Hyperdimensional Ellipsoid Sampling

Naveen Durvasula

February 2017

1 Introduction

While the mathematical convention of "we" is used in this document, all work was done only by the student researcher.

We have two different pre-processing methods, random sampling s times, or sampling at even units across each dimension. By representing the space of all donor-patient pairs defined by n features as an n dimensional figure, we may analyze how effective these methods are.

2 Definitions

1. Let n denote the number of features that define a donor-patient pair
2. Let p denote the point in n-space representing a donor-patient pair that is requesting information
3. Let S denote the set of all pre-processed points
4. Let q denote the point in n-space $\in S$ closest to p
5. Let $D_1, D_2, D_3, \ldots, D_n$ denote the domains of each of the dimensions
6. Let ϵ denote the distance threshold that we wish to have on the minimum distance between p and q
7. Let F denote the n dimensional prism with dimensions D_i, that represents all of n-space
8. Let E denote the n dimensional figure with center p, that represents the locus of points that are $\leq \epsilon$ away from p
9. Let α denote $\mathbf{Pr}[\text{dist}(p, q) \leq \epsilon]$, this is essentially how confident we are

We wish to minimize $|S|$ while ensuring that $\mathbf{Pr}[\text{dist}(p, q) \leq \epsilon] = \alpha$

3 Random Sampling

Recall that F represents all of n-space. From this, for each sample $s \in S$, we have

$$\mathbf{Pr}[\text{dist}(p, s) > \epsilon] = 1 - V_E/V_F$$

Thus

$$\mathbf{Pr}[\forall s \in S, \text{dist}(p, s) > \epsilon] = (1 - V_E/V_F)^s$$

and as q is the nearest point to p,

$$\mathbf{Pr}[\text{dist}(p, q) > \epsilon] = (1 - V_E/V_F)^s$$

3.1 Calculating volumes

In order to compute this probability, we must compute the volumes of E and F. As F is simply an n dimensional rectangular prism, its volume

$$V_F = \prod_{i=1}^{n} D_i$$

The volume of E is given by

$$V_E = \frac{\pi^{\frac{n}{2}} \epsilon^n}{\Gamma(\frac{n}{2}+1)}$$

3.2 Computing the probability

From the above, we have

$$\mathbf{Pr}[\text{dist}(p,q) \geq \epsilon] = \left(1 - \frac{\pi^{\frac{n}{2}} \epsilon^n}{\Gamma(\frac{n}{2}+1) \prod_{i=1}^{n} D_i}\right)^{|S|} = 1 - \alpha \tag{1}$$

Solving for $|S|$ we get

$$|S| = \frac{\ln(1-\alpha)}{\ln\left(1 - \frac{\pi^{\frac{n}{2}} \epsilon^n}{\Gamma(\frac{n}{2}+1) \prod_{i=1}^{n} D_i}\right)} \tag{2}$$

Let $|S|$ for the randomized sampling method be R_1

4 Evenly Spaced Samples

In the above pre-processing method, we randomly sampled from all of n-space. The following is another pre-processing sampling model, where for each dimension i of n, we partition it into t segments, each of which has length $\frac{D_i}{t}$. The total number of samples $|S|$ is then t^n.

4.1 Calculating volumes

Each unit cell U_k with dimensions $D_1, D_2, \ldots, D_n$ is bounded by 2^n vertices. The volume of U_k is then given by

$$V_U = \prod_{i=1}^{n} \frac{D_i}{t} = \frac{\prod_{i=1}^{n} D_i}{t^n} = \frac{\prod_{i=1}^{n} D_i}{|S|}$$

We know that p lies in some unit cell U_p. For any sphere X with n dimensions, there exists 2^n regions divided by the n axes. For each of the 2^n vertices $v_l \in U_p$, we construct an n dimensional sphere E_l with radius ϵ, representing the locus of points such that for any point δ in n-space, $\text{dist}(\delta, v_l) \leq \epsilon$. However, only $\frac{1}{2^n}$ of E_l lies in U_p. Therefore, the volume of the intersection between all 2^n of these spheres and $U_p = V_E$.

