YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

GÖRSEL VERİ MADENCİLİĞİ TEKNİKLERİNİN KÜMELEME ANALİZLERİNDE KULLANIMI VE UYGULANMASI

İstatistikçi Metin Vatansever

FBE İstatistik Anabilim Dalı İstatistik Programında Hazırlanan

YÜKSEK LİSANS TEZİ

Tez Danışmanı: Doç. Dr. Ali Hakan BÜYÜKLÜ

İSTANBUL, 2008

İÇİNDEKİLER

SİMGE LİSTESİ

X	:	Veri matrisi
i,j	:	Matris indisleri
w	:	Ağırlık
Σ	:	Varayans kovaryans matrisi
R	:	Korelasyon matrisi
I	:	Birim matris
λ	:	Özdeğer
e	:	Özvektör
Var	:	Varyans
Cov	:	Kovaryans
s	:	Sandart sapma, benzerlik ölçüsü
C	:	Küme
d	:	Uzaklık ölçüsü
p	:	Değişken sayısı
n	:	Birim sayısı
N	:	Toplam birim(gözlem) sayısı
r	:	Pearson korelasyon katsayısı
T	:	Transpoze
J	:	Giriş vektörünün en yakın olduğu çıkış nöronları.
α	:	Öğrenme katsayısı.
h	:	Komşuluk fonksiyonu
c	:	Kazanan nöron
$f_x(t)$	:	Andrews eğrisi

KISALTMA LİSTESİ

VM	:	Veri Madenciliği
RadViz	:	Radyal koordinat görüntüleme
PolyViz	:	Geliştirilmiş RadViz
PCA	:	Temel bileşenler analizi
MDS	:	Çok boyutlu ölçekleme
SOM	:	Kendinden düzenlenen haritalar
S	:	Silhouette endeksi
DB	:	Davies-Bouldin Endeksi
D	:	Dunn endeksi
CH	:	Calinski ve Harabasz endeksi
KL	:	Krzanowski ve Lai Endeksi
H	:	Hartigan endeksi

ŞEKİL LİSTESİ

ÇİZELGE LİSTESİ

ÖNSÖZ

Sayın Doç. Dr. Ali Hakan BÜYÜKLÜ’ ye tez çalışmamın gerçekleştirilmesinde gerekli yönlendiriciliği sağladığı, her türlü sorumu sabırla cevapladığı, benden desteğini esirgemediği ve tez metnini inceleyerek biçim ve içerik açısından son şeklini almasında katkıda bulunduğu için teşekkür ederim.

Arş. Görevlisi Ömer Bilen’ e veri madenciliği konusunda verdiği bilgi ve kaynaklardan dolayı teşekkür ederim.

Tez çalışması ve tezin yazımı süresi boyunca dostluklarını ve sevgilerinin bir an bile esirgemeyen, bu süre boyunca bana tahammül eden biricik dostlarım Caner Esen, Hülya Derin ve diğer sevgili dostlarıma teşekkürlerimi sunarım.

Ayrıca sevgi ve güvenlerini her zaman hissettiğim aileme teşekkür ederim.

ÖZET

GÖRSEL VERİ MADENCİLİĞİ TEKNİKLERİNİN KÜMELEME ANALİZLERİNDE KULLANIMI VE UYGULANMASI

Metin VATANSEVER
İstatistik, Yüksek Lisans Tezi

Veri madenciliği, geniş veri yığınları içerisinde, yararlı olma potansiyeline sahip, aralarında bilinmedik ilişkilerin olduğu verilerin keşfedilerek, veri sahibi için hem anlaşılır hem de kullanılabilir bir biçime getirilmesine yönelik geliştirilmiş yöntemler topluluğudur. Verilerin grafiksel bir formda temsil edilmesi veri yapılarının anlaşılmasını kolaylaştırır. Ancak çoğunlukla veri madenciliği teknikleri büyük miktarda veri yığınlarıyla uğraşırlar ki veri görselleştirme teknikleri ekran çözünürlüğü, insan algı sistemi gibi sınırlardan dolayı çokta başarılı olamayabilirler. Tezde bu gibi sınırları ortadan kaldırabilmek için çeşitli yeni görselleştirme teknikleri tanıtılmış ve bu görselleştirme teknikleri çok boyutlu, büyük miktarda veri kayıtlarına sahip verilerle örneklendirilmiştir. Bu yeni görselleştirme teknikleri küme yapılarının ve aşırı değerlerin keşfedilmesinde kullanılmıştır. Hatta bu görselleştirme teknikleri farklı kümeleme algoritmalarının bulduğu küme sonuçlarını değerlendirmek için de kullanılmıştır.

Uygulamada, görsel teknikler kullanılarak Türkiye ilçe veri setindeki aşırı değerler ve küme yapıları tespit edilmiştir. Daha sonra bu ilçe veri seti, tek bağlantılı hiyerarşik, tam bağlantılı hiyerarşik, k-ortalamalar ve SOM gibi çoğunlukla kullanılan dört farklı kümeleme algoritmalarıyla kümelenmiştir. Çoğunlukla kullanılan altı küme doğruluk endeksi uygun küme sayısının tespitinde kullanılmıştır. Son olarak da görsel teknikler küme sonuçlarının değerlendirilmesinde kullanılmıştır. Uygulamada ki sonuçlar göstermiştir ki büyük veri setlerinde kullanılan görsel tekniklerin kümeleme çalışmalarında bulunan araştırmacılara aşırı değerlerin tespitinde, kaliteli küme sonuçlarının üretilmesinde ve uygun kümeleme algoritmalarının seçilmesinde yol gösterebilir.

Anahtar Kelimler: Veri madenciliği, görselleştirme teknikleri, görsel veri madenciliği, kümeleme analizi, küme doğruluk, görsel küme doğruluk

ABSTRACT

USING VISUAL DATA MINING TECHNIQUES
IN CLUSTERING ANALYSIS AND AN APPLICATION

Metin VATANSEVER
Statistics, M.S. Thesis

Data mining is the process of automatically discovering useful information in large data repositories. Data mining techniques are deployed to scour large database in order to find novel and useful patterns that might otherwise remain unknown. Data mining techniques frequently focus on the discovery of unknow structures such a clusters, trends, associations and correalations and other structures for which a visual data analysis is very appropriate quite likely to yield insight. However, data mining techniques are often applied to massive data sets where visualization may not be very successful because of the limits of both screen resolution, human visual system resolution as well as the limits of available computational resources. In this thesis, we present new visual tecniques for overcoming such limitations and illustrate the visual tecniques with some examples of successful challenges on high-dimensional and large data sets. The visual tecniques are applied to detect cluster structures and outliers. Also the visual tecniques are applied to evaluate the results of a number of different clustering algorithms.

In practice, cluster structures and outliers in administrative district data set in Turkey are detected by the visual techniques. Then four widely applicable clustering algorithms such as single link hierarchical, complete link hierarchical, k-means and SOM are used to cluster the data set. Six frequently used cluster validity indices are employed to estimate the right number of clusters in the data set. Finally visual tecniques are used to evaluate the results of a number of different clustering algorithms. Our results show that visual techniques let the researcher involve in the clustering process to detect outliers, to generate high-quality clustering results and to choose the right number of cluster algorithms for large datasets

Keywords: Data mining, visualization tecniques; visual data mining; cluster analaysis; cluster validity; visual cluster validity

1. GİRİŞ

Günümüzde bilişim alanında yaşanan hızlı gelişmeler sayesinde bilgisayar sistemleri her geçen gün hem daha ucuzlamakta, hem de güçleri artmaktadır. Artık bilgisayarlar daha büyük miktardaki veriyi saklayabilmekte ve daha kısa sürede işleyebilmektedir. Bunun yanında bilgisayar ağlarındaki ilerleme ile bu veriye başka bilgisayarlardan da hızlı bir şekilde ulaşabilmek mümkün. Bilgisayarların ucuzlaması ile birlikte artık bilgi teknolojileri yaygın olarak hayatın her alanında kullanılabilmektedir. Verilerin dijital ortamlarda saklanmasıyla birlikte yeryüzündeki bilgi miktarı sürekli artmaktadır. Bu veri içersinden anlamlı ve yararlı bilgiyi ortaya çıkartmaksa giderek zorlaşmaktadır. Geleneksel istatistiki yöntemlerle büyük boyuttaki veriyi çözmek kolay değildir. Bu nedenle verileri işlemek ve çözümlemek için özel yöntemlere gereksinim duyulmuştur. Veri madenciliği, veri görselleştirme ve görsel veri madenciliği yöntemleri bu gereksinimi karşılamak üzere ortaya atılmıştır. Bu tez çalışmasında veri madenciliği ve veri görselleştirme yöntemlerinin çok boyutlu veri setlerinde kullanılması sırasında karşılaşılan zorluklar irdelenerek bunlar için çözüm yöntemleri geliştirilmiştir.

Tez çalışması altı bölümden oluşmaktadır. Tezin ikinci bölümde öncelikle veri madenciliğine ilişkin değişik kaynaklarda yapılan tanımlardan yola çıkılarak veri madenciliği kavramları tanımlanmış ve veri madenciliği yöntemleri incelenmiştir.

Üçüncü bölümde, literatürde mevcut bulunan, çok boyutlu veri setlerinde kullanılan görselleştirme teknikleri, çok boyutluluk sorunu, görsel veri madenciliği kavramları ve görsel veri madenciliği yaklaşımları incelenmiştir. Görsel teknikler avantaj ve dezavantajlarıyla tanıtılarak, XmdvTool, Orange ve MATLAB R2007a programları kullanılarak, süsen veri setiyle örneklendirilmiştir.

Dördüncü bölümde, veri madenciliği çalışmalarında önemli bir yeri bulunan kümeleme analizlerinden bahsedilmiştir. Kümeleme analizlerinde kullanılan istatistik tabanlı k-ortalamalar, tek bağlantılı hiyerarşik, tam bağlantılı hiyerarşik ve yapay sinir ağları tabanlı kendi kendine düzenlenen haritalar (SOM) yöntemlerine değinilerek her bir algoritma için MATLAB R2007a programı kullanılarak süsen veri setiyle örnek uygulamalar gerçekleştirilmiştir. Bu bölümde son olarak küme doğruluk (cluster validity) ve kümeleme analiz sonuçlarının çok boyutlu uzayda görselleştirilmesini sağlayan çeşitli görselleştirme yöntemlerine değinilmiştir.

Beşinci bölümde, kümeleme ve görselleştirme yöntemleri kullanılarak bir uygulama gerçekleştirilmiştir. Uygulamada 2000 yılı verileri kullanılarak 81 ildeki 918 ilçe 7 coğrafi

bölge bazında ele alınarak tek bağlantılı hiyerarşik, tam bağlantılı hiyerarşik, k-ortalamalar ve SOM kümeleme yöntemleriyle kümelenmiştir. Daha sonra, Silhouette, Davies-Bouldin, Dunn, Calinski ve Harabasz, Krzanowski ve Lai ve Hartigan küme doğruluk endeksleri doğru küme sayısının tespitinde kullanılmıştır. Son olarak ta görsel teknikler küme doğruluk endekslerinin sonuçlarını değerlendirmek ve küme sonuçlarının anlaşılmasını sağlamak için kullanılmıştır. Bu sayede uygun küme sayısı ve uygun kümeleme algoritmaları tespit edilmiştir. Ayrıca uygulama sonucunda Türkiye' nin bölgeler arası kalkınmasında ciddi dengesizliklerin olduğu ortaya çıkartılmıştır.

Son bölümde her bölümde elde edilen sonuç ve çıkarımlar özetlenmiştir.

2. VERİYİ BİLGİYE DÖNÜŞTÜRMENİN YOLU

Günümüzde bilişim alanında hızlı gelişmeler yaşanmaktadır. Bilgisayar teknolojilerinde de her gün başka bir yenilik ortaya çıkmaktadır. Sadece bilgisayarlar değil veri iletişim teknolojilerinde de hızlı bir gelişme söz konusudur. Bu teknolojilerdeki gelişmeler neticesinde bilgisayar sistemleri her geçen gün hem daha ucuzlamakta, hem de güçleri artmaktadır. İşlemciler gittikçe hızlanmakta, disk kapasiteleri de artmaktadır. Artık bilgisayarlar daha büyük miktardaki veriyi saklayabilmekte ve daha kısa sürede işleyebilmektedir. Bunun yanında bilgisayar ağlarındaki ilerleme ile bu veriye başka bilgisayarlardan da hızlı bir şekilde ulaşabilmek mümkün. Bilgisayarların ucuzlaması ile birlikte bilgi teknolojileri yaygın olarak hayatın her alanında kullanılabilmektedir. Artık kullanıcılar daha yetenekli, daha hızlı ve kullanışlı bilgisayar teknolojilerine kolayca sahip olabilmektedir (Özkan, 2008; Alpaydın, 2000).

Bilişim teknolojilerindeki gelişmeler beraberinde bir sorunu da getirmiştir. Bilişim sistemleri sayesinde artık her bilgi sayısal olarak toplanmakta ve sayısal ortamlarda kaydedilmektedir. Örneğin eskiden süpermarkette kasa basit bir işlem makinesinden ibaretti. Müşterinin o anda satın aldığı malların fiyatlarının toplamını hesaplamak için kullanılırdı. Günümüzde ise bilgi teknolojisindeki gelişmeler sonucu kasa yerine kullanılan satış noktası terminalleri sayesinde alışveriş esnasında müşterinin bütün detaylı bilgileri toplanabilmekte ve dijital ortamlarda saklanabilmektedir. Binlerce müşterisi olan süpermarketler her gün çok sayıda veri toplamak zorunda kalmıştır. Böylece ilgili firma bilgisayarlarında çok büyük miktarda veri biriktirmektedir (Alpaydın, 2000).

Verilerin dijital ortamlarda saklanmasıyla birlikte yeryüzündeki bilgi miktarı sürekli artmaktadır. Berkeley Üniversitesi araştırmacılarının tahminlerine göre her yıl 1 milyon terabayt kadar veri toplanmakta ve dijital ortamlarda saklanmaktadır. Bu, gelecek üç yıl içersinde insanların şimdiye kadar topladığı veriden çok daha fazla veri elde edileceği anlamına gelmektedir (Keim, 2002).

Sensörler ve monitörler aracılıyla veriler günlük hayatta sıklıkla toplanmaktadır. Hatta basit kredi kartı kullanımları, telefon görüşmeleri gibi günlük işler bile bilgisayarlar aracılığıyla kayıt altında tutulmaktadır. Genellikle toplanan veriler çok boyutlu verilerdir. İnsanlar, bu verilerin ticari anlamda rakiplerine karşı üstünlük sağlamalarını sağlayacak, bilimsel anlamda çalışmalara yeni açılımlar getirebilecek değerli bilgiler taşıyan potansiyel kaynaklar olduklarına inandıkları için verileri toplamakta ve saklamaktadırlar (Keim, 2002).

Bilişim teknolojisi hızlı artış gösteren devasa veriyi saklamaya yeterli olabilir. Ancak veriler ne işe yarayacaktır? Bu verilerden firma bazı avantajlar kazanabilecek midir? Biriken veri gerçek anlamda bilgiye dönüştürülebilecek midir? Bu tür sorulara olumlu yanıt vermek mümkündür. Bu veriler üzerinde analizler yapılarak özellikle stratejik seviyede kararalar alınabilir (Özkan, 2008).

Süpermarket örneğimize dönersek binlerce malın ve binlerce müşterinin hareket bilgileri sayesinde her malın zaman içindeki hareketlerine ve eğer müşteriler bir müşteri numarası ile kodlanmışsa bir müşterinin zaman içindeki verilerine ulaşmak ve analiz etmek de olası. Verileri analiz ederek her mal için bir sonraki ayın satış tahminleri yapılabilir. Müşteriler satın aldıkları mallara göre gruplanabilir, yeni bir ürün için potansiyel müşteriler belirlenebilir. Binlerce malın ve müşterinin olabileceği düşünülürse bu analizin gözle ve elle yapılamayacağı, otomatik olarak yapılmasının gerektiği ortaya çıkar. İşte veri madenciliği burada devreye girer (Alpaydın, 2000).

Veriler üzerinde analizler yapabilmek için çeşitli istatistiksel ve matematiksel yöntemler kullanılabilir. Anacak veri sayısı arttıkça sorunlar ortaya çıkacaktır. Veriyi yönetmek için veritabanları, veri ambarı ve veriyi analiz ederek yarlı bilgiye ulaşılmasını sağlayan veri madenciliği kavramları ortaya atılmıştır. Bu bölümde veritabanı, veri ambarı ve veri madenciliği tanım ve kavramları üzerinde durulacaktır.

2.1 Veritabanı

Veri madenciliği büyük miktarda veri inceleme amacı üzerine kurulmuş olduğu için veritabanları ile yakından ilişkilidir. Veri madenciliği tekniklerinin uygulanmasında girdi olarak kullanılan verilerin kaynağı veritabanıdır. Bu yüzden veri madenciliği konusu incelenirken veritabanlarına da mutlaka değinmek gerekir.

“Veritabanı incelenen konu ile ilgili değişkenlerin birimlerdeki görünümlerinin aynı tabloda veya farklı tablolarda saklanabildiği, değiştirilebildiği, araştırılabildiği bir ortamdır. Kısaca söylemek gerekirse veritabanı bir enformasyon kümesidir. Veritabanlarında bulunan enformasyonun bilgiye dönüştürülmesi ise veri madenciliğinin temel amaçlarından birisidir (Bilen, 2004)”.

Büyük miktarlarda verinin veritabanlarında tutulduğu bilindiğine göre bu verilerin veri madenciliği teknikleriyle işlenmesine de veri tabanında bilgi keşfi denir. Büyük hacimli olan ve genelde veri ambarlarında tutulan verilerin işlenmesi yeni kuşak araç ve tekniklerle

mümkün olabilmektedir. Bundan dolayı bu konularda yapılan çalışmalar güncelliğini korumaktadır. Bazı kaynaklara göre; veritabanı bilgi keşfi daha geniş bir disiplin olarak görülmektedir ve veri madenciliği terimi sadece bilgi keşfi metotlarıyla uğraşan veri tabanı bilgi keşfi sürecinde yer alan bir adımdır.

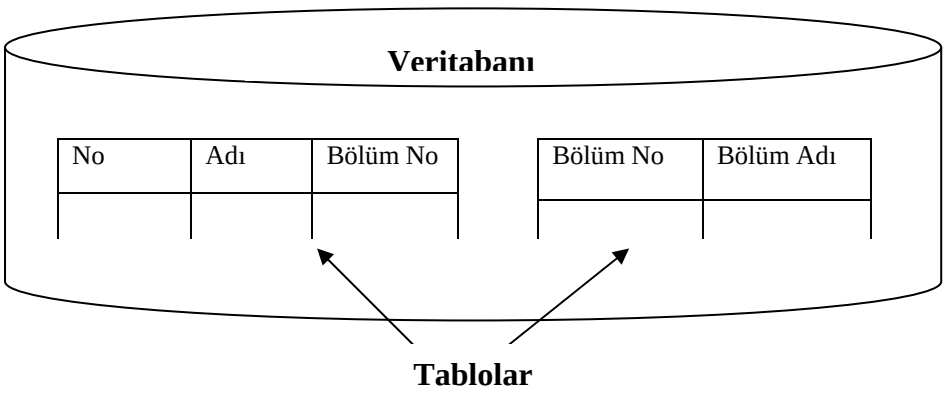
Veri Tabanında asıl önemli kavram, kayıt yığını ya da bilgi parçalarının tanımlanmasıdır. Bu tanıma şema adı verilir. Şema veri tabanında kullanılacak bilgi tanımlarının nasıl modelleneceğini gösterir. Buna veri modeli denir. Şu ana kadar birçok veri modeli geliştirilmiştir. Bu veri modellerinin dört ana grupta toplamak mümkündür (Özkan, 2008).

- Hiyerarşik veri modeli
- Ağ veri modeli
- İlişkisel veri modeli
- Nesneye yönelik veri modeli

2.1.1 İlişkisel Veri Modeli

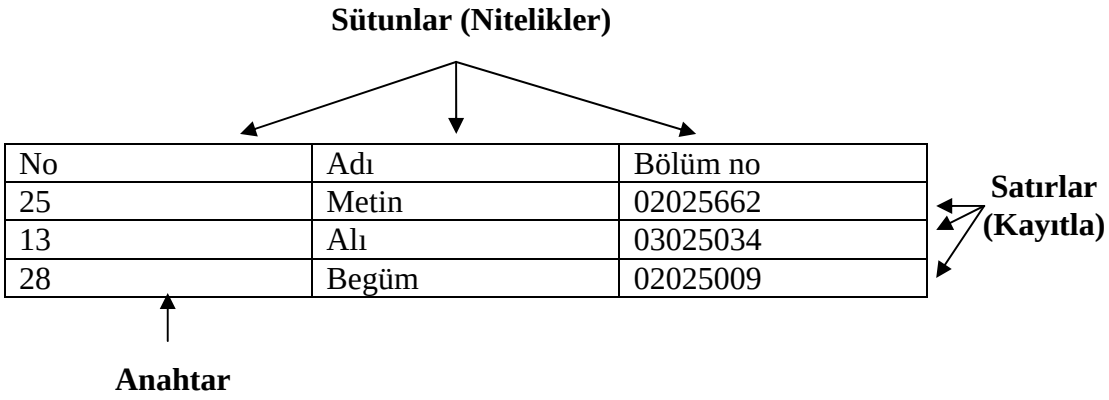
Günümüzde en yaygın kullanılan model, ilişkisel veri modelidir. Layman'ın değimiyle bu modelde veriler tablolarda saklanır. Tablolarda bulunan satırlar (row) kayıtların kendisini, sütunlar (column) ise bu kayıtları oluşturan bilgi parçalarının ne türden olduklarını belirtir. Başka modeller (Sistem Modeli ya da Ağ Modeli gibi.) daha belirgin ilişkiler kurarlar.

İlişkisel model, varlıklar arasında oluşan karmaşık ilişkileri basite indirgemek amacıyla geliştirilmiştir. Bu yaklaşımda veritabanındaki tüm ilişkiler tablolar biçiminde tanımlanmaktadır.



Şekil 2.1 İlişkisel veritabanında tablolar birbirleriyle ilişkilendirilmiştir (Özkan, 2008)

İlişkisel veritabanı, her biri özel isimlere sahip tablolardan oluşur. Burada her tablo bir varlığa ya da bir ilişkiye karşılık gelmektedir. Tablonun sütunları nitelikleri; satırları ise bu niteliklerin değerlerini ifade eder. Her bir satır "kayıt" olarak da düşünülebilir. Anahtar alan tablonun tanımlayıcısıdır.



Şekil 2.2 Tablonun satırları ve sütunları

2.1.2 Veri Tabanı Yazılımları

Verileri sistematik bir biçimde depolayan yazılımlara veritabanı yazılımları denir. Birçok yazılım bilgi depolayabilir ama aradaki fark, veri tabanın bu bilgiyi verimli ve hızlı bir şekilde yönetip değiştirebilmesidir. Veri tabanı, bilgi sisteminin kalbidir ve etkili kullanmakla değer kazanır. Bilgiye gerekli olduğu zaman ulaşabilmek esastır. Bağıntısal veritabanı yönetim sistemleri büyük miktarlardaki verilerin güvenli bir şekilde tutulabildiği, bilgilere hızlı erişim imkanlarının sağlandığı, bilgilerin bütünlük içerisinde tutulabildiği ve birden fazla kullanıcıya aynı anda bilgiye erişim imkanının sağlandığı programlardır. Oracle veri tabanı da bir bağıntısal veri tabanı yönetim sistemidir.

Başlıca ilişkisel veri tabanı sistemleri:

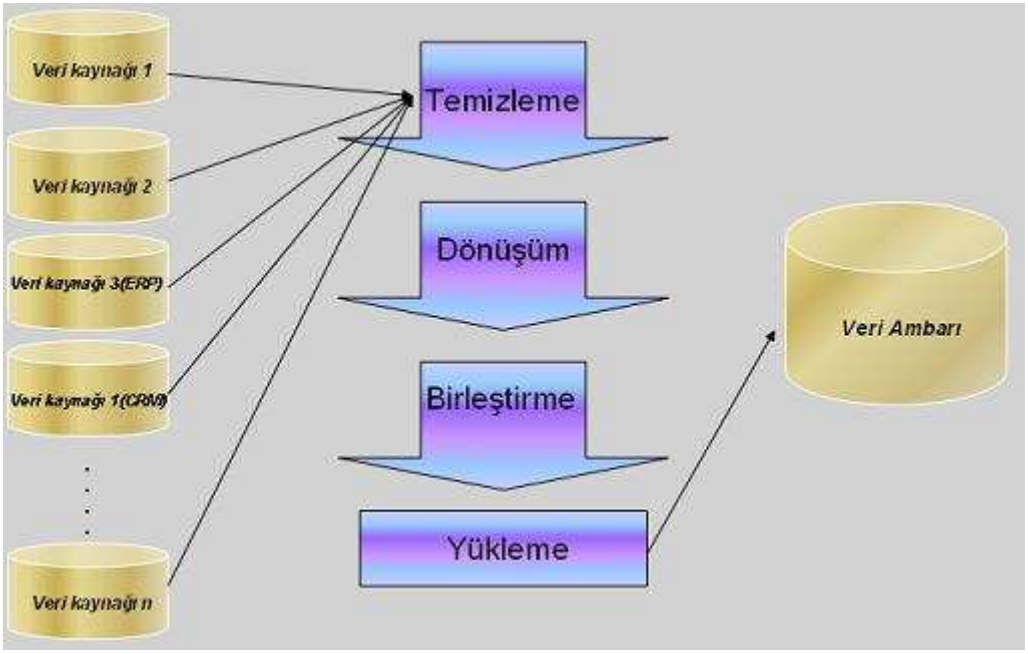
- PostgreSQL
- MySQL
- Oracle
- Sybase
- BerkeleyDB
- Firebird

Başlıca veri tabanı dilleri:

- SQL
- PL/SQL
- Tcl

2.2 Veri Ambarı (Data Warehouse)

Veri ambarları birçok kaynaktan toplanan ve ortak bir format altında birleştirilen bilgi deposudur. Bir işletme sahip olduğu verileri karar destek amacıyla kullanabilmek için bir veri ambarına ihtiyaç duyar. Veri ambarı, bir zaman boyutu içersinde analitik işlemlerin yapılmasını sağlamak için gerekli bilgi temelini sağlar. Özellikle günümüzdeki rekabet koşulları altında doğru kullanıma sahip veri ambarları, şirketler için çok kıymetli yapılardır. Müşteri hakkında daha çok bilgiye erişerek veya eldeki mevcut verilerden yeni tahminler yürüterek, müşterinin ihtiyaçları kolayca öğrenilebilir. Buda şirketimiz için önemli bir avantajdır. Birçok firma farklı tipte veriler toplamakta ve bu verileri veritabanlarında saklamaktadır. Böyle verileri birleştirmek ve veriler arasında etkili iletişimi kurmak çok istenen fakat oldukça zor bir süreçtir. Ticari bir uygulamaya uygun veri ambarını düşünecek olursak, müşteriler, bayiler, ürünler, zaman vb. gibi konularda farklı kanallardan gelecek bilginin veri ambarında tek bir formatta olması gerekir. Veri ambarlarında bulunan verilerin önemli özelliklerinden birisi de temiz ve dönüşümden geçmiş olmasıdır. Çünkü veriler farklı kaynaklarda farklı biçimde saklanabilirler. Ayrıca aynı veri farklı biçimde de temsil ediliyor olabilir. Örneğin, bir uygulamada tarihler gg/aa/yy biçiminde yer alırken bir başka veri kaynağında aa.gg.yy biçiminde olabilir. Bu gibi veriyi veritabanından doğrudan kaydetmek sorunlara neden olabilir. Söz konusu veri, veri ambarına aktarılmadan önce temizlenmeli, gerekli dönüşümleri yapılmalı ve birleştirilmeli daha sonra ise veri ambarına yüklenmelidir. Bu sayede veri ambarındaki verinin hazır bir şekilde bulunması sağlanmış olur. Aşağıdaki Şekil 2.3 bir veri ambarının kuruluş sürecini göstermektedir.



Şekil 2.3 Veri ambarı kuruluş süreci

Veri ambarında toplanan veriler kullanılmaya hazır olduklarından kullanıcılar amaçlarına göre doğrudan veri ambarına erişerek sorgulama yapabilirler. Bazı ilişkisel veritabanları çok boyutlu analizlere olanak tanıyan analiz araçlarına sahiptir. İlişkisel veritabanlarının dili olan SQL içinde bu tür analizleri yapmaya olanak sağlayan komutlar yer almaktadır (Özkan, 2008). Verinin elde edilmesinden sonraki aşama analiz aşamasıdır. Veri madenciliği dediğimiz bölümde bu noktadan itibaren devreye girmektedir.

Veri ambarlarının önemli özellikleri aşağıda yer almaktadır.

- Bütünleşik olma
- Konuya yönelik olma
- Zaman boyutu olma
- Sadece okunabilir olma

2.2.1 Bütünleşik Olma

Veri ambarları değişik farklı bölgelerde bulunan veya değişik özellikleri olan veritabanlarının entegrasyonu sonucu oluşturulur. Bu entegrasyon SQL Server'da DTS (Data Transformation Services) aracı ile yapılır.

2.2.2 Konuya Yönelik

Veri ambarları işletmenin belli başlı amaçlarına ya da konularına yönelik olmalıdır. Konuya yönelik olmasının anlamı, veri ambarının karar destek sürecinde kullanılmayacak veriyi içermemesidir. Örneğin, sigorta şirketine dair bir ambar için konuşacak olursak; veri ambarına konacak bilgiler bireysel mesuliyet, hayat ve kazanın yerine müşteri, poliçe ve sigorta primi olarak ayırılacaktır.

2.2.3 Zaman Boyutu

Veri ambarlarının kullanım amaçları, veri madenciliği analizleri yardımıyla geçmişten alınan verilerin yardımıyla geleceğe dair fikir yürütmektir. Dolayısıyla veri ambarındaki her yapı bir şekilde zaman içerir. Veri ambarlarında saklanan verilerdeki zaman boyutu yüklenen verilerin dönemlerinin birbirlerine karışmasına engel olur. Genellikle veri ambarlarının zaman olarak uzunluğu 5-10 yıl olarak kabul edilir.

2.2.4 Sadece Okunabilen

Veri ambarının diğer bir özelliği, veri ambarında yer alan verinin sadece okunabilir olmasıdır.

Veri ambarındaki veriler güncelleştirilemez ve de silinemezler.

2.3 Veritabanı ile Veri Ambarları Arasındaki Fark

Veritabanları ile veri ambarları arasındaki farkı açıklayacak olursak; veri tabanları genellikle günlük işlemlerde sıklıkla kullanılmaktadır. Veri tabanlarında, günlük veri giriş çıkışı çok fazla yapılmaktadır. Veri tabanlarının kullanıcıları çoğunlukla şirkette normal çalışan personellerdir.(Muhasebe, halkla ilişkiler, vs.) Veri ambarlarının kullanıcıları ise analistler ve şirkete dair karar alabilme yetkisine sahip kişilerdir. Örnek verecek olursak; müşteri ilişkileri yönetimi ile ilgili genellemeler yaparak, genel bir müşteri karakteristiği ortaya çıkarılarak üretilen mal veya hizmetin değiştirilmesi. Veya satın alma tercihi, satın alma zamanı, harcama istekleri gibi müşteri satın alma biçimlerini analiz ederek müşteri odağını arttırmak. Veri ambarlarının kullanıcı sayısı (<100) veri tabanlarını kullanıcı sayısına göre çok daha azdır.

2.4 Veri Madenciliği

Veri madenciliği konusunda çeşitli tanımlar yapılmaktadır. Basit bir tanım yapmak gerekir ise veri madenciliği, büyük ölçekli veriler arasından bilgiye ulaşma, bilgiyi madenleme işidir. Ya da bir anlamda büyük veri yığınları içerisinden gelecek ile ilgili tahminde bulunabilmemizi

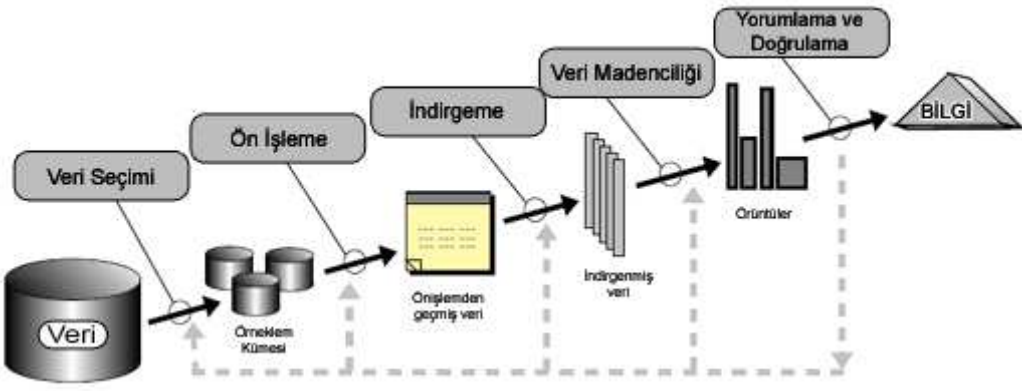
sağlayabilecek bağıntıların bilgisayar programı kullanarak aranmasıdır. Bunun anlamı, veri madenciliği bir kurumda üretilen tüm verilerin belirli yöntemler kullanılarak var olan ya da gelecekte ortaya çıkabilecek gizi bilgiyi su yüzüne çıkarabilecek bir süreç olarak değerlendirilebilir.

Yukarıdaki tanıma ek olarak veri madenciliği için yapılmış farklı tanımlar aşağıda yer almaktadır.

Frawley veri madenciliğini "Daha önceden bilinmeyen ve potansiyel olarak yararlı olma durumuna sahip verinin keşfedilmesi" olarak tanımlamıştır. Berry ve Linoff bu kavrama "Anlamlı kuralların ve örüntülerin bulunması için geniş veri yığınları üzerine yapılan keşif ve analiz işlemleri" şeklinde bir açıklama getirmiştir (Öğüt, 2005). Gartner grubuna göre veri madenciliği, büyük miktarda veri içersinde istatistik, matematik ve örüntü tanıma tekniklerini kullanarak anlamlı korelasyonlar, trendler, örüntüler keşfetme süreci olarak tanımlanmaktadır (Larose, 2005).

Bu tanımlamaları da göz önünde bulundurarak veri madenciliği kavramına şöyle bir yaklaşım getirmek mümkündür:

"Veri madenciliği, geniş veri yığınları içerisinde, yararlı olma potansiyeline sahip, aralarında beklenmedik / bilinmedik ilişkilerin olduğu verilerin keşfedilerek, veri sahibi için hem anlaşılır hem de kullanılabilir bir biçime getirilmesine yönelik geliştirilmiş yöntemler topluluğudur. Bahsi geçen bu yöntemler karar verme sürecinde oldukça etkili rol oynamaktadırlar. Nihayetinde amaç bilgiyi keşfederek ona ulaşmak ve bu yolla fayda sağlamaktır. Veri madenciliği, aynı zamanda bir süreçtir. Veri yığınları arasında, soyut kazılar yaparak veriyi ortaya çıkarmanın yanı sıra, bilgi keşfi sürecinde örüntüleri ayrıştırarak süzmek ve bir sonraki adıma hazır hale getirmek de bu sürecin bir parçasıdır (Öğüt, 2005)".



Şekil 2.4 Bilgi keşfi sürecinde veri madenciliğinin yeri (Öğüt, 2005)

Şekil 2.4’ de bilgi keşfi süreci içersinde veri madenciliğinin yeri gösterilmektedir.

Yukarıda da özetlenmeye çalışıldığı gibi veri madenciliği çalışmaları günümüz bilgi toplumunda kritik bir alan olmaya başlamıştır. Bilişim, Internet ve medya teknolojilerindeki olağan üstü gelişmeler bizleri bir veri okyanusu ile karşı karşıya bırakmıştır. Bu veri okyanusundan bilgiye ulaşmak için bir başka ifade ile balık tutmak için özellikle Avrupa ve ABD de veri madenciliği konusunda birçok araştırma gurubu kurulmuş ve kurulmaktadır. 22 Mayıs 2000 tarihli Time dergisinde yer alan bir yazıda veri madenciliği en sıcak on iş alanından birisi olarak gösterilmiştir (Öğüt, 2005). Online teknoloji dergisi ZDNET’ in 2001 yılındaki haberlerine göre veri madenciliği gelecek on yılın devrimsel gelişmelerinden bir tanesi olarak nitelendirilmiştir. MIT’s Magazine of Technology Review dergisinin Oçak-Şubat 2001 nolu sayısında veri madenciliği dünyayı değiştirecek 10 teknolojiden bir tanesi olarak tanımlanmaktadır (Larose, 2005).

2.4.1 Veri Madenciliğinin Yararları ve Uygulama Alanları

Veri madenciliğinin karar verici için olası yararları aşağıdaki gibi sıralanabilir:

1. Mevcut müşterilerin karar verici tarafından daha iyi tanınmasını sağlayabilir.
2. Özellikle finans sektöründe mevcut müşterileri bölümlere ayırıp, kredi risk davranış modelleri oluşturarak, yeni başvuruda bulunan müşterilere karşı riskin minimize edilmesini sağlayabilir.
3. Finans ve borsa kuruluşlarında stok fiyatları tahminleri, portföy yönetimi yapılabilir.
4. Mevcut müşterilerin ödeme performansları incelenip kötü ödeme performansı gösteren müşterilerin ortak özellikleri belirlenerek, benzer özelliklere sahip tüm müşteriler için yeni risk yönetim politikaları oluşturulabilir.
5. En iyi müşteriler veya müşteri bölümlerinin bulunmasında kullanılabilir. Bulunan bu iyi müşteri bölümlerine yönelik yeni pazarlama stratejileri oluşturulabilir.
6. Kuruluşlar tarafından düzenlenecek çeşitli kampanyalarda mevcut müşteri kitlesinin seçimi ve bu müşterilerin davranış özelliklerine yönelik kampanya şartlarının oluşturulması sağlanabilir.
7. Bankacılık faaliyetlerinde, küçük işletmelere yönelik olarak makine ve ekipman satışı yapan dağıtıcı firmalarla ortak hareket ederek oluşturulacak satış paketleri ile pazarlama stratejileri geliştirilebilir. Mevcut müşteriler üzerinde firma ürünlerinin

çapraz satış kapasitesinin arttırılması sağlanabilir.

8. Veri madenciliği ile mevcut müşteriyi tanıyarak kuruluşların müşteri ilişkileri yönetimlerinde düzenleme ve geliştirmeler yapılabilir. Bu sayede kuruluşun müşterilerini daha iyi tanıyarak müşteri gibi düşünme kapasitelerinin arttırılması sağlanabilir.
9. Günümüzde var olan yoğun rekabet ortamında kuruluşların hızlı ve kendisi için en doğru kararı almalarını sağlayabilir.
10. Kuruluşlar veri analizi ile müşterilerini kişiselleştirilmiş ürün ve hizmetler hakkında bilgilendirebilirler.
11. Veri madenciliği ile kuruluşların müşteriyle bütünleşmiş satış politikaları oluşturması sağlanabilir.
12. Laboratuar veya bilgisayar ortamında sistemlerin benzetimi ve analizi sürecinde elde edilen yüksek miktarda bilimsel veriler anlamlandırılabilir.
13. Sağlık alanında tarama testlerinden elde edilen verileri kullanarak çeşitli kanserlerin ön tanısı, kalp verilerini kullanarak kalp krizi riskinin tespiti, acil servislerde hasta semptomlarına göre risk ve öncelikler tespit edilebilir.
14. Öğrenci işlerinde veriler analiz edilerek öğrencilerin başarı ve başarısızlık nedenleri, başarının arttırılması için hangi konulara ağırlık verilmesi gerektiği, üniversite giriş puanları ile okul başarısı arasındaki bir ilişkinin var olup olmadığı belirlenebilir.
15. Bir çok web sunucusu veya online servisten kullanıcı erişim desenlerinin analizi ve keşfi yapılabilir.
16. Dokümanlar arasında elle bir tasnif gerektirmeden benzerlikler hesaplanabilir.

Veri madenciliği günümüzde yaygın bir kullanım alanı bulmaktadır. Örneğin, pazarlama, bankacılık ve sigortacılık gibi alanlarda ve elektronik ticaret ile ilgili alanlarda yaygın şekilde kullanılmaktadır. Bunlar kullanım yerlerine göre aşağıdaki gibi sınıflandırılmıştır: (Özkan, 2008)

Pazarlama

- Müşterilerinin satın alma alışkanlıklarının belirlenmesi
- Müşterilerin demografik özellikleri arasındaki bağlantıların ortaya konması

- Mevcut müşterilerin elde tutulması, yeni müşterilerin kazanılması
- Pazar sepeti analizi
- Müşteri ilişkileri yönetimi
- Müşteri değerlendirme
- Satış tahmini

Bankacılık

- Farklı finansal göstergeler arasındaki gizli korelasyonların tespiti
- Kredi kartı dolandırıcılıklarının tespiti
- Kredi kartı harcamalarına göre müşteri guruplarının belirlenmesi
- Kredi taleplerinin değerlendirilmesi

Sigortacılık

- Yeni poliçe talep edecek müşterilerin tahmin edilmesi
- Sigorta dolandırıcılıklarının tespit edilmesi
- Riskli müşteri guruplarının belirlenmesi

Elektronik Ticaret

- Saldırıların çözümlenmesi
- e-CRM uygulamalarının yönetimi
- WEB sayfalarına yapılan ziyaretlerinin çözümlenmesi
- Kullanıcı davranışlarına göre web sitesinin yenilenmesi

Telekomünikasyon

- İletişim ağlarında sorunlu bölgelerin tespiti
- Kaçak hat kullanımlarının belirlenmesi
- Kullanıcı davranışlarının belirlenmesi
- Müşteri davranışlarına göre yeni hizmetlerin sunulması

Tıbbi Araştırmalarında

- DNA içersindeki genlerin sıralarının belirlenmesi

- Protein analizlerinin yapılması
- Hastalık haritalarının hazırlanması
- Hastalık tanıları
- Sağlık politikalarına yön verilmesi

Bunların dışında da veri madenciliğinin faydalı olabileceği ve kullanılabileceği alanlardan bazıları şunlardır:

- Taşımacılık ve ulaşım
- Turizm ve otelcilik
- Belediyeler
- Eğitim
- Bilim ve mühendislik

2.4.2 Veri Madenciliğinin Ortaya Çıkışı Ve Gelişim Süreci

1960' lı yıllarda bilgisayarların veri analizi amacıyla kullanılmaya başlanmasıyla birlikte veri madenciliği kavramsal olarak ortaya çıktı. O dönemlerde yeterince uzun taramalar yapılarak istenilen verilere ulaşmanın mümkün olacağına inanılıyordu. Bu işleme veri madenciliği yerine önceleri veri taraması (data dredging), veri yakalanması (data fishing) gibi isimler verildi (Öğüt, 2005).

1990' lı yıllara gelindiğinde veri madenciliği ismi, bilgisayar mühendisleri tarafından ortaya atıldı. Bilgisayar mühendislerinin amacı geleneksel istatistiksel modeller yerine, veri analizlerini algoritmik bilgisayar modelleri tarafından yapılabileceğini göstermekti. Bundan sonra ise veri madenciliğine çeşitli yaklaşımlar getirilmeye başlandı. Bu yaklaşımların kökeninde istatistik, makine öğrenimi (machine learning), veritabanları, otomasyon, pazarlama, araştırma gibi disiplinler ve kavramlar yatmaktaydı (Öğüt, 2005).

İstatistik, süre gelen zaman içerisinde verilerin değerlendirilmesi ve analizleri konusunda hizmet veren bir yöntemler topluluğuydu. Bilgisayarların veri analizi için kullanılmaya başlanmasıyla birlikte istatistiksel çalışmalar hız kazandı. Hatta bilgisayarın varlığı daha önce yapılması mümkün olmayan karmaşık istatistiksel araştırmaların yapılmasını mümkün kıldı. 1990' lar dan sonra istatistik, veri madenciliği ile ortak bir platforma taşındı. Verinin, yığınlar içerisinden çekip çıkarılması ve analizinin yapılarak kullanıma hazırlanması sürecinde veri

madenciliği ve istatistik sıkı bir çalışma birlikteliği içine girmiş bulundular. Öyle ki veri madenciliğini istatistikçiler elleriyle yapar denilmektedir (Öğüt, 2005).

Bunun yanı sıra veri madenciliği, veri tabanları ve makine öğrenimi disiplinleriyle birlikte yol aldı. Günümüzdeki yapay zeka çalışmalarının temelini oluşturan makine öğrenimi kavramı, öğrenme kavramıyla yakından ilişkilidir. Öyle ki makine öğrenimini anlayabilmek için öncelikle öğrenme kavramının bilinmesi gerekmektedir. Simon öğrenmeyi "zaman içinde yeni bilgilerin keşfedilmesi yoluyla davranışların iyileştirilmesi süreci" olarak tanımlamaktadır. Makine öğrenmesi ise öğrenme işleminin bilgisayarlar tarafından gerçekleştirilmesinin sağlanmasıdır. Diğer bir deyişle makine öğrenmesi bilgisayarların bir olayla ilgili bilgileri ve tecrübeleri öğrenerek gelecekte oluşacak benzeri olaylar hakkında kararlar verebilmesi ve problemlere çözümler üretebilmesidir (Öztemel, 2003).

Önceleri makineler, insan öğrenimine benzer bir yapıda inşa edilmeye çalışıldı. Ancak 1980' lerden sonra bu konuda yaklaşım değişti ve makineler daha spesifik konularda kestirim algoritmaları üretmeye yönelik inşa edildi. Bu durum ister istemez uygulamalı istatistik ile makine öğrenim kavramlarını, veri madenciliği altında bir araya getirdi.

2.4.3 Veri Madenciliği Süreci

Veri madenciliği tanımını incelerken veri madenciliğinin farklı disiplinlerden yararlandığını vurgulamıştık. Veri madenciliği veri analizinde yapay zeka, istatistik, veri tabanı teknolojisi ve veri ambarlarından önemli ölçüde yararlanmaktadır. Çünkü veri madenciliği yalnızca hazır verinin analizinden ibaret değildir. Veri madenciliği, veri analizinin yanı sıra araştırılacak problemle ilgili veritabanının hazırlanması, verinin ilgili veri tabanlarından sorgulanması, verinin analize hazır hale getirilip analiz sonucunda elde edilen enformasyonun bilgiye dönüştürülmesi işlemlerini içeren uzun bir süreçtir. Veri madenciliği sürecini karar probleminin belirlenmesi, veri önişleme, veri analizi ve sonuçların yorumlanması şeklinde kabaca dört bölüme ayırabiliriz.

2.4.3.1 Karar Probleminin Belirlenmesi

Veri madenciliği çalışması büyük veri tabanları içersinde rastgele olarak bir enformasyon arama işlemi olarak görülmemelidir**.** Veri madenciliği çalışması yapmak için öncelikle çalışmanın amacı açık bir şekilde tanımlanmalıdır. Çalışma amacı, sorun üzerine odaklanmış ve açık bir dille ifade edilmiş olmalı, elde edilecek sonuçların başarı düzeylerinin nasıl ölçüleceği tanımlanmalıdır. Sorun ile tam örtüşmeyen bir veri madenciliği çalışması, sorunu

çözmeye yetmeyeceği gibi sonuçta başka problemlerin de ortaya çıkmasına neden olabilecektir. Amaç belirlendikten sonra, veri hazırlama işlemine geçilir. Amaca uygun veriler seçilir ve bu veriler analize hazır hale getirilir. Yüzlerce ya da binlerce değişkenlerle araştırma yapmak yerine ilgili değişkenlerle çalışılarak zaman kazanılmış olur. Ayrıca karar probleminin belirlenmesi aşamasında yanlış kararlarda katlanılacak olan maliyetlere ve doğru kararlarda kazanılacak faydalara ilişkin öngörülere de yer verilmelidir.

2.4.3.2 Veri Ön İşleme (Data Preprocessing)

Veri kalitesi veri madenciliğinde anahtar bir konumdadır. Veri madenciliğinde güvenirliliğin arttırılması için veri ön işleme aşamasına ihtiyaç duyulmaktadır. Aksi halde hatalı girdi verileri bizi hatalı sonuçlara götürebilir.

Veri ön işleme aşağıdaki sebeplerden dolayı verilere uygulanmaktadır (Oğuzlar, 2004).

1. Veriler üzerinde herhangi bir analiz türünün uygulanmasını engelleyecek veri problemlerinin çözümü
2. Verilerin doğasının anlaşılması ve anlamlı veri analizinin başarılması
3. Verilen bir veri kümesinden daha anlamlı bilginin çıkarılması

Bu adımda yapılacak her işlem, en son adımda verilecek olan kararı etkileyecektir. Veri ambarından toplanan veri hatalar, aşırı değerler içerebilir. Veriler içersinde uyumsuzluklar, hatta eksiklikler olabilir. İşte veri ön işleme aşamasında verideki kusurlar, eksiklikler ve hatalar giderilerek veriler analiz aşamasına hazırlanır.

Modelin kurulması aşamasında ortaya çıkacak sorunlar, bu aşamaya sık sık geri dönülmesine ve verilerin yeniden düzenlenmesine neden olacaktır. Bu durum verilerin hazırlanması ve modelin kurulması aşamaları için, bir karar vericinin veri keşfi sürecinin toplamı içerisindeki enerji ve zamanının % 50 - % 85’ ini harcamasına neden olmaktadır. Veri ön işleme teknikleri veri madenciliğinden önce uygulanarak elde edilen sonuçların kalitesi ve veri madenciliği için harcanacak zaman arttırılmış olur.

Veri ön işlemlerini veri temizleme, veri birleştirme, veri dönüştürme ve veri indirgeme olarak dört bölümde inceleyebiliriz.

2.4.3.2.1 Veri Temizleme (Data Cleaning)

Gerçek hayatta elde edilen veriler içersinde mutlaka bazı sorunlar bulunur. Bu sorunlar içersinde en çok karşılaşılan sorunlar veri içersinde kayıp değerlerin bulunması, verilerin aşırı

değerler (çok yüksek veya çok düşük) içermesi, veri içersinde uyumsuzlukların bulunmasıdır. İşte veri temizleme aşamasında, kayıp verilerin, aykırı değerlerin teşhis edilmesi ve verilerdeki uyumsuzlukların giderilmesi gibi işlemler gerçekleştirilir.

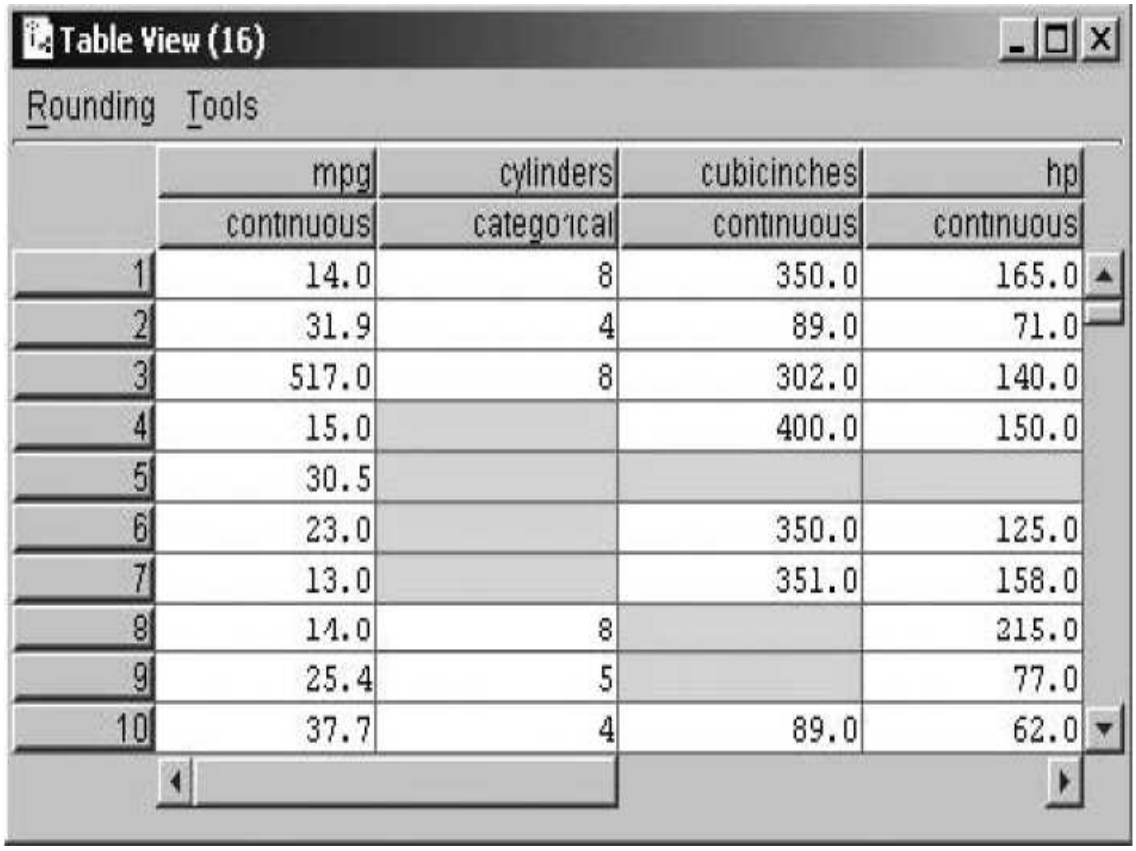
2.4.3.2.1.1 Kayıp Değerler

Kayıp değerler çok sayıda sebepten kaynaklanabilir. Kayıp değerler, veri giriş hatalarından, eksik bilgi toplanmasından, bilginin alındığı birimin cevap vermekten kaçınmasından ya da birimin o bilgiye sahip olmamasından kaynaklanabilir.

İlgilenilen değişkenler veri tabanında bulunmayabilir. Örneğin, satışlara ilişkin bir veri tabanında müşteri bilgileri yer almayabilir.

Kayıp değerlerden veri setini arındırmak için yapılabilecek bazı işlemler şunlardır (Bilen, 2004; Oğuzlar, 2004) :

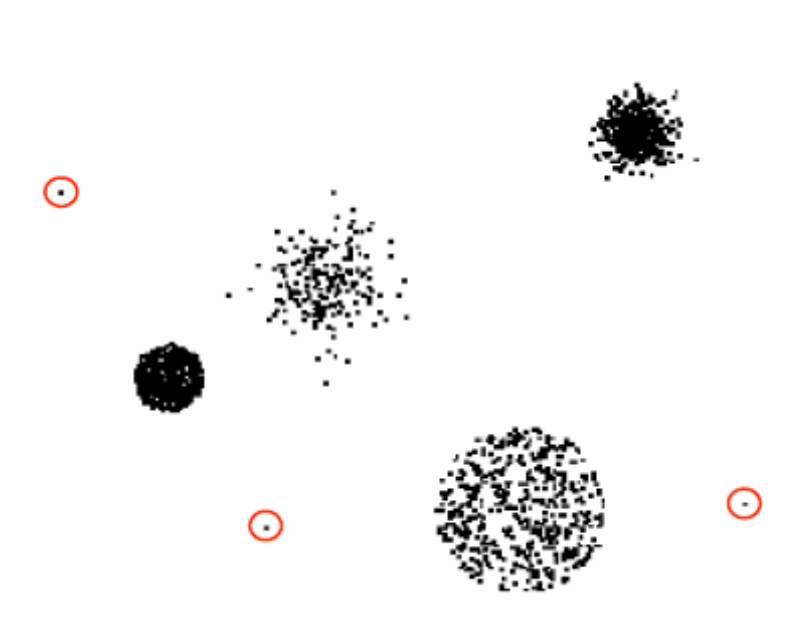
1. Eğer maliyetli değilse ve uzun zaman almayacaksa eksik değerin ait olduğu birime başvurmak.
2. Eksik değerin bulunduğu birim yada birimler veri setinden çıkartılabilir. Bu yöntem veri kaybına neden olduğu için genellikle tercih edilmez.
3. Eksik değer eğer tamamlanabiliyorsa, eksik değerin tamamlanması. Örneğin, askerliğini yapmış bir kişinin cinsiyeti boş ise cinsiyetin erkek olarak tamamlanması.
4. Değişkenin genel eğiliminin gösterdiği değeri kayıp değerin yerine atamak. Örneğin, değişkenin ortalaması, modu ya da medyanının kayıp değer yerine atanması. Aynı kredi risk kategorisine giren müşteriler için ortalama gelir değeri eksik değerler yerine kullanılabilir.
5. Var olan verilere dayalı olarak kayıp değerin tahmin edilmesi. Burada regresyon veya karar ağaçları gibi teknikler kullanılarak kayıp değerler tahmin edilebilir.
6. Kayıp değer yerine “Bilinmeyen” gibi global bir sabitin atanması.



Şekil 2.5 Kayıp değerler bulunan bir veri örneği (Larose, 2005)

2.4.3.2.1.2 Aşırı Değerler (Outlier)

Aşırı değerler, veri setindeki diğer birimlerden ciddi şekilde farklı olan veri şıklarının gösterdiği karakteristiktir. Aşırı değerler, veri madenciliği sürecinin analiz aşamasında regresyon, kümeleme analizi gibi uygulamalarda sorunlara neden olurlar. Bu nedenle aşırı değerlerin veri setinde bulunması istenmeyen bir durumdur.



Şekil 2.6 Aşırı değerlerin bulunduğu bir veri örneği

Veri setinde bulunan aşırı değerleri bulmak için kullanılabilecek bazı yöntemler aşağıda yer almaktadır (Bilen, 2004; Oğuzlar, 2004).

1. Veriler küçükten büyüğe sıralanır. Sıralanmış veri bölmelere ayrılarak aşırı değerler bulunabilir.
2. Veri seti kümeleme analizi ile kümelere ayrılır. Benzer değerler aynı grup veya küme içinde yer alırken, aykırı değerler kümelerin dışında yer alırlar.
3. Regresyon yöntemiyle veri setindeki verilere bir fonksiyon uydurularak aykırı değerler bulunabilir. Uydurulan bu fonksiyona uymayan değerler aykırı değerlerdir.
4. Değişkenlere ait kutu diyagramları çizilir. Kutu diyagramlarından aşırı değerler gözlemlenebilir.
5. Değişkenlerin grafikleri aracılığıyla aşırı değerler bulunabilir.
6. Temel bileşenler analizinde elde edilen ilk iki temel bileşenin serpilme diyagramı incelenerek aşırı değerler bulunabilir.

Aşırı değerler bulunduktan sonra yapılacak işlem aşırı değerlerin arındırılmasıdır. Aşırı değerlerden arındırma işlemlerinden bazıları şunlardır (Bilen, 2004) :

1. Aşırı değerlerin bulunduğu birim sayısı çok fazla değilse (buna çalışmayı yapan uzman karar verebilir) bu birimler analiz dışında tutulabilir.

2. Aşırı değerler yerine değişkenin genel ortalaması kullanılabilir.
3. Aşırı değer bulunan değişkenler dışarıda tutularak regresyon, karar ağacı gibi modelleme yöntemi kullanılarak model kurulur. Kurulan modele göre aşırı değerlerin yerine geçecek değer tahmin edilir.

2.4.3.2.1.3 Uyumsuz (Inconsistent) Veriler

Veri setinde bulunan değişkenlerin alabileceği bazı şıkların birden çok farklı biçimde gösterilmesi, aynı birimin iki değişkeninde almış olduğu değerlerin çelişkili olması ya da değişkene ait olmayan bir şıkkın veri setinde yer alması uyumsuz veriye örnek olarak gösterilebilir (Bilen, 2004).

Tablo 1.1 Uyumsuz veri örneği (Bilen, 2004)

Şehir	Bölge	Yaş
Bursa	Marmara	-15
İstanbul	Marmara	15
Ankara	Doğu Anadolu	26

2.4.3.2.2 Veri Birleştirme (Data Aggregate)

Veriler farklı veri tabanlarında bulunabilirler. Bu durumda farklı veri tabanlarında bulunan verilerin tek bir çatı altında - ki bu genellikle veri ambarıdır – birleştirilmeleri gerekir. İşte bu işleme veri birleştirme adı verilir.

Veri madenciliğinde kullanılacak verilerin farklı kaynaklardan toplanması, doğal olarak veri uyumsuzluklarına neden olacaktır. Bu uyumsuzlukların başlıcaları farklı zamanlara ait olmaları, güncelleme hataları, veri formatlarının farklı olması, kodlama farklılıkları (örneğin bir veri tabanında cinsiyet özelliğinin e/k, diğer bir veri tabanında 0/1 olarak kodlanması), farklı ölçü birimleri ve varsayım farklılıklarıdır. Ayrıca verilerin nasıl, nerede ve hangi koşullar altında toplandığı da önem taşımaktadır. Güvenilir olmayan veri kaynaklarının kullanımı tüm veri madenciliği sürecinin de güvenilirliğini etkileyecektir.

Bu nedenlerle, iyi sonuç alınacak veri madenciliği çalışmaları ancak iyi verilerin üzerine kurulabileceği için, toplanan verilerin ne ölçüde uyumlu oldukları bu adımda incelenerek değerlendirilmelidir.

2.4.3.2.3 Veri Dönüştürme (Data Transformation)

Veriyi bazı durumlarda veri madenciliği analizlerine aynen katmak uygun olmayabilir. Değişkenlerin ortalama ve varyansları birbirinden önemli ölçüde farklı olduğu durumlarda büyük ortalama ve varyansa sahip değişkenlerin diğerlerin üzerindeki baskısı daha fazla olur ve onların rollerini önemli ölçüde azaltır. Ayrıca veri setinde farklı ölçü birimleri kullanılarak elde edilen değişken değerlerinin birimlerinden arındırılması, aşırı değerlerin etkisinin azaltılması, nitel değişkenlerin nicel değişkenlere dönüştürülmesi gibi nedenlerle de veri dönüştürme işlemi kullanılır. Veri dönüştürme ile analize dahil edilecek değişkenlerin, yapılacak analizlerin varsayımları sağlaması sağlanabilir. Veri dönüştürme işlemlerinden bazıları şunlardır:

1. Düzleştirme (Smoothing): Genellikle aşırı değerleri arındırmak için kullanılır. Kümeleme, regresyon yöntemlerini içerir.
2. Veri aşırı detaylıysa, veriyi özet bir hale getirmek için kullanılır.
3. Aylık gelir düzeyi gibi düşük, orta, yüksek gibi sınıflanabilecek olan sürekli değişkenlerin genelleştirilerek nitel hale dönüştürülmesi.
4. Verilerin normalleştirme işlemlerinden geçirilerek, 0-1 ya da 1-1 aralıklarına indirgenmesi.

2.4.3.2.3.1 Verilerin Normalleştirilmesi

Verilerin analizler için uygun bir hale getirilmesi için yapılan dönüştürme işlemlerinden en sık kullanılanı verilerin normalleştirilmesidir. Veriler normalleştirilme işleminden geçirilerek 0-1 ya da 1-1 aralıklarına indirgenmiş olurlar. Verilerin normalleştirilmesinde kullanılan bazı dönüşümler: z skorlarına dönüştürme, $-1 \le x \le 1$, $0 \le x \le +1$ aralıklarına indirgeme, ortalama 1 olacak şekilde indirgeme, standart sapma 1 olacak şekilde indirgeme, maksimum değer bir olacak şekilde indirgemedir.

2.4.3.2.3.1.1 Z Skorlarına Dönüştürme

Bu yöntem, oransal ve aralık ölçekli veriler söz konusu olduğunda verilerin çok değişkenli normal dağılım gösterdiği varsayımıyla verilere uygulanan bir yöntemdir. Ver madenciliği çalışmalarında veri sayımız çok büyük olacağından değişkenlerin çok değişkenli normal dağılım gösterdiği varsayılabilir.

$$z_i = \frac{x_i - \bar{x}}{s} \tag{2.1}$$

2.4.3.2.3.1.2 $-1 \le x \le 1$ Aralığına İndirgeme

Eğer veri seti heterojen bir yapıda ve aşırı değerler söz konusu ise bu yöntem tercih edilir. Dönüştürülecek değişkenlerde + ya da – değerler var olduğunda ve $|x_{\min}| \le x_{\max}$ olduğu durumlarda uygulanır. Dizideki en büyük değer $x_{\max}$ ise, indirgeme formülü şöyle olur.

$$x_i^{'} = \frac{x_i}{x_{\max}} \tag{2.2}$$

2.4.3.2.3.1.3 $0 \le x \le 1$ Aralığına İndirgeme

Veri seti heterojen bir yapıda ve aşırı değerler söz konusu ise değişkenlerin değerleri 0 ile 1 aralığına dönüştürülebilir. Dizideki en büyük değer x_{max}, dizideki en küçük değer x_{min} ve rank $R = x_{max} - x_{min}$ olmak üzere indirgeme şöyle yapılır:

$$x_i^{'} = \frac{x_i - x_{\min}}{R} \tag{2.3}$$

2.4.3.2.3.1.4 Ortalama 1 Olacak Biçimde İndirgeme

Oluşturulacak olan indirgenmiş değişkenin ortalamasının pozitif ve 1 olması gerektiğinde uygulanan bir yöntemdir. Dönüştürme işlemi şu formülle yapılır:

$$x_i^{'} = \frac{x_i}{\bar{x}} \tag{2.4}$$

Eğer ortalama sıfıra eşit ise formül şöyle olur:

$$x_i^{'} = \frac{x_i + 1}{\bar{x} + 1} \tag{2.5}$$

2.4.3.2.3.1.5 Standart Sapma 1 Olacak Şekilde İndirgeme

Eğer indirgenmiş değişkenin standart sapmasının 1 olması isteniyorsa, bu yöntem tercih edilir. İndirgeme formülü şöyledir:

$$x_i^{'} = \frac{x_i}{s_x} \tag{2.6}$$

Orijinal değişkenin standart sapması 0 ise, bu dönüşüm uygulanamaz. Dönüşümün şart olduğu düşünülürse, diğer yöntemlerden uygun olan bir yöntem seçilir.

2.4.3.2.3.1.6 Maksimum Değer Bir Olacak Şekilde İndirgeme

İndirgenmiş değişkenin maksimum değerinin 1 olması isteniyorsa bu yöntem tercih edilir. İndirgeme formülü şöyledir:

$$x_i^{'} = \frac{x_i}{x_{\max}} \tag{2.7}$$

Eğer dizide maksimum değer sıfır ise formül şöyledir.

$$x_i^{'} = \frac{x_i}{|x_{\min}|} + 1 \tag{2.8}$$

2.4.3.2.4 Veri İndirgeme (Data reduction)

Bir veri madenciliği çalışmasında seçilen karar problemi ile ilgili değişkenler, birimler oldukça fazla olabilir. Bu durumda veri indirgeme teknikleri kullanılarak, daha küçük hacimli ve veri kümesinin indirgenmiş bir örneğinin elde edilmesi çalışılır. Veri indirgeme hem değişken sayısının azaltılması hem de birim sayınsın azaltılması olarak anlaşılmalıdır. Veri indirgemesi sayesinde küçük hacimli verilerle daha az bellek ve zaman gereksinimi duyularak, çeşitli kompleks veri madenciliği algoritmaları etkin bir şekilde kullanılabilir.

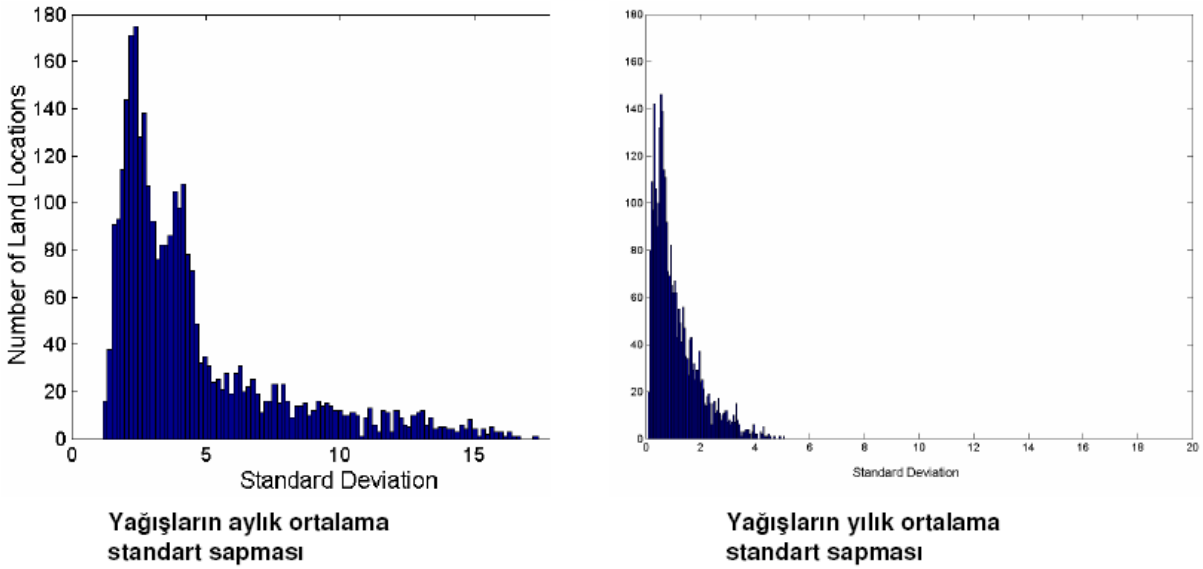
Veri indirgeme yöntemleri aşağıdaki gibi özetlenebilir (Oğuzlar, 2004) :

1. Veri birleştirme veya veri küpü (Data Aggregation)
2. Boyut indirgeme (Dimension Reduction)
3. Veri Sıkıştırma (Data Compression)
4. Kesikli Hale Getirme (Discretization)
5. Örnekleme (Sampling)

2.4.3.2.4.1 Veri Birleştirme

2002-2003 yılları için çeyrek dönemlik satış tutarlarından oluşan bir veri kümesinin bulunduğunu varsayalım. Bu yıllar için yıllık satış tutarları tek bir tabloda toplanarak veri birleştirme işlemi gerçekleştirilmiş olur. Veri birleştirme işlemi sonucunda veri kümesinin hacmi daha küçük bir hale gelmiş olur fakat yapılacak analiz için bir bilgi kaybı söz konusu

olmaz. Ayrıca veri birleştirilmesiyle değişken sayısı azaltılması, veri ölçeğinin değiştirilmesi ve bir araya toplanan verinin daha az değişkenliğe sahip olması söz konusudur.



Şekil 2.7 Avusturalya da yağışların varyasyonu (Tan vd., 2006)

Şekil 2.7’ den de görüldüğü gibi bir araya getirilen verinin standart sapması düşmüştür.

Veri küpleri ise çok değişkenli birleştirilmiş bilginin saklandığı küplerdir. Veri küpleri sayesinde çözümlemeler sadece belirlenen boyutlara göre yapılır. Veriler arasında bir seçme işlemi yapılarak, gereksiz veriler veri tabanından çıkarılır ve boyut azaltılması sağlanabilir. Örneğin, bir firmanın satış tutarları yıllar, satışı yapılan ürünler ve firmanın farklı satış yerleri için aynı küp üzerinde gösterilebilir. Veri küpleri özet bilgiye herhangi bir hesaplama yapmadan hızlı bir şekilde erişilmesini sağlar.

2.4.3.2.4.2 Boyut İndirgeme (Dimension Reduction)

Veri madenciliği yapılacak veri kümesi bazen gereksiz olarak yüzlerce değişken içerebilir. Örneğin bir ürünün satışına ilişkin olarak düzenlenen bir veri kümesinde, tüketicilerin telefon numaraları gereksiz bir değişken olarak yer alabilir. Bu tür gereksiz değişkenler elde edilecek örüntüleri kalitesizleştirebileceği gibi veri madenciliği sürecinin yavaşlamasına da yol açacaktır. Gereksiz değişkenlerin elenmesi amacıyla ileri veya geri yönlü olarak sezgisel seçimler yapılabilir. İleri yönlü sezgisel seçimde orijinal değişkenleri en iyi temsil edecek değişkenler belirlenir. Ardından her bir değişken veya değişkenler grubunun, bu kümeye dahil edilip edilmeyeceği sezgisel olarak belirlenir. Geri yönlü sezgisel seçimde ise öncelikle değişkenlerin tüm kümesi ele alınır. Daha sonra gereksiz bulunan değişkenler kümeden dışlanarak, en iyi değişken kümesi elde edilmeye çalışılır. Boyut indirgeme amacıyla

kullanılacak bir diğer yöntem ise karar ağaçlarıdır. Karar ağaçları ele alınacak çıktı değişkenini en iyi temsil edecek değişken kümesini verecektir (Oğuzlar, 2004).

2.4.3.2.4.3 Veri Sıkıştırma

Veri sıkıştırmada ise orijinal verileri temsil edebilecek indirgenmiş veya sıkıştırılmış veriler, veri şifreleme veya dönüşümü ile elde edilirler. Bu şekilde indirgenmiş veri kümesi, orijinal veri kümesini bir bilgi kaybı olacak biçimde temsil edebilecektir. Bununla beraber bilgi kaybı olmaksızın indirgenmiş veri kümesi elde edilmesine yarayacak bir takım algoritmalar da mevcuttur. Bu algoritmalar bir takım sınırlamalara sahip olduklarından sıkça kullanılamamaktadır. Bununla beraber temel bileşenler analizi gibi yöntemler, bir bilgi kaybına göz yumularak sıkıştırılmış veri kümesi elde edilmesinde kullanışlıdır (Oğuzlar, 2004).

2.4.3.2.4.4 Kesikli Hale Getirme

Kesikleştirme, bazı veri madenciliği algoritmalarının yalnızca kategorik değerleri ele aldığından, sürekli verilerin kesikli değerlere dönüştürülmesini sağlar. Bu şekilde sürekli verilerin kesikli değer aralıklarına dönüştürülmesiyle elde edilen kategorik değerler, orijinal veri değerlerinin yerine kullanılırlar. Bu şekilde düşük düzeyli kavramların, yüksek düzeyli kavramlarla değiştirilmesiyle verilerin indirgenmesi sağlanır. Örneğin, yaş değişkeni 1-15, 16-40, 40+ olacak biçimde daha yüksek kavram düzeyinde ifade edilebilir. Bu şekilde veri indirgemede detay bilgiler kayboluyorsa da, genelleştirilmiş veriler daha anlamlı olacak, daha kolay yorumlanabilecek ve orijinal verilerden daha düşük hacim kaplayacaktır.

2.4.3.2.4.5 Örnekleme

Örnekleme, veri seçimi için üzerinde durulan en temel tekniktir. Örnekleme, sıklıkla hem başlangıç araştırmaları için ve hem de final veri analizleri için kullanılır.

Veri madenciliği çalışmasında kullanılan veri tabanının çok büyük olması durumunda, verinin tamamı ile ilgilenmek oldukça masraflı bir iş olduğu için istatistikçiler ve veri madencileri verinin bir kısmını elde etmeye çalışırlar. Ayrıca burada seçilen örneklem kümesinin tüm popülasyonu temsil edip etmediği de kontrol edilmelidir. Eğer örnek, orijinal veriyi temsil edecek nitelikte ise örnek ile çalışmak bütün veri seti ile çalışmak kadar iyi sonuç verecektir. Örnek veri yaklaşık orijinal veriyle aynı özelliğe sahipse o örnek veri temsil edici veridir. Halen kullanılan işletim sistemleri ve paket programlar ne kadar gelişmiş olursa olsun, çok

büyük veri tabanları üzerinde çok sayıda modelin denenmesi zaman kısıtı nedeni ile mümkün olmamaktadır. Bu nedenle tüm veri tabanını kullanarak bir kaç model denemek yerine, rastgele örneklenmiş bir veri tabanı parçası üzerinde bir çok modelin denenmesi ve bunlar arasından en güvenilir ve güçlü modelin seçilmesi daha uygun olacaktır. Diğer bir deyişle modellerin performansları uygun bir karar yöntemi ile sınanmalıdır.

Orijinal veriden örnek çekme tiplerine göre basit rast gele örnekleme, yer değiştirmeden örnekleme, yer değiştirme ile örnekleme ve katmanlaşma ile örnekleme gibi değişik örnekleme tipleri bulunmaktadır.

Basit rast gele örneklem, herhangi bir elemanın seçilme olasılığı diğer parçaların seçilme olasılığına eşittir. Yer değiştirmeden örnekleme de, herhangi bir eleman seçildiğinde o popülasyondan silinir. Yer değiştirme ile örnekleme de seçilen elemanlar popülasyondan silinmezler. Örneklemede aynı eleman birden fazla seçilebilir. Katmanlaşmış örnekleme de ise veri kümesi tüm veri kayıtlarını kapsayacak şekilde katman olarak adlandırılan parçalara bölünerek her katmandan basit rast gele örnekleme yapılarak gerçekleştirilir.

Şekil 2.8 Örnek seçimi (Tan vd., 2006)

Şekil 2.8’ den anlaşıldığı üzere, örnek boyutu arttıkça çözünürlük artmakta, örneklem popülasyonu iyi temsil etmektedir.

2.4.3.3 Veri Analizi

Modelimize dahil edeceğimiz değişkenler belirlenip, veriler uygun veri hazırlama işlemlerinden geçirilerek analizlere uygun hale getirilir. Daha sonra veri madenciliği algoritmaları arasından karar probleminin çözümüne yönelik yöntem(ler) seçilerek veri madenciliği modellemesi gerçekleştirilir. Veri madenciliği teknikleri bir birilerinden bağımsız kullanılabilecekleri gibi bir birileriyle etkileşim içersinde de kullanılabilirler. Başka bir değişle, veri madenciliği tekniklerinden birisinin çıktısı diğer bir veri madenciliği tekniğinin

girdisi olabilir. Örneğin, temel bileşenler analizi sonucu elde edilen skor değerleri kümeleme analizi, regresyon analizi için girdi oluşturabilir. Kümeleme analiziyle oluşturulan kendi içlerinde homojen fakat kendi aralarında heterojen gruplarsa diğer veri madenciliği teknikleri için de kullanılabilirler. Veri madenciliğinde genellikle veri madenciliği tekniklerinin beraber kullanılması daha etkin sonuçlar vermektedir (Bilen, 2004).

Karar problemi için en uygun modelin bulunabilmesi, olabildiğince çok sayıda modelin kurularak denenmesi ile mümkündür. Bu nedenle veri hazırlama ve model kurma aşamaları, en iyi olduğu düşünülen modele varılıncaya kadar yinelenen bir süreçtir. Yinelenen süreç esnasında başarısız olan örnekler incelenerek bunlar üzerindeki başarının nasıl arttırılabileceği araştırılır. Örneğin standart forma yeni alanlar ekleyerek programa verilen bilgi arttırılabilir; veya olan bilgi değişik bir şekilde kodlanabilir; veya amaç daha değişik bir şekilde tanımlanabilir.

2.4.3.4 Sonuçların Yorumlanması

Veri madenciliği algoritma ya da algoritmaları veriler üzerine uygulandıktan sonra sonuçlar düzenlenerek ilgili yerlere sunulur. Bu sonuçlar mümkün olduğunca görselleştirilerek, son kullanıcıya uygun hale getirilir.

2.4.4 Veri Madenciliği Yöntemlerinin Sınıflandırılması

Veri madenciliği yöntemlerini denetimli, denetimsiz yöntemler olmak üzere iki ana kategoride sınıflandırabiliriz. Denetimli ve denetimsiz yöntemler için kabul görmüş tanımlama aşağıdaki gibidir :

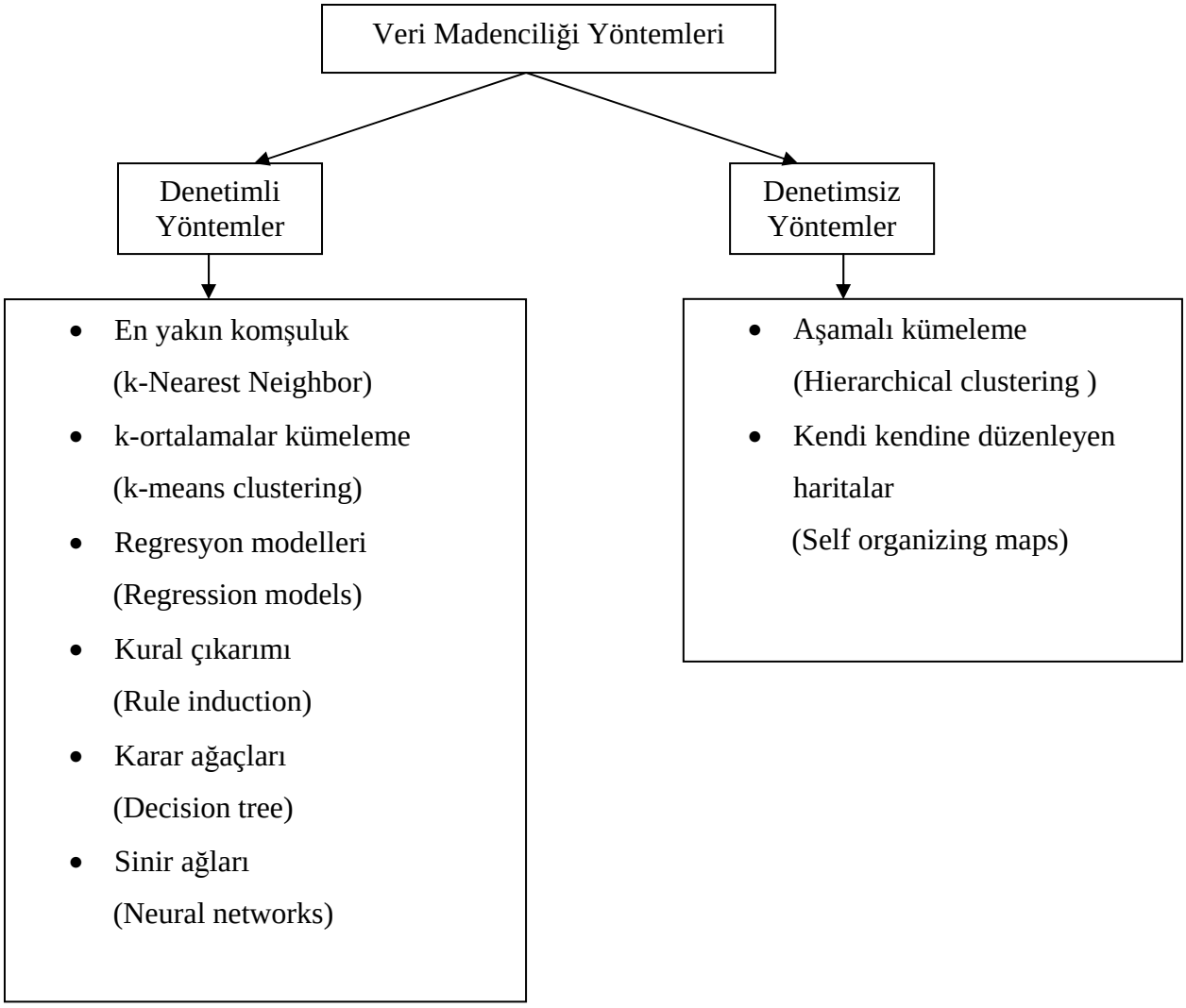
Denetimli (Supervised) : İyi tanımlanmış veya kesin bir hedef olduğunda denetimli ifadesi kullanılır. Denetimli yöntemlerde, bir öğretmen tarafından ilgili sınıflar önceden belirlenen bir kritere göre ayrılarak, her sınıf için çeşitli örnekler verilir. Sistemin amacı verilen örneklerden hareket ederek her bir sınıfa ilişkin özelliklerin bulunması ve bu özelliklerin kural cümleleri ile ifade edilmesidir.

Denetimsiz (Unspervised) : Elde edilmesi istenilen sonuç için özel bir tanımlama yapılmamışsa veya belirsizlik söz konusu ise denetimsiz ifadesi kullanılır. Denetimsiz yöntemlerde, kümeleme analizinde olduğu gibi ilgili verilerin gözlenmesi ve bu verilerin özellikleri arasındaki benzerliklerden hareket ederek sınıfların tanımlanması amaçlanmaktadır.

Denetimli ve denetimsiz ifadeler birbirilerinin tersine karşılık gelmektedir. Denetimsiz yöntemler daha çok veriyi anlamaya, tanımaya, keşfetmeye yönelik olarak kullanılır ve sonraki uygulanacak yöntemler için fikir vermeyi amaçlamaktadır. Denetimli yöntemler ise veriden bilgi ve sonuç çıkarmak için kullanılmaktadır. Denetimsiz bir yöntemle elde edilen bir bilgi veya sonucu, eğer mümkünse denetimli bir yöntemle teyit etmek, elde edilen bulguların doğruluğu ve geçerliliği açısından önem taşımaktadır.

Denetimli ve denetimsiz yöntemlerin arasındaki farkı en güzel anlatacak yöntem, kümeleme analizidir. Örneğin, aşamalı kümeleme analizinde hem birimler hem de değişkenler birbirileriyle değişik benzerlik ölçülerine göre kümelenmesinde küme sayısı baştan verilmemektedir. Küme sayısı baştan belli olmadığı için aşamalı kümeleme analizi denetimsiz bir yöntemdir. Aşamalı olmayan kümeleme analizi yöntemlerinden k-ortalamalar kümeleme yönteminde ise birimlerin uygun sayıda k kümeye ayrılması hedeflenmektedir. Küme sayısı baştan belli olduğu için k-ortalamalar kümeleme analizi denetimli bir yöntemdir. (Koyuncugil, 2007)

Çok kullanılan veri madenciliği yöntemleri denetimli ve denetimsiz yöntemler olmak üzere Şekil 2.9’ daki gibi kategorize edilmiştir.



Şekil 2.9 Veri madenciliği yöntemleri

Diğer başlıca veri madenciliği yöntemleri de aşağıda verilmiştir. (Koyuncugil, 2007)

- Temel bileşenler analizi
- Discriminant analiz
- Birliktelik kuralları
- Bulanık mantığa dayalı yöntemler
- Genetik algoritmalar
- Bayesci ağlar
- Pürüzlü (Rough) küme teorisine dayalı yöntemler

Yukarıdaki yöntemler dışında hibrit yöntemler ve zaman serilerine dayalı yöntemlerden de veri madenciliği yöntemi olarak kullanılmaktadır. Özet olarak, bilgi keşfine yarayan her yöntem veri madenciliği yöntemi olarak kullanılabilmektedir. (Koyuncugil, 2007)

2.4.5 Veri Madenciliği İşlevleri

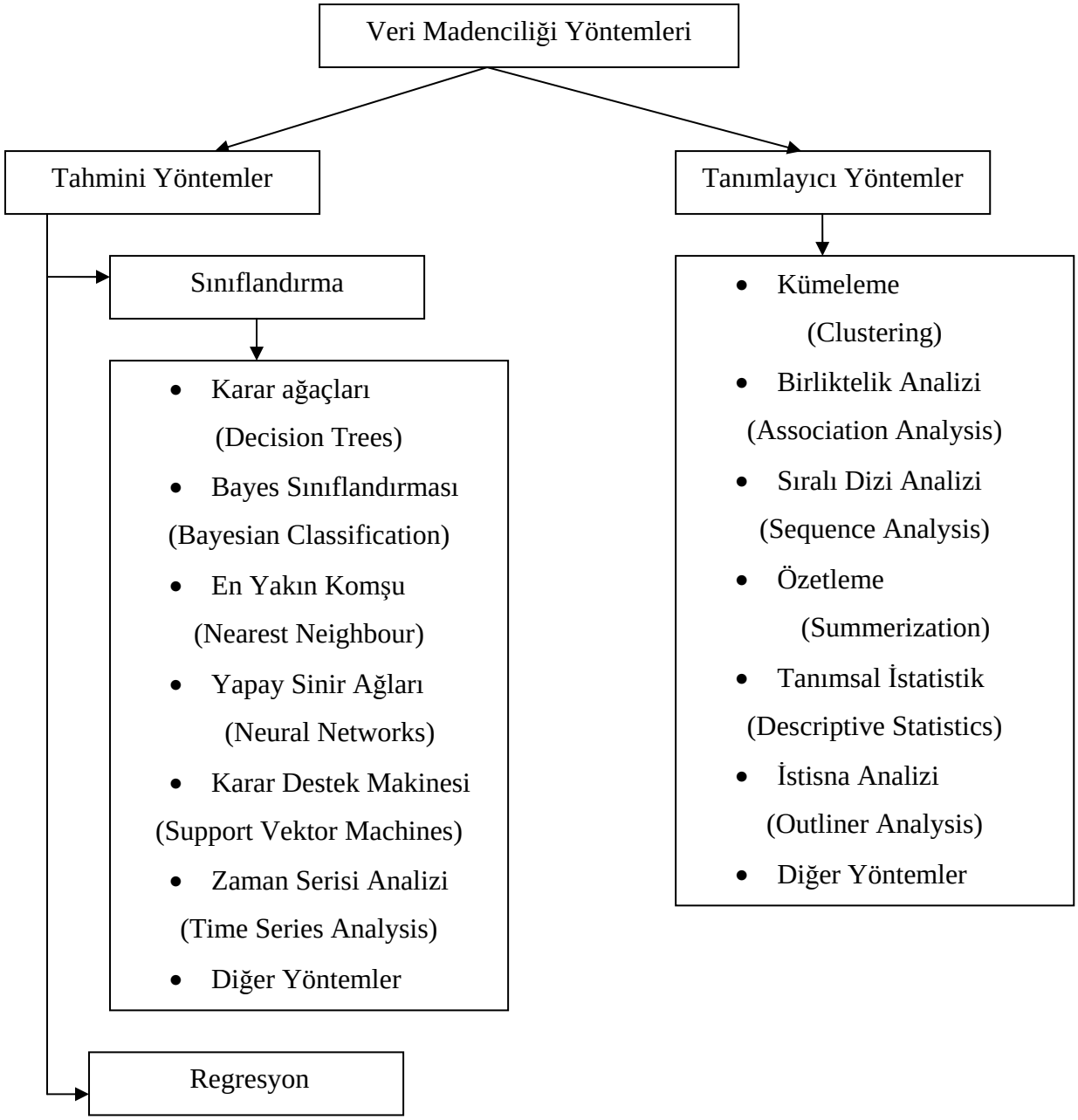
Veri madenciliği yöntemlerine işlevleri açısından bakacak olursak, veri madenciliği yöntemleri iki sınıf altında toplanmaktadır. Bunlar: tanımlayıcı yöntemler (Desciptive Methods), tahmini yöntemlerdir (Predictive Methods).

