections of the data set? Integrated rank-weighted depth comes from asking the question: on average, how 'deep', in the univariate sense, is a point relative to the other points, when considering all possible univariate projections of the data set?

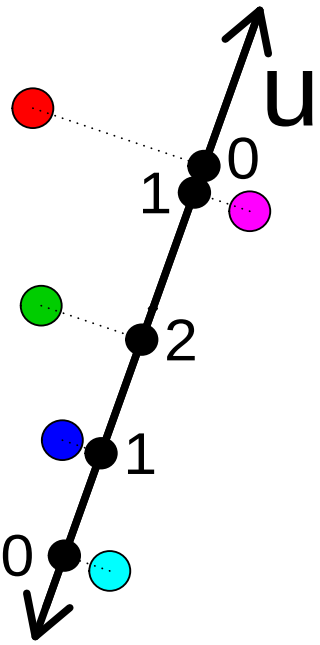
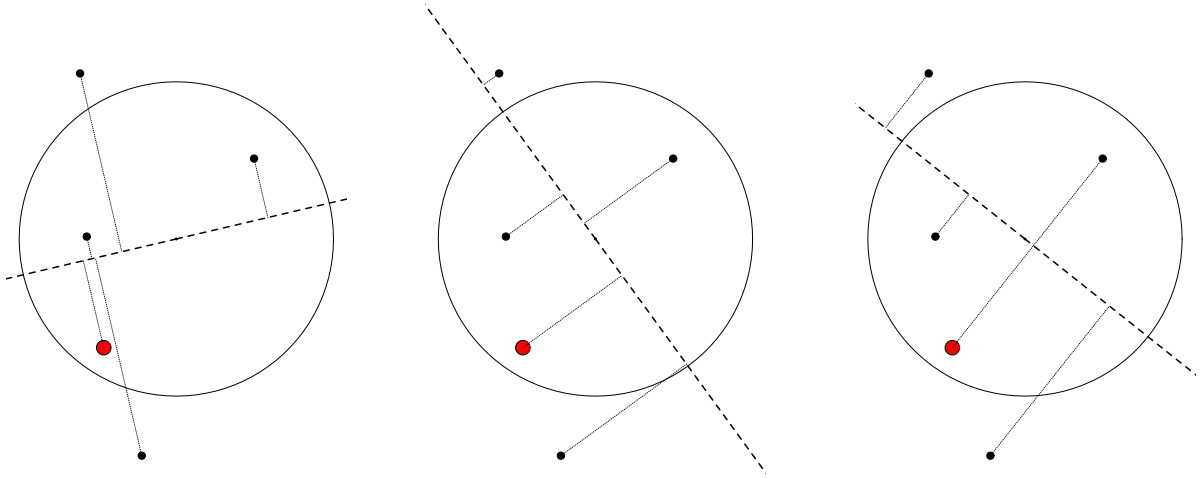



Figure 3.2: The points projected onto the line determined by the vector $u$ and their center outward rank.

Now consider a multivariate point set, $\mathbf{X_n}$. Consider a unit vector, $u$, and project the points in $\mathbf{X_n}$ onto the line parallel to that unit vector (and through the origin) as in Chapter 2; see Figure 3.2.

We can determine how deep a point is in a given direction by calculating the univariate depth of that point's projection with respect to the other projections of points in $\mathbf{X_n}$. We will refer to the depth of a point in a direction $u$ as $d_u(y; F_n) = d(y \cdot u; F_{n,u})$, where $F_n$ is the empirical distribution of $\mathbf{X_n}$, $F_{n,u}$ is the empirical distribution of $\{X_1 \cdot u, \ldots, X_n \cdot u\}$, the projected sample. A point's univariate depth can vary significantly in different directions; see Figure 3.3 for an example.



(a) (b) (c)

Figure 3.3: Depth of red point in different directions. (a) Depth of the red point is 0. (b) Depth of the red point is 1. (c) Depth of red point is the same as another point.

Notice that in Figure 3.3c; the red point has the same projection as another point. Recall from Chapter 2 that this happens because the unit vector is perpendicular to the line passing through the two points; the unit vector is along the bisector. We shall refer to these unit vectors by their polar angle, $\theta_i$. If there are $n^* \leq n$ unique points in $\mathbf{X_n}$, then an arbitrary point, $y \in \mathbb{R}^2$ (the query point), has $n^*$ different bisectors; one for each point in the set. For each bisector, there are two unit vectors that are parallel to it, and so the plane is thus divided into $2n^*$ sections where a point's univariate depth remains the same. However, by symmetry of projections onto $u$ and $-u$, we only need to consider $n^*$ sections when determining the depth of $y$. This implies that $y$'s univariate depth is constant for unit vectors in each of these sections. Figure 3.4 shows a partition of the plane determined by the red point's projections relative to the point set. From here, we can define integrated rank-weighted depth.

**Definition 15.**

Let $\mathbf{X_n}$ be a sample (with possible duplicates) from a distribution on $\mathbb{R}^2$. Let $F_n$ represent the empirical cumulative distribution function determined by $\mathbf{X_n}$. The *integrated rank-weighted*

*depth*, $D(y; F_n)$, of point $y \in \mathbb{R}^2$ with respect to $F_n$, is

$$D(y; F_n) = \frac{2}{n} \sum_{i=1}^{n^*} w_i d_{u_i}(y; F_n), \qquad \text{with} \qquad w_i = \frac{(\theta_{(i)} - \theta_{(i-1)})}{\pi}, \tag{3.1}$$

and where $u_i \in (\theta_{(i)}, \theta_{(i-1)})$ and $\theta_{(i)}$ is the $i^{th}$ ordered angle with $\theta_{(0)} = \theta_{(n^*)} - \pi$.

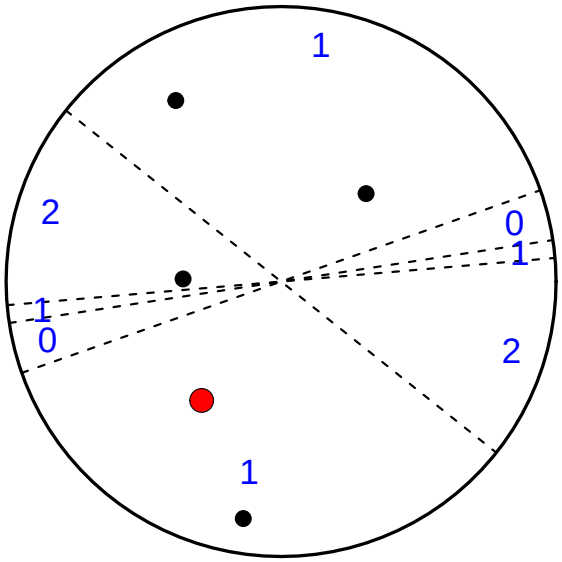
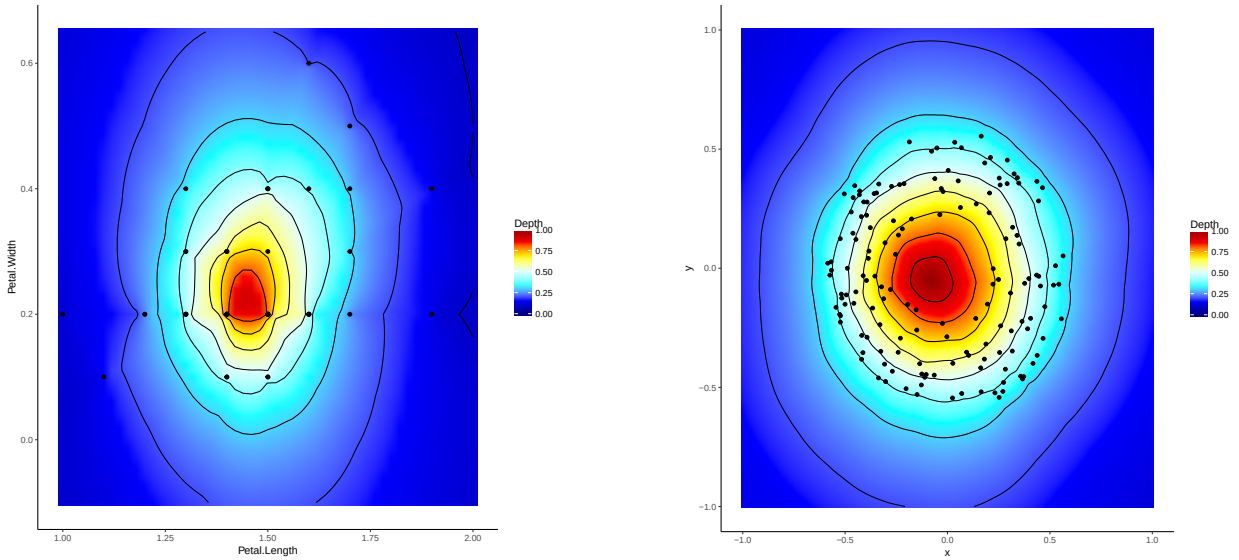



Figure 3.4: Division of plane into sections where the red point's univariate depth remains the same.

In other words, *integrated rank-weighted depth* is an average of univariate center outward ranks (a univariate depth), where the average is taken over all unit vectors or directions. The weights $w_i$ are proportional to the size of section $i$ when considering the sections into which $y$'s bisectors divide the plane. The $\theta_{(i)}$'s represent the ordered section boundaries such that the univariate depth of $y$ changes when we sweep or rotate the unit vector past that angle. The univariate depth of $y$ in section $i$ is determined by projecting $\mathbf{X_n} \cup \{y\}$ onto a line contained in section $i$ and viewing it as a univariate set of points. Note that we multiply by 2 and divide by $n$ to normalize the depth of a point to the interval $(0, 1]$. This is done so that depths are comparable between different sample sizes. The depth of a point can be interpreted as its average centrality relative to the other points, where 1 is maximally central and 0 is maximally outlying. One could imagine adding $y$ to the point set, and then calculating its univariate ranks as if it was in the point set. Notice here that the weights also account for the multiplicity of $y$ in the rank function, thus, ties are handled appropriately. Computing the depth of a point $y$, with respect to some point set $\mathbf{X_n}$, is done by calculating the $n$ bisectors between $y$ and each point of $\mathbf{X_n}$. We can then compute the $n$ unit vectors through each of




(a) (b)

Figure 3.5: Some IRW depth contours. (a) Example 1, setosa scatter and contour plot. (b) 'Donut' point set and contour plot.

these bisectors. The vectors are then converted to spherical coordinates and we calculate and sort the $n$ angles $\theta_i$ that make up the subdivision of the plane. This is analogous to the critical vectors from Chapter 2. We then simply calculate the univariate depths in each section, along with normalizing the differences in sorted angles to use as weights. The time is dominated by the sorting process, thus taking $O(n \log n)$ time. We can illustrate this depth measure with a quick example.

**Example 3** (Iris Data)**.**

We use the famous Iris data (Anderson, 1936; Fisher, 1936). This data set contains 150 observations, 50 from each of three species of flowers, with four different flower features. We will use the Petal Length and Petal Width measurements from the different Iris species to demonstrate the use of this depth measure throughout this paper. Figure 3.5a shows a scatter plot of the setosa species measurements along with a heat map and contours of the depth measure. Observe that areas that are surrounded by many points do indeed have high depth values. The contours are also nested and somewhat elliptical, with the middle-valued contours' shapes 'bent' toward the right to capture the right tilt. The outer contours' shape

balloons outward in two corners to reflect the empty areas to the top left and bottom right of the scatter plot; the measure captures the geometry of the data.

It is important to keep in mind that most depth measures, including IRW depth, are defined in such a way that a point only needs to be somewhat 'surrounded' by other points to be deep; it does not necessarily have to be a point of high density. Consider the set of points and its depth heat map in Figure 3.5b. Note that the deepest region is associated with an area of low density. If it is desired that deepness be related to high density, one must restrict their attention to unimodal densities. Traditionally, in the context of which depth based analysis is used, unimodality is assumed (Zuo and Serfling, 2000), but this assumption is application dependent. Let $F(x-) = P(X < x)$. Note that we can rewrite (3.1) as

$$\begin{aligned} D(y; F_n) &= \frac{2}{n} \sum_{i=1}^{n^*} w_i d_{u_i}(y; F_n) \\ &= \frac{2}{\pi} \sum_{i=1}^{n^*} (\theta_{(i)} - \theta_{(i-1)}) \frac{d_{u_i}(y; F_n)}{n} \\ &= \frac{2}{2\pi} \int_S \frac{d_{u_i}(y; F_n)}{n} du \\ &= \frac{1}{\pi} \int_S \min(F_{n,u}(y \cdot u), 1 - F_{n,u}(y \cdot u-)) du. \end{aligned} \tag{3.2}$$

This leads to a natural definition of IRW depth for any distribution in any dimension $d$. This equivalence is also very nice in that it allows us to leverage properties of the empirical cumulative distribution function when studying the theoretical properties of this depth measure. It also lends itself nicely to a population definition which simply replaces $F_{n,u}(y \cdot u)$ with $F_u(y \cdot u)$, where $F_u$ is the distribution of $X \cdot u$, the linear projection of $X$ onto $u$, if $X$ comes from the multivariate parent distribution.

## 3.3 Integrated Rank-Weighted Depth in $\mathbb{R}^d$

We now generalise the definition of IRW depth to any distribution over $\mathbb{R}^d$ using (3.2).

