T.C.

FIRAT ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

ÖZELLİK SEÇİMİ, SINIFLAMA VE ÖNGÖRÜ UYGULAMALARINA YÖNELİK BİRLİKTELİK KURALI ÇIKARIMI VE YAZILIM GELİŞTİRİLMESİ

Murat KARABATAK

Tez Yöneticisi

Yrd. Doç. Dr. Melih C. İNCE

DOKTORA TEZİ

ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

ELAZIĞ, 2008

T.C.

FIRAT ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

ÖZELLİK SEÇİMİ, SINIFLAMA VE ÖNGÖRÜ UYGULAMALARINA YÖNELİK BİRLİKTELİK KURALI ÇIKARIMI VE YAZILIM GELİŞTİRİLMESİ

Murat KARABATAK

Doktora Tezi

Elektrik-Elektronik Mühendisliği Anabilim Dalı

Bu tez, 25.07.2008 tarihinde aşağıda belirtilen jüri tarafından oybirliği /~~oyçokluğu~~ ile başarılı / ~~başarısız~~ olarak değerlendirilmiştir.

Danışman: Yrd. Doç. Dr. Melih Cevdet İNCE

Üye: Prof. Dr. Muammer GÖKBULUT

Üye: Prof. Dr. Mustafa POYRAZ

Üye: Doç. Dr. İbrahim TÜRKOĞLU

Üye: Yrd. Doç. Dr. Seydi Vakkas ÜSTÜN

Bu tezin kabulü, Fen Bilimleri Enstitüsü Yönetim Kurulu'nun ...../...../..... tarih ve ..... sayılı kararıyla onaylanmıştır.

TEŞEKKÜR

Bu tez çalışması süresince yardımlarını ve desteğini esirgemeyen, değerli fikirleriyle bana yol gösteren tez yöneticim Sayın Yrd. Doç. Dr. Melih Cevdet İNCE hocama en içten teşekkür ve şükranlarımı sunarım.

Yine bu süre zarfında, başta Doç. Dr. İbrahim TÜRKOĞLU, Yrd. Doç. Dr. Abdulkadir ŞENGÜR, Yrd. Doç. Dr. Engin AVCI, Yrd. Doç. Dr. Davut HANBAY ve Özal YILDIRIM olmak üzere desteklerini ve yardımlarını esirgemeyen tüm bölüm hocalarıma ve arkadaşlarıma teşekkürü bir borç bilirim.

Ayrıca tez çalışması süresince oldukça fazla ihmal ettiğim kızım Elif ve oğlum Murat Emre'den özür diler, onları çok sevdiğimi belirtmek isterim.

Murat KARABATAK

İÇİNDEKİLER

ŞEKİLLER LİSTESİ

TABLOLAR LİSTESİ

SİMGELER LİSTESİ

D	Veri tabanı
|D|	Veri tabanı boyutu
(X,Y)	Dokularda birliktelik kuralı Gösterimi
μ	Piksellerin ortalama değerleri
σ	Piksellerin standart sapması
I	Veri tabanı nitelikleri
T	Veri tabanı işlemleri
C_i	i-elemanlı aday nesne kümeleri
L_i	i-elemanlı yoğun nesne kümeleri
$X \rightarrow Y$	Birliktelik kuralı gösterimi
c	Güven değeri
s	Destek değeri
x	Gerçek giriş uzayındaki gözlemler veya ölçümler
y	Çıkış sınıfları
w_{ij}	Yapay sinir ağı ağırlıkları
h_j	Çok katmanlı yapay sinir ağında j. ara katmanın hatası
α	Öğrenme oranı
θ_i	i. işlem elemanının eşik değeri
a(.)	Etkinleştirme fonksiyonu
$\psi(t)$	Dalgacık fonksiyonu
m	Dalgacık ayrışım seviyesi
A	Dalgacık ayrışımının yaklaşık katsayısı
H	Dalgacık ayrışımının detay katsayısı (Horizontal)
D	Dalgacık ayrışımının detay katsayısı (Diagonal)
V	Dalgacık ayrışımının detay katsayısı (Verticall)
h[n]	Yüksek geçiren filtre
g[n]	Alçak geçiren filtre
X(f)	İşaretin Fourier dönüşümü
F(.)	Dalgacık sinir ağı etkinleştirme fonksiyonu
t_k, 2^k	Dalgacık fonksiyonunun parametreleri

KISALTMALAR LİSTESİ

AIS	Agrawal Imielinski Swami
BKTÖS	Birliktelik Kuralı Tabanlı Özellik Seçimi
BÖS	Bireysel Özellik Seçimi
ÇAİ	Çevrimiçi Analitik İşleme - *Online Analytical Processing (OLAP)*
ÇANKB	Çevrimdışı Aday Nesne Kümesi Belirleme
DAA	Doğrusal Ayrışım Analizi - *Linear Discriminant Analysis (LDA)*
DD	Dalgacık Dönüşümü
DHP	Dynamic Hashing and Pruning
DIC	Dynamic Itemset Counting
DVM	Destek Vektör Makinesi - *Support Vector Machine (SVM)*
GA	Genetik Algoritma
GÖS	Geri Özellik Seçimi
GSEM	Gri Seviyesi Eş Oluşum Matrisi - *Gray Level Cooccurence Matrix (GLCM)*
GSTU	Gri Seviyesi Tekrarlama Uzunluğu - *Gray Level Run Lenght (GLRL)*
HFD	Hızlı Fourier Dönüşümü
İÖS	İleri Özellik Seçimi
KZFD	Kısa Zamanlı Fourier Dönüşümü
MDD	Multi Dimensional Database
MRA	Markov Rasgele Alanlar
SETM	Set Oriented Mining
SQL	Structured Query Language
TBA	Temel Bileşen Analizi - *Principal Component Analiysis (PCA)*
VTBK	Veri Tabanlarında Bilgi Keşfi
YNK	Yoğun Nesne Kümesi
YNKTÖS	Yoğun Nesne Kümesi Tabanlı Özellik Seçimi
YSA	Yapay Sinir Ağı
ZFG	Zaman Frekans Gösterimi

ÖZET

Doktora Tezi

ÖZELLİK SEÇİMİ, SINIFLAMA VE ÖNGÖRÜ UYGULAMALARINA YÖNELİK BİRLİKTELİK KURALI ÇIKARIMI VE YAZILIM GELİŞTİRİLMESİ

Murat KARABATAK

Fırat Üniversitesi
Fen Bilimleri Enstitüsü
Elektrik – Elektronik Mühendisliği Anabilim Dalı
2008, Sayfa: 116

Son yıllarda, bilgisayar sistemlerinin yaygın olarak kullanılmasıyla beraber tüm veriler veri tabanlarında saklanmaya başlanmış ve gün geçtikçe veri tabanları büyük kapasitelere ulaşmıştır. Bunun sonucu veri tabanlarından bilgi keşfi önemli bir araştırma alanı olmuştur. Bu alandaki en önemli yöntemlerden biri birliktelik kuralı çıkarımıdır. Bu tez çalışmasında, birliktelik kuralı yöntemi farklı alanlara uygulanmış ve gösterdiği başarımlar değerlendirilmiştir. Bu doğrultuda üç farklı uygulama yapılmıştır:

1. Herhangi bir veri tabanına ait nitelikler arasındaki ilişkiler tespit edilerek özellik seçimi uygulaması yapılmıştır. Birliktelik kuralına dayalı özellik seçimi yöntemi önerilerek elde edilen sonuçlar diğer özellik seçimi yöntemleri ile karşılaştırılmıştır.
2. Birliktelik kuralı, veri tabanına ait nitelikler arasındaki ilişkileri ortaya çıkarabilmektedir. Bu özelliği sayesinde, birliktelik kuralı yöntemi kullanılarak doku sınıflama işlemi gerçekleştirilmiştir. Bu işlemlerden hız ve başarım artışı sağlayabilmek için, kenar çıkarma ve dalgacık dönüşümü yöntemleri ile dokudan özellik çıkarımı yapılmıştır.
3. Birliktelik kuralı, öğrenci verilerine uygulanarak öğrencilerin derslerden aldığı notlar analiz edilmiş ve geleceğe yönelik notlar ile ilgili öngörüler yapılmıştır. Ayrıca bu amaca uygun bir yazılım geliştirilmiştir.

Yapılan bu üç ayrı uygulama alanında birliktelik kuralı yönteminin önemli başarılar elde ettiği görülmüştür. Elde edilen sonuçların literatürde yer alan diğer yöntemlerle karşılaştırılması sonucu, birliktelik kuralı yönteminin, söz konusu özellik seçimi, doku sınıflama ve not öngörüsü uygulamalarındaki başarımının kayda değer olduğu sonucuna varılmıştır.

Anahtar Kelimeler: Birliktelik kuralı, özellik seçimi, doku sınıflama, öngörü, kenar çıkarma, dalgacık dönüşümü.

ABSTRACT

PhD Thesis

ASSOCIATION RULE EXTRACTION FOR FEATURE SELECTION, CLASSIFICATION AND PREDICTION APPLICATIONS AND SOFTWARE DEVELOPMENT

Murat KARABATAK

Firat University
Graduate School of Natural and Applied Sciences
Department of Electrical - Electronics Engineering
2008, Page: 116

In recent years, together with wide spread use of computer systems, data has been begun to keep in databases and these databases have reached a huge capacity day by day. Therefore, the knowledge discovery from databases has been become an important research area. One of the most important methods in this area is Associated Rules Extraction. In this thesis, the associated rule method is applied on different areas and its performance is evaluated. As intended for this, three different implementations were carried out:

1. Determining relationships between any database quantities, the feature selection application is realized. The obtained results using the feature selection method based on associated rules are compared with the other feature selection methods.
2. The associated rule method is able to extract the relationships between databases. With this feature, the texture classification process using the associated rule method is fulfilled. To increase the success rate and speed, the feature extraction process from textures is fulfilled by using the methods of edge detection and wavelet transformation.
3. Applying associated rule method on student records, the student scores were analyzed and the score predictions for the future were carried out in advance. In addition, software to fulfill this purpose was developed.

Performing these three applications, it can be seen that the associated rule method can reach an important success rate. After the obtained results were compared with the other methods in literature, it has been observed that the associated rule method provides an appreciable success rate on the applications of feature extraction, texture classification and score foreseeing.

Keywords: Association rule, feature selection, texture classification, prediction, edge detection, wavelet transform.

1. GİRİŞ

1.1. Genel

Bilgisayar sistemlerinin olağanüstü bir hızla yaygınlaşması ve günümüzde hemen hemen tüm alanlarda kullanılmasıyla beraber, veriler de artık dijital ortamlarda depolanmaya başlanmıştır. Zamanla depolanan bu bilgiler, öngörü yapılamayacak kadar artış göstermiş ve büyük boyutlara ulaşmıştır. Bu durum, verilerin saklanması için gerekli disklerin kapasitelerinin de artması gerektiği ihtiyacını doğurmuş ve bilgisayarlar daha büyük verileri daha kısa sürede işler hale gelmiştir.

Elektronik veri toplayıcılarının kullanılması, çevrimiçi alışverişin ve elektronik ticaretin yaygınlaşması ve bu alandaki rakip firmaların çalışmaları, veri miktarındaki artışın en önemli sebeplerinden olmuş ve veri madenciliğini ön plana çıkarmıştır. Öyle ki günümüzde yapılan günlük alışverişler, bankacılık işlemleri, devlet ve işletme yönetimlerinde yapılan işlemler, iş yerlerine giriş ve çıkışların kontrolü gibi birçok rutin işlemler, büyük veri tabanlarında kaydedilmektedir.

Eskiden marketlerde kullanılan kasalar sadece müşterilerin o anda satın aldıkları malların toplamını hesaplayan birer toplama makinesinden ibaretti. Oysa günümüzde ise kasa yerine kullanılan terminaller sayesinde yapılan işlemlerin bütün detayları saklanabilmekte, gerek satılan ürünlerin olsun, gerekse müşterilerin olsun, zaman içindeki tüm hareketlerine ulaşmak mümkün olabilmektedir. Bu da göstermektedir ki, bir marketin bile veri tabanı her geçen gün giderek artmaktadır.

Veri tabanı sistemlerinin kullanımındaki bu olağanüstü artış ile elde toplanan veriler, insanları bu verilerden nasıl faydalanabileceği problemi ile karşı karşıya bırakmıştır. Çünkü veri kendi başına değersizdir. Veriden ziyade asıl lazım olan, bu veriler içerisinde amaca uygun bilgiye ulaşmaktır. Veriyi bilgiye çevirmeye veri analizi denir. Bilgi ise, veri analizi sonucunda elde edilen anlamlı ve bir soruya cevap olabilecek nitelikteki verilerdir. Geleneksel sorgu ve raporlama araçları, oldukça büyük olan veri yığınları arasında yetersiz kalmaktadır. Bu nedenle Veri Tabanlarında Bilgi Keşfi (VTBK) adı altında sürekli ve yeni arayışlar ortaya çıkmaktadır. VTBK süreci içerisinde modelin kurulması ve değerlendirilmesi aşamalarında meydana gelen veri madenciliği, en önemli kısmı oluşturmaktadır. Bu nedenden dolayı, birçok araştırmacı tarafından VTBK ve veri madenciliği terimleri eş anlamlı olarak kullanılmaktadır [1].

Market örneğinde veri analizi yaparak her mal için bir sonraki ayın satış öngörüleri çıkarılabilir, müşteriler satın aldıkları mallara bağlı olarak gruplanabilir, yeni bir ürün için

potansiyel müşteriler belirlenebilir, müşterilerin zaman içindeki hareketleri incelenerek onların davranışları ile ilgili öngörüler yapılabilir. Binlerce malın ve müşterinin olabileceği düşünüldüğünde bu analizin gözle ve elle yapılamayacağı, otomatik olarak yapılmasının gerektiği ortaya çıkmaktadır. İşte bu aşamada veri madenciliği devreye girmektedir.

Kısaca veri madenciliği; büyük miktarda veri içinden gelecekle ilgili öngörü yapılmasını sağlayacak bağıntı ve kuralların bilgisayar programları kullanarak aranmasıdır.

Veri madenciliği; makine öğrenmesi, yapay zekâ, istatistik, örüntü tanıma gibi yöntemler kullanılarak anlamlı veriler ve kurallar çıkarmaktır. Yakın geleceğin de, geçmişten çok fazla farklı olmayacağını varsayarsak, çıkarılmış olan bu kurallar gelecekte de geçerli olacak ve ilerisi için doğru öngörüler yapılmasını sağlayacaktır.

Veri madenciliği tekniklerinden biri de birliktelik kuralıdır. Birliktelik kuralı, eldeki verilerden, istenilen ve kayda değer bilgilere ulaşmak için kullanılan bir tekniktir. Birliktelik kuralı, nesnelerin veya niteliklerin bir arada olma durumlarını belirlemekte ve birçok alanda kullanılabilmektedir. Birliktelik kuralı bulma işlemi, yoğun nesne kümesi hesaplamaya dayalı bir işlem olup büyük veri tabanları üzerinde uygulanması oldukça pahalı bir işlemdir. Bu nedenle daha önceden tespit edilen birliktelik kurallarının korunması da oldukça önemli bir konu olmaktadır.

Genellikle büyük süpermarketlerde oluşan satış verilerine, market sepet verisi adı verilmektedir. Birçok kuruluş market sepet verilerinin önemini kullanarak bu verilerden büyük faydalar sağlamayı amaçlamaktadır. Market sepet verisi üzerinde birlikteli kuralı problemi ilk olarak 1993 yılında ele alınmıştır [2]. Sepet analizinde amaç, nitelikler (ürün satışları) arasındaki ilişkiyi bulmaktır. Bu ilişkilerin bilinmesi şirketin kârını arttırmak için kullanılabilir. Eğer *X* malını alan müşterilerin *Y* malını da çok yüksek bir olasılıkla aldıkları biliniyorsa veya bir müşteri *X* malını alıyor ama *Y* malını almıyorsa o potansiyel bir *Y* müşterisidir. Sepet analizi günlük işlemler sonucu elde edilen verilerden anlamlı bağıntılar çıkarmada kullanılır. "Eğer *A* malını alıyorlarsa *% x* ihtimalle *B* malını almaya da meyillidirler" şeklinde bir sonuç *A* malını satan bir mağaza için çok faydalı bir bilgi olabilmektedir.

Birliktelik kuralı daha sonra birçok araştırmada ele alınmış ve birliktelik kuralı üreten algoritmalar geliştirilmiştir. Kaynak [3]'te önerilen Apriori algoritması bu algoritmalar arasında en fazla bilinen ve en yaygın kullanıma sahip algoritmalardan biridir. Yine [3]'te Apriori_TID algoritması da önerilmektedir. [4]'te önerilen Apriori_Hybrid algoritması, Apriori ve Apriori_TID algoritmalarının her ikisini beraber kullanan melez bir algoritmadır. Ayrıca [5]'te çevrimdışı aday nesne kümesi belirleme algoritması, [6, 7]'de SETM algoritması, [8]'de DHP algoritması, [9]'da Partition algoritması, [10]'da MONET algoritması, [11]'de Sampling algoritması, [12]'de DIC algoritması, [13]'te Eclat, Maxeclat, Clique, Maxclique algoritmaları,

[14]'te Max-Miner algoritması ve [15]'te Carma algoritması birliktelik kuralı üreten algoritmalar olarak bilinmektedir.

Birliktelik kuralı üreten algoritmalar, sadece ürün birlikteliklerini dikkate almayıp ürünlerin miktar, ağırlık ve hiyerarşik bilgi düzeni gibi özelliklerini de göz önüne almaktadır. Bunun için [16-20]'de hiyerarşik birliktelik kuralı, [21-27]'de sınırlandırılmış birliktelik kuralı, [28-32]'de nicel birliktelik kuralı, [33-36]'da sıralı örüntü keşfi, [37]'de ağırlıklandırılmış birliktelik kuralı ve [38-40]'ta negatif birliktelik kuralı üreten algoritmalar geliştirilmiştir.

Özellik seçimi, sınıflandırma problemlerinde ve birçok öğrenme algoritmalarında önemli bir yer tutmaktadır. Özellik seçiminde amaç, çok sayıda niteliğe sahip olan bir veri tabanında konu ile ilgili olan en önemli niteliklerin seçilmesidir. Bu sayede daha az işlem yükü ile yüksek başarım elde edilebilmektedir. Özellik seçimi yöntemleri ile ilgili literatürde çok sayıda araştırma bulunmakta ve birçok alanda uygulamaları ile karşılaşılmaktadır [41]. Temel bileşen analizi [42,43] ve doğrusal ayrışım analizi [44], bu alandaki en önemli yöntemlerdendir. Bu yöntemler özellik seçimi yapmaktansa daha çok boyut indirgeme yaptığı için yeterince cazip olamamaktadır. Bu nedenle [45-55]'te önerilen özellik seçimi yöntemleri ile literatürde yaygın olarak karşılaşmak mümkündür. Ancak veritabanındaki tüm niteliklerden kaç tanesinin seçilmesinin daha uygun olacağı konusunda henüz tam bir çözüm önerisi getirilememiştir.

Benzer yapısal özelliklere sahip bölgeleri bulunan yapılar doku olarak tanımlanmaktadır. Doku sınıflama ise bilinmeyen doku örneklerinin belirli bir kural veya kurallar dizisine bağlı olarak, daha önceden bilinen doku sınıflarına atanması işlemi olarak tanımlanmaktadır [56]. Literatürde doku sınıflama yöntemlerine sıkça rastlanmaktadır. Bu çalışmalar [56-63]'te detaylı olarak verilmektedir. Birliktelik kuralı kullanarak doku sınıflama yapılan temel çalışma ise Rushing ve ark. [64] tarafından yapılmıştır. Daha sonra ise [65]'te bu yöntem bulut kümelerinin tanınması uygulamasında kullanılmıştır.

Birliktelik kuralı kullanılarak veri tabanları üzerinden bilgi keşfi yapılabildiğine göre, öğrenci verileri üzerinde de birliktelik kuralı kullanımının faydalı sonuçlar vermesi olası bir durumdur. Literatürde bu konuda yapılan çalışmalara da az da olsa rastlanmaktadır. Bu konuda yapılan temel çalışmalar kaynak [66-69]'da belirtilen çalışmalardır.

1.2. Tezin Amacı

Bu tez çalışmasının genel amacı veri madenciliği tekniklerinden biri olan birliktelik kuralını detaylı bir şekilde incelemek ve kullanım alanlarına yeni yaklaşımlar oluşturmaktır. Özellikle market sepet analizi olarak yaygın kullanıma sahip olan birliktelik kuralı, bu çalışmada özellik seçimi, doku sınıflama ve öğrenci başarısı analizinden yola çıkarak not

öngörüsü yapma gibi alanlara uygulanmıştır. Literatürde bu alanlarda yapılmış benzer çalışmalar olmasına rağmen bu tez çalışmasında birliktelik kuralı, farklı teknikler ile bir araya getirilmiş ve bazı kazanımlar elde edilmiştir.

Bu tezin amaçlarından biri birliktelik kuralının, özellik seçimi amaçlı olarak kullanılmasıdır. Nitelikler arasındaki ilişkileri ortaya çıkarmak için kullanılan birliktelik kuralı sayesinde niteliklerin ne derece etkili olup olamayacağı hakkında bilgiler de üretilebilmektedir. Bu sayede de herhangi bir sınıflandırma probleminde veri tabanına ait nitelik sayısının birliktelik kuralı kullanılarak azaltılması amaçlanmaktadır. Ayrıca literatürde bulunan diğer özellik seçimi yöntemlerine nazaran, seçilecek özellik sayısının da en uygun olacak şekilde elde edilmesi beklenmektedir.

Tezin amaçlarından bir diğeri, birliktelik kuralı ile doku sınıflama işlemidir. Birliktelik kuralı kullanarak doku sınıflama literatürde az da olsa bulunmaktadır. Ancak tezde hedef olarak, birliktelik kuralı kullanarak doku sınıflanmanın yanı sıra hız ve yüksek sınıflama başarımı elde edilmesi amaçlanmaktadır. Bu nedenle, literatürde bulunan farklı yöntemlerin bir araya getirilmesi ile gerekli hız ve başarım artışı kazanımı sağlanması amaçlanmıştır.

Son olarak tezde hedeflenen bir diğer amaç ise birliktelik kuralının eğitim alanında kullanılmasını sağlamak ve öğrenci başarısı analizini yapabilmektir. Öğrencilerin özlük bilgileri ile not bilgileri arasındaki ilişkiler tespit edildiğinde, öğrenci başarısın arttırabilecek yeni yaklaşımlar geliştirmek mümkündür. Bu nedenle öğrenci bilgileri arasından birliktelik kuralları üretilerek öğrenci başarı analizi yapılması ve öğrencinin geleceğe yönelik notları ile ilgili öngörü yapılmasını sağlayacak uygun bir yazılım geliştirilmesi amaçlanmaktadır.

1.3. Tezin Organizasyonu ve Katkılar

Tezin birinci bölümünde, teze genel bir bakış açısı kazanmaya yönelik olarak temel bilgiler verilmiştir. Diğer bölümlerin organizasyonu ve tezdeki orijinal katkılar ise aşağıda sunulmuştur:

Bölüm 2'de, veri madenciliği kavramı tanıtılarak, veri madenciliğinde uygulanan teknikler ve karşılaşılan problemler açıklanmıştır. Ayrıca veri tabanında bilgi keşfi süreci ve veri madenciliği modelleri hakkında bilgiler verilmiştir.

Bölüm 3'te, veri madenciliği modellerinden biri olan birliktelik kuralı detaylı olarak açıklanmıştır. Birliktelik kuralı algoritmalarından biri olan, en çok bilinen ve en yaygın kullanıma sahip olan Apriori algoritmasının çalışma prensibi yine bu bölümde detaylı olarak

işlenmiştir. Ayrıca Apriori algoritması haricindeki diğer birliktelik kuralı algoritmaları ve birliktelik kuralı türleri hakkında genel bilgiler de verilmiştir.

Bölüm 4'te, birliktelik kuralının özellik seçimi amaçlı olarak kullanılması gerçekleştirilmiştir. Hız ve başarım artışı istenen birçok alanda, özellik seçimi işlemine oldukça fazla ihtiyaç duyulabilmektedir. Bu bölümde de birliktelik kuralına dayalı özellik seçimi yapabilmek için iki farklı yöntem önerilmiş ve iki farklı veri tabanı üzerinde bu yöntemler test edilmiştir. Ayrıca elde edilen sonuçlar, diğer özellik seçimi yöntemleri ile de karşılaştırılmıştır.

Bölüm 5'te, birliktelik kuralı kullanarak doku sınıflama işlemi gerçekleştirilmiştir. Literatürde birliktelik kuralı kullanılarak doku sınıflamaya az da olsa rastlanmaktadır. Ancak bu bölümde hem doku sınıflama işlemini hızlandırabilmek hem de sınıflama başarımını arttırabilmek için dokudan iki ayrı yöntem ile özellik çıkarımı yapılmıştır. Hız ve sınıflama başarımının artışını sağlayabilmek için kenar çıkarma yöntemi ve dalgacık dönüşümü yöntemleri ile birliktelik kuralı yöntemi beraber kullanılıp sınıflama işlemi gerçekleştirilmiş ve diğer yöntemlerle karşılaştırmalı sonuçlar verilmiştir.

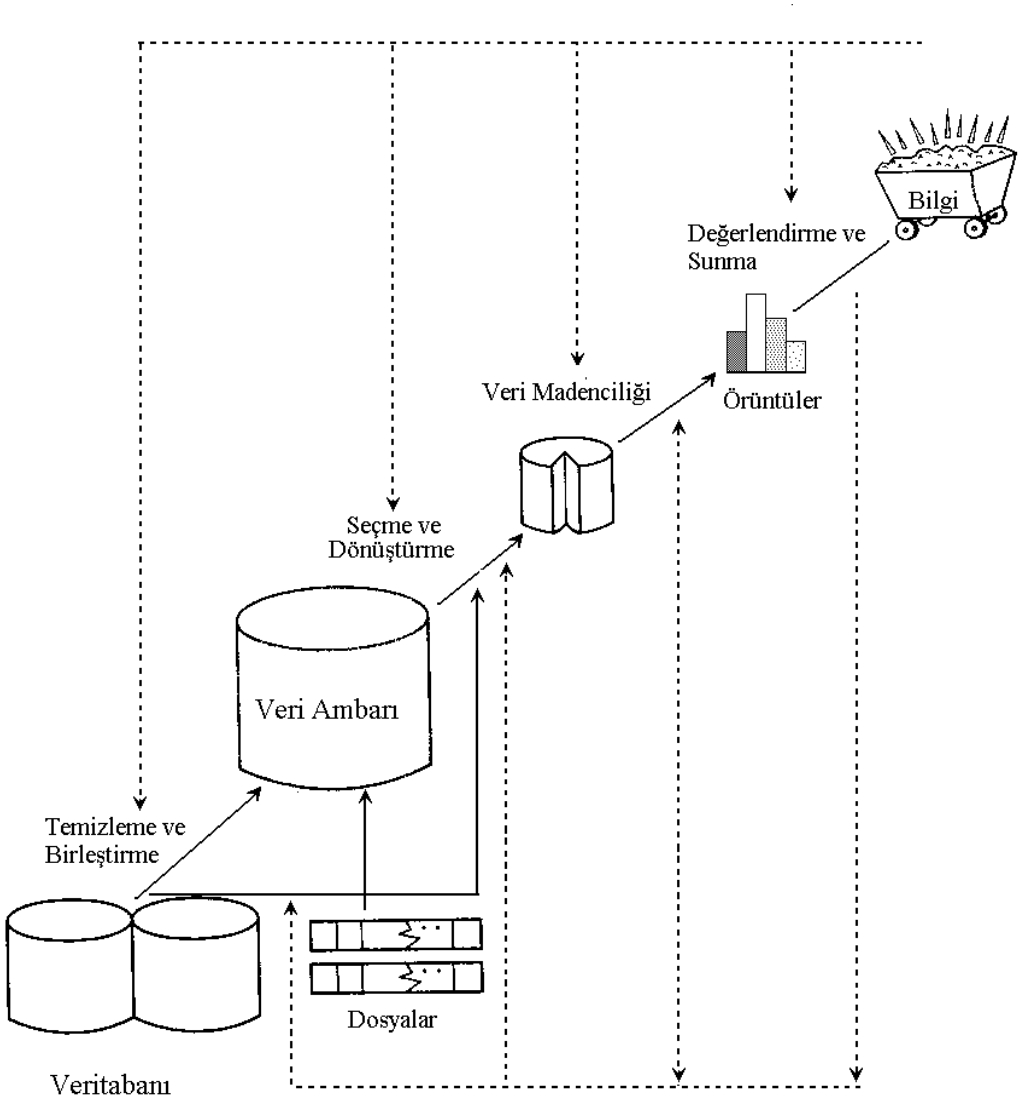
Bölüm 6'da, birliktelik kuralının öğrenci verileri üzerinde uygulaması gerçekleştirilmiştir. Öğrenci not bilgileri ve öğrenci başarısına etki edebilecek özlük bilgileri üzerinden birliktelik kuralları çıkarılmıştır. Böylece, öğrenci bilgilerini kullanarak başarı analizi yapılabilmekte ve öğrencilerin gelecekte alabilecekleri notlarla ilgili öngörüler üretilebilmektedir. Ayrıca bu işlemleri kolaylıkla gerçekleştirebilmek için ise bir yazılım geliştirilmiştir.

Bölüm 7'de, tezin sonuçları irdelenmiş ve tez kapsamında yapılan katkılar vurgulanmıştır.

2. VERİ MADENCİLİĞİ

Veri madenciliği, eldeki verilerden üstü kapalı, çok net olmayan, önceden bilinmeyen ancak potansiyel olarak kullanışlı bilginin çıkarılmasıdır. Bu da; kümeleme, veri özetleme, değişikliklerin analizi, sapmaların tespiti gibi belirli sayıda teknik yaklaşımları içermektedir [70].

Kısaca veri madenciliği; geniş veri tabanlarındaki veriler arasından bilgi çıkarma işlemidir. Kaya ve kum taneleri arasından altın arama işine altın madenciliği denildiği gibi verilerden bilgi keşfetme işine de veri madenciliği denilmektedir. Veri tabanlarından bilgi keşfi, bilgi çıkarma, veri analizi, veri tarama gibi bazı terimler de veri madenciliği ile benzer anlam taşıyan diğer terimlerdir.



Şekil 2.1 Veri tabanında bilginin keşfi

Birçok insan veri madenciliği ile eş anlamlı olarak veri tabanlarında bilgi keşfi terimini kullanmaktadır. VTBK sürecinin nasıl gerçekleştiği Şekil 2.1'de adım adım gösterilmiştir.

- **Veri Temizleme:** Tutarsız ve gürültülü verilerin veri tabanından silinmesidir. Bu aşama, keşfedilecek bilgilerin kalitesini arttıracaktır.
- **Veri Birleştirme:** Birden fazla veri tabanının bulunduğu durumlarda verilerin birleştirilmesidir.
- **Veri Seçimi:** Veri tabanından, konu ile ilgi verilerin bulunup seçilmesidir. Bu adım, birkaç veri kümesini birleştirerek sorguya uygun örneklem kümesini elde etmeyi gerektirir.
- **Veri Madenciliği:** Veriden örüntüler çıkarmak için kullanılan çeşitli yöntemleri içeren en önemli aşamadır.
- **Örüntü/Model Değerlendirme:** Veri tabanından gerçekte ilginç ve doğru olan verilerin tanımlanmasıdır.
- **Bilgi Sunumu:** Keşfedilen ve elde edilen bilgilerin geçerlilik, yenilik, yararlılık ve basitlik kıstaslarına göre değerlendirilmesi ve sunulmasıdır.

Çok geniş veri tabanlarında saklı olan bilgileri elde etmek için uzun yıllar yapılan çalışmalar sonucunda birçok yöntem geliştirilmiştir. Veri madenciliği, yıllardır özellikle batı ülkelerinin üzerinde çalıştığı bir konu olmasına rağmen, gerçek hayatta yazılım endüstrisinin son yıllarda üretmiş olduğu ileri teknoloji ürünü yazılımlar ile kullanılmaya başlanmıştır. Ayrıca veri madenciliği, makine öğrenmesi, istatistik, veri tabanı yönetim sistemleri, veri ambarlama ve paralel programlama gibi farklı disiplinlerde kullanılan yaklaşımları da birleştirmektedir. Belirli bir sınırda yer alması gerekliliği, veri madenciliği algoritmalarında paralel programlama kullanılması ihtiyacını da beraberinde getirmiştir [71].

Veri madenciliğini etkin bir biçimde uygulayabilmek için dikkat edilmesi gereken noktalar aşağıdaki gibi özetlenebilir [72]:

Farklı tipteki verileri ele alma: Gerçek hayattaki uygulamalar makine öğreniminde olduğu gibi yalnızca sembolik veya kategorik veri türleri değil, aynı zamanda tamsayı, kesirli sayılar, çoklu ortam verisi, coğrafi bilgi içeren veri gibi farklı tipteki veriler üzerinde işlem yapılmasını da gerektirir. Kullanılan verinin saklandığı ortam, düz bir kütük veya ilişkisel veri tabanında yer alan tablolar olacağı gibi, nesneye yönelik veri tabanları, çoklu ortam veri tabanları, coğrafi veri tabanları vb. olabilir. Saklandığı ortama göre veri, basit tipte olabileceği gibi karmaşık veri tipleri (çoklu ortam verisi, zaman içeren veri, yardımlı metin, coğrafi, vb.) de olabilir. Bununla birlikte veri tipi çeşitliliğinin fazla olması bir veri madenciliği algoritmasının tüm veri tiplerini ele alabilmesini olanaksızlaştırmaktadır. Bu yüzden veri tipine özgü adanmış veri madenciliği algoritmaları geliştirilmektedir.

Veri madenciliği algoritmasının etkinliği ve ölçeklenebilirliği: Çok büyük oylumlu veri içinden bilgi elde etmek için kullanılan veri madenciliği algoritmasının etkin ve ölçeklenebilir olması gerekir. Bu, veri madenciliği algoritmasının çalışma zamanının öngörü yapılabilir ve kabul edilebilir bir süre olmasını gerektirir. Üssel veya çok terimli bir karmaşıklığına sahip bir veri madenciliği algoritmasının uygulanması kullanışlı değildir.

Sonuçların yararlılık, kesinlik ve anlamlılık kıstaslarını sağlaması: Elde edilen sonuçlar, analiz için kullanılan veri tabanını doğru biçimde yansıtmalıdır. Bunun yanı sıra gürültülü ve aykırı veriler ele alınmalıdır. Bu işlem elde edilen kuralların kalitesini belirlemede önemli bir rol oynar.

Keşfedilen kuralların çeşitli biçimlerde gösterimi: Bu özellik keşfedilen bilginin gösterim biçiminin seçilebilmesini sağlayan yüksek düzeyli bir dil tanımının yapılmasını ve grafik ara yüzünü gerektirir.

Farklı birkaç soyutlama düzeyi ve etkileşimli veri madenciliği: Büyük veri tabanlarından elde edilecek bilgi ile ilgili öngörü yapılması zordur. Bu yüzden veri madenciliği sorgusu, elde edilen bilgilere göre kullanıcıya etkileşimli olarak sorgusunu değiştirebilmeyi, farklı açılardan ve farklı soyutlama düzeylerinden keşfedilen bilgiyi inceleyebilme esnekliğini sağlamalıdır.

Farklı ortamlarda yer alan veri üzerinde işlem yapabilme: Kurumlar yerel ağlar üzerinden pek çok dağıtık ve heterojen veri tabanı üzerinde işlem yapmaktadır. Bu veri madenciliğinin farklı kaynaklarda birikmiş formatlı ya da formatsız veriler üzerinde analiz yapabilmesini gerektirir. Verinin büyüklüğünün yanı sıra dağıtık olması, yeni araştırma alanlarının ortaya çıkmasına sebep olmuştur. Bunlar, paralel ve dağıtık veri madenciliği algoritmalarıdır.

Gizlilik ve veri güvenliğinin sağlanması: Bir VTBK sisteminde keşfedilen bilgi pek çok farklı açıdan ve soyutlama düzeyinden izlenebildiği için, gizlilik ve veri güvenliği, veri madenciliği sistemini kullanan kullanıcının haklarına ve erişim yetkilerine göre sağlanmalıdır.

Veri madenciliği astronomi, biyoloji, finans, pazarlama, tıp ve daha birçok alanda uygulanmaktadır. Son 20 yıldır Amerika'da çeşitli veri madenciliği algoritmalarının, vergi kaçakçılıklarını ortaya çıkartmaya kadar birçok alanda kullanıldığı bilinmektedir. [73]

Bununla birlikte günümüzde veri madenciliği teknikleri özellikle işletmelerde çeşitli alanlarda başarı ile kullanılmaktadır. Bu uygulamalardan başlıcaları ilgi alanlarına göre aşağıda özetlenmiştir [1].

Pazarlama

- Müşterilerin satın alma örüntülerinin belirlenmesi,
- Müşterilerin demografik özellikleri arasındaki bağlantıların bulunması,
- Posta kampanyalarında cevap verme oranının artırılması,
- Mevcut müşterilerin elde tutulması, yeni müşterilerin kazanılması,
- Pazar sepeti analizi,
- Müşteri ilişkileri yönetimi,
- Müşteri değerlendirme,
- Satış öngörüsü.

Bankacılık

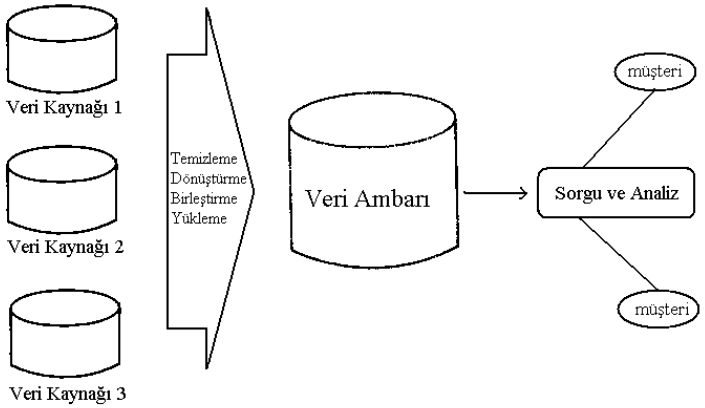
- Farklı finansal göstergeler arasında gizli korelasyonların bulunması,
- Kredi kartı dolandırıcılıklarının tespiti,
- Kredi kartı harcamalarına göre müşteri gruplarının belirlenmesi,
- Kredi taleplerinin değerlendirilmesi.

Sigortacılık

- Yeni poliçe talep edecek müşteriler ile ilgili öngörü yapılması,
- Sigorta dolandırıcılıklarının tespiti,
- Riskli müşteri örüntülerinin belirlenmesi.

2.1. Veri Ambarları

Veri madenciliği büyük miktarlardaki verileri inceleme amacı üzerine kurulmuş olduğu için veri tabanları ile yakından ilişkilidir. Gerekli verinin hızla ulaşılabilecek, amaca uygun bir şekilde saklanması ve gerektiğinde hızla ulaşılabilmesi gerekmektedir. Günümüzde yaygın olarak kullanılmaya başlanan veri ambarları günlük kullanılan veri tabanlarının birleştirilmiş ve işlemeye daha uygun bir özetini saklamayı amaçlamaktadır. Günlük veri tabanlarından istenen özet bilgi seçilip, gerekli önişleme yapıldıktan sonra veri ambarında saklanmaktadır. Ardından amaç doğrultusunda gerekli veri ambardan alınarak veri madenciliği çalışması için standart bir forma çevrilmektedir.



Şekil 2.2 Bir firma için tipik bir veri ambarı

Veri ambarında veri oluşturulduktan sonra bu verinin elle veya gözle analizi yapılabilmektedir. Bunun için Çevrimiçi Analitik İşleme (ÇAİ) programları kullanılmaktadır. Bu programlar, her boyutu veride bir alana karşılık gelen çok boyutlu bir küp olarak veriye bakmayı ve incelemeyi sağlamaktadırlar. Böylece boyut bazında guruplama, boyutlar arasındaki korelasyonları inceleme ve sonuçları grafik veya rapor olarak sunma olanağı sağlanmaktadır [74].

2.1.1. Çevrimiçi Analitik İşleme ve Veri Madenciliği

Çoğu kişi tarafından, çevrimiçi analitik işleme ve veri madenciliği kavramlarını birbirine karıştırılmaktadır. Her ikisi de veri ambarı üzerinde yürütülen iki önemli fonksiyondur. Öz bilginin yönetimi açısından her iki fonksiyonun da amacı, ham veri içinde gizli duran işle ilgili, yararlı bilgileri ortaya çıkarmaktır. Çevrimiçi analitik işleme ile veri madenciliği birbirini tamamlayan öğeler olmasına rağmen, çevrimiçi analitik işleme veri madenciliğinden farklıdır.

Veri madenciliğinde amaç, kullanıcının bilgi çıkarma sürecinde katkısının olabildiğince az tutması, işin olabildiğince otomatik olarak yapılabilmesidir. Çünkü çevrimiçi analitik işleme programını kullanırken bulunabilecek sonuçlar kullanıcının sormayı düşündüğü sorgularla sınırlıdır. Ama veri içindeki çocuk bezi ile bira örneğindeki bağıntı gibi kullanıcının hiç aklına gelmeyecek bilgiler de olabilir. Zaten veri madenciliğinde amaç bu tip bilgileri bulabilmektir. Veri madenciliğinde, veriye bağlı olarak bilgi büyük veri tabanlarından çekilip çıkarılır. Veri madenciliğinde, veri içinde örtülü olarak bulunan belirli bir örüntü açığa çıkarılır. Kullanıcı, ortaya çıkarılan bu olgulara bakarak, olayın önemini anlar. Bu işlemde insan etkeni oldukça önemlidir. Veri madenciliği süreci genelde öz bilginin oluşturulması ile son bulur.

Veri madenciliğinde; istatistik, matematik, makine öğrenmesi ve yapay zekâ disiplinlerinden oldukça fazla yararlanılmaktadır. IBM, veri madenciliği sürecini çevrimiçi

analitik işleme sürecine üstün kılan özelliğini şu şekilde açıklamaktadır. Veri madenciliği tekniği, "ne?" sorusunun arkasında yatan "niçin?" sorusundan başka "başka ne?" ve "neden o?" gibi soruları yanıtlayarak çevrimiçi analitik işleme tekniğini geçmeye çalışır [75].

Veriye birden fazla perspektiften bakma imkânı sağladığından çevrimiçi analitik işlemeyi bir küp şeklinde hayal etmek gerekir. Standart sorgulama dilinin (*Structured Query Language-SQL*) saatle veya gün ile ölçülen cevaplama süresi, çevrimiçi analitik işlemede dakikalarla ölçülmektedir. Bunun dışında elle analiz edebilme imkânı da sağlamaktadır. Çevrimiçi analitik işlemeye verilen diğer bir isim çok boyutlu veri tabanıdır. İyi tasarlanmış bir çevrimiçi analitik işleme küpünde her kayıt sadece bir alt kümeye girmelidir. Bu, çok boyutlu veri tabanı olmanın en önemli kuralıdır [76].

