

a. Kelebihan/Kelemahan k-Means

1. Sempel untuk diimplementasikan dan menghasilkan output yang mudah diinterpretasi
2. Dapat mengolah dataset yang besar
3. Efisien dalam mensegmentasikan dataset yang clusternya berbentuk hyperspherical.
4. Memiliki kompleksitas linear  $O(tknd)$ , t adalah jumlah iterasi, k adalah jumlah cluster, n jumlah data untuk setiap dimensi (d)
1. Efek uniform meskipun input data yang dimasukan berbeda ukurannya.
2. Tidak ada jumlah cluster yang optimal, operator harus menset jumlah clusternya
3. K-means sensitif terhadap posisi inialisasi titik centroid
4. Sensitif terhadap perubahan data anomali/outlier
5. Clustering berbentuk lingkaran

b. Agglomerative Clustering

Agglomerative clustering adalah salah satu teknik hierarchical clustering untuk mengklasifikasikan data secara hirarkis. “Agglomerative” berarti mulai secara bottom-up.

Algoritma dasar:

Input: Sebuah list data C

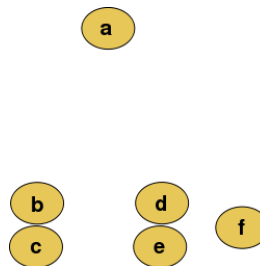
Lakukan perulangan sampai tersisa 1 kluster

Cari pasangan dari clusters yang terdekat  $\min D(c_i, c_j)$

Gabung clusters  $c_i, c_j$  menjadi cluster baru  $c_{i+j}$

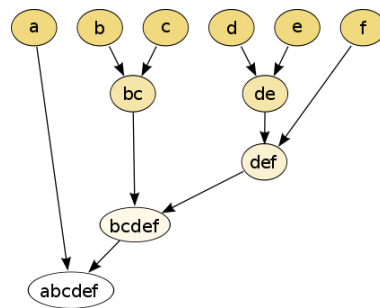
Hapus  $c_i, c_j$  dari list data C, masukan cluster baru tadi ke list C

Agglomerative Clustering bekerja dengan mengklasifikasi secara bottom up



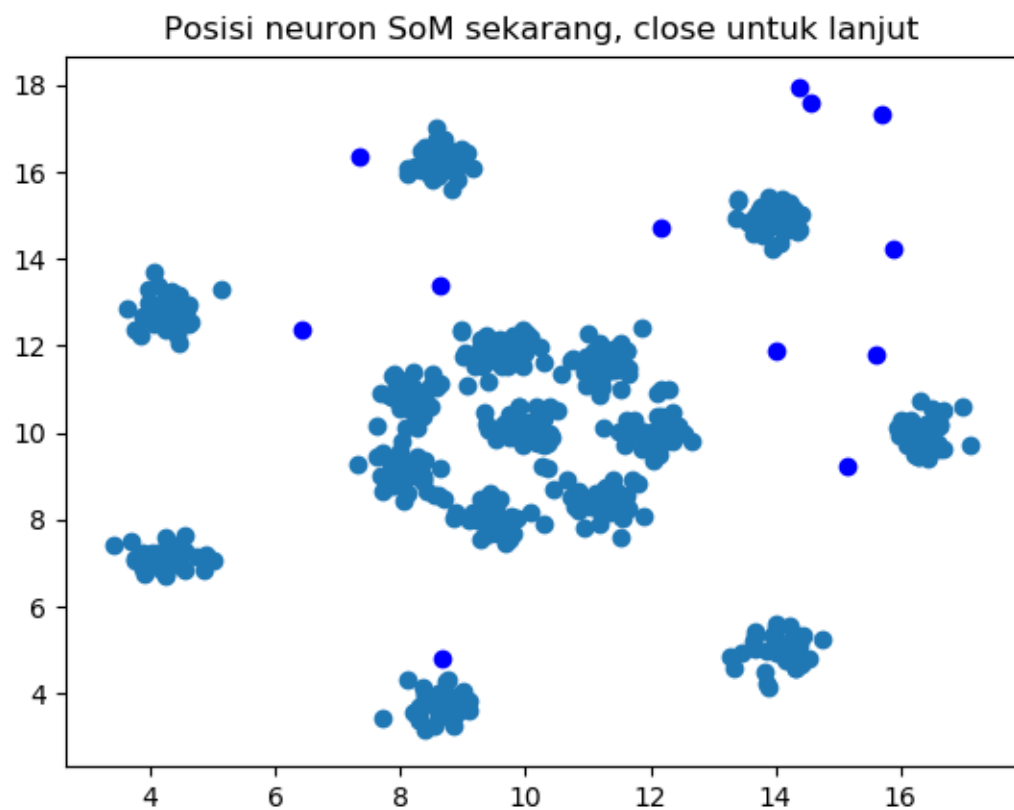
Gambar 1 contoh data input

Pada contoh ini, algoritma mengklasifikasi data b, c lalu d, e. Data f diklasifikasikan dengan de. Data bc, def, digabungkan dengan a.



