



**T.C.
İSTANBUL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**



DOKTORA TEZİ

**VERİ MADENCİLİĞİNDE BİRLİKTELİK YÖNTEMLERİ VE
MÜŞTERİ İLİŞKİLERİ YÖNETİMİNE İLİŞKİN BİR UYGULAMA**

Gökçe KARAHAN ADALI

Enformatik Anabilim Dalı

Enformatik Programı

**DANIŞMAN
Prof. Dr. M. Erdal BALABAN**

Şubat, 2017

İSTANBUL

Bu çalışma 15.02.2017 tarihinde ařağıdaki jüri tarafından Enformatik Anabilim Dalı Enformatik programında Doktora Tezi olarak kabul edilmiştir.

Tez Jürisi



İmza

Prof. Dr. M. Erdal BALABAN (Danışman)
İstanbul Üniversitesi
İşletme Fakültesi



İmza

Prof. Dr. Sevinç GÜLSEÇEN
İstanbul Üniversitesi
Enformatik Bölümü



İmza

Prof. Dr. Zuhul TANRIKULU
Boğaziçi Üniversitesi
Uygulamalı Bilimler Yüksek Okulu



İmza

Prof. Dr. Çiğdem ARICIGİL ÇILAN
İstanbul Üniversitesi
İşletme Fakültesi



İmza

Prof. Dr. İbrahim DOĞAN
Marmara Üniversitesi
İşletme Fakültesi



20.04.2016 tarihli resmi gazetede yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince; Bu Lisansüstü teze, İstanbul Üniversitesi'nin aboneli olduğu intihal yazılım programı kullanılarak Fen Bilimleri Enstitüsü'nün belirlemiş olduğu ölçütlere uygun rapor alınmıştır.

ÖNSÖZ

Gerek akademik donanımı, gerek insani yönü ile kendisini rol model olarak benimsediğim çok değerli danışman hocam Sayın Prof. Dr. M. Erdal BALABAN'a doktora öğrenimim ve bu tez konusunun belirlenmesinden, tamamlanmasına kadar geçen her aşamada bana katmış olduğu donanım, vermiş olduğu destek ve göstermiş olduğu anlayış için sonsuz teşekkürlerimi sunarım.

Doktora öğrenimim sırasında bana göstermiş olduğu her türlü destek ve hoşgöründen dolayı değerli hocam Prof. Dr. Sevinç GÜLSEÇEN'e teşekkür ederim.

Tez çalışması için gerekli verinin teminini sağlayan başta danışmanım Prof. Dr. M. Erdal BALABAN olmak üzere, Sayın Suha ÇİLMİ'ye, verinin analizi aşamasında sorularımı cevapsız bırakmayan ve tam destek gördüğüm Sayın Cüneyt SARAÇCI ve Harun SARIÇAM'a katkılarından dolayı teşekkür ederim.

Tez çalışmasının gelişmesindeki değerli katkıları ve her zaman göstermiş olduğu pozitif yaklaşımı ile destek veren değerli hocam Yrd. Doç. Dr. Yalçın ÖZKAN'a teşekkür ederim.

Tezimin her aşamasında, bana vermiş olduğu her türlü destek, moral, çalışmalarına katmış olduğu değerli katkılarını asla unutmayacağım kıymetli arkadaşım Dr.Elif KARTAL' a en içten dileklerle teşekkürlerimi sunarım.

Çalışmamda değerli katkılarını benden esirgemeyen sevgili arkadaşım Dr. Zeki ÖZEN'e teşekkür ederim.

Tezimi hiçbir zaman benden desteğini ve teşviklerini esirgemeyen, babam Veli KARAHAN, annem Emel KARAHAN, bu tezin hazırlanma süresince göstermiş olduğu sabır ve çalışmalarına katmış olduğu kıymetli katkıları ile sevgili eşim Şükran ADALI ve varlığı ile en büyük destekçim olan biricik oğlum Ruşen Mert ADALI'ya ithaf ederim.

Şubat 2017

Gökçe KARAHAN ADALI

İÇİNDEKİLER

Sayfa No

ÖNSÖZ.....	iv
İÇİNDEKİLER.....	v
ŞEKİL LİSTESİ.....	viii
TABLO LİSTESİ.....	xi
SİMGE VE KISALTMA LİSTESİ	xii
ÖZET.....	xiii
SUMMARY.....	xiv
1. GİRİŞ.....	1
2. GENEL KISIMLAR	4
2.1. VERİ MADENCİLİĞİ.....	4
2.2. VERİ MADENCİLİĞİ SÜREÇLERİ.....	8
2.2.1. Problemin Tanımlanması.....	10
2.2.2. Veriyi Anlama	11
2.2.3. Verilerin Hazırlanması	11
2.2.3.1. Veri Toplama (Collection).....	12
2.2.3.2. Veriye Değer Bıçme (Assessment)	12
2.2.3.3. Veri Birleştirme ve Temizleme.....	12
2.2.3.4. Veri İndirgeme	14
2.2.3.5 Veri Dönüştürme (Transformation)	15
2.2.4. Modelin Kurulması ve Değerlendirilmesi	17
2.2.5. Modelin Kullanılması	18
2.2.6. Modelin İzlenmesi.....	18
2.3. VERİ MADENCİLİĞİ MODELLERİ	18
2.3.1. Sınıflama ve Regresyon Modelleri.....	20
2.3.1.1. Karar (Sınıflandırma) Ağaçları.....	21
2.3.1.2. Yapay Sinir Ağları (YSA).....	21
2.3.1.3. Genetik Algoritmalar	21
2.3.1.4. Bellek Tabanlı Yöntemler (K En Yakın Komşu)	22

2.3.1.5. Naive Bayes	22
2.3.2. Kümeleme Modelleri (Clustering)	22
2.3.3. Birliktelik Kuralları ve Ardışık Zamanlı Örüntüler	23
2.4. BİRLİKTELİK KURALLARI	25
2.4.1. Birliktelik Kuralları Temel Kavramları	25
2.4.2. Birliktelik Kurallarının Çeşitleri	29
2.4.2.1. Tek Seviyeli Birliktelik Kuralları	29
2.4.2.2 Çok Seviyeli Birliktelik Kuralları	30
2.4.2.3. Çok Boyutlu Birliktelik Kuralları	32
2.4.3. Birliktelik Kurallarında Başvurulan Diğer Ölçütler	33
2.4.4. Kısıt Tabanlı Birliktelik Analizi (Constraint-Based Association Mining) ...	36
2.4.5. Kantitatif Birliktelik Kurallarının Çıkarılması	38
2.5. MARKET SEPET ANALİZİ	40
2.6. BİRLİKTELİK KURALLARININ BELİRLENMESİNDE KULLANILAN TEMEL ALGORİTMALAR	42
2.6.1. Apriori Algoritması	42
2.6.1.1. Apriori Algoritmasının Adımları	44
2.6.1.2. Apriori Algoritmasının Kısıtları	48
2.6.2. Eclat Algoritması	50
2.6.3. Fp-Growth Algoritması	51
2.6.4. NBMiner Algoritması	52
2.7. MÜŞTERİ İLİŞKİLERİ YÖNETİMİ (CRM) VE VERİ MADENCİLİĞİ	53
2.7.1. Müşteri İlişkileri Yönetimi	53
2.7.2. Veri Madenciliğinde CRM	53
2.7.2.1. Müşteri Segmentasyonu	54
2.7.2.2. Doğrudan Pazarlama Kampanyaları	55
2.7.2.3. Market Sepet ve Sıralı Analizler	56
3. MALZEME VE YÖNTEM	57
3.1. PROBLEMİN TANIMLANMASI	57
3.2. VERİYİ ANLAMA	58
3.3. VERİNİN HAZIRLANMASI	76
3.4. MODELLEME	77
4. BULGULAR	79
4.1. APRIORI ALGORİTMASINDAN ELDE EDİLEN BULGULAR	79
4.2. ECLAT ALGORİTMASINDAN İLE ELDE EDİLEN BULGULAR	91

4.3. NBMINER ALGORİTASINDAN ELDE EDİLEN BULGULAR.....	98
4.4. ALGORİTMA PERFORMANSLARININ KARŞILAŞTIRMASI	98
4.5. UYGULAMANIN SHINY İLE WEB ORTAMINA TAŞINMASI	101
5. TARTIŞMA VE SONUÇ	110
KAYNAKLAR.....	115
EKLER.....	126
ÖZGEÇMİŞ	133



ŞEKİL LİSTESİ

Sayfa No

Şekil 2.1: Veri Madenciliği Disiplinlerarası İlişkisi (Akpınar, 2014).	5
Şekil 2.2: Veritabanı Sistemlerinin Tarihsel Gelişimi.....	7
Şekil 2.3: Veri Tabanlarında Bilgi Keşfi Süreci (Fayyad ve diğ., 1996).	9
Şekil 2.4: CRISP-DM Süreç Diyagramı (Shearer, 2000).	10
Şekil 2.5: Veri Bütünleştirme (Dondurmacı ve Çınar, 2014).	13
Şekil 2.6: Veri İndirgeme Yöntemleri (Witen ve Frank, 2005).	15
Şekil 2.7: Tanımlayıcı ve Tahmin Edici Veri Madenciliği Modelleri.....	20
Şekil 2.8: Taksonomi Örneği (Lai ve Cerpa, 2001).	30
Şekil 2.9: Çok Seviyeli B. Kurallarda Destek Değeri (Han ve Kamber, 2006).....	31
Şekil 2.10: Çok Seviyeli Kurallarda Destek Değeri_2 (Han ve Kamber, 2006).	31
Şekil 2.11: İki Boyutlu Şebeke Gösterimi(Han ve Kamber, 2006).....	40
Şekil 2.12: Genişlik Öncelikli Arama (Tan Ve Kumar, 2005).	43
Şekil 2.13: Apriori Algoritmasının Adımları (Tan ve diğ., 2005).	47
Şekil 2.14: Verilere Bölümlere Ayırarak Madencilik (Han ve Kamber, 2006).	49
Şekil 2.15: CRM yaşam döngüsü (Tsipsis ve Chorianopoulos, 2009).	55
Şekil 3.1: 2014 Yılı Veri Setinde Yer Alan Niteliklere ait Tanımlayıcı Bilgi	63
Şekil 3.2: 2015 Yılı Veri Setinde Yer Alan Niteliklere ait Tanımlayıcı Bilgi	63
Şekil 3.3: 2014 Yılı Kutu Grafikleri.....	64
Şekil 3.4: 2015 Yılı Kutu Grafikleri.....	64
Şekil 3.5: 2014 Yılı Histogramlar.	65
Şekil 3.6: 2015 Yılı Histogramlar.	66
Şekil 3.7: 2014 Yılı Müşteri Ciro Pareto Analizi.....	67
Şekil 3.8: 2014 Yılı Marka Ciro Pareto Analizi.....	67
Şekil 3.9: 2014 Yılı Bölge Ciro Pareto Analizi.	68
Şekil 3.10: 2015 Yılı Müşteri Ciro Pareto Analizi.....	68
Şekil 3.11: 2015 Yılı Marka Ciro Pareto Analizi.....	69
Şekil 3.12: 2015 Yılı Bölge Ciro Pareto Analizi.....	69

Şekil 3.13: 2014-2015 Aylara Göre Cirolar.....	72
Şekil 3.14: Çalışmada izlenen adımları gösteren akış şeması.....	78
Şekil 4.1: Apriori Algoritmasında Kullanılan Veri Seti.....	80
Şekil 4.2: Apriori Algoritması ile 2014 Yılı Birliktelik Kuralları_1.....	81
Şekil 4.3: Apriori Algoritması ile 2014 Yılı Birliktelik Kuralları_2.....	81
Şekil 4.4: Apriori Algoritması ile 2014 Yılı, Birliktelik Kuralları_3.....	82
Şekil 4.5: Apriori Algoritması ile 2014 Yılı Birliktelik Kuralları_4.....	83
Şekil 4.6: Atık Kuralların Budanmasına İlişkin Kullanılan Kodlar.....	84
Şekil 4.7: Apriori Algoritması 2014 Yılı Budanmış Birliktelik Kuralları.....	84
Şekil 4.8: Apriori Algoritması ile 2014 Yılı Budanmış Birliktelik Kuralları_2.....	84
Şekil 4.9: Apriori Algoritması ile 2015 Yılı, Birliktelik Kuralları_1.....	85
Şekil 4.10: Apriori Algoritması ile 2015 Yılı Birliktelik Kuralları_2.....	85
Şekil 4.11: Apriori Algoritması ile 2015 Yılı Budanmış Kurallar.....	85
Şekil 4.12: Apriori Algoritması ile 2015 Yılı, Birliktelik Kuralları_2.....	86
Şekil 4.13: Apriori Algoritması ile 2015 Yılı Budanmış Kurallar_2.....	86
Şekil 4.14: Apriori Algoritması ile 2015 Yılı Birliktelik Kuralları_3.....	86
Şekil 4.15: Apriori Algoritması ile 2015 Yılı Birliktelik Kuralları_4.....	87
Şekil 4.16: Apriori Algoritması ile 2015 Yılı Birliktelik Kuralları_5.....	87
Şekil 4.17: Apriori Algoritması ile 2015 Yılı Birliktelik Kuralları_6.....	87
Şekil 4.18: Apriori Algoritmasında Destek Serpilme Diyagramları_1.....	88
Şekil 4.19: Apriori Algoritmasında Serpilme Diyagramı_2.....	89
Şekil 4.20: Apriori Algoritmasında Two Key Diyagramı.....	90
Şekil 4.21: Apriori Algoritmasında Graf Diyagramı.....	90
Şekil 4.22: Eclat Algoritmasında Verilerin R Ortamına Tanıtılması.....	91
Şekil 4.23: Veri setinin R Ortamında Tanıtılması Sonrasında Görüntülenmesi.....	92
Şekil 4.24: Eclat Algoritması, 2014 Yılı Birliktelik Kuralları_1.....	92
Şekil 4.25: Eclat Algoritması, 2014 Yılı Birliktelik Kuralları_2.....	93
Şekil 4.26: Eclat Algoritması, 2014 Yılı Birliktelik Kuralları_3.....	93
Şekil 4.27: Eclat Algoritması, 2014 Yılı Serpilme Diyagramı_1.....	93
Şekil 4.28: Eclat Algoritması, Serpilme Diyagramı_2.....	94
Şekil 4.29: Eclat Algoritması 2014 Yılı Graf Diyagramı.....	95
Şekil 4.30: Eclat Algoritması, 2015 Yılı Birliktelik Kuralları_1.....	95
Şekil 4.31: Eclat Algoritması, 2015 Yılı Birliktelik Kuralları_2.....	96

Şekil 4.32: 2015 Yılı Eclat Algoritması Serpilme Diyagramı_1.	97
Şekil 4.33: 2015 Yılı Eclat Algoritması Serpilme Diyagramı_2.	97
Şekil 4.34: 2015 Verileri Eclat Algoritması Destek = 0.002 iken Serpilme Diyagramı.	97
Şekil 4.35: NBMiner Algoritmasından Elde Edilen Birliktelik Kuralları.	98
Şekil 4.36: Eclat Algoritması Kural Yaratılması.	99
Şekil 4.37: Apriori Algoritması Kural Yaratılması.	99
Şekil 4.38: Apriori Eclat Çalışma Zamanları (2014 Yılı).....	99
Şekil 4.39: Apriori Eclat Çalışma Zamanları (2015 Yılı).....	100
Şekil 4.40: Shiny Uygulamasında Sorgulanan Veri Seti.	102
Şekil 4.41: Apriori ile Geliştirilen Shiny Uygulaması Arayüzü.	103
Şekil 4.42: Shiny Uygulamasında Yaratılan Kuralların Görüntülenmesi.	104
Şekil 4.43: Shiny Uygulamasında Yaratılan Kurallara Ait Grafiğin Görüntülenmesi.	105
Şekil 4.44: Shiny Uygulamasında Analiz Edilen Veri Seti.	106
Şekil 4.45: Shiny Apriori Uygulaması Kurallar_1.	106
Şekil 4.46: Shiny Apriori Uygulaması Kurallar_2.	107
Şekil 4.47: Shiny Apriori Uygulaması Kurallar_3.	107
Şekil 4.48: Shiny Apriori Uygulaması Kurallar_4.	108

TABLO LİSTESİ

Sayfa No

Tablo 2.1: Birliktelik Kurallarında Destek ve Güven Ölçülerinin Hesaplanması.	26
Tablo 2.2: Destek Değerinin Hesaplanması Örneği (Lai ve Cerpa, 2001).	28
Tablo 2.3: Güven Değerinin Hesaplanması Örneği (Lai ve Cerpa, 2001).....	29
Tablo 2.4: Destek Güven Hesaplaması.	34
Tablo 2.5: Apriori Algoritmasının Kodları (Han ve Kamber, 2006).	46
Tablo 3.1: 2014 Yılı satış Tablosu Açıklamaları.....	60
Tablo 3.2: 2015 Yılı Satış Tablosu.....	61
Tablo 3.3: 2014-2015 Yılı Müşteri Finansal Durum Tablosu.	62
Tablo 3.4: 2014-2015 Yılları Veri Tabanı Nitelik Karşılaştırma.....	71
Tablo 3.5: 2014-2015 Yılları Cironun Aylık Dağılım Karşılaştırması.	71
Tablo 3.6: 2015 Yılı Bölgelere Göre Ciro Dağılımı.....	73
Tablo 3.7: 2014-2015 Yılı Marka Ciro Dağılımları.....	74
Tablo 4.1: Apriori Eclat Çalışma Zamanları (2014 Yılı).....	99
Tablo 4.2: Apriori Eclat Destek Değerleri ile Çalışma Zamanları (2015 Yılı).....	100

SİMGE VE KISALTMA LİSTESİ

C	: Aday öge küme
C_k	: k elemanlı aday öge küme
L	: Sık Öge küme
L_k	: k elemanlı sık öge küme
X²	: Tüm güven

Kisaltmalar	Açıklama
A ⇒ B	: Birliktelik Kuralı
BFS	: Genişlik Öncelikli Arama (Breadth First Search)
c	: Güven değeri (confidence)
CRISP-DM	: Veri Madenciliği için Çarpaz Endüstri Standard Süreç Modeli (Cross Industry Standart Process for Data Mining)
CRM	: Müşteri İlişkileri Yönetimi (Customer Relationship Management)
DFS	: Derinlik Öncelikli Arama (Depth First Search)
Lhs	: Left Hand Side
Rhs	: Right Hand Side
s	: Destek değeri (support)
T	: İşlemler kümesi
TID	: İşlem numarası
VTBK	: Veri Tabanlarında Bilgi Keşfi
YSA	: Yapay sinir ağları

ÖZET

DOKTORA TEZİ

VERİ MADENCİLİĞİNDE BİRLİKTELİK YÖNTEMLERİ VE MÜŞTERİ İLİŞKİLERİ YÖNETİMİNE İLİŞKİN BİR UYGULAMA

Gökçe KARAHAN ADALI

İstanbul Üniversitesi

Fen Bilimleri Enstitüsü

Enformatik Anabilim Dalı

Danışman : Prof. Dr. M. Erdal BALABAN

Bu tez çalışmasında, birliktelik kurallarına ait algoritmalar kullanılarak, müşterilerin birlikte satın almayı tercih ettikleri ürünler tespit edilerek, müşteri ilişkileri yönetimine ilişkin bir uygulama gerçekleştirilmesi amaçlanmıştır. Çalışmada kullanılan veri seti elektrik sektöründe faaliyet gösteren bir firmadan temin edilmiştir. Veri analizi süresince CRISP-DM modeli takip edilmiştir. Veri 2014-2015 yıllarını içeren veri seti üzerinde birliktelik kuralları algoritmalarından Apriori, Eclat ve NBMiner uygulanarak farklı modeller oluşturulmuştur. Her iki yıla ait sonuçlar ve modellerin performansları karşılaştırılmıştır. Veri analizleri R programlama dili ile gerçekleştirilmiştir. Kodların gerçekleştirilmesinde RStudio geliştirme ortamından yararlanılmıştır. En uygun performansı gösteren Apriori algoritmasından elde edilen model Shiny (shiny.apps.io) aracılığı ile web ortamına taşınmıştır. Kullanıcıya analiz edilen veri setini sorgulama ve algoritma ile ilgili temel düzenlemeleri yapabilme imkânı verilmiştir. Böylelikle uygulamanın zaman ve mekândan bağımsız, dinamik bir hal alması sağlanmıştır.

Şubat 2017, 145. sayfa.

Anahtar kelimeler: Veri Madenciliği, Birliktelik Kuralları, Market Sepet Analizi, Apriori, Müşteri İlişkileri Yönetimi

SUMMARY

Ph.D. THESIS

ASSOCIATION METHODS IN DATA MINING AND AN APPLICATION RELATED TO CUSTOMER RELATIONSHIP MANAGEMENT

Gökçe KARAHAN ADALI

İstanbul University

Institute of Graduate Studies in Science and Engineering

Department of Informatics

Supervisor : Prof. Dr. M. Erdal BALABAN

In this thesis, it is aimed to determine the products that customers prefer to buy together by using algorithms of association rules and to implement an application related to customer relationship management. The data set used in this study was obtained from a company operating in the electricity sector. CRISP-DM model was used during data analysis. Different models were created by applying association rules techniques such as Apriori, Eclat and NBMiner containing the years 2014-2015. The performance of both models and the results of the two years were compared. Data analysis is performed with R language. RStudio was used as a development tool for R codes. The model performed with Apriori which has the most proper performance was transferred to web environment via Shiny (shinyapps.io). The user is given the opportunity to query the analyzed data set and make basic arrangements related to the algorithm. This allows the application to be dynamic, independent of time and space.

February 2017, 145 pages.

Keywords: Data Mining, Association Rules, Market Basket Analysis, Apriori, Customer Relationship Management.

1. GİRİŞ

Günümüzde işletmelerin ayakta kalabilmelerini sağlayan en büyük sermayeleri müşterileridir. Kurumun yaratmış olduğu entelektüel sermayenin yanı sıra, var olma sebebi olan kâr elde etmesi, bulundurduğu mevcut müşteri potansiyeline ve elde etmeye çalıştığı yeni müşterilerine endekslidir. Artan rekabet ortamında, bir adım önde olabilmek için mevcut müşterileri tutundurmak oldukça zor hale gelmektedir. Bu nedenle işletmeler, ürüne göre müşteri arayışından çıkarak, müşteriye göre ürün yaratma gerekliliğinin bilincinde bulunmaktadırlar. Bu nedenle geliştirilmiş olan müşteri ilişkileri yönetimi akıllı alman boyutlara varmış olup, müşterinin satın alma davranışlarını belirlemek, bir sonraki alışverişinde satın alabileceği ürün seçeneklerini kendisine sunmak, ilgisini çekebilecek alternatif ürünler sunmak gibi müşteri için bir sonraki adımı belirler hale gelmiştir.

Bu süreçte yığınlar halindeki veriden saklı örüntüleri çıkararak, anlamlı hale getiren veri madenciliği modellerinden faydalanılmaktadır. Verinin olduğu her alanda uygulanabilecek olan veri madenciliği ile işletmeler, kendi veri kaynaklarından beslenerek elde ettikleri ilişki kurulabilir, anlamlı veri ile stratejik kararlar almaktadırlar. Müşteri İlişkileri Yönetiminde kullanılan Veri Madenciliği uygulamalarında amaç, müşteri davranışlarını tanımlayarak, ileriye yönelik tahminler yapabilmek için modeller kurmaktır.

Bu çalışmada satın alma eğilimlerinin tanımlanmasını sağlamak için kullanılan veri madenciliği yöntemlerinden birliktelik kurallarının; ticaret, finans, mühendislik, fen ve sağlık sektörlerinin birçok alanlarında uygulama sahası bulunmaktadır (Birant ve diğ., 2010). İşletmelerde satış, çapraz satış programları, katalog ve yerleşim düzeni tasarımları, promosyon analizlerinde, metin araması işlemleri ve web sayfalarında ziyaretçilerin hangi sayfaları birlikte tıkladığı ile ilgili analizlerde kullanılabilmektedir (Erpolat, 2012). Literatür incelendiğinde, eğitimde öğrenci başarılarının analiz edilmesi, tıp dünyasında hastalık belirtilerinin tespitinde (Delen ve diğ., 2005), trafik kazalarının hangi koşullarda gerçekleştiğinin tespiti gibi farklı alanlarda uygulandığına dair örnekler görülebilmektedir.

Bu tez çalışmasında, elektrik sektöründe faaliyet gösteren bir firmanın 2014-2015 yıllarını kapsayan satış verileri üzerinden, müşteri ilişkileri yönetimi kapsamında, firmanın müşterilerini daha iyi tanıması, satın alma davranışlarını belirlemesi, müşterilerini segmentlere ayırarak bu doğrultuda kampanyalar, pazarlama ve satış stratejileri geliştirebilmesi amacı ile yol gösterici bir uygulama gerçekleştirilmesi amaçlanmıştır. Firmanın kendi değerlendirmesini yapabilmesi adına, her iki yıla ait veriler ayrı ayrı analiz edilmiş olup, sonuçlar karşılaştırmalı olarak sunulmuştur. Bu kapsamda veri madenciliği çatısı altında farklı birliktelik kuralları yöntemlerinden faydalanılarak modeller geliştirilmiş ve bu modellerin performansları karşılaştırılmıştır.

Uygulamanın bir sonraki aşamasında, seçilen birliktelik kuralı modeli web ortamına taşınarak, herkes tarafından erişilebilir ve uygulanabilir hale getirilmiştir. Geliştirilen model sayesinde, belirlenen zamanda, olası ürün paketlerinin kullanıcı tercihleri doğrultusunda, bir müşteri veya müşteri grupları üzerinde dinamik olarak uygulanabilirliği sağlanmıştır. Çalışmanın bu yönü ile benzer alanda yapılacak çalışmalara temel oluşturması amaçlanmıştır.

Tezin GENEL KISIMLAR bölümünde, çalışmanın kavramsal çerçevesini oluşturmak adına, Veri Madenciliği, Veri Madenciliği Süreçleri, Veri Madenciliği Modelleri, Birliktelik Kuralları, Market Sepet Analizi, Birliktelik Kurallarının Belirlenmesinde Kullanılan Modeller, Müşteri İlişkileri Yönetimi ve Veri Madenciliği ile ilgili literatür araştırmalarına yer verilmiştir.

Veri Madenciliği Süreçleri kısmında, problemin tanımlanmasından, modelin uygulamaya geçirilmesine kadar, Veri Madenciliği için Çapraz Endüstri Standard Süreç Modeli (CRoss-Industry Standard Process for Data Mining - CRISP) adımları takip edilmiştir. Bu süreçte veri madenciliğinde ele alınan Problemin Tanımlanması, Veriyi Anlama, Verilerin Hazırlanması, Modelin Kurulması ve Değerlendirilmesi, Modelin Kullanılması, Modelin İzlenmesi temel bilgiler ışığında açıklanmıştır. Tez çalışmasının modelleme kısmında başvurulan algoritmalara, Bölüm 2.6. Birliktelik Kurallarının Belirlenmesinde Kullanılan Modeller bölümünde yer verilmiştir.

MALZEME ve YÖNTEM bölümünde, birliktelik kuralları ile geliştirilen müşteri ilişkileri yönetimine ilişkin uygulama, CRISP-DM modeli süreç adımları çerçevesinde açıklanmıştır. Çalışmada Türkiye’de elektrik sektöründe faaliyet gösteren bir firmanın iki yılı içeren satış verileri veri madenciliği sürecinde analiz edilmiştir. Bu bölümde, veri setinin toplanma sürecinden, veriyi daha iyi tanımlamak ve analize hazır hale getirilmesi için geçirilen veri ön işleme sürecinden bahsedilmiştir.

BULGULAR bölümünde, veri analizine ve analizler sonucunda elde edilen sonuçlara ve bunların yorumlanmasına yer verilmiştir. Aynı bölümde yazılan web arayüzü ile geliştirilen modelin web ortamına taşınarak, mekan ve zamandan bağımsız bir şekilde dinamik bir hal alması sağlanmıştır.

TARTIŞMA ve SONUÇ bölümünde, genel bir değerlendirme yapılarak, elde edilen bulgular yorumlanmış, gelecek çalışmalara yön verebilecek önerilerde bulunulmuştur.

2. GENEL KISIMLAR

2.1. VERİ MADENCİLİĞİ

Veri madenciliği, büyük miktarda verinin geniş ölçüde ulaşılabilir olması ve bu verilerin faydalı bilgiye dönüştürülmesi gereği nedeniyle son yıllarda bilgi endüstrisi ve bilgi toplumunda büyük ilgi görmektedir (Han ve Kamber, 2006). Veritabanlarında bilgi keşfi, depolanmış veri içerisindeki geçerli, yeni, faydalı ve sonuç olarak anlaşılabilir örüntülerin (patterns) çıkarılması sürecidir. Özkan (2008) veri madenciliğini büyük ölçekli veriler arasından “değeri olan” bir bilgiyi elde etme işi olarak tanımlamaktadır. Parr Rud (2001), veri madenciliğini çeşitli endüstrilerde kullanılan ve çeşitli teknikleri kapsayan bir terim olarak açıklamaktadır. Veri madenciliği büyük miktardaki veriden, gelişmiş modelleme teknikleri kullanarak saklı örüntüleri ve bilgiyi çıkarmayı amaçlamakta, veriyi uygulanabilir bilgiye dönüştürmektedir (Tsipstsis ve Chorianopoulos, 2009).

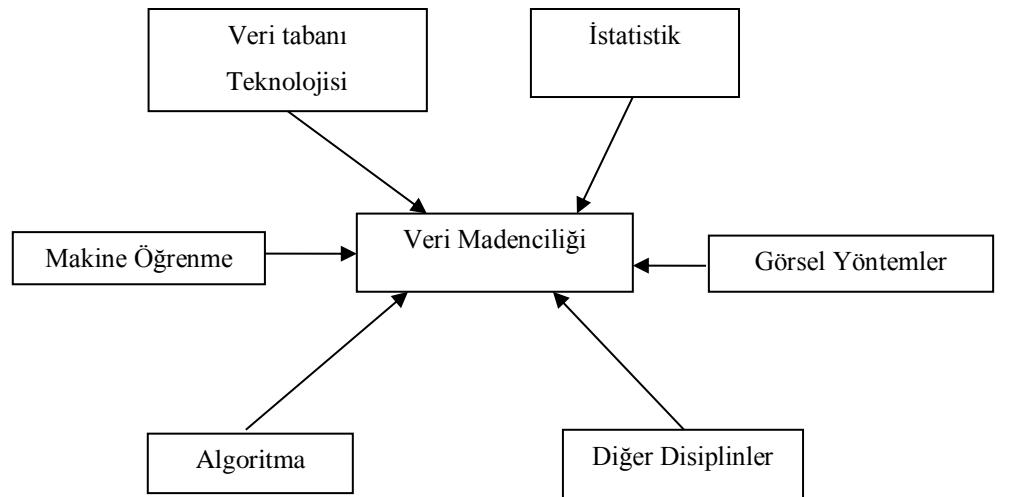
Fayad ve diğ. (1996), veri madenciliğini; veriden saklı örüntüleri çıkartmak için bir takım özel algoritmaların uygulanması olarak tanımlamaktadır. Bu sayede veriler arasındaki ilişkileri ortaya koymak ve gerektiğinde ileriye yönelik kestirimlerde bulunmak mümkün görülmektedir. Artan rekabet ve pazarlama arenasındaki pazar payı nedeniyle, veri madenciliği, müşteri yaşam döngüsünün her aşamasında rekabet avantajı sağlamak için önemli bir uygulama haline gelmiştir.

Analiz edilecek veriler iyi düzenlenmiş ya da özelleşmiş veri ambarlarından veya yapısal olmayan veri kaynaklarından elde edilebilmektedir. Elde edilen bilgi, piyasa analizi, dolandırıcılık tespiti, müşteri sadakati ve üretim kontrolünden, bilim araştırmalarına kadar uzanan çok çeşitli uygulamalar için kullanılabilir. Bir başka tanıma göre; veri madenciliği, veri ambarlarında saklanan çok miktarda ve farklı türlerdeki veriye dayanarak daha önce keşfedilmemiş bilgileri ortaya çıkarmak, bunları karar verme ve eylem planını gerçekleştirmek için kullanma sürecidir (Swift, 2000).

Veri madenciliği, büyük miktarda veri içerisinde, gelecekle ilgili tahmin yapmamızı sağlayacak bağlantı ve kuralların ortaya çıkarılmasını, daha önceden fark edilmemiş veri örüntülerini tespit edebilmemizi sağlamaktadır. Veri madenciliğinin amacı, geleceğe yönelik tahminlerde bulunarak bilgiyi gelecekteki müşteri ilişkilerini yönetmek amacıyla değerlendirmektir.

Veri madenciliği geniş miktarlardaki veriye ulaşılabilirliği ve verinin kullanılabilir bilgiye dönüşümünün gerekliliği sayesinde, bilgi endüstrisi ve bilgi toplumlarında son yıllarda oldukça fazla dikkat çekmiştir. Buradan elde edilen anlamlı bilgi piyasa analizlerinden, dolandırıcılık tespitine, müşteri tutundurmada, üretim kontrolü ve bilimsel araştırmalara kadar geniş bir yelpazedeki uygulamalar için kullanılabilir (Han, 2006). Hand (1998), veri madenciliğini istatistik, veritabanı teknolojisi, örüntü tanıma, makine öğrenme ile etkileşimli yeni bir disiplin ve geniş veritabanlarında önceden tahmin edilemeyen ilişkilerin ikincil analizi olarak tanımlamıştır.

Veri madenciliği yöntemleri genel olarak istatistik, yapay zeka ve yapay zekanın uzantısı makine öğrenimi olmak üzere iki ana kökten beslenmektedir. Bu iki disiplinin dışında, Şekil 2.1’de görüldüğü üzere, veritabanı teknolojisi, görsel yöntemler, algoritmalar da veri madenciliği yöntemlerinde önemli rol oynamaktadır (Akpınar, 2014).



Şekil 2.1: Veri Madenciliği Disiplinlerarası İlişkisi (Akpınar, 2014).

İstatistiğin genel olarak tanımlayıcı ve yorumlayıcı oluşu, veri madenciliğinde kümeleme, ilişki kurma, tahmin yürütme ve karşılaştırma amaçları ile kullanılmaktadır. Bu sayede birçok model çıkarılmaktadır. İstatistikte daha sıklıkla birincil veri analizine

rastlanılmaktadır. Veriler belirli bir soru veya soru seti ile toplanılmaktadır. Veri madenciliğinde ise birincil veri analizi ile ilgilenilmemektedir (Hand, 1998). İstatistiğin yanında veri tabanlarının ve bilgi öğrenme metotlarının, yeni algoritmaların geliştirilmesi ile birlikte veri madenciliği, birçok alanın kesişmesinin bir ürünü olarak ortaya çıkmaktadır.

Veri madenciliği klasik istatistiksel uygulamalara çok benzemektedir. Ancak klasik istatistiksel uygulamalar yeterince düzenlenmiş ve çoğunlukla özet veriler üzerinde çalıştırılmaktadır. Ayrıntılı bilgi olsa bile burada kayıtlar binlerce olabilmektedir. Veri madenciliği ise milyonlarca ve hatta milyarlarca veri ve çok daha fazla değişken ile ilgilenmektedir. Veri sayısı çok olunca, bazı özel analiz algoritmalarının geliştirilmesi gerekmiş, ayrıca verinin saklandığı ortamların da yeniden düzenlenmesini gerekli kılmıştır (Özkan, 2008).

Zhang ve Zhang (2002) veri madenciliğinin, geleneksel istatistikten ayrıldığını şu şekilde ifade etmiştir.

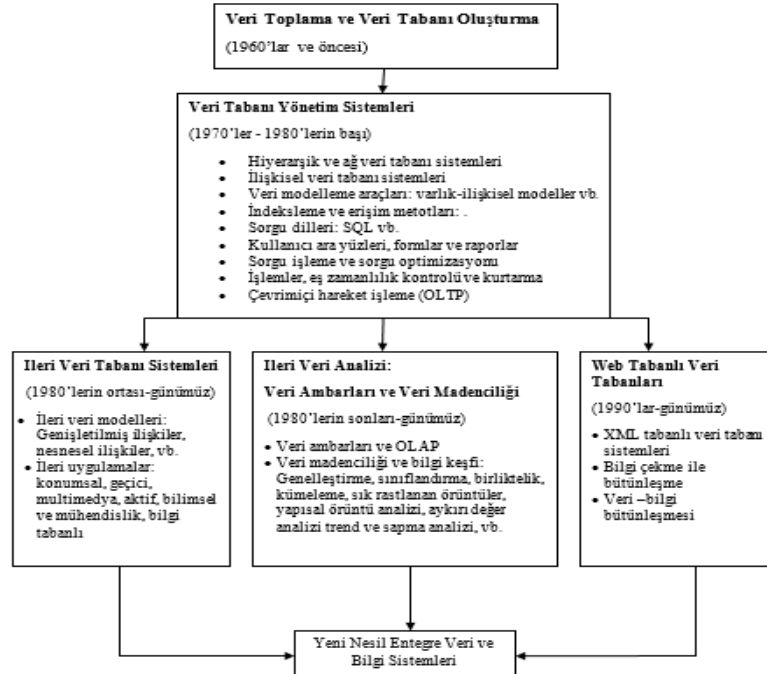
İstatistik varsayım odaklıdır, hipotez kurulur ve verilere dayalı olarak geçerliliği doğrulanırken, veri madenciliği, istatistiğin aksine tahmin odaklıdır. Örüntüler ve hipotezler, verilerden doğrudan çıkartılır. Bir başka deyişle, veri madenciliği veri odaklı iken, istatistik insan odaklıdır.

Çağiltay (2011), veri madenciliğini farklı veri kaynaklarından elde edilen verilerin, farklı görüş açılarıyla özetlenmesi ve analizi sonucunda, istenilen yararlı bilgi haline getirilmesi işlemi olarak tanımlamıştır. Veri madenciliği yöntemleri verilerin farklı boyutları kullanarak analiz edilmesi, kategorize edilmesi, özetlenmesi ve bağıntıların belirlenmesi amacıyla kullanılan yöntemlerdir.

Balaban ve Kartal (2015)'a göre bilgi ve iletişim teknolojilerinde yaşanan gelişmeler, özellikle elektronik ortamda depolanan veri miktarındaki önemli artış ve bu veriyi yorumlamanın insan kapasitesi açısından zorluğu, araştırmacıları veri madenciliği alanındaki çalışmalara yöneltmiştir. Fazla miktardaki veri arasından işe yarar/stratejik bilginin elde edilmesi bu sayede mümkün olmaktadır.

Veri madenciliği, bilgi teknolojileri evriminin doğal bir sonucu olarak görülebilmektedir. Veritabanı sistemleri endüstrisi, veri toplama ve veritabanı oluşturma, veri yönetimi (veri depolama, veri kurtarma, veritabanını işlem süreçleri) ve ileri veri analizi (veri ambarı, veri madenciliği.) işlevlerinin geliştirilmesinde evrimsel bir yola tanıklık etmiştir (Han, 2006).

Geçmişten günümüze veriler her zaman yorumlanmış, bilgi elde etmek istenmiştir ve bunun için donanımlar oluşturulmuştur. Bu sayede bilgi, geçmişten günümüze taşınır hale gelmiştir. 1950'lerde ilk bilgisayarların sayım amaçlı kullanılması ile başlayan süreç, 1970'lerde ilişkisel veri modeli, ilişkisel veritabanı yönetim sistemleri uygulamaları ile devam etmiştir. Bu gelişmeleri 1980'lerde ilişkisel veritabanı yönetim sistemlerinin yaygınlaşması, 1990' larda büyük miktardaki veriden anlamlı bilgiyi elde etme yollarının aranmaya başlanması takip etmiştir. 1992 yılında veri madenciliği konusunda ilk yazılım gerçekleştirilmiştir. 2000'li yıllara gelindiğinde veri ambarları, veri madenciliği yaygınlaşmış ve hemen hemen her alanda uygulanmaya başlanmıştır (Savaş ve diğ., 2012). Han (2006), veri madenciliğinin tarihsel gelişim sürecini aşağıdaki şekilde (Şekil 2.2.) olduğu gibi göstermiştir:



Şekil 2.2: Veritabanı Sistemlerinin Tarihsel Gelişimi

2.2. VERİ MADENCİLİĞİ SÜREÇLERİ

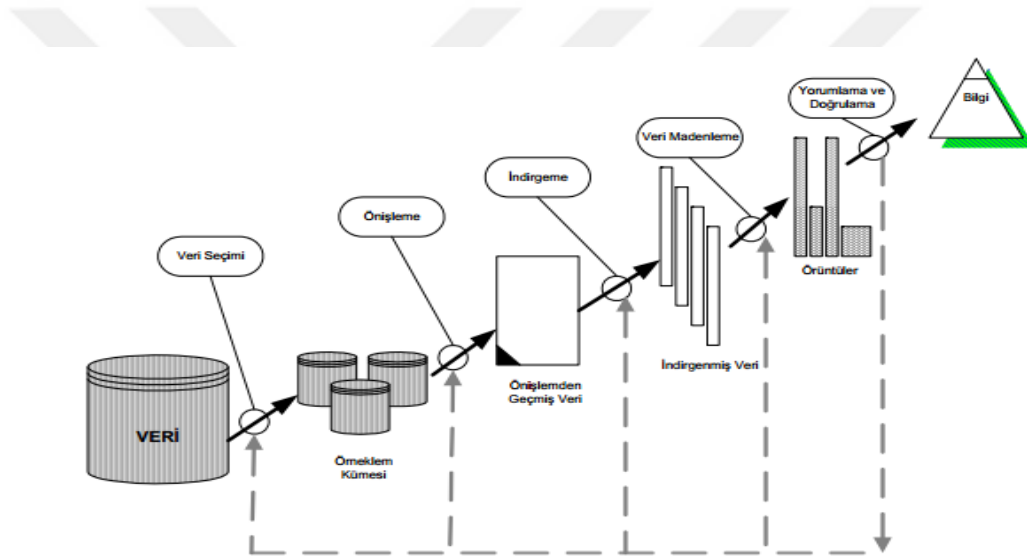
Bilgi çağında veriye, veriden anlamlı sonuçlar üretmeye ve veritabanlarına yapılan yatırım her geçen gün artmaktadır. Veriye erişimin git gide kolaylaşması ve veri miktarındaki bolluk, veri madenciliği ve bilginin keşfini önemli bir gereklilik haline getirmiştir. İşletmeler veritabanlarında her geçen gün daha fazla veri depolama ihtiyacı duymaktadır (Özçakır ve Çamurcu,2007). Ancak çok büyük boyutlara ulaşan veriyi işlemek ve bunlardan anlamlı sonuçlar çıkartmak hiç kolay olmamaktadır. Toplanan veri miktarı büyüdükçe ve toplanan verilerdeki karmaşıklık arttıkça, daha iyi çözümlene tekniklerine olan gereksinim artmaktadır (Şimşek, 2006). Bu büyük boyuttaki verilerden yararlanmak için, bu veriler üzerinde yöntem ve kurallar uygulanarak değerli bilginin keşfedilmesine gerek duyulmuştur. Bu sürece, Veri Tabanlarında Bilgi Keşfi VTBK (Knowledge Discovery in Databases – KDD) adı verilmektedir (Özçakır ve Çamurcu, 2007).

Veriden anlamlı bilgilerin çıkarılması bilgi keşfi olarak adlandırılırken, veri madenciliği, VTBK sürecinde, kabul edilebilir hesaplama verimliliği sınırlamaları altında veri analizlerinin ve algoritmaların uygulanarak veri üzerindeki örüntülerin ya da modellerin ortaya çıkarıldığı bir adımdır (Fayyad ve diğ., 1996). VTBK süreci, hem kuruluş içindeki hem de kuruluş dışındaki veritabanlarından toplanan verileri kullanmaktadır. Bu veriler, ilişkileri keşfetmek için farklı açılardan ve perspektiflerden analiz edilmektedir (Gilchrist ve diğ., 2012). Eğer veri tabanlarında bilgi keşfi süreci başarılı ise keşfedilen bilgi, organizasyonların karar verme sürecinin geliştirmesi amacıyla kullanılabilir (Kumar ve diğ., 2011).

Veritabanlarında bilgi keşfi, verinin gelişimi ve bilgi tabanlı sistemler için önemli bir konudur. Bilgi keşfi için veri tabanlarında, nitelik odaklı girişim metodları (*attribute oriented induction method*) geliştirilmiştir. Bu metotlar özellikle örneklerden öğrenme teknikleri içeren makine öğrenme paradigması ile küme yönelimli (set oriented) veritabanı operasyonlarını bütünleştirerek, gerçek veriden genelleşmiş veriyi çıkartmaktadır.

Günümüz gerçek veri tabanları, büyük boyutları (genellikle birkaç gigabyte veya daha fazla) sebebiyle gürültüye, eksik ve tutarsız verilere, muhtemel çoklu heterojen

kaynaklara karşı son derece duyarlıdır. Düşük kaliteli veri, düşük kalitelideki analiz sonuçlarına sebebiyet verecektir. ”Verinin kalitesini ve dolayısıyla veri madenciliği sonuçlarını iyileştirmeye yönelik veri nasıl bir ön işlemeden geçirilebilir ?” sorusuna cevap aranmaktadır. Bu noktada, bir takım veri ön işleme teknikleri bulunmaktadır (Han ve Kamber, 2006). Fayyad, Piatetsky-Shapiro ve Smith tarafından 1996 yılında yayımlanan *From Data Mining to Knowledge Discovery in Databases* isimli makalede okuyucuya sunulan veri madenciliği süreci Şekil 2.3’de görülmektedir (Fayyad ve diğ., 1996). Bilgi keşfi süreci iteratif ve interaktif bir süreçtir. İteratif olması her adımda bir önceki adıma dönmeniz gerekebilmekte olduğu anlamına gelmektedir (Maimon ve Rokach, 2005).



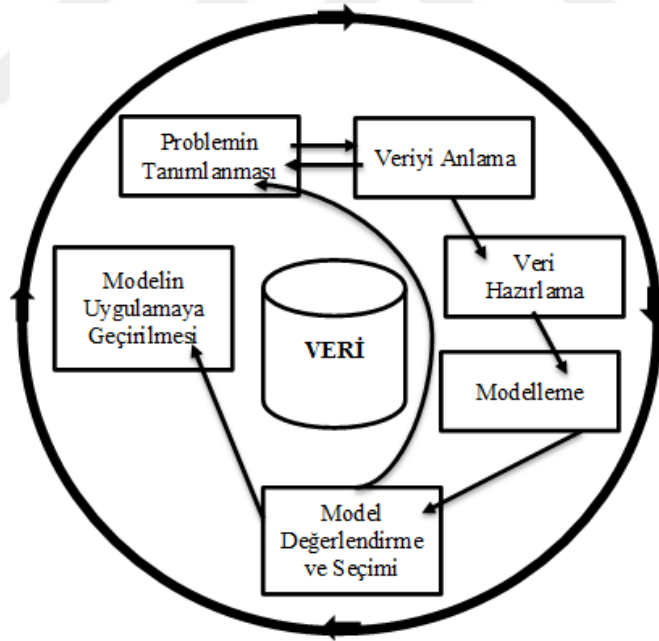
Şekil 2.3: Veri Tabanlarında Bilgi Keşfi Süreci (Fayyad ve diğ., 1996).

Veri madenciliği, işletme problemlerinin veri madenciliği görevlerine dönüştürülmesine yardımcı olacak, uygun veri dönüşümleri ve veri madenciliği tekniklerini sunacak ve sonuçların etkinliğini değerlendirerek bu deneyimi raporlayacak standart bir yaklaşıma ihtiyaç duymaktadır (Wirth, 2000). Endüstri liderlerinin ve iki yüzden fazla uzmanın, veri madenciliği aracı kullanıcıları ve hizmeti sağlayıcılarının katkılarıyla Veri Madenciliği için Çarpraz Endüstri Standard Süreç Modeli (*CRoss-Industry Standard Process for Data Mining - CRISP*) geliştirilmiştir (Shearer, 2000). CRISP-DM projesi, endüstri sektöründen ve kullanılan teknolojiden bağımsız veri madenciliği projelerinin yürütülmesine yönelik bir çerçeve sağlayan bir süreç modeli tanımlayarak belirtilen

sorunların bir bölümünü ele almıştır (Wirth, 2000). Metodolojinin ilk sürümü Brüksel’de 4.CRISP-DM *Workshop*’ında 1999 yılı Mart ayında sunulmuştur.

Veri madenciliği ve veritabanlarında bilgi keşfi sürecinde; veriden bilgiye giden yolda geliştirilen çeşitli süreçler bulunmaktadır. CRISP-DM veri madenciliği sürecinde takip edilmesi önerilen veri madenciliği süreç modelidir. CRISP-DM, veri madenciliği sürecini altı ana aşamaya ayırmaktadır. Aşamaların sırası kesin değildir ve farklı aşamalar arasında ileri ve geri geçişler söz konusu olabilmektedir. CRISP-DM süreç modeli, büyük veri madenciliği projelerini, daha az maliyetli, daha güvenilir, daha tekrarlanabilir, daha yönetilebilir ve daha hızlı hale getirmeyi amaçlamaktadır.