Tanımlayıcı yöntemler ne olduğu önceden belirlenmemiş bir fikir ya da hipotez olmadan, veri tabanı içersinden gizli desenleri aramayı çalışır. Geniş veri tabanlarında kullanıcının pratik olarak aklına gelmeyecek ve bulmak için gerekli doğru soruları bile düşünemeyeceği bir çok gizli desen, tanımlayıcı yöntemlerle keşfedilebilir. Buradaki asıl amaç, bulunacak desenlerin zenginliği ve bunlardan çıkarılacak bilginin kalitesidir.

Tahmini yöntemlerde, veri tabanından çıkarılan desenler, geleceği tahmin için kullanılır. Bu yöntemler, kullanıcının bazı alan bilgilerini bilmese bile kayıt etmesine izin verir. Tahmini yöntemler, bu boşlukları, önceki kayıtlara bakarak tahmin yoluyla doldurur. Tanımlayıcı yöntemler, verideki desenleri bulmaya yönelikken, tahmini yöntemler, bu desenleri yeni veri nesnelerini bulmak için uygular.

Tahmini yöntemler, bağımsız değişkenlerin bir fonksiyonu olarak bağımlı değişkeni tahmin etmek için kullanılırlar. Burada sınıflandırma (classificarion), regresyon (regression) olmak üzere iki çeşit tahmini yöntem vardır. Bunlardan sınıflandırma kesikli (discrete) bağımlı değişkenler, regresyon sürekli (continuous) bağımlı değişkenler için kullanılır. Örneğin, bir web kullanıcısının online bir kitapçıdan kitap satın alıp almayacağını tahmin etmek isteyelim. Burada alıp alamama gibi iki durumlu nitelik (binary-valued) verisi bulunduğu için sınıflandırma yöntemleri kullanılır. Diğer taraftan, gelecek stok fiyatlarını öngörmek istediğimizde stok fiyatlarının (continuous-valued) sürekli nitelikli veri olmasından dolayı regresyon yöntemi kullanılır. İki tahmini yöntemin de amacı, tahmin edilen ve gerçek bağımlı değişken arasındaki hatayı minimum kılacak bir model kurmaktır. Tahmini modeller, müşterilerin market kampanyalarına cevap verip vermeyeceklerini belirlemede, dünyanın ekosistem karışıklıklarını tahmin etmede veya bir hastanın tıbbi test sonuçlarına göre hastalığının saptanmasında kullanılabilirler. (Tan vd., 2006)

Şekil 2.10’ da işlevlerine göre veri madenciliği yöntemleri sınıflandırılmıştır.



Şekil 2.10 İşlevlerine göre veri madenciliği yöntemleri

3. GÖRSEL VERİ MADENCİLİĞİ (VISUAL DATA MINING)

Bu bölümde çok boyutlu veri setlerinin görselleştirilmesindeki zorluklara değinilerek bu gibi zorlukları ortadan kaldırabilmek için geliştirilen görselleştirme teknikleri avantaj ve dezavantajlarıyla tanıtılmıştır. Veri madenciliği döngüsünü daha etkili hale getiren görsel veri madenciliği kavramından bahsedilmiştir. Ayrıca süsen veri seti kullanılarak görselleştirme teknikleri XmdvTool, Orange, MATLAB R2007a programları kullanılarak örneklendirilmiştir.

3.1 Görselleştirme (Visualization)

Birçok veri madenciliği uygulamasında verilerin birbiri ile olan ilişkilerinin iyi anlaşılması büyük önem taşır. Bunu gerçekleştirmenin yolu, veri keşfi esnasında, insan algı sistemiyle bilgisayar sistemleri arasında esnek, yaratıcı, köprüler kurmaktan geçer. Bu köprüyü kurmanın en iyi yolu da verinin görselleştirilmesidir. Veri görselleştirme teknikleri, bilgisayar grafikleri, görüntü işleme, bilgisayar görüşü (computer vision), kullanıcı arayüzü tasarımı gibi birçok bilim dalının birleşiminden oluşur. Bu teknikler sayesinde bankalar, sayısal kütüphaneler, internet siteleri ve metin veritabanları gibi büyük veritabanlarının görselleştirilmesi mümkün olmaktadır (Çamurcu ve Bilgin, 2007).

Veri görselleştirme, insanın algılama yeteneklerini ve insanlar arası yorumlama farklılıklarını dikkate alarak analiz gerçekleştirilmesine olanak sağlar. Veri görselleştirme teknikleri sayesinde veri hakkında genel bir kanıya varılabilir ve analiz esnasında önemli olabilecek gizli kalmış küçük örüntülerin keşfedilmesi mümkün olabilir. Örnek vermek gerekirse, veri görselleştirilmesi sayesinde değişkenlerin dağılımları, değişken grupları arasındaki kümelenmeler, korelasyonlar gibi ilişkiler gözler önüne serilebilir.

Veri keşfi esnasında, analizciler veri kayıtları arasında yapılar, örüntüler, ilişkiler ararlar. Verilerin grafiksel bir formda temsil edilmesi analizcinin veri yapılarını anlamasını kolaylaştır. Ancak çoğunlukla analistçiler çok boyutlu verilerle uğraşırlar. İnsanların algılama sistemleri de yalnızca 3 boyutla sınırlı olduğu için daha fazla boyut içeren veriler insan algı sisteminin dışına çıkmaktadır. Bundan dolayı, veri görselleştirme teknikleri çok boyutlu veriyi 2 veya 3 boyuta indirgeyerek görselleştirmeli, diğer taraftan da veriler arasındaki ilişkiyi muhafaza edebilmelidir. Bu indirgeme sırasında bir miktar bilgi kayıp olması kaçınılmazdır. Görselleştirmede temel hedeflerden biri bu kaybı minimum düzeyde tutmaktır.

Veri görselleştirme teknikleri kuralların, kavramların daha iyi anlaşılması, yeni yapıların keşfedilmesi veya bu yapıları düzenlemek gibi çeşitli amaçlar için kullanılabilmektedir.

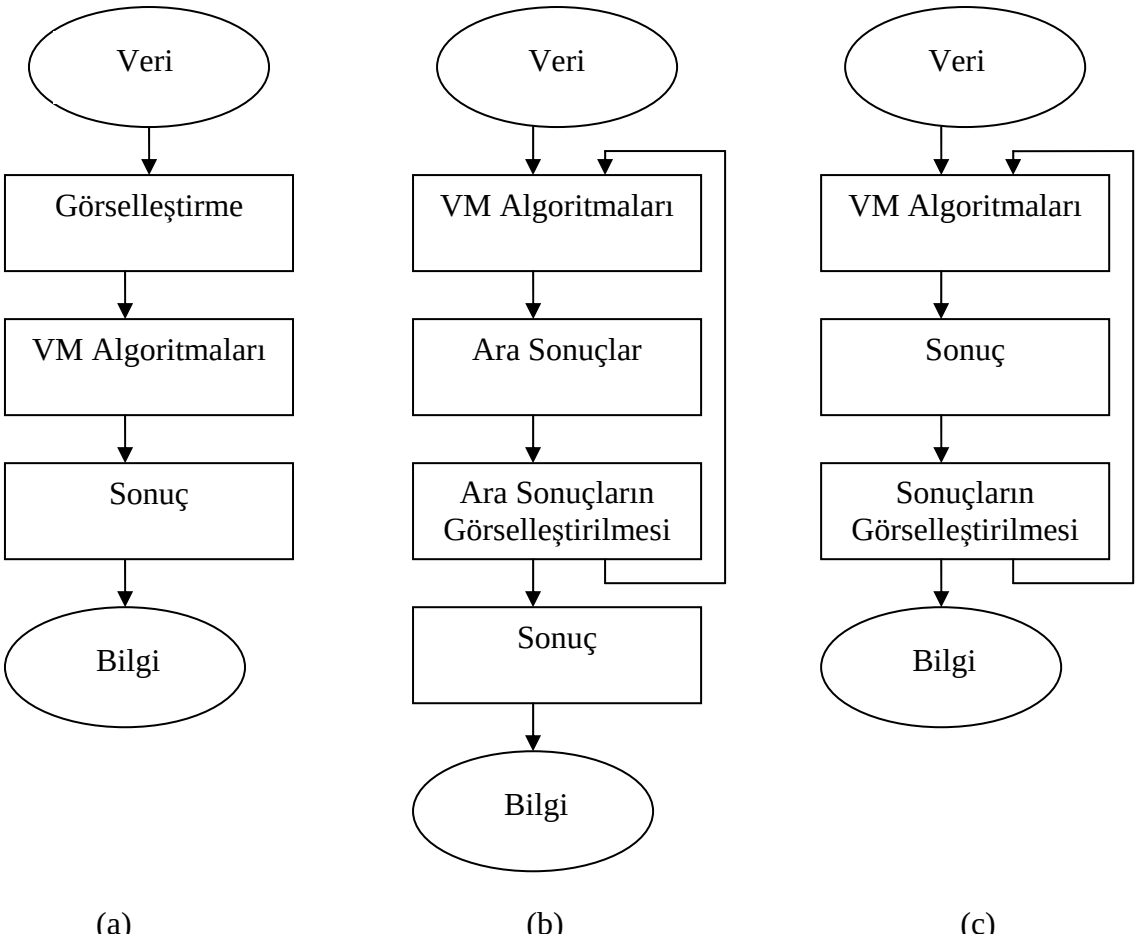
Bunlardan kuralların ve kavramların daha iyi anlaşılması için kullanılan görselleştirmeye "bilgi görselleştirilmesi" (knowledge visualization), grafiklerin ve resimlerin yeni yapılar keşfetmek veya bu yapıları düzenlemek için kullanılan görselleştirmeye "görsel bilgi keşfi" (visual data exploration) denir. Bilgi görselleştirilmesi daha çok var olan bilginin görselleştirilmesi için kullanılırken, görsel bilgi keşfi insanın görsel algılama sistemini mantıksal problemlerin çözümü için kullanmaktadır (Çamurcu ve Bilgin, 2007).

3.2 Görsel Veri Madenciliği Nedir?

Görsel veri madenciliğinin amacı görselleştirme ile veri madenciliğini sentezleyerek veri madenciliği döngüsünü daha efektif hale getirmektir. Görsel veri madenciliği, veri tabanı bilgi keşfi sürecinin bir aşaması olarak bilgisayarla kullanıcı arasında iletişim aracı olarak görselliği kullanan bir adımdır. Görsel veri madenciliği sayesinde veriden yeni, yorumlanabilir örüntüler elde edilir (Ankerst, 2000).

Görsel veri madenciliği, veri madenciliğini oluşturan yaşam döngüsünün veriyi hazırlama, modelin çıkarımı ve onaylama safhalarının üçünüde de görsel gösterimle keşfetmeye çalışır. Veri madenciliği aşamasında, kullanılan görselleştirme yaklaşımlarına göre, görsel veri madenciliği verilerin görselleştirilmesi, ara sonuçlarının görselleştirilmesi ve veri madenciliği sonuçlarının görselleştirmesi olarak üç sınıfa ayrılabilir (Ankerst, 2000).

Şekil 3.1' de görsel veri madenciliğinde kullanılan yaklaşımlara göre, veri madenciliği süreçleri gösterilmektedir. Şekil 3.1 (a)' da verilerin görselleştirilmesi, Şekil 3.1 (b)' de ara sonuçlarının görselleştirilmesi ve Şekil 3.1 (c)' de son sonuçlarının görselleştirilmesi gösterilmektedir.



Şekil 3.1 Görsel veri madenciliğinde kullanılan farklı yaklaşımlar

3.2.1 Görsel Bilgi Keşfi (Visual Data Exploration)

Görsel bilgi keşfi diğer adıyla da veri görselleştirilmesi, herhangi özel veri madenciliği algoritmalarının kullanılmasından önce verinin hızlı ve kolay bir şekilde keşfedilmesi amacıyla kullanılır. Veri görselleştirmesi sayesinde etkili bir biçimde verinin portresi çıkartılabilir ve veri hakkında genel bir kanıya varılabilir.

Görsel bilgi keşfinin özellikleri aşağıdaki gibi sıralanabilir (Keim, 2002) :

- Homojen olmayan, gürültülü veri setlerine kolayca uygulanabilir.
- Görsel veri keşfi sezgiseldir. Karmaşık matematiksel, istatistiksel algoritmalara ihtiyaç duymaz.
- Görsellik veri seti üzerine kalitatif bir bakış açısı sağlar. Görsellik neticesinde elde edilen bilgiler ışığında kantitatif incelemeler için ip uçları elde edilebilir.

3.2.2 Ara Sonuçların Görselleştirmesi (Visualization Of An Intermediate Result)

Veri analizi aşamasında bir algoritma sadece sonuç örüntüleri elde etmez. Algoritma ara sonuçlar da üretebilir. Ara sonuçlar uygun görselleştirme teknikleriyle de görselleştirilebilirler. Bu sayede veri madencisi ara sonuçların görselleştirilmesiyle ilginç ara örüntüler elde edebilir. Bu yaklaşımın ana fikri, uygulamadan bağımsız algoritmik parçalar elde etmektir.

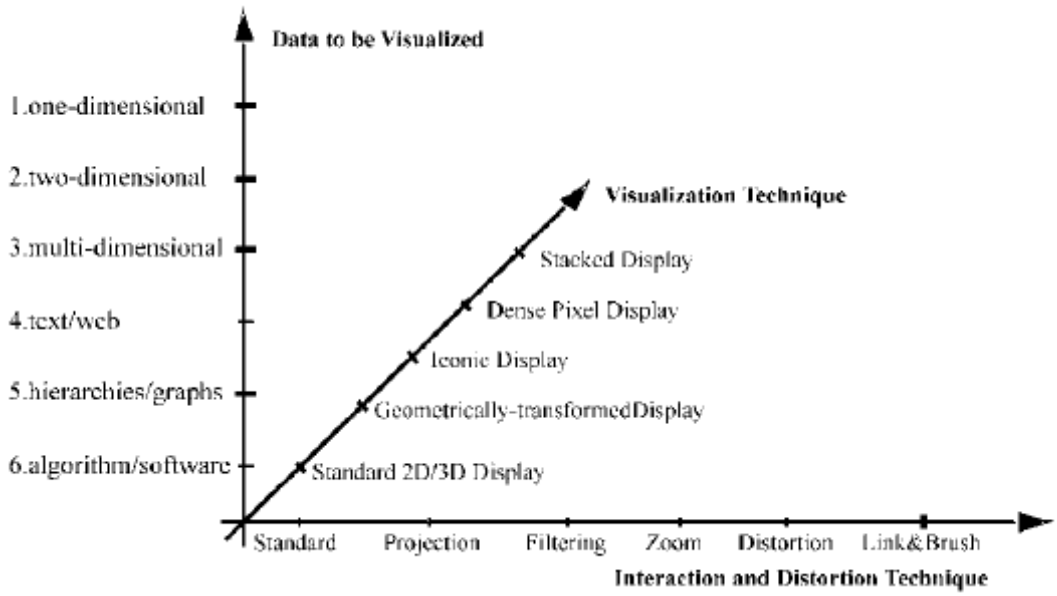
Zaman zaman veri madenciliği çalışmalarında bir veri madenciliği algoritması çok kullanışlı olabilir. Ancak, her veri madenciliği çalışması için uygun bir veri madenciliği algoritması olmayabilir. Bu durumda kullanılan algoritmasının ara sonuçları görselleştirilerek çeşitli çok amaçlı sonuçlar üretilebilir. Daha sonra bu çok amaçlı sonuçlara uygun bir veri madencililiği algoritması seçilerek veri madenciliği çalışması başarıyla tamamlanabilir. Yani veri madenciliği algoritması bu durumda insan ile sürekli etkileşim halinde olur.

3.2.3 Veri Madenciliği Sonuçlarının Görselleştirilmesi (Visualization Of The Data Mining Result)

Veri madenciliği çalışmasında herhangi bir algoritmanın elde ettiği örüntüler görselleştirme sayesinde yorumlanabilir hale getirilir. Görsellik sayesinde, kullanılan algoritmaların giriş parametreleri değiştirilerek, algoritma sonuçları gözlemlenebilir. Veri madenciliği sonuçlarının görselleştirilmesi bir bakıma bulduğumuz değerlerin doğruluğunun sağlanmasında insan algı sisteminin etkin bir şekilde kullanılmasını da sağlar.

3.3 Görsel Veri Madenciliği Yöntemlerinin Sınıflandırılması

Görsel veri madenciliği teknikleri literatürde oldukça farklı adlarla sınıflandırılmaktadır. En yaygın olarak görsel veri madenciliği görselleştirilecek veri tipine (Data Type to Be Visualized), görselleştirme tekniklerine (Visualization Techniques), etkileşim ve bozulma tekniklerine (Interaction and Distortion Techniques) göre sınıflandırılmaktadır (Keim 2002).



Şekil 3.2 Görsel veri madenciliği tekniklerin sınıflandırılması (Keim, 2002)

3.3.1 Görselleştirilecek Veri Tipine Göre (Data Type to Be Visualized)

Veri tiplerine göre çeşitli görselleştirme teknikleri bulunmaktadır. Bu alt bölümde veri tiplerine göre görselleştirme teknikleri tanıtılmaktadır.

3.3.1.1 Tek Boyutlu Veriler (One-Dimensional Data)

Tek boyutlu veri bir tek değişkene sahiptir. Tipik bir örnek olarak tek boyutlu verilere zamansal verileri örnek verebiliriz (Oğur, 2004; Keim 2002).

3.3.1.2 İki Boyutlu Veriler (Two-Dimensional Data)

İki boyutlu veri iki farklı boyuta sahiptir. İki farklı boyuta sahip (enlem ve boylam) coğrafi verileri, iki boyutlu veriye tipik bir örnektir. X-Y çizenekleri (plot) iki boyutlu verilerin gösterimi için tipik bir yöntemdir. Ayrıca haritalar, iki boyutlu coğrafi veriyi gösteren özel bir X-Y çizeneği tipidir (Oğur, 2004; Keim 2002).

3.3.1.3 Çok Boyutlu Veriler (Multidimensional Data)

Çoğu veri seti üçten fazla özellik içerir, dolayısıyla 2 veya 3 boyutlu grafikler gibi basit görselleştirme teknikleri ile gözlenemez. Bu tip verilerin görselleştirilebilmesi için oldukça karmaşık görselleştirme teknikleri kullanılmaktadır (Oğur, 2004; Keim 2002).

3.3.1.4 Metin ve Yardımlı Metin (Text And Hypertext)

Bütün veri tiplerini boyutlandırma terimleriyle tanımlamak mümkün değildir. Çoklu ortam (multimedia) internet sayfaları metin ve yardımlı metinlerden oluşmaktadır. Bu tür veriler kolayca sayısallaştırılamadığından standart görselleştirme teknikleri bu verilere uygulanamaz. Bu durumda görselleştirme teknikleri kullanılmadan önce veri setimizin kullanılabilir bir hale dönüştürülmesi gerekir. Bu tür verilerde kullanılan yöntem, basit bir dönüşüm olan kelime sayımı ile temel bileşenler analizi (PCA, Principle Component Analyses) ya da çok boyutlu ölçekleme (MDS, Multi-Dimensional Scaling)' in birleştirilerek kullanımından oluşmaktadır (Oğur, 2004; Keim 2002).

3.3.1.5 Hiyerarşikler ve Grafikler (Hierarchies and Graphs)

Veri kayıtları sıklıkla bilginin diğer parçaları ile ilişkilidirler. Grafikler genelde bu tip ilişkileri göstermekte kullanılırlar. Bir grafik, düğüm (node) adı verilen nesne seti ile bu nesneleri birbirine bağlayan kenar (edge) adı verilen bağlantılardan oluşur. İnsanlar arasındaki e-mail ilişkileri, insanların alışveriş davranışları, bilgisayarlardaki hard disklerin dosya yapıları ilişkili verilere örnek olarak verebilir. Hiyerarşik ve grafiksel verilerin görselleştirilmesinde kullanılan özel teknikler vardır (Oğur, 2004; Keim 2002).

3.3.1.6 Algoritmalar ve Yazılımlar (Algorithms and Software)

Görselleştirmenin amacı algoritmaların anlaşılmasına yardımcı olacak yazılımların geliştirilmesini sağlamaktır. Bu göreve yönelik bir sürü yazılım vardır. Yazılımların sunduğu grafiksel olanaklara göre de grafiksel teknikler sınıflandırılabilir (Oğur, 2004; Keim 2002).

3.3.1.6.1 Veri Görselleştirme Programları

Görselleştirme ile veri madenciliğini sentezleyerek veri madenciliği döngüsünü daha efektif hale getiren ücretli ve ücretsiz birçok bilgisayar programı bulunmaktadır. Bu alt bölümde veri madenciliği algoritmalarını çeşitli görsel tekniklerle etkili hale getiren bilgisayar programları tanıtılmaktadır.

Mathametica, S/S-Plus/R ve MATLAB programları veri görselleştirmelerinde çoğunlukla kullanılan ticari programlardır. Matematica çok sofistike ve kullanımı kolay bir programlama dilidir. Matematica programıyla grafik, sayısal, sembolik ve teknik birçok işlem yapılabilmektedir. S/S-Plus/R programı veri görselleştirme ve istatistik alanlarında araştırmacılara çok geniş olanaklar sağlayan diğer bir bilgisayar programıdır. S-Plus ticari bir

program olmasına rağmen R programı http://www.r-project.org/ internet adresinden ücretsiz olarak indirilebilir. MATLAB ise çoğunlukla mühendisler ve uygulamalı matematikçiler tarafından kullanılan başka bir bilgisayar programıdır. MATLAB, verilerin grafiksel gösterimi konusunda çok çeşitli teknikler kullanabilmeyi mümkün kılar. Grafik biçimlendirme konusundaki etkileşimli araçları sayesinde, eldeki veri hakkında önemli bilgilerin ve değişimlerin etkileyici bir şekilde sunulması sağlanabilir. Ayrıca MATLAB' in sunduğu programlama özellikleri sayesinde görsel, matematiksel ve istatistiksel analizler kullanıcının isteği doğrultusunda geliştirilerek veri madenciliği sürecinin daha etkili bir şekilde gerçekleşmesi sağlanabilir. MATLAB programının çeşitli analizleri ve görsel teknikleri hızlı bir şekilde kullanılmasına olanak sağlayan internette çeşitli ücretsiz araç kutuları da bulunmaktadır.

Veri görselleştirilmesinde kullanılan diğer ücretli popüler istatistik paket programları SAS, SPSS, Minitab ve Stata' dır. Bunlar sınırlı sayıda görselleştirme tekniklerine sahiptirler. Ek olarak XGobi, Orange, XmdvTool ve CrystalVision gibi birçok ücretsiz görselleştirme programları da bulunmaktadır. Bunlardan XGobi en bilinen veri görselleştirme programıdır. SAS Enterprise Guide, SPSS Clementine diğer görselleştirmede kullanılan veri madenciliği programlarıdır.

XGobi, XmdvTool ve CrystalVision verilerin görünümlerindeki manipulasyonlar için etkileşimli ve dinamik yöntemler kullanan modern veri görselleştirme programlarıdır. Bu programlar, çok boyutlu uzaydaki çizgi ve noktaların izdüşümlerinin 2 boyutlu gösterimlerini gerçeklerler. Görsel, grafiksel ara yüzlerinin gelişkinliği ve hem verinin incelenmesinde hem de sonuçların görselliğini arttırmaya yönelik renk değiştirme, farklı grafiklere ve boyutlara taşıyabilme gibi özelliklerde kullanıcı müdahalesine maksimum düzeyde izin vermesi bu programları kullanışlı hale getirmektedir. XGobi hatta S-Plus/R kütüphane dosyası olarak ta çalıştırabilen bir programdır. Sırasıyla, XGobi, XmdvTool ve CrystalVision programları http://www.research.att.com/areas/stat/sgobi/, http://davis.wpi.edu/~xmdv/downloadxmdv.html ve ftp://www.galaxy.gmu.edu/pub/software/CrystalVisionDemo.exe internet adreslerinden ücretsiz olarak indirilebilirler. Orange programı ise kendi başına bir veri madenciliği programıdır. Orange kullanıcıya veri hazırlama, modelleme, keşifsel veri analizi gibi çeşitli kullanım olanakları sağlamaktadır. Orange programı http://www.ailab.si/orange/downloads.asp internet adresinden ücretsiz olarak indirile bilmektedir.

MATLAB programının sağladığı ileri görselleştirme ve algoritma geliştirme özelliğinden dolayı bu tez çalışmasında ağırlıklı olarak MATLAB programının kullanılması uygun bulunmuştur. MATLAB programının içersinde bulunan hazır komutlarla yapılamayan bir takım grafiksel teknikler ve algoritmalar, internette hazır bulunan araç kutuları kullanılarak yapılabilir. Örneğin bu tez çalışmasında, kendinden düzenlenen haritalar (Self Organizing Map) algoritmasını kullanabilmek ve sonuçlarını görselleştirebilmek için http://www.cis.hut.fi/projects/somtoolbox/download/ internet adresinden indirilebilen somtoolbox' ı ve daha başka grafiksel teknikleri kullanabilmek için Wendy and Angel Martinez tarafından geliştirilen, http://lib.stat.cmu.edu/matlab/ internet adrsinden indirilebilen edatoolbox' ı kullanılmıştır. Ayrıca tez kapsamında kümeleme analizi sonuçlarının doğruluğunu göstermek adına kullanılan küme doğruluk (cluster validity) endeksleri http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=14620&objectType=file internet adresinden indirilebilen clusterValidityAnalysisPlatform3.42 toolbox' ı kullanarak yapılmıştır. Çeşitli amaçlar için internette hazır bulunan toolbox' lar la ve MATLAB programıyla karmaşık veri madenciliği algoritmalarını hem kullanabilme hem de geliştirebilme, görselleştirme ile de veri madenciliğini sentezleyerek veri madenciliği döngüsünü daha efektif hale getirebilme fırsatları yakalanabilir.

3.3.2 Etkileşim ve Bozulma Tekniklerine Göre (Interaction and Distortion Techniques)

Veri görselleştirme tekniklerini etkili bir şekilde kullanabilmek için bazı araçlara ihtiyaç duyulmaktadır. Bu bölümde görselleştirme tekniklerinin etkinliğini arttıran araçlar tanıtılmaktadır.

3.3.2.1 Dinamik İzdüşümler (Dynamic Projections)

Bu yöntemin mantığı çok boyutlu veri setinin izdüşümlerinin dinamik olarak değiştirilmesi suretiyle veri seti içerisinde bilgi arayışı yoluna gidilmesidir. Dinamik izdüşümlerin klasik örneği Grand Tour sistemidir. Grand Tour, bir dizi serpilme grafiği serisi olarak çok boyutlu veri setinin bütün ilginç iki boyutlu izdüşümlerini göstermeye çalışır (Oğur, 2004; Keim 2002).

3.3.2.2 Etkileşimli Filtreleme (Interactive Filtering)

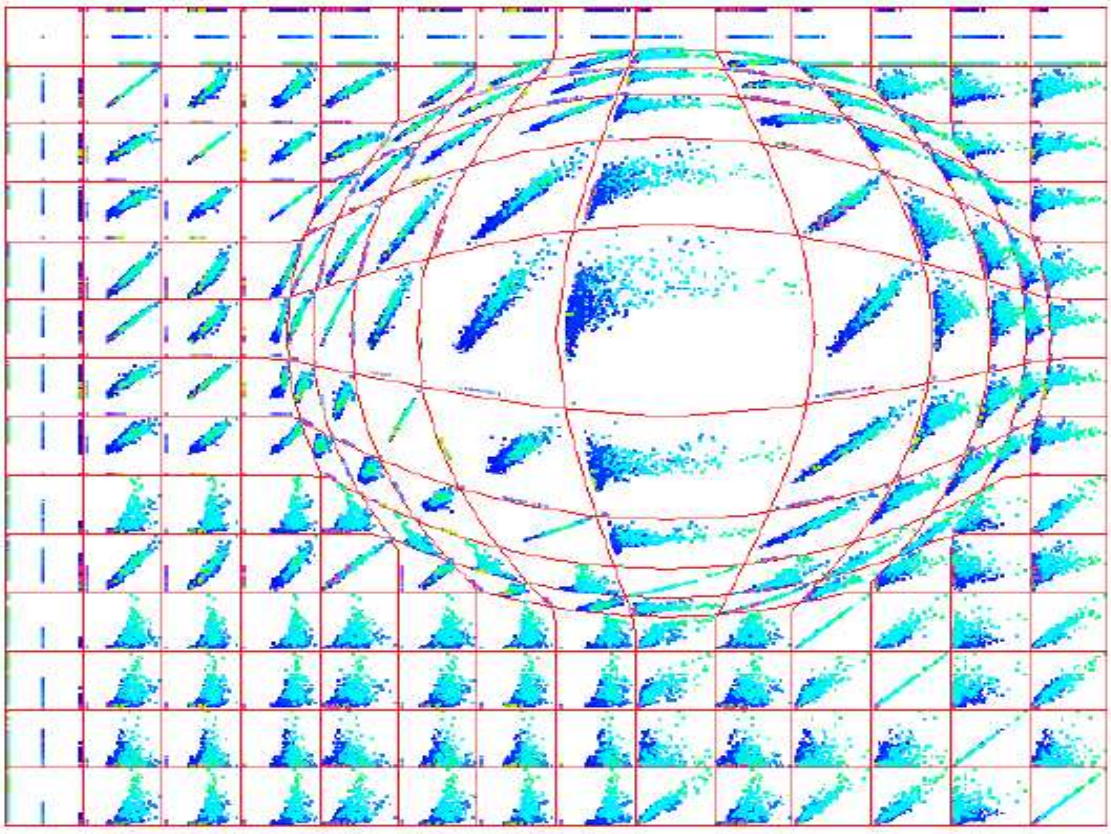
Büyük veri setlerinde bilgi aranırken veri setinin interaktif olarak segmentlere ayrılması ve ilginç alt setlerle ilgilenilmesi zaman kazanmak ve bilginin bulunuşunu kolaylaştırması açısından önemlidir. İnteraktif filtreleme tekniği bunu sağlamaktadır.

3.3.2.3 Etkileşimli Mesafe Ayarlama (Interactive Zooming)

Mesafe ayarlama (zooming) çok geniş bir yelpazede kullanıldığından oldukça iyi bilinen bir tekniktir. Verinin farklı çözünürlüklerde izlenebilmesini, işe yarar parçaların gözlenebilmesini sağlar. Ayrıca çok büyük veri setlerinde tüm veriler oldukça fazla sıkıştırıldıklarında hepsine birden genel bir görünüş hâkimiyeti sağlar (Oğur, 2004).

3.3.2.4 Etkileşimli Bozulma (Interactive Distortion)

İnteraktif bozulma teknikleri, veri içerisinde bilgi arama süreci esnasında drill-down diye isimlendirilen veri yığını hiyerarşisinin alt dosyalarının taranması işlemi gerçekleştirilirken verilerin genel yapısının korunmasını sağlayan bir tekniktir. Bu teknikle verilerin bir kısmı tüm ayrıntılarıyla gözlenebilirken geri kalan kısmı daha az detayla incelenebilmektedir (Oğur, 2004).



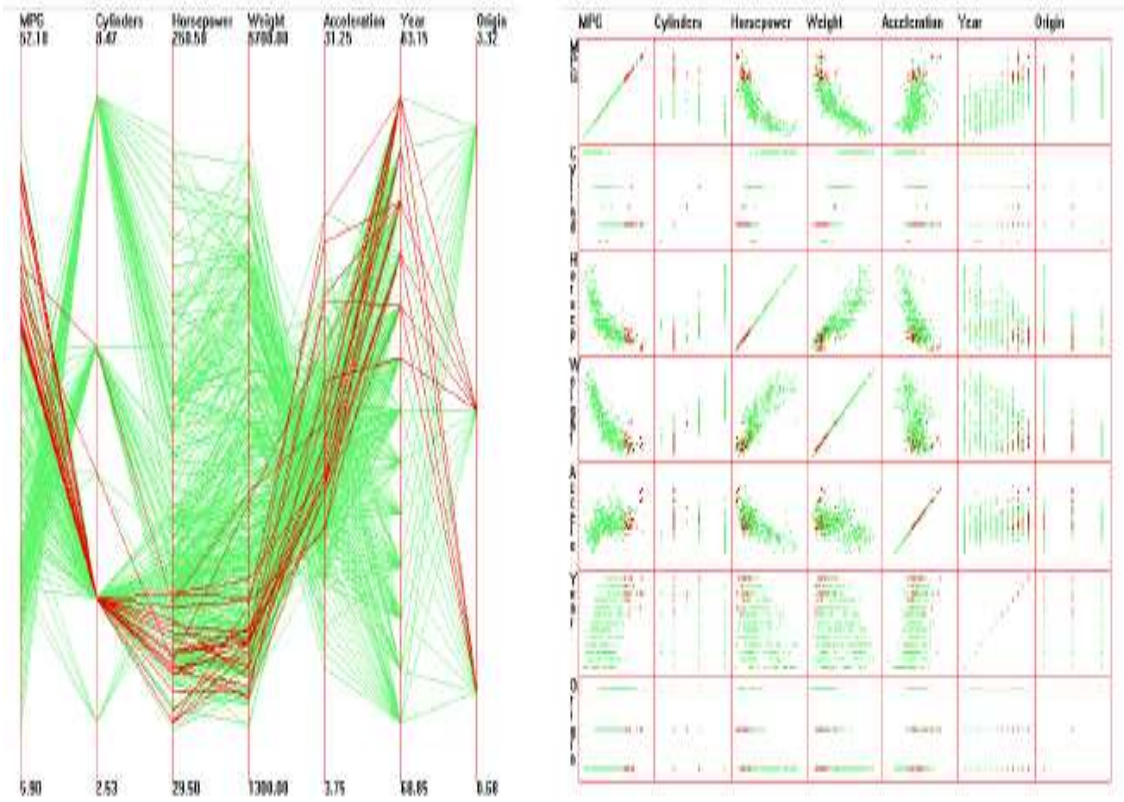
Şekil 3.3 Serpilme matris grafiği için etkileşimli bozulma grafiği (Keim, 2002)

Şekil 3.3’ de XmvdTool programıyla çizilen serpilme grafiği bulunmaktadır. Etkileşimli bozulma yöntemiyle serpilme grafiğindeki verilerin bir kısmı tüm ayrıntılarıyla gözlenebilirken geri kalan kısmı daha az detayla incelenebilmektedir.

3.3.2.5 Etkileşimli Birleştirme ve Temizleme (Interactive Linking and Brushing)

Çok boyutlu verilerin görselleştirilmesi için kullanılabilecek bir sürü yöntem vardır; fakat her yöntemin diğer yöntemlere göre daha güçlü ve daha zayıf olduğu yönleri vardır. Etkileşimli birleştirme ve temizlemenin kullanılma amacı tek başına kullanılan tekniklerin eksikliklerini gidermek için farklı görselleştirme yöntemlerinin birlikte kullanılmasından ileri gelmektedir (Keim, 2002).

Bir grafiğin alt kümelerinin değişik sembol ve renklerle gösterilme işlemine temizleme (bruhshing), bir grafikte işaretlenen gözlemlerin diğer grafikte de otomatik olarak işaretlenmesine birleştirme (linking) denilmektedir. Etkileşimli birleştirme ve temizlemede ise bir grafikte farklı renk ve sembolde işaretlenen veri alt kümeleri diğer grafikte de eş zamanlı olarak faklı renk ve sembolde gösterilmektedir (Keim, 2002).



Şekil 3.4 Paralel koordinat ve serpilme matris grafikleri için etkileşimli birleştirme ve temizleme grafiği (Keim, 2002)

Şekil 3.4’ de XmvdTool programıyla çizilen paralel koordinat ve serpilme grafiği bulunmaktadır. Etkileşimli birleştirme ve temizleme yöntemiyle bir grafikte belirli veri alt kümeleri kırmızı renkle işaretlenerek aynı veri alt kümeleri diğer grafikte de kırmızı renkle işaretlenmiş olur. İşaretlenen gözlemler değiştirildiğinde aynı değişim diğer grafikte de

gözlenmektedir. Bu sayede bir grafiğin zayıf yönleri diğer grafiğin kuvvetli yanıyla birleştirilerek etkin bir görsellik elde edilmiş olur.

Etkileşimli birleştirme ve temizleme tekniklerinin en bilinen örnekleri serpilme matrisleri, çubuk grafikleri, paralel koordinatlar ve piksel gösterimlerinde yapılmaktadır. XmvdTool, XGobi programlarıyla etkileşimli birleştirme ve temizleme işlemleri kolaylıkla yapılabilmektedir.

3.3.3 Görselleştirme Tekniklerine Göre (Visualization Techniques)

Verileri görselleştirmek için çeşitli görselleştirme teknikleri bulunmaktadır. x-y (x-y-z), kutu, çizgi ve benzeri grafikler gibi standart 2, 3 boyutlu ve hatta yüksek boyutlu veriler için geliştirilen özel görsel teknikler de bulunmaktadır.

3.3.3.1 Standart 2 ve 3 Boyutlu Gösterimler

Bu tür tekniklerin en bilineni iki ve üç değişkenli veri setini x, y ve z eksenleri boyunca kartezyen koordinat sistemine işaretleyen serpilme grafikleridir (scatterplots). Ayrıca çizgi, kutu, histogram, pasta, kontör grafikleri de standard 2 ve 3 boyutlu gösterim grafikleri arasında bulunmaktadır.

3.3.3.1.1 2 ve 3 Boyutlu Serpilme Grafikleri (2-D and 3-D Scatterplots)

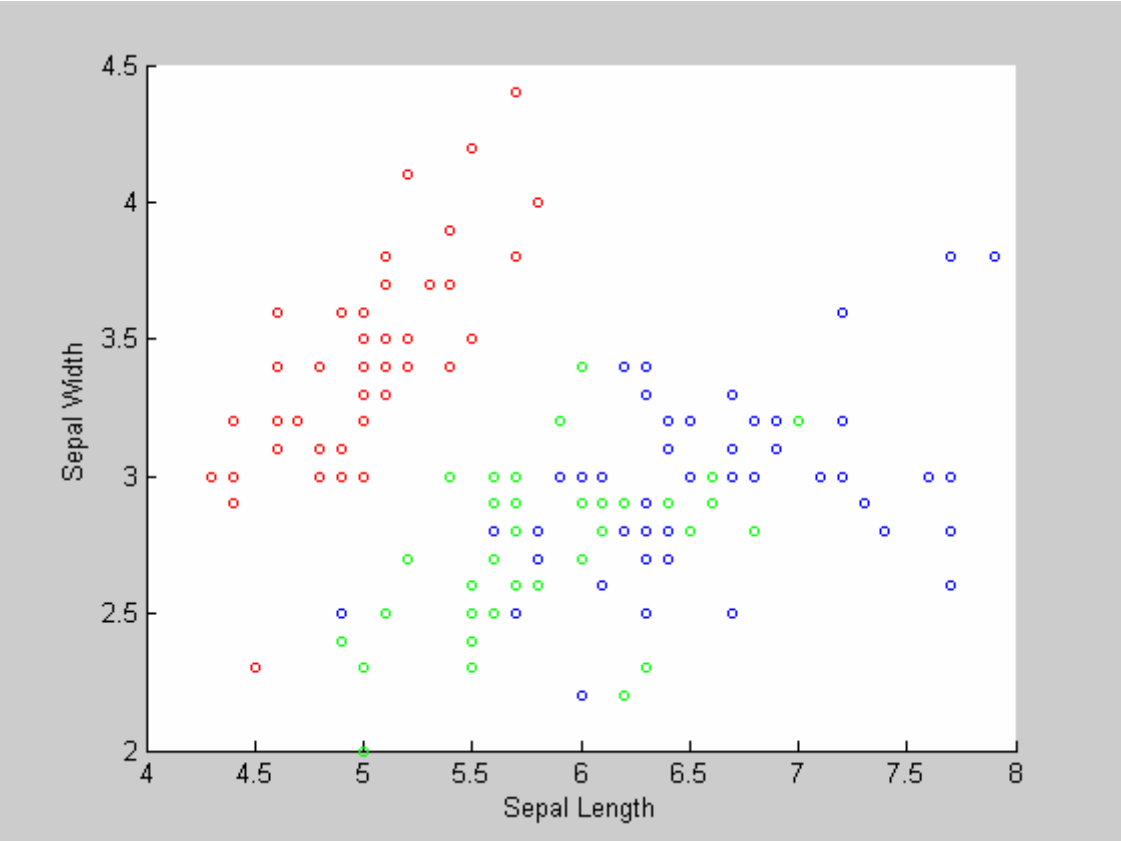
Serpilme grafiği iki veya üç değişken arasındaki ilişkinin yönünü, tipini ve büyüklüğünü belirlemeye yardımcı olan görsel bir tekniktir. 2 boyutlu serpilme grafiğinde, n birimden elde edilen (x_i, y_i) değerler çifti xy koordinat düzleminde, x_i değerleri x ekseninde, y_i değerleri y ekseninde gösterilecek şekilde serpilme grafiği çizilir. 3 boyutlu serpilme grafiğinde ise n birimden elde edilen (x_i, y_i, z_i) değerler üçlüsü xyz koordinat düzleminde, x_i değerleri x ekseninde, y_i değerleri y ekseninde ve z_i değerleri z ekseninde gösterilecek şekilde serpilme grafiği çizilir.

Örnek 3.1 :

Süsen veri seti için 2 ve 3 boyutlu serpilme grafikleri çizelim.

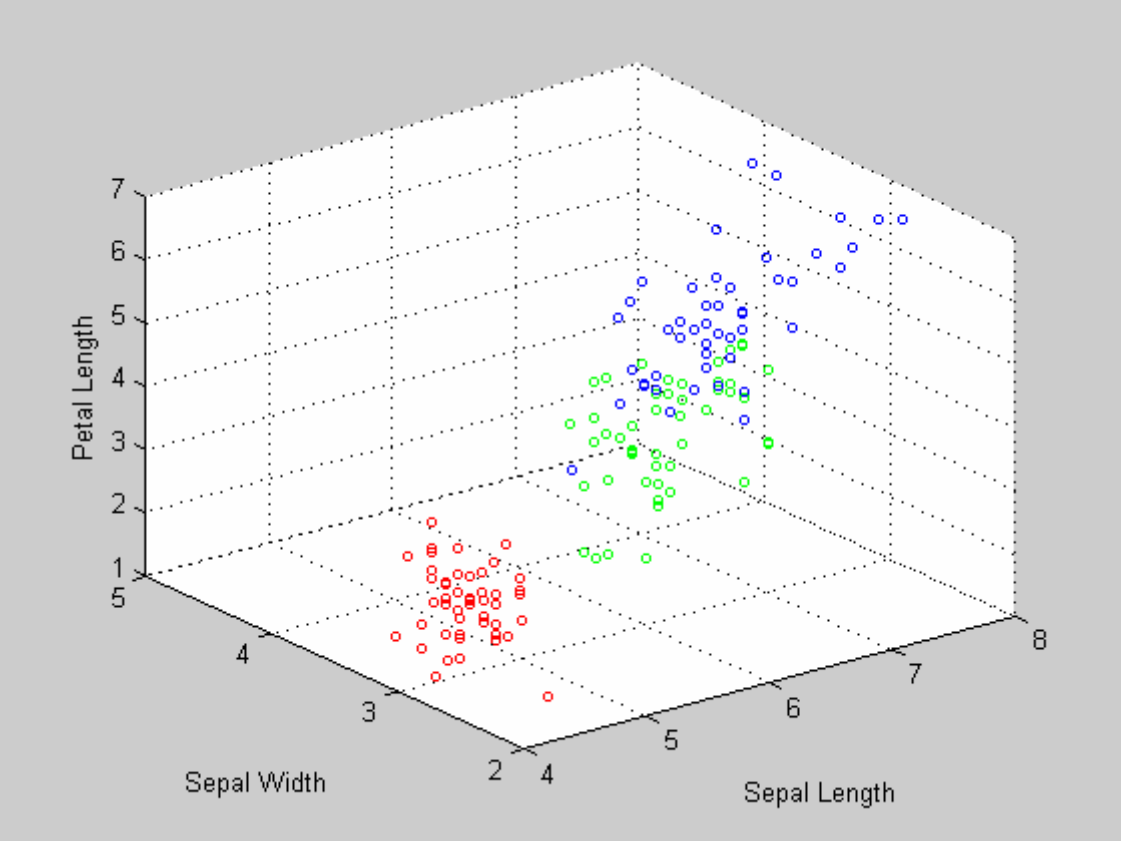
```
load fisheriris
for i=1:150
    if i<=50
        a(i,:)=1;
    elseif i>50 & i<=100
            a(i,:)=2;
        else
```

```
            a(i,:)=3;
    end
end
ind1=find(a==1); % Kırmızı
ind2=find(a==2); % Yeşil
ind3=find(a==3); % Mavi
c=zeros(length(meas),3);
c(ind1,1)=1;
c(ind2,2)=1;
c(ind3,3)=1;
scatter(meas(:,1),meas(:,2),10,c)
xlabel('Sepal Length')
ylabel('Sepal Width')
```



Şekil 3.5 Süsen veri seti için serpilme grafiği

```
scatter3(meas(:,1),meas(:,2),meas(:,3),10,c)
xlabel('Sepal Length')
ylabel('Sepal Width')
zlabel('Petal Length')
```



Şekil 3.6 Süsen veri seti için 3 boyutlu serpilme grafiği

Şekil 3.5 ve Şekil 3.6' da süsen veri seti için 2 ve 3 boyutlu serpilme grafikleri bulunmaktadır. Sırasıyla, setosa, versicolor ve virginica bitki cinsleri, kırmızı, yeşil ve mavi renklerle gösterilmektedir.

3.3.3.1.2 Kutu Grafikleri (Box Plots)

Analatik çalışmalara başlamadan önce verileri tanımak bakımından çizilmesi uygun bir grafiktir.

Kutu grafiği çeşitli grafik elemanlarına sahiptir. Bunlar:

- Grafiğin altındaki ve üstündeki uzun çizgiler büyüklüklerine göre sıralanmış verilerin % 25' inci ve %75' inci yüzdelik değerlerine karşılık gelir. Grafikteki bu iki çizgi arasındaki mesafeye değişim aralığı denir.
- Grafiğin ortasındaki çizgi, verinin medyanına karşılık gelir. Eğer medyan grafiğin ortasında değil ise, bu grafik elimizdeki verinin çarpık olduğunu gösterir.
- Grafiğin en altında ki ve en üstünde ki kısa çizgiler, verilerde aşırı değer (outlier) olmadığını varsayarsak, verilerin geri kalan kısmını gösterir. Ayrıca, bu kısa çizgiler en küçük ve en büyük veri değerlerini gösterir. Bir aşırı değer (outlier) değişim

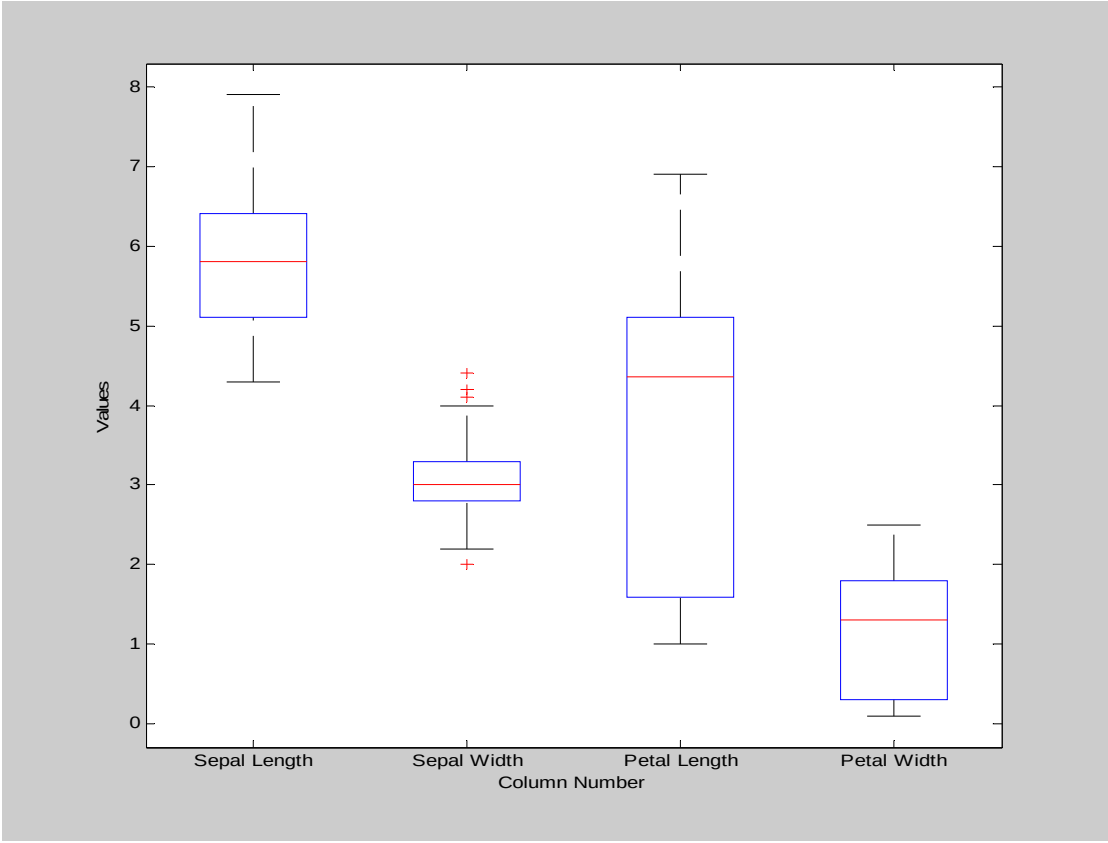
aralığının en az 1.5 katı kadar %25’ inci ve % 75’ inci yüzdelik değerlerden uzaktadır.

- Grafiğin en tepesinde ve en aşağısındaki artı işaretler, verideki aşırı değerleri (outlier) gösterir. Bu artı işaretleri veri girişinde veya verilerin ölçülmesinde bir hata yapıldığını gösterebilir ya da tesadüfî olarak elimizdeki veri setinde aşırı değerler gerçekten vardır.

Örnek 3.2 :

Süsen veri seti için kutu grafikleri çizelim.

```
load fisheriris
boxplot(meas)
```



Şekil 3.7 Süsen veri seti için kutu grafiği

Şekil 3.7’ da süsen veri seti için çizilen kutu grafiği bulunmaktadır. Şekilden de gözüktüğü gibi Sepal width değişkeninin de aşırı değerler bulunmaktadır. Petal length değişkenin de ise verilerin belirgin bir şekil de çarpık dağıldığı gözlenmiştir.

3.3.3.1.3 Çizgi ve Çoklu Çizgi Grafikleri (Line and Multiple Line Graphs)

Çizgi grafikleri, bir değişkenin tek değerlerli veya parça parça sürekli fonksiyonlarını göstermekte kullanılır. Çizgi grafikleri genellikle (x,y) gibi iki boyutlu verileri göstermekte

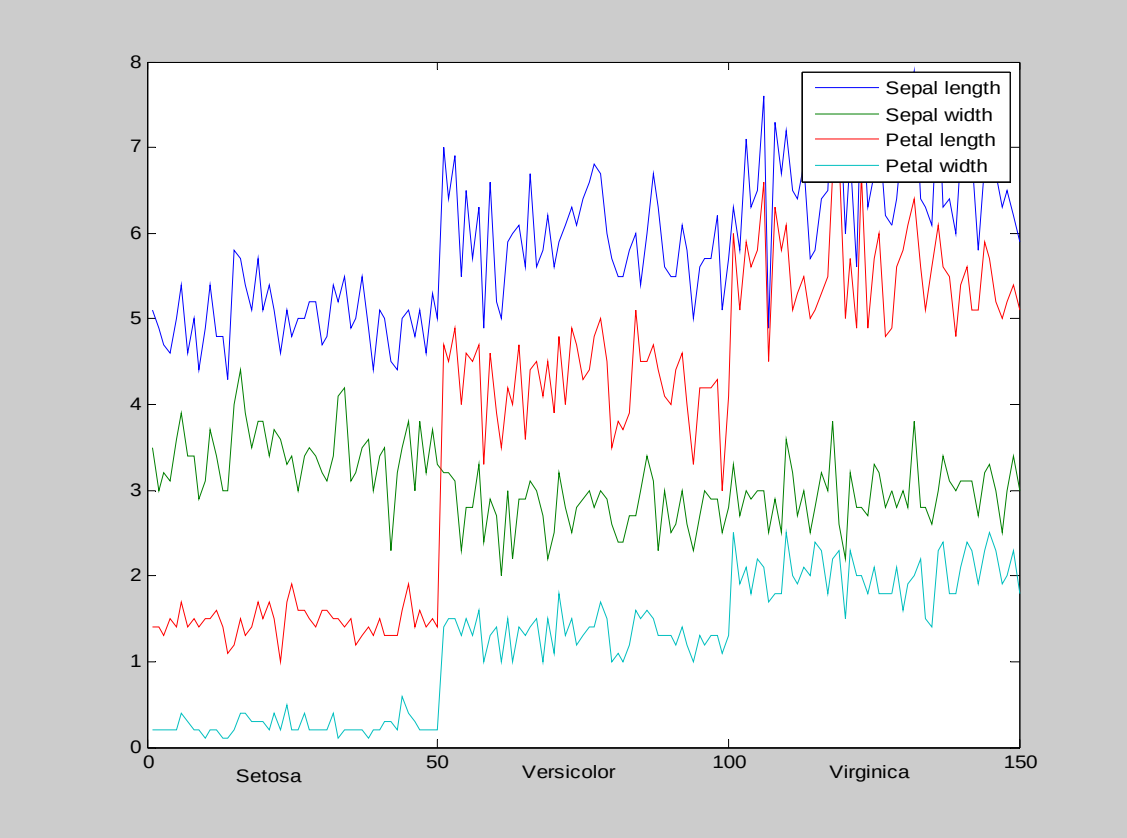
tercih edilir. Burada x bir değişkeni, y de x değişkenine karşılık gelen değerleri gösterir. Çizgi grafiklerinde grafiğin zemin rengi grafiklerde bulunan değerleri belirginleştirmek için özel renklerde seçilebilir (Hoffman, 1999).

Çoklu çizgi grafikleri ($x,y_1,y_2,y_3\ldots$) gibi iki değişkenden daha çok değişkenli verileri görselleştirmek için kullanılır. Çoklu çizgi grafiklerinde farklı sürekli çizgilerin her biri birer değişkene karşılık gelmektedir. Ancak yüksek boyutlu verilerde kullanılan çoklu çizgi grafiklerde boyut ayrımlarının kolay olmamasından dolayı bazı sorunlar bulunmaktadır. Boyut ayrımını kolaylaştırmak için boyut çizgileri farklı renk ve biçimde çizilebilirler. Çoklu çizgi grafiklerinde her boyut farklı ölçeklerde olabilir. Bunun için değişkenleri standardize etmek yerinde olur (Hoffman, 1999).

Örnek 3.3 :

Süsen veri seti için çoklu çizgi grafikleri çizelim.

```
load fisheriris
plot(meas)
```



Şekil 3.8 Süsen veri seti için çoklu çizgi grafiği

Şekil 3.8’ de, süsen bitkisinin sepal length (çanakyaprak uzunluğu), sepal width (çanak yaprak genişliği), petal length (taçyaprak uzunluğu) ve petal width (taçyaprak genişliği) dört değişkeni farklı renklerde olacak şekilde, çizgi grafiği bulunmaktadır.

3.3.3.2 Geometrik Olarak Dönüştürülmüş Gösterimler (Geometrically Transformed Displays)

Geometrik olarak dönüştürülmüş gösterimler, çok boyutlu veri setlerinin içerisinde ilginç dönüşümler ararlar. Bu sınıfta yer alan, serpilme matrisleri, RadViz, PolyViz, survey plot, paralel koordinatlar, Andrews eğrileri, temel bileşenler analizi (PCA, Principle Component Analyses), çok boyutlu ölçekleme (MDS, Multi-Dimensional Scaling), kendinden düzenlenen haritalar (SOM, Self Organizing Maps) en sık kullanılan görselleştirme teknikleridir.

3.3.3.2.1 Serpilme Matrisleri (Scatterplot Matrices)

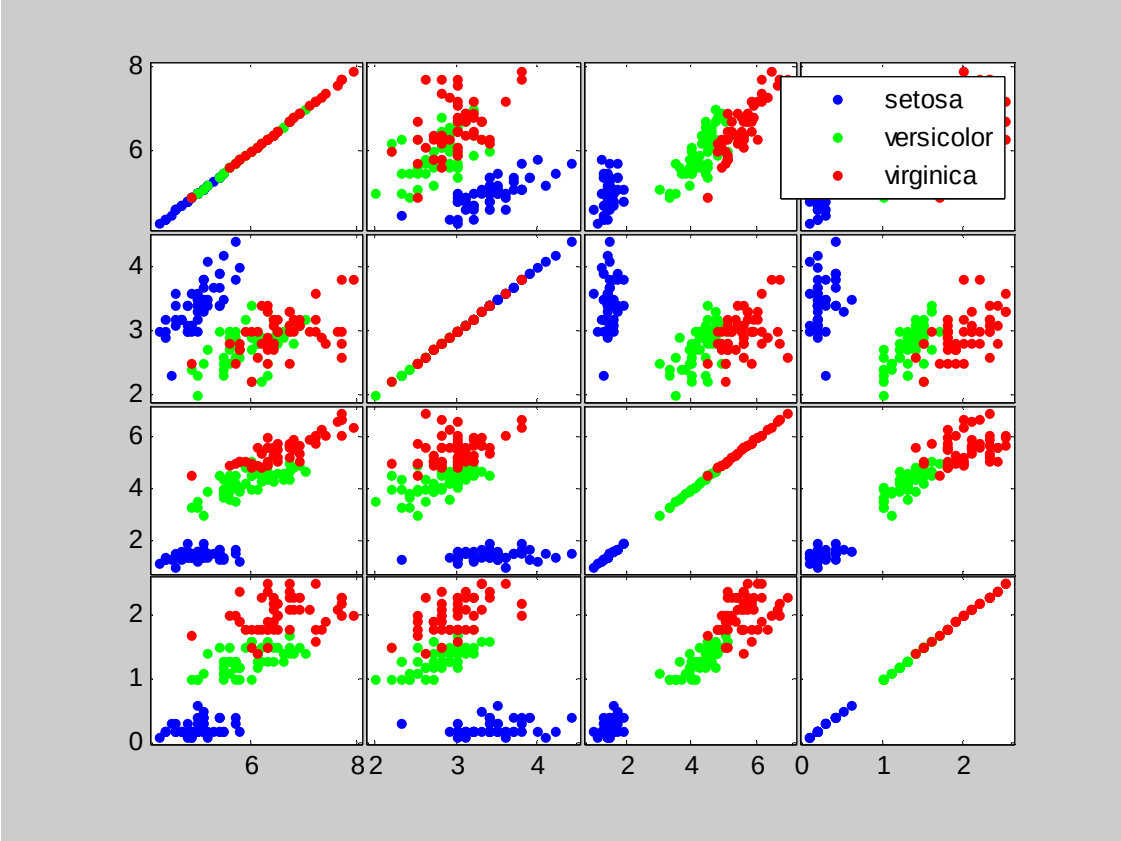
Serpilme matrisleri, iki ve daha fazla değişkenin aynı anda birbirileriyle ikili ilişkilerini (karşılıklı ilişki) gösteren bir grafiktir. Serpilme matrislerinde n tane değişkenin (boyut) $n(n-1)/2$ tane ikili ilişki grafiği bulunur.

Serpilme matrisleri, ikili değişkenler arasındaki korelasyonların tespiti açısından sağladığı kolaylıklar sayesinde en sık kullanılan çok boyutlu görselleştirme tekniğidir. Ancak veri boyutumuzun büyük olması nedeniyle serpilme matrislerinden bilgi elde edilmesi zorlaşmaktadır. Ayrıca serpilme matrisinin sağladığı ikili ilişki grafikleri veri madenciliği gibi çalışmalarında yeterli bilgi verememektedir.

Örnek 3.4 :

Süsen veri seti için matris grafiği çizelim.

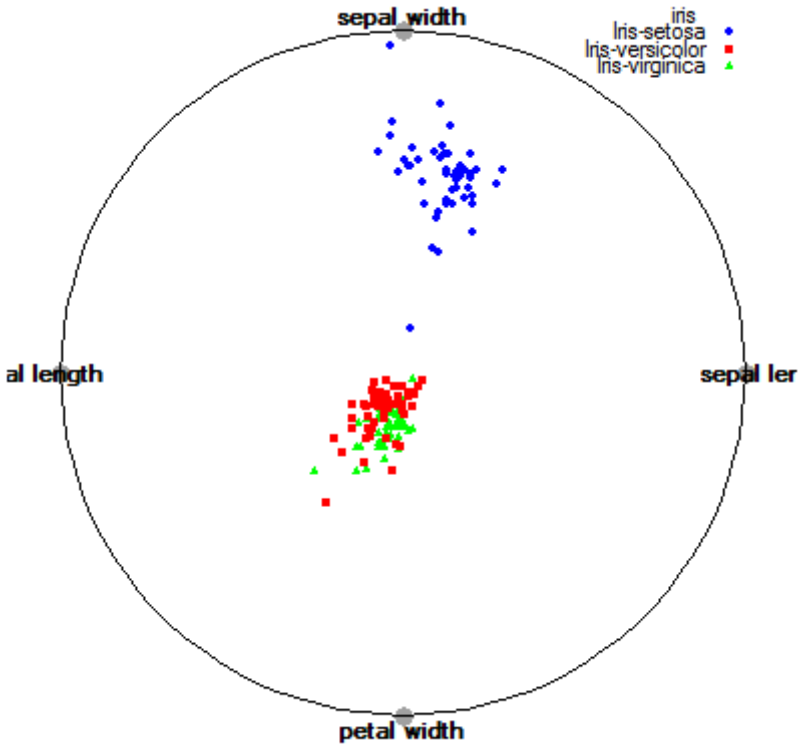
```
load fisheriris
gplotmatrix(meas,meas,species)
```



Şekil 3.9 Süsen veri seti için serpilme matrisleri

3.3.3.2.2 Radyal Koordinat Görüntüleme (Radial Coordinate Visualization, RadViz)

Radyal koordinat görüntüleme (RadViz) tekniğinde veri noktalarının her biri bir dairenin içersinde dağılmış durumdadır. Burada ki amaç, serpilme grafiklerindeki gibi eksenlerde yer alan değişkenler arasındaki serpilmeyi çizmektir. Ancak serpilme grafiğinden farklı olarak RadViz' de dairenin merkezinden eşit açılı dağılmış bütün değişken değerlerine göre serpilme grafiği çizilir. Veri noktalarının her birinin konumu, dairenin merkezinden eşit açılı, dairenin çevresinde sıralanmış olan değişken değerlerine göre belirlenir. Burada bir veri noktasından daire çevresindeki değişkenlere çizilen vektörlerin toplamlarının sıfır olması gerekmektedir. RadViz yönteminde, grafikteki bütün değişkenlere eşit önem verilebilmesi adına, veri setimizde yer alan değişkenlerin standartlaştırılması gerekmektedir (Leban vd., 2006).



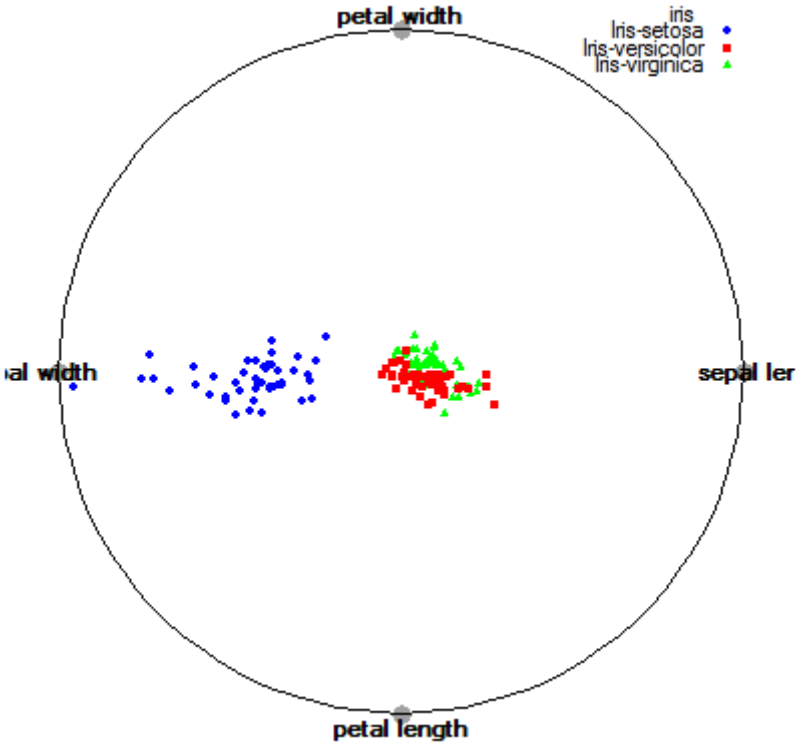
Şekil 3.10 Süsen veri seti için Orange programıyla çizilen RadViz

RadViz grafiğinde potansiyel küme yapıları, sapan değerler gözlenebilmektedir. Şekil 10' da süsen veri seti için çizilen RadViz grafiği bulunmaktadır. Şekil 3.10' da setosa, versicolor ve virginica süsen bitki cinsleri farklı renklerle gösterilmiştir. Şekil 3.10' dan gözüktüğü üzere bu üç cins farklı özelikler sergilemektedirler. Özelikle setosa bitki cinsi diğer bitki cinslerine göre oldukça farklı bir yapı göstermektedir. Versicolor ve virginica bitki cinsleri bir birilerine göre benzer özelikler sergilemektedirler.

RadViz yönteminin bazı özellikleri (Leban vd., 2006) :

1. Standartlaştırma işleminden sonra her değişken için hemen hemen eşit değerlere sahip olan veriler, dairenin merkezine yakın bir şekilde dağılırlar.
2. Hemen hemen her değişken için eşit değerlere sahip olan veriler dairenin merkezi etrafında dağılırlar.
3. Verinin herhangi bir değişken değeri diğer değişken değerlerine göre oldukça büyük bir değere sahipse; veri bu değişkene yakın olacak şekilde dairenin içersinde konumlandırılır.
4. Verinin bir değişken değeri diğer değişken değerlerine göre oldukça küçük bir değere sahipse; veri bu değişkene uzak olacak şekilde dairenin içersinde konumlandırılır.

RadViz grafik yönteminde daire içersinde dağılan verilerin şekilleri, daire etrafında dizilen değişkenlerin sıralarına göre farklılık göstermektedir. Mesela m değişken için dairenin etrafında $(m-1)!/2$ tane farklı şekilde değişken sıralanabilir. Her farklı sıramaya göre RadViz grafiğinde verilerimizin dağılımları farklılık göstermektedir (Leban vd. 2006).



Şekil 3.11 Süsen veri seti için Orange programıyla çizilen RadViz

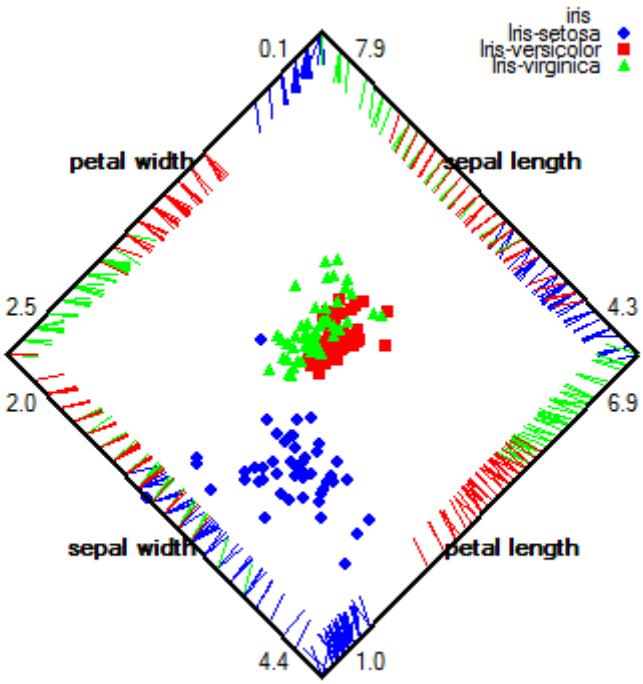
Şekil 3.11' de, süsen veri seti için, değişkenleri farklı sırlanmış RadViz grafiği bulunmaktadır. Şekil 3.10' a göre değişkenleri farklı sıralanmış, Şekil 3.11' de ki RadViz grafiğinde, veri noktalarının dağılımları farklılık göstermektedir.

3.3.3.2.3 Geliştirilmiş RadViz (PolyViz)

Radviz grafiğinin en büyük dezavantajlarından bir tanesi farklı boyut değerlerine sahip verilerin grafik üzerinde bir birine çakışan noktalarla temsil edilebilmesidir. Bu sorunu büyük oranda ortadan kaldırabilmek için RadViz grafiğinden PolyViz grafiği geliştirilmiştir (Hoffman, 1999).

PolyViz grafiğinde, RadViz grafiğinden farklı olarak, farklı değerlere sahip veri noktalarının çakışmasını önleyebilmek için çember üzerinde konumlandırılan değişkenler yerine (tıpkı

serpilme grafiklerinde olduğu gibi) bir eksen boyunca konumlandırılmış değişkenler kullanılmaktadır. Veri değerlerimiz, değişken eksenleri boyunca kısa çizgilerle temsil edilmiş ve değişkenlerde kesişen değerler grafikte nokta olarak konumlandırılmıştır. PolyViz grafiğinde, RadViz grafiğinde olduğu gibi küme yapıları, sapan değerler gözlenebilmektedir. Ayrıca PolyViz grafiğinde verinin her değişkeni için veri dağılımları hakkında da bilgiler elde edilebilmektedir. Veri dağılımları değişken eksenleri boyunca çizilen kısa çizgilerin dağılımından anlaşılmaktadır. Çünkü bu kısa çizgilerin her biri veri setindeki veri birimlerine karşılık gelemektedir (Hoffman, 1999).

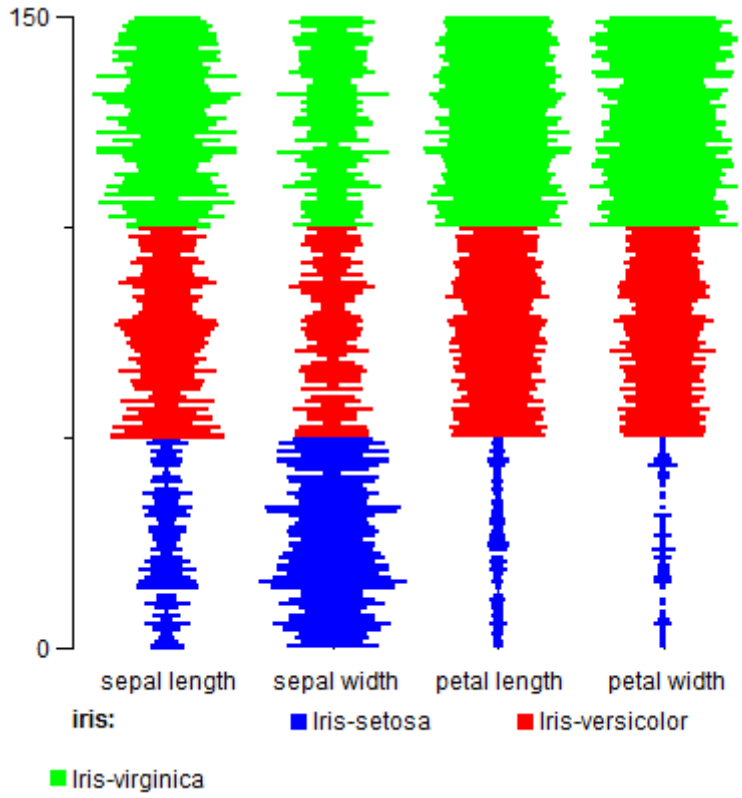


Şekil 3.12 Süsen veri seti için Orange programıyla çizilen PolyViz

3.3.3.2.4 Survey Grafiği (Survey Plot)

Survey grafiği çoklu çizgi grafiğinin geliştirilmiş şeklidir. Survey grafiğinde değişkenler dik eksenlerle temsil edilirler. Daha sonra verinin değişkenlerde aldığı değerleri, bu dik eksenlere büyüklükleriyle orantılı olarak yatay çizgilerle çizilirler.

Survey grafiği değişkenlerle olan korelasyonları görmemiz açısında kullanışlı olabilmektedir. Ayrıca veri setimizin sınıfları farklı renklerle gösterilerek küme yapıları görsel bir şekilde gözlemlenebilir.



Şekil 3.13 Süsen veri seti için Orange programıyla çizilen Survey grafiği

Şekil 3.13' de süsen veri seti için çizilen Survey grafiği bulunmaktadır. Grafikden de anlaşıldığı üzere sepal length, petal length ve petal width arasında güçlü bir korelasyon bulunmaktadır. Ayrıca sınıfları bilinen üç bitki cinsinin de (versicolor ve virginica benzer özellikler göstermek üzere) birbirilerine göre farklı küme yapıları gösterdiği gözlenmiştir.

3.3.3.2.5 Paralel Koordinatlar (Parallel Coordinates)

Kartezyen koordinat sistemlerinde eksenler birbirilerine diktirler. Bu yüzden biz bilgisayar ekranlarında ya da kâğıtta üç boyutu algılayabiliriz. Onun yerine birbirilerine paralel eksenler çizerek çok sayıda ekseni aynı iki boyutlu düzlemde gösterebiliriz. Bu sayede dik eksenleri kullanmak yerine paralel eksenleri kullanmış oluruz. Bu teknik ilk olarak 1985 yılında Inselberg tarafında bulunmuş olup, Wegman tarafından 1986 yılında çok boyutlu verileri analiz etmek ve görselleştirmek için geliştirilmiştir (Martinez ve Martinez 2005).

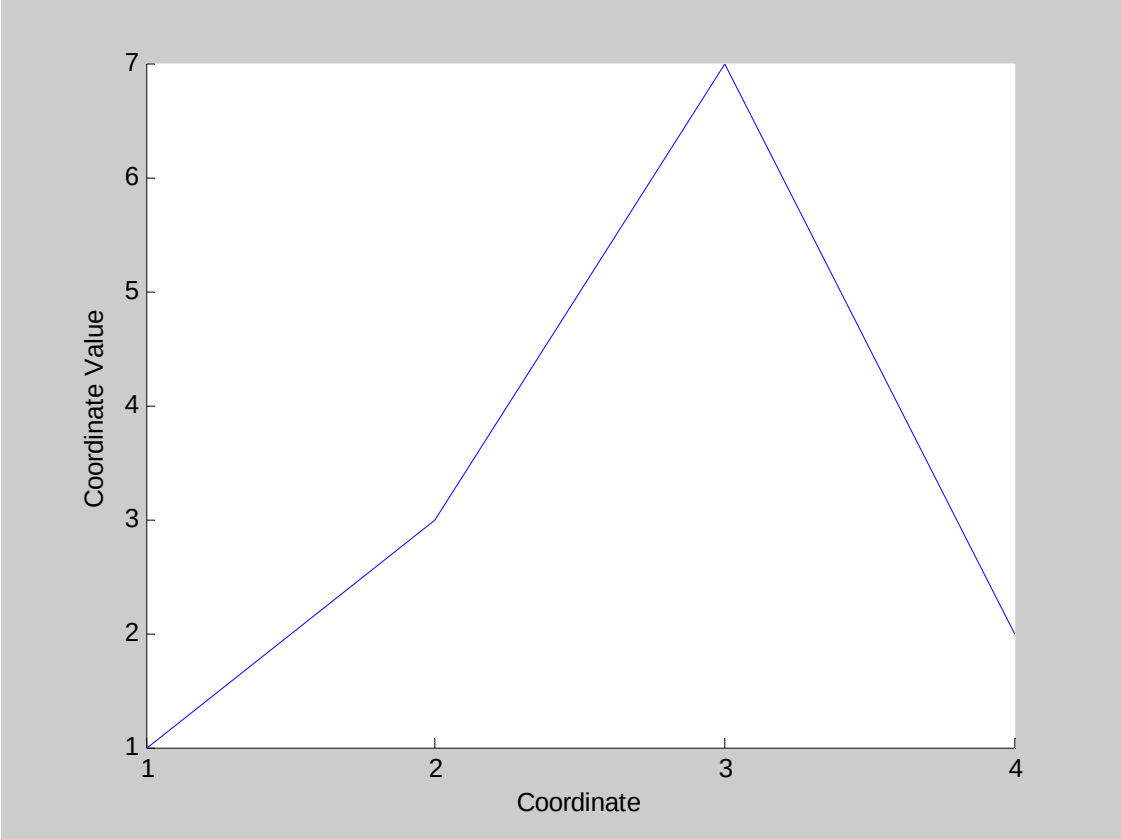
Paralel Koordinatlar, p boyutlu $x_1, x_2, \ldots, x_p$ veri setini 2 boyutlu uzaya haritalayan, p adet birbirine paralel konumlandırılmış eksenlerden oluşan bir görselleştirme tekniğidir. Her eksen veri setine ait bir değişken ile ilişkilendirilmiştir. Eksenler üzerindeki değerler işaretlendikten sonra bu değerler düz çizgiler ile birleştirilirler. Bu tekniğin en büyük dezavantajı birkaç bin adetten daha fazla nesne içeren veri setleri için uygun olmamasıdır. Nesne sayısı arttıkça üst

üste binen çok sayıda çizgi görüntüyü yorumlanabilir olmaktan çıkarmaktadır (Bilgin ve Çamurcu, 2007).

Örnek 3.5 :

Elimizde $x = [1,3,7,2]$ şeklinde bir gözlemden oluşan 4 değişkenli veri seti bulunsun. Bunun paralel koordinatını çizelim.

```
x=[1,3,7,2];
parallelcoords(x)
```



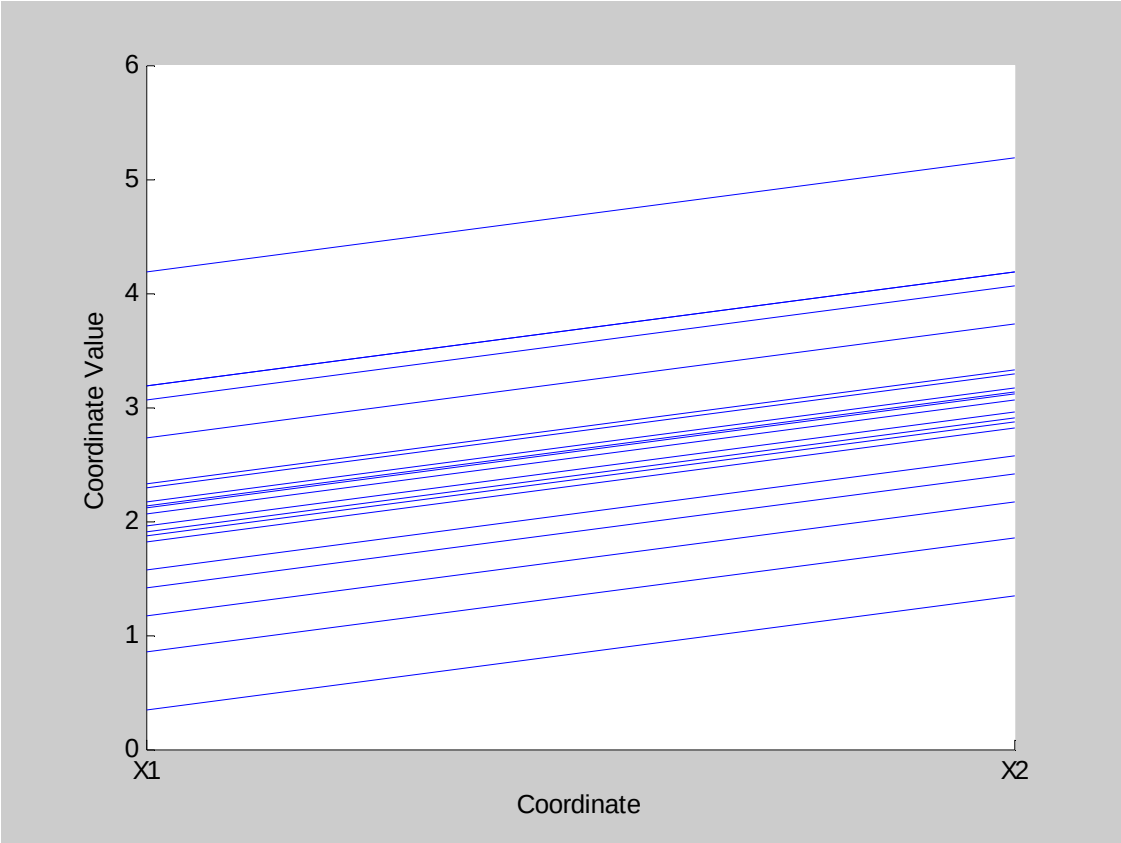
Şekil 3.14 Paralel Koordinat

Paralel koordinatlar kümeleri, sapan değerleri, değişken çiftleri arasındaki korelasyonları bulmada kullanılabilirler. Kategorik değişkenler paralel eksenlerden bir tanesi olmak üzere grupları ya da sınıfları belirtmekte kullanılabilir. Burada her kategori için değişik renkler kullanılarak grupları veya sınıfları ayrıştırabiliriz (Martinez ve Martinez 2005).

Örnek 3.6 :

Paralel koordinatlarda korelasyonları görebilmek için korelasyonu 1 ve -1 olan iki değişkenli normal dağılımdan 20 tane rastgele değer üretelim. Bu üretilen 20 değerin paralel koordinatlarını çizerek korelasyonların paralel koordinatlarda nasıl gözüktüğüne bakalım.

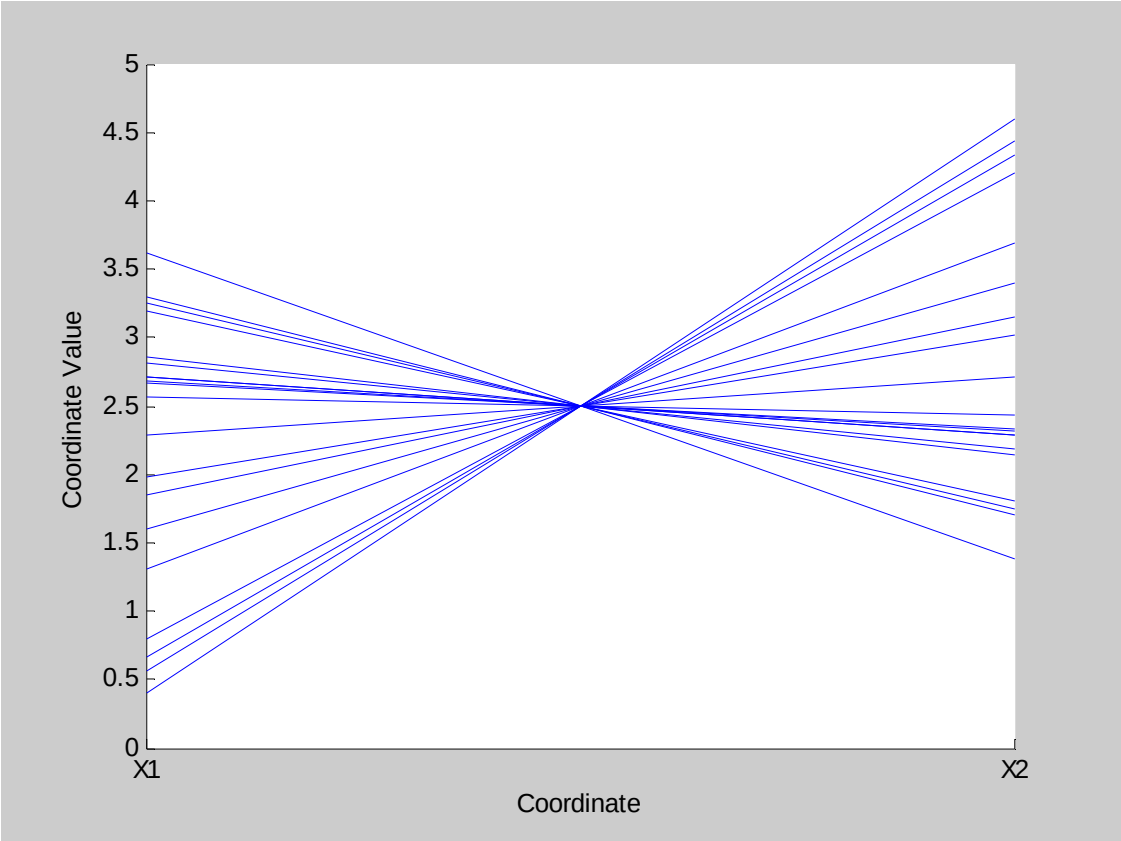
```
mu=[2,3];                       % ortalamalar vektörü
sigma=[1,1;1,1];                % kovaryans matrisi
r = mvnrnd(mu,sigma,20);        % iki değişkenli normal dağılımdan sayı üretme
parallelcoords(r)
```



Şekil 3.15 Korelasyonu 1 olan ikili normal değişkenin paralel koordinatı

Şekil 3.15' de aralarında 1 korelasyon bulunan iki değişkenin paralel koordinat grafiği bulunmaktadır. Şekil 3.15 ' den de gözüktüğü gibi birinci ve ikinci değişken arasında doğrusal bir ilişki bulunmaktadır. Paralel koordinatlar birbirilerini kesmemektedir.

```
mu=[2,3];                       % ortalamalar vektörü
sigma=[1,-1;-1,1];
r = mvnrnd(mu,sigma,20);        % kovaryans matrisi
parallelcoords(r)
```



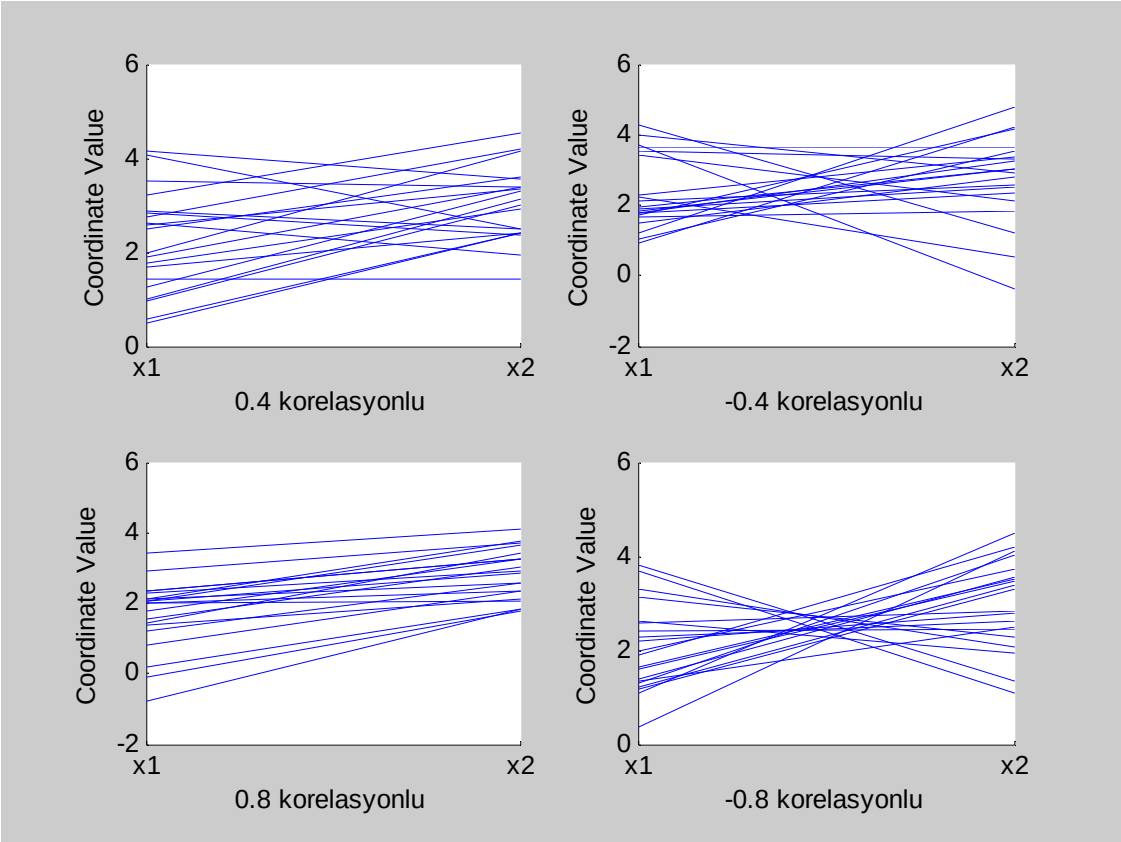
Şekil 3.16 Korelasyonu -1 olan ikili normal değişkenin paralel koordinatı

Şekil 3.16' da aralarında -1 korelasyon bulunan iki değişkenin paralel koordinat grafiği bulunmaktadır. -1 korelâsyonlu paralel koordinat, 1 korelasyonlu paralel koordinattan oldukça farklı bir yapı sergilemektedir. Paralel koordinatların tümü bir noktada kesişmektedir.