**Definition 16.**

The *integrated rank-weighted depth*, $D(y; F)$, of point $y \in \mathbb{R}^d$ with respect to a distribution $F$, is

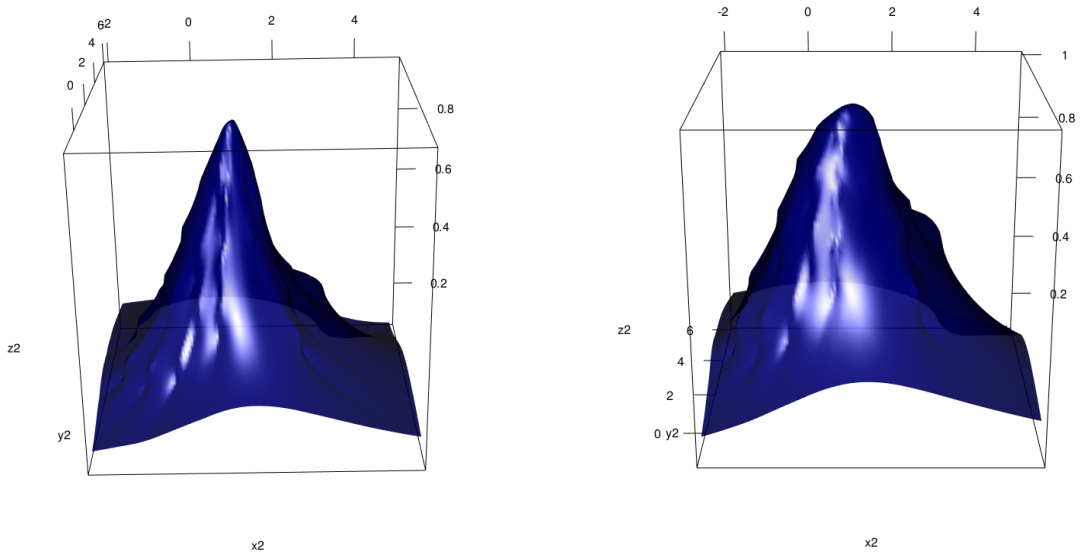
$$\begin{aligned} D(y;F) &= \frac{2}{V_d}\int_{S^{d-1}} \min(F_u(y\cdot u), 1 - F_u(y\cdot u-))du \\ &= \frac{1}{V_d}\int_{S^{d-1}} F_u(y\cdot u) + 1 - F_u(y\cdot u-) - |1 - F_u(y\cdot u-) - F_u(y\cdot u)|du, \end{aligned} \tag{3.3}$$

where $V_d = \int_{S^{d-1}} 1 \; du$ and $F_u$ is the distribution of $X \cdot u$ if $X \sim F$.

The interpretation remains the same: $D$ again measures the average centrality of a point with respect to a distribution. Sample depth simply replaces $F_u(y \cdot u)$ with $F_{n,u}(y \cdot u)$ as above. Though we present only this definition, it should be noted that Definition 15 extends to $d$ dimensions as well. The expression for $w_i$ is simply modified to represent a region on the unit hypersphere, rather than an interval of angles.

Cuevas and Fraiman (2009) studied another integrated depth measure, based on the average of a different univariate measure of depth. We will refer to this depth as 'Cuevas depth'. They replace the univariate Tukey depth: $\min(F_u(y \cdot u), 1 - F_u(y \cdot u-))$ (this is also the univariate convex hull peeling depth) in our definition, with the univariate simplicial depth: $F_u(y \cdot u)(1 - F_u(y \cdot u))$. Further, they study the more general case of integrating over any measure, not just the Haar measure (uniform on the unit hypersphere) as is done here. They mention that one could use integrated univariate Tukey depth as a separate depth measure but do not study its properties, as is done in this chapter.

It is important to discuss the similarities and differences between the two depths, especially which properties of the Cuevas depth extend to IRW depth and vice versa. The use of




(a) (b)

Figure 3.6: (a) 3D plot of the IRW depth measure with respect to a 2-dimensional point set. (b) Normalized (to (0,1]) Cuevas depth measure with respect to the same 2-dimensional point set. (The same point is also in Figure 3.10.)

univariate simplicial depth causes the depth measure to weight very central points higher, relative to the deepest point, than would the use of univariate Tukey depth. This is due to the steeper slope associated with the Tukey measure. To illustrate this fact, Figure 3.6 shows a side-by-side comparison of the two measures, and the steeper interior of the IRW depth is quite clear. Integrating the Tukey depth also makes IRW depth an intuitive generalization of integer univariate ranks, which using simplicial depth does not.

Further, multivariate Tukey depth is decreasing along rays, which is an important property for a depth measure to have, however, multivariate simplicial depth does not have this property (Zuo and Serfling, 2000). The motivation is that, by using univariate Tukey depth, IRW depth may also have this property.

Some of the properties of the Cuevas depth measure are shared with IRW depth including invariance under similarity transformations but not all affine transformations, maximality at centre, vanishing at infinity, continuity under continuous distributions and consistency.

We provide our own versions of these proofs, however, we acknowledge that these properties can be proved using the methods of Cuevas and Fraiman (2009). Their Monte Carlo algorithm for approximate depth also applies. Their proof of asymptotic normality does not apply to IRW depth. Some of the properties investigated in this paper would extend to their depth measure, under the Haar measure, including decreasing along rays (Section 3.3.1), continuity under discrete distributions (Section 3.3.1), the weighted average representation of the depth measure (Section 3.2) and the algorithm for exact computation (Section 3.2 and 3.3). Another important fact shown by Cuevas and Fraiman (2009) is that when the parent distribution is absolutely continuous, the contours of the two depths admit the same level sets, including the deepest point or region (Cuevas and Fraiman, 2009). This implies our result on the breakdown point of the deepest point as a location estimate (Section 3.4.1) extends to their depth as well, under continuous distributions.

We can compute $d$-dimensional IRW depth using a generalized version of the $\mathbb{R}^2$ algorithm. We know two points have the same projection when the unit vector is contained in the hyperplane (containing the origin) to which the line through those 2 points is normal. Thus, for a fixed point, we have $n$ hyperplanes that divide the hypersphere into sections; see Figure 3.7. We can calculate this division of the unit hypersphere in $O(n^{d-1})$ time. It is useful to note that the Cuevas depth can be computed using the same partition of the hypersphere; the same algorithm can be applied to their depth function as well. In fact any function of the projected univariate CDFs can be computed in this manner.

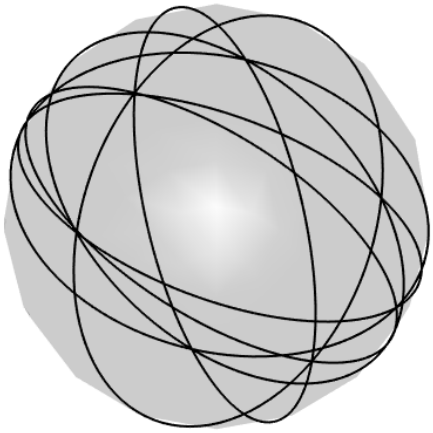


Figure 3.7: Division of the unit sphere into sections where the permutations of points remain the same.

A compelling feature of IRW depth is that computationally, in contrast to many other depth measures, it does not require solving an optimization problem. Computing the

projection median requires computing an average. This provides a very straightforward Monte Carlo approximation algorithm for computing the depth of a point when $d$ is very large. We can sample many, say $m$, unit vectors uniformly over $S^{d-1}$, calculate a point's univariate sample depth for each vector and take an average of the univariate depths. This estimate, denoted $D_m(y; F)$, can be computed in $O(mnd)$ time. The law of large numbers guarantees this algorithm does indeed converge to the exact value of the depth being estimated as $m$ tends to infinity. This is the same idea as Algorithms 3 and 4 described in Chapter 2 as well as the one presented by Cuevas and Fraiman (2009).

**Proposition 1.**

The Monte Carlo estimate of the depth of a point, $y$, with respect to a distribution $F$, converges almost surely to the depth of $y$ as the number of sampled unit vectors approaches infinity, $D_m(y; F) \xrightarrow{\text{a.s.}} D(y; F)$ as $m \to \infty$.

*Proof.* Proposition 1

Let $U \sim G$ where $G$ is the uniform distribution over all unit vectors on the $(d-1)$-dimensional hypersphere. Then, as $m \to \infty$,

$$D_m(y; F) = \frac{2}{m} \sum_{j=1}^{m} \min(F_{u_j}(y \cdot u), 1 - F_{u_j}(y \cdot u-)) \xrightarrow{\text{a.s.}} 2\mathbb{E}_G(min(F_U(y \cdot U), 1 - F_U(y \cdot U-))),$$

by the strong law of large numbers. Now, it suffices to note

$$2\mathbb{E}_G(\min(F_U(y \cdot U), 1 - F_{n,U}(y \cdot U-))) =$$

$$\frac{2}{V_d} \int_{S^{d-1}} \min(F_u(y \cdot u), 1 - F_u(y \cdot u-))udu = D(y; F),$$

which completes the proof. □

### 3.3.1 Properties

We are now ready to discuss the properties of integrated rank-weighted depth. Integrated rank-weighted depth satisfies three of the four properties introduced by Liu (1990) and later

studied by Zuo and Serfling (2000): they are monotonicity with respect to center, maximality at centre and vanishing at infinity. However, integrated rank-weighted depth is not affine invariant, it is only invariant with respect to similarity transformations, which include any combination of rotation, uniform scaling, translation and reflection. We investigate continuity and then establish the asymptotics of the depth measure. First recall this definition of symmetry. A distribution that is *H-symmetric* about a point $\gamma$ if the following property holds. If $H$ is a closed half-space containing $\gamma$, then $P(X \in H) \geq \frac{1}{2}$.

**Theorem 4.**

Integrated rank-weighted depth has the following properties:

1. (Similarity Invariance) $D(y; F)$ is invariant under similarity transformations.
2. (Maximality at Centre) If $F$ is H-symmetric about $\gamma$, then $D(y; F)$ is maximal at $\gamma$.
3. (Decreasing Along Rays) For any H-Symmetric $F$ having deepest point $\gamma$, then $D(y; F) \leq D(\alpha y + (1-\alpha)\gamma; F) \leq D(\gamma; F)$ for $\alpha \in [0, 1]$.
4. (Vanishing at Infinity) Let $c > 0$, $\lim_{c\to\infty} D(cy; F) = 0$.

*Proof.* Theorem 4

*Property* 1*:*

Let $A$ be an orthogonal matrix[1], $r \in \mathbb{R}$, $t \in \mathbb{R}^d$ and $rAX + t \sim G$. From the definition

$$D(rAy + t; G) = \frac{2}{V_d} \int_{S^{d-1}} \min(G_u(rAy + t \cdot u), 1 - G_u(rAy + t \cdot u-))\, du.$$

Since we are integrating over the unit sphere,

$$D(rAy + t; G) = \frac{2}{V_d} \int_{S^{d-1}} \min(F_{(rAu+t)/\|rAu+t\|}((rAy + t) \cdot (rAu + t)/\, \|rAu + t\|),$$

[1]The above doesn't hold for all affine transformations because non-orthogonal matrices do not preserve the dot product; it is not always true that $Ay \cdot Au = y \cdot u$.

$$1 - F_{(rAu+t)/\|rAu+t\|}((rAy+t)\cdot(rAu+t)/\|rAu+t\| -))\ du.$$

Note that because $A$ is orthogonal, $\|Au\| = 1$ and $Ay \cdot Au = y \cdot u$. Also note that $P(aX+b \leq ax+b) = P(X \leq x)$ if $a > 0$ and $1 - P(X < x)$ if $a < 0$. These two facts give

$$F_{(rAu+t)/\|rAu+t\|}((rAy+t)\cdot(rAu+t)/\|rAu+t\|) = F_u(y \cdot u) \text{ for } r > 0,$$

$$F_{(rAu+t)/\|rAu+t\|}((rAy+t)\cdot(rAu+t)/\|rAu+t\|) = 1 - F_u(y \cdot u-) \text{ for } r < 0.$$

Similarly, we have

$$F_{(rAu+t)/\|rAu+t\|}((rAy+t)\cdot(rAu+t)/\|rAu+t\| -) = F_u(y \cdot u-) \text{ for } r > 0,$$

$$F_{(rAu+t)/\|rAu+t\|}((rAy+t)\cdot(rAu+t)/\|rAu+t\| -) = 1 - F_u(y \cdot u) \text{ for } r < 0.$$

As a result, we can write

$$D(rAy+t;G) = \frac{2}{V_d}\int_{S^{d-1}} \min(F_u(y\cdot u), 1 - F_u(y\cdot u-))\ du = D(y;F).$$

*Property 2:*

Without loss of generality assume that $\gamma = 0$. Let $H$ be a closed half-space and let $h$ denote its boundary. Assume that $h$ contains 0. Now, by assumption $H^c \cup h$ is also a closed half-space containing 0. By definition of half-space symmetry, we have both

$$\frac{1}{2} \leq P(X \in H) \qquad \text{and} \qquad \frac{1}{2} \leq P(X \in H^c \cup h).$$

Now, define $u$ as a unit vector normal to $h$. Then, we have that,

$$F_u(0) = P(X \cdot u \leq 0) = \min(P(X \in H), P(X \in H^c \cup h)) = \frac{1}{2}.$$

Since there is a bijection between $S^{d-1}$ and half-spaces that contain 0, $F_u(0) = \frac{1}{2}$ for all $u$. Therefore $D(0;F) = 1$ which is the maximum depth that can be achieved.

*Property* 3*:*

Without loss of generality again assume that $\gamma = 0.$, and $y$ is on the positive x-axis; $y = (y_1, 0, \ldots, 0)$, $y_1 > 0$. Recall

$$u = (\cos\phi_1, \sin\phi_1\cos\phi_2, \ldots, \sin\phi_1\sin\phi_2, \ldots, \sin\phi_{d-1}),$$

where $\phi_1, \ldots, \phi_{d-1}$ are the usual spherical coordinates. Note that $u \cdot y = y_1\cos\phi_1$, $\phi_1 \in [0, \pi]$. Note also that $u \cdot y \geq 0$ on $\phi_1 \in [0, \frac{\pi}{2}]$ and $u \cdot y \leq 0$ on $\phi_1 \in [\frac{\pi}{2}, \pi]$. Since $F_u(0) = \frac{1}{2}$, we know that $\min(F_u(y \cdot u), 1 - F_u(y \cdot u-)) = 1 - F_u(y \cdot u-)$ for $\phi_1 \in [0, \frac{\pi}{2}]$ and $\min(F_u(y \cdot u), 1 - F_u(y \cdot u-)) = F_u(y \cdot u)$ otherwise. Note that the same applies to $F_u(\alpha y \cdot u)$, $(\alpha \in [0, 1])$. Now for $\phi_1 \in [0, \frac{\pi}{2}]$, $1 - F_u(y \cdot u-) \leq 1 - F_u(\alpha y \cdot u-)$ since $y \cdot u \geq \alpha y \cdot u$. Similarly on $\phi_1 \in [\frac{\pi}{2}, \pi]$ $F_u(y \cdot u) \leq F_u(\alpha y \cdot u)$ since $y \cdot u \leq \alpha y \cdot u$ (recall that $y \cdot u < 0$). After projecting onto any unit vector, $\alpha y$ has univariate depth greater than or equal to that of $y$. Hence, it must have greater average depth. Therefore $D(y; F) < D(\alpha y; F)$ and the proof is complete.