CRISP-DM süreç modeli, Problemin Tanımlanması, Veriyi Anlama, Veri Hazırlama, Modelleme, Modelin Değerlendirme ve Seçimi ve Modelin Uygulamaya Geçirilmesi olmak üzere altı aşamadan oluşmaktadır (Şekil 2.4). Bu tez çalışmasında CRISP-DM süreç modeli adımları izlenmiştir.



Şekil 2.4: CRISP-DM Süreç Diyagramı (Shearer, 2000).

2.2.1. Problemin Tanımlanması

Başlangıç aşamasında işletmenin bakış açısına uygun olarak projenin amaç ve gereksinimlerinin anlaşılmasına odaklanılmaktadır. Ardından amaç ve gereksinimler bir

veri madenciliği problemi olarak tanımlanmakta ve amaçlara erişebilmek üzere öncü bir plan hazırlanmaktadır (Tsiptsis ve Chorianopoulos, 2009).

Veri madenciliği çalışmalarında başarılı olmanın ilk şartı, uygulamanın amacının açık bir şekilde tanımlanmasıdır. Amaç, problem odaklı olup, açık ve net bir dille ifade edilmiş olmalı, elde edilecek sonuçların başarı düzeylerinin nasıl ölçüleceği tanımlanmalıdır (Döşlü, 2003). Ayrıca, yanlış tahminlerde katlanılacak olan maliyetlere ve doğru tahminlerde kazanılacak faydalara ilişkin tahminlere de bu aşamada yer verilmelidir (Ayık ve diğ., 2007).

2.2.2. Veriyi Anlama

Bu aşama veri kalitesine ilişkin problemleri tanıyabilmek, veri hakkında ilk izlenimleri edinebilmek, ilginç alt dizileri belirleyebilmek veya veri dizilerinde gizli bulunan enformasyonun çıkartılması için hipotezlerin geliştirilmesi gibi çeşitli amaçları gerçekleştirecek faaliyetleri içermektedir (Tsiptsis ve Chorianopoulos, 2009).

Modelin kurulmasından önce mevcut verinin iyi anlaşılması gerekmektedir. Veri seti üzerinde analizlere başlamadan önce veri hakkında ön fikir edinilmesi için bazı basit tanımlayıcı istatistiki hesaplamalar yapılabilen ve bunların üzerinden grafikler elde edilebilmektedir. Tüm bu işlemler, veri ön-işleme sürecinde hangi analizlerin gerçekleştirilmesi gerektiği hakkında başvurabileceğimiz ön bilgiyi sunmaktadır (Kartal, 2015).

2.2.3. Verilerin Hazırlanması

Veri hazırlama aşaması ham veriden başlayarak nihai veri dizisine erişinceye kadar gerekli tüm faaliyetleri kapsamaktadır. Veri hazırlama işlemleri önceden belirlenmeyen, muhtemelen birçok defa icra edilebilir. İlgili işlemler tablo, kayıt, öznitelik seçimi ve modelleme araçları için verinin temizlenmesi ve dönüştürülmesini içermektedir (Tsiptsis ve Chorianopoulos, 2009).

Veri hazırlamanın amacı veri madenciliği algoritması için girdi olabilecek veri kümesini oluşturabilmektir. Modelin kurulması sırasında yaşanabilecek sorunlar, bu aşamaya tekrar tekrar dönülüp, verilerin hazırlanma aşamasının yeniden gözden geçirilmesini gerektirebilmektedir. Bu nedenle bu aşama içerisindeki adımların özenle ve titizlikle

yerine getirilmesi gerekmektedir. Seçilen verilerin minimum alt kümesinin değerlendirilmesi gerekmekte, ilgili nitelikler ve uygun zaman periyodu göz önünde bulundurulmalıdır (Raju, ve diğ., 2014). Veri madenciliği öncesinde verinin hazırlanması süreci olarak da kabul edilen bu işlemler özellikle veritabanındaki bozuk değerleri ve veriler arasındaki tutarsızlıkları kaldırmayı amaçlamaktadır (Dondurmacı ve Çınar, 2014). Verilerin hazırlanması aşaması kendi içerisinde toplama, değer biçme, birleştirme ve temizleme, seçme ve dönüştürme adımlarından meydana gelmektedir.

2.2.3.1. Veri Toplama (Collection)

Toplama, tanımlanan problem için gerekli olduğu düşünülen verilerin ve bu verilerin edinileceği veri kaynaklarının tespit edilmesidir. Verilerin toplanmasında kuruluşun kendi veri kaynaklarının dışında, dış kaynaklardan da faydalanılabilmektedir (Ayık ve diğ., 2007).

2.2.3.2. Veriye Değer Biçme (Assessment)

Analiz yapılacak verilerin farklı veritabanları ve çeşitli kaynaklardan toplanarak, bir araya getirilmesi veri bütünlüğünü bozacak bir takım uyumsuzluklara neden olabilmektedir. Bu uyumsuzlukların bazıları farklı zamanlarda kaydedilmiş olmaları, farklı kullanıcılar tarafından kaydedilmesi nedeni ile ortaya çıkabilecek kodlama farklılıkları ve bunlar gibi bir takım sıkıntıları beraberinde getirebilmektedir. Bu nedenlerle, değerlendirmeye alınabilecek nitelikte modeller ancak anlamlı verilerin üzerinde yaratılabileceği için, toplanan verilerin ne ölçüde uyumlu oldukları bu adımda incelenerek değerlendirilmelidir.

2.2.3.3. Veri Birleştirme ve Temizleme

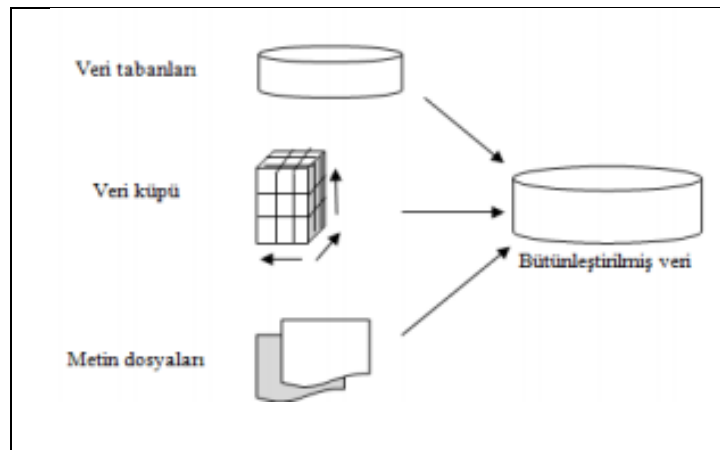
Veritabanında yer alan tutarsız ve hatalı verilere gürültülü veri denilmektedir. Bazı uygulamalarda, üzerinde çözümleme yapılacak verilerin istenilen özelliklere sahip olmadığı görülebilir. Örneğin eksik verilerle ve uygun olmayan verilerin oluşturduğu tutarsız verilerle karşılaşılabilir (Han, 2006). Veri temizleme aşaması, kayıp ya da eksik verileri doğru şekilde tamamlamak, uç değerleri belirleyerek gürültüyü azaltmak gibi kullanılan birçok tekniği kapsamaktadır. Eksik verinin analiz sonucunun kesinliğini azaltması, kayıp değerler nedeniyle meydana gelecek boşlukların hesaplamada karmaşıklık yaratması gibi olumsuz etkileri olabilmektedir (Suthar ve diğ., 2012; Somasundaram ve Nedunchezian, 2011).

Verilerdeki gürültüyü temizlemek için;

- Eksik değer içeren kayıtlar atılabilir,
- Kayıp değerlerin yerine sabit bir değer atanabilir,
- Diğer verilerin ortalaması hesaplanarak kayıp veriler yerine bu değer yazılabilir,
- Değişkenin tüm verileri kullanılarak ortalaması hesaplanır ve eksik değer yerine bu değer kullanılabilir,
- Değişkenin tüm verileri yerine, sadece bir sınıfa ait örneklerin değişken ortalaması hesaplanarak eksik değer yerine kullanılabilir,
- Verilere uygun tahmin yapılarak eksik değer tahmin edilebilir ve eksik değer yerine kullanılabilir (Özkan, 2013).

Verilerin saklandığı veri ambarlarının oluşturulması sırasında farklı kaynaklardan elde edilen veriler arasında bütünlüğü bozmamak adına verilerin aynı çatı altında bütünleştirilmesi işlemine başvurulmaktadır.

Bu süreçte, verinin bütünleştirilmesi için mevcut bütün veri kaynakları karakteristik özelliklerine ve toplanma seviyelerine göre eşleştirilerek, tutarlılık sağlanmaktadır. Bu aşamaya kurumun sahip olduğu birden fazla bilgi kaynağında kayıtlı olan verilerin bir araya getirilmesi aşamasında, aynı verinin farklı biçimlerde tutulması söz konusu olduğunda veri bütünleştirilmesi işlemine başvurulmaktadır (Dondurmacı ve Çınar, 2014).



Şekil 2.5: Veri Bütünleştirme (Dondurmacı ve Çınar, 2014).

Veri kapasitelerindeki artış, veri ambarlarının boyutlarının artmasına, saklanan verilerin erişiminin artırılmasına neden olmaktadır. Veriye farklı kullanıcılar tarafından sağlanan

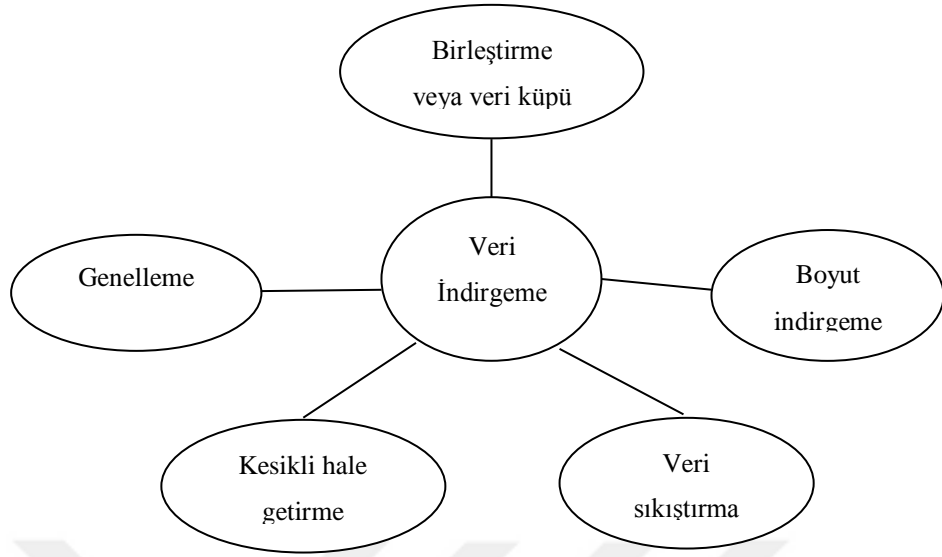
erişim sürecini arttırsa da, bir takım problemleri beraberinde getirebilmektedir (Namey ve diğ., 2007). Örneğin iki farklı kullanıcı tarafından kullanılan bir veritabanında, “MüşteriNo” niteliği Müşteri_NO ve MüşteriNo gibi farklı iki şekilde kayıt altına alınmış olabilmektedir. Bu yanlışlık tüm süreçlerimize yansıtacak bir takım hataları ardında getirecektir. Bu ve bunun gibi bir takım yanlışların önlenmesi adına veriler standart hale getirilmektedir.

2.2.3.4. Veri İndirgeme

Günümüzde özellikle bellek kapasitelerinin artması, veriye erişimin çok daha kolay hale gelmesi sonucunda, veri fazlalığı problemi yaşanmaktadır. Çok büyük ölçeklerde gerekli/gereksiz saklanan bu veri yığınları, veri analizi çalışmalarının verimli olarak yürütülememesine neden olmaktadır. Eğer veritabanları bu veri kirliliğinden arınmadan çalışmalara dahil edilirse, elde edilecek sonuçların doğruluğundan şüphe etmek gerekmektedir (Kim ve diğ., 2003). Bu sorunları giderebilmek için çeşitli veri indirgeme yöntemleri geliştirmiştir (Akpınar, 2014). Veri indirgeme teknikleri, daha küçük hacimli olarak ve veri kümesinin indirgenmemiş bir örneğinin elde edilmesi amacıyla uygulanmaktadır.

Böylelikle elde edilen indirgenmiş veri kümesine veri madenciliği teknikleri uygulanarak daha etkin sonuçlar elde edilebilir. Veri indirgeme yöntemleri aşağıdaki biçimde özetlenebilmektedir:

- Veri birleştirme veya veri küpü (Data Aggregation or Data Cube)
- Boyut indirgeme (Dimension Reduction)
- Veri sıkıştırma (Data Compression)
- Kesikli hale getirme (Discretization)



Şekil 2.6: Veri İndirgeme Yöntemleri (Witen ve Frank, 2005).

Veri madenciliği yapılacak veri kümesi bazen gereksiz değişkenler içerebilmektedir. Gereksiz değişkenlerin veri setinden çıkarılması amacıyla ileri veya geri yönlü olarak sezgisel seçimler yapılabilmektedir. Veri sıkıştırmada orijinal verileri temsil edebilecek indirgenmiş veya sıkıştırmış veriler, veri şifreleme veya dönüşümü ile elde edilmektedir. Kesikleştirmede, bazı veri madenciliği algoritmaları yalnızca kategorik değerleri ele aldığından, sürekli verilerin kesikli değerlere dönüştürülmesini içermektedir (Oğuzlar, 2003).

2.2.3.5 Veri Dönüştürme (Transformation)

Veri dönüştürme, daha sağlıklı sonuçların elde edilebilmesi veya verinin kullanılan algoritmalarla uyumlu olabilmesi için, verinin tanımlanan bir fonksiyona uygun olarak farklı değer veya ölçeklere dönüştürülmesi işlemidir (Akpınar, 2014). Kısaca, verinin kullanılacak modele göre içeriğini koruyarak veri madenciliği için uygun formlara dönüştürülmesi işlemidir de denilebilir.

Dönüştürme süreci şemalar, dönüşüm eşlemeleri, iş akışı tanımları vb. gibi büyük miktarda meta veri gerektirmektedir. Tutarlılık, esneklik ve tekrar kullanım kolaylığı için bu meta veri, VTYS (Veritabanı Yönetim Sistemleri) tabanlı bir depoda tutulmalıdır. Veri kalitesini desteklemek için, dönüşüm süreciyle ilgili ayrıntılı bilgi hem depoda hem de dönüştürülen örneklerde kayıt altına alınmalıdır (Lomet, 2001).

Veri dönüştürmede sıklıkla kullanılan yöntem normalizasyondur. Veri dizisinde bulunan değerlerin $[-1, +1]$ veya $[0, +1]$ aralığında yer alacak şekilde dönüştürülmesidir. Farklı büyüklükte verinin bulunduğu dizilerde sıklıkla kullanılan dönüştürme işlemidir. Normalizasyon başlığı altında başvuru olan dönüşümler aşağıda açıklanmıştır.

Ondalık Ölçekleme

Ondalık ölçekleme ile normalleştirmede, ele alınan değişkenin değerlerinin ondalık kısmı hareket ettirilerek normalleştirme gerçekleştirilir. Söz konusu ölçekleme, sayısal değerlerin -1 ile $+1$ arasında yer almalarını sağlayacak biçimde dönüştürülmesine karşılık gelmektedir (Oğuzlar, 2003).

Min-Max Normalleştirme

Min-Max normalleştirme ile orijinal veri üzerinde doğrusal bir dönüşüm yapılır. Bu yöntem aracılığıyla veriler genellikle $[0, 1]$ aralığına dönüştürülmektedir. Bu yöntem, veri içindeki en büyük ve en küçük sayısal değer belirlenerek diğerlerini buna uygun biçimde dönüştürme esasına dayanmaktadır. Söz konusu değıştirme bağıntısı şu şekilde ifade edilmektedir:

$$X^* = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (2.1)$$

Burada X^* dönüştürülmüş değerleri, X gözlem değerlerini, X_{min} en küçük gözlem değerini ve X_{max} en büyük gözlem değerini ifade etmektedir (Özkan, 2013).

Z-Score Standartlaştırma

İstatistiksel veri dönüştürme teknikleri arasında yer alan ve en yaygın biçimde kullanılan bu yöntem, ele alınan verinin ortalama ve standart sapma değerlerini kullanır. Söz konusu değerlerin dönüşümünde şu şekilde bir bağıntıya yer verilmektedir:

$$X^* = \frac{X - \bar{X}}{\sigma_x} \quad (2.2)$$

Burada X^* dönüştürülmüş değerleri, X gözlem değerlerini, verilerin aritmetik ortalamasını ve σ_x gözlem değerlerinin standart sapmasını ifade etmektedir (Özkan, 2013).

2.2.4. Modelin Kurulması ve Değerlendirilmesi

Bu aşamada çeşitli modelleme yöntemleri seçilir ve uygulanır. Tipik olarak, aynı veri madenciliği problem tipi için çok çeşitli yöntemler bulunmaktadır. Tanımlanan problem için en uygun modelin bulunabilmesi, olabildiğince çok sayıda modelin kurularak denenmesi ile mümkün olmaktadır (Tsipstis ve Chorianopoulos, 2009). Tercih edilen model, verinin içindeki olası esas gizli düzen ya da örüntüyü yakalamak için kullanılmaktadır (Vellido ve diğ., 2012) Bazı yöntemler verinin düzenlenmesinde özel şartlara sahip olabilmektedir. Bu nedenle veri hazırlama ve model kurma aşamaları, en iyi olduğu düşünülen modele varılıncaya kadar yinelenen bir süreçtir.

Bir müşterinin davranış modelini kurabilmek için, ihtiyacımız olan:

- Modelin içerisinde yer alacak değişkenlerin belirlenmesi; satın alınan ürünler, müşteri sadakati oluşum süreci, vb.
- Modelin kurulması; farklı müşterilerin tanımlanarak, segmentlere ayrılması
- Modeli kullanarak tahminlerde bulunulması; hangi müşterinin satın almaya daha meyilli olduğu, çapraz satış tekliflerinin kime sunulacağı ya da kimin rakip firmayı tercih edebileceği gibi.
- En ayırt edici veri değişkenin tanımlanması; örneğin, müşterinin satın alma eğiliminin tahmini için gerekli en etkili değişken, çapraz satış teklifinin kabul edilme ya da reddedilme olasılığı (Baunsaythip ve Runsala, 2001).

Model kuruluş süreci denetimli (supervised) ve denetimsiz (unsupervised) öğrenimin kullanıldığı modellere göre ikiye ayrılmaktadır.

Denetimli öğrenmede, bir danışman tarafından ilgili sınıflar önceden belirlenen bir kritere göre ayrılır ve her bir sınıf için çeşitli örnekler verilir. Amaç verilen örneklerden hareket ederek her bir sınıfa ilişkin özelliklerin bulunmasıdır (Kotu ve Deshpande, 2015).

Denetimsiz öğrenmede, denetimli öğrenmede olduğu gibi her hangi bir bilgi verilmeden, ilgili örneklerin gözlenmesi ve bu örneklerin özellikleri arasındaki benzerliklerden hareket ederek sınıfların tanımlanması amaçlanmaktadır (Akpınar, 2000) Denetimli öğrenmede eğitim kümesindeki her bir girdi vektörü için ayrı bir çıktı değeri mevcutken, denetimsiz öğrenmede hiçbir çıktı değeri yoktur (Musicant, 2007).

2.2.5. Modelin Kullanılması

Modelin oluşturulması genellikle projenin sonu değildir. Modelin amacı, verilerin bilgisini arttırmak olsa da, kazanılan bilgilerin organize edilmesi ve müşteriye yararlı olacak şekilde sunulması gerekecektir. İhtiyaçlara bağlı olarak, uygulama fazı, bir rapor üretmek kadar basit olabileceği gibi, bir veri madenciliği süreci uygulamak kadar karmaşık olabilmektedir. Birçok durumda, uygulama adımlarını yürütecek olan veri analisti değil, müşteri olacaktır. Analist modeli kullansa bile, müşterinin yarattığı modelleri gerçekten kullanabilmek için yapılması gereken işlemleri anlamak önem arz etmektedir (Chapman, 2000).

2.2.6. Modelin İzlenmesi

Zaman içerisinde sistemlerin özelliklerinde ve dolayısıyla ürettikleri verilerde ortaya çıkan değişiklikler, kurulan modellerin sürekli olarak izlenmesini ve gerekiyorsa yeniden düzenlenmesini gerektirecektir (Akpınar, 2000).

2.3. VERİ MADENCİLİĞİ MODELLERİ

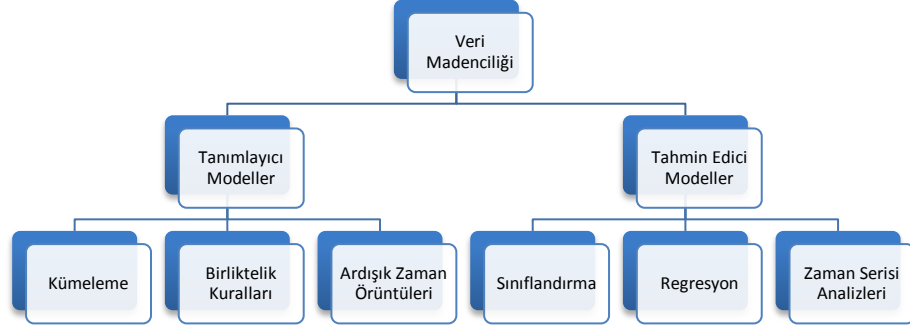
Veri madenciliği mevcut veri içerisinde var olan geçerli, potansiyel kullanışlı, kolayca anlaşılabilen korelasyonlar ve örüntüler bularak amacına ulaşmaktadır. Veri madenciliğinin bu amacı tanımlayıcı ya da tahmin edici modeller yaratarak sağlanmaktadır (Sondwale, 2015).

Veri madenciliği birçok tekniğin uygulandığı bir süreçtir. Veri madenciliği projelerinin uygulama alanlarına, sahip olunan veri setinin özelliklerine, hedeflenen yönetime göre farklılık arz edebilecek teknikler bulunmaktadır. Veri madenciliğinde geçerli teknikler, veri madenciliği projelerinde ihtiyaca göre tek başlarına kullanılacakları gibi, veri madenciliği projelerinin farklı farklı aşamalarında birlikte de kullanılabilirlerdir. Bir

teknikğin çıktısı diğer bir teknik girdisi olabilmektedir. Yine bir teknik için gerekebilecek en optimum veri seti, diğer bir teknik yardımıyla bulunabilmekte ve kullanıma hazır hale getirilebilmektedir (Kiremitçi, 2005).

Tanımlayıcı modellerde amaç, veri setinin temel özelliklerini ve karar vermeye rehberlik etmede kullanılacak mevcut verilerdeki örüntülerin tanımlanması sağlanmaktadır (Akpınar, 2000). Genel olarak, tanımlayıcı model yönsüz (undirected) veri madenciliği kapsamında görülmektedir. Yönsüz veri madenciliğinde aşağıdan yukarıya (bottom-up) yaklaşım benimsenmiştir. Veri içerisindeki örüntüler bulunur ancak bu örüntülerin yorumlanması veri madencisine bırakılmaktadır. Kümeleme, Birliktelik Kuralları ve Ardışık zamanlı örüntüler öngörülecek herhangi bir hedef değişken bulunmayan tanımlayıcı veri madenciliği modelleridir; Dolayısıyla test veri setlerine ihtiyaçları bulunmamaktadır.

Tahmin edici bir model, veritabanındaki bazı değişkenleri kullanarak, bilinmeyen ya da ilgili diğer değişkenlerin gelecek değerlerini tahmin etmemizi içermektedir (Fayyad ve diğ., 1996). Tahmin edici model de, modeli "öğrenmek" için önceden bilinen bir veri kümesine ihtiyaç duymaktadır (Kotu ve Deshpande, 2015). Bu modellerde, sonuçları bilinen verilerden hareket edilerek bir model geliştirilmesi ve kurulan bu modelden yararlanılarak sonuçları bilinmeyen veri kümeleri için sonuç değerlerin tahmin edilmesi amaçlanmaktadır (Karagöz, 2007). Veri madencisine, belirli bir değişkenin (hedef değişken), bilinmeyen bir değeri üzerinde tahminde bulunmasına izin vermektedir. Eğer hedef değer, önceden tanımlanmış belirli etiketlerden (sınıflardan) biri ise, veri madenciliği görevi sınıflandırma olarak adlandırılır. Bu nedenle tahmin edici model, bilinen muhtemel hedef değişkenin önceki değerlerini içeren, bilinen değerlerinden oluşturulmaktadır (Jain ve Srivastava, 2013). Sınıflandırma, regresyon modelleri ve zaman serisi analizleri tahmin edici modellerdir. Veri madenciliği modellerinin tanımlayıcı ve tahmin edici olarak sınıflandırılmasına Şekil 2.7' de yer verilmiştir.



Şekil 2.7: Tanımlayıcı ve Tahmin Edici Veri Madenciliği Modelleri.

2.3.1. Sınıflama ve Regresyon Modelleri

Denetimli (tahmin edici) öğrenme grubuna dahil olan sınıflandırma ve regresyon modelleri, girdi değişkenlerine dayalı bir hedef değişken tahmin eder. Tahmin, daha önce bilinen bir veri kümesinden oluşturulmuş genelleştirilmiş bir modele dayanmaktadır. Hedef değişken, reel bir sayı ise, model regresyondur (ör. Bir kredideki ipotek faiz oranı). Sınıflandırma, kategorik çıktı değişkenlerini öngörmektedir (Kotu ve Deshpande, 2015).

Sınıflandırma tekniği, müşteri segmentasyonunda, işletme sorunlarının modellenmesinde, kredi analizleri ve daha birçok alanda uygulama alanı bulunmaktadır (Kiang, 2002). Bu teknik, veritabanı kayıtlarını, belirli kriterlere göre daha önceden tanımlanmış sınıflara ayırarak, gelecekteki müşteri davranışlarını tahmin etmek için bir model oluşturmayı amaçlamaktadır (Ngai ve diğ., 2009). Regresyon, her veri nesnesini gerçek değere eşlemek için kullanılan istatistiksel tahmin tekniğidir (Carrier ve Povel, 2003). Regresyonun kullanımı; eğri uydurma, tahmin, nedensel ilişkilerin modellenmesi ve değişkenler arasındaki ilişkiler hakkında bilimsel hipotezlerin test edilmesini içermektedir. Regresyon için ortak araçlar arasında doğrusal regresyon ve lojistik regresyon bulunmaktadır.

Sınıflandırma ve regresyon modellerinde karar ağaçları, yapay sinir ağları, genetik algoritmalar, k-en yakın komşu ve naive-bayes gibi teknikler kullanılmaktadır (Ayık ve diğ., 2007).

2.3.1.1. Karar (Sınıflandırma) Ağaçları

İstatistiksel yöntemlerde veya yapay sinir ağlarında veriden bir fonksiyon öğrenildikten sonra bu fonksiyonun insanlar tarafından anlaşılabilir bir kural olarak yorumlanması zordur.

Bu teknik, birbiriyle ilişkili karar dizilerini açıklayan modelleri çıkarmak veya gelecekteki veri eğilimlerini tahmin etmek için kullanılabilir (Jensen, 2008). Karar ağaçları, yukarıdan aşağı, genelden spesifik bir doğrultudaki eğitim verilerinden üretilmektedir. Bir karar ağacının başlangıç durumu, eğitim setindeki tüm örnekleri atayan kök düğümdür. Tüm örneklerin aynı sınıfa ait olması durumunda, örnekleri bölmek için daha fazla karar alınması gerekmez ve çözüm tamamlanmış olur. Bu düğümdeki örnekler iki veya daha fazla sınıfa ait ise, düğümde bir bölünmeye neden olacak bir test yapılır. Bu işlem, tamamen ayrımcı bir ağaç elde edilinceye kadar yeni ara düğümlerin her biri için yinelemeli bir şekilde tekrarlanır. (Apte ve diğ., 1997).

2.3.1.2. Yapay Sinir Ağları (YSA)

1980'lerden sonra yaygınlaşan yapay sinir ağlarında, fonksiyon birbirine bağlı basit işlemci ünitelerinden oluşan bir ağ üzerine dağıtılmıştır. YSA istatistiksel yöntemler gibi veri hakkında parametrik bir model varsaymaz yani uygulama alanı daha geniştir ve bellek tabanlı yöntemler kadar yüksek işlem ve bellek gerektirmemektedir (Alpaydın,200). Gelecek değerlerin etkili bir şekilde tahmin edilebilmelerini amaçlamaktadır (Enke ve Thawornwong, 2005).

YSA, motorlu bir taşıtın yönlendirilmesi, karakterize edilmemiş DNA dizilerindeki genlerin tanınması, döviz kurlarının öngörülmesi gibi ilginç görevlerin yerine getirilmesini sağlayacak modellerin öğrenilmesini yerine getirebilecek modeller için çok çeşitli alanlarda uygulanmaktadır (Craven ve Shaylik, 1997).

2.3.1.3. Genetik Algoritmalar

Genetik algoritmalar doğal sistemlerin mekanizmalarını açıklamak için geliştirilen arama algoritmalarıdır: Genetik algoritma uygulamalarında çözümün değişkenleri, genlerin bir listesini sunan yapılandırılmış bir dizgeye kodlanmaktadır (Koonce ve Tsai, 200).

Karmaşık çok boyutlu arama uzayında en iyinin hayatta kalması ilkesine göre bütünsel en iyi çözümü aramaktadırlar. Genetik algoritmalar problemlere tek bir çözüm üretmek yerine farklı çözümlerden oluşan bir çözüm kümesi üretmektedirler (Ayık ve diğ., 2007).

2.3.1.4. Bellek Tabanlı Yöntemler (*K En Yakın Komşu*)

Bellek Tabanlı Yöntemler, günümüzde bilgisayarların ucuzlaması ve kapasitelerinin artmasıyla, özellikle de çok işlemcili sistemlerin yaygınlaşmasıyla, kullanılabilir hale gelmiştir (Han ve Kamber, 2006). Bu yönteme en iyi örnek **en yakın k komşu algoritmasıdır** (k-nearest neighbor) (Alpaydın, 2000). Bu algoritmada sınıflandırılmak istenen nesne, öznitelik değerlerine göre kendisine en yakın komşu veya komşuların sınıfına atanmaktadır. Performansı en yakın komşuları tanımlamak için kullanılan mesafe metriğine göre belirlenmektedir (Weinberger ve Saul, 2009).

2.3.1.5. Naive Bayes

Naive Bayes sınıflandırıcısı Bayes teoreminin bağımsızlık önermesiyle basitleştirilmiş halidir. Olasılık yaklaşımları verinin nasıl üretildiğine dair güçlü varsayımlarda bulunmakta ve bu varsayımları somutlaştıran olasılıksal modeller oluşturmaktadır: Naive bayes, örüntü tanıma problemine ilk bakışta oldukça kısıtlayıcı görülen bir önerme ile kolaylıkla kullanılabilen olasılıkçı bir yaklaşımdır (McCallum ve Nigam, 1998). Bu önerme örüntü tanıma da kullanılacak her bir tanımlayıcı öznitelik ya da parametrenin istatistik açıdan bağımsız olması gerekliliğidir. Bu yöntemle bir örneğin hedef niteliğin sınıf değerlerine ait olma olasılıkları bulunabilmektedir (Han ve Kamber, 2006).

Bayes teoremi aşağıdaki denklemle ifade edilmektedir;

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \quad (2.3)$$

$P(A|B)$; B olayı gerçekleştiği durumda A olayının meydana gelme olasılığıdır.

$P(B|A)$; A olayı gerçekleştiği durumda B olayının meydana gelme olasılığıdır.

$P(A)$ ve $P(B)$; A ve B olaylarının önsel olasılıklarıdır.

Burada önsel olasılık bayes teoreminine öznellik katar. Diğer bir ifadeyle örneğin $P(A)$ henüz elde veri toplanmadan, A olayı hakkında sahip olunan bilgidir. Diğer taraftan $P(B|A)$ veri toplandıktan sonra, A olayının gerçekleşmiş olduğu durumlarda B olayının gerçekleşme ihtimali hakkında bilgi verdiği için ardıl olasılıktır (Pawlak, 2003).

2.3.2. Kümeleme Modelleri (Clustering)

Kümeleme, verilerin kümelerle gruplandırılması işlemidir, böylece, ancak diğer kümelerdeki nesnelere kıyasla, bir kümedeki nesnelere birbirleri ile yüksek benzerlik

göstermektedir. Amaç, gerçekleştirilen bölümlenme işlemi sonunda elde edilen kümelerin, küme içi benzerliklerinin maksimum ve kümeler arası benzerliklerinin ise minimum olmasını sağlamaktır. Genellikle mesafe ölçütleri kullanılır (Han ve Kamber, 2006). Sınıflandırma yöntemleri öngöründe bulunurken, kümeleme yöntemleri tanımlayıcıdır (Rokach ve Maimon, 2010).

Kümeleme analizi, pazar araştırması, örüntü tanıma, veri analizi ve görüntü işleme gibi birçok uygulamada yaygın olarak kullanılmaktadır. Pazarlamacıların müşteri tabanında farklı grupları keşfetmesine yardımcı olabilmektedir. Müşteri gruplarını satın alma örüntülerine dayalı olarak karakterize edebilmektedir (Tutorialspoint, 2016). Tüm müşteriler kendisiyle benzer özelliklere, niteliklere sahip olan müşterilerle aynı gruba atanmaktadır. Kümeleme analizinin sonuçlarını kullanacak kişilerin, ayrışan bu grupları daha sonradan tanımlaması ve pazar bölümü olarak hedeflemesi mümkündür (Kiremitçi, 2005). Kümeleme yöntemleri Bölümlenme Yöntemleri, Hiyerarşik Yöntemler, Yoğunluk Tabanlı Yöntemler, Model Tabanlı Yöntemler, Şebeke Tabanlı Yöntemler (Grid-Based Method), Kısıt Tabanlı Yöntemler şeklinde kategorilere ayrılmaktadır.

2.3.3. Birliktelik Kuralları ve Ardışık Zamanlı Örüntüler

Olayların birlikte gerçekleşme durumlarını çözümleyen veri madenciliği yöntemlerine birliktelik kuralları adı verilmektedir (Özkan, 2008). Birliktelik kuralları bir arada sık olarak görülen ilişkilerin ortaya çıkarılmasını ve özetlenmesini sağlayan veri madenciliğinin tanımlayıcı modellerindendir (Taş ve diğ., 2012). Geçmiş verilerden çıkarılmış olan kurallar, gelecek ile ilgili elimizde referans oluşturup, ilerisi ile ilgili kararlar verebilmek için temel oluşturacak ve doğru tahminler yapmamamızı sağlayacaktır (Vijayalakshmia ve Pethalakshmi, 2014).

Birliktelik kuralları, pazarlama amaçlı olarak pazar sepet analizi (Market Basket Analysis) adı altında veri madenciliği uygulamaları arasında sıkça kullanılmaktadır. Bu analizde amaç müşterilerin alışveriş alışkanlıklarının veri tabanındaki bilgiler aracılığıyla ortaya çıkartılması işlemidir (Silahtaroglu, 2016). Bir alış-veriş sırasında müşterinin hangi ürün veya hizmetleri satın almaya eğilimli olduğunun belirlenmesi, böylelikle gelecekte müşteriye özel ürün paketleri geliştirilerek, müşteri satın almaya teşvik edilmektedir.

Satın alma eğilimlerinin tanımlanmasını sağlayan birliktelik kuralları, çok çeşitli gerçek dünya uygulamalarında kullanılabilir. Ticaret, finans, mühendislik, fen ve sağlık sektörlerinin birçok alanlarında uygulama sahası bulunmaktadır (Birant ve diğ., 2010). Süper marketlerde satış, çapraz satış programları, katalog ve yerleşim düzeni tasarımları (raf düzenlemeleri), promosyon analizleri ve metin araması işlemleri, web sayfalarında ziyaretçilerin hangi sayfaları birlikte tıkladığı ile ilgili analizler için kullanılabilir (Erpolat, 2012). Sadece pazarlama alanında değil literatürde; eğitimde öğrenci başarılarının analiz edilmesi, tıp dünyasında hastalıkların belirtilerinin tespitinde (Delen ve diğ., 2005); bağıntılı olarak geçirilen hastalıkları belirlemede, kablosuz sensor ağlarında, özellikle de insan vücudunda veya ev kullanım nesnelerinde, insan bedeninde, sensörler bulunan akıllı evlerde ya da kullanıcının ortamının izlenmesini gerektiren diğer uygulamalar (gaz sızıntısı, yangın, patlama gibi) için kullanılabilirliğine rastlanmaktadır (Rashid ve diğ., 2013). Trafik kazalarının hangi koşullarda gerçekleştiğinin analizinde, demans hastalarının gün içerisindeki aktivitelerinin takibinde kullanılabilir. Demans hastaları arasında işlevsel düşüşü izlemek için akıllı ortamlardaki günlük yaşam aktivitelerini izleyebilme yeteneği önemli bir yaklaşım olarak görülmektedir (Nasreen ve diğ., 2014).

Ardışık zamanlı örüntüler, sıralı olarak verilmiş verileri ele almaktadır. Geçmiş olay dizilerinden oluşan sıralı veritabanı göz önüne alınarak, mevcut olay dizisindeki bir sonraki olay tahmin edilmeye çalışılmaktadır (Rudin ve diğ., 2011).

Müşteri işlemlerinden oluşan bir veritabanını ele alacak olursak, sırasıyla şu niteliklere sahip olduğu düşünülmektedir; sıra-kimliği ya da müşteri kimliği, işlem-zamanı ve işlemde yer alan ürünler (Srikant ve Agrawal, 1996). Böyle bir veritabanına veri dizileri tabanı denilmektedir. Daha doğrusu, her işlem bir takım öğelerden oluşup, işlem zamanına göre sıralanmaktadır. Karar verme sürecine etkili bir şekilde yardımcı olabilmek için, amaç kullanıcının bakış açısına göre tipik davranışlar elde etmektir (Masseglia ve diğ., 2005).

Ardışık zaman örüntüleri, birçok veri türü zamanla ilgili olduğundan geniş bir uygulama yelpazesinde sunulabilmektedir. Örneğin, bir müşteriye ait satın alma veritabanında sıralı bir düzen, pazarlama ve ürün stratejileri geliştirmek için kullanılabilir. Web log analizi yoluyla, veri örüntüleri bir şirketin web sitesini en popüler bağlantılara daha kolay

erişim sağlamak için yapılandırmaktadır. Ayrıca telekomünikasyon ağ alarm veritabanları, saldırı tespiti, DNA dizileri gibi alanlarda uygulamalarına rastlanabilmektedir.

Ardışık zamanlı örüntüler ile ilgili örnekleri aşağıdaki şekilde çoğaltabiliriz:

- X ameliyatı yapıldığında, 15 gün içinde % 45 ihtimalle Y enfeksiyonu oluşacaktır.
- Çekiç satın alan bir müşteri, ilk üç ay içerisinde % 15, bu dönemi izleyen üç ay içerisinde %10 ihtimalle çivi satın alacaktır (Şen, 2008).
- Otomobil satın alanlar, %90 ihtimalle 1 hafta içerisinde trafik sigortası yaptıracaklardır.

Bölüm 2.4' de birliktelik kuralları daha detaylı olarak ele alınmıştır.

2.4. BİRLİKTELİK KURALLARI

Birliktelik kuralları analizi, yaygın olarak kullanılan veri madenciliğinin tanımlayıcı yöntemlerinden birisidir. Veriler arasındaki birlikteliklerin, ilişkilerin ve bağıntıların kurallar halinde bulunması işlemidir. Birliktelik Kuralları, Agrawal ve Srikant tarafından 1993 yılında geliştirilmiştir. $I = \{i_1, i_2, \dots, i_m\}$ kümesine öge kümesi denilmekte, her bir i öge (ürün) olarak adlandırılmaktadır. D , tüm hareketleri, T ise öğelerin her bir hareketini temsil etmektedir. Her hareket bir tanımlayıcı olan TID ile temsil edilmektedir. A ve B öğelerin kümeleri olsun. Bir T işlemler kümesi ancak ve ancak $A \subseteq T$ ise yani A , T 'nin alt kümesi ise A 'yı kapsıyor denilmektedir.

Bir birliktelik kuralının gösterim formu $A \Rightarrow B$ şeklindedir.

2.4.1. Birliktelik Kuralları Temel Kavramları

Birliktelik kuralları, işlemlerden oluşan ve her bir işlemin de elemanlarının birlikteliğinden oluştuğu düşünülen bir veritabanında, bütün birliktelikleri tarayarak, sık tekrarlanan birliktelikleri veritabanında ortaya çıkarmaktır (Agrawal ve Srikant,1996).

Veritabanı kayıtlarında, nesnelerin gruplandırılması ile elde edilen bağımlılık ilişkilerinin yüzde yüz geçerli olması beklenemez. Ancak, çıkarsaması yapılan kuralın, veritabanının önemli bir kısmı tarafından desteklenmesi, yani söz konusu durumun sıkça görülüyor

olması gerekmektedir. Burada, $A \subset I, B \subset I$ ve $A \cap B = \emptyset$ şeklindedir. $A \Rightarrow B$ kuralı için, tüm işlem hareketlerinin tutulduğu D kümesinde yer alan öğeler için s ; **destek** değeri olarak tanımlanmaktadır (Agrawal ve Srikant, 1994).

Destek değeri, A ve B nesnelerinin birlikte bulunduğu kayıt sayısının, veritabanındaki tüm kayıt sayısına oranı olarak hesaplanmaktadır (Birant ve diğ., 2010). Bu kriter, veride öğeler arasındaki bağıntının ne kadar sık olduğunu belirtmektedir. T işleminin A ve B 'yi (A ve B kümelerinin birleşimi, ya da hem B , hem A) içermeye olasılığıdır. Bu çalışmada destek değeri $P(A, B)$ olarak ele alınacaktır. İkinci olarak, $A \Rightarrow B$ kuralının g ile gösterilen **güven** değeri tanımlanmaktadır. Bu kriter çıkarımın (kuralın) gücünü ve güvenilirliğini temsil etmektedir (Nahar ve diğ., 2013). $P(B / A)$ şartlı olasılığı olarak ele alınmaktadır. A ve B nesnelerinin birlikte bulunduğu kayıt sayısının, A nesnesinin (veya nesnelerinin) bulunduğu kayıt sayısına oranıdır. Tablo 2.1' de bu ölçülerin hesaplanmasına yer verilmiştir:

Tablo 2.1: Birliktelik Kurallarında Destek ve Güven Ölçülerinin Hesaplanması.

$$\begin{aligned} \text{Destek}(A \Rightarrow B) &= P(A, B) \\ \text{Güven}(A \Rightarrow B) &= P(B / A) \text{ ya da} \\ \text{Güven}(A \Rightarrow B) &= \text{Destek}(A \Rightarrow B) / \text{Destek}(A) \end{aligned}$$

Kuralın destek ve güven değerleri, kuralın ilginçliğini ifade eden iki ölçüdür. Bu değerler sırasıyla keşfedilen kuralların yararlılığını (kullanışlılığını) ve kesinliğini (doğruluğunu) ifade etmektedir (Han ve Kamber, 2006).

Destek ve güven değerlerinin eşik değerleri kullanıcı tarafından belirlenmektedir. Çıkarılan kuralın geçerliliği aşağıdaki şartları sağlıyor ise onaylanmaktadır (Zhang ve Zhang, 2007).

$$\text{i) destek}(X \Rightarrow Y) > \text{mindestek} \quad (2.4)$$

$$\text{ii) güven}(X \Rightarrow Y) = \frac{\text{destek}(x \Rightarrow y)}{\text{destek}(x)} \geq \text{min}(\text{güven}) \quad (2.5)$$

Bir ürün kümesindeki destek, D ile ifade edilen tüm hareketler içinde ilgili ürün kümesini içeren hareketlerin yüzdesidir. Çalışma boyunca sıklıkla kullanılacak olan terimlerden bazılarının açıklamalarına aşağıda yer verilmiştir:

Öge kümesi (itemset): öğelerden oluşan küme olarak adlandırılmaktadır. K-öge kümesi dediğimizde, k tane öğeden oluşan bir küme düşünülebilmektedir.

Sık öge kümesi (frequent itemset): minimum destek değerine sahip öge kümesi için kullanılmaktadır. Bazı kaynaklarda yaygın öge kümesi olarak da kullanılmaktadır.

Aday küme: Belli gereksinimlere uyup uymadığını görmek için bir dizi test gerektiren öğelerin kümesidir.

Kullanıcının belirlediği minimum destek değerinden yüksek değere sahip kurallar için, minimum destek değerine sahiptirler denilebilmektedir. Kullanıcının belirlediği minimum destek değerine eşit veya büyük destek değerine sahip öge kümelerinin oluşturulması, buna sık öge kümelerinin oluşturulması da denilmektedir (Lai ve Charpa, 2001). Minimum destek eşik değerini ve minimum güven eşik değerini sağlayan kurallar güçlü (strong) kurallar şeklinde ifade edilmektedir.

Genel olarak bu değerler 0 - 1 aralığından ziyade, % 0 - % 100 aralığında belirtilmektedir (Han ve Kamber, 2000). **Bu değerler 1'e ne kadar yakınlarsa nesnelere arasında o kadar güçlü ilişki olduğunu göstermektedirler. Dolayısıyla, bağlantının önemli olması için her iki değer de olabildiğince yüksek olması gerekmektedir.**

Birliktelik kuralı çıkarımında kullanılan bazı algoritmalar mevcuttur. Bu algoritmalarda minimum güvenilirlik ve destek metriklerini sağlayan birliktelik kuralı çıkarım problemi iki aşamada gerçekleşmektedir. Birinci adım, kullanıcı tarafından belirlenmiş minimum destek kriterini sağlayan ürün kümelerinin bulunmasıdır. Bu kümelere sık geçen öge kümesi adı verilmektedir. Verilen örnekte n adet ürün (öge) var ise, potansiyel olarak 2^n adet sık geçen öge kümesi olabilmektedir ve bunların hepsi sık geçen öge küme olabilmektedir. Birliktelik kuralları madenciliğinin en önemli problemi sık geçen nesne kümelerinin sayısının çok fazla olmasıdır. İkinci adım ise sık geçen öge kümeleri kullanılarak minimum güvenlik kriterini sağlayan birliktelik kurallarının bulunmasıdır. Minimum destek eşikğine göre üretilen çözüm uzayında, minimum güvenilirlik eşikğine göre taranarak bulunan birliktelikler kullanıcının ilgilendiği ve potansiyel olarak önemli bilgi içeren birlikteliklerdir (Liu ve diğ., 1998).

Büyük veri tabanlarında özellikle de minimum destek eşik değerinin çok düşük olduğu veri tabanlarında sık geçen nesne küme sayısı çok büyük sayılara ulaşabilmektedir (Ay ve Çil, 2010). Bu nedenle algoritmanın performansını belirleyen adım birinci adımdır denilmektedir. Sık geçen öge kümeleri belirlendikten sonra, birliktelik kurallarının bulunması daha basit bir adım olarak nitelendirilmektedir (Karabatak ve İnce, 2004).

Birliktelik kuralları sonucunda bir dizi kural keşfedilmektedir. Amaç; bu kurallar içerisinden kullanıcıya yararlı olanları çekip çıkarabilmektir. Yararlılığı ölçme objektif ve sübjektif olmak üzere iki şekilde yapılabilmektedir. Objektif ölçümler destek, güven gibi verinin istatistiksel ölçümlerini içermektedir (Agrawal, 1993). Sübjektif ölçümler ise kullanıcı yönelimlidir. Beklenmezlik ve işe yararlılık sübjektif ölçümlere örnek gösterilebilmektedir.

Beklenmezlik de vurgulanan, kural eğer öncesinde kullanıcı tarafından hiç bir şekilde bilinmiyorsa değerlidir. Örneğin bir market veritabanında, meşrubat alan kişinin yanında büyük olasılıkla cips ya da kuruyemiş alması potansiyeli yüksek bir birliktelik türüdür. Bu gibi örneklerde önceden tahmin edilmesi zor olmayan ve çıplak gözle seçilebilir birliktelikler mevcuttur. Amaçlanan ise saklı olan örüntüleri ortaya çıkartmaktır.

İşe yararlılık da ise vurgulanan, kurallar kullanıcıya avantaj sağlıyor ise kullanışlıdır. Bu nedenle ortaya çıkardıkları bilgi hiçbir şekilde kullanılmayan kurallar kullanışlı değildir sonucu ortaya çıkmaktadır. Bu nedenle sübjektif ölçülerin tespitinin durum bazında değerlendirilmesi gerekebilir. Bu ölçülerin belirlenmesi karmaşık olabilmektedir. Bu nedenle sıklıkla objektif ölçüler tercih edilmektedir. Tablo 2.2 ve Tablo 2.3' de destek ve güven değerlerinin bulunmasına yönelik hesaplama örneklerine yer verilmiştir.

