

# Ray-Shooting Depth in $R^2$ - Algorithms and Applications

William Steiger\*

Mudassir Shabbir

## Abstract

The notion of *Ray Shooting Depth* was introduced in recent papers of Gromov, and of Fox, Gromov, Lafforgue, Naor, and Pach. It played a key role in their results. It also represents a new concept for data depth in  $R^d$  and offers interesting possibilities for applications. Here, we study some computational aspects of ray-shooting depth in dimension two via algorithms and some complexity results. In addition we advocate the use of ray-shooting depth in statistical data analysis and in other applications. We illustrate some of the desirable properties via comparisons with other notions of depth.

## 1 Introduction

Given a set  $S = \{P_1, \dots, P_n\}$  of  $n$  points in general position in  $R^2$  (the data)  $G = (V, E)$  denotes the complete geometric graph whose vertices are the points of  $S$  and edges, the segments  $\overline{P_i P_j}, i < j$ .

For a point  $Q \in R^2$ , the *ray-shooting depth* of  $Q$  is defined as

$$d_\rho(Q) = \min_{u: \|u\|=1} (r(Q, u)), \quad (1)$$

where  $r(Q, u)$  is the number of segments in  $E$  that meet  $\{Q+tu, t \geq 0\}$ , the ray through  $Q$  in direction  $u$ . A (ray-shooting) median is a point  $\mu_\rho \in R^2$  of maximal ray-shooting depth. It need not be unique nor an element of  $S$ .

Ray shooting depth is the specialization to two dimensions of a notion that was proposed in recent papers of Gromov [5] and Fox et.al. [3]. In [3] it was instrumental in establishing a new lower bound for the maximal number of  $d$ -simplices that are defined by  $n$  points in general position in  $R^d$ , and which have non-empty intersection. The notion has other important combinatorial consequences as well.

In the present paper we focus on the two dimensional version and recommend adding it to the set of concepts used to define data depth in terms of a set  $S$  of  $n$  given points in  $R^2$ . For this purpose it is necessary to understand the computational properties.

We give a simple algorithm, along with a matching lower bound to show that

**Theorem 1** *Given a set  $S$  of  $n$  points in  $R^2$  and a query point  $Q$ ,  $d_\rho(Q)$  can be computed in time  $\Theta(n \log n)$ .*

In addition we show that

**Theorem 2** *Given a set  $S$  of  $n$  points in  $R^2$ , a median  $\mu_\rho \in R^2$  and its depth  $d_\rho(\mu_\rho)$  can be found in  $O(n^4)$ .*

We don't know a lower bound better than that given by Theorem 1. Because of this gap, the following result may be useful.

**Theorem 3** *Given a set  $S$ , of  $n$  points in  $R^2$ , we can compute  $d_\rho(P_i), i = 1, \dots, n$  in  $O(n^2)$  time. In addition, in  $O(n(\log n)^2)$  time, we can find a point  $Q \in R^2$  with  $d_\rho(Q) \geq d_\rho(P_i), i = 1, \dots, n$ .*

The algorithms are described in the next section, along with the arguments for the asserted complexities. The remaining sections of the paper are offered in support of our endorsement of ray-shooting depth for data analysis and other applications.

In Section 3 we compare ray shooting depth to four known data depth measures that have some currency in application. Each of them defines the depth of a point  $Q \in R^d$  based on a given set  $S$  of  $n$  points in general position in  $R^d$ , though we will only be considering the  $d = 2$  versions. These depth notions are:

- **Tukey depth  $d_\tau$ :** Defined by John Tukey [12] and sometimes called the halfspace depth of a point  $Q \in R^d$ , it is defined as

$$d_\tau(Q) = \min_{u: \|u\|=1} (h_{min}(Q, u)), \quad (2)$$

where  $h_{min}(Q, u)$  denotes the minimum of the numbers of points of  $S$  in the two closed halfspaces determined by the hyperplane through  $Q$  with normal vector  $u$ . A Tukey median is a point  $\mu_\tau \in R^d$  of maximal Tukey depth. It need not be unique nor an element of  $S$ . It is known that  $d_\tau(\mu_\tau) \geq \lfloor n/(d+1) \rfloor$ , a simple consequence of Helly's theorem.