*Property* 4*:*

Let $y$ be non-zero, also note $u$ is always non-zero. By the following properties of cumulative distribution functions

$$\lim_{x\to\infty} F(x) = 1 \qquad \text{and} \qquad \lim_{x\to-\infty} F(x) = 0,$$

It is easy to see that

$$\lim_{c\to\infty} \min(F_u(c(y \cdot u)), 1 - F_u(c(y \cdot u)-)) = \min(1, 0) = 0.$$

There may be one or more $u$'s such that $cy \cdot u = 0$, but the integrand is bounded by 1 and converges to 0 almost surely. Further, $S^{d-1}$ is independent of $c$. Consequently, by the bounded convergence theorem,

$$\lim_{c\to\infty} \int_{S^{d-1}} \min(F_u(cy \cdot u), 1 - F_u(cy \cdot u-)\, du = \int_{S^{d-1}} \lim_{c\to\infty} \min(F_u(cy \cdot u), 1 - F_u(cy \cdot u-)\, du = 0,$$

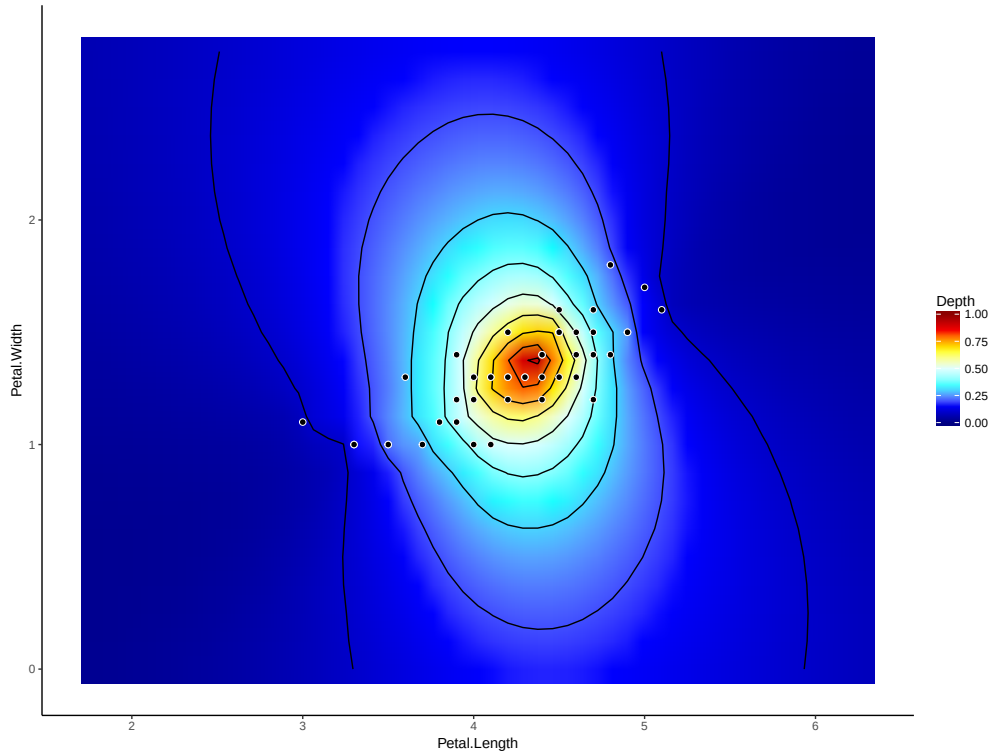
this completes the proof. □

Integrated rank-weighted depth is not invariant under all affine transformations, but it is invariant under similarity transformations. The fact that it is affected by non-uniform scaling causes axes with larger scales to have greater influence on the depth of a point than ones with smaller scales. To remove this feature one could scale the data in a robust way, such as scaling each coordinate by its median absolute deviation, or its sample variance using the deepest 50% of the data. This can be seen below as well as in Chapter 4. Alternatively, if there was a sufficient reason, one could use different scales to adjust the influence of certain variables. For an in-depth study on invariance, equivariance and how scaling affects estimators see Serfling (2010).

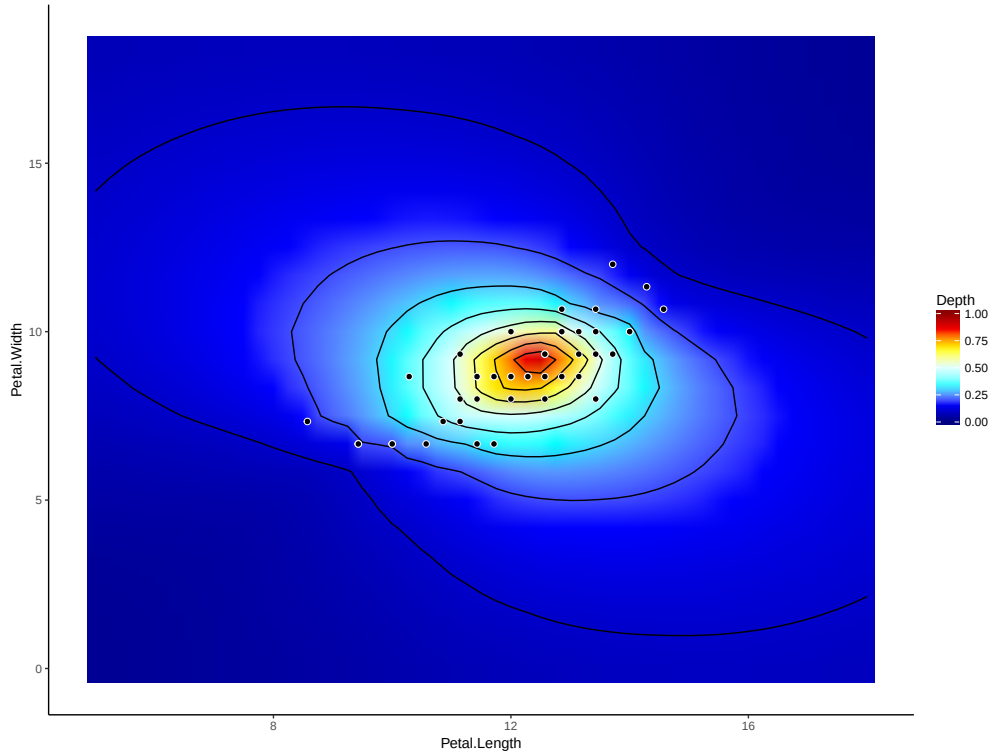
**Example 3** (Iris Data cont.)**.**

Figure 3.8a shows the depth contours for the Virginica species raw data. Figure 3.8b shows the depth contours for the same data, but with each coordinate scaled by its median average deviation. You can see that, in the unscaled data, a point's depth in the X-direction plays a larger role in its overall depth when compared with the scaled data. Points originally not deep in the X-direction but deep in the Y-direction become deeper after scaling and points originally not deep in the Y-direction but deep in the X-direction become shallower after scaling. Notice also that the deepest regions are less affected by the scaling. This is due to the fact that very deep points are deep in all directions. Placing an emphasis on certain directions is then not expected to affect their depth very much.

Integrated rank-weighted depth has the most general manifestation of Property 2; H-symmetry is the most general description of symmetry considered when analysing depth measures (Zuo and Serfling, 2000). On the topic of symmetry, we can also note that if the underlying population is antipodal-symmetric about some $\gamma$ ($X - \gamma \overset{\mathrm{d}}{=} X + \gamma$), so is $D$ (Liu, 1990). Property 3 ensures that $D$ measures depth or centrality. As we move away from the deepest point, the depth decreases.




(a)




(b)

Figure 3.8: Example 3, (a) Virginica unscaled depth contours and (b) Virginica scaled depth contours.

Property 4 shows that, as a point is pulled to infinity along a ray, its depth approaches 0. It is interesting to note that a point's univariate depth is based on rank; it does not provide any information about the spread of the points or how far the lowest depth is from the remainder of the sample. However the average rank over all unit vectors does contain this information, as demonstrated by Properties 2 and 3 in Theorem 4. We see next that integrated rank-weighted depth has good continuity properties with respect to the query point.

**Proposition 2.**

$D(y; F)$ is continuous in $y$ for discrete and absolutely continuous distributions.

*Proof.* Proposition 2

*Case* 1*:*

$F$ is discrete. First, assume that the support of $F$ is finite. Let $z \in \mathbb{R}^d, \delta_1 \in \mathbb{R}$. Choose $\delta_1$ small enough such that the graph of vertices on the unit hypersphere where the univariate depth of $y$ changes is the same for $z$. Further, assume that $0 < \|y - z\| < \delta_1$ and let $N$ denote the number of sections into which the graph divides the unit hypersphere. We have

$$|D(y; F) - D(z; F)| \leq \frac{2}{n} \sum_{i=1}^{N} |(w_i(y) - w_i(z))| d_{u_i}(y; F) \leq 2 \sum_{i=1}^{N} |(w_i(y) - w_i(z))|.$$

All that we need to show is that $\sum_{i=1}^{N} |w_i(y) - w_i(z)| < \frac{\epsilon}{2N}$.

Assume $x$ is some vector in the support of $F$. We are now interested in the vertices of these sections on the unit sphere. The angles in spherical coordinates, denoted by $\phi_i, ..., \phi_{d-1}$, such that $x$ and $y$ have that same projection is given by $\phi_i = \prod_{j=d-1-i}^{d-1} \tan(\phi_j) \text{atan}(f_1(x, y))$ for $i > 2$ and $\phi_1 = \text{atan}(f_1(x, y))$, where $f_1$ is a continuous function in $y$. (We know this because $f_1$ is a product and composition of continuous trigonometric functions.) The functions atan() and tan() are also continuous, thus, for each $i$, there is a $\delta_{2i}$ such that $\|y - z\| < \delta_{2i}$ implies

$|w_i(y) - w_i(z)| < \frac{\epsilon}{2N}$. Choose $\delta = \min(\delta_1, \delta_{21}, \ldots, \delta_{2N})$. Then, if $\|y - z\| < \delta$, we have that

$$|D(y; F) - D(z; F)| \leq \frac{2}{n} \sum_{i=1}^{N} |(w_i(y) - w_i(z))| d_{u_i}(y; F) < 2 \sum_{i=1}^{N} |(w_i(y) - w_i(z))| < \epsilon.$$

Consider the case where $y$ lies in a $(d-1)$-dimensional hyperplane with $d$ distinct points in the support of $F$, call this set of points $B$. In this case, if $z$ lies outside this hyperplane determined by $B$, there does not exist a $\delta_1$ small enough such that the graph of vertices on the unit hypersphere where the univariate depth of $y$ is the same as $z$. Rather, if $\delta_1$ is small enough, the univariate depth will be the same in all sections on the hypersphere, except for one. There will be one section, call it $s$, such that the univariate depth of $z$ differs from the univariate depth of $y$ by $\pm k$, $k < \frac{n}{2}$. This section is contained inside one of the sections in which the univariate depth of $y$ does not change. Let $N$ denote the number of sections into which $y$ divides the unit hypersphere, and consequently $z$ divides the hypersphere into $N + 1$ sections. Let $w_{N+1}$ be the size of the section $s$. In this case, we have

$$|D(y; F) - D(z; F)| \leq 2 \sum_{i=1}^{N} |(w_i(y) - w_i(z))| + 2k \; w_{N+1}(z).$$

Note $w_{N+1}$ is the size of the region enclosed by the $d$ hyperplanes, each of which contains $z$ and $d-1$ points in $B$. Thus, the size of this region is strictly decreasing as $z$ approaches this plane; it is decreasing in $\delta_1$. Thus, there is a $\delta_1$ such that $w_{N+1}(z) < \frac{\epsilon}{2N+2k}$ whenever $\|y - z\| < \delta_1$. As above, there are for each $i$, $\delta_{2i} > 0$ such that $|w_i(y) - w_i(z)| < \frac{\epsilon}{2N+2k}$ whenever $\|y - z\| < \delta_{2i}$. Choose $\delta = \min(\delta_1, \delta_{21}, \ldots, \delta_{2N})$. Then, if $\|y - z\| < \delta$, we have that

$$|D(y; F) - D(z; F)| < 2 \sum_{i=1}^{N} |(w_i(y) - w_i(z))| + 2k \; w_{N+1}(z) < (2N + 2k) \frac{\epsilon}{2N + 2k} = \epsilon,$$

still implying continuity. Clearly this extends to uncountable scenarios, where $\delta = \inf\{\delta_1, \delta_{21}, \ldots\}$.

*Case 2:*

$F$ is absolutely continuous. Since $F$ is continuous, $\min(F_u(y \cdot u), 1 - F_u(y \cdot u-))$ is continuous in $y$. Therefore $D(y; F)$ is continuous in $y$. □

We now establish the asymptotic properties of the sample IRW depth.

**Theorem 5.**

Integrated rank-weighted sample depth is uniformly weakly consistent and asymptotically normal.