Destek değerinin hesaplanmasına Tablo 2.2'de yer verilmiştir;

Tablo 2.2: Destek Değerinin Hesaplanması Örneği (Lai ve Cerpa, 2001).

TID Öğeler Destek = Bulunma/Toplam Destek

1	ABC	Toplam Destek = 5
2	ABD	Destek{AB} = 2/5 = %40
3	BC	Destek {BC} = 3/5 = %60
4	AC	Destek{ABC} = 1/5 = %20
5	BCD	

Güven değerinin hesaplanması;

Tablo 2.3: Güven Değerinin Hesaplanması Örneği (Lai ve Cerpa, 2001).

TID	Öğeler	A ⇒ B için; Güven = {B}'nin Bulunması / {A}'nın Bulunması
1	ABC	
2	ABD	Güven {A⇒ B} = 2/3= %66
3	BC	Güven {B⇒ C} = 3/4= %75
4	AC	Güven {AB⇒C} = 1/2 = %50
5	BCD	

2.4.2. Birliktelik Kurallarının Çeşitleri

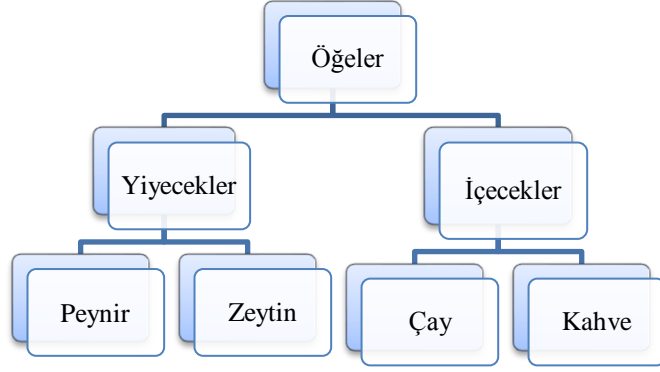
Sık örüntü madenciliğinden birliktelik kuralları, aşağıdaki ölçütlere dayalı olarak çeşitli şekillerde sınıflandırılabilir. Çalışmanın bu kısmında literatürde veri madenciliğinin duayeni sayılabilecek Han ve Kamber'in veri madenciliği alanındaki çalışmalarına yer verdikleri "Data Mining Concepts and Techniques, 2006" adlı eserleri esas alınmıştır.

Kuralda yer alan verilerin seviyelerine göre:

2.4.2.1. Tek Seviyeli Birliktelik Kuralları (Single Level Association Rules)

Bu kurallar yalnızca aynı sınıflandırma düzeyinde (taksonomi) bulunan kurallardan elde edilebilmektedir (Agrawal,1993). Burada bahsedilen sınıflandırma düzeyi (taksonomi), aynı seviyede bulunan ilişki hiyerarşisini temsil etmektedir.

Şekil 2.8' de taksonomi örneğine yer verilmiştir:

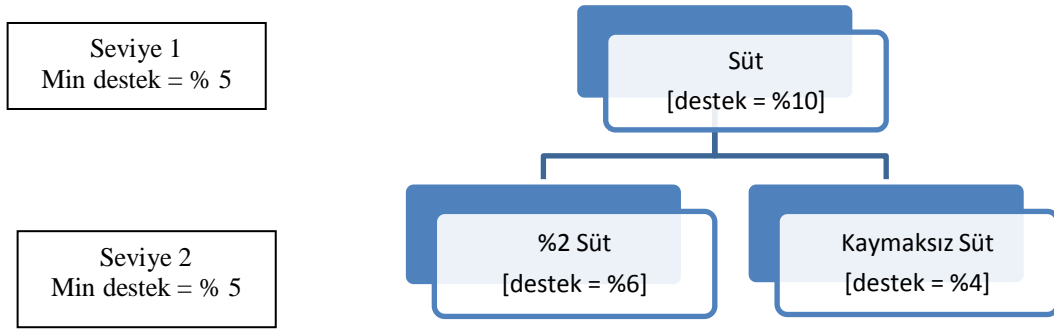


Şekil 2.8: Taksonomi Örneği (Lai ve Cerpa, 2001).

Tek seviyeli madencilik de birliktelik kuralları, kuralları iki aşamada ortaya çıkartmaktadır. İlk aşamada, verilerin analizi ile sık öge kümeleri bulunmakta, ikinci aşamada, öğelerden birliktelik kuralları çıkarılmaktadır (Agrawal vd., 1993). Problemin hesaplama karmaşıklığının sık öge setlerini bulmakta yattığı kabul edilmektedir. Bu sık öge kümelerinden yaratılan birliktelik kuralları doğrusal olduğundan, toplam performans üzerinde çok az etkisi bulunmaktadır (Lai ve Cerpa, 2001). Tseng (2000) tarafından tek seviyelinin bir avantajı, taksonomideki atalarla (ancestor) ilgilenmediği için daha az hesaplama gerektirmesi, bir başka avantajı ise kullanıcıların basit taksonomi altındaki birliktelik kuralları ile ilgilenmeleri olarak belirtilmiştir.

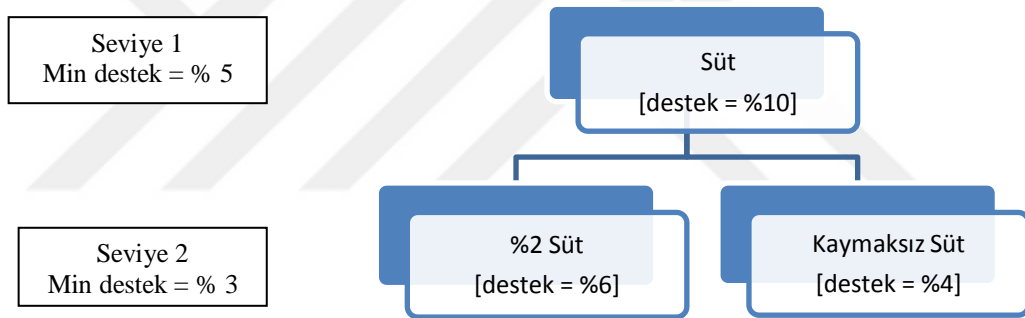
2.4.2.2 Çok Seviyeli Birliktelik Kuralları (Multilevel Association Rules)

Eğer öğelerin geçme sıklıkları çok değişiyorsa iki temel sıkıntı vardır: Eğer minimum destek değeri çok yüksek seçilirse, nadir geçen öğeleri içeren kurallar bulunamaz. Bu kuralları da çıkartabilmek için, minimum destek değeri çok düşük seçilmelidir. Bu seferde çok fazla kural elde edilecektir. Çok fazla yaygın öge olacağından, her öge bir diğeri ile ilişkilymiş gibi gözükcektir. Çok seviyeli yapı bu sorunu çözmektedir. Burada öğeler genellikle hiyerarşik bir yapı içermektedir. Daha alt seviyedeki ürünlerin daha düşük desteğe sahip olması beklenmektedir. Bu öğelerden güçlü kurallar çıkarmak zordur, bu nedenle birden fazla min_ destek seviyesi kullanılmaktadır. Şekil 2.9'da her iki seviyede de aynı destek değeri kullanılarak, ikinci seviyedeki öge elenmiştir. Burada atası minimum desteğe sahip olmayan herhangi bir öğeyi içeren öge kümesini test etmeye gerek bulunmamaktadır. Düşük seviye öğeler sık olarak ortaya çıkmamaktadır.



Şekil 2.9: Çok Seviyeli B. Kurallarda Destek Değeri (Han ve Kamber, 2006).

Şekil 2.10'da ise azaltılmış destek değeri kullanılmıştır. Her seviyenin kendi destek değerleri belirlenmiştir. Daha alt seviye için daha düşük destek değeri belirlenerek, her iki seviyedeki öğeler sık olarak nitelendirilmektedir.



Şekil 2.10: Çok Seviyeli Kurallarda Destek Değeri_2 (Han ve Kamber, 2006).

Han (2006), kuralda yer alan her bir farklı eylemi (predicate) bir boyut olarak ifade etmiştir. Bu nedenle, Denklem 2.6'da belirtilen kural için tek boyutludur denilebilmektedir. Sadece almak eylemi ve onun çoklu oluşumlarından (kural içerisinde birden fazla kez tekrarlanması) meydana gelmektedir.

$$alır(X, "dijital kamera") \rightarrow alır(X, "HP yazıcı") \quad (2.6)$$

Bir firmanın satış ve bununla ilişkili diğer bilgileri işlemsel veritabanlarında tutmaktansa, ilişkisel veritabanlarında ya da veri ambarlarında saklandıklarında, sadece işlem verilerine ulaşmak yerine, o satışta yer alan ürünlere ait miktar, fiyat, satışın gerçekleştiği şube gibi ürün ile ilgili diğer nitelikler de saklanabilmektedir. Bunun yanı sıra ürünü satın alan müşteriye ait, müşterinin yaşı, işi, kredi aralığı, adresi gibi ekstra bilgiler de

saklanabilmektedir. Her bir veritabanı niteliğini ya da veri ambarı boyutunu bir eylem olarak göz önünde bulundurursak, buradan hareketle, kural 2.7' deki gibi çoklu eylemler içeren birliktelik kuralları çıkarılabilmektedir.

$$yaş(X, "20...29) \wedge iş(X, "öğrenci") \rightarrow satın alır(X, "laptop") \quad (2.7)$$

Kuralda yer alan verilerin boyutlarına göre;

2.4.2.3. Çok Boyutlu Birliktelik Kuralları

İki veya daha fazla boyut ya da eylem içeren birliktelik kurallarıdır. Kural 2.7' de verilen örnekte üç eylem bulunmaktadır (yaş, iş ve satın alma) ve her biri, kuralda yalnızca birer kez geçmektedir. Bu nedenle, tekrar eden eylem yoktur. Tekrarsız (sağ (rhs) ve sol (lhs) taraftaki nitelikler farklı) nitelikler içeren çok boyutlu birliktelik kurallarına **boyutlar arası birliktelik kuralları (interdimensional)** denilmektedir. Tekrar eden eylemlerden oluşan çok boyutlu birliktelik kurallarına **karma boyutlu birliktelik kuralları (hybrid-dimensional)** denilmektedir.

Bu tür kurallara örnek olarak, satın alma eyleminin tekrar edildiği 2.8' deki kural örnek verilebilmektedir.

$$yaş(X, "20...29) \wedge satın alır(X, "laptop") \rightarrow satın alır(X, "HP printer") \quad (2.8)$$

Veritabanında yer alan niteliklerin türüne göre: Veritabanı nitelikleri kategorik ya da kantitatif olabilmektedir. Kategorik nitelikler; sadece sınırlı değerler veya kategorilerle ölçülebilen niteliklerdir (marka, meslek, renk vb.). Değerler arasında bir sıralama yoktur (Powers ve Xie, 2000). Kantitatif nitelikler ise, doğrudan sayılarla ölçülebilen (ücret, fiyat, ağırlık, uzunluk vb.) niteliklerdir. Kategorik nitelikler aynı zamanda nominal nitelikler olarak da anılmaktadır. Çok boyutlu birliktelik kurallarının çıkarımı kantitatif niteliklere yaklaşım açısından bazı kaynaklara göre iki, bazı kaynaklar göre ise üç gruba ayrılmaktadır. Bu çalışmada üç yaklaşıma da yer verilmiştir.

İlk yaklaşımda, kantitatif nitelikler önceden tanımlanmış kavram hiyerarşisi ile ayrıklaştırılmaktadırlar. Ayrıklaştırma çıkarım öncesinde meydana gelmektedir. Örneğin gelir için bir kavram hiyerarşisi, aralık etiketleri sayesinde "0...20", "21...30", "31...40" gerçek sayısal değerleri yerleştirmemiz sırasında kullanılabilir. Burada

ayrıklaştırma **statik ve önceden tanımlıdır**. Aralık etiketleri ile ayrıklaştırılmış sayısal değerler, kategorik nitelikler olarak değerlendirilebilmektedir. Bu yaklaşıma **kantitatif niteliklerin statik ayrıklaştırılması kullanılarak çok boyutlu birliktelik kurallarının çıkarımı** olarak adlandırılmaktadır.

İkinci yaklaşım, kantitatif niteliklerin **verilerin dağılımı esas alınarak** ayrıklaştırılması ya da gruplara ayrılmasıdır. Bu gruplar daha sonra çıkarım süresince bir araya getirilebilmektedir. Ayrıklaştırma süreci dinamiktir ve çıkan kuralların güven değerlerini maksimize etmek gibi çıkarım kriterlerini sağlamak için kurulmuştur. Çünkü bu stratejide sayısal nitelikler, önceden tanımlanmış aralık ya da kategori olarak ele alınmaktadır. Bu yaklaşımdan çıkarılan birliktelik kuralları (dinamik) **kantitatif birliktelik kuralları** olarak adlandırılmaktadır.

Üçüncü yaklaşımda, kantitatif veriler bu aralık verisinin semantik anlamını yakalayacak şekilde ayrık hale getirilmektedir. Bu dinamik ayrıklaştırma prosedürü veri noktaları arasındaki uzaklığı göz önünde bulundurmaktadır. Bu nedenle bu kantitatif birliktelik kuralları **uzaklığa dayalı birliktelik kuralları** olarak adlandırılmaktadır (Han ve Kamber, 2006).

Aşağıda, ikinci yaklaşım olan Kantitatif Birliktelik Kuralları daha detaylı olarak ele alınmıştır.

2.4.3. Birliktelik Kurallarında Başvurulan Diğer Ölçütler

Birliktelik kurallarını elde ederken, dikkat edilecek bazı hususlar bulunmaktadır. Bütün güçlü birliktelik kuralları ilginç kurallar kapsamına girmemektedir. İstatistiksel olarak ilişkili olanları ortaya çıkarmak gerekmektedir. Bu nedenle, birliktelik kuralları daha anlamlı ilişkili kurallar elde etmek için bir korelasyon ölçüsü ile birlikte arttırılmalıdır. Bu noktada başvurulabilecek birçok korelasyon ölçüsü bulunmaktadır. Bunlardan bazıları kaldıraç, X^2 , tüm_güven (all_conf) ve cosine olarak literatürde yerini almaktadır (Hashler ve Kornik, 2007).

Bunlardan en sıklıkla kullanılan kaldıraç (lift) ölçüsüdür. Güçlü kurallar her zaman önemli olmayabilir. Bu sadece bir yaklaşımdır. İlişkinin gerçek gücünü ölçemez.

Kaldıraç değeri, X ve Y'nin istatistiksel olarak bağımsız olması halinde, ne kadar birlikte geçtiklerini ortaya koymaktadır.

Tablo 2.4: Destek Güven Hesaplaması.

	Basketbol evet	Basketbol hayır	Toplam
Mısır gevreği - evet	2000	1750	3750
Mısır gevreği- hayır	1000	250	1250
Toplam	3000	2000	5000

Tablo 2.4'de, 5000 öğrenci içerisinde;

- 3000 öğrenci basketbol oynamaktadır
- 3750 öğrenci mısır gevreği yemektedir
- 2000 öğrenci hem basketbol oynamakta hem mısır gevreği yemektedir.

Basketbol oynayıp ⇒ mısır gevreği yemek kuralı sonucunda çıkan [destek,güven] [%40, %66.7] değerleri yanıltıcı olabilmektedir. Çünkü mısır gevreği yiyenler tüm öğrencilerin %75'ini oluşturmaktadır ve bu oran %66.7'den yüksektir.

Basketbol oynayıp ⇒ mısır gevreği yemeyenler kuralı sonucunda çıkan [destek,güven] [%20, %33.3] daha düşük destek ve güven değeriyle de olsa çok daha doğrudur (Han ve Kamber, 2006).

Bu nedenle kuralın doğruluğunu sınarken ek bazı ölçütlere ihtiyaç duyulmaktadır. Bağımlı ve birbiri ile ilişkili olaylar için bir ölçüte ihtiyaç duyulmaktadır.

$$Lift(X,Y) = \frac{P(X,Y)}{P(X)P(Y)} \quad (2.9)$$

Denklem (2.9) 'un sonucu 1'den küçükse, X'in oluşumu Y'nin oluşumu ile negatif yönde ilişkilidir. Eğer sonuç 1'den büyükse, X ve Y pozitif yönde ilişkilidir. Bu demek oluyor ki, birinin oluşumu diğerinin oluşumunu etkilemektedir. Eğer sonuç 1'e eşitse, X ve Y birbirinden bağımsızdır ve aralarında herhangi bir korelasyon bulunmamaktadır şeklinde yorumlanmaktadır.

Denklem 2.10 denklem 2.9' un farklı bir gösterimidir. **Bir başka deyişle, olaylardan bir tanesinin meydana gelmesi, diğerinin oluşumunu kuvvetlendirerek, kaldıraç görevi görmektedir.**

$$\frac{P(Y|X)}{P(Y)} = \frac{conf(X \rightarrow Y)}{sup(Y)} \quad (2.10)$$

İkinci korelasyon ölçüsü, X^2 'dir. X^2 'yi kullanarak korelasyon hesaplamak için, gözlemlenen ve beklenen değerlere ihtiyacımız vardır. Bu değerleri bilebilmemiz için, bir olasılık tablosu içerisinde gözlemlenen ve beklenen değerlere yer verilmektedir.

$$X^2 = \sum \frac{(observed - expected)^2}{expected} \quad (2.11)$$

Bir diğer korelasyon ölçüsü tüm_güven'dir:

Verilen bir $X = \{i_1, i_2, \dots, i_k\}$ kümesi için, şu şekilde açıklanmaktadır:

$$tüm_güven = \frac{s(X)}{\max_item_s(X)} = \frac{s(X)}{\max\{s(i_j) | \forall i_j \in X\}} \quad (2.12)$$

2.12'deki eşitlikte, $\max\{s(i_j) | \forall i_j \in X\}$ X öge kümesindeki tüm öğeler içerisinde en yüksek (tekil) öge desteğini temsil etmektedir. X'in tüm_güven değeri $i_j \rightarrow X \rightarrow i_j$, $i_j \in X$ kuralları arasında minimum güven değerine sahiptir.

Bir başka korelasyon ölçüsü; X ve Y için cosine şu şekilde hesaplanmaktadır:

$$cosine(X, Y) = \frac{P(XUY)}{\sqrt{P(X) \times P(Y)}} = \frac{s(XUY)}{\sqrt{s(X) \times s(Y)}} \quad (2.13)$$

cosine ölçüsünün kaldıraç ölçüsü ile uyumlu olduğu görülmektedir: Her ikisinin formülü de cosine'in paydasında yer alan X ve Y olasılıklarının kök içerisinde bulunması dışında aynıdır. Tabii ki bu, sonuç açısından önemli bir farktır. cosine değeri yalnızca X, Y ve XUY 'nin destek değerlerinden etkilenmektedir. Toplam işlem sayısı bu değeri etkilememektedir.

Bir işlem eğer boş değerlerin etkisinden bağımsız ise, **boş-değişmez** (null invariant) olarak adlandırılmaktadır. Kaldıraç ve X^2 değerleri boş değerlerden etkilenmektedir. Tüm_güven ve cosine ölçüleri ise boş değerlerin etkisini hesaplanmaları gereği göz ardı etmektedirler. Büyük veritabanlarında birçok boş değer olacağından, tüm_güven ve cosine gibi bir boş-değişmez ölçünün bu gibi durumlarda daha belirleyici olduğu söylenebilmektedir (Han ve Kamber, 2006).

2.4.4. Kısıt Tabanlı Birliktelik Analizi (Constraint-Based Association Mining)

Bir veri madenciliği süreci, verilen veriler arasından binlerce kuralı ortaya çıkarmaktadır. Bu kurallardan birçoğu kullanıcıya anlamsız ya da ilgisiz gelecektir. Kullanıcıların hangi kurallarda daha ilgili olması gerektiği yönünde başvuracakları bazı kısıtlar bulunmaktadır. Bu stratejiye kısıt tabanlı madencilik denilmektedir. Bu kısıtlar şunları içerebilmektedir:

Bilgi türünde kısıtlar: Birliktelik ya da korelasyon gibi çıkarılacak bilginin türünü belirlemektedirler

Veri kısıtları: İlgili veri kümesini belirlemektedirler.

Boyut/seviye kısıtları: Verinin istenilen boyutunu çıkarım esnasında kullanılmak üzere belirlemektedirler.

İlginçlik kısıtları: Kural üzerindeki destek, güven gibi istatistiksel ölçülerin eşik değerlerini belirlemektedirler.

Kural kısıtları: Çıkarılacak kuralların yapısını belirlemektedirler (Nitelik değerleri, nitelikler arasındaki ilişkiler, kural içerisindeki öncül ve ardılların belirlenmesi gibi).

Yukarıdaki kısıtlar yüksek seviyede açıklayıcı veri madenciliği sorgu dili ve kullanıcı arayüzü kullanılarak özelleştirilebilmektedirler. Kısıt madenciliği, süreçleri daha interaktif ve belirleyici analizlere yönlendirmektedir. Kural kısıtları, kullanıcının ele almak istediği kuralları açıklamasına yardımcı olur, böylelikle veri madenciliği süreci daha etkili hale getirilmektedir

Meta Kurallar GÜdümlü Birliktelik Kuralları Madenciliği (Metarule guided mining of association rules)

Meta kurallar kullanıcılara, kuralların söz dizimsel formlarını belirleme imkanı verir. Kural yapısı, madencilik sürecinin etkinliğini arttırmak adına kısıt gibi kullanılabilir. Meta kurallar analistin tecrübesine, beklentilerine, öngörülerine bağlı olarak oluşturulabilir.

Örneğin metakural madenciliği ile ilgili HAN, kitabında şöyle bir örneğe yer vermiştir; *AllElectronics* mağazasında, piyasa analisti olarak çalışıyorsunuz, müşteri işlemlerinin yanı sıra, müşteriyi tanımlayan (yaşı, adresi, kredi notu) bilgilere de sahiptir. Müşterinin özellikleri ile, satın almaları arasındaki birlikteliği bulmaya çalışıyorsunuz. Bu ilişkiyi yansıtan tüm birliktelik kurallarını bulmak yerine, özellikle ofis yazılımlarının satışlarını arttıran müşterilerin özellikleri ile ilgilenebilirsiniz. Meta kurallar burada devreye girmektedir. İlgili olduğunuz kuralın yapısını belirlemede size yardımcı olmaktadır. Aşağıda bir meta kural örneği verilmiştir.

$$P_1(X,Y) \wedge P_2(X,W) \Rightarrow \text{satın alır } (X,\text{ofis yazılımı}) \quad (2.14)$$

Örnekte P_1 ve P_2 ile belirtilenler, verilen veritabanında, çıkarım süresince niteliklere dönüştürülerek somutlaştırılan eylem değişkenleridir. X müşteriyi temsil eden bir değişken, Y ve W , sırasıyla P_1 ve P_2 'ye atanan niteliklere göre değer almaktadır.

Genel olarak, bir meta kural kullanıcının aradığı ilişki hakkında hipotezler oluşturmaktadır. Veri madenciliği sistemi daha sonrasında, verilen meta (anahtar) kurallara uyan kurallar araştırmaktadır. Örneğin, kural 2.14, kural 2.15' de belirtilen meta kural ile örtüşmektedir.

$$\text{yaş}(X, "30...39") \wedge \text{gelir}(X, "41K....60K") \Rightarrow \text{satın alır}(X, "ofis yazılımı") \quad (2.15)$$

Meta kuralda belirtilen tahminleyici sayısına bakılarak, oluşturulacak kural hakkında öngöründe bulanabilir ve kural çıkarımı sırasında bu bilgiler kullanıcıya rehberlik edebilmektedir (Han ve Kamber, 2006).

Kural kısıtları sık öge madenciliğine göre beş kategoriye ayrılabilir: (1) tek düze olmayan (antimonotonic), (2) tek düze (monotonic), (3) kısa (succinct), (4) değiştirilebilir (convertible), (5) değiştirilemez (unconvertible). Sırasıyla açıklamak gerekirse, bir öge kümesi kurallar kısıtına uymuyor ise, üst kümelerinden hiç biri bu kısıta uymaz. Bir kural kısıtı bu özelliğe uyuyorsa, tek düze olmayan (antimonotonic) kurallardandır. Eğer verilen bir öge kümesi, minimum destek değerine uymuyorsa, hiç bir üst kümesi uymayacaktır. Bu özellik apriori algoritmasının her bir iterasyonunda, aday öge kümesinin sayısını azaltmak için kullanılmaktadır. Böylelikle birliktelik kuralı için arama alanı azalmaktadır (Han ve Lamber, 2006).

Kısıtlar için ikinci kategori tek düzeliktir. Eğer bir öge kümesi, kural kısıtına uyuyorsa, onun tüm üst kümeleri de bu kısıta uyacaktır. Eğer bir kural kısıtı bu özelliğe uyuyor ise, tek düzedir denilebilmektedir.

Üçüncü kategori kısa kısıtlar, bu kategorideki kısıtlar için, bir kez belirtilirler ve yalnızca bu belirtilenlerin verilen kısıtlara uyduğu garanti edilebilir. Bu durum, azımsanmayacak ölçüde yarat-test et paradigmasını önlemektedir.

Dördüncü kategori değiştirilebilir kısıtlardır. Bazı kısıtlar yukarıda saydığımız kısıtlardan hiç birisine ait değildir. Buna rağmen, öge kümesi içerisinde yer alan öğeler belirli bir sırada düzenlenirse, sık öge madenciliği göz önünde bulundurularak, tek düze ya da tek düze olmayan kısıtlara çevrilebilir (Han ve Kamber, 2006).

Bazı kısıtlar ise beşinci kategoride yer alan değiştirilemez kısıtlara aittir.

2.4.5. Kantitatif Birliktelik Kurallarının Çıkarılması

Kantitatif birliktelik kuralları sayısal niteliklerin, güvenilirliği maksimize etmek gibi bazı çıkarım kriterlerini yerine getirmek için dinamik olarak ayrıklaştırıldığı, çok boyutlu birliktelik kurallarıdır. Kuralın sol tarafında iki kantitatif nitelik, sağ tarafında bir kategorik nitelik bulunan kurallara iki boyutlu (2-D) kantitatif birliktelik kuralları denilmektedir. Çünkü toplamda sahip olduğu kantitatif nitelik sayısı ikidir.

$$A_{quan1} \wedge A_{quan2} \Rightarrow A_{cat}$$

Yukarıdaki kuralda A_{quan1} ve A_{quan2} kantitatif nitelikleri, A_{cat} bir kategorik niteliği temsil etmektedir.

Bu tür kuralların bulunabilmesi için geliştirilmiş bir yöntem ARCS (Association Rule Clustering System) (Birliktelik Kuralları Kümeleme Sistemi -BKKS) olarak adlandırılmaktadır. Mevcut yöntemde, görüntü işlemeden esinlenilmektedir. Temel olarak bu yaklaşımda, kantitatif nitelik çiftlerini kategorik nitelik koşullarına uyan bir koşula göre iki boyutlu bir şebeke (grid) üzerine eşleştirmektedir. Bu şebeke daha sonra hangi birliktelik kurallarından, hangi nokta kümeleri için yaratıldığını bulmak için taranmaktadır. BKKS' de yer alan adımlara aşağıda sırasıyla yer verilmiştir:

Binning (Gruplama): Kantitatif nitelikler, tanım kümelerini tanımlayan çok geniş bir aralıkta yer alan değerlere sahip olabilir. Üç temel gruplama stratejisine yer verilmektedir. Bunlar aşağıdaki gibidir:

- **Eşit aralıklarda gruplama:** Her gruba ait aralığın genişliği eşittir.
- **Eşit frekanslı gruplama:** Her grup yaklaşık olarak aynı sayıda veri grubuna (tuples) sahiptir.
- **Kümeleme tabanlı gruplama:** Komşu noktaları aynı bölmede gruplamak için kantitatif nitelik üzerinde kümeleme yapılır.

BKKS eşit aralıklarla gruplamayı kullanmaktadır. Burada her bir kantitatif nitelik için grup genişliği, kullanıcı tarafından belirlenmektedir. Her iki kantitatif niteliği de içeren her bir olası grup (bin) kombinasyonu için iki boyutlu bir dizi yaratılmaktadır (Han ve Kanber, 2006).

Sık öge kümelerinin bulunması: İki boyutlu dizi yaratıldığında, sık öge kümelerinin bulunması için taranabilmektedir (öyle kümeler ki minimum destek değerini sağlayan). Aynı zamanda minimum güven değerini de tatmin etmektedirler. Daha sonra bu eylem kümelerinden, güçlü birliktelik kuralları yaratılabilmektedir.

Birliktelik kurallarının kümelenmesi: Güçlü birliktelik kuralları bir önceki adımdan elde edilen birliktelik kuralları iki boyutlu şebekeye eşlenmektedir. Şekil 2.11' de verilen yaş ve gelir kantitatif nitelikleri göz önünde bulundurularak, kuralın sağ tarafında

satınalma (X , "HDTV") koşulunu öngören iki boyutlu birliktelik kurallarının, iki boyutlu şebekede gösterimine yer verilmiştir. Dört X de kurallara uymaktadır.

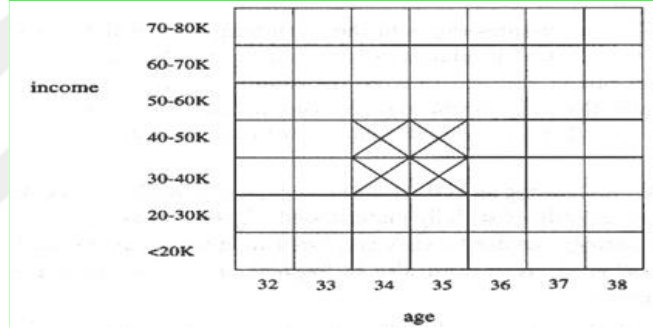
$$yaş(X,34) \wedge gelir(X, "31K....40K") \Rightarrow satın alır(X, "HDTV")$$

$$yaş(X,35) \wedge gelir(X, "31K....40K") \Rightarrow satın alır(X, "HDTV")$$

$$yaş(X,34) \wedge gelir(X, "41K....50K") \Rightarrow satın alır(X, "HDTV")$$

$$yaş(X,35) \wedge gelir(X, "41K....50K") \Rightarrow satın alır(X, "HDTV")$$

Birbirine çok yakın olan dört kuralı tek bir kuralda toplamak mümkün müdür diye bakıldığında, aslında bu dört kuralın aşağıdaki basit kuralı oluşturmak üzere kümelenebileceği görülmektedir.



Şekil 2.11: İki Boyutlu Şebeke Gösterimi(Han ve Kamber, 2006).

$$yaş(X, "34..35") \wedge gelir(X, "31K....50K") \Rightarrow satın alır(X, "HDTV")$$

BKKS, bu amaçla bir kümeleme algoritması kullanmaktadır. Algoritma, şebekeyi taramakta, kuralların dikdörtgen kümelerini aramaktadır. Bu şekilde, bir kural kümesinde yer alan kantitatif niteliklerin grupları, ileride birleştirilebilmektedir. Böylelikle kantitatif niteliklerin dinamik ayrıklaştırılması meydana gelmektedir (Han ve Kamber, 2006).

2.5. MARKET SEPET ANALİZİ

Sık öge madenciliği, büyük işlemsel veya ilişkisel veri kümelerindeki öğeler arasındaki birlikteliklerin ve ilişkilerin keşfedilmesine yol açmaktadır. Büyük miktarda veri sürekli

toplanıp depolanırken, pek çok endüstri veritabanlarındaki saklı kalmış örüntüleri açığa çıkarmak istemektedir. Büyük miktardaki işlem kayıtları arasındaki ilginç ilişkilerin keşfi; katalog dizaynı, çapraz satış ve müşterilerin alışveriş esnasındaki davranışlarının analizi gibi birçok karar verme sürecinde karar vericilere yardımcı olabilmektedir (Han, 2006).

Bir alışveriş sırasında veya birbirini izleyen alışverişlerde müşterinin hangi mal veya hizmetleri satın almaya eğilimli olduğunun belirlenmesi, müşteriye daha fazla ürünün satılmasını sağlama yollarından birisidir (Karabatak ve İnce, 2004).

Satın alma eğilimlerinin tanımlanmasını sağlayan birliktelik kuralları, en yaygın olarak pazar sepet analizi adı altında veri madenciliğinde yaygın olarak kullanılmaktadır (Birant ve diğ., 2010). Pazar sepet analizi, bir alışveriş deneyimi sırasında müşteri tarafından satın alınan ürünlerin birleşimini incelemektedir.

Bu modeller, pazar sepeti analizi için ve birlikte satılabilen ürün veya hizmet paketleri için kullanılabilir (Tsipstis ve Chorianopoulos, 2009). Bu süreç, müşterilerin "alışveriş sepetlerine" yerleştirdikleri farklı öğeler arasında ilişkiler bularak müşteri satın alma alışkanlıklarını analiz etmektedir. Bu tür birlikteliklerin keşfi perakendecilere hangi öğelerin müşteriler tarafından sıklıkla birlikte satın alındığına dair fikir edindirerek pazarlama stratejileri geliştirmelerine yardımcı olabilmektedir. Sepet analizi, farklı bakış açılarındaki müşteri dağılımını da göstermektedir. Bu dağılım bilgileri planlama, reklam dizaynı, indirim-promosyon, mağaza düzeni ve ürün yatırımı şeklindeki kararları alma aşamasında yardımcı olmaktadır (Yang ve Lai, 2006).

Örneğin müşteri süt satın alıyorsa, süpermarkette aynı alışveriş deneyimi sırasında eklemek almaya ne kadar ihtimal verilmektedir? Bu bilgiler, perakendecilere seçici pazarlama yapmalarına ve raf alanlarını planlamalarına yardım ederek satışların artmasına neden olabilmektedir (Han, 2006). Bu tür birliktelik-ilişki örüntüleri ancak, örüntü de yer alan öğelerin birden fazla harekette tekrarlandığında potansiyel olarak mevcut olabilmektedirler (Özçakır, 2006). Gerçek hayattaki uygulamalarda veri madenciliği teknikleri milyonlarca veri üzerinde uygulandığından birliktelik-ilişki sorgusu sırasında kullanılan algoritmalar hızlı olmalıdırlar (Agrawal ve Srikant, 1996). Market sepet analizleri sadece satın alma işlemlerine dayanarak müşteri davranışları hakkında bilgi

edinmeyi sağlamaktadır. Bu bilgiler müşteri özelliklerini içermemektedir ancak farklı bakış açılarındaki müşteri dağılımı hakkında bilgi vermektedir.

Ay ve Çil (2010), bir market raf düzenlemesine yönelik yaptıkları çalışmada, son yıllarda birliktelik yöntemlerine yönelik yapılan literatür çalışmalarını şu şekilde özetlemiştir: Nahar ve diğ. (2014) apriori algoritmasını kullanarak erkekler ve kadınlarda kalp hastalığı olanlarda görülen ortak faktörleri tespit etmeye çalışmış, Palaniappan ve Awang (2008), birliktelik ve sınıflandırma yöntemlerini bir arada kullanarak kalp hastalığı riskinin önceden tespiti üzerinde çalışmış, Liao ve diğ.(2010), apriori algoritmasını kullanarak yeni ürün geliştirme, etkili promosyon çalışmaları ve satış stratejileri geliştirerek müşteri ilişkileri yönetiminin iyileştirilmesi üzerine çalışmış, Wen ve diğ.(2012) Tayvan’da lüks marka tüketim alışkanlıklarını çözümlenmek adına, lüks tüketim ürünleri ile bu ürünleri satın alan müşteriler arasındaki ilişkiyi irdelemek için birliktelik kuralları yaklaşımı ve kümeleme analizini kullanılmışlardır. Chen ve Lin (2007) ise Apriori algoritmasını kullanarak raf düzeni yönetimi üzerine çalışmıştır. Bunun yanı sıra yapılmış olan çalışmalar ile bankalarda kredi risk değerlendirme aşamalarında, özellikle web üzerinden alışverişlerde sahtekârlılıkların tespitinde, tıp dünyasında hastalık belirtilerinin tespitinde, satın almada müşterilerin sıralı satın alma işlemlerinin (araba satın aldıktan sonra arabanızı sigortalatmanız gibi) tespiti gibi birçok alanda örnek çalışmalara rastlamak mümkündür.

2.6. BİRLİKTELİK KURALLARININ BELİRLENMESİNDE KULLANILAN TEMEL ALGORİTMALAR

Birliktelik kurallarının elde edilmesi sırasında başvurulabilecek olan bir çok algoritma mevcuttur. Bunlardan bazıları aşağıda ele alınmıştır:

2.6.1. Apriori Algoritması

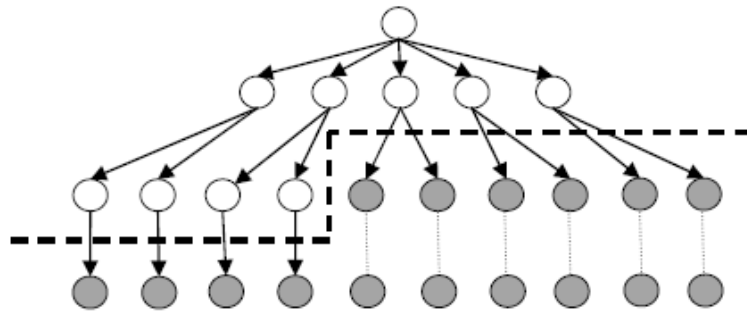
Apriori algoritması , *Agrawal ve Srikant* tarafından 1994 yılında geliştirilmiştir. Algoritmanın ismi, yaygın nesnelere önsel bilgilerini kullanmasından yani bilgileri bir önceki adımdan almasından “önceki (prior)” kelimesinden gelmektedir. Bu algoritmada sık geçen öğe kümelerini bulmak için birçok kez veritabanını taramak gerekmektedir. (Karabatak ve İnce, 2004).

Algoritmanın en bilinen özelliği bir öge kümesi sık geçen ise, bu öge kümesine ait tüm alt kümelerde sık geçen öge kümesi olmalıdır şeklindedir. İteratif bir özelliğe sahip olan Apriori, veri tabanlarında sık geçen öge kümelerinin ortaya çıkarılmasında kullanılmaktadır (Han ve Kamber, 2006).

Apriori algoritmasının sık öge yaratma aşamasında, iki önemli özelliği bulunmaktadır. Birincisi, seviye mantığı ile işleyen bir algoritma olmasıdır; başka bir deyişle, öge kafesindeki bir seviyeyi bir seferde gezinir, İkinci olarak, sık öge kümelerini bulma sırasında yarat-ve-test et stratejisine başvurmaktadır. Her iterasyon seviyesinde, bir önceki iterasyonda bulunmuş olan yeni aday kümeler yaratılmaktadır. Her aday için destek hesaplanır ve belirlenen minimum eşik değeri ile karşılaştırılmaktadır. Algoritma için gerekli olan toplam iterasyon sayısı; maksimum sık öge kümesi k_{max} olduğu durumlarda $k_{max}+1$ ' dir.

Apriori algoritması genişlik öncelikli (Breadth first search) algoritmalara örnektir. Bu yaklaşım, ilk olarak sık geçen 1 elemanlı öge kümeleri bulmakta, sonrasında 2 elemanlı öge kümeleri ve bu şekilde oluşturulacak yeni sık geçen öge küme kalmayana kadar devam etmektedir (Tan ve diğ., 2005).

Şekil 2.12' de genişlik öncelikli yaklaşıma yer verilmiştir



Şekil 2.12: Genişlik Öncelikli Arama (Tan Ve Kumar, 2005).

Algoritmada geçen sık öge kümeleri L karakteri ile aday öge kümeleri ise C (candidate) karakteri ile temsil edilmektedir. L_k : k adet elemana sahip sık öge kümesi, C_k : k adet elemana sahip aday öge kümesinin gösterimi için kullanılmaktadır.

Bu yaklaşımda k adet elemana sahip öge kümeleri (k-1) adet elemana sahip öge kümelerinin birleştirilmesiyle oluşturulmaktadır. Öncelikle tek öge içeren öge kümeleri oluşturmak için her bir ögenin destek değeri hesaplanmaktadır, (k-1)'inci geçişte elde edilen sık ögeler (L_{k-1}), k'nıncı geçişte sık geçen öge aday kümesini (C_k) oluşturmak için kullanılmaktadır. Veritabanı taranır ve C_k 'ların destek değerleri hesaplanır ve L_k 'lar oluşturulmaktadır. Sık geçen öge küme bulamayınca kadar algoritma veritabanını taramaya devam etmektedir.

2.6.1.1. Apriori Algoritmasının Adımları

Apriori özelliğinin algoritmada nasıl kullanıldığını anlamak için, L_k 'yi elde etmek için L_{k-1} 'in nasıl kullanıldığına bakmak gerekmektedir. Birleştirme (Join) ve Budama (Prune) işlemlerini kapsayan adımlar aşağıda anlatıldığı gibidir (Han ve Kamber, 2006).

1. **Birleştirme Adımı:** L_k 'nın bulunması için k adet elemana sahip aday öge kümeleri (C_k), L_{k-1} 'in kendisi ile birleştirilmesinden oluşturulmaktadır. I_1 ve I_2 , L_{k-1} 'de bulunan öge kümeler olduğunu varsayalım. $I_i[j]$ gösterimi, I_i 'deki j. ögeyi temsil etmektedir. Apriori, bir işlemdeki ögelerin veya öge kümelerinin alfabetik sıraya göre sıralandığını varsaymaktadır. k-1 ögeli öge kümeler için ögeler $I_i[1] < I_i[2] < \dots < I_i[k-1]$ şeklinde sıralanmaktadır. L_{k-1} 'in L_k (k-2) ögeleri ortak ise $L_{k-1} \in L_k$ birleşmesi gerçekleştirilebilmektedir. Bu demek oluyor ki, L_{k-1} 'in I_1 ve I_2 ögeleri, $(I_1[1] = I_2[1]) (I_1[2] = I_2[2]) \wedge \dots \wedge (I_1[k-2] = I_2[k-2]) \wedge (I_1[k-1] < I_2[k-1])$ şartını sağlıyor ise birleştirilmektedir. $I_1[k-1] < I_2[k-1]$ şartı, yinelemelerin gerçekleşmesini önlemektedir.

2. **Budama Adımı:**

L_k 'nın üst kümesi olan C_k 'nın elemanları yaygın olabilir veya olmayabilir fakat tüm k- elemanlı sık öge kümeleri C_k içerisinde yer almaktadır. C_k 'daki her bir adayın destek değerinin belirlenmesi için yapılan veritabanının taranma sonuçları L_k 'yi tespit etmektedir (minimum destek değerinden büyük veya eşit değere sahip bütün adaylar sık geçen öge kümesi olarak tanımlanmaktadır ve L_k 'da yer almaktadırlar). C_k çok büyük boyutta olabilir ve çok ağır hesaplamalar gerektirebilmektedir. C_k 'nın boyutunu azaltmak için, Apriori özelliği

kullanılmaktadır. Herhangi sık olmayan $(k-1)$ öge kümesi sık geçen k -öge kümesinin alt kümesi olamamaktadır. Bu nedenle, herhangi aday k -öge kümenin bütün $(k-1)$ ögeli altkümeleri L_{k-1} 'de yer almıyorsa, bu aday sık geçen öge küme olamamaktadır ve C_k ' dan çıkartılmaktadır (Han ve Kamber, 2006).

Tablo 2.5' de apriori algoritmasının işleyişini anlatan kodlara yer verilmiştir. İlk adımda sık geçen öge kümelerin (L_k) bulunması, ikinci adımda sık geçen öge kümelerin aday küme olup olamayacağını tespiti ve eğer aday küme olamayacaksa, bu kümelerin C_k ' dan çıkarılması, son olarak da C_k aday kümeleri içinde gereksiz tekrarlardan kurtulmak adına, apriorinin tekrarlayan (prior) özelliğini kullanarak budama işlemine yer verildiği görülmektedir. Tablo 2.'de apriori kodu için girdi ve çıktılar aşağıdaki gibi belirtilmiştir.

Algoritma: Apriori

Girdi: İşlemsel D veritabanı, minimum destek eşik değeri

Çıktı: L, D içerisindeki sık öge kümeleri

Tablo 2.5: Apriori Algoritmasının Kodları (Han ve Kamber, 2006).

```

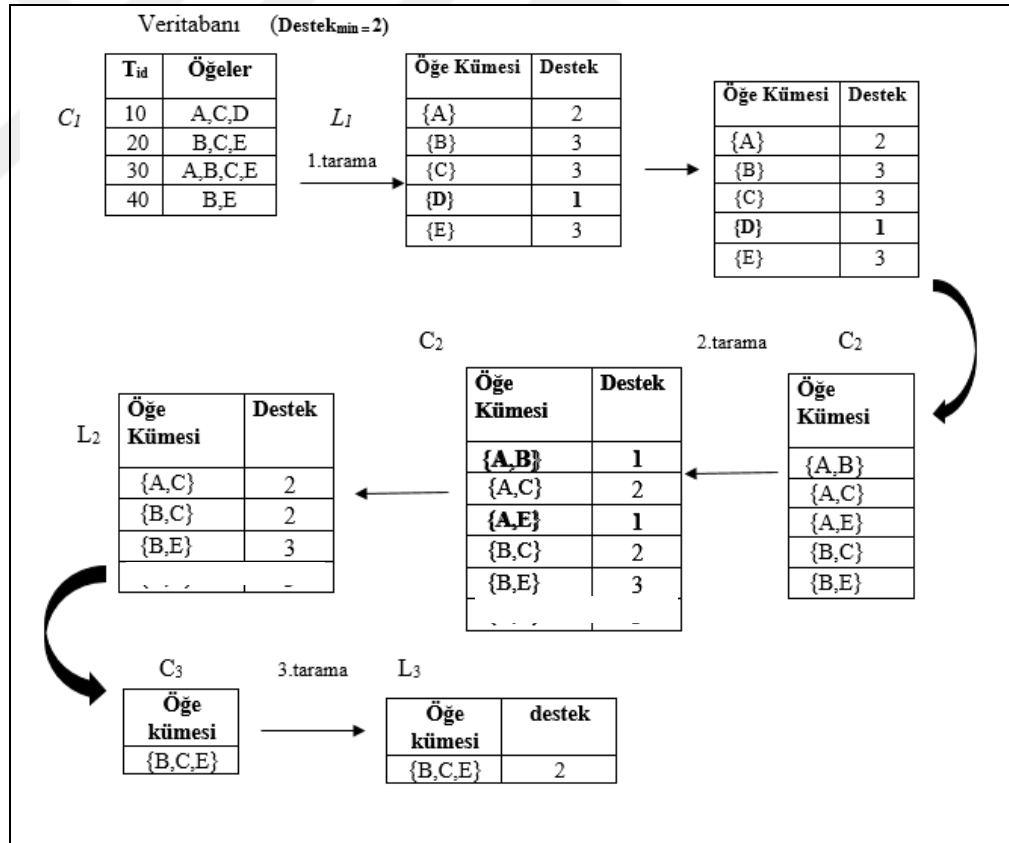
Metod:
1)  $L_1 = \text{bul} \{ \text{large 1-nesnekümeler} \};$ 
2) for {  $k=2; L_{k-1} \neq \emptyset; k++$  } do begin
3)    $C_k = \text{apriori-gen} (L_{k-1});$  // yeni adaylar
4)   forall işlemler  $t \in D$  do begin // D destek sayıları için taranır
5)      $C_t = \text{alt küme} (C_k, t);$  // Aday olan t' lerin alt kümeleri elde edilir
6)     forall adaylar  $c \in C_t$  do
7)        $c.\text{sayısı} ++;$ 
8)     end
9)    $L_k = \{ c \in C_k \mid c.\text{sayısı} \geq \text{min}_s \}$ 
10) end
11) return  $U_k L_k;$ 
Apriori-gen prosedürü ( $L_{k-1}$ : sık geçen (k-1)-nesnekümeler)
1) forall nesneküme  $l_1 \in L_{k-1}$ 
2)   forall nesneküme  $l_2 \in L_{k-1}$ 
3)     if  $(l_1[1] = l_2[2]) \wedge (l_1[2] = l_2[2]) \wedge \dots \wedge (l_1[k-2] = l_2[k-2]) \wedge (l_1[k-1] < l_2[k-1])$ 
4)       then {
5)          $c = l_1 \cup l_2;$  // birleştirme adımı: aday oluşturma
6)         if sık olmayan alt kümeler( $c, L_{k-1}$ ) varsa then
7)           sil  $c;$  // budama adımı: gereksiz adaylar çıkarılır
8)         else  $c'$  yi  $C_k'$  ya ekle;
9)       }
10)    return  $C_k'$ ;
Sık olmayan alt kümeler prosedürü ( $c$ : aday k-nesneküme;
 $L_{k-1}$ : sık geçen (k-1)-nesnekümeler); // prior bilgisini kullan
1) forall (k-1)-alt küme  $c'$  nin  $s'$  i
2)   if  $s \in L_{k-1}$  then
3)     return TRUE;
4)   return FALSE;

```

Tablo 2.5' de kodları verilen algoritma adımları aşağıda maddeler halinde anlatılmış ve Şekil 2.13'de bu adımlar özetlenmiştir.

