Kümeleme Algoritmaları

Kümeleme algoritmaları, veri noktalarını benzer özelliklerine göre gruplara ayırarak, verideki gizli desenleri ve yapıları keşfetmeye yarayan yöntemlerdir. Kümeleme, denetimsiz bir öğrenme türüdür çünkü veriler üzerinde herhangi bir etiketleme yapılmaz. Kümeleme algoritmaları, verileri birbirine en çok benzeyen gruplar (kümeler) halinde düzenlemeye çalışır. İşte en yaygın kullanılan kümeleme algoritmalarının detaylı açıklamaları:

K-Means Kümeleme

K-Means Kümeleme, belirli sayıda küme belirleyerek (K) veri noktalarını bu kümelere ayıran bir algoritmadır.

1. **Başlangıç Adımı**:
    - K değeri belirlenir (kaç küme oluşturulacağı).
    - K tane rasgele merkez (centroid) seçilir.
2. **Atama Adımı**:
    - Her veri noktası, en yakın merkeze atanır. Bu işlem, genellikle Öklidyen mesafe kullanılarak yapılır.
3. **Güncelleme Adımı**:
    - Her küme için yeni bir merkez hesaplanır. Yeni merkez, o kümedeki veri noktalarının ortalamasıdır.
4. **Tekrar**:
    - Atama ve güncelleme adımları, merkezlerin konumu değişmeyene kadar veya belirli bir iterasyon sayısına ulaşılana kadar tekrarlanır.

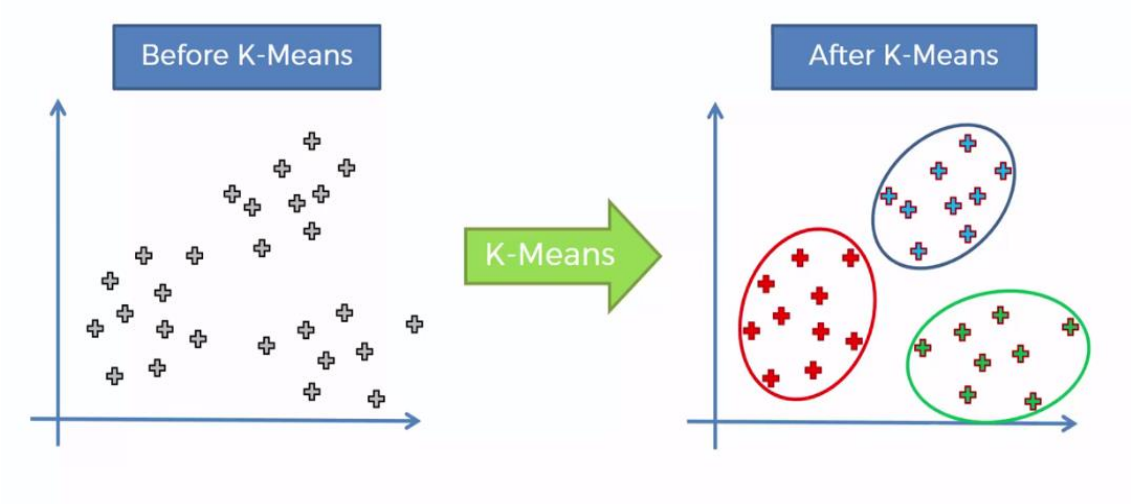
Örnek: Müşterileri satın alma alışkanlıklarına göre segmentlere ayırmak için kullanılabilir.

Avantajları:

- Basit ve hızlıdır.
- Büyük veri setlerinde etkili çalışır.

Dezavantajları:

- Küme sayısının (K) önceden belirlenmesi gerekir.
- Rasgele başlatma merkezleri farklı sonuçlara yol açabilir.
- Küresel olmayan kümelerde ve farklı yoğunluklardaki veri setlerinde iyi performans göstermeyebilir.



K MEANS

Hiyerarşik Kümeleme

Hiyerarşik Kümeleme, veri noktalarını hiyerarşik bir yapı içinde kümeleyen bir algoritmadır. İki ana türü vardır: Aglomeratif (birleştirici) ve bölücü (divisive).