Kovaryans matrisini değiştirerek aralarında farklı korelasyonlar bulunan veriler üreterek paralel koordinatların nasıl gözüktüğüne bakalım. Bunun için 0.4, -0.4, 0.8, -0.8 korelasyonlu veriler üretelim.

```
mu=[2,3];                    % ortalamalar vektörü
sigma=[1,0.4;0.4,1];         % kovaryans matrisi
r = mvnrnd(mu,sigma,20);     % iki değişkenli normal dağılımdan sayı üretme
subplot(2,2,1)
parallelcoords(r)
xlabel('0.4 korelasyonlu')
sigma=[1,-0.4;-0.4,1];
r = mvnrnd(mu,sigma,20);
subplot(2,2,2)
parallelcoords(r)
xlabel('-0.4 korelasyonlu')
sigma=[1,0.8;0.8,1];
r = mvnrnd(mu,sigma,20);
subplot(2,2,3)
parallelcoords(r)
xlabel('0.8 korelasyonlu')
sigma=[1,-0.8;-0.8,1];
r = mvnrnd(mu,sigma,20);
```

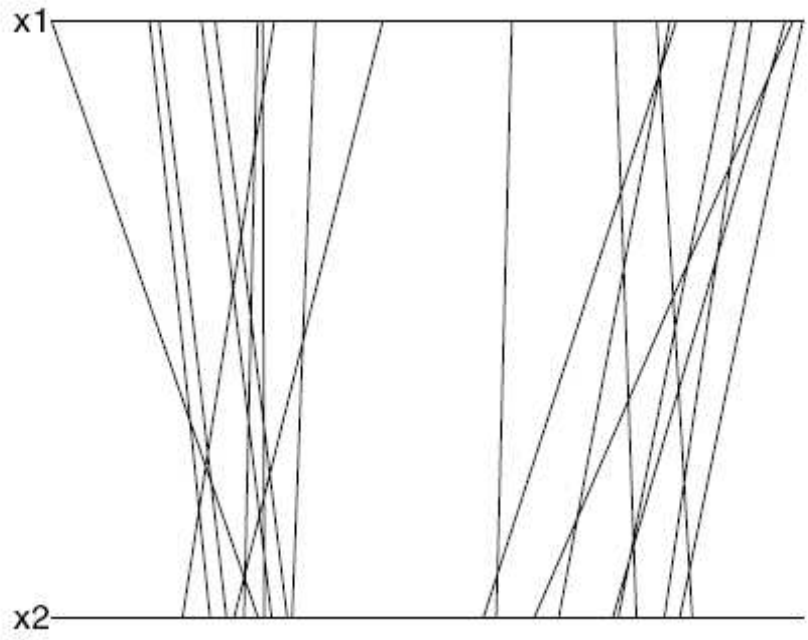
```
subplot(2,2,4)
parallelcoords(r)
xlabel('-0.8 korelasyonlu')
```



Şekil 3.17 Farklı korelasyonlu ikili normal değişkenin paralel koordinatları

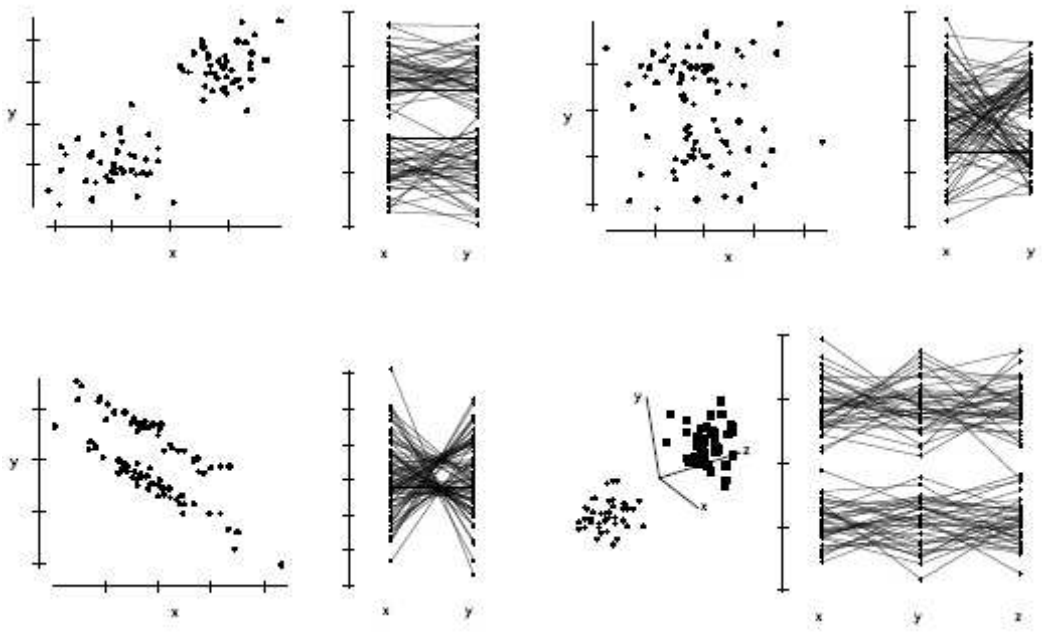
Şekil 3.17' den de gözüktüğü gibi pozitif korelasyonlar artıkça değişkenler arasındaki çizgiler bir birine paralel uzanmaya, negatif korelasyonlar arttıkça ise değişkenler arasındaki çizgiler bir nokta civarında kesişmeye başlarlar.

Paralel koordinatların değişkenler arasındaki ilişkileri göstermekte kullanıldığını gördük. Hatta paralel koordinatlar boyutlardaki değişkenlerin küme yapılarını göstermekte de kullanılabilirler. Şekil 3.18' e bakacak olursak iki değişkenli verimizde yani x_1, x_2 eksenlerinde paralel çizgiler arasında göze çarpan bir boşluk bulunmaktadır. Buradan hareketle değişkenlerin her ikisinde de ikili küme yapılarının gözüktüğü anlaşılır. Şekil 3.19' a da bakacak olursak iki değişkenli verimizde sadece x_1 paralel koordinat ekseninde göze çarpan bir boşluk bulunmaktadır. Buradan hareketle de sadece x_1 değişkenine göre verimizde ikili bir küme yapısının olduğu anlaşılır.



Şekil 3.18 x1, x2 eksenlerine göre küme (Martinez ve Martinez, 2002)

Şekil 3.19 x1 eksenine göre küme (Martinez ve Martinez, 2005)



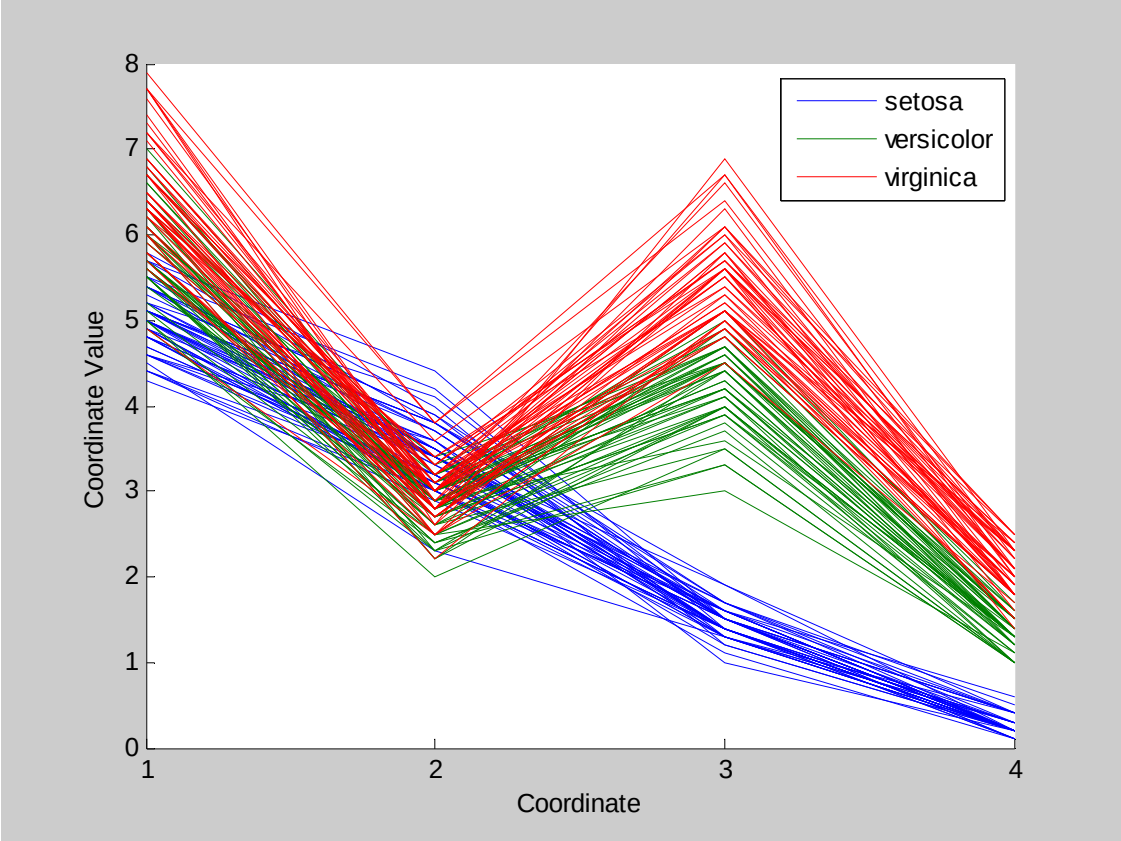
Şekil 3.20 Paralel koordinatların küme yapılarını göstermesi (Unwin vd., 2006)

Şekil 3.20’ da farklı değişkenler için serpilme ve paralel koordinat grafikleri bulunmaktadır. Şekilden de gözüktüğü gibi paralel koordinatlar küme yapılarını, korelasyonları, sapan değerleri göstermede oldukça başarılıdırlar.

Örnek 3.7 :

Süsen veri seti için paralel koordinatları kullanarak bitki cinslerinin farklılıkları konusunda bazı kanıtlar elde edebiliriz.

```
load fisheriris
parallelcoords(meas,'group',species)
```



Şekil 3.21 Süsen veri seti için parelel koordinat

Şekil 3.21 de setosa, versicolor ve virginica süsen bitkisi cinsleri farklı renklerle çizilmiştir. Şekil 3.21' den gözüktüğü üzere bu üç cins farklı özelikler sergilemektedirler. Özelikle setosa bitki cinsi diğer bitki cinslerine göre oldukça farklı bir yapı göstermektedir. Versicolor ve virginica bitki cinsleri ise bir birilerine benzemektedirler.

3.3.3.2.6 Andrews Eğrileri (Andrews Curves)

Andrews eğrileri çok boyutlu verileri görselleştirmek için bir yöntem olarak 1972 yılında Andrews tarafından geliştirilmiştir. Adrews eğrilerinde gözlem değerleri aşağıdaki denklem (3.1) deki fonksiyon kalıbı kullanılarak dönüştürülürler. Dönüşen bu değerlerin daha sonra çizgi grafikleri çizilerek Andrews eğrilerine ulaşılır.

$$f_x(t) = x_1/\sqrt{2} + x_2 \sin t + x_3 \cos t + x_4 \sin 2t + x_5 \cos 2t + ..., \qquad -\pi < t < +\pi \qquad (3.1)$$

Burada x_1, x_2, … verilerimizin değişkenleridir.

Örnek 3.8 :

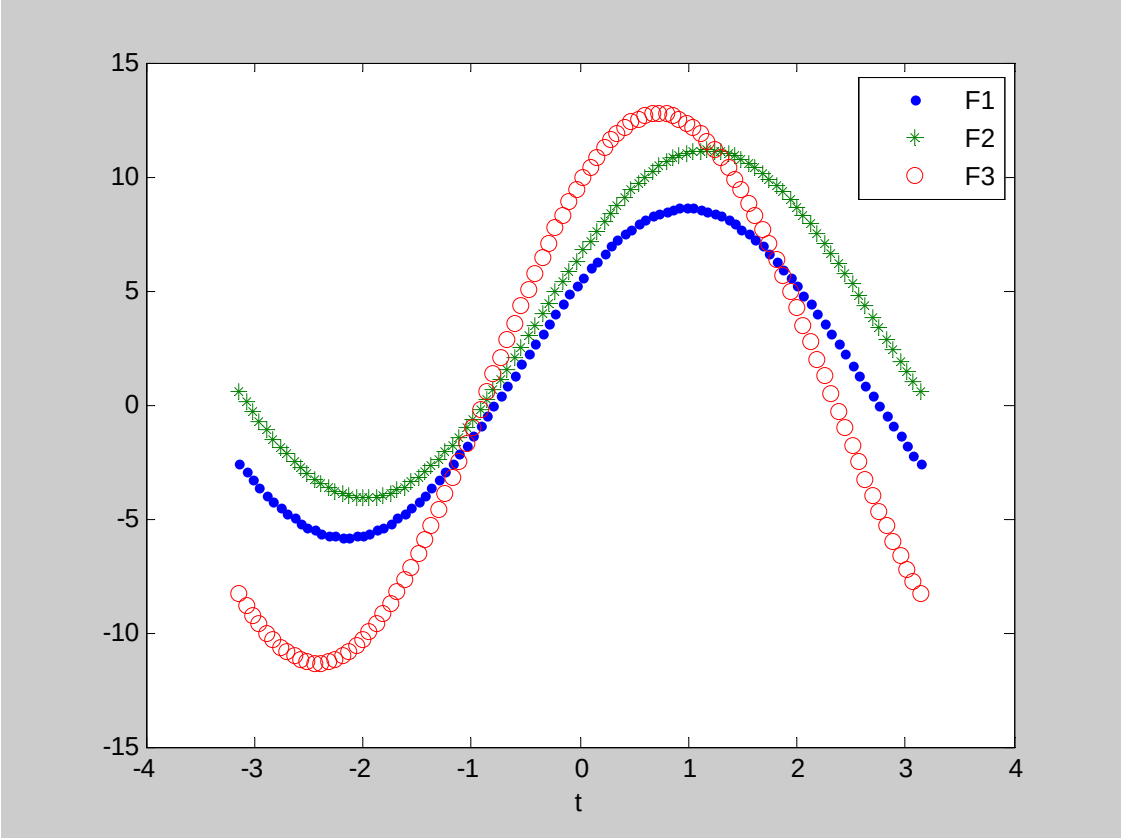
Andrews eğrilerinin nasıl çizildiğini göstermek için basit bir örnek verelim. Bunun için x_1, x_2, x_3 gözlem değerlerinden oluşan 3 değişkenli verileri kullanalım.

$x_1 = (2,6,4)$
$x_2 = (5,7,3)$
$x_3 = (1,8,9)$

Gözlem değerlerimize denklem (3.1)' deki dönüşümü aşağıdaki gibi uygulayalım.

$$f_{x_1}(t) = 2/\sqrt{2} + 6\sin t + 4\cos t$$
$$f_{x_2}(t) = 5/\sqrt{2} + 7\sin t + 3\cos t$$
$$f_{x_3}(t) = 1/\sqrt{2} + 8\sin t + 9\cos t$$

```
t = linspace(-pi,pi);
% Her gözlem değeri için elde edilen dönüşüm değerleri
f1 = 2/sqrt(2)+6*sin(t)+4*cos(t);
f2 = 5/sqrt(2)+7*sin(t)+3*cos(t);
f3 = 1/sqrt(2)+8*sin(t)+9*cos(t);
plot(t,f1,'.',t,f2,'*',t,f3,'o')
legend('F1','F2','F3')
xlabel('t')
```



Şekil 3.22 x1, x2, x3 verileri için Andrews eğrisi

Andrews eğrileri orijinal veri setinin ortalamasını ve varyansını içersinde barındırırlar. Andrews eğrilerinin kullandığı fonksiyon kalıplarından elde edilen değerlerinin birbirine yakın olması gözlem değerlerinin birbirine yakın olduğunu, birbirine uzak olması gözlem

değerlerinin de birbirine uzak olduğunu gösterir. Buradan hareketle Andrews eğrileri verilerin küme yapılarının anlaşılmasında da, aşırı değerlerin tespitinde de kullanılabilirler. (Martinez ve Martinez, 2005)

Andrews eğrilerinin özellikleri:

Andrews eğrilerinin çeşitli kullanışlı özellikleri bulunmaktadır. Andrews eğrilerinin bazı özellikleri aşağıdaki gibidir : (Garcia-Osorio ve Fyfe., 2005)

1. Andrews eğrileri veri setimizin ortalamasını içinde barındırmaktadır.

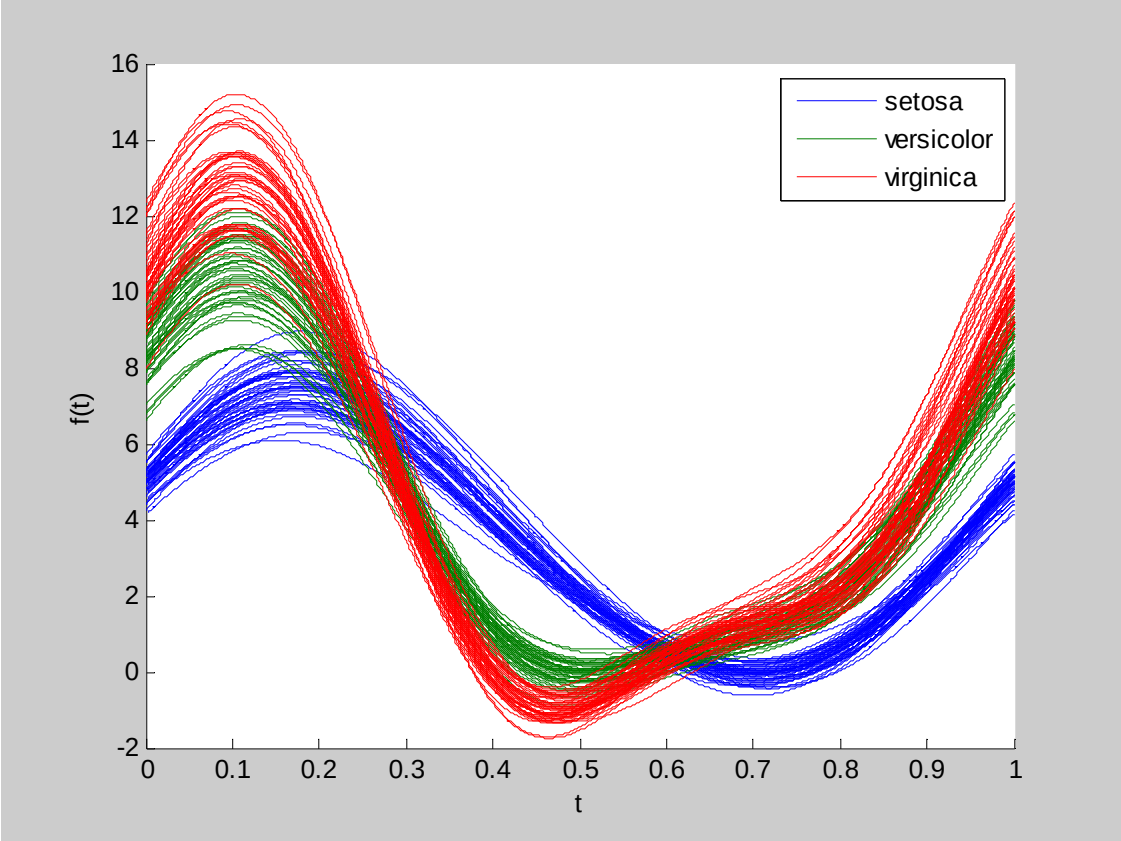
   $$f_{\bar{x}}(t) = \frac{1}{N}\sum_{i=1}^{N} f_{x_i}(t)$$

2. Andrews eğrilerinde $f_x(t)$, $f_y(t)$ iki fonksiyon arasındaki uzaklık $\|f_x(t) - f_y(t)\| = \int_{-\pi}^{\pi} [f_x(t) - f_y(t)]^2 dt$ olmak üzere; x_i, y_i noktaları arasındaki uzaklık $\|f_x(t) - f_y(t)\| = \pi\sum_{i=1}^{d}(x_i - y_i)^2 = \pi\|x - y\|^2$ ' dir. Bu da iki veri noktası arasındaki uzaklığın iki Andrews fonksiyonu arasındaki uzaklıkla orantılı olduğunu gösterir.
3. Doğrusal ilişki: Eğer bir y noktası x ile z' yi birleştiren bir çizgi arasında kalıyorsa bütün t' ler için $f_y(t)$, $f_x(t)$ ile $f_z(t)$ fonksiyonları arasında kalır.

Örnek 3.9 :

Süsen veri seti için Andrews eğrilerini kullanarak süsen bitki cinslerinin farklılıkları konusunda bazı kanıtlar elde edebiliriz.

```
load fisheriris
andrewsplot(meas,'group',species);
```



Şekil 3.23 Süsen veri seti için Andrews eğrisi[1]

Şekil 3.23 de setosa, versicolor ve virginica süsen bitkisi cinsleri farklı renklerle çizilmiştir. Şekil 3.23' den gözüktüğü üzere bu üç cins farklı özelikler sergilemektedir. Özelikle setosa bitki cinsi diğer bitki cinslerine göre oldukça farklı bir yapı göstermektedir. Versicolor ve virginica bitki cinsleri ise bir birilerine benzemektedirler.

3.3.3.2.7 Permutasyon Turları (Permutation Tour)

Paralel koordinat ve Andrews eğrilerine getirilen en büyük eleştirilerden bir tanesi de çizilen şeklin biçiminin değişkenlerin sıralarına olan bağımlılığıdır. Yani değişken sıraları değiştikçe paralel koordinatların ve Andrews eğrilerinin şekilleri değişecektir.

Paralel koordinatlarda değişkenleri temsil eden eksenlerin pozisyonları, değişken çiftleri arasındaki ilişkileri göstermesi açısından önemlidir. Ancak bir biri ardına sıralanan değişkenler için değişkenler arasındaki ilişkileri paralel koordinatlarla tespit etmek ve karşılaştırmak mümkün değildir. Andrews eğrilerinde ise dönüştürme işlemi için kullanılan denklemde ilk sıraya yerleşen değişken grafiğimizin üzerinde en büyük ağırlığa sahip olan değişkendir. Yani değişkenlerin sırası değiştikçe, ilk sırada olan değişkenin, grafiğin şekline

[1] MATLAB R2007a programı Andrews eğrilerini çizerken t' için $0 < t < +1$ açık aralığını kullanır.

olan ağırlığı fazla olacak şekilde grafiğimizin biçimi değişecektir. Sonuç olarak değişken sayısının permutasyonu kadar farklı sayıda grafik çizilecektir. Bu yüzden değişkenlerin sırasının değişmesine bağlı olarak permutasyon turları geliştirilmiştir (Martinez ve Martinez 2005).

İki tip permutasyon turu bulunmaktadır. Birinci tip permutasyon turunda değişkenlerin permutasyonları alınarak Andrews eğrileri ve paralel koordinatlar çizilmektedir. Ancak ikili değişkenler arasındaki ilişkileri gösteren paralel koordinatlarda, permutasyonları alınan değişkenler, ikili değişken çiftlerinin gösterimlerinde tekrara sebep olmaktadır. Örneğin, 1-2-3 ve 3-2-1 farklı sıralanmış değişkenler olmak üzere; iki sıralamada da paralel koordinatlarla aynı bilgilere ulaşılmaktadır. Her iki sıralamada da 1 ile 2 ve 3 ile 2 değişkenleri arasındaki ilişkiler gözlenebilmektedir. Bu tekrardan kurtulmak için minimum sayıda permutasyonu sağlayan kısmi permutasyon turları geliştirilmiştir (Martinez ve Martinez 2005).

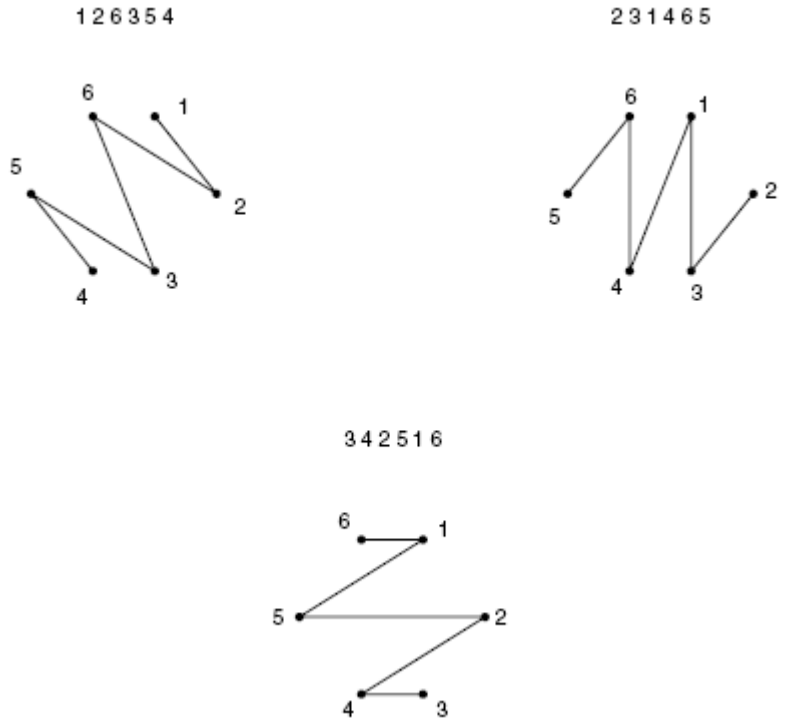
Kısmi permutasyon turlarında değişkenlerin dizilimlerini bulmak için denklem (3.2) kullanılır. Burada $p_1 = 1$ dir.

$$p_{k+1} = \left(p_k + (-1)^{k+1} k\right) \quad \text{mod p} \quad k = 1,2,\ldots, p-1 \tag{3.2}$$

Kısmi permutasyon turlarında minimum permutasyon sayısını bulmak için $\left[\frac{p-1}{2}\right]$ en büyük tam sayı fonksiyonu kullanılır. Ayrıca farklı permutasyonlar için faklı başlangıç noktaları denklem (3.3) kullanılarak bulunur. Burada $p_k^1 = p_k$ ve $[\bullet]$ en büyük tam sayı fonksiyonudur.

$$p_k^{(j+1)} = \left(p_k^j + 1\right) \quad \text{mod p} \quad j = 1,2,\ldots,\left[\frac{p-1}{2}\right] \tag{3.3}$$

Örneğin, 6 değişkenli veri için (yani p=6) $\left[\frac{6-1}{2}\right] = 3$ tane farklı dizilimli permutasyon bulunur. Bu 3 farklı dizilim Şekil 3.24’ da zig-zaglarla gösterilmiştir.



Şekil 3.24 6 değişkenli veri seti için minimum permutasyon dizilimi

(Martinez ve Martinez, 2005)

Birinci zig-zag için değişken dizilimleri aşağıdaki gibi bulunur.

$$p_2 = \left(p_1 + (-1)^{1+1}1\right) \text{ mod p } \Rightarrow \; p_2 = 2 \text{ mod p} \tag{3.4}$$

$$p_3 = \left(p_2 + (-1)^{2+1}2\right) \text{ mod p } \Rightarrow \; p_3 = 0 \text{ mod p} \Rightarrow \; p_3 = p \text{ mod p} \tag{3.5}$$

$$p_4 = \left(p_3 + (-1)^{3+1}3\right) \text{ mod p } \Rightarrow \; p_4 = 3 \text{ mod p} \tag{3.6}$$

$$p_5 = \left(p_4 + (-1)^{4+1}4\right) \text{ mod p } \Rightarrow \; p_5 = -1 \text{ mod p} \Rightarrow \; p_5 = 5 \text{ mod p} \tag{3.7}$$

$$p_6 = \left(p_5 + (-1)^{5+1}5\right) \text{ mod p } \Rightarrow \; p_6 = 4 \text{ mod p} \tag{3.8}$$

Yukarıdaki sonuçlara göre birinci zig-zagın dizilimleri 1-2-6-3-5-4 şeklinde olur. Farklı zig-zag dizilimlerini bulabilmek içinse denkem (3.3) ve denklem (3.2) kullanılmalıdır. Denklem (3.3) sayesinde ikinci zig-zag için farklı bir başlangıç noktası elde edilir.

$$p_k^1 = p_k \tag{3.9}$$

$$p_k^2 = \left(p_k^1 + 1\right) \text{ mod p } \Rightarrow \; p_k^2 = 2 \text{ mod p} \tag{3.10}$$

İkinci zig-zag için başlangıç noktası 2 olarak bulunur.

Denklem (3.2)’ nin MATLAB komutları aşağıdaki gibi yazılabilir.

```
p = 6;
N = ceil((p-1)/2);
% Get the first sequence.
P(1) = 1;
for k = 1:(p-1)
tmp(k) = (P(k) + (-1)^(k+1)*k);
P(k+1) = mod(tmp(k),p);
end
% To match our definition of 'mod':
P(find(P==0)) = p;
```

Denklem (3.3)’ ün MATLAB komutları aşağıdaki gibi yazılabilir.

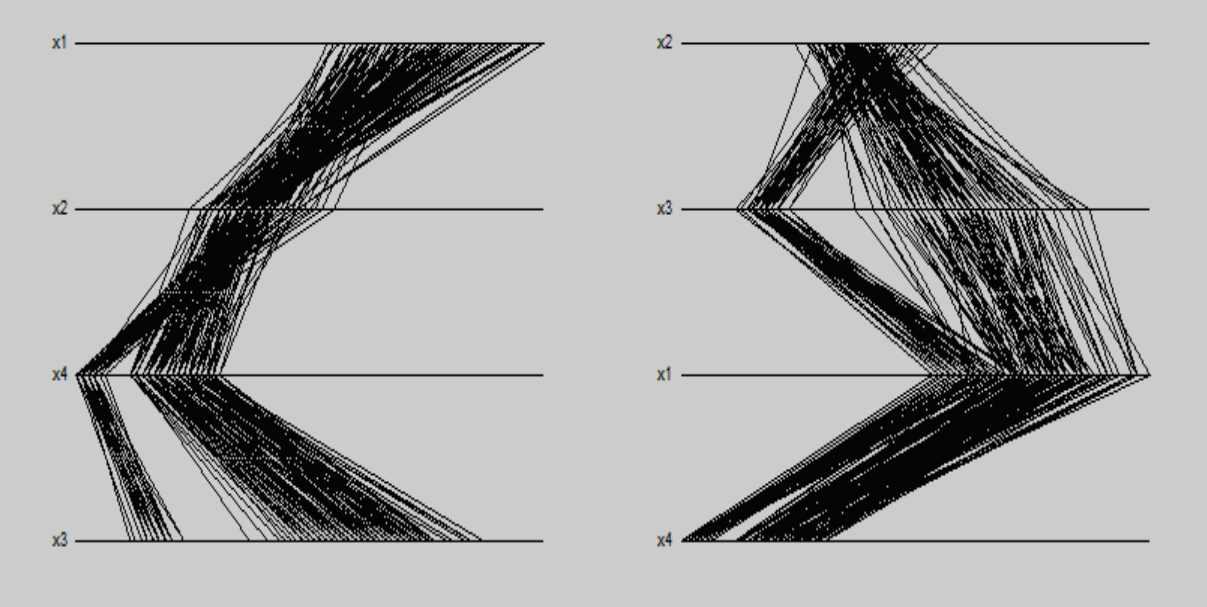
```
for j = 1:N;
P(j+1,:) = mod(P(j,:)+1,p);
ind = find(P(j+1,:)==0);
P(j+1,ind) = p;
end
```

Örnek 3.10 :

Süsen veri seti için paralel koordinatları kullanarak kısmi permutasyon turlarını çizelim.

Süsen veri setinde 4 tane değişken olduğu için $\left[\frac{(p=4)-1}{2}\right]=\left[\frac{4-1}{2}\right]=2$ dir. Yani 2 minimum permutasyon sayısı söz konusudur.

```
load fisheriris
permtourparallel(meas)
```



Şekil 3.25 Süsen veri seti için kısmi permutasyon turları

3.3.3.2.8 Temel Bileşenler Analizi (Principal Component Analysis - PCA)

“Temel bileşenler analizi orijinal p değişkenin varyans yapısını daha az sayıda ve bu değişkenlerin doğrusal bileşenleri olan yeni değişkenlerle ifade etme yöntemidir. Aralarında korelasyon bulunan p sayıda değişkenin açıkladığı yapıyı, aralarında korelasyon bulunmayan ve sayıca orijinal değişkenin sayısından daha az sayıda (p>k) orijinal değişkenlerin doğrusal bileşenleri olan değişkenlerle ifade etme yöntemine temel bileşenler analizi denir (Özdamar, 2004).”

“Temel bileşenler analizi sonucunda, p boyutlu (değişken) uzayı çok iyi temsil eden k tane yeni dik (korelasyonsuz) değişken (bileşen ya da öz vektör) elde edilir. Elde edilen temel bileşenlerin birimi yoktur. p tane değişkenin taşıdığı bilginin k tane (p>k) yeni değişkenle açıklanması ise temel bileşenlerin ana amacını oluşturur (Alpar, 2003).”

Temel bileşenler analizinin amaçlarını aşağıda ki gibi yazabiliriz:

1. Veri indirgemesi
2. Tahmin yapmak
3. Veri setini bazı yöntemler için analiz edilebilir hale dönüştürmek
4. Veri görselleştirmesi

Temel bileşenler analizi, ufak veri kayıplarına göz yumarak oluşturduğu boyut indirgemesiyle çok boyutlu veri yapılarının görselleştirilmesini sağlayabilir. Bu sayede veri seti içersinde küme yapıları, sapan değerler görsel bir şekilde gözlemlenebilir. Temel bileşenler analizi aynı zamanda çok geniş incelemeler için bir ara adım olma özelliği de taşır. Örneğin, çoklu regresyon analizinde çoklu bağımlılık olması durumunda, değişkenler arası bağımlılık yapısını yok etmek için veriler temel bileşenlere göre temel bileşen skorlarına dönüştürülür ve yeni elde edilen verilere çoklu regresyon uygulanır (Özdamar, 2004). Kümeleme analizinde ise korelasyonlu değişkenlerle oluşturulan kümelerin tam ayrıştırılamaması durumunda temel bileşenler analizi sonucunda elde edilen skorlar kullanılarak kümeleme analizi gerçekleştirilebilir. Boyut sayısı fazla olduğu zaman; kümeleme ve sapan değer bulmada kritik öneme sahip olan iki nokta arasındaki uzaklık ve yoğunluğun tanımları daha az anlamlı hale gelmektedir. Bunun için temel bileşenler analizi kümeleme analizlerinde veriyi hazırlamada kritik öneme sahip olabilir.

3.3.3.2.8.1 Temel Bileşenlerinin Elde Edilmesi

Cebirsel olarak temel bileşenler p tane X_1, X_2,…,X_p rastgele değişkenin doğrusal bileşimidir. Geometrik olarak bu doğrusal bileşenler, koordinat eksenleri, X_1, X_2,…,X_p olan orijinal sistemi döndürerek bulunan yeni koordinat sisteminin seçimini gösterirler. Yeni eksenler maksimum değişkenliği içeren yönleri gösterirler ve birlikte değişim yapısının daha basit ve daha az sayıda değişken ile açıklamasını sağlarlar (Özdamar, 2004).

Değişkenler kümesinin doğrusal bileşenlerini bulmak için korelasyon ya da kovaryans matrisinin özdeğer ve özvektörleri kullanılır. Özdeğerlere karşılık gelen özvektörler birbirine diktir (birbirlerinden bağımsızdırlar). Bu özellikten yararlanarak temel bileşenler, özdeğerlerin büyüklük sırası izlenerek önem sırasına göre hesaplanır. (Özdamar, 2004)

Varyans kovaryans matrisinin özdeğer ve özvektörleri aşağıdaki gibi bulunur:

$(\Sigma - \lambda I) = 0$ eşitliğinden yararlanarak $\lambda_1, \lambda_2, \ldots, \lambda_p$ öz değerleri bulunur.

$(\Sigma - \lambda_i I)e_i = 0$, $i = 1, \ldots, p$ olmak üzere i. özdeğer için i. e_i özvektörü bulunur.

$i, j = 0, \ldots, p$ ve $i \neq j$ olmak üzere, e özvektörleri, $e_i e_i^T = 1$ $e_i e_j^T = 0$ özelliklerini sağlarlar. Böylece özdeğerlere karşılık gelen birbirine dik yani bağımsız özvektörler elde edilmiş olur.

$X^{T'} = [X_1, X_2, ..., X_p]$ rastgele vektörü $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_p \geq 0$ öz değerleri bulunan Σ varyans kovaryans matrisine sahip olsun.

Değişkenlere ilişkin doğrusal bileşenler;

$$\begin{aligned} Y_1 &= e_1^T X^T = e_{11}X_1 + e_{21}X_2 + ... + e_{p1}X_p \\ Y_2 &= e_2^T X^T = e_{12}X_1 + e_{22}X_2 + ... + e_{p2}X_p \\ &\ldots \\ Y_{p1} &= e_p^T X^T = e_{1p}X_1 + e_{2p}X_2 + ... + e_{pp}X_p \end{aligned} \qquad (3.11)$$

olarak hesaplanır. $Y = eX$' de e' nin sütunları temel bileşenlerin katsayılarına karşılık gelmektedir.

Her bir temel bileşenin varyansı;

$$Var(Y_i) = e_i^T \Sigma e_i \qquad i = 1,2,\ldots, p \qquad (3.12)$$

ve kovaryansı;

$$Cov(Y_i, Y_k) = e_i^T \Sigma e_j \quad i = 1,2,\ldots, p \quad i \neq j \tag{3.13}$$

olarak hesaplanır.

Temel bileşenler, $Y_1, Y_2, \ldots, Y_p$ birbirleriyle ilişkili değillerdir yani korelasyonsuzdur. Her temel bileşen öz değerlere karşılık gelir ve varyans büyüklüklerine göre sıralanırlar. Orijinal değişkenlik ile p temel bileşenin varyansları toplamı birbirine eşittir. Ayrıca toplam varyans özdeğerler toplamına eşittir.

$$ToplamVaryans = \lambda_1 + \lambda_2 + \ldots + \lambda_p = Var(Y_1) + Var(Y_2) + \ldots + Var(Y_p) \tag{3.14}$$

$$ToplamVaryan = Var(X_1) + Var(X_2) + \ldots + Var(X_p) \tag{3.15}$$

Temel Bileşenlerin Varyansı $Var(Y_i) = e_i^T \Sigma e_i$ (3.16)

İlk temel bileşen maksimum varyanslı doğrusal bileşendir. İkinci temel bileşen ikinci büyüklükte varyansa sahip olan bileşendir. Sırasıyla temel bileşenler özdeğerlerin sırasına göre yani orijinal veri setinin varyans açıklayıcılık oranlarına göre sıralanırlar.

Temel bileşenleri aşağıdaki gibi belirlenir:

1. temel bileşen, varyansı, yani $Var(e_1 X)$' i $e_1^T e_1 = 1$ olacak şekilde maksimize eden $e_1^T X$ doğrusal bileşendir. 2. temel bileşen, varyansı, yani $Var(e_2 X)$' i $e_2^T e_2 = 1$ ve $Cov(e_1^T X, e_2^T X) = 0$ olacak şekilde maksimize eden $e_2^T X$ doğrusal bileşendir. k. temel bileşen, k>i olmak üzere, varyansı, yani $Var(e_i X)$' i $e_i^T e_i = 1$ ve $Cov(e_i^T X, e_k^T X) = 0$ olacak şekilde maksimize eden $e_i^T X$ doğrusal bileşendir.

$X = [X_1, X_2, \ldots, X_p]$ rastgele vektörünün Σ varyans kovaryans matrisi olsun. Σ' nın özdeğer ve özvektör çiftleri ise $(\lambda_1, e_1), (\lambda_2, e_2), \ldots, (\lambda_p, e_p)$ olsun. $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_p \geq 0$ olmak üzere i. temel bileşen:

$$Y_i = e_i^T X = e_{1i} X_1 + e_{2i} X_2 + \ldots + e_{pi} X_p \qquad i = 1,2,\ldots, p \tag{3.17}$$

Bu durumda aşağıdaki durumlar gerçekleşir.

$$Var(Y_i) = e_i^T \Sigma e_i = \lambda_i \qquad i = 1,2,\ldots, p \tag{3.18}$$

$$Cov(Y_i, Y_k) = e_i^T \Sigma e_k = 0 \qquad i \neq k \tag{3.19}$$

Eğer bazı λ_i ler eşitse, e_i katsayı vektörleri de eşit olacaktır. Bu durumda Y_i ler tek olmazlar ve bazı temel bileşenler birbirilerine eşit olurlar.

$e_1^T = \left[e_1, e_{2i}, \ldots e_{pi}\right]$ özvektörlerinin her bir elemanı orijinal değişkenlerin temel bileşene olan katkısını belirtmede kullanılabilirler. Diğer bir değişle özvektör elemanı temel bileşen ile orijinal değişken arasındaki korelasyon katsayısıyla orantılı büyüklüklerdir.

3.3.3.2.8.2 Temel Bileşenlerin Hangi Matristen Elde Edileceğinin Seçilmesi

Temel bileşenler hem orijinal verilerden hem de standardize değerlerden elde edilebilir. Bu verilere varyans kovaryans ya da korelsyon matrisinden yararlanılarak temel bileşenler analizi uygulanabilir. Her iki matristen hesaplanan temel bileşenler birbirilerinden farklılık göstermektedir. Bu yüzden temel bileşenleri hesaplamada kullanılacak matrisin seçimi önemlidir.

Veri setinde bulunan p değişkenin ölçü birimleri birbirilerinden oldukça farklı ve değişkenlerin değişim aralıkları bir birilerinden büyük farklılık gösteriyorsa temel bileşenler analizi uygulamalarında standardize veri matrisinden ya da korelasyon matrisinden yararlanmak uygundur. Aksi durumda kovaryans matrisi kullanılabilir.

Standardize edilmiş değişkenlerin korelasyon matrisi ile varyans kovaryans matrisleri aynıdır. Benzer şekilde orijinal değişkenlerin korelasyon matrisi ile standardize edilmiş değişkenlerin korelasyon matrisleri de aynıdır. Bundan dolayı temel bileşenleri hesaplarken orijinal değişkenlerin korelasyon matrislerinin özdeğerlerine karşılık gelen özvektörlerini kullanmak ile standardize edilmiş olan değişkenlerin korelasyon matrislerinin özdeğerlerine karşılık gelen özvektörleri kullanmak arasında bir fark yoktur.

İspat : (Bilen, 2004)

$X_1^{'}$ ve $X_2^{'}$ değişkenleri sırasıyla X_1, X_2 değişkenlerinin standardize edilmiş halleri olsunlar. $X_1^{'}$ ve $X_2^{'}$ değişkenlerinin kovaryansını hesapladığımızda orijinal değişkenlerin korelasyonunu hesaplamış oluruz.

$$Cov\left(X_1^{'}, X_2^{'}\right) = E\left(X_1^{'}, X_2^{'}\right) - E\left(X_1^{'}\right)E\left(X_2^{'}\right) \tag{3.20}$$

$$Cov\left(X_1^{'}, X_2^{'}\right) = E\left(X_1^{'}, X_2^{'}\right) - \mu_{X_1^{'}}\mu_{X_2^{'}} \tag{3.21}$$

$$Cov\left(X_1^{'}, X_2^{'}\right) = E\left(X_1^{'}, X_2^{'}\right) - 0 \tag{3.22}$$

$$Cov\left(X_1^{'}, X_2^{'}\right) = E\left(X_1^{'}, X_2^{'}\right) \tag{3.23}$$

$$Cov\left(X_1^{'}, X_2^{'}\right) = E\left[\frac{X_1 - \mu_{X_1}}{\sigma_{X_1}} * \frac{X_2 - \mu_{X_2}}{\sigma_{X_2}}\right] \tag{3.24}$$

$$Cov\left(X_1^{'}, X_2^{'}\right) = \frac{1}{\sigma_{X_1}\sigma_{X_2}} Cov\left(X_1, X_2\right) = Cor\left(X_1, X_2\right) \tag{3.25}$$

3.3.3.2.8.3 Kaç Adet Temel Bileşenin Kullanılacağının Seçimi

Temel bileşenler analizinde verilerin kaç adet temel bileşenle temsil edileceğini belirlemek önemli sorunlardan bir tanesidir. Çünkü temel bileşenler analizi sonucunda orijinal veri setindeki değişkenlerin sayısı kadar temel bileşen elde edilir. Burada amaç varyans açıklama oranı düşük olan birkaç temel bileşeni dışlamaktır. Bu dışlama için değişik yöntemler söz konusudur. Bu yöntemler aşağıda sıralanmıştır.

1. Korelasyon ya da kovaryans matrisinden elde edilen özdeğerlerin 1 den büyük olanlarının sayısı kadar temel bileşen seçmek. Bu yöntem en sık kullanılan yöntemdir.
2. Temel bileşen sayısını belirlerken en az temel bileşen sayısının toplam varyansın 2/3' ünün (%67) açıklayabilecek kadar temel bileşen seçmek. Bu oran % 95' e kadar arttırılabilir. Fakat %67 oranından sonra açıklanan varyans oranını artırmak için çok sayıda temel bileşen seçilmesi söz konusu olacaksa oranı sınırlı tutmakta yarar varıdır.
3. Yamaç eğim testi yapılır. Bu testte kovaryans ya da korelasyon matrisinden elde edilen özdeğerlere ait çizgi ya da çubuk grafiği çizilir. Grafikte özdeğerlerin eğimlerine bakılır. Azalan değerlere göre bir eğim izleyen eğim çizgisinin eğiminin sabitleştiği ya da çok küçük azalan değerlere kavuştuğu noktaya kadar olan özdeğer sayısı kadar temel bileşen seçilir.

Araştırıcı veri matrisinin yapısına ve problemin özelliğine göre her üç yöntemi göz önüne alarak temel bileşen sayısına karar vermelidir. Genel kural olarak ikiden daha az temel bileşen ile çalışmamak gerekir (Özdamar, 2004).

Örnek 3.11 :

Süsen veri seti için temel bileşenlerimizi bulalım.

```
load fisheriris
[coefs,scores,variances,t2] = princomp(meas);
```

Princomp fonksiyonun birinci coefs çıktısı, 4 tane temel bileşenin katsayılarını içerir. Orijinal değişkenler bu 4 katsayıların doğrusal bileşimi şeklinde yazılarak yeni değişkenlere dönüştürülür.

Dört temel bileşen katsayıları aşağıda verilmiştir.

```
c4 = coefs
c4 =
    0.3614   -0.6566    0.5820    0.3155
   -0.0845   -0.7302   -0.5979   -0.3197
    0.8567    0.1734   -0.0762   -0.4798
    0.3583    0.0755   -0.5458    0.7537
```

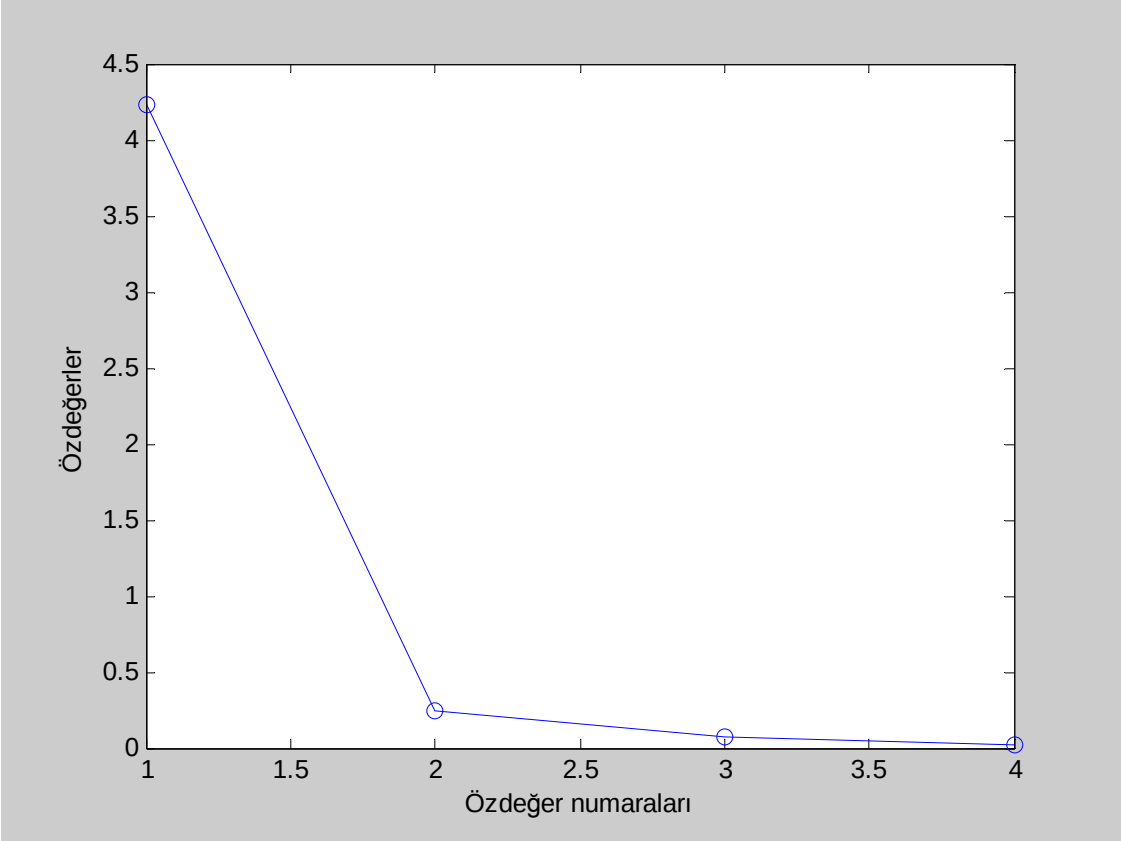
Temel bileşenler birim uzunlukta ve birbirilerine diktirler.

```
I = c4'*c4
I =
    1.0000   -0.0000   -0.0000   -0.0000
   -0.0000    1.0000   -0.0000   -0.0000
   -0.0000   -0.0000    1.0000   -0.0000
   -0.0000   -0.0000   -0.0000    1.0000
```

Kaç adet temel bileşenle çalışılacağına karar verebilmek için özdeğerlerin varyans açıklama oranlarına ve özdeğerlerin yamaç eğim grafiğine bakılabilir.

```
load fisheriris
[coefs,scores,variances,t2] = princomp(meas);
t=0;
for i=1:4
t = t + 100*variances(i)/sum(variances);
toplam_aciklama(i)=t;
end
plot(variances,'bo-')
xlabel('Özdeğer numaraları');
ylabel('Özdeğerler');
toplam_aciklama


toplam_aciklama =                          % 4 özdeğerin açıklama oranı
   92.4619   97.7685   99.4788  100.0000
```

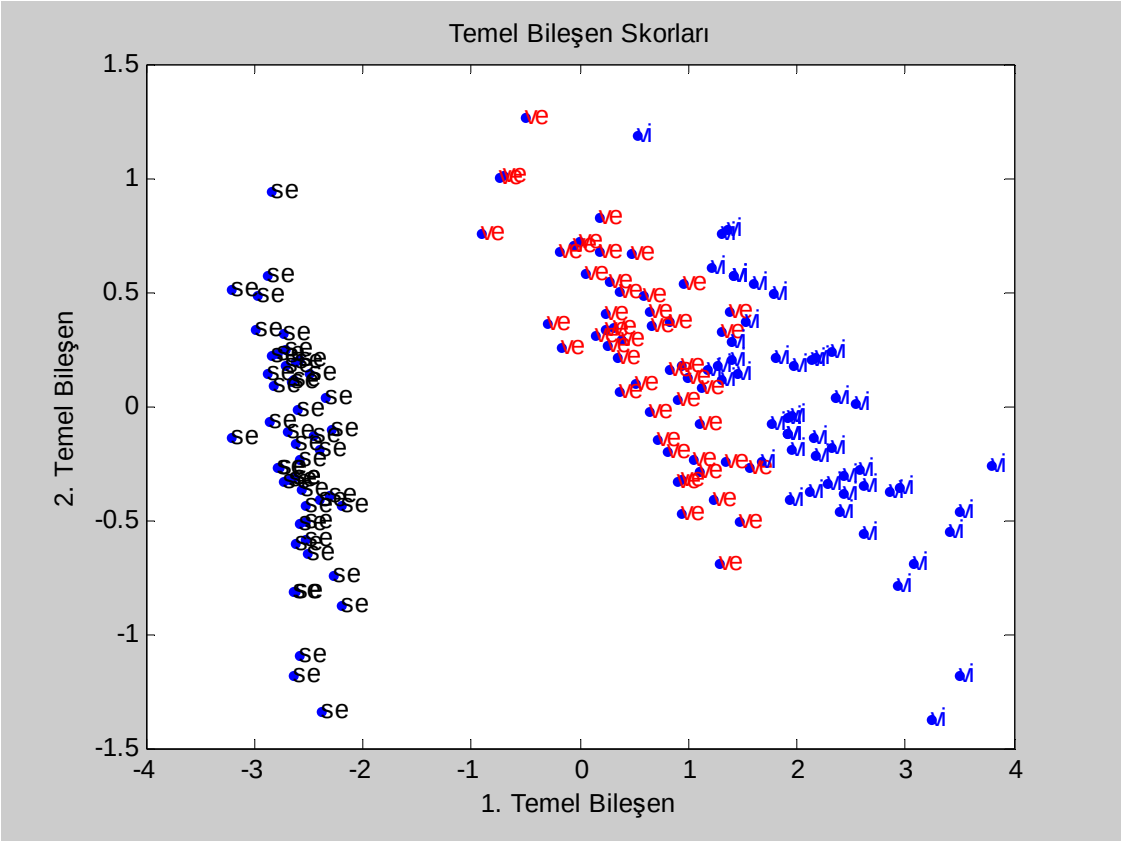
Özdeğerlerin varyans açıklama oranlarına bakıldığında ilk iki özdeğerin toplam varyansın % 97.77' sini açıkladığı gözlenmiştir. Ayrıca Şekil 3.26' deki özdeğerlerin yamaç eğim grafiğine bakıldığında da, ikinci özdeğerden sonra grafiğin eğiminde anlamlı bir azalış gözlenmemektedir. Dolayısıyla süsen veri setini iki temel bileşenle temsil edebiliriz.



Şekil 3.26 Süsen veri seti için yamaç grafiği

Temel bileşenlerin doğrusal birleşimi şeklinde yazılan orijinal değişkenlerden, elde edilen değerler, score değerleri olarak adlandırılırlar. Score değerleri, temel bileşenlerle tanımlanan orijinal verilerin yeni koordinat sistemleridir. Princomp fonksiyonun verdiği score değeri giriş değerimiz olan X veri matrisiyle aynı boyuta sahip bir matristir. Ayrıca bu score değerlerin ortalamaları sıfırdır.

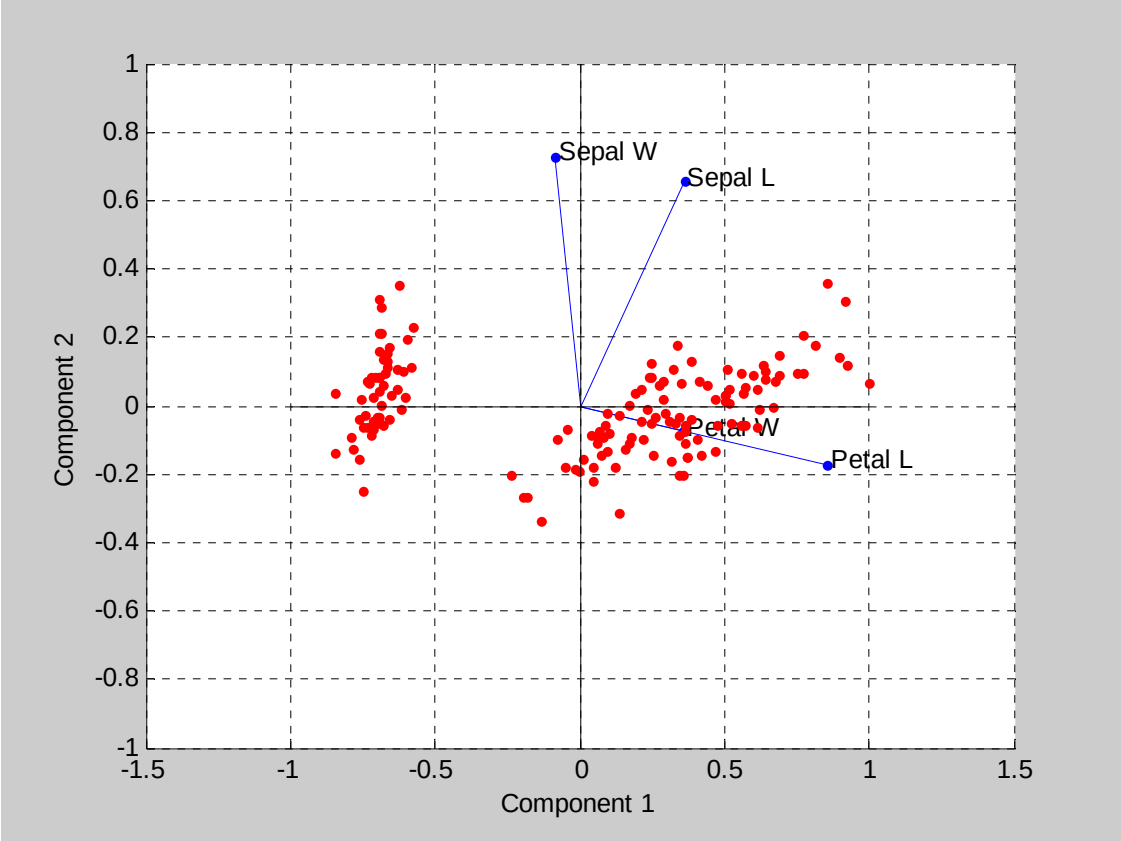
```
for i=1:150
    if i<=50
        spec{i}='se';
    elseif i>50 & i<=100
            spec{i}='ve';
        else
            spec{i}='vi';
    end
end

plot(scores(:,1),scores(:,2),'.');
text(scores(1:50,1),scores(1:50,2),spec(1:50),'color','black');
text(scores(51:100,1),scores(51:100,2),spec(51:100),'color','red');
text(scores(101:150,1),scores(101:150,2),spec(101:150),'color','blue');
xlabel('1. Temel Bileşen')
ylabel('2. Temel Bileşen')
title('Temel Bileşen Skorları')
```



Şekil 3.27 Süsen veri seti için temel bileşen skorları grafiği

Her değişken için bulunan temel bileşenleri ve her gözlem için bulunan temel bileşen skorlarını aynı grafikte iki boyutlu olarak gösterebiliriz. Şekil 3.27' de süsen veri setinden elde edilen temel bileşen skorlarına göre çizilen serpilme grafiği bulunmaktadır. Şekil 3.27' de setosa, versicolor ve virginica süsen bitki cinsleri farklı renklerle gösterilmiştir. Şekil 3.27' den gözüktüğü üzere bu üç cins farklı özelikler sergilemektedirler. Özelikle setosa bitki cinsi diğer bitki cinslerine göre oldukça farklı bir yapı göstermektedir. Versicolor ve virginica bitki cinsleri ise bir birilerine benzemektedirler.

```
kategori=['Sepal L';'Sepal W';'Petal L';'Petal W'];
biplot(coefs(:,1:2), 'scores',scores(:,1:2),'varlabels',kategori);
axis([-1.5 1.5 -1 1]);
```



Şekil 3.28 Süsen veri seti için temel bileşen skor ve katsayıları (Biplot grafiği)

Şekil 3.28' de süsen veri setinde bulunan 4 değişken vektörlerle temsil edilmiştir. Burada vektörlerin yönleri ve uzunlukları 4 değişkenin temel bileşenlere olan katkılarını göstermektedir. Örneğin, Şekil 3.28' de x ekseninde bulunan 1. temel bileşende "Sepal L", "Petal L" ve "Petal W" pozitif, "Sepal W" negatif etkiye sahiptir.

Her değişken için bulunan temel bileşenleri ve her gözlem için bulunan temel bileşen skorlarını aynı grafikte üç boyutlu olarak da görselleştirebiliriz. Orijinal verimizin varyans yapısını iki boyutlu grafikte yeterince açıklayamazsak üç boyutlu grafiği çizebiliriz. Ancak grafiksel gösterimlerde yorumlanmanın kolay olmasından dolayı iki boyutlu gösterim tercih edilmektedir.

3.3.3.3 Simgesel Gösterimler (Iconic Displays)

Simgesel gösterim yöntemlerindeki düşünce çok boyutlu verilerin özelliklerinin simgesel özelliklere haritalanmasından ibarettir. Simgenin her bir görsel özelliği verinin içerdiği değerlere göre değişir. Simgesel gösterimlerin tipik örnekleri Chernoff yüzleri ve yıldız grafikleridir.

3.3.3.3.1 Chernoff Yüzleri (Chernoff Faces)

1973 yılında Herman Chernoff, çok boyutlu verilerde örüntüleri, kümeleri, korelasyonları, trendleri saptamak ve göstermek için the Journal of the American Statistical Association dergisinde "The Use of Faces to Represent Points in k-Dimensional Space Graphically," adlı makalede ilk defa Chernoff yüzlerini görselleştirme yöntemi olarak ortaya atmıştır. Herman Chernoff çok değişkenli karışık verileri insanların kolayca algılayabileceği çizgi yüzlere (cartoon face) benzetmeye çalışmıştır. Bu sayede, insanların kolayca yüz farklılıklarını algılaması özelliğinden yararlanılarak veri yapıları hakkında görsel bilgi elde edilmesi çalışılmıştır (Spinelli ve Zhou).

Herman Chernoff' ın ortaya attığı yöntemde 18 ve daha az değişkenli verilerde her bir gözlem değeri bir çizgi yüzde temsil edilirler. Chernoff yüzlerinde burun uzunluğu (length of face), ağız eğriliği (mouth curvature), kaş biçimi (eyebrow shape), göz büyüklüğü (size of eyes) gibi yüzün 18 özelliği, değişkenlerin aldığı değerlerle orantılı olacak şekilde çizgi yüzlere çizilirler. Bu teknik verilerdeki uç değerlerin keşfinde ve veri biçimlerinin anlaşılmasında kullanışlıdır; ancak bu tekniğin bazı dezavantajları da bulunmaktadır. Chernoff yüzlerinin en büyük dezavantajı değişkenlerin kantitatif görselleştirmelerine olan yoksunluğudur. Çünkü biz Chernoff yüzlerinde değişkenleri, örüntüleri, kümeleri, korelasyonları ve trendleri anlamak için kalitatifsel bir inceleme yaparız (Martinez ve Martinez, 2005). Chernoff yüzlerinin diğer bir dezavantajı ise insan yüzündeki bazı organların diğerlerine göre daha fazla dikkat çekmesidir. Örneğin gözler kulaklardan daha dikkatli algılandığı için karşılaştırma yanılgıları oluşabilir (Bilgin ve Çamurcu, 2007).

3.3.3.3.2 Yıldız Grafikleri (Star Plots)

Yıldız grafikleri 1979 yılında Fienberg tarafından çok boyutlu verileri görselleştirmek için bir yöntem olarak geliştirilmiştir. Kullanım amacı Chernoff yüzlerinde olduğu gibidir (Martinez ve Martinez, 2005).

Yıldız grafikleri, paralel koordinatlara benzer yaklaşım kullanırlar; ancak burada eksenler merkez noktadan saçılma yaparlar. Gözlemlerin değerleri çizgiler ile bağlanarak bir poligon oluşturulur. Başka bir deyişle, her bir değişken bir merkez noktasından itibaren eşit açılı çizgilerle gösterilirler. Daha sonra çizgilerde bulunan değişkenlerin değerleri çizgilerle birbirilerine bağlanırlar ve bu şekilde yıldız grafikleri çizilir (Martinez ve Martinez, 2005).

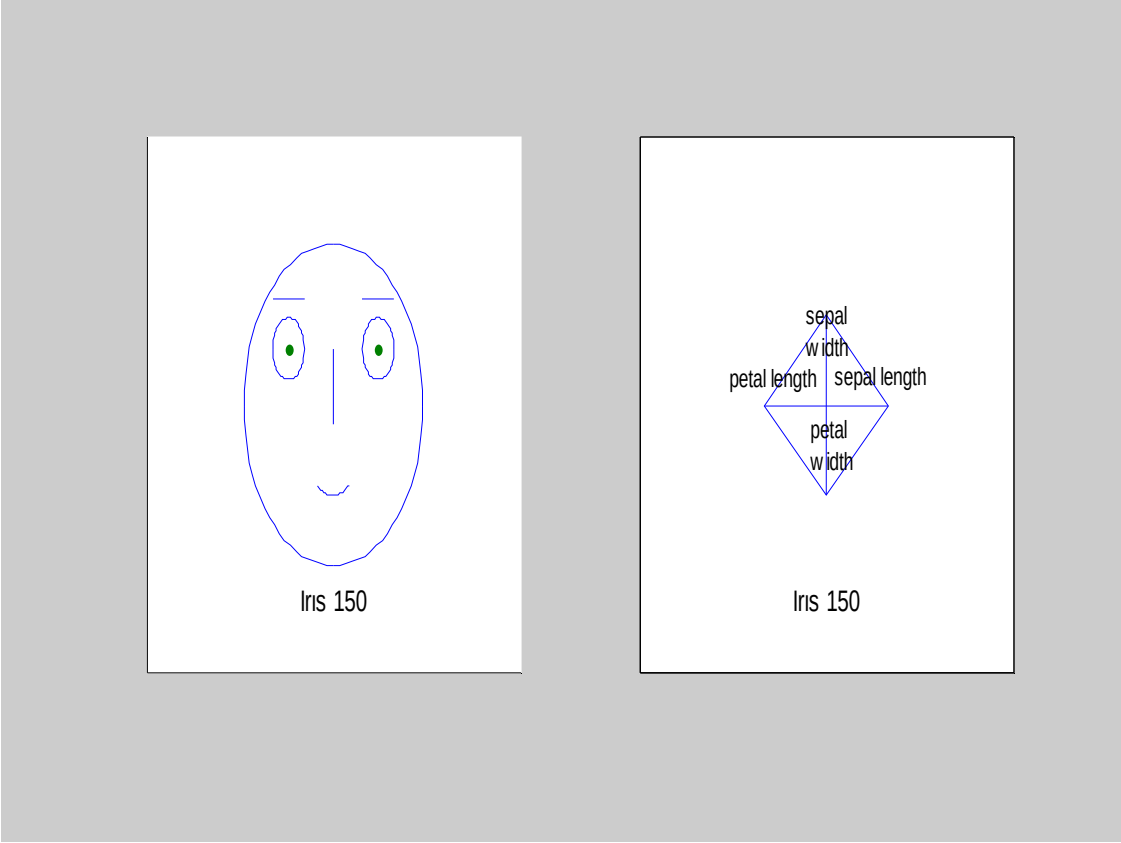
Chernoff yüzlerinde olduğu gibi verideki her gözlem için bir yıldız grafiği çizilir. Yıldız grafikleri verileri incelemede güzel bir yöntem olmasına rağmen sırasıyla 10 ve 15 den daha

fazla gözlemli ve boyutlu verilerde kullanışlı değillerdir. Chernoff yüzlerinde olan dezavantajlar burada da geçerlidir (Martinez ve Martinez, 2005).

Örnek 3.12 :

Süsen veri setinin 150. çiçeği için Chernoff ve yıldız grafiğini çizelim.

```
load fisheriris
subplot(1,2,1)
glyphplot(meas(150,:),'Glyph','face','Features',F,'ObsLabels','Irıs150')
subplot(1,2,2)
glyphplot(meas(150,:),'ObsLabels','Irıs 150')
```



Şekil 3.29 Süsen veri setinin 150. gözlem değeri için Chernoff ve yıldız grafiği

Şekil 3.29' un sol tarafında bulunan yıldız grafiğinde, süsen bitkisinin 4 özelliği, temsil etikleri çizgilerin üstlerine özeliklerin adları gelecek şekilde yer almaktadır. Şekil 3.29' un sağ tarafında bulunan Chernoff yüzünde, çanakyapraklarının uzunluğu (Sepal length) yüz hacmine, çanakyapraklarının genişliği (Sepal width) alın, çene bağlantılı kavis uzunluğuna, taçyaprağı uzunluğu (Petal length) alın biçimine ve taçyaprağı genişliği (Petal width) çene biçimine karşılık gelmektedir.

Örnek 3.13 :

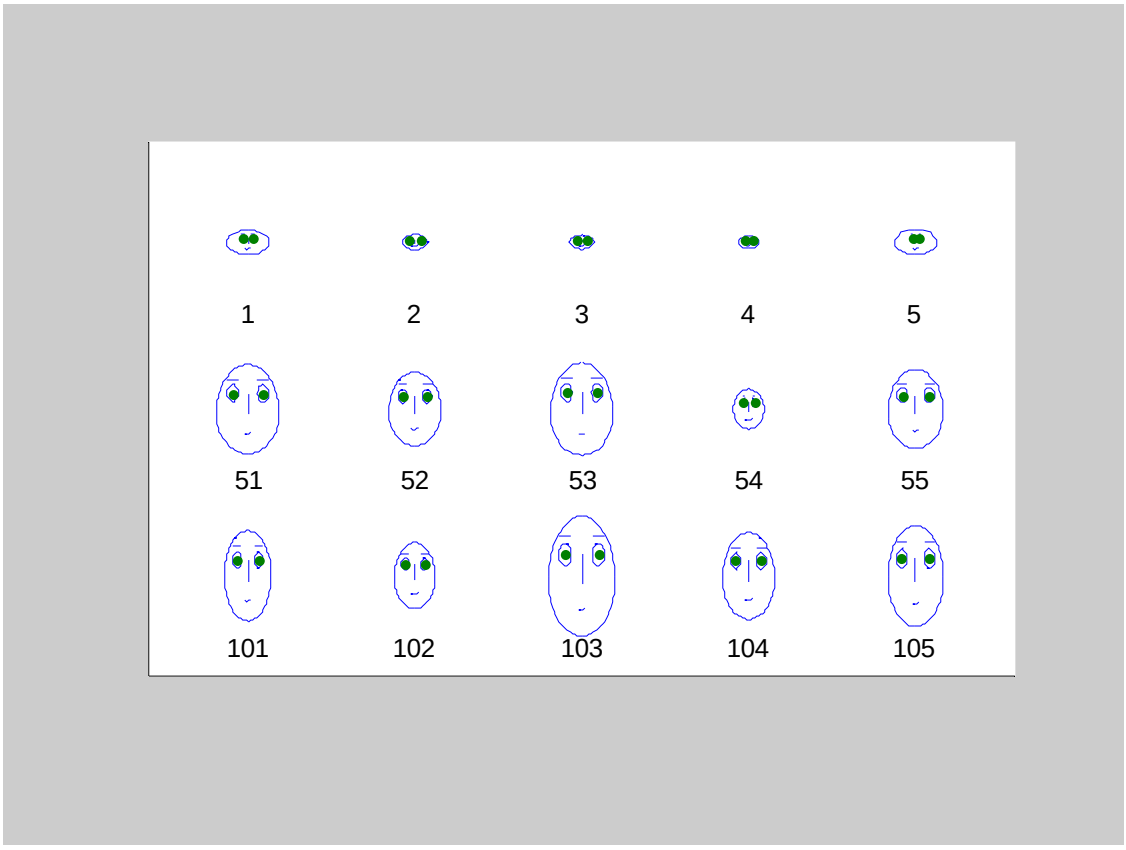
Süsen bitkisinin 150 çiçeğinin Chernnoff yüzlerini ve yıldız grafiklerini çizelim. Bu 150 süsen çiçeğin ilk 50'i setosa, ikinci 50'i vericolor ve son 50' i virginica cinslerine aittir.

Chernoff yüzlerini çizerken çizgi yüzlerde bulunan yüz özelikleri Tablo 3.1 de yer alan süsen bitkisinin değişkenlerine karşılık gelmektedir. Yani çanakyapraklarının uzunluğu yüz hacmine, çanakyapraklarının genişliği alına, çene bağlantılı kavis uzunluğuna, taçyaprağı uzunluğu alın biçimine ve taçyaprağı genişliği çene biçimine karşılık gelmektedir.

Tablo 3.1 Chernoff yüzlerinde değişkenlerin bağlı oldukları yüz özellikleri

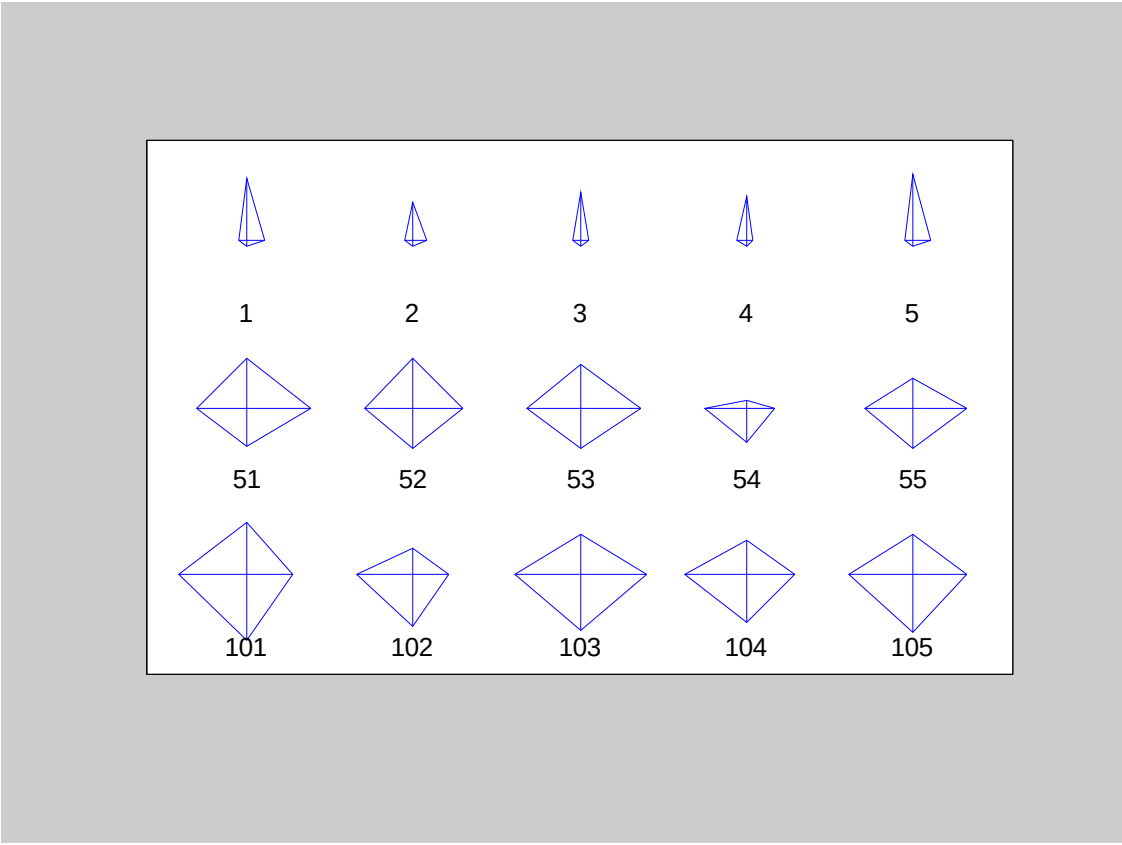
Veri Özelikleri	**Yüz Özellikleri**
Çanakyaprak uzunluğu (Sepal length)	Yüz Hacmi
Çanakyaprak genişliği (Sepal width)	Alın, çene bağlantılı kavis uzunluğu
Taçyaprağı uzunluğu (Petal length)	Alın biçimi
Taçyaprağı genişliği (Petal width)	Çene biçimi

```
label={'1','2','3','4','5','51','52','53','54','55',...
'101','102','103','104','105'};
glyphplot([meas(1:5,:);meas(51:55,:);meas(101:105,:)],'Glyph','face',...
'Grid',[3,5],'ObsLabels',label)
```



Şekil 3.30 Süsen çiçeğinin 15 Chernoff yüzü

```
label={'1','2','3','4','5','51','52','53','54','55',...
'101','102','103','104','105'};
glyphplot([meas(1:5,:);meas(51:55,:);meas(101:105,:)],...
'Grid',[3,5],'ObsLabels',label)
```



Şekil 3.31 Süsen çiçeğinin 15 yıldız grafiği

3.3.3.4 Yoğun Piksel Gösterimler (Dense Pixel Displays)

Bu gösterimin mantığı basitçe; her boyuttaki değerlerin farklı renkli piksellere haritalanması ve her boyuta ait piksellerin komşulukları da göz önünde bulundurularak yeni gruplara atanmasıdır. Bu teknik çok boyutlu büyük veri setlerinin görselleştirilmesi için elverişlidir. En bilinen piksel tabanlı teknik matris grafiğidir.

3.3.3.4.1 Matris Grafikleri (Matrix Plots)

Matris grafiği değişkenler arasındaki ikili ilişkileri kullanıcıya göstermeye yarayan bir çeşit saçılma çizgisidir. Matris grafiğinin ana fikri, veri matrisinde bulunan verilerin büyüklüğünü matris grafiğinde renkli karelerle temsil etmektir. n satır p sütundan oluşan veri matrisi için matris grafiği nxp şeklinde renkli karelerden oluşur. Veri matrisinde bulunan değerler matris grafiğinde büyüklüklerine göre renklendirilirler. Hangi değerlerin hangi renkleri aldığını görebilmek için matris grafiğinde bulunan colorbar' a bakabilirsiniz.

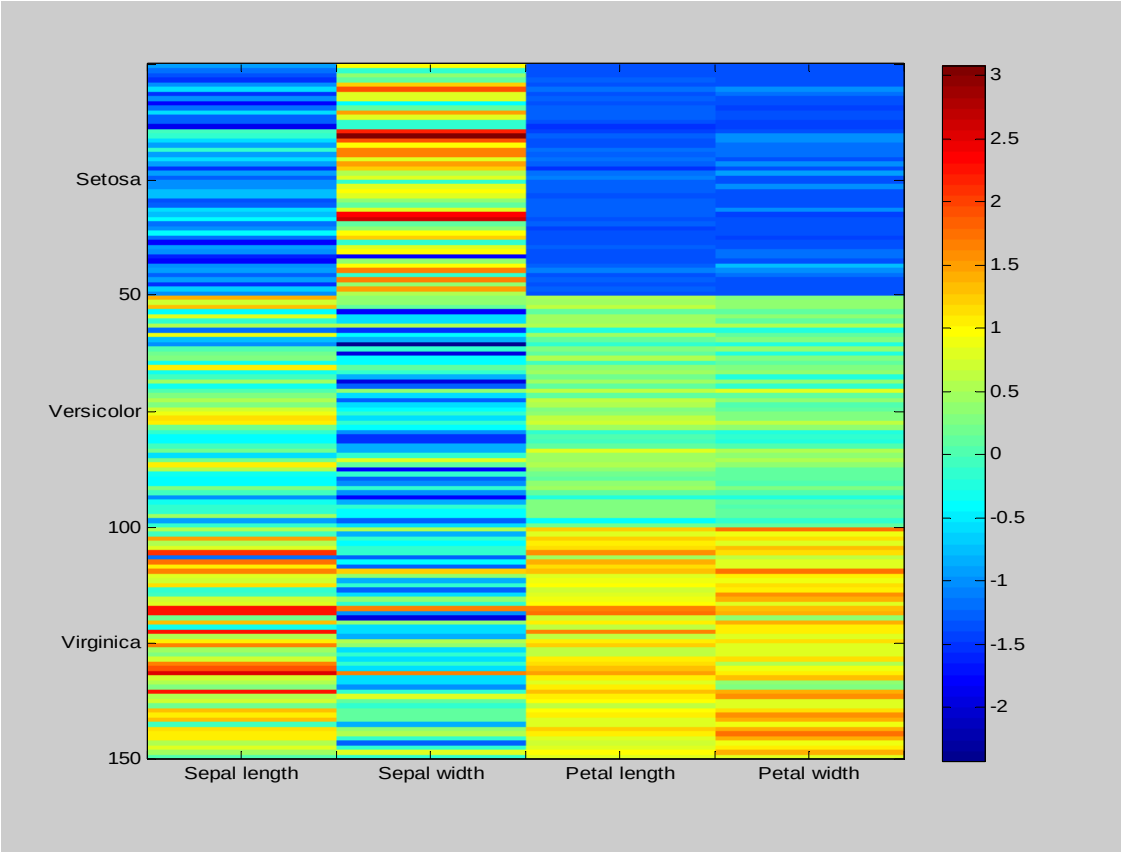
Matris grafikleri veri matrisini görsel olarak göstermede oldukça başarılıdırlar. Ayrıca matris grafikleri birden fazla saçılma çizgisini izleme imkanı da verir. Eğer verilerin bulunduğu sınıflar biliniyorsa, aynı sınıflara sahip olan veriler bir araya gelecek şekilde veri matrisi düzenlenerek elde edilen matris grafiğinde, hangi değişkenler için benzerlikler olduğu hangi değişkenler için farklılıkların olduğu görsel bir şekilde gözlemlenebilir. Benzerlik veya uzaklık matrislerinin matris grafikleri veriler arasındaki ilişkileri görselleştirmek için de faydalıdır.

Genel olarak matris grafikleri kullanımında, değişkenlerin başka değişkenleri bastırmaması için verilerin normalleştirilmeden geçirilmesi oldukça yararlıdır.

Örnek 3.14 :

Süsen veri seti için, değişkenleri 0 ortalama ve 1 standart sapma olacak şekilde, standartlaştırarak matris grafiğini çizelim.

```
clear
clc
load fisheriris
[n,p]=size(meas);
sig=std(meas);
mu=mean(meas);
for i=1:p
    y(:,i)=(meas(:,i)-mu(:,i))/sig(:,i);
end
imagesc(y)
```



Şekil 3.32 Süsen bitkisinin standart veri matrisinin görselleştirmesi

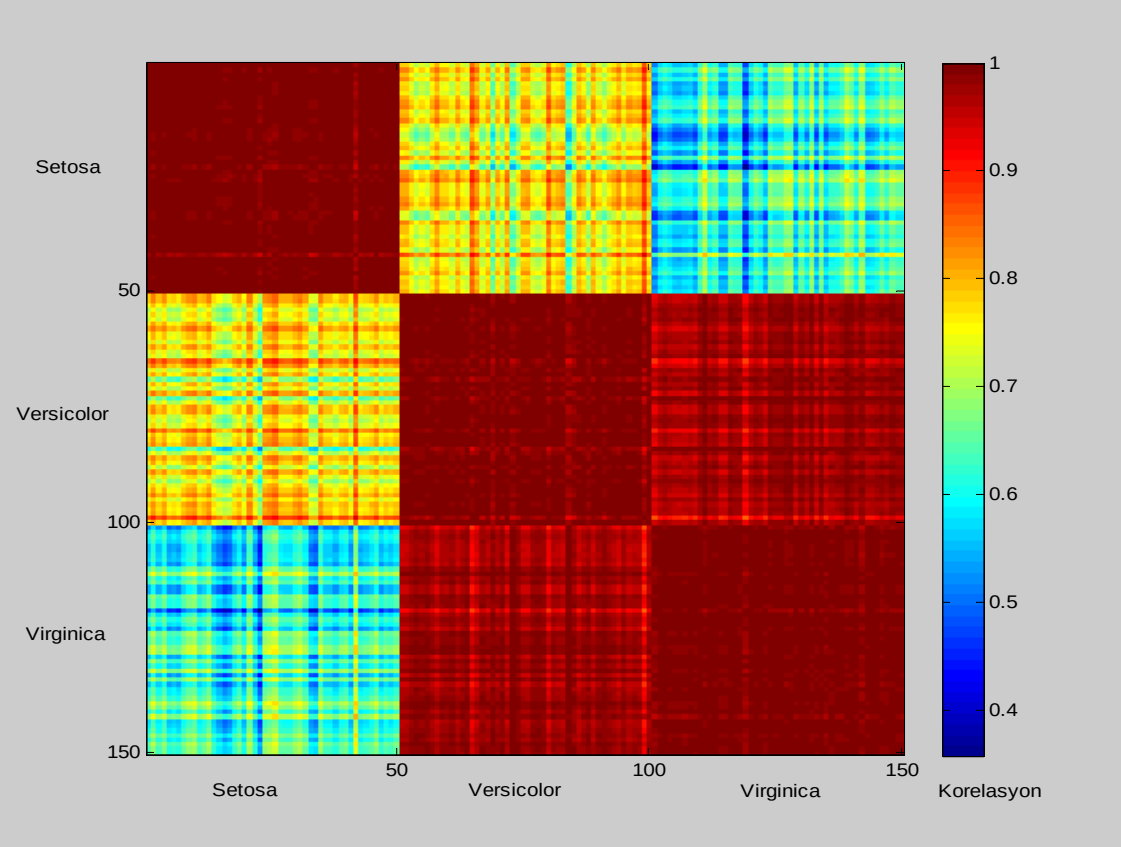
Şekil 3.32' de süsen bitkisi için bitki özelliklerinin standartlaştırılarak çizilen matris grafiği bulunmaktadır. Şekil 3.32' de ilk 50 gözlem setosa, ikinci 50 gözlem versicolor ve son 50 gözlem virginica bitki cinslerine aittir. Şekil 3.32' den setosa bitki cinsinin petal length ve petal width özelliklerinin ortalamanın altında, versicolor bitki cinsinin petal length ve petal width özelliklerinin ortalama etrafında ve virginica petal length ve petal width özelliklerinin ortalamanın üstünde sayısal değerlere sahip olduğu söylenebilir.

Benzerlik veya uzaklık matrislerinin matris grafikleri veriler arasındaki ilişkileri görselleştirmek için de faydalıdır.

Örnek 3.15 :

Süsen veri seti için korelasyon matrisini kullanarak matris grafiği çizelim.

```
load fisheriris
yc=corr(meas');
imagesc(yc);
```



Şekil 3.33 Süsen bitkisinin korelasyon matrisinin görselleştirmesi

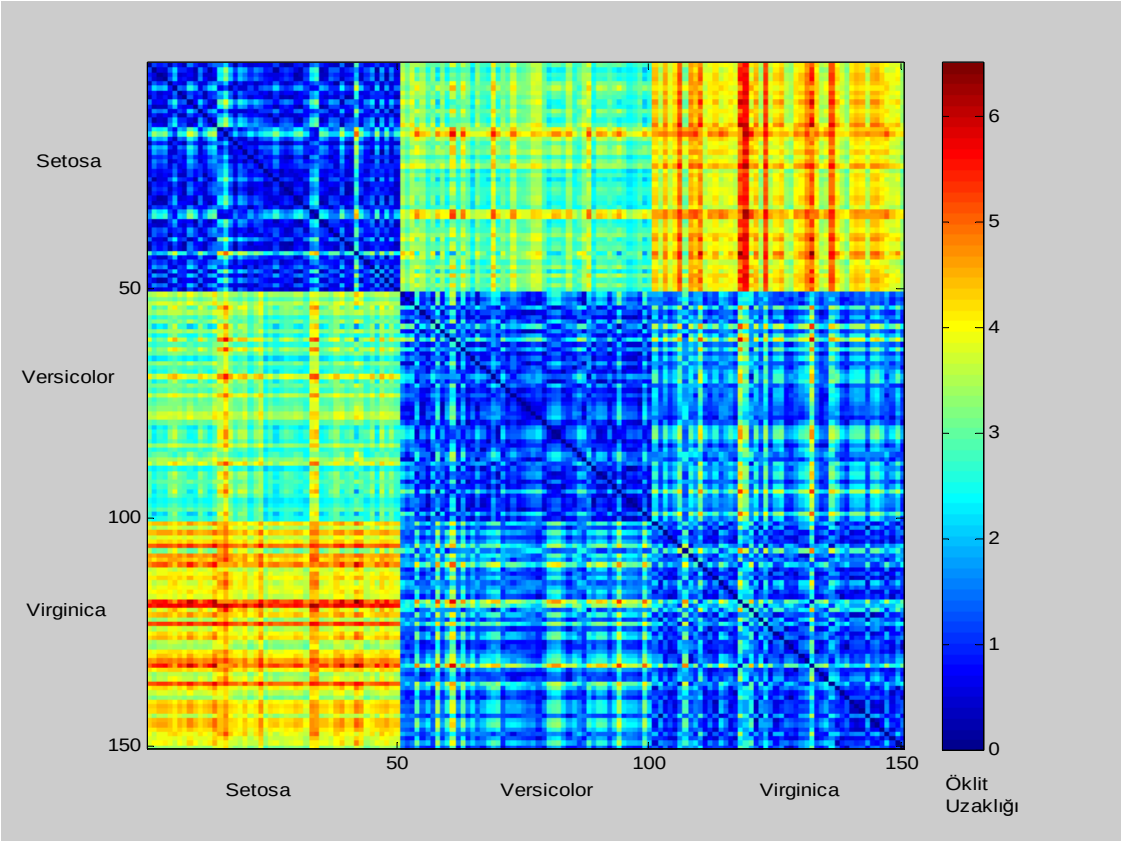
Şekil 3.33 de süsen bitkisi için bitki özelliklerinin korelasyon matrisleri kullanılarak çizilen matris grafiği bulunmaktadır. Şekil 3.33' de ilk 50 gözlem setosa, ikinci 50 gözlem versicolor ve son 50 gözlem virginica bitki cinslerine aittir. Şekil 3.33' de setosa, versicolor ve virginica bitki cinslerinin birbirilerine korele oldukları gözlemlenmiştir; fakat versicolor ve virginica bitki cinsleri birbirilerine setosa bitki cinsine göre daha çok korelidir.

Küme yapılarını görmek adına da matris grafikleri de kullanılabilir. Bu sayede kendi içinde homojen kendi aralarında heterojen guruplar görsel bir şekilde gözlemlenebilir. Bu özelliğinden dolayı matris grafikleri kümeleme analizlerinden elde edilen sonuçların doğruluğunu göstermede de kullanılmaktadırlar. Matris grafikleri çok büyük veri setlerinde kullanılsa da veri setindeki küme sayısı hakkında genel bir fikir edinmeyi ve buna uygun iyileştirilmelerin yapılmasına olanak tanır.

Örnek 3.16 :

Süsen veri seti için öklid uzaklık matrisini kullanarak matris grafiği çizelim.

```
load fisheriris
ys=squareform(pdist(meas));
imagesc(ys)
```

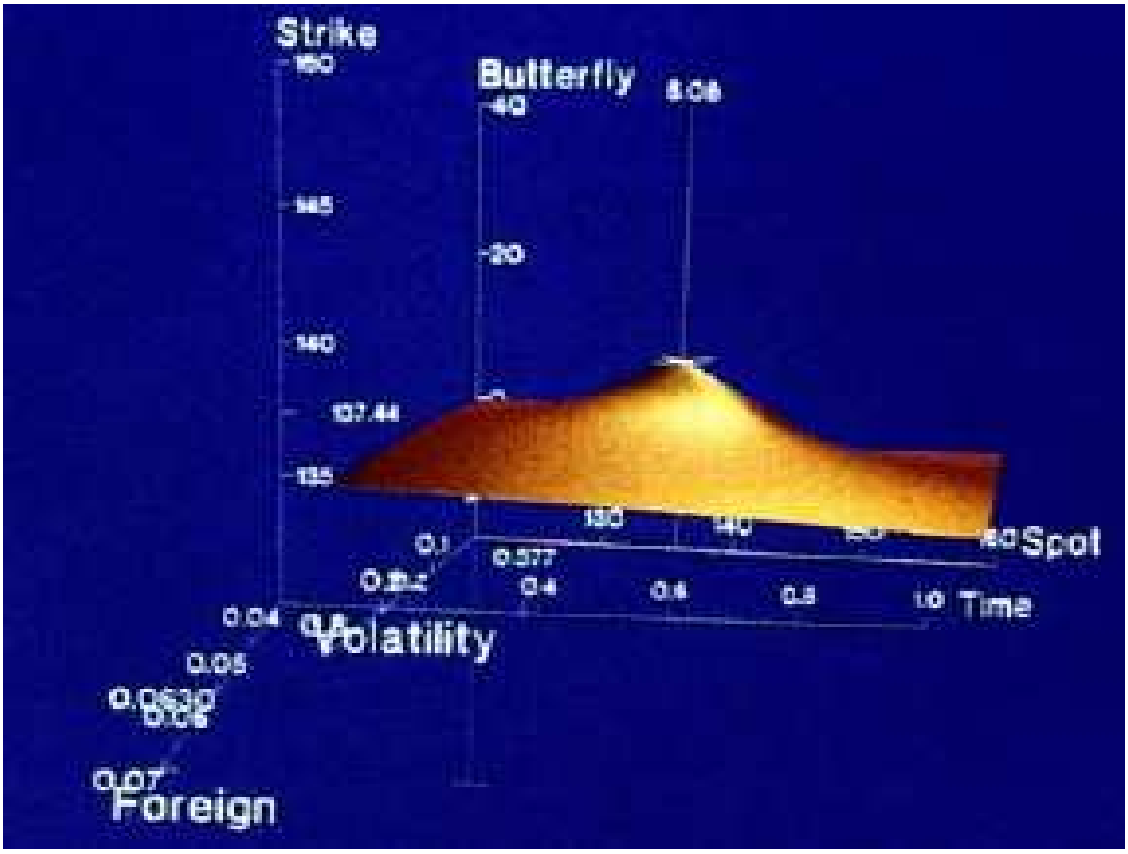


Şekil 3.34 Süsen bitkisi uzaklık matrisinin görselleştirilmesi

Şekil 3.34 de süsen bitkisi için öklid uzaklık matrisleri kullanılarak çizilen matris grafiği bulunmaktadır. Şekil 3.34’ de ilk 50 gözlem setosa, ikinci 50 gözlem versicolor ve son 50 gözlem virginica bitki cinslerine aittir. Şekil 3.34 de setosa, versicolor ve virginica bitki cinslerinin üç küme oluşturduğu ancak versicolor ve virginica bitki cinslerinin setosaya göre birbirilerine daha benzer yapıda olduğu gözlemlenir.

