metin, ekran görüntüsü, yazı tipi, harita içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.**metin, ekran görüntüsü, harita içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.**

ekran görüntüsü, metin içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**NMI Metriği :**

K means: 0.372 hiyerarşik : 0.508 dbscan : 0.480

**Time complexity:**

k means : 0.06 agglomerative :2.6 dbscan : 0.1

Bu veri setinde 6 sınıf bulunuyor. Hiyerarşik Kümeleme, her veri noktasının en yakın uzaklıklarını karşılaştırarak çalıştığı için diğer yöntemlere göre **daha yavaş** sonuç verdi.

**K-Means** ise en hızlı algoritma oldu çünkü başlangıçta küme sayısı belliydi ve her adımda yalnızca en yakın merkezle karşılaştırma yapıldı.

**DBSCAN**, K-Means’e göre daha yavaş çalıştı ancak **daha yüksek skor** elde etti. Veri boyutu arttıkça DBSCAN’in hızı düşebilir. Ancak bu tür **karmaşık ve düzensiz** veri yapılarında DBSCAN ve Hiyerarşik Kümeleme, yoğunluk temelli yaklaşımlarından dolayı daha avantajlıdır.

metin, ekran görüntüsü, yazı tipi, harita içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**NMI Metriği :**

k means: 0.57 agglomeraitve: 0.55 dbscan: 0.41

**Zaman Karmaşıklıkları:**

k means: 0.04 hiyerarşik: 4.04 dbscan: 0.19

Bu veri yapısında 9 tane class vardır. Bu yapıda karmaşık ve daha çok düzensiz şekilli yapılar vardır. Bundan dolayı DBSCAN gibi yoğunluk temelli bir kümeleme yapısında çok iyi çalışamaz. (Bu resimdeki K-MEANS KÜMELEMEDİR.) Her ne kadar hiyerarşik kümeleme ile K-MEANS kümeleme benzer metrik değerlerini verseler de zmaan karmaşıklıklarına baktığımızda K-MEANS kümelemenin çok daha hızlı olmasından dolayı bu veri için en iyi kümeleme K-MEANS kümeleme çeşididir.

metin, harita, ekran görüntüsü, diyagram içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

8 tane class var.

**NMI Metriği:**

k means: 0.64 agglomearative: 0.65 dbscan : 0.85

**Zaman karmaşıklığı**

k means : 0.08 agglomerative: 2.39 dbscan: 0.13

Veri yapısı önceki örneklere göre daha düzenli ve toplu olduğundan, K-Means ve Hiyerarşik Kümeleme benzer skorlar üretti. Ancak Hiyerarşik Kümeleme'nin zaman karmaşıklığı daha yüksek olduğundan, algoritmanın çalışma süresi diğerlerine kıyasla daha uzun oldu.

Ayrıca, DBSCAN gibi yoğunluk tabanlı algoritmalar; belirli bir geometrik yapıya sahip olmayan ancak belirli bölgelerde yoğunlaşan veri kümeleri için daha uygundur. Bu veri setinde de DBSCAN daha başarılı sonuçlar verdi. Algoritma hızı, K-Means kadar yüksek olmasa da Hiyerarşik Kümeleme'ye göre oldukça hızlıdır. Ayrıca DBSCAN, veri içindeki gürültüleri (aykırı değerleri) başarılı şekilde ayırabilmiştir. K-Means ve Hiyerarşik Kümeleme ise bu tür gürültülere karşı daha hassastır.

ekran görüntüsü, metin, kalıp, desen, düzen içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

Burada 6 tane class var .

**NMI Metriği:**

k means: 0.68 hiyerarşik: 0.69 dbscan: 0.904

**Zaman karmaşıklığı :**

k means: 0.102 hiyerarşik: 0.0105 dbscan: 0.0071

Bu veri yapısında etrafında gürültüler olan ve dairesel şekilleri çok olan veriler bulunuyor. Bu nedenle hem dairesel şekilleri iyi ayırabilen hem de gürültüleri ayıklayan dbscan clustering neredeyse 1 e yakın bir skor verdi.

Ama zaman karmaşıklığı k meansten büyük olduğundan dolayı k meanse göre daha yavaş kaldı. Bu verilerde öncekilere göre daha az sayıda veri olduğundan dolayı algoritma hızı pek etkilemez. Ama çok daha büyük veri setlerinde k meanse göre çok daha yavaş kalabilir.

**ekran görüntüsü, metin, diyagram içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.**

**NMI Metriği:**

K means: 0.78 hiyerarşik : 0.78 dbscan: 0.90

**Zaman karmaşıklığı:**

K means: 0.032 hiyerarşik : 0.11 dbscan: 0.02

Bu veri setinde 6 tane class var. Burada veri sayısı da az olduğundan dolayı genel olarak hepsinin skor sayısı yüksek çıktı. Bundan dolayı da algoritma süreleri de küçük değerler çıktı. Özellikle sürelerde k means ve dbscan yakın süreler almıştır .

Bu veri seti için en uygunu dbscan kümeleme olmuştur. Çünkü burada daha çok yoğunluk temelli yani birbirine yakın noktalar var. Ve dbscan genel olarak daha çok dairesel şekillerde daha başarılı oluyor. Bu şekilde az verili ve belirli yerlerde nokta olan veri setleri için dbscan daha uygun olmuş.

harita, metin, ekran görüntüsü içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**NMI Metriği:**

K means: 0.95 hiyerarşik: 0.95 dbscan: 0.08

**zaman karmaşıklığı:**

K means: 0.07 hiyerarşik: 0.23 dbscan: 0.03

Burada 31 tane class var. K means ve hiyerarşik kümeleme çok yüksek bir skor değerine ulaştı. Bunun sebebi ise bu veri setinde herhangi bir şekil değil de daha çok yoğunluk üzerine olduğu içindir.

hiyerarşik ve k means uzaklıklara göre hesapladığında daha iyi bir skor verdi. Ama zaman karmaşıklığına bakıldığında ne kadar ikisi de yüksek skoru da olsa hiyerarşik kümelemenin algoritmik karmaşıklığı daha büyük olduğundan dolayı süre olarak çok farklar vardır. Bu yüzden k means clustering bu veri setine daha uygun denilebilir.

ekran görüntüsü, metin, öykü gelişim çizgisi; kumpas; grafiğini çıkarma, diyagram içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

Burada 2 tane class var.

**NMI Metriği:**

K means: 0.05 hiyerarşik: 6.412183531072217e-16 ( 0 a çok yakın.) dbscan: 1.0

**zaman karmaşıklığı :**

k means : 0.03 hiyerarşik: 0.03 dbscan: 0.01

Bu veri seti yalnızca 2 sınıfa sahip, gürültü içermiyor ve veri boyutu da küçük. Bu nedenle, **DBSCAN** algoritması epsilon (yarıçap) değeri sayesinde sınıfları düzgün şekilde ayırarak **1.0 gibi mükemmel bir skor** elde etti.

Hiyerarşik ve K-Means algoritmalarının skorları ise oldukça düşüktü. Bu sonuçlara göre, **bu tür yapıda en uygun kümeleme yöntemi DBSCAN** olmuştur.

harita, metin, ekran görüntüsü, diyagram içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**NMI Metriği:**

K means:  **0.49 hiyerarşik: 0.64 dbscan: 0.72**

**Zaman karmaşıklığı:**

**K means: 0.03 hiyerarşik: 3.06 dbscan: 0.15**

**DBSCAN'in çalışma süresi K-Means'e göre daha uzun olsa da, Hiyerarşik Kümeleme'ye kıyasla daha hızlıdır. Zaten Hiyerarşik Kümeleme algoritmasının zaman karmaşıklığı yüksek olduğu için, çalışma süresi yaklaşık 3 saniye gibi büyük bir değer almıştır.** **Bu da yapının ne kadar karışık olduğunu gösteriyor.**

**Metrik sonuçlarına bakıldığında, en yüksek skoru DBSCAN almıştır. Onu Hiyerarşik Kümeleme takip etmektedir. Ancak Hiyerarşik Kümeleme, yüksek zaman karmaşıklığı nedeniyle tercih edilmesi zor bir yöntemdir.**

**Veri kümesi genel olarak gürültü içerse de, sınıflar karmaşık ve bazı bölgelerde yoğunluk göstermektedir. Bu nedenle, DBSCAN gibi yoğunluk tabanlı bir algoritma, bu tür verilerde daha başarılı sonuçlar veriyor.**

metin, ekran görüntüsü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

Görüldüğü gibi dbscan belirli küçük kısımlarda k means ‘e göre daha iyi bir clustering yapmış.

Genel olarak K-MEANS kümeleme çok karmaşık veriler olmadığı sürece algoritmik karmaşıklığı da küçük olduğundan dolayı genelde sık sık kullanılan kümeleme çeşididir.

Hiyerarşik kümeleme k measntan farklı olarak gürültülerle daha iyi başa çıkıyor ve daha karmaşık veri yapılarında daha iyi metrik skorları veriyor. Ancak algoritmik karmaşıklığı çok daha büyük olduğundan çok daha büyük verilerle çalıştığımızda bu süre çok fazla uzayabiliyor.

DBSCAN kümeleme ise daha çok küçük veri setlerinde ve gürültüsü olmayan yada az gürültülü veri setlerinde daha başarılı çıkıyor. Daha çok dairsel şekillerde en çok işe yarayan kümeleme oldu.