T.C.

MARMARA ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
EKONOMETRİ ANABİLİM DALI
İSTATİSTİK BİLİM DALI

ÇOK BOYUTLU BİRLİKTELİK KURALLARI ANALİZİ VE İŞLETME UYGULAMASI

Yüksek Lisans Tezi

ŞENGÜL GEDLEÇ

İstanbul, 2019

T.C.
MARMARA ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
EKONOMETRİ ANABİLİM DALI
İSTATİSTİK BİLİM DALI

ÇOK BOYUTLU BİRLİKTELİK KURALLARI ANALİZİ VE İŞLETME UYGULAMASI

Yüksek Lisans Tezi

ŞENGÜL GEDLEÇ

Tez Danışmanı

Prof. Dr. AHMET METE ÇİLİNGİRTÜRK

İstanbul, 2019

T.C.
MARMARA ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ MÜDÜRLÜĞÜ

TEZ ONAY BELGESİ

EKONOMETRİ Anabilim Dalı İSTATİSTİK Bilim Dalı TEZLİ YÜKSEK LİSANS öğrencisi ŞENGÜL GEDLEÇ'nın ÇOK BOYUTLU BİRLİKTELİK KURALLARI ANALİZİ VE İŞLETME UYGULAMASI adlı tez çalışması, Enstitümüz Yönetim Kurulunun 18.04.2019 tarih ve 2019-10/23 sayılı kararıyla oluşturulan jüri tarafından oy birliği / ~~oy çokluğu~~ ile Yüksek Lisans Tezi olarak kabul edilmiştir.

Tez Savunma Tarihi 08/05/2019

Öğretim Üyesi Adı Soyadı			**İmzası**
1.	Tez Danışmanı	Prof. Dr. AHMET METE ÇİLİNGİRTÜRK	
2.	Jüri Üyesi	Doç. Dr. SELAY GİRAY YAKUT	
3.	Jüri Üyesi	Dr. Öğr. Üyesi YELİZ SEVİMLİ SAİTOĞLU	

ÖNSÖZ

Öncelikle tez konusunun seçiminden hazırlama sürecine kadar beni her zaman destekleyen, zaman, tecrübe ve bilgi birikimini paylaşan, eleştirel bakış açısı kazanmamı sağlayan ve değerli bilgileri ile bana yol gösteren saygıdeğer tez danışmanım Prof. Dr. Ahmet Mete ÇİLİNGİRTÜRK' e minnet ve şükranlarımı sunarım.

Lisans ve yüksek lisans öğrenimim boyunca eğitimime önemli katkıları olan saygıdeğer hocam Doç. Dr. Selay GİRAY YAKUT' a teşekkürü bir borç bilirim.

Tezin savunulması anında olumlu geri bildirimleriyle katkı sağlayan saygıdeğer hocam Sayın Dr. Öğretim Üyesi Yeliz Sevimli SAİTOĞLU' na teşekkürü bir borç bilirim.

Son olarak çalışma sürecinin her anında bana inanan, güvenen, destek veren ve katkıları paha biçilemeyen annem Hediye GEDLEÇ' e, abim Arif GEDLEÇ' e, değerli dostlarım Derya SANIR ve Tuğçe ÜNALDI' ya sonsuz teşekkür ederim.

Şengül GEDLEÇ

İstanbul, 2019

İÇİNDEKİLER

TABLOLAR LİSTESİ

ŞEKİLLER LİSTESİ

GRAFİKLER LİSTESİ

KISALTMALAR LİSTESİ

AGNES	Birleştirici Kümeleme Teknikleri
AID	Automatic Interaction Detector Algorithm
bs.	Baskı, Basım
B3LAB	Bulut Bilişim ve Büyük Veri Araştırma Laboratuvarı
c	Güven Değeri
C.	Cilt
C&RT	Classification And Regression Trees Algorithm
CARMA	Sürekli Birliktelik Kuralı Madenciliği Algoritması
CART	Classification And Regression Trees Algorithm
CCPD	Ortak Aday Ürün Kümelerine Bölünmüş Veritabanı
CD	Sayım Dağılımı
CHAID	Chi-Squared Automatic Interaction Detector Algorithm
CLARA	Geniş Uygulamaların Kümelenmesi Algoritması
CLARANS	Rasgele Aramaya Dayalı Geniş Uygulamaları Kümeleme Algoritması
CRM	Müşteri ilişkileri Yönetimi
Çev.	Çeviren
DBMS	Veri Tabanı Yönetim Sistemleri
DBSCAN	Density Based Spatial Clustering of Applications with Noise Algorithm
DD	Veri Dağılımı

DENCLUE	Density Based Cluestering Algorithm
DIANA	Ayırıcı Kümeleme Teknikleri
DIC	Dinamik Ürün Kümesi Sayımı
DM	Veri Madenciliği
DMA	Dağılım Madenciliği Algoritması
DVD	Digital Versatile Disc
SVM	Karar Destek Vektör Makinesi
FP-Growth	Frequent Pattern Growth
FP-Tree	Frequent Pattern Tree
HD	Hibrit Dağılımı
HPA	Bağlantı Kurallarının Çırpı Temelli Paralel Madenciliği Algoritması
IDD	Akıllı Veri Dağılımı
KDD	Veritabanlarında Bilgi Keşfi
KVA	Keşifsel Veri Analizi
LHS	Birliktelik Kurallarında Öncül Kısım
LR	Lojistik Regresyon
MARS	Multivariate Adaptive Regression Splines
MBR	Bellek Tabanlı
MDIM	Multi Dimensional Indexing Mining
MEB	Millî Eğitim Bakanlığı

MEBBİS Millî Eğitim Bakanlığı Bilişim Sistemleri

MEDULA Reçete Onay Sistemi

MHRS Doktor Randevu Sistemi

MGD Minimum Güven Değeri

MDD Minimum Destek Değeri

OTOSINIF AutoClass Algorithm

PAM Temsilciler Etrafında Bölümleme

PAR Paralel Bağlantı Kuralları

PDM Paralel Veri Madenciliği

RARM Rapid Association Rule Mining

RHS Birliktelik Kurallarında Ardıl Kısım

OCD Sıra Dışı Ürün Kümesi Tespiti

OLAP Çevrimiçi Çözümsel İşlemler

OLAM Çevrimiçi Birliktelik Madenciliği

OMARS Çevrimiçi Çok Boyutlu Birliktelik Kuralları Madencilik Sistemi

QUEST Quick, Unbiased, Efficient Statistical Tree

RDBMS Bağıntısal Veri Tabanı Yönetim Sistemleri

RVM Geçerli Vektör Makinesi

s Destek Değeri

s. Sayfa

S.	Sayı
SETM	Set Yönelimli Madencilik
SGK	Sosyal Güvenlik Kurumu
SH	Çarpık İşleme
SLIQ	Supervised Learning In Quest
SPRINT	Scalable Parallelizable Induction of Decision Trees
TB	Terabayt
TID	İşlem Numarası
TÜBİTAK	Türkiye Bilimsel ve Teknolojik Araştırma Kurumu
UBYS	Ulusal Sağlık Bilgi Sistemi
USB	Universal Serial Bus
YSA	Yapay Sinir Ağları

ÖZET

Günümüzde, karışık ve yüksek boyuttaki veri setlerinde saklanan verilerden faydalı ve anlamlı bilgilerin ortaya çıkarılması amacıyla veri madenciliği ve modellerine sık sık başvurulmaktadır. Bu tez çalışmasında veritabanlarında bilgi keşfi süreci, veri madenciliği ve veri madenciliğinde yer alan modeller açıklanmıştır. Tanımlayıcı veri madenciliği modellerinden biri olan Birliktelik Kuralları ve algoritmaları ayrıntılı olarak incelenmiştir. Tezin uygulama bölümünde, Türkiye'de perakende sektöründe yer alan bir işletmeye ait bir aylık satış verileri kullanılmıştır. Satış verilerinde yer alan ürünler arasındaki ilişkiler, "Birliktelik Kuralları" ile "Apriori algoritması" ve "Market Sepet Analizi" çalışması ile tespit edilmiştir. Ürünler arasındaki mevcut birliktelikler ürün satış miktarlarının arttırılması, dolayısı ile gelirde artış sağlamak için kullanılması amaçlanmıştır.

Anahtar Kelimeler: Veritabanlarında Bilgi Keşfi Süreci, Veri Madenciliği, Birliktelik Kuralları, Apriori, Market Sepet Analizi.

ABSTRACT

Nowadays, data mining and models are frequently used in order to reveal useful and meaningful information from the data stored in mixed and high data sets. In this thesis, knowledge discovery process in databases, data mining and data mining models are explained. One of the descriptive data mining models Association Rules and algorithms are examined in detail. In the application part of the thesis, a monthly sales data belonging to a company located in the retail sector in Turkey were used. The relations between the products included in the sales data were determined by "Association Rules" and "Apriori algorithm" and "Market Basket Analysis" study. The existing partnerships between the products are intended to increase the amount of product sales and thus to increase the revenue.

Keywords: Knowledge Discovery Process in Databases, Data Mining, Association Rules, Apriori, Market Basket Analysis.

GİRİŞ

Eski çağlardan bugüne insanoğlu merak ederek çevresini keşfetme eğilimine girmiştir. İnsanlar bu keşfetme sürecinde deneme yanılma yöntemine dayanarak kazandığı tecrübeleri biriktirerek ilerleme kaydetmekle beraber edindiği bilgileri çevresindeki insanlar ile paylaşma gayretine girmiştir. Hayatı kolaylaştıran buluşlar öncesinde insanlar tecrübelerini ve bilgilerini taş ve kâğıt gibi materyaller ile aktarmaktaydılar. Daha sonrasında bilgisayarın icadı ile beraber dünya artık dijitalleşmeye adım atmış ve hızla gelişen teknoloji ürünleri sayesinde bilgiler ölümsüzleştirilebilir bir hale gelmiştir.

Günümüz teknolojisinin her geçen gün hızla geliştiği bu çağda bilgisayar ve iletişim sistemlerinin ucuzlaması ve tüm kitleler tarafından ulaşılabilmesi ile birlikte bu sistemlerin insanoğlunun olduğu her alanda temellerini sağlam atmış oldukları görülmektedir. Sosyal medya ağları, online alışveriş siteleri ve ilkokul sürecine robotik kodlama derslerinin eklenmesi gibi gelişmelerle insanların birçok alanda teknolojiden fazlasıyla beslendiği görülmektedir.

Dünya nüfusun 7 trilyona yaklaştığı günümüzde bilim ve teknolojideki hızlı gelişmeler veri miktarında müthiş bir artışa sebep olmuştur. Veri kaynaklarındaki bu olağanüstü artış, her alanda bu verilerin işlenmesini ve bu verileri en verimli şekilde işleyecek bilgisayar sistemlerin geliştirilmesini gündeme getirmiştir. Bu problemler uzmanları "Veritabanlarında Bilgi Keşfi (Knowledge Discovery in Databases- KDD)" adlı yeni bir arayışa sürüklemiştir. Ham verilerden, güncel ve yararlı bilgilerin üretilmesi yani bir anlam ifade eden parçaların ortaya çıkarılması esnasında yaşanan süreç "Veri Tabanlarında Bilgi Keşfi" olarak adlandırılmaktadır. KDD süreci içinde gerçekleşen "Veri Madenciliği (Data Mining-DM)" anlamlı ve faydalı olacak betimleme, kural ve ilişkilere ulaşmak amacıyla büyük miktardaki verilerin araştırılması ve analiz edilmesidir (Berry & Linoff, 2004, s. 7).

Tıp, ekonomi ve sosyal bilimlerde araştırmalar var olan verilerde saklı olan örüntüler hakkında yeni bir önerme çıkarmak, söz konusu duruma en çok etki eden etkenleri ve bu etkenlerin etki seviyelerini, hangi durumlarda etkinin arttığını keşfetmek amaçlarını gerçekleştirmek için yapılmaktadır (Doğan & Özdamar, 2003, s. 392). Veri madenciliği ise bu çalışmalardaki mevcuttaki büyük ve anlamlandırılmamış veriden anlamlı örüntüler çıkarmayı sağlayan bir disiplindir. Veritabanlarındaki milyonlarca veri üzerinden analiz yapabilmek için bilgisayar programları ve bu programları kullanabilecek nitelikli iş gücü gerekmektedir. Üretilen veriler bilgisayar programlarında belli bir amaca dayalı olarak işlenmesi sonucu anlam ve değer

kazanmaktadırlar. Bu değere yani bilgiye ulaşabilmek için başta veri madenciliği olmak üzere çok yüksek boyuttaki veri setlerini işleyebilecek sistemleri kullanmak gerekmektedir.

Dijital Dünya, tüketim çağı gibi kavramlarının çok fazla kullanıldığı bu dönemlerde insanlar kayda alınan her eylemde aslında veri üretmektedir. İşletmelerin başarılı adımlar atabilmesi için milyonlarca veri yığını arasından kendilerine ışık tutacak bilgiye ulaşması öncelikli öneme sahiptir. İşletmeler de teknolojinin gelişmesi ile beraber bu verileri çeşitli araç ve yöntemler ile kayıt altına almaktadır. Bu kayıtlar aracılığıyla kendi ürün ve hizmetlerini öne çıkaracak fikirler, maliyetlerinde azalış ve karlarında artış sağlayacak planlamaları sağlamaktadırlar. Burada önemli olan kıstas bu büyük verinin sürekli dinamik bir yapıda tutulması ve makine öğrenmesi yönteminin temeline dayanarak veriye dayalı ürünler üretebilmektir.

Bu tezin amacı tanımlayıcı veri madenciliği modellerinden olan "Birliktelik Kuralları" ile müşterilerin alışveriş alışkanlıklarına dair birliktelik kurallarını keşfetmektir. Birliktelik kuralları analizinde sıklıkla başvurulan "Apriori algoritması" aracılığıyla kurallar elde edilmiştir. Bu kurallar aracılığıyla, müşteri profillerinin oluşturulması ve müşteri profillerine uygun satış stratejilerinin geliştirilmesi hedeflenmiştir. Çalışmanın birinci bölümünde veri madenciliği hakkında genel bir literatür çalışması yapılmıştır. Çalışmanın ikinci bölümünde Birliktelik Kuralları modeli hakkında detaylı bilgi verilmiştir. Çalışmanın son bölümünde Türkiye'de perakende sektöründe yer alan bir işletmenin satış verileri üzerinde market sepet analizi çalışması yapılmıştır.

1. VERİ MADENCİLİĞİ

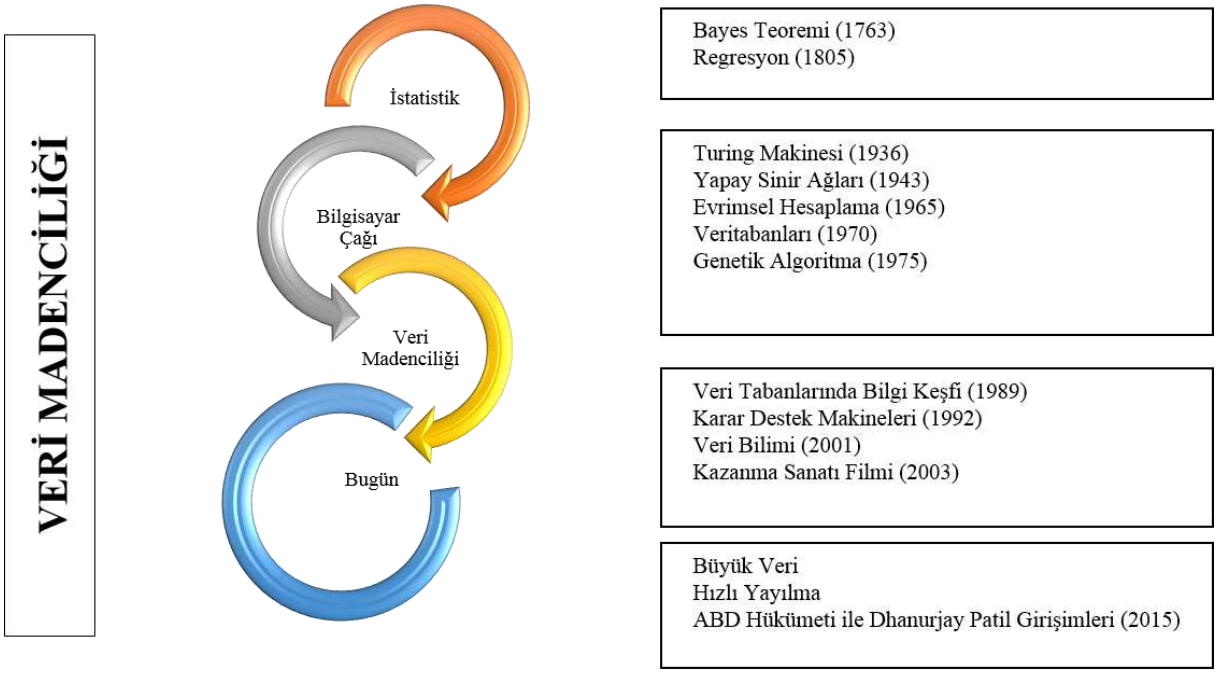
İnsanlığın gelişimi ve icatlar ile beraber bilgi birikimi artma eğiliminde olmuş ve tüm bu buluşların yetersiz kaldığı dönemlerde bilim insanları yeni bir buluş ile geçmiş tecrübeleri daha kullanılabilir hale getirmişlerdir. Bu imkân 1950'li yıllarda basit sayma ve aritmetik işlemler yapmak amacı ile bugünkü bilgisayar boyutlarına göre devasa bir büyüklüğe sahip olan ilk bilgisayar ile gerçekleşmiştir. İkinci Dünya Savaşı'nın Dünya tarihi açısından en büyük getirisi John Mauchy ve John Presper Eckert'in ABD ordusuna sunduğu ENIAC (Electrical Numerical Integrator and Calculator) adlı ilk bilgisayardır. Von Neumann (1945) mimarisindeki yapı ile geliştirilen ENIAC, bugün kişisel bilgisayar olarak kullanılmaktadır (Karaibrahimoğlu, 2014, s. 5).

İlk bilgisayarın geliştirilmesi sonrası veriye yönelim gerçekleşmiştir. Aritmetik bazı hesaplamaları insan hızından daha hızlı yapabilmesi için tasarlanan bilgisayarın zamanla değişime ve özelliklerinin arttırılmasına ihtiyaç duyulmuştur. Bu ihtiyaçların başında bilgi saklama ve bu bilgileri saklayabilecek depolama ünitelerinin oluşturulması ile bu bilgilerin hızlı bir şekilde işlenmesi gelmiştir.

Veri madenciliğinin ilk izlenimi 1960'lı yıllarda istatistikçilerin "Veri Avlama" veya "Veri Tarama" kavramlarını kullanmalarıdır. İstatistik uzmanları bu kavramları analiz öncesinde verilere ait bazı özellikleri belirleme anlamında kullanmışlardır (Pektaş, 2013, s. 99). Bu yıllarda bilişim alanında veritabanı kullanılmaya başlanmıştır. Veritabanı hem depolama ünitesi hem de işlem yapma görevi ile tercih edilmiştir. Uzmanlar "basit öğrenme algoritmalarını" bilgisayar sistemlerinde kullanılabilecek bir hale getirmişlerdir. Yapay zekanın kurucuları olarak bilinen Marvin Minksy ve Seymour Papert, tek katmanlı sinir ağı olan perseptronların bazı basit kurallarının öğrenebileceğini göstermişlerdir (Dolgun Ö. , 2015). 1970'li yıllara gelindiğinde insanlar "A Relational Model of Data for Large Shared Data Banks" adlı makale ile ilişkisel veritabanı hakkında bilgi sahibi oldular (Codd, 1970). Bu makale, ilgili alandaki uzmanların makine öğrenimi konusunda ilerlemesini sağlamıştır. 1976 yılında "Varlık Bağıntı Modelleri (Entity Relationship Models)" önerilmiştir (Chen P. , 1976). 1976 yılında "R" dili geliştirilirken, 1980'li yıllarda "SQL" standart bir dil olarak görülmüştür (Dolgun Ö. , 2015).

Veri madenciliği mantığına ve amacına yönelik kaydedilen verilerden hareketle, geçmiş hakkında bilgi sağlayıp geleceğe ışık tutabilmesi açısından bir sistem olarak düşünülebilir. Bu sistemin her dönemde evrim geçirerek geliştiği görülmektedir. Veri madenciliğine ortam

hazırlayan ve veri madenciliği sonrasında gerçekleşen bazı gelişmeler Şekil 1'de gösterilmektedir.



Şekil 1: Veri Madenciliği

Kaynak: EXASTAX. Veri Madenciliğinin Tarihi. https://www.exastax.com.tr/veri-analitigi/veri-madenciliginin-tarihi/ (15 Temmuz 2018).

Veri miktarı ve veri depolama araçlarının hem sayısal anlamda hem de kapasitelerindeki artış sonucu uzmanlar ve kurumlar bu verilerden nasıl yararlanılabileceği konusunda arayışa girmişlerdir. 20 Ağustos 1989 tarihinde Piatetsky-Shapiro'nun katkılarıyla Veritabanlarında Bilgi Keşfi Çalıştayı düzenlenmiştir. Çalıştay, makine öğrenimi, uzman veritabanları, bilgi toplama, bulanık kümeler ve diğer alanlarda çalışmalar yapan birçok araştırmacı ve uzmanı buluşturmuştur (Piatetsky Shapiro, 1990, s. 68). Bu dönemde, klasik raporlama ve sorgulama araçları veri kümeleri karşısında verimsiz kalmıştır. Bu sebeple, bu organizasyonda ilk kez veritabanlarında bilgi keşfi kavramı ifade edilmiş ve diğer araştırmacılar konu üzerinde çalışmalara devam etmişlerdir. Piatetsky-Shapiro, "Knowledge Discovery in Real Databases: A Report on the IJCAI-89 Workshop" adlı yazısında, "Veritabanlarında Bilgi Keşfi" sürecini; uzman sistemler, makine öğrenimi, akıllı veri tabanları, bilgi edinimi, vaka temelli akıl yürütme

ve istatistik gibi çeşitli alanlardan faydalanarak "veriyi görselleştiren" farklı bir konu olarak yorumlamıştır (Piatetsky Shapiro, 1990, s. 68).

Veritabanlarında Bilgi Keşfi Çalıştayı araştırmacılar üzerinde olumlu bir etki yaratması sonucu belli aralıklarla düzenlenmeye başlanmış ve 1995 yılında Veritabanlarında Bilgi Keşfi adlı konferans gerçekleşmiştir. Veri madenciliği alanı için temel olarak görülen bu konferansın bildiri kitabında, "Dünyadaki enformasyon miktarının her 20 ayda bir ikiye katlandığı tahmin edildiği" ifade edilmiştir (Akpınar, 2000, s. 1). Bu durumu, günümüzde birçok kurumun günlük kayıtlarında olay veya kişilere ait birçok olguyu her an kayıt altına alması desteklemektedir. Milyonlarca veri yığını üzerinde çözümlemeler yapabilmek için yeni sistemlere ihtiyaç olacaktır. Eldeki mevcut veriye yön verebilmek için "veri ambarı" ile verileri işleyerek "etkin bilgiye" ulaştıracak "veri madenciliği" terimleri ortaya çıkmıştır (Almalı, Özdemir & Ulucan, Özkul, 2011, s. 108). KDD sürecinde bilginin ortaya çıkarılması veri madenciliği modelleri ile gerçekleşmektedir (Erdoğan Ş. Z., 2004, s. 4). Genel bir bakış açısı ile veri madenciliği, bilgisayar yapılarında saklı tutulan veriler için istatistik ve matematik bilimlerinden faydalanarak gözle görülemeyen ve anlaşılamayan bilgileri gün yüzüne çıkaran ve gelecek durumlar için tahminlere olanak sağlayan bir süreçtir (Almalı, Özdemir & Ulucan, Özkul, 2011, s. 108). Veritabanlarında bilgi keşfi süreci ve veri madenciliği taşıdıkları amaçlar bakımından benzerlikler taşımaktadırlar. İki süreç arası en belirgin fark, veri madenciliğinin ikincil belleklerde depolanan büyük miktardaki veri kümeleri üzerinde çalışmasıdır. Veri madenciliği modelleri genel olarak istatistik, matematik, makine öğrenimi, genetik algoritmalar ve nöron ağlar ile ilgilidir (Yarımağan, 2000, s. 295).

1990'lı yıllarda uluslararası kaynaklarda "Data Mining" olarak kullanılan ifade veritabanı konusu ile ilgilenen bilirkişiler tarafından Türkçe'ye "Veri Madenciliği" olarak çevrilmiştir (Pektaş, 2013, s. 99). Bu gelişmeler ardından uzmanlar tarafından "The Discovery, Analysis, and Representation of Data Dependencies in Databases" (Ziarko, 1991), "Fast Discovery of Association Rules, Advances in Knowledge Discovery and Data Mining" (Agrawal, Mannila, Srikant, Toivonen, & Verkamo, 1995), "A Statistical Perspective on KDD" (Elder & Pregibon, 1995), "From Data Mining to Knowledge Discovery in Databases" (Fayyad, Piatetsky-Shapiro, & Smyth, 1996), "Building Data Mining Applications for CRM" (Berson, Smith, & Thearling, 1999) çalışmaları gerçekleştirilmiş ve bunlara ek olarak daha bir çok köklü çalışma yapılmış ve veri madenciliği hızla geliştirilmeye çalışılmıştır. Çalışmalarda uzmanlar tarafından veri madenciliği hakkında farklı yorumlar dile getirilmiştir. "Bilgi keşfi, örtük, önceden bilinmeyen

ve potansiyel olarak yararlı bilginin veriden ayrılmasıdır." (Frawley, Matheus, & Shapiro, 1992, s. 58), "Veri madenciliği, verilerdeki geçerli, yeni, potansiyel olarak yararlı ve nihayetinde anlaşılabilir kalıpları belirleyen önemli bir süreçtir." (Fayyad, Piatetsky-Shapiro, & Smyth, 1996, s. 40-41), ve "Veri madenciliği, bir bilgisayar programı kullanarak geleceğe dair tahminler yapmaya yardımcı olacak ve çok sayıda veri arasındaki ilişkileri bulmamızı sağlayacak bilgileri elde etmektedir." (Doğan & Türkoğlu, 2008, s. 164), "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." (Öğüt, 2009, s. 6), "Veri madenciliği, büyük veri kümelerinden yararlı bilgiler çıkarmak ve yorumlanması kolay görselleştirmelerde görüntülemek için kullanılır." (Lu & Song, 2015, s. 130) ve "Veri madenciliği, yüksek boyutlu veritabanlarında "tanımlanamayan bilgileri" keşfetmenin potansiyel yollarını öneren teknolojidir." (Lakshmanan, Srinivasan, & Ponoraja, 2015, s. 43) şeklinde yorumlar dile getirilmiştir. Veri madenciliği hakkında yapılan bu yorumlardan hareketle, disiplin hakkında bazı ortak noktalara varılmaktadır. Bunlar, belirlenen amaç doğrultusunda ulaşılacak bilginin önceden bilinmiyor olması (Saklı kalmış tahmin edilememiş bir bilgi olması), bu çalışmaların bilgisayar ve bilgisayar sistemleri tarafından yapılıyor olması ve tekniklerin uygulandığı veri setinin yüksek boyutlu olması şeklindedir.

Dünya'da değişimin ve gelişimin hızla devam etmekte olduğu 1990'larda ilk yazılım gerçekleştirilmiş ve bilgisayar sistemlerinde Visual Basic, ODBC, Excel ve Access programları geliştirilmeye çalışılmıştır. Bilgisayarların bellek kapasitesinin ve hızının artması, birçok bilgisayar ve kullanıcı tarafından kolayca verilere ulaşılabilme ve veriyi uzun süre saklayabilme avantajları sayesinde kurumlar milyonlarca verinin saklanabildiği veri tabanlarına sahip oldular. Veritabanı ve veri ambarı depolama üniteleri başta olmak üzere hem kurumlar hem de insanlar tarafından kullanılan depolama üniteleri Delikli Kart (PunchCard), disket, Compact Disc (CD), Digital Versatile Disc (DVD) ve Universal Serial Bus (USB) gibi araçlar zamanla geliştirilmiştir. Teknolojinin gelişimi ve her gelişimin arkasından gelen ihtiyaçlar sonucu bugün depolama ünitelerindeki boyut, terabaytlara (TB) ulaşmıştır. Ancak ilerleyen zamanlarda bu depolama boyutunun da yetersiz kalacağını öngörmek yadsınamaz bir durumdur. Ayrıca, bu gelişmeler günümüzde bilginin ölümsüzleştirilmesine kalıcı imkân sağlamıştır.

Sürekli artan veri ile karşı karşıya kalan uzmanlar, istatistiksel bazı analizlerin ve veritabanı üzerinde kullanılan bazı sorgulama araçlarının yetersiz kaldığını fark etmişlerdir. Zaman, işlem

ve maliyet alanında tasarruf sağlayacak daha iyi çözümler arayışına girmişlerdir. Ayrıca, 1990'larda internetin gelişimi ile beraber hem veri boyutu artmaya hem de veri daha çok kullanılabilir hale gelmiştir. Bu durum, mevcutta kullanılan depolama sistemlerinin ve farklı yapılardaki veri kaynaklarının bütünleştirilmesi için "Veri Ambarı" ve "Veritabanı Yönetim Sistemleri" üzerinde kullanılan raporlama araçlarının geliştirilmesini zorunluluk haline getirmiştir. 2000'li yılların başlamasıyla, uzmanlar artık entegre şebekeler ile beraber büyük veri yığınları üzerinde tam anlamıyla profesyonelce çalışmalar gerçekleştirmeye başlamışlardır. Bu yıllarda veri madenciliği alanında "The x's of Statistical Learning; Data Mining, Inference and Prediction" (Friedman, Hastie, & Tibshirani, 2001) ve "Business Intelligence Roadmap: The Complete Project Lifecycle for Decision-Support Applications" (Atre & Moss, 2003) adlı önemli çalışmalar gerçekleştirilmiştir. Ardından, San Diego'da faaliyet gösteren kurumlardan biri olan HNC firması girişimciliği ön planda tutarak "Veritabanı Madenciliği" kavramının patentini alarak piyasaya sunmuştur (Özker, Dursun, 2016, s. 14).

Veri madenciliğine günümüzde Microsoft, Apple, Linkedln ve sosyal ağlara sahip kurumlar ciddi yatırımlar yaparken diğer kurumların sürdürülebilirlik anlayışları için veri madenciliği vazgeçilmez bir alan haline gelmiştir. Ayrıca, bu alanda hem Dünya'da hem de Türkiye'de makale ve tez çalışmalarının sayısı gittikçe artmaktadır. YÖK ulusal tez merkezi aracılığıyla Türkiye'de 2000 ve 2019 yılları arasında veri madenciliği alanına dair yapılan tez istatistikleri Tablo 1'de görülmektedir. Bu tabloya göre, Fen bilimlerinde eğitim gören öğrencilerin veri madenciliği alanında daha aktif olduğu anlaşılmaktadır.

Tablo 1: Türkiye'de Veri Madenciliği Alanında Yapılan Tez İstatistikleri, 2000-2019

Alan	Yüksek Lisans	Doktora
Fen	384	54
Sosyal	94	35
Tıp	8	2

Kaynak: Yazar tarafından derlenmiştir.

Özünde İstatistik ve Matematik bilimlerinin bir sentezini barındıran veri madenciliğinin tam anlamıyla özümsemek için bazı temel kavramların açıklanması gerekmektedir. Veri madenciliğinin anlaşılması için kavram olarak veri, veri türleri, veri ölçümü, enformasyon, bilgi, veri ve bilgi arasındaki farklar hakkında iyi bir literatür bilgisine sahip olunmalıdır.

1.1.Genel Tanım ve Kavramlar

Veri madenciliği alanında genel bir bakış açısı kazanmak ya da analizler yapabilmek için bazı temel kavramların iyi anlaşılması gerekmektedir. Bu kavramların başında araştırmanın temeli olan veri kavramı gelmektedir. Kelime olarak Latin dilinde gerçek ve reel anlamlarını ifade eden veri, kelime anlamını taşımamakta ve her zaman somut bir gerçekliğe karşılık gelmemektedir. Kavramsal anlamda veri çeşitli araçlarla kayıt altına alınan kişi, durum ve olaylara ait olgulardır. Bu olguların kayıt altına alınması ile oluşan ham veri, üzerinde silme ve düzeltme gibi herhangi bir işlem yapılmamış ve bir anlama kavuşmamış yani değeri olmayan kayıt yığınlarından ibarettir. Ham veri işlenerek yeni bir boyut kazanır ve bu düzenlenmiş yeni haliyle saklandığında farklı bir konu için veri halini korumaktadır (Öğüt, 2009, s. 1-2).

Analizlerde genel olarak araştırma ekibi tarafından toplanan verilerin belli amaç veya amaçlar doğrultusunda işlenmesi sonucunda faydalı bilgiye ulaşmak hedeflenmektedir. Veri setinin oluşturulması aşamasında araştırmacının karşısına birincil, ikincil ve üçüncül olmak üzere üç temel veri kaynağı çıkmaktadır. Birincil veri kaynakları, anket, gözlem, mülakat, odak grup görüşmesi ve deney gibi araştırma yöntemleri ile araştırma için bizzat toplanan veri kaynaklarıdır. İkincil veri kaynakları, daha önceki çalışmalar için toplanan verilerden oluşmaktadır. Araştırmacı veya araştırma ekibi tarafından araştırma problemine uygun olabilecek verileri sağlayabilecek, ulusal ve uluslararası birçok resmi, yarı resmi, internet ağları gibi birçok ikincil veri kaynağı mevcuttur. Verilerin toplanmasında, kuruluşun kendi veri kaynaklarının dışında, nüfus sayımı, merkez bankası kara listesi gibi veritabanlarından veya veri pazarlayan kuruluşların veritabanlarından da faydalanılabilir. Bu veri kaynakları başta zaman ve maliyet tasarrufu olmak üzere güvenilir veri, araştırmacıya ülke, bölge, şehir vs. temelli karşılaştırmalar yapma avantajlarını sağlamaktadır. Bu avantajlara rağmen verilerin araştırma konusuna içerik olarak uyumsuz olması, geçerliliği ve güvenilirliği sağlamaması, kurumların verileri pahalıya satması veya kullanıma hazır formatta olmaması gibi dezavantajları da mevcuttur. Üçüncül veri kaynakları, önceki araştırmalarda ilk iki veri kaynağı ile elde edilen verilerin analiz edilmesi ile oluşan çıktılardır. Bu toplanan veriler ölçülebilme ve sayısal olarak bir anlam ifade etme yönünden ikiye ayrılmaktadırlar. Bunlar; ölçülebilir ve sayı ile ifade edilebilir veriler nicel (metrik, kardinal) veriler ve ölçülemeyen ve sayı ile ifade edilemeyen veriler de nitel (kategorik) verilerdir.

İstatistik biliminin özünde, ölçmek, karşılaştırmak ve sonuç çıkarmak gibi temeller yatmaktadır. Bu temellerden hareketle, veriler üzerinde ayrım yapmamızı sağlayacak ve sağlıklı bir sonuca giden yolda verinin iyi değerlendirilebilmesi için ölçek tipleri mevcuttur. Nitel ve nicel veriler için ölçek tipleri birbirinden ayrılmaktadır. Nitel ölçekler; sınıflayıcı (nominal) ve sıralayıcı (ordinal), nicel ölçekler aralıklı (interval) ve oransal (Ratio) olmak üzere gruplara ayrılmaktadır. Nitel veriler sayısal bir değer ifade etmedikleri için bu veriler için kullanılan ölçek tiplerinde aritmetik işlemler yapılamamaktadır. Nitel ölçekli verilerin kullanıldığı çalışmalar, nicel ölçekli verilerin kullanıldığı çalışmalara göre daha büyük örneklem hacmi ile çalışılmalıdır. Yani, ölçek tipi zayıfladıkça, örneklem hacmi arttırılmalıdır. Bunun sebebi, bulguların güvenilirliğini garantileyebilmektir. Nicel verilerde kullanılan ölçekler nitel veriler için kullanılan ölçeklerden daha güçlüdür ve nitel ölçekler için kullanılan tüm istatistikler nicel ölçekler için kullanılabilmektedir. Ölçek tiplerine ait özellikler Tablo 2'de gösterilmektedir.

Tablo 2: Ölçek Tiplerine İlişkin Özellikler

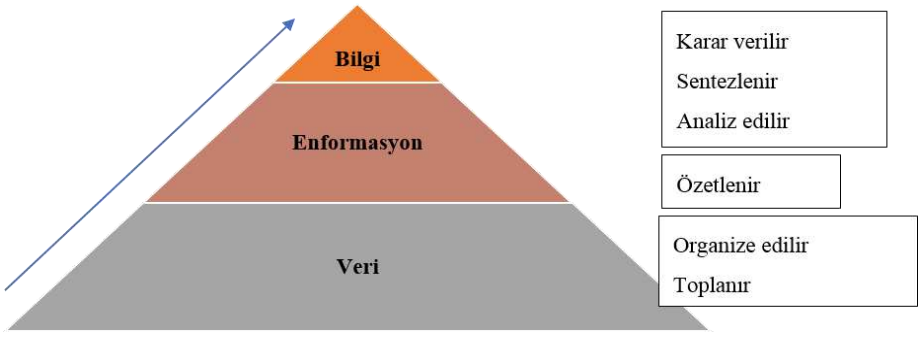
Ölçekler	**Sınıflayıcı**		**Sıralayıcı**	**Aralıklı**	**Oransal**
	Dikotom	**Multi-Nominal**			
Frekans Dağılımı	Var			Var (Verilerin kategorize edilme şartı ile)	
Betimleyici İstatistikler	Mod (Anlamlı sonuç vermez)	Mod	Mod, Medyan, Kantiller, Kantil Sapmaları, Spearman Katsayısı	Mod (Verilerin kategorize edilme şartı ile), Medyan, Aritmetik ve Kareli Ortalama, Standart Sapma, Varyans, Ortalama Sapma, Asimetri ve Basıklık Ölçüleri	
Grafik Türü	Pasta ve Sütun		Sütun	Frekans Poligonu ve Histogram	
İstatistiksel Testler		Ki-kare, Mc Nemar, Cochran	Sıra, İşaret, Mann-Whitney U, Kolmogrov-Smirnov, Wilcoxon, Friedman, Kruskal Wallis	F Testi, t Testi	Parametrik Testler, Parametrik Olmayan Testler

Kaynak: Yazar tarafından derlenmiştir.

Tablo 2'de, ölçek tiplerine özgü yapılabilecek hesaplamalar hakkında bilgi verilmektedir. Sınıflayıcı ölçek, verinin tipine göre dikotom ve multinominal olmak üzere ikiye ayrılmaktadır. İki ölçek arası fark, değişkenlerin dikotom ölçekte iki sonuçlu ve multinominal ölçekte ikiden fazla sonuç vermesidir. Bu ayrım verilerin inceleme yöntemlerinde daha iyi gözlenmektedir.

Multinominal ölçek için başvurulan mod değeri dikotom ölçek için bir anlam teşkil etmemekle beraber multinominal ölçek için önemli bir değere karşılık gelmektedir. Sıralayıcı ölçekte birimler arasında üstünlük özelliği var olması ile birimlerin ait olduğu sıra önem taşımaktadır. Aralıklı ölçek ve oransal ölçek birimler (sayılar) arası uzaklığın bir anlamı olup olmaması yani, sıfırın yokluk ifade edip etmemesine göre birbirinden ayrılmaktadır. Özellikleri bakımından en güçlü ölçek olan oransal ölçekte, sıfır yokluk ifade etmektedir. Bu ölçek türlerine ek olarak, sosyal araştırmalarda likert, boyutsal ayırma, thurstone, bogardus ve stapel gibi ölçekler kullanılmaktadır.

Mevcuttaki ham verilerin işlenmesi ile enformasyon ve bilginin elde edilmesi amaçlanır. Enformasyon ham verinin belli bir konuya göre işlenip düzenlendikten sonra çalışmada kullanılabilecek ve araştırmanın amacına ışık tutacak bilgiye dönüşür. Veri ve enformasyon kavramlarının odak noktasında yer alan bilgi kavramı, bir soruya karşılık vermek için verinin işlenmesi ile elde edilen enformasyondan çıkarılan olarak tanımlanabilir (Alpaydın, 2000, s. 1). Bilgi, sayesinde geçmişte yaşanan ve ileride tekrarlanabilir olgu ve durumlar hakkında tahmin yapılabilir, durumun iyiye gitmesini sağlamak için önlemler alınabilir. Ham veri gerekli düzenlemelerin sonrasında analizlerde kullanılarak bilgiye dönüşmektedir. Herhangi bir örneklem verisinden tesadüfi olarak seçilen "56" değerine ne bir yorum getirmek ne de bir anlam vermek imkansızdır. Çünkü bu birim veridir ve bilgi içermemektedir. Ancak, belli bir amaç doğrultusunda veritabanından çekilen "56" verisi işlenerek bir değer kazanır ve bilgi haline dönüşmektedir. Birbirini çevreleyen bu üç kavram çoğu zaman karıştırılsa da bu kavramlar farklı 3 yapıya sahip parçalardır. Bu fark, Şekil 2'de görülmektedir.



Şekil 2: Veri Madenciliğinde Temel Kavramlar

Kaynak: YAY, O. (2014). BI Dünyası. https://bidunyasi.wordpress.com/2014/06/04/is-zekasi-business-intelligence/ (17 Haziran 2018).

Bilgi kavramının diğer iki kavrama göre daha önemli olmasının sebebi, araştırmada gerçekleştirilecek eyleme daha yakın olmasıdır (Daştan, 2008, s. 6). Ayrıca, bilgi veriye göre daha az boyut kaplarken daha fazla önem ve değer taşımaktadır (Gürsakal, 2001, s. 48).

Bilgiye ulaşma yolunda araştırma problemine uygun veri toplama ve sonrasında Keşifsel Veri Analizi (Exploratory Data Analysis- KVA), araştırmanın başarılı olabilmesi için kurtarıcı rol üstlenmektedir. Veriler üzerinde yapılacak analizlerin doğru sonuçlar vermesi için birtakım istatiksel incelemeler öncesinde ya da hipotez sürecini başlatmadan veriler, olabildiğince ayrıntılı bir şekilde incelenmelidir. Çünkü, KVA ile veri kümesinden geçerliliği ve güvenilirliği optimum olan bir modele ulaşabilmek için verilerin ön incelemeye tabi tutulmasına ihtiyaç vardır. KVA, mevcut bir veri setinin yapısı ve içeriği hakkında hızlı ve pratik bir şekilde fikir sağlayacak yöntemleri barındırmaktadır. Yani, verinin ifade ettiklerini kavramak ile ilgili bir konudur (Karaibrahimoğlu, 2014, s. 9). KVA, verinin analiz öncesi düzenlenmesi ve anlaşılmasında rol alırken, veri madenciliği barındırdığı teknik ve algoritmalar ile kapsamlı bir veri analizi yapmaktadır. Böylece, veri madenciliği KVA'ya göre daha ileri çözümlemeler sunmaktadır.

Dünya'da ve Türkiye'de büyük veri potansiyeli ile büyük kazanımlar sağlanmıştır. Büyük veri ile 2008 ve 2012 ABD Başkanlık seçimlerinde Barack Hussein Obama ve 2016 yılı seçimlerinde ise Donald Trump başarılı sonuçlar elde etmiştir. Bu başarıların ardında kişilere ait verilerden hareketle, çoğunluğun karar ve isteklerini önceden analiz etme ve analiz sonuçlarına göre seçim kampanyalarını hazırlama şeklinde bir strateji yatmaktadır. Eski ABD Başkanı Obama'nın 2008 seçim kampanyalarında insanların önem verdiği konulara yönelik konuşmaları ile 2012 yılı için mikro hedeflemeye önderlik yapması ve şu an ABD Başkanı olan Trump'ın başkanlık seçimleri döneminde Facebook sosyal ağının 50 milyondan daha fazla kullanıcısına ait bilgilerden faydalanarak seçim stratejilerinde başarı göstermesi verinin önemini ortaya koymaktadır (Güçdemir, 2018). Türkiye'de büyük veri alanında hükümet tarafından yapılan girişimler Tablo 3'te ve özel sektör tarafından yapılan girişimler Tablo 4'te gösterilmektedir.

Tablo 3: Hükümet Tarafından Gerçekleştirilen Büyük Veri Projeleri

Devlet Projeleri	
Millî Eğitim Bakanlığı (MEB)	FATİH, MEBBİS, E-okul, E-yaygın ve ALO 147
Sağlık Bakanlığı	E-nabız, Ulusal Sağlık Bilgi Sistemi (UBYS), Doktor Randevu Sistemi (MHRS) ve Aşı Takip Sistemi
Sosyal Güvenlik Kurumu (SGK)	MEDULA, E-tavsiye ve ALO 170
Ulaştırma Bakanlığı	2013- 2013 Eylem Planı'nda yer alan Akıllı Kent Projesi
Kalkınma Bakanlığı ve Türkiye Bilimsel ve Teknolojik Araştırma Kurumu (TÜBİTAK)	Bulut Bilişim ve Büyük Veri Araştırma Laboratuvarı (B3LAB)

Kaynak: Türkiye Büyük Veri Piyasasına Genel Bakış: Büyük Veri. (2017). Exastax. URL: https://www.exastax.com.tr/buyuk-veri/turkiye-buyuk-veri-piyasasina-genel-bakis/ (18 Temmuz 2018).

Tablo 4: Özel Sektör Tarafından Gerçekleştirilen Büyük Veri Projeleri

Özel Sektör Projeleri	
2014	IBM tarafından "Türk Akıllı Şehirler Teknoloji Merkezi" projesi gerçekleştirildi.
2015	Türk Telekom (TT) Innova Karaman iş birliği ile Akıllı KenTT projesi gerçekleştirildi.
2015	SAAS'ın desteği ile Sabancı Üniversitesi ve MIT'nin ortak projesi olan Türkiye'nin ilk ve tek "Büyük Veri Davranışsal Analiz ve Görselleştirme Laboratuvarı" kuruldu.
2016	Turkcell tarafından en büyük veri merkezi kuruldu.

Kaynak: Türkiye Büyük Veri Piyasasına Genel Bakış: Büyük Veri. (2017). Exastax. URL: https://www.exastax.com.tr/buyuk-veri/turkiye-buyuk-veri-piyasasina-genel-bakis/ (18 Temmuz 2018).

Yeni dünya düzeninde tüm alanlarda sonu olmayan veri, belki de elde edilmesi için savaşılabilecek bir potansiyel ile yeni bir kaynak olarak karşımıza çıkmaktadır. Gerçekleştirilen çalışmalarda yer alacak araştırmacılar veri madenciliği alanındaki temel kavramları iyice özümsemeli ve araştırma süreci boyunca konu hakkında uzman ve istatistikçilerin görüşlerine başvurmalıdırlar. Ek olarak, veri madenciliğinin uygulama kısmında yer alan ve birbirini saran yapılar olan veri ambarı ve veritabanı sistemlerinin anlaşılması gerekmektedir. Karşılıklı akışın olduğu bu iki yapı veri madenciliğinde zaman tasarrufu sağlaması yanı sıra çalışmaların başarılarını arttırmaktadır. Ayrıca, dinamik veri kaynaklarına sahip olmak için bu iki yapı arasındaki benzerliklerin ve farklılıkların incelenmesi gerekmektedir.

1.1.1. Veritabanı Yönetim Sistemleri

Son yıllarda bilgisayar teknolojisinin gelişimi ile beraber önemli ilerlemeler kaydeden veritabanı, ilk ortaya çıktığı zamanlarda "kütük" kelimesinin devamı niteliğinde kullanılmıştır (Yarımağan, 2000, s. 1). Veritabanı birbiri ile bağlantılı olan kütüklerin birleştirilip, birbirinden ayrı alanlarda ortak paylaşımı sağlayan bir yapılanmadır (Erdoğan M. , 2006, s. 140). Veritabanı ve sistemlerini tanımlamak, oluşturmak, yararlanmak, değiştirmek ve bu sistemler ile ilgili tüm işletimsel ihtiyaçları karşılamak için başvurulan sisteme "Veritabanı Yönetim Sistemleri (Database Management Systems -DBMS)" denilmektedir (Yarımağan, 2000, s. 2). "Bağıntısal Veritabanı Yönetim Sistemleri (Relational Database Management Systems -RDBMS)", yüksek boyutta verileri depolama ve bu verilere aynı anda birden fazla kullanıcı tarafından hızlı erişebilme imkanlarını sağlamaktadır. ORACLE, POSTGRESQL, MYSQL, SYBASE, BERKELEYDB ve FIREBIRD bağıntısal veritabanı yönetim sistemleridir. Ayrıca, bu sistemlerde SQL, PL/SQL ve TCL dilleri kullanılmaktadır. Veritabanı yapısı gereği tekrarlı kayıtlar gibi tutarsız girişlere izin vermemesi ile beraber hem depolama alanı için avantaj sağlamakta hem de gereksiz yinelemelerin önüne geçmektedir. Belirli bir konuya dair hazırlanan veritabanındaki veriler arasında anlam bütünlüğü vardır (Yarımağan, 2000, s. 1-2). Veritabanının sağladığı avantajlar ve dezavantajlar Şekil 3'te gösterilmektedir.

Şekil 3: Veritabanının Avantajları ve Dezavantajları

Kaynak: Yazar tarafından derlenmiştir.

Şekil 3'te görüleceği üzere, veritabanındaki mevcut veriler kuruluşların birden fazla çalışmasında ortak kullanılabilen, birbiri ile bağlantılı sürekli verilerdir. Kalıcı olmayan veriler veritabanında yer almamaktadır. Veritabanında değişiklikler yapmak için erişim sadece veritabanı yönetim sistemleri denen yazılımlar ile sağlanabilmektedir. Veritabanı üzerinde günlük güncelleme faaliyetleri sağlanmasının yanısıra detaylı veya özel raporlar çekilebilecek bir yapıda dinamik veri sağlanmaktadır. Bir veritabanı yapısı bilgisayar sistemlerindeki kurulu diğer yapılarda olduğu gibi kullanıcılara arka planı göstermemektedir. Kullanıcılar, arka yüzü görmeden istediği veri ve sonucu görüntüleyebileceği sade bir ön yüz ve bu ön yüz aracılığıyla özet veriyle karşılaşmaktadır (Yarımağan, 2000, s. 1-2). Veritabanı sağladığı bu avantajları olması yanı sıra birtakım dezavantajları da bünyesinde barındırmaktadır. Veritabanı yapısından kaynaklanan bu dezavantajlar; boş değer, sürekli değişim, boyut ve sınırlı bilgi problemleridir. Veritabanında bulunan boş değer, kendisi ve diğer tüm niteliklere karşılık gelmeyen bir değerdir ve böyle bir değerin varoluşu araştırma sonuçlarında problem yaratmaktadır (Savaş, Topaloğlu, & Yılmaz, 2011, s. 6). Kurumların sahip olduğu veritabanları, sürekli değişimin olduğu dinamik bir yapıdadır. Veritabanındaki bilgiler, veri eklendikçe ya da silindikçe değişmektedir. (İnceoğlu & Vahaplar, 2003, s. 32). Bu özellik, dinamik veri avantajını sağlamakla beraber dezavantaja da neden olmaktadır. Dolayısıyla, veri madenciliği metotları için ciddi sorunlar yaratabilmektedir (Savaş, Topaloğlu, & Yılmaz, 2011, s. 6). Şöyle ki; önceden oluşturulan kuralların hala geçerli olup olmadığı ve verilerin zamana karşı hassasiyet taşıması uygulama sonuçlarını ciddi anlamda etkileyecektir. Depolama ünitelerindeki gelişime rağmen veritabanı boyutları yetersiz kalmakta ve boyut problemi oluşmaktadır. Bu durum, veritabanı algoritmasının küçük örneklemleri inceleyebilecek sınırlı kapasite ile geliştirilmiş olmasıdır (Savaş, Topaloğlu, & Yılmaz, 2011, s. 6). Son olarak, veritabanında karşılaşılan problem ise sınırlı bilgidir. Veritabanı yapısı gereği pratik ve kolay öğrenilebilecek görevleri yerine getirebilecek özellikleri sunmaktadır. Bu sebeple, bilgi edinme işini basitleştiren bazı nitelikleri bünyesinde barındırmamaktadır (Savaş, Topaloğlu, & Yılmaz, 2011, s. 6). Veritabanı aracılığıyla, bir okulda eğitim gören öğrencilerin sıralaması ve derslerdeki not ortalaması bilgilerine ulaşılabilir. Ancak, başarılı öğrencilerin ortak başarı sağladığı derslerdeki sınıfların ortalaması gibi bir sorguya veritabanı yetersiz kalacak ve bu durum sınırlı bilgi problemini yaratacaktır. Bu tür derin sorgulamalar için veri ambarı alternatif olmaktadır.

1.1.2. Veri Ambarı

Inmon, 1992 yılında ortaya koyduğu "Building the Data Warehouse" adlı çalışma ile veri ambarının ilk tanımlarını ortaya koymuştur (Inmon, 1992). Veri ambarı karar destek uygulamaları, öncelikle veri madenciliği ve çevrimiçi çözümsel işlemler (On-Line Analytical Processing/ OLAP) olmak üzere birden fazla teknik için veri kaynağıdır (Yarımağan, 2000, s. 293). Veri ambarı kurumların özellikle analiz, geliştirme ve raporlama faaliyetlerinin gerçekleştirilmesinde sıklıkla başvurulan ve kurum bilgilerinin tek bir çatı altında toplandığı büyük bir veritabanı topluluğudur. Genel anlamda veri ambarı, karar vericilere, seçim konusunda yardımcı olacak çeşitli veri kaynakların belli bir konu çerçevesinde zamana bağlı olmak üzere birleştirilmiş bir bütündür. Veri ambarı konu odaklı olma (Subject Oriented), bütünleşik olma (Integrated), özel bir zaman aralığına sahip olma (Time-variant) ve kalıcı olma (Non-Volatility) özelliklerine sahip bir sistemdir. Bir veri ambarının sakladığı veriler, araştırmacının yöneleceği konu ile ilgili bilgi sağlayacak gereksiz detaylardan arınmış bir yapıda olmalıdır. Veri ambarları tek bir konu etrafında oluştuğu gibi tek bir zaman dilimine ait şeklinde oluşturulmaktadır. Yani bir veri ambarında hem aylık hem de yıllık veriler saklı tutulamaz. Bu veri ambarının yapısını bozacağı gibi, zamana göre bir trend yakalamayı ve bu trendler üzerinde analiz yapmaya engel teşkil etmektedir. Veri ambarları, veritabanlarında olduğu gibi günlük güncelleme işlevlerine yani ekleme ve silme gibi işlevlerine kapalı sistemlerdir. Veri ambarı üzerinde sadece okunabilir özellikte verilere erişim ve yükleme işlevleri yapılabilmektedir.

Han ve ark. veri ambarının kurumlardaki dört büyük avantajını sıralamışlardır (Han & Kamber, 2001, s. 150).

- Veri ambarına sahip bir kurum, kendi performansını değerlendirebilir. Rakiplerine karşılık rekabet üstünlüğü sağlayabilir.
- Kurum faaliyetlerinin verimliliğini arttırır. Çünkü, kurum amaçlarını sağlayacak bilgileri çok hızlı bir şekilde sağlayabilir.
- Kurumdaki tüm birimlerde ve kurumun bulunduğu tüm alanlarda müşteri ilişkileri yönetimi verimli bir şekilde kolaylaştırılabilir.
- Kurum maliyetleri, zamana bağlı olan model, trend ve değişimleri takip edilerek azaltılabilir.

Veri ambarındaki veriler birden fazla heterojen kaynaktan elde edilerek bir araya getirilmektedir. Dolayısıyla, bu durum kaynakların hem veri türü hem de yapı gereği birbiri ile uyumsuz olma sorunlarını beraberinde getirmektedir. Bu duruma ek olarak, konum olarak kaynakların birbirinden uzak oluşu ek problemlere yol açabilmektedir. Veri ambarı bu tür çalışmaların ihtiyaçlarını karşılamak için farklı kaynaklarda, farklı tipte görünüm ve yapılarda ve farklı tarihlere ait verilerin lüzumsuz noktalarının süzülmesinden sonra gereken dönüşümlerin yapılması ile tek bir çatı altında bir araya getirilmesi ile elde edilmektedir (Yarımağan, 2000, s. 293). İşte, veri ambarındaki bu heterojen yapının varlığı karşımıza metaveriyi çıkarmaktadır. Veri hakkında veri olarak tanımlanan "Metaveri", veri ambarında veriler için tanımlamaların yapıldığı bölümden oluşmaktadır (Alpaydın, 2000, s. 1). Her bir veri öğesinin ne ifade ettiğini, hangi öğelerin hangileriyle ne şekilde ilişkili olduğunu göstermektedir. Ayrıca, metaveri kaynak verisi ile ulaşılabilecek veri gibi bilgileri de kapsamaktadır (Döşlü, 2008, s. 7). Metaveri dışında veri ambarında depolanan diğer veri türleri Şekil 4'te gösterilmektedir.

Şekil 4: Veri Ambarındaki Veri Türleri

Kaynak: Yazar tarafından derlenmiştir.

Veri ambarında, sisteme alınan ve işlenmemiş ve dolayısıyla yüksek boyuta sahip verilere ayrıntı veri denmektedir. Veri ambarında ayrıntı verinin oluşma tarihinden daha eski tarihte kalan ayrıntı veriler, eski ayrıntı veri olarak tanımlanır ve bunlar ayrıntı veriye göre özet olarak saklanmaktadır. Ayrıntı veri için bir diğer ayrım ise işlenmemiş veriler üzerinde hangi düzeyde ve hangi husus dikkate alınarak indirgeme yapılmasıdır. Bu konu veri ambarının yapımı ve tasarımı esnasında belirlenmektedir. Detaylı bir şekilde özetlenen ayrıntı veri, daha fazla kriter ortaya koymakta ise bu verilere yüksek düzeyde özetlenmiş veri denir. Tam tersi durumda, yani ayrıntı veride daha dar bir özetlemeye gidildiğinde oluşan verilere düşük düzeyde özetlenmiş veri denir. Son olarak, veri ambarında bulunan "Veri Pazarı (Datamart)", sadece bir birim tarafından belirlenen konu ve ihtiyaca göre hazırlanan ve veri ambarının sadece bir kısmını kapsayan bölümlerdir.

1.1.3. Veri Ambarı ve Veritabanı Sistemleri Arasındaki Farklılıklar

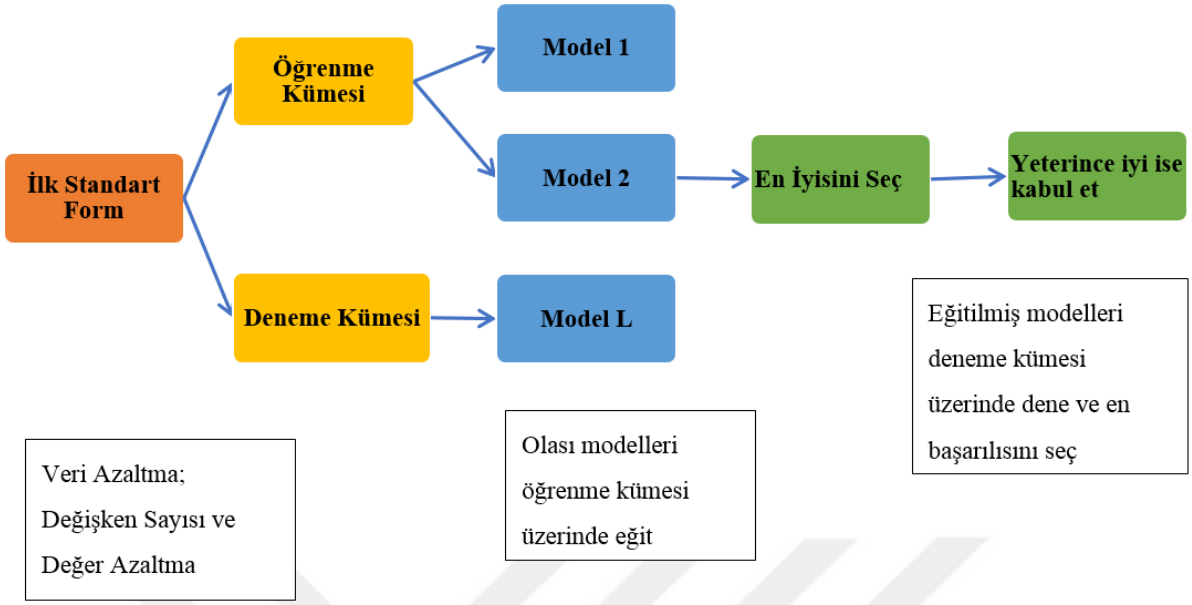
Bilgisayar sistemleri üzerine kurulu olan veritabanı ve veri ambarı birbiri ile ilişkili iki farklı yapılanmadır. Veritabanı yönetim sistemleri konusunda bazı detaylı ve özel raporlara ulaşabileceğinden bahsedilmişti. Ancak, veritabanı verinin boyutu ve yapılan sorgulamanın kapsamı genişledikçe etkinliğini kaybetmektedir. Böyle bir noktada, veri madenciliği metodolojisi ve uygulamalarında esas olan kural elde etme görevini veritabanları dahil olmak üzere birden fazla heterojen veri kaynağını bünyesinde bütünleştiren veri ambarları karşılamaktadır. Böylece, veri ambarı analiz ve sorgulamalarda birçok avantaj sağlayan veritabanına hem alternatif olmakta hem de veritabanını bünyesinde barındırmaktadır. Veri ambarı, özel bir tarih aralığına sahip ve söz konusu çalışmanın konusuna göre planlanan, bir araya getirilen ve sabitleştirilen veritabanlarından oluşmaktadır (Silahtaroğlu, 2013, s. 17). Dolayısıyla, birbirinden farklı bu iki yapının birtakım yöntemler sonucu birbiri üzerine eklenen bilgi depolama sistemleri oldukları ve veri madenciliğinde uygulamalarında aynı amaç için kullanılabileceği anlaşılmaktadır.