- 1) Algoritmanın ilk iterasyonunda veri setinde yer alan her bir öge, C_1 aday kümesinin elemanıdır. Her bir ürünün destek değerini hesaplamak için veri setinde yer alan işlemler taranır.
- 2) Örnekte, destek eşik değeri 2 olarak belirlenmiştir. C_1 kümesindeki adayların tek tek destek değerlerine bakılır. 1 elemanlı öge kümeler ile sık geçen L_1 kümesi oluşturulur. Destek değeri 1'in altında olan adaylar elenir. Bu örnek için C_1 aday kümesinde yer alan D ögesi 2 olan destek değerini sağlayamadığı için kümeden çıkartılmıştır.

- 3) 2 öğeli sık geçen öge kümeleri (L_2) algoritma $L_1 \in L_1$ birleşmesini kullanmaktadır ve L_1 'in ikili kombinasyonlarından 2 elemanlı öge kümeleri (C_2) oluşturur.
- 4) Veritabanındaki işlemler taranarak, C_2 kümesindeki öğelerin destek değerleri tespit edilir.
- 5) C_2 kümesindeki minimum destek değerine sahip olan 2 elemanlı öge kümeler L_2 'yi oluşturur.
- 6) 3 öğeli adaylar kümesini (C_3) oluşturmak için $L_2 \in L_2$ birleşimi kullanılır. Birleşme işlemi sonrası budama işlemi de bu aşamada gerçekleştirilmiştir.
- 7) Veritabanı tabanı taranır ve C_3 kümesindeki öğelerin destek değerleri belirlenir.
- 8) (C_4) oluşturmak için $L_3 \in L_3$ birleşimi kullanılmaktadır. L_3 kümesinde tek bir öge küme bulunduğundan dolayı C_4 kümesinde hiç eleman yer almayacaktır. Algoritma tüm sık geçen öğeleri bulduğundan dolayı burada sonlandırılmaktadır (Han ve Kamber, 2006).



Şekil 2.13: Apriori Algoritmasının Adımları (Tan ve diğ., 2005).

2.6.1.2. *Apriori Algoritmasının Kısıtları*

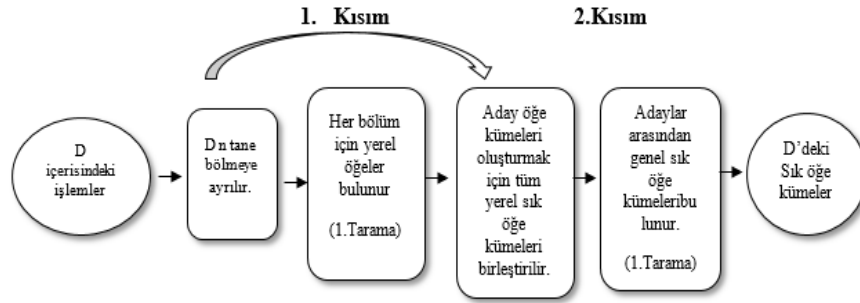
Apriori ile ilgili iki temel sınırlama bulunmaktadır: Birincisi geniş bellek ve muazzam yürütme süresini tüketen karmaşık aday öge kümesi oluşturma sürecidir. İkincisi; aday oluşturma sırasında meydana gelen veritabanı taramalarıdır. Bu noktada, n en uzun örüntü olmak üzere, $(n+1)$ taramaya gereksinim duyulmaktadır. Genellikle bu kısıtlamaların üstesinden gelmenin iki yolu vardır: Birinci yol, aday öge kümesini daha küçük hale getirmek için farklı budama ve filtreleme teknikleri keşfetmektir. İkinci yaklaşım, orijinal veritabanını geniş sık öge kümesine dayalı işlem alt kümesiyle değiştirmek veya veritabanındaki tarama sayısını en aza indirmektir (Nasreen ve diğ., 2014).

Apriori Algoritmasının Verimliliğini Arttırma: Apriori algoritması, birliktelik kural madenciliğinin klasik bir algoritması olup, sık öge kümelerinin yaratılmasında yaygın olarak kullanılmaktadır. Bu klasik algoritma, veritabanını çok fazla taramasından dolayı verimsiz durumlarla karşılaşılabilir. Eğer veritabanı da çok büyükse, tarama işlemi çok daha fazla zaman alacaktır (Vijayalakshmi ve Pethalakshmi, 2014). Algoritmanın verimliliğini arttırma konusunda çeşitli tekniklere rastlanmaktadır. Bu teknikler aşağıdaki başlıklar altında özetlemiştir (Han ve Kamber, 2006).

Karma Tabanlı Teknik (Hash Based Technique): Karma tabanlı apriori uygulaması, doğrudan bir karma tabloyu temsil eden veri yapısı kullanmaktadır. Bu teknik, aday k -öge kümelerinin sayısını azaltarak, apriori algoritmasının bazı zayıflıklarının üstesinden gelmeyi önermektedir. (Park ve diğ., 1995). Bu yöntem ile, L_2 'yi belirlemek için gereken taramanın daha verimli olması için, üretilen C_2 'deki öge sayısının azaltılabileceği savunulmaktadır (özellikle $k=2$ olduğu durumlarda etkilidir) (Hash ve diğ., 2011).

İşlemlerin Azaltılması (gelecek iterasyonlarda işlem sayısını azaltmak): Herhangi bir sık k -öge seti içermeyen işlem, herhangi bir sık $(k+1)$ öge seti de içeremez. Bu nedenle, $j > k$ olan j -öge kümeleri veritabanı taramalarını içermeyeceğinden dolayı böyle bir işlem işaretlenebilir ya da daha sonraki incelemelerden kaldırılabilir (Han ve Kamber, 2006). Gereksiz işlem kayıtlarını silerek, tarama süresini azalttığı gibi, aday öge kümelerinin budanması sırasında, gereksiz alt-öğelerin yaratılmasını da önlemektedir (Sing ve Sodhi, 2013).

Bölümleme (Aday Öğe Kümelerini Bulmak İçin Veriyi Bölümleme): Bölümleme algoritmaları işlemsel veri seti D 'yi n tane çakışmayan bölümlere ayırmaktadır. Bu algoritma veri setinin tarama sayısını ikiye düşürmektedir. İlk tarama sırasında, algoritma her bölümdeki tüm öğe setlerini bulur. Bu yerel öğe setleri (local itemset), genel aday öğe setlerinde (global itemset) toplanır. İkinci tarama sırasında bu genel öğe setleri, tüm veri setinde var olup olmadıklarını belirlemek için sayılmaktadır. Bölümleme algoritmaları sık öğe setleri bulma performansını geliştirir ve aynı zamanda birçok avantaj sağlamaktadır. Küçük bölümler büyük bellek yerine ana bellek içine sığabilmektedir. Her bölümün boyutu küçük olduğundan, algoritmalar aday madde kümelerinin boyutunu azaltabilmektedir. Buna ek olarak, bölümleme algoritmaları veri setinin boyutunu azaltır ve veri kümesinde yalnızca iki tarama gerektirmektedir (Gupta ve Satsangi, 2012).



Şekil 2.14: Verilere Bölümlere Ayırarak Madencilik (Han ve Kamber, 2006).

Örnekleme (Verilen veri setinin alt kümesi üzerinde madencilik): Örnekleme yaklaşımının temel fikri, verilen bir D verisi üzerinden S örneği seçerek, sık öğe kümelerini D yerine S üzerinde aramaktır. Örnekleme yaklaşımı, sıklıkla çalıştırılması gereken hesaplamaya dayalı uygulamalar gibi verimliliğin çok önemli olduğu durumlarda tercih edilmektedir.

Dinamik öğe kümesi sayımı (Bir tarama sırasında farklı noktalarda aday öğeler ekleme): Bu tekniğin veritabanının başlangıç noktaları ile işaretlenmiş bloklara bölündüğü veri tabanlarında kullanımı önerilmektedir. Apriori'de tüm veritabanı tarandıktan sonra belirlenen aday öğe kümeleri, bu varyasyonda, herhangi bir başlangıç noktasına

eklenebilmektedir. Böylelikle daha az veritabanı taraması yapılmaktadır (Brin ve diğ., 1997).

2.6.2. Eclat Algoritması

ZAKI tarafından 1997 yılında geliştirilen algoritma sık öge madenciliğinde özellikle market sepet analizleri için geliştirilmiş bir veri madenciliği yöntemidir (Ma ve diğ., 2016). Sık öge madenciliği, süper marketlerde, online alışveriş mağazalarında, e-posta ile sipariş yapan mağazaların müşterilerinin alışverişlerindeki düzeni bulmayı amaçlamaktadır. Özellikle birlikte ve sıklıkla satın alınan ürünleri belirlemeye çalışmaktadır.

Bir kere belirlenen bu tür ilişkili ürün grupları, bir süpermarketin raf diziliminde veya bir mağazanın e-posta-sipariş kataloğunda ya da web mağazasının sayfalarında sunulan ürünlerin organizasyonunu optimize etmek için kullanılabilir. Ayrıca bu yöntem, müşterilere diğer ürün seçeneklerini önermek için de kullanılabilir (Borgelt, 2003).

Bu algoritma aşağıdaki aşamalara sahiptir:

- a) Destek ve güven ölçülerini karşılaştırmak üzere eşik değerler belirlenir. Uygulamadan elde edilen sonuçların bu eşik değerlere eşit ya da büyük olması beklenmektedir
- b) Gözlemler tekli olarak taranarak her bir ürünün ait olduğu örnek numaraları belirlenir.
- c) Yukarıdaki adımda seçilen ürünler ikişerli gruplandırılarak, bu grupların örnek numaraları belirlenir.
- d) Bu kez üçerli, dörderli vb. gruplandırmalar yapılarak işlemler tekrarlanır.
- e) Ürün grubu belirlendikten sonra kural destek ölçüsüne bakılarak birliktelikler belirlenir (Özkan ve Selçukcan Erol, 2015).

Eclat algoritması sık öğeleri yalnızca bir kez yaratmaktadır. Apriori algoritmasında olduğu gibi Eclat da aday (k+1) aday öge kümeleri yaratmak için birleştirme işlemi gerçekleştirmektedir (Ma ve diğ., 2016). Bu aşamada sıklıkla tercih edilen Apriori algoritmasının sık öğeleri bulma sırasında veritabanını tekrar tekrar taraması daha fazla zaman harcamasına neden olmaktadır.

Eclat algoritmasında, aday öge oluşturma eşit sınıflar arasında yapılmaktadır. Böylelikle eclat arama uzayını sayısız alt uzaylara ayırmaktadır. Denklik sınıfları (equivalence) teknolojisi, aday öge kümesi yaratma verimliliğini arttırmakta ve bellekte işgal edilen yeri azaltmaktadır (Ma ve diğ., 2016). Apriori algoritmasında, destek ve güven değerleri hesaplanmakta iken Eclat' da yalnızca destek değeri hesaplanmaktadır.

Apriori ile Eclat algoritmalarının bilinen en temel farkı, önek ağacını (prefix tree) nasıl gezindikleri ve bir öge setinin destek değerini nasıl belirledikleridir. Apriori önek ağacı genişlik öncelikli gezinmektedir. Sırayla önce 1 boyutlu öge kümeleri, daha sonra 2 ve daha büyük boyutlu öge kümeleri gezinir. Diğer taraftan Eclat ise önsel ağacı derinlik öncelikli gezinmektedir (Borgelt, 2003a).

Apriori de yatay (horizontal) veritabanı kullanılırken, Eclat da dikey (vertical) veritabanı kullanılmaktadır (Zaki ve diğ., 1997). Böylelikle veritabanını yalnızca bir kez taraması yeterli olmaktadır (Garg ve Kumar, 2013). Her iki algoritmada Rstudio programında arules paketinden faydalanmaktadır. Tüm bunlardan yola çıkarak, Eclat algoritması Apriorinin bazı sınırlamalarını kaldırmak için geliştirilmiştir denilebilmektedir (Kaur ve Grag, 2014). Eclat genellikle ziyaret ettiği tüm öge kümelerine ait destekleri saklamadığı için Apriori' ye göre daha fazla aday küme üretmektedir (Hashler ve diğ., 2014).

2.6.3. Fp-Growth Algoritması

Apriori algoritmasında gerçekleştirilen aday küme oluşturulması, çok fazla sayıda örüntü olduğu takdirde maliyetli bir işlem haline gelmektedir (Han ve diğ., 2004). Sık öge kümelerinin bulunması için veri madenciliğinde başvurulan bir başka algoritma FP-Growth algoritmasıdır. Sık öge kümesi madenciliğinde başvurulan en hızlı yaklaşımlardan birisi olarak nitelendirilmektedir (Borgelt, 2005). 2000 yılında aday küme yaratmaksızın birliktelik kurallarının çıkarımı adına Han tarafından ortaya sürülmüştür.

FP-Growth metodu genişlik öncelikli yaklaşıma bir örnektir FP-Growth, aday oluşturma sürecinde fazla zaman kaybetmeden sık geçen öge kümeleri oluşturmaktadır. Esasen bu algoritma, arama maliyetlerini düşürmektedir. Bu algoritmanın öne çıkan özellikleri arasında; veritabanını sadece iki kez tarama ve sık ögeleri bulurken kullandığı sık örüntü ağacı bulunmaktadır. Veritabanı taramalarının ilki, tüm ögelerin destek değerlerini

hesaplamak, ikincisi ise ağaç yapısının oluşturulması içindir. Algoritma ilk olarak sık geçen öge kümeleri bir FP-ağacına sıkıştırır daha sonra şartlı veri tabanlarını oluşturarak her bir veri tabanını ayrı ayrı analiz etmektedir (Ay, 2009). Bu ağaç yapısı sayesinde daha verimli bir algoritmanın ortaya çıktığı savunulmaktadır (Sinha ve Ghosh, 2013).

Fp-Growth algoritması özellikle büyük veri kümelerinde, veritabanı taramalarını düşük seviyede tuttuğu, sistem kaynaklarını verimli bir şekilde kullandığı için tercih edilebilmektedir.

Algoritma böl ve yönet stratejisini kullanmaktadır. En önemli özelliği tüm veritabanını FP-Ağaç (Fp-Tree, Frequent Pattern Tree) adı verilen sıkıştırılmış bir ağaç veri yapısı şeklinde tutmasıdır. Fp-Growth algoritması yaygın ögeler kümesini bulurken aday ögeler kümesi kullanmaz, yaygın ögeler Fp-Ağaç yapısından bulunmaktadır (Gündüz, 2016).

2.6.4. NBMiner Algoritması

Sık öge setlerinin çıkarımı için, birlikteliği oluşturan destek, ögelerin birlikte bulunma sıklığı birlikteliklerinin başlıca göstergesi olarak kullanılmaktadır. Birlikteliklerin daha detaylı araştırılıp araştırılmayacağına karar vermek noktasında tek bir kullanıcı tarafından belirlenen destek eşik değeri kullanılmaktadır. Desteğin veri setinde daha az görülen ögelerle ilgili bilinen bazı problemleri vardır. Daha kısa öge setlerini kayırabilmekte ve bu da bazen yanıltıcı ilişkiler üretmesine sebep olabilmektedir.

Tek bir kullanıcı tarafından belirlenen minimum desteğin bir alternatifi olarak yeni bir model tabanlı frekans kısıtlaması geliştirilmiştir. Kısıtlama, işlem verilerini üreten işlem bilgisini, işlem verilerinin genelde çarpık madde frekans dağılımına izin veren basit bir stokastik karışım modeli (NB modeli) uygulayarak kullanmaktadır.

Öge kümelerinin grupları için yerel frekans eşikleri bulmak adına modelle birlikte kullanıcı tarafından tanımlanan hassas bir eşik değeri kullanılmaktadır. Kısıtlamaya dayanarak, NB sık öge setleri kavramı geliştirmiş ve veritabanındaki tüm NB sık öge setlerini bulmak için bir çıkarım algoritması uyarlanmıştır. Önerilen model aşağıdaki özelliklere sahiptir:

Kullanılan stokastik model, çarpık frekans dağılımlarına izin verdiği için nadir geçen ürünlerle olan sorunu azaltmaktadır. Modelin gerektirdiği destek, yerel olarak

belirlendiğinden, yapay birliktelikler üretmeden daha uzun ilişkilendirmeler üretebileceği savunulmaktadır. Hassas eşik parametresi, tahmin edilen hata oranı olarak yorumlanabilmektedir. Bu özellikte daha kolay ayarlanmasını sağlamaktadır. Ayrıca, bu parametrenin, destek değerine göre, veritabanı yapısına daha az bağımlı olduğu görülmektedir. Halka açık işlem veritabanlarıyla yapılan denemelerde, yeni kısıtlamanın tek bir minimum destek eşiğinin üzerinde iyileştirme sağladığını, hassas eşiğin çok daha güçlü olduğu ve kullanıcı tarafından daha kolay belirlenip, yorumlandığı savunulmaktadır (Hashler, 2006).

2.7. MÜŞTERİ İLİŞKİLERİ YÖNETİMİ (CRM) VE VERİ MADENCİLİĞİ

2.7.1. Müşteri İlişkileri Yönetimi

Müşteriler bir kuruluşun en önemli varlıklarıdır. Müşteri kurumdan memnun kaldığı sürece, firmaya sadık ve ilişkilerini geliştirmeye devam eden bir yolda ilerleyecektir. Kurumlar, sadık kalmaya ve kurumla olan ilişkilerini sıcak tutmaya çalışan memnun müşteri potansiyelini elde tutmaya çalışmaktadırlar. Bu nedenle kurumlar müşterileri tutundurabilmek için açık bir strateji planlamalı ve kullanmalıdırlar. CRM, sadık ve uzun süreli müşteri ilişkileri kurma, yönetme ve güçlendirme stratejisidir. Müşteri anlayışına dayalı, müşteri odaklı bir yaklaşımdır. Kapsamı gereği, farklılaşmış ihtiyaçları, tercihleri ve davranışların tanımlanması ve anlaşılması yoluyla müşterilerin ayrı varlıklar olarak "kişiselleştirilmiş" şekilde ele alınmasını gerektiğini esas almaktadır (Tsiptsis ve Chorianopoulos, 2009).

2.7.2. Veri Madenciliğinde CRM

Veri madenciliği, CRM stratejisi kurabilmek için hayati önem taşıyan, mevcut müşteri anlayışını oluşturabilmek için gerekmektedir. Şirketlerin müşteri odaklı olmalarına yardımcı olan bir yöntemdir (Rygielski ve diğ, 2002). Müşteri ilişkileri yönetiminde, veri madenciliğinin temel fikri şudur: “Geçmişteki veriler gelecekte faydalı olacak bilgileri içermektedir” Bu fikir işe yaramaktadır çünkü kurumsal verilerde yakalanan müşteri davranışları rastgele değil, müşterilerin farklı ihtiyaçları, tercihleri ve eğilimlerini yansıtmaktadır (Berry ve Linoff, 2004). Böylelikle şirketlerin, müşteriler ile kişiselleştirilmiş etkileşime geçilebilmesini sağlamaktadır. Bunun sonucunda; veri analizi ile gelen artan memnuniyet ve kârlı müşteri ilişkileri görülmektedir.

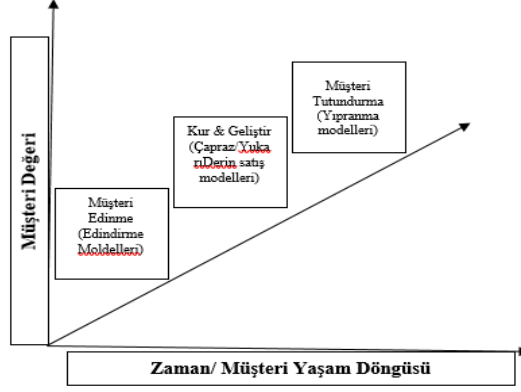
Veri madenciliği, daha fazla pazar payı edinme, müşterilerin elde edinilmesi, geliştirilmesi ve tutundurulmasına kadar, tüm bu aşamalarda yardımcı rol oynamaktadır. Daha spesifik olarak, veri madenciliği kullanımı ile desteklenebilir pazarlama faaliyetleri aşağıdaki başlıklar altında toplanmıştır.

2.7.2.1. Müşteri Segmentasyonu

Segmentasyon kendi özelliklerine göre farklılaştırılmış pazarlama stratejileri geliştirmek amacıyla farklı gruplar halinde müşteri tabanının ayrılma işlemidir. Segmentasyon için kullanılan spesifik kriterler ya da özelliklere göre farklı segmentasyon türleri vardır.

Davranışsal segmentasyonda, müşteriler davranışlarına ve kullanım özelliklerine göre gruplandırılmaktadır. Davranışsal segmentler iş kurallarına göre oluşturulabilmektedir, bu yaklaşımının bazı dezavantajları bulunmaktadır. Sadece birkaç segmentasyon alanı içerebilir ve objektifliği sorgulanabilmektedir. Öte yandan veri madenciliği ile veri odaklı davranışsal segmentler oluşturabilmektedir (Rud, 2001). Bölüm 2.3.2' de bahsedilen kümeleme modelleri bu aşamada kullanılabilir. Kümeleme algoritmaları, davranışsal verileri analiz edebilmekte, müşterilerin doğal gruplaşmalarını belirleyebilmekte ve gözlenen veri örüntüleri üzerine kurulmuş bir çözüm önerilebilmektedir.

Bu segmentler, her müşterinin önemine göre, müşteriyi taşıma ve pazarlama girişimlerini öncelik vermek üzere gerekmektedir. Böylece belirlenen müşteri segmentlerine göre kampanyalar geliştirmek, farklı segmentlerdeki müşteri anlayışları doğrultusunda çalışmalar yapabilmek mümkün olmaktadır (Tsiptsis ve Chorianopoulos, 2009). Veri madenciliği, müşteri yaşam döngüsünde yer alan; edinmeden, tutundurmaya, müşterilerle güçlü ilişkiler kurmaya, kayıp müşterilerin kazanca dönüştürülmesine kadar her fazda bireyselleşmiş ve optimize edilmiş müşteri yönetimini desteklemektedir. Şekil 2.15 bu süreçte yer alan müşteri yaşam döngüsüne yer verilmiştir. Müşteri yaşam döngüsü, bir müşteri ile işletme arasındaki ilişkinin aşamalarını göstermektedir. Müşteri yaşam döngüsü CRM alanında veri madenciliği uygulamalarını gerçekleştirebilmek için kurumlara iyi bir çerçeve sağlamaktadır (Rygielsk ve diğ., 2002).



Şekil 2.15: CRM yaşam döngüsü (Tsiptsis ve Chorianopoulos, 2009).

2.7.2.2. Doğrudan Pazarlama Kampanyaları

Pazarlamacılar, dalgalanmaları önlemek ve müşteri kazanımını sağlamak için, müşterilerine mail, internet, e-mail, tele pazarlama ve diğer doğrudan kanallar ile ulaşmak için, doğrudan pazarlama kampanyalarını kullanmaktadırlar. Kazanım kampanyaları yeni ve potansiyel değerli müşterileri rekabetten uzak tutmayı hedeflemektedir. Çapraz/derin/yukarı satış kampanyaları mevcut müşterilerine aynı ya da alternatif ürünlerden çok, daha kârlı ek ürünler sunabilmek için gerçekleştirilmektedir. Tutundurma kampanyalarında ise, değerli müşterilerin kurumla olan ilişkilerinin sonlandırılmasının önlenmesi amaçlanmaktadır.

Bu kampanyalar rafine edilmediği zaman, potansiyel olarak ne kadar etkili olurlarsa olsunlar, devasa bir kaynak çöpüne yol açabilmektedirler. Veri madenciliği müşteri özelliklerini analiz edip, hedef müşterilerin profillerini tanımlayarak sınıflandırma modelleri ve hedef kitle pazarlama kampanyaları gelişimini desteklemektedir. Aşağıdaki sınıflandırma modelleri pazarlama kampanyalarını optimize etmek için kullanılmaktadır:

Kazandırma(Acquistion) Modelleri: Potansiyel karlı müşterileri tanımlamak için kullanılmaktadır.

Çapraz/Derin/Yukarı Satış Modelleri: Mevcut müşterilerin satın alma potansiyellerini ortaya çıkartmaktadır.

Gönüllü Yıpratma ya da Gönüllü Dalgalanma Modelleri Dalgalanma sinyallerini bularak, gönüllü olarak ayrılma olasılığı yüksek olan müşterileri tespit etmektedir).

Dođru inřa edildiđinde bu modeller; iletiřime geçmek için dođru müřterileri belirleyebilmekte, hedef müřterilere ulařmanızı sađlayan kampanya listelerinin oluřturulmasına yardımcı olabilmektedir (Ngai ve Chau, 2009).

2.7.2.3. Market Sepet ve Sıralı Analizler

Veri madenciliđi ve birliktelik modelleri, genellikle ilgili, birlikte satılan ürünleri tanımlamak için kullanılmaktadır. Bu modeller market sepet analizi ve birlikte satılan ürün ve hizmetleri ortaya çıkarmak için kullanılmaktadır.

Sıralı modeller, olayların/satın almaların sırasını göz önüne alarak, olayların sırasını belirleyebilmektedir. Bu kısım Bölüm 2.5' de detaylı olarak ele alınmıştır.

3. MALZEME VE YÖNTEM

Tezin bu bölümünde veri madenciliğinde birliktelik kuralı yöntemleri ve müşteri ilişkileri yönetimine ilişkin, elektrik sektöründe faaliyet gösteren bir firmaya ait satış verilerinden yola çıkılarak gerçekleştirilen uygulamaya yer verilmiştir. Uygulamada CRISP modeli adımları izlenmiş olup, modele ait tüm aşamalara sırasıyla yer verilmiştir.

3.1. PROBLEMİN TANIMLANMASI

Ticari firmaların temel amacı kâr elde etme amacı gütmeleridir. Bu amaç doğrultusunda atılacak her bir adımın, firmanın vizyonuna, misyonuna, temel değerlerine uygun bir şekilde gerçekleştirilmesi gerekmektedir. Firma kendisine katma değer sağlarken, sürekliliğini devam ettirebilmesi adına en değerli varlığı olan müşterilerini, onların beğenilerini, tercih sıralamalarını göz önüne alarak ilerlemesi gerekmektedir. Bu nedenle geliştirilmiş olan müşteri ilişkileri yönetimi anlayışında, bu husus göz önünde bulundurularak yöntemler geliştirilmektedir. Müşteri ilişkileri yönetim programları işletmeler için model oluşturmaktadır. İşletmelerin mevcut müşterilerini korumaları, gelecekte müşteri sadakati oluşturmaları açısından müşteri beğenilerine uygun şekilde hareket edebilecekleri yazılımlar ile müşteri bilgilerini kayıt altında tutarak işletme davranışlarını geliştirmektedir (Video Eğitim Portalı, 2016). Buradan da anlaşıldığı üzere, müşteri beğenileri, satın alma alışkanlıkları firmalar için geliştirecekleri stratejilerde önemlilik arz etmektedir.

Müşteri alışkanlıklarını tanımlarken, müşterilerin satın alma hareketleri izlenmektedir. Birliktelik yöntemleri bu noktada devreye girmektedir. Satın alma hareketlerinden yola çıkarak, hangi ürün hangi ürün ile birlikte satılmakta, bu ürünleri satın alan müşteri grupları arasında ne tür benzerlikler var. Firmanın düzenleyeceği satış/pazarlama kampanyalarında hangi grup müşteriye yöneleceği, hangi ürün grubunu öne çıkarması gerektiği, en çok satılan ürün ile en az satılan ürünü nasıl bir araya getirebileceğine dair çözüm önerileri geliştirme aşamasında büyük önem arz etmektedir.

Bu noktadan hareketle, uygulamada üzerinde durulacak olan temel nokta, firmanın ürün yelpazesindeki ürünlerden hangi ikili, üçlü, dörtlü kombinasyonların birlikte satın alındığını keşfetmek olacaktır. Böylelikle geliştirilecek olan pazarlama kampanyalarında bu birliktelikler göz önünde bulundurulabilir, çok satılan ürün ile daha az satılan ürünler bir araya getirilerek ürün bazında yeni satış hedefleri oluşturulabilir, müşterilere sunulacak olan kataloglarda bu ürün kombinasyonlarına birlikte yer verilebilecektir. Aynı şekilde, satın alma davranışları incelenen müşteriler gruplara ayrılabilir. Böylece farklı gruplarda yer alan müşterilere farklı satış politikaları sunulabilmektedir. Ödeme tablolarından riskli müşteriler ve cironun büyük bir kısmını karşılayan müşteriler tespit edilerek, geliştirilecek satış politikaları ve müşteri ilişkilerinde bu bilgiler ışığında ilerleme yoluna gidilebilir.

Bu tez çalışmasında esas olarak *müşteri ilişkileri yönetimi çerçevesinde, müşterilerin tercihleri göz önünde bulundurularak, birliktelik kuralları ile gerçekleştirilen analizlerle algoritmaların performansını karşılaştırarak, ürün birlikteliklerinin ortaya çıkarılması* probleminin ele alınmasıdır.

3.2. VERİYİ ANLAMA

Analizlerde kullanılmış olan veri seti, elektrik sektöründe faaliyet gösteren X Firması tarafından temin edilmiştir. Veri setinde firmanın müşterilerine ait herhangi bir açıklayıcı bilgi bulunmamaktadır. Müşteriler “Cari Kod” alanı ile ayırt edilmektedir. Veriler 2014-2015 yıllarını kapsamaktadır. Veriler firmadan kademeli olarak temin edilmiştir. Firma, istenilenler doğrultusunda, verileri üç ayrı tabloda paylaşma yoluna gitmiştir. Bunlar, ilk olarak firmanın gerçekleştirmiş olduğu fatura bazında satışların ve satılan ürün detaylarının yer aldığı “Satışlar Tablosu”, çeşitli nedenlerden dolayı, müşteri tarafından geri iade edilmiş olan ürün detaylarının yer aldığı “Satış-İade Tablosu” ve son olarak müşterilerin firmaya karşı olan finansal göstergelerin (alacak, borç bilgileri) yer aldığı “Müşteri Finansal Durum Tablosu” şeklindedir.

Çalışmalar ilk aşamada “Satışlar Tablosu” üzerinden ilerletilmiştir. Diğer tablolara da yapılan analizleri desteklemek amacı ile başvurulmuştur. Analizler yıl bazında yapılmış ve karşılaştırmalı olarak ele alınmıştır. İlk olarak 2014 yılı ele alınmıştır: Veri setinin birinci kısmı Ocak 2014-Aralık 2014 dönemine ait olup, toplamda 19 nitelik ve 177393

kayıttan oluşmaktadır. İkinci kısım 2015 yılına ait olup, 23 nitelik ve 183401 kayıttan oluşmaktadır. Yıllar arasındaki nitelik sayısı farkı, firmanın veri tabanında verilerini saklama biçiminde getirdiği yenilikten kaynaklanmaktadır. Firma 2015 yılında satış tablosunda yer alan niteliklerine 4 ayrı alan ilave etmiştir. Bunlar sırasıyla; “Mal Adı”, “Marka Son”, ”Kablo Türü” ve “Prim” dir. Analizlerde ürünler birbirlerinden mal kodları ile ayrıldıklarından “Mal Adı” alanı ürünlere anlaşılabilirlik getirmek adına başvurulmuştur.

Analize her hangi bir etkisi bulunmamaktadır. “Marka Son” alanı marka adlarında, standartlığı yakalamak adına yenilemeye gidilmesi sebebi ile oluşturulmuştur. Aynı şekilde firma, 2014 yılı içerisinde kablo türlerini veritabanında ayrıca belirtmezken, 2015 yılında, firmanın ana ürünü kabloyu dört ayrı kategoride sınıflandırma yoluna gitmiştir. Bunlar sırasıyla; Aydın (Aydınlatma), Genel, Kablo, KKablo şeklindedir. Tablolar ve niteliklerinin detaylı açıklaması yıl bazında ele alınmış olup, Tablo 3.1 ve Tablo 3.2’deki gibidir:

Tablo 3.1: 2014 Yılı satış Tablosu Açıklamaları.

	Nitelik Adı	Açıklama	Veri Tipi
1.	Evrak_No	Fatura numarası	Nümerik
2.	Tarih :	Satış tarihi	Tarih
3.	Cari_kod :	Müşteri numarası	Nümerik
4.	Mal_kodu :	Mal kodu	Kategorik
5.	Marka :	Ürün markası	Kategorik
6.	Marka alt grup :	Marka alt grubu	Kategorik
7.	Valör_günü :	Faiz hesaplamasına başlangıç oluşturan gün	Nümerik
8.	Vade_tarihi :	Vade tarihi	Tarih
9.	İskonto_oranı_1 :	İskonto oranı	Nümerik
10.	İskonto_toplamı :	İskonto toplamı	Nümerik
11.	Depo :	Ürünün çıkış deposu	Kategorik : ("Adn", "Ank", "Bur", "Crl", "Ddl", "Ddr", "Gbz", "Gop", "Hdk", "Prp", "Tmp", "Zmr")
12.	Birim :	Ürün birimi	Kategorik : ("ad", "adet", "bule", "kg", "metr", "mt", "pk")
13.	Birim_miktar	Ürün miktarı	Nümerik
14.	Birim_fiyat :	Ürün birim fiyat	Nümerik
15.	Birim_tutar :	Ürün birim tutar	Nümerik
16.	Tipkod	Satış temsilcisi	Kategorik : ("Akyol", "Ali", "Bayram", "Berrin", "Burak", "Cüneyt", "Dursun", "Eyüp", "Ferhat", "Gökhan", "Gökmen", "Harun", "Hasan", "İhracat", "İhsan", "Kaan", "Kemal", "Kerim", "Mustafa", "Nüvey", "Oğuz", "Ömer", "Proje1", "Proje2", "Proje3", "Proje4", "Samancı", "Saticı", "Sepet", "Servet", "Soner", "Şenyurt", "Timur", "Tolga", "Ulaş", "Ümit", "Veysel")
17.	Grupkod :	Satış bölgesi	(kategorik : "Pazar", "Adana", "anadolu", "Ankara", "Aydınlatma", "Bursa", "Çorlu", "Gebze", "İhracat", "İzmir", "Kadıköy", "Karaköy", "Pano", "Sat", "Tarkyal", "Trakya2")
18.	Net_birim_fiyat :	Ürün net birim fiyat	Nümerik

Tablo 3.2: 2015 Yılı Satış Tablosu.

	Nitelik Adı	Açıklama	Veri Tipi
1.	Evrak_No	Fatura numarası	Nümerik
2.	Tarih	Satış tarihi	Tarih
3.	Cari_kod	Müşteri numarası	Nümerik
4.	Mal_kodu	Mal kodu	Kategorik
5.	Mal adı	Mal adı	Kategorik
6.	Marka	Ürün markası	Kategorik
7.	Marka son	Revize edilmiş ürün markası	Kategorik
8.	Ürün alt grup	Ürün alt grubu	Kategorik
9.	Kablo Türleri	Kabloları ait 4 farklı tür	Kategorik
10.	Valör_günü	Faiz hesaplamasına başlangıç oluşturan gün	Nümerik
11.	Vade_tarihi	Vade tarihi	Tarih
12.	İskonto_oranı_1	İskonto oranı	Nümerik
13.	İskonto_toplamı	İskonto toplamı	Nümerik
14.	Depo	Ürünün çıkış deposu	Kategorik : ("adn","ank","bur","crl","ddl","gbz","gop","hdk","prp","tmp","zmr","zmr2")
15.	Birim	Ürün birimi	Kategorik : ("ad","adet","bule","kg","metr","mt","pk")
16.	Birim_miktar	Ürün miktarı	Nümerik
17.	Birim_fiyat	Ürün birim fiyat	Nümerik
18.	Birim_tutar	Ürün birim tutar	Nümerik
19.	Prim	Prim	Kategorik
20.	Tipkod	Satış temsilcisi	Kategorik : ("Akyol","Ali","Bayra","Bayram","Berrin","Burak","Dursun","Eyüp","Fatih","Ferhat","Gökhan","Harun","Hasan","İhracat","İsmail","Kaan","Kemal","Kerim","Musta","Nüvey","Oğuz","Orhan","Ömer","Proje1","Proje2","Proje3","Proje4","Rukiye","Satıcı","Sepet","Servet","Soner","Şenyurt","Timur","Tolga","Ulaş","Ümit","Veysel")
21.	Grupkod	Satış bölgesi	Kategorik : ("Adana","Anadolu","Ankara","Aydınlatma","Bursa","Çorlu","Gebze","İhracat","İzmir","Kadıköy","Karaköy","Pano","Pazar","Perpa","Proje","Sat","Tarkya1","Trakya2")
22.	Net_birim_fiyat	Ürün net birim fiyat	Nümerik
23.	Net_tutar	Ürün net tutar	Nümerik

2014 ve 2015 yıllarına ait iade tabloları satış tabloları ile aynı alan adlarını içermekte olup, çeşitli nedenler ile müşteriler tarafından iadesi yapılan ürünleri içermektedir. Bu nedenle iade tablosuna bu kısımda ayrıca yer verilmemiştir. Tablo 3.3' de müşterilerin finansal durum bilgilerine yer verilmiştir.

Tablo 3.3: 2014-2015 Yılı Müşteri Finansal Durum Tablosu.

	Nitelik Adı	Açıklama	Veri Tipi
1.	Grup kodu	Bölge	Kategorik
2.	Tip kodu	Satış Temsilcisi	Kategorik
3.	Hesap Kodu	Cari Kod (Müşteri Numarası)	Nümerik
4.	2013 Devir Bakiye	2013'den devreden bakiye miktarı	Nümerik
5.	2014 Borç	2014'den borç	Nümerik
6.	2014 Alacak	2014'den alacak	Nümerik
7.	Toplam Bakiye	Toplam bakiye	Nümerik
8.	Ortalama Vade	Müşterinin kalan bakiyesinin ortalama vadisi	Tarih
9.	Vade Farkı	Geciken borçlar için uygulanacak vade farkı	Nümerik
10.	Gün	Müşterilerin yapmış olduğu geç ödemelerin ortalama gün sayısı	Nümerik
11.	Kapatın Vade	Müşterinin borçlarını bir sonraki dönemde ödeyebileceği tarih.	Tarih
12.	Net Ciro	2014 yılındaki alım iadeler toplamı	Nümerik

Şekil 3.1 ve Şekil 3.2' de her iki yıla ait veri setinin özeti yer almaktadır.

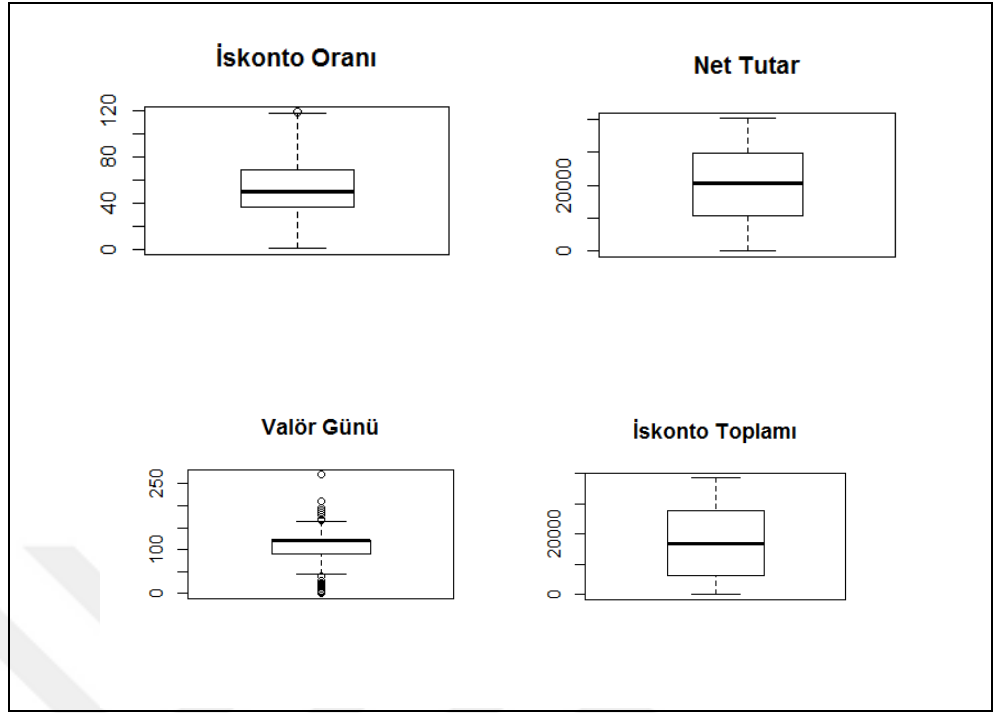
EVRAK_NO	TARİH	CARI_KOD	MAL_KODU	MARKA	MARKA.ALT.GRUP	VALOR_GUNU
IH420646:	38 30.9.2014	: 1482	1718720: 1916	NEXHFNYM07: 2120	SCHNEIDER :49322	MGPWCTR: 17293 Min. : 0
IH420736:	38 8.1.2014	: 1306	3422900: 1882	NEXCAT6U : 1794	LEGRAND :36734	MGPWFDB: 12814 1st Qu.: 90
IH420513:	36 30.4.2014	: 1263	3611500: 1834	NEXCAT5U : 1286	HES KABLO :23383	MGIDSIG: 9004 Median :120
IH420725:	36 6.1.2014	: 1028	3707900: 1684	KONKV110 : 1129	NEXANS KABLO:14043	LGARM4 : 7983 Mean :108
IH420677:	35 24.11.2014	: 1015	1933800: 1275	NEXHFNYM06: 1121	KONDAŞ :11402	KO-KV : 6817 3rd Qu.:120
IH420598:	31 20.1.2014	: 986	1937500: 1194	BULEG77213: 1059	ERSE KABLO : 6680	H-NYAF : 6013 Max. :270
(other) :171718	(other) :170312	(other):167607	(other) :168883	(other) :35828	(other):117468	
VADE_TARIHI	ISKONTO_ORANI_1	ISKONTO_TOPLAMI	DEPO	BIRIM	BIRIM_MIKTAR	BIRIM_FIYAT
25.6.2014 :	955 Min. : 1.00	Min. : 1	HDK :83998	ADET :118318	Min. : 1.0	Min. : 1
6.5.2014 :	880 1st Qu.: 37.00	1st Qu.: 6486	GOP :25645	MT : 57883	1st Qu.: 4.0	1st Qu.:1400
25.12.2014:	879 Median : 50.00	Median :17037	DDL :19652	AD : 389	Median : 601.0	Median :3660
28.1.2015 :	835 Mean : 53.05	Mean :17326	PRP :12399	PK : 258	Mean : 732.9	Mean :3652
19.6.2014 :	833 3rd Qu.: 69.00	3rd Qu.:27699	TMP :10264	KG : 130	3rd Qu.:1194.0	3rd Qu.:5735
28.8.2014 :	831 Max. :118.00	Max. :38542	ZMR : 5200	METR : 124	Max. :2048.0	Max. :7671
(other) :172179	(other) :170312	(other):167607	(other):20234	(other): 290		
BIRIM_TUTAR	Tipkod	Grupkod	NET_BIRIM_FIYAT	NET_TUTAR		
Min. : 1	ÜMİT : 10870	KARAKÖY:41569	Min. : 1	Min. : 1		
1st Qu.: 6174	SAMANCI: 10541	TRAKYA2:24432	1st Qu.: 183	1st Qu.:10584		
Median :12301	EYÜP : 10372	KADIKÖY:21369	Median :2851	Median :20643		
Mean :12196	HARUN : 8105	TRAKYAI:20829	Mean :3252	Mean :20248		
3rd Qu.:18218	GÖKHAN : 7695	ANADOLU:11220	3rd Qu.:5371	3rd Qu.:29743		
Max. :24272	SONER : 7516	PANO : 9064	Max. :8822	Max. :40148		
(other):122293	(other):48909					

Şekil 3.1: 2014 Yılı Veri Setinde Yer Alan Niteliklere ait Tanımlayıcı Bilgi.

EVRAK_NO	TARİH	CARI_KOD	MAL_KODU	MARKA	MARKA.ALT.GRUP	VALOR_GUNU
IH420829:	49 31.8.2015	: 1507	1718720: 1743	NEXCAT6U : 2111		
IH421955:	49 30.11.2015:	1454	2406500: 1253	NEXHFNYM07 : 1875		
IH422096:	40 5.1.2015	: 1246	3509430: 1244	KONKV105 : 1117		
IH422220:	39 30.6.2015	: 1234	1933800: 1156	KONKV110 : 1114		
IH422171:	34 30.9.2015	: 1182	2568950: 1140	NEXCAT6U HF LZSH: 1113		
IH421971:	33 15.1.2015	: 1148	1575510: 1138	NEXCAT5U : 1093		
(other) :183158	(other) :175631	(other):175728	(other) :174979			
MAL_ADI	MARKA_SON	URUN.ALT.GRUBU	KABLO.TURLERİ			
CAT6 UTP	NEXANS : 2111	SCHNEIDER :58163	MGPWCTR : 20373			
3x2,5 (N)HXMH NYM 500V NEX	: 1875	LEGRAND :28875	MGPWFDB : 14591	AYDIN : 892		
5 KVAR KONDANSATÖR KONDAŞ	: 1117	HES KABLO :22450	MGIDSIG : 9730	GENEL :125656		
10 KVAR KONDANSATÖR KONDAŞ	: 1114	NEXANS KABLO:14109	KO-KV : 7077	KABLO : 40214		
CAT6 UTP (HF)	NEXANS : 1113	KONDAŞ :12371	H-NYA : 6723	KKABLO: 16639		
CAT5 UTP	NEXANS : 1093	VATAN KABLO : 6364	LG-ARMADA: 6575			
(other)	:174979	(other) :41070	(other) :118333			
VALOR_GUNU	VADE_TARIHI	ISKONTO_TOPLAMI	ISKONTO_ORANI..	DEPO	BIRIM	BIRIM_MIKTAR
Min. : 0	11.6.2015 : 986	Min. : 1	Min. : 1.00	HDK :89978	ADET :122016	Min. : 1
1st Qu.: 90	28.1.2016 : 985	1st Qu.: 6749	1st Qu.: 43.00	GOP :24723	MT : 60970	1st Qu.: 7
Median :120	7.7.2015 : 897	Median :17780	Median : 55.00	DDL :22419	KG : 120	Median : 571
Mean :109	17.6.2015 : 885	Mean :17953	Mean : 57.74	PRP :11363	AD : 85	Mean : 689
3rd Qu.:120	19.11.2015: 882	3rd Qu.:28753	3rd Qu.: 68.00	TMP :10705	METR : 80	3rd Qu.:1122
Max. :900	28.10.2015: 882	Max. :39727	Max. :123.00	ZMR : 6119	AD : 61	Max. :1909
NA's :1	(other) :177885		(other):18095	(other): 70		
BIRIM_FIYAT	BIRIM_TUTAR	PRIM	Tipkod	Grupkod	NET_BIRIM_FIYAT	NET_TUTAR
Min. : 1	Min. : 1	: 1	ÜMİT : 11753	TRAKYA2:25067	Min. : 1	Min. : 1
1st Qu.: 745	1st Qu.: 5927	: 697	RUKİYE : 9633	KADIKÖY:24422	1st Qu.: 189	1st Qu.:10542
Median :2640	Median :11901	Tüm:182704	EYÜP : 9488	TRAKYAI:22589	Median :3285	Median :20538
Mean :2541	Mean :11806		SONER : 9170	PERPA :21386	Mean :3574	Mean :20571
3rd Qu.:4060	3rd Qu.:17741		ULAŞ : 8343	KARAKÖY:16858	3rd Qu.:5855	3rd Qu.:30702
Max. :5787	Max. :23545		ORHAN : 8090	ANADOLU:11713	Max. :9636	Max. :41022
(other):126925	(other):61367					

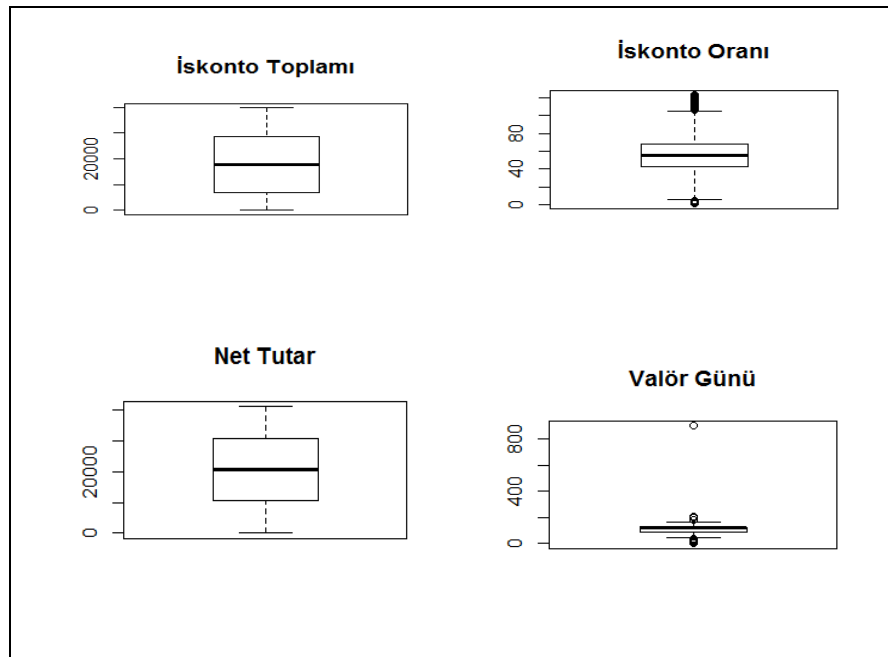
Şekil 3.2: 2015 Yılı Veri Setinde Yer Alan Niteliklere ait Tanımlayıcı Bilgi.

Sayısal değerlerin dağılımını görmek, var ise uç noktaları tespit etmek amacı ile kutu grafiğine başvurulmuştur.



Şekil 3.3: 2014 Yılı Kutu Grafikleri.

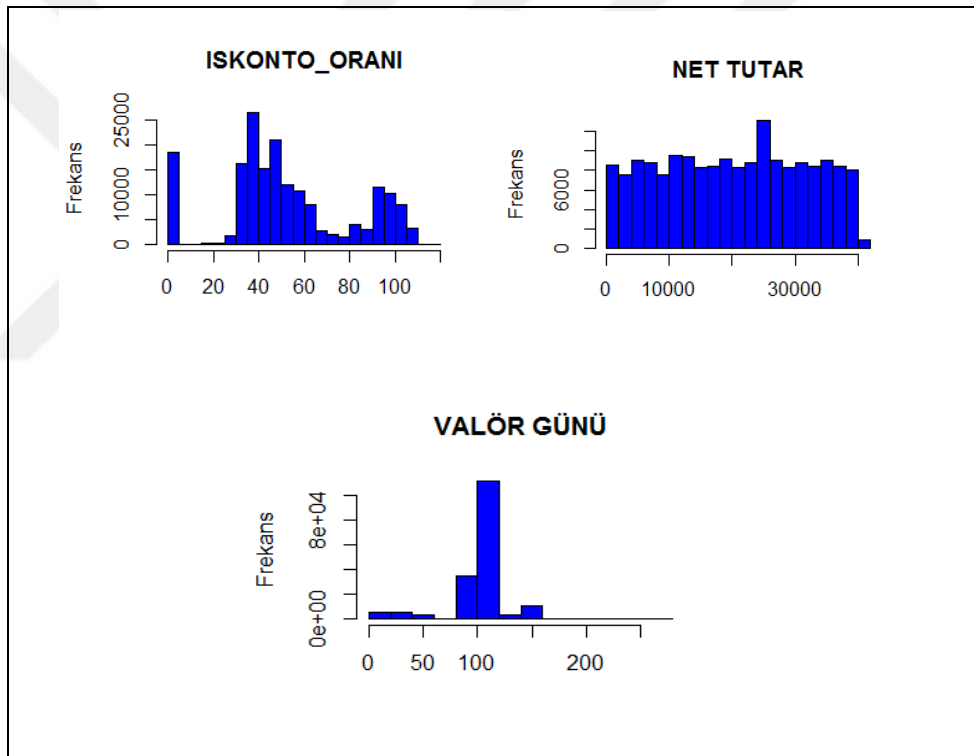
2014 yılına ait nümerik niteliklerin kutu grafikleri incelendiğinde, iskonto toplamı ve net tutar’ da verilerin simetrik dağıldığı, uç değerlerin olmadığı gözlemlenmiştir. İskonto oranında, verilerin birinci çeyrekte toplandığı ve bazı uç değerlerin olduğu, valör gününde ise uç değerlerin sıklıkla yer aldığı, değerlerin farklı aralıklarda bulunduğu görülmüştür.



Şekil 3.4: 2015 Yılı Kutu Grafikleri.

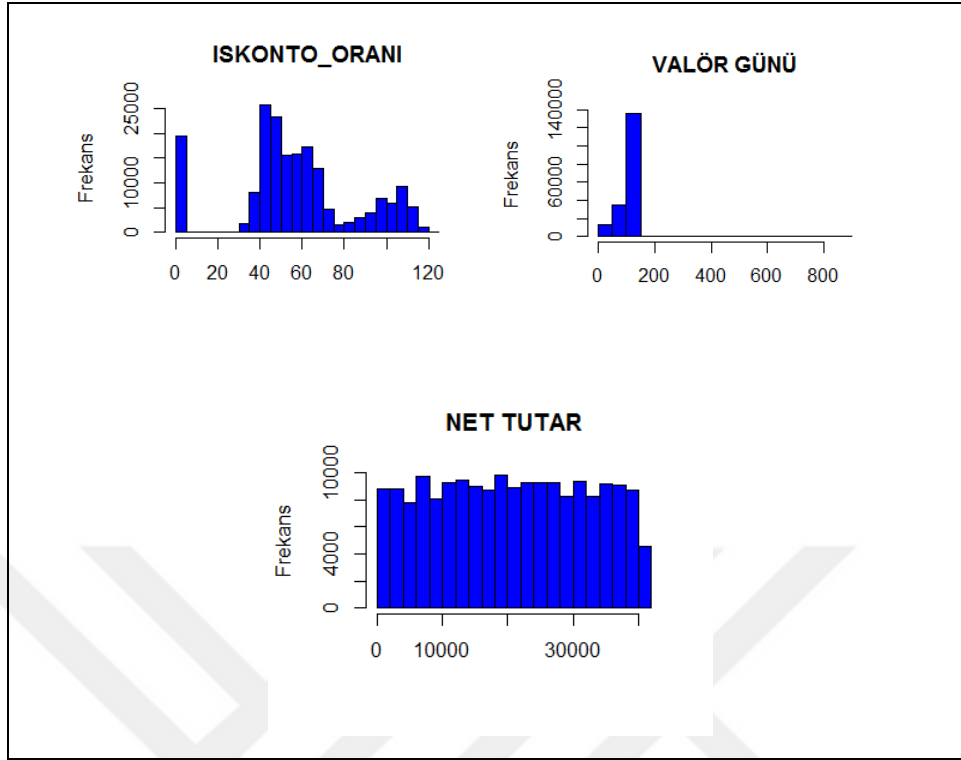
2015 yılına ait nümerik niteliklerin kutu grafikleri incelendiğinde, iskonto toplamı ve net tutar' da verilerin simetrik dağıldığı, uç değerlerin olmadığı gözlemlenmiştir. iskonto oranında, verilerin normal dağıldığı ancak uç değerlerin olduğu, valör gününde ise uç değerlerin yer aldığı, değerlerin sıklıkla bir aralıkta biriktiği görülmüştür.

Şekil 3.5' de 2014 yılına ait iskonto oranı, net tutar, valör günü niteliklerinin dağılımını gösteren histogramlara yer verilmiştir. Histogramda görülen iskonto oranlarının sıklıkla %40 civarında olduğu görülmektedir. Ürün bazında net tutarlara bakıldığında, net tutarların 30.000 TL civarında toplandığı görülmektedir. Valör günü niteliği incelendiğinde, valör gününün sıklıkla 100 günü aştığı görülmektedir.



Şekil 3.5: 2014 Yılı Histogramları.

2015 yılı veri setinde yer alan niteliklerinin dağılımlarını görmek amacıyla Şekil 3.6' da yer alan histogramları incelediğimizde, iskonto oranlarının %40 civarında toplandığı, valör günlerinin ise 180 gün civarında toplandığı görülmektedir. Ürün net tutarları incelendiğinde ise homojen bir dağılım olduğu görülmektedir.



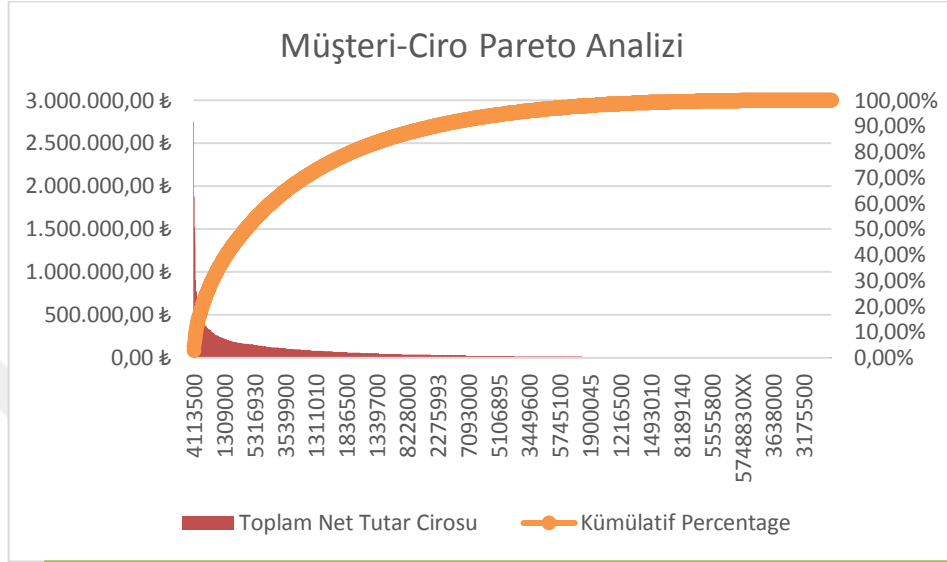
Şekil 3.6: 2015 Yılı Histogramlar.

Pareto Analizi

Pareto analizi, karar vermede önemli bir sonuç veren sınırlı sayıda görev seçimi için kullanılan istatistiksel bir tekniktir. Bir problemi oluşturan sebeplerin ne olduklarını ve önem derecelerini belirlemek amacı kullanılmaktadır. En çok tercih edilen yöntemlerden birisidir. Pareto analizi, bir organizasyon içindeki hangi görevlerin veya faktörlerin en fazla etkileneceğini belirlemeye çalışırken kullanılan bir yöntemdir (Talib ve diğ., 2011). Karar alma işlemine yardımcı olmaktadır. Analiz, kaynakların % 20'sinin sorunların % 80'ine neden olduğuna dair kanıtlanmış 80-20 kuralı olarak da bilinen İtalyan Ekonomist Vilfredo Pareto tarafından geliştirilmiş Pareto ilkesine dayanmaktadır (He ve Khouja, 2011).

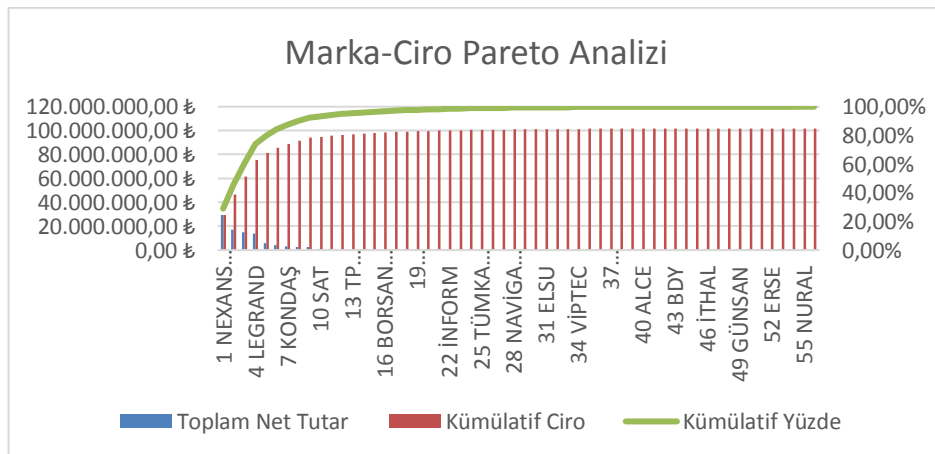
2014 yılı veri seti üzerinde, en kritik olduğu düşünülen ve dağılımını incelemek istediğimiz nitelikler üzerinde pareto analizi uygulanmıştır. Pareto analizi sırasında Ms. Excel'den faydalanılmıştır. Mevcut veri setinde müşteri ciroları, marka ve bölgeler üzerinde pareto analizi uygulanmıştır.

Böylelikle müşteri ciroları, marka ve bölge ciroları ayrı ayrı incelenerek toplam ciroda önemli role sahip olan müşteri, marka ve bölgelerin ortaya çıkarılması hedeflenmiştir. 2014 yılı veri setine ait pareto analizlerine Şekil 3.7, Şekil 3.8 ve Şekil 3.9'da yer verilmiştir.



Şekil 3.7: 2014 Yılı Müşteri-Ciro Pareto Analizi.

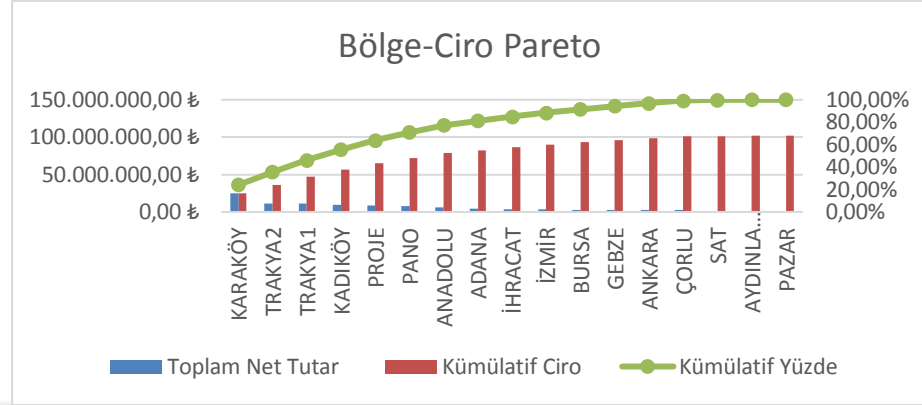
Müşteri ciroları üzerinden yapılan pareto analizinde, toplam cironun %80'ini yani cironun önemli kısmını oluşturan müşterileri grafik üzerinden tespit etmek mümkündür. Geriye kalan müşterilerin, cironun ancak %20'lik kısmına etki ettiği tespit edilmiştir.



Şekil 3.8: 2014 Yılı Marka-Ciro Pareto Analizi.

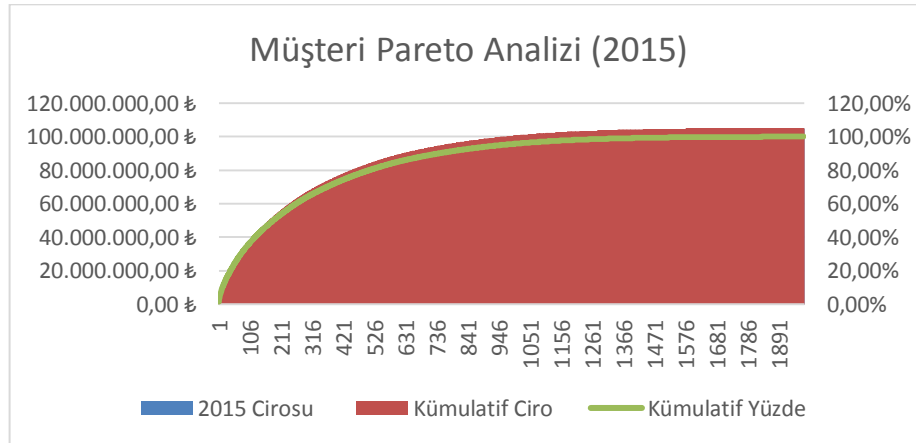
Marka ciroları üzerinden yapılan pareto analizinde, toplam cironun %80'ini yani cironun önemli kısmını oluşturan markaları grafik üzerinden tespit etmek mümkündür.

Grafikten bakıldığında, ilk 10 daki markaların, firma cirosunun önemli payı olan yaklaşık %80'nini oluşturduğu tespit edilmiştir.



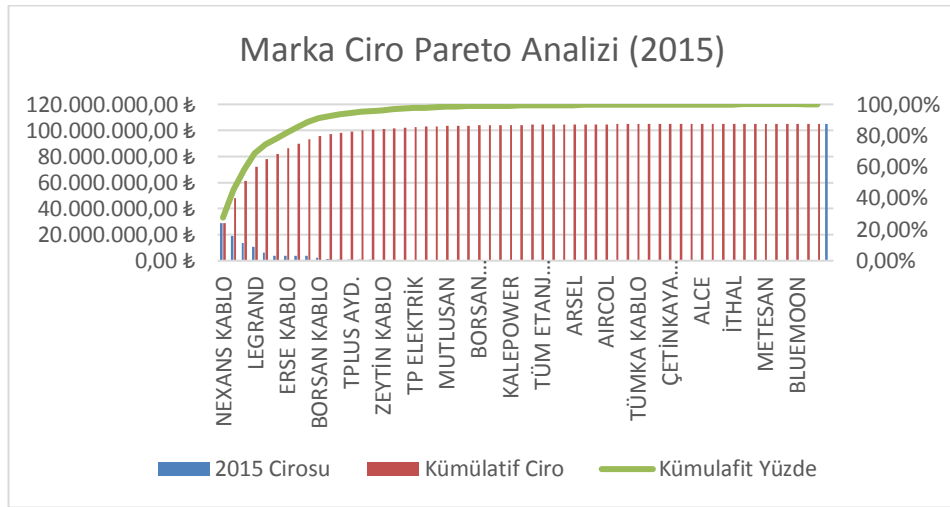
Şekil 3.9: 2014 Yılı Bölge Ciro Pareto Analizi.

Bölge ciroları üzerinden yapılan pareto analizinde, toplam cironun %80'ini yani cironun önemli kısmını oluşturan bölgeleri grafik üzerinden tespit etmek mümkündür. Geriye kalan bölgelerin, cironun ancak %20'lik kısmına etki ettiği tespit edilmiştir. Grafik incelendiğinde, Karaköy Bölgesinden Bursa Bölgesine kadar olan bölgelerin firma cirosunun yaklaşık %80'nini oluşturduğu tespit edilmiştir. Pareto analizi 2015 yılı verilerine uygulandığında elde edilen grafiklere Şekil 3.10, Şekil 3.11 ve Şekil 3.12'de yer verilmiştir.



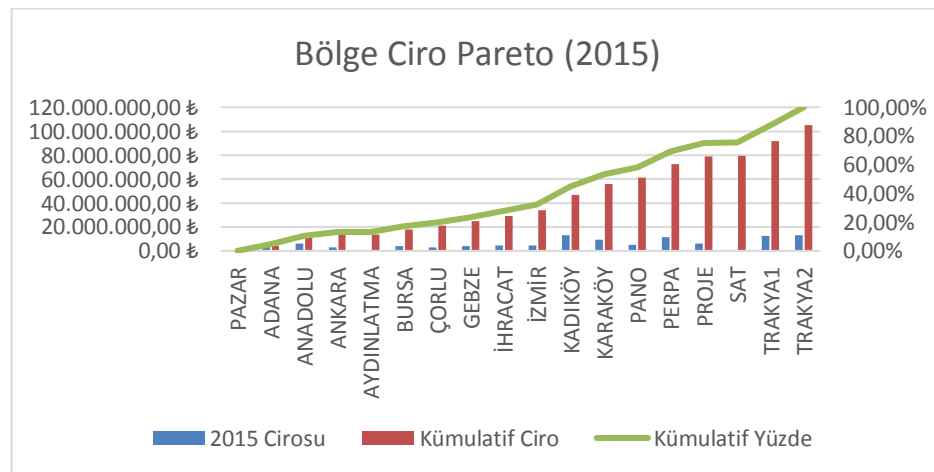
Şekil 3.10: 2015 Yılı Müşteri Ciro Pareto Analizi.

Müşteri ciroları üzerinden yapılan pareto analizinde, toplam cironun %80'ini yani cironun önemli kısmını oluşturan müşterileri grafik üzerinden tespit etmek mümkündür. Geriye kalan müşterilerin, cironun ancak %20'lik kısmına etki ettiği tespit edilmiştir.



Şekil 3.11: 2015 Yılı Marka Ciro Pareto Analizi.

Marka ciroları üzerinden yapılan pareto analizinde, toplam cironun %80'nini yani cironun önemli kısmını oluşturan markaları grafik üzerinden tespit etmek mümkündür. Grafik incelendiğinde, Nexans Kablo'dan, Erse Kablo'ya uzanan kısımda yer alan markaların firma cirosunun %80'ni oluşturduğu tespit edilmiştir. Geriye kalan markalar, firmanın %20 'lik cirosunda payını almaktadırlar. 2014 yılında beş marka (Nexans, Schneider, Hes Kablo, Legrand, Vatan Kablo) %80'lik dilimi paylaşıırken, 2015 yılında pastanın büyük dilimine Pektaş ve Erse Kablo'nun da katıldığı tespit edilmiştir.



Şekil 3.12: 2015 Yılı Bölge Ciro Pareto Analizi.

Bölge ciroları üzerinden yapılan pareto analizinde, toplam cironun %80'nini yani cironun önemli kısmını oluşturan bölgeleri grafik üzerinden tespit etmek mümkündür. Grafik incelendiğinde, SAT bölgesine kadar (18 bölge) olan bölgelerin firma cirosunun yaklaşık (%75,36) %80'ni oluşturduğu, geriye kalan bölgelerin, cironun ancak %20'lik kısmına

etki ettiđi tespit edilmiřtir. 2014 yılında yalnızca sekiz bölge pastanın büyük diliminde pay alırken, 2015 yılında bu dilimi paylaşan marka sayısının on sekize çıktıđı tespit edilmiřtir. Buradan hareketle firmanın bölge satışlarında dengelemeye gittiđi düşünölebilmektedir. Veriyi anlamlandırmak adına, özellikle büyük veri kümesini daraltmak, veri noktaları arasındaki ilişkileri analiz etmek adına pivot analizlerinden faydalanılmıřtır.

Pivot Analizleri :

2014-2015 yıllarına ait satış veri tabanlarında yer alan temel niteliklerin nicelik olarak karşılaştırılmasına Tablo 3.4' de yer verilmiřtir. Karřımıza çıkan tabloda, firmanın 2015 yılında firmanın 133 yeni müşteri edindiđi görölmektedir. Buna karşın, mal kodu ürün çeřitliliğinde azalmaya gittiđi görölmektedir. "Mal adı" alanı 2015 yılında veri tabanında yapılan düzenleme sırasında içeriđe dahil edilmiřtir.

Böylelikle mal hakkında, mal kodunun yanı sıra, açıklamalar ile daha detaylı bilgi edinilebilmesi amaçlanmıřtır. Bazı markaların adlandırılmasında detaylandırılmaya, marka isminin açık olarak yazılmaya başlandıđı (örneđin 2014 yılında Zeytin řeklinde tutulan Zeytin Kablo markası, 2015 yılında tam adı ile "Zeytin Kablo" olarak veri tabanında yerini almıřtır.) tespit edilmiřtir. Firma veritabanı kayıtlarında standartlařmayı yakalamak adına bu gibi bir takım yeniliklere gitmiřtir. 2014 yılında firmanın ana ürünü kabul edilen kabloda her hangi bir sınıflandırma yapılmamaktadır. Bu nedenle 2014 yılı için kablo türü sayısı Tablo 3.4' de gösterilmiřtir. 2015 yılında kablolar "AYDIN", "GENEL", "KABLO", "KKABLO" olmak üzere dört sınıfa ayrılmıřtır. TipKod (satış temsilcisi) sayısı aynı kalırken, grupkod (bölge) ve depo sayısı 2015 yılında bir artış göstermiřtir.

Tablo 3.4: 2014-2015 Yılları Veri Tabanı Nitelik Karşılaştırma.

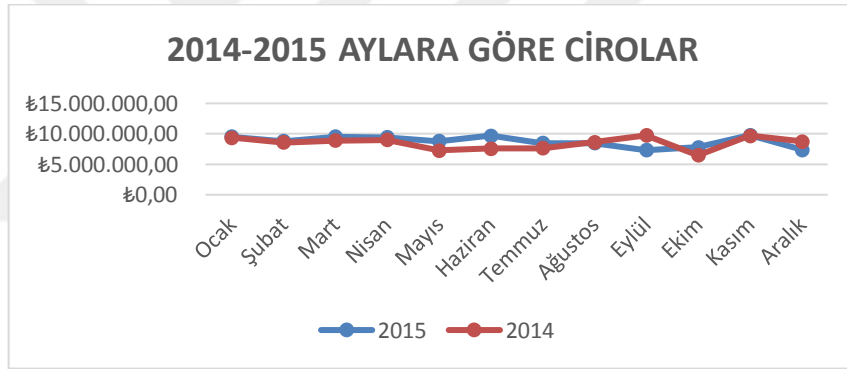
Nitelik	2014	2015
Yapılan ciro tutarı	101.678.568,4 TL	105.056.674,25 TL
İşlem yapan toplam cari hesap sayısı	1842	1975
İşlem gören Mal Kodu Sayısı	8521	7559
İşlem gören Mal Adı Sayısı	X	7192
İşlem gören Marka Sayısı	56	57
İşlem gören Ürün Alt Grubu Sayısı	205	196
İşlem gören Kablo Türü Sayısı	X	4
İşlem yapan TipKod Sayısı	38	38
İşlem yapan GrupKod Sayısı	17	18
İşlem yapılan Depo Sayısı	12	13

*X: Mevcut değil

Tablo 3.5: 2014-2015 Yılları Cironun Aylık Dağılım Karşılaştırması.

ÇİROLAR		
AYLAR	2015	2014
Ocak	9.506.987,59 ₺	9.373.406,35 ₺
Şubat	8.796.879,97 ₺	8.603.069,28 ₺
Mart	9.536.381,51 ₺	8.912.427,15 ₺
Nisan	9.447.487,72 ₺	9.009.661,06 ₺
Mayıs	8.791.950,16 ₺	7.269.836,10 ₺
Haziran	9.686.086,68 ₺	7.611.267,10 ₺
Temmuz	8.483.627,64 ₺	7.657.596,64 ₺
Ağustos	8.503.898,04 ₺	8.621.595,61 ₺
Eylül	7.306.505,81 ₺	9.739.031,77 ₺
Ekim	7.818.166,14 ₺	6.472.318,64 ₺
Kasım	9.785.158,00 ₺	9.691.570,84 ₺
Aralık	7.393.544,99 ₺	8.716.787,86 ₺
TOPLAM	105.056.674,25 ₺	101.678.568,40 ₺

Firmanın aylık ciro hareketleri üzerinde ayların bir etkisi olup olmadığını tespit etmek adına, her iki yılın aylarını da kapsayan satış eğrisine Şekil 3.13 yer verilmiştir. 2014-2015 yılları ay bazında cirolar karşılaştırıldığında, Haziran ve Eylül aylarındaki hareketlenmeler göze çarpmaktadır. Haziran ayında 2014'den 2015'e pozitif yönde 2.074.819,78 ₺'lik bir artış olurken, Eylül ayında negatif yönde -2.432.525,96 ₺'lik bir azalma olduğu görülmektedir. Cirosunu koruyan ay ise yalnızca 133.581,24 ₺'lik fark ile Ocak ayı olduğu gözlemlenmektedir. Genel olarak aşağıdaki grafik incelendiğinde firma satışlarında net bir dönem etkisinin olduğundan bahsetmenin mümkün olmadığı görülmektedir. Bu noktada inşaat sektörüne elektrik malzemeleri satışı yapan firmanın satışlarını stabil bir eğride devam ettirebilmesini inşaat sektörünün de her ay canlı kalıyor olmasına bağlayabiliriz. Böylelikle küçük iniş çıkışlarla, pazarda sürekli arz talep dengesinin korunduğu tespit edilmiştir.



Şekil 3.13: 2014-2015 Aylara Göre Cirolar.

2014-2015 yıllarında aylara göre ciro dağılımını gösteren Şekil 3.13'de satışların aylara göre dağılımının analizine bakıldığında, 2014 yılında en düşük cironun Ekim (7.306.505,81 TL) ayında, 2015 yılında en düşük cironun Kasım (6.472.318,64 TL) ayında yapıldığı görülmektedir. En yüksek cirolara bakıldığında, 2014 yılında Eylül (9.739.031,77 TL), 2015 yılında Ekim ayının en yüksek ciroya sahip olduğu görülmektedir.

2015 yılı için firmanın yapmış olduğu cironun bölgelere göre dağılımına Tablo 3.6' da yer verilmiştir. Tabloda cirolar büyüklüğe göre artandan azalana göre sıralanmıştır. 2015 yılı ciro sıralamasında, 2. ve 3. Sıradaki Kadıköy ve Trakya1 bölgeleri arasında çok büyük fark olmadığı gözlenirken, Trakya2 bölgesinin en yüksek ciroyu yaparak en üst sırada yerini almaktadır. Müşteri sayılarına yer verilme sebebi müşteri başına düşen ciroyu göz

önünde bulundurularak, o bölgeye ait müşterilerin satın alma gücünü ölçülmektedir. Örneğin en yüksek cironun sahibi Trakya2 bölgesinde müşteri başına düşen ortalama ciro 52.473,65 TL iken, 68 müşteri ile Perpa bölgesinde müşteri ortalama cirosu 77.811,67 TL'dir. Öne çıkan başka bir örnek vermek gerekirse, 38 müşterisi ile İhracat bölgesine ait müşteri cirosu 122.573,03 TL'dir. Müşteri sayısı az olmasına rağmen yüksek meblağda satın alımlar yapılan bölgeler olduğundan, bu bölgelerde bulunan müşterilerin toplam ciroya katkısının yüksek olduğu tespit edilmiştir.

Tablo 3.6: 2015 Yılı Bölgelere Göre Ciro Dağılımı.

Bölge	Ciro_2015	Müşteri Sayısı	Ortalama Ciro	Toplam Ciroya Oranı
TRAKYA2	13.170.885,16 ₺	251	52.473,65 ₺	12,54%
KADIKÖY	12.899.426,24 ₺	217	59.444,36 ₺	12,28%
TRAKYA1	12.717.299,21 ₺	168	75.698,21 ₺	12,11%
PERPA	11.593.938,48 ₺	149	77.811,67 ₺	11,04%
KARAKÖY	9.083.706,26 ₺	215	42.249,80 ₺	8,65%
PROJE	6.219.348,07 ₺	110	56.539,53 ₺	5,92%
ANADOLU	6.148.730,45 ₺	173	35.541,79 ₺	5,85%
PANO	5.086.071,14 ₺	68	74.795,16 ₺	4,84%
ADANA	4.744.524,48 ₺	121	39.210,95 ₺	4,52%
İHRACAT	4.657.775,22 ₺	38	122.573,03 ₺	4,43%
İZMİR	4.655.363,80 ₺	131	35.537,13 ₺	4,43%
BURSA	4.074.878,98 ₺	88	46.305,44 ₺	3,88%
GEBZE	3.869.787,48 ₺	60	64.496,46 ₺	3,68%
ÇORLU	3.001.818,16 ₺	42	71.471,86 ₺	2,86%
ANKARA	2.782.499,90 ₺	96	28.984,37 ₺	2,65%
SAT	251.605,58 ₺	23	10.939,37 ₺	0,24%
AYDINLATMA	96.682,59 ₺	23	3.744,77 ₺	0,09%
PAZAR	2.333,05 ₺	1	2.333,05 ₺	0,00%

Tablo 3.7: 2014-2015 Yılı Marka Ciro Dağılımları.

MARKA	2014 Ciro	2015 Ciro	Cirosal Fark	Oransal Fark
NEXANS KABLO	29.399.710,14 ₺	28.991.523,41 ₺	-408.186,73 ₺	-1,4%
SCHNEİDER	17.101.836,89 ₺	18.777.585,28 ₺	1.675.748,39 ₺	9,8%
HES KABLO	14.959.673,50 ₺	13.695.689,20 ₺	-1.263.984,30 ₺	-8,4%
LEGRAND	13.948.669,93 ₺	10.465.608,90 ₺	-3.483.061,03 ₺	-25,0%
VATAN KABLO	5.897.855,61 ₺	6.211.220,20 ₺	313.364,59 ₺	5,3%
PEKTAŞ KABLO	2.384.794,04 ₺	3.988.291,44 ₺	1.603.497,40 ₺	67,2%
ERSE KABLO	4.359.451,97 ₺	3.987.349,38 ₺	-372.102,59 ₺	-8,5%
MAVİLİ	2.538.642,28 ₺	3.690.844,89 ₺	1.152.202,61 ₺	45,4%
KONDAŞ	3.333.116,79 ₺	3.566.151,43 ₺	233.034,64 ₺	7,0%
BORSAN KABLO	443.311,83 ₺	2.333.868,94 ₺	1.890.557,11 ₺	426,5%
SAT	1.061.077,84 ₺	1.475.864,25 ₺	414.786,41 ₺	39,1%
ETKS	892.160,96 ₺	1.074.492,70 ₺	182.331,74 ₺	20,4%
TPLUS AYD.	120,00 ₺	869.161,42 ₺	869.041,42 ₺	724201,2%
PKS K.KANALI	639.812,18 ₺	808.170,40 ₺	168.358,22 ₺	26,3%
BAKIR	454.173,57 ₺	619.023,70 ₺	164.850,13 ₺	36,3%
ZEYTİN KABLO	458.400,77 ₺	581.603,38 ₺	123.202,61 ₺	26,9%
GLOBE İZOLEBANT	257.899,00 ₺	519.343,50 ₺	261.444,50 ₺	101,4%
ENTES	0,00 ₺	424.996,69 ₺	424.996,69 ₺	
TP ELEKTRİK	528.780,36 ₺	415.661,41 ₺	-113.118,95 ₺	-21,4%
KAEL	361.868,62 ₺	396.350,94 ₺	34.482,32 ₺	9,5%
İNFORM	202.396,15 ₺	380.311,60 ₺	177.915,45 ₺	87,9%
MUTLUSAN	298.385,98 ₺	204.095,49 ₺	-94.290,49 ₺	-31,6%
DOĞAN	412.754,76 ₺	157.876,89 ₺	-254.877,87 ₺	-61,8%
NOKTA ZAMAN SAATİ	0,00 ₺	131.317,57 ₺	131.317,57 ₺	
BORSAN MALZEME	163.227,77 ₺	130.439,19 ₺	-32.788,58 ₺	-20,1%
EMFA	113.080,63 ₺	107.644,56 ₺	-5.436,07 ₺	-4,8%
NAVİGA AYD.	108.711,33 ₺	94.261,04 ₺	-14.450,29 ₺	-13,3%

Tablo 3.7 (devam): 2014-2015 Yılı Marka Ciro Dağılımı

KALEPOWER	217.905,25 ₺	86.260,39 ₺	-131.644,86 ₺	-60,4%
GOLD İZOLEBANT	95.443,70 ₺	81.973,00 ₺	-13.470,70 ₺	-14,1%
RENKSAN AYD.	1,00 ₺	81.082,50 ₺	81.081,50 ₺	8108150,0%
TÜM ETANJ PANO	64.831,86 ₺	67.127,40 ₺	2.295,54 ₺	3,5%
GLOBAL AYDINLATMA	127.640,62 ₺	63.365,67 ₺	-64.274,95 ₺	-50,4%
OZCAN AYD.	81.995,40 ₺	63.236,82 ₺	-18.758,58 ₺	-22,9%
ARSEL	5.134,01 ₺	61.186,38 ₺	56.052,37 ₺	1091,8%
ELSU	70.930,40 ₺	52.455,50 ₺	-18.474,90 ₺	-26,0%
PEKTAŞ GRUP PRİZ	15.663,46 ₺	48.937,51 ₺	33.274,05 ₺	212,4%
AIRCOL	39.742,58 ₺	45.677,18 ₺	5.934,60 ₺	14,9%
CHINT	56.796,65 ₺	44.459,50 ₺	-12.337,15 ₺	-21,7%
TÜMPA	32.447,24 ₺	43.740,74 ₺	11.293,50 ₺	34,8%
TÜMKA KABLO	133.146,07 ₺	43.709,81 ₺	-89.436,26 ₺	-67,2%
SU SAATİ	0,00 ₺	41.785,53 ₺	41.785,53 ₺	
ÖZNUR KABLO	17.012,27 ₺	26.018,78 ₺	9.006,51 ₺	52,9%
ÇETİNKAYA PANO	39.148,61 ₺	25.206,58 ₺	-13.942,03 ₺	-35,6%
BDY	6.393,60 ₺	21.418,00 ₺	15.024,40 ₺	235,0%
EATON	195.204,34 ₺	12.767,23 ₺	-182.437,11 ₺	-93,5%
ALCE	23.490,83 ₺	12.631,93 ₺	-10.858,90 ₺	-46,2%
OVİVO	0,00 ₺	12.401,50 ₺	12.401,50 ₺	
YAVUZPANO	33.404,70 ₺	6.050,82 ₺	-27.353,88 ₺	-81,9%
İTHAL	3.212,00 ₺	5.610,50 ₺	2.398,50 ₺	74,7%
VİPTEC	45.045,98 ₺	5.250,19 ₺	-39.795,79 ₺	-88,3%
PEDAŞ	451,10 ₺	2.837,54 ₺	2.386,44 ₺	529,0%
METESAN	201,20 ₺	1.735,60 ₺	1.534,40 ₺	762,6%
OKTAY AYD.	6.101,70 ₺	525,00 ₺	-5.576,70 ₺	-91,4%
PEKTAS	0,00 ₺	240,00 ₺	240,00 ₺	
BLUEMOON	0,00 ₺	94,40 ₺	94,40 ₺	
DEMİRBAĞ	43.469,38 ₺	90,00 ₺	-43.379,38 ₺	-99,8%
SELDA	0,00 ₺	50,45 ₺	50,45 ₺	
NURAL	99,40 ₺	0,00 ₺	-99,40 ₺	-100,0%

Tablo 3.7 (devam): 2014-2015 Yılı Marka Ciro Dağılımı

ERSE	215,00 ₺	0,00 ₺	-215,00 ₺	-100,0%
HİMEL	360,80 ₺	0,00 ₺	-360,80 ₺	-100,0%
GÜNSAN	545,97 ₺	0,00 ₺	-545,97 ₺	-100,0%
ERBAG	885,78 ₺	0,00 ₺	-885,78 ₺	-100,0%
TOSH LED	1.734,60 ₺	0,00 ₺	-1.734,60 ₺	-100,0%

Markaların her iki yıla ait ciro dağılımlarını gösteren Tablo 3.7, 2015 yılı esas alınarak cirolar artandan azalana doğru sıralatılmıştır. Öne çıkan değişimler ilk dört marka arasında yer alan Legrand markasının 2015 yılında -3.483.061,03 TL ‘lik bir düşüşe geçtiği görülmektedir. Sırasıyla ilk dört marka içerisinde yer alan Nexans, Schneider, Hes, Legrand markaları arasında yalnızca Schneider markasının satışlarında (1.675.748,39TL) bir artış olduğu görülmektedir. En çok artışın BORSAN KABLO tarafından 1.890.557,11 TL’lik artış ile yapıldığı görülmektedir.

3.3. VERİNİN HAZIRLANMASI

Veri setinin analizlere hazır hale getirilmesi için uygulanan ön işlemede CRISP-DM sürecinin adımlarında yer aldığı üzere aşağıdaki sıra ile yer verilmiştir.

Kayıp Değerler

Veri setinde kayıp değer olup olmadığına bakılmış ve grupkod ve tipkod alanı boş bırakılmış olan 39 kayıt tespit edilmiştir. Grupkod ve Tipkod alanları kategorik alan türünde olduklarından, en çok tekrar eden alan ile doldurulması mümkündür. Ancak firma ile irtibata geçilerek bu kayıtların Aydınlatma Bölgesine ait satışlar olduğu tespit edilmiş olup, mevcut 39 adet kayıt Aydınlatma bölgesine dahil edilmiştir. Aydınlatma bölgesine ait Tipkod “Servet” olarak tespit edilmiş olup, Tipkod alanları da bu şekilde güncellenmiştir.

Birim alanı boş bırakılmış 3 kayıt tespit edilmiştir. Kategorik nitelik olan bu alanda, ürün türünü belirlememizde yardımcı olacak olan Kablo Türü alanı referans alınarak tamamlanmıştır. Kablo Türü genel olan ürünlerde birim niteliğinin adet olarak girildiği tespit edilmiş olup, en çok tekrarlanan olma özelliği göz önünde bulundurularak, birim

alanı boş bırakılmış 3 kayıt adet olarak tamamlanmıştır. Veri setindeki kayıp değerlerin tamamlanmasından sonra yapılan analizlerde bir değişiklik olmadığı tespit edilmiştir. Buna sebep olarak Aydınlatma bölgesinin küçük bir grubu temsil etmesinden dolayı, analizlere tesiri olmadığı düşünülmektedir.

Uç değerler

Sadece Net Tutar niteliği dolu olup, diğer tüm niteliklerin boş bırakıldığı bir kayıt veri setinden çıkarılmıştır. 183403 olan gözlem sayısı 183402' ye düşmüştür. Analizlerde bu gözlem sayısı ile çalışılmıştır.

Tekrar eden gözlem

Veri setinde tekrar eden gözlem olmadığı tespit edilmiştir. Bu nedenle veri setinden bu noktada her hangi bir çıkarım da bulunulmamıştır.

3.4. MODELLEME

Bu tez çalışmasında, firmanın satmış olduğu ürünler arasındaki birlikteliği ölçümlemek adına birliktelik kuralları adı altında yer alan Apriori, Eclat ve Naive Bayes algoritmalarından faydalanılmıştır. Algoritmaların performansını karşılaştırabilmek için, algoritmaların çalışma zamanları ve destek değerleri kıyaslanmış, grafik üzerine yansıtılmıştır. Algoritma ve performans sonuçlarına Bölüm 4.1'de yer verilmiştir.

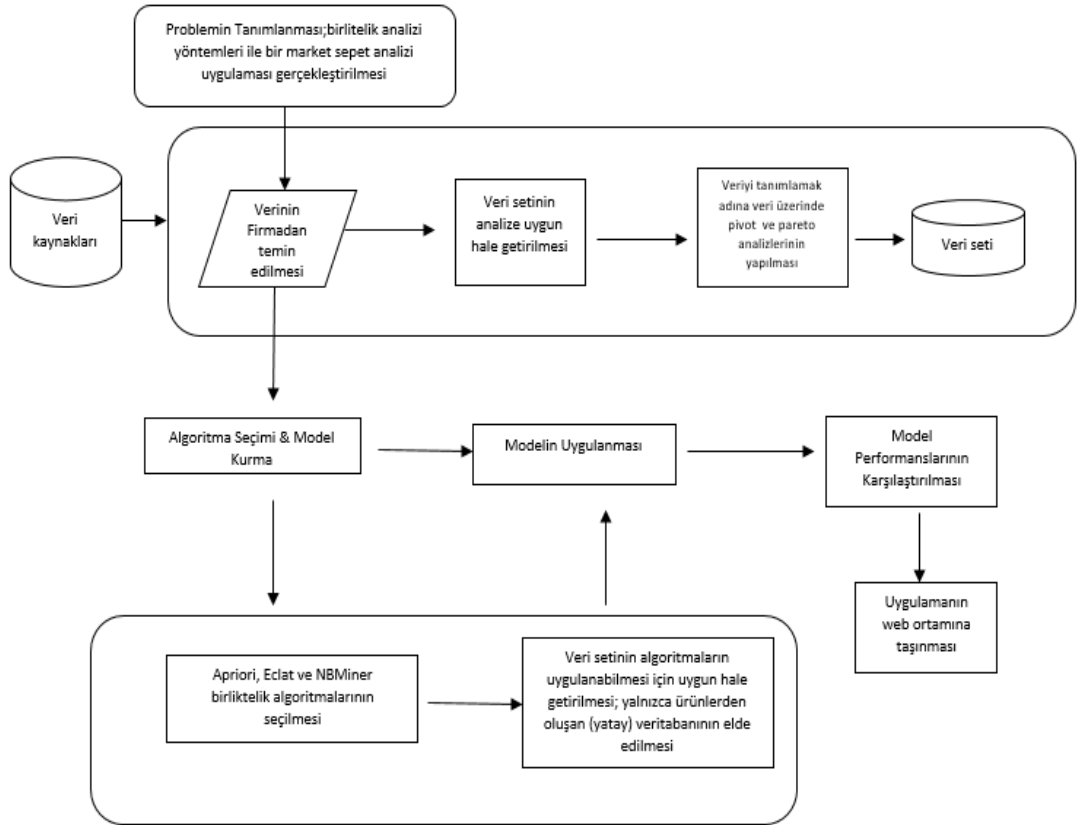
Analizler R programla dili ile gerçekleştirilmiştir. R istatistiksel hesaplama ve grafikleri için bilgisayar programı olup aynı zamanda programlama dilidir (rproject-2016). Yeni Zelanda Auckland Üniversitesinden Ross Ihaka ve Robert Gentleman tarafından ortaya çıkarılmıştır. Bir kullanıcı, R kaynak kodunu Genel Kamu Lisansı (*General Public License - GPL*) kapsamında yayınlatabilmektedir. Böylece, diğer kullanıcılar da bu koddan ücretsiz faydalanabilmektedir (Kartal, 2015). Bu tez çalışmasında R'nin görselleştirilmiş ve bütünleşik geliştirme ortamı olan Rstudio kullanılmıştır.

RStudio'nun iki sürümü mevcuttur: RStudio Desktop ve RStudio Sunucusu, RStudio Desktop programı düzenli bir masaüstü uygulaması olarak yerel olarak çalıştırmakta iken; RStudio Sunucusu uzaktan bir Linux sunucu üzerinde çalışırken bir web tarayıcısı

kullanarak RStudio'ya erişme imkan sağlamaktadır. RStudio, açık kaynak kodlu ya da ticari olarak kullanılabilir. RStudio, açık kaynak kodlu ya da ticari olarak kullanılabilir.

Tezde uygulanan apriori algoritmasının Shiny üzerinde uygulaması geliştirilmiştir. Shiny R'in web uygulama çerçevesidir (Rstudio, 2015a). Böylelikle geliştirilen modelin web üzerinde rahatlıkla son kullanıcı tarafından da uygulanabilmesi sağlanmıştır. Ayrıca çalışmada, veriyi tanımlama kısımlarında kullanılan pivot analizlerinde ve görsellerinde Microsoft Office 2016 Excel'den faydalanılmıştır.

Aşağıda çalışma boyunca yapılan adımlar Şekil 3.14'de bir akış şeması ile özetlenmiştir.



Şekil 3.14: Çalışmada izlenen adımları gösteren akış şeması.

4. BULGULAR

Tezin bu bölümünde, firmaya ait iki yıl verilerine (2014-2015) birliktelik kuralları algoritmalarından Apriori, Eclat, NBMiner uygulanmasına, analizlerin Shiny web ortamına taşınmasına ve tüm bu analizlerin sonucunda elde edilen bulgulara yer verilmiştir. Her iki yıla ait veri ile ilgili temel bir takım veri ön işleme işlemleri gerçekleştirildikten sonra, birliktelik analizleri her iki yılın verilerine uygulanmıştır. Excel dosyasındaki veriler, csv formatına dönüştürülerek RStudio programına aktarılmıştır. İlk olarak uygulamaya konulan apriori algoritması olmuştur. Uygulamanın ilk aşamasında, birliktelik algoritmasına; apriori algoritmasında kullanılmak üzere niteliklerin kategorik hale getirildiği veri setinin tüm değişkenleri dahil edilmiştir. Ancak asıl ölçümlenmek istenenin ürünler arasındaki birliktelik olmasından dolayı, veri setinde yer alan “mal kodları” ayrı bir excel dosyasına aktarılmıştır. Bu veriler algoritma dahilinde analiz edilebilecek hale gelebilmesi için bir takım işlemlerden geçirilmiştir.

4.1. APRIORI ALGORİTMASINDAN ELDE EDİLEN BULGULAR

Birliktelik algoritmasının gerçekleştirilebilmesi için temel şartlardan bir tanesi, aralarında birliktelik aranacak olan değişkene (ürün) ait kayıtların yan yana ya da bit matrix şeklinde yazdırılmış olmasıdır (Chaturvedi ve Gawande, 2014). Bu özellik bölüm 2.6.1’ de bahsedildiği üzere apriori algoritmasının temel özellikleri arasında yer almaktadır. Şöyle ki, bir faturada satın alınmış ürünlerin yan yana yer alıyor olması, alt satırda yeni işleme (faturaya) ait ürünlere yer verilmesi gerekmektedir. Böylelikle algoritma, her bir satırı bir transaction (işlem) olarak görmekte ve aday kümelerini bu doğrultuda oluşturmaktadır. Bu özellik, Apriori algoritmasında, veritabanının yatay olma şartının aranmasından kaynaklanmaktadır. Bu nedenle ürünler yer aldıkları fatura bazında, Excel formülleri kullanılarak şekildeki gibi yan yana yazılı hale getirilmiştir.

