



T.C.
KONYA TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ



ESNEK HESAPLAMA YÖNTEMLERİYLE ÇOK
GİRİŞLİ MEDİKAL SİSTEMLERDE GİRİŞ
NİTELİK SAYILARININ İNDİRGENMESİYLE
SINIFLANDIRMA

Kürşat ZÜHTÜOĞULLARI
DOKTORA TEZİ

Bilgisayar Mühendisliği Anabilim Dalı

Haziran 2022
KONYA
Her Hakkı Saklıdır

TEZ KABUL VE ONAYI

Kürşat Zühtüoğulları tarafından hazırlanan “Esnek Hesaplama Yöntemleriyle Çok Girişli Medikal Sistemlerde Giriş Nitelik Sayılarının İndirgenmesiyle Sınıflandırma” adlı tez çalışması 29/06/2022 tarihinde aşağıdaki jüri tarafından oy çokluğu ile Konya Teknik Üniversitesi Lisansüstü Eğitim Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı’nda DOKTORA TEZİ olarak kabul edilmiştir.

Jüri Üyeleri

İmza

Başkan

Prof. Dr. Harun UĞUZ

.....

Danışman

Prof. Dr. Novruz ALLAHVERDİ

.....

Üye

Prof. Dr. Halife KODAZ

.....

Üye

Dr. Öğr. Üyesi Uğur TAŞKIRAN

.....

Üye

Dr. Öğr. Üyesi Murat KÖKLÜ

.....

Yukarıdaki sonucu onaylarım.

Prof. Dr. Saadettin Erhan KESEN
Enstitü Müdürü

TEZ BİLDİRİMİ

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

DECLARATION PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Kürşat Zühtüoğulları

Tarih: 29 06 2022

ÖZET

DOKTORA TEZİ

ESNEK HESAPLAMA YÖNTEMLERİYLE ÇOK GİRİŞLİ MEDİKAL SİSTEMLERDE GİRİŞ NİTELİK SAYILARININ İNDİRGENMESİYLE SINIFLANDIRMA

Kürşat Zühtüoğulları

**Konya Teknik Üniversitesi
Lisansüstü Eğitim Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı**

Danışman: Prof. Dr. Novruz ALLAHVERDİ

2022, 105 Sayfa

Jüri

Prof. Dr. Novruz ALLAHVERDİ

Prof. Dr. Harun UĞUZ

Prof. Dr. Halife KODAZ

Öğr. Üyesi Dr. Uğur TAŞKIRAN

Öğr. Üyesi Dr. Murat KÖKLÜ

Çok girişli medikal sistemlerde baskın girişlerin tespit edilmesi önemli bir problemdir. Bu tez çalışmasında genetik algoritma ve kaba küme temelli hibrit bir indirgeme sistem yazılımı geliştirilerek çok girişli medikal sistemlerin giriş sayısının azaltılması ve baskın girişlerin tespit edilmesi sağlanmıştır. Modifiye edilmiş ve önerilen genetik seçim mekanizmasıyla sistem performansı artırılmıştır. Sistem hem ürolojik hem de standart veri tabanında denenmiş ve baskın giriş değerleri hesaplanmıştır. Geliştirilen indirgeme yazılımı ve seçim mekanizmalarına eklenen modifikasyonlar sayesinde daha yüksek giriş değerlerine sahip sistemlere yazılım tarafından desteklenmesi sağlanmıştır. Ayrıca genetik algoritma temelli sistemlerde görülen lokal optimuma takılma problemi çözülmüş ve indirgeme yazılımının hafızayı daha verimli kullanması sağlanmıştır.

Anahtar Kelimeler: Genetik Algoritma, Kaba Küme Teorisi, Medikal Sistemlerde Esnek Hesaplama

ABSTRACT

PhD THESIS CLASSIFICATION BY REDUCTION OF THE INPUT ATTRIBUTES OF MEDICAL SYSTEMS WITH SOFT COMPUTING TECHNIQUES.

**Kursat ZUHTUOGULLARI
Konya Technical University
Institute of Graduate Studies
Department of Computer Engineering**

**Advisor: Prof. Dr. Novruz ALLAHVERDI
2022 105 Pages**

**Jury
Prof. Dr. Novruz ALLAHVERDI
Prof. Dr. Harun UGUZ
Prof. Dr. Halife KODAZ
Asst. Prof. Dr. Ugur TASKIRAN
Asst. Prof Dr. Murat KOKLU**

Detecting dominant inputs is an important problem in multi-input medical systems. In this thesis, a hybrid reduction system software based on a genetic algorithm and the rough set is developed to reduce the number of inputs and detect the dominant inputs in multi-input medical systems. The system performance is improved by using the proposed modified genetic selection mechanisms. The system was tested on both urological and standard databases and the dominant input values were calculated. Modifications added to genetic algorithm selection mechanism allow the software to support databases with high number of input variables. Local optimum problem of genetic algorithm based systems has been solved and effective memory usages have been obtained during tests.

Keywords. Genetic Algorithm, Rough Sets Theory, Soft Computing Systems

ÖNSÖZ

Çalışmalarımnda her türlü bilimsel ve manevi desteğini benden esirgemeyen danışmanım Prof. Dr. Novruz ALLAHVERDİ'ye teşekkür ederim.