Bunun yanında çok boyutluluk çevrimiçi analitik işlemenin en temel özelliğidir. Çok boyutlu görünümler, verinin üretildiği veya yakalandığı biçimde görünmek istenmeyip, bilginin bir iş kullanıcısı gözüyle algılanmak istenmesi çabasıdır. Örneğin kullanıcı sadece satış verisini görmek istemeyecek, aynı zamanda belirli bir ürüne veya belirli bir zaman periyoduna yönelik satış bilgilerini görmek isteyecektir. Ürün, zaman ve periyodun her biri, satış verisinin boyutlarıdır. Kullanıcı çoğunlukla, verinin kendisine, aynı anda çeşitli boyutlarda düzenlenerek sunulmasını ister. Örneğin bir kullanıcı, geçen yıla ilişkin satışları, ürünlere, müşteriye, satış temsilcisine, dağıtım kanalına ve bölgeye göre görmek isteyebilir. Çevrimiçi analitik işleme sistemleri kullanıcılara verinin çok boyutlu görünümlerini doğal olarak sunmakta ve onları karmaşık sorgu sentaksından yalıtmaktadır [75]. Son olarak, gerçek işletme problemlerini modelleyebilme yeteneğini ve kullanıcıların kaynakları daha verimli kullanma imkânı sağlaması çevrimiçi analitik işlemeyi günümüzde işletmeler için vazgeçilmez kılmaktadır. Pazar taleplerinin daha hızlı cevaplanmasını sağlayan çevrimiçi analitik işleme bu sayede işletmelerin yatırımlarının geri dönüş sürelerini kısaltmaktadır [77].

2.1.2. Veri Ambarının Yapısı

Birçok kaynakta bulunan veriyi bir araya toplayarak veri madenciliği için katalizör görevi gören veri ambarında, verinin akışı ve son kullanıcıya kadar ulaşma süreci yakından incelenecek olursa, veri ambarının çok katmanlı yapısı ile karşılaşılmaktadır. Bu çok katmanlı veri kısaca özetlenirse [76];

- **Kaynak Sistemler:** Verinin ilk elden toplandığı ve soyutlama seviyesinin en düşük olduğu kısımdır. Operasyonel anlamda yararlanılan verinin karar destek için kullanılmasını söz konusu değildir.

- **Veri Nakli ve Temizlenmesi:** Bu kısımda veriyi kaynak sistemlerden çıkararak veri ambarı ve analiz ortamına nakleden yazılımlar kullanılır.
- **Merkezi Depo:** Veri ambarının teknik olarak en gelişmiş kısmıdır. Veriyi içinde bulunduran çok büyük bir veri tabanıdır.
- **Meta Veri:** Meta veri verinin fiziksel alt yapısını hazırlamaktadır. Analiz için gerekli olan kısımların ön plana çıkarılmasına ve indeks, tablo, alan sayılarının belirlenmesine çalışılır. Veriye kendisi hakkında bilgi sağlar. Çoğu zaman veri ambarı çerçevesinde göz ardı edilen bir konudur.
- **Datamartlar:** Bir işletmede aynı anda farklı bilgilere ihtiyaç duyan insanlar olacaktır. Verinin tümü veri ambarında olduğuna göre aynı anda bu veri ambarından farklı bilgiler sağlamak mümkün mü? Sorusunun cevabı datamart'tadır. Datamartlar bir departman için gerekli olan bilgiyi merkezileştirme özelliğine sahip bir sistemdir.
- **Operasyonel Geri Besleme:** Bu noktaya kadar olan veri işleme sonuçlarının geri besleme olarak operasyonel sisteme verilmesi sürecidir. Veri madenciliğinin hayati döngüsünü tamamlama yeteneğine sahip bir süreçtir. Bu nedenle oldukça önemlidir.
- **Son Kullanıcı:** Veri ambarının yapısı içindeki en önemli kısımdır. Son kullanıcıdan amaç analizciler, uygulama geliştiriciler ve işletmecilerdir.

2.1.3. Veri Madenciliği ve Veri Ambarı

Son yıllarda hemen hemen her alanda veri ambarı kullanılmaya başlanmıştır. Günümüzde hipermarket satışlarından bankacılığa, astronomiden fiziğe birçok alanda büyük veri tabanları kullanılmaktadır. Veri madenciliğinin kaynak olarak değerlendirildiği alan veri ambarlarıdır. Veri madenciliği bilginin veri ambarlarından çekilip çıkarıldığı araçlar kümesi sağlamaktadır [78].

Verinin bir arada toplu bir şekilde bulunduğu veri ambarında hareket edilebilir bilgi üretmek oldukça zordur. Raporlama ve faturalama gibi faaliyetleri kolaylaştıran veri ambarı, bilgi çıkarımı konusunda çok etkili değildir. Fakat bu konuda, veri madenciliğine yardımcı olmaktadır. Verinin bir arada, temizlenmiş olarak bulunması ve veri madenciliğinin hayati döngüsünü tamamlayıcı özelliği, veri ambarının veri madenciliği için ne kadar önemli olduğunu kısaca özetlemektedir.

Bilindiği gibi veri madenciliğini standart istatistiksel yöntemlere üstün kılan özelliği, çok fazla miktarda veriyle çalışılabilir olmasıdır. Standart istatistikte, ana kütleden seçilen bir örneklem üzerinde çalışarak genelleştirme yapılmaya çalışılır. Fakat bu durumun, gelecekteki

işletme ihtiyaçları ile ilgili tam olarak öngörü yapamama, iş çevresindeki gelişmelere ve değişimlere cevap verememe gibi olumsuz yönleri vardır.

Bu amaçla pahalı da olsa veri madenciliği tekniklerini uygulamak daha isabetli karar verilmesini sağlamaktadır. Tüm veriyle çalışan veri madenciliğinde, bütün veriyi sağlayan organ veri ambarıdır. Bu amaçla veri madenciliğinin, veri ambarına göründüğünden daha fazla ihtiyacı vardır.

Bunun dışında, veri ambarı veri madenciliğinde kullanılacak veriyi temizlemektedir. Veri ambarının olmaması durumunda veri madenciliği süreci gereğinden fazla uzamaktadır. Ayrıca, veri ambarı çok basit ve cevabı kullanışlı olacak soruların cevabını hızlı bir şekilde alarak veri madenciliğinin işini kolaylaştırmaktadır. Yapılan bir kampanyanın sonuçlarının başarısını belirlemek gibi geri beslemesi yüksek olan noktaları belirlemede ise veri ambarı oldukça etkilidir.

2.2. Veri Madenciliğinde Karşılaşılan Problemler

Veri madenciliği uygulamalarında kullanılan esas öğe veri tabanı olduğuna göre, veri tabanından kaynaklanabilecek birçok problemle karşılaşmak mümkündür. En azından veri tabanı, normalizasyon şartlarını taşımalıdır. Özellikle küçük veri tabanlarında çalışan sistemler hızlı ve doğru bir biçimde çalışabilirken, çok büyük veri tabanlarında tamamen farklı davranabilir. Veriye gürültü de eklendiğinde, başarım çok kötü bir biçimde etkilenebilir. Veri madenciliği uygulamalarında genel olarak aşağıdaki problemlerle karşılaşılmaktadır [74]:

- Veri tabanının boyutu
- Gürültülü veri
- Boş değerler
- Eksik veri
- Artık veri
- Dinamik veri

2.2.1. Veri Tabanının Boyutu

Veri tabanı boyutları inanılmaz bir hızla artmaktadır. Pek çok makine öğrenmesi algoritması bir kaç yüz tutanaklık oldukça küçük örneklemleri ele alabilecek biçimde geliştirilmiştir. Aynı algoritmaların yüz binlerce kat büyük örneklemlerde kullanılabilmesi için azami dikkat gerekmektedir. Örneklemin büyük olması, örüntülerin gerçekten var olduğunu göstermesi açısından bir avantajdır ancak böyle bir örneklemden elde edilebilecek olası örüntü sayısı çok büyüktür. Bu yüzden veri madenciliği sistemlerinde karşılaşılan en önemli

sorunlardan biri veri tabanı boyutunun çok büyük olmasıdır. Dolayısıyla veri madenciliği yöntemleri ya sezgisel/buluşsal bir yaklaşımla arama uzayını taramalıdır ya da örneklemi yatay/dikey olarak indirgemelidir.

Yatay indirgeme, nitelik değerlerinin önceden belirlenmiş genelleme sıradüzenine göre, bir üst nitelik değeri ile değiştirilme işlemi yapıldıktan sonra aynı olan çokluların çıkarılması işlemidir. Dikey indirgeme, artık niteliklerin indirgenmesi işlemidir. Özellik seçimi yöntemleri ya da nitelik bağımlılık çizelgesi uygulanarak yapılır.

2.2.2. Gürültülü Veri

Büyük veri tabanlarında pek çok niteliğin değeri yanlış olabilir. Bu hata, veri girişi sırasında yapılan insan hataları veya girilen değerin yanlış ölçülmesinden kaynaklanır. Veri girişi ya da veri toplanması sırasında oluşan sistem dışı hatalara gürültü adı verilir. Ancak günümüzde kullanılan ticari ilişkisel veri tabanları veri girişi sırasında oluşan hataları otomatik biçimde gidermek konusunda az bir destek sağlamaktadır. Hatalı veri gerçek dünya veri tabanlarında ciddi problemler oluşturabilmektedir. Bu durum, bir veri madenciliği yönteminin kullanılan veri kümesinde bulunan gürültülü verilere karşı daha az duyarlı olmasını gerektirmektedir. Gürültülü verinin yol açtığı problemler tümevarımsal karar ağaçlarında uygulanan yöntemler bağlamında kapsamlı bir biçimde araştırılmıştır. Eğer veri kümesi gürültülü ise sistem bozuk veriyi tanımalı ve ihmal etmelidir. Quinlan [79], gürültünün sınıflama üzerindeki etkisini araştırmak için bir dizi deney yapmıştır. Deneysel sonuçlar, etiketli öğrenmede etiket üzerindeki gürültü öğrenme algoritmasının başarımını doğrudan etkileyerek düşmesine sebep olmuştur. Buna karşın eğitim kümesindeki nesnelerin özellikleri/nitelikleri üzerindeki en çok %10'luk gürültü miktarı ayıklanabilmektedir. Chan ve Wong [80], gürültünün etkisini analiz etmek için istatistiksel yöntemler kullanmışlardır.

2.2.3. Boş Değerler

Veri tabanlarında boş değer birincil anahtarda yer almayan herhangi bir niteliğin değeri olabilir. Birçokluda eğer bir nitelik değeri boş ise o nitelik bilinmeyen ve uygulanamaz bir değere sahiptir. Bu durum ilişkisel veri tabanlarında sıkça karşımıza çıkmaktadır. Bir ilişkide yer alan tüm çoklular aynı sayıda niteliğe, niteliğin değeri boş olsa bile, sahip olmalıdır. Örneğin kişisel bilgisayarların özelliklerini tutan bir ilişkide bazı model bilgisayarlar için ses kartı modeli niteliğinin değeri boş olabilir.

Lee [81], boş değeri, bilinmeyen, uygulanamaz ve bilinmeyen veya uygulanamaz olacak biçimde üçe ayıran bir yaklaşımı ilişkisel veri tabanlarını genişletmek için öne

sürmüştür. Mevcut boş değer taşıyan veri için herhangi bir çözüm sunmayan bu yaklaşımın dışında bu konuda sadece bilinmeyen değer üzerinde çalışmalar yapılmıştır [82, 83]. Boş değerli nitelikler veri kümesinde bulunuyorsa, ya bu çoklular tamamıyla ihmal edilmeli ya da bu çoklularda niteliğe olası en yakın değer atanmalıdır [84].

2.2.4. Eksik Veri

Evrendeki her nesnenin ayrıntılı bir biçimde tanımlandığını ve bu nesnelerin alabileceği değerler kümesinin belirli olduğu varsayılsın. Verilen bir bağlamda her bir nesnenin tanımı kesin ve yeterli olsa idi, sınıflama işlemi basitçe nesnelerin altkümelerinden faydalanılarak yapılırdı. Bununla birlikte veriler kurum ihtiyaçları göz önünde bulundurularak düzenlenip toplandığından, mevcut veri gerçek hayatı yeterince yansıtmayabilir. Örneğin hastalığın tanısını koymak için kurallar sadece çok yaşlı insanların belirtilerinin bulunduğu bir veri kümesi kullanılarak üretilseydi, bu kurallara dayanarak bir çocuğa tanı koymak pek doğru olmazdı. Bu gibi koşullarda bilgi keşif modeli, belirli bir güvenlik derecesinde öngörüsel kararlar alabilmelidir.

2.2.5. Artık Veri

Verilen veri kümesi, eldeki probleme uygun olmayan veya artık nitelikler içerebilmektedir. Bu duruma, pek çok işlem sırasında karşılaşmak mümkündür. Örneğin eldeki problem ile ilgili verinin elde edilmesi için iki ilişki birleştirildiğinde, elde edilen ilişkide kullanıcının farkında olmadığı artık niteliklerin ortaya çıkması olası bir durumdur. Artık nitelikleri elemek için geliştirilmiş algoritmalar özellik seçimi olarak adlandırılmaktadır.

Özellik seçimi, tümevarıma dayalı öğrenmede budama öncesi yapılan işlem, hedef bağlamı tanımlamak için yeterli ve gerekli olan niteliklerin küçük bir alt kümesinin seçimi problemidir. Özellik seçimi yalnız arama uzayını küçültmekle kalmayıp, sınıflama işleminin kalitesini de arttırmaktadır [85, 86].

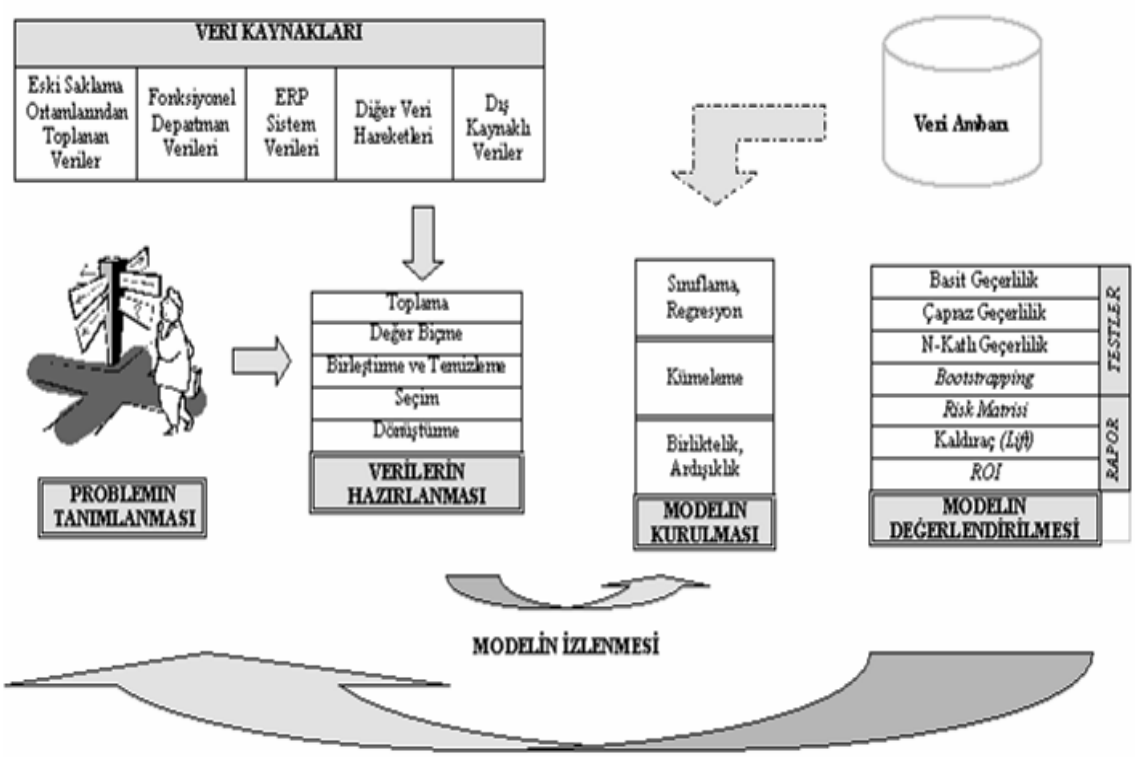
2.2.6. Dinamik Veri

Kurumsal çevrimiçi veri tabanları dinamiktir, yani içeriği sürekli olarak değişmektedir. Bu durum, bilgi keşfi yöntemleri için önemli sakıncalar doğurmaktadır. İlk olarak sadece okuma yapan ve uzun süre çalışan bilgi keşfi metodu bir veri tabanı uygulaması olarak mevcut veri tabanı ile birlikte çalıştırıldığında mevcut uygulamanın da başarımı ciddi ölçüde düşmektedir. Diğer bir sakınca ise, veri tabanında bulunan verilerin kalıcı olduğu varsayılıp, çevrimdışı veri üzerinde bilgi keşif metodu çalıştırıldığında, değişen verinin elde edilen örüntülere yansıması

gerekmektedir. Bu işlem, bilgi keşfi metodunun ürettiği örüntüleri zaman içinde değişen veriye göre sadece ilgili örüntüleri yığmalı olarak günleme yeteneğine sahip olmasını gerektirir. Aktif veri tabanları tetikleme mekanizmalarına sahiptir ve bu özellik bilgi keşif yöntemleri ile birlikte kullanılabilir.

2.3. Veri Tabanlarında Bilgi Keşfi Süreci

Ne kadar etkin olursa olsun hiç bir veri madenciliği algoritması, üzerinde inceleme yapılan işin ve verilerin özelliklerinin bilinmemesi durumunda fayda sağlaması mümkün değildir. Bu nedenle aşağıda tanımlanan tüm aşamalardan önce, iş ve veri özelliklerinin öğrenilmesi / anlaşılması başarının ilk şartı olacaktır [1].



Şekil 2.3 Veri tabanında bilgi keşfi süreci

Şekil 2.3'te ayrıntılı olarak görüldüğü gibi,
Problemin Tanımlanması,
Verilerin Hazırlanması,
Modelin Kurulması ve Değerlendirilmesi,
Modelin Kullanılması,
Modelin İzlenmesi,
veri tabanlarında bilgi keşfi sürecinde izlenmesi gereken temel aşamalardır.

2.3.1. Problemin Tanımlanması

Veri madenciliği çalışmalarında başarılı olmanın ilk şartı, uygulamanın hangi işletme amacı için yapılacağının açık bir şekilde tanımlanmasıdır. İlgili işletme amacı işletme problemi üzerine odaklanmış ve açık bir dille ifade edilmiş olmalı, elde edilecek sonuçların başarı düzeylerinin nasıl ölçüleceği tanımlanmalıdır. Ayrıca yanlış öngörülerde katlanılacak olan maliyetlere ve doğru öngörülerde kazanılacak faydalara ilişkin öngörülere de bu aşamada yer verilmelidir.

2.3.2. Verilerin Hazırlanması

Modelin kurulması aşamasında ortaya çıkacak sorunlar, bu aşamaya sık sık geri dönülmesine ve verilerin yeniden düzenlenmesine neden olacaktır. Bu durum verilerin hazırlanması ve modelin kurulması aşamaları için, bir analistin veri keşfi sürecinin toplamı içerisinde enerji ve zamanının % 50 - % 85'ini harcamasına neden olmaktadır.

Verilerin hazırlanması aşaması kendi içerisinde toplama, değer biçme, birleştirme temizleme, seçme ve dönüştürme adımlarından meydana gelmektedir.

2.3.2.1. Toplama

Tanımlanan problem için gerekli olduğu düşünülen verilerin ve bu verilerin toplanacağı veri kaynaklarının belirlenmesi adımıdır. Verilerin toplanmasında kuruluşun kendi veri kaynaklarının dışında, nüfus sayımı, hava durumu, merkez bankası kara listesi gibi veri tabanlarından veya veri pazarlayan kuruluşların veri tabanlarından faydalanılabilir.

2.3.2.2. Değer Biçme

Veri madenciliğinde kullanılacak verilerin farklı kaynaklardan toplanması, doğal olarak veri uyumsuzluklarına neden olacaktır. Bu uyumsuzlukların başlıcaları farklı zamanlara ait olmaları, kodlama farklılıkları (örneğin bir veri tabanında cinsiyet özelliğinin e/k, diğer bir veri tabanında 0/1 olarak kodlanması), farklı ölçü birimleridir. Ayrıca verilerin nasıl, nerede ve hangi koşullar altında toplandığı da önem taşımaktadır.

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

2.3.2.3. Birleştirme ve Temizleme

Bu adımda, farklı kaynaklardan toplanan verilerde bulunan ve bir önceki adımda belirlenen sorunlar mümkün olduğu ölçüde giderilerek veriler tek bir veri tabanında toplanır. Ancak basit yöntemlerle ve baştan savma olarak yapılacak sorun giderme işlemlerinin, ileriki aşamalarda daha büyük sorunların kaynağı olacağı unutulmamalıdır.

2.3.2.4. Seçim

Bu adımda, kurulacak modele bağlı olarak veri seçimi yapılır. Örneğin öngörü yapan bir model için, bu adım bağımlı ve bağımsız değişkenlerin ve modelin eğitiminde kullanılacak veri kümesinin seçilmesi anlamını taşımaktadır.

Sıra numarası, kimlik numarası gibi anlamlı olmayan ve diğer değişkenlerin modeldeki ağırlığının azalmasına da neden olabilecek değişkenlerin modele girmemesi gerekmektedir. Bazı veri madenciliği algoritmaları konu ile ilgisi olmayan bu tip değişkenleri otomatik olarak elese de, pratikte bu işlemin kullanılan yazılıma bırakılmaması daha akılcı olacaktır.

Verilerin görselleştirilmesine olanak sağlayan grafik araçlar ve bunların sunduğu ilişkiler, bağımsız değişkenlerin seçilmesinde önemli yararlar sağlayabilir. Genellikle yanlış veri girişinden veya bir kereye özgü bir olayın gerçekleşmesinden kaynaklanan verilerin, önemli bir uyarıcı enformasyon içerip içermediği kontrol edildikten sonra veri kümesinden atılması tercih edilir.

Modelde kullanılan veri tabanının çok büyük olması durumunda tesadüfîliği bozmayacak şekilde örnekleme yapılması uygun olabilir. Günümüzde hesaplama olanakları ne kadar gelişmiş olursa olsun, çok büyük veri tabanları üzerinde çok sayıda modelin denenmesi zaman kısıtı nedeni ile mümkün olamamaktadır. Bu nedenle tüm veri tabanını kullanarak bir kaç model denemek yerine, tesadüfî olarak örneklenmiş bir veri tabanı parçası üzerinde birçok modelin denenmesi ve bunlar arasından en güvenilir ve güçlü modelin seçilmesi daha uygun olacaktır.

2.3.2.5. Dönüştürme

Kredi riskinin öngörüsü için geliştirilen bir modelde, borç/gelir gibi önceden hesaplanmış bir oran yerine, ayrı ayrı borç ve gelir verilerinin kullanılması tercih edilebilir. Ayrıca modelde kullanılan algoritma, verilerin gösteriminde önemli rol oynayacaktır. Örneğin bir uygulamada bir Yapay Sinir Ağı (YSA) algoritmasının kullanılması durumunda, kategorik değişken değerlerinin evet/hayır olması; bir karar ağacı algoritmasının kullanılması durumunda

ise örneğin gelir değişken değerlerinin yüksek/orta/düşük olarak gruplanmış olması modelin etkinliğini artıracaktır.

2.3.3. Modelin Kurulması ve Değerlendirilmesi

Tanımlanan problem için en uygun modelin bulunabilmesi, olabildiğince çok sayıda modelin kurularak denenmesi ile mümkündür. Bu nedenle veri hazırlama ve model kurma aşamaları, en iyi olduğu düşünülen modele varılıncaya kadar yinelenen bir süreçtir.

Model kuruluş süreci, denetimli ve denetimsiz öğrenmenin kullanıldığı modellere göre farklılık göstermektedir.

Örnekten öğrenme olarak da isimlendirilen denetimli öğrenmede, bir denetçi tarafından ilgili sınıflar önceden belirlenen bir kritere göre ayrılarak, her sınıf için çeşitli örnekler verilir. Sistemin amacı verilen örneklerden hareket ederek her bir sınıfa ilişkin özelliklerin bulunması ve bu özelliklerin kural cümleleri ile ifade edilmesidir.

Öğrenme süreci tamamlandığında, tanımlanan kural cümleleri verilen yeni örneklere uygulanır ve yeni örneklerin hangi sınıfa ait olduğu kurulan model tarafından belirlenir.

Denetimsiz öğrenmede, kümeleme analizinde olduğu gibi ilgili örneklerin gözlenmesi ve bu örneklerin özellikleri arasındaki benzerliklerden hareket ederek sınıfların tanımlanması amaçlanmaktadır.

Denetimli öğrenmede seçilen algoritmaya uygun olarak ilgili veriler hazırlandıktan sonra, ilk aşamada verinin bir kısmı modelin öğrenimi, diğer kısmı ise modelin geçerliliğinin test edilmesi için ayrılır. Modelin öğrenimi, öğrenim kümesi kullanılarak gerçekleştirildikten sonra, test kümesi ile modelin doğruluk derecesi belirlenir.

Kurulan modelin değerinin belirlenmesinde kullanılan diğer bir ölçü, model tarafından önerilen uygulamadan elde edilecek kazancın bu uygulamanın gerçekleştirilmesi için katlanılacak maliyete bölünmesi ile edilecek olan yatırımın geri dönüş oranıdır.

Kurulan modelin doğruluk derecesi ne kadar yüksek olursa olsun, gerçek dünyayı tam anlamı ile modellediğini garanti edebilmek mümkün değildir. Yapılan testler sonucunda geçerli bir modelin doğru olmamasındaki başlıca nedenler, model kuruluşunda kabul edilen varsayımların ve modelde kullanılan verilerin doğru olmamasıdır. Örneğin modelin kurulması sırasında varsayılan enflasyon oranının zaman içerisinde değişmesi, bireyin satın alma davranışını belirgin olarak etkileyecektir.

2.3.4. Modelin Kullanılması

Kurulan ve geçerliliği kabul edilen model doğrudan bir uygulama olabileceği gibi, bir başka uygulamanın alt parçası olarak ta kullanılabilmektedir. Kurulan modeller risk analizi, kredi değerlendirme, dolandırıcılık tespiti gibi işletme uygulamalarında doğrudan kullanılabileceği gibi, promosyon planlaması simülasyonuna entegre edilebilir veya öngörü yapılan envanter düzeyleri yeniden sipariş noktasının altına düştüğünde, otomatik olarak sipariş verilmesini sağlayacak bir uygulamanın içine gömülebilir.

2.3.5. Modelin İzlenmesi

Zaman içerisinde bütün sistemlerin özelliklerinde ve dolayısıyla ürettikleri verilerde ortaya çıkan değişiklikler, kurulan modellerin sürekli olarak izlenmesini ve gerekiyorsa yeniden düzenlenmesini gerektirecektir. Öngörüsü yapılan ve gözlenen değişkenler arasındaki farklılığı gösteren grafikler model sonuçlarının izlenmesinde kullanılan yararlı bir yöntemdir.

2.4. Veri Madenciliği Modelleri

Veri madenciliği modelleri, temelde iki ana başlıkta incelenmektedir. Birincisi, elde edilen örüntülerden sonuçları bilinmeyen verilerin öngörüsü için kullanılan öngörü yapan model, diğeri ise eldeki verinin tanımlanmasını sağlayan tanımlayıcı modeldir [1].

Öngörü yapan modellerde, sonuçları bilinen veriler kullanılarak bir model geliştirilir. Oluşturulan bu model kullanılarak sonuçları bilinmeyen veri kümeleri için sonuç değerleri ile ilgili öngörü yapılması amaçlanmaktadır. Örneğin bir banka, daha önceki dönemlerde müşterilerine verdiği tüm kredilerle ilgili bilgilere sahiptir. Bu bilgileri kullanarak daha sonraki dönemlerde müşterilere vereceği kredinin geri dönüp dönmeyeceğini müşteri bilgilerini kullanarak öngörü yapılabilir.

Tanımlayıcı modeller ise karar vermeye rehberlik etmede kullanılabilecek mevcut verilerdeki örüntülerin tanımlanmasını sağlamaktadır. Belirli özelliklere sahip insanların bazı davranışlarının birbirine benzerlik göstermesi tanımlayıcı modele bir örnek olabilir.

Veri madenciliği modellerini gördükleri işlevlere göre ise üç ana başlık altında incelemek mümkündür. Bunlar;

- Sınıflama ve Regresyon,
- Kümeleme,
- Birliktelik kuralları ve ardışık zamanlı örüntülerdir.

2.4.1. Sınıflama ve Regresyon Modelleri

Sınıflama ve regresyon, veri madenciliği tekniklerinde en çok kullanılan yöntemlerden biridir. Mevcut verilerden hareket ederek gelecekteki durumlar ile ilgili öngörü yapılması durumunda faydalanılır ve yeni bir veri elemanını daha önceden belirlenmiş sınıflara atamayı amaçlar. Sınıflama ve regresyon arasındaki temel fark, öngörü yapılan bağımlı değişkenin kategorik veya süreklilik gösteren bir değere sahip olmasıdır. Ancak her iki model de birbirine giderek yaklaşmakta ve bunun sonucu olarak aynı tekniklerden yararlanılması mümkün olmaktadır. Sınıflama ve regresyon modellerinde kullanılan başlıca teknikler şunlardır;

- Karar Ağaçları
- Yapay Sinir Ağları
- Genetik Algoritmalar
- K-En Yakın Komşu
- Bellek Temelli Nedenleme
- Lojistik Regresyon

Karar ağaçları tekniği, çok güçlü bir sınıflandırma ve öngörü aracıdır. Denetimli öğrenmenin kullanıldığı veri madenciliği tekniklerinden biridir. Diğer veri madenciliği tekniklerine nazaran çok daha anlaşılır bir dile sahiptir. Örneğin; kredi kartı başvurusunda bulunan bir müşteri için başvurusunun reddedilmesinin sebebinin *gelir<1 milyar* ve *kredi kartı sayısı<3* olması yeteri kadar açıklayıcı olacaktır. Ayrıca, modelin başarısı kadar, başarılı ve başarısız bir modelin nasıl çalıştığını araştırması da bu tekniği diğer tekniklere göre farklı kılmaktadır.

YSA; tanınması istenen nesnenin öznitelik vektörünü giriş olarak alan ve çıkış ünitelerinin birinde bu nesnenin sınıfını belirleyen bir cevap üreten, pek çok doğrusal olmayan hesaplama elemanlarının paralel işleyişinden meydana gelmiş tümleşik bir yapıdır. YSA'da her çıkış ünitesi gözlenen olayın farklı bir sınıfını belirler. YSA'nın paralel yapıları, bilgisayarları geleneksel yöntemlerden çok daha farklı kullanarak özellikle seri bilgisayarlarda bilinen yöntemlerle yapılması mümkün olmayan ve çok zor olan bir takım işlevleri (ses ve örüntü tanıma gibi) rahatlıkla yapmaları, YSA'yı üstün kılmıştır [87].

Genetik Algoritmalar (GA); en iyinin korunumu ve doğal seçilim ilkesinin benzetim yoluyla bilgisayarlara uygulanması ile elde edilebilir bir arama yöntemidir. Standart bir GA'da, aday sonuçlar eşit boyutlu vektörler olarak ifade edilir. Başlangıçta, bu vektörlerden bir grup, rastlantısal olarak seçilerek belirli bir büyüklükte bir popülasyon (toplum) oluşturulur. Kromozom adı verilen bu vektörler, yeni nesiller (nesil) oluşturarak değişikliklere uğrar. Bir kromozomun üzerindeki genler, n boyutlu vektörlerin bir boyutuna karşılık gelmektedir. Her

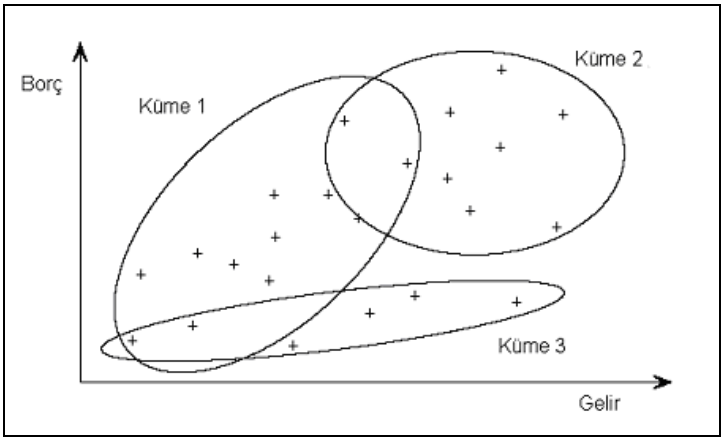
yeni nesilde kromozomların iyiliği ölçülür, yani her vektör (kromozom), amaç fonksiyonuna yerleştirilerek vermiş olduğu sonuç hesaplanır. Bir sonraki nesil oluşturulurken, bazı kromozomlar yeniden üretilir, çaprazlanır ve mutasyona uğratılır.

En yakın komşu yöntemi; nesne tanımanın en klasik yöntemlerinden birisi olup, tanımlanması istenen nesnenin vektörünü, veri tabanındaki en yakın komşusunun sınıfına dâhil ederek tanımlar. Yöntem, örnek vektörün istatistiksel dağılımından bağımsız olup, yalnızca en yakın komşunun sınıfına göre bir sınıflandırma yaparak tanıma işlemini gerçekleştirir.

Bellek tabanlı yöntemler; denetimli öğrenmenin kullanıldığı veri madenciliği tekniklerindendir. Bu tekniğin temel özelliği, daha önceki deneyimlerimizden faydalanarak elimizdeki problemlere benzer durumları tanımlayıp geçmiş benzer problemlere getirdiğimiz uygun çözümleri mevcut problemlerimize uygulamaya çalışmaktır.

2.4.2. Kümeleme

Kümeleme, veri tabanındaki verileri alt kümelere ayıran bir yöntemdir. Her bir kümede yer alan elemanlar birbirlerine çok benzemekte, özellikleri farklı olan elemanlar ise farklı kümelerde bulunmaktadır. Başlangıç aşamasında veri tabanındaki kayıtların hangi kümelere ayrılacağı veya kümelerin hangi özelliklere sahip olacağı bilinmemekte ve konunun uzmanı olan bir kişi tarafından bu özellikler belirlenmektedir.



Şekil 2.4 Kümeleme

Kümeleme analizinde amaç birbirine en çok benzeyen nesneleri aynı grupta toplamaktır. Benzemekten kasıt, geometrik anlamda uzaklık olarak birbirine en yakın nesnelerin seçilmesidir. Bu nedenle nesnelerin sayısal değer olması gerekir. Bu noktada değişkenler dörde ayrılmaktadır. Bunlar; [77]

- Kategorik Değişkenler: Bu değişkenler arasında sadece birbirine benzeme söz konusudur. Sıralama mümkün değildir. (örneğin siyah>beyaz durumu söz konusu değildir.
- Sıralama Değişkenleri: $X>Y$ şeklinde bir sıralama yapmak mümkündür. Ama büyüklüğün ne kadar olduğu belli değildir. (X-Y bulunamaz)
- Aralık Ölçekli Değişkenler: İki nokta arasındaki uzaklık hesaplanır. Fakat bu tür değişkenlerde gerçek "0" değeri yoktur.
- Oran Ölçekli Değişkenler: Anlamlı "0" noktasının bulunduğu, her türlü dört işleme açık değişkenler topluluğudur. (örneğin 20 yaşındaki bir kişi 10 yaşındaki bir kişiden iki katı yaştadır denilebilir.)

Kategorik ve sıralama değişkenlerinin matematiksel hesaplamaların yapılabileceği sayısal değerlere dönüştürülmesi şarttır.

2.4.3. Birliktelik Kuralları ve Ardışık Zamanlı Örüntüler

Birliktelik kuralları, işlemlerden oluşan ve her bir işlemin de ürünlerin birlikteliğinden oluştuğu düşünülen bir veri tabanında, bütün ürün birliktelilerini tarayarak, sık tekrarlanan ürün birlikteliklerini veri tabanından ortaya çıkarmaktır [71]. Bir alışveriş sırasında veya birbirini izleyen alışverişlerde müşterinin hangi mal veya hizmetleri satın almaya eğilimli olduğunun belirlenmesi, müşteriye daha fazla ürünün satılmasını sağlama yollarından biridir. Satın alma eğilimlerinin tanımlanmasını sağlayan birliktelik kuralları ve ardışık zamanlı örüntüler, pazarlama amaçlı olarak pazar sepet analizi adı altında veri madenciliğinde yaygın olarak kullanılmaktadır.

Birliktelik kuralları eş zamanlı olarak gerçekleşen ilişkilerin tanımlanmasında kullanılır. Örneğin, bira satın alan müşterilerin %75 ihtimalle patates cipsi de almaları veya ekmek ve yağ alanların %90'ının süt de satın almaları birliktelik kuralları kapsamında tespit edilebilir.

Ardışık zamanlı örüntüler ise birbiri ile ilişkisi olan ancak birbirini izleyen dönemlerde gerçekleşen ilişkilerin tanımlanmasında kullanılır. *X* ameliyatı yapıldığında, 15 gün içerisinde %45 ihtimalle *Y* enfeksiyonu oluşması, çekiç alan bir müşterinin ilk üç ay içerisinde %15, bu dönemi izleyen süre içerisinde ise %10 ihtimalle çivi alması ardışık zamanlı örüntüler olarak tanımlanmaktadır.

3. BİRLİKTELİK KURALI VE ALGORİTMALARI

3.1. Birliktelik Kuralı Tanımı

Günümüzde, birçok alandaki veriler bilgisayarlarda ve veritabanları üzerinde saklanmaktadır. Bu verilerden, istenilen ve kayda değer bilgilere ulaşmak için kullanılan tekniklerden biri de birliktelik kurallarıdır. Birliktelik kuralı, birçok alanda geniş kullanım alanına sahiptir ve nesnelerin veya niteliklerin bir arada olma durumlarını belirlemede kullanılmaktadır. Birliktelik kuralı bulma işlemi, yoğun nesne kümesi hesaplamaya dayalı bir işlem olup büyük veritabanları üzerinde uygulanması oldukça pahalı bir işlemdir. Bu nedenle daha önceden tespit edilen birliktelik kurallarının korunması da oldukça önemli bir konu olmaktadır [88].

Son zamanlarda, otomatik tanıma ve veri toplama uygulamalarındaki gelişmeler sayesinde firmaların satış noktalarında barkot sistemleri kullanımı yaygınlaşmaya başlamıştır. Bu gelişmeler ile beraber bir harekete ait satış verilerinin elektronik ortamlara aktarılmasına olanak sağlanmıştır. Genellikle büyük süpermarketlerde oluşan bu tür verilere market sepet verisi adı verilmektedir. Birçok kuruluş, market sepet verilerini kullanarak bu verilerden büyük faydalar sağlamayı amaçlamaktadır.

Market sepet verisi üzerinde birliktelik kuralı problemi, ilk olarak 1993 yılında ele alınmıştır [2]. Sepet analizinde amaç, nitelikler (ürün satışları) arasındaki ilişkiyi bulmaktır. Bu ilişkilerin bilinmesi şirketin kârını arttırmak için kullanılabilir. Eğer *X* malını alan müşterilerin *Y* malını da çok yüksek bir olasılıkla aldıkları biliniyorsa veya bir müşteri *X* malını alıyor ama *Y* malını almıyorsa o potansiyel bir *Y* müşterisidir. Sepet analizi günlük işlemler sonucu elde edilen verilerden anlamlı bağıntılar çıkarmada kullanılır. "Eğer *A* malını alıyorlarsa *%x* ihtimalle *B* malını almaya da meyillidirler" şeklinde bir sonuç *A* malını satan bir mağaza için çok faydalı bir bilgi olabilmektedir. Sepet analizi uygulamaları; çapraz satış (cross-selling), mağaza raflarının düzenlenmesi (layout), katalog tasarımı ve fiyatlandırma (pricing) gibi alanlarda kullanılmaktadır.

Birliktelik kuralı sorgusunun matematiksel modeli Agrawal ve ark. [2] tarafından tanımlanmıştır. Bu modelde $I = \{i_1, i_2, .., i_m\}$ ürün kodlarını; *D*, tüm hareketleri; *T*, bir hareketteki ürün kodlarını; ($T \subseteq I$), *tid* her harekete ait birincil anahtar numarayı, *k*-nesne kümesi, *k* adet ürünü içeren kümeyi temsil etmektedir. *X* bir ürün kümesi olmak üzere *T* hareketi *X* ürün kümesini ancak ve ancak $X \subseteq T$ şartını sağlıyorsa içerir. *X* ve *Y*, *I* ürün kümesinin bir alt kümesi ve $X \cap Y = \varnothing$ olsun. O zaman, bir birliktelik kuralı $X \rightarrow Y$ biçimindeki bir bağımlılık ifadesi ile

gösterilebilir. Bu ifade ile *X*, *Y*'yi belirler (*X* ürününü içeren hareketler kümesi *Y*'yi içeren hareketler kümesi içinde kapsanır) veya *Y*, *X*'e bağımlıdır denir (*Y* kümesinin varlığı *X* kümesinin var olmasına bağlıdır). Hareket numaraları öbeklendirilerek (veya gruplandırılarak) elde edilen ürünler arasındaki bağımlılık ilişkisinin yüzde yüz doğru olması beklenemez. Benzer şekilde, çıkarsama yapılan kuralın eldeki hareketler kümesinin önemli bir kısmı tarafından desteklenmesi istenir. Bu nedenlerden dolayı, $X \rightarrow Y$ eşleştirme kuralı kullanıcı tarafından minimum değeri belirlenmiş güvenilirlik (*c:confidence*) ve destek (*s:support*) eşik değerlerini sağlayacak biçimde üretilir. $X \rightarrow Y$ eşleştirme kuralına, *c* güvenilirlik ölçütü ve *s* destek ölçütü iliştirilir ve biçimsel olarak $\Phi(D)=<X \rightarrow Y,c,s>$ ile gösterilir. Burada *D*, örneklemi; $X \rightarrow Y$ ifadesi eşleştirme kuralını; *c* eşik değeri, ilgili kuralın minimum güvenirliğini; *s* eşik değeri ise ilgili kuralın minimum destek değerini göstermektedir. Sepet analizinde ürünler arasındaki bağıntı, destek ve güven kriterleri ile hesaplanmaktadır. Destek ve güven kriterlerinin tanımları aşağıdaki gibidir.

$$\text{Destek} : \quad P(X \text{ ve } Y) = \frac{X \text{ ve } Y \text{ mallarını satın alan müşterilerin sayısı}}{\text{Toplam müşteri sayısı}} \tag{3.1}$$

$$\text{Güven} : \quad P(X/Y) = \frac{P(X \text{ ve } Y)\ \{X \text{ ve } Y \text{ mallarını satın alanların sayısı}\}}{P(Y)\ \{Y \text{ malını satın alanların Sayısı}\}} \tag{3.2}$$

Destek kriteri, veri içerisinde ürünler arasındaki bağıntının ne kadar sık olduğunu, güven kriteri ise *Y* ürününü almış olan bir kişinin hangi olasılıkla *X* ürününü alacağını gösterir. İki ürünün satın alınmasındaki bağıntının önemli olması için hem destek, hem de güven kriterinin olabildiğince yüksek olması gerekmektedir.

Tablo 3.1 İşlemler, satın alınan ürünler, destek değerleri ve güven değerleri

a)

İşlemler	Satın Alınan Ürünler
1	Süt, Ekmek, Yumurta
2	Süt, Yumurta
3	Süt, Şeker
4	Ekmek, Yağ

b)

Tekrarlanan Ürün	Destek Değeri
Süt	% 75
Ekmek	% 50
Yumurta	% 50
Şeker	% 25
Yağ	% 25
Süt, Yumurta	% 50

c)

Tekrarlanan Ürün	Destek Değeri	Güven Değeri
Süt → Yumurta	% 50	% 66
Yumurta → Süt	% 50	% 100

Tablo 3.1-a'da; bir markette farklı zamanlarda yapılan 4 farklı işlem görülmektedir. Tablo 3.1-b'e ise her ürünün ve bazı ürün birlikteliklerinin tekrar edilme yüzdeleri verilmiştir. Bu tekrar edilme yüzdeleri, bize ürünlerin destek değerini vermektedir. Şeker ve yağ ürünlerinin destek değerlerinin referans destek değerinden düşük olmaları, bu ürünleri içeren ürün birlikteliklerinin ihmal edilebileceği anlamına gelmektedir. Tablo 3.1.c'de ise destek değeri referans destek değeri olarak kabul edilen %50 olan ve en az iki üründen oluşan ürün birliktelikleri ele alınarak bunların güven değerleri sonucu görülmektedir. Burada dikkat edilirse, *Süt → Yumurta* ilişkisi %66 güven değerine sahip iken *Yumurta → Süt* ilişkisi %100 güven değerine sahiptir. Çünkü yumurta satın alınma durumu, süt satın alınma durumlarının sadece %66'sında gerçekleşirken süt satın alınması ise yumurta alınma durumlarının tümünde gerçekleşmiştir. Bu nedenle *Yumurta → Süt* birlikteliğinde %100 güven değeri elde edilmektedir. Bu örnekten de anlaşıldığı gibi veri tabanı üzerindeki bir ürün gurubunun tekrar edilme miktarı destek değerini, bir ürün grubunu oluşturan herhangi bir ürünün veri tabanındaki işlem sayısı ile destek değerine sahip olan ürün birlikteliği içerisindeki yer alış sayısına oranı da güven değerini vermektedir.

Sepet analizi yapılırken bazı problemlerle karşılaşmak mümkündür. Bunlardan en önemlisi, bu tekniğin analiz sonunda bulunan bağıntının raslantısal olup olmadığını anlayamamasıdır. Bu duruma bir örnek verecek olursak, bir markette tuz alan müşterilerin %40'ının şeker de almakta olduğunu düşünelim. Bu bilgi ilk bakışta ilginç görünse de çok anlamlı olmayabilir. Çünkü marketten alışveriş yapan insanların belki de %40'ı zaten şeker alıyordur, dolayısı ile tuz ve şeker arasındaki bağıntı sadece raslantısaldır. Alışveriş yapanların sadece %15'i şeker alırken, tuz alanların % 40'ının şeker almasının bilgisi analizin sonucunu çok daha farklı etkileyecektir. Bu duruma ürünlerin birbirini çekmesi denilmektedir. Başka bir durumda şeker alanlar tüm alışveriş verisinin %65'ini oluştururken tuz alanların sadece %40'ının şeker almasının bilgisi bu ürünlerin birbirini ittiğini gösterir. Karşılaşılan problemlerin çoğu sepet analizi tekniğinin bağıntılar arasındaki raslantısallığının derecesini anlamıyor olmasından kaynaklanmaktadır. Özetle, sepet analizi tekniğinin güçlü ve zayıf yönlerini sıralamak gerekirse;

Güçlü yönleri;

- Denetimsiz veri madenciliği yöntemini başarı ile uygular
- Anlaşılabilir sonuçlar verir

Zayıf yönleri;

- İleri derecede bilgisayar performansına bağımlıdır
- Arasında bağıntı oluşturacak ürünlerin doğru olarak seçilmesi, analiz süreci için oldukça önemlidir.

Birliktelik kuralı çıkarmada kullanılan bazı algoritmalar mevcuttur. Minimum güvenirlik ve destek değerlerini sağlayan birliktelik kuralı çıkarım problemi iki adıma ayrılmıştır [2, 3].

Birinci adım, kullanıcı tarafından belirlenmiş minimum destek kıstasını sağlayan ürün kümelerinin bulunmasıdır. Bu kümelere sık geçen nesne kümesi veya yoğun nesne kümesi adı verilmektedir. Verilen örneklemde N adet ürün (nesne) var ise, potansiyel olarak 2^N adet yoğun nesne kümesi olabilir. Bu adımda üstel arama uzayını etkili biçimde tarayarak yoğun nesne kümelerini bulan etkili yöntemler kullanılmalıdır.

İkinci adım ise yoğun nesne kümeleri kullanılarak minimum güvenlik kıstasını sağlayan birliktelik kurallarının bulunmasıdır. Bu adımdaki işlem oldukça düzdür ve şöyle yapılmaktadır. Yoğun olan her l nesne kümesi için, boş olmayan l'nin tüm alt kümeleri üretilir. l'in boş olmayan alt kümeleri a ile gösterilsin. Her a kümesi için $a \Rightarrow (l - a)$ gerektirmesi, l kümesinin destek ölçütünün a kümesinin destek ölçütüne oranı minimum güvenirlik eşiği ölçütünü sağlıyorsa $a \Rightarrow (l - a)$ birliktelik kuralı olarak üretilir. Böylece minimum destek eşiğine göre

üretilen çözüm uzayında, minimum güvenirlik eşiğine göre taranarak bulunan birliktelikler kullanıcının ilgilendiği ve potansiyel olarak önemli bilgi içeren birlikteliklerdir.

Birliktelik sorgusu algoritmalarının başarımını belirleyen adım birinci adımdır. Yoğun nesne kümeleri belirlendikten sonra, birliktelik kurallarının bulunması ise düz bir adımdır [89].

3.2. Apriori Algoritması

Yoğun nesne kümelerini bulmak için birçok kez veri tabanını taramak gerekir. İlk taramada bir elemanlı minimum destek değerini sağlayan yoğun nesne kümeleri bulunur. İzleyen taramalarda bir önceki taramada bulunan yoğun nesne kümeleri, aday kümeler adı verilen yeni potansiyel yoğun nesne kümelerini üretmek için kullanılır. Aday kümelerin destek değerleri tarama sırasında hesaplanır ve aday kümelerinden minimum destek değerini sağlayan kümeler o geçişte üretilen yoğun nesne kümeleri olur. Yoğun nesne kümeleri bir sonraki geçiş için aday küme olurlar. Bu süreç yeni bir yoğun nesne kümesi bulunmayana kadar devam eder.

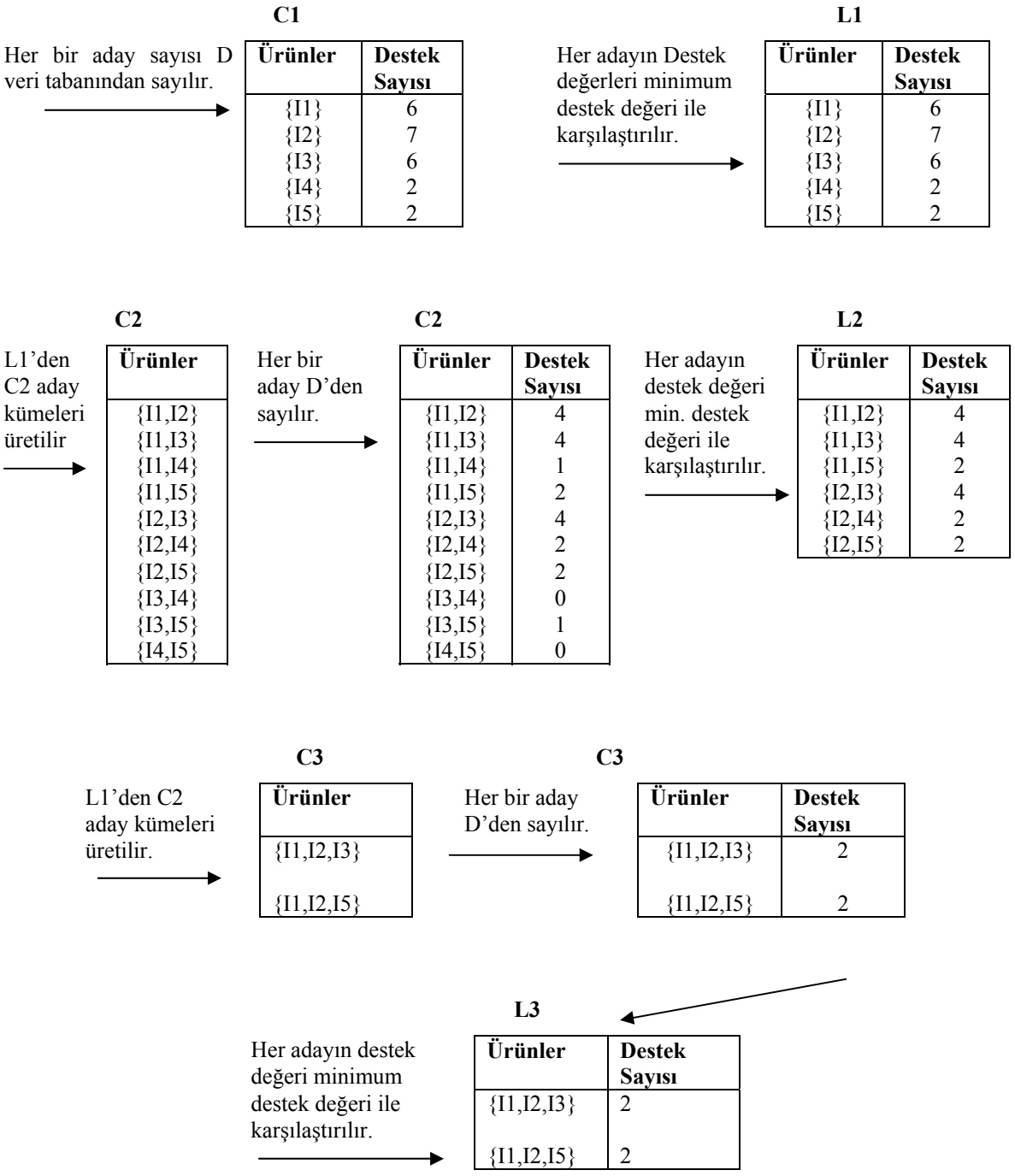
Bu algoritmada temel yaklaşım eğer k-nesne kümesi minimum destek değerini sağlıyorsa bu kümenin alt kümeleri de minimum destek değerini sağlar şeklindedir.

Apriori algoritmasında önce aday nesne kümeleri oluşturulur. Bu kümeler potansiyel olarak yoğun nesne kümeleridir. *C* ile gösterilir ve *C1*, *C2*, *C3*, …..*Ck* olarak k-nesne kümesini oluştururlar. Her *Ck* nesne kümesi *C(k-1)* nesne kümesini içerir ve $C1 < C2 < C3 \ldots . < Ck$ şeklinde sıralıdır. Yoğun olan k-nesne kümeleri ise *L* ile gösterilir ve minimum destek kıstaslarını sağlarlar.

Tablo 3.2 Bir marketten satın alınan ürünler listesi

İşlemler	**Satın Alınan Ürün Listesi**
T1	I1, I2, I5
T2	I2, I4
T3	I2, I3
T4	I1, I2, I4
T5	I1, I3
T6	I2, I3
T7	I1, I3
T8	I1, I2, I3, I5
T9	I1, I2, I3

Örnek : Tablo 3.2'de bir firmada satın alınmış ürünlerle ilgili bir veri tabanı görülmektedir. Bu veri tabanını *D* olarak adlandırılsın. Veri tabanında 9 adet işlem görüldüğüne göre $|D| = 9$ olmaktadır. Bu durumda *D* veri tabanı üzerinde Apriori algoritması kullanılarak yoğun nesnelerin nasıl bulunabileceği Şekil 3.1'de görülmektedir [90].



Şekil 3.1 Apriori algoritması ile aday nesne ve yoğun nesne kümelerini belirlenmesi. (Minimum destek değeri 2 alınmıştır.)

1) Algoritmanın ilk iterasyonunda her nesne, yani bir elemanlı nesneler kümesi C1 olarak elde edilir. Algoritma, her nesnenin sayısını bulmak için veri tabanını basitçe taramaktadır.

2) Bu veri tabanı için minimum destek sayısı 2 olarak belirlenmiştir. Minimum destek değeri 2/9 = %22'dir. Buradan bir elemanlı nesneler kümesi olan L1 elde edilebilir. C1 deki tüm nesneler minimum destek değerini sağlamakta ve L1 elde edilmektedir.
3) L2'yi bulamak için önce iki elemanlı aday nesne kümesini elde etmek gerekir ve L1xL1 den C2 elde edilir. L1'in ikili kombinasyonları C2'nin eleman sayısını vermektedir.
4) Daha sonraki işlem D'yi taramak ve C2 deki her aday nesnenin minimum destek değerini bulmaktır. Şekil 3.3'te ortadaki ikinci tablodan bu durum görülmektedir.
5) C2'de minimum destek değerine sahip olmayan aday nesnelerin çıkarılması ile L2 nesne kümesi elde edilir.
6) 3 elamanlı aday nesne kümesinin elde edilmesi daha sonraki bölümde detaylı olarak anlatılacaktır. Normalde C3=L2xL2 ={{I1,I2,I3}, {I1,I2,I5}, {I1,I3,I5}, {I2,I3,I4}, {I2,I3,I5}, {I2,I4,I5}}olarak elde edilir. Bunların hepsi sık tekrarlanıyor görülebilir ancak Apriori algoritması özelliğine göre son dört tanesi sık tekrarlanmıyor olabilir. Bu nedenle son dördünü C3'ten çıkarabiliriz. Böylece D'yi taramak ve L3'ü belirlemek için bazı işlemler boşu boşuna yapılmamış olacaktır.
7) C3 üzerinde minimum destek değerine sahip olmayan aday nesnelerin çıkarılması ile L3 nesne kümesi elde edilir.
8) Algoritmada L3xL3'ü kullanılarak 4 elemanlı aday nesne kümesi elde edilir. Bunun sonucunda {I1,I2,I3,I5} elde edilmesine rağmen bu kümenin alt kümesi olan {I2,I3,I5} L3'te bulunmamaktadır. Bu nedenle C4=0'dır ve tüm sık kullanılan birliktelikler elde edilerek algoritma sona ermektedir.

3.2.1. Apriori ile L2 Kullanılarak C3 Oluşturulması

Apriori algoritmasında L(k-1)'ler kullanılarak Ck elde edilmektedir. k>2 için Ck'ların elde edilmesi, k=3 için aşağıda detaylı olarak anlatılmıştır [90].

1) C3=L2xL2 ={{I1,I2}, {I1,I3}, {I1,I5}, {I2,I3}, {I2,I4}, {I2,I5}} X {{I1,I2}, {I1,I3}, {I1,I5}, {I2,I3}, {I2,I4}, {I2,I5}} = {{I1,I2,I3}, {I1,I2,I5}, {I1,I3,I5}, {I2,I3,I4}, {I2,I3,I5}, {I2,I4,I5}}.

2) Elde edilen aday nesne kümelerinden bazıları kullanılmayabilir. Çünkü bunlar Apriori'ye göre yoğun nesneler değildir. Bunu elde etmek için bir eleme işlemi gerçekleştirilir.

- {I1,I2,I3} kümesinin iki elemanlı alt kümeleri {I1,I2}, {I1,I3} ve {I2,I3} tür. Bu iki elamanlı alt kümelerin tümü L2'nin bir nesnesi olduğuna göre {I1,I2,I3} C3'ün bir aday nesnesi olabilir.

- {I1,I2,I5} kümesinin iki elemanlı alt kümeleri {I1,I2}, {I1,I5} ve {I2,I5} tir. Bu iki elamanlı alt kümelerin tümü L2'nin bir nesnesi olduğuna göre {I1,I2,I5} C3'ün bir aday nesnesi olabilir.
- {I1,I3,I5} kümesinin iki elemanlı alt kümeleri {I1,I3}, {I1,I5} ve {I3,I5} tir. Bu iki elamanlı alt kümelerden {I3,I5}, L2'nin bir nesnesi değildir ve bu nedenle sık tekrarlı değildir. Bundan dolayı {I1,I3,I5} C3'ün aday nesneliğinden çıkarılmalıdır.
- {I2,I3,I4} kümesinin ün iki elemanlı alt kümeleri {I2,I3}, {I2,I4} ve {I3,I4} tür. Bu iki elamanlı alt kümelerden {I3,I4}, L2'nin bir nesnesi değildir ve bu nedenle sık tekrarlı değildir. Bundan dolayı {I1,I3,I5} C3'ün aday nesneliğinden çıkarılmalıdır.
- {I2,I3,I5} kümesinin iki elemanlı alt kümeleri {I2,I3}, {I2,I5} ve {I3,I5} tir. Bu iki elamanlı alt kümelerden {I3,I5}, L2'nin bir nesnesi değildir ve bu nedenle sık tekrarlı değildir. Bundan dolayı {I1,I3,I5} C3'ün aday nesneliğinden çıkarılmalıdır.
- {I2,I4,I5} kümesinin iki elemanlı alt kümeleri {I2,I4}, {I2,I5} ve {I4,I5} tir. Bu iki elamanlı alt kümelerden {I4,I5}, L2'nin bir nesnesi değildir ve bu nedenle sık tekrarlı değildir. Bundan dolayı {I1,I3,I5} C3'ün aday nesneliğinden çıkarılmalıdır.

3) Sonuç olarak eleme sonucunda C3= {{I1,I2,I3}, {I1,I2,I5}} elde edilir.

3.2.2. Algoritma

Giriş: Database (*D*), minimum destek *min_sup*.

Çıkış: *L*, D de yoğun olan nesne kümeleri

```
(1)        L1=find_frequent_1-itemsets(D)
(2)        for (k=2;L(k-1)≠0;k++) {
(3)              Ck=apriori_gen(L(k-1), min_sup)
(4)              for each transaction t∈D{     //sayım için D yi tara
(5)                    Ct=subset(Ck,t);
(6)                    for each candidate c∈Ct
(7)                          c.count++;
(8)              }
(9)              Lk={ c∈Ck | c.count ≥min_sup }
(10) }
(11) return L=Uk Lk;
```

```
procedure apriori_gen (L(k-1);min_sup)
(1)       for each itemset l1∈L(k-1)
(2)          for each itemset l2∈L(k-1)
(3)             if (l1[1]= l2[1] )∧(l1[2]=l2[2]) ∧…∧ (l1[k-1]=l2[k-1]) then {
(4)                c=l1x l2;   // aday nesnelerin üretilmesi
(5)                if has_infrequent_subset(c,L(k-1)) then
(6)                   delete c; // gereksiz adayların silinmesi
(7)                else add c to Ck;
(8)             }
(9)       return Ck;

procedure has_infrequent_subset(c,L(k-1));
(1)       for each (k-1)-subset s of c
(2)          if s∉L(k-1) then
(3)             return TRUE
(4)       return FALSE
```

Yukarıda Apriori algoritması ve bu algoritmaya ait alt programlar görülmektedir. Algoritmanın 1. satırında 1-elemanlı aday nesneler yani *L1* bulunmaktadır. 2 ile 10. satırlar arasında ise Lk elde etmek amacı ile *L(k-1)* kullanılarak *Ck*'lar elde edilmektedir. Bu satırlarda bulunan **apriori_gen** alt programı (3. satır) ise aday nesneleri bulmakta ve bazı yoğun olmayan nesne kümelerini tespit ederek eleme yapmaktadır. 4. satırda ilk önce üretilen adaylar veri tabanından taranmakta ve 5. satırda ise aday nesnelerin tüm alt kümeleri bulunması için **subset** fonksiyonu kullanılmaktadır. 6 ve 7. satırlarda da bu adayların her birinin sayısı toplanmaktadır. Sonuç olarak, minimum destek değerine sahip tüm yoğun nesne kümeleri (*L*) elde edilmektedir.

Apriori_gen alt programı 2 çeşit görev yerine getirmektedir. Bunlardan birincisi aday nesne oluşturma ve ikincisi ise eleme işlemleridir. Aday oluşturmada *L(k-1)* kendisi ile birleştirilerek aday nesneler oluşturulur. Eleme işleminde ise Apriori özelliğine göre yoğun olmayan aday nesne kümelerinin elenmesi işlemi gerçekleştirilmektedir. Yoğun olmayan nesne kümelerinin test edilmesi işlemi de **has_infrequent_subset** alt programı ile gerçekleştirilmektedir.

3.2.3. Yoğun Nesne Kümelerinden Birliktelik Kurallarının Çıkarılması

Veri tabanı üzerinden yoğun nesne kümeleri bulunduktan sonra bu nesne kümeleri kullanılarak çok güçlü kurallar üretilebilmektedir. Bu kuralar minimum destek ve güven

değerlerinin her ikisini de sağlamaktadır. Bu kurallar aşağıdaki güven denklemleri kullanılarak elde edilebilir [90].

$$\text{Güven } (A \Rightarrow B) : P(B \backslash A) = \frac{\textit{Destek değeri (A U B)}}{\textit{Destek değeri (A)}} \qquad (3.3)$$

(A U B) Destek değeri, hem *A* hem de *B*'nin birlikte bulunduğu nesne sayısıdır. *(A) Destek Değeri*, sadece *A*'yı içeren nesne sayısıdır. Bu denklemden, aşağıdaki gibi birliktelik kuralları elde edilebilir.

- *l* nesnesini tara ve *l*'nin boş olmayan alt kümelerini bul.
- Her boş olmayan altküme nesnesi için elde edilen kural;

$$\text{“s} \Rightarrow (l\text{-s)” if } \frac{destek(l)}{destek(s)} \geq min\_guv \qquad (3.4)$$

Denklem 3.4'teki *min_guv* değeri, minimum güven değeridir. Kurallar, yoğun nesne kümelerinden üretildiği için her bir kural otomatik olarak minimum destek değerini sağlamaktadır.

Örnek: Kuralların çıkarılmasını Şekil 3.1'de gösterilen veri tabanı (*D*) üzerinde bir örnekle göstermek mümkündür. Bu veri tabanındaki yoğun nesne kümelerinden biri *l*= {I1,I2,I5} tir. Bu *l* yoğun nesne kümesi üzerinden hangi kuralların üretilebileceği şu şekilde hesaplanır: *l*'nin tüm boş olmayan alt kümeleri {I1,I2}, {I1,I5}, {I2,I5}, {I1}, {I2} ve {I5} tir. Buna göre çıkarılabilecek birliktelik kuralı sonuçları aşağıdaki listede verilmektedir.

(1) $I1 \wedge I2 \Rightarrow I5$, güven = 2 / 4 = % 50

(2) $I1 \wedge I5 \Rightarrow I2$, güven = 2 / 2 = % 100

(3) $I2 \wedge I5 \Rightarrow I1$, güven = 2 / 2 = % 100

(4) $I1 \Rightarrow I2 \wedge I5$, güven = 2 / 6 = % 33

(5) $I2 \Rightarrow I1 \wedge I5$, güven = 2 / 7 = % 29

(6) $I5 \Rightarrow I1 \wedge I2$, güven = 2 / 2 = % 100

Yukarıdaki listeye göre eğer minimum güven değeri %70 ise sadece 2. 3. ve 6. kurallar istenilen sonuçlardır.

3.3. Diğer Birliktelik Kuralı Algoritmalarının Analizi

3.3.1. AIS Algoritması

AIS (Agrawal Imielinski Swami) [2] algoritması, birliktelik kuralı alanında yapılmış ilk çalışmadır. Tekrarlamalı yapı kullanarak çalışır ve algoritmanın *k*'nıncı tekrarında, *k*-elemanlı

yoğun nesne kümeleri hesaplanmaktadır. Böylelikle, veri tabanı üzerinde maksimum uzunlukta nesne kümeleri elde edilebilmektedir. Ayrıca her tekrar sırasında, aday nesne kümeler de sayılmakta ve üretilmektedir. Ancak, üretilen bu aday nesne kümelerinin birçoğunun yoğun nesne kümesi olmaması ve üretilen birliktelik kurallarında eşitliğin sağ tarafında sadece bir nesneni bulunması bu algoritmanın en büyük kısıtlarıdır [3].

3.3.2. Apriori_TID Algoritması

Kaynak [3]'te önerilen bu algoritma Apriori algoritmasına benzer bir yapıya sahiptir. Apriori_TID, her tekrarda var olan nesne kümelerinin kodlayarak depolayan bir yapı kullanmaktadır. Bu nedenle, bir sonraki adımlar için gerekli olan nesne kümelerine ait destek değerleri veri tabanı kullanılarak elde edilmek yerine bir önceki aday kümelerine ait destek değerlerinden elde dilmektedir. Böylece, ilerleyen adımlarda nesne kümelerine ait destek değeri hesaplama işlemleri daha verimli hale dönüşmektedir. Ancak Apriori algoritması, geniş veri kümeleri üzerinde daha verimli çalışmaktadır [3].

3.3.3. Apriori_Hybrid Algoritması

Kaynak [4]'te önerilen bu algoritma Apriori ve Apriori_TID algoritmalarının ikisini beraber kullanan melez bir algoritmadır. Algoritma çalışması esnasında her tekrarda aynı algoritma kullanılmak yerine Apriori algoritmasının avantajlı olduğu durumlarda Apriori algoritmasını, Apriori_TID algoritmasının avantajlı olduğu durumlarda da ise Apriori_TID algoritma yapısını kullanarak çalışan bir yapıya sahiptir. Apriori_Hybrid algoritması birçok durumda Apriori algoritmasından daha iyi sonuçlar vermektedir [21].

3.3.4. Çevrimdışı Aday Nesne Kümesi Belirleme Algoritması

Çevrimdışı Aday Nesne Kümesi Belirleme (ÇANKB) algoritması [5], Apriori algoritmasına oldukça benzerlik göstermektedir. Apriori' den, sadece aday nesne kümesi üretme bakımından ayrılmaktadır. Her iki algoritma da *L(k-1)* nesne kümesinden yeni adaylar kümeler üretmektedir, ancak ÇANKB *k-1* elemana sahip iki büyük nesne kümesinin *k-2* nesnesi aynıysa bu iki *k-1* elemanlı nesne kümesini kullanarak yeni bir aday üretmektedir [91]. Yani, aday nesne kümesi üretimi esnasında, mümkün olduğunca önceki veri tabanı tarama işlemlerinde elde edilen bilgilerden faydalanmaya çalışmaktadır.

3.3.5. SETM Algoritması

SETM (Set Oriented Mining) [6, 7] algoritmasında, birliktelik kurallarını aramak için SQL komutlarını kullanılmaktadır. Aday nesne kümesi üretme işleminde, bilinen SQL birleştirme işlemi kullanmaktadır. Üretilen her aday ve nesne kümesine *TID* tekil anahtar verme zorunluluğundan dolayı ekstradan bellek alanı ihtiyacı doğurması algoritmanın dezavantajlarındandır [92]. SETM algoritması, Apriori' ye göre çok daha fazla aday üretmektedir ve daha az etkilidir.

3.3.6. DHP Algoritması

DHP (Dynamic Hashing and Pruning) [8] algoritmasının amacı, 2 elemanlı aday nesne kümelerinin sayısını azaltmaya dayalıdır. Çünkü bu aşamada yapılacak azaltma, birliktelik kuralının ilerleyen tekrarlarında büyük avantajlar sağlamaktadır. İlk önce büyük 1-elemanlı yoğun nesne kümeleri bulunur ve daha sonra 2 elemanlı aday nesne kümeleri üretmek için bir hash tablosu oluşturulmaktadır. Daha sonraki tüm tekrarlarda hash tablosundaki bilgiler kullanılarak L(k-1) kümelerinden yeni aday kümeler üretilmektedir. Bu mantıkla çalışan DHP algoritması, özellikle ikinci tekrarda olmak üzere tüm tekrarlarda daha iyi performans vermektedir [91].

3.3.7. Partition Algoritması

Partition [9] algoritması, bilinen diğer algoritmalardan biraz farklı olup veri tabanı üzerindeki en iyi tarama özelliğine sahip algoritmadır. Veri tabanını üzerinde en fazla iki tarama işlemi gerçekleştirerek veri tabanını n adet parçaya ayırmakta ve her parça için yoğun nesne kümelerini hesaplamaktadır. Algoritma, daha sonra tüm parçaların yoğun nesne kümelerini birleştirilmekte ve süper bir yoğun nesne kümesi hesaplamaktadır. Ancak tüm veri tabanında yoğun olan nesne kümelerinin, her parça setinde de yoğun olması gerekmektedir. Partition algoritmasının temel avantajı Giriş/Çıkış maliyetini azaltmak ve yoğun kümeleri sayarken ana hafızayı kullanmaktır [91].

3.3.8. MONET Algoritması

MONET [10] algoritması, sadece genel amaçlı veritabanları üzerinde birleştirme ve kesiştirme işlemlerini kullanarak birliktelik kuralı üretmektedir. Veri tabanı üzerindeki nesneler

kolonlar halinde veri tabanına yüklenir ve her birine bir tanımlama numarası verilerek numaralandırılır. Aday nesne kümeleri Apriori algoritmasındaki yapıya benzer bir şekilde elde edilir. Ancak, nesne kümelerinin destek değerlerinin bulunması için tüm veri tabanı taranmaz. Bunun yerine küme kesiştirme işlemleri kullanarak destek değerleri elde edilir. Bu metot aslında partition algoritmasında kullanılan yapıya benzemektedir. Bu algoritmanın performansı tamamen birleştirme ve kesiştirme işlemlerinin etkinliğine bağlıdır.

3.3.9. Sampling Algoritması

Kaynak [11]'de birliktelik kuralı üretmek için Sampling algoritması önerilmiştir. Algoritma, büyük veri tabanı üzerinden aralıklı rasgele örnekler seçerek daha düşük bir destek değerine göre yoğun nesne kümelerini hesaplamaktadır. Bu aşamada seçilen destek değerinin daha küçük olmasının sebebi, olası muhtemel bir yoğun nesne kümesini atlamamaktır. Sonra elde edilen bu yoğun nesne kümeleri, veri tabanı üzerinde test edilir. Elde edilen bu yoğun nesne kümelerinde herhangi bir olumsuzluk yok ise veri tabanının toplam bir kez taranması ile yoğun nesne kümeleri elde edilmiş olur. Aksi durumda ise ikinci bir veri tabanı taraması gerekmekte ve aday nesne kümeleri Apriori de olduğu şekilde taranmakta ve sayılmaktadır. Algoritmanın en avantajlı yönü Giriş/Çıkış maliyetini azaltmasıdır. Kaynak [93]'te birliktelik kurallarının bulunmasında örneklerin etkisini analiz edilmekte ve örnek ölçüsünü seçmek için etkili ve yararlı stratejiler önerilmektedir.

3.3.10. DIC Algoritması

DIC (Dynamic Itemset Counting) [12] algoritması, veri tabanı üzerindeki tarama sayısını azaltmaya yönelik bir algoritmadır. Herhangi bir *k*-elemanlı nesne kümesinin yoğun olabileceğini fark edilir edilmez *k*'nıncı tekrar beklenmeden destek değeri hesaplanmaktadır. Böylelikle veri tabanı tarama sayısı, maksimum nesne kümesi boyutundan genellikle daha küçük olmaktadır.

3.3.11. Eclat, MaxEclat, Clique, MaxClique Algoritmaları

Kaynak [13]'te önerilen bu dört algoritma veri tabanı üzerinde sadece bir kez geçiş yapmaktadır. Potansiyel olarak var olan yoğun nesne kümelerini bulmak için nesne kümesi kümeleme şemalarından birini kullanmaktadır. Her bir küme bir alt bölme oluşturmakta ve bu alt bölme, gerekirse yukarıdan aşağıya gerekirse de hem yukarıdan aşağıya hem de aşağıdan

yukarıya taranarak sırasıyla önce aday nesne kümeleri sonra da yoğun nesne kümeleri elde edilmektedir. Bu algoritmalarda da yine Partition algoritmasında olduğu gibi *tidlist* yapısı kullanılmakta ve aday nesne kümelerine ait destek değerleri basit bir kesiştirme işlemi ile elde edilmektedir. Ayrıca bu algoritmalar çok fazla hafıza ihtiyacı gerektirmemektedirler.

3.3.12. Max-Miner Algoritması

Kaynak [14]'te önerilen bu algoritma, mümkün olduğunca hızlı bir şekilde yoğun nesne kümelerini elde etmek ve alt kümeleri temizlemek üzere geliştirilmiştir. Yoğun nesne kümelerini ve boyutunu doğrusal olarak hesaplamaktadır. Aday nesne kümelerini üretme ve sayma işlemi Apriori'ye benzemekte ve *N* elemanlı yoğun nesne kümesi elde etmek için en fazla *N* defa veri tabanı taranmaktadır. Max-Miner özellikle yoğun nesne kümesinin boyutunun arttığı zamanlarda yüksek performans göstermektedir.

3.3.13. Carma Algoritması

Carma [15] algoritması, birliktelik kurallarının çevrimiçi olarak hesaplanması için önerilen bir algoritma olup veri tabanı üzerinden tam iki kez tarama gerçekleştirmektedir. Veri tabanının ilk taranması esnasında potansiyel yoğun nesne kümeleri ile ilgili bir bölüm oluşturulur ve istenirse destek değeri bu aşamada değiştirilebilmektedir. Veri tabanının ikinci taranmasında ise algoritma her bölümdeki nesne kümesinin destek değerini hesaplamakta ve yoğun olmayan nesne kümelerini temizlemektedir. Bölümler oluşturulurken her nesne kümesi ile ilgili alt ve üst sınır değerlerine göre yeni adaylar bu bölümlere eklenir veya çıkarılır. Ayrıca, sayma işlemi ikinci taramada yer almaktadır.

Tablo 3.3 Birliktelik kuralı algoritmaları ve özellikleri

Algoritma	Veri Tabanını Tarama Sayısı	Algoritmanın Özelliği
AIS	N	Adaylar, veri tabanı taranarak elde edilir. Küçük ölçekli veritabanlarında uygundur.
Apriori	N	Adaylar, *L(k-1)* ile *L(k-1)'*in birleştirilmesiyle elde edilir ve veri tabanı taranarak sayılır. Orta ölçekli veritabanlarında uygundur. AIS in dezavantajlarından arındırılmıştır.
Apriori_TID	N	Adaylar, *L(k-1)* ile *L(k-1)'*in birleştirilmesinden elde edilir. *Ck* çok ise yavaş aksi halde Apriori'den daha performanslıdır.
Apriori_Hybrid	N	Apriori ve Apriori_TID'e göre daha iyi çalışır. Her iki algoritmayı da kullanan melez bir yapıdır.
OCD	N	Adaylar, veri tabanı taranarak elde edilir. Büyük veritabanlarında ve küçük destek değerlerinde uygulanabilir.
SETM	N	SQL komutları uyumluluğu mevcuttur.
DHP	N	Adaylar, *L(k-1)* ile *L(k-1)'*in birleştirilmesi ile elde edilir ve veri tabanı taranarak sayılır. Hash tablosu kullanır.
Partition	2	Geniş veritabanlarında uygulanabilir. Homojen veriler üzerinde etkilidir.
MONET	N	Adaylar, *L(k-1*) ile *L(k-1)'*in birleştirilmesiyle elde edilir ve kolonların kesiştirilmesi ile sayılır.
Sampling	≤ 2	Adaylar, *L(k-1)* ile *L(k-1)'*in birleştirilmesi ile elde edilir ve veri tabanı taranarak sayılır. Geniş veritabanlarında ve küçük destek değerlerinde uygulanabilir.
DIC	$\leq$ N (genelde 2)	Adaylar, veri tabanı taranarak sayılır. Yoğun olabilecek tüm nesne kümeleri kontrol edilir.
MaxClique	1	Olası yoğun nesne kümeleri incelenir. *tidlist* liste yapısı kullanır ve kesiştirme işlemi ile adaylar sayılır.
Max-Miner	N	Adaylar, *L(k-1)* ile *L(k-1)'*in birleştirilmesiyle elde edilir ve veri tabanı taranarak sayılır.
Carma	2	Veri tabanı verileri ağ üzerinden elde edildiğinde uygulanabilir. Destek ve güven değerleri çevrimiçi olarak değiştirilebilmektedir.

3.4. Birliktelik Kuralı Türleri

Bilindiği gibi birliktelik kuralları algoritmaları, veri tabanını tarayarak nitelikler arasındaki ilişkileri ortaya çıkarmakta ve bir niteliğin varlığına veya yokluğuna bağlı olarak ilişkiler ve kurallar üretmektedir. Diğer özellikler olan miktar, ağırlık ve hiyerarşik bilgi düzeni gibi özellikler göz önüne alınmamaktadır. Bu bölümde birliktelik kurallarının nesne kümesi üretimine bağlı olarak çeşitleri hakkında kısaca bilgi verilecektir.

3.4.1. Hiyerarşik Birliktelik Kuralları

Birçok durumda veri tabanı nitelikleri belirli bir hiyerarşi yapısında olabilmektedir. Örneğin bir market veri tabanı için, içecekler→ alkolsüz içecekler→ meyve suları→ portakal suyu gibi bir hiyerarşi söz konusu olabilir. Bu tür durumlarda Genelleştirilmiş (çok seviyeli) birliktelik kuralları [16], aynı seviyede olan nitelikler arasındaki kurallarda olduğu gibi hiyerarşinin farklı seviyelerindeki birliktelik kurallarını bulmayı amaçlamaktadır. Genelleştirilmiş bir birliktelik kuralının örneği "*alkolsüz içecekler → meyve*" şeklindedir. Orijinal veri tabanında olmayan hiyerarşi seviyeleri için yeni bir kolon üretilmesi bu durumlar için yetersiz bir çözüm olabilmektedir. [16, 17]'de Hiyerarşik bilgiyi algoritmayla birleştiren yeterli ve etkili algoritmalar önerilmektedir. Nesne merkezli bir metot [18]'de ve [19]'da çok seviyeli birliktelik kurallarını bulmak için SQL sorguları kullanılmıştır. Son olarak [20]'de esnek çok seviyeli birliktelik kuralları tartışılmaktadır.

3.4.2. Sınırlandırılmış Birliktelik Kuralları

Gerçek hayatta, kullanıcılar genellikle veri tabanından çıkarılmış birliktelik kurallarının küçük bir kısmı ile ilgilenmektedirler. Örneğin bir kullanıcı sadece bazı nitelikler arasındaki birlikteliklerin varlığı veya yokluğu üzerine yoğunlaşmış olabilmektedir. Bu konuda birçok algoritma olmakla beraber [22]'de sınırlandırılmış birliktelik kuralı için bir algoritma önerilmektedir. Algoritmada kullanılan destek ve güven değerleri de elde edilen birliktelik kurallarını sınırlandırmaktadır. Ayrıca, son zamanlarda elde edilen kurallar için ilginçlik ölçütü de kural sınırlandırmada kullanılmaktadır [21, 23-25]. Bir niteliğin değeri beklenenden oldukça yüksek bir değere sahip olduğunda bu nitelik ile elde edilen kurallardan birçoğu önemsiz olabilmektedir. Bir diğer ölçüt [26]'da önerilen ve ortalama karesel hataya dayalı metottur. [27]'de de ki-kare korelasyon testi birliktelik kurallarında ölçüt olarak kullanılmıştır.

3.4.3. Nicel Birliktelik Kuralları

Orijinal birliktelik kuralı problemi, niteliklerin var/yok (*boolean*) değeri ile ilgilenmektedir. Ancak niteliklerin aldığı değerler birçok problemde ikiden fazla olabilmektedir. Bu durumlarda, niteliklerin sayısal veya kategorik değerleri ile kural üretmek için [28]'de etkili bir algoritma önerilmektedir. Örneğin, Yaş:30..39 ve Evli: Evet → Arabaların Sayısı: 2 (%40, 90%) şeklinde verilen bir kural nicel birliktelik kuralı olarak değerlendirilmektedir. Bu kural, yaşı 30 ile 39 arasında olan ve evli olan insanların %90'nın iki araba sahibi olduğu anlamına gelmekte ve veri tabanı üzerindeki destek değeri %40'tır.

Nicel birliktelik kurallarının hesaplanmasındaki önemli nokta, nicel verilerin değerlerinin nasıl bir şekilde bölümlere ayrılacağıdır. Fukuda ve arkadaşları [29] optimize edilmiş birliktelik kurallarını geliştirerek destek veya güven değerini maksimuma çıkaracak şekilde nümerik verilen değerlerin bölümlerini bulmaya çalışmışlardır. Nümerik veya kategorik veriler için aynı kavram [30]'da da araştırılmıştır. Ayrıca Wang ve arkadaşları [31] bir kuralın ilginçliğini maksimuma çıkarmak için farklı aralıkları kullanarak ilginçliğe dayalı bir yöntem önermiştir. Bir başka çalışmada, nümerik değerler için bulanık birliktelik kuralları önerilmektedir [32]. Bulanık birliktelik kuralları, $X = A \rightarrow Y = B$ formatında olup buradaki X ve Y nesne kümelerini, A ve B ise sırasıyla X ve Y'yi tanımlayan bulanık değerleri göstermektedir.

3.4.4. Sıralı Örüntüler

Birliktelik kurallarının amacı, belli bir zamanda meydana gelen oluşumları keşfetmektir. Ancak, verilerin uzun bir zaman süreci ile depolanması ile beraber sıralı örüntülerin keşfi önemli bir konu haline gelmiştir. Sıralı örüntüler için bir müşterinin "Yıldız Savaşları" filmini kiralaması, daha sonra "İmparator Geri Dönüyor" filmi ve daha sonra da "Jedi'nin Dönüşü" adlı filmleri kiralaması örnek olarak verilebilir [33]. Sıralı örüntülerde nesnelerin sıralı olması gerekmez ancak oluşumları sıralı olmalıdır. Kaynak [33]'de sıralı örüntü keşfi için üç ayrı algoritma önerilmektedir. Sıralı örüntü keşfi, kaynak [34]'te ise hiyerarşik bilgi ve kayan pencereler kullanılarak geliştirilmiştir. Ayrıca kaynak [35, 36]'da ise sıralı örüntü keşfi için etkili algoritmalar önerilmektedir.

3.4.5. Periyodik Kurallar

Birliktelik kuralları, sıralı yapıda olan bir veri üzerindeki periyodik özellikleri ortaya çıkarabilmektedir. Elde edilen bazı kurallar, belirli bir zaman periyodu için yeterli destek değerine sahip olabilmektedir. Ancak bu kurallar tüm veri tabanı ele alındığında bazen minimum destek değerini sağlamayabilir. Özden ve arkadaşları tarafından [94] belirli bir zaman periyodunda minimum destek ve güven değerini sağlayan kurallar için Periyodik Birliktelik Kuralları kavramı öne sürülmüştür. Örneğin "insanlar her Pazar süt ile beraber gazete de alırlar" gibi bir kural her zaman için geçerli olmayıp sadece periyodik olarak tekrar eden bir kuraldır. Kaynak [94]'te her t zaman süresince tekrarlayan durumlar için iki algoritma önerilmektedir. [95]'te ise takvimsel birliktelik kurallarını tespit edilmeye çalışılmıştır. Burada kullanıcının tarafından belirli bir takvim dönem bilgisi izlenmektedir. Bu çalışmalar özellikle tam periyodik bir yapıya sahiplerdir. Elde edilen kurallar, periyodik bir yapıdaki her zaman noktası için geçerli olmak zorundadır. Han ve arkadaşları [96] ise bu kısıtlamayı ortadan kaldırmak için kısmen periyodik örüntüler üzerine çalışmışlardır.

