CSCI567 2013 Homework Assignment 1

Student Name	Arpit Bansal
Last 4 digits of USC ID	0979
I have collaborated with	Udit Agrawal Jia Li

1 Analysis of K-NN

1.1 Question 1

Given a ball $B(x_c, r)$ with center as x_c and radius r. Two cases arises here. one, when x is outside of the ball. Two, when x is inside of the ball. We will be using triangle inequality here which states that sum of any two sides of a triangle is greater than the third side of that triangle.

So using the triangle ineqaulity:

$$d(x, x_i) < d(x, x_c) + d(x_i, x_c)$$

where x_i is a data point inside the ball, x_c is center of the ball and $d(x, y) = \|x - y\|_2$. So upper bound u_c for $d(x, x_i)$ is $d(x, x_c) + d(x_i, x_c)$. Now as $d(x_i, x_c)$ is a distance between a point within the ball and its center, the maximum distance between these two points can only be r (radius of ball). So the upper bound u_c would be for a point x outside of the ball:

$$d(x, x_c) + r \tag{1}$$

Now, if the point x is inside of ball, then maximum possible value for $d(x, x_c)$ would also be r. So in this case u_c would be:

$$r + r = 2r \tag{2}$$

So we can conclude from (1) and (2) that u_c would be:

$$max\{d(x, x_c) + r, 2r\} \tag{3}$$

Now, for lower bound l_c, again using triangle inequality for different sides as below:

$$d(x, x_i) + d(x_i, x_c) > d(x, x_c)$$

or

$$d(x, x_i) > d(x, x_c) - d(x_i, x_c)$$

Here, for l_c, the right hand side of the above inequality should be minimized, i.e. maximize $d(x_i, x_c)$. So maximum value of $d(x_i, x_c)$ would be r. so lower bound l_c for a point x outside of the ball would be:

$$d(x, x_c) - r \tag{4}$$

Now, if the point x is inside the ball, minimum value of $d(x, x_i)$ would be zero when the choosen point x is same as x_i, i.e. $x = x_i$. So concluding from this deduction and equation (4), we can say that l_c would be:

$$max\{0, d(x, x_c) - r\} \tag{5}$$

So using (3) and (5), we can say that

$$\forall i, max\{0, d(x, x_c) - r\} \leq d(x, x_i) \leq max\{d(x, x_c) + r, 2r\} \tag{6}$$

where equality comes from when the points are nearly co-linear.

1.2 Question 2

Given is a ball $B(x_a, r)$ with x_a as center and radius r. Let us suppose that point x is outside of both the ball $B(x_c, r)$ and $B(x_a, r)$. Now for a point x_j inside the ball $B(x_a, r)$ by using (6):

$$\forall j, max\{0, d(x, x_a) - r\} \leq d(x, x_j) \leq max\{d(x, x_a) + r, 2r\} \tag{7}$$

Using (1), $u_c = d(x, x_c) + r$
Using (7), $l_a = d(x, x_a) - r$
So by putting above values in $u_c < l_a$

$$\begin{aligned} d(x, x_c) + r &< l_a = d(x, x_a) - r \\ \Rightarrow d(x, x_a) - d(x, x_c) &> 2r \end{aligned} \tag{8}$$

Also note here that we can not use other combinations of u_c and l_a because it will either give us an impossible inequality or an ineqaulity which can be computed from (8) itself.

1.3 Question 3

Using (8), we can say, if there are balls placed somewhere in the space and we take one test point x, if the difference between the distances from this point x to centers of those balls is greater than 2r, we will discard the ball having larger distance from point x. Now, if we look closely at "$2r''$, we can write it as $r_1 + r_2$, where r_1 and r_2 are the radiuses of two balls in consideration. So (8) transforms into

$$d(x, x_a) - d(x, x_c) > r_1 + r_2 \tag{9}$$

Now, we know that we can place ball anywhere in the space. We don't know about the data distribution so we can not decide about the placement of the balls directly. So intuitively, our first approach to the problem leads us to use some clustering algorithm so that we could place the balls for each cluster. So I, along with my group, devised an algorithm which used ***k-means***.