Kürşat Zühtüoğulları
2022



İÇİNDEKİLER

ÖZET	IV
ABSTRACT.....	V
İÇİNDEKİLER	VII
1. GİRİŞ	1
1.1 Çalışmanın Amacı.....	1
1.2 Çalışmanın Literatüre Katkısı	2
1.3 Veri Madenciliği ve Esnek Hesaplama	2
1.4 Veri Madenciliğine Genel Bakış.....	6
2. KAYNAK ARAŞTIRMASI	13
3. MATERYAL VE YÖNTEM.....	16
3.1 Üroflovetrik ve Rezidüel ölçümler ve Kullanılan Veri Tabanları	16
3.2 İşlenen Standart (UCI) Dermatolojik Veri Tabanı	20
3.3 Kaba Küme Temelli Bilgi (Enformasyon) Sistemleri	26
3.4 Ayırt Edilemezlik (Indiscernibility).....	27
3.5 Ayırt Edilebilirlik.....	27
3.6 Karara Bağlı Ayırt Edilebilirlik Matrisi.....	27
3.7 Çekirdek İndirgenmiş Özellik Kümesi (The Core Reduct)	28
3.8 Eşdeğerlik Sınıfları ve Ayırt Edilemezlik.....	29
3.9 Küme Yaklaşımları	30
3.10 Kaba Küme Yaklaşımlarında Kullanılan Mekanizmalar	30
3.11 Tahmin Doğruluğu (Yaklaşım Kesinliği)	32
3.12 Özellik (Giriş Değerlerinin) Bağımlılığı.....	33
3.13 Kaba Kümeler ve Temel Elemanları	34
3.14 Genetik Algoritma ve Genetik Algoritma Tabanlı Sistemler ve Biyolojik Sistemlerle Matematiksel Sistemlerin Benzerliği.....	36
3.15 Genetik Algoritma Temelli Sistemlerin Genel Yapısı ve Terminolojisi	40
3.16 Permutasyon Kodlama ya da Sayısal Kodlama	43

3.17 Değer Kodlama	45
3.18 Genetik Algoritmada Ağaç Kodlama.....	45
3.19 Çaprazlama Operatörleri	46
3.20 Kullanılan Mutasyon Yöntemleri.....	50
3.21 Genetik Algoritma Temelli Sistemlerde Kullanılan Diğer Mutasyon Yöntemleri.....	53
3.22 Genetik Algoritmada Kullanılan Seleksiyon Yöntemleri	53
3.23 Genetik Algoritma Kullanmayan ve Alt Kümeleri	54
3. 24 Diğer Bazı İndirgeme Yaklaşımları	59
3.25 Yapay Sinir Ağı (YSA) ve YSA Tabanlı Sınıflandırıcı.....	61
3.26 Yapay Sinir Ağları Temelli Sistemlere Genel Bakış	61
3.27 Yapay Sinir Ağları Modellemesinde Dikkat Edilecek Noktalar.....	66
3.28 Yapay Sinir Ağlarında Eğitim Kuralları ve Tanımları.....	67
3.29 Yapay Sinir Ağlarında Geri Yayılım (Back Propagation) Algoritmasının Temel Özellikleri	68
3.30 Geriye Doğru Hesaplama.....	69
3.31 Sınıflandırma Doğruluğu Formülü.....	71
3.32 Medikal Alanda Sensitivite, Spesifite ve Kesinlik Kavramları	71
3.34 Yapay Sinir Ağı Temelli Öğretmensiz Öğrenme Modelleri	74
3.35 Veri Madenciliğinde Diğer Bazı Bilgi Keşfi Algoritmaları.....	76

4. ÖNERİLEN YÖNTEM

4.1 Genetik Algoritma Kullanılan Bazı Terimler ve Geliştirilen Yazılım Arayüzü	78
4.2 Lokal ve Global Optimum Durumları.....	81
4.3 GARSBS’de Kullanılan Durdurma Kriteri	83
4.4 Yeni Geliştirilen Modifiye Suni seçim Mekanizması.....	83
4.5 Rulet Çemberi Mekanizmasında Eklenen Yeni Modifikasyon.....	84
4.6 Geliştirilen Genetik Alg. ve Kaba Küme Temelli Yazılımda Kullanılan Durdurma Kriterleri	86

5. DENEYSEL SONUÇLAR

5.1 Geliştirilen İndirgeme Yazılımı ve Diğer Yazılımlarla Karşılaştırılması.....	88
--	----

5.2 Yeni Önerilen Modifiye Rulet Seçim Mekanizmasındaki Ekran Çıktısı (Dışarıdan Aday Çözüm Kümesi Eklemeli)	97
6. SONUÇLAR.....	101
7. KAYNAKLAR	102
8. EK ÜROLOJİK VERİ KULLANIM İZİN FORMU	105



1 GİRİŞ

1.1 Çalışmanın Amacı

Veri madenciliği birçok girişin bulunduğu sistemlerde, baskın olan girişleri tespit etmek için çalışan ve esnek hesaplama sistemlerinin gelişen bir koludur. Çok girişli sistemlerin işlenmesi ve hangi giriş değerlerinin daha önemli olduğunun belirlenmesi veri madenciliğinin en önemli kollarındandır. Veri madenciliği teknikleri, karmaşık görünen birçok sistemin hangi faktörlerden daha çok etkilendiğinin ortaya çıkarılmasının yanı sıra bu giriş değerleri arasındaki bağlantılar ve birliktelikler vb. konularda da çalışmaktadır. Veri madenciliği ve bilgi keşfi sistemleri tıp, mühendislik, biyomedikal ve birçok branşta kullanılmaktadır.

Çok girişli ve büyük veri tabanlarını işlenmesi bilgisayar destekli sistemlerde yüksek hafıza kullanımına, kilitlenmelere ve aşırı yüklenmeden dolayı sistemlerin hata vermesine sebep olmaktadır. Bu yüzden baskın ve çıkış değeri üzerinde belirleyici bir öneme sahip olan giriş değerlerinin belirlenmesi gerekmektedir. Aynı zamanda medikal elektronikte hastalıklar ve hastalıkların hangi giriş ve ölçüm değerlerinden etkilendiğinin belirlenmesi oldukça önemlidir.