3.3.3.5 İstiflenmiş gösterimler (Stacked Displays)

Bu gösterimde en sık kullanılan yöntem boyutsal istiflemedir. Bu yöntemin mantığı basitçe bir koordinat sisteminin diğerinin içine gömülmesi olarak tanımlanabilir. Bir boyuta ait iki özelliğin daha dışta olan bir boyuta taşınması gibi. Bu türün en bilinen örneği N-Vision veya diğer adı ile “dünya içinde dünyalar” (Worlds-within-Worlds) adlı sistemdir.



Şekil 3.35 6 boyutlu uzayın N-Vision ile görüntülenmesi (Bilgin ve Çamurcu, 2008)

N-Vision aracı k-boyutlu uzayı birçok üç boyutlu alt uzaya ayırarak görselleştirir. Şekil 3.35' de altı boyutlu uzayın görüntülenmesi görülmektedir. İlk üç boyut dış koordinat sistemi ile sonraki üç boyut ise iç koordinat sistemi ile gösterilmiştir (Çamurcu ve Bilgin, 2007).

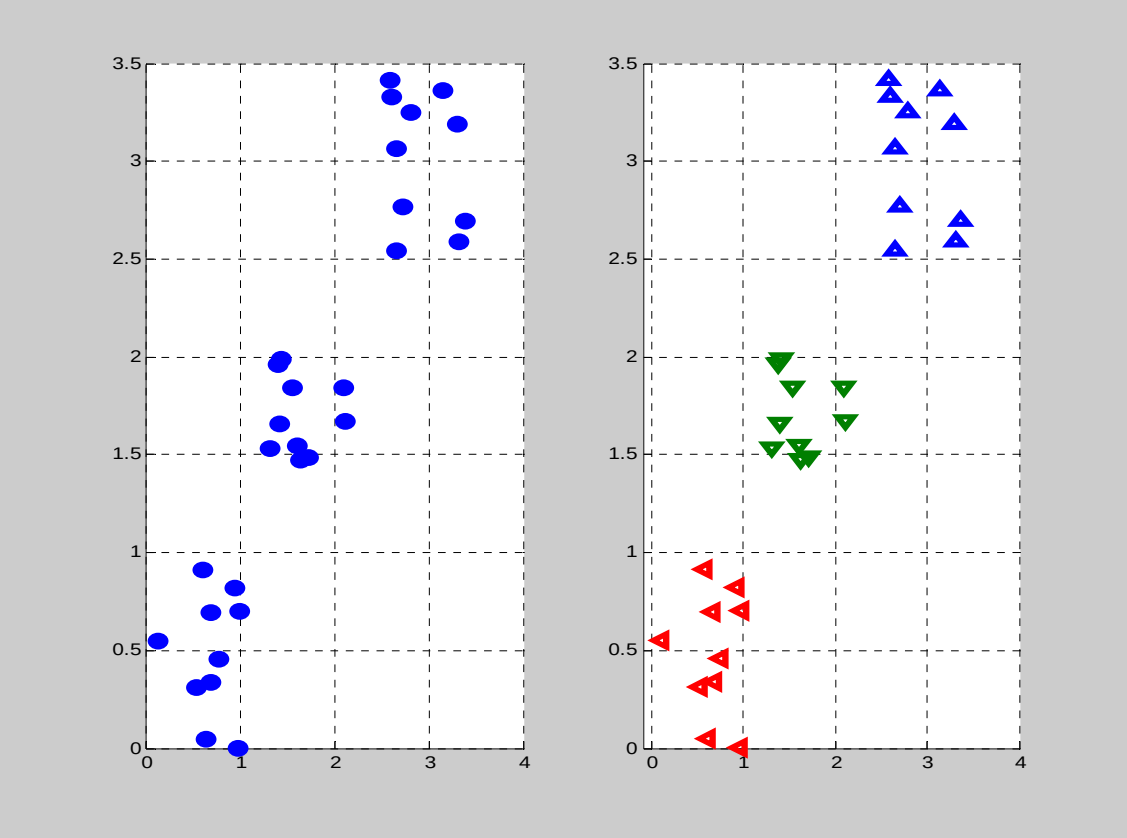
4. ÖZEL GÖRSEL VERİ MADENCİLİĞİ TEKNİKLERİ (SPECIFIC VISUAL DATA MINING TECHNIQUES)

Birliktelik, sınıflandırma ve kümeleme analizleri gibi özel veri madenciliği algoritmalarını desteklemek amacıyla çeşitli görselleştirme teknikleri geliştirilmiştir. Bu tez çalışmasında kümeleme analizleri için geliştirilen görselleştirme tekniklerini incelenecektir. Bunun için önce kümeleme analizi kavramları ve kümeleme algoritmaları tanıtılacaktır.

4.1 Kümeleme Analizi (Cluster Analysis)

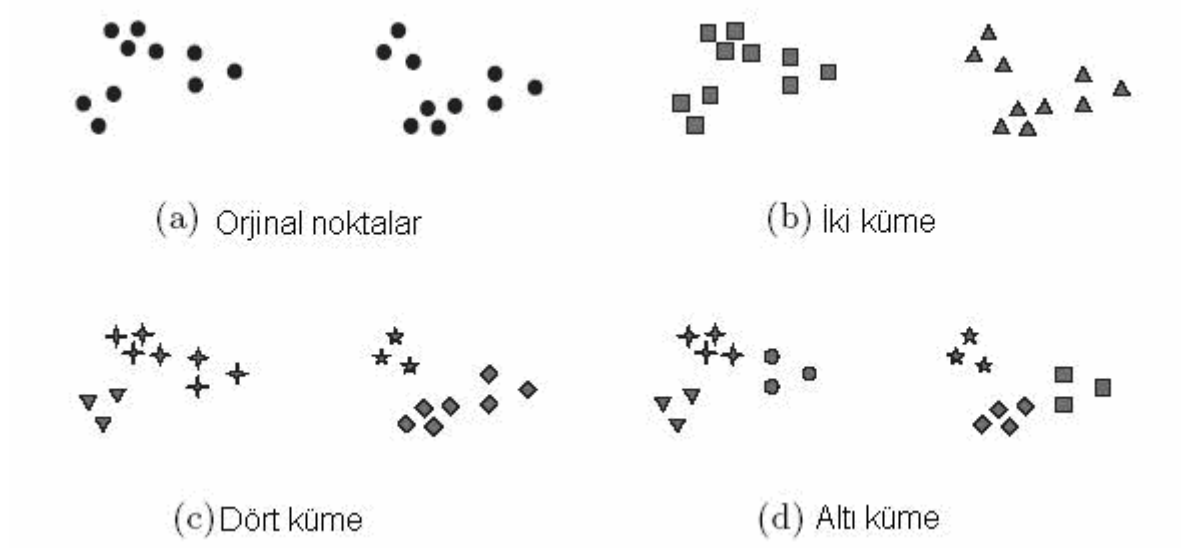
"Kümeleme analizi X veri matrisinde yer alan ve doğal grupları kesin olarak bilinmeyen birimleri, değişkenleri ya da birim ve değişkenleri birbirleri ile benzer olan alt kümelere (grup, sınıf) ayırmaya yardımcı olan yöntemler topluluğudur (Özdamar, 2004)."

Kümeleme analizi, birimleri değişkenler arası benzerlik ya da uzaklıklara dayalı olarak hesaplanan bazı ölçülerden yararlanarak homojen gruplar oluşturmaya çalışır (Özdamar, 2004). Kümeleme analizi sonucunda kümeleri oluşturan elemanlar birbirine benzerlik, başka kümelerin elemanlarından farklılık gösterirler. Kümeleme işlemi başarılı olursa, bir geometrik çizim yapıldığında birimler küme içerisinde birbirilerine çok yakın, kümeler ise birbirilerinden uzak olacaktır. Şekil 4.1' de basit bir kümeleme işlemi gösterilmiştir.



Şekil 4.1 Veri birimleri ve kümeleri

Birçok uygulamada, küme kavramı net bir şekilde tanımlanmamıştır. Bir kümeyi ortaya koyan şeylerin neler olduğuna karar verme güçlüğünü daha iyi anlayabilmek için, Şekil 4.2' yi dikkate alalım. Şekil 4.2' de ki değişik noktalar, veriyi kümelere ayırmanın üç farklı yolunu göstermektedir.



Şekil 4.2 Aynı veri setinin değişik yollarla kümelenmesi (Tan vd., 2006)

Şekil 4.2 (b) ve Şekil 4.2 (d) sırasıyla veriyi iki ve altı parçaya ayırır. Bununla beraber, iki tane büyükçe kümenin her birinin daha küçük üç alt kümeye net olarak bölünmesi yalnızca insanın görme sistemine ilişkin bir yanılma olabilir. Aynı zamanda, şunu söylemek de mantıksız olmaz "noktalar dört tane küme oluşturur", bu da Şekil 4.2 (c)' de görülmektedir. Bu şekil bize bir küme tanımının kesin ve net olmadığını, en iyi kümelemenin de verinin doğasına ve arzu edilen sonuçlara bağlı olduğunu gösterir.

Kümeleme analizinin genel amacı, gruplanmamış verileri benzerliklerine göre sınıflandırmak ve araştırmacıya uygun, işe yarar özetleyici bilgiler elde etmede yardımcı olmaktır. Doğal kümelerin yapılarını bulmak için veriyi araştırmak önemli bir açıklayıcı tekniktir. Kümeleme analizinin aşağıdaki özel amaçlarından da söz edilebilir:

- Gerçek tiplerin (cinslerin-ırkların) belirlenmesi
- Model uydurmanın kolaylaştırılması
- Gruplar için ön tahmin
- Hipotezlerin testi
- Veri yapısının netleştirilmesi
- Veri indirgemesi (veriler yerine kümelerin değerlendirilmesi)
- Aykırı değerlerin bulunması.

Kümeleme analizi sınıflandırma modellerinden farklıdır. Kümeleme analizinde, sınıflama modelinde olan veri sınıfları yoktur. Verilerin herhangi bir sınıfı bulunmamaktadır. Sınıflama modelinde, verilerin sınıfları bilinmekte ve yeni bir veri geldiğinde bu verinin hangi sınıftan olabileceği tahmin edilmektedir. Oysa kümeleme analizinde, sınıfları bulunmayan veriler gruplar halinde kümelere ayrılırlar. Bazı uygulamalarda kümeleme analizi, sınıflama modelinin bir önişlemi gibi görev alabilmektedir (Özekes, 2003).

Kümeleme analizleri güçlü bir gelişim göstermektedir. Veri tabanlarında toplanan veri miktarının artmasıyla orantılı olarak, kümeleme analizi son zamanlarda, özellikle veri madenciliği araştırmalarında genişçe yer bulur hale gelmiştir. Kümeleme analizi ayrıca istatistik, biyoloji, psikoloji, tıp, arkeoloji, sosyoloji ve makine öğrenim gibi daha pek çok alanda kullanım olanağı bulmaktadır.

Kümeleme analizlerinde yaşanan gelişmelerden dolayı çeşitli kümeleme algoritmalarını yapan bilgisayar programları geliştirilmiştir. SAS, SPSS, MATLAB, S-PLUS kümeleme analizlerini yapan en bilinen bilgisayar programlarıdır.

4.2 Değişken Türlerine Göre Benzerlik ve Uzaklık Ölçüleri

Bir veri setinde yer alan birimlerin kümelenmesi, temel bileşenler analizi, çok boyutlu ölçekleme ve kendinden düzenlenen haritalar gibi boyut azaltma işlemlerinin yapılabilmesi, bu birimlerin birbirleriyle olan benzerlikleri (similarity) ya da birbirine olan uzaklıkları (dissimilarity) kullanılarak gerçekleştirilmektedir.

Benzerlik ve uzaklık ölçülerinden kısaca bahsetmek gerekirse, benzerlik ölçüleri, birimlerin birbirilerine olan benzerliklerini göstermekte kullanılan ölçümlerdir. Benzerlik ölçüleri maksimum 1 değerini benzerlik değeri olarak alabilirler. Benzerlik ölçülerinin değerleri arttıkça birimler arasındaki benzerlikler artar, azaldıkça da birimler arasındaki benzerlikler azalır. Uzaklık ölçüleri ise benzerlik ölçülerinin tam tersi bir yaklaşım sergilerler. Uzaklık ölçülerinin küçük olması birimlerin birbirilerine benzer olduğunu gösterir. Uzaklık ölçülerinde 0 maksimum benzerliği ifade eder. Uzaklık ölçülerinin değerleri arttıkça birimler arasındaki benzerlik azalar, azaldıkça da birimler arasındaki benzerlik artar (Martinez ve Matinez, 2005).

Değişkenlerin kesikli ya da sürekli olmalarına ya da değişkenlerinin nominal, ordinal, aralık ya da oransal ölçekte olmalarına göre hangi uzaklık ölçüsünün ya da benzerlik ölçüsünün kullanılacağına karar verilir. Aşağıdaki alt bölümlerde değişken türlerine göre kullanılan uzaklık veya benzerlik ölçülerinden bahsedilecektir. Uzaklık ve benzerlik ölçülerini tanımlarken kullanılan i ve j indisleri 1,2,…,n değerlerini, k indisi 1,2,…,p değerlerini alabilir. Burada n birim sayısı, p değişken sayısıdır.

4.2.1 Aralık ve Oransal Ölçekli Değişkenler

Aralık ölçeği sayısal olarak ifade edilebilen, toplama ve çıkarma gibi matematiksel işlemleri mümkün kılan ölçeklerdir. Aralık ölçekte gerçek anlamda bir sıfır yoktur. Ölçüm farklılıklarının ve düzenlerinin önemli olduğu bir ölçektir. Aralık ölçeğine ısı ölçümlerinde kullanılan Celciusu ölçeğini örnek verebiliriz. Bilindiği gibi suyun donma derecesi Celciusu ölçeğinde 0^{o}'' dir. 0 ^{o}C hiçlik anlamına gelmemektedir. 0 ısının yok olduğunu göstermeyen keyfi bir değerdir. Aralık ölçeklerinin oranları da herhangi bir anlam taşımaz. 4 ^{o}C, 8 ^{o}C iki katı değildir, ancak ondan daha düşük bir ısıyı ifade etmektedir. Oran ölçeği ise, aralık

ölçeğinin özelliğini taşıdığı gibi belli bir sıfır değerine sahip olan ölçeklerdir. Ölçümler arasında düzen ve uzaklık olduğu gibi ölçümler arasındaki oranda önemlidir. Örneğin, 20 metrekarelik bir alan 10 metrekarelik bir alanın 2 katı olduğu gibi. Aralık ve oran ölçekli veriler arasındaki uzaklık ve benzerlikler aşağıdaki gibi hesaplanır (Orhunbilge, 2000).

4.2.1.1 Öklidyen ya da Karesel Öklit Uzaklık Ölçüsü

Öklidyen uzaklık ölçüsü kullanılarak iki birim arasındaki uzaklık

$$d(i,j)=\sqrt{\sum_{k=1}^{p}\left(x_{ik}-x_{jk}\right)^2} \tag{4.1}$$

formülüyle hesaplanır. En çok kullanılan ölçü birimidir. Değişkenler belirli bir önem derecesinde ağırlıklandırılmışsa, Öklidyen uzaklık ölçüsü formülü aşağıdaki gibi olur.

$$d(i,j)=\sqrt{\sum_{k=1}^{p}w_k\left(x_{ik}-x_{jk}\right)^2} \tag{4.2}$$

4.2.1.2 Pearson Uzaklık Ölçüsü

Pearson uzaklık ölçüsü kullanılarak iki birim arasındaki uzaklık

$$d(i,j)=\sqrt{\sum_{k=1}^{p}\left(x_{ik}-x_{jk}\right)^2 \Big/ s_k^2} \tag{4.3}$$

formülü ile hesaplanır. Bu formülde kullanılan s_k, uzaklığın hesaplandığı değişkene ait standart sapmadır. Bununla birlikte farklı gruplar hakkında önceden bilgi sahibi olunmadığı için, uzaklık hesaplanmasında s değerinin kullanılması doğru olmaz. Bu nedenle Pearson uzaklık ölçüsü yerine genellikle Öklidyen uzaklık ölçüsü tercih edilir. Kümeleme analizinde kullanılacak olan değişkenler belirli önem derecelerine göre ağırlıklandırılmışlarsa, Pearson uzaklık ölçüsü formülü aşağıdaki gibi olur.

$$d(i,j)=\sqrt{\sum_{k=1}^{p}w_k\left(x_{ik}-x_{jk}\right)^2 \Big/ s_k^2} \tag{4.4}$$

Pearson uzaklık ölçüsüne, “karesel Pearson uzaklık” ya da “standardize öklid uzaklığı” adı da verilir.

4.2.1.3 Manhattan Uzaklık (Veya City Block) Ölçüsü

Manhattan uzaklık ölçüsü kullanılarak iki birim arasındaki uzaklık

$$d(i,j)=\sum_{k=1}^{p}\left(\left|x_{ik}-x_{jk}\right|\right) \tag{4.5}$$

formülü ile hesaplanır. Bu ölçü de birimler arasındaki mutlak uzaklık kullanılır. Değişkenler eğer belirli bir önem derecesinde ağırlıklandırılmışlarsa, Manhattan uzaklık ölçüsü formülü aşağıdaki gibi olur.

$$d(i,j)=\sum_{k=1}^{p}\left(w_k\left|x_{ik}-x_{jk}\right|\right) \tag{4.6}$$

Manhattan uzaklık ölçüsüne, “city block uzaklık ölçüsü” adı da verilir.

4.2.1.4 Minkowski Uzaklık Ölçüsü

Minkowski uzaklık ölçüsü genel bir formüldür. Formülde yer alan λ değerinin alacağı farklı değerlere göre yeni formüller türetilir. Minkowski uzaklık ölçüsü kullanılarak iki birim arasındaki uzaklık

$$d(i,j)=\left\{\sum_{k}^{p}\left|x_{ik}-x_{jk}\right|^{\lambda}\right\}^{\frac{1}{\lambda}} \qquad \lambda\geq 1 \tag{4.7}$$

formülü ile hesaplanır. Değişkenler belirli önem derecelerine göre ağırlıklandırılmışlarsa, Minkowski uzaklık ölçüsü formülü aşağıdaki gibidir.

$$d(i,j)=\left\{\sum_{k}^{p}w_k\left|x_{ik}-x_{jk}\right|^{\lambda}\right\}^{\frac{1}{\lambda}} \tag{4.8}$$

Minkowski uzaklık ölçüsündeki λ değeri büyük ve küçük farklara verilen ağırlığı değiştirir. Minkowski uzaklık ölçüsü λ = 1 için Manhattan uzaklık ölçüsüne, λ = 2 için Öklidyen uzaklık ölçüsüne dönüşür.

4.2.1.5 Mahalanobis Uzaklık Ölçüsü

Mahalanobis uzaklık ölçüsü kullanılarak birimler arasındaki uzaklık

$$d^2(i,j)=\left(x_i-x_j\right)^T\Sigma^{-1}\left(x_i-x_j\right) \tag{4.9}$$

formülü ile hesaplanır. Burada Σ kovaryans matrisidir. Çoğunlukla mevcut veriler kullanılarak Σ kovaryans matrisi tahmin edilir. Burada bulunan Σ kovaryans matrisi aşırı değerlere karşı duyarlıdır.

4.2.1.6 Açısal Benzerlik Ölçüsü (Cosine Similarity Measure)

Açısal benzerlik ölçüsü iki veri noktasının özellik vektörleri arasındaki açının kosinüsüdür. [+1,-1] arasında değerler alır. İki vektör arasındaki açısal benzerlik ölçüsü aşağıdaki gibi belirlenir.

$$s_{ij} = \frac{x_i^T x_j}{\sqrt{x_i^T x_i}\sqrt{x_j^T x_j}} \tag{4.10}$$

4.2.1.7 Korelasyon Benzerlik Ölçüsü (Correlation Similarity Measure)

İki özellik vektörü arasındaki korelasyon değeri bu iki vektörün birbirine olan benzerlik derecesini gösterir. [+1,-1] aralığında değerler alır. Herhangi iki vektör arasındaki korelasyon benzerlik ölçüsü aşağıdaki gibi belirlenir.

$$s_{ij} = \frac{\left(x_i - \bar{x}\right)^T \left(x_j - \bar{x}\right)}{\sqrt{\left(x_i - \bar{x}\right)^T \left(x_i - \bar{x}\right)}\sqrt{\left(x_j - \bar{x}\right)^T \left(x_j - \bar{x}\right)}} \tag{4.11}$$

4.2.1.8 Uzaklık Fonksiyonun Özellikleri

- $d(i, j) \geq 0$; Uzaklık negatif değil
- $d(i,i) = 0$; Her birim kendisine olan uzaklığı sıfırlar.
- $d(i, j) = d(j,i)$; Uzaklık fonksiyonu simetriktir.
- $d(i, j) \leq d(i,h) + d(h, j)$; İki birimin arasındaki uzaklık bu iki birimin üçüncü bir birime olan uzaklıkları toplamından büyük olamaz (üçgen eşitsizliği)

4.2.2 Nominal Ölçekli Değişkenler

Bu ölçekte kullanılan rakamlar veya isimler birimleri sınıflara veya kategorilere ayırmaktadır. Örneğin nüfusu cinsiyet özelliğine göre sınıflandırırken kadın ve erkek gibi şıklar kullanılabileceği gibi kadınlar için 1, erkekler için 0 rakamları birimlerin sınıflandırılması için de kullanılabilir. Bu ölçekte bulunan verilerin toplamları hiçbir anlam taşımaz. Nominal

ölçekli değişkenler ikili (binary), ikili olmayan ölçekli değişkenler olarak ikiye ayrılırlar (Orhunbilge, 2000).

4.2.2.1 İkili Nominal Değişkenler

Erkek - Kadın, Evet – Hayır, Olumlu – Olumsuz gibi iki seçenek alabilen değişkenler nominal ölçekli değişkenlerdir. İkili nominal değişkenlerde, aralık ve oransal ölçekli verilerde kullanılan Pearson, Öklidyen, Manhatten (City block) Minkowski gibi birimler arası uzaklıklarının kullanılması uygun değildir.

İkili nominal ölçekli değişkenlerde dört gözlü kontenjans tablolarından yararlanarak benzerlik ölçüleri elde edilebilir. Dört gözlü kontenjans tablosundan elde edilen 1-1, 0-0, 0-1, 1-0 eşleşmelerin frekansları kullanılarak sözü edilen benzerlik ölçüleri hesaplanır.

Tablo 4.1' de dört gözlü kontenjans tablosu yer almaktadır.

Tablo 4.1 İkili değişkenler için kontenjans tablosu

j. gözlem				
i. gözlem		1	0	Toplam
	1	a	b	a + b
	0	c	d	c + d
	Toplam	a + c	b + d	p = a + b + c + d

Tablo 4.1' de yer alan kontenjans tablosundaki frekans değerleri kullanılarak aşağıdaki Tablo 4.2' de yer alan 6 adet benzerlik ölçüsü elde edilir.

Tablo 4.2 Benzerlik Ölçüleri

Katsayı	**Benzerlik Ölçüleri**
Jaccard Benzerlik Katsayısı	$\frac{a}{a+b+c}$
Ochiai Benzerlik Katsayısı	$\frac{a}{\sqrt{(a+b)(a+c)}}$
Rao Benzerlik Katsayısı	$\frac{a}{a+b+c+d}$

Basit Eşleşme Benzerlik Katsayısı (simple matching coefficient)	$\frac{a+d}{a+b+c+d}$
Binary Öklid uzaklığı	$\sqrt{b+c}$
Binary karesel Öklid uzaklığı	$b+c$

4.2.2.2 İkili Olmayan Nominal Değişkenler

İki seçenekten daha fazla seçeneğe sahip olan değişkenlerin uzaklıklarının hesaplanması ise şu formülle hesaplanır.

$$d(i,j)=\frac{p-m}{p} \tag{4.12}$$

p toplam değişken sayısı, m eşleşen değişken sayısı olmak üzere eşleşmeyen değişken sayısı (p-m) toplam değişken sayısına oranlanarak iki birim arasındaki uzaklık bulunur. Bu uzaklık 1' den çıkarılırsa, iki değişken arasındaki benzerlik katsayısı bulunur.

4.2.3 Ordinal Ölçekli Değişkenler

Bu ölçekte rakamlar, büyüklük, tercih gibi çeşitli özelliklerin sıralanmasında kullanılır. Markaların en çok beğenilenden en az beğenilene doğru 1' den başlayarak sıralanmasında ordinal ölçek kullanılmaktadır. Burada bireylerin markalardan ne kadar memnun olduğu anlaşılamamakta, birinci sıradaki markanın dördüncü sıradakinden 4 katı kadar beğenildiği anlamı çıkartılamamaktadır (Orhunbilge, 2000). Ordinal ölçekli değişkenlere sahip birimlerin uzaklık ölçüsü aşağıdaki adımları takip ederek elde edilir.

- x_f değişkenine ait i. birimin sıralaması M birim içerisinde r_{if} olur.
- Her bir ordinal değişken farklı sayıda durumlara sahip olacağından, her bir değişken aşağıdaki formül ile [0,1] aralığına indirgenir.

$$z_{if}=\frac{r_{if}-1}{M_f-1} \tag{4.13}$$

- [0,1] aralığına indirgenen değişkenlere ait birimler arasındaki uzaklıklar oransal ve aralık ölçekte kullanılan uzaklık ölçüleri ile bulunur.

4.2.4 Uzaklık ve Benzerlik Ölçülerinin Birbirilerine Dönüşümü

Benzerlik ölçüleri, uzaklık ölçülerine çeşitli dönüşümlerle dönüştürülebilirler. Burada uzaklık ölçüsü $d_{i,j}$ benzerlik ölçüsü $s_{i,j}$ olmak üzere aşağıdaki gibi dönüşümlerle birbirilerine dönüştürülürler.

Benzerlik ölçüsü aşağıdaki gibi uzaklık ölçüsüne dönüştürülebilir.

$$d_{i,j} = 1 - s_{i,j} \tag{4.14}$$

$$d_{i,j} = c - s_{i,j} \qquad \text{Buradaki c sabit bir değerdir.} \tag{4.15}$$

$$d_{i,j} = \sqrt{2\left(1 - s_{i,j}\right)} \tag{4.16}$$

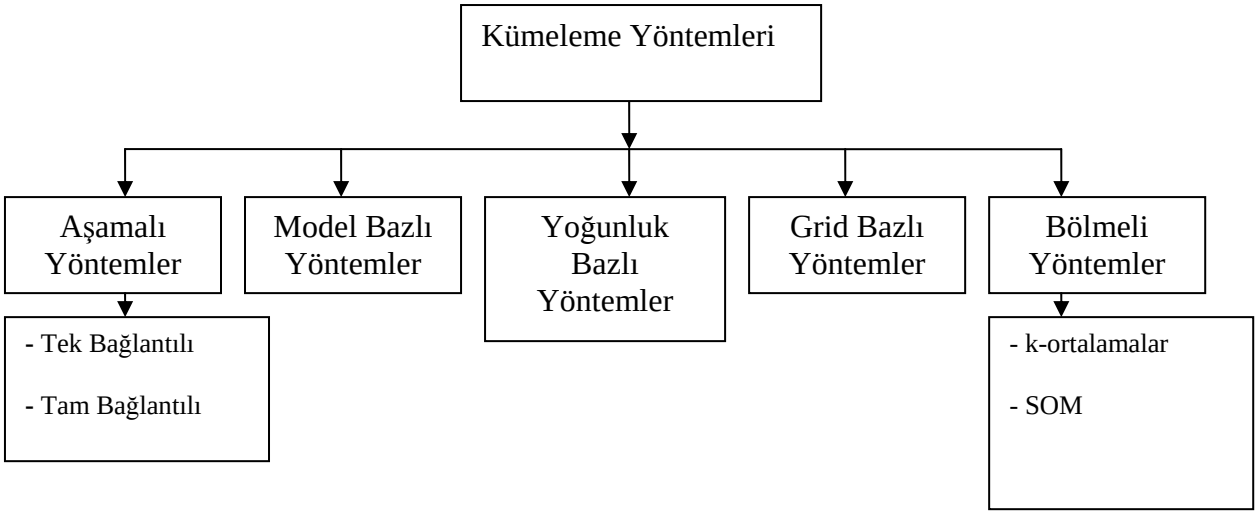
$$d_{i,j} = \sqrt{s_{ii} - 2s_{ij} + s_{jj}} \tag{4.17}$$

Bazı durumlarda uzaklık ölçüsünden benzerlik ölçüsünü elde etmek isteriz. Bu durumda aşağıdaki dönüşümü kullanabiliriz.

$$s_{i,j} = \left(1 + d_{i,j}\right)^{-1} \tag{4.18}$$

4.3 Kümeleme Yöntemlerinin Sınıflandırılması

Veri madenciliğinde, uygulamanın amacına, veri tipine, verinin büyüklüğüne göre farklı kümeleme yöntemleri bulunmaktadır. Değişik kaynaklarda farklı kümeleme yöntemleri farklı şekillerde sınıflandırılmaktadır. Örneğin çok değişkenli istatistik kitaplarında en genel haliyle kümeleme yöntemleri aşamalı ve aşamalı olmayan kümeleme yöntemleri diye ikiye ayrılmaktadır. Veri madenciliği yöntemleriyle ilgili kaynaklarda kümeleme yöntemleri aşağıdaki gibi sınıflandırılmaktadır.



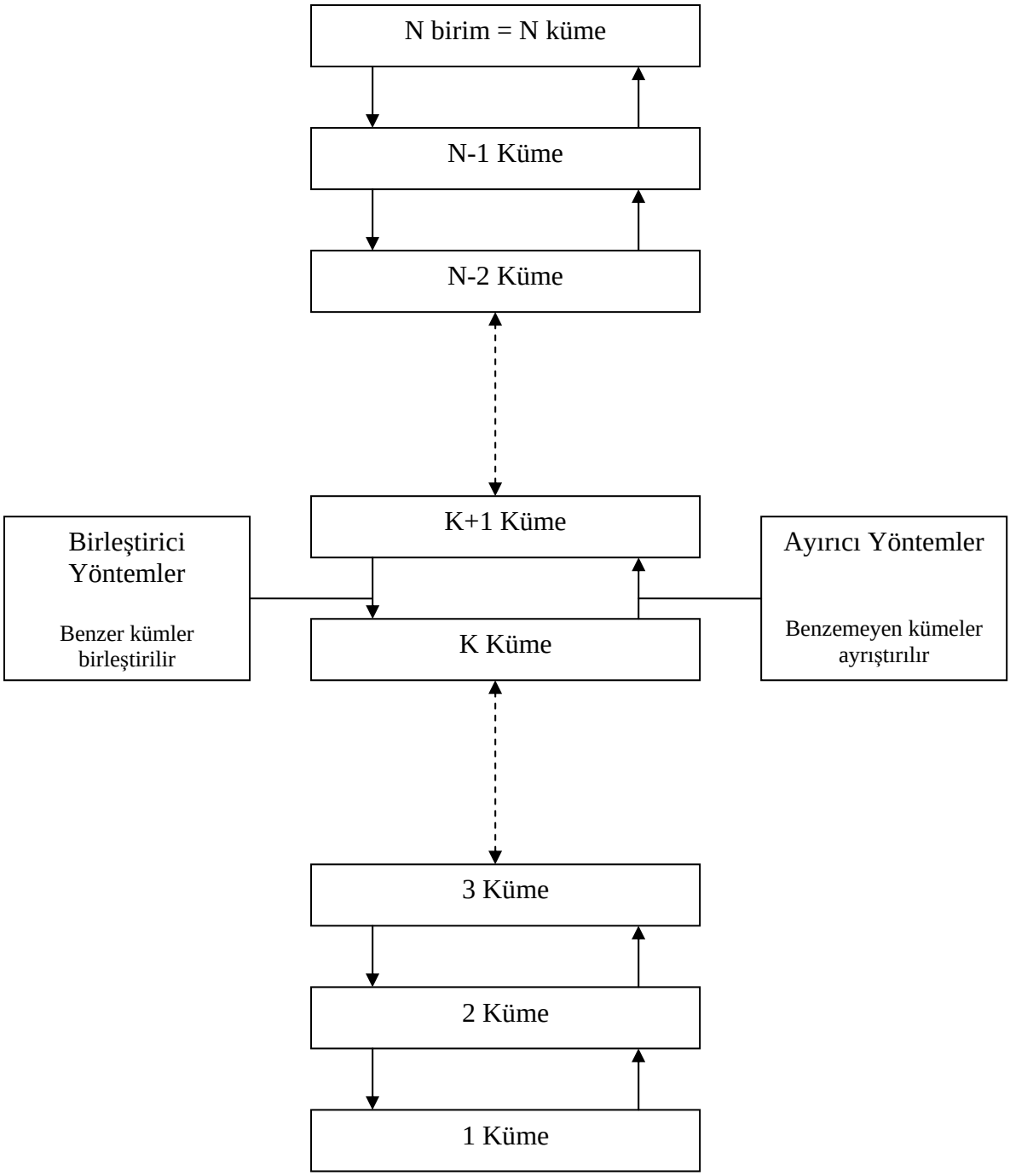
Şekil 4.3 Veri madenciliğinde kullanılan kümeleme yöntemleri

Şekil 4.3' de yer alan kümeleme yöntemlerinden aşamalı metotlar ve bölmeli metotlar çok değişkenli istatistik konusu içinde incelendiğinden bu yöntemlere tez çalışmasında değinilecektir. Aşamalı yöntemlerden tek bağlantılı (single link) ve tam bağlantılı (complete link), bölmeli kümeleme yöntemlerinden k-ortalamalar (k-means) ve kendinden düzenlenen haritalar (SOM) yöntemlerinden bahsedilecektir.

4.3.1 Aşamalı Kümeleme Yöntemleri (Hierarchical Clustering Methods)

Aşamalı (hiyerarşik) kümeleme yöntemleri veri setindeki birimlerin birbirilerine göre uzaklık veya benzerliklerini dikkate alarak birimleri birbirleriyle değişik aşamalarda bir araya getirerek ardışık biçimde kümeler belirlemeye ve bu kümelere girecek elemanların hangi uzaklık veya benzerlik düzeyinde küme elemanlarının olduğunu belirlemeye yönelik yöntemlerdir (Özdamar, 2004).

Hiyerarşik kümeleme yöntemleri, hiyerarşik ayrışmanın aşağıdan yukarıya veya yukarıdan aşağıya doğru olmasına göre birleştirici (agglomerative) ve ayırıcı (divisive) hiyerarşik kümeleme olmak üzere iki ana gruba ayrılır (Tan vd., 2006).



Şekil 4.4 Hiyerarşik kümeleme yöntemleri

Birleştirici hiyerarşik kümeleme yöntemleri, başlangıçta tüm birimleri tek başlarına ayrı birer küme oluşturduğunu kabul ederek, n birimi aşamalı olarak sırasıyla n, n-1, n-2, …, n-r, … , 3, 2, 1 kümeye ayırmayı amaçlar. Ayırıcı hiyerarşik kümeleme yöntemleri ise başlangıçta tüm birimleri birer küme olarak kabul ederek birimleri aşamalı olarak n birimi sırasıyla 1, 2, 3, …, n-r, …, n-3, n-2, n-1, n kümeye ayırmayı amaçlar. Ayırıcı hiyerarşik kümeleme yöntemi birleştirici hiyerarşik kümelemenin tersi bir yaklaşım kullanır (Özdamar 2004).

Şekil 4.4 birleştirici ve ayırıcı hiyerarşik kümeleme yöntemlerinin sergilediği kümeleme yaklaşımlarını güzel bir şekilde anlatmaktadır. Görüldüğü gibi hiyerarşik yöntemler iteratif yöntemlerdir. “Bu yöntemlerin en büyük olumsuzluğu, bir adım gerçekleştirildikten sonra bir daha tekrar aynı adıma geri dönülememesidir. Bu yüzden de yanlış kararları düzeltme imkanı vermemektedir (Bilen, 2004).” Ayrıca hiyerarşik kümeleme analizleri büyük veri setlerinde hesapsal karışıklıktan ve bellek gereksinimlerinden dolayı kullanışlı değillerdir. Bunun yanında bir çok çalışmada doğru sayıda küme seçiminde yol gösterici olabilmektedirler (Tan vd., 2006).

Birleştirici hiyerarşik kümeleme yöntemleri en sık kullanılan hiyerarşik kümeleme yöntemleridir. Bunun için tez çalışmasında en bilinen birleştirici hiyerarşik kümeleme yöntemlerinden tek bağlantılı (single link) ve tam bağlantılı (complete link) yöntemlerinden bahsedilecektir.

4.3.1.1 Tek Bağlantılı (Single Link) Hiyerarşik Kümeleme Yöntemi

En basit hiyerarşik kümeleme yöntemidir. Veri setindeki bir birimin m. küme olarak hangi birimlerle veya kümelerle birleştirileceği, birimlerin yeni oluşan kümelere olan minimum uzaklıkları dikkate alınarak belirlenir (Özdamar, 2004). Minimum uzaklıkları dikkate aldığı için tek bağlantılı kümeleme yöntemi, en yakın komşuluk (nearest neighbor) yöntemi olarak ta bilinmektedir (Martinez, 2005). Tek bağlantılı kümeleme yöntemi aşırı değerlere karşı duyarlıdır (Tan vd., 2006).

m. kümenin daha önce oluşan k. ve l. kümelerden hangisi ile birleşerek oluşacağını belirlemek için j. küme ile k. ve l. kümelerin uzaklıklarına bakılır. Bu uzaklıklardan en küçük olanı ile birleştirme yaparak m. küme belirlenir. m. kümenin j. kümeye olan uzaklığı d(m,j) aşağıdaki gibi bulunur.

$$d(m, j) = \min\{d(k, j), d(l, j)\} \tag{4.19}$$

Kümeler arasındaki minimum uzaklığa baktığımızda birimlerinde tek başlarına küme oluşturabileceklerini unutmamalıyız (Özdamar 2004).

4.3.1.2 Tam Bağlantılı (Complete Link) Hiyerarşik Kümeleme Yöntemi

Tam bağlantılı kümeleme yöntemi tek bağlantılı kümeleme yöntemine oldukça benzerdir. Ancak burada veri setindeki bir birimin m. küme olarak hangi birimlerle veya kümelerle birleşeceği birimlerin yeni oluşan kümelere olan maksimum uzaklıkları dikkate alınarak

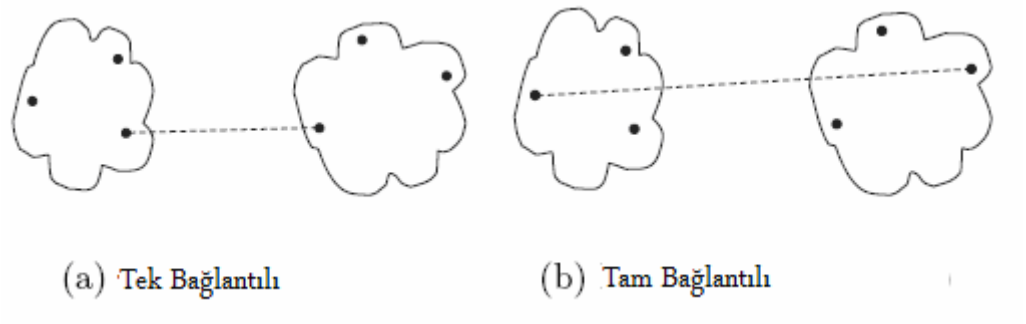
belirlenir (Özdamar 2004). Maksimum uzaklıkları dikkate aldığı için tam bağlantılı kümeleme yöntemi, en uzak komşuluk (furthest neighbor) yöntemi olarak ta bilinmektedir (Martinez ve Martinez, 2005). Tam bağlantılı kümeleme yöntemi tek bağlantılı kümeleme yöntemlerine göre aşırı değerlere karşı daha az duyarlıdır (Tan vd., 2006).

m. kümenin daha önce oluşan k. ve l. kümelerden hangisi ile birleşerek oluşacağını belirlemek için j. küme ile k. ve l. kümelerin uzaklıklarına bakılır. Bu uzaklıklardan en büyük olanı ile birleştirme yaparak m. küme belirlenir. m. kümenin j. kümeye olan uzaklığı d(m,j) aşağıdaki gibi bulunur.

$$d(m, j) = \max\{d(k, j), d(l, j)\} \tag{4.20}$$

Kümeler arasındaki maksimum uzaklığa baktığımızda birimlerinde tek başlarına küme oluşturabileceklerini unutmamalıyız (Özdamar 2004).

Hiyerarşik kümeleme yöntemlerinden tek bağlantılı ve tam bağlantılı yöntemler Şekil 4.5' de grafiksel olarak gösterilmişlerdir. Şekil 4.5' de bulunan bölgeler hiyerarşik aşamadaki kümeleri, bölgelerin içindeki noktalar birimleri göstermektedir.



Şekil 4.5 Kümeleme yöntemlerinin grafikleri (Tan. vd., 2006)

4.3.2 Bölmeli Kümeleme Yöntemleri (Partition Clustering Method)

Bölmeli kümeleme yöntemleri aşamalı olmayan kümeleme yöntemleridir. Bölmeli yöntemlerde, n birimim k<n olmak üzere k kümeye parçalanması rastgele veya gelişigüzel yapılabilir. Bu yöntemde birimleri ayırmak istediğimiz küme sayısını belirledikten sonra, kümeler için belirlenen küme ayırma kriterlerine göre birimlerin hangi kümelere gireceğine karar verilir ve atama işlemi gerçekleştirilir. Kümeler tarafsız bölme kriteri olarak nitelendirilen bir kritere uygun oluşturulduğu için aynı kümedeki birimler birbirlerine

benzerken, farklı kümedeki birimlerden farklıdırlar. Bölmeli yöntemler hiyerarşik yöntemlere oranla daha büyük veri setlerine uygulanabilir (Özdamar, 2004).

Bölmeli yöntemler sonucu elde edilen k adet küme şu koşulları sağlar:

- Her bir küme en azından bir birim içerir.
- Her birim yalnızca bir kümeye yerleştirilebilir.

Fakat fuzzy bölmeleme tekniklerinde 2. koşulun sağlanması beklenmez (Bilen, 2004)

Bölmeli yöntemler arasında en popüler ve en basit istatistik tabanlı yöntem k-ortalamalar bölmeli yöntemidir.

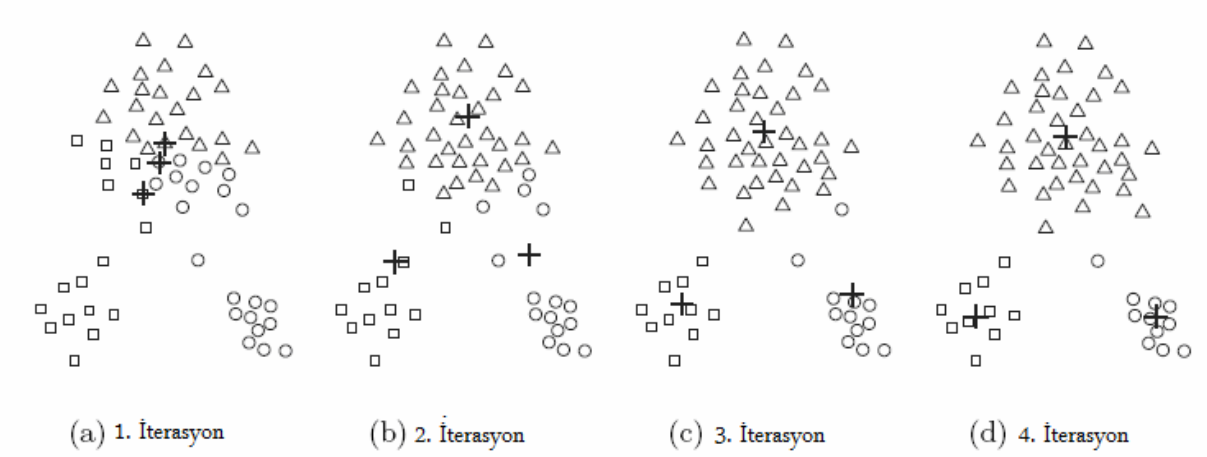
4.3.2.1 K- Ortalamalar Kümeleme Yöntemi (K-Means Clustering Method)

"k- ortalamalar kümeleme yöntemi, çok sayıda birimden elde edilen p değişkenli veri setlerini küme içi kareler toplamını minimize edecek biçimde k kümeye ayırmayı amaçlar. Birimlerin az sayıda kümeye yerleştirilmesi iteratif bir biçimde yapılır. Birimler her iterasyonda farklı kümelere atanarak en uygun çözüm permutasyonel bir yaklaşım ile belirlenir (Özdamar, 2004)." Her bir iterasyonda oluşan küme de, değişkenlerinin ortalamaları alınarak yeni küme merkezleri belirlendiği için k-ortalamalar yönteminin uygulanabilmesi için veri setindeki değişkenlerin en azından aralık ölçekte bulunması gerekir (Bilen, 2004).

k-ortalamalar kümeleme yönteminin algoritması şu biçimdedir:

1. Ayırmak istediğimiz k küme sayısı belirlenir.
2. k adet birim başlangıç küme merkezi olarak rastgele veya özel olarak seçilir.
3. Küme merkezi olmayan birimler belirlenen uzaklık ölçülerine göre küme merkezlerine olan uzaklıkları hesaplanır.
4. Her birim en yakın olduğu küme merkezine atanır.
5. Yeni küme merkezleri, oluşturulan k adet başlangıç kümesindeki değişkenlerin ortalamaları alınarak oluşturulur.
6. Birimler belirlenen uzaklık ölçülerine göre en yakın oldukları oluşturulan yeni küme merkezlerine atanırlar.
7. Birimlerin, bir önceki küme merkezlerine olan uzaklıkları ile yeni oluşturulan küme merkezlerine olan uzaklıkları karşılaştırılır.
8. Uzaklık makul görülebilir oranda azalmış ise 6. adıma dönülür.

9. Eğer çok büyük bir değişiklik söz konusu olmamış ise iterasyon sona erdirilir (Bilen 2004; Martinez ve Martinez 2005).



Şekil 4.6 Örnek bir veride 3 kümenin k-ortalamalar yöntemiyle bulunması (Tan vd., 2006)

Şekil 4.6' de k-ortalamalar kümeleme yönteminin nasıl çalıştığı iterasyonları ile verilmektedir. Şekil 4.6' da örnek verimiz 3 kümeye ayrılmak istenmektedir. Bunun için ilk önce Şekil 4.6 (a)' da ki gibi 3 küme merkezi '+' simgesiyle gösterilmek üzere rastgele seçilir. Küme merkezlerine göre aynı kümede bulunan birimlerin her biri aynı simgeyle gösterilir. Daha sonra küme merkezleri küme birimlerinin ortalamalarını alarak güncellenir. Şekil 4.6 (b), Şekil 4.6 (c)' de güncellenen küme merkezleri gösterilmektedir. Güncellenme ta ki küme merkezlerinde herhangi bir değişim olmayıncaya kadar devam eder. Algoritmanın sonunda iterasyon sona erdirilir. Algoritma sonucunda Şekil 4.6 (d)'de ki gibi doğal kümeler bulunur.

k-ortalamalar kümeleme yönteminde iterasyonun durdurulması için kullanılan ölçütlerden birisi, kareli hata ölçüsüdür. Bu ölçüt, belirlenen uzaklık ölçülerine göre, her birimin en yakın olduğu küme merkezlerine olan kareli toplam uzaklıkları (SSE Sum of Square Eroor) minimize etmeyi amaçlar. SSE' lerin küçük olması küme merkezlerinin kümeleri iyi temsil ettiğinin bir göstergesidir.

Öklidyen uzaklık ölçüsünü göz önüne alırsak SSE aşağıdaki gibi gösterilir. Aşağıdaki denklemlerde C_i i. kümeyi, x i. kümdeki bir birimi ve m_i i. kümedeki birim sayısını, c_i i. kümenin ortalamasını ve K küme sayısını gösterir.

$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} d(c_i, x) = \sum_{i=1}^{K} \sum_{x \in C_i} (c_i - x)^2 \qquad (4.21)$$

Öklidyen uzaklık ölçüsünü göz önüne alırsak, SSE' yi minimum yapan değer, kümenin merkezinin, kümenin birimlerinin ortalaması olarak alınmasıdır. Aşağıdaki denklemde bu gösterilir.

$$c_i = \frac{1}{m_i} \sum_{x \in C_i} x \tag{4.22}$$

İspat:

Öklidyen uzaklık ölçüsünü göz önüne alarak aşağıdaki gibi toplam kareler uzaklığı (SSE) minimize edilir. Özelikle kümeleme analizlerinde üzerinde durduğumuz nokta, küme merkezlerini, SSE' leri olabildiğince minimize edecek şekilde seçmektir.

$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} d(c_i, x) = \sum_{i=1}^{K} \sum_{x \in C_i} (c_i - x)^2 \tag{4.23}$$

$$\frac{\partial}{\partial c_k} SSE = \frac{\partial}{\partial c_k} \sum_{i=1}^{K} \sum_{x \in C_i} (c_i - x)^2 \tag{4.24}$$

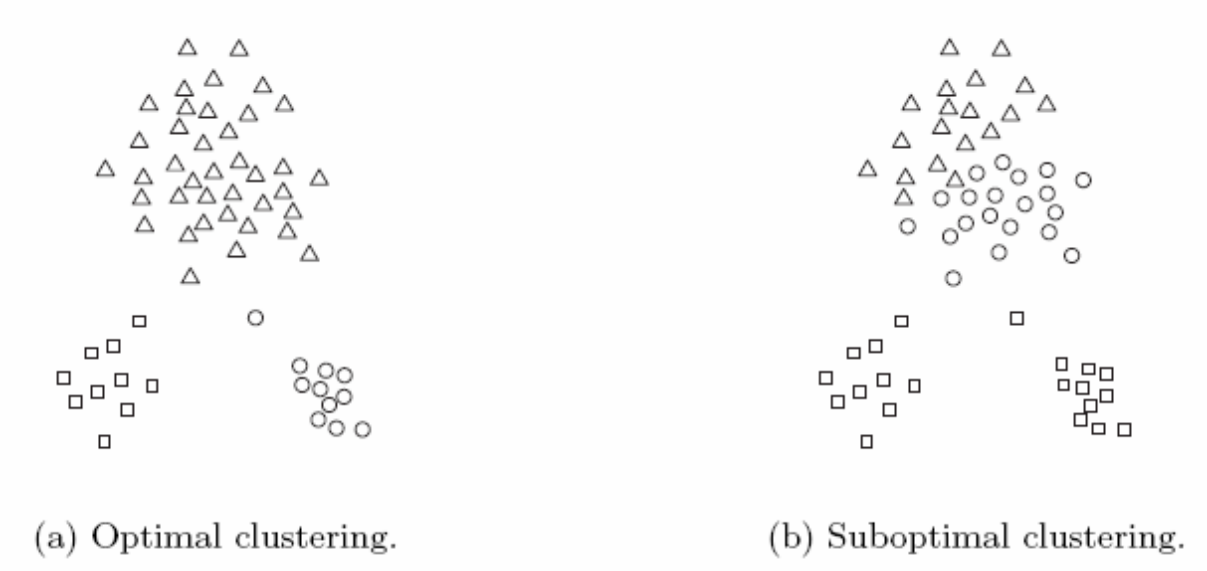
$$= \sum_{i=1}^{K} \sum_{x \in C_i} \frac{\partial}{\partial c_k} (c_i - x)^2 \tag{4.25}$$

$$= \sum_{x \in C_k} 2 * (c_k - x_k) = 0 \tag{4.26}$$

$$\sum_{x \in C_k} 2 * (c_k - x_k) = 0 \Rightarrow \sum_{x \in C_k} x_k \Rightarrow c_k = \frac{1}{m_k} \sum_{x \in C_k} x_k \tag{4.27}$$

Yukarıdaki denklemlerden gözüktüğü gibi k. kümenin ortalaması SSE yi minimum yapar. Bu durumda, örneğin (1,1), (2,3) ve (6,2) nokta çiftlerinin en uygun küme merkezi ((1+2+6)/3,(1+3+2)/3) = (3,2) olur. Çeşitli uzaklık fonksiyonları kullanarak değişik SSE yi minimum kılan noktalar da bulunabilir. Örneğin öklid uzaklık fonksiyonu yerine manhattan uzaklık fonksiyonu kullanılırsa SSE' yi minimum kılan küme merkezi kümenin medyanı olur.[*] Ancak burada bütün olanaklı sonuçlar denenerek elde edilen minimum SSE, yerel minimumu bulmada işe yarar; toplam SSE' yi yani global SSE' yi minimum kılmayı garantilemez (Tan vd., 2006).

[*] İspatı için, Tan vb. 2006, Introduction to Data Mining, sf 514



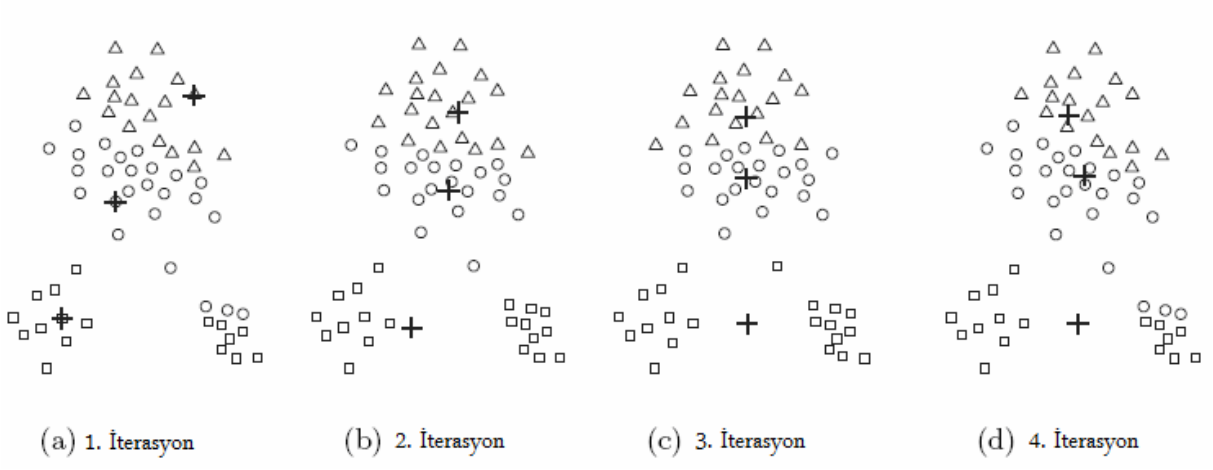
Şekil 4.7 Globel ve yerel optimum kümeler (TAN P. vd., 2006)

Şekil 4.7’ de global (optimal clustering) ve yerel (suboptimal clustering) en küçük SSE’ ler için elde edilen 3 küme bulunmaktadır.

4.3.2.1.1 Başlangıç Küme Merkezlerinin Seçilmesi

K-ortalamalar kümeleme yönteminde başlangıç küme merkezleri genelde tesadüfi olarak seçilir. Bunun neticesinde aynı veri seti için k-ortalamalar kümeleme yöntemi farklı zamanlarda uygulandığında farklı SSE toplamları elde edilebilir. Bu yüzden başlangıç küme merkezlerini tesadüfî olarak seçmek her zaman elverişli olmayabilir.

Şekil 4.6, 4.7, 4.8 deki şekillerde aynı veri seti için farklı başlangıç noktaları kullanarak k-ortalamalar yöntemiyle elde edilen küme sonuçları bulunmaktadır. Şekil 4.6’ da aynı doğal kümeden seçilen başlangıç küme merkezleri kullanılarak kümeleme sonucunda minimum SSE’ ler elde dilmişken, Şekil 4.8’ de iyi dağıldığı gözüken başlangıç küme merkezleri kullanarak elde edilen kümeleme sonucunda daha yüksek SSE’ ler elde edilmiştir.



Şekil 4.8 K-ortalamalar kümeleme için kötü başlangıç noktaları (Tan vd., 2006)

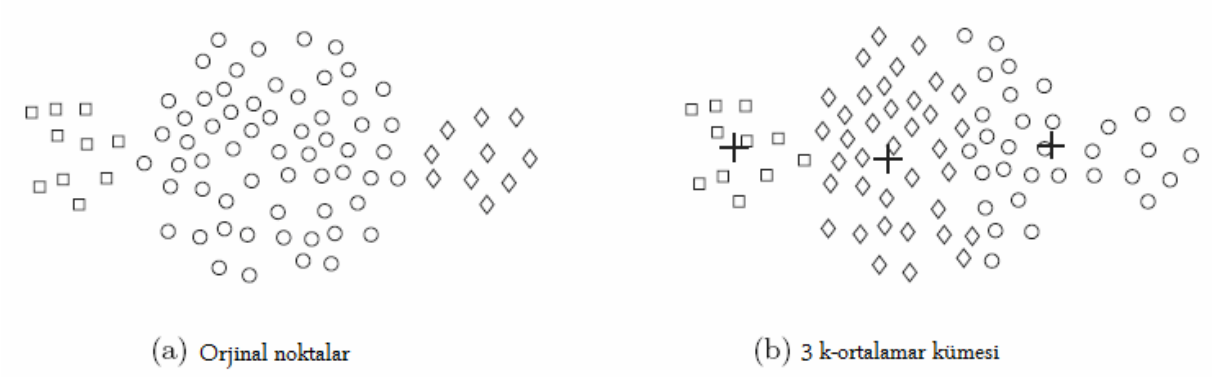
Başlangıç küme merkezlerinin tesadüfî seçilmesinden doğan kümeleme yaklaşımının yetersizliğini ortadan kaldırmak için değişik öneriler ortaya atılmıştır. En etkili yaklaşımlardan bir tanesi de veri setimizden bir örnek alarak örnek veri setine hiyerarşik kümeleme analizini uygulamaktır. Daha sonra hiyerarşik kümelemeden elde edilen sonuçlardan istenilen k adet küme çıkartılır. Çıkartılan bu kümelerin ortalamaları ya da merkezleri, k-ortalamalar kümeleme yönteminde başlangıç küme merkezi olarak kullanılmaktadır. Bu yaklaşımın pratikte iyi sonuçlar vermesine rağmen karşılaştığı sorunlar da bulunmaktadır. Öncelikle, hiyerarşik kümeleme analizinin büyük verilerde kullanılmasının elverişsizliğinden dolayı, ancak küçük örnekler seçilerek hiyerarşik kümeleme analizi uygulanabilir. Karşılaşılan diğer bir sorunda büyük veri setini temsil edebilecek iyi bir örneğin seçilebilmesidir (Tan vd., 2006).

Başlangıç küme merkezlerini belirlemede kullanılan diğer bir yaklaşımda birbirinden uzak birimler belirlemektir. Birbirinden uzak birimler belirleyebilmek için öncelikle tesadüfi yada bütün birimlerin merkezi olacak şekilde bir birim seçilir. Daha sonra şimdiye kadar seçilen birimlerden mümkün olduğunca uzakta olacak yeni bir birim daha seçilir. Bu şekilde istenilen küme sayısı kadar küme merkezi seçilebilir. Böylece kümelemede kullanılacak başlangıç küme merkezleri tesadüfî seçilmemiş ve bir birinden uzakta birimler olarak seçilmiş olur. Ancak bu yaklaşımda da bazı sorunlarla bulunmaktadır. Birbirinden uzak birimleri seçme esnasında birimlerin yoğun olduğu yerden ziyade sapan değerler başlangıç küme merkezi olarak seçilebilir. Hatta seçilen başlangıç birimlerden mümkün olduğunca uzak birimlerin seçilmesi kolay olmayabilir. Bu problemlerin üstesinden gelmek için veri setinden örnek çekme yoluna gidebilir. Bu sayede sapan değerlerin örneğimizde bulunma olasılığı düşer. Örneğin küçük seçildiği durumlarda da özellikle yoğun bölgelerden birimlerin örneğimizde

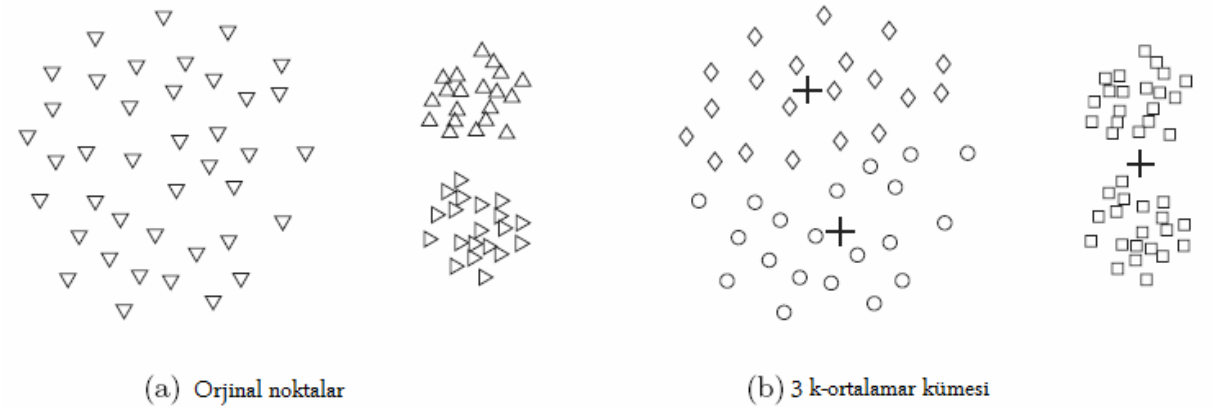
yer alma olasılığı oldukça yüksek olabilmektedir. Örneğimizin veri setimize göre küçük olmasından dolayı birbirinden uzak noktaların seçilmesi kolaylaşmaktadır (Tan vd., 2006).

4.3.2.1.2 K-Ortalamalar Kümeleme Yöntemi ve Farklı Tipte Küme Yapıları

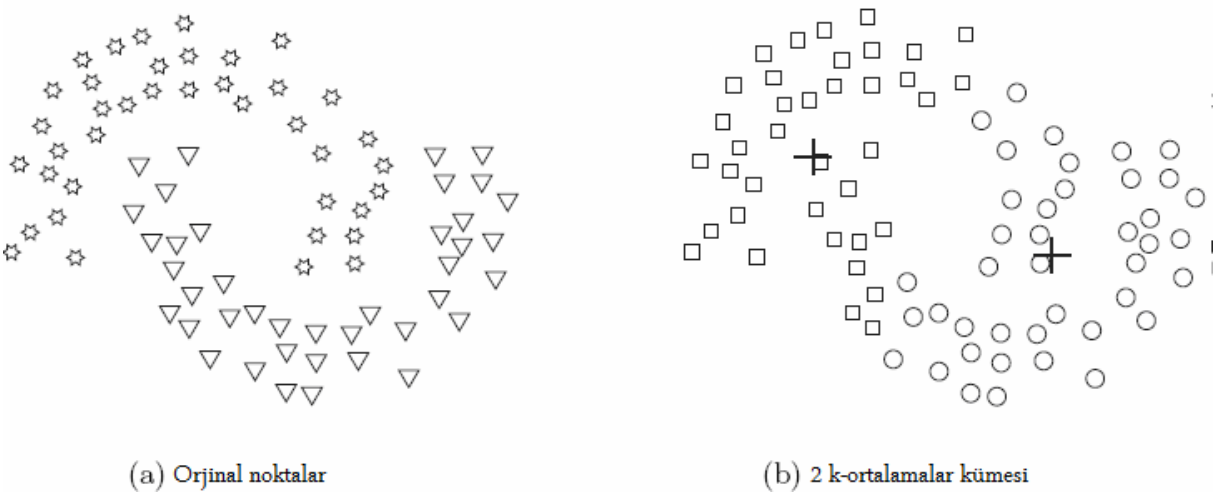
K-ortalamalar kümeleme yöntemi farklı küme yapıları için birtakım kısıtlara sahiptir. Özellikle doğal küme yapılarının küresel biçimde olmaması, oldukça farklı küme hacimlerine ve yoğunluklarına sahip olması k-ortalamalar yönteminin başarısız sonuçlar vermesine neden olabilmektedir. Bu durumu örneklendirmek için Şekil 4.9, 4.10, 4.11' e bakabiliriz. Şekil 4.9' daki kümelerden bir tanesinin hacmi diğer iki kümeye göre oldukça büyüktür. Küme hacminin büyük olmasından dolayı k-ortalamalar yöntemi doğal küme yapılarını bulmada başarısız olmuştur. Şekil 4.9' da doğal küme yapısı küçük olan bir küme, k-ortalamalar yöntemi sonucunda büyük bir küme olarak bulunmuştur. Şekil 4.10' daki kümelerden iki tanesi diğer büyük hacimli kümeye göre oldukça büyük bir yoğunluğa sahip olduğu için k-ortalamalar yöntemi doğal küme yapılarını bulmada başarısız olmuştur. Son olarak, Şekil 4.11' de küme yapıları küresel olmadığı için k-ortalamalar yöntemi doğal küme yapılarını bulmada başarısızdır (Tan vd., 2006).



Şekil 4.9 Farklı hacimli kümeler için k-ortalamalar (Tan vd., 2006)



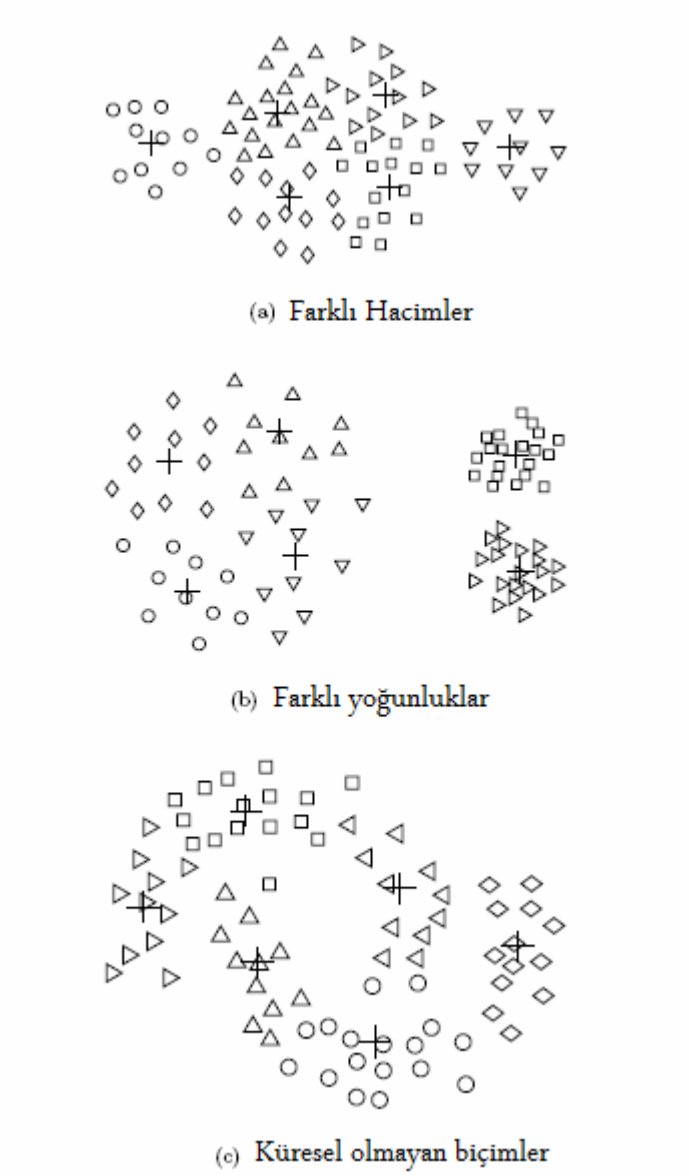
Şekil 4.10 Farklı yoğunluklu kümeler için k-ortalamalar (Tan vd., 2006)



Şekil 4.11 Küresel biçimde olmayan kümeler için k-ortalamalar (Tan vd., 2006)

Doğal küme yapılarını bulmak için kullandığımız k-ortalamalar kümeleme yöntemi yukarıda bahsedilen üç kısıttan dolayı başarısız sonuçlar verebilmektedir. Bu kısıtların yarattığı sorunlar, doğal küme yapılarının birden fazla alt kümeye ayrılmasıyla ortadan kaldırılabilir.

Şekil 4.9, Şekil 4.10 ve Şekil 4.11’ de kullanılan veriler için çizilen Şekil 4.12’ de, iki ve üç doğal küme yapısından altı tane alt küme elde edilmiştir. Bu sayede k-ortalamalar yöntemiyle doğal küme yapılarının yanlış kümelenmesinin önüne geçilmiştir (Tan vd., 2006).



Şekil 4.12 Doğal kümelerin alt kümeleri için k-ortalamalar (Tan vd., 2006)

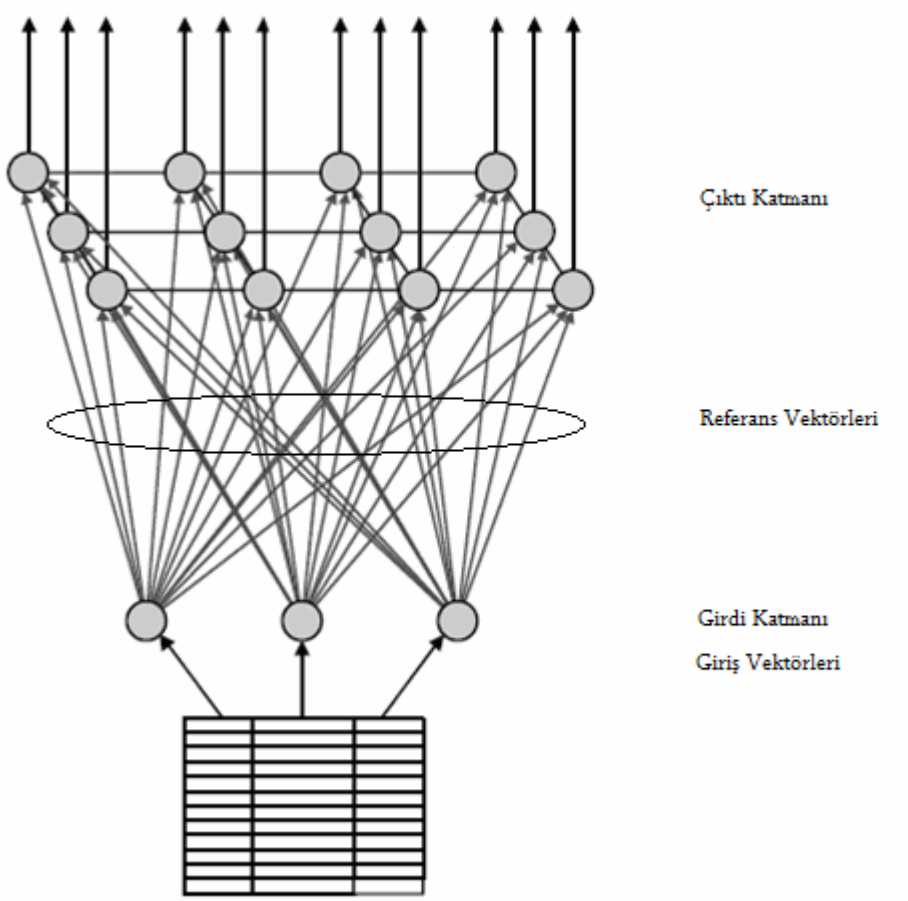
4.3.2.2 Kendinden Düzenlenen Haritalar (Self Organizing Maps - SOM)

Kümeleme çalışmalarında, klasik istatistiksel yöntemler yerine yapay sinir ağları da kullanılabilmektedir. Kümeleme çalışmalarında en çok kullanılan yapay sinir ağları, 1982 yılında Teuvo-Kohonen tarafından geliştirilen, SOM (Self-Organizing Maps) sinir ağlarıdır. SOM sinir ağları Kohonen SOM ağları olarak ta bilinmektedirler (Zontul vd., 2004).

SOM ağları, denetimsiz öğrenme (Unsupervised Learning) algoritmasıyla çalışırlar. Bu sebeple SOM ağında olayları öğrenmek için bir öğretmen veya ağın üretmesi gereken çıktıların ağa söylenme zorunluluğunun yoktur. Bu özeliğiyle SOM ağları özellikle beklenen çıktıların belirlenemediği durumlar için kullanılmaktadır. Bu sebeple, SOM ağları kümeleme analizi gibi problemlerin çözümünde tercih edilebilmektedir (Öztemel, 2006).

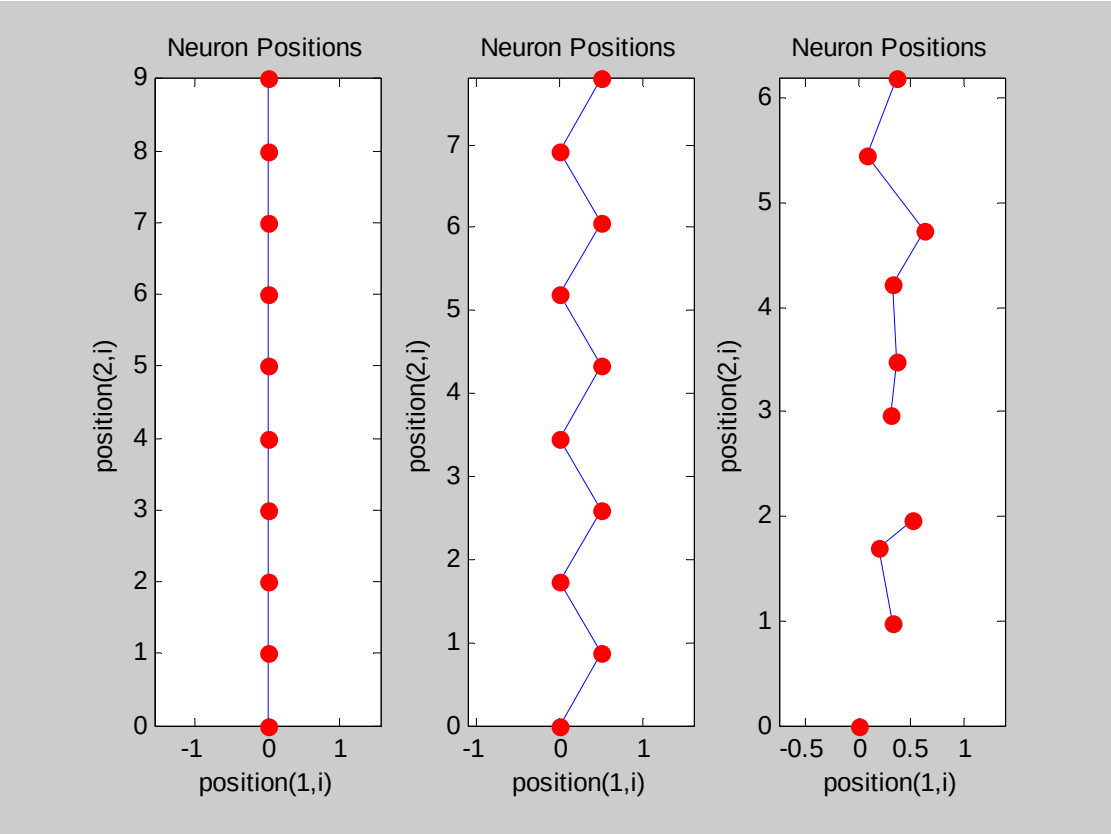
SOM ağları, veri setindeki birimleri hem kümelendirebilir hem de görsel olarak haritalandırabilir. Bu sebeple SOM ağları, klasik istatistikteki k-ortalamalar ile çok boyutlu ölçekleme yöntemlerinin her ikisinin de işlevlerini yerine getirebilmektedir. SOM ağları, hem verilerin kümelenmesi, hem de görselleştirilmesi için tercih edilmektedirler (Zontul vd., 2004).

“SOM ağları, tek katmanlı bir ağ olup giriş ve çıkış nöronlarından oluşur. Giriş nöronlarının sayısını veri setindeki değişken sayısı belirler. Çıkış nöronlarının her biri bir kümeyi temsil eder. Şekil 4.13’de bir SOM ağı görülmektedir. Diğer yapay sinir ağlarından farklı olarak SOM ağlarında, çıkış katmanındaki nöronların dizilimi çok önemlidir. Bu dizilim doğrusal, dikdörtgensel, altıgen veya küp şeklinde olabilir. En çok dikdörtgensel ve altıgen şeklindeki dizilimler tercih edilmektedir. Pratikte, çoğu kez dikdörtgensel dizilim karesel dizilim olarak uygulanır. Buradaki dizilim topolojik komşuluk açısından önemlidir. Aslında, çıkış nöronları arasında doğrudan bir bağlantı yoktur. Giriş nöronları ile her bir çıkış nöronu arasındaki bağlantıyı referans vektörleri (code-book vectors) gösterir. Bu vektörler bir katsayılar matrisinin sütunları olarak da düşünülebilir. SOM sinir ağları eğitilirken bu topolojik komşuluk referans vektörlerinin yenilenmesinde kullanılır (Zontul vd., 2004).”

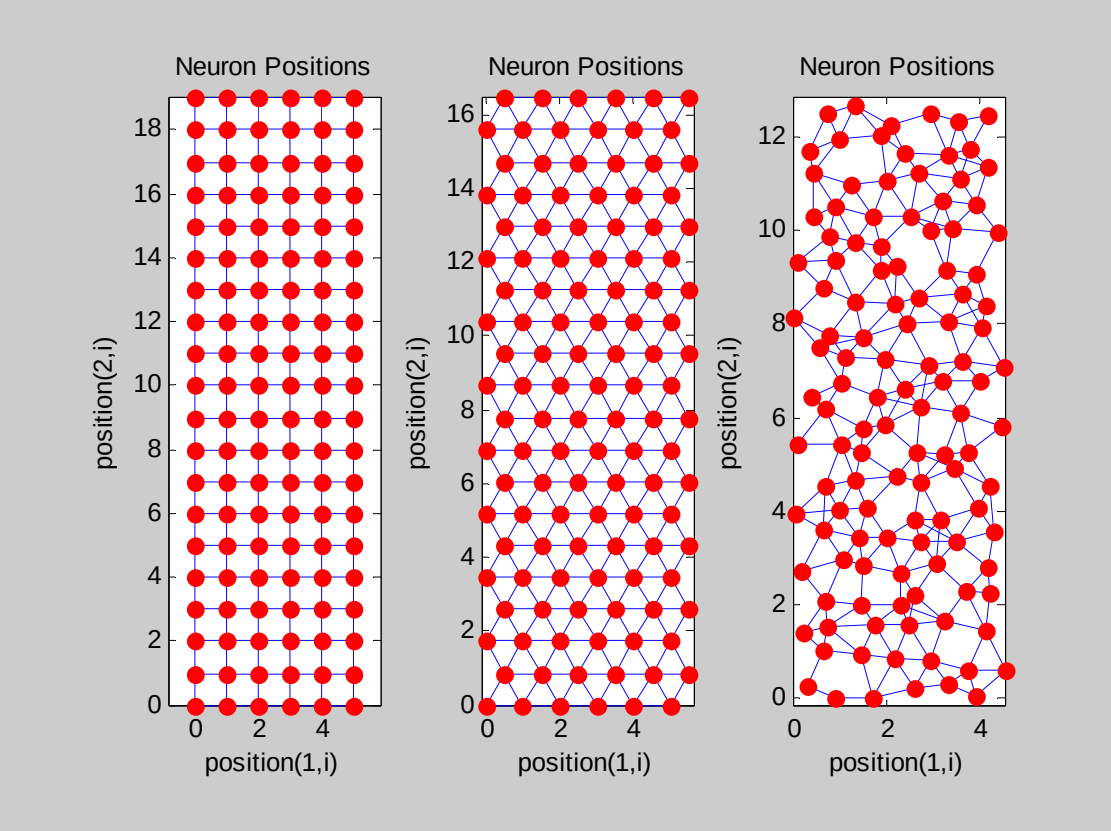


Şekil 4.13 Kohonen SOM sinir ağı (Beryy ve Linoff, 2004)

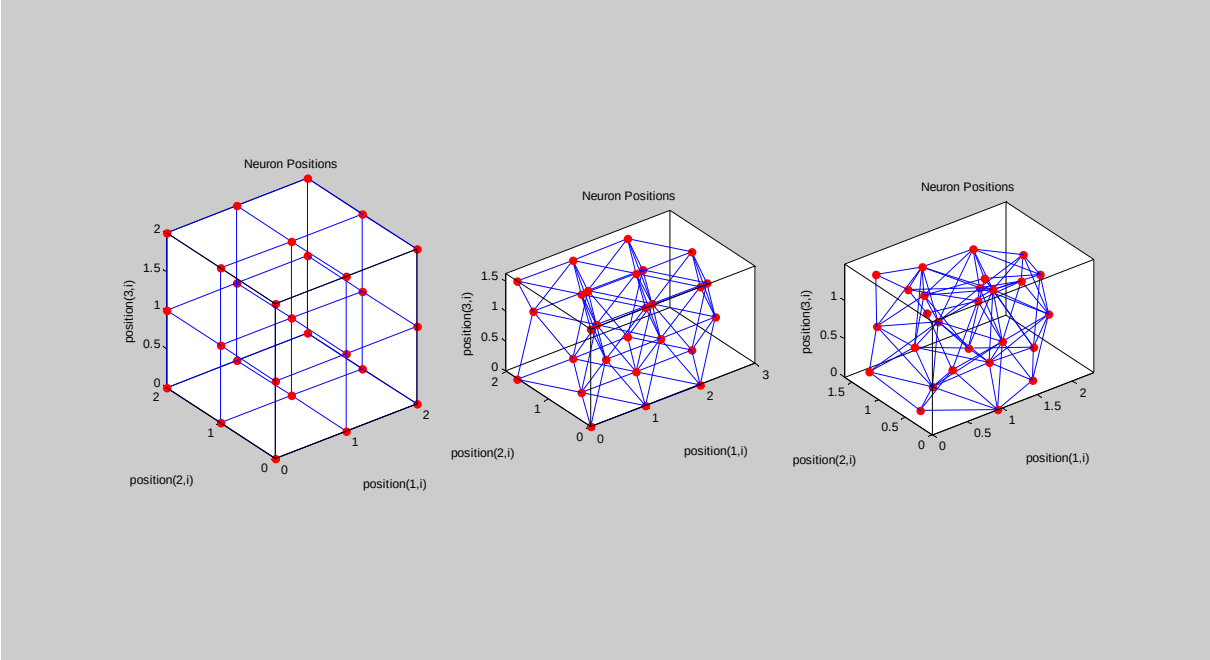
SOM ağlarındaki çıkış katmanlarına, Şekil 4.14, Şekil 4.15 ve Şekil 4.16’ da ki dikdörtgensel, altıgen ve rastsal nöron dizilimleri örnek verebilir. Şekil 4.14, Şekil 4.15 ve Şekil 4.16’ da ki nöron dizilimleri sırasıyla tek, iki ve üç boyutludur. Nöron dizilimleri hatta üç boyuttan daha yüksek boyutta da olabilir. Ancak, SOM’ da nöronların 2 boyutlu bir düzlemde dizilmesi sonuçların görselleştirilmesi ve yorumlanmasının kolaylığı açısından tercih edilmektedir.



Şekil 4.14 Tek boyutlu nöron dizilimleri



Şekil 4.15 İki boyutlu nöron dizilimleri



Şekil 4.16 Üç boyutlu nöron dizilimleri

4.3.2.2.1 SOM Öğrenme Algoritması

"Kohonen ağlarında kullanılan öğrenme algoritması bu ağlara ismini de veren, SOM (Self Organizing Maps) algoritmasıdır. Bu ağlarda kullanılan öğrenme algoritması denetimsizdir. Yani, ağ eğitilirken bağımlı değişken kullanılmaz. Veri setindeki giriş vektörleri ağa girildikçe ağ kendi kendini düzenler ve referans vektörleri oluşur. Bu algoritma aşağıda verilmiştir:" (Zontul vd., 2004)

Bu algoritmada kullanılan semboller:

$x_n = x_{n1}, x_{n2}, \cdots, x_{nm}$: m özellik ve n kayıttan oluşan mxn' lik x veri matrisi için giriş vektörleri

$w_j = w_{1j}, w_{2j}, \cdots, w_{mj}$: m tane ağırlıktan oluşan j çıkış nöronları için referans vektörleri

d(i,j) : giriş vektörünün (i,j) koordinatındaki çıkış nöronuna olan öklid uzaklığının karesi.

J : giriş vektörünün en yakın olduğu çıkış nöronları.

α : öğrenme katsayısı.

h : komşuluk fonksiyonu

c : kazanan nöron

Algoritma:

1. w_{ij} katsayılarına ilk değer ata.
   Topolojik komşuluk (R) parametrelerini belirle
   Öğrenme katsayısı (α) parametrelerini ayarla
2. Giriş vektörü x_n ve ağırlık vektörü w_j için $d(w_j, x_n) = \sqrt{\sum_i (w_{ij} - x_{ni})^2}$ şeklinde öklid uzaklıklarını hesapla.
3. Bütün nöronlar için $d(w_j, x_n)$' nin minimum olduğu j kazanan nöronu bul
4. Komşuluk parametresi R için j kazanan nöronun J komşuluk nöronlarını bul.
5. t. iterasyonda j' nin belirtilen komşuluğundaki bütün çıkış nöronları (J) için aşağıdaki gibi referans vektörlerini güncelle
   $$w_{ij}(t+1) = w_{ij}(t) + \alpha(t) h_{ci}(t)(x_{ni} - w_{ij}(t)) \quad (4.28)$$
6. Öğrenme katsayısını güncelle.
7. Belirtilen zamanlarda topolojik komşuluk parametresini azalt. (Larose, 2005 ve Zontul vd., 2004)

Yukarıdaki algoritmadan da anlaşılacağı üzere, ilk önce referans vektörlerine bir ilk değer atanır. Bu atama işlemi genellikle rastlantısal yapılır. Döngüye başlamadan önce öğrenme katsayısı (α) ve komşuluk değişkenine (R) yüksek bir değer atanır. α'ya 0 ile 1 arasında bir değer atanır. Bu değerin 1'e yakın olması tercih edilir. Algoritma için bir döngü veri setindeki tüm satırların birer kere SOM ağına girdi olarak sunulmasıdır. Veri setinin bir satırı x_n vektörüdür. x_n vektörünün çıkış katmanındaki her bir nörona olan öklid uzaklığının karesi bulunur. Çıkış katmanındaki her bir nöronu bir referans vektörü (w_j) temsil eder. Dolayısıyla, bu uzaklık x_n vektörü ile (w_j) arasındaki uzaklıktır. Hesaplanan uzaklıklardan en küçüğü bulunur. Bu uzaklık hangi çıkış nöronuna aitse o nöron kazanan (Winner neuron) nörondur. Literatürde kazanan nöron için BMU (Best Matching Units) kısaltması çoğunlukla tercih edilir. Kazanan nöron aşağıdaki gibi ifade edilebilir (Zontul vd., 2004).

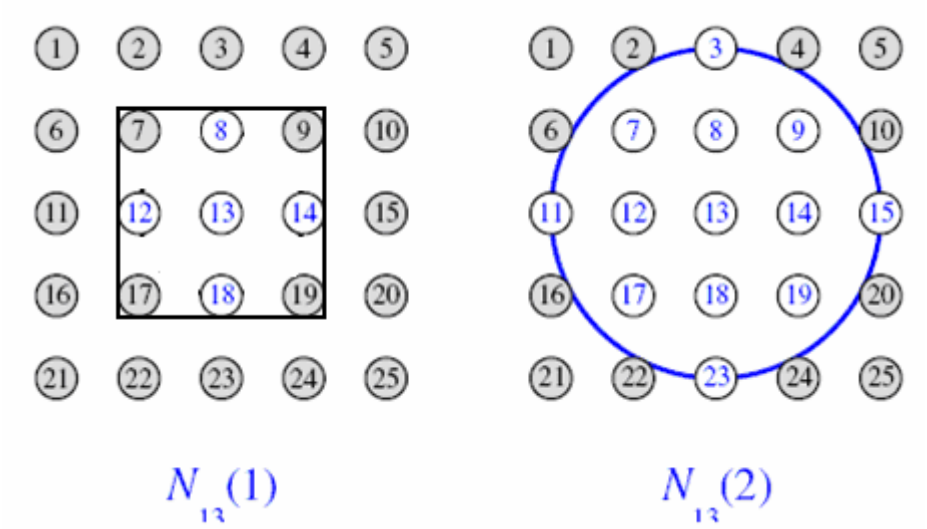
$$c : w_c(t) = \min_t \|x(t) - w_i(t)\| \quad (4.31)$$

SOM ağları "yarışmacı " bir ağdır. Kazanan nöron ve komşu nöronların referans vektörleri yeniden hesaplanır. SOM öğrenme algoritmasında kazanan nöronu bulmak için öklit uzaklıklar yerine kümeleme analizlerinde kullanılan diğer uzaklık ölçüleri de kullanılabilir.

