1 Introduction

Let $a_1, \ldots, a_m$ be $m$ distinct points in $\mathbb{R}^n$ where each point $a_i$ is associated with a positive weight $w_i$. The Fermat Weber Facility Location problem is to find the point in $\mathbb{R}^n$ that will minimize the sum of the (weighted) Euclidean Distances from the $m$ given points, also called the Geometric Median. The Geometric median is statistically more robust to outlier dominance as opposed to other estimators like the centroid. However, unlike the centroid it is difficult to compute as it has been proved that no formula using only arithmetic operation and $k$th roots can exist in general for the geometric median. However, approximations to the geometric median can be calculated using iterative schemes which exploit the convexity of the distance function to ensure that step wise decrease of the distance summation does not get trapped in a local optimum.

2 The Algorithm

A popular standard (and very old-1937) algorithm is the Weiszfeld Algorithm which is a form of iteratively reweighted squares. The Algorithm defines a set of weights inversely proportional to the distances from the current estimate to the samples and creates a new estimate that is the weighted average of the samples according to those weights. The scheme can be extended to manifolds but has no guarantees of run time. Run time guarantees in Euclidean Space can be achieved through Maheshwari Bose scheme, but it is not satisfactory as it does not extend to manifolds, and requires space and time exponential in the dimensions.

Formally, the algorithm can be defined as follows:
\[ y_{i+1} = \left( \sum_{j=1}^{m} \frac{x_j}{||x_j - y_i||} \right) / \left( \sum_{j=1}^{m} \frac{1}{||x_j - y_i||} \right) \]

If \( y \) is distinct from all the given points, \( x_j \), then \( y \) is the geometric median if and only if it satisfies:

\[ 0 = \sum_{j=1}^{m} \frac{x_j - y}{||x_j - y||} \]

which is closely related to the Weiszfeld Algorithm:

\[ y = \left( \sum_{j=1}^{m} \frac{x_j}{||x_j - y||} \right) / \left( \sum_{j=1}^{m} \frac{1}{||x_j - y||} \right) \]

As we can see, the derivatives become undefined at the vertices and hence the algorithm is stuck if the optimum coincides with a vertex (or even the iterates as seen later by Kuhn and analysed here).

### 3 Goal

The main goal of this project was thus to analyze the convergence of the Weiszfeld Algorithm in a plane, because it is still the most widely applied and is very relevant. Examples show that run time of the Weiszfeld Algorithm can be arbitrarily bad. Also, the sequence of points generated by the algorithm converges to the optimal solution provided that no iterate coincides with one of the fixed points. In such an eventuality, the iteration functions are undefined and the algorithm will terminate prematurely. There are some well known analyses which give us an insight into the probability of such an event occurring and ensuring optimality through good initial point set.

### 4 Motivating Example

One motivating example that demonstrates the run time issues of the Weiszfeld Algorithm is the following: 3 points at (-1,0), (0,1) and (0,-1), and 3 points at
(100,0), i.e a cluster near the origin and one far off. The optimal solution is to place the 1-median at (100,0), but the iterative scheme takes a long to get to it. Changing the starting point for the iteration too does not help much as can be seen from the results:

With one of the points as starting point (origin):

5 points [one less at (100,0)]:
19th iteration (0.577350,0.000000)

6 points:
1000 iterations (23.536451,0.000000)
10,000 iterations (61.220390,0.000000)
100,000 iterations (98.663605,0.000000)
200,000 iterations (99.251526,0.000000): converged

With centroid as starting point:

5 points:
39th iteration (0.577350,0.000000)

6 points:
115494 iterations (99.251526,0.000000): converged.

The high multiplicity at (100,0) causes the plunge. Also, as we can see, the algorithm spends most of its time in the near vicinity of the optimum.

5 Properties and Proofs

Formalizing the problem for proofs that follow:

Let there be \( m \) given points \( A_i = (a_{i1} \ldots a_{in}) \) called vertices and \( m \) positive numbers, \( w_i \) called weights. For \( P = (x_1 \ldots x_n) \), let

\[
d_i(P) = \sqrt{\sum_j (x_j - a_{ij})^2}
\]

be the Euclidean distances from \( P \) to \( A_i \), for \( i = 1 \ldots m \). The problem seeks to find a point that minimizes \( f(p) = \sum_i w_id_i(P) \)
If the vertices are not collinear, then $f$ is positive and strictly convex in $E^n$. Hence the minimum is achieved at a unique point M. In case of vertices being collinear, the geometric median coincides with the 1-median. It will always exist but may or may not be unique. Defining the negative gradient of $f$ is essential for further analysis. Thus, let

$$R(P) = \sum_i \frac{w_i}{d_i(P)} (A_i - P) \text{ if } P \neq A_i \text{ for all } i.$$  

Obviously $R$ is not defined at any vertex $A_i$. However, by physical analogy, we set

$$R_k = \sum_{i \neq k} \frac{w_i}{d_i(A_k)} (A_i - A_k) \text{ for } k = 1 \ldots m,$$

and extend the definition of $R$ by setting $R(A_k) = \max(|R_k| - w_k, 0)(R_k/|R_k|)$ for $k = 1 \ldots m$.

Kuhn’s Paper (1973) was one of the earliest to analyse the convergence properties of the algorithm. He gave us the following valuable insights:

1. A point $P = M$ (required median) if and only if $R(P) = 0$.

   **Proof:** If $P$ is not a vertex then convexity and differentiability of $f$ implies that the first order conditions $R(P) = 0$ are both necessary and sufficient for a minimum. If $P = A_k$, then considering a change from $A_k$ to $(A_k + tZ)$ for $|Z| = 1$. This yields:

   $$\frac{d}{dt} f(A_k + tZ) = w_k - R_k.Z \text{ for } t = 0$$

   and hence the direction of greatest decrease of $f$ from $A_k$ is $Z = R_k/|R_k|$. Clearly, $A_k$ will be the local minimum if and only if

   $$w_k - R_k^2/|R_k| \geq 0$$

   which is the same as $R(A_k) = 0$. The convexity of $f$ implies that $R(A_k) = 0$ is both necessary and sufficient for $A_k$ to be a global minimum.

2. The point $M$ lies in the convex hull of vertices $A_i$

   **Proof:** If $M$ is not a vertex, then it is trivially in the convex hull. Otherwise, the
condition $R(M)=0$ yields the consequence

$$M = \sum_i \frac{w_i}{d_i(M)} A_i / \sum_i \frac{w_i}{d_i(M)}$$

Thus $M$ is a weighted sum of the vertices with positive weights that sum to one.

The equation used in the proof above suggests quite naturally a method of successive approximation. For $P \neq A_i, i=1 \ldots m$, a mapping is defined as follows:

$$T:P \rightarrow T(P) = \sum_i \frac{w_i}{\pi_i(P)} A_i / \sum_i \frac{w_i}{\pi_i(P)}$$

For the sake of continuity, we set $T(A_i) = A_i$ for $i = 1 \ldots m$. We then get a corollary to the above statement:

3. If $P=M$, then $T(P)=P$. If $P$ is not a vertex and $T(P)=P$, then $P=M$.

4. Convergence Properties

The algorithm uses a long step gradient method that takes the direction of maximum decrease. Since $R(P)$ is the gradient of $-f$ whenever it exists, direct calculation yields

$$T(P) = P + h(P)R(P)$$

where $h(P) = \pi_i d_i(P) / \sum_k (w_k \pi_i d_i(P))$

for all points $P$.

(a) Algorithm follows long step descent gradient method. It follows direction of greatest decrease with precalculated step length $h(P)R(P)$. Such algorithms tend to overshoot. But this does not.

Formally, if $T(P) \neq P$, where $T(P)$ is the function value evaluated at $P$, $f(T(P)) < f(P)$.

**Proof:** Since
\[ T(P) \neq P, \text{ P is not a vertex and} \]
\[ T(P) = \sum_i \frac{w_i}{d_i(P)} A_i \sum_i \frac{w_i}{d_i(P)} \]

This says that \( T(P) \) is the center of gravity of weights \( w_i/d_i(P) \) placed at the vertices \( A_i \). By Calculus \( T(P) \) is the unique minimum of the strictly convex function.

\[ g(Q) = \sum_i \frac{w_i}{d_i(P)} d_i^2(Q) \]

Since \( P \neq T(P); \)

\[ g(T(P)) = \sum_i \frac{w_i}{d_i(P)} d_i^2(T(P)) < g(P) = \sum_i \frac{w_i}{d_i(P)} d_i^2(P) = f(P) \]

On the other hand,

\[ g(T(P)) = \sum_i \frac{w_i}{d_i(P)} [d_i(P) + (d_i(T(P)) - d_i(P))]^2 \]

\[ f(P) + 2(f(T(P)) - f(P)) \sum_i \frac{w_i}{d_i(P)} [d_i(T(P)) - d_i(P)]^2 \]

Combining these results:

\[ 2f(T(P)) + \sum_i \frac{w_i}{d_i(P)} [d_i(T(P)) - d_i(P)]^2 < 2f(P) \]

and the assertion \( f(T(P)) < f(P) \) is proved.

(b) Sequence does not remain in the neighborhood of non optimal vertices. \( \delta \) neighborhood of each non optimal vertex present so that if approx. sequence enters , it is eventually pushed out.

**Proof:**

\[ T(P) - A_k = P + h(P)R(P) - A_k \]
\[ = h(P) \sum_{i \neq q} \frac{w_i}{d_i(P)} (A_i - P) + (\frac{h(P)w_k}{d_k(P)} - 1)(A_k - P) \]

Since \( A_k \neq M \), we have

\[ |\sum_{i \neq k} \frac{w_i}{d_i(P)} (A_i - A_k)| > w_k \]

Hence there exists a \( \delta' \) and \( \epsilon > 0 \), such that

\[ |\sum_{i \neq k} \frac{w_i}{d_i(P)} (A_i - A_k)| \geq (1 + 2\epsilon)w_k \text{ for } d_k(P) \leq \delta' \]

By definition of \( h \), we have

\[ \lim_{P \to A_k} \frac{h(P)w_k}{d_k(P)} = 1 \]

Hence there exists a \( \delta'' > 0 \) such that

\[ \left| \frac{h(P)w_k}{d_k(P)} - 1 \right| < \frac{\epsilon}{2(1 + \epsilon)} \text{ for } 0 < d_k(P) \leq \delta' \]

Set \( \delta = \min(\delta, \delta') \). For \( 0 < d_k(P) \leq \delta \), we have

\[ d_k(T(P)) > h(P)(1 + 2\epsilon)w_k - \frac{\epsilon}{2(1 + \epsilon)}d_k(P) \]
\[ > (1 - \frac{\epsilon}{2(1 + \epsilon)})(1 + 2\epsilon)d_k(P) - \frac{\epsilon}{2(1 + \epsilon)}d_k(P) \]
\[ = (1 + \epsilon)d_k(P) \]

Since \( d_k(P) > 0, (1 + \epsilon)^t d_k(P) > \delta \) for some positive integer \( t \) and hence \( d_k(T^s(P)) > \delta \) for some positive integer \( s \) with \( d_k(T^{s-1}(P)) \leq \delta \)

(c) Behavior of function \( T \) in the \( \delta \) neighborhood of all vertices, optimal or not is given by:

\[ \lim_{P \to A_k} \frac{d_k(T(P))}{d_k(P)} = \frac{\|R_k\|}{w_k} \text{ for } k = 1 \ldots m \]

Proof: For \( P \) not a vertex,
\[ T(P) = \sum \frac{w_i}{d_i(P)} A_i / \sum w_i d_i(P) \]

\[ \sum_i w_i d_i(P) A_i / \sum w_i d_i(P) \]

Hence,

\[ T(P) - A_k = \sum_{i \neq k} \frac{w_i}{d_i(P)} (A_i - A_k) / \sum w_i d_i(P) A_i \]

\[ \frac{1}{d_k(P)} (T(P) - A_k) = \sum_{i \neq k} \frac{w_i}{d_i(P)} (A_i - A_k) / w_k (1 + \frac{d_k(P)}{w_k} \sum_{i \neq k} \frac{w_i}{d_i(P)}) \]

Taking the limits of the lengths on both sides,

\[ \lim_{r \to \infty} \frac{d_k(T(P))}{d_k(P)} = \frac{|R_k|}{w_k} \]

(d) Given any starting point \( P_0 \), \( P_r = T^r(P_0) \) for \( r = 1,2,\ldots n \). If no \( P_r \) is a vertex \( \lim_{r \to \infty} P_r = M \)

**Proof:** Details in the paper.

(e) Even if \( P_0 \) is chosen distinct from all vertices (non collinearity being the constraint), some \( P_r = T^r(P_0) \) maybe a vertex. For all but a denumerable number of \( P_0 \), \( P_r = T^r(P_0) \) converges to \( M \).