3.4.6. Ağırlıklandırılmış Birliktelik Kuralları

Birliktelik Kuralı algoritmalarında genellikle veri tabanı üzerindeki tüm nitelikler aynı önemi taşımaktadır. Cai ve arkadaşları [37], çalışmalarında her bir niteliğin kendi önemini yansıtmasını amaçlamıştır. Bu ağırlıklar, bazı ürünler üzerindeki promosyonlar veya ürünlerin fiyatları olabilmektedir. Bu durumda bazı nitelikler için ağırlıklandırılmış destek değeri kullanılması gerekir. Daha önceki birliktelik kuralı algoritmalarına sadece hesaplama yaparak ağırlıklı destek değerinin uygulanabilmesi mümkün olmamaktadır. Bu nedenle [37]'de yeni bir algoritma önerilmiştir.

3.4.7. Negatif Birliktelik Kuralları

Savasere ve arkadaşları [38] tarafından yapılan çalışmada, nitelikler arasındaki olumlu birliktelik kuralları yerine olumsuz birliktelik kuralları araştırılmıştır. $A \rightarrow B$ şeklinde gösterilen birliktelik kuralı $A \rightarrow \neg B$ şeklinde negatif birliktelik kuralı olarak gösterilmektedir. Yani A ürününü alan müşterilerin B ürününü almaması şeklindedir. Örneğin, "Asitli içecek alan insanların çoğu meyve almaz" şeklinde bir kural negatif birlikteli kuralı olarak ele alınmaktadır. Negatif birlikteliklerin bulunması için minimum destek ve güven değerlerinin mümkün

olduğunca azaltılması gerekir. Bu durumda $A \rightarrow B$ birliktelik kuralına ait destek ve güven değerleri kullanılarak $A \rightarrow \neg B$ şeklindeki negatif birliktelik kuralına ait destek ve güven değerini hesaplamak mümkündür [39]. Fakat bunun sonucunda çok miktarda ilginç olmayan kurallar da üretilmektedir. Kaynak [40]'ta bu durum için etkili ve yeterli bir algoritma bulunmaktadır.

4. BİRLİKTELİK KURALI KULLANARAK ÖZELLİK SEÇİMİ

4.1. Giriş

Özellik seçimi, sınıflandırma problemlerinde ve birçok öğrenme algoritmalarında önemli bir yer tutmaktadır. Özellik seçiminde amaç, konu ile ilgili olan, en faydalı ve en önemli özelliklerin seçilerek veri tabanına ait özellik sayısının azaltılmasıdır. Bu sayede daha az hesaplama yükü ile daha yüksek başarım elde edilebilmektedir. Özellik seçimi yöntemleri ile ilgili literatürde çok sayıda araştırma bulunmakta ve birçok alanda uygulamaları ile karşılaşılmaktadır. Bunlar [41, 97];

1) Çok sayıda sensörler tarafından veri elde edilen uygulamalar,
2) Sınıflandırma amacıyla bir araya getirilen çoklu modellere ait matematiksel parametrelerin sayısının fazla olduğu durumlar,
3) Veri madenciliği uygulamalarında, çok sayıda nitelik arasında bulunan gizli ilişkileri elde etmek amacıyla,
4) Giriş uzayı boyutunun azaltılması istenilen uygulamalardır.

Özellik seçimi yöntemlerini iki sınıfta incelemek mümkündür. Bunlardan birincisi istatistiğe dayalı yöntemler ikincisi ise sınıflandırmaya dayalı yöntemlerdir [97]. İstatistiğe dayalı yöntemler sınıflandırma probleminden bağımsız olarak en uygun özellik alt kümesini istatistiksel yöntemlerle tespit etmektedir. Sınıflandırmaya dayalı yöntemler ise sınıflandırma problemi için faydasız olan özellik alt kümelerinin tespit edilip atılmasına veya en önemli özelliklerin bulunması ölçütüne dayalıdır.

4.2. Özellik Seçimi Yöntemleri

Özellik seçimi problemi, birçok araştırmada ele alınmıştır. Bu alandaki en önemli yöntemlerden biri Temel Bileşen Analizi (TBA) yöntemidir [42, 43]. TBA, bir taraftan özgün verinin en önemli karakteristik özelliklerini açığa çıkartırken bir taraftan da verinin boyutunu azaltarak ayırt edici özellikleri içeren bilginin kullanılmasını sağlamaktadır. Bu yöntem var olan özelliklere bir dönüşüm uygulayarak yeni bir forma çevirmekte ve özellik sayısını azaltmaktadır. Dönüşümle elde edilen yeni veri kümesi, daha az nitelik değerinden oluştuğu için bundan sonraki işlemlerde hesaplama yükü azalmış olacaktır. Fakat bu yöntem yeterince cazip olamamaktadır. Çünkü probleme yeni bir verinin eklenmesi durumunda tüm verilerin tekrar işleme sokularak yeni dönüşümün yeniden elde edilmesi gerekmektedir.

Özellik seçimi yöntemlerinden bir diğeri ise Doğrusal Ayrışım Analizi (DAA) yöntemidir. Bu yöntem, nesne ya da olayları birden fazla sınıfa en iyi şekilde ayıracak olan özelliklerin doğrusal kombinasyonlarını bulmayı amaçlamaktadır. DAA sonucunda elde edilen kombinasyonlar, doğrusal bir sınıflandırıcı olarak ya da bir sonraki sınıflandırma işlemi için boyut azaltmak amacı ile kullanılabilmektedir. DAA'da sınıf içi farklılıklar en az, sınıflar arası farklılıklar en çok olmalıdır [44].

TBA ve DAA algoritmaları genellikle *d* boyutlu giriş uzayını *d'<d* olacak şekilde farklı bir uzaya aktarmaktadır. Özellik elde edilmesi esnasında zaman kazanılması istenen durumlarda ise TBA ve DAA gibi boyut indirgemesi yapan algoritmalar yerine özellik seçme yöntemleri kullanılmaktadır. İleriye doğru özellik seçimi (İÖS) ve geriye doğru özellik seçimi (GÖS) literatürde yaygın olarak kullanılan özellik seçimi yöntemleridir. Bu yöntemler, her niteliğin seçilecek özellik alt kümesine eklenmesi veya çıkarılması ile gerçekleşir. Her bir adımda, niteliğin etkisi bir sınıflandırıcı ile test ederek özellik kümesi seçimi yapan yöntemlerdir. Çok geniş veri kümeleri için bu yöntemlerin uygulanması olanaksız olabilmektedir. Ayrıca İÖS yöntemi, tek başına değersiz olan ancak diğer özellikler ile bir araya geldiğinde değerli olabilen özellikleri kaçırabilmektedir [45-49].

İÖS ve GÖS algoritmalarının dezavantajlarını ortadan kaldırmak için geliştirilen ve her iki yöntemin birleşiminden oluşan özellik seçimi yöntemi ise *l* ekle-*r* çıkar (plul *l* – takaway *r*) yöntemidir [50]. Bu yöntem *l* defa İÖS yöntemini *r* defa ise GÖS yöntemini kullanarak özellik seçimi yapmaktadır. Bu yöntem *d* özellik sayısını *d'* özellik sayısına ulaştırdığı zaman sona ermektedir.

Pudil ve ark. [51] tarafından önerilen özellik seçimi yöntemi ise *l* ekle-*r* çıkar yöntemi ile benzer yapı göstermektedir. Ancak Pudil ve arkadaşları tarafından önerilen yöntemde İÖS ve GÖS adımları dinamik olarak kontrol edilmektedir. GÖS adımı uygulanırken elde edilen özellik altkümesinin kötü başarım göstermesi durumunda GÖS adımı atlanarak tekrar İÖS adımı devreye girmektedir. Bu süreç istenilen özellik altkümesi elde edilinceye kadar devam etmektedir.

Bireysel özellik seçimi (BÖS) yöntemi ise her bir niteliğin kendi başına sınıflandırma başarım etkisini ölçerek özelliklerin birbirilerinden bağımsız olarak etkisini ortaya çıkarmaktadır. Çok başarılı bir başarım sağlamayacağı düşünülse de bazı veritabanlarında etkili olabildiği görülmektedir. Ayrıca [52, 53]'te *branch and bound* özellik seçimi yöntemi, [54, 55]'te genetik algoritma tabanlı bir özellik seçimi yöntemi ve bu algoritmaların değişik türevleri literatürde karşılaşılan diğer özellik seçimi algoritmalarıdır.

4.3. Birliktelik Kuralı Yöntemi Kullanılarak Özellik Seçimi

Birliktelik kuralı, veritabanlarında bulunan nitelikler arasındaki ilişkileri ortaya çıkarmak amacıyla kullanılan bir yöntemdir. Bu özelliği, birliktelik kuralının özellik seçimi yöntemi olarak kullanılabilmesini mümkün kılmaktadır. Eğer bir veya birkaç nitelik kümesi birbiri ile ilişkili ise bu niteliklerin bir altkümesi bir başka nitelik altkümesini tanımlar denilebilir. Bu görüş ile özellik seçimi yapabilmek için iki ayrı yöntem önerilmektedir [98].

Yöntem 1: Bu yöntem, tamamen üretilen birliktelik kurallarına dayalı olan Birliktelik Kuralı Tabanlı Özellik Seçimi (BKTÖS) yöntemidir. I = $\{i_1,i_2,..,i_m\}$ veri tabanına ait nitelikler olsun. $X \subseteq I$, $Y \subseteq I$ ve $X \cap Y = \varnothing$ olacak şekilde elde edilen bir $X \Rightarrow Y$ birliktelik kuralı için eğer bu kural yeterli destek ve güven değerine sahipse "*Y* nitelik altkümesi, *X* nitelik alt kümesi ile bağıntılıdır" veya "*X* nitelik alt kümesi, *Y* nitelik altkümesini tanımlayabilir" denilebilir. Bu sayede *Y* nitelik altkümesi, *I* nitelikler kümesinden çıkarılarak özellik çıkarma işlemi gerçekleştirilebilir.

Yöntem 2: Bu yöntem, özellikle sınıflandırma problemlerinde kullanılabilecek bir yöntemdir. Bu yöntemde birliktelik kuralı üretmek yerine, yeterli destek değerine sahip yoğun nesne kümeleri kullanılır. Yani Yoğun Nesne Kümesi Tabanlı Özellik Seçimi (YNKTÖS) yöntemi olarak adlandırılabilir. Ayrıca veri tabanı olarak tüm kayıtlar değil, her sınıfa ait kayıtlar ayrı ayrı ele alınmaktadır. Her bir sınıfa ait yoğun nesne kümeleri ayrı ayrı bulunmakta ve sadece buradan elde edilen nitelikler sınıflandırmada işleminde kullanılmaktadır. Bu sayede nitelik sayısı azaltılmış olacaktır.

Bu yöntemde, sadece her sınıfa ait yoğun nesne kümelerini bulmak özellik seçimi için yeterli olmayabilir. Eğer herhangi bir yoğun nesne kümesi tüm veri tabanı için yoğun olma özelliğini taşıyorsa bu nesne kümesinin her sınıf için de yoğun olması zaten olası bir durumdur. Bu nedenle veri tabanının tamamı ele alınarak elde edilen ve yeterli destek değerine sahip yoğun nesne kümesi içerisindeki niteliklerin belirlenmesi gerekir. Çünkü bu nitelikler sınıflandırma problemlerinde en etkisiz nitelikler olup özellik seçimi problemlerinde de kullanılmayacak nitelikler arasında olması gereken niteliklerdir.

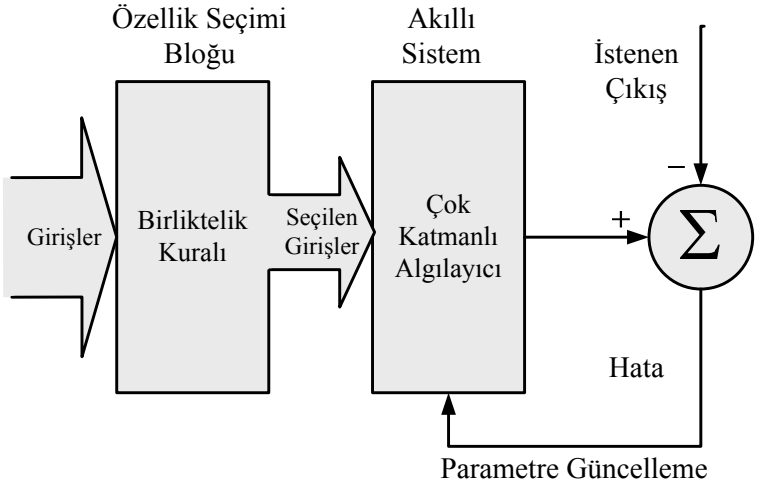
I = $\{i_1,i_2,..,i_m\}$ veri tabanına ait nitelikler, *X* = $\{x_1,x_2,..,x_n\}$ n sınıflı bir sınıflandırma probleminde her sınıfa ait yoğun nesne kümeleri ve *Y* ise veri tabanının tamamına ait yoğun nesne kümeleri olsun. Bu durumda özellik seçimi için kullanılacak nitelikler I_s olursa, I_s'i

$$I_s = \left(x_1 \cup x_2 \cup ... \cup x_n\right) - Y$$

olarak tanımlamak mümkündür.

4.4. Birliktelik Kuralı Tabanlı Özellik Seçimi Uygulamaları

Birliktelik kuralı ve yoğun nesne kümelerinin özellik seçimi amaçlı olarak kullanımı ve etkinliğinin tespiti için iki ayrı veri tabanı üzerinde uygulama yapılmıştır. Uygulamalarda, göğüs kanseri verileri ve dermatoloji alanında cilt hastalıkları ile ilgili veriler kullanılmış olup sınıflandırma işlemi gerçekleştirilmiştir. Öncelikle, birliktelik kuralı kullanılarak eldeki verilerden özellik seçimi işlemi yapılmış daha sonrada yapay sinir ağları kullanılarak sınıflandırma işlemi gerçekleştirilmiştir. Yapılan uygulamanın genel olarak blok diyagramı Şekil 4.1'de verilmektedir.



Şekil 4.1 Uygulamanın blok diyagramı

Şekil 4.1'de görüldüğü gibi, eldeki veri tabanına ait girişlere öncelikle birliktelik kuralı uygulanarak elde edilen veriler ışığında özellik seçimi yapılmıştır. Böylece giriş sayısı azaldığı için yapay sinir ağı sınıflandırıcının işlem yükü azaltılmış olacaktır. Ayrıca, birliktelik kuralı kullanarak tespit edilen ve kullanılmaması gereken girişler, sınıflandırma başarımını düşüreceği düşünülen girişler olduğu için sınıflandırma başarımında da artış beklenmesi olağan bir durumdur.

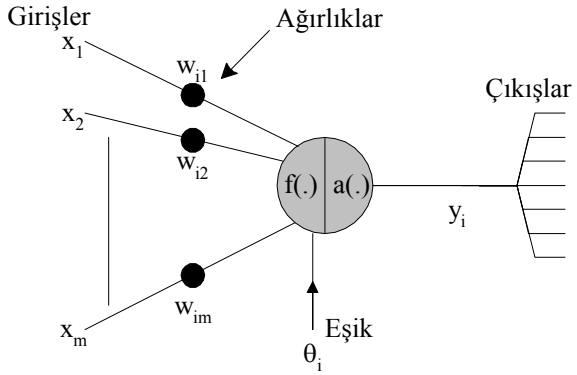
4.4.1. Yapay Sinir Ağı Sınıflandırıcıları

Yapay sinir ağları biyolojik nöron hücresinin yapısı ve öğrenme karakteristiklerinden esinlenerek geliştirilmiş bir hesaplama sistemi olup, örüntü tanımada çok kuvvetli sınıflandırıcılardır. Yapay sinir ağları, özellikle bilgisayar teknolojisinin gelişmesi ile

mühendislik sahasında çok geniş bir uygulama alanı bulmuştur [99-101]. Yapay sinir ağları aşağıdaki karakteristiklere sahip paralel bilgi işleme yapılarıdır:

- Biyolojik bir nörondan esinlenerek matematiksel modeli ortaya konmuştur.
- Birbirine bağlanan çok fazla sayıdaki işlem elemanlarından oluşur.
- Bağlantı ağırlıkları ile bilgiyi tutar.
- Bir işlem elemanı giriş uyarılarına dinamik olarak tepki verebilir ve tepki tamamen yerel bilgilere bağlıdır (ilgili işlem elemanını etkileyen bağlantılar ve bağlantı ağırlıkları yoluyla gelen giriş sinyali).
- Eğitim verisi ile ayarlanan bağlantı ağırlıkları sayesinde öğrenme, hatırlama ve genelleme yeteneklerine sahiptir.

Bu üstün özellikleri, yapay sinir ağlarının karmaşık problemleri çözebilme yeteneğini gösterir. Şekil 4.2'de biyolojik nörondan esinlenerek ortaya konmuş işlem elemanının basit bir matematiksel modeli gösterilmiştir [102]. Bu modelde, *i.* işlem elemanının çıkışı Denklem 4.1'de verilmiştir.



Şekil 4.2 Bir nöron hücresinin matematiksel modeli.

$$y(t+1) = a\left(\sum_{j=1}^{m} w_{ij}\, x_j(t) - \theta_i \right) \tag{4.1}$$

Burada *a(.)* etkinleştirme fonksiyonu, θ_i ise *i.* işlem elemanının eşik değeridir. İşlem elemanlarının bilgi işlemeleri iki kısımdan oluşur: Giriş ve çıkış. Bir işlem elemanı dışardan almış olduğu x_j giriş bilgilerini bağlı bulundukları w_{ij} ağırlıkları üzerinden birleştirerek bir *net* değeri üretir. *i.* işlem elemanının *net* değeri Denklem 4.2. ile hesaplanır.

$$f_i \triangleq net_i = \sum_{j=1}^{m} w_{ij}\, x_j - \theta_i \qquad (4.2)$$

Her bir işlem elemanının ikinci süreci, *net* değerini bir *a(.)* etkinleştirme fonksiyonundan geçirerek çıkış değerini bulmaktır. Etkinleştirme fonksiyonları işlem elemanlarının çok geniş aralıktaki çıkışını belli aralıklara çekmektedir. Böylece her bir işlem elemanının tepkisi yumuşak olmaktadır ve bağlantı ağırlıklarının değişimlerinin de daha küçük değerlerde olması sağlanır. Dolayısıyla yapay sinir ağının eğitiminde, hata değişiminin ıraksaması engellenerek kararlılığa ulaşmasına yardımcı olunur. Çok yaygın olarak kullanılan bazı etkinleştirme fonksiyonları: Birim basamak, signum, rampa, tek ve çift yönlü sigmoid gibi [102].

4.5. Göğüs Kanseri Veri Tabanı Üzerinde Uygulama

Göğüs kanseri, özellikle kadınlarda görülen ve çok yaygın olarak karşılaşılan kanser türlerinden biridir. Bu kanser türü, her sekiz kadından birinde yaşamları boyunca etkili olmakta, erkeklerde ise nadiren görülmektedir. Göğüs kanseri kötü huylu bir tümördür ve göğüs hücrelerinden gelişir. Bilim adamları, göğüs kanserinin gelişmesine yardımcı olan yaşlanma, genetik risk faktörü, aile geçmişi, çocuk sahibi olamama, oburluk v.b. gibi birçok etkeni bilmelerine rağmen henüz bu etkenlerin etkilerini tamamen ortadan kaldırabilecek çözümlere ulaşamamışlardır [103, 104].

Bu bölümde, göğüs kanseri teşhisi için birliktelik kuralı ve yapay sinir ağı tabanlı bir sınıflandırma yapısı gerçekleştirilmektedir. Sınıflandırma işleminde, Wisconsin'e ait göğüs kanseri verileri kullanılmıştır. Veri tabanına, önce birliktelik kuralı yöntemi uygulanarak elde edilen verilerden özellik seçimi yapılmış daha sonra ise seçilen özellikler kullanılarak sınıflandırma işlemi gerçekleştirilmiştir.

4.5.1. Wisconsin Göğüs Kanseri Veri Tabanı

Bu çalışmada, Wisconsin göğüs kanseri veri tabanı kullanılmıştır. Bu veri tabanı Dr. William H. Wolberg tarafından Wisconsin-Madison Hastanelerinde toplanmıştır. Bu veri tabanında toplam 699 kayıt vardır. Ancak bu kayıtlardan 18'inde veri eksikliği bulunmaktadır. Veri tabanındaki her bir kayıt 9 adet niteliğe sahiptir. Tablo 4.1'de bu özellikler verilmiştir.

Tablo 4.1 Wisconsin göğüs kanseri veri tabanının özellikleri.

No	Özellik Türü	Özelliklerin Değerleri	Ortalama	Standart Sapma
1	Hücre kalınlığı	1 - 10	4,42	2,82
2	Hücre boyutlarının benzerliği	1 - 10	3,13	3,05
3	Hücre şekillerinin benzerliği	1 - 10	3,20	2,97
4	Sınırsal uygunluk	1 - 10	2,80	2,86
5	Tekli epitel hücre boyutu	1 - 10	3,21	2,21
6	Tek çekirdekli	1 - 10	3,46	3,64
7	Esnek kromatin	1 - 10	3,43	2,44
8	Çok çekirdekli	1 - 10	2,87	3,05
9	Mitoz bölünme	1 - 10	1,59	1,71

Bu tabloda verilen özellik değerleri 1-10 değerleri arasına skalalanmıştır. Buna göre özellik değeri 1 ise özelliklerin normal olduğu ancak 1'den 10'a doğru gittikçe özellik değerlerinin anormalleştiği ve göğüs kanser ihtimalinin arttığı göz önünde bulundurulmalıdır. Bu veri tabanındaki 699 verinin, 241'i yani % 34,5'i göğüs kanseri için iyi huylu, 458'i yani % 65,5'i ise kötü huylu tümör olduğunu göstermektedir.

4.5.2. Uygulama Sonuçları

Winsconsin göğüs kanseri veri tabanı üzerinde birliktelik kuralı tabanlı özellik seçimi için öncelikle yukarıda önerilen yöntemlerin uygulanması gerekmektedir. Bu yöntemler sonucunda elde dilen bulgular aşağıda çıkarılmıştır.

Birliktelik kuralı tabanlı özellik seçimi yönteminin veri tabanına uygulanması sonucu 9 girişten sadece 1 tanesinin elenmesinin uygun olduğu tespit edilmiştir. Çünkü elde edilen birliktelik kuralı formu

Giriş : $1-3-8-9 \Rightarrow 2$

Değer : $1-1-1-1 \Rightarrow 1$ güven değeri= 100%.

şeklindedir. Yani bu kural yorumlandığında, veri tabanına ait 1., 3., 8. ve 9. niteliğin değerinin 1 olduğu durumlarda kesinlikle 2. niteliğin de değerinin 1 olduğu ortaya çıkmaktadır. Bu kural doğrultusunda ve önerilen yönteme göre 2. niteliğin değerinin zaten 1., 3., 8. ve 9. niteliğe bağlı olduğu ve sınıflandırma yapılırken kullanılmasına gerek olmadığı ortaya çıkmaktadır.

Yoğun nesne kümesi tabanlı özellik seçimi yöntemi, her sınıf için ayrı ayrı yoğun nesne kümelerinin bulunmasına dayalıdır. Göğüs kanseri veri tabanı, iki sınıftan meydana gelmektedir. Bunlardan birincisi iyi huylu tümör sınıfı ve ikincisi ise kötü huylu tümör sınıflarıdır. Birinci sınıf olarak ele alınan sınıfa birliktelik kuralı uygulandığında;

Giriş : 2-8-9

Değer : 1-1-1 (iyi huylu tümör sınıfı içi yoğun nesne kümesi)

elde edilmiştir. Yani 2., 8., ve 9. niteliklerin değerlerinin 1 değerini alması, veri tabanının bu sınıfında sık tekrarlanan bir özelliktir. Yine aynı şekilde ikinci sınıf olarak ele alınan sınıfa birliktelik kuralı uygulandığında;

Giriş : 6

Değer : 10 (kötü huylu tümör sınıfı içi yoğun nesne kümesi)

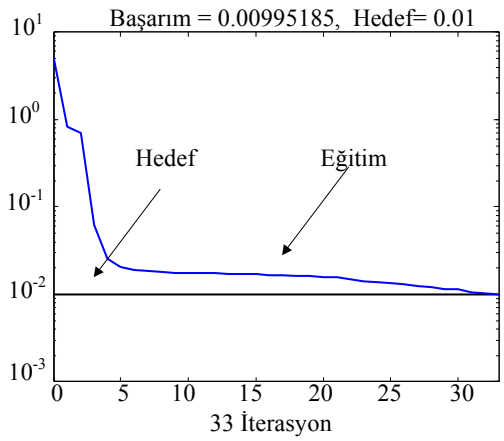
elde edilmiştir. Yani kötü huylu tümör sınıfında 6. niteliğin değerinin 10 olması sıklıkla görülen bir durumdur.

Bu sonuçlara göre iyi huylu tümör sınıfını 2, 8 ve 9. niteliklerin, kötü huylu tümör sınıfını ise 6. niteliğin iyi bir şekilde temsil edilebileceği kanısından yola çıkarak sınıflandırıcı olarak kullanılan yapay sinir ağı sınıflandırıcısına sadece 2., 8., 6. ve 9. nitelikler giriş olarak uygulanmış ve bu şekilde sınıflandırma yapılmıştır.

Bu aşamadan sonra yapay sinir ağı sınıflandırıcısı devreye girmektedir. Özellik seçimi amaçlı olarak önerilen ve her iki yöntemde de elde edilen sonuçlar kullanılarak sınıflandırma gerçekleştirilmiştir. Ayrıca diğer özellik seçimi yöntemleri ile de sınıflandırma yapılıp elde edilen sonuçlar karşılaştırılmıştır. Sınıflandırıcı eğitim ve test aşamalarında 3 katlı çapraz geçerlilik yöntemi kullanılmıştır. Kullanılan yapay sinir ağı sınıflandırıcıya ait özellikler Tablo 4.2'de eğitim başarımı ise Şekil 4.3'te verilmektedir.

Tablo 4.2 Çok katmanlı algılayıcı yapısı ve eğitim parametreleri

YSA Yapısı	
Katman Sayısı	3
Yapay Sinir Ağı Katmanlarındaki nöron sayısı	Giriş Sayısı : 4, 8, 9 Gizli Katman :11 Çıkış: 1
Başlangıç Ağırlıkları ve bias	Rasgele
Aktivasyon Fonksiyonu	Tanjant-sigmoid Doğrusal
Eğitim Parametreleri	
Öğrenme Kuralı	Levenberg–Marquardt Geriye yayılım
Hata Kareleri toplamı	0.01



Şekil 4.3 Yapay sinir ağı eğitim başarımı

Wisconsin göğüs kanseri veri tabanına ait sınıflandırma sonuçları karşılaştırmalı olarak Tablo 4.3'te gösterilmiştir. Tabloda, 3 katlı çapraz geçerlilik yöntemi kullanılarak elde edilen sonuçların ortalama değerleri verilmektedir. Tablodan görüldüğü gibi, 9 girişle ve yapay sinir ağı sınıflandırıcısıyla elde edilen sınıflandırma başarımı %92,2'dir. Oysa birliktelik kuralı kullanılarak ve BKTÖS yöntemi ile önerildiği şekilde özellik seçimi yapıldığında kullanılan özellik sayısı 1 azalarak 8 olmuş ve sınıflandırma başarımı %93,2 olarak hesaplanmıştır. YNKTÖS yöntemi ile önerilen şekilde özellik seçimi yapıldığında ise giriş sayısı 4'e inmiş ve %93,3 doğru sınıflandırma başarımı elde edilmiştir. Sonuç olarak en iyi doğru sınıflandırma başarımı YNKÖS yöntemi kullanılarak elde edilmiş olup ayrıca BKTÖS yöntemi ile de diğer özellik seçimi yöntemlerinden daha başarılı sonuçlar elde edilmiştir. Yani her iki yöntemin başarımı da tatmin edici sonuçlar vermiştir.

Tablo 4.3 Özellik seçimi yöntemleri ve YSA kullanılarak elde edilen sınıflandırma başarımları

Özellik Seçimi yöntemi ve YSA yapısı	**Kullanılan Nitelikler**	**Sınıflandırma Başarımı (%)**
İÖS + YSA (4, 6, 1)	1-2-4-6	92,7
GÖS + YSA (4, 6, 1)	1-4-6-8	93,0
l-Ekle, *r*-Çıkar + YSA (4, 6, 1)	1-2-4-6	92,7
BÖS + YSA (4, 6, 1)	1-2-3-8	90,8
Pudil +YSA (4, 6, 1)	1-2-4-6	92,9
Rasgele* + YSA (4, 6, 1)	3-4-5-7	90,3
YNKTÖS + YSA (4, 6, 1)	**2-6-8-9**	**93,3**
İÖS + YSA (8, 10, 1)	1-2-3-4-5-6-8-9	92,8
GÖS + YSA (8, 10, 1)	1-2-3-4-5-6-8-9	92,8
l-Ekle, *r*-Çıkar + YSA (8, 10, 1)	1-2-3-4-5-6-8-9	92,8
BÖS + YSA (8, 10, 1)	1-2-3-4-6-7-8-9	93,1
Pudil +YSA (8, 10, 1)	1-2-3-4-5-6-8-9	92,8
BKTÖS + YSA (8, 10, 1)	**1-3-4-5-6-7-8-9**	**93,2**
Tüm Nitelikler + YSA (9, 11, 1)	1-2-3-4-5-6-7-8-9	92,2

* literatürde var olan bir yöntem olmayıp sadece özelliklerin rasgele olarak seçildiğini göstermektedir.

4.6. Dermatoloji Veri Tabanı Üzerinde Uygulama

Kızartılı-Pullu hastalıklar, dermatoloji alanında görülen ve teşhisi zor olan hastalıklardandır. Kızartılı hastalıklar ile pullu hastalıklar birbirine oldukça benzemekle beraber çok küçük farklılıklarla birbirinden ayrılabilirler. Kızartılı-Pullu hastalıklar 6 sınıftan meydana

gelmektedir. Bunlar; Sedef hastalığı, yağlı egzama, liken planus, gül hastalığı, kronik egzama ve pitriasis rubra plaris olarak adlandırılan hastalıklardır. Bu hastalık sınıfları dermatolojide sıklıkla karşılaşılan hastalıklardır.

Bu bölümde, Kızartılı-Pullu hastalıklar için birliktelik kuralı ve yapay sinir ağı tabanlı bir sınıflandırma yapısı gerçekleştirilmiştir. Sınıflandırma işleminde, UCI (University of California-Irvine) makine öğrenmesi havuzu olarak adlandırılan veri tabanı arşivinden alınan veriler kullanılmıştır. Veri tabanına, önce birliktelik kuralı uygulanarak elde edilen verilerden özellik seçimi yapılmış ve seçilen özellikler kullanılarak sınıflandırma işlemi gerçekleştirilmiştir.

4.6.1. Kızartılı-Pullu Hastalıklar Veri Tabanı

Kullanılan veri tabanı 34 nitelikten ve toplam 366 kayıttan oluşmaktadır. Bu 34 nitelikten 12'si klinik olarak elde edilen veriler, kalan 22'si ise dokusal patolojik niteliklerdir. Bu nitelikler Tablo 4.4'te, kızartılı pullu hastalıklara ait sınıflar ise Tablo 4.5'te görülmektedir. Uygulama sırasında klinik olarak elde edilen özelliklerden olan "yaş" özelliğinde eksik veriler olduğundan bu özellik sınıflandırma yapılırken kullanılmamıştır.

Tablo 4.4 Kızartılı-pullu hastalıklar veri tabanının özellikleri

Klinik Nitelikler

Özellik No	**Özelliğin Tanıtımı**	**Özelliklerin değerleri**
1	Kızartı	0-3
2	Pul dökme	0-3
3	Belirlenmiş Sınırlar	0-3
4	Kaşıntı	0-3
5	Travma yerinde benzer lezyon çıkması	0-3
6	Poligonal kabartılar	0-3
7	Kıl kökünde çıkan kabartılar	0-3
8	Ağız içinin tutulumu	0-3
9	Diz ve dirseklerin tutulumu	0-3
10	Saçlı deri tutulumu	0-3
11	Aile öyküsü	0-1
34	Yaş	Doğrusal

Dokusal patolojik nitelikler

Özellik No	Özelliğin Tanıtımı	Özelliklerin değerleri
12	Belirli bölgede melanin hücre sayısının artması	0-3
13	Eozinofilin deride görülmesi	0-3
14	PNL (lökosit) birikmesi	0-3
15	Papiller dermis tabakasının kalınlaşması	0-3
16	Hücre dışına çıkma	0-3
17	Derinin spinal tabakasının kalınlaşması	0-3
18	Derinin epidermis tabakasının kalınlaşması	0-3
19	Derinin stratum korneum tabakasında çekirdekli hücrelerin görülmesi	0-3
20	Retelerin clubbing	0-3
21	Retelerin uzaması	0-3
22	Suprapapiller epidermisin kalınlaşması	0-3
23	İltihaplı kabarıklık	0-3
24	Munro mikroabseleri	0-3
25	Belirli bir alanda granüler tabakanın kalınlaşması	0-3
26	Granüler katmanın kaybolması	0-3
27	Bazal tabakanın su toplaması ve hasarı	0-3
28	Hücreler arasındaki sıvı miktarının artması	0-3
29	Retelerin kaybolması	0-3
30	Foliküler tıkaç	0-3
31	Kıl kökleri etrafında stratum korneum tabakasında çekirdekli hücrelerin bulunması	0-3
32	Mononükleer inflamatuar hücrelerin toplanması	0-3
33	Bant tarzında birikim	0-3

Tablo 4.5 Kızartılı-pullu hastalıklara ait sınıflar.

Sınıf kodu	Hastalık Sınıfı	Örnek Sayısı
1	Sedef	112
2	Yağlı Egzama	61
3	Liken Planus	72
4	Gül hastalığı	49
5	Kronik Egzama	52
6	Pitriasis Rubra Plaris	20
Toplam Kayıt Satısı (N)		366

4.6.2. Uygulama Sonuçları

Kızartılı-Pullu hastalıklar veri tabanı üzerinde özellik seçimi uygulaması için sadece önerilen YNKTÖS yöntemi kullanılmıştır. Çünkü BKTÖS yöntemi veri tabanına uygulandığında, yeterli destek ve güven değerine sahip $X \Rightarrow Y$ formunda bir kural üretilememiştir. Yani Kızartılı-Pullu hastalıklar veri tabanı, önerilen BKTÖS yöntemini uygulamak için elverişli verilere sahip değildir.

Özellik seçimi için önerilen YNKTÖS yöntemi uyarınca veri tabanındaki her sınıfın ayrı ayrı yoğun nesne kümelerini bulmak gerekmektedir. Kızartılı-Pullu hastalıklar, 6 ayrı sınıftan oluşmaktadır. Bu 6 sınıfa ait yoğun nesne kümeleri aşağıda çıkarılmıştır.

Sedef hastalığı sınıfı için yoğun nesne kümesi :
6-7-8-12-13-15-25-27-28-29-30-31-33

Yağlı egzaman hastalı sınıfı için yoğun nesne kümesi :
5-6-7-8-9-11-12-15-20-22-24-25-26-27-29-30-31-33

Lichen planus hastalığı sınıfı için yoğun nesne kümesi :
7-9-10-11-14-15-20-21-22-23-24-30-31

Gül hastalığı sınıfı için yoğun nesne kümesi :
6-7-8-9-10-11-12-13-15-20-21-22-23-24-25-27-29-30-31-33

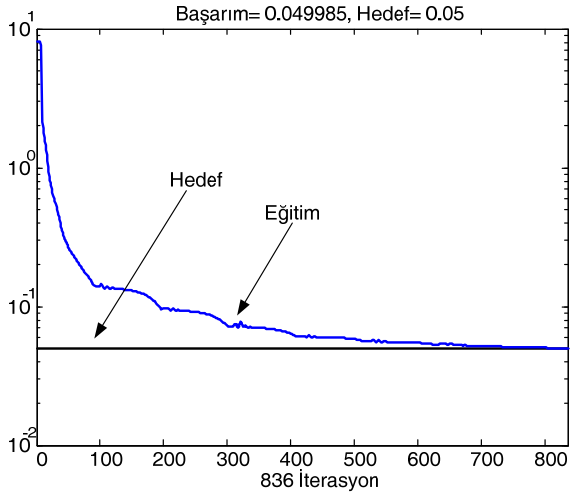
Kronik egzama hastalığı sınıfı için yoğun nesne kümesi :
5-6-8-9-10-11-12-13-14-20-22-23-24-25-26-27-29-30-31-33

Pityriasis Rubra Pilaris sınıfı için yoğun nesne kümesi :
5-6-8-12-13-15-20-21-22-23-24-25-26-27-29-33

Bu sonuçlara göre her sınıfı en iyi şekilde temsil edecek nitelikler tespit edilmiştir. Yani 33 niteliğe sahip olan Kızartılı-Pullu hastalıklar veri tabanında 1, 2, 3, 4, 16, 17, 18, 19 ve 32 numaralı niteliklerin çok fazla gerekli olmadığı anlaşılmaktadır. Bu aşamadan sonra yapay sinir ağı sınıflandırıcısı devreye girmektedir. Özellik seçimi amaçlı olarak önerilen ve YNKTÖS yöntemi ile elde edilen sonuçlar kullanılarak sınıflandırma gerçekleştirilmiştir. Ayrıca özellik seçimi yapılmadan da sınıflandırma yapılıp sonuçlar karşılaştırılmıştır. Sınıflandırıcı eğitim ve test aşamalarında 3 katlı çapraz geçerlilik yöntemi kullanılmıştır. Kullanılan yapay sinir ağı sınıflandırıcısına ait özellikler Tablo 4.6'da, eğitim başarımı ise Şekil 4.4'te verilmektedir.

Tablo 4.6 Çok katmanlı algılayıcı yapısı ve eğitim parametreleri

YSA Yapısı	
Katman Sayısı	3
Yapay Sinir Ağı Katmanlarındaki nöron sayısı	Giriş Sayısı : 24 Gizli Katman :26 Çıkış : 1
Başlangıç Ağırlıkları ve bias	Rasgele
Aktivasyon Fonksiyonu	Tanjant-sigmoid Doğrusal
Eğitim Parametreleri	
Öğrenme Kuralı	Levenberg–Marquardt Geriye yayılım
Hata Kareleri toplamı	0.05



Şekil 4.4 Yapay sinir ağı eğitim başarımı

Kızartılı-Pullu hastalıklar veri tabanına ait sınıflandırma sonuçları karşılaştırmalı olarak Tablo 4.7'de gösterilmiştir. Tabloda, 3 katlı çapraz geçerlilik yöntemi kullanılarak elde edilen sonuçların ortalama değerleri verilmektedir. Tablodan görüldüğü gibi, 33 girişle ve yapay sinir ağı sınıflandırıcısı kullanılarak elde dilen doğru sınıflandırma başarımı %97,6'dır. Oysa YNKTÖS yöntemi kullanılarak özellik seçimi yapıldığında kullanılan özellik sayısı 9 azaltılarak 24'e indirilmiş ve doğru sınıflandırma başarımı %97,9 olarak hesaplanmıştır.

Tablo 4.7 Özellik seçimi yöntemleri ve YSA kullanılarak elde edilen sınıflandırma başarımları

Özellik Seçimi yöntemi ve YSA yapısı	**Kullanılan Nitelikler**	**Sınıflandırma Başarımı (%)**
İÖS +YSA (24, 26, 1)	2-5-6-7-8-9-10-11-13-14-15-20-21-22-23-24-25-26-27-28-29-30-31-33	97,7
GÖS + YSA (24, 26, 1)	2-3-4-5-7-8-9-13-14-16-17-19-21-22-24-25-26-27-28-29-30-31-32-33	97,1
l-Ekle, *r*-Çıkar + YSA (24, 26, 1)	2-5-6-7-8-9-10-11-12-14-15-20-21-22-23-24-25-26-27-28-29-30-31-33	97,6
BÖS+ YSA (24, 26, 1)	2-3-4-5-6-7-8-9-10-11-12-14-15-16-18-20-21-25-27-28-29-30-31-33	96,6
Pudil +YSA (24, 26, 1)	2-4-5-7-8-9-10-11-12-15-16-17-18-19-21-22-24-25-26-28-29-30-32-33	96,6
Rasgele* + YSA (24, 26, 1)	2-3-5-7-9-11-12-13-14-16-17-19-20-21-22-23-24-25-26-28-29-30-31-32	96,1
YNKTÖS + YSA (24, 26, 1)	**5-6-7-8-9-10-11-12-13-14-15-20-21-22-23-24-25-26-27-28-29-30-31-33**	**97,9**
Tüm Nitelikler + YSA (33, 35, 1)	1-2-3-4-5-6-7-8-9-10-11-12-13-14-15-16-17-18-19-20-21-22-23-24-25-26-27-28-29-30-31-32-33	97,6

* literatürde var olan bir yöntem olmayıp sadece niteliklerin rasgele olarak seçildiğini göstermektedir.

Tablo 4.7'den görüldü üzere YNKTÖS yöntemi, dermatoloji alanındaki veri tabanı üzerinde en iyi özellik seçimi yöntemi olarak görülmektedir. Ayrıca tablodaki sonuçlar incelendiğinde özellik seçimi yöntemlerinin her zaman başarımı arttırmadığı da görülmektedir.

4.7. Sonuçlar

Bu bölümde, veri madenciliği tekniklerinden biri olan birliktelik kuralı, özellik seçimi yöntemi olarak kullanılmıştır. Özellik seçimi, sınıflandırma problemlerinde ve birçok öğrenme algoritmasında önemli bir yere sahiptir. Bu nedenle, iyi bir özellik seçimi yöntemi ile hızlı ve etkili bir sınıflandırma işlemi gerçekleştirmek mümkün olabilmektedir. Bu bölümde birliktelik kuralı kullanarak iki farklı veri tabanı üzerinde özellik seçimi yapılmış ve sınıflandırma işlemi gerçekleştirilmiştir. Birliktelik kuralı kullanarak özellik seçimi yapılırken ise iki farlı yöntem önerilmiş ve sınıflandırma başarımları diğer özellik seçimi algoritmaları ile karşılaştırılmıştır.

