Data Visualization

Finding Best Epsilon Problem:

Problem Statement

Given point x in R^d and y in R^d, If x and y overlap in some subspaces, can you give guarantee for which epsilon neighborhood of x and y, if I take points x' and y' in these neighborhoods, respectively; the same subspace overlap happens.

Solution

PartA:

Given a datapoint X (d dimensional) and its K Nearest Neighbors (data points) find best epsilon such that if we take any new datapoint within epsilon distance i.e., $p\ \varepsilon\ (X-\varepsilon, X+\varepsilon)$ K-NN of new datapoint still remains same.

Solution:

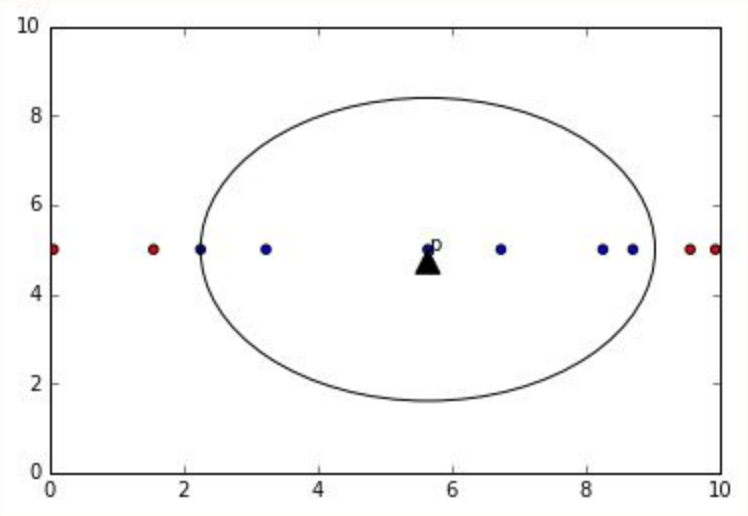
1-dimension:

Let

Number of nearest neighbors : 5

Total number of data points : 10

```
array([[ 5.64399276,  5.        ],
       [ 9.5631135 ,  5.        ],
       [ 0.05198069,  5.        ],
       [ 9.93909182,  5.        ],
       [ 1.54790498,  5.        ],
       [ 8.25926319,  5.        ],
       [ 8.70631162,  5.        ],
       [ 6.73668026,  5.        ],
       [ 3.2278668 ,  5.        ],
       [ 2.25016061,  5.        ]])
```



Circle represent the cluster of 5NN's of data point **'p'.**

Let this circle be a cluster of 6 data points including p and we need to find range of cluster boundaries such that data point within and outside cluster are unchanged.

Step2:

Two extreme points x1 and x2 are selected such that x1<p<x2 and if any point lies between x1 and x2 then no extra data points within cluster are added and no data point from with original cluster was removed.

For time being let us deal with positive and negative direction.

All points to the left of 'p' are in negative direction and points in right of 'p' are in positive direction.

x_1=midpoint of line joining leftmost extreme point (within cluster) and rightmost nearest point (outside cluster)

x_2=midpoint of line joining right most extreme(farthest) point (within cluster) and leftmost nearest point (outside cluster)

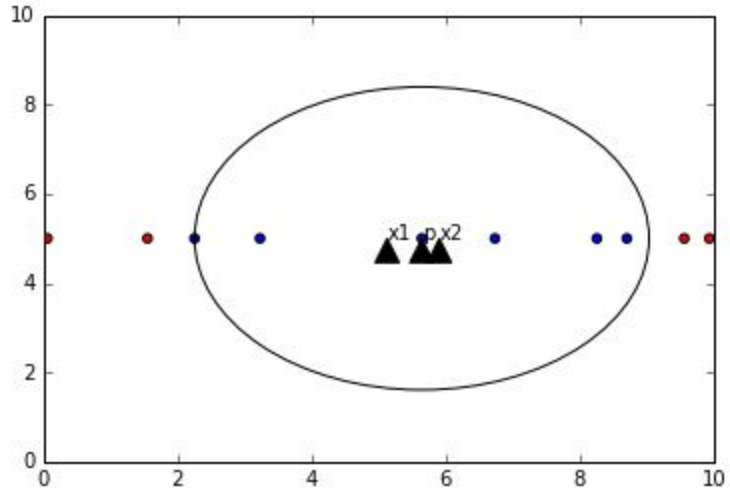
[NOTE: This can be generalized to higher dimension using cluster boundary concept. We can consider line passing through **'p'** (cluster center) joining each boundary point with nearest opposite direction non-cluster point and then find the midpoint] .

