



Python ile Yapay Zekaya Giriş

Makine Öğrenimine Giriş

Terminoloji

- Model (Model)
- Öznitelik (Feature)
- Eğitim (Training)
- Tahmin (Prediction)
- Etiket (Label)

Python ile Yapay Zekaya Giriş

Makine Öğrenimine Giriş

Terminoloji

- Model (Model)
- Öznitelik (Feature)
- Eğitim (Training)
- Tahmin (Prediction)
- Etiket (Label)



"model," bir veri kümesi üzerinde öğrenilen ve gelecekteki veriler hakkında tahminlerde bulunabilen matematiksel bir yapıyı ifade eder. Model, genellikle bir algoritma kullanılarak oluşturulur ve veriden örüntüler, ilişkiler veya yapılar çıkarır. Bu model, bir "öğrenme" sürecinin sonucudur.

Python ile Yapay Zekaya Giriş

Makine Öğrenimine Giriş

Terminoloji

- Model (Model)
- Öznitelik (Feature)
- Eğitim (Training)
- Tahmin (Prediction)
- Hedef (Target)



Veri kümesinin ölçülebilir bir özelliği ya da parametresidir.

Ev Fiyatı Tahmini Dataset Örneği:

Ev No	Yatak Odası Sayısı (Feature)	Banyo Sayısı (Feature)	Ev Büyüklüğü (m ²) (Feature)	Mahalle (Feature)	Ev Fiyatı (Target)
1	3	2	120	A	250,000
2	2	1	80	B	180,000
3	4	3	200	A	350,000
4	3	2	150	C	230,000
5	5	4	300	A	500,000

Python ile Yapay Zekaya Giriş

Makine Öğrenimine Giriş

Terminoloji

- Model (Model)
- Öznitelik (Feature)
- Eğitim (Training) → Buna da mı tanım yazalım ?
- Tahmin (Prediction)
- Hedef (Target)

Python ile Yapay Zekaya Giriş

Makine Öğrenimine Giriş

Terminoloji

- Model (Model)
- Öznitelik (Feature)
- Eğitim (Training)
- Tahmin (Prediction) → Modelin çıktısı
- Hedef (Target)

