KATEGORİK VERİ ANALİZİNDE KULLANILAN ALGORİTMALARIN PERFORMANSLARININ KARŞILAŞTIRILMASI ÜZERİNE BİR ÇALIŞMA

Ferhan BAŞ

YÜKSEK LİSANS TEZİ

İSTATİSTİK

GAZİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

EYLÜL 2013

ANKARA

KATEGORİK VERİ ANALİZİNDE KULLANILAN ALGORİTMALARIN PERFORMANSLARININ KARŞILAŞTIRILMASI ÜZERİNE BİR ÇALIŞMA

Ferhan BAŞ

YÜKSEK LİSANS TEZİ

İSTATİSTİK

GAZİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

EYLÜL 2013

ANKARA

Ferhan BAŞ tarafından hazırlanan "KATEGORİK VERİ ANALİZİNDE KULLANILAN ALGORİTMALARIN PERFORMANSLARININ KARŞILAŞTIRILMASI ÜZERİNE BİR ÇALIŞMA" adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylıyorum.

Prof. Dr. Semra ORAL ERBAŞ ……
Tez Danışmanı, İstatistik Anabilim Dalı

Bu çalışma, jürimiz tarafından oy birliği ile İstatistik Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiştir.

Prof. Dr. Hülya BAYRAK ……
İstatistik Anabilim Dalı, G.Ü.

Prof. Dr. Semra ORAL ERBAŞ ……
İstatistik Anabilim Dalı, G.Ü.

Doç. Dr. Sibel ATAN ……
Ekonometri Anabilim Dalı, G.Ü.

Tez Savunma Tarihi: 19/09/2013

Bu tez ile G.Ü. Fen Bilimleri Enstitüsü Yönetim Kurulu Yüksek Lisans derecesini onamıştır.

Prof. Dr. Şeref Sağıroğlu ……...
Fen Bilimleri Enstitüsü Müdürü

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

Ferhan BAŞ

KATEGORİK VERİ ANALİZİNDE KULLANILAN ALGORİTMALARIN PERFORMANSLARININ KARŞILAŞTIRILMASI ÜZERİNE BİR ÇALIŞMA

(Yüksek Lisans Tezi)

Ferhan BAŞ

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
Eylül 2013

ÖZET

Kümeleme analizi nesnelerin doğal gruplarını bulmak için kullanılan bir yöntemdir. Kümeleme yapılırken küme içi homojenlik ile kümeler arası heterojenliğin yüksek olması istenir. Sayısal verilerle kümeleme yapmak oldukça kolaydır ve sayısal verileri kümeleyen birçok yöntem vardır. Fakat kategorik veriler ile çalışmak sayısal verilerde olduğu kadar kolay değildir. Kategorik verileri kümelemek için çok fazla yöntem yoktur ve var olanların hangisinin en iyi olduğu ile ilgili kesin bir bilgi bulunmamaktadır. Veri sayısına ve veri yapısına göre her bir yöntemin birbirine üstünlükleri ve eksiklikleri vardır. Ayrıca iyi bir kümeleme yapmak için kullanılacak değişken sayısı büyük önem taşımaktadır. Bu çalışmada kategorik verilerin kümelenmesi ile ilgilenildi. Hiyerarşik kümeleme tekniklerinden tek bağlantı tekniği, tam bağlantı tekniği, ortalama bağlantı tekniği ve bölmeli kümeleme tekniklerinden K-modes algoritması kullanılarak kümeleme analizi yapıldı ve sonuçlar karşılaştırıldı. Nitelikli bir karşılaştırma yapmak için literatürde bu tür karşılaştırmaların yapılmasında yaygın olarak kullanılan gerçek veri setlerinden yararlanıldı. Analiz sonuçlarına göre veri sayısı büyüdükçe kümeleme performansı hiyerarşik tekniklerde azalırken K-modes algoritmasında arttığı tespit edildi.

Bilim Kodu : **205.1.066**

Anahtar Kelimeler : **Kümeleme analizi, kategorik veri**

Sayfa Adedi : **51**

Tez Yöneticisi : **Prof. Dr. Semra ORAL ERBAŞ**

A STUDY COMPARİNG PERFORMANCES USED ALGORİTHMS İN CATEGORİCAL DATA ANALYSİS

(M. Sc. Thesis)

Ferhan BAŞ

GAZİ UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

September 2013

ABSTRACT

Cluster analysis is a method used to find natural groups of objects. Given a data set the main goal is to produce a partition with high internal intra-cluster similarity and high inter-cluster dissimilaity. Clustering with numerical data is quite easy and there are many methods for them. But clustering of categorical data is more difficult than clustering of numerical data. There is not many methods for clustering of categorical data and there is no certain information about which one is best. According to the number of data and data sutructure each has advantages and limitations. Also variable number is important for good clustering results. In this thesis dealt with clustering of categorical data. Hierarchical clustering techniques which are single linkage, complete linkage, average linkage and partitional clustering technique which is K-modes algorithm were compared. Well known real data sets were used for quality comparison. According to the analysis results when the number of data set grows clustering performances are decreasing in single linkage, complete linkage, average linkage while K-modes algoritm's is increasing.

Science Code : **205.1.066**
Key words : **Cluster analysis, categorical data**
Page Number : **51**
Supervisor : **Prof. Dr. Semra ORAL ERBAŞ**

TEŞEKKÜR

Çalışmalarım boyunca değerli yardım ve katkılarıyla beni yönlendiren hocam Prof. Dr. Semra ORAL ERBAŞ'a, uzun ve yorucu çalışma dönemlerimde maddi ve manevi destekleriyle beni hiç yalnız bırakmayan sevgili aileme ve dostum Çiğdem ÇAKIR'a en içten dileklerimle teşekkürlerimi sunarım.

İÇİNDEKİLER

Sayfa

Sayfa

ÇİZELGELERİN LİSTESİ

Çizelge **Sayfa**

Çizelge **Sayfa**

ŞEKİLLERİN LİSTESİ

Şekil **Sayfa**

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılan bazı simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	**Açıklama**
k	Küme sayısı
n	Örnek hacmi
p	Değişken sayısı

Kısaltmalar	**Açıklama**
CACTUS	Özetleri kullanarak kategorik kümeleme (CAtegorical Clustering Using Summaries)
QNNS	Nitelikli en yakın komşulukların seçimi (Qualified Nearest Neighbors Selection)
QROCK	Bağlantıları kullanarak hızlı sağlam kümeleme (Quick RObust Clustering using linKs)
ROCK	Bağlantıları kullanarak sağlam kümeleme (RObust Clustering using linKs)

1.GİRİŞ

Kümeleme analizi, büyük boyuttaki verileri, aynı gruptaki veriler birbirine en çok benzerlikte farklı gruplardakiler en az benzerlikte olacak şekilde gruplara bölen bir yöntemdir. Kümeleme analizinin temel amacı nesnelerin doğal gruplarını bulmaktır[1].

Kümeleme analizi ilk kez John Snow tarafından kullanılmıştır. 1854 yılında Londra da kolera salgını olmuş ve John Snow bu hastalığın görüldüğü yerleri tespit ederek özel bir harita oluşturmuştur. Harita oluşturulduktan sonra hastalığın belli bir sokakta yoğunlaştığını gözlemlemiştir. Bundan dolayı o bölgedeki kuyu pompaları salgına bir son vermek için kaldırılmıştır. Bu olay kümeleme analizinin en basit ve bilinen ilk uygulamasıdır [2]. O zamandan beri kümeleme analizi büyük miktardaki verilerin doğal gruplarını bulmak için istatistik, yazılım mühendisliği, biyoloji, psikoloji ve diğer sosyal bilimlerde geniş çapta kullanıldı. Üretim, radar tarama, tıp, nükleer bilim ve araştırma sektörü gibi alanlarda çok sık kullanılan kümeleme analizine örnek vermek gerekirse, Wu ve arkadaşları 2002 yılında genlerin karmaşıklığını çözmek için özel bir algoritma geliştirmişlerdir [3]. Jiang ve arkadaşları da 2004 yılında genlerin tanımlanması için birçok kümeleme tekniği ile analizler yapmışlardır [4]. May ve arkadaşları 2006 yılında nükleer tıp görüntüleme yönteminde pozitron emisyon tomografisi olarak bilinen dokuları kümelere bölmek için bu analizlerden yararlanmışlardır [5]. Son olarak Mathieu ve Gibson 2004 yılında kümeleme analizini geniş çaplı araştırmalar için karar destek araçlarının bir parçası olarak kullanmışlardır [6].

Fraley ve Raftery 1998 yılında ilk kez kümeleme metotlarını hiyerarşik ve bölmeli metotlar diye iki ayrı gruba ayırmıştır [7]. Han ve Kamber ise 2001 yılında bunlara ek olarak 3 ayrı kategori daha ilave etmiştir. Bunlar yoğunluğa dayalı metotlar (density-based methods), model tabanlı kümeleme (model-based clustering) ve ızgara tabanlı metotlardır (grid-based methods) [8].

Geleneksel kümeleme algoritmaları çoğunlukla sayısal verilerle ilgilenmektedir. Çünkü hesaplamaların yapılması ve benzerliklerin bulunması sayısal verilerle daha kolaydır. Fakat gerçek hayatta birçok veri kategoriktir ve kategorik veriler ile yapılan işlemler daha karmaşıktır. Kategorik verilerde değişkenler birden fazla değere sahip olduğu için benzerlik, ortak nesneler ve ortak değerlerin ilişkisi olarak tanımlanmaktadır. Kategorik verileri kümelemek için birçok algoritma önerilmiştir fakat hangisinin daha iyi sonuçlar verdiği tam olarak belli değildir. Farklı koşullara göre hepsinin birbirine üstünlükleri ve eksiklikleri bulunmaktadır [9]. Kategorik veriler için önerilen bazı kümeleme algoritmaları ROCK , CACTUS , Squeezer , K-Modes ve STIRR dır .

K-modes algoritması Huang tarafından 1998 yılında kategorik verileri kümelemek için önerilmiş K-means algoritmasının genişletilmiş bir versiyonudur. K-means kümeleme algoritması kullandığı benzerlik ölçüsünden dolayı kategorik verileri kümeleyemez. K-modes kümeleme algoritması K-means algoritması temeline dayanır fakat K-means algoritması tarafından getirilen kısıtlamalarda değişiklikler yapılarak kategorik verileri kümelemeye uygun hale getirilmiştir. Bu değişiklikler, kategorik veriler için basit eşleşme katsayısı (Hamming uzaklığı) kullanmak, kümelerin ortalamaları yerine modlarını kullanarak yerlerini değiştirmektir [10].

Gibson ve arkadaşları 1998 yılında STIRR adlı bir algoritma önermiştir. Bu algoritma kategorik veriler için genelleştirilmiş bir spektral grafik bölümleme metodudur. STIRR doğrusal olmayan dinamik sistemler için kategorik veri haritası olan tekrarlı bir metottur. Şayet dinamik sistem yakınsarsa kategorik veri kümelenebilir. Kümelerin doğal kombinasyonunun elde edilmesi oldukça kolaydır, fakat STIRR değişken değerleriyle ilişkili yakınlığı tanımlamak için bir ön hazırlık adımı gerektirir. Ek olarak STIRR tarafından kümelerin kesin sınıfları keşfedilemez [11].

Ganti ve arkadaşları tarafından 1999 yılında CACTUS adı verilen verileri özetlemeye dayanan bir algoritma önerilmiştir. CACTUS algoritması bütün

değişkenlerin alt kümelerini bulur ve böylece verinin alt uzay kümelemesini gerçekleştirir [12].

Guha ve arkadaşları 1999 yılında bir hiyerarşik kümeleme metodu olan ROCK algoritmasını sunmuşlardır. ROCK algoritması iki nesne arasındaki ortak komşulukların sayısını hesaplarken bağlantıları kullanmıştır. Bu algoritmada ilk olarak her bir nesne farklı bir küme olarak atanır. Daha sonra kümeler, kümeler arası yakınlığa göre birleştirilir bu yakınlıkta bütün nesne çiftleri arasındaki bağlantıların sayısının toplamı olarak ifade edilir [13].

He ve arkadaşları 2004 yılında Squeezer adında bir algoritma önermiştir. Squeezer ilk demeti küme içine koyar ve daha sonraki demetler benzerlik fonksiyonuna dayalı yeni oluşturulmuş bir kümenin benzerliğine göre o kümeye konulur veya reddedilir [14].

Bütün algoritmaların ortak varsayımı her bir nesne sadece bir küme içinde sınıflanabilir ve bütün nesneler bir küme içinde gruplandığında aynı derecede öneme sahiptir. Fakat gerçek hayattaki uygulamalarda kümeler arasında kesin sınırlar çizmek zordur. Bu yüzden bir kümeye ait olmak için nesnelerin belirsizliğine dikkat etmek gereklidir [9].

Çalışmanın ikinci bölümünde kümeleme analizinde kullanılan teknikler tanıtıldı. Üçüncü bölümde gerçek veri setleri ile çalışılarak tek bağlantı, tam bağlantı, ortalama bağlantı teknikleri ve K-modes algoritmasının kümeleme performansları değerlendirildi. Son olarak dördüncü bölümde analiz sonuçları yorumlandı ve önerilerde bulunuldu.