Veri ambarı ve veritabanı, veri kaynakları, veri saklama biçimi, veri güncelleme aksiyonları ve sakladıkları veri türleri açısından farklılıklar barındırmaktadır. Bir veritabanı amacı gereği, elde edilen tüm verileri saklamasına karşılık bir veri ambarı farklı kaynaklardaki verileri işleyerek ve özet hale getirerek saklı tutmaktadır (Şeker, 2015, s. 6). Veritabanı genel olarak ayrıntı düzeyi yüksek, zaman özelliği bulunmayan, oynak ve güncel verilerden oluşmaktadır. Veri

ambarı ise genel olarak karar destek yapıları gibi yönetsel görevler, veri madenciliği ve OLAP'ta kullanılan, özetlenmiş, zaman özelliği bulunan ve oynak olmayan, verilerden oluşmaktadır. Veritabanlarında güncelleme (ekleme ve silme gibi) işlemleri sıklıkla yapılırken, veri ambarında bu gibi işlemler pek yapılmamaktadır (Yarımağan, 2000, s. 291). Bazı algoritmalar, veritabanındaki verilerin güncellemeleri hakkında bilgi sağlamaktadır. Veri ambarında ise trend ve değişen değerler baz alınarak güncelleme aksiyonu alınmaktadır. Yani, bu iki sistem arasında tek taraflı bir veri aktarma olayı söz konusudur (Şeker, 2015, s. 7).

1.2.Veri Madenciliği Metodolojisi

Metodoloji kavramı, belirli bir alana özgü uygulanmakta olan metotların felsefesini ve mantığını ortaya koyan bir kavramdır ve bazı kaynaklarda yöntem bilimi olarak da adlandırılmaktadır. Veri madenciliği alanındaki kurulu olan metodoloji genel olarak Şekil 5'te gösterilmektedir. Buna göre, veritabanında saklanan veriler öğrenme ve deneme kümesi olmak üzere iki gruba ayrılmaktadır. Mevcuttaki verileri uygulamaya dökmekte birden fazla yöntem karşımıza çıkmaktadır ve en optimal sonucu hangisinin vereceğini bilmek ve bu karara varmak gerçekten zordur. Öğrenme kümesinde L tane farklı yöntem ile L tane model kurulur. Bu modeller deneme kümesi üzerinde test edilerek, en güçlü tahmine sahip olan model başarılı olarak seçilir. Ancak sistem burada sonlanmamaktadır. Başarılı bulunan model eğer gerçekten optimal sonuçları verebilecekse kabul edilir, değilse ret edilir ve çalışma baştan tekrar edilir. Bu noktaya kadar başarısız olan modeller üzerinde incelemeler yapılır ve iyileştirme çalışmaları gerçekleştirilir. Bunu gerçekleştirmek adına veri setine yeni değişkenler atama, amaç değişikliği ve mevcut veriyi dönüştürme gibi işlemler gerçekleştirilebilir (Alpaydın, 2000, s. 6).



Şekil 5: Veri Madenciliğinde Kullanılan Metodoloji

Kaynak: ALPAYDIN, E. (2000). Zeki Veri Madenciliği: Ham Veriden Altın Bilgiye Ulaşma Yöntemleri. Bilişim 2000 Veri Madenciliği Eğitim Semineri. İstanbul. s.6., 1-10.

1.3.Veri Madenciliği Amacı

Veri madenciliğinin seçilen yöntem ve üzerine çalışılan konu dışında genel amacı, veri ambarında saklanan büyük miktardaki veriden ihtiyacımız olan güncel bilgileri üretebilmek, üretilen bilgiler ile ilişkileri tanımlamak, veriye ait özellikler sayesinde konu hakkındaki eğilimleri yakalamak ve tahminler yapmaktır. Veri madenciliği; araştırmacıya veya araştırma ekibine faaliyet sürecinde meydana gelen veriler arasındaki desenleri ve bağlantıları keşfetme konusunda yardımcı olmaktadır (Baykal A. , 2006, s. 96). Veri madenciliği uygulamaları genel olarak tahmin, betimleme, sınıflandırma ve eniyileme amaçları üzerinde yoğunlaşmaktadır. Verilerden hareketle, gelecekte yaşanacak durumlar için tahminlerde bulunarak bilgiye ulaşılmaktadır. Bu bilgi ilerleyen zamanlarda kullanılabilir. Hangi ürün, hangi dönemde, hangi koşulda, hangi miktarda satılacağına ilişkin tahminlerde bulunmak (Yarımağan, 2000, s. 295) tahmin amacına uygun örnekler olarak sayılabilir. Betimleme, geçmiş verilerden hareketle, yeni gerçekleşen bir olgunun tanımlanan eski olgular ile eşleştirilmesini sağlamaktadır. Müşterilerinin bağımlılığını arttırmak için kurumun onlara özel fırsatlar sunması ya da mevcutta olmayan müşteriler için müşteri profilleri oluşturarak yeni müşteri kitlelerini

bünyesine katması gibi girişimler betimleme amacını taşımaktadır. Sınıflandırma, betimleme amacının devamı niteliğindedir. Veri setinden çıkarılan birden fazla parametrenin bileşimi ile hizmetler, çalışanlar veya müşteriler bazı özelliklere göre gruplandırılabilir (Yarımağan, 2000, s. 291). Bu grupların özellikleri incelenerek ve elde edilen bilgi ile gruplar hakkında ileriye dönük tahmin çalışmaları yapılabilir. Eniyileme, zaman, mekân, maddi ve manevi kaynaklarda harcamaları azaltma ile üretim, satış ve kar gibi konularda artışı sağlama veri madenciliği amaçlarını içerir (Yarımağan, 2000, s. 295). Personel ve verimlilik optimizasyon çalışmaları da eniyileme amacı güden veri madenciliği çalışmaları olabilir.

1.4.Veri Madenciliğini Etkileyen Etmenler

Tarihsel süreçte, veri madenciliği tekniklerinde karşılaşılan eksiklikler veya hataların giderilmesi için çalışmalar yapılarak yeni çözümler keşfedilmiştir. Ancak, yeni çözümlerin de ilerleyen süreçlerde araştırma problemlerine cevap verememesi ile tekrar araştırmalar yapılarak veri madenciliği teknikleri geliştirilmiştir. Genel olarak veri madenciliği çalışmalarını veri, iş gücü, donanım, bilgisayar ağları, bilimsel hesaplamalar ve ticari eğilimler olmak üzere altı etmen etkilemiştir.

Veri, veri madenciliği alanında hem araştırma kaynağı hem de araştırma konusu olma sebebiyle birinci öneme sahip etmendir. Teknolojinin gelişimi ve nüfus ile beraber tüketimin artması veri madenciliği alanında yapılan çalışmaların artışını ve diğer beş etmenin ilerlemesini etkilemiştir. Veri madenciliği teknikleri ve algoritmalarındaki hesaplamalar bilgisayar ve bilgisayar sistemlerinin varlığını zorunlu kılmıştır. İlk bilgisayar, basit aritmetik çalışmaları gerçekleştirmesi üzerine tasarlanmıştı. Ancak günümüz bilgisayar ve bilgisayar sistemleri uzmanlara uygulama alanında ciddi kolaylıklar sağlamaktadır. Büyük veriyi depolayabilmek ve üzerinde hızlı bir şekilde istatistiksel çözümlemeler yapabilmek için güçlü depolama araçlarına ve işlem gücüne gereksinim duyulmuştur. Bilgisayar sistemleri donanımının güçlü hale gelmesi ile önceden üzerinde işlem yapılamayan yüksek boyutlu veri kaynaklarında hızlı işlemler yapılabilir hale gelmiştir. Entegre ağlar, bilgisayar sistemlerinin birbiri ile bağlantı kurmasını ve veri madenciliği çalışmalarına uyumlu hale getirilmesini sağlamıştır. Bilgisayar ve bilgisayar sistemlerine günümüz şartlarının tam aksine geçmişte maddi anlamda erişimin kolay olmaması veri madenciliği alanındaki çalışmalara kısıtlama getirmiştir. Ayrıca bilgisayar ve bilgisayar sistemlerinin varlığı ile beraber bu alanda yetiştirilen veri madenciliği uzmanları

ve veri toplama alanında nitelikli işgücü ihtiyacı doğmuştur. Veri madenciliğinin kurumlar için önemi, "1.6. Veri Madenciliği Kullanım Alanları ve Yapılan Çalışmalar" adlı başlıkta bahsedilecektir. İşletmeler hem çalışan sayısı hem de diğer kaynaklar açısından en az girdi ile en çok çıktıyı elde edebilmek için veri madenciliğinden faydalanması kaçınılmazdır. Ayrıca, kurumlar bulunduğu sektörde rakiplerine karşı rekabet üstünlüğü sağlama da veri madenciliğinden çok fazla yararlanmaktadır.

1.5.Veri Madenciliği ve İstatistiğin Karşılaştırılması

Özellikle son yıllarda teknolojinin gelişimi ile beraber veri hem miktar hem de çeşitlilik bakımından hızlı artmaktadır. Veriye olan ilginin de artması sonucu analizlerde istatistiksel çözümler yetersiz kalmıştır. Bilim insanları ve kuruluşlar bünyelerinde topladığı veriyi istatistiksel yöntemlerin de ilerisinde çözümler bulma arayışına girmişlerdir. Bu arayışta en etken güç bilgisayar ve bilgisayar sistemleri olurken, istatistiksel temellere dayanan veri madenciliği ise bu arayışın asıl çözümü olarak ortaya çıkmıştır. Bilgisayarlar aracılığıyla veriden bilgi sağlayacak örüntüler elde edilmiş ve bu sürece veri madenciliği denilmiştir (Adirans & Zantinge, 1997, s. 5). Verilerin birtakım yöntemler ile toplanması ve incelenmesi, genel durum hakkında faydalı bilgiler elde edilmesi ve bunların sunulması için başvurulan tekniklerin bütünü istatistik bilimini oluşturmaktadır. Veri madenciliğinden daha uzun bir geçmişe sahip olan istatistik, günümüzde kullanılmakta ve çalışmalarda çok yararlı bilgileri ortaya çıkarmaktadır. Ayrıca, veri madenciliği çalışmalarında da çoğunlukla tercih edilmektedir. Karar ağaçları modelinde bahsedildiği üzere CHAID algoritmasında hedef (bağlı) değişkenin nitel veya sürekli olması durumuna göre Ki-Kare veya F testine başvurulması bu duruma örnek olarak verilebilir (Oğuzlar, 2004, s. 81).

İstatistik ve veri madenciliği çalışmalarının buluştuğu nokta; veriden faydalı bilgiler çıkarma uğraşıdır. Bu iki disiplin bilinmeyen durum ve olguları netleştirmek ve ileriye yönelik tahminler yapmak için geliştirilmiştir (Tüzüntürk, 2010, s. 72). İstatistiğe genel bir bakış açısı ile yaklaşıldığında belirli durumları tanımlaması ve değerlendirmesi veri madenciliği alanında gruplandırma, bağıntı oluşturma, ileriye yönelik tahminler geliştirme ile durum ve olguları karşılaştırma işlevleri için başvurulmaktadır. Bilgisayarın icat edilmesinden sonra veritabanı sistemleri ve veri ambarının ortaya çıkışı ile yeni öğrenme yapıları ve algoritmaların geliştirilmesi sonucunda veri madenciliği birden fazla yöntemin bütünleştiği bir yapı halini

almıştır. Çalışmalarda veri madenciliği ve istatistiksel yöntemlerin kullanım tercihi birden fazla unsura göre değişebilir. Bu tercih, çalışmada yer alan uzmanın tercihi ve çalışma yapılan alandan öte veri setinin boyutu, yapısı ve yapılmak istenen çözümlemenin niteliğine göre değişmektedir.

Veri madenciliğinin içerdiği alan ve disiplinlere dair örnek Şekil 6'da gösterilmektedir. Veri madenciliği başta istatistik olmak üzere veri görselleştirme, yapay zekâ ve veritabanı sistemleri gibi farklı konuların bir arada olduğu bir alandır. Veri madenciliğinin amacı pratik bir şekilde mantıksal kurallar ya da görsel çıktılar üretebilecek kategorik modeller üretmektir (Baykasoğlu, 2005, s. 1).

Şekil 6: Veri Madenciliği İlgi Alanları

Kaynak: BAYKASOĞLU, A. (2005). Veri Madenciliği Ve Çimento Sektöründe Bir Uygulama. Akademik Bilişim 2005. İnternet Teknolojileri Derneği. Gaziantep. s.2., 1-14.

Veri madenciliği ile elde edilen çıktılarda, cazip bir grafik ara yüzü, sorgulama dili, veri çözümleme prosedürleri dili, esnek uygun girdi, tıklama ikonları ve menüler, girdi için iletişim kutuları, çözümlemeleri betimleyen diyagramlar ve çıktıların hızlı ve çok yönlü grafikler özellikleri bulunmaktadır. Ancak, veri madenciliği paketlerinin çoğunda, hipotez testleri, deneysel tasarım, varyans analizi, çok değişkenli varyans analizi, regresyon analizi, diskriminant analizi, kanonik korelasyon, faktör analizi gibi istatistiksel yöntemler mevcut değildir. Bu gibi istatistiksel analizlerin veri madenciliği ürünlerinde olmaması, istatistik ve veri madenciliği arasında bir ilişki olmadığı düşüncesine sebep olmaktadır. Bu durum, veri madenciliği alanında çoğunlukla bilgisayar ve bilgisayar sistemleri alanındaki kişilerin gelişim

göstermesinden kaynaklanmaktadır. Ancak, veri madenciliğinin merkezinde ciddi bir matematik ve istatistik temeli yer almaktadır (Karaibrahimoğlu, 2014, s. 20-21).

Veri madenciliği ve istatistik arasında bazı farklılıklar bulunmaktadır. Bir karara ulaşmaya çalışırken anakitle hakkında yapılan varsayımlara hipotez (hypothesis) denmektedir (Çilingirtürk, 2011, s. 107). Ortaya konulan hipotez çerçevesinde veriler incelenmektedir. Hipoteze dayalı uygulanan yöntemler ender olaylar fikrine dayanmaktadır (Çilingirtürk, 2011, s. 107). İstatistiksel yöntemlerde çözüme giden yolda hipotezler belirleyici rol olmakta iken veri madenciliğinde farklı yol alınmaktadır. İstatistiksel yöntemlerde hipotezler incelenirken, anakitlenin tamamının araştırılması yerine anakitleyi temsil edebilecek daha az sayıda gözlem ile araştırma yapılmaktadır. Bu şekilde hipotezin kabul veya ret edilmesine karar verilmektedir. Yani, araştırma sonuçları tüm anakitleyi temsil etmektedir. Veri madenciliğinde ise daha yüksek boyutlu veriler ile çalışılmasına rağmen örneklemeye gidilmemektedir. Bu durumda istatistikte çok önemli olan hata payı veri madenciliğinde önemini kaybetmektedir. Ayrıca, değişkenler arasındaki ilişkiler incelendiğinde, veri madenciliği istatistiğe göre daha az varsayıma bağlı kalmaktadır (Tüzüntürk, 2010, s. 72).

Veri madenciliği tümden gelim ile ilgilenirken, istatistik tümevarım ile ilgilenir (Tüzüntürk, 2010, s. 72). İstatiksel yöntemlerin kullanıldığı çalışmalarda anakitleden rasgele çekilen örneklem ile anakitle hakkında kestirim yapılmaktadır. Bu durum istatistiksel çözümlemelerin detay bilgiden çok anakitleyi en iyi temsil eden belli bir kesim ile genel çıkarımları sağladığını göstermektedir. Oysa, veri madenciliği çalışmaları veri setinde gizli kalmış anlamlı örüntüleri bulmaya odaklanmaktadır.

Araştırmalarda hem girdi hem de çıktı olarak karşımıza çıkan veri istatistik ve veri madenciliği arasındaki en belirgin farkın sebebi olmaktadır. Daha önceki konularda da ifade edildiği gibi bilgisayarların hesaplamalarda yer alması ve verinin hızlı artışı ve ile beraber çözümlemelerde yetersiz kalan istatistiksel yöntemlere veri madenciliği çözümlemeleri alternatif olmuştur. “Bir istatistik uzmanı için büyük veri seti birkaç yüz veya bin veri ile sınırlı iken veri madenciliği ile ilgilenen bir uzman için milyarlık veri normal karşılanacak veri seti boyutudur (Oğuzlar, 2003, s. 69)” yorumdan hareketle, istatistik ve veri madenciliği arasındaki bir diğer fark veri setinin büyüklüğüdür.

Teknolojinin sunduğu imkanlar ile verileri konuşturma sanatı olarak görülen veri madenciliği, her ne kadar istatistik bilimine dayanmakta olsa da farklı iki süreçten oluşmaktadır. Daha önceki konularda da değinildiği üzere sağlam ve kaliteli çıktılar elde edebilme imkânı temiz girdilerde

saklıdır. Farklı ve büyük veri kaynaklarının söz konusu olması veri madenciliği çalışmalarında veri setinin ön işlemeye tabi tutulmasına sebep olmaktadır. Buna karşılık istatistiksel çözümlemelerde veriler ön işleme aşamasına tabi tutulmadan çalışma yapılmaktadır. Bu durumda, istatistik ve veri madenciliği arasında işleyiş farkı oluşmaktadır.

İstatistiksel yöntemler, belirli bir amaca dayalı olarak araştırmaya özgü birincil, ikincil ve üçüncül veri kaynakları üzerinde uygulanmaktadır. Veri madenciliği çalışmaları ise, farklı veritabanları üzerinden faydalı bilgiler çıkarmak için yapılmaktadır. istatistik ve veri madenciliği arasındaki bir diğer fark veri toplama yöntemidir.

1.6.Veri Madenciliği Kullanım Alanları ve Yapılan Çalışmalar

Günümüzde veri madenciliği, teknolojik gelişmeler ve bu alanda yetişen nitelikli iş gücünün artması ile en hızlı gelişim gösterdiği dönemlerini yaşamaktadır. Veri madenciliği hemen hemen tüm alanlarda başvurulan ve sonuçları gözden çıkarılamayacak bir disiplin olarak temellerini sağlam atmış bulunmaktadır.

Kurumlar maliyet optimizasyonu, personel verimliliği ve üretime karşılık tüketici gruplarını sağlamak; yani, özünde tüketicinin ne istediği bilgisine dayanarak üretmek ve bu şekilde rakiplerinin önüne geçmek için veri madenciliğine başvurmaktadırlar. Kurumların sahip oldukları veritabanlarında kişilerin demografik bilgileri ile beraber her bir bireyin ticari kayıtları bulunmaktadır. Kurumlar bu kayıtlardan büyük yarar sağlamaktadır (Kotler, 2007, s. 166). Tüketim çağında veri madenciliği ve uygulamaları ile en çok müşteri odaklı sistemlere sahip kuruluşlar ilgilenmektedir. Veri madenciliği genellikle Müşteri ilişkileri Yönetimi (Customer Relationship Management- CRM) departmanlarında tercih edilmektedir. Müşterilerin firmaya olan bağımlığını sürekli hale getirebilmek için müşteri segmentasyon uygulaması, müşteri yaşam boyu değer analizi, sadakat çalışmaları ve müşteri değerlendirme çalışmaları yapılmaktadır. Kurumlar, müşteri anlayışını tespit etmek adına öngörü analizleri gerçekleştirerek kurum için en iyi aday müşterilere odaklanır ve başarılarını arttırabilirler (Kotler, 2007, s. 162). Firma veritabanında saklı tuttuğu geçmiş verilerden hareketle, elde edeceği müşteri profilleri ile satış arttırıcı kampanyalar geliştirebilir. Ayrıca satış tahmini çalışmaları veya mail ve telefon ile kampanyalara dönüş sağlamayan müşterileri filtreleyip maliyet optimizasyonunu ve dolayısıyla karını arttırmayı başarabilir. Sonuç olarak kurumlar bu çıktıları elde edebilmek için veri madenciliğinden faydalanmak zorundadırlar. Çünkü bir

işletme müşterilerini ne kadar çok tanır ve analiz edebilirse, rakiplerine karşılık o kadar üstünlük sağlayabilir.

Finans sektörünün banka ve sigorta dallarında dolandırma vakalarının tespiti, birimlerin daha verimli çalışmasını sağlayacak optimizasyon çalışmaları, sigorta ve kredi taleplerinin tahmini, geçmiş müşteri bilgilerinden hareketle kurum için risk oranı yüksek olan müşterilerin tespiti, ekonomik birimler arasındaki gizli kalmış bağıntıların belirlenmesi ve stoklar için yapılan optimizasyon çalışmaları için veri madenciliğine başvurulmaktadır. İnternet ağının gelişimi ardından gelen ileri düzey web sistemleri ve 2000'lerde ön plana çıkana online satış tekniği sonucu oluşan elektronik ticarette (E- ticaret) veri madenciliğinin en çok kullanıldığı alanlardan biridir. E- ticarette sosyal ağlar aracılığıyla müşteri profilleri ortaya çıkartılabilir. Değerlendirmelere göre sitelerde her bir kullanıcı için akıllı reklam düzenlemeleri veri madenciliği çalışmaları yapılmaktadır. Web madenciliği alanında ise trafik optimizasyonu çalışmaları (Gökmen, 2014, s. 39), belgeler arası benzerlik: haber kümeleri, e-posta (Eker, 2004) ve web trafiği analizleri ile müşterinin internet sitesi üzerinde gerçekleştirdiği hareketlerin analizi, saldırıların amacına ulaşmadan tespiti ve önlenmesi çalışmaları (Gökmen, 2014, s. 39) yapılmaktadır.

Ürün ve hizmet sektöründe olduğu kadar veri madenciliği tıp, genetik ve biyoloji alanlarında da tercih edilmektedir. Bu alanlarda, hastalar için erken uyarı sistemleri, laboratuvar testlerinde hata tespiti, yerleşim yerlerine göre hastalık haritalarının çıkartılması (Eker, 2004), DNA sıraları içerisinde genlerin tespiti, gen haritalarının analizi, genetik hastalıkların tespiti, kanserli hücrelerin tespiti (Erdoğan Ş. Z., 2004, s. 6), hastalıklarda kesin teşhis yöntemleri, ameliyatlarda cerrahi risk oranı, hastane yönetim sistemleri, sağlıkta maliyet düşürme, radyolojik görüntüleme (Silahtaroğlu, 2008, s. 83-98) gibi çalışmalar veri madenciliğine örnek olarak verilebilir.

Ulaşım ve yerleşim birimlerini kapsayan yüzey analizi ve coğrafi bilgi sistemleri alanlarında uygulanan veri madenciliği çalışmalarına bölgelerin coğrafi özelliklerine göre sınıflandırılması, kentlerde yerleşim yerlerinin belirlenmesi, kentlerde suç oranının tespiti, otomatik para makinelerinin yerlerinin tespiti, otobüs duraklarının yerlerinin belirlenmesi (Erdoğan Ş. Z., 2004, s. 7) örnek olarak verilebilir.

Endüstriyel sektörde veri madenciliği, iyileştirme ve geliştirme çalışmaları, kalite kontrol çalışmaları ve üretim çeşitliliğine bağlı politikaların belirlenmesinde kullanılmaktadır (Eker, 2004).

Veri madenciliği, bu çalışmalar dışında veritabanı analizi, karar verme desteği, kaynakların doğru kullanımı, telekomünikasyon hatlarındaki parazitlenmeyi tespit etme ve gürültü giderme (Eker, 2004) gibi çalışmalarda da kullanılmaktadır. Bu alanlarda yapılan çalışmaların bazıları Tablo 5'te gösterilmektedir.

Tablo 5: Veri madenciliği Alanında Yapılan Bazı Çalışmalar

Çalışma	Açıklama
Apriori Algoritması ile Öğrenci Başarı Analizi	Çalışmada veri seti Fırat Üniversitesi Teknik Eğitim Fakültesi Bilgisayar Eğitimi bölümü öğrencilerinin notlarından oluşmaktadır. Veri seti üzerinde Apriori algoritmasından faydalanarak, Birliktelik Kuralları analizi yapılmıştır. Öğrencilerin dersleri ve notlarından anlamlı örüntüler bulmak için SQL ve MATLAB programları kullanılmıştır (Karabatak & İnce, 2004).
Gırtlak Kanseri Ameliyat Verilerinin K-means Yöntemiyle Analizi	Çalışmada veri seti Kocaeli Üniversitesi Tıp Fakültesi Hastanesi, Kulak, Burun ve Boğaz Bölümündeki gırtlak kanseri tespiti ile ameliyat olan hastalara ait ameliyat verilerinden oluşmaktadır. k-means algoritması ile bir yazılım tasarlamışlardır. Veri seti üzerinde Kümeleme modeli uygulanmıştır. Araştırma sonuçlarından hareketle, aynı tip ve seviyedeki hastalar için uygun tedavi seçenekleri belirlenebilmekte ameliyat öncesi ameliyat başarısı tahmin edilebilmektedir (Dinçer & Duru , 2006).
Veri Madenciliği ile Deprem Verilerinin Analizi	Çalışmada veri seti Yeryüzü ve Uzay Bilimleri Araştırma Merkezi (YUBAM)'nden temin edilen Kilikya bölgesindeki 4 ve 4 şiddetin üstündeki deprem verilerinden oluşmaktadır. Veri seti üzerinde Lineer Regresyon analizi ile bir bölgenin deprem afetine karşılık risk analizini ortaya koymaya çalışılmıştır. MS Access veritabanı ve Microsoft Visual Studio programları kullanılmış ve araştırmaya özgü bir yazılım geliştirilmiştir. Ayrıca çalışma sonunda elde edilen model ve yazılımın dünyadaki her bir nokta için uyarlanabileceği savunulmuştur (Canbay & Duru , 2007).
Ege Bölgesi'ndeki bir Araştırma ve Uygulama Hastanesinin Acil Hasta Verilerinin Veri Madenciliği ile Analiz Edilmesi	Çalışmada veri seti Ege Bölgesi'ndeki bir araştırma ve uygulama hastanesinin acil servisinde muayene edilen 214 bin hastaya ait verilerden oluşmaktadır. Veri seti üzerinde Apriori algoritmasından faydalanarak, Birliktelik Kuralları analizi yapılmıştır. Çalışma sonuçlarından hareketle, hasta profilleri çıkarılmış ve acil servislerin bölgelere göre farklı düzenlenmesini sağlayacak bilgilere ulaşılmıştır (Erdem & Özdağoğlu, 2008).
Türkiye'de Bir Havayolu İşletmesine Ait Parça Söküm Raporlarına İlişkin Veri Madenciliği Uygulaması	Çalışmada veri seti Türkiye'de faaliyet gösteren bir havayolu kurumunun parça söküm raporlarından oluşmaktadır. Bu çalışmanın amacı, uçak parçalarında herhangi bir sorun oluşmadan önce düzeltici ve önleyici eylemlerin alınabilmesi için ikaz seviyelerinin belirlenmesinde kullanılacak kuralları ortaya çıkarabilmektir. Sınıflandırma modelinden yararlanılan çalışma sonucunda her bir parça için bulunan ve sayısal bir parametre sayılan falert (İkaz Seviyesi Değeri) parametresini etkileyecek, diğer değerler belirlenerek, ikaz seviyesi değerini tahmin edecek kurallar çıkarılmıştır (Gürbüz, Özbakır, & Yapıcı , 2009).
Veri Madenciliği: Karar Ağacı Algoritmaları Ve İMKB Verileri Üzerine Bir Uygulama	Çalışmada veri seti İMKB 100 endeksinde sanayi ve hizmet sektörlerinde faaliyet gösteren 173 işletmenin 2004–2006 yıllarına ait yıllık finansal göstergelerinden oluşmaktadır. Karar ağacı analizinin uygulandığı çalışmada, CHAID algoritmasından faydalanılarak baz alınan göstergelere göre işletmeleri gruplamada yardımcı olacak değişkenler tespit edilmiştir. Çalışma sonucu işletmeler 7 gruba ayrılmış ve işletmelerin ayrımında hedef değişken yani en önemli değişkenin sektör değişkeni olduğu sonucuna varılmıştır (Albayrak & Koltan, Yılmaz, 2009).

Kaynak: Yazar tarafından derlenmiştir.

1.7.Veri Madenciliğinde Kullanılan Yazılımlar

Dünyada hızla artan veriyi analiz edebilmek için yapılan hesaplamalarda ciddi kolaylık sağlayan bilgisayar ve bilgisayar programları günümüzde hala geliştirilmektedir. Veri madenciliğinin yapılabilirliğini sağlayan yazılımlar teknoloji ve veri madenciliğinde yaşanan gelişmeler ile şekillenmiştir. 1960'lı yıllarda Fortan yazılımı; 1980'lerde Oracle, IBM, DB ve SQL; 1990'larda SAS, IBM Intelligent Miner, Angoss, SPSS CRISP, DM, DARWIN ve standart veri ambarları; 1990'lı yılların sonundan günümüze kadar Oracle Data Miner, IBM DB2 UDB MINING, SAS Enterprise Miner ve SPSS Clementine yazılımları kullanılmıştır. Bu yazılımlar ile başarılı veri madenciliği çalışmaları yapılmıştır. HSBC Amerikan şirketi veri madenciliği çalışmasında SPSS programından yararlanarak müşteri ihtiyaçlarını ve tüketim davranışlarını elde etmiştir. Çalışma sonuçlarından hareketle, giderlerini %30 düşürürken satışlarını da %50 arttırmıştır. VERİZON şirketi ise, giden müşterilerinin sadakatini sağlamak amacıyla veri madenciliği çalışmasında SPSS Clementine programından faydalanmıştır. Çalışma sonuçlarından hareketle, müşterilerini belirli gruplara ayırarak en çok kayıp olan müşteri gruplarını tespit etmiştir. Yeni pazarlama ve promosyon çalışmaları ile müşterilerin bağlılığı arttırılmaya çalışılmıştır. Ek olarak, maliyet ve satış optimizasyonu da sağlanmıştır. Genel olarak çalışmalarında SAS programını tercih eden Türkiye İstatistik Kurumu (TÜİK), 2013 yılında gerçekleştirdiği "Hane Halkı Bütçe Anketi" adlı projesinde de bu programdan yararlanmıştır. Garanti Bankası, müşteri ilişkilerinde memnuniyeti sağlamak için müşteri verileri üzerinde gerçekleştirdiği veri madenciliği çalışmalarında SAS Enterprise Miner programına başvurmaktadır (Çıngı, 2007, s. 9-10). Tablo 6, veri madenciliği uygulamalarında kullanılan yazılımların çalışmakta olduğu platform ve hangi modeller için kullanıldıklarına dair bilgi vermektedir.

Tablo 6: Yazılım Sağlayıcılar

Ürün Adı	IBM Intelligent Miner	Oracle	SAS	Angoss	NCR Teraminer Stats	WEKA
Platform	AIX 4.1, NVS, AS/400, Windows	Windows	Masintosh, Windows, Unix	Windows, Unix	Windows, Unix	Masintosh Windows U
Karar Ağacı	X	X	X	X		X
Sinir Ağları	X	X	X		X	
Zaman Serisi	X		X			
Tahmin	X	X	X	X		X
Kümeleme	X		X			X
Birliktelik	X		X			X
Görselleştirme	X	X	X	X	X	X

Kaynak: AKBULUT, S. (2006). Veri Madenciliği Teknikleri İle Bir Kozmetik Markanın Ayrılan Müşteri Analizi Ve Müşteri Segmentasyonu. Yüksek Lisans Tezi. Ankara: Gazi Üniversitesi Fen Bilimleri Enstitüsü. s.13.

Günümüzde veri madenciliğinde kullanılan yazılımlar ücretli ve ücretsiz erişilebilir olmak üzere ikiye ayrılmaktadır. Ticari yazılımlar; SPSS, SPSS Clementine, EXCEL, SAS (Enterprise Miner), Angoss, Kxen, MS SQL Server, MATLAB, Oracle'ın geliştirdiği modeller iken, açık kaynak kodlu yazılımlar; Keel, Knime, Orange, R, C4.5, Rapıd Miner (Yale) ve WEKA'dır.
Veri biliminde önemli araştırmaları gerçekleştiren ve bu alana özgü her yıl anket düzenleyen KDnuggets dergisi, 912 seçmenin katıldığı "Veri Madenciliği/Analitik Araçlar Kullanılan Anket" adlı araştırmasında veri madenciliği projelerinde başvurulan yazılımların tercih edilme oranları elde edilmiştir (KDnuggets, 2010). Tablo 7'deki sonuçlara ulaşılmıştır.

Tablo 7: Veri Madenciliği Projelerinde Başvurulan Yazılımların Tercih Edilme Oranları

Açık Kaynak Kodlu Yazılımlar		Ticari Yazılınlar			
RAPİD MİNER	37.8 %	EXCEL	24.3 %	SQL	6.9 %
R	29.8 %	SAS	12 %	SAS MINER	5.5 %
KNIME	19.2 %	MATLAB	9.2 %	KXEN	2.1 %
WEKA	14.3 %	SPSS	7.9 %	ANGOSS	0.9 %
ORANGE	2.7 %	SPSS CLEMENTINE	7.3 %		

Kaynak: KDnuggets Dergisi (2010). Veri Madenciliği/Analitik Araçlar Kullanılan Anket. https://www.kdnuggets.com/polls/2010/data-mining-analytics-tools.html (3 Mart 2018).

Tablo 7'den hareketle, 2010 yılında uzmanlar tarafından açık kaynak kodlu yazılımlardan en çok tercih edilen ilk iki yazılımın Rapid Miner ve R programlarının olduğu görülmektedir. Ticari yazılımlarda en çok tercih edilen yazılım ise büyük bir çoğunlukla Excel programı olduğu görülmektedir.

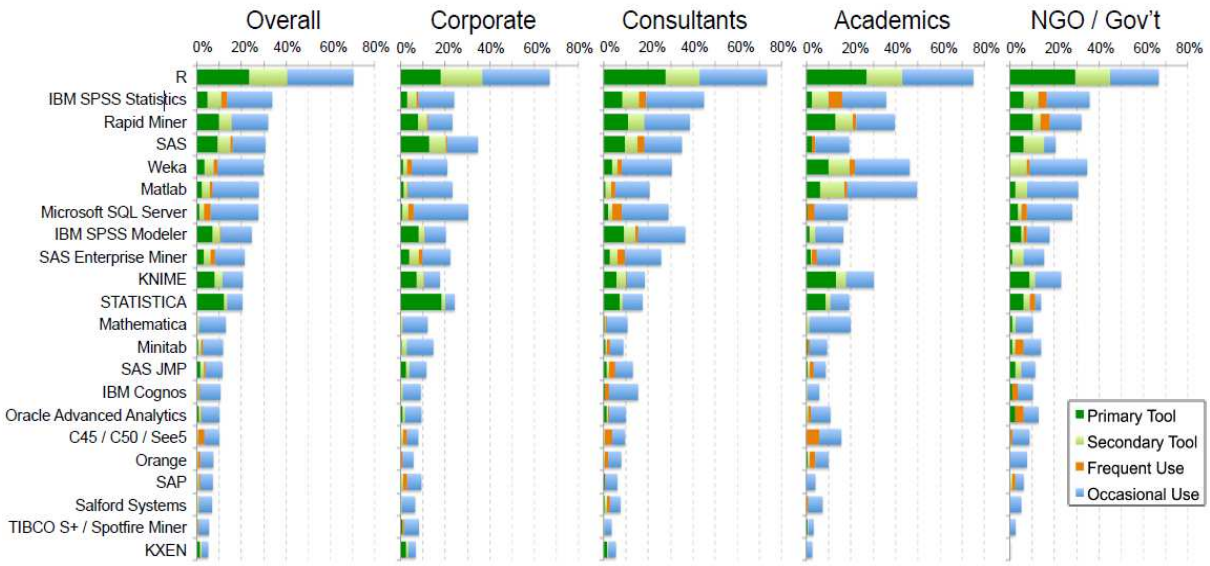
Veri biliminde önemli araştırmalara ev sahipliği yapan bir diğer kurum Rexer Analytics'in 2013 ve 2015 yılında düzenlediği anketlerde bu alanda yetişmiş nitelikli iş gücü tarafından en çok tercih edilen yazılımların bilgileri yer almaktadır. Yazılımlara sahip olan kurumların yazılımlarını iyileştirmesi ve sektör çalışanların ihtiyaçlarını cevap verecek hale getirme çabalarından dolayı iki anket arasındaki yazılım sıralamalarında değişiklikler göze çarpmaktadır. 2013 anket sonuçları Grafik 1'de ve bu grafikte yer alan değerlerin açıklamaları Tablo 8'de gösterilmektedir.

Tablo 8: Grafik 1'de Yer Alan Değerler için Açıklamalar

Gösterge	Açıklama
Overall Grafik	Veri madenciliğindeki tüm kullanıcılar için birincil araç seçimleri
Corporate Grafik	Kurumsal bünye çalışanları için birincil araç seçimleri
Consultants Grafik	Danışmanlık yapan uzmanlar için birincil araç seçimleri
Academics Grafik	Akademisyenler için birincil araç seçimleri
Ngo/Gov't Grafik	Sivil toplum kuruluşlarındaki kullanıcılar için birincil araç seçimleri
Primary Tool	Birincil tercih edilen yazılım araç değeri
Secondary Tool	İkincil tercih edilen yazılım araç değeri
Frequent Use	Sık tercih edilen yazılım araç değeri
Occasional Use	Ara sıra kullanılan yazılım araç değeri

Kaynak: Rexer Analytics. (2013). Data Miner Survey/ Summary Report. s.31., 1-41. URL: www.RexerAnalytics.com (12 Nisan 2018).

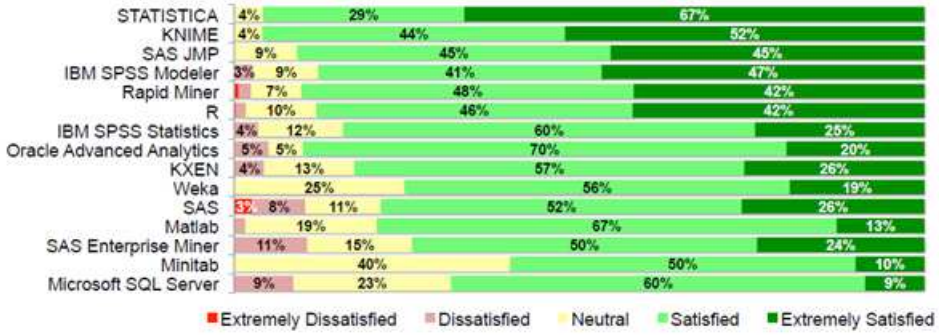
Grafik 1: Uzmanların Birincil Araç Seçimleri



Kaynak: Rexer Analytics. (2013). Data Miner Survey/ Summary Report. s.31., 1-41. URL: www.RexerAnalytics.com (12 Nisan 2018).

2013 anket sonuçlarına göre, veri madenciliği alanındaki kullanıcılar tarafından R, IBM STATISTICS, RAPID MINER, SAS ve WEKA çoğunlukla kullanılan yazılım araçları olarak ilk beşte yer almaktadırlar. Uzmanlar ortalama olarak bu beş yazılımı seçerken, bu yazılımları çalışmalarının %76'sında birincil olarak tercih ettiklerini ifade etmişlerdir. Ayrıca, veri madencilerinin, %64 ü kendi kodlarını yazdıklarını belirtmiştir. En çok tercih edilen diller arasında SQL (%43), JAVA (%26) ve PHYTON (%24) olarak belirlenmiştir. Ayrıca, Academics adlı grafikten hareketle akademisyenler tarafından birincil olarak en çok tercih edilen programlar R, RAPID MINER ve KNIME olarak görünmektedir (Rexer, 2013, s. 31). Rexer 2013 anketinden alınan Grafik 2'de kullanıcıların birincil ve ikincil olarak tercih ettikleri yazılımları "Son Derece Memnuniyetsiz (Extremely Dissatisfied), Hoşnutsuz (Dissatisfied), Nötr (Neutral), Memnun (Satisfied) ve Son Derece Memnun (Extremely Satisfied)" şeklinde 5'li likert ölçeğe dayanarak değerlendirmesini vermektedir (Rexer, 2013, s. 32).

Grafik 2: Birincil ve İkincil Yazılım Araçlarında Sağlanan Memnuniyet Oranları (%)



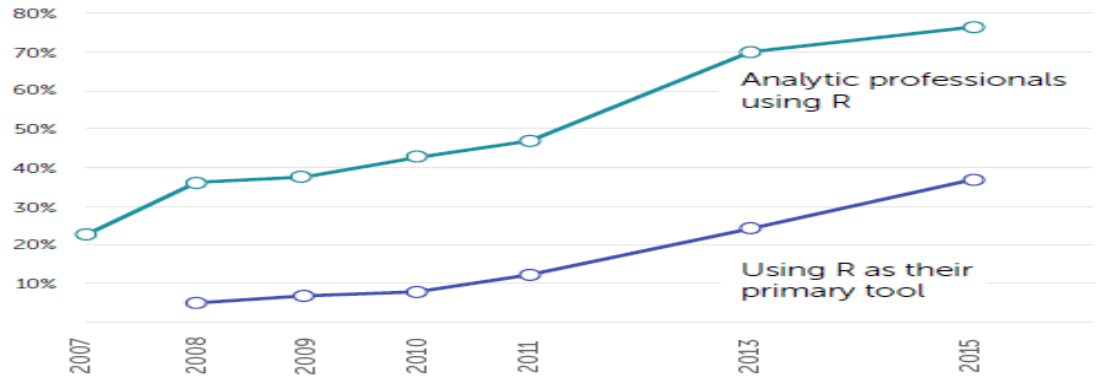
Kaynak: REXER, K. (2013). 2013 Data Miner Survey/ Summary Report. s.32., 1-41. www.RexerAnalytics.com (12 Nisan 2018).

2013 yılı anketinden alınan Grafik 2 ise veri uzmanlarının Birincil ve ikincil olarak başvurdukları yazılımlardaki memnuniyetlik durumlarının 5'li likert ölçek tipinde değerlendirilme sonuçlarını vermektedir. Grafikte göze çarpan STATISTICA ve KNIME yazılımları 2011 anket sonuçlarında olduğu gibi 2013 anket sonuçlarında da en yüksek memnuniyet değerlerine sahip yazılımlar olmuşlardır. Bu iki yazılımı memnuniyetin olumlu olması açısından SAS JMP, IBM, SPSS MODELER, RAPID MINER ve R yazılımları takip etmektedir. Bu grafikte en dikkat çekici durum ise, kullanıcıların birincil olarak seçtiği yazılımdan memnun olduklarında ikincil olarak seçtiği yazılımlardan da memnun kalmışlardır ve bu durum şekillerin benzerliği ile de anlaşılmaktadır. Ancak, SAS ENTERPRISE MINER ve IBM SPSS MODELER yazılımlarını ikincil araç olarak seçen kullanıcılar bu yazılımları değerlendirirken düşük değerler verirken, IBM SPSS STATISTICS yazılımlarını ikincil araç olarak seçen kullanıcılar daha yüksek değerler vermişlerdir. Kullanıcıların büyük bir çoğunluğu birincil tercih ettiği yazılımlar ile çalışmaya devam edeceğini belirtmişlerdir. En yüksek devam oranı KNIME yazılımını birincil olarak tercih edenler arasında ve kullanıcıların %85'inin önümüzdeki 3 yıl süresince KNIME yazılımını birincil araç olarak tercih edecekleri beklenmektedir. Ayrıca, R ve STATISTICA yazılımları da kullanıcıları tarafından çok yüksek devamlılık paylarına sahiptirler. Kullanıcıların geneli birincil araç olarak seçtiği yazılımı değiştirmek istediğinde R yazılımını tercih etmektedirler (Rexer, 2013, s. 32).

Rexer Analytics 2015 anket sonuçları raporundan elde edilen Grafik 3'ten hareketle, 2010 anket sonuçlarında en çok tercih edilen R programına olan talep, 2015 yılında hem uzman analistler

tarafından hem de diğer kullanıcılar tarafından talebin arttığı gözlenmiştir. Raporda kullanıcıların gözünden R programının maliyet, model performansı, algoritma çeşitliliği ve algoritmaları değiştirmek, kendi kodunu yazma özgürlüğü, tekrarlanan işleri otomatik hale dönüştürmek ve grafik görselliğinin sunduğu imkanlar R programının olumlu yönleri olarak belirlenmiştir. Programın hız ve kullanım kolaylığı kriterleri ise olumsuz yönleri olarak belirlenmiştir (Allen, Gearan, & Rexer, 2015, s. 10).

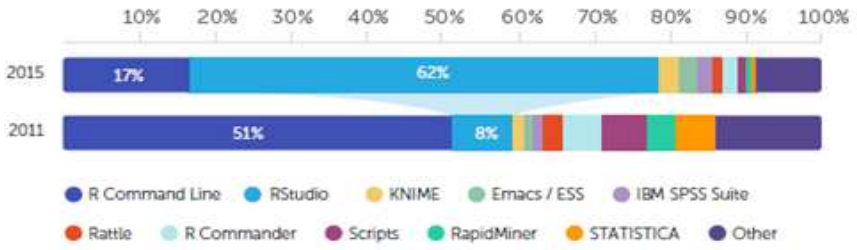
Grafik 3: Yıllara Göre R Programının Kullanımındaki Artış



Kaynak: ALLEN, H., GEARAN, P. ve REXER, K. (2015). 2015 Data Science Survey, Rexer Analytics. s.13., 1-39. www.RexerAnalytics.com (4 Mayıs 2018).

2015 yılı anket sonuçlarında R programının en çok tercih edilen program olmasının yanı sıra, Grafik 4'te görüldüğü üzere, %62'lik bir kullanıcı grubu ile RStudio en çok tercih edilen ara yüz seçilmiştir (Allen, Gearan, & Rexer, 2015, s. 12).

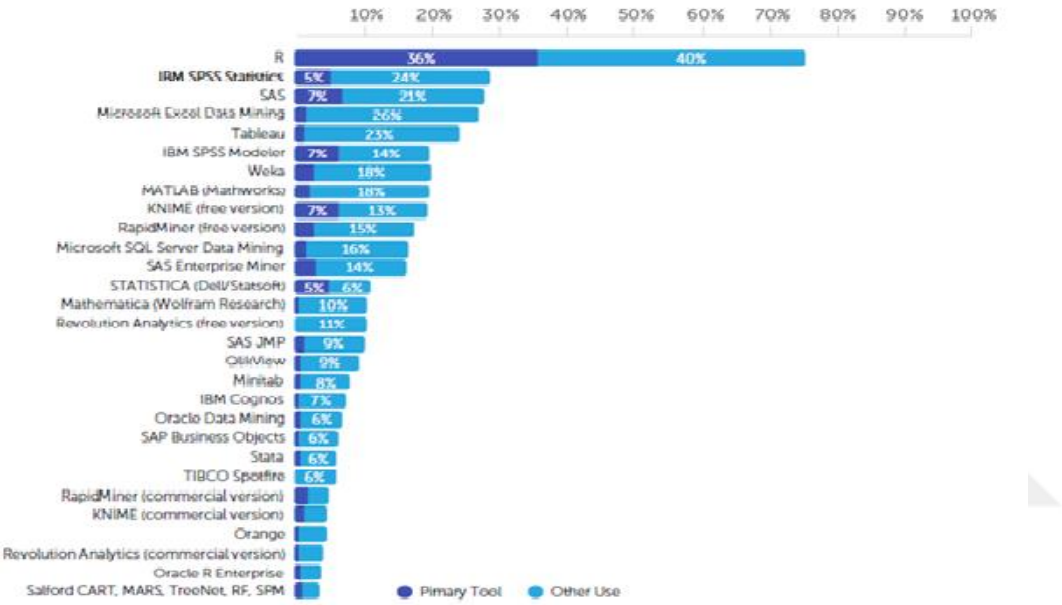
Grafik 4: En Çok Tercih Edilen Ara Yüz: RStudio



Kaynak: ALLEN, H., GEARAN, P. ve REXER, K. (2015). 2015 Data Science Survey, Rexer Analytics. s.15, 1-39. www.RexerAnalytics.com (4 Mayıs 2018).

2015 yılı anketinden alınan Grafik 5'te veri uzmanları tarafından yazılımların birincil olarak tercih edilme oranları gösterilmektedir. Buna göre, "%36'lık bir pay ile R en çok tercih edilen program olarak belirlenirken, kalan %64'lük pay içinde SAS, IBM SPSS MODELER, KNIME, IBM SPSS STATISTICS ve STATISTICA programları diğer en çok tercih edilen programlar arasında yer almıştır (Allen, Gearan, & Rexer, 2015, s. 13).

Grafik 5: Yazılımların Tercih Edilme Oranları (%)

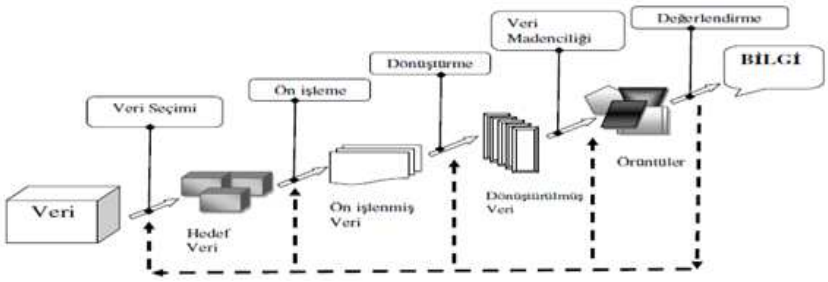


Kaynak: ALLEN, H., GEARAN, P. ve REXER, K. (2015). 2015 Data Science Survey, Rexer Analytics. s.16., 1-36. www.RexerAnalytics.com (4 Mayıs 2018).

1.8.Veritabanlarında Bilgi Keşfi Süreci

Genel analiz araçlarının büyük veri karşısında etkinliğinin azalması ile Veri tabanlarında Bilgi Keşfi adı altında oluşan yeni yapılanmanın genel işleyişi Şekil 7'de gösterilmektedir. KDD sürecinde her bir adım birbirine bağımlı olarak gelişmektedir. Buradaki bağımlılıktan kasıt uygulama sürecinde her bir adımın bitişi, kendisinden sonraki adımın başlangıcına yol açmasıdır. Yani, bir zincirdeki halkalar gibi birbiri ile bütünleşik yapıdadır ve herhangi bir adımdaki problem doğru bir şekilde ve doğru zamanda giderilmediğinde büyük problemler yaşanabilmektedir. KDD süreci altı adımdan oluşmakla beraber modelin kurulması ve

değerlendirilmesi adımlarında gelişen veri madenciliği, bu sürecin parçası olması yanı sıra başlı başına bir süreçtir.



Şekil 7: Bilgi Keşfi Sürecinde Veri Madenciliği

Kaynak: SAVAŞ, S., TOPALOĞLU, N. ve YILMAZ, M. (2012). Veri Madenciliği ve Türkiye'deki Uygulama Örnekleri. İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi. Yıl:11, S.21, Bahar, s.8., 1-23.

KDD sürecindeki ilk adım araştırmaya konu olacak ve araştırmanın çerçevesini sunacak araştırma probleminin tanımlanması (Business Understanding) adımıdır. İkinci adım ile bu problemin çözümüne cevap vereceği düşünülen veri setini anlama yani veri tanıma (Data Understanding) adımıdır. Veri hazırlama (Data Preperation) üçüncü adımdır. Veri hazırlama sürecinde zamanla mevcut ham veri de meydana gelen tahribat veya farklı nedenlerden dolayı oluşan ve araştırma sonuçlarının etkileyecek birçok problem ile karşılaşılmaktadır. Veri de karşılaşılan bu problemleri düzeltmeden ileri aşamalara geçilmemelidir. Gerçek hayatta, güçlü bir araba için motor nasıl önemli bir faktör ise sağlam sonuçlara sahip bir veri madenciliği uygulaması için veri öncelikli öneme sahiptir. Bu problemler farklı tipteki verilerin uyumsuzluğu, artık veri, kayıp veri (missing data) eksik veri, uç değerler, gürültülü veri ve belirsizlik şeklindedir. Güvenilirliği ve geçerliliği sağlanan bir çalışma için bu problemlerin ortadan kaldırılması gerekmektedir.

Araştırma problemi veya problemlerine çözüm olacak veri setinde yinelenen değerler ya da işe yaramayacak gereksiz değerlerin olması artık veri problemine yol açmaktadır. Bu değerlerin veri setinden ayıklanması gerekmektedir. Gerçek hayatta veri sadece nicel ya da nitel nitelikte bulunmamaktadır. Ayrıca, veriler kendi içinde bazı özellikleri itibariyle birbirinden ayrılmaktadır. Bu durum, özellikle heterojen kaynakların birleşiminde farklı tipte verilerin uyumsuzluğu problemini oluşturmaktadır. Verilerin birbiri arasındaki tutarlılığın sağlanması ile

bu problem çözülebilir. Yani, veri seti oluşturulurken veriler özenle seçilmeli ve tutarlılık gözden kaçırılmamalıdır.

Veri toplama ve veri girişi görevini üstlenen çalışanlar tarafından veriler hatalı ve eksik girilebilir ya da hiç veri girişi yapılmayabilir. Bu sistem dışı hatalar gürültü ve kayıp veri olarak adlandırılmaktadır. Gürültü, veri setindeki uç, eksik ve yanlış gözlemlerin var olması ile oluşmakta ve araştırma sonuçlarında ciddi anlamda yanılmaya sebep olmaktadır. Bir veri setinde sayısal anlamda gürültünün fazla oluşu belirsizlik problemine neden olmaktadır. Gürültüye sebep olan uç değerler için istatistiksel yöntemlerden hareketle, tahmini değerler belirlenebilir. Araştırma planı açısından zaman kısıtlaması yoksa yanlış giriş ile oluşan gürültü probleminde manuel olarak düzeltme tercih edilebilir. Bu teknikler haricinde gürültü problemine karşılık olarak veri setindeki değerler küçükten büyüğe doğru dizilip, 3'erli gruplar oluşturulur. Bu gruplara ait aritmetik ortalamalar söz konusu grup için ortak değer olarak atanabilir. Buradaki grup sayısının araştırma konusu ve yöntemi esas alınarak belirlenmesi doğru olacaktır. Bu teknik yerine yine her bir gruptaki en az ve en fazla olan değerler belirlenip, her grup için bu değerler, diğer değerlere yakınlık esas alınarak yerine yazılması yöntemi uygulanabilir. Bir başka teknik ise, veri setindeki her bir grup için en az ve en fazla değerler belirlenerek, bu iki değer arası fark hesaplanır. Her gruba ait fark değeri grup eleman sayısına oranlanır. Çıkan değer ilgili gruptaki değerlerin yerine yazılır. Böylece, düzgünleştirilmiş bir veritabanına ulaşılır (Köse, 2015, s. 66-67-68).

Veri madenciliği ve diğer çalışmalarda yapılan eksik giriş ve hiç yapılmayan giriş sebepleri ile oluşan kayıp veri problemi araştırma sonuçlarını olumsuz yönde etkilemektedir. Bu durumun en önemli sebebi uzmanların yöntem ve algoritmaları tüm değerlerin var olması şartı ile doğru çalışır mantığından hareketle geliştirmesidir. Genel olarak, kayıp veriler çalışan ihmali, veri seti boyutu veya yapısından kaynaklanmaktadır. Kayıp verinin araştırma sonuçlarındaki olumsuz etkisini azaltma ve ortadan kaldırma amacı ile bazı teknikler geliştirilmiştir. Öncelikle kayıp veri, veri setinde kapladığı alana bağlı olarak gözden çıkartılabilir. Veri setindeki büyük bir orana sahip kayıp veri probleminde bu teknik araştırma sonuçlarını tehlikeye düşürmektedir. Diğer bir yol ise araştırma planı açısından zaman sınırlaması olmaması ve verilere ulaşım mümkün olması durumunda kayıp veriler manuel olarak veri setine eklenebilir. Araştırmada kullanılan yöntem ve algoritmanın sonuçlarında sapma ihtimali taşıyan bir diğer yol ise veri setindeki tüm kayıp değerlere aynı sembolik değerin atanmasıdır. Veri setinde oluşturulan her bir kategori altındaki değerler ortalamasının bağlı olduğu kategori altındaki kayıp veriye

atanması yöntemi de uygulanmaktadır. Kayıp verilerin ortadan kaldırılması için istatistiksel yöntemlere de başvurulmaktadır. Regresyon yöntemi ile mevcut verilerden hareketle, bir regresyon denklemi oluşturarak kayıp veriler için tahmin yapılabilir. Bu yönteme ek olarak zaman serisi analizi ve karar ağaçları kayıp veri problemi için tercih edilmektedir.

Veri hazırlama adımı gerçekleştirildikten ve problemler giderildikten sonra sırasıyla modelin kurulması ve değerlendirilmesi (Modeling and Evaluation), uygulama (Deployment) ve modelin izlenmesi adımları gerçekleştirilir.

1.8.1. Araştırma Problemin Tanımlanması

İlk ve en önemli aşama olan araştırma probleminin tanımlanması, araştırmanın amacını, araştırma sonuçlarının nasıl değerlendirileceği, veri madenciliği amaçlarını ve araştırma planlaması sürecinin belirlenmesini kapsamaktadır.

1.8.2. Veri Anlama

Tüm bilimsel çalışmalarda kaynak olan veri, yapılan çözümlemelerin hem başlangıç sebebi hem de bitiş meyvesi olarak görülmektedir. "1.1. Genel Tanım ve Kavramlar" adlı başlıkta açıklanan KVA'da araştırma öncesi verinin neler ifade ettiği ve ne kadar önemli olduğundan bahsedilmişti. Verinin değeri ve kalitesi çalışma sonucu ulaşılan modelde kendini göstermektedir. Araştırma seyri için çok önemli olan bu adım, en az ilk adım kadar üzerinde durulması gerekmektedir. Veri toplama ile başlayan bu adımdan sonra araştırmacı için kritik nokta toplanan veriyi iyi bir ön işlemden geçirebilmektir. Elde edilen veriler ön işleme ve veri hazırlama adımına tabi tutularak, verilerin tanımlamaları yapılır, benzer özellikte olan veriler bir araya getirilir ve veri keşfine başlanarak gözle görülemeyen ilişkiler gruplandırılmaktadır.

1.8.3. Veri Hazırlama

KDD sürecinin dörtte üçünü kapsayan veri hazırlama adımı, bir önceki adımda toplanan ve araştırma problemi için kabul gören ham verinin incelenmesi ve bu adımın alt süreçlerinden geçerek sağlam, güvenilir ve araştırma amacına ışık tutacak bilgilere ulaşmayı sağlayacak son veriye kadar olan süreci kapsamaktadır. Bu adım toplama (Collection), değer biçme (Assessment), birleştirme (Consolidation), temizleme (Cleaning), seçim (Selection) ve dönüştürme (Transformation) adı altında 6 alt süreçten oluşmaktadır. Bu adımdaki amaç

modelde kullanılacak verinin doğru sonuçlar vermesini sağlamaktır. Bu durumda veri kalitesinin önemi ve güvenilirliği gündeme gelmektedir. Sonuç olarak, kaliteli ve sağlam veriler araştırma sonunda kaliteli ve güvenilir çıktılar verecektir (Oğuzlar, 2003, s. 67).

Araştırma ekibinin veri toplama aşamasında yapacağı en önemli tespit, ihtiyaç duyduğu verinin ne olduğudur. Araştırmacının perspektifi ve konuyu ele ele alış biçimi toplanacak verinin karakterinde belirleyici olmaktadır. Toplama, belirlenen araştırma problemi için gerekli olan veriler için veri kaynaklarının belirlendiği ve bu verilerin toplandığı adımdır (Döşlü, 2008, s. 18).

Büyük veritabanlarına sahip firmaların ya da büyük bir çerçeveye sahip veri madenciliği çalışmalarında verilerin birden fazla farklı kaynaktan elde ediliyor olması veri yapısında uyumsuzluklara sebep olmaktadır. Farklı veri kaynaklarından toplanan verilerde en çok karşılaşılan durumlara verilerin aynı tarih aralığına ait olmamaları, verilerin yapısı sebebiyle farklı ölçü birimine tabi olmaları, veri kaynaklarına sahip kişilerin ve kurumların kendilerine özel kodlar ile veri girişi yapmaları, veri kaynaklarında yapılan eksik ve hatalı girişler ve verilerin toplanma amacı gereği asıl sorulara yanıt vermede zayıf kalması sayılabilir. Araştırmacı/araştırmacıların sonuçta ulaşmak istedikleri model ve model sonuçlarının güvenilirliği ve etkinliği açısından değer biçme adımında, veri setindeki her bir birim için doğru değerler biçilmeli ve bu birimler arası uyum sağlanması gerekmektedir.

Veriler, "1.1. Genel Tanım ve Kavramlar" adlı başlıkta açıklanan veri kaynaklarına ek olarak online sitelerden veya kurumların farklı birimler için hazırlanan veritabanlarından elde edilebilir. Çalışma için farklı kaynaklardan toplanan verilerin ayıklanıp tek bir form altında bütünleştirilmesi gerekmektedir. Dolayısıyla, veri kaynaklarının birleştirilmeden önce temizlenmesi ve standardize edilmesi gerekmektedir. Temizleme görevi, değer biçme adımında veri kaynaklarındaki kaydedilen sorunları ortadan kaldırmak ile başlamaktadır.

Birleştirme görevinde elde edilen temiz ve heterojen kaynaklar bir araya getirilerek standart bir form oluşturulmaktadır. Heterojen veri yapısını bütünleştirebilmek için veri ambarına ihtiyaç duyulmaktadır. Veri ambarında veriler ve veri kaynakları zaman ve diğer nitelikler bakımından birbirleriyle uyumlu hale gelmektedir (Şentürk, 2006, s. 28). Seçim adımında, araştırma problemine cevap verebilecek model için uygun bir veri havuzu oluşturulmaktadır. Veri kaynakları incelenirken önişleme, aşırı değer, sapan, eksik veya hatalı yapılan girişlerin düzeltilmesi ve hatalı birimlerin veri setinden arındırılması sağlanmaktadır (Albayrak &

Koltan, Yılmaz, 2009, s. 35). Kısacası, bu aşama planlanan model için değişken seçimi ve modelin denenmesinde yararlanılacak veri setine karar verme sürecini kapsamaktadır.
Veri madenciliği, her bir ihtiyaç ve veri türü için uygun birçok model ve bu modellere özel algoritmalar içermektedir. Bu sebeple, veri setinden çıkarılmak istenen değişkenler ve bu değişkenler için uygun olacak model ve algoritmaya karar vermek için dönüştürme aşaması etkin rol oynamaktadır. Çalışmalarda genel olarak, veri dönüştürme işlemlerinden veri normalleştirmeye başvurulmaktadır. Bunun dışında veriden en iyi şekilde yararlanmak için düzeltme, birleştirme ve genelleştirme gibi dönüştürme işlemleri yapılmaktadır (Tüzüntürk, 2010, s. 69).