- **Simplicial depth  $d_\sigma$ :** First proposed by Regina Liu [7], it is defined by

$$d_\sigma(Q) = \sum_{\Delta \in \binom{S}{d+1}} I_{Q \in \Delta} \quad (3)$$

\*Department of Computer Science, Rutgers University, 110 Frelinghuysen Road, Piscataway, New Jersey 08854-8004; ({steiger,mudassir}@cs.rutgers.edu.)

where  $\binom{S}{d+1}$  denotes the set of  $d$ -simplices with vertices in  $S$  and  $I$ , the indicator function. A simplicial median is a point  $\mu_\sigma \in R^d$  of maximal simplicial depth. It need not be unique nor an element of  $S$ . It is known that  $d_\sigma(\mu_\sigma) \geq c_d \binom{n}{d+1}$ ; i.e., for every set  $S$ , there is a point in a fixed fraction of all simplices. Boros and Furedi proved that  $c_2 = 2/9$  and Bárány showed  $c_d \geq (d+1)^{-d}$ , a lower bound recently improved by Gromov [5] to  $2d/[d!(d+1)^2]$ , exponentially larger than Bárány's bound.

- **Oja Depth** It is defined by

$$d_\sigma(Q) = \sum_{\Delta \in \binom{S}{d}} \text{VOL}(Q\Delta) \tag{4}$$

the sum of the volumes of all  $d$ -simplices having  $Q$  as a vertex, along with  $d$  points of  $S$ .

———— follow these theoretical results with simple and easily portable implementations. Particularly we discuss *RSplot*, a new bivariate data visualization tool that we developed based on Ray Shooting algorithms. Software we developed are open source and freely available in R, the standard statistical development environment.

We also conducted an experimental study to compare accuracy and robustness of RS depth with already known data depth measures Tukey, Simplicial, Oja, and Spatial medians on relatively large data sets chosen randomly from eleven different distributions. We discuss details of our method and results for this simulation in the last section.

## 2 Algorithms

Let  $P = \{P_1 \dots, P_n\}$  and  $Q = \{Q_1, \dots, Q_n\}$  be sets of points in  $R^2$  of sizes  $n$  and  $m$ , respectively. we give following algorithm to compute RS Depth,  $\rho(P, q_i), \forall q_i \in Q$ . We abuse following notations for the discussion below:  $(a, b)$  to mean a line passing through, and  $[a, b]$  to mean a line segment through points

$a, b$  and  $[a, b)$  to mean a half infinite ray starting at  $a$  and passing through  $b$ . Also for a point  $p \in R^2$ , its dual line is denoted by  $p^*$ ; we do not fix any point line duality, and any definition that preserves incidence and above/below relationship will suffice.

### 2.1 Algorithm to compute RS-Median of a Pointset

Given a set  $P$  of  $n$  points in plane, we give an algorithm to compute RS-Depth of all points in plane, also locate an arbitrary point  $z$  such that  $\delta(z) \geq \delta(z'), \forall z' \in R^2$ .

We divide plane into a set of faces defined by the arrangement  $\mathcal{A}(E_P)$ , of  $\binom{n}{2}$  lines induced on points in  $P$ . We observe that all points within a face have same RS-Depth, and also we note a relation between  $\delta$  values

