This part is about locality-sensitive families construction for Hamming distance, the cosine distance and for the normal Euclidean distance.

Terms in this section:

**Hyperplane**: a hyperplane is a subspace whose dimension is one less than that of its ambient space.

It sperate the space into two space.

**Normal** vector: a normal is an object such as a line, ray, or vector that is perpendicular to a given object. In this section is a vector that is perpendicular to a plane (or a hyperplane).

In a given space (it could be multiple-dimension space), two intersecting vectors will define a plane, and there must be an angle, is used in the textbook, between two vectors. And this angle is the cosine distance.

For the hyperplane choosing, there are uncountable Hyperplanes through intersection point of two vectors, origin is used in the textbook. Two hyperplanes are chosen randomly.

Two normal lines, dash line (a) and dot line (b), is provide in following picture.

Chart

Description automatically generated with low confidence

Vector x and y are in the different sides of a, which means the point on the x line has a different sign to the points on the y line. But if we choose line b as the normal vector, vector x and vector y will have a same sign.

The formular of the calculating the probability that a normal vector is like line a.

Only the line in the angle can be similar to line a. In this case, the probability is . Of course the probability that a normal vector as line b is 1- .

To make expression simple, we use +1 and -1 to express the normal vector, let’s say v, to multiple the vector x and y.

It can be written as: . If the result is positive, we record it as 1, otherwise, it si -1. We collection the result as a new vector which is the sketch. The sketches only contain +1, and -1. At last we can compare two sketches to calculate the ratio of the same index, use this percentage multiple 180.

However, this method is not accurate enough.

In a 2-dimensional Euclidean space.

Chart, diagram

Description automatically generated

All the hash function in the family will be associated with a randomly chosen line in this space.

The buckets on the line have a length of a. the points on d line would have a projection point in one of these buckets, which means it will be hashed into different bucket by a hashing function.

In this picture, it is clear that the length of d and the width of a, will decide the probability that a point hash into a bucket. If the two points is close enough, they will hash into the same bucket. These two points can be seen as equal.

For the example given by the textbook, d = a/2, so there are 50% chance the two points hash to the same bucket. I think 50% is made by 25%+ 25%.

For the picture above, the chance is that the probability that two points hashing into the left bucket add the probability for middle bucket and add the probability for the right bucket.

We may find that if the d is longer than a, the probability that two different point hashing into the same bucket is lower.

Of course, another element is the angle , if this angle is big enough, the probability two points hashing into one bucket is greater, if the angle is 90 degree, the percentage would be 100%.

For multiple-dimension space, there will be a locality-sensitive-function (d1, d2, p1, p2). It is still possible to use a random line and buckets with width of a to partition a line.