1.8.4. Modelin Kurulması ve Değerlendirme

Çalışmalar sonrasında oluşturulan veri setinin bir istatistik uzmanı bakış açısı ile incelenerek öncesinde fark edilmeyen ve ortaya çıkarılmamış niteliklerin gün yüzüne çıkarılması sağlam bir veri madenciliği çalışması olarak kabul görebilmektedir (Vahaplar, 2003, s. 35-36). Araştırmacı elde edilen veriyi özetleyip, veri madenciliği uygulamalarında kullanılan ileri çözümlemeler sunan yazılımların sunduğu imkanlar sayesinde gözlemlenemeyen örüntüleri bulmaya çalışmaktadır. Bu aşama, veriden bu bilgiyi çekebilmek için veriye uygun model ve algoritmanın seçimi, bir pivot çalışmanın üretimi ve model geliştirme adımlarını kapsamaktadır. Bir veya birden fazla geliştirilen modellerin araştırma problemine cevap verip vermediğine bakılır. Bilgisayar sistemlerinin sunduğu gelişmiş hesaplama seçeneklerine rağmen yüksek boyutlu veri ambarlarında birden fazla modelin çalıştırılması zaman açısından sorun yaratmaktadır. Bu seçenek yerine veri ambarından rasgele seçilen bir bütün üzerinden hareketle, birden fazla modelin denenmesi ve optimum sonuçlara ulaşan modelin tercih edilmesi daha sağlıklı ve hızlı bir sonuç verecektir. Araştırmacı bu aşamaya kadar yaptığı çalışmayı gözden geçirir ve bundan sonra araştırma sürecinin ne şekilde devam edeceğine dair sonuçlara varmaktadır. Araştırmacı, araştırma problemine cevap veren en uygun modele karar verip, uygulama aşamasına geçmektedir.

1.8.5. Uygulama

Veri madenciliği sürecinde bu aşama, bir önceki aşama da geçerliliği ve güvenilirliği kabul gören en uygun modelin uygulama da kullanıldığı aşamadır. Seçilen model, doğrudan bir

çalışma olabileceği gibi, farklı bir çalışmanın alt parçası olarak da işlem görebilir (Akpınar, 2000, s. 13). Bu aşamaya gelen bir çalışmada araştırma ekibi sorularını yanıtlamaya başlamaktadır. Uygulama sonuçları bazı problemlere çözüm olarak kullanılabilir. Araştırma ekibi tarafından çalışma, a'dan z'ye gözden geçirilerek bir araştırma raporu hazırlanır. Bu şekilde çalışma sonlanmaktadır.

1.8.6. Modelin İzlenmesi

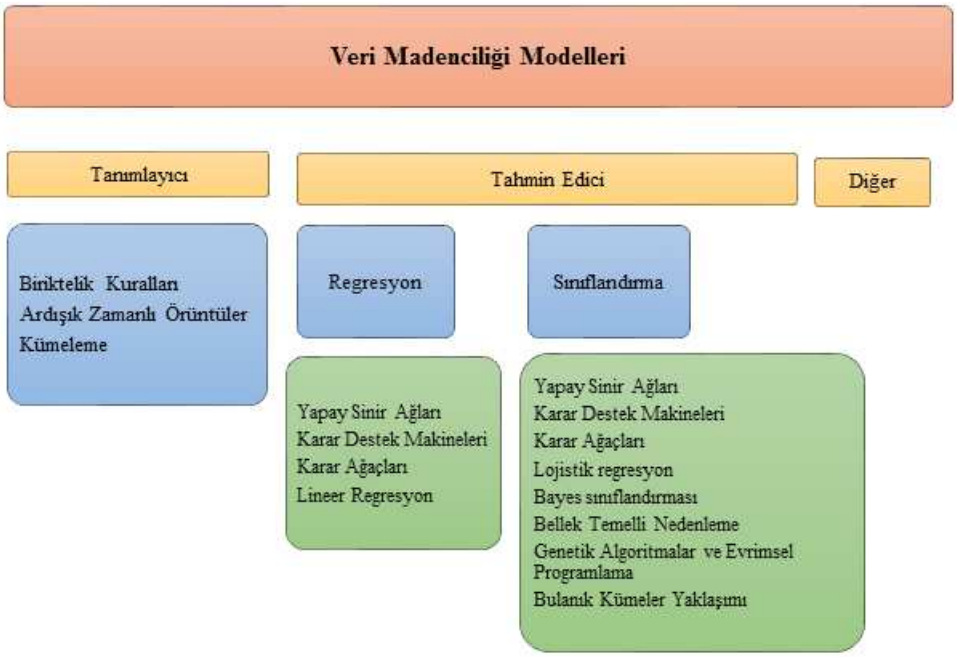
Veri madenciliği süreci sonucunda en güvenilir ve geçerli sonuçları veren bir modele ulaşılmaktadır. Ancak, bu modele veri sağlayan alt yapının zamana bağlı olarak değişimi ve gelişimi sonucu model artık en uygun sonuçları veremez, hale gelebilir. Yani, zaman hassasiyetinden kaynaklı elde edilen kurallar geçerliliğini kaybedebilmektedir. Bu sebeple elde edilen modelin devamlı takip edilmesi ve belirlenen ihtiyaçlara göre yeniden yapılandırılması şarttır.

1.9.Veri Madenciliği Modelleri

Veri madenciliği, verilerdeki gizli bağıntıları keşfetmek ve bu bağıntılar aracılığıyla geleceğe yönelik tahminlerde bulunacak modelleri yaratan bir süreçtir. Burada ilk önce veri seti KVA ile ön incelemeye tabi tutulmaktadır. Veri yapısı, türü, istatistiksel özellikler ve veriden çıkarılabilecek değişkenler ve bu değişkenlerin arasındaki ilişki üzerinde durulur. Bu özellikler görsel düzenlemelerle ifade edilmektedir. Ön incelemeden geçirilen veri seti gerekli düzenleme işlemleri ardından değişken veya değişkenler ve uygun bir veri madenciliği modeli seçimi yapılarak optimum bir model oluşturulmaya çalışılmaktadır. Optimum olarak atanan model üzerinde model oluşumunda kullanılan veri seti dışındaki bir başka veri seti kullanılarak modelin doğruluğu test edilir. Testin olumlu sonuçlanması durumunda model kabul görmektedir.

Bir veri madenciliği çalışmasında başta veri seti olmak üzere ulaşmak istenen sonuçları sağlayacak bir modelin oluşturulması için veri madenciliği modellerine başvurulmaktadır. Veri madenciliğinde kullanılan modeller Tablo 9'da gösterilmektedir. Veri madenciliğinde başvurulan modeller tahmin edici (Predictive), tanımlayıcı (Descriptive) ve diğer istatistiksel yöntemler olmak üzere üç gruba ayrılmaktadır.

Tablo 9: Veri Madenciliği Modelleri



Kaynak: Yazar tarafından derlenmiştir.

Veri madenciliği modelleri üzerinde yapılan bir başka ayrım ise modelleri üstlendikleri nitelik ve işlevlere göre iki dala ayırarak incelemektir. Bu ayrım yöntemi detaylı olarak Şekil 8'de görülmektedir. Bir veri madenciliği uygulaması için hem tahmin edici hem de tanımlayıcı model bir arada uygulanabilmektedir. Ancak, bu durum elde edilen tahmin edici bir modelin aslında bir tanımlayıcı model olması ya da oluşturulan bir tanımlayıcı modelin ileri de tahmin edici bir model olarak kullanılması zorunluluğunu gerektirmemektedir.

Şekil 8: Veri Madenciliği Modellerinin Gruplandırılması

Kaynak: KÖKTÜRK, M. ve DİRSEHAN, T., (2012). Veri Madenciliği ile Pazarlama Etkileşimi. 1.bs. Ankara: Nobel Akademik Yayıncılık. s.6.

Tahmin edici veri madenciliği modelleri, geçmişte yaşanmış ve sonuçları kaydedilen verilerden hareketle bir modelin oluşturulması ve oluşturulan bu modelin henüz gerçekleşmemiş yani sonuçları bilinmeyen veriler üzerinde kullanılarak sonuçların tahmin edilmesi amacını taşımaktadır (Özekes, 2003, s. 67). Özetle, geçmişte yaşanmış ve gelecekte de yaşanması muhtemel durum ve olguların sonuçlarına ulaşılmaya çalışılmaktadır. Bir yatırım kuruluşunun belirli kıstaslara göre geçmişte yaptığı yatırımlardan hareketle, gelecekte daha iyi sonuçları verecek yatırımların hangisi olduğu bilgisine tahmin edici veri madenciliği modelleri ile ulaşabilir. Bir sigorta kuruluşu, bugüne kadar yapmış olduğu poliçeler ve bu poliçelere ait bilgilerden (süre, şartlar, ödeme türü ve şekli, tercih edilen acente bilgisi, öğrenim durumu, cinsiyet, yaş, gelir düzeyi ve meslek gibi demografik nitelikler) hareketle özel kategorilerden oluşan müşteri profili oluşturulabilir. Kuruluş aynı bilgilerden hareketle, gelecek yıl için müşterilere özel sigorta risk düzeylerini tahmin edebilir. Tahmin edici veri madenciliği modelleri sınıflandırma ve regresyon olmak üzere ikiye ayrılmaktadır.

Tanımlayıcı veri madenciliği modelleri, karar vermeye yardımcı olacak potansiyel veri setinden bağıntıların ortaya çıkarılmasını hedeflemektedir (Akpınar, 2000, s. 5). Tablo 10'daki örnek veri seti gelir, cinsiyet, evlilik ve evi olma durumu olmak üzere 4 değişkenden oluşmaktadır.

Tablo 10: Örnek Veri Seti

Değişkenler	Durum1	Kod	Durum2	Kod	Durum3	Kod
Cinsiyet	Erkek	0	Kadın	1		
Gelir	Düşük	1	Orta	2	Yüksek	3
Evlilik Durumu	Bekar	0	Evli	1		
Ev Sahiplik Durumu	Yok	0	Var	1		

Kaynak: Yazar tarafından derlenmiştir.

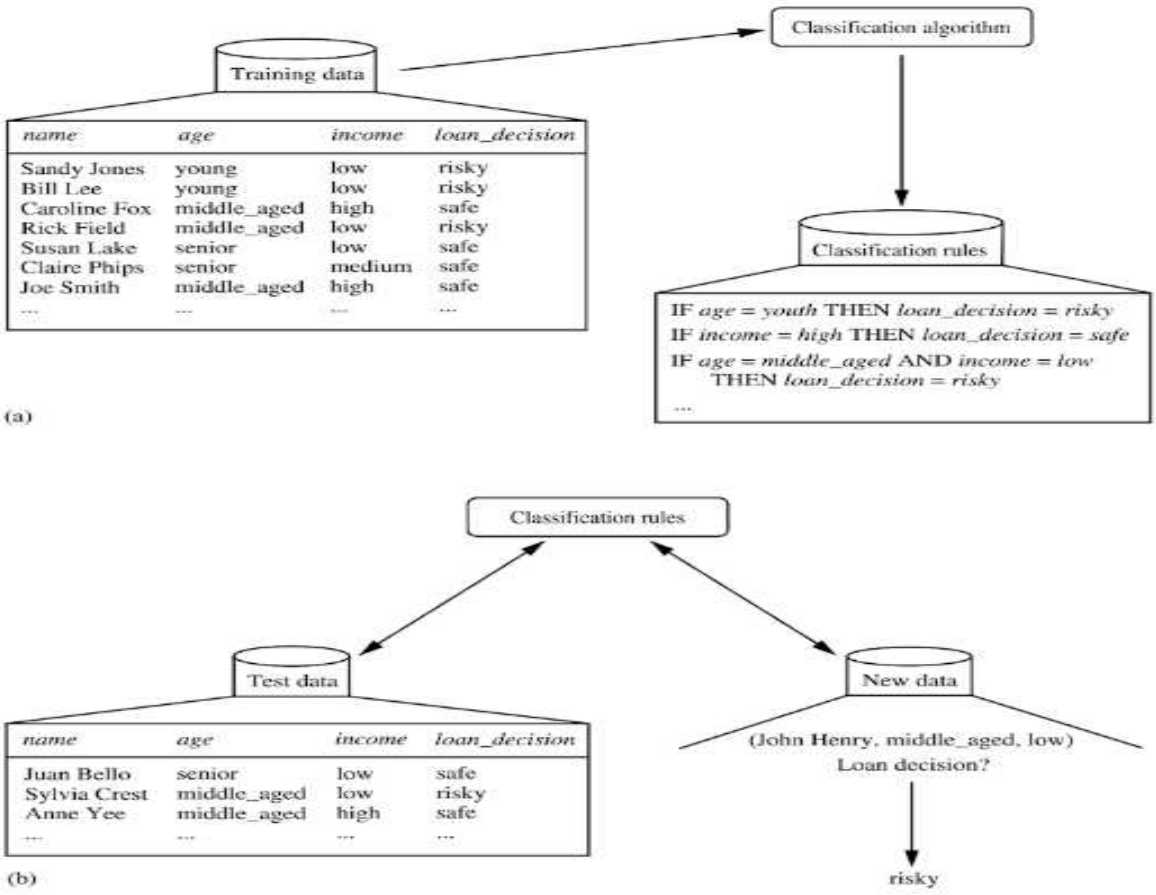
Tablo 10'daki veri setinden X (yüksek maaşı aralığında bekar ve evi olan erkekler) ve Y (düşük maaş aralığında evli ve evi olmayan erkekler) şeklinde 2 grup elde edilmiş olsun. Bu iki gruba ait alışveriş alışkanlıkları arasındaki benzer bağıntıları bulmak için tanımlayıcı veri madenciliği modellerine başvurulabilir. Tanımlayıcı veri madenciliği modelleri birliktelik kuralları, ardışık zamanlı örüntüler ve kümeleme analizi olmak üzere üçe ayrılmaktadır.

Veri madenciliği ile aynı amacı taşıyan birkaç istatistiksel yöntem mevcuttur. İstatistiksel yöntemlerde yer alan çok boyutlu analiz konusu veri madenciliğinde esas alınan amaçlara hizmet etmektedir. Bu konu faktör analizi, varyans analizi, diskriminant analizi, regresyon ve hipotez testleri gibi teknikleri barındırmaktadır.

Veri madenciliği modelleri "1.6. Veri Madenciliği Kullanım Alanları ve Yapılan Çalışmalar" adlı başlık altında bahsedildiği üzere hemen hemen tüm alanlarda karşılaşılan problem ve ihtiyaçlara cevap verecek nitelikte geliştirilmiş ve bu modeller uzmanlar tarafından farklı konularda kullanılarak geliştirilmelerine devam edilmektedir.

1.9.1. Sınıflandırma

Sınıflandırma (Classification), her bir birimin niteliğinin önceden belirlenmiş sınıflardan hangisine ait olduğunu keşfetmeyi hedeflemektedir. Yani, verileri sahip oldukları ortak niteliklere göre ayırır (Diler, 2016, s. 16). Sınıflandırma modeli, kategorik verilerin tahmininde kullanılmaktadır. Öncelikle, veri seti üzerinde bu tekniği uygulayabilmek için uygun bir model geliştirilmelidir. Modeli kurabilmek için, sonuçları önceden kaydedilen durumlar ve bu durumlarda ilgili etkenlerin aldığı değerler gerekmektedir (Argüden & Erşahin, 2008, s. 37). Sınıflandırma modelinin genel işleyişi için Şekil 9'da örnek bir çalışma gösterilmektedir. Burada, bir bankaya kredi talebinde bulunan kişilerin belirli özelliklerine göre riskleri tahmin edilmeye çalışılmıştır. Veri ve değişkenler teorik anlatım için daraltılmıştır. Şüphesiz, gerçek hayatta daha fazla değişken dikkate alınarak modelin uygulanması doğru olacaktır. Veri seti banka müşterilerine ait ad, yaş (genç, orta yaşlı ve yaşlı), gelir (düşük, orta ve yüksek) ve kredi kararı (riskli ve güvenilir) bilgilerinin olduğu değişkenlerden oluşmaktadır. Rasgele seçilmiş bu veriler öğrenme kümesini oluşturur ve bu veriler üzerinde uygun bir algoritma çalıştırılarak sınıflandırma modeli oluşturulmaktadır. Bu model öğrenme kümesinde kullanılan veriler dışındaki diğer veriler üzerinde çalıştırılır. Deneme kümesi olarak adlandırılan bu verilerde modelin doğru çalışması durumunda model kabul edilmektedir. Sonuç olarak, öğrenme kümesine seçilen veriler modelin kurulmasında ve deneme kümesine seçilen veriler modelin test edilmesinde kullanılmaktadır. Kredi talebinde bulunan genç yaştaki kişiler riskli, geliri yüksek olan kişiler güvenilir, orta yaşlı ve düşük gelirli kişiler riskli gibi sınıflandırma kuralları elde edilmiştir (Han & Kamber, 2006, s. 286-287).



Şekil 9: Sınıflandırma Modeli için Örnek Çalışma

Kaynak: Han, J. ve Kamber, M. (2006). Data Mining Concepts and Techniques. (Second Edition). San Francisco. USA: Morgan Kaufmann Publishers/ Elsevier Inc. s.287.

Sınıflandırma modeli, veri madenciliği çalışmalarında en sık başvurulan yöntemlerden biri olması sebebiyle birçok alanda kullanılma fırsatı bulmuştur. Sınıflandırma tahminleyici bir modeldir; havanın bir sonraki gün seyri ya da bir kutudaki renkli topların tahmin edilmesi aslında sınıflandırma işlemidir (Silahtaroğlu, 2013, s. 17). Ürünlerin nitelikleri ile müşteri niteliklerini eşleştirmek için sınıflandırma modelinden yararlanılabilir. Bu sayede, müşteri için en uygun ürün ya da ürün için en uygun müşteri tespiti yapılabilir (Gürgen, 2008, s. 8). Sınıflandırma modeli ve alt teknikleri için başvurulan alanlar Tablo 11'de gösterilmektedir.

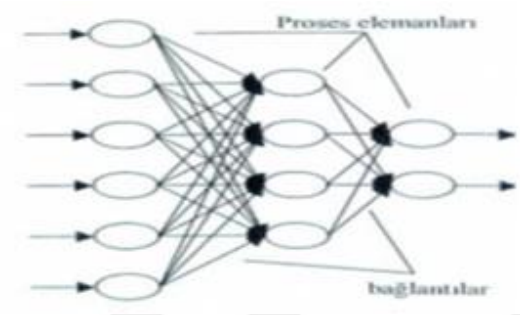
Tablo 11: Sınıflandırma Modeli ve Alt teknikleri İçin Başvurulan Alanlar

Sınıflandırma
Kişisel imzaların eşleştirilmesi, parmak izi tanımlama ve eşleştirilmesi, kredi talep değerlendirmesi, kurumun konumuna bağlı olarak müşteri kalitesinin ve iş fırsatlarının değerlendirilmesi, güvenlik sistemlerindeki görüntü verilerinden şahıs tespit etme, bir ilacın hastalık tiplerindeki iyileştirme yetkinliği, hastadan alınan kan örnekleri aracılığıyla dijital ortamda kötü huylu hücrelerin saptanması (Acharya & Mitra, 2003, s. 18), üretilen ürünler için satış miktarı ile fiyatının ileriki dönemler için analiz edilmesi, personellerin performans ve niteliklerine bağlı olarak sınıflandırılması ve sınıflara göre ücret politikasının belirlenmesi, hastalık türünün tespiti, kalite kontrol çalışmalarında ürün performans değerlendirmesi, güvenlik çalışmalarında sahtekarlık ve dolandırma eylemlerinin engellenmesi, yeni ürün ve hizmet için satış performansının değerlendirilmesi, satışı en iyi olan ürünlerin adet, getiri ve müşteri memnuniyeti gibi faktörlere bağlı olarak tespit edilmesi, risk analizlerinde müşterilerin sınıflandırılması ve risk seviyesinin tahmin edilmesi ve bir ürün veya hizmetin herhangi bir konum için talep görme olasılığının belirlenmesinde yararlanılabilir.
Genetik Algoritmalar
Hisse senedi fiyatlarındaki değişim kalıplarını tahmin edilmesi, müşteri kredi değerliliğinin analiz edilmesi, kredi kartları puanlama, piyasa ile ilgili tahminler yapma, montaj hattı içinde her iş istasyonundaki toplam işlem zamanını minimize etmeyi amaçlayan problemin çözümü ve bazı kısıtlamalar altında kuruluşun lokasyon seçimi için en fazla verimliliği sağlayacak konumun seçiminde (Erdoğan Ş. Z., 2004, s. 24-25) yararlanılabilir.
Karar Ağaçları
Pazarlama biriminde mektup ile yüksek cevap verme oranına sahip kitlelerin belirlenmesi, kişilere ait eski verilerden hareketle kredi taleplerinin belirlenmesi, personel seçiminde eski çalışan bilgilerinden hareketle en iyi aday seçimine karar verilmesi, tıbbi gözlem verilerinden yararlanarak optimum kararların verilmesi, satışı olumlu/olumsuz etkileyen birimlerin tespiti (Akpınar, 2000, s. 14-15) ve sigorta işlemlerinde risk analiz çalışmalarında yaş, cinsiyet, meslek, eğitim ve araba markası gibi faktörlere göre farklı sigorta prim oranlarının uygulanmasında yararlanılabilir.
YSA
Çek okuma, bir ürünün yer aldığı pazar içindeki verimliliğinin tahmin edilmesi, akıllı araçlar için en doğru yolun belirlenmesi, bir sistemdeki işlerin en iyi şekilde planlanmasında ve iletişim araçlarındaki parazitlerin giderilmesinde (Öztemel, 2006, s. 36) yararlanılabilir.

Kaynak: Yazar tarafından derlenmiştir.

Sınıflandırma modeli, ulaşılmak istenen amaç ve veri setine uyum açısından birden fazla teknik içermektedir. Sınıflandırma modeli alt teknikleri Tablo 9'da gösterilmektedir. Bunlardan biri olan Yapay Sinir Ağları (Artificial Neural Networks- YSA) tekniği, Mcculloch ve Pitts'in 1943 yılında yayınladıkları makale ile tanınmıştır (Mcculloch & Pitts, 1943). YSA, John Von Neumann, Marvin Minsky, Kohonen, Grossberg ve Hopfield gibi uzmanlar tarafından geliştirilmiştir. İlk YSA sempozyumu 1987 yılında yapılmış ve bu sempozyumdan sonra uzmanlar tarafından uygulamalarda kullanımı artmıştır (Erdoğan Ş. Z., 2004, s. 25). YSA'nın veri yapısı Şekil 10'da gösterilmektedir. Teknik anlamda bir yapay sinir ağının asıl amacı girdi olarak tanımlanan bir veri seti için bir çıktı oluşturabilmektir. Yapay sinir ağının bu işlevi yerine getirebilmesi için ağa benzer durumun örnekleri öğretilerek genelleme yeteneği kazandırılır.

Bu yetenek sayesinde benzer durumlar için çıktı oluşturulabilmektedir. Ayrıca, YSA literatürde "bağıntılı ağlar (connectionist networks)", paralel dağıtılmış ağlar (paralell distributed networks)", "nuromorfik sistemler (neuromorfic systems)" adları ile de kullanılmaktadır (Öztemel, 2006, s. 30).



Şekil 10: YSA Yapısı

Kaynak: ÖZTEMEL, E. (2003). Yapay Sinir Ağları. 1.bs. İstanbul: Papatya Yayıncılık. s.33.

YSA, adaptif öğrenme, uyarlanabilirlik, doğrusal olmama, hata toleransı, genelleme ve donanım hızı gibi özelliklere sahiptir (Karaibrahimoğlu, 2014, s. 45). YSA, bellek tabanlı yöntemlere nazaran daha az boyutta bellek ve işlem ile çalışmaktadır. Ayrıca, istatistiksel tekniklerdeki gibi veriye dayalı bir parametrik model oluşturmamaktadır. Bu sebeple, çalışma alanı bu tekniklere göre daha geniştir (Döşlü, 2008, s. 21). YSA tekniğinin avantajları yanı sıra dezavantajları da olan bir tekniktir. Bütünüyle sistem içinde ne olduğu bilinemeyen YSA'nın diğer sistemlere uygulama aktarımı zordur. Ek olarak, sistem içindeki tüm ağlar için kararlılık analizinin yapılması mümkün olmamaktadır (Silahtaroğlu, 2008, s. 83-98) 1997 yılında HNC firması tarafından kredi kartlarına ilişkin yapılan bir YSA sistemi olan "Falcon", aynı yıl Caere INC. firması tarafından "Optik Karakter Okuma Sistemi" ve 1998 yılında Sensöry INC. firması tarafından ses okuma sistemleri için yapılan "Yonga" başarılı YSA uygulamalarıdır (Öztemel, 2006, s. 203).

Karar ağaçları (Decision Trees), geçmişte yaşanmış ve sınıfı belirlenmiş verilerden hareketle hangi sınıfa ait olduğu belli olmayan verilerin sınıflarının belirlenmesini sağlayan bir tekniktir. Teknik, veri madenciliğinde tahminleyici ve tanımlayıcı niteliklere sahip olması, uygulamasının ve değerlendirilmesinin kolay oluşu, güvenilirlik düzeylerinin yüksek olması ve bilgisayar sistemlerindeki depolama ünitelerinde kolay entegre edilebilmeleri sebebiyle sıklıkla

tercih edilmektedir. Karar ağaçları, doğrusal olmayan ilişkilerin, karma veri türlerinin ve aykırı değerlerin olduğu veri setlerinde sınıflandırma modelinin çıkarılmasına imkân verir (Kurt, Tokatlı, & Türe, 2009, s. 2017). Karar ağaçları genel olarak veri setini kendisine tanınan komutlar aracılığıyla küçük parçalara ayırarak, model tasarımı ağaç yapısında olan bir tekniktir. Veri setinde her bir seviyedeki parçalanma sonucunda gruptaki birimler bir önceki grup birimlerine nazaran daha fazla ortak özelliklere sahip olmaktadır.

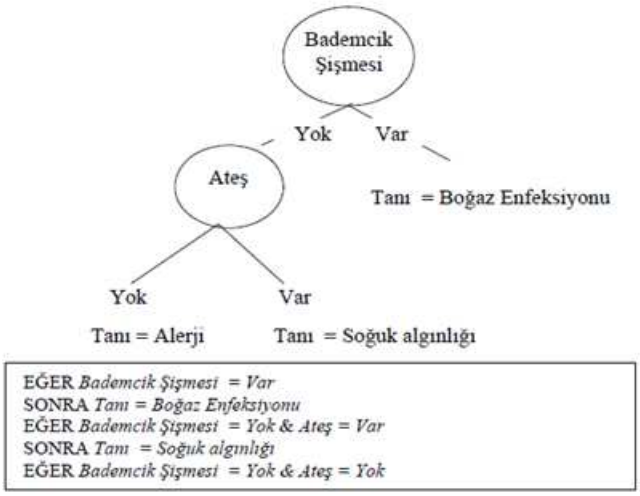
Karar ağacı uygulamasındaki ilk aşama bir ağaç oluşturmak, ardından veri setindeki verileri bu ağaca işlemektir. Ağaç yapısının oluşturulması için tekniğe özel algoritmalara başvurulmaktadır. Tablo 12'de bir klinikte hastalık tipinin teşhis edilmesi için hastalara ait nitel veriler bulunmaktadır.

Tablo 12: Bir Hasta Veritabanı

Sıra No	Boğaz Ağrısı	Ateş	Bademcik Şişmesi	Kan Toplaması	Baş Ağrısı	Tanı
1	Var	Var	Var	Var	Var	Boğaz Enfeksiyonu
2	Yok	Yok	Yok	Var	Var	Alerji
3	Var	Var	Yok	Var	Yok	Soğuk Algınlığı
4	Var	Yok	Var	Yok	Yok	Boğaz Enfeksiyonu
5	Yok	Var	Yok	Var	Yok	Soğuk Algınlığı
6	Yok	Yok	Yok	Var	Yok	Alerji
7	Yok	Yok	Var	Yok	Yok	Boğaz Enfeksiyonu
8	Var	Yok	Yok	Var	Var	Alerji
9	Yok	Var	Yok	Var	Var	Soğuk Algınlığı
10	Var	Var	Yok	Var	Var	Soğuk Algınlığı

Kaynak: EMEL, G.G. ve TAŞKIN, Ç. (2005). Veri Madenciliğinde Karar Ağaçları ve Bir Satış Analizi Uygulaması. Eskişehir Osmangazi Üniversitesi Sosyal Bilimler Dergisi. C.6, S.2, s.226-227., 221-239.

Tablo 12'de verilen örnek veri seti üzerinde karar alıcının seçimlerine dayanarak bir kök düğüm belirlenir. Burada kök düğüm olarak "Bademcik Şişmesi" değişkeni atanarak, teknik için en zor nokta geçilmiştir. Diğer kalan değişkenler düğüm olarak atanır ve bademcik şişmesi değişkeni altında diğer değişkenler sınıflara yani yapraklara ayrılmıştır. Bu ayrım, alt düğümlerde devam ederek ağaç yapısı tamamlanır. Sonrasında sisteme girilen her yeni kayıt buradaki yapraklara göre sınıflandırılır. Ağaç yapısı ile bu şekilde kurallar oluşturulmaktadır (Gökay & Taşkın, 2005, s. 226). Ağaç yapısı ile oluşan kurallar Şekil 11'de gösterilmektedir.



Şekil 11: Hasta Veritabanı İçin Bir Karar Ağacı ve Kurallar

Kaynak: EMEL, G.G. ve TAŞKIN, Ç. (2005). Veri Madenciliğinde Karar Ağaçları ve Bir Satış Analizi Uygulaması. Eskişehir Osmangazi Üniversitesi Sosyal Bilimler Dergisi. C.6, S.2, s.228., 221-239.

YSA ve diğer istatistiksel tekniklere göre karar ağaçlarında, veriden türetilen bir değerin bir kişinin yorumlayabilmesi daha kolaydır. Elde edilen sonuçların anlamlılığının denetlenmesi için alanında uzman kişilere danışılabilir. Çalışma sonrasında farklı bir yöntem uygulanacak olsa bile bir pivot çalışma olarak karar ağaçları, önemli değişkenler ve takriben geçerli olabilecek kurallar hakkında araştırmacıya bilgi verecektir (Alpaydın, 2000, s. 7). Genellikle karar ağaçlarına genellikle belirlenen bir grubun olası üyelerinin belirlenmesinde, çeşitli nitelikleri evet/hayır gibi kategorilere ayırmada, geleceğe yönelik tahminler sağlayacak kuralların çıkarılmasında, bazı alt kategorilere özgü olan bağıntıların tanımlanmasında, parametrik modeller için yararlı olacak veri seti ve değişken seçiminde ve kategorilerin birleştirilmesi ve sürekli değişkenlerin kesikli hale getirilmesi konularında başvurulmaktadır (Akpınar, 2000, s. 14-15).

Doğrusal olmayan matematiğin öncülerinden Holland, 1975 yılında Genetik algoritmalar konusunun yapı taşlarını ortaya koyan ilk bilim insanı olmuştur (Holland, 1975). Bu konudaki çalışmalarını "Adaptation in Natural and Artificial Systems" adlı kitabında toplayarak genetik algoritmalar konusunu uygulamaya başlamış ve uygulamalara Michigan Üniversitesi ev sahipliği yapmıştır (Goldberg, 1989, s. 1-2). YSA tekniğindeki gibi genetik algoritmalar

tekniğinde de insan beyni ve duyuları örnek alınmıştır. Genetik algoritmaları bir arama ve seçim algoritması olmakla beraber temeli doğal seçim ve genetiğin mantığına dayanır (Goldberg, 1989, s. 1) Olasılık kurallarını baz alarak çalıştırılan genetik algoritmalar, tüm mümkün çözümlerin tamamı yerine belli bir parçasında arama yaparak kısa vadede sonuca ulaşır (Watson, 2002, s. 435).

Bellek temelli (Memory-Based/ MBR) yöntemler istatistik biliminde 1950'li yıllarda önerilmiştir. Ancak, yetersiz bellek ve dar hesaplama şartları sebebiyle kullanılamamıştır. Daha sonra yaşanan teknolojik ilerlemeler sayesinde bellek seviyesindeki artış ve güçlü hesaplama olanakları ile uygulanmaya başlanmıştır (Döşlü, 2008, s. 21). MBR, çözülmesi istenen problemi önceki benzer problemlerin çözümünden elde edilen deneyim ve çözümleri mevcut durumdaki probleme uygulamaya çalışan bir veri madenciliği tekniğidir. Güvenlik sistemlerinde dolandırıcılık ve sahtekarlıkların belirlenmesi ve işletme tarafından müşterilere yollanan mektup ve çağrılara geri dönüş tahmin edilmesinde tercih edilmektedir. Teknik için en iyi algoritma en yakın k komşu algoritması (Nearest Neighbour)'dır. Bu yöntem, verileri olduğu gibi kullanabilme yetisine sahiptir.

Bulanık kümeler yaklaşımı, bulanık mantık esasına dayanmaktadır ve belirsizlik kavramı söz konusudur. Tam anlamıyla farklı formatlarda karışık ve belli olmayan yani kesinliği sağlanmayan bilgi kaynaklarına bulanık (fuzzy) denmektedir (Şen, 2001, s. 9). En önemli özellikleri konu anlamında çok fazla girdi ile eğer ve ise şeklinde oluşan bir kural tabanına sahip olması ve çıkarım motoru ile çalışarak tek bir çıktı elde ediyor olmasıdır. Bulanık mantık konusu sistemler mühendislik, tıp, sosyoloji, psikoloji, işletme, uzman sistemler, yapay zekâ, sinyal işlenmesi, ulaştırma, kavşak sinyalizasyonu, görüntü işleme, zaman serileri esaslı tahmin yapmak, kontrol sorunlarını çözmek ve haberleşme alanlarında uygulanmaktadır (Şen, 2001, s. 26).

Lojistik regresyon (Logistic Regression-LR) analizi, çok değişkenli istatistiksel verilerde sınıflandırma, bu tür veriler için yapılabilecek birtakım istatistiksel tekniklerde ön inceleme yapmaya ek olarak, özellikle sosyal araştırmalarda tek başına bir analiz olarak da tercih edilmektedir (Tatlıdil, 1992, s. 225). Analizde güdülen amaç, nitel açıklanan değişkenin değerini tahmin etmektir. Bu sebeple, burada iki veya ikiden fazla grup için "üyelik tahmini" yapılmaktadır. Dolayısıyla, analizde hem sınıflandırma hem de açıklanan ve açıklayan değişkenler arasındaki bağlantıları keşfetmek amaçları mevcuttur (Çokluk, 2010, s. 1362). LR yöntemi, diskriminant analizi ve çapraz tablolarda normallik ve ortak kovaryansa sahip olma

gibi varsayımlar sağlanamadığında ve doğrusal regresyon analizinde açıklanan değişkenin ikili (binary) ya da ikiden fazla düzeye sahip (polychotomous) kesikli değişken olması sebebiyle normallik varsayımı bozulduğunda alternatif yöntem olmaktadır. LR yöntemi, diğer yöntemlere göre varsayımlarda esnek olması, kolay ve anlaşılabilir olması ve analiz sonucunda elde edilen fonksiyonun matematiksel açıdan esnek olması sebepleri ile araştırmalarda sıklıkla tercih edilmektedir (Tatlıdil, 1992, s. 225).

Bayes teoremine dayanan Bayes sınıflandırması (Bayesian Classification), söz konusu yeni bir niteliğin daha önceden geçmiş verilere dayanılarak oluşturulan sınıflara girme olasılığını hesaplamaktadır. Bu teorem; belirsizlik taşıyan bir olay için model oluşturulması, bu olaya dair gerçekçi gözlemler ve evrensel doğrulara dayanarak bazı sonuçlara ulaşılmasına imkân tanır. Bu sebeple, belirsizlik söz konusu olan olaylarda karar vermeye yardımcı olmaktadır. Yöntemin en belirgin zayıflığı, değişkenler arasında olan ilişkinin bir model olarak sunulamıyor olması ve tüm değişkenlerin birbirinden bağımsız olduğu varsayımıdır (Argüden & Erşahin, 2008, s. 52).

Vladimir Vapnik ve Alexey Chervonenkis tarafından Karar Destek Vektör Makinesi (Support Vector Machine-SVM), 1960'lı yıllarda geliştirilmiştir. Parametrik olmayan bir makine öğrenmesi tekniğidir. Geleneksel istatistiksel yöntemlerdeki gibi yüksek boyutlu veri setlerinin doğrusal olarak ayrılamaması büyük bir problemdir. SVM tekniği iki sınıflı problemlerin çözülmesinde doğrusal sınıflandırma tekniği olarak geliştirilmiştir. Bu teknik daha sonra doğrusal ayrılmanın mümkün olmadığı problemler ile çok sınıflı sınıflandırma problemlerine uyarlanmıştır. SVM, öğrenme olayının teori ve pratiğinin bir arada olduğu bir tekniktir. Sınıflandırma ve regresyon modellerinde bayesyen yaklaşıma dayanan ve daha hassas çıktılar veren geçerli vektör makinesi (Relevance Vector Machine-RVM) adlı bir teknik mevcuttur (Karaibrahimoğlu, 2014, s. 43).

1.9.2. Regresyon

Regresyon, iki ya da daha çok değişken arasındaki ilişkinin fonksiyonel halini veren bir tahmin edici veri madenciliği tekniğidir. Yani, değişkenler arasındaki ilişkinin matematiksel bir ifadesidir. Finans alanında ve zaman serisi analizinde tahmin üretme, biyomedikal ve ilaç tepkimelerinin değerlendirilmesi, atmosferdeki CO2 oranının hesaplanması ve konut fiyatlarının analiz edilmesi gibi konularda regresyon analizine başvurulabilir (Argüden &

Erşahin, 2008, s. 38). Sınıflandırma modelinde olduğu gibi regresyon modelinin de alt teknikleri mevcuttur. Bu alt teknikler Tablo 9'da gösterilmektedir.

Sürekli değerleri tahmin etmek amacını taşıyan regresyon modellerinde, girdiler ile çıktılar arası bağ kurabilecek bir fonksiyon oluşturup, tahmin başarısı eniyilemeye çalışılmaktadır. Çıktı "açıklanan", girdi "açıklayan" değişken olarak isimlendirilir. Açıklanan değişkenin alacağı değer belli bir güven aralığı içinde ifade edilir (Argüden & Erşahin, 2008, s. 38). Aşağıdaki modelde, "Y" ifadesi modelin açıklanan değişkenini (bağımlı veya etkilenen), "x" ifadesi ise modelin açıklayan değişkenini (bağımsız veya etkileyen) vermektedir. Determinist bir yapıda β ifadesi sabit kalır ve $\Delta(Y)/\Delta(x)$ değerini verir.

$$Y = f(x) \tag{1.1}$$

Gözlemlerin stokastik bir yönünün olması sebebiyle, modele hata terimini eklenir ve model determinist yapıdan kurtulur. Stokastik bir yapıda, β ifadesi, x bağımsız değişkenindeki bir birimlik değişime karşılık Y bağımlı değişkeninin alacağı ortalama değeri vermektedir. Bu iki yapı arasındaki fark regresyon katsayısı üzerinde kendini göstermektedir.

Değişkenler arası ilişkiyi doğrusal varsayarak, bir bağımsız değişkene sahip modellere basit doğrusal regresyon modeli, iki veya ikiden fazla bağımsız değişkene sahip modellere çoklu doğrusal regresyon modeli adları verilmektedir. Bu modeller sayısal veriler üzerinde çalışmaktadır. Veri seti veya açıklanan değişken nitel verilerden oluşmakta ise yine LR yöntemine başvurulmaktadır.

En küçük kareler (EKK) yöntemine dayanarak elde edilen bir basit doğrusal regresyon modeli ve parametreleri;

$$Y = \alpha + \beta(x) + \varepsilon \tag{1.2}$$

α : Sabit (Otonom) değer,

β : Regresyon katsayısı (Y bağımlı değişkeninin x bağımsız değişkeni arasındaki ilişkiye ait doğrunun eğimi) ve

ε : Hata terimidir.

EKK yöntemine dayanarak elde edilen bir çoklu doğrusal regresyon modeli ve parametreleri;

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_m x_{im} + \varepsilon_i \tag{1.3}$$

β_0 : Sabit (Otonom) değer,

$\beta_1, \beta_2, \ldots, \beta_{im}$: regresyon katsayıları ve

ε : Hata terimidir.

Sınıflandırma ve regresyon tahmin edici veri madenciliği modelleri olması sebebi ile ortak amaçlar için kullanılmaktadır. Ancak, her iki modelde de elde edilen fonksiyonlarda açıklanan değişkenin nitel veya nicel olması yöntemler arası farka sebep olmaktadır. Bir sınıflandırma modeli bir hastalık türü için başvurulan ilaçların yan etkilerini gruplandırmak amacıyla oluşturulurken; bir regresyon modeli, bina yaşı, bulunduğu kent bilgisi ve oda sayısı gibi bilgiler ile evlerin kira fiyatını tahmin etmek için oluşturulabilir. Ek olarak, bu iki modelin bir arada kullanıldığı analizlerde yapılmaktadır. Çok terimli lojistik regresyon analizinde nitel değerler tahmin edilerek, bu iki model birbirine yaklaşmakta ve beraber kullanılabilmektedir (Akpınar, 2000, s. 6).

1.9.3. Kümeleme Analizi

Kümeleme analizi (Clustering Analysis), veriden uzaklık ölçüleri aracılığıyla, birbiri ile benzerlik gösteren birimleri aynı kümelerde bir araya getirerek, kümeler arasında heterojen bir yapı oluşturma şeklinde gerçekleşmektedir. "Mimar Sinan Üniversitesi Güzel Sanatlar fakültesi resim bölümü öğrencileri sosyo-ekonomik yönden kaç gruba ayrılır?" problemi kümeleme analizine uygun bir örnek olacaktır. Bu analizdeki en önemli nokta veri setinden aykırı ve gürültü verileri çıkarmaktır. Bu işlem, seçilen algoritmanın başarılı sonuçlar vermesini sağlayacaktır.

Kategorik, sürekli ya da karma veri setleri için uygun kümeleme tekniğinin seçilmesi suretiyle veri indirgeme imkânı sağlayan bu analizde, öncelikle orijinal veri setinden benzerlik/uzaklık matrisi elde edilir. Kümeleme işlemi, bu benzerlik/uzaklık matrisler, baz alınarak gerçekleştirilir (Yıldırım, 2015, s. 163). Kümeleme analizinde veri türüne göre tercih edilen uzaklık matrisleri Tablo 13'te gösterilmektedir. Benzerlik matrisleri Jaccard, Dıce, Overlap ve Cosne'dir. Matrisler aracılığıyla veri setindeki her bir gözlemin diğer gözlemlere olan benzerliği/uzaklığı ile aralarındaki ilişki değerlendirilir.

Tablo 13: Kümeleme Analizinde Uzaklık Matrisleri

Uzaklık Matrisleri				
Karma Veri	Bhattachoryya	Metrik Veri	Öklid Minkoswki Manhatten Mahalanobis D^2	Kareli Öklid Canberra City-Black Hotteling-T^2

Kaynak: Yazar tarafından derlenmiştir.

Kümeleme analizinde yararlanılan veri setindeki her bir veri sadece bir kümeye ait olmaktadır. Bir veri seti için kümeleme analizinde araştırma probleminin belirlenmesi sonrasında araştırma planını oluşturulmaktadır. Bu noktada araştırmanın nasıl şekilleneceği ve benzerlik/uzaklık matrislerinin seçimine karar verilir. Matris seçimi veri tipine göre şekil almaktadır. Bu durum, analizin veriye dayalı olarak geliştiği anlaşılmaktadır. Matris seçiminden sonra analizin yapılabilmesi için varsayımların incelenmesine geçilir. Ancak, kümeleme analizinin en çok tercih edilme sebeplerinden biri çoklu normallik ve homojenlik gibi varsayımlarının olmamasıdır. Bu özelliği sebebiyle diskriminant analizinde varsayımların sağlanmaması noktasında alternatif bir yöntem olmaktadır. Analize başlanmadan önce sadece ölçüm hatasının olup olmadığına ve örneklemin anakitleyi temsil gücüne bakılmaktadır. Varsayımlar sağlandıktan sonra gruplandırma seçimleri yapılmaktadır. Gruplandırma seçiminde hem değişken hem de gözlem seçeneklerinin olması ile kümeleme analizi diğer yöntemlere göre büyük bir avantaj taşımaktadır. Analiz öncesinde veri setinin kaç kümeye ayrılabileceği ya da küme işleminin hangi faktörlerin niteliklerine göre yapılabileceği bilgisi bilinmemektedir. Bu konu, uzmanlara danışılarak netleştirilebilir. Küme sayısının bilinip bilinmemesi durumu kümeleme analizi ile sınıflandırma modeli arasındaki temel farktır. Yani, kümeleme modeli mevcut veri setinden hareketle oluşturulmaktadır ve önceden kurulmuş bir model üzerinden uygulanmamaktadır. Yani, kümeleme analizi tamamlandıktan sonra veri setinde yeni bir birim gündeme geldiğinde bu birimin hangi küme de yer alması gerektiğini anlamak için veri seti tekrar analize tabi tutulmaktadır. Sınıflandırma modelinde ise önceden tanımlanmış sınıfların aldıkları değerlere göre yeni veriler sınıflara dağıtılmaktadır. Yani, kümeleme analizinde sınıflandırma modelinde olduğu gibi bir fonksiyon oluşmamakla beraber veriye dayalı bir analiz türüdür. Dolayısıyla, kümeleme analizi statik bir yapıya sahip ileriye yönelik analiz türüdür.

Küme oluşmasında değişken seçimi etkin bir rol almaktadır. Şöyle ki; dönüştürülen bir değişkenin dönüşümden önceki halinin analizde yer alması analiz sonuçlarının doğruluğunu tehdit etmektedir. Değişkenlerin kaç kümede toplanabileceği konusunda ön bilgi olup olmamasına göre kümeleme teknikleri (grafik yöntemi, hiyerarşik, hiyerarşik olmayan ve two steps) arasında tercih yapılır (Yıldırım, 2015, s. 163). Tablo 14'te hiyerarşik ve hiyerarşik olmayan kümeleme yöntemleri detaylı olarak gösterilmektedir. Hiyerarşik ve hiyerarşik olmayan kümeleme tekniklerinin seçiminde küme sayısının bilinip bilinmemesine göre hareket edilmektedir. Küme sayısının bilinmediği durumlarda hiyerarşik kümeleme tekniği ve küme

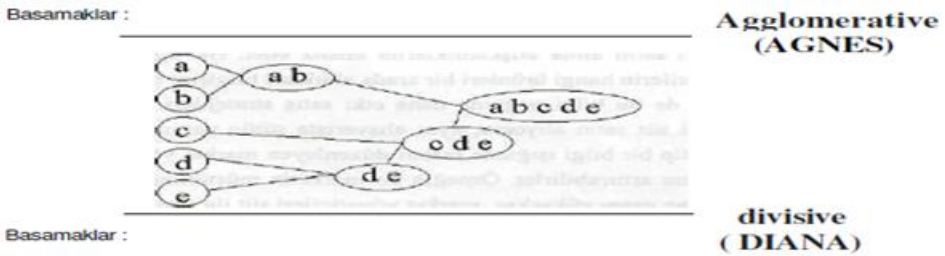
sayısının bilindiği durumlarda hiyerarşik olmayan kümeleme tekniği kullanılmaktadır. Bu yöntemlerin uygulama süreci ve kullanıldıkları durumlar birbirine göre farklıdır.

Tablo 14: Hiyerarşik ve Hiyerarşik Olmayan Kümeleme Teknikleri

Hiyerarşik Yöntemler	Hiyerarşik Olmayan Yöntemler
Ayırıcı Kümeleme Teknikleri Birleştirici Kümeleme Teknikleri • Tek Bağlantı • Tam Bağlantı • Ortalama Bağlantı	Bölümlemeli Yöntemler Yoğunluğa Dayalı Yöntemler Izgara Tabanlı Yöntemler Olasılıksal Algoritmalar Kategorik Verinin Yinelenmesine Dayalı Yöntemler Model Tabanlı Yöntemler Yer Değiştirmeli Algoritmalar Kısıtlara Dayanan Yöntemler Makine Öğrenmesi Alanında Kullanılan Yöntemler

Kaynak: Yazar tarafından derlenmiştir.

Grafik yöntemi ikiden fazla değişkene sahip bir veri setinde kullanılamamaktadır. Kullanımı söz konusu ise faktör analizi yapılabilir. Hiyerarşik kümeleme yöntemleri Birleştirici Kümeleme Teknikleri (Agglomenative Nesting/ AGNES) ve Ayırıcı Kümeleme Teknikleri (Divisie Analysis/ DIANA) olmak üzere ikiye ayrılmaktadır. Şekil 12'de görüleceği üzere bu iki yöntem arasındaki fark ayrışma noktasının aşağıdaki kümelerden mi yoksa yukarıdaki kümelerden başlamasıdır. Hiyerarşik kümeleme tekniklerinde genel olarak başlangıçta veri setindeki her bir gözlem küme olarak kabul edilmektedir. Daha sonra benzerlik/uzaklık matrisi sonucu yakalanan benzerliklere göre gözlemler birleşir ve oluşacak son kümeye kadar işlem devam etmektedir. Küme sayısını belirlemek için ağaç diyagramı, diğer bir adıyla dendogram grafiği oluşturulur. Analiz sonuçlarında uç değer ve çok küçük gözleme sahip küme olup olmadığına bakılmaktadır. Oluşturulan kümeler yorumlanıp, geçerliliği sınanarak isimlendirilir. Bu şekilde uygulama son bulmaktadır.



Şekil 12: Hiyerarşik Kümeleme Örneği

Kaynak: ÖZEKES, S. (2003). Veri Madenciliği Modelleri ve Uygulama Alanları. İstanbul Ticaret Üniversitesi Dergisi. C.2, S.3, s.75., 66-8.

Bölümlemeli Yöntemler (Partitioning Methods)'de en çok tercih edilen yöntemlerin başında k-means ve k-medoids yöntemleri gelmektedir. 1956 yılında Hugo Steinhaus tarafından ileri sürülen algoritma, diğer algoritmalara göre avantajı çoğu veri tipinde onaylanabilir çıktılar vermesi, dezavantajı bölgesel optimumlarda kalarak genel optimumlara yayılamamasıdır (Yünel, 2010, s. 2). Bu yöntem, veri setindeki birimlerden k tane birim seçerek işleme başlar. Bu k tane birim veri setindeki bir kümenin merkez noktasına en yakın noktayı temsil etmektedir. Veri setindeki diğer birimler, k tane noktadan en yakın olanın altında toplanmaktadır. k adet küme oluştuktan sonra bu kümelerin ortalaması alınır. Her bir ortalama hesaplandığı kümenin orta noktası olarak atanır ve bu işlem döngüsü veri setindeki tüm birimlerin tam anlamıyla dağıtılmasına kadar devam ederek, son bulmaktadır. Model tabanlı yöntemler (Model-based methods), veri setindeki birimleri birtakım matematiksel modellerde kullanabilmeyi hedeflemektedir. Esasen, olasılık dağılımları kullanılarak oluşan veri seti varsayımına dayanmaktadır. Genelde, sinir ağları ve sınıf ağaçları tercih edilen tekniklerdir (Mocan, 2016, s. 43). Izgara tabanlı yöntemler (Grid-based methods), çok büyük veri içeren veritabanlarında tercih edilmekte ve çok çözümlü grid veri yapısını kullanmaktadır. Prensip, ilk olarak veri setini ızgara görünümlü bir yapı üzerinde belli bir sayıda hücrelere bölünmesi biçimindedir. Tekniğin, esas avantajı genellikle birim sayısından bağımsız olarak hızlı işlem yapmasıdır (Mocan, 2016, s. 43). Yoğunluk temelli kümeleme yöntemleri (Density-based Methods) farklı görünümlerde küme yapıları elde edebilmek için geliştirilmiştir. Tüm kümelere ait noktaların bazı olasılık dağılımlarına dayandığını iddia eden bir tekniktir. Teknik, işlem kümelerinin geri kalan verilerin oluşturduğu kümelerden daha fazla yoğunluklu alanların tespiti

şeklinde gelişmektedir. Atanan yoğunluk bölümlerinin dışında kalan birimler uç nokta veya gürültü sayılmaktadır (Karaibrahimoğlu, 2014, s. 55-56).

Two Steps yöntemi, hiyerarşik ve hiyerarşik olmayan yöntemlerin bir karması olarak, akaike ve schwarz bilgi kriterlerine göre maximum küme sayısı belirlendikten sonra en büyük sıçrama görülen noktada küme sayısı olarak alınır. Belirlenen kümelere gözlemler atanarak uygulama tamamlanmaktadır.

Kümeleme analizi, genellikle çalışma öncesi uygulanarak karar verme sürecinde anahtar rol oynamaktadır. Bu durumun sebebi, çalışma sürecinde kümeleme analizi ile elde edilen kümelerin diğer yöntemlere girdi olarak sağlanmasıdır. Bu sayede değişkenler hakkında detaylı bilgi sağlanmış olmaktadır. Bu sebeple, sınıflandırma modeli öncesi kümeleme analizinin uygulanması sınıfların belirlenmesinde yararlı olabilir. Kümeleme analizinin uygulamalarda, Likert ölçek gibi kararsızlık noktasına da sahip olan ölçeklerde, kararsız deneklerin muhtemel reaksiyonlarını tahmin etmek amacıyla da kullanılabildiği belirtilmelidir. Özellikle, seçim anketlerinin değerlendirilmesinde, kararsız seçmenler için yapılacak atamalarda Kümeleme Analizinden yararlanılabilmektedir (Yıldırım, 2015, s. 163-164). Bunlara ek olarak, dijital ortamda dokümanların gruplandırılması, biyolojide gerçek türlerin belirlenmesi, model uydurmanın kolaylaştırılması, gruplar hakkında ön tahmin (18-24 yaş grubundaki üniversitelerin siyasete ilgisinin artması gibi), hipotez testi (her grubun merkezi noktasını belirlemek), veri yapısının kesinleştirilmesi, veri indirgeme, uç değerlerin bulunması (grafik veya frekans tablosu ile), belirli özelliklere göre şehirlerin coğrafik konumlara göre sınıflandırılması, en iyi müşterilerin kimler olduğu, farklı müşteri gruplarının neler olduğu ve bu grupların alışveriş örüntüleri hakkında bilgi, müşterilerin satın alma davranışlarının neler olduğu, hangi gelir grubunun hangi markayı satın almayı tercih ettiği bilgisi, müşteri sadakat derecesi ve bunun öngörülebilirliği (Alagöz, Ortakarpuz, & Öge, 2014, s. 16) gibi amaçlar için tercih edilmektedir.

1.9.4. Ardışık Zamanlı Örüntüler

Ardışık zamanlı örüntüler, veritabanında belli zaman aralıklarına göre gelişen ve aralarında bağlantı bulunan durumların ortaya çıkarılmasında kullanılmaktadır. Yani, belli aralıklarla gelişen iki veya daha çok durum arasında bağlantı olup olmadığı ardışık zamanlı örüntüler ile incelenmektedir. Modelde, öncelikle veritabanındaki bağlantılı birimler incelenir.

Veritabanında çok fazla gündeme gelen bağlantılı birimlere dair trendler belirlenir. Bu trend bilgileri aracılığıyla geçerli ilişkiler tanımlanmaktadır.

Kurum müşterilerinin alışveriş alışkanlıklarının belirlenmesi için pazarlama alanında ardışık zamanlı örüntüler modeline başvurulabilir. Örneğin, bir perakende mağazası, müşterilerinin gerçekleştirdiği işlemleri değer bazında ve davranış bazında ayrıştırıp analiz ederek birçok makro ve mikro parça oluşturulabilir. Her parçanın davranışları analiz edilebilir ve parçaya özel teklifler, kampanyalar oluşturulabilir. Bu çalışmalar ile müşteriye özel hizmet geliştirme ve yüksek müşteri memnuniyeti sağlanabilir (Gökmen, 2014, s. 49-50). Bu çalışmalara ek olarak; bir teknoloji mağazasından video-kamera alıcısının ilk üç ay içinde hangi araç ve gereçlerini almış olduğu bilgisinden hareketle önümüzdeki altı ay içinde neler satın alabileceği bilgisi, belirli bir hastalık tanısı ile ameliyat olan hastanın ilk bir yıllık süreçte ne tür komplikasyonlar yaşayacağı ve bu hastanın ilk birkaç yıl içinde hangi hastalıklara yakalandığı bilgisi (Yarımağan, 2000, s. 297) gibi konularda da "Ardışık Zamanlı Örüntüler" modelinden faydalanılabilir.

1.9.5. Birliktelik Kuralları

Agrawal tarafından 1993 yılında geliştirilen Birliktelik Kuralları (Association Rules), veritabanında yer alan birimlerin aynı anda ve sıklıkla gerçekleşme kıstaslarını dikkate alarak birimler arasındaki gizli ilişkileri keşfetmektedir. Birliktelik kuralları ile yapılan veri madenciliği çalışmasında en önemli olay kural çıkarmadır. Birliktelik kuralları modelinde diğer veri madenciliği modellerindeki gibi başlangıçta bir araştırma problemi bulunmamaktadır. Sadece araştırmacı veya araştırma ekibi tarafından belirlenen minimum destek değeri (Minimum Support Value/ MDD) ve minimum güven değeri (Minimum Confidence Value/ MGD) ile verilerden kurallar üretilir ve sonucunda geçerli ve güvenilir kurallara ulaşılmaya çalışılır. Bu noktada hızlı ve güvenilir örüntü tanıma işlemi için "Birliktelik Kuralları Algoritmalarına" başvurulmaktadır. Birliktelik kuralları çalışmalarında genellikle Apriori algoritması tercih edilmiştir. Birliktelik kuralları analizinde veriler arasındaki ilişkiler, IF-THEN ifadeleri ile aşağıdaki gibi gösterilmektedir (Albayrak & Koltan, Yılmaz, 2009, s. 38).

IF <bazı şartlar sağlanırsa> THEN <bazı niteliklerin değerlerini tahmin et> (Albayrak & Koltan, Yılmaz, 2009, s. 38).

Veri madenciliği modellerinden Birliktelik Kuralları hemen hemen her alanda uygulanabilirliği açısından sıklıkla tercih edilmektedir. En önemli uygulama alanları arasında; "Market Sepet Analizi (Market Basket Analysis)", "Çapraz-Pazarlama (Cross-Marketing)", "Promosyon Analizleri", "Katalog" ve "Yerleşim Düzeni Tasarımları" bulunmaktadır (Döşlü, 2008, s. 3). Market sepet analizinde müşterilerin satın aldığı malların analizi yapılır (Alpaydın, 2000, s. 9). Elde edilen kurallara göre market yöneticileri mağaza raf sistemini düzenleyebilir ve bu sayede satış yapma ihtimallerini arttırmış olmaktadırlar. Bunlara ek olarak, müşterilerin kişisel tercihleri, promosyon düzenlemeleri gibi tüketiciye yönelik aktiviteler daha bilinçli bir şekilde yapılabilmektedir. Bu yöntem, online sistemlere de kolaylıkla entegre edilerek, internet siteleri, yayınlar ve dokümanlar arasındaki benzerlikler keşfedilebilmektedir.

Market sepet analizi yöntemi; anlaşılabilir ve kolay çıktılar üretebilme, farklı boyuttaki veri setleri üzerinde uygulanabilme ve uygulanan hesaplamaların yapay sinir ağları, genetik algoritmalar gibi diğer yöntemlere göre daha kolay olması (Erdoğan Ş. Z., 2004, s. 20) gibi birtakım avantajlar taşımaktadır. Analizin taşıdığı avantajlara rağmen bünyesinde bazı dezavantajlar da barındırmaktadır. Bu dezavantajlar; kayıtların sayısı ve kombinasyon seçimine göre yapılacak hesaplamalar artmaktadır. En iyi sonuç, tüm ürünlerin kayıtlar arasında hemen hemen aynı frekansa sahip olması durumunda oluşmaktadır. Ancak, yöntemin kayıtlarda nadir rastlanan ürünleri es geçmesi en iyi sonucu yakalamaya engel olmaktadır. Analizde MDD, MGD, destek ve güven değerleri aracılığıyla kurallar oluşturulmaktadır. Eşik değerlerinin yani MDD ve MGD değerleri sayesinde kural sayısına sınırlama getirilebilir. Ancak, eşik değerlerinin çok düşük seviyelerde tutulması, araştırmacının ilgilenmekte olduğu kuralları kaybetmesine yol açabilmektedir (Erdoğan Ş. Z., 2004, s. 20).

1.10. Veri Madenciliğinde Başvurulan Algoritmalar

İlk bilgisayarın icadı ve veri madenciliğinin ortaya çıkması ile ilerleyen zamanlarda algoritmalara ihtiyaç duyulmuştur. Uzmanların, özellikle veri madenciliği ve istatistiksel çalışmalarda karmaşık ve zor hesaplamaları yapmaları için bilgisayarlara gereksinim duyduğu gibi, bilgisayarlarda bu hesaplama ve analizlere temel sağlayacak algoritmalara da gereksinim duyulmuştur. Algoritma, mevcut bir problemi çözümlemek ya da belli bir amaç doğrultusunda belli bir düzende mantıksal adımlar içeren bir çözüm yoludur. Yani, bir problemi çözmek için uygulanacak adımların tamamı algoritmayı oluşturmaktadır.