URUN_1	URUN_2	URUN_3	URUN_4	URUN_5	URUN_6	URUN_7	URUN_8	URUN_9	URUN_10
MGLV429	MGLV429	MGLV430	MGLV430	MGLV431	MGLV431	MGLV432	MGLV432	MGGVAE1	MGGV2M
MGLC1K1	MGA9C20	MGA9C20	MGA9C20	MGLC1D8	MGA9C20	MGA9C20	MGA9K24	MGA9K24	MGA9K24
SATGW10	SATGW10	SATGW10	SATGW10	SATGW10	SATGW10	SATGW10	SATGW10	SATGW10	SATGW10
MGA9K24	MGA9K24	MGA9K24	MGA9K24	MGA9K24	MGA9F74	MGA9N18	MGA9N18	MGA9N18	MGA9R50
MSNML01	MSNML31	MSPK5X6	MSPK5X6	MSPK5X6	MSPK5X1	MSPK5X1	PKSO803	MGA9K24	MGA9K24
BULEG107	BULEG107	BULEG104	BULEG104	BULEG109	BULEG109	BULEG109	BULEG143	BULEG328	BULEG333
MGXB4BD	MGZCKD0	MGLC1D0	MGLC1D1	MGLC1D1	MGLC1D3	MGZCKE0	MGZBE10	MGGVAE1	MGGVAN
BULEG312	BULEG371	BULEG371	BULEG371	BULEG372	BULEG375	MGGV2M	MGLC1D1	BULEG765	MGLRD14
BAKIRTOP	BAKIRTOP	BAKIRTOP	BAKIRLAM	BAKIRLAM	BAKIRLAM	BAKIRLAM	BAKIRLAM	BAKIRLAM	BAKIRLAM
BULEG770	BULEG109	BULEG412	MGLC1D1	MGLRD06	MGLRD07	BULEG105	BULEG104	BULEG104	BULEG105
MGA9K24	MGA9K24	MGA9K23	MGA9K24	MGGV2M	MGLC1K1	MGLP1K1	MGLR2K0	MGLR2K0	MGLC1D2
MGLC1D4	MGA9K24	MGA9K24	MGA9K24	MGA9F74	MGA9R41	MGA9R41	MGA9R41	MGA9R41	MGGV2M
PKSO253	PKSO253	PKSO403	PKSO603	PKSO103	MSNML01	MSNML31	MSNML03	MS3201-3	MSPK5X6
MGZCKD0	MGLT3SM	MGCAD32	MGLA4DA	MGLC1D1	MGLC1D1	MGLP1K0	MGLRD14	MGLRD16	MGZB4BH
MGA9V41	MGCA2KN	MGCA3KN	MGCAD32	MGCAD50	MGGV2P0	MGGV2P2	MGLA5FG	MGLADR0	MGLADR2

Şekil 4.1: Apriori Algoritmasında Kullanılan Veri Seti.

Birliktelik kuralları olaylar arasındaki probabalistik korelasyonu tanımlamaktadır. Olaylar arasındaki korelasyon sık sık beraber gözlenen olaylardır. Herhangi bir veri tabanında birliktelik kurallarının tanımlanması, veri tabanı bilgi keşfi sürecinin ilk adımıdır (De Alwis ve diğ., 2010).

Birliktelik kurallarıyla ilgili problem, birliktelik kurallarını belirlemede bir eşik değeri bulmaktır. İlginç birliktelik kurallarından ilginç olmayanları ayırt edebilmek için de bazı ölçütlerin belirlenmesi gereklidir. Bu ölçütler destek ve güven oranlarıdır.

Birliktelik kurallarının amacı, kullanıcı tarafından belirlenen minimum destek ve güven oranlarını sağlayan kuralların tespit edilmesidir. Anlamlılığı destek ve güven değerleri ile ölçülen birliktelik kuralları, "X nesnesini alan bir müşterinin muhtemelen Y nesnesini de alması" tipindeki kuralların tanımlanmasını amaçlamaktadır.

Bölüm 2.4.1' de detayları ile verilmiş olan destek oranı, veride bu bağıntının ne kadar sık olduğunu; güven oranı ise Y ürününü almış bir kişinin hangi olasılıkla X ürününü alacağını belirtmektedir. İki ürünün satın alınmasındaki bağıntının önemli olması için her iki değer de olabildiğince yüksek olması gerekmektedir (Timör ve diğ., 2011).

İlgilendiğimiz nokta, ürünlerin birlikte alınma olasılığı olduğundan dolayı, algoritmada güven değeri bizim öncelikli ölçütümüz olmaktadır. Ancak destek değeri de hesaba katılarak, belirlenen destek ve güven eşik değerleri sırası ile 2014 – 2015 yıllarına ait veri setlerine uygulanmıştır.

Aşağıda elde edilen birliktelik kuralları sırası ile paylaşılmıştır. 2014 yılına ait faturalarda yer alan ürünler arasında apriori algoritması uygulandığında elde edilen sonuçlara aşağıda sırasıyla yer verilmiştir:

Destek değeri = 0.01, güven değeri = 0.55, minlen = 2, maxlen = 10 olarak verildiğinde, (minlen ve maxlen ifadeleri, kuralda yer almasını istediğiniz öğelerin en az iki, en çok on olmasını belirtmemiz anlamına gelmektedir.)

```
kurallar <- apriori(veri, parameter = list(supp = 0.01, conf = 0.55, target = "rules", minlen=2, maxlen=10))
```

```
> inspect(kurallar)
```

	lhs	rhs	support	confidence	lift
[1]	{NEXHFNYM06}	=> {NEXHFNYM07}	0.01428483	0.7536705	29.47426
[2]	{NEXHFNYM07}	=> {NEXHFNYM06}	0.01428483	0.5586457	29.47426
[3]	{BULEG674420}	=> {BULEG77213}	0.01082184	0.7291667	26.00078
[4]	{HESNYA03-MAV}	=> {HESNYA03-SIY}	0.01242966	0.7416974	39.97996
[5]	{HESNYA03-SIY}	=> {HESNYA03-MAV}	0.01242966	0.6700000	39.97996
[6]	{KONKV105}	=> {KONKV102,5}	0.01394472	0.5804376	24.63584
[7]	{KONKV102,5}	=> {KONKV105}	0.01394472	0.5918635	24.63584
[8]	{KONKV105}	=> {KONKV110}	0.01518150	0.6319176	22.93769
[9]	{KONKV110}	=> {KONKV105}	0.01518150	0.5510662	22.93769
[10]	{KONKV102,5}	=> {KONKV110}	0.01314081	0.5577428	20.24525

Şekil 4.2: Apriori Algoritması ile 2014 Yılı Birliktelik Kuralları_1.

Elde ettiğimiz kuralları destek, güven ya da kaldıraç değerine göre sıralatabiliriz. Aşağıda güven değerine göre en iyi ilk 10 kural sıralatılmıştır:

```
inspect(head(sort(kurallar, by = "confidence"),10))
```

	lhs	rhs	support	confidence	lift
[1]	{NEXHFNYM06}	=> {NEXHFNYM07}	0.01428483	0.7536705	29.47426
[2]	{HESNYA03-MAV}	=> {HESNYA03-SIY}	0.01242966	0.7416974	39.97996
[3]	{BULEG674420}	=> {BULEG77213}	0.01082184	0.7291667	26.00078
[4]	{HESNYA03-SIY}	=> {HESNYA03-MAV}	0.01242966	0.6700000	39.97996
[5]	{KONKV105}	=> {KONKV110}	0.01518150	0.6319176	22.93769
[6]	{KONKV102,5}	=> {KONKV105}	0.01394472	0.5918635	24.63584
[7]	{KONKV105}	=> {KONKV102,5}	0.01394472	0.5804376	24.63584
[8]	{NEXHFNYM07}	=> {NEXHFNYM06}	0.01428483	0.5586457	29.47426
[9]	{KONKV102,5}	=> {KONKV110}	0.01314081	0.5577428	20.24525
[10]	{KONKV110}	=> {KONKV105}	0.01518150	0.5510662	22.93769

Şekil 4.3: Apriori Algoritması ile 2014 Yılı Birliktelik Kuralları_2.

Böylelikle elde edilen kural sayısı fazla olduğunda, ilginçlik katsayısı daha yüksek olan kurallara ulaşmanız kolaylaşmaktadır.

Destek değeri = 0.01 olarak sabitken, güven değeri = 0.65' a çıkarıldığında, toplamda dört kural elde edilmiş ve aşağıdaki sonuçlara ulaşılmıştır:

```
kurallar <- apriori(veri, parameter = list(supp = 0.01, conf = 0.65,
target ="rules", minlen=2, maxlen=10)
```

```
> inspect(kurallar)
```

	lhs	rhs	support	confidence	lift
[1]	{NEXHFNYM06}	=> {NEXHFNYM07}	0.01428483	0.7536705	29.47426
[2]	{HESNYA03-MAV}	=> {HESNYA03-SIY}	0.01242966	0.7416974	39.97996
[3]	{BULEG674420}	=> {BULEG77213}	0.01082184	0.7291667	26.00078
[4]	{HESNYA03-SIY}	=> {HESNYA03-MAV}	0.01242966	0.6700000	39.97996

Şekil 4.4: Apriori Algoritması ile 2014 Yılı, Birliktelik Kuralları_3.

Şekil 4.4' de elde edilen kurallar şu şekilde yorumlanmaktadır: NEXANS marka NEXHFNYM06 ürününü satın alanların %75'i NEXANS marka NEXHFNYM07 ürününü de satın almaktadır. Bu iki ürünü birlikte alanlar, toplam satışların %1'ini oluşturmaktadır. Bir başka deyişle, NEXHFNYM06 ve NEXHFNYM07 ürünlerinin birlikte görülme sıklığı %1'dir. Kaldıraç (lift) değeri 29 >1 olduğundan bu kuralın güvenilir olduğu söylenebilmektedir.

HES marka HESNYA03-MAV ürününü satın alanların %74'ü HES marka HESNYA03-SIY ürününü de satın almaktadır. Bu iki ürünü birlikte alanlar, toplam satışların %1'ini oluşturmaktadır. Bir başka deyişle, HESNYA03-SIY ve HESNYA03-MAV ürünlerinin birlikte görülme sıklığı %1'dir. Kaldıraç (lift) değeri 42 >1 olduğundan bu kuralın güvenilir olduğu söylenebilmektedir.

LEGRAND marka BULEG674420 ürününü satın alanların %72'si LEGRAND marka BULEG7723 ürününü de satın almaktadır. Bu iki ürünü birlikte alanlar, toplam satışların %1'ini oluşturmaktadır. Bir başka deyişle, BULEG674420 ve BULEG7723 ürünlerinin birlikte görülme sıklığı %1'dir. Kaldıraç (lift) değeri 26 > 1 olduğundan, bu kuralın güvenilir olduğu söylenilebilmektedir.

Algoritma parametrelerinden minlen maxlen değerleri değiştirilerek veri seti analize dahil edilmiştir. Kural diziliminde, minlen = 2 ölçütünü minlen = 3'e çıkardığımızda, kural diziliminde yer alan öge sayısının en az üç olmasını istediğimizi belirtmiş bulunuyoruz. İkili öge kümelerinin analizimizde yer almasını istemediğimizi belirterek, taranacak veri setini küçültmüş oluyoruz.


```
kurallar <- apriori(veri, parameter = list(supp = 0.01 ,conf = 0.55, target ="
rules", minlen=3, maxlen=10))
```

Bu şartları sağlayan birliktelik kurallarını mevcut olmadığı görülmüştür. Bu demek oluyor ki, oluşan kurallar yalnızca ikili öge kümelerini içermektedir. Bu nedenle minlen = 3 olduğu takdirde, sonuç boş küme olarak dönmektedir.

Destek değerini 0.002, güven değerini 0.75 olarak belirttiğimizde,

```
> kurallar <- apriori(veri, parameter = list(supp = 0.002,conf = 0.75, target
="rules", minlen=2, maxlen=10))
```

Sonucunda 489 kural oluşmuştur. Yaratılan kuralları, güvene göre sıralatıp, güven değeri en yüksek olan ilk 10 kuralı görüntülemek istediğimizde çıkan sonuçlar aşağıdaki gibidir:

```
> inspect(head(sort(kurallar, by ="confidence"), 10))
```

lhs	rhs	support	confidence	lift
{HESNYA02-SIY, HESNYA03-KAH, HESNYA03-KIR, HESNYA03-MAV}	=> {HESNYA03-SIY}	0.002288047	0.9487179	51.13906
{HESNYA02-KAH, HESNYA02-KIR, HESNYA02-SIY, HESNYA03-KAH, HESNYA03-KIR}	=> {HESNYA03-SIY}	0.002071610	0.9436620	50.86653
{HESNYA02-KIR, HESNYA02-MAV, HESNYA02-SIY, HESNYA03-KAH, HESNYA03-MAV}	=> {HESNYA03-SIY}	0.002071610	0.9436620	50.86653
{HESNYA02-KIR, HESNYA02-SIY, HESNYA03-KAH, HESNYA03-KIR}	=> {HESNYA03-SIY}	0.002380805	0.9390244	50.61654

Şekil 4.5: Apriori Algoritması ile 2014 Yılı Birliktelik Kuralları_4.

Çıkan kurallarda üçlü birlikteliklerin olduğu dikkat çekmektedir. İlk satırı incelediğimizde, Hes Kablo markasına ait HESNYA02-SIY, HESNYA03-KAH, HESNYA03-KIR, HESNYA03-MAV ürünlerini satın alanların %94'ünün HESNYA03-SIY ürününü de satın aldığı gözlemlenmektedir. Bu noktada getirilebilecek yorum, müşterilerin kablo alımları sırasında HESNYA03 ürünü satın alanların 4 farklı rengi birlikte satın aldığı görülmektedir. Bu üçlü ürün grubunun tüm satış hareketleri içerisindeki yüzdesine baktığımız zaman, % 0.2 olduğu görülmektedir. Bir kuralın diğer kural içerisinde tekrar ettiği durumlar olabilmektedir. Atık kural (redundant rules) denilen bu durumu ortadan kaldırmak için, kuralların budanması (prune) yoluna gidilmektedir. Böylelikle bir biri içerisinde tekrarlayan kurallar sonuçlara yansıtılmamaktadır. Atık kuralların budanması (pruning redundant rules) denilen bu işlem için gerekli R kodlarına aşağıda yer verilmiştir:

```
#atık kuralların budanması
subset.matrix <- is.subset(kurallar.sorted,kurallar.sorted)
subset.matrix[lower.tri(subset.matrix,diag=T)]<- NA
redundant <- colSums(subset.matrix,na.rm=T) >=1
which(redundant)
kurallar.pruned<-kurallar.sorted[!redundant]
inspect(kurallar.pruned)
```

Şekil 4.6: Atık Kuralların Budanmasına İlişkin Kullanılan Kodlar.

Destek değerini 0.002, güven değerini 0.75 olarak belirttiğimizde, atık kuralların budanması sonrasında elde edilen kurallara aşağıda yer verilmiştir.

	lhs	rhs	support	confidence	lift
[1]	{NEXHFNYM06}	=> {NEXHFNYM07}	0.01428483	0.7536705	29.47426
[2]	{HESNYA03-MAV}	=> {HESNYA03-SIY}	0.01242966	0.7416974	39.97996
[3]	{BULEG674420}	=> {BULEG77213}	0.01082184	0.7291667	26.00078
[4]	{KONKV105}	=> {KONKV110}	0.01518150	0.6319176	22.93769
[5]	{KONKV102,5}	=> {KONKV105}	0.01394472	0.5918635	24.63584
[6]	{KONKV102,5}	=> {KONKV110}	0.01314081	0.5577428	20.24525

Şekil 4.7: Apriori Algoritması 2014 Yılı Budanmış Birliktelik Kuralları.

Destek değişkenini 0.001' e düşürüp, güven' i 0.95'e çıkardığımızda elde edilen kural sayısı 14'e düşmektedir. Aşağıda ulaşılan kurallara yer verilmiştir.

	lhs	rhs	support	confidence	lift
1	{HESNYA02-KAH, HESNYA02-SIY, HESNYA03-KAH, HESNYA03-KIR, HESNYA03-MAV}	=> {HESNYA03-SIY}	0.001886092	0.9682540	52.19212
2	{HESNYA02-KAH, HESNYA02-KIR, HESNYA02-SIY, HESNYA03-KAH, HESNYA03-KIR, HESNYA03-MAV}	=> {HESNYA03-SIY}	0.001700575	0.9649123	52.01199
3	{HESNYA02-KAH, HESNYA02-MAV, HESNYA02-SIY, HESNYA03-KAH, HESNYA03-KIR, HESNYA03-MAV}	=> {HESNYA03-SIY}	0.001669656	0.9642857	51.97821

Şekil 4.8: Apriori Algoritması ile 2014 Yılı Budanmış Birliktelik Kuralları_2.

Şekil 4.8'den elde edilen İlk kurala baktığımızda, HESNYA02-KAH, HESNYA02-SIY, HESNYA03-KAH, HESNYA03-KIR ürünlerini satın alan müşterilerin %96 sının, HESNYA03-SIY ürününü de satın aldığı görülmektedir. Bu ürünleri birlikte alanlar, tüm satışların %0.01'ini oluşturmaktadır.

2015 verileri

Firmanın 2015 satış verilerine birliktelik algoritmalarından apriori algoritması uygulandığında, elde edilen sonuçlara aşağıda sırasıyla yer verilmiştir.

Destek değeri = 0.002, güven değeri = 0.55, minlen = 2 iken, birliktelik kuralı elde edilememiştir. Bu nedenle, destek değerinin de düşürülmesi yoluna gidilmiştir. Ancak kural oluşumu gözlemlenmediği için destek eşik değerini düşürme yoluna gidilmiştir. **Destek değeri 0.001, güven değeri 0.85** olarak belirlendiğinde, aşağıdaki kural elde edilmiştir:

lhs	rhs	support	confidence	lift
[1] {HESNYA02-MAV, HESNYA03-SIY}	=> {HESNYA03-MAV}	0.001132052	0.8547718	217.2392

Şekil 4.9: Apriori Algoritması ile 2015 Yılı, Birliktelik Kuralları_1.

HES marka kablolarda HESNYA02-MAV (02 türü kabloda mavi renk) , HESNYA03-SIY (03 türü kabloda siyah renk) satın alanların, 0.85'inin 03 türdeki kablolarda Mavi rengi tercih ettiği görülmektedir. Bu ürünleri birlikte alanlar tüm satışların % 0.1'ini oluşturmaktadır.

Destek değerini 0.002 olarak değiştirip, güven değerini 0.55 olarak belirlediğimizde, kural elde edilemediği görülmüştür. Bu nedenle destek değeri daha fazla düşürmeyerek 0.001'de sabitlenmiş, aşağıda yer verileceği üzere güven değeri üzerinde değişiklikler yapılmıştır: **Destek değeri sabitliğini korurken (0.001), güven değerini 0.75'e** düşürdüğümüzde kural sayısı 4'e çıkmaktadır:

lhs	rhs	support	confidence	lift
[1] {KONKV101,5, KONKV105}	=> {KONKV102,5}	0.001159529	0.8053435	166.5327
[2] {KONSİL102,5, KONSİL110}	=> {KONSİL105}	0.001022144	0.7782427	191.8932
[3] {HESNYA02-KAH, HESNYA03-MAV}	=> {HESNYA02-MAV}	0.001000162	0.8387097	232.2990
[4] {HESNYA02-MAV, HESNYA03-SIY}	=> {HESNYA03-MAV}	0.001132052	0.8547718	217.2392

Şekil 4.10: Apriori Algoritması ile 2015 Yılı Birliktelik Kuralları_2.

Bu değerler seçili iken budanmış kurallara bakmak istediğimizde (destek=0.001, güv=0.75)

lhs	rhs	support	confidence	lift
[1] {NEXHFNYM06}	=> {NEXHFNYM07}	0.01428483	0.7536705	29.47426
[2] {HESNYA03-MAV}	=> {HESNYA03-SIY}	0.01242966	0.7416974	39.97996
[3] {BULEG674420}	=> {BULEG77213}	0.01082184	0.7291667	26.00078
[4] {KONKV105}	=> {KONKV110}	0.01518150	0.6319176	22.93769
[5] {KONKV102,5}	=> {KONKV105}	0.01394472	0.5918635	24.63584
[6] {KONKV102,5}	=> {KONKV110}	0.01314081	0.5577428	20.24525

Şekil 4.11: Apriori Algoritması ile 2015 Yılı Budanmış Kurallar.

Yukarıdaki sonuçlara ulaşılmıştır. Burada göze çarpan; lhs(left hand side)'de yani kuralın öncülünü oluşturan kısımda ikili birlikteliklerin tek öğeye düşmüş olmasıdır. Aynı

zamanda daha dikkat çekici olan budanmış kuralların güven değerlerinin yükselmesi, tekrar eden kuralların çıkarılması beklenirken, Şekil 4.11’ de Şekil 4.10’a göre farklı sonuçlar elde edilmiştir. Bu nedenle budama algoritmasının bu veri seti için geçerliliğinden şüphe duyulmuş olup, Bölüm 4.4 ‘ de son kullanıcının müdahalesi altında çalışacak olan Shiny uygulamasında budama algoritmasına yer verilmemiştir.

Destek değeri = 0.001, güven değeri = 0.65 olarak belirlendiğinde toplam 7 kural elde edilmiştir. Budanmış kurallarına baktığımızda ise, 6 kural elde edilmiştir. Aşağıda sırası ile bu kurallara yer verilmiştir:

lhs	rhs	support	confidence	lift
[1] {KONKV101,5,KONKV102,5}	=> {KONKV105}	0.001159529	0.6677215	109.2676
[2] {KONKV101,5,KONKV105}	=> {KONKV102,5}	0.001159529	0.8053435	166.5327
[3] {KONSİL102,5,KONSİL110}	=> {KONSİL105}	0.001022144	0.7782427	191.8932
[4] {HESNYA02-KAH,HESNYA03-MAV}	=> {HESNYA02-MAV}	0.001000162	0.8387097	232.2990
[5] {HESNYA02-MAV,HESNYA03-SIY}	=> {HESNYA03-MAV}	0.001132052	0.8547718	217.2392

Şekil 4.12: Apriori Algoritması ile 2015 Yılı, Birliktelik Kuralları_2.

Destek değeri = 0.001, güven değeri = 0.65 budanmış kurallar :

lhs	rhs	support	confidence	lift
[1] {NEXHFNYM06}	=> {NEXHFNYM07}	0.01428483	0.7536705	29.47426
[2] {HESNYA03-MAV}	=> {HESNYA03-SIY}	0.01242966	0.7416974	39.97996
[3] {BULEG674420}	=> {BULEG77213}	0.01082184	0.7291667	26.00078
[4] {KONKV105}	=> {KONKV110}	0.01518150	0.6319176	22.93769
[5] {KONKV102,5}	=> {KONKV105}	0.01394472	0.5918635	24.63584

Şekil 4.13: Apriori Algoritması ile 2015 Yılı Budanmış Kurallar_2.

Destek eşik değeri = 0.001, güven değeri = 0.55 olarak belirlendiğinde toplamda 12 kural elde edilmiştir. Elde edilen kurallara yer verilmiştir:

lhs	rhs	support	confidence	lift
[1] {KONKV101,5,KONKV102,5}	=> {KONKV105}	0.001159529	0.6677215	109.2676
[2] {KONKV101,5,KONKV105}	=> {KONKV102,5}	0.001159529	0.8053435	166.5327
[3] {KONSİL102,5,KONSİL110}	=> {KONSİL105}	0.001022144	0.7782427	191.8932
[4] {KONSİL105,KONSİL110}	=> {KONSİL102,5}	0.001022144	0.5758514	164.2445
[5] {KONSİL102,5,KONSİL105}	=> {KONSİL110}	0.001022144	0.5705521	159.9748
[6] {HESNYA02-KAH,HESNYA02-MAV}	=> {HESNYA03-MAV}	0.001000162	0.5669782	144.0968

Şekil 4.14: Apriori Algoritması ile 2015 Yılı Birliktelik Kuralları_3.

Atık kuralları budama işlemi için gerekli kodları çalıştırdığımızda kural sayısı 12’den 6’ya düşmektedir.

lhs	rhs	support	confidence	lift
[1] {NEXHFNYM06}	=> {NEXHFNYM07}	0.01428483	0.7536705	29.47426
[2] {HESNYA03-MAV}	=> {HESNYA03-SIY}	0.01242966	0.7416974	39.97996
[3] {BULEG674420}	=> {BULEG77213}	0.01082184	0.7291667	26.00078
[4] {KONKV105}	=> {KONKV110}	0.01518150	0.6319176	22.93769
[5] {KONKV102,5}	=> {KONKV105}	0.01394472	0.5918635	24.63584
[6] {KONKV102,5}	=> {KONKV110}	0.01314081	0.5577428	20.24525

Şekil 4.15: Apriori Algoritması ile 2015 Yılı Birliktelik Kuralları_4.

Atık kuralların budanmasından sonra göze çarpan, destek değerlerinin yükselmesi olmuştur. 0.001 civarında olan destek değerleri, budama işleminden sonra 0.01'lere yükselmiştir. 0.85- 0.80 güven değerlerine sahip olan kuralların listeden çıktığı gözlemlenmektedir. Budanmış kurallarda çıkan ilk kurala baktığımızda şu şekilde yorumlanabilmektedir: NEXHFNYM06 ürününü satın alan müşterilerin %75'inin, NEXHFNYM07 ürününü de satın aldığı görülmektedir. Bu ürünleri birlikte alanlar, tüm satışların %0.014'ünü oluşturmaktadır.

Destek değerini 0.001'de sabit bırakıp, güven değerini 0.45'e düşürdüğümüzde, eklenen kurallara aşağıda yer verilmiştir:

lhs	rhs	support	confidence	lift
[13] {KONKV105,KONKV110}	=> {KONKV102,5}	0.001280427	0.5224215	108.02876
[14] {KONKV102,5,KONKV105}	=> {KONKV110}	0.001280427	0.4678715	76.70163

Şekil 4.16: Apriori Algoritması ile 2015 Yılı Birliktelik Kuralları_5.

Yukarıdaki kriterler için budanmış kurallara baktığımızda ise bir önceki parametrelerle elde edilmiş kuralların aynısının elde edildiği görülmektedir:

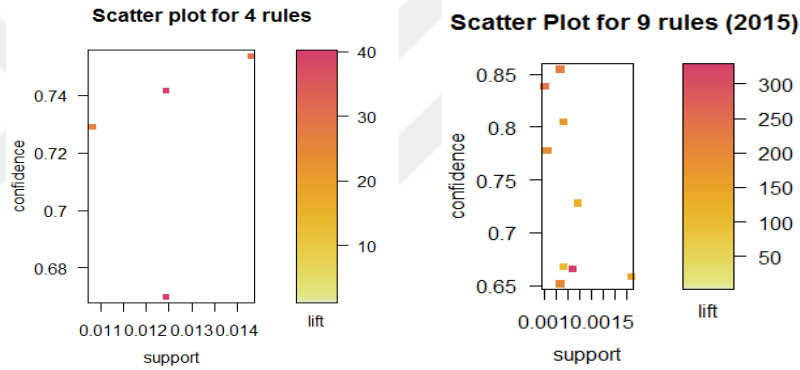
lhs	rhs	support	confidence	lift
[1] {NEXHFNYM06}	=> {NEXHFNYM07}	0.01428483	0.7536705	29.47426
[2] {HESNYA03-MAV}	=> {HESNYA03-SIY}	0.01242966	0.7416974	39.97996
[3] {BULEG674420}	=> {BULEG77213}	0.01082184	0.7291667	26.00078
[4] {KONKV105}	=> {KONKV110}	0.01518150	0.6319176	22.93769
[5] {KONKV102,5}	=> {KONKV105}	0.01394472	0.5918635	24.63584
[6] {KONKV102,5}	=> {KONKV110}	0.01314081	0.5577428	20.24525

Şekil 4.17: Apriori Algoritması ile 2015 Yılı Birliktelik Kuralları_6.

Güven değerini düşürdükçe kural sayısını arttığı rahatlıkla görülebilmektedir. Kuralların oluşturulması sırasında uygun destek değerini bulduktan sonra, güven değeri üzerinde değişiklikler yaparak elde edilen sonuçlar yorumlanmıştır.

Elde edilen birliktelik kuralların **arulesViz** paketinin `plot()` fonksiyonu kullanılarak görselleştirilmesi gerçekleştirilmiştir. `arulesViz` paketinde `plot()` fonksiyonu için, varsayılan method serpilme diyagramıdır. Serpilme diyagramında eksenlerde destek ve güven değerleri yer almaktadır. Buna ilaveten, üçüncü bir ölçü olarak noktalarının rengi ile temsil edilen kaldırmaç değeri kullanılabilir (Hashler ve Chelluboina, 2016-10-03). Renk skalasını gösteren cetvel grafiğin sağında yer almaktadır. İlk olarak 2014-2015 yıllarına ait grafiklere yer verilmiştir. Serpilme diyagramı ile ilgili koda aşağıda yer verilmiştir:

```
install.packages("arulesviz")
library(arulesviz)
kurallar <- apriori(veri, parameter = list(supp = 0.001 ,conf = 0.65, t
arget = "rules", minlen=2, maxlen=10))
plot(kurallar)
```

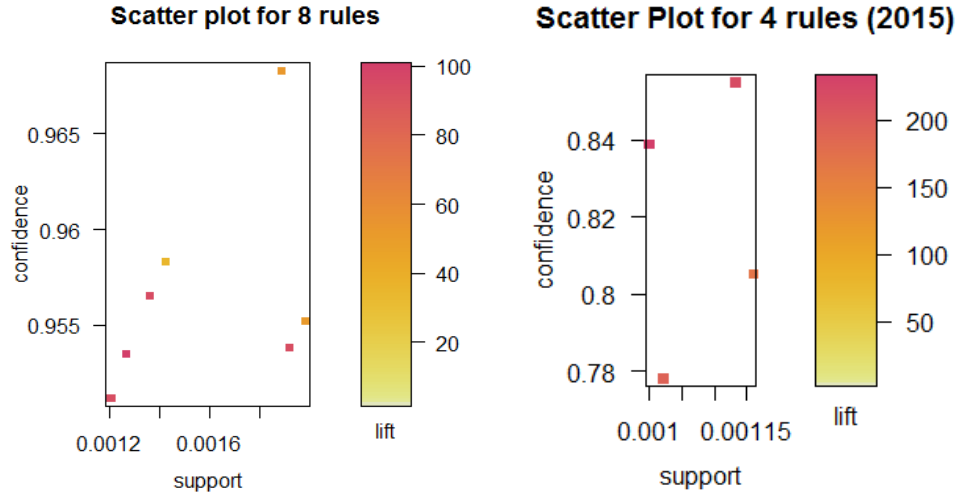


Şekil 4.18: Apriori Algoritmasında Destek Serpilme Diyagramları_1

Yukarıdaki grafikten, yüksek lift değerine sahip kuralların 0.012-0.013 destek değerleri etrafında toplandığı görülmektedir. Düşük kaldırmaç değerine sahip kuralların düşük destek değerine sahip oldukları gözlemlenmektedir. Güven değerlerinin ise, belirlendiği üzere 0.65 ile 0.75 arasında değiştiği görülmektedir.

Kural parametreleri destek=0.002, güven=0.75 olarak belirlendiğinde 489 kural elde edilmiştir.

```
kurallar <- apriori(veri, parameter = list(supp = 0.001 ,conf = 0.75, t
arget = "rules", minlen=2, maxlen=10))
subset.matrix <- is.subset(kurallar.sorted,kurallar.sorted)
subset.matrix[lower.tri(subset.matrix,diag=T)]<- NA
redundant <- colSums(subset.matrix,na.rm=T) >=1
which(redundant)
kurallar.pruned<-kurallar.sorted[!redundant]
inspect(kurallar.pruned)
plot(kurallar.pruned)
```

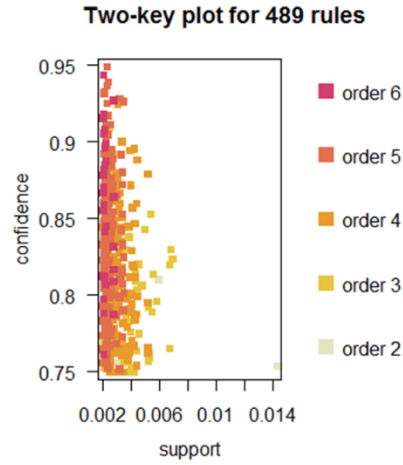


Şekil 4.19: Apriori Algoritmasında Serpilme Diyagramı_2.

Kurallar için kısıtlar destek=0.002, güven=0.75 olarak belirlendiğinde, elde edilen kural sayısını tekrar eden kuralları devre dışı bırakmak için bu kurallar budanma yöntemi ile indirgenmiştir. Budanma işlemi sonrası elde edilen kural sayısı 8' e düşmektedir. Yukarıda Şekil 3.19'da budanmış kurallara ait serpilme diyagramına yer verilmiştir.

Serpilme diyagramının farklı bir versiyonu olan two key plota destek değeri = 0.002, güven değeri = 0.75 değerleri ile Şekil 4.20'de yer verilmiştir. Two key plot diyagramında, eksenlerde destek ve güven değerleri yer alırken, noktaların renkleri ile "order" yani yaratılan kuralda yer alan ürünlerin sayısı temsil edilmektedir. Şekilden anlaşılacağı üzere, ürün sayısı ile destek değeri arasında ters yönlü bir ilişki mevcuttur (Hashler ve Chelluboina, 2016-10-03).

```
plot(kurallar, shading="order", control=list(main = "Two-key plot for 4
89 rules"))
```

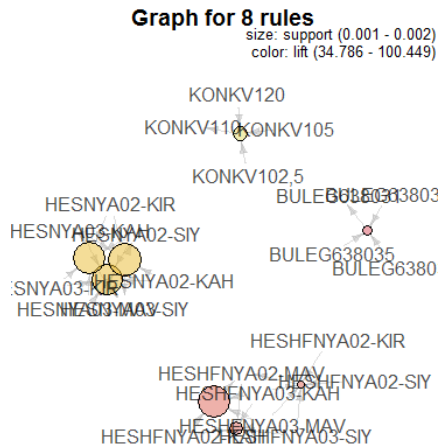


Şekil 4.20: Apriori Algoritmasında Two Key Diyagramı.

arulesViz paketinde yer alan Graf temelli diyagramlardan graf diyagramında kenar ve düğümlerden faydalanarak, kurallar görselleştirilmektedir. Graf temelli görselleştirmeler kuralların gösterimlerini oldukça açık ve okunaklı halde sunmaktadır. Graf diyagramlar daha çok küçük kural setleri için tercih edilmektedir. Şekil x.y’de kuralların hangi tekli öğelerden oluştuğu ve hangi kuralların hangi öğeleri paylaştığı görülebilmektedir.

Aşağıdaki Şekil 4.21’de yüksek lift değerlerine göre 10 kural için graf diyagramı uygulanmıştır.

```
subrules2 <- head(sort(kurallar, by="lift"), 10)
plot(subrules2, method="graph")
```



Şekil 4.21: Apriori Algoritmasında Graf Diyagramı.

4.2. ECLAT ALGORİTMASINDAN İLE ELDE EDİLEN BULGULAR

Birliktelik kurallarının üretilmesi için kullanılan bir diğer yöntem Eclat algoritması olarak bilinmektedir. Eclat algoritması Apriori algoritmasının çalışma biçimine benzemektedir (Özkan ve Erol, 2015).

Bu algoritma aşağıda belirtilen aşamalara sahiptir:

Destek ve güven ölçütlerini karşılaştırmak üzere eşik değerleri belirlenir. Uygulamadan elde edilen sonuçların bu eşik değerlere eşit ya da büyük olması beklenmektedir.

- Gözlemler tekli olarak taranarak her bir ürünün ait olduğu örnek numaraları belirlenmektedir.
- Yukarıdaki adımda seçilen ürünler ikişerli gruplandırılarak, bu grupların örnek numaraları belirlenmektedir.
- Bu kez üçerli, dörderli vb. gruplandırmalar yapılarak işlemler tekrarlanmaktadır.
- Ürün grubu belirlendikten sonra kural destek ölçütüne bakılarak birliktelikler belirlenmektedir.

RStudio' da Eclat algoritması ile birliktelik kuralları elde etmek için Apriori de olduğu gibi arules paketinden faydalanılmaktadır. Bu paket şu şekilde tanımlanmaktadır:

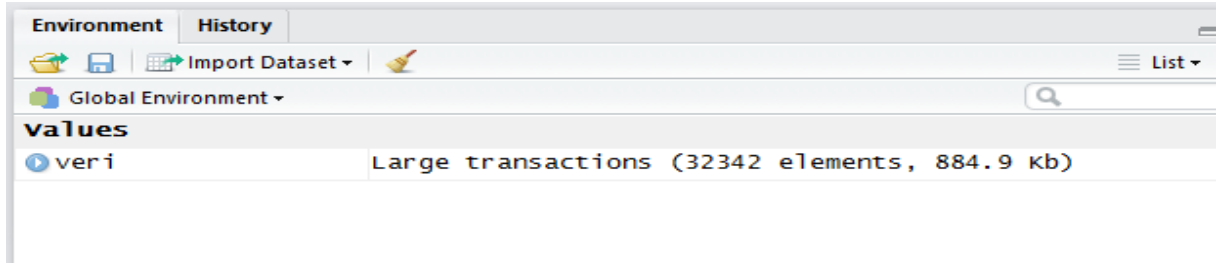
```
> library(arules)
```

2014 verilerini içeren veri seti kullanılarak, R ortamında uygulanan eclat algoritması ile birliktelik kuralları elde etmek için ilk olarak veri seti R ortamına tanıtılmaktadır:

```
veri <- read.transactions(file="D://Documents//R//workspace//asc_product.txt",rm.duplicates=TRUE, format="basket", sep="\t");
```

Şekil 4.22: Eclat Algoritmasında Verilerin R Ortamına Tanıtılması.

Veri setinin R ortamına sorunsuz bir şekilde tanıtıldığı R Environment' dan görülebilmektedir. “veri” isimli veri setinin 32342 işlem (transaction) ile R ortamına tanıtıldığı görülebilmektedir.



Şekil 4.23: Veri setinin R Ortamında Tanıtılması Sonrasında Görüntülenmesi.

ECLAT 2014 Yılı Bulguları

Veri `ec1at ()` fonksiyonunda kullanılarak birliktelik kurallarına ulaşılmaktadır. `ec1at ()` fonksiyonu içerisinde parametre olarak destek değeri ve minlen değerleri girilmektedir. Destek değeri ilk aşamada 0.001 olarak, oluşacak kuralların uzunluğunu (yer alacak öge sayısı) belirleyen minlen parametresi ise 2 olarak belirlenmiştir.

```
kurallar<-ec1at(veri, parameter = list(supp=0.001, minlen=2),control = NULL)
```

Belirlenen kriterler doğrultusunda elde edilen 1421 kural destek ölçütüne göre sıralanarak Şekil 4.24'de görüntülenmiştir. Görüntülenmesini istediğimiz kural sayısını elle belirlemek istersek; örneğin destek'e göre en iyi 10 kuralın görüntülenmesi için gerekli kod satırına yer verilmiştir:

```
> kurallar.sorted<-sort(kurallar,by="support")
> inspect(head(sort(kurallar, by ="supp"),10))
```

	items	support
[1]	{KONKV105, KONKV110}	0.015181498
[2]	{NEXHFNYM06, NEXHFNYM07}	0.014284831
[3]	{KONKV102, 5, KONKV105}	0.013944716
[4]	{KONKV102, 5, KONKV110}	0.013140808
[5]	{HESNYA03-MAV, HESNYA03-SIY}	0.012429658
[6]	{BULEG674420, BULEG77213}	0.010821842
[7]	{KONKV101, 5, KONKV102, 5}	0.009863336
[8]	{BULEG76561, BULEG77213}	0.009801497
[9]	{HESNYA02-SIY, HESNYA03-SIY}	0.009739657
[10]	{KONKV107, 5, KONKV110}	0.009708738

Şekil 4.24: Eclat Algoritması, 2014 Yılı Birliktelik Kuralları_1.

Yukarıda elde edilen kuralların yorumlanması şu şekilde yapılabilmektedir:

KONKV105, KONKV110 ürünlerinin tüm işlemler içerisinde birlikte görülme sıklığı % 1.5 'dir. NEXHFNYM06, NEXHFNYM07 ürünlerinin birlikte görülme sıklığı % 0.14'dür.

KONKV102, 5, KONKV105 ürünlerinin birlikte görülme sıklığı %1.4' dür. Aşağıda farklı destek değerlerinin eclat algoritmasındaki sonuçlarına yer verilmiştir:

Destek eşik değeri = 0.002 olarak girildiğinde elde edilen toplam 557 sonuç kümesinin bir kısmına aşağıda yer verilmiştir:

```
> inspect(head(sort(rules, by = "supp"), 10))
```

	items	support
[1]	{KONKV105, KONKV110}	0.015181498
[2]	{NEXHFNYM06, NEXHFNYM07}	0.014284831
[3]	{KONKV102, 5, KONKV105}	0.013944716
[4]	{KONKV102, 5, KONKV110}	0.013140808
[5]	{HESNYA03-MAV, HESNYA03-SIY}	0.012429658

Şekil 4.25: Eclat Algoritması, 2014 Yılı Birliktelik Kuralları_2.

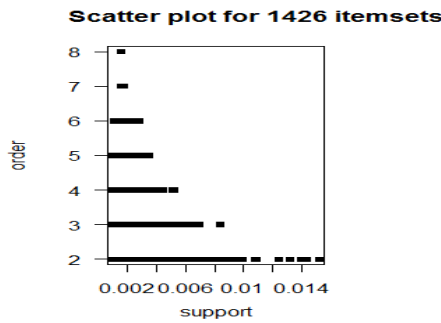
Destek eşik değerini yükseltip 0.01 olarak belirlediğimizde, oluşan kurallar kümesi 6'ya düşmüştür.

```
> kurallar<-eclat(veri, parameter = list(supp = 0.01, minlen = 2), control = NULL)
inspect(kurallar)
```

	items	support
[1]	{NEXHFNYM06, NEXHFNYM07}	0.01428483
[2]	{BULEG674420, BULEG77213}	0.01082184
[3]	{HESNYA03-MAV, HESNYA03-SIY}	0.01242966
[4]	{KONKV105, KONKV110}	0.01518150
[5]	{KONKV102, 5, KONKV105}	0.01394472
[6]	{KONKV102, 5, KONKV110}	0.01314081

Şekil 4.26: Eclat Algoritması, 2014 Yılı Birliktelik Kuralları_3.

Birliktelikler ile ilgili grafiği elde etmek üzere **arulesViz** paketinin `plot()` fonksiyonu kullanılmaktadır. Destek değeri = 0.001 iken elde edilen serpilme diyagramına Şekil 4.27'de yer verilmiştir.



Şekil 4.27: Eclat Algoritması, 2014 Yılı Serpilme Diyagramı_1.

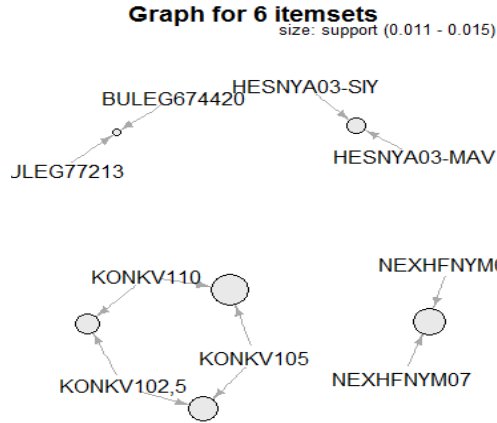
Şekil 4.27'deki grafik şu şekilde yorumlanabilmektedir: Destek değeri, 0.002-0.014 aralığında ikili ürün kümelerine, destek değeri 0.002 ve yaklaşık 0.01 aralığındaki iken, aynı zamanda üçlü ürün kümelerine, destek değeri 0.002 ve 0.006 arasındaki iken, aynı zamanda dördü ürün kümelerine rastlanmaktadır.

Serpilme diyagramını destek değerini değiştirerek farklı sonuçlar elde etmemiz mümkündür. Destek değeri 0.01 iken elde edilen 6 kurala ait grafiğe Şekil 4.28'de yer verilmiştir.



Şekil 4.28: Eclat Algoritması, Serpilme Diyagramı_2.

Destek eşik değeri 0.01 iken elde edilen 6 kurala ait graf diyagramına Şekil 4.29'da yer verilmiştir. Sadece iki öğeli kuralların elde edildiği görülmektedir. Graf diyagramlar daha çok küçük kural setleri için tercih edilmektedir. Şekil 4.29'da kuralların hangi tekli öğelerden oluştuğu ve hangi kuralların hangi öğeleri paylaştığı görülebilmektedir.



Şekil 4.29: Eclat Algoritması 2014 Yılı Graf Diyagramı.

ECLAT-2015 Yılı Bulguları

Firmanın 2015 yılına ait verilerine de Eclat algoritması aynı destek değerleri ile uygulanmıştır. Elde edilen sonuçlara aşağıda yer verilmiştir. Destek değeri = 0.001 verildiğinde; 42 kural elde edilmiştir. Destek değerine göre sıralatılan kurallara aşağıda yer verilmiştir.

```
> inspect(rules)
> rules.sorted<-sort(rules,by="support")
> inspect(rules.sorted)
> inspect(head(sort(rules,by="supp"),10)
```

	items	support
[1]	{NEXHFNYM06,NEXHFNYM07}	0.002934542
[2]	{KONKV102,5,KONKV105}	0.002736707
[3]	{KONKV105,KONKV110}	0.002450947
[4]	{HESNYA03-MAV,HESNYA03-SIY}	0.001906902
[5]	{BULEG77118,BULEG77213}	0.001862939
[6]	{KONSİL102,5,KONSİL105}	0.001791499
[7]	{KONSİL105,KONSİL110}	0.001775013
[8]	{HESNYA02-KAH,HESNYA02-MAV}	0.001764022
[9]	{KONKV102,5,KONKV110}	0.001758527
[10]	{KONKV101,5,KONKV102,5}	0.001736545

Şekil 4.30: Eclat Algoritması, 2015 Yılı Birliktelik Kuralları_1.

Aynı destek değeri ile 2014 verilerinden elde edilen birliktelik kurallarını incelediğimizde, aynı ürün grupları arasında yer alan destek ölçüsünün azaldığı göze çarpmaktadır. Örneğin 2014 yılında örnekteki ürün gruplarının birlikte satın alınma sıklıklarını ifade eden tablo aşağıdaki gibi iken,

2014 yılı		
	İtems	support
[1]	{KONKV105, KONKV110}	0.015181498
[2]	{NEXHFNYM06, NEXHFNYM07}	0.014284831

2015 yılında aynı ürün grubu arasında yer alan birliktelik destek ölçüsünün 0.015 ‘den 0.002’ ye düştüğü gözlemlenmektedir.

2015 yılı		
	İtems	support
[3]	{KONKV105, KONKV110}	0.002450947
[1]	{NEXHFNYM06, NEXHFNYM07}	0.002934542

Aynı şekilde, {NEXHFNYM06, NEXHFNYM07} ikilisi için, tüm işlemler arasında görülme sıklığını temsil eden destek değerlerinde 2015 yılına gelindiğinde bir düşüş olduğu gözlemlenmektedir. Bu durumda ürünler tek satın alınma ya da farklı ürün grupları ile birlikte satın alınma yoluna gidilmiş şeklinde yorum getirilebilmektedir.

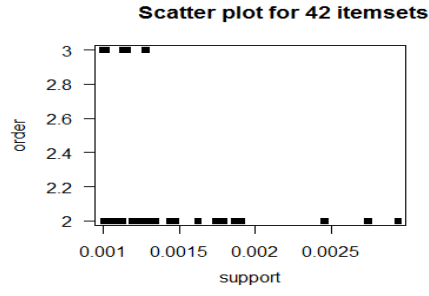
Destek değeri = 0.002 verildiğinde; 3 kural elde edilmiştir. Destek değerine göre sıralatılan kurallara aşağıda yer verilmiştir.

```
> rules.sorted<-sort(rules,by="support")
> inspect(rules.sorted)
  items                                support
[1] {NEXHFNYM06, NEXHFNYM07} 0.002934542
[2] {KONKV102, 5, KONKV105} 0.002736707
[3] {KONKV105, KONKV110} 0.002450947
```

Şekil 4.31: Eclat Algoritması, 2015 Yılı Birliktelik Kuralları_2.

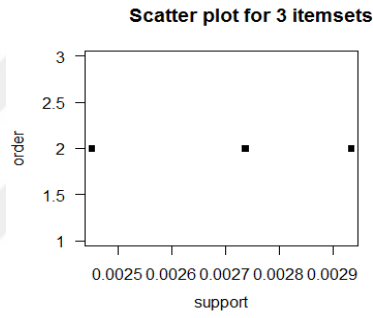
Destek değeri = 0.01 verildiğinde, 2015 yılına ait verilerde kural elde edilememiştir. Eclat algoritması için 2015 verilerine ait serpilme diyagramlarına aşağıda yer verilmiştir.

Destek değeri = 0.001 iken elde edilen 42 kural için serpilme diyagramına Şekil 4.32’de yer verilmiştir.