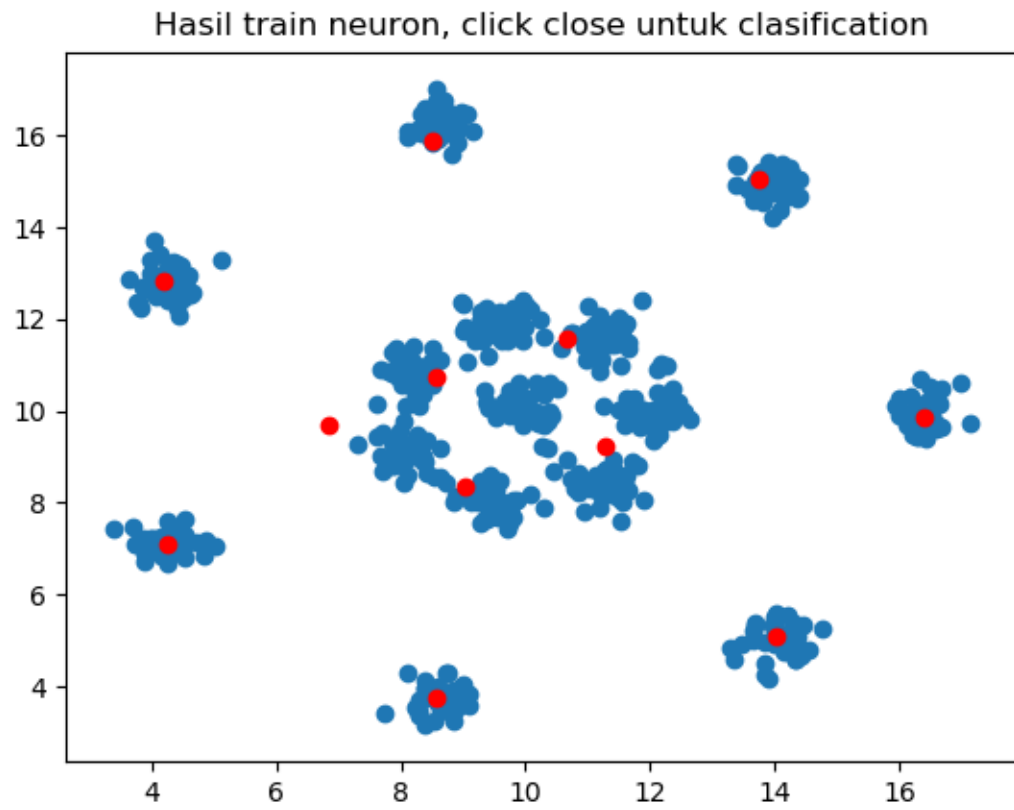
*Gambar 2 Setelah diproses menjadi tree*

c. Hasil clustering dengan SOM

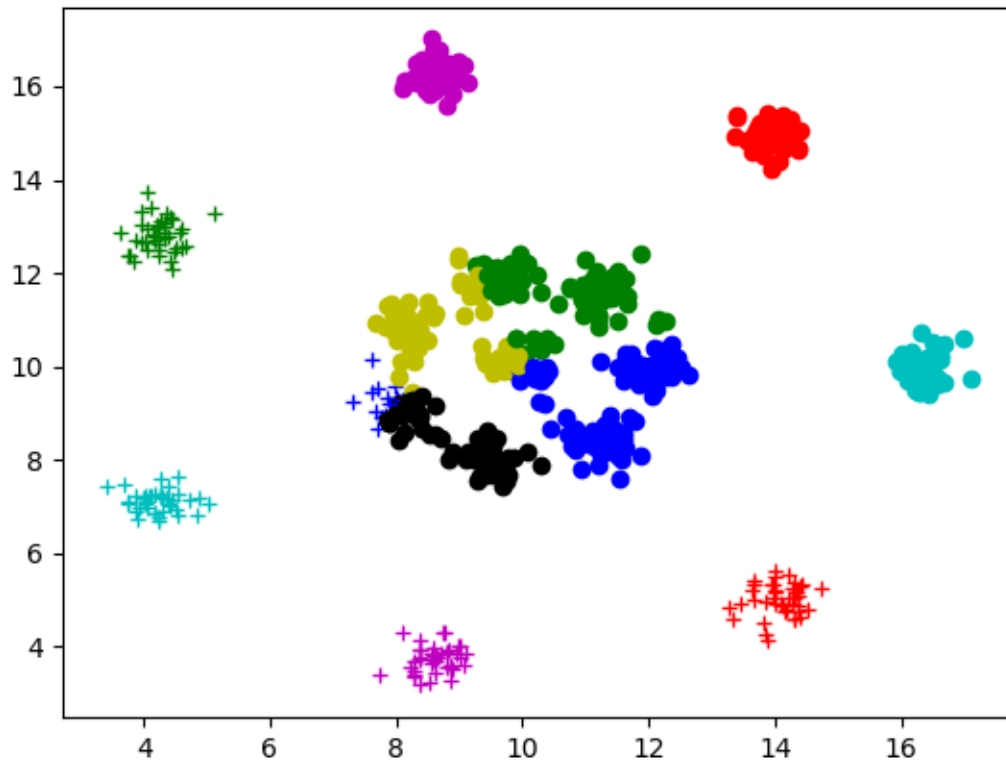


*Gambar 3 Inisialisasi Training*

Titik biru adalah data mentahnya sedangkan biru tua adalah peta SOMnya. Ini adalah hasil akhirnya.



*Gambar 4 Hasil Training*



*Gambar 5 Hasil Klasifikasi*

Parameter:

```
epochs = 10000
neighbour = 1
col = 5
row = 3
learning_rate = 0.4
```

Algoritmanya:

1. Inisialisasi array untuk node SOM panjang  $\times$  lebar  $\times$  dimensi dan learning rate
2. Lakukan perulangan hingga dengan epochs tertentu
3. Pilih data random dari data set
4. Cari BMU (best matching unit) dengan menggunakan rumus ecludian untuk data tersebut terhadap setiap kolom dan baris neuron SOM

5. Update SOM tersebut dengan rumus berikut

$$\text{som}[x][y] = \text{som}[x][y] + \text{currentRate} * -( \text{som}[x][y] - \text{dataRandomYangDipilih} )$$