Northeastern University, Data Mining Techniques - CS6220
Fall 2017

Solutions to Homework 2, Part 2

Nakul Camasamudram

October 25, 2017

1. K-Means

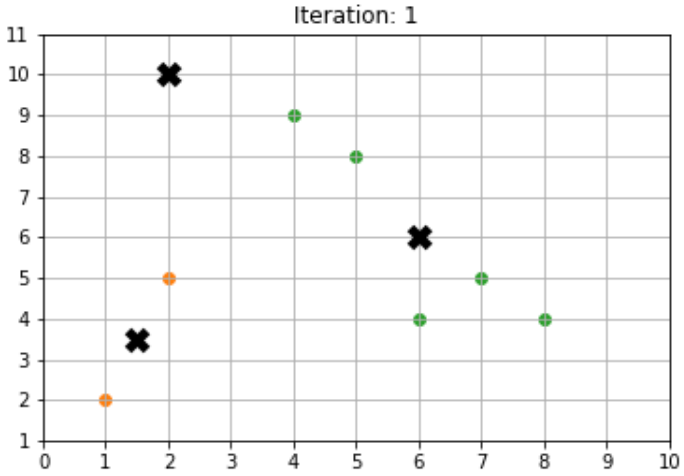
Solution:

- **Iteration 1:**
  The initial centroids are $C_1 = (2, 10)$ $C_2 = (1, 2)$ $C_3 = (5, 8)$. Let $\mathrm{D}(C_i)$ represent the **euclidean** distance between the respective point and the i^{th} centroid.

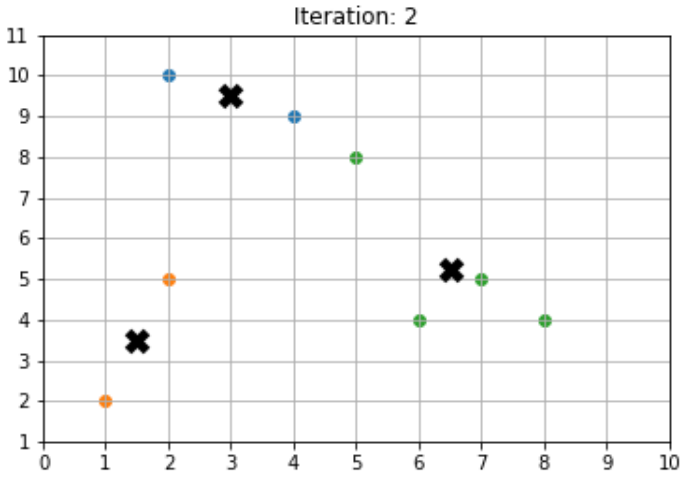
  The E-step:

Data Point	D(C_1)	D(C_2)	D(C_3)	Optimal centroid
(4,9)	2.23606797749979	7.615773105863909	**1.4142135623730951**	C_3
(2,10)	**0.0**	8.06225774829855	3.605551275463989	C_1
(1,2)	8.06225774829855	**0.0**	7.211102550927978	C_2
(2,5)	5.0	**3.1622776601683795**	4.242640687119285	C_2
(6,4)	7.211102550927978	5.385164807134504	**4.123105625617661**	C_3
(8,4)	8.48528137423857	7.280109889280518	**5.0**	C_3
(7, 5)	7.0710678118654755	6.708203932499369	**3.605551275463989**	C_3
(5, 8)	3.605551275463989	7.211102550927978	**0.0**	C_3

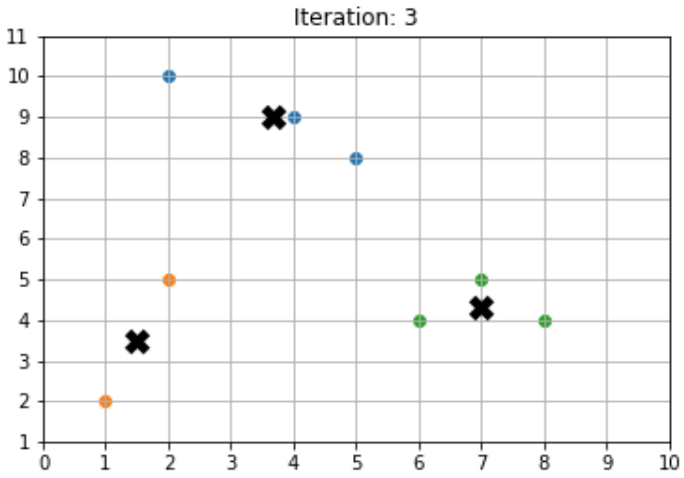
  The M-step:
  C_1 = mean[(2, 10)] = (2.0, 10.0)
  C_2 = mean[(1, 2), (2, 5)] = (1.5, 3.5)
  C_3 = mean[(4, 9), (6, 4), (8, 4), (7, 5), (5, 8)] = (6.0, 6.0)