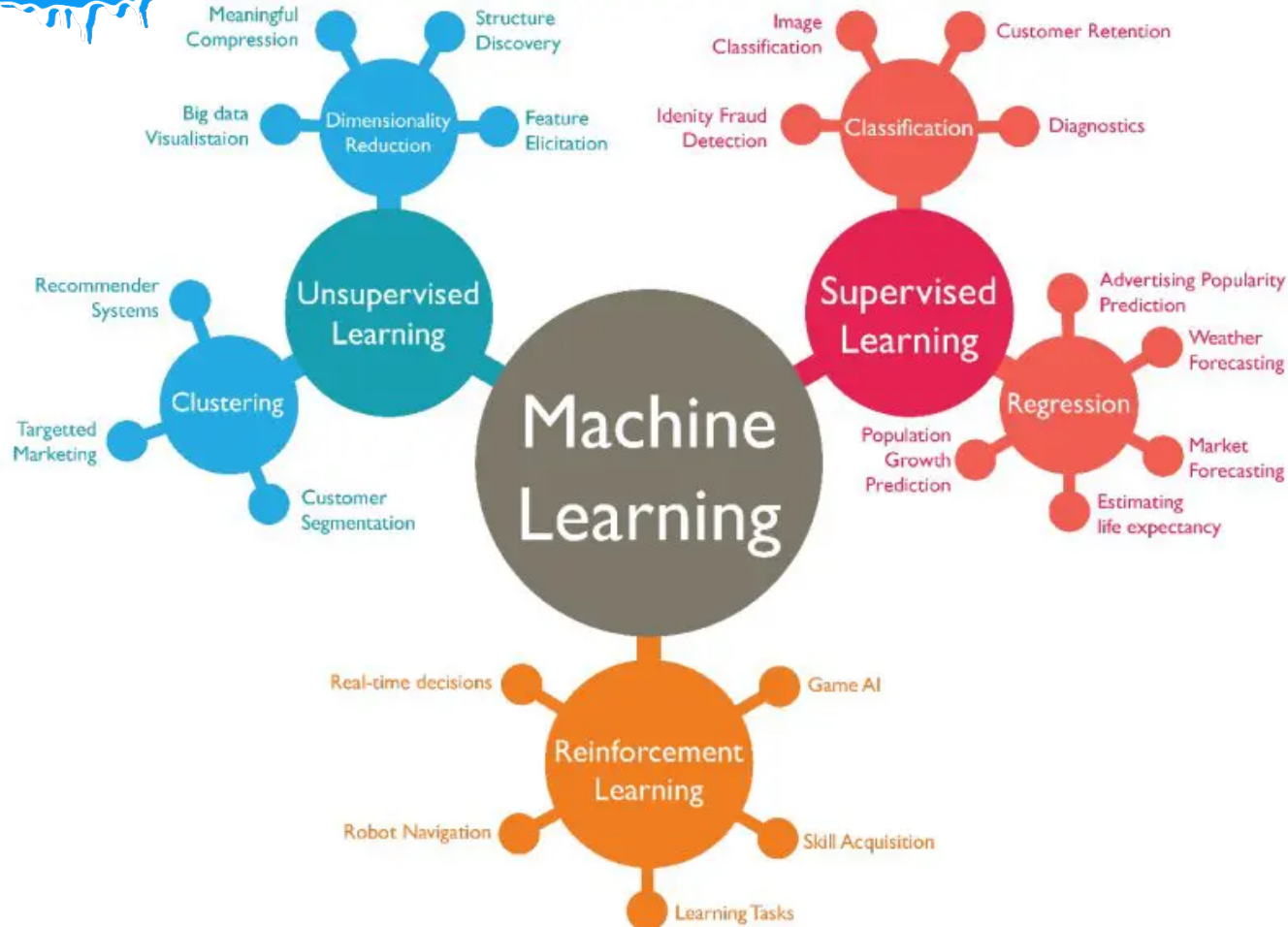
Python ile Yapay Zekaya Giriş

Makine Öğrenimine Giriş

Terminoloji

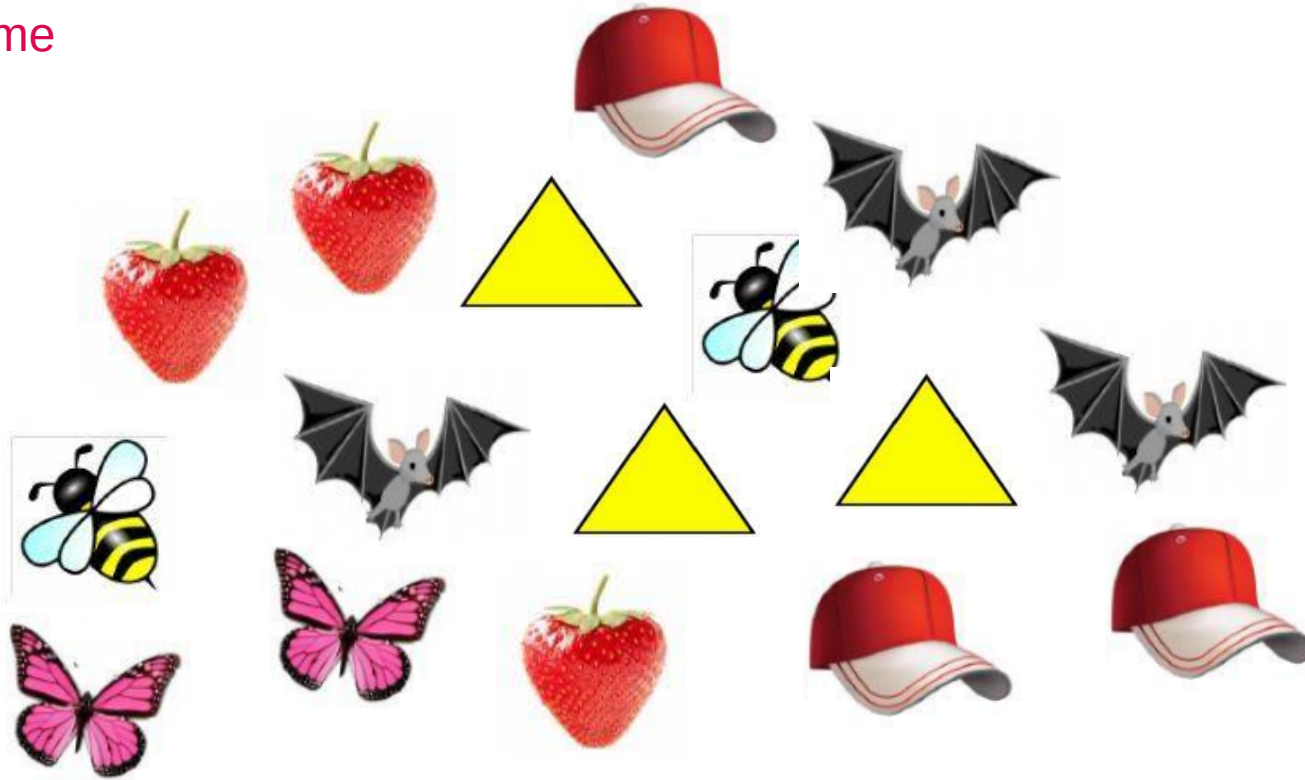
- Model (Model)
- Öznitelik (Feature)
- Eğitim (Training)
- Tahmin (Prediction)
- Hedef (Target) → Beklenen değer