Veri madenciliği çalışmalarında algoritmaların temel kullanım amacı milyonlarca veri yığını arasından anlamlı örüntülere ulaşmayı sağlayacak kuralları oluşturmasıdır. Veri madenciliği modellerinde kullanılan algoritmalar birçok alanda tercih edilmektedir. Örneğin, ABD hükümeti tarafından veri madenciliği algoritmaları kullanılarak gizli dinleme ve vergi kaçakçılıkların deşifre edilmesi gibi birçok çalışmada tercih edilmektedir (Akpınar, 2000, s. 4). Veri madenciliği modelleri için algoritma seçimi başta veri madenciliği tekniği olmak üzere bazı kriterlere bağlıdır. Bazı algoritmalar yalnızca nicel veriler üzerinde bazıları ise yalnızca nitel veriler üzerinde çalışmaktadır. Bazıları ise 0 ve l değerlerinden oluşan veri setleri üzerinde çalışmaktadır (Köse, 2015, s. 69). Bir YSA çalışmasında nitel değişkenin var-yok şeklinde değerlendirilmesi ya da bir karar ağaçları çalışmasında nicel bir değişkenin kategorilere ayrılması modelin performansını arttırmaktadır (Akpınar, 2000, s. 10) Bu örnek durumlardan hareketle, algoritmaların yöntemlere göre avantajları ve dezavantajları mevcuttur. Bu sebeple, algoritmaların seçimi model başarısını önemli ölçüde etkilemektedir.

Sınıflandırma modelinde uygulanan tekniklerin özelliklerine göre algoritmalar geliştirilmiştir. Bir karar ağacı uygulamasında algoritma seçimi kök, düğüm ve dallanma seçeneklerinin seçimine yani ağaç yapısının oluşumuna bağlıdır. Karar ağacı tekniği için geliştirilen bazı algoritmalara ilişkin özellikler Tablo 15'te gösterilmektedir. 1970'li yılların başlarında Morgan ve Sonquist adlı uzmanlar tarafından ileri sürülen Automatic Interaction Detector Algorithm (AID) algoritması, karar ağacı tabanlı hem ilk algoritma hem de ilk yazılımdır (Gökay & Taşkın, 2005, s. 228). G. V. Kass tarafından 1980'de geliştirilen Chi-Squared Automatic Interaction Detector (CHAID) algoritması istatistik tabanlı bir algoritmadır. Dallanma kriterinde hedef (bağlı) değişken nitel verilerden oluşuyorsa ki-kare testine, sürekli verilerden oluşuyorsa F testine başvuru yapmaktadır (Oğuzlar, 2004, s. 81). CHAID algoritması, hedef (bağlı) değişkene göre istatistiksel anlamda homojen sayılabilecek değerleri bir araya toplamaktadır. Diğer kalan değerleri ise heterojen olarak kabul etmektedir. Sonrasında karar ağacında oluşan ilk dal yapısına göre en uygun ön kestirici değişken belirlenir. Her bir düğüm noktası belirlenen değişkene ait homojen değerlerden bir grup oluşturur. Bu döngü, sürekli devam ederek ağaç büyütülür (Oğuzlar, 2004, s. 81). 1984 yılında Breiman ve ark. tarafından çalışmalarında bir karar ağacı algoritması olan Classification And Regression Trees (CART) algoritması kullanılarak, literatüre kazandırılmıştır (Breiman, Friedman, Olshen, & Stone, 1987). CART algoritması bazı kaynaklarda C&RT olarak ifade edilir. Bu algoritma entropiye dayalı olup, dallanma kriterinin hesaplanmasında Twoing ve Gini tekniklerinden

faydalanmaktadır. Nicel ve nitel verilerde çalışabilen CART algoritmasındaki temel nokta, karar noktalarında ikili seçim ile birimlerin homojen sınıflar oluşacak şekilde ayrılmasıdır (Oğuzlar, 2004, s. 82). J. Ross Quinlan tarafından 1986 yılında ID3 adlı bir karar ağacı algoritmasını geliştirmiştir. Quinlon yayınladığı kitabı ile "ID3 algoritmasının ileri bir versiyonu olan C4.5 karar ağacı algoritmasını literatüre kazandırmıştır (Quinlan, 1993). J. Ross Quinlan, ID3, C4.5 ve C5 karar ağacı algoritmalarını makine öğrenmesi ve bilişim kuramına göre geliştirmiştir. Bu algoritmaların temeli, bir sistemdeki belirsizliğin değeri olarak tanımlanan entropiye dayanmaktadır. ID3 algoritması geliştirilerek, C4.5 ve daha sonra C4.5 algoritması geliştirilerek, C5 algoritması elde edilmiştir. Bu yüzden bu iki algoritma aynı sonucu vermektedir. Aralarındaki tek fark, C5 algoritmasının daha hızlı olması ve uygulama sonucunda şekil açısından daha özenli karar ağaçları sunabilmesidir (Oğuzlar, 2004, s. 81).

Tablo 15: Bazı Karar Ağacı Algoritmaları Özellikleri

Algoritma	Özellikler
C&RT	Gini'ye dayalı ikili bölme işlemi mevcuttur. Son veya uç olmayan her bir düğümde iki adet dal bulunmaktadır. Budama işlemi ağacın karmaşıklık ölçüsüne dayanır. Sınıflandırma ve regresyonu destekleyici bir yapıdadır. Sürekli ve kategorik hedef değişkenler ile çalışır. Verinin hazırlanmasına gereksinim duyar.
C4.5 ve C5.0	Her düğümden çıkan çoklu dallar ile ağaç oluşturur. Dalların sayısı tahmin edicinin kategori sayısına eşittir. Tek bir sınıflayıcı da birden çok karar ağacını birleştirir. Ayırma işlemi için bilgi kazancı kullanır. Budama işlemi her yapraktaki hata oranına dayanır.
CHAID	Ki-kare testleri kullanarak bölme işlemini gerçekleştirir. Dalların sayısı iki ile tahmin edicinin kategori sayısı arasında değişir.
SLIQ	Hızlı ölçeklenebilir bir sınıflayıcıdır. Hızlı ağaç budama algoritması mevcuttur.
SPRINT	Büyük veri kümeleri için idealdir. Bölme işlemi tek bir niteliğin değerine dayanır. Tüm bellek sınırlamaları üzerinde nitelik listesi veri yapısı kullanarak işlem yapar.

Kaynak: EMEL, G.G. ve TAŞKIN, Ç. (2005). Veri Madenciliğinde Karar Ağaçları ve Bir Satış Analizi Uygulaması. Eskişehir Osmangazi Üniversitesi Sosyal Bilimler Dergisi. C.6, S.2, s.230., 221-239.

1997 yılında Loh ve Shih uzmanları tarafından geliştirilen Quick, Unbiased, Efficient Statistical Tree (QUEST) algoritması, dallanma sürecinde optimum bölünmeyi sağlayacak değişkene ve optimum bölünme sağlanacak noktanın karar verilmesine ayrı zamanlar ayırmaktadır. Bu yönüyle CART ve CHAID algoritmalarından ayrılmakta ve sınıflandırma modellerinde daha hızlı sonuç sağlamaktadır. Regresyon modellerinde bir karar ağacı uygulamasında açıklanan değişkenin sürekli olması durumunda bu algoritmadan yararlanılamamaktadır. Mehta, Agrawal ve Rissanen uzmanlarının 1996 yılında geliştirdiği Supervised Learning In Quest (SLIQ)

algoritması, nitel ve nicel veri tiplerinde kullanılabilmektedir. Bu algoritma, dallanma kriterinin hesaplanmasında "Gini tekniği" nden faydalanmaktadır. Scalable Parallelizable Induction of DecisionTrees (SPRINT) algoritması, entropiye dayanmaktadır. Bu algoritma, ağaç yapısında optimum dallanmayı sağlayabilmek için her bir değişkene ait verileri bir kez yapmak suretiyle sıraya dizmekte ve ağaç yapısını bu şekilde oluşturmaktadır. Exhaustive CHAID ve Multivariate Adaptive Regression Splines (MARS) karar ağacı tekniği için geliştirilen diğer algoritmalardır (Gökay & Taşkın, 2005, s. 221-228-229).

Bellek tabanlı nedenleme tekniğinde en optimal sonuçları veren en yakın k komşu algoritmasında (Nearest Neighbour) öklid uzaklığı kullanılmaktadır. Algoritma, hangi gruba ait olduğu bilinmeyen örneklemi, grubu belli olan örneklemlerin olduğu bir veritabanında araştırmaktadır. Öklid uzaklığı ile bilinmeyen örnekleme k en yakın örneklemleri bulmaktadır. Bu örneklem, k en yakın örneklem içinden en çok benzediği gruba atanır.

Kümeleme algoritmaları analizin genel amacına göre küme içindeki birimler arasındaki ilişkinin en çok kümeler arasındaki ilişkinin en az olmasını sağlamaktadır. Tablo 16'da hiyerarşik ve hiyerarşik olmayan kümeleme tekniklerinde kullanılan algoritmalar gösterilmektedir.

Tablo 16: Hiyerarşik ve Hiyerarşik Olmayan Kümeleme Teknikleri İçin Algoritmalar

Hiyerarşik Yöntemler	**Hiyerarşik Olmayan Yöntemler**		
	Bölme Yöntemleri	Izgara Tabanlı Yöntemler	Yoğunluğa Dayalı Yöntemler
SLINK CURE CHAMELEON BIRC	PAM CLARA CLARANS	CLINQUE WAVECLUSTER STING	DBSCAN OPTICS DENCLUE OTOSINIF

Kaynak: Yazar tarafından derlenmiştir.

Temsilciler Etrafında Bölümleme (Partitioning Around Medoids/ PAM) algoritması, 1990 yılında Kaufman ve Rousseuw uzmanları tarafından ortaya konulmuştur. Algoritma, k tane küme oluşturmak için medoid adı verilen temsilciler belirlemektedir. Medoid, bulunduğu kümenin merkezine en yakın noktayı temsil etmekte ve rasgele belirlenmektedir. Başlangıçta veri seti tek bir küme kabul edilerek buradaki birimleri temsilci olarak atadığı noktalar altında bir araya getirmektedir. Her bir döngüde temsilci olarak atanan noktalar değiştirilerek kümeleme işlemi son bulur. Geniş Uygulamaların Kümelenmesi (Clustering Large

Applications/ CLARA) algoritması, 1990 yılında Kaufman ve Rousseuw uzmanları tarafından ortaya konulmuştur. Algoritma, temsilci noktaları belirlemek için bütün veritabanını taramayı tercih etmemektedir. Bu işlem yerine rastgele bir küme belirlenir. Bu küme üzerinde PAM algoritması çalıştırılır. Bu işlem sonrasında, elde edilen kümeler için birer temsilci atanır. Ana küme olan veritabanından bir örneklem seçilir. Tekrar küme temsilcileri belirlenmeden önceki temsilcilerden yararlanılır. Bu tercih, algoritmayı hızlandırarak, kaliteli çıktılar oluşturacaktır (Karaibrahimoğlu, 2014, s. 54). Rasgele Aramaya Dayalı Geniş Uygulamaları Kümeleme (Clustering Large Applications based on Randomized Search/ CLARANS) algoritması, veri setinin çok büyük olduğu durumlar için PAM ve CLARA algoritmalarının geliştirilmiş ve karma halidir. CLARANS, *n* adet temsilciler ve bir şebeke diyagramından faydalanarak *k* adet küme oluşturmaktadır (Karaibrahimoğlu, 2014, s. 55). DBSCAN (Density Based Spatial Clustering of Applications with Noise) algoritması 1996 yılında Martin Esler, Peter Kriegel, Jörg Sander ve Xiaowei Xu tarafından ortaya çıkarılmıştır. Veritabanındaki her bir birimin komşusunu belirler ve en az sayıda birime sahip olup olmadığını incelemektedir (Mocan, 2016, s. 42-43). Daha sonra bu algoritma geliştirilerek, OPTICS algoritması oluşturulmuştur. Density Based Cluestering (DENCLUE) algoritması, veritabanında yer alan noktaların etki fonksiyonları toplamı aracılığıyla üretilen genel bir yoğunluk fonksiyonunun, lokal maksimumlarından faydalanarak yoğunluğa dayalı bir kümeleme süreci oluşturmaktadır (Karaibrahimoğlu, 2014, s. 55-56). Bir bayesgil kümeleme yöntemi olan AutoClass (OTOSINIF), Gaussian, Bernoulli, Poisson, ve log-normal dağılımları içeren bir algoritmadır.

Birliktelik kuralları yöntemine genel bir bakış açısıyla yaklaşmak gerekirse, aynı anda ve sıklıkla gerçekleşen ögeler arasında anlamlı bağıntılar olup olmadığı ve anlamlı bağıntılar varsa, bu bağıntıları kurallar aracılığıyla en iyi şekilde sunmaktır. Birliktelik kuralları algoritmaları, yüksek boyutlu bir veritabanında var olan değişkenler arasında pozitif, negatif ve farklı türlerde ilişkileri elde etmeye çalışmaktadır. Bu mantık çerçevesinde farklı ihtiyaçlara cevap verecek nitelikte birçok "Birliktelik Kuralları Algoritması" geliştirilmiştir. Algoritma seçimi verilerin yapısına bağlıdır. Birliktelik Kuralları modelinde algoritmalar ardışık ve paralel algoritmalar olmak üzere iki ana başlık altında incelenmektedir. Veritabanlarında genel olarak ögelerin isimleri ön koşul olarak ögekümeler alfabetik sıra ile kayıt altında tutulmaktadır. Bu depolama şekli ile ögekümelerin veritabanında üretilirken ve hesaplanırken sistemde kolaylık sağlamaktadır. Ardışık algoritmalar için bu teknik klasik bir yaklaşımdır. Algoritma içerisinde

bu ögekümeleri oluşturacak ve sıralamayı sağlayacak mantıksal sorgular bulunmaktadır. Birliktelik kuralları için bir diğer algoritma grubu olan paralel algoritmalar geniş ögekümelerinin üretilmesi görevinin paralelleştirilmesi mantığına dayanmaktadır. Apriori algoritması mantığına dayanan paralel algoritmalar ile geliştirilen algoritmalarda paralellik sisteminin getirilmesi ile verimliliğin arttırmak istenilmiştir (Döşlü, 2008, s. 31). Birliktelik kuralları algoritmaları detaylı olarak 2. Bölümde anlatılacaktır.

2. BİRLİKTELİK KURALLARI

Dünyada faaliyette bulunan tüm kuruluşlar bünyelerinin altında yaşanan gelişmeleri geçmişten bugüne kayıt altına almaktadırlar. 90'lı yılların öncesinde bu kayıtların çoğunluğu dar kapsamlı mali verilerden oluşmak ile beraber kayıtların tamamına istenildiği anda ulaşılamamaktaydı. Ancak, teknoloji nimetlerinden bilgisayar, ilişkisel veritabanları, veri ambarı, barkod okuyucular ve müşteri kartları gibi araçlar sayesinde veri içeriklerinde hem detay bilgiye inme hem de bilgiye ulaşmak için zaman tasarrufu avantajları sağlandı. Bu sayede, kuruluş bir müşterisinin hangi ürünü, ne zaman, kaç adedi ne tutarda aldığı ve ilgili ürünü ne sıklıkla aldığı gibi bilgilere ulaşabilmektedir. Bu detaylı verileri düzenli ve sağlıklı olarak kayıt altına almak için uzmanlar tarafından geliştirilen veritabanları ve veri ambarları işletmelerin her bir departmanı için önemli strateji kaynakları sayılmaktadır.

Kuruluşlar için toplanan veriler üzerinde doğru çalışarak sağlıklı ve güvenilir sonuçlar elde etmek çok önemlidir. Kuruluşlar bu çıkarımlara ulaşmak için veri madenciliğinden faydalanmaktadırlar. Veri madenciliği, veri kaynaklarındaki yüksek boyutlu verinin içerisinde saklı olan ilişkilerin ve bilinmeyen kuralların bir takım teknolojik sistemlerde algoritmaların çalıştırılması sonucu keşfedilmesi ve verinin çok yönlü olarak analiz edilmesidir (Gargano & Raggad, 1999, s. 81-82). Veri madenciliği uzmanların bilimsel çalışmaları ve teknoloji ile beraber ilerlemiş ve her bir durum ve amaca yönelik modeller ve algoritmalar geliştirilmiştir. Son gelinen nokta ise, faaliyette olan ve bu gelişmelerden elde edilecek faydanın bilincinde olan kuruluşlar mevcut verilerinin işlenmesi için büyük yatırımlar yapmaktadırlar.

Veri madenciliğinde Birliktelik Kuralları, ilişkisel veritabanlarında ya da diğer veri kaynaklarında nitelikler arasında sıklıkla rastlanılan durumları incelemektedir. Amaç, veri setindeki farklı ilişki yapılarını ortaya çıkarmaktır (Kanellopoulos & Kotsiantis, 2006, s. 71). Birliktelik kuralları, veri seti içindeki birimler arasındaki ilişkileri analiz eden ve bu ilişkilerden hareketle hangi durumların aynı anda oluşacağına karar veren bir veri madenciliği modelidir.

2.1.Birliktelik Kuralları Analizinin Diğer Modellerle İlişkisi

Güçlü istatistikleri ortaya koyan birliktelik kuralları, ardışık zamanlı örüntüler yöntemi ile benzer amacı taşımaktadır. Her iki analizinde amacı birbirine bağlı olarak gelişen olgular arasındaki birliktelikleri, ilişkileri ortaya çıkarmaktır. Yöntemler arası tek fark, olguların

gerçekleşme zamanlarının farklı olmasıdır. Birliktelik kuralları, bir mağazadan X ürününü satın alan bir bireyin aynı anda Y ürününü satın alması ile oluşan ilişkiyi incelerken; ardışık zamanlı örüntüler, şu an X ürününü satın alan bir bireyin 3 hafta gibi bir süre sonrasında Y ürününü satın alması ile oluşan ilişkiyi incelemektedir. Ek olarak, her iki yöntem de müşterilerin alışveriş davranışlarını analiz etmek için kullanılabilir.

Birliktelik kuralları ve sınıflandırma modeli veritabanında sürekli meydana gelen durumları farklı amaç ve teknikler ile incelemektedir. Yöntemler arası en büyük fark veri madenciliğinde model ayrımlarında göze çarpmaktadır. Sınıflandırma modelinde bahsedildiği üzere, yöntem ayrıcalıklı bir değere sahip kategorik bir niteliğin değerini tahmin etmekteydi (Bramer, 2007, s. 237). Sınıflandırma, verilerin işlenmesi ve seçilen amaç değişkeninin veritabanındaki hangi değişkenlerin etkisi ile oluştuğunu belirleyip, bir model oluşturmaktadır. Bu model aracılığıyla veri setinde yeni bir birim söz konusu olduğunda amaç değişkeninin hangi değeri alacağı tahmin edilmeye çalışılmaktadır (Doğan O. , 2015, s. 53). Sınıflandırma kuralı tamamen tahmin yapabilme becerisi olacak bir model ortaya koymayı amaçlamaktadır. Birliktelik kuralları, bir veritabanından türetilebilecek herhangi bir bağıntıyı çıkararak daha genel bir yaklaşım izlemektedir. Birliktelik Kuralları analizinde "IF" ve "THEN" döngüleri aracılığıyla oluşturulan "nitelik=değer" eşitliğinden çıkan kurallara odaklanılmaktadır. Öncül (Antecedent/ LHS) ve ardıl (Consequent/ RHS) bölümleri bulunan kuralların çıkarıldığı veri setindeki niteliklerin kategorik olduğu varsayılmaktadır. Birliktelik kuralları, keşfedilmeyen ve bilinemeyen ilişkileri ortaya çıkaracak karakteristik kurallar oluşturma amacı taşımaktadır. Yani, bu modellerden biri geleceğe yönelik öngörüler oluşturma odağında çalışırken diğer model bilgi sağlamaya yönelik çalışmaktadır (Bramer, 2007, s. 237). Birliktelik kuralları analizinde sınıflandırma analizinden farklı olarak, veritabanındaki herhangi bir nitelik test edilebilir. Sadece, her bir kuralın ardıl ve öncül taraflarındaki en az bir nitelik için bazı açık kısıtlamalara tabi olan hiçbir nitelik görünmeyebilir (Bramer, 2007, s. 237). Ayrıca, sınıflandırma modelinde yer alan amaç değişkeni birliktelik kurallarında yer almamaktadır (Doğan O. , 2015, s. 53). Bu iki model arası farklılık için uzmanlar yorumlarda bulunmuşlardır. Witten ve Frank, birliktelik kurallarının veritabanında var olan bütün değişkenler arasındaki korelasyonları incelediğini ve sınıflandırma modelinin ise sadece bir sınıfı baz alarak kuralları ortaya çıkardığını savunmaktadır (Witten & Frank, 2005, s. 69). Ghosh ve Nath ise sınıflandırma modelinde sonuç bölümünün sadece bir özelliğe ait değerleri kapsadığını ve bu

özelliğin geçmiş verilerden hareketle daha önceden belirlendiğini ifade etmişlerdir. Ayrıca, bu kısıtlamanın birliktelik kurallarında geçerli olmadığını ve bu modeldeki tek bir kısıtlamanın her iki taraf içinde ortak bir özelliğe sahip olmamasını dile getirmişlerdir (Ghosh & Nath, 2004, s. 124).

2.2.Birliktelik Kuralları Analizi Hakkında Gelişmeler ve Uygulama Alanları

Birliktelik kuralları analizi, veritabanında eş zamanlı olarak meydana gelen olayların incelenerek, bu olaylardaki güçlü birlikteliklerin ortaya çıkarılmasını hedefleyen bir analizdir (Özkan Y. , 2013, s. 48). Birliktelik kuralları bazı çalışmalarda bağıntı analizi ve market sepet analizi olarak da adlandırılmaktadır. Birliktelik kuralları hakkında literatür incelendiğinde bu modelin 1993 tarihinde Agrawal, Imielinski ve Srikant'ın veri madenciliği alanına kazandırdığı görülmektedir. Çalışmalarının temel kaynağı apriori algoritmasına dayanmaktadır (Agrawal, Imielinski, & Swami, 1993).

Eski çalışmalarda nesneler arasındaki ilişkileri bulmak için örneklem yöntemleri ve birliktelik kuralları analizinin beraber kullanılması ile beraber, bu tarz yaklaşımın matematiksel ifadesi makalelerde belirtilmiş ve sonraki süreçlerde optimum gözlem sayısı konusunda bazı teknikler geliştirilmiştir. Çalışmalarda örneklem boyutu belirlenirken veriye odaklanılmamıştır. Veri haricindeki hata payı değeri ve araştırmacı tarafından atanan minimum destek ve güven değerleri gibi parametreler aracılığıyla örneklem boyutu belirlenmeye çalışılmıştır (Zaki, 1997, s. 4-6-7). Sonraki çalışmalarda ise veri niteliklerini de dikkate alarak en az işlem uzunluğu gibi kıstasları sağlayan formüller türetilmiştir (Chakaravarthy, Pandit, & Sabharwal, 2009, s. 6-7).

Birliktelik kuralları analizinde örneklem boyutu ile ilgili çalışmalar dışında farklı konu ve veri yapıları üzerinde farklı tekniklerle çalışmalar gerçekleştirilerek, modeller geliştirilmiştir. "Dependency Inference" adlı çalışmada, söz konusu ilişki için fonksiyonel bağımlılıkların ortaya çıkarılması problemi ele alınmıştır (Mannila & Raiha, 1987). "Data-Driven Discovery of Quantitative Rules in Relational Databases" adlı makalede ilişkisel veritabanları üzerinde nicel birliktelik kurallarının ortaya çıkarılması için bir tümevarım yöntemi geliştirilmiştir. Geliştirilen yöntem, karakteristik ve sınıflandırma kurallarının keşfedilmesini içermektedir (Cai, Cercone, & Han, 1993, s. 29). "Efficient Algorithms Association for Discovering Rules" adlı makalede yüksek boyutlu iki farklı veri kaynağında AIS ve Sıra Dışı Ürün Kümesi Tespiti

(Off-line Candidate Determination Algorithm/ OCD) algoritmaları çalıştırılmıştır. Bu veri kaynaklarından biri üniversite bünyesinde bulunan bir kursa kayıt yapan öğrenci bilgilerinden oluşmaktadır. Diğer veri kaynağı ise bir telefon şirketine ait arıza bildirimlerinden oluşan bir veritabanıdır. Çalışmanın amacı bu tip veri kaynaklarında algoritmaları karşılaştırarak birliktelik kuralları çıkarmaktır (Mannila, Toıvonen, & Verkamo, 1994, s. 181-186). Gosain ve Maneela, nicel birliktelik kurallarının çok önemli olduğunu ve gerçek hayatta veri kaynaklarında mutlaka var olacağını savunarak, bu kurallara özgü yeni algoritmalar ortaya koymuşlardır (Gosain & Maneela, 2013). 2016 tarihinde Chen ve ark. tarafından nicel veri kaynaklarında birliktelik kuralları elde edebilmek için nicel birliktelik kuralları yönteminde "Geçici Bulanıklaştırma" adlı bir yaklaşım ileri sürmüşlerdir (Chen, Hong, Lan, & Lin, 2016). Birliktelik kuralları analizi ile gerçekleştirilen çalışmalardan bazıları Tablo 17'de gösterilmektedir.

Tablo 17: Birliktelik Kuralları Analizi için Yapılan Çalışmalar

Yıl	Eser Adı
1996	Mining Quantitative Association Rules in Large Relational Tables (Agrawal & Srikant, 1996)
	Data Mining: An Overview from Database Perspective (Chen, Han, & Yu , 1996)
1999	Data Mining the Most Interesting Rules (Agrawal & Bayardo, 1999)
	Introduction to Data Mining and Knowledge Discovery (Edelstein, 1999)
	Parallel and Distributed Association Mining: A Survey (Zaki , 1999)
2001	A Survey of Association Rules (Dunham, Gruenwald, Hossain, & Xiao, 2000)
	Incremental Minig of Constrained Association Rules (Ayad, Nagwa, & Taha , 2001)
2002	Data Mining of Association Structures to Model Consumer Statistics & Data Analysis (Giudici & Passerone , 2002)
2003	An Improved Collaborative Filtering Approach for Predicting Cross-Category Purchases on Binary Market Basket Data (Mild & Reutterer , 2003)
2004	Multirelational Association Rule Mining (Flank, 2004)
	Data Mining Techniques: For Marketing, Sales and Customer Relationship Management (Berry & Linoff, 2004)
	Direct and Indirect Efects of Retail Promotions on Sales and Profits in the Do-It-Yourself Market (Poel, Schamphelaere, & Wets, 2004)
2005	A Computational Environment for Mining Association Rules and Frequent Item Sets (Grün, Hahsler, & Hornik, 2005)
	Discovering Knowledge In Data: An Introduction to Data Mining (Larose, 2005)
	Market Basket Analysis in A Multiple Store Enviroment (Chen, Hu, Shen, & Tang, 2005)
2006	Association Rules Mining: A Recent Overview (Kanellopoulos & Kotsiantis, 2006)
2007	Ranking Discovered Rules from Data Mining with Multiple Criteria by Data Envelopment Analysis (Chen M. , 2007)
2008	Diagnosing Hyperlipidemia Using Association Rules (Doğan & Türkoğlu, 2008)
2011	Uygulamalı Veri Madenciliği: Sektörel Analizler (Şimşek, Gürsoy, 2011)
2012	Data Mining Concepts and Techniques (Han, Kamber, & Pei, Data Mining: Concepts and Techniques, 2012)

Kaynak: Yazar tarafından derlenmiştir.

Birliktelik Kuralları analizinde elde edilen kuralların çok boyutlu olması sebebiyle Çok Boyutlu Birliktelik Kuralları analizi adında çalışmalar gerçekleştirilmiştir. Bu yöntem, farklı

çalışmalarda farklı algoritmalar ile çalışma imkânı bulmuştur. Sıklıkla meydana gelen niteliklerden kurallar oluştururken baz alınan minimum destek değerine eşit veya daha büyük destek değerlerini sağlayan kuralları oluşturmada ve saklamada OLAP ve Çevrimiçi Birliktelik Madenciliği (On-Line Association Mining/ OLAM) küpleri kullanılmaktadır. Büyük veritabanlarında çok boyutlu birliktelik kuralları ortaya çıkarılırken yaşanan karmaşıklıklara çözüm olarak OLAP sistemine benzer bir nitelikte ve OLAM sistemine dayalı Çevrimiçi Çok Boyutlu Birliktelik Kuralları Madencilik Sistemi (On-Line Multidimensional Association Rules Mining System/ OMARS) adlı bir yardımcı küp önerilmiştir (Lin, Tseng, & Wang, 2004, s. 1276-1277). "A Novel Algorithm of Mining Multidimensional Association Rules" adlı bir başka çalışmada ise ilişkisel veritabanlarında doğrudan çok boyutlu birliktelik kuralları yöntemini hızlıca uygulayacak Multi Dimensional Indexing Mining (MDIM) adlı yeni bir algoritma önerisinde bulunulmuştur (Xu & Wang, 2006, s. 771).

Çok boyutlu birliktelik kuralı yöntemi ile diğer yöntemlerin beraber kullanıldığı çalışmalar gerçekleştirilmiştir. Sug'un bu analiz ile beraber karar ağaçlarını kullandığı çalışması (Sug, 2003), Adlakhe ve ark.'nın çok boyutlu birliktelik kuralları analizinde yüksek destek değerlerini bulanık mantık ile elde ettikleri çalışmaları (Adlakha, Khare, & Pardasani, 2009) ve Pandey ve Pardasani'nin birliktelik kurallarını iki aşamada elde edebilmesi için yaklaşımlı küme modelinden yararlandıkları çalışma (Pandey & Pardasani, 2009) örnek olarak verilebilir. Analiz, farklı yöntemler ile beraber kullanılmasının yanı sıra, farklı konular üzerinde de çalışma fırsatı yakalamıştır. Chen ve ark. trafik kazaları konusunda bu yöntemden faydalanarak kazaya sebep olan etkenler arasındaki bağıntıları incelemişlerdir (Chen, He, Gao, Sun, & Wang, 2008). Çamurcu ve Doğan, eğitim alanında online bir sınav sonucu değerlendirme sistemi ile beraber yanlış verilen cevaplar arasındaki bağıntıyı inceleyerek, aynı anda en çok yapılan yanlış cevapları keşfetmişlerdir. Bu bilgiden hareketle, sınav sorularında yeniden yapılanmaya gidilmiştir (Çamurcu & Doğan, 2008).

Birliktelik kuralları analizinde iki veya daha çok olayın aynı anda gerçekleşmesi ile oluşan ilişkilerden pozitif kurallar elde edilmektedir. Ancak, gerçek hayatta tüm ilişkilerin pozitif olması beklenemez. Bu sebeple, zıt durumlar arasında gerçekleşen ilişkilerden negatif birliktelik kuralları elde edilmektedir. Negatif birliktelik kurallarında birbiri ile çakışan ögeleri ya da birbirini tamamlayacak ögeleri ifade etmek için market sepet analizi tercih edilir. Negatif

birliktelik kuralları yöntemini ilk kez Brin ve ark. dile getirmişlerdir. Ki-kare tabanlı modellerinde iki öge arasındaki bağıntıyı göstermek için istatistiksel teste başvurmaktadırlar. Pozitif ve negatif birliktelikleri yakalamak için korelasyon metriği yöntemini ileri sürmüşlerdir (Brin, Motwani, & Silverstein, 1997). Navathe ve ark. negatif birliktelik kurallarına yeni bir yöntem ile yaklaşmışlardır (Navathe, Omiecinski, & Savasere, 1998). Benzer bir yaklaşımı Buckles ve ark. savunmuştur (Buckles, Yuan, Yuan, & Zhang, 2002). Wu ve ark. hem pozitif hem de negatif kuralları yakalayacak farklı bir algoritma geliştirmişlerdir (Zhang, Zhang, & Wu, 2004). Veritabanından negatif birliktelik kurallarını çıkarabilmek için MDD ve MGD değerlerine küçük değerler atamak gerekmektedir. Bu yöntem, kural sayısında artış sağlamak ile beraber araştırmacının önemli birliktelikleri kaçırmasına ve gereksiz birliktelikler ile uğraşmasına sebep olmaktadır. Bu duruma çözüm olarak, ayrı bir algoritma ile az sayıda kural çıkartılabilir (Ayad A. M., 2000); (Karaibrahimoğlu, 2014, s. 69-70).

Birliktelik Kuralları yukarıda bahsedilen akademik alandaki gelişmeler ile beraber iş sahalarında da uygulama imkânı bulabilmiştir. Birliktelik kuralları, telekomünikasyon, envanter kontrolü, pazar ve risk değerlendirmesi (Kanellopoulos & Kotsiantis, 2006, s. 71), tıp biliminde teşhis ve araştırma çalışmaları, internet sitesi analizleri ve ülke savunmaları olmak üzere oldukça yaygın bir kullanım alanına sahiptir (Kanellopoulos & Kotsiantis, 2006, s. 80). Kuruluşların birliktelik kurallarına dayanarak geliştirdiği uygulamalara "BookMatcher" adlı program örnek olarak verilebilir. Bu program, E-ticaret sektöründe yer alan Amazon Şirketi tarafından hazırlanmıştır. Bu program aracılığıyla okuyucu kitlesi her bir kitabı 1 ve 5 arasında bir beğeni puanı ile değerlendirmektedir. Böylece, müşteriler ait oldukları müşteri profiline uygun kitap önerileri ile karşılaşmaktadır. Sonucunda hem satışlar arttırılmakta hem de müşteri bağlılığı güçlendirilmektedir (Alpaydın, 2000, s. 9). İşletmeler ürettikleri mal ve hizmetler için ideal üretim, stok yönetimi, ideal dağıtım, müşteri profilleri çıkarma ve müşteri sadakatini sürdürülebilir kılma gibi hedefleri sağlamak için birliktelik kuralları analizine başvurmaktadırlar.

2.3.Algoritmaların Gelişimi

Veri madenciliği yöntemleri sayısal anlamda çok fazla miktarda veriyi inceleyerek, ilgili konularda sonuca ulaşmaktadır. Manuel olarak hesaplanması zor hatta imkânsız olan veriler, bilgisayar ortamlarında uygun yöntem ve algoritmalar aracılığıyla incelenmektedir.

Algoritmalar hakkındaki çalışmalar KDD çalışmaları ardından yani 1990'lı yıllara dayanmaktadır. Yeni algoritmaların geliştirilmesi daha önce geliştirilen algoritmaların mevcut duruma uyum sağlayamaması ya da yeni ihtiyaçlara cevap verememesinden kaynaklıdır. Bu anlamda tüm algoritmalar için ortak nokta analizlerdeki gelişim ihtiyacıdır. Genel olarak bu ihtiyaçlar, veri türü ve boyutu, işlem süresi ve ideal sayıda faydalı kural üretme problemlerinden kaynaklıdır. Günümüzde de uzmanlar tarafından algoritmalar günün ihtiyaçları doğrultusunda hala geliştirilmekte ve ileride yeni algoritmaların elde edilmesi muhtemeldir.

Literatür taramasına göre birliktelik kuralları analizi için yapılan ilk algoritma Agrawal, Imielinski ve Swami tarafından geliştirilen baş harflerini vererek adlandırdıkları "AIS" adlı algoritmadır. AIS algoritması ile, veri setinden yaygın öge kümeleri elde etmeye çalışmışlardır. 1993 tarihinde Houtsma ve Swami tarafından set yönelimli madencilik (Set-Oriented Mining/ SETM) algoritması öne sürülmüştür (Houtsma & Swami, 1993). Agrawal ve Srikant 1994 tarihinde yaptıkları çalışmanın temeli Apriori algoritmasına dayanmaktadır. Ayrıca, bu çalışmalarında Apriori ile Apriori-TID algoritmalarının karması olan Apriori-Hybrid algoritmasını tanıtmışlardır (Agrawal & Srikant, 1994). Mannila ve ark. 1994 tarihinde zaman ve kural üretimi avantajı sağlayan OCD adlı algoritmayı önermişlerdir (Mannila, Toıvonen, & Verkamo, 1994, s. 181-182). Veritabanının birçok kez taranması sonucu oluşan problemlere çözüm olarak 1995 tarihinde Savasere ve ark. Partition (Bölümleme) adlı algoritmayı önermişlerdir (Omiecinski, Navathe, & Savasere, 1995, s. 432). Agrawal ve Srikant, birliktelik kuralları hakkında çalışmalara devam etmiştir. 1995 tarihindeki çalışmalarında sıralı örüntüleri bulmak için AprioriAll, AprioriSome ve DynamicSome adında üç algoritma geliştirmişlerdir. Bu çalışmada, her bir işleme ait müşteri, satın alınan nesneler ve işlem süresi bilgisi yer alan bir veritabanı üzerinde bir sıralı desen geliştirme konusu üzerinde çalışılmıştır. Bu veritabanı üzerinde AprioriAll, AprioriSome ve DynamicSome algoritmaları çalıştırılmış ve güçlü kurallar üretilmesine odaklanılmıştır. Sonuç olarak, veritabanı üzerinde AprioriAll ve AprioriSome algoritmaları aynı başarıyı yakalamıştır. Ancak, sıralı desenin oluşması için gerekli olan en az müşteri sayısının çok az olması durumunda AprioriSome algoritması daha başarılı sonuçlar vermiştir (Agrawal & Srikant, 1995, s. 3). 1996 tarihinde yaptıkları bir diğer çalışma da nicel birliktelik kuralları için Apriori algoritmasını tercih etmişlerdir (Agrawal & Srikant, 1996). 1996 tarihinde Toivonen, yüksek boyutlu veri setlerinde kural üretiminde tarama sayısının artması ile çalışma etkinliğinin azalmasını engellemek amacıyla Örnekleme

(Sampling) algoritmasını önermiştir (Toivonen, 1996, s. 134). 1996 tarihinde Agrawal ve ark. Veri Dağılımı (Data Distribution/ DD) adlı yeni bir algoritma geliştirmişlerdir. Ancak, aynı yıl hem Sayım Dağılımı (Count Distribution/ CD) hem de DD algoritmasına alternatif olarak Aday Küme Dağılımı (Candidate Distribution) adlı yeni bir algoritma önerilmiştir. 1996 tarihinde Zaki tarafından Ortak Aday Ürün Kümelerine Bölünmüş Veritabanı (Common Candidate Partitioned Database/ CCPD) algoritması geliştirilmiştir. 1997 tarihinde Han tarafından DD algoritması geliştirilerek Akıllı Veri Dağılımı (Intelligent Data Distribution/ IDD) algoritması, Zaki tarafından Paralel Bağlantı Kuralları (Parallel Association Rules/ PAR) ve Han tarafından Hibrit Dağılımı (Hybrid Distribution/ HD) adlı yeni algoritmalar geliştirilmiştir. Cai ve ark. ağırlıklı birliktelik kuralları ile ilgili bir çalışmada, destek değerlerine dayanan iki farklı algoritma önermişlerdir. MINWAL (O) ve MINWAL (W) algoritmalarının birbirinden farkı uygulandıkları verilere ait destek değerinin normalizasyon işlemine tabi tutulup tutulmamış olmasıdır. MINWAL (O) algoritması hem normalizasyon işlemi uygulanan hem de normalizasyon işlemi uygulanmayan veriler için kullanılırken, MINWAL (W) algoritması sadece normalizasyon işlemi uygulanan veriler için kullanılmaktadır. Bu iki algoritma karşılaştırıldığında, MINWAL (O) algoritması genellikle daha iyi sonuçlar vermiştir (Cai, Cheng, Fu, & Kwong, 1998, s. 77). 1998 tarihinde Harada tarafından çarpık işleme (Skew Handling/ SH) adlı bir başka algoritma öne sürülmüştür. 2000 tarihinde Han ve ark. FP-Growth (Frequent Pattern Growth) adlı bir algoritma geliştirilmiştir (Han, Pei, & Yin, 2000). 2001 tarihinde Flach ve Lachiche tarafından Tertius (Flach & Lachiche, 2001), Scheffer tarafından Tahminci Apriori (Predective-Apriori) (Scheffer, 2001) ve Das tarafından Rapid Association Rule Mining (RARM) (Das, Ng, & Woon, 2001) adlı yeni algoritmalar ortaya konmuştur. 2002 tarihinde Zaki ve Hsiao tarafından CHARM algoritması geliştirilmiştir (Hsiao & Zaki, 2002). FP-Growth algoritmasının geliştirilmesi adına Győrödi C., ve ark. tarafından "Dinamik FP ağacı" adı ile anılan bir FP-ağaç tabanlı teknik öne sürülmüştür. Bu yeni yöntemin, gerçek hayatta karşılaşılabilecek yüksek boyutlu veritabanları için kullanıldığında ciddi anlamda yüksek performans sağladığı görülmüştür (Győrödi, Győrödi, & Holban, 2004, s. 214).

2.4.Terminoloji ve Notasyon

Birliktelik Kuralları Analizinin temelleri 1993 tarihinde atılmış ve günümüzde popüler bir analiz olarak gelişimi devam etmektedir. Bu analiz türünün popüler olmasının en büyük

sebepleri uygulama alanının fazla oluşu, araştırmacı açısından kullanım kolaylığı sağlaması ve anlaşılabilir olmasıdır. Birliktelik kuralları analizinde kuralların elde edilmesi için veri tiplerine özgü algoritmalar olmasının yanında araştırmacının kural tipini ve sayısını belirleyebileceği yöntemler mevcuttur. Birliktelik kuralları analizinin matematiksel sürecinin iyi yürütülmesi için bazı teknik kavramların iyi anlaşılması gerekmektedir. Bu kavramlar; birliktelik, kayıt (İşlem- Transaction), öge, öge küme, yaygın öge küme, minimum destek değeri, minimum güven değeri, destek değeri (Support Value/ s), güven değeri (Confidence Value/ c) ve kaldıraç (Lift) değeri şeklindedir.

Veritabanında yer alan her bir kayıt (t_k), her bir faktör öge küme ($\mathcal{X}$) ve bu öge kümede yer alan her bir birim öge (x_k) ve veritabanındaki ortak olarak gerçekleşen olgular birliktelik olarak adlandırılmaktadır. Aşağıda $\mathcal{T}$ kayıt kümesi ve $\mathcal{X}$ öge küme verilmiştir.

$$\mathcal{T}: \{t_1, t_2, \dots, t_k\} \tag{2.1}$$

$$\mathcal{X}: \{x_1, x_2, \dots, x_k\} \tag{2.2}$$

Analiz sonucunda elde edilen kurallar ile ortaya konan ilişkiler veritabanında birlikte en çok tekrarlanan öğelere aittir (Çıngı, 2007, s. 16). Bu ögelerin yer aldığı öge kümeler yaygın öge küme olarak ifade edilir. Öge kümelerin yaygın öge küme olabilmesi için her bir ögeye ait destek değerinin MDD değerine eşit veya daha büyük olması yani veritabanında bu değer kadar tekrar gündeme gelmesi gerekmektedir (Han & Kamber, 2006, s. 345).

Birliktelik kuralları, önceki kayıtların işlenmesi ile spesifik kararların alınmasını sağlayan ve kayıtlar arasındaki ilişkileri ortaya çıkararak alınacak kararların güvenilirliğini arttırmayı hedefleyen bir analizdir. Bu analizdeki amaç, veritabanından araştırmacının baz aldığı minimum olasılık değeri ile şartlı olasılık değerlerine uygun kuralların ortaya çıkarılmasıdır (Çıngı, 2007, s. 16). Araştırmacı tarafından belirlenecek olan bu olasılık değerleri literatürde MDD ve MGD olarak adlandırılmaktadırlar. Bu kriterler için kesin bir sınır değer yoktur. Bu yüzden bu değerlerin belirlenmesi araştırmacının ulaşmak istediği kurallar için ciddi önem taşımaktadır. Veritabanından çıkarılacak kurallara sayısal olarak sınır koyan bu değerlerin çok düşük tutulması kural patlamasına ve araştırmacının ulaşmak istediği faydalı kuralları elde edememesine sebep olabilmektedir. Bu kriterler için bazı kaynaklarda MDD değeri için evrenin küçük olması durumunda %30-40 ve çok büyük olması durumunda %1-2 aralıklarında

atanmasının uygun olacağı savunulmuştur. MGD içinde %60-80 gibi yüksek aralıklarda bir değerin atanması önerilmiştir (Yarımağan, 2000, s. 296).

Birliktelik kuralları analizinde elde edilen kuralların geçerliliğinin %100 olması beklenmemektedir. Ancak, veritabanındaki veriden anlamlı bağıntıları ortaya çıkaracak kuralların elde edilmesi ve bu kuralların kabul görmesi için destek ve güven parametreleri hesaplanmaktadır. Literatürde bu parametreler için ilginçlik ölçüleri adı da verilmiştir. Genel bir ifade ile, güven ölçüsü herhangi bir olgunun gerçekleşmesi anında bir başka olayın gerçekleşme ihtimalini ve destek ölçüsü veritabanındaki birlikteliklerin sayısının, toplam gerçekleşen olguların sayısına oranını vermektedir. Destek ve güven değerleri MDD ve MGD değerleri ile karşılaştırılır ve bu değerlerden yüksek olan destek ve güven değerlerine sahip kurallar kabul edilmektedir. Elde edilen bir birliktelik kuralının gücünü yani öncül olgunun gerçekleşmesi sonucunda ardıl olgunun ne kadar ihtimal ile meydana geleceğini güven parametresi gösterirken, veritabanındaki öge kümelerin kaç kez meydana gelişini yani birlikteliklere ait frekansı destek parametresi gösterir.

Güçlü birliktelik kuralları içinde kurallara ait destek ve güven değerlerinin yüksek olması beklentisi mevcuttur. Ancak, her zaman bu şekilde destek ve güven değerleri de elde edilememektedir. Örneğin; bir ürün ile beraber nadiren satılan başka bir ürün söz konusu olduğunda iki ürün arasındaki bağıntı yöneticiler tarafından çok faydalı bilgiler sağlamamaktadır (Gürgen, 2008, s. 21). Destek ve güven değerleri yüksek olan her bir kural da bazen önemli olmayabilmektedir. Bu durumda elde edilen kuralların farklılığını ve önemi ölçmek için kaldıraç (lift) değeri hesaplanmaktadır. Bu değer, ilgili kuralın destek değerinin beklenen değere oranı şeklindedir (Berry & Linoff, 2004, s. 310). Daha açık bir ifadeyle, ilk gerçekleşen olayın sonrasında gerçekleşecek olayı hangi (0-1) oran ve hangi yönde (+/-) etkilediğini ortaya koymaktadır (Karaibrahimoğlu, 2014, s. 63).

2.5.Matematiksel Model

Birliktelik kurallarının kendisine özel bir kural yapısı mevcuttur. Elde edilen birliktelik kuralı iki bölümden oluşmaktadır. Kuralın sol kısmı öncül ve sağ kısmı ardıl olarak ifade edilir (Agrawal, Imielinski, & Swami, 1993, s. 2). Bu iki bölümde gerçekleştirilen kayıta dair nitelikler ve verilerin birbiri ile arasındaki bağıntılar, “IF (Eğer)” ve “THEN (Sonra)” döngüleri

kullanılarak ifade edilmektedir (Oğuzlar, 2004, s. 46). Model mantığı, "eğer bu durum gerçekleşirse, sonrasında şu durum gerçekleşir" yapısındadır (Fayyad U. M., Piatetsky-Shapiro, Smyth, & Uthurusamy, 1996, s. 28). Yani, IF döngüsünde aynı anda meydana gelen nitelikler kendini göstermekte ve THEN döngüsünde bu nitelikler arasında bağıntıyı temsil eden olaylar meydana gelmektedir. Bu şekilde ulaşılan birliktelik kuralları, $A \Rightarrow B$ şeklinde semboller ile gösterilebilir. Birliktelik kuralları için oluşturulan ilk modele göre;

$I = I_1, I_2, \dots, I_m$ m adet farklı ögenin bir kümesini,

$T = T_1, T_2, \dots, T_n$ kayıt kümesini ve

D kümesi ise her bir kayıt (T) için bir tanımlayıcısı olan veritabanını ifade etmektedir.

D veritabanında gerçekleşen her bir kayıt ve her bir (I) ise öge olarak adlandırılmaktadır. Veritabanlarında her bir kaydı gerçekleştiren veri grupları vardır. Kayıt numarası (Transaction Identification/ TID), her bir kaydı ifade eden tekil bir numaradır. D veritabanı $T \subseteq I$ şeklinde atanan ögelerin kümesi (öge küme) olsun. Veritabanında her bir kayıt t ikili bir vektör olarak ifade edilir ve t ögesinin satın alımı gerçekleştiğinde $t[k] = 1$, gerçekleşmediğinde ise $t[k] = 0$ sonucuna ulaşılır. Birliktelik kuralları analizinin matematiksel süreci için Tablo 18 aydınlatıcı olacaktır.

Tablo 18: Örnek Veri Seti

T	X	Y
1	1	1
.	.	.
.	.	.
.	.	.
k	0	1

Kaynak: Yazar tarafından derlenmiştir.

Tablo 18'e göre veritabanında yer alan kayıtlara ait tekil numaraların olduğu küme T ile gösterilmektedir. T kümesinde k adet kayıt bulunmaktadır. X ve Y ise birer öge kümedir. Burada, X ve Y öge kümeleri yer aldıkları kayıtlarda 1, yer almadıkları kayıtlarda o değerini almıştır. Tablo 1'de yer alan herhangi bir T_k kaydı X öge kümesini içeriyorsa, $X \subseteq T$ şeklinde gösterilir. Bu ifade X öge kümesinin T kümesinin bir alt kümesi olduğunu göstermektedir. Bu tablodan bir birliktelik kuralı elde etmek için $X \cap Y = \emptyset; X, Y \subset I$ yapısı sağlanmalıdır. Bu yapı

gereği, X ve Y öge kümeleri arasında bir bağıntı gerçekleşmiştir ve bu bağıntı I öge kümeleri birimini oluşturmuştur. Matematiksel olarak, analiz iki veya ikiden fazla öge küme arasındaki kesişimin boş küme olmaması ve gerçekleşen kayıtlar arasında kesişim olması durumunda, bu öge kümeler arasındaki gizli ilişkiyi bulmaktadır. Bu şekilde ulaşılan bir birliktelik kuralı aşağıdaki gibi gösterilmektedir.

$$X \Rightarrow Y \tag{2.3}$$

Yukarıdaki denkliğe göre, X öge kümesi Y öge kümesini belirlemektedir. Yani, Y öge kümesinin yer aldığı hareketler, X öge kümesinin yer aldığı kayıtlar içerisindedir. Daha açık bir ifadeyle Y öge kümesinin gerçekleşmesi öncelikle X öge kümesinin gerçekleşmesine bağlıdır (Agrawal, Imielinski, & Swami, 1993, s. 2). Birliktelik kuralları analizinde genel matematiksel süreç bu şekilde ilerlemektedir.

Birliktelik kuralları analizi süreci araştırmacı tarafından MDD ve MGD değerlerinin atanması ile başlar. Daha sonrasında süreç yaygın öge kümelerin elde edilmesi ve elde edilen yaygın öge kümelerden güçlü örüntülerin çıkarılması şeklinde ikiye ayrılmaktadır. İlk adım, analizde kullanılacak olan algoritmaların çalışma kalitesini belirlemektedir. Burada her bir öge için destek değeri hesaplanmaktadır. Daha sonra her bir ögeye ait destek değeri MDD değeri ile kıyaslanır. MDD değerine eşit veya büyük olan destek değerleri tespit edilir. Bu destek değerlerinin ait olduğu ögelerin bulunduğu öge kümeler yaygın öge küme olarak seçilir. İkinci adıma geçildiğinde, yaygın öge kümeler hem MDD hem de MGD değeri ile kıyaslanarak birliktelik kuralları elde edilir. Bir birliktelik kuralı için destek ve güven değerleri aşağıdaki gibi hesaplanmaktadır. D veritabanındaki toplam kayıt sayısı T olmak üzere,

$$\mathrm{s}(\mathrm{A} \Rightarrow \mathrm{B}) = \frac{\sigma(\mathrm{A} \cup \mathrm{B})}{\mathrm{T}} \tag{2.4}$$

$$\mathrm{c}(\mathrm{A} \Rightarrow \mathrm{B}) = \frac{\sigma(\mathrm{A} \cup \mathrm{B})}{\sigma(\mathrm{A})} \tag{2.5}$$

şeklinde gösterilmektedir (Adlakha, Khare, & Pardasani, 2009, s. 72). Burada destek değeri, veritabanındaki A ve B ögesinin kapsayan kayıtların sayısının, veritabanındaki toplam kayıt sayısına oranlanması ile elde edilmektedir. Güven değeri ise A ve B ögesini kapsayan kayıt sayısının, A ögesini kapsayan kayıt sayısına oranlanması ile elde edilmektedir. Bir birliktelik

kuralı için, destek parametresi ile kuralın öncül ve ardıl taraflarında yani evrenin yüzde kaçında bağıntının gerçekleştiğine ulaşılır. Güven parametresi ise, kuralın öncül kısımda sağlanan durum veya durumların yüzde kaçının kuralın ardıl kısmında gerçekleştiğini vermektedir (Yarımağan, 2000, s. 296). Yani, destek değeri bir birliktelik kuralının istatistiksel olarak ifade ettiği anlamı ve güven değeri, iki veya daha fazla öge kümeleri arasındaki karşılıklı olan ilişkinin derecesini göstermektedir (Gürgen, 2008, s. 21). Düzenlenmiş bir veritabanından elde edilen ve geçerliliği sağlanan bir birliktelik kuralı aşağıdaki gibi olsun.

$$A \Rightarrow B\ [s = \%40,\ c = \%50] \tag{2.6}$$

Yukarıdaki birliktelik kuralına göre, destek ve güven değerleri şu şekilde yorumlanmaktadır. İncelenen tüm kayıtlarda A ve B öge kümeleri beraber meydana gelme olasılığı %40'tır. A öge kümesinin gerçekleştiği kayıtların %50'sinde B öge kümesi de gerçekleşmektedir.

Kaldıraç değeri, veritabanından çıkarılan bir kuralın ne kadar ilginç olduğunu tespit etmek amacıyla kullanılmaktadır. Kaldıraç değeri hesaplanırken, ilgili kuralın her iki tarafındaki öge kümelerin beraber görülme olasılıklarının beklenen değeri gündeme gelmektedir. Burada beklenen değer, her iki taraftaki öge kümenin destek değerlerinin çarpımı ile hesaplanmaktadır.

$$\text{Kaldıraç (A)} = \frac{\text{c(A)}}{\text{s(Y)}} = \frac{\text{P(Y|X)}}{\text{P(Y)}} = \frac{\text{P(X}\cap\text{Y)}}{\text{P(X).P(Y)}} = \frac{\text{X ve Y}'\text{yibirlikte içeren kayıtların sayısı}}{\text{X}'\text{i içeren kayıtların sayısı}\times\text{Y}'\text{yi içeren kayıtların sayısı}} \tag{2.7}$$

Birliktelik kuralları analizinde kaldıraç değerinin önemini anlatmak için Tablo 19'daki örnek veri seti gösterilmektedir. Bu veri seti 10.000 kayıt ve X ve Y adında iki öge kümeden oluşmaktadır. Öge kümelerin kayıtlarda yer alıp almadıklarına göre 1-0 değeri atanmıştır. Yapılan incelemelerde kayda alınan kayıtların 6.000'inde sadece X, 7.500'ünde sadece Y öge kümeleri görülmüştür. X ve Y değerlerinin ortak görüldüğü kayıt sayısı 4.000'dir. Analiz sürecinde MDD değeri %30 ve MGD değeri %60 olarak atanmıştır.

Tablo 19: Kaldıraç Değeri İçin Örnek Veri Seti

Kayıtlar	X	Y
1	1	1
.	.	.
10.000	0	1

Kaynak: Yazar tarafından derlenmiştir.

Analiz sonucunda aşağıdaki birliktelik kuralı elde edilmiştir.

Destek değeri,

$$s\ (X \Rightarrow Y) = \mathrm{P(X\ ve\ Y)} = \mathrm{P(X\ ve\ Y)/T} = {}^{4.000}/_{10.000} = \%40 \quad (2.8)$$

Güven değeri,

$$c\ (X \Rightarrow Y) = \mathrm{P(X|Y)} = \mathrm{P(X\ ve\ Y)/P(Y)} = {}^{4.000}/_{6.000} = \%66 \quad (2.9)$$

$$X \Rightarrow Y\ [s = \%40, c = \%66] \quad (2.10)$$

Yukarıda destek ve güven değeri belirlenen birliktelik kuralı MDD ve MGD değerini sağlamıştır. Bu durumda, kural geçerli sayılacaktır. Kuralın gerçekten geçerliliğinin sınanması için aşağıdaki hesaplamalar yapılmıştır.

Bir X faktörünün gerçekleşme olasılığı

$$P(\{X\}) = {}^{6.000}/_{10.000} = \%60 \quad (2.11)$$

Bir Y faktörünün gerçekleşme olasılığı

$$P(\{Y\}) = {}^{7.500}/_{10.000} = \%75 \quad (2.12)$$

X ve Y faktörlerinin beraber gerçekleşme olasılığı

$$P(\{X,Y\}) = {}^{4.000}/_{10.000} = \%40 \quad (2.13)$$

Kaldıraç değeri,

$$Kaldıraç(X \Rightarrow Y) = \frac{P(\{X,Y\})}{P(\{X\}) \times P(\{Y\})} = \frac{0.40}{0.60 \times 0.75} = \%89 \quad (2.14)$$

Yukarıda, Y faktörünün gerçekleşme olasılığının %75 olup, güven değeri %66 dan büyük olması elde edilen kuralın geçerliliğini tehlikeye sokmaktadır. Bu iki öge küme arasındaki ilişki yanlış ilişkilendirilmiş olup, öge kümelerden birinin gerçekleşmesi diğer öge kümenin gerçekleşme olasılığını aşağıya çekmektedir. Böyle bir durumda destek ve güven değerleri

önemini yitirmekte ve bu problem için kaldıraç değerine ihtiyaç duyulmaktadır. Aslında bu değer öge kümeler arasındaki korelasyonu değerlendirmektedir. Kaldıraç değeri hesabındaki pay kısmı öge kümelerin beraber gerçekleşme olasılığını ve payda kısmı iki öge kümenin birbirinden bağımsız olarak meydana gelme olasılığı ile ilgilidir. Kaldıraç değeri 1 değerine göre değerlendirilir. Değerin 1'e eşit olması öge kümeler arasında bir bağlantının olmadığını, 1'den büyük olması öge kümeler arasında pozitif bir korelasyon olduğunu (Yani, bir öge kümenin gerçekleşmesi ile diğer öge küme de gerçekleşecektir) ve 1'den küçük olması öge kümeler arasında negatif bir korelasyon olduğunu göstermektedir. Yukarıda ulaşılan 0,89 değeri ile bu iki öge küme arasında negatif bir korelasyon olduğu anlaşılmaktadır. Böyle bir sonuca destek ve güven değerleri ile ulaşılması imkansızdır (Han, Kamber, & Pei, 2012, s. 265-266).

Literatürde yukarıda uygulanan yöntem dışında araştırmacı tarafından sınırlanmış öge tanımlaması şeklinde farklı bir yöntem daha mevcuttur. Bu yönteme göre, sınırlandırılan öge kuralların detayına getirilen kısıtlama için kullanılan mantıksal bir söyleyiştir. Sınırlanmış öge olarak veritabanındaki bir veya birden fazla öge seçilir. Böylece, çalışmada sadece bu öge/ögeleri içeren kurallar ile ilgilenilmektedir (Dolgun M. Ö., 2006, s. 36).

2.6.Birliktelik Kuralları Türleri

Birliktelik kuralları, sık aralıklarla, beraber ve aynı anda meydana gelen ögelerin veya niteliklerin keşfedilmesi işlemidir (Fayyad U. M., Piatetsky-Shapiro, Smyth, & Uthurusamy, 1996, s. 28). Yani, birliktelik kuralları analizi nitelikler arasındaki korelasyonlar üzerinde durmaktadır. Birliktelik kuralları analizi genel olarak elde edilen kuralın boyutuna göre tek boyutlu birliktelik kuralları ve çok boyutlu birliktelik kuralları olmak üzere ikiye ayrılmaktadır. Birliktelik kuralları analizinde bu ayrım dışında uygulama kapsamına (konu, çalışma biçimi ve veri türü) göre Nicel Birliktelik Kuralları (Quantitative Association Rules), Genelleştirilmiş Birliktelik Kuralları (Generalized Association Rules), Çoklu Min-Destek Birliktelik Kuralları (Multiple Min-Supports Association Rules), Maksimal Birliktelik Kuralları (Maximal Association Rules), Multimedya Birliktelik Kuralları (Multimedia Association Rules) ve Zamansal ve Mekânsal Birliktelik Kuralları (Temporal and Spatial Association Rules) olmak üzere altı alt gruba ayrılmaktadır.

Bazı uzmanlar birliktelik kurallarını boyut farkı ve uygulama kapsamı ayrımları dışında sınıflandırmışlardır. Bu sınıflandırmaya göre Boolean Birliktelik Kuralları, Nicel Birleştirme Kuralları ve Soyutlama Sınıflarına Göre Birliktelik Kuralları elde edilmiştir. Boolean Birliktelik Kurallarında alakalı kural ögelerin varlığı ve yokluğu ile ilgilendiğinde oluşmaktadır. Bu tip kurallar "Boelan (Doğru-Yanlış)" yani mantıksal kurallardır. Burada kurallar için alınan değerlerin türlerinde ayrıma gidilmektedir. Bu gibi kurallara market sepet analizi ile kolayca ulaşılabilir (Han & Kamber, 2001) (Dolgun M. Ö., 2006, s. 40). Tek boyutlu birliktelik kuralları için verilen örnek bu tip bir kuraldır. Nicel birleştirme kurallarında kurallarda alınan değerlerin türleri önemlidir. Eğer, alakalı kural nicel ögeler veyahut nitelik değerler arasındaki bağıntıyı çözümlüyorsa, bu tür kurallar "Nicel Birleştirme Kuralları" olarak adlandırılır. Bu tip kurallarda, ögeler için söz konusu olan nitelikler ya da nicel değerler aralıklara ayrılmıştır (Han & Kamber, 2001);(Dolgun M. Ö., 2006, s. 40). Çok boyutlu birliktelik kuralları için verilen örnek bu tip bir kuraldır. Çalışmalarda elde edilen birliktelik kuralları, farklı tekniklerle soyutlama sınıflarına ayrılabilir. Örneğin, aşağıdaki iki farklı birliktelik kuralı için söz konusu ürünler ayrı soyutlama sınıflarına ayrılmaktadır.

$$Cinsiyet(A,"Kadın")\text{^}Gelir\ (A,"30k\ldots.38k) \Rightarrow Alışlar(A,"Spor\ Araba") \quad (2.15)$$

$$Cinsiyet(A,"Kadın")\text{^}Gelir\ (A,"30k\ldots.38k) \Rightarrow Alışlar(A,"Araba") \quad (2.16)$$

Yukarıda, aynı müşteri profili için farklı iki kural elde edilmiştir. Müşteri profili, veri setinde geliri 30k ile 38k arasında olan kadınlardır. 2.15'te yer alan kurala göre; bu müşteri profili araba alımlarında tercihlerini spor arabadan yana kullanmaktadırlar. 2.16'da yer alan kurala göre, bu müşteri profili araba almaya eğilimlidirler. Bu müşteri profili için ilk kuralda hem araba alımına eğilimli olduğu hem de araba alımındaki öncelikleri bildirilirken; ikinci kuralda sadece araba alma eğiliminde oldukları bildirilmektedir.