**Proof:** The convergence theorem stated above and proved in the paper establishes that if no \( P_r \) is a vertex, \( P_r \) converges to \( M \). We must solve the algebraic system of equations to characterise the bad starting points. Thus, we obtain a finite number of \( P_0 \) such that \( T(P_0) = A_i \). Hence for a fixed positive \( r \),

\[ (P_0 : T^r(P_0) = A_i \text{ for some } i=1\ldots m) \]

is finite. Finally,

\[ (P_0 : T^r(P_0) = A_i \text{ for some } i \text{ and some } r \text{ is denumerable.} \]

Much later in 1989, Tamir however proved that non collinearity is not sufficient for ensuring convergence. Kuhn’s above proof was based on the fact that
the algebraic system $T(x)=a^i$ has a finite number of solutions if all the given vertices are non-collinear. Tamir showed, using counterexamples, that the system $T(x)=a^i$ can have a continuum set of solutions even when point $a_1 \ldots a_m$ are not collinear. Weiszfeld Algorithm fails to converge when it starts at any point of the above continuum set. Both sides of the claim was proved as follows:

1. Considering an unweighted problem in $\mathbb{R}^3$ defined by four points $a_1=(1,0,0), a_2=(0,1,0), a_3=(-1,-1,0)$ and $a_4=(0,0,0)$. It can be verified that $T(x)=a_4$ for any $x=(-1/6, -1/6, x_3)$ in $\mathbb{R}^3$. However, the point $a_4$ is the optimal point. This contradicts the claim of Kuhn that if $a^i$ is the optimal solution, then $T(x)=a^i$ has a finite number of solutions.

2. This example was used to demonstrate that the system $T(x)=a^i$ can have a continuum set of solutions even if $a_i$ is not optimal. Considering the problem in $\mathbb{R}^3$ defined by the following points: $a_1=(1,0,0), a_2=(-1,0,0), a_3=(0,0,0), a_4=(0,2,0)$ and $a_5=(0,-2,0)$. Let $w_1=w_2=w_3=w_5=1$ and $w_4=3$. Considering point $a_3$. $a_3$ is not optimal since $f(a_3)=10 > f(0,1,0)=2\sqrt{2}+1+3+3=7+2(\sqrt{2})$. Considering points $x=(0,x_2, x_3)$, $T(x)=a_3$ is equivalent to the system:

$$\frac{3a_4}{||x-a^4||} + \frac{a_5}{||x-a^5||} = 0$$

Therefore, $6\sqrt{(x_2+2)^2+x_3^2} = 2\sqrt{(x_2-2)^2+x_3^2}$

Thus all points $(0,x_2, x_3)$, on the circle $(x_2+\frac{5}{2})^2 + x_3^2 = \frac{9}{4}$ satisfy $T(x)=a_3$.

In view of these examples they conjectured that:

If Convex hull of points $a_1 \ldots a_m$ is of full dimension, then algebraic system $T(x)=a^i$ has finite number of solutions for $i=1 \ldots m$.

If this could be proved to be true, convergence would be guaranteed for all but denumerable set of points if initiated at the affine set containing points $a_1 \ldots a_m$.

This happens to be true as proved by the Brimberg paper. It drew a(weak) conclusion, that if you have $n$ points on a simplex, then convergence is guaranteed (for all but a set of measure zero). But if you have more points, then it’s not.

Katz paper inherited many of the definitions of the Kuhn’s paper and stated new results for local rapidity of convergence. Following are the interesting results
1. Convergence is linear if optimum not one of the m vertices.

2. If optimum coincides with an original point, convergence can be linear, super-linear or sub-linear depending on the parameter $|\frac{R_k}{w_k}|$ which was proved to define behavior around a vertex (in the $\delta$ neighborhood) by the Kuhn’s paper.

   (a) if $|\frac{R_k}{w_k}| < 1$ convergence is linear with asymptotic convergence factor equal to the same.