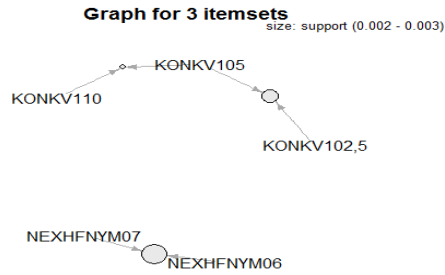
Şekil 4.32: 2015 Yılı Eclat Algoritması Serpilme Diyagramı_1.

Destek değeri = 0.002 iken elde edilen 3 kural için serpilme diyagramına Şekil 4.33'de yer verilmiştir.



Şekil 4.33: 2015 Yılı Eclat Algoritması Serpilme Diyagramı_2.

Destek değeri 0.002 iken 2015 verileri için elde edilen graf diyagramına Şekil 4.34'de yer verilmiştir.



Şekil 4.34: 2015 Verileri Eclat Algoritması Destek = 0.002 iken Serpilme Diyagramı.

4.3. NBMINER ALGORİTASINDAN ELDE EDİLEN BULGULAR

Birliktelik kurallarının üretilmesi için kullanılan bir diğer yöntem NBMiner algoritmasıdır. NBMiner algoritması ile birliktelik kuralları elde etmek için RStuido programında arulesNBMiner paketinden faydalanılmaktadır. İlgili paket ve kütüphane yüklendikten sonra, daha önce aşağıdaki şekilde tanıtılan veri seti tanımlanmıştır.

```
>install.packages(arulesNBMiner)
library(arulesNBMiner)
veri<-
read.transactions(file="D://Documents//R//workspace//asc_product.txt",
rm.duplicates=TRUE, format="basket", sep="\t");
```

Veri seti çok büyük olduğundan, NBMiner algoritmasındaki bir takım kısıtlar, algoritmanın daha hızlı çalışabilmesi adına kaldırılmıştır.

```
param <- NBMinerParameters(veri)
kurallar_NB <- NBMiner(veri, parameter = param, control = NULL)
inspect(head(kurallar_NB))
inspect(kurallar_NB)
```

	items	precision
[1]	{BULEG10521, BULEG77118, BULEG77213, BULEG78730}	1.0000000
[2]	{BULEG10722, BULEG10767, BULEG77213, BULEG78730}	1.0000000
[3]	{MGGV2ME16, MGGV2ME20, MGLC1D18M7, MGLC1D65AM7, MGLC1D80M7}	1.0000000
[4]	{BUWOT303457, BUWOT606174, BUWOT606193, BUWOT606198}	0.9999999
[5]	{BULEG420007, KONKV102, 5, KONKV105}	1.0000000

Şekil 4.35: NBMiner Algoritmasından Elde Edilen Birliktelik Kuralları.

Şekil 4.35’de NBMiner algoritmasının uygulanmasına ve sonuçlarına yer verilmiştir. Precision değeri, ürünlerin frekanslarını (tekrarlanma) temsil etmektedir.

4.4. ALGORİTMA PERFORMANSLARININ KARŞILAŞTIRMASI

Algoritmalarının performanslarını karşılaştırmak amacı ile çalışma zamanları ve destek değerleri göz önüne alınmıştır. NBMiner algoritmasında diğer iki algortmada ortak olarak yer alan destek değeri yer almadığı için performans karşılaştırmasına dahil edilmemiştir. Apriori ve Eclat algoritmalarına ait destek ve çalışma zamanlarını içeren tabloya aşağıda yer verilmiştir. Öncelikle 2014 veri setine ait sonuçlara yer verilmiştir. Çalışma zamanlarını gösteren alan kuralların yaratılması sırasında elde edilen “set transactions” alanından temin edilmiştir.


```

create itemset ...
set transactions ... [7129 item(s), 32342 transaction(s)] done [0.01s].
sorting and recoding items ... [735 item(s)] done [0.00s].
creating sparse bit matrix ... [735 row(s), 32342 column(s)] done [0.00s].
writing ... [1426 set(s)] done [0.18s].
Creating 54 object ... done [0.00s].

```

Şekil 4.36: Eclat Algoritması Kural Yaratılması.

```

set item appearances ... [0 item(s)] done [0.00s].
set transactions ... [7129 item(s), 32342 transaction(s)] done [0.02s].
sorting and recoding items ... [735 item(s)] done [0.00s].
creating transaction tree ... done [0.01s].
checking subsets of size 1 2 3 4 5 6 7 8 done [0.00s].
writing ... [1443 rule(s)] done [0.00s].
creating 54 object ... done [0.02s].

```

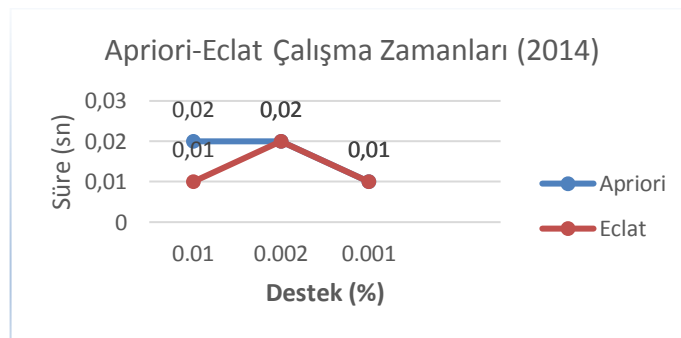
Şekil 4.37: Apriori Algoritması Kural Yaratılması.

2014 veriseti üzerinde belirlenen destek ve güven değerleri doğrultusunda kuralların yaratılması esnasında geçen süreler aşağıdaki Tablo 4.1’de saniye cinsinden belirtilmiştir.

Tablo 4.1: Apriori Eclat Çalışma Zamanları (2014 Yılı).

Destek Güven Değerleri		Toplam Çalışma Zamanı	
Güven	Destek	Apriori	Eclat
0.55	0.01	0.02s.	0.01s.
0.65	0.01	0.02s.	0.01s
0.65	0.001	0.01s.	0.01s.
0.65	0.002	0.02s.	0.02s.
0.75	0.002	0.02s	0.02s
0.95	0.001	0.01s	0.01s

2014 yılı veriseti üzerindeki algoritmaların çalışma zamanlarını gösteren grafiğe, Şekil 4.38’de yer verilmiştir.



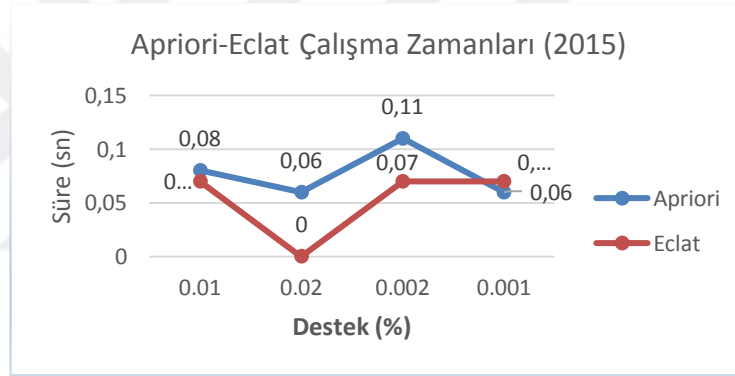
Şekil 4.38: Apriori Eclat Çalışma Zamanları (2014 Yılı).

Destek değerleri 0.01 ile 0.001 arasında verilmiş, bu kıstaslar arasındaki iken algoritmaların çalışma zamanları 0.00 sn ile 0.03 sn arasında değiştiği gözlemlenmektedir. Grafikte destek değeri 0.01 ile 0.002 arasında iken Eclat'ın daha kısa sürede cevap verdiği, yani çalışma zamanının daha düşük olduğu gözlemlenmektedir.

Tablo 4.2: Apriori Eclat Destek Değerleri ile Çalışma Zamanları (2015 Yılı).

Destek Güven Değerleri		Toplam Çalışma Zamanı	
Güven	Destek	Apriori	Eclat
0.75	0.01	0.08s.	0.07s.
0.75	0.02	0.06s	0s
0.75	0.002	0.06s	0.07s
0.75	0.001	0.11s	0.07s

Grafiksel gösterimi Şekil 3.40'daki gibidir:



Şekil 4.39: Apriori Eclat Çalışma Zamanları (2015 Yılı).

Performans analizi için çalışma zamanları metrik olarak kullanılmıştır. Destek değerleri 0.01 ile 0.001 arasında verilmiş, bu kıstaslar arasındaki iken algoritmaların çalışma zamanları 0.00 sn ile 0.77 sn arasında değiştiği gözlemlenmektedir. Grafikte görüldüğü üzere Eclat'ın çalışma zamanları Apriori'ye göre az farkla daha düşük olarak gözlemlenmiştir. Ancak Eclat algoritmasında güven kriterinin yer almıyor olması ve genellikle düşük boyutlardaki veri setlerinde daha başarılı sonuçlar elde ediliyor olması sebebi ile çalışmanın bundan sonraki kısmında Apriori algoritması ile devam edilme kararı alınmıştır.

4.5. UYGULAMANIN SHINY İLE WEB ORTAMINA TAŞINMASI

Shiny, R programlama dilini kullanarak web uygulamaları geliştirmemizi sağlayan bir çatıdır (shiny.rstudio.com). Bu uygulamaları geliştirirken HTML, CSS yada Javascript bilgisine ihtiyacınız olmamaktadır. Shiny sayesinde interaktif ve tepkisel web uygulamaları geliştirmek mümkün olmaktadır. Tepkisel kelimesi ile kast edilen, bir değer değiştiğinde değişen değere bağlı tüm değerlerin yeniden hesaplanmasıdır (CIO, 2016) Shiny uygulamalarını kendi donanımınız üzerinde çalıştırabileceğiniz gibi shinyapps.io sunucusunda yayınlanabilmektedir.

Shiny uygulamasında, apriori algoritması ile gerçekleştirilen birliktelik analizi web ortamına taşınmıştır. Böylelikle uygulamanın sadece analistler tarafından değil, kurum içerisinde yer alan ve analiz sonuçlarını kullanabilecek her çalışan için (saha yetkilisi, pazarlama departmanı çalışanları, departman yöneticileri, ürün, bölge temsilcileri vb.) program bilgisayarlarında yüklü olma şartı aranmaksızın, internet ve bir tarayıcı vasıtası ile her yerden ulaşılabilir ve uygulanabilir hale getirilmiştir.

Bölüm 2.4.2.1 Çok seviyeli birliktelik kurallarında anlatıldığı üzere farklı seviyeler için farklı destek, güven değerleri uygulayarak daha doğru sonuçlara ulaşmamız mümkündür. Shiny uygulaması sayesinde bu işlem kullanıcının tercihlerine bırakılarak mümkün kılınmaktadır. Kullanıcının analiz aşamasını geçerek sonuçlara hızlıca ve en doğru yoldan ulaşabilmesi hedeflenmiştir. Basit, sade kullanıcı dostu bir arayüz tasarlanarak, kullanıcının veri seti üzerinde filtreleme yapması sağlanarak, analizi yapmak istediği veri seti kendi tercihi bırakılmıştır.

Apriori algoritması yalnızca aralarında benzerlik aranacak öğelerin bulunduğu veri seti üzerinde çalıştırılmaktadır. Bu nedenle bir önceki bölümde apriori algoritması uygulanırken, yalnızca ürünlerin işlemler bazında yatay olarak yerleştirildiği veri seti ile çalışılmıştır. Shiny uygulamasında, kullanıcıya veri seti üzerinde filtreleme imkanı verildiği için, ürünlerin yanına diğer tanımlayıcı nitelikler Şekil 4.40'da olduğu gibi getirilmiştir. İlgili işlem sırası ile alakalı diğer niteliklerin yan yana getirilebilmesi için Ms. Excel 2016' da formüllerinden yararlanılmıştır.

	A	B	C	D	E	F	G	H	I	J	K
1	URUNLER	EVRAK	NC	AY	TEMSILCI	Bolge	CIRO_GRL	CARI_KOD	URUN_1	URUN_2	URUN_3
2	NEXHFNYI	IH420829	2015	SUBAT	IHRACAT	IHRACAT	B	7040430	MGLV4296	MGLV4296	MGLV4306
3	ERSEYSLYC	IH421955	2015	MAYIS	IHRACAT	IHRACAT	B	7040430	MGLC1K16	MGA9C20	MGA9C20
4	ERSEYSLYC	IH422220	2015	ARALIK	IHRACAT	IHRACAT	C	7050070	SATGW10	SATGW10	SATGW10
5	MGLC1D1	IH422171	2015	EKIM	IHRACAT	IHRACAT	A	7040578	MGA9K24	MGA9K24	MGA9K24
6	MGXB5AV	IH421971	2015	MAYIS	IHRACAT	IHRACAT	A	7040578	MSNML01	MSNML31	MSPK5X6
7	BULEG403	IH421880	2015	NISAN	IHRACAT	IHRACAT	C	7009060	BULEG107	BULEG107	BULEG104
8	ERSEPDPA	IH422096	2015	AGUSTOS	IHRACAT	IHRACAT	C	7000043	MGXB4BD	MGZCKD0	MGLC1D0
9	MGXB5AD	IH422183	2015	KASIM	IHRACAT	IHRACAT	C	7009060	BULEG312	BULEG371	BULEG371
10	BUWOT40	IH421910	2015	MART	IHRACAT	IHRACAT	A	7040578	BAKIRTOP	BAKIRTOP	BAKIRTOP

Şekil 4.40: Shiny Uygulamasında Sorgulanan Veri Seti.

Daha önce yıllık bazda tutmuş olduğumuz veri setleri, Shiny ortamında kullanıcıya verisetini seçme imkanı sağladığından, iki yıla ait veriler bir araya getirilmiştir. Öncesinde veri setinde yer almayan Ay niteliği tarihten türetilerek, çalışmaya dahil edilmiştir.

Kullanıcıya veri üzerinde, şu alanlar için filtreleme şansı verilmiştir: Yıl (Tümü/2014/2015), mevsimsel olarak satışları inceleyebilmek adına: Ay, bölgesel satışları inceleyebilmek adına; Bölge, Temsilci, Müşterileri ciro kategorilerine göre değerlendirmek ve aynı segmentteki müşterilerin satın alma alışkanlıklarını incelemek adına: Ciro Grubu, firmanın müşterilerini tek tek ele almak istemesi ya da satış temsilcilerinin kendi müşterilerinin yapmış olduğu tercihlere odaklanarak satın aldığı ürünler arasındaki benzerlikleri ortaya koyabilmek adına: Cari Kod yer almaktadır.

Belirlenmiş olan seçim alanları veriyi daha iyi temsil ettiği ve kategorik olmaları sebebi ile seçilmiş olup, ileri çalışmalarda tercihe bağlı olarak değiştirilebilir. Burada veri setine sonradan dahil olan nitelikler arasında Ciro Grubunun elde edilişi Bölüm 3' de ayrıntılı olarak ele alınmıştır.

Segmentasyon kendi özelliklerine göre farklılaştırılmış pazarlama stratejileri geliştirmek amacıyla farklı gruplar halinde müşteri tabanının ayrılma işlemi olarak belirtilmiştir. Bu çalışmada, müşteriler pareto analizi yardımı ile sene içerisinde yapmış oldukları cirolarına göre A, B, C olmak üzere sınıflara ayrılmıştır. Böylelikle veriyi, satışları belirlenen bu ciro grupları üzerinden de analiz etme imkanı doğmaktadır. Ciro gruplarının satın alma alışkanlıkları çıkarılarak, aynı grupta yer alıp X, Y, Z ürünlerini satın alan müşteri aynı grupta yer alıp yalnızca X, Y ürünlerini satın alan bir başka müşteri için referans oluşturmakta ve potansiyel Z ürünü alıcısı olarak düşünülebilmektedir.

Aşağıda Shiny ortamında geliştirilen Apriori Algoritmasına ait arayüzü bulunmaktadır.

Şekil 4.41: Apriori ile Geliştirilen Shiny Uygulaması Arayüzü.

Arayüz ile kullanıcıya veri seti ve algoritma üzerinde seçim yapma imkanları sağlanmaktadır: Veri seti üzerinde kullanıcı tercihlerini şu şekilde belirleyebilmektedir: Kullanıcı Yıl seçeneği ile elimizdeki 2014-2015 yıllarına ait satışları dilerse aylar, bölgeler ve bu bölgelerden sorumlu temsilciler bazında tek tek ya da “Tüm” seçeneğini seçerek incelenebilmektedir. Seçimler sırasında bağıl değişim esas alınmıştır. Örneğin Bölge Seçiniz alanından Perpa Bölgesi seçildiğinde, ilgili bölgeye ait temsilciler Rukiye ve Ümit olduğundan Temsilci seçiniz alanında yalnızca bu iki temsilci görüntülenebilmektedir. Ciro Grubu Seçiniz alanı sayesinde müşterileri daha önce gruplandırılmış olan seçenekler doğrultusunda cirolarına göre ayrı ayrı analiz edebilmektedir. Eğer kullanıcı bu ciro gruplarının dışında, hedef aldığı bir müşterinin alışverişlerini incelemek istediği takdirde, “Cari Kod Seçiniz” alanından müşterinin kod numarası seçilerek, yalnızca o müşteriye ait veriler getirilmektedir.

Apriori algoritması çerçevesinde, kullanıcının belirlemiş olduğu destek eşik değerine arayüzden erişim imkanı verilmiştir. Ekranda 0.001 olarak belirlenen minimum değer kullanıcı tarafından arttırılabilmektedir. Analizde esas alacağımız güven değerinin eşiği 0.5 olarak belirlenmiş olup, algoritma bilgisi olmaksızın sıradan bir kullanıcı açısından sade bir arayüz yaratılması adına bu değer program içerisinde belirlenmiş ve kullanıcı erişimine kapatılmıştır.

Yaratılan kuralların çok fazla olması durumunda program kullanıcıyı uarmakta ve destek değerini arttırması yönünde kullanıcıyı yönlendirmektedir. Aynı şekilde, belirlenen eşik değerleri sebebi ile kural elde edilemediği durumlarda, program kullanıcıyı destek eşik değerini düşürmesi yönünde uarmaktadır. Böylelikle güven değerine arayüzde yer vermeyerek, sıradan bir kullanıcının müdahale etmesi gereken alan sabitlemiştir.

Kullanıcının görüntüleyebileceği kural sayısı ilk aşamada on olarak belirlenmiş olup, kullanıcı bu sayıyı dilerse arttırıp azaltma imkanına sahiptir. Ürün Seçiniz alanı sayesinde, kullanıcı eğer bir ya da birkaç ürün üzerindeki birliktelik kurallarını incelemek isterse, bu alandan dilediği ürünleri listeden seçerek, bu ürün ya da ürünlerle ilgili kuralları görüntüleyebilmektedir. Aynı şekilde çok satan ürünlerin tespiti yapılarak, bu ürünlerin yanında satışı daha az gerçekleştirilen ürünler getirilerek, bu ürünlerin satışlarının arttırılması sağlanabilmektedir.

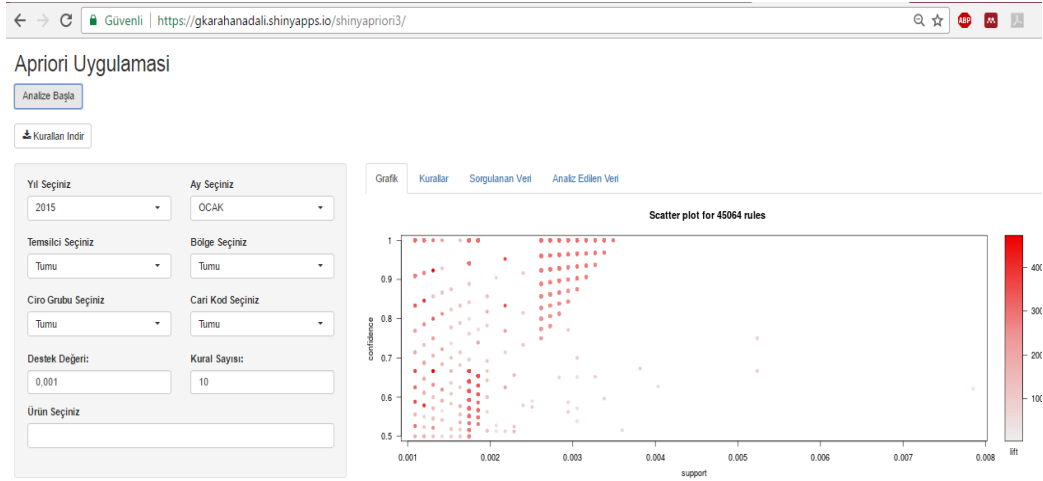
Kullanıcı sol panelden veri seti ve algoritma ile ilgili seçimlerini yaparak “Analyze Başla” diyerek ekranın sağ kısmında ayrılan Tabpanellerden yaratılan kurallara ait detayları ve seçimler doğrultusunda oluşturulan veri seti ile ilgili görüntüleme yapabilmektedir.

The screenshot shows the Apriori Uygulaması interface. On the left, there are configuration options for the analysis, including Year (2015), Month (OCAK), Terminals (Tumu), Region (Tumu), Sales Group (Tumu), Support Value (0.001), and Item (empty). The number of rules is set to 17. On the right, there are tabs for Grafik, Kurallar, Sorgulanan Veri, and Analiz Edilen Veri. The Kurallar tab is active, showing a list of 17 rules with their support and confidence values.

lhs	rhs	support	confidence	lift
[1] [V1PTEC88N0/85T1] => [PEKTASGRP01] @.001853467 1		305.7333		
[2] [V1PTEC88N0/85T1] => [PEKTASGRP09] @.001853467 1		305.7333		
[3] [V1PTEC88N0/85T1] => [PEKTASGRP04] @.001853467 1		286.6250		
[4] [V1PTEC88N0/85T1] => [PEKTASGRP18] @.001853467 1		295.8710		
[5] [V1PTEC88N0/85T1] => [PEKTASGRP11] @.001853467 1		295.8710		
[6] [V1PTEC88N0/85T1] => [PEKTASGRP22] @.001853467 1		295.8710		
[7] [PEKTASGRP09] => [PEKTASGRP18] @.003278824 1		295.8710		
[8] [PEKTASGRP09] => [PEKTASGRP11] @.003278824 1		295.8710		
[9] [PEKTASGRP09] => [PEKTASGRP22] @.003278824 1		295.8710		
[10] [PEKTASGRP08] => [PEKTASGRP05] @.003379852 1		295.8710		
[11] [PEKTASGRP08] => [PEKTASGRP06] @.003379852 1		295.8710		
[12] [PEKTASGRP08] => [PEKTASGRP16] @.003379852 1		286.6250		
[13] [PEKTASGRP08] => [PEKTASGRP15] @.003379852 1		286.6250		
[14] [PEKTASGRP08] => [PEKTASGRP16] @.003379852 1		286.6250		
[15] [PEKTASGRP08] => [PEKTASGRP15] @.003379852 1		286.6250		
[16] [PEKTASGRP08] => [PEKTASGRP07] @.003379852 1		286.6250		
[17] [PEKTASGRP10] => [PEKTASGRP11] @.003379852 1		295.8710		

Şekil 4.42: Shiny Uygulamasında Yaratılan Kuralların Görüntülenmesi.

Kullanıcının görüntüleyebileceği alanlarda ilk olarak dağılım grafiğine yer verilmiştir. Grafik üzerindeki noktalar kuralları temsil ediyor olup, X ekseninde destek, Y ekseninde güven değerleri yer almaktadır. Grafiğin dışında yer alan renk skalasında kaldıraç değerlerinin renk skalasına yer verilmektedir. Nokta ile gösterilen kuralların rengi ne kadar koyu ise kaldıraç değeri o kadar yüksek yani ilgili kuralın güvenilirliği ve performansı bir o kadar iyi olduğu anlamına gelmektedir.



Şekil 4.43: Shiny Uygulamasında Yaratılan Kurallara Ait Grafiğin Görüntülenmesi.

İkinci sekmede kurallara yer verilmiştir. Kullanıcının sol panelde yapmış olduğu seçimler doğrultusunda yaratılan kurallar buradan görüntülenebilmektedir. Kullanıcı dilerse, yaratılan kuralları “Kuralları İndir” diyerek, kuralları web ortamından kendi bilgisayarına indirme şansına sahiptir. Görüntülenebilecek kural sayısı sınırlı olduğundan, kullanıcıya böyle bir seçenek sunulmuştur.

Üçüncü sekmede, “Sorgulanan Veri” alanı alanında, kullanıcının veri seti için sol panelden yapmış olduğu tercihler doğrultusunda filtrelenilerek elde edilen yeni veri setine yer verilmektedir (Şekil 4.40). Ancak algoritmaya verinin bu hali değil, ürünle ilgili tanımlayıcı diğer bilgilerden arınarak yalnızca işlemlerde yar alan ürünlerin yan yana yazılmış hali sunulmaktadır. Bu nedenle, “Analiz Edilen Veri” sekmesi ile, “Sorgulanan Veri” de yer alan ilk yedi sütunun çıkarılması ile elde edilmiş olan ürünlere yer verilmiştir

	A	B	C	D	E	F	G	H	I	J
1	URUN_1	URUN_2	URUN_3	URUN_4	URUN_5	URUN_6	URUN_7	URUN_8	URUN_9	URUN_10
2	MGLV429€	MGLV429€	MGLV430€	MGLV430€	MGLV431€	MGLV431€	MGLV432€	MGLV432€	MGGVAE1	MGGV2ME04
3	MGLC1K1€	MGA9C20	MGA9C20	MGA9C20	MGLC1D8€	MGA9C20	MGA9C20	MGA9K24	MGA9K24	MGA9K24125
4	SATGW10	SATGW10	SATGW10	SATGW10	SATGW10	SATGW10	SATGW10	SATGW10	SATGW10	SATGW10051
5	MGA9K24	MGA9K24	MGA9K24	MGA9K24	MGA9K24	MGA9F74	MGA9N18	MGA9N18	MGA9N18	MGA9R50240
6	MSNML01	MSNML31	MSPK5X6€	MSPK5X6€	MSPK5X6€	MSPK5X12	MSPK5X12	PKSO803	MGA9K24	MGA9K24116
7	BULEG107	BULEG107	BULEG104	BULEG104	BULEG109	BULEG109	BULEG109	BULEG143	BULEG328	BULEG33327
8	MGB4BD	MGZCKD0	MGLC1D0	MGLC1D1	MGLC1D1	MGLC1D3	MGZCKE0	MGBE10	MGGVAE1	MGGVAN11
9	BULEG312	BULEG371	BULEG371	BULEG371	BULEG372	BULEG375	MGGV2M	MGLC1D1	BULEG765	MGLRD14
10	BAKIRTOP	BAKIRTOP	BAKIRTOP	BAKIRLAW	BAKIRLAW	BAKIRLAW	BAKIRLAW	BAKIRLAW	BAKIRLAW	BAKIRLAWA5010

Şekil 4.44: Shiny Uygulamasında Analiz Edilen Veri Seti.

Yapılan çalışmada, her iki yılı kapsayacak veri seti üzerinde destek = 0.001 güven = 0.5 eşik değerleri ile analiz edildiğinde, aralık ve ocak ayları haricinde tüm aylara yayılmış olarak HES MARKA kablonun ürünlerinde gözle görülür oranda bir satış birlikteliği dikkati çekmektedir. Aralık ve ocak ayları incelendiğinde {PEKTASGRP09} => {PEKTASGRP10} ürün ikilisinin destek değeri = 0.001, güven değeri = 1, kaldıraç değeri = 887.8000 olarak gözlemlendiğinden bu iki ürünün müşterilere paket halinde sunulması uygun görülmüştür. Bu kural şu şekilde yorumlanmaktadır: PEKTASGRP09 ürününü alanların hepsi (güven), PEKTASGRP10 ürününü de satın almaktadır. Bu iki ürünü birlikte satın alanlar tüm alışverişlerin %0.1'ini (destek) oluşturmaktadır. Kaldıraç değeri 887.800 > 1 olduğundan bu kuralın güvenilir, performansının yüksek olduğu söylenebilmektedir. Ürün tercihlerinde, yıl sonu aralık ayı ve yıl başlangıcı ocak aylarında PEKTAS ürün grubundaki satışlarda görülen birliktelik yoğunluğu net satış tutarları da dikkate alınarak incelendiğinde, diğer önde gelen markaların yıl sonunda ki satışlarının öncesine göre düşüş göstermesi, firmanın eldeki stokları bitirmeye çalışması olarak yorumlanabilir. Piyasadaki büyük oyuncuların geri çekilmesi ile diğer markalar bu aylarda ön plana çıkmaktadır. Tüm veri üzerindeki aylık incelemeden yola çıkılarak şubat ayında karşılaşılan, önerilebilecek bir başka ürün paketi aşağıdaki tablodaki gibidir:

lhs	rhs	support	conf	lift
{HESHFNVA03-KIR, HESHFNVA03-MAV} => {HESHFNVA03-SIY}	0.001200624	1	333.1600	
{HESHFNVA02-MAV, HESHFNVA03-KIR, HESHFNVA03-MAV} !=> {HESHFNVA03-SIY}	0.001020531	1	333.1600	

Şekil 4.45: Shiny Apriori Uygulaması Kurallar_1.

Tablonun 1. Satırında gösterilen HESHFNVA03 ürünün kırmızı ve mavi renklerini satın alanların, aynı ürünün Siyah rengini de aldıkları, bu ürünlerin birlikte görüldüğü işlem sayısının, tüm işlemlerin % 0.1'ni oluşturduğu görülmektedir. Kaldıraç değeri 333.1600 > 1 olması bu kuralın güvenilir ve performansının yüksek olmasını göstermektedir. Buradan hareketle, HESHFNVA03 ürününe ait (kırmızı, mavi, siyah) renklerinden oluşan bir renk paleti birlikte sunulabileceği önerisi getirilebilmektedir. Aynı şekilde, çapraz bir ürün paketi tabloda ikinci sırada görüntülenen, HESHFNVA02 HESHFNVA03 ürünleri arasında önerilebilmektedir.

Yapılan çalışmada, A (en yüksek), B (orta), C (en düşük) olarak ayrılan müşteri segmentlerini kendi içerisindeki birlikteliği ölçümlmek adına yıl ayrımı gözetilmeksizin tüm veri seti üzerinde analiz edildiğinde aşağıdaki kurallara ulaşılmıştır:

A ciro grubu için elde edilen 100 kural içerisinde güven değerlerine göre sıralatıldığında, en yüksek ilk 5 kural aşağıdaki gibidir. HESHFNYA02-siyah/yeşil ve HESHFNYA03-MAVi

	lhs	rhs	support	confidence	lift
[1]	{HESHFNYA02-S/Y, HESHFNYA03-MAV}	=> {HESHFNYA02-MAV}	0.001086494	0.9200000	246.5251
[2]	{KONKV101, KONKV101, 5, KONKV105}	=> {KONKV102, 5}	0.001086494	0.9200000	142.1568
[3]	{HESHFNYA02-MAV, HESHFNYA03-S/Y}	=> {HESHFNYA03-MAV}	0.001369928	0.9062500	204.0894
[4]	{KONKV101, 5, KONKV110}	=> {KONKV102, 5}	0.001180972	0.8928571	137.9627
[5]	{KONKV101, KONKV102, 5, KONKV105}	=> {KONKV101, 5}	0.001086494	0.8846154	237.0433

Şekil 4.46: Shiny Apriori Uygulaması Kurallar_2.

ürününün satın alanlar %92 (güven) olasılıkla HESHFNYA02-MAVi ürününü de satın almaktadır. Bu üçlü ürün paketini satın alanlar tüm satın alma işlemlerinin % 0.1'ini oluşturmaktadır. İkinci kuralı ele alacak olursa, Kondaş markasına ait KONKV101, KONKV101, 5, KONKV105 ürününü alanların, %92 olasılıkla, KONKV102, 5 ürününü de satın almaktadır. A ciro grubu, müşterilerin üst segmentini oluşturmaktadır. Üst müşteri grubunun Kondaş ve Hes markası üzerinde yoğunlaştığı görülmektedir. B ciro grubu için elde edilen 362 kural içerisinde güven değerlerine göre sıralatıldığında, en yüksek ilk 5 kural aşağıdaki gibidir:

	lhs	rhs	support	confidence	lift
[1]	{HESNYA02-MAV, HESNYA03-KAH, HESNYA03-S/Y}	=> {HESNYA03-MAV}	0.001315315	0.9182390	129.8402
[2]	{HESNYA02-KAH, HESNYA03-KIR, HESNYA03-MAV}	=> {HESNYA02-MAV}	0.001036036	0.9126984	145.3508
[3]	{HESNYA02-MAV, HESNYA03-S/Y, HESNYA03-SIY}	=> {HESNYA03-MAV}	0.001279279	0.9044586	127.8916
[4]	{HESNYA02-MAV, HESNYA03-KIR, HESNYA03-S/Y}	=> {HESNYA03-MAV}	0.001171171	0.8965517	126.7736
[5]	{HESNYA02-KAH, HESNYA02-KIR, HESNYA03-MAV}	=> {HESNYA02-MAV}	0.001027027	0.8906250	141.8355

Şekil 4.47: Shiny Apriori Uygulaması Kurallar_3.

Orta seviye ciro grubunu temsil eden B ciro grubunda, Hes marka ürünlerin tercih edildiği ve renk paletleri kullanılarak ürün paketleri oluşturulabileceği görülmektedir. HESNYA02-MAV, HESNYA03-KAH ve HESNYA03-Sarı/Yeşil ürünlerini satın alanların %91 olasılıkla HESNYA03-MAV ürününü de satın aldıkları görülmektedir. Bu dörtlü ürün paketini satın alanlar tüm satın alma işlemlerinin % 0.1'ini oluşturmaktadır. C ciro grubu için elde edilen 306 kural içerisinde güven değerlerine göre sıralatıldığında, en yüksek ilk 5 kural Şekil 4.48'deki gibi elde edilmiştir.

lhs	rhs	support	confidence
[1] {HESNYA02-MAV, HESNYA02-SIY, HESNYA03-S/Y}	=> {HESNYA03-MAV}	0.001225217	0.8953488
[2] {KONKV101, 5, KONKV105, KONKV107, 5}	=> {KONKV102, 5}	0.001066098	0.8933333
[3] {HESNYA02-MAV, HESNYA03-S/Y, HESNYA03-SIY}	=> {HESNYA03-MAV}	0.001193393	0.8928571
[4] {HESNYA02-KAH, HESNYA02-MAV, HESNYA03-KAH, HESNYA03-S/Y}	=> {HESNYA03-MAV}	0.001002450	0.8873239
[5] {HESNYA02-SIY, HESNYA03-S/Y, HESNYA03-SIY}	=> {HESNYA03-MAV}	0.001113834	0.8860759

Şekil 4.48: Shiny Apriori Uygulaması Kurallar_4.

Hes Marka marka kabloların kendi arasındaki birliktelik şu şekilde yorumlanmaktadır: HESNYA02-MAV, HESNYA02-SIY, HESNYA03-S/Y ürünlerini satın alanların %89.5'i HESNYA03-MAV ürününü de satın almaktadır. Bu dörtlü ürün grubunu satın alanlar, tüm satışların % 0.1'ini oluşturmaktadır.

Veri seti yıl bazında ele alındığında, 2014 ve 2015 yılları arasında göze çarpan fark;

2015 yılında Legrand Marka BULEG638038 ürününde görülen birliktelik olmuştur. HES Kablo ürünlerine ait hakimiyet 2015 yılında bu ürünle kırılmıştır. BULEG638038, BULEG674420 ürünlerini alanların %89 olasılıkla BULEG77213 ürününü de aldığı gözlemlenmektedir.

Firmanın çalışmış olduğu 18 bölge ayrı ayrı analize dahil edilmiştir. Sıralama olarak en yüksek ciroya sahip Karaköy, Trakya2 ve Trakya1 bölgeleri ele alındığında, çıkan sonuçlar şu şekildedir:

2014 ve 2015 yıllarında toplam en yüksek ciroya sahip Karaköy bölgesinde, Hes kablunun ağırlığı görülmektedir. HESHFNYA02 ve HESHFNYA03 ürünleri arasındaki güçlü birliktelik dikkat çekmektedir. HESHFNYA02-KAH, HESHFNYA03-KAH, HESHFNYA03-MAV ürünlerini satın alanların HESHFNYA02-MAV ürününü de mutlaka (%100) satın aldıkları görülmektedir. Bu dörtlü ürün paketini satın alanlar, tüm satışların %0.1'ini oluşturmaktadır. Kaldıraç değeri 185.0291 olan bu kuralın güvenilirliği yüksektir diyebiliriz. Bu iki üründen oluşan renk paleti haline getirilerek müşterilere sunulabilir ve bu paket yanında başka ürünler sunularak müşterilerin sunulan bu ek ürünleri satın almaları sağlanabileceği öngörülmektedir.

Ciro sıralamasında ikinci sırada yer alan Trakya2 bölgesi incelendiğinde, Kondaş marka ürün grubu arasındaki birliktelik dikkat çekmektedir. Elde edilen kurallar güven değerine göre sıralatıldığında, ilk sırada gelen kuralı yorumlamak gerekirse, KONKV101,5

KONKV105, KONKV107,5 ürünlerini satın alanların KONKV 102,5 ürününü de satın aldıkları görülmektedir. Bu dörtlü ürün paketini satın alanlar, tüm satışların % 0.1'ini oluşturmaktadır.

Ciro sıralamasında üçüncü sırada yer alan Trakya1 bölgesi incelendiğinde, Hes marka kablolarda HESHFNYA02 ve HESHFNYA03 ürünleri arasında bir renk paleti oluşturulması uygun görülmüştür.

Seçimler bu şekilde çoğaltılabilmektedir. Shiny uygulaması ile elde edilmek istenen amaçlardan birisi de, kullanıcının seçmiş olduğu veriseti üzerinde daraltma (filtreleme) yaparak elindeki veriyi derinlemesine analiz etme imkanı sağlamaktır.

5. TARTIŞMA VE SONUÇ

İçinde bulunduğumuz bilgi çağında, verinin çeşitli enformasyonlar geçirerek, kullanılabilir anlamlı bilgi haline gelmesi önemlilik arz etmektedir. Artan veri miktarı, veriye erişim kanallarının giderek artması ve aynı oranla veri erişim hızının da artması, büyük veri kavramının doğuşunu beraberinde getirmiştir. Büyük veri doğru bir şekilde ele alınıp, doğru enformasyonlardan geçirilerek analiz edildiğinde, firmaların stratejik kararlarını alma aşamasında büyük ölçüde yardımcı rol oynamaktadır. Kurumların memnun müşteri potansiyelini elde tutmaya çalışırken geliştirdikleri müşteriye göre ürün devrini başlatan CRM stratejileri büyük veriden beslenmektedir. Firmaları rakiplerinden sıyrılarak öne çıkaran, büyük veriye sahip olmaktan çok, doğru analizler sayesinde işe yarar anlamlı sonuçlar çıkarabilmektir. Veri madenciliği, CRM stratejisi kurabilmek için hayati önem taşıyan müşteri anlayışını oluşturabilmek için gerekmektedir. Müşterinin satın alma tercihlerinin belirlenerek, müşteriye özel ürün seçeneklerinin oluşturulması, mevcut müşterinin tutundurulmaya çalışılarak, yeni müşteri potansiyelinin elde edilmesi şirket kârını arttırıcı rol oynamaktadır.

Bu tez çalışmasında, elektrik sektöründe faaliyet gösteren bir firmaya ait 2014-2015 yıllarını içeren gerçek satış verilerine ulaşıp, bu veriler üzerinde birliktelik kurallarına ait algoritmalar uygulanarak, müşterilerin birlikte satın almayı tercih ettikleri ürünler tespit edilip, müşteri ilişkileri yönetimine ilişkin bir uygulama gerçekleştirilmiştir. Öncelikle 2014-2015 yılları ayrı ayrı ele alınarak her iki yıla ait sonuçlar değerlendirilmiştir. 2014 yılına ait veri, 19 nitelik ve 177393 kayıttan oluşmaktadır 2015 yılına ait veri, toplamda 23 nitelik ve 183401 kayıttan oluşmaktadır. Firma 2015 yılında, kayıtların saklanması aşamasında getirdiği bir takım yenilikler sebebi ile nitelik sayısında artış olmuştur. Nitelikler her iki yıl göz önünde bulundurularak düzenlenmiştir.

Analiz aşamasında CRISP-DM süreci adım adım izlenmiştir. Bu sürecin veriyi anlama aşamasında, veri seti üzerinde, en kritik olduğu düşünülen ve dağılımını incelemek istediğimiz nitelikler üzerinde 2014 – 2015 yılları olarak ayrı ayrı pareto analizleri uygulanmıştır. Pareto analizi sırasında Ms. Excel'den faydalanılmıştır. Bu analiz mevcut

veri setinde müşteri ciroları, marka ve bölgeler üzerinde uygulanmıştır. Böylelikle toplam ciroda önemli role sahip olan müşteri, marka ve bölgelerin ortaya çıkarılması hedeflenmiştir. Buradan elde edinilen sonuçlar birliktelik algoritmasını uygulama aşamasında yol gösterici rol üstlenmektedir.

Veriyi anlama aşamasında Ms. Excel 2016 pivot analizlerinden faydalanılmıştır. Veri seti içeriğinde yer alan, markalar, bölgeler, satış temsilcileri, müşteri ciroları açısından detaylıca ele alınmıştır.

Bu uygulama kapsamında, elde edilen bilgiler ışığında, firmanın müşterilerine özel satış stratejileri, ürün kampanyaları geliştirmesi, mevcut müşterilerini tutundurması amaçlanmıştır. Bu çalışma literatürde birliktelik kuralları kullanılarak yapılan diğer çalışmalardan şu yönleri ile farklılık göstermektedir.

Çalışmada, veri setinde müşterilerin ciro grubu bilgilerine yer verilmemiş olduğundan, öncelikle eldeki veri setinde yer alan net birim tutar alanı müşteri bazında toplanarak müşteri ciroları elde edilmiştir. Müşteri cirolarının toplam ciroya olan etkisinden yola çıkarak, pareto analizi aracılığı ile müşteri ciro grupları 80-20 kuralı çerçevesinde üç sınıf olarak elde edilmiştir (Bölüm 3.2). Yorumlama kısmında, elde edilen bu kriterden faydalanılmıştır. Aynı şekilde veri setinde ay bilgisine de yer verilmemiş olup, paylaşılan tarih bilgisi üzerinden ay detayı temin edilmiştir. Bu bilgi ile verinin daha ayrıntılı olarak incelenmesi amaçlanmıştır. Satışlar üzerinde ayların bir etkisinin olup olmadığı araştırılmış, çok net bir etkinin olmadığı ancak Haziran ve Eylül aylarındaki hareketlenmeler göze çarpmaktadır. Haziran ayında 2014'den 2015' e pozitif yönde 2.074.819,78 ₺' lik bir artış olurken, Eylül ayında negatif yönde 2.432.525,96 ₺' lik bir azalma olduğu görülmüştür. Firma yetkileri ile de paylaşılan bu durumun, özellikle inşaat firmalarına elektrik malzemeleri satan firma için inşaat sektörünün gelmiş olduğu gelişmişlik düzeyinin katkısı olduğu tespit edilmiştir. Günümüzde her yeni gün başlayan bir proje ile inşaat sektörü neredeyse on iki ay boyunca canlı kaldığı görülmektedir.

Tez kapsamında birliktelik kurallarını elde edebilmek için Apriori algoritması temel alınarak, Eclat ve NBMiner algoritmaları kullanılarak farklı modeller elde edilmeye çalışılmıştır. Apriori algoritması ile özellikle perakende sektöründe sıklıkla başvuru alan bir market sepet analizi örneği gerçekleştirilmiştir. Apriori ve Eclat algoritmalarından

elde edilen birliktelik kurallarında ortak olarak destek değeri yer aldığından bu iki algoritma arasında yürütme zamanları baz alınarak performans karşılaştırılması 2014 ve 2015 yıllarına ait veri setleri üzerinde ayrı ayrı yapılmış, sonuçlara Tablo 4.1 ve Tablo 4.2’ de yer verilmiştir. Eclat algoritmasının az sayılabilecek zaman farkı ile analizleri önde tamamladığı tespit edilmiştir. Ancak bunda güven değerini göz ardı edip, sadece destek değeri ile analizleri gerçekleştiriyor olması göz ardı edilmemelidir. Bu nedenle Apriori algoritmasından elde edilen sonuçların daha güvenilir olduğu görüşü kabul edilmiştir.

NBMiner algoritması kural çıktısı olarak yalnızca kesinlik (precision) değerine yer verdiği için, Apriori ve Eclat algoritmaları ile performans karşılaştırmasına dahil edilmemiş olup, sadece bulgularına yer verilmiştir.

Tezde uygulanan Apriori Algoritmasının Shiny üzerinde uygulaması geliştirilmiştir. Shiny R’in web uygulama çerçevesidir (Rstudio, 2015a). Böylelikle geliştirilen modelin, program bağımlılığı olmaksızın, web üzerinden rahatlıkla son kullanıcı tarafından da uygulanabilmesi sağlanmıştır.

Çalışma kapsamında yapılan tüm analizler R programlama dili ile gerçekleştirilmiştir. R dili, 20 yılı aşkın süredir kullanılan ve geliştirilen açık kaynak kodlu bir script dilidir. İstatistiksel hesaplamalar, tahmine dayalı analizler, veri görselleştirme ve raporlama alanlarında sıklıkla tercih edilmektedir. Açık kaynak kodlu olması sayesinde sürekli geliştirilebilir olmasını sağlamaktadır. En son geliştirilen yöntemleri izler ve yapısında taşır. Kullanıcılar tarafından geliştirilen çok sayıda fonksiyon ve pakete sahiptir (DataLabTR, 2015). Bu tez çalışmasındaki R kodlarının geliştirme aracı olarak RStudio’nun açık kaynak kod ürünü kullanılmıştır.

Bu tez çalışmasının literatürdeki çalışmalar arasından öne çıkan bir diğer özelliği ise, Apriori Algoritması ile geliştirilen birliktelik kurallarının shinyapps.io aracılığı ile web üzerinden herkes tarafından erişilebilir ve uygulanabilir hale getirilmiş olmasıdır. Shiny, R kodları ile desteklenen ve geliştiricilerin web üzerinden daha geniş kitlelere ulaşmasını sağlayan, R uygulamalarının web üzerinden gerçekleştirilmesini sağlayan RStudio uygulamasıdır.

Böylelikle herhangi bir yazılım bilgisi olmaksızın firma yetkilileri ya da ilgili son kullanıcı tarafından ürün analizleri gerçekleştirilebiliyor olacaktır. Kullanıcıya sunulan arayüz sayesinde, veri setine müdahale etme şansı ortaya çıkmaktadır. Kullanıcı seçim yapmış olduğu değerler doğrultusunda veriyi tekrar tekrar şekillendirmektedir. Böylelikle uygulama dinamik bir hal almıştır. Kullanıcı iki yıla dağılmış olan veri setinden dilediği yılı seçebilir ya da tüm yılları kapsayacak şekilde analiz yapabilmektedir.

Veride yer alan 18 farklı bölgeyi Shiny sayesinde tek tek incelemek mümkün hale gelmektedir. Belirlenmiş ciro gruplarına ait ürün satışlarındaki birliktelik, yapılan seçimler doğrultusunda incelenebilmektedir. Çok daha spesifik bir alana yoğunlaşarak, seçilmiş bir satış temsilcisinin bölgesindeki aylık satış verileri incelenebilmektedir. Belirli bir ürünle ilgili çalışma yapılmak istenildiğinde kuralda yer alan bir ürünle ilgili oluşmuş tüm kuralları incelemek mümkün hale getirilmiştir. Oluşturulan kuralların ekranda görüntülenme sayısı kullanıcının tercihine sunulmuştur. Yaratılan kuralları daha ayrıntılı görüntülenebilmesi ve incelenebilmesi açısından, kullanıcının kendi bilgisayarına indirebilme seçeneği sunulmuştur. Apriori algoritmasında kullanıcı tarafından bir defa belirlenen destek eşik değeri dinamik hale getirilmiştir. Yaratılan kuralların analiz edilemeyecek kadar çok ya da az olması durumunda, kullanıcı destek eşik değerini değiştirmesi gerektiği hususunda yönlendirilmiştir.

Elde edilen bulgularda özellikle en çok rastlanan HES marka kablo satışlarındaki birliktelik dikkat çekmektedir. Bu markaya ait HESNYA-03 ve HESNYA-02 ürünleri arasında gerçekleşen çapraz satış ve farklı renk seçeneklerinin birlikte satılıyor olması ürünü ön plana çıkarmaktadır (Şekil 4.45). Bu sonuçtan yola çıkarak ürün ile ilgili çeşitli renk paketleri oluşturularak, müşterinin bu ve benzeri paketler yanında yeni ürünler sunularak, daha az tercih edilen ürünleri satın alması sağlanabileceği düşünülmüştür.