Genel olarak yukarıda adı geçen birliktelik kuralları yöntemlerinde pozitif ilişkilere bakılmaktadır. Yani, bir mağaza veri seti üzerinden "bir müşteri C ürününü satın aldığında, yüksek bir tahminle F ürününü de satın alabilir" şeklinde pozitif birliktelik kuralları elde edilmektedir. Ancak, gerçek hayat verilerinde rastlanması muhtemel olan negatif ilişkilerde önemlidir. Aynı mağaza veri seti üzerinden "bir müşteri D ürününü satın aldığında, yüksek bir tahminle H ürününü satın almayacaktır" şeklinde negatif birliktelik kuralları elde edilmektedir.

Burada iki ürün arasında korelasyon sıfır olsaydı, bu şekilde bir negatif birliktelik kuralı elde edilemezdi. Öge küme içerisinde negatif ögeler olması durumunda bu kümeye negatif öge küme adı verilir. İlgili kümenin destek değeri MDD değerine eşit veya büyük ve güven değeri MGD değerine eşit veya büyük ise bu küme ile negatif birliktelik kuralları elde edilmiş olur. Bu tür birliktelik kuralları yöntemlerine ek olarak, seyrek birliktelik kuralları (Rare Association Rule) ve işlemsel birliktelik kuralları (Transactional Association Rule)'da vardır.

2.6.1. Tek Boyutlu Birliktelik Kuralları

Birliktelik kuralları analizinde iki öge arasında bulunan bağıntıyı temsil eden kurallar "Tek Boyutlu Birliktelik Kuralları" olarak adlandırılır. Tablo 20'de yer alan 256 birimlik örnek mağaza veri seti fatura bilgisi ve faturalarda X ve Y ögelerinin aynı işlemde yer alıp almadığı bilgisinden oluşmaktadır.

Tablo 20: Tek Boyutlu Birliktelik Kurallar İçin Örnek Veri Seti

Fatura No	X	Y
1	1	1
2	1	0
3	1	0
.	.	.
256	0	0

Kaynak: Yazar tarafından derlenmiştir.

Tablo 20'deki veri seti aracılığıyla aşağıdaki tek boyutlu birliktelik kuralı elde edilmiştir.

$$X \Rightarrow Y\ [s = \%40,\ c = \%50] \qquad (2.17)$$

Yukarıdaki kurala göre incelenen 256 adet faturanın %40'ında kişiler tarafından bu iki öge beraber tercih edilmiştir. Yani, kişiler tarafından ögelerin aynı anda tercih edilme olasılığını vermektedir. X ögesinin tercih edildiği kayıtların %50'sinde aynı anda Y ögesi de tercih edilmektedir. Yani, X ögesini tercih eden kişilerin beraberinde tercih edeceği Y ögesinin satın alma olasılığını vermektedir.

2.6.2. Çok Boyutlu Birliktelik Kuralları

Birliktelik kuralları analizinde üç veya üçten fazla öge arasında bulunan bağıntıyı temsil eden kurallar "Çok Boyutlu Birliktelik Kuralları" olarak adlandırılır. Daha açık bir ifadeyle, çok

boyutlu birliktelik kuralları yönteminde bir öge birden çok öge ile beraber mukayese edilerek bu ögeler arasındaki ilişkiler ortaya çıkarılmaktadır. Tablo 21'de yer alan 303 birimlik örnek bir veri setinde öğrencilerin (Kadın:0 ve Erkek:1) X, Y ve Z derslerindeki başarı durumları (Başarılı: 2, Geçer: 1 ve Başarısız: 0) yer almaktadır.

Tablo 21: Çok Boyutlu Birliktelik Kuralları için Örnek Veri Seti

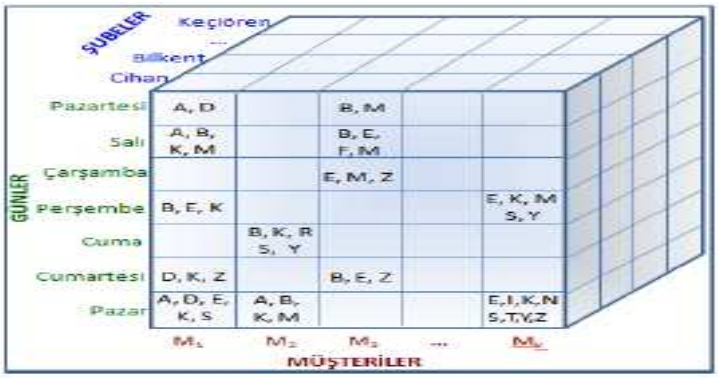
Birim	Yaş	Gelir (TL)	Y	Z
1	18-22	1.000-2.500	0	1
. .	. .	. .	. .	. .
303	23-27	8.501-10.000	1	0

Kaynak: Yazar tarafından derlenmiştir.

Tablo 21'deki veri seti aracılığıyla aşağıdaki çok boyutlu birliktelik kuralı elde edilmiştir.

$$Yaş\ (A,"23-27)\text{^}Cinsiyet(A,"Kadın")\text{^}Gelir\ (A,"4.001\ldots.5.500) \Rightarrow Alışlar(A,"Spor\ Araba") \quad (2.18)$$

A bir kişi olmak üzere, 23-27 yaş grubunda yer alan ve 4.001- 5.500 TL aralığında geliri olan bir kadın spor araba satın alabilir, sonucuna varılmaktadır. Buradaki birliktelik kuralı, yaş, cinsiyet, gelir ve alışlar olmak üzere dört boyut içermektedir. Bu örnek dışında, çok boyutlu birliktelik kuralları ile bir mağaza verisinde A ürününü tercih eden bir kişinin aynı anda B ve C ürünlerini de tercih etme olasılıkları elde edilebilir. Bu tür analizler için çok boyutlu birliktelik kuralları analizinde OLAP küpleri kullanılmaktadır. Şekil 13'te görüldüğü üzere müşterinin işletmenin hangi noktasında, ne zaman ve neler satın aldığı gibi bilgileri içeren kurallara ulaşılabilmektedir. İşletme yöneticileri OLAP küpü üzerinden değişkenlerin ortak yanlarını bir araya getirerek gizli kalan bağıntıları elde edebilir (Birant, ve diğerleri, 2010, s. 259-260).



Şekil 13: OLAP Küpü

Kaynak: Birant, D., Kut, A., Altınok, B., Altınok, H., Altınok, E., Ihlamur, M. ve Ventura, M. (2010). İş Zekâsı Çözümleri İçin Çok Boyutlu Birliktelik Kuralları Analizi. Akademik Bilişim'10- Xıı. Akademik Bilişim Konferansı Bildirileri. Muğla Üniversitesi. ss.:257-263.

Çok boyutlu birliktelik kuralları, karşılaştırma ögeleri tekrar gerçekleşmeyen kurallar "Boyutlar Arası Birliktelik Kuralları" ve karşılaştırma ögeleri tekrar gerçekleşen kurallar "Hibrit Birliktelik Kuralları" olmak üzere ikiye ayrılmaktadır. Şekil 13'e göre elde edilen boyutlar arası birliktelik kuralları ve hibrit birliktelik kuralları aşağıda gösterilmiştir (Birant, ve diğerleri, 2010, s. 260).

Boyutlar arası birliktelik kuralları

Şube ("Cihan")^Gün("Salı") ⇒ Satın Alma ("A ürünü") (2.19)

Hibrit birliktelik kuralları

Şube ("Cihan")^Gün("Salı") ^Satın Alma ("A ürünü") ⇒ Satın Alma ("B ürünü") (2.20)

2.6.3. Nicel Birliktelik Kuralları

Birliktelik kuralları analizi ile ilgili çalışmalarda veri türleri nitel veri şeklindedir. Ancak, gerçek hayatta bir veritabanı nitel ve nicel verilerin bir arada olduğu karma bir yapıya sahiptir. Bazı birliktelik kuralları algoritmalarının işleyişi açısından veritabanının nicel verilerden oluşması sorun yaratabilmektedir. Bu yöntemde, nicel değerlerin olduğu veritabanları üzerinde, nicel verilerin bazı kriterlerce gruplara ayrılması ile nicel birliktelik kuralları elde edilmektedir.

Gruplar, belli bir sıra düzeninde adlandırılır. Bunun sebebi, veriler arasındaki sıranın korunmasıdır (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 44). Nicel birliktelik kuralları için aşağıdaki örnek verilebilir (Agrawal & Srikant, 1996, s. 1-2).

Bir müşteri,

$$Bir\ ekmek\ için\ 3-5\$\ aralığında\ ücret\ öder\ \Rightarrow Bir\ şişe\ şarap\ için\ 10-20\$\ aralığında\ ücret\ öder$$

Nicel birliktelik kuralları, boolean birliktelik kuralları şeklinde eşleştirilebildiğinde Boolean Birliktelik Kuralları için kullanılabilen herhangi bir algoritma seçeneği, nicel birliktelik kuralları için de kullanılabilmektedir (Agrawal & Srikant, 1996, s. 1-2). Nicel birliktelik kuralları yönteminde boolean birliktelik kuralları sisteminin uygulanması ile başarılı sonuçlar elde edilebilir. Ancak, bu sistemin uygulanması ile nicel birliktelik kuralları yönteminde iki önemli problem de gelişebilir. Nicel niteliklerin gruplara ayrılması aşamasında, bir nicel niteliğe ait grup sayısı çok ise veyahut bu nitelik gruplara ayrılmadığı zaman gruplardan birine ait destek değeri düşük çıkabilir. Bu nedenle, niteliğe ait bazı gruplar için destek değerleri MDD değerini aşamadığı için kural elde etmek mümkün olamayacaktır. Nicel birliktelik kurallarındaki bu probleme "MDD" (Minimum Destek) Problemi" denmektedir (Agrawal & Srikant, 1996, s. 2). Nicel birliktelik kurallarındaki bir diğer problem ise "MGD" (Minimum Güven) Problemi" dir. Bir nicel niteliğe ait grup sayısı az ise bilgi kaybı oluşmaktadır. Bu nedenle, niteliğe ait bazı gruplar için güven değerleri MGD değerini aşamadığı için kural elde etmek mümkün olamayacak ve gruplara ait boyutlar arttıkça bilgi kaybı da artacaktır. Örneğin; Tablo 22'deki veri setinden hiç arabası olmayan ve evli olmayan durumu içeren bir kural için güven değeri %100'dür. Ancak, Araba sayısı faktöründe arabası hiç olmayan ve bir arabası olan kişiler bir gruba atansaydı, kuralın güven değeri, %66,6 olacaktır (Agrawal & Srikant, 1996, s. 2). Bu sebeple, bilgi kaybında artış meydana gelecektir.

Nicel birliktelik kurallarındaki bu iki problemden kaynaklı olarak, gruplar arasındaki uzunluk çok ise MGD ve gruplar arasındaki uzunluk kısa ise MDD değerleri sağlanamamaktadır. Bu durumlara genel çözüm olarak, mümkün tüm gruplar nicel niteliklere ait değerler üzerinde veyahut gruplandırılmış kısımların üzerindedir. Komşu olan gruplar birleştirildiğinde MDD problemi çözülmektedir. Devam etmekte olan MGD problemi için de grup sayısının arttırılmasıyla çözüm bulunmaktadır.

MDD ve MGD problemleri için sunulan çözüm yolları iki farklı sorunu beraberinde getirmektedir: Uzun işlem süresi ve fazla kural üretimi. Nicel bir niteliğe ait grup sayısı ortalamanın üstünde arttırılması durumunda veritabanında çok fazla taramaya yol açması ile işlem süresi oldukça artmaktadır. Ayrıca, bir nicel niteliğe ait bir değer veya grup MDD değerini sağlıyorsa, bu değer ya da aralığı kapsayan grup farklı bir kural daha oluşturacaktır. Bu durum kural sayısında patlamaya sebep olmaktadır. Bu problemler için, işlem süresinin çok fazla kural üretilmesine rağmen arttırılması ile beraber çok fazla kural üretme maliyetiyle bilgi kaybı azaltılabilir. Bu da MGD problemine işaret etmektedir. Diğer bir taraftan nicel bir niteliğe ait değerleri arasında hiyerarşi söz konusu değilse, nitel bir niteliğe ait değerlerin birleştirilmesi mantıklı olmayacaktır. Bu esnada, sınıflandırma ile nitel niteliklere ait değerler birleştirilebilir (Agrawal & Srikant, 1996, s. 2).

Nicel birliktelik kuralları için örnek veri seti Tablo 22'de gösterilmektedir. Veri setinde, yaş ve araba sayısı değişkenleri nicel değişkenler ve evlilik durumu değişkeni nitel değişken olmak üzere üç değişken mevcuttur (Agrawal & Srikant, 1996, s. 1-2).

Tablo 22: Kişilere Ait Demografik Veri Kayıtları

Kayıt No	Yaş	Evlilik	Araba Sayısı
100	23	Hayır	1
200	25	Evet	1
300	29	Hayır	0
400	34	Evet	2
500	38	Evet	2

Kaynak: AGRAWAL, R. ve SRİKANT R. (1996). Mining Quantiative Association Rules İn Large Relational Tables. ACM SIGMOD'96 International Conference of Management of Data. Montreal. p.1-12. ss.:2.

Tablo 23'te, Tablo 22'de yer alan kayıtların boolean birliktelik kuralları problemine göre eşleştirilmesi gösterilmektedir. Yaş faktörü iki, evlilik faktörü iki boolean özniteliğe ayrılmıştır. Araba sayısı faktörü değerleri küçük olduğu için aralıklara ayrılmadan her bir değeri bir boolean alanına eşit sayılmıştır (Agrawal & Srikant, 1996, s. 1-2). Ek olarak, aşağıda verilen örnek nicel birliktelik kuralları yönteminin özel bir hali olan aralık veri birliktelik kurallarına dahil olmaktadır.

Tablo 23: Boolean Birliktelik Kuralları Probleminin Eşleştirilmesi

Tanım	Yaş		Evlilik		Araba Sayısı		
	20-29	30-39	Evet	Hayır	0	1	2
100	1	0	0	1	0	1	0
200	1	0	1	0	0	1	0
300	1	0	0	1	1	0	0
400	0	1	1	0	0	0	1
500	0	1	1	0	0	0	1

Kaynak: AGRAWAL, R. ve SRİKANT R. (1996). Mining Quantiative Association Rules İn Large Relational Tables. ACM SIGMOD'96 International Conference of Management of Data. Montreal. p.1-12. ss.:3.

Tablo 23'e göre 100 kayıt numarasında yaşı 23 olan biri 20…29 yaş grubunda "1" değerini, 30…39 yaş grubunda "0" değerini alır. Bu mantıkla oluşturulan boolean birliktelik kuralları sistemine göre nicel birliktelik kuralları elde edilebilir (Agrawal & Srikant, 1996, s. 1-2). Çözüm süreci hakkında bilgi veren ve elde edilen diğer tablolar Ek 1'de gösterilmektedir.

Tablo 24: Nicel Birliktelik Kuralları Örneği

Numune Kurallar	Destek	Güven
⟨Yaş: 30 ... 39⟩ ve ⟨ Evlilik: Evet⟩ ⇒ ⟨Araba Sayısı: 2⟩	%40	%100
⟨Araba Sayısı: 0 ... 1⟩ ⇒ ⟨ Evlilik: Hayır⟩	%40	%66,6

Kaynak: AGRAWAL, R. ve SRİKANT R. (1996). Mining Quantiative Association Rules İn Large Relational Tables. ACM SIGMOD'96 International Conference of Management of Data. Montreal. p.1-12. ss.:2.

Veri setinden hareketle, MDD değeri %40 ve MGD değeri %50 olmak üzere Tablo 24'teki nicel birliktelik kuralları elde edilmiştir (Agrawal & Srikant, 1996, s. 1-2). Birinci kurala göre tüm kayıtlar incelendiğinde yaşı 30-39 aralığında ve evli olan kişilerin aynı anda 2 arabaya sahip olma olasılığı %40'tır. Yaşı 30-39 aralığında ve evli olan kişilerin %100'ünün 2 araba sahibi olma olasılığı vardır. İkinci kurala göre tüm kayıtlar incelendiğinde arabası hiç olmayan ya da bir arabası olan kişilerin aynı anda evli olma olasılığı %40'tır. Arabası hiç olmayan ya da bir arabası olan kişilerin %66,6'sının evli olma olasılığı vardır.

Nicel birliktelik kuralları yönteminin dört farklı özel hali mevcuttur. Bunlar; Aralık Veri Birliktelik Kuralları (Interval Data Association Rules), Bulanık Birliktelik Kuralları (Fuzzy Association Rules), Nicel Profil Birliktelik Kuralları (Quantitative Profile Association Rules) ve Nicel Ratio Birliktelik Kuralları (Quantitative Ratio Association Rules) şeklindedir.

2.6.3.1. Aralık Veri Birliktelik Kuralları

Birliktelik kuralları analizi matematiksel olarak incelendiğinde veritabanı önemli bir yer almaktadır. Veritabanındaki karmaşıklıkta hem niteliklerin sayısının hem de her bir niteliğe ait değerlerin sayısının etkili olduğu anlaşılmıştır. Çok büyük veritabanlarındaki bu karmaşıklığı ortadan kaldırmak amacıyla veriler belli bir kritere göre gruplara ayrılmasına ve bunların bir bütün olarak düşünülmesi kararına varılmıştır. Bunun sebebi, bir niteliğe ait değerler doğrusal olarak sıralanırsa bu değerleri gruplara ayırmak mümkün olmaktadır. Aralık veri birliktelik kuralları için Tablo 25 örnek olarak verilebilir. Tablodaki yaş değişkenindeki tüm değerlerin ayrı ayrı incelenmesi yerine tüm değerlerin bölümlendirildiği aralıklar incelenecektir. Bu sayede hem taramadaki karmaşıklık giderilebilir hem de daha net birliktelik kuralları sağlanabilir (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 48).

Tablo 25: Aralık Veri Seti

Maaş	Aralık 18K-38K	Aralık 38K-58K	Aralık 58K-78K	Aralık 78K-98K
18K	1	0	0	0
30K	1	0	0	0
31K	1	0	0	0
80K	0	0	0	1
81K	0	0	0	1
82K	0	0	0	1

Kaynak: DUNHAM, M. H., XİAO, Y., GRUENWALD L. ve HOSSAİN, Z., (2000). *A Survey of Association Rules Teknik Rapor.* Southern Methodist University, Department of Computer Science, TR00-CSE-8. ss.:49.

2.6.3.2. Bulanık Birliktelik Kuralları

Nicel birliktelik kuralları genel olarak, veri gruplarının kesin sınırlar ile ayrıldığını varsaymaktadır. Ancak, net olmayan noktalar için bölümlemeye başvurulabilir. Bu şekilde elde edilen nicel birliktelik kuralları "Bulanık Birliktelik Kuralları" olarak adlandırılmaktadır. Örneğin; Arabalar, alış fiyatlarına göre (0-29.999 TL), (30.000-59.999 TL), (60.000, …. TL)

olarak üç gruba ayrılsın. Normal şartlarda herhangi bir araba bu üç gruptan birine dahil olacaktır. Bu durumda, 59.999 TL değerindeki bir araba sadece 2. grup için değerlendirmeye alınacaktır. Bulanık birliktelik kuralları için aşağıdaki örnek verilebilir.

$$X: A \Rightarrow Y: B \tag{2.21}$$

Yukarıda verilen bulanık birliktelik kurallarına göre X ve Y değerleri birer niteliktir. A ve B değerleri ise X ve Y'deki özellikler için bulanık grup üyelik işlevleridir. X niteliğinin, A'yı karşıladığını söylemek, bulanık üyelik değerler toplamının belirlenen eşik değerinin üzerinde olması demektir. X için bu değeri bulmak için, A'ya ait üyelik değerlerinin toplamının; toplam kayıt sayısına oranlanması gerekmektedir (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 47-48).

2.6.3.3. Nicel Profil Birliktelik Kuralları

Nicel profil birliktelik kuralları yapısı gereği, bir veritabanından elde edilen bir kuralın öncül kısmı bir birime ait özellikleri ve ardıl kısmı bu özellikler ile oluşan olaylar hakkında bilgileri kapsamaktadır. Nicel profil birliktelik kuralları için aşağıdaki kural örnek olarak verilebilir. Burada bir müşteri veritabanından hareketle, elde edilen bir kuralın öncül kısmı müşteri profil bilgileri, ardıl kısmı müşteri davranış biçimleri hakkında bilgileri sağlamaktadır.

Kazanç > 40.000 ⇒ Araba Alış Fiyatı > 80.000 TL (2.22)

İşletme yöneticileri yukarıdaki kuralı "Kazancı 40.000'den fazla olan bir müşteri, fiyatı 80.000 TL'nin üzerinde satılan arabalardan birini satın alabilir" şeklinde yorumlayabilir. İşletme bu şekilde, müşterilere ait kişisel bilgiler ile alışveriş davranışlarını bir havuz da toplayarak, var olan müşteri kitlesine özgü yeni satış stratejileri geliştirebilir (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 47).

2.6.3.4. Nicel Ratio Birliktelik Kuralları

Nicel Ratio Birliktelik Kuralları, bir orana karşılık gelen nitelikler arasındaki bağıntıyı ifade etmektedir. Nicel Ratio birliktelik kuralları sadece iki nitelik ile sınırlı kalmayıp, birden fazla nitelik arasındaki ilişkiden elde edilebilir. Yukarıda nicel profil birliktelik kuralları için verilen

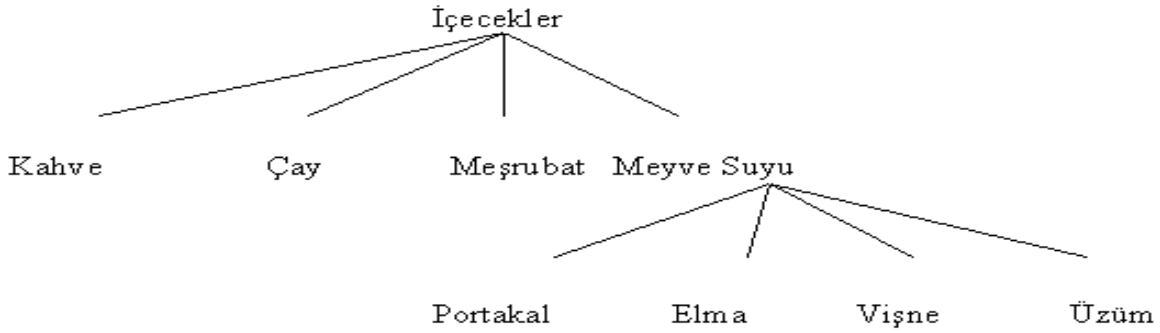
örnek kural oran sınırlamasını karşılamaktadır. Aynı örnek için nicel ratio birliktelik kuralları aşağıdaki gibidir.

Kazanç: Araba Fiyatı = 1:2 (2.23)

İşletme yöneticileri yukarıdaki kuralı, "Bir müşterinin kazancı bir birim iken, iki birimlik bir arabayı satın alabilir" şeklinde yorumlayabilir. Buradaki oran değeri ½'dir (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 47).

2.6.4. Genelleştirilmiş Birliktelik Kuralları

Genelleştirilmiş birliktelik kurallarında veri üzerinde hiyerarşik bir sınıflandırma düzenine dayanarak farklı düzeylerde ve farklı birliktelik kuralları elde etmeye çalışılmaktadır (Gürgen, 2008, s. 26). Genelleştirilmiş birliktelik kuralları için şu örnek verilebilir. Şekil 14'te bir markete ait veriler üzerinde hiyerarşik bir sınıflandırma yapılmıştır. Görüleceği üzere, sınıflandırma yüksek seviyeye sahip ögeleri kapsarken destek ve güven değerleri yükselecektir. Veritabanındaki belirli bir işlem hem aynı hem de farklı ögeler için çoklu sınıflandırma sağlayabilir. $X \Rightarrow Y$ bir genelleştirilmiş birliktelik kuralı ve Y'ye dahil olan hiçbir öge X'e dahil olan ögelerin üstü olmadığı durumlar haricinde düzenli birliktelik kuralı olarak ifade edilir. Bir ögenin üstü tanımı, veritabanındaki bir sınıflandırmada o ögenin üstünde olmasını ifade etmektedir.



Şekil 14: Market Sepet Gruplandırması

Kaynak: DUNHAM, M. H., XİAO, Y., GRUENWALD L. ve HOSSAİN, Z., (2000). *A Survey of Association Rules Teknik Rapor.* Southern Methodist University, Department of Computer Science, TR00-CSE-8. ss.42.

Şekil 14'te verilen örneğe göre, market yöneticileri meşrubat ile ilgili ya da herhangi bir marka veya herhangi bir meşrubat çeşidine özgü birliktelik kuralları elde etmek isteyebilir. Bu durumda, genelleştirilmiş birliktelik kuralları bu ihtiyacı karşılamakta ve bütün birliktelik kurallarının (her bir sınıflandırmanın her bir düzeyi de olmak üzere) elde edilmesini sağlamaktadır (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 41-42).

2.6.5. Çoklu Min-Destek Birliktelik Kuralları

Literatürde genel olarak birliktelik kuralları ile ilgili çalışmalar tek bir minimum destek değeri üzerinden değerlendirilmiştir. Veritabanından çıkarılacak her bir kural için tek bir MDD değerinin kullanılması ile tüm ögeler için aynı frekansın görüldüğü kabul edilmiş olur. Ancak, verideki her öge farklı sıklıklarla görülmektedir. Bu sebeple, tek bir MDD değeri aktifliğini yitirmektedir. Bu problem şu şekilde daha iyi anlaşılabilir. Bir market verisinde ekmek ile peynirin olduğu bir kural için destek değeri %8 ve tam buğday ekmeği ile İsviçre peynirinin olduğu bir kural için destek değeri %3 olsun. İlk kuralın destek değeri yüksek olsa bile, ikinci kuralın destek değerinin düşük olmasına rağmen araştırmacı için ikinci kural daha dikkat çekici olmaktadır (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 49).

Çoklu min-destek birliktelik kurallarında MDD değerinin yüksek ve düşük olmasına bağlı olarak "Nadir Ürün Sorunu" ortaya çıkmaktadır. Bir çalışmada, MDD değerinin çok yüksek tutulması durumunda, nadiren bulunabilecek kuralların elde edilememesine sebep olmaktadır. Sık tekrarlanan ve nadir ögeleri kapsayan kuralları elde edebilmek için MDD değeri düşük tutulabilir. Ancak, bu durumda da anlamlı kurallar ile beraber anlamsız kurallar da ortaya çıkacağı için kural patlaması yaşanacaktır. Bu durum, "Nadir Ürün Sorunu" olarak adlandırılmaktadır. Bu sorunu ortadan kaldırmak adına aşağıdaki iki strateji önerilmiştir:

- Veri setindeki ögeleri güven durumlarına göre birden fazla gruba ayırmak ve her bir grup için ayrı bir MDD değeri atayarak birliktelik kuralları elde etmek,
- Veri setindeki birbiri ile bağıntılı olan nadir görülen ögeleri soyut bir öge olarak gruplandırıp, daha fazla görülmesi sağlanır. Sonrasında ise nicel aralık verisinde birliktelik kuralları bulan algoritma uygulanır (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 49).

Nadir ürün sorunu için yukarıda önerilen iki strateji tüm veritabanı için yetersiz kalabilmektedir. Tüm veritabanı için birden fazla MDD değerinin atanabileceği "MISapriori" algoritması önerilmiştir. Bu algoritmanın özelliği, veritabanındaki her bir öge için ayrı bir MDD değerini atamasıdır. Böylece, nadir görülen ögelerin gereksiz kurallar üretmesine izin verilmeden anlamlı nadir birliktelik kuralları elde edilmiş olur (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 50).

2.6.6. Maksimal Birliktelik Kuralları

Maksimal birliktelik kuralları, niteliklerin oluşturduğu kümeler arasındaki bağıntının daha iyi sunulmasına imkân sağlamaktadır. Maksimal birliktelik kuralları için aşağıdaki örnek verilebilir.

$T_1 = \{Gazete, Tereyağı\}$,

$T_2 = \{Gazete, Tereyağı, Yumurta\}$ olmak üzere,

$$Gazete \Rightarrow Tereyağı = [s = 0{,}40, c = 0{,}50] \tag{2.24}$$

Bir market veri setinden yukarıdaki sonuçlara ulaşılmış olsun. Bu durumda, bulunan birliktelik kuralı her iki işlemde de onay görmektedir. Ayrıca, bu iki ürün için;

$$(Gazete \subseteq Tereyağı, Gazete{:}Tereyağı) \tag{2.25}$$

T_2 işlemi ile bağıntısı olduğunda $(T_2 \cap Tereyağı = Gazete)$ maksimaldir. Her iki ürün maksimal küme niteliğinde ise $Gazete \Rightarrow Tereyağı$ birliktelik kuralı bir maksimal birliktelik kuralıdır (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 51). Yani, araştırılan tüm kayıtların %40'ında müşteriler tarafından gazete ile beraber tereyağı satın alınmıştır. Gazete satışı olan kayıtların %50'sinde aynı alışverişte gazete ile beraber tereyağı satın alınmıştır.

2.6.7. Multimedya Birliktelik Kuralları

Gerçek hayatta teknoloji ve ürünlerinin önemli bir yere sahip olması ile beraber sektörel anlamda birliktelik kuralları multimedya, internet siteleri, iletişim ve e-ticaret alanlarında da tercih edilmeye başlanmıştır. İnternet dünyasında aralarında ilişki kurulacak ögeler fotoğraf, müzik, reklam, video ve elektronik dokümanlardır. Bu ögeler arasında birliktelik kuralları

çıkarılmaya çalışılmaktadır. Aşağıdaki örnek bir multimedya birliktelik kuralı sayılabilir (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 51).

Okyanus ile ilgili bir görüntü^ Boyut büyük ⇒ Renk Mavi (2.26)

Yukarıdaki birliktelik kuralına göre okyanus ile ilgili görüntüler ve büyük boyut nitelikleri arasında ilişki mavi renk sonucunu doğurmaktadır. Multimedya birliktelik kuralları için aşağıdaki kural da örnek olarak verilebilir.

Sabah Kuşağı Programları ^ Kadın İzleyiciler ⇒ Yemek Programları (2.27)

Yukarıdaki birliktelik kuralına göre sabah kuşağı programlarını takip eden kadın izleyiciler yemek programlarını tercih etmektedir.

2.6.8. Zamansal ve Mekânsal Birliktelik Kuralları

Veritabanları dikey (zaman veri) verileri içeriyorsa zamansal birliktelik kuralları ve yatay (mekânsal veri) verileri içeriyorsa mekânsal birliktelik kuralları yöntemleri tercih edilmektedir. Ancak, gerçek hayatta veritabanlarının hem dikey hem de yatay verilerin birlikte olduğu panel (karma) veri tipi şeklinde olabilmektedir. Böyle bir veritabanı için zamansal birliktelik kuralları yöntemi tercih edilebilir.

Mekânsal veritabanları, kayda alınan verilerin konum detaylarını kapsamaktadır. Bunlar; cadde, sokak, enlem ve boylam gibi tüm coğrafik terimleri içermektedir. Mekânsal kayıtlar yanında ve yakınında gibi kavramlar ile veritabanındaki kayıtları oluşturan veri grupları arasındaki ilişkileri açıklamaya çalışmaktadır. Birliktelik kuralları için geçerli olan güven ve destek kavramları mekânsal birliktelik kurallarında da olduğu gibi geçerlidir. Bir mekânsal birliktelik kuralı,

$X \Rightarrow Y$ şeklinde olsun.

Yukarıdaki X ve Y birimleri birer mekânsal öge küme özelliğini taşımaktadır. A ilçesinde bulunan devlet okulları ile ilgili coğrafi nitelikler (göl gibi), özel yapılar ve devletin yaptığı alt yapılar (yol gibi) gibi bilgilerin olduğu bir veri setinden aşağıdaki birliktelik kuralına ulaşılabilir.

$$\text{İ}lkokul\ (T)\text{^}Yakınlık\ (T, Konut\ Geliştirme) \Rightarrow Bitişik(T, Park) \quad (2.28)$$

Yukarıdaki kural, konut geliştirmeye yakın olan alanlarda parka yakın bir okulun olduğunu ifade etmektedir. Ancak, mekânsal birliktelik kuralları, geleneksel birliktelik kurallarına göre bir dezavantaj taşımaktadır. Şöyle ki, market sepet analiz çalışmalarından ayrı olarak mekânsal birliktelik kurallarının geçerliliğini ispatlamak için incelenen kayıt dışındaki diğer kayıtlara da bakmak gerekmektedir. Kayıtlarda, okul ögesinin ayrımının (ilkokul olup olmadığı) görülmesi gerekmektedir. Bu sebeple, mekânsal bir birliktelik kuralındaki yakınında gibi sonuçları yorumlayabilmek için veri seti ve ilgili diğer veri setleri incelenmelidir. Bu durumda, mekânsal verilerin incelenmesi hem maliyetli hem de zor olmaktadır (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 43-44).

2.7.Birliktelik Kuralları Algoritmaları

Birliktelik algoritmaları uygulandığı araştırma problemi, veri türü ve boyutuna göre performans açısından birbirinden ayrılmaktadır. Bu nedenle birliktelik algoritmaları uzmanlar tarafından birbirinden ayrılmıştır. Daha önce bahsedildiği üzere, kural oluşturmak için kullanılan birliktelik algoritmaları ardışık (Sequential) ve paralel (Parallel) algoritmalar olmak üzere 2 ana gruba ayrılmaktadır. Bu gruplara dahil edilen algoritmalar Tablo 26'da gösterilmektedir. Birden çok belleğe sahip olan paralel algoritmalar veri, iş ve diğer paralel algoritmalar olarak üçe ayrılmaktadır. Bu ayrımın sebebi, aday öge kümelerinin sistemde mevcut işlemcilerin hepsine dağılıp dağılmamasından kaynaklıdır. Veri paralel algoritmalar, veritabanındaki tüm bölümler tarafından aynı aday öge kümelerini sayarken, iş paralel algoritmaları öncelikle aday öge kümelerini farklı gruplara böler ve işlemcilere atar. Böylece, her bir bölüm farklı aday öge kümelerini saymış olmaktadır. Yani, bu iki sınıf ilerledikleri teknik açısından birbirinden ayrılmaktadır. Veri ve iş paralel algoritmalar dışında yer alan diğer paralel algoritmalar farklı özelliklere sahiptirler.

Tablo 26: Birliktelik Kuralları Algoritmaları

Ardışık Algoritmalar	Paralel Algoritmalar		
AIS SETM Apriori Apriori-TID Apriori-Hybrid OCD Partitioning Sampling CARMA DIC FP- Growth	**Veri Paralel**	**İş Paralel**	**Diğer**
	CD PDM DMA CCPD	DD IDD HPA PAR	Candidate Distribution SH HD

Kaynak: Yazar tarafından derlenmiştir.

Algoritma seçimi yapılırken en önemli etken olan veri türüne bağlı olarak performansı en yüksek olan algoritma tercih edilmektedir. Birliktelik kuralları analizi için geliştirilen algoritmalar arasında çalışmalarda genel olarak en çok Apriori ve FP-Growth algoritması tercih edilmektedir.

2.7.1. AIS

1993 tarihinde Agrawal, Imielinski ve Swami'nin beraber geliştirdikleri AIS algoritması, veri setinden yaygın öge kümeleri elde etmektedir. AIS, karar destek işlemlerini yerine getirmek için uygun yapıya sahip veritabanlarını geliştirmeye çalışmıştır. Nitel kurallar üretme amacı taşıyan algoritma, veritabanındaki ögelerde alfabetik sıralama koşulunu sağlayarak üreteceği birliktelik kuralları aşağıdaki gibidir:

$$X \Rightarrow I_j | \alpha \tag{2.29}$$

X: Herhangi bir öge küme,

I_j: Tek bir öge ve

α: Birliktelik kuralı güven değeri

Birliktelik kuralının ardıl tarafında görüleceği üzere AIS algoritması tek ögeli kurallar elde etmektedir. Bu durum sadece AIS algoritmasına özgüdür (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 6).

AIS algoritması ile kural üretimi, veritabanında bulunan tüm kayıtların birçok kez taranması prensibine dayanmaktadır. Bu çalışma prensibine göre, birinci taramada her bir öge için destek değerleri hesaplanarak veritabanında ne sıklıkla tekrarlandığı (ne kadar yaygın oldukları) elde edilir. En çok tekrarlanan ögeler (yaygın öge kümeler) bulunur. İki tarama arasında bulunan yaygın öge kümeler arasındaki ortak öge kümeler tespit edilir. Tespit edilen bu ortak öge kümeler işlemde olan diğer ögeler ile bir araya getirilerek aday öge kümeler elde edilir. Yaygın öge küme olan ι sadece işlemdeki büyük olan ve ι'daki ögelerden herhangi birinin ardından gelen ögelerin alfabetik sıralamasında elde edilen ögelerle genişletilir. Bu aksiyon için tahmin ve budama tekniklerine başvurulur. Bu tekniklere göre, aday öge kümelerden gereksiz nitelikler atılır. Elde edilen her bir aday öge kümenin destek değerleri hesaplanarak, MDD değeri ile karşılaştırılır. MDD değerine eşit veya daha büyük destek değerlere sahip kümeler tespit edilir. Bu görevler, daha çok ürün bulunmayana kadar sürmektedir (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 6-7).

2.7.2. Set Yönelimli Madencilik Algoritması (SETM)

1993 tarihinde Houtsma ve Swami'nin beraber geliştirdiği SETM adlı algoritma, yaygın öge kümelerini elde etmek için SQL sorgulama dilinden faydalanmaktadır. SETM algoritmasının SQL dili ile kodlanması ve genel işleyişi Tablo 27'deki gibidir. Kodlarda görüleceği üzere, algoritma iki sıralama ve birer birleştirme-tarama görevlerinin bir araya getirildiği tek bir döngüden oluşmaktadır. Birinci sıralamadan sonra birleştirme-tarama görevi yerine getirilir. Destek değerlerini doğru bir şekilde sıralamak içinde ikinci sıralama başlatılır. Kuralların üretilmesi için Rk değerleri (Ögelere ait destek değerleri) üzerinde kolay bir sıralı taramaya ihtiyaç vardır. MDD değerini sağlayamayan Rk değerleri silinir ve Ck (Yaygın öge kümeler) değerleri kurallar için bilgi sağlamaktadır (Houtsma & Swami, 1993, s. 29-30-31). Ayrıca, burada MDD değeri ne olursa olsun, algoritmanın işlem süresi değişmemektedir. Gerçek bir veri seti için işlem süresi 4-7 saniye olduğu anlaşılmıştır (Houtsma & Swami, 1993, s. 33).

Tablo 27: SQL Kodları ile SETM Algoritmasının Genel İşleyişi

$k := 1;$

$sort\ R_1\ on\ item;$

$C_1 := generate\ counts\ from\ R_1;$

repeat

$k := k + 1;$

$sort\ R_{k-1}\ on\ trans - id, item_1, \ldots, item_{k-1};$

$R'_k := merge - scan\ R_{k-1}, R_1;$

$sort\ R'_k\ on\ item_1, \ldots, item_k;$

$C_k := generate\ counts\ from\ R'_k;$

$R_k\ filter\ R'_k\ to\ retain\ supported\ patterns;$

$until\ R_k = \{\ \}.$

Kaynak: DUNHAM, M. H., XİAO, Y., GRUENWALD L. ve HOSSAİN, Z., (2000). *A Survey of Association Rules Teknik Rapor.* Southern Methodist University, Department of Computer Science, TR00-CSE-8. ss.:31.

SETM algoritması matematiksel anlamda AIS algoritmasında olduğu gibi veritabanını birçok kez taramaktadır. SETM algoritması, AIS algoritmasından farkı ise yaygın öge kümelerindeki her bir nitelik iki değerden meydana gelmektedir. Bu değerler, niteliğin adı ve nitelikleri farklı kılmak için özel kodlardır. Algoritma çalıştırıldığında bu kodlar TID olarak adlandırılmakta ve bunlar saklı tutulmaktadır. Algoritma, aday öge kümeler için bağıntısal birleştirme teknikleri kullanmaktadır. Bu kümeler oluşturulurken, aday öge kümelerini TID bilgisi dahil olmak üzere birer sıralı yedeğini oluşturmaktadır. Daha sonrasında, aday öge kümeler öge adlarına göre sıralanarak, küçük olan öge kümeler silinir. Bu noktada, söz konusu veritabanı TID bilgisine göre sıralanmışsa, yaygın öge kümeleri daha sonra yapılacak taramada $\overline{L_k}$ (yaygın öge kümelerinin her bir niteliği veya elemanı)'yı TID sırasına göre sıralanarak elde edecektir. Küçük öge kümeleri silinir. Bu durumda, veritabanı birden çok kez taranmaktadır. Bu işlem, yaygın öge kümeler bulunamayana kadar devam etmektedir. SETM algoritmasının TID değerlerini depolaması, algoritmada alan karmaşasına sebep olmaktadır. Ek olarak,

veritabanında hem niteliğe ait adlara göre hem de TID değerlerine göre sıralama yapmak, zamanında iyi kullanılmamasına sebep olmuştur (Agrawal & Srikant, 1994, s. 492-493-494).

2.7.3. Apriori

1994 tarihinde Agrawal ve Srikant'ın beraber geliştirdikleri Apriori algoritması, birliktelik kuralları analizi için en verimli algoritmalardan biridir (Agrawal & Srikant, 1994). Günümüzde kullanım alanı artan ve gelişimi devam eden Apriori algoritması, birliktelik kuralları ile ilgili birçok çalışmada tercih edilmiştir. Apriori algoritması kullanılarak gerçekleştirilen çalışmaların bazıları Tablo 28'deki gibidir.

Tablo 28: Apriori Algoritması ile Gerçekleştirilen Çalışmalar

Yıl	Eser ve Açıklama
2007	**A Scalable Algorithm for the Market Basket Analysis**
	Bu çalışma, birliktelik kuralları analizinde hesaplamaları en kısa sürede sağlayacak algoritma seçimi konusu üzerine yapılmıştır. Çalışmada işlem süresine etki eden 3 önemli faktör; işlem sayısı, veri boyutu ve öge sayısıdır. Geleneksel Apriori algoritmasındaki zaman probleminin bu 3 faktörden kaynaklı iken FP-Growth algoritmasının sadece veri boyutu ve öge sayısından etkilendiği ve yeni tanıtılan Smillis algoritmasının ise sadece öge sayısından etkilendiği belirtilmiştir. Bu nedenle, FP-Growth algoritmasının Apriori algoritmasına göre daha iyi olduğu belirtilmiştir. Smillis algoritmasının ise bu iki algoritmaya kıyasla, daha iyi performans sağladığını ifade etmişlerdir (Cavique, 2007).
2014	**Sigortacılık Sektöründe Müşteri İlişkileri Yönetimi için Birliktelik Kurallarının Kullanılması**
	Bu çalışmada, bir sigorta şirketinin veritabanından müşterilerine ait veriler üzerinde apriori algoritması kullanılarak birliktelik kuralları analizi yapılmıştır. Çalışma sonunda müşterilerin ne tür sigorta çalışmalarını tercih ettikleri bilgisinden hareketle, üzerine düşünülmesi gereken müşteri profilleri elde edilmiştir. Ayrıca, müşteriler tarafından sigorta ürünlerinin tercih edilme sıklıkları ile beraber tercih edilen sigorta ürünleri bilgisine ulaşılarak yeni pazarlama stratejileri geliştirilebilir (Buldu, Doğan, & Erol, 2014).
	Duvar İnşa Edilmesinde Verimliliği Etkileyen Faktörlerin Apriori Veri Madenciliği Yöntemi Kullanılarak Analizi
	Bu çalışmada, duvar işçilerinin verimliliğini etkileyen faktörleri tespit etmek için Türkiye'de bu konuda faaliyet gösteren firmalar ile beraber birliktelik kuralları analizi apriori algoritması kullanılarak yapılmıştır. Birliktelik kuralları ekip sayısı, tecrübe ve yaş faktörlerine göre değişim göstermiştir (Kaya & Keleş, 2014).

2015	**Veri Madenciliği Teknikleri Kullanılarak Ortaokul Öğrencilerinin Sosyal Ağ Kullanım Analizi: Kocaeli İli Örneği**
	Bu çalışma, Kocaeli İlindeki bir okulda eğitim gören ortaokul öğrencileri için yapılan bir anket çalışması verileri üzerinde birliktelik kuralları analizi yapılmıştır. Veri setinde öğrencilere ait kişisel bilgiler ile beraber sosyal medya araçları üzerindeki faaliyet bilgileri bulunmaktadır. Bu iki bilgi arasındaki ilişki WEKA programında Apriori algoritması çalıştırılarak incelenmiştir (Duru & Pehlivanoğlu, 2015).
	Trafik Kazalarının Birliktelik Kuralları ile Analizi
	Bu çalışmada, Çankırı'da gerçekleşen trafik kaza verileri üzerinde birliktelik kuralları analizi uygulanmıştır. Apriori algoritması, SPSS Clementine programında çalıştırılarak, kaza yapan sürücülerin çoğunluğunun erkek olduğu bilgisine varılmıştır. Ayrıca, birliktelik kuralları ile kaza yapılan yerin yerleşim içinde/dışında olması, hava durumu, gün ve ay bilgisine göre şekillenmiştir (Akay, Doğrul, & Kurt, 2015).
2017	**Veri Madenciliğinde Birliktelik Kuralları ve İkinci El Otomobil Piyasası Üzerine Bir Uygulama**
	İkinci el piyasasında yaşanan hızlı gelişmelere karşın araştırmacı, MATLAB programı aracığıyla ikinci el arabalarına ait özellikleri keşfetmek için birliktelik kuralları analizi yapmıştır. Çalışma için, 2016 yılı temmuz ayı ve ağustos ayının ilk 3 haftasına kadar olan zaman diliminde bir araba satış sitesinden veriler çekilmiştir. Bu veriler üzerinde Apriori algoritması çalıştırılmıştır. Çalışma esnasında, algoritmada sayısal değişkenleri (vergi, uzunluk, yükseklik gibi) kategorik değişkenlere çevirmede sıkıntılar yaşanmıştır (Özçalıcı, 2017).

Kaynak: Yazar tarafından derlenmiştir.

Birliktelik kuralları analizi çalışmalarında AIS ve SETM algoritmasına göre performansı daha yüksek olan Apriori algoritmasının genel işleyişi Tablo 29'daki gibidir. Buradaki kodlamalar IBM RS/600 530 H iş istasyonu üzerinde yapılan denemelerden elde edilmiştir. Bu algoritma, önceki iki algoritmada da olduğu üzere yaygın öge kümeleri elde ederken, veritabanında birçok kez tarama gerçekleştirmektedir. Öncelikle, çalışma için MDD ve MGD değerleri belirlenmektedir. Veritabanındaki birinci $(k-1)$ taramada, MDD değerini aşan tek ögeli öge kümeler elde edilmektedir. Birinci taramada elde edilen yaygın öge kümelerinden (L_{k-1}) yararlanarak İkinci (k) taramada, apriori-gen fonksiyonu aracılığıyla, aday öge kümeler (C_k) elde edilir. Daha sonra, bu aday kümelere ait destek değerleri hesaplanmaktadır. Bu işlemin kısa sürmesi için, alt küme görevi devreye girmektedir. MDD değerinin aşan aday öge kümeler yaygın öge kümeler olarak kabul edilir. Böylece, bir diğer taramaya geçilir ve elde edilen yaygın öge kümeler önündeki tarama içi aday öge kümeler olarak sayılmaktadır. Bu durum, yeni bir yaygın öge küme bulamayana kadar sürmektedir. Buradaki anlam, bir yaygın öge

kümenin alt kümelerinin de yaygın olacağı yani MDD değerini aşması gerektiği beklentisidir (Agrawal & Srikant, 1994, s. 489).

Tablo 29: Apriori Algoritması

$L_1 = \{large\ 1 - itemsets\}$;

for $(k = 2;\ L_{k-1} \neq \emptyset; k++)$ do begin

$C_k = apriori - gen(L_{k-1})$; // New candidates

foral transactions $t \in \mathcal{D}$ do begin

$C_t = subset\ (C_k, t)$; // Candidates contained in t

foral candidates $c \in C_t$ do

$$c.count_{++}$$

end

$$L_k = \{c \in C_k |\ c.count\ \geq minsup\}$$

End

Answer $= \cup_k L_k$;

Kaynak: AGRAWAL, R. SRİKANT, R. (1994). Fast Algorithms for Mining Association Rules. Proceedings of the 20th VLDB Conference. Santiago, Chile. 487-499 ss.:489.

Apriori algoritması uzmanlar tarafından geliştirilerek, Apriori-TID algoritması geliştirilmiştir. Daha sonra ise, bu iki algoritmanın en iyi özelliklerinin bir araya getirilmesi ile karma bir algoritma olan Apriori-Hybrid algoritması elde edilmiştir.

2.7.4. Apriori-TID

1994 tarihinde, Agrawal ve Srikant tarafından Apriori algoritmasının bir uzantısı olarak geliştirilen Apriori-TID algoritmasının genel işleyişi Tablo 30'daki gibidir. Algoritma, tarama öncesinde apriori-gen (aday üretim) fonksiyonunu kullanmaktadır. İlk aşamada veritabanı ($\mathcal{D}$) bir kez taranmakta ve destek değerleri hesaplanmaktadır. Bunun için, ilk aşamada elde edilen aday öge kümelerinin C_k kodlanmış haline ihtiyaç vardır. İlk tarama sonunda Apriori-TID

Algoritmasında, SETM algoritmasında da olduğu gibi, $\overline{C_k}$ ya ait her bir öge $\langle TID, \{X_k\}\rangle$ biçimindedir. Bu durumda, X_k değeri, TID numaralı kayıtta var olan mevcut yaygın k-öge kümedir. Burada, k=1 iken $\overline{C_1}$, $\mathcal{D}$ veritabanına denk gelir. Fakat, veritabanındaki her bir öge, öge kümesi ile yer değiştirmeye gitmektedir. $\overline{C_1}$'deki destek değerleri hesaplanarak L_1 yaygın öge kümeler elde edilmektedir. $k = 2$ için $\overline{C_2}$'yi üretmek için apriori-gen fonksiyonuna başvurulmaktadır. TID numaralı işlemdeki $\overline{C_k}$ içinde bulunan tüm aday öge kümeler tespit edilir. Burada, ikinci kez veritabanının taranması gereksizdir. $\overline{C_1}$ değeri taranması tercih edilmektedir. Elde edilen $\overline{C_2}$ için destek değerleri hesaplanır ve L_2 yaygın öge kümeler elde edilmektedir. Bu işlem, boş olan aday öge kümeler elde edilene kadar devam etmektedir. Apriori-TID algoritmasında k>1 ise $\overline{C_k}$ değeri Grafik 6'daki onuncu adıma göre algoritma tarafından oluşturulmaktadır. t işleminde olan $\overline{C_k}$'nın bir ögesi $\langle TID,\ c\rangle$ biçimindedir. Bu forma göre c, t işlemi için $\overline{C_k}$'nın bir aday ögesidir. Veritabanındaki herhangi bir işlem k-öge kümesini kapsamıyorsa, $\overline{C_k}$'nın ilgili işlem için bir ögesi bulunmayacaktır. Özetle, TID numarası atanamayan $\overline{C_k}$'nın üye sayısı, (özellikle, k için büyük değerleri söz konusu ise) veritabanındaki kayıt sayısından daha az olmaktadır. Ek olarak, k büyük değerleri söz konusu iken $\overline{C_k}$ için olan her bir kayıt daha küçüktür. Bunun sebebi, çok az sayıda adayın işlemde yer almasıdır. Buna karşılık, küçük k değerlerinde tam tersi durum söz konusudur (Agrawal & Srikant, 1994, s. 490-491-492).

Tablo 30: Apriori-TID Algoritması

$L_1 = \{large\ 1-itemset\};$
$\overline{C_1} = database\ \mathcal{D};$
for $(k = 2;\ L_{k-1} \neq \emptyset; k++)$ **do begin**
$C_k = apriori - gen\ (L_{k-1})$; // New candidates
$\overline{C_k} = \emptyset;$
For all entries $t \epsilon \overline{C_{k-1}}$ **do begin**
// determine candidate itemsets in C_k contained
// in the transaction with identifier t.TID
$C_t = \{c \in C_k | (c - c[k]) \in t.set - of - itemsets\ \hat{}\ (c - c[k-1]) \in t.set - of - itemsets\};$
For all candidates $c \in C_t$ **do**
c.count++;
if $(C_t \neq \emptyset)$ ***then*** $\overline{C_k} += < t.TID, C_t >;$
end
$L_k = \{c \in C_k | c.count \geq minsup\}$
end
Answer $= \cup_k L_k$;

Kaynak: AGRAWAL, R. SRİKANT, R. (1994). Fast Algorithms for Mining Association Rules. Proceedings of the 20th VLDB Conference. Santiago, Chile. 487-499 ss.:491.

2.7.5. Apriori-Hybrid

1994 tarihinde Agrawal ve Srikant tarafından Apriori ve Aprirori-TID algoritmalarının karması şeklinde geliştirilen Apriori-Hybrid algoritması, her bir işlemde aynı algoritmanın kullanılmasının zorunlu olmadığını savunan bir algoritmadır. Buna göre, apriori ve apriori-TID algoritmaları başarılı olduğu noktalara göre tercih edilmesi önerilmektedir (Agrawal & Srikant, 1994, s. 496).

Grafik 6: Apriori-TID (T10.14.D100K, MDD =0.75 %)

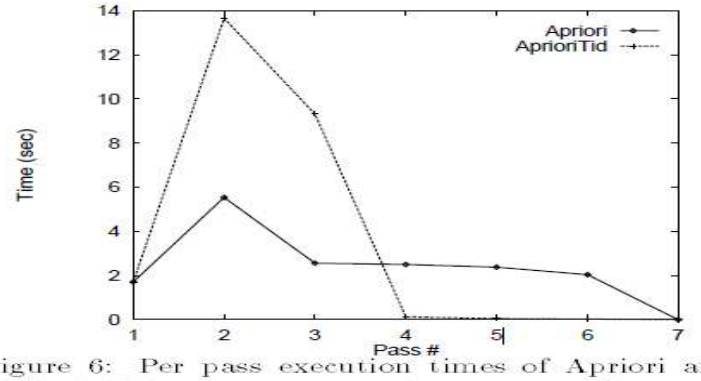


Figure 6: Per pass execution times of Apriori and AprioriTid (T10.14.D100K, minsup = 0.75%)

Kaynak: AGRAWAL, R. SRİKANT, R. (1994). Fast Algorithms for Mining Association Rules. Proceedings of the 20th VLDB Conference. Santiago, Chile. 487-499 ss.:496.

Grafik 6'da algoritmanın geliştiricilerinin ilgili çalışmalarındaki veritabanı üzerindeki her bir geçiş için Apriori ve Apriori-TID algoritmalarına ait süreler gösterilmektedir. Buna göre, ilk aşamasında Apriori ve kalan aşamalarda ise Apriori-Hybrid algoritması daha iyi sonuç vermektedir. Her iki algoritma aynı aday üretme fonksiyonunu kullanmaktadır. Ancak, destek değerlerini hesaplamak için Apriori algoritması her aşamada veritabanının tümünü incelemeye devam etmektedir. Oysa, Apriori-TID algoritması bir kez veritabanını taradıktan sonra veritabanı yerine $\overline{C_k}$ taramayı tercih etmektedir. Böylece, veritabanı yerine daha küçük boyuttaki $\overline{C_k}$ bütünü incelenmektedir. Bellek tarafından $\overline{C_k}$ parçasının karşılanıp karşılanmayacağını anlamak için her bir aşamada tahmin edilmektedir (Agrawal & Srikant, 1994, s. 496).

2.7.6. Sıra Dışı Ürün Kümesi Tespiti Algoritması (OCD)

1994 tarihinde Manilla ve ark. tarafından geliştirilen OCD algoritmasına göre yaygın öge kümeleri tespit etmek için, veri setinden çekilen küçük örneklemler ile daha yüksek performans sağlanabilir. Bu algoritma, gereksiz olan aday öge kümeleri tespit etmek ve ortadan kaldırmak için daha önceden gerçekleştirilen geçişlerde kazanılan bilgilerin birleştirme analizi skorlarına başvurmaktadır. Bir $X \subseteq R$ alt kümesinin yaygın olup olmadığını tespit etmek için MDD değeri $\mathcal{s}$ iken, ilişkilerin minimum $(1-\mathcal{s})$ kadar işlemi incelenmesi gerekmektedir. Buda en düşük $\mathcal{s}$ değeri için, veritabanından elde edilen tüm ilişkilerin incelenmesini göstermektedir. Yüksek boyutlu veritabanlarında en az sayıda geçişin yapılması önem kazanmaktadır (Mannila, Toıvonen, & Verkamo, 1994, s. 183-184).

Tablo 31: OCD Algoritması

$$C_{\mathcal{s}+1} = \{X \subseteq R || X| = \mathcal{s}+1; X, L_{\mathcal{s}}' nin\ (\mathcal{s}+1)\ elemanlarını\ kapsar\}$$

$C_1 = \{\{A\}|A \in R\};$
$\mathcal{s} = 1;$
while $C_{\mathcal{s}} \neq \emptyset$ ***do***
$database\ pass$: $let\ L_{\mathcal{s}}\ be\ the\ elements\ of\ C_{\mathcal{s}}\ that\ are\ covering;$
$candidate\ generation$: $compute\ C_{\mathcal{s}+1}\ from\ L_{\mathcal{s}};$
$\mathcal{s} := \mathcal{s}+1;$
od;

Kaynak: MANNILA, H; TOIVONEN, H. ve VERKAMO, I.A. (1994). Efficient Algorithms for Discovering Association Rules. AAIWS'94 Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining. AAAI. Seattle, WA. 181-192. ss.:184.

OCD algoritmasının genel işleyişi Tablo 31'deki gibidir. AIS algoritması veritabanını birden çok kez taramaya tabi tutarken, OCD algoritması çok farklı bir yaklaşım izlemektedir. Tablo 31'de görüleceği üzere, gereksiz aday öge kümeleri ortadan kaldırılması için önceki geçişlerde kazanılan tüm bilgiler kullanılmaktadır. Algoritma, $\mathcal{s}$ boyutundaki tüm yaygın öge kümeleri kapsayan bir $L_{\mathcal{s}}$ kümesi üretmektedir. $L_{\mathcal{s}}$ kümesinde var olan yaygın öge kümeleri kapsayan ve $L_{\mathcal{s}+1}$'de olabilecek $(s+1)$ boyutlu aday kümeler $C_{\mathcal{s}+1}$ dahildir. L_k'nın bir uzantısı olan e, $X \in L_{\mathcal{s}+e}$ iken en az sıfıra eşit veya daha büyük ise X, $L_{\mathcal{s}}$'den $\binom{\mathcal{s}+e}{\mathcal{s}}$ kadar kümeyi

kapsamaktadır. Bu durumda, C_{k+1} aşağıdaki formüle göre hesaplanmaktadır (Mannila, Toıvonen, & Verkamo, 1994, s. 184-185).

2.7.7. Bölümleme Algoritması

1995 tarihinde Savasere ve ark. tarafından geliştirilen bölümleme algoritması, veritabanında sadece iki kez tarama gerçekleştiren algoritma, veritabanını küçük gruplara ayırıp, her bir grubun ana hafızada saklanacağını savunmaktadır. Burada gruplara ayırma işleminin sebebi, hafızada dolu olan alanı düşürüp, daha kısa sürede işlemi bitirmeyi sağlamaktır. Bu sebeple, bölümleme tekniği adı da verilmektedir. Bölümleme algoritmasında kullanılan parametreler Tablo 32'de ve genel işleyişi Tablo 33'te gösterilmektedir. Tablo 33'te yer alan kodlamalar 150 MHZ'lik bir saat hızı ve 32 Mbyte'lık ana bellek ile bir Silicon Graphics Indy R4400SC iş istasyonu üzerinde denenmiştir (Omiecinski, Navathe, & Savasere, 1995, s. 434).

Tablo 32: Bölümleme Algoritması İçin Göstergeler

C_k^p	Yerel öge kümeler
L_k^p	Yerel yaygın öge kümeler
L_p	Tüm yerel yaygın öge kümeler
C_k^G	Genel öge kümeler
C^G	Tüm genel öge kümeler
L_k^G	Tüm genel yaygın öge kümeler

Kaynak: SAVASARE, NAVATHE, OMIECINSKI. (1995). An Efficient Algorithm for Mining Association Rules in Large Databases. VLDB '95 Proceedings of the 21th International Conference on Very Large Data Bases. San Francisco. Morgan Kaufmann Publishers Inc. S.4324-444. ss.:435.

Bölümleme algoritması, öncelikle veritabanı (P) üzerinde tarama gerçekleştirir ve veritabanı x adet gruba ayrılır. Her bir grup için yerel yaygın öge kümeler elde edilmektedir. Yerel yaygın öge kümeler arasından veritabanındaki gruplardan en az birinde var olan şartı sağlayan kümeler birleştirilir ve elde edilen süper yaygın öge kümelerin gruplarda yaygın olmasına karşın tüm veritabanı içinde de yaygın olup olmadığı sorgulanır. İkinci taramada, tüm veritabanında yaygın olan öge kümelere ait destek değerleri hesaplanarak, veritabanı için tüm yaygın öge kümeler elde edilmektedir (Omiecinski, Navathe, & Savasere, 1995, s. 435). Burada görüleceği üzere,

bu algoritmanın en önemli avantajı yaygın öge kümeleri sayma işleminde ana hafızayı kullanması ve giriş/çıkış maliyetini düşürmesidir (Karabatak M. , 2018, s. 35).

Tablo 33: Bölümleme Algoritması

$P = partition\_database(D)$
$x =$ Number of partitions
$for\ i = 1\ to\ k\ begin$ // *Phase I*
$read\_in\_partition\ (p_i \epsilon P)$
$L^i = gen\_large\_itemset(p_i)$
end
$for\ (i = 2; L_j^i \neq \emptyset, j = 1,2, \dots, k; i++)$ do
$C_i^G = U_{j=1,2,\dots,n} L_i^j$ // *Merge Phase*
$for\ i = 1\ to\ k$ begin // *Phase II*
$read\_in\_partition\ (p_i \epsilon r)$
$for\ all\ candidates\ c \epsilon C^G\ gen\_count(c, p_i)$
end
$L^G = \{c \in C^G | c.count \geq minsup\}$

Kaynak: SAVASARE, NAVATHE, OMIECINSKI. (1995). An Efficient Algorithm for Mining Association Rules in Large Databases. VLDB '95 Proceedings of the 21th International Conference on Very Large Data Bases. San Francisco. Morgan Kaufmann Publishers Inc. 432-444. ss.:435.

2.7.8. Örnekleme Algoritması

Çok büyük veritabanlarında kurallar üretilirken, tarama sayısı artmaktadır. Toivonen, bu durumun veri madenciliği uygulamasının etkinliğini düşürdüğünü savunmaktadır. Bu probleme çözüm olarak 1996 tarihinde örnekleme algoritmasını önermiştir. Böümleme algoritması veritabanındaki geçişleri indirgemek ve çok sayıda anlamlı kurallar elde etmek için veritabanını ana bellekte çalışabilecek kadar gruplara ayırmaktadır. Ancak, aynı sorunu gidermek amacıyla Sampling algoritması farklı bir yol izlemektedir. Örnekleme algoritması ile net birliktelik kuralları elde edilmektedir. Olasılıksal bir yaklaşıma sahip olan algoritma, veritabanındaki bütün kuralları üretemediği anlarda geri kalan kurallar için ikinci bir taramaya ihtiyaç duyabilmektedir (Toivonen, 1996, s. 134).

Örnekleme algoritmasının genel işleyişi şu şekildedir. Veritabanından ana bellek için yeterli olacak şekilde rasgele bir örneklem çekilir. Bu örnek için yaygın öge kümeler elde edilmektedir. Negatif kenar fonksiyonu aracılığıyla bulunan yaygın öge kümeler üzerinden tüm veritabanı için geçerli olan adaylar tespit edilir. Aslında, bu fonksiyon Apriori algoritmasında kullanılan gen fonksiyonunun genelleştirilmiş halidir. Bu iki fonksiyon arasındaki en temel fark

apriori_gen fonksiyonunun tek boyut ile negatif kenar fonksiyonunun ise farklı boyutlardaki öge kümeler ile ilgilenmesidir. Aday öge kümeler tespit edildikten sonra, veritabanı için kesin aday küme sayısını tespit etmek için veritabanı bir kez daha taranmaktadır. Tüm yaygın öge kümelerin adaylar arasında ve geri kalan kısımdaki öge kümeler yeni yaygın öge küme olarak ifade edilemediğinde tüm yaygın öge kümeler bulunmuş anlamına gelir. Algoritma, bu durumda sonlandırılmaktadır. Fakat, algoritmada yeni öge kümelerin var olup olunmadığından emin olunamıyorsa, bir kez daha taramaya ihtiyaç duyulmaktadır (Toivonen, 1996, s. 136-137).

2.7.9. Dinamik Ürün Kümesi Sayımı Algoritması (DIC)

Dinamik Ürün Kümesi Sayımı (Dynamic Itemset Counting/ DIC) algoritması, veritabanı üzerinde gerçekleştirilen tarama sayısını indirgemek amaçlı geliştirilen bir ardışık algoritmadır. Algoritmanın genel işleyişi Tablo 34'te gösterilmektedir. Bu algoritma, bölümleme algoritması gibi ortak homojen bir dağılım sağlamaktadır. Bu algoritmanın mekanizması öge kümeleri tarama öncesi meydana getirip ve sayma işlemini gerçekleştirerek tarama sayısını azaltmaya çalışmaktadır. Veritabanında bulunan kayıtlar aralıklı olarak gösterilmekte ve ardışık düzende taranmaktadır (Brin, Motwani, Ullman, & Tsur, 1997, s. 255).

Tablo 34: DIC Algoritması

$$
\begin{array}{l}
Result = \emptyset \\
k := 1; \\
C_1 = set\ of\ all\ 1-itemsets; \\
while\ C_k \neq \emptyset\ do \\
\quad create\ a\ counter\ for\ each\ itemset\ in\ C_k; \\
\quad forall\ transactions\ in\ database\ do \\
\qquad\qquad Increment\ the\ counters\ of\ itemsets\ in\ C_k \\
which\ occur\ in\ the\ transaction; \\
\quad L_k := All\ candidates\ in\ C_k \\
which\ exceed\ the\ support\ threshold; \\
\quad Result := Result \cup L_k; \\
\quad C_{k+1} := all\ k+1-itemsets \\
\qquad\qquad which\ have\ all\ of\ their\ k-item\ subsets\ in\ L_k. \\
\quad k := k+1; \\
end
\end{array}
$$

Kaynak: BRIN, S. MOTWANI, R; ULLMAN, J.D; TSUR, S. (1997). *Dynamic Itemset Counting and Implication Rules for Market Basket Data.* SIGMOD '97 Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data. *26,2. Tucson, Arizona, USA255-264. ss.:256.*

Birliktelik kurallarını elde etmek için kullanılan algoritmalarda genel olarak iki önemli faktör bulunmaktadır. Bunlar tarama sayısı ve yapılan taramaların verimliliğidir. DIC algoritması bu

iki faktörü şu şekilde sağlamaktadır. Algoritma, ilk taramayı gerçekleştirerek 1-öge kümelerini bulmaktadır. Tespit edilen 1-öge kümelere ait destek değerlerini hesaplamaktadır. MDD değerini aşan öge kümeler 2-öge kümeler olarak adlandırır. İkinci taramaya geçişte hem 1-öge kümeler hem de 2-öge kümeler taranarak destek değerleri hesaplanır. Bu tarama sonunda 3-öge kümeler elde edilmektedir. Bu çalışma veritabanının sonuna ulaşılana dek devam etmektedir. Son taramadan sonra, veritabanı hiç sayılmayan öge kümeler için taranır ve algoritma çalışması bu şekilde sonlandırılarak kurallar elde edilmektedir (Brin, Motwani, Ullman, & Tsur, 1997, s. 255-256).

2.7.10. Sürekli Birliktelik Kuralı Madenciliği Algoritması (CARMA)

Ardışık algoritmalardan biri de yaygın öge kümelerinin çevrimiçi olarak hesaplanmasını sağlayan Sürekli Birliktelik Kuralı Madenciliği (Continuous Association Rule Mining Algorithm/ CARMA) algoritmasıdır. CARMA, veritabanı üzerinde tüm yaygın öge kümeleri elde edebilmek için en fazla iki taramaya izin vermektedir. Bu algoritma, ilk tarama gerçekleştirilirken, herhangi bir zamanda MDD değerini değiştirme imkânı sağlamaktadır (Hidber, 1999, s. 145).

CARMA algoritmasının genel işleyişi şu şekildedir. Veritabanı üzerinde birinci tarama gerçekleştirilirken, veritabanındaki mümkün tüm öge kümeler ve bunlar için bir bölüm oluşturulmaktadır. Bu esnada, MDD değeri kullanıcı tarafından değişeme tabi tutulabilir. CARMA, ilk taramada yapılan işlemler bakımından DIC algoritmasına benzemektedir. İkinci taramada ise, oluşturulan bölümdeki her bir öge küme için destek değeri hesaplandıktan sonra tüm yaygın öge kümeler tespit edilmektedir. Yaygın olmayan öge kümeler ise söz konusu bölümden çıkarılmaktadır. CARMA algoritması ile birlikte bu şekilde birliktelik kurallarına ulaşılmaktadır (Hidber, 1999, s. 145-146).

2.7.11. FP-Growth

Birliktelik kuralları analizi için sıklıkla başvurulan popüler algoritmalardan biri de 2000 tarihinde Han ve ark. tarafından geliştirilen FP-Growth algoritmasıdır. Bu algoritma için yapılan çalışmalara göre büyük yaygın öge kümelerin madenlenmesi için verimli ve ölçeklenebilir yapıdaki algoritmanın Apriori algoritmasına göre hız performansının daha iyi olduğu görülmüştür (Han & Kamber, 2001, s. 150). Aslında, Apriori algoritmasının

performansını arttırmak için geliştirilmiş olan FP-Growth algoritması kompakt bir veri yapısının geliştirilmesiyle oluşturulmuştur (Győrödi, Győrödi, & Holban, 2004, s. 213). Algoritma, aday üretimi olmaksızın sık tekrarlanan adaylar setini madenleyen sık-desen büyümesi algoritmasıdır (Han & Kamber, 2006, s. 243).

FP-Growth algoritmasının genel işeyişi Tablo 35'te gösterilmektedir. Yüksek boyuta sahip veritabanlarında aday öge kümeler üretiminde sürekli veritabanını taramaya gerek duymayan algoritma hem çok hızlı çalışmakta hem de sistem araçlarını etkin bir şekilde kullanmaktadır Bunun temel sebebi, algoritmanın Frequent Pattern Tree (FP-Tree) ile isimlendirilen sıkıştırılmış bir ağaç veri yapısı içinde veritabanını saklı tutmasıdır (Győrödi, Győrödi, & Holban, 2004, s. 220). Sonrasında bu veritabanını, her biri sık öge veyahut desen parçası ile ilişkilendirilen bir şartlı veritabanına (yeni bir veritabanı yapısı olan yansıtılmış veritabanı) böler ve yeni bir yapıya sahip bu veritabanları üzerinde bağımsız bir şekilde madencilik çalışması gerçekleştirmektedir (Han & Kamber, 2006, s. 243). Ayrıca, veritabanını sadece iki kez taramak algoritma için yeterli olmaktadır (Győrödi, Győrödi, & Holban, 2004, s. 220). İlk taramada tüm ögelere ait destek değerlerini bulunur. İkinci taramada FP-Tree adındaki ağaç yapısını kurmaktadır (Győrödi, Győrödi, & Holban, 2004, s. 213). Ancak, veritabanını iki kez taramak algoritmanın verimliliğini düşürmektedir. Ayrıca, algoritmanın madencilik sürecinde aday öge kümelerini oluşturmaması çalışmanın verimliliğini ciddi anlamda arttırmaktadır (Liu, Yin, Zeng, & Zhang, 2015, s. 1).

Tablo 35: FP-Growth Algoritmasının Genel İşleyişi

```
Algoritma FP-Growth (VT, mindestek)
        Boş liste tanımla: F [];
        foreach Hareket H_i in VT do
                foreach Nesne n_j in H_i do
                        F[n_j] + +;
                end
        end
        foreach Nesne n inF do
                if F[n] < mindestek then
                    n nesnesini F listesinden sil
                    end
        end
        Sırala F [];
        FPtree ağaç yapısının kök düğümünü tanımla: kök;
        foreach Hareket H_i in VT do
                    H_i kaydını F listesine göre sırala;
                    AgacaEkle(H_i, kök);
        end
        foreach nesne n_i in N do
                    Growth (kök, n_i, mindestek);
        end
End
```

Kaynak: Birant, D., Kut, A., Altınok, B., Altınok, H., Altınok, E., Ihlamur, M. ve Ventura, M. (2010). İş Zekâsı Çözümleri İçin Çok Boyutlu Birliktelik Kuralları Analizi. Akademik Bilişim'10- Xıı. Akademik Bilişim Konferansı Bildirileri. Muğla Üniversitesi. ss.:260.

Veritabanı üzerinde iki tarama gerçekleştiren algoritma, ilk taramasında sık tekrarlanan 1-öge kümeleri keşfeder ve bunlara ait destek değerlerini hesaplamaktadır. FP-Growth bu aşamada Apriori ile benzerlik taşımaktadır (Han & Kamber, 2006, s. 243). Ögelere ait destek değerleri, algoritma için belirlenen MDD değerine göre karşılaştırılır. Bu değere eşit veya büyük olan destek değerlerine sahip ögeler tespit edilir. Bu ögelere ait destek değerleri büyükten küçüğe doğru sıralama ile bir listeye atanır. Bu seçme işlemi ile yaygın olmayan ögelerin ağaca dahil olması önlenmektedir. Veritabanında var olan her bir kayda ait ögeler sahip oldukları destek değerine göre aynı şekilde sıralanmaktadır. Ağaç yapısının oluşturulması için öncelikle "Root" adı verilen yeni bir düğüm üretilir. Sonrasında her işlem sıralı olmak kaydıyla ağaca dahil edilir. Bu adım şu şekilde sağlanmaktadır. İlgili kayıtta var olan bir öge ağaca dahil değilse ilgili öge için yeni bir düğüm üretilir. Bunun için destek değeri 1 olarak atanarak ögeler bir arada tutulması sağlanır. Ancak ilgili öge önceden üretilmişse, yalnızca o düğümün destek değeri 1 değerinde arttırılmaktadır. Ayrıca, oluşturulan düğümler arasındaki bağıntıyı korumak için "Header Table" adında bir başlık tablosu oluşturularak, her bir düğüme ait başlangıç tepesi tespit edilerek, ağaç içindeki aynı düğümler birbirine bu noktalar ile birbirine bağlanmaktadır (Abul, Özdoğan, & Yazıcı, 2009, s. 8-12). FP-Tree kurulduktan sonra bu yapı üzerinde algoritma çalıştırılır. Algoritma, ilk olarak en az tekrarlan ögeden itibaren geçtiği dallar tespit edilir. Tek bir dal olması durumunda yaygın öge kümeler dalı oluşturan ögelerin kombinasyonuna eşittir. Birden çok dal olması durumunda destek değeri olarak ilgili daldaki en küçük destek değeri olarak belirlenir. Sonrasında bu dallar ilgili öge için şartlı örüntü temelini oluşturmaktadır. Her bir şartlı örüntü temelinden şartlı örüntü ağacı oluşur. Bu şartlı örüntü ağacı üzerinde FP-Growth algoritması özyinelemeli olarak tekrar çalıştırılmaktadır. Tabloya dahil olan her bir öge için bu işlem tekrarlanarak, sık öge kümeler tespit edilmektedir (Han, Pei, & Yin, 2000, s. 45).

FP-Growth algoritmasına özgü çok fazla farklı araştırma konuları mevcuttur. Örneğin; FP-ağaç yapısının SQL tabanlı bir çalışmada yüksek ölçüde ölçeklendirilmesi veya FP ağaçlarına başvurulan sık sık oluşturulan kalıpların şartlı madenciliğinin ve uzantılarının genişletilmesi ve uygulanması olabilir. Bunun dışında sıralı desenler, max-desenler, kısmi periyodiklik ve diğer ilginç sık desenler üzerinde veri madenciliği çalışmaları için FP-ağaç tabanlı madencilik yönteminden yararlanılabilir (Han, Pei, & Yin, 2000, s. 12).

2.7.12. Tahminci-Apriori Algoritması

2001 tarihinde Scheffer tarafından geliştirilen Tahminci-Apriori algoritması, mevcut Apriori algoritmasının geliştirilmiş bir halidir (Scheffer, 2001, s. 1). Her iki algoritmada destek değerlerini baz alarak veritabanında arama yapmaktadır. Destek değerlerinin alt yönde kapanmasından yararlanarak üst arama kısmında olası ilişkilendirme kurallarının budanmasını gerçekleştirir. Kuralların derecelendirilmesi için Apriori algoritması güveni, tahminci-apriori algoritması ise destek değerlerini baz almaktadır. Apriori algoritması budama sürecinde, öge kümelere ait destek değerleri hesaplanır ve yaygın öge kümeler elde edilir. Daha sonrasında elde edilen kurallar MGD değerine göre incelenmektedir. Tahminci-Apriori algoritması ile yüksek destek değerli ya da güven değerleri çok yüksek ve destek değerleri çok düşük kurallar elde edilebilir. Bu sebeple, algoritmada ilginçlik ölçütü olarak MGD ve MDD yerine doğruluk değeri atanmaktadır. Araştırmacı tarafından birliktelik kuralı sayısı (n) belirlenerek, n sayıda kural elde edilmektedir (Mutter, 2004, s. 15-16).

2.7.13. Tertius

2001 tarihinde Flach ve Lachiche tarafından geliştirilen Tertius algoritması, 7500 satırlık C kodundan oluşan yukarıdan aşağıda doğru bir kural keşif sistemine sahiptir (Flach & Lachiche, 2001, s. 75). Tertius algoritması, n tane en fazla kabul edilen hipotezin belirlenmesinde, optimum birinci değer arama yapısını kullanan bir makine öğrenme sistemidir. Bu özellik, kuralların en güncel ve kullanılabilirlik değerlerinin dengelenmesini gerektirmektedir. Tarama esnasında tekrar etme problemi yaşanmaması için "Yedeksiz Arıtma Operatörü" kullanmaktadır (Flach & Lachiche, 2001, s. 76-77). Veri analizi için Wickens'in düzenlediği olasılık tablosundan faydalanmaktadır. Gözlenen ve beklenen frekanslar arası ilişkinin incelenmesi için Pearson Ki-Kare tekniğine başvurulmaktadır (Flach & Lachiche, 2001, s. 64).

2.7.14. Sayım Dağılımı Algoritması (CD)

Paralel algoritmalar, budama ve sayma teknikleri bakımından birbirine göre farklılıklar taşımaktadırlar. CD, bir veri paralellik algoritmasıdır. Algoritmanın genel işleyişi Tablo 36'da gösterilmektedir. Algoritmanın sisteminde, superscript işlemci sayısını ve altscript aday boyutunu temsil etmektedir. Algoritma, veritabanını $\{D^1, D^2, \dots, D^p\}$ şeklinde gruplara ayırmaktadır. Bu grupları da n adet işlemciye dağıtmaktadır. İşlem kısmı genel olarak 3

adımdan oluşmaktadır. İlk adımda, her bir D^i yerel veritabanındaki C_k aday öge kümelere ait destek değerleri hesaplanır. İkinci adımda, tüm aday öge kümelere ait genel destek değerlerini hesaplamak amacıyla tüm aday öge kümelerine ait yerel destek değerlerini birbiri ile değiştirir. Son aşamada ise genel veritabanı için yaygın öge kümeler L_k şeklinde tanımlanmakta ve n adet işlemcinin her biri birbirinden bağımsız olmak kaydıyla L_k üzerinde apriori_gen fonksiyonunu uygulayarak k+1 büyüklükte aday öge kümeler oluşturulmaktadır. Algoritma, daha fazla aday bulamayana dek bu 3 adımı tekrar etmektedir (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 25-26).

Tablo 36: CD Algoritması

Input:
I, s, $D^1, D^2, \ldots, D^p$

Output:
L

Algorithm:

1) C_1 =I;
2) **for** $k = 1$; $C_k \neq \emptyset$; $k++$ **do begin**
//step one: counting to get the local counts
3) count(C_k, D^i); //local processor is i
//step two: exchanging the local counts with other processors
//to obtain the local counts in the whole database.
4) **forall** itemset $X \in C_k$ **do begin**
5) X.count=$\sum_{j=1} p \{X^j.count\}$;
6) **end**
//step three: identifying the large itemsets and
//generating tha candidates of size k+1
7) $L_k = \{c \in C_k | c.count \geq s \times |D^1 \cup D^2 \cup \cdots \cup D^p|\}$;
8) $C_{k+1} = apriori\_gen(L_k)$;
9) **end**
10) return L= $L_1 \cup L_2 \cup \ldots \cup L_k$

Kaynak: DUNHAM, M. H., XİAO, Y., GRUENWALD L. ve HOSSAİN, Z., (2000). *A Survey of Association Rules Teknik Rapor.* Southern Methodist University, Department of Computer Science, TR00-CSE-8. ss.:26.

2.7.15. Paralel Veri Madenciliği Algoritması (PDM)

Veri paralel algoritmalar arasında yer alan Paralel Veri Madenciliği Algoritması (Parallel Data Mining/ PDM) algoritmasının genel işleyişi Tablo 37'de gösterilmektedir. Algoritma, aday öge kümelerinin paralel üretimi ve yaygın öge kümelerinin paralel tespit edilmesinden oluşmaktadır. Algoritma, erken geçişlerden aday ögeleri üretmek için "Hash" adında karma bir tablodan yararlanmaktadır. Gelecek geçişlerde bir önceki yaygın öge kümelerden doğrudan aday kümeleri üretecek bir yöntem kullanmaktadır (Chen, Park, & Yu, 1995, s. 32). Bu yönüyle PDM, "Hashing" yani doğrudan karışma yöntemi ile CD algoritmasının sentezlenmiş halidir (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 26).

Tablo 37: PDM Algoritması

P-1) Each node generates L_{1i} and H_{2i} by scanning D_{1i}.
P-2) Determine L_1 by exchanging L_{1i} through
all-to-all broadcast.
P-3) Call procedure Clue-and-Poll to find $\{x|H_2[x] \geq s\}$
from all H_{2i}'s.
P-4) k=2
P-5) While ($L_{k-1} \geq$ k) do
P-6) If (k=2), each node generates C_{2i} from L_1 and
$\{x|H_2[x] \geq s\}$,
P-7) Otherwise, each node generates C_{ki} directly from L_{k-1},
P-8) Determine $C_k = \bigcup_{i=0}^{m} p^{-1}\, C_{ki}$ by exchanging C_{ki}
among all nodes.
P-9) Each node generates L_{ki} by scanning and trimming D_{ki}.
P-10) Determine L_k by exchanging L_{ki} among all nodes.
P-11) k=k+1.

Kaynak: CHEN, M.S., PARK, J.S. ve YU, P. (1995). Efficient Parallel Data Mining for Association Rules. Proceedings of the International Conference on Information and Knowledge Management. Baltimore, Maryland. 31-36.ss.:33

Hash tekniği ile Apriori algoritmasında L_1'den C_2'nin üretilmesinde hiçbir budama uygulamadığı için ikinci geçişte kullanışlı olmaktadır. PDM algoritması, birinci geçişte 1-öge kümeleri saymakla beraber 2-öge kümelerin sayılarını kayıtlı tutmak için Hash tablosuna başvurmaktadır. Bu tabloda, 2-öge kümelerin aslını saklamak gerekmemektedir. Ancak, her bir bölüm için sayım yapılır ve bu bilgi saklı tutulur. Örneğin; $\{X, Y\}$ ve $\{Z\}$ yaygın öge kümeler olsun. Hash tablosunda 2-öge kümeler için $\{XY, ZT\}$'yi kapsayan bölümün küçük olduğunu varsayalım. Yani, bölüme ait sayının MDD değerinden daha düşük olsun. PDM algoritması Hash yöntemi ile birlikte AB ikilisini aday küme olarak üretmezken, ikinci geçiş sırasında Apriori algoritması bu ikiliyi aday küme olarak üretebilir. Bunun sebebi, ilk geçiş sırasında 2-öge kümeler hakkında bilgiye ulaşılamaz. k. geçiş sırasında iletişimi sağlamak için algoritma, Hash tablosundan k+1 öge kümeleri yerel destek sayıları ile beraber k-öge kümelerin destek sayılarını değiştirmek zorunda kalmaktadır (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 26-27).

2.7.16. Dağılım Madenciliği Algoritması (DMA)

Dağılım Madenciliği (Distributed Mining Algorithm/ DMA) algoritması, budama ile iletişim mesajı indirgeme yöntemlerinin dahil edilmesiyle veri paralellik sistemine sahip bir yöntemdir. Algoritma, bir öge kümenin hem veritabanının bölümlerinin herhangi birinde hem de tüm

veritabanı için yaygın olup olmadığını tespit etmek ile beraber her bir işlemci için yaygın öge kümelerin destek değerlerinden yararlanarak, yaygın öge kümeler oluşturmaktadır. Örneğin; X, 1 numaralı işlemcide ve Y, 2 numaralı işlemde yaygın olduğunu düşünelim. Yani, A ve B kendi işlemcilerinde yerel olarak yaygın öge küme olmakla beraber tüm veritabanı için genel olarak yaygın öge kümelerdir. AB 2-öge küme DMA algoritmasında üretilemezken, Apriori algoritmasında üretilebilir. Çünkü, Apriori algoritması A ve B'nin işlemcilerindeki yerel destek değerlerini dikkate almamaktadır. CD algoritması, iletişim için tüm aday kümelerin yerel destek değerlerini yayınlamaktadır. DMA algoritması aksine, tüm aday kümelerin yerel destek değerleri bir seçim bölümüne götürür. Böylece, mesajın boyutunu $O(p^2)$'den $O(p)$'ye indirgemektedir (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 27).

2.7.17. Ortak Aday Ürün Kümelerine Bölünmüş Veritabanı Algoritması (CCPD)

Veri paralel algoritmalar arasında yer alan CCPD algoritması aslında birtakım güncelleştirmeler ile SGI Power Challenge adındaki paylaşımlı bellek üzerinde CD algoritmasının işletilmesi sonucu geliştirilmiştir. Söz konusu bellekte verimli aday kümelerin oluşturulması ve değerlendirilmesine yönelik yöntemler önerilmektedir. Yaygın öge kümeleri genel olarak ilk ögeye göre eşit gruplara ayırmakta ve her bir eşit gruptan aday kümeleri üretmektedir. Yaygın öge kümelerin sınıflandırılması aday sayısını düşürmeyeceğini, bununla beraber aday oluşturma zamanını düşüreceği fikri mevcuttur. Ek olarak, her bir kayıtta aday kümelerin verimli bir şekilde değerlendirilmesi için kısa-devreli bir alt küme denetim metodu sunmaktadır (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 27).

2.7.18. Veri Dağılımı Algoritması (DD)

İş paralel algoritmaları arasından yer alan ve 1996 tarihinde Agrawal ve ark. tarafından geliştirilen DD algoritması, "Round-Robin" adlı yöntem ile aday öge kümeler var olan işlemciler üzerinde daire biçiminde parçalara bölünmektedir. Algoritmanın işleyişi Tablo 38'de gösterilmektedir. Buna göre algoritma genel olarak 3 aşamadan ileri gelmektedir. Birinci aşamada, her bir işlemci kendisine atanan aday öge kümelerin destek değerlerini hesaplamak için veritabanının bir kısmını taramaktadır. İkinci aşamada, her bir işlemci veritabanında kendisine ayrılan kısmı diğer işlemciler ile paylaşır veritabanının diğer kısımlarını diğer işlemcilerden alır. Sonrasında, veritabanını tarayarak, veritabanı için genel destek değerlerini

hesaplar. Üçüncü aşamada ise, her bir işlemci diğerlerinden bağımsız olarak aday öge kümesi kısmındaki yaygın öge kümeleri tespit eder. Veritabanının genelindeki yaygın öge kümeleri bulmak için her bir işlemci kendi yaygın öge kümelerini diğer işlemcilere dağıtarak aday öge kümeler üretilir. Aday öge kümeler parçalara ayrılır ve bunları diğer işlemciler ile paylaşır. Bu işlemler yeni aday öge kümeler bulamayana kadar sürmektedir (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 28).

Tablo 38: DD Algoritması

Input:
I, s, $D^1, D^2, \dots, D^p$
Output:
L
Algorithm:
1) $C_1^i \subseteq$I;
2) **for** ($k = 1$; $C_k^i \neq \emptyset; k++$ **do begin**
//step one: counting to get the local counts
3) count(C_k^i, D^i); //local processor is i
//step two: broadcast the local database partition to others,
//receive the remote database partitions from others,
//scan $D^j (1 \leq j \leq p, j \neq i)$ to get global counts.
4) broadcast (D^i);
5) **for** $(j = 1; (j \leq p \ and \ j \neq i); j++)$ **do begin**
6) receive (D^j) from processor j;
7) $count\ (C_k^i, D^j)$;
8) **end**
//step three: identify the large itemsets in C_k^i
//Exchange with other processors to get all large itemsets C_k,
//generate the candidates of size k+1,
//partition the candidates and distribute over all processors.
9) $L_k^i = \{c | c \in C_k^i, c.count \geq s * |D^1 \cup D^2 \cup \cdots D^p|\}$;
10) $L_k = \cup_{i=1} p\ (L_k^i)$;
11) $C_{k+1} = apriori\_gen(L_k)$;
12) $C_{k+1}^i \subseteq C_{k+1}$; // partition the candidate itemsets across the processors.
13) **end**
14) return L= $L_1 \cup L_2 \cup \dots \cup L_k$;

Kaynak: DUNHAM, M. H., XİAO, Y., GRUENWALD L. ve HOSSAİN, Z., (2000). *A Survey of Association Rules Teknik Rapor.* Southern Methodist University, Department of Computer Science, TR00-CSE-8. ss.:28-29.

2.7.19. Akıllı Veri Dağılımı Algoritması (IDD)

İş paralel algoritmalar arasında yer alan IDD algoritması, 1997 tarihinde Han tarafından DD algoritmasının geliştirilmesi sonucu ortaya konulmuştur. IDD algoritmasında bir paketleme yöntemine başvurulmaktadır. Bu yöntem, aday kümelerdeki yük dengeli bir dağılım oluşturmak için aday kümeleri gruplandırmayı sağlamaktadır. Öncelikle, her bir öge için herhangi bir öge ile başlayan aday kümelerin destek değeri hesaplanır. Her bir gruptaki aday küme sayısı eşit olmak kaydıyla, aday küme gruplarına ögeleri paylaştırmak için bir "Bin-Packing" algoritmasından yararlanmaktadır (Han, Karypis, & Kumar, 1997, s. 281-282).

2.7.20. Bağlantı Kurallarının Çırpı Temelli Paralel Madenciliği Algoritması (HPA)

İş paralel algoritmalar arasında yer alan Bağlantı Kurallarının Çırpı Temelli Paralel Madenciliği (Hash-based Parallel Mining of Association Rules/ HPA) algoritması, "Hash" fonksiyonu aracılığı ile aday öge kümeleri gruplara ayırarak işlemcilere dağıtmaktadır (Kitsuregawa & Shintani, 1996, s. 1). HPA algoritması "Skew Handling" teknolojisi ile daha çok geliştirilmiştir. Bu teknoloji, tüm işlemcilerde kullanılabilecek ana hafıza sağlandığında tüm işlemcilere iletilen görevin daha dengeli olması için bazı aday kümelerin kopyalanması şeklindedir (Kitsuregawa & Shintani, 1996, s. 12). Algoritmanın genel işleyişi Tablo 39'da gösterilmektedir.

Tablo 39: HPA Algoritması

$\{C_1^p\}$:= All items assigned to the p-th processor based on hashed value
forall $t \in D^p$ do
    Determine the destination processor ID by applying the same hash function whish is used in item partitioning, and send that item to it. If it is its pwn ID, increment the Support_count for the item. Receive the item form the other processors and increment the support_count for that item
  end
end
$\{L_1^p\}$:= All tha candidates in C_1^p with minimum sup-
    /* Each processor can determine individually whether assigned candidate k-itemset satisfy user-specified minimum support or not */
    Send L_1^p to the coordinator
        /* Coordinator make up $\mathcal{L}_1 = \cup_p L_1^p$ and
    Receive $\mathcal{L}_1$ from the coordinator
while $(\mathcal{L}_{k-1} \neq \emptyset)$ **do**
        $\{C_k^p\}$:= All the candidate k-itemsets, whose hashed value corresponding to the p-th processor
        forall $t \in D^p$ **do**
            forall k-itemset $x \in t$ **do**
                Determine the destination processor ID by applying the same hash function which is used in item partitioning, and send that k-itemset to it. If it is its own ID, increment the support_count for the itemset.
                Receive k-itemset form the other processors and increment the support_count for that itemset.
    end
end
$\{L_k^p\}$:= All the candidates in C_k^p with minimum Support
Send L_k^p to the coordinator
  /* Coordinator make up $\mathcal{L}_k = \cup_p L_k^p$ and broadeast to all the processors
Recevive $\mathcal{L}_k$ from the coordinator
$k := k + 1$
end

Kaynak: KITSUREGAWA, M. ve SHINTANI, T. (1996). Hash Based Parallel Algorithms for Mining Association Rules. Fourth International Conference on Parallel and Distributed Information Systems. USA. Proceedings of PDIS. 1-12. ss.:6.

Algoritma, aday öge kümeleri oluşturmak için her bir işlemciyi aynı Hash fonksiyonu üzerinde çalıştırmaktadır. Böylece her bir işlemci (k-1) boyutundaki büyük öge kümeler aracılığıyla, k boyutundaki aday öge kümeler oluşturmaktadır. Parçalanmış veritabanları üzerinde çalışmak yerine, aynı fonksiyon aracılığıyla kayıtların alt öge kümeler hedef işlemcilere yönlendirilirler.

Bu sayede, bir kayıtta bulunan alt öge küme i adet işlemciye iletilmesi yerine sadece bir işlemciye iletilmektedir (Kitsuregawa & Shintani, 1996, s. 6).

2.7.21. Paralel Bağlantı Kuralları Algoritması (PAR)

İş paralel algoritmalar arasında yer alan ve 1997 tarihinde Zaki tarafından geliştirilen PAR algoritması, farklı adayları ayırma ve sayımında başvurulan bir takım algoritmadan oluşmaktadır. İşlem yapılmamış yatay yapıdaki veritabanına ait bölüm (kayıt listeleri) olmasına karşın dikey yapıdaki bir veritabanı bölümü (her bir öge için ayrı sıra yani TID listesi) dikkate alınmaktadır. Dikey yapıdaki bir veritabanında herhangi bir öge kümenin sayım işlemi, öge kümedeki ögelerin TID ile kesişimi ile yapılmaktadır. Burada önemli olan veritabanının yapısının yatay olması durumunda dikey yapıya doğru gerçekleşecek dönüşümdür. Veritabanındaki senkronizasyonu en aza indirgemek için veritabanı kopyalanabilir. Birliktelik kuralları elde edilirken PAR algoritmasının türevleri olan Par-Eclat ve Par-MaxEclat algoritmaları ile Par-Clique ve Par-MaxClique algoritmalarından yararlanılmaktadır. İlk iki algoritma aday kümelerin ilk ögesine göre eşdeğerlilik sınıfını üretmektedir. Diğer iki algoritma ise aday kümeleri gruplara ayırmak için en çok hipergraf grubundan (hypergraph clique) yararlanmaktadır. Hipergraf için bir köşe bir öge demektir. Ayrıca, x köşeleri arasındaki herhangi bir kenar, x tepe noktaları ile alakalı ögeleri kapsayan bir öge kümedir. Bir sınıf birbiri ile bağlantılı ve tüm köşeleri kapsayan bir alt grafiktir. Ek olarak, Par-MaxEclat ve Par-MaxClique algoritmaları ile en çok öge kümeleri yani diğer öge kümelerin alt kümesine dahil olmayan öge kümelere ulaşabilmektedir. Öge kümeler için sayma görevi, yukarıdan aşağıya veya tam tersi yönde ya da her iki şeklinde olduğu karma bir şekilde yapılabilir. Bu algoritmaya göre aday kümeleri gruplara ayırmak için yaygın 2-öge kümelere gereksinim olana dek (hipergraf sınıfı ya da denklik sınıfı aracılığıyla), söz konusu tüm 2-öge kümelerdeki üretimleri yakalamak için ön işlem aşamasına başvurmaktadır (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 30).

2.7.22. Aday Küme Dağılımı Algoritması

Diğer paralel algoritmalar arasında yer alan Aday Küme Dağılımı algoritması 1996 tarihinde Agrawal tarafından CD ve DD paralel algoritmalarındaki iletişim yükünü indirgemek için geliştirilmiştir. Her iki algoritmanın bir araya getirilip geliştirilmesi nedeniyle üretilen Aday

Küme Dağılımı, karma algoritma özelliğini taşımaktadır (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 31).

Algoritma, birinci tarama esnasında adayları bölümlere ayırmak amacıyla kullanmaktadır. Bunun için sık tekrarlanan ögelerin özelliğinden faydalanmaktadır. Bu avantaj sayesinde her bir işlemci diğer işlemcilerden bağımsız olarak farklı gruplar üretebilmektedir. Ayrıca, veritabanı çoğaltılabilir. Bu sayede, herhangi bir işlemci genel hesaplamaları bağımsız olarak yapabilir (Zaki, Li, & Parthasarathy, 1997, s. 323).

2.7.23. Çarpık İşleme Algoritması (SH)

Diğer paralel algoritmalar arasında yer alan SH algoritması, 1998 tarihinde Harada tarafından geliştirilmiştir. SH algoritması aday kümeleri Apriori algoritmasında olduğu gibi bir önceki yaygın öge kümelerden üretmemektedir. Bu metot yerine veritabanındaki bölümler incelenirken aday öge kümeler, uç birimlerdeki her bir işlemci tarafından bağımsız olarak üretilmektedir. İterasyon (yineleme) işleminde k, her bir işlemci kendisine ait veritabanı parçasındaki kayıtlardan k-öge kümeleri üretir ve bunlara ait sayım işlemini gerçekleştirir. Her iterasyonun sonunda her bir işlemci k-öge kümeler ve bunlara ait yerel destek değerlerini değiştirir. Böylece, tüm k-öge kümelere ait genel destek değerlerini elde etmiş olmaktadır (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 31).

2.7.24. Hibrit Dağılımı Algoritması (HD)

Diğer paralel algoritmalar arasında yer alan HD algoritması, 1997 tarihinde Han tarafından geliştirilmiştir. HD algoritması, CD ve IDD algoritmalarının artı yönlerini bir araya getirmektedir. HD algoritması da IDD algoritmasında olduğu gibi ana belleğin kullanımı açıktır. Ancak, HD algoritması sınıflandırılmış aday öge kümenin her bir işlemcinin ana belleğe sığması ve her birinin hesaplamayı sağlayacak sayıda aday kümeyi içeren bölmelerin sayısını dinamik olarak hesaplamaktadır. HD, CD algoritmasının avantajlarını destek değerlerini takas ederek ve işlemcilerin daha küçük alt kümeleri arasında en az işlem sayısı elde etmek için kullanmaktadır. Ayrıca, HD algoritması ana belleği CD algoritmasına göre daha iyi kullanmaktadır. Bu sayede, geçiş başına bir veritabanının taranır ve daha küçük MDD değeri ile daha çok birliktelik kuralına erişebilmektedir. HD algoritması, iletişimdeki yük dengesini

minimuma indirgemek için ileriki geçişlerde kendiliğinden CD algoritmasına geçiş sağlayabilir (Han, Karypis, & Kumar, 1997, s. 286-287).

2.7.25. Algoritmaların Karşılaştırılması

Birliktelik Kuralları Analizinin kullanım alanının artması ve veri yapısına uygun olması için birçok algoritma geliştirilmiştir. Geliştirilen diğer birçok algoritma, apriori algoritmasının türevi veya uzantılarıdır (Győrödi, Győrödi, & Holban, 2004, s. 213). Tablo 40'ta algoritmaların matematiksel açıdan karşılaştırılması gösterilmektedir.

Tablo 40: Algoritmaların Karşılaştırılması

Algoritma	Tarama Sayısı	Veri Yapısı	Yorumlar
AIS	m+1	Belirtilmemiş	Belli başlı, az veri içeren seyrek hareketsel veritabanları için uygundur.
SETM			SQL uyumlu.
APRIORI		L_{k-1} : Hash tablosu C_k: Hash ağacı	Belli başlı, orta düzey veri içeren hareketsel veritabanları için uygundur. AIS ve SETM algoritmalardan performans açısından daha iyidir. Paralel algoritmaların için temel bir algoritma niteliğindedir.
APRIORI-TID		L_{k-1} : Hash tablosu C_k: TID ile dizinlenmiş dizi $\overline{C_k}$: Sıralı mimari ID: bitmap	Bu algoritma $\overline{C_k}$ değerinin yüksek olduğu noktalarda yavaşlamaktadır. Ancak, algoritma $\overline{C_k}$ değerinin daha düşük değer alması durumunda Apriori'ye göre daha iyi performans sağlamaktadır.
APRIORI-HYBRİD		L_{k-1} : Hash tablosu İlk safhada: C_k: Hash ağacı İkinci safhada: C_k: TID ile dizinlenmiş dizi $\overline{C_k}$: Sıralı mimari ID: bitmap	Apriori algoritmasına göre daha iyi çalışmaktadır. Apriori algoritmasından, apriori-TID algoritmasına geçiş zor olmaktadır. Bu noktada bu algoritma geçiş noktasını belirlemede etkilidir.
OCD	2	Belirtilmemiş	Yüksek boyutlu veritabanlarında düşük destek eşik değerleri söz konusu ise tercih edilebilir.
SAMPLING			Yüksek boyutlu veritabanlarında düşük destek eşik değerleri söz konusu ise tercih edilebilir.
PARTITION		Hash tablosu	Yüksek boyutlu veritabanları için idealdir. Homojen veri dağılımını desteklemektedir.
CARMA			İşlem dizilerinin bir ağdan okunmasına uygundur. Süreç boyunca çevrimiçi olarak kullanıcılar her an geri bildirim alarak destek ve/veya güven değerlerini değiştirebilirler.
FP-GROWTH		FP- Tree	Aday üretimsiz bir tekniktir. Apriori ve türevlerinden daha iyi performansa sahiptir.

DIC	Aralık boyutuna bağlıdır.	Ağaç	Veritabanı, hareketsel aralıklar şeklindedir. Yüksek boyutlu adaylar, bir aralığın sonunda oluşturulur.
CD	m+1	Hash tablosu ve ağacı	Veri Paralelliği
PDM			Veri Paralelliği ile erken aday budama
DMA			Veri Paralelliği ile aday budama
CCPD			Veri Paralelliği, paylaşımlı bellek makinesinde kullanılabilir.
DD			Görev Paralelliği; round-robin bölümü
IDD			Görev Paralelliği; ilk ögelere göre bölüm.
HPA			Görev Paralelliği; hash fonksiyonu ile bölüm.
SH			Veri Paralelliği; Her bir işlemcide adaylar bağımsız olarak oluşturulmaktadır.
HD			Hibrit veri ve görev paralelliği; ızgara paralel mimarisi.

Kaynak: DUNHAM, M. H., XİAO, Y., GRUENWALD L. ve HOSSAİN, Z., (2000). *A Survey of Association Rules Teknik Rapor.* Southern Methodist University, Department of Computer Science, TR00-CSE-8. ss.:39-40.

Birliktelik kuralları analizi için ilk geliştirilen algoritma, AIS algoritmasında uzmanlar tarafından eksiklik ve problemler olması nedeniyle SETM algoritması geliştirilmiştir. AIS algoritmasında görülen temel iki problem gerçekte küçük olan çok fazla aday öge küme üretmesi ve büyük aday kümelerin elde edilmesi için uygun veri yapılarını belirtmemesidir (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 8). Bu sebeple bu algoritma birim (ürün, öge) sayısının fazla olduğu veritabanlarında verimli sonuçlar vermeyecektir. Dolayısıyla, algoritmanın düşük önemlilikteki işlevsel veritabanlarında uygulanması daha uygun sonuçlara ulaştırmaktadır. Ayrıca, AIS algoritmasının veritabanını birden çok kez tarama ihtiyacı olması işlem hızını da düşürmektedir. AIS algoritmasında sonra geliştirilen SETM algoritmasında da aynı durum geçerlidir (Agrawal & Srikant, 1994, s. 492-493-494).

Birliktelik kuralları analizinde en popüler algoritma olan Apriori algoritması, veri madenciliği çalışmalarında yüksek performans sağlamaktadır. Algoritmanın bu performansı kendisinden önce geliştirilen AIS ve SETM algoritmalarına göre ciddi anlamda kendini göstermektedir. Apriori algoritması birliktelik kuralları için en çok tercih edilen algoritma olmasına karşın, bu iki algoritmada olduğu gibi öge kümeler için destek değerlerini hesaplarken, her aşamada veritabanını taramaktadır (Agrawal & Srikant, 1994, s. 489). Bu durum hem işlem süresinin hem de işlem boyutunun artmasına sebep olmaktadır. 1994 tarihinde, Agrawal ve Srikant bu probleme çözüm olarak Apriori algoritmasını geliştirerek, Apriori-TID algoritmasını ortaya koymuşlardır. Apriori-TID algoritmasının destek değerlerini hesaplamak için tek bir taramaya

ihtiyaç duyması Apriori algoritmasına göre en dikkat çekici farkıdır. Ayrıca, önceki geçişlerde Apriori algoritması Apriori-TID algoritmasına göre daha kısa sürede işlemleri gerçekleştirmesine karşın, ilerideki geçişlerde ise tam tersi durum yaşanması ile Apriori-TID algoritmasının performansı daha iyi sonuç vermektedir. Dolayısıyla bu iki algoritmanın avantajlarının bir araya getirildiği Apriori-Hybrid adlı karma bir algoritma geliştirilmiştir. Apriori-Hybrid algoritması, yüksek boyutlu veritabanlarında test edildiğinde başarılı sonuçlar verdiği görülmüştür. Ayrıca, geçişlerden sonraki değişim süreci dışında Apriori-Hybrid algoritmasının Apriori algoritmasına göre daha başarılı sonuçlar verdiği de görülmüştür (Agrawal & Srikant, 1994, s. 496).

Birliktelik kuralları analizi için geliştirilen bir diğer algoritma olan OCD algoritması, ortalama bir yöntem olmasına karşın düşük MDD değeri ile sık aralıklı öge kümelerinin bulunması açısından çok verimlidir. Ayrıca, AIS algoritması veritabanını birden çok kez taramaya tabi tutarken, OCD algoritması çok farklı bir yaklaşım izlemektedir. "Birliktelik Kuralları Analizi Hakkında Gelişmeler ve Uygulama A lanları" adlı başlıkta bahsedildiği üzere, bu algoritmalar iki farklı veritabanı üzerinde test edilmiştir. Çalışma sonucunda OCD algoritması hem süre hem de etkinlik açısından daha iyi sonuçlar sağlamıştır (Mannila, Toıvonen, & Verkamo, 1994, s. 186-187). AIS algoritması OCD algoritmasına göre çok fazla aday sayısı üretmektedir. Bunun nedeni, AIS'nin aşamalarda kopya adaylar üretmesidir. Ancak, OCD bir adayı sadece bir kez üretip, veritabanında işlem yapmadan önce söz konusu adayın alt kümelerinin yaygın olup olmadığını incelemektedir. Bu, OCD algoritmasının bir diğer avantajıdır (Mannila, Toıvonen, & Verkamo, 1994, s. 189).

DIC algoritmasında gerçek bir veritabanı için tarama sayısı, veritabanındaki aralıkların uzunluğuna bağlıdır. Aralıkların uzunluğu gerektiği kadar küçük ve homojenlik sağlandığında birinci taramada tüm öge kümeler üretilir ve ikinci taramada tüm veritabanı sayılabilmektedir. Bu durum, DIC algoritmasının Apriori algoritmasına oranla daha hızlı çalışmasını sağlamaktadır (Brin, Motwani, Ullman, & Tsur, 1997, s. 259).

CARMA algoritması, görev sıralamalarının online bir ağdan okunması ile online kullanıcı etkileşimli geri besleme mekanizması ile yönlendirilmiş bir yöntem olması açısından en uygunudur. Ayrıca, Apriori ve DIC algoritmalarına kıyasla daha hızlı olmayan CARMA algoritması, daha yüksek bellek etkinliğine sahiptir (Hidber, 1999, s. 145-146).

Sık desenlerle ilgili sıkıştırılan ve önemli bilgileri saklı tutmak için oluşturulan FP-Tree ile yeni bir veri yapısı olan FP-Growth algoritmasının diğer yöntemlere göre çeşitli avantajlara sahiptir. Genel olarak asıl veritabanından daha küçük boyutlu ve veri madenciliği çalışmalarında maliyetli veritabanı taramalarını gerçekleştiren bir ağaç yapısı oluşturmaktadır. Ayrıca, algoritma maliyetli olan aday kümelerin üretilmesine engeller ve (şartlı) FP ağaçlarında olan sık tekrarlanan 1-öge kümeyi birbirlerinin devamında birleştirerek sınayan bir model büyüme metodu uygulamaktadır. Bu sayede, Apriori benzeri algoritmaların çoğuna göre aday öge küme üretme ve desen eşleşme görevlerini genel olarak daha az maliyetli yapmaktadır. Son olarak, şartlı desen tabanları ile şartlı FP ağacının boyutunu ciddi boyutta indirgeyen bir bölümleme tabanlı "böl ve yönet" metodunu uygulamaktadır. Bir ağaç yolu için direkt model üretme ve en minimum tekrara sahip durumları kapsayan birkaç diğer optimizasyon yöntemi de algoritmanın verimliliğine katkı sağlamaktadır (Han, Pei, & Yin, 2000, s. 11-12).

FP-Growth algoritması için veritabanının en çok iki kez taranması yeterli iken, Apriori için veritabanının tarama sayısı aday ögelerin boyutu ile paralellik göstermektedir. FP-Growth algoritmasının performansını destek faktörü etkilemezken, Apriori algoritmasının performansını düşürmektedir (Győrödi, Győrödi, & Holban, 2004, s. 220). Yüksek boyutlu veritabanlarında MDD değerinin düşük olduğu durumlarda da FP-Growth, Apriori'ye göre üstünlük sağlamaktadır. Daha açık bir ifadeyle, MDD değerinin düşük olması, birden çok ögenin destek değerinin onaylanması ve Apriori algoritmasında oluşturulan öge kümelerin sayısal anlamda artışına sebep olmaktadır. Bu artış daha verimli bir yapıya sahip olan FP-Growth algoritması için geçerli olmamaktadır (Erpolat, 2012, s. 140).

IDD algoritması, CD algoritmasından daha verimli olacak ana belleği kullanmaktadır (Han, Karypis, & Kumar, 1997, s. 286). IDD algoritması, aday öge kümelerindeki ilk ögeye göre aday öge kümeleri işlemciler arasında pay etmektedir. Böylece, ilk ögesi aynı olan kümeler aynı parçalarda bulunacaktır. Bu sebeple, her bir işlemci sadece kendisine atanan ögelerden biri ile başlayan alt kümeleri denetlemektedir. DD algoritmasında ise her bir işlemci her bir kayıttaki alt kümeleri denetlemek zorunda kalmaktadır. Bu yüzden, IDD algoritması ile DD algoritmasında yaşanan gereksiz hesaplamalar ortadan kaldırılmaktadır (Han, Karypis, & Kumar, 1997, s. 281-282).

SH algoritması, Apriori algoritmasına göre farklı görünse de esasen birbirine çok yakın iki algoritmadır. Her iki algoritma iteratif algoritmalardır. Yani, sadece bir yinelemenin sonuna gelindiğinde artan boyutta yeni aday öge kümeleri üretilmektedir. Bu iki algoritmanın farkı ise, aday kümelerin elde ediliş şeklidir. Apriori algoritması aday kümeleri her bir yineleme sonunda oluştururken, SH algoritması aday kümeleri işlemler yapılırken elde etmektedir. Ek olarak, veritabanı eşit olarak paylaştırıldığında, SH ve Apriori algoritmalarının aday öge kümeleri aynı olmaktadır. Bu benzerliği, veritabanının çok fazla çarpık olması bozmaktadır. Bu durumda algoritmaların oluşturduğu aday öge kümeler farklı olacaktır (Dunham, Gruenwald, Hossain, & Xiao, 2000, s. 32).

3. MARKET SEPET ANALİZİ ÇALIŞMASI

Bireyler tarafından yapılan alışverişlerin arka planında bu bireylerin satın alma tercihleri saklıdır. İşletmeler tarafından yapılan ürün veya hizmet satış kayıtları bu gizli kalan bilgiyi ortaya çıkarmada kilit rol oynamaktadır. Bu kayıtların bilgiye dönüştürülmesi için işletmeler veri madenciliği çalışmalarına başvurmaktadır. Veri madenciliği çalışmaları ile hem mevcut durum görülmekte hem de gelecekteki durumlar için eylemler organize edilebilmektedir.

Süper market uygulamalarında kullanılan birliktelik kuralları analizi literatürde market sepet analizi olarak geçmektedir. Market sepet analizi, işletmelerin veritabanında yer alan ziyaretçilerin satın aldığı ürünler arasındaki bağlantıları ortaya çıkarmaktadır. Bu çalışmalarda veritabanındaki verileri toplayan en iyi veri sağlayıcı araç mağazalarda indirim sağlayan müşteri kartlarıdır.

Çalışmada elde edilen kurallar ile işletme yöneticileri mağazalardaki reyon-raf planlaması, ürün yerleştirmeleri, mağaza yapısının dizayn edilmesi, çapraz satış gibi satış stratejileri oluşturma, promosyon, indirim ve katalog gibi kampanyaların düzenlemelerini destekleyici müşteri profilleri çıkarabilir ve satış arttırıcı fırsat ürün ve paket ürünleri tespit edilebilir. Çamurcu ve Özçakır'ın pastane sektöründeki uygulaması veya Erpolat'ın otomotiv sektöründeki uygulaması bu çalışmalara örnek olarak verilebilir. Çamurcu ve Özçakır, bir kurumun pastane bölümüne ait satış verileri üzerinde Apriori algoritmasını uygulamıştır. Analiz ile müşterilerin ne zaman, nerde ve hangi ürünleri birlikte aldıklarını çözümleyecek örüntülere ulaşmak hedeflenmiştir. Çalışmada, çoğunlukla aynı ürün grubuna ait ürünlerin en çok birlikte satın alındığı gözlemlenmiştir (Çamurcu & Özçakır, 2007). Erpolat ise, Türkiye'de otomotiv sektöründe faaliyet gösteren bir kuruma ait müşteri veritabanından yararlanarak Birliktelik Kuralları analizi yapılmıştır. Çalışmada Apriori ve FP-Growth algoritmalarının sonuçları karşılaştırılmıştır. Müşterilerin alışveriş alışkanlıklarına dair kurallar elde edilerek, satış arttırıcı kampanyaların oluşturulması hedeflenmiştir (Erpolat, 2012).

Birliktelik kuralları analizine yukarıdaki çalışmalara ek olarak; her bir ürün için gelecek dönemdeki satış tahminleri, yeni çıkarılacak bir ürün için ideal satış alanı seçimi, beraber satın alınan ürünlere dair il, bölge, şube ve tarih bazlı olarak promosyon çalışması, işletmenin internet sitesinde belirli bir ürün alımı gerçekleştirecek müşteriye beraber alabileceği ürünü önererek

çapraz satış stratejileri geliştirme (Birant, ve diğerleri, 2010, s. 6-7), personel ve üretim arasındaki ilişkileri ortaya çıkarma ve müşterilerin dilek, öneri ve şikayetleri incelenerek müşteri ilişkilerini geliştirme (Alagöz, Ortakarpuz, & Öge, 2014, s. 16) amaçları için başvurulabilir.

Bu çalışmada Türkiye'de perakende sektöründe yer alan bir işletmenin satış verileri üzerinde market sepet analizi yapılmıştır. Detaylı kurallar bulabilmek için R yazılımı ile gerçekleştirilen analiz hem ürün grupları hem de ürünlerin ait olduğu alt kategoriler dikkate alınarak, iki farklı şekilde gerçekleştirilmiştir.

3.1.Araştırma Verileri

Araştırma kapsamı, Türkiye'de perakende sektöründe yer alan bir işletmenin Aralık 2018 ayında bazı şubelerinde gerçekleştirilen 1000 adet fiş kaydından oluşmaktadır. Gizlilik kuralları sebebiyle, çalışmada işletme bilgilerine ve ürünlerin markalarına dair belirleyici bilgilere yer verilmeyecektir.

Veri hazırlama sürecinde veriler Microsoft Excel programında düzenlenmiştir. Bir ürün alımı gerçekleştirilen 170 adet fiş kaydı veri setinden çıkarılarak, 830 adet fiş kaydı ile analize devam edilmiştir. Veri seti; fiş, ürün, konum ve müşteri bilgilerinden oluşmaktadır. Örnekleme sonucunda fiş kayıtlarında bulunan birimler Tablo 41'deki gibi isimlendirilerek, bir veri çerçevesi (Data Frame) oluşturulmuştur.

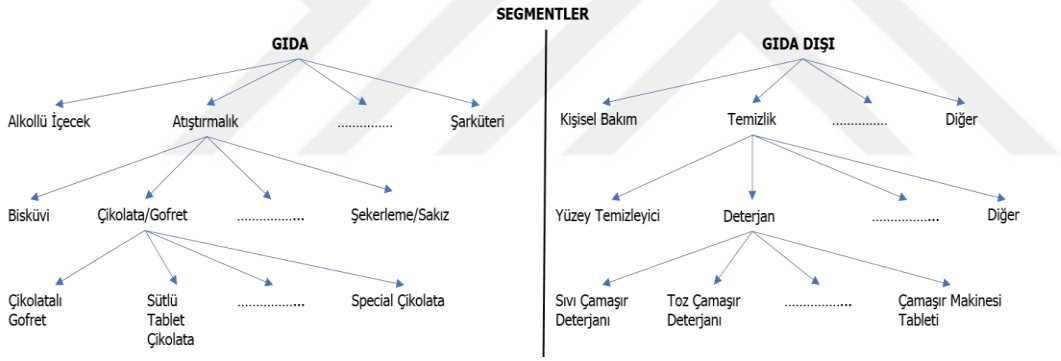
Tablo 41: Analizde Kullanılan Değişkenler

Alanlar	Bir fişte yer alan birimler	Değişkenler
FİŞ	Fiş Numarası	FIS_ID
	Alışveriş Tarihi	DATE
	Alışveriş Başlangıç Saati	START_TIME
	Alışveriş Bitiş Saati	END_TIME
	Alışveriş Süresi	THE PASSING TIME
	Alışveriş Tutarı	SHOP_AMOUNT
	Ödeme Türü	PAYMENT_METHOD
KONUM	İl	CITY
	İl Numarası	CITY_ID
	Bölge	REGION
	Bölge Numarası	REGION_ID

	Mağaza No	MARKT_ID
ÜRÜN	Ürün Adı	SKU
	Ana Kategori	SEGMENT
	Kategori	CATE
	Alt Kategori	SUB_CATE
	Miktar	QUANTITY
MÜŞTERİ	Müşteri No	CUST_ID
	Cinsiyet	GENDER
	Yaş	AGE

Kaynak: Yazar tarafından derlenmiştir.

Fiş kayıtlarında yer alan ürün grupları gıda ve gıda dışı olmak üzere iki segmente ayrılmıştır. Segmentler toplamda 39 kategori ve 200 alt kategoriye ayrılmaktadır. Ürünlerin kategorilere göre dağılımını veren kategori şeması Şekil 15'te gösterilmektedir. Satın alınan ürünlerin çoğunluğu gıda ürünleri olmak üzere, en çok ürün alımı %18 oran ile atıştırmalık kategorisinde ve %7 oran ile sebze alt kategorisinde gerçekleşmiştir.

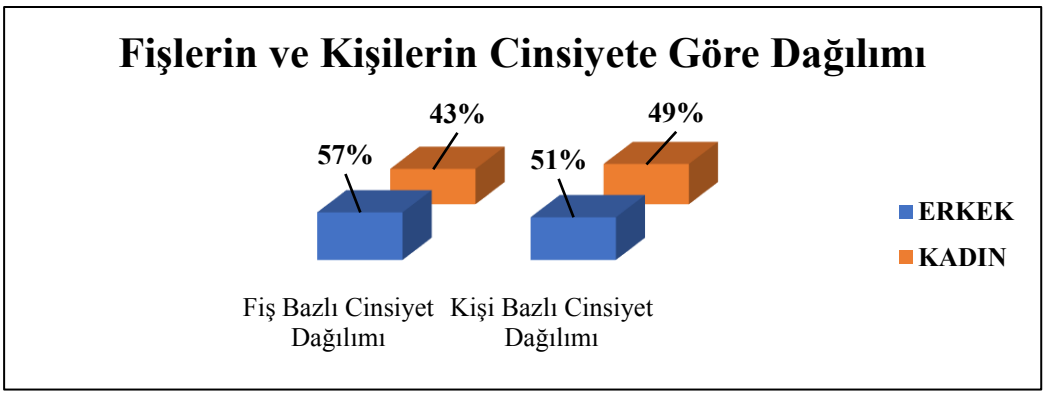


Şekil 15: Kategori Şeması

Kaynak: Yazar tarafından derlenmiştir.

48 ili ve 301 mağazayı kapsayan çalışmada en çok alışveriş yapılan il %33 oran ile İstanbul ve en çok alışveriş yapılan mağaza %6 oran ile İstanbul'da yer alan 25 numaralı mağazadır. En küçük yaşın 19 ve en büyük yaşın 78 olduğu çalışmada, %7 oran ile en çok 41 yaşındaki müşterilerin alışveriş yaptığı gözlenmiştir. Fiş hareketlerine göre; %5 oran ile en çok alışveriş yapılan tarih 31.12.2018 ve en az alışveriş yapılan tarih 13.12.2018 olduğu gözlenmiştir. Gün içindeki zaman aralıklarına göre; alışverişlerin %12'si sabah, %52'si öğle ve %36'sı akşam saatlerinde gerçekleştiği görülmüştür. Veri setindeki fiş kayıtları ve kişilerin cinsiyete göre dağılımı Grafik 7'de gösterilmektedir.

Grafik 7: Fişlerin ve Kişilerin Cinsiyete Göre Dağılımı

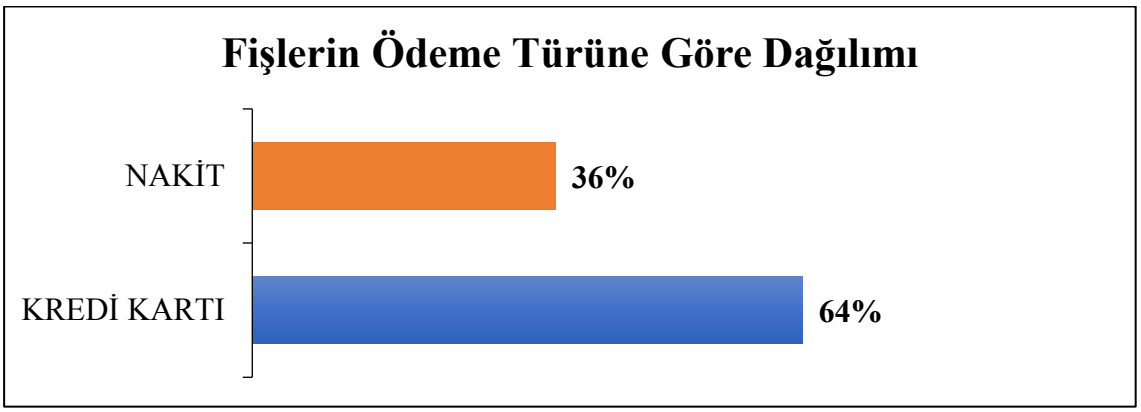


Kaynak: Yazar tarafından derlenmiştir.