Komşu nöronlarının bulunmasında doğrusal, karesel, dikdörtgensel ve altıgen gibi değişik yöntemler kullanılabilir. Şekil 4.17' de kazanan nöronun dikdörtgensel ve dairesel komşulukları görülmektedir. Bu şekillerden de görüldüğü gibi, dikdörtgensel komşulukta kazanan nöronun etrafında daha fazla komşu nöron bulunmaktadır (Demuth vd., 2005).

j kazanan nöronun komşulukları denklem (4.29) da ki ifadeyle bulunur.

$$N_i(d) = \{j, d_{ij} \le d\} \tag{4.29}$$



Şekil 4.17 Kazanan Nöronun dikdörtgensel R=1 ve dairesel R=2 komşuluğu

Şekil 4.17 da kazanan nöron için 1 dikdörtgensel ve 2 dairesel komşuluğu aşağıdaki gibidir.

$$N_{13}(1) = \{8,12,13,14,18\} \tag{4.30}$$

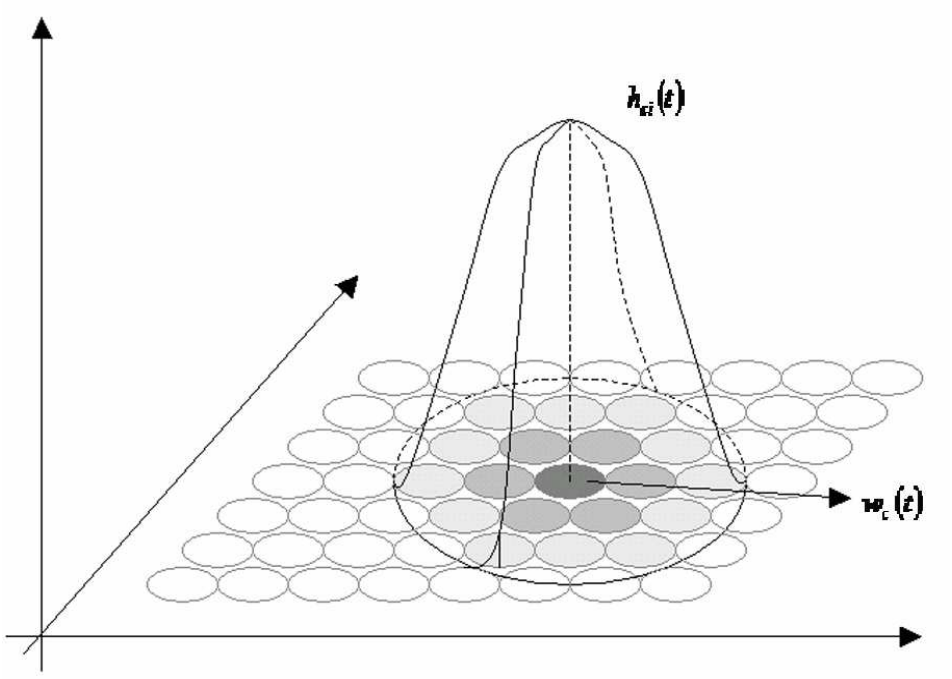
$$N_{13}(2) = 3,7,8,9,11,12,13,14,15,17,18,19,23 \tag{4.31}$$

t iterasyonu için kazanan nöron ve komşu nöronların referans vektörleri $w_{ij}(t+1) = w_{ij}(t) + \alpha(t)h_{ci}(t)(x_{ni} - w_{ij}(t))$ eşitliği kullanılarak güncellenir. Burada, x_n vektörü ile w_j referans vektörü arasındaki fark, öğrenme katsayısı α ile çarpılır ve w_j referans vektörüne ilave edilir. "Bu sebeple, w_j referans vektörlerine ilk değer olarak çok küçük değerler verilmişse α değeri 1'e yakın bir değer almalıdır. Böylece, referans vektörleri kendilerini oluşturma şansına sahip olurlar. Veri setindeki tüm satırlar için bu işlemler tekrarlandığında bir döngü tamamlanmış olur (Zontul vd., 2004).

Eğitim yani döngüler devam ettikçe referans vektörleri değişmeye devam eder. Eğitim boyunca, güncellenen birimlerin ağırlık vektörleri giriş desenine bir miktar yaklaşmaktadır. Ağırlık vektörlerinin değişim hızı öğrenme oranı denilen α(t) ile belirlenir ve bu oran zamanla

azaltılarak en sonunda 0 yapılır. Etkileşime dahil edilecek birimler, komşuluk fonksiyonu denilen $h_{ci}(t)$ ile belirlenir. Etkileşime dahil edilen bu birimlerin sayısı da zamanla azalır ve eğitim işleminin sonuna doğru sadece kazanan birim etkileşime girer. Tipik olarak, komşuluk fonksiyonu tek tepeli bir fonksiyon olup kazanan birimin bulunduğu yerin çevresinde simetrik ve kazanandan uzaklaştıkça tekdüze azalan bir yapıdadır. Komşuluk fonksiyonunu modellemek için denklem (4.34)' deki gibi bir Gauss fonksiyonu kullanılabilir. Şekil 4.18' de SOM ağında kazanan birim üzerine konumlandırılmış Gauss fonksiyonun grafiği bulunmaktadır.

$$h_{ci}(t) = \exp\left(-\frac{\|r_c - r_i\|^2}{2\sigma^2(t)}\right) \tag{4.32}$$



Şekil 4.18 Gauss fonksiyonu grafiği (Alpdoğan, 2007)

Denklem (4.32)' de r , i. biriminin ızgaradaki yerini gösteren 2 boyutlu bir vektördür. Eşitlikteki $\|r_c - r_i\|$ ise aktif eğitim iterasyonundaki kazanan nöron c ile çıkış uzayındaki i birimi arasındaki uzaklığı göstermektedir. Etkileşimin uzaysal genişliği, zamanla değişen σ parametresi ile belirlenir. σ komşuluk alanın genişliğini göstermektedir. $h_{ci}(t)$ komşuluk fonksiyonun Gauss fonksiyonu olarak seçilmesiyle kazanan nöron ve çevresindeki komşularının ağırlığının güncellenmesi kazanan nöron için en fazla ve nörondan uzaklaştıkça daha az olmaktadır.

Ağırlık vektörlerinin hareketiyle giriş deseni ve ağırlık vektörü arasındaki Öklid uzaklığı sürekli azalır ve sonuçta ağırlık vektörleri giriş desenine çok benzer hale gelir. Böylece ilgili birimin sonraki iterasyonlarda kazanma olasılığı artmaktadır. Sadece kazanan birimin değil bu birime komsu diğer birimlerin de kazananla birlikte etkileşime dahil edilmesi neticesinde birbirine benzer desenlerin uzaysal kümelenmesi sağlanmaktadır. Böylece n boyutlu bir giriş uzayında bulunan giriş desenlerinden benzer olanları kendinden düzenlenen haritalar ile 2 boyutlu çıkış uzayında komsu olmaktadır. Çıkış uzayında benzer olan desenlerin coğrafik olarak birbirine yakın olacak şekilde kümelenmesi kendinden düzenlenen haritaların eğitim süreci ile sağlanmış olmaktadır (Alpdoğan, 2007).

Döngünün belirli periyotlarında α ve R değerleri azaltılır. Kaç döngüde bir bu değişkenlerin azaltılacağı kesin kurallara bağlanmamıştır. Bu konuda değişik görüşler vardır. Çoğu zaman bu değişkenlerin doğrusal bir fonksiyonla azaltılması yeterli olur. Referans vektörlerindeki değişim sona erdiğinde döngü de sona ermiş olur.

4.3.2.2.2 SOM Modelinde Kümelemeyi Etkileyen Faktörler

“Ağ yapısı ve öğrenme algoritması yukarıda açıklanan bir SOM modelinde başarılı bir kümeleme çalışması gerçekleştirebilmek için bazı faktörlere dikkat etmek gerekir. Ancak, bunlar kesin kurallara bağlı olmayıp sadece bir çerçeve çizmek için verilecektir. Çoğu kez deneme yanılma yoluyla bu faktörler için en iyi değerler bulunur. Giriş vektörüyle referans vektörleri arasındaki fark hata olarak kabul edilirse en küçük mutlak hata ortalamasına sahip model en iyidir denilebilir (Zontul vd., 2004).” SOM modelinde başarılı bir kümeleme çalışması yapabilmek için dikkat edilmesi gereken faktörler şunlardır:

1) Çıkış katmanındaki nöron sayısı:

“Çıkış katmanındaki nöron sayısı, elde edilebilecek maksimum küme sayısını belirtir. Genellikle, veri setindeki eleman sayısının %10’u civarında çıkış nöronu tercih edilir (Zontul vd., 2004).”

2) Verilerin normalleştirilmesi:

Veri setindeki değişken değerleri arasında büyük farklar varsa ve farklı ölçeklerde ölçülmüşlerse değerler normalleştirilmelidir. Böylece, veri setindeki değişken değerleri belli aralıkta ölçeklendirilmiş ve çok büyük ve çok küçük değerlerin etkisi ortadan kaldırılmış olur (Öztemel, 2006 ve Zontul vd., 2004).

3) Referans vektörlerine ilk değer atanması:

"Referans vektörlerine ilk değer atanması, SOM modelinde çok kritik bir yere sahiptir. Bu ilk değerler atanırken veri setindeki giriş vektör değerleri göz önünde bulundurulmalıdır. Pratikte, referans vektörlerine rastgele değerler atamak bazen sakıncalı olabilir. Tüm vektörlere 0'dan çok az büyük bir değer atanırsa öğrenme katsayısı 1'e yakın bir değerle başlatılmalı ve belli bir döngü boyunca (1000 döngü gibi) azaltılmamalıdır. Ayrıca, komşuluk değişkeni büyük bir değerle başlatılmalı ve öğrenme katsayısının değişmediği periyotta sabit kalmalıdır. Böylece, referans vektörleri giriş vektörlerine uygun bir forma kavuşurlar. Referans vektörlerine, giriş vektörlerinin dağılımına uygun bir ilk atama yapıldığında öğrenme katsayısı ve komşuluk değişkeni daha küçük bir değerle başlatılabilir. Bu da algoritmanın öğrenme hızını artırır (Zontul vd., 2004)."

4) Uzaklık ölçüsü:

SOM algoritmasında giriş vektörleriyle referans vektörleri arasındaki uzaklık öklid uzaklığının karesi ile ifade edilmektedir. Ancak, Öklid uzaklığından başka Minkowski, Manhattan gibi uzaklık ölçüleri de kullanılabilir (Zontul vd., 2004).

5) Öğrenme katsayısı ve komşuluk parametreleri:

Genel olarak öğrenme katsayısı ve komşuluk değişkenleri lineer olarak azalan fonksiyonlarla temsil edilirler.

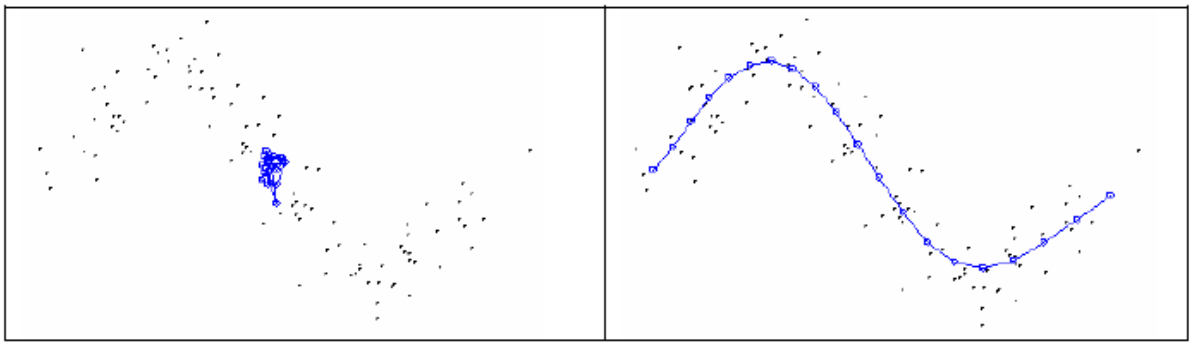
Öğrenme katsayısı 0 ile 1 arasında bir değerle başlamalı ve döngü arttıkça 0'a yaklaşmalıdır. Öğrenme katsayısını belirlemede $\alpha(t) = \alpha_0\left(1 + \frac{100t}{T}\right)$ gibi bir fonksiyon kullanılabilir. Burada t iterasyonu, α_0 başlangıç öğrenme katsayısını ve T toplam döngü sayısını simgelemektedir. Komşuluk parametresini belirleme de $h_{ci}(t) = \exp\left(-\frac{\|r_c - r_i\|^2}{2\sigma^2(t)}\right)$ gibi bir fonksiyon kullanılabilir (Öztemel, 2006).

4.3.2.2.3 Kendinden Düzenlenen Haritalarla K-Ortalamalar Kümeleme Yöntemi Arasındaki Farklar

K-ortalamalar kümeleme algoritmasında merkez noktalar arasında herhangi bir ilişki yoktur. K-ortalamalar kümeleme algoritmasında sadece kazanan merkez güncellenirken SOM' da bütün merkezler kazanan nörona komşuluklarına göre güncellenirler. Yakın komşular uzak

komşulara göre daha fazla hareket ederler. Merkezlerin birbirlerine bağlı oluşu verinin 2 boyutlu uzayda yakınsamasının da elde edilmesini sağlar (Amasyalı, 2006).

Şekil 4.19' da SOM merkezleri gösterilmektedir. Merkezlerin başlangıçtaki durumları rastgele atandığı için bir yuma şeklindedirler. Eğitim tamamlandığında ise SOM merkezleri verinin şeklini almıştır.



Şekil 4.19 SOM ile veri kümeleme (Amasyalı, 2006)

4.4 Küme Sonuçlarının Değerlendirilmesi

Kümeleme analizlerinde, sonuç kümelemelerinin değerlendirilmesi kümeleme modeli geliştirme işleminin ayrılmaz bir parçasıdır. Çünkü bir veri kümesinde küme yapısı olmasa bile kümeleme algoritmaları bu veri seti içersinde istenilen sayıda küme bulacaktır. Ancak elimizdeki veri kümesinde herhangi bir küme yapısı bulunmayabilir. Bundan dolayı kümeleme algoritmalarının sonuçlarının değerlendirilmesine yönelik çeşitli küme doğruluk (cluster validity) yöntemleri geliştirilmiştir. Bu sayede kümeleme çalışmalarında küme kalitesi ve uygun küme sayısı belirlenerek kümeleme işlemleri başarıyla tamamlanabilir (Toledo, 2005).

Literatürde birçok küme doğruluk (cluster validity) yöntemi bulunmaktadır. Bu çalışmada 6 adet küme doğruluk yöntemi kullanılmıştır. Bunlar Silhouette, Davies-Bouldin, Calinski-Harabasz, Krzanowski ve Lai ve Hartigan küme doğruluk (cluster validity) endeksleridir. Bu endeksler hesaplanırken küme merkezlerine ve birimlerin ait oldukları küme indislerine ihtiyaç duyulmaktadır.

4.4.1 Silhouette Endeksi

Kaufman ve Rousseeuw veri setimizde doğal küme sayılarını doğru tahmin etmek için 1990 yılında Silhouette endeksini bir yöntem olarak öne sürmüşlerdir. i gözlemin, i gözlemini içinde bulunduran kümelerinin birimlerine olan ortalama uzaklığı a_i, i gözlemin, i gözlemini

içinde bulundurmayan herhangi bir c kümesinin birimlerine olan ortalama $\bar{d}(i,c)$ uzaklıklarının minimumu b_i olmak üzere i. gözlem için Silhouette endeksi S aşağıdaki gibi hesaplanır (Martinez ve Martinez, 2005).

$$S_i = \frac{(b_i - a_i)}{\max(a_i, b_i)} \tag{4.33}$$

Bütün gözlemler için Silhouette endeksinin ortalaması aşağıdaki gibi bulunur.

$$\bar{S} = \frac{1}{n}\sum_{i=1}^{n} S_i \tag{4.34}$$

Silhouette endeksi -1 ile 1 arasında değerler alabilir. Eksi değer alması istenmeyen bir durumdur çünkü bu $a_i > b_i$ olduğu durumdur yani küme içindeki noktalara olan ortalama uzaklık farklı bir kümedeki noktaların minimum ortalama uzaklıklarından daha büyüktür. Silhouette endeksinin 1' e yakın pozitif büyük değerler alması, i. gözlemin doğal kümesine diğer kümelere oranla daha yakın kümelendiğini, -1' e yakın küçük değerler alması i. gözleminin iyi kümelenmediğini yani yanlış kümelendiğini gösterir. Silhouette endeksi, a_i' nin 0 olması durumunda maksimum 1 değerini alabilir (Martinez ve Martinez, 2005; Tan vd., 2006).

Silhouette endeksinin ortalamasını kullanarak Kaufman ve Rousseeuw veri setindeki doğal küme sayılarını tahmin etmeye çalışmışlardır. Onlara göre ortalama Silhouette endeksinin 0.5 den büyük olması veri setini ayırdığımız küme sayısının yeterli, 0.2' den küçük olması veri setini ayırdığımız küme sayısının doğal küme sayısını karşılamadığını göstermektedir (Martinez, 2005).

4.4.1.1 Silhouette Grafiği

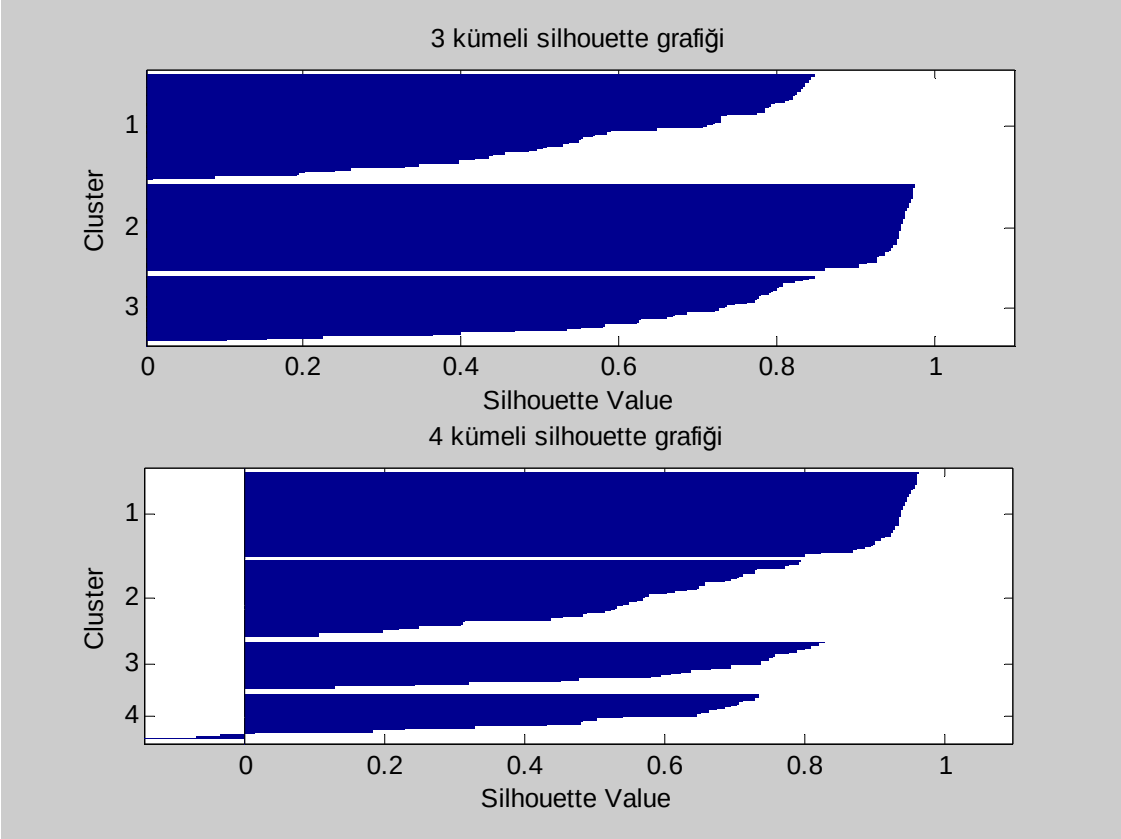
Silhouette endeksinin aldığı değerleri anlaşılır bir şekilde göstermek için Silhouette grafiği geliştirilmiştir. Silhouette grafiğinde, kümelere göre her bir küme biriminin aldığı Silhouette değerleri azalan bir sırada görsel olarak gösterilmiştir. Bu sayede veri setimizde bulunan doğal küme yapılarını hızlı ve doğru bir şekilde tahmin edebiliriz (Martinez, 2005).

Örnek 4.1 :

Süsen veri setini k-ortalamalar kümeleme yöntemiyle 3 ve 4 kümeye ayıralım. Daha sonra Silhoutte endeksi değerleri ve grafiğiyle hangi küme sayısının verimiz için uygun olacağına karar verelim.

```
load fisheriris
```

```
kmus3=kmeans(meas,3)
kmus4=kmeans(means,4);
subplot(2,1,1)
[sil3,h]=silhouette(meas,kmus3);
title('3 kümeli silhouette grafiği')
subplot(2,1,2)
[sil4,h]=silhouette(meas,kmus4);
title('4 kümeli silhouette grafiği')
```



Şekil 4.20 3 ve 4 kümeye ayrılmış süsen verisinin Silhouette grafiği

Şekil 4.20’ de k-ortalamalar kümeleme yöntemiyle elde edilen 3 ve 4 küme için Silhouette grafiği bulunmaktadır. Şekle göre sırasıyla en büyük Sihouette endeksi değerlerini alan birimler 2., 1. ve 3. küme elemanlarıdır. Tüm küme birimlerinin Sihouette endeksleri 1’ e yakın değerler almaktadır. Sihouette grafiğine göre 1’ e en yakın değerler alan küme elemanlarına sahip olan küme en homojen kümedir sonucunu çıkartabiliriz. Şekle göre bu 2. kümedir. Şekil 4.20’ nin alt grafiğine bakacak olursak 4 küme için elde edilen Silhouette değerleri 3 küme için elde edilen Silhouette değerlerine göre küçük çıkmıştır. Hatta Şekil 4.20’ nin alt grafiğine bakacak olursak 4. kümenin bazı değerlerinin negatif Silhouette değerleri aldığı gözlenir; yani 4. kümede kötü kümelenen birimler bulunmaktadır. Sonuç olarak Şekil 4.20’ ye bakarak süsen verisini 3 kümeye ayırmanın daha doğru olacağını söyleyebiliriz.

```
mean(sil3)
ans =
    0.7357              % 3 küme için ortalama Sihouette endeksi
mean(sil4)
ans =
    0.6768              % 4 küme için ortalama Sihouette endeksi
```

Silhouette grafiğini çizmeden de özetleyici bilgiye Silhouette değerlerinin ortalamalarını alarakta ulaşabiliriz. 3 küme için elde edilen Silhouette ortalaması 0.7357, 4 küme için elde edilen Silhouette ortalaması 0.6768' dır. Silhouette değerlerinin ortalamalarına bakarak ta süsen veri setini 4 küme yerine 3 kümeye ayırmanın daha doğru olacağını söyleyebiliriz.

4.4.2 Davies-Bouldin Endeksi

Veri seti n adet kümeye bölündükten sonra $C=\{C_1,C_2,\ldots,C_n\}$ her bir kümenin Davies-Bouldin indeksi bulunur. Denklem (4.35) ve (4.36)' da i. küme için Devies-Bouldin indeksinin nasıl bulunduğu gösterilmiştir

$$DB_{ij}=\frac{\{sc(C_i)+sc(C_j)\}}{cd(C_i,C_j)} \tag{4.35}$$

$$DB_i=\max_{j=1\ldots N,i\neq j}(DB_{ij}) \tag{4.36}$$

Denklem (4.37)' deki $sc(C_i)$ terimi C_i kümesinin içindeki birimlerin küme merkezine olan ortalama uzaklığını, $cd(x,y)$ iki kümenin merkezlerinin birbirlerine olan uzaklığını ifade etmektedir. Her bir küme için indeksler bulunduktan sonra ortalaması alınarak küme algoritmasının kalitesini ifade etmekte kullanılan tüm veriye ait Davies-Bouldin indeksi bulunur. Denklem (4.37)' da tüm veriye ait Davies-Bouldin indeksinin nasıl bulunduğu gösterilmiştir.

$$DB=\frac{1}{n}\sum_{i=1}^{n}DB_i \tag{4.37}$$

Davies-Bouldin indeksi için herhangi bir aralık verilememektedir. Ancak Davies-Bouldin indeksi küme kalitesiyle ters orantılıdır. Davies-Bouldin endeksine göre en uygun küme sayısı D(k) değerini minimum yapan k küme sayısıdır (Amasyalı, 2008).

4.4.3 Dunn Endeksi

Veri seti n adet kümeye bölündükten sonra $C=\{C_1,C_2,\ldots,C_n\}$ her bir kümenin Dunn endeksi denklem (4.38)' de ki gibi hesaplanır:

$$D_n=\min_{i=1,\cdots,n}\left\{\min_{j=i+1,\ldots,n}\left(\frac{d(C_i,C_j)}{\max_{k=1,\cdots,n} diam(C_k)}\right)\right\} \tag{4.38}$$

Denklem (4.38)' de ki C_i ile C_j kümeleri arasındaki uzaklık fonksiyonu denklem (4.39)' da ki gibidir.

$$d(C_i,C_j)=\min_{x\in C_i,\,y\in C_j} d(x,y) \tag{4.39}$$

Bir C kümesinin diameteri denklem (4.40)' da ki gibidir.

$$diam(C)=\max_{x,\,y\in C} d(x,y) \tag{4.40}$$

Dunn endeksi $[0,+\infty)$ açık aralığında değerler alabilir. Kümeleme algoritmaları sonucu X veri seti yoğun ve iyi ayrılmış kümelere ayrılmışsa kümeler arası uzaklıkların büyük ve küme içi diameterin küçük olması beklenir. Dolayısıyla Dunn endeksi büyük pozitif değerler alır. Sonuç olarak Dunn endeksinin mümkün olduğunca büyük pozitif değerler alması istenir (Toledo, 2005).

4.4.4 Calinski ve Harabasz Endeksi

Veri seti n adet kümeye bölündükten sonra $C=\{C_1,C_2,\ldots,C_n\}$ her bir kümenin Calinski ve Harabasz (CH) endeksi aşağıdaki gibi hesaplanır:

$$CH(k)=\frac{B(k)/(k-1)}{W(k)/(n-k)} \tag{4.41}$$

Burada k küme sayısı, $B(k)$ ve $W(k)$ sırasıyla kümeler arası ve küme içi kareler toplamlarıdır. Kümeler arası kareler toplamı her bir küme merkezinin veri setinin bütün küme merkezlerine olan uzaklıklarının kareleri toplamıdır. Küme içi kareler toplamı her bir küme elemanın bulunduğu kümenin merkezine olan uzaklıklarının kareleri toplamıdır. Küme merkezleri olarak küme ortalamaları kullanır. $B(k)$ ve $W(k)$ aşağıdaki gibi hesaplanır.

$$B(k)=\sum_{k=1}^{n} n_k \left\|\bar{x}_{(k)}-\bar{x}\right\|^2 \tag{4.42}$$

$$W(k)=\sum_{k=1}^{n}\sum_{i}^{n_k}\left\|x_{i(k)}-\bar{x}_{(k)}\right\|^2 \tag{4.43}$$

CH endeksine göre en uygun küme sayısı CH(k) değerini en yüksek yapan k küme sayısıdır (Cai ve Lı, 2004).

4.4.5 Krzanowski ve Lai Endeksi

Veri seti n adet kümeye bölündükten sonra $C=\{C_1,C_2,\ldots,C_n\}$ her bir kümenin Krzanowski ve Lai (KL) endeksi aşağıdaki gibi hesaplanır:

$$KL(k)=\left|\frac{DIFF(k)}{DIFF(k+1)}\right| \tag{4.44}$$

Burada p veri setindeki değişken sayısıdır. $DIFF(k)$ denklem (4.45)' deki gibi hesaplanır.

$$DIFF(k)=(k-1)^{2/p}W(k-1)-(k)^{2/p}W(k) \tag{4.45}$$

KL endeksine göre en uygun küme sayısı KL(k) değerini en yüksek yapan k küme sayısıdır. (Mufti vd., 2005)

4.4.6 Hartigan Endeksi

Veri seti n adet kümeye bölündükten sonra $C=\{C_1,C_2,\ldots,C_n\}$ her bir kümenin Hartigan (H) endeksi aşağıdaki gibi hesaplanır:

$$H(k)=(n-k-1)\frac{W(k)-W(k+1)}{W(k+1)} \tag{4.46}$$

Burada $W(k)$ denklem (4.47)' deki gibi hesaplanır.

$$W(k)=\sum_{k=1}^{n}\sum_{i}^{n_k}\left\|x_{i(k)}-\bar{x}_{(k)}\right\|^2 \tag{4.47}$$

Denklem (4.46) deki (n-k-1) düzeltme çarpanı büyük küme sayıları için bir ceza faktörüdür. Hartigan endeksine göre en uygun küme sayısı H(k) değerini en yüksek yapan k küme sayısıdır (Li, 2008).

4.5 Kümeleme Analizi Sonuçlarının Görselleştirilmesi

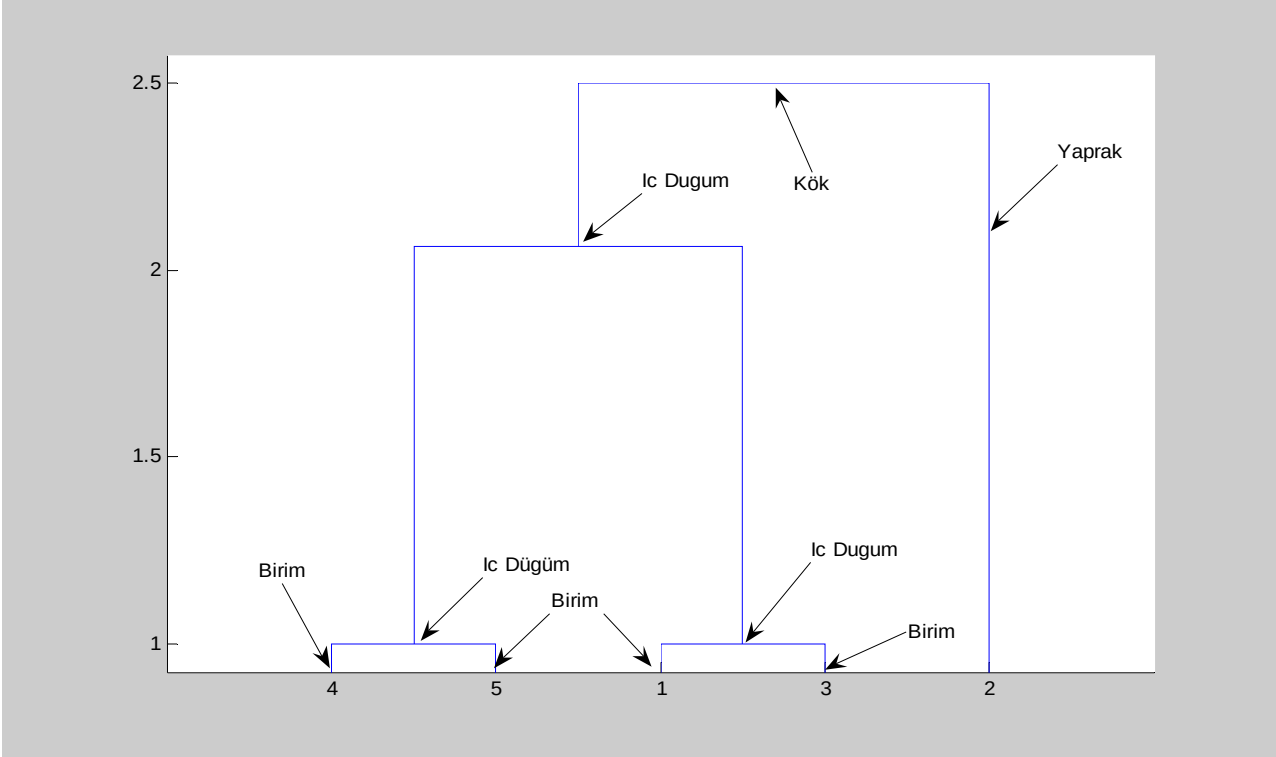
Önceki bölümlerde hiyerarşik, k-ortalamalar ve SOM kümeleme yöntemlerinden bahsedildi. Bu bölümde ise kümeleme analizleri sonuçlarını görselleştirmede kullanılan özel grafiksel

tekniklerden bahsedilecektir. Kümeleme analizleri sonuçlarını görselleştirmede kullanılan teknikler bir bakıma kümeleme analizlerinden elde edilen sonuçların doğruluğunun insan algı sistemiyle test edilmesidir.

4.5.1 Dendrogram

Hiyerarşik kümeleme yöntemleri, veri setinin birimlerinin birbirlerine olan uzaklık değerlerini kullanarak, veri setindeki birimlerin hiyerarşik ayrıştırmasını yapar. Hiyerarşik ayrıştırma sırasında, "ağaç veri yapısı" olarak da bilinen dendrogram kullanılır. Dendrogram, hiyerarşik kümeleme tekniğiyle elde edilen kümelerin görselleştirilmesini sağlar. Bir dendrogramın yapısı kökler (root), iç düğümler (internal node), yapraklar (leaf) ve birimlerden (terminal node) oluşur.

Dendrogram kökü, tüm birimlerin bir araya gelmesiyle oluşan anaküme, dendrogram yaprakları, bir araya getirilmeyen tek birimden oluşan küme, dendrogram iç düğümü, birimlerin bir araya gelerek oluşturdukları kümedir. Şekil 4.21' de 5 birimden oluşan bir verinin tek bağlantılı birleştirici hiyerarşik kümeleme yöntemiyle çizilen dendrogram örneği bulunmaktadır.



Şekil 4.21 Dendrogram

4.5.2 Ağaç Haritaları (Treemaps)

Hiyerarşik kümeleme yöntemleri, veri setinin birimlerinin birbirlerine olan uzaklık değerlerini kullanarak, veri setindeki birimlerin hiyerarşik ayrıştırmasını yaptığını önceki bölümlerde söylemiştik. Hiyerarşik kümeleme tekniğiyle elde edilen kümelerin görselleştirilmesi için ağaç yapılarına benzeyen dendrogramlar kullanılmaktadır. Dendrogramlar sayesinde karışık veriler içinde yatan küme yapıları dallarla ve yapraklarla kolayca anlaşılmaktadır.

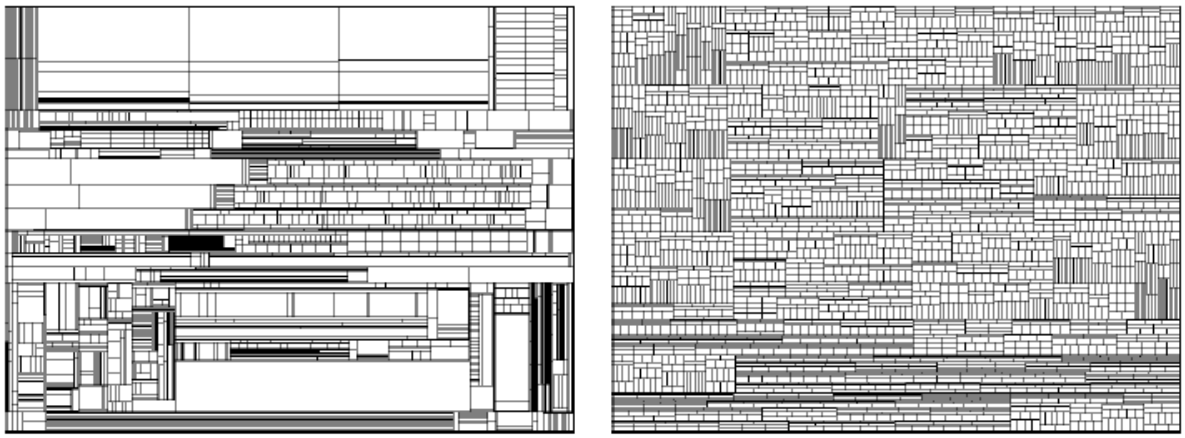
Johnson ve Shneiderman 1991 yılında dendrogramların büyük miktarda, karışık veriler içinde bulunan küme yapılarını göstermede yetersiz olacağını işaret etmişlerdir. Çünkü veriler için çizilen dendrogram grafiğinde dal ve yaprakları gösteren çizgilerden çok boşluklar bulunmaktadır. Bu da verileri göstermek için kullanılan dendrogram grafiklerinin mevcut alanı verimsiz bir şekilde kullandığı anlamına gelmektedir. Ayrıca büyük veriler için çizilen dendrogramlar da yaprak ve dallarının iç içe geçmesinden dolayı birimlerin hiyerarşik yapılarını algılamak pekte kolay olmayabilir. Bundan dolayı Johnson ve Shneiderman dendrogram grafiğinin yetersizliğini ortadan kaldırmak için verilerin hiyerarşik yapılarını göstermede kullanılan ağaç haritalarını (treemaps) geliştirmişlerdir. Ağaç haritalarında, dendrogramlarda bulunan her iç düğüm, aynı karakteristiklere sahip birimlere uygun dikdörtgen şekillerle gösterilmektedir (Martinez ve Martinez 2005).

Ağaç haritalarının asıl kullanım amaçları bilgisayarların disket sürücülerinde ki (hard drives) dosya yapılarını, göstermektir. Hatta ağaç haritaları, üniversitedeki departman yapıları gibi organizasyonları göstermek için de kullanılabilirler. Stok portföyü, tenis maçları ve fotoğraf koleksiyonları gibi geniş alanlardaki hiyerarşik yapıları göstermek için de ağaç haritaları kullanılabilir. Ağaç haritalarının hiyerarşik yapıları göstermede kullanılmasının elverişliğinden dolayı hiyerarşik kümeleme sonucunda elde edilen küme veya iç düğümlerdeki kollar ağaç haritalarıyla görselleştirilebilirler (Martinez ve Martinez, 2005).

Ağaç haritaları, hiyerarşik bilgi ve ilişkileri bir dizi içe içe geçmiş dikdörtgenlerle gösterirler. Ebeveyn dikdörtgen veya dendrogram kökü, ağaç haritalarında karşımızda duran büyük dikdörtgensel görüntüdür. Ağaç haritaları, ebeveyn dikdörtgenin tekrarlamalı olarak alt dikdörtgenlere bölünmesiyle oluşurlar. Ağaç haritalarında alt dikdörtgensel bölgelerin büyüklüğü dendrogramlardaki veya ağaçlardaki düğümlerin büyüklüğüyle orantılıdır. Bir organizasyon birimindeki çalışan sayısı, dosyaların bayt büyüklüğü gibi büyüklükler ağaç yapılarındaki dikdörtgensel büyüklüklerle temsil edilebilirler. Ağaç haritaları kümeleme amaçlı kullanılacaksa eğer, dikdörtgenlerin büyüklüğü kümeler içersinde bulunan gözlem sayılarına bağlıdır. Ağaç haritaları algoritmasında alt dikdörtgenler yatay, dikey şeklinde,

kümeleme işlemimizde herhangi bir küme veya iç düğüm kalmayıncaya kadar, çizilirler (Martinez ve Martinez, 2005).

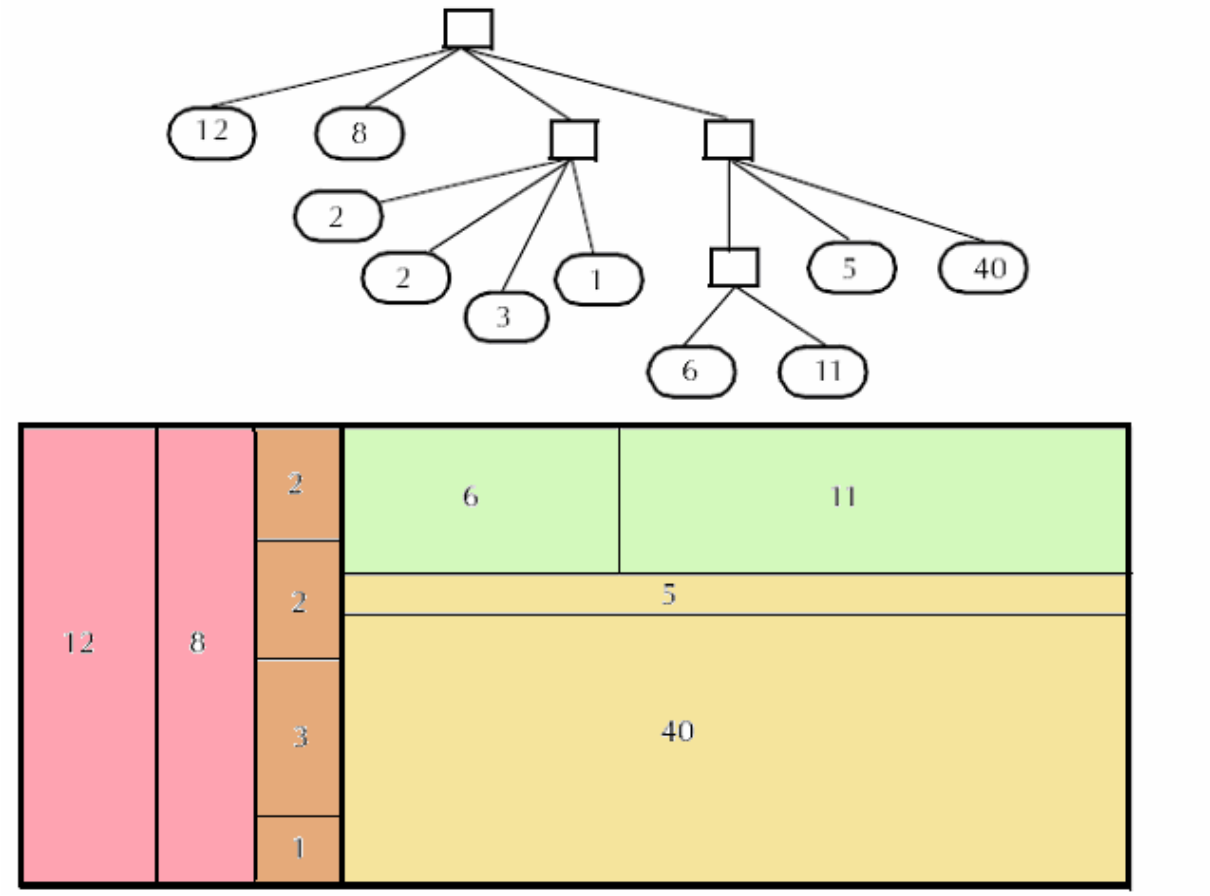
Eğer büyüklüklerin gösterimi çok önemli bir özellikse, ağaç haritalarının kullanılması çok kullanışlı olabilmektedir. Aşağıdaki Şekil 4.22 (a) ‘ da 1400 dosyadan oluşan bir dosya sisteminin ağaç hartası bulunmaktadır. Şekil 4.22 (a)’ ya bakarak hangi dosyanın büyük olduğu kolayca anlaşılabilmektedir.



Şekil 4.22 Ağaç haritaları (Wijk, ve Wetering, 1999)

Ancak ağaç haritaları da bazı sınırlara sahiptir. Hiyerarşik yapılarda düğümlerin büyüklüğü birbirilerinin aynıysa bu durumda ağaç haritalarından hiyerarşik yapıları çıkartmak pekte kolay olmayabilir. Bu gibi durumlarda ağaç yapıları düzenli gridlerden oluşur. Buna örnek olarak, aslında 6 temel gruptan oluşan, Şekil 4.22 (b)’ yi örnek olarak gösterebiliriz. Şekil 4.22 (b) de aynı büyüklüğe sahip 3060 tane dikdörtgen bulunmaktadır. Şekil 4.22 (b) de 6 tane temel grup gözükmemektedir.

Aşağıdaki Şekil 4.23’ de bir ağaç diyagramı için çizilen bir ağaç haritası örneği bulunmaktadır. Burada bulunan ağaç diyagramı hiyerarşik kümeleme yöntemleri için kullanılan dendrogram yapısından biraz farklıdır. Çünkü ağaç diyagramlarında dendrogramlardan farklı olarak her düğüm ikiden fazla düğüme ayrılabilmektedir. Şekil 4.23’ deki ağaç diyagramında kök düğümümüz 4 alt düğüme, aynı zamanda iç düğüme ayrılmaktadır. Bu alt düğümlerden (iç düğümler) ilk ikisi 12 ve 8 büyüklüklerine sahiptir. Diğer iki alt düğüm (iç düğüm) sırasıyla 4 ve 3 alt düğüme sahiptir. Ağaç haritalarını çizerken iç düğümlerimiz dikey dikdörtgenlerle temsil edilirler. İç düğüme bağlı olan yapraklar ise yatay dikdörtgenlerle temsil edilirler.



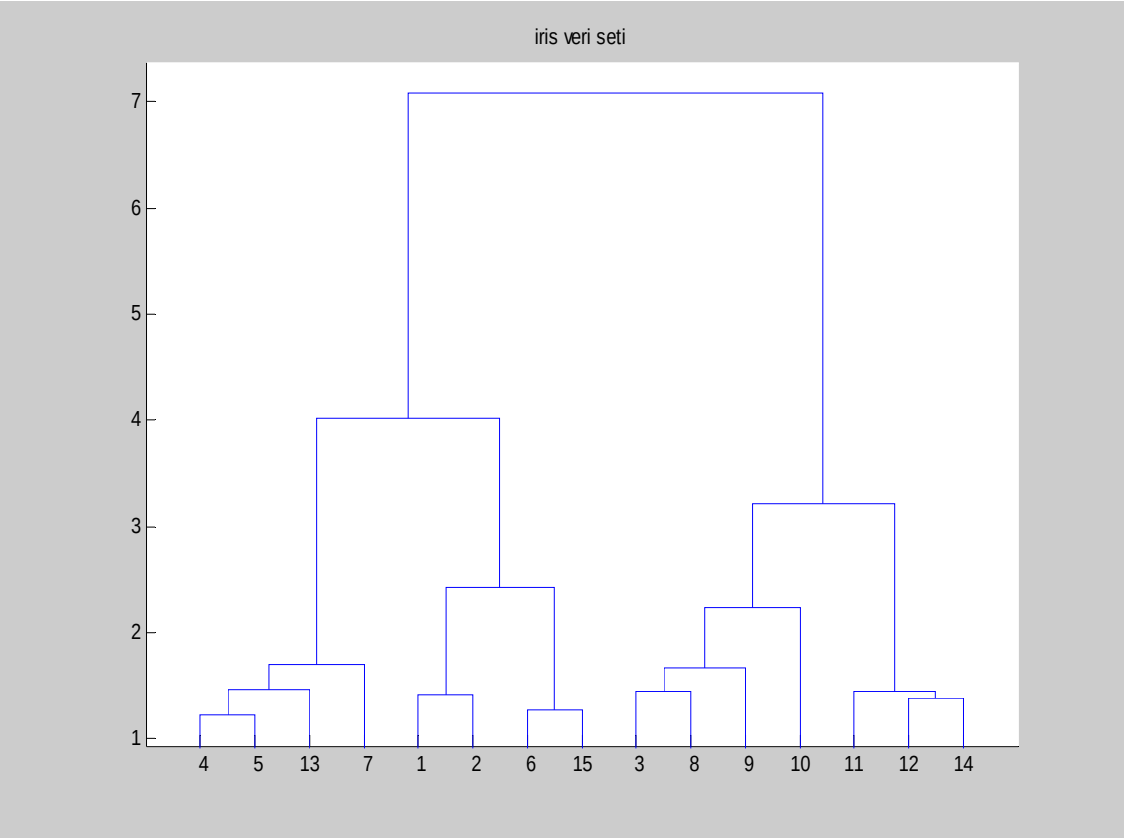
Şekil 4.23 Ağaç haritası (Martinez ve Martinez, 2005)

Hiyerarşik kümeleme sonuçlarını görselleştirmek için ağaç haritalarını kullanacağımız zaman, ayırmak istediğimiz küme sayısının önceden bilinmesi gerekmektedir. Ağaç haritalarında, dendrogramlarda ki gibi, kümelerle birlikte uzaklık veya bezerlik ölçümleri bulunmamaktadır. Hatta dikdörtgenler sadece birim numaralarıyla ya da küme gruplarıyla etiketlendikleri için ağaç hartaları orijinal veri hakkında bilgiye ihtiyaç duyarlar.

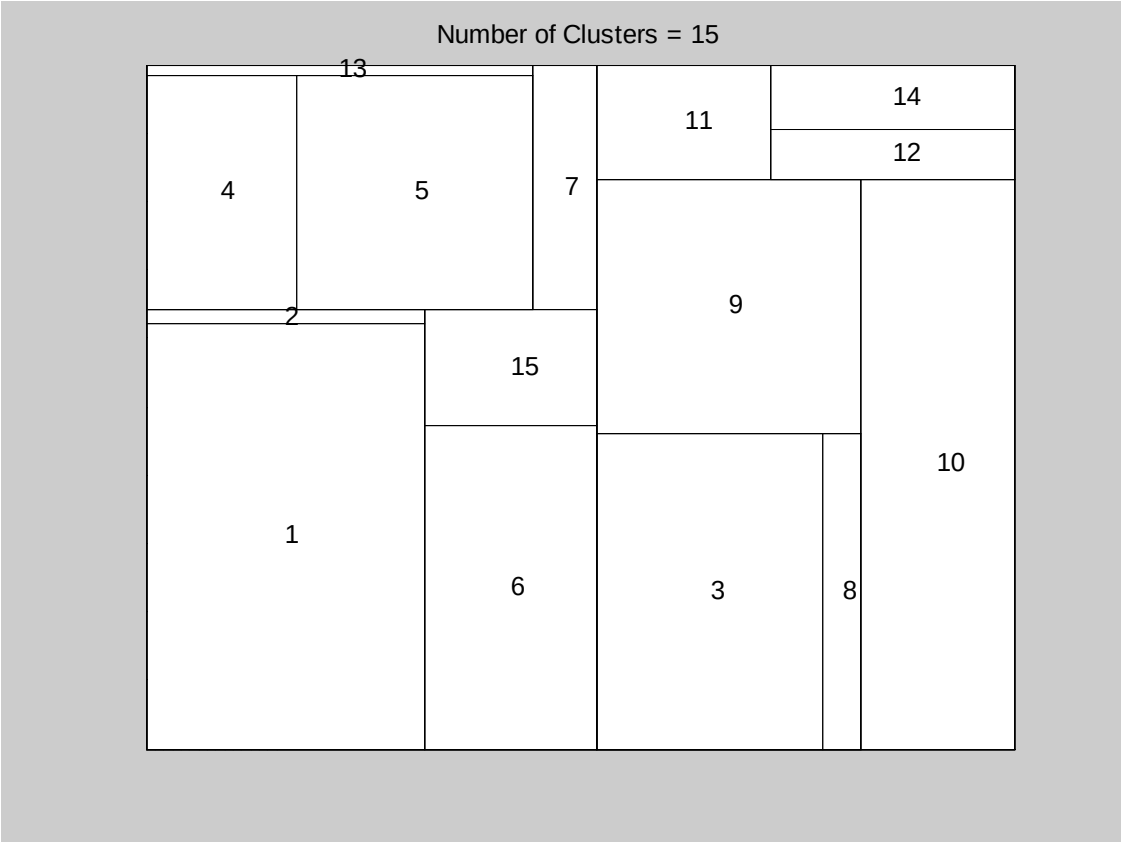
Örnek 4.2 :

Süsen veri setini tam bağlantılı hiyerarşik kümeleme yöntemiyle kümeleyerek kümeleme sonuçlarını dendrogram ve ağaç haritasıyla gösterelim.

```
clear
clc
load fisheriris
[n,p]=size(meas);
y=pdist(meas);
z=linkage(y,'complete');
dendrogram(z,15);
title('iris veri seti')
treemap(z,15)                    % 15 birim için ağaç haritası
```



Şekil 4.24 Süsen veri seti dendrogramı



Şekil 4.25 Süsen veri seti için ağaç haritası

Şekil 4.24’ de süsen veri seti için 15 birimden oluşan tam bağlantılı hiyerarşik kümeleme dendrogamı bulunmaktadır. Şekil 4.25’ de de Şekil 4.24’ deki dendrogram için çizilen ağaç haritası bulunmaktadır. Şeklin sol tarafından açıklamaya başlarsak, şekle göre 1 ile 2 daha sonra 15 ile 6 daha sonra 1, 2 ile 15, 6 birleşmeye başlar. Bu şekilde aynı yüzey sınırına sahip olan kareler birleşerek dendrogramdaki şekil, kare ve dikdörtgenlerden oluşan ağaç haritasında çizilir.

Şekil 4.25 deki ağaç grafiği, dendrogramlardaki gibi düğümlerin birleşme uzaklıkları hakkında bilgi vermemektedir. Ancak, dendrogramlarda ki hiyerarşik düzeni görmemiz için ağaç haritaları kullanışlı olabilmektedir.

4.5.3 Rectangle Grafiği (Rectangle Plot)

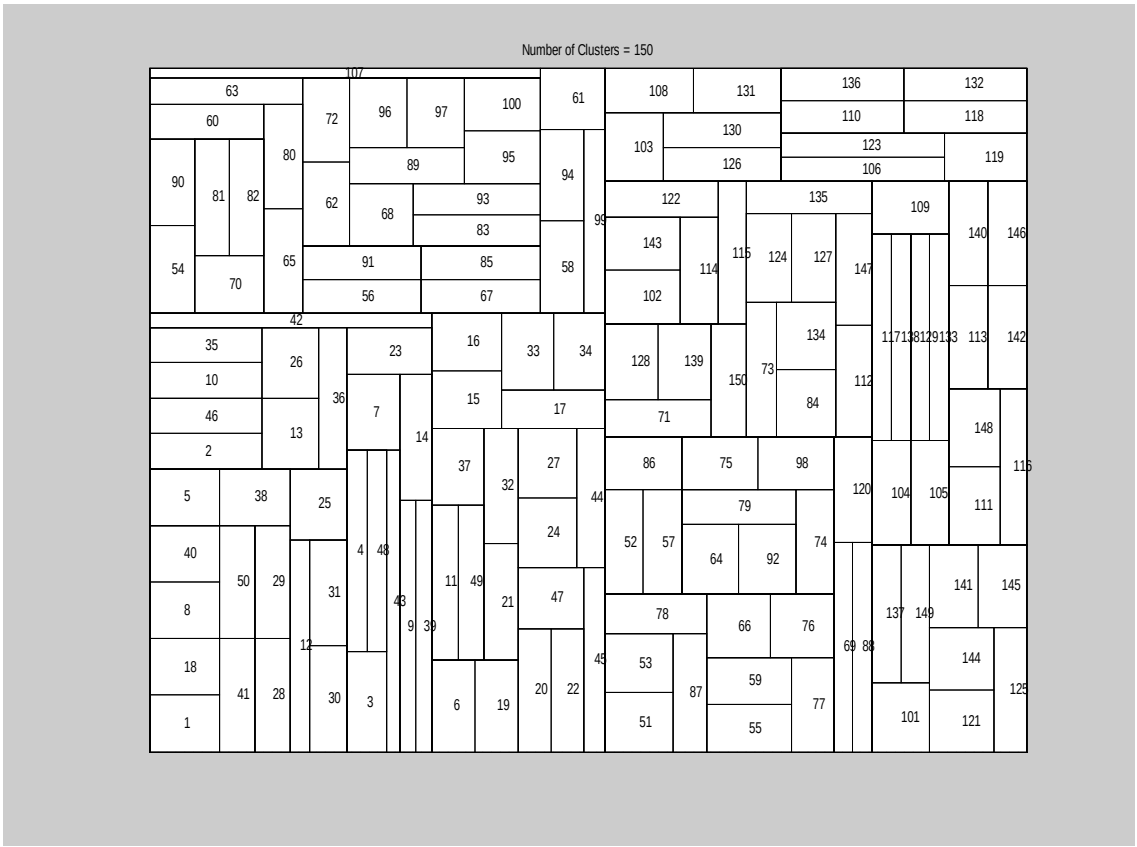
Hiyerarşik kümeleme yöntemiyle tüm gözlemlerin hiyerarşik yapıları dendrogramlar yardımıyla görselleştirilebiliyordu. Büyük miktarda veriler içinse ağaç haritaları kullanarak hiyerarşik yapıların daha kolay anlaşılması sağlanabiliyordu. Ancak, kullanıcı kişi farklı küme yapılarını görmek istediğinde ağaç haritaları kullanışlı olmayabilir. Çünkü ağaç haritalarında birimleri simgeleyen dikdörtgenlerin hangi uzaklıkta birleştiği bilgisi bulunmamaktadır. Bundan dolayı istenilen sayıda kümeyi ağaç haritalarından elde edemeyiz. Ağaç haritalarının bu sorununu gidermek için Wills tarafından 1998 yılında rectangle görselleştirme yöntemi geliştirilmiştir (Martinez ve Martinez, 2005).

Rectangle yöntemi ağaç haritalarına benzemektedir. Rectangle de ağaç haritaları gibi dendrogramın kökünü, yatay dikey olacak şekilde dikdörtgenlere böler. Ancak rectangle grafiği, tüm birimler için dikdörtgenler çizen ağaç haritalarından farklı olarak, istenilen sayıda küme için dendrogramın kökünü dikdörtgenlere böler.

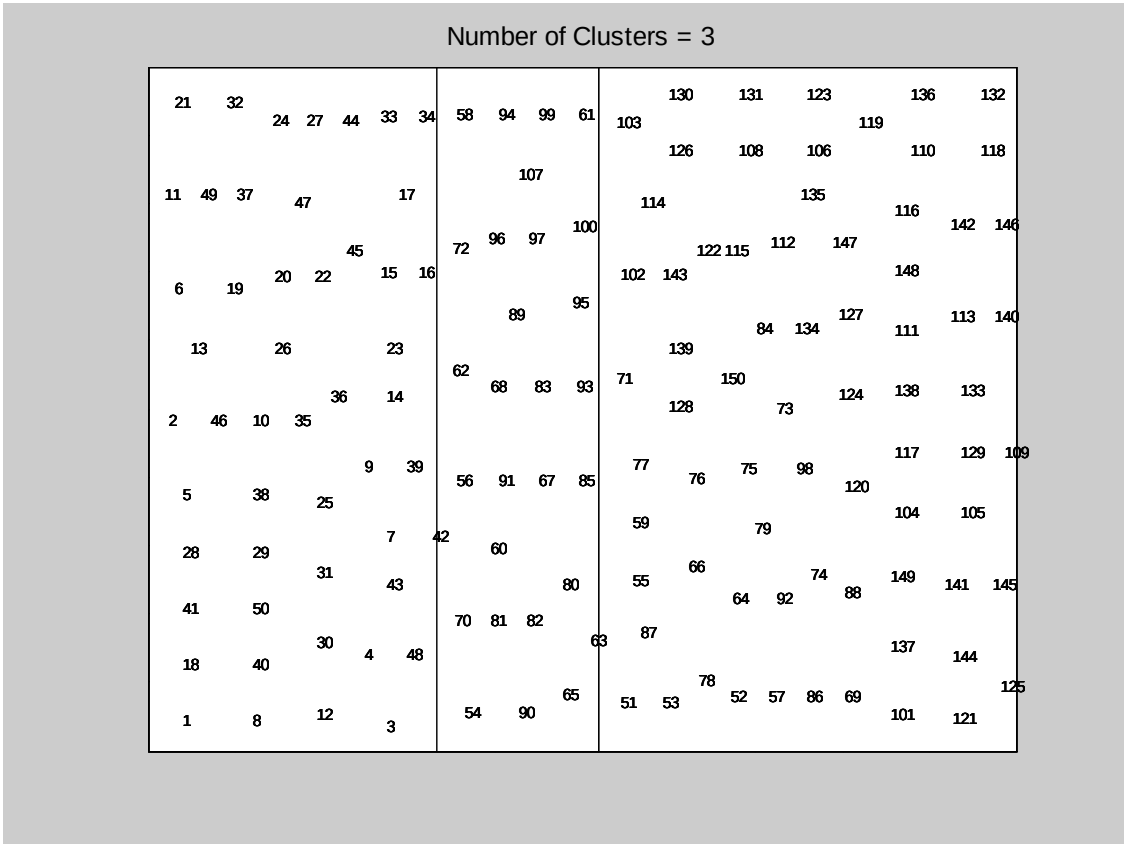
Örnek 4.3 :

Süsen veri setini tam bağlantılı hiyerarşik kümeme yöntemiyle 3 kümeye ayırarak, küme sonuçlarını rectangle grafiğiyle gösterelim. Ayrıca süsen veri setinin 150 gözlemi için de ağaç haritalarını çizelim.

```
load fisheriris
y=pdist(meas);
z=linkage(y,'complete');
treemap(z,150)
rectplot(z,3,'nclus')
```



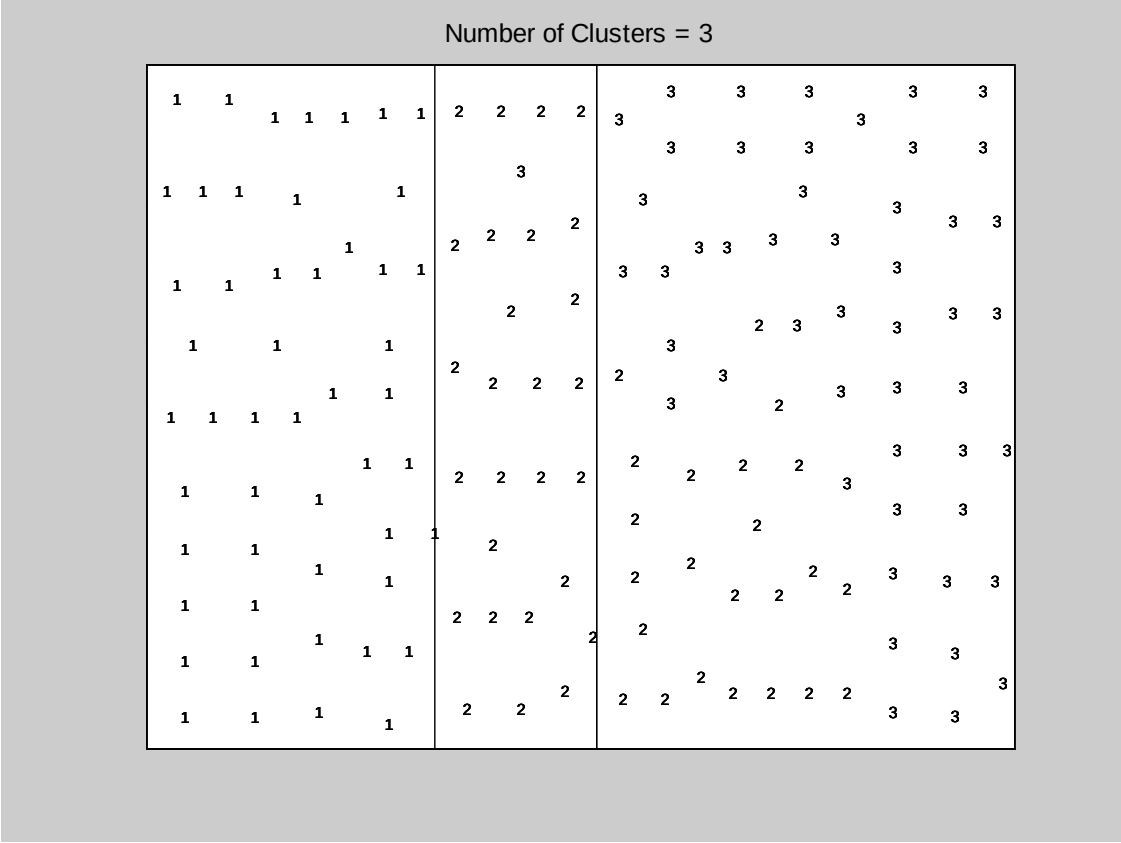
Şekil 4.26 Süsen veri seti için ağaç haritası



Şekil 4.27 Süsen veri seti için rectangle grafiği

Setosa 1, versicolor 2 ve virginica 3 olacak şekilde rectangle grafiğini düzenleyelim.

```
load fisheriris
y=pdist(meas);
z=linkage(y,'complete');
for i=1:150
    if i<=50
        spec(i)=1;
    elseif i>50 & i<=100
            spec(i)=2;
        else
            spec(i)=3;
    end
end
rectplot(z,3,'nclus',spec)
```



Şekil 4.28 Etiketleri doğru rectangle grafiği

Şekil 4.26’ da 150 gözlemden oluşan süsen veri seti için ağaç haritası bulunmaktadır. Şekil 4.26’ da ki ağaç haritasına bakarak, uzaklık değerlerinin yoksunluğundan dolayı, ilk 3 küme elde edilememektedir. Ancak Şekil 4.27’ de ki rectangle grafiğiyle 3 küme elde edildiği gibi hangi kümeye hangi gözlemlerin dahil olduğu da gözlenebilmektedir. Şekil 4.28’ de gözlemlere Setosa için 1, versicolor için 2 ve virginica için 3 sınıf etiketleri verilerek çizilen rectangle grafiği bulunmaktadır. Şekil 4.28’ e bakarak hiyerarşik kümeleme analizi sonucunda elde edilen kümelerin doğru kümelenip kümelenmediği sınıf etiketlerine bakarak anlaşılabilir.

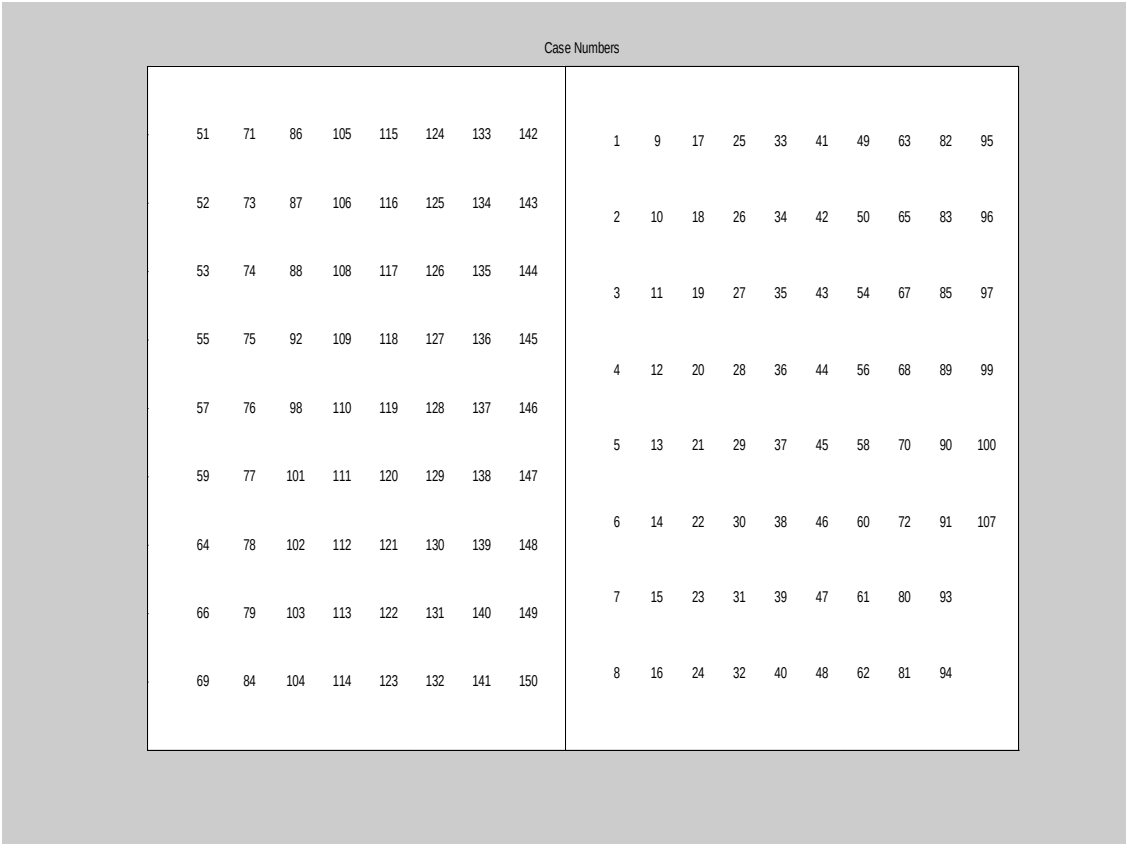
4.5.4 ReClus Grafiği (ReClus Plots)

K-ortalamalar, SOM gibi hiyerarşik olmayan kümeleme yöntemlerinin sonuçlarını görsel olarak göstermek için ReClus grafiği Martinez tarafından geliştirilmiştir. Seçilen küme sayısına göre hiyerarşik kümeleme yöntemlerinin sonuçlarını göstermek için de ReClus grafiği kullanılabilir. ReClus grafiği ağaç ve rectangle yöntemlerine benzemektedir.

Örnek 4.4 :

Süsen veri setini tam bağlantılı hiyerarşik kümeleme yöntemiyle 2 kümeye ayırarak, küme sonuçlarını ReClus grafiğiyle gösterelim. Ayrıca süsen veri setinin gerçek küme sınıfları içinde ReClus grafiğini çizelim.

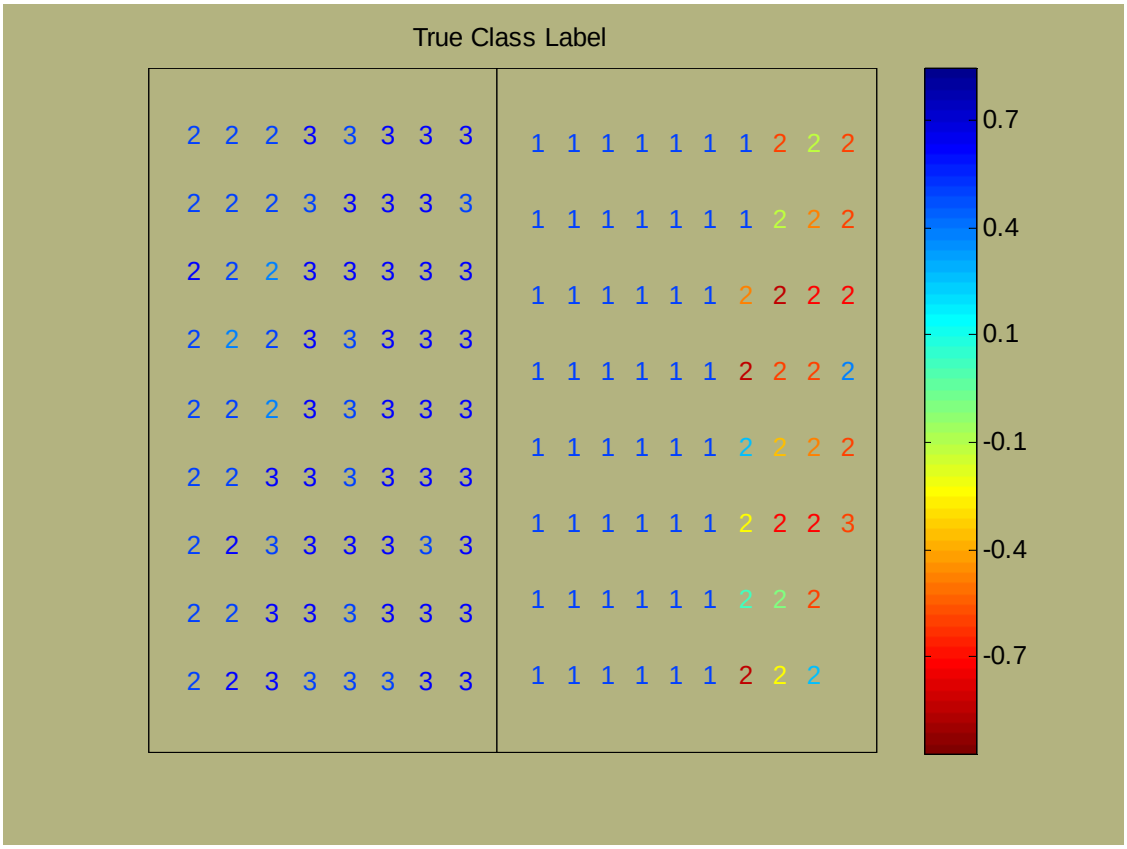
```
load fisheriris
y=pdist(meas);
z=linkage(y,'complete');
cids=cluster(z,'maxclust',2);            % 2 kümeye ayrılır
reclus(cids,1:150)
```



Şekil 4.29 Süsen veri seti için ReClus grafiği

Setosa 1, versicolor 2 ve virginica 3 olacak şekilde ReClus grafiğini düzenleyelim. Ayrıca gözlemlerimizin doğru kümelenip kümelenmediğini anlamak için de Silhouette endeksini kullanalım.

```
load fisheriris
y=pdist(meas);
z=linkage(y,'complete');
cids=cluster(z,'maxclust',2);            % 2 kümeye ayrılır
ind=find(cids==1);                       % 1. küme elemanları seçilir
ind=[ind;find(cids==2)];                 % 2. küme elemanları seçilir
for i=1:150
    if i<=50
        spec(i)=1;
    elseif i>50 & i<=100
            spec(i)=2;
        else
            spec(i)=3;
    end
end
s=silhouette(meas,cids)
reclus(cids,spec',s)
```



Şekil 4.30 Etiketleri doğru ReClus grafiği

Şekil 4.29' da süsen veri seti için tam bağlantılı hiyerarşik kümeleme yöntemiyle elde edilen 2 küme için ReClus grafiği bulunmaktadır. Büyük kare içersindeki dikey çizgi küme sınırını göstermektedir. Şekil 4.30' da ise geçek küme sınıflarına 1, 2, 3 numaraları verilerek elde edilen ReClus grafiği bulunmaktadır. Ayrıca, Şekil 4.30' da her gözlemin Silhouette endeksi değerleri colorbar da yer alan renklerle gösterilmektedir. Buna göre mavi renkler gözlemlerin

doğru kümelendiğini, kırmızı renkler gözlemlerin yanlış kümelendiğini gösterir. Buna göre gözlemlerin renkleri ne kadar koyu maviyse gözlemlerimiz o derece doğru, kırmızı renkler ne kadar koyuysa gözlemlerimiz o derece yanlış kümelenmektedir.

4.5.5 Matris Grafikleri (Matrix Plots)

Matris grafiklerinin değişkenler arasındaki ikili ilişkileri kullanıcıya göstermeye yarayan bir çeşit saçılma çizgisi olduğunu daha önceki bölümlerde değinmiştik. Matris grafiğinin ana fikri, veri matrisinde bulunan verilerin büyüklüğünü matris grafiğinde renkli karelerle temsil ederek görsel bir anlatım sağlamaktır.

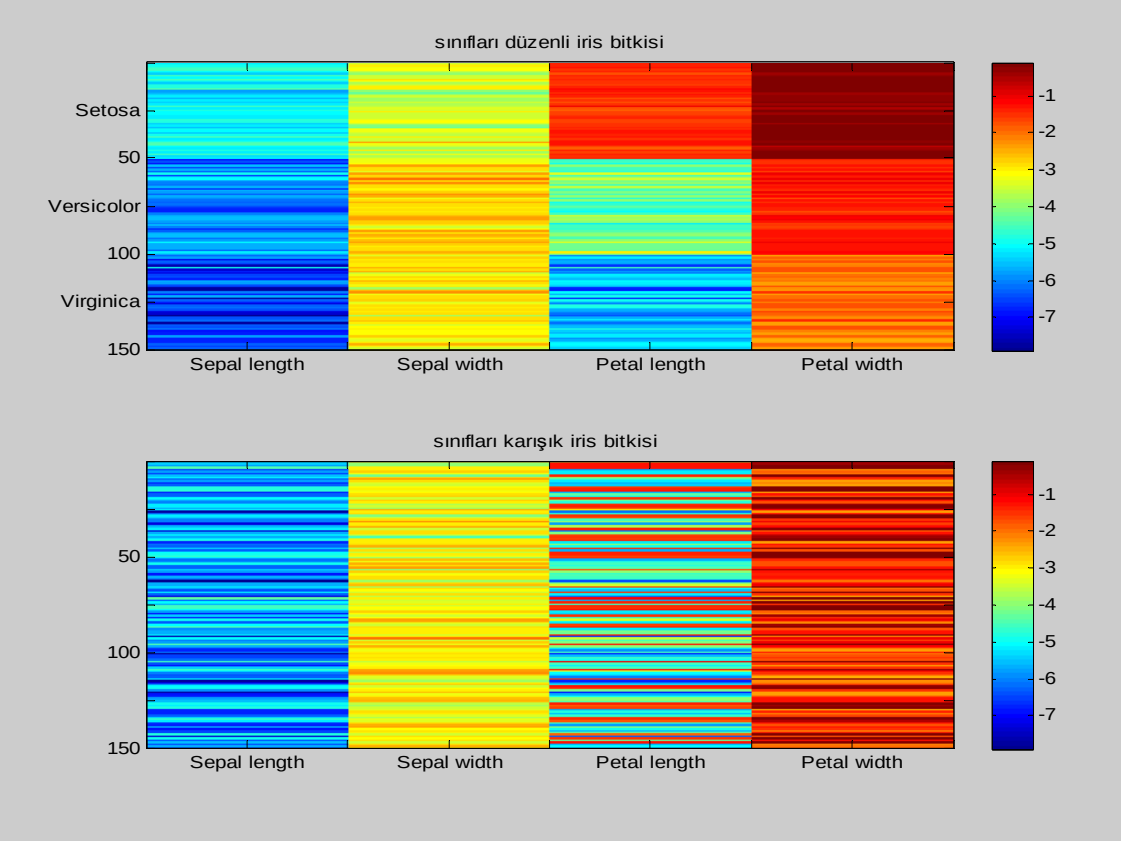
Matris grafiği en basit olarak veri matrisindeki matris değerlerini renkli karelerle göstermek için kullanılmaktadır. Veri matrisindeki matris değerleri arasındaki uzaklık veya benzerlik değerleri kullanarak veriler arasındaki ilişkiler de gözlemlenebilir. Hatta kümeleme analizleri sonucu elde edilen küme sınıflarına göre uzaklıklar veya benzerlikler matrisi düzenlenerek kümeleme analizlerinin sonucu görsel bir şekilde de gözlemlenebilir. Bu yöntem, çok büyük veri setlerinde kullanılsa da veri setindeki küme sayısı hakkında genel bir fikir edinmeyi ve buna uygun iyileştirmeler yapmayı sağlar. Matris grafiği kümeleme analizi sonuçlarını göstermesiyle çok boyutlu ölçekleme analizini anımsatmaktadır (Martinez ve Martinez, 2005).

Yukarıdan da anlaşıldığı üzere matris grafikleri veri madencilerine yüksek boyutlu veri yapılarının anlaşılmasında büyük kolaylıklar sağlayabilmektedir. Tezin bu kısmında matris grafiklerinin kümeleme analizi sonuçlarını görselleştirmede kullanılmasına değinilecektir.

Örnek 4.5 :

Süsen veri setinin sınıfları düzenlenmiş ve de sınıfları karıştırılmış veri değerleri için matris grafiğini çizelim.

```
load fisheriris
data=meas(randperm(150),:);        % veri matrisini karıştıralım
subplot(2,1,1)
imagesc(-1*meas)
title('sınıfları düzenli iris bitkisi')
subplot(2,1,2)
imagesc(-1*data)
title('sınıfları karışık iris bitkisi ')
```



Şekil 4.31 Sınıfları düzenli ve karışık süsen bitkisi

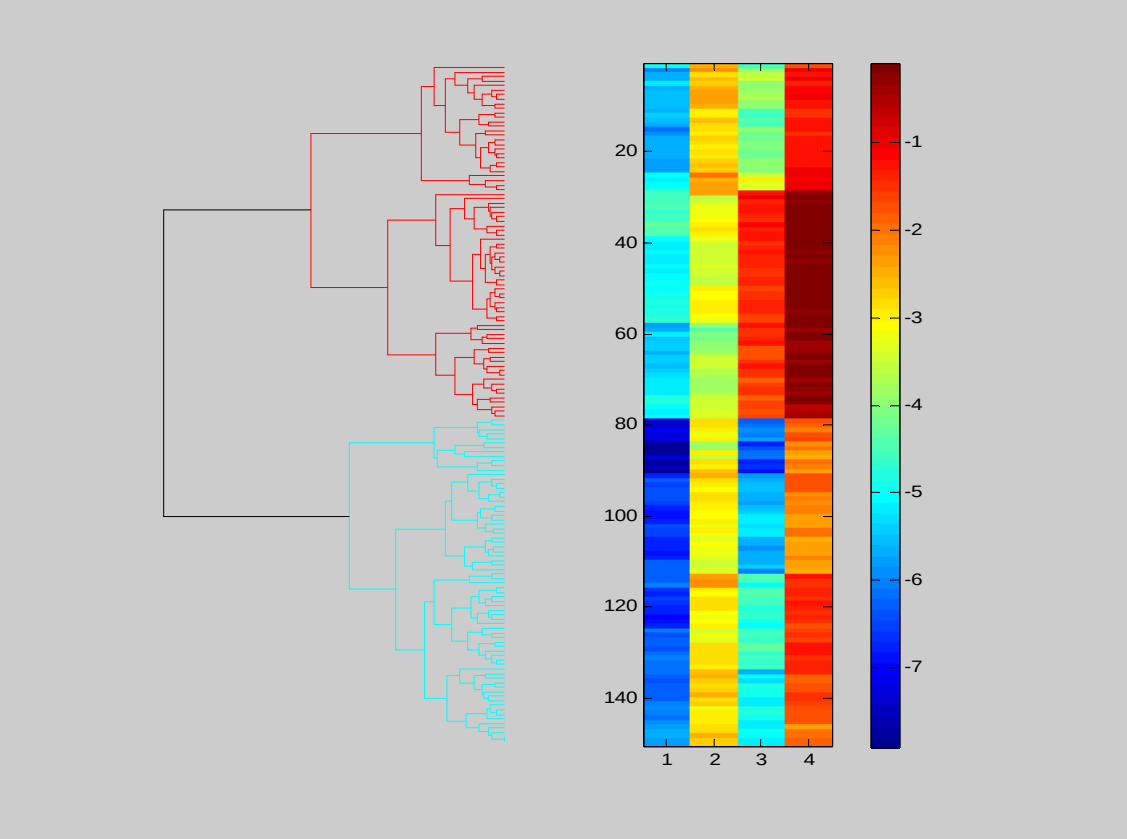
Şekil 4.31' de süsen bitkisi için sınıfları düzenli ve de sınıfları rastgele karışık veri değerleri için matris grafikleri bulunmaktadır. Şekil 4.31' den de gözüktüğü gibi aynı sınıflarda bulunan veriler aynı değişkenler için benzer özellikler, karışık sınıflarda bulunan veriler farklı özellikler sergilemektedir. Şekil 4.31' de sınıfları düzenli süsen bitkisinde küme yapıları saptanabilirken, sınıfları karışık süsen bitkisinde küme yapıları saptanamamaktadır.

Yukarıdaki örnekte meas veri değişkenini -1 ile çarpmamın nedeni aşağıdaki örnek sonuçlarına benzer matris grafiği renkleri elde ermek içindir.

Örnek 4.6 :

Süsen veri seti için tam bağlantılı hiyerarşik kümeleme analizini uygulayarak, küme sonuçlarını matris grafiğinde gösterelim.

```
load fisheriris
data=meas(randperm(150),:);
Y = pdist(data);
Z = linkage(Y,'complete');
subplot(1,2,1)
[H, T, perm] = dendrogram(Z,0,'orientation','left',...
    'colorthreshold','default');
axis off
subplot(1,2,2)
imagesc(flipud(-1*data(perm,:)))    % dendrogram sonuçlarına uygun matris
```



Şekil 4.32 Süsen bitkisi için tam bağlantılı hiyerarşik kümeleme sonuçları

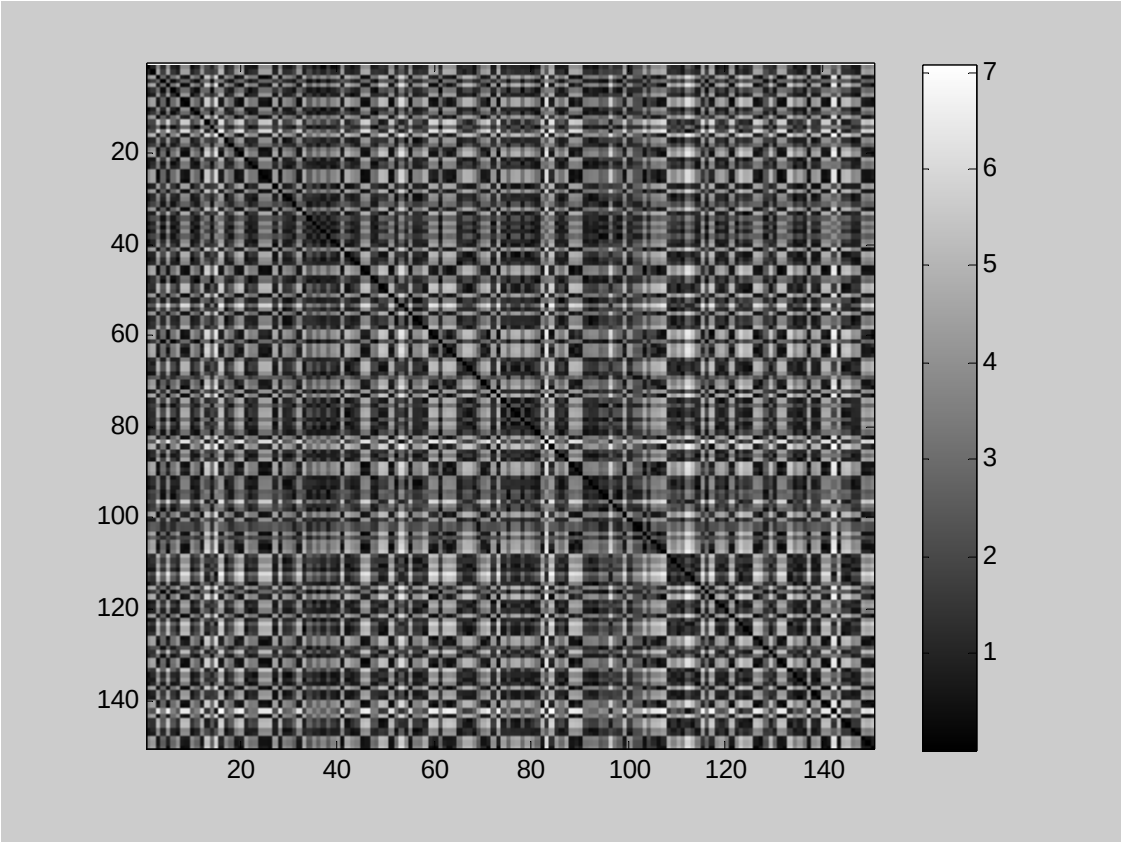
Şekil 4.32 de, hiyerarşik kümeleme analizi sonuçlarına göre veri matrisi düzenlenerek çizilen matris ve dendrogram grafiği bulunmaktadır. Şekil 4.32' de dendrogram grafiğine bağlı olarak 3 küme matris grafiğinde gözükmektedir.

Örnek 4.7 :

Öklid uzaklıklar matrisini kullanarak 3 kümeye ayrılan sınıfları karışık süsen veri seti için tam bağlantılı hiyerarşik ve k-ortalamalar kümeleme sonuçlarını matris grafiğinde gösterelim.

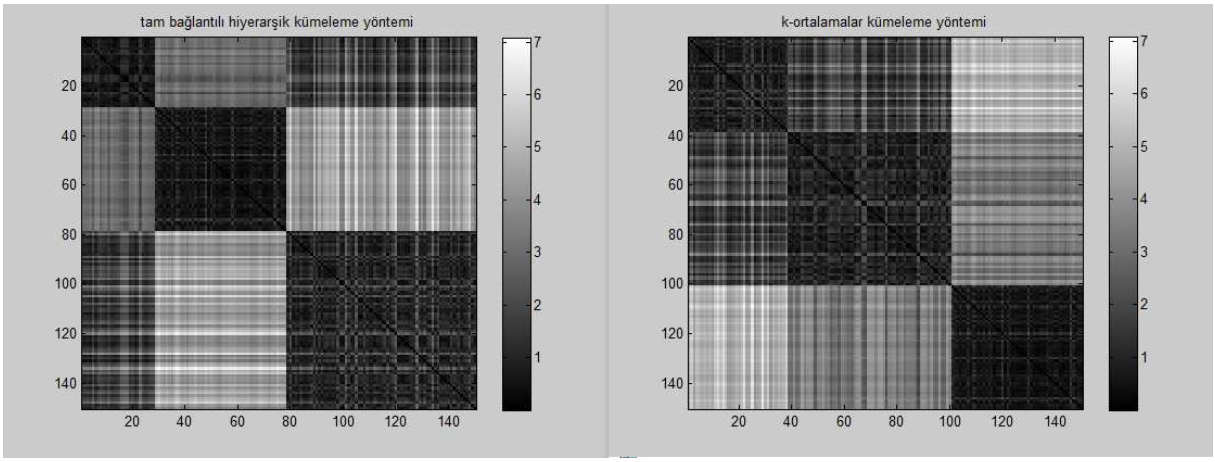
```
load fisheriris
data=meas(randperm(150),:);
Y=pdist(data);
Ys=squareform(Y);
figure
imagesc(Ys);
colormap(gray(256))
Z=linkage(Y,'complete');
T=cluster(Z,'maxclust',3);
[Ts,inds]=sort(T);
figure
imagesc(Ys(inds,inds))
colormap(gray(256))
title('tam bağlantılı hiyerarşik kümeleme yöntemi')
T=kmeans(data,3);
[Ts,inds]=sort(T);
```

```
figure
imagesc(Ys(inds,inds))
colormap(gray(256))
title('k-ortalamalar kümeleme yöntemi')
```



Şekil 4.33 Sınıfları karışık süsen bitkisinin öklid uzaklıkları

Şekil 4.33’ de sınıfları rastgele karıştırılmış süsen bitkisinin öklid uzaklıklar matrisi için çizilen matris grafiği bulunmaktadır. Şekil 4.33’ den de gözüktüğü gibi matris grafiğinde küme yapıları gözükmemektedir.



Şekil 4.34 Hiyerarşik ve k-ortalamalar kümeleme sonuçları

Şekil 4.34’ de tam bağlantılı hiyerarşik ve k-ortalamalar kümeleme algoritmalarına göre 3 kümeye ayrılan süsen bitkisin, küme sınıflarına göre düzenlenmiş öklid uzaklıklar matrisinin matris grafikleri bulunmaktadır. Şekil 4.34’ den de gözüktüğü gibi matris grafiğinde 3 küme simetrik bir şekilde gözükmektedir.

Şekil 4.33’ de öklid uzaklıklar matrisinin kümeleme sonuçları Şekil 4.34’ de ki gibidir.

4.5.6 U matrisi

SOM algoritması başlı başına bir kümeleme algoritması değildir. SOM algoritması yüksek boyutlu verileri keşfetmek ve görselleştirmek için kullanılan bir araçtır. SOM algoritmasıyla boyut indirgemesi yapılarak görselleştirme sağlanmaktadır. Algoritma sayesinde yüksek boyutlu veri içersindeki karışık, lineer olmayan ilişkiler daha basit metrik, topolojik ilişkilere dönüştürür. Bu sayede yüksek boyutlu veri içersindeki benzer gruplar belirlenmekte ve bunlar görselleştirmeyle gözlenmektedir. Bu özeliğiyle SOM, MDS’ ye benzemektedir (Martinez, Martinez 2005).

