Çalışma Notları: Kümeleme Analizi: Temel Kavramlar ve Yöntemler (Bölüm 10)

18 Kasım 2025

1 Bölüm 10: Kümeleme Analizi: Temel Kavramlar ve Yöntemler

1.1 Kümeleme Analizi (Cluster Analysis)

1.1.1 Kümeleme Analizi Nedir? (Sayfa 444)

Kümeleme (Clustering), nesneleri, bir küme içindeki nesnelerin birbirine benzeyeceği, ancak diğer kümelerdeki nesnelerden farklı olacağı şekilde gruplara ayıran bir **denetimsiz öğrenme** (unsupervised learning) görevidir [1-4].

- **Amaç:** Kümeleme, **sınıf etiketlerine bakılmaksızın** veri nesnelerini analiz eder ve verilerdeki doğal grupları veya **içsel olarak türetilmiş kategorileri** belirlemek için kullanılır [2, 3, 5, 6].
- **Ayırma Tipi:** Temel kümeleme yöntemleri genellikle **özel küme ayırmayı** (exclusive cluster separation) benimser; yani her nesne tam olarak bir gruba ait olmalıdır. Bu gereklilik, bulanık bölümleme (fuzzy partitioning) gibi tekniklerle esnetilebilir [7].

1.1.2 Temel Kümeleme Yöntemlerine Genel Bakış (Sayfa 448)

Temel kümeleme yöntemleri dört ana kategoriye ayrılır:

- **Bölümleme Yöntemleri (Partitioning Methods):** Veri kümesini k parçaya böler. k-means ve k-medoids gibi yöntemler, küçük-orta boyutlu veri setlerinde **küresel şekilli** kümeleri bulmak için iyi çalışır [7-9].
- **Hiyerarşik Yöntemler (Hierarchical Methods):** Nesneleri bir hiyerarşi içinde düzenler, ya aşağıdan yukarıya (**kümelemeli** / agglomerative) ya da yukarıdan aşağıya (**bölücü** / divisive) strateji kullanır [4, 10].
- **Yoğunluk Tabanlı Yöntemler (Density-Based Methods):** Küme olmayan seyrek bölgelerle ayrılmış **yoğun bölgeler** olarak kümeleri modeller. Bu, **keyfi şekilli** (arbitrary shape) kümelerin keşfedilmesini sağlar [4, 11].
- **Izgara Tabanlı Yöntemler (Grid-Based Methods):** Nesne uzayını bir ızgara yapısına ayırır. Temel avantajı, işlem süresinin genellikle **nesne sayısından bağımsız** olmasıdır, sadece ızgaradaki hücre sayısına bağlıdır [12].

1.2 Bölümleme Yöntemleri (Partitioning Methods) (Sayfa 451)

Bölümleme, nesneleri k exclusive gruba (kümelere) ayırır [13].

1.2.1 k-Means: Merkez Tabanlı Bir Teknik (Sayfa 451)

k-Means, k sayısını önceden girdi olarak alır ve küme merkezlerini hesaplamak için **ortalama** (mean) kullanır [9].

1.2.2 k-Medoids: Temsilci Nesne Tabanlı Bir Teknik (Sayfa 454)

k-Medoids, kümeyi temsil etmek için kümenin **medoid'ini** (küme içindeki en merkezi nesneyi) kullanır [9].

- **PAM (Partitioning Around Medoids):** Tüm veri setini inceleyerek en iyi k medoid'i bulmaya çalışır.
- **CLARA ve CLARANS:** Büyük veri setleri için, **rastgele örnekler** (random samples) kullanarak performans artışı sağlayan k-medoids yöntemleridir [14].

1.3 Hiyerarşik Yöntemler (Hierarchical Methods) (Sayfa 457)

Hiyerarşik yöntemler, nesneleri, ya **kümelemeli** (bottom-up) ya da **bölücü** (top-down) bir strateji kullanarak hiyerarşik bir düzende düzenler [4, 10].

1.3.1 Kümelemeliye Karşı Bölücü Hiyerarşik Kümeleme (Sayfa 459)

- **Kümelemeli (Agglomerative):** Her nesnenin ayrı bir küme olarak başladığı ve birbirine **en yakın** (en benzer) küme çiftlerinin yinelemeli olarak **birleştirildiği** yöntemdir [10].
- **Bölücü (Divisive):** Tüm nesnelerin tek bir kümede toplandığı yerden başlar ve kümeyi yinelemeli olarak **böler** (split) [10]. **DIANA** bu yönteme örnektir [10].

Bağlantı Türleri (Linkages): Küme yakınlığını ölçmek için farklı stratejiler kullanılır:

- **Tek Bağlantı (Single Linkage):** Yerel yakınlığa dayalı kümeler bulur [15].
- **Tam Bağlantı (Complete Linkage):** Küresel yakınlığı tercih eden kümeler bulma eğilimindedir [15].

1.3.2 Probabilistik Hiyerarşik Kümeleme (Sayfa 467)

Bu şemada, birleştirme işlemi **maksimum olabilirlik** (maximum likelihood) ilkesine dayanır [16]. İki küme, C_i ve C_j, ancak ve ancak aralarındaki mesafe (veya küme kalitesindeki artış) pozitif ise birleştirilir. İterasyon, aşağıdaki logaritmik değer pozitif olduğu sürece devam eder:

$$\log \frac{P(C_i \cup C_j)}{P(C_i)P(C_j)} > 0$$

Bu, küme kalitesinde bir iyileşme olduğunu gösterir [17, 18].

1.4 Yoğunluk Tabanlı Yöntemler (Density-Based Methods) (Sayfa 471)

Bu yöntemler, bölümleme ve hiyerarşik yöntemlerin aksine, keyfi şekilli (nonspherical shape) kümeleri bulmak için tasarlanmıştır [11].

1.4.1 DBSCAN (Sayfa 471)

DBSCAN (**D**ensity-**B**ased **S**patial **C**lustering of **A**pplications with **N**oise), yüksek yoğunluğa sahip bağlı bölgelere dayanır [11].

- **Parametreler:** Bir nesnenin komşuluğunu tanımlamak için ϵ (epsilon) ve yoğunluk eşiğini belirlemek için **MinPts** kullanılır [19, 20].
- **Çekirdek Nesne (Core Object):** ϵ-komşuluğu içinde en az $MinPts$ nesnesi bulunan nesnedir [20].

1.4.2 OPTICS (Sayfa 474)

OPTICS, tek bir ϵ ve $MinPts$ parametre seti kullanmanın zorluğunu aşmak için önerilmiştir [21].

- **Çıktı:** Veri kümesinin kümeleme yapısını temsil eden, bir **küme sıralaması** (cluster ordering) çıktısı verir; bu, çeşitli parametre ayarlarından elde edilen yoğunluk tabanlı kümelemeye eşdeğerdir [21, 22].
- **Mesafe Ölçüleri: Çekirdek Mesafesi** (Core-distance) ve **Erişilebilirlik Mesafesi** (Reachability-distance) kavramlarını kullanır [23].

1.4.3 DENCLUE (Sayfa 477)

DENCLUE, yoğunluğu tahmin etmek için **Gaussian çekirdeği** (kernel) gibi çekirdek fonksiyonları kullanır [24].

- **Yoğunluk Çekicileri (Density Attractors):** Tahmin edilen yoğunluk fonksiyonunun yerel maksimumlarıdır (x^*). Yalnızca yoğunluk eşiği ξ'yi aşan çekiciler dikkate alınır [24].
- **İşleyiş:** Nesneler, aşamalı bir tepe tırmanma (hill-climbing) prosedürü kullanarak yoğunluk çekicilerine atanır [24, 25].

1.5 Izgara Tabanlı Yöntemler (Grid-Based Methods) (Sayfa 479)

Izgara tabanlı yöntemler, nesne uzayını ızgara hücrelerine ayırır ve kümeleme işlemlerini bu nicelleştirilmiş uzayda gerçekleştirir [12].

1.5.1 STING (Sayfa 479)

STING (**ST**atistical **IN**formation **G**rid), **çok çözünürlüklü bir ızgara veri yapısı** kullanır [13].

- **Hiyerarşi:** Yüksek seviye hücrelerin istatistiksel parametreleri (sayım, ortalama, standart sapma, min, max, dağılım tipi) alt seviye hücrelerden kolayca hesaplanabilir [26].

1.5.2 CLIQUE (Sayfa 481)

CLIQUE (**CL**ustering **I**n **QUE**st), alt uzaylarda (subspaces) yoğunluk tabanlı kümeler bulmak için Apriori özelliğinin **monotonluğunu** kullanır [27].

- **İşleyiş:** Her boyutu bölümlere ayırarak hücreler oluşturur ve yoğun hücreleri belirlemek için bir yoğunluk eşiği kullanır. k-boyutlu bir hücre, her $(k-1)$-boyutlu izdüşümü de yoğun ise en az l noktaya sahip olabilir [27].
- **Küme Oluşturma:** Maksimal bölgeler (maximal regions) kullanılarak, birbirine bağlı yoğun hücreler birleştirilerek keyfi şekilli kümeler oluşturulur [28].

1.6 Kümeleme Değerlendirmesi (Evaluation of Clustering) (Sayfa 483)

Kümeleme değerlendirmesi, kümeleme analizinin uygulanabilirliğini ve sonuçların kalitesini ölçer.

1.6.1 Kümeleme Eğilimini Değerlendirme (Assessing Clustering Tendency) (Sayfa 484)

Amaç: Veride rastgele olmayan (non-random) bir yapının (anlamlı kümelerin) var olup olmadığını belirlemektir [29, 30].

- **Hopkins İstatistiği:** Veri kümesinin tek tip (uniform) dağılıp dağılmadığını test etmek için kullanılır [31]. Eğer $H > 0.5$ ise, veri kümesinde istatistiksel olarak anlamlı kümeler bulunması **düşüktür** [31].

1.6.2 Küme Sayısının Belirlenmesi (Determining the Number of Clusters) (Sayfa 486)

Küme sayısını belirlemek için, sınıflandırmada da kullanılan **çapraz doğrulama** (cross-validation) tekniği kullanılabilir. Farklı k değerleri için kümeleme modeli kalitesi karşılaştırılır [32].

1.6.3 Kümeleme Kalitesini Ölçme (Measuring Clustering Quality) (Sayfa 487)

Kümeleme kalitesi, İçsel (Intrinsic, sadece veri setine dayalı) veya Dışsal (Extrinsic, zemin gerçekliğine/ground truth'a dayalı) yöntemlerle ölçülür. Dışsal ölçümler için temel gereksinimler:

- **Küme Homojenliği (Cluster Homogeneity):** Aynı zemin gerçekliği kategorisine ait nesnelerin aynı kümede toplanması gerekir [33].
- **Küme Tamlığı (Cluster Completeness):** Aynı zemin gerçekliği kategorisine ait tüm nesnelerin aynı kümeye atanması gerekir [33].
- **BCubed Metrikleri:** Dışsal kümeleme değerlendirmesi için kullanılan ölçü türlerindendir (Örn: BCubed Precision) [34].

1.7 Özet (Summary) (Sayfa 490)

Kümeleme, verilerdeki latent (gizli) kategorileri bulmak için kullanılan bir denetimsiz öğrenme yöntemidir [4, 5]. Temel yöntemler arasında **bölümleme**, **hiyerarşik**, **yoğunluk tabanlı** ve **ızgara tabanlı** yöntemler bulunur [4]. Bir kümeleme yönteminin **ölçeklenebilirlik**, **keyfi küme şekilleriyle başa çıkma** ve **gürültüye karşı insansızlık** gibi gereksinimleri karşılaması beklenir [4].