1. (Consistency) Let $F$ be a distribution on $\mathbb{R}^d$, and $\mathbf{X_n}$ be a random sample from $F$. $D(y; F_n)$ is uniformly, weakly consistent; that is

$$\sup_{y \in \mathbb{R}^d} |D(y; F_n) - D(y; F)| \xrightarrow{\text{p}} 0 \quad \text{as} \quad n \to \infty.$$

2. (Asymptotic Normality) Let $A_1 = \{u \ : \ F_u(y \cdot u) < 1/2\}$, $A_2 = \{u \ : \ F_u(y \cdot u) > 1/2\}$ and $A_3 = \{u \ : \ F_u(y \cdot u) = 1/2\}$. Assume $F$ is continuous and is such that $A_3$ has Haar measure 0. Then, $\sqrt{n} D_n(y; \mathbf{X_n})$ is asymptotically normal; that is as $n \to \infty$:

$$\sqrt{n}(D_n(y; \mathbf{X_n}) - D(y; F)) \xrightarrow{\text{d}} N(0, \sigma^2),$$

where

$$\begin{aligned}
\sigma^2 = \left(\frac{2}{V_d}\right)^2 \Bigg[ & \int_{A_1} \int_{A_1} P(X \cdot u_1 \le y \cdot u_1, X \cdot u_2 \le y \cdot u_2) \\
& - P(X \cdot u_1 \le y \cdot u_1) P(X \cdot u_2 \le y \cdot u_2) \, du_1 du_2 \\
& + 2 \int_{A_1} \int_{A_2} P(X \cdot u_1 \le y \cdot u_1, X \cdot u_2 \ge y \cdot u_2) \\
& - P(X \cdot u_1 \le y \cdot u_1) P(X \cdot u_2 \ge y \cdot u_2) \, du_1 du_2 \\
& + \int_{A_2} \int_{A_2} P(X \cdot u_1 \ge y \cdot u_1, X \cdot u_2 \ge y \cdot u_2)
\end{aligned}$$

$$- P(X \cdot u_1 \geq y \cdot u_1)P(X \cdot u_2 \geq y \cdot u_2)\ du_1 du_2\Big].$$

*Proof.* Theorem 5

*Property* 1*:*

Now, using the triangle inequality and the reverse triangle inequality twice we have

$$\begin{aligned} &|F_{n,u}(y \cdot u) + 1 - F_{n,u}(y \cdot u-) - |1 - F_{n,u}(y \cdot u-) - F_{n,u}(y \cdot u)| \\ &\quad - F_u(y \cdot u) - 1 + F_u(y \cdot u-)) + |1 - F_u(y \cdot u-) - F_u(y \cdot u)|| \\ &\leq 4\Big|F_{n,u}(y \cdot u) - F_u(y \cdot u)\Big|. \end{aligned} \tag{3.4}$$

Now, using (3.4)

$$\begin{aligned} \sup_{y \in \mathbb{R}^d} |D(y; F_n) - D(y; F)| &\leq \frac{4}{V_d} \int_{S^{d-1}} \sup_{y \in \mathbb{R}^d} \Big|F_{n,u}(y \cdot u) - F_u(y \cdot u)\Big| du \\ &\leq \frac{4}{V_d} \int_{S^{d-1}} \sup_{v \in \mathbb{R}^d} \Big|F_{n,u}(v) - F_u(v)\Big| du. \end{aligned}$$

Following the proof of Theorem 2 from Cuevas and Fraiman (2009), we have

$$\sup_{y \in \mathbb{R}^d} |D(y; F_n) - D(y; F)| \xrightarrow{\text{p}} 0 \text{ as } n \to \infty.$$

*Property* 2*:*

We use the central limit theorem. Let $A_1 = \{u : F_u(y \cdot u) < 1/2\}$,
$A_2 = \{u : F_u(y \cdot u) > 1/2\}$, $A_3 = \{u : F_u(y \cdot u) = 1/2\}$
$A_{n,1} = \{u : F_{n,u}(y \cdot u) < 1/2\}$, $A_{n,2} = \{u : F_{n,u}(y \cdot u) > 1/2\}$. Since $\text{Vol}(A_3) = 0$, we have

$$\begin{aligned} D(y; F_n) &= \frac{2}{V_d} \int_{S^{d-1}} \min(F_{n,u}(y \cdot u), 1 - F_{n,u}(y \cdot u-))\ du \\ &= \frac{2}{V_d}\Big(\int_{A_1} F_{n,u}(y \cdot u)\ du + \int_{A_2} 1 - F_{n,u}(y \cdot u-)\ du + B_{n1} + B_{n2}\Big), \end{aligned}$$

where

$$B_{n1} = \int_{A_1 \cap A_{n,2}} (1 - F_{n,u}(y \cdot u-) - F_{n,u}(y \cdot u))\ du,$$

and

$$B_{n2} = \int_{A_2 \cap A_{n,1}} (F_{n,u}(y \cdot u) + F_{n,u}(y \cdot u-) - 1)\ du.$$

Note that $|F_{n,u}(y \cdot u-) + F_{n,u}(y \cdot u) - 1| \leq 1$, which implies the following simple bounds:

$$\begin{aligned} |B_{n1}| &\leq \int_{A_1} \mathbb{1}(F_{n,u}(y \cdot u) > \frac{1}{2})\ du, \\ |B_{n2}| &\leq \int_{A_2} \mathbb{1}(F_{n,u}(y \cdot u) < \frac{1}{2})\ du. \end{aligned} \tag{3.5}$$

Now, we prove that $\sqrt{n}B_{n1}$ and $\sqrt{n}B_{n2}$ are $o_p(1)$ by showing that their expectation and variance converge to 0. Let $\delta_n = \frac{1}{n^{\frac{1}{3}+\epsilon}}$, $0 < \epsilon < \frac{1}{6}$, and $A_1^{\delta_n} = \{u \ : \ F_u(y \cdot u) \in (\frac{1}{2} - \delta_n, \frac{1}{2})\}$. We can now write the following.

$$\begin{aligned} \mathbb{E}(|B_{n1}|) &\leq \mathbb{E}\left(\int_{A_1} \mathbb{1}(F_{n,u}(y \cdot u) > \frac{1}{2})\ du\right), \\ &= \int_{A_1^{\delta_n}} P(F_{n,u}(y \cdot u) > \frac{1}{2})\ du + \int_{(A_1^{\delta_n})^c \cap A_1} P(F_{n,u}(y \cdot u) > \frac{1}{2})\ du. \end{aligned}$$

Note that for the second term, we can use the Bernoulli special case of Hoeffding's inequality (rf. Hoeffding (1963)) with $t = \delta_n$. That is, if $H(n)$ is a binomially distributed random variable with $n$ trials and success probability $p$, for $t > 0$

$$P(H(n) \geq (p + t)n) \leq \exp\{-2t^2 n\}.$$

In our case, this inequality implies

$$\int_{(A_1^{\delta_n})^c \cap A_1} P(F_{n,u}(y \cdot u) > \frac{1}{2}) du \leq C_1(F, y) \exp\{-2n^{\frac{1}{3}-2\epsilon}\},$$

where $C_1(F, y)$ is a finite, bounded constant that depends on $F$ and $y$. Now, looking at the second term, let $\Delta_n = n^{-(\frac{1}{3}+\epsilon')}$, $0 < \epsilon' < \epsilon$. Choosing $\Delta_n$ in this way implies that, for $u \in A_1^{\delta_n}$, $\Delta_n < |\frac{\lfloor \frac{n}{2} \rfloor + 1}{n} - F_u(y \cdot u)|$. Thus,

$$\int_{A_1^{\delta_n}} P(F_{n,u}(y \cdot u) > \frac{1}{2})\, du \leq \int_{A_1^{\delta_n}} \sum_{|\frac{i}{n} - F_u| > \Delta_n} \binom{n}{i} (F_u(y \cdot u))^i (1 - F_u(y \cdot u))^{n-i} du.$$

By Lorentz (1986), (see (8) on page 15), for any $k > 0$,

$$\int_{A_1^{\delta_n}} \sum_{|\frac{i}{n} - F_u| > \Delta_n} \binom{n}{i} (F_u(y \cdot u))^i (1 - F_u(y \cdot u))^{n-i} du \leq \int_{A_1^{\delta_n}} \frac{C_2(k)}{n^k} du \leq V_d \frac{C_2(k)}{n^k}.$$

From the fact that $C_2$ is finite and does not depend on $u$, we can say the first term is $o(\frac{1}{n^k})$. Thus,

$$\mathbb{E}(|B_{n1}|) = o\left(\frac{1}{n^k}\right). \tag{3.6}$$

Now note that from (3.5)

$$\left(\frac{|B_{n1}|}{\text{Vol}(A_1)}\right)^2 < \frac{|B_{n1}|}{\text{Vol}(A_1)} < 1,$$

which implies

$$\mathbb{E}(B_{n1}^2) < \mathbb{E}(|B_{n1}|)\text{Vol}(A_1).$$

Thus,

$$\mathbb{E}(B_{n1}^2) = o\left(\frac{1}{n^k}\right), \tag{3.7}$$

(3.6) and (3.7) together imply $\sqrt{n} B_{n1} \xrightarrow{\text{p}} 0$. By a similar (but lengthier) argument, we have $\sqrt{n} B_{n2} \xrightarrow{\text{p}} 0$. This leads to

$$\sqrt{n} D(y; F_n) = \frac{2\sqrt{n}}{V_d} \left[ \int_{A_1} F_{n,u}(y \cdot u)\, du + \int_{A_2} 1 - F_{n,u}(y \cdot u-)\, du + o_p(n^{-\frac{1}{2}}) \right]$$

$$= \frac{2}{\sqrt{n}V_d}\left[\int_{A_1}\sum_{i=1}^{n}\mathbb{1}(X_i\cdot u \le y\cdot u)\ du + \int_{A_2}\sum_{i=1}^{n}\mathbb{1}(X_i\cdot u \ge y\cdot u)\ du\right] + o_p(1).$$

By switching the order of summation, we now get that

$$\sqrt{n}D(y;F_n) = \frac{2}{\sqrt{n}V_d}\sum_{i=1}^{n}\left[\int_{A_1}\mathbb{1}(X_i\cdot u \le y\cdot u)\ du + \int_{A_2}\mathbb{1}(X_i\cdot u \ge y\cdot u)\ du\right] + o_p(1),$$

$$= \frac{1}{\sqrt{n}}\sum_{i=1}^{n}\frac{2}{V_d}h(X_i) + o_p(1), \tag{3.8}$$

where $h(X_i)$ do not depend on $n$ and are given by

$$h(X_i) = \frac{2}{V_d}\left[\int_{A_1}\mathbb{1}(X_i\cdot u \le y\cdot u)\ du + \int_{A_2}\mathbb{1}(X_i\cdot u \ge y\cdot u)\ du\right]. \tag{3.9}$$

Now note,

$$\mathbb{E}(h(X)) = \frac{2}{V_d}\left[\int_{A_1}F_u(y\cdot u)\ du + \int_{A_2}1 - F_u(y\cdot u-)\ du\right] = D(y;F_n)),$$

$$\mathbb{E}(h(X)^2) = \left(\frac{2}{V_d}\right)^2\left[\int_{A_1}\int_{A_1}P(X\cdot u_1 \le y\cdot u_1, X\cdot u_2 \le y\cdot u_2)\ du_1du_2\right.$$

$$+ 2\int_{A_1}\int_{A_2}P(X\cdot u_1 \le y\cdot u_1, X\cdot u_2 \ge y\cdot u_2)\ du_1du_2$$

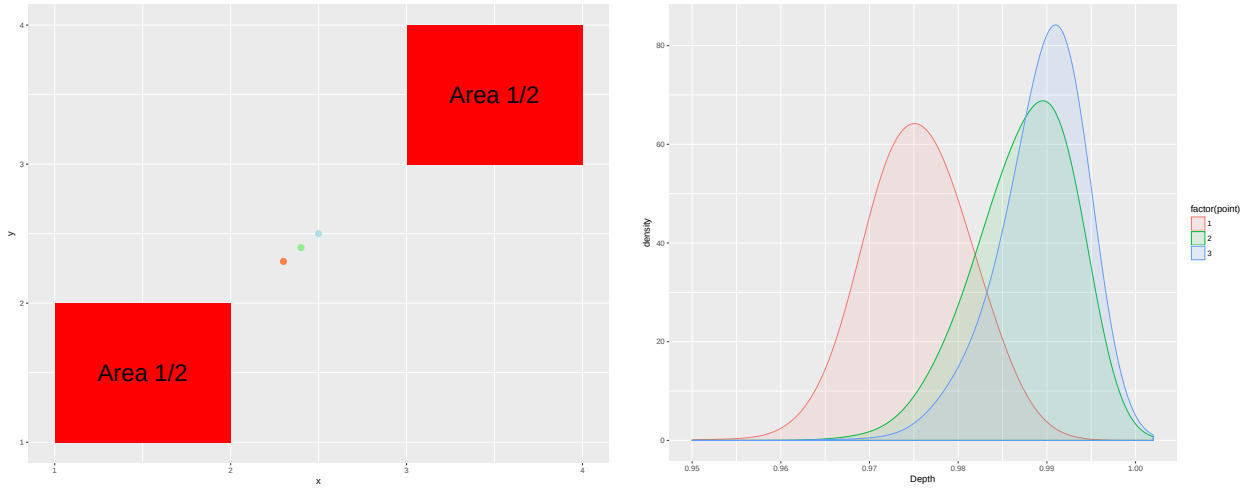
$$\left.+ \int_{A_2}\int_{A_2}P(X\cdot u_1 \ge y\cdot u_1, X\cdot u_2 \ge y\cdot u_2)\ du_1du_2\right].$$

This implies,

$$\mathbb{V}ar(h(X)) = \left(\frac{2}{V_d}\right)^2\left[\int_{A_1}\int_{A_1}P(X\cdot u_1 \le y\cdot u_1, X\cdot u_2 \le y\cdot u_2)\right.$$

$$\left.- P(X\cdot u_1 \le y\cdot u_1)P(X\cdot u_2 \le y\cdot u_2)\ du_1du_2\right.$$

$$+ 2\int_{A_1}\int_{A_2}P(X\cdot u_1 \le y\cdot u_1, X\cdot u_2 \ge y\cdot u_2)$$




(a) (b)

Figure 3.9: (a) Birds eye view of density with points at which we calculate depth. (b) Asymptotic density estimates of the points in Figure (a).

$$
\begin{aligned}
&\quad - P(X \cdot u_1 \leq y \cdot u_1) P(X \cdot u_2 \geq y \cdot u_2)\, du_1 du_2 \\
&+ \int_{A_2} \int_{A_2} P(X \cdot u_1 \geq y \cdot u_1, X \cdot u_2 \geq y \cdot u_2) \\
&\quad - P(X \cdot u_1 \geq y \cdot u_1) P(X \cdot u_2 \geq y \cdot u_2)\, du_1 du_2 \Bigg].
\end{aligned}
$$

Notice (3.8) is written in the form $\frac{1}{\sqrt{n}} \sum_{i=1}^{n} h(X_i) + o_p(1)$, where $h(X_i)$ is independent of $n$ (and has finite variance). By the central limit theorem and Slutsky's lemma,

$$
\frac{\sqrt{n}(D(y; F_n) - \mathbb{E}(h(X)))}{\sqrt{\mathbb{V}ar(h(X))}} \xrightarrow{\mathrm{d}} Z \ , \ Z \sim N(0, 1).
$$

This completes the proof. □

In other words, $D_n$ is asymptotically normal when there is no non-negligible proportion of unit vectors such that $F_u(y \cdot u) = \frac{1}{2}$. This condition depends both on $F$ and $y$ and implies that $F$ is continuous, and $y$ is not a point of symmetry, nor does it lie in a 'hole' of the support of $F$. When $A_3$ is non-negligible, there is a proportion of unit vectors such that $F_u(y \cdot u) = \frac{1}{2}$.