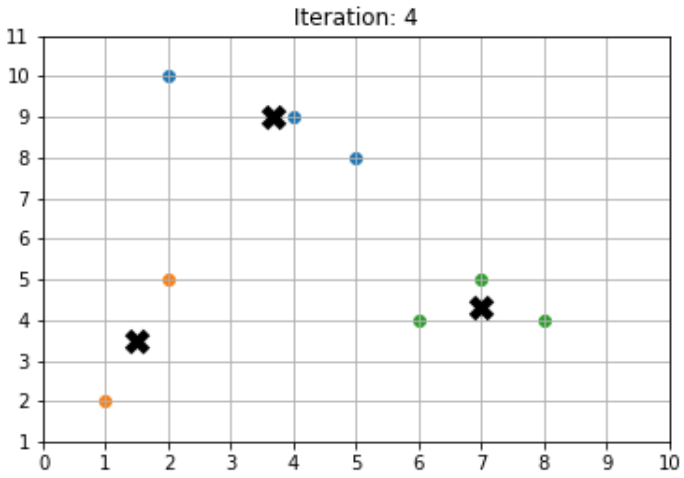
- **Iteration 2:**



- **Iteration 3:**



- **Iteration 4:**


Iteration: 4
11
10
9
8
7
6
5
4
3
2
1
0
1
2
3
4
5
6
7
8
9
10

2. Agglomerative Hierarchical

Solution:

1. Using MIN as an inter-cluster measure

Iteration 1:

	p_1	p_2	p_3	p_4	p_5	p_6
p_1						
p_2	0.2421					
p_3	0.2159	0.1523				
p_4	0.3677	0.1965	0.1581			
p_5	0.3418	0.1334	0.2846	0.2842		
p_6	0.2354	0.2530	**0.1020**	0.2195	0.3860	

Cluster 1: {3, 6}

- d({1},{3,6}) = min(d({1},{3}), d({1},{6})) = d({1},{3})
- d({2},{3,6}) = min(d({2},{3}), d({2},{6})) = d({2},{3})
- d({4},{3,6}) = min(d({4},{3}), d({4},{6})) = d({4},{3})
- d({5},{3,6}) = min(d({5},{3}), d({5},{6})) = d({5},{3})

Iteration 2:

	p_1	p_2	p_3	p_4	p_5	p_6
p_1						
p_2	0.2421					
p_3	0.2159	0.1523				
p_4	0.3677	0.1965	0.1581			
p_5	0.3418	**0.1334**	0.2846	0.2842		
p_6	~~0.2354~~	~~0.2530~~	~~0.1020~~	~~0.2195~~	~~0.3860~~	

Cluster 1: {3, 6}
Cluster 2: {2, 5}

- d({1},{2,5}) = min(d({1},{2}), d({1},{5})) = d({1},{2})
- d({4},{2,5}) = min(d({4},{2}), d({4},{5})) = d({4},{2})

- d({3,6},{2,5}) = min(d({3},{2}), d({3},{5}), d({6},{2}), d({6},{5})) = d({3},{2})

Iteration 3:

	p_1	p_2	p_3	p_4	p_5	p_6
p_1						
p_2	0.2421					
p_3	0.2159	**0.1523**				
p_4	0.3677	0.1965	0.1581			
p_5	~~0.3418~~	~~0.1334~~	~~0.2846~~	~~0.2842~~		
p_6	~~0.2354~~	~~0.2530~~	~~0.1020~~	~~0.2195~~	~~0.3860~~	

Cluster 1: {3, 6}
Cluster 2: {2, 5}
Cluster 3: {{2, 5}, {3, 6}}

- d{{2, 5, 3, 6}, {1}} = min(d{2, 1}, d{5, 1}, d{3, 1}, d{6, 1}) = d{3, 1}
- d{{2, 5, 3, 6}, {4}} = min(d{2, 4}, d{5, 4}, d{3, 4}, d{6, 4}) = d{3, 4}