---

**Algorithm 1** RS Depth of  $|Q| = m$  with respect to  $|P| = n$

---

```

    build  $\mathcal{A}(P)$ , the arrangement of  $p_i^*$  for  $p_i \in P$ 
2: for all  $q_i \in Q$  do
     $L_i \leftarrow \{l_{i,j} : \text{slope of } (q_i, p_j)\}$ 
4: sort  $L_i$  by inserting  $q_i^*$  in  $\mathcal{A}(P)$ . W.L.O.G.  $l_{i,1} \leq l_{i,2} \dots \leq l_{i,n}$  be in anti-clockwise order on unit circle with  $q_i$  as center. Also  $p_{i,j} \in P$  be alias of a point with which makes slope  $l_{i,j}$  with  $q_i$ .
    end for
6: for all  $q_i \in Q$  do
     $\text{arc}_{i,j} \leftarrow$  number of points as we walk from  $p_{i,j}$  to  $-p_{i,j}$ .
8:  $\rho_{i,1} \leftarrow \sum_{k=1}^n (n - \text{arc}_{i,k} - k - 1)$ 
    for  $r = 2 \rightarrow n$  do
10:  $\rho_{i,r} \leftarrow \rho_{i,r-1} - n + 2 \times \text{arc}_{i,j-1} + 1)$ 
    end for
12:  $\rho_i \leftarrow \min_{j=1}^n \rho_{i,j}$ 
    end for
     $\rho$ 

```

---

of  $\delta$  values of points in neighboring faces. We define  $N(f_i, P)$  as set

of all neighbor faces of face  $f_i$  where two faces are neighbors to each other if they share a common edge.  $N(f_i) = N(f_i, P)$  where  $P$  is obvious from the context. We fix  $f_0$  to be the unbounded face of this arrangement which is rather a union of all unbounded faces but we will use it just one face whose description is readily available by taking the convex hull of  $P$ . Algorithm follows:

**Lemma 4** For any two points  $x, y$  in some  $f_i$ ,  $\delta(x) = \delta(y)$  always holds.

**Lemma 5** Finding neighbors takes const time.

### 2.2 Lower bound on computing RS-Depth of a point

In this section we prove  $\Omega(n \log n)$  lower bound on algorithm computing RS-Depth of a query point  $q$  for a  $n$ -pointset  $P \in R^2$  in algebraic computation tree model. We provide a linear time reduction from set-equality problem: Given two sets  $A = \{x_1, x_2, \dots, x_n\}$  and  $B = \{y_1, y_2, \dots, y_n\}$  in  $R$ , it takes  $\Omega(n \log n)$  computations to decide whether or not  $A = B$ .

**Lemma 6** Any algebraic computation tree that solves  $n$ -points RS-Depth problem has a complexity  $\Omega(n \log n)$

**Proof:** Given two sets  $A = \{x_1, x_2, \dots, x_n\}$  and  $B = \{y_1, y_2, \dots, y_n\}$  we construct  $P \in R^2$ , such that by finding out depth of origin *w.r.t*  $P$  we will be able to decide  $A = B$ ?. First, without loss of generality we assume that  $0 < x_i < \frac{\pi}{2}, \forall x_i \in A$ , if not we know that a mapping always exists. Similarly  $0 < y_i < \frac{\pi}{2}, \forall y_i \in B$ . Now

---

**Algorithm 2** RS Median of a set  $|P| = n$

---

```

    build  $\mathcal{A}(E_P)$ , the arrangement of  $(p_i, p_j) \in P \times P$ 
2: Let  $F =$  set of all faces in  $\mathcal{A}(E_P)$ .
   for all  $f_i \in F$  do
4:    $label(f_i) \leftarrow \infty$ 
   end for
6:  $count \leftarrow 0$ 
    $label(f_i) \leftarrow count$ 
8:  $L \leftarrow \{f_i\}$ 
    $S \leftarrow L$ 
10: repeat
     $R \leftarrow S$ 
12:  $S \leftarrow \emptyset$ 
     $count \leftarrow count + 1$ 
14: for all  $l_i \in L$  do
    Remove  $l_i$  from  $L$ 
16:   for all  $f_j \in N(l_i)$ , s.t.  $label(f_j) = \infty$  do
     $label(f_j) \leftarrow count$ 
18:    $S \leftarrow S \cup \{f_j\}$ 
   end for
20: end for
     $L \leftarrow S$ 
22: until  $L = \emptyset$ 
    RETURN  $R$  as set of RS-Median points.