Epsilon (ε) = min $(d_{p,x1}, d_{p,x2})$

While dealing with more than 1-dimension

Epsilon (ε) = min $(d_{p,x1}, d_{p,x2}, d_{p,x3}, d_{p,x4}, \ldots\ldots\ldots\ldots\ldots, d_{p,xn})$

N: number of boundary points

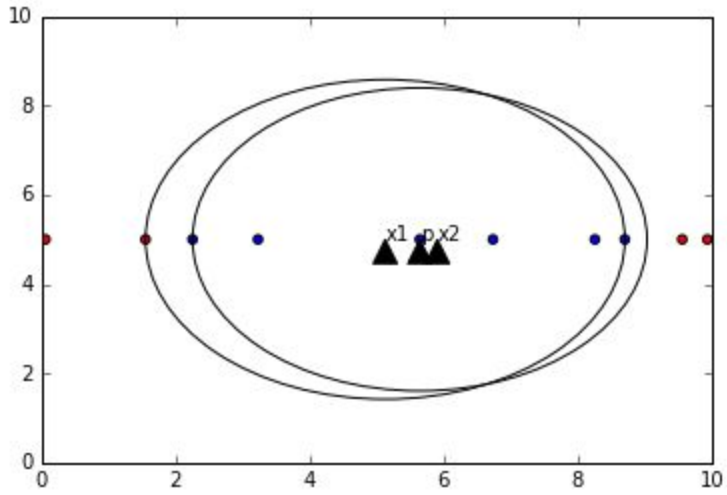


p=array([5.64399276, 5.   ])

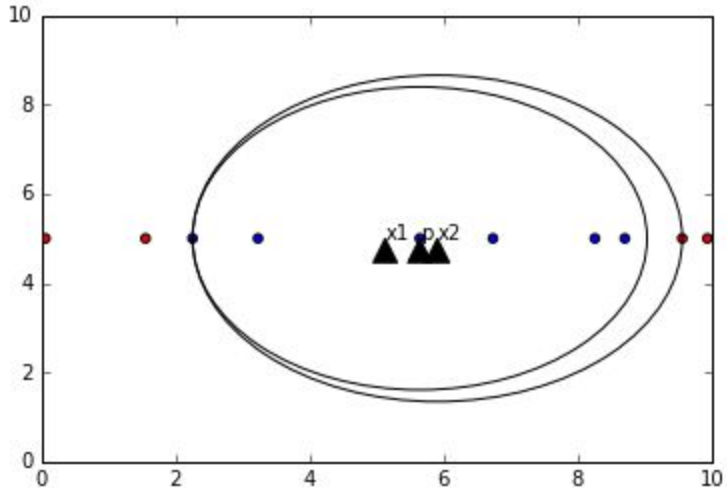
x_1=array([5.1271083, 5.   ])

x_2=array([5.90663705, 5.   ])

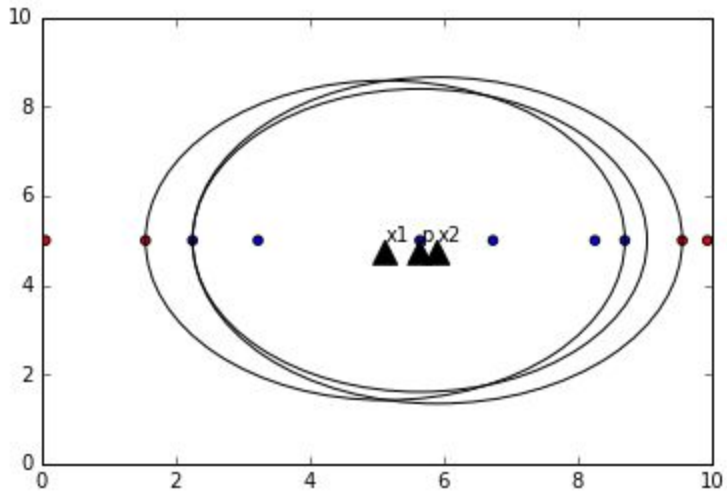
Circle depicting 5NN of x_1:



Circle depicting 5NN of x_2:



Both (5NN of x_1 and x_2):



Epsilon (ε) = min $(d_{p,x1}, d_{p,x2})$

Epsilon (ε) = min(0.5168844618528529 7,0.26264429201211925)

= 0.26264429201211925

Algorithm:

1) Find all boundary points (within cluster)
2) For each boundary point y
    a) Draw a vector passing through p, y and meeting the nearest outlier z
    b) Find midpoint of distance between y and z and call it x_k
3) Epsilon (ε) = min $(d_{p,x1}, d_{p,x2}, d_{p,x3}, d_{p,x4}, \ldots\ldots\ldots\ldots\ldots, d_{p,xn})$