Grafik 7'ye göre yapılan alışverişlerin 476'sı 107 erkek müşteri tarafından ve 354'ü 101 kadın müşteri tarafından gerçekleştirilmiştir.

Alışverişlerde müşterilerin tercih ettikleri ödeme türü hakkında bilgiler mevcuttur. Grafik 8'de yapılan alışverişlerin ödeme türüne göre dağılımı gösterilmektedir. Yapılan alışverişlerin 535'i kredi kartı ve 295'i nakit olarak yapılmıştır.

Grafik 8: Fişlerin Ödeme Türüne Göre Dağılımı



Kaynak: Yazar tarafından derlenmiştir.

Grafik 8'e göre, müşteriler alışverişlerinde çoğunlukla kredi kartı ile ödeme yapmayı tercih etmişlerdir.

Ürün grupları arasındaki ilişkileri keşfetmek için birliktelik kuralları yönteminde en çok başvurulan apriori algoritması tercih edilmiştir. Apriori algoritmasının faydaları ve genel işleyişi hakkındaki bilgiler "2.7.3. Apriori" başlığı altında verilmişti. Ürün grupları arasındaki ilişkileri bulmak için algoritma çalışması yapılmadan önce çalışmada bir ürün alımı gerçekleştirilen fişler veri setinden arındırılmalıdır. Daha sonrasında çalışmayı gerçekleştiren uzman tarafından çalışmanın kapsamına göre, en az kaç ürün alımı gerçekleştirilen fişlerin dikkate alınacağı belirlenmektedir. Bu çalışmada en az 2 ürün alımı gerçekleştirilen fişler dikkate alınmıştır.

Çalışma kapsamında yer alan fiş kayıtlarında en az ürün bulunan fişteki ürün sayısı 2 ve en fazla ürün bulunan fişteki ürün sayısı 49'dur. Veri setinde en çok gündeme gelen ürün adedi yani tepe değeri 2'dir. Ürün grupları ve alt kategori adet bilgilerine ilişkin bazı tanımlayıcı istatistikler Tablo 42'de gösterilmektedir.

Tablo 42: Ürün ve Alt Kategori Adetlerine İlişkin Tanımlayıcı İstatistikler

Ürünler		**Alt Kategoriler**	
Gözlem Sayısı	592	Gözlem Sayısı	200
En Az Tekrar Eden Ürün Adedi	1	En Az Tekrar Eden Alt Kategori Adedi	1
En Çok Tekrar Eden Ürün Adedi	209	En Çok Tekrar Eden Alt Kategori Adedi	346
Geçerli Gözlem Sayısı	592	Geçerli Gözlem Sayısı	200

Kaynak: Yazar tarafından derlenmiştir.

Tablo 42'ye göre veri setinde 592 adet tekil ürün grubu ve 200 alt kategori mevcuttur. En çok görülen ürün 209 kez ve en çok görülen alt kategori 346 kez alınmıştır. En az görülen ürün ve alt kategori ise 1 kez alınmıştır.

Analiz sonuçları değerlendirilirken şu noktalar dikkate alınmalıdır. Bu çalışmada analizin yapıldığı şubeler ve burada gerçekleştirilen fiş kayıtları dikkate alınmıştır. Çalışma bir ayı kapsamakla beraber, sonuçlar farklı dönemlerde geçerli olmayacaktır. Ayrıca, bu dönemde işletme yöneticilerinin ilgili şubelerde yaptıkları satış arttırıcı kampanyalar çalışmayı etkilemektedir. Tüm bu noktalar dikkate alındığında, müşterilerin alışveriş alışkanlıkları çalışma kapsamındaki şubelere özgü olmakla beraber, diğer şubelere göre farklılık göstereceği bilinmektedir.

3.2.Bulgular

Microsoft Office Excel programında düzenlenen ve "3.1. Araştırma Verileri" başlığı altında bahsedilen özelliklere sahip olan veri seti, açık kaynak kodlu yazılımlar arasında yer alan R programının ara yüzü olan RStudio'da işlenmiştir. Analiz yapılırken, RStudio'da başvurulan paketlerin işlev alanları hakkındaki bilgiler Tablo 43'te gösterilmektedir. Analizde başvurulan komutlar Ek 2'de gösterilmektedir. Verilerin RStudio ara yüzünde işlenmesi için XLSX dosya formatındaki Excel dosyası CSV dosya formatına dönüştürülerek, bir veri dosyası oluşturulmuştur. Bu dosya üzerinde çalışmalar yapılmıştır.

Tablo 43: RStudio'da Başvurulan Paketler

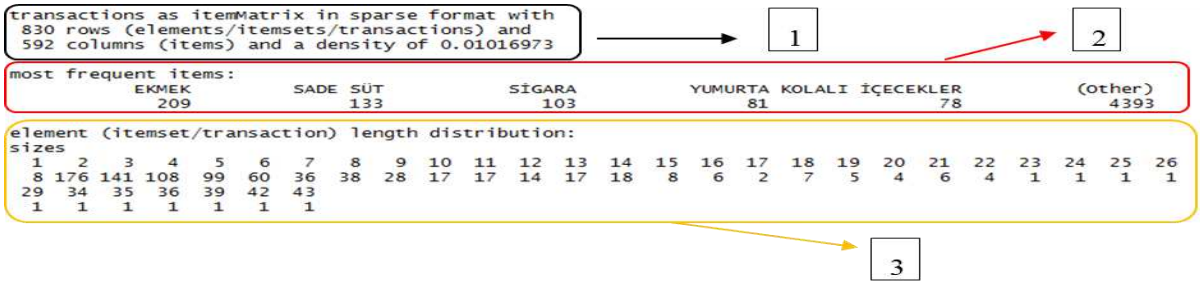
İşlev Alanı	Paket Adı	İşlev Alanı	Paket Adı
Excel programında XLSX formatındaki dosyaların RStudio'a aktarılmasını sağlar.	readxl	Veri setleri ile ilgili tanımlayıcı istatistiklerin çıkarılmasını sağlar.	summarytools
Büyük veri setlerinde ekleme, çıkarma, silme ve listeleme gibi birçok düzenleme işleminde kolaylık sağlar.	data.table	Programda dinamik raporlar oluşturmak için genel kapsamda bir pakettir.	knitr
Veri setindeki tarih ve saat kolonlarında hızlı bir şekilde düzenleme sağlar. Bu düzenlemeler arasında saat, dakika, saniye veya gün, ay ve yıl gibi kısımlarda çıkarma ve güncelleme yapılması mevcuttur.	lubridate	Çalışmada oluşturulan dosyalar için bir yol oluşturur. Bu sayede, paketin aktif edildiği andan itibaren dosyalarınıza kolayca erişimi sağlar.	here
Bu paket aktif hale getirildiğinde aynı zamanda stringr, tibble ve tidyr alt paketleri de aktif hale gelmektedir. Çoklu ve düzenli paketlerin kısa sürede kolaylıkla yüklenmesini sağlar.	tidyverse	Veriler üzerinde parçalara ayırma, parçaları ayrı ayrı inceleme ve sonrasında bu parçaları birleştirmek için gerekli araçları içerisinde barındırır.	plyr
Programda arka tarafta bir veritabanı işlevi görür. Veri çerçevesi ile ilgili işlemlerin yapılmasına olanak sağlar.	dbplyr	Veri çerçevesinde kullanılan bir veri işleme kılavuzudur. Veri çerçevesinde hızlı ve tutarlı sonuçların gerçekleştirilmesini sağlar.	dplyr
Html uzantılı işlemlerde kullanılan araçlardan biridir.	htmltools	R konsolu, 'R Markdown' belgeleri ve bazı web uygulamaları için farklı şekilde işleyen HTML gereçlerini oluşturmak için başvurulan bir çerçevedir.	htmlwidgets

İşlem verilerini işlemek ve analiz etmek için altyapı sağlar. Apriori ve Eclat birliktelik kuralları algoritmalarının uygulanmasını sağlar.	Arules	Birliktelik kuralları ve itemsetslerin görselleştirilmesinde başvurulur. Ayrıca, kuralların daha iyi anlaşılması için içerisinde birden çok grafik paketi yer almaktadır.	arulesViz
Grafikler için renk şemaları sunar.	RColorBrewer	Renkli grafikler oluşturulurken bu renk paketine başvurulur.	colorspace
Grafiklerin çizilmesi için veri setini işlemede başvurulur.	graph	Grafik dilbilgisine dayanarak, verilerin görselleştirilmesinde yardımcı olur.	ggplot2
Web grafiklerini oluşturmayı sağlar. Paket, ggplot2 grafiklerini aktif web grafiklerine çevirir.	plotly	Verileri görselleştirmede çeşitli R araçları sunar. Sütun, balon ve kutu grafiği, renk değiştirme araçları, iki boyutlu verilerin hesaplanması ve görselleştirilmesinde özel grafikler ve iyileştirilmiş regresyon grafikleri gibi farklı veri görselleştirme araçları mevcuttur.	gplots
Büyük grafikler için rasgele ve düzenli grafikler oluşturur. Ağ analizi için basit grafikler sunar.	igraph	igraph paketinin bir uzantısıdır. Daha etkin ağ analizi ve görselleştirme imkânı sağlar.	IntrecativeIGraph
Çizilen grafiklerin jpeg formatında saklanmasını ve okunmasını sağlar.	jpeg		

Kaynak: Bilgiler yazar tarafından R kütüphanesinde paket açıklamalarından derlenmiştir.

RStudio programında algoritma çalıştırılmadan önce veri dosyasının işlem datası olarak atanması gerekmektedir. Tablo 44'te ürün gruplarına göre işlem miktarları incelenecektir.

Tablo 44: Ürün Gruplarına İlişkin İşlem Dosyası Özet Bilgileri



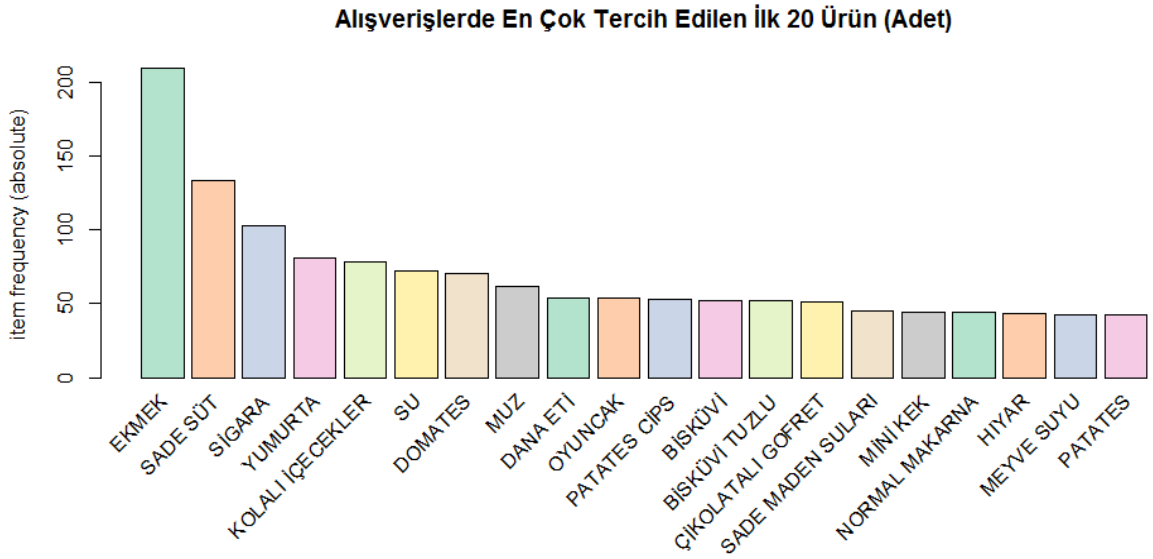
Kaynak: Yazar tarafından derlenmiştir.[1]

[1] RStudio çıktıları resim veya PDF olduğundan Türkçe'ye çevrilememiştir.

Tablo 44'teki birinci alana göre; veri setinde 830 satır yani fiş kaydı ve 592 sütun yani ürün grubu bulunmaktadır. Programdaki data formatı seyrek matris denilen özel bir yapıdadır. Bu matriste boş olmayan hücrelerin yüzdesi yani matrisin yoğunluğu %1'dir. Matrisin yoğunluk değeri, satır ve sütun adetlerin çarpımı ile 4.996,999 adet kadar ürün alımı yapıldığı görülmektedir. İkinci alanda en çok alınan ürünler ve adetleri verilmiştir. Üçüncü alanda ürünlerin fişlerdeki dağılımı verilmiştir. Bu kısma göre, 2 adet farklı ürün içeren 176 işlem olduğu ve 1 işlemde farklı 43 adet farklı ürün alımı yapıldığı görülmektedir.

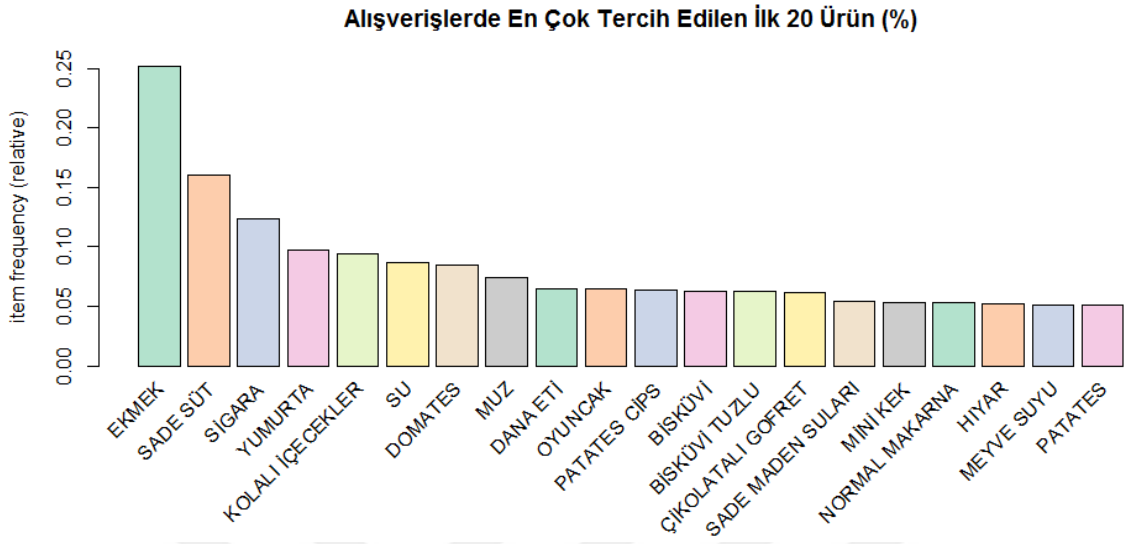
Ürün adetlerine ait frekans dağılımından yararlanarak, müşteriler tarafından en çok tercih edilen ilk 20 ürün tespit edilmiştir. Ürünlerin evrende yer aldığı miktarlara göre gösterimi Grafik 9'da ve evrende kapladıkları oranlara göre gösterimi Grafik 10'da gösterilmektedir. Grafik 10'a göre tüm alışverişlerin %25,18'inde ekmek ve %9,39'unda kolalı içecekler yer almaktadır.

Grafik 9: Alışverişlerde En Çok Tercih Edilen İlk 20 Ürün (Adet)



Kaynak: R çıktısı yazar tarafından derlenmiştir.

Grafik 10: Alışverişlerde En Çok Tercih Edilen İlk 20 Ürün (%)



Kaynak: R çıktısı yazar tarafından derlenmiştir.

Tablo 45'te alt kategorilere göre işlem miktarları incelenecektir. Tablo 45'teki birinci alana göre; veri setinde 830 satır yani fiş kaydı ve 200 sütun yani alt kategori bulunmaktadır. Seyrek matriste boş olmayan hücrelerin yüzdesi yani matrisin yoğunluğu %2,6'dır. Matrisin yoğunluk değeri, satır ve sütun adetlerin çarpımı ile 4.287 adet kadar alt kategori alımı yapıldığı görülmektedir. İkinci alanda en çok alınan alt kategoriler ve adetleri verilmiştir. Üçüncü alanda alt kategorilerin fişlerdeki dağılımı verilmiştir. Bu kısma göre, 2 adet farklı alt kategori içeren 191 işlem olduğu ve 1 işlemde farklı 34 adet farklı alt kategori alımı yapıldığı görülmektedir.

Tablo 45: Alt Kategori Adetlerine İlişkin İşlem Dosyası Özet Bilgileri

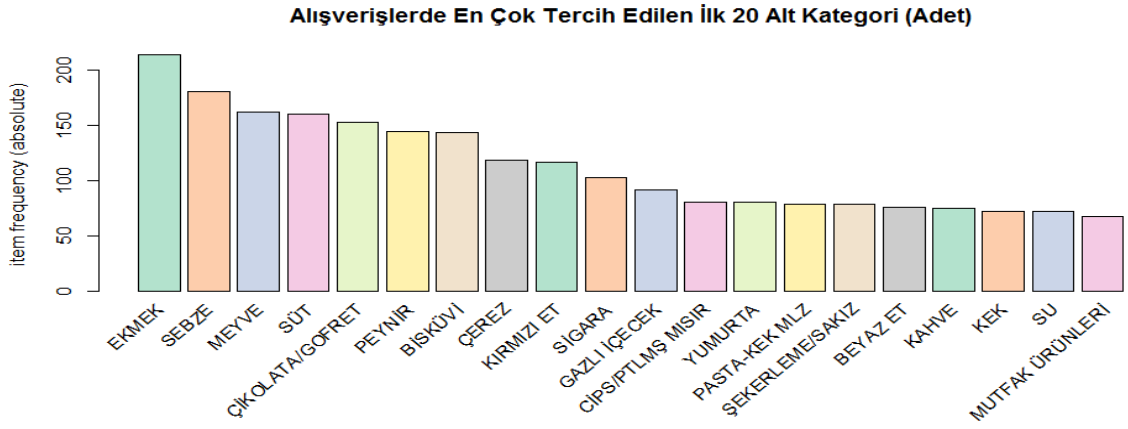
```
transactions as itemMatrix in sparse format with
 830 rows (elements/itemsets/transactions) and
 200 columns (items) and a density of 0.0258253

most frequent items:
          EKMEK          SEBZE          MEYVE SÜT ÇİKOLATA/GOFRET         (Other)
            214            181            162            160            153           3417

element (itemset/transaction) length distribution:
sizes
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  20  21  23  25  27  29  30  34
 20 191 152 128  90  61  33  29  25  20  17  13   9   9   6   9   6   2   1   2   1   2   1   1   1   1
```

1 2 3

Kaynak: R çıktısı yazar tarafından derlenmiştir.[2]

Alt kategori adetlerine ait frekans dağılımından yararlanarak, müşteriler tarafından en çok tercih edilen ilk 20 alt kategori tespit edilmiştir. Alt kategorilerin evrende yer aldığı miktarlara göre gösterimi Grafik 11'de ve evrende kapladıkları oranlara göre gösterimi Grafik 12'de gösterilmektedir. Grafik 12'ye göre, tüm alışverişlerin %25,78'inde ekmek ve %18,43'ünde çikolata/gofret yer almaktadır.

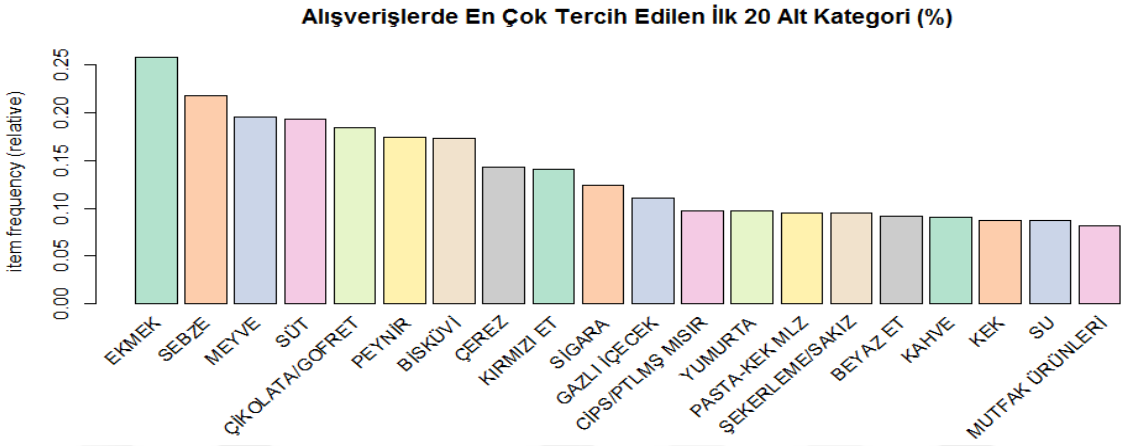
Grafik 11: Alışverişlerde En Çok Tercih Edilen İlk 20 Alt Kategori (Adet)



Kaynak: R çıktısı yazar tarafından derlenmiştir.

[2] RStudio çıktıları resim veya PDF olduğundan Türkçe'ye çevrilememiştir.

Grafik 12: Alışverişlerde En Çok Tercih Edilen İlk 20 Alt Kategori (%)



Kaynak: R çıktısı yazar tarafından derlenmiştir.

Ürün grupları baz alınarak başvurulan apriori algoritması uygulamasında model parametreleri, MGD değeri %25 ve MDD değeri %3 olarak atanmıştır. Algoritma sonuçları Tablo 46'da gösterilmektedir.

Tablo 46: Ürün Grupları İçin Apriori Sonuçları

```
set of 9 rules

rule length distribution (lhs + rhs):sizes
2
9

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
      2       2       2       2       2       2

summary of quality measures:
    support         confidence        lift            count
 Min.   :0.033   Min.   :0.25   Min.   :1.0   Min.   :27
 1st Qu.:0.034   1st Qu.:0.37   1st Qu.:1.6   1st Qu.:28
 Median :0.036   Median :0.40   Median :2.0   Median :30
 Mean   :0.043   Mean   :0.42   Mean   :3.1   Mean   :35
 3rd Qu.:0.048   3rd Qu.:0.49   3rd Qu.:2.3   3rd Qu.:40
 Max.   :0.064   Max.   :0.65   Max.   :7.7   Max.   :53

mining info:
 data ntransactions support confidence
   TR           830    0.03       0.25
```

Kaynak: R çıktısı yazar tarafından derlenmiştir.[3]

Tablo 46'da görüldüğü üzere %25 MGD ve %3 MDD değeri ile 2 ürün içeren 9 adet tekli kural elde edilmiştir. Kurallara ait destek, güven, kaldıraç ve görülme sıklığı için en düşük, en yüksek, ortalama ve medyan değerleri hesaplanmıştır. Elde edilen kurallar arasında en düşük destek

[3] RStudio çıktıları resim veya PDF olduğundan Türkçe'ye çevrilememiştir.

değeri %3,3 iken en yüksek güven değeri %65'tir. Kaldıraç değerleri için medyan yani ortanca değer 2 iken görülme sıklığı değerleri için ortalama 35 olarak hesaplanmıştır. Kurallar, güven değeri en yüksek kuraldan en düşük kurala göre sıralanarak Tablo 47'de gösterilmektedir.

Tablo 47: Ürün Gruplarına İlişkin Tekli Birliktelik Kuralları

Kurallar	Destek Değeri (Support Value) %	Güven Değeri (Confidence Value) %	Kaldıraç Değeri (Lift Value)	Görülme Sıklığı (Count)
{HIYAR} => {DOMATES}	3,4%	65%	7,7	28
{DANA ETİ} => {EKMEK}	3,5%	54%	2,1	29
{YUMURTA} => {EKMEK}	4,8%	49%	2	40
{DOMATES} => {EKMEK}	3,7%	44%	1,8	31
{DOMATES} => {HIYAR}	3,4%	40%	7,7	28
{SADE SÜT} => {EKMEK}	6,4%	40%	1,6	53
{YUMURTA} => {SADE SÜT}	3,6%	37%	2,3	30
{SİGARA} => {EKMEK}	3,3%	26%	1	27
{EKMEK} => {SADE SÜT}	6,4%	25%	1,6	53

Kaynak: R çıktısı yazar tarafından derlenmiştir.

Elde edilen 9 kuralda 7 farklı ürün söz konusudur. Bu ürünler arasında yer alan ekmek, sade süt ve yumurta müşteriler tarafından en çok alınan ilk 3 üründür. Tablo 47'de yer alan kurallar arasında hıyar ile domates ve ekmek ile sade süt ürünleri farklı kurallarda beraber yer almaktadır. Aynı zamanda bu ürünler, kaldıraç ve güven değeri en yüksek kurallar (hıyar-domates) ile destek değeri en yüksek kuralları (ekmek-sade süt) oluşturmaktadır. Bu kurallardan ikisi aşağıda açıklanmıştır.

{HIYAR} => {DOMATES} $s = \%3{,}4$, $c = \%65$ ve $kaldıraç = 7{,}7$

Yukarıda gösterilen kural, en yüksek güven değerine sahip kuraldır. Bu kurala göre; hıyar ve domates ürünlerinin beraber satın alınma olasılığı %3,4'tür. Hıyar satın alan müşteriler %65 olasılık ile domates satın alma eğilimine girmektedir. Bu kural için kaldıraç değeri 7,7 olarak hesaplanmıştır. Bu değer, kuralın öncül ve ardıl bölümünde yer alan ürünlerin birbirinden bağımsızlığını ölçen oransal bir destek değeridir. İlgili kural için kaldıraç değeri aşağıdaki gibi hesaplanmaktadır.

$$Kaldıraç\{Hıyar\} \Rightarrow \{Domates\} = \frac{Destek\{Hıyar\} \Rightarrow \{Domates\}}{Destek\{Hıyar\} \times Destek\{Domates\}} = 7{,}7$$

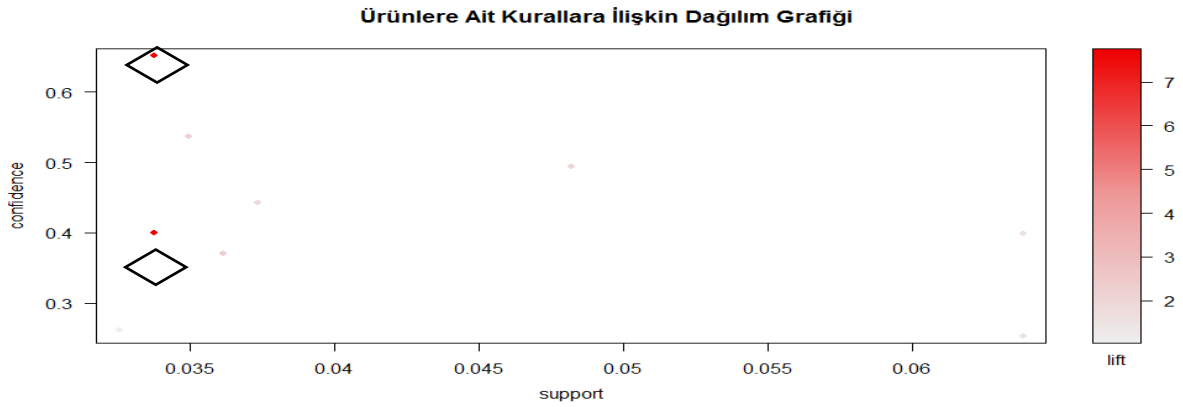
Aynı zamanda yukarıdaki kural en yüksek kaldıraç değerine sahip kurallardan biri olarak, kaldıraç değeri birden büyük olduğu için kuralda yer alan iki ürünün bağımlı olduğu ve kuralın geçerli olduğu görülmektedir. Daha açık bir ifadeyle, alışverişlerde hıyar alımları domates satışlarını 7,7 kat arttırmaktadır.

{SADE SÜT} => {EKMEK} $s = \%6{,}4$, $c = \%40$ ve $kaldıraç = 1{,}6$

Yukarıda gösterilen ikinci kural, en yüksek destek değerine sahip kuraldır. Bu kurala göre; sade süt ve ekmeğin beraber satın alınma olasılığı %6,4'tür. Sade süt satın alan müşteriler %40 olasılık ile ekmek satın alma eğilimine girmektedir. Sade süt alımları ekmek satışlarını 1,6 kat arttırmaktadır.

Ürün grupları üzerinde elde edilen 9 kural bazı grafikler ile görselleştirilmiştir. Dağılım (Scatter), gruplandırılmış matris (Grouped Matrix) ve iki anahtarlı (Two-Key) grafikleri büyük kural gruplarını görselleştirmek için kullanılabilir. Dağılım grafiği destek, güven ve kaldıraç değerine göre kategorize edilebilir. Grafik 13'te kuralların kaldıraç değerine göre dağılım grafiği gösterilmektedir.

Grafik 13: Ürünlere Ait Kurallara İlişkin Dağılım Grafiği

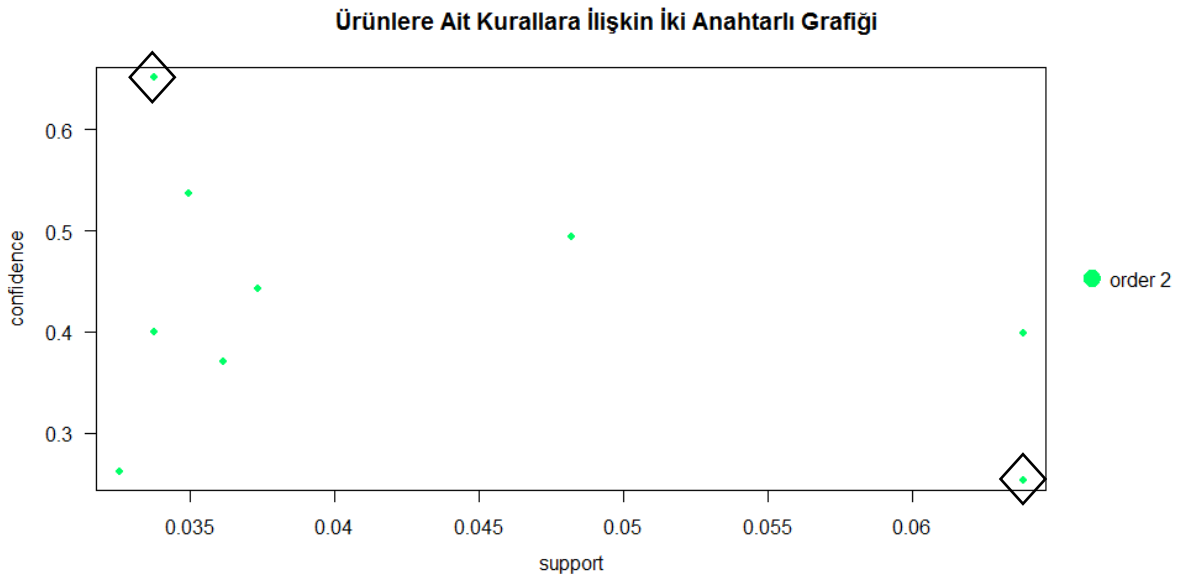


Kaynak: R çıktısı yazar tarafından derlenmiştir.

Grafik 13'te, yüksek kaldıraç değerine sahip kurallar daha koyu renkte gösterilmektedir. İki en yüksek kaldıraç değerine sahip kuralın çok düşük destek değeri aldığı gözlenmektedir.

İki anahtarlı grafik, x ekseninde destek değerlerini ve y ekseninde güven değerlerini kullanarak, kuralları grafik üzerinde nokta olarak göstermektedir. Order terimi kuralda geçerli olan ürün sayısını vermektedir. Grafik 14'te kuralların destek ve güven değerine göre iki anahtarlı grafiği gösterilmektedir.

Grafik 14: Ürünlere Ait Kurallara İlişkin İki Anahtarlı Grafik

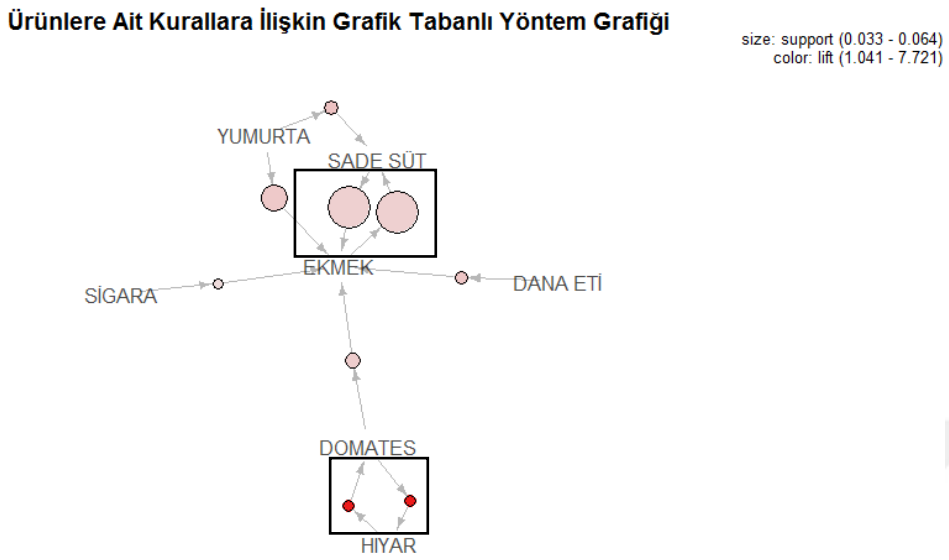


Kaynak: R çıktısı yazar tarafından derlenmiştir.

Grafik 14'te elde edilen tüm kurallar iki üründen oluştuğu için sadece tek bir renk ve sıra değeri gösterilmektedir. Kural sayısı kadar sembolün olduğu grafikte, yüksek destek değerine sahip kuralın düşük güven değerine ve düşük destek değerine sahip başka bir kuralın yüksek güven değerine sahip olduğu gözlenmiştir.

Grafik tabanlı yöntem (Graphics-based Method) grafiği ürünlerden hareketle, kuralların nasıl oluştuğunu incelemektedir. Küçük kural grupları için tercih edilebilir. Hangi kuralda hangi ürünlerin yer aldığı net bir şekilde gösterilmektedir. Grafikte bulunan daireler kuralları temsil etmektedir. Grafikte elde edilen kural sayısı kadar daire bulunmaktadır. Kuralların destek değerlerine göre dairelerin büyüklüğü ve kaldıraç değerlerine göre renkleri şekil almaktadır. Bir üründen daireye doğru giden oklar kuralın öncül tarafını, daireden herhangi bir ürüne doğru giden oklar kuralın ardıl tarafını temsil etmektedir.

Grafik 15: Ürünlere Ait Kurallara İlişkin Grafik Tabanlı Yöntem Grafiği

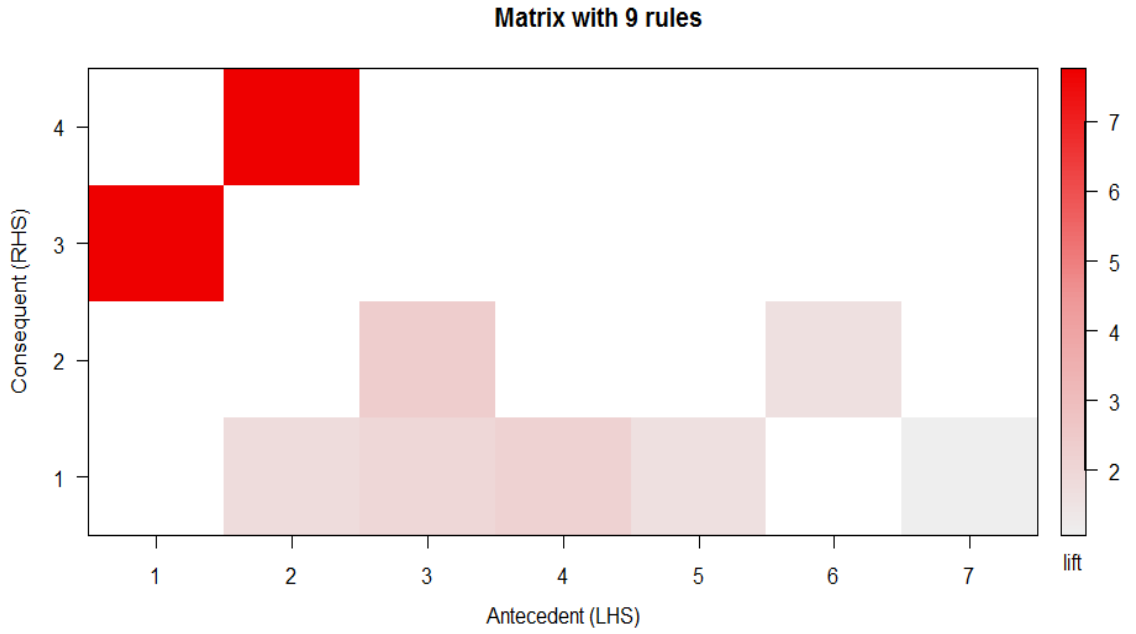


Kaynak: R çıktısı yazar tarafından derlenmiştir.

Grafik 15'te yer alan en büyük dairelere sahip iki kurala göre, alışverişlerde ekmek ve sade sütün beraber görülme olasılığı yüksektir. Grafiğin alt tarafında yer alan koyu renge sahip dairelere göre hıyar ve domates ürünlerinin birbirinin satışlarını etkilediği görülmektedir.

Matris tabanlı yöntem (Matrix-based Method) grafiği, küçük veya orta büyüklükteki kural grupları için tercih edilebilir. Grafikte kuralları kareler temsil etmektedir. Grafiğin x ekseni kuralların öncül tarafında bulunan ürünler ve y ekseni kuralların ardıl tarafında bulunan ürünleri temsil etmektedir. Grafikteki karelerin renkleri güven veya kaldıraç değerlerine göre değişmektedir. Değerler büyüdükçe renkler koyulaşmaktadır.

Grafik 16: Ürünlere Ait Kurallara İlişkin İçin Matris Tabanlı Yöntem Grafiği



Kaynak: R çıktısı yazar tarafından derlenmiştir.

Grafik 16'da kuralların kaldıraç değerine göre renk aldığı matris tabanlı yöntem grafiği gösterilmektedir. Grafik 16'da elde edilen 9 kuralın öncül tarafında 7 farklı ürünün ve ardıl tarafında 4 farklı ürünün bulunduğu gözlenmiştir. Grafiğin sol üst köşesinden sağ alt köşesine gidildikçe, kurallara ait kaldıraç değerleri düşmekte ve karelerin aldığı renkler solmaktadır.

Alt kategoriler baz alınarak başvurulan apriori algoritması uygulamasında model parametreleri, MGD değeri %34 ve MDD değeri %3 olarak atanmıştır. Algoritma sonuçları Tablo 48'de gösterilmektedir.

Tablo 48: Alt Kategori Bazlı Apriori Sonuçları

```
set of 36 rules

rule length distribution (lhs + rhs):sizes
 2  3
25 11

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   2.00    2.00    2.00    2.31    3.00    3.00

summary of quality measures:
    support         confidence        lift           count
 Min.   :0.030   Min.   :0.34   Min.   :1.5   Min.   :25
 1st Qu.:0.031   1st Qu.:0.38   1st Qu.:1.9   1st Qu.:26
 Median :0.035   Median :0.41   Median :2.1   Median :29
 Mean   :0.045   Mean   :0.44   Mean   :2.3   Mean   :37
 3rd Qu.:0.052   3rd Qu.:0.49   3rd Qu.:2.5   3rd Qu.:44
 Max.   :0.089   Max.   :0.81   Max.   :4.6   Max.   :74

mining info:
 data ntransactions support confidence
   TR            830    0.03       0.34
```

Kaynak: R çıktısı yazar tarafından derlenmiştir.[4]

Tablo 48'de görüldüğü üzere %34 MGD ve %3 MDD değeri ile 2 ürün içeren 25 adet tekli ve 3 ürün içeren 11 çoklu birliktelik kuralı elde edilmiştir. Kurallara ait destek, güven, kaldıraç ve görülme sıklığı için en düşük, en yüksek, ortalama ve medyan değerleri hesaplanmıştır. Elde edilen kurallar arasında en düşük destek değeri %3 iken en yüksek güven değeri %81'dir. Kaldıraç değerleri için medyan yani ortanca değer 2,1 iken görülme sıklığı değerleri için ortalama 37 olarak hesaplanmıştır. Kurallar, güven değeri en yüksek kuraldan en düşük kurala göre sıralanarak, Tablo 49'da gösterilmektedir.

Tablo 499: Alt Kategorilere Ait Tekli ve Çoklu Birliktelik Kuralları

Kurallar	Destek Değeri (Support Value) %	Güven Değeri (Confidence Value) %	Kaldıraç Değeri (Lift Value)	Görülme Sıklığı (Count)
{SÜT,YUMURTA} => {PEYNİR}	3%	81%	4,62	25
{PEYNİR,YUMURTA} => {SÜT}	3%	66%	3,41	25
{MEYVE,PEYNİR} => {SEBZE}	3%	60%	2,73	25
{MEYVE,SÜT} => {SEBZE}	3%	54%	2,48	26
{PEYNİR,SEBZE} => {MEYVE}	3%	51%	2,61	25
{SEBZE,SÜT} => {MEYVE}	3%	51%	2,61	26
{YUMURTA} => {EKMEK}	5%	51%	1,96	41
{PEYNİR,SÜT} => {EKMEK}	3%	50%	1,94	29

[4] RStudio çıktıları resim veya PDF olduğundan Türkçe'ye çevrilememiştir.

{EKMEK,PEYNİR} => {SÜT}	3%	50%	2,59	29
{BEYAZ ET} => {SEBZE}	4%	49%	2,23	37
{YUMURTA} => {PEYNİR}	5%	47%	2,69	38
{EKMEK,SÜT} => {PEYNİR}	3%	47%	2,68	29
{MEYVE} => {SEBZE}	9%	46%	2,09	74
{PEYNİR,SÜT} => {YUMURTA}	3%	43%	4,42	25
{PASTA-KEK MLZ} => {SÜT}	4%	43%	2,23	34
{KIRMIZI ET} => {EKMEK}	6%	41%	1,59	48
{BİSKÜVİ} => {ÇİKOLATA/GOFRET}	7%	41%	2,22	59
{SEBZE} => {MEYVE}	9%	41%	2,09	74
{BEYAZ ET} => {EKMEK}	4%	41%	1,58	31
{YUMURTA} => {SEBZE}	4%	41%	1,87	33
{YOĞURT} => {SEBZE}	3%	40%	1,85	27
{YOĞURT} => {EKMEK}	3%	40%	1,56	27
{PEYNİR} => {SÜT}	7%	40%	2,08	58
{PEYNİR} => {EKMEK}	7%	40%	1,55	58
{YOĞURT} => {SÜT}	3%	39%	2,01	26
{SÜT} => {EKMEK}	7%	39%	1,50	62
{ÇİKOLATA/GOFRET} => {BİSKÜVİ}	7%	39%	2,22	59
{YUMURTA} => {SÜT}	4%	38%	1,99	31
{YOĞURT} => {PEYNİR}	3%	37%	2,14	25
{BEYAZ ET} => {PEYNİR}	3%	37%	2,11	28
{SÜT} => {PEYNİR}	7%	36%	2,08	58
{KEK} => {BİSKÜVİ}	3%	36%	2,08	26
{KIRMIZI ET} => {PEYNİR}	5%	36%	2,05	42
{MEYVE,SEBZE} => {SÜT}	3%	35%	1,82	26
{KEK} => {ÇİKOLATA/GOFRET}	3%	35%	1,88	25
{ŞEKERLEME/SAKIZ} => {BİSKÜVİ}	3%	34%	1,97	27

Kaynak: R çıktısı yazar tarafından derlenmiştir.

Tablo 49'da yer alan bazı birliktelik kuralları aşağıda açıklanmıştır.

{SÜT,YUMURTA} => {PEYNİR}$s = \%3$, $c = \%81$ ve $kaldıraç = 4{,}62$

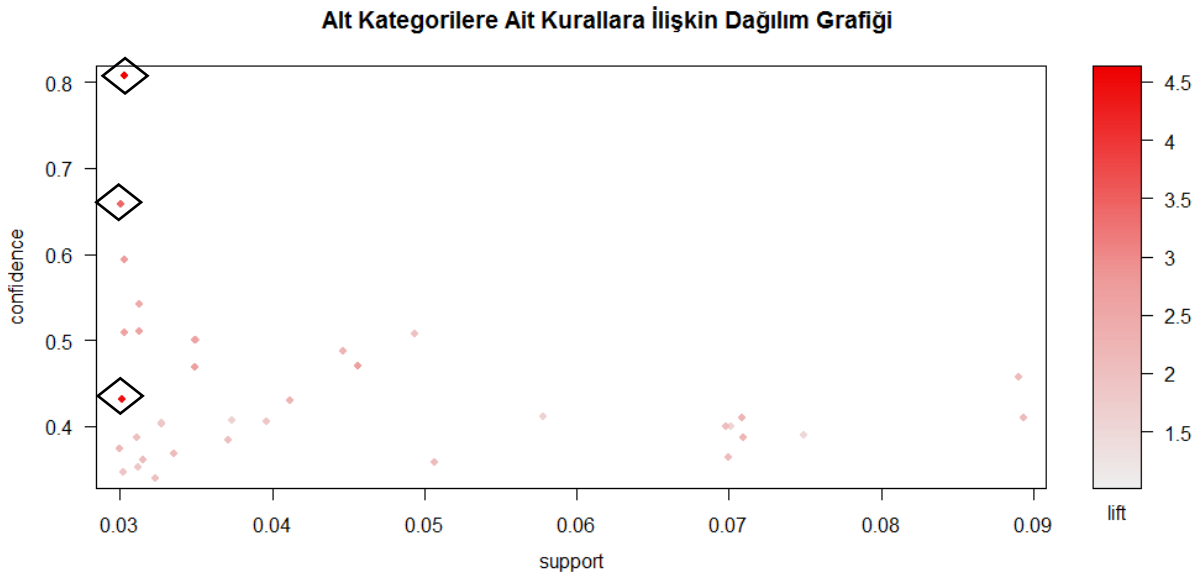
830 fiş kaydı arasında süt, yumurta ve peynir ürünlerinin beraber görülme olasılığı %3,'tür. Süt ve yumurta satın alan müşterilerin %81 olasılık ile peynir satın alma eğilimine girmektedir.

{PEYNİR,YUMURTA} => {SÜT} $s = \%3$, $c = \%66$ ve $kaldıraç = 3{,}41$

830 fiş kaydı arasında peynir, yumurta ve süt ürünlerinin beraber görülme olasılığı %3'tür. Peynir ve yumurta satın alan müşterilerin %66 olasılık ile süt satın alma eğilimine girmektedir.

Alt kategoriler için elde edilen 36 kural bazı grafikler ile görselleştirilmiştir. Grafik 17'de kuralların kaldıraç değerine göre düzenlendiği dağılım grafiği gösterilmektedir.

Grafik 17: Alt Kategorilere Ait Kurallara İlişkin Dağılım Grafiği

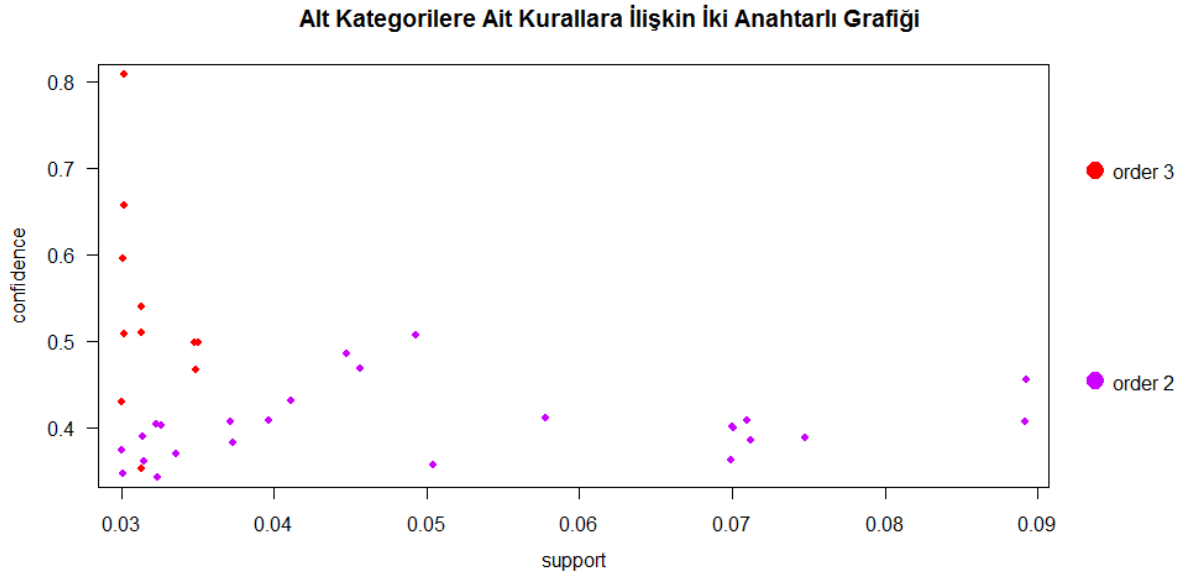


Kaynak: R çıktısı yazar tarafından derlenmiştir.

Grafik 17'de, yüksek kaldıraç değerine sahip kurallar daha koyu renkte gösterilmektedir. En yüksek kaldıraç değerine sahip üç kuralın çok düşük destek değeri aldığı gözlenmektedir.

Alt kategorilerden elde edilen kurallar için Grafik 18'de iki anahtarlı grafik oluşturulmuştur. Grafik 18'de kuralların bazıları 2 ve bazıları 3 alt kategoriden oluştuğu için order ve renk sayısı iki adettir. İki alt kategori içeren kurallar mor ve 3 alt kategori içeren kurallar kırmızı renk ile gösterilmektedir.

Grafik 18: Alt Kategorilere Ait Kurallara İlişkin İki Anahtarlı Grafiği

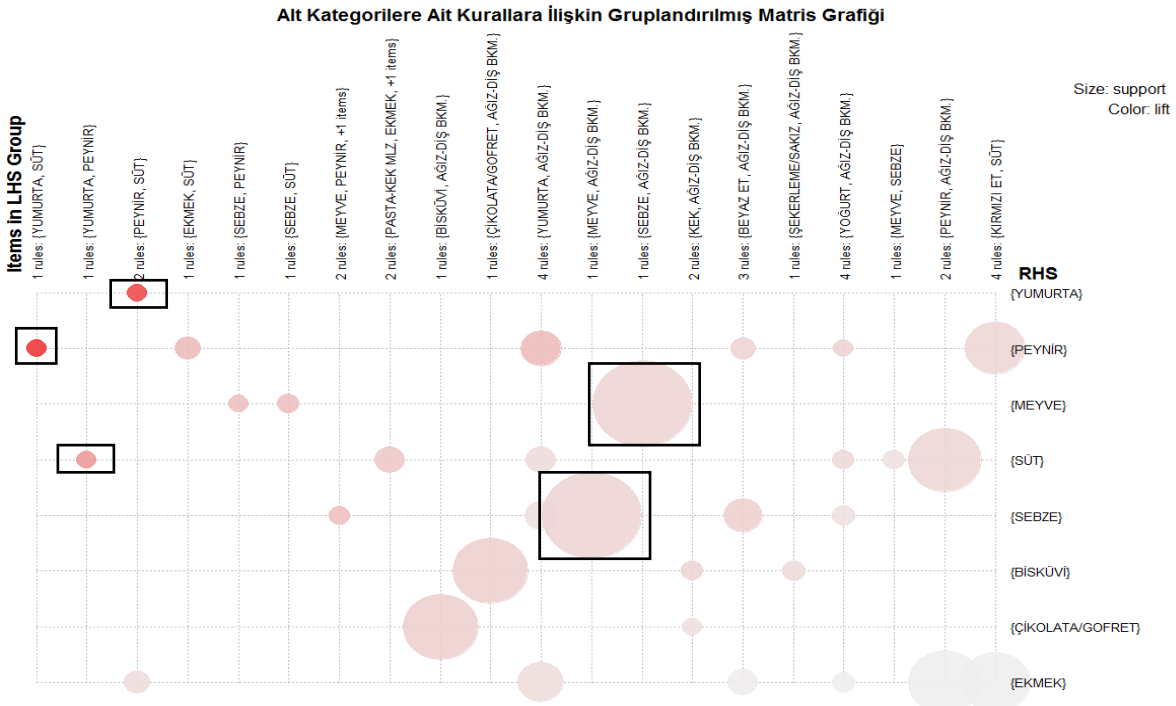


Kaynak: R çıktısı yazar tarafından derlenmiştir.

Grafik 18'de kural sayısı kadar sembol olması ile beraber, 3 alt kategori içeren kuralların iki alt kategori içeren kurallara nazaran düşük destek değerlerine ve genel olarak yüksek güven değerlerine sahip olduğu görülmektedir.

Gruplandırılmış Matris grafiği, kuralın öncül ve ardıl tarafında bulunan ürünlerin kesişim noktalarını daire şekli ile belirtmektedir. Kuralları temsil eden dairelerin büyüklüğü destek ve renkleri kaldıraç değerine göre bağlı olarak değişmektedir. Kaldıraç değeri en yüksek olan kurallar sol üstte ve koyu renkte gösterilir. Destek değerleri arttıkça kuralları temsil eden daireler büyümektedir. Alt kategorilerden elde edilen kurallar için Grafik 19'da gruplandırılmış matris grafiği gösterilmektedir.

Grafik 19: Alt Kategorilere Ait Kurallara İlişkin Gruplandırılmış Matris Grafiği

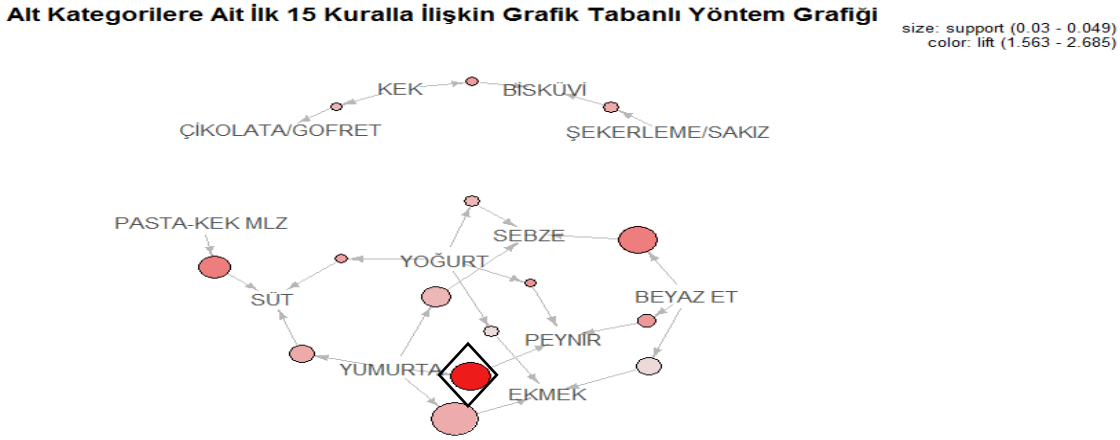


Kaynak: R çıktısı yazar tarafından derlenmiştir.

Grafik 19'a göre, en yüksek kaldıraç değerine sahip 3 kuralın (sol baştaki işaretli kurallar) öncül ve ardıl kısımlarında yumurta, süt ve peynir yer almaktadır. En yüksek destek değerine sahip 2 kuralın öncül ve ardıl kısımlarında sebze, meyve ve ağız ve diş bakımı yer almaktadır.

Alt kategorilerden elde edilen ilk 15 kural için Grafik 20'de grafik tabanlı yöntem grafiği gösterilmektedir. Grafiğe göre, alışverişlerde yumurta, süt ve peynir ürünleri birbirlerinin satışlarını arttırmaktadır.

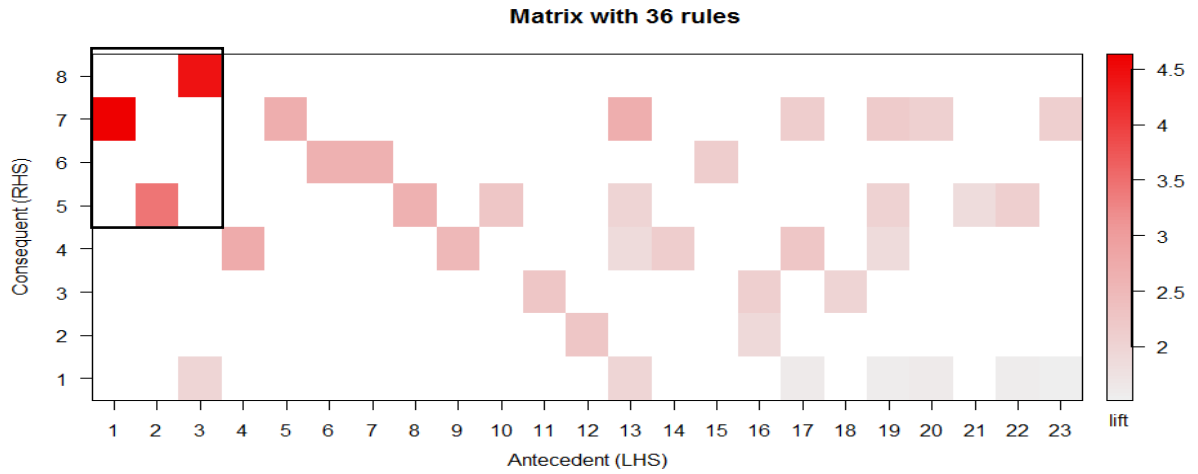
Grafik 20: Alt Kategorilere Ait İlk 15 Kural İçin Grafik Tabanlı Yöntem Grafiği



Kaynak: R çıktısı yazar tarafından derlenmiştir.

Alt kategorilerden elde edilen kurallar için Grafik 21'de matris tabanlı yöntem grafiği gösterilmektedir. Grafiğin x ekseninde kuralların öncül ve y ekseninde kuralların ardıl kısımlarını temsil etmektedir. Grafikte yer alan kareler kaldıraç değerlerine göre şekil almaktadır.

Grafik 21: Alt Kategorilere Ait Kurallara İlişkin İçin Matris Tabanlı Yöntem Grafiği

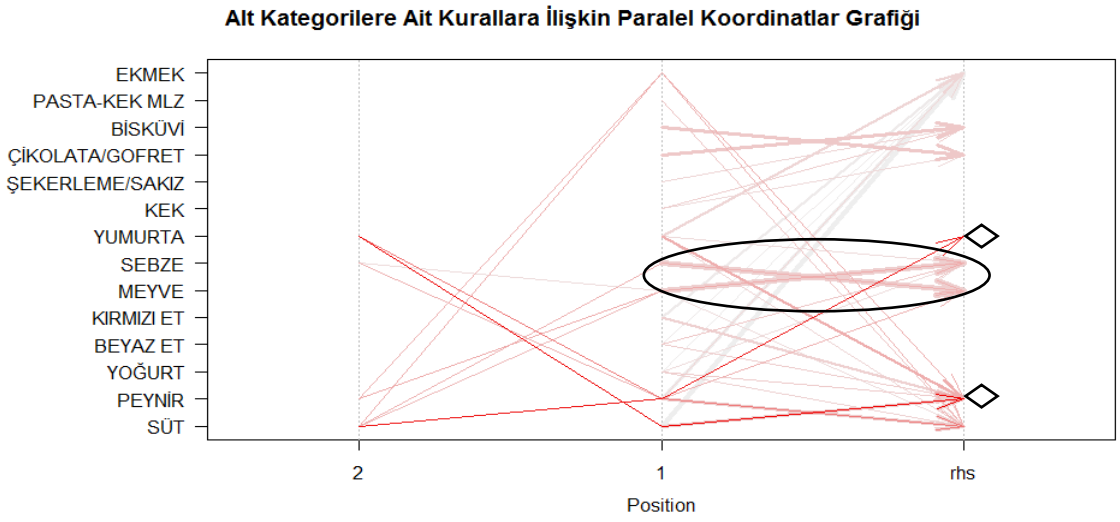


Kaynak: R çıktısı yazar tarafından derlenmiştir.

Grafik 21'de en yüksek kaldıraç değerine sahip 3 kural sol köşede yer almaktadır. Grafikte, sol üst köşeden sağ alt köşeye gidildikçe kurallar için kaldıraç değerleri düşmektedir.

Paralel koordinatlar (Parallel Coordinates) grafiği küçük kural gruplarını analiz etmek için tercih edilebilir. Grafik tabanlı yöntem grafiğinde olduğu gibi kuralların hangi alt kategorilerden oluştuğunu ve hangi alt kategoriler alındığında beraberinde alım olasılığı artan alt kategorilerin hangileri olduğu görülmektedir. Grafiğin x ekseninde alt kategorilerin kuraldaki sırasını belirten pozisyon sırası ile RHS bilgisi bulunmaktadır. Grafiğin y ekseninde ise kurallarda yer alan alt kategorilerin bilgisi bulunmaktadır. Pozisyon eksenindeki 1 değeri alınan ilk alınan alt kategoriyi, 2 değeri ikinci alınan alt kategoriyi ve RHS alınan son alt kategoriyi temsil etmektedir. Grafikte yer alan her bir ok kuralları temsil etmektedir. Bu okların şekli kuralların sahip olduğu destek ve güven değerlerine göre değişmektedir. Yüksek destek değerine sahip kurallar için okların genişliği artar ve yüksek güven değerine sahip kurallar için okların rengi koyulaşmaktadır. Grafik 22'de alt kategorilerden elde edilen kurallar için paralel koordinatlar grafiği gösterilmektedir.

Grafik 22: Alt Kategorilere Ait Kurallara İlişkin Paralel Koordinatlar Grafiği



Kaynak: R çıktısı yazar tarafından derlenmiştir.

Grafik 22'ye göre 2 ve 3 alt kategorinin bulunduğu birden fazla kural olduğu görülmektedir. En yüksek güven değerine sahip iki kural çoklu birliktelik kuralı olmak üzere süt, yumurta ve peynir alt kategorilerini kapsamaktadır. En yüksek destek değerine sahip iki kural tekli birliktelik kuralı olmak üzere meyve ve sebze alt kategorilerini kapsamaktadır.

SONUÇ VE DEĞERLENDİRME

Bu tez çalışmasında veri madenciliği uygulamalarında sıklıkla başvurulan birliktelik kurallarından yararlanılmıştır. Perakende sektöründe uygulanan birliktelik kuralları market sepet analizi olarak adlandırılmaktadır. Çalışmada birliktelik kuralları algoritmaları arasında sıklıkla başvurulan apriori algoritması tercih edilmiştir.

Market sepet analizi, Türkiye'de perakende sektöründe yer alan bir işletmenin belirli dönemlere ait fiş kayıtları üzerinde yapılmıştır. Tezin amacı, çalışma kapsamına giren müşterilerin alışveriş alışkanlıklarını belirlemek ve ürün grupları ile alt kategoriler arasındaki ilişkileri keşfetmektir. Bu bilgiler doğrultusunda, işletmenin satış stratejilerini destekleyici bulgular elde edilmeye çalışılmıştır.

Ürün grupları üzerinde yapılan apriori uygulamasında 9 adet tekli birliktelik kuralı ve alt kategoriler üzerinde yapılan apriori uygulamasında 25 adet tekli ve 11 adet çoklu birliktelik kuralı elde edilmiştir. Kurallar Türkiye'de müşteri kitlelerinde en çok görülen durumları ifade etmektedir.

Ürün gruplarına istinaden elde edilen kurallarda yer alan 7 ürün (hıyar, domates, ekmek, sade süt, yumurta, sigara ve dana eti) müşteriler tarafından en çok tercih edilen ürünlerdir. Bu kurallar, genel olarak Türk hane halkının tüketiminde birincil olarak yer alan ürünleri kapsamaktadır. Türkiye nüfusunun büyük bir çoğunluğu düşük gelire sahip kişilerden oluştuğu için elde edilen kurallar hala geliri kısıtlı kişilerin alışveriş alışkanlıklarını göstermektedir. Ayrıca, insanların alışverişlerinde ilk tercihini ekmekten yana kullanmayıp, alışverişlerinin sonunda ekmek alma eğiliminde oldukları görülmüştür. Bu kuralları oluşturan ilgili 422 fiş kaydının %59'u erkek ve %41'i kadın müşteriler tarafından gerçekleştirilmiştir. Bu müşterilerin büyük bir payı orta yaştaki müşteriler olmakla beraber, genel olarak alışverişlerin %50'sinden fazlası öğle ve akşam saatlerinde yapılmıştır. Alışverişlerin çoğunluğu kredi kartı aracı ile yapılmıştır. Bu demografik bilgiler ışığında, bu kuralları oluşturan müşteri kitlesinin alışverişlerinde en çok temel ihtiyaçların yer alması ve daha çok kredi kartı ödeme aracına başvurması ile düşük gelir seviyesine sahip olduğu, orta yaştaki kadın ve erkek müşterilerinin alışverişlerini çoğunlukla öğle ve akşam saatlerinde tercih ettikleri söylenebilir. Fişlerin bölgelere ve illere göre dağılımı incelendiğinde; %44,8 oranla en çok Marmara bölgesi ve

%36,7 oranla en çok İstanbul ilinde alışveriş gerçekleşmiştir. Türkiye'de bulunan 7 bölge arasında o bölgede en çok alışverişin yapıldığı iller; İstanbul, Ankara, İzmir, İçel, Kastamonu, Tunceli ve Batman şeklindedir. 9 adet tekli kural arasında yer alan önemli kurallar detaylı olarak aşağıda incelenmiştir.

{HIYAR} => {DOMATES} $s = \%3{,}4$, $c = \%65$ ve $kaldıraç = 7{,}7$

{DOMATES} => {HIYAR} $s = \%3{,}4$, $c = \%40$ ve $kaldıraç = 7{,}7$

Yukarıdaki gösterilen iki kuralda yer alan ürünler evrende 85 fişi kapsamaktadır. Bu iki ürünü en çok orta yaş grubundaki erkekler tercih etmektedir. Çoğunluğu İstanbul'da yaşayan müşteriler çoğunlukla öğle saatlerinde kredi kartı ile alışverişlerini yapmayı tercih etmektedirler. Kuralların güven değerleri dikkate alındığında; hıyar alan müşterilerin domates alma olasılığı, domates alan müşterilerin hıyar alma olasılığından daha yüksektir.

{SADE SÜT} => {EKMEK} $s = \%6{,}4$, $c = \%40$ ve $kaldıraç = 1{,}6$

{EKMEK} => {SADE SÜT} $s = \%6{,}4$, $c = \%25$ ve $kaldıraç = 1{,}6$

Yukarıdaki gösterilen iki kuralda yer alan ürünler evrende 289 fişi kapsamaktadır. Bu iki ürünü en çok orta yaş grubundaki erkekler tercih etmektedir. Büyük bir bölümü İstanbul'da yaşayan müşteriler çoğunlukla öğle saatlerinde kredi kartı ile alışverişlerini yapmayı tercih etmektedirler. Kuralların güven değerleri dikkate alındığında; sade süt alan müşterilerin ekmek alma olasılığı, ekmek alan müşterilerin sade süt alma olasılığından daha yüksektir.

Alt kategorilere istinaden el edilen kurallar 14 alt kategoriyi içermektedir. Ürün gruplarına ait kurallarda görüldüğü üzere, alt kategorilere ait kurallarda da müşteriler tarafından en çok tercih edilen ürün ekmek olup, alışverişlerde ilk tercih edilen ürün sırasında yer almamaktadır. 36 kuralı oluşturan ilgili 680 fiş kaydının %58'i erkek ve %42'si kadın müşteriler tarafından gerçekleştirilmiştir. Bu müşterilerin büyük bir payı orta yaştaki müşteriler olmakla beraber, genel olarak alışverişlerin %88'i öğle ve akşam saatlerinde yapılmıştır. Alışverişlerin %50'sinden fazlası kredi kartı aracı ile yapılmıştır. Fişlerin bölgelere ve illere göre dağılımı incelendiğinde; %42,79 oranla en çok Marmara bölgesi ve %33,97 oranla en çok İstanbul ilinde alışveriş gerçekleşmiştir. Türkiye'de bulunan 7 bölge arasında o bölgede en çok alışverişin

yapıldığı iller; İstanbul, Ankara, İzmir, İçel, Kastamonu, Tunceli ve Batman şeklindedir. Ürün gruplarına dair müşteri kitlesi ile alt kategorilere dair müşteri kitlesinin bazı demografik özellikleri örtüşmektedir. Her iki müşteri kitlesinin çoğunluğu İstanbul'da yaşamaktadır. Bu müşteriler, orta yaşta olup alışverişlerini genel olarak öğle ve akşam saatlerinde kredi kartı ile yapmaktadır. Ancak, alt kategorilere dair müşteri kitlesinin tercih ettiği ürünler doğrultusunda biraz daha geliri yüksek müşteriler olduğu düşünülmektedir. 36 adet tekli kural arasında yer alan önemli kurallar detaylı olarak aşağıda incelenmiştir.

{SÜT,YUMURTA} => {PEYNİR}$s = \%3$, $c = \%81$ ve $kaldıraç = 4{,}62$

{PEYNİR,YUMURTA} => {SÜT} $s = \%3$, $c = \%66$ ve $kaldıraç = 3{,}41$

Yukarıdaki gösterilen iki kuralda yer alan alt kategoriler evrende 284 fişi kapsamaktadır. Bu üç alt kategoriyi en çok orta yaş grubundaki erkekler tercih etmektedir. Çoğunluğu İstanbul'da yaşayan müşteriler çoğunlukla öğle saatlerinde kredi kartı ile alışverişlerini yapmayı tercih etmektedirler. Kuralların güven değerleri dikkate alındığında; süt ve yumurta alan müşterilerin peynir alma olasılığı, peynir ve yumurta alan müşterilerin süt alma olasılığından daha yüksektir.

Ürün grupları ve alt kategorilere dair açıklanan 6 kurala göre, Türkiye'de çalışan nüfusun büyük bir payına sahip olan İstanbul'da öğle veya akşam saatlerinde alışveriş yapan müşterilerin ev hanımı veya bir işe sahip erkekler olduğu düşünülmektedir.

Apriori algoritması hem ürün grupları hem de alt kategoriler için çalıştırılmıştır. Ürün grupları ve alt kategoriler baz alınarak elde edilen kuralların gelir durumu dışında aynı çıkarımları sağladığı görülmüştür. Bu çalışmanın devamında, müşteri kitlesini veya ürünler arasındaki ilişkileri daha iyi anlamak için ek analizler yapılabilir.

- Aynı çalışma, İstanbul'da yer alan şubeler baz alınarak, daha büyük bir veri seti ile yapılabilir. Bu kapsam doğrultusunda İstanbul'da alışveriş yapan müşterilerin ilçelere göre dağılımına bakılabilir.
- Veri setinde tercih edilen ve tercih edilmeyen ürün veya alt kategoriler arasında ilişkinin varlığı incelenebilir. Elde edilen kurallar ayrı ayrı incelenerek, kurallara özgü müşteri profilleri çıkartılabilir. Sonrasında bu müşteri profilleri üzerinde kümeleme analizi yapılarak, profiller gruplara ayrılabilir.

- Birliktelik kuralları üzerinde "Çok Boyutlu Ölçekleme" analizi yapılarak, mağaza ve reyon planı çıkartılabilir (Ay & Çil, 2008, s. 25).
- Bir birliktelik kuralında yer alan ürünler dikkate alınarak, karar ağaçları analizi yapılabilir. Böyle bir analiz için birliktelik kurallarında başvurulan veri seti dikkate alınmalıdır. İlgili birliktelik kuralında yer alan ürünlerin aynı anda olduğu fişlere 1 ve diğer fişlere 0 değeri atanır. Bu şekilde hazırlanan veri seti için bir hedef değişken belirlenir. Analiz sonucunda bu değişkeni en çok etkileyen ürünler, bireylerin satın alma biçimini en çok etkileyen ürünler olarak kabul edilir.

KAYNAKÇA

Abul, O., Özdoğan, G. Ö., & Yazıcı, A. (2009). Paralel Veri Madenciliği Algoritmaları. *BAŞARIM'09, 1. Ulusal Yüksek Başarım ve Grid Konferansı* (s. 1-26). Ankara: ODTÜ-KKM.

Acharya, T., & Mitra, S. (2003). *Data Mining Multimedia, Soft Computing and Bioinformatics.* USA: Wiley Publication.

Açık Veri ve Veri Gazeteciliği Derneği. (tarih yok). https://www.avvg.org.tr/yazilar/343-veri-distopyasi-buz-daginin-gorunen-parcasi-cambridge-analytica.html adresinden alındı

Adirans, P., & Zantinge, D. (1997). Data Mining. England: Addison Wesley.

Adlakha, N., Khare, N., & Pardasani, K. R. (2009, September 28). An Algorithm for Mining Multidimensional Fuzzy Assoiation Rules. *International Journal of Computer Science and Information Security (IJCSIS), 5*(1), 72-76. http://sites.google.com/site/ijcsis/ adresinden alındı

Agrawal, R., & Bayardo, R. (1999). Data Mining The Most Interesting Rules. *Proceedings Of Sıgmod Int'l Conference On Knwledge Discovery and Data Mining*, (s. 145-154).

Agrawal, R., & Srikant, R. (1994). Fast Algorithms for Mining Association Rules. *Proceedings of the 20th VLDB Conference*, (s. 487-499). Santiago, Chile. http://rakesh.agrawal-family.com/papers/vldb94apriori.pdf adresinden alındı

Agrawal, R., & Srikant, R. (1995). Mining Sequential Patterns. *In Proceedings of the 11th IEEE International Conference on Data Engineering* (s. 1-12). Taipei, Taiwan: IEEE Computer Society Press.

Agrawal, R., & Srikant, R. (1996). Mining Quantiative Association Rules in Large Relational Tables. *ACM SIGMOD'96 International Conference of Management of Data* (s. 1-12). Montreal: ACM.

Agrawal, R., Imielinski, T., & Swami, A. (1993). Mining Association Rules Between Sets of Items in Large Databases. *SIGMOD '93 Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data. 22*, s. 207-216. Washington: ACM.

Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., & Verkamo, A. I. (1995). Fast Discovery of Association Rules, Advances in Knowledge Discovery and Data Mining. *AAAI/MIT Press*, 307-328.

Akay, D., Doğrul, G., & Kurt, M. (2015). Trafik Kazalarının Birliktelik Kuralları ile Analizi. *Gazi Mühendislik Dergisi, 1*(2), 265-284.

Akbulut, S. (2006). *Veri Madenciliği Teknikleri ile Bir Kozmetik Markanın Ayrılan Müşteri Analizi ve Müşteri Segmentasyonu Yüksek Lisans Tezi.* Ankara: Gazi Üniversitesi Fen Bilimleri Enstitüsü.

Akpınar, H. (2000, Nisan). Veritabanlarında Bilgi Keşfi ve Veri Madenciliği. *İstanbul Üniversitesi İşletme Fakültesi Dergisi, 29*(1), 1-22.

Alagöz, A., Ortakarpuz, M., & Öge, S. (2014). Bir Kurumsal Zekâ Teknolojisi Olarak Veri Madenciliği İle Muhasebe Bilgi Sistemi İlişkisi. *Selçuk Üniversitesi Sosyal Bilimler Enstitüsü Dergisi, 1*(Dr. Mehmet YILDIZ Özel Sayısı (21)), 1-21.

Albayrak, A. S., & Koltan, Yılmaz, Ş. (2009). Veri Madenciliği: Karar Ağacı Algortimaları ve İMKB Verileri Üzerine Bir Uygulama. *Süleyman Demirel Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi, 14*(1), 31-52.

Albayrak, M. (2008, Haziran). *EEG Sinyallerindeki Epileptiform Aktivitenin Veri Madenciliği Süreci ile Tespiti Doktora Tezi.* Sakarya, Sakarya: Sakarya Üniversitesi Fen Bilimleri Enstitüsü.

Alkan, A. (2007, Haziran 8). Finansal Uygulamalarda Veri Madenciliği. 1-48. İstanbul, İstanbul: TBD İstanbul Bilişim Kongresi. http://docplayer.biz.tr/1672580-8-haziran-2007-tbd-istanbul-bilisim-kongresi. html. adresinden alındı

Allen, H., Gearan, P., & Rexer, K. (2015). 2015 Data Science Survey. 1-36. Rexer Analytics. www.RexerAnalytics.com adresinden alındı

Almalı, Özdemir, Z., & Ulucan, Özkul, F. (2011). *İşletmelerde Hile Riski Yönetimi* (1 b.). İstanbul: BETA.

Alpaydın, E. (2000). Zeki Veri Madenciliği: Ham Veriden Altın Bilgiye Ulaşma Yöntemleri. *Bilişim 2000 Veri Madenciliği Eğitim Semineri*, (s. 1-10). İstanbul.

Argüden, Y., & Erşahin, B. (2008). *Veri Madenciliği Veriden Bilgiye, Masraftan Değere* (1 b.). İstanbul: Arge Danışmanlık A.Ş. www.arge.com adresinden alındı

Atılgan, E. (2011). *Karayollarında Meydana Gelen Trafik Kazalarının Karar Ağaçları ve Birliktelik Kuralı ile Analiz Edilmesi.*

Atre, S., & Moss, L. T. (2003). *Business Intelligence Roadmap: The Complete Project Lifecycle for Decision-Support Applications.* USA: Addison-Wesley Publishing.

Ay, D., & Çil, İ. (2008). Migros Türk A.Ş.de Birliktelik Kurallarının Yerleşim Düzeni Planlamada Kullanılması. *Endüstri Mühendisliği Dergisi, 21*(2), 14-29.

Ayad, A. M. (2000). *A New Algorithm for Incremental Mining of Constrained Association Rules Master Thesis.* EGYPT: Alexandria University.

Ayad, A., Nagwa, E.-M., & Taha , Y. (2001). Incremental Minig of Constrained Association Rules. *Proceedings of the 2001 SIAM International Conference on Data Mining.* Society for Industrial and Applied Mathematics (SIAM).

Aydın, C., & Bakır, M. A. (2010). *İstatistik* (3 b.). Ankara, Ankara: Nobel Yayın Dağıtım.

Babadağ, K. K. (2003). *Veri Madenciliği Yaklaşımı ve Veri Kalitesinin Artması İçin Kullanılması Basılmamış Uzmanlık Tezi.* Ankara: Türkiye İstatistik Kurumu (TÜİK).

Başkent Üniversitesi. (tarih yok). http://mail.baskent.edu.tr: http://mail.baskent.edu.tr/~20394676/0302/bil483/HW2.pdf adresinden alındı

Baykal, A. (2006). Veri Madenciliği Uygulama Alanları. *Dicle Üniversitesi Ziya Gökalp Eğitim Fakültesi Dergisi, 7*, 95-107.

Baykasoğlu, A. (2005, Şubat 2-4). Veri Madenciliği Ve Çimento Sektöründe Bir Uygulama. *Akademik Bilişim 2005.* Gaziantep: İnternet Teknolojileri Derneği.

Berry, M. J., & Linoff, G. S. (2004). *Data Mining Techniques for Marketing, Sales and Customer Relationship Management* (2 b.). Wiley Publishing, Inc., Indianapolis.

Berson, A., Smith, S., & Thearling, K. (1999). *Building Data Mining Applications for CRM.* U.S.A.: McGraw-Hill.

Birant, D., Kut, A., Altınok, B., Altınok, H., Altınok, E., Ihlamur, M., & Ventura, M. (2010). İş Zekâsı Çözümleri İçin Çok Boyutlu Birliktelik Kuralları Analizi. *Akademik Bilişim'10- Xıı. Akademik Bilişim Konferansı Bildirileri* (s. 257-263). Muğla: Muğla Üniversitesi.

Boran, L., & Özkan, M. (2014). Veri Madenciliğinin Finansal Kararlarda Kullanımı. *Çankırı Karatekin Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi, 4*(1), 59-82.

Bramer, M. (2007). *Principles of Data Mining* (2 b.). London : Springer-Verlag.

Breiman, L., Friedman, J., Olshen, R., & Stone, C. J. (1987). *Classification and Regression Trees.*

Brin, S., Motwani, R., & Silverstein, C. (1997). Beyond Market Baskets: Generalizing Association Rules to Correlations. *Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data* (s. 265-276). Tucson, Arizona, USA: ACM.

Brin, S., Motwani, R., Ullman, J. D., & Tsur, S. (1997). Dynamic Itemset Counting and Implication Rules for Market Basket Data. *SIGMOD '97 Proceedings of the 1997 ACM SIGMOD international conference on Management of data. 26*, s. 255-264. Tucson, Arizona, USA: ACM SIGMOD.

Buckles, B. P., Yuan, X., Yuan, Z., & Zhang, J. (2002). Mining Negative Association Rules. *Seventh International Symposium on Computers and Communications (ISCC)* (s. 623-629). Taormina-Giardini Naxos, Italy: IEEE.

Buldu, A., Doğan, B., & Erol, B. (2014). Sigortacılık Sektöründe Müşteri İlişkileri Yönetimi için Birliktelik Kurallarının Kullanılması. *Marmara Fen Bilimleri Dergisi, 3*, 105-114.

Cai, C. H., Cheng, C. H., Fu, A. W., & Kwong, W. W. (1998, In Proceedings of 1998 International Database Engineering and Applications Symposium). Mining Association Rules with Weighted Items. 68-77. Cardiff, Wales.

Cai, Y., Cercone, N., & Han, J. (1993, February 1). Data-Driven Discovery Ofquantitative Rules İn Relational Databases. *IEEE Transactions on Knowledge and Data Engineering, 5*(1), 29-40.

Canbay, M., & Duru , N. (2007). Veri Madenciliği ile Deprem Verilerinin Analizi. *International Earthquake Symposium* (s. 556-560). Kocaeli: Kocaeli Üniversitesi.

Cavique, L. (2007, November). A Scalable Algorithm for the Market Basket Analysis. *Journal of Retailing and Consumer Services, 14*(6), 400-407.

Chakaravarthy, V. T., Pandit, V., & Sabharwal, Y. (2009). Analysis of Sampling Techniques for Association Rule Mining. *Proceedings of the 12th International Conference on Catabase Theory* (s. 276-283). St. Petersburg, Russia,: ACM.

Chen, C.-H., Hong, T.-P., Lan, G.-C., & Lin, S.-B. (2016, April). Mining Fuzzy Temporal Association Rules By Item Lifespans. *Applied Soft Computing, 41*(C), 265-274.

Chen, M. (2007, November). Ranking Discovered Rules from Data Mining with Multiple Criteria by Data Envelopment Analysis. *Expert Systems with Applications, 33*(4), 1110-1116.

Chen, M. S., Han, J., & Yu , P. S. (1996, December 1). Data Mining: An Overview from Database Perspective. *IEEE Transactions on Knowledge and Data Engineering, 8*(6), 866-883.

Chen, M. S., Park, J. S., & Yu, P. (1995). Efficient Parallel Data Mining for Association Rules. *Proceedings of the International Conference on Information and Knowledge Management*, (s. 31-36). Baltimore, Maryland.

Chen, P. (1976, March). The Entity Relationship Model - Toward a Unified View of Data. *ACM Transactions on Database Systems, 1*(1), 311-339.

Chen, Q., He, S. B., Gao, W. W., Sun, Y. K., & Wang, Y. J. (2008, Kasım 18). The Research of Multidimensional Association Rule in Traffic Accidents. *2008 4th International*

Conference on Wireless Communications, Networking and Mobile Computing (s. 1-4). Dalian, China: IEEE.

Chen, Y. L., Hu, Y. H., Shen, R. J., & Tang, K. (2005, August). Market Basket Analysis in A Multiple Store Enviroment. *Decision Support Systems, 40*(2), 339-354.

Codd, E. F. (1970, June). A Relational Model of Data for Large Shared Data Banks. *Communications of the ACM, 13*(6), 377-387.

Çağıltay, N. E. (2010). *İş Zekası ve Veri Ambarı Sistemleri* (1 b.). Ankara, Ankara, TÜRKİYE: Odtü Geliştirme Vakfı Yayıncılık.

Çağlayan, E., & Güriş, S. (2010). *Ekonometri: Temel Kavramlar* (3 b.). İstanbul, İstanbul: Der Yayınları.

Çakır, F., & Akgöbek, Ö. (2009). Veri Madenciliğinde Bir Uzman Sistem Tasarımı. *Akademik Bilişim'09 - XI. Akademik Bilişim Konferansı Bildirileri*, (s. 801-806). Şanlıurfa.

Çamurcu, A. Y., & Doğan, B. (2008, August 20). Association Rule Mining from an Intelligent Tutor. *Journal of Educational Technology Systems, 36*(4), 433-447.

Çamurcu, A. Y., & Özçakır, F. C. (2007). Birliktelik Kuralı Yöntemi İçin Bir Veri Madenciliği Yazılımı Tasarımı ve Uygulaması. *İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi*(12), 21-37.

Çilingirtürk, A. M. (2011). *Veri Analizi* (1 b.). İstanbul: Seçkin Yayıncılık.

Çıngı, H. (2007). Veri Madenciliğine Giriş Ders Notları. 1-35. Ankara. http://yunus.hacettepe.edu.tr/~hcingi/ist376a/6Bolum.doc adresinden alındı

Çokluk, Ö. (2010, Temmuz 1). Lojistik Regresyon Analizi: Kavram ve Uygulama. *10*(3), 1357-1407.

Das, A., Ng, W. K., & Woon, Y. K. (2001). Rapid Association Rule Mining. *Proceedings of The Tenth International Conference on Information and Knowledge Management* (s. 474-481). Atlanta, Georgia, USA: ACM.

Daştan, A. (2008). *Bilgi ve Eğitim Teknolojilerinde Yaşanan Gelişmelerin Muhasebe Eğitimine Etkisi: Türkiye Değerlendirmesi.* Ankara: Sermaye Piyasası Kurulu Yayınları.

Davenport, T. (2014). Big Data@Work. *Big Data@Work*. İstanbul: BZD Yayın ve İletişim Hizmetleri (Türk Hava Yolları Yayınları).

Demirörs, O., Özcan, Ö., & Usgurlu, B. (2010). A Clustering Based Functional Similarity Measurement Approach. C. P. IEEE (Dü.), *IEEE, Conference Proceedings of 36th EUROMICRO Conference on Software Engineering and Advanced Applications* içinde (s. 371-375). IEEE.

Diler, S. (2016). *Veri Madenciliği Süreçleri ve Karar Ağaçları Algoritmaları ile Bir Uygulama, Yüksek Lisans Tezi.* Van: Yüzüncü Yıl Üniversitesi Fen Bilimleri Enstitüsü.

Dinçer, E., & Duru , N. (2006). Gırtlak Kanseri Ameliyat Verilerinin K-means Yöntemiyle Analizi. *2, 1*.

Doğan, N., & Özdamar, K. (2003, Ekim 5). CHAID Analizi ve Aile Planlaması İle İlgili Bir Uygulama. *Türkiye Klinikleri Tıp Bilimleri Dergisi, 23*(5), 392-398.

Doğan, O. (2015). Bir E-Ticaret Sitesi Kullanıcı Hesaplarında Şifre Yapılarının Birliktelik Kuralları ile İncelenmesi. *İnternet Uygulamaları ve Yönetimi Dergisi, 6*(2), 49-61.

Doğan, Ş., & Türkoğlu, İ. (2008, Şubat 2). Hypothyroidi and Hyperthyroidi Detection From Thyroid Hormone Parameters by Using Decision Trees. *Doğu Anadolu Bölgesi Araştırmaları Dergisi, 5*(2), 163-169.

Dolgun, M. Ö. (2006). *Büyük Alışveriş Merkezleri için Veri Madenciliği Uygulamaları, Yüksek Lisans Tezi.* Ankara: Hacettepe Üniversitesi Fen Bilimleri Enstitüsü.

Dolgun, Ö. (2015, Ağustos 13). *Özgür Dolgun Veri Madenciliği ile ilgili Herşey.* http://ozgurdolgun.com: http://ozgurdolgun.com/?p=19 adresinden alındı

Döşlü, A. (2008). *Veri Madenciliğinde Market Sepet Analizi ve Birliktelik Kurallarının Belirlenmesi, Yüksek Lisans Tezi.* İstanbul: Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü.

Dunham, M. H., Gruenwald, L., Hossain, Z., & Xiao, Y. (2000). *A Survey of Association Rules Teknik Rapor.* Southern Methodist University Department of Computer Science, TR00-CSE-8.

Duru, N., & Pehlivanoğlu, M. K. (2015). Veri Madenciliği Teknikleri Kullanılarak Ortaokul Öğrencilerinin Sosyal Ağ Kullanım Analizi: Kocaeli İli Örneği. *Düzce Üniversitesi Bilim ve Teknoloji Dergisi, 3*, 508-517.

Edelstein, H. A. (1999). *Introduction to Data Mining and Knowledge Discovery* (3 b.). U.S.A.: Two Crows Corporation.

Eker, H. (2004). *Veri Madenciliği veya Bilgi Keşfi*. http://www.bilgiyonetimi.org/cm/pages/mkl_gos.php?nt=538 adresinden alındı

Elder, J. F., & Pregibon, D. (1995). A Statistical Perspective on KDD. *The 1st International Conference on Knowledge Discovery and Data Mining*, (s. 87-93). Montreal.

Erar, A. (1985). Bağlanım (Regresyon) Çözümlemesi Ders Notları. Ankara: Hacettepe Üniversitesi Fen Fakültesi İstatistik Bölümü.

Erbaş, S. O., & Olmuş, H. (2003). Bayes Ağlarda Koşullu Bağımsızlıkların İncelenmesi üzerine bir Çalışma. *TÜİK İstatistik Araştırma Dergisi, 2*(1), 89-103.

Erdem , S., & Özdağoğlu, G. (2008). Ege Bölgesi'ndeki bir Araştırma ve Uygulama Hastanesinin Acil Hasta Verilerinin Veri Madenciliği ile Analiz Edilmesi. *Anadolu Üniversitesi Bilim ve Teknoloji Dergisi, 9*(2), 261-270.

Erdoğan, M. (2006). *Denetim Kavramsal ve Teknolojik Yapı* (3 b.). Ankara: Maliye ve Hukuk Yayınları.

Erdoğan, Ş. Z. (2004). *Veri Madenciliği Ve Veri Madenciliğinde Kullanılan K-Means Algoritmasının Öğrenci Veri Tabanında Uygulanması, Yüksek Lisans Tezi.* İstanbul: İstanbul Üniversitesi Sosyal Bilimler Enstitüsü.

Erpolat, S. (2012). Otomobil Yetkili Servislerinde Birliktelik Kurallarının Belirlenmesinde Apriori ve FP-Growth Algoritmalarının Karşılaştırılması. *Anadolu Üniversitesi Sosyal Bilimler Dergisi, 12*(1), 137-146.

exastax. (2017, 12 14). *Türkiye Büyük Veri Piyasasına Genel Bakış: Büyük Veri*. 6 17, 2018 tarihinde Exastax: https://www.exastax.com.tr/buyuk-veri/turkiye-buyuk-veri-piyasasina-genel-bakis/ adresinden alındı

Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P., & Uthurusamy, R. (1996). *Advances in Knowledge Discovery and Data.* California: American Association for Artificial Intelligence Series- AAAI.

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). From Data Mining to Knowledge Discovery in Databases. *AI Magazine, 17*(3), 37-54.

Flach, P. A., & Lachiche, N. (2001). Confirmation-Guided Discovery of First- Order Rules with Tertius. *Machine Learning, 42*, 61-95.

Flank, A. (2004, September 7). *Multirelational Association Rule Mining.*

Frawley, W. J., Matheus, C. J., & Shapiro, G. P. (1992). Knowledge Discovery in Databases: An Overview. *AI Magazine, 13*(3), 57-70.

Friedman, J., Hastie, T., & Tibshirani, R. (2001). *The Elements of Statistical Learning: Data Mining, Inference and Prediction.* USA: Springer Bilim ve İş Medyası.

Gargano, M. L., & Raggad, B. G. (1999). Data Mining -A Powerful Information Creating Tool. *OCLC Systems & Services, 15*(2), 81-90.

Geatz, M. W., & Roiger, R. J. (2003). Data Mining: A Tutorial-Based Primer. USA.

Ghosh, A., & Nath, B. (2004, June 14). Multi-Objective Rule Mining Using Genetic Algorithms. *Information Sciences-Informatics and Computer Science, Intelligent Systems, Applications, 163*(1-3), 123-133.

Giudici, P., & Passerone , G. (2002, February 28). Data Mining of Association Structures to Model Consumer Statistics & Data Analysis. *Computational Statistics & Data Analysis - Nonlinear Methods and Data Mining, 38*(4), 533-541.

Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning* (1 b.). USA: Addison-Wesley Publishing Company, Inc.

Gordon, S. L., & Michael , J. B. (2011). *Data Mining Techniques* (3 b.).

Gosain, A., & Maneela, B. (2013). A Comprehensive Survey of Association Rules on Quantitative Data in Data Mining. *Proceedings of 2013 IEEE Conference on Information and Communication Technologies (ICT 2013).* IEEE.

Gökay Emel, G., & Taşkın, Ç. (2002). Genetik Algoritmalar ve Uygulama Alanları. *Uludağ Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi, 21*(1), 129-152.

Gökay Emel, G., & Taşkın, Ç. (2005). Pazarlama Stratejilerinin Oluşturulmasında Bir Karar Destek Aracı: Birliktelik Kuralları Madenciliği. *Dokuz Eylül Üniversitesi Sosyal Bilimler Enstitüsü Dergisi, 7*(3), 30-59.

Gökay, E. G., & Taşkın, Ç. (2005, Aralık). Veri Madenciliğinde Karar Ağaçları ve Bir Satış Analizi Uygulaması. *Eskişehir Osmangazi Üniversitesi Sosyal Bilimler Dergisi, 6*(2), 221-239.

Gökmen, Ş. (2014). *Müşteri İlişkileri Yönetiminde Bir Araç Olarak Veri Madenciliği ve Perakende Sektöründe Bir Uygulama, Yüksek Lisans Tezi.* İstanbul: Yıldız Teknik Üniversitesi Sosyal Bilimler Enstitüsü.

Grün, B., Hahsler, M., & Hornik, K. (2005). A Computational Environment For Mining Association Rules And Frequent Item Sets. *Journal Of Statistical Software, 14*, 1-25.

Güçdemir, D. (2018, Nisan 14). *Veri Distopyası: Buz dağının görünen parçası Cambridge Analytica.* Açık Veri & Veri Gazeteciliği Derneği: https://www.avvg.org.tr/yazilar/343-veri-distopyasi-buz-daginin-gorunen-parcasi-cambridge-analytica.html adresinden alındı

Gürbüz, F., Özbakır, L., & Yapıcı , H. (2009). Türkiye'de Bir Havayolu İşletmesine Ait Parça. *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi, 24*(1), 73-78.

Gürgen, G. (2008). *Birliktelik Kuralları İle Sepet Analizi ve Uygulaması, Yüksek Lisans Tezi.* İstanbul: Marmara Üniversitesi Sosyal Bilimler Enstitüsü.

Gürsakal, N. (2001). *Sosyal Bilimlerde Araştırma Yöntemleri.* Bursa: Vipaş.

Güvenen, O. (2011, Ekim). İstatistik Bilimi, Etik, Dünya Dinamikleri, Bilgi Tahrifatı ve Karar Sistemlerine Etkileri. *TÜİK İstatistik Araştırma Dergisi, 8*(2), 1-12.

Győrödi, C., Győrödi, R., & Holban, S. (2004). A Comparative Study of Association Rules Mining Algorithms. *SACI 1st Romanian-Hungarian Joint Symposium on Applied Computational Intelligence*, (s. 213-222). Timisoara, Romania.

Hahsler, M., Hornik, K., & Grün, B. (2005). A Computational Environment For Mining Association Rules And Frequent Item Sets. *Journal Of Statistical Software, 14*, 1-25.

Han, E. H., Karypis, G., & Kumar, V. (1997). Scalable Parallel Data Mining For Association Rules. *SIGMOD '97 Proceedings of the 1997 ACM SIGMOD international conference on Management of data. 26*, s. 277-288. Tucson, Arizona, USA: ACM SIGMOD.

Han, J., & Kamber, M. (2001). *Data Mining: Consepts and Tecniques.* San Francisco: Morgan Kaufmann Publishers, Inc.

Han, J., & Kamber, M. (2006). *Data Mining Concepts and Techniques* (2 b.). San Francisco: Morgan Kauffmann Publishers Inc.

Han, J., Kamber, M., & Pei, J. (2012). *Data Mining: Concepts and Techniques* (3 b.). USA: Morgan Kaufmann Publishers.

Han, J., Pei, H., & Yin, Y. (2000). Mining Frequent Patterns without Candidate Generation. *SIGMOD '00 Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data. 29.* New York, USA: ACM SIGMOD.

Hand, D. J. (1998, December). Data Mining: Statistics and More? *52(2)*, 112-118. The American Statistician.

Hidber, C. (1999). Online Association Rule Mining. *SIGMOD '99 Proceedings of the 1999 ACM SIGMOD international conference on Management of data. 28*, s. 145-156. Philadelphia, Pennsylvania, USA: ACM SIGMOD.

Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems.* USA: University Of Michigan Press.

Houtsma, M., & Swami, A. (1993). *Set-Oriented Mining for Association Rules in Relational Databases.* IBM Almaden Research Center, Research Report RJ 9567, San Jose.

Hsiao, C. J., & Zaki, M. J. (2002). CHARM: An Efficient Algorithm for Closed Itemset Mining. *Proceedings of the 2002 SIAM International Conference on Data Mining* (s. 457-473). SIAM.

İnceoğlu, M., & Vahaplar, A. (2003). Veri Madenciliği ve Elektronik Ticaret. *Türkiye' de İnternet Konferansları.* İnternet Kullanıcıları Derneği. http://inet-tr.org.tr/inetconf7/bildiriler/78.doc adresinden alındı

Inmon, W. H. (1992). *Building the Data Warehouse.* New York, USA: John Wiley & Sons, Inc.

Inmon, W. H. (1996). The Data Warehouse and Data Mining. *39(11)*, 49-50. New York, USA: ACM.

Jacops, P. (1999). *Data Mining: What General Managers Need to Know* (Cilt 4). Harvard Management Update.

Kanellopoulos, D., & Kotsiantis, S. (2006). Association Rules Mining: A Recent Overview. *GESTS International Transactions on Computer Science and Engineering, 32*(1), 71-82.

Karaağaoğlu, E. (2012). Veri Madenciliği ve Yeni Nesil Yöntemleri. *n'den N'ye Gezinti İstatistik Dergisi*(9), 34-38.

Karabatak, M. (2018). *Özellik Seçimi, Sınıflama ve Öngörü Uygulamalarına Yönelik Birliktelik Kuralı Çıkarımı ve Yazılım Geliştirilmesi Doktora Tezi.* Elazığ: Fırat Üniversitesi Fen Bilimleri Enstitüsü.

Karabatak, M., & İnce, M. C. (2004). Apriori Algoritması ile Öğrenci Başarısı Analizi. Elazığ: Fırat Üniversitesii.

Karaibrahimoğlu, A. (2014, AĞUSTOS). *Veri Madenciliğinden Birliktelik Kuralı İle Onkoloji Verilerinin Analiz Edilmesi: Meram Tıp Fakültesi Onkoloji Örneği, Doktora Tezi.* Konya: Selçuk Üniversitesi Fen Bilimleri Enstitüsü.

Kaya, M., & Keleş, A. E. (2014, Şubat 5-7). Duvar İnşa Edilmesinde Verimliliği Etkileyen Faktörlerin Apriori Veri Madenciliği Yöntemi Kullanılarak Analizi. *Akademik Bilişim'14 - XVI. Akademik Bilişim Konferansı Bildirileri* (s. 831-836). Mersin: Mersin Üniversitesi.

Kayri, M. (2014, Haziran 6). Karar Ağaçları. Muş: Muş Alparslan Üniversitesi.

KDnuggets. (2010, May). Data Mining / Analytic Tools Used Poll.

KDnuggets. (tarih yok). Veri madenciliğinde Tercih Clementine. http://arsiv.ntv.com.tr/news/170054.asp adresinden alındı

Kitsuregawa, M., & Shintani, T. (1996). Hash Based Parallel Algorithms for Mining Association Rules. *Fourth International Conference on Parallel and Distributed Information Systems* (s. 1-12). USA: Proceedings of PDIS.

Kotler, P. (2002). *Kotler ve Pazarlama* (2 b.). (A. Özyağcılar, Çev.) İstanbul: Sistem Yayıncılık.

Kotler, P. (2005). *A' Dan Z' Ye Pazarlama* (5 b.). (A. Kalem, Bakkal, Çev.) Mediacat Yayınları.

Kotler, P. (2007). *Soru ve Cevaplarla Günümüzde Pazarlamanın Temelleri* (1 b.). (Ü. Şensoy, Çev.) İstanbul: Optimist Yayınları.

Köse, Y. (2015). *Değerli Müşterilerde Ürün KAtegorileri Arasındaki Satış ilişkilerinin Veri Madenciliği Yöntemlerinden Birliktelik Kuralları ve Kümeleme Analizi İle Belirlenmesi ve Ulusal Bir Perakendecide Örnek Bir Uygulama, Yüksek Lisans Tezi.* Konya: Selçuk Üniversitesi Sosyal Bilimler Enstitüsü.

Kurt, Ü., Tokatlı, F., & Türe, M. (2009). Using Kaplan-Meier Analysis Together With Decision Tree Methods (C&RT, CHAID, QUEST, C4.5 and ID3) in Determining Recurrence-Free Survival of Breast Cancer Patients. *Expert Systems with Applications, 36*, 2017-2026.

Lakshmanan, C. B., Srinivasan, V., & Ponoraja, C. (2015). Data Mining with Decision Tree to Evaluate the Pattern on Efectiveness of Treatment for Pulmonary Tuberculosis: A Clustering and Classification Techniques. *3*, 43-48. Scientific Research Journal (SCIRJ).

Larose, D. T. (2005). *Discovering Knowledge in Data: An Introduction to Data Mining.* USA: John and Wiley Sons Incorporated.

Li, H., & Sun, J. (2008, Şubat). Data Mining Method for Listed Companies, Financial Distress Prediction. *Knowledge-Based Systems, 21*(1), 1-5.

Lin, W. Y., Tseng, M. C., & Wang, M. F. (2004). OLAM Cube Selection in On-line Multidimensional Association Rules Mining System. *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems. 3214*, s. 1276-1282. Berlin Heidelberg: Springer Verlag.

Liu, J., Yin, S., Zeng, Y., & Zhang, M. (2015, January). Research of Improved FP-Growth Algorithm in Association Rules Mining. *Scientific Programming, 2015*, 1-6.

Love, B. (1993). Enterprise Information Technologies.

Lu, Y., & Song, Y. Y. (2015). Decision Tree Methods: Applications for Classification and Prediction. *Shanghai Archives of Psychiatry, 27*(2), 130-135.

Mannila, H., & Raiha, K. J. (1987). Dependency Inference. *The 1987 International Conference Very Large Data Bases*, (s. 155-158). Brighton, England.

Mannila, H., Toıvonen, H., & Verkamo, I. A. (1994). Efficient Algorithms for Discovering Association Rules. *AAAIWS'94 Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining* (s. 181-192). Seattle, WA: AAAI.

Mcculloch, W. S., & Pitts, W. (1943). A Logical Calculus of Ideas Immanent in Nervous Activity. *Bulletin of Mathematical Biology, 5*, 115-133.

Mild, A., & Reutterer , T. (2003, May 1). An Improved Collaborative Filtering Approach for Predicting Cross-Category Purchases on Binary Market Basket Data. *Journal of Retailing and Consumer Services, 10*(3), 123-133.

Mitchell, M. (1999). *An Introduction to Genetic Algorithm.* England: The MIT Press.

Mocan, G. (2016). *Perakendecilikte Veri Madenciliği Uygulamaları ve Sorunları, Yüksek Lisans Tezi.* İstanbul: Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü.

Mutter, S. (2004). *Classification using Association Rules Yüksek Lisans Tezi.* Yeni Zelanda: Waikato Üniversitesi.

Navathe, S. B., Omiecinski, E., & Savasere, A. (1998). Mining for Strong Negative Associations in a Large Database of Customer Transactions. *Proceedings of the*

Fourteenth International Conference on Data Engineering (s. 494-502). Washington, USA: IEEE Computer Society.

OECD, Online Kütüphanesi. (2014). Data Mining-Related Scientific Articles. http://www.oecd-ilibrary.org/science-and-technology/measuring-the-digital-economy/data-mining-related-scientific-articles-1995-2014_9789264221796-graph6-en. adresinden alındı

Oğuzlar, A. (2003, Temmuz-Aralık). Veri Ön İşleme. *Erciyes Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*(21), 67-76.

Oğuzlar, A. (2004). CART Analizi ile Hane Halkı İş Gücü Anketi Sonuçlarının Özetlenmesi. *Atatürk Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi, 18*(3-4), 79-90.

Oğuzlar, A. (2004). *Veri Madenciliğine Giriş*. Bursa: Ekin Kitabevi.

Omiecinski, E., Navathe, S., & Savasere, A. (1995, September). An Efficient Algorithm for Mining Association Rules in Large Databases. *VLDB '95 Proceedings of the 21th International Conference on Very Large Data Bases* (s. 432-444). San Francisco: Morgan Kaufmann Publishers Inc.

Öğüt, S. (2009). Veri Madenciliği Kavramı ve Gelişim Süreci. İstanbul. http://www.sertacogut.com/blog/wp-content/uploads/2009/03/sertac_ogut_-_veri_madenciligi_kavrami_ve_gelisim_sureci.pdf adresinden alındı

Özçalıcı, M. (2017, Mart). Veri Madenciliğinde Birliktelik Kuralları ve İkinci El Otomobil Piyasası Üzerine Bir Uygulama. *Ordu Üniversitesi Sosyal Bilimler Araştırmaları Dergisi, 7*(1), 45-58.

Özekes, S. (2003). Veri Madenciliği Modelleri ve Uygulama Alanları. *İstanbul Ticaret Üniversitesi Dergisi, 2*(3), 65-82.

Özkan, M. (2014). Veri Madenciliğinin Finansal Kararlarda Kullanımı. *Çankırı Karatekin Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi, 4*(1), 17.

Özkan, Y. (2013). *Veri Madenciliği Yöntemleri* (2 b.). İstanbul: Papatya Yayınevi.

Özker, Dursun, G. (2016). *İç Kontrol SistemindeBir Araç Olarak Verş Madenciliğinin Kullanılmasının İşletmenin Bilançosuna Etkisi, Yüksek Lisans Tezi.* İstanbul: Marmara Üniversitesi Sosyal Bilimler Üniversitesi.

Özmen, Ş. (2001). İş Hayatı Veri Madenciliği ile İstatistik Uygulamalarını Yeniden Keşfediyor. *5. Ulusal Ekonometri ve İstatistik Sempozyumu.*

Özmen, Ş. (2003). Veri Madenciliği Süreci. *Veri Madenciliği ve Uygulama Alanları Konferansı.* İstanbul: İstanbul Ticaret Üniversitesi.

Öztemel, E. (2006). *Yapay Sinir Ağları* (2 b.). İstanbul: Papatya Yayıncılık.

Pandey, A., & Pardasani, K. R. (2009, June). Rough Set Model for Discovering Multidimensional Association Rules. *International Journal of Computer Science and Network Security (IJCSNS), 9*(6), 159-164.

Pehlivanlı, D. (2010). *Modern İç Denetim* (1 b.). İstanbul: BETA.

Pektaş, A. O. (2013). *SPSS ile Veri Madenciliği* (1 b.). İstanbul: Dikeyeksen Yayın Dağıtım, Yazılım ve Eğitim Hizmetleri San. Ve Tic. Ltd. Şti.

Piatetsky Shapiro, G. (1990). Knowledge Discovery in Real Databases: A Report on the IJCAI-89 Workshop. *AI Magazine, 11*(5), 68-70.

Poel, D. V., Schamphelaere, J. D., & Wets, G. (2004). Direct and Indirect Efects of Retail Promotions on Sales and Profits in the Do-It-Yourself Market. *27*, 53-62.

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning.* Morgan Kaufmann Publishers.

Rexer, K. (2013). *2013 Data Miner Survey/ Summary Report.* Rexer Analytics. www.RexerAnalytics.com adresinden alındı

Savaş, S., Topaloğlu, N., & Yılmaz, M. (2011). Veri Madenciliği ve Türkiye'deki Uygulama Örnekleri. *İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi*(21), 1-23.

Scheffer, T. (2001). Finding Association Rules that Trade Support Optimally Against Confidence. *Proceedings of the 5th European Conference on Principles and Practice of Knowledge Discovery in Databases Conference*, (s. 424-435). Freiburg, Almanya.

Silahtaroğlu, G. (2008). *Kavram ve Algoritmalarıyla Temel Veri Madenciliği* (1 b.). İstanbul: Papatya Yayıncılık.

Silahtaroğlu, G. (2013). *Veri Madenciliği Kavram ve Algoritmaları* (2 b.). İstanbul: Papatya Yayıncılık Eğitim A.Ş.

SPSS, AnwerTree Algorithm Summary. (1999). *SPSS, AnwerTree Algorithm Summary, SPSS White Paper*. USA.

Sug, H. (2003). Comparison of Multidimensional Association Rules with Decision Trees for Large Database. *Proceedings of the International Conference on Artificial Intelligence*, (s. 121-126). Las Vegas, Nevada, USA.

Şeker, S. E. (2005). Veri Ambarı. *2(4)*. YBS Ansiklopedi.

Şeker, S. E. (2015, Aralık 4). Veri Ambarı (Data Warehouse). *Veri Ambarı [Electronic Version], 2*. Yönetim Bilişim Sistemleri Ansiklopedisi (YBS). www.YBSAnsiklopedi .com adresinden alındı

Şen, Z. (2001). *Bulanık Mantık ve Modelleme İlkeleri.* İstanbul: Bilge Sanat Yapım Yay. Tant. Kağ. Turz. San. Tic. Ltd. Şti.

Şenesen, Ü. (2013). *İstatistik- Sayıların Arkasını Anlamak.* Literatür Yayıncılık.

Şentürk, A. (2006). *Veri Madenciliği Kavram ve Teknikler* (1 b.). Bursa: Ekin Kitabevi.

Şimşek, Gürsoy, U. T. (2011). *Uygulamalı Veri Madenciliği Sektörel Analizler.* Pegem Akademi.

Şimşek, U. T., & Timor, M. (2008, Şubat). Veri Madenciliğinde Sepet Analizi ile Tüketici Davranışı Modellemesi. *İstanbul Üniversitesi İşletme Fakültesi İşletme İktisadı Enstitüsü Dergisi, 19*(59), 3-10.

Tatlıdil, H. (1992). *Uygulamalı Çok Değişkenli İstatistiksel Analiz.* Ankara.

Thuraisingham, B. (2003). *Web Data Mining and Applications in Bussiness Intelligence and Counter- Terrorism.* U.S.A.: CRC Press.

Toivonen, H. (1996). Sampling Large Databases for Association Rules. *VLDB '96 Proceedings of the 22th International Conference on Very Large Data Bases* (s. 134-145). San Francisco, USA: Morgan Kaufmann Publishers Inc.

Tüzüntürk, S. (2010). Veri Madenciliği ve İstatistik. *Uludağ Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi, 29*(1), 65-90.

Vahaplar, A. (2003). *Bir Coğrafi Veri Madenciliği Uygulaması, Yüksek Lisans Tezi.* İzmir: Ege Üniversitesi Fen Bilimleri Enstitüsü.

Vahaplar, A., & İnceoğlu, M. (2003). Veri Madenciliği ve Elektronik Ticaret. http://inet-tr.org.tr/inetconf7/bildiriler/78.doc adresinden alındı

Watson, R. T. (2002). *Data Management Databases and Organizations*. New York: Wiley Sons Inc.

wikipedia. (tarih yok). https://tr.wikipedia.org/wiki/Veri adresinden alındı

wikipedia. (tarih yok). https://tr.wikipedia.org/wiki/Enformasyon adresinden alındı

Witten, I. H., & Frank, E. (2005). *Data Mining Practical Machine Learning Tools and Techniques - WEKA* (2 b.). San Francisco, USA: Morgan Kaufmann Publishers.

Xu, W. X., & Wang, R. J. (2006). A Novel Algorithm of Mining Multidimensional Association Rules. *International Conference on Intelligent Computing. 344*, s. 771-777. Kunming, China: Springer Verlag.

Yaralıoğlu, K. (2013). http://www.deu.edu.tr/userweb/k.yaralioglu/dosyalar/ver_mad.doc adresinden alındı

Yarımağan, Ü. (2000). *Veri Tabanı Sistemleri* (2 b.). Ankara: Akademi&Türkiye Bilişim Vakfı Ortak Yayını.

Yay, O. (2014, 6 4). 6 17, 2018 tarihinde bidunyasi.wordpress.com: https://bidunyasi.wordpress.com/2014/06/04/is-zekasi-business-intelligence/ adresinden alındı

Yıldırım, İ. E. (2015). *İstatistiksel Araştırma Yöntemleri* (1 b.). Ankara: Seçkin Yayıncılık.

Yünel, Y. (2010, Mayıs). *K-Means Kümeleme Algoritmasının Genetik Algoritma Kullanılarak Geliştirilmesi.* İstanbul: İTÜ Fen Edebiyat Fakültesi Matematik Mühendisliği Bölümü.

Zaki , M. J. (1999). Parallel and Distributed Association Mining: A Survey.

Zaki, M. J. (1997). Evaluation of sampling for data mining of Association rules. *Proceedings Seventh International Workshop on Research Issues in Data Engineering. High Performance Database Management for Large-Scale Applications.* IEEE.

Zaki, M. J., Li, W., & Parthasarathy, S. (1997). A Localized Algorithm for Parallel Association Mining. *SPAA'97 Proceedings of the 9th ACM Symposium on Parallel Algorithms and Architectures*, (s. 321-330). Newport, Rhode Island, USA.

Zhang, C., Zhang, S., & Wu, X. (2004, July). Efficient Mining of Both Positive and Negative Association Rules. *ACM Transactions on Information Systems (TOIS), 22*(3), 381-405.

Zhong, N., & Zhou, L. (1999, APRIL). Methodologies for Knowledge Discovery and Data Mining. *Third Pacific-Asia Conference, Pakdd-99*. Beijing, China: Springer Verlag.

Ziarko, W. (1991, January). The Discovery, Analysis, and Representation of Data Dependencies in Databases, Knowledge Discovery in Databases. 195-212. AAAI/MIT Press.

EKLER

EK-1 Nicel Birliktelik Kuralları Yönteminde Boolean Birliktelik Kuralları Sistemi Yaklaşımı

Aşağıdaki Tablo 50'de Tablo 22'de verilen yaş değişkeni Boolean Birliktelik Kuralları sistemi yaklaşımına dayanarak, 4 gruba ayrılmıştır. Gruplandırma sonrasında yeni veri setinin ve değişkenlerin gösterimi Tablo 51'de gösterilmektedir. Daha sonrasında hesaplamalarda kolaylık sağlaması açısından yaş gruplarına tamsayı olacak şekilde kodlar verilmiştir. Bu süreç, yaş değişkeninde olduğu gibi evlilik değişkeni için de gerçekleştirilmiştir. Tablo 53'te evlilik değişkeninin iki gruba ayrılmış hali mevcuttur. Düzenlenmiş yeni veri setinden hareketle, Tablo 55'teki sık rastlanan öge kümeler elde edilmiştir. Araştırma sonucunda elde edilen birliktelik kuralları Tablo 24'te gösterilmektedir.

Tablo 50: Yaş Grupları

Interval
20…24
25…29
30…34
35…39

Kaynak: AGRAWAL, R. ve SRİKANT R. (1996). Mining Quantiative Association Rules İn Large Relational Tables. ACM SIGMOD'96 International Conference of Management of Data. Montreal. p.1-12. ss.:5.

Tablo 51: Yaş Değişkeninin Gruplanması Sonrası Durum

Kayıt No	Yaş	Evlilik	Araba Sayısı
100	20…24	Hayır	0
200	25…29	Evet	1
300	25…29	Hayır	1
400	30…34	Evet	2
500	35…39	Evet	2

Kaynak: AGRAWAL, R. ve SRİKANT R. (1996). Mining Quantiative Association Rules İn Large Relational Tables. ACM SIGMOD'96 International Conference of Management of Data. Montreal. p.1-12. ss.:5.

Tablo 52: Planlanan Yaş Değişkeni

Aralık	Tamsayı
20…24	1
25…29	2
30…34	3
35…39	4

Kaynak: AGRAWAL, R. ve SRİKANT R. (1996). Mining Quantiative Association Rules İn Large Relational Tables. ACM SIGMOD'96 International Conference of Management of Data. Montreal. p.1-12. ss.:5.

Tablo 53: Planlanan Evlilik Değişkeni

Aralık	Tamsayı
Evet	1
Hayır	2

Kaynak: AGRAWAL, R. ve SRİKANT R. (1996). Mining Quantiative Association Rules İn Large Relational Tables. ACM SIGMOD'96 International Conference of Management of Data. Montreal. p.1-12. ss.:5.

Tablo 54: Planlama Sonrası Değişkenlerin Durumu

Kayıt No	Yaş	Evlilik	Araba Sayısı
100	1	2	0
200	2	1	1
300	2	2	1
400	3	1	2
500	4	1	2

Kaynak: AGRAWAL, R. ve SRİKANT R. (1996). Mining Quantiative Association Rules İn Large Relational Tables. ACM SIGMOD'96 International Conference of Management of Data. Montreal. p.1-12. ss.:5.

Tablo 55: Sık Tekrarlanan Öge Kümeler

Ürün Kümesi	Destek
{<Yaş: 20-29>}	3
{<Yaş: 30-39>}	2
{<Evlilik: Evet>}	3
{<Evlilik: Hayır>}	2
{<Araba Sayısı: 0>}	2
{<Araba Sayısı: 0-1>}	3
{<Yaş: 30-39> ve <Evlilik: Evet>}	2

Kaynak: AGRAWAL, R. ve SRİKANT R. (1996). Mining Quantiative Association Rules İn Large Relational Tables. ACM SIGMOD'96 International Conference of Management of Data. Montreal. p.1-12. ss.:5.

EK-2 RStudio'da Geliştirilen Komutlar

- Excel programında RStudio'a veri aktarma

```
data <- read_excel("rdata.xlsx")
```

- RStudio'da veri düzenleme

```
data <- data[complete.cases(data), ]

data %>% mutate(SKU = as.factor(SKU))

data %>% mutate(CITY = as.factor(CITY))

data$INVOCEDATE <- as.Date(data$DATE)

TransTime <- format(data$INVOCEDATE,"%H:%M:%S")

FIS_ID <- as.numeric(as.character(data$FIS_ID))

cbind(data,TransTime)

cbind(data,FIS_ID)

glimpse(data)
```

- Transaction data seti oluşturma

```
Transactiondata <- ddply(data,c("FIS_ID","INVOCEDATE"),function(df1)paste(df1$SKU,
collapse = ","))
```

- Transaction data seti görüntüleme

```
Transactiondata
```

- Transaction datayı metin belgesi olarak dışarı aktarma

write.table(Transactiondata, file='C:/Users/pc/Desktop/klasör adı/çıktılar klasörü/Transactiondata.txt', sep = "\t",row.names = F)

- Birleştirilen fiş numarası sütunu ve alışveriş zamanı sütunlarını veri transaction data setinden çıkarma

Transactiondata$FIS_ID <- NULL

Transactiondata$INVOCEDATE <- NULL

- Düzenlenen transaction datayı metin belgesi olarak dışarı aktarma

write.table(Transactiondata, file='C:/Users/pc/Desktop/klasör adı/çıktılar klasörü/Transactiondata2.txt', sep = "\t",row.names = F)

- Düzenlenen transaction datayı CSV Excel uzantılı dosya olarak dışarı aktarma

write.csv(Transactiondata,"C:/Users/pc/ Desktop/klasör adı/çıktılar klasörü/market_basket_transactions.csv", quote = FALSE, row.names = TRUE)

- Dışarı aktarılan CSV uzantılı dosyayı içeri aktarma

TR <- read.transactions('C:/Users/pc/ Desktop/klasör adı/çıktılar klasörü/market_basket_transactions.csv',

format = c("basket"),

sep=",",

cols = NULL,

skip = 1,

rm.duplicates = TRUE)

- Transaction data hakkında özet bilgiler oluşturma

```
summary(TR)

s <- summary(TR)

s@itemSummary

s@lengths
```

- Transaction data hakkında tanımlayıcı bilgiler oluşturma

```
typeof(TR)

class(TR)

dim(TR)

length(TR)

str(TR)
```

- Transaction datada yer alan ilk 10 Items'i listeleme

```
inspect(TR[1:10])
```

- ItemSetList Oluşturma

```
ItemSetList <- TR@itemInfo

ItemSetList
```

- ItemSetList Görüntüleme

```
View(ItemSetList)
```

- Itemselist dışarı aktarma

```
write.table(ItemSetList, file='C:/Users/pc/Desktop/klasör adı/çıktılar klasörü/ıtemsetlist.txt', sep = "\t",row.names = F)
```

- En çok tekrar eden ilk 20 Items için ="Absolute Item Frequency Plot" grafiği oluşturma

itemFrequencyPlot(TR,topN=20,type="absolute",col=brewer.pal(8,'Pastel2'), main="Absolute Item Frequency Plot")

- En çok tekrar eden ilk 20 Items için "Relative Item Frequency Plot" grafiği oluşturma

itemFrequencyPlot(TR,topN=20,type="relative",col=brewer.pal(8,'Pastel2'), main="Relative Item Frequency Plot")

- Apriori algoritmasının çalıştırılması için parametrelere değer atama

Öncelikle, MDD ve MGD değerleri atanır. Atanmaması durumunda, program MDD değerine %10 ve MGD değerine %80 atayarak, algoritmayı başlatır.

options(digits = 2)

min_supp <- 0.03

min_conf <- 0.25

min_lenght <- 2

- Kural Oluşturma

rules <- apriori(TR, parameter = list(supp = min_supp, conf = min_conf,

minlen = min_lenght, target = "rules"))

- Kurallar hakkında özet bilgiler oluşturma

summary(rules)

- Kuralları CSV dosya uzantısı ile Excel programına aktarma

rules_DF <- as(rules, "data.frame")

write.csv2(rules_DF, file = "C:/Users/pc/Desktop/klasör adı/çıktılar klasörü/rules.csv", row.names = FALSE)

- Kuralları html uzantısı ile kaydetme

html_page <- inspectDT(rules)

saveWidget(html_page, file = paste0("C:/Users/pc/Desktop/son/SKU analiz/çıktılar/", "BasketRules.html"), selfcontained = FALSE)

- Model ve değişkenleri.RData olarak Output dizini içerisinde yer alan BasketRules.RData dosyasına kaydetme

save.image(file = "C:/Users/pc/Desktop/son/SKU analiz/çıktılar/BasketRules.RData")

- Kuralları güven değerlerine göre sıralama ve görüntüleme

rules_conf <- arules::sort(rules, by = "confidence", decreasing = TRUE)

inspect(head(rules_conf, 9))

- Dağılım grafiği oluşturma

plot(rules, main="Scatter Plot for Association Rules")

- İki anahtarlı grafiği oluşturma

plot(rules, shading="order", control=list(main = "Two-key plot",

col=rainbow(5)))

- Gruplandırılmış matris grafiği oluşturma

plot(rules, method="grouped", main="Grouped Matrix for rules")

- Grafik tabanlı yöntem grafiği oluşturma

plot(rules[1:9], method="graph", control=list(alpha="1"), main="Graph Method for rules")

- Matris tabanlı yöntem grafiği oluşturma

```
plot(rules, method = "matrix", measure = c("support", "confidence"), shading = "lift")
```

- Paralel koordinatlar grafiği oluşturma

```
iplot(rules, method ="paracoord", control = list(reorder = TRUE), main="GraphName6")
```