Iteration 4:

	p_1	p_2	p_3	p_4	p_5	p_6
p_1						
p_2	~~0.2421~~					
p_3	0.2159	~~0.1523~~				
p_4	0.3677	~~0.1965~~	**0.1581**			
p_5	~~0.3418~~	~~0.1334~~	~~0.2846~~	~~0.2842~~		
p_6	~~0.2354~~	~~0.2530~~	~~0.1020~~	~~0.2195~~	~~0.3860~~	

Cluster 1: {3, 6}
Cluster 2: {2, 5}
Cluster 3: {{2, 5}, {3, 6}}
Cluster 4: {{2, 5, 3, 6}, {4}}

2. Using MAX as an inter-cluster measure

Iteration 1:

	p_1	p_2	p_3	p_4	p_5	p_6
p_1						
p_2	0.2421					
p_3	0.2159	0.1523				
p_4	0.3677	0.1965	0.1581			
p_5	0.3418	0.1334	0.2846	0.2842		
p_6	0.2354	0.2530	**0.1020**	0.2195	0.3860	

Cluster 1: {3, 6}

- d({1},{3,6}) = max(d({1},{3}), d({1},{6})) = d({1},{6})
- d({2},{3,6}) = max(d({2},{3}), d({2},{6})) = d({2},{6})
- d({4},{3,6}) = max(d({4},{3}), d({4},{6})) = d({4},{6})
- d({5},{3,6}) = max(d({5},{3}), d({5},{6})) = d({5},{6})

Iteration 2:

	p_1	p_2	p_3	p_4	p_5	p_6
p_1						
p_2	0.2421					
p_3	~~0.2159~~	~~0.1523~~				
p_4	0.3677	0.1965	~~0.1581~~			
p_5	0.3418	**0.1334**	~~0.2846~~	0.2842		
p_6	0.2354	0.2530	~~0.1020~~	0.2195	0.3860	

Cluster 1: {3, 6}
Cluster 2: {2, 5}

- d({1},{2,5}) = max(d({1},{2}), d({1},{5})) = d({1},{5})
- d({4},{2,5}) = max(d({4},{2}), d({4},{5})) = d({4},{5})
- d({3, 6},{2,5}) = max(d({3},{2}), d({3},{5}), d({6},{2}), d({6},{5})) = d({6},{5})

Iteration 3:

	p_1	p_2	p_3	p_4	p_5	p_6
p_1						
p_2	~~0.2421~~					
p_3	~~0.2159~~	~~0.1523~~				
p_4	0.3677	~~0.1965~~	~~0.1581~~			
p_5	0.3418	~~0.1334~~	~~0.2846~~	0.2842		
p_6	0.2354	~~0.2530~~	~~0.1020~~	**0.2195**	0.3860	

Cluster 1: {3, 6}
Cluster 2: {2, 5}
Cluster 3: {{3, 6}, {4}}

- d({1},{3,4,6}) = max(d({1},{3}), d({1},{4}), d({1},{6})) = d({1},{4})
- d({2,5},{3,4,6}) = max(d({2},{3}), d({2},{4}), d({2},{6}), d({5},{3}), d({5},{4}), d({5},{6})) = d({5},{6})

Iteration 4:

	p_1	p_2	p_3	p_4	p_5	p_6
p_1						
p_2	~~0.2421~~					
p_3	~~0.2159~~	~~0.1523~~				
p_4	0.3677	~~0.1965~~	~~0.1581~~			
p_5	**0.3418**	~~0.1334~~	~~0.2846~~	~~0.2842~~		
p_6	~~0.2354~~	~~0.2530~~	~~0.1020~~	~~0.2195~~	0.3860	

Cluster 1: {3, 6}
Cluster 2: {2, 5}
Cluster 3: {{3, 6}, {4}}
Cluster 4: {{2, 5}, {1}}

1. Using AVG as an inter-cluster measure

Iteration 1:

	p_1	p_2	p_3	p_4	p_5	p_6
p_1						
p_2	0.2421					
p_3	0.2159	0.1523				
p_4	0.3677	0.1965	0.1581			
p_5	0.3418	0.1334	0.2846	0.2842		
p_6	0.2354	0.2530	**0.1020**	0.2195	0.3860	