Çalışmada, çok girişli medikal sistemlerdeki baskın giriş değerlerini tespit eden genetik algoritma ve kaba küme teorisini kullanan bir indirgeme sistemi yazılımı geliştirilmiştir. Çok girişli medikal sistemlerin yapay zeka tekniklerini kullanarak işlenebilir hale gelmesi için, baskın giriş değerlerinin geliştirilen genetik algoritma ve kaba küme kullanan hibrit sistem yazılımı tarafından belirlenmesi amaçlanmıştır. Birçok veri madenciliği yazılımı çok girişi olan sistemleri hafıza, kilitlenme ve aşırı yüklenme gibi problemlerden dolayı işleyememektedir. Bu problemlerin hibrit sistem yazılımı ile çözülmesi amaçlanmıştır. Çalışmada önerilen modifiye seçim mekanizmaları sistem performansını arttırmıştır. Ürolojik veriler işlenerek baskın giriş değerleri tespit edilmiştir. Ayrıca dermatolojik veri tabanı (UCI) kullanılarak da baskın giriş değerleri tespit edilmesi amaçlanmıştır.

Esnek hesaplama ve esnek hesaplama tabanlı sistemler, yapay sinir ağları genetik algoritma gibi birçok yapay zekâ tekniğini içermektedir. Esnek hesaplama temelli sistemler, veri madenciliği algoritmalarını da içermekte ve geliştirilen ve hibrit olarak kullanılabilen yazılımlar sayesinde, veri tablolarının analizini sağlamaktadır.

1.2 Çalışmanın Literatüre Katkısı

Geliştirilen genetik algoritma ve kaba küme temelli yazılımla ve önerilen modifiye genetik seçim mekanizmaları sayesinde çok girişli sistemler ve veri kümeleri hafıza hatası olmadan analiz edilebilmektedir. Ayrıca genetik algoritma temelli sistemlerde karşılaşılan yerel optimuma takılma problemi çözülmüştür. Yapılan testlerde yüksek sınıflandırma doğruluğu elde edilmiştir. Geliştirilen yazılım ile medikal veri madenciliği sistemlerinde baskın giriş değerleri hesaplanmıştır. Literatürde yer alan bir çok veri madenciliği algoritması bilgisayar ortamında yüksek hafıza kullanımından dolayı çok girişli sistemlerde hata verilebilmekte ve bazı algoritmalar sadece az giriş sayısına sahip olan sistemlerde kullanılabilmektedir. Bu problemler geliştirilen hibrit sistem yazılımıyla çözülmüştür. Çalışmada önerilen modifiye edilmiş seçim mekanizmalarıyla sistem performansı artırılmıştır. Farklı indirgeme algoritmaları da yazılım ortamına taşınarak destekledikleri giriş sayıları ve performansları bilgisayar ortamında test edilmiştir.

1.3 Veri Madenciliği ve Esnek Hesaplama

Veri madenciliği günümüzde tıp, biyoloji ve daha birçok sektörde kullanılmaktadır. Verilerin içinde saklı verilere ve değerli bilgilere ulaşılması, değişkenler arasındaki ilişkilerin belirlenmesi ve veri setlerindeki baskın değerlerin ortaya çıkarılması amaçlarıyla kullanılmaktadır.

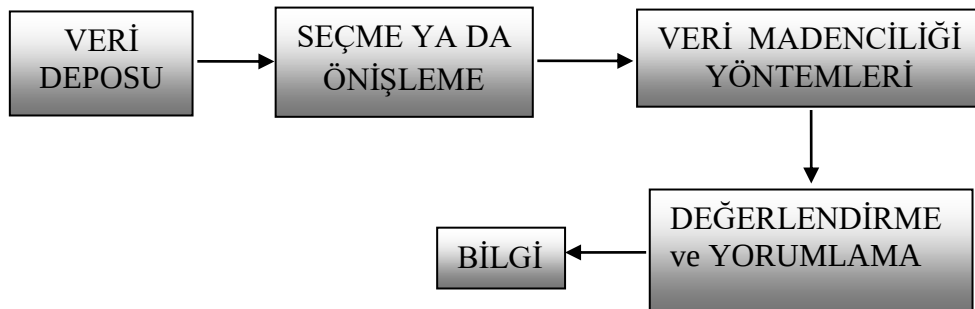
Esnek hesaplama sistemleri içerisinde, veri madenciliği, yapay sinir ağları vb. birçok yapay zekâ modellemesi ve bunların hibrit modifiye ve ilerletilmiş versiyonlarını içermektedir. Günümüzde esnek hesaplamanın ve bu branşa bağlı olan sistemlerin medikal sistemlerdeki önemi hızla artmaktadır. Bilgi keşfi ve esnek hesaplama temelli sistemler, makine öğrenmesi, yapay zeka, bilgi çıkarımı, uzman sistemler ve yüksek performanslı hesaplama sistemleri dahil olmak üzere gün geçtikçe ilerlemektedir.

Günümüzde veri keşfi ve esnek hesaplama teknikleri makine öğrenmesi, yapay zekâ tekniklerini de kapsamaktadır. Veri keşfinin temel adımları aşağıdaki gibi gösterilebilir. Veri keşfinin temel aşamaları kullanılan tekniğe ve giriş verisinin türüne göre değişiklikler göstermekle birlikte aşağıdaki gibi ifade edilebilir. Büyük veri kümelerinin bulunduğu birçok alan bulunmaktadır. Bunlar arasında biyoinformatik ve medikal sistemler, sağlık ve hasta takibi, medikal görüntüleme, üretim ve daha birçok

alan sayılabilir. Veri madenciliğinin ilgilendiği konular, genellikle açıklayıcı ya da öngörücü durumdadır. Esnek hesaplama sistemleri, çok çeşitli olmakla birlikte birçok yapay zekâ branşını kullanan sistemlerdir. Veri madenciliği ve bilgi keşfi temelli sistemler, veri tabanlarında veriler arasındaki ilişkileri ve bunlardan faydalananarak, öngörücü ya da sınıflandırıcı özelliklere sahiptir.

