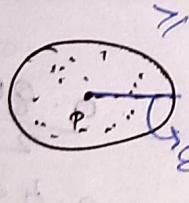
DBScan Algorithm:

* Arbitary select a point p
* Retrieve all pts density-reachable from (p) w.r.t ε_p and minPts
* If p is a core pt, a cluster is formed
* If p is a border points no points are density-reachable from p and DBSCAN visits the next of the database
* Continue the process until all of point have been processed

Core point:-



$\geq$ min pts.

p → core point

border point

(1) < min pts

(2) N_ε (core point) neighbour.

Noise if it is neither core or border point

Symmetric Distance Matrix | distance ⇒ Euclidean

pts	A	B	C	D	E	F
A	0					
B	0.7	0				
C	5.7	4.9	0			
D	3.6	2.9	2.9	0		
E	1.2	3.5	1.4	1	0	
F	3.2	2.5	2.5	0.5	1.1	0

Condition:

* $\varepsilon = 1.5$
* Min pts = 2

* Initially mark the core points (so $\varepsilon = 1.5$, min pts = 3, atleast we have 3 pts whose distance should be less than 3)

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$ Euclidean.

pts	A B/N	B B/N	C B/N	D core point	E core	F core
A	0 ✓	0.7	5.7	3.6	4.2	3.2
B	0.7 ✓	0 ✓	4.9	2.9	3.5	2.5
C	5.7 x	4.9 x	0 ✓	2.2	1.4 ✓	2.5
D	3.6 x	2.9 x	2.9 x	0 ✓	1 ✓	0.5 ✓
E	4.2 x	3.5 x	1.4 ✓	1 ✓	0 ✓	1.1 ✓
F	3.2 x	2.5 x	2.5 x	0.5 ✓	1.1 ✓	0 ✓
	↑ 2pts	↑ 1pt	↑ 2pt	↑ 3pts	↑ 4pts	↑ 3pts

* Then mark (find) Border point

pts	A Noise	B Noise	C Border	D core	E core	F core
A	0	0.7	5.7	3.6	4.2	3.2
B	0.7	0	4.9	2.9	3.5	2.5
C	5.7	4.9	0 ✓	2.2	1.4	2.5
D	3.6	2.9	2.9	0	1	
E	4.2	3.5	1.4	1	0	1.1
F	3.2	2.5	2.5	0.5	1.1	0

↑ 2pt ↑ 1pt ↑ 2pts

To determine C, whether C is Border or Noise.

condition (i) < min pts ($2 < 3$)

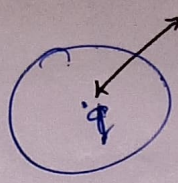
(ii) N_ε (with core point)

neighbor means falls under the radius ε

here C is neighbor with "E" which is the core point

Therefore "C" is border point

Directly density Reachable:



$p \in N_\varepsilon(q)$

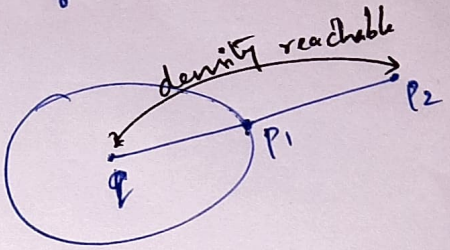
$q \rightarrow$ core point

$|N_\varepsilon(q)| \geq$ min pts

Density reachable :-

Chain of pts $\rightarrow p_1 \ldots p_n$

$p_1 = q$, $p_n = p$



Density connected :-