```

---

for each  $x_i \in A$  we define  $p_i = (\frac{i}{n}, \cos(x_i))$ , and for each  $y_j \in B$ ,  $p_{n+j} = (-\frac{j}{n}, \cos(y_j + \frac{\pi}{2}))$ . We claim.

**Claim 1** If  $\delta(O) = \frac{n^2}{8}$  for set  $P = \{p_1, p_2, \dots, p_{2n}\}$ , where  $O = (0, 0)$ , then  $A = B$ .

**Proof:** Let us suppose otherwise that  $A \neq B$ , and W.L.O.G.  $x_1 \leq x_2 \dots \leq x_n$  and  $y_1 \leq y_2 \dots \leq y_n$  and let  $i$  be smallest such that  $x_i < y_i$ . By our construction it implies that  $p_i \neq p_{n+i}$ . Now consider the line  $l$  passing through  $O$  such that both

Consider the line passing through  $O$  s.t. both  $p_i, p_{n+i}$  lie to the right of it and points  $p_{i+1}, p_{i+2} \dots, p_n$  are on left. We see that  $(\frac{n}{2} - 1) \cdot (\frac{n}{2} + 1)$  line segments induced on  $P$  intersect this line, hence at least one of the two rays in  $l$  starting at  $O$  intersects at most  $\frac{n^2}{8} - 1$  line segments contradicting the assumption that  $\delta(O) = \frac{n^2}{8}$ . ■

**Claim 2** If  $\delta(O) < \frac{n^2}{8}$  for set  $P = \{p_1, p_2, \dots, p_{2n}\}$ , where  $O = (0, 0)$ , then  $A \neq B$ .

**proof**

Again assume  $A = B$ . By construction of  $P$ , we note that for any  $p_i \in P$ , there is exactly one antipodal point  $p_j \in P$ . Furthermore any line  $l$  passing through  $O$  is a halving line of  $P$ : both half-planes such defined have exactly  $\frac{n}{2}$  points (unless  $l$  passes through some  $p_i$  which we discuss later). So  $\frac{n^2}{4}$  line segments intersect each line, and as points are arranged symmetrically around

$O$  in first and third quadrants of plane, both half infinite rays of such  $l$  starting at  $O$  intersect equal number of line segments.

In case of a line passing through two antipodal points in  $P$ , line intersects  $(\frac{n}{2} - 1)^2 \cdot (n - 1)$  segments and each ray intersects at least  $\frac{n^2}{8} + \frac{n-1}{2}$  line segments. This contradicts the assumption that  $\delta(O) < \frac{n^2}{8}$  implying  $A \neq B$ . ■

Claim 1 and Claim 2 complete the reduction by showing that computing RS Depth of set of  $n$  points in plane, we can decide set equality. Lower bound follows. ■

### 3 RSplot: A Bivariate Data Visualization Tool

Given remarkable accuracy of Ray Shooting depth to rank points in plane, as observed in Simulations section below, its natural to consider it for purpose of bivariate data representation and outlier identification. We present RSplot, a variant of Bagplot, using Ray Shooting depth instead of Tukey depth. Main components of RSplot include:

- RS median point: a point of maximum RS depth.
- Median bag: convex hull of set of all median points.
- Half bag: a convex polygon that contains 50% points in  $P$ .
- Fence: a polygon that identifies outliers in data.

We implemented RSplot, in R, the standard statistical computing language so that it is readily available to use and extend on all recognised platforms. As we also provide its source in C++ plus, it can easily be ported to other environment. R package for current implementation is available on all R CRAN servers and link to main repository is <http://cran.r-project.org/web/packages/rsdepth/>. Some examples where RSplot is used to represent some real world data follow:

### 4 RSplot: A Bivariate Data Visualization Tool

Given remarkable accuracy of Ray Shooting depth to rank points in plane, as observed in Simulations section below, its natural to consider it for purpose of bivariate data representation and outlier identification.