4.2 Computing the probability

From the above, we have

$$\begin{aligned} \mathbf{Pr}[\text{dist}(p,q) \leq \epsilon] = \alpha &\leq \frac{V_E}{V_U} \\ &= \frac{\frac{\pi^{\frac{n}{2}} \epsilon^n}{\Gamma(\frac{n}{2}+1)}}{\frac{\prod_{i=1}^{n} D_i}{|S|}} \\ &= \frac{\pi^{\frac{n}{2}} \epsilon^n |S|}{\Gamma(\frac{n}{2}+1) \prod_{i=1}^{n} D_i} \end{aligned} \tag{3}$$

Solving for $|S|$ we have

$$|S| \geq \frac{\alpha\Gamma(\frac{n}{2}+1)\prod_{i=1}^{n} D_i}{\pi^{\frac{n}{2}}\epsilon^n} \tag{4}$$

This is a lower bound on $|S|$ and not an exact value, because if $\epsilon > \frac{D_k}{2t}$ for some k, then the regions of the spheres would intersect, thus causing us to over-count the volume of occupied space in U_k. However, we still get a conclusive comparison even with the lower bound. Let $|S|$ for the evenly spaced sampling method be R_2

5 Comparing the two methods

Recall that in the random sampling method,

$$R_1 = \frac{\ln(1-\alpha)}{\ln\left(1 - \frac{\pi^{\frac{n}{2}}\epsilon^n}{\Gamma(\frac{n}{2}+1)\prod_{i=1}^{n} D_i}\right)}$$

Let

$$f(\epsilon) = R_1$$

In the evenly spaced sampling method,

$$R_2 \geq \frac{\alpha\Gamma(\frac{n}{2}+1)\prod_{i=1}^{n} D_i}{\pi^{\frac{n}{2}}\epsilon^n}$$

Let

$$g(\epsilon) \leq R_2$$

We aim to determine which method has $|S|$ grow more slowly as $\epsilon \to 0$
Let

$$\beta = \frac{V_E}{V_U}$$

$$f(\epsilon) = \frac{\ln(1-\alpha)}{\ln(1-\beta)}$$

$$g(\epsilon) = \frac{\alpha}{\beta}$$

If we can show that $f(\epsilon) < g(\epsilon)$, then we know that the random sampling method is more efficient, as $g(\epsilon)$ gives a lower bound for R_2 in the even sampling method.

$$\begin{aligned} f(\epsilon) &< g(\epsilon) \\ \frac{\ln(1-\alpha)}{\ln(1-\beta)} &< \frac{\alpha}{\beta} \\ \beta\ln(1-\alpha) &< \alpha\ln(1-\beta) \\ e^{\beta\ln(1-\alpha)} &< e^{\alpha\ln(1-\beta)} \\ (1-\alpha)^{\beta} &< (1-\beta)^{\alpha} \\ \beta &< \alpha \end{aligned} \tag{5}$$

To prove that $f(\epsilon) < g(\epsilon)$, we must show that $\beta < \alpha$. We know that $|S| > 1$, as we must pre-process more than 1 point for pre-processing to be of any use.

Thus we have

$$\begin{aligned} |S| &> 1 \\ f(\epsilon) &= |S| \\ \frac{\ln(1-\alpha)}{\ln(1-\beta)} &> 1 \\ \beta &< \alpha \\ \therefore f(\epsilon) &< g(\epsilon) \end{aligned} \tag{6}$$

From this, we know that the random sampling method is more efficient than the even spaced sampling method.

6 Using a weighted euclidean distance

When actually computing $\text{dist}(p, q)$, it is important to note that the distance between p and q will be computed using a weighted Euclidean distance. This allows for the dominance of more important factors in computing the distance. The weights of each of the features could be the normalized vector of the weights of the features of LKDPI. By weighting the distance, the shape of E is an n-dimensional ellipsoid, not a sphere.

6.1 Calculating volumes

The volume of F and U remains the same, however, the volume of E now must change to represent the new weighted distance.
Let $c_1, c_2, \ldots, c_n$ be the weight coefficients of each of the n unit vectors. Note that these would also be the length of the semi-axes of the ellipsoid.