1. **Aglomeratif Kümeleme**:
    - Başlangıçta her veri noktası kendi kümesi olarak kabul edilir.
    - En yakın iki küme bulunur ve birleştirilir.
    - Bu işlem, tek bir küme kalana kadar tekrarlanır.
2. **Bölücü Kümeleme**:
    - Başlangıçta tüm veri noktaları tek bir küme olarak kabul edilir.
    - Bu küme, alt kümelere bölünür.
    - Bu işlem, her bir veri noktası kendi kümesi olana kadar devam eder.

Örnek: Genetik veri analizinde, türler arasındaki benzerlikleri incelemek için kullanılabilir.

Avantajları:

- Küme sayısının önceden belirlenmesi gerekmez.
- Kümeleme sonucu dendrogram (ağaç diyagramı) ile görselleştirilebilir.

Dezavantajları:

- Büyük veri setlerinde hesaplama maliyeti yüksektir.
- Kümeleme sonucunun birleştirme veya bölme stratejisine (örneğin, en yakın komşu, en uzak komşu, ortalama bağlantı) bağımlılığı vardır.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

DBSCAN, veri noktalarını yoğunluk temelli kümeleme yapan bir algoritmadır. Bu algoritma, veri noktalarını yoğun bölgelerdeki kümelere ayırır ve düşük yoğunluklu bölgelerdeki noktaları gürültü olarak kabul eder.

1. **Parametreler**:
    - **eps**: Bir noktanın komşusu sayılacak mesafe.
    - **minPts**: Bir noktanın çekirdek nokta sayılması için sahip olması gereken minimum komşu sayısı.
2. **Algoritma Adımları**:
    - Bir çekirdek nokta seçilir ve **eps** yarıçapındaki tüm komşuları bulunur.
    - Eğer komşu sayısı **minPts**'den fazla ise bu nokta bir kümenin parçası olarak kabul edilir.
    - Komşulara aynı işlem uygulanarak küme genişletilir.
    - Bu işlem tüm noktalar için tekrarlanır.

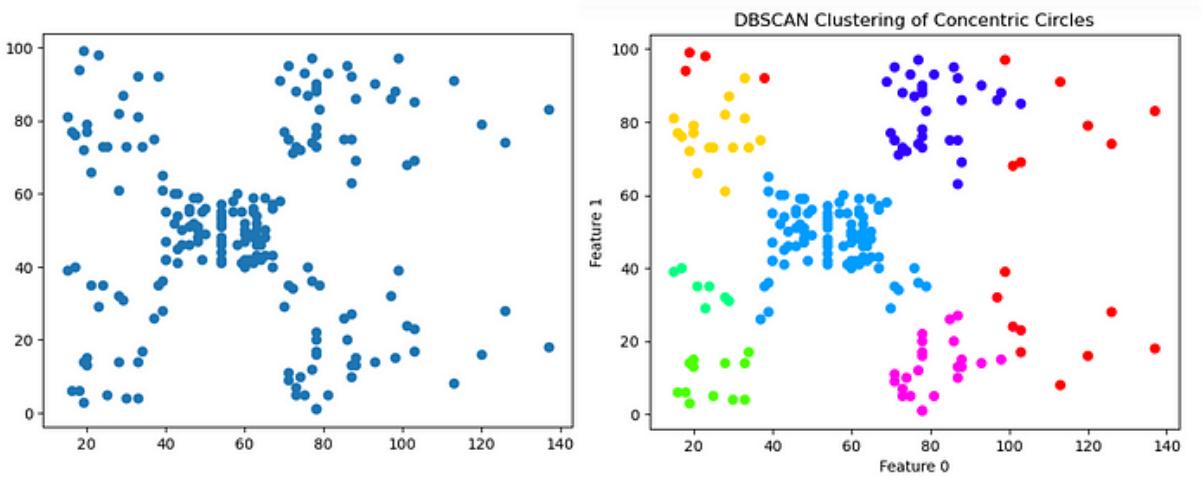
Örnek: Coğrafi veri analizi, gürültülü veri kümelerinin tanımlanması.

Avantajları:

- Küme sayısını önceden belirlemek gerekmez.
- Gürültüyü (aykırı değerleri) belirleme yeteneği vardır.
- Küresel olmayan ve farklı yoğunluktaki kümeleri keşfetme yeteneği vardır.

Dezavantajları:

- **eps** ve **minPts** parametrelerinin seçimi zordur ve veri kümesine bağlı olarak değişebilir.
- Yüksek boyutlu veri kümelerinde performans düşebilir.



DBSCAN

Mean-Shift Kümeleme

Mean-Shift Kümeleme, yoğunluk temelli bir kümeleme algoritmasıdır. Veri noktalarını yoğunluk tepe noktalarına göre gruplar.

1. **Başlangıç Adımı**:
    - Veri noktalarına başlangıçta rastgele bir pencere (window) atanır.
2. **Güncelleme Adımı**:
    - Her bir veri noktasının yoğunluk merkezi hesaplanır ve bu noktaya doğru pencere kaydırılır.
3. **Tekrar**:
    - Pencereler kaydırılmaya devam edilir ve pencere yoğunluk merkezleri sabitlenene kadar işlem tekrarlanır.

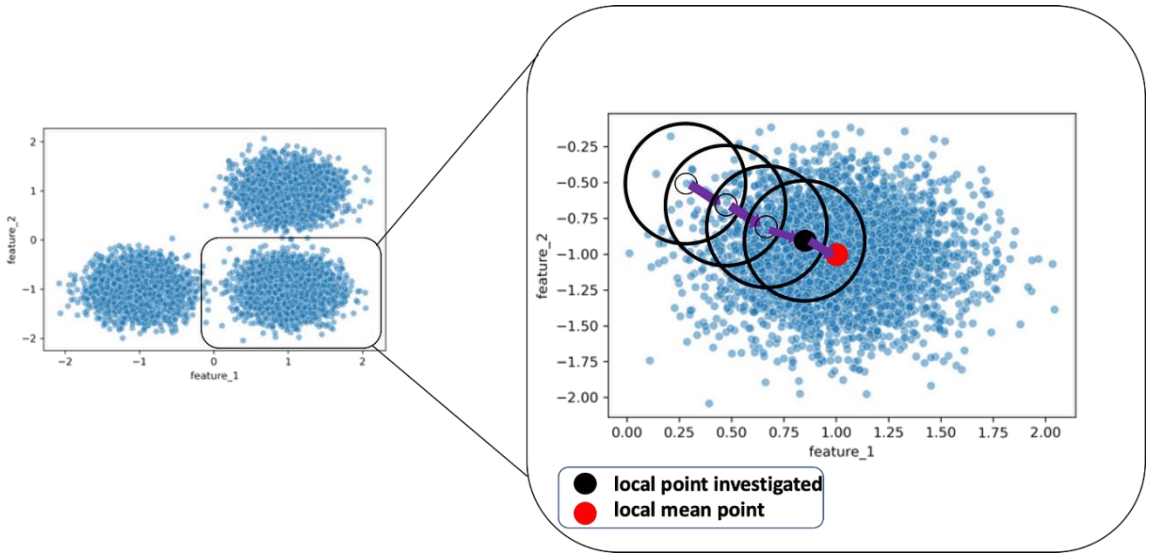
Örnek: Görüntü işleme, nesne tespiti.

Avantajları:

- Küme sayısını önceden belirlemek gerekmez.
- Küresel olmayan ve farklı yoğunluktaki kümeleri keşfetme yeteneği vardır.

Dezavantajları:

- Hesaplama maliyeti yüksektir.
- Büyük veri setlerinde yavaş çalışabilir.



MEAN-SHİFT

Özet

Kümeleme algoritmaları, veri setlerindeki gizli desenleri ve yapıları ortaya çıkarmak için kullanılır. Hangi algoritmanın seçileceği, veri setinin özelliklerine ve problemin doğasına bağlıdır. K-Means, Hiyerarşik Kümeleme, DBSCAN ve Mean-Shift gibi yaygın algoritmaların her birinin avantajları ve dezavantajları vardır. Verilerinizi analiz ederken ve kümeleme algoritmalarını seçerken, bu faktörleri göz önünde bulundurmanız önemlidir.