Elde edilen sonuçlara bakıldığında, her iki veri tabanı üzerinde de birliktelik kuralının özellik seçimi yöntemi olarak kullanılması, başarılı sonuçlar vermiştir. Birliktelik kuralı

üretilirken elde edilen yoğun nesne kümeleri, herhangi bir sınıflandırma probleminde her sınıfa ait önemli ve kayda değer nitelikleri ortaya çıkardığından başarılı bir performans göstermektedir. Veri tabanı üzerinde birliktelik kuralı üretildiğinde de elde edilen kurallardan herhangi bir niteliğin bir başka nitelik ile ilişkili olduğu tespit edilmesi sonucu özellik seçimi yapılması mümkün olmaktadır. Önerilen YNKTÖS ve BKTÖS yöntemlerinden her ikisi de göğüs kanseri veri tabanı üzerinde sonuç üretebilmiş ve nitelik sayısı 9'dan 4'e indirildiğinde en iyi sonuç %93,3 sınıflandırma başarımı ile YNKTÖS yönteminden elde edilmiştir. Göğüs kanseri veri tabanında, BKTÖS yönteminden elde edilen sonuç ise diğer özellik seçimi yönteminden daha başarılı olmuştur.

Dermatoloji veri tabanı üzerinde yapılan testlerde, veri tabanı üzerinde yüksek destek ve güven değerine sahip birliktelik kuralı üretilemediğinden sadece YNKTÖS yöntemi kullanılmıştır. Yine bu veri tabanından da elde edilen sonuçlara bakıldığında en iyi özellik seçimi yöntemi olarak %97,9 sınıflandırma başarımı ile YNKTÖS yöntemi karşımıza çıkmaktadır. Ayrıca bazı özellik seçimi yöntemlerinin başarımı düşürdüğü de göz önüne alındığında YNKTÖS yönteminin çok başarılı bir özellik seçimi yöntemi olduğu görülmektedir.

Ayrıca, birliktelik kuralı üretimi esnasında kullanılan ve çok önemli iki değer olan destek ve güven değerlerinin değiştirilmesi ile özellik seçiminde elde edilen alt özellik kümesi boyutun değiştirilmesi de mümkündür. Böylece özellik seçimi esnasında özellik sayısı, istenilen boyuta indirgenebilmektedir.

5. BİRLİKTELİK KURALI KULLANARAK DOKU SINIFLAMA

5.1. Giriş

Doku, manzara resimleri, uydu fotoğrafları ve biyomedikal görüntüleri de kapsayan, geniş bir yelpazede görüntü analizi için kullanılan önemli bir karakteristiktir. Diğer bir tanımla, bir görüntüdeki herhangi bir bölgenin istatistiksel özellikleri yavaş yavaş değişiyorsa veya yaklaşık olarak periyodikse, görüntünün o bölgesi doku özelliği gösteriyor denilir [105]. Dokuların insan görü sistemi tarafından algılanışı sınıflandırma ve sunum açısından da oldukça önemlidir [106, 107].

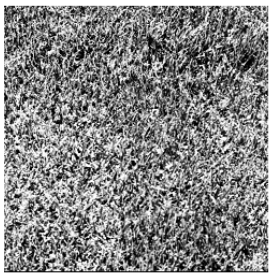
Doku sınıflama, bilinmeyen doku örneklerinin belirli bir kural veya kurallar dizisine bağlı olarak, daha önceden bilinen doku sınıflarına atanması işlemi olarak tanımlanır [56]. Literatürde, doku sınıflama problemine yönelik çözümlere sıkça rastlanılmaktadır. Weszka ve diğ [57], Fourier güç spektrumu öznitelikleri ile birinci ve ikinci dereceden gri seviyesi istatistiksel özellikleri kullanarak doku sınıflama için karşılaştırmalı bir çalışma gerçekleştirmişlerdir. Muneeswaran ve diğ. [58], doku görüntüleri için döndürmeden ve ölçeklendirmeden bağımsız yeni öznitelikler önermişlerdir. Önerilen yöntemde değişmez öznitelik çıkarımı için Meksika şapkası, Gauss ve ortogonal dalgacık aileleri kullanılmıştır. Li ve diğ. [59], doku sınıflama için diyadik dalgacık, dalgacık çerçeve ve Gabor özniteliklerini hem bireysel hem de kombine olarak kullanmışlardır. Sınıflandırıcı olarak ise Destek Vektör Makinelerini (DVM) kullanılmıştır. Yine doku sınıflama için DVM ve ayrık dalgacık çerçeve dönüşümü kullanılarak, döndürmeden bağımsız öznitelik çıkarımı Kwok ve diğ [60] tarafından önerilmiştir. Arivazhagan ve diğ. [56], dalgacık dönüşümü istatistiksel öznitelikleri, dalgacık eş oluşum matrisi öznitelikleri ve bunların birleşimini kullanarak farklı öznitelik vektörleri ile doku sınıflandırmışlardır. Önerilen yöntemde basit bir mesafe sınıflandırıcısı kullanılmıştır. Dokuların karakterize edilmesinde kullanılan ayrışım filtresinin ne kadar önemli olduğunu ve dokuları karakterize edecek baskın öznitelik vektörünün tayin edilmesini, Mojsilovic ve diğ. [61], araştırma makalelerinde göstermişlerdir. Mojsilovic ve diğ., 23 adet Brodatz doku örneği ile uygulamalar gerçekleştirmişler ve elde ettikleri sonuçlar ile kullanılan ayrışım filtresinin dokuların özelliklerinin çıkarılmasındaki etkisi ortaya çıkarılmıştır. Çalışmada en uygun 19 adet ayrışım filtresi tespit edilmiştir. Chen ve diğ.[62], doku sınıflama için önerilen filtreleme yöntemlerinin başarımlarını değerlendiren, karşılaştırmalı bir çalışma gerçekleştirmişlerdir. Çalışmada incelenen filtreleme yöntemleri, Fourier dönüşümü, uzamsal filtre, Gabor filtresi ve dalgacık dönüşümüdür. Çalışmanın deneysel sonuçları ile dalgacık dönüşümünün üstünlüğü

gösterilmiştir. Wang ve Liu [63], dokuları tanımlamak için çoklu çözünürlük bir MRA modeli önermişlerdir. Bu model hem MRA' nın avantajını hem de çoklu çözünürlüğün avantajını tek bir modelde birleştirerek, doku sınıflama için dayanıklı bir yapı oluşturmuştur. Bu model de ayrıca hem yüksek geçiren hem de alçak geçiren bileşenler kullanılmıştır.

Birliktelik kuralları, bir ilişkide bir niteliğin aldığı değerler arasındaki bağımlılıkları, anahtarda yer almayan diğer niteliklere göre gruplama yapılmış verileri kullanarak bulur. Keşfedilen örüntüler örneklemde sıklıkla birlikte geçen nitelik değerleri arasındaki ilişkiyi gösterir. Bu bağlamda doku görüntülerindeki benzerliklerinin ve piksel birlikteliklerinin keşfinde bu yöntemin kullanımı iyi sonuçlar verecektir. Birliktelik kurallarının doku sınıflama açısından incelendiği temel çalışma Rushing ve ark. [64] tarafından gerçekleştirilmiştir. Bu çalışmalarda dokuları karakterize eden piksellerin, dokudaki konumlarını ve parlaklık değerlerini göz önüne alarak kurallar üreten yeni bir yaklaşım sunulmuştur. Bu yöntem daha sonra bulut kümelerinin tanınması uygulamasında da kullanılmıştır [65].

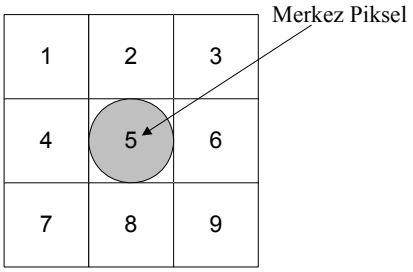
5.2. Resimdeki Birlikteliklerin Bulunması

Birliktelik kuralları, genel olarak market sepet analizi gibi alanlarda kullanılan ve büyük veritabanlarında bulunan nitelikler arasındaki birliktelikleri ortaya çıkarmaya yarayan bir tekniktir. Benzer yapısal özelliklere sahip bölgeleri bulunan yapılar doku olarak tanımlandığı için birliktelik kuralları dokular üzerindeki sık tekrarlanan yapıları tespit etmekte rahatlıkla kullanılabilir. Şekil 5.1'de Gri-seviyeli bir doku örneği verilmektedir. Dokular üzerinde birliktelik kurallarını uygulayabilmek için bazı terimlerin bilinmesi gerekmektedir. Bu terimler aşağıda verilmektedir.



Şekil 5.1 Doku örneği (Çim)

Merkez Piksel: Şekil 5.2'de görüldüğü gibi resmin *nxn* boyutundaki bir çerçevesinin tam ortasındaki pikseldir. Bu yüzden *NxN* ebatındaki bir resimde $(N-n+1)^2$ adet merkez piksel bulunmaktadır.



Şekil 5.2 3x3'lük çerçevede merkez piksel

Nesne: Her bir merkez piksel ve bu pikselin komşu pikselleri nesneleri oluşturmaktadır. Her nesne bir adres numarası ve bir yoğunluk değeri olmak üzere 2 bileşenden (X, Y) oluşmaktadır. X değeri pikselin hangi piksel olduğunu Y değeri ise bu pikselin nicemlenmiş (kuantalanmış) piksel değerini göstermektedir. Örneğin Şekil 5.2'de, 5 numaralı piksel merkez piksel olduğunu, 6 numaralı piksel, merkez pikselin sağındaki piksel olduğunu vb. göstermektedir. Nicemlenmiş piksel değerini gösteren Y değeri ise ileriki bölümde anlatılacaktır.

Nesne Kümesi: Nesnelerin bir araya gelmesi ile nesne kümelerini oluşmaktadır. Amaç resimlerde geçen yoğun nesne kümelerinin bulunmasıdır.

İşlem: Merkez piksel ve bu merkez piksele ait tüm komşu piksellerden meydana gelen nesne kümesi algoritmadaki her bir işlemi oluşturmaktadır.

Veri Tabanı (D): Elde edilen tüm işlemler bir satır olmak üzere bir araya geldiklerinde veri tabanını oluşturmaktadır. Doku tanımada kullanılacak tüm yoğun nesne kümeleri ve birliktelik kuralları bu veri tabanı üzerinden elde edilmektedir.

Birliktelik Kuralı: Her bir nesne iki değere(piksel yeri ve piksel değeri) sahip olduğuna göre, bir görüntüdeki birliktelik kuralı yapısı aşağıda belirtildiği formda gösterilir;

$$(X_1, Y_1) \Lambda \ldots\ldots \Lambda (X_m, Y_m) \rightarrow (X_{m+1}, Y_{m+1}) \Lambda \ldots\ldots \Lambda (X_{m+n}, Y_{m+n})$$

Yukarıdaki formda elde edilen, minimum destek ve güven değerini sağlayan tüm kurallar, doku görüntüsünü tanımada kullanılacaktır. Şekil 5.3'te 5x5 boyutunda örnek bir doku görüntüsü gösterilmiştir. Gri seviyeli olan bu doku görüntüsü üzerindeki yeni değerlerin nasıl hesaplandığı ileriki bölümde anlatılacaktır.

2	0	2	1	0
0	2	0	2	1
1	1	2	0	2
2	2	0	2	1
1	0	2	1	2

(5, 2) Λ (9, 2) → (6,0)

Destek: 3 / 9 = % 33,33

Güven: 3 / 5 = % 60

Şekil 5.3 Örnek bir doku görüntüsü ve birliktelik kuralı gösterimi

Şekil 5.3'te görüldüğü gibi 5x5 lik bir doku görüntüsü üzerinde, 3x3 boyutunda çerçeveler ele alındığında $(5–3+1)^2 = 9$ adet merkez piksel bulunmaktadır. Gri seviyeli (0–255) olan bu piksellere, bölüm 5.2'de anlatılan algoritma uygulandıktan sonra 0, 1 ve 2 değerlerini almıştır. Bu şekilde elde edilen örnek kurala bakacak olursak merkez pikselin 2 (5,2) değerini aldığı, merkezin sağ alt çaprazındaki pikselin 2 (9,2) değerini aldığı ve merkezin sağındaki pikselin 0 (6,0) değerini aldığı durum sayısı 3'tür. Bu durumda destek= 3/9 = 0.333 olmaktadır. Güven değeri ise Tablo 5.1'den de hesaplanabileceği üzere 3/5= 0.6 yani %60'tır.

Tablo 5.1 5,6 ve 9 nolu pikseller için nesne kümeleri ve destek değerleri

Nesne Kümesi	**Destek**	**Yoğun mu?**
(5,0)	3	Evet
(5,1)	1	Hayır
(5,2)	5	Evet
(6,0)	3	Evet
(6,1)	2	Hayır
(6,2)	4	Evet
(9,0)	2	Hayır
(9,1)	2	Hayır
(9,2)	5	Evet
(5,0) Λ (6,0)	0	Hayır
(5,0) Λ (6,2)	3	Evet
(5,0) Λ (9,2)	0	Hayır
(5,2) Λ (6,0)	3	Evet
(5,2) Λ (6,2)	0	Hayır
(5,2) Λ (9,2)	5	Evet
(6,0) Λ (9,2)	3	Evet
(6,2) Λ (9,2)	0	Hayır
(5,2) Λ (6,0) Λ (9,2)	3	Evet

Minimum Destek =%30

Örnek olarak verilen bu üç (5, 6 ve 9 numaralı pikseller) piksele ait tüm destek değerleri Tablo 5.1'de görülmektedir. Bir elemanlı (5,0) ve (6,2) nesne kümeleri tablodan görüldüğü gibi yoğun nesne kümeleridir. Bu nesnelere ait iki farklı birliktelik kuralı elde edilebilir. Bunlar (5,0)→(6,2) ve (6,2)→(5,0) kurallarıdır. Eğer minimum güven değeri 0,8 (%80) olarak alınırsa bu kurallardan sadece (5,0)→(6,2) kuralı geçerli kural olabilmektedir. Çünkü bu kuralların güven değerleri sırasıyla 1,0 ve 0,75'dir. Bu tablodan minimum destek ve güven değerlerini sağlayan tüm kurallar çıkarılmak istenirse Tablo 5.2'deki değerler elde edilir. Bu kurallar doku tanımada kullanılabilecektir.

Tablo 5.2 Şekilde gösterilen resim için üretilen birliktelik kuralları

Nesne Kümesi	Destek	Güven
(5,0) → (6,2)	0.33	1.0
(6,0) → (5,2)	0.33	1.0
(5,2) → (9,2)	0.55	1.0
(9,2) → (5,2)	0.55	1.0
(6,0) → (9,2)	0.33	1.0
(5,2) Λ (6,0) → (9,2)	0.33	1.0
(6,0) Λ (9,2) → (5,2)	0.33	1.0

5.3. Gri Seviyeli Doku Görüntüleri Üzerinde Birliktelik Kuralı

Gri seviyeli doku görüntüleri [0–255] arası değerler alabilen piksellerden oluşmaktadır. Doku görüntülerine birliktelik kuralını uygulayabilmek ve etkili sonuçlar alabilmek için dokular üzerinde bir dönüşüm uygulamak gerekir. 8 bitlik doku görüntüleri, birliktelik kuralı üretmek için elverişli değildir. Aşağıda gösterilen algoritma kullanılarak doku üzerindeki tüm piksel değerleri 0, 1 ve 2 olmak üzere üç seviyeye indirgenir.

Algoritma: Doku resmi üzerinde, herhangi nxn boyutundaki çerçeveye ait merkez piksel değeri şu şekilde hesaplanmaktadır [64]. Kullanılan çerçevedeki tüm piksellerin ortalama değeri μ ve standart sapması σ olsun. Buna göre merkez pikselin yeni değeri

$$yeni\ değer = \begin{cases} 0 & merkez \le \mu - c*\sigma \\ 1 & \mu - c*\sigma < merkez < \mu + c*\sigma \\ 2 & merkez \ge \mu + c*\sigma \end{cases} \qquad (5.1)$$

olarak hesaplanır. Denklemdeki *c* katsayısı [0.1–0.5] aralığında pozitif bir tam sayıdır. Bu denklemden elde edilen yeni merkez piksel değerleri, o pikselin komşu piksellere göre renk

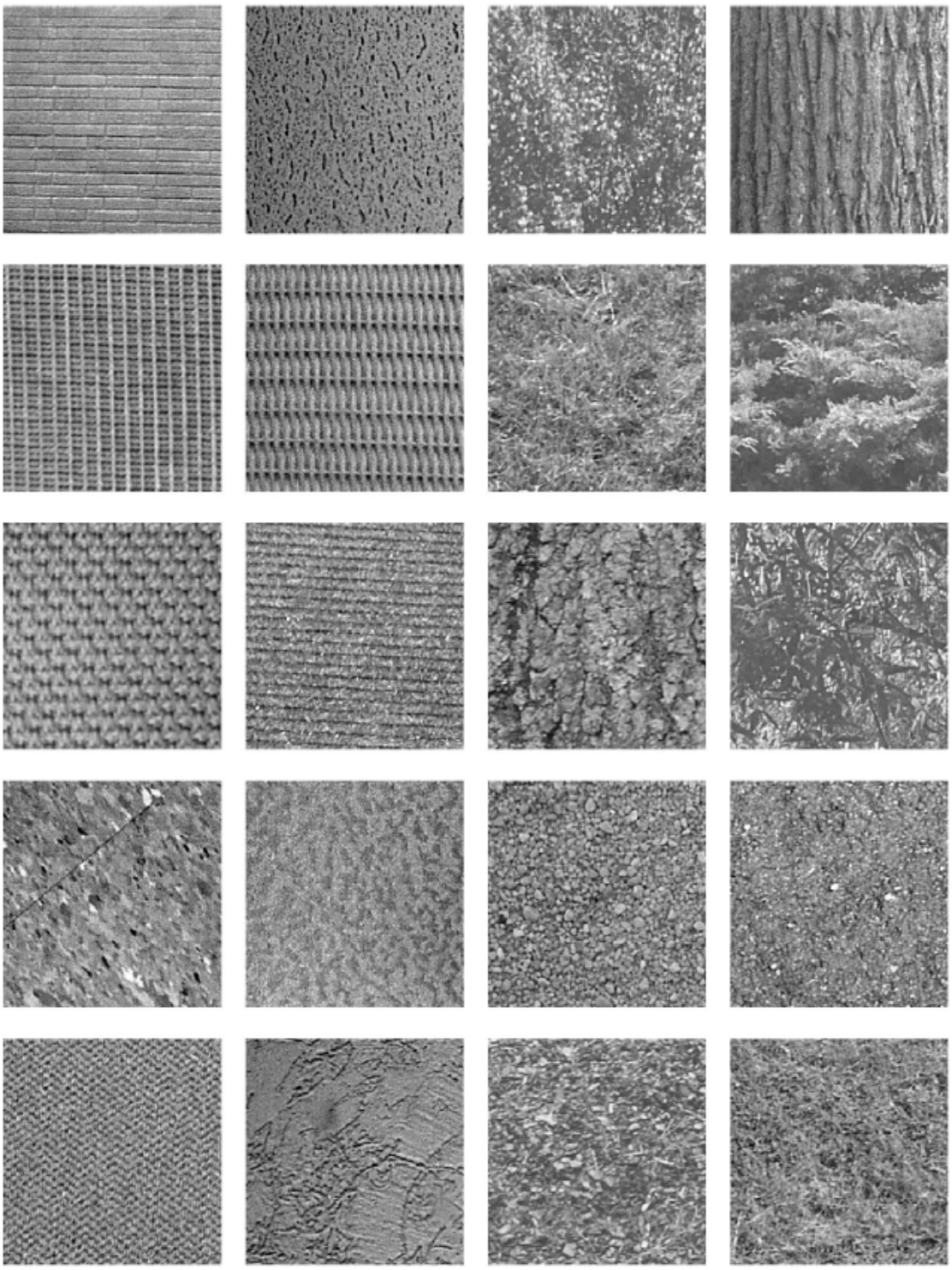
tonu yoğunluğu hakkında bilgi vermektedir. Denklemdeki *c* katsayısı değiştirilerek renk tonu yoğunluğuna ait hassasiyet değeri değiştirilebilir.

5.4. Birliktelik Kuralı Kullanılarak Doku Sınıflama

Doku sınıflama işleminin, bilinmeyen doku örneklerinin belirli bir kural veya kurallar dizisine bağlı olarak, daha önceden bilinen doku sınıflarına atanması olarak tanımlandığından daha önceden bahsedilmişti. Bu nedenle ilk önce, birliktelik kuralı kullanılarak doku sınıflamada kullanılacak kural veya kurallar dizisinin elde etmek gerekir. Birliktelik kuralları, bir doku üzerindeki yapısal özellikleri çıkarabileceği için bu yöntemle doku sınıflamada yüksek başarım elde edileceği beklenmektedir. Dokular üzerinde, belirli destek ve güven değerine göre elde edilecek kurallar, dokudan dokuya farklılıklar gösterecektir. Bu sayede başarılı bir sınıflama gerçekleştirmek mümkündür.

Doku sınıflama işlemindeki en önemli kısım, her bir dokuyu bir diğerinden ayırt edecek niteliklerin ve özelliklerin bulunmasıdır. Birliktelik kuralı tabanlı bir sınıflamanın etkinliğini değerlendirebilmek için diğer doku sınıflama yöntemleriyle karşılaştırılması gerekmektedir. Bu nedenle, birliktelik kuralı tabanlı sınıflamanın etkinliği, geniş kullanıma sahip ve kabul görmüş sınıflama yöntemleri ile karşılaştırılmıştır. Bu yöntemler, gri seviyesi eş oluşum matrisi (GSEM) [108], Gri seviyesi tekrarlama uzunluğu (GSTU) [109], fraktal boyut [110], Markov rasgele alanlar [111], Gabor dönüşümü [112] yöntemleridir.

Yapılan uygulamada daha güvenilir sonuçlar elde edebilmek için, kullanılan doku resimleri üç gruba ayrılmıştır. Bunlardan birinci grupta 10 adet insanlar tarafından üretilen ürünlere ait doku resimleri, ikinci grupta 10 adet doğal nesnelere ait doku resimleri bulunmaktadır. Üçüncü grup ise birinci ve ikinci grupta bulunan toplam 20 adet doku resminden oluşmaktadır. Bu görüntülerin tamamı Şekil 5.4'te görülmektedir.



Şekil 5.4 Sınıflamada kullanılan örnek doku resimleri

Sınıflama işlemlerinde çok sınıflı sınıflama yapısı kullanılmıştır. Bu nedenle sınıf sayısı arttıkça sınıflama başarımı da azalmaktadır. M-sınıflı bir sınıflama yapısı, kullanılan resimler içerisinden M tanesinin seçilmesi ile gerçekleştirilmektedir. Bundan sonraki uygulama, iki

aşamadan oluşmaktadır. Birinci aşamada her doku resminden 32x32'lik 32 adet resim parçası seçilerek sınıflandırıcı tasarlanmakta ve eğitim gerçekleştirilmektedir. İkinci aşamada ise yine her bir resimden elde edilen 32x32 lik 32 adet farklı parçaların kullanılmakta ve sınıflandırıcı test edilmektedir. Sınıflama başarımı sonuçları, doğru sınıflandırılan örneklerin toplam örnek sayısına oranının yüzdesi olarak hesaplanmıştır. Sonuçların daha geçeri ve tutarlı olması için, sınıflama işlemi on kez tekrarlanmış ve elde edilen sonuçların ortalama değerleri verilmiştir. Yapılan tüm uygulamalarda sınıflandırıcı olarak en yakın uzaklık ve maksimum olabilirlik metodu kullanılmaktadır. Sınıflama işleminde kullanılan yöntemler ve bu yöntemlere ait özellik ve parametreler Tablo 5.3'te verilmektedir.

Tablo 5.3 Sınıflama işlemlerinde kullanılan yöntemler ve parametreleri

Yöntem	**Özellik ve Parametreler**
Birliktelik Kuralı	Birliktelik kuralı yaklaşımında minimum destek değeri 0.05 ve minimum güven değeri 0.60 olarak alınmıştır. Doku resimleri 3, 4, 8, 16 ve 32 seviyeye indirgenmiş ancak en iyi başarım 3 seviyede elde edilmiş ve kurallar bu şekilde üretilmiştir.
GSEM (Gri-seviyesi eş - oluşum matrisi)	GSEM öznitelikleri (0, 1), (1, 0) ve (1, 1) konumsal operatörlerine bağımlıdır. Kullanılan öznitelikler, Entropi, tek biçimlilik, varyans, maksimum olabilirlik, elementsel fark momenti ve ters elementsel fark momentidir (toplam 24 öznitelik). 3, 4, 8, 16 ve 32 seviye ile kuantalanan görüntüler için gerçekleştirilen deneylerden en iyi sonucun 3 seviyeli kuantalama ile elde edildiği görülmüştür.
GSTU (Gri-seviyesi tekrarlama uzunluğu)	Gri-seviyesi tekrarlama uzunlukları, 0, 45, 90 ve 135 derecelik açı yönelimlerine bağlı olarak hesaplanmaktadır. Kullanılan öznitelikler, SRE, LRE, GLN, RLN, RP, LGRE ve HGRE dir (toplam 28 öznitelik). 3, 4, 8, 16 ve 32 seviye ile kuantalanan görüntüler için gerçekleştirilen deneylerden yapay doku görüntüleri için en iyi sonucun 3 seviyeli kuantalama ile elde edildiği, 4 seviyeli kuantalama ile ise doğal doku görüntüleri için en iyi sonucun elde edildiği görülmüştür.
Fraktal	Fraktal boyut (FB) özniteliklerini, orijinal görüntünün fraktal boyutu, düşük seviyeli gri ton görüntünün fraktal boyutu, yüksek seviyeli gri ton görüntünün fraktal boyutu, yumuşatılmış yatay görüntünün fraktal boyutu, yumuşatılmış dikey görüntünün fraktal boyutu ve 2. dereceden üssel çoklu fraktal öznitelikleri oluşturur.
MRA (Markov Rasgele Alanlar)	MRA öznitelikleri kaynak [113]'te anlatılan model parametreleri v* ve θ^*_i'a bağlıdır. Kullanılan 8, 12, 16 ve 20 komşuluk değerlerinden 16 yapay dokular için 12 ise doğal dokular için en iyi sonucu üretmiştir.

Gabor	Gabor öznitelikleri verilen filtre parametrelerine bağlıdır. φ(0, 90), σ(2), θ(0, 22.5, 45, 67.5, 90, 112.5, 135, 157.5) ve w(0.5, 0.8, 1.2, 1.6) olmak üzere toplam 64 adet filtre kullanılmaktadır. Uygulama ve öznitelik değerleri [111]'den alınmıştır. Filtre boyutu 3, 5, 7 ve 9 değerlendirilmiş ve yapay dokular için 5, doğal dokular için ise 3 en iyi sonucu vermiştir.

Tablo 5.4'te grup-1 resimler, grup-2 resimler ve grup-3 resimler için elde edilen sınıflama sonuçları verilmektedir. Görüldüğü gibi çok sınıflı sınıflama işlemlerinde sınıf sayısı arttıkça sınıflama başarımı azalmaktadır. Tüm sınıflama algoritmalarında aynı özellik geçerliliğini korumuştur. Ayrıca yine tablodan da görülebileceği gibi insan yapımı dokulara ait sınıflama başarımı doğal dokuların sınıflama başarımından daha iyi sonuçlar vermiştir. Ayrıca, kullanılan yöntemler incelendiğinde, birliktelik kuralı ve gabor yöntemlerinin diğer yöntemlerden daima daha başarılı olduğu görülmektedir.

Tablo 5.4 Kullanılan yöntemlerin doku grupları üzerinde doğru sınıflama başarımları

Doku Gurubu	**Yöntem**	**Sınıf Sayısı**								
		2	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Grup-1 İnsan Yapımı Dokular	Birliktelik Kuralı	99.1	98.5	98.5	98.4	96.9	96.0	95.2	94.1	94.4
	GLCM	96.2	92.9	88.7	86.3	83.8	81.7	79.3	76.8	74.4
	GLRL	99.1	98.2	96.6	95.6	93.5	92.1	90.5	89.2	88.1
	Fraktal	87.3	81.3	77.8	74.6	72.1	70.2	68.7	67.3	65.6
	Markov	90.3	85.9	81.2	78.1	77.5	75.9	74.5	73.4	71.2
	Gabor	99.8	99.1	97.9	97.2	96.1	95.4	94.8	94.4	94.0
Grup-2 Doğal Dokular	Birliktelik Kuralı	92.6	85.2	80.6	78.4	76.9	75.5	73.7	71.9	70.6
	GLCM	92.6	85.8	79.5	73.3	68.1	63.5	59.5	56.3	53.7
	GLRL	92.3	85.3	78.8	72.4	67.3	62.9	59.0	55.8	53.1
	Fraktal	82.6	75.5	67.3	61.7	57.2	53.8	50.5	47.8	45.6
	Markov	91.2	84.5	76.9	71.0	67.8	65.5	63.0	60.3	58.1
	Gabor	90.6	90.3	85.7	81.4	76.5	74.4	71.6	70.5	69.1
Grup-3 Karışık Doku	Birliktelik Kuralı	99.5	98.0	97.2	95.6	95.3	93.3	91.9	91.0	89.8
	GLCM	97.3	93.2	88.3	84.0	81.1	78.7	75.5	71.6	68.1
	GLRL	97.3	94.3	89.6	84.4	80.4	77.2	74.7	72.5	70.6
	Fraktal	92.3	84.0	78.6	74.4	70.4	67.4	65.2	62.8	60.2
	Markov	95.1	88.9	85.9	82.3	79.8	76.3	72.8	70.6	68.0
	Gabor	95.3	93.6	93.3	91.3	90.8	89.8	87.9	86.3	85.9

5.5. Kenar Çıkarma ve Birliktelik Kuralı Kullanarak Doku Sınıflama

Kenar çıkarma, görüntü işleme, bilgisayar görüsü ve örüntü tanıma alanlarında önemli bir yer tutmaktadır [105]. Bunun nedenleri, kenarların bulunmasının az zaman alması ve cisimlerin tanımlanmasıyla ilgili önemli ipuçları taşımasıdır [114]. Bu özellik birçok alanda olduğu gibi doku sınıflama alanında da birliktelik kuralı ile beraber kullanılabilecek bir

özelliktir. Böylece çok fazla olan hesaplama yükü hem azaltılmış hem de daha az zamanda istenilen sınıflama gerçekleştirilmiş olacaktır. Bu bölümde, [115]'ten elde edilen beş adet Brodatz'a ait 100x100 boyutunda doku görüntüsü kullanılmıştır. Bu dokulara, bu dokuların gürültü eklenmiş durumlarına ve bu dokuların ölçeklendirilmiş durumlarına kenar çıkarma algoritmalarından Canny kenar çıkarma algoritması uygulanarak deneysel çalışmalar gerçekleştirilmiştir.

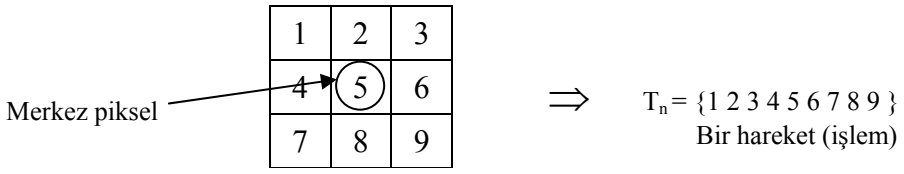
5.5.1. Kullanılan Yöntemler

Bu bölümünde doku görüntülerine birliktelik kuralı uygulamak için gerekli yöntem ve aşamalar açıklanacaktır. Bu aşamalardan birincisi, eldeki doku görüntülerinin kenarlarının çıkarılmasıdır. Kenar çıkarmadaki amaç, görüntünün içerdiği bilgiyi değerlendirip, nesnenin gereksiz ve tanıma işlemlerinde zaman kaybettiren bilgisini eleyerek yeterli düzeye indirgemektir. Kenarlar, görüntü analizinde ve nesnelerin sınıflandırılmasında çok değerli bilgiler taşıdığından, ilgi çeken bir araştırma alanıdır. Kenar çıkarmada önemli olan, nesnenin kenar çizgisinin tek piksel kalınlığında elde edilebilmesidir. Böylece nesneyi işlemek için daha az bilgi kullanıldığından, nesne tanımada oldukça önemli olan hız problemine zamandan tasarruf sağlanarak çözüm getirilebilir. Aynı zamanda nesne tanımak için oluşturulacak veri tabanları daha kolay oluşturulabilir. Bu çalışmada, kenar çıkarma algoritmalarından biri olan Canny kenar çıkarma yöntemi kullanılmıştır. Bu yöntem [114]'te detaylı olarak anlatılmıştır.

Kenar çıkarımı uygulanan doku resimlerine iki farklı yöntem kullanılarak veri tabanı (*D*) elde edilmiş ve bu *D*'ler üzerinden Birliktelik kuralı çıkarımı yapılmıştır. Bu yöntemler aşağıda anlatılmaktadır.

5.5.1.1. Yöntem-1

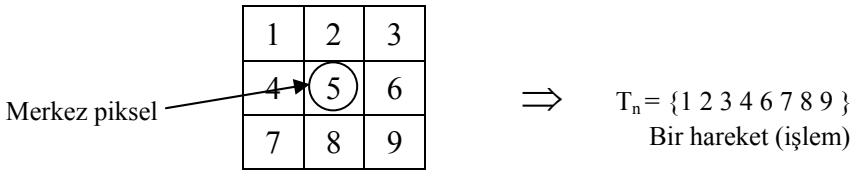
Şekil 5.7 c ve d' de kenar çıkarımı uygulanmış resimler görülmektedir. Bu resimlere birliktelik kuralı uygulamak için kullanılan yöntemlerden birincisi, tüm resmi referans almaktadır. Kenar çıkarımı gerçekleştirdikten sonra resmi oluşturan pikseller 1 ve 0 değerinden oluşmaktadır. Merkez kabul edilen her piksel ve o merkez piksele ait 8 adet komşu piksel birer nesne olarak kabul edilecek ve bunlar bir hareketi oluşturacak şekilde *NxN* boyutundaki resimden *(N-2)x(N-2)* hareket elde edilecektir. Birliktelik kuralı algoritması, bu hareketler üzerinde çalışacak ve negatif birliktelikler de bulunacaktır.



Şekil 5.5 Merkez pikselden bir hareketin (T) oluşturulması

5.5.1.2. Yöntem-2

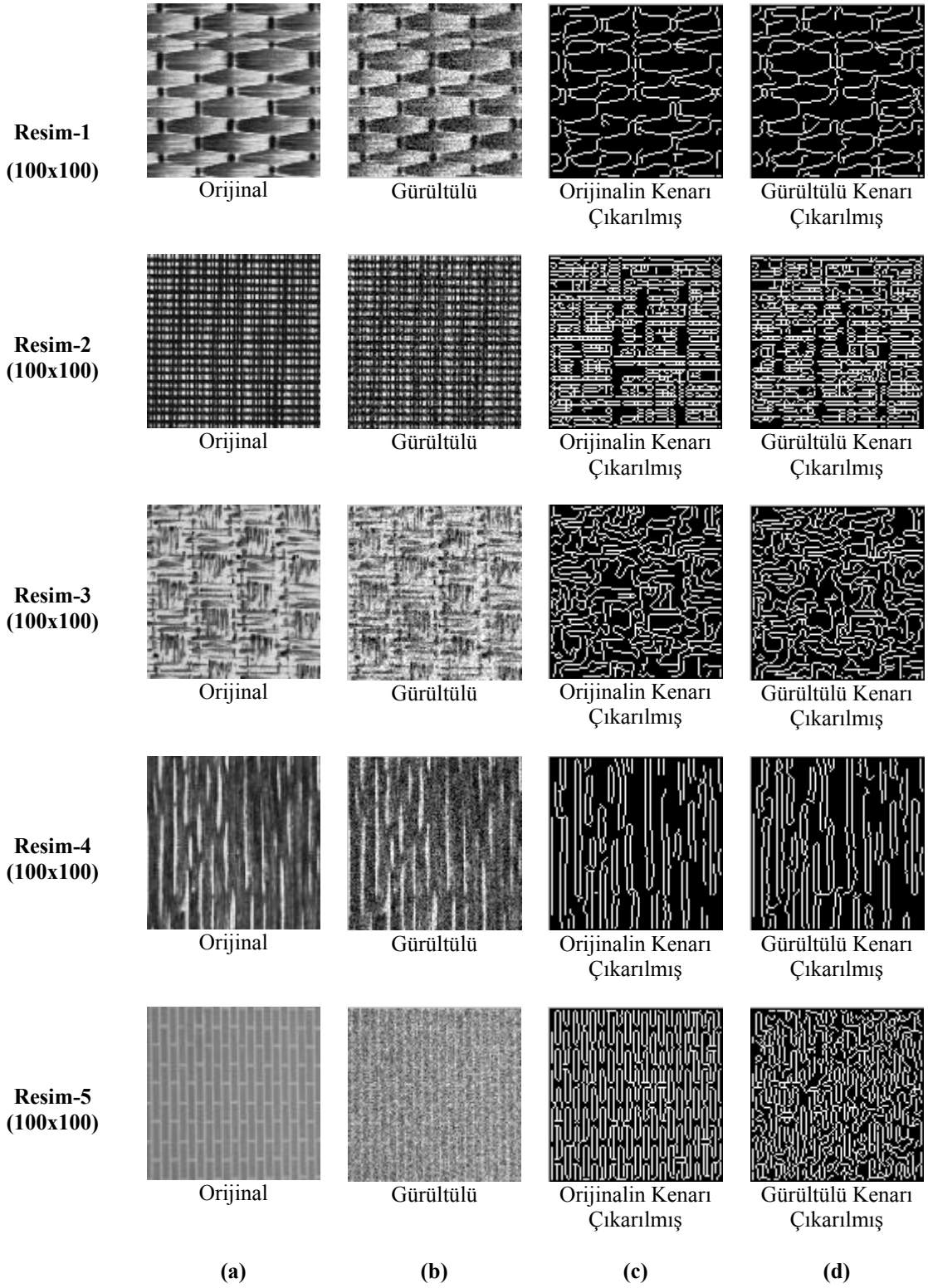
Bu yöntemde de yine Şekil 5.7 c ve d'de görülen kenar çıkarımı uygulanmış görüntüler kullanılacaktır. Ancak birinci yöntemden farklı olarak merkez kabul edilen tüm pikseller değil, sadece kenar olarak çıkarılmış pikseller referans alınacaktır. Bu da |D| nin boyutunda büyük ölçüde azalma sağlamaktadır. Ayrıca, seçilen merkez piksellerin tümü kenar olarak nitelendirildiği için ve 1 değerini aldığı için bu pikselin kullanılmasına da gerek kalmamaktadır. Kenar üzerinde olan her piksel için yine Şekil 5.6'da görülen hareketler oluşturulacak ve aynı birliktelik kuralı çıkarım algoritması bu hareketler üzerine uygulanacaktır. Şekil 5.6'dan da görüldüğü gibi tüm hareketlerdeki nitelik sayısı 9'dan 8'e inmiş yani 1 azalmıştır. Bu da birliktelik kuralı üretimi aşamasında hız kazancı sağlamış olacaktır.



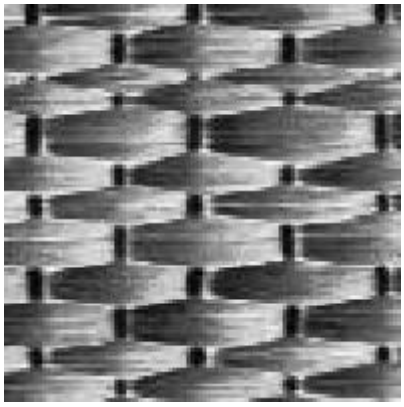
Şekil 5.6 Merkez piksel kullanılmadan bir hareketin (T) oluşturulması

5.5.2. Deneysel Çalışma

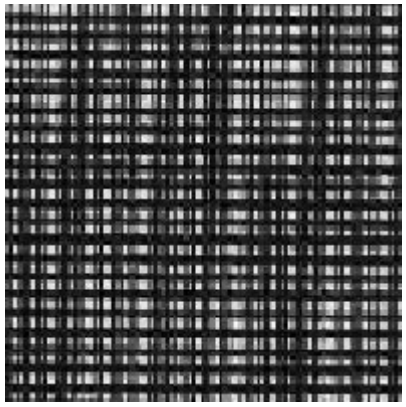
Yapılan çalışmada [115]'den elde edilen Brodatz'a ait beş adet doku görüntüsü kullanılmıştır. Bu resimlerin boyutu 100x100 olarak alınmış ve görüntülerin orijinali, gürültü eklenmiş hali, orijinalinin kenarı çıkarılmış hali ve gürültülü görüntünün kenarı çıkarılmış hali Şekil 5.7'de görülmektedir. Ayrıca Şekil 5.8'de, test amaçlı kullanmak amacıyla resimlerin 2 kat büyültülerek ölçeklenmiş hali de görülmektedir.



Şekil 5.7 Kenarı çıkarılmış doku resimleri

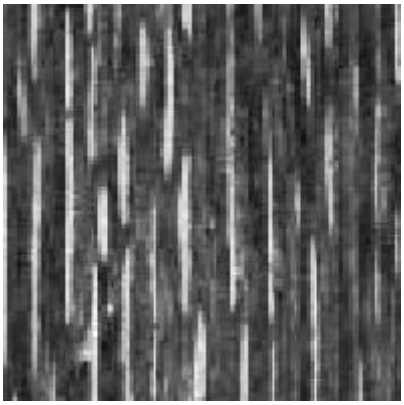


Resim-1 (200x200)

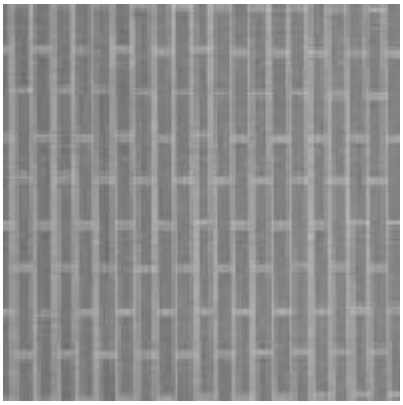


Resim-2 (200x200)

Resim-3 (200x200)



Resim-4 (200x200)



Resim-5 (200x200)

Şekil 5.8 Ölçeklenmiş doku resimleri

Tablo1'de birinci yöntemden elde edilen sonuçlar Tablo 2'de ise ikinci yöntemden elde edilen sonuçlar verilmiştir. Tablolardaki her satırda, doku görüntülerinden elde edilen Yoğun Nesne Kümelerinin (YNK) diğer görüntülerle yüzde olarak tanınma oranı verilmektedir. Her resim için YNK'lar *Ln*, bu YNK'ların destek değerleri ise *Sn* olsun. Bu YNK'ların, test için verilen görüntüdeki destek değerleri ise S_n^t olsun. Buna göre;

$$\% \text{ uygunluk} = 100 - \left(\sum \left| S_n - S_n^t \right| / n \right) * 100 / \text{ortalama (S)} \qquad (3.2)$$

formülü ile hesaplanmıştır [116].

Örneğin Tablo 5.5'te resim-1'in YNK'sı resim-2 ile yüzde 9,2 uymakta, resim-4 ile yüzde 79,6 uymaktadır. Tablo 5.6'da ise resim-1'in YNK'sı resim-2 ile yüzde 70,6 oranında uyarken resim-4 ile hiç uyum göstermemektedir. Bu da iki yöntemin görüntülerden görüntülere çok farklılıklar gösterdiğini ortaya çıkarmaktadır. Ancak hangi doku görüntüsü olursa olsun YNK'ya en yakın sonuç kendi resmine ait resimlerdir. Bu durum Tablo 5.7'de görülmektedir.

Tablo 5.5 Yöntem-1'den elde edilen sonuçlar

		Resim-1	Resim-2	Resim-3	Resim-4	Resim-5
100x100 YNK	Resim-1	**100**	9,2	46,5	79,6	5,3
	Resim-2	43,2	**100**	56,4	0,0	1,7
	Resim-3	1,8	31,1	**100**	21,9	10,3
	Resim-4	38,8	10,3	33,7	**100**	46,2
	Resim-5	6,7	2,6	13,8	76,7	**100**

Tablo 5.6 Yöntem-2'den elde edilen sonuçlar

		Resim-1	Resim-2	Resim-3	Resim-4	Resim-5
100x100 YNK	Resim-1	**100**	70,6	75,7	0,0	0,0
	Resim-2	65,8	**100**	87,1	0,0	1,6
	Resim-3	72,2	60,5	**100**	0,0	0,0
	Resim-4	13,1	1,8	16,5	**100**	67,9
	Resim-5	14,9	2,1	18,8	76,1	**100**

Tablo 5.7'de görüntülerin farklı işlemlerden geçirilmiş hallerinin, resimlerin YNK'larına ne kadar uyum gösterdiği görülmektedir. Bunun için resimlere 3 farklı işlem gerçekleştirilmiştir. Bunlardan birincisi resimlerin sadece 50x50'lik bir parçası, ikincisi resimlerin iki kat büyültülmüş hali, üçüncüsü ise resimlere gürültü eklenmiş halidir. Ayrıca bu tabloda hem yöntem-1'den hem de yöntem-2'den elde edilen sonuçlar karşılaştırmalı olarak verilmiştir.

Tablo 5.7 Farklı durumular için test ve yöntemlerin karşılaştırılması

	Resmin parçası Yöntem-1	Resmin parçası Yöntem-2	Büyük Resim Yöntem-1	Büyük Resim Yöntem-2	Gürültülü Resim Yöntem-1	Gürültülü Resim Yöntem-2
Resim-1	**99,8**	87,8	**93,7**	66,0	**96,8**	91,1
Resim-2	**97,9**	95,5	45,9	**66,5**	97,5	**97,8**
Resim-3	87,6	**92,9**	18,8	**67,4**	81,8	**93,8**
Resim-4	88,2	**95,2**	79,5	**97,9**	**90,5**	89,6
Resim-5	90,1	**93,4**	80,5	**92,7**	0,0	**47,6**

Tablo 5.7'den görüldüğü üzere görüntülerden elde edilen YNK'lar, resimlerin 50x50'lik bölümleri ile büyük oranda uyum göstermektedir. Buradaki sonuçlardan, yöntem-2'nin daha başarılı olduğu söylenebilir.

Dokuların iki kat büyültülmesi sonucu elde edilen görüntülerde de yine yöntem-2'nin yöntem-1'e göre daha başarılı olduğu görülmektedir. Sadece resim-2 ve resim-3'ün büyümeye karşı bozulması sonucu yöntem-1 tarafından çok fazla başarı elde edilmediği görülmüştür.

Görüntülere gürültü eklenmesi sonucu elde edilen değerlerde ise yine sadece resim-5 hariç diğerlerinin büyük oranda tanındığı görülmüştür. Şekil 5.6'dan görüldüğü gibi resim-5 in gürültü sonucunda aşırı bozulması yöntem-1 tarafından tanınmamasına yöntem-2 tarafından ise yüzde 47,6 oranında tanınmasına neden olmuştur. Yine gürültülü resimlerde de yöntem-2'in yöntem-1'e göre daha başarılı olduğu görülmektedir.

5.5.3. Değerlendirme

Bu çalışmadan görüldüğü üzere, doku görüntülerinin sınıflandırılması için kenar çıkarma ve birliktelik kuralların kullanıldığı yeni bir algoritma rahatlıkla kullanılabilmektedir. Doku, benzer yapısal özelliklere sahip az ya da çok değişim gösteren, tekrarlanan bir takım örüntülerdir. Geleneksel sınıflandırıcılar doku görüntülerini doğrudan kullandıkları için tekrarlamalı bölgelerin tekrar tekrar işleme dâhil edilmesi gereksiz hesaplama yükü ve zaman kaybına neden olmaktadır. Bunun yerine kenarları çıkarılmış doku görüntülerinin kullanılması hem sınıflama başarımını artırmakta hem de gerekli işlem yükünü azaltmaktadır. Önerilen yöntemin test sonuçlarından da görüldüğü gibi, beş adet doku görüntüsü, her bir dokuya gürültü eklenerek elde edilen görüntüler ve ölçeklendirme yapılarak elde edilen görüntülerde yöntem başarılı sonuçlar üretmiştir. İlgili sonuçlar bir önceki bölümde gösterilmiştir. Bu sonuçlardan önerilen yöntemin dayanıklılığı ve güvenirliliği görülmektedir.

5.6. Dalgacık Dönüşümü ve Birliktelik Kuralı Kullanarak Doku Sınıflama

Dalgacık dönüşümü, çoklu çözünürlük özelliğine sahip yaklaşımlardan biridir ve son zamanlarda doku analizinde kullanılan önemli bir araç olarak karşımıza çıkmaktadır [105]. Getirmiş olduğu çoklu çözünürlük özelliği ile farklı ölçütlerdeki dokuların dayanıklı tanınma başarımlarını artırmaktadır. Bundaki en büyük etkenlerden biri olarak da dalgacık dönüşümünde kullanılan düşük ve yüksek bant süzgeçlerinin, bir resmin farklı ardışık ölçeklerdeki parçaları için aynı kalabilmesidir. Bu özellikleri sayesinde dalgacık dönüşümünün birliktelik kuralı ile beraber etkili bir şekilde doku sınıflamada kullanılabileceği düşünülmektedir. Bu bölümde, dalgacık bölgesi birliktelik kurallarına dayalı doku sınıflama gerçekleştirilmiştir. Önerilen yöntem birliktelik kurallarını, doku görüntülerinin dalgacık dönüşümünden elde edilen detay görüntülerden üretmektedir. Yapılan çalışmada [115,117]'den elde edilen on adet Brodatz'a ait

doku görüntüsü kullanılmıştır. Deneysel çalışmalarda hem gri seviyesi birliktelik kuralları hem de dalgacık bölgesi birliktelik kuralları elde edilerek karşılaştırılmalı bir analiz yapılmıştır.

5.6.1. Dalgacık Dönüşümü

Dalgacık Dönüşümü (DD) işaretin ölçeklenebilir bir Zaman Frekans Gösterimi (ZFG) ile analizini sağlar ve geleneksel işaret işleme teknikleri olan Hızlı Fourier Dönüşümü (HFD) ve Kısa Zamanlı Fourier Dönüşümü (KZFD) tarafından görülmeyen detayları meydana çıkarır. KZFD' nin sınırlamalarından biri olan, kullanılan pencerenin sabit olması DD'de ölçeklenebilir bir pencere ile giderilmiştir. Böylece işaret içindeki düşük frekans eğilimlerini açmak için geniş bir pencere, yüksek frekans detaylarını analiz etmek için sıkıştırılmış bir pencere kullanılır. Bunun için, DD ölçeklenebilir temel bir dalgacık fonksiyonu kullanıp sabit çözünürlük problemine çözüm getirerek, işaretin farklı çözünürlüklerde daha esnek bir zaman bölgesi analizini yapar [112]. Dalgacık dönüşümü zaman – ölçek metoduna dayalı olduğu için, odağa tam olarak ayarlanarak işaretin farklı kısımlarını gözetleyebilen bir matematik mikroskobu gibi davranır[118]. Dalgacık analiz yöntemlerinin geleneksel yöntemlere göre üstünlükleri sıralanırsa [112]:

- Frekans spektrumundaki farklı bölgeler için daha kolay farklı frekans çözünürlükleri seçebilir.
- Eğer analizde spektrumdaki birkaç frekans bandı kullanılacaksa, tüm spektrumu hesaplamaya gerek yoktur.
- Spektrumun düşük frekans kısımlarında dalgacık dönüşümü oldukça hızlıdır.

Bu özelliklerinden dolayı, dalgacık dönüşümü durağan olmayan işaretlerin daha esnek ZFG 'lerinin elde edilmesinde etkili bir araç olduğu literatürde belirtilmiştir [119].

Dalgacıklar, veriyi farklı frekans bileşenlerine ayıran ve sonra kendi ölçekleriyle eşleştirilmiş bir çözünürlüğe sahip bileşenler üzerinde çalışan matematiksel fonksiyonlardır. İşaretin süreksizliklere ve keskin, sivri uçlara sahip olduğu fiziksel durumları incelemede, Fourier yöntemlerine göre daha avantajlıdır. Dalgacıklar matematik, kuantum fiziği, elektrik mühendisliği ve sismik jeoloji alanlarından bağımsız olarak geliştirilmişlerdir. Son on yıl içerisinde bu alanlar arasındaki yer değiştirmeler resim sıkıştırma, haberleşme, insanın görme gücü ve radar gibi birçok yeni dalgacık uygulamalarına yol açmıştır. Dalgacık dönüşümü, fonksiyonları, operatörleri veya veriyi farklı frekanstaki bileşenlerine ayıran ve ayrı ayrı her bileşen üzerinde çalışmamıza izin veren bir araçtır [112].

$\psi(x)$ ile gösterilen ana dalgacığın ötelenip genleşmesiyle elde edilen ve aşağıda tanımlanan $\psi_{m,n}(x)$ fonksiyonları bir birim küme oluştururlar şöyle ki;

$$\psi_{m,n}(x) = 2^{-m/2}\psi(2^{-m}x - n) \tag{5.3}$$

Burada m ve n tam sayılardır. Herhangi bir $f(x)$ sinyalinin $\psi_{m,n}(x)$ fonksiyonları ile çözümlenmesi DD olarak tanımlanır. $\psi_{m,n}(x)$ fonksiyonları birim dik bir küme oluşturduklarından $f(x)$ sinyali için, sırasıyla, aşağıdaki analiz ve sentez formülleri yazılabilir;

$$c_{m,n} = \int_{-\infty}^{+\infty} f(x)\psi_{m,n}(x)dx \tag{5.4}$$

$$f(x) = \sum_{m,n} c_{m,n}\psi_{m,n}(x) \tag{5.5}$$

Ana dalgacık iki aşamada tanımlanabilir. Birinci aşama aşağıdaki fark denklemini sağlayan bir ölçekleme fonksiyonu $\phi(x)$'in tanımlanmasıdır. İkinci aşamada ise ana dalgacık ölçekleme fonksiyonu ile ilişkilendirilir. Bu ifadeler sırasıyla aşağıda verilmiştir;

$$\phi(x) = \sqrt{2}\sum_{k} h(k)\,\phi(2x-k) \tag{5.6}$$

$$\psi(x) = \sqrt{2}\sum_{k} g(k)\phi(2x-k) \tag{5.7}$$

Burada $g(k) = (-1)^k h(1-k)$. Çözümleme aşamasında ölçekleme ve ana dalgacık fonksiyonlarının hesaplanması gerekmemektedir; h(k) ve g(k) fonksiyonlarını kullanılarak dönüşüm katsayıları yinelemeli olarak bulunabilir. J- seviyesinde bir çözümleme şu şekilde ifade edilebilir;

$$\begin{aligned} f(x) &= \sum_{k} c_{0,k}\phi_{0,k}(x) \\ &= \sum_{k}(c_{J+1,k}\phi_{J+1,k}(k) + \sum_{j=0}^{J} d_{j+1,k}\psi_{j+1,k}(x)) \end{aligned} \tag{5.8}$$

Burada $c_{0,k}$ katsayıları bilinmektedir ve j+1 seviyesindeki $c_{j+1,n}$ ve $d_{j+1,n}$ katsayıları, j seviyesindeki $c_{j,n}$ katsayıları ile şu şekilde ilişkilidirler ($0 \le j \le J$);

$$c_{j+1,n} = \sum_k c_{j,k} h(k-2n) \tag{5.9a}$$

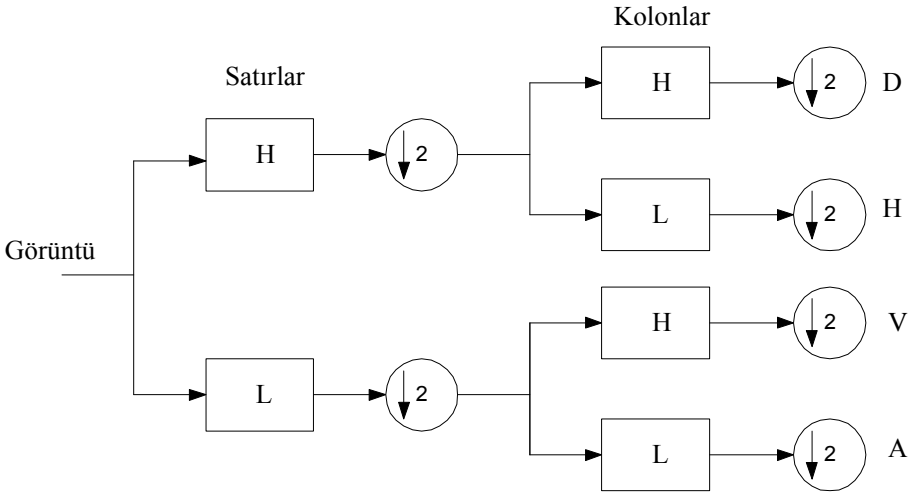
$$d_{j+1,n} = \sum_k d_{j,k} g(k-2n) \tag{5.9b}$$

Bu ifadelerin sinyal işleme yönünden yorumu ise şöyledir: j+1 çözünürlüğündeki $c_{j+1,n}$ ve $d_{j+1,n}$ katsayılarının elde edilmesi için, j çözünürlüğündeki $c_{j,n}$ ve $d_{j,n}$ katsayılarının $\tilde{h}(n)$ ve $\tilde{g}(n)$ fonksiyonları ile evriştirilip örnek sayısının iki kere seyreltilmesi gerekmektedir. Burada $\tilde{h}(n)$ ve $\tilde{g}(n)$ fonksiyonları, sırasıyla L alçak geçirgen dördün ikiz ve H yüksek geçiren dördün ikiz süzgeçleridir ve aşağıdaki şekilde tanımlanmışlardır;

$$\tilde{h}(n) = h(-n) \tag{5.10a}$$

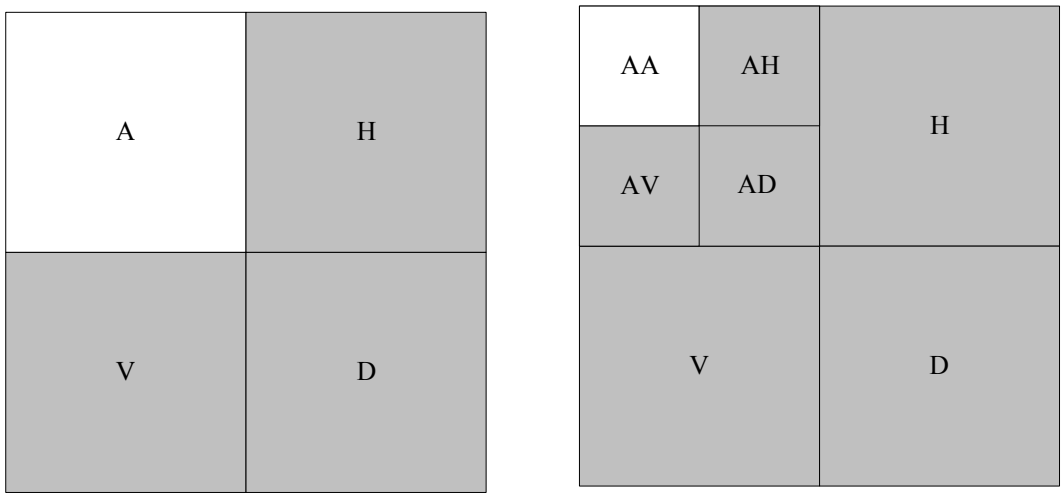
$$\tilde{g}(n) = g(-n) \tag{5.10b}$$

Bu işlemin sonucunda her j-seviyesindeki çözünürlük için alçak çözünürlüğe sahip $c_{j,n}$ katsayıları ve detayları gösteren $d_{j,n}$ katsayıları elde edilir. Şekil 5.9'da tek boyutlu dalgacık filtrelerin iki boyutlu görüntüye nasıl uyarlandığı gösterilmiştir. Görüntünün satırları ve sütunları ayrı ayrı alçak ve yüksek geçiren filtrelerden geçirilip, örnek sayısı yarıya düşürülür. Böylece görüntünün yaklaşığı ile detayları elde edilir.



Şekil 5.9 Tek seviyeli dalgacık analiz filtre bankası

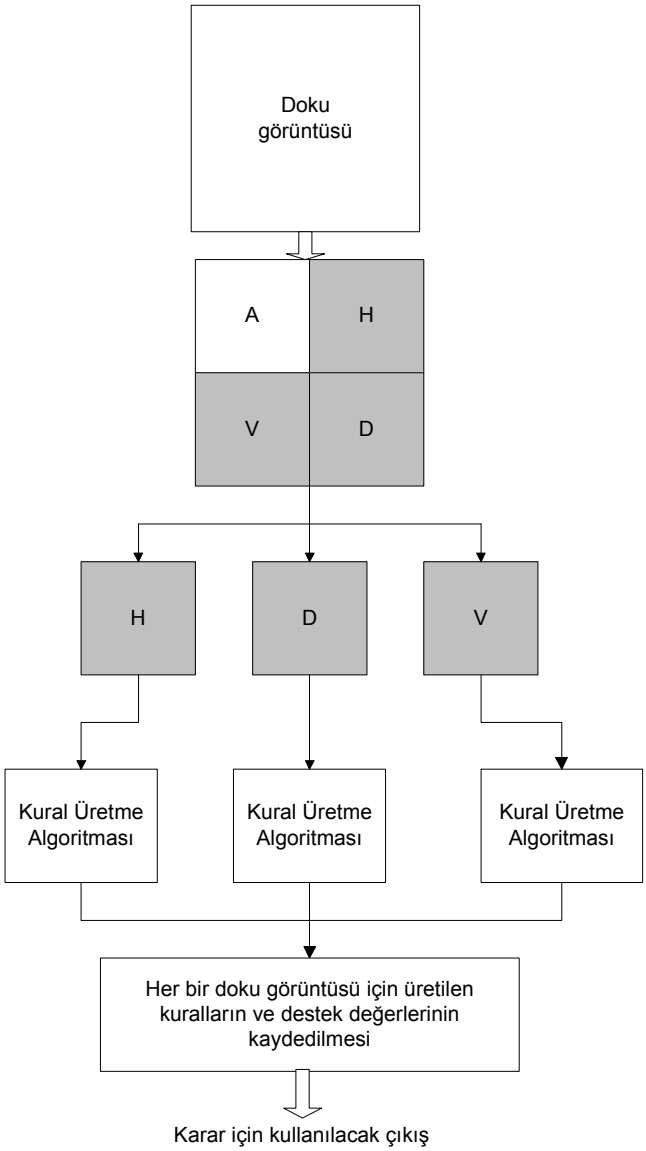
Sinyaller sadece alçak geçirgen bantların kullanılması ile piramit yapıda dalgacık dönüşümü ile alt bantlara ayrıştırılabileceği gibi tüm alt bantları kullanan dalgacık paket dönüşümü ile de alt bantlara ayrıştırılabilir. 2-boyutlu uzaydaki 2-düzeyli bir dalgacık paket ağaç yapısı Şekil 5.10'da gösterilmiştir. A harfi yaklaşığı, H harfi yatay, V harfi dikey, D harfi ise köşegensel detayı temsil etmektedir.



Şekil 5.10 DD ayrışımı (a) Tek seviyeli ayrışım (b) İki seviyeli ayrışım.

5.6.2. Yöntem

Dalgacık dönüşümü kullanılarak doku sınıflama yöntemin blok diyagramı Şekil 5.11'de verilmiştir. Şekil 5.11'den de görülebileceği gibi yöntemin ilk adımını dalgacık dönüşümü oluşturmaktadır. Burada 9/7 biortagonal (bior 4.4) dalgacık ayrışım filtresi, hesaplama etkinliği ve ayrılabilir özelliklerine binaen tercih edilmiştir [120]. Daha sonra, tek seviyeli ayrışım uygulanmış ve sadece detay katsayılar ileriki adımlar için kaydedilmiştir.

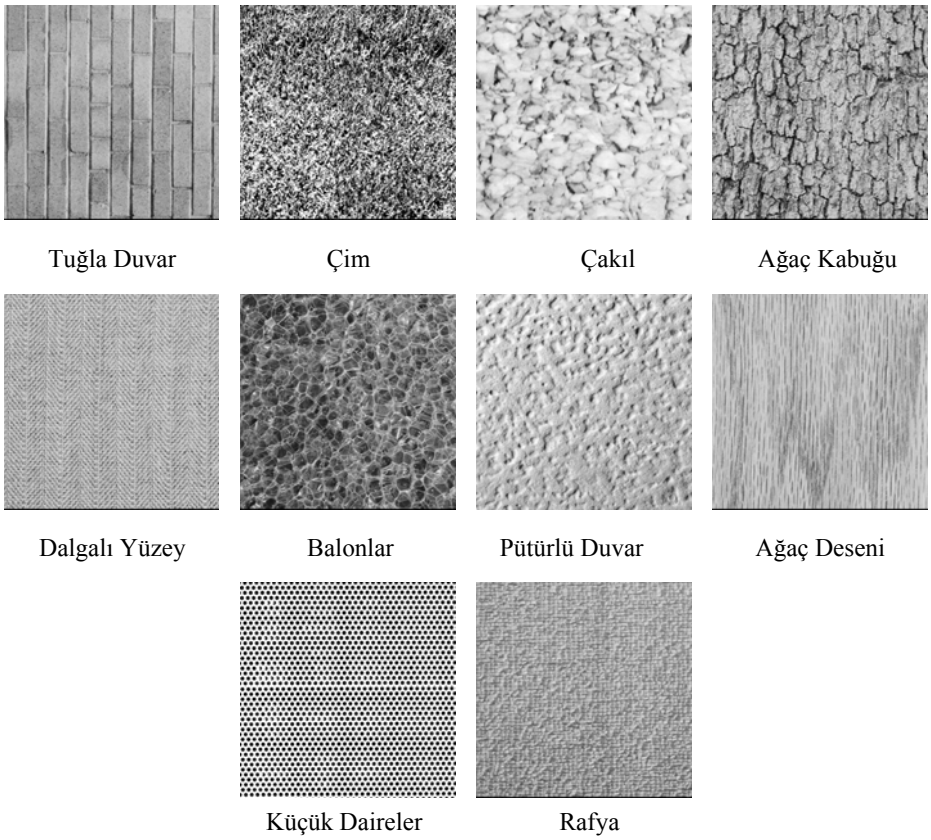


Şekil 5.11 Önerilen sistemin blok diyagramı

Önerilen metodun ikinci aşaması, birliktelik kuralının elde edilen örüntülere uygulaması aşamasıdır. Algoritmayı uygulayabilmek için örüntülerin işlemlere ve nesnelere dönüştürülmesi gerekmektedir. Bu aşama, doku resimlerimden özellikler çıkarabilmek ve bu özellikleri etkili bir biçimde kullanabilmek için çok önemlidir. Dalgacık dönüşümü uygulanmış doku resimlerine birliktelik kuralının uygulanması aşaması önceki bölümlerde bahsedilen yöntemlerle gerçekleştirilmiştir. Öncelikle DD'den elde edilen resimlere Bölüm 5.2'de anlatılan yöntem uygulanarak resimler 3 seviyeye indirgenmiş daha sonra da bu görüntüler üzerinde birlikteli kuralı uygulanarak sınıflama yapılmıştır [121].

5.6.3. Deneysel Çalışma

DD ve birliktelik kuralı kullanarak yapılan doku sınıflama çalışması, kaynak [115,117]'den elde edilen 512x512 boyutlu 10 adet Brodatz'a ait doku görüntüsü ile gerçekleştirilmiştir. Kullanılan doku görüntüleri Şekil 5.12'de görülmektedir. Önerilen yöntemin eğitim aşaması için görüntünün rasgele 128x128'lik bir parçası kullanılırken, test aşaması için ise orijinal görüntülerden rasgele örtüşen 128x128'lik yeni görüntüler oluşturulmuştur. Her bir doku görüntüsü için 50 adet test görüntüsü seçilerek toplam 500 adet 128x128 boyutunda test doku görüntüsü kaydedilmiştir.



Şekil 5.12 Kullanılan doku görüntüleri

Doku görüntüleri üzerine DD uygulandığında her bir doku görüntüsü için 3 ayrı DD detay görüntüsü elde edilmektedir. Örneğin, Şekil 5.12'de görülen çim dokusuna, DD uygulandıktan sonra Şekil 5.13'te görülen detay görüntüler elde edilmektedir. Bu görüntüler normalize edildiğinde iki seviyeye indiğinden ve boyutları gerçek boyutlarından daha küçük olduğundan dokuları temsil etme aşamasında değerli bilgiler taşımasının yanı sıra hız kaybı da yaşanmamaktadır.

H D V

Şekil 5.13 Çim dokusuna ait DD'den elde edilen detay görüntüler

Yöntemin eğitim aşamasında bilinen doku örneklerinden oluşturulan kurallar ve destek değerleri daha sonra test aşamasında kullanılmak üzere kaydedilmekte ve test için oluşturulan giriş doku görüntüleri de yine önerilen yöntemle işlenmekte ve elde edilen kurallar ve destek değerleri minimum mesafe sınıflandırıcısı ile önceden bilinen doku sınıflarına atanmaktadır. Tablo 5.8'de elde edilen sonuçlar görülmektedir.

Tablo 5.8 Dalgacık bölgesi birliktelik kuralları için başarım değerleri

Doku Adı	**Doğru**	**Yanlış**	**Toplam**
Tuğla Duvar	46	4	50
Çim	46	4	50
Çakıl	50	0	50
Ağaç Kabuğu	49	1	50
Dalgalı Yüzey	49	1	50
Balonlar	50	0	50
Pütürlü Duvar	50	0	50
Küçük Daireler	47	3	50
Rafya	50	0	50
Ağaç Deseni	50	0	50
Toplam	**487**	**13**	**500**

Bu yöntemle elde edilen ortalama başarım %97,4 olarak hesaplanmıştır. 5 adet doku görüntüsü %100 oranında doğru olarak sınıflandırılabilmiştir. Diğer taraftan önerilen yöntemin başarımını göstermek için aynı deneysel çalışmalar doku görüntülerinin gri seviyesi bölgelerinde de gerçekleştirilmiş ve ilgili sonuçlar Tablo 5.9'da verilmiştir. Sadece gri seviyesi bölgelerinde elde edilen toplam başarım %90,8 olup sadece 2 doku görüntüsü %100 oranında doğru sınıflandırılabilmiştir.

Tablo 5.9 Gri seviyesi birliktelik kuralları için başarım değerleri

Doku Adı	Doğru	Yanlış	Toplam
Tuğla Duvar	42	8	50
Çim	43	7	50
Çakıl	42	8	50
Ağaç Kabuğu	50	0	50
Dalgalı Yüzey	49	1	50
Balonlar	50	0	50
Pütürlü Duvar	48	2	50
Küçük Daireler	41	9	50
Rafya	44	6	50
Ağaç Deseni	45	5	50
Toplam	454	46	500

5.6.4. Değerlendirme

Deneysel çalışmalar sonucunda elde edilen istatistiksel göstergeler önerilen dalgacık bölgesi birliktelik kurallarının dokuları karakterize etmekteki üstünlüğünü açıkça ortaya koymuştur. Karşılaştırma amaçlı iki farklı deneysel çalışma gerçekleştirilmiştir. Yapılan deneysel çalışmaların ilki doku görüntülerinin gri seviyesi değerlerine bağlı olarak birliktelik kurallarının üretilmesini amaçlarken, ikinci deneysel çalışma ile doku görüntülerinin dalgacık dönüşü ile elde edilen detay görüntülerden birliktelik kuralı üretme temel amacı teşkil etmektedir. Deneysel çalışmalar ile önerilen yöntem ile gri seviyesi bölgesi için sırası ile %97,4 ve %90,8'lik başarım değerleri elde edilmiştir. Önerilen yöntemin tek dezavantajı, gri bölgesi birliktelik kurallarına oranla üç farklı detay görüntüsü için üç defa birliktelik kuralı üretilmesidir. Ancak bu durum aşırı bir zaman kaybına neden olmamaktadır.

5.7. Yöntemlerin Karşılaştırması ve Sonuçlar

Dokular, benzer yapısal özelliklere sahip bölgelerden oluştuğu için birliktelik kuralı tabanlı doku sınıflama problemleri yüksek başarım değerlerine sahip olabilmektedir. Eğer bir doku parçası, o dokuyu yeterince tanımlayabilecek boyutlarda ise bu doku parçalarının birliktelik kuralı ile sınıflandırılmalarında %100 başarım elde edilmesi mümkündür. Bu nedenle birliktelik kuralı tabanlı doku sınıflama işlemlerinde hız, işlem yükü ve kullanılan dokuların

boyutları daha fazla ön plana çıkmaktadır. Bu nedenle, yöntemler karşılaştırılırken hız ve işlem yükü açısından değerlendirilmiştir.

Yöntemlerin karşılaştırılırken 10-sınıflı bir sınıflama yapısı kullanılmıştır. Şekil 5.4'te görülen dokulardan rasgele 10 tanesi seçilmiş ve her resimden 100x100 boyutunda bir parça alınarak üç ayrı yöntemle yoğun nesne kümeleri ve birliktelik kuralları üretilmiştir. Daha sonra, bilinmeyen bir doku görüntüsü sisteme giriş olarak verilerek bu dokunun sınıflandırılması sağlanmıştır. Tablo 5.10'da, yöntemlerden elde edilen sınıflama sonuçlarının ortalama değerleri karşılaştırmalı olarak verilmektedir.

Tablo 5.10 Yöntemlerden elde edilen sonuçlar (Veri tabanı boyutu, nitelik sayısı, zaman ve başarım karşılaştırması)

Yöntem	**İşlem Sayısı |D|**	**Nitelik Sayısı**	**Zaman (sn)**	**Başarım (%)**
Gri Seviyesi+Birliktelik Kuralı	9.604	27	51	93,4
Kenar Çıkarma+Birliktelik Kuralı	2.528	16	18	92,8
Dalgacık Dönüşümü+Birliktelik Kuralı	1.570	16	36	94,2

Kenar çıkarma ve dalgacık dönüşümü tabanlı birliktelik kuralı yöntemleri, doku görüntülerini iki seviyeye indirmektedir. Bu sayede, Tablo 5.10'dan görüldüğü gibi hem işlem sayısında hem de nitelik sayısında kayda değer bir şekilde azaltma meydana gelmektedir. Böylece bu yöntemler oldukça hızlı bir şekilde sonuç üretmektedir. Dalgacık dönüşümü, doku görüntülerini üç detay görüntüye ayırdığı için biraz daha yavaş çalışmakta ancak dokular için daha önemli bilgiler içerdiğinden daha yüksek başarım üretilmesini sağlamaktadır.

Sonuç olarak bu bölümde, birliktelik kuralı kullanarak doku sınıflama gerçekleştirilmiştir. Literatürde var olan gri seviyesi birliktelik kuralı yerine kenar çıkarma ve dalgacık dönüşümü yöntemleri kullanılarak hem hız kazancı hem de sınıflama başarım kazancı sağlanmıştır. Kenar çıkarma, hem görüntülerle ilgili çok önemli bilgiler içermesi açısından hem de görüntüleri iki seviyeye indirgemesi açısından sınıflama esnasında başarım kaybı yaşatmadan büyük bir hız kazancı sağlamaktadır. Başarım düşüklüğü yaşandığı durumlarda ise sınıflandırılması istenen doku görüntüsünün boyutunun biraz daha büyük tutulması ile bu problem giderilebilmektedir. Doku boyutunun büyümesi, çok hızlı olan kenar çıkarma tabanlı birliktelik kuralı sınıflamada, hızı çok fazla etkilememektedir.

Dalgacık dönüşümü ise doku görüntülerini detay görüntülere ayırdığından, dokular için daha anlamlı bilgiler içerebilmektedir. Doku görüntülerinin daha iyi sınıflandırılabilmesi, elde edilen birliktelik kurallarının daha etkili ve belirleyici olması ile doğru orantılı olduğundan dalgacık dönüşümü bu aşamada daha etkili olabilmektedir. Bu işlem, üç farklı detay görüntüsü

kullandığı için ekstra bir yük getiriyor gibi görünmesine rağmen detayların görüntülerin boyutlarının daha küçük olması ve görüntülerin iki seviyeli olması olası bir dezavantajı ortadan kaldırmaktadır. Bunun yanı sıra dalgacık dönüşümü kullanarak daha yüksek başarım elde edilmektedir.

Hem kenar çıkarma hem de dalgacık dönüşümü tabanlı birliktelik kuralı doku sınıflama yöntemleri dokular üzerinde etkili ve sınıflama yapılabilmesine olanak sağlamaktadır. Ayrıca yapılan doku sınıflama işlemlerinde birliktelik kuralı yöntemi kullanılmasına rağmen kurallar üretmek yerine sadece yoğun nesne kümeleri kullanılmıştır. Bu sayede hem işlem yükü azalmakta hem hız artışı sağlanmakta hem de en önemlisi destek ve güven değerlerinin etkisinden bağımsız olarak sınıflama yapılabilmektedir.

6. BİRLİKTELİK KURALI KULLANARAK ÖĞRENCİ NOT TAHMİNİ

6.1. Giriş

Bilgisayar teknolojisindeki gelişmeler her geçen gün büyük bir hızla artmakta ve insan hayatının birçok alanında önemli bir yer tutmaktadır. Bilgisayar sistemlerinin geliştirilmesindeki en önemli amaç, insan yükünün hafifletilmesi ve daha hızlı işlem yapabilme yeteneğine sahip olabilmektir. Günümüzde, kişiyi anlamaya ve davranışlarını tespit etmeye yönelik yapılan testler ve yorumlarla birlikte, veri miktarı her geçen gün artmakta ve büyük veri yığınları oluşmaktadır. Bu veri yığınları arasından amaca uygun, gizli kalmış örüntülerin keşfedilmesi ve bu işlemlerin bilgisayar sistemleriyle yapılması günümüzde ön plana çıkmış alanlardandır. Böylece, özellikle eğitim alanında geliştirilen bu sistemlerle ve kişilerin belirlenen ölçütleri ile başarı ve başarısızlık sebeplerinin belirlenmesi, geleceğe yönelik öngörülerle şartların düzenlenmesi ve yönetilmesi sağlanabilir.

Veri madenciliği teknikleri, eldeki verileri kullanarak herhangi bir kişiyi tanımaya yönelik gizli kalmış önemli bilgileri ortaya çıkarmada oldukça etkili yöntemler içermektedir. Veri madenciliği teknikleri kullanılarak bu tür araştırmalar için değerli ve gerçekçi bilgiler sağlanabilir. Kaynak [66]'da veri madenciliği tekniklerinden birliktelik kuralı yöntemi kullanılarak öğrenci başarısı analizi yapılmıştır. Fırat Üniversitesi, Teknik Eğitim Fakültesi, Elektronik ve Bilgisayar Eğitimi Bölümü öğrencileri üzerinde yapılan çalışma ile öğrencilerin genel kültür derslerinden aldıkları notlar arasındaki ilişkiler çıkarılmıştır. Kaynak [67]'de de yine benzer bir çalışma yapılarak veri madenciliği ile bir öğrenci notları analiz aracı geliştirilmiştir. Geliştirilen bir web ara yüzü yardımıyla veri tabanı üzerinde işlem yapılabilmekte ve öğrencilerin dersleri ve o derslerden aldıkları notları arasındaki birliktelikler keşfedilmektedir. Kaynak [68]'de veri madenciliği ile, öğrenci ders başarısı öngörü aracı oluşturulmuştur. Öğrenciler üzerinde uygulanan anket sonucu elde edilen verilerden karar ağacı oluşturulmuş ve sınıflandırma işlemi gerçekleştirilmiştir. Kaynak [69]'da ise veri madenciliğinin öğrenci öğrenmesi alanında uygulamalarını içermektedir. Yapılan çalışmada sadece öğrenci notları kullanılmayıp öğrencilere ait bazı özlük verileri de kullanılmış ve bu veriler arasındaki ilişkiler elde dilmiştir.

Tezin bu bölümünde, birliktelik kuralı tabanlı bir öğrenci not öngörü ve başarı analiz modeli geliştirilmiş olup ayrıca bu amaca uygun bir yazılım hazırlanmıştır. Önerilen model, öğrenci not ve özlük verileri içerisinde gizli ve kayda değer ilişkiler bulmayı amaçlayarak

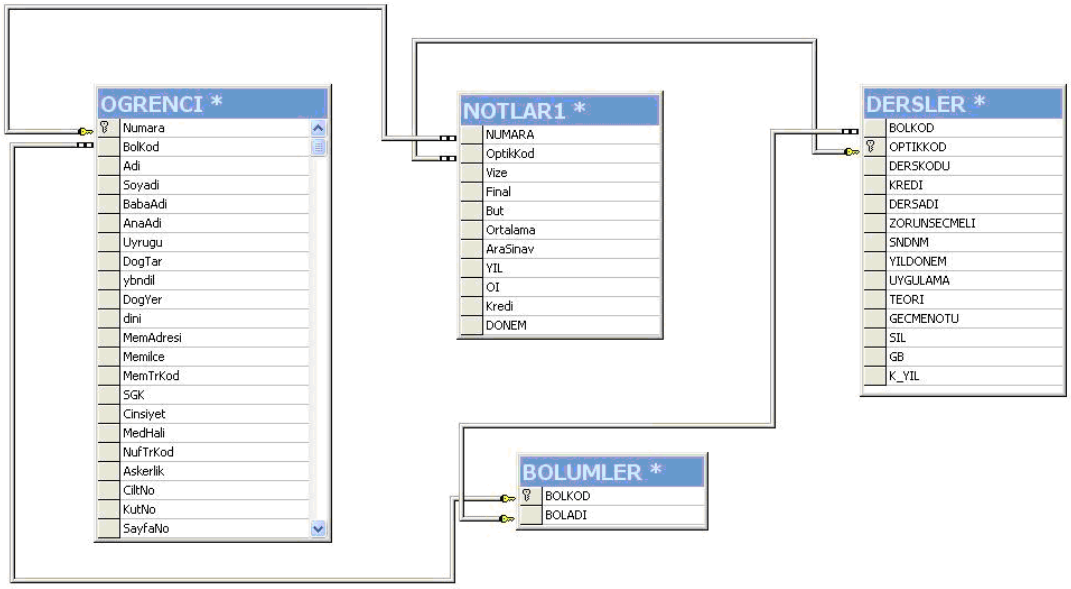
öğrencinin gelecekte derslerden alacağı notlarla ilgili öngörülerde bulunmakta ve içerisinde bulunduğu şartlara göre ders başarısını etkileyen durumlar ile ilgili bilgiler üretmektedir.

6.2. Öğrenci Veri Tabanı

Öğrenciler hakkında bilgi üretmek ve not öngörüsünde bulunmak için kullanılan en önemli araçlardan biri veri tabanıdır. Günümüzde bütün dijital veriler veritabanlarında tutulmakta ve bunun için veri tabanı yönetim sistemleri kullanılmaktadır. Bu çalışmada kullanılan veri tabanı, Fırat Üniversitesi Öğrenci İşleri Daire Başkanlığına ait olduğu için yapılan işlemler bu veri tabanı formatına uygun olarak gerçekleştirilmektedir. Veri tabanı yönetim sistemi programı olarak yapısal sorgu dili sunucusu (SQL Sunucusu) kullanılmaktadır.

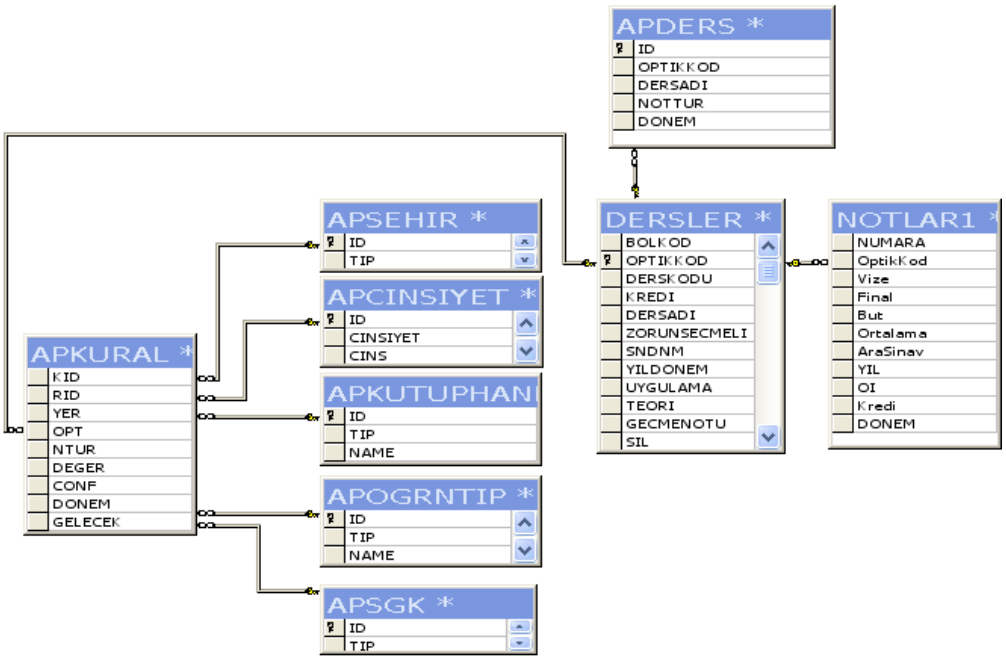
6.2.1. Veri Tabanının Genel Yapısı

Tablolar, veritabanlarının en önemli bileşenleridir. Her veri tabanında en az bir tane tablo mutlaka bulunmalıdır. Çünkü tüm veriler tablolarda saklanmaktadır ve her bir tablo belirli bir konuya ve amaca uygun verileri tutmaktadır. Öğrenci veri tabanı birçok tablodan meydana gelmekte ve öğrencilerle ilgili en önemli verilerin bulunduğu tablolar olan öğrenci, dersler ve notlar tablolarına ait SQL sunucusu diyagram görüntüsü Şekil 6.1'de görülmektedir. Bu tablolar haricinde kullanılan diğer tablolar genellikle veri tabanında normalizasyon sağlamak amacı ile kullanılan tablolardır.



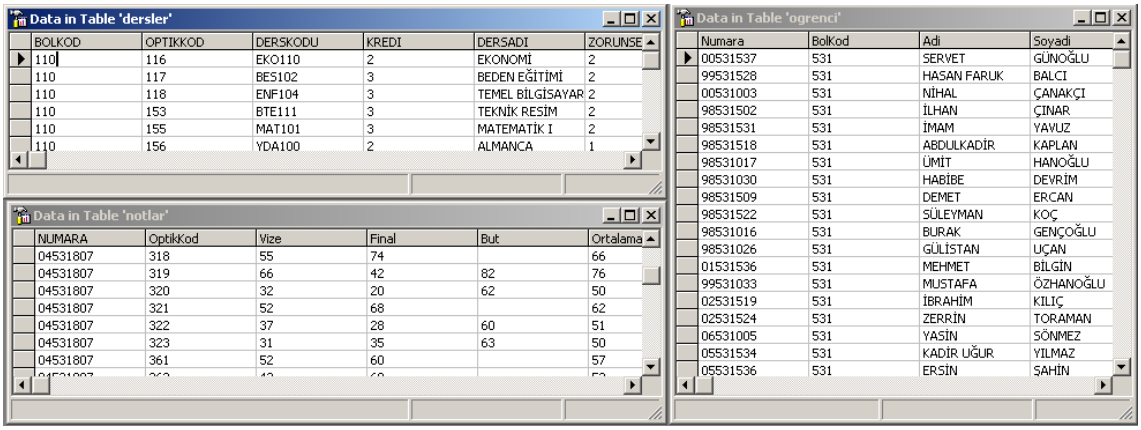
Şekil 6.1 SQL sunucu diyagramı

Birliktelik kuralı kullanarak veri tabanından bilgi keşfi yapabilmek ve ayrıca elde edilen bilgilerin veri tabanında tutulmasını sağlamak amacıyla da, veri tabanında bu konuyla ilgili ek tablolara ihtiyaç duyulmaktadır. Şekil 6.2'de birliktelik kuralı kullanarak kural üretebilmek ve saklamak için hazırlanmış tablolara ait diyagram yapısı görülmektedir. Bu tabloların bir kısmı, kural üretimine dâhil edilecek öğrenci ile ilgili kriterleri belirleme amaçlı, bir kısmı ise üretilen kuralları kaydetme amaçlı olarak kullanılan tablolardır. Bu tabloların işlevleri ileriki bölümlerde detaylı olarak anlatılacaktır.



Şekil 6.2 Birliktelik kuralı üretmeyi sağlayan tabloların SQL sunucu diyagramı

Öğrenci veri tabanına ait en önemli tablolar, öğrenci özlük verilerinin, not verilerinin ve derslere ait verilerin olduğu tablolardır. Bu tabloların tasarım şekli ve yapısı birliktelik kuralı üretimi esnasında önemli rol oynamaktadır. Hatalı veya eksik tasarlanmış tablolardan, istenilen bilgilerin elde edilebilmesi oldukça zordur ve çok karmaşık işlemler gerektirebilir. Bu nedenle öğrenci, not ve ders bilgileri ayrı ayrı tablolar halinde oluşturulmaktadır. Veri tabanında kayıtlı öğrenci, ders ve notlara ilişkin tabloların kayıt yapısından bir kesit Şekil 6.3'te görülmektedir.



Şekil 6.3 Öğrenci özlük, not ve ders bilgilerinin bulunduğu tabloların içeriği

6.3. Verilerin Hazırlanması

Herhangi bir modelin kurulması aşamasında ortaya çıkacak sorunlar, verilerin hazırlanması aşamasına sık sık geri dönülmesine ve verilerin yeniden düzenlenmesine neden olmaktadır. Bu durum, verilerin hazırlanması ve modelin kurulması aşamaları için %50 ile %85 oranında enerji ve zaman kaybı yaşanmasını sağlamaktadır. Bu nedenle bu aşama dikkatli ve titiz bir şekilde yapılmalıdır.

Veri hazırlama işleminin ilk aşaması verilerin toplanması aşamasıdır. Burada verilerin hangi kaynaklardan alınacağı tespit edilir. Birliktelik kuralı tabanlı öğrenci not öngörü ve başarı analiz modelinde Fırat Üniversitesi Öğrenci İşleri Daire Başkanlığı veri tabanı verileri kullanılmıştır. Ayrıca veri toplama aşamasında gerekiyorsa kuruluşun kendi veri kaynaklarının dışında farklı veritabanlarından da faydalanılabilir.

Veri hazırlama işleminde ikinci aşama değer biçme aşamasıdır. Veri madenciliğinde kullanılacak verilerin farklı kaynaklardan toplanması, doğal olarak veri uyumsuzluklarına neden olmaktadır. Bu uyumsuzluklardan en sık karşılaşılanları, verilerin farklı zamanlara ait olmaları, kodlama farklılıkları (örneğin bir veri tabanında cinsiyet özelliğinin e/k, diğer bir veri tabanında 0/1 olarak kodlanması), farklı ölçü birimleridir. Bu nedenlerle, hem öğrenci işlerinden alınan veriler hem dış kaynaklardan alınan veriler hem de birliktelik kuralı üretimi için kullanılacak veriler ile veri tabanı tablolarının birbiri ile uyumlu hale getirilmesi ve buna göre tasarlanması gerekmektedir.

Veri tabanı hazırlamanın üçüncü aşaması ise birleştirme ve temizleme işlemlerini kapsamaktadır. Bu adımda farklı kaynaklardan toplanan verilerde bulunan ve bir önceki adımda belirlenen sorunlar mümkün olduğu ölçüde giderilmekte ve verilerin tek bir veri tabanında olması sağlanmaktadır. Bu nedenle, tasarımı yapılan modelin tüm tabloları öğrenci veri tabanı içerisine dâhil edilmiştir.

Dördüncü aşama olan seçim aşamasında, kurulacak modele bağlı olarak veri seçimi yapılmaktadır. Tahmin edici bir model için, bağımlı ve bağımsız değişkenlerin ve modelin eğitiminde kullanılacak veri kümesinin seçilmesi gerekmektedir. Bu nedenle birlikteli kuralı üretebilmek için ilk önce kural üretilecek dersler ve öğrenci özlük bilgileri seçilmektedir. Şekil 6.4'te seçme işlemi uygulanmış verilerden örnek bir kesit görülmektedir. Veri tabanı tabloları üzerinde yapılacak sorgulamalar sayesinde Şekil 6.4'te görülen formata uygun veri seçimi yapmak mümkündür.

No	Matematik-I	Matematik-II	Kimya-I	Cinsiyet	…
99530001	60	93	70	1	
99530002	75	52	35	0	
…	…	…	...	…	

Şekil 6.4 Seçme işlemi uygulanmış verilerin yapısı

Öğrenci kimlik numarası, telefon numarası, dersin kredisi gibi anlamlı olmayan ve diğer değişkenlerin modeldeki ağırlığının azalmasına da neden olabilecek değişkenlerin, modele girilmemesi gerektiği için temizlenmiştir. Ayrıca öğrenci veri tabanında tüm bölümlere ait veriler bulunmaktadır. Farklı bölümlerdeki öğrenci verilerinden elde edilen birliktelik kuralları da herhangi bir anlam ifade etmeyeceğinden, seçim işleminde aynı bölüme ait öğrencilerin seçilmesi gerekmektedir.

Son aşama olan dönüştürme aşamasında, eldeki verilerin algoritma uygulanabilecek düzenli bir yapıya dönüştürülmesi gerekmektedir. Çünkü modelde kullanılan algoritma, verilerin gösteriminde önemli rol oynamaktadır. Örneğin öğrencinin her bir dersteki not değeri yerine değerlerinin yüksek/orta/düşük olarak gruplanmış olması veya A/B/C/F gibi not aralıklarına karşılık gelen değerlerin kullanılması modelin etkinliğini artıracaktır. Notların karşılık gelen harf değerlerine göre düzenlenmiş yapısı Şekil 6.5'te görülmektedir. Herhangi bir dersten alınan not 50'den küçük ise F (Kaldı), 50–64 aralığında ise C, 65–84 aralığında ise B ve 85–100 aralığında ise A olarak tanımlanmaktadır.

No	Matematik-I	Matematik-II	Kimya-I	Cinsiyet	…
99530001	C	A	B	1	
99530002	B	C	F	0	
…	…	…	...	…	

Şekil 6.5 Dönüştürme işleminin birinci aşamasında elde edilen verilerin yapısı

Şekil 6.5'te elde edilen yapı, henüz birliktelik kuralı üretme algoritmalarını uygulamaya elverişli bir yapı değildir. Çünkü farklı alanlarda farklı veri tipinde değerler mevcut olabilmektedir. Son bir dönüştürme işlemi gerçekleştirilerek verilerin algoritma uygulanabilecek yapıya dönüştürülmesi gerekmektedir. Birliktelik kuralı algoritmalarından en çok bilineni ve yaygın olarak kullanılanı Apriori algoritmasıdır. Bu modelde de birliktelik kuralı üretmek için Apriori algoritmasından faydalanılmıştır. Apriori algoritması çalışma prensibi önceki bölümlerde detaylı olarak anlatılmıştır. Eldeki veriler, Apriori algoritması uygulanabilecek yapıya dönüşebilmesi için önce Şekil 6.5'te gösterilen yapıya, daha sonrada Şekil 6.6'da gösterilen yapıya dönüştürülmesi gerekmektedir.

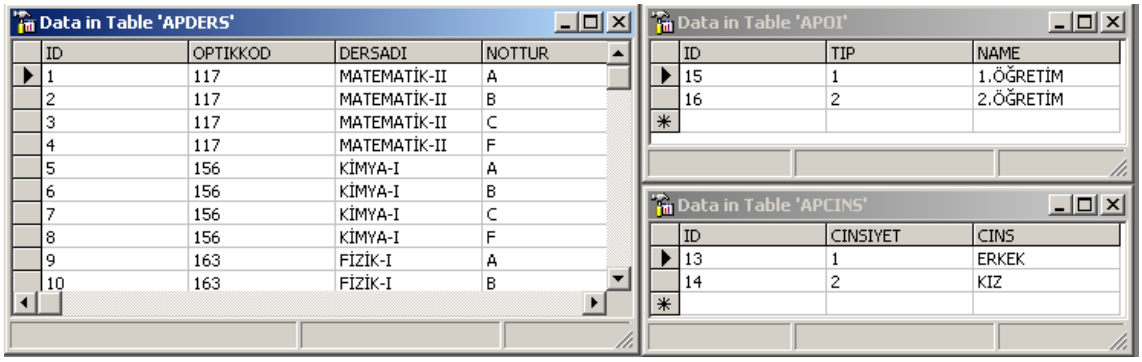
No	MatI-F	MatI-C	MatI-B	MatI-A	MatII-F	MatII-C	MatII-B	MatII-A	…
1	0	1	0	0	0	0	0	1	…
2	0	0	1	0	0	1	0	0	…
…		…	…	…		…	…	…	…

Şekil 6.6 Dönüştürme işleminin ikinci aşamasında elde edilen verilerin yapısı

		Nitelikler												
		0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**...**
İşlemler	**1**	1	0	0	0	0	0	0	1	1	0	0	0		
	2	1	0	0	0	0	1	0	0	0	1	0	0		
	3	0	1	0	0	1	0	0	0	0	0	1	0		
	4	0	1	0	0	0	0	1	0	0	0	1	0		
	5	0	0	0	1	0	1	0	0	0	1	0	0		
	6	0	0	0	1	1	0	0	0	0	0	0	1		
	:														
	:														

Şekil 6.7 Dönüştürme işlemleri sonunda elde edilen verilerin yapısı

Şekil 6.7'de görülen yapıdaki her bir satır Apriori algoritması uygulanacak veri tabanı yapısındaki işlemleri her bir sütun ise veri tabanının niteliklerini göstermektedir. Nitelik değerleri 0/1 değerlerinden oluştuğu için Apriori algoritması uygulamaya oldukça müsaittir. Herhangi bir derse ait dört nitelik değerinden sadece bir tanesi 1 değerini aldığı için nitelik sayısının artmış olması algoritmada herhangi bir yavaşlamaya sebebiyet vermemektedir. Verilerin Şekil 6.7'de gösterilen yapıya dönüştürülmesi esnasında niteliklerin anlamlarının kaybedilmemesi gerekmektedir. Bu nedenle ek tablolar kullanılarak bu değerlere ulaşılabilmektedir. Şekil 6.8'de her bir niteliğin açıklaması SQL Sunucusu ortamında tablo olarak verilmektedir. Bu sayede elde edilen kurallar anlamlı hale getirilebilmektedir.



Şekil 6.8 SQL sunucusu ortamında niteliklerin gösterimi

6.4. Modelin Kullanılması

Seçilen algoritmaya uygun olarak ilgili veriler hazırlandıktan sonra, ilk aşamada verinin bir kısmı modelden birliktelik kuralı üretimi için, diğer kısmı ise modelin geçerliliğinin test edilmesi için ayrılmaktadır. Verilerin bir kısmı kullanılarak birliktelik kuralı üretimi gerçekleştirildikten sonra, geriye kalan test kümesi verileri ile modelin doğruluk derecesi belirlenebilir. Kurulan modelin doğruluk derecesi ne kadar yüksek olursa olsun, gerçek dünyayı tam anlamı ile modellediğinin garanti edilmesi mümkün değildir. Yapılan testler sonucunda geçerli bir modelin doğru olmamasındaki başlıca nedenler, zaman içerisinde ortam parametrelerinin değişmesi, kabul edilen varsayımların ve modelde kullanılan verilerin doğru olmamasıdır. Örneğin modelin kurulması sırasında derslere giren öğretim elemanlarının değişmesi ve sınav sistemlerinde veya geçme notlarında olabilecek değişiklikler elde edilen kuralları belirgin olarak etkileyebilecektir.

6.5. Uygulama Sonuçları

Birliktelik kuralı tabanlı öğrenci not öngörü ve başarı analizi modeli kullanılarak Fırat Üniversitesi Teknik Eğitim Fakültesi Elektronik ve Bilgisayar Eğitimi Bölümü öğrencileri üzerinde iki ayrı uygulama yapılmıştır. İlk olarak öğrencilerin birinci ve ikinci sınıfta aldıkları genel kültür dersleri üzerinde bir çalışma yapılmıştır. Bu çalışmada amaç, öğrencilerin genel kültür derslerinden aldıkları notlar arasındaki ilişkileri tespit etmek ve yine bu derslerden gelecekte alabilecekleri notlar hakkında öngörü üretmektir.

Veri tabanında kayıtlı 250 öğrencinin genel kültür dersleri kullanılarak yapılan birliktelik kuralı tabanlı öğrenci not öngörü ve başarı analizi modeli uygulamasında %25 destek ve %100 güven değeri için üretilen kuralların bir kısmı Şekil 6.8'de, %50 destek ve %80 güven değeri için üretilen kuralların bir kısmı da Şekil 6.9'da görülmektedir.

```
1: tur1_B-ata1_B---->mat3_C-       destek= % 100
2: kim2_C-ata1_C---->mat4_C-       destek= % 100
3: mat1_C-fiz2_C---->mat2_C-mat3_C-destek= % 100
4: mat2_C-fiz2_C---->mat4_C-       destek= % 100
```

Şekil 6.9 %25 destek ve %100 güven değeri için elde edilen kuralların bir kısmı

```
5: mat1_C-mat2_C---->mat3_C- mat4_C-  destek= % 82
6: tur1_C-mat2_C---->mat4_C-          destek= % 83
7: mat1_B-fiz2_C---->mat2_B- mat3_C-  destek= % 83
8: mat2_C-fiz2_C---->mat3_C- mat4_C-  destek= % 83
```

Şekil 6.10 %50 destek ve %80 güven değeri için elde edilen kuralların bir kısmı

Şekil 6.9 ve Şekil 6.10'da verilen kurallar, öğrenci not ve dersleri arasındaki ilişkileri göstermektedir. Örneğin Türkçe-1 dersini B ve Atatürk İlkeleri ve İnkılap-1 dersini B notu ile geçen öğrencilerin tamamının Matematik-III dersini C notu ile geçtiği görülmektedir. Eğer elde edilen kuralların sağ tarafında bulunan dersler sol tarafta bulunan derslerden daha sonraki dönemlere ait dersler ise bu kural kullanılarak öğrencilerin gelecekte alacağı notlar ile ilgili öngörüde bulunulabilir. Bu şekilde 5. kural ele alınırsa Matematik-I ve Matematik-II dersinden C notu alan öğrencilerin %82.1 ihtimalle Matematik-II ve Matematik-IV dersinden de C notu ile geçeceği öngörüsü yapılabilmektedir.

Birliktelik kuralı üretimi esnasında sadece notlar değil, öğrencilere ait özellikler de kullanılabilmektedir. Şekil 6.11'de öğrencilerin derslerden aldıkları not ve bazı özlük bilgileri kullanılarak elde edilen kurallardan bir kısmı görülmektedir. Bu kurallar yorumlanarak öğrencilerin hem gelecek dönemde alabilecekleri notlarla ilgili öngörüler yapılabilmekte hem de ders başarılarını etkileyebilecek kriterler belirlenebilmektedir. Örneğin Şekil 6.11'de gösterilen 16. kural incelendiğinde II. Öğretim öğrencilerinden ELT101 kodlu dersi C notu alarak geçenlerin %86 ihtimalle ELT102 kodlu dersten de C notu alarak geçeceği görülmektedir.

```
 9: ERKEK  --> KÜT. ÜYE DEĞİL                                  destek = % 85
10: FIZ108 C -MAT168 B --> KÜT. ÜYE DEĞİL                      destek = % 86
11: TDE102 C -FIZ108 C -MAT168 C -ERKEK -->KÜT. ÜYE DEĞ.       destek = % 98
12: BIL262 F -BIL364 F--> BIL381 F                             destek = % 94
13: BIL364 B -BIL382 C -ELT327 C--> BIL444 B - BIL364 C        destek = %100
14: BIL444 B -BIL493 C -BIL363 C--> ELT327 C                   destek = % 94
15: II. ÖĞRETİM -->ELT102 C                                    destek = % 75
16: ELT101 C –II. ÖĞRETİM --> ELT102 C                         destek = % 86
```

Şekil 6.11 %25 destek ve %75 güven değeri için elde edilen kuralların bir kısmı

Birliktelik kuralı tabanlı öğrenci not öngörü ve başarı analizi modelinden elde edilen sonuçlar kadar bu sonuçların geçerliliği de oldukça önemlidir. Elde dilen kuralların ve kurallara göre yapılan öngörülerin ne derece doğru olduğunun tespit edilmesi amacıyla bu sonuçların öğrenciler üzerinde test edilmesi gerekmektedir. Yukarıdaki şekillerde gösterilen 16 farklı kural için toplam 100 öğrenci üzerinde test yapılmış ve elde edilen başarım yüzdesi hesaplanmıştır. Her bir kuralın güven değeri, test sonucu elde edilen yüzdesi ve başarım değeri Tablo 6.1'de verilmektedir.

Tablo 6.1 Birliktelik kuralı tabanlı öğrenci not tahmin ve başarı analiz modelinden elde edilen kuralların test sonuçları ve başarım değerleri

Kural No	**Güven Değeri (%)**	**Test Sonucu (%)**	**Başarım (%)**
1	100	96	96.0
2	100	95	95.0
3	100	92	92.0
4	100	96	96.0
5	82	80	97.6
6	83	85	97.6
7	83	77	92.8
8	83	82	98.8
9	85	88	96.5
10	86	80	93.0
11	98	96	98.0
12	94	92	97.9
13	100	94	94.0
14	94	88	93.6
15	75	72	96.0
16	86	80	93.0
Ortalama Başarım (%)			**95.5**

Tablo 6.1'de görüldüğü gibi elde edilen tüm kuralların öğrenciler üzerinde test edilmesi sonucunda oldukça başarılı olduğu görülmektedir. Elde edilen 16 kural ele alındığında ortalama olarak %95,5 başarım elde edilmiştir. Bu da kuralların oldukça geçerli ve başarılı olduğunu göstermektedir.

6.6. Uygulama Yazılımı

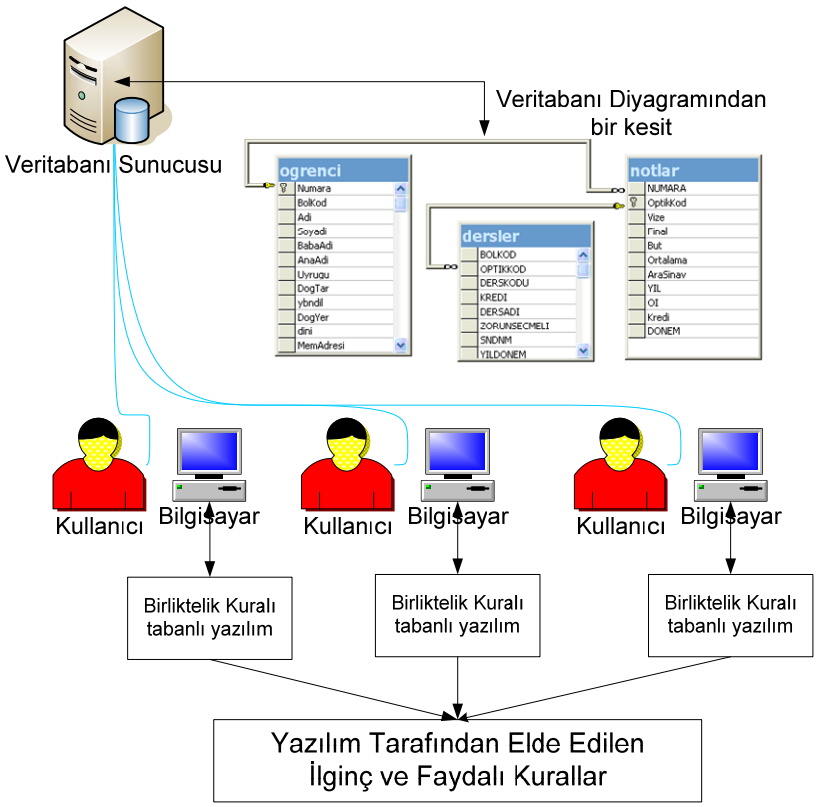
6.6.1. Uygulama Yazılımı İçin Kullanılan Araçlar

Birliktelik kuralı tabanlı öğrenci not tahmin ve başarı analizi modelini kullanabilmek ve sonuçlar üretebilmek için Borland C++ Builder 6.0 ara yüzü kullanan bir yazılım hazırlanmıştır. Borland C++ Builder programı, grafiksel ara yüzü aracılığıyla programcıya hızlı bir şekilde program tasarımı ve yazımı sunmaktadır. Borland C++ Builder gibi görsel programlama dillerinde programın ekran yapısı komutlarla değil, hazır formlarla tasarlanmaktadır. Bu ekran tasarımı sayesinde programın kullanımı esnasında kullanıcıya büyük kolaylıklar sağlanmaktadır. C++ dili, komut yapısı bakımından birçok kütüphaneyi programcının hizmetine sunmaktadır. Bu dilin en büyük avantajlarından birisi, veri tabanı uygulamalarıyla yüksek performans ve yüksek seviyede iletişim sağlanmasıdır. Mevcut bir veri tabanından bilgi okumak, bilgi eklemek ve değiştirmek Borland C++ Builder'in bünyesinde bulunan veri tabanı bileşenleri sayesinde oldukça kolaydır.

Yazılım sisteminin veri tabanının oluşturulması aşamasında MS-SQL Veri tabanı yönetim sistemi yazılımı kullanılmıştır. Zaten uygulama için kullanılan veri tabanı olan Fırat Üniversitesine ait öğrenci verileri de MS-SQL ortamında bulunmaktadır. MS-SQL sorgu dili sayesinde veri tabanında yapılmak istenen bütün işlemler kolaylıkla yapılabilmektedir. Yüksek kapasiteli veri tabanlarının saklanması ve veri tabanları üzerinde normalizasyon işlemine imkân vermesinden dolayı bu dil yardımıyla veri tabanı tablolarını tasarlamak oldukça kolaydır. Ayrıca görsel programlama dilleriyle kolay iletişimi de programcıya sunduğu en büyük avantajlardan birisidir.

6.6.2. Uygulama Yazılımı Ortam Yapısı

Oluşturulan yazılım sisteminin ortam yapısı Şekil 6.12'de görülmektedir. Şekilden görüldüğü gibi tüm kullanıcılar tek bir veri tabanı sunucusu üzerinden iletişim kurmaktadır. Bu sayede klasik dosya sistemlerinin dezavantajlarından olan çoklu güncelleme, veri tekrarı ve bellek hacmi israfı gibi birçok dezavantajlar engellenmiştir.

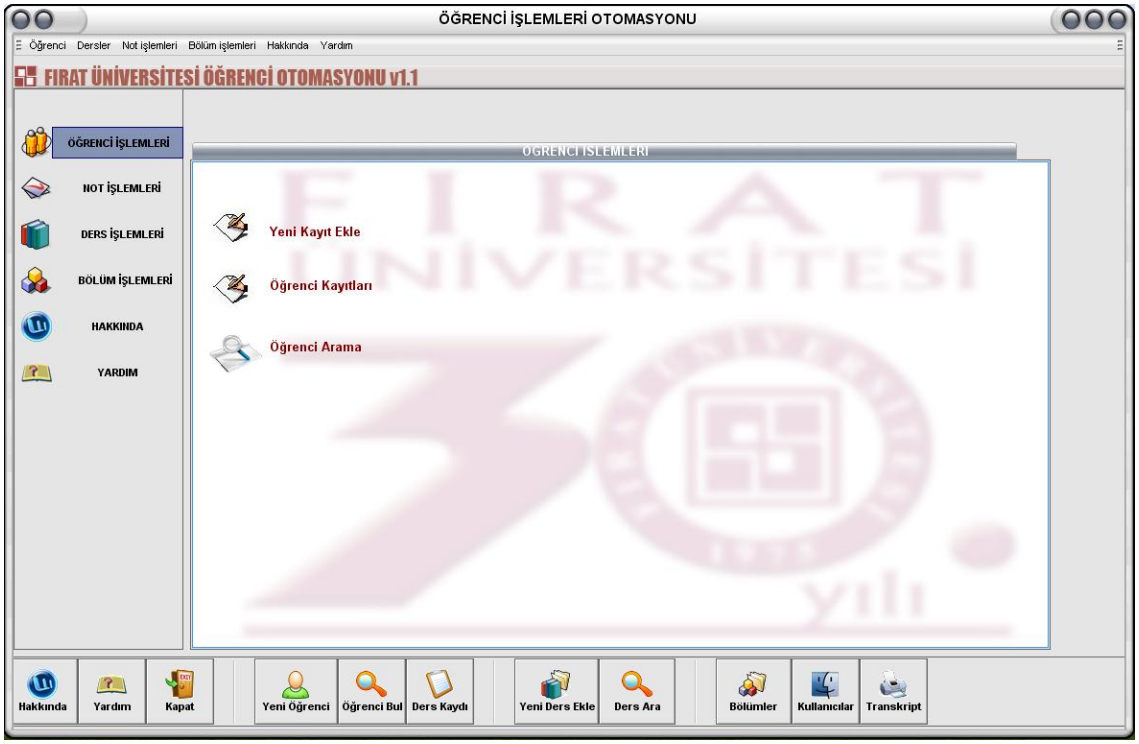


Şekil 6.12 Yazılım ortamı yapısı

Yazılımda kullanılan veri tabanı, Fırat Üniversitesi Öğrenci İşleri Otomasyonunda kullanılan veri tabanı ile aynı yapıya sahiptir. Bu nedenle, hazırlanan yazılım Fırat Üniversitesi'nin herhangi bir biriminde kullanılmaya oldukça elverişlidir. Yazılımı kullanan kullanıcılar, geliştirilen yazılım aracılığı ile gerekli verilere yetkileri dâhilinde ulaşabilmekte ve gerekli işlemleri yapabilmektedir. Ayrıca, yazılımın en önemli bölümünü oluşturan birliktelik kuralı üretim modülünü kullanıp buradan kural taraması yapabilmektedirler.

6.6.3. Uygulama Yazılımı Ara Yüzü

Birliktelik kuralı tabanlı öğrenci not tahmin ve başarı analizi amaçlı hazırlanan yazılım, normal bir öğrenci otomasyon sisteminin gereksinimlerini karşılamakla birlikte, içerdiği ek bir modül sayesinde öğrencilerin özlük bilgileri, not bilgileri gibi verileri süzerek öğrenci başarısına etki eden ve etki edebilecek birçok gizli kuralların çıkarımını yapmaktadır. Bu kurallar sayesinde öğrencinin hangi dersten hangi notu alabileceğinin tahmini de yapılabilmektedir. Şekil 6.13'te yazılımın çalıştırılması ile karşımıza çıkacak ana ekran görüntüsü görülmektedir. Yazılımda, öğrenci işlemleri, not işlemleri, ders işlemleri ve bölüm ile ilgili işlemler yer almaktadır.



Şekil 6.13 Yazılımın ana penceresi

6.6.3.1. Öğrenci İşlemleri

Öğrenci işlemleri menüsünde yeni bir öğrencinin kaydını yapabilmek için *Yeni Kayıt Ekle,* veri tabanındaki tüm kayıtları görebilmek için *Öğrenci Kayıtları* ve herhangi bir öğrenciye ait tüm bilgileri ekrana getirebilmek için *Öğrenci Arama* kısımları bulunmaktadır. Öğrenci işlemleri menüsünde yapılabilecek işlemler ve ara yüz penceresi Şekil 6.14'te görülmektedir.

Şekil 6.14 Yazılımın öğrenci işlemleri penceresi

Öğrenci kayıt menüsü, yeni bir öğrenci kaydı yapmak için kullanılan bölümdür. Öğrenci tablosunda, öğrenciler ile ilgili birçok verinin bulunması ve bunların tek tek girilmesi gerektiği düşünüldüğünde yeni bir öğrenci kaydının yapılması esnasında birçok görsel karmaşıklık

olacağı bilinmektedir. Bu yüzden yeni kayıt ekleme kısımda, verilerin düzenli ve rahat bir şekilde girilmesi için okul bilgileri, kimlik bilgileri, adres bilgileri ve diğer bilgiler şeklinde bölümler oluşturulmuş ve kayıt girişi karmaşıklığı giderilmiştir.

Öğrenci kayıtları bölümünde, kayıtlı öğrenci bilgilerinin görüntülenmesi mümkün olabilmektedir. Bu kısımda yeni kayıt yapılamayıp sadece var olan kayıtlar görüntülenebilmekte ve var olan kayıtlar üzerinde güncellemeler yapılabilmektedir.

Veri tabanında kayıtlı herhangi bir öğrencinin bilgilerinin sorgulanması için kullanılan bölüm ise *Öğrenci Arama* bölümüdür. Bu kısımda istenilen kaydın kolayca bulunabilmesi için belirli kriterler eklenerek öğrenci bulma işlemi kolaylaştırılmıştır. Sadece öğrencinin numarası girilerek sorgulama yapılabildiği gibi öğrencinin adına, soyadına veya bölümüne göre de arama yapılabilmektedir.

6.6.3.2. Not İşlemleri

Not işlemleri bölümü, herhangi bir ders için öğrencilerin notlarını veri tabanına girebilmek ve öğrencilere ait not bilgilerini görüntülemek için kullanılan menüdür. Hazırlanan yazılım, üniversiteye yönelik olduğundan bu aşamada öğrencilerin vize, final ve bütünleme notları veri tabanına işlenmektedir. Not işlemleri menüsünde yapılabilecek işlemler ve ara yüz penceresi Şekil 6.15'te görülmektedir.

Şekil 6.15 Not işlemleri penceresi

Not Girişi kısımda ders adı ve bölüm seçildikten sonra seçilen ders için notları girilmesi gereken öğrenciler ekrana gelmekte ve öğretim elemanı bu dersin notlarını programdan toplu olarak işleyebilmektedir. *Öğrenciye Not Ver* bölümünde ise not girişi bölümünden farklı olarak seçilen herhangi bir öğrenciye belirtilen dersten not girişi yapılabilmesi sağlanmaktadır. Vize, final, bütünleme alanlarına gerekli notlar girildikten sonra *Notları İşle* butonu aktif hale gelecektir ve girilen bu notlar veri tabanına işlenecektir. Ayrıca, istenildiği takdirde herhangi bir öğrenciye ait tüm not bilgilerine *Tüm Notlara Bak* bölümünden ulaşılabilmektedir.

6.6.3.3. Ders İşlemleri

Ders işlemleri menüsü, üniversitenin bölümlerinde verilmekte olan derslerin düzenlenmesi, kayıtlı olan derslerin sorgulanması, her dönem başlangıcında öğrencilerin ders kayıtlarının veri tabanına işlenmesi ve danışman öğretmenleri tarafından onaylanmasını sağlayan kısımdır. Bu kısım *Yeni Ders Ekleme*, *Ders Bul*, *Ders Kaydı İşlemleri* ve, *Ders Kaydı Güncelle* bölümlerinden oluşmaktadır. Bu bölüme ait ana menü Şekil 6.16'da görülmektedir

Şekil 6.16 Ders işlemleri ana menüsü

Yeni ders ekleme bölümü, veri tabanında bulunan dersler tablosuna yeni bir ders girişi yapılmasını sağlayan bölümdür. Bu kısımda, bölüm kodu alanı yeni dersin hangi bölüme ait olduğunu, optik kodu alanı dersin veri tabanındaki ayırt edici kimlik bilgisini temsil etmektedir. Veri tabanındaki kayıtlı dersler içerisinden, istenilen kriterlere uygun olan herhangi bir dersin veya birkaç dersin aranmasını sağlayan bölüm ise Ders Bul bölümüdür.

Ders kayıt işlemleri öğrencilerin her dönem içerisinde aldığı derslerin seçilmesi için kullanılan bir bölümdür. Öğrenci numarası ve dönem bilgisi girildiğinde öğrencinin alması gereken dersler ve alabileceği dersler otomatik olarak ekranda görüntülenmektedir. İstenilen dersler seçilerek kaydedildiğinde öğrenciler belirtilen dönem için seçilen dersleri almış olacaklardır.

6.6.3.4. Bölüm İşlemleri

Bölüm işlemleri kısmı, bölüm bazında yapılan işlemleri kapsamakla birlikte birliktelik kuralı tabanlı öğrenci not tahmin ve başarı analizi modelini de içeren menüdür. Bu kısımdan, öğrenci not ve özlük bilgileri arasından birliktelik kuralları oluşturulabilmektedir. Ayrıca bu kısımda, bölümdeki genel işlemler de gerçekleştirilebilmektedir. Öğrencilere danışman öğretim

elemanı atama, öğrenci şifreleri düzenleme ve ders aktif ve pasif işlemlerine de bu kısımdan ulaşılmaktadır. Bu bölüme ait ana menü Şekil 6.17'de görülmektedir

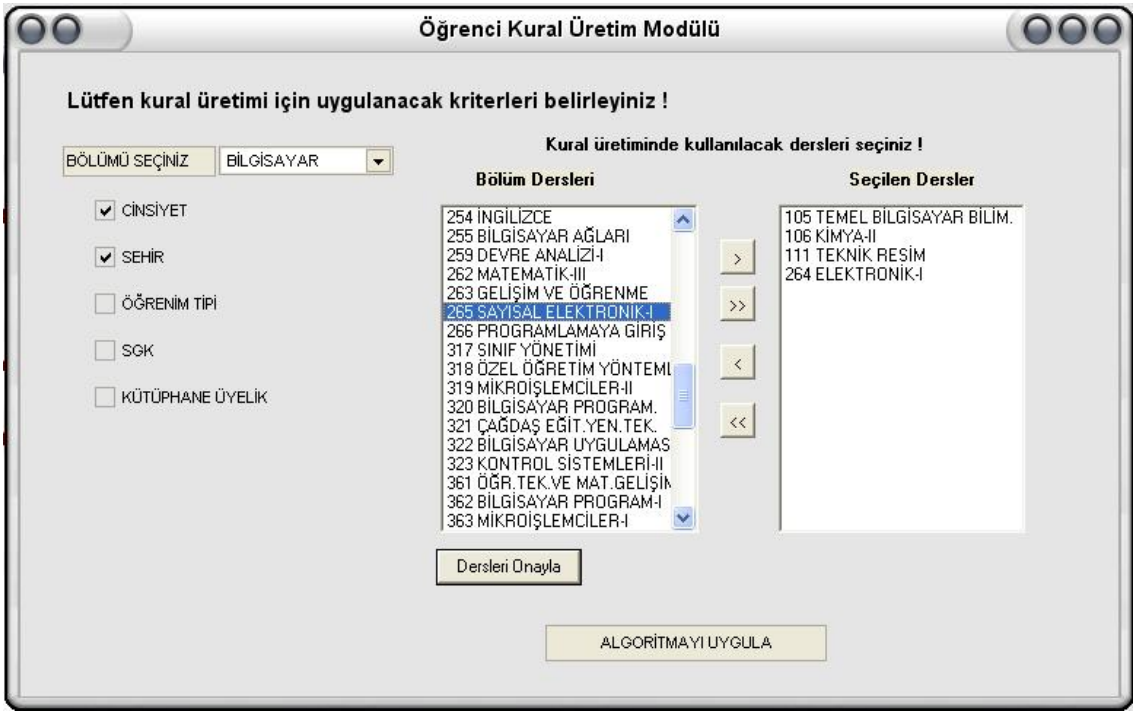
Şekil 6.17 Öğrenci ders kayıt işlemleri penceresi

Öğrenci kural üretim modülü, yazılımın en önemli özelliğini içeren bölümdür. Bu bölüm, çeşitli kriterleri kullanarak mevcut veri tabanından öğrenciler ile ilgili saklı ilişkilerin ortaya çıkarılmasını, saklanmasını ve bu ilişkilerin kullanılmasını sağlamaktadır. Eldeki verilerden elde edilen kurallar sayesinde yazılımın, öğrencilerin gelecekte alacağı notları tahmin edebilmesi sağlanmaktadır. Bu kısımda, birliktelik kuralı algoritmalarından biri olan ve en yaygın kullanıma sahip olan Apriori algoritması kullanılmış ve seçilen veriler arasındaki birliktelikler elde edilmiştir.

Rehberlik hizmetlerinin ön plana çıktığı eğitim kurumlarında öğrenciler ile ilgili veriler rehber öğretmenler tarafından incelenerek çeşitli kanılara varılmaktadır. Aynı şekilde rehberlik hizmetleri çerçevesinde yapılan anketler uzmanlara öğrenciler hakkında çeşitli bilgiler vermektedir. Fakat binlerce kaydın bulunduğu veri tabanlarında öğrencilerin başarısına etki eden etmenlerin ilişkilerini bulmak oldukça zor bir iştir. Ancak, hazırlanan bilgisayar programları yardımıyla bu işlemler oldukça kolay ve hızlı bir şekilde gerçekleştirilebilmektedir. Bu işlemleri yazılımın *Öğrenci Kural Üretim Modülü* gerçekleştirmektedir.

Veri tabanındaki saklı ilişkilerin bulunması için yapılacak ilk işlem, aralarındaki ilişkilerin bulunacağı kriterlerin belirlenmesidir. Bu aşamada önemli olan hangi kriterlerin kullanılacağının tespitidir. Şekil 6.18'de programda birliktelik kuralı uygulanacak kriterlerin belirlendiği pencere görülmektedir. Şekil 6.18'den de görüldüğü gibi ilk önce bölüm seçimi yapılmaktadır. Çünkü üretilecek kurallar her bölüm için ayrı olmalıdır. Bölüm seçildiği anda o bölüme ait ders listesi otomatik olarak listelenmektedir. Bu listeden aralarında ilişki oluşturabileceği düşünülen istenilen dersler seçilebilmektedir.

Yine bu bölümde sadece notlar arasındaki ilişkiler değil öğrencilerin bazı özlük bilgileri de kullanılmaktadır. Bunun için veri tabanında öğrencilere ait özlük bilgilerinden seçilen beş kriter programda kullanılmıştır. Bunlar; öğrenci cinsiyeti, öğrencinin memleketi, öğrencinin öğrenim şekli, aile bilgisine bağlı olarak öğrencinin sağlık güvenlik kuruluşu (SGK) ve kütüphane üyeliğinin olup olmadığı verileridir. Bu veriler arasından istenilenler kolaylıkla seçilebilmektedir.



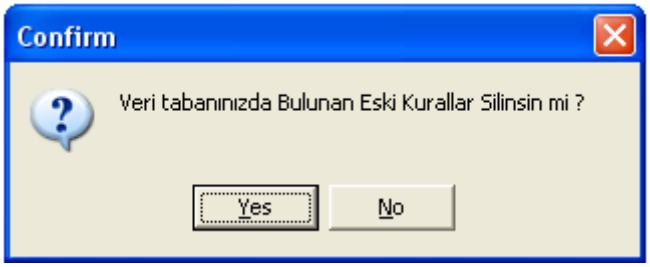
Şekil 6.18 Öğrenci kural üretim modülü penceresi

Algoritmanın uygulanacağı özlük bilgisi kriterleri ve dersler seçildikten sonra *Algoritmayı Uygula* düğmesi tıklanarak seçim işlemleri tamamlanmış olmaktadır. Bu aşamadan sonra programda kullanılan Apriori algoritmasına ait Destek ve Güven değerlerinin belirlenmesi aşaması gelmektedir. Bu pencere Şekil 6.19'da görülmektedir.

Şekil 6.19 Destek ve güven değerlerinin belirlendiği pencere

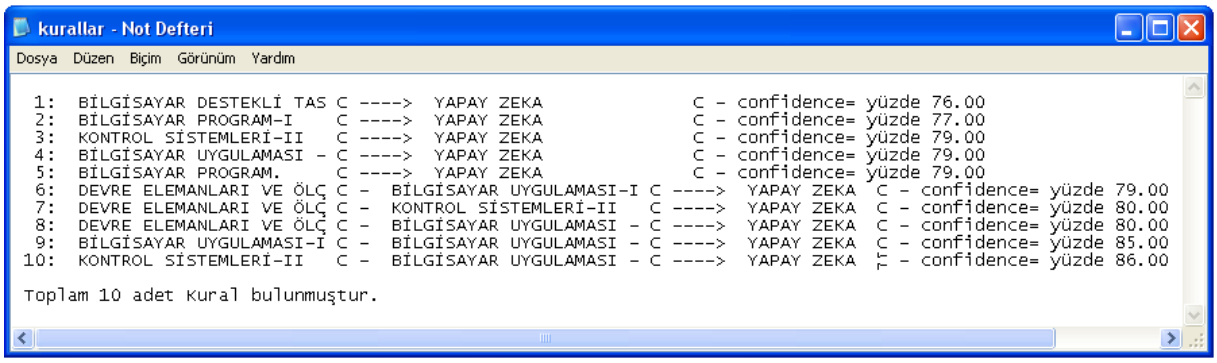
Şekil 6.19'da görüldüğü gibi karşımıza çıkan pencerede *Destek Değeri* ve *Güven Değeri* olmak üzere iki adet parametre giriş değerinin ayarlanabileceği görülmektedir. Bu parametreler birliktelik kuralı üretiminde kullanılan çok önemli parametreler olup kural üretilmesinde birinci dereceden rol oynamaktadır. Destek değeri, üretilen kuralların ne kadar sıklıkla gerçekleştiğini belirtmektedir. Bu değer ne kadar yüksek olursa, üretilen kurallar da o kadar geçerli olacağı denilebilir. Ancak destek değerinin olması gerekenden oldukça yüksek bir değere sahip olması durumunda ise kural üretimi gerçekleşmeyebilir. Bu nedenle destek değerinin çok iyi bir şekilde tercih edilmesi gerekmektedir. Bu seçim aşamasında uzman kişini önemi oldukça artmaktadır.

Yazılımdan elde edilen kurallar bir metin dosyasına da kaydedilebilmektedir. Destek değeri ve güven değerini ayarladıktan sonra kayıt yapılacak dosya adı da seçilerek *Çalıştır* düğmesi ile uygulama başlatılabilir. Program çalışmaya başlamadan önce Şekil 6.20'deki gibi bir uyarı verecektir. Program bu kısımda daha önceden oluşturulmuş kuralların sistemde tutulup tutulmayacağı ile ilgili onay istemektedir. *Hayır* düğmesi seçildiğinde, her defasında elde edilen yeni kurallar veri tabanında daha önceden elde edilmiş olan kurallara eklenerek çok geniş bir kural kümesi elde edilebilmektedir.



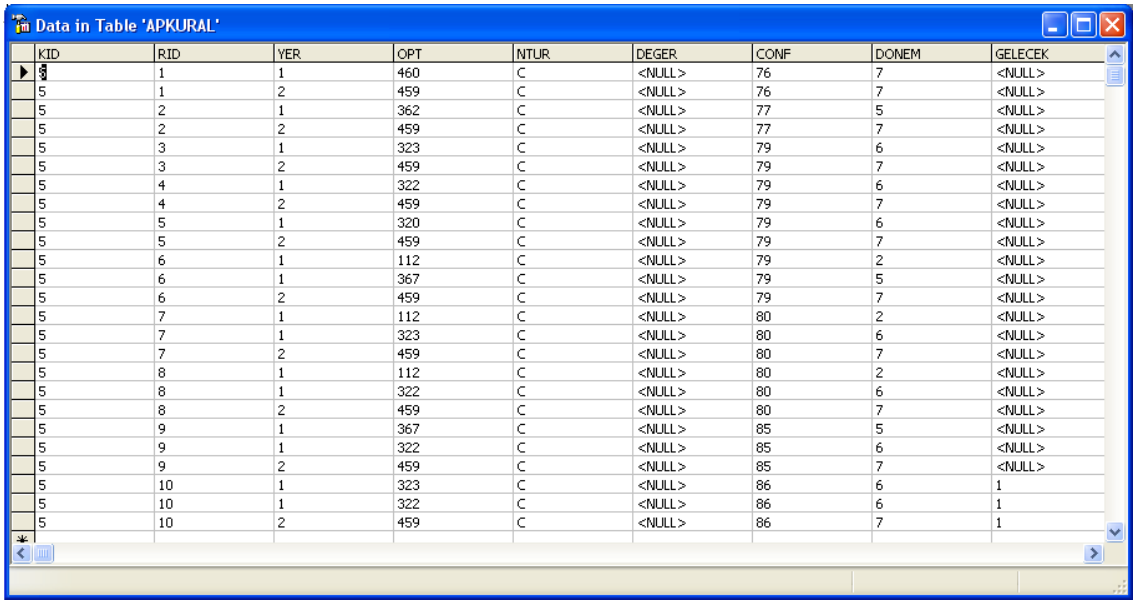
Şekil 6.20 Eski kayıtların silinmesi ile ilgi uyarı penceresi

Elde dilen kurallar, veri tabanında *APKURAL* isimli bir tabloda tutulmaktadır. Her defasında farklı ders ve kriterlerle bulunan kuralların saklanması ile birlikte kurallar kümesi genişleyeceğinden hemen hemen her öğrenciye uyacak ve öğrenci hakkında tahmini bilgiler verecek sonuçlara ulaşmak mümkün olabilecektir. Gelen uyarı penceresinde *Eski kurallar silinsin mi?* sorusuna *Evet* cevabı verildiğinde ise veri tabanında daha önce kaydedilen bütün kurallar silinerek ve sadece yeni elde edilen kurallar veri tabanına kaydedilecektir. Algoritmanın çalışması tamamlandıktan sonra ekrana bir ileti gelerek kuralların üretildiği ve bir metin dosyasına kaydedildiği belirtilmektedir. Şekil 6.21'de elde edilen kuralların kullanıcı tarafından belirlenen bir metin dosyasına kaydedilmiş hali ve içeriği görülmektedir.



Şekil 6.21 Elde edilen kuralların yazıldığı metin dosyası

Şekil 6.21'den de görüldüğü gibi metin dosyası, seçilen derslerden öğrencilerin aldığı notlar arasındaki ilişkileri göstermektedir. Bu kurallar öğrencilerin hangi derslerden hangi notu alacağını yüzde olarak tahmin edebilmektedir. Örneğin birinci kural incelenecek olursa, Bilgisayar destekli tasarım dersinden C notu alan öğrencilerin yüzde 76 ihtimalle Yapay Zeka dersinden de C notu alacağı sonucu ortaya çıkmaktadır. Ayrıca Bu kuralların veri tabanında *APKURAL* tablosuna kaydedilmiş hali de Şekil 6.22'de görülmektedir.



Data in Table 'APKURAL'

KID	RID	YER	OPT	NTUR	DEGER	CONF	DONEM	GELECEK
5	1	1	460	C	<NULL>	76	7	<NULL>
5	1	2	459	C	<NULL>	76	7	<NULL>
5	2	1	362	C	<NULL>	77	5	<NULL>
5	2	2	459	C	<NULL>	77	7	<NULL>
5	3	1	323	C	<NULL>	79	6	<NULL>
5	3	2	459	C	<NULL>	79	7	<NULL>
5	4	1	322	C	<NULL>	79	6	<NULL>
5	4	2	459	C	<NULL>	79	7	<NULL>
5	5	1	320	C	<NULL>	79	6	<NULL>
5	5	2	459	C	<NULL>	79	7	<NULL>
5	6	1	112	C	<NULL>	79	2	<NULL>
5	6	1	367	C	<NULL>	79	5	<NULL>
5	6	2	459	C	<NULL>	79	7	<NULL>
5	7	1	112	C	<NULL>	80	2	<NULL>
5	7	1	323	C	<NULL>	80	6	<NULL>
5	7	2	459	C	<NULL>	80	7	<NULL>
5	8	1	112	C	<NULL>	80	2	<NULL>
5	8	1	322	C	<NULL>	80	6	<NULL>
5	8	2	459	C	<NULL>	80	7	<NULL>
5	9	1	367	C	<NULL>	85	5	<NULL>
5	9	1	322	C	<NULL>	85	6	<NULL>
5	9	2	459	C	<NULL>	85	7	<NULL>
5	10	1	323	C	<NULL>	86	6	1
5	10	1	322	C	<NULL>	86	6	1
5	10	2	459	C	<NULL>	86	7	1

Şekil 6.22 Elde edilen kuralların yazıldığı *APKURAL* tablosu

Hazırlanan yazılım, sadece dersler ve belirlenen bazı özlük kriterleri arasındaki saklı ilişkileri bulmayıp bu kuralları öğrenciler üzerinde uygulayıp öğrencilerin gelecekte alacağı notlar ile ilgili tahminler de yapmaktadır. Yani bir öğrenci için hangi kuralların uygun olduğu program tarafından otomatik olarak bulunmakta ve sonuçlar görüntülenebilmektedir. Bu sonuçlara öğrenci işlemleri menüsü altındaki *Öğrenci Kayıtları* veya *Öğrenci Arama* seçeneklerinden ulaşılabilmektedir. Öğrenci bilgileri penceresindeki *Diğer Bilgiler* bölümü bu amaç için hazırlanmıştır. Şekil 6.23'te veri tabanındaki kuralların herhangi bir öğrenci üzerinde çalıştırılması sonucu karşılaşılan sonuçların bulunduğu pencere görülmektedir.

Şekil 6.23 Öğrencilere ait not tahmin penceresi

6.7. Sonuç

Bu bölümde, birliktelik kuralı tabanlı öğrenci not tahmin ve başarı analiz modeli ile bu modeli kullanan bir yazılım geliştirilmiştir. Bu model sayesinde hem derslerden alınan notlar arasında hem de dersler ile öğrenci özlük bilgileri arasında bulunan ilişkiler ortaya çıkarılabilmekte ve bu ilişkiler sayesinde geleceğe yönelik tahminler yapılabilmektedir. Geliştirilen yazılım sayesinde ise hem öğrenci kayıtları, ders kayıtları, öğrenci not girişleri gibi genel işlemler hem de istenen özlük bilgileri ve not bilgileri kullanılarak öğrencilerle ilgili not tahminleri kolaylıkla ve görsel bir şekilde yapılabilmektedir.

Öncelikle, birliktelik kuralı tabanlı not tahmin ve başarı analiz modelinin genel yapısı incelenmiş ve veri tabanı üzerinde nasıl uygulanacağı bu bölümde detaylı olarak anlatılmıştır. Modelden elde edilen sonuçların değerlendirilmesi ve tutarlığının test edilmesi amacı ile de Fırat Üniversitesi Elektronik ve Bilgisayar öğrencileri verileri üzerinde uygulama yapılmıştır. Elde edilen sonuçlar, modelin başarılı olduğunu ve kullanılabileceğini açık bir şekilde göstermektedir.

Modelin kullanılabilirliği tespit edildikten sonra kullanım kolaylığı sağlamak ve işlemleri görselleştirmek amacı ile C++ Builder programlama dili kullanılarak bir yazılım gerçekleştirilmiştir. Yazılımda veri tabanı olarak SQL Sunucusu veri tabanı yönetim sistemi kullanılmıştır. C++ Builder programına eklenen bileşenler sayesinde yazılım oldukça görsel bir hale getirilmiştir. Yazılımda kullanılan veri tabanı, Fırat Üniversitesi öğrencilerine ait olduğu için Fırat Üniversitesi öğrenci işlerinde kullanılan veri tabanı yönetim sistemi ve veri tabanı tabloları temel alınarak yazılım geliştirilmiştir.

Hazırlanan yazılımın bir bölümü olan öğrenci kural üretim modülü, yazılımın en önemli bileşenini teşkil etmektedir. Çünkü yazılım hazırlanırken hedeflenen temel amaç, öğrencilerin gelecekte alacakları notlarla ilgili tahminler üretebilmektir. Bu amaca ulaşmak için hazırlanan öğrenci kural üretim modülü sayesinde hem öğrencilerin gelecekte alacağı notlarla ilgili tahminler üretilebilmekte hem de öğrencilere ait bazı özlük bilgileri de kullanılarak ilginç olabilecek güçlü kurallar üretilebilmektedir. Öğrenci kural üretim modülünün temeli, birliktelik kuralı üretimine dayanmaktadır. Birliktelik kuralı üretim algoritmalarından biri olan Apriori algoritması, öğrenci veri tabanı üzerinde çalışabilecek duruma getirilerek kuralların üretimi gerçekleştirilmiştir. Kaynak [66-69]'da yapılan çalışmalar, öğrenci kural üretim modülü ile elde edilecek kuralların ve sonuçların, öğrenci başarısını tespit etmede ve öğrenciye yönelik etkili bir rehberlik hizmeti sunmada avantajlar sağlayabileceğini ve öğrencinin gelecekte alacağı notlarla ilgili öngörülerde bulunulabileceğini açık bir şekilde göstermektedir.

Hazırlanan yazılım sadece öğrenci işlemleri için hazırlanmasına rağmen benzer yapıda hazırlanacak yazımlar sayesinde farklı alanlarda da uygulanabileceği düşünülmektedir. Bu bölümde birliktelik kuralının hem notlara hem de öğrenci özlük bilgilerine uygulanması, farklı bilgi alanları üzerine de uygulanabileceğini göstermektedir. Bu sayede, hazırlanacak yazılımların yapısına birliktelik kuralı üretim algoritmaları eklenerek birçok alanda kullanılabilmesi mümkündür.

7. SONUÇ VE DEĞERLENDİRME

Bu tez çalışmasında, öncelikle veri madenciliği kavramına değinilmiş ve veri madenciliği tekniklerinden biri olan birliktelik kuralı tekniği detaylı olarak irdelenmiştir. Birliktelik kuralı, oldukça yaygın kullanıma sahip olmakla beraber birliktelik kuralı üreten birçok algoritma da literatürde geliştirilmiştir. Bu tezde de birliktelik kuralının farklı alanlarda kullanımına değinilmiş ve üç farklı alanda çalışma yapılmıştır. Bu üç alanda da yapılan çalışmalar birliktelik kuralının etkili ve kullanılabilir olduğunu göstermiştir. Bölüm 4'te birliktelik kuralı, özellik seçimi yapmak için kullanılmış ve farklı iki veritabanı üzerinde yapılan test sonuçları başarılı sonuçlar vermiştir. Bölüm 5'te birliktelik kuralı kullanarak yapılan doku sınıflandırma işlemlerine, hız ve başarım kazandırmak açısından literatürde bulunan diğer algoritmalar eklenmiştir. Birliktelik kuralı üretimi esnasında, doku üzerinden elde edilen nitelik sayısını ve veritabanı boyutunu azaltmak için kenar çıkarma algoritması, doku görüntüsünün detaylarına inip daha yüksek başarım elde edebilmek için de dalgacık dönüşümü tekniği kullanılmıştır. Buradan elde edilen sonuçlardan da yüksek başarımlar elde edilmiştir. Son olarak tezde, birliktelik kuralı öğrenci verileri üzerinde uygulanmıştır. Birliktelik kuralının özelliklerinden faydalanarak, öğrenciler hakkında değerli bilgiler edilebileceği ve hatta geleceğe yönelik not tahminleri yapılabileceği elde edilen sonuçlardan görülmektedir. Bunun sonucu olarak ta birliktelik kuralı tabanlı bir yazılım geliştirilmiştir. Yazılım, Fırat Üniversitesi öğrenci işleri otomasyonuna ait veritabanını kullanmakta ve istenilen bölüme ait öğrenciler ile ilgili kurallar üretebilmekte ve gelecekle ilgili not tahminleri yapabilmektedir.

7.1. Sonuçların Değerlendirilmesi

1. Literatürde, özellik seçimi yapabilmek için birçok algoritma bulunmaktadır. Ancak önerilen birliktelik kuralı tabanlı özellik seçimi yöntemi de başarılı sonuçlar vermiştir. Farklı iki veritabanı üzerinde yapılan testler sonucunda, önerilen yöntemlerin kaynak [45-51]'de belirtilen yöntemlerden daha iyi sınıflama başarım sonuçları verdiği görülmüştür.
2. Birliktelik kuralı kullanarak özellik seçimi yaparken iki farklı yöntem önerilmiştir. Bunlardan biri üretilen birliktelik kurallarından elde edilen sonuçlara, bir diğeri ise birliktelik kuralı üretilmeden sadece yoğun nesne kümelerinden elde edilen sonuçlara dayalı özellik seçimi yöntemleridir. Her iki yöntemden de başarılı sonuçlar elde edilmektedir. Ancak veritabanının yapısına göre bu yöntemlerden herhangi biri sonuç

üretemeyebilmektedir. Yinede birliktelik kuralı algoritmasındaki destek ve güven değerlerinin uygun seçilmesi ile tatmin edici sonuçlar elde etmek mümkündür.

3. Literatürde bulunan birliktelik kuralı tabanlı doku sınıflama yöntemi ile başarılı bir sınıflama yapmak mümkündür. Ancak doku görüntülerinin çeşitlerine ve kullanım yerlerine göre bazen hız veya başarım artışı istenebilir. Bu durumda birliktelik kuralı üretirken bazı ihtiyaçlar ortaya çıkmaktadır. Birliktelik kuralı üretimine hız kazandırabilmek için birkaç alternatif söz konusudur. Nitelik sayısını azaltmak, veritabanı boyutunu azaltmak ve destek değerini yüksek tutmak bu alternatiflerin başında gelmektedir. Doku görüntüleri üzerinde birliktelik kuralı uygulamadan önce kullanılan kenar çıkarma algoritması ile bu üç alternatiften de faydalanılmış olunduğundan, başarım azalmadan hızlı ve etkili bir şekilde doku sınıflandırma gerçekleştirilmiştir.
4. Doku görüntülerinin daha iyi sınıflandırılabilmesi için, elde edilen birliktelik kurallarının etkili ve belirleyici olması gerekmektedir. Bu nedenle de üretilen kuralların yüksek güven değerine sahip olmaları gerekmektedir. Bu durumu sağlayabilmek için ise doku görüntülerine önce dalgacık dönüşümü kullanarak detay görüntüleri elde edilip daha sonra birliktelik kuralı üretimi gerçekleştirilmiştir. Bu işlemde, üç farklı detay görüntüsü elde edilmesinden dolayı ekstra bir yük getireceği düşünülse bile detay görüntülerin boyutlarının daha küçük olması ve nitelik sayısının az olması, çok fazla hız kaybı yaşanmadan daha yüksek başarım elde edilmesini sağlamaktadır.
5. Önerilen iki farklı doku sınıflama yönteminin, kaynak [64,65]'te önerilen yöntemlerle karşılaştırılması sonucu kenar çıkarma yönteminin birliktelik kuralı ile kullanılmasıyla büyük ölçüde hız artışı sağlandığı, dalgacık dönüşümü yönteminin ise birliktelik kuralı ile beraber kullanımıyla daha iyi sınıflama başarımı sağladığı görülmüştür.
6. Doku sınıflama için önerilen yöntemlerde, birliktelik kuralı yöntemi ile kurallar üretmek yerine doğrudan yoğun nesne kümeleri kullanılmıştır. Bu da hem işlem yükünü azaltmış hem de hız kazancı sağlamıştır. Ayrıca birliktelik kuralında çok önemli iki kriter olan destek ve güven değerleri de bu yöntemlerde etkisiz bırakılmıştır. Bu yöntemlerde güven değeri kullanılmazken destek değeri ise istenildiği gibi seçilebilmektedir.
7. Birliktelik kuralı üreterek öğrenci özlük ve not bilgileri arasından birçok ilginç kural bulunması mümkündür. Elde edilen kurallar öğrenci başarısına etki edebilecek birçok faktörü ortaya çıkarabilmektedir. Bu kurallardan yola çıkarak öğrencilere yol gösterici bir rehberlik yapılması mümkündür.

8. Birliktelik kuralı sayesinde elde edilen kuralların yakın gelecekte de çok fazla değişmeyeceği varsayıldığında, birçok durum için geleceğe yönelik tahminlerin de yapılması mümkündür. Yapılan uygulama sonuçlarında, öğrencilerin geleceğe ait not tahminlerinde yüksek başarım değerleri elde edilmiştir.
9. Geliştirilen birliktelik kuralı tabanlı yazılım otomasyonu, öğrenci bilgilerinden kural üretmekte ve geleceğe yönelik tahminler yapabilmektedir. Geliştirilen yazılım, Fırat Üniversitesi öğrenci işlerine ait veritabanına göre ve C++ Builder ortamında hazırlanmıştır. Sadece Fırat Üniversitesi öğrencileri üzerinde kullanılabilmesine rağmen benzer yapıda hazırlanabilecek yazılımlar sayesinde birliktelik kuralının istenilen birçok alanda rahatlıkla iyi sonuçlar verebileceği düşünülmektedir.

7.2. Yayınlar

Bu tez çalışması, 1 adet uluslararası dergi yayını [98], 2 adet uluslararası konferans yayını [69, 122] ve 4 adet ulusal konferans yayını [43, 66, 116, 121] ile sonuçlanmıştır. Ayrıca 2 adet uluslararası dergi yayını [123, 124] ise hakem değerlendirmesinde bulunmaktadır.

KAYNAKLAR

1. Akpınar, H., 2000, "Veri tabanlarında bilgi keşfi ve veri madenciliği", İ.Ü. İşletme Fakültesi Dergisi, 29, 1-22.

2. Agrawal, R., Imielinski, T., Swami, A., 1993, "Mining association rules between sets of items in large databases", In ACM SIGMOD Conf. Management of Data.

3. Agrawal, R. and Srikant, R., 1994, Fast algorithms for mining association rules, Proceeding of the 20th Int'l Conference on Very Large Databases, Santiago, Chile.

4. Agraval, R., Manilla, H., Srikant, R., Toivonen, H., Verkamo, A.I., 1996, Fast discovery of association rules, Advanced in Knowledge Discovery and Data Mining, 307-328.

5. Manila, H., Toivonen, H., Verkamo, A.I., 1994, Efficient algorithms for discovering association rules. In Proceedings of AAAI'94 Workshop on Knowledge Discovery in Databases (KDD'94), 181-192, Seattle, Washington, USA.

6. Houtsma, M., Swami, A., 1995, Set-Oriented mining for association rules in relational databases, Proceedings of the 11th IEEE International Conference on DataEngineering, 25-34, Taipei, Taiwan,

7. Srikant, R., 1996, Fast algorithms for mining association rules and sequential patterns, Wisconsin Üniversitesi Doktora Tezi, Madison.

8. Park, J.S., Chen, M.S., Philip, S.Y., 1995, An effective hash based algorithm for mining association rules, Proceeding of ACM SIGMOID, s175-186, San Jose, California, USA.

9. Savasere, A., Omiecinski, E., Navathe, S.B., 1995, An efficient algorithm for mining association rules in large databases, 21. International Conference on Very Large Databases, 432-444, Zurih, İsviçre

10. Holsheimer, M., Kertsen, M., Manila, H., Toivonen, H., 1995, A perspective on databases and data mining, Proceeding of 1st International Conference on Knowledge and Data Mining, 150-155, Montreal, Kanada

11. Toivonen, H., 1996, Sampling large databases for association rules, 22. International Conference on Very Large Databases, 134-145, Mumbai, Hindistan

12. Brin, S., Motwani, R., Ullman, J.D., Tsur, S., 1997, Dynamic Itemset Counting and Implication Rules for Market Basket Data, Proceedings of the ACM SIGMOD Conference, 255-264.

13. Zaki, M.J., Parthasarathy, S., Ogihara, M., Li, W., 1997 New algorithms for fast discovery of association rules, 3. International. Conference. on Knowledge Discovery and Data Mining.

14. Bayardo, R.J., 1998, Efficiently mining long patterns from databases, Proceeding of ACM SIGMOID International Conference 85-93 Seattle, Washington, USA.

15. Hidber, C., 1999, Online association rule mining, proceedings ACM SIGMOD International Conference on Management of Data, 145-156, Philadephia, Pennsylvania,

16. Srikant, R., Agrawal, R., 1995, Mining generalized association rules, proceedings of the 21nd International Conference on Very Large Databases, 407-419, Zurich, İsviçre

17. Han, J., Fu, Y., 1995, Discovery of multiple-level association rules from large Databases, Proceedings of the 21nd International Conference on Very Large Databases, 420-431, Zurich, İsviçre.

18. Fortin, S., Liu, L., 1996, An object-oriented approach to multilevel association rule mining. In Proceeding of 5th International Conference on Information and Knowledge Management, 56-72, Rockville, Maryland, USA.

19. Thomas, S., Sarawagi, S., 1998, Mining generalized association rules and sequential patterns Using SQL Queries, In Proceeding of the 4th International Conference on Knowledge Discovery on Data Mining, 344-348, New York City, USA

20. Shen, L., Shen, H., Mining flexible multiple-level association rules in all concept hierarchies, In Proceeding of 9th International Conference on Database and Expert Sysyem Applications, 786-795, Vienna, Avusturya.

21. Dunham, M.H., Xiao, Y., Gruenwald, L., Hossain, Z., 2000. A survey of association rules, ACM Survey Journal.

22. Srikant, R., Vu, Q., Agraval, R., 1997, Mining association rules with item constraints, In Proceeding of 3rd International Conference on Knowledge Discovery and Data Mining, California, USA,

23. Aggarwal, C.C., Sun, Z., Yu P.S., 1998, Online algorithms for finding profile association rules, Proceedings of the ACM CIKM Conference, 86- 95.

24. Tsur, D., Ullman, J.D., ağabeyteboul, S., Clifton, C., Motwani, R., Nestorov, S., Rosenthal, A., 1998, Query flocks: a generalization of association rule mining, Proceedings ACM SIGMOD International Conference on Management of Data, Seattle, Washington, USA.

25. Srikant, R., Vu, Q., Agrawal, R., 1997, Mining association rules with item constraints, American Association for Artificial Intelligence,

26. Korn, F., Labrinidis, A., Kotidis, Y., Faloutsos, C., 1998, Ratio rules: A new paradigm for fast quantifiable data mining, Proceedings of the 24th VLDB Conference.

27. Brin, S., Motwani, R., Silverstein, C., Beyond market baskets: Generalizing association rules to correlations, Proceedings of the ACM SIGMOD Conference, 265-276.

28. Srikant, R., Agrawal, R., 1996, Mining quantitative association rules in large relational tables, In Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data, 1-12, Montreal, Quebec, Kanada.

29. Fukuda, T., Morimoto, Y., Morishita, S., Tokuyama, T., 1996, Mining optimized association rules for numeric attributes, PODS96, 182-191, Montreal, Quebec, Kanada.

30. Rastogi, R, Shim, K., 1998, Mining optimized association rules with categorical and numerical attributes, In Proceedings of 14th International Conference on Data Engineering. 503-512, Orlando, Florida, USA.

31. Wang, K., Tay, S.H.W., Liu, B., 1998, Interestingness based interval merger for numeric association rules, In Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining, 121-128, New York, USA

32. Kuok, C.M., Fu, A., Wong, M.H., 1998, Mining fuzzy association rules in databases, ACM SIGMOD RECORD, 27, No 1.

33. Agrawal, R., Srikant, R., 1995, Mining sequential patterns, In Proceedings of the 11th IEEE International Conference on Data Engineering, Taipei, Taiwan, IEEE Computer Society Press.

34. Srikant, R., Agraval, R., 1996, Mining sequential patterns: Generalizations and performance improvements, In Proceedings of the 5th International Conference on Extending Database Technology, 3-17, Avignon, Fransa

35. Guralnik, V., Wijesekera, D., Srivastava, J., 1998, Pattern directed mining of sequence data. In Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining, 51-57, New York, USA.

36. Das, G., Lin, K., Manila, H., Renganathan, G., Smyth, P., 1998, Rule discovery from time series. In Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining, 16-22, New York, USA

37. Cai, C.H., Fu, A.W.C., Cheng, C.H., Kwong, W.W., 1998, Mining association rules with weighted items, In Proceedings of 1998 International Database Engineering and Applications Symposium, 68-77, Cardiff, Wales.

38. Savasere, A., Omiecinski, E., Navathe, S.B., 1998, Mining for strong negative associations in a large database of customer transaction, In Proceedings of the 14nd International Conference on Data Engineering, 494-502, Orlando, Florida, USA.

39. Yuan, X., Buckles, B., Yuan, Z., Zhang, J., 2002, Mining negative association rules, Seventh International Symposium on Computers and Communications, 623-629, Taormina–Giardini Naxos, İtalya,

40. Savasere, A., Omiecinski, E., Navathe, S.B., 1995, An efficient algorithm for mining association rules in large databases, In Proceedings of the 21nd International Conference on Very Large Databases, 432-444, Zurich, İsviçre

41. Zongker, D., Jain, A., 1996, Algorithms for feature selection: An evaluation, Proceedings of ICPR 96.

42. Chatterjee, C., Roychowdhury, V.P., Chong, E.K.P., 1998, On relative convergence properties of principal component analysis algorithms, Neural Networks, IEEE Transactions on, 9 (2), 319- 329.

43. Karabatak, M., İnce, M.C., Avcı E., 2008, Göğüs kanseri için temel bileşen analizi yöntemi tabanlı uzman teşhis sistemi, IEEE 16. Sinyal İşleme ve İletişim Uygulamaları Kutultayı, Didim-Aydın.

44. R. Duda et al., 2001, Pattern Classification, John Wiley, 117-124,

45. Kitler, J., 1978, Feature set search algorithm, Pattern Recognition and Signal Processing, C. H. Chen, Ed., 41-60, Hollanda

46. Hyvarinen, A., Oja, E., 1999, Independent component analysis: Algorithms and applications, Neural Networks, 13, 411-430.

47. Hyvärinen, A., 1999, Fast and robust fixed-point algorithms for independent component analysis. IEEE Transactions on Neural Networks 10(3), 626-634.

48. Duda, R., Hart, P., Stork, D., 2001, Pattern classification, 2nd ed. New York: John Wiley and Sons.

49. Genç, H.M., Çataltepe Z., Pearson, T., 2007, Yeni bir temel/bağımsız bileşen analizi (TBA/BBA) tabanlı öznitelik seçme yöntemi, IEEE 15. Sinyal İşleme ve İletişim Uygulamaları Kutultayı, Eskişehir.

50. Stearns, S. D., 1976, On selecting features for pattern classifiers, Third International Conference on Pattern Recognition, 71-75.

51. Pudil, P., Ferri, F.J., Novovicova, J., Kittler, J., 1994, Floating search methods for feature selection with nonmonotonic criterion functions. IEEE 12th International Conference on Pattern Recognition, 2, 279–283.

52. Narendra, P.M., Fukunaga, K., 1977, A Branch and Bound algorithm for feature subset selection, IEEE Transaction on Computers, 26, no:9, 917-922.

53. Yu, B., Yuan B., 1993, A more efficient branch and bound algorithm for feature subset selection. Pattern Recognition, 26(6), 883-889.

54. Vriesenga, M.R., 1995, Genetic selection and neural modeling for designing pattern classifier. Doctora Tthesis, University of California-Irvine

55. Siedlecki, W., Sklansky, J., 1989, Anote on genetic algorithms for large-scale feature selection. Pattern Recognition Letters, 10, 335-347.

56. Arivazhagan, S. Ganesan, L., 2003, Texture classification using wavelet transform, Pattern Recog. Lett. 24, 1513-1521.

57. Weszka, J.S., Dyer, C.R., Rosenfeld, A., 1976, A comparative study of texture measures for terrain classification, IEEE Trans. Syst. Man. Cybernet. 6, 269-285,

58. Muneeswaran, K., Ganesan, L., Arumugam, S. Soundar, K., 2005, Texture classification with combined rotation and scale invariant wavelet features, Pattern Recognition, 38, 10, 1495-1506.

59. Li, S. Taylor, J.S., 2005, Comparison and fusion of multiresolution features for texture classification, Pattern Recognition Letters, 26, 5, 633-638.

60. Li, S., Kwok, J.T., Zhu, H. Wang, Y., 2003, Texture classification using the support vector machines Pattern Recognition, 36, 12, 2883-2893.

61. Mojsilovic, A., Rackov, D.M., Popovic M., 2000, On the selection of an optimal wavelet basis for texture characterization, IEEE Trans. On Image Processing, 9(12).

62. Chen, C.C., Chaur, C.C., 1999, Filtering methods for texture discrimination, Patt. Rec. Lett. 20, 783-790.

63. Wang, L. and Liu, J., 1999, Texture classification using multiresolution markov random field models, Pattern Recognition Letters, 20, 2, 171-182.

64. Rushing, J.A., Ranganath, H.S., Hinke, T.H., Graves, S.J.,2001, Using association rules as texture features, Pattern Analysis and Machine Intelligence, IEEE Transactions on 23, 845-858

65. Rushing, J.A., Ranganath, H.S., Hinke, T.H., Graves, S.J., 2002, Image segmentation using association rule features. IEEE Transactions on Image Processing 11(5), 558-567.

66. Karabatak, M., İnce, M.C., 2004, Apriori algoritması ile öğrenci başarısı analizi, Elektrik-Elektronik-Bilgisayar Mühendisliği Sempozyumu (ELECO2004), Elektronik-Bilgisayar 348-352, Bursa.

67. Başkaya H., Duru N., İmamoğlu T., 2005, Veri Madenciliği ile bir öğrenci notları analiz aracı", Bilimde Modern Yöntemler Sempozyumu (BMYS'2005), 751-758, Kocaeli.

68. İmamoğlu T., Duru N., Başkaya H., 2005, Veri Madenciliği ile bir öğrenci ders başarısı tahmin aracı, Bilimde Modern Yöntemler Sempozyumu (BMYS'2005), 782-789, Kocaeli.

69. Karabatak M., İnce M.C., 2006, The usage of data mining to analyze learning, International Conference on E-Portfolio Process in Vocational Education (EPVET'2007), Bucharest, Romanya.

70. Frawley, W. J., Piatetsky-Shapiro, G., Matheus, C. J. 1991, "Knowledge discovery databases: An overview", Knowledge Discovery in Databases (G. Piatetsky-Shapiro and W. J. Frawley, eds.), 1-27, Cambridge, MA: AAAI/MIT.

71. Agrawal, R. and Shafer, J.C., 1996, "Parallel mining of association rules: Design, Implementation and Experience", IBM Research Report RJ 10004.

72. Chen, M. S., Han, J., Yu, P. S., 1996, "Data mining: An overview from a database perspective", IEEE Transactions on Knowledge and Data Engineering, 8(6), 866-883.

73. Dilly, Rulh. 1995, Data Mining: An Introduction www-pcc.qub.ac.uk/tec/ courses/datamining/stu_notes/dm_book_1.html (10/06/1999).

74. Alataş, B., 2002, Veri madenciliği, Fırat Ünv. Fen Bilimleri Enstitüsü Yüksek Lisans Semineri, Elazığ.

75. Cambazoğlu, T., 2002, Kurumlarda yararlı bilginin yönetimi ve ilintili teknolojiler-18, www.bilisimrehber.com.tr/arastirma (28.01.2002) 2-3

76. Berry, M.J., Linoff, G., 1997, Data Mining Techniques, Hohn Wiley & Sons Inc Publisher, 389

77. Biçer, P., 2002, Veri madenciliği: Sınıflandırma ve tahmin yöntemlerini kullanarak bir uygulama" Yıldız Tek. Ünv. Sosyal Bil. Ens. Yüksek Lisans Tezi, İstanbul.

78. Gürsakal, N., Acar, F., 1999, İstatistik, veri analizi ve veri madenciliği, IV. Ulusal Ekonometri ve İstatistik Sempozyumu Bildirileri, 5-6

79. Quinlan, J. R., 1986, "The effect of noise on concept learning. In Machine Learning: In An Artificial Intelligence Aproach", R. Michalski, J. Carbonell, and T. Mitchell, (eds.), 2, 149-166, SanMateo, CA: Morgan Kauffmann Inc.

80. Chan K.C.C., Wong, A.K.C., 1991, A statistical technique for extracting classificatory knowledge from databases, Knowledge Discovery in Databases (G. Piatetsky-Shapiro and W. J. Frawley, eds.), 107-123, Cambridge, MA: AAAI/MIT.

81. Lee, S. K., 1992, An extended relational database model for uncertain and imprecise information, Proceeding of the ISth VLDB conference, (Vancouver, British Columbia, Canada), 211-218.

82. Luba, T., Lasocki, R., 1994, On unknown attribute values in functional dependencies, Proc. of the International Workshop on Rough Sets and Soft Computing, (San Jose, CA), 490-497.

83. Grzymala-Busse, J.W., 1991, On the unknown attribute values in learning from examples, Proceeding of Methodologies for Intelligent Systerrzs, Z. W. Ras and M. Zemankowa, (eds.), Lecture Notes in AI, 542, 368-377, New York: Springer-Verlag.

84. Quinlan, J. R., 1986, Induction of decision trees, Machine Learning, l, 81-106

85. Kira, K. and Rendell, L., 1992, The feature selection problem: Traditional methods and a new algorithm" Proc. of AAAI 92, 129-134, AAAI Press.

86. Almuallim, H. and Dietterich, T., 1991, Learning with many irrelevant features, Proceeding of AAAI 91, (Menlo Park, CA), 547-552, AAAI Press.

87. Türkoğlu İ., 1996, Yapay sinir ağları ile nesne tanıma, F.Ü. Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, Elazığ, 112s.

88. Koyuncu F., 2004, Veri madenciliğinde apriori temelli ilişkilendirme kurallarının uygulama ve karşılaştırması, Yıldız Teknik Ünv. Fen Bilimleri Enstitüsü Yüksek Lisans Tezi, İstanbul, 113s.

89. Ding, Q., 2001, Association rule mining survey.

90. Han, J., Kamber , M., 2001, Data Mining: Concepts and Techniques, Morgan Kaufman Publishers, 225

91. Ayan, N.F., 1999, Updating large itemset with early pruning, Bilkent Üniversitesi, Mühendislik ve Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, Ankara, 66s

92. Başkaya, H., 2005, Veri madenciliği ve birliktelik kuralları ile bir öğrenci notları çözümleme aracı, Kocaeli Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, Kocaeli, 58s.

93. Zaki, M.J., Parthasarathy, S., Li, W., Ogihara M., 1997, Evaluation of sampling for data mining of association rules, 7. International Workshop on Research Issues in Data Engineering, Birmingham, UK.

94. Özden, B., Ramaswamy, S., Silberschatz, A., 1998, Cyclic association rules, In Proceedings of the 16th British National Conference on Database, 49-63, Cardiff, Wales.

95. Ramaswamy, S., Mahajan, S., Silberschatz, A., On the discovery of interesting patterns in association rules, 1998, In Proceedings of the 24th International Conference on Very Large Datanasaes, 368-379, New York, USA

96. Han, J., Dong, G., Yin, Y., 1999, Efficient mining of partial periodic patterns in time series databases, In Proceedings of the 15th International Conference on Data Engineering, 1206-115, Sydney, Avustralya.

97. Onnia, V., Tico, M., Saarinen, J., 2001, Feature selection method using neural network, Proceeding 2001 International Conferencee on Image Processing, Thessaloniki, Yunanistan.

98. Karabatak, M., Ince, M.C., 2008, An expert system for detection of breast cancer based on association rules and neural network, Expert Systems with Applications, In Press, Corrected Prof.

99. Principe, J.C., Euliano, N.R. Lefebvre, W.C., 2000, Neural and adaptive systems, John Wiley & Sons, 1. Baskı, New York, 656s,

100. Türkoğlu, İ., ve Hanbay, D., 2001, Yapay sinir ağı ve HFD kullanarak DTMF sinyal örüntülerini tanıma sistemi", Elektrik - Elektronik – Bilgisayar Mühendisliği 9. Ulusal Kongresi, 431-434, Kocaeli.

101. Şengür, A., Türkoğlu, İ., ve İnce, M.C., 2005, Eğiticisiz yapay sinir ağları ile görüntü bölütleme uygulamaları, IEEE 13. Sinyal İşleme ve İletişim Uygulamaları Kurultayı, 271-274, Kayseri.

102. Turkoğlu, İ., 2002, Durağan olmayan işaretler için zaman-frekans entropilerine dayalı akıllı örüntü tanıma, Doktora Tezi, Fırat Üniversitesi, Fen Bilimleri Enstitüsü.

103. Choua, S.-M., Leeb, T.-S., Shaoc, Y.E., Chenb, I-F., 2004, Mining the breast cancer pattern using artificial neural networks and multivariate adaptive regression splines, Expert Systems with Applications, 27, 133-142

104. Elmore, J., Wells, M., Carol, M., Lee, H., Howard, D., Feinstein, A., 1994, Variability in radiologists interpretation of mamograms, New England Journal of Medicine 331(22), 1493–1499.

105. Şengür, A., 2006, Endoskopik görüntülerin değerlendirilmesinde görüntü işleme temelli akıllı karar destek sistemi, F.Ü. Fen Bilimleri Enstitüsü, Doktora Tezi, Elazığ.

106. Sengur, A., Turkoglu, İ, Ince, M.C., 2007, Wavelet packet neural networks for texture classification, Expert Systems With Applications, 32(2).

107. Avcı, E., 2007, An expert system based on wavelet neural network-adaptive norm entropy for scale invariant texture classification, Expert Systems with Applications, 32(3), 919-926.

108. Haralick, R.M., 1979, Statistical and structural approaches to texture, Proc. IEEE, 67, no:5, 786-804.

109. Chu, A., Sehgal, C.M., Greenleaf, J.F., 1990, Use of gray value distribution of run lengths for texture analysis, Pattern Recognition Letters, 11, 415-420.

110. Chaudhuri, B.B., Sarkar, N., 1995, Texture Segmentation Using Fractal Dimension, IEEE Trans. Pattern Analysis and Machine Intelligence, 17, no:1, 72-77.

111. Chellappa, R., Chatterjee, S., 1985, Classification of textures using gaussian markov random fields, IEEE Trans. Acoustics, Speech, and Signal Processing, 33(4), 959-963.

112. Fogel, I., Sagi, D., 1989, Gabor filters as texture discriminator, Biological Cybernetics, 61, 103-113.

113. Bigun, J., Granlund, G.H., Wiklund, J.,1991, Multidimensional orientation estimation with applications to texture analysis and optical flow, IEEE Trans. Pattern Analysis and Machine Intelligence, 13, no:8, 775-790.

114. Canny, J., 1986, A computational approach to edge detection, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-9, No.6, 679-698.

115. Brodatz, P., 1966, TEXTURES, A photographic album for artists and designers. Dover, New York.

116. Karabatak M., Şengür A., İnce M.C., 2006, Kenar çıkarma ve birliktelik kuralı kullanarak doku resimlerinin sınıflandırılması, IEEE 14. Sinyal İşleme ve İletişim Uygulamaları Kutultayı, Antalya.

117. İnternet: http: // sipi.usc.edu/ database/ database.cgi ? volume = textures & image = 1#top, (10.10.2006)

118. Akin, M., Arserim, M.A., Kıymık, M.K., Türkoğlu, İ., 2001, A new approach for diagnosing epilepsy by using wavelet transform and neural networks, 23rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC2001), 25-28 October 2001, İstanbul.

119. Akay, M., 1994, Investigating the effects of vasodilator drugs on the turbulent sound caused by femoral artery stenosis using short-term fourier and wavelet transform methods, IEEE Transaction ob Biomedical Engineering, 41,921-928.

120. Daubechies, I., 1988, Orthogonal bases of compactly supported wavelets. Comm. Pure. Appl. Math. 41, 909-996.

121. Karabatak M., Şengür A., İnce M.C., 2007, Dalgacık bölgesi birliktelik kuralları kullanarak doku sınıflandırma, IEEE 15. Sinyal İşleme ve İletişim Uygulamaları Kutultayı, Eskişehir.

122. Karabatak M., Şengür A., İnce M. C., Türkoğlu İ., 2006, Association rules for texture classification, Proceedings of 5th International Symposium on Intelligent Manufacturing Systems, May 29-31, 96-104

123. Karabatak, M., İnce, M.C., 2007, Wavelet domain association rules for efficient texture classification, Applied Soft Computing (Hakem değerlendirmesinde)

124. Karabatak, M., İnce, M.C., 2008, A hybrid method for diagnosis of erythemato-squamous diseases based on association rules and neural network, Expert Systems with Applications (Hakem değerlendirmesinde)

ÖZGEÇMİŞ

Murat KARABATAK

Fırat Üniversitesi
Teknik Eğitim Fakültesi
Elektronik ve Bilgisayar Eğitimi Bölümü
23119, Elazığ

Tel : 424 – 2370000 – 4292
E.posta : mkarabatak@firat.edu.tr

1976 Elazığ da doğdu.

1990 – 1994 Elazığ Anadolu Meslek Lisesini tamamladı.

1995 – 1999 Fırat Üniversitesi Teknik Eğitim Fakültesi Elektronik ve Bilgisayar Eğitimi Bölümünden mezun oldu.

1999 – 2002 Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elektronik ve Bilgisayar Eğitimi Anabilim Dalında "Web'e Dayalı Uzaktan Eğitimde Otomasyon" konusunda yaptığı tez çalışması ile Yüksek Lisans derecesini aldı.

1999 – 2000 6 ay süre ile T.C. M.E.B. da çalıştı.

2000 – ... Fırat Üniversitesi, Teknik Eğitim Fakültesi Elektronik-Bilgisayar bölümünde araştırma görevlisi ve öğretim görevlisi kadrolarında çalışmaktadır.