Now, suppose there are total of N data-points in our space and also we will assume that each ball will have N_c data-points which can vary according to cluster size. Now lets look at the algorithm defined as below:

Steps

(a) Randomly select k centroids (means) and initialize them. Now from each of the data-points, calculate the distance from these clusters (centroids) and assign these points to it's nearest cluster, i.e. merge the closest points to relevant centroid and make a bigger cluster. Now calculate the new centroids (means) for newly made bigger clusters. Repeat these steps until there is no change in the values of these means.

(b) Now as we have got k clusters, we have to place the balls. Find the farthest point (within a cluster itself) from the mean of the cluster and draw a circle of this radius with the mean as center of circle. Do this for all the k clusters.

(c) Now for a test point x, find its pairwise distance from each of the clusters' means in a loop. This arises several cases:

- If x satisfies the condition define by (9) and both of the balls have $N_c \geq K$ points, then discard the ball with larger distance. Here K is number of nearest neighbours to x.
- If x satisfies the condition define by (9) and the ball with center as x_c has $N_c < K$ points then consider both of the balls to find the K nearest neighbours.
- If x doesn't satisfy the condition define by (9), consider both of the balls to find the K nearest neighbours.

Now, the question arises how to choose k here. As there are N data-points and each ball contains N_c data-points. So k can be written as $\frac{N}{N_c}$. But ideally we want $N_c \geq K$, which implies that k should be little less than $\frac{N}{K}$. Also, the main drawback of this algorithm is that if none of the chosen balls has $N_c \geq K$, then we can not discard any of the balls, which increases computational complexity.

1.4 Question 4

Our goal is to minimize the function

$$\min_{m} \sum_{m,n} [||x_m - x_n||_2^2 - (p^T x_m - p^T x_n)^2]$$

We are given that $||p||_2 = 1$. so we can write these two functions in terms of Lagrange function which we can optimize with respect to p and λ, where $p \in R^D$ is a unit-length vector and λ is Lagrange Multiplier.

$$L(p, \lambda) = f(x, p) + \lambda(g(p))$$

where

$$f(x, p) = \sum_{m,n} [||x_m - x_n||_2^2 - (p^T x_m - p^T x_n)^2]$$

and

$$g(p) = ||p||_2 - 1$$

So by putting these values in function L we get,

$$L(p, \lambda) = \sum_{m,n} [||x_m - x_n||_2^2 - (p^T x_m - p^T x_n)^2] + \lambda(||p||_2 - 1)$$

As there are two parameters, we will be differentiating the above function with respect to both of these parameters and put them equal to zero to optimize the function. Lets simplify the function first.

$$L(p, \lambda) = \sum_{m,n} ||x_m - x_n||_2^2 - \sum_{m,n} \{p^T (x_m - x_n)\}^2 + \lambda(p^T p - 1)$$

$$L(p, \lambda) = \sum_{m,n} ||x_m - x_n||_2^2 - \sum_{m,n} [\{p^T (x_m - x_n)\}\{(x_m - x_n)^T p\}] + \lambda(p^T p - 1)$$

$$L(p, \lambda) = \sum_{m,n} ||x_m - x_n||_2^2 - p^T \{\sum_{m,n} (x_m - x_n)(x_m - x_n)^T\} p + \lambda(p^T p - 1)$$

$$L(p,\lambda) = \sum_{m,n} ||x_m - x_n||_2^2 - p^T X^T X p + \lambda(p^T p - 1)$$

where $\mathrm{X} = \begin{pmatrix} \vdots \\ (x_m - x_n)^T \\ \vdots \end{pmatrix} \in R^{N^2 \times D}$. So $f(x,p)$ becomes

$$f(x,p) = \sum_{m,n} ||x_m - x_n||_2^2 - p^T X^T X p \tag{10}$$

Now,

$$\frac{\delta L(p,\lambda)}{\delta p} = -(X^T X + (X^T X)^T)p + 2\lambda p$$

Now as $X^T X$ is a symmetric matrix of $R^{D\times D}$, $X^T X + (X^T X)^T$ is equivalent to $2X^T X$. Putting this value in above equation,