Shiny uygulaması üzerinden yıllar, bölgeler, aylar, temsilciler, ciro grupları doğrultusunda seçimler yaparak elde edilen birliktelik kuralları yorumlamada esas aldığımız güven değerine göre sıralatılarak görüntülenmiştir. Elde edilen bulguların detaylarına Bölüm 4.5' de yer verilmiştir.

Bu çalışma doğrultusunda, ileride yapılması planlananlara sırasıyla yer verilmiştir

Web üzerinde farklı algoritmaları çalıştırması, yeni verileri dışarıdan okutarak çalıştırılabilir olması, verinin algoritmanın istediği formata program içerisinde dönüştürülüyor olması, performans karşılaştırması aşamasında daha çok algoritmanın farklı açılardan ele alınarak kıyasının yapılması.

Pareto analizi ile gerçekleştirilen müşteri segmentasyonunun kümeleme algoritmaları kullanılarak gerçekleştirilmesi ve müşterileri kredibiliteleri açısından risk derecelerine göre ele alınması planlanmaktadır.

Bu tez çalışması kapsamında Shiny uygulamasında analizler yıllık ve aylık bazda yapılmıştır. Daha detaylı sonuçlar elde edilmek istenildiği takdirde, analizlerin haftalık ve günlük bazda yapılması önerilmektedir.

Sonuç olarak, veri madenciliği konularından birliktelik analizi kullanılarak olası ürün paketlerinin seçilen bir müşteri veya müşteri grupları üzerinde de uygulanabilirliği sağlanmıştır. Geliştirilen web arayüzüyle yaratılan modelin arka yüzde çalıştırılması uygulamaya mekan ve zamandan bağımsız dinamik bir özellik katılmasını sağlamıştır. Bu tez çalışmasında; müşteri ve ürün analizlerinin yapılabildiği ve rekabet üstünlüğü sağlanabileceği düşünülen dinamik bir model geliştirilerek, literatüre önemli bir katkı sağlandığı düşünülmektedir. Verinin toplanmasından, satışların artırılmasına yönelik geliştirilen bu modelde, web üzerinde R modelinin çalıştırılarak, müşterilere ve satışlara yönelik daha hızlı kararların alınabileceği öngörülmektedir.

Bu çalışmanın sektörde uygulanabilir hale taşınıp, farklı modeller eklenerek, veri madenciliğinin farklı alanlarında da hayata geçirilebilir olması amaçlanmaktadır. Gerçek veri üzerinde geliştirilen birliktelik analiz modelimizin, iş hayatında kullanılarak, değer yaratması ve akademik çalışmalara temel oluşturması en büyük temennimizdir.

KAYNAKLAR

- Agrawal, R. and Shafer, J.C., 1996, "Parallel mining of association rules: Design, Implementation and Experience", IBM Research Report RJ 10004.
- Agrawal, R. and Srikant, R. (1994) 'Fast algorithms for mining association rules', *Proceedings of the 20th International Conference on Very Large Databases*, pp. 487–499.
- Agrawal, R., Imielinski, T. and Swami, A. (1993) 'Mining association rules between sets of items in large databases', *ACM SIGMOD Record*, 22(May), pp. 207–216. doi: 10.1145/170036.170072.
- Akpınar, H., 2000, Veri Tabanlarında Bilgi Keşfi ve Veri Madenciliği, *İstanbul Üniversitesi İşletme Fakültesi Dergisi*, 29(1), 1-22.
- Akpınar, H., 2014, *Data Veri Madenciliği, Veri Analizi*, 1. bs, Papatya, İstanbul, ISBN: 978-605-4220-81-6.
- Alpaydın E., (2000), Zeki Veri Madenciliği: Ham Veriden Altın Bilgiye Ulaşma Yöntemleri, *Bilişim2000*, 1-5.
- Apte, C., Weiss, S. M., Apte, C. and Weiss, S. (1997) 'Data Mining with Decision Trees and Decision Rules', *Computer Systems*, 13(November), pp. 197–210.
- Ay, D. and Çil, İ. (2010) 'Migros türk a.ş.de birliktelik kurallarının yerleşim düzeni planlamada kullanılması', pp. 14–29.
- Ayık ve diğ., 2007, Lise Türü Ve Lise Mezuniyet Başarısının, Kazanılan Fakülte İle İlişkisinin Veri Madenciliği Tekniği İle Analizi, *Atatürk Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 10(2), 443-447.
- Balaban, M. E., (2015), Project Management in Data Mining, PM Summit 2015, 2-3 Ekim 2015, Boğaziçi Üniversitesi, İstanbul Türkiye.
- Balaban, M. E., Kartal, E., 2015, *Veri Madenciliği ve Makine Öğrenmesi*, Çağlayan Kitabevi, ISBN: 978-975-436-089-9.
- Balasubramanian M., Selvarani, M., 2014, Churn Prediction in Mobile Telecom System Using Data Mining Techniques, *International Journal of Scientific and Research Publications*, 4 (4), 1-5.
- Berry, M. J. A. and Linoff G.S., 2004, *Data Mining Techniques For Marketing, Sales and Customer Relationship Management*, Canada, Wiley, ISBN: 0-471-47064-3.

- Bhargavi, P. and Jyothi, S. (2009) ‘Applying Naive Bayes Data Mining Technique for Classification of Agricultural Land Soils’, *IJCSNS International Journal of Computer Science and Network Security*, 9(8), pp. 117–122.
- Birant, D., Kut ve diğ., (2010) ‘İş Zekası Çözümleri için Çok Boyutlu Birliktelik Kuralları Analizi’, pp. 215–222.
- Borgelt, C. (2003) ‘Efficient Implementations of Apriori and Eclat’, *FIMT 03: Proceedings of the IEEE ICDM workshop on frequent itemset mining implementations*, (April). doi: 10.1016/j.cor.2003.09.014.
- Borgelt, C. (2005) ‘An implementation of the FP-growth algorithm’, *Proceedings of the 1st international workshop on open source data mining frequent pattern mining implementations - OSDM '05*, pp. 1–5. doi: 10.1145/1133905.1133907.
- Borgelt, C., (2003), *Find Frequent Item Sets with the Eclat Algorithm*, http://www.borgelt.net/doc/eclat/eclat.html#Borgelt_2012, [Ziyaret tarihi: 4 Eylül 2016].
- Bounsaythip, C. ve Runsala, R., Overview of Data Mining for Customer Behavior Modelling, 2001, VTT, Information Technology, Research Report, TTE1-2001-18.
- Brin, S., Motwani, R., Ullman, J. D. ve Tsur, S. (1997) ‘Dynamic itemset counting and implication rules for market basket data’, *ACM SIGMOD Record*, 26(2), pp. 255–264. doi: 10.1145/253262.253325.
- Carpenter, J., Bartlett, J. ve Kenward, M., 2015, *Introduction to missing data*, http://missingdata.lshtm.ac.uk/index.php?option=com_content&view=section&id=7&Itemid=96, [Ziyaret Tarihi: 20.04.2016].
- Chang, W., Cheng, J., Allaire, J. J., Xie, Y. ve McPherson, J., 2015, *Shiny: Web Application Framework for R*, <http://CRAN.R-project.org/package=shiny>, [Ziyaret Tarihi: 02.06.2015].
- Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C. and Wirth, R. (2000) ‘Crisp-Dm 1.0’, *CRISP-DM Consortium*, p. 76. doi: 10.1109/ICETET.2008.239.
- Chaturvedi, G. K. and Gawande, R. M. (2014) ‘Secure Mining of Association Rules in Horizontally Distributed Databases Using FDM and K & C algorithm’, 3(3), pp. 263–266.
- CIO, “R ile Büyük Veriyi Harmanlamak”, <http://www.cio.com.tr/haber/r-ile-buyuk-veriyi-harmanlamak/3/>. [Ziyaret Tarihi : 05.10.2016].
- Cichosz, P., 2015, *Data Mining Algorithms: Explained Using R*, John Wiley & Sons, ISBN: 978-1-118-33258-0.

- Craven, M. W. and Shavlik, J. W. (1997) ‘Using neural networks for data mining’, *Future Generation Computer Systems*, 13(2–3), pp. 211–229. doi: 10.1016/S0167-739X(97)00022-8.
- Çağiltay, N. E., 2011, *İş Zekası Ve Veri Ambarı Sistemleri*, ODTÜ Geliştirme Vakfı Yayıncılık, ISBN: 978-605-4362-18-9.
- DataLabTR,ha R Yazılımı, <http://datalabtr.com/index.php/r-yazilimi/>, [Ziyaret Tarihi: 02.01.2017].
- De Alwis, B., Malinga, S., Pradeeban, K., Weerasiri, D. and Perera, S. (2010) ‘Horizontal format data mining with extended bitmaps’, *Proceedings of the 2010 International Conference of Soft Computing and Pattern Recognition, SoCPaR 2010*, 4, pp. 220–223. doi: 10.1109/SOCPAR.2010.5686156.
- Delen, D., Walker, G. and Kadam, A. (2005) ‘Predicting breast cancer survivability: A comparison of three data mining methods’, *Artificial Intelligence in Medicine*, 34(2), pp. 113–127. doi: 10.1016/j.artmed.2004.07.002.
- Deshmukh, R., A., Gupta, S., R., 2014, Data mining based soft computing methods for web intelligence, *International Journal of Application or Innovation in Engineering & Management (IJAIEM)*, 3 (3), 376-382.
- DMS Tutorial, “Data Preparation”, http://dms.irb.hr/tutorial/tut_data_prepare.php. [Ziyaret Tarihi: 10.11.2016].
- Dondurmacı, G. ve Çınar, A., 2014, Finans Sektöründe Veri Madenciliği Uygulaması, *Akademik Sosyal Araştırmalar Dergisi*, 2(1), 258-271.
- Döşlü, A., 2008, *Veri Madenciliğinde Market Sepet Analizi ve Birliktelik Kurallarının Belirlenmesi*, Yükek Lisans, Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü.
- Enke, D. and Thawornwong, S. (2005) ‘The use of data mining and neural networks for forecasting stock market returns’, *Expert Systems with Applications*, 29(4), pp. 927–940. doi: 10.1016/j.eswa.2005.06.024.
- Erpolat, S., (2012), ‘Otomobil Yetkili Servislerinde Birliktelik Kurallarının Belirlenmesinde Apriori ve FP-Growth Algoritmalarının Karşılaştırılması’, *Anadolu Üniversitesi Sosyal Bilimler Dergisi*, pp. 137–146
- Fayyad, U. M. ve diğ., 1996, From Data Mining to Knowledge Discovery in Databases, *AI Magazine*, 17(3), 37-39.
- Fournier-viger, P., Wu, C. and Tseng, V. S. (2012), ‘Mining Top-K Association Rules’, 2, pp. 1–12.
- G.J. Russell, A. Petersen, Analysis of cross category dependence in market basket selection, *Journal of Retailing* 76 (3) (2000) 367–392.

- Garg, K. ve Kumar, D. (2013) ‘Comparing the Performance of Frequent Pattern Mining Algorithms’, *International Journal of Computer Applications*, 69(25), pp. 21-28. Available at: http://www.ijrcce.com/upload/2015/march/116_Comparing.pdf.
- Gilchrist, M., Mooers, D. L., Skrubbeltrang, G. and Vachon, F. (2012) ‘Knowledge Discovery in Databases for Competitive Advantage’, *Journal of Management and Strategy*, 3(2), p. 2. doi: 10.5430/jms.v3n2p2.
- Giraud-Carrier, C. and Povel, O. (2003) ‘Characterising data mining software’, *Intelligent Data Analysis*, 7(February), pp. 181–192.
- Grag, U. ve Kaur, M. (2014) ‘ECLAT Algorithm for Frequent Itemsets Generation’, *International Journal of Computer Systems*, 1(3), pp. 82–84. Ulaşılabilir: <http://www.ijcsonline.com/>.
- Gupta, R. ve Satsangi, C. S. (2012) ‘An Efficient Range Partitioning Method for Finding Frequent Patterns from Huge Database’, (2).
- Gündüz, D., 2016, ve *Teknoloji Güncelerim*, [http:// http://deryagunduz.com/?tag=fp-growth-algoritmasi](http://deryagunduz.com/?tag=fp-growth-algoritmasi), [Ziyaret tarihi: 1 Mart 2016].
- Hahsler, M. and Hornik, K. (2007) ‘New probabilistic interest measures for association rules’, *Intelligent Data Analysis*, 11(5), pp. 437–455. Available at: <http://arxiv.org/abs/0803.0966>.
- Han, J. ve Kamber, M., 2006, *Data mining: concepts and techniques (the Morgan Kaufmann Series in data management systems)*, 2nd Edition., Morgan Kaufmann Publishers, ISBN: 978-1-55860-901-3.
- Han, J. ve Kamber, M., 2006, *Data mining: concepts and techniques (the Morgan Kaufmann Series in data management systems)*, 2nd Edition., Morgan Kaufmann Publishers, ISBN: 978-1-55860-901-3.
- Han, J., Pei, J., Yin, Y. ve Mao, R. (2004) ‘Mining frequent patterns without candidate generation: A frequent-pattern tree approach’, *Data Mining and Knowledge Discovery*, 8(1), pp. 53–87. doi: 10.1023/B:DAMI.0000005258.31418.83.
- Hand, D.J., (1998), “Data Mining: Statistics and More?”, *The American Statistician*, Cilt 52, 112-118.
- Hash, U., Apriori, B., To, A., The, R., Studies, C. and Studies, C. (2011) ‘Journal of Global Research in Computer Science RESEARCH PAPER Available Online at www.jgrcs.info’, 2(5), pp. 78–80.
- Hashler, M. (2006) ‘A model-based frequency constraint for mining associations from transaction data’, *Data Mining and Knowledge Discovery*, 13(2), pp. 137–166. doi: 10.1007/s10618-005-0026-2.

- Hashler, M., Buchta, C., Gruen, B., Michael, A., Buchta, C., Gruen, B., Hornik, K. ve Borgelt, C. (2014) ‘Package “ arules ”’, *R Package Documentation*. doi: 10.1111/j.2041-210X.2011.00182.x.See.
- He, X. ve Khouja, M. (2011) ‘Pareto analysis of supply chain contracts under satisficing objectives’, *European Journal of Operational Research*, 214(1), pp. 53–66. doi: 10.1016/j.ejor.2011.03.047.
- Jain, N. ve Srivastava, V., *IJRET: International Journal of Research in Engineering and Technology* eISSN: 2319-1163 | pISSN: 2321-7308, Volume: 02 Issue: 11 | Nov-2013, Available @ <http://www.ijret.org>.
- Jensen, W. a (2008) ‘Decision Trees for Business Intelligence and Data Mining: Using SAS® Enterprise Miner™’, *Technometrics*, 50(3), pp. 409–410. doi: 10.1198/tech.2008.s904.
- Karadeniz, N., 2008, *Müşteri İlişkileri Yönetimi Açısından Veri Madenciliği Yöntemi Ve Hizmet Sektörü Üzerine Bir Uygulama*, Yüksek Lisans, Marmara Üniversitesi Sosyal Bilimler Enstitüsü.
- Karagöz, N., 2007, *Market Veri Tabanında Veri Madenciliği Uygulaması*, Yüksek Lisans, İstanbul Ticaret Üniversitesi Sosyal Bilimler Enstitüsü.
- Kartal, E., 2015, Sınıflandırmaya dayalı makine öğrenmesi teknikleri ve kardiyolojik risk değerlendirmesine ilişkin bir uygulama, Doktora Tezi, İstanbul Üniversitesi Fen Bilimleri Enstitüsü.
- Kiang, M. Y. (2003) ‘A comparative assessment of classification methods’, *Decision Support Systems*, 35(4), pp. 441–454. doi: 10.1016/S0167-9236(02)00110-0.
- Kim, W., Choi, B. J., Hong, E. K., Kim, S. K. and Lee, D. (2003) ‘A Taxonomy of Dirty Data’, *Data Mining and Knowledge Discovery*, 7(1), pp. 81–99. doi: 10.1023/A:1021564703268.
- Kiremitçi, B., 2013, *Veri Ambarlarında Veri Madenciliği ve Ulaştırma-Lojistik Sektöründe Bir Uygulama*, Yüksek Lisans, İstanbul Üniversitesi Sosyal Bilimler Enstitüsü.
- Koonce, D. A. and Tsai, S.-C. (2000) ‘Using data mining to find patterns in genetic algorithm solutions to a job shop schedule’, *Comput. Ind. Eng.*, 38(3), pp. 361–374. doi: [http://dx.doi.org/10.1016/S0360-8352\(00\)00050-4](http://dx.doi.org/10.1016/S0360-8352(00)00050-4).
- Kotu, V., ve Deshpande, B., (2015) *Predictive analytics and data mining Concepts and Practice with RapidMiner*, Elseiver, ISBN: 978-0-12-801460-8.
- Kumar, P., ve diğ., (2011) ‘Knowledge Discovery in Databases (KDD) with Images : A Novel Approach toward Image Mining and Processing’, *International Journal*, 27(6), pp. 10–13.

- Lai, K. ve Cerpa, N. (2001) 'Support vs Confidence in Association Rule Algorithms', *OPTIMA Conference*, (OCTOBER 2001), pp. 1–14.
- Liao, S. -h., Chen, Y.-J. ve Deng, M. -y. (2010) 'Mining customer knowledge for tourism new product development and customer relationship management', *Expert Systems with Applications*. Elsevier Ltd, 37(6), pp. 4212–4223. doi: 10.1016/j.eswa.2009.11.081.
- Liu, B., Hsu, W., Ma, Y. and Ma, B. (1998) 'Integrating Classification and Association Rule Mining', *Knowledge Discovery and Data Mining*, pp. 80–86. doi: 10.1.1.48.8380.
- Lomet, D. B. (2001) 'Bulletin of the Technical Committee on Data Engineering', *Bulletin of the Technical Committee on Data Engineering*, 24(4), pp. 1–56.
- Ma, Z., Yang, J., Zhang, T. ve Liu, F. (2016) 'An Improved Eclat Algorithm for Mining Association Rules Based on Increased Search Strategy', 9(5), pp. 251–266.
- Maimon, O. and Rokach, L. (2005) 'Introduction to knowledge discovery in database', *Data Mining and Knowledge Discovery Handbook*, pp. 1–17. doi: 10.1007/0-387-25465-X_1.
- Masseglia, F., Teisseire, M. and Poncelet, P. (2005) 'Sequential Pattern Mining', *Encyclopedia of Data Warehousing and Mining*, pp. 1028–1032. doi: 10.1002/pmic.200700657.
- McCallum, A. and Nigam, K. (1998) 'A Comparison of Event Models for Naive Bayes Text Classification', *AAAI/ICML-98 Workshop on Learning for Text Categorization*, pp. 41–48. doi: 10.1.1.46.1529.
- Mild, A. ve Reutterer, T. (2003) 'An improved collaborative filtering approach for predicting cross-category purchases based on binary market basket data', *Journal of Retailing and Consumer Services*, 10(3), pp. 123–133. doi: 10.1016/S0969-6989(03)00003-1.
- Musicant, D. R., Christensen, J. M. and Olson, J. F. (2007) 'Supervised learning by training on aggregate outputs', *Proceedings - IEEE International Conference on Data Mining, ICDM*, pp. 252–261. doi: 10.1109/ICDM.2007.50.
- Nahar, J., Imam, T., Tickle, K. S., Chen, Y.-P. P., Yang, H. H., Fong, S., Lara, J. a., Lizcano, D., Martínez, M. a., Pazos, J., Riera, T., Yang, H. H., Chen, Y.-P. P., Nisbet, R., Elder, J., Miner, G., Garg, S., Sharma, A., Han, J., Kamber, M., Contributions, O., Koh, H. C., Tan, G., Narwal, S. and Zhao, Y. (2013) 'Association rule mining to detect factors which contribute to heart disease in males and females', *Expert Systems with Applications*. Elsevier Ltd, 102(3), pp. 335–351. doi: 10.1016/j.jss.2014.07.010.
- Namey, E., Guest, G., Thairu, L. and Johnson, L. (2007) 'Data reduction techniques for large qualitative data sets', *Handbook for team-based qualitative research*, pp. 137–163.

- Nasreen, S., ve diğ., 2014, Frequent Pattern Mining Algorithms for Finding Associated Frequent Patterns for Data Streams: A Survey, *The 5th International Conference on Emerging*
- Ngai, E. W. T., Xiu, L. ve Chau, D. C. K. (2009) 'Application of data mining techniques in customer relationship management: A literature review and classification', *Expert Systems with Applications*. Elsevier Ltd, 36(2 PART 2), pp. 2592–2602. doi: 10.1016/j.eswa.2008.02.021.
- Nikita J. ve Vishal S., 2013, Data Mining Techniques: A Survey Paper, *International Journal of Research in Engineering and Technology*, 2(11), 116-118.
- Oğuzlar, A., 2003, Veri Önişleme, *Erciyes Üniversitesi, İktisadi ve İdari Bilimler Fakültesi Dergisi*, 21(2), 67-76
- Özçakır, C. F. ve Çamurcu, Y., 2007, Birliktelik Kuralı Yöntemi İçin Bir Veri Madenciliği Yazılımı Tasarımı Ve Uygulaması, *İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi*, 22-23.
- Özçakır, C. F., 2006, Müşteri İşlemlerindeki Birlikteliklerin Belirlenmesinde Veri Madenciliği Uygulaması, Yüksek Lisans Tezi, Marmara Üniversitesi Fen Bilimleri Enstitüsü.
- Özkan, Y ve Selçukcan Erol, Ç., *Biyoenformatik DNA Mikrodizi Veri madenciliği*, Papatya Yayıncılık Eğitim, ISBN: 978-605-4220-89-2.
- Özkan, Y., 2008, *Veri madenciliği yöntemleri*, Papatya Yayıncılık Eğitim, ISBN: 978-975-6797-82-2.
- Palaniappan, S. ve Awang, R. (2008) 'Intelligent Heart Disease Prediction System Using Data Mining Techniques', *IEEE/ACS International Conference on Computer Systems and Applications (AICCSA)*, pp. 108–115. doi: 10.1109/AICCSA.2008.4493524.
- Park, J. S., Chen, M. S. ve Yu, P. S. (1995) 'An effective hash-based algorithm for mining association rules', *Proceedings of the 1995 ACM SIGMOD international conference on Management of data*, pp. 175–186. doi: 10.1145/568271.223813.
- Pawlak, Z. (2003) 'A rough set view on Bayes' theorem', *International Journal of Intelligent Systems*, 18(5), pp. 487–498. doi: 10.1002/int.10099.
- Powers, D. A. ve Xie Y., 2000, *Statistical Methods for Categorical Data Analysis*, 2nd edition., Academic Press, San Diego.
- Raju, P. Ve diğ., 2014, Data mining: Techniques for Enhancing Customer Relationship Management in Banking and Retail Industries, *International Journal of Innovative Research in Computer and Communication Engineering*, 2(1), 2651-2653.
- Rashid, M. M., Gondal, I. and Kamruzzaman, J. (2013) 'Mining associated sensor patterns for data stream of wireless sensor networks', in *Proceedings of the 8th*

ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks - PM2HW2N '13. New York, New York, USA: ACM Press, pp. 91–98. doi: 10.1145/2512840.2512853.

Rokach, L. and Maimon, O. (2010) ‘Chapter 15— Clustering methods’, *The Data Mining and Knowledge Discovery Handbook*, p. 32. doi: 10.1007/0-387-25465-X_15.

r-project, 2014, *The R Project for Statistical Computing*, <http://www.r-project.org/>, [Ziyaret Tarihi: 10.11.2014].

RStudio, 2015a, *Home - RStudio*, <http://www.rstudio.com/>, [Ziyaret Tarihi: 13.01.2015].

RStudio, 2016b, *Shiny*, <http://shiny.rstudio.com/>, [Ziyaret Tarihi: 30.06.2016].

RStudio, 2016c, *Shinyapps.io*, <https://www.shinyapps.io/>, [Ziyaret Tarihi: 12.07.2016].

Rud Parr, O., 2001, *Data Mining Cookbook Modeling Data For Marketing, Risk, and Customer Relationship Management*, 2nd ed, Wiley, New York

Rudin, C., ve diğ., (2011) ‘Sequential Event Prediction with Association Rules’, 19, pp. 615–634.

Rygielski, C., Wang, J.-C. and Yen, D. C. (2002) ‘Data mining techniques for customer relationship management’, *Technology in Society*, 24(4), pp. 483–502. doi: 10.1016/S0160-791X(02)00038-6.

Savaş, S. ve diğ., 2012, Veri Madenciliği Ve Türkiye’deki Uygulama Örnekleri, *İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi*, 11(21), 1-23.

Sever, H., ve Oğuz, B., 2002, Veri Tabanlarında Bilgi Keşfine Formel Bir Yaklaşım, Kısım I: Eşleştirme Sorguları ve Algoritmalar, *Bilgi Dünyası*, 3(2), 5-7.

Shearer, C., 2000, The CRISP-DM model: the new blueprint for data mining, *Journal of data warehousing*, 5 (4), 13–22.

Shim ve diğ., 2012, CRM strategies for a small-sized online shopping mall based on association rules and sequential patterns, *Expert Systems with Applications*, 39(9), 7736–7742.

Silahtaroglu, G., 2016, Veri Madenciliği Kavram ve Algoritmaları, Papatya Yayıncılık Eğitim, ISBN: 9756797815.

Singh, J. ve Ram, H. (2013) ‘Improving efficiency of Apriori algorithm using transaction reduction’, *International Journal of Scientific and Research Publications*, 3(1), pp. 1–4.

Sinha, G. ve Ghosh, S. M. (2014) ‘Identification of Best Algorithm in Association Rule Mining Based on Performance’, 3(11), pp. 38–45.

- Somasundaram, R. S. ve Nedunchezian, R., 2011, Evaluation of three simple imputation methods for enhancing preprocessing of data with missing values, *International Journal of Computer Applications*, Vol21, (10).
- Sondwale, P. P. (2015) ‘Overview of Predictive and Descriptive Data Mining Techniques’, *International Journal of Advanced Research in Computer Science and Software Engineering*, 5(4), pp. 262–265.
- Srikant, R. ve Agrawal, E. (1996) ‘Mining Sequential Patterns: Generalization and Performance Improvements’, *5th International Conference on Extending Database Technology (EDBT '96)*, pp. 3–17. doi: 10.1109/ICDE.1995.380415.
- Suthar, B., Patel, H. ve Goswami, A., 2012, A Survey: Classification of Imputation Methods in Data Mining, *International Journal of Emerging Technology and Advanced Engineering*, 2 (1).
- Swift, R.S., 2000, *Accelerating Customer Relationship: Using CRM and Relationship Technologies*, Prentice Hall PTR, ISBN:0-13-088984-9.
- Şen, F., 2008, *Veri madenciliği ile Birliktelik Kuralları Bulunması*, Yüksek Lisans Tezi, Sakarya Üniversitesi Fen Bilimleri Enstitüsü.
- Şimşek, U., T., 2006, *Veri madenciliği ve müşteri ilişkileri yönetiminde (CRM) bir uygulama*, Doktora, İstanbul Üniversitesi Sosyal Bilimler Enstitüsü.
- Talib, F., Rahman, Z. and Qureshi, M. N. (2011) ‘Pareto analysis of total quality management factors critical to success for service industries’, *International Quality Conference*, 5th(20ty May), pp. 707–726.
- Tan, P.-N., Steinbach, M. and Kumar, V. (2005) ‘Association Analysis: Basic Concepts and Algorithms’, *Introduction to Data mining*, pp. 327–414. doi: 10.1111/j.1600-0765.2011.01426.x.
- Taş, M., Adak, M. F. and Yurtay, N. (2012) ‘Öğrencilerin Staj Verileri Üzerine Uygulanan Apriori Algoritması ile Birliktelik Kurallarının Çıkarılması ve Staj Eğiliminin Belirlenmesi’, pp. 1086–1095.
- Terzi, Ö., ve diğ., *Veri Madenciliği Süreci Kullanılarak Güneş Işınımı Tahmini*, Süleyman Demirel Üniversitesi International Technologic Science, 3(2), 30-33.
- Tezcanlar, P., 2007, *Müşteri İlişkileri Yönetimi, Veri Madenciliği Ve Bir Uygulama*, Yüksek Lisans, İstanbul Üniversitesi Sosyal Bilimler Enstitüsü.
- Timör ve diğ., 2011, *Müşteri Profili Ve Alışveriş Davranışlarını Belirlemede Kümeleme Ve Birliktelik Kuralları Analizi : Perakende Sektöründe Bir Uygulama*, *İstanbul Üniversitesi İşletme Fakültesi İşletme İktisadi Enstitüsü Dergisi--c. 22*, s. 68.
- Tseng S., 2000, *Efficient mining of categorized association rules in large databases*. Systems, Man, and Cybernetics, 2000 IEEE International Conference, Vol. 5, pp 3606-3610.

- Tsiptsis, K. ve Chorianopoulos, A. (2009), *Data Mining Techniques in CRM: Inside Customer Segmentation*, Wiley Publications ISBN: 978-0-470-74397-3.
- Tutorialspoint Simple Easy Learning, 2016, *Data Mining- Cluster Analysis*, https://www.tutorialspoint.com/data_mining/dm_cluster_analysis.htm, [Ziyaret Tarihi: 08.10.2016.]
- UCLA: Statistical Consulting Group, 2013, *R Data Analysis Examples: Logit Regression*, <http://www.ats.ucla.edu/stat/r/dae/logit.htm>, [Ziyaret Tarihi: 01.08.2016].
- Ubiquitous Systems and Pervasive Networks (EUSPN-2014)*, 22-25 Septmber 2014, Halifax, Nova Scotia, Canada, *Procedia Computer Science*, 109-116.
- Vellido, A., Martin-Guerrero, J. D. ve Lisboa, P. J. G., 2012, Making machine learning models interpretable, 2012 Louvain-La-Neuve, I6doc.com, ISBN: 978-2-87419-049-0.
- Video Eğitim Portalı, 2016, Uşak Üniversitesi Video Eğitim Portalı, <http://usak.videoegitim.com/kurs/123>, [Ziyaret Tarihi: 12.11.2016.]
- Vijayalakshmi, V. ve Pethalakshmi, A. (2014) ‘An efficient count based transaction reduction approach for mining frequent patterns’, *Procedia Computer Science*. Elsevier Masson SAS, 47(C), pp. 52–61. doi: 10.1016/j.procs.2015.03.183.
- Weinberger, K. Q. and Saul, L. K. (2009) ‘Distance metric learning for large margin nearest neighbor classification’, *The Journal of Machine Learning Research (JMLR)*, 10, pp. 207–244. Available at: <http://portal.acm.org/citation.cfm?id=1577078>.
- Wen, C. H., Liao, S. H., Chang, W. L. ve Hsu, P. Y. (2012) ‘Mining shopping behavior in the Taiwan luxury products market’, *Expert Systems with Applications*. Elsevier Ltd, 39(12), pp. 11257–11268. doi: 10.1016/j.eswa.2012.03.072.
- Wirth, R. (2000) ‘CRISP-DM: Towards a Standard Process Model for Data Mining’, *Proceedings of the Fourth International Conference on the Practical Application of Knowledge Discovery and Data Mining*, (24959), pp. 29–39. doi: 10.1.1.198.5133.
- Witten ve Frank, 2005, *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd Edition, Elsevier, ISBN: 0-12-088407-0.
- Yang, T. C. ve Lai, H. (2006) ‘Comparison of product bundling strategies on different online shopping behaviors’, *Electronic Commerce Research and Applications*, 5(4), pp. 295–304. doi: 10.1016/j.elerap.2006.04.006.
- Zaki, M. J., Parthasarathy, S., Ogihara, M. and Li, W. (1997) ‘New Algorithms for Fast Discovery of Association Rules’, *3rd Intl Conf on Knowledge Discovery and Data Mining*, 20(651), pp. 283–286. doi: 10.1.1.42.5143.

Zhang, C. ve Zhang, S., 2007, Association Rule Mining Models and Algorithms, Berlin, Springer, ISBN 3-540-43533-6.

Ziarati, R. (2003) 'Safety at Sea – Applying Pareto Analysis', (September).



EKLER

EK1. Uygulamada Kullanılan Kodlar

#Apriori Algortimasına Ait Kodlar

```
install.packages("arules")
install.packages("arulesNBMminer")
library(arules)
library(arulesNBMminer)
install.packages("arulesviz")
library(arulesNbmminer)
library(arulesviz)
install.packages("rCMA")
library(cmaes)
library(xlsx)
```

#veri

```
read.transactions(file="D://Documents//R//workspace//asc_product.txt",
rm.duplicates=TRUE, format="basket", sep="\t");
```

```
set.seed(1)
```

```
kurallar <- apriori(veri, parameter = list(supp = 0.01 ,conf = 0.06,
target = "rules", minlen=2, maxlen=10))
```

#Koşul belirtilmek istenirse :

```
#kurallar<-apriori(veri,parameter = list(supp=0.005, conf=0.7,
minlen=2), appearance=list(rhs=c("Marka="Nexans"))))
```

#Kurallar güven değerine sıralanıyor, en iyi 10 kural görüntülensin

```
inspect(head(sort(kurallar, by ="confidence"),10))
```

#Kurallar destek değerine sıralanıyor, en iyi 10 kural görüntülensin

```
inspect(head(sort(kurallar, by ="support"),10))
inspect(kurallar)
```

```
View(product_2015)
```

```
kurallar.sorted <- sort(kurallar, by="confidence")
inspect(kurallar.sorted)
```

#Kuralların Budanması

```
subset.matrix <- is.subset(kurallar.sorted,kurallar.sorted)
subset.matrix[lower.tri(subset.matrix,diag=T)]<- NA
redundant <-colSums(subset.matrix,na.rm=T) >=1
which(redundant)
kurallar.pruned<-kurallar.sorted[!redundant]
inspect(kurallar.pruned)
```

#Kuralların Görselleştirilmesi

```

install.packages("arulesviz")
library(arulesviz)
subrules2 <- head(sort(kurallar, by="lift"), 10)
plot()

#Tools--Install Packages yolu ile indirilenler:
install.packages("arules","scatterplot3d","vcd",
"seriation","igraph","grid","cluster","TSP","gclus","colorspace")
str(veri)
plot(kurallar.pruned, method = NULL, measure = "support", shading =
"lift",interactive = FALSE, data = NULL, control = NULL)
plot(kurallar,main="Scatter Plot for 9 rules (2015)")

# lift ve confidence yerini değiştirebiliriz :
#plot(rules, measure=c("support", "lift"), shading="confidence")

#Two key plot
plot(kurallar, shading="order", control=list(main = "Two-key plot for
489 rules"))

#Interactive
sel <- plot(kurallar, measure=c("support", "lift"),
shading="confidence", interactive=TRUE)

head(quality(rules))
plot(kurallar, main="Scatter Plot for 4 rules (2015)")

#Grouped Matrix
plot(kurallar.pruned, method="grouped")

#2D Matrix
plot(kurallar, method="matrix", measure="lift")

#Graph
plot(subrules2, method="graph")
#Paracoord
plot(kurallar, method="paracoord",control=list(reorder=TRUE))
quality(kurallar) <- interestMeasure(kurallar, trans=veri)
head(quality(rules))
# olmad?-->plot(kurallar, measure=c("support", "allconfidence"),
shading="lift")

plot(kurallar.pruned)
inspect(kurallar.pruned)

```

#Eclat Algoritmasına Ait Kodlar#

```

show(veri)
rules<-eclat(veri, parameter = list(supp = 0.002, minlen = 2), control
= NULL)
inspect(rules)
rules.sorted<-sort(rules,by="support")
inspect(rules.sorted)
inspect(head(sort(rules, by ="supp"),10))
library(arulesviz)
plot(rules)
plot(rules.sorted)

#subrules<-subset(rules,supp>0.2)
plot(rules,method="graph")
plot(rules, method="paracoord", control=list(alpha=.5, reorder=TRUE))

inspect(rules)
inspect(head(sort(rules, by ="supp"),15))

```

```
inspect(rules)
```

#NBMiner Algoritmasına Ait Kodlar#

```
param <- NBMinerParameters(veri)
rules_NB <- NBMiner(veri, parameter = param, control = NULL)
inspect(head(rules_NB))
inspect(rules_NB)

param <- NBMinerParameters(veri, pi=0.99, theta=0.5,
maxlen=5,rules=TRUE, minlen=2, trim = 0,plot = TRUE)

rules_NB <- NBMiner(veri, parameter = param, control = list(verb = TRUE,
debug=FALSE))

inspect(head(itemsets_NB))
```

#Shiny Uygulamasında Apriori Algoritmasının Uygulandığı R Kodu

#Tekrarlayan kısımlarda kodların örnek teşkil etmesi adına bir kısmına yer verilmiştir.

```
library(shiny)
library(arules)
library(arulesNBMiner)
library(arulesViz)
library(cmaes)
library(xlsx)
library(readr)
library(ggplot2)
library(stringr)
library(data.table)

datam <- read_delim("datam.csv",
";", escape_double = FALSE, locale = locale(encoding
= "UTF-8"),
na = "empty", trim_ws = TRUE)

dummy.df <- data.frame(datam)
urunlistesi <- unique(datam$URUNLER)

shinyServer(function(input, output) {

  #YIL SECIMI
  output$Yillar = renderUI(
    if(is.null(input$tms) || is.null(input$yil) || is.null(input$ay) ||
is.null(input$blg)|| is.null(input$cg)|| is.null(input$ck)){
      #Null Kontrolü
      Yillar<- c("Tumu",unique(datam$YIL))
      selectInput("yil","Yıl Seçiniz", choices = Yillar, selected=
input$yil)
    }else
      if(input$blg!="Tumu"){
        if(input$ay!="Tumu"){
          if(input$tms!="Tumu"){
            if(input$cg!="Tumu"){
              if(input$ck!="Tumu"){
                #Bolgex,Ayx,Temsilcix,cirogrubux,carikodx seçili ise
                Yillar<-
c("Tumu",unique(datam$YIL[which(datam$Bolge==input$blg &
datam$AY==input$ay & datam$TEMSILCI==input$tms &
datam$CARI_KOD==input$ck & datam$CIRO_GRUBU == input$cg)]))
```

```

        selectInput("yil","Yıl Seçiniz", choices = Yillar,
selected= input$yil)
      }else {
        #"Bolgex,Ayx,Temsilcix,cirogrubux seÅşili ise
        Yillar<-
c("Tumu",unique(datam$YIL[which(datam$Bolge==input$blg
datam$AY==input$ay & datam$TEMSILCI==input$tms & datam$CIRO_GRUBU ==
input$cg)]))
        selectInput("yil","Yıl Seçiniz", choices = Yillar,
selected= input$yil)}
      }else if(input$sck!="Tumu"){
        #Bolgex,Ayx,Temsilcix,carikodx seÅşili ise
        Yillar<-
c("Tumu",unique(datam$YIL[which(datam$Bolge==input$blg
datam$AY==input$ay & datam$TEMSILCI==input$tms
datam$CARI_KOD==input$sck)]))
        selectInput("yil","Yıl Seçiniz", choices = Yillar,
selected= input$yil)
      }else {
        #"Bolgex,Ayx,Temsilcix seÅşili ise
        Yillar<-
c("Tumu",unique(datam$YIL[which(datam$Bolge==input$blg
datam$AY==input$ay & datam$TEMSILCI==input$tms)]))
        selectInput("yil","Yıl Seçiniz", choices = Yillar,
selected= input$yil)}
      }else if(input$cg!="Tumu"){
        if(input$sck!="Tumu"){
          #"Bolgex,Ayx,cirogrubux,carikodx seÅşili ise
          Yillar<-
c("Tumu",unique(datam$YIL[which(datam$Bolge==input$blg
datam$AY==input$ay & datam$CIRO_GRUBU == input$cg
datam$CARI_KOD==input$sck)]))
          selectInput("yil","Yıl Seçiniz", choices = Yillar,
selected= input$yil)
        }
      }
#Bolge SECIMI
  output$Bolgem = renderUI(
    if(is.null(input$tms) || is.null(input$yil) || is.null(input$ay) ||
is.null(input$blg)|| is.null(input$cg)|| is.null(input$sck)){
      Bolgeler<- c("Tumu","Hiccbiri 2",unique(datam$Bolge))
      selectInput("blg","Bölge Seçiniz", choices = Bolgeler, selected=
input$blg)
    }else
      if(input$yil!="Tumu"){
        if(input$ay!="Tumu"){
          if(input$tms!="Tumu"){
            if(input$cg!="Tumu"){
              if(input$sck!="Tumu"){
                #"Yil,Ayx,Temsilcix,cirogrubux,carikodx",
                Bolgeler<-
c("Tumu",unique(datam$Bolge[which(datam$YIL==input$yil
datam$AY==input$ay & datam$TEMSILCI==input$tms
datam$CARI_KOD==input$sck & datam$CIRO_GRUBU == input$cg)]))
                #.....#
              }
            }
          }
        }
      }
# AY SECIMI
  output$Aylar = renderUI(
    if(is.null(input$tms) || is.null(input$yil) || is.null(input$ay) ||
is.null(input$blg)|| is.null(input$cg)|| is.null(input$sck)){
      Aylar2<- c("Tumu","Hiccbiri 2",unique(datam$AY))
      selectInput("ay","Ay Seçiniz", choices = Aylar2, selected=
input$ay)
    }else

```

```

        if(input$yil!="Tumu"){
          if(input$blg!="Tumu"){
            if(input$tms!="Tumu"){
              if(input$cg!="Tumu"){
                if(input$ck!="Tumu"){
                  #"Yil,Bolgex,Temsilcix,cirogrubux,carikodx",
                  Aylar2<-
c("Tumu",unique(datam$AY[which(datam$YIL==input$yil           &
datam$Bolge==input$blg           &           datam$TEMSILCI==input$tms           &
datam$CARI_KOD==input$ck & datam$CIRO_GRUBU == input$cg)]))
                  selectInput("ay","Ay Seçiniz", choices = Aylar2,
selected= input$ay)
#.....#

#Ciro Grubu SECIMI
  output$CiroGrubu = renderUI(

    if(is.null(input$tms) || is.null(input$yil) || is.null(input$ay) ||
is.null(input$blg)|| is.null(input$cg)|| is.null(input$ck)){
      cirogruplari<- c("Tumu","Hicbiri 2",unique(datam$CIRO_GRUBU))
      selectInput("cg","Ciro Grubu Seçiniz", choices = cirogruplari,
selected= input$cg)
    }else
      if(input$yil!="Tumu"){
        if(input$ay!="Tumu"){
          if(input$tms!="Tumu"){
            if(input$blg!="Tumu"){
              if(input$ck!="Tumu"){
                #"Yil,Ayx,Temsilcix,Bolgex,carikodx",
                cirogruplari<-
c("Tumu",unique(datam$CIRO_GRUBU[which(datam$YIL==input$yil           &
datam$AY==input$ay           &           datam$TEMSILCI==input$tms           &
datam$CARI_KOD==input$ck & datam$Bolge == input$blg)]))
                selectInput("cg","Ciro Grubu Seçiniz", choices =
cirogruplari, selected= input$cg)
              }else {
                #"Yil,Ayx,Temsilcix,Bolgex ",
                cirogruplari<-
c("Tumu",unique(datam$CIRO_GRUBU[which(datam$YIL==input$yil           &
datam$AY==input$ay & datam$TEMSILCI==input$tms & datam$Bolge ==
input$blg)]))
                selectInput("cg","Ciro Grubu Seçiniz", choices =
cirogruplari, selected= input$cg)}
            }else if(input$ck!="Tumu"){
              #"Yil,Ayx,Temsilcix,carikodx",
              cirogruplari<-
c("Tumu",unique(datam$CIRO_GRUBU[which(datam$YIL==input$yil           &
datam$AY==input$ay           &           datam$TEMSILCI==input$tms           &
datam$CARI_KOD==input$ck)]))
              selectInput("cg","Ciro Grubu Seçiniz", choices =
cirogruplari, selected= input$cg)
            }
#.....#

#Cari Kod SECIMI
  output$Carikod = renderUI(
    if(is.null(input$tms) || is.null(input$yil) || is.null(input$ay) ||
is.null(input$blg)|| is.null(input$cg)|| is.null(input$ck)){
      # Null kontrol
      Carikodlar<- c("Tumu",unique(datam$CARI_KOD))
      selectInput("ck","Cari Kod Seçiniz", choices = Carikodlar,
selected= input$ck)
    }else

```



```

        if(input$yil!="Tumu"){
          if(input$ay!="Tumu"){
            if(input$tms!="Tumu"){
              if(input$cg!="Tumu"){
                if(input$blg!="Tumu"){
                  #"Yil,Ayx,Temsilcix,cirogrubux,Bolgem",
                  Carikodlar<-
c("Tumu",unique(datam$CARI_KOD[which(datam$YIL==input$yil &
datam$AY==input$ay & datam$TEMSILCI==input$tms & datam$Bolge==input$blg
& datam$CIRO_GRUBU == input$cg)]))
                  selectInput("ck","Cari Kod Seçiniz", choices =
Carikodlar, selected= input$sck)
#.....#

#TEMSILCI SECIMI
output$Temsilci = renderUI(
  if(is.null(input$tms) || is.null(input$yil) || is.null(input$ay) ||
is.null(input$blg)|| is.null(input$cg)|| is.null(input$sck)){
    # Null Kontrol
    Temsilciler<- c("Tumu",unique(datam$TEMSILCI))
    selectInput("tms","Temsilci Seçiniz", choices = Temsilciler,
selected= input$tms)
  }else
  if(input$yil!="Tumu"){
    if(input$ay!="Tumu"){
      if(input$blg!="Tumu"){
        if(input$cg!="Tumu"){
          if(input$sck!="Tumu"){
            #"Yil,Ayx,Bolgem,cirogrubux,carikodx",
            Temsilciler<-
c("Tumu",unique(datam$TEMSILCI[which(datam$YIL==input$yil &
datam$AY==input$ay & datam$Bolge==input$blg & datam$CARI_KOD==input$sck
& datam$CIRO_GRUBU == input$cg)]))
            selectInput("tms","Temsilci Seçiniz", choices =
Temsilciler, selected= input$tms)
#.....#

### ui.R ####

library(shiny)
shinyUI(fluidPage(
  titlePanel("Apriori Uygulaması"),

  actionButton("go", "Analyze Başla"),
  br(),
  br(),
  downloadButton("downloadData", label = "Kuralları İndir"),
  br(),
  br(),

  sidebarLayout(
    sidebarPanel(
      fluidRow(
        column(6,uiOutput("Yillar")),
        column(6,uiOutput("Aylar"))),
      fluidRow(
        column(6,uiOutput("Temsilci")),
        column(6,uiOutput("Bolgem"))),
      fluidRow(
        column(6,uiOutput("CiroGrubu")),
        column(6,uiOutput("Carikod"))),
      fluidRow(
        column(6,numericInput("supp", "Destek Değeri:", min = 0, max =
1,value = 0.001, step = 0.001)),

```

```

        column(6,numericInput("kuralSayisi", "Kural Sayısı:", min = 1,
max = 100,value = 10))
    ),
    fluidRow(
        column(12,selectizeInput("urun","Ürün
Seçiniz",c("Tümü",urunlistesi), multiple = TRUE,options =
list(maxOptions = 20))))
    ),
    mainPanel(
        tabsetPanel(type = "tabs",
                    tabPanel("Grafik", plotOutput("distPlot")),
                    tabPanel("Kurallar",br(), textOutput("print"), br(),
                    verbatimTextOutput("rulesTable" ) ,
                    tabPanel("Sorgulanan Veri", tableOutput("table2")),
                    tabPanel("Analiz Edilen Veri", tableOutput("table3"))
        )
    ))))

```

ÖZGEÇMİŞ

Kişisel Bilgiler	
Adı Soyadı	Gökçe KARAHAN ADALI
Doğum Yeri	İstanbul
Doğum Tarihi	05.08.1985
Uyruğu	<input checked="" type="checkbox"/> T.C. <input type="checkbox"/> Diğer:
Telefon	0535 642 56 70
E-Posta Adresi	gkarahanadali@gmail.com
Web Adresi	



Eğitim Bilgileri	
Lisans	
Üniversite	Bahçeşehir Üniversitesi
Fakülte	Fen Edebiyat
Bölümü	Matematik-Bilgisayar(% 50 Burslu)
Mezuniyet Yılı	2008

Yüksek Lisans	
Üniversite	Yıldız Teknik Üniversitesi
Enstitü Adı	Sosyal Bilimler
Anabilim Dalı	İşletme
Programı	İşletme Yönetimi
Mezuniyet Tarihi	2011

Doktora	
Üniversite	İstanbul Üniversitesi
Enstitü Adı	Fen Bilimleri Enstitüsü
Anabilim Dalı	Anabilim Dalı
Programı	Enformatik Programı
Mezuniyet Tarihi	2017

Makale ve Bildiriler	
Erkollar, A., Tezcan, N., Özyurt Serim, B., Karahan Adalı, G., “Üretim Sistemlerinde Matematiksel Model Entegrasyonu İle Optimizasyon” 12. Üretim Araştırmaları Sempozyumu (Üas), 27-29 Eylül 2012, pp.123-135.	