Büyük veri tabanlarında işlenen verilerin değerlendirilmesi oldukça zahmetli zor ve zaman alıcıdır. Ayrıca giriş değişkenlerinin sayısının artması da veri işleme zamanlarını arttırmakta ve bu yüzden çok yüksek girişli verilerde veri işlenirken işletim sistemi tarafından ayrılan hafıza aşıldığından dolayı hafıza yetersiz kalmaktadır. Bu yüzden genetik algoritma ve kaba küme temelli hibrit bir indirgeme yazılımı geliştirilmiştir. Geliştirilen yazılıma yeni modifiye edilmiş seçim mekanizmaları eklenerek sistem performansı arttırılmıştır. Yapılan çalışmada medikal giriş verilerinin sayısı indirgenerek baskın girişler tespit edilmiştir. Çok giriş sayısını desteklemesi sebebiyle birçok indirgeme algoritmasına göre üstünlükleri bulunan medikal indirgeme yazılımının elde ettiği indirgenmiş giriş değerleri daha sonra değişken girişli yapay sinir ağı kullanan sisteme bağlanarak sistem test edilmiştir. Geliştirilen yazılım sayesinde hem çok girişli sistemler desteklenmiş hem de hafızanın etkin bir şekilde kullanımı sağlanmıştır. Çalışmada, Üroflovmetrik ve rezidüel giriş değerleri ve *UCI* (Makine Öğrenmesi Veri Deposu) den elde edilen dermatolojik veri tabanı veri işleme kısmında kullanılmıştır.

Aşağıda temel bir veri tabanından veri keşfi sistemi ve basamakları Şekil1.1’de gösterilmiştir (Mitra ve Acharya 2003).



Şekil 1.1. Veri keşfi sistemi ve genel yapısı

Veri Keşfi ve veri madenciliğinin yukarıda gösterilmiş basamaklardan daha az ya da daha fazla basamakları da olabilmektedir. İşlenilen verinin cinsi, sınıflandırma sisteminin yapısı vb. birçok faktörden dolayı ek bazı sistemlere de ihtiyaç duyabilmekte ve hibrit yaklaşımlar da kullanabilmektedir. Veri keşfinde aşağıdaki adımlar oldukça önem taşımaktadır (Mitra ve Acharya 2003).

1. **Uygulama Alanının Değerlendirilmesi:** Uygulama alanı ve amaçlarının gözden geçirilmesi işlemidir.
2. **Hedef Veri Tabanının Belirlenmesi:** Hedef veri tabanı seçimi, veri seti seçimi ya da giriş değerlerinin belirlenmesi olarak ifade edilebilir. Kullanılacak özellik çıkarımı mekanizmalarının hedef veri tabanı seçimine uygun olması gerekmektedir. Ayrıca kullanılacak giriş değerlerini indirgeme yazılımının, giriş sayısı desteğinin giriş verisinin boyutuna uygun olması gerekmektedir.
3. **Veri Önileme ve Veri Madenciliği Aşamaları:** Veri önileme, genellikle veri madenciliğinde, veri kalitesinin artırılması için yapılmaktadır. Veri önileme aşamaları veri işleme aşamalarıyla iç içe geçmiş olabilir. Veri önileme ve veri madenciliği sistemleri aşağıdaki aşamaları içerebilir. Veri madenciliği sistemleri ve modellemeleri hızla ilerlediğinden dolayı bir veya daha çok aşama birbirini içermekte ve ondan faydalanmaktadır. Esnek hesaplama sistemlerinin en önemli özelliği yapay zekâ teknikleri ve onların arasındaki ilişkilerden faydalanılarak daha gelişmiş sistemlerin geliştirilmesini sağlamasıdır (Hand ve ark, 2001).
 - a)**Veri Temizleme:** Bu aşama verinin daha net algılanmasını sağlamakla birlikte bu aşamaya örnek olarak normalizasyon, parazit sinyallerin temizlenmesi, veri dönüşümü, veri entegrasyonu, veri azaltımı ve sıkıştırılması gösterilebilir.
 - b)**Veri Entegrasyonu:** Bu aşama genelde farklı veri tabanlarına uyumluluğu sağlama aşaması olarak da kabul edilmektedir.
4. **c)Veri Azaltımı ve projeksiyonu ve yorumlaması:** Bu aşama genelde, veriyi temsil edecek özelliklerin bulunmasını içermektedir. Ve genellikle veri azaltımı, boyut azaltımı, özellik (nitelik) çıkarımı aşamalarını içermektedir. Ayrıca

fonksiyonel bağımlılıkların hesaplanması da bu aşamada olmaktadır (Hand ve ark, 2001).

Veri madenciliği aşamaları ise aşağıdaki adımlardan bazılarını içermektedir. Bu adımlar, sınıflandırma, regresyon, kümeleme, görüntü çıkarımı, bağımlılıkları belirleme vb. aşamalardır. Yorumlama aşamasında keşfedilen örnekler yorumlanır. Diğer bir aşama ise keşfedilen bilginin kullanılmasıdır. Bu bilginin bir performans sistemiyle birleştirilmesi ya da sistemin bilgi tabanlı olarak kullanılması olarak yorumlanabilir. Veri madenciliğinde en çok kullanılan algoritmalar ve yaklaşımlar ve genellikle çıkarımsal olarak elde edilen soyutlamanın nesnel hale getirilmesinde ve belirsiz görülen verilerin arasındaki bağlantıları ortaya çıkarmada kullanılır. Veri madenciliğinin pratik uygulamaları arasında aşağıdaki alanlar sayılabilir. Ayrıca veri madenciliği sistemleri aşağıda listelenen alanlarla sınırlı olmamakla birlikte uygulamada aşağıdaki işlem ve yaklaşımları içerebilir.