Python ile Yapay Zekaya Giriş



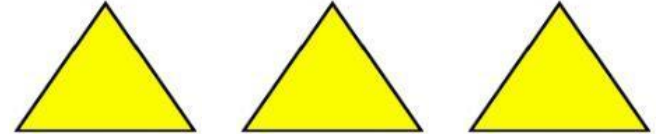
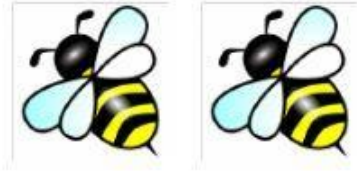
Python ile Yapay Zekaya Giriş

Gözetimsiz Öğrenme



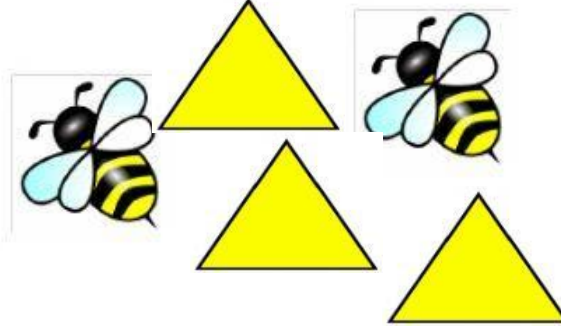
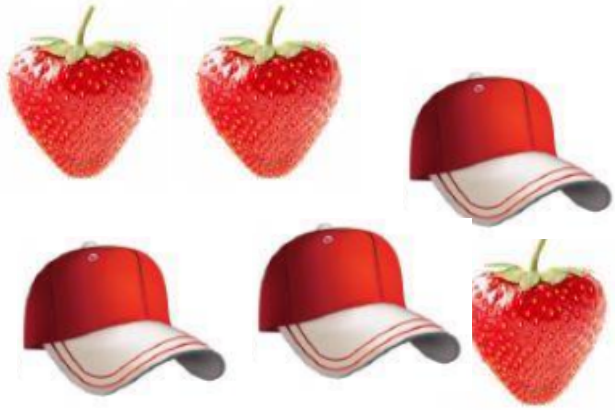
Python ile Yapay Zekaya Giriş

Gözetimsiz Öğrenme



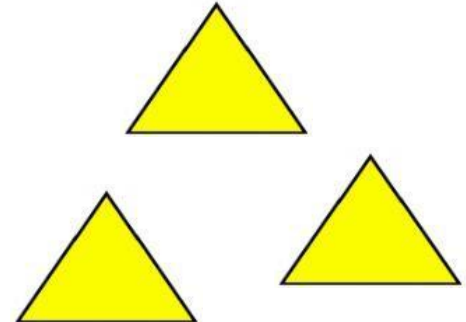
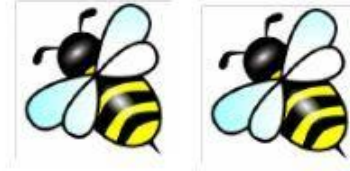
Python ile Yapay Zekaya Giriş

Gözetimsiz Öğrenme



Python ile Yapay Zekaya Giriş

Gözetimsiz Öğrenme



Python ile Yapay Zekaya Giriş

Gözetimsiz Öğrenme

Gözetimsiz öğrenme, hedef içermeyen veri üzerinde gerçekleştirilen bir makine öğrenmesi yaklaşımıdır.