2. KÜMELEME ANALİZİ

Günümüzde artan bilgi ve buna bağlı olarak artan verilerle sağlıklı bilgiler elde edebilmek için bu verilerden özet bilgi edinme ve sayısını indirgeme ihtiyacı duyulur. Veri setleri büyüdükçe verilerin boyutları, analizleri zorlaştırmakta ve sonuçların doğruluğunu engellemektedir. Böyle büyük veri setlerini kümelemek ise yapacağımız istatistiksel analizlerde bize kolaylık sağlamaktadır [15]. Kümeleme analizinin amacı gruplanmamış verileri benzerliklerine göre sınıflayarak araştırmacıya özet bilgi sağlamak ve çok fazla olan veri sayısını gruplayarak daha az sayıya indirgemektir.

Kümeleme analizi birbirine benzer nesneleri aynı grupta toplarken benzemeyenleri farklı gruplarda toplayan kullanışlı bir çok değişkenli analiz tekniğidir. Kümeleme analizinde küme içi benzerliğin çok yüksekken kümeler arası benzerliğin düşük olması istenir. Böylece gruplar anlamlı bir şekilde sınıflanmış ve küme içindeki nesneler kümeyi iyi temsil etmiş olur [13].

Kümeleme analizinin uygulanabilmesi için verilerin normal dağılma varsayımı olsa da bu varsayım daha çok teoride kalmakta ve uygulamada kullanılmamaktadır. Uzaklık değerlerinin normal dağılıma uygunluğuna bakılmakta ve varsayımın sağlanması durumunda Kovaryans matrisi için farklı bir varsayım gerekmemektedir [16].

Kümeleme analizi veri setindeki nesneler arasında ilişki ve anlamlı şekiller bulmaya odaklandığı için bir veri modelleme problemi olarak düşünülebilir. Kümeleme analizi ayrıca gruplanacak veriler ile ilgili önceden tanımlanan sınıflarla ve veri setiyle ilgili başlangıç bilgisine sahip olmadığı için denetimsiz öğrenme yöntemidir [17].

Kümeleme analizinde verilerin sayısına bağlı olarak algoritma tarafından harcanan hesaplama zamanına zaman karmaşıklığı adı verilir ve büyük “O” harfi ile gösterilir. Bu şekilde ifade edildiğinde, zaman karmaşıklığının asimptotik olarak tanımlandığı söylenebilir. Örneğin, “n” büyüklüğünde veri setine sahip olduğumuzda algoritma

tarafından gereken zaman en çok $5n^3+3n$ ise asimptotik zaman karmaşıklığı $O(n^3)$ olur. Zaman karmaşıklığı genel olarak algoritmanın uygulandığı temel işlemlerin sayısıyla tahmin edilir [18].

Kümeleme analizi; istatistik, yazılım mühendisliği, biyoloji, tıp, psikoloji ve diğer sosyal bilimlerde büyük veri setlerinin doğal gruplarını belirlemek için yaygın bir şekilde kullanılmaktadır [2]. Kümeleme analizinin çok geniş bir uygulama alanı vardır. Örneğin; bir sigorta şirketi kümelemeyi kazançlarına, yaşlarına, satın aldıkları poliçe tiplerine göre benzer müşteri gruplarını bulmak için kullanabilir [17]. Biyologlar kümelemeyi hayvanları ve bitkileri karakteristik özelliklerine veya genlerinin işlevsel fonksiyonlarına göre sınıflamak için kullanabilir [19]. Tıpta kümeleme hastalıklar ve hastalıkların semptomlarının kümelenmesi için kullanılabilir [20]. Web analizi, bilgi erişimi, veri madenciliği, konuşma ve karakter tespiti yapma kümeleme analizinin bazı diğer örnekleridir.

Değişkenler sayısal (nicel) ve kategorik (nitel) olmak üzere ikiye ayrılır. Sayısal değişkenlerle kümeleme yapabilmek için genellikle uzaklık ölçümleri kullanılır. Bunlardan en sık kullanılan Öklid uzaklık ölçüsü ve Manhattan City Block uzaklık ölçüsüdür. Kategorik verilerde kümeleme yapabilmek ise sayısal verilerde olduğu kadar kolay değildir. Sayısal veriler üzerinde yapabildiğimiz toplama, bölme, ortalama alma gibi işlemleri kategorik veriler üzerinde yapamayız. Dolayısıyla kategorik verilerde kümeleme yapabilmek için uzaklık ölçümlerini kullanmak uygun değildir. Bunun yerine daha farklı benzerlik kriterleri kullanmak gereklidir. Bu çalışmada kategorik verilerin kümelenmesi ile ilgilenildi. Kategorik verileri kümelemek için birçok algoritma parametre değerlerinin dikkatli bir şekilde seçimini gerektirir [21]. Kategorik verileri kümelemek için geliştirilen algoritmalar üzerinden en uygun kümeleme yöntemi belirlenmeye çalışıldı.

2.1. Kümeleme Analizi Teknikleri

Bu bölümde en iyi bilinen kümeleme algoritmaları tanıtılacaktır. Birçok kümeleme metodu olmasının temel sebebi “küme” ifadesinin tam olarak tanımlanmamasıyla

ilgilidir. Dolayısıyla her biri farklı prensiplerde kullanılan birçok kümeleme metodu geliştirilmiştir [22]. Yukarıda da verildiği gibi Fraley ve Raftery kümeleme metotlarını hiyerarşik ve bölmeli metotlar diye iki ayrı gruba ayırmıştır. Han ve Kamber ise bunlara ek olarak 3 ayrı kategoriye daha ayırmıştır. Yoğunluğa dayalı metotlar (density-based methods), model tabanlı kümeleme (model-based clustering) ve ızgara tabanlı metotlar (grid-based methods).

2.1.1. Hiyerarşik teknikler

Hiyerarşik algoritmalar nesnelerin hiyerarşik olarak ayrışmasıyla elde edilir [2]. Kümeler dendrogram yardımıyla görülebilecek şekilde birleştirilir. En çok kullanılan hiyerarşik kümeleme algoritmaları CURE, BIRCH ve ROCK' dır [23, 24].

Hiyerarşik algoritmalar toplamalı (agglomerative) ve bölücü (divisive) olmak üzere ikiye ayrılır.

Bölücü hiyerarşik kümeleme

Bölücü hiyerarşik kümeleme toplamalının tam tersi bir süreç gösterir. Başlangıçta bütün nesneler tek bir küme içerisindeyken her adımda kendi içinde başarılı bir şekilde ayrılmış alt kümelere bölünür. Bu süreç istenen küme yapısı elde edilene kadar devam eder.

Toplamalı hiyerarşik kümeleme

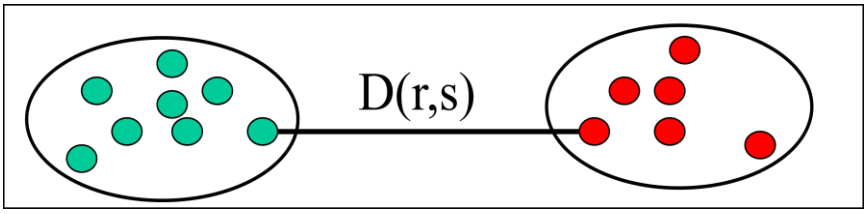
Başlangıçta her bir nesne kendi içinde bir kümeyi temsil eder. Her adımda birbirine en benzer kümeler birleştirilir ve yeni bir küme elde edilir. Bu süreç istenen küme sayısı veya belirli koşullar elde edilene kadar devam eder. Genellikle final kümelerinin sayısı başlangıçta kullanıcı tarafından bilinmediği için kümelerin belirtilen bir sayısı yerine uzaklık veya benzerlik için bir eşik değeri kullanılabilir. Bu kullanıcı tanımlı eşik değeri algoritmayı sonlandırmak için kullanılacaktır. Şayet bütün küme çiftleri arasındaki uzaklık / benzerlik eşik değerinden daha büyük / daha

küçük ise kümeleme süreci sonlandırılır aksi taktirde kümeleri birleştirme süreci devam eder [17].

Bazı hiyerarşik kümeleme teknikleri aşağıda verilmiştir.

En yakın komşuluk tekniği (Tek bağlantı)

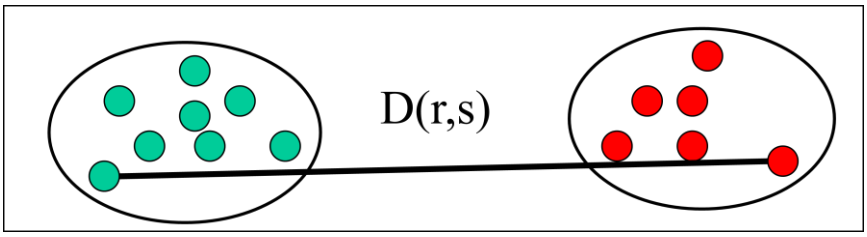
Tek bağlantı tekniğinde iki küme arasındaki uzaklık mümkün tüm kümeler arasındaki en küçük uzaklık ile başlar. Yani ilk önce en yakın iki küme birleştirilir. D(r,s): r ve s kümeleri arası uzaklık, en yakın iki nesne çifti arasındaki uzaklık olarak tanımlanır.



Şekil 2.1. Tek bağlantı tekniği

En uzak komşuluk tekniği (Tam Bağlantı)

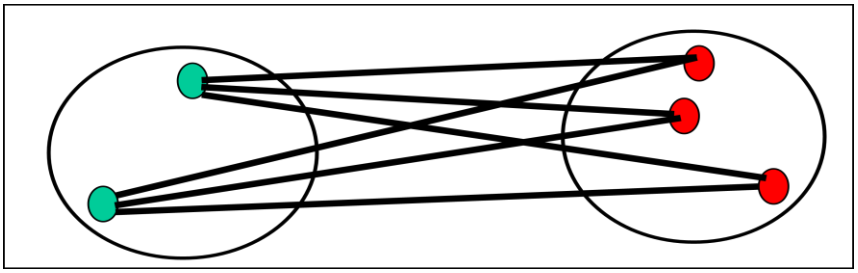
Tam bağlantı tekniğinde iki küme arasındaki uzaklık mümkün tüm kümeler arasındaki en büyük uzaklık ile başlar. Yani ilk önce en uzak iki küme birleştirilir. D(r,s): r ve s kümeleri arası uzaklık, en uzak iki nesne çifti arasındaki uzaklık olarak tanımlanır.



Şekil 2. 2. Tam bağlantı tekniği

Ortalama bağlantı tekniği

Uzaklık r ve s nesnelerinin bütün çiftleri arasındaki uzaklığın ortalaması olarak tanımlanır. Burada r ve s farklı kümelere aittir.



Şekil 2.3. Ortalama bağlantı tekniği

Ward tekniği

Ward, 1963 yılında kısmi problemlerde yardımcı olacak genel bir hiyerarşik kümeleme tekniği oluşturmaya çalışmıştır. Bu teknik, minimum varyans metodu olarak da adlandırılır. Teknik, kümeler içi kareler toplamı minimum olan (grup içi varyans minimum) iki kümeyi birleştirmeye çalışmaktadır. Kısaca teknik varyansları esas alarak birleştirme işlemini yapar ve birleştirme işlemine değişkenliği en az olan kümeler ile başlar [16] .

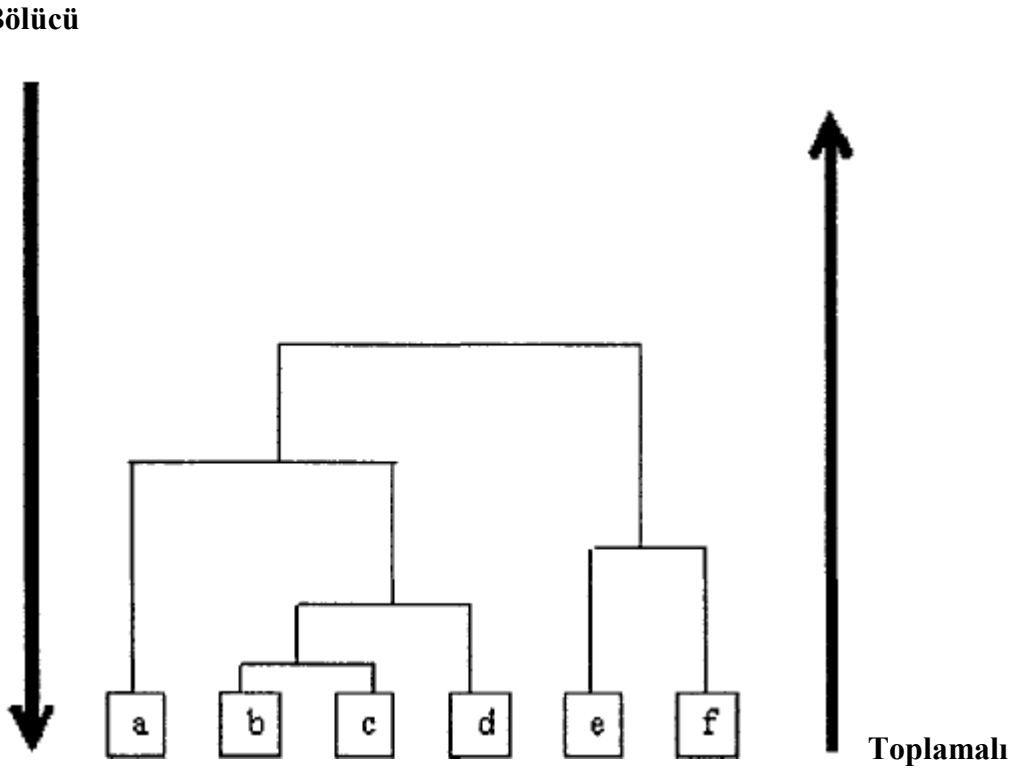
Merkezi bağlantı tekniği

Merkezi bağlantı yönteminde iki küme arasındaki uzaklık kümelerin kendi merkezleri arasındaki uzaklık olarak alınır. Kısaca küme merkezi olarak küme ortalamalarının ağırlıklı ortalaması alınmaktadır [16].

Medyan bağlantı tekniği

Merkezi bağlantı yönteminden farklı olarak, medyan bağlantı yönteminde iki küme arasındaki uzaklık, iki kümenin merkezleri arasındaki uzaklığın eşit ağırlıklı olarak hesaplanmasıyla elde edilir [25] .

Şekil 2.4. toplamalı ve bölücü hiyerarşik algoritmaları göstermektedir.



Şekil 2.4. Hiyerarşik kümeleme algoritması

Hiyerarşik kümeleme algoritmalarının dezavantajları:

- Bu algoritma benzerlik / komşuluk matrisini hafızada muhafaza etmektedir. Bu durum algoritmayı büyük veri setleri için ölçeklenemez yapmaktadır. Bu algoritmalar veri noktalarının yerine benzerlik matrisi olduğunda daha uygundur.
- Bu algoritmaların zaman karmaşıklığı veri setindeki nesnelerin sayısı "n" olduğunda en az $O(n^2)$ dır.

- Zaman karmaşıklığı yüksek olduğu için örnekleme, sonuçların niteliğini etkilerken karmaşıklığı azaltmak için kullanılabilir.
- Bu algoritma kümeleme sonuçlarını asla bir önceki düzeye geri almaz. Bu yüzden şayet bir nesne yanlış bir kümeye atanırsa doğru kümeye yeniden atanması mümkün değildir [17].

2.1.2. Bölmeli kümeleme tekniği

Bölmeli temelli algoritmalar veri setini kullanıcı tarafından belirlenmiş küme sayısına böler. Nesnel bir kriter fonksiyonunun optimize edilmesiyle oluşan kümelerin algoritması olan bölmeli temelli algoritma küme içindeki nesneler arası benzerliği maksimize eder veya uzaklığı minimize eder [17]. K-means algoritması iyi bilinen bölmeli temelli algoritmalara örnek olarak verilebilir [26]. Genellikle bölmeli temelli kümeleme algoritmaları 3 adımdan oluşur; başlatma, bölümleme ve düzeltme [27].

Başlatma aşamasında "k" nesne kümeyi temsil etmek için seçilir. Sonra kalan nesneler optimize edilmiş bir kriter fonksiyonuna göre bu "k" küme içine atanır. Bir nesne kümeye atandığı zaman o kümeyi temsil eden nesne güncellenmelidir. İlk "k" nesnenin seçimi birçok yolla yapılabilir; örneğin, rastgele olarak seçilebilir. Başlangıçtaki bölümlemeden sonra veri seti tekrar taranır ve nesneler kriter fonksiyonuna göre en iyi kümeye yeniden atanır. Düzeltme aşaması bütün nesneler tekrar atanana kadar devam eder.

Bölmeli temelli kümeleme algoritmalarının bazı dezavantajları:

- "k" küme sayısını seçmek problem yaratmaktadır. Uygun bir "k" sayısını seçmek için etkili bir yol yoktur.
- Algoritmanın uygunluğu başlangıç kısmına yani "k" başlangıç nesnesinin seçimine bağlıdır.
- Algoritma aykırı gözlemlere duyarlı olabilir. Hatta diğer bütün nesnelere çok uzak bir nesne kümelere atanmayı zorlaştırabilir [17].

2.1.3. Yoğunluğa dayalı kümeleme teknikleri

Yoğunluk tabanlı algoritmalar nesneleri belirli bir nesnel yoğunluk fonksiyonuna göre gruplar. Yoğunluk genellikle bir veri nesnesinin belirli bir komşuluğundaki nesnelerin sayısı olarak tanımlanır. Bu yaklaşımda, verilen bir küme komşuluktaki nesnelerin sayısı eşik değerini aşıncaya kadar büyümeye devam eder [2]. En iyi bilinenleri; DBSCAN, OPTICS ve DENCLUE'dır [28-30].

2.1.4. Model tabanlı kümeleme teknikleri

Bu algoritmalar veri için matematiksel bir model dikkate alır ve model parametresinin değerini veriye en uygun şekilde seçer [2]. Model tabanlı kümeleme ayrıca her bir kümenin bir kavram veya sınıf temsil edecek şekilde karakteristik özelliklerini bulur. En çok kullanılan metotları karar ağaçları ve yapay sinir ağlarıdır [31]. Bu algoritmalara EM algoritması örnek verilebilir [32].

2.1.5. Izgara tabanlı kümeleme teknikleri

Veri setini hücrelere bölerek ızgaralı bir yapı oluşmasını sağlar ve kümeleme bu ızgaralı yapı üzerinde gerçekleşir [33]. Bu algoritmaların temel odak noktası uzaysal veridir. Yani uzaydaki nesnelerin geometrik yapısı, ilişkileri, özellikleri ve işlemlerini içeren veridir. Bu algoritmaların amacı birçok hücre içindeki veri setinin sayısını belirlemek ve bu hücrelere ait nesnelerle çalışmaktır. Bu algoritma nesnelerin yerini değiştirmekten ziyade nesnelerin gruplarının birçok hiyerarşik düzeyini oluşturur. Bu bakımdan hiyerarşik algoritmalara benzer fakat ızgaraların ve dolayısıyla kümelerin birleştirilmesi bir uzaklık ölçüsüne bağlı değildir. Genellikle ızgaranın belirli bir hücresine düşen nesnelerin sayısına dayanan önceden tanımlı parametre tarafından karar verilir [2]. Bu algoritmalara örnek olarak STING, Wave Cluster ve CLIQUE verilebilir [20, 34, 35].

2.2. Kümeleme Analizinde Kullanılan Ölçüler

Kümeleme benzer nesnelerin gruplanması için iki nesnenin benzer veya benzemez olup olmadığına karar veren bazı ölçüler gerektirir. Bu ilişkiyi tahmin etmek için iki temel ölçü vardır: uzaklık ölçüleri ve benzerlik ölçüleri.

2.2.1. Uzaklık ölçüleri

Birçok kümeleme metodu herhangi iki nesne çifti arasındaki benzerlik veya benzememeyi belirlemek için uzaklık ölçülerini kullanır. x_i ve x_j olan iki nesne çifti arasındaki uzaklığı ifade etmek için $d(x_i, x_j)$ ifadesi kullanılır. İyi bir uzaklık ölçüsü simetriktir ve benzerlik durumunda minimum değer (genellikle sıfır) alır. En çok kullanılan uzaklık ölçüleri aşağıda verilmiştir.

- *Öklid Uzaklık Ölçüsü*

p-boyutlu nesne arasındaki öklid uzaklığı;

$x = (x_1, x_2, ..., x_i)$ ve

$x = (x_1, x_2, ..., x_j)$ için

$$d(x_i, x_j) = \sum_{k=1}^{p} [(x_{ik} - x_{jk})^{\lambda}]^{1/2} \qquad (2.1)$$

$$d(x_i, x_j) = \sum_{k=1}^{p} [(x_{ik} - x_{jk})^{2}]^{1/2} \qquad (\lambda = 2 \text{ durumu için}) \qquad (2.2)$$

ile ifade edilir.

- *Manhattan City Block Uzaklık Ölçüsü*

Nesneler arasında mutlak uzaklıkların toplamı alınarak hesaplanan bir uzaklık ölçüsüdür.

$$d(x_i, x_j) = \sum_{k=1}^{p} \left| x_{ik} - x_{jk} \right| \tag{2.3}$$

2.2.2. Benzerlik ölçüleri

Veri sayısal olduğu zaman kümeleme yapabilmek için genellikle uzaklık ölçüleri kullanılır. Fakat veri kategorik ise uzaklık ölçülerini kullanmak uygun olmamaktadır. Dolayısıyla kategorik verilere sahip olunduğunda daha çok benzerlik ölçüleri kullanılır.

Bir Ω uzayı üzerinden tanımlı benzerlik fonksiyonu aşağıdaki özellikleri sağlar;

1. Benzerlik: $\Omega \times \Omega \to [0,1]$. Bu fonksiyondaki 1 değeri nesne çiftleri arasındaki en yüksek benzerliği gösterirken 0 değeri nesnelerin tamamen farklı olduğunu gösterir.
2. Değişebilme özelliği: $Benzerlik(x, y) = Benzerlik(y, x)$, $\forall x, y \in \Omega$
3. $x = y$ ise $Benzerlik(x, y) = 1$'dir, yani her nesne kendisi ile ya da kendisine bire bir eş bir nesne ile en yüksek benzerliğe sahiptir [9].
4. Benzerlik fonksiyonları genellikle binary ve nominal gibi kategorik değerlerle kullanılır.

<u>İki sonuçlu (binary) değişkenlerde benzerlik ölçüleri</u>

Verideki tüm değişkenlerin sadece iki sonuçlu değerler aldığı durumlara iki sonuçlu (binary) değişken adı verilir. İki sonuçlu değişkenler içeren nesne çiftleri arasındaki benzerlik ölçüsü hesaplanırken eşleştirme tipli benzerlik ölçütleri kullanılır. Her biri p değişkenli iki nesne çifti arasındaki eşleştirme tipli benzerlik ölçüsü bulunurken çaprazlık tablosundan faydalanılır. Çaprazlık tablosu, iki sonuçlu değişkenler içeren nesne çiftlerinin karşılıklı eşleşen değişken değerlerinin sayısından oluşur. Çizelge 2.1. ile verilen çaprazlık tablosunda her bir nesne çiftinin karşılıklı değişkenlerinin aynı anda 1değerini alanlarının sayısı a, 0 değerlerini alanların sayısı d, değişkenleri

biri 1 diğeri 0 alanların sayısı b, tam tersi durumda c ile ifade edilmiştir [36]. İki sonuçlu p değişkenden oluşan veriler için birçok eşleştirme tipli benzerlik ölçüsü kullanılmakla birlikte en yaygın olarak kullanılanlar aşağıda verilmiştir. İki sonuçlu p değişkenden oluşan gözlemler için bu kadar çok eşleştirme tipli benzerlik ölçüsünün kullanılmasının nedeni 0-0 eşleştirmesinin nasıl yorumlandığı ile ilgilidir. Bazı durumlarda 0-0 eşleştirmesi, 1-1 eşleştirmesi ile eşdeğer olabilir. Örneğin cinsiyetin erkek için 1 değeri ve bayan için 0 değeri ile kodlanması veya tam tersi şeklinde kodlanması arasında bir fark yoktur. Bazı durumlarda ise 0-0 eşleştirmesi bir gözlem üzerinde herhangi bir özelliğin bulunup bulunmadığı ile ilgili olabilir. Örneğin bir bitkinin bir özelliği yenilebilir olup olmaması ise bu özelliğin olmasının 1, olmamasının 0 ile kodlanması gibi.

Çizelge 2.1. İki sonuçlu p sayıda değişken içeren bir nesne çiftinin çapraz tablosu

	i. nesne			
	Değişken Değeri	**1**	**0**	**Toplam**
j. nesne	**1**	a	b	$a+b$
	0	c	d	$c+d$
	Toplam	$a+c$	$b+d$	$p=a+b+c+d$

Binary veriler için literatürde en çok kullanılan benzerlik ölçüleri aşağıda verilmiştir.

Russel / Rau Index :

$$S_{ij} = \frac{1}{a+b+c+d} \tag{2.4}$$

Jaccard Katsayısı :

$$S_{ij} = \frac{a}{a+b+c} \tag{2.5}$$

Eşleştirme Katsayısı :

$$S_{ij} = \frac{a+d}{a+b+c+d} \tag{2.6}$$

Dice Katsayısı :

$$S_{ij} = \frac{2a}{2a+b+c} \tag{2.7}$$

Nominal değişkenlerle benzerlik ölçüleri

Kesikli değişken değerinin ikiden fazla değerler aldığı durumlara nominal değişken adı verilir. Nominal değişkenlerle kümeleme yaparken benzerlik ölçüleri kullanılır. Varsayalım ki D p-boyutlu bir veri seti ve V de bütün değişkenlerin birleşimi olsun yani, $V = D_1 \cup D_2 \cup ... \cup D_p$. $x, y \in D$ nesnelerinin herhangi bir çifti için eşleşen ve eşleşmeyen değişken değerlerinin sayısı bir ilişki tablosu kullanılarak gösterilebilir.

Çizelge 2. 2. İkiden fazla sonuçlu p sayıda değişken içeren bir nesne çiftinin çapraz tablosu

	$\alpha \in x$	$\alpha \notin x$
$\alpha \in y$	n_{11}	n_{01}
$\alpha \notin y$	n_{10}	n_{00}

Bu tabloda,

n_{11} : x ve y nesnelerindeki değişken değerlerinin sayısı,

n_{10} : Sadece x nesnesindeki değişken değerlerinin sayısı,

n_{01} : Sadece y nesnesindeki değişken değerlerinin sayısı,

n_{00} : Ne x ne de y nesnelerindeki değişken değerlerinin sayısıdır.

Eğer biz x ve y ‘yi değişken değerlerinin bir dizisi olarak düşünürsek;

$n_{11} = |x \cap y|$

$n_{01} = |x - y|$

$n_{10} = |y - x|$

$n_{00} = |V - (x \cup y)|$

olur.

İlişki tablosu kullanılarak kategorik değişken değerli nesneler için birçok benzerlik ölçüsü tanımlanabilir.

Jaccard Katsayısı :

$$Benzerlik(x, y) = \frac{n_{11}}{(n_{11} + n_{01} + n_{10})} = \frac{|x \cap y|}{|x \cup y|} \tag{2.8}$$

Basit Eşleşme Katsayısı :

$$Benzerlik(x, y) = \frac{n_{11} + n_{00}}{(n_{11} + n_{01} + n_{10} + n_{00})} = \frac{|x \cap y| + |V - (x \cup y)|}{|V|} \tag{2.9}$$

Kosinüs Katsayısı :

$$Benzerlik(x, y) = \frac{n_{11}}{\sqrt{(n_{11} + n_{01}) \times (n_{11} + n_{10})}} = \frac{|x \cap y|}{\sqrt{|x| \times |y|}} \tag{2.10}$$

Örtüşme Katsayısı :

$$Benzerlik(x, y) = \frac{|n_{11}|}{\min(n_{01}, n_{10})} = \frac{|x \cap y|}{\min(|x|, |y|)} \tag{2.11}$$

Kategorik veriler üzerine çalışan birçok algoritma benzerlik ölçüsü olarak yukarıdaki formülleri kullanır [37, 38].

İyi bir kümeleme algoritması için olması gereken bazı özellikler aşağıda verilmiştir [8]:

- ✓ *Ölçeklenebilirlik:* Kümeleme algoritmaları ana belleğe sığmayan büyük veri setleri için uygulanabilir olmalıdır. Veri setindeki nesnelerin sayısı arttığında doğrusal olarak uygulama zamanının da artması istenir.
- ✓ *Büyük Boyutlu Veri Setleriyle Çalışabilme:* Bazı algoritmalar sadece düşük (2 veya 3) boyutlu veri setleri için uygundur. Bazı veri tipleri örneğin kategorik veriler genellikle büyük boyutlu olduğu için bu verilere uygun algoritmalar geliştirmek çok önemlidir.
- ✓ *Farklı Veri Tiplerine Uygulanabilme:* Veri setleri birçok tipte olabilir. İdeal bir kümeleme tekniği sadece belirli tip veriler için değil farklı veri seti tiplerine de uygulanabilmelidir.
- ✓ *Kümeyi İsteğe Bağlı Herhangi Bir Şekilde Belirleyebilmek:* Öklid ve Manhattan uzaklık fonksiyonunu kullanan bazı algoritmalar sadece küresel kümeler üretme eğilimindedirler. Bir kümeleme algoritması isteğe bağlı şekilde küme tespit edebilmelidir. Sayısal ve uzaysal veri tipleri kullanıldığında bu özellik önemlidir.
- ✓ *Aykırı Değerler Açısından Sağlamlık:* Pratikte veri setleri aykırı gözlemler içerdiği için bir kümeleme algoritması gerçek veriden aykırı değerleri ayırabilmeli ve veri aykırı değerler içerdiğinde yüksek nitelikli sonuçlar üretebilmelidir.
- ✓ *Sıralama Bağımsızlığı:* Nesnelerin veri girişi sırası kümeleme sonuçlarını etkilememelidir.
- ✓ *Az Sayıda Parametre Bulunmalı:* Kümeleme sonuçlarının niteliği kullanılan parametrelere bağlıdır. Çok fazla parametreye sahip olmak kümelerin niteliğini ayarlamayı ve kontrol etmeyi zorlaştırır.
- ✓ *Üretilen Sonuçların Yorumlanabilirliği:* Kümeleme algoritmalarında üretilen sonuçların anlamlı ve kullanıcı tarafından yorumlanabilir olması önemlidir.

2.3. Kategorik Verilerde Kümeleme

Bu bölümde kategorik verilerde kümelemeden bahsedilecektir. Kategorik veriler de kendi içinde iki sonuçlu (binary), sınıflamalı (nominal) ve sıralı (ordinal) olmak üzere üçe ayrılır. Kategorik verilerde nesne çiftleri arasında doğal bir uzaklık ölçüsü

olmadığı için kümeleme yapmak sayısal verilere kıyasla çok daha zordur. Şayet kategorik verileri kümelemek için uzaklık ölçülerini kullanırsak doğabilecek sıkıntılar aşağıda örneklendirilmiştir.

Örnek:

1, 2, 3, 4, 5, 6 birimleri üzerinden aşağıdaki 4 işlemi içeren bir pazar sepet verisi dikkate alalım.

- {1, 2, 3, 5}
- {2, 3, 4, 5}
- {1, 4}
- {6}

İşlemler, birimlerin 1, 2, 3, 4, 5, 6 birimlerine karşılık gelip gelmemesine göre (0 \ 1) binary değişkenleri ile gösterilir. Böylece 4 nokta;

- (1, 1, 1, 0, 1, 0)
- (0, 1, 1, 1, 1, 0)
- (1, 0, 0, 1, 0, 0)
- (0, 0, 0, 0, 0, 1)

şeklinde yazılır. Noktalar \ kümeler arasındaki yakınlığı ölçmek için Öklid uzaklığı kullanıldığında, ilk iki nokta arasındaki uzaklık nokta çiftleri arasındaki en kısa uzaklık olur ve $\sqrt{2}$ dir. Sonuç olarak bunlar merkez tabanlı hiyerarşik algoritma tarafından birleştirilir. Yeni birleştirilen kümenin merkezi (0. 5, 1, 1, 0. 5, 1, 0) olur.
Bir sonraki adımda değeri $\sqrt{3}$ olan ve birinci den sonra en küçük olan üçüncü ve dördüncü noktalar birleştirilir. Fakat bu birleştirmeye karşılık gelen işlemler yani {1, 4} ve {6} hiç ortak birime sahip değildir. Bu yüzden, kümelerin merkezleri arasında uzaklık kullanılarak yapılan birleştirme işlemi farklı kümeye ait noktaları aynı kümeye atayabilir. Dolayısıyla kategorik verileri kümelerken uzaklık ölçülerini kullanmak uygun değildir.

Açık bir şekilde görüldüğü gibi kategorik verileri doğru bir şekilde kümeleyebilmek için daha farklı benzerlik ölçülerine ihtiyaç duyulur. Son zamanlarda birçok kümeleme algoritması önerilmiştir. Bunlar BIRCH, CURE, PAM, CLARANS, DBSCAN, MAFIA, CHAMELEON [39-41]. Fakat bu algoritmalar daha çok sayısal veriler için uygundur. Kategorik veriler için ise ROCK ,CACTUS, K-modes, STIRR algoritmaları önerilmiştir. Ayrıca ROCK algoritması üzerinden geliştirilen kullandıkları parametreler, benzerlik ölçüleri ve aykırı gözlemleri dikkate alıp almamasıyla farklılık gösteren bazı algoritmalar vardır. Bunlar QROCK ve QNNS'dır.

2.3.1. ROCK algoritması

Guha ve arkadaşları tarafından önerilen ROCK (RObust Clustering using linKs) algoritması benzerliği hesaplarken uzaklık ölçülerini kullanmak yerine yeni bir kavram olan bağlantıları ortaya atmıştır. Kümeleme analizi iki nesne çifti arasında uzaklık veya benzerlik kullanıldığında sadece o iki nesne arasındaki yerel bilgiyi içerirken bağlantılar kullanıldığında iki noktanın komşuluğundaki diğer noktalarla ilgili evrensel bilgiyi de içerir. Bu yüzden bağlantıları kullanarak yapılan kümeleme, kümeleme sürecine evrensel bilgiyi de kattığı için çok daha sağlam bir yöntemdir [13].

ROCK algoritması bağlantı kavramına dayanan toplamalı hiyerarşik bir kümeleme algoritmasıdır ve kümeleme yaparken aşağıdaki adımları takip eder.

Komşuluk: Komşuluk kavramı basitçe en benzer noktalar olarak ifade edilebilir. $Benzerlik(p_i, p_j)$ p_i ve p_j nokta çiftleri arasındaki yakınlığı yakalayan normalleştirilmiş bir benzerlik fonksiyonu olsun. Benzerlik 0 ve 1 arasında değerler alır ve yüksek değerli olduğunda benzerliğin yüksek olduğunu ifade eder. 0 ile 1 arasında kullanıcı tarafından bir θ eşik değeri tanımlanır ve şayet;

$$Benzerlik(p_i, p_j) \geq \theta \tag{2.12}$$

ise p_i ve p_j noktaları komşudur denir. Burada θ, komşuluğu dikkate almak için nokta çiftleri arasındaki yakınlığı kontrol eden kullanıcı tanımlı bir parametredir.
ROCK algoritması benzerlik fonksiyonunun 0 ve 1 arasındaki bir değeri normalleştirdiğini varsayar. Benzerlik fonksiyonu için Jaccard Katsayısı temelli bir benzerlik ölçüsü kullanır yani;

$$Benzerlik(T_1, T_2) = \frac{(T_1 \cap T_2)}{(T1 \cup T_2)} \tag{2.13}$$

T_1 ve T_2 işlemlerinin sahip olduğu ortak birimler ne kadar fazla ise ($T_1 \cap T_2$) o kadar büyük ve benzerlik daha fazladır.

Bağlantılar: Bağlantı(p_i, p_j) p_i ile p_j arasındaki ortak komşulukların sayısı olarak tanımlansın. Bağlantıların tanımı için şayet Bağlantı(p_i, p_j) büyük ise büyük olasılıkla p_i ile p_j aynı kümededir. Noktaları tek bir kümede birleştirme kararını vermek için bağlantıların bu özelliğinden yararlanılacaktır. Bağlantı temelli yaklaşım kümeleme problemi için evrensel bir yaklaşımdır. Bu yöntem her bir nokta çifti arasındaki ilişkide komşu veri noktalarındaki evrensel bilgiyi yakalar. Bundan dolayı ROCK kümeleme algoritması noktaları tek bir kümede birleştirme kararı verirken noktalar arasındaki bağlantıyla ilgili bilgiden yararlanır ve bu daha sağlam bir yöntemdir.

Kriter Fonksiyonu: Her kümedeki yüksek derecedeki bağlantıyla ilgilenildiği için tek bir kümeye ait p_q, p_r veri çifti için Bağlantı(p_q, p_r)' nin toplamını en büyüklemek ve aynı zamanda farklı kümelerdeki p_q, p_s için Bağlantı(p_q, p_s)'nin toplamını en küçüklemek isteriz.

Uyum Ölçüsü: Uyum ölçüsü için en yüksek değere sahip küme çifti birleştirmeye en uygun çifttir. Her adımda en yüksek uyum ölçüsüne sahip çiftler birleştirilerek kümeleme yapılır. Kümeleme sırasında kriter fonksiyonunu en büyüklemek için bu uyum ölçüsü kullanılır.

ROCK algoritması ; “k” küme sayısını, θ benzerlik eşik değerini ve dolayısıyla f(θ)’yı önceden belirlemektedir. Aykırı gözlemleri kaldırarak kümelemeyi yapmaktadır ve en kötü ihtimalle $O(n^2 + nm_m m_\alpha + n^2 \log n)$ zaman karmaşıklığına sahiptir. Burada “m_m” komşulukların en büyük sayısı, “m_α” komşulukların ortalama sayısı ve “n” veri girişi yapılan noktaların sayısıdır [13].

2.3.2. QROCK algoritması

QROCK (Quick ROCK) algoritması temel olarak ROCK algoritması ile aynı prensibe sahiptir. ROCK nesne çiftleri arasında bağlantılara dayanan ve kümeleri birleştirme sürecini bitirmek için küme çiftleri arasında hiç bağlantı kalmamasını veya istenen küme sayısının elde edilmiş olmasını isteyen toplamalı bir hiyerarşik kümeleme algoritmasıdır. QROCK algoritması ise eğer kümelerin birleştirilmesi kümeler arasında hiç bağlantı olmayana kadar devam ederse elde edilen final kümelerinin bir grafiğin bağlantılı parçalarından başka bir şey olmayacağını söylemektedir [42].

QROCK algoritması bir grafiğin bağlantılı parçalarının belirlenmesiyle kümeleri hesaplar. Böylece kümelerin elde edilmesinde ROCK algoritmasının hesaplama zamanına göre ciddi bir azalma saylayacak etkili bir metottur. Ayrıca belirli bir benzerlik eşiği bulmanın küme sayısını belirlemekten daha pratik olduğunu söyler ve sadece θ benzerlik eşiğini önceden belirler “k” küme sayısını belirlemez. QROCK algoritması aykırı gözlemleri tespit eder ve yeni bir benzerlik ölçüsü önerir.

Ağırlıklandırılmış Benzerlik Ölçüsü: Dikkat edilmelidir ki Jaccard Katsayısı her bir kategorik değişkenin önemini yakalayamamaktadır. Bütün değişkenlere aynı tutumda davranmaktadır. Fakat iyi bir benzerlik ölçüsü kategorik kümeler için her bir değişkenin ağırlıklarıyla ilişkili olarak tanımlanmalıdır. Bu benzerlik ölçüsünün arkasındaki temel düşünce eğer iki küme arasındaki farklı olan değişken iki değere sahipse bu kümeler arasındaki benzerlik başka bir iki küme arasında farklı olan ve

yüz değere sahip olan değişken olursa bu kümeler arasındaki benzerlikten farklı olması gerektiğidir. Ağırlıklandırılmış benzerlik ölçüsü aşağıda verilmiştir.

$$Benzerlik(T_i, T_j) = \frac{\left|T_i \cap T_j\right|}{\left|T_i \cap T_j\right| + 2\sum_{k \notin T_i \cap T_j} \frac{1}{\left|D_k\right|}} \tag{2.14}$$

QROCK algoritması küme olarak bileşenlerin bir dizisini belirler, komşulukları hesapladıktan sonra bağlantıların hesaplanmasına ihtiyaç duymaz. Herhangi bir "i" için i nesnesinin komşuluğundaki (x, y) nokta çiftleri için x ve y gözlemlerini içeren parçalar basitçe birleştirilir. Böylece hesaplama zamanında ciddi bir azalma olur. QROCK algoritması için zaman karmaşıklığı $O(n^2)$ olur [42].

ROCK kümeleme algoritmasının durdurulması için 2 durum önerir. Birincisi bir θ benzerlik eşiği kullanarak kümeler arasında hiç bağlantı kalmayana kadar devam etmektir. İkincisi istenen küme sayısı "k" elde edilene kadar devam etmektir. Fakat QROCK sadece isteğe bağlı bir θ önermekte küme sayısını vermemektedir. Nitekim bazı durumlarda k'yı istenen durumdan fazla seçmiş olursak sahte kümelerin oluşabileceğinden bahsetmiştir. QROCK algoritması aykırı gözlemleri de tespit etmektedir [42].

2.3.3. QNNS algoritması

QNNS (Qualified Nearest Neighbors Selection) modeli en yakın komşulukların uygun bir seçimi ile kümelerin niteliğini arttıran bir algoritmadır. ROCK algoritmasının eksikliklerini gidermek için geliştirilmiştir [43].

QNNS uygun bir şekilde komşulukların seçimini önceden belirlenmiş bir θ benzerlik eşiği olmadan otomatik olarak yapmaktadır. Ayrıca küme sayısını ifade eden "k" parametresine ihtiyaç duymamak için bir uyum ölçüsü önermekte ve bu uyum ölçüsü ile kümeleme sürecini kontrol etmektedir.

QNNS algoritmasında benzerlik eşiği θ parametresi ve küme sayısı "k" değerinin önceden belirlenmesi zor olacağı için bu parametreler olmadan otomatik olarak hesaplama yapmaktadır. Algoritmanın zaman karmaşıklığı $O(n^2 + nm_m m_\alpha + n^2 \log n)$ dır. Burada "m_m" komşulukların en büyük sayısı ve "m_α" komşulukların ortalama sayısıdır [43].

2.3.4. K-MODES algoritması

K-modes algoritması Huang tarafından kategorik verileri kümelemek için önerilmiş K-means algoritmasının genişletilmiş bir versiyonudur. K-means kümeleme algoritması kullandığı benzerlik ölçüsünden dolayı kategorik verileri kümeleyemez. K-modes kümeleme algoritması K-means algoritması temeline dayanır fakat K-means algoritması tarafından getirilen kısıtlamalarda değişiklikler yapılarak kategorik verileri kümelemeye uygun hale getirilmiştir. Bu değişiklikler;

- Kategorik veriler için basit eşleşme katsayısı veya Hamming uzaklığı kullanmak,
- Kümelerin ortalamaları yerine modlarını kullanarak yerlerini değiştirmektir [44].

Basit eşleşme benzemezlik ölçüsü aşağıdaki gibi tanımlanabilir. x ve y, F kategorik değişken tarafından tanımlanan veri nesneleri olsun. x ve y arasındaki benzemezlik ölçüsü d(x,y), iki nesnenin karşılık gelen kategorik değişkenlerinin eşleşmeyenlerinin toplam sayısıdır. Eşleşmeme ne kadar küçükse iki nesne arasındaki benzerlik o kadar fazladır. Matematiksel olarak aşağıdaki gibi ifade edilebilir.

$$d(x,y) = \sum_{j=1}^{F} \delta(x_j, y_j) \tag{2.15}$$

Burada $\delta(x_j, y_j)$, $(x_j = y_j)$ ise 0, $(x_j \neq y_j)$ ise 1 değerini alır.

Z kategorik değişkenlerden oluşan bir veri nesne seti olsun, $Z = \{Z_{1,} Z_{2,}, ..., Z_n\}$ nesnesinin modu $A_1, A_2, ..., A_F$, $Q = [q_1, q_2, ..., q_F]$ olan bir vektör,

$$D(Z,Q) = \sum_{i=1}^{n} d(Z_i, Q) \tag{2.16}$$

eşitliğini minimize eder.

Kategorik verilere sahip nesneler için yukarıdaki benzemezlik ölçüsü kullanıldığında amaç fonksiyonu;

$$C(Q) = \sum_{l=1}^{k} \sum_{i=1}^{n} \sum_{j=1}^{F} \delta(z_{ij}, q_{ij}) \tag{2.17}$$

Burada $Q_l = [q_{l1}, q_{l2}, ..., q_{lm}] \in Q$

K-modes algoritması Eş. 2.17 de tanımlanan amaç fonksiyonunu minimize eder.

K-modes algoritması aşağıdaki adımları içerir:

1. "k" başlangıç modu seçilir.
2. Eş. 2.15'e göre kimin modu kümeye en yakınsa küme içindeki nesneler yer değiştirir.
3. Her yer değiştirmeden sonra küme modları güncellenir.
4. Kümedeki nesnelerde yer değişimi bitinceye kadar 2. ve 3. adımlar tekrarlanır [44].

K-modes algoritması sadece "k" küme sayısının kullanıcı tarafından verilmesini gerektirir. Zaman karmaşıklığı O(nkL) dır. Burada "n" veri setinin büyüklüğü, "k" küme sayısı ve "L" iterasyon sayısıdır [10].

3. ALGORİTMALARIN KARŞILAŞTIRILMASI

Geleneksel hiyerarşik kümeleme tekniklerinden tek bağlantı tekniği, tam bağlantı tekniği, ortalama bağlantı tekniği ve kategorik veriler için geliştirilmiş bölmeli bir kümeleme algoritması olan K-modes algoritması kullanılarak gerçek veri setleri üzerinden kümeleme performansları değerlendirildi. Kategorik verileri kümelemede, kullanılan yöntemlerin nitelikli bir karşılaştırmasını yapmak için literatürde de en çok kullanılan gerçek veri setleri kullanıldı. Bunlar; kongre oylaması, soya fasulyesi, hayvanlar ve arabalara ait veri setleridir. Bu veri setleri Machine Learning Repository web sitesinden indirilmiştir [45]. Bu analizler MATLAB R2009'da yapılmıştır. Hiyerarşik kümeleme algoritmalarından tek bağlantı tekniği, tam bağlantı tekniği ve ortalama bağlantı tekniği için MATLAB R2009'da ki hazır fonksiyonlar kullanılmıştır ve benzerlik ölçüsü olarak Jaccard katsayısı kullanılmıştır. K-modes algoritması için de Chaturvedi ve arkadaşlarının 2001 yılında önerdiği MATLAB kodu kullanılmıştır [46].

3.1. Hastalıklı Soya Fasulyesine Ait Veri Seti

Bu veri seti soya fasulyesi rahatsızlıkları ile ilgili bilgi içermektedir. 47 tane nesneden oluşmaktadır. Bu nesneler 4 gruba ayrılmıştır: Diaporthe kök pamukçuğu (D1), kömür çürüklüğü (D2), Rhizoctonia kök çürüklüğü (D3) ve Phytopthora çürümesi (D4). Bu grupların dağılımı; D1 için 10, D2 için 10, D3 için 10 ve D4 için 17 tanedir. Her rahatsızlık 36 değişken tarafından tanımlanmaktadır ve hiç kayıp gözlem yoktur. Çizelge 3.1. hastalıklı soya fasulyesi veri setini tanımlamaktadır. Sınıf adlarını içeren son değişken analize dahil edilmeyecektir.

Çizelge 3.1. Soya fasulyesi veri setine ait değişkenler

Değişken Numarası	Değişken İsmi	Değişken Değeri
1	Tarih	Nisan, Mayıs, Haziran, Temmuz, Ağustos, Eylül, Ekim
2	Bitki standı	Normal, Normalden az

Çizelge 3.1. (Devam) Soya fasulyesi veri setine ait değişkenler

3	Yağış	Normalden az, Normal, Normalden fazla
4	Sıcaklık	Normalden az, Normal, Normalden fazla
5	Dolu yağışı	Evet, Hayır
6	Mahsul geçmişi	Geçen yıldan farklı, Geçen yılla aynı, Geçen 2 yılla aynı, Geçen birkaç yılla aynı
7	Hasar görmüş bölüm	Seyrek, Alçak alanlar, Yüksek alanlar, Bütün alan
8	Şiddeti	Küçük, Büyük, Çok büyük
9	Tohum olguları	Hiçbiri, Mantar ilacı, Diğer
10	Çimlenme	%90-100, %80-89, %80'den az
11	Bitki büyümesi	Normal, Anormal
12	Yapraklar	Normal, Anormal
13	Halka yaprak lekeleri	Yok, Sarı halkalar, Sarı olmayan halkalar
14	Marg yaprak lekeleri	w-s marg, w-s marg yok, dna
15	Yaprak lekesi büyüklüğü	1/8'den az, 1/8'den fazla, dna
16	Parça yaprak	Var, Yok
17	Malf yaprak	Var, Yok
18	Yumuşak yaprak	Yok, Üst surf, Alt surf

Çizelge 3.1. (Devam) Soya fasulyesi veri setine ait değişkenler

19	Kök	Normal, Anormal
20	Konaklama	Evet, Hayır
21	Kök pamukçuğu	Yok, Toprağın altı, Toprağın üstü, Yukarıda hasarsız
22	Pamukçuk lezyonu	Dna, Kahverengi, Kahverengi bilinmiyor, Taba rengi
23	Meyve organları	Yok, Var
24	Dış çürüme	Yok, Sağlam ve kuru, Islak
25	Miselyum	Yok, Var
26	Renk solması	Yok, Kahverengi, Siyah
27	Sclerotia	Yok, Var
28	Meyve bölmeleri	Normal, Hastalıklı, Çok az, dna
29	Meyve lekeleri	Yok, Renkli, Kahverengi lekeli, Çarpık, dna
30	Tohum	Normal, Anormal
31	Küf gelişimi	Yok, Var
32	Rengi değişik tohum	Yok, Var
33	Tohum büyüklüğü	Normal, Normalin altı
34	Buruşma	Yok, Var
35	Kökler	Normal, Çürümüş, Tümörlü
36	Sınıf adları	D1, D2, D3, D4

3.2. Hayvanlara Ait Veri Seti

Bu veri seti 101 hayvana ait bilgi içermektedir. Veri setinde 18 değişken vardır ve bunlardan birisi hayvanların isimleri, 15 tanesi iki sonuçlu (binary) değerler ve iki tanesi de sayısal değerler içermektedir. Sayısal değişkenlerden birisi hayvanların bacak sayısını vermiştir fakat bunlar kategorikmiş gibi alınıp her birine aynı önem verilecek şekilde analize dahil edilecektir. Yani 2 bacağa sahip olanlar 4 bacağa sahip olanlara daha benzer olmayacak şekilde 0,2,4,5,6,8 bacağa sahip hayvanların hepsi aynı önemde olacaktır. Diğer sayısal değere sahip olan değişken ise hayvanların ait oldukları sınıfları verdiği için analize dahil edilmeyecektir. 7 tane sınıf var ve hiç kayıp gözlem yoktur. Çizelge 3.2. değişkenleri ve değişkenlerin değerlerini tanımlamakta, Çizelge 3.3 ise hangi hayvanın hangi sınıfa ait olduğunu vermektedir.

Çizelge 3.2. Hayvanlar veri setine ait değişkenler

Değişken Numarası	**Değişken İsmi**	**Değişken Değeri**
1	Hayvan isimleri	Her biri için ayrı ayrı 101 tane
2	Saç	İki sonuçlu
3	Tüyleri	İki sonuçlu
4	Yumurta	İki sonuçlu
5	Süt	İki sonuçlu
6	Uçabilme	İki sonuçlu
7	Suda yaşama	İki sonuçlu
8	Yırtıcılık	İki sonuçlu
9	Dişli	İki sonuçlu
10	Omurgalı	İki sonuçlu
11	Soluma	İki sonuçlu
12	Zehirli	İki sonuçlu
13	Yüzgeçli	İki sonuçlu
14	Bacak	Sayısal (0,2,4,5,6,,8)

Çizelge 3.2. (Devam) Hayvanlar veri setine ait değişkenler

15	Kuyruk	İki sonuçlu
16	Evcil	İki sonuçlu
17	Kedi boyutlu	İki sonuçlu
18	Sınıfları	Sayısal (1,2,3,4,5,6,7)

Çizelge 3.3. Hayvanlara ait veri seti için sınıfların dağılımı

Sınıf Numarası	**Veri sayısı**	**Sınıf Dağılımı**
1	41	Yaban domuzu, antilop, ayı, domuz, bufalo, dana, Güney Amerika'ya özgü kobay, çita, geyik, yunus, fil, yarasa, zürafa, kız, keçi, goril, hamster, tavşan, leopar, aslan, vaşak, vizon, köstebek, firavun faresi, keseli sıçan, Afrika antilopu, ornitorenk, kokarca, midilli, domuz balığı, puma, kedi, rakın, ren geyiği, ayı balığı, deniz aslanı, sincap, vampir, tarla faresi, küçük kanguru, kurt
2	20	Tavuk, karga, güvercin, ördek, flamingo, martı, şahin, kivi kuşu, tarla kuşu, deve kuşu, muhabbet kuşu, penguen, sülün, Amerika deve kuşu, sıyırıcı, yırtıcı martı, serçe, kuğu, akbaba, çalıkuşu
3	5	Çukur engerek, deniz yılanı, kör yılan, kaplumbağa, tuatara
4	13	Levrek, sazan, yayın balığı, tatlı su kefali, kedi balığı, mezgit balığı, ringa balığı, pirana, turna balığı, denizatı, dil balığı, vatoz, ton balığı
5	4	Kurbağa, kurbağa, semender, kara kurbağası
6	8	Pire, sivri sinek, bal arısı, kara sinek, uğur böceği, güve, beyaz karınca, arı
7	10	İstiridye, yengeç, kerevit, ıstakoz, ahtapot, akrep, deniz arısı, sümüklü böcek, deniz yıldızı, solucan

3.3. Kongre Oylarına Ait Veri Seti

Bu veri seti 1984 yılında Amerika Birleşik Devletlerinde yapılan kongre oylarını içermektedir. Her bir gözlem kongredeki bir kişinin oylarını göstermektedir. Oylar vergisiz ihracat, göç etmek gibi 16 durumu içermektedir. Her bir oy için cevap evet ya da hayırdır. Ne evet ne de hayır demeyenler kayıp gözlem olarak ifade edilmiştir fakat bunlar belirsiz olarak alınacaktır. Bu yüzden her bir kayıp gözlem üçüncü bir durumu ifade eden yani belirsiz anlamına gelen oyları temsil edecektir. 168 tane Cumhuriyetçi ve 267 tane Demokrata ait toplam 435 gözlem vardır. Çizelge 3.4. değişkenleri tanımlamaktadır. İlk değişken sınıf adlarını içerdiği için analize dahil edilmeyecektir. Veri setindeki kayıp gözlem sayısı 288'dir.

Çizelge 3.4. Kongre oyları veri setine ait değişkenler

Değişken Numarası	**Değişken İsmi**
1	Sınıf adları
2	Özürlü bebekler
3	Su projesi maliyet paylaşımı
4	Bütçe devriminin benimsenmesi
5	Doktor ücretini dondurmak
6	El-Salvador yardımı
7	Okuldaki dini gruplar
8	Önleyici uydu test yasağı
9	Nikaragua'lılar için yardım
10	Füze karışımı
11	Göç etme
12	Şirket azaltma
13	Eğitim harcamaları
14	Süper fon dava hakkı
15	Suç
16	Vergisiz ihracat
17	Afrika dışında ihracat yönetim kanunu

3.4. Arabaların Değerlendirilmesine Ait Veri Seti

Bu veri seti arabaların modelleriyle ilgili bilgi içermektedir. 4 farklı model içeren veri setinde 1728 gözlem vardır ve araba modellerinin dağılımı birinci model için 1210, ikinci model için 384, üçüncü model için 69 ve dördüncü model için 65 tanedir. Arabaların değerlendirilmesi için 6 değişken vardır ve bunlardan 2 değişken sayısaldır. Kapıların sayısını ve arabanın aldığı kişi sayısını içeren sayısal değişkenlerde her birine aynı önem verilecek şekilde kategorikmiş gibi davranılacaktır. Veri setinde kayıp gözlem yoktur.

Çizelge 3.5. Arabalar veri setine ait değişkenler

Değişken Numarası	**Değişken İsmi**	**Değişken Değeri**
1	Satın alımı	Çok yüksek, yüksek, orta, düşük
2	Bakımı	Çok yüksek, yüksek, orta, düşük
3	Kapıları	2, 3, 4, 5 ve daha fazla
4	Kişi sayısı	2, 4, 4 den fazla
5	Taşıma kapasitesi	Küçük, orta, büyük
6	Güvenlik	Düşük, orta, yüksek

3.5. Soya Fasulyesi Verisine Ait Kümeleme Sonuçlarının Değerlendirilmesi

Hastalıklı soya fasulyesi verisi kullanılarak yapılan analizlerde hiyerarşik kümeleme tekniklerinden tek bağlantı tekniği, tam bağlantı tekniği ve ortalama bağlantı tekniği ile Jaccard benzerlik ölçüsü kullanılarak kümeleme yapılmıştır. Hiyerarşik kümeleme tekniklerinde ‘k’ küme sayısı veya ‘θ’ benzerlik eşiği kullanıcı tarafından belirlenmektedir. Kümeleme sonuçlarını düzgün bir şekilde değerlendirebilmek için bütün tekniklerde küme sayısı aynı alınmıştır ve küme sayısı belirlenirken verilerin uzmanlar tarafından belirlenmiş sınıf sayıları dikkate alınmıştır. Soya fasulyesi verisi için dört sınıf vardır ve kümelemeden önce küme sayısı dört olarak belirlenmiştir. K-modes algoritmasında da benzerlik ölçüsü olarak basit eşleşme katsayısı

kullanılmakta ve yine ‘k’ küme sayısı önceden kullanıcı tarafından belirlenebilmektedir. Hiyerarşik kümeleme tekniklerinde de küme sayısı uzmanlar tarafından belirlenmiş sınıf sayısı ile eşit olarak alınmış ve sonuçlar buna göre değerlendirilmiştir.

Çizelge 3.6. Soya fasulyesi veri seti için K-modes algoritmasının kümeleme sonuçları

	D1	**D2**	**D3**	**D4**
1. küme	**10**	6	0	0
2. küme	0	**1**	2	0
3. küme	0	0	**8**	1
4. küme	0	3	0	**16**
Toplam	10	10	10	17

Çizelge 3.6 incelediğinde K-modes algoritmasının kümeleri net olarak ayıramadığı görülmektedir. Birinci küme birinci sınıfa ait nesneleri tam olarak alırken ikinci sınıfa ait nesnelerin de 6 tanesini almıştır. Üçüncü kümeye ait 10 nesnenin 8 tanesi yine üçüncü kümede, dördüncü kümeye ait 17 nesnenin 16 tanesi dördüncü kümede yani doğru kümede kümelenmiştir. K-modes algoritmasının doğru kümeleme yüzdesi %75 olarak tespit edildi. Kümeleme sonucunda kümeler net olarak ayrılamasa da iyi bir kümeleme için küme içi homojenlik kümeler arası heterojenlik kriteri yüksek derecede sağlanmıştır.

Çizelge 3.7. Soya fasulyesi veri seti için tek bağlantı tekniğinin kümeleme sonuçları

	D1	**D2**	**D3**	**D4**
1. küme	**10**	0	0	0
2. küme	0	**10**	0	0
3. küme	0	0	**10**	0
4. küme	0	0	0	**17**
Toplam	10	10	10	17

Çizelge 3.7 incelendiğinde hiyerarşik kümeleme tekniklerinden tek bağlantı tekniği bütün kümeleri net bir şekilde oluşturmuş ve doğru kümeleme yüzdesi %100 olarak tespit edilmiştir. İyi bir kümeleme için gerekli olan küme içi homojenlik kümeler arası heterojenlik kriteri ise tam olarak sağlanmıştır.

Çizelge 3.8. Soya fasulyesi veri seti için tam bağlantı tekniğinin kümeleme sonuçları

	D1	**D2**	**D3**	**D4**
1. küme	**10**	0	0	0
2. küme	0	**10**	0	0
3. küme	0	0	**10**	0
4. küme	0	0	0	**17**
Toplam	10	10	10	17

Çizelge 3.8 incelendiğinde hiyerarşik kümeleme tekniklerinden tam bağlantı tekniği bütün kümeleri net bir şekilde oluşturmuş ve doğru kümeleme yüzdesi %100 olarak tespit edilmiştir. İyi bir kümeleme için gerekli olan küme içi homojenlik kümeler arası heterojenlik kriteri ise tam olarak sağlanmıştır.

Çizelge 3.9. Soya fasulyesi veri seti için ortalama bağlantı tekniğinin kümeleme sonuçları

	D1	**D2**	**D3**	**D4**
1. küme	**10**	0	0	0
2. küme	0	**10**	0	0
3. küme	0	0	**10**	0
4. küme	0	0	0	**17**
Toplam	10	10	10	17

Çizelge 3.9 incelendiğinde hiyerarşik kümeleme tekniklerinden ortalama bağlantı tekniği de bütün kümeleri net bir şekilde oluşturmuş ve doğru kümeleme yüzdesi %100 olarak tespit edilmiştir. İyi bir kümeleme için gerekli olan küme içi homojenlik kümeler arası heterojenlik kriteri ise tam olarak sağlanmıştır.

Çizelge 3.7, çizelge 3.8 ve çizelge 3.9 ile verilen kümeleme sonuçlarına baktığımızda 47 gözlem ve 35 değişkene sahip soya fasulyesi veri setinde, hiyerarşik tekniklerden tek bağlantı tekniği, tam bağlantı tekniği ve ortalama bağlantı tekniğinin her üçü de net bir şekilde kümeleme yaparak K-modes algoritmasından daha iyi sonuçlar vermiştir.

3.6. Hayvanlar Verisine Ait Kümeleme Sonuçlarının Değerlendirilmesi

Hayvanlara ait veri kullanılarak yapılan analizlerde hiyerarşik kümeleme tekniklerinden tek bağlantı tekniği, tam bağlantı tekniği ve ortalama bağlantı tekniği ile Jaccard benzerlik ölçüsü kullanılarak kümeleme yapılmıştır. Kümeleme sonuçlarını düzgün bir şekilde değerlendirebilmek için yukarıda da olduğu gibi bütün tekniklerde küme sayısı aynı alınmıştır ve küme sayısı belirlenirken verilerin uzmanlar tarafından belirlenmiş sınıf sayıları dikkate alınmıştır. Hayvanlara ait veri seti için yedi sınıf vardır ve kümelemeden önce küme sayısı yedi olarak belirlenmiştir. K-modes algoritmasında da benzerlik ölçüsü olarak basit eşleşme katsayısı kullanılmakta ve yine ‘k’ küme sayısı önceden kullanıcı tarafından belirlenebilmektedir. Hiyerarşik kümeleme tekniklerinde de küme sayısı uzmanlar tarafından belirlenmiş sınıf sayısı ile eşit olarak alınmış ve sonuçlar buna göre değerlendirilmiştir.

Çizelge 3.10. Hayvanlara ait veri seti için K-modes algoritmasının kümeleme sonuçları

	1.sınıf	**2.sınıf**	**3.sınıf**	**4.sınıf**	**5.sınıf**	**6.sınıf**	**7.sınıf**
1. küme	**32**	0	2	0	1	0	0
2. küme	2	**20**	0	0	0	5	0
3. küme	4	0	**3**	0	0	0	1
4. küme	3	0	0	**13**	0	0	0
5. küme	0	0	0	0	**3**	0	0
6. küme	0	0	0	0	0	**2**	3
7. küme	0	0	0	0	0	1	**6**
Toplam	41	20	5	13	4	8	10

Çizelge 3.10 incelediğinde K-modes algoritması ile 41 tane hayvan içeren birinci sınıftan 32 tanesinin, 20 tane hayvan içeren ikinci sınıftan 20 tanesinin, 5 tane hayvan içeren üçüncü sınıftan 3 tanesinin, 13 tane hayvan içeren dördüncü sınıftan 13 tanesinin, 4 tane hayvan içeren beşinci sınıftan 3 tanesinin, 8 tane hayvan içeren altıncı sınıftan 2 tanesinin ve 10 tane hayvan içeren yedinci sınıftan 6 tanesinin doğru kümelendiği görülmektedir. K-modes algoritmasının doğru kümeleme yüzdesi %79'dur. Ayrıca küme içi homojenlik kümeler arası heterojenlik kriterini yüksek oranda sağlamaktadır.

Çizelge 3.11. Hayvanlara ait veri seti için tek bağlantı tekniğinin kümeleme sonuçları

	1.sınıf	**2.sınıf**	**3.sınıf**	**4.sınıf**	**5.sınıf**	**6.sınıf**	**7.sınıf**
1. küme	**41**	0	0	0	0	0	0
2. küme	0	**20**	1	0	0	0	0
3. küme	0	0	**1**	0	0	0	0
4. küme	0	0	0	**13**	0	0	0
5. küme	0	0	3	0	**4**	0	0
6. küme	0	0	0	0	0	**0**	1
7. küme	0	0	0	0	0	8	**9**
Toplam	41	20	5	13	4	8	10

Çizelge 3.11 incelendiğinde tek bağlantı tekniğinin 41 tane hayvan içeren birinci sınıftan 41 tanesini, 20 tane hayvan içeren ikinci sınıftan 20 tanesini, 5 tane hayvan içeren üçüncü sınıftan 1 tanesini, 13 tane hayvan içeren dördüncü sınıftan 13 tanesini, 4 tane hayvan içeren beşinci sınıftan 4 tanesini, 8 tane hayvan içeren altıncı sınıftan hiçbirini ve 10 tane hayvan içeren yedinci sınıftan 9 tanesini doğru kümelediği görülmektedir. Doğru kümeleme yüzdesine bakılacak olursa %86'lık bir orana sahiptir. Ayrıca küme içi homojenlik kümeler arası heterojenlik kriterini yüksek oranda sağlamaktadır.

Çizelge 3.12. Hayvanlara ait veri seti için tam bağlantı tekniğinin kümeleme sonuçları

	1.sınıf	2.sınıf	3.sınıf	4.sınıf	5.sınıf	6.sınıf	7.sınıf
1. küme	**36**	0	0	0	0	0	0
2. küme	0	**20**	0	0	0	0	0
3. küme	1	0	**1**	0	0	0	0
4. küme	0	0	4	**13**	4	0	0
5. küme	4	0	0	0	**0**	0	0
6. küme	0	0	0	0	0	**8**	2
7. küme	0	0	0	0	0	0	**8**
Toplam	41	20	5	13	4	8	10

Çizelge 3.12 incelendiğinde tam bağlantı tekniğinin 41 tane hayvan içeren birinci sınıftan 36 tanesini, 20 tane hayvan içeren ikinci sınıftan 20 tanesini, 5 tane hayvan içeren üçüncü sınıftan 1 tanesini, 13 tane hayvan içeren dördüncü sınıftan 13 tanesini, 4 tane hayvan içeren beşinci sınıftan hiçbirini, 8 tane hayvan içeren altıncı sınıftan 8 tanesini ve 10 tane hayvan içeren yedinci sınıftan 8 tanesini doğru kümelediği görülmektedir. Doğru kümeleme yüzdesine bakılacak olursa %85'lik bir orana sahiptir. Ayrıca küme içi homojenlik kümeler arası heterojenlik kriterini yüksek oranda sağlamaktadır.

Çizelge 3.13. Hayvanlara ait veri seti için ortalama bağlantı tekniğinin kümeleme sonuçları

	1.sınıf	2.sınıf	3.sınıf	4.sınıf	5.sınıf	6.sınıf	7.sınıf
1. küme	**40**	0	0	0	0	0	0
2. küme	0	**20**	0	0	0	0	0
3. küme	1	0	**1**	0	0	0	0
4. küme	0	0	4	**13**	4	0	0
5. küme	0	0	0	0	**0**	0	1
6. küme	0	0	0	0	0	**8**	2
7. küme	0	0	0	0	0	0	**7**
Toplam	41	20	5	13	4	8	10

Çizelge 3.13 incelendiğinde ortalama bağlantı tekniğinin 41 tane hayvan içeren birinci sınıftan 40 tanesini, 20 tane hayvan içeren ikinci sınıftan 20 tanesini, 5 tane hayvan içeren üçüncü sınıftan 1 tanesini, 13 tane hayvan içeren dördüncü sınıftan 13 tanesini, 4 tane hayvan içeren beşinci sınıftan hiçbirini, 8 tane hayvan içeren altıncı sınıftan 8 tanesini ve 10 tane hayvan içeren yedinci sınıftan 7 tanesini doğru kümelediği görülmektedir. Doğru kümeleme yüzdesine bakılacak olursa %88'lik bir orana sahiptir. Ayrıca küme içi homojenlik kümeler arası heterojenlik kriterini yüksek oranda sağlamaktadır.

Bu sonuçlara göre 101 tane gözlem ve 16 tane değişken içeren hayvanlara ait veri setinde en iyi kümeleme sırasıyla ortalama bağlantı tekniği, tek bağlantı tekniği, tam bağlantı tekniği ve K-modes algoritması şeklinde olmuştur. Fakat aradaki fark oldukça küçüktür ayrıca bütün yöntemler küme içi homojenlik kümeler arası heterojenlik kriterini yüksek oranda sağlamıştır.

3.7. Kongre Oyları Veri Setine Ait Kümeleme Sonuçlarının Değerlendirilmesi

Kongre oylarına ait veri seti kullanılarak yapılan analizlerde hiyerarşik kümeleme tekniklerinden tek bağlantı tekniği, tam bağlantı tekniği ve ortalama bağlantı tekniği ile Jaccard benzerlik ölçüsü kullanılarak kümeleme yapılmıştır. Kümeleme

sonuçlarını düzgün bir şekilde değerlendirebilmek için bütün tekniklerde küme sayısı aynı alınmıştır ve küme sayısı belirlenirken verilerin uzmanlar tarafından belirlenmiş sınıf sayıları dikkate alınmıştır. Kongre oylarına ait veri seti için iki sınıf vardır ve kümelemeden önce küme sayısı iki olarak belirlenmiştir. K-modes algoritmasında da benzerlik ölçüsü olarak basit eşleşme katsayısı kullanılmakta ve yine 'k' küme sayısı önceden kullanıcı tarafından belirlenebilmektedir. Hiyerarşik kümeleme tekniklerinde de küme sayısı uzmanlar tarafından belirlenmiş sınıf sayısı ile eşit olarak alınmış ve sonuçlar buna göre değerlendirilmiştir.

Çizelge 3.14. Kongre oylarına ait veri seti için K-modes algoritmasının kümeleme sonuçları

	Cumhuriyetçiler	**Demokratlar**
1. küme	**157**	69
2.küme	11	**198**
Toplam	168	267

Çizelge 3.14 incelendiğinde K-modes algoritması 168 cumhuriyetçiden 157 tanesini ve 267 demokrattan 198 tanesini doğru kümeleyerek iyi bir kümeleme performansı sergilemiştir. Doğru kümeleme yüzdesi %82 olarak tespit edildi. Ayrıca küme içi homojenlik kümeler arası heterojenlik kriterini de yüksek oranda sağlamaktadır.

Çizelge 3.15. Kongre oylarına ait veri seti için tek bağlantı tekniğinin kümeleme sonuçları

	Cumhuriyetçiler	**Demokratlar**
1. küme	**2**	1
2.küme	166	**266**
Toplam	168	267

Çizelge 3.15 incelediğinde hiyerarşik kümeleme tekniklerinden tek bağlantı tekniğinin kümeleme sonucunun çok kötü olduğu görülmektedir. Kümelerden birinde nesnelerin çoğunluğu bulunurken ikincisinde sadece 3 nesne bulunmaktadır ve bunlardan ikisi cumhuriyetçilere aitken birisi de demokratlara aittir. Tek bağlantı tekniğine ait doğru kümeleme yüzdesi %61 olarak bulundu. Küme içi homojenlik ve kümeler arası heterojenlik kriteri ise çok düşük olarak tespit edildi.

Çizelge 3.16. Kongre oylarına ait veri seti için tam bağlantı tekniğinin kümeleme sonuçları

	Cumhuriyetçiler	**Demokratlar**
1. küme	**161**	62
2.küme	7	**205**
Toplam	168	267

Çizelge 3.16 incelendiğinde tam bağlantı tekniği 168 cumhuriyetçiden 161 tanesini ve 267 demokrattan 205 tanesini doğru kümeleyerek iyi bir kümeleme performansı

sergilemiştir. Doğru kümeleme yüzdesi %84 olarak tespit edildi. Ayrıca küme içi homojenlik kümeler arası heterojenlik kriterini de yüksek oranda sağlamaktadır.

Çizelge 3.17. Kongre oylarına ait veri seti için ortalama bağlantı tekniğinin kümeleme sonuçları

	Cumhuriyetçiler	**Demokratlar**
1. küme	**2**	1
2.küme	166	**266**
Toplam	168	267

Çizelge 3.17 incelediğinde hiyerarşik kümeleme tekniklerinden ortalama bağlantı tekniğinin kümeleme sonucunun da tek bağlantı tekniği gibi çok kötü olduğu görülmektedir. İki tane kümeden birinde nesnelerin çoğunluğu bulunurken ikincisinde sadece 3 nesne bulunmaktadır ve bunlardan ikisi cumhuriyetçilere aitken birisi de demokratlara aittir. Ortalama bağlantı tekniğine ait doğru kümeleme yüzdesi %61 olarak bulundu. Küme içi homojenlik ve kümeler arası heterojenlik kriteri ise çok düşük olarak tespit edildi.

435 gözlem ve 16 tane değişken içeren kongre oylarına ait veri setinde tam bağlantı tekniği ve K-modes algoritması birbirine yakın ve iyi bir kümeleme performansı sergilerken tek bağlantı tekniği ve ortalama bağlantı tekniği çok kötü kümeleme sonuçları vermiştir.

3.8. Arabalar Verisine Ait Kümeleme Sonuçlarının Değerlendirilmesi

Arabalara ait veri seti kullanılarak yapılan analizlerde hiyerarşik kümeleme tekniklerinden tek bağlantı tekniği, tam bağlantı tekniği ve ortalama bağlantı tekniği ile Jaccard benzerlik ölçüsü kullanılarak kümeleme yapılmıştır. Kümeleme

sonuçlarını düzgün bir şekilde değerlendirebilmek için bütün tekniklerde küme sayısı aynı alınmıştır ve küme sayısı belirlenirken verilerin uzmanlar tarafından belirlenmiş sınıf sayıları dikkate alınmıştır. Arabalara ait veri seti için dört sınıf vardır ve kümelemeden önce küme sayısı dört olarak belirlenmiştir. K-modes algoritmasında da benzerlik ölçüsü olarak basit eşleşme katsayısı kullanılmakta ve yine 'k' küme sayısı önceden kullanıcı tarafından belirlenebilmektedir. Hiyerarşik kümeleme tekniklerinde de küme sayısı uzmanlar tarafından belirlenmiş sınıf sayısı ile eşit olarak alınmış ve sonuçlar buna göre değerlendirilmiştir.

Çizelge 3.18. Arabalara ait veri seti için K-modes algoritmasının kümeleme sonuçları

	1. model	**2. model**	**3. model**	**4. model**
1. küme	**295**	20	0	0
2. küme	394	**203**	28	37
3. küme	209	114	**35**	12
4. küme	312	47	6	**16**
Toplam	1210	384	69	65

Çizelge 3.18 incelendiğinde K-modes algoritması kümelerin net olarak ayrılmasını sağlayamamıştır fakat küme içi homojenlik kümeler arası heterojenlikte hiyerarşik tekniklere göre daha iyi sonuçlar vermiştir. Doğru kümeleme yüzdesi %33 olarak tespit edilmiştir.

Çizelge 3.19. Arabalara ait veri seti için tek bağlantı tekniğinin kümeleme sonuçları

	1. model	**2. model**	**3. model**	**4. model**
1. küme	**0**	0	0	1
2. küme	0	**1**	0	0
3. küme	0	0	**1**	0
4. küme	1210	383	68	**64**
Toplam	1210	384	69	65

Çizelge 3.19 incelendiğinde tek bağlantı tekniğinin bütün sınıfları tek bir grupta toplayarak kötü bir kümeleme yaptığı görülmektedir. Doğru kümeleme yüzdesi %04

olarak tespit edildi. Dolayısıyla küme içi homojenlik ve kümeler arası heterojenlik kriteri sağlanamamıştır.

Çizelge 3.20. Arabalara ait veri seti için tam bağlantı tekniğinin kümeleme sonuçları

	1. model	2. model	3. model	4. model
1. küme	**169**	29	0	0
2. küme	761	**249**	57	40
3. küme	150	42	**12**	12
4. küme	130	64	0	**13**
Toplam	1210	384	69	65

Çizelge 3.20 incelendiğinde tam bağlantı tekniği de tek bağlantı tekniğinde olduğu gibi birçok sınıfı tek bir kümede toplamıştır, tek bağlantı tekniği ve ortalama bağlantı tekniğine göre daha iyi sonuçlar vermesine rağmen iyi bir kümeleme sonucu vermemiştir. Doğru kümeleme yüzdesi de %25 olarak tespit edilmiştir.

Çizelge 3.21. Arabalara ait veri seti için ortalama bağlantı tekniğinin kümeleme sonuçları

	1. model	2. model	3. model	4. model
1. küme	**271**	64	10	6
2. küme	229	**71**	12	12
3. küme	247	109	**23**	26
4. küme	463	140	24	**21**
Toplam	1210	384	69	65

Çizelge 3.21 incelendiğinde ortalama bağlantı tekniği bütün kümeleri eşit olarak bölmüş ve her sınıfı her kümeye eşit olarak dağıtmıştır. Ortalama bağlantı tekniğinin kümeleme sonucu, küme içi homojenlik kümeler arası heterojenliği sağlamak yerine tam tersi küme içi heterojenlik kümeler arası homojenliği sağlamıştır ve kötü bir kümeleme sonucu vermiştir. Doğru kümeleme yüzdesi ise %21 olarak tespit edildi.

Arabalarla ilgili veri setinde kümeleme sonuçlarının iyi çıkmamasının en önemli sebebi de 1728 gözleme sahipken bunları açıklayacak sadece 6 değişken olmasıdır.

Değişken sayısı çok az olduğu için bütün yöntemler kötü bir kümeleme performansı sergilemiştir fakat burada incelenmek istenen büyük veri setlerinde hangi yöntemlerin daha iyi sonuçlar verdiğini görmektir.
Kullanılan veri setlerine göre doğru kümeleme yüzdelerini bir tablo ile gösterelim.

Çizelge 3.22. Algoritmaların doğru kümeleme yüzdesi

Veri setleri ve veri sayısı	**Tek Bağlantı Tekniği**	**Tam Bağlantı Tekniği**	**Ortalama Bağlantı Tekniği**	**K-modes Algoritması**
Soya fasulyesi veri seti (47)	%100	%100	%100	%75
Hayvanlar veri seti (101)	%86	%85	%88	%79
Kongre oyları veri seti (435)	%61	%83	%61	%82
Arabalar veri seti (1728)	%04	%25	%22	%33

Çizelge 3.22. incelendiğinde veri sayısı arttıkça kümeleme performansının hiyerarşik tekniklerde azaldığı, K-modes algoritmasında ise arttığı gözlemlenmektedir. Arabalara ait veri setinde ise yukarıda da açıkladığımız gibi veri sayısı fazla iken değişken sayısının çok az olmasından dolayı düşük kümeleme performansları elde edilmiştir.

5.SONUÇ VE ÖNERİLER

İyi bir karşılaştırma yapmak için kategorik verileri kümelemede literatürde de en çok kullanılan gerçek veri setleri kullanılarak yapılan analizlerde tek bağlantı tekniği, tam bağlantı tekniği, ortalama bağlantı tekniği ve K-modes algoritmalarının kümeleme performansları değerlendirildi.

Kümeleme sonuçlarına göre 47 nesne ve 35 değişkene sahip soya fasulyesi verisinde hiyerarşik tekniklerin hepsi kümeleri net bir şekilde oluşturabilmiş ve hiç yanlış kümeleme yapılmamıştır. K-modes algoritması ise iyi bir kümeleme performansı sergilemesine rağmen kümeleri hatasız olarak ayıramamıştır. Hiyerarşik tekniklerden her üçü de %100'lük bir doğru kümeleme performansı sergilerken K-modes algoritması %75 oranında doğru kümeleme yapmıştır.

101 tane nesne ve 16 değişken içeren hayvanların sınıflandığı veri setinde bütün yöntemlerde kümeler net bir şekilde ayrılamamıştır fakat ortalama bağlantı tekniği %88, tek bağlantı tekniği %86, tam bağlantı tekniği %85ve K-modes algoritması %79'luk bir kümeleme performansı sergileyerek iyi sonuçlar vermişlerdir.

435 nesne ve 16 değişken içeren kongre oylarına ait veri setinde ise soya fasulyesi ve hayvanlara ait veri setinde en iyi sonuçları veren tek bağlantı tekniği ve ortalama bağlantı tekniği kötü sonuçlar vererek kümeleri ayıramamıştır. Buna karşılık K-modes algoritması ve tam bağlantı tekniği birbirine yakın sonuçlar vererek iyi bir kümeleme performansı sergilemişlerdir. Tek bağlantı tekniği ve ortalama bağlantı tekniği %61, tam bağlantı tekniği %84 ve K-modes algoritması ise %82 oranında doğru kümeleme yapmıştır.

Son olarak 1728 nesne ve 6 değişken içeren arabalara ait veri setinde bütün kümeleme yöntemleri kötü bir kümeleme performansı sergilemiştir. Tek bağlantı tekniği %04, ortalama bağlantı tekniği %22, tam bağlantı tekniği %25 ve K-modes algoritması %33 oranında doğru kümeleme yapmıştır. Arabalara ait veri setinde kümeleme sonuçlarının kötü olmasının en önemli sebebi nesne sayısı çok fazla iken

bunları açıklayan değişken sayısının çok az olmasıdır. Kümeleme analizinde iyi sonuçlar elde edebilmek için her zaman değişken sayısını yüksek tutmak gerekmektedir. Görüldüğü gibi kümelerin net bir şekilde ayrımı 47 gözlem ve 35 değişken içeren soya fasulyesi verisinde sağlanmıştır. Farklı veri ve değişken sayılarına göre yapılan analizlerde değişken sayısının fazla olması küme içi homojenlik kümeler arası heterojenlik kriterinin sağlanması için önemli bir unsur olduğu görülmektedir.

Analiz sonuçlarına göre kullanılan dört farklı veri setinde veri sayısı arttıkça hiyerarşik tekniklerden tek bağlantı, tam bağlantı ve ortalama bağlantı tekniklerinin kümeleme performansı azalırken bölmeli bir kümeleme yöntemi olan K-modes algoritmasının kümeleme performansının arttığı gözlemlenmiştir. İyi bir kümeleme algoritmasının önemli özelliklerinden biri de büyük veri setlerine uygulanabilir olmaktır. Elde edilen bulgular doğrultusunda hiyerarşik tekniklerden her üçünün de küçük veri setlerinde iyi sonuçlar verdiği fakat veri seti büyüdükçe K-modes algoritmasının hiyerarşik tekniklerden daha iyi sonuçlar verdiği tespit edilmiştir. Sonuçlar doğrultusunda veri seti büyük olduğunda K-modes algoritmasının kullanılması önerilmektedir.

KAYNAKLAR

1. Guo, L. , "Clustering Categorical Response", Master Thesis, ***Office of Graduate Studies College of Arts and Sciences***, Georgia State University, 1-2, (2008).

2. Andritsos, P., "Scalable Clustering of Categorical Data and Applications", PHD Thesis, ***Graduate Department of Computer Science, University of Toronto***, 1-22, (2004).

3. Wu, L.F., Hughes, T.R., Davierwala, A.P., Robinson, M.D., Altschuler, S.J., "Large-Scale Prediction of Saccharomyces Cerevisiae Gene Function Using Overlapping Transcriptional Cluster", ***Nature Genetics,*** 31(3):255-265, (2002).

4. Jiang, D. , Tang, C. , Zhang, A.,"Cluster Analysis for Gene Expression Data" ***IEEE Transactions on Knowledge and Data Engineering,*** 16(11):1370-1386, (2004).

5. May, A., Leore, M., Boecker, H., Sprenger, H., Juergens, T., Bussone, G., Tolle, T.R., "Hypothalamic Deep Brain Stimulation in Pozitron Emission Tomography", ***The Journal of Neuroscience***, 26(13):3589-3593, (2006).

6. Mathieu, R., Gibson, J.,"A Methodology for Large Scale R&D Planning Based on Cluster Analysis" ***IEEE Transactions on Engineering Management***, 40(3):283–292, (2004).

7. Fraley, C., Raftery, A.E., "How Many Clusters? Which Clusterring Method? Answers Via Model Based Cluster Analysis", ***Department of Statistics, Universitynof Washington***, Technical Report No:329, 41(8):579-587, (1998).

8. Han, J., Kamber, M., "Data Mining Concepts and Techniques", ***Morgan Kaufmann Publishers Inc.***, San Francisco CA, USA, 418-429, (2000).

9. Triphaty, B. K., Ghosh, A.,"SSDR: An algorithm for Clustering Categorical Data Using Rough Set Theory", ***Advances in Applied Science Research***, 2(3):314-326, (2011).

10. Huang, Z. ," Extensions to the k-Means Algorithm for Clustering Large Data Sets with Categorical Values", ***Data Mining and Knowledge Discovery***, 2(3): 283-304, (1998).

11. Gibson, D., Kleinberg, J., Raghavan, P. ," Clustering categorical data: an approach based on dynamical systems" ***In Proceedings of the 24th VLDB Conference***, New York, USA, 311-322, (1998).

12. Ganti, V., Gehrke, J., Ramakrishan, R.” CACTUS: Clustering categorical data using summaries”, ***In Proceedings of ACM SIGKDD, International Conference on Knowledge Discovery&Data Mining***, San Diego, CA, USA, 73- 83, (1999).

13. Guha, S., Rastogi, R., Shim, K., ” ROCK: A robust clustering algorithm for categorical attributes”, ***Proceedings of the IEEE International Conference on Data Engineering***, Sydney, 345-366, (1999).

14. He, Z., Xu, X., Deng, S., “Squeezer: An Efficient Algorithm for Clustering Categorical Data”, ***Department of Computer Science and Engineering, Harbin Institue of Technology***, 17(5):611-624, (2002).

15. Rezankova, H.,”Cluster Analysis and Categorical Data”, ***Vysoka Skola Economicka v Praze, Praha***, 223-234, (2009).

16. Koldere Akın, Y. ,”Veri Madenciliğinde Kümeleme Algoritmaları ve Kümeleme Analizi”, Doktora Tezi, ***Marmara Üniversitesi Sosyal Bilimler Enstitüsü***, 4-16, (2008).

17. Nemalhabib, A., “A Cohesion-based Clustering Technique for Categorical Data”, Master Thesis, ***Graduate Department of Computer Science and Software Engineering***, Concordia University, 1-37, (2006).

18. Hartmanis, V., Stearns, R. E., “On the Computational Complexity of Algorithms”, ***Transactions of the American Society,*** 117: 285-306, (1965).

19. Rui, X., Wunsch, D., “Survey of Clustering Algorithms”, ***IEEE Transactions on Neural Networks***, 16(3): 645-675, (2005).

20. Wang, W. , Yang, J., Muntz, R. ,”STING: A Statistical Information Grid Approach to Spatial Data Mining”, ***Proceedings of the 23rd VLDB Conference***, Athens, Greece, 186-195, (1997).

21. Abdu, E., “Clustering Categorical Data Using Summaries and Spectral Techniques”, PHD Thesis, ***Graduate Department of Computer Science***, The University of New York, 1-3, (2009).

22. Estivill-Castro, V., Yang, J., “A Fast and robust general purpose clusteringAlgorithm”, ***Pacific Rim International Conference on Artificial Intelligence***, 208-218, (2000) .

23. Guha, S., Rastogi, R., Shim, K., “ CURE: An efficient algorithm for clustering large databases”, ***Proceedings of ACM-SIGMOD 1998 International Conference on Management of Data***, ***Publication: SIGMOD Record,*** 27(2):73-84, (1998).

24. Zhang, T., Ramakrishnan, R., Livny, M., "BIRCH: An efficient data clustering method for very large databases", ***In Proceedings of ACM SIGMOD International Conference on Management of Data***, 103-114, (1996).

25. Latin, M., Carroll, D. J., Green, P. E., "Analyzing Multivariate Data", ***United States:Thomson Brooks-Cole***, 525, (2003).

26. MacQueen, J.B., "Some methods for classification and analysis of multivariate observations", ***Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability***, 281–297, (1967).

27. Gionis, A., Mannila, H., Tsaparas, P., "Clustering Aggregation", ***The 21 st International Conference on Data Engineering (ICDE)***, 341-352,(2005).

28. Ester, M., Kriegel, H. P., Sander, J., Xu, X., " A Density- Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise", In ***Proceedings of the 2nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)***, 226-231, (1996).

29. Ankerst, M., Breunig, M. M. , Kriegel, H. P., Sander, J., "OPTICS: Ordering Points To Identify the Clustering Structure", ***In Proceedings of the ACM SIGMOD International Conference on the Management of Data***, 49-60, (1996).

30. Hinneburg, A., Keim, D. A. ," An Efficient Approach to Clustering in Large Multimedia Databases with Noise" ***In Proceedings of the 4th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)***, 58-65, (1998).

31. Rokach, L., Maimon, O.,"Data Mining and Knowledge Discovery Handbook", ***Springer,*** Chapter 15,322-349,(2010) .

32. Dempster, A. P., Laird, N. M., Rubin, D. B., "Maximum Likelihood from Incomplete Data Via The EM Algorithm". ***Journal of the Royal Statistical Society***, series B, 34:1-38, (1977).

33. Wang, W., "Data Mining Concept, Algorithms and Applications", ***Comp 290-090 Research Seminar***, University of North Carolina at Chapel Hill, 197-232, (2006).

34. Sheikholeslami, G., Chatterjee, S., Zhang, A.,"WaveCluster: A Multi-Resolution Clustering Approach for Very Large Spa-tial Databases", ***In Proceedings of the 24th International Conference on VeryLarge Data Bases (VLDB),*** 428-439, (1998).

35. Agrawal, R. ,Gehrke, J., Gunopulos, D. And Raghavan, P.,"Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications", ***In Proceedings of the ACM SIGMOD International Conference on the Management of Data***, Seattle, WA, USA, 94-105, (1998).

36. Servi, T. ,"Çok Değişkenli Karma Dağılım Modeline Dayalı Kümeleme Analizi", Doktora Tezi, ***Çukurova Üniversitesi Fen Bilimleri Enstitüsü***, 33-34, (2009).

37. Michel, C.,"Cardinal Nominal or Similarity Measures in Comparative Evaluation of Information Retrieval Process", ***The 2st International Conference on Language Resources & Evaluation (LREC)***, 367, (2000).

38. Barbara, D., Couto, J., Yi L., " COOLCAT: An Entropy-based Algorithm for Categorical Clustering", ***In Proceedings of the 11th International Conference on Information and Knowledge Management (CIKM)***, McLean, VA, USA, 582- 589, (2002).

39. Ng, R. T., Han, J. ," Efficient and effective clustering methods for spatial data mining", ***Proceedings of 1994 International Conference on Very Large DataBases (VLDB'94)***, Santiago, Chile, 144-155, (1994).

40. Goil, S. ,Choudhary, H., " MAFIA: Efficient and scalable subspace clustering for very large data sets",***Submitted for publication to ICDE***, 2-19, (2000).

41. Karyapis, G., Han, E. H., Kumar, V., "CHAMELEON: A hierarchical clustering algorithm using dynamic modeling" ***In IEEE Computer***, 32(8):68-75, (1999).

42. Dutta, M., Kakoti Mahanta, A., Pujari, A. K., "QROCK:A quick version of the ROCK algorithm for clustering ofcategorical data," ***Pattern Recognition Letters***, 26(15): 2364–2373, (2005).

43. Jin, Y., Zuo, W., "Clustering Categorical Data Using Qualified Nearest Neighbors Selection Model***", College of Computer Science and Technology, Jilin University***, Changchun, China, 4304:1037-1041, (2006).

44. Khan, S. S., "Computation of Initial Modes for K-modes Clustering Algorithm Using Evidence Accumulation", ***IJCAI'07 Proocedings of the 20th International Joint Conference on Artifical Intelligence***, 2784-2789, (2007).

45. İnternet: UCI Machine Learning Repository, (2013) http://www.ics.uci.edu/~mlear/MLRepository.html

46. Chaturvedi, A., Green, P., Carrol, J., "K-Modes Clustering", ***Journal of Classification,*** 18:35-55, (2001).

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : BAŞ, Ferhan

Uyruğu : T.C.

Doğum tarihi ve yeri : 05.07.1987 Antalya

Medeni hali : Bekar

Telefon : 0(505)7872811

e-mail : basferhan@gmail.com

Eğitim

Derece	Eğitim Birimi	Mezuniyet tarihi
Yüksek Lisans	Gazi Üniversitesi/İstatistik Bölümü	-
Lisans	Ondokuz Mayıs Üniversitesi/İstatistik Bölümü	2011
Lise	A. Fevzi Alaettinoğlu Lisesi	2005

Yabancı Dil

İngilizce

Hobiler

Spor , bilgisayar teknolojileri, sinema ve tiyatro.