1) Sınıflandırma: Sınıflandırmada kullanılan model ya da yaklaşım veri kümesini önceden belirlenen bir kategorik sınıfa dahil eder. Verilerin sınıflandırma süreçlerinin genellikle en temel adımları şunlardır:

- a) İlk adım, veri kümelerine uygun bir modelin oluşturulmasıdır. Veri tabanında kayıtların yani giriş değerlerinin belirlenmesi aşamasıdır.
- b) İkinci adım olarak sınıflandırmada kullanılacak model belirlenir.

2) Regresyon: Regresyon analizi 2 ya da daha çok değişken arasındaki ilişkiyi incelemek için kullanılan yöntemlerdir. Analizde tek değişken kullanılması halinde çok değişkenli regresyon olarak adlandırılmaktadır. Regresyonun ilk şekillerinden biri, en küçük kareler yöntemidir. Regresyon yöntemi yapay zeka temelli bazı yaklaşımlarda geliştirilerek literatürde kullanılmıştır. Bu terim bağımsız değişkenlerle bağımlı değişken arasında bir tahmin fonksiyonu oluşturmaya çalışmaktadır.

5. Kümeleme: Bu fonksiyon genelde bir nesneyi birçok küme grubundan birine atamaktadır. Kümeleme analizi ya da kümeleme, nesnelerin gruplandırılması

anlamına gelmektedir. Veriler arasındaki benzerlikler gruplandırma yapılırken dikkate alınmaktadır. Veri madenciliğinin önemli aşamalarından biri olarak kabul edilmektedir. Kümeleme ya da gruplandırma aynı grup içinde yer alan objeleri takım olarak ifade etmektedir. Veri madenciliği, örnek tanıma görüntü analizi ve biyoinformatik gibi alanlarda kullanılmaktadır. Kümeleme analizi, bir çözümleme problemi olarak ifade edilir. Genelde popüler kümeleme terimleri, küme üyeleri arasındaki küçük mesafelerle ifade edilmektedir. Veri uzayında yer alan yoğunlaşmış alan kümeleriyle gösterilmektedir.

1.4 Veri Madenciliğine Genel Bakış

1.4.1 Veri madenciliğinin bazı uygulama alanları

Veri madenciliği uygulamalarının kullanıldığı alanlar arasında medikal alandaki uygulamalar, hastalar, hastalıklar ve semptomların değerlendirilmesi, tıp elektroniği, ölçümler ve hastalıklar arasındaki ilişkiler, satış tahminleri, promosyon vb. hizmetler, müşteri profilinin tespit edilmesi, finansal veriler arasındaki gizli ilişkilerin belirlenmesi, veri depolarında çeşitli değişkenlere göre yapılan analizler ve gözardı edilen veriler arasındaki ilişkilerin ortaya çıkarılması, veri madenciliğinin araştırdığı ve problem çözümüne katkı sağladığı alanlardır.

1.4.2 Kaba kümeler (rough sets) ve bilgi tabanlı sistemler

Veri madenciliği ve yapay zekânın kapsamındaki problemlerin çözülmesinde son yıllarda geliştirilmekte olan ve halen de gelişen kaba küme (rough sets) temelli yaklaşımlar, birçok tecrübeye dayalı yapay zekâ temelli yöntemden daha başarılı sonuçlar vermektedir. Kaba küme temelli olarak geliştirilen ve geliştirilmeye devam edilen algoritmalar, verinin indirgenmesi, niteliklerin minimal alt kümelerinin bulunması, kuralların en iyi minimal kümesinin hesaplanması, verinin sınıflandırılması, analizi vb. amaçlarla kullanılmışlardır. Veri tabanı ve bilgi sistemlerine ait verilerin sınıflandırılması, kural çıkarımlarının oluşturulması için kullanılan kaba küme temelli yaklaşımlar, son yıllarda oldukça önem kazanmıştır. Literatürde, tecrübeye dayalı yapay zeka yöntemleriyle kaba küme yaklaşımı entegre edilerek yeni algoritmalar geliştirilmiştir ve halen geliştirilmeye devam etmektedir.

Bu tez çalışmasında, kaba küme genetik algoritma temelli hibrit bir indirgeme yazılımı geliştirilmiştir. Değişken girişli yapay sinir ağı temelli sınıflandırma yazılımı da sistem performansını ve elde edilen indirgenmiş özellik kümelerini test etmek için geliştirilen indirgeme yazılımının çıkışına eklenmiştir.

Kaba kümeler temelli hibrit yaklaşımlar, veri madenciliği ve esnek hesaplama yöntemlerinin arasında yer alan, özellik çıkarımı vb. birçok amaçla kullanılan ve hale gelişmekte olan yapay zekâ temelli yöntemlerden de faydalanan yöntemler bütünüdür. Geliştirilmiş olan genetik algoritma kaba küme ve yapay sinir ağları temelli hibrit yaklaşımı kullanarak giriş niteliklerinin indirgemesi yapılarak baskın girişler tespit edilmiştir.

Kaba kümeler teorisinde özellik indirgemesinde karara bağlı ayırt edilebilirlik matrisi (decision relative discernibility matrix) yaklaşımı ve kaba küme teorisine ait indirgeme prosedürleri test amacıyla oluşturulan yazılımda kullanılmıştır. Kaba küme teorisi yapılan çalışmada özellik indirgeme safhasında kullanılmıştır.

Ayrıca özellik bağımlılığına ait olan indirgeme metodu da özellik indirgeme için kullanılmış ve indirgenmiş özellik kümeleri elde edilmiştir. Test aşamasında özellik bağımlılığı kullanan sistem için oluşturulan yazılımda, bütün alt küme kombinasyonları için özellik bağımlılığı değerleri hesaplanmıştır. Oluşturulan yazılımlar esnek bir yapıya sahip olup text dosyasından veri okuyabilmekte ve giriş sayısı kullanıcı isteğine göre arttırılabilmektedir.

Özellik bağımlılığı ve modifiye edilmiş seçim mekanizmalarını içeren indirgeme yazılımı ve karara bağlı ayırt edici matris içeren indirgeme yazılımı performanslarının ve sınıflandırma doğruluklarının hesaplanabilmesi için değişken girişli yapay sinir ağı temelli sınıflandırıcıya bağlanmıştır. Sınıflandırma doğruluğu indirgenmiş özellik kümelerine göre ürolojik ve dermatolojik test verilerine göre yapılmıştır.

Kaba Küme teorisi, yalnızca kümenin elemanları ile tanımlandığı ve kümenin elemanları hakkında ilave hiçbir bilginin bulunmadığı klasik küme teorisinin aksine, bir kümenin tanımlanması için başlangıçta küme elemanları ve onların özellikleri hakkındaki bazı bilgilerin değerlendirme süreci için kullanılması gerektiğini öne süren bir yaklaşımlar bütünüdür. Eğer özellik uzayında yer alan nesneler, belirtilen nitelik değerleri göz önüne alındığı zaman aynı değerleri alıyorsa o zaman nesneler, belirtilen

nitelik değerleri için ayırt edilemez olarak kabul edilir. Elemanter küme,bütün aynı nesnelerin kümesine denmektedir.

Kaba küme yaklaşımlarının farklı versiyonları bulunmakla birlikte, bu yaklaşımda herhangi bir muğlak kavram, bir kesin kavramlar çifti ile karakterize edilir ve bunlara belirsiz 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 içermektedir. Alt ve üst yaklaşımlar arasındaki fark sınır bölgesini oluşturmaktadır (Pawlak 1997).

Kaba küme 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 oluşturulması olarak belirtilebilir. Kaba küme teorisinin kullanılabileceği ana problemler; verilerin indirgenmesi, bağımlılıklarının keşfi, verilerin öneminin tahmini, verilerden karar (kontrol) algoritmalarının oluşturulması, verilerin yaklaşık sınıflaması, verilerdeki benzerlik ve farklılıkların keşfi, verilerdeki örneklerin (pattern) keşfi, neden ve sonuç ilişkilerinin keşfini kapsamaktadır (Pawlak 1997). Ayırt edilemezlik fonksiyonu ve Boole cebri indirgeme kuralları küme teorisinin geliştirilmesiyle elde edilmiştir ve her geçen gün de ileriletilmektedir.

1.4.3 Klasik küme teorisi

Klasik küme teorisinde, elemanlar bir kümenin elemanı olabilir ya da olmayabilirler. Bu teoride, nesne ya kümenin elemanları arasındadır ya da eleman değildir şeklinde sınıflandırılır. Genel bir ifadeyle açıklanacak olursa U , nesnelerin uzayı olarak kabul edilir ve x de U 'nun bir elemanıdır. A (Kesin küme) için $A \subseteq U$, $x \in U$ dur. Burada x , kümenin elemanı olabilir ya da olmayabilir. İki kümenin eşit olduğu ancak A' da yer alan her ancak her ikisi de aynı elemanları içerdiği zaman eşit olduğu zaman söylenebilir. Bir kümenin kardinalitesi, küme içindeki elemanların sayısının bir ölçüsüdür ve $|A|$ şeklinde ifade edilir. İçinde elemanı olmayan küme boş küme olarak kabul edilir ve küme teorisinde " $\emptyset$ " ile gösterilir ve kardinalitesi sıfır kabul edilir. Kaba küme teorisi ve indirgeme sistemlerinde klasik küme teorisi ve aksiyomları oldukça önemli bir yer tutar (Polkowski, 2002).

1.4.4 Eşit Kümeler

A ve B 'yi herhangi iki küme kabul etmek üzere, A kümesinin her elemanı B kümesinin de elemanı ise A ile B kümeleri eşittir denir. Kısaca aynı elemanları içeren ve aynı elemanlardan oluşan kümeler eşittir. Eşit olmayan kümelere farklı kümeler denilmektedir. Klasik küme teorisindeki bazı kavramlar aşağıda ifade edilmiştir (Polkowski 2002).

1.4.5 Kardinalite

Bir kümenin kardinalitesi, bir kümenin elemanlarının sayısının bir ölçüsüdür. Örneğin A kümesi $A=\{a_1, a_2, a_5, a_7, a_9, a_{11}, a_{12}\}$ elemanlarından oluştuğunda A kümesinin kardinalitesi 7'dir. Kardinalite, literatürde farklı anlamlara sahiptir. X ve Y herhangi iki küme olarak kabul edildiğinde, X ile Y aynı sayıda elemana ve dolayısıyla aynı kardinaliteye sahiptirler. X 'in elemanları Y 'nin bütün elemanları ile eşleşebiliyor ise Y ile Y aynı kardinaliteye sahiptirler. Boş kümenin kardinalitesi sıfıra eşit olacağından boş kümeyle aynı kardinaliteye sahip bir Y kümesi varsa, Y kümesinin kardinalitesi de sıfıra eşittir. " n " sayıda elemanı olan bir kümeyle aynı kardinaliteye sahip olan kümenin de kardinalitesi de " n "dir (Polkowski, 2002).

Kardinalite özellikle kaba küme teorisi için büyük önem taşımaktadır. Yapılan çalışmada çok giriş sayılı sistemler değerlendirildiği için alt küme sayıları ve dolayısıyla alt küme sayılarının sayısı da oldukça fazladır.

Sonlu küme eleman sayısı nümerik olarak ifade edilebilen kümeleri yani sonsuz olmayan kümeleri ifade etmektedir.

1.4.6 Cantor teoremi

Küme teorisinin en önemli teoremlerinden biri kabul edilir. $P(Y)$ kuvvet kümesini temsil ettiği zaman, Y kümesinin elemanları ile $P(Y)$ kümesinin elemanları ile birebir eşleme yapılamaz bunun sebebi kardinalitelerinin eşit olmamasıdır. Ancak kardinaliteleri eşit olan kümeler arasında birebir eşleme yapılabilir. Herhangi bir " Y " kümesinin " n " adet elemanı olduğu düşünülürse, " Y " kümesinin alt küme sayısı 2^n ile ifade edilir. Her " n " doğal sayısı için 2^n , " n " sayısına göre daha büyük olduğu için

bir Y kümesinin kuvvet kardinalitesi, Y kümesinin kardinalitesinden büyüktür. Cantor Teoremi Denklem 1.1’ de verilmiştir. (Wikipedia, 2022)

$$Car(X) < Car(P(X)) \dots \dots \dots (1.1)$$

1.4.7 Altküme Kavramı

Eğer A kümesinde yer alan her bir eleman B kümesinin bir üyesi olması durumunda A kümesine B kümesinin alt kümesi denmektedir ve $A \subseteq B$ olarak ifade edilir. Eğer A, B ’nin bir alt kümesi ise ve B ’ye eşit değilse, o zaman A özalt kümedir ve $A \subset B$ şeklinde gösterilir. Aşağıdaki denklemlerde alt küme ve özalt küme sayısını göstermektedir. Bir kümenin eleman sayısı " n " olmak üzere kümenin alt küme sayısı Denklem 1.2’de, özalt küme sayısı Denklem 1.3’ de gösterilmiştir. Alt küme kavramı özellikle çok girişli sistemlerdeki indirgenmiş özellik kümelerinin sayısı bakımından önemlidir.

$$\text{Alt küme Sayısı} \dots \dots \dots m = 2^n \dots \dots \dots (1.2)$$

$$\text{Öz Alt küme Sayısı} \dots \dots \dots z = 2^n - 1 \dots \dots \dots (1.3)$$

“ n ” elemanlı bir kümenin “ r ” elemanlı alt küme sayısı Denklem 1.4 ile ifade edilir.

$$C(n, r) = \binom{n}{r} = \frac{n!}{r!(n-r)!} \dots \dots \dots (1.4)$$

Geliştirilen indirgeme yazılımı, çok girişli sistemleri desteklediğinden dolayı indirgenmiş özellik kümelerinin sayısı da fazla olmaktadır. Genetik algoritma ve kaba küme kullanan modifiye seçim mekanizmalı yaklaşım, hafıza hatalarından sistemi kurtarmakta ve çok girişi desteklemektedir. Örneğin 20 girişteki toplam öz altküme sayısı ve boş küme hariç altkümelerin toplam sayısı (kendisi dahil), 1048575’tir. Aşağıda klasik küme teorisi ve kaba küme teorisine ait bazı özellikler verilmiştir.

“n” Elemanlı Bir Kümenin En Çok “r” Elemanlı Alt Küme Sayısı

“n” elemanlı bir alt kümenin bir elemandan başlayarak en çok “r” elemanlı alt küme sayısı aşağıdaki Denklem 1.5 ile hesaplanmaktadır.

$$\binom{n}{1} + \binom{n}{2} + \binom{n}{3} + \binom{n}{4} + \dots + \binom{n}{r} \dots\dots\dots(1.5)$$

“n” Elemanlı Bir Kümenin En Az “r” Elemanlı Alt Küme Sayısı

“n” elemanlı bir kümenin en az “r” elemanlı alt kümelerinin sayılarını hesaplamak için ise Denklem 1.6’ daki formül kullanılmaktadır.

$$\binom{n}{r} + \binom{n}{r+1} + \binom{n}{r+2} + \binom{n}{r+3} + \dots + \binom{n}{n} \dots\dots\dots(1.6)$$

Kuvvet Kümesi

Küme teorisinde A herhangi bir küme olmak üzere, A kümesinin tüm alt kümelerinin sayısı o kümenin kuvvet kümesi olarak ifade edilmektedir. “n” elemanlı bir “a” kümesinin kuvvet kümesinin eleman sayısı Denklem 1.7 ile belirlenir.

$$s(p(A)) = 2^n \dots\dots\dots(1.7)$$

Yansıma Özelliği: Her küme, kendisinin bir alt kümesidir.

Geçişme Özelliği: “A”, “B” ve “C” verilen üç alt küme olmak üzere, Eğer A kümesi B kümesinin alt kümesi ise ve “B” kümesi de “C” nin alt kümesi olması durumunda, “A” kümesi “C” kümesinin alt kümesidir. Geçişme özelliği Denklem 1.8’ de gösterilmiştir.

$$(A \subset B) \wedge (B \subset C) \Rightarrow (A \subset C) \dots \dots \dots (1.8)$$

Ters Simetri Özelliği: “ A ” ve “ B ” kümeleri ele alındığı zaman, Eğer A kümesi B kümesinin alt kümesi ise ve B kümesi de A kümesinin alt kümesi ise, o zaman A kümesi B kümesine eşittir. Birbirlerinin alt kümesi olan iki küme birbirinin alt kümesidir. Bu özelliğe Ters Simetri özelliği denmektedir ve Denklem 1.9’da gösterilmiştir.

$$(A \subset B) \wedge (B \subset A) \Leftrightarrow (A = B) \dots \dots \dots (1.9)$$



2. KAYNAK ARAŞTIRMASI

Literatürde Kaba küme temelli sistemler ve bunları kullanan karar destek sistemleri ile ilgili çalışmalar bulunmaktadır. Ayrıca çözümlenmesi zor problemler için genetik algoritma temelli bazı yaklaşımlarda bulunmaktadır.

Genelleştirilmiş kaba kümeler ve temel özelliklerini yaptığı çalışmada tartışmışlar ve oluşan ikili ilişkileri değerlendirmişlerdir. Genelleştirilmiş kaba küme kavramını genel olarak incelemişlerdir (Kondo 2006). Kaba küme yaklaşımından hangisinin seçileceğini ve hangi yaklaşım için nasıl bir kriter belirleneceği de veri madenciliğinde önemli bir yer tutmaktadır. Veri analizi için nasıl bir kaba küme yaklaşımının kullanılması gerektiğinin belirlenmesi de veri madenciliği ve buna bağlı veri analizi yöntemleri için önemlidir. Herbert ve Yao yaptıkları çalışmada 2 adet kaba küme modelini incelemişlerdir. Bunlar Pawlak modeli ve probablistik (ihtimallere bağlı olan) modellerdir (Herbert, Yao 2006).

Tıbbi uygulamalar ve kaba kümelerin kullanımları üzerine literatürde çeşitli çalışmalar vardır. Tıbbi bilgiler analiz edilmesi kolay olmayan birçok modelleme ve bileşen içerebilmektedirler. Kaba küme geliştirildiğinde ve teorik olarak ilerledikçe bu teorinin tıbbi uygulamalar için kullanılması bu çalışmada araştırılmıştır. Pattaraintakorn ve Cercone çalışmalarında Pawlak'ın kaba küme temellerinden başlayarak kaba kümeler ve veri madenciliğindeki ilerlemeleri incelemişlerdir (Pattaraintakorn ve Cercone 2008).

Shen ve Loh yaptıkları çalışmada kaba kümeleri pazar zamanlama problemi üzerine kullanmışlardır. Yapılan çalışmada yatırımcılar için stok tahmini problemi için kaba küme teorisinin kullanılması amaçlanmıştır (Shenve Loh 2003).

Johnson yaptığı çalışmada, Kombinatorial problemler için yaklaşım algoritmaları araştırılmıştır. Çalışmada basit, polinomal zamanlı ve sezgisel algoritmaların bir çok polinomal optimizasyon problemine uygulamaları yaptığı çalışmada araştırılmıştır (Johnson 1974).

Yao yaptığı çalışmada küme kökenli kaba kümeler ve kaba küme temelli yaklaşımlar tanıtmış ve genel özelliklerini incelemiştir (Yao 1998).

Genetik algoritma birçok problemin çözümünde kullanılmıştır. Levintin ve ark. yaptıkları çalışmada üretim hattı problemi için genetik algoritmayı kullanmışlardır. Farklı robotların farklı üretim hatlarında çalıştığı ve her bir robotun farklı bir işe atandığı durumlarda genetik algoritma kullanarak en uygun çözüm ve sırayı bulmaya çalışmışlardır (Levintin ve ark 2006).

Yeh ve Shan Lin yaptıkları çalışmada simulasyon tekniği ve genetik algoritma kullanarak hastane acil servisteki bakım hizmetini iyileştirmeye çalışmışlardır. Simülasyon modeli hastanın acil servis akışına göre çalışmada geliştirilmişler ve en hızlı hasta bakımı için optimal hemşire bakım programını genetik algoritmayla çözmeye çalışmışlardır. Böylece acil serviste hastaların daha az beklemesi amaçlanmıştır (Jinn-YiYehWen-Shan Lin, 2007).

Ghosh ve Mitchell yaptıkları çalışmada tıbbi görüntülerin segmentasyonunu (bölümlendirilmesini) genetik algoritma kullanarak yapmışlardır. Yapılan çalışmada iki boyutlu bilgisayarlı tomografi görüntülerinde prostat üzerindeki segmentasyon genetik algoritma kullanılarak yapılmıştır. Yapılan çalışmada bazı segmentasyon teknikleri de gözden geçirilmiştir (Ghosh ve Mitchell 2006).

Gábor Rennera, Aniko' Eka' rta, bilgisayar destekli tasarımda genetik algoritma ve kullanım alanlarını genel olarak incelemişlerdir. Genetik algoritmanın karmaşık optimizasyon problemlerinin de çözümünde kullanıldığını belirtmişlerdir. Bir farklı tasarım probleminin çözümünde genetik algoritma kullanımının örnekleri verilmiştir. Mühendislik tasarımında farklı alanlarda yapılan genetik algoritma uygulamalarına değinilmiştir (Rennerand ve Ekart, 2003).

Bhattacharjee ve Pant yaptıkları çalışmada hibrit parçacıklı sürü optimizasyonlu ve çok katmanlı persepton kullanan moleküler beyin neoplasti verilerinden insan gliomunu sınıflandıran bir yaklaşım üzerinde çalışmışlardır. Çok katmanlı perseptonların özel uygulama ve eğitim berilerine ihtiyaç duyduğunu belirtmişlerdir (Bhattacharjee ve Pant, 2019).

Siori ve Arkadasları yaptıkları çalışmada selülöz üretiminde genetik algortima modellemesi ve optimizasyonunu yapmışlardır. Matematik ve genetik optimizeler en yüksek enzim kombinasyonunu elde etmek için kullanmışlardır. Bu çalışma aynı

zamanda genetik algoritmanın atık yönetiminde de kullanılabileceğini vurgulamıştır (Siori ve Ark. 2018).

Lee, yaptığı çalışmada genetik algoritma uygulamalarını işlem yönetiminde incelemiştir. İşlem yönetimi deterministik olmayan Polinomal zor problemlerdir bu yüzden sezgisel metodlar karar mekanizmasını ilerