VERİ MADENCİLİĞİNDE SINIFLANDIRMA PROBLEMLERİ İÇİN EVRİMSEL ALGORİTMA TABANLI YENİ BİR YAKLAŞIM:ROUGH-MEP ALGORİTMASI

Emel Kızılkaya AYDOĞAN

DOKTORA TEZİ
ENDÜSTRİ MÜHENDİSLİĞİ

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

OCAK 2008
ANKARA

Emel Kızılkaya AYDOĞAN tarafından hazırlanan VERİ MADENCİLİĞİNDE SINIFLANDIRMA PROBLEMLERİ İÇİN EVRİMSEL ALGORİTMA TABANLI YENİ BİR YAKLAŞIM:ROUGH-MEP ALGORİTMASI adlı bu tezin Doktora tezi olarak uygun olduğunu onaylarım.

Prof. Dr. Cevriye GENCER ……..
Tez Danışmanı, Endüstri Müh.A.D.

Bu çalışma, jürimiz tarafından oy birliği ile Endüstri Mühendisliği Anabilim Dalında Doktora tezi olarak kabul edilmiştir.

Prof. Dr. Semra Oral ERBAŞ …….....
İstatistik Bölümü A.D., G.Ü.

Prof. Dr. Hadi GÖKÇEN ……..
Endüstri Mühendisliği A.D., G.Ü.

Prof. Dr. Cevriye GENCER …….
Endüstri Mühendisliği A.D., G.Ü.

Doç. Dr. Mehmet Ali AKÇAYOL ……..
Bilgisayar Mühendisliği A.D., G.Ü.

Yrd.Doç.Dr.Tuğba Taşkaya Temizel …….
Enformatik A.D., O.D.T.Ü.

Tarih: 15/01/2008

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

Prof. Dr. Nermin ERTAN …….
Fen Bilimleri Enstitüsü Müdürü

TEZ BİLDİRİMİ

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

Emel Kızılkaya AYDOĞAN

VERİ MADENCİLİĞİNDE SINIFLANDIRMA PROBLEMLERİ İÇİN EVRİMSEL ALGORİTMA TABANLI YENİ BİR YAKLAŞIM:ROUGH-MEP ALGORİTMASI

(Doktora Tezi)

Emel Kızılkaya AYDOĞAN

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
Ocak 2008

ÖZET

Veri madenciliği, önceden bilinmeyen ilişki ve eğilimlerin bulunması için büyük miktarlardaki veriyi analiz eden, kullanıcılar için anlamsız bilgiyi anlamlı hale dönüştüren bir yöntemdir. Veri madenciliğinde sınıflandırma ise, verilen örneklerden hareket ederek her bir sınıfa ilişkin özellikleri bulan ve bu özelliklerin kural cümleleri ile ifade edilmesini sağlayan bir yaklaşımdır.

Bu tezde, veri madenciliğinde sınıflandırma kurallarının keşfi için kaba küme yaklaşımıyla evrimsel algoritmalara dayanan yeni bir algoritma "Rough-Mep algoritması" önerilmiştir. Rough-Mep algoritmasının etkinliği, klasik makine öğrenimi algoritmaları ve literatürde bulunan algoritmalarla karşılaştırılmış; ikili veya çoklu sınıflı veri kümeleri üzerinde test edilmiştir.

Bilim Kodu : 906.1.148
Anahtar Kelimeler: Veri madenciliği, sınıflandırma, nitelik indirgeme, kaba küme, evrimsel programlama.
Sayfa Adedi : 137
Tez Yöneticisi : Prof. Dr. Cevriye GENCER

A NEW APPROACH BASED ON EVALUATION ALGORITHM FOR CLASSIFICATION PROBLEMS IN DATA MINING
(Ph.D. Thesis)

Emel Kızılkaya AYDOĞAN

GAZİ UNIVERSITY
INSTITUTE OF SCIENCE AND TECHNOLOGY
January 2008

ABSTRACT

Data mining is a method for finding unknown relation and trends that analyses great amount of data and transforms insignificant information to significant knowledge for users. Classification in data mining is an approach finding out related attributes of each class and providing display with rule statements from given data sets.

In this thesis a new algorithm "Rough-Mep algorithm" is proposed for discovering of classification rules based on rough set theory and evaluation algorithms. The effectiveness of our approach is tested on eight publicly available binary and n-ary classification data sets.

Science Code : 906.1.148
Key Words : Data mining, classification, , attribute reduction, rough set, evolutionary programming.
Page Number : 137
Adviser : Prof.Dr.Cevriye GENCER

TEŞEKKÜR

Çalışmalarım boyunca değerli yardım ve katkılarıyla beni yönlendiren, bana her konuda destek olan, sonsuz saygı ve sevgi duyduğum çok değerli hocam, Prof. Dr. Cevriye GENCER'e, yine kıymetli tecrübelerinden faydalandığım, çok sevdiğim değerli hocalarım Prof. Dr. Hadi GÖKÇEN, Doç.Dr. Mehmet Ali AKÇAYOL'a, savunma jürimde bulunan hocalarım Semra Oral ERBAŞ, Tuğba Taşkaya TEMİZEL'e, Endüstri Mühendisliği bölümündeki tüm hocalarıma, aile gibi olduğumuz tüm asistan arkadaşlarıma, manevi destekleriyle beni hiçbir zaman yalnız bırakmayan sevgili eşim, oğlum ve aileme teşekkürü bir borç bilirim.

Ayrıca verdiği proje desteğinden dolayı Bilimsel araştırma projeleri birimine de teşekkür ederim.

İÇİNDEKİLER

Sayfa

ÖZET……iv

ABSTRACT……v

TEŞEKKÜR……vi

İÇİNDEKİLER……vii

ÇİZELGELERİN LİSTESİ……x

ŞEKİLLERİN LİSTESİ……xi

SİMGELER VE KISALTMALAR……xii

1. GİRİŞ……1

2. VERİ MADENCİLİĞİ……5

2.1. Neden Veri Madenciliği?……10

2.2. Veri Madenciliğinin Gereksinimleri……11

2.3. Veri Madenciliğinin Adımları……11

2.4. Veri Madenciliği'nin Amaçları ve Veri Madenciliğinde Karşılaşılan Problemler……12

2.5. Veri Madenciliği Faaliyetleri ve Kullanıcıları……14

2.6. Veri Madenciliğinin Kullanım Alanları……15

2.7. Veri Madenciliğinin Faktörleri……17

2.8. Veri Madenciliğinde Kullanılan Yaklaşımlar……18

2.8.1. Sınıflandırma ve tahmin……18

2.8.2. Kümeleme……19

2.8.3. Birliktelik analizi ve ardışık analiz……19

2.9. Veri Madenciliği'nde Sınıflandırma Problemleri için Kullanılan Yöntemler……20

2.9.1. Karar ağacı ile sınıflandırma yöntemi 20

2.9.2. Bayes ile sınıflandırma yöntemi 20

2.9.3. Naive bayes ile sınıflandırma yöntemi 21

2.9.4. k- en yakın komşuluk (k-NN) ile sınıflandırma yöntemi 21

2.9.5. Vaka tabanlı nedenleşme (CBR) ile sınıflandırma yöntemi 21

2.9.6. Genetik algoritma ile sınıflandırma yöntemi 21

2.9.7. Kaba küme teorisi ile sınıflandırma yöntemi 23

2.9.8. Genetik programlama (GP) ile sınıflandırma yöntemi 23

2.10. Literatürde Evrimsel Algoritmalar Kullanılarak Sınıflandırma Amaçlı Yapılmış Olan Çalışmalar 25

2.11. Veri Madenciliği'nde Nitelik İndirgeme Yaklaşımı 31

2.11.1. Boyut indirgeme yöntemleri 32

3. KABA KÜME TEORİSİ 42

3.1. Kaba Küme Analizinin Temel Kavramları 46

3.1.1. Bilgi sistemleri 46

3.1.2. Karar tabloları 47

3.1.3. Nitelik Bağımlılığı 47

3.1.4. Nitelik indirgeme 48

3.1.5. Ayırt edilebilirlik matrisi ve fonksiyonlar 48

3.1.6 Niteliklerin önemi 49

3.1.7. Değişken duyarlı kaba kümeler (VPRS) 50

3.2. Literatürdeki Çalışmalar 51

4. VERİ MADENCİLİĞİNDE SINIFLANDIRMA PROBLEMLERİ İÇİN YENİ BİR YAKLAŞIM: ROUGH-MEP ALGORİTMASI 58

4.1. Rough-Mep Algoritması 59

4.1.1. Rough-Mep algoritmasında sürekli niteliklerin kesikleştirilmesi ..... .59

4.1.2. Rough-Mep algoritmasında kaba küme yaklaşımıyla GA-tabanlı nitelik indirgemesi 63

4.1.3. Rough-Mep algoritmasıyla sınıflandırma kurallarının çıkarımı 64

4.2. Deneysel Çalışma..... 72

4.3. Deneysel Düzenek..... 78

4.4. Rough-Mep Algoritmasının Performansının Karşılaştırması 81

4.4.1. Klasik makina öğrenimi algoritmalarıyla karşılaştırma 81

4.4.2. Diğer kural tabanlı sınıflandırma algoritmalarıyla karşılaştırma 84

5. SONUÇ VE ÖNERİLER 86

KAYNAKLAR 89

EKLER……100
EK-1……101
EK-2……102
EK-3……105
EK-4……108
EK-5……111
EK-6……115
EK-7……119
EK-8……122
EK-9……125
EK-10……126
EK-11……127
EK-12……128
EK-13……129
EK-14……130
EK-15……131
EK-16……132
EK-17……133

ÖZGEÇMİŞ..... ……134

ÇİZELGELERİN LİSTESİ

Çizelge **Sayfa**

Çizelge 2.1. Veri madenciliği'nde kullanılan yazılım sağlayıcılardan örnekler 9

Çizelge 2.2. Veri madenciliğinin faktörleri 17

Çizelge 4.1. Olumsallık tablosu 61

Çizelge 4.2. Veri kümelerinin ana özellikleri 75

Çizelge 4.3. Rough-Mep algoritmasının tahmini doğruluk değerleri 76

Çizelge 4.4. Rough-Mep algoritmasında, varsayılan sınıf yapısı olarak TP kullanıldığında oluşan tahmini doğruluk değerleri 78

Çizelge 4.5. Rough-Mep algoritmasının klasik makina öğrenimi algoritmalarıyla karşılaştırma sonuçları 83

Çizelge 4.6. Rough-Mep algoritması ile veri madenciliği algoritmalarının tahmini doğruluklarının karşılaştırılması 85

ŞEKİLLERİN LİSTESİ

Şekil **Sayfa**

Şekil 2.1. Bilgi keşfi süreci 5

Şekil 2.2. Veri madenciliği süreci 6

Şekil 2.3. Veri madenciliği sürecine katkı sağlayan alanlar 10

Şekil 2.4. Veri kümeleme 19

Şekil 2.5. Genetik Programlama Döngüsü 24

Şekil 2.7. Dönüşüm tabanlı yöntemler 34

Şekil 2.8. Nitelik seçim stratejileri 36

Şekil 2.9. Nitelik Seçimi 36

Şekil 2.10. Filtre yaklaşımı 39

Şekil 2.11. Sarma yaklaşımı 40

Şekil 4.1. Rough-Mep algoritmasındaki örnek kromozom yapısı 66

Şekil 4.2. Çaprazlamadan önce aile kromozom yapıları 69

Şekil 4.3. Çaprazlama sonucu oluşan döl kromozomlar 70

Şekil 4.4. Rough-Mep algoritmasında mutasyon operatörü uygulanmadan önceki ve sonraki kromozom yapıları 71

Şekil 4.5. Rough-Mep algoritmasının akış şeması 70

Şekil 4.6. Popülasyon büyüklüğü için parametre en iyileme sonuçları 79

Şekil 4.7. Kod uzunluğu için parametre en iyileme sonuçları 79

Şekil 4.8. Mutasyon oranı için parametre en iyileme sonuçları 80

Şekil 4.9. Jenerasyon sayısı için parametre en iyileme sonuçları 80

SİMGELER VE KISALTMALAR

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

Kısaltmalar	**Açıklama**
GP	Genetik Programlama
GA	Genetik Algoritma
k-NN	k- En Yakın Komşuluk
CBR	Vaka Tabanlı Nedenleşme
VPRS	Değişken Duyarlı Kaba Kümeler
ES	Evrimsel Stratejiler
EP	Evrimsel Programlama
KK	Kaba Küme

1. GİRİŞ

Bugünün iş dünyasında şirketler, müşteri verilerinden bilgi türetmenin rekabet avantajı sağlayan önemli bir yetenek olduğunu keşfetmişlerdir. Bir karar verici için verilen kararın doğruluğu, onun yeteneklerine, deneyimine ve bilgi birikimine olduğu kadar sahip olduğu veri kümesinin yeterliliğine de bağlıdır. İşte bu noktada devreye giren veri madenciliği, önceden bilinmeyen ilişki ve trendlerin bulunması için karar vericiye yardımcı olan ve büyük miktarlardaki veriyi analiz eden bir alan olarak karşımıza çıkmaktadır.

Günümüzde ham veri eksikliği yaşanmamaktadır. Tek sorun rekabet avantajı sağlayacak ve ürünleri daha cazip kılacak şekilde ham verinin nasıl kullanılacağıdır. Ayrıca şirket için ham verilerin tam zamanında elde edilebilmesi de bir o kadar önemlidir. Örneğin; bir ürünün üretiminin bir haftada tamamlandığını düşünelim. Bu ürünün kalite kontrolü bu bir hafta içinde yapılmalıdır. Ürün üretildikten, nakledildikten ve raflardaki yerini aldıktan sonra ortaya çıkan bir kalite kontrol bilgisi yararsız olacaktır. Bir istatistikçinin bu verideki eğilimleri bir haftadan daha uzun sürede keşfetmesi, şirketi muhtemel bir felaketle karşı karşıya bırakır. Oysa ki, veri madenciliği süreç boyunca veriyi izler ve hatalı ürünlere neden olan benzer eğilimleri ortaya çıkararak gelir kaybını önlediği gibi müşteri sadakatini de korur. Ayrıca üreticiler için önemli olan bilgilerin elde edilmesinde ve doğru kararın alınmasının yanında, verilerin doğru depolanması, doğru sınıflanması, doğru ayıklanması, doğru işlenmesi ve doğru yorumlanması da çok önemlidir.

Veri madenciliği çalışmalarında başarılı olmanın ilk şartı, uygulamanın hangi amaç için yapılacağının açık bir şekilde tanımlanmasıdır. Uygulamanın amacı, sorun üzerine odaklanmış ve açık bir dille ifade edilmiş olmalıdır. Sorunu tanımlayamayan bir veri madenciliği çalışması, sorunu çözmeye yetmeyeceği gibi sonuçta başka problemlerin de ortaya çıkmasına neden olabilecek ve bu da çeşitli maliyetlere sebep olacaktır.

Uygulama amacı belirlendikten sonra verilerin hazırlanması gerekir. Verilerin hazırlanması aşaması kendi içerisinde toplama ve uyumlaştırma, birleştirme ve temizleme ile seçim adımlarından meydana gelmektedir.

Toplama ve uyumlaştırma, tanımlanan sorun için gerekli olduğu düşünülen verilerin ve bu verilerin toplanacağı veri kaynaklarının belirlenmesi adımıdır. Hangi veri kaynaklarından yararlanılacağı önemli bir karardır. Çünkü gereğinden az veri kaynağı veri madenciliği çalışmasını eksik bırakacağı gibi, gereğinden fazla veri kaynağı sürecin uzamasına neden olabilecek veri kirliliğine yol açabilecektir.

Birleştirme ve temizleme adımında, farklı kaynaklardan toplanan veriler düzenlenerek tek bir veri tabanında toplanır.

Seçim adımında, kurulacak modele bağlı olarak veri seçimi yapılır.

Veri madenciliğinde verilerin, farklı kaynaklardan toplanması durumunda kullanılacak verilerin farklı zamanlara ait olabilmeleri, veri formatlarının farklı olması, farklı kodlama sistemlerinin kullanılması, güncelleme hataları vb. birçok problem ortaya çıkabilecektir. Güvenilir olmayan veri kaynaklarının kullanımı tüm veri madenciliği sürecinin de güvenilirliliğini etkileyecektir. Bu nedenlerle, iyi sonuç alınacak veri madenciliği çalışmaları ancak iyi verilerin üzerine kurulabileceği için, toplanan verilerin ne ölçüde uyumlu oldukları verilerin hazırlanması aşamasında incelenerek değerlendirilmelidir.

Veri madenciliği çalışmasında geliştirilen modelde kullanılan veri tabanının çok büyük olması durumunda, rastgeleliği bozmayacak şekilde örnekleme yapılması uygun olabilir. Ayrıca burada seçilen örneklem kümesinin tüm popülasyonu temsil edip etmediği de kontrol edilmelidir.

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, denetimsiz ve denetimli öğrenmenin kullanıldığı modellere göre farklılık göstermektedir.

Denetimsiz öğrenmede, ilgili örneklerin gözlemlenmesi ve bu örneklerin özellikleri arasındaki benzerliklerden hareket edilerek sınıfların tanımlanması amaçlanmaktadır.

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 [Han, 2001].

Tezde geliştirilen algoritmada bir denetimli öğrenme algoritmasıdır.

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

Veri madenciliği sürecinde önemli olan bir diğer unsur modelin izlenmesidir. Zaman içerisinde bütün sistemlerin özelliklerinde ve dolayısıyla ürettikleri verilerde ortaya çıkan değişiklikler, kurulan modellerin sürekli olarak izlenmesini ve gerekiyorsa yeniden düzenlenmesini gerektirecektir. Tahmin edilen ve gözlenen değişkenler arasındaki farklılığı gösteren grafikler model sonuçlarının izlenmesinde kullanılan yararlı bir yöntemdir.

Yukarıda bahsedilen bilgiler ışığında sınıflandırma problemleri için kural üreten yeni bir veri madenciliği algoritması geliştirilmiş ve etkinliği literatürde bulunan test problemleri üzerinde sınanmıştır.

Tezin ikinci bölümünde; veri madenciliğinin literatürdeki çeşitli tanımları, amaçları, adımları, gereksinimleri ve kullanım alanları anlatılmıştır.

Üçüncü bölümde, kaba küme teorisinin temel kavramlarından bahsedilmiş ve literatür taraması verilmiştir.

Dördüncü bölümde, kaba küme yaklaşımı kullanarak nitelik indirgemesi yapan ve genetik algoritma ve genetik programlama tabanlı kural çıkarım mekanizmasına sahip yeni bir evrimsel algoritma: “Rough-Mep algoritması” geliştirilmiş ve anlatılmıştır.

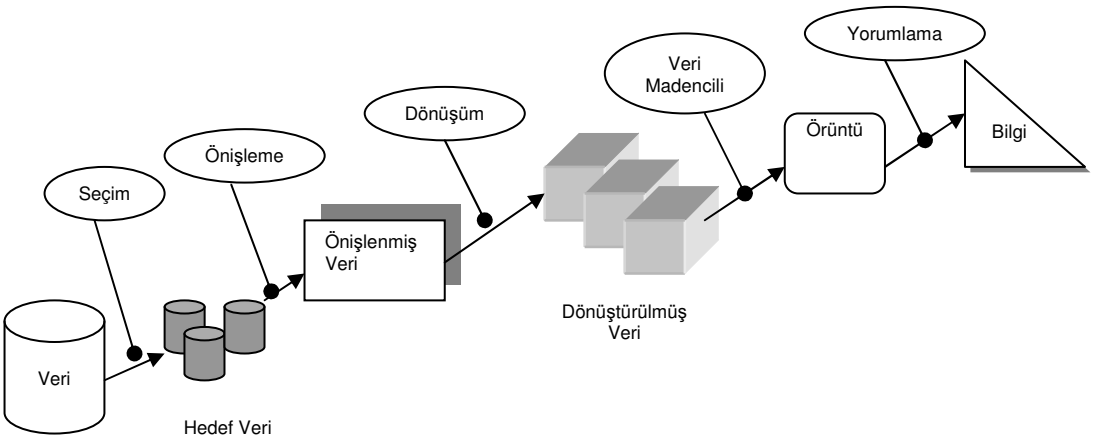
Sonuç bölümü olan beşinci bölümde, geliştirilen Rough-Mep algoritmasının sınıflandırma performansı, klasik makine öğrenimi algoritmaları ve literatürdeki iyi sonuç vermiş bulunan çeşitli algoritmalarla karşılaştırılmış, sekiz veri üzerinde yapılan deneysel çalışma ve sonuçları sunulmuştur.

2. VERİ MADENCİLİĞİ

Bilgi keşfi; önceden bilinmeyen, kapalı ve potansiyel olarak yararlı bilginin veriden çekilmesidir. Bilgi keşfi sürecinin adımları Şekil 2.1'de verilmektedir.

Bilgi keşfi sürecinin bir parçası olan veri madenciliği, özellikle yararlı örüntülerin ve ilişkilerin bulunması için istatistiksel analizler ve modelleme tekniklerinin uygulandığı alandır.

Veri madenciliği, geniş veritabanlarından prediktif bilginin otomatik olarak çekilme sürecidir. Gelecek trendlerini tahmin eder ve uzmanların gözden kaçırdığı davranışları bulur [Han, 2001].



Şekil 2.1. Bilgi keşfi süreci

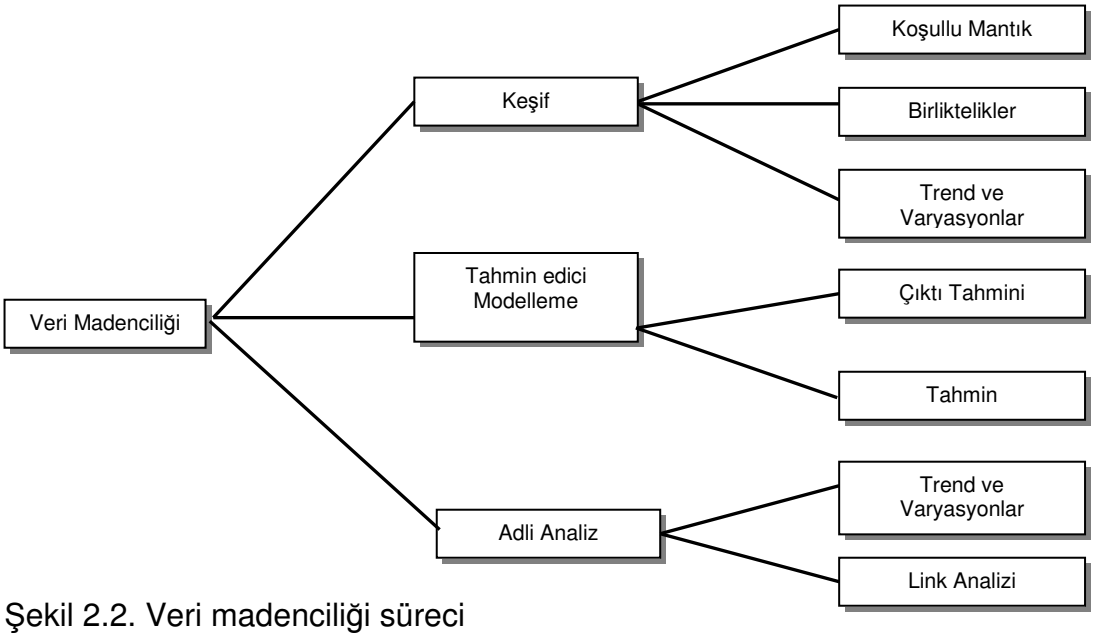
Fayyad'a göre veri tabanlarından bilgi keşfi sürecinde yer alan adımlar şu şekilde sıralanmaktadır [Fayyad ve ark., 1996]:

- *Veri seçimi:* Bu adım, veri kümelerinden, keşfin gerçekleştirileceği uygulama alanına uygun örneklem kümesinin seçilmesidir.
- *Veri temizleme ve ön işleme:* Seçilen örneklem kümesinde yer alan hatalı tutanakların çıkarıldığı ve eksik nitelik değerlerinin nasıl ele alınacağının

belirlenerek, değiştirildiği aşamadır. Bu aşama keşfedilen bilginin kalitesini arttırır.

- *Veri indirgeme:* Seçilen örneklemden ilgisiz niteliklerin atıldığı ve tekrarlı kayıtların ayıklandığı adımdır. Bu aşama seçilen veri madenciliği sorgusunun çalışma zamanını iyileştirir.
- *Veri madenciliği:* Kullanılacak veri madenciliği algoritmasının (sınıflandırma, kümeleme vb.) belirlenerek uygulanmasıdır.
- *Değerlendirme:* Veri madenciliği sonucunda keşfedilen örüntülerin geçerlilik, yenilik, yararlılık ve basitlik kıstaslarına göre değerlendirilmesi aşamasıdır.

Veri madenciliği, verideki bilgi örüntülerinin arandığı karar destek sürecidir. "Örüntü" ile anlatılmak istenen "Verideki İlişkiler" dir. Veri madenciliğinin süreci Şekil 2.2'de verilmektedir.



Şekil 2.2. Veri madenciliği süreci

Veri madenciliği; trendleri, ilişkileri ve profilleri belirlemek için veriyi sınıflandıran bir analitik araç ve bilgisayar yazılım paketidir [SAS web site, 2007].

Veri madenciliği, pazarda şirkete rekabetçi bir yapı kazandırmak için veri analizine gerek duyar, bu yüzden; kolaylaştırıcı bir faktör olarak görülebilir [Cabena, 1998]. "*Bugün birçok işletme için müşterilerinin istek ve ihtiyaçlarını bilmek en önemli önceliktir*" [LaMonica, 1997]. Bu nedenle; veri madenciliği bir şirketin rekabetçi yapıya kavuşabilmesi için son derece önemlidir. Bugünün iş dünyasında; veri miktarındaki önemli artışa paralel olarak veri madenciliği de çok daha önemli hale gelmiştir. İnsanlar üç beş boyutla ilgilenebilmektedirler fakat 10 boyutdan fazla olduğu zaman bu iş zorlaşmaktadır. Bu durum veri toplanmasında yeni teknolojilerin geliştirilmesinde önemlidir [Fayyad, 2001].

Aşağıda veri madenciliği ile ilgili çeşitli tanımlar yer almaktadır:

- Solid State Technology'e göre veri madenciliği, bir ya da daha fazla cevap değişkeninin davranışını açıklamak için veritabanlarında mevcut geniş kümelerdeki gizli ilişkileri bulan bir metedolojidir [Solid State Technology, 1999].
- Jacobs veri madenciliğini, ham verinin tek başına sunamadığı bilgiyi çıkaran veri analizi süreci olarak tanımlamıştır [Jacobs, 1999].
- Davis, veri madenciliğinin büyük hacimli verilerdeki örüntüleri araştıran matematiksel algoritmaları kullandığını söylemiştir. Davis' e göre veri madenciliği hipotezleri keşfeder, sonuçları birleştirmek için insan yeteneklerini de kullanır. Davis'e göre veri madenciliği sadece bir bilim değil, aynı zamanda bir sanattır [Davis, 1999].
- DuMouchel, veri madenciliğinin geniş veritabanlarındaki birliktelikleri araştırdığını söylemiştir [DuMouchel, 1999].
- Hand, veri madenciliğini, istatistik, veritabanı teknolojisi, örüntü tanıma, makine öğrenme ile etkileşimli yeni bir disiplin ve geniş veritabanlarında önceden tahmin edilemeyen ilişkilerin ikincil analizi olarak tanımlamıştır [Hand, 1998].

• Kittler, veri madenciliğini tahmin edici anahtar değişkenlerin binlerce potansiyel değişkenden izole edilmesi olarak tanımlamıştır [Kittler, 1999].
• SAS Şirketi ise veri madenciliğini, işletme avantajına yönelik olarak önceden bilinmeyen örüntüleri keşfetmek için büyük miktarlarda verinin seçilmesi, incelenmesi ve modellenmesi süreci olarak tanımlamıştır [SAS web site, 2007].
• Birmingham Üniversitesi Bilgisayar Bilimleri web sitesinde veri madenciliğini, önceden bilinmeyen, tam ve yaralı bilginin veriden çekilmesi olarak tanımlamış; veri madenciliğinin insanların kolayca anlayabileceği bir formda veriyi keşfeden ve sunan makine öğrenme, istatistik ve görselleştirme teknikleri gibi teknikleri kullandığını belirtmiştir [School of Computer Science, 2007].
• Bransten'a göre veri madenciliği, insanın asla bulmayı hayal bile edemeyeceği trendlerin keşfedilmesini sağlar [Bransten, 1999].
• Applied Technology Group'a göre veri madenciliği, müşteri odaklı büyük veritabanlarından bilgi örüntülerini çıkaran uygun teknoloji kümelerini kullanır. Bununla birlikte, veri madenciliği tek bir teknoloji değildir. Tam tersine veriden bilgi çekmekte kullanılan araçların bir kümesidir [Applied Technology Group, 1997].

Spesifik veri madenciliği araçları; kümeleme, doğrusal regresyon, sinir ağları, bayesian ağı, görselleştirme ve ağaç tabanlı modeller gibi pek çok modeli içerir. Veri madenciliği uygulamalarında yıllar boyu istatistiksel yöntemler kullanılmıştır. Bununla birlikte, bugünün veri madenciliği teknolojisinde eski yöntemlerin tersine büyük veri kümelerindeki trend ve ilişkileri kısa zamanda saptayabilmek için yüksek hızlı bilgisayarlar kullanılmaktadır. Veri madenciliği gizli trendleri en az çaba ve emekle ortaya çıkarır [Giudici, 2003].

Veri ambarı, organizasyonun ihtiyaçları ile uyumlu büyük miktarlardaki verinin kolay erişilebilir bir yapıda tutulmasını sağlayan bilgisayar tabanlı depolama sistemleridir. 1990'lı yıllarda ortaya çıkmıştır. Veri ambarları veriyi

kullanılabilir trend, ilişki ve profillerde sınıflandırmazlar, sadece potansiyel bilgiye sahip veritabanlarıdırlar. Veride saklı bilgiyi keşfetmeyi sağlayan ise "veri madenciliği" gibi tekniklerdir. Veri ambarından veriyi çekebilmek için hangi verinin gerekli olduğunu ve bu verinin nerede olduğunu tespit etmek önemlidir. Çoğunlukla gerekli veri, farklı sistemler üzerinde ve farklı formatlardadır. Bu nedenle, ilk aşamada veri temizleme ve düzenleme işlemi gerçekleştirilmelidir. Veri ambarının yaratıcısı W.H.Inmon'a göre veri ambarı "verinin temizlendiği, birleştirildiği ve yeniden düzenlendiği, merkezi ve entegre bir depodur" [Inmon, 2002].

Sonuç olarak veri madenciliği, önceden bilinmeyen ilişki ve trendlerin bulunması için bugünün endüstrisinde yaratılan büyük miktarlardaki veriyi analiz eden bir yoldur. Yüksek güçlü bilgisayarlara ve gereken yazılımlara kolay ve düşük fiyatlarla ulaşılabilmesi bu teknolojinin işlemesini olanaklı kılmıştır. Internet ise birçok noktadaki verinin toplanmasını sağlamaktadır. "World wide web" sayesinde çok büyük miktarlarda verinin elde edilmesi sağlanmıştır. Bilgisayar üzerinde çalışma aynı zamanda emek ve zaman tasarrufunu sağlamıştır. Veri madenciliğinde kullanılan yazılım sağlayıcılardan bazıları Çizelge 2.1'de verilmiştir [Han, 2001]:

Çizelge 2.1. Veri madenciliği'nde kullanılan yazılım sağlayıcılardan örnekler

ÜRÜN ADI	**IBM Intelligent Miner**	**Oracle Darwin**	**SAS Enterprise Miner**	**Angoss Knowledge Seeker**	**NCR Teraminer Stats**
Platform	AIX 4.1, NVS, AS/400, Windows NT	Windows Windows NT	Masintosh, Windows NT, Unix	Windows, Windows NT, Unix	Windows NT, Unix
Karar Ağacı	X	X	X	X	
Sinir Ağları	X	X	X		X
Zaman Serileri	X		X		
Tahmin	X	X	X	X	
Kümeleme	X		X		
Birliktelik	X		X		
Bayesian Ağları					
Görselleştirme	X	X	X	X	X

2.1. Neden Veri Madenciliği?

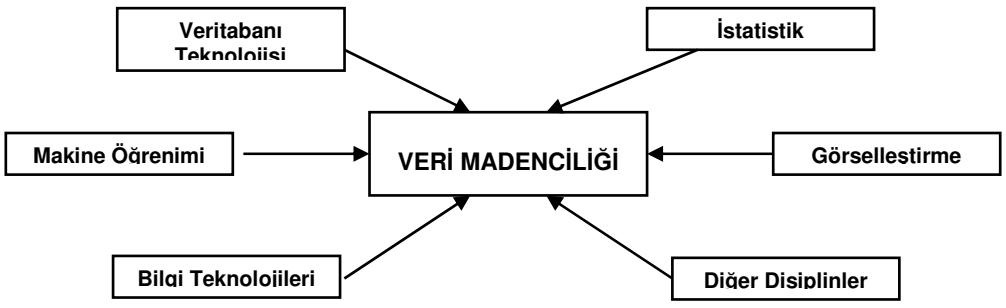
Veri madenciliği aşağıdaki 3 teknoloji tarafından desteklenir:

- Kitlesel veri toplama,
- Güçlü çok işlemcili bilgisayarlar,
- Veri madenciliği algoritmaları.

Ticari veritabanları özellikle perakendecilik sektöründe hızla gelişmektedir. Sayısal makinelere olan ihtiyaç, düşük maliyetli ve paralel çok işlemcili bilgisayar teknolojisindeki gelişmeler sayesinde karşılanmıştır.

Veri madenciliği teknolojisi, istatistik, yapay zeka ve makine öğrenimi gibi tekniklerdeki araştırmalar ile uzun bir süredir gelişme içerisindedir. Günümüzde bu tekniklerin olgunlaşması, yüksek performanslı veritabanı araçları ve veri entegrasyon çalışmaları sayesindedir. Bunlar aynı zamanda bu teknolojileri veri madenciliğine uygun hale getirmiştir [Buntine, 1996]. Bilgi teknolojilerindeki gelişmelere bağlı olarak hesaplama gücü önemli ölçüde artmış, veri toplama ve veri yönetimi daha kolay hale gelmiştir.

Şekil 2.3'de veri madenciliğine katkı sağlayan alanlar gösterilmektedir.



Şekil 2.3. Veri madenciliği sürecine katkı sağlayan alanlar

2.2. Veri Madenciliğinin Gereksinimleri

Genel olarak veri madenciliği, bir veri ambarına ve bir yazılım paketine gereksinim duyar. Diğer temel gereksinimleri aşağıdadır:

- Veriye erişilebilirlik,
- Etkin erişim yöntemleri,
- Veri problemlerinde dinamiklik,
- Etkin algoritmalar,
- Yüksek performanslı uygulama sunucusu (server),
- Sonuç dağıtımında esneklik,
- Verinin temizlenmesi.

Bir veri ambarını etkin olarak kurmadan önce verinin ne için kullanılacağı planlanmalıdır. Doğru veri bulunmaz ve verinin sınırları bilinmezse, veri madenciliği analiz işlemleri zorlaşır [Han, 2001].

2.3. Veri Madenciliğinin Adımları

Veri madenciliğinin başarılı olabilmesi için bir araçlar kümesinden ziyade bir süreç olarak kabul edilmelidir. Bu süreçte başarılı olabilmek için aşağıdaki adımlar izlenmelidir:

Adım 1: Hem istatistiksel hem de görsel olarak incelenebilecek bir veri örneği olmalıdır. Yani; toplanacak ve işlenecek veri tanımlanmalıdır.

Adım 2: En anlamlı tahmin edici ölçütlerin seçimi ve dönüştürülmesi için istatistik ve görselleştirme teknikleri kullanılmalıdır.

Adım 3: Çıktıları tahmin etmek için ölçütler modellenmelidir. Bu adımda eğilimler için hem görsel hem de nümerik olarak incelenir.

Adım 4: Modelin doğruluğundan emin olmak için sonuçlar test edilmelidir.

Adım 5: Gerektiğinde seçilen / toplanan veri model seçim sürecine odaklanmayı kolaylaştırmak için değiştirilebilir. Bazı durumlarda, boş alanların doldurulması ya da silinmesi söz konusu olabilir.

Veri madenciliğinin iteratif bir süreç olduğu unutulmamalı ve bu adımlar sürekli tekrarlanmalıdır [Han, 2001].

2.4. Veri Madenciliği'nin Amaçları ve Veri Madenciliğinde Karşılaşılan Problemler

Veri madenciliğinin amaçları öngörü, tanıma, sınıflandırma ve en iyileme olarak dört başlık altında toplanabilir [Yarımağan, 2000].

Öngörü, hangi ürünlerin hangi dönemlerde, hangi koşullarda, hangi miktarlarda satılacağına ilişkin öngörülerde bulunmak gibi tanımlanabilir.

Tanıma, aldığı ürünlerden bir müşterinin tanınması veya kullanıldığı programlar ve yaptığı işlemlerden bir kullanıcının tanınması gibi ifade edilebilir.

Sınıflandırma, bir çok parametrenin birleşimi kullanılarak, örneğin ürünlerin, müşterilerin ya da öğrencilerin sınıflandırılması olarak tanımlanabilir.

En iyileme, belirli kısıtlamalar çerçevesinde zaman, yer, para ya da ham madde gibi sınırlı kaynakların kullanımını en iyileme ve üretim miktarı, satış miktarı ya da kazanç gibi değerleri büyütme olarak tanımlanabilir.

Veri madenciliğinde karşılaşılan problemleri ise şu şekilde sıralayabiliriz:

- *Veri tabanı boyutu:* Örneklemin büyük olması, örüntülerin gerçekten var olduğunu göstermesi açısından bir avantajdır, ancak bu durumda elde edilen örüntü sayısı çok büyümektedir. Bu yüzden veri tabanı boyutu veri madenciliği sistemleri için önemli problemlerden biridir. Bu sorunu çözmek için uygulanabilecek yöntemlerden bazıları: veri madenciliği sisteminin sezgisel/buluşsal bir yaklaşımla arama uzayını taraması, örneklemin yatay ve dikey boyutta indirgenmesi olarak sayılabilir [Oğuz, 2000].
- *Gürültülü veri:* Veri girişi ya da veri toplanması sırasında oluşan sistem dışı hatalara gürültü adı verilir. Eğer veri kümesi gürültülü ise sistem bozuk veriyi tanımalı ve ihmal etmelidir. Quinlan'ın gürültünün sınıflama üzerindeki etkisini araştırmak için yapmış olduğu çalışma sonucunda, etiketli öğrenmede etiket üzerindeki gürültü, öğrenme algoritmasının performansını doğrudan etkileyerek düşmesine neden olmuştur [Quinlan, 1986].
- *Eksik veri:* Örneklem kümesindeki kayıtların eksik olması ya da bazı kayıtlar için bazı niteliklerin veya nitelik değerlerinin olmamasıdır. Bu eksiklik; hatalı ölçüm araçlarından, veri toplama sürecinde deneyin tasarımında yapılan değişiklikten ya da birbirine benzer ancak özdeş olmayan veri kümelerinin birleştirilmesinden kaynaklanıyor olabilir.
- *Artık veri:* Örneklem kümesi, eldeki probleme uygun olmayan, gereksiz nitelikler içerebilir, bunlara artık veri denir. Artık nitelikleri elemek için geliştirilmiş algoritmalar özellik seçimi olarak adlandırılır. Özellik seçimi, hedef bağlamı tanımlamak için yeterli ve gerekli olan niteliklerin küçük bir alt kümesinin seçilmesidir. Özellik seçimi yalnızca arama uzayını küçültmekle kalmayıp, sınıflama işleminin kalitesini de arttırır [Almuallim ve Dietterich, 1991].
- *Null değerler:* Niteliğin bilinmeyen ve uygulanamaz bir değere sahip olduğunu gösterir. Veri kümelerinde yer alan null değerleri için çeşitli çözümler söz konusudur. Bunlar; null değerli kayıtlar tamamıyla ihmal edilebilir, null değerler yerine olası bir değer atanabilir. Bu değerler o nitelikteki en fazla frekansa sahip bir değer veya ortalama bir değer olabilir,

varsayılan bir değer olabilir, null değerin kendisine en yakın değer olabilir [Quinlan, 1986].

• *Dinamik veri:* İçeriği sürekli olarak değişen veri tabanlarına dinamik veri tabanları denir. Bu veri tabanlarında veri madenciliği örüntüleri elde edildikten sonra değişen verilerin örüntülere yansıması gerekmektedir. Bu işlem, veri madenciliği uygulamasının ürettiği örüntüleri değişen veriye göre, gereken yerlerini günleme yeteneğine sahip olmasını gerektirir.

2.5. Veri Madenciliği Faaliyetleri ve Kullanıcıları

Veri madenciliği faaliyetleri, aşağıda tanımlanan 3 farklı kullanıcı sınıfı tarafından yürütülür;

• Uzmanlar; üst düzey kavrayışa gereksinim duyarlar ve bilgisayarda diğer gruplardan daha az zaman geçirirler.
• Son kullanıcılar; satış elemanları, pazar araştırmacıları, bilim adamları, mühendisler, doktorlar vb. dir.
• Analistler; finansal analistler, istatistikçiler, danışmanlar ve veritabanı tasarımcılarıdır.

Tanımlanan kullanıcılar; bölümlü, stratejik ve sürekli veri madenciliği olmak üzere 3 çeşit veri madenciliği faaliyeti gerçekleştirirler.

Bölümlü veri madenciliğinde, veriye spesifik bir açıdan bakılır. Amaç; bu veri kümesini anlamak ve yeni karar destek sistemine yönelik tahmin için kullanmaktır. Analistler genellikle bölümlü veri madenciliği çalışmalarını yürütür.

Stratejik veri madenciliğinde; spesifik ölçütleri anlamak için daha geniş veri kümeleri incelenir. Stratejik veri madenciliğinde, “destek nereden geliyor?” ya da “insanlar nasıl gruplandırılabilir?” soruları cevaplandırılmaya çalışılır.

Sürekli veri madenciliğinde, verilen bir zaman diliminde dünyanın nasıl değiştiği ve bu değişimi etkileyen faktörlerin neler olduğu araştırılır [Yuarn, 2003].

2.6. Veri Madenciliğinin Kullanım Alanları

Veri madenciliğinin ana kullanım alanları,

- Sigortacılık,
- Sağlık,
- Finans,
- Kredi kartı şirketleri,
- Telefon şirketleri,
- Perakende ve pazarlama süreçleridir.

Veri madenciliğinin kullanımına otellerde, kataloglarda, süpermarketlerde, kiralık araba, Internet ve E-Ticaret uygulamalarında ve havayolu endüstrisinde de rastlanmaktadır. Bu endüstriler veri madenciliğini, müşteri, reklam ya da kredi kartı yoluyla müşteri tercihlerini belirlemek için kullanırlar. Bu veri ile belirli müşterilere hangi promosyon reklamlarının gönderileceğine karar verilir. Bu endüstrilerin veri madenciliğini kullanma nedeni, müşteri, ürün ve işlem bazında büyük miktarlardaki veriyi anlama ve bu bilgiyi kullanma ihtiyacıdır. Davranış örüntülerini, tercih örüntülerini ve verinin çokluğundan dolayı insan beyni tarafından algılanamayacak ilişkileri belirlemek için kullanırlar.

Yeni bir teknoloji ortaya çıktığında bu teknolojinin belirli bir ortama uygun olup olmadığı şüpheleri her zaman ortaya çıkar. Her şirket problemini cevaplayan bir teknoloji olmadığından, veri madenciliği de bazı durumlar için yararlıdır. Veri madenciliği aşağıdaki karakteristiklere sahip problemlerin çözümünde daha çok tercih edilir:

- Büyük miktarlarda veriye erişilebildiği zaman,

- Veri birçok değişkene sahipse,
- Veri karmaşık, çok değişkenli ve doğrusal değil ise,
- Çıktıları ya da davranışı tahmin etmek gerekiyorsa,
- Henüz anlaşılmayan birliktelik ve ilişkileri bulmak gerekiyorsa.

Veri madenciliğinin kullanıldığı alanlarda başarılı çözüme ulaştığı örnek problemler aşağıdadır;

- Benzer satın alma örüntüleri gösteren müşterilerin doğru segmentlerde ifade edilmesi,
- Bireysel İlişki Yönetimi için etkin bir şekilde müşteri profillerinin çıkarılması,
- Maillere gelen yanıt oranın arttırılması,
- Hangi müşterilerin daha sadık ve hangilerinin belirli promosyonlara cevap verme olasılığının daha yüksek olduğunu tanımlayacak müşteri profillerinin oluşturulması,
- Bir müşteriyi rakip şirketi tercih etmeye iten faktörlerin neler olduğunu anlamak,
- Satın alma örüntülerini, ödemeleri ve tepki oranlarını etkileyen faktörleri keşfetmek,
- Bir kredi kartı işleminin ya da sigorta talebinin sahtekârlıkla sonuçlanıp sonuçlanmayacağını tahmin etmek,
- Kredi kartı müşterilerinin verilen bir zaman çerçevesinde hesaplarını başka bir bankaya nakledip nakletmeyeceklerini tahmin etmek,
- Geçmişlerine ve karakteristiklerine bakarak müşterilerin gelecekteki davranışlarını tahmin etmek,
- Tıp merkezlerine ve sigorta şirketlerine hangi prosedürlerin birleşiminin en çok istenen çıktıları üreteceğini belirleyerek maliyet yönetiminde yardım etmek [Han, 2001].

2.7. Veri Madenciliğinin Faktörleri

Veri madenciliği teknolojisinin temel bileşenleri; istatistik, yapay zeka ve makine öğrenimi gibi alanlardaki araştırmalar ile uzun bir süredir gelişme içerisindedir. Günümüzde, yüksek performanslı ilişkisel veritabanı araçları ve veri entegrasyon çalışmaları bu teknolojileri veri madenciliğine uygun hale getirmiştir.

Veri madenciliğinin farklı faktörlerini anlamak için, veri madenciliği uygulamaları ile operasyon, teknik ve algoritmaları ayırmak gerekir. Bunların ayırımı Çizelge 2.2'de verilmiştir.

Çizelge 2.2. Veri madenciliğinin faktörleri

UYGULAMALAR	Müşteri Segmantasyonu Müşteri Tutma Sahtekarlık Tespiti Kredi Kontrol Web Sitesi Analizi
OPERASYONLAR	Sınıflama ve Tahmin Kümeleme Birliktelik Analizi
TEKNİKLER	Sinir Ağları Karar Ağaçları k- En Yakın Komşu Bayesien Kümeleme Analizi

Veri madenciliği teknolojisini kullanan bir uygulama bir ya da daha fazla veri madenciliği operasyonunu yürütür. Her operasyon; kompleks bir veri kümesindeki örüntü ve trendleri ayırmak için farklı bir yolu yansıtır [Han, 2001].

2.8. Veri Madenciliğinde Kullanılan Yaklaşımlar

Veri madenciliğinde sınıflandırma ve tahmin, kümeleme, birliktelik analizi ve ardışık analiz olmak üzere üç tür yaklaşım kullanılmaktadır.

2.8.1. Sınıflandırma ve tahmin

Sınıflandırma, önceden belirlenen veri sınıflarına göre model geliştirilip bu modelin sınıflandırma için kullanıldığı iki adımlı bir süreçtir.

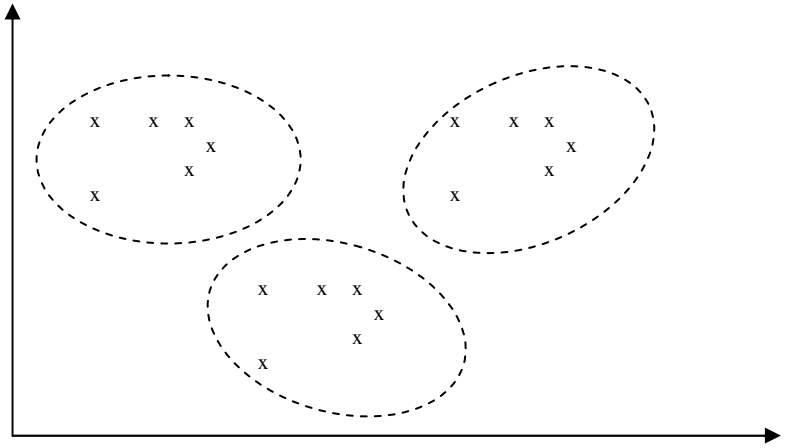
Veri kümeleri önceden tanımlanmış bir sınıfa göre etiketlenmiştir. Bu veri kümeleri eğitim ve test veri kümesini oluşturmak üzere ikiye ayrılır. Eğitim veri kümesinden hareket edilerek test veri kümesindeki verilerin hangi sınıfa ait olduğu bulunur.

Sınıflandırma ve tahmin yöntemlerinde aşağıdaki karşılaştırma kriterleri önerilmiştir [Han, 2001]:

- *Tahmin doğruluğu (Predictive Accuracy):* Modelin yeni ya da daha önceden bilinmeyen bir verinin sınıf etiketinin doğru olarak tahmin edilme yeteneğidir.
- *Hız (speed):* Modelin genelleştirilmesindeki hesaplama maliyetine karşılık gelir.
- *Sağlamlık (Robustness):* Modelin, gürültülü ya da kayıp veri verildiğinde doğru tahmin yapma yeteneğidir.
- *Ölçeklendirilebilirlik (Scalability):* Büyük miktarlarda veri verildiğinde, modeli etkin bir şekilde kurma yeteneğidir.
- *Yorumlanabilirlik (Interpretability):* Model tarafından sağlanan anlaşılma seviyesidir.

2.8.2. Kümeleme

Kümeleme, nesneleri, benzer nesne sınıflarına gruplandırma işlemidir. Kümeleme algoritması, birbirine benzer nesnelerin aynı kümede, birbirinden farklı nesnelerin ayrı kümelerde yer almasıyla gerçekleştirilir [Chen, 2001]. Sınıflamanın tersine kümeleme önceden tanımlanacak hedef bir değişkene gereksinim duymaz. Şekil 2.4'de örnek bir kümeleme görülmektedir.



Şekil 2.4. Veri kümeleme

2.8.3. Birliktelik analizi ve ardışık analiz

Birliktelik analizi, bir veri kümesindeki kayıtlar arasındaki bağlantıları arayan yönlendirilmemiş veri madenciliği şeklidir. Nitelikler arasındaki bütün kombinasyonlar çıkarılarak, her niteliğin bütün değerleri bütün kombinasyonlarda denenerek örüntüler keşfedilmeye çalışılır [Agrawal ve ark., 1993]. Birliktelik analizi, pazar sepet analizi, katalog tasarımı, mağaza ürün yerleşim planı vb. alanlarda yaygın olarak kullanılır.

Ardışık analiz ise birbirleri ile ilişkisi olan ancak birbirini izleyen dönemlerde gerçekleşen ilişkilerin tanımlanmasında kullanılır [Akpınar, 2000]. Bazen ayrı bir veri madenciliği operasyonu olarak değerlendirilebilir.

2.9. Veri Madenciliği'nde Sınıflandırma Problemleri için Kullanılan Yöntemler

Veri madenciliğinde sınıflandırma operasyonları bir dizi teknik ya da yöntem kullanılarak gerçekleştirilir. Her teknik için çeşitli algoritmalar kullanılmaktadır. Bunlardan bazıları aşağıda tanımlanmıştır.

2.9.1. Karar ağacı ile sınıflandırma yöntemi

Karar ağacı, her iç düğümün (internal node) bir nitelik üzerindeki testini ve her dalın bu testin çıktısını gösterdiği, her yaprak düğümünün (leaf node) ise sınıfları ya da sınıf dağılımlarını temsil ettiği ağaç yapılı akış şemasıdır. En üstteki düğüm ise kök düğüm (root node) olarak adlandırılır [Han, 2001].

Tahmin edici ve tanımlayıcı özelliklere sahip olan karar ağaçları, veri madenciliğinde aşağıdaki sebeplerden dolayı yaygın olarak kullanılmaktadır [Han, 2001] :

- Kuruluşlarının ucuz olması,
- Yorumlanmalarının kolay olması,
- Veri tabanı sistemleri ile kolayca entegre edilebilmeleri,
- Güvenilirliliklerinin daha iyi olması vb.

nedenleri ile sınıflandırma modelleri içerisinde en yaygın kullanıma sahiptir.

2.9.2. Bayes ile sınıflandırma yöntemi

Bayes sınıflandırıcılar istatistikseldir. Verilen bir üyeliğin belli bir sınıfa ait olup olmaması gibi sınıf üyeliği olasılıklarını tahmin ederler.

2.9.3. Naive Bayes ile sınıflandırma yöntemi

Naive Bayes; hem tahmin edici hem de tanımlayıcı bir sınıflama tekniğidir. Her ilişkide koşullu bir olasılık türetmek için bağımlı ve bağımsız değişkenler arasındaki ilişkiyi analiz eder.

2.9.4. k- en yakın komşuluk ile sınıflandırma yöntemi

En yakın komşuluk (k-NN), sınıflama modelleri için uygun olan tahmin edici tekniklerden biridir.

Modelde yeni bir vaka ortaya çıktığında; algoritmanın tüm veriyi inceleyerek buna en çok benzeyen vakaların bir altkümesini oluşturduğu ve onları çıktıyı tahmin etmek için kullandığı benzerlik ile öğrenmeye dayanan bir yöntemdir.

2.9.5. Vaka tabanlı nedenleşme ile sınıflandırma yöntemi

Sınıflandırmak için yeni bir vaka verildiğinde, vaka tabanlı nedenleyici benzer bir eğitim vakasının olup olmadığını kontrol eden, bulduğunda aynı çözüm uygulayan, böyle bir vaka bulunamazsa, yeni vakanın bileşenlerine benzeyen eğitim vakalarını araştıran bir yöntemdir [Han, 2001].

2.9.6. Genetik algoritma ile sınflandırma yöntemi

Genetik ortamın, programlama teknikleri kullanılarak kodlanması genetik algoritma olarak adlandırılır. En iyinin korunumu ve doğal seçilim ilkesinin benzetim yoluyla bilgisayarlara uygulanması ile elde edilen bir arama yöntemidir. Genetik algoritmaların bugünkü biçimi ilk olarak Holland tarafından 1975 yılında ortaya konulmuştur [Nabiyev, 2003].

Genetik algoritma (GA), bir problemin olası çözümlerinden oluşan sabit büyüklükte bir çözüm grubu içinde tekrarlanarak yürütülen işlemlerden oluşan

bir yöntemdir. Çözüm grubu popülasyon, çözümlerin kodları kromozom, tekrarlanan her basamağa ise jenerasyon adı verilir.

Basit bir genetik algoritma şu adımlardan oluşur:

- Olası çözümlerin kodlandığı bir popülasyon oluşturulur. Popülasyonda bulunacak birey sayısı için bir standart yoktur. Birey sayısı belirlendikten sonra probleme bağlı olarak kromozomların kodlanması gerekir.
- Popülasyondaki her kromozomun ne kadar iyi olduğunu bulmak amacıyla kullanılan fonksiyona uygunluk fonksiyonu denir. Uygunluk fonksiyonu, kromozomları problemin parametreleri haline getirerek bu parametreler yardımıyla hesaplama yapmaktadır. Çoğu zaman genetik algoritmaların başarısı bu fonksiyona bağlıdır.
- Uygunluk fonksiyonu hesaplandıktan sonra en iyi kromozomların seçim işlemi yapılır. Bunun için rulet tekeri seçimi, sıralama seçimi, sabit durum seçimi gibi birçok yöntem vardır. Yeniden kopyalama genlerindeki genetik bilginin birinden diğerine geçmesi işlemine benzediği için çaprazlama olarak adlandırılır. Çaprazlama popülasyonda çeşitliliği yani, iyi özelliklerin bir araya gelmesini kolaylaştırarak en iyiye yaklaşmayı sağlar. Mutasyon kromozomun bir parçasının dışarıdan değiştirilmesidir.
- Yeni kromozomlara yer açmak için eski kromozomlar çıkartılarak sabit bir büyüklükte popülasyon sağlanır.
- Tüm kromozomların uygunlukları tekrar hesaplanır ve yeni popülasyonun başarısı bulunur.
- İşlemler tekrarlanarak verilmiş zaman içerisinde daha iyi olan yeni kuşakların oluşturulması gerçekleştirilir.
- Sonuçta popülasyonların hesaplanması sırasında en iyi bireyler bulunduğundan çözüm elde edilmiş olur [Ergüneş, 2004].

Genetik algoritmalar, sınıflandırma problemlerinde olduğu kadar diğer optimizasyon problemlerinde de sık kullanılır [Gen ve Cheng, 2000].

2.9.7. Kaba küme teorisi ile sınıflandırma yöntemi

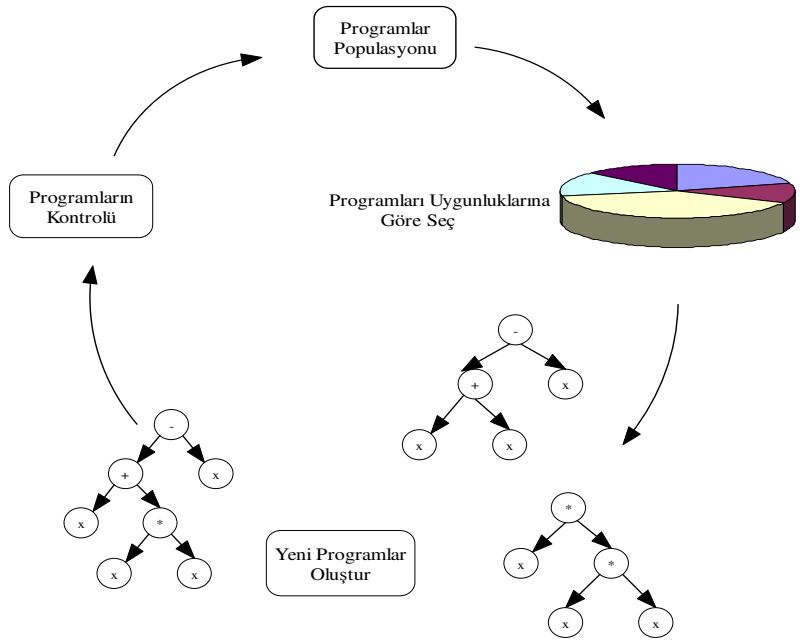
Kaba küme teorisi sınflandırmada, kesin olmayan ya da gürültülü veri içindeki yapısal ilişkileri keşfetmek için kullanılmaktadır. Kesikli değerli niteliklere uygulanır. Sürekli değerli nitelikler, kaba küme uygulanmadan önce kesiklileştirilmelidir [Pawlak, 1982]..

Bu tezde, kaba küme teorisi kullanılarak nitelik indirgemesi yapılmıştır.

2.9.8. Genetik programlama ile sınıflandırma yöntemi

Genetik programlama (GP), zor problemlerin çözümünde kullanılan evrimsel bir çözüm tekniğidir. GP bireyleri genellikle ağaç yapıları ile doğrusal olmayan bir şekilde gösterilir ve işlem görürler. Yakın zamanda GP'nın bir çok doğrusal gösterime sahip biçimleri için farklı yapılar önerilmiştir. Bunlardan bazıları gramere dayalı evrim, doğrusal genetik programlama ve gen denklem programlamadır (GEP). Bu farklı yapılardaki gösterimlerin amacı GP'nın performansını artırmak, aynı zamanda da programlanmasını kolaylaştırmaktır. Bu GP yaklaşımlarının ortak özelliği, doğrusal olmayan GP yapısının, doğrusal bireyler olarak ifade edilmesidir.

Genetik algoritmadan en önemli farklılığı, çözüm dizisinin değişken uzunlukta olma özelliğini taşımasıdır. Genetik algoritma, sayılardan oluşan diziler üzerinde işlerken genetik programlama bilgisayar programlarını bireyler olarak ele almaktadır. Bireylerin ağaç yapısındaki gösterimleriyle birlikte çaprazlama ve mutasyon operatörleri genetik algoritmadan farklı olarak uygulamaya geçirilir. İkisi arasındaki temel fark, bireylerin gösterimi ve uygunluk fonksiyonunun hesaplanmasıdır. Genetik programlamada, genetik operatörlerle yeni programların oluşturulmasını kolaylaştırmak için programlar ağaç yapısı ile ifade edilirler. Genetik programlamanın genel işleyiş döngüsü Şekil 2.5'de gösterilmiştir.



Şekil 2.5. Genetik Programlama Döngüsü

Genetik programlamada, bir popülasyon bireyi hiyerarşik olarak yapılandırılmış fonksiyonlardan ve terminallerden oluşan bir programdır. Fonksiyonlar ve terminaller, önceden belirlenmiş fonksiyon ve terminaller kümesinden seçilir. Örneğin, bir fonksiyon kümesi temel aritmetik operatörlerden (F= {+,-,*,/}) oluşabilir. Fonksiyon kümesi probleme bağlı olarak farklı mantıksal operatörleri de içerebilir. Terminal kümesi ise, fonksiyonların argümanlarını oluşturacak elemanları içerir.

Genetik programlamada fonksiyon ve terminal kümelerinin iki önemli özelliği sağlamaları gerekmektedir.

- Kapalılık özelliği: Fonksiyon kümesindeki her bir fonksiyon diğer fonksiyonlar tarafından türetilmiş olmalı ya da terminaller kümesindeki argümanların tamamını işleyebilmelidir. Örneğin matematiksel fonksiyonlar, 0'a bölünmeden, negatif logaritmadan, negatif karekökten korunmalıdır.
- Yeterlilik özelliği: Fonksiyon ve terminal kümesinden seçilen elemanlar, ele alınan problemin çözümü için yeterli olmalıdır. Yani elde edilen sonuç, problemin çözümü için uygun ve problemi ifade edebilir nitelikte olmalıdır.

Ayrıca, elde edilen programların uygunluk fonksiyonlarının kolaylıkla değerlendirilebilecek biçimde belirlenmesi de gerekmektedir.

GP'ye uygulanan operatörler temelde genetik algoritma operatörlerinin aynısıdır. Ancak uygulamada, birey yapısı farklı olduğu için bazı farklılıklar göstermektedir.

- Üreme: Üremenin farklı uygulamaları söz konusudur:

a. Uygunluk oranlı üreme yöntemi,

b. Sıra seçim yöntemi,

c. Turnuva seçim yöntemi.

- Çaprazlama: Uygunluk fonksiyonuna bağlı olarak seçilen iki program rasgele belirlenmiş çaprazlama noktalarından çaprazlanırlar.
- Mutasyon: Mutasyon iki farklı biçimde gerçekleştirilmektedir:

d. Rasgele bir terminal veya fonksiyon seçilir ve yine rasgele seçilmiş bir başka fonksiyon ya da terminalle değiştirilir.

e. Rasgele seçilmiş bir dal, yine rasgele oluşturulmuş bir dalla yer değiştirilir [Özbakır, 2004].

2.10. Literatürde Evrimsel Algoritmalar Kullanılarak Sınıflandırma Amaçlı Yapılmış Olan Çalışmalar

Evrimsel algoritmalar doğal seçim ve genetik mekanizmasıyla çalışan rasgele arama prosedürleri olarak tanımlanır. Evrimsel algoritmaların genetik algoritma, genetik programlama, evrimsel stratejiler (ES) ve evrimsel programlama (EP) gibi farklı tipleri vardır. Bu algoritmalar aynı kavramlara dayanır fakat çözümler ve bir sonraki jenerasyonu oluşturmak için kullanılan operatörler farklı yollardan gösterilir.

Genetik programlama, bireyleri programlardan oluşan bir popülasyona genetik algoritma operasyonlarının uygulanmasıdır [Koza, 1992]. GP, GA'nın bir uzantısıdır. Aralarındaki temel farklılık bireylerin gösterimi ve uygunluk fonksiyonunun hesaplanmasıdır. Yapay zeka tabanlı teknikler altında sınıflandırılmış genetik programlama temelli yaklaşımlar, kompleks veri madenciliği yaklaşımlarında oldukça popülerleşmiştir. Bunlar genellikle tahmin problemlerine uygulanmıştır. Fakat sınıflandırma problemleri içinde oldukça uygundur. Çünkü GP veri sınıflandırmada iki uç nokta arasında bir çözüm yapısı sağlamaya oldukça elverişlidir. Bu uç noktalardan biri yapay sinir ağlarıdır. Yapay sinir ağları doğru sınıflandırmalar sağlar, fakat kapalı kutu olarak çalışırlar. Bir diğer ekstrem nokta C4.5 gibi oldukça kompleks karar ağaçları üreten sınıflandırma algoritmalarıdır.

GP niteliklerin bir çok farklı kombinasyonlarını üretebildiği için sınıflandırmada oldukça kullanışlıdır.

Freitas, veri madenciliğinde genetik programlama ve genetik algoritma gibi evrimsel algoritmaların bir literatürünü sunmuş, sınıflandırma tipli problemlerde odaklanmıştır [Freitas, 2002].

Freitas, sınıflandırma için bir GP yapısı önermiş ve kural indirgemede genelleştirmiştir [Freitas, 1997].

Carvalho ve Freitas, sınıflandırma kurallarının keşfi için hibrit karar ağacı/genetik algoritma yaklaşımı önermişlerdir [Carvalho ve Freitas, 2002].

Takac, genetik algoritmaların hücresel paralel modeliyle GP algortimasını birleştirmiştir [Takac, 2003].

Zhou ve arkadaşları, Ferreira tarafından bir lineer genetik programlama yaklaşımı olarak önerilen Gen Programlama (GEP) ya göre sınıflandırma

kurallarını içeren yeni bir yaklaşım sunmuşlardır. Literatürden alınmış 12 veri kümesi üzerinde yaklaşımlarını test etmişlerdir [Zhou ve ark., 2003].

De Falco ve arkadaşları, sınıflandırma kurallarının otomatik keşfini yapabilen bir genetik programlama yapısı geliştirmişlerdir [De Falco ve ark., 2002].

Veri madenciliğinin ana uygulama alanlarından biri veri miktarının oldukça fazla olması nedeniyle klinik tıp alanındadır [Kusiak ve ark., 2005; Alonso ve ark., 2002; Chen ve ark., 1996]. Yapay sinir ağları [Pendharkar ve ark., 1999; Santos ve ark., 2000; Pendharkar, 2001], karınca kolonisi optimizasyonu [Parpinelli ve ark., 2001; Parpinelli ve ark., 2002], yapay bağışıklık sistemleri [Goodman ve ark., 2002], genetik programlamanın değişik biçimleri [Eggermont ve ark., 1999] gibi bir çok farklı teknikler medikal sınıflandırma problemlerine uygulanmıştır. Lavrac tıpta uygulanabilecek bazı veri madenciliği tekniklerinin analizlerini sunmuştur [Lavrac, 1999].

Sınıflandırmada uygulanan diğer teknikler arasında evrimsel algoritmalar medikal alanda oldukça ümit verici bir yaklaşım olarak ortaya çıkmaktadır. Araştırma alanında, medikal veri kümeleri için GA ve GP nin bazı son uygulamaları vardır. Fidelis ve arkadaşları medikal alanlardan karşılaştırılabilir IF-THEN kurallarının keşfi için GA ya dayalı esnek kromozom kodlama yapısı önermişlerdir [Fidelis ve ark., 2000].

Bojarczuk ve arkadaşları, göğüs hastalığının teşhisi için bir GP yaklaşımı geliştirmişlerdir ve 2004 yılında da medikal veri kümelerinde sınıflandırma kurallarının keşfi için yeni bir kısıtlandırılmış sentaks (constrained-syntax) genetik programlama algoritması önermişlerdir [Bojarczuk ve ark., 2000; Bojarczuk ve ark., 2004].

Hassan ve Tazaki, medikal veri tabanlarından bilgi kuralları çıkarımı için kaba küme yaklaşımı ve genetik programlama algoritmasını birleştirmişlerdir [Hassan ve Tazaki, 2001].

Bramier ve Banzhaf, yeni bir lineer genetik programlama yaklaşımı önermişler ve performansını GP ile karşılaştırmışlardır [Bramier ve Banzhaf, 2001].

Ngan ve arkadaşları, Bayesian ağlarını kullanan öğrenen bir sistem geliştirmişlerdir [Ngan ve ark., 1999].

Tan ve arkadaşları, medikal teşhiste kullanılabilmesi için sınıflandırma kuralları çıkaran iki aşamalı hibrit bir evrimsel sınıflandırma tekniği geliştirmişlerdir. İlk aşamada, GP ve GA evrimsel algoritmaları kullanılarak iyi aday kuralları belirlenmiş, ikinci aşamada doğru ve karşılaştırılabilir kural kümeleri yapısı elde etmek için bu aday kuralların sayısı ve farklı sıraları birleştirilmiştir [Tan ve ark., 2002].

Mugambi ve arkadaşları, bulanık ve karar ağacı yapılarından polinomial oluşturarak çok değişkenli karar ağacı geliştirmişlerdir [Mugambi ve ark., 2004].

Baykasoğlu ve Özbakır 2007 yılında veri madenciliği problemleri için yeni bir sınıflandırma tekniği olarak MEPAR-miner algoritmasını geliştirmişlerdir [Baykasoğlu ve Özbakır, 2006]. MEPAR-miner algoritması, sınıflandırma kurallarının keşfi için orijinal MEP Algoritması'nın [Oltean ve Dumitrescu, 2002; Oltean ve Grosan, 2004] modifiye edilmesiyle oluşturulmuştur. Orjinal MEP kromozom gösteriminde, fonksiyon ve terminal kümeleri yeniden düzenlenerek tekrar tasarlanmış ve sınıflandırma kurallarını gösteren mantıksal ifadeler oluşturulmuştur. Kural kümesinin ana yapısı aşağıdaki gibidir:

IF antecedent $_1$ **THEN** class $_1$
ELSE IF antecedent $_2$ **THEN** class $_2$
…
…
ELSE class $_{default}$

• Varsayılan sınıf yapısı (Default class structure): Kural listesinin değerlendirilmesine en üstteki kuraldan başlanır ve kuralla eşleşen örnekler belirlenir. Yeni örnek kural listesindeki kuralların hiçbirisiyle eşleşmiyorsa varsayılan sınıf olarak yapılandırılır. MEPAR-miner algoritmasında varsayılan sınıf veri kümesinde en çok kullanılan sınıf olarak etiketlenmiştir.

• Fonksiyon ve terminal kümeleri: MEPAR-miner algoritmasında kromozom yapısı, terminal kümesinden ve mantıksal fonksiyonları içeren fonksiyon kümesinden oluşur. Terminal kümesi, nitelikleri, ilişkisel operatörleri ve nitelik değerlerini içerir. Verilen bir sınıflandırma probleminde terminal genlerin uzunluğu nitelik sayısına eşittir. Herbir terminal gene, rasgele bir nitelik atanır. Nitelik atanmasından sonra niteliğin tipine göre (kategorik ya da sürekli) ilişkisel operatörler atanır.

Mantıksal operatörler tarafından yapılandırılmış fonksiyon kümesi kromozomun ikinci parçasını oluşturur. AND, OR operatörleri iki argument işaretçiye sahipken, NOT operatörü tek bir argument işaretçiye sahiptir.

Kromozom yapısında kullanılan fonksiyon ve terminal kümelerinin yapısı aşağıdaki gibidir.

X_i	i^{nci} nitelik
İlişkisel Operatör	Niteliğin Tipi
=	Kategorik Nitelikler
≤, ≥	Sürekli Nitelikler
V_{xi}	i^{nci} niteliğin değeri
Terminal Kümesi	$\{x_0 – RO - V_{xo}, x_1 – RO - V_{x1}, .., x_n – RO - V_{xn}\}$
Fonksiyon Kümesi	{AND, OR, NOT}

- Algoritma tipi: Basit genetik algoritma kullanılmıştır. Bu algoritmaya göre, popülasyondaki en iyi mantıksal ifadeli kromozom bir sonraki jenerasyona değişmeksizin kopyalanmıştır (elitizasyon).
- Çaprazlama: Tek nokta çaprazlama kullanılmıştır.
- Mutasyon: Kromozomdaki herbir sembol (terminal işaretçi, fonksiyon, fonksiyon işaretçi) mutasyon operatörünün hedefi olabilir. Kromozomdaki bazı semboller mutasyonla önceden tanımlanmış mutasyon olasılığına göre değişikliğe uğrarlar. Kromozom içerisinde rasgele mutasyon noktası ya da noktaları belirlenir. Eğer mutasyona uğrayacak nokta terminal gen ise terminal işaretçiler bir başka ilişkisel operatörle yer değiştirir ve nitelik değeri de nitelik değer kümesi içinde bir başka değerin değerini alır. Eğer mutasyon noktası bir fonksiyon gen ise mantıksal fonksiyon bir başka mantıksal fonksiyonla yer değiştirir.

Bu tezde geliştirilen algoritma, MEPAR-miner algoritmasının kromozom yapısını dikkate almaktadır. Bu yüzden bu algoritma daha ayrıntılı anlatılmıştır.

Veri madenciliğinde sınıflandırmada yukarıda belirtilen çalışmaların yanısıra karma olarak yapılan diğer çalışmalarda aşağıda kısaca belirtilmiştir.

- Kaba küme teorisi-Yapay sinir ağları [Stepaniuk ve Kierzkowska, 2003].
- K en yakın komşu-C4.5 algoritması [Yahia ve Ibrahim, 2003].
- Bilgi teorisi-Küme teorisi [Liu ve Lu, 2001].
- Naive bayes-Karar ağacı [Zhipeng ve ark., 2003].
- Kümeleme-Yapay sinir ağları [Hsieh, 2005].
- Kaba küme teorisi- Bulanık küme [Pal ve Pabitra, 2004].
- Genetik programlama-Karar ağacı [Marmelstein ve Lamont, 1998].
- Genetik algoritma-Tabu arama [Zhong ve ark., 2004].
- Yapay sinir ağları-Genetik algoritma [Sumathi ve ark., 2001].
- K en yakın komşu-Genetik algoritma [Anand ve Hughes, 1998].

- Karar ağacı-Yapay sinir ağları [Feng ve Mcclean, 2001; Paprzycki ve ark., 2004; Pan ve ark., 2003].
- Naive Bayes-Karar ağacı [Ali, 2004].
- Uzman Sistemler-Yapay sinir ağları [Ciesielski ve Palstra, 1996].
- Bulanık Küme-Karar ağacı [Chen ve ark., 2003].
- Kaba küme teorisi-Yapay sinir ağları-Genetik algoritma [Bhaskar ve Kamath, 2004].

Bu tezde de evrimsel algoritmalar kullanılmıştır.

2.11. Veri Madenciliği'nde Nitelik İndirgeme Yaklaşımı

Veri madenciliği uygulamalarının en önemli noktalarından biri toplanan verinin ilgilenilen konu ile ilgili olması durumudur. Milyonlarca kaydı taşıyan niteliklerin çok azının karar kuralı olarak karşımıza çıkması, veri madenciliği uygulaması yapılmadan önce verinin hazırlanmasının gerekliliğini ortaya koymaktadır. Dolayısı ile gereksiz veya fazlalık olarak belirlenen niteliklerin atılması ile sınıflandırma algoritmalarının performansları kolaylıkla arttırılabilmektedir. Bu yüzden veri azaltılması, oldukça gerekli bir adımdır.

Nitelik çıkartma, nitelik seçimi ve nitelik kurulumu veri azaltılmasında etkin yaklaşımlardır.

Nitelik çıkartma yeni nitelik kümesinin, orijinal nitelik kümesinden bazı fonksiyonel işlemlerle çıkartılması işlemidir.

Nitelik seçimi, nitelik alt kümesinin orijinal nitelik kümesinden seçimi işlemidir.

Nitelik kurulumu ise, nitelikler arası ilişkiler hakkında kayıp bilgiyi keşfetme ve ek nitelikler yaratma ile nitelik uzayını arttırma işlemidir. Veri azaltılmasında kullanılan bu etkin yaklaşımların temel hedefleri: veri boyutunu azaltmak, ilgili

veri üzerine yoğunlaşarak veri kalitesinin ve böylece öğrenme zamanı ve tahmin kesinliğinin arttırılmasını sağlamaktır [Huan ve Yu, 2003].

2.11.1. Boyut indirgeme yöntemleri

Bazı durumlarda yüksek boyutlu karmaşık olgular çok az ve basit değişkenlerle ifade edilebilmektedir. Dolayısı ile modelleme yapılırken indirgeme işleminin yapılması, gereksiz nitelikler ile sınıflandırma yapılmasını engelleyecektir.

Boyut indirgemede, matematiksel olarak durum şu şekilde özetlenebilir; p-boyutlu rassal değişken $\mathrm{x} = (x_1,....x_p)^T$ olsun. Bunun için orijinal verinin içeriğini koruyacak şekilde $\mathrm{s} = (s_1,....s_k)^T$ (k<p) düşük boyutlu gösterimini herhangi bir kritere göre bulmaya boyut indirgeme denilmektedir. s'nin bileşenlerine bazen gizli bileşen denir. p'nin ise birçok ismi olduğu gibi genelde "değişken" ya da "nitelik" de denilmektedir [Fodor, 2002].

Boyut indirgeme problemleri 3 kategoride değerlendirilebilir.

Yüksek boyut indirgeme metotları, binlerce bileşenden oluşan boyutların azaltılması için kullanılır. Tipik metotlar: temel bileşen analizi (PCA- Principal Component Analysis) ve kaba küme yaklaşımlarıdır [Chouchoulas ve Shen, 2001; Dong ve ark., 1999].

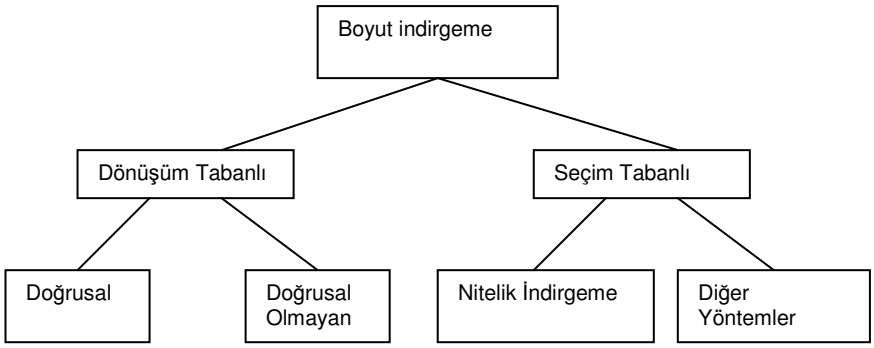
Düşük boyut indirgeme problemleri için tipik metot faktör analizidir [Fodor, 2002].

Görselleştirme problemleri**,** burada yapılacak şey verinin çıkartılması ve bir veri kümesi içerisinde ilişkilerin tanımlanmasıdır. Bunu yapabilecek metotlar projeksiyon izleme (Projection Pursuit) ve çok boyutlu ölçekleme (Multidimensional Scaling) dir. Bu metotlar içeriği düşük boyutlara indirir ve

grafik sunumu gibi görselleştirme araçlarından faydalanarak etkin bir şekilde kullanılabilir [Fodor, 2002].

Yüksek boyut indirgeme metotları genelde bir veri kümesi içerisinde niteliğin temsil ettiği anlamı bozar. Bu birçok uygulama için istenmeyen bir durumdur. Dolayısı ile verinin taşıdığı anlamın bozulmaması çoğu problem için kritik öneme sahiptir. Kaba küme yaklaşımının kullanımı bu sebepten dolayı faydalı olarak değerlendirilmektedir.

Şekil 2.6'da anlamsal korumalı, yani verinin içerdiği bilginin bozulmadığı sadece niteliklerin arasından en işe yarar olanların seçildiği boyut indirgeme yöntemlerinin bir sınıflandırması gösterilmektedir. Buradaki tekniklerin kullanımı problemde kullanılan verinin tipine ve uygulanan problem sahasına göre de değişebilmektedir.

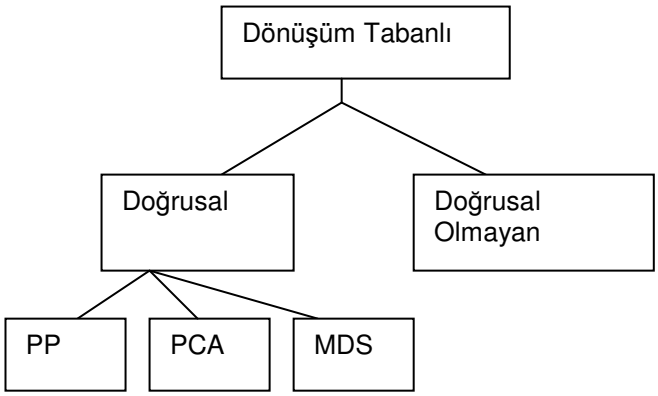


Şekil 2.6. Boyut indirgeme yöntemlerinin sınıflandırılması

Boyut indirgeme metotları dönüşüm tabanlı yöntemler ve seçim tabanlı yöntemler olmak üzere ikiye ayrılır.

Dönüşüm tabanlı yöntemler

Dönüşüm tabanlı yöntemler Şekil 2.7'de görüldüğü gibi kendi içinde doğrusal ve doğrusal olmayan şeklinde ikiye ayrılır.



Şekil 2.7. Dönüşüm tabanlı yöntemler

Doğrusal Yöntemler

Doğrusal yöntemler, temel bileşen analizleri (Principal Component Analysis-PCA), projeksiyon izleme (Projection Pursuit-PP) ve çok boyutlu ölçekleme yöntemlerini (Multidimensional Scaling-MDS) içeren tekniklerdir. Veri kümesinin içsel ilişkilerinin öklit yapısını belirlemek için kullanılırlar. Bununla beraber yüksek boyutlu veriler için bu yöntemler ilişkileri bulmakta başarısız olabilirler.

PCA [Jolliffe, 1986]; bir veri uzayından daha küçük bir veri uzayına en ilişkili niteliklerin dönüşümünü sağlayan istatistiksel bir tekniktir. Amacı p'<p ve verideki varyansın p'-boyutlu uzayda en maksimal açıklanabileceği şekilde p-boyutlu uzayı p'-boyutlu uzaya dönüştürmektir. Küçük varyansa sahip nitelikler atılır. Veri kümesinin prensipsel bileşenleri veri kalıpları matrisinin kovaryansını ve kovaryans matris uzayına dağılan orthogonal vektörleri hesaplayarak bulunabilir. Orthogonal vektörler bilindiğinde, uzaydaki herhangi bir vektör özvektörlerin lineer kombinasyonu ile kurulabilir, küçük özvektörlü değerler atılır [Engelbrecht, 2002; Wang, 2006].

PP [Friedman ve Tukey, 1974]; düşük boyutlu projeksiyonlar kullanarak yüksek boyutlu verinin analizi için dizayn edilmiştir. Amacı yüksek boyutlu veri içerisinde olası lineer olmayan ve ilginç yapıları ortaya çıkarmaktır.

MDS [Torgerson, 1952], veri içerisindeki gizli yapıların açığa çıkarılmasını sağlayan bir yöntemdir. Psikoloji, sosyoloji, antropoloji, ekonomi ve eğitimsel araştırmalarda kullanılır.

Doğrusal olmayan yöntemler

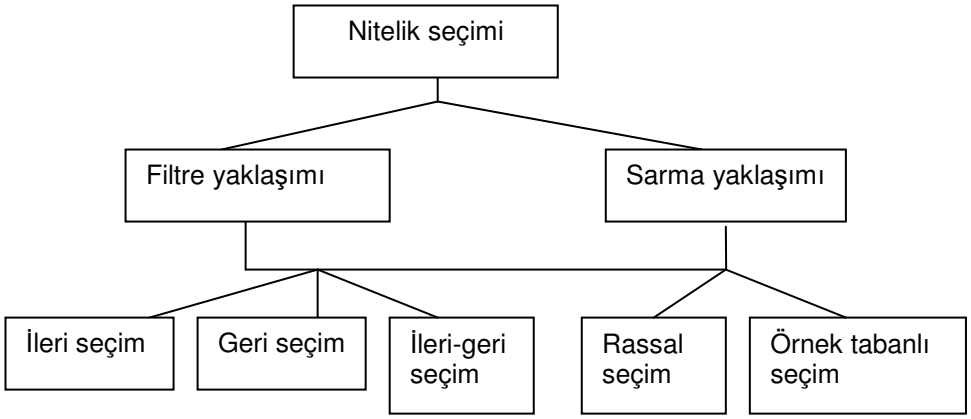
Yukarıda belirtilen yöntemlerin en büyük dezavantajı doğrusal olmayan veri ile boyut indirgeme yapamamalarıdır. Doğrusal olmayan ilişkilere sahip veri kümesinde bu yöntemler sadece öklit yapısını bulabilirler. Bu yöntem ise doğrusal olmama durumu ile başa çıkmaya çalışan tekniklerin gelişmesine yol açmıştır.

Seçim tabanlı yöntemler

Seçim tabanlı yöntemleri nitelik seçimi ve diğer yöntemler olarak ikiye ayrılır.

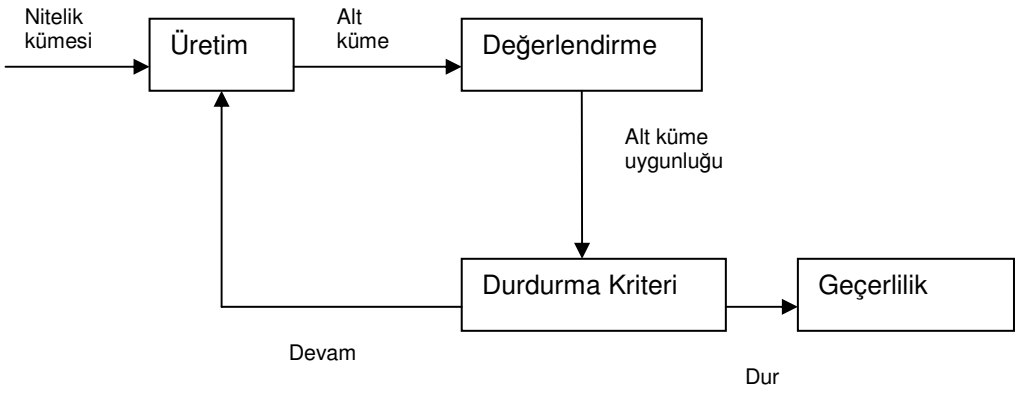
Nitelik seçimi

Nitelik seçimi işlemlerinin bir sınıflandırması Şekil 2.8.'de görülmektedir. n boyutlu bir nitelik kümesi verildiğinde nitelik seçimi 2^n aday alt küme içerisinden optimal nitelik alt kümeyi seçme işlemidir. Optimal alt küme tanımı çözülecek probleme göre değişir.



Şekil 2.8. Nitelik seçim stratejileri

Üretme işlemi, değerlendirme için nitelik alt kümelerini üreten bir araştırma metodu içerir. 0 nitelikle, tüm niteliklerle ya da belirli bir nitelik alt kümesi ile başlayabilir. Bu metotlar nitelikleri sezgisel olarak birer birer seçerler. Nitelikler her aşamada eklenebilir ya da çıkarılabilir. Son durumda ise nitelikler her bir aşamada eklenip çıkarılabilir ya da ondan sonra rassal olarak üretilebilir. Alternatif bir seçim stratejisi nitelikler arası farklılıkların bulunmasıdır. Değerlendirme fonksiyonu bir nitelik alt kümesinin uygunluğunu, üretim işlemi ile hesaplar, bunu en son en iyi aday ile karşılaştırır ve eğer en iyi olarak bulunmuşsa yer değiştirir. Bu durum Şekil 2.9.'da gösterilmiştir.



Şekil 2.9. Nitelik Seçimi

Durdurma ölçütü, her bir aşamada nitelik seçimi işleminin devam edip etmeyeceğine karar verir. Tipik bir durdurma kriteri optimal alt kümeye ulaşıldığında değerlendirme işleminin durdurulmasıdır. Durdurma ölçütü sağlandığında döngü yok edilir. Kullanım için sonuçtaki nitelik alt kümesi doğrulanabilir. Alt küme optimalitesini belirlemek zor bir problemdir. Alt küme minimalitesi ve alt küme uygunluğu arasında devamlı bir değiş tokuş vardır.

Nitelik seçim algoritmaları değerlendirme ölçütüne göre iki kategoriye ayrılabilirler. Eğer nitelik seçimini herhangi bir öğrenme algoritmasından bağımsız olarak yaparlarsa buna "Filtre yaklaşımı" denmektedir. Burada ilgisiz nitelikler kural çıkarımından önce ayıklanmaktadır. Filtreler özel bir kural çıkarım algoritmasının bir kısmı olmadıklarından birçok alanda kullanılabilmektedir [John ve ark., 1994].

Eğer öğrenme algoritmasının değerlendirme işlemi bir göreve (yani sınıflandırmaya) bağlı ise nitelik seçim algoritması sarma yaklaşımı kullanmaktadır. Bu metot bir kural çıkarım algoritmasından ölçülen kesinliği bir uygunluk ölçütü olarak kullanmak suretiyle nitelik altküme uzayını araştırır. Sarmalar daha iyi sonuçlar çıkarmalarına rağmen çok fazla sayıda nitelikle başa çıkamamaları, pahalı olmaları nedeniyle genelde daha az tercih edilirler.

Filtre yaklaşımı yöntemleri

RELIEF; filtre yaklaşımına dayanan ilk nitelik seçimi algoritmasıdır. RELIEF algoritmasında her bir niteliğe karar sınıf etiketleri arasında ayırt edilebilme kabiliyetini gösteren "ilgililik ağırlığı" verilir [Kira ve Rendell, 1992].

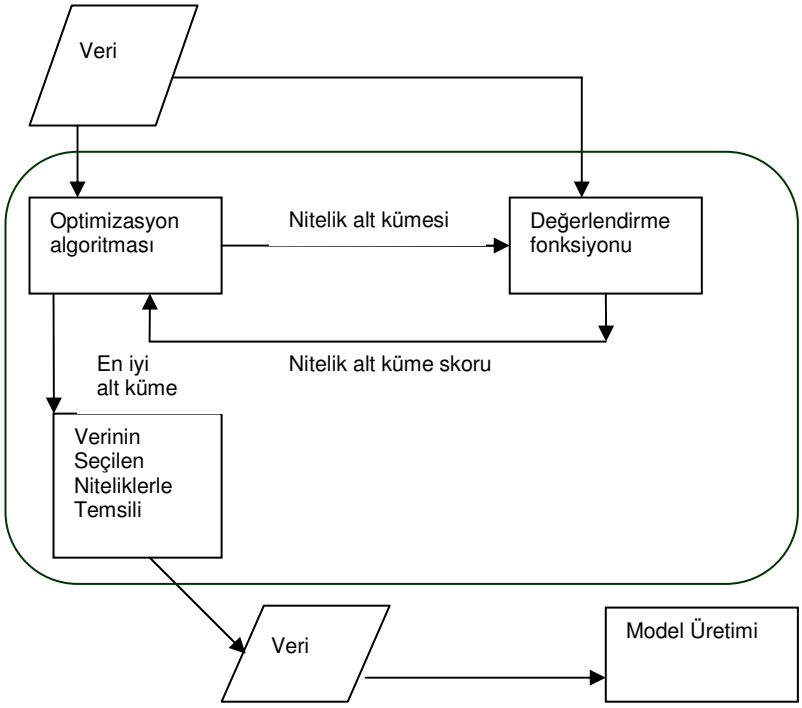
FOCUS; diğer bir filtreleme metodudur. Öncelikle genişlik stratejisi kullanır ve bütün nitelik alt kümelerinde eğitim verisinin tutarlı etiketlenmesini sağlayan minimal nitelik kümesini araştırır [Almuallim ve Dietterich, 1991].

SCRAP (Selection, construction ranking using attribute pattern), bir örnek uzayı içerisinde sırasal araştırma yaparak nitelik alakalılığını hesaplayan örnek tabanlı bir fitredir. SCRAP, diğer ileri ve geri araştırma tekniklerinin aksine nesneleri bir kerede ele alır. Buradaki ana fikir veri tablosunda karar sınırlarını değiştiren niteliklerin tespit edilmesidir. Bu nitelikler en çok bilgi verici olarak kabul edilirler [Raman ve Loerger, 2002] .

EBR (Entropy-based reduction); diğer bir filtre tabanlı nitelik indirgeme tekniğidir. Bu yaklaşım C4.5 gibi makine öğrenimi teknikleri ile uygulanan entropi sezgiseline dayanır. Bir veri kümesi içerisinde en çok bilgi kazancını sağlayan niteliklerin bulunmasını sağlar [Jensen ve Shen, 2001] .

FDR (Fractal dimension reduction); değişik ölçeklerde veri tarafından sergilenen kendine benzerlik kavramına dayanan bir nitelik seçimi yaklaşımıdır [Traina ve ark., 2000] .

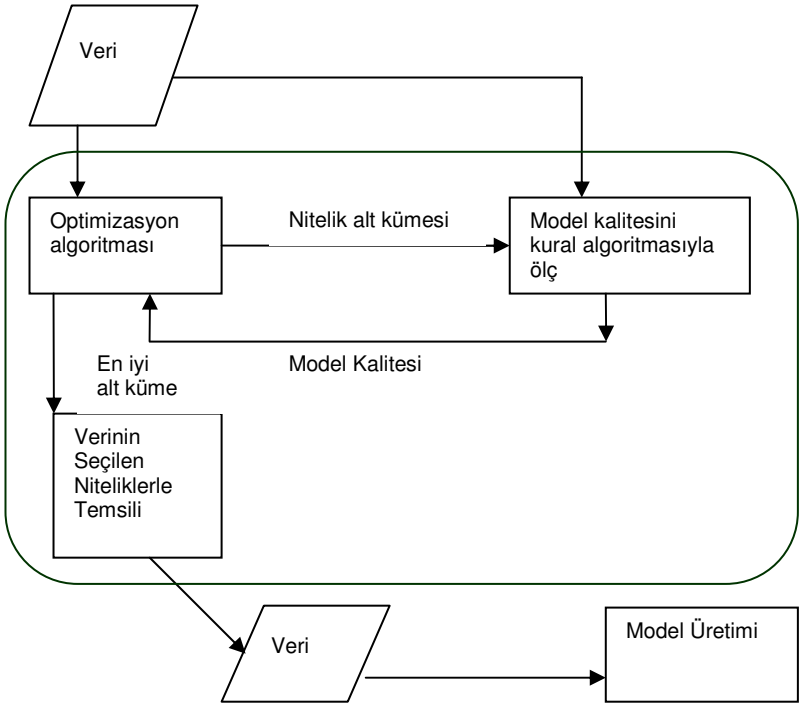
FG (Feature grouping); genelde nitelik seçiminde üretim işlemi artan oranda tek tek nitelikleri ekler veya çıkarır. Son zamanlarda her aşamada nitelikler gruplandırılarak araştırmalar yapılmaktadır. Bu strateji aynı anda çeşitli nitelikleri seçerek optimal alt kümeleri bulmak suretiyle hesap zamanını azaltır. Şekil 2.10.'da filtre yaklaşımı yöntemi gösterilmiştir [Yao, 2001] .



Şekil 2.10. Filtre yaklaşımı [Miadenic, 2006]

Sarma yaklaşımı yöntemleri

Sarma yaklaşımı yöntemleri rassal seçim ve örnek tabanlı seçim olarak ikiye ayrılır. Bu yöntemlerin dezavantajı hesaplama zamanı açısından pek uygun olmamaları, avantajı ise durdurma kriterinin bir kural çıkarım mekanizmasına bağlı olması nedeniyle daha kesin sonuçlar verebilmeleridir. Şekil 2.11.'de sarma yaklaşımı gösterilmiştir.



Şekil 2.11. Sarma yaklaşımı

Diğer yaklaşımlar

Genetik algoritmalar

Genelde geniş, lineer olmayan ve az anlaşılabilen uzayların hızlı bir şekilde araştırılması için oldukça etkili bir yöntemdir. Tek bir çözümün optimize edildiği klasik nitelik seçimi stratejilerinin tersine, çözüm topluluğu aynı anda değiştirilebilir. Bu çıktı olarak optimale yakın nitelik alt kümeleri üretir [Holland, 1975].

Tavlama benzetimi tabanlı nitelik seçimi

Tavlama maddenin kolay kırılganlığını azaltmak ve sertleştirmek için yavaşça ısıtılma ve soğutulma işlemlidir. Bu işlem bir metalin minimum enerji ile belli bir yapılanmaya ulaşabilmesi için yapılır. Eğer metal çok hızlı şekilde

tavlanırsa bu mükemmel organizasyonun başarıya ulaşması mümkün değildir [Kirkpatrick ve ark., 1983].

3. KABA KÜME TEORİSİ

Kaba küme (KK) teorisi 1980 lerin başında Pawlak tarafından önerilmiştir [Pawlak, 1982]. Diğer yöntemlerin aksine kaba küme yaklaşımı sadece veri içerisindeki bilgiyi kullanır ve istatistiksel parametrelere veya belirli varsayımlara dayanmaz.

Kaba küme kuramı, kümenin tek olarak elemanları ile tanımlandığı ve kümenin elemanları hakkında ilave hiçbir bilginin bulunmadığı klasik küme kuramının aksine, bir kümenin tanımlanması için başlangıçta evrenin elemanları hakkında bazı bilgilere gereksinim olduğu varsayımına dayanan yaklaşımdır. Nesneler, aynı bilgi ile nitelendiriliyorlarsa aynıdırlar veya ayırt edilemezlerdir. Ortaya konulan ayırt edilememe ilişkisi, KK kuramının temelini oluşturur. Bütün aynı nesnelerin kümesine elemanter küme denir ve bilginin temel taşını oluştururlar. Elemanter kümelerin herhangibir birleşimine kesin (crisp) küme adı verilir, aksi takdirde bir küme kabadır (rough, imprecise). Her KK'nin kesinlikle kümenin kendisinin ya da tümleyen kümesinin elemanları olarak sınıflandırılamayan elemanları (sınır hattı elemanları) vardır.

KK yaklaşımı kullanılarak çözülebilen ana problemler; özellik değerleri cinsinden nesnelerin kümesinin tanımı, özellikler arasındaki tam veya kısmi bağımlılıkların belirlenmesi, özelliklerin indirgenmesi, özelliklerin öneminin ortaya konulması ve karar kurallarının oluşturulmasıdır [Pawlak, 1991].

KK kuramının kullanılabileceği ana problemler; verilerin indirgenmesi, bağımlılıklarının keşfi, verilerin öneminin tahmini, verilerden karar (kontrol) algoritmalarının oluşturulması, verilerin yaklaşık sınıflaması, verilerdeki benzerlik ve farklılıkların keşfi, verilerdeki örüntülerin keşfi, neden-sonuç ilişkilerinin keşfini kapsamaktadır [Pawlak, 1994].

Kaba küme yaklaşımıyla ilgili incelenebilecek kavramlar;

- Verideki gizli kalıpların bulunması için etkili algoritmalara giriş,

- Verinin optimal kümelerinin hesaplanması (veri indirgeme),
- Verinin öneminin belirlenmesi,
- Veriden karar kurallarının kümelerinin oluşturulması,
- Formülasyonunun anlaşılabilmesinde kolaylık,
- İncelenen sonuçların dosdoğru yorumlanabilmesi,
- Bir çok algoritmalarla paralel işlem için uygunluğu.

Nesnelerin mevcut enformasyon ile görülebileceği varsayımı bilginin taneli bir yapısı olduğu görüşünü ortaya koyar. KK yaklaşımında herhangi bir muğlak kavram, bir kesin kavramlar çifti ile karakterize edilir ve bunlara muğlak kavramın alt ve üst yaklaşımları denir. Alt yaklaşım (lower approximation) kesin olarak kavrama ait olan bütün nesnelerden oluşur. Üst yaklaşım (upper approximation) ise kavrama ait olması muhtemel bütün nesneleri içerir. Alt ve üst yaklaşımlar arasındaki fark sınır bölgesini oluşturur [Pawlak, 1994].

Kaba küme yaklaşımı muğlaklık ile ilgilenir. Bulanık küme teorisine benzer olarak klasik küme teorisine bir alternatif değil onun içine gömülmüş ya da adapte edilmiş şekildedir [Pawlak, 2004].

S=(U,A) bir bilgi sistemi,
U, S'nin evreni adı verilen nesnelerin boş olmayan sonlu kümesi,
A, niteliklerin boş olmayan sonlu kümesi,
$R \subseteq A$ ve $X \subseteq U$ olsun.

X kümesinin R'ye göre alt yaklaşımı; R'ye göre *kesinlikle* X olarak sınıflandırılan nesnelerin kümesine,

X kümesinin R'ye göre üst yaklaşımı; R'ye göre *muhtemelen* X olarak sınıflandırılabilen nesnelerin kümesine,

X kümesinin R'ye göre sınır bölgesi; R'ye göre *ne* X *de ne de* X *de olmayan* nesnelerin kümesine denir.

Bu bilgilere göre *kaba küme*nin tanımı yapılacak olursa;

- Eğer X'in sınır bölgesi boş ise, X kümesi R'ye göre *kesin*dir.
- Eğer X'in sınır bölgesi boş değil ise X kümesi R'ye göre *kaba*dır.

R(x); x elemanı ile belirlenen R eşdeğerlik sınıfını ifade etsin. Ayırtedilemezlik ilişkisi evren hakkındaki bilgi eksikliğini ifade eder. Ayırtedilemezlik ilişkisinin eşdeğerlik sınıfları R tarafından üretilen granüller diye adlandırılırlar ve bilginin temel kısmını ifade ederler.

X'in R'ye göre alt, üst yaklaşım ve sınır bölgesinin fonksiyonel tanımları aşağıdadır;

- *X' in R Alt yaklaşımı*

$$R_*(X) = \bigcup_{x \in U} \{R(x) : R(x) \subseteq X\} \tag{3.1}$$

- *X' in R üst yaklaşımı*

$$R^*(X) = \bigcup_{x \in U} \{R(x) : R(x) \cap X \neq \varnothing\} \tag{3.2}$$

- *X' in R sınır bölgesi*

$$BN_R(X) = R^*(X) - R_*(X) \tag{3.3}$$

daha öncede tanımlandığı gibi alt yaklaşım, kümeye kesin olarak ait tüm elemanlardan; üst yaklaşım, kümeye ait olması muhtemel bütün elemanlardan; sınır bölgesi, kümeye ait olarak ya da onun tümleyeni olarak sınıflandırılamayacak elemanlardan oluşur. Böylece kaba küme, klasik kümenin tersi olarak boş olmayan sınır bölgesine sahiptir.

Alt ve üst yaklaşımların özellikleri aşağıda verilmektedir:

RS-1. $$R_*(X) \subseteq X \subseteq R^*(X), \tag{3.4}$$

RS-2. $$R_*(\varnothing) = R^*(\varnothing) = \varnothing; R_*(U) = R^*(U) = U, \tag{3.5}$$

RS-3. $$R^*(X \cup Y) = R^*(X) \cup R^*(Y), \tag{3.6}$$

RS-4. $$R_*(X \cap Y) = R_*(X) \cap R_*(Y), \tag{3.7}$$

RS-5. $$R_*(X \cup Y) \supseteq R_*(X) \cup R_*(Y), \tag{3.8}$$

RS-6. $$R^*(X \cap Y) \subseteq R^*(X) \cup R^*(Y), \tag{3.9}$$

RS-7. $$X \subseteq Y \rightarrow R_*(X) \subseteq R_*(Y) \,\&\, R^*(X) \subseteq R^*(Y), \tag{3.10}$$

RS-8. $$R_*(-X) = -R^*(X), \tag{3.11}$$

RS-9. $$R^*(-X) = -R_*(X), \tag{3.12}$$

RS-10. $$R_*R_*(X) = R^*R_*(X) = R_*(X), \tag{3.13}$$

RS-11. $$R^*R^*(X) = R_*R^*(X) = R^*(X), \tag{3.14}$$

(Burada –X, U-X' i göstermektedir.)

Kaba küme, üyelik fonksiyonları ile de tanımlanabilir [Pawlak, 2004]. Üyelik fonksiyonu;

$$\mu_X^R : U \rightarrow [0,1], \tag{3.15}$$

Burada,

μ_X^R ; x'in X kümesine R ilişkisi içerisinde aitliğini ifade eder.

$$\mu_X^R(x) = \frac{card(X \cap R(x))}{card(R(x))}, \tag{3.16}$$

Card(x), X'in kardinalitesi yani küme içindeki eleman sayısıdır. Kaba üyelik fonksiyonu R bilindiğinde, x'in X'e aitliğinin koşullu olasılığını ifade eder.

Alt, üst yaklaşımları ve sınır bölgelerinin kaba üyelik fonksiyonu kullanılarak yapılan tanımlamaları aşağıdadır;

$$R_*(X)=\left\{x\in U:\mu_X^R(x)=1\right\}, \tag{3.17}$$

$$R^*(X)=\left\{x\in U:\mu_X^R(x)>0\right\}, \tag{3.18}$$

$$BN_R(X)=\left\{x\in U:0<\mu_X^R(x)<1\right\} \tag{3.19}$$

Kaba üyelik fonksiyonunun alt, üst yaklaşım ve sınır bölgesine göre özellikleri ise aşağıdadır;

RS'-1. $\mu_X^R(x)=1$ eğer $x\in R_*(X)$ (3.20)

RS'-2. $\mu_X^R(x)=0$ eğer $x\in U-R^*(X)$ (3.21)

RS'-3. $0<\mu_X^R(x)<1$ eğer $x\in BN_R(X)$ (3.22)

3.1. Kaba Küme Analizinin Temel Kavramları

3.1.1. Bilgi sistemleri

U ve A diye iki sonlu, boş olmayan evren ele alalım. Burada U evren, A ise nitelik kümesidir. S=(U,A) çiftine bilgi sistemi denir. Her bir $a\in A$ için a'nın *yayılım alanı* denilen V_a değer kümesi bulunur. A'nın herhangi bir B altkümesi U üzerinde ikili ilişki I(B)'yi belirler, buna ayırt edilemezlik ilişkisi denilir. Gösterimi denklem 3. 23'de verilmiştir.

x I(B)y eğer a(x)=a(y) her bir $a\in A$ için. (3.23)

Burada,
a(x) : x elemanı için a niteliğinin değerini,
I(B) : bir eşdeğerlik ilişkisini ifade eder.

U/I(B) ya da basitçe U/B; I(B) nin bütün eşdeğerlik sınıfları ailesini, yani B tarafından belirlenen partisyonunu gösterir. I(B) eşdeğerlik sınıfı, yani U/B partisyon bloğu B(x) ile gösterilir. Eğer (x,y), I(B) ye aitse buna x ve y, B-ayırt

edilemez denilir. I(B) ilişkisinin eşdeğerlik ilişkisine de B-elemanter küme denilir. Kısaca I(B) ilişkisi ile birbirinden ayırt edilen sınıflara B(x), eğer ayırt edilen nitelik yoksa da bu duruma B-ayırt edilemez denilir.

3.1.2. Karar tabloları

İçindeki niteliklerin durum ve karar şeklinde iki sınıfa ayrıldığı bilgi sistemine *karar tablosu* denilir. Durum ve karar nitelikleri ile oluşturulan karar tablosu evreninin partisyonlarını ifade eder. Partisyon yaklaşımlarının durum nitelikleri yardımıyla karar nitelikleri ile tanımlanması amaçlanır. Bir karar tablosu S; durum nitelikleri C ve karar nitelikleri D ile ifade edildiğinde; S=(U,C,D) şeklinde gösterilir. Karar tablosunun her bir satırı ilgili durumlar sağlandığında alınacak kararları belirleyen bir karar kuralını ifade eder.

Karar kuralları tutarlı ve tutarsız olabilir. Buna bazen kesin ve olası kurallar da denilebilmektedir. Karar tablosundaki tutarlı kuralların tüm kurallara oranı tutarlılık faktörü olarak tanımlanır ve $\gamma(C,D)$ olarak gösterilir. Eğer $\gamma(C,D)$ =1 ise karar tablosu tutarlıdır.

3.1.3. Nitelik Bağımlılığı

Veri analizinde diğer önemli bir husus nitelikler arası bağımlılıkların bulunmasıdır. Eğer D deki bütün nitelikler C deki nitelik değerleri ile belirleniyorsa, yani D nitelikleri tamamen C niteliklerine dayanıyorsa bu C=>D ile gösterilir. Eğer D, C ye tamamen bağlı ise $I(C) \subseteq I(D)$ dir. Bu C ile yaratılan partisyonun D den daha iyi olduğu anlamına gelir.

Eğer D, C ye bir k derecesi ile bağımlı ise, 0<k<1;

$$\gamma(C,D) = \frac{Card(POS_C(D))}{Card(U)} \tag{3.24}$$

Burada $POS_C(D) = \bigcup_{X \in U/I(D)} C_*(X)$

$POS_C(D)$ ifadesine U/D partisyonunun C'ye göre pozitif alanı denir ve bu ifade C yardımıyla U/D partisyonu bloklarına kesinlikle sınıflandırılabilecek U'nun elemanlarını içerir.

3.1.4. Nitelik indirgeme

"Bir veri tablosunda bazı veriler, veri tablosunun temel özellikleri kaybedilmeden atılabilir mi?" sorusu nitelik seçimi kavramını ifade eder.

Küçük nitelik kümeleri kullanılarak aynı bağımlılık derecesi ve yaklaşım kesinliği bulunabilir. Bu fikri daha iyi ifade etmek için B, A kümesinin bir alt kümesi ve a niteliği B'ye ait olsun; eğer $I(B) = I(B - \{a\})$ ise a niteliği B'de *gereksiz*dir aksi halde a niteliği B'de gereklidir. B bütün nitelikleri gerekli ise *bağımsızdır* denir.

Eğer B' bağımsız ve I(B')=I(B) ise B'nin indirgenmişi B' altkümesidir.

Diğer önemli bir özellik ise çekirdek (core) özelliğidir. $Core(B) = \bigcap \mathrm{Re}\, d(B)$ şeklinde ifade edilir. Red(B); B'nin tüm indirgenmiş kümeleridir.

3.1.5. Ayırt edilebilirlik matrisi ve fonksiyonlar

İndirgemeleri ve kor'u kolayca hesaplamak için Skowron tarafından ayırt edilebilirlik matrisi önerilmiştir [Skowron ve Rauszer, 1991].

$B \subseteq A$ nın ayırt edilebilirlik matrisi, M(B) ile gösterilir. Matrisin girdileri şu şekilde hesaplanır.

$$c_{ij} = \{a \in B : a(x_i) \neq a(x_j)\}, \text{ i,j=1,2,...n.} \quad (3.25)$$

Buradaki c_{ij} girdisi x_i ile x_j nesnelerini ayırt eden bütün nitelikler anlamındadır. M(B) ile x ve y nesne çiftini $\delta(x, y) \subseteq B$ nitelik alt kümesine eşleştirir. Bu durum şu özelliklere sahiptir;

M(B)-1 $\delta(x, y) = 0,$ (3.26)

M(B)-2 $\delta(x, y) = \delta(y, x),$ (3.27)

M(B)-3 $\delta(x, z) \subseteq \delta(x, y) \bigcup \delta(y, z)$ (3.28)

Burada indirgeleri bulmak için nitelik kümesi tarafından bütün nesnelerin ayırt edilebildiği minimal nitelik kümesi bulunmalıdır. Her bir M(B) matrisi bir boole ayırt edilebilirlik fonksiyonu f(B) ile tanımlanır. Her bir $a \in B$ için bir $\bar{a}$ boole değişkeni tanımlansın. $\sum \delta(x, y)$, $\delta(x, y)$ nitelik kümesine atanan bole değişkenlerin toplamı olsun. Ayırt edilebilirlik matrisinin matematiksel ifadesi;

$$f(B) = \prod_{(x,y) \in U^2} \{\sum \delta(x, y) : (x, y) \in U^2 ve \delta(x, y) \neq 0\}. \quad (3.29)$$

3.1.6 Niteliklerin önemi

Bir niteliğin önemi; nitelik bilgi tablosundan çıkarıldığındaki etki ile ölçülür. $\gamma(C, D)$ karar tablosunun tutarlılık derecesini ve C ile D arasındaki bağımlılık derecesini gösterir. a niteliğinin önemi $\gamma(C, D)$ ile $\gamma(C - a, D)$ arasındaki farkla ifade edilir. Bir niteliğin önemi;

$$\sigma_{(C,D)}(a) = \frac{(\gamma(C, D) - \gamma(C - \{a\}, D))}{\gamma(C, D)} = 1 - \frac{\gamma(C - \{a\}, D)}{\gamma(C, D)} \quad \text{veya} \quad (3.30)$$

$$\sigma_{(C,D)}(B) = \frac{(\gamma(C, D) - \gamma(C - B, D))}{\gamma(C, D)} = 1 - \frac{\gamma(C - B, D)}{\gamma(C, D)} \quad \text{veya} \quad (3.31)$$

$$\varepsilon_{(C,D)}(B) = \frac{(\gamma(C,D) - \gamma(B,D))}{\gamma(C,D)} = 1 - \frac{\gamma(B,D)}{\gamma(C,D)} \tag{3.32}$$

ölçütlerinden biri ile ölçülebilir [Geng ve Zhu, 2006]. $0<\sigma<1$ dir.

3.1.7. Değişken duyarlı kaba kümeler

Değişken duyarlı kaba küme (VPRS) [40]; veri kalıplarını fonksiyonelden ziyade istatistiksel eğilimlerle ifade etmeye çalışır. VPRS'in ana amacı önceden belirlenen belli bir seviyede küçük bir hata oranı ile nesnelerin sınıflandırılmasının sağlanmasıdır.

$X, Y \subseteq U$ olsun, göreceli sınıflandırma hatası;

$$c(X,Y) = 1 - \frac{|X \cap Y|}{|X|} \quad \text{dir.} \tag{3.33}$$

c(X,Y)=0 ise $X, Y \subseteq U$ olduğu gözlemlenebilir. Sınıflandırmada kapsama derecesi için belli bir hata oranı β ile izin verilebilir.

$$X \subseteq_\beta Y \text{ if } c(X,Y) \le \beta, \quad 0 \le \beta \le 0.5 \tag{3.34}$$

Bu şekilde X kümesinin β alt ve üst yaklaşımları sırasıyla;

$$\underline{R}_\beta X = \bigcup \left\{ [x]_R \in U/R \middle| [x]_R \subseteq_\beta X \right\} \tag{3.35}$$

$$\overline{R}_\beta X = \bigcup \left\{ [x]_R \in U/R \middle| c([x]_R, X) \le 1 - \beta \right\} \tag{3.36}$$

olarak ifade edilebilir.

Böylece pozitif, negatif ve sınır alanları sırasıyla;

$$POS_{R,\beta}(X) = \underline{R}_{\beta}X \tag{3.37}$$

$$NEG_{R,\beta}(X) = U - \overline{R}_{\beta}X \tag{3.38}$$

$$BND_{R,\beta}(X) = \overline{R}_{\beta}X - \underline{R}_{\beta}X \quad \text{dır.} \tag{3.39}$$

Dolayısı ile belirli bir β oranı ile nitelik indirgemeleri bulunabilir. β oranının yerine $0 \le l < u \le 1$ ile alt ve üst sınır tanımlanarak alt ve üst yaklaşımlar;

$$\underline{R}_{u}X = \bigcup \left\{ [x]_R \in U/R \middle| c([x]_R, X) \le 1-u \right\} \tag{3.40}$$

$$\overline{R}_{\beta}X = \bigcup \left\{ [x]_R \in U/R \middle| c([x]_R, X) \le 1-l \right\} \tag{3.41}$$

olarak ifade edilir [Katzberg ve Ziarko, 1996].

3.2. Literatürdeki Çalışmalar

Pawlak, kaba sınıflandırma adında yaptığı çalışmada [Pawlak, 1983] yine kendisi tarafından tanımlanan kaba küme temelinde nesnelerin yaklaşık sınıflandırmasıyla ilgilenmiştir.

Chan, veri madenciliğinde nitelik oluşturmak için kaba küme yaklaşımı önermiştir. Yapılan çalışmada yeni bir niteliğin eklenmesi ve çıkarılması durumunda alt yaklaşımların ve üst yaklaşımların güncellenmesi sonuçları temelinde, veriden öğrenen sınıflandırma kuralları (learning clasification rules) için yukarı-aşağı (top-down) algoritması geliştirmiştir. Algoritma kurallar oluşturmak için LERS öğrenen algoritmaları kullanır, bu yüzden öğrenilen kurallar hedef sınıfların minimal diskriminantına ayıran açıklamalarıdır. Tablo tutarsız olduğunda algoritma alt yaklaşımlardan kesin kuralları, üst yaklaşımlardan da muhtemel kuralları öğrenir.Tablolar tutarlı ise yalnızca tek bir kurallar kümesi vardır. Yaklaşım dinamik nitelik oluşturmak için etkili bir araç olarak kullanılabilir. Önerilen yöntem ve LERS endüktif öğrenme

algoritmasının birleşimiyle veri tabanlarından sınıflandırma kuralları öğrenmek için “quasi incremental” algoritması verilmiştir [Chan, 1997].

Joseph ve arkadaşları tarafından, diyabetik veri tabanlarında veri madenciliği yaklaşımında kaba kümeler incelenmiş ve ROSETTA paket programı tanıtılmıştır. Kaba kümeler diyabetik veri tabanlarının analizinde oldukça etkili çıkmıştır [Joseph ve ark., 1998].

Felix ve Ushio, kaba kümeleri kullanarak tamamlanmamış ve tutarsız veriden kural indirgeme yaklaşımı önermişlerdir. Bu çalışmada kaba küme teorisi temelinde tutarsız ve eksik bir bilgi sisteminde minimal kuralları içeren iki yöntem önerilmiştir. Her iki yöntemde ikili ayırt edilebilirlik matrisinin tanımlamasını kullanarak minimal örüntülerin (minimal coverings) aramasında bit-wise operasyonlar kullanmışlar ve kümelerdeki işlemler yeniden düzenlenmiştir. İlk yöntem örüntülerin geniş bir aramasıdır (exhaustive search of coverings) ve ikincisi genetik algoritma tabanlı bir arama kullanır. Tutarsızlıklar alt ve üst yaklaşımlarla çözülür ve tamamlanmamış problem kaba ayırt edilebilir durumdaki örnek çiftleri arasındaki ayırt edilebilirliğin tanımının modifiye edilmesiyle çözülür [Felix ve Ushio, 1999].

Kusiak ve arkadaşları, veri madenciliği yaklaşımında kaba küme teorisi, kümeleme analizi, ölçü teorisi temelinde bağımsız karar alma için bir çalışma yapmışlardır. Eğitim kümesinden karar kurallarının çıkartılması için iki algoritma önermişlerdir. Nitelik çıkarım algoritması (feature extraction algorithm) çoklu nitelik kümeleri oluşturabilir. Bu nitelik kümeleri primer ve doğrulama algoritmalarıyla (primary and confirmation algorithms) bir nesnenin çıktısını tahmin etmek için kullanılabilir. Primer karar alma algoritması karar kurallarının nitelikleriyle yeni bir nesnenin nitelik değerlerini karşılaştırır. Eşleştirme kriteri ile karşılaşıldığı zaman nesnenin kararı eşleştirme karar kuralına eşit atanır ve doğrulama algoritmasına yardım için başvurulur. Özdeş iki algoritmayla kararlara ulaşıldığı zaman son kararada

ulaşılmış olur. Yüksek doğruluklu kararlara ulaşmada bu iki yaklaşımın birleşimi etkilidir. Klinik verilerine uygulandığında çok iyi doğruluk derecesi elde edilmiştir. Yüksek doğruluklu bağımsız tanı için niteliklerin sayısının orjinal veri kümesindekinden küçük olması önemlidir. Niteliklerin indirgenmiş sayısı test etme maliyetlerini azaltır. Noninvasif testlerdeki veri teşhis için kullanıldığından hastaların mortalite ve morbidite riskleri önemli derecede azalır [Kusiak ve ark., 2000].

Pawlak, karar algoritmalarında kaba kümeler ve bayes teoremi arasındaki ilişkiyi incelemiştir. Bayes teoremine dayanan kaba küme teoremi önceki veya sonraki (prior or posterior) olasılıklara başvurmaksızın karar kurallarından şartlar ve kararlar arasındaki ilişkiyi açıklar. Klasik Bayes nedenleme (classical Bayesian reasoning)'nin tersine karar algoritmalarındaki şartlar ve kararlar arasındaki probabilistik ilişkiyi tanımlar. Bu iki yaklaşımın tek başına kullanılmasından daha iyi sonuç vermiştir [Pawlak, 2000].

Pawlak, tarafından 2000 yılında yine kendisi tarafından yapılan çalışma temelinde bunun uzantı versiyonu sunulmuştur. Bayes kuralının anlamı istatistiksel çıkarımdan farklı anlamda kullanılmıştır. İstatistiksel çıkarım Bayes kuralı temelinde veri hakkında bilgi olmaksızın bazı parametreler hakkında öncelikli bilgi yani öncelikli olasılık (prior probability) sunar. Daha sonra veri, mevcut olduğu zaman sonralıklı (posterior) olasılık hesaplanır. Sonralıklı olasılık, öncelikli olasılığı doğrulamak için kullanılır. Kaba kümede kesinlik (certainty) ve kapsam (coverage) faktörleri olarak isimlendirilen iki şart olasılığı her karar kuralıyla atanır. Bu iki faktör kaba küme teorisi temelinde bir kümenin alt ve üst yaklaşımlarıyla yakından ilişkilidir. Öncelikli ve sonralıklı olasılıklara başvurulmaksızın Bayes kuralı uygulanarak karar kuralı tersine (invert) çevrilir [Pawlak, 2001].

Zhong ve Skowron, veri tabanlarından sınıflandırma kurallarının keşfi için kaba küme yaklaşımı temelinde kural keşif prosesi önermişlerdir. Prosesin

çekirdeğini belirsiz ve tamamlanmamış veriyi içeren veri tabanlarından sınıflandırma kurallarının keşfi için genelleştirilmiş dağılım tablosu (generalized distribution table, GDT) ve kaba küme sistemlerinden oluşan soft hibrit indirgeme sistemi (GDT-RS) oluşturur. Yapılan çalışmada slope-collapse veri tabanı kullanılmıştır. Sonuçlar diskriminant analizi kullanarak aynı veri üzerinde aynı işi yapan bir uzman tarafından hesaplanmştır. Öngörülen yaklaşım kullanılarak seçilen nitelikler uzman tarafından seçilen çok önemli niteliklerle aynı çıkmıştır [Zhong ve Skowron, 2001].

Ananthanarayana ve arkadaşları tarafından kaba kümeler kullanılarak etkili bir veri madenciliği için "kaba PC ağaç (rough PC tree)" yapısı önerilmiştir. Bu yaklaşımla işlem zamanı ve hafıza gereksinimleri sınıflandırma doğruluğu değişmeksizin azalmıştır [Ananthanarayana ve ark., 2002].

Hua ve Yuan, kaba küme teorisi temelinde karar sisteminde minimal indirgeme için sezgisel bir genetik algoritma önermiştir. Geliştirilen algoritmada yeni bir operatör, değiştirme (modify) operatörü eklenmiştir. Popülasyon çaprazlama, mutasyona tabi tutulduktan sonra değiştirme operatörüne de tabi tutulmaktadır. Bu yeni operatör şart nitelikleriyle aynı karar kabiliyetine sahip her bir kromozomu garanti eder [Hua ve Yuan, 2002].

Hassan ve arkadaşları, kaba küme ve sinir ağlarını birleştirerek kaba nöral sınıflandırma adı altında bir yapı geliştirmişlerdir. Kaba kümeler veri analizi ve sinir ağlarıyla nitelik seçimi için kullanılmaktadır. Kaba küme teorisi veri içindeki doğru olmayan bağımlılıkların açıklanması için araçlar sağlar. Verinin ön işlenmesi adımında kullanılmaktadır. İndirgenmiş özelliklerin kümesi hesaplandıktan sonra girdi vektör değerlerinin çıkartılmasıyla indirgenmiş veri kümesine göre kaba nöronlar kullanılarak sinir ağı modelinin yapısı yapılandırılır. Geliştirilen algoritma prostat kanseri veri setine ve biyopsi veri setine uygulanmıştır. Ağın öğrenme zamanını ve sınıflandırma hata oranını oldukça azaltmıştır. Kaba kümeler kullanılarak standart sinir ağı yapısından farklı bir yapı geliştirilmiştir [Hassan ve ark., 2002].

Zhong ve arkadaşları, genelleştirilmiş dağılım tablosu ve kaba küme (generalized distribution table and rough set, GDT-RS), Boolean nedenlemesiyle kaba kümeler (rough sets with Boolean reasoning, RSBR) olarak isimlandirilen iki kaba küme temelli hibrit sistemin uygulamasını menenjit veri kümesinde gerçekleştirmişlerdir [Zhong ve ark., 2002]. RSBR, GDT-RS başlamadan önce verinin ön işlenmesi adımında gerçek değerli niteliklerin kesikleştirilmesi için kullanılmıştır. RSBR yalnızca kesikleştirilmiş nitelik değerlerinin etkisini incelemekle kalmaz GDT-RS'ninde performansını etkiler. GDT-RS belirsiz ve tamamlanmamış veri tabanlarından sınıflandırma kurallarının keşfi için kullanılan soft hibrit bir indirgeme sistemidir. Geniş gerilimli (large strength) kurallar kümesinin üretilmesinde etkilidir. Sonuçlar göstermektedir ki RSBR ve GDT-RS hibrit sisteminin kullanılması özellikle niteliklerin karışık tipte bulunduğu veri tabanlarından kural keşfinde iyi bir yöntemdir.

Hassan ve Tazaki, yapmış oldukları çalışmada karar tablosuna kaba küme teorisi kullanarak veriyi analiz etmiş, C4.5 algoritması kullanarak başlangıç popülasyonu oluşturmuş daha sonra tekrar kaba küme yaklaşımı kullanarak değerlendirme yapmışlar ve genetik programlama operatörleri kullanarak modifikasyonlar oluşturup en etkin kurallar kümesi elde etmeye çalışmışlardır. Elde edilen sonuçlar C4.5 ve standart kaba küme yaklaşımıyla karşılaştırılmış ve bu yaklaşımlardan daha iyi sonuçlar elde edilmiştir [Hassan ve Tazaki, 2003].

Renpu ve Wang, kaba küme ve sinir ağları yaklaşımlarının avantajlarını birleştirerek karar tablolarından etkin sınıflandırma kuralları çeken bir hibrit sistem sunmuşlardır. Kaba kümeler daha önceki çalışmalarda sadece veri madenciliğinde sinir ağlarının kullanım sürecini basitleştirmek ya da hızlandırmak için kullanılırken; bu çalışmada karar tablolarını indirgemek ve eğitilmiş bir sinir ağından kural çekme esnasında gürültüleri süzmek için kullanılmıştır. Karar tablosundaki nitelikler iki adımda indirgenmektedir. Birinci adımda, sınıflama bilgisi kaybı olmadan kaba küme yaklaşımı ile

tablodan fazla ve gereksiz nitelikler uzaklaştırılır. İkinci adımda istenen sınıflama doğruluğu korunurken; tablodan gürültülü nitelikleri silmek için sinir ağları yaklaşımı kullanılır. Hibrit sistemde sinir ağları sadece karar tablosunu indirgemek ve gürültüyü süzmek için bir araç olarak kullanılır. İndirgenmiş karar tablosundan sınıflama kuralları üretilirken kaba küme teorisi kullanılmaktadır. Hem yapay problemleri hem de gerçek dünya problemlerini içeren bir dizi deney yapılmıştır. Geleneksel yöntemlerle karşılaştırmak için üç veri madenciliği problemine uygulanmış, dört adet standart veri tabanı kullanılmıştır. İkinci olarak, sistemi gürültülü koşullarda test etmek için veriye farklı düzeylerdeki gürültü rassal olarak eklenerek MONK3 veri kümesi üzerinde deneyler yapılmıştır. Karşılaştırma deneylerinin sonuçları geleneksel sinir ağları ve kaba küme yaklaşımından daha özlü ve doğru kurallar üretilmiştir. Deney sonuçları farklı gürültü seviyeleri altında bu yaklaşımın iyi bir şekilde çalıştığını göstermektedir [Renpu ve Wang, 2003].

Jaroslaw ve Katarzyna, tarafından önerilen kaba küme ve yapay sinir ağları temelinde hibrit sınıflandırma yaklaşımında sistem iki parçaya bölünmüştür: karar tablosunun dönüşümü ve yapay sinir ağlarının uygulaması. Eğitim tablosu olarak isimlendirilen karar tablosu ikiye bölünmüştür $DT_1=(U_1,A\cup\{d\})$ ve $DT_2=(U_2,A\cup\{d\})$. Kural kümesini içeren kurallar DT1'den üretilmiştir. DT_2 tablosu ConRes programı kullanılarak yeni tablo $DT_2'=(U_2,A_{\text{kural kümesi}}\cup\{d\})$ elde edilir, niteliklerin değerleri dönüşüm esnasında hesaplanır. Yapay sinir ağlarının yapılandırılması için eğitim tablosu $DT_2'=(U_2,A_{\text{kural kümesi}}\cup\{d\})$'dir. ConRes programıyla sınıflandırma sonuçları sunulmuştur. Iris, diyabet ve Australian verileri kullanılmıştır. Sınıflandırmadan önce Rosetta paket programı kullanılarak kümeler ikiye bölünmüştür. Boolean nedenleme algoritması kullanılarak kesikleştirme işlemi yapılmıştır. Alt kümelerden birisiyle kurallar oluşturulmuş ve bu alt küme ağın öğrenmesinde kullanılmıştır, ikinci alt küme sınıflandırmayı test etmek için kullanılmıştır. Hibrit sistem yapay sinir ağlarının tek başına kullanılmasından çok daha iyi sonuç vermiştir [Jaroslaw ve Katarzyna, 2003].

Hassanien, medikal veri kümelerinin bir kümesinden nitelik indirgeme ve sınıflandırma kurallarının oluşturulması için bir kaba küme yaklaşımı önermiştir. Bu amaçla sınıflandırma için bir sınıf etiketine atanan niteliklerin minimal alt kümesini içeren verinin tüm indirgemelerini (reduct) bulan kaba küme indirgeme tekniği kullanmıştır. Hassanien'in indirgeme algoritması (reduct algorithm)olarak kullandığı algoritma aslında Jensen ve arkadaşları tarafından yapılan SAT ile kaba küme indirgemelerinin bulunması çalışmasında geçen hızlı indirgeme algoritmasıyla (Quick reduct algorithm) aynıdır. Niteliklerin yaklaşım kalitesi temelinde kuralların geçerliliği hesaplanmıştır. Sonuçlar ID3 sınıflandırma algoritmasıyla karşılaştırılmış ve oldukça iyi çıkmıştır [Hassanien, 2004].

Busse ve Siddhaye, eksik veriden kural indirgemek için kaba küme yaklaşımını önermiştir. Karar tablolarında bazı nitelik değerleri eksik olduğu zaman karar tabloları tamamlanmamış durumda olur. Eksik nitelik değerlerinin iki ana durumu incelenmiştir: kayıp (orijinal değer silinmiş) ve ilgisiz (orijinal değer ilgisiz). Nitelik-değer çifti blokları karakteristik kümelerin yapılandırılmasında, karakteristik ilişkiler ve eksik nitelik değerli karar tabloları için alt ve üst yaklaşımlar kullanmıştır. LEM2 kural indirgeme algoritmasının modifiye versiyonu kullanılmıştır. Sonuçlar göstermektedir ki eksik nitelik değerleri kayıp olarak değerlendirmeye alındığında sınıflandırmada ki hata oranı oldukça azalmaktadır [Busse ve Siddhaye, 2004].

Wang, kaba küme temelinde gürültülü veriden sınıflandırma bilgisi elde etmek adlı çalışmasında gürültülü bilgi sistemlerini analiz etmek için yalnızca kaba küme teorisinin kullanılmasının iyi olmadığını vurgulamıştır. Bulanık alt yaklaşım (fuzzy lower approximation) temelinde toleranslı yaklaşım (tolerant approximation) gürültülü veriden etkili kuralların keşfiyle ilgili bir kavramdır. Toleranslı alt yaklaşım temelinde etkili bir kural indirgeme algoritması önerilmiştir. Geliştirilen algoritma promoters, nursery, lenses, TicTacToe, car veri tabanlarında denenmiş IBM Intelligent Miner 'dan deneysel (empirical)

sınıflandırma algoritması kullanılarak karşılaştırmalar yapılmıştır. Etkinlik ölçütleri tahmin doğruluğu, maliyet oranı, ve randomizasyon analizi temelinde kural geçerlilik oranını içermektedir. Sonuçlar göstermektedir ki geliştirilen algoritma oldukça iyi sonuçlar üretmiştir [Wang, 2005].

Kaba küme teorisi ve genetik algoritmanın avantajı temelinde, hata tespiti çıkarım modeli öneren bir yaklaşım Huang ve arkadaşları tarafından ortaya konulmuştur [Huang ve ark., 2005]. Entegre yaklaşım üç ana modülü içerir: önişleme ve kesikleştirme, GA tabanlı indirgeme ve indirgenmiş özellik seçimi, kural oluşturucu-seçici. Kaba küme tabanlı paket program ROSETTA, girdi nitelik kümesinin indirgenmesinde ve GA 'nın optimizasyon operasyonunun yürütülmesinde çalıştırılmıştır. Bu yaklaşım temelinde, minimal şart değişken alt kümeleri ve atama kuralları kurulmuş ve hata tanısı test eden anakart elektromagnetik girişim (motherboard electromagnetic interference-EMI) için bir uygulama kullanılarak örneklendirilmiştir. Sonuçlar göstermektedir ki önerilen metod anakart EMI hata tanısında kullanılan şart niteliklerinin sayısını azaltmakta ve kabul edilebilir bir sınıflandırma doğruluğu %80 göstermektedir ki hibrit model EMI tanı destek sistemleri için ümit verici bir yaklaşımdır. Bu çalışmada yeni bir algoritma geliştirilmemiş; bir uygulama çalışması yapılmıştır. Direk olarak ROSETTA paket programını kullanmakta olup genetik algoritma tabanlı nitelik indirgeme işlemi de paket program yardımıyla yapılmaktadır.

4. VERİ MADENCİLİĞİNDE SINIFLANDIRMA PROBLEMLERİ İÇİN YENİ BİR YAKLAŞIM: ROUGH-MEP ALGORİTMASI

Literatürdeki evrimsel algoritmalara dayanan çalışmalar incelendiğinde (Bölüm 2) bu algoritmaların arama uzayını geniş tutması sebebiyle avantajlı oldukları fakat oluşturdukları kuralların daha karmaşık yapıda olması nedeniyle çoğunun daha sonradan kural budama algoritmalarıda kullandıkları görülmektedir. Bu dezavantajı gidermek ve evrimsel algoritmaların avantajlarından da faydalanabilmek için kural budama yerine nitelik

indirgeme yaklaşımı kulllanma fikri doğmuştur. Bu şekilde bir entegre çalışma literatürde mevcut değildir.

Bu amaç doğrultusunda tezde, veri madenciliğinde sınıflandırma kurallarının keşfi için yeni bir algoritma – Rough-Mep Algoritması - önerilmiştir. Algoritmada verilen niteliklerin indirgenmesinde genetik algoritma tabanlı kaba küme yaklaşımı kullanılmıştır. Daha sonra bu indirgenmiş nitelikler kullanılarak evrimsel algoritma tabanlı bir yapıyla sınıflandırma kuralları keşfedilmeye çalışılmıştır. Geliştirilen algoritmanın performansı, literatürde mevcut olan ikili (binary) ve çoklu (n-ary) sınıflı veri kümelerinde test edilerek karşılaştırılmıştır.

4.1. Rough-Mep Algoritması

Rough-Mep Algoritması üç ana adımdan oluşmaktadır. İlk adımda, veri kümelerindeki sürekli nitelikler kesikleştirilmekte; ikinci adımda, kaba küme yaklaşımı kullanılarak ROSETTA paket programı yardımıyla genetik algoritma tabanlı nitelik indirgeme gerçekleştirilmekte ve son adımda, indirgenen nitelikler temel alınarak algoritmamızda girdi olarak kullanılıp, sınıflandırma kuralları keşfedilmektedir.

4.1.1. Rough-Mep algoritmasında sürekli niteliklerin kesikleştirilmesi

Veri kümelerindeki sürekli nitelikler Irani ve Fayyad tarafından 1993 yılında geliştirilen entropi tabanlı kesikleştirme metodu (MDLP) kullanılarak kesikleştirilmiştir [Fayyad ve Irani, 1993].

Fayyad ve Irani kesikli hale getirmek için bir özyineli (recursive) entropi minimizasyonu sezgisel yöntemi kullanmışlar ve sürekli uzayda üretilen aralıkların sayısını kontrol etmek için Rissanen'in En Küçük Tanım Uzunluğu İlkesini (Minimum Description Length Principle-MDLP) kullanmışlardır. Bir U örnekler kümesi, bir A özelliği ve bir T bölüntüleme sınırı verilirse, T

tarafından tüme varılan bölüntünün sınıf enformasyonu entropisi aşağıdaki biçimde tanımlanır:

$$E(A,T;U) = \frac{|U_1|}{n} Ent(U_1) + \frac{|U_2|}{N} Ent(U_2) \quad (4.1)$$

Burada n, U'daki nesnelerin sayısı ve U_1 ile U_2, c ayrımının sol tarafındaki nesnelerin kümesidir.

Verilen bir A özelliği için bütün ayrımlardan entropi fonksiyonunu minimize eden T_{min} sınırı ikili bir kesikli hale getirme sınırı olarak seçilir. Bu yöntem bir durdurma koşulu elde edilinceye kadar T_{min} tarafından tüme varılan bölüntülenmiş kümelerinin her ikisine tekrar uygulanabilir ve böylece A özelliğine göre çoklu aralıklar oluşturulur.

Fayyad ve Irani tarafından önerilen yöntemde kesikli hale getirme işleminde MDLP şu şekilde kullanılır: Sistem her ikisi de eğitme durumlarının aynı kopyaları olan bir alıcı ve bir göndericiye bölünür. Fakat alıcının, sınıflama bilgisi yoktur. Gönderici alıcıya bu eksik bilgiyi göndermelidir. MDL ilkesi, büyük sayıda istisnaları olan basit bir kuram ile az sayıda istisnaları olan karmaşık bir kuram arasındaki optimum bölmenin hem kuram hem de istisnaları şifrelemek için gerekli bit sayısının minimize edildiği durum olduğunu ifade etmektedir. MDL ilkesi sadece bir durdurma kriteri olarak kullanılır.

Entropi yöntemi önce bütün örnekleri kesikli hale getirilen özelliğe göre sıralar. Algoritma aşağıdaki adımlarla tanımlanabilir:

1. Entropi kriterine (T_a) göre en iyi ayırma noktası seçilir.
2. Ayırma noktasının MDL ilkesine göre anlamlı olup olmadığı değerlendirilir, anlamlı değilse geriye dönülür. Aksi takdirde (T_a) ayırma noktası tarafından ayrılan aralıkların her biri için kesikli hale getirme algoritması tekrarlanır.

Bu algoritmanın temel kısmı birinci adımdır. Entropi örneklerin pozitif ve negatif sınırlara dağılımının rassallık derecesinin bir ölçüsüdür. Çizelge 4.1. ‘de verilen olumsallık tablosunda C_1 kriteri için iki muhtemel aralığı gösteren iki sütun (V_1 ve V_2) vardır. C_2 kriteri karar özelliğidir. Her $D_{1.....}$ D_r kararı için tabloda bir satır vardır. n_{ij}, gözlenen V_j özellikli ve D_i sınıflı durumların sayısıdır.

Çizelge 4.1. Olumsallık tablosu

i/j	V_1	V_2	
D_1	n_{11}	n_{12}	n_1
.	.	.	.
.	.	.	.
.	.	.	.
D_r	n_{r1}	n_{r2}	n_2
	n_1	n_2	N

Çizelge 4.1’deki satır ve sütun toplamları aşağıdaki gibi tanımlanır:

$$n_i = \sum_{j=1}^{2} n_{ij} \tag{4.2}$$

$$n_j = \sum_{i=1}^{k} n_{ij} \tag{4.3}$$

$$p_{ij} = \frac{n_{ij}}{N} \tag{4.4}$$

$$p_i = \frac{n_i}{N} \tag{4.5}$$

$$p_j = \frac{n_j}{N} \tag{4.6}$$

p_{ij}, p_i ve p_j karar sisteminden yaklaşık olasılıklardır. Bu tanımları kullanarak aşağıdaki entropileri tanımlayabiliriz:

$$H_D = -\sum_{i=1}^{r} p_i \log_2 p_i \tag{4.7}$$

H_D karar sınıflarının entropisidir. Karar sınıflarının dağılımının rassallık derecesinin bir ölçüsüdür.

$$H_c = -\sum_{j=1}^{2} p_j \log_2 p_j \tag{4.8}$$

H_C özelliğin kendisinin entropisidir. H_C özelliğin enformasyon içeriğinin bir ölçüsüdür.

$$H_{DC} = -\sum_{i=1}^{r} \sum_{j=1}^{2} p_{ij} \log_2 p_{ij} \tag{4.9}$$

H_{DC} bileşik olayların sınıf-özellik değerinin entropisidir.

$$H_{D/C} = H_{DC} - H_C \tag{4.10}$$

$H_{D/C}$ verilen özellik değeri için sınıfların entropisidir.

Kullanılan basit bir ölçü H_C olabilir. Ayırma noktası olumsallık tablosunu her tablo ayırma noktası tarafından tanımlanan iki aralığın her biri için yeni bir karar sistemini gösteren tek sütunlu iki yeni tabloya bölüntülenebilir. O zaman kesikli hale getirme algoritması olumsallık tabloları için entropilerin toplamını minimize eden ayırma noktasını seçecektir. Sıkça aşağıdaki ölçüyü veren bir oran ölçüsü kullanılır.

$$E(T_a) = \frac{|U_1|}{|U|} H_C(U_1) + \frac{|U_2|}{|U|} H_C(U_2) \tag{4.11}$$

Burada $|U_1|, |U_2|$ ve $|U|$ sırasıyla iki aralığın her birindeki nesnelerin sayısını ve nesnelerin toplam kümesinin sayısını göstermektedir. Fayyad ve Irani tarafından kullanılan MDLP' ye göre durdurma kriteri aşağıdaki gibi bulunur:

$$\text{Kazanç (A,T;U)} < \frac{\log_2(N-1)}{N} + \frac{\Delta(A,T;U)}{N} \tag{4.12}$$

Burada N, U kümesindeki örneklerin sayısıdır.

Kazanç(A,T;U)=E(U)-E(A,T;U)

ve

$$\Delta(A,T;U) = \log_2(3^k - 2) - (kE(U) - k_1 E(U_1) - k_2 E(U_2)) \tag{4.13}$$

k_1 ve k_2, sırasıyla U_1 ve U_2'de bulunan karar sınıflarını göstermektedir.

4.1.2. Rough-Mep algoritmasında kaba küme yaklaşımıyla GA-tabanlı nitelik indirgemesi

Kesikleştirme işleminden sonra, bilgi tablosu GA-tabanlı indirgeme ve seçim işlemi için entegre bir sisteme gönderilir. Entegre sistem olarak kaba küme tabanlı bir yazılım olan ROSETTA paket programı kullanılmıştır. Norveç Üniversitesi Bilim ve Teknoloji'deki takım tarafından geliştirilmiş olan yazılım girdi nitelik kümesinin indirgenmesinde ve GA'nın optimizasyon işleminin yürütülmesinde kullanılmıştır. Rosetta paket programında GA tabanlı nitelik indirgemesinin paket program içerisinde nasıl yapıldığı aşağıda anlatılmıştır.

Vinterbo ve Øhrn tarafından minimal uygun kümelerin (minimal hitting sets) hesaplanması için bir GA tanımlanmıştır [Vinterbo ve Øhrn, 2000]. Algoritma

hem maliyet bilgisini hem de yaklaşık çözümleri destekler. Algoritmanın uygunluk fonksiyonu aşağıdaki şekilde tanımlanabilir:

$$f(\mathrm{B}) = (1-\alpha) \times \frac{\cos t\ (A) - \cos t\ (B)}{\cos t\ (A)} + \alpha \times \min\left\{\varepsilon, \frac{|[S \quad in \quad S \mid S \cap B \neq \varnothing]|}{|S|}\right\} \qquad (4.14)$$

Burada,

S : ayırtedilebilirlik fonksiyonuna uyan kümelerin kümesi,

α : parametresi alt küme maliyeti ve uygun bölge (hitting fraction) arasındaki ağırlığı,

ε : yaklaşık çözümlerin durumuyla alakalıdır.

A' nın B alt kümeleri evrimsel aramaya göre bulunur. En düşük ε'na sahip olanlar bir saklı listede (keep list) toplanır. Saklı listenin genişliği k olarak tanımlanır. Yaklaşık çözümler iki parametre ε ve k 'ya göre kontrol edilir. Tüm uygunluk fonksiyonu hesaplama iterasyonlarından sonra, hedef nesneye göre indirgemelerin minimum alt kümesi veri kümesinden bulunur.

4.1.3. Rough-Mep algoritmasıyla sınıflandırma kurallarının çıkarımı

Kural çıkarım adımında indirgenmiş nitelikler, Rough-Mep Algoritması'na girdi teşkil eder. Rough-Mep algoritmasında, MEPAR-miner'ın kromozom gösterimi modifiye edilmiş, algoritmik yapıya yeni operatörler eklenmiş ve varsayılan sınıf yapısı için de, yeni bir formülasyon önerilmiştir.

Rough-Mep algoritmasındaki varsayılan sınıf yapısı (Default class structure)

Literatürdeki birçok çalışmada varsayılan sınıf, veri kümesinde en çok kullanılan sınıf olarak etiketlenmiştir. Bu şekilde etiketleme hiçbir kurala uymayan test verilerinin doğru sınıflandırma performansını düşürebilmektedir.

Eğitim verisi kullanılarak her bir sınıf için yanlış olarak tanımlanmış parçaları varsayılan sınıf olarak belirlersek mümkün olduğunca fazla sayıda test verisi doğru olarak sınıflandırılabilir fikrinden yola çıkılarak geliştirilen algoritmada, yeni bir de varsayılan sınıf yapısı önerilmiştir. Önerilen varsayılan sınıf yapısının formülasyonu aşağıdadır;

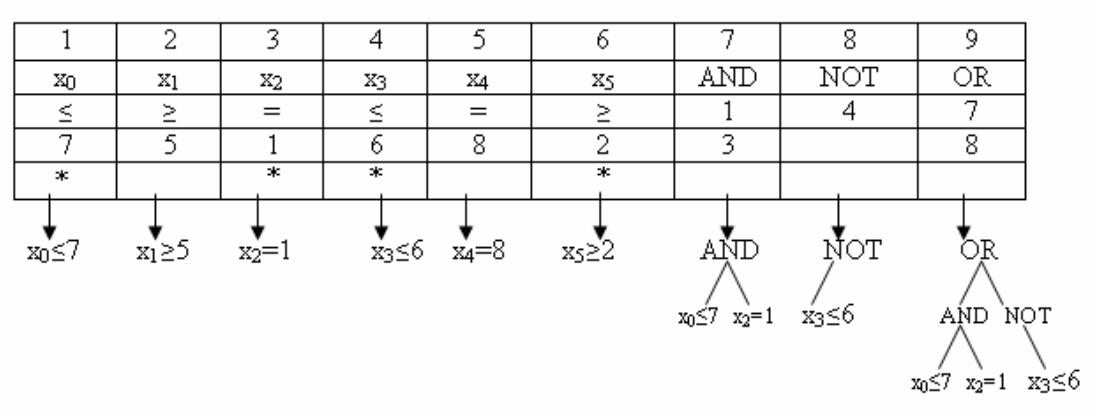
Varsayılan sınıf= $\max_i(FN_i)$, i=sınıf sayısı dır.

FN, kural tarafından kapsanmayan, sınıfı eğitim hedef sınıfıyla eşleşen örneklerin sayısını gösterir.

Önerilen varsayılan sınıf yapısının etkinliğini göstermek için, Rough-Mep algoritmasında tüm parametreler aynı bırakılıp, varsayılan sınıf yapımız bir önerdiğimiz formülasyonla bir de veri kümesinde en çok kullanılan sınıf yapısı kullanılarak etiketlenmiş elde edilen tahmini doğruluk değerleri karşılaştırılmıştır. Sayfa 90'da görüldüğü gibi önerilen formülasyon, tahmini doğruluk değerimizi arttırmaktadır.

Rough-Mep algoritmasındaki fonksiyon ve terminal kümeleri

Kromozomun ilk parçasını oluşturan herbir terminal gene rasgele bir nitelik atanır. Nitelikler atandıktan sonra niteliğin tipine (kategorik veya sürekli olmasına) göre ilişkisel operatörler belirlenir. Terminal genler yapılandırıldıktan sonra, GA tabanlı indirgenmiş nitelik kümelerinden rasgele bir indirgenmiş nitelik kümesi seçilir ve bu seçilen nitelikler kromozom yapısı üzerinde işaretlenir. Literatürden farklı olarak bu işlemin yapılmasındaki amaç, sınıflandırma performansını düşüren niteliğin karar kurallarında kullanılmasını önleyerek, gereksiz kurallardan kaçınılmasını ve sınıflandırma doğruluğumuzun arttırılmasının sağlanmasıdır. Şekil 4.1'de Rough-Mep algoritmasındaki örnek kromozom yapısı verilmiştir.



Şekil 4.1. Rough-Mep algoritmasındaki örnek kromozom yapısı

Kromozomdaki her bir genin uygunluk değeri tüm farklı sınıflar için hesaplanır. En yüksek uygunluk değerine sahip sınıf o genin sınıf etiketi olarak atanır. Kromozomdaki tüm genler için bu işlem tekrarlanır. En yüksek uygunluk değerine sahip genin uygunluk değeri ve sınıfı kromozomun uygunluk değerini ve sınıfını temsil eder.

Kural sınıflandırıcı sistemlerde birey veya parça gösterimi için iki farklı yaklaşım mevcuttur: Michigan ve Pittsburgh yaklaşımları [Freitas, 2001]. Michigan yaklaşımında, her bir birey yalnızca bir kuralla kodlanırken; Pittsburgh yaklaşımında, her bir birey kuralların bir kümesiyle kodlanır. Yapmış olduğumuz çalışmada Michigan kodlama yaklaşımı kullanılmıştır.

Rough-Mep algoritmasındaki uygunluk fonksiyonu

Kural hesaplama fonksiyonu, yalnızca doğru olarak sınıflandırılanları değil aynı zamanda yanlış olarak sınıflandırılanları da değerlendirmelidir. Bu yüzden dört muhtemel kavram tanımlanmıştır [Parpinelli ve ark., 2001]:

• Pozitif doğru (True Positive-TP), hem kural tarafından kapsanmış hem de doğru olarak sınıflandırılmış örneklerin sayısı (yani sınıfı eğitim hedef sınıfıyla eşleşiyor).

• Pozitif yanlış (False Positive-FP), kural tarafından kapsanan ancak yanlış olarak sınıflandırılan örneklerin sayısı (yani sınıfı eğitim hedef sınıfından farklı).
• Negatif doğru (True Negative-TN), kural tarafından kapsanmayan örneklerin sayısı, eğitim hedef sınıfından sınıfı farklı.
• Negatif yanlış (False Negative-FN), kural tarafından kapsanmayan örneklerin sayısı, sınıfı eğitim hedef sınıfıyla eşleşiyor.

Duyarlılık (sensitivity-S_e) ölçütü gerçekten doğru olarak sınıflandırılan pozitif örneklerin bir kesitidir. S_e , aşağıdaki gibi hesaplanır.

$$S_e = TP/(TP + FN) \tag{4.15}$$

Belirlilik (specificity-S_p) ölçütü doğru olarak sınıflandırılan negatif örneklerin bir kesitidir. S_p , aşağıdaki gibi hesaplanır.

$$S_p = TN/(TN + FP) \tag{4.16}$$

Bu kavramlar kullanılarak, uygunluk fonksiyonu şu şekilde tanımlanmıştır [Parpinelli ve ark., 2001];

$$Uygunluk = S_e \times S_p \tag{4.17}$$

Uygunluk fonksiyonunun değeri 0-1 arasındadır. Eğer kural tarafından tüm örnekler doğru olarak sınıflandırılmışsa uygunluk değeri 1'dir.

Rough-Mep algoritmasının tipi

Rough-Mep algoritmasında, durgun durum (steady state) genetik algoritma kullanılmıştır [Whitley ve Kauth, 1988]. Basit genetik algoritmada, popülasyondaki en iyi mantıksal ifadeli kromozom bir sonraki jenerasyona değişmeksizin kopyalanırken; bu algoritma yapısında, yeni bir birey

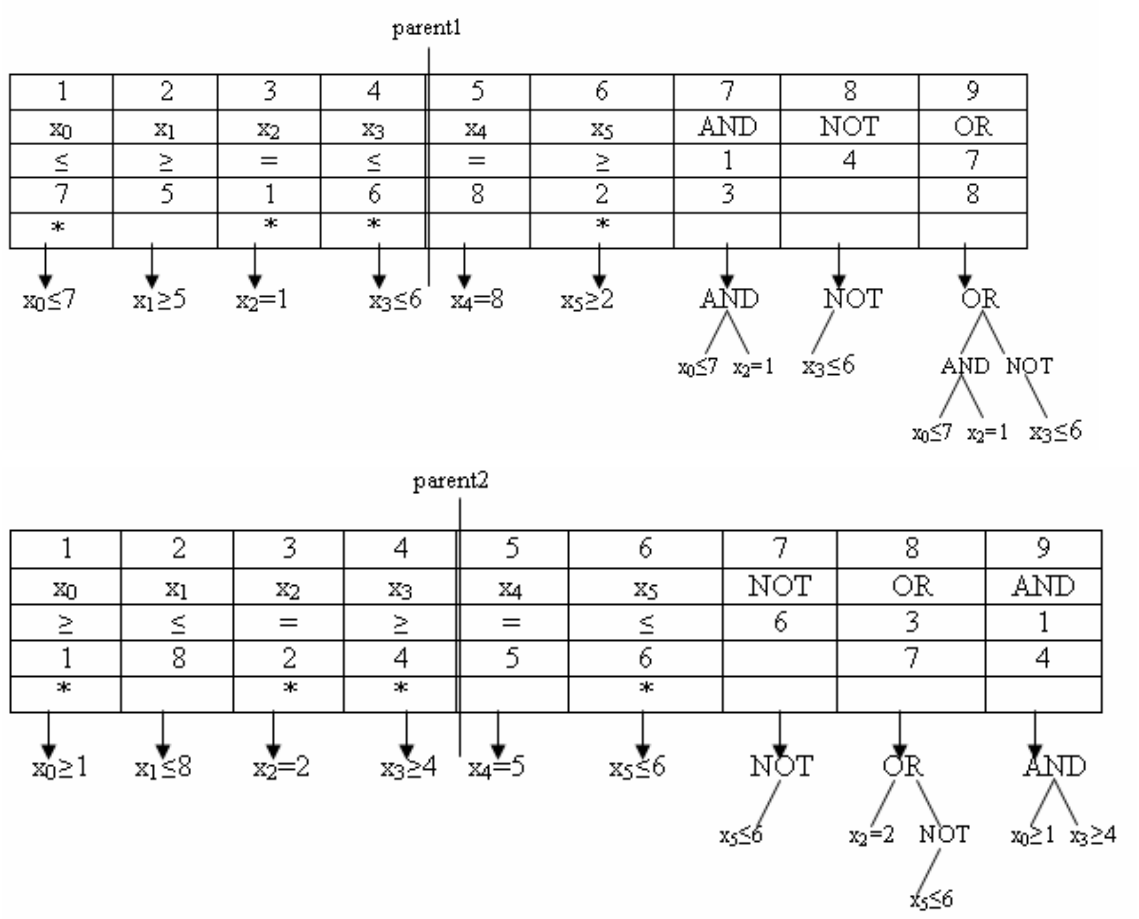
popülasyona eklendiğinde genel olarak popülasyondaki en kötü birey popülasyondan çıkartılır. Dolayısıyla bu durum, algoritmanın performansınıda olumlu yönde etkiler.

Rough-Mep algoritmasının seçim mekanizması

Değerlendirme için ikili turnuva seçim yöntemi kullanılmıştır. Bu mekanizmaya göre mevcut popülasyondan rasgele iki birey seçilerek çaprazlama operatörleri ve mutasyon operatörleri uygulanır.

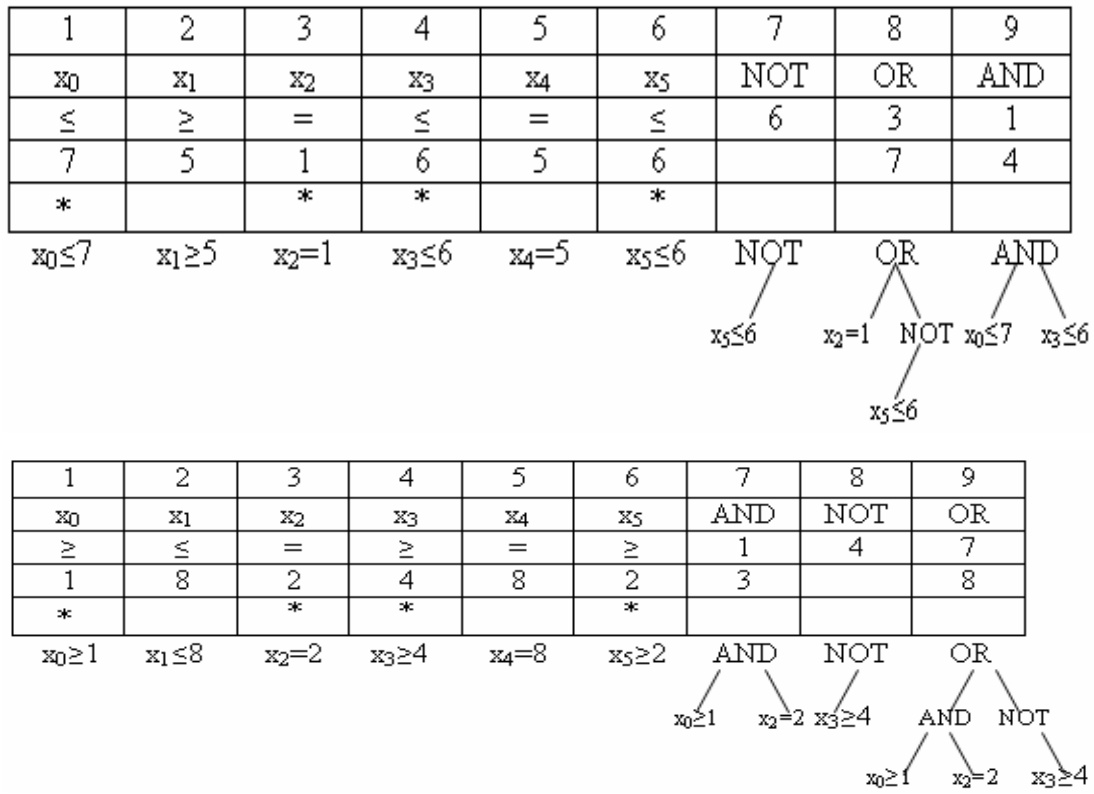
Rough-Mep algoritmasında çaprazlama operatörü

Çaprazlama esnasında önceden tanımlanmış çaprazlama olasılığına göre seçilen iki aile yeniden birleştirilir. Yapılan çalışmada tek nokta çaprazlama kullanılmıştır. Eşleştirme havuzundan rasgele iki aile kromozomu seçilir ve rasgele belirlenen çaprazlama noktasına göre yeniden birleştirilirler. Yeniden birleştirilme işlemi Şekil 4.2'de görülmektedir.



Şekil 4.2. Çaprazlamadan önce aile kromozom yapıları

Şekil 4.2'de çaprazlama noktası olarak kalın çizgiyle gösterildiği gibi aile kromozomlarının 4. pozisyonu seçilmiştir. Tek nokta çaprazlama yapıldıktan sonraki oluşan döl kromozomları Şekil 4.3'de gösterilmiştir.

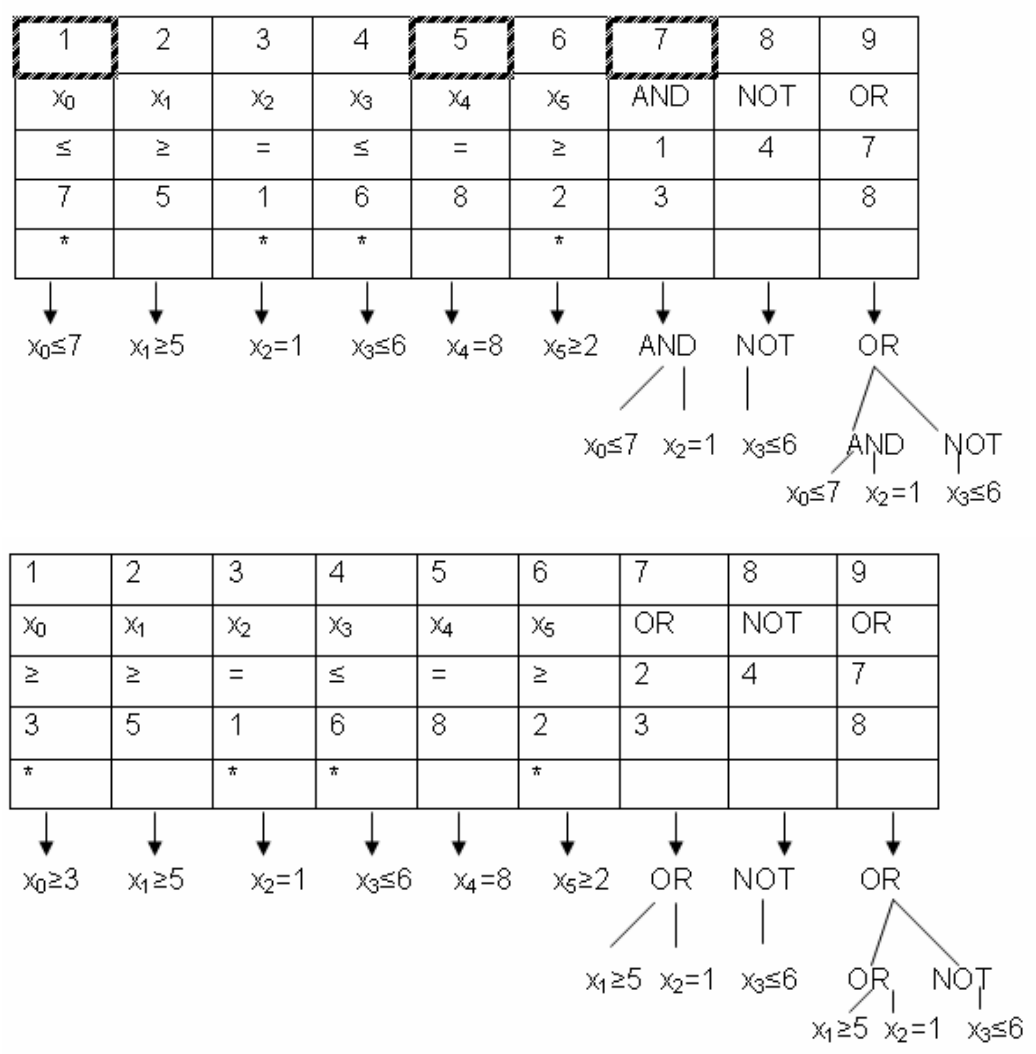


Şekil 4.3. Çaprazlama sonucu oluşan döl kromozomlar

Rough-Mep algoritmasında mutasyon operatörü

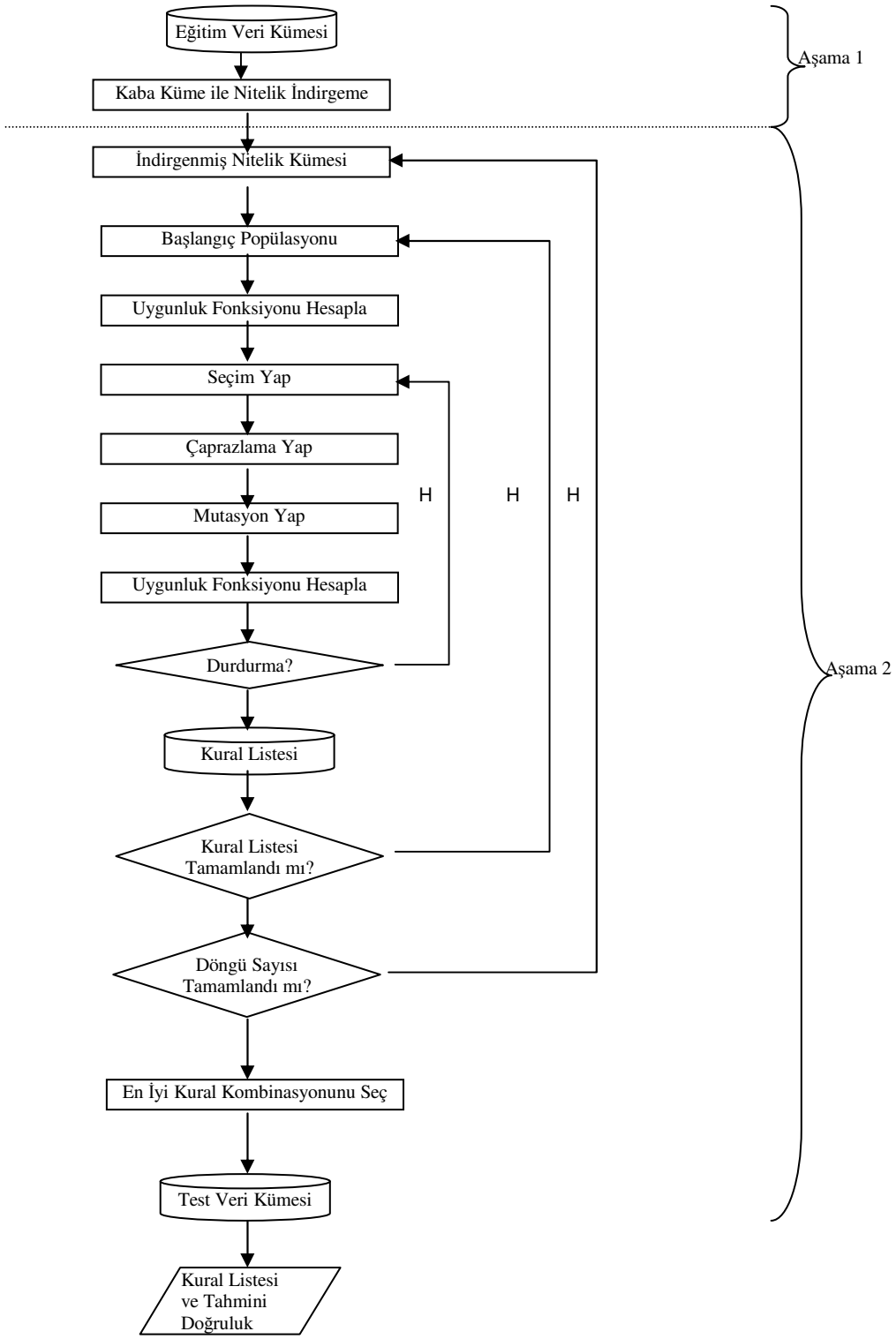
Kromozomdaki herbir sembol (terminal işaretçi, fonksiyon, fonksiyon işaretçi) mutasyon operatörünün hedefi olabilir. Kromozomdaki bazı semboller mutasyonla önceden tanımlanmış mutasyon olasılığına göre değişikliğe uğrarlar. Kromozom içerisinde rasgele mutasyon noktası ya da noktaları belirlenir. Mutasyona uğrayacak noktanın sadece terminal gen olup olmadığına bakılmaz. Aynı zamanda işaretlenmiş gen olup olmadığına da bakılır. Eğer mutasyona uğrayacak nokta terminal gen ve işaretlenmiş gen ise, terminal işaretçiler bir başka ilişkisel operatörle yer değiştirir ve nitelik değeri de nitelik değer kümesi içinde bir başka değerin değerini alır. Eğer seçilen terminal gen işaretlenmiş gen değilse, hesaplamalar üzerinde bir değişikliğe sebep olmayacağı için aynen kalır. Eğer mutasyon noktası bir fonksiyon gen ise, mantıksal fonksiyon bir başka mantıksal fonksiyonla yer

değiştirir. Şekil 4.4'de Rough-Mep algoritmasında mutasyon operatörü uygulanmadan önceki ve sonraki kromozom yapıları verilmiştir.



Şekil 4.4. Rough-Mep algoritmasında mutasyon operatörü uygulanmadan önceki ve sonraki kromozom yapıları

Rough-Mep algoritmasındaki sistemin genel yapısı aşağıda özetlenmiştir:



Şekil 4.5. Rough-Mep algoritmasının akış şeması

4.2. Deneysel Çalışma

Rough-Mep algoritmasının performansını sınamak için, UCI Machine Learning Repository'de bulunan veri kümeleri kullanılmıştır [UCI Machine Learning Repository, 2007]. Kullanılan veri kümeleri Wisconsin Breast Cancer, Ljubljana Breast Cancer, Tic-Tac-Toe, CRX, Nursery, Cleveland, Iris, Lymphography veri kümeleridir.

Wisconsin Breast Cancer Veri Kümesi (WBC)

WBCD veri kümesi 699 örneği içermektedir. Her bir örnek 9 tane sürekli nitelikten oluşmaktadır. Niteliklere 1 ve 10 arasında tamsayılı bir değer atanmıştır. Veri kümesinde 16 örnek eksik değerlidir. Eksik veri sayısının çok az olması sebebiyle bunlar veri kümesinden çıkartılmış, kalan 683 veri çalışmada kullanılmıştır. WBC veri kümesinin özellikleri Ek 1'de verilmiştir.

Ljubljana Breast Cancer Veri Kümesi (LBC)

Veri kümesi 286 örnekten oluşmaktadır. Herbir örnekte 9 kategorik nitelik mevcuttur. Veri kümesinde 9 örnek eksik değerlidir. Eksik veri sayısının çok az olması sebebiyle bunlar veri kümesinden çıkartılmış, kalan 277 veri çalışmada kullanılmıştır. LBC veri kümesinin özellikleri Ek 2'de verilmiştir.

Tic-Tac-Toe Veri Kümesi

Veri kümesi 958 örnekten oluşmaktadır. Herbir örnekte 9 kategorik nitelik mevcuttur. Veri kümesinde eksik nitelik değeri bulunmamaktadır. Tic-Tac-Toe veri kümesinin özellikleri Ek 3'de verilmiştir.

CRX Veri Kümesi

CRX veri kümesi kredi kartı uygulamalarını içerir. Bu veri kümesini kullanmanın en önemli avantajı yapısında kategorik ve sürekli nitelikleri barındırmasıdır. Veri kümesinde 690 örnek ve 15 nitelik mevcuttur. 37 tane durumda eksik nitelik değeri vardır. Yapılan çalışmada bu 37 durum veriden çıkartılmamış yalnızca eksik olarak ifade edilmiştir. CRX veri kümesinin özellikleri Ek 4'de verilmiştir.

Nursery Veri Kümesi

Veri kümesi 12960 örnekten oluşmaktadır. Herbir örnekte 8 kategorik nitelik mevcuttur. Veri kümesinde eksik nitelik değeri bulunmamaktadır. Nursery veri kümesinin özellikleri Ek 5'de verilmiştir.

Cleveland Heart Disease Veri Kümesi

Veri kümesi 303 örnekten oluşmaktadır. Bu veri tabanı 75 nitelik içerir fakat literatürde yapılan çalışmalarda bunlardan yalnız 13'ü kullanılmıştır. 6 tane eksik nitelik değeri mevcuttur. Yapılan çalışmada bu 6 durum veriden çıkartılmamış yalnızca eksik olarak ifade edilmiştir. Cleveland veri kümesinin özellikleri Ek 6'da verilmiştir.

Lymphography Veri Kümesi

Veri kümesi 148 örnekten oluşmaktadır. Herbir örnekte 18 nitelik mevcuttur. Veri kümesinde eksik nitelik değeri bulunmamaktadır. Lymphography veri kümesinin özellikleri Ek 7'de verilmiştir.

Iris Veri Kümesi

Veri kümesi 150 örnekten oluşmaktadır. Herbir örnekte 4 nitelik mevcuttur, tüm nitelikler süreklidir. Veri kümesinde eksik nitelik değeri bulunmamaktadır. Iris veri kümesinin özellikleri Ek 8'de verilmiştir. Veri kümelerinin ana özellikleri Çizelge 4.2'de özetlenmiştir.

Çizelge 4.2. Veri kümelerinin ana özellikleri

Veri Kümesi	nesneler	#kategorik nitelikler	#sürekli nitelikler	#sınıflar
Wisconsin breast cancer	683	-	9	2
Ljubljana breast cancer	277	9	-	2
Tic-Tac-Toe	958	9	-	2
CRX	690	9	6	2
Nursery	12960	8	-	5
Cleveland heart disease	303	8	5	5
Lymphography	148	18	-	4
Iris	150	-	4	3

Sınıflandırma ölçütlerinin tahmini doğruluğu (predictive accuracy), eşitlik (4.18)'de verildiği gibi, doğru sınıflandırılmış örneklerin oranıdır [Tan ve Gilbert, 2003].

$$Dogruluk = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.18}$$

Literatürde geliştirilen algoritmaların geçerliliğinin test edilmesi gerekir. Bunun için çeşitli yöntemler mevcuttur.

Rough-Mep algoritmasının geçerliliğinin testi için on katlı çapraz geçerlilik testi uygulanmış; yukarıdaki veri kümelerinin tahmini doğruluğu 10 katlı çapraz doğrulama (10-fold cross validation) işlemi ile ölçülmüştür.

10 katlı çapraz doğrulama işlemine göre, her bir veri kümesi 10 parçaya bölünmüş, her bir parça için algoritma 10 kez çalıştırılmıştır. Her bir seferde test kümesi olarak farklı bir parça kullanılmış, kalan 9 parça eğitim için kullanılmıştır. 10 kez çalıştırma sonucu bulunan max, min ortalama değerler ve standart sapmalar not edilmiştir.

Rough-Mep algoritması, C# programlama dilinde kodlanmıştır. Hazırlanan programın görsel gösterimi Ek 9'da verilmiştir.

Çizelge 4.3'de yukarıda tanımlanan 8 veri kümesi için 10 katlı çapraz doğrulama işleminin sonuçları özetlenmiştir.

Çizelge 4.3. Rough-Mep algoritmasının tahmini doğruluk değerleri

Veri Kümeleri	*Max Tahmini Doğruluk*	*Ortalama Tahmini Doğruluk*	*Min Tahmini Doğruluk*	*Standart Sapma (%)*
CRX	1	**0,978**	0,956	1.71
Nursery	1	**0,996**	0,986	0.52
Iris	1	**0,973**	0,933	5.44
Ljubljana BC	1	**0,928**	0,893	4.45
Tic-Tac-Toe	0,947	**0,915**	0,875	2.53
Wisconsin BC	1	**0,997**	0,985	0.62
Cleveland HD	0,933	**0,916**	0,879	3.96
Lymphography	1	**0,938**	0,767	8.16

Çizelge 4.3 incelendiğinde, max satırı, 10 katlı çapraz doğrulama işlemi sonucunda bulunan en yüksek tahmini doğruluk değerini göstermektedir. Yani CRX veri kümesi için max'ın 1 olması bu verilere göre %100 doğru sınıflandırma yapıldığını; Tic-Tac-Toe veri kümesi için 0,94791 olması, ~% 95 doğru sınflandırma yapıldığını göstermektedir. min satırı, 10 katlı çapraz doğrulama işlemi sonucunda bulunan en düşük tahmini doğruluk değerini göstermektedir. Yani CRX veri kümesi için min'in 0,95588 olması bu verilere göre ~% 96 doğru sınıflandırma yapıldığını; Tic-Tac-Toe veri kümesi için 0,875 olması ~% 88 doğru sınflandırma yapıldığını göstermektedir. Ortalama satırı 10 katlı çapraz doğrulama işlemi sonucunda bulunan tüm sonuçların ortalamasını göstermektedir. Yani CRX veri kümesi için ortalamanın 0,97815 olması bu verilere göre ortalama ~% 98 doğru sınıflandırma yapıldığını; Tic-Tac-Toe veri kümesi için 0,91542 olması ortalama ~% 91 doğru sınflandırma yapıldığını göstermektedir. Standart sapma ise elde edilen sonuçların standart sapmasını ifade etmektedir.

Çizelge 4.3.'de C# programlama dilinde kodlanan Rough-Mep algoritması ile çözülen veri kümelerinin, ayrı ayrı sonuçları Ek 10-17'de verilmiştir.

Rough-Mep algoritmasında önerilen varsayılan sınıf yapısı kullanılmayıp, veri kümesinde en çok kullanılan sınıf yapısı kullanılarak etiketleme yapıldığında oluşan tahmini doğruluk değerlerimiz Çizelge 4.4'de verilmiştir.

Çizelge 4.4. Rough-Mep algoritmasında, varsayılan sınıf yapısı olarak TP kullanıldığında oluşan tahmini doğruluk değerleri

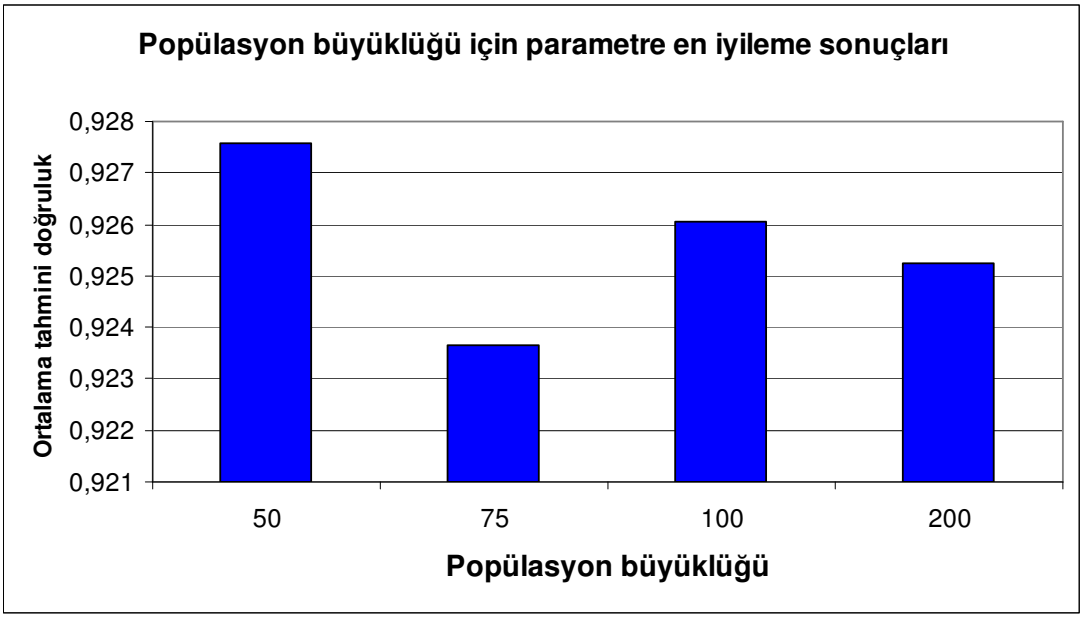
Veri Kümeleri	*Max Tahmini Doğruluk*	*Ortalama Tahmini Doğruluk)*	*Min Tahmini Doğruluk*	*Standart Sapma (%)*
CRX	1	**0,966**	0,956	1,83
Nursery	1	**0,985**	0,976	0,74
Iris	1	**0,954**	0,913	6,01
Ljubljana BC	1	**0,908**	0,866	4,56
Tic-Tac-Toe	0,927	**0,901**	0,875	2,97
Wisconsin BC	1	**0,991**	0,985	0,67
Cleveland HD	0,933	**0,909**	0,879	4,01
Lymphography	1	**0,927**	0,741	8,21

Çizelge 4.3 ve Çizelge 4.4 karşılaştırıldığında önerilen varsayılan sınıf yapısının daha etkin çıktığı görülmektedir. Bu yapı geliştirmiş olduğumuz algoritmamızın performansını da arttırmaktadır.

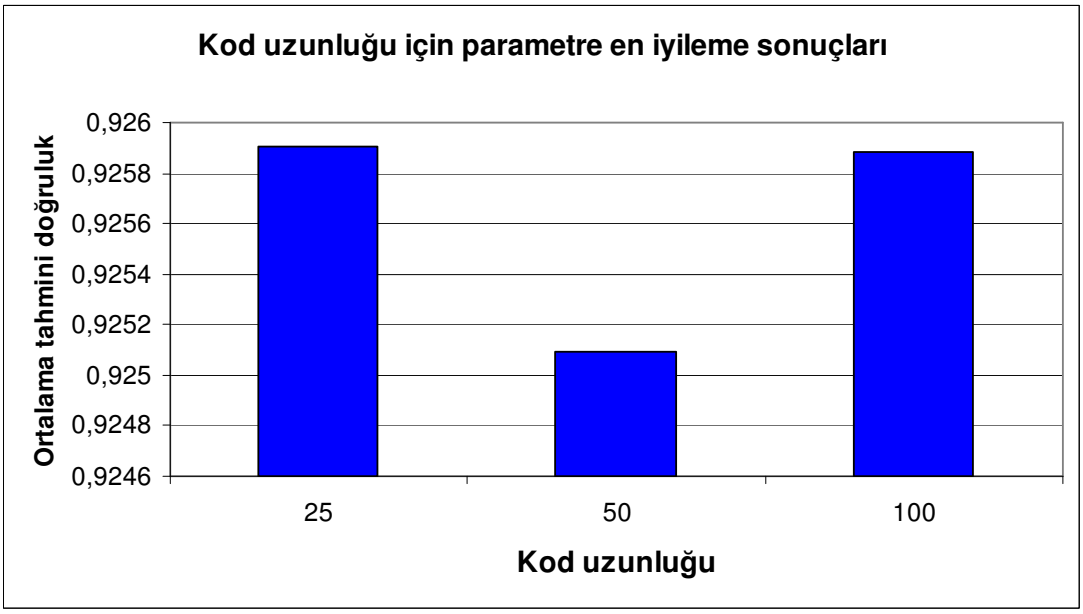
4.3. Deneysel Düzenek

Geliştirilen Rough-Mep algoritmasında kullanılacak olan parametreleri belirlemek için parametre optimizasyon işlemi yapılmıştır. Bunun için Hepatit veri kümesi kullanılmıştır. Hepatit veri kümesinde; popülasyon büyüklüğü 50-75-100-200, kromozom uzunluğu 25-50-100, mutasyon oranı 0.1-0.2-0.3, jenerasyon sayısı 100-200-250 değerleri kullanılarak Rough-Mep algoritmasının parametre kümesi bulunmuştur.

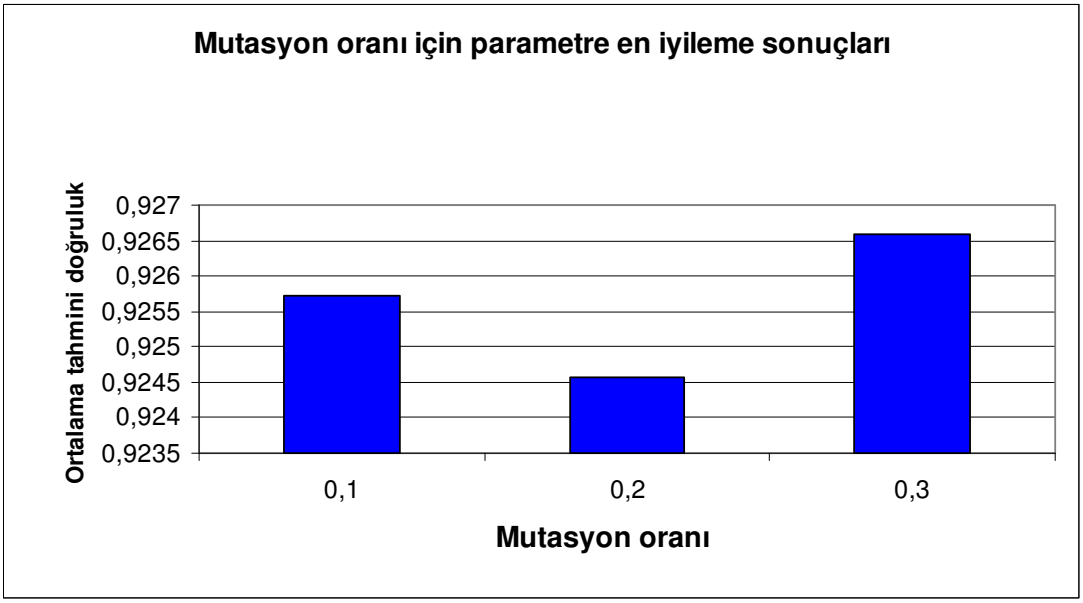
Aşağıda Şekil 4.6'da popülasyon büyüklüğü için, Şekil 4.7'de kromozom uzunluğu için, Şekil 4.8'de mutasyon olasılığı için, Şekil 4.9'da jenerasyon sayısı için en iyileme sonuçları verilmiştir.



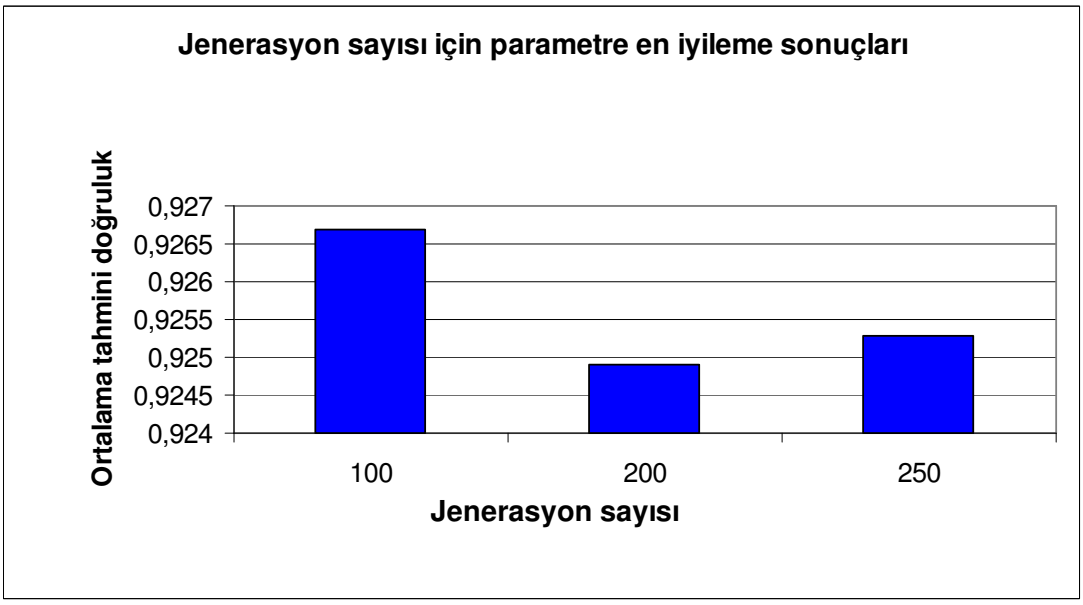
Şekil 4.6. Popülasyon büyüklüğü için parametre en iyileme sonuçları



Şekil 4.7. Kod uzunluğu için parametre en iyileme sonuçları



Şekil 4.8. Mutasyon oranı için parametre en iyileme sonuçları



Şekil 4.9. Jenerasyon sayısı için parametre en iyileme sonuçları

Elde edilen bu değerlere göre Rough-Mep algoritmasının parametre kümesi belirlenmiştir.

Algoritma Tipi	:	Durgun durum genetik algoritma
Seçim Mekanizması	:	İkili Turnuva Seçimi

Popülasyon genişliği	:	50
Kromozomun Uzunluğu	:	25
Çaprazlama Operatörü	:	Tek Nokta Çaprazlama
Çaprazlama Olasılığı	:	1
Mutasyon Olasılığı	:	0,30
Jenerasyon Sayısı	:	100
İç Döngü Sayısı	:	10

Yukarda tanımlanan tüm parametreler tüm veri kümeleri için Rough-Mep algoritmasıyla çözümde kullanılmıştır.

4.4. Rough-Mep Algoritmasının Performansının Karşılaştırması

Rough-Mep algoritmasının performansının karşılaştırması, hem klasik makine öğrenimi algoritmalarıyla; hemde literatürde en iyi sonuç veren algoritmalarla yapılmıştır. Genellikle literatürde performans karşılaştırması, sadece diğer çalışmaların sonuçlarının karşılaştırılması şeklinde yapılmaktadır. Fakat bu karşılaştırmalarda aynı 10 katlı değerler kullanılmamaktadır. Bu yüzden Rough-Mep algoritmasının performans karşılaştırması, aynı 10 katlı değerler kullanılarak aynı şartlar altında klasik makine öğrenimi algoritmalarıyla da yapılmıştır.

4.4.1. Klasik makina öğrenimi algoritmalarıyla karşılaştırma

Rough-Mep algoritmasının performansının karşılaştırması için veri kümelerine, kural tabanlı makina öğrenimi algoritmalarından J48 [Witten ve Frank, 1999] ve PART [Frank ve Witten, 1998] algoritması; istatistiksel sınıflandırıcı olarak da NaiveBayes [John ve Langley, 1995] uygulanmıştır. Sonuçları aynı şartlar altında karşılaştırabilmek için aynı 10 katlı çaprazlama işlemi uygulanmıştır. Test verisi için minimum, maksimum, ortalama doğruluklar ve standart sapma değerleri Çizelge 4.5'de belirtilmiştir.

Literatürde algoritmaların karşılaştırması için max, min ve ortalama değerleri hesaplanmakta fakat karşılaştırma ortalama değerlere göre yapılmaktadır.

Çizelge 4.5 ortalama değerlere göre incelendiğinde, sekiz veri kümesinin yedisinde Rough-Mep algoritmasının performansı diğer algoritmalardan yüksek çıkmıştır (Çizelge 4.5'de koyu yazılanlar). Yalnızca TicTacToe veri kümesi için algoritmamızın ortalaması (91,542), PART algoritmasının ortalamasından (93,85296) çok az düşük çıkmıştır. Fakat aynı veri kümesi için standart sapma Rough-Mep algoritmasında daha küçüktür.

Çizelge 4.5. Rough-Mep algoritmasının klasik makina öğrenimi algoritmalarıyla karşılaştırma sonuçları

Veri Kümesi		NaiveBayes	PART	C4.5	Rough-mep
Ljubljana Breast Cancer	Ortalama	72,69704	69,4064	74,28079	**92,857**
	Standart Sapma	7,737207	7,628285	6,050664	4,45
	Minimum	51,72414	48,27586	53,57143	89,285
	Maximum	89,65517	86,2069	86,2069	100
Iris	Ortalama	94	94,67	93,33	**97,333**
	Standart Sapma	4,92	6,13	5,44	5,44
	Minimum	86,67	86,67	86,67	93,333
	Maximum	100	100	100	100
Wisconsin Breast Cancer	Ortalama	97,19648	94,69462	95,00828	**99,705**
	Standart Sapma	1,711341	2,514011	2,730619	0,62
	Minimum	92,85714	88,57143	87,14286	98,529
	Maximum	100	100	100	100
Nursery	Ortalama	90,50021	98,66554	96,21913	**99,560**
	Standart Sapma	0,389248	0,354198	0,291374	0,52
	Minimum	90,04083	97,91146	95,82293	98,611
	Maximum	91,17113	99,13774	96,64094	100
TicTacToe	Ortalama	69,64232	**93,85296**	85,28103	91,542
	Standart Sapma	4,402807	3,076832	3,184181	2,53
	Minimum	58,94737	83,33333	75,78947	87,5
	Maximum	82,10526	100	93,75	94,791
Cleveland Heart Disease	Ortalama	56,38065	51,40215	52,05914	**91,616**
	Standart Sapma	7,13104	7,636893	6,688745	3,96
	Minimum	40	25,80645	33,33333	87,878
	Maximum	70	70	66,66667	93,333
Lymphography	Ortalama	84,26	80,26	78,15	**93,794**
	Standart Sapma	9,03	8,49	10,63	8,16
	Minimum	69,23	69,23	61,54	76,666
	Maximum	100	93,33	100	100
CRX	Ortalama	77,85507	84,44928	85,56522	**97,815**
	Standart Sapma	4,181567	4,349558	3,95653	1,71
	Minimum	66,66667	73,91304	73,91304	95,588
	Maximum	86,95652	94,2029	92,75362	100

4.4.2. Diğer kural tabanlı sınıflandırma algoritmalarıyla karşılaştırma

Rough-Mep algoritmasının performansı literatürde mevcut olan diğer kural tabanlı sınıflandırıcılar ile karşılaştırılmış ve sonuçlar Çizelge 4.6'da verilmiştir.

Çizelge 4.6 incelendiğinde, sekiz veri kümesi içinde karşılaştırıldığı algoritmaların hepsinde Rough-Mep algoritmasının daha iyi sonuç verdiği görülmektedir. Rough-Mep algoritmasının en düşük değeri Cleveland HD veri setinde 91,62±3,96; en yüksek değeri Wisconsin BC veri setinde 99,71±0,62 dir.

Çizelge 4.6. Rough-Mep algoritması ile veri madenciliği algoritmalarının tahmini doğruluklarının karşılaştırılması

Veri Kümesi	Sınıflandırıcı	Doğruluk
LjubljanaBC	Ant-Miner[*]	75,28±2,24
	CN2[*]	67,69±3,59
	MEPAR-miner[**]	90,63±4,48
	Rough-Mep Algorithm	**92,86±4,45**
Wisconsin BC	Ant-Miner[*]	96,04±0,93
	CN2[*]	94,88±0,88
	MEPAR-miner[**]	99,41±0,76
	Rough-Mep Algorithm	**99,71±0,62**
Tic-tac-toe	Ant-Miner[*]	73,04±2,53
	CN2[*]	**97,38±0,52**
	MEPAR-miner[**]	94,47±1,31
	Rough-Mep Algorithm	91,54±2,53
Cleveland HD	Ant-Miner[*]	59,67±2,50
	CN2[*]	57,48±1,78
	MEPAR-miner[*]	87,78±3,51
	Rough-Mep Algorithm	**91,62±3,96**
CRX	C4.5[***]	91,79±2,1
	Double C4.5[***]	90,78±1,2
	C4.5/AG[***]	91,66±1,8
	MEPAR-miner[**]	96,96±2,50
	Rough-Mep Algorithm	**97,82±1,71**
Nursery	C4.5[***]	95,4±1,2
	Double C4.5[***]	97,23±1,0
	C4.5/AG[***]	96,77±0,7
	MEPAR-miner[**]	95,83±1,80
	Rough-Mep Algorithm	**99,56±0,52**
Iris	DCC[****]	96,73
	GP-Co[****]	95,3
	GGP[****]	91,04
	Rough-Mep Algorithm	**97,33±5,44**
Lymphography	CN2[*****]	81,6
	MLP[*****]	81,6
	DIMLP[*****]	80,4
	SIM[*****]	86,2
	Rough-Mep Algorithm	**93,79±6,1**

(±'ler standart sapmaları ifade etmektedir.)

*[Parpinelli ve ark., 2002] **[Baykasoglu ve Özbakır, 2006] ***[Carvalho ve Freitas, 2002] ****[Tan ve ark., 2005] *****[Luukka, 2006] *, **, ***, ****, ***** işaretleri kıyasladığımız sonuçların alındığı makaleleri belirtmektedir.

5. SONUÇ VE ÖNERİLER

Son birkaç yıldır veritabanlarından bilgi keşfi çok ilgi çekici bir disiplin olmuştur. Bilgi keşfinin amacı, çok büyük veri tabanlarından bilgi parçalarının veya kalıplarının çıkartılmasıdır.

Kaba küme yöntemi, büyük ve tamamlanmamış veri kümelerinden bilgi keşfi için güçlü bir araçtır. Kaba küme teorisi temelinde verilen örnekleri sınıflandırmak için çok sayıda algoritma ve sistemler geliştirilmiştir. Fakat kaba küme teorisinin nitelik indirgeme kabiliyeti yüksek olmakla birlikte, sınıflandırma performansı düşüktür. Bu da bu yaklaşımın tek başına sınıflandırmada kullanılması yerine başka algoritmalarla entegre edilmesi ihtiyacını doğurmaktadır.

Literatürde diğer bölümlerde de bahsedildiği gibi veri madenciliği ve bilgi keşfinde çeşitli evrimsel algoritmalarda çok fazla kullanılmıştır. Evrimsel algoritmalar arama uzayını arttırdıkları ve niteliklerinde farklı kombinasyonlarını oluşturmada başarılı oldukları için etkin kurallar üretebilmektedir. Fakat bu algoritmalar tüm nitelikleri kullanarak kural oluşturdukları için oluşturdukları kurallar daha karmaşık yapıda ve çözüm süreleri de daha uzun olabilmektedir.

Yapılan çalışmada, sınıflandırma kurallarının keşfedilmesinde genel arama yeteneği yüksek olması ve nitelikler arasındaki etkileşimleri dikkate alabilmesi sebebiyle evrimsel algoritmalar uygun bir yöntem olarak seçilmiştir. Literatürde bulunan çalışmalar incelendiğinde, kaba küme kullanılarak nitelik indirgenmesinin yapıldığı ve indirgenmiş bu niteliklerle evrimsel algoritmaların kural çıkarımında kullanıldığı bir yapıya rastlanmamıştır. Yapılan tezde, kural çıkarımında yani sınıflandırmada, kaba küme yaklaşımı ve evrimsel algortimaların avantajlarından faydalanabilmek; dezavantajlarını ortadan kaldırabilmak için yeni bir algoritma Rough-Mep algoritması geliştirilmiştir.

Geliştirilen algoritma da ilk adımda, veri kümelerindeki sürekli nitelikler kesikleştirilmiş; ikinci adımda, kaba küme yaklaşımı kullanılarak genetik algoritma tabanlı nitelik indirgeme gerçekleştirilmiş ve son adımda, indirgenen nitelikler temel alınarak algoritmada girdi olarak kullanılıp, sınıflandırma kuralları keşfedilmiş ve ayrıca yeni bir varsayılan sınıf yapısı da önerilmiştir.

Literatürdeki çalışmalarda varsayılan sınıf olarak en çok kullanılan sınıf yapısı (TP) etiketlenmektedir. Bu şekilde etiketleme hiçbir kurala uymayan test verilerinin doğru sınıflandırma performansını düşürebildiğinden; tezde yeni bir sınıf yapısı da önerilmiştir. Önerilen varsayılan sınıf yapısı, en fazla yanlış olarak tanımlanmış parçaların bulunduğu sınıf (max FP) olarak belirlenmektedir. Rough-Mep algoritmamızın önerilen varsayılan sınıf yapısı ile literatürdeki varsayılan sınıf yapısı, diğer parametreler aynı bırakılıp Rough-Mep algoritmasında kullanılarak sonuçlar karşılaştırılmıştır. Sonuçlar değerlendirildiğinde önerilen varsayılan sınıf yapısının daha iyi sonuç verdiği görülmektedir (BKZ Çizelge4.3-Çizelge4.4).

Rough-Mep algoritmasının deney düzeneğinin hazırlanmasında, UCI veri tabanından alınan hepatit veri kümesi kullanılmıştır. Bu veri kümesi üzerinde parametre optimizasyon işlemi yapılmış; bulunan en iyi parametre değerleri algoritmanın performansının değerlendirilmesinde kullanılmıştır.

Rough-Mep algoritmasının programlanmasında C# programlama dili kullanılmıştır. Hazırlanan programda parametre optimizasyon işlemi içinde bir tool bulunmaktadır.

Geliştirilen algoritmanın performansını sınamak için hepatit veri kümesinden elde edilen deney düzeneği kullanılarak UCI veri tabanından alınan 8 veri kümesi üzerinde çalışılmıştır. Bu veriler veri ön hazırlama işlemlerine tabi tutulup düzenlendikten sonra kullanılarak algoritma çalıştırılmış, sonuçlar hem aynı 10 katlar kullanılarak klasik makine öğrenimi algoritmalarıyla hem

de literatürdeki diğer algoritmalarla karşılaştırılmış ve oldukça iyi sonuçlar elde edilmiştir. Sekiz veri kümesinin yedisinde Rough-Mep algoritmasının performansı diğer algoritmalardan yüksek çıkmıştır.

Program parametre optimizasyonu yapılmadan da çalıştırıldığında algoritmanın performansı benzer çıkmaktadır.

Kaba küme yaklaşımlarında veriler kesikleştirilirken bazı bilgiler kaybolabilmektedir. Bunu önleyebilmek için nitelik indirgeme kısmında kullanılan kaba küme yaklaşımı bulanıklaştırılabilir veya elde edilen kurallar bulanık olarak ifade edilmeye çalışılabilir. Gelecekte bu düşünceler ışığında yeni algoritmalar oluşturulmaya çalışılacaktır.

KAYNAKLAR

Agrawal, R., Imielinski, T., Swami, A.N., "Mining association rules between sets of items in large databases", ***International Conference on Management of Data***, Washington, 207-216 (1993).

Akpınar, H., "Veri tabanlarında bilgi keşfi ve veri madenciliği", ***İstanbul Üniversitesi İşletme Fakültesi Dergisi***, 29(1): 1-22 (2000).

Alexander, S., "Users find tangible rewards digging into data mines", ***InfoWorld***, 19: 61 (1997).

Ali, J.M.H., "A hybrid naive bayesian classifier with decision trees algorithm for classification problems", ***WSEAS Transactions on Systems***, 3(3): 1286-1291 (2004).

Almuallim, H. And Dietterich, T.G., "Learning with many irrelevant features", ***In the 9th National Conference on Artificial Intelligence***, USA, 547–552 (1991).

Alonso, F., Caraça-Valente¸ J.P., Gonza´lez, A.L. And Montes, C., "Combining expert knowledge and data mining in a medical diagnosis domain", ***Expert Systems with Applications***, 23: 367–375 (2002).

Anand, S.S. And Hughes, J.G., "Hybrid data mining systems: The next generation", ***Research and Development in Knowledge Discovery and Data Mining***, Second Pasific- Asia Conference, Australia, 13-24 (1998).

Ananthanarayana, V.S., Narasimha, M.M. And Subramanian, D.K.,"Tree structure for efficient data mining using rough sets", ***Elsevier Science***, 24: 851-862 (2002).

Baykasoglu, A. And Ozbakir, L., "MEPAR-miner: Multi-expression programming for classification rule mining", ***European Journal of Operational Research***, 183(2): 767-784 (2007).

Bhaskar, T. And Kamath, B.N., "A rough neuro data mining approach for network intrusion detection", ***Lecture Notes in Computer Science***, 3326: 534 (2004).

Bojarczuk, C.C., Lopes, H.S. And Freitas, A.A., "Genetic programming for knowledge discovery in chest pain diagnosis" ***IEEE Engineering in Medicine and Biology magazine - Special Issue on Data Mining and Knowledge Discovery*** , 19(4): 38-44 (2000).

Bojarczuk, C.C., Lopes, H.S. And Freitas, A.A., Michalkiewicz, E.L., "A constrained-syntax genetic programming system for discovering classification rules: application to medical data sets", ***Artificial Intelligence in Medicine***, 30: 27-48 (2004).

Bozdogan, H., "Statistical data mining and knowledge discovery", ***John Wiley***, New York, 25-27 (2003).

Brameier, M. And Banzhaf, W., "A comparison of linear genetic programming and neural networks in medical data mining", ***IEEE Transactions on Evolutionary Computation***, 5(1): 17-26 (2001).

Bransten, L., "Technology-power tools-looking for patterns: Data mining enables companies to better manage the reams of statistics they collect; the goal:spot the unexpected" ***Wall Street Journal-Eastern Edition***, 7: 1-16 (1999).

Buntine, W., "A guide to the literature on learning probabilistic networks from data", ***IEEE Transactions on Knowledge and Data Engineering***, 8(2): 195-210 (1996).

Busse, J. And Siddhaye, S., "Rough set approaches to rule induction from incomplete data", ***The 10th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems***, Italy, 923-930 (2004).

Carvalho, D.R. And Freitas, A.A., "New results for a hybrid decision tree/genetic algorithm for data mining", ***4th Int. Conf. on Recent Advances in Soft Computing (RASC-2002),*** Nottingham Trent University, 260-265 (2002).

Carvalho, D.R. And Freitas, A.A., "A genetic algorithm with sequential niching for discovering small-disjunct rules", ***Proceedings Genetic and Evolutionary Computation Conference (GECCO-2002)***, New York, 1035-1042 (2002).

Chan, C., "A rough set approach to attribute generalization in data mining", ***Journal of Information Sciences***, 107: 169-176 (1997).

Chen, Z., "Data Mining and Uncertain Reasoning: An Integrated Approach 2nd ed.", ***Wiley***, England, 200-212 (2001).

Chen, J., Chen, J. And Kemp, G.P., "Fuzzy clustering and decision tree learning for time-series tidal data classification", ***Proceedings of the 12th IEEE International Conference on Fuzzy Systems***, 1: 732-737 (2003).

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

Chouchoulas, A., And Shen, Q., "Rough set-aided keyword reduction for text categorisation", ***Applied Artificial Intelligence***, 15(9): 843–873 (2001).

Ciesielski, V. And Palstra, G., "Using a hybrid neural/expert system for data base mining in market survey data", ***Second International Conference on Knowledge Discovery and Data Mining***, Canada, 38-43 (1996).

Davis, B., "Data mining transformed", ***Information Week***, 751: 86 (1999).

De Falco, I., Cioppa, A.D. And Tarantino, E., "Discovering interesting classification rules with genetic programming" ***Applied Soft Computing***, 1: 257-269 (2002).

Dong, J., Zhong, N., And Ohsuga, S., "Using Rough Sets with Heuristics for Feature Selection", ***7th International Workshop (RSFDGrC 99)***, Japan, 178–187 (1999).

DuMouchel, W., "Bayesian data mining inlarge frequency tables with an application to the FDA spontaneous", ***American Statistician***, 53(3): 177 (1999).

Eggermont, J., Eiben, A.E. And Hemert, J.I., "A comparison of genetic programming variants for data classification", ***Lecture Notes in Computer Science***, ***Proceedings of the Third International Symposium on Advances in Intelligent Data Analysis***, 1642: 281-290 (1999).

Engelbrecht, A.P. "Computational Intelligence An Introduction" ***John Wiley & Sons***, England, 59 (2002).

Ergüneş, H.F., "Genetik algoritmaların veri madenciliğinde kullanılmasıyla ilginç kuralların bulunması", Yüksek Lisans Tezi, ***Hacettepe Üniversitesi Fen Bilimleri Enstitüsü***, Ankara, 25-26 (2004).

Fayyad, U., Piatetsky-Shapiro, G. And Smyth, P., "The KDD process for extracting useful knowledge from volumes of data", ***Communications of the ACM***, 39(11): 27-34 (1996).

Fayyad, U., “The digital physics of data mining”, ***Communications of the association for computing machinery***, 44(3): 62-65 (2001).

Fayyad, U. And Irani, K.B., “Multi-interval discretization of continuous-valued attributes for classification learning”, ***Proc. of 13th International Joint Conference on Artificial Intelligence***, 1022-1027 (1993).

Felix, R. And Ushio, T., ”Rule induction from inconsistent and incomplete data using rough sets”, ***Systems, Man, and Cybernetics***, 5: 154-158 (1999).

Feng, Y.L. And McClean, S., “A data mining approach to the prediction of corporate failure”, ***Knowledge-Based Systems***, 14(3-4): 189-195 (2001).

Fidelis, M.V., Lopes, H.S., And Freitas, A.A., “Discovering comprehensible classification rules with a genetic algorithm” ***Proc. Congress on Evolutionary Computation (CEC-2000)***, USA, 805-810 (2000).

Fodor, I.K, “A survey of dimension reduction techniques”, ***Technical Report-Lawrence Livermore National La***, UCRL-ID-148494 (2002).

Frank, E. And Witten, I.H., “Generating accurate rule sets without global optimization” ***15th Int. Conf. Machine Learning (ICML’98)***, New Zealand, 144–151 (1998).

Freitas, A.A., “A survey of evolutionary algorithms for data mining and knowledge discovery”, ***Advances in Evolutionary Computation***, 3: 819-845 (2002).

Freitas, A. A., “A genetic programming framework for two data mining tasks: classification and generalized rule induction”, ***2nd Annual Conf. Genetic Programming***, California, 96-101 (1997).

Freitas, A., “Survey of evolutionary algorithms for data mining and knowledge discovery”, ***Advances in Evolutionary Computation***, 1582: 819-845 (2001).

Friedman, J.H. And Tukey, J.W., “A projection pursuit algorithm for exploratory data analysis”, ***IEEE Transactions on Computers***, C-23(9): 881–890 (1974).

Gen, M. And Cheng, R., “Genetic algorithms and engineering optimization”, ***Wiley-Interscience Publication***, New York , 11-14 (2000).

Geng, Z. And Zhu, Q., “A new rough set based heuristic algorithm for attribute reduct”, ***Intelligent Control and Automation***, 1: 3085-3089 (2006).

Giudici, P., ”Applied data mining statistical methods for business and industry”, ***John Wiley Press***, England, 3-8 (2003).

Goodman, D., Boggess, L. And Watkins, A., “Artificial immune system classification of multiple-class problems”, ***Intelligent Engineering Systems Through Artificial Neural Networks***, 12: 179-184 (2002).

Han, J. And Kamber, M., “Data mining concepts and techniques”, ***Morgan Kaufmann Publishers***, Tokyo, 30-33 (2001).

Hand, D.J., “Data mining: statistics and more?”, ***The American Statistician***, 52: 112-118 (1998).

Hassan, Y. And Tazaki, E., “Rule extraction based on rough set theory combined with genetic programming and its application to medical data analysis”, ***Seventh Australian and New Zeland Intelligent Information Systems Conference***, Australia, 18-21 (2001).

Hassan, Y., Tazaki, E., Egava, S. And Suyama, K.,” Rough neural classifier system”, ***Systems, Man and Cybernetics***, 5: 1-6 (2002).

Hassan, Y. And Tazaki, E., “Induction of knowledge using evolutionary rough set theory”, ***Cybernetics and Systems: An International Journal***, 34(8): 617-643 (2003).

Hassanien, A. “Rough Set Approach for Attribute Reduction and Rule Generation: A Case of Patients With Suspected Breast Cancer”, ***Journal of the American Society for Information Science and Technology***, 55(11): 954-962 (2004).

Holland, J., “Adaptation in natural and artificial systems”, ***The University of Michigan Press***, Ann Arbour, 3-11 (1975).

Hsieh, N. C., “Hybrid mining approach in the design of credit scoring models”, ***Expert Systems with Applications***, 28(4): 655-665 (2005).

Hua, D. And Yuan, L., “Heuristic genetic algorithm for minimal reduct in decision system based on rough set theory”, ***IEEE***, China, 833-836 (2002).

Huan, L. And Yu, L., ”The Handbook of Data Mining 2nd ed.”, ***Lawrence Erlbaum Associates***, United States of America, 410-422 (2003).

Huang, C.L., Li, T.S. And Peng, T.K., ”A hybrid approach of rough set theory and genetic algorithm for fault diagnosis”, ***Int.J.Adv. Manuf.Technol***, 27: 119-127 (2005).

İnternet : SAS web site “Data Mining” http://www.sas.com (2007).

İnternet : School of Computer Science Web Site “Introduction to Data Mining” http://www.cs.bham.as.uk (2007).

İnternet : Applied Technology Group “Mining for a competitive advantage in your data warehous” www. Techguide.com (1997).

İnternet : Machine Learning Repository “UCI-University of California at Irvine” http://www.ics.uci.edu/~mlearn/MLRepository.html (2007).

Jacobs, P., “Data mining: What general managers need to know”, ***Harvard Management Update***, 4(10): 8 (1999).

Jaroslaw, S. And Katarzyna, K., “Hybrid classifier based on rough sets and neural networks”, ***Electronic Notes in Theoretical Computer Science***, 82(4): 1-11 (2003).

Jensen, R. And Shen, Q., “A rough set-aided system for Sorting www Bookmarks**”**, ***Lecture Notes In Computer Science***, 2198: 95–105 (2001).

John, G.H., Kohavi, R. And Pfleger, K., “Irrelevant features and the subset selection problem”, ***Proceedings of the 11th International Conference on Machine Learning***, San Francisco, 121–129 (1994).

John, G.H. And Langley, P., “Estimating continuous distributions in bayesian classifiers”, ***11th Conf. Uncertainty in Artificial Intelligence***, San Mateo, 338–345 (1995).

Joseph, L. And Breault, “Data mining diabetic databases: are rough sets a useful addition” In ***Proc. 33rd Symposium on the Interface, Computing Science and Statistics***, Fairfax, 1-10 (2001).

Katzberg, J.D. And Ziarko, W., “Variable precision extension of rough sets”, ***Fundamental Informaticae***, 27: 155-168 (1996).

Kira, K. And Rendell, L.A., “The feature selection problem: Traditional methods and a new algorithm”, ***In Proceedings of Ninth National Conference on Artificial Intelligence***, Cambridge, 129–134 (1992).

Kirkpatrick, S., Gelatt, C. And Vecchi, M., “Optimization by simulated annealing”, ***Science***, 220(4598): 671–680 (1983).

Kittler, R. And Wang, W., “Data mining in brief”, ***Solid State Technology***, 42(11): 48 (1999).

Kittler, R. And Wang, W., "The emerging role for data mining", ***Solid State Technology***, 42(11): 45 (1999).

Koza, J.R., "Genetic programming", ***MIT Press***, Cambridge, 8-25 (1992).

Kusiak, A., Kern, J.A., Kernstine, K.H. And Tseng T.L., "Autonomous decision-making: a data mining approach", ***IEEE Transactions on Information Technology in Biomedicine***, 4(4): 274-284 (2000).

Kusiak, A., Dixon, B. And Shah, S., "Predicting survival time for kidney dialysis patients: a data mining approach", ***Computers in Biology and Medicine***, 35(4): 311-327 (2005).

Lavrac, N., "Selected techniques for data mining in medicine", ***Artificial Intelligence in Medicine***, 16: 3–23 (1999).

Liu, H. And Setiono, R., "A probabilistic approach to feature selection - a filter solution", ***In Proceedings of the 9th International Conference on Industrial and Engineering Applications of AI and ES***, USA, 284–292 (1996).

Liu, F. And Lu, S., "An inference approach based on rough sets", ***Lecture Notes in Artificial Intelligence***, 2639: 346-349 (2003).

Luukka, P., "Similarity classifier using measure derived from yu's norms applied to medical data sets", ***IEEE International Conference on Fuzzy Systems***, Canada, 7259-7264 (2006).

Marmelstein, B.E. And Lamont, G.B., "Pattern classification using a hybrid genetic program-decision tree approach", ***Genetic Programming 1998 Proceedings of the Third Annual Conference***, San Francisco, 223-231 (1998).

Mladenic, D., "Feature selection for dimensionality reduction", ***Lecture Notes in Computer Science,*** 3940: 84–102 (2006).

Mugambi, E.M., Hunter, A., Oatley, G. And Kennedy, L., "Polynomial-fuzzy decision tree structures for classifying medical data", ***Knowledge-Based Systems***, 17: 81–87 (2004).

Ngan, P.S., Wong, M.L., Lam, W., Leung, K.S., And Cheng, J.C.Y., "Medical data mining using evolutionary computation" ***Artificial Intelligence in Medicine***, 16: 73-96 (1999).

Oğuz, B., "Eşleştirme haznelemesinin biçimsel kavram analizi ile modellenmesi", Yüksek Lisans Tezi, ***Hacettepe Üniversitesi Fen Bilimleri Enstitüsü***, Ankara, 21-26 (2000).

Oltean, M. And Dumitrescu, D., "Multi Expression Programming", ***Technical Report UBB-01-2002***, ***Babeş-Bolyai University-Romania***, 1-3 (2002).

Oltean, M. And Grosan, C., "Evolving digital circuits using multi expression programming" ***NASA/DoD Conference on Evolvable Hardware, Seattle***, 87-90 (2004).

Özbakır, L., "Çok objektifli esnek atölye çizelgeleme problemlerinin sezgisel yöntemlerle modellenmesi, analizi ve çözümü", Doktora Tezi, ***Erciyes Üniversitesi Fen Bilimleri Enstitüsü***, Kayseri, 135-138 (2004).

Pal, S.K. And Pabitra, M., "Case generation using rough sets with fuzzy representation", ***IEEE Transactions on Knowledge and Data Engineering***, 16(3): 293-300 (2004).

Parpinelli, R.S., Lopes, H.S. And Freitas, A.A., "An ant colony based system for data mining: applications to medical data", ***Proc. Genetic and Evolutionary Computation Conf. (GECCO-2001)***, San Francisco-California, 791-798 (2001).

Parpinelli, R.S., Lopes, H.S. And Freitas, A.A., "Data mining with an ant colony optimization algorithm", ***IEEE Transactions on Evolutionary Computation*** , 6(4): 321-332 (2002).

Pawlak, Z., "Rough Sets", ***International Journal of Computer and Information Sciences***, 11: 341-356 (1982).

Pawlak, Z., "Rough classificaton", ***Int.J.Human-Computer Studies***, 51: 369-383 (1983).

Pawlak, Z., "Rough sets theoretical aspects of reasoning about data". ***Kluwer Academic Publishers***, Boston, 1-30 (1991).

Pawlak, Z. And Slowinski, R., "Rough set approach to multi-attribute decision analysis", ***Europan Journal Of Operational Research***, 72: 443-459 (1994).

Pawlak, Z., "Transactions on rough sets, 1", Skowron A., ***Springer***, Berlin, 1-54 (2004).

Pawlak, Z. And Skowron, A., “Rough membership functions”, Advances in the Dempster-Shafer Theory of Evidence, ***John Wiley & Sons***, New York, 251–271 (1994).

Pawlak, Z., “Rough Sets, decision algorithms and bayes theorem”, ***European Journal of Operational Research***, 136: 181-189 (2000).

Pawlak, Z., “Combining rough sets and bayes’ rule”, ***Computational Intelligence***, 17(3): 401–408 (2001).

Pawlak, Z., “Rough Sets”, ***Kluwer Academic Publishers***, London, 9-16 (1991).

Pendharkar, P.C., Rodger, J.A., Yaverbaum, G.J., Herman, N. And Benner, M., “Association, statistical, mathematical and neural approaches for mining breast cancer patterns”, ***Expert Systems with Applications***, 17: 223–232 (1999).

Pendharkar, P.C., “An empirical study of design and testing of hybrid evolutionary–neural approach for classification”, ***Omega***, 29(4): 361-374 (2001).

Quinlan, J. R., “Induction of decision trees”, ***Machine Learning***, 1: 81-106 (1986).

Raman, B. And Loerger, T.R., “Instance-based filter for feature selection”, ***Journal of Machine Learning Research***, 1: 1–23 (2002).

Renpu, L.R. And Wang, Z., “Mining classification rules using rough sets and neural networks”, ***Computing, Artificial Intelligence and Information Technology***, 157: 439-448 (2003).

Sang, J.L. And Keng, S., “A review of data mining techniques”, ***Industrial Management and Data Systems***, 101(1): 41-46 (2001).

Santos, R., Nievola, J.C. And Freitas, A.A., “Extracting comprehensible rules from neural networks via genetic algorithms” ***IEEE Symp. on Combinations of Evolutionary Computation and Neural Networks (ECNN-2000)***, USA, 130-139 (2000).

Skowron, A. And Rauszer, C., “The discernibility matrices and functions in information system”, ***ICS Research Report***, 1: 1-44 (1991).

Stepaniuk, J., “Hybrid classifier based on rough sets and neural networks”, ***Electronic Notes in Theoretical Computer Science***, 82(4): 235-245 (2003).

Sumathi, S., Sivanandam, S.N. And Balachandar, “Design and development of self-organised neural network schemes as a data mining tool”, ***Engineering Intelligent Systems for Electrical Engineering and Communications***, 9(2): 109-125 (2001).

Takac, A., “Genetic programming in data mining: cellular approach” MSc Thesis, ***Institute of Informatics Faculty of Mathematics, Physics and Informatics Comenius University***, Slovakia, 20-35 (2003).

Tan, K.C., Yu, Q. And Lee, T.H., “A distributed evolutionary classifier for knowledge discovery in data mining”, ***IEEE Transactions on Systems***, 35(2): 131-142 (2005).

Tan, A.C. And Gilbert, D., “An empirical comparison of supervised machine learning techniques in bioinformatics”, ***Proceedings of the First Asia Pacific Bioinformatics Conference,*** Australia, 219-222 (2003).

Tan, K.C., Tay, A., Lee, T.H. And Heng, C.M., “Mining multiple comprehensible classification rules using genetic programming” ***IEEE Congress on Evolutionary Computation***, Honolulu, 1302–1307 (2002).

Tan, K.C., Khor, E.F., Cai, J., Heng, C.M. And Lee, T.H., “Automating the drug scheduling of cancer chemotherapy via evolutionary computation”, ***Artificial Intelligence in Medicine***, 25: 169–185 (2002).

Torgerson, W.S., “Multidimensional scaling: Theory and method”, ***Psychometrika***, 17: 401–419 (1952).

Traina, C., Traina, A., Wu, L. And Faloutsos, C., “Fast feature selection using the fractal dimension”, ***In Proceedings of the 15th Brazilian Symposium on Databases (SBBD)***, Brazil, 158–171 (2000).

Vinterbo, S. And Øhrn, A., “Minimal approximate hitting sets and rule templates”**.** ***International Journal of Approximate Reasoning***, 25(2): 123-143 (2000).

Wang, J., “Encyclopedia of Data Warehousing and Mining”, ***Idea Group Inc.***, USA, 108 (2006).

Wang, F., “On acquiring classification knowledge from noisy data based on rough set”, ***Expert Systems with Applications***, 29(1): 49-64 (2005).

Whitley, D. And Kauth, J., ”A different genetic algorithm”, ***Proc. Rocky Mountain Conf. Artificial Intelligence***, Denver, 118-130 (1988).

Witten, H. And Frank, E., “Data mining: practical machine learning tools and techniques with java implementations”, ***CA: Morgan Kaufmann***, San Mateo, 15-46 (1999).

Yahia, M.E. And Ibrahim, B.A., ”K-nearest neighbour and C4.5 algorithms as data mining methods: advantages and difficulties”, ***ACS/IEEE International Conference on Computer Systems and Applications***, Tunisia, 103 (2003).

Yao, J. “Feature selection for fluorescence image classification”, ***KDD Lab Proposal***, 1: 610-620 (2001).

Yarımağan, Ü., “Veri Tabanı Sistemleri, 2. baskı”, ***Akademi&Türkiye Bilişim Vakfı***, Ankara, 7-9 (2000).

Yuarn, J. G., “Rough set-based approach to data mining”, Ph.D. Thesis, ***Department of Electrical Engineering and Computer Science Case Wester University***, USA, 1-22 (2003).

Zhipeng, X., Wynne, H. And Mong, L.L., “Generalization of classification rules” ***15th IEEE International Conference on Tools with Artificial Intelligence***, California, 522-529 (2003).

Zhong, X., Zhang, L. And Zhang, Y., “A classification rule mining method using hybrid genetic algorithms”, ***IEEE Region Conference***, 2: 207-210 (2004).

Zhong, N. And Skowron, A., “A rough set-based knowledge discovery process”, ***Int. J. Appl. Math. Comput. Sci.***, 11: 429-457 (2001).

Zhong, N., Dong, J. And Ohsuga, S., “Menengitis data mining by cooperatively using GDT-RS and RSBR”, ***Pattern Recognition Letters***, 24: 887-894 (2003).

Zhou, C., Xiao, W., Tirpak, T.M. And Nelson, P.C., “Evolving accurate and compact classification rules with gene expression programming” ***IEEE Transactions on Evolutionary Computation***, 7: 519-531 (2003).

EKLER

EK-1 Wisconsin Breast Cancer (WBC) veri kümesinin özellikleri

Çizelge 1.1. WBC veri kümesindeki nitelikler ve değişkenler

Değişkenler	Wisconsin Breast Cancer Nitelikleri
X0	Clump Thickness: 1 - 10
X1	Uniformity of Cell Size: 1 - 10
X2	Uniformity of Cell Shape: 1 - 10
X3	Marginal Adhesion: 1- 10
X4	Single Epithelial Cell Size: 1 - 10
X5	Bare Nuclei: 1 - 10
X6	Bland Chromatin: 1 - 10
X7	Normal Nucleoli: 1 - 10
X8	Mitoses: 1 - 10

EK-2 Ljubljana Breast Cancer veri kümesinin özellikleri

Göğüs kanseri veri kümesi Yugoslavya'daki Ljubljana onkoloji enstitüsünden elde edilmiştir. M. Zwitter ve M. Soklic bu veriyi sağlamışlardır.

1. Title: Breast cancer data (Michalski has used this)

2. Sources:

-- Matjaz Zwitter & Milan Soklic (physicians)
Institute of Oncology
University Medical Center
Ljubljana, Yugoslavia

--Donors: Ming Tan and Jeff Schlimmer (Jeffrey.Schlimmer@a.gp.cs.cmu.edu)

-- Date: 11 July 1988

3. Past Usage: (Several: here are some)

-- Michalski,R.S., Mozetic,I., Hong,J., & Lavrac,N. (1986).
The Multi-Purpose Incremental Learning System AQ15 and its Testing Application to Three Medical Domains.
In Proceedings of the Fifth National Conference on Artificial Intelligence, 1041-1045, Philadelphia, PA: Morgan Kaufmann.

-- accuracy range: 66%-72%

-- Clark,P. & Niblett,T. (1987). Induction in Noisy Domains. In Progress in Machine Learning (from the Proceedings of the 2nd European Working Session on Learning), 11-30, Bled, Yugoslavia: Sigma Press.

EK-2 (Devam) Ljubljana Breast Cancer veri kümesinin özellikleri

-- 8 test results given: 65%-72% accuracy range

-- Tan, M., & Eshelman, L. (1988). Using weighted networks to represent classification knowledge in noisy domains.
Proceedings of the Fifth International Conference on Machine Learning, 121-134, Ann Arbor, MI.

-- 4 systems tested: accuracy range was 68%-73.5%

-- Cestnik,G., Konenenko,I, & Bratko,I. (1987). Assistant-86: A Knowledge-Elicitation Tool for Sophisticated Users. In I.Bratko & N.Lavrac (Eds.) Progress in Machine Learning, 31-45, Sigma Press.

-- Assistant-86: 78% accuracy

4. Relevant Information:
This is one of three domains provided by the Oncology Institute that has repeatedly appeared in the machine learning literature.
(See also lymphography and primary-tumor.)

This data set includes 201 instances of one class and 85 instances of another class. The instances are described by 9 attributes, some of which are linear and some are nominal.

5. Number of Instances: 286

6. Number of Attributes: 9 + the class attribute

EK-2 (Devam) Ljubljana Breast Cancer (LBC) veri kümesinin özellikleri

Çizelge 2.1. LBC veri kümesindeki nitelikler ve değişkenler

Variables	Ljubljana Breast Cancer
X0	Age: 10-19, 20-29, 30-39, 40-49, 50-59, 60-69, 70-79, 80-89, 90-99.
X1	Menopause: Lt40, Ge40, Premeno (1,2,3).
X2	Tumor-Size: 0-4, 5-9, 10-14, 15-19, 20-24, 25-29, 30-34, 35-39, 40-44, 45-49, 50-54, 55-59.
X3	İnv-Nodes: 0-2, 3-5, 6-8, 9-11, 12-14, 15-17, 18-20, 21-23, 24-26, 27-29, 30-32, 33-35, 36-39.
X4	Node-Caps: Yes, No (1,0)
X5	Deg-Malig: 1, 2, 3.
X6	Breast: Left, Right (0,1).
X7	Breast-Quad: Left-Up, Left-Low, Right-Up, Right-Low, Central (1,2,3,4,5).
X8	İrradiat: Yes, No (1,0).

7. Missing Attribute Values: (denoted by "?")

Attribute #: Number of instances with missing values:
6. 8
9. 1

8. Class Distribution:

1. no-recurrence-events: 201 instances
2. recurrence-events: 85 instances

EK-3 Tic-Tac-Toe veri kümesinin özellikleri

1. Title: Tic-Tac-Toe Endgame database

2. Source Information
-- Creator: David W. Aha (aha@cs.jhu.edu)
-- Donor: David W. Aha (aha@cs.jhu.edu)
-- Date: 19 August 1991

3. Known Past Usage:
1. Matheus,~C.~J., \& Rendell,~L.~A. (1989). Constructive induction on decision trees. In {\it Proceedings of the Eleventh International Joint Conference on Artificial Intelligence}
(pp. 645--650). Detroit, MI: Morgan Kaufmann.

-- CITRE was applied to 100-instance training and 200-instance test sets. In a study using various amounts of domain-specific knowledge, its highest average accuracy was 76.7% (using the final decision tree created for testing).

2. Matheus,~C.~J. (1990). Adding domain knowledge to SBL through feature construction. In {\it Proceedings of the Eighth National Conference on Artificial Intelligence}
(pp. 803--808). Boston, MA: AAAI Press.

-- Similar experiments with CITRE, includes learning curves up to 500-instance training sets but used _all_ instances in the database for testing. Accuracies reached above 90%, but specific values are not given (see Chris's dissertation for more details).

3. Aha,~D.~W. (1991). Incremental constructive induction: An instance based approach. In {\it Proceedings of the Eighth International Workshop on Machine Learning} (pp. 117--121). Evanston, ILL: Morgan Kaufmann.

EK-3 (Devam) Tic-Tac-Toe veri kümesinin özellikleri

-- Used 70% for training, 30% of the instances for testing, evaluated over 10 trials. Results reported for six algorithms:

-- NewID: 84.0%

-- CN2: 98.1%

-- MBRtalk: 88.4%

-- IB1: 98.1%

-- IB3: 82.0%

-- IB3-CI: 99.1%

-- Results also reported when adding an additional 10 irrelevant ternary-valued attributes; similar _relative_ results except that IB1's performance degraded more quickly than the others.

4. Relevant Information:

This database encodes the complete set of possible board configurations at the end of tic-tac-toe games, where "x" is assumed to have played first. The target concept is "win for x" (i.e., true when "x" has one of 8 possible ways to create a "three-in-a-row").

Interestingly, this raw database gives a stripped-down decision tree algorithm (e.g., ID3) fits. However, the rule-based CN2 algorithm, the simple IB1 instance-based learning algorithm, and the CITRE feature-constructing decision tree algorithm perform well on it.

5. Number of Instances: 958 (legal tic-tac-toe endgame boards)

6. Number of Attributes: 9, each corresponding to one tic-tac-toe square

EK-3 (Devam) Tic-Tac-Toe veri kümesinin özellikleri

7. Attribute Information: (x=player x has taken, o=player o has taken, b=blank)

Çizelge 3.1. Tic-Tac-Toe veri kümesindeki nitelikler ve değişkenler

Variables	*Tic-Tac-Toe*
X0	Top-Left-Square: {X,O,B} (0,1,2)
X1	Top-Middle-Square: {X,O,B}
X2	Top-Right-Square: {X,O,B}
X3	Middle-Left-Square: {X,O,B}
X4	Middle-Middle-Square: {X,O,B}
X5	Middle-Right-Square: {X,O,B}
X6	Bottom-Left-Square: {X,O,B}
X7	Bottom-Middle-Square: {X,O,B}
X8	Bottom-Right-Square: {X,O,B}

8. Missing Attribute Values: None

9. Class Distribution: About 65.3% are positive (i.e., wins for "x")

EK-4 CRX veri kümesinin özellikleri

1. Title: Credit Approval

2. Sources:
(confidential)
Submitted by quinlan@cs.su.oz.au

3. Past Usage:

See Quinlan,
* "Simplifying decision trees", Int J Man-Machine Studies 27, Dec 1987, pp. 221-234.
* "C4.5: Programs for Machine Learning", Morgan Kaufmann, Oct 1992

4. Relevant Information:

This file concerns credit card applications. All attribute names and values have been changed to meaningless symbols to protect confidentiality of the data.

This dataset is interesting because there is a good mix of attributes -- continuous, nominal with small numbers of values, and nominal with larger numbers of values. There are also a few missing values.

5. Number of Instances: 690

6. Number of Attributes: 15 + class attribute

EK-4 (Devam) CRX veri kümesinin özellikleri

Çizelge 4.1. CRX veri kümesindeki nitelikler ve değişkenler

Variables	*CRX*
X0	A1: b, a-1,2
X1	A2: (-,38.96):1, (38.96,-):2
X2	A3: (-,4.20):1,(4.20,-):2
X3	A4:u, y, l, t – 1,2,3,4
X4	A5:g, p, gg-1,2,3
X5	A6:c, d, cc, i, j, k, m, r, q, w, x, e, aa, ff-1-14
X6	A7:v, h, bb, j, n, z, dd, ff, o-1-9
X7	A8: (-,1.27):1, (1.27,-):2
X8	A9: t, f-1,2
X9	A10: t, f-1,2
X10	A11: (-,0.5):1,(0.5,2.5):2,(2.5,-):3
X11	A12: t, f-1,2
X12	A13: g, p, s-1,2,3
X13	A14: (-,105):1,(105,289):2,(289,-):3
X14	A15: (-,492):1, (492,-):2

EK-4 (Devam) CRX veri kümesinin özellikleri

7. Missing Attribute Values:

37 cases (5%) have one or more missing values. The missing values from particular attributes are:

A1: 12
A2: 12
A4: 6
A5: 6
A6: 9
A7: 9
A14: 13

8. Class Distribution

+: 307 (44.5%)
-: 383 (55.5%)

EK-5 Nursery veri kümesinin özellikleri

1. Title: Nursery Database

2. Sources:
   (a) Creator: Vladislav Rajkovic et al. (13 experts)
   (b) Donors: Marko Bohanec (marko.bohanec@ijs.si)
   Blaz Zupan (blaz.zupan@ijs.si)
   (c) Date: June, 1997

3. Past Usage:

The hierarchical decision model, from which this dataset is derived, was first presented in M. Olave, V. Rajkovic, M. Bohanec: An application for admission in public school systems. In (I. Th. M. Snellen and W. B. H. J. van de Donk and J.-P. Baquiast, editors) Expert Systems in Public Administration, pages 145-160. Elsevier Science Publishers (North Holland)}, 1989.

Within machine-learning, this dataset was used for the evaluation of HINT (Hierarchy INduction Tool), which was proved to be able to completely reconstruct the original hierarchical model. This, together with a comparison with C4.5, is presented in B. Zupan, M. Bohanec, I. Bratko, J. Demsar: Machine learning by function decomposition. ICML-97, Nashville, TN. 1997 (to appear)

4. Relevant Information Paragraph:

Nursery Database was derived from a hierarchical decision model originally developed to rank applications for nursery schools. It was used during several years in 1980's when there was excessive enrollment to these

EK-5 (Devam) Nursery veri kümesinin özellikleri

schools in Ljubljana, Slovenia, and the rejected applications frequently needed an objective explanation. The final decision depended on three subproblems: occupation of parents and child's nursery, family structure and financial standing, and social and health picture of the family. The model was developed within expert system shell for decision making DEX (M. Bohanec, V. Rajkovic: Expert system for decision making. Sistemica 1(1), pp. 145-157, 1990).

The hierarchical model ranks nursery-school applications according to the following concept structure:

NURSERY	Evaluation of applications for nursery schools
EMPLOY	Employment of parents and child's nursery
parents	Parents' occupation
has_nurs	Child's nursery
STRUCT_FINAN	Family structure and financial standings
STRUCTURE	Family structure
form	Form of the family
children	Number of children
housing	Housing conditions
finance	Financial standing of the family
SOC_HEALTH	Social and health picture of the family
social	Social conditions
health	Health conditions

Input attributes are printed in lowercase. Besides the target concept (NURSERY) the model includes four intermediate concepts:
EMPLOY, STRUCT_FINAN, STRUCTURE, SOC_HEALTH. Every concept is in the original model related to its lower level descendants by a set of

EK-5 (Devam) Nursery veri kümesinin özellikleri

examples (for these examples sets see http://www-ai.ijs.si/BlazZupan/nursery.html).

The Nursery Database contains examples with the structural information removed, i.e., directly relates NURSERY to the eight input attributes: parents, has_nurs, form, children, housing, finance, social, health.

Because of known underlying concept structure, this database may be particularly useful for testing constructive induction and structure discovery methods.

5. Number of Instances: 12960
(instances completely cover the attribute space)

6. Number of Attributes: 8

7. Missing Attribute Values: none

8. Class Distribution (number of instances per class)

class	N	N[%]
not_recom	4320	(33.333 %)
recommend	2	(0.015 %)
very_recom	328	(2.531 %)
priority	4266	(32.917 %)
spec_prior	4044	(31.204 %)

EK-5 (Devam) Nursery veri kümesinin özellikleri

Çizelge 5.1. CRX veri kümesindeki nitelikler ve değişkenler

Variables	*Nursery*
X0	Parents: usual, pretentious, great_pret -1,2,3 has_nurs: proper,less_proper, improper, critical,
X1	v.c.-1,2,3,4,5
X2	Form: complete, completed, incomplete,
X3	foster=1,2,3,4
X4	children: 1, 2, 3, more -1,2,3,4
X5	Housing: convenient, less_conv, critical -1,2,3
X6	Finance: convenient, inconv-1,2
X7	Social: non-prob, slightly_prob, problematic -1,2,3
X8	Health: recommended, priority, not_recom - 1,2,3

EK-6 Cleveland veri kümesinin özellikleri

1. Title: Heart Disease Databases

2. Source Information:
(a) Creators:
-- 1. Hungarian Institute of Cardiology. Budapest: Andras Janosi, M.D.
-- 2. University Hospital, Zurich, Switzerland: William Steinbrunn, M.D.
-- 3. University Hospital, Basel, Switzerland: Matthias Pfisterer, M.D.
-- 4. V.A. Medical Center, Long Beach and Cleveland Clinic Foundation: Robert Detrano, M.D., Ph.D.
(b) Donor: David W. Aha (aha@ics.uci.edu) (714) 856-8779
(c) Date: July, 1988

3. Past Usage:
1. Detrano,~R., Janosi,~A., Steinbrunn,~W., Pfisterer,~M., Schmid,~J., Sandhu,~S., Guppy,~K., Lee,~S., \& Froelicher,~V. (1989). {\it International application of a new probability algorithm for the diagnosis of coronary artery disease.} {\it American Journal of Cardiology}, {\it 64},304--310.
-- International Probability Analysis
-- Address: Robert Detrano, M.D.
Cardiology 111-C
V.A. Medical Center
5901 E. 7th Street
Long Beach, CA 90028
-- Results in percent accuracy: (for 0.5 probability threshold)
Data Name: CDF CADENZA
-- Hungarian 77 74
Long beach 79 77
Swiss 81 81
-- Approximately a 77% correct classification accuracy with a logistic-regression-derived discriminant function

EK-6 (Devam) Cleveland veri kümesinin özellikleri

2. David W. Aha & Dennis Kibler

-- Instance-based prediction of heart-disease presence with the Cleveland database

-- NTgrowth: 77.0% accuracy

-- C4: 74.8% accuracy

3. John Gennari

-- Gennari, J.~H., Langley, P, \& Fisher, D. (1989). Models of incremental concept formation. {\it Artificial Intelligence, 40}, 11--61.

-- Results:

-- The CLASSIT conceptual clustering system achieved a 78.9% accuracy on the Cleveland database.

4. Relevant Information:

This database contains 76 attributes, but all published experiments refer to using a subset of 14 of them. In particular, the Cleveland database is the only one that has been used by ML researchers to this date. The "goal" field refers to the presence of heart disease in the patient. It is integer valued from 0 (no presence) to 4.

Experiments with the Cleveland database have concentrated on simply attempting to distinguish presence (values 1,2,3,4) from absence (value 0).

The names and social security numbers of the patients were recently removed from the database, replaced with dummy values.

One file has been "processed", that one containing the Cleveland database. All four unprocessed files also exist in this directory.

EK-6 (Devam) Cleveland veri kümesinin özellikleri

5. Number of Instances:

Database:	# of instances:
Cleveland:	303
Hungarian:	294
Switzerland:	123
Long Beach VA:	200

6. Number of Attributes: 76 (including the predicted attribute)

Çizelge 6.1. Cleveland veri kümesindeki nitelikler ve değişkenler

Variables	Cleveland Heart Disease
X0	Age: 20,30,40,50,60,70,80
X1	Sex: Sex (1 = Male; 0 = Female)
X2	Cp: Chest Pain Type1:Typical Angina2:Atypical Angina3:Non-Anginal Pain4: Asymptomatic
X3	Trestbps: Resting Blood Pressure (90-200)
X4	Chol: Serum Cholestoral İn Mg/Dl (120-570)
X5	Fbs: (Fasting Blood Sugar > 120 Mg/Dl) (1 = True; 0 = False)
X6	Restecg: 0: Normal 1: Having St-T Wave 2: Left Ventricular Hypertrophy
X7	Thalach: Maximum Heart Rate Achieved (70-202)
X8	Exang: Exercise İnduced Angina (1 = Yes; 0 = No)
X9	Oldpeak = St Depression İnduced By Exercise Relative To Rest (0.6-6.4)
X10	Slope: 1: Upsloping 2: Flat 3: Downsloping
X11	Ca: Number Of Major Vessels (0-3) Colored By Flourosopy
X12	Thal: 3 = Normal; 6 = Fixed Defect; 7 = Reversable Defect

7. Missing Attribute Values: Several. Distinguished with value -9.0.

EK-6 (Devam) Cleveland veri kümesinin özellikleri

8. Class Distribution:

Database:	0	1	2	3	4	Total
Cleveland:	164	55	36	35	13	303
Hungarian:	188	37	26	28	15	294
Switzerland:	8	48	32	30	5	123
Long Beach:	51	56	41	42	10	200

EK-7 Lymphography veri kümesinin özellikleri

1. Title: Lymphography Domain

2. Sources:

(a) See Above.

(b) Donors: Igor Kononenko,

University E.Kardelj
Faculty for electrical engineering
Trzaska 25
61000 Ljubljana (tel.: (38)(+61) 265-161

Bojan Cestnik
Jozef Stefan Institute
Jamova 39
61000 Ljubljana
Yugoslavia (tel.: (38)(+61) 214-399 ext.287)

(c) Date: November 1988

3. Past Usage: (sveral)

1. Cestnik,G., Konenenko,I, & Bratko,I. (1987). Assistant-86: A Knowledge-Elicitation Tool for Sophisticated Users. In I.Bratko& N.Lavrac (Eds.) Progress in Machine Learning, 31-45, Sigma Press.

-- Assistant-86: 76% accuracy

2. Clark,P. & Niblett,T. (1987). Induction in Noisy Domains. In I.Bratko & N.Lavrac (Eds.) Progress in Machine Learning, 11-30,

Sigma Press.

-- Simple Bayes: 83% accuracy

-- CN2 (99% threshold): 82%

3. Michalski,R., Mozetic,I. Hong,J., & Lavrac,N. (1986). The Multi-Purpose Incremental Learning System AQ15 and its Testing Applications to Three

EK-7 (Devam) Lymphography veri kümesinin özellikleri

Medical Domains. In Proceedings of the Fifth National Conference on Artificial Intelligence, 1041-1045. Philadelphia, PA: Morgan Kaufmann.

-- Experts: 85% accuracy (estimate)

-- AQ15: 80-82%

4. Relevant Information:

This is one of three domains provided by the Oncology Institut that has repeatedly appeared in the machine learning literature.

(See also breast-cancer and primary-tumor.)

5. Number of Instances: 148

6. Number of Attributes: 19 including the class attribute

7. Attribute information:

--- NOTE: All attribute values in the database have been entered as numeric values corresponding to their index in the list of attribute values for that attribute domain as given below.

1. class: normal find, metastases, malign lymph, fibrosis
2. lymphatics: normal, arched, deformed, displaced
3. block of affere: no, yes
4. bl. of lymph. c: no, yes
5. bl. of lymph. s: no, yes
6. by pass: no, yes
7. extravasates: no, yes
8. regeneration of: no, yes
9. early uptake in: no, yes
10. lym.nodes dimin: 0-3
11. lym.nodes enlar: 1-4
12. changes in lym.: bean, oval, round

EK-7 (Devam) Lymphography veri kümesinin özellikleri

13. defect in node: no, lacunar, lac. marginal, lac. central
14. changes in node: no, lacunar, lac. margin, lac. central
15. changes in stru: no, grainy, drop-like, coarse, diluted, reticular,stripped, faint,
16. special forms: no, chalices, vesicles
17. dislocation of: no, yes
18. exclusion of no: no, yes
19. no. of nodes in: 0-9, 10-19, 20-29, 30-39, 40-49, 50-59, 60-69, >=70

8. Missing Attribute Values: None

9. Class Distribution:

Class:	Number of Instances:
normal find:	2
metastases:	81
malign lymph:	61
fibrosis:	4

EK-8 Iris veri kümesinin özellikleri

1. Title: Iris Plants Database
Updated Sept 21 by C.Blake - Added discrepency information

2. Sources:
(a) Creator: R.A. Fisher
(b) Donor: Michael Marshall (MARSHALL%PLU@io.arc.nasa.gov)
(c) Date: July, 1988

3. Past Usage:
- Publications: too many to mention!!! Here are a few.
1. Fisher,R.A. "The use of multiple measurements in taxonomic problems" Annual Eugenics, 7, Part II, 179-188 (1936); also in "Contributions to Mathematical Statistics" (John Wiley, NY, 1950).
2. Duda,R.O., & Hart,P.E. (1973) Pattern Classification and Scene Analysis. (Q327.D83) John Wiley & Sons. ISBN 0-471-22361-1. See page 218.
3. Dasarathy, B.V. (1980) "Nosing Around the Neighborhood: A New System Structure and Classification Rule for Recognition in Partially Exposed Environments". IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-2, No. 1, 67-71.
-- Results:
-- very low misclassification rates (0% for the setosa class)
4. Gates, G.W. (1972) "The Reduced Nearest Neighbor Rule". IEEE Transactions on Information Theory, May 1972, 431-433.
-- Results:
-- very low misclassification rates again
5. See also: 1988 MLC Proceedings, 54-64. Cheeseman et al's AUTOCLASS II
conceptual clustering system finds 3 classes in the data.

EK-8 (Devam) Iris veri kümesinin özellikleri

4. Relevant Information:

--- This is perhaps the best known database to be found in the pattern recognition literature. Fisher's paper is a classic in the field and is referenced frequently to this day. (See Duda & Hart, for example.) The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant. One class is inearly separable from the other 2; the latter are NOT linearly separable from each other.

--- Predicted attribute: class of iris plant.

--- This is an exceedingly simple domain.

--- This data differs from the data presented in Fishers article

(identified by Steve Chadwick, spchadwick@espeedaz.net)

The 35th sample should be: 4.9,3.1,1.5,0.2,"Iris-setosa"

where the error is in the fourth feature.

The 38th sample: 4.9,3.6,1.4,0.1,"Iris-setosa"

where the errors are in the second and third features.

5. Number of Instances: 150 (50 in each of three classes)

6. Number of Attributes: 4 numeric, predictive attributes and the class

7. Attribute Information:

1. sepal length in cm
2. sepal width in cm
3. petal length in cm
4. petal width in cm
5. class:
   -- Iris Setosa
   -- Iris Versicolour
   -- Iris Virginica

EK-8 (Devam) Iris veri kümesinin özellikleri

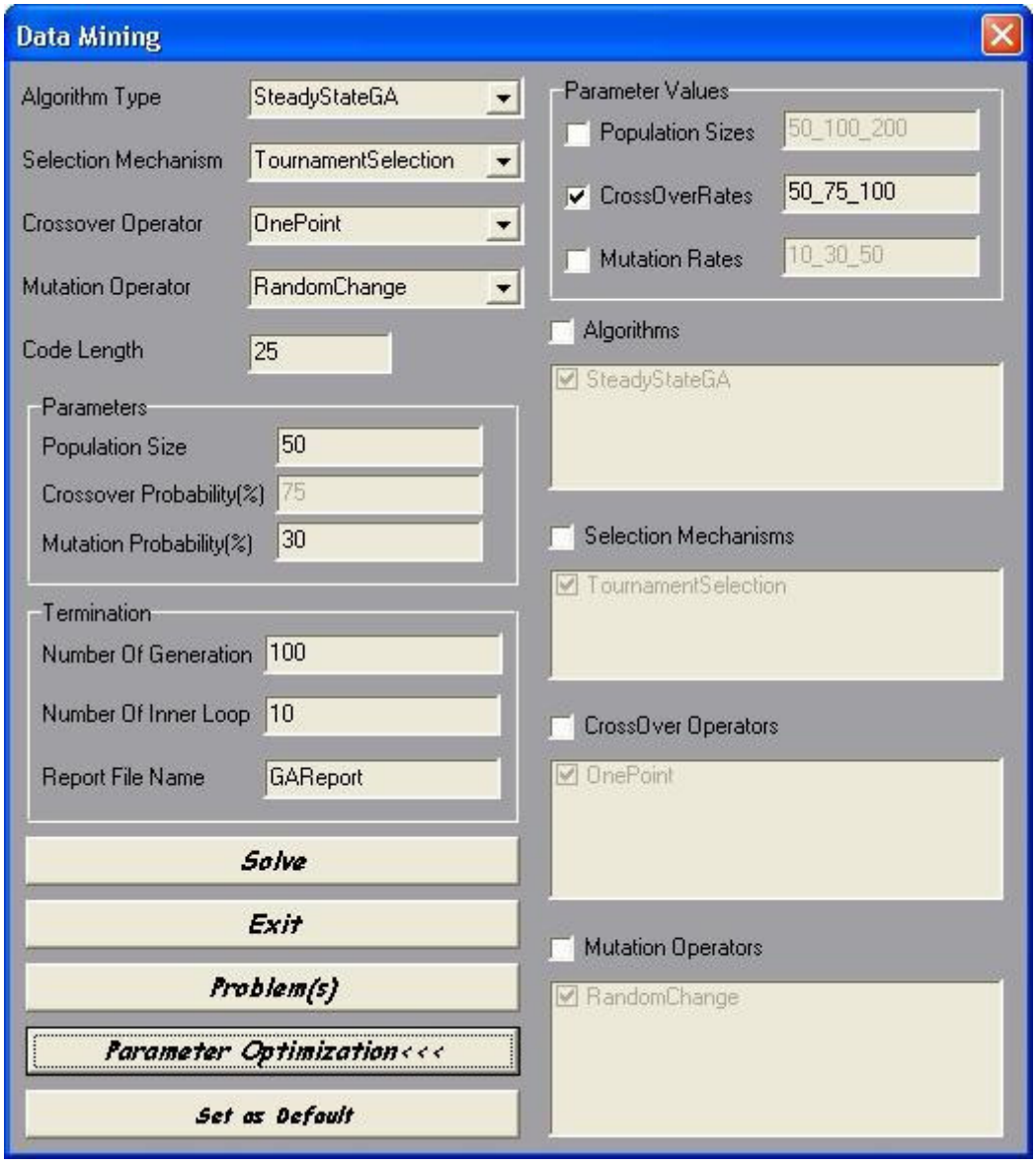
8. Missing Attribute Values: None

Summary Statistics:

	Min	Max	Mean	SD	Class Correlation
sepal length:	4.3	7.9	5.84	0.83	0.7826
sepal width:	2.0	4.4	3.05	0.43	-0.4194
petal length:	1.0	6.9	3.76	1.76	0.9490 (high!)
petal width:	0.1	2.5	1.20	0.76	0.9565 (high!)

9. Class Distribution: 33.3% for each of 3 classes.

EK-9 Rough-Mep algoritmasının C# programındaki görsel görünümü



EK-10 CRX veri kümesinin sonuçları

Crx	Kod Uzunluğu	Popülasyon Büyüklüğü	Çaprazlama Oranı	Mutasyon Oranı	Jenerasyon Sayısı	Sonuç	Süre(sn)
training1.txt	25	50	1	0,3	100	0,956521739	3
training10.txt	25	50	1	0,3	100	0,956521739	3.125
training2.txt	25	50	1	0,3	100	0,985074627	2.939
training3.txt	25	50	1	0,3	100	0,985507246	3.063
training4.txt	25	50	1	0,3	100	0,955882353	2.876
training5.txt	25	50	1	0,3	100	1	0,297
training6.txt	25	50	1	0,3	100	1	1.297
training7.txt	25	50	1	0,3	100	0,971014493	3.078
training8.txt	25	50	1	0,3	100	0,985507246	2.953
training9.txt	25	50	1	0,3	100	0,985507246	3.078
						0,978153669	
					stdsapma	**0,017151318**	

EK-11 Nursery veri kümesinin sonuçları

Nursery	Kod Uzunluğu	Popülasyon Büyüklüğü	Çaprazlama Oranı	Mutasyon Oranı	Jenerasyon Sayısı	Sonuç	Süre(sn)
training1.txt	25	50	1	0,3	100	0,996141975	212.813
training2.txt	25	50	1	0,3	100	0,991512346	214.531
training3.txt	25	50	1	0,3	100	0,986111111	212.829
training4.txt	25	50	1	0,3	100	0,989197531	208.453
training5.txt	25	50	1	0,3	100	1	127.844
training6.txt	25	50	1	0,3	100	1	83.141
training7.txt	25	50	1	0,3	100	1	110.016
training8.txt	25	50	1	0,3	100	0,993055556	208,5
training9.txt	25	50	1	0,3	100	1	130.656
training10.txt	25	50	1	0,3	100	1	44.859
						0,995601852	
					stdsapma	**0,005284226**	

EK-12 Iris veri kümesinin sonuçları

Iris	Kod Uzunluğu	Popülasyon Büyüklüğü	Çaprazlama Oranı	Mutasyon Oranı	Jenerasyon Sayısı	Sonuç	Süre(sn)
training1.txt	25	50	1	0,3	100	1	212.813
training2.txt	25	50	1	0,3	100	1	214.531
training3.txt	25	50	1	0,3	100	1	212.829
training4.txt	25	50	1	0,3	100	1	208.453
training5.txt	25	50	1	0,3	100	1	127.844
training6.txt	25	50	1	0,3	100	1	83.141
training7.txt	25	50	1	0,3	100	1	110.016
training8.txt	25	50	1	0,3	100	0,93333	208,5
training9.txt	25	50	1	0,3	100	1	130.656
training10.txt	25	50	1	0,3	100	1	44.859
						0,97333	
					stdsapma	**0,0544226**	

EK-13 LBC veri kümesinin sonuçları

LBC	Kod Uzunluğu	Popülasyon Büyüklüğü	Çaprazlama Oranı	Mutasyon Oranı	Jenerasyon Sayısı	Sonuç	Süre(sn)
Training1.txt	25	50	1	0,3	100	0,928571429	1,75
Training10.txt	25	50	1	0,3	100	1	0,359
Training2.txt	25	50	1	0,3	100	0,892857143	1.704
Training3.txt	25	50	1	0,3	100	0,892857143	1.828
Training4.txt	25	50	1	0,3	100	0,928571429	1.766
Training5.txt	25	50	1	0,3	100	0,892857143	1.703
Training6.txt	25	50	1	0,3	100	0,892857143	1.718
Training7.txt	25	50	1	0,3	100	0,892857143	1.734
Training8.txt	25	50	1	0,3	100	1	0,172
Training9.txt	25	50	1	0,3	100	0,964285714	1.735
						0,928571429	
					stdsapma	**0,04454354**	

EK-14 Tic-Tac-Toe veri kümesinin sonuçları

Tic-Tac-Toe	Kod Uzunluğu	Popülasyon Büyüklüğü	Çaprazlama Oranı	Mutasyon Oranı	Jenerasyon Sayısı	Sonuç	Süre(sn)
training1.txt	25	50	1	0,3	100	0,947916667	3
training10.txt	25	50	1	0,3	100	0,904255319	3
training2.txt	25	50	1	0,3	100	0,895833333	3,532
training3.txt	25	50	1	0,3	100	0,916666667	3
training4.txt	25	50	1	0,3	100	0,916666667	3,188
training5.txt	25	50	1	0,3	100	0,947916667	4
training6.txt	25	50	1	0,3	100	0,895833333	3
training7.txt	25	50	1	0,3	100	0,947916667	3,329
training8.txt	25	50	1	0,3	100	0,875	3
training9.txt	25	50	1	0,3	100	0,90625	3
						0,915425532	
					stdsapma	**0,025344293**	

EK-15 WBC veri kümesinin sonuçları

WBC	Kod Uzunluğu	Popülasyon Büyüklüğü	Çaprazlama Oranı	Mutasyon Oranı	Jenerasyon Sayısı	Sonuç	Süre(sn)
Training1.txt	25	50	1	0,3	100	0,985294118	3.328
Training10.txt	25	50	1	0,3	100	1	0,359
Training2.txt	25	50	1	0,3	100	0,985294118	3.437
Training3.txt	25	50	1	0,3	100	1	0,656
Training4.txt	25	50	1	0,3	100	1	0,734
Training5.txt	25	50	1	0,3	100	1	0,64
Training6.txt	25	50	1	0,3	100	1	1.063
Training7.txt	25	50	1	0,3	100	1	1.375
Training8.txt	25	50	1	0,3	100	1	0,344
Training9.txt	25	50	1	0,3	100	1	0,375
						0,997058824	
					stdsapma	**0,006200544**	

EK-16 Cleveland veri kümesinin sonuçları

Cleveland	Kod Uzunluğu	Popülasyon Büyüklüğü	Çaprazlama Oranı	Mutasyon Oranı	Jenerasyon Sayısı	Sonuç	Süre(sn)
training1.txt	25	50	1	0,3	100	0,9	6.016
training10.txt	25	50	1	0,3	100	0,878787879	5.671
training2.txt	25	50	1	0,3	100	0,933333333	5.719
training3.txt	25	50	1	0,3	100	0,933333333	5.719
training4.txt	25	50	1	0,3	100	0,933333333	5,89
training5.txt	25	50	1	0,3	100	0,966666667	5.765
training6.txt	25	50	1	0,3	100	0,833333333	5.907
training7.txt	25	50	1	0,3	100	0,933333333	5.969
training8.txt	25	50	1	0,3	100	0,933333333	6.078
train9						**0,916161616**	
					stdsapma	**0,039597377**	

EK-17 Lymphography veri kümesinin sonuçları

Lymphography	Kod Uzunluğu	Popülasyon Büyüklüğü	Çaprazlama Oranı	Mutasyon Oranı	Jenerasyon Sayısı	Sonuç	Süre(sn)
train1.txt	25	50	1	0,3	100	0,933333333	1,876
train10.txt	25	50	1	0,3	100	0,846153846	1,844
train2.txt	25	50	1	0,3	100	1	0,171
train3.txt	25	50	1	0,3	100	0,766666	1,64
train4.txt	25	50	1	0,3	100	0,933333333	1,844
train5.txt	25	50	1	0,3	100	1	0,875
train6.txt	25	50	1	0,3	100	1	2,062
train7.txt	25	50	1	0,3	100	1	1,673
train8.txt	25	50	1	0,3	100	1	0,563
train9.txt	25	50	1	0,3	100	1	0,469
						0,947948651	
					stdsapma	**0,081587618**	

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : KIZILKAYA AYDOĞAN, Emel
Uyruğu : T.C.
Doğum tarihi ve yeri : 26.06.1979 Kayseri
Medeni hali : Evli
Telefon : 0 505 266 08 57
e-mail : ekizilkaya@gazi.edu.tr

Eğitim

Derece	Eğitim Birimi	Mezuniyet tarihi
Doktora	Gazi Üniversitesi / Endüstri Müh.	-
Yüksek lisans	Gazi Üniversitesi / Endüstri Müh.	2003
Lisans	Erciyes Üniversitesi/ Endüstri Müh.	2001
Lise	Melikgazi Süper Lisesi	1997

İş Deneyimi

Yıl	Yer	Görev
2001-2002	Erciyes Üniversitesi	Araştırma Görevlisi
2002-2007	Gazi Üniversitesi	Araştırma Görevlisi

Yabancı Dil

İngilizce

Katıldığı Kurslar

SPSS Clementine Veri Madenciliği Temel Eğitimi
SPSS Clementine İleri Veri Madenciliği Eğitimi
SPSS Clementine Veri Manüplasyon Eğitimi
C# Programlama

Yayınlar

SCI Yayınlar

1. Gökçen, H., Ağpak, K., Gencer, C. and Kızılkaya, E., "A Shortest Route Formulation Of Simple U-Type Assembly Line Balancing Problem", Applied Mathematical Modelling, 29 (4): 373-380, 2005.

2. Gencer, C., Top, İ. and Aydoğan, K.E., "A New Intuitional Algorithm For Solving Heterogeneous Fixed Fleet Routing Problems: Passenger Pickup Algorithm", Applied Mathematics and Computation, 181 (2): 1552-1567, 2006.

3. Gencer, C., Aydoğan, E.K., Çelik, C., "A Decision Support System for Locating VHF/UHF Radio Jammer Systems on the Terrain", Information Systems Frontiers, basımda, 2007.

4. Aydoğan, E.K., Gencer, C., "Mining classification rules with Reduced MEPAR-miner Algorithm", Applied Mathematics and Computation, basımda, 2007.

5. Gencer, C.; Soydemir, A.; Aydoğan, E.K., "Chemical agent detector placement methodology", Applied Mathematics and Computation, basımda, 2007.

6. Aydoğan, E.K.; Gencer, C., "A New Genetic Algorithm Approach for Discovering Classification Rules in Medical Data Mining", Information Sciences, incelemede.

7. Gencer, C., Aydoğan, E.K., Aytürk, S., "Evaluating Weapon System Using Analytic Network Process", IEEE Transactions on Engineering Management, incelemede.

Diğer Yayınlar

1. Aydoğan, E.K, Gencer, C., “Veri Madenciliği Problemlerinde Kaba Küme Yaklaşımı Kullanılarak Sınıflandırma Amaçlı Yapılmış Olan Çalışmalar”, Kara Harp Okulu Savunma Bilimleri Dergisi, basımda, 2007.

2. Aydoğan, E.K., Gencer, C., Carlıoğlu, S., “Ege Bölgesinde Deniz Kuvvetleri Komutanlığına Ait Sahil Gözetleme Radarlarının Yeniden Yerleştirilmesi”, Kara Harp Okulu Savunma Bilimleri Dergisi, basımda, 2007.

3. Aydoğan, E.K.; Gencer, C.; Gökçen, H. ve Ağpak, K., “Basit U Tipi Montaj Hattı Dengeleme Problemi İçin Yeni Bir Optimal Çözüm Yöntemi: En Kısa Yol Modeli”, Yöneylem Araştırması ve Endüstri Mühendisliği XXIII. Ulusal Kongresi, Adana, 16-18 Haziran 2004.

4. Aydoğan, E.K., Gencer, C., Özbakır, L., “Müşteri Segmentasyonu Ve Ayrılan Müşteri Analizi”, 12. Ulusal Ergonomi Kongresi, Ankara, 16-18 Kasım 2006.

5. Aydoğan, E.K., Gencer, C., Özbakır, L., “Irak Savaşı’nda Asker Yerleştirme Politikaları Ve Askeri Kayıpların İncelenmesi”, 12. Ulusal Ergonomi Kongresi, Ankara, 16-18 Kasım 2006.

6. Aydoğan, E.K., “Fırın Üretimi Gerçekleştiren Bir Firmada İş Yükü Dengesizliğinin Giderilmesine Yönelik Bir Çalışma”, 13. Ulusal Ergonomi Kongresi, Kayseri, 6-8 Aralık 2007.

7. Aydoğan, E.K., Gencer, C., “Bir Beyaz Eşya İşletmesi’nde Depo Yerleşiminin Yeniden Düzenlenmesi”, 13. Ulusal Ergonomi Kongresi, Kayseri, 6-8 Aralık 2007.

8. Aydoğan, E.K., Gencer, C., Kocabaş, S., “A Decision Support System For Routing Of Unmanned Aerial Vehicles”, Beykent Üniversitesi Dergisi’nde incelemede.

9. Gencer, C., Aydoğan, E.K., Akbulut, S., “Veri Madenciliği Teknikleri ile Bir Kozmetik Markanın Ayrılan Müşteri Analizi ve Müşteri Segmentasyonu”, SİGMA Dergisinde incelemede.

Hobiler

Sinema, Kitap Okumak, Spor Yapmak