Cluster 1: {3, 6}

- d({1},{3,6}) = avg[d({1},{3}) + d({1},{6})] = 0.2256
- d({4},{3,6}) = avg[d({4},{3}) + d({4},{6})] = 0.1888
- d({2},{3,6}) = avg[d({2},{3}) + d({2},{6})] = 0.2026
- d({5},{3,6}) = avg[d({5},{3}) + d({5},{6})] = 0.3353

Iteration 2:

	p_1	p_2	p_3, p_6	p_4	p_5
p_1					
p_2	0.2421				
p_3, p_6	0.2256	0.2026			
p_4	0.3677	0.1965	0.1888		
p_5	0.3418	**0.1334**	0.3353	0.2842	

Cluster 1: {3, 6}
Cluster 1: {2, 5}

- d({1},{2,5}) = avg[d({1},{2}) + d({1},{5})] = 0.2919
- d({3,6},{2,5}) = avg[d({3,6},{2}) + d({3,6},{5})] = 0.2837
- d({4},{2,5}) = avg[d({4},{2}) + d({4},{5})] = 0.2404

Iteration 3:

	p_1	p_2, p_5	p_3, p_6	p_4
p_1				
p_2, p_5	0.2919			
p_3, p_6	0.2256	0.2185		
p_4	0.3677	0.2404	**0.1888**	

Cluster 1: {3, 6}
Cluster 2: {2, 5}
Cluster 3: {{3, 6}, {4}}

- d({1},{{3, 6}, {4}}) = avg[d({1},{3, 6}) + d({1},{4})] = 0.2967
- d({2, 5},{{3, 6}, {4}}) = avg[d({2, 5},{3, 6}) + d({2, 5},{4})] = 0.2295

Iteration 4:

	p_1	p_2, p_5	p_3, p_6
p_1			
p_2, p_5	0.2919		
p_3, p_4, p_6	0.2256	**0.2185**	

Cluster 1: {3, 6}
Cluster 2: {2, 5}
Cluster 3: {{3, 6}, {4}}
Cluster 4: {{3, 4, 6}, {2, 5}}

Iteration 5:

	p_1	p_2, p_3, p_4, p_5, p_6
p_1		
p_2, p_3, p_4, p_5, p_6	**0.2919**	

Cluster 1: {3, 6}
Cluster 2: {2, 5}
Cluster 3: {{3, 6},{4}}
Cluster 4: {{3, 4, 6}, {2, 5}}
Cluster 5: {1,{{3, 4, 6}, {2, 5}}}

3. DBSCAN

Solution:

- ⋆ **pt 0:** $2 <$ *MinPts*, so cluster=-1
- ⋆ **pt 1:** $3 \geq$ *MinPts*, so cluster=0
  `to_visit`=[40,75], `visited`={1}
  - · **pt 40:** cluster=0, $3 \geq$ *MinPts*, so adding neighbors
    `to_visit`=[75,28], `visited`={1,40}
  - · **pt 75:** cluster=0, $3 \geq$ *MinPts*, so adding neighbors
    `to_visit`=[28,4], `visited`={1,40,75}
  - · **pt 28:** cluster=0, $3 \geq$ *MinPts*, so adding neighbors
    `to_visit`=[4,12], `visited`={1,28,40,75}
  - · **pt 4:** cluster=0, $3 \geq$ *MinPts*, so adding neighbors
    `to_visit`=[12,56], `visited`={1,4,28,40,75}
  - · **pt 12:** cluster=0, $2 <$ *MinPts*,
    `to_visit`=[56], `visited` ={1,4,12,28,40,75}
  - · **pt 56:** cluster=0, $3 \geq$ *MinPts*, so adding neighbors
    `to_visit`=[66], `visited` ={1,4,12,28,40,56,75}
  - · **pt 66:** cluster=0, $3 \geq$ *MinPts*, so adding neighbors
    `to_visit`=[], `visited` ={1,4,12,28,40,56,66,75}
- ⋆ **pt 2:** $1 <$ *MinPts*, so cluster=-1
- ⋆ **pt 3:** $1 <$ *MinPts*, so cluster=-1
- ⋆ **pt 4:** cluster=0, so skip
- ⋆ **pt 5:** $3 \geq$ *MinPts*, so cluster=1
  `to_visit`=[70,74], `visited`={5}
  - · **pt 70:** cluster=1, $5 \geq$ *MinPts*, so adding neighbors
    `to_visit`=[74,32,69,72], `visited`={5,70}
  - · **pt 74:** cluster=1, $4 \geq$ *MinPts*, so adding neighbors
    `to_visit`=[32,69,72,19,54], `visited`={5,70,74}
  - · **pt 32:** cluster=1, $5 \geq$ *MinPts*, so adding neighbors
    `to_visit`=[69,72,19,54,63], `visited`={5,32,70,74}