Bu yaklaşımda, veri kümesindeki yapıyı anlamak, gizli desenleri keşfetmek ve veri noktaları arasındaki ilişkileri belirlemek amaçlanır.

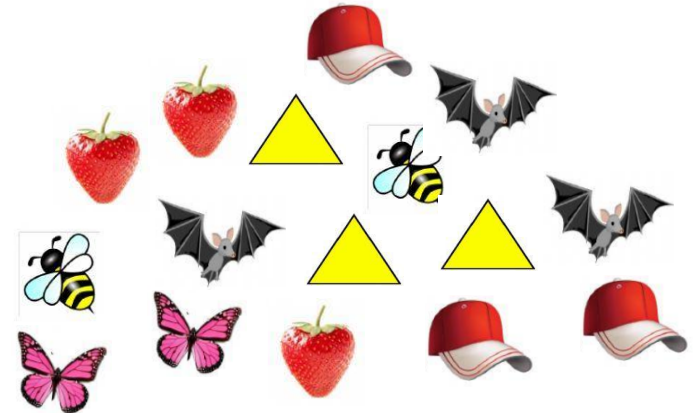


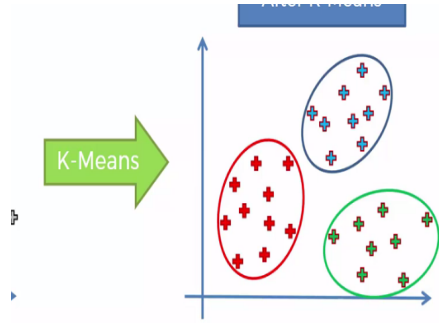
Diagram illustrating the process of data processing:

Girdi (Input) → **Algoritma** (Algorithm) → **Çıktı** (Output)

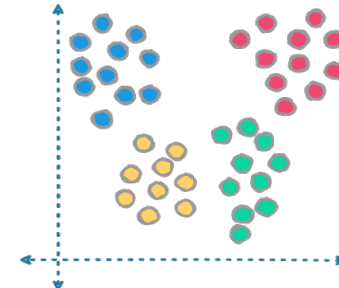
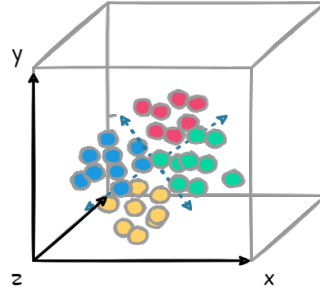
The diagram shows a sequence of three stages connected by arrows. The first stage, labeled 'Girdi', shows a large, disorganized cloud of small dots in red, green, and purple. An arrow points to the second stage, labeled 'Algoritma', which features a white robot with a blue visor sitting on a large computer mouse. Another arrow points to the third stage, labeled 'Çıktı', which displays three vertically stacked ovals. The top oval is red and contains red dots, the middle is green and contains green dots, and the bottom is purple and contains purple dots. The dots in the output ovals are arranged in a more structured, grid-like pattern compared to the input cloud.

Python ile Yapay Zekaya Giriş

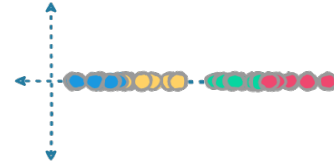
Gözetimsiz Öğrenme



Kümeleme



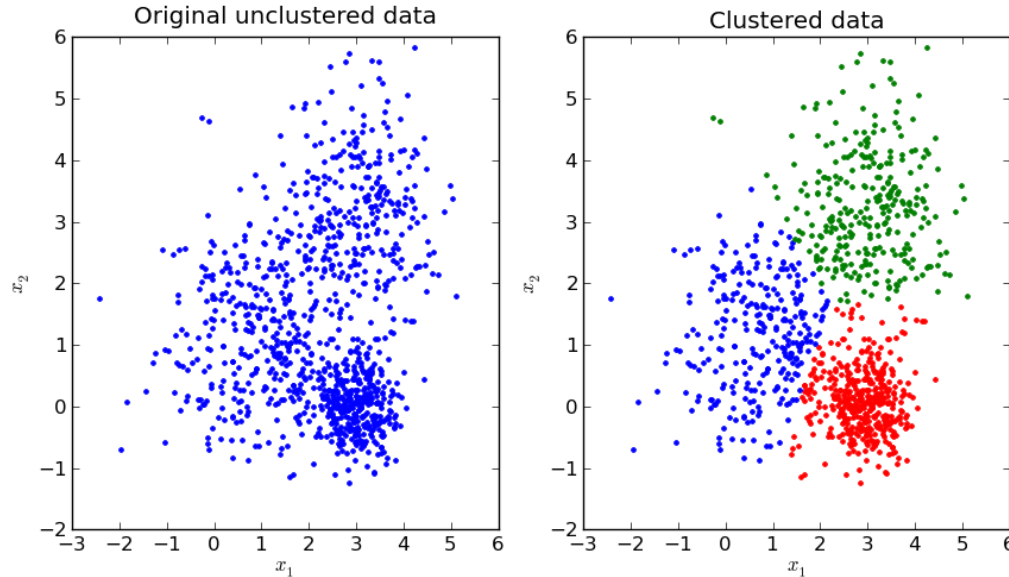
Boyut azaltma



Python ile Yapay Zekaya Giriş

Kümeleme

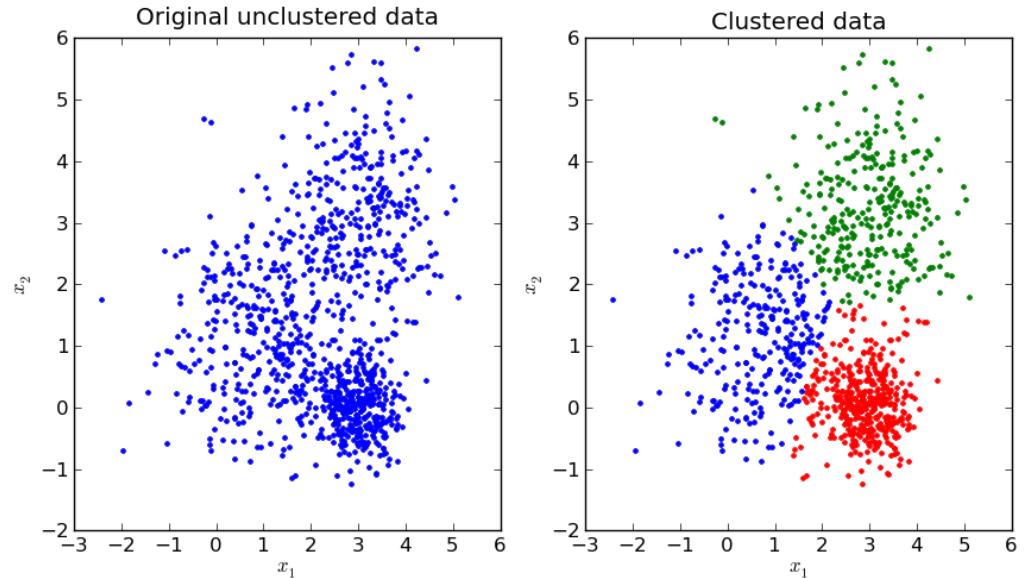
Kümeleme (Clustering), denetimsiz öğrenme (unsupervised learning) yöntemlerinden biridir ve verileri benzerliklerine göre gruplara (kümelere) ayırma işlemidir. Bu gruplar, verilerin doğal yapısına dayanarak oluşturulur ve her küme, içerisindeki öğeler arasında benzerlik, kümeler arasındaysa farklılık gösterir.



Python ile Yapay Zekaya Giriş

Kümeleme

- K-Means
- MeanShift
- DBSCAN
- Hierarchical clustering
- BIRCH





Python ile Yapay Zekaya Giriş

Kümeleme

K-means kümeleme

- Uzaklık (veriler arası) odaklıdır.

Python ile Yapay Zekaya Giriş

Kümeleme

K-means kümeleme

- Uzaklık (veriler arası) odaklıdır.

Avantajları:

- Basit ve Hızlı
- Esneklik
- Verimli

Dezavantajları:

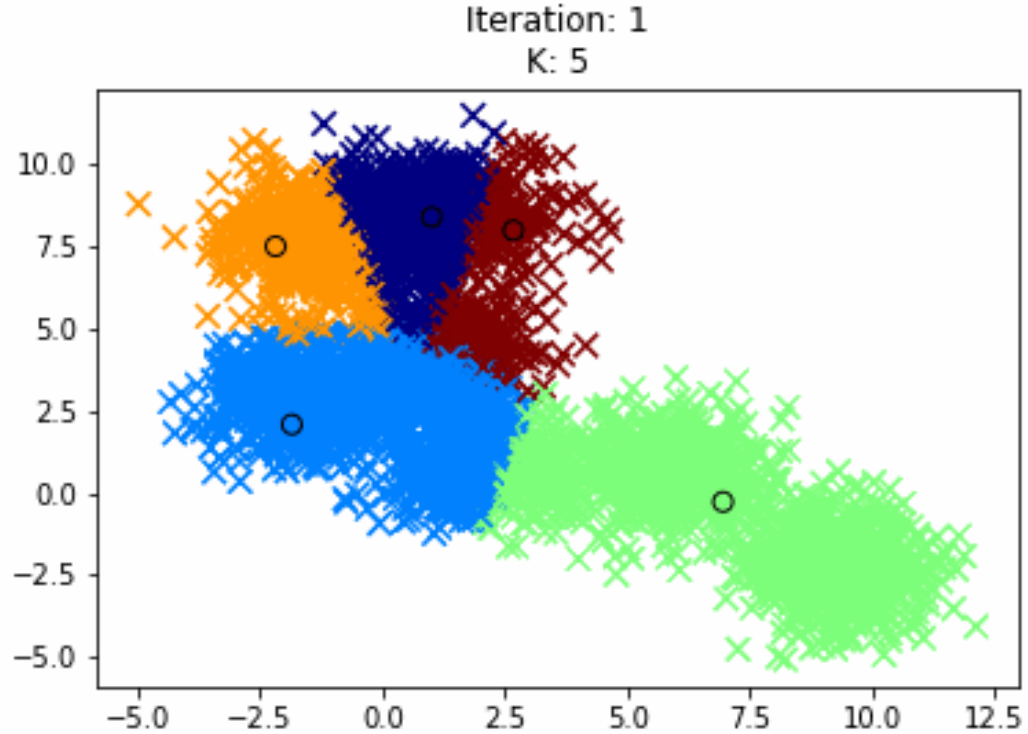
- K değeri tahmin edilmeli
- Başlangıç noktalarına duyarlı
- Yuvarlak kümeler varsayımı
- Aykırı değerlere (outliers) duyarlı

Python ile Yapay Zekaya Giriş

Kümeleme

K-means kümeleme

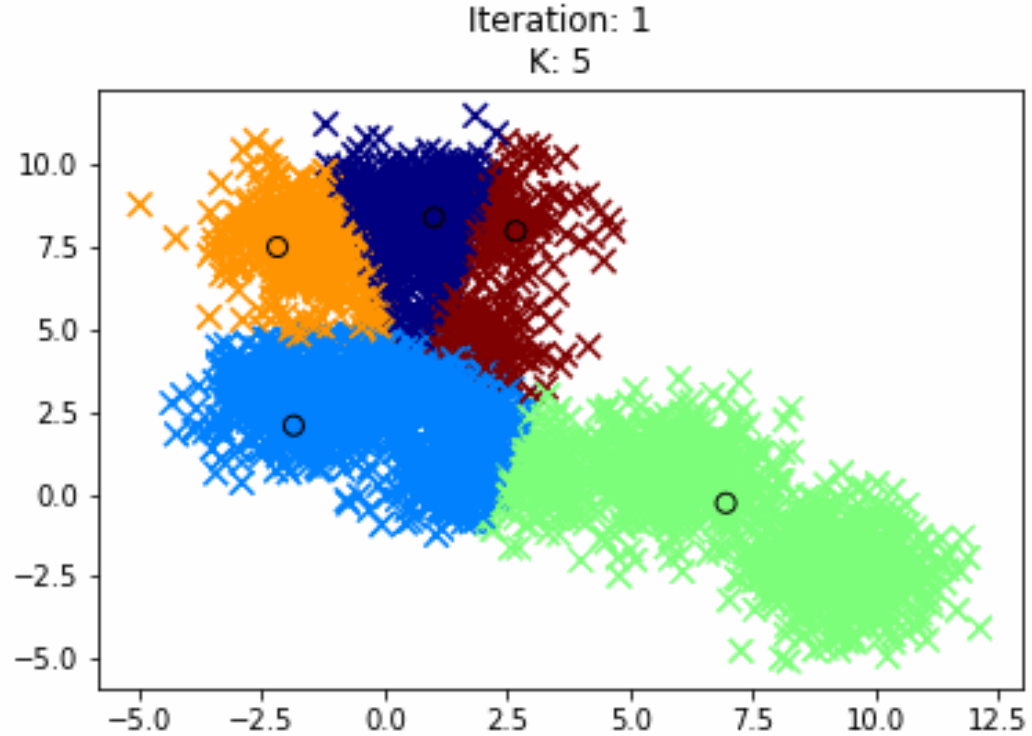
1. Başlangıç Kümeleri Seçme (K değerini belirlemek)
2. Rastgele Merkezler Seçme
3. Her Veri Noktasını En Yakın Merkeze Atama
4. Yeni Merkezleri Hesaplama
5. Adımları Tekrarlama
6. Sonuçların Değerlendirilmesi



Kümeleme

K-means kümeleme

1. Başlangıç Kümeleri Seçme (K değerini belirlemek)
2. Rastgele Merkezler Seçme
3. Her Veri Noktasını En Yakın Merkeze Atama
4. Yeni Merkezleri Hesaplama
5. Adımları Tekrarlama
6. Sonuçların Değerlendirilmesi



Python ile Yapay Zekaya Giriş

Kümeleme

1. Öklidyen Mesafe (Euclidean Distance):

Bu, K-means için en yaygın kullanılan mesafe ölçüsüdür. İki nokta arasındaki düz bir çizgi mesafesini ölçer ve genellikle "düz mesafe" olarak adlandırılır. Matematiksel formülü şu şekildedir:

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + \dots + (n_1 - n_2)^2}$$

Öklidyen mesafe, özellikle sayısal verilerle çalışırken iyi bir seçimdir ve doğrusal ilişkiler için uygundur.

Python ile Yapay Zekaya Giriş

Kümeleme

2. Manhattan Mesafesi (Manhattan Distance):

Manhattan mesafesi, iki nokta arasındaki mesafeyi, yalnızca yatay ve dikey hareketlerle ölçer. Bu, şehir planlamasında kullanılan ve "taxicab" mesafesi olarak da bilinen bir yaklaşımdır. Matematiksel formülü şu şekildedir:

$$d = |x_1 - x_2| + |y_1 - y_2| + \dots + |n_1 - n_2|$$

Manhattan mesafesi, genellikle dikey veya yatay hareketin önemli olduğu, verilerin "kare" yapılarında düzenli olduğu durumlarda daha iyi çalışır.

Python ile Yapay Zekaya Giriş

Kümeleme

3. Minkowski Mesafesi (Minkowski Distance):

Minkowski mesafesi, Öklidyen ve Manhattan mesafelerinin genelleştirilmiş halidir. Bu mesafe, bir p parametresi kullanılarak belirlenir ve aşağıdaki şekilde hesaplanır:

$$d = (|x_1 - x_2|^p + |y_1 - y_2|^p + \dots + |n_1 - n_2|^p)^{\frac{1}{p}}$$

- $p = 1$ olduğunda Manhattan mesafesi,
- $p = 2$ olduğunda Öklidyen mesafesi elde edilir.

Minkowski mesafesi, genel olarak daha esnek bir mesafe ölçüsüdür ve p değerine göre çeşitli mesafeleri kapsar.

Python ile Yapay Zekaya Giriş

Kümeleme

4. Cosine Similarity (Kosinüs Benzerliği):

Cosine similarity, özellikle metin verisi gibi yüksek boyutlu verilerle çalışırken kullanılır. Bu metrik, iki vektör arasındaki açığı ölçer ve iki vektörün yönlerinin ne kadar benzer olduğunu belirler. Matematiksel olarak, aşağıdaki şekilde hesaplanır:

$$\text{cosine similarity} = \frac{A \cdot B}{\|A\| \|B\|}$$

Bu metrik, metin madenciliği ve belgeler arasındaki benzerlikleri değerlendirmek için yaygın olarak kullanılır.

Python ile Yapay Zekaya Giriş

Kümeleme

5. Hamming Mesafesi (Hamming Distance):

Hamming mesafesi, özellikle ikili veriler veya kategorik verilerle çalışırken kullanılır. Bu metrik, iki dizideki farklı pozisyonların sayısını ölçer. Örneğin, iki dizideki aynı yerdeki değerlerin ne kadar farklı olduğunu sayar.

Python ile Yapay Zekaya Giriş

Kümeleme

6. Chebyshev Mesafesi:

Chebyshev mesafesi, her ekseninde en büyük farkı ölçer. İki nokta arasındaki mesafeyi şu şekilde hesaplar:

$$d = \max(|x_1 - x_2|, |y_1 - y_2|, \dots, |n_1 - n_2|)$$

Bu metrik, özellikle her eksenin birbirine eşit önem taşıdığı durumlarda kullanılır.



Python ile Yapay Zekaya Giriş

Kümeleme

DBSCAN (Density-Based Spatial Clustering of Applications)

- DBSCAN, yoğunluk temelli bir kümeleme algoritmasıdır. Verileri, yoğunluklarına göre kümelere ayırırken, yoğun olmayan bölgelerdeki noktaları gürültü (outliers) olarak kabul eder.

Python ile Yapay Zekaya Giriş

Kümeleme

DBSCAN (Density-Based Spatial Clustering of Applications)

- DBSCAN, yoğunluk temelli bir kümeleme algoritmasıdır. Verileri, yoğunluklarına göre kümelere ayırırken, yoğun olmayan bölgelerdeki noktaları gürültü (outliers) olarak kabul eder.

Avantajlar:

- Küme sayısını önceden belirlemeye gerek yok.
- Yoğun bölgeleri tanıyıp aykırı değerleri dışlar.
- Düz olmayan kümeleri doğru şekilde tespit eder.

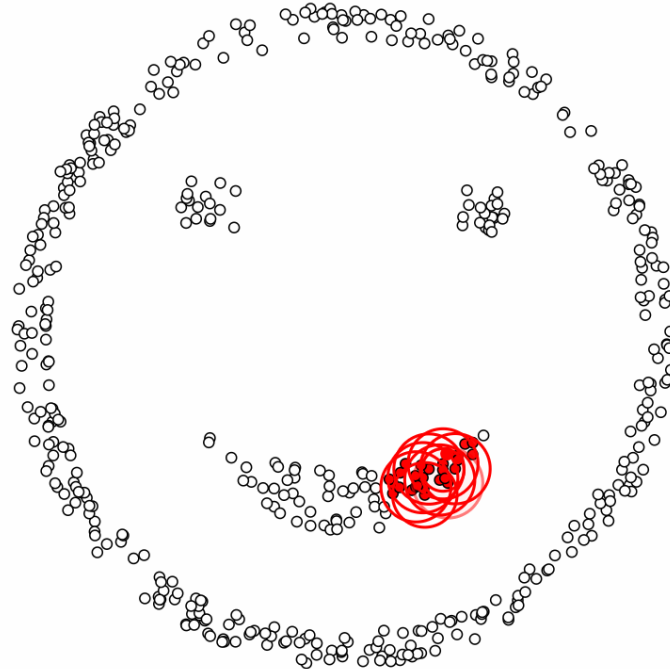
Dezavantajlar:

- Eps ve minPts parametrelerinin doğru seçilmesi gerekir.
- Yüksek boyutlu verilerde zorlanabilir.

Python ile Yapay Zekaya Giriş

Kümeleme

DBSCAN (Density-Based Spatial Clustering of Applications)



Python ile Yapay Zekaya Giriş

Kümeleme

DBSCAN (Density-Based Spatial Clustering of Applications)