$$V_E = \frac{\pi^{\frac{n}{2}} \prod_{i=1}^{n} c_i}{\Gamma(\frac{n}{2}+1)}$$

6.2 Calculating the probability

The argument above still holds for an ellipsoid rather than a sphere. As we still have a calculable volume for E, the logic behind the derivation of $|S|$ for random sampling holds. An n dimensional ellipsoid can also be partitioned into 2^n congruent sectors, each constructed by the intersection of the n axes, therefore the logic behind the derivation of $|S|$ for evenly spaced samples holds. As these two arguments hold, the comparison between the two methods also remains valid.
Thus, as the randomized sampling method remains more optimal,

$$|S| = \frac{\ln(1-\alpha)}{\ln\left(1 - \frac{\pi^{\frac{n}{2}} \prod_{i=1}^{n} c_i}{\Gamma(\frac{n}{2}+1) \prod_{i=1}^{n} D_i}\right)}$$

This can be reduced to be put in terms of ϵ, as $\frac{\prod_{i=1}^{n} c_i}{\prod i{=}1^{n} D_i} = \epsilon^n$ where ϵ is the weighted distance as well as the constant dimensional scaling factor of the ellipsoid.
Thus we have

$$|S| = \frac{\ln(1-\alpha)}{\ln\left(1 - \frac{\pi^{\frac{n}{2}} \epsilon^n}{\Gamma(\frac{n}{2}+1)}\right)}$$

7 Using a non-uniform probability distribution

In all of the above, it was assumed that there was a uniform probability distribution in choosing points in n-space. However, in reality there exists a probability distribution for each dimension. Although this

does not change the method of even sampling, it does change the method of random sampling, as now,

$$\mathbf{Pr}[\text{dist}(p,q) \geq \epsilon] \neq 1 - V_E/V_F$$

for any $s \in S$. In order to recalculate $\mathbf{Pr}[\text{dist}(p,q) \geq \epsilon]$ for one sample, we would need calculate the expected probability of a sample being in E. As this would require an n dimensional integral over the probability distributions, this is very difficult to analyze. Instead, the following is a Las Vegas algorithm that will minimize $|S|$, while maintaining that $\mathbf{Pr}[\text{dist}(p,q) \leq \epsilon] \leq \alpha$.

7.1 When dimensions are assumed to be independent

Algorithm 1: A Las Vegas algorithm assuming dimension independence to minimize $|S|$

Input : The probability distributions for each of the n dimensions, $p_1, p_2, \ldots, p_n$
α, $\mathbf{Pr}[\text{dist}(p,q) \leq \epsilon]$
All previous definitions listed already

Output: S, such that $|S|$ is minimized, and $\mathbf{Pr}[\text{dist}(p,q) \leq \epsilon] \leq \alpha$

$T \leftarrow false$
$S \leftarrow nil$ `//Set of samples`
$V \leftarrow 0$ `//Volume we have covered`
while $T \neq true$ **do**
  $s \leftarrow$ `random coordinate based on distributions //coordinates of the new sample`
  while *dist(s,t)* $\leq \frac{\epsilon}{2}$ `for any` $t \in S$ **do**
    $s \leftarrow$ `random coordinate based on distributions`
  end
  Add s to S
  $C \leftarrow \prod_{i=1}^{n} max(p_i)$ `//current highest probability point assuming independence`
  $V \leftarrow V + V_E$ `//change volume covered`
  if $C * (\prod_{i=1}^{n} D_i - V) < 1 - \alpha$ **then**
    $T \leftarrow true$ `//We know Pr[dist`$(p,q) \leq \epsilon$`] ≤` α
  else
    $H \leftarrow$ `the locus of points centered at` s `such that dist`$(e,s) \leq \frac{\epsilon}{2}$
    for $i = 1$ *to* n **do**
      p_i `from` $[s.i - c_i, s.i + c_i] \leftarrow$

$$p_i \text{ from } [s.i - c_i, s.i + c_i] - \frac{\sqrt{1-(\frac{x_1^2}{c_1^2}+\frac{x_2^2}{c_2^2}+\cdots+\frac{s.i^2}{c_i^2}+\cdots+\frac{x_{j-1}^2}{c_{j-1}^2}+\frac{x_{j+1}^2}{c_1^2}+\cdots+\frac{x_n^2}{c_n^2})}\pi^{\frac{n}{2}} \prod_{j=1}^{n} c_j}{\Gamma(\frac{n}{2}+1)c_i} * D_i / \prod_{j=1}^{n} D_j$$

    end
  end
end

7.2 Proof of optimality / Explanation

The algorithm randomly samples, then constructs a hyper-ellipsoid around the point. It then updates each of the probability distributions that assumes independence. It terminates when $C * V_F - V < 1 - \alpha$, as we can bound α to be less than this number.