SOM algoritmasının sonucunu görselleştirmek için çeşitli yöntemler bulunmaktadır. Verinin genel dağılımını ve küme yapılarını göstermesi açısından U matrisi çoğunlukla tercih edilmektedir.

U matrisi, SOM algoritması sonucu güncellenen referans vektörlerinin birbirlerine olan uzaklıklarını kullanarak görselleştirme sağlamaktadır. U matrisinde, nöronun komşu nöronların referans vektörlerine olan ikili uzaklıkları, matris grafiklerinde olduğu gibi, renkli piksellerle gösterilerek bir görselleştirme sağlanmaktadır.

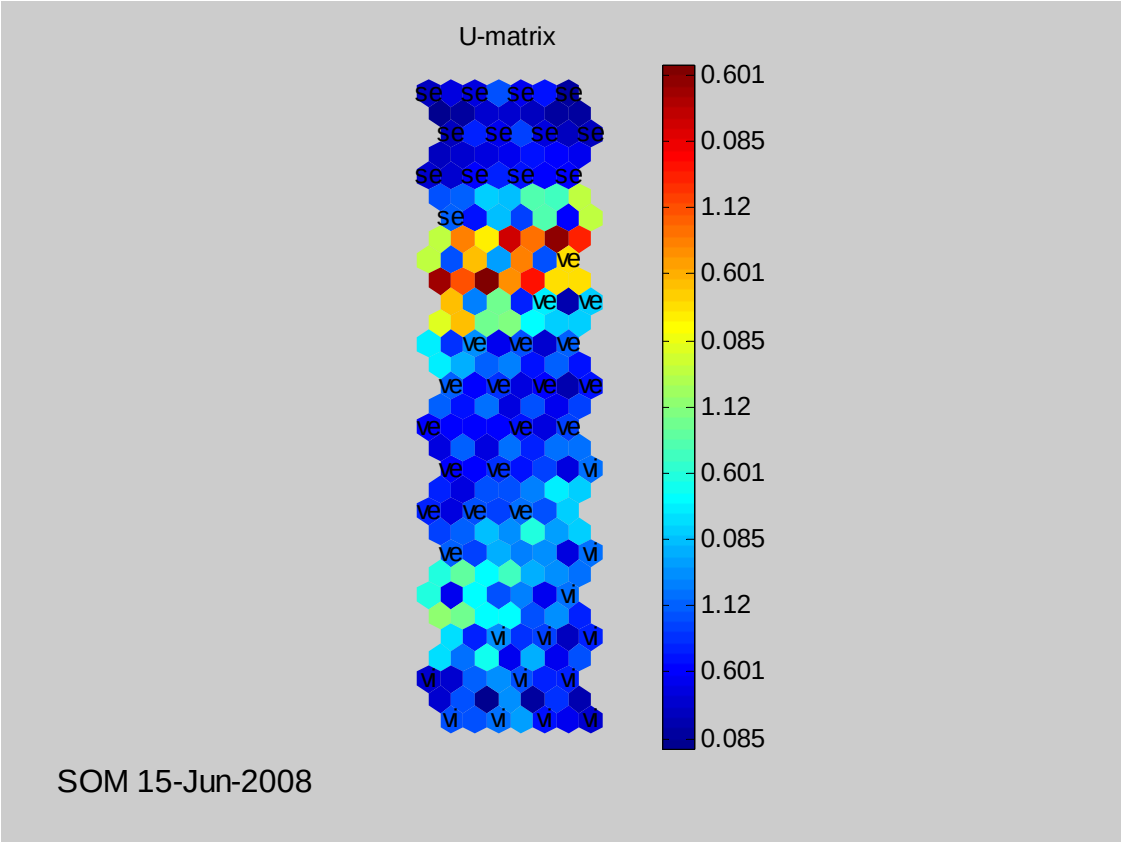
SOM algoritmasında kazanan nöron ve kazanan nöronun etrafındaki nöronlar güncellenerek birbirine benzer nöron yani kümeler elde edilmekteydi. Bu nöronların U matrisiyle görselleştirilmesi sayesinde aynı kümede bulunan nöronların renkleri, düşük sayısal değerlere karşı gelen koyu renklerle göstermektedir. Aynı küme içersinde bulunan nöronların etrafını saran, yüksek sayısal değerlere karşı gelen açık renkler ise kümeleri ayıran küme sınırlarını göstermektedir.

Örnek 4.8 :

Süsen veri setini kullanarak U matrisini çizelim.

```
sD=som_read_data('iris.data');           % iris veri seti okunur
D=som_normalize(sD,'var');               % iris veri seti normalleştirilir.
sM=som_make(sD,['seq']);
for i=1:150
    if i<=50
```

```
        sD.labels{i}='se';
    elseif i>50 & i<=100
        sD.labels{i}='ve';
    else
        sD.labels{i}='vi';
    end
end
sM=som_autolabel(sM,sD,'vote');          % SOM' a etiketler yerleştirilir
som_show(sM,'umat','all');               % SOM grafiği çizilir.
som_show_add('label',sM,'subplot',1);    % SOM' a etiketler yerleştirilir.
```



Şekil 4.35 Süsen veri seti için U matrisi

Şekil 4.35' de süsen veri seti için çizilen U matris grafiği bulunmaktadır. U matrisinden çok açık bir şekilde iki tane küme yapısı gözlenmektedir. U matrisine bitkilerin bulunduğu cinslerin adları eklenirse, setosa bitki cinsinin bir grupta, versicolor ve virginica bitki cinslerinin diğer grupta yer aldığı gözlenmektedir. Şekilden de gözüktüğü gibi U matrisiyle belirgin küme sınırları gözlenmemektedir. Ancak birimlerin bulunduğu cinslerin adlarını U matrisine etiketlemek suretiyle küme yapıları gözlenebilmektedir. Bu sayede versicolor ve virginica bitki cinslerinin de farklı şekilde kümelendikleri gözlenmektedir (Martinez, Martinez 2005).

Teorik olarak SOM algoritmasında ki hücrelerin her birini küme merkezi olarak düşünebiliriz. Kümeler bu küme merkezlerine bağlı olarak bulunmaktadır ve görselleştirilebilmektedir.

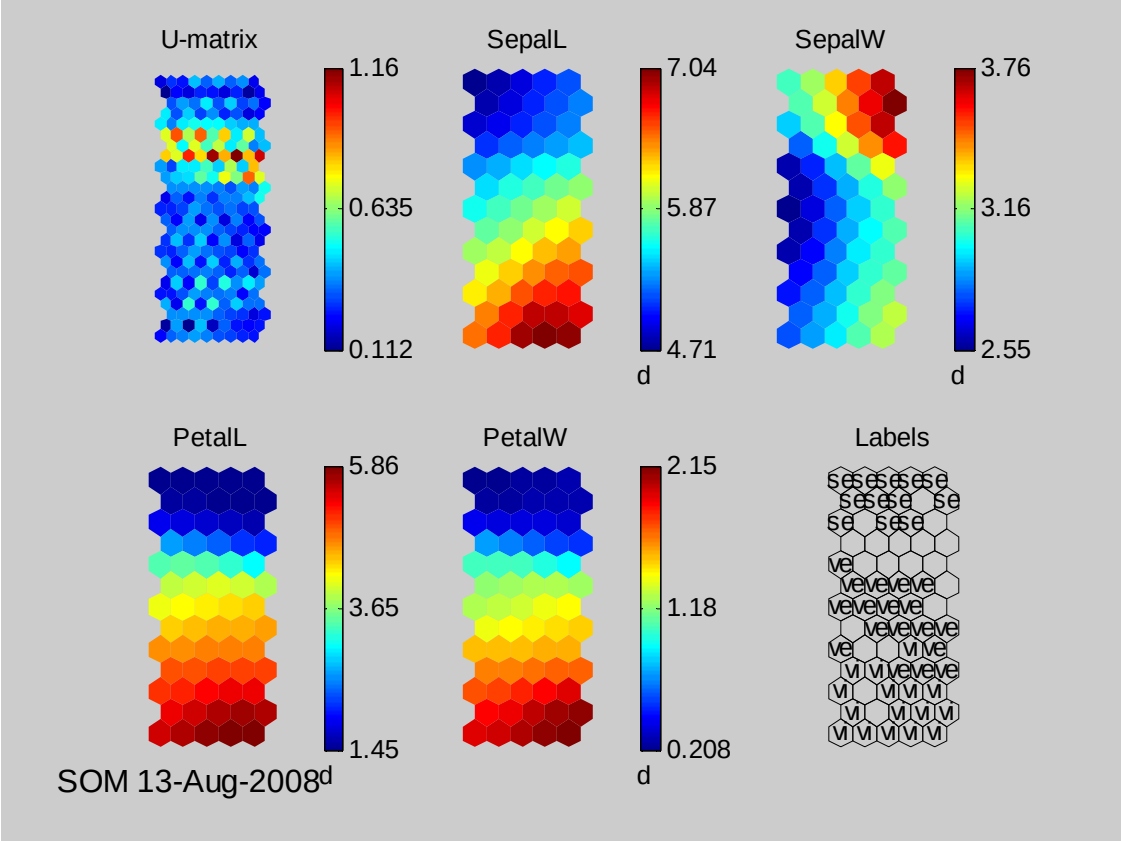
Ancak veri seti hakkında genel bir kanıya varabilmek için çok sayıda nöron dizilimi kullanmamız gerekmektedir. Nöron sayılarının artması SOM algoritması sonucu elde edilen U matrisinden küme yapılarının anlaşılmasını kolaylaştırmakta ve küme yapılarının saklı kalması önlenmektedir (Schatzmann, 2003).

SOM algoritmasıyla veri seti için elde edilen sonuçlar U matrisiyle görselleştirilebilmektedir. Ancak biz her bir değişken için SOM algoritması sonuçlarını da görselleştirebiliriz. Bu görselleştirme U matrisinden biraz farklılık göstermektedir. Her bir değişken için çizilen haritalarda nöronların birbirilerine göre uzaklıkları değil sadece referans vektörleri renkli piksellerle gösterilmektedir. Bu sayede nöronların referans vektörleri görselleştirilerek değişkenlerin dağılımları hakkında bilgiler elde edilebilmektedir (Vesanto ve Himberg, 2000).

Örnek 4.9 :

Süsen veri setini kullanarak U matrisini ve her bir değişken için elde edilen haritaları çizelim.

```
sD=som_read_data('iris.txt');             % iris veri seti okunur
sD=som_normalize(sD,'var');               % iris veri seti normalleştirilir.
sM=som_make(sD,['seq'],'msize',[13 5]);
for i=1:150
    if i<=50
        sD.labels{i}='se';
    elseif i>50 & i<=100
        sD.labels{i}='ve';
    else
        sD.labels{i}='vi';
    end
end
sM=som_autolabel(sM,sD,'vote');           % SOM' a etiketler yerleştirilir
sM.comp_names{1}='SepalL';
sM.comp_names{2}='SepalW';
sM.comp_names{3}='PetalL';
sM.comp_names{4}='PetalW';
som_show(sM,'umat','all','comp',1:4,'empty','Labels','norm','d');
som_show_add('label',sM,'subplot',6);     % grafiklere etiketler yerleştirir
```



Şekil 4.36 Süsen veri seti için SOM haritaları

Şekil 4.36' da süsen veri seti için SOM algoritması sonucu elde edilen haritalar bulunmaktadır. Grafiğin sol üst köşesinde U matrisi, sağ alt köşede etiketlenen harita birimleri, diğer yerlerde her bir değişken için elde edilen SOM haritaları bulunmaktadır. Değişkenler için elde edilen SOM haritalarıyla değişkenlerin dağılımları hakkında bilgiler elde edilebilmektedir. Örneğin, setosa bitki cinsi kısa petale (Petal Length), dar petale (Petal Width), dar sepale (Sepal Width) ve uzun sepale (Sepal Length) sahiptir. Bitki cinslerindeki ayrım yaprakların büyüklüklerinden ileri gelmektedir.

Değişkenlerin haritalanmalarındaki renklere bakılarak değişkenler arasındaki korelasyonlar kestirilebilir. Şekil 4.36' ya göre, birbirine benzer renklenmeler gösteren PetalW ve PetalL ve hatta SepalL ile aralarında güçlü bir korelasyon bulunmaktadır. Diğer değişkenlere göre farklı renklendirilen PetalL haritasında bu değişkenin diğer değişkenlere göre negatif korele olduğu düşünülebilir. SOM haritalarından elde edilen sonuçları ispatlamak için Tablo 4.3' de ki değişkenlerin korelasyonlarına bakılabilir.

Tablo 4.3 Süsen bitki özelliklerine göre korelasyonlar

Korelasyon	SepalL	SepalW	PetalL	PetalW
SepalL	1	-0.1176	0.8718	0.8179
SepalW	-0.1176	1	-0.4284	-0.3661
PetalL	0.8718	-0.4284	1	0.9629
PetalW	0.8179	-0.3661	0.9629	1

5. UYGULAMA

5.1 Açıklama

Sosyo-ekonomik gelişme, gerek zaman, gerek mekân açısından farklılıklar göstermekte ve sürekli değişen bir olgu olarak kabul edilmektedir. Dolayısıyla ülkelerin gelişme çizgileri zamanla değiştiği gibi, mevcut gelişme düzeylerinin yöreler itibariyle dağılımında da farklılıklar gösterdiği bilinmektedir.

Bu bölümde 81 ildeki 918 ilçe 7 coğrafi bölge bazında ele alınarak gelişmişlik düzeylerine göre görsel veri madenciliği teknikleri yardımıyla kümelenecektir.

5.2 Analizde Kullanılan Değişkenler

Ülkemizde, iller ve özellikle ilçeler itibariyle yapılacak ekonomik ve sosyal araştırmalar için ihtiyaç duyulan verilerin yeterli ölçüde ve sistematik bir şekilde temin edilmesinin ortaya koyduğu zorluklar nedeniyle, bu çalışmada kullanılan değişkenler Türkiye deki ilçelerin gelişmişlik düzeyleri belirlenmesi için yayınlanmakta olan ve kolay ulaşılabilen 20 adet değişkenden derlenmiştir. Bu çalışmada kullanılan sosyo-ekonomik nitelikteki değişkenler ve bu değişkenlerin analiz aşamasında kullanılan isimleri aşağıda sıralanmaktadır:

X1 : Toplam nüfusun yıllık ortalama artış hızı (%) (1990-2000)

X2 : Şehirleşeme oranı (%) (2000)

X3 : Toplam nüfus yoğunluğu (kişi/km^2) (2000)

X4 : Ücretli çalışan kadınların toplam istihdama oranı (%) (2000)

X5 : İşsizlik (%) (2000)

X6 : Erkek okuryazar oranı (%) (2000)

X7 : Kadın okuryazar oranı (%) (2000)

X8 : Erkek yüksek okul bitirenlerin oranı (%) (2000)

X9 : Kadın yüksek okul bitirenlerin oranı (%) (2000)

X10 : Tarım kesiminde çalışanların toplam istihdama oranı (%) (2000)

X11 : İmalat sanayinde çalışanların toplam istihdama oranı (%) (2000)

X12 : İnşaat kesiminde çalışanların toplam istihdama oranı (%) (2000)

X13 : Toplam perakende ticarette çalışanların toplam istihdama oranı (%) (2000)

X14 : Ulaştırma depolamada çalışanların toplam istihdama oranı (%) (2000)

X15 : Mali kurumlarda çalışanların toplam istihdama oranı (%) (2000)

X16 : İlmi ve teknik mesleğe sahip olan kişilerin toplam istihdama oranı (%) (2000)

X17 : İşverenlerin toplam istihdama oranı (%) (2000)

X18 : İdari personel ve benzeri çalışanların toplam istihdama oranı (%) 2000

X19 : Fert başına düşen gelir (GSMH TL) (1996)

X20 : 100000 kişiye düşen banka şube sayısı (2000)

Söz konusu istatistiklerin geç yayınlanıyor olması nedeniyle çalışmada kullanılan verilerin güncelliği 2000 yılı ile sınırlı kalmıştır. İlçelere ait veriler TÜİK tarafından yayınlanan 2000 yılı nüfus sayım sonuçlarından alınmıştır.

2000 yılı GSMH değerleri bulunamadığı için çalışmada 1996 yılı GSMH değerleri kullanılmıştır. Bu sebepten dolayı çalışmada, 1996 yılında ilçe olmayan Kocaeli-Derince, Osmaniye-Hasanbeyli, Osmaniye-Sumbas, Osmaniye-Toprakkale, Düzce-Kaynaşlı ilçeleri kapsam dışı bırakılmıştır.

Kümeleme analizi hesaplama çalışmalarında MATLAB R2007a, Orange ve SPSS 15 Evaluation programları kullanılmıştır. Değişkenlerin birimlerinin farklı olması nedeniyle, kümeleme analizi çalışmalarında standart veri matrisi kullanılmıştır. Standartlaştırılmış veri matrisi,

$$Z_i = \frac{orjinal\ veri - verinin\ ortalaması}{verinin\ s\tan dart\ hatası} \tag{5.1}$$

formülünden hesaplanmıştır.

5.3 Aşırı Değer Analizi

Aşırı değerler, veri madenciliği sürecinin analiz aşamasında regresyon, kümeleme analizi gibi uygulamalarda sorunlara neden olurlar. Bu nedenle aşırı değerlerin veri setinden den arındırılması gerekmektedir.

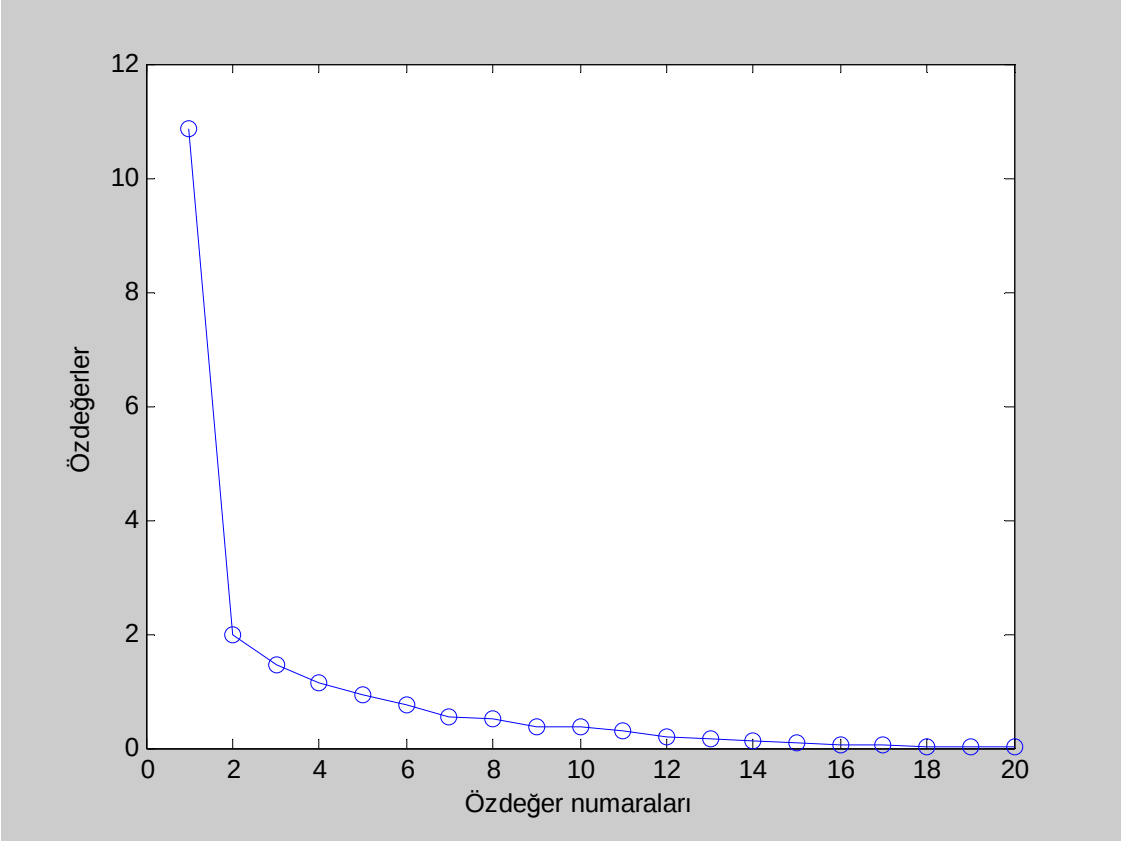
918 ilçenin 20 değişkeninden oluşan veri setinde gerek değişken sayısının fazla olması, gerekse veri birimlerinin fazla olmasından dolayı aşırı değerleri ayıklamak oldukça zor bir iştir. Bunun için görselleştirme teknikleri kullanılarak aşırı değerleri görsel bir şekilde tespit edip, veri setinden ayıklayabiliriz.

Tez çalışmasında, değişken bazında değil de global anlamda aşırı değerleri tespit etme özelliğinden dolayı Andrews eğrileri aşırı değerlerin tespitinde kullanılmıştır. Bunun için temel bileşenler analiziyle boyut indirgemesi yapılmış ve birbirinden bağımsız bileşenler elde edilmiştir. Daha sonra bu bileşenler Andrews eğrilerinde kullanılarak aşırı değerler görsel bir şekilde tespit edilmiştir. Temel bileşenler analiziyle elde edilen bileşenlerin sırası veri setinin toplam değişkenliğini açıklama oranlarıyla orantılıdır. Bu sayede Andrews eğrileri, en büyük açıklayıcılık oranına sahip bileşen en büyük etkiye sahip olacak şekilde çizilir.

918 ilçenin 20 değişkeninden oluşan veri setinin temel bileşenler analizinden elde edilen özdeğerler ve toplam varyans açıklama oranları Tablo 5.1 de verilmiştir. Bileşenlerin toplam varyansı açıklama oranları için çizilen yamaç grafiği ise Şekil 5.1’ de verilmiştir.

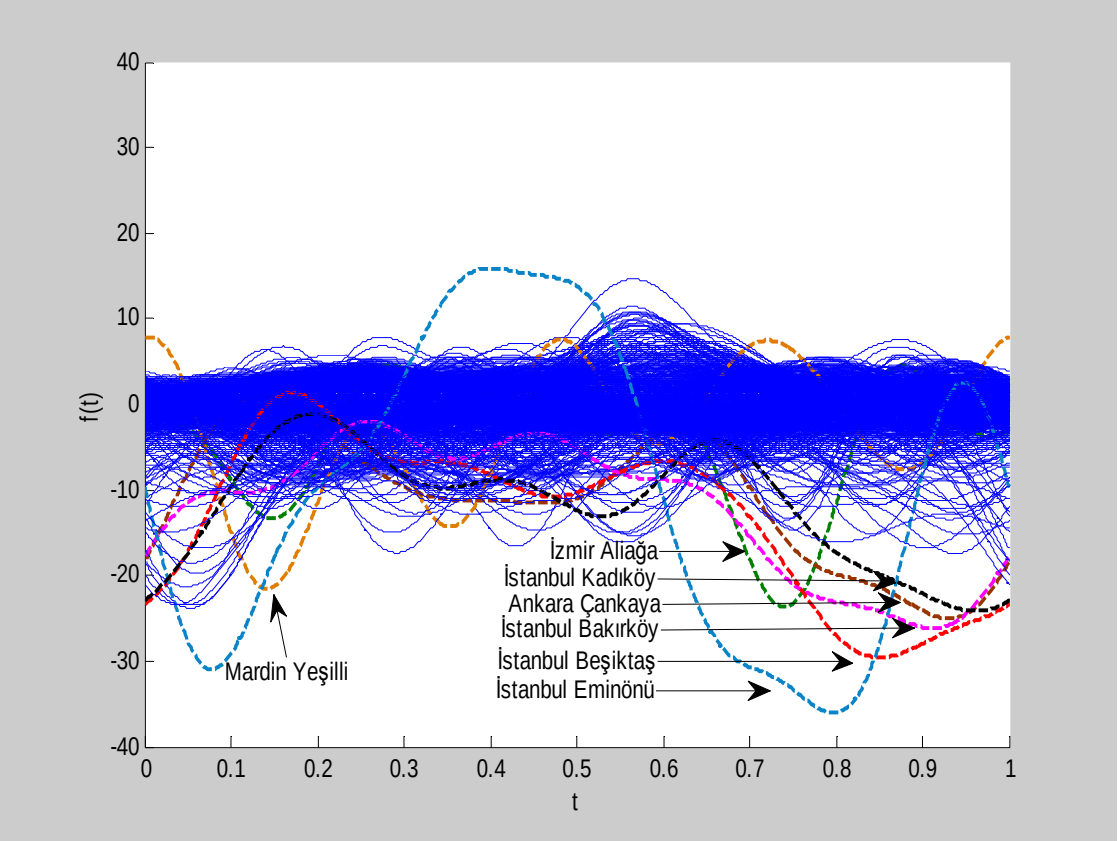
Tablo 5.1 Temel bileşenlere ilişkin özdeğerler

Bileşen No	Özdeğerler	Toplam Varyans Açıklama Oranları
1	10.8754	54.3771
2	1.9698	64.2261
3	1.4605	71.5287
4	1.1443	77.25
5	0.9426	81.9628
6	0.7702	85.8138
7	0.5423	88.5251
8	0.5118	91.0843
9	0.3727	92.9476
10	0.3616	94.7557
11	0.2869	96.1905
12	0.2065	97.2229
13	0.1418	97.9319
14	0.1228	98.546
15	0.0866	98.9789
16	0.0697	99.3272
17	0.0601	99.6275
18	0.0312	99.7835
19	0.0252	99.9094
20	0.0181	100



Şekil 5.1 Özdeğerlerin yamaç eğim grafiği

Bileşenlere ilişkin toplam varyans açıklama oranlarına ve yamaç eğim grafiklerine bakıldığında toplam varyansın yaklaşık % 95' ini açıklayan ilk 10 temel bileşenle çalışmanın uygun olacağı düşünülmüştür. Şekil 5.1' den de gözüktüğü gibi 10. temel bileşenden sonra yamaç eğim grafiğinin eğiminin sabitleştiği gözükmektedir. İlk 10 temel bileşen kullanılarak çizilen Andrews eğrileri grafiği Şekil 5.2' de ki gibidir. Şekil 5.2' ye göre aşırı değerler Eminönü, Beşiktaş, Bakırköy, Çankaya, Kadıköy, Aliağa ve Yeşilli olarak gözükmektedir. Dolayısıyla bu çalışmada Eminönü, Beşiktaş, Bakırköy, Çankaya, Kadıköy, Aliağa ve Yeşilli ilçeleri aşırı değer olarak düşünülmüştür. Şüphesiz aşırı değerleri çıkartılan veri seti için tekrar Andrews eğrileri grafiği çizildiğinde y ekseninde bulunan ölçek hassasiyetinin değişmesine göre yeni aşırı değerler tespit edilebilir. Ancak bu ikinci Andrews eğrileri grafiğinde bulunacak olan aşırı değerler ilk Andrews eğrilerinde bulunan aşırı değerler kadar belirgin olmayacaktır. Unutulmamalıdır veri görselleştirme teknikleri insan algılama yeteneklerini ve insanlar arası yorumlama farklılıklarını dikkate alarak analiz gerçekleştirilmesine olanak sağlar. Görselleştirme teknikleri ile diğer yöntemlerle fark edilmesi daha zor olan bilgiye erişilmesi ve bilginin yorumlanması kolaylaşmaktadır. Ancak grafiksel tekniklerin matematiksel sonuçlar vermemesi gibi bir dezavantajı da bulunmaktadır.



Şekil 5.2 10 temel bileşen için Andrews eğrileri

Tablo 5.2’ de aşırı değerlerin değişken değerleri, Tablo 5.3’ de aşırı değerleri arındırılmış ilçelerin değişkenlerine ilişkin istatistikler bulunmaktadır. Tablo 5.3’ deki ortalama etrafındaki % 99’ luk güven bölgesinin üstüne çıkan aşırı değerler sarı, altına çıkan değişkenler yeşil olmak üzere Tablo 5.2’ deki gibi boyanmıştır. Tabloları incelediğimizde en yüksek değişken değerlerine Eminönü, Beşiktaş, Bakırköy, Çankaya, Kadıköy, Aliağa ilçelerinin sahip olduğu ve en düşük değişken değerlerine Yeşilli ilçesinin sahip olduğu gözlenmektedir. Renklenmelerden Andrews eğrileriyle bulduğumuz aşırı değerlerin doğru olabileceği gözlenmiştir. Değişken bazında aşırı değerlerin tespiti için EK 1’ de ki kutu grafikleri incelenebilir.

Tablo 5.2 Aşırı değerlerin değişken değerleri

Bölgeler	Şehirler	İlçeler	X1	X2	X3	X4	X5
Marmara	İstanbul	Eminönü	-40,53	100,00	6954,38	14,07	21,21
Marmara	İstanbul	Beşiktaş	-0,73	100,00	9086,33	33,40	9,88
Marmara	İstanbul	Bakırköy	-36,98	100,00	6512,44	29,57	10,83
İç Anadolu	Ankara	Çankaya	7,42	98,59	2870,64	30,16	9,92
Marmara	İstanbul	Kadıköy	2,29	100,00	16582,48	31,18	13,01
Ege	İzmir	Aliağa	30,51	65,63	208,73	31,16	6,10
Güneydoğu Anadolu	Mardin	Yeşilli	41,38	89,44	458,88	22,94	32,02
Bölgeler	**Şehirler**	**İlçeler**	**X6**	**X7**	**X8**	**X9**	**X10**
Marmara	İstanbul	Eminönü	95,10	85,68	10,13	13,62	1,18
Marmara	İstanbul	Beşiktaş	99,08	96,31	31,97	25,50	0,18
Marmara	İstanbul	Bakırköy	98,74	95,96	25,97	19,02	0,19
İç Anadolu	Ankara	Çankaya	98,46	93,73	26,71	21,84	0,66
Marmara	İstanbul	Kadıköy	98,42	94,40	26,21	18,80	0,21
Ege	İzmir	Aliağa	97,30	90,59	7,83	5,13	33,48
Güneydoğu Anadolu	Mardin	Yeşilli	89,39	59,57	1,90	0,62	45,06
Bölgeler	**Şehirler**	**İlçeler**	**X11**	**X12**	**X13**	**X14**	**X15**
Marmara	İstanbul	Eminönü	24,73	3,43	38,92	8,51	4,57
Marmara	İstanbul	Beşiktaş	15,52	3,34	21,56	5,51	20,25
Marmara	İstanbul	Bakırköy	20,26	2,41	22,77	10,38	13,98
İç Anadolu	Ankara	Çankaya	7,39	5,99	14,26	4,32	15,04
Marmara	İstanbul	Kadıköy	18,76	4,73	22,26	7,27	18,70
Ege	İzmir	Aliağa	30,25	5,72	9,26	5,04	2,55
Güneydoğu Anadolu	Mardin	Yeşilli	2,25	4,24	6,25	20,08	0,79
Bölgeler	**Şehirler**	**İlçeler**	**X16**	**X17**	**X18**	**X19**	**X20**
Marmara	İstanbul	Eminönü	9,62	1,67	10,07	3698,88	217,49
Marmara	İstanbul	Beşiktaş	28,59	6,88	15,92	696,65	91,19
Marmara	İstanbul	Bakırköy	25,40	7,56	15,60	1305,52	83,01
İç Anadolu	Ankara	Çankaya	28,59	7,35	18,13	321,32	45,10
Marmara	İstanbul	Kadıköy	26,65	5,23	17,33	395,59	44,93
Ege	İzmir	Aliağa	11,16	1,71	6,05	2565,40	26,23
Güneydoğu Anadolu	Mardin	Yeşilli	5,33	2,19	3,35	37,95	4,54

Tablo 5.3 Aşırı değerleri arındırılmış ilçelere ilişkin istatistikler

Tanımlayıcı İstatistikler						
Değişkenler	Ortalama	Ortalama etrafında %99 güven aralığı		Standart Sapma	Minimum	Maksimum
		Alt sınır	Üst sınır			
X1	6,07	4,13	8,01	22,71	-81,11	105,26
X2	46,36	44,55	48,18	21,18	8,04	100,00
X3	475,47	232,39	718,55	2842,39	2,75	35143,71
X4	41,34	40,61	42,08	8,61	10,66	58,92
X5	6,63	6,27	7,00	4,30	0,88	29,12
X6	91,98	91,55	92,41	5,06	60,58	98,48
X7	75,65	74,63	76,68	11,96	19,81	94,74
X8	4,48	4,27	4,68	2,39	1,22	17,97
X9	2,09	1,91	2,26	2,03	0,11	17,27
X10	66,41	64,56	68,25	21,60	0,21	96,13
X11	6,58	5,87	7,28	8,25	0,05	51,23
X12	3,61	3,39	3,82	2,50	0,39	22,27
X13	5,46	5,04	5,88	4,90	0,32	48,54
X14	2,13	2,00	2,26	1,57	0,26	11,03
X15	1,45	1,31	1,59	1,68	0,00	13,96
X16	4,71	4,46	4,96	2,93	0,98	23,06
X17	0,84	0,78	0,90	0,67	0,07	5,39
X18	3,29	3,05	3,52	2,75	0,39	18,95
X19	171,69	157,30	186,08	168,28	15,21	1875,47
X20	9,00	8,45	9,54	6,37	0,00	73,15

5.4 Kümeleme Analizi

Bu alt bölümde tespit edilen aşırı değerlin veri setinden ayıklanmasıyla elde edilen yeni veri setinin kümelenmesi gerçekleştirilecektir.

5.4.1 Temel Bileşenler Analizi

Uygulamada ilk olarak değişkenler arasındaki bağımlılık yapısının ortadan kaldırılması ve veri boyutunun indirgenerek aynı şeyi ifade eden değişkenlerin birleştirilmesi amacıyla verilere temel bileşenler analizi uygulanmıştır. Böylece ilçelerin kümelenmesi korelasyonsuz daha az değişkenle gerçekleştirilebilecektir.

Değişkenler arasında anlamlı ilişkilerin olup olmadığını görmek için R korelasyon matrisini incelemek ve verilere temel bileşenler analizi uygulamanın gerekli olup olmadığını görmek, eğer değişkenler arasında ilişki varsa bunların önemli olup olmadığını anlamak için veri setine küresellik testi uygulanmalıdır. SPSS 15 Evaluation programıyla hesaplanan küresellik testi sonuçları Tablo 5.4’ de verilmiştir.

Tablo 5.4 Küresellik test sonuçları

KMO and Bartlett's Test

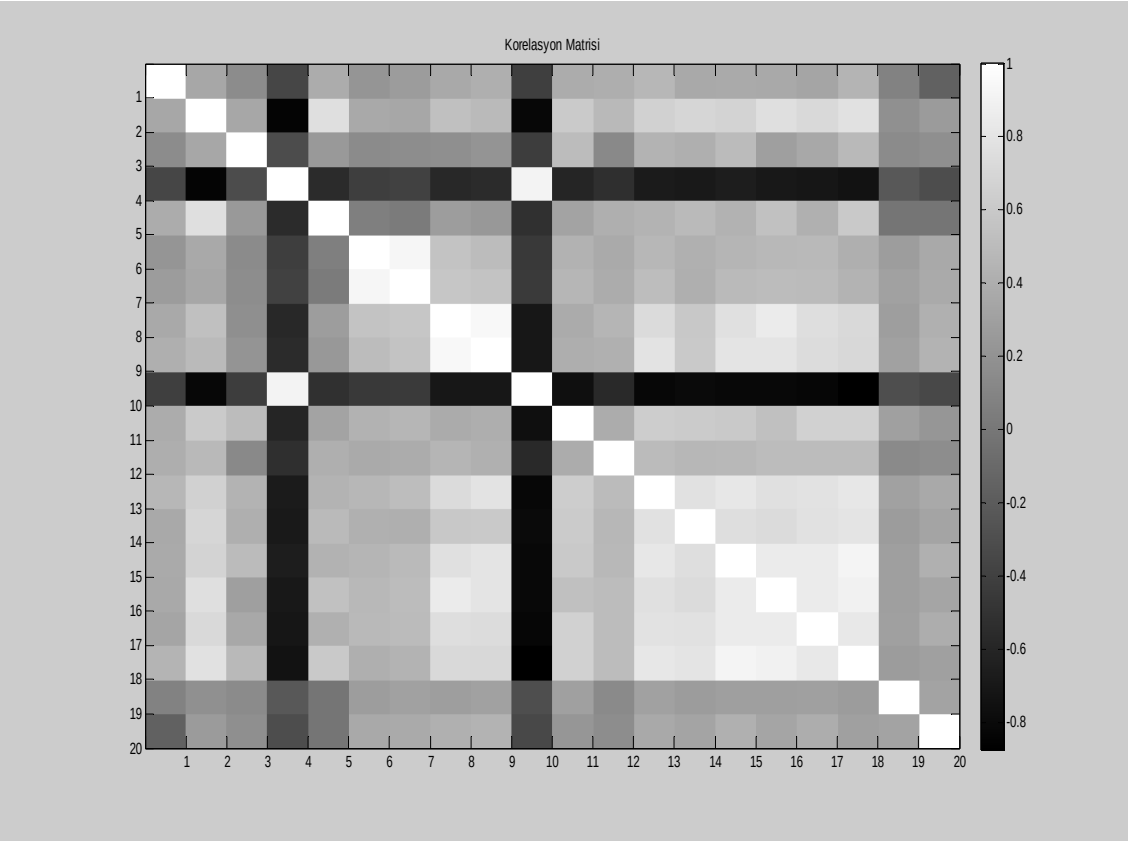
Kaiser-Meyer-Olkin Measure of Sampling Adequacy.		,883
Bartlett's Test of Sphericity	Approx. Chi-Square	22692,117
	df	190
	Sig.	,000

Küresellik testi için;

H0: R=I (İlişki matrisi ile birim matris arasında fark yoktur. Değişkenler arasındaki ilişkiler önemsizdir.)

H1: R≠I (İlişki matrisi ile birim matris arasında fark vardır. Değişkenler arasındaki ilişkiler önemlidir.)

Olasılık değeri olan Sig. değerine bakıldığında; 0.000<0.05 olduğundan hipotez reddedilir. Bu nedenle ilişki matrisi ile birim matris arasında fark olduğu diğer bir ifade ile değişkenler arasındaki ilişkilerin önemli olduğu 0.95 olasılıkla söylenebilir. Bu da temel bileşenler analizi uygulanmasının gerekliliğini ortaya koymaktadır. Şekil 5.3' de değişkenlere ilişkin korelasyon matrisinin matris grafiği bulunmaktadır. Matristeki beyaz renkler pozitif yöndeki korelasyonu, siyah renkler negatif yöndeki korelasyonu göstermektedir.



Şekil 5.3 Değişkenlere ilişkin korelasyon matrisini gösteren matris grafiği

Aşırı değerleri çıkartılan yeni veri setinin temel bileşenler analizinden elde edilen özdeğerler ve toplam varyans açıklama oranları Tablo 5.5 de verilmiştir. Yeni özdeğerlere karşılık gelen özvektörlerin yani bileşenlerin katsayıları Tablo 5.6 ve Tablo 5.7' da verilmiştir. Bileşenlerin toplam varyansı açıklama oranları için çizilen yamaç grafiği ise Şekil 5.4' de verilmiştir.

Tablo 5.5 Yeni temel bileşenlere ilişkin özdeğerler

Bileşen No	Özdeğerler	Toplam Varyans Açıklama Oranları
1	11.0202	55.1008
2	1.9787	64.9943
3	1.2529	71.2588
4	1.0743	76.6304
5	0.8985	81.1229
6	0.79	85.073
7	0.5701	87.9233
8	0.5064	90.4553
9	0.4438	92.6743
10	0.3435	94.3916
11	0.2949	95.8659
12	0.2042	96.8868
13	0.1593	97.6836
14	0.1449	98.408
15	0.0927	98.8716
16	0.0801	99.2721

17	0.0602	99.573
18	0.0331	99.7385
19	0.0329	99.903
20	0.0194	100

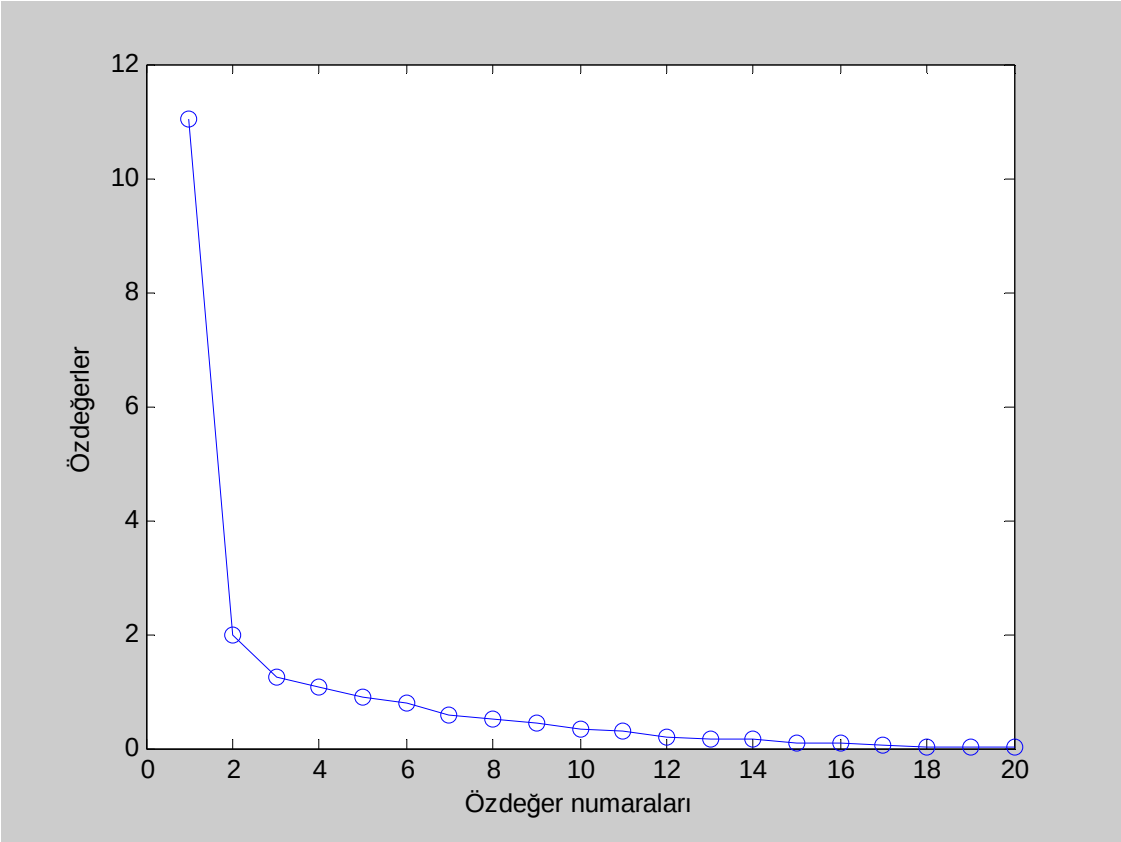
Tablo 5.6 Yeni özdeğerlere ilişkin özvektörler

Özdeğer No	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10
1	-0.1434	-0.1891	0.4717	0.2629	-0.421	-0.1822	0.0643	-0.1164	0.5435	-0.0494
2	-0.2474	-0.243	-0.0405	-0.0165	0.3395	-0.026	-0.2343	-0.0695	0.0932	0.1139
3	-0.1342	-0.1377	-0.4867	0.3752	-0.3123	0.3148	0.1682	-0.3867	-0.1174	0.316
4	0.248	0.1453	0.0219	-0.0049	-0.3137	0.0732	0.0488	-0.2512	-0.198	-0.4653
5	-0.163	-0.4584	0.0875	-0.0856	0.2576	-0.0497	-0.2037	-0.525	0.0581	-0.1262
6	-0.1781	0.4004	0.2204	0.3302	0.2442	0.1975	-0.2032	-0.1448	-0.0582	-0.0326
7	-0.1842	0.4166	0.2285	0.3268	0.1453	0.1715	-0.1627	-0.148	-0.0428	-0.0195
8	-0.2472	0.2019	0.1695	-0.3204	-0.1756	0.0521	-0.0479	-0.0136	-0.0475	0.2483
9	-0.2459	0.2026	0.1275	-0.3026	-0.2951	0.0277	0.0186	0.0098	0.0845	0.1833
10	0.2812	0.1005	0.0693	-0.0212	-0.0769	0.0482	-0.0412	-0.306	-0.0527	-0.2232
11	-0.2175	-0.0636	-0.1759	0.4544	-0.0107	-0.0197	-0.0039	0.4344	0.0954	-0.0867
12	-0.1821	-0.0687	0.3181	0.0812	0.2595	-0.1527	0.7895	-0.096	-0.3358	0.0382
13	-0.2687	0.0039	-0.0196	-0.0126	-0.1974	0.0186	0.088	0.017	0.1111	-0.137
14	-0.2533	-0.0862	-0.1092	0.0111	0.0099	0.0037	0.016	0.1218	-0.0329	-0.6406
15	-0.2737	0.0098	-0.1057	-0.1151	-0.1969	0.1174	0.0433	-0.0684	-0.1429	-0.0726
16	-0.2736	-0.008	0.0568	-0.2335	-0.0316	0.01	-0.1851	-0.0559	-0.2388	-0.0013
17	-0.2712	0.0187	-0.0585	-0.1002	-0.0217	0.0308	-0.0106	0.2278	-0.1685	-0.1804
18	-0.2797	-0.1246	-0.0625	-0.058	-0.1049	0.0437	-0.0807	-0.058	-0.1379	-0.0911
19	-0.1067	0.2643	-0.2596	0.133	-0.0838	-0.8613	-0.1328	-0.1946	-0.1481	0.0479
20	-0.1268	0.3564	-0.3829	-0.2445	0.2754	0.0494	0.324	-0.2064	0.5832	-0.1388

Tablo 5.7 Yeni özdeğerlere ilişkin özvektörler

Özdeğer No	X11	X12	X13	X14	X15	X16	X17	X18	X19	X20
1	0.0503	-0.3121	0.055	-0.0518	0.0767	-0.0804	0.0697	-0.0311	-0.0111	-0.0462
2	0.0653	0.0127	0.1645	-0.1688	0.5889	0.3971	0.2585	0.184	0.122	-0.0203
3	-0.1445	-0.0802	0.2402	0.0588	0.0506	-0.0632	-0.0216	-0.0096	0.041	0.0014
4	0.3659	0.1949	0.0951	0.1286	0.1636	0.1158	0.2201	0.1258	0.1677	-0.4025
5	0.1398	0.2549	0.0768	0.115	-0.4373	0.0167	-0.1808	-0.0871	-0.0565	-0.0668
6	-0.1226	-0.0374	0.0199	0.0445	-0.2861	-0.0581	0.5876	-0.0973	0.1141	0.1152
7	0.0109	0.0257	-0.0984	-0.1091	0.1939	0.0435	-0.6443	0.0934	-0.0519	-0.2095
8	0.0217	0.0369	0.2676	0.3588	-0.0249	0.0664	0.0902	0.3449	-0.5753	-0.0392
9	-0.0425	0.2277	0.1012	0.1173	-0.1166	0.3304	-0.1617	-0.119	0.6106	0.2194
10	0.1102	-0.0233	0.1479	-0.1827	0.1426	0.0895	-0.0903	0.026	-0.1967	0.7782
11	0.515	0.2621	0.0256	0.2924	-0.0532	0.1082	-0.0411	-0.0609	-0.1011	0.2349
12	0.0521	0.0264	0.0154	0.0336	0.0695	0.035	0.0186	0.0164	0.0313	0.0671
13	-0.2581	0.6938	-0.119	-0.403	0.1207	-0.24	0.1356	-0.0033	-0.1827	0.0173
14	-0.5417	-0.1615	0.1382	0.3355	0.0831	0.1325	-0.1023	-0.0188	-0.0171	0.0546
15	0.1298	-0.2415	-0.4492	-0.2455	-0.1151	0.4943	0.0882	-0.3472	-0.2973	-0.0648
16	0.2138	-0.1062	-0.0065	0.1895	0.3902	-0.5177	-0.0006	-0.5027	0.0418	0.064
17	0.2121	-0.2299	0.5485	-0.5337	-0.2703	-0.1298	-0.0674	0.0988	0.0687	-0.0758
18	0.1213	-0.1666	-0.4959	0.0291	-0.0605	-0.2412	0.0203	0.6342	0.2322	0.2031

19	-0.0462	-0.01	0.0141	-0.0208	-0.0316	0.0299	0.0117	0.0091	-0.0122	-0.0046
20	0.1923	-0.1006	-0.0222	0.0435	0.0188	-0.1227	0.0116	0.0028	0.0006	-0.0167



Şekil 5.4 Yeni özdeğerlerin yamaç eğim grafiği

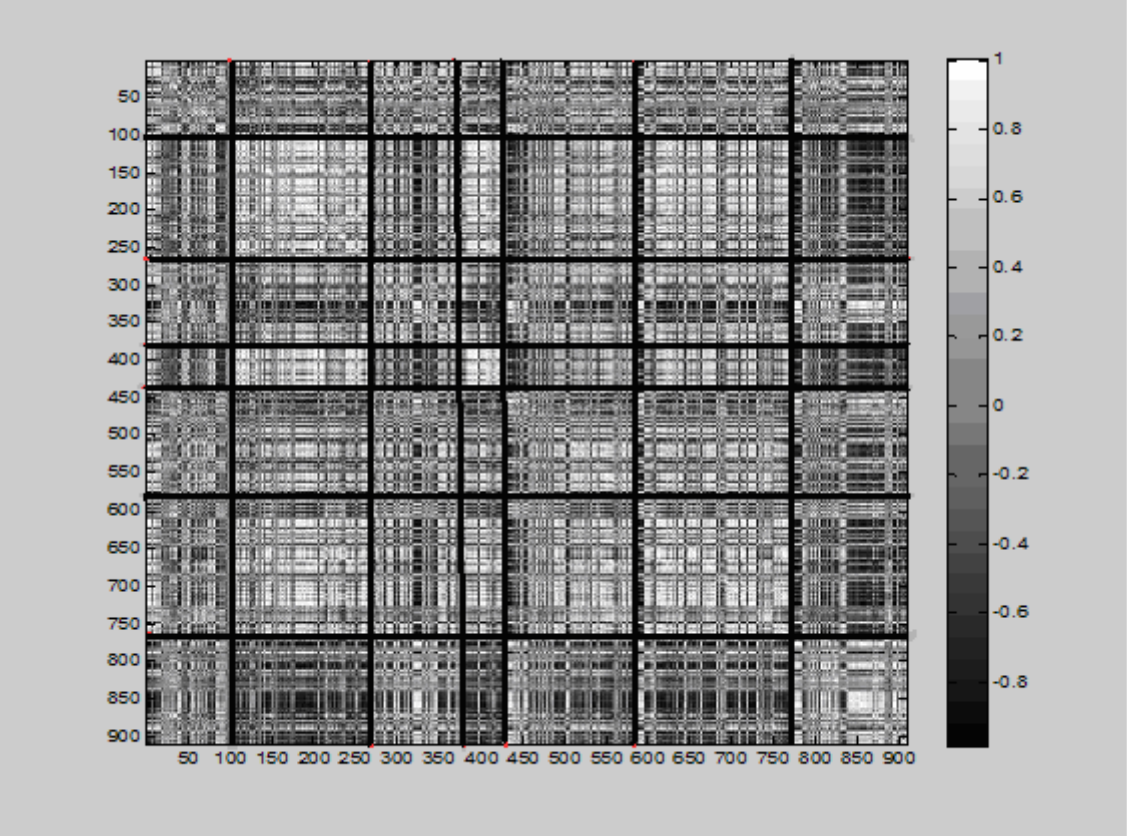
Bileşenlere ilişkin toplam varyans açıklama oranlarına ve yamaç eğim grafiklerine bakıldığında toplam varyansın yaklaşık % 94' ünü açıklayan ilk 10 temel bileşenle çalışmanın uygun olacağı düşünülmüştür. Bundan sonraki analiz aşamalarında aşırı değerlerden arındırılmış veri seti için elde edilen ilk 10 temel bileşenle çalışılmaya devam edilecektir. Bu sayede veri setleri korelasyondan arındırılmış ve daha az değişkenle kümeleme analizlerinin yapılabilmesi mümkün olacaktır. Dolayısıyla, kümeleme analizlerinde kritik öneme sahip olan iki nokta arasındaki uzaklıkların daha az anlamlı hale gelmesi değişken sayısı indirilerek önlenmiş olacaktır.

5.4.2 Uygun Küme Sayısının ve Algoritmasının Belirlenmesi

Kümeleme analizlerinde, doğru küme sayısının belirlenmesi kümeleme modeli geliştirme işlemlerinin ayrılmaz bir parçasıdır. Çünkü bir veri kümesinde küme yapısı olmasa bile kümeleme algoritmaları bu veri seti içersinde istenilen sayıda küme bulacaktır. Ancak elimizdeki veri kümesinde herhangi bir küme yapısı bulunmayabilir. Bundan dolayı bir veri

kümesindeki doğal küme yapılarını bulmak için görsel ve analitik yöntemler birlikte kullanılabilirler. Bu alt bölümde öncelikle veri setimizdeki doğal küme yapılarının varlığı görsel yöntemlerle keşfedilmeye çalışılmıştır. Daha sonra analatik yöntemlerle doğru küme sayılarının kestirilmesine gidilmiş; analatik ve görsel yöntemlerin birlikte kullanılmasıyla uygun kümeleme algoritması seçilmiş ve kümeleme işlemlerinin kalitesi arttırılmıştır.

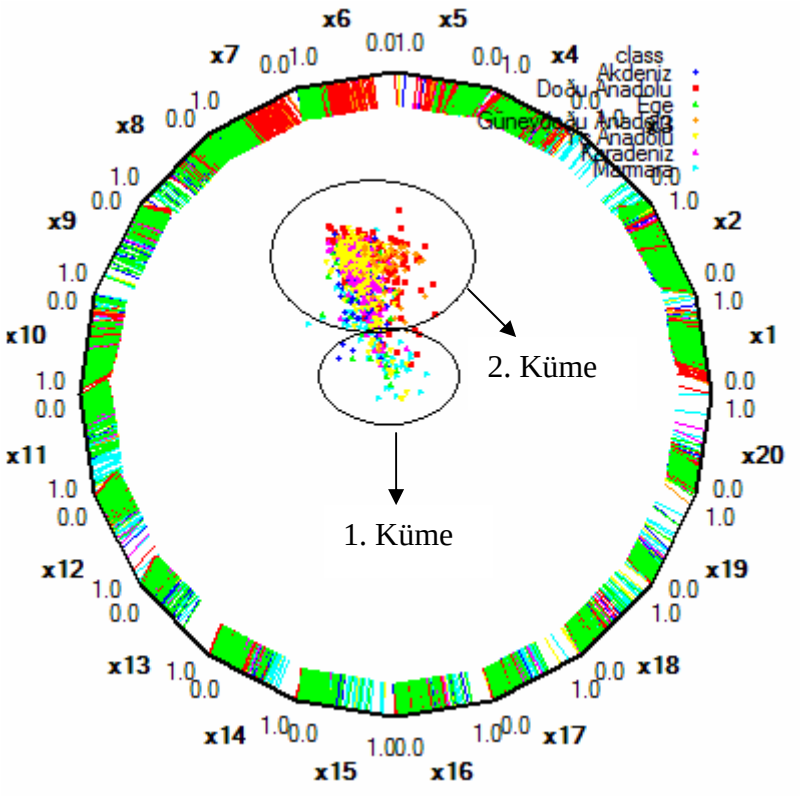
Veri setimde bulunan ilçeler bulundukları bölgelere göre sıralanmışlardır. Buna göre 1-104 aralığı Akdeniz, 105-268 aralığı Doğu Anadolu, 269-384 aralığı Ege, 385-433 aralığı Güneydoğu Anadolu, 434-585 aralığı İç Anadolu, 586-776 aralığı Karadeniz ve 777-911 aralığı Marmara Bölgesi olmak üzere ilk 10 temel bileşene göre bölgeler arası korelasyon benzerlik matrisinin matris grafiği Şekil 5.5' deki gibidir. Şekle göre Doğu Anadolu ve Güneydoğu Anadolu bölgeleri birbirine benzer özellikler sergilemektedir. Marmara bölgesi ise kendi içinde homojen diğer bölgelere göre heterojen bir yapı sergilemektedir. Karadeniz ve İç Anadolu bölgeleri ise Marmara, Doğu Anadolu ve İç Anadolu bölgeleri kadar olmasa da, kendi içinde homojen bir yapı sergilemektedir. Karadeniz bölgesinin tamamına yakını ve İç Anadolu' nün bir kısmı Doğu ve Güneydoğu Anadolu bölgelerine benzer bir yapı sergilemektedir. Ege ve Akdeniz bölgeleri ise kendi içlerinde homojen olmayan bir yapı sergilemektedirler. Sonuç olarak Şekil 5.4' e bakarak Doğu ve Güneydoğu Anadolu bölgelerinin bir gurup, Marmara bölgesininse farklı bir gurup oluşturduğu, diğer bölgelerin ilçeleri Doğu Anadolu, Güneydoğu Anadolu ve Marmara Bölgelerinin ilçelerine benzer özellikler sergilediği gözlenmiştir. Sonuç olarak bölgelere göre düzenlenmiş veri setimizde doğal küme yapıları net bir şekilde gözükmemektedir; ancak kabaca bir sonuca varılabilmektedir. Sonuç olarak kabaca veri setimizde iki kümenin olabileceğini düşünebiliriz.



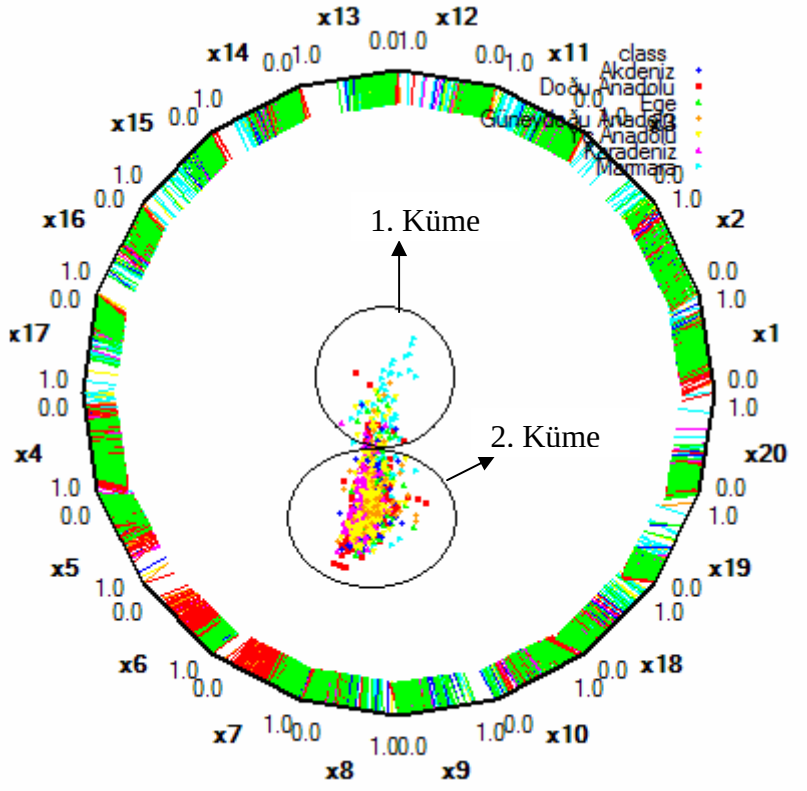
Şekil 5.5 Bölgeler arası korelasyon benzerlik matrisi

İlçelerin küme yapılarını kavrayabilmek için başka grafiklerde kullanılabilir. Şekil 5.6 ve Şekil 5.7’ de ilçe veri seti için değişkenleri farklı sırlanmış PolyViz grafiği bulunmaktadır. Şekil 5.6 ve Şekil 5.7’ de veri setimizde 2 kümenin bulunabileceğine yönelik ipuçları elde dilmiştir. Şüphesiz değişkenlerin farklı sıralanmasına göre PolyViz grafiğinde verilerin dağılımları ve biçimleri değişecektir. Bu durumda PolyViz grafikleri küme yapıları tespitinde sadece yol gösterici olabilmektedir.

Değişkenleri farklı sıralanmış PolyViz grafiğinde veri setimizde 2 kümenin bulunabileceğine yönelik ipuçları elde edilmektedir. PolyViz grafiklerine göre kabaca Marmara bölgesi ilçelerinin çoğunluğu bir kümeyi, geri kalan ilçelerinse diğer bir kümeyi oluşturduğu gözlenmiştir. PolyViz grafiklerinde dikkati çeken diğer bir nokta da küme potansiyeli taşıyan veri dağılımlarının küresel olmaması, yoğunluk ve hacimlerinin farklı olmasıdır. Bu da daha sonra yapılacak kümeleme çalışmalarını olumsuz yönde etkileyecektir.



Şekil 5.6 Farklı sıralanmış değişkenler içi PolyViz grafiği



Şekil 5.7 Farklı sıralanmış değişkenler için PolyViz grafiği

Uygun küme sayılarının tespitine yönelik grafik yöntemlerden başka çeşitli analitik yöntemlerde bulunmaktadır. Tez çalışmasında, uygun küme sayısının tespitine yönelik, en çok kullan Silhouette (S), Davies-Bouldin (DB), Dunn (D), Calinski ve Harabasz (CH), Krzanowski ve Lai (KL) ve Hartigan (H) küme doğruluk (cluster validity) endekslerine değinilmiştir.

Doğal gurupları bilinen süsen veri setinin ve doğal gurupları bilinmeyen 911 ilçenin tek bağlantılı hiyerarşik, tam bağlantılı hiyerarşik, k-ortalamalar ve SOM[1] kümeleme yöntemlerine göre 10 küme için Silhouette (S), Davies-Bouldin (DB), Dunn (D), Calinski ve Harabasz (CH), Krzanowski ve Lai (KL) ve Hartigan (H) endeksleriyle hesaplanmış uygun küme sayıları Tablo 5.8' de verilmiştir.[2] Tablo 5.8' de süsen veri setinin, ilçe verileriyle aynı tabloda bulunmasının sebebi: doğal küme sayıları bilinen süsen veri setinin çeşitli kümeleme ve sınıflandırma algoritmalarının doğruluğunu göstermesi açısından sağladığı faydadır.

Tablo 5.8' deki çeşitli kümeleme yöntemleri için hesaplanan endeksler, süsen veri setinin doğal küme sayılarının tespitine yönelik her zaman doğru sonuçlar vermemektedir. Endeksler çoğunluk itibariyle süsen veri setinde 2 ya da 3 küme bulunabileceğini söylemektedir. Aynı şey 911 ilçe veri seti için de geçerlidir. Endekslere göre ortak bir küme sayısı söylenememektedir. Ancak çoğunluk itibariyle endeksler 911 ilçe veri setinde 2 ya da 3 küme bulunabileceğini söylemektedir. Veri setlerine ilişkin küme doğruluk endeks değerleri EK 3' de verilmiştir.

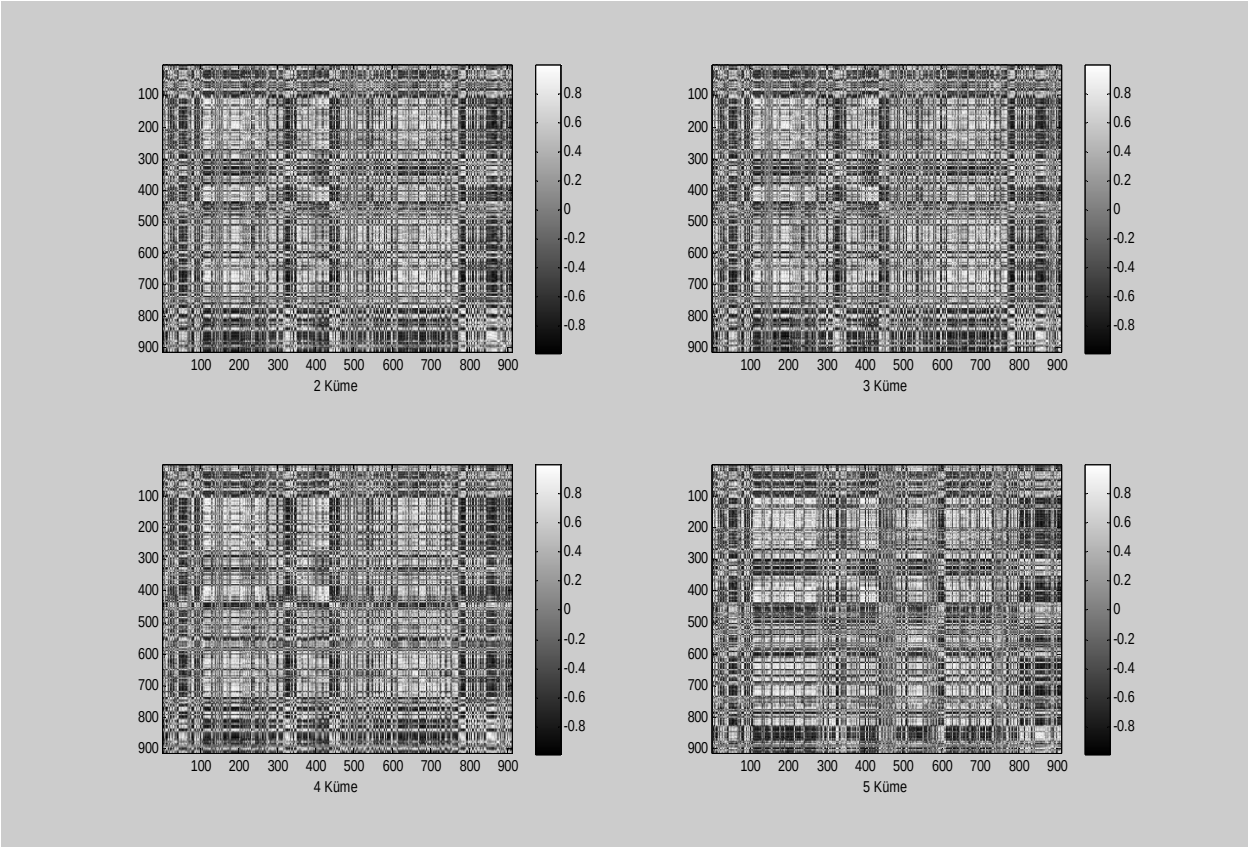
Tablo 5.8 İki veri seti için küme doğruluk endeksleri

Kümeleme Algoritmaları		S	DB	D	CH	KL	H
Süsen Veri Seti	Tek Bağlantılı	2	10	2	2	2	2
	Tam Bağlantılı	4	4	2	4	4	2
	K-Ortalamalar	2	2	2	3	8	2
	SOM	2	3	3	3	3	3
911 İlçe Veri Seti	Tek Bağlantılı	2	10	2	6	6	6
	Tam Bağlantılı	2	2	2	3	3	3
	K-Ortalamalar	2	2	2	2	2	2
	SOM	2	3	2	2	2	2

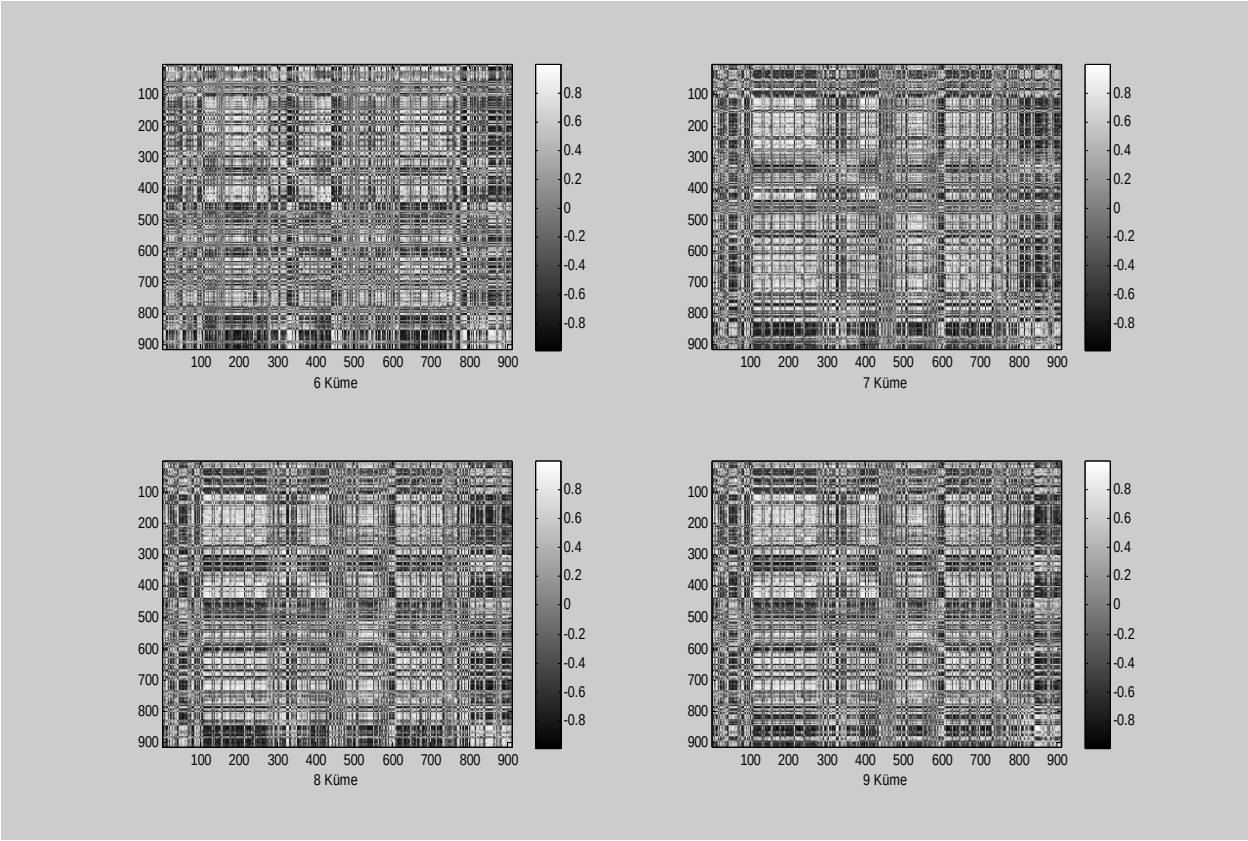
Uygun küme sayısının tespitine yönelik endeks hesaplamalarında net bir sonuca varılamamıştır. Ancak bu 4 kümeleme yöntemi için elde edilen sonuçlar görselleştirilerek küme kaliteleri anlaşılabilir. Bu sayede uygun küme sayısı tespit edilebilir. Şekil 5.8, 5.9, 5.10 5.11, 5.12, 5.13, 5.14 ve 5.15' de 4 farklı kümeleme yöntemiyle elde edilen sonuçların

[1] Tez çalışmasında SOM kümeleme algoritmasında iterasyon sayısı 1000 olarak kullanılmıştır.
[2] Tez çalışmasında bütün kümeleme işlemelerinde öklid uzaklığı kullanılmıştır.

korelasyon benzerlik ölçülerine göre çizilmiş matris grafikleri bulunmaktadır. Matris grafiklerinde beyaz renkler ilçe veri setindeki ilçelerin birbirilerine çok benzediğini, koyu renkler ilçe veri setindeki ilçelerin birbirilerine hiç benzemediğini gösterir. Bu sayede matris grafikleriyle ilçelerin kendi içinde homojen kendi aralarında heterojen bir yapıda kümelenip kümelenmediği görsel bir şekilde anlaşılabilir.

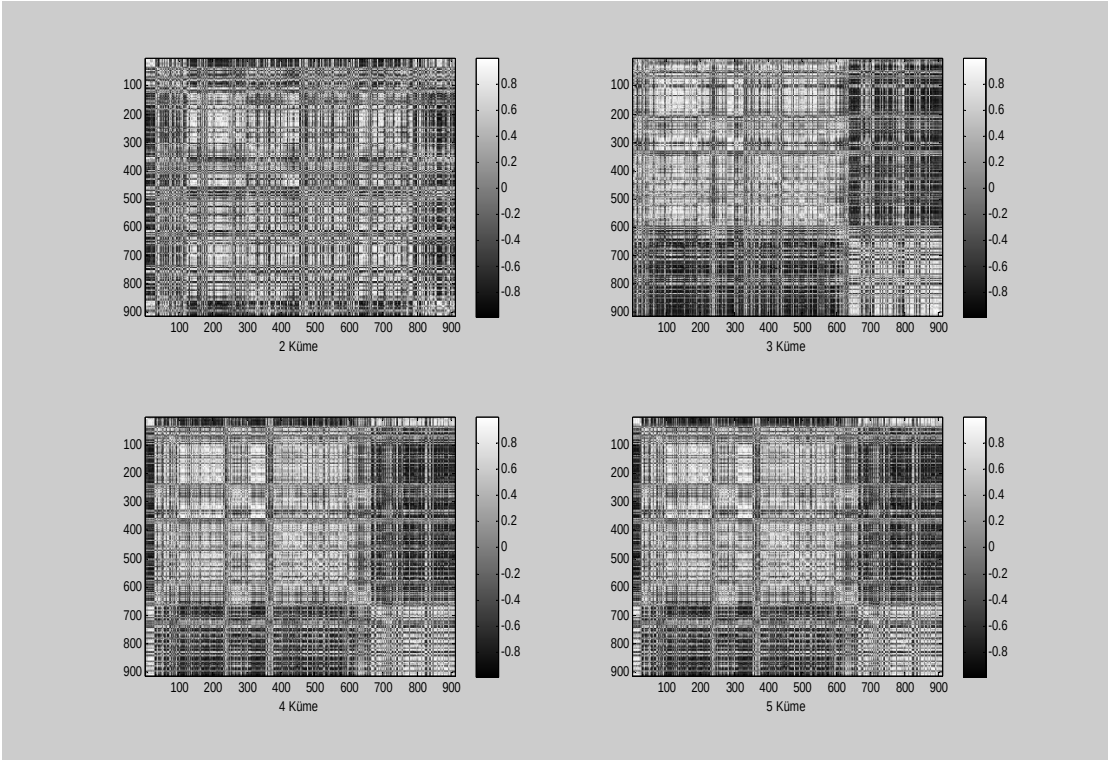


Şekil 5.8 Tek bağlantılı hiyerarşik kümeleme yöntemiyle 2, 3, 4 ve 5 kümeye ayrılan ilçe veri setinin küme sonuçlarını gösteren matris grafikleri

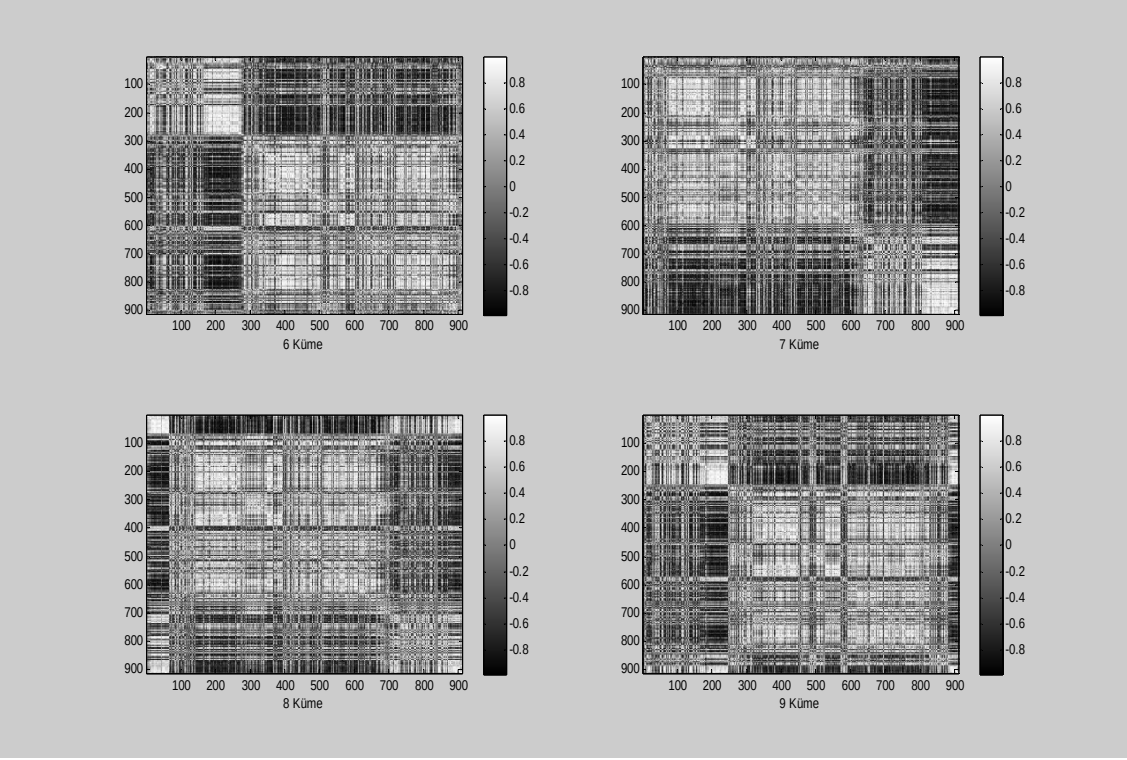


Şekil 5.9 Tek bağlantılı hiyerarşik kümeleme yöntemiyle 6, 7, 8 ve 9 kümeye ayrılan ilçe veri setinin küme sonuçlarını gösteren matris grafikleri

Şekil 5.8 ve Şekil 5.9' da tek bağlantılı hiyerarşik kümeleme yöntemiyle elde edilen 2, 3, 4, 5, 6, 7, 8 ve 9 kümenin sonuçları matris grafikleriyle gösterilmiştir. Şekillere göre tek bağlantılı hiyerarşik kümeleme yöntemiyle belirgin, kendi içinde homojen, kendi aralarında heterojen küme sonuçları elde edilememiştir. Dolayısıyla tek bağlantılı hiyerarşik kümeleme yönteminin ilçeleri kümelemede başarısız olduğu söylenebilir.

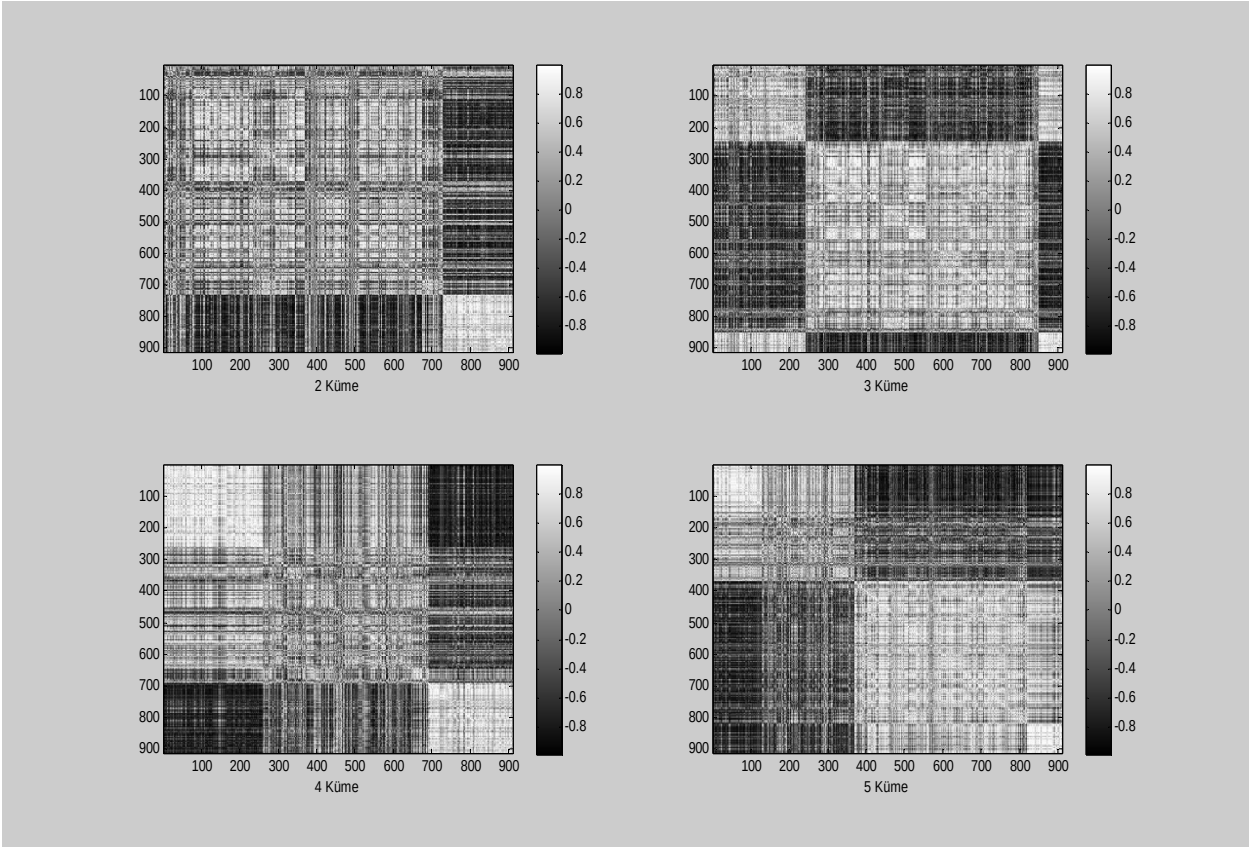


Şekil 5.10 Tam bağlantılı hiyerarşik kümeleme yöntemiyle 2, 3, 4 ve 5 kümeye ayrılan ilçe veri setinin küme sonuçlarını gösteren matris grafikleri

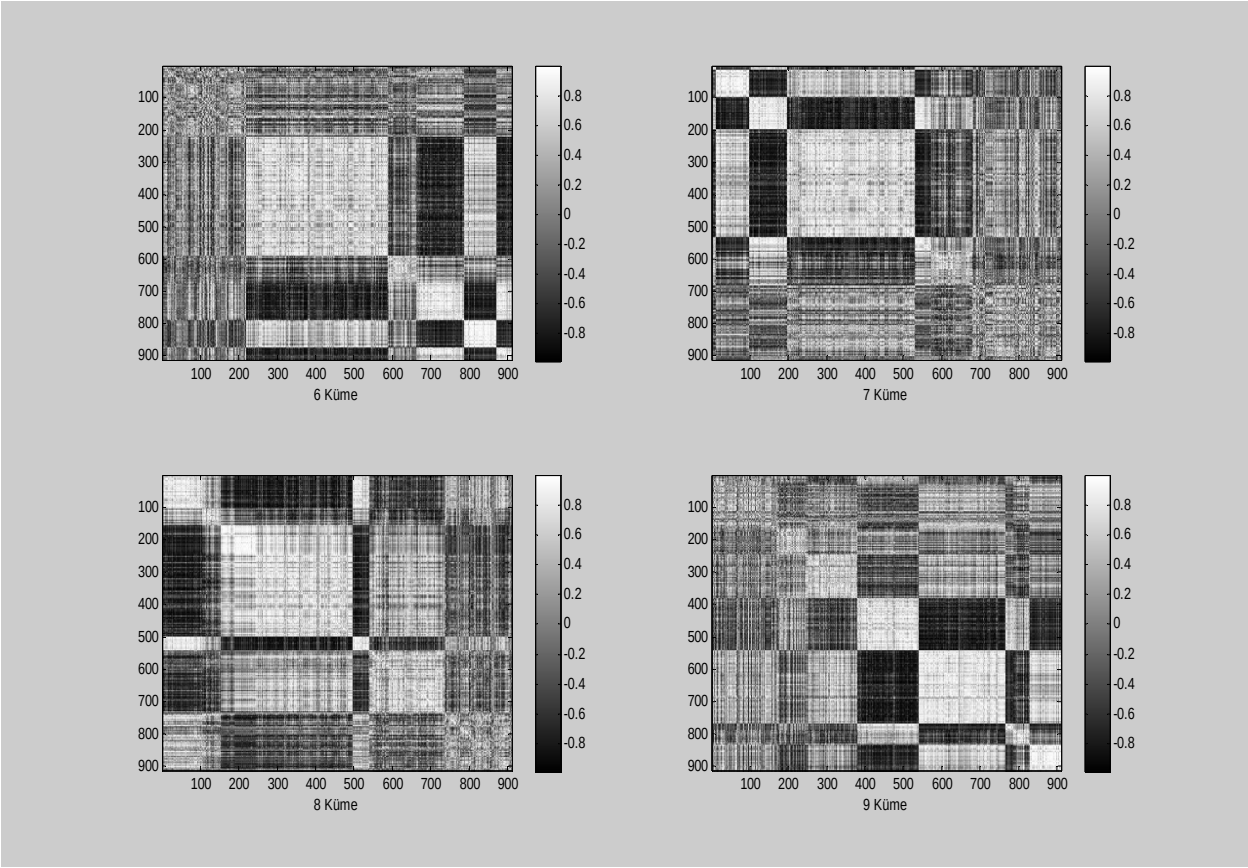


Şekil 5.11 Tam bağlantılı hiyerarşik kümeleme yöntemiyle 6, 7, 8 ve 9 kümeye ayrılan ilçe veri setinin küme sonuçlarını gösteren matris grafikleri

Şekil 5.10 ve Şekil 5.11’ de tam bağlantılı hiyerarşik kümeleme yöntemiyle elde edilen 2, 3, 4, 5, 6, 7, 8 ve 9 kümenin sonuçları matris grafikleriyle gösterilmiştir. Şekillere göre tam bağlantılı hiyerarşik kümeleme yöntemiyle kendi içinde homojen kendi aralarında heterojen küme yapıları net bir şekilde gözlenememiştir. Ancak tam bağlantılı kümeleme yönteminin tek bağlantılı kümeleme yöntemine göre daha belirgin kümeler elde ettiği söylenebilir.



Şekil 5.12 K-ortalamalar kümeleme yöntemiyle 2, 3, 4 ve 5 kümeye ayrılan ilçe veri setinin küme sonuçlarını gösteren matris grafikleri

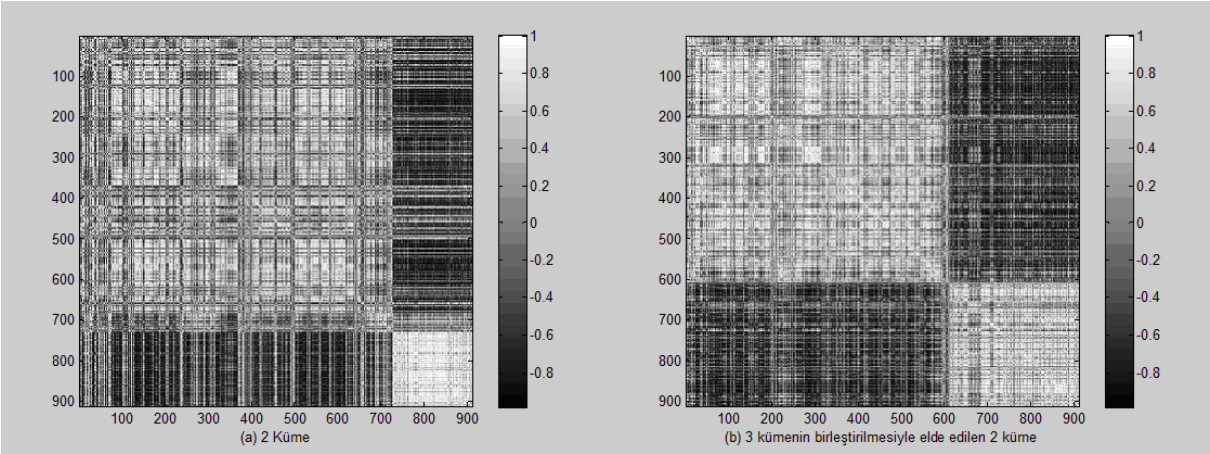


Şekil 5.13 K-ortalamalar kümeleme yöntemiyle 6, 7, 8 ve 9 kümeye ayrılan ilçe veri setinin küme sonuçlarını gösteren matris grafikleri

Şekil 5.12 ve Şekil 5.13' de k-ortalamalar kümeleme yöntemiyle elde edilen 2, 3, 4, 5, 6, 7, 8 ve 9 kümenin sonuçları matris grafikleriyle gösterilmiştir. Şekillere göre sadece k-ortalamalar kümeleme yöntemiyle 2' ye ayrılan ilçe veri seti kendi içinde homojen kendi aralarında heterojen bir yapı sergilemektedir. Dolayısıyla k-ortalamalar kümeleme yöntemine göre ilçe veri seti için en uygun küme sayısı 2' dir. K-ortalamalar kümeleme yöntemiyle 2 kümeye ayrılan ilçe veri seti için elde edilen Silhouette endeksi 0.6791' dir. Bu da küme sonuçlarının kabul edilebilir olduğunu gösterir. Bu iki küme için çizilen matris grafiklerine baktığımızda küme içi homojenliği ve kümeler arası heterojenliği bozan bazı noktaların bulunduğu gözlenmiştir. Yani k-ortalamalar kümeleme yöntemiyle 2' ye ayrılan ilçe veri setinin iyi kümelenmesini bozan bazı ilçelerin bulunduğu anlaşılmıştır.

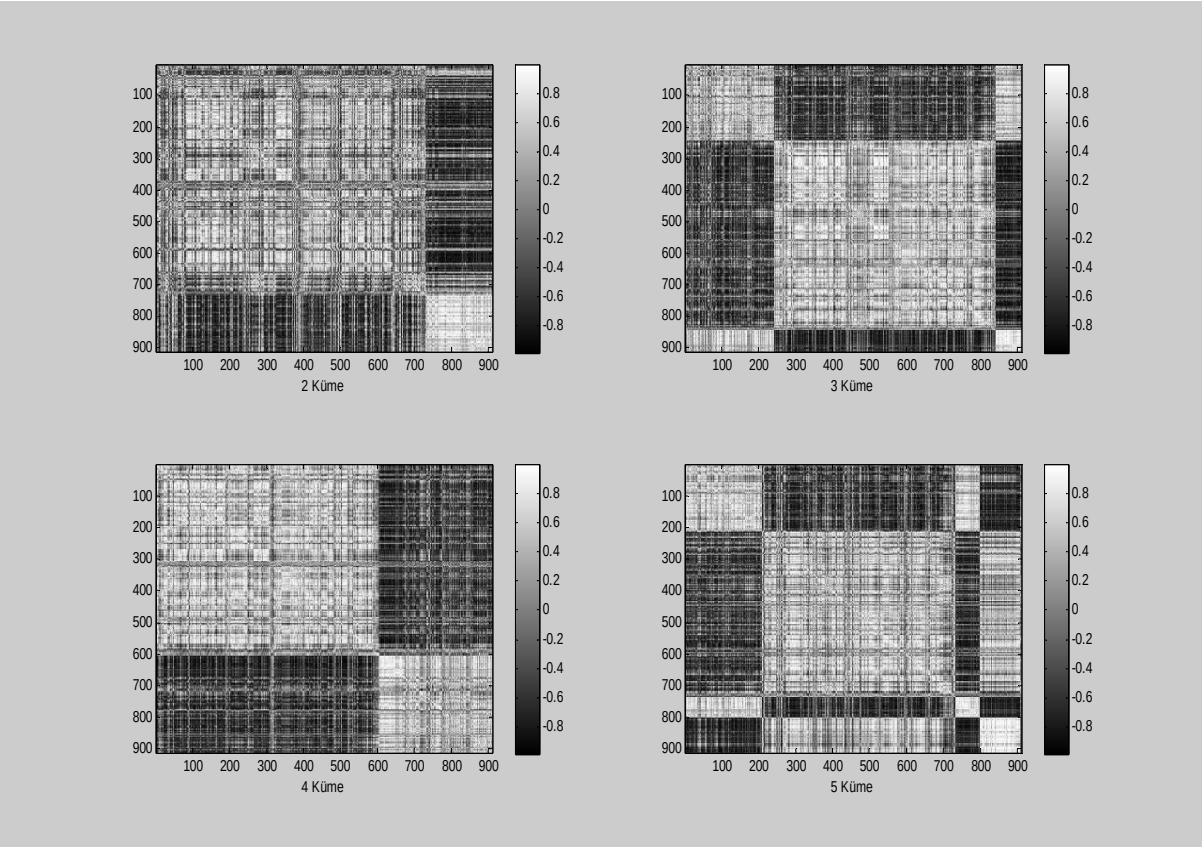
Şekil 5.6 ve Şekil 5.7' deki PolyViz grafiklerinde küme potansiyeli taşıyan veri dağılımlarının küresel olmadığı, yoğunluk ve hacimlerinin farklı olduğu tespit edilmişti. Bu da k-ortalamalar kümeleme yönteminin başarısını olumsuz yönde etkilemektedir. Veri seti 2' den fazla alt kümeye ayrılarak bu başarısız kümelenmenin önüne geçilebilir. K-ortalamalar kümeleme yönteminin farklı küme sonuçları için çizilen Şekil 5.12 ve Şekil 5.13' deki matris grafiklerine

bakıldığında sadece Şekil 5.12’ de 3 kümeye ayrılan ilçe veri setinin kendi içinde homojen kümelenmeler sergilediği gözlenmiştir. Bu 3 kümeden 2 tanesinin birbirine benzediği gözlenmiştir. Bu 2 benzeyen kümenin birleştirilmesi suretiyle 3 küme 2 kümeye indirgenerek k-ortalamalar kümeleme yönteminin başarısı arttırılabilir.

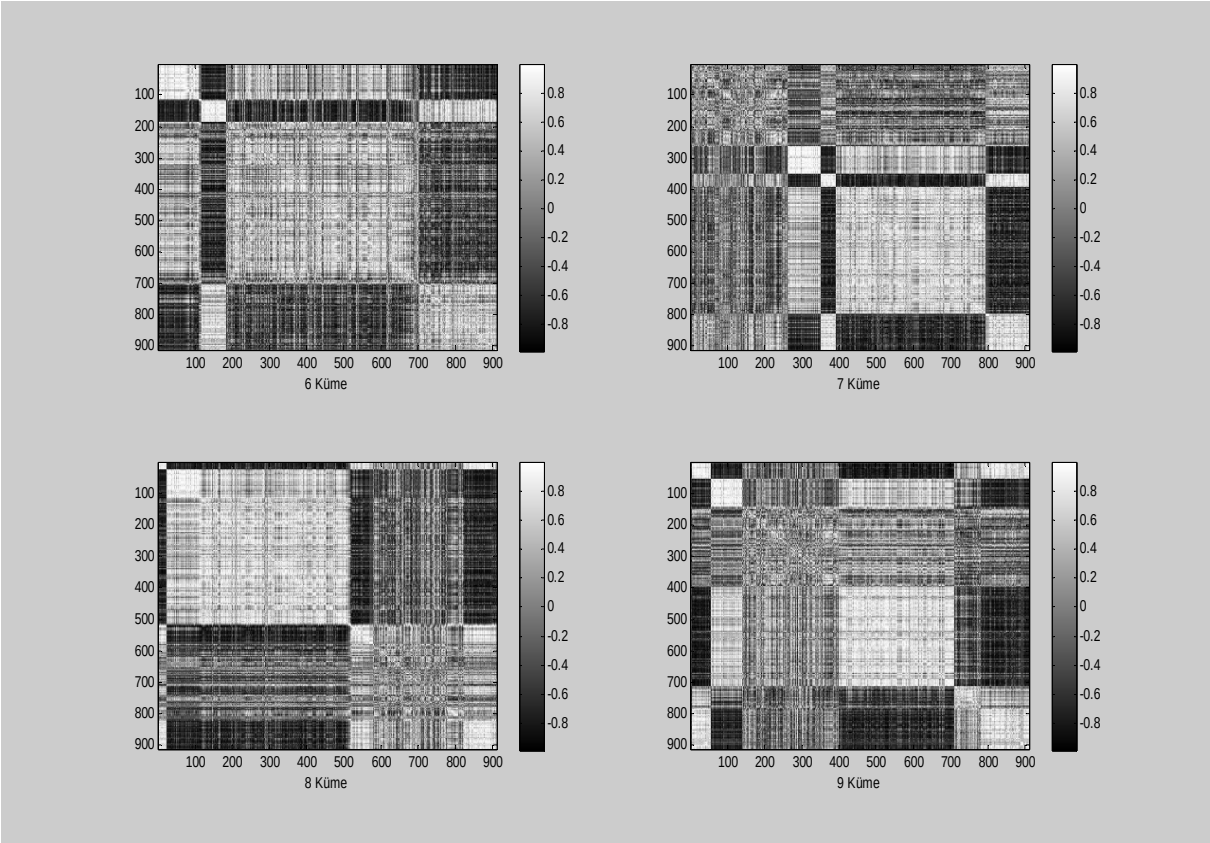


Şekil 5.14 K-ortalamalar kümeleme yöntemiyle 2 ve 3 kümeye ayrılan ilçe veri setinin küme sonuçlarını gösteren matris grafiği

Şekil 5.14 (a)’ da k-ortalamalar kümeleme yöntemiyle 2 kümeye ayrılan ilçe veri setini, Şekil 5.14 (b)’ de k-ortalamalar kümeleme yöntemiyle önce 3 kümeli daha sonra kümelerin birleştirilmesiyle elde edilen 2 kümeli ilçe veri setini gösteren matris grafiği bulunmaktadır. Şekil 5.14 (b)’ deki matris grafiği, Şekil 5.14 (a)’ daki matris grafiğine göre kendi içinde daha homojen kendi aralarında daha heterojen bir yapı sergilemektedir. Dolayısıyla Şekil 5.14 (b)’ nin elde edilmesinde kullanılan kümeleme yönteminin daha başarılı olduğu sonucu ortaya çıkmaktadır.



Şekil 5.15 SOM kümeleme yöntemiyle 2, 3, 4 ve 5 kümeye ayrılan ilçe veri setinin küme sonuçlarını gösteren matris grafikleri

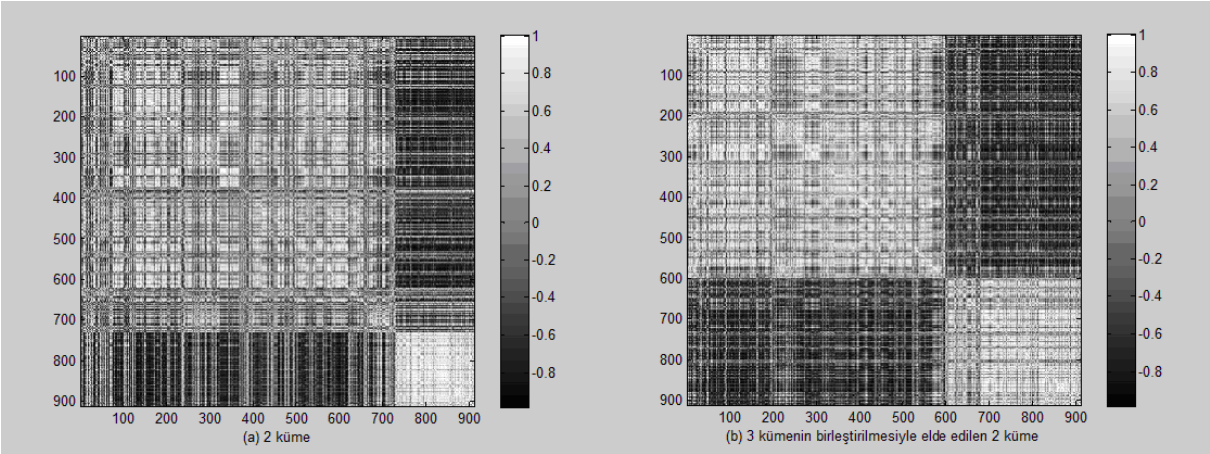


Şekil 5.16 SOM kümeleme yöntemiyle 6, 7, 8 ve 9 kümeye ayrılan ilçe veri setinin küme sonuçlarını gösteren matris grafikleri

Şekil 5.15 ve Şekil 5.16' da SOM kümeleme yöntemiyle elde edilen 2, 3, 4, 5, 6, 7, 8 ve 9 kümenin sonuçları matris grafikleriyle gösterilmiştir. Şekillere göre sadece SOM kümeleme yöntemiyle 2' ye ayrılan ilçe veri seti kendi içinde homojen kendi aralarında heterojen bir yapı sergilemektedir. Dolayısıyla SOM kümeleme yöntemine göre ilçe veri seti için en uygun küme sayısı 2' dir. SOM kümeleme yöntemiyle 2 kümeye ayrılan ilçe veri seti için elde edilen Silhouette endeksi 0.6814' dür. Bu da küme sonuçlarının kabul edilebilir olduğunu gösterir. Bu iki küme için çizilen matris grafiklerine baktığımızda küme içi homojenliği ve kümeler arası heterojenliği bozan bazı noktaların bulunduğu gözlenmiştir. Yani SOM kümeleme yöntemiyle 2' ye ayrılan ilçe veri setinin iyi kümelenmesini bozan bazı ilçelerin bulunduğu anlaşılmıştır.

Şekil 5.6 ve Şekil 5.7' deki PolyViz grafiklerinde küme potansiyeli taşıyan veri dağılımlarının küresel olmadığı, yoğunluk ve hacimlerinin farklı olduğu tespit edilmişti. Bu da k-ortalamalar gibi belirli bir merkeze göre kümeleme yapan SOM kümeleme yönteminin başarısını olumsuz yönde etkilemektedir. K-ortalamalar kümeleme yönteminde yapıldığı gibi veri seti 2' den fazla alt kümeye ayrılarak bu başarısız kümelenmenin önüne geçilebilir. SOM kümeleme

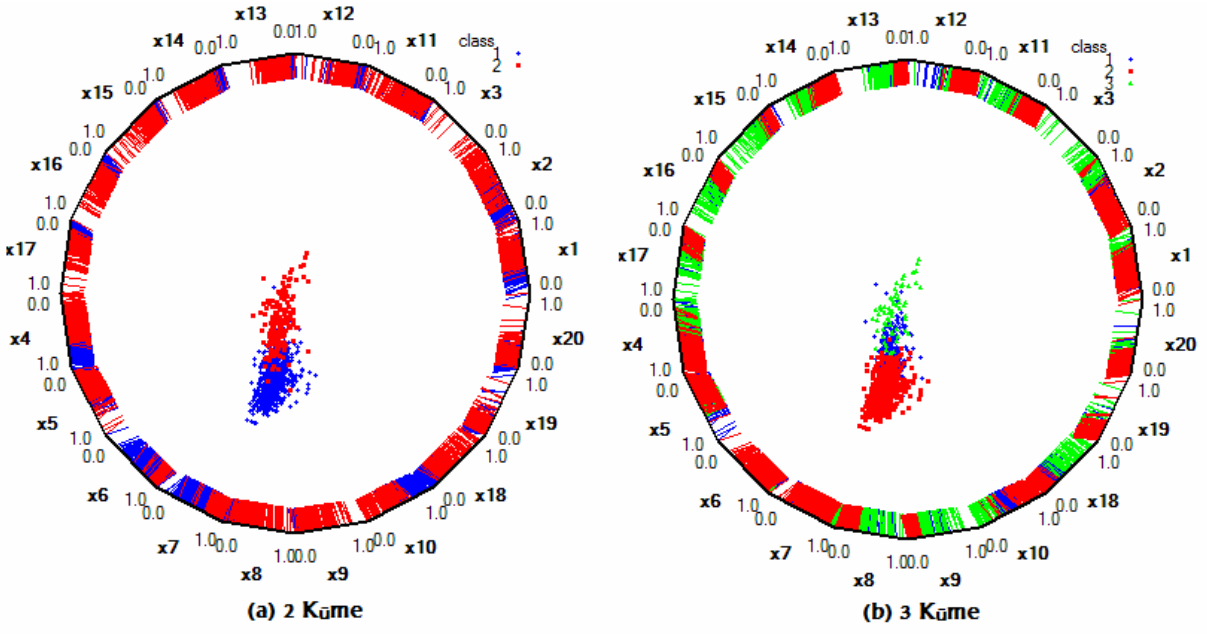
yönteminin farklı küme sonuçları için çizilen Şekil 5.15 ve Şekil 5.16' daki matris grafiklerine bakıldığında sadece Şekil 5.15' de 3 kümeye ayrılan ilçe veri setinin kendi içinde homojen kümelenmeler sergilediği gözlenmişti. Bu 3 kümeden 2 tanesinin birbirine benzediği gözlenmiştir. Bu 2 benzeyen kümenin birleştirilmesi suretiyle 3 küme 2 kümeye indirgenerek SOM kümeleme yönteminin başarısı arttırılabilir.



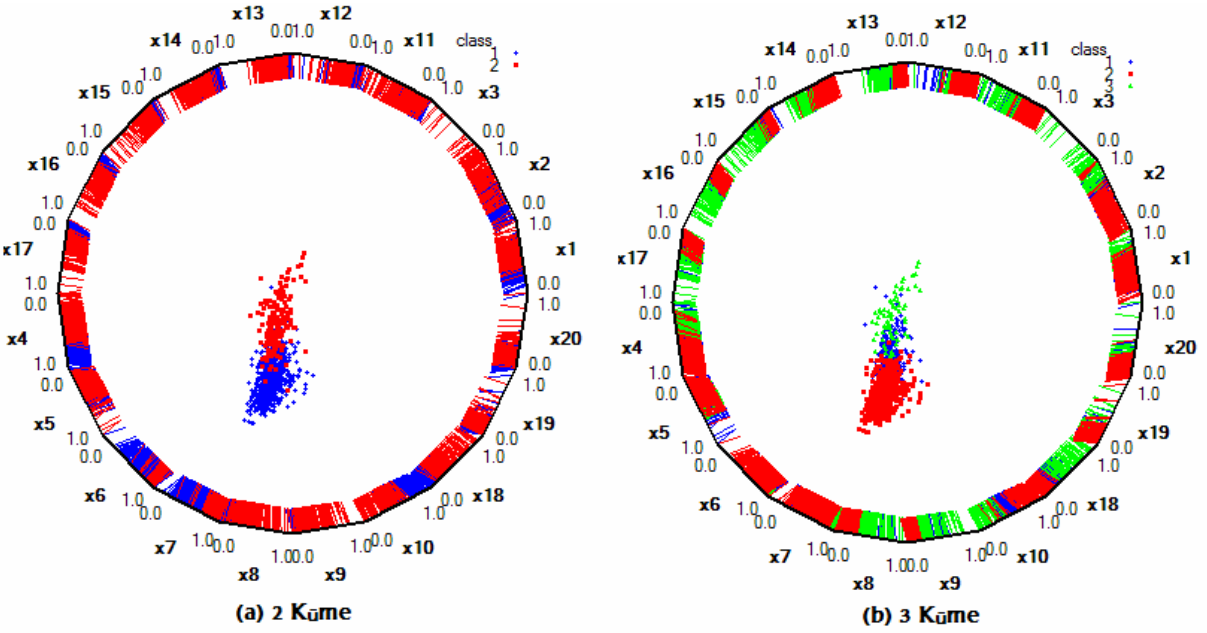
Şekil 5.17 SOM kümeleme yöntemiyle 2 ve 3 kümeye ayrılan ilçe veri setinin küme sonuçlarını gösteren matris grafiği

Şekil 5.17 (a)' da SOM kümeleme yöntemiyle 2 kümeye ayrılan ilçe veri setini, Şekil 5.17 (b)' de SOM kümeleme yöntemiyle önce 3 kümeli daha sonra kümelerin birleştirilmesiyle elde edilen 2 kümeli ilçe veri setini gösteren matris grafiği bulunmaktadır. Şekil 5.17 (b)' deki matris grafiği Şekil 5.17 (a)' daki matris grafiğine göre kendi içinde daha homojen kendi aralarında daha heterojen bir yapı sergilemektedir. Dolayısıyla Şekil 5.17 (b)' nin elde edilmesinde kullanılan kümeleme yönteminin daha başarılı olduğu sonucu ortaya çıkmaktadır.

4 farklı kümeleme yönteminden elde edilen küme sonuçlarının matris grafiklerinde gösterilmesi sonucu hiyerarşik kümeleme yöntemlerinin ilçe veri setini kümelemede başarılı olamadığı gözlenmiştir. K-ortalamalar ve SOM kümeleme yöntemleriyle ilçe veri seti 2 kümeye iyi bir şekilde kümelenmektedir. Ancak k-ortalamalar ve SOM kümeleme yöntemlerinin küme biçim, hacim ve yoğunluklarına gösterdiği zayıflıktan dolayı ilçe veri setinin kümelenmesinde bazı sorunlar olabileceği düşünülmüştür. Zaten matris grafiklerinden de 2 kümeli ilçe veri setinin küme kalitelerini bozan bazı ilçelerin bulunduğu gözlenmişti. Her iki kümeleme yönteminin başarısını arttırabilmek için ilçe veri seti önce 3 kümeye ayrılmış daha sonra benzer kümelerin birleştirilmesi suretiyle 2 küme elde edilmiştir. Bu sayede her iki kümeleme yönteminin başarısı arttırılmıştır.



Şekil 5.18 K-ortalamalar kümeleme yöntemiyle elde edilen 2 ve 3 kümenin dağılımını gösteren PolyViz grafiği



Şekil 5.19 SOM kümeleme yöntemiyle edilen 2 ve 3 kümenin dağılımını gösteren PolyViz grafiği

Şekil 5.18 ve Şekil 5.19' da k-ortalamalar ve SOM kümeleme yöntemleriyle elde edilen 2 ve 3 kümenin dağılımları PolyViz grafikleriyle gösterilmiştir. Her iki şekil de 2 ve 3 kümenin, küme biçim, hacim ve yoğunluklara duyarlı olan k-ortalamalar ve SOM kümeleme sonuçlarına nasıl etki ettiğini göstermektedir. Şekillere göre 2 kümeye ayrılan ilçe veri setininde her bir küme eşit hacimde kümelenmiştir; ancak 3 kümeye ayrılan ilçe veri setinin

düşük yoğunluk ve küçük hacimdeki kümeleri saptadığı gözlenmiştir. Dolayısıyla veri setini 3 kümeye ayırmak daha doğrudur.

5.5 Küme Sonuçları

Çalışmanın bundan sonraki aşamalarında k-ortalamalar ve SOM kümeleme yöntemlerinin bulduğu 2 küme denilince önce 3 kümeye ayrılmış daha sonra benzer kümelerin birleştirilmesiyle elde edilen 2 küme akla gelecektir. K-ortalamalar ve SOM kümeleme yöntemlerinin bulduğu bu 2 küme sonuçları ilçe, şehir ve bölge bazında EK 4‘ de verilmiştir.

Dört farklı kümeleme yöntemiyle iki kümeye ayrılan 911 ilçe veri setinin sonuçları Tablo 5.9’ da verilmektedir.

Tablo 5.9 Dört farklı kümeleme yönteminden elde edilen küme sonuçları

İlçeler	Tek Bağlantılı	Tam Bağlantılı	K-Ortalamalar	SOM
Küme 1	1	29	304	312
Küme 2	910	882	607	599
Genel Toplam	911	911	911	911

K-ortalamalar ve SOM kümeleme yöntemlerinin bir birine benzer sonuçlar ürettiği Tablo 5.9’ dan gözlenmektedir. Tablo 5.10’ da k-ortalamalar ve SOM kümeleme sonuçlarının çapraz tablosu bulunmaktadır. İki küme yöntemi arasında toplam 8 ilçe farklı kümelenmiştir.

Tablo 5.10 K-ortalamalar ve SOM küme sonuçları

İlçeler	SOM		
K-Ortalamalar	Küme 1	Küme 2	Genel Toplam
Küme 1	304	0	304
Küme 2	8	599	607
Genel Toplam	312	599	911

Tablo 5.11’ de k-ortalamalar ve Tablo 5.12’ de SOM kümeleme yöntemleriyle 2 kümeye ayrılan ilçelerin istatistikleri bulunmaktadır. İstatistiklerden görüldüğü gibi 1. kümenin X4 (ücretli çalışan kadınların toplam istihdama oranı) ve X10 (tarım kesiminde çalışanların toplam istihdama oranı) değişken ortalamaları haricindeki bütün değişken ortalamaları 2. kümenin değişken ortalamalarından yüksektir. Kümeleme yaptığımız değişkenlerden sadece X10 ve X5 (işsizlik oranı) değişkenleri gelişmişlikle ters orantılıdır. Dolayısıyla X4 ve X5 değişkenleri dışında 1. küme 2. kümeye göre daha gelişmiş durumdadır. Buradan hareketle 1. kümeye gelişmiş, 2. kümeye daha az gelişmiş ilçeler topluluğu diyebiliriz.

Tablo 5.11 K-ortalamalar kümeleme yöntemiyle kümelenen ilçelerin istatistikleri

Tanımlayıcı İstatistikler										
K-Ortalamalar	1. Küme					2. Küme				
Değişkenler	İlçe Sayısı	Minimum	Maksimum	Ortalama	Standart Sapma	İlçe Sayısı	Minimum	Maksimum	Ortalama	Standart Spama
X1	304	-52,28	105,26	20,33	22,12	607	-81,11	64,23	-1,07	19,41
X2	304	9,34	100,00	66,86	19,29	607	8,04	76,42	36,10	13,09
X3	304	5,55	35143,71	1312,80	4816,80	607	2,75	493,45	56,11	58,41
X4	304	10,66	46,23	32,61	7,60	607	15,58	58,92	45,72	5,01
X5	304	2,21	29,12	9,50	4,85	607	0,88	21,51	5,20	3,14
X6	304	81,41	98,48	95,20	2,50	607	60,58	96,85	90,37	5,25
X7	304	41,94	94,74	83,62	7,85	607	19,81	88,85	71,66	11,67
X8	304	1,91	17,97	6,67	2,82	607	1,22	6,55	3,38	1,01
X9	304	0,60	17,27	3,86	2,63	607	0,11	3,59	1,20	0,59
X10	304	0,21	73,50	42,96	20,55	607	20,62	96,13	78,15	8,71
X11	304	0,24	51,23	13,39	10,83	607	0,05	21,84	3,17	2,91
X12	304	1,48	22,27	5,58	2,91	607	0,39	10,97	2,62	1,48
X13	304	1,96	48,54	10,13	5,85	607	0,32	9,06	3,11	1,56
X14	304	1,04	11,03	3,64	1,75	607	0,26	6,74	1,37	0,68
X15	304	0,64	13,96	2,90	2,25	607	0,00	3,21	0,73	0,37
X16	304	2,56	23,06	7,65	3,29	607	0,98	6,68	3,24	0,99
X17	304	0,41	5,39	1,48	0,78	607	0,07	1,79	0,52	0,24
X18	304	1,83	18,95	5,95	3,33	607	0,39	5,41	1,95	0,68
X19	304	26,74	1875,47	244,01	188,34	607	15,21	1812,29	135,47	144,39
X20	304	0,00	73,15	11,57	8,05	607	0,00	36,14	7,71	4,85

Tablo 5.12 SOM kümeleme yöntemiyle kümelenen ilçelerin istatistikleri

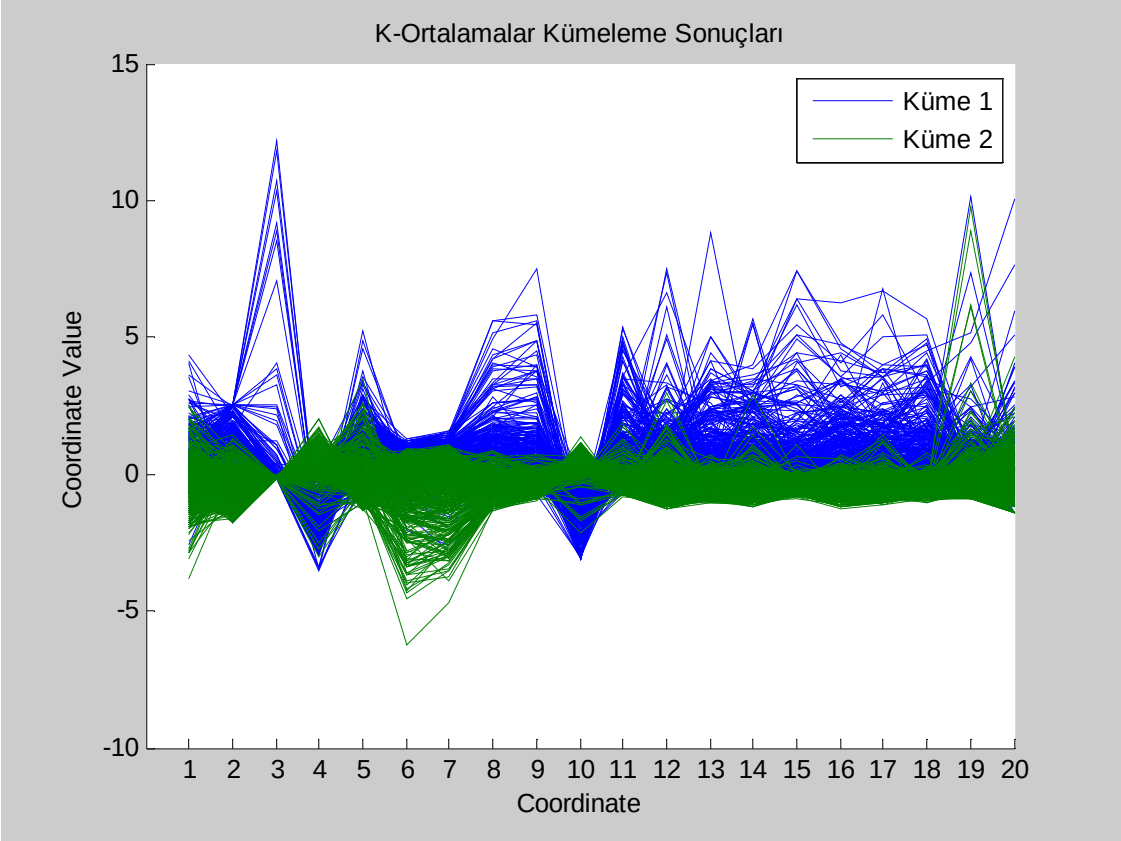
Tanımlayıcı İstatistikler										
SOM	1. Küme					2. Küme				
Değişkenler	İlçe Sayısı	Minimum	Maksimum	Ortalama	Standart Sapma	İlçe Sayısı	Minimum	Maksimum	Ortalama	Standart Sapma
X1	312	-81,11	105,26	19,65	22,80	599	-63,79	64,23	-1,00	19,19
X2	312	9,34	100,00	66,43	19,29	599	8,04	76,42	35,91	13,03
X3	312	2,75	35143,71	1280,44	4758,64	599	4,05	493,45	56,18	58,61
X4	312	10,66	46,23	32,75	7,61	599	15,58	58,92	45,82	4,92
X5	312	2,21	29,12	9,44	4,82	599	0,88	21,51	5,17	3,13
X6	312	81,41	98,48	95,17	2,49	599	60,58	96,85	90,32	5,27
X7	312	41,94	94,74	83,48	7,87	599	19,81	88,85	71,58	11,70
X8	312	1,91	17,97	6,60	2,81	599	1,22	6,55	3,37	1,01
X9	312	0,60	17,27	3,81	2,62	599	0,11	3,59	1,19	0,59
X10	312	0,21	73,50	43,53	20,64	599	20,62	96,13	78,32	8,57
X11	312	0,24	51,23	13,18	10,78	599	0,05	21,84	3,14	2,91
X12	312	1,48	22,27	5,54	2,90	599	0,39	10,97	2,60	1,47
X13	312	1,96	48,54	10,00	5,84	599	0,32	9,06	3,09	1,55
X14	312	1,04	11,03	3,61	1,74	599	0,26	6,74	1,36	0,67
X15	312	0,64	13,96	2,86	2,23	599	0,00	3,21	0,72	0,36
X16	312	2,56	23,06	7,57	3,28	599	0,98	6,68	3,22	0,98
X17	312	0,41	5,39	1,46	0,78	599	0,07	1,79	0,51	0,24
X18	312	1,83	18,95	5,88	3,32	599	0,39	5,41	1,94	0,67
X19	312	26,74	1875,47	241,96	187,11	599	15,21	1812,29	135,08	144,80
X20	312	0,00	73,15	11,59	8,01	599	0,00	36,14	7,65	4,80

Tablo 5.11 ve Tablo 5.12’ de kümelerin değişken ortalamalarının bir birinden ayrıştığı gözlenir. ANOVA yaklaşımı ile de bu değişken ortalamalarının istatistikî olarak birbirlerinden farklı olduğu tespit edilebilir. Tablo 5.13’ de k-ortalamalar ve SOM kümeleme yöntemleriyle kümelenen ilçelerin değişkenleri için hesaplanan ANOVA tablosu bulunmaktadır. Tabloya göre 0.05 anlamlılık düzeyine göre kümelerde bulunan tüm değişken ortalamaları birbirinden istatistikî olarak faklıdır (değişkenler farklı ana kütlelerden gelmektedir) Dolayısıyla küme yapıları tüm değişkenler bazında farklılık göstermektedir.

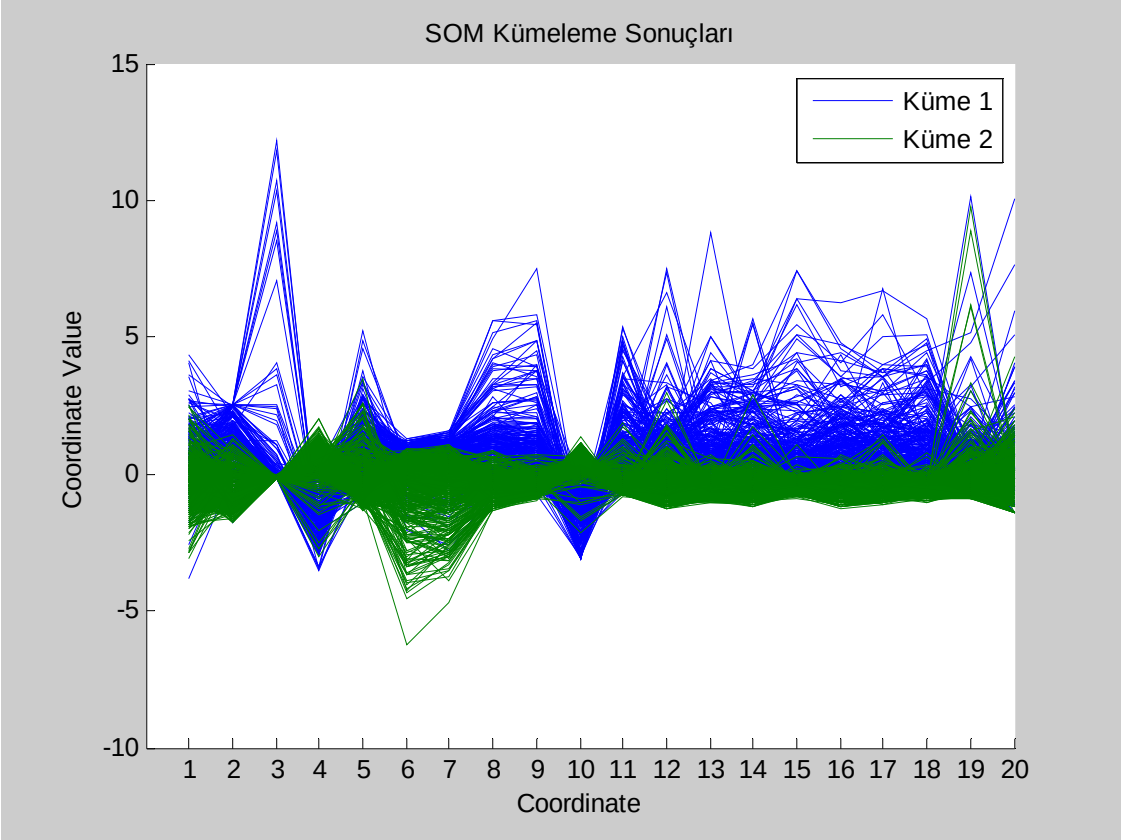
Tablo 5.13 Küme sonuçları için ANOVA

		K-Ortalamalar Küme Sonuclari Icin ANOVA					SOM Kume Sonuclari Icin ANOVA				
Değişkenler	İstatistikler	Sum of Squares	df	Mean Square	F	Sig.	Sum of Squares	df	Mean Square	F	Sig.
X1	Between Groups	92723,75	1,00	92723,75	223,86	0,00	87443,43	1,00	87443,43	208,19	0,00
	Within Groups	376509,43	909,00	414,20			381789,75	909,00	420,01		
	Total	469233,18	910,00				469233,18	910,00			
X2	Between Groups	191738,28	1,00	191738,28	804,62	0,00	191060,35	1,00	191060,35	799,28	0,00
	Within Groups	216610,69	909,00	238,30			217288,62	909,00	239,04		
	Total	408348,97	910,00				408348,97	910,00			
X3	Between Groups	319888292,05	1,00	319888292,05	41,35	0,00	307475386,36	1,00	307475386,36	39,68	0,00
	Within Groups	7032140704,03	909,00	7736128,39			7044553609,72	909,00	7749783,95		
	Total	7352028996,08	910,00				7352028996,08	910,00			
X4	Between Groups	34795,23	1,00	34795,23	966,37	0,00	35022,50	1,00	35022,50	979,48	0,00
	Within Groups	32729,52	909,00	36,01			32502,25	909,00	35,76		
	Total	67524,75	910,00				67524,75	910,00			
X5	Between Groups	3758,40	1,00	3758,40	261,19	0,00	3732,05	1,00	3732,05	258,84	0,00
	Within Groups	13080,14	909,00	14,39			13106,49	909,00	14,42		
	Total	16838,54	910,00				16838,54	910,00			
X6	Between Groups	4740,59	1,00	4740,59	231,63	0,00	4839,87	1,00	4839,87	237,75	0,00
	Within Groups	18603,40	909,00	20,47			18504,12	909,00	20,36		
	Total	23343,99	910,00				23343,99	910,00			
X7	Between Groups	28969,46	1,00	28969,46	260,22	0,00	29053,49	1,00	29053,49	261,19	0,00
	Within Groups	101197,98	909,00	111,33			101113,95	909,00	111,24		
	Total	130167,44	910,00				130167,44	910,00			
X8	Between Groups	2192,74	1,00	2192,74	659,21	0,00	2150,06	1,00	2150,06	637,38	0,00
	Within Groups	3023,63	909,00	3,33			3066,30	909,00	3,37		
	Total	5216,36	910,00				5216,36	910,00			
X9	Between Groups	1435,39	1,00	1435,39	563,34	0,00	1406,96	1,00	1406,96	545,49	0,00
	Within Groups	2316,11	909,00	2,55			2344,54	909,00	2,58		
	Total	3751,50	910,00				3751,50	910,00			
X10	Between Groups	250794,40	1,00	250794,40	1310,83	0,00	248406,28	1,00	248406,28	1280,77	0,00
	Within Groups	173913,77	909,00	191,32			176301,89	909,00	193,95		
	Total	424708,18	910,00				424708,18	910,00			
X11	Between Groups	21157,52	1,00	21157,52	472,45	0,00	20681,58	1,00	20681,58	456,49	0,00
	Within Groups	40707,31	909,00	44,78			41183,25	909,00	45,31		
	Total	61864,82	910,00				61864,82	910,00			
X12	Between Groups	1781,02	1,00	1781,02	415,86	0,00	1768,07	1,00	1768,07	411,47	0,00
	Within Groups	3893,02	909,00	4,28			3905,97	909,00	4,30		
	Total	5674,04	910,00				5674,04	910,00			
X13	Between Groups	9978,41	1,00	9978,41	765,96	0,00	9795,61	1,00	9795,61	740,50	0,00
	Within Groups	11841,79	909,00	13,03			12024,59	909,00	13,23		
	Total	21820,20	910,00				21820,20	910,00			
X14	Between Groups	1040,68	1,00	1040,68	783,36	0,00	1039,99	1,00	1039,99	782,39	0,00
	Within Groups	1207,58	909,00	1,33			1208,28	909,00	1,33		
	Total	2248,26	910,00				2248,26	910,00			
X15	Between Groups	953,62	1,00	953,62	536,61	0,00	942,67	1,00	942,67	526,88	0,00
	Within Groups	1615,41	909,00	1,78			1626,35	909,00	1,79		
	Total	2569,02	910,00				2569,02	910,00			
X16	Between Groups	3950,84	1,00	3950,84	928,70	0,00	3889,92	1,00	3889,92	900,20	0,00
	Within Groups	3867,02	909,00	4,25			3927,94	909,00	4,32		
	Total	7817,86	910,00				7817,86	910,00			
X17	Between Groups	186,73	1,00	186,73	768,88	0,00	184,90	1,00	184,90	755,11	0,00
	Within Groups	220,76	909,00	0,24			222,59	909,00	0,24		
	Total	407,49	910,00				407,49	910,00			
X18	Between Groups	3239,14	1,00	3239,14	807,20	0,00	3190,95	1,00	3190,95	784,82	0,00
	Within Groups	3647,64	909,00	4,01			3695,83	909,00	4,07		
	Total	6886,78	910,00				6886,78	910,00			
X19	Between Groups	2386330,45	1,00	2386330,45	92,77	0,00	2343426,46	1,00	2343426,46	90,93	0,00
	Within Groups	23382893,38	909,00	25723,76			23425797,37	909,00	25770,95		
	Total	25769223,83	910,00				25769223,83	910,00			
X20	Between Groups	3011,23	1,00	3011,23	80,83	0,00	3186,40	1,00	3186,40	85,98	0,00
	Within Groups	33864,18	909,00	37,25			33689,01	909,00	37,06		
	Total	36875,41	910,00				36875,41	910,00			

Şekil 5.20 ve Şekil 5.21’ de sırasıyla k-ortalamalar ve SOM kümeleme yöntemiyle ikiye ayrılan ilçe değişkenlerinin paralel koordinat grafiği bulunmaktadır. Paralele koordinat grafiklerinden kümeler arası değişkenlerin ayrışımı görsel olarak da gözlenmektedir.



Şekil 5.20 K-ortalamalar kümeleme yöntemiyle elde edilen 2 küme



Şekil 5.21 SOM kümeleme yöntemiyle elde edilen 2 küme

Tablo 5.14 ve Tablo 5.15’ de sırasıyla k-ortalamalar, SOM kümeleme yöntemleriyle bölgelere göre bulunan kümeler verilmiştir. Tablolardan da gözüktüğü gibi Doğu, Güneydoğu Anadolu ve Karadeniz bölgelerinin ilçelerinin çoğunluğu 2. kümede yer almaktadır. Marmara bölgesinin ilçeleri ise çoğunlukla 1. kümede yer almaktadır. Bölgelerin ilçelerinin çoğunluk itibariyle yer aldıkları bölgelere göre küme sınıfları Marmara Bölgesi için 1. küme, Akdeniz, Doğu Anadolu, Ege, Güneydoğu Anadolu, İç Anadolu ve Karadeniz Bölgeleri içinse 2. kümedir.

Tablo 5.14 K-ortalamalar kümeleme yöntemine göre bölgelere göre kümeler

İlçeler	k-ortalamalar					
Bölgeler	Küme 1	Küme 2	Küme 1 (%)	Küme 2 (%)	Küme Sınıfı	Genel Toplam
Akdeniz	50	54	48,08%	51,92%	2. Küme	104
Doğu Anadolu	26	138	15,85%	84,15%	2. Küme	164
Ege	37	79	31,90%	68,10%	2. Küme	116
Güneydoğu Anadolu	8	41	16,33%	83,67%	2. Küme	49
İç Anadolu	55	97	36,18%	63,82%	2. Küme	152
Karadeniz	40	151	20,94%	79,06%	2. Küme	191
Marmara	88	47	65,19%	34,81%	1. Küme	135
Genel Toplam	304	607	33,37%	66,63%	2. Küme	911

Tablo 5.15 SOM kümeleme yöntemine göre bölgelere göre kümeler

İlçeler	SOM					
Bölgeler	Küme 1	Küme 2	Küme 1 (%)	Küme 2 (%)	Küme Sınıfı	Genel Toplam
Akdeniz	51	53	49,04%	50,96%	2. Küme	104
Doğu Anadolu	27	137	16,46%	83,54%	2. Küme	164
Ege	38	78	32,76%	67,24%	2. Küme	116
Güneydoğu Anadolu	8	41	16,33%	83,67%	2. Küme	49
İç Anadolu	57	95	37,50%	62,50%	2. Küme	152
Karadeniz	42	149	21,99%	78,01%	2. Küme	191
Marmara	89	46	65,93%	34,07%	1. Küme	135
Genel Toplam	312	599	34,25%	65,75%	2. Küme	911

Büyük şehir belediyeleri bulunan illerinin kümelere göre dağılımı Tablo 5.16’ da ki gibidir. Tabloya göre büyükşehir belediyeleri bulunduran illerden sadece İstanbul, Ankara, İzmir ve Bursa ilçelerinin büyük bir çoğunluğu 1. kümededir. Bu dört il Türkiye’ nin en büyük 4 ili olarak ta bilinmektedir. Her iki kümeleme yöntemine göre de ilk 4 büyük şehir ilçeleri 1. kümenin yaklaşık % 20’sinden fazlasını oluşturmaktadır. Büyük şehir belediyeleri barındıran illerin 1. kümenin ve 2. kümenin sırasıyla yaklaşık % 35, %20’ sini oluşturmaktadır. Buradan hareketle Türkiye’nin ilçelerinin bölgelere göre dağılımın homojen olmadığı ve Marmara

bölgesi, İzmir ve Ankara ilçelerinin gelişmiş ilçeler gurubunda yer aldığı, diğer bölgelerinse ilçeler itibariyle gelişmemiş olduğunu söyleyebiliriz.

Tablo 5.16 K-ortalamalar ve SOM kümeleme yöntemlerine göre büyük şehirlerin durumu

İlçeler		k-ortalamalar		
Bölgeler	Şehirler	Küme 1	Küme 2	Toplam
Akdeniz	Adana	5	8	13
	Antalya	6	9	15
Doğu Anadolu	Erzirum	1	18	19
Ege	İzmir	20	7	27
Güneydoğu Anadolu	Diyarbakır	1	13	14
İç Anadolu	Ankara	19	4	23
	Eskişehir	5	8	13
	Kayseri	5	11	16
	Konya	7	24	31
Karadeniz	Samsun	1	14	15
Marmara	Bursa	10	7	17
	İstanbul	28		28
Toplam		108	123	231
Tüm İlçeler İçindeki Oranı		35,53%	20,26%	25,36%
İlçeler		SOM		
Bölgeler	Şehirler	Küme 1	Küme 2	Toplam
Akdeniz	Adana	6	7	13
	Antalya	6	9	15
Doğu Anadolu	Erzirum	**1**	**18**	**19**
Ege	İzmir	20	7	27
Güneydoğu Anadolu	Diyarbakır	1	13	14
İç Anadolu	Ankara	19	4	23
	Eskişehir	5	8	13
	Kayseri	5	11	16
	Konya	8	23	31
Karadeniz	Samsun	1	14	15
Marmara	Bursa	10	7	17
	İstanbul	28		28
Toplam		110	121	231
Tüm İlçeler İçindeki Oranı		35,26%	20,20%	25,36%

6. SONUÇLAR VE ÖNERİLER

Verilerin dijital ortamlarda saklanmasıyla birlikte yeryüzündeki bilgi miktarı sürekli artmaktadır. İnsanlar, bu verilerin ticari anlamda rakiplerine karşı üstünlük sağlamalarını sağlayacak, bilimsel anlamda çalışmalara yeni açılımlar getirebilecek değerli bilgiler taşıyan potansiyel kaynaklar olduklarına inandıkları için verileri toplamakta ve saklamaktadırlar. Ancak biriken veri gerçek anlamda bilgiye dönüşebilecek midir? Bu soruya olumlu yanıt verebilmek için veri madenciliği kavramı ortaya atılmıştır. Veri madenciliği dijital ortamlarda saklanan büyük verilerin bilgiye dönüştürülmesi sürecidir.

Birçok veri madenciliği uygulamasında verilerin birbiri ile olan ilişkilerinin iyi anlaşılması büyük önem taşır. Verilerin grafiksel bir formda temsil edilmesi analizcinin veri yapılarını anlamasını kolaylaştırır. Ancak veri madenciliği teknikleri çok fazla sayıda kayıt ve çok fazla sayıda boyuttan oluşan veri yığınlarıyla uğraşırlar. Bu gibi sebeplerden dolayı veri görselleştirme teknikleri, ekran çözünürlüğü, insan algılama sistemi gibi sınırlardan dolayı başarılı olamayabilirler. Bu tez çalışmasında bu gibi sınırları ortadan kaldırabilmek için çeşitli yeni görselleştirme teknikleri, çok boyutlu veri uzayında görselleştirme konusunda karşılaşılan sorunlara değinilerek avantaj ve dezavantajlarıyla tanıtılmıştır. Bu yeni görselleştirme teknikleri görsel algılama sisteminin gücünden yararlanarak veri içindeki ilişkileri keşfetmeye, kümeleme sürecine rehberlik etmeye ve kümeleme sonucunun kalitesini değerlendirmeye olanak sağlar. Bu sayede aşırı değerler, uygun küme sayısı ve kümeleme algoritmaları insan algı sistemiyle keşfedilebilir.

Tez kapsamında 2000 yılı verileri kullanılarak 81 ildeki 918 ilçe 7 coğrafi bölge bazında ele alınarak sosyoekonomik özelliklerine göre, görsel veri madenciliği yöntemleri yardımıyla kümelenmiştir. Kümeleme algoritmalarının kullanılmasından önce temel bileşenler analiziyle birlikte kullanılan Andrews eğrileriyle aşırı değerler insan algılama sisteminin yardımıyla tespit edilmiştir. Andrwes eğrileriyle İstanbul Eminönü, İstanbul Beşiktaş, İstanbul Bakırköy, Anakara Çankaya, İstanbul Kadıköy, İzmir Aliağa ve Mardin Yeşilli ilçeleri aşırı değer olarak tespit edilmiştir. Tespit edilen aşırı değerler ilçe veri setinden ayıklanarak kümeleme işlemine devam edilmiştir.

Uygun küme sayısı kümeleme analizlerinde büyük öneme sahiptir. Bu sebepten dolayı kümeleme analizlerine başlamadan önce çok boyutlu veri yapılarını göstermede kullanılan PolyViz grafikleriyle uygun küme sayısının 2 olabileceğine yönelik ipuçları elde edilmiştir. Ayrıca PolyViz grafiğiyle potansiyel küme yapılarının biçimlerinin kürsel olmadığı, yoğunluk ve hacimlerinin faklı olduğu keşfedilmiştir. Daha sonra tek bağlantılı hiyerarşik, tam

bağlantılı hiyerarşik, k-ortalamalar ve SOM kümeleme yöntemleriyle elde edilen 10 küme için Silhouette (S), Davies-Bouldin (DB), Dunn (D), Calinski ve Harabasz (CH), Krzanowski Lai (KL) ve Hartigan (H) küme doğruluk (cluster validity) endeksleriyle uygun küme sayısına yönelik matematiksel sonuçlar elde edilmiştir. Ancak farklı küme doğruluk endeksleri faklı kümeleme algoritmaları için uygun küme sayısına yönelik farklı yanıtlar vermektedir. Bu yüzden farklı kümeleme algoritmalarının farklı küme sayıları için elde edilen sonuçları matris grafikleriyle gösterilerek uygun küme sayı insan algılama sistemi yardımıyla keşfedilmeye çalışılmıştır. Matris grafiklerine göre tek bağlantılı hiyerarşik, tam bağlantılı hiyerarşik kümeleme yöntemleriyle belirgin küme yapıları bulunamamış, k-ortalamalar ve SOM kümeleme yöntemlerine göre uygun küme sayısı 2 olarak tespit edilmiştir. Bu iki küme için çizilen matris grafiklerine baktığımızda küme içi homojenliği ve kümeler arası heterojenliği bozan bazı noktaların bulunduğu gözlenmiştir. Yani k-ortalamalar ve SOM kümeleme yöntemleriyle 2' ye ayrılan ilçe veri setinin iyi kümelenmesini bozan bazı ilçelerin bulunduğu anlaşılmıştır.

PolyViz grafiklerinde küme potansiyeli taşıyan veri dağılımlarının küresel olmadığı, yoğunluk ve hacimlerinin farklı olduğu tespit edilmişti. Bu da k-ortalamalar ve SOM kümeleme yöntemlerinin başarısını olumsuz yönde etkilemektedir. Bu sorunu ortadan kaldırabilmek için ilçe veri setinin 3 kümeye ayrılması yoluna gidilmiş ve daha sonra bu 3 kümeden benzer olan kümelerin birleştirilmesiyle kümeleme işlemi başarıyla sonlandırılmıştır.

Sonuç olarak ilçe veri setini kümeleme de hiyerarşik kümeleme yöntemlerinin başarısız olduğu, k-ortalamalar ve SOM kümeleme yöntemlerinin birbirine benzer 2 başarılı küme elde ettiği gözlenmiştir. Elde edilen bu kümeler sosyoekonomik düzeylerine göre ilçeleri gelişmiş ve daha az gelişmiş olarak göstermektedir.

K-ortalamalar ve SOM kümeleme yöntemleriyle bulunan küme sonuçlarına göre kendi içinde homojen bölgeler bulunmamaktadır. Her iki kümeleme yöntemiyle elde edilen gelişmiş ilçeler kümesinde en fazla Marmara bölgesinin ilçeleri bulunmaktadır. Gelişmemiş ilçeler kümesinde ise en fazla Doğu ve Güneydoğu Anadolu bölgelerinin ilçeleri bulunmaktadır. Gelişmiş ilçeler kümesine şehir bazında bakıldığında İstanbul, İzmir, Ankara ve Bursa şehirlerinin gelişmiş ilçeler kümesinin çoğunluğunu oluşturduğu gözlenmiştir. Genel bir değerlendirme yapıldığında sosyoekonomik özelliklere göre Türkiye' de bölgeler itibariyle homojen bir dağılım olmadığı gözlenmiştir. Marmara Bölgesinin tek başına homojen bir yapı sergilediği ve Türkiye' nin diğer bölgelerinden daha gelişmiş olduğu tespit edilmiştir.

En gelişmiş bölge olan Marmara bölgesinde sanayinin yaygın olması, okullaşma ve buna bağlı olarak okuryazar oranının yüksek olması önemli bir faktör iken yine en az gelişmiş bölge olan Doğu ve Güneydoğu Anadolu Bölgelerine yapılan yatırımların azlığı, kız çocuklarının okula gönderilmemesi gerçeği ve okullaşma oranının da düşük olması da bu bölgenin gelişmesini kısıtlayıcı bir faktördür. Türkiye'nin bölgelerarası gelişmişlik farkını düzeltmek için öncelikle eğitim düzeyini yükseltici, işsizliği önleyici ve ekonomik refahı arttırıcı önlemler alınması gerekliliği bu tez çalışmasından çıkan diğer bir sonuçtur.

Tez çalışması süresince kazanılan deneyimler ve elde edilen bilgiler ışığında, veri madenciliği çalışmalarında, grafiksel yöntemlerin, kullanıcıya analiz aşamasında aşırı değerlerin tespitinde, doğru küme sayısı ve doğru kümeleme algoritmalarının seçiminde kullanılması önerilmektedir. Bu şekilde kümeleme analizleri daha efektif hale getirilebilinir.

KAYNAKLAR

Alpar, R., (2003), Uygulamalı Çok Değişkenli İstatistiksel Yöntemlere Giriş 1, Nobel Kitabevi, Ankara.

Alpdoğan Y., (2007), Kendinden Düzenlenen Haritalar ile Doküman Sınıflandırma, Yüksek Lisans Tezi, FEN Bilimleri Enstitüsü, Gazi Üniversitesi, Ankara.

Amasyalı, M.F., (2006), Makine Öğrenmesine Giriş, http://www.ce.yildiz.edu.tr/mygetfile.php?id=868

Amasyalı, M.F., (2008), Yeni Makine Öğrenmesi Metotları Ve İlaç Tasarımına Uygulamaları, Doktora Tezi, Fen Bilimleri Enstitüsü, Yıldız Teknik Üniversitesi, İstanbul

Ankerst, M. (2000), Visual Data Mining, Doktora Tezi, Institute for Computer Science, University of Munchen, Berlin

Aydoğan, F., (2003), E-Ticarette Veri Madenciliği Yaklaşımlarıyla Müşteriye Hizmet Sunan Akıllı Modüllerin Tasarımı ve Gerçekleştirimi, Yüksek Lisans Tezi, FEN Bilimleri Enstitüsü, Hacettepe Üniversitesi, Ankara.

Bilen, Ö., (2004), ÖSS Sınav Sonuçlarının Okul Bazında Veri Madenciliği İle İncelenmesi, Yüksek Lisans Tezi, FEN Bilimleri Enstitüsü, Yıldız Teknik Üniversitesi, İstanbul (Yayınlanmamış).

Bilgin, T. ve Çamurcu, A., (2007), "Çok Boyutlu Veri Görselleştirme Teknikleri", Akademik Bileşim 2008, 30 Ocak–1 Şubat 2007, Çanakkale Onsekiz Mart Üniversitesi, Çanakkale

Bilgin, T.T., (2008), Çok Boyutlu Uzayda Görsel Veri Madenciliği İçin Üç Yeni Çatı Tasarımı v e Uygulamaları, Doktora Tezi, Fen Bilimleri Enstitüsü, Marmara Üniversitesi, İstanbul

Cai, W. ve Lı, L., (2004), "Anomaly Detection Using TCP Header Information" http://www.galaxy.gmu.edu/stats/syllabi/stat753/Cai_Li_Stat753_paper.pdf

Chen, K. ve Liu, L., (2006), "IVABRATE: Interactive Visualization-Based Framework for Clustering Large Datasets", ACM Transactions on Infarmation Systems, 24(2):245-294.

DİE, "Genel Nüfus Sayımı, İdari Bölünüş", İstanbul, 2000

Ding, Y. Ve Harrison, R. F., (2007), "Relational visual cluster validaty (RVCV)", Pattern Recognition Letters, 28:2071-2079.

Everitt, B. S. ve Nicholls, P., (1975), "Visual Techniques for Representing Multivariate Data", The Statistician, 1(24):37-49

Garcia-Osorio, C., (2005), "Visualization of High – Dimensional Data via Orthogonal Curves ", Journal of Universal Computer Science, 11: 1806-1819

Grinstein, G., Trutschı, M. ve Cvek, U., (2001), "High-Dimensional Visualizations", http://www.cs.uml.edu/~mtrutsch/research/High-Dimensional_Visualizations-KDD2001-color.pdf

Hardle, W. ve Simar, L., (2003), Applied Multivariate Statistical Analysis, Method and Data Technologies, Berlin.

Hoffman, P.E., (1999), Table Visualizations: A Formal Model and Its Applicaitons, Doktora Tezi, Institute for Visualization and Perception Research, University of Massachusetts Lowell, USA

Keim D. A., (2000), “Designing Pixel-Oriented Visualization Techniques: Theory and Applications ”, IEEE on Transactions on Visualizations and Computer Graphics, 6: 59-77

Keim D. A., (2002), “Information Visualization and Visual Data Minind”, IEEE on Transactions on Visualizations and Computer Graphics, 8:100-107

Keim, D. ve Ward M., (2002), Visual Data Mining Techniques, http://infovis.uni-konstanz.de/papers/2002/onlyCh12.pdf

Koyuncugil, A.S., (2007), Veri Madenciliği ve Sermaye Piyasalarına Uygulanması, Sermaya Piyasası Kurulu Araştırma Raporu, İstanbul

Larose, D.T., (2005), Discovering Knowledge in Data, John Wiley & Sons, New Jersy.

Leban, G., Zupan, B., Vidmar, G. ve Bratko, Ivan, (2006), “VizRank: Data Visualization Guided by Machine Learning”, http://eprints.fri.uni-lj.si/archive/00000210/01/Leban-_VizRank-DMKD.pdf

Li, X., (2008), “Storm Clustering for Data –driven Weather Forecasting”, http://ams.confex.com/ams/pdfpapers/133557.pdf

Martinez, W.L. ve Martinez A. R., (2002), Computational Statistics Handbook with MATLAB, Boca Raton : CRC Press, USA.

Martinez, W.L. ve Martinez A. R., (2005), Exploratory Data Analysis with MATLAB, Boca Raton : CRC Press, USA.

Mufti, G.B., Bertrand, P. ve Moubarki L. (2005), “Determining the number of groups from measures of cluster stability ”, http://asmda2005.enst-bretagne.fr/IMG/pdf/proceedings/404.pdf

Oğur, U., (2004), Görsel ve Dinamik Veri Madenciliği Kullanarak Sesin Akustik Parametrelerinin İncelenmesi, Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü, Ankara Üniversitesi, Ankara.

Oğuzlar, A., (2003), “Veri Ön İşleme”, Erciyes Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi, 21: 67-76

Orhunbilge, N., (2000), Örnekleme Yöntemleri ve Hipotez Testleri, Avcıol Basım Yayın, İstanbul

Öğüt, S., (2005), “Veri Madenciliği Kavramı ve Gelişim Süreci”, Veri Madenciliği Paneli, 5 Mart 2005, İstanbul

Özdamar, K., (2004), Paket Programlar İle İstatistiksel Veri Analizi 1, Kaan Kitabevi, Eskişehir.

Özdamar, K., (2004), Paket Programlar İle İstatistiksel Veri Analizi 2, Kaan Kitabevi, Eskişehir.

Özekes, S. (2003), “Veri Madenciliği Modelleri ve Uygulama Alanları”, İstanbul Ticaret Üniversitesi Dergisi, 65-82

Özkan, Y., (2008), Veri Madencliği Yöntemleri, Papatya Yayıncılık Eğitim, İstanbul

Rencher, A.C., (2002), Methods of Multivariate Analysis, John Wiley & Sons, USA.

Saraçlı, S., Yılmaz, V. ve Kaygısız, Z., (2004), “Türkiye’de Beşeri Kalkınmışlığın Coğrafi Dağılımının Çok Değişkenli İstatistiksel Tekniklerle İncelenmesi”, 3. Ulusal Bilgi, Ekonomi ve Yönetim Kongresi 25–26 Kasım 2004, Osmangazi Üniversitesi İktisadi ve İdari Bilimler Fakültesi, Eskişehir.

Schatzmann, J., (2003), Using Self-Organizing Maps to Visualize Clusters and Trends in Multidimensional Datasets, Final Year Individual Project Report, Department of Computing Data Mining Group, Imperial College, London.

Sığırlı, D., Ediz B., Cangür, Ş., Ercan, İ. ve Kan, İ., (2006), "Türkiye ve Avrupa Birliği' ne Üye Ülkelerin Sağlık Düzeyi Ölçütlerinin Çok Boyutlu Ölçekleme Analizi İle İncelenmesi" İnönü Üniversitesi Tıp Fakültesi Dergisi, 13(2) 81-85

Spinelli, G.J., ve Zhou, Y., http://gis.esri.com/library/userconf/educ04/ papers/pap5000.pdf

Tan, P., Steinbach M. ve Kumar V., (2006), Introduction To Data Mining, Addison Wesley, USA.

Toledo, M.D.G., (2005), A Comparison in Cluster Validation Techniques, Yüksek Lisans Tezi, University of Puerto Rico Matematics Department, Puerto Rico

Unwin, A., Theus, Martin., Hofmann, Heike, (2006), Graphics of Large Datasets, Springer Science, Singapore

Uzunoğlu, M., Geçer, T., Eren, A. K., Kızıl, A. ve Onar, Ö. Ç., (2005), MATLAB İle Risk Yönetimi, Türkmen Kitabevi, İstanbul.

Vatansever, M., (2006), MATLAB İle İstatistik, Lisans Tezi, YTÜ Fen-Edebiyat Fakültesi İstatistik Bölümü, İstanbul (Yayınlanmamış).

Vatansever, M., (2007), MATLAB İle Regresyon, Lisans Tezi, YTÜ Fen-Edebiyat Fakültesi Matematik Bölümü, İstanbul (Yayınlanmamış).

Vesanto, J., Himberg J., Alhoniemi, E. ve Parhankangas J., (2000), SOM Toolbox for MATLAB 5, SOM Toolbox Team, Helsinki University of Technology, Finland.

Wijk, J. ve Wetering, H., (1999), "Cushion Treemaps: Visualization of Hierarchical Information", IEEE Symposium 25-26 Ekim 1999, San Fransisco

Zontul, M., Kaynar, O. ve Bircan, H., (2004), "SOM Tipimde Yapay Sinir Ağlarını Kullanarak Türkiye' nin İthalat Yaptığı Ülkelerin Kümelenmesi Üzerine Bir Çalışma",Cumhuriyet Üniversitesi, İktisadi ve İdari Bilimler Dergisi, 5(2):47-68.

İNTERNET KAYNAKLARI

http://www.mathworks.com/

http://www.mathworks.com/access/helpdesk/help/pdf_doc/stats/stats.pdf

http://www.mathworks.com/access/helpdesk/help/pdf_doc/nnet/nnet.pdf

http://www.mathworks.com/access/helpdesk/help/pdf_doc/bioinfo/bioinfo_ug.pdf

http://www.mathworks.com/access/helpdesk/help/pdf_doc/bioinfo/bioinfo_ref.pdf

http://www.mathworks.com/products/statistics/demos.html?file=/products/demos/shipping/stats/clusterdemo.html

http://www.sayisalyontemler.com/taxonomy/term/58

http://www.isletme.istanbul.edu.tr/surekli_yayinlar/dergiler/nisan2000/1.htm

http://www.r-project.org/

http://www.research.att.com/areas/stat/sgobi/

http://davis.wpi.edu/~xmdv/ downloadxmdv.html

ftp://www.galaxy.gmu.edu/pub/software/CrystalVisionDemo.exe

http://www.ailab.si/orange/downloads.asp

http://www.cis.hut.fi/projects/somtoolbox/download/

http://lib.stat.cmu.edu/matlab/

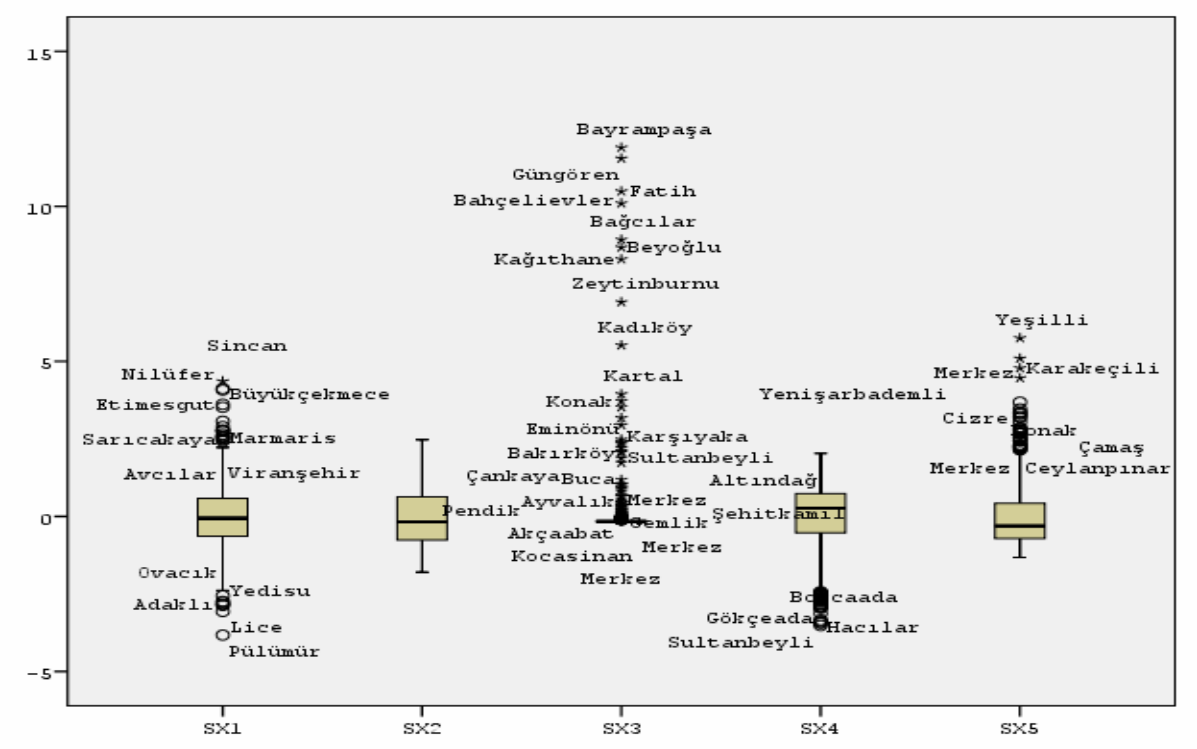
http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=14620&objectType=file

http://www.tbb.org.tr/net/subeler/

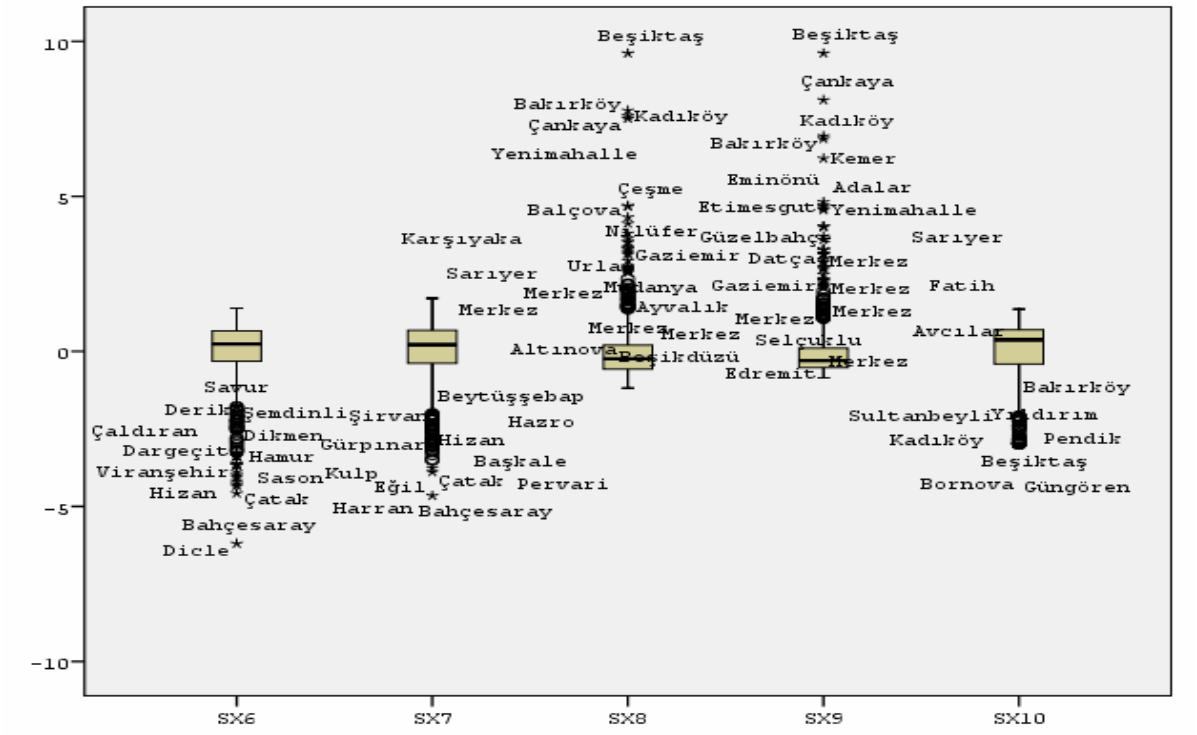
http://www.die.gov.tr/TURKISH/SONIST/GSYIH/241097t1.htm

EKLER

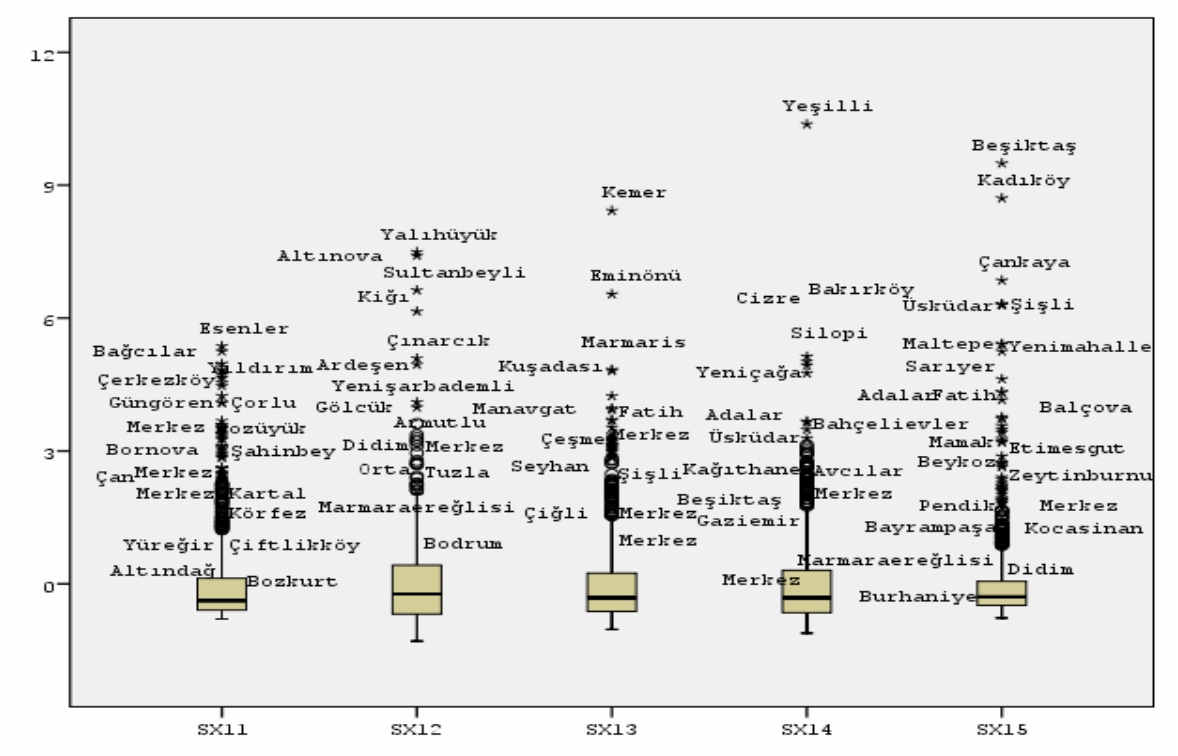
EK 1 918 İlçe Veri Setinde Bulunan Standartlaştırılmış Değişkenlere İlişkin Kutu Grafikleri



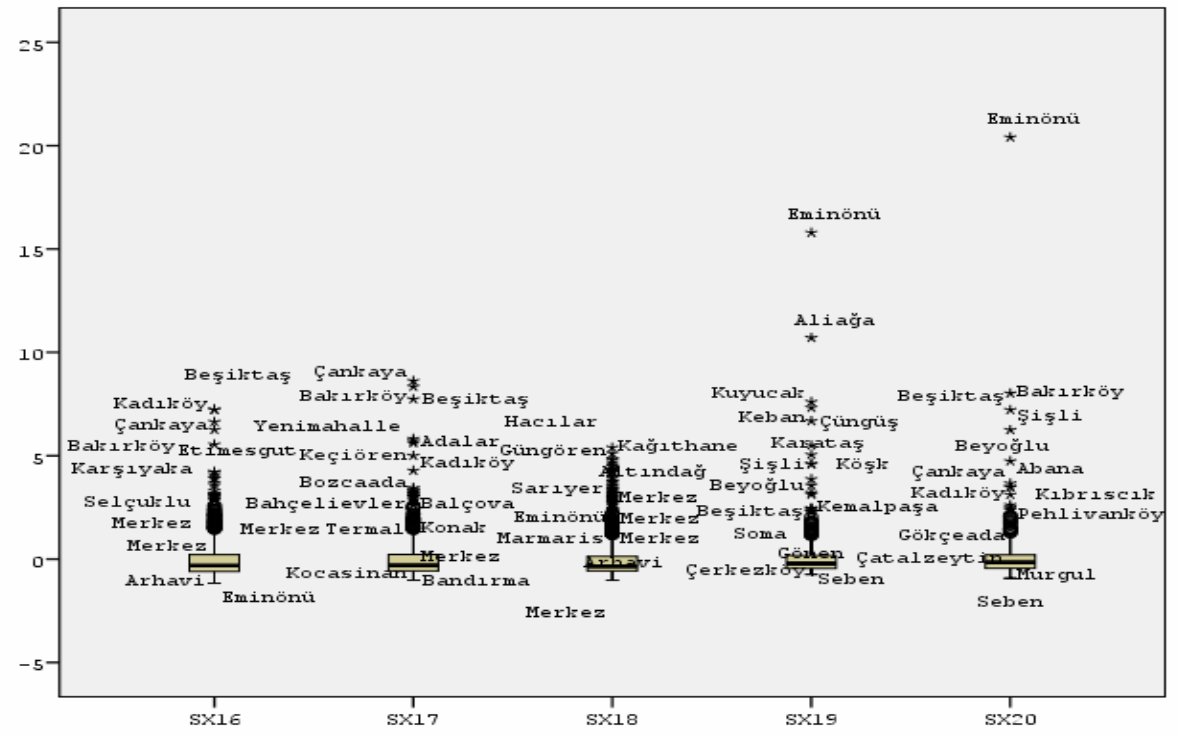
Şekil Ek 1.1 X1, X2, X3, X4 ve X5 standartlaştırılmış değişkenlere ilişkin kutu grafikleri



Şekil Ek 1.2 X6, X7, X8, X9 ve X10 standartlaştırılmış değişkenlere ilişkin kutu grafikleri

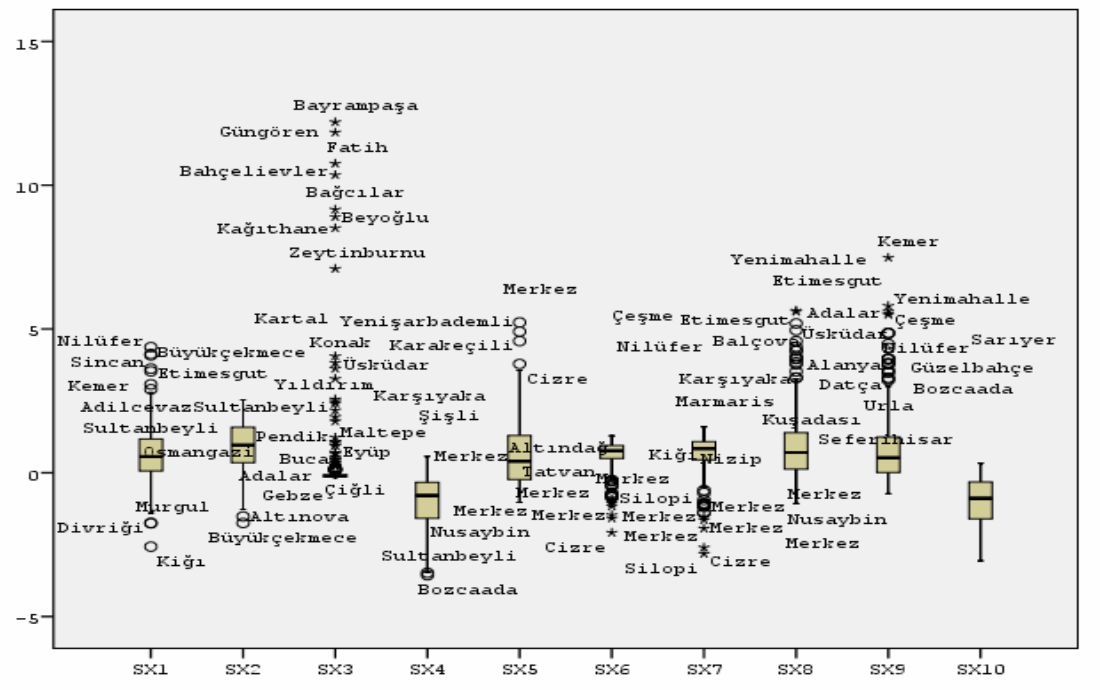


Şekil Ek 1.3 X11, X12, X13, X14 ve X15 standartlaştırılmış değişkenlere ilişkin kutu grafikleri

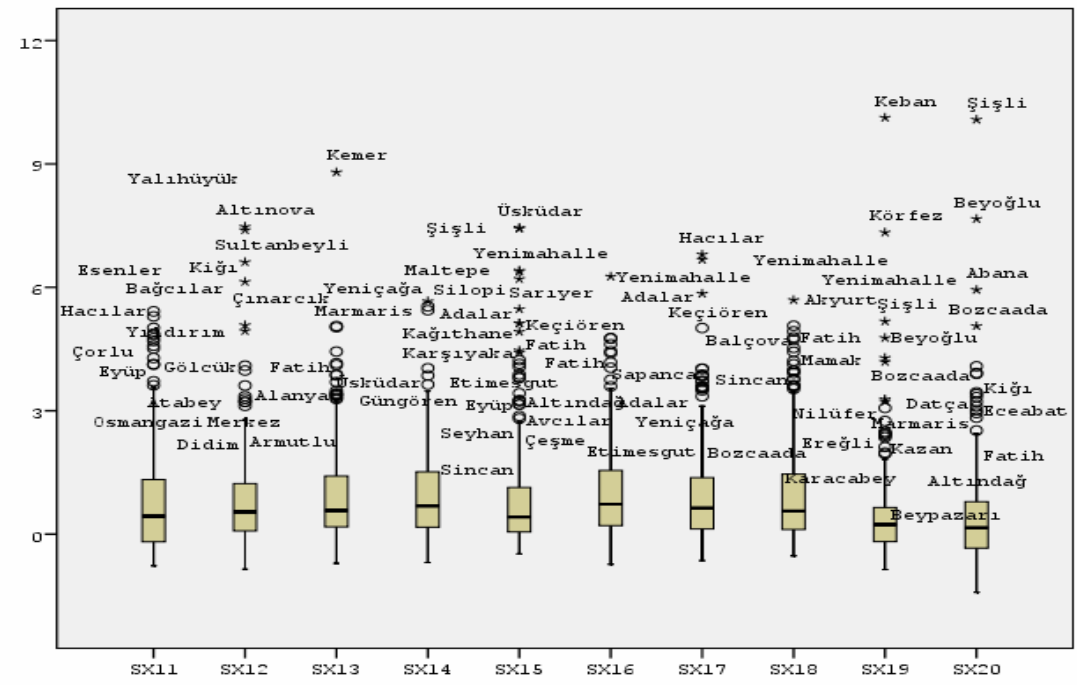


Şekil Ek 1.4 X16, X17, X18, X19 ve X20 standartlaştırılmış değişkenlere ilişkin kutu grafikler

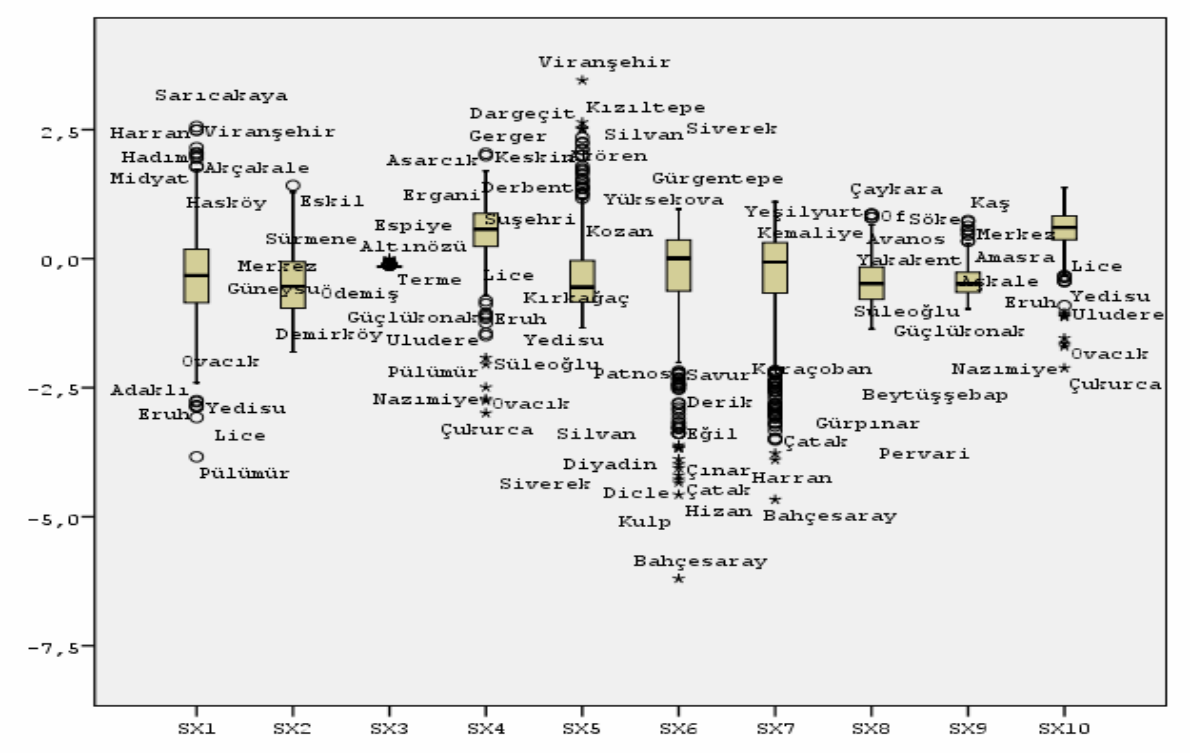
EK 2 K-Ortalamalar ve SOM Kümeleme Yöntemleriyle Önce 3' e Daha Sonra 2' ye Birleştirilen Kümelerde Bulunan Standartlaştırılmış Değişkenlere İlişkin Kutu Grafikleri



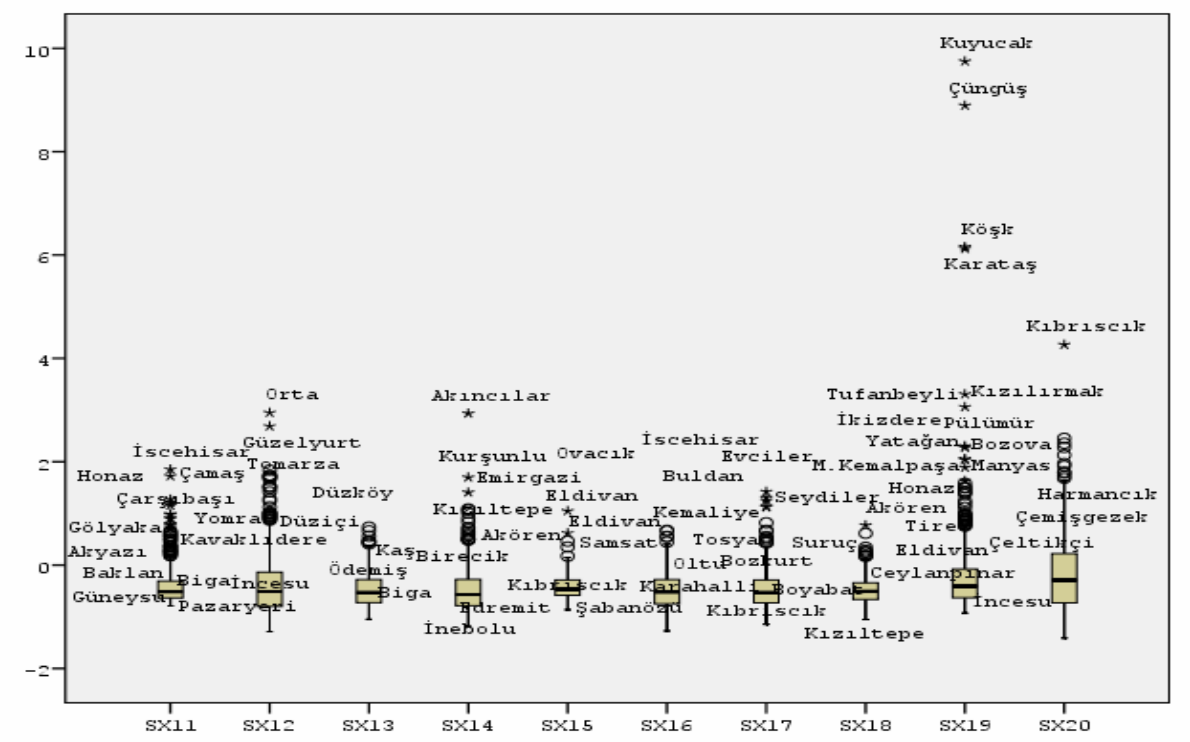
Şekil Ek 2.1 K-ortalamalar yöntemine göre 1. kümede bulunan X1, X2, X3, X4, X5, X6, X7, X8, X9 ve X10 standartlaştırılmış değişkenlere ilişkin kutu grafikleri



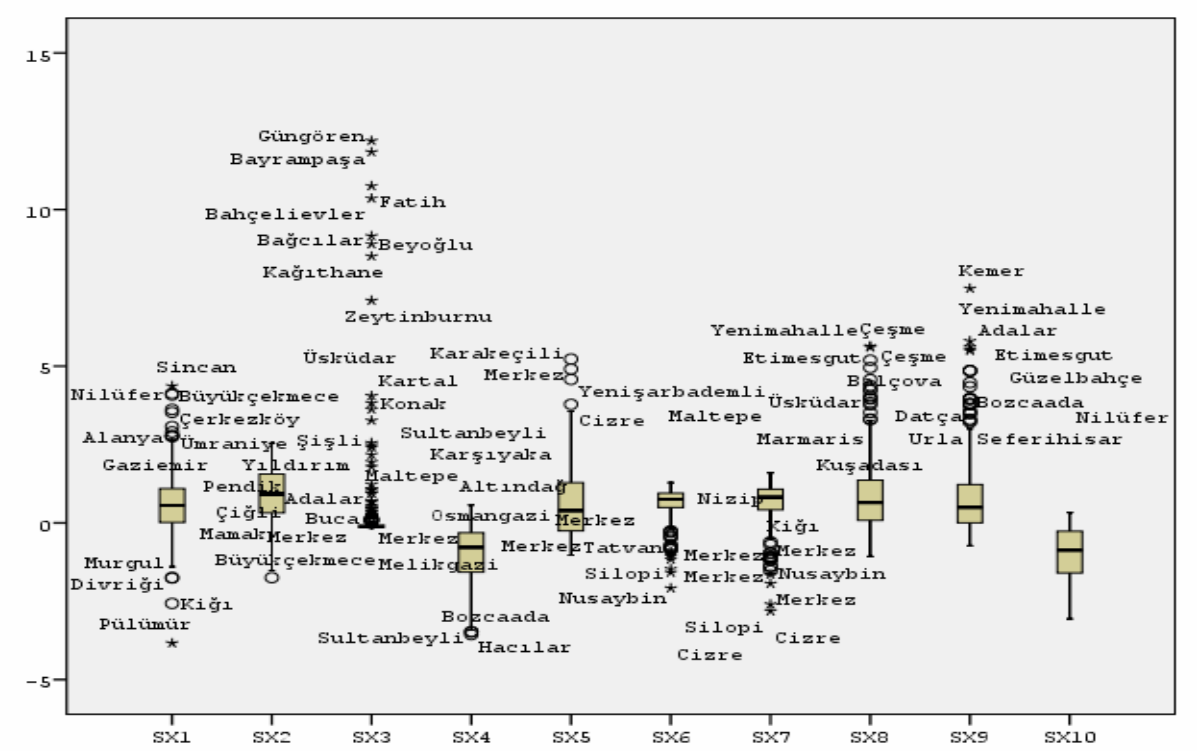
Şekil Ek 2.2 K-ortalamalar yöntemine göre 1. kümede bulunan X11, X12, X13, X14, X15, X16, X17, X18, X19 ve X20 standartlaştırılmış değişkenlere ilişkin kutu grafikleri



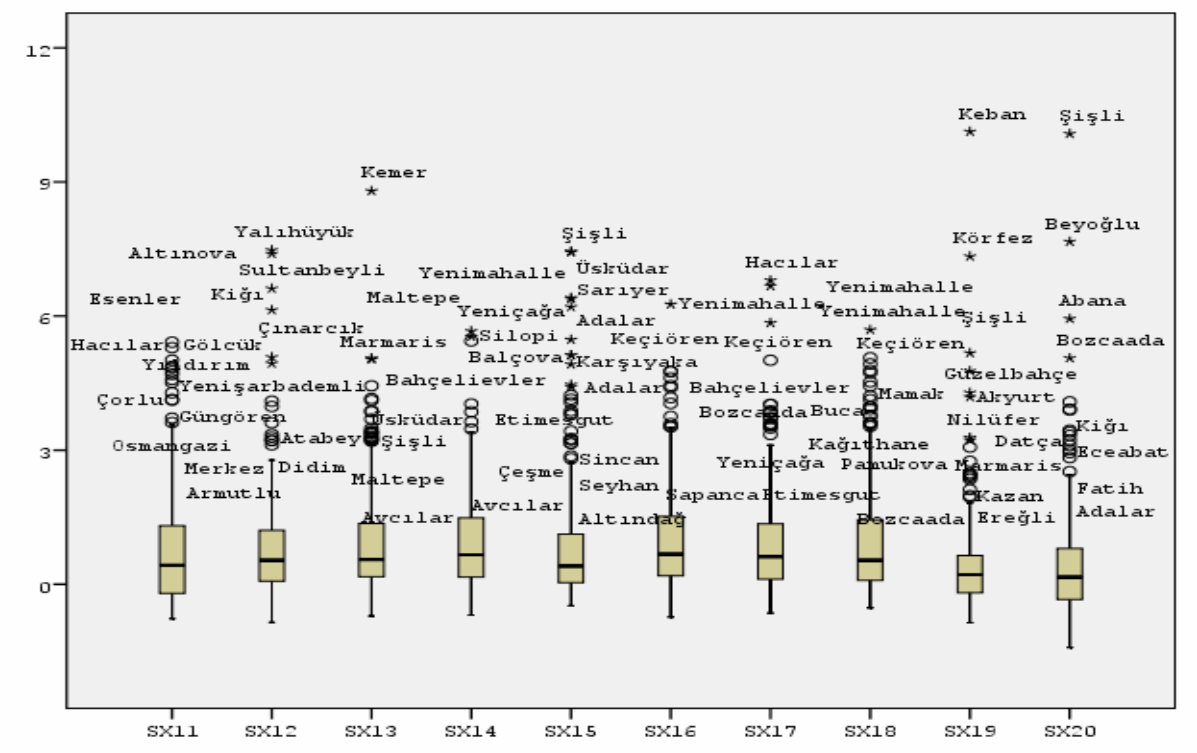
Şekil Ek 2.3 K-ortalamalar yöntemine göre 2. kümede bulunan X1, X2, X3, X4, X5, X6, X7, X8, X9 ve X10 standartlaştırılmış değişkenlere ilişkin kutu grafikleri