- **pt 69:** cluster=1, 4 ≥ *MinPts*, so adding neighbors `to_visit`=[72,19,54,63], `visited`={5,32,69,70,74}
- **pt 72:** cluster=1, 7 ≥ *MinPts*, so adding neighbors `to_visit`=[19,54,63,8,60], `visited`={5,32,69,70,72,74}
- **pt 19:** cluster=1, 3 ≥ *MinPts*, so adding neighbors `to_visit`=[54,63,8,60], `visited`={5,19,32,69,70,72,74}
- **pt 54:** cluster=1, 4 ≥ *MinPts*, so adding neighbors `to_visit`=[63,8,60,25], `visited`={5,19,32,54,69,70,72,74}
- **pt 63:** cluster=1, 7 ≥ *MinPts*, so adding neighbors `to_visit`=[8,60,25,50,68], `visited`={5,19,32,54,63,69,70,72,74}
- **pt 8:** cluster=1, 5 ≥ *MinPts*, so adding neighbors `to_visit`=[60,25,50,68,11], `visited`={5,8,19,32,54,63,69,70,72,74}
- **pt 60:** cluster=1, 6 ≥ *MinPts*, so adding neighbors `to_visit`=[25,50,68,11], `visited`={5,8,19,32,54,60,63,69,70,72,74}
- **pt 25:** cluster=1, 4 ≥ *MinPts*, so adding neighbors `to_visit`=[50,68,11,26,67], `visited`={5,8,19,25,32,54,60,63,69,70,72,74}
- **pt 50:** cluster=1, 5 ≥ *MinPts*, so adding neighbors `to_visit`=[68,11,26,67,39], `visited`={5,8,19,25,32,50,54,60,63,69,70,72,74}
- **pt 68:** cluster=1, 5 ≥ *MinPts*, so adding neighbors `to_visit`=[11,26,67,39], `visited`={5,8,19,25,32,50,54,60,63,68,69,70,72,74}
- **pt 11:** cluster=1, 3 ≥ *MinPts*, so adding neighbors `to_visit`=[26,67,39,14], `visited`={5,8,11,19,25,32,50,54,60,63,68,69,70,72,74}
- **pt 26:** cluster=1, 3 ≥ *MinPts*, so adding neighbors `to_visit`=[67,39,14,34], `visited`={5,8,11,19,25,26,32,50,54,60,63,68,69,70,72,74}
- **pt 67:** cluster=1, 2 < *MinPts*, `to_visit`=[39,14,34], `visited`={5,8,11,19,25,26,32,50,54,60,63,67,68,69,70,72,74}
- **pt 39:** cluster=1, 5 ≥ *MinPts*, so adding neighbors `to_visit`=[14,34,10,71], `visited`={5,8,11,19,25,26,32,39,50,54,60,63,67,68,69,70,72,74}
- **pt 14:** cluster=1, 3 ≥ *MinPts*, so adding neighbors `to_visit`=[34,10,71,6], `visited`={5,8,11,14,19,25,26,32,39,50,54,60,63,67,68,69,70,72,74}
- **pt 34:** cluster=1, 4 ≥ *MinPts*, so adding neighbors `to_visit`=[10,71,6,29,46], `visited`={5,8,11,14,19,25,26,32,34,39,50,54,60,63,67,68,69,70,72,
- **pt 10:** cluster=1, 4 ≥ *MinPts*, so adding neighbors `to_visit`=[71,6,29,46,22], `visited`={5,8,10,11,14,19,25,26,32,34,39,50,54,60,63,67,68,69,70,
,74}