$$\alpha = \mathbf{Pr}[\text{dist}(p,q) \leq \epsilon]$$

This means that during runtime of the algorithm, α is the n dimensional integral of the probability distributions over the volume of remaining space. However, we can bound this by assuming that across the entire volume, each point has probability C of being chosen. As we know that C is the point of

highest probability ($\prod_{i=1}^{n} \max(p_i)$), the n dimensional integral cannot be more than C multiplied by the remaining volume. If we knew the dependence between dimensions, then we would be able to get a much better bound for C, as we would also be able to determine if the point that is chosen with probability C is already in a hyper-ellipsoid.

H is defined to be the locus of points centered at s such that $\text{dist}(e, s) \leq \frac{\epsilon}{2}$ for any point e. We make the distance threshold $\frac{\epsilon}{2}$ in order to avoid unnecessary intersection between two hyper-ellipsoids, while at the same time ensuring that hyper-ellipsoids are not too close to each other. The question now becomes how to calculate the volume of the cross section of H at a given point. Let the cross section of H and a given coordinate $s.i$ be called I. I is an $n-1$ dimensional ellipsoid.
The volume of I can be expressed as

$$V_I = k(s.i)\frac{\pi^{\frac{n}{2}} \prod_{j=1}^{n} c_j}{\Gamma(\frac{n}{2}+1)c_i}$$

for some scaling function k(x). This is because all of the cross sections of the ellipsoid are similar to each other. The scaling function k(x) is equal to the magnitude of the vector of any dimension other than i where the value of the coordinate in the ith dimension is $s.i$ divided by the magnitude of the vector where the value of the coordinate in the ith dimension is 0. More specifically, the equation of E is

$$\frac{x_1^2}{c_1^2} + \frac{x_2^2}{c_2^2} + \cdots + \frac{x_n^2}{c_n^2} = 1$$

for some varying values $x_1, x_2, \ldots, x_n$.

$$k(x) = \sqrt{1 - (\frac{x_1^2}{c_1^2} + \frac{x_2^2}{c_2^2} + \cdots + \frac{s.i^2}{c_i^2} + \cdots + \frac{x_{j-1}^2}{c_{j-1}^2} + \frac{x_{j+1}^2}{c_1^2} + \cdots + \frac{x_n^2}{c_n^2})}$$

Thus

$$V_I = \frac{\sqrt{1 - (\frac{x_1^2}{c_1^2} + \frac{x_2^2}{c_2^2} + \cdots + \frac{s.i^2}{c_i^2} + \cdots + \frac{x_{j-1}^2}{c_{j-1}^2} + \frac{x_{j+1}^2}{c_1^2} + \cdots + \frac{x_n^2}{c_n^2})}\pi^{\frac{n}{2}} \prod_{j=1}^{n} c_j}{\Gamma(\frac{n}{2}+1)c_i} \tag{7}$$

As the algorithm continues adding samples only until both constraints are satisfied, it is optimal. Having better bounds on C will increase performance.

8 Proving the uniform probability bound

Intuitively, it seems as though $|S|$ when we assume uniform probability distributions will always be greater than when we assume there to be probability distributions, as by creating areas where points tend to cluster, we decrease the number of hyper ellipsoids that we must construct to cover the area. We can prove this by looking at the volumes of the spaces in which samples can fall.

8.1 Calculating volumes

For the uniform probability space, Let V_C denote the total volume that the hyper-ellipsoids need to cover. To show that by having non constant probability distributions always decreases $|S|$, we must show that the volume that has to be covered when assuming a non uniform distribution is strictly less than the volume that needs to be covered when assuming a uniform probability distribution.