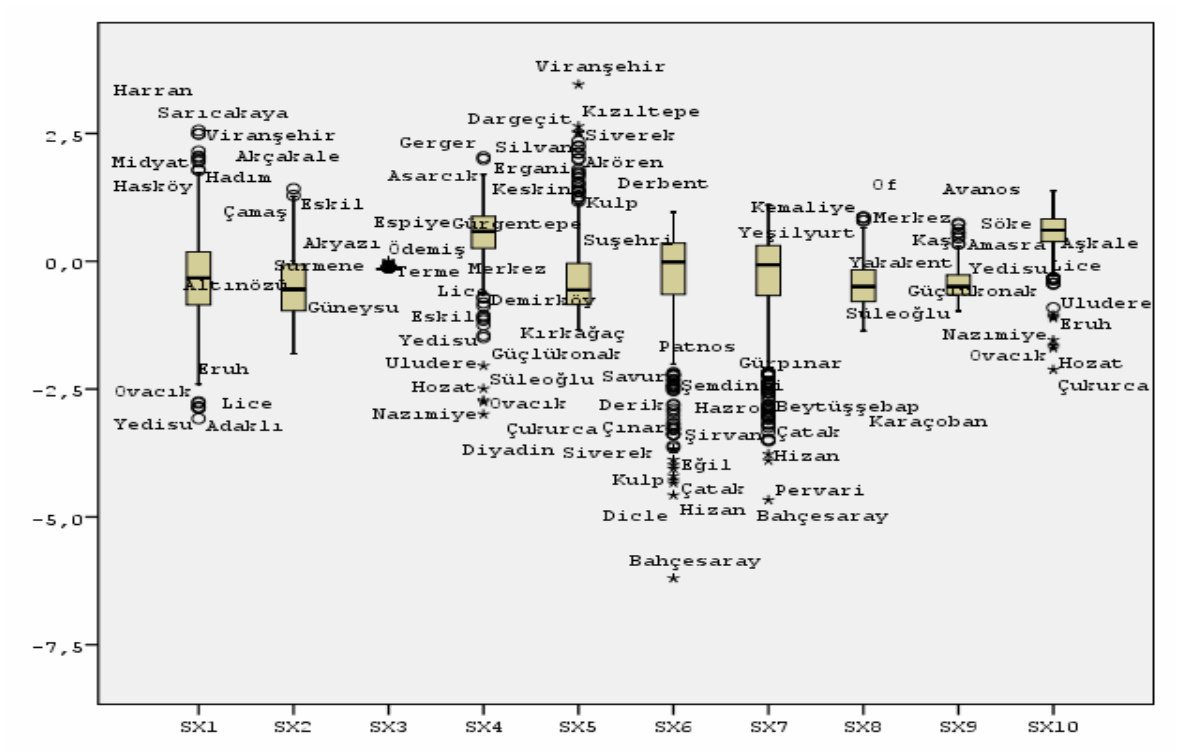
Şekil Ek 2.4 K-ortalamalar yöntemine göre 2. kümede bulunan X11, X12, X13, X14, X15, X16, X17, X18, X19 ve X20 standartlaştırılmış değişkenlere ilişkin kutu grafikler



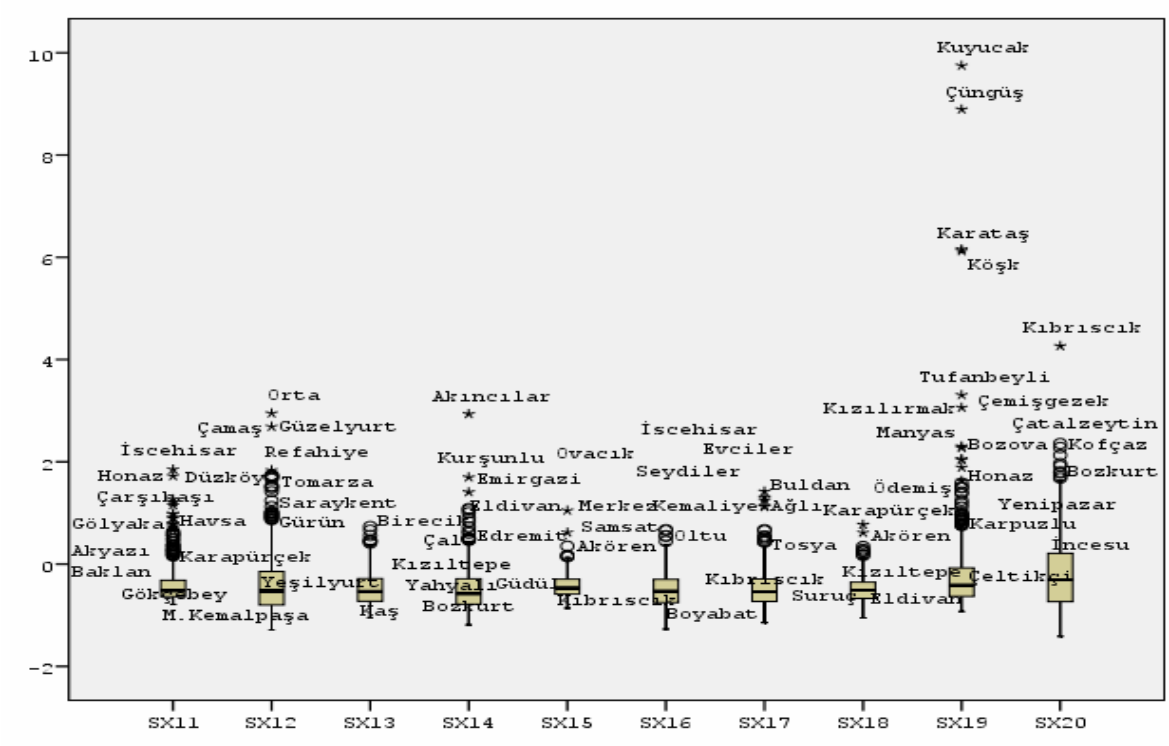
Şekil Ek 2.5 SOM kümeleme yöntemine göre 1. kümede bulunan X1, X2, X3, X4, X5, X6, X7, X8, X9 ve X10 standartlaştırılmış değişkenlere ilişkin kutu grafikleri



Şekil Ek 2.6 SOM kümeleme yöntemine göre 1. kümede bulunan X11, X12, X13, X14, X15, X16, X17, X18, X19 ve X20 standartlaştırılmış değişkenlere ilişkin kutu grafikleri



Şekil Ek 2.7 SOM kümeleme yöntemine göre 2. kümede bulunan X1, X2, X3, X4, X5, X6, X7, X8, X9 ve X10 standartlaştırılmış değişkenlere ilişkin kutu grafikleri



Şekil Ek 2.8 SOM kümeleme yöntemine göre 2. kümede bulunan X11, X12, X13, X14, X15, X16, X17, X18, X19 ve X20 standartlaştırılmış değişkenlere ilişkin kutu grafikleri

EK 3 Tek Bağlantılı Hiyerarşik, Tam Bağlantılı Hiyerarşik, K-Ortalamalar ve SOM Kümeleme Yöntemleri İçin Küme Doğruluk Endeksleri

Tablo Ek 3.1 Süsen veri seti için küme doğruluk endeksleri

Tek Bağlantılı	**Endeksler**					
Küme Sayıları	**S**	**DB**	**D**	**CH**	**KL**	**H**
2	**0.8466**	0.3836	**3.8212**	**501.9249**	**16.6741**	**501.9249**
3	0.6184	0.4305	2.502	277.4927	0.812037	12.85514
4	0.2705	0.2929	2.4998	187.1371	2.85719	2.136232
5	0.3136	0.2462	2.3779	168.8299	0.682599	24.30273
6	0.227	0.2004	0	138.7974	0.929011	4.122886
7	0.1281	0.1846	0	116.7446	1.04443	1.941762
8	0.0177	0.1611	0	100.5034	1.02079	1.348597
9	-0.0371	0.1436	0	87.92749	2.79243	0.814639
10	-0.0216	**0.1272**	0	80.85616	2.79243	4.888179
Tam Bağlantılı	**Endeksler**					
Küme Sayıları	**S**	**DB**	**D**	**CH**	**KL**	**H**
2	0.6507	0.6566	**2.4529**	281.0463	1.970996	**281.0463**
3	0.6644	0.5646	2.0968	484.899	5.346094	238.2407
4	**0.6718**	**0.5031**	2.2181	**493.9501**	**199.7075**	68.26793
5	0.5307	0.6402	1.0493	415.0192	0.00710211	16.89526
6	0.5254	0.7353	1.2145	456.5423	7.12163	50.93525
7	0.5104	0.6646	1.2145	424.8448	0.5327707	16.74622
8	0.5036	0.7716	1.1634	416.0136	2.435416	20.23045
9	0.5022	0.6777	1.1634	373.4154	6.196342	4.45123
10	0.479	0.7214	1.1959	349.8569	6.196342	8.229077
K-Ortalamalar	**Endeksler**					
Küme Sayıları	**S**	**DB**	**D**	**CH**	**KL**	**H**
2	**0.8504**	**0.4048**	**3.9131**	513.3038	5.9089	**513.3038**
3	0.7357	0.5879	2.4347	**560.3999**	3.5645	136.734
4	0.6057	0.7007	1.7226	529.3983	2.0885	55.07799
5	0.5594	0.6721	1.7483	494.0944	1.2166	33.59649
6	0.6184	0.7805	1.3139	474.8542	1.7729	28.12839
7	0.639	0.7438	1.2155	450.7701	0.82532	19.83291
8	0.4311	0.7382	1.2155	441.9314	**9.4318**	20.47926
9	0.5022	0.8276	1.065	411.527	0.34212	9.67647
10	0.5198	0.8209	0.96845	392.6583	0.34212	10.88577
SOM	**Endeksler**					
Küme Sayıları	**S**	**DB**	**D**	**CH**	**KL**	**H**
2	**0.8504**	NaN	NaN	0	0	0
3	0.7357	**0.5257**	**2.4347**	**560.3999**	**38.1538**	**1120.7998**
4	0.6723	1.1338	0.7588	390.2158	0.307458	6.6638177
5	0.5553	0.7611	1.8435	491.2997	0.67581	88.995108
6	0.6441	0.7113	0.4994	248.4396	0.929154	-48.748925
7	0.5049	0.7522	1.217	408.175	2.00504	126.26543
8	0.4881	1.1438	0.74368	266.4931	0.721421	-31.251533
9	0.5166	0.8772	0.85961	405.1457	0.897884	98.242394
10	0.629	1.1583	0.37964	220.2558	0.897884	-51.522795

Tablo Ek 3.2 911 ilçe küme doğruluk endeksleri

Tek Bağlantılı	**Endeksler**					
Küme Sayıları	**S**	**DB**	**D**	**CH**	**KL**	**H**
2	**0.8615**	183.7188	**0.005443**	17.1849	1.6742	17.1849
3	0.8412	49.54154	0	15.9423	1.1919	14.4455
4	0.6465	27.10232	0	12.4528	1.2113	5.32199
5	0.5314	19.08597	0	10.2472	0.78422	3.52637
6	0.5608	4.845101	0	**25.7399**	**2.3517**	**83.9575**
7	0.5581	3.81489	0	22.8565	1.177	7.51329
8	0.545	3.136366	0	20.7533	1.1405	7.19428
9	0.5342	2.649757	0	19.109	1.5801	6.68451
10	0.538	**2.270338**	0	18.2584	1.5801	9.93873
Tam Bağlantılı	**Endeksler**					
Küme Sayıları	**S**	**DB**	**D**	**CH**	**KL**	**H**
2	**0.8005**	**0.89345**	**1.6739**	226.0025	0.520763	226.0025
3	0.4541	1.1246	0.644	**284.0541**	**147.2199**	**274.1845**
4	0.4568	0.97638	0.81134	218.9792	0.057388	55.02656
5	0.4537	0.91734	0.92484	172.3961	0.219177	19.35349
6	0.3549	1.0928	0.66567	188.7747	28.2433	144.8219
7	0.3655	0.96923	0.66711	168.1133	0.580626	32.23221
8	0.3553	1.0732	0.66711	149.3057	1.363201	17.75986
9	0.3449	1.0762	0.65738	134.9868	0.064827	16.64594
10	0.313	1.0065	0.62081	144.3626	0.064827	100.384
K-Ortalamalar	**Endeksler**					
Küme Sayıları	**S**	**DB**	**D**	**CH**	**KL**	**H**
2	**0.6791**	**1.022**	**1.5279**	**597.6061**	**4.5123**	**597.6061**
3	0.46	1.2626	0.75885	450.8473	1.603	183.8663
4	0.4159	1.0779	0.62639	382.4091	1.4993	123.6923
5	0.3464	1.4021	0.48274	337.2922	0.93125	89.72154
6	0.3571	1.1535	0.59527	313.3741	2.1207	88.05843
7	0.2824	1.1912	0.45356	285.6775	0.91314	54.52465
8	0.2473	1.1253	0.47647	266.9008	0.86176	53.91301
9	0.2562	1.1506	0.45834	255.0921	1.0189	56.85887
10	0.2481	1.1412	0.40187	246.4757	1.0189	55.11402
SOM	**Endeksler**					
Küme Sayıları	**S**	**DB**	**D**	**CH**	**KL**	**H**
2	**0.6814**	1.0145	**1.5387**	**597.356**	**6.202**	**597.356**
3	0.4938	**0.90844**	0.92389	424.608	2.9771	152.3797
4	0.4159	1.4257	0.33979	333.8345	0.23272	79.17436
5	0.4658	1.037	0.66242	335.8903	1.0802	163.0846
6	0.4323	1.7769	0.23525	235.5536	0.57744	-66.1751
7	0.363	1.2057	0.50653	287.6282	1.1282	238.6818
8	0.367	2.0833	0.22547	192.1813	0.7949	-130.143
9	0.3243	1.3029	0.39885	250.9189	1.6415	266.5184
10	0.3165	1.6575	0.23039	183.7835	1.6415	-108.845

EK 4 K-Ortalamalar ve SOM Kümeleme Yöntemleriyle 3 Kümeye Ayrılan 911 İlçe Veri Seti Sonuçları

Tablo Ek 4.1 3 kümeye ayrılan 911 ilçe veri setinin küme sonuçları

Bölgeler	Şehirler	İlçeler	K-Ortalamalar	SOM	Bölgeler	Şehirler	İlçeler	K-Ortalamalar	SOM
Akdeniz	Adana	Seyhan	3	3	İç Anadolu	ÇANKIRI	Merkez	1	1
Akdeniz	Adana	Yüreğir	1	1	İç Anadolu	ÇANKIRI	Atkaracalar	1	1
Akdeniz	Adana	Aladağ	2	2	İç Anadolu	ÇANKIRI	Bayramören	2	2
Akdeniz	Adana	Ceyhan	1	1	İç Anadolu	ÇANKIRI	Çerkeş	1	1
Akdeniz	Adana	Feke	2	2	İç Anadolu	ÇANKIRI	Eldivan	2	2
Akdeniz	Adana	İmamoğlu	1	1	İç Anadolu	ÇANKIRI	Ilgaz	2	2
Akdeniz	Adana	Karaisali	2	2	İç Anadolu	ÇANKIRI	Kızılırmak	2	2
Akdeniz	Adana	Karataş	2	2	İç Anadolu	ÇANKIRI	Korgun	1	1
Akdeniz	Adana	Kozan	2	1	İç Anadolu	ÇANKIRI	Kurşunlu	2	2
Akdeniz	Adana	Pozantı	1	1	İç Anadolu	ÇANKIRI	Orta	2	2
Akdeniz	Adana	Saimbeyli	2	2	İç Anadolu	ÇANKIRI	Şabanözü	2	1
Akdeniz	Adana	Tufanbeyli	2	2	İç Anadolu	ÇANKIRI	Yapraklı	2	2
Akdeniz	Adana	Yumurtalık	2	2	İç Anadolu	ESKİŞEHİR	Merkez	3	3
Akdeniz	Antalya	Merkez	3	3	İç Anadolu	ESKİŞEHİR	Alpu	2	2
Akdeniz	Antalya	Akseki	2	2	İç Anadolu	ESKİŞEHİR	Beylikova	2	2
Akdeniz	Antalya	Alanya	1	1	İç Anadolu	ESKİŞEHİR	Çifteler	1	1
Akdeniz	Antalya	Elmalı	2	2	İç Anadolu	ESKİŞEHİR	Günyüzü	2	2
Akdeniz	Antalya	Finike	2	2	İç Anadolu	ESKİŞEHİR	Han	2	2
Akdeniz	Antalya	Gazipaşa	2	2	İç Anadolu	ESKİŞEHİR	İnönü	1	1
Akdeniz	Antalya	Gündoğmuş	2	2	İç Anadolu	ESKİŞEHİR	Mahmudiye	1	1
Akdeniz	Antalya	İbradi	1	1	İç Anadolu	ESKİŞEHİR	Mihalgazi	1	1
Akdeniz	Antalya	Kale	2	2	İç Anadolu	ESKİŞEHİR	Mihalıççık	2	2
Akdeniz	Antalya	Kaş	2	2	İç Anadolu	ESKİŞEHİR	Sarıcakaya	2	2
Akdeniz	Antalya	Kemer	3	3	İç Anadolu	ESKİŞEHİR	Seyitgazi	2	2
Akdeniz	Antalya	Korkuteli	2	2	İç Anadolu	ESKİŞEHİR	Sivrihisar	2	2
Akdeniz	Antalya	Kumluca	2	2	İç Anadolu	KAYSERİ	Kocasinan	1	3
Akdeniz	Antalya	Manavgat	1	1	İç Anadolu	KAYSERİ	Melikgazi	3	3
Akdeniz	Antalya	Serik	1	1	İç Anadolu	KAYSERİ	Akkışla	2	2
Akdeniz	Burdur	Merkez	1	1	İç Anadolu	KAYSERİ	Bünyan	2	2
Akdeniz	Burdur	Ağlasun	2	2	İç Anadolu	KAYSERİ	Develi	2	2
Akdeniz	Burdur	Altınyayla	1	1	İç Anadolu	KAYSERİ	Felahiye	2	2
Akdeniz	Burdur	Bucak	1	1	İç Anadolu	KAYSERİ	Hacılar	3	3
Akdeniz	Burdur	Çavdır	2	2	İç Anadolu	KAYSERİ	İncesu	2	2
Akdeniz	Burdur	Çeltikçi	2	2	İç Anadolu	KAYSERİ	Özvatan	1	1
Akdeniz	Burdur	Gölhisar	1	1	İç Anadolu	KAYSERİ	Pınarbaşı	2	2
Akdeniz	Burdur	Karamanlı	1	1	İç Anadolu	KAYSERİ	Sarıoğlan	2	2
Akdeniz	Burdur	Kemer	2	2	İç Anadolu	KAYSERİ	Sarız	2	2
Akdeniz	Burdur	Tefenni	2	2	İç Anadolu	KAYSERİ	Talas	1	1
Akdeniz	Burdur	Yeşilova	2	2	İç Anadolu	KAYSERİ	Tomarza	2	2
Akdeniz	HATAY	Merkez	1	1	İç Anadolu	KAYSERİ	Yahyalı	2	2
Akdeniz	HATAY	Altınözü	2	2	İç Anadolu	KAYSERİ	Yeşilhisar	2	2
Akdeniz	HATAY	Belen	1	1	İç Anadolu	KIRŞEHİR	Merkez	1	1
Akdeniz	HATAY	Dörtyol	1	1	İç Anadolu	KIRŞEHİR	Akçakent	2	2
Akdeniz	HATAY	Erzin	1	1	İç Anadolu	KIRŞEHİR	Akpınar	2	2
Akdeniz	HATAY	Hassa	2	2	İç Anadolu	KIRŞEHİR	Boztepe	2	2
Akdeniz	HATAY	İskenderun	1	1	İç Anadolu	KIRŞEHİR	Çiçekdağı	2	2

Akdeniz	HATAY	Kırıkhan	2	2	İç Anadolu	KIRŞEHİR	Kaman	2	2
Akdeniz	HATAY	Kumlu	2	2	İç Anadolu	KIRŞEHİR	Mucur	1	1
Akdeniz	HATAY	Reyhanlı	1	1	İç Anadolu	KONYA	Karatay	1	1
Akdeniz	HATAY	Samandağ	2	2	İç Anadolu	KONYA	Meram	1	3
Akdeniz	HATAY	Yayladağı	2	2	İç Anadolu	KONYA	Selçuklu	3	3
Akdeniz	ISPARTA	Merkez	1	3	İç Anadolu	KONYA	Ahırlı	2	2
Akdeniz	ISPARTA	Aksu	2	2	İç Anadolu	KONYA	Akören	2	2
Akdeniz	ISPARTA	Atabey	1	1	İç Anadolu	KONYA	Akşehir	1	1
Akdeniz	ISPARTA	Eğirdir	1	1	İç Anadolu	KONYA	Altınekin	2	2
Akdeniz	ISPARTA	Gelendost	2	2	İç Anadolu	KONYA	Beyşehir	2	2
Akdeniz	ISPARTA	Gönen	1	1	İç Anadolu	KONYA	Bozkır	2	2
Akdeniz	ISPARTA	Keçiborlu	1	1	İç Anadolu	KONYA	Cihanbeyli	2	2
Akdeniz	ISPARTA	Senirkent	1	1	İç Anadolu	KONYA	Çeltik	2	2
Akdeniz	ISPARTA	Sütçüler	2	2	İç Anadolu	KONYA	Çumra	2	2
Akdeniz	ISPARTA	Şarkikaraağaç	2	2	İç Anadolu	KONYA	Derbent	2	2
Akdeniz	ISPARTA	Uluborlu	3	3	İç Anadolu	KONYA	Derebucak	2	2
Akdeniz	ISPARTA	Yalvaç	2	2	İç Anadolu	KONYA	Doğanhisar	2	2
Akdeniz	ISPARTA	Yenişarbademli	1	1	İç Anadolu	KONYA	Emirgazi	2	2
Akdeniz	İÇEL	Merkez	1	1	İç Anadolu	KONYA	Ereğli	1	1
Akdeniz	İÇEL	Anamur	1	1	İç Anadolu	KONYA	Güneysınır	2	2
Akdeniz	İÇEL	Aydıncık	2	2	İç Anadolu	KONYA	Hadım	2	2
Akdeniz	İÇEL	Bozyazı	1	1	İç Anadolu	KONYA	Halkapınar	2	2
Akdeniz	İÇEL	Çamlıyayla	2	2	İç Anadolu	KONYA	Hüyük	2	2
Akdeniz	İÇEL	Erdemli	2	2	İç Anadolu	KONYA	Ilgın	2	2
Akdeniz	İÇEL	Gülnar	2	2	İç Anadolu	KONYA	Kadınhanı	2	2
Akdeniz	İÇEL	Mut	2	2	İç Anadolu	KONYA	Karapınar	2	1
Akdeniz	İÇEL	Silifke	1	1	İç Anadolu	KONYA	Kulu	2	2
Akdeniz	İÇEL	Tarsus	1	1	İç Anadolu	KONYA	Sarayönü	2	2
Akdeniz	K.MARAŞ	Merkez	1	1	İç Anadolu	KONYA	Seydişehir	1	1
Akdeniz	K.MARAŞ	Afşin	2	2	İç Anadolu	KONYA	Taşkent	2	2
Akdeniz	K.MARAŞ	Andırın	2	2	İç Anadolu	KONYA	Tuzlukçu	2	2
Akdeniz	K.MARAŞ	Çağlayancerit	2	2	İç Anadolu	KONYA	Yalıhüyük	1	1
Akdeniz	K.MARAŞ	Ekinözü	2	2	İç Anadolu	KONYA	Yunak	2	2
Akdeniz	K.MARAŞ	Elbistan	1	1	İç Anadolu	NEVŞEHİR	Merkez	1	1
Akdeniz	K.MARAŞ	Göksun	2	2	İç Anadolu	NEVŞEHİR	Acıgöl	2	2
Akdeniz	K.MARAŞ	Nurhak	2	2	İç Anadolu	NEVŞEHİR	Avanos	2	2
Akdeniz	K.MARAŞ	Pazarcık	2	2	İç Anadolu	NEVŞEHİR	Derinkuyu	2	2
Akdeniz	K.MARAŞ	Türkoğlu	2	2	İç Anadolu	NEVŞEHİR	Gülşehir	2	2
Akdeniz	MUĞLA	Merkez	1	1	İç Anadolu	NEVŞEHİR	Hacıbektaş	2	2
Akdeniz	MUĞLA	Bodrum	3	3	İç Anadolu	NEVŞEHİR	Kozaklı	2	2
Akdeniz	MUĞLA	Dalaman	1	1	İç Anadolu	NEVŞEHİR	Ürgüp	1	1
Akdeniz	MUĞLA	Datça	1	1	İç Anadolu	NİĞDE	Merkez	2	2
Akdeniz	MUĞLA	Fethiye	1	1	İç Anadolu	NİĞDE	Altunhisar	2	2
Akdeniz	MUĞLA	Kavaklıdere	2	2	İç Anadolu	NİĞDE	Bor	1	1
Akdeniz	MUĞLA	Köyceğiz	2	2	İç Anadolu	NİĞDE	Çamardı	2	2
Akdeniz	MUĞLA	Marmaris	3	3	İç Anadolu	NİĞDE	Çiftlik	2	2
Akdeniz	MUĞLA	Milas	1	1	İç Anadolu	NİĞDE	Ulukışla	2	2
Akdeniz	MUĞLA	Ortaca	1	1	İç Anadolu	YOZGAT	Merkez	1	1
Akdeniz	MUĞLA	Ula	1	1	İç Anadolu	YOZGAT	Akdağmadeni	2	2
Akdeniz	MUĞLA	Yatağan	2	2	İç Anadolu	YOZGAT	Aydıncık	2	2
Akdeniz	KİLİS	Merkez	1	1	İç Anadolu	YOZGAT	Boğazlıyan	2	2
Akdeniz	KİLİS	Elbeyli	2	2	İç Anadolu	YOZGAT	Çandır	1	1

Akdeniz	KİLİS	Musabeyli	2	2	İç Anadolu	YOZGAT	Çayıralan	2	2
Akdeniz	KİLİS	Polateli	2	2	İç Anadolu	YOZGAT	Çekerek	2	2
Akdeniz	OSMANİYE	Merkez	1	1	İç Anadolu	YOZGAT	Kadışehri	2	2
Akdeniz	OSMANİYE	Bahçe	1	1	İç Anadolu	YOZGAT	Saraykent	2	2
Akdeniz	OSMANİYE	Düziçi	2	2	İç Anadolu	YOZGAT	Sarıkaya	2	2
Akdeniz	OSMANİYE	Kadirli	1	1	İç Anadolu	YOZGAT	Sorgun	2	2
Doğu Anadolu	Adıyaman	Merkez	1	1	İç Anadolu	YOZGAT	Şefaatli	2	2
Doğu Anadolu	Adıyaman	Besni	2	2	İç Anadolu	YOZGAT	Yenifakılı	2	2
Doğu Anadolu	Adıyaman	Çelikhan	2	2	İç Anadolu	YOZGAT	Yerköy	1	1
Doğu Anadolu	Adıyaman	Gerger	2	2	İç Anadolu	AKSARAY	Merkez	1	1
Doğu Anadolu	Adıyaman	Gölbaşı	2	2	İç Anadolu	AKSARAY	Ağaçören	2	2
Doğu Anadolu	Adıyaman	Kahta	2	2	İç Anadolu	AKSARAY	Eskil	2	2
Doğu Anadolu	Adıyaman	Samsat	2	2	İç Anadolu	AKSARAY	Gülağaç	2	2
Doğu Anadolu	Adıyaman	Sincik	2	2	İç Anadolu	AKSARAY	Güzelyurt	2	2
Doğu Anadolu	Adıyaman	Tut	2	2	İç Anadolu	AKSARAY	Ortaköy	2	2
Doğu Anadolu	Ağrı	Merkez	1	1	İç Anadolu	AKSARAY	Sarıyahşi	2	2
Doğu Anadolu	Ağrı	Diyadin	2	2	İç Anadolu	KARAMAN	Merkez	1	1
Doğu Anadolu	Ağrı	Doğubeyazıt	2	2	İç Anadolu	KARAMAN	Ayrancı	2	2
Doğu Anadolu	Ağrı	Eleşkirt	2	2	İç Anadolu	KARAMAN	Başyayla	2	2
Doğu Anadolu	Ağrı	Hamur	2	2	İç Anadolu	KARAMAN	Ermenek	2	2
Doğu Anadolu	Ağrı	Patnos	2	2	İç Anadolu	KARAMAN	Kazımkarabekir	1	1
Doğu Anadolu	Ağrı	Taşlıçay	2	2	İç Anadolu	KARAMAN	Sarıveliler	2	2
Doğu Anadolu	Ağrı	Tutak	2	2	İç Anadolu	KIRIKKALE	Merkez	3	3
Doğu Anadolu	Bingöl	Merkez	2	2	İç Anadolu	KIRIKKALE	Bahşılı	1	1
Doğu Anadolu	Bingöl	Adaklı	2	2	İç Anadolu	KIRIKKALE	Balışeyh	2	2
Doğu Anadolu	Bingöl	Genç	2	2	İç Anadolu	KIRIKKALE	Çelebi	2	2
Doğu Anadolu	Bingöl	Karlıova	2	2	İç Anadolu	KIRIKKALE	Delice	2	2
Doğu Anadolu	Bingöl	Kiğı	1	1	İç Anadolu	KIRIKKALE	Karakeçili	1	1
Doğu Anadolu	Bingöl	Solhan	2	2	İç Anadolu	KIRIKKALE	Keskin	2	2
Doğu Anadolu	Bingöl	Yayladere	1	1	İç Anadolu	KIRIKKALE	Sulakyurt	2	2
Doğu Anadolu	Bingöl	Yedisu	2	2	İç Anadolu	KIRIKKALE	Yahşihan	1	1
Doğu Anadolu	Bitlis	Merkez	1	1	Karadeniz	Amasya	Merkez	1	1
Doğu Anadolu	Bitlis	Adilcevaz	1	1	Karadeniz	Amasya	Göynücek	2	2
Doğu Anadolu	Bitlis	Ahlat	1	1	Karadeniz	Amasya	Gümüşhacıköy	2	2
Doğu Anadolu	Bitlis	Güroymak	2	2	Karadeniz	Amasya	Hamamözü	2	2
Doğu Anadolu	Bitlis	Hizan	2	2	Karadeniz	Amasya	Merzifon	1	1
Doğu Anadolu	Bitlis	Mutki	2	2	Karadeniz	Amasya	Suluova	1	1
Doğu Anadolu	Bitlis	Tatvan	1	1	Karadeniz	Amasya	Taşova	2	2
Doğu Anadolu	ELAZIĞ	Merkez	1	1	Karadeniz	Artvin	Merkez	1	1
Doğu Anadolu	ELAZIĞ	Ağın	1	1	Karadeniz	Artvin	Ardanuç	2	2
Doğu Anadolu	ELAZIĞ	Alacakaya	2	2	Karadeniz	Artvin	Arhavi	1	1
Doğu Anadolu	ELAZIĞ	Arıcak	2	2	Karadeniz	Artvin	Borçka	2	1
Doğu Anadolu	ELAZIĞ	Baskil	2	2	Karadeniz	Artvin	Hopa	1	1
Doğu Anadolu	ELAZIĞ	Karakoçan	2	2	Karadeniz	Artvin	Murgul	1	1
Doğu Anadolu	ELAZIĞ	Keban	1	1	Karadeniz	Artvin	Şavşat	2	2
Doğu Anadolu	ELAZIĞ	Kovancılar	2	2	Karadeniz	Artvin	Yusufeli	2	2
Doğu Anadolu	ELAZIĞ	Maden	2	2	Karadeniz	Bolu	Merkez	1	1
Doğu Anadolu	ELAZIĞ	Palu	2	2	Karadeniz	Bolu	Dörtdivan	2	2
Doğu Anadolu	ELAZIĞ	Sivrice	2	2	Karadeniz	Bolu	Gerede	1	1
Doğu Anadolu	ERZİNCAN	Merkez	1	1	Karadeniz	Bolu	Göynük	2	2
Doğu Anadolu	ERZİNCAN	Çayırlı	2	2	Karadeniz	Bolu	Kıbrıscık	2	2
Doğu Anadolu	ERZİNCAN	İliç	2	2	Karadeniz	Bolu	Mengen	2	2

Doğu Anadolu	ERZİNCAN	Kemah	2	2	Karadeniz	Bolu	Mudurnu	2	2
Doğu Anadolu	ERZİNCAN	Kemaliye	2	2	Karadeniz	Bolu	Seben	2	2
Doğu Anadolu	ERZİNCAN	Otlukbeli	1	1	Karadeniz	Bolu	Yeniçağa	1	1
Doğu Anadolu	ERZİNCAN	Refahiye	2	2	Karadeniz	ÇORUM	Merkez	1	1
Doğu Anadolu	ERZİNCAN	Tercan	2	2	Karadeniz	ÇORUM	Alaca	2	2
Doğu Anadolu	ERZİNCAN	Üzümlü	2	2	Karadeniz	ÇORUM	Bayat	2	2
Doğu Anadolu	ERZURUM	Merkez	3	3	Karadeniz	ÇORUM	Boğazkale	2	2
Doğu Anadolu	ERZURUM	Aşkale	2	2	Karadeniz	ÇORUM	Dodurga	2	2
Doğu Anadolu	ERZURUM	Çat	2	2	Karadeniz	ÇORUM	İskilip	2	2
Doğu Anadolu	ERZURUM	Hınıs	2	2	Karadeniz	ÇORUM	Kargı	2	2
Doğu Anadolu	ERZURUM	Horasan	2	2	Karadeniz	ÇORUM	Laçin	2	2
Doğu Anadolu	ERZURUM	Ilıca	2	2	Karadeniz	ÇORUM	Mecitözü	2	2
Doğu Anadolu	ERZURUM	İspir	2	2	Karadeniz	ÇORUM	Oğuzlar	2	2
Doğu Anadolu	ERZURUM	Karaçoban	2	2	Karadeniz	ÇORUM	Ortaköy	2	2
Doğu Anadolu	ERZURUM	Karayazı	2	2	Karadeniz	ÇORUM	Osmancık	2	2
Doğu Anadolu	ERZURUM	Köprüköy	2	2	Karadeniz	ÇORUM	Sungurlu	2	2
Doğu Anadolu	ERZURUM	Narman	2	2	Karadeniz	ÇORUM	Uğurludağ	2	2
Doğu Anadolu	ERZURUM	Oltu	2	2	Karadeniz	GİRESUN	Merkez	1	1
Doğu Anadolu	ERZURUM	Olur	2	2	Karadeniz	GİRESUN	Alucra	2	2
Doğu Anadolu	ERZURUM	Pasinler	2	2	Karadeniz	GİRESUN	Bulancak	1	1
Doğu Anadolu	ERZURUM	Pazaryolu	2	2	Karadeniz	GİRESUN	Çamoluk	2	2
Doğu Anadolu	ERZURUM	Şenkaya	2	2	Karadeniz	GİRESUN	Çanakçı	2	2
Doğu Anadolu	ERZURUM	Tekman	2	2	Karadeniz	GİRESUN	Dereli	2	2
Doğu Anadolu	ERZURUM	Tortum	2	2	Karadeniz	GİRESUN	Doğankent	2	2
Doğu Anadolu	ERZURUM	Uzundere	2	2	Karadeniz	GİRESUN	Espiye	2	2
Doğu Anadolu	HAKKARİ	Merkez	1	1	Karadeniz	GİRESUN	Eynesil	2	2
Doğu Anadolu	HAKKARİ	Çukurca	2	2	Karadeniz	GİRESUN	Görele	2	2
Doğu Anadolu	HAKKARİ	Şemdinli	2	2	Karadeniz	GİRESUN	Güce	2	2
Doğu Anadolu	HAKKARİ	Yüksekova	2	2	Karadeniz	GİRESUN	Keşap	2	2
Doğu Anadolu	KARS	Merkez	1	1	Karadeniz	GİRESUN	Piraziz	2	2
Doğu Anadolu	KARS	Akyaka	2	2	Karadeniz	GİRESUN	Ş.Karahisar	1	1
Doğu Anadolu	KARS	Arpaçay	2	2	Karadeniz	GİRESUN	Tirebolu	2	2
Doğu Anadolu	KARS	Digor	2	2	Karadeniz	GİRESUN	Yağlıdere	2	2
Doğu Anadolu	KARS	Kağızman	2	2	Karadeniz	GÜMÜŞHANE	Merkez	1	1
Doğu Anadolu	KARS	Sarıkamış	2	2	Karadeniz	GÜMÜŞHANE	Kelkit	2	2
Doğu Anadolu	KARS	Selim	2	2	Karadeniz	GÜMÜŞHANE	Köse	2	2
Doğu Anadolu	KARS	Susuz	2	2	Karadeniz	GÜMÜŞHANE	Kürtün	2	2
Doğu Anadolu	MALATYA	Merkez	1	1	Karadeniz	GÜMÜŞHANE	Şiran	2	2
Doğu Anadolu	MALATYA	Akçadağ	2	2	Karadeniz	GÜMÜŞHANE	Torul	2	2
Doğu Anadolu	MALATYA	Arapkir	2	2	Karadeniz	KASTAMONU	Merkez	1	1
Doğu Anadolu	MALATYA	Arguvan	2	2	Karadeniz	KASTAMONU	Abana	1	1
Doğu Anadolu	MALATYA	Battalgazi	2	2	Karadeniz	KASTAMONU	Ağlı	2	2
Doğu Anadolu	MALATYA	Darende	2	2	Karadeniz	KASTAMONU	Araç	2	2
Doğu Anadolu	MALATYA	Doğanşehir	2	2	Karadeniz	KASTAMONU	Azdavay	2	2
Doğu Anadolu	MALATYA	Doğanyol	2	2	Karadeniz	KASTAMONU	Bozkurt	2	2
Doğu Anadolu	MALATYA	Hekimhan	2	2	Karadeniz	KASTAMONU	Cide	2	2
Doğu Anadolu	MALATYA	Kale	2	2	Karadeniz	KASTAMONU	Çatalzeytin	2	2
Doğu Anadolu	MALATYA	Kuluncak	2	2	Karadeniz	KASTAMONU	Daday	2	2
Doğu Anadolu	MALATYA	Pötürge	2	2	Karadeniz	KASTAMONU	Devrekani	2	2
Doğu Anadolu	MALATYA	Yazıhan	2	2	Karadeniz	KASTAMONU	Doğanyurt	2	2
Doğu Anadolu	MALATYA	Yeşilyurt	2	2	Karadeniz	KASTAMONU	Hanönü	2	2
Doğu Anadolu	MUŞ	Merkez	2	2	Karadeniz	KASTAMONU	İhsangazi	2	2

Doğu Anadolu	MUŞ	Bulanık	2	2	Karadeniz	KASTAMONU	İnebolu	2	2
Doğu Anadolu	MUŞ	Hasköy	2	2	Karadeniz	KASTAMONU	Küre	2	2
Doğu Anadolu	MUŞ	Korkut	2	2	Karadeniz	KASTAMONU	Pınarbaşı	2	2
Doğu Anadolu	MUŞ	Malazgirt	2	2	Karadeniz	KASTAMONU	Seydiler	2	2
Doğu Anadolu	MUŞ	Varto	2	2	Karadeniz	KASTAMONU	Şenpazar	2	2
Doğu Anadolu	SİİRT	Merkez	1	1	Karadeniz	KASTAMONU	Taşköprü	2	2
Doğu Anadolu	SİİRT	Aydınlar	2	2	Karadeniz	KASTAMONU	Tosya	2	2
Doğu Anadolu	SİİRT	Baykan	2	2	Karadeniz	ORDU	Merkez	1	1
Doğu Anadolu	SİİRT	Eruh	2	2	Karadeniz	ORDU	Akkuş	2	2
Doğu Anadolu	SİİRT	Kurtalan	2	2	Karadeniz	ORDU	Aybastı	2	2
Doğu Anadolu	SİİRT	Pervari	2	2	Karadeniz	ORDU	Çamaş	2	2
Doğu Anadolu	SİİRT	Şirvan	2	2	Karadeniz	ORDU	Çatalpınar	2	2
Doğu Anadolu	SİVAS	Merkez	1	1	Karadeniz	ORDU	Çaybaşı	2	2
Doğu Anadolu	SİVAS	Akıncılar	2	2	Karadeniz	ORDU	Fatsa	1	1
Doğu Anadolu	SİVAS	Altınyayla	2	2	Karadeniz	ORDU	Gölköy	2	2
Doğu Anadolu	SİVAS	Divriği	1	1	Karadeniz	ORDU	Gülyalı	2	1
Doğu Anadolu	SİVAS	Doğanşar	2	2	Karadeniz	ORDU	Gürgentepe	2	2
Doğu Anadolu	SİVAS	Gemerek	2	2	Karadeniz	ORDU	İkizce	2	2
Doğu Anadolu	SİVAS	Gölova	2	2	Karadeniz	ORDU	Kabadüz	2	2
Doğu Anadolu	SİVAS	Gürün	2	2	Karadeniz	ORDU	Kabataş	2	2
Doğu Anadolu	SİVAS	Hafik	2	2	Karadeniz	ORDU	Korgan	2	2
Doğu Anadolu	SİVAS	İmranlı	2	2	Karadeniz	ORDU	Kumru	2	2
Doğu Anadolu	SİVAS	Kangal	2	2	Karadeniz	ORDU	Mesudiye	2	2
Doğu Anadolu	SİVAS	Koyulhisar	2	2	Karadeniz	ORDU	Perşembe	2	2
Doğu Anadolu	SİVAS	Suşehri	2	2	Karadeniz	ORDU	Ulubey	2	2
Doğu Anadolu	SİVAS	Şarkışla	2	2	Karadeniz	ORDU	Ünye	2	2
Doğu Anadolu	SİVAS	Ulaş	2	2	Karadeniz	RİZE	Merkez	1	1
Doğu Anadolu	SİVAS	Yıldızeli	2	2	Karadeniz	RİZE	Ardeşen	1	1
Doğu Anadolu	SİVAS	Zara	2	2	Karadeniz	RİZE	Çamlıhemşin	2	2
Doğu Anadolu	TUNCELİ	Merkez	1	1	Karadeniz	RİZE	Çayeli	2	2
Doğu Anadolu	TUNCELİ	Çemişgezek	2	2	Karadeniz	RİZE	Derepazarı	1	1
Doğu Anadolu	TUNCELİ	Hozat	2	2	Karadeniz	RİZE	Fındıklı	1	1
Doğu Anadolu	TUNCELİ	Mazgirt	2	2	Karadeniz	RİZE	Güneysu	2	2
Doğu Anadolu	TUNCELİ	Nazımiye	2	2	Karadeniz	RİZE	Hemşin	1	1
Doğu Anadolu	TUNCELİ	Ovacık	2	2	Karadeniz	RİZE	İkizdere	2	2
Doğu Anadolu	TUNCELİ	Pertek	2	2	Karadeniz	RİZE	İyidere	2	2
Doğu Anadolu	TUNCELİ	Pülümür	2	1	Karadeniz	RİZE	Kalkandere	2	2
Doğu Anadolu	VAN	Merkez	1	1	Karadeniz	RİZE	Pazar	2	2
Doğu Anadolu	VAN	Bahçesaray	2	2	Karadeniz	SAMSUN	Merkez	3	3
Doğu Anadolu	VAN	Başkale	2	2	Karadeniz	SAMSUN	Alaçam	2	2
Doğu Anadolu	VAN	Çaldıran	2	2	Karadeniz	SAMSUN	Asarcık	2	2
Doğu Anadolu	VAN	Çatak	2	2	Karadeniz	SAMSUN	Ayvacık	2	2
Doğu Anadolu	VAN	Edremit	2	2	Karadeniz	SAMSUN	Bafra	2	2
Doğu Anadolu	VAN	Erciş	2	2	Karadeniz	SAMSUN	Çarşamba	2	2
Doğu Anadolu	VAN	Gevaş	2	2	Karadeniz	SAMSUN	Havza	2	2
Doğu Anadolu	VAN	Gürpınar	2	2	Karadeniz	SAMSUN	Kavak	2	2
Doğu Anadolu	VAN	Muradiye	2	2	Karadeniz	SAMSUN	Ladik	2	2
Doğu Anadolu	VAN	Özalp	2	2	Karadeniz	SAMSUN	19.May	2	2
Doğu Anadolu	VAN	Saray	2	2	Karadeniz	SAMSUN	Salıpazarı	2	2
Doğu Anadolu	ŞIRNAK	Merkez	1	1	Karadeniz	SAMSUN	Tekkeköy	2	2
Doğu Anadolu	ŞIRNAK	Beytüşşebap	2	2	Karadeniz	SAMSUN	Terme	2	2
Doğu Anadolu	ŞIRNAK	Cizre	1	1	Karadeniz	SAMSUN	Vezirköprü	2	2

Doğu Anadolu	ŞIRNAK	Güçlükonak	2	2	Karadeniz	SAMSUN	Yakakent	2	2
Doğu Anadolu	ŞIRNAK	İdil	2	2	Karadeniz	SİNOP	Merkez	1	1
Doğu Anadolu	ŞIRNAK	Silopi	1	1	Karadeniz	SİNOP	Ayancık	2	2
Doğu Anadolu	ŞIRNAK	Uludere	2	2	Karadeniz	SİNOP	Boyabat	2	2
Doğu Anadolu	ARDAHAN	Merkez	2	2	Karadeniz	SİNOP	Dikmen	2	2
Doğu Anadolu	ARDAHAN	Çıldır	2	2	Karadeniz	SİNOP	Durağan	2	2
Doğu Anadolu	ARDAHAN	Damal	2	2	Karadeniz	SİNOP	Erfelek	2	2
Doğu Anadolu	ARDAHAN	Göle	2	2	Karadeniz	SİNOP	Gerze	2	2
Doğu Anadolu	ARDAHAN	Hanak	2	2	Karadeniz	SİNOP	Saraydüzü	2	2
Doğu Anadolu	ARDAHAN	Posof	2	2	Karadeniz	SİNOP	Türkeli	2	2
Doğu Anadolu	IĞDIR	Merkez	1	1	Karadeniz	TOKAT	Merkez	1	1
Doğu Anadolu	IĞDIR	Aralık	2	2	Karadeniz	TOKAT	Almus	2	2
Doğu Anadolu	IĞDIR	Karakoyunlu	2	2	Karadeniz	TOKAT	Artova	2	2
Doğu Anadolu	İĞDIR	Tuzluca	2	2	Karadeniz	TOKAT	Başçiftlik	2	2
Ege	Afyon	Merkez	1	1	Karadeniz	TOKAT	Erbaa	2	2
Ege	Afyon	Başmakçı	2	2	Karadeniz	TOKAT	Niksar	2	2
Ege	Afyon	Bayat	2	2	Karadeniz	TOKAT	Pazar	2	2
Ege	Afyon	Bolvadin	1	1	Karadeniz	TOKAT	Reşadiye	2	2
Ege	Afyon	Çay	2	2	Karadeniz	TOKAT	Sulusaray	2	2
Ege	Afyon	Çobanlar	2	2	Karadeniz	TOKAT	Turhal	1	1
Ege	Afyon	Dazkırı	1	1	Karadeniz	TOKAT	Yeşilyurt	2	2
Ege	Afyon	Dinar	2	2	Karadeniz	TOKAT	Zile	2	2
Ege	Afyon	Emirdağ	2	2	Karadeniz	TRABZON	Merkez	3	3
Ege	Afyon	Evciler	2	2	Karadeniz	TRABZON	Akçaabat	2	2
Ege	Afyon	Hocalar	2	2	Karadeniz	TRABZON	Araklı	2	2
Ege	Afyon	İhsaniye	2	2	Karadeniz	TRABZON	Arsin	2	2
Ege	Afyon	İscehisar	2	2	Karadeniz	TRABZON	Beşikdüzü	1	1
Ege	Afyon	Kızılören	2	2	Karadeniz	TRABZON	Çarşıbaşı	2	2
Ege	Afyon	Sandıklı	2	2	Karadeniz	TRABZON	Çaykara	2	2
Ege	Afyon	Sincanlı	2	2	Karadeniz	TRABZON	Dernekpazarı	1	1
Ege	Afyon	Sultandağı	2	2	Karadeniz	TRABZON	Düzköy	2	2
Ege	Afyon	Şuhut	2	2	Karadeniz	TRABZON	Hayrat	2	2
Ege	Aydın	Merkez	1	1	Karadeniz	TRABZON	Köprübaşı	2	2
Ege	Aydın	Bozdoğan	2	2	Karadeniz	TRABZON	Maçka	2	2
Ege	Aydın	Buharkent	2	2	Karadeniz	TRABZON	Of	2	2
Ege	Aydın	Çine	2	2	Karadeniz	TRABZON	Sürmene	2	2
Ege	Aydın	Didim	1	1	Karadeniz	TRABZON	Şalpazarı	2	2
Ege	Aydın	Germencik	2	2	Karadeniz	TRABZON	Tonya	2	2
Ege	Aydın	İncirliova	2	2	Karadeniz	TRABZON	Vakfıkebir	1	1
Ege	Aydın	Karacasu	2	2	Karadeniz	TRABZON	Yomra	2	2
Ege	Aydın	Karpuzlu	2	2	Karadeniz	ZONGULDAK	Merkez	1	1
Ege	Aydın	Koçarlı	2	2	Karadeniz	ZONGULDAK	Alaplı	2	2
Ege	Aydın	Köşk	2	2	Karadeniz	ZONGULDAK	Çaycuma	2	2
Ege	Aydın	Kuşadası	3	3	Karadeniz	ZONGULDAK	Devrek	2	2
Ege	Aydın	Kuyucak	2	2	Karadeniz	ZONGULDAK	Ereğli	1	1
Ege	Aydın	Nazilli	1	1	Karadeniz	ZONGULDAK	Gökçebey	2	2
Ege	Aydın	Söke	2	2	Karadeniz	BAYBURT	Merkez	2	2
Ege	Aydın	Sultanhisar	2	2	Karadeniz	BAYBURT	Aydıntepe	2	2
Ege	Aydın	Yenipazar	2	2	Karadeniz	BAYBURT	Demirözü	2	2
Ege	DENİZLİ	Merkez	1	1	Karadeniz	BARTIN	Merkez	2	2
Ege	DENİZLİ	Acıpayam	2	2	Karadeniz	BARTIN	Amasra	2	2
Ege	DENİZLİ	Akköy	2	2	Karadeniz	BARTIN	Kurucaşile	2	2

Ege	DENİZLİ	Babadağ	1	1	Karadeniz	BARTIN	Ulus	2	2
Ege	DENİZLİ	Baklan	2	2	Karadeniz	KARABÜK	Merkez	1	3
Ege	DENİZLİ	Bekilli	2	2	Karadeniz	KARABÜK	Eflani	2	2
Ege	DENİZLİ	Beyağaç	2	2	Karadeniz	KARABÜK	Eskipazar	2	2
Ege	DENİZLİ	Bozkurt	2	2	Karadeniz	KARABÜK	Ovacık	2	2
Ege	DENİZLİ	Buldan	2	2	Karadeniz	KARABÜK	Safranbolu	1	1
Ege	DENİZLİ	Çal	2	2	Karadeniz	KARABÜK	Yenice	2	2
Ege	DENİZLİ	Çameli	2	2	Karadeniz	DÜZCE	Merkez	1	1
Ege	DENİZLİ	Çardak	1	1	Karadeniz	DÜZCE	Akçakoca	1	1
Ege	DENİZLİ	Çivril	2	2	Karadeniz	DÜZCE	Cumayeri	1	1
Ege	DENİZLİ	Güney	2	2	Karadeniz	DÜZCE	Çilimli	2	2
Ege	DENİZLİ	Honaz	2	2	Karadeniz	DÜZCE	Gölyaka	2	2
Ege	DENİZLİ	Kale	2	2	Karadeniz	DÜZCE	Gümüşova	1	1
Ege	DENİZLİ	Sarayköy	2	2	Karadeniz	DÜZCE	Yığılca	2	2
Ege	DENİZLİ	Serinhisar	1	1	Marmara	Balıkesir	Merkez	1	1
Ege	DENİZLİ	Tavas	2	2	Marmara	Balıkesir	Ayvalık	1	1
Ege	İZMİR	Balçova	3	3	Marmara	Balıkesir	Balya	2	2
Ege	İZMİR	Bornova	3	3	Marmara	Balıkesir	Bandırma	1	3
Ege	İZMİR	Buca	3	3	Marmara	Balıkesir	Bigadiç	2	2
Ege	İZMİR	Çiğli	3	3	Marmara	Balıkesir	Burhaniye	1	1
Ege	İZMİR	Gaziemir	3	3	Marmara	Balıkesir	Dursunbey	2	2
Ege	İZMİR	Güzelbahçe	3	3	Marmara	Balıkesir	Edremit	1	1
Ege	İZMİR	Karşıyaka	3	3	Marmara	Balıkesir	Erdek	1	1
Ege	İZMİR	Konak	3	3	Marmara	Balıkesir	Gömeç	1	1
Ege	İZMİR	Narlıdere	3	3	Marmara	Balıkesir	Gönen	1	1
Ege	İZMİR	Bayındır	2	2	Marmara	Balıkesir	Havran	2	2
Ege	İZMİR	Bergama	2	2	Marmara	Balıkesir	İvrindi	2	2
Ege	İZMİR	Beydağ	2	2	Marmara	Balıkesir	Kepsut	2	2
Ege	İZMİR	Çeşme	3	3	Marmara	Balıkesir	Manyas	2	2
Ege	İZMİR	Dikili	1	1	Marmara	Balıkesir	Marmara	1	1
Ege	İZMİR	Foça	1	1	Marmara	Balıkesir	Savaştepe	2	2
Ege	İZMİR	Karaburun	1	1	Marmara	Balıkesir	Sındırgı	2	2
Ege	İZMİR	Kemalpaşa	1	1	Marmara	Balıkesir	Susurluk	1	1
Ege	İZMİR	Kınık	2	2	Marmara	Bilecik	Merkez	1	1
Ege	İZMİR	Kiraz	2	2	Marmara	Bilecik	Bozüyük	1	1
Ege	İZMİR	Menderes	1	1	Marmara	Bilecik	Gölpazarı	2	2
Ege	İZMİR	Menemen	1	1	Marmara	Bilecik	İnhisar	2	2
Ege	İZMİR	Ödemiş	2	2	Marmara	Bilecik	Osmaneli	1	1
Ege	İZMİR	Seferihisar	1	1	Marmara	Bilecik	Pazaryeri	2	1
Ege	İZMİR	Selçuk	1	1	Marmara	Bilecik	Söğüt	1	1
Ege	İZMİR	Tire	2	2	Marmara	Bilecik	Yenipazar	2	2
Ege	İZMİR	Torbalı	1	1	Marmara	Bursa	Nilüfer	3	3
Ege	İZMİR	Urla	3	3	Marmara	Bursa	Osmangazi	3	3
Ege	KÜTAHYA	Merkez	1	1	Marmara	Bursa	Yıldırım	3	3
Ege	KÜTAHYA	Altıntaş	2	2	Marmara	Bursa	Büyükorhan	2	2
Ege	KÜTAHYA	Aslanapa	2	2	Marmara	Bursa	Gemlik	1	1
Ege	KÜTAHYA	Çavdarhisar	2	2	Marmara	Bursa	Gürsu	1	1
Ege	KÜTAHYA	Domaniç	2	2	Marmara	Bursa	Harmancık	2	2
Ege	KÜTAHYA	Dumlupınar	2	2	Marmara	Bursa	İnegöl	1	1
Ege	KÜTAHYA	Emet	2	2	Marmara	Bursa	İznik	2	2
Ege	KÜTAHYA	Gediz	2	2	Marmara	Bursa	Karacabey	1	1
Ege	KÜTAHYA	Hisarcık	2	2	Marmara	Bursa	Keles	2	2

Ege	KÜTAHYA	Pazarlar	2	2	Marmara	Bursa	Kestel	1	1
Ege	KÜTAHYA	Simav	2	2	Marmara	Bursa	Mudanya	1	1
Ege	KÜTAHYA	Şaphane	2	2	Marmara	Bursa	M.Kemalpaşa	2	2
Ege	KÜTAHYA	Tavşanlı	2	2	Marmara	Bursa	Orhaneli	2	2
Ege	MANİSA	Merkez	1	1	Marmara	Bursa	Orhangazi	1	1
Ege	MANİSA	Ahmetli	2	1	Marmara	Bursa	Yenişehir	2	2
Ege	MANİSA	Akhisar	2	2	Marmara	ÇANAKKALE	Merkez	1	3
Ege	MANİSA	Alaşehir	2	2	Marmara	ÇANAKKALE	Ayvacık	2	2
Ege	MANİSA	Demirci	2	2	Marmara	ÇANAKKALE	Bayramiç	2	2
Ege	MANİSA	Gölmarmara	2	2	Marmara	ÇANAKKALE	Biga	2	2
Ege	MANİSA	Gördes	2	2	Marmara	ÇANAKKALE	Bozcaada	3	3
Ege	MANİSA	Kırkağaç	2	2	Marmara	ÇANAKKALE	Çan	1	1
Ege	MANİSA	Köprübaşı	2	2	Marmara	ÇANAKKALE	Eceabat	1	1
Ege	MANİSA	Kula	2	2	Marmara	ÇANAKKALE	Ezine	2	2
Ege	MANİSA	Salihli	1	1	Marmara	ÇANAKKALE	Gelibolu	1	1
Ege	MANİSA	Sarıgöl	2	2	Marmara	ÇANAKKALE	Gökçeada	1	1
Ege	MANİSA	Saruhanlı	2	2	Marmara	ÇANAKKALE	Lapseki	2	2
Ege	MANİSA	Selendi	2	2	Marmara	ÇANAKKALE	Yenice	2	2
Ege	MANİSA	Soma	1	1	Marmara	EDİRNE	Merkez	3	3
Ege	MANİSA	Turgutlu	1	1	Marmara	EDİRNE	Enez	2	2
Ege	UŞAK	Merkez	1	1	Marmara	EDİRNE	Havsa	2	2
Ege	UŞAK	Banaz	2	2	Marmara	EDİRNE	İpsala	2	2
Ege	UŞAK	Eşme	2	2	Marmara	EDİRNE	Keşan	1	1
Ege	UŞAK	Karahallı	2	2	Marmara	EDİRNE	Lalapaşa	2	2
Ege	UŞAK	Sivaslı	2	2	Marmara	EDİRNE	Meriç	2	2
Ege	UŞAK	Ulubey	2	2	Marmara	EDİRNE	Süleoğlu	2	2
Güneydoğu Anadolu	DİYARBAKIR	Merkez	1	1	Marmara	EDİRNE	Uzunköprü	1	1
Güneydoğu Anadolu	DİYARBAKIR	Bismil	2	2	Marmara	İSTANBUL	Adalar	3	3
Güneydoğu Anadolu	DİYARBAKIR	Çermik	2	2	Marmara	İSTANBUL	Avcılar	3	3
Güneydoğu Anadolu	DİYARBAKIR	Çınar	2	2	Marmara	İSTANBUL	Bağcılar	3	3
Güneydoğu Anadolu	DİYARBAKIR	Çüngüş	2	2	Marmara	İSTANBUL	Bahçelievler	3	3
Güneydoğu Anadolu	DİYARBAKIR	Dicle	2	2	Marmara	İSTANBUL	Bayrampaşa	3	3
Güneydoğu Anadolu	DİYARBAKIR	Eğil	2	2	Marmara	İSTANBUL	Beykoz	3	3
Güneydoğu Anadolu	DİYARBAKIR	Ergani	2	2	Marmara	İSTANBUL	Beyoğlu	3	3
Güneydoğu Anadolu	DİYARBAKIR	Hani	2	2	Marmara	İSTANBUL	Esenler	3	3
Güneydoğu Anadolu	DİYARBAKIR	Hazro	2	2	Marmara	İSTANBUL	Eyüp	3	3
Güneydoğu Anadolu	DİYARBAKIR	Kocaköy	2	2	Marmara	İSTANBUL	Fatih	3	3
Güneydoğu Anadolu	DİYARBAKIR	Kulp	2	2	Marmara	İSTANBUL	Gaziosmanpaşa	3	3
Güneydoğu Anadolu	DİYARBAKIR	Lice	2	2	Marmara	İSTANBUL	Güngören	3	3
Güneydoğu Anadolu	DİYARBAKIR	Silvan	2	2	Marmara	İSTANBUL	Kağıthane	3	3
Güneydoğu Anadolu	GAZİANTEP	Şahinbey	1	1	Marmara	İSTANBUL	Kartal	3	3
Güneydoğu Anadolu	GAZİANTEP	Şehitkamil	1	1	Marmara	İSTANBUL	Küçükçekmece	3	3
Güneydoğu Anadolu	GAZİANTEP	Araban	2	2	Marmara	İSTANBUL	Maltepe	3	3
Güneydoğu Anadolu	GAZİANTEP	İslahiye	2	2	Marmara	İSTANBUL	Pendik	3	3
Güneydoğu Anadolu	GAZİANTEP	Karkamış	2	2	Marmara	İSTANBUL	Sarıyer	3	3
Güneydoğu Anadolu	GAZİANTEP	Nizip	1	1	Marmara	İSTANBUL	Şişli	3	3
Güneydoğu Anadolu	GAZİANTEP	Nurdağı	2	2	Marmara	İSTANBUL	Tuzla	3	3
Güneydoğu Anadolu	GAZİANTEP	Oğuzeli	2	2	Marmara	İSTANBUL	Ümraniye	3	3
Güneydoğu Anadolu	GAZİANTEP	Yavuzeli	2	2	Marmara	İSTANBUL	Üsküdar	3	3
Güneydoğu Anadolu	MARDİN	Merkez	1	1	Marmara	İSTANBUL	Zeytinburnu	3	3
Güneydoğu Anadolu	MARDİN	Dargeçit	2	2	Marmara	İSTANBUL	Büyükçekmece	3	3
Güneydoğu Anadolu	MARDİN	Derik	2	2	Marmara	İSTANBUL	Çatalca	1	1

Güneydoğu Anadolu	MARDİN	Kızıltepe	2	2	Marmara	İSTANBUL	Silivri	1	1
Güneydoğu Anadolu	MARDİN	Mazıdağı	2	2	Marmara	İSTANBUL	Sultanbeyli	3	3
Güneydoğu Anadolu	MARDİN	Midyat	2	2	Marmara	İSTANBUL	Şile	1	1
Güneydoğu Anadolu	MARDİN	Nusaybin	1	1	Marmara	KIRKLARELİ	Merkez	1	1
Güneydoğu Anadolu	MARDİN	Ömerli	2	2	Marmara	KIRKLARELİ	Babaeski	1	1
Güneydoğu Anadolu	MARDİN	Savur	2	2	Marmara	KIRKLARELİ	Demirköy	2	2
Güneydoğu Anadolu	ŞANLIURFA	Merkez	1	1	Marmara	KIRKLARELİ	Kofçaz	2	2
Güneydoğu Anadolu	ŞANLIURFA	Akçakale	2	2	Marmara	KIRKLARELİ	Lüleburgaz	1	1
Güneydoğu Anadolu	ŞANLIURFA	Birecik	2	2	Marmara	KIRKLARELİ	Pehlivanköy	1	1
Güneydoğu Anadolu	ŞANLIURFA	Bozova	2	2	Marmara	KIRKLARELİ	Pınarhisar	1	1
Güneydoğu Anadolu	ŞANLIURFA	Ceylanpınar	2	2	Marmara	KIRKLARELİ	Vize	2	2
Güneydoğu Anadolu	ŞANLIURFA	Halfeti	2	2	Marmara	KOCAELİ	Merkez	1	1
Güneydoğu Anadolu	ŞANLIURFA	Harran	2	2	Marmara	KOCAELİ	Gebze	1	1
Güneydoğu Anadolu	ŞANLIURFA	Hilvan	2	2	Marmara	KOCAELİ	Gölcük	1	1
Güneydoğu Anadolu	ŞANLIURFA	Siverek	2	2	Marmara	KOCAELİ	Kandıra	2	2
Güneydoğu Anadolu	ŞANLIURFA	Suruç	2	2	Marmara	KOCAELİ	Karamürsel	1	1
Güneydoğu Anadolu	ŞANLIURFA	Viranşehir	2	2	Marmara	KOCAELİ	Körfez	3	3
Güneydoğu Anadolu	BATMAN	Merkez	1	1	Marmara	SAKARYA	Merkez	1	1
Güneydoğu Anadolu	BATMAN	Beşiri	2	2	Marmara	SAKARYA	Ferizli	1	1
Güneydoğu Anadolu	BATMAN	Gercüş	2	2	Marmara	SAKARYA	Söğütlü	2	2
Güneydoğu Anadolu	BATMAN	Hasankeyf	2	2	Marmara	SAKARYA	Akyazı	2	2
Güneydoğu Anadolu	BATMAN	Kozluk	2	2	Marmara	SAKARYA	Geyve	2	2
Güneydoğu Anadolu	BATMAN	Sason	2	2	Marmara	SAKARYA	Hendek	2	2
İç Anadolu	Ankara	Altındağ	3	3	Marmara	SAKARYA	Karapürçek	2	2
İç Anadolu	Ankara	Etimesgut	3	3	Marmara	SAKARYA	Karasu	2	2
İç Anadolu	Ankara	Gölbaşı	1	1	Marmara	SAKARYA	Kaynarca	2	2
İç Anadolu	Ankara	Keçiören	3	3	Marmara	SAKARYA	Kocaali	2	2
İç Anadolu	Ankara	Mamak	3	3	Marmara	SAKARYA	Pamukova	1	1
İç Anadolu	Ankara	Sincan	3	3	Marmara	SAKARYA	Sapanca	1	3
İç Anadolu	Ankara	Yenimahalle	3	3	Marmara	SAKARYA	Taraklı	2	2
İç Anadolu	Ankara	Akyurt	1	1	Marmara	TEKİRDAĞ	Merkez	1	1
İç Anadolu	Ankara	Ayaş	1	1	Marmara	TEKİRDAĞ	Çerkezköy	1	1
İç Anadolu	Ankara	Bala	2	2	Marmara	TEKİRDAĞ	Çorlu	1	1
İç Anadolu	Ankara	Beypazarı	1	1	Marmara	TEKİRDAĞ	Hayrabolu	2	2
İç Anadolu	Ankara	Çamlıdere	1	1	Marmara	TEKİRDAĞ	Malkara	2	2
İç Anadolu	Ankara	Çubuk	1	1	Marmara	TEKİRDAĞ	Marmaraereğlisi	1	1
İç Anadolu	Ankara	Elmadağ	1	1	Marmara	TEKİRDAĞ	Muratlı	1	1
İç Anadolu	Ankara	Evren	1	1	Marmara	TEKİRDAĞ	Saray	1	1
İç Anadolu	Ankara	Güdül	2	2	Marmara	TEKİRDAĞ	Şarköy	1	1
İç Anadolu	Ankara	Haymana	2	2	Marmara	YALOVA	Merkez	3	3
İç Anadolu	Ankara	Kalecik	2	2	Marmara	YALOVA	Altınova	1	1
İç Anadolu	Ankara	Kazan	1	1	Marmara	YALOVA	Armutlu	1	1
İç Anadolu	Ankara	Kızılcahamam	1	1	Marmara	YALOVA	Çınarcık	1	1
İç Anadolu	Ankara	Nallıhan	1	1	Marmara	YALOVA	Çiftlikköy	1	1
İç Anadolu	Ankara	Polatlı	1	1	Marmara	YALOVA	Termal	1	1
İç Anadolu	Ankara	Ş.Koçhisar	1	1					

EK 5 Veri Setlerinin Tanıtılması

Tezimin bu bölümünde, çeşitli uygulamalarda kullanılan süsen veri seti tanıtılacaktır. Tezde kullanılan süsen veri seti internette çeşitli adreslerden bulunabilmektedir. Ancak tezimde kullanılan süsen veri seti MATLAB binary formatındadır. (MAT-files)

EK 5.1 Süsen Veri Seti (İris Data Set)

Kuzey Amerika da yaşayan süsen bitkisinin setosa, versicolor ve virginica cinslerine ait 50 şer gözlemden oluşan toplam 150 gözlemli veri seti, 1935 yılında Anderson tarafından toplanmış olup 1936 yılında ilk defa Fisher tarafından analiz edilmiştir. Bu yüzden süsen veri seti, Fisher iris veri seti olarak da bilinmektedir. Süsen veri seti, sırasıyla çanakyapraklarının (sepal) ve taçyapraklarının (petal) uzunluk (length) ve genişliklerini (width), santimetre cinsinden gösteren 4 değişken ve 150 gözlemden oluşmaktadır.

Matlab' da süsen veri seti hazır bir şekilde bulunmaktadır. Veri setini kullanabilmek için `load fisheriris` komutu yazılarak veri setinin çağrılması gerekmektedir. Süsen veri setinin çağrılmasıyla meas ve specias diye iki değişken gelmektedir. Burada meas değişkeni süsen verisinin sayısal değerleri olan 4 sütunlu, 150 satırlı bir veri matrisidir. species değişkeni ise 150 bitkinin bulundukları cinsleri gösteren 1 sütunlu 150 satırlı hücre dizisidir. Meas değişkenin ilk sütunu sepal length, ikinci sütunu sepal width, üçüncü sütunu petal length ve dördüncü sütunu petal width' e karşılık gelmektedir. species hücre dizisinin ilk 50' si setosa, ,ikinci 50' si versicolor ve üçüncü 50' si virginica cinslerine karşılık gelmektedir.

		
Süsen Setosa	Süsen Versicolor	Süsen Virginica

Şekil Ek 5.1 Süsen bitkisinin setosa, versicolor ve virginica cinsleri

EK 6 MATLAB R2007a Fonksiyonlarının Tanıtılması

Tezimin bu bölümünde, çeşitli uygulamalarda kullanılan MATLAB R2007a komutları tanıtılacaktır.

EK 6.1 Temel Fonksiyonlar

clear fonksiyonu, çalışma alanındaki değişken değerlerini siler.

clc fonksiyonu, çalışma alanını temizler

load fonksiyonu, verilerin, önceden kaydedilmiş bilgilerin çalışma alanına tekrar yüklenmesini sağlar.

size(x) fonksiyonu, x matrisinin satır ve sütun sayısını verir.

find, fonksiyonu matrislerde ve dizilerde mantıksal eşitlikler aracılığıyla eleman araması yapmaya yarayan bir fonksiyondur.

zeros(n,n) fonksiyonu, nxn boyutunda 0' lardan oluşan matris üretir.

length(x) fonksiyonu x matrisinin satır sayısını yani uzunluğunu verir.

mod(x,y) fonksiyonu x sayısının y sayısına göre modunu alır.

sum (x) fonksiyonu x matrisinin sütun toplamlarını verir.

mvnrnd(mu,sigma,n); ortalaması mu, kovaryansı matrisi sigma olan çok değişkenli normal dağılımdan n adet sayı üretir

corr(x), fonksiyonu x matrisinin sütunları arasındaki korelasyon katsayılarını hesaplar.

mean(x) fonksiyonu x matrisinin her bir sütunu için aritmetik ortalamalar hesaplar.

p=randperm(n) fonksiyonu n sayı değerleri için hesapladığı tesadüfi permutasyon değerlerini p değişkenine verir.

for döngüsü

for döngüsü; ifadelerin kullanıcı tarafından belirlenen sayıdaki tekrarının söz konusu olduğu durumlarda kullanılır. for döngüsü genel olarak;

```
for değişken=deger
      ifadeler     ifadeler bloğu
end
```

şeklindeki yapıya sahiptir. Bu yapıda değişken, döngü değişkenidir. deger ise, döngünün değişkene eşitliğinin kontrol edildiği diğer bir ifadedir. Program akışında, her bir durum için döngü değişkenin durumu kontrol edilir ve ifadeler bloğunda yer alan komutlar ve ifadeler işletilir. deger dizisinin her değeri için bir kez döngü işletilir. deger ifadesi, çoğu zaman ilk_deger:artıs_miktarı:son_deger şeklinde belirtilmiş bir dizidir.

if şartlı deyimi

```
if durum1
      ifadeler     1. ifadeler bloğu
elseif durum2
      ifadeler     2. ifadeler bloğu
else
      ifadeler     3. ifadeler bloğu
end
```

durum1' in değerlendirilmesi sonucu üretilen cevap 0' dan farklı bir değere sahipse (if şartlı değimiyle önerilen durum doğru ise) program, 1. blokta bulunan ifadeleri işletir. Bunun dışındaki durumlar için program, durum2' yi denetler. Durum2, eğer 1 cevabını üretirse MATLAB, 1. bloktaki ifadeleri işletmeden atlar ve 2. bloktaki ifadeleri yürütür. Eğer durum1 ve durum2' nin denetlenmesi sonucu üretilen cevaplar 0 ise, program else deyimini takip eden bloktaki ifadeleri işletir. Çünkü else deyimine, ancak if ve elseif deyimlerinin tamamının 0 cevabını üretmeleri sonucunda başvurulabilir.

EK 6.2 Grafik fonksiyonları

scatter(x, y) fonksiyonu x ve y vektörleri arasındaki serpilme grafiğini çizer.

scatter3(x, y) fonksiyonu x ve y vektörleri arasındaki üç boyutlu serpilme grafiğini çizer.

boxplot(x) fonksiyonu x veri matrisinin sütunları için kutu grafiği çizer.

plot fonksiyonu, y bir vektör olmak üzere plot(y) şeklinde kullanıldığında y vektörü elemanlarının konumları x ekseni, vektör elemanları değerleri de y ekseni kabul edilerek bir grafik çizer. x ve y birer vektör olmak üzere plot(x,y) y verisinin x' e göre değişim grafiğini çizer.

axis([xmin xmax ymin ymax]) fonksiyonu, x ekseni [xmin xmax] ve y ekseni [ymin ymax] aralıklarında olmak üzere grafikleri yapılandırır.

axis off fonksiyonu grafik penceresindeki eksen takımlarını, etiketleri ve eksen üzerindeki sayıları kaldırır.

grid on fonksiyonu, grafik üzerindeki bölmelendirme çizgilerinin görüntülenmesini sağlar

subplot(a,b,c) fonksiyonu, aynı anda birden fazla grafik açmak ve bunlardan farklı fonksiyonlarının grafiklerinin görüntülenmesini sağalar. Bu fonksiyonun işletilmesiyle grafik penceresi, a*b (a;satır ve b;sütun) olacak şekilde yapılandırılır. c ise, grafik komutlarının işletileceği alt pencerelerinin numaralarıdır.

figure fonksiyonu yeni boş bir grafik penceresi açar. Artık grafik işlemleri açılan bu son boş grafik üzerinde yapılır.

title komutu grafiğe bir başlık oluşturmak için kullanılır.

xlabel komutu x eksenini isimlendirmek için kullanılır.

ylabel komutu, y eksenini isimlendirmek için kullanılır.

text(x, y, 'string') fonksiyonu x ve y koordinatları için string değerlerini etiket olarak grafiğe yapıştırır.

gplotmatrix(x, x, g) fonksiyonu x ve y matrislerinin sütunları arasındaki g grup özelliklerine göre serpilme grafikleri çizer.

MATLAB' de temel bileşenler analizi **[coefs, score, variances,t2] = princomp(X)** ve **[coefs, variances, explained] = pcacov(X)** fonksiyonları kullanılarak yapılır.

[coefs, score, variances,t2] = princomp(X) fonksiyonu X veri matrisinin temel bileşenler katsayılarını coefs değişkenine, kovaryans matrisinin özdeğerlerini variances değişkenine, temel bileşen skorlarını standardize z-skorları cinsinden Score değişkenine ve her bir veri noktasındaki Hotelling T^2 değerlerini t2 değişkenine verir.

biplot(coefs, 'scores', score) fonksiyonu, temel bileşenler analizi sonucu elde edilen temel bileşen katsayılar matrisi coefs kullanarak biplot grafiğini çizer. biplot grafiğinde koordinat eksenleri temel bileşenleri ve grafikteki nokta değerleri temel bileşenlerin skor değerlerini gösterirler. Burada orijinal değişkenler biplot grafiğinde vektörlerle gösterilirler.

biplot(coefs, 'scores', score,' varlabels', varlabels) fonksiyonu kullanarak biplot grafiğinde bulunan vektörlere varlabel sitring matrisinde bulunan değerler etiket olarak konulabilir.

parallelcoords(x) fonksiyonuyla x veri matrisinin paralel koordinat grafiği çizilir. Paralel koordinat grafiğinde x veri matrisimizin gözlem değerleri y ekseninde, sütunları x eksenindedir.

parallelcoords(x, 'Group', group) fonksiyonu, x veri matrisimizde farklı gruplarda bulunan değerleri farklı renklerde göstererek paralel koordinat grafiği çizer. Burada gruplar group değişkeniyle tanımlanırlar. group değişkeninde gözlemlerin hangi gruplarda bulunduğunu gösteren sayısal ya da sitring ifadeler bulunur.

andrewsplot(x) fonksiyonu kullanılarak x veri matrisinin Andrews eğrileri çizilir. Andrews eğrilerinde x veri matrisimizin gözlem değerleri y ekseninde, sütunları x eksenindedir.

andrewsplot(x, 'Group', group) fonksiyonu, x veri matrisimizde farklı gruplarda bulunan değerleri farklı renklerde göstererek Andrews eğrilerini çizer. Burada gruplar group değişkeniyle tanımlanırlar. group değişkeninde gözlemlerin hangi gruplarda bulunduğunu gösteren sayısal ya da sitring ifadeler bulunur.

glyphplot(x) fonksiyonu, çok değişkenli x veri matrisi için yıldız grafikleri çizer. Burada x veri matrisinin her satırı yani her gözlem değeri için yıldız grafikleri çizilir. gyphplot fonksiyonu, x veri matrisinin sütunlarını [0,1] aralığında olacak şekilde standartlaştırma yaparak yıldız grafikleri çizer. glyphplot fonksiyonu yıldız grafiği çizimi için glyhplot(x, 'Glyph', 'star') şeklinde de kullanılabilir.

glyphplot(x, 'Glyph', 'face') fonksiyonu, x veri matrisi için Chernoff yüzleri çizer. Burada x veri matrisinin her satırı yani her gözlem değeri için yüzler çizilir. Çizilen yüzlerin taşıdığı göz büyüklüğü, burun uzunluğu gibi yüz özellikleri, gözlemlere karşılık gelen değişkenler yani x veri matrisinin sütunlardır. Chernoff yüzlerinin taşıdığı 17 tane yüz özelliği bulunmaktadır. Bu 17 tane yüz özelliği, aşağıda Tablo Ek 6.1 de yanlarında numaraları bulunacak biçimde sıralanmıştır.

glyphplot(x,'Glyph','face','Features',F) fonksiyonu, x veri matrisinin i. sütunu için seçilen F yüz özeliğine karşılık gelen Chernoff yüzlerini çizer. Burada F, 0 ile 17 arasında sayısal değerleri alabilen bir vektör olmalıdır. F değişkenin kullanılmadığı durumlarda default olarak 1'den 17 kadar sıralı numaralı özellikler dikkate alınarak Chernoff yüzleri çizilir.

glyphplot(x,...,'ObsLabels',labels) fonksiyonu x veri matrisi için çizdiği Chernoff yüzlerine labels string değişkeninde bulunan isimleri verir. labels değişkenin kullanılmaması durumunda default olarak yüzlere sıra numara değerleri verilir.

glyphplot(X,...,'Standardize',method) fonksiyonu x veri matrisinin sütunlarını [0,1] aralığına karşılık gelecek şekilde standartlaştırma yaparak Chernoff yüzlerini çizer. Burada metot değişkeni yerine 'off' yazılarak standartlaştırma işleminden vazgeçilebilir.

Tablo Ek 6.1 glyphplot yüz özellikleri

Sütunlar	Yüz Özellikleri
1	Yüz Hacmi
2	Alın, çene bağlantılı kavis uzunluğu
3	Alın biçimi
4	Çene biçimi
5	Gözler arası genişlik
6	Gözlerin dikey pozisyonu
7	Gözlerin yüksekliği
8	Gözlerin genişliği (bu kaşların genişliğini etkiler)
9	Gözlerin sivriliği (bu kaşların sivriliğini etkiler)
10	Kaşların dikey pozisyonu
11	Kaşların genişliği (gözlere bağlıdır)
12	Kaşların sivriliği (gözlere bağlıdır)
13	Gözbebekleri yönü
14	Burun uzunluğu
15	Ağzın dikey pozisyonu
16	Ağız biçimi
17	Ağız uzunluğu

imagesc(c) komutu, c veri matrisi için renkli matris grafiği çizer. Burada c veri matrisinde bulunan her değerin büyüklüğü matris grafiğinde renkli karelerle temsil edilirler. Matris grafiğinde bulunana colorbar' da renklerin sayısal değerleri gösterilmektedir.

colormap(gray(256)) fonksiyonu, renkli çizilen matris grafiklerini gri tonlarda yeniden renklendirir.

EK 6.3 Kümeleme Analizi Fonksiyonları

pdist(x) fonksiyonu, x veri matrisinde bulunan ikili gözlemler arasında ki öklid uzaklıklarını hesaplar. m gözlemden yani satırdan oluşan x veri matrisi için pdist(x) fonksiyonu, hesapladığı uzaklıkları m*(m-1)/2 uzunluğunda bir vektör olarak verir. Uzaklıklar matrisinin elde edilmesi için pdist fonksiyonun geri dönüş değerlerinin squareform komutuna giriş olacak şekilde kullanılması gerekmektedir. Bu sayede x veri matrisinde bulunan gözlemlerin bir birilerine olan uzaklıkları matris şeklinde elde edilebilir.

pdist(x,metric) fonksiyonu metric parametresinde verilen yönteme göre x veri matrisinde bulunan gözlemler arasındaki uzaklıkları ya da benzerlikleri hesaplar. metric parametresi Tablo Ek 6.2' de bulunan string idalerini alabilir.

Tablo Ek 6.1 pdist komutunda kullanılan uzaklıklar/benzerlikler

metric parametresi	**Yöntem**
'euclidean'	Öklid uzaklığı (önceden tanımlı)
'seuclidean'	Pearson (Standart) öklid uzaklığı
'mahalanobis'	Mahalanobis uzaklığı
'cityblock'	Manhattan uzaklıkğı (City block)
'minkowski'	Minkowski uzaklığı
'cosine'	Bir eksi açısal benzerlik ölçüsü
'correlation'	Bir eksi korelasyon benzerlik ölçüsü

Z = linkage(y) fonksiyonu, pdist fonksiyonuyla bulunan y uzaklıklar vektörü için tek bağlantılı hiyerarşik kümeleme yapar. m birimden olaşan verimiz için çıkış değişkeni Z, (m-1) satır ve 3 sütundan oluşan hiyerarşik kümeleme analizi matrisidir. Z matrisinin ilk iki sütununda hiyerarşik kümelemede birleştirilecek birimleri, 3. sütunda birleştirilen birimlerin birleşme uzaklıkları yer almaktadır. Birleştirilen birimler daha sonra m+1, m+2 gibi numaralar alırlar.

linkage(y, method) fonksiyonu, method parametresinde verilen yönteme göre hiyerarşik kümeleme yapar. method parametresi Tablo Ek 6.3' de bulunan string ifadelerini alabilir.

linkage(y, method, metric) fonksiyonu, pdist fonksiyonunda olduğu gibi metric parametresine göre uzaklık değerlerini kullanarak hiyerarşik kümeleme yapar.

Tablo Ek 6.2 linkage fonksiyonunda kullanılan yöntemler

method parametresi	**Yöntem**
'single'	Tek bağlantılı
'complete'	Tam bağlantılı

H = dendrogram(Z) fonksiyonu, linkage fonksiyonun ürettiği Z matrisi için dendrogram grafiği çizer. Ancak burada çizilen dendrogram maksimum 30 birim için oluşturulur. dendrogram fonksiyonun ürettiği H geri dönüş vektörü, çizilen dendrogramdaki çizgilere karşılık gelen sayısal numaralar içerir.

H = dendrogram(Z, p) fonksiyonu istenilen p adet birim için dendrogram grafiği çizer. Eğer p yerine 0 değeri yazılırsa dendrogram fonksiyonu tüm birimler için dendrogram grafiği çizer.

[H, T, perm] = dendrogram(Z, p) fonksiyonu, çizdiği dendrogramdaki birim numaralarını (etiketlerini) perm vektörüne atar. T vektörü ise 1' den p' ye kadar sıralanmış olan sayıları içerir.

[…] = dendrogram(…, 'colorthreshold', 'default') fonksiyonu, çizdiği dendrogramı küme yapılarına göre renklendirir.

[…] = dendrogram(…, 'orientation','orient') fonksiyonu orient parametresinin aldığı değerlere göre hiyerarşik kümelemenin kökünü konumlandırarak dendrogram grafiği çizer. orient parametresi, 'top', 'bottom', 'left', 'right' değerlerini aldığında sırasıyla hiyerarşik kümelemenin kökü yukarıda, aşağıda, solda ve sağda olacak şekilde denrogram grafiği çizilir.

[…] = dendrogram(…, 'labels', S) fonksiyonu, çizdiği dendrogram grafiğindeki birim numaraları yerine S karakter dizisi yada sitring hücre dizisi içinde yer alan değerleri yerleştirir.

T=kmeans(x, k) fonksiyonu x ver matrisini k-ortalamalar kümeleme yöntemiyle k adet kümeye ayırır ve küme numaralarını T değişkenine verir.

cids=cluster(z,'maxclust',n) fonksiyonu, linkage fonksiyonun ürettiği z matrisini maksimum n olacak şekilde kümeye ayırır ve küme numaralarını cids değişkenine verir.

[sil,h] = silhouette(x,cids) fonksiyonu x veri matrisi için kullanılan hiyerarşik, k-ortalamalar ve SOM gibi kümeleme yöntemlerinin ürettiği cids küme numaralarını kullanarak silhouette endekslerini hesaplar ve sil değişkenine atar. Burada bulunan h değişkeni silhouette grafiğinin çizilmesine neden olur. h değişkeni olmazsa silhouette grafiği çizilmez.

[Ts,inds]=sort(T) fonksiyonu hiyerarşik, k-ortalamalar ve SOM gibi kümeleme yöntemlerinin ürettiği T küme numaralarını küçükten büyüğe dizerek Ts değişkenine ve T değişkenlerinin sıra numaralarını inds değişkenine verir.

EK 6.4 Exploratory Data Analysis Toolbox Fonksiyonları

treemap(z,n) fonksiyonu, linkage fonksiyonun ürettiği z matrisini kullanarak ilk n kutudan oluşan ağaç grafiği çizer.

rectplot(z,n,'nclus') fonksiyonu, linkage fonksiyonun ürettiği z matrisinin ilk n kümesini gösterecek şekilde rectangle grafiği çizer.

reclus(cids,a) fonksiyonu hiyerarşik, k-ortalamalar ve SOM gibi kümeleme yöntemlerinin ürettiği cids küme numaraları için reclus grafiği çizer. reclus grafiğinde bulunan kümelere a

vektöründe bulunan değerlerler etiket olarak verilir. reclus(cids,a,s) fonksiyonu s değişkeninde yer alan, her küme birimleri için hesaplanan, Silhouette küme doğruluk endeks değerlerini renkli piksellerle göstererek reclus grafiğinde gösterir.

permtourparallel(x), x veri matrisinin paralel koordinatları için kısmi permutasyon turları grafiği çizer.

permtourandrews(x), x veri matrisinin Andrews eğrileri için permutasyon turları grafiğini çizer.

EK 6.5 Somtoolbox Fonksiyonları

som_read_data('dosyaismi.data') fonksiyonu tırnak içinde yazılan data uzantılı dosyayı okur.

som_normalize(sD,'var') fonksiyonu sD veri setini ortalaması 0, standart sapması 1 olacak şekilde standartlaştırarak SOM sinir ağını oluşturur.

som_make(sD,['seq']) fonksiyonu sD sinir ağını eğitir.

som_autolabel(sM,sD,'vote') fonksiyonu eğitilmiş sM ağına sD' deki etiket isimlerini ekler.

som_show(sM,'umat','all') fonksiyonu eğitilmiş sM ağı için U matris grafiğini çizer.

som_show(sM,'umat','all','comp',1:n,'empty','Labels','norm','d') fonksiyonu eğitilmiş sM ağı için U matrisi yanında n tane değişken için SOM haritaları da çizer.

som_show_add('label',sM,'subplot',n) fonksiyonu çizilen n numaralı SOM grafiklerine etiketler yerleştirir.

[codes,T1,err] = som_kmeans('seq',scores(:,1:10), k, [epochs]) fonksiyonu D veri matrisi içi epochs sayısı kadar döngüde k küme sayısını bulur. som_kmeans fonksiyonu ürettiği refarans vektörlerini codes, küme numaralarunu T1 ve toplam quantization hatalarını err değişkenine verir.

ÖZGEÇMİŞ

Doğum tarihi	05.11.1984	
Doğum yeri	İstanbul	
Lise	1998-2001	Yeşilköy 50. Yıl Lisesi
Lisans	2001-2006	Yıldız Teknik Üniversitesi Fen-Edebiyat Fak. İstatistik Bölümü
Lisans	2004-2007	Yıldız Teknik Üniversitesi Fen-Edebiyat Fak. Matematik Bölümü
Yüksek Lisans	2006-2008	Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü İstatistik Anabilim Dalı, İstatistik Programı

Çalıştığı kurumlar

	2007-2007	Seri Bilgi Teknolojisi Limitet Şirketi (IBM Company)