- **pt 71:** cluster=1, 4 $\geq$ *MinPts*, so adding neighbors `to_visit`=[6,29,46,22], `visited`={5,8,10,11,14,19,25,26,32,34,39,50,54,60,63,67,68,69,70,71,72,74}
- **pt 6:** cluster=1, 3 $\geq$ *MinPts*, so adding neighbors `to_visit`=[29,46,22,42], `visited`={5,6,8,10,11,14,19,25,26,32,34,39,50,54,60,63,67,68,69,70,71,72,74}
- **pt 29:** cluster=1, 4 $\geq$ *MinPts*, so adding neighbors `to_visit`=[46,22,42,16], `visited`={5,6,8,10,11,14,19,25,26,29,32,34,39,50,54,60,63,67,68,69,70,71,72,74}
- **pt 46:** cluster=1, 4 $\geq$ *MinPts*, so adding neighbors `to_visit`=[22,42,16], `visited`={5,6,8,10,11,14,19,25,26,29,32,34,39,46,50,54,60,63,67,68,69,70,71,72,74}
- **pt 22:** cluster=1, 3 $\geq$ *MinPts*, so adding neighbors `to_visit`=[42,16], `visited`={5,6,8,10,11,14,19,22,25,26,29,32,34,39,46,50,54,60,63,67,68,69,70,71,72,74}
- **pt 42:** cluster=1, 4 $\geq$ *MinPts*, so adding neighbors `to_visit`=[16,17,20], `visited`={5,6,8,10,11,14,19,22,25,26,29,32,34,39,42,46,50,54,60,63,67,68,69,70,71,72,74}
- **pt 16:** cluster=1, 4 $\geq$ *MinPts*, so adding neighbors `to_visit`=[17,20,48], `visited`={5,6,8,10,11,14,16,19,22,25,26,29,32,34,39,42,46,50,54,60,63,67,68,69,70,71,72,74}
- **pt 17:** cluster=1, 3 $\geq$ *MinPts*, so adding neighbors `to_visit`=[20,48], `visited`={5,6,8,10,11,14,16,17,19,22,25,26,29,32,34,39,42,46,50,54,60,63,67,68,69,70,71,72,74}
- **pt 20:** cluster=1, 4 $\geq$ *MinPts*, so adding neighbors `to_visit`=[48,38], `visited`={5,6,8,10,11,14,16,17,19,20,22,25,26,29,32,34,39,42,46,50,54,60,63,67,68,69,70,71,72,74}
- **pt 48:** cluster=1, 2 $\geq$ *MinPts*, `to_visit`=[38], `visited`={5,6,8,10,11,14,16,17,19,20,22,25,26,29,32,34,39,42,46,48,50,54,60,63,67,68,69,70,71,72,74}
- **pt 38:** cluster=1, 5 $\geq$ *MinPts*, so adding neighbors `to_visit`=[30,37,45], `visited`={5,6,8,10,11,14,16,17,19,20,22,25,26,29,32,34,38,39,42,46,48,50,54,60,63,67,68,69,70,71,72,74}
- **pt 30:** cluster=1, 4 $\geq$ *MinPts*, so adding neighbors `to_visit`=[37,45,52], `visited`={5,6,8,10,11,14,16,17,19,20,22,25,26,29,30,32,34,38,39,42,46,48,50,54,60,63,67,68,69,70,71,72,74}

- **pt 37:** cluster=1, 4 ≥ *MinPts*, so adding neighbors
  `to_visit`=[45,52,53], `visited`={5,6,8,10,11,14,16,17,19,20,22,25,26,29,30,32,34,37,38,39,42,
  46,48,50,54,60,63,67,68,69,70,71,72,74}
- **pt 45:** cluster=1, 4 ≥ *MinPts*, so adding neighbors
  `to_visit`=[52,53], `visited`={5,6,8,10,11,14,16,17,19,20,22,25,26,29,30,32,34,37,38,39,
  42,45,46,48,50,54,60,63,67,68,69,70,71,72,74}
- **pt 52:** cluster=1, 4 ≥ *MinPts*, so adding neighbors
  `to_visit`=[53,49,64], `visited`={5,6,8,10,11,14,16,17,19,20,22,25,26,29,30,32,34,37,38,39,42,
  45,46,48,50,52,54,60,63,67,68,69,70,71,72,74}
- **pt 53:** cluster=1, 3 ≥ *MinPts*, so adding neighbors
  `to_visit`=[49,64,47], `visited`={5,6,8,10,11,14,16,17,19,20,22,25,26,29,30,32,34,37,38,39,42,
  45,46,48,50,52,53,54,60,63,67,68,69,70,71,72,74}
- **pt 49:** cluster=1, 4 ≥ *MinPts*, so adding neighbors
  `to_visit`=[64,47,31,76], `visited`={5,6,8,10,11,14,16,17,19,20,22,25,26,29,30,32,34,37,38,39,
  45,46,48,49,50,52,53,54,60,63,67,68,69,70,71,72,74}
- **pt 64:** cluster=1, 2 < *MinPts*,
  `to_visit`=[47,31,76], `visited`={5,6,8,10,11,14,16,17,19,20,22,25,26,29,30,32,34,37,38,39,42,
  45,46,48,49,50,52,53,54,60,63,64,67,68,69,70,71,72,74}
- **pt 47:** cluster=1, 2 < *MinPts*,
  `to_visit`=[31,76], `visited`={5,6,8,10,11,14,16,17,19,20,22,25,26,29,30,32,34,37,38,39,
  42,45,46,47,48,49,50,52,53,54,60,63,64,67,68,69,70,71,72,74}
- **pt 31:** cluster=1, 2 < *MinPts*,
  `to_visit`=[76], `visited`={5,6,8,10,11,14,16,17,19,20,22,25,26,29,30,31,32,34,37,38,39,42,
  45,46,47,48,49,50,52,53,54,60,63,64,67,68,69,70,71,72,74}
- **pt 76:** cluster=1, 3 ≥ *MinPts*, so adding neighbors
  `to_visit`=[21], `visited`={5,6,8,10,11,14,16,17,19,20,22,25,26,29,30,31,32,34,37,38,39,42,
  45,46,47,48,49,50,52,53,54,60,63,64,67,68,69,70,71,72,74,76}
- **pt 21:** cluster=1, 2 < *MinPts*,
  `to_visit`=[], `visited`={5,6,8,10,11,14,16,17,19,20,21,22,25,26,29,30,31,32,34,37,38,39,
  42,45,46,47,48,49,50,52,53,54,60,63,64,67,68,69,70,71,72,74,76}