$$\Rightarrow -2X^T X p + 2\lambda p = 0$$
$$\Rightarrow X^T X p = \lambda p$$

So we can conclude here that p is an eigen vector and λ is the corresponding eigen value of the matrix $X^T X$. Also, $X^T X$ is a positive definite matrix which can be seen easily. So all the eigen values i.e. λ will be ≥ 0 for this matrix. As $X^T X$ is a $D \times D$ matrix, it will have D eigen vectors. So we will put every eigen vector in (10) by making sure that it's a unit vector and we will choose the p which will give the minimum value of (10).

1.5 Question 5

5.1

Probability for *"x's label is 1 while x(1)'s label is 0"* would be

$$P(y=1|x) * P(y=0|x(1))$$
$$\Rightarrow P(y=1|x) * (1 - P(y=1|x(1)))$$
$$\eta(x)(1-\eta(x(1))) \tag{11}$$

Probability for *"x's label is 0 while x(1)'s label is 1"* would be

$$P(y=0|x) * P(y=1|x(1))$$
$$\Rightarrow (1 - P(y=1|x)) * P(y=1|x(1))$$
$$(1-\eta(x))\eta(x(1)) \tag{12}$$

5.2

$R(f,x)$ = probability of making mistakes

We have already calculated the two cases of mistakes as in (11) and (12). So we can write $R(f,x)$ as sum of these equations for total mistakes made:

$$R(f,x) = \eta(x)(1-\eta(x(1))) + (1-\eta(x))\eta(x(1)) \tag{13}$$

5.3

Case 1: Risk $R(f^*, x)$ when $\eta(x) > 1 - \eta(x)$

$$P(y = 0|x) \Rightarrow 1 - P(y = 1|x) = 1 - \eta(x)$$

Case 2: Risk $R(f^*, x)$ when $\eta(x) < 1 - \eta(x)$

$$P(y = 1|x) \Rightarrow \eta(x)$$

As we can see that $R(f^*, x)$ is always equal to smaller value of the given condition. So we can write as below:

$$R(f^*, x) = min\{\eta(x), 1 - \eta(x)\} \tag{14}$$

5.4

We have to prove

$$R(f, x) = 2R(f^*, x)(1 - R(f^*, x)) \tag{15}$$

We know from (14) that $R(f^*, x)$ can have only two different values. So if $R(f^*, x) = \eta(x)$, then putting value in (15)

$$R(f, x) = 2\eta(x)(1 - \eta(x))$$

Now if $R(f^*, x) = 1 - \eta(x)$, then putting value in (15)

$$R(f, x) = 2\eta(x)(1 - \eta(x))$$

Now as per given lemma when $N \to \infty, \eta(x(1)) \to \eta(x)$. So using this property in (13), we get

$$R(f, x) = 2\eta(x)(1 - \eta(x))$$

which is same as above. So we can say that (15) holds true.

5.5

$$2E_x R(f^*, x)^2 = 2(R(f^*))^2 + Missing\ Item$$

We know that

$$2E_x R(f^*, x)^2 = 2E_x(R(f^*, x))^2 + 2\sigma^2(R(f^*, x))$$

We know that expectation of Bayesian Expected Conditional Risk ($R(f^*, x)$) is called as Bayesian Expected Risk ($R(f^*)$). So

$$E_x(R(f^*, x))^2 = (R(f^*))^2$$

So we can conclude that

$$Missing\ Item = 2\sigma^2(R(f^*, x)) \tag{16}$$

So given equation can be written as

$$R(f) = 2R(f^*)(1 - R(f^*)) - 2\sigma^2(R(f^*, x)) \tag{17}$$

The right hand side of (16) is a varaince of bayesian expected condition risk which would never be negative. So (17) can be rewritten as

$$R(f) \leq 2R(f^*)(1 - R(f^*))$$