Recall we estimate $F_u(y \cdot u)$ with $\min(F_{n,u}(y \cdot u), 1 - F_{n,u}(y \cdot u-))$. Over this proportion, $\min(F_{n,u}(y \cdot u), 1 - F_{n,u}(y \cdot u-)) < F_u(y \cdot u) = \frac{1}{2}$ thus, we will always underestimate $F_u(y \cdot u)$ for $u \in A_3$. This fact creates skewness in the asymptotic distribution that depends on the size of $A_3$. Figure 3.9a shows a density in $\mathbb{R}^2$ such that the mass is uniformly distributed between the two disjoint rectangles shown. Figure 3.9b shows asymptotic density estimates, using a sample size of $n = 100$[2], for the depth of each of the coloured points in Figure 3.9a, based on 10000 sample depths. Notice that the higher proportion of angles for which $F_u(y \cdot u) = \frac{1}{2}$ leads to more skewness in the asymptotic distribution and the blue point is a point of symmetry.

### 3.3.2 Comparison to Other Depth Functions

Another nice property of integrated depth measures is that their contours appear much smoother than the contours of many popular depth measures. Figure 3.10 shows a comparison of Tukey, IRW, simplicial and Cuevas depth contours for a 2-dimensional multivariate normal sample, with 5 outliers in a cluster at the top right. This data can be found in Appendix B. Notice that the Cuevas and IRW depths are very smooth compared to Tukey and simplicial depth. The similarity between the Cuevas and IRW depths is also very apparent, this is most likely due to the fact that the contours of Cuevas and IRW depth are both converging to the same level sets (Cuevas and Fraiman, 2009). However if you look at Figure 3.6, which shows the 3-dimensional surface of the depth measure, it is clear that the depths are not as similar as the contours make them appear. Notice that the deeper regions of the IRW and Cuevas depth are not pulled upwards nearly as much as the ones in Tukey and simplicial depth; the outliers appear to affect the geometry of the inner contours of simplicial and Tukey depth more than the inner contours of IRW and Cuevas depths. The deeper regions of the integrated depths appear to be less affected by outliers.

[2]The plots looked similar for a sample size of $n = 10000$.

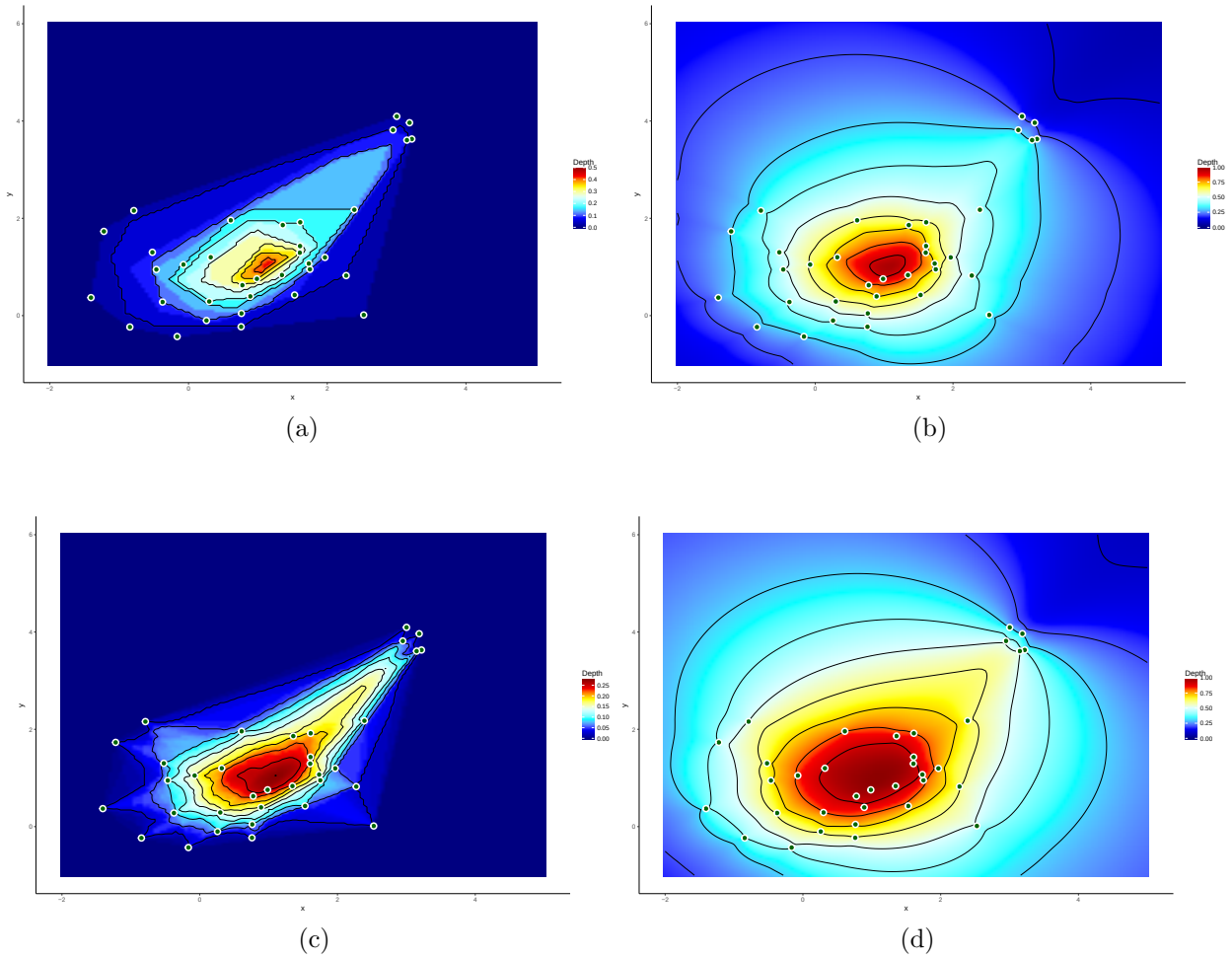


Figure 3.10: (a) Tukey depth contours (b) IRW depth contours (c) simplicial depth contours (d) Normalized Cuevas (to (0,1]) depth contours (Same point set as Figure 3.6).

## 3.4 The IRW Deepest Point

As mentioned in Chapter 1, it is natural to consider the deepest point as a measure of location; the deepest point refers to the point of highest centrality.

**Definition 17.**

The *integrated rank-weighted sample median*, denoted $\mu(\mathbf{X_n})$, is defined as

$$\mu(\mathbf{X_n}) = \underset{y \in \mathbb{R}^d}{\operatorname{argmax}}\, D(y; F_n). \tag{3.10}$$

Note we can also define the population median as

$$\mu(F) = \underset{y \in \mathbb{R}^d}{\operatorname{argmax}} D(y; F). \tag{3.11}$$

The maximizer is not always unique. In such cases we can define the *unique IRW median* as the average of the maximizing region. We suggest computing this median via convex optimization, or hill-climbing techniques, even though it is not in general convex and it has not been shown that IRW depth is decreasing along rays in all set ups. We conjecture that in fact it is always decreasing along rays, as we have been unable to find a counterexample; this provides the basis for our recommendation.

Note that we can rewrite (3.11) as

$$\underset{y \in \mathbb{R}^d}{\operatorname{argmin}} \int_{S^{d-1}} \frac{P(X=y)}{2} + \left| \frac{1}{2} - F_u(y \cdot u) - \frac{P(X=y)}{2} \right| du \approx \underset{y \in \mathbb{R}^d}{\operatorname{argmin}} \int_{S^{d-1}} \left| \frac{1}{2} - F_u(y \cdot u) \right| du.$$

So, in effect, this estimator's projections have the minimum average distance $\mathbf{d}$ to the univariate median, where $\mathbf{d}(x, y) = |x - y|$. One could replace $\mathbf{d}$ with a different distance measure, such as $\mathbf{d}^*(x, y) = |x - y|^a,\ a > 0$. In fact, setting $a$ to 2 gives the Cuevas median. Note that when $d$ is large, we can use the following estimator to save time in computing.

**Definition 18.**

The *in-sample integrated rank-weighted median*, denoted $\hat{\mu}(\mathbf{X_n})$, is defined as

$$\hat{\mu}(\mathbf{X_n}) = \underset{y \in \mathbf{X_n}}{\operatorname{argmax}} D(y; F_n). \tag{3.12}$$

This can be paired with the Monte Carlo algorithm described in Section 3.3 to compute the approximate deepest point in $O(mn^2d)$ time. This algorithm can be used to produce an approximation that converges to the deepest point in the sample as $m \to \infty$.

### 3.4.1 Properties

Before discussing the properties of the IRW medians, it is useful to discuss the relationship between IRW depth and Tukey depth. Recall Tukey depth (Definition 5) and its median (Definition 6) from Chapter 1. The Tukey median $T(\mathbf{X}_n)$ maximizes Tukey depth (Tukey, 1974). In Chapter 1 we defined Tukey depth in terms of numbers of points in half-spaces for which the query point, $y$, lies on the boundary. However, Tukey depth can also be viewed as minimizing the univariate depth described in Section 3.2, $d_u()$, over the unit hypersphere.

**Definition 19.**

The *Tukey depth* of a point $y \in \mathbb{R}^d$ with respect to the point set $\mathbf{X}_n$ (with possible multiplicities), denoted $D_t(y; F_n)$, is the minimum univariate depth achieved by $y$ when considering all unit vectors;

$$D_t(y; F_n) = \inf_{u \in S^{d-1}} d_u(y; F_n).$$

We now relate Tukey depth with IRW depth.

**Theorem 6.**

The integrated rank-weighted depth satisfies

$$D(y; F_n) \geq 2\frac{D_t(y; F_n)}{n}.$$

*Proof.* Theorem 6

Let $N$ be the number of sections into which $F_n$ divides the unit hypersphere. From the definitions,

$$D(y; F_n) = \frac{2}{n}\sum_{i=1}^{N} w_i d_{u_i}(y; F_n) \geq \frac{2}{n}\inf_{i \in \{1,\ldots \mathrm{N}\}}\{d_{u_i}(y; F_n)\} = \frac{2}{n}D_t(y; F_n).$$

This completes the proof. □

**Corollary 1.**

For any point set $\mathbf{X_n}$,

$$\frac{2}{n}\lfloor\frac{n}{d+1}\rfloor \leq D(\mu(\mathbf{X_n}); F_n) \leq 1.$$

This corollary ensures that the depth of the deepest point is at least $\frac{2}{d+1}$.

*Proof.* Corollary 1

It follows directly from the following fact, shown by Donoho and Gasko (1992),

$$\sup_{y\in\mathbb{R}^d} D_t(y; F) \geq \lfloor\frac{n}{d+1}\rfloor.$$

This completes the proof. □

Recall the breakdown point from Chapter 1 (Definition 1). The next Theorem bounds the breakdown of the IRW median below and shows it has breakdown at least as high as the Tukey median.

**Theorem 7.**

The finite sample breakdown of the IRW median satisfies

$$\epsilon^*(\mu, n) \geq \frac{\lceil\frac{n}{d}\rceil}{\lceil\frac{n}{d}\rceil + n}.$$

*Proof. Theorem 1.*

Let $0 < \alpha,\ \beta < 1$. Set $m = \lfloor n\epsilon^*(T, n)\rfloor - 1$; the proportion of points for which the Tukey median is not corrupt. Without loss of generality assume that $n - m$ points lie inside the unit hypersphere and the other points may be located anywhere (we can simply scale the points down and recenter them at 0). Let $F_n$ be the empirical distribution determined by

$X \cup Y$. We know from the finite sample breakdown of the Tukey median (Liu et al., 2017), $T$, and the relationship between IRW depth and Tukey depth that:

$$D(y; F_n) = \alpha(2 \left\lfloor \frac{m}{n} \right\rfloor) + (1-\alpha)2a > 2 \left\lfloor \frac{m}{n} \right\rfloor,$$

where $a > 2 \left\lfloor \frac{m}{n} \right\rfloor$. Now, it should be clear that in order for $d_u(y; F_n) > m$, the projection of $y$ onto $u$ must lie in the unit hypersphere. Consider an arbitrary point outside $S^{d-1}$, call it $cu^*$, where $c > 1$ and $u^*$ is a unit vector. Note that $cu^*$ has IRW depth that satisfies $D(cu^*; F_n) < \beta(2 \left\lfloor \frac{m}{n} \right\rfloor) + (1-\beta)$, where $1-\beta$ is the proportion of unit vectors for which $cu^*$'s projection lies inside the unit hypersphere. $1-\beta$ is a function of the space between the hyperplanes $u^* \cdot u = 1/c$ and $u^* \cdot u = 0$ which is approaching 0 as $c$ approaches infinity. Clearly this is decreasing in $c$. We choose $c$ large enough such that $1-\beta < (1-\alpha)a$, then $D(cu^*; F_n) < D(T, F_n)$. Therefore, the deepest point lies inside the hypersphere with center at the origin and radius $c$. From this, $\epsilon^*(\mu(F), n) \geq \epsilon^*(T, n)$ and $\lim_{n\to\infty} \epsilon^*(\mu(F), n) \geq \lim_{n\to\infty} \epsilon^*(T, n)$. □

A direct consequence of Theorem 1 is the following, which is given without proof.

**Corollary 2.**

The finite sample breakdown of the IRW median satisfies

$$\epsilon^*(\mu, n) \geq \frac{\lceil \frac{n}{d} \rceil}{\lceil \frac{n}{d} \rceil + n}.$$

The asymptotic breakdown point of the IRW median satisfies

$$\lim_{n\to\infty} \epsilon^*(\mu, n) \geq \frac{1}{d+1}$$

for all $F$ and

$$\lim_{n\to\infty} \epsilon^*(\mu(F), n) \geq \frac{1}{3}$$

when $F$ is half-space symmetric.

Next, we establish the consistency of the two IRW medians.

**Theorem 8.**

Let $supp(F)$ be the support of $F$. The integrated rank-weighted sample median and the in-sample integrated rank-weighted median are weakly consistent. Let $F$ be a distribution on $\mathbb{R}^d$, and $\mathbf{X_n}$ be a random sample from $F$. Then

$$\mu(\mathbf{X_n}) \xrightarrow{\text{p}} \underset{y\in\mathbb{R}^d}{\text{argmax}}\, D(y;F)$$

when $n \to \infty$. Also

$$\hat{\mu}(\mathbf{X_n}) \xrightarrow{\text{p}} \underset{y\in supp(F)}{\text{argmax}}\, D(y;F)$$

when $n \to \infty$.

Theorem 5 implies both IRW medians are weakly consistent for their population counterparts. The deepest point may be outside of the support of $F$ in scenarios where the support is a proper subset of $\mathbb{R}^d$, as in Figure 3.5b.

**Example 3** (Iris Data cont.)**.**

Here we demonstrate the robustness of the IRW median. Figure 3.11 shows a scatter plot of the Iris data with 20% of the data in each group corrupted. The corrupted data was randomly mislabelled to one of the other two groups. The sample mean vector and IRW median before and after corruption are shown. Even with this high proportion of corrupted data, the IRW median remains close to the uncorrupted median. The group in the bottom left has mislabelled data very far from its correctly labelled ones, however, the IRW median barely moves. The green sample's median estimator is pulled half as far as the sample mean of the green sample is pulled, but both are pulled farther than in the bottom left group. The close proximity of some of the green points labelled as red points make them more difficult to interpret as outliers. It may be surprising that 'closer' mislabelled points have a greater impact on the estimator than 'far' ones. One could think that it is easier to identify 'far' observations as mislabelled rather than 'closer', more ambiguous, ones.

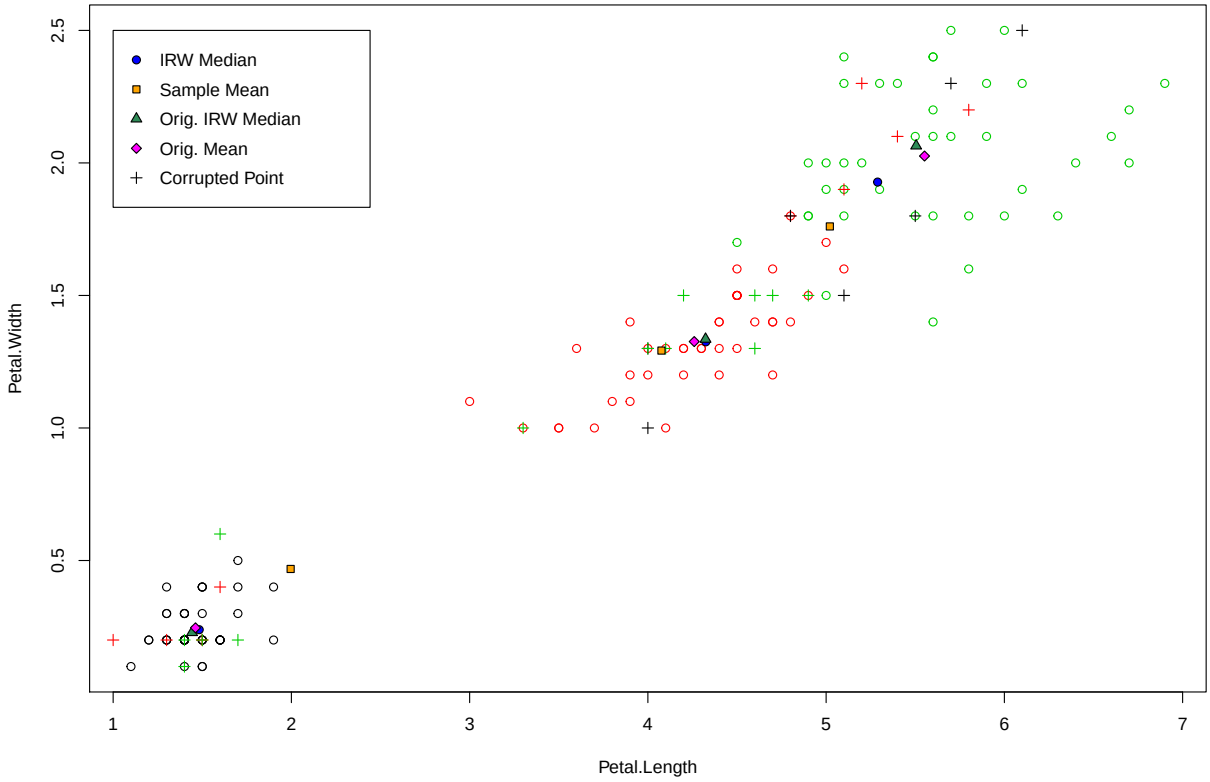
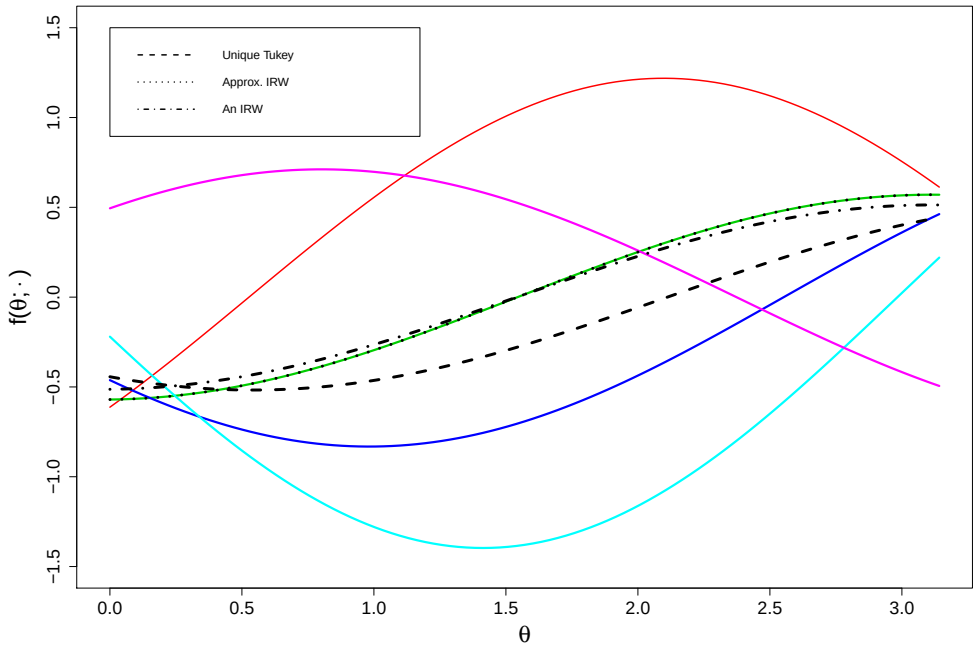



Figure 3.11: Example 3, Corrupted Iris Data and the IRW median

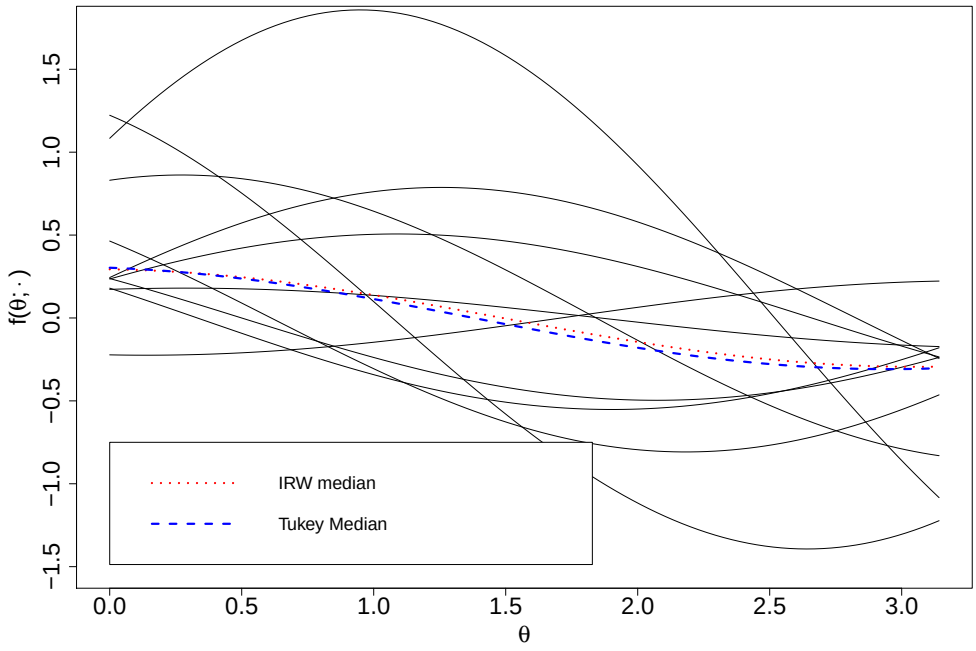
### 3.4.2 Geometric Interpretation

Recall the trajectories from Section 2.2. It can be useful to visualize these medians in the form of trajectories. Figure 3.12a shows the trajectories of each point in Figure 3.2 for $\theta$ between 0 and $\pi$. A point is deep in terms of integrated rank-weighted depth if its trajectory is often 'sandwiched' between other trajectories. A point is deep in terms of Tukey depth if its entire trajectory is surrounded by the entire trajectories of many other points. Figure 3.12a also shows trajectories for different medians. The IRW median has depth 0.8, which means it is 'surrounded' by a minimum of two curves on average. Notice also how the IRW median is also a Tukey median; it is in fact always surrounded by 2 curves and thus, has Tukey depth 2. In fact any Tukey median is also an IRW median in this case; they are both contained in the set of points which have IRW depth 0.8 or Tukey depth 2. In fact any Tukey median is

also an IRW median in this case; they are both contained in the set of points which have IRW depth 0.8 or Tukey depth 2. The two medians do not always coincide, see Figure 3.12b, where the trajectories of a 10 point multivariate normal sample as well as the trajectories of the three medians are shown. Notice that the Tukey median here is again non-unique, all of them having Tukey depth 0.4, whereas the IRW median is unique with IRW depth 0.84. The displayed Tukey median is obtained by averaging over a region that includes the IRW median. Note also that Theorem 6 holds here; both medians have IRW depth above $\frac{2}{3}$.




(a)




(b)

Figure 3.12: (a) Trajectories, $u_\theta = (\cos\theta, \sin\theta)$, of the five points from Figure 3.2 and different medians of this point set. (b) Tukey median vs. IRW median for a 10 point multivariate normal sample.

# Chapter 4

# Applications and Conclusion

In this chapter we consider two applications of IRW depth and the two medians, using real data. We use high dimensional data to emphasize the computability of these statistics. The first application is a permutation test involving the projection median.

## 4.1 Testing for Location Difference using the Projection Median

To demonstrate a practical use of the projection median we implement a permutation test for a difference in location between two groups. This is a nonparametric hypothesis test designed to detect if two populations differ in terms of their location parameter. The procedure is analogous to the one described by Zhang and Pan (2016). However, their test statistic involves the mean vector and sample covariance matrix. We replace these estimators with the projection median and a robust estimate of the covariance matrix. The test proceeds as follows.

**Algorithm 5** (Permutation Test of Zhang and Pan, 2016)**.**

1. Select $B_1$ subsets of size $k$ from the set of features (covariates, predictors).

2. Calculate the observed test statistic, $T^2_{obs}$, (A function of the selected subsets and the data).

3. Regroup the observations and randomly split them into two groups, the same size as the original groups $B_2$ times.

4. Calculate $T_1^{2*}$ ... $T_{B_2}^{2*}$ for each of the sets of groups from step 3.

5. Calculate the p value $\frac{1}{B_2}\sum_{i=1}^{B_2}\mathbb{1}(T_{obs}^2 > T_i^2)$.

Let $n_1$ and $n_2$ denote the number of observations in group 1 and 2 respectively, the test statistic is:

$$T^2 = \frac{1}{B_1}\sum_{i=1}^{B_1}(M(\mathbf{X_{k,i}}) - M(\mathbf{Y_{k,i}}))'(S_p(\frac{1}{n_1}+\frac{1}{n_2}))^{-1}(M(\mathbf{X_{k,i}}) - M(\mathbf{Y_{k,i}})),$$

where $M(A)$ is the projection median of $A$, $\mathbf{X_{k,i}}$ and $\mathbf{Y_{k,i}}$ are the observations of the $k$ features selected in subset $i$ and $S_p$ is the usual pooled sample variance matrix, but only using the deepest 50% of observations in each group, with respect to IRW depth. In the original paper (Zhang and Pan, 2016), the test statistic can be thought of the average Hotelling's $T^2$ statistic, over many subsets of the data. $T^2$ here has the same interpretation, however, it is also robust.

**Example 4** (Prostate Data)**.**

We now look at a data set which has very high dimensions compared to its sample size. We use the prostate cancer data set obtained by Welsh et al. (2001), which consists of 102 observations of 6033 different gene expression values. There are 50 subjects in the control group and 52 with prostate cancer. We would like to determine if there is a difference in distribution of the genetic expression values between the two groups. This difference would indicate gene expression values may be used as a marker for prostate cancer and could contribute to a better understanding of the disease. To accomplish this we will test for a location difference (and in turn distribution since we have no reason to believe that, under the null, their covariance matrices would differ) between these two groups. We are unaware of the procedure followed to collect this data and so we think it best to use a robust test.

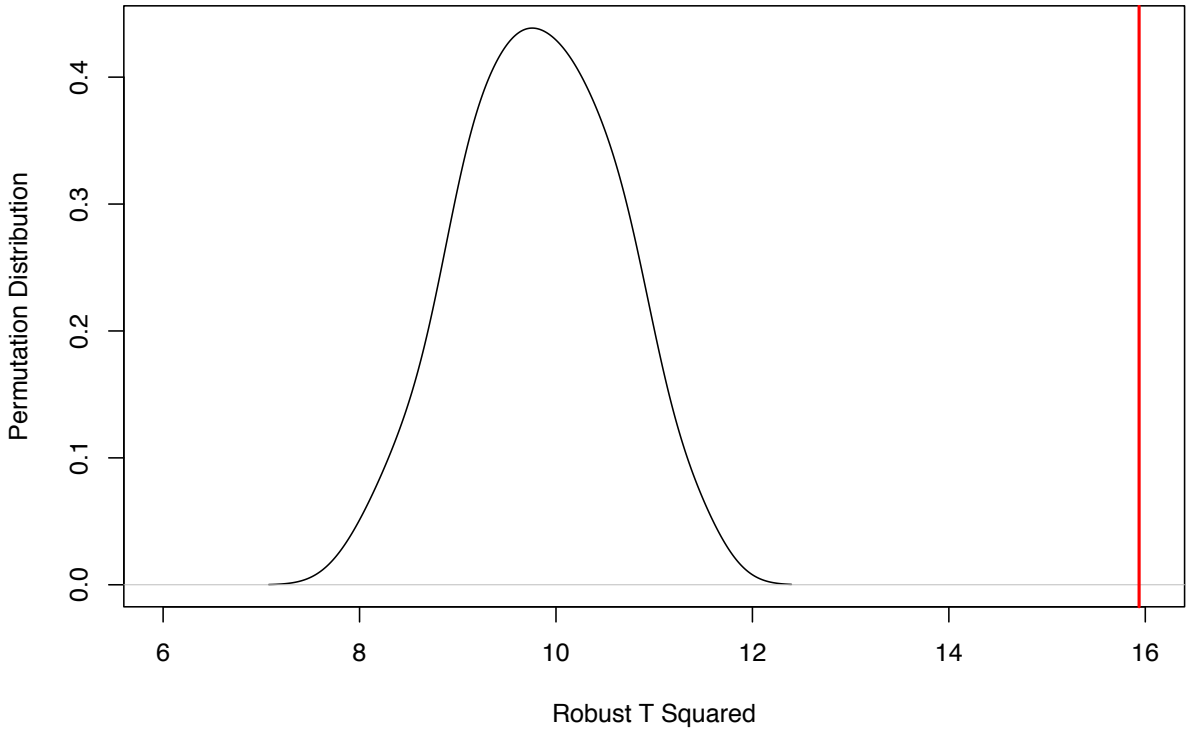
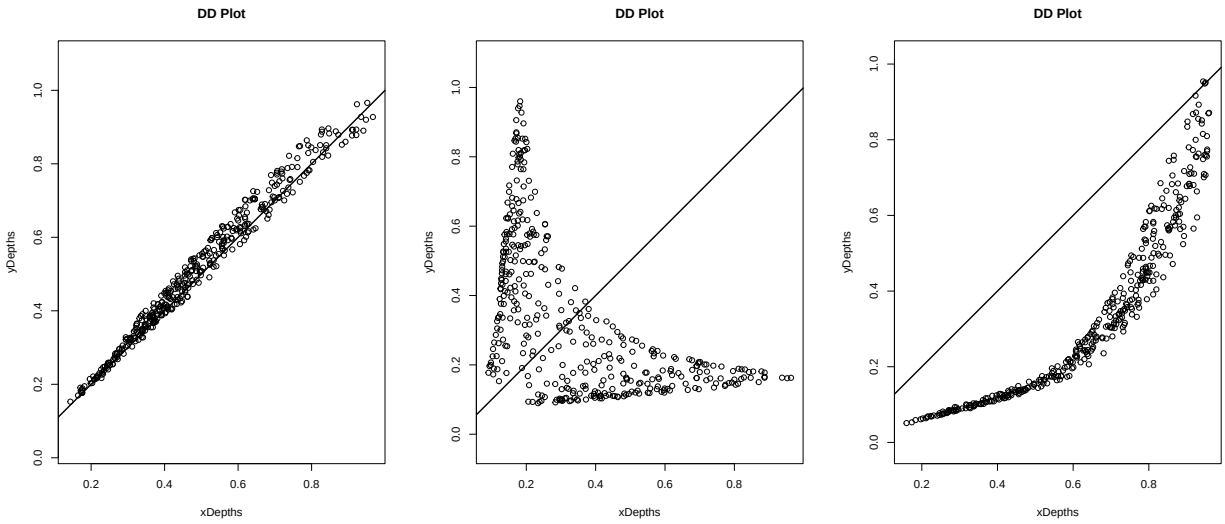



Figure 4.1: Permutation Distribution

We choose $k = 5$, $B_1 = 100$ and $B_2 = 100$. Zhang and Pan (2016) explain that there is a trade off between computational efficiency and the power of the test when it comes to these parameters, especially $k$. They mention that the power curves do not change much after $B2,\ B1 > 100$. The best $k$ in terms of power is $\lfloor \frac{n}{2} \rfloor$, however higher $k$ values require more computational effort. We have no reason to believe the results would be very different with the modified test statistic, however it could be done in the interest of rigour. Robustness does not usually imply a large loss of power or a change in nature of the relationship between test parameters and power.

To calculate the projection median and the depth values for each subset we use the approximation algorithms discussed in Chapters 3 and 4 with the same $m = 10\ 000$ vectors. Figure 4.1 shows the smoothed permutation distribution with $T^2_{obs} = 15.9$ indicated by a red




(a) (b) (c)

Figure 4.2: dd-plot for equal distributions, location shift and scale difference.

vertical line. Clearly $T^2_{obs}$ is very extreme with respect to the permutation distribution; we never saw a value as large as $T^2_{obs}$. Therefore, we have a p-value of $\sim$0, and we should reject the hypothesis that the two groups have the same distribution. We would conclude that there is significant evidence to say that the distribution of gene expression values of a male who has prostate values differs from the distribution of males without prostate cancer.

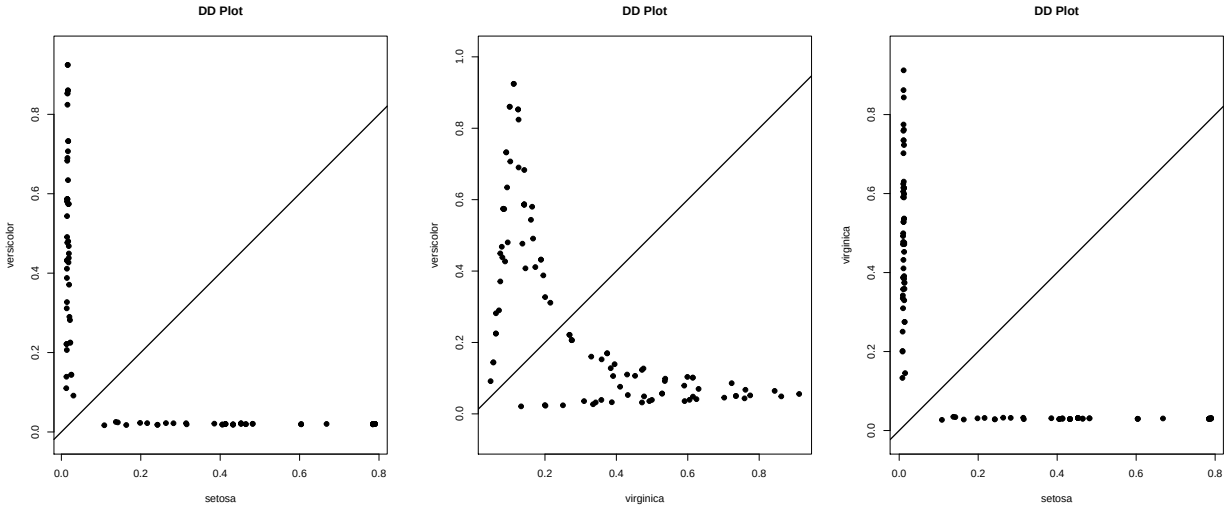
Zhang and Pan (2016) mention that when it comes to tests with this kind of data, invariance under non-uniform scaling is important. Since both the projection median and IRW depth are affected by this transformation, the test was run again with the data in each group scaled by its median average deviation. Specifically for each gene, within each group, the set of expression values was scaled by their median average deviation. The test produced similar results which are not reported here [1].

## 4.2 DD-Plots with IRW Depth

We now demonstrate and discuss a basic use of this depth measure, aside from the location estimation described above. We continue Examples 3 and 4. All of the coding has been done with the R software.

Specifically, we demonstrate the use of the dd-plot, which is a dimension free plot used to assess how similar the parent distribution of two samples, say $\mathbf{X_{n_1}}$ and $\mathbf{Y_{n_2}}$, are. For a thorough introduction see the excellent description by Li and Liu (2004) or see Liu et al. (1999) where dd-plots were first introduced. To construct the plot we calculate the depth, with respect to each sample, of each point in the combined sample. Therefore, for each point we have two depths which are plotted as pairs. In other words, for each $y \in \mathbf{X_{n_1}} \cup \mathbf{Y_{n_2}}$, we plot $(D(y; F_{n1}), D(y; G_{n2}))$, where $F_{n1}$ is the empirical distribution of $\mathbf{X_{n_1}}$ and $G_{n2}$ is the empirical distribution of $\mathbf{Y_{n_2}}$. This is similar to a qq-plot except we are plotting depths against each other rather than quantiles. These plots describe different distributional characteristics depending on how the data is transformed. To assess location, the dd-plot should be made from the raw data or scale normalized data (using a robust measure of scale) if the scales differ widely. This is different from affine invariant depths, where the scale does not play a role in assessing location with dd-plots. Figure 4.2a-4.2c show dd-plots for samples from distributions that are the same, differ only in location and differ in scale respectively. In particular, in Figure 4.2a, the points follow a linear pattern and are somewhat evenly scattered about the $y = x$ line, which is typical when there is no distributional difference. Notice in Figure 4.2b that the location shift is associated with triangular shaped patterned dd-plots, with the tip of the triangle being somewhere around the bottom left corner or equivalently the top right corner. To assess scale we must normalize each group by its median. Note that even though the depth measure is translation invariant, we will see a difference in the plots since the depth values are taken with respect to each sample, which are location normalized by two

[1]We also ran the test for some simulated data from a multivariate t, where both samples were from the same distribution. The p-value was 0.35.




(a) (b) (c)

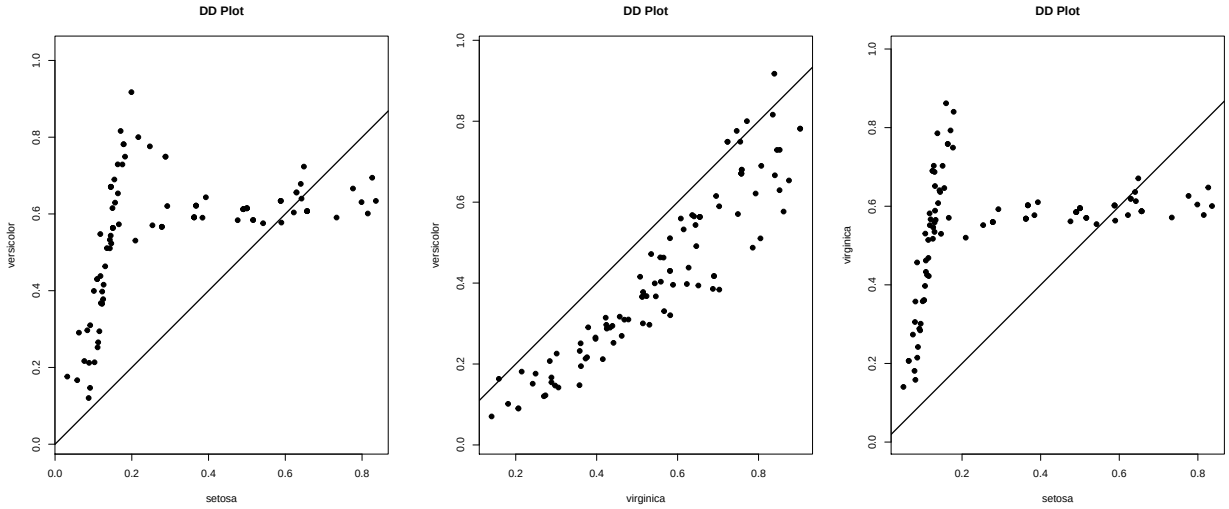
Figure 4.3: dd-plots for different pairs of iris species using the raw data

different values. Scale differences are characterized by banana shaped dd-plots as in Figure 4.2c, such as the one in the Example 3. Other characteristics such as kurtosis and skewness can be assessed (Liu et al., 1999; Li and Liu, 2004). Li and Liu (2004) also describe formal location and scale tests associated with dd-plots.

**Example 3** (Iris Data cont.)**.**

Figures 4.3a-4.3c shows dd-plots for the three pairs of classes with the uncontaminated, raw Iris data. The location differences between the 3 classes are apparent; all 3 exhibit that lower triangle appearance. Figure 4.4a-4.4c show the dd-plots for the same 3 pairs, but each class has been centred by the IRW median. Note the banana shape is most pronounced in the first and last plots, and is less distinct in the middle one. This is a reflection of a more pronounced scale difference, which is apparent in Figure 3.11.

It seems that dd-plots using IRW depth for very high dimensional, sparse data do not exhibit the same patterns as data with $n > d$. Our experience is that in-sample depth values from a sparse sample tend to be somewhat uniform. For example the in-sample depth values in the prostate data for the control group are in the range (0.5,0.54). This uniformity




(a) (b) (c)

Figure 4.4: dd-plots for different pairs of iris species using location centred data.

can change the way location differences appear in the plot. It seems useful to do a small simulation study of dd-plots that have the same $n$ and $d$ as the specific dataset at hand before using this plot to compare samples.

**Example 4** (Prostate Data cont.)**.**

We return to the prostate cancer data, but use all 6033 different gene expression values (Welsh et al., 2001). We use $m =$ 100 000 vectors for the depth values. Figure 4.5 show the dd-plot of the raw prostate data. Notice the range of depth values in the samples is very small, which is a reflection of the large dimension and/or sparseness. In this case the lower bound on the deepest point is $\frac{1}{3017}$, this lower bound reflects the curse of dimensionality. The points are, however, relatively far away from the $y = x$ line,

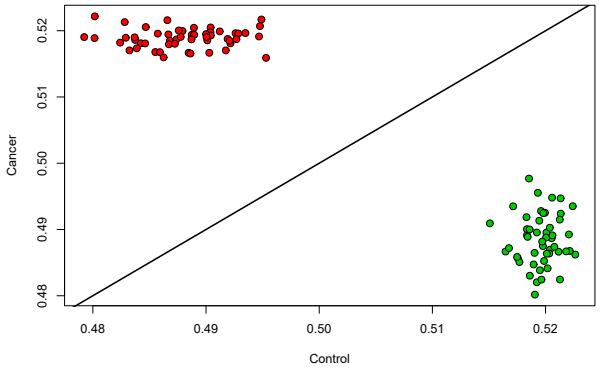


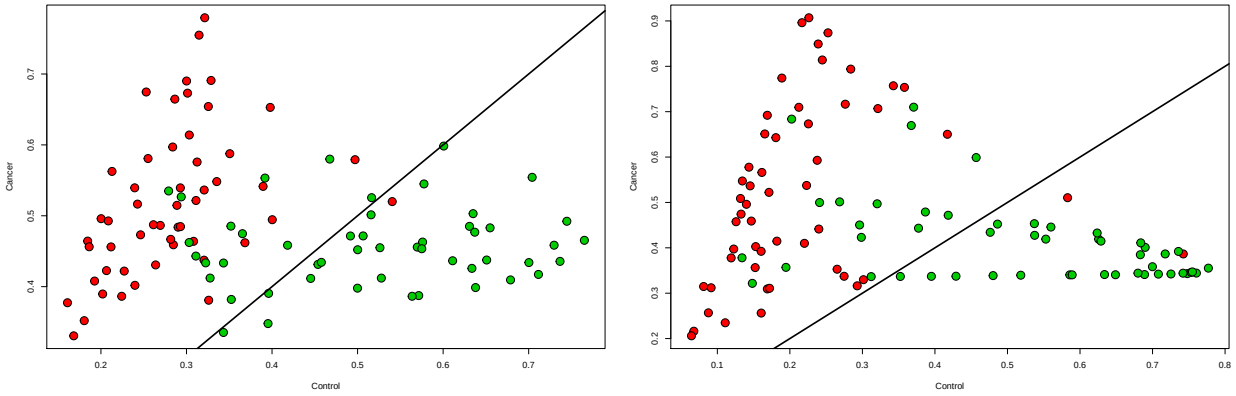

Figure 4.5: dd-plot of raw prostate data, red for the cancer group and green for the control, using the 6033 gene expression values.

and thus we would suspect a location difference between these two groups. A dd-plot (not shown) was also made for the same data after the axes were brought to the same scale (after splitting into control and cancer). There was an even more pronounced location difference. The discrepancy between these two plots suggests that some of the features that displayed more variability in the raw data set may have less of a location difference than ones that displayed less variability.

Figure 4.6a and 4.6b show the dd-plots for the 5 and 2 most significant genes in terms of a t-test for location difference on each coordinate, respectively. Notice the much clearer triangular appearance. After reducing the dimension, the plots seem to take more of the 'standard' appearance in the presence of a location shift. This leads us to consider taking many, say $k$, subsets of the data features or variables and producing plots for those subsets. Further one could perform a modified version of the dd-plot test for distributional differences (Li and Liu, 2004), by performing the test on each subset. The final test statistic could be the average of those $k$ statistics, such as is done by Zhang and Pan (2016). Performing a test of this type may eliminate the sparsity issues while keeping some of the geometrical features of the data. Note simply testing each coordinate fails to account for dependencies between variables; it does not preserve geometrical features.

## 4.3 Discussion

The projection median and depth measures introduced and discussed in this thesis have many potential other applications, including classification, outlier trimming and scale assessment. Classification is done via simply assigning the new observation to the group in which it has the largest depth. For more information see Mosler and Hoberg (2006). Outlier trimming can be done, for instance, by removing observations of low depth. Scale can be assessed via a *scale curve*. A scale curve plots the volume of the convex hull of the deepest $\lfloor np \rfloor$ points, as $p$ goes from 0 to 1. Scale curves however can be difficult to compute in high dimensions, one




(a) (b)

Figure 4.6: (a) dd-plot of raw prostate data using the 5 gene expression values with the largest location difference. (b) dd-plot of raw prostate data using the 2 gene expression values with the largest location difference.

exception being the curves introduced by Lopez-Pintado and Romo (2009). See the following for more ways in which depth measures can be used in analysis: Liu et al. (1999); Serfling (2006).

The projection median can replace the vector of means or medians in any problem where location needs to be estimated. As an example, one could use the projection median in the k-means clustering algorithm to assess the centre of each cluster. These methods should especially be used when the data is possibly corrupted. Recall we emphasized that IRW depth and the projection median are computable in high dimensions.

This all being said, the usual problems with the analysis of high dimensional data still remain. For example, in their current form, scale curves are not easily computable in high dimensions as they involve convex hulls. On top of this, in $d > n$ scenarios the convex hull is a lower dimensional structure. Another method for measuring scale in high dimensions and sparse situations would be very useful. In fact, it would be beneficial to better understand the application of depth-based analysis to sparse scenarios such as the ones in Example 4. It would also be interesting to compare the performance of the rank test introduced by Li and Liu (2004) against other methods, especially in high dimensional settings.

# Bibliography


Agarwal, P. K., M. De Berg, and M. Sek (1998). Constructing Levels in Arrangements and Higher Order Voronoi Diagrams *. *SIAM Journal of Computing 27*(3), 654–667. (Cited on pages 31, 32 and 33.)

Aloupis, G. (2006). Geometric Measures of Data Depth. In *Data Depth: Robust Multivariate Analysis, Computational Geometry and Applications*, Volume 72. (Cited on page 9.)

Anderson, E. (1936). The species problem in Iris. *Annals of the Missouri Botanical Garden 23*(3), 467–503. (Cited on page 41.)

Andrzejak, A. and E. Welzl (1997). k-sets and j-facets-A tour of discrete geometry. Accessed: 10-04-2016. (Cited on pages 30, 31 and 32.)

Basch, J., L. J. Guibas, and J. Hershberger (1999). Data structures for mobile data. *J. Algorithms 31*(1), 1–28. (Cited on pages 15, 17 and 24.)

Basu, R., B. Bhaswar, and T. Tanmoy (2011). The projection median of a set of points in Rˆd. *Discrete Computational Geometry 47*, 329–346. (Cited on pages 12, 14, 15, 24, 28 and 30.)

Chen, Z. and D. E. Tyler (2002). The influence function and maximum bias of Tukey's median. *Annals of Statistics 30*(6), 1737–1759. (Cited on pages 4 and 10.)

Cuevas, A. and R. Fraiman (2009). On depth measures and dual statistics. A methodology for dealing with general data. *Journal of Multivariate Analysis 100*(4), 753–766. (Cited on pages 43, 45, 46, 55 and 60.)

Donoho, D. and M. Gasko (1992). Breakdown Properties of Location Estimates Based on Halfspace Depth and Projected Outlyingness. *Statistics 20*(4), 1803–1827. (Cited on pages 10, 15 and 64.)

Durocher, S., R. Fraser, A. Leblanc, and M. Skala (2014). On Combinatorial Depth Measures. In *CCCG 2014 Conference Proceedings*. (Cited on page 15.)

Durocher, S. and D. Kirkpatrick (2009). The projection median of a set of points. In *Computational Geometry: Theory and Applications*, Volume 42, pp. 364–375. (Cited on pages 5, 7, 8, 9, 12, 14 and 15.)

Durocher, S., A. Leblanc, and M. Skala (2017). The Projection Median as a Weighted Average. *Journal of Computational Geometry 8*(1), 78–104. (Cited on pages 6, 11, 12, 15, 34, 35 and 36.)

Dyckerhoff, R. and P. Mozharovskyi (2016). Exact computation of the halfspace depth. *Computational Statistics and Data Analysis 98*, 19–30. (Cited on page 10.)

Edelsbrunner, H. (1997). *Algorithms in Combinatorial Geometry*. Berlin, Germany: Springer Verlag. (Cited on pages 17, 30, 31 and 32.)

Fisher, R. A. (1936). The use of Multiple Measurements inTaxonomic Problems. *Annals of Eugenics 7*(2), 179–188. (Cited on page 41.)

Hampel, F. R., E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel (1986, mar). *Robust Statistics*. Hoboken, NJ, USA: John Wiley & Sons, Inc. (Cited on page 4.)

He, X. and D. G. Simpson (1993, mar). Lower Bounds for Contamination Bias: Globally Minimax Versus Locally Linear Estimation. *The Annals of Statistics 21*(1), 314–337. (Cited on page 4.)

Hoeffding, W. (1963). Probability Inequalities for Sums of Bounded Random Variables. *American Statistical Association 58*(301), 13–30. (Cited on page 56.)

Huber, P. J. and E. M. Ronchetti (2009). *Finite Sample Breakdown Point*, pp. 279–287. John Wiley & Sons, Inc. (Cited on page 3.)

Li, J. and R. Y. Liu (2004). New Nonparametric Tests of Multivariate Locations and Scales Using Data Depth. *Statistical Science 19*(4), 686–696. (Cited on pages 75, 76, 78 and 79.)

Liu, R. (1990). On a Notion of Data Depth Based on Random Simplices. *The Annals of Statistics 18*(1), 405–414. (Cited on pages 9, 46 and 50.)

Liu, R., R. Serfling, and D. Souvaine (2008). *Data Depth: Robust Multivariate Analysis, Computational Geometry, and Applications*. DIMACS series in discrete mathematics and theoretical computer science. American Mathematical Soc. (Cited on pages 9, 10 and 15.)

Liu, R. Y., J. M. Parelius, and K. Singh (1999, 06). Multivariate analysis by data depth: descriptive statistics, graphics and inference, (with discussion and a rejoinder by liu and singh). *Ann. Statist. 27*(3), 783–858. (Cited on pages 5, 75, 76 and 79.)

Liu, X. H., Y. Zuo, and Q. H. Wang (2017). Finite sample breakdown point of Tukey's halfspace median. *Science China Mathematics 60*(5), 861–874. (Cited on page 65.)

Lopez-Pintado, S. and J. Romo (2009). On the concept of depth for functional data. *Journal of the American Statistical Association 104*(486), 718–734. (Cited on pages 9 and 79.)

Lopuhaa, H. and P. Rousseeuw (1991). Breakdown Points of Affine Equivariant Estimators of

Multivariate Location and Covariance Matrices. *The Annals of Statistics 19*(1), 229–248. (Cited on pages 3 and 8.)

Lorentz, G. (1986). *Bernstein Polynomials*. AMS Chelsea Publishing Series. Chelsea Publishing Company. (Cited on page 57.)

Mosler, K. and R. Hoberg (2006). Data analysis and classification with zonoid depth. In *Data Depth: Robust Multivariate Analysis, Computational Geometry and Applications*, Volume 72. (Cited on pages 9 and 78.)

Muller, M. E. (1959, April). A note on a method for generating points uniformly on n-dimensional spheres. *Commun. ACM 2*(4), 19–20. (Cited on page 34.)

Ramsay, K. A. (2017). Projection median. https://github.com/12ramsake/projectionmedian. (Cited on pages 15 and 34.)

Serfling, R. (2006). Depth Functions in Nonparametric Multivariate Inference. In S. Liu, Serfling (Ed.), *Data Depth: Robust Multivariate Analysis, Computational Geometry, and Applications*, pp. 1–16. American Mathematical Society. (Cited on pages 2, 5, 9 and 79.)

Serfling, R. (2010). Equivariance and invariance properties of multivariate quantile and related functions, and the role of standardization. *Journal of Nonparametric Statistics 22*(7), 915–936. (Cited on page 50.)

Tukey, J. W. (1974). Mathematics and the Picturing of Data*. In *Proceedings of the International Congress of Mathematicians*. (Cited on pages 9, 10, 15 and 63.)

Welsh, J. B., L. M. Sapinoso, A. I. Su, S. G. Kern, J. Wang-Rodriguez, C. A. Moskaluk, H. F. Frierson, and G. M. Hampton (2001). Analysis of gene expression identifies candidate markers and pharmacological targets in prostate cancer. http://statweb.stanford.edu/ ckirby/brad/LSI/datasets-and-programs/datasets.html. (Cited on pages 72 and 77.)

Zhang, J. and M. Pan (2016). A high-dimension two-sample test for the mean using cluster subspaces. *Computational Statistics and Data Analysis 97*, 87–97. (Cited on pages 6, 71, 72, 73, 74 and 78.)

Zuo, Y. (2004). Influence Function and Maximum Bias of Projection Depth Based Estimators. *The Annals of Statistics 32*(1), 189–218. (Cited on page 4.)

Zuo, Y. and R. Serfling (2000). General notions of statistical depth function. *Annals of Statistics 28*(2), 461–482. (Cited on pages 5, 9, 14, 42, 44, 47 and 50.)

# Index