⋆ **pt 6:** cluster=1, so skip

⋆ **pt 7:** 1 < *MinPts*, so cluster=-1

⋆ **pt 8:** cluster=1, so skip

⋆ **pt 9:** 3 ≥ *MinPts*, so cluster=2
`to_visit`=[33,78], `visited`={9}

- **pt 33:** cluster=2, 3 $\geq$ *MinPts*, so adding neighbors `to_visit`=[78], `visited`={9,33}
- **pt 78:** cluster=2, 3 $\geq$ *MinPts*, so adding neighbors `to_visit`=[], `visited`={9,33,78}

⋆ **pt 10:** cluster=1, so skip

⋆ **pt 11:** cluster=1, so skip

⋆ **pt 12:** cluster=0, so skip

⋆ **pt 13:** 2 < *MinPts*, so cluster=-1

⋆ **pt 14:** cluster=1, so skip

⋆ **pt 15:** 1 < *MinPts*, so cluster=-1

⋆ **pt 16:** cluster=1, so skip

⋆ **pt 17:** cluster=1, so skip

⋆ **pt 18:** 1 < *MinPts*, so cluster=-1

⋆ **pt 19:** cluster=1, so skip

⋆ **pt 20:** cluster=1, so skip

⋆ **pt 21:** cluster=1, so skip

⋆ **pt 22:** cluster=1, so skip

⋆ **pt 23:** 1 < *MinPts*, so cluster=-1

⋆ **pt 24:** 1 < *MinPts*, so cluster=-1

⋆ **pt 25:** cluster=1, so skip

⋆ **pt 26:** cluster=1, so skip

⋆ **pt 27:** 2 < *MinPts*, so cluster=-1

⋆ **pt 28:** cluster=0, so skip

⋆ **pt 29:** cluster=1, so skip

⋆ **pt 30:** cluster=1, so skip

⋆ **pt 31:** cluster=1, so skip

⋆ **pt 32:** cluster=1, so skip

⋆ **pt 33:** cluster=2, so skip

⋆ **pt 34:** cluster=1, so skip

⋆ **pt 35:** $2 < MinPts$, so cluster=-1

⋆ **pt 36:** $1 < MinPts$, so cluster=-1

⋆ **pt 37:** cluster=1, so skip

⋆ **pt 38:** cluster=1, so skip

⋆ **pt 39:** cluster=1, so skip

⋆ **pt 40:** cluster=0, so skip

⋆ **pt 41:** $1 < MinPts$, so cluster=-1

⋆ **pt 42:** cluster=1, so skip

⋆ **pt 43:** $2 < MinPts$, so cluster=-1

⋆ **pt 44:** $1 < MinPts$, so cluster=-1

⋆ **pt 45:** cluster=1, so skip

⋆ **pt 46:** cluster=1, so skip

⋆ **pt 47:** cluster=1, so skip

⋆ **pt 48:** cluster=1, so skip

⋆ **pt 49:** cluster=1, so skip

⋆ **pt 50:** cluster=1, so skip

⋆ **pt 51:** $2 < MinPts$, so cluster=-1

⋆ **pt 52:** cluster=1, so skip

⋆ **pt 53:** cluster=1, so skip

⋆ **pt 54:** cluster=1, so skip

⋆ **pt 55:** 2 < *MinPts*, so cluster=-1

⋆ **pt 56:** cluster=0, so skip

⋆ **pt 57:** 1 < *MinPts*, so cluster=-1

⋆ **pt 58:** 1 < *MinPts*, so cluster=-1

⋆ **pt 59:** 2 < *MinPts*, so cluster=-1

⋆ **pt 60:** cluster=1, so skip

⋆ **pt 61:** 1 < *MinPts*, so cluster=-1

⋆ **pt 62:** 2 < *MinPts*, so cluster=-1

⋆ **pt 63:** cluster=1, so skip

⋆ **pt 64:** cluster=1, so skip

⋆ **pt 65:** 1 < *MinPts*, so cluster=-1

⋆ **pt 66:** cluster=0, so skip

⋆ **pt 67:** cluster=1, so skip

⋆ **pt 68:** cluster=1, so skip

⋆ **pt 69:** cluster=1, so skip

⋆ **pt 70:** cluster=1, so skip

⋆ **pt 71:** cluster=1, so skip

⋆ **pt 72:** cluster=1, so skip

⋆ **pt 73:** 1 < *MinPts*, so cluster=-1

⋆ **pt 74:** cluster=1, so skip

⋆ **pt 75:** cluster=0, so skip

⋆ **pt 76:** cluster=1, so skip

⋆ **pt 77:** 2 < *MinPts*, so cluster=-1

⋆ **pt 78:** cluster=2, so skip

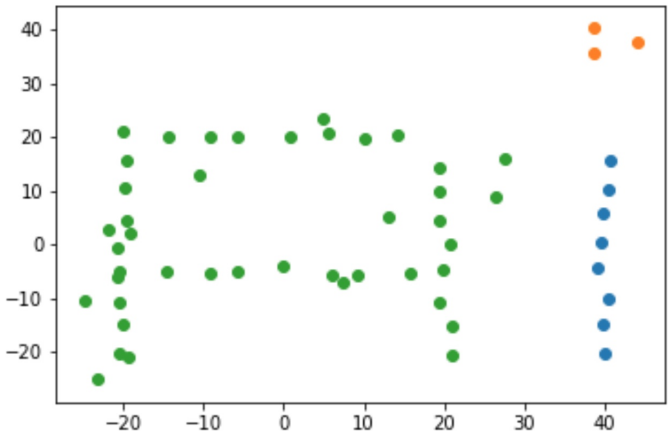
⋆ **pt 79:** $1 < MinPts$, so cluster=-1

Final Clusters:

1. Cluster 0: 1, 4, 12, 28, 40, 56, 66, 75
2. Cluster 1: 5, 6, 8, 10, 11, 14, 16, 17, 19, 20, 21, 22, 25, 26, 29, 30, 31, 32, 34, 37, 38, 39, 42, 45, 46, 47,
   48, 49, 50, 52, 53, 54, 60, 63, 64, 67, 68, 69, 70, 71, 72, 74, 76
3. Cluster 2: 9, 33, 78
4. Cluster -1: 0, 2, 3, 7, 13, 15, 18, 23, 24, 27, 35, 36, 41, 43, 44, 51, 55, 57, 58, 59, 61, 62, 65, 73, 77, 79

Extra Credit

Upon plotting the above points highlighted by their clusters, and ignoring cluster "-1",



I'm inferring that the above plot is a reference to the AI Magazine. This is the link to the Table of Contents of the most recent issue(Volume 38, Number 3).

The author of the first article "Steps Toward Robust Artificial Intelligence" is "Thomas G. Dietterich". He is Emeritus Professor of computer science at Oregon State University.

Interesting facts

- Thomas G. Dietterich has 347 publications to his credit on Google Scholar.
- From 2014-2016, he was the President of the Association for the Advancement of Artificial Intelligence(AAAI)
- As a PhD student, he sang in the Stanford University Chorus. It was in the choir that he met Carol Rivin, who would later become his wife