We implemented RSplot, in R, the standard statistical computing language so that it is readily available to use and extend on all recognised platforms. As we also provide its source in C++, it can easily be ported to other environment. R package for current implementation is available on all R CRAN servers and link to main repository is <http://cran.r-project.org/web/packages/rsdepth/>. Some examples

where RSplot is used to represent some real world data follow:

#### 4.1 Motivations for RSplot

We provide three interfaces for all the functionality

#### 4.2 RSplot Interfaces

To provide all the promised functionality, RSplot exposes a number of interfaces in main RSdepth package. A brief usage, variations, requirements, examples and implementation details of each interface follows.

##### 4.2.1 rsrings: contours of rs-depths

Given a two dimensional matrix of a bivariate sample  $P$ , *rsrings* creates a plot of data, identifies a point in  $P$  as deepest point, and divides  $P$  into a set of five convex contours or rings of gradually increasing sizes. Each ring contains 20% more points than the next smaller ring, and smaller rings signify a deeper set of points with high RS depth. Interface provides a way to change to number of rings to be generated, fraction of points in each ring changes accordingly. Contours can optionally be made colorful. We observed that, while *rsrings* may give a successful plot in any case, for rings to be provide any meaningful information it is a good that number of rings be less than  $\frac{n}{50}$  and more than 3. *rsrings* gives a visual description of centrality ranking of points and perfect for a study of bivariate Spacings. Below we used it on three sets of 500 points each drawn respectively from bivariate uniform, bivariate normal and bivariate exponential distributions at random. Please observe that in all cases rings drawn provide a near optimal estimate of underlying distribution.

To construct these rings, we

- Sort points in  $P$  according to their RS depth
- Partition sorted  $P$  into blocks of appropriate sizes, of size  $\frac{n}{5}$  by default.
- Take convex hull of each part part.
- And find the median point.

While we gave an  $O(n^2)$  algorithm for constructing *rsrings*, we used a simpler  $O(n^2 \log n)$  approach. That's because we could not find any efficient implementation of line arrangement; hidden constant for line arrangement in CGAL library is too high that for all data sets of sizes less than 10000, it's more feasible to use our current implementation. We also note that relative to other similar software in statistics development environments, our implementation is quite efficient for practical purposes, for example, it took less than 3 seconds to figure out and draw 100 rings on a set of 500 points

on 2GHz Intel processor. A well known implementation that uses Tukey median to draw contours, took more than a minute on same set of points on same machine. Unfortunately we could not find any open source implementation of drawing contours of Simplicial depth.

##### 4.2.2 rstruerings: "true" contours of rsdepth

*rstruerings* also generates contours based *rsdepth*, but unlike *rsrings*, all points within a contour, including points that are not in sample, fall in same range of *rsdepth*. It gives a more fine grained ranking of points with respect to *rsdepth*. This makes more sense for some particular types of samples: for example given a sample  $P$ , such that all points in  $P$  are in convex positions, there is essentially a single contour based on *rsrings* because all points in  $P$  have same rank according to centrality. But if we use *rstruerings*, we can still get  $O(n^2)$  meaningful rings or contours. So, although *rsrings* give a more appropriate picture of the underlying distribution, it is sometimes more relevant to visual the "true" contours based on RS depth of all points in plane (not just the points from sample  $P$ ) for it has different geometric structure.

It is a strictly harder computational problem and compared to  $O(n^2)$  algorithm for constructing *rsrings*, there is, as observed above, an  $\Omega(n^4)$  on time complexity of constructing *rstruerings*.

##### 4.2.3 rsplot: a Bagplot with rsdepth

We present *rsplot*, a variant of Bagplot, using Ray Shooting depth instead of Tukey depth. Main components of RSplot include:

- RS median point: a point of maximum RS depth.
- Median bag: convex hull of set of all median points.
- Half bag: a convex polygon that contains 50% points in  $P$ .
- Fence: a polygon that identifies outliers in data from inliers.

Below we show few applications of *rsplot* on plasma readings of 60 patients. Plot on left is, an approximate one, where Half bag is constructed by the convex hull of  $\frac{n}{2}$  deepest points with respect to their *rsdepth*.

##### 4.2.4 rstinterval: a tolerance interval of rsdepth

Tolerance interval is distinct from a confidence interval, and is defined in a ref as below:

## 5 Simulations

### 5.1 Data Source and their Sizes

For our experiments we choose bivariate data points randomly from a set different distributions. This set includes a mixture of uniform, centrally concentrated, and heavy-tailed distributions to give a better idea of performance of various median estimators and data depth function. We also choose a set of contaminated distributions, as described below, to measure effect of outliers on these estimations.

Five pure distributions are as follow:

1. Normal distribution with  $\mu = 0$  and  $\sigma = 1$
2. Uniform distribution with  $\mu = 0$
3. Cauchy distribution with  $\mu = 0$
4. F distribution with  $\mu = 0$ , first degree of freedom = 10 and second degree of freedom = 100
5. t Student distribution with  $\mu = 0$ , degree of freedom = 2

where  $\mu$  represents the mean of a distribution while  $\sigma$  is the standard deviation. All pure distribution are symmetric around origin. Six contaminated distributions are

6. Normal distribution with 5% contamination from Normal with a displaced  $\mu$ .
7. Normal distribution with 10% contamination from Normal with a displaced  $\mu$ .
8. Normal distribution with 30% contamination from Normal with a displaced  $\mu$ .
9. Cauchy distribution with 5% contamination from Normal with a displaced  $\mu$ .
10. Cauchy distribution with 10% contamination from Normal with a displaced  $\mu$ .
11. Cauchy distribution with 30% contamination from Normal with a displaced  $\mu$ .

For one experiment we choose a set of 2500 points in plane from each of these distributions as our actual set of points. Then we choose a sample of 1000 points uniformly from the range of data set as our candidate median points. For each candidate point in sample we calculate

- **RS depth**
- **Tukey depth**
- **Simplicial depth**

- **Oja depth**

and take the point with highest depth as our respective approximate median.

- **Spatial approximate median** has its own iterative algorithm to converge to a suitable median.

Each experiment was replicated 500 times, and averaged out results were used to minimize errors in approximation.

### 5.2 Algorithms

Following algorithms were employed for this study.

- **RS depth** : For *rsdepth* we use simple  $O(n \log n)$  algorithm: that is to sort all points in data set radially around query point, and calculate number of segments interesting a particular ray. As given depth in particular ray, depth in neighboring rays can be found in constant time, linear time is enough to find least ray depth. If depths of two points is equal, we pick arbitrarily. C++/R implementation of algorithm was used from *RSdepth* package in R.
- **Tukey depth** : calculation of Tukey Depth is based on Fortran code from Rousseeuw and Ruts (1996) available in R as *depth* package.
- **Simplicial depth** : calculation of Simplicial or Liu depth is again based on Fortran code from Rousseeuw and Ruts (1996) available in R as *depth* package.
- **Oja depth** : is derived from a location measure considered by Oja. Implementation comes from *depth* package in R.
- **Spatial Median** : Implementation comes from R package *ICSNP* which follows algorithm of Vardi and Zhang [13]

We assume that all these median estimators try to approximate mean of random distributions that we used. Furthermore we expect a robust estimator to behave well even in case when mean has been dragged in some arbitrary direction due to a “certain amount of erroneous data”.

Functions to generate random number generation for all our distributions that we used were borrowed from R base packages. We used internal R language environment to implement rest of the algorithm and run experiment scripts. *RSdepth* package that we wrote to conduct this study is now available in main R CRAN repository along with its C plus plus Source code [11].

### 5.3 Measure of Performance

For our simulated data, we quantify accuracy, biasness and robustness of five median estimator, described above, according to two standard measure of bivariate point estimators. Root Mean Squared Error  $RMSE$ , is a precision quantifier and for our simulations, is defined as numerical function of estimated and true value as below:

$$RMSE(\hat{\theta}, \theta) = \sqrt{MSE(\hat{\theta}, \theta)} = \sqrt{\frac{1}{500} \sum_{i=1}^{500} E((\hat{\theta}_i - \theta)^2)} \tag{5}$$

where  $\hat{\theta}_i$  takes the value of median estimators, and  $\theta$  is actual mean of the distribution to be tested.

Our second estimator for the accuracy is squared bias of estimators, which is a relation between  $MSE$  and variance as below:

$$(Bias(\hat{\theta}, \theta))^2 = MSE(\hat{\theta}, \theta) - Var(\hat{\theta})$$

Both of these measures give an account of the accuracy of a particular estimator: smaller value of ?? and ?? is an indicator of better estimator. We will also use as test of robustness by only applying them to heavy-tailed and contaminated distributions.

### 5.4 Results

Results of our experiments follow in ten tables at the end of this section, one for each distribution used as source of data set; two rows in each table represent tested median approximator’s value for  $RMSE$  and  $Bias^2$ . Our interpretation of these results follow:

Ray Shooting or RS depth behaved well against all pure distributions beating Tukey and Simplicial estimators in both error value and  $bias$  in many cases and remaining close by in rest of them.  $Bias$  value of RS-Depth in case of 10% (and 30% as well) contamination of Normal by displaced mean Normal was higher than other estimators. We observe that in rest of the cases  $bias$  and error values were not that bad. This performance of Ray Shooting depth on contaminated data sets shows that empirically RS Depth is more robust than Tukey Depth; difference is clearly discernible in Table 5.4.

Tukey median is extremely low error for all pure distributions except perhaps Uniform distribution, for which, all estimators that are based on random process for selecting a median, are expected to show some error. For contaminated distributions, performance of Tukey median was bit worse, specially in case when Cauchy distribution is contaminated with 30% Normal, estimates by Tukey median are observed to be significantly worse than all other candidate medians.

Simplicial median started poorly with extremely high  $bias^2$  (170+) for Normal distribution, didn’t do much well for F distribution either. Performance on contaminated data sets was, comparatively better being a clear winner in case of Cauchy distribution contaminated with 30% Normal, but error and bias in case Normal distribution contaminated with 30% Normal, is almost unacceptable.

Oja Depth behave very well in most of the cases on both bias value and error term. At a couple of occasion in contaminated distributions, Oja Depth was beaten by a nicer valued Liu/Simplicial Depth.

Spatial median is the only in our experiments for which we had an efficient deterministic implementation to approximate center. And expectedly, it outperforms rest of the candidates with respect to accuracy and robustness in all distributions, as is clear from very small values of error and bias.

Need for efficient and easy to implement median algorithms for such data depth measures can’t be overemphasized as current implementation of all these medians fall short against any sufficiently large data. Yet using RS-depth measure on a randomly chosen sample set to approximate is shown to be promising estimator for bivariate data sets.

	rs	tukey	liu	oja	spatial
$RMSE$	0.01212	0.01213	12.542	0.01206	0.00100
$Bias^2$	0.00031	0.00030	170.80	0.00030	0.00001

#### Normal Distribution

	rs	tukey	liu	oja	spatial
$RMSE$	0.51137	0.51038	6.2419	0.51109	0.4999
$Bias^2$	0.591	0.28396	51.173	0.28071	0.25038

#### Uniform Distribution

	rs	tukey	liu	oja	spatial
$RMSE$	0.03497	0.03547	0.68259	0.03832	0.00257
$Bias^2$	0.00287	0.00221	0.63617	0.00426	0.00001

#### Cauchy Distribution

### References

- [1] I. Bárány). “A Generalization of Carathéodory’s Theorem”. *Discrete Math.* 40(2-3) 141-152 (1982).
- [2] E. Boros and Z. Fredi. “The Number of Triangles Covering the Center of an n-Set” *Geom. Dedicat* 17 69-77 (1984)
- [3] J. Fox, M. Gromov, V. Lafforgue, A. Naor, J. Pach. “Overlap Properties of Geometric Expanders”. *Journal fur die reine und angewandte Mathematik (Crelle’s*

	rs	tukey	liu	oja	spatial
<i>RMSE</i>	0.03517	0.03474	17.264	0.03625	0.00336
<i>Bias</i> <sup>2</sup>	0.002	0.0026333	398.846	0.00268	0.00001

F Distribution

	rs	tukey	liu	oja	spatial
<i>RMSE</i>	0.03314	0.03389	0.66620	0.03307	0.00155
<i>Bias</i> <sup>2</sup>	0.00228	0.00193	0.67679	0.00220	0.00001

t Student Distribution

	rs	tukey	liu	oja	spatial
<i>RMSE</i>	0.04925	0.04919	47.217	0.04909	0.00793
<i>Bias</i> <sup>2</sup>	0.00480	0.00555	3569.0	0.00465	0.00008

Normal Distribution contaminated with 5% Normal

*Journal*). A preliminary version appeared in *SODA 2011*.

- [4] J. Gill, W. Steiger, and A. Wigderson. Geometric Medians, *Discrete Math.* 108, 37-51. (1992),
- [5] M. Gromov. “Singularities, Expanders and Topology of Maps, part 2: From Combinatorics to Topology via Algebraic Isoperimetry” *Geometric and Functional Analysis* 20,2 416-526 (2010).
- [6] S. Jadhav and A. Mukhopadhyay. Computing a Centerpoint of a Finite Planar Set of Points in Linear Time. *Discrete and Comp. Geom.* 12, 291-312. (1994).
- [7] R. Liu. “On a Notion of Data Depth Based on Random Simplices”. *Ann. Statistics* 18,405-414 (1990).
- [8] H. Oja. “Descriptive Statistics for Multivariate Distributions”. *Statistics and Probability Letters*, 1 327-332 (1983).
- [9] P. Rousseeuw and M. Hubert. Depth in an Arrangement of Hyperplanes. *Discrete and Comp. Geom.* (1999)
- [10] P. Rousseeuw and M. Hubert. Regression Depth. *J. Amer. Statist. Assoc.* 94, 388-402 (1999).
- [11] .
- [12] John Tukey. “Mathematics and the Picturing of Data”. *Proceedings of the 1974 International Conf. of Mathematicians, Vol. 2* 523-531 (1975).
- [13] Y. Vardi and C.H. Zhang, “The Multivariate  $L_1$ -median and Associated Data Depth” *Proceedings of the National Academy of Sciences of the US*, 97(4) 1423-1426 (2000).

	rs	tukey	liu	oja	spatial
<i>RMSE</i>	1.30199	1.32381	0.68704	1.3495	1.2893
<i>Bias</i> <sup>2</sup>	6.6741	1.7770	0.68264	1.90427	1.6708

Normal Distribution contaminated with 10% Normal

	rs	tukey	liu	oja	spatial
<i>RMSE</i>	1.2144	0.8427	42.408	0.98836	0.51845
<i>Bias</i> <sup>2</sup>	2.6980	0.74255	3133.43	1.01717	0.26941

Normal Distribution contaminated with 30% Normal

	rs	tukey	liu	oja	spatial
<i>RMSE</i>	0.06544	0.06153	0.67546	0.06178	0.02044
<i>Bias</i> <sup>2</sup>	0.00959	0.00699	0.67158	0.00815	0.00052

Cauchy Distribution contaminated with 5% Normal

	rs	tukey	liu	oja	spatial
<i>RMSE</i>	0.13426	0.13440	0.67908	0.12152	0.08139
<i>Bias</i> <sup>2</sup>	0.03491	0.02599	0.62325	0.02810	0.00695

Cauchy Distribution contaminated with 10% Normal

	rs	tukey	liu	oja	spatial
<i>RMSE</i>	2.93678	3.6032	0.68649	1.5975	1.6688
<i>Bias</i> <sup>2</sup>	3.4831	13.0535	0.6657	2.7039	2.8049

Cauchy Distribution contaminated with 30% Normal