Let us partition F into Q congruent rectangular prisms, each of which is similar to F. Let us also assume that we only know the probability distribution among each of these Q prisms. In other words, for each prism, we know the probability of a random sample falling into it, and nothing else. To prove that setting the probability r_i of landing in a cuboid q_i is $\frac{1}{Q}$ leads to having the greatest number of samples, we must show that the function denoting the probability of a donor-patient pair p requesting their value

for the metric and a sample x falling in the same cuboid is convex. If the function is convex, we know that if one cuboid has probability $\frac{1}{Q}+\delta$, and one cuboid has probability $\frac{1}{Q}-\delta$, there will be a net positive in the probability of their intersection in the same cuboid. Thus if we let Q to be , allowing us to sample individual points, having a uniform probability distribution still gives an upper bound.

$$\mathbf{Pr}[p \text{ falls in } q_i \cup x \text{ falls in } q_i] = r_i(1-(1-r_i)^s) = u_i$$

$$\frac{d^2u}{dx^2}u_i = -s(1-x)^{s-2}(sx+x-2)$$

The second derivative is positive, thus making the function convex when $s > 1, x < \frac{2}{s+1}$ This does not inherently prove that sampling from a uniform distribution is better, as the second derivative of the function is only positive given those conditions. However, the following argument is used to show that the uniform distribution is a conservative bound to the number of samples that needs to be taken.

8.2 Proving the uniform probability distribution bound given the second derivative

Let $b = \frac{2}{s+1}$. The function describing $\mathbf{Pr}[p \text{ falls in } q_i \cup x \text{ falls in } q_i]$ is convex up to b, and concave from b to 1. By the properties of convexity, there are three cases that must be considered.

1. All of the r_i are the same value in the range [0,b]
2. $\exists r_i > b$, and the remaining r_i are all the same, and $< b$
3. $\exists r_i = 1$ and the rest are 0

Case 3 can be ignored, as it is the ideal scenario, only one location must be sampled. Thus, it must be shown that case 1 provides a more conservative bound that case 2.

If case 1,

$$\mathbf{Pr}[p \text{ falls in } q_i \cup x \text{ falls in } q_i] = \frac{1}{Q}(1-(1-\frac{1}{Q})^s)$$

As we take the sum of these probabilities over all i, the total probability taking the union over all q_i is given by

$$(1-(1-\frac{1}{Q})^s)$$

If case 2, let the $r_i > b$ be w. The remaining samples must be $< b$ and the same. Let $v = \frac{1-w}{Q-1} = r_i$, $\forall r_i \neq b$. The total probability taking the union over all q_i is given by

$$(Q-1)v(1-(1-v)^s) + w(1-(1-w)^s) = (1-w)(1-(1-v)^s) + w(1-(1-w)^s)$$

As we are sampling individual points, we want to take the limit as Q approaches infinity of both of these sums.

$$\lim_{Q\to\infty}(1-(1-\frac{1}{Q})^s) = 0$$

$$\lim_{Q\to\infty}[(1-w)(1-(1-\frac{1-w}{Q-1})^s) + w(1-(1-w)^s)] = w-(1-w)^s w$$

As the latter limit is greater than the first limit, the uniform bound is more conservative.

Thus, to provide a simpler implementation at the cost of longer running times for pre-processing, it is viable to simply sample uniformly at random as it provides a conservative bound to the number of samples.

9 Determining utility offset from the number of samples

In the sampling process, it is important to know how much error in LKDPI or time we can expect to see in the outputted results. To determine this, we can make a reverse Lipschitz assumption based on ϵ.

9.1 Lipschitz assumption

Let x_1 and x_2 be two donor - patient pairs, and let $f_l(x_{i1}, x_{i2}, \ldots, x_{in})$ denote the expected LKDPI of pair x_1, and $f_t(x_{i1}, x_{i2}, \ldots, x_{in})$ denote the expected time it takes to match the pair. Let $\epsilon = \text{dist}(x_1, x_2)$. We assume that for a suitably small ϵ,

$$|f_l(x_{11}, x_{12}, \ldots, x_{1n}) - f_l(x_{21}, x_{22}, \ldots, x_{2n})| \geq K_l \text{dist}(x_1, x_2)$$

$$|f_t(x_{11}, x_{12}, \ldots, x_{1n}) - f_t(x_{21}, x_{22}, \ldots, x_{2n})| \geq K_t \text{dist}(x_1, x_2)$$

The reason why the Lipschitz assumption is flipped is because when two points are very close together, we claim that there is a bound to the amount of LKDPI and time offsets resulting from this distance. To find the Lipschitz constants K_l and K_t, we can randomly sample across hyperdimensional space, and calculate the offsets between LKDPI and time values within an ellipsoid. Dividing this offset by the distance between the samples gives the Lipschitz constants K_l and K_t. Finding the values of K_l and K_t such that the value $|1 - K_l|$ and $|1 - K_t|$ is maximized gives the maximum sampled error.