   (b) $|\frac{R_k}{w_k}| = 0$, convergence will be quadratic.

   (c) $|\frac{R_k}{w_k}| = w_k$, convergence will be sub-linear.

So, if we know that the optimal solution does not coincide with the vertices, we can say something about the convergence rate. However, the open question here is to devise a method to determine if any of the vertices qualify as the optimum. Even though we can’t say much about the running time of the algorithm, we can actually bound the progress made by the algorithm in each step so as to be able to verify whether a given answer of the algorithm should be accepted for being within our desired range of tolerance, or not.

Let $\Omega$ be the convex hull generated by the fixed points $a_j, j = 1 \ldots n$ and for $x \in E_2$, let $\sigma' = \max_{y \in \sigma} d(x, y)$

Since the maximum of a convex function defined on a compact set occurs at an extreme point, it follows that:

$\sigma' = \max_{j=1 \ldots n} d(x, a_j)$. Let $x_k$ be the solution given by the Weiszfeld Algorithm at the kth iteration. As we already know that the optimum solution $x^*$ in $\omega$ and the Weiszfeld procedure generates a sequence $x_{k+1 \rightarrow \infty}$ such that $x_k \rightarrow x^*$. Let $y \in \sigma$ such that $y \neq x_k$. Then $y = x_k + \sigma r$, where $\sigma > 0$ and $r$ is the directional vector of unit length from $x_k$ to $y$.

Since $W(x)$ is convex,

\[
W(y) \geq W(x_k) + \nabla W(x_k)'(y - x_k) \\
= W(x_k) + \sigma \nabla W(x_k)'r \\
\geq W(x_k) + \sigma \nabla W(x_k)'\left(\frac{-\nabla W(x_k)}{||\nabla W(x_k)||}\right) \\
W(x_k) - \sigma ||\nabla W(x_k)||
\]
\[
\geq W(x_k) - \sigma'(x_k)||\nabla W(x_k)||.
\]

The second inequality is due to the fact that the direction of the negative gradient minimises the direction of descent. Since the above inequality holds for all \( y \in \omega \), we have,

\[
W(x_k) - W(x*) \leq \sigma'(x_k)||\nabla W(x_k)||
\]

Using this information, the user can determine how close the solution is to the optimum. In order to have a rigorous stopping criteria, a bound to the relative error

\[
\frac{W(x_k) - W(x*)}{W(x*)}
\]

is needed where \( W(x*) \) is the optimum solution that is unknown. But since

\[
\frac{\sigma'(x_k)||\nabla W(x_k)||}{W(x_k) - \sigma'(x_k)||\nabla W(x_k)||} \geq \frac{W(x_k) - W(x*)}{W(x*)},
\]

we can adopt the following procedure. At iteration \( k \), let \( x_k \) be the current solution point given by the Algorithm with current objective function value \( W(x_k) \). We compute \( \sigma'(x_k) \) and \( ||\nabla W(x_k)|| \). Then, if

\[
\frac{\sigma'(x_k)||\nabla W(x_k)||}{W(x_k) - \sigma'(x_k)||\nabla W(x_k)||} \times 100 \leq \alpha \text{ percent},
\]

then we stop and accept the solution where \( \alpha \) is a prespecified tolerance.

6 Approximation Scheme: eg: Vardi

As we can see from the original definition of the algorithm, the algorithm gets stuck; not only if one of the vertices qualify as the optimum (since the derivatives become undefined as initially pointed out by Weiszfeld), but also if one of the iterates coincide with the vertices (as corrected by Kuhn). Unless the vertices are in full dimension, it is not possible to characterise the bad starting points and hence chances of getting stuck with an arbitrarily chosen initial point are high. The Vardi Algorithm, and in general, other approximation algorithms modify the Weiszfeld Algorithm so that it behaves in exactly the same manner when the iterates do not collide with a vertex, but modifies the algorithm to force it out of the vertex neighborhood in case of such an eventuality.
7 Flowchart

Given the input vertices, when we are trying to find the geometric median using the Weiszfeld Algorithm, we need to follow the attached flowchart to avoid arbitrarily bad run times or to be able to predict the time that would be required for convergence. Of course, there is an unanswered question block which can probably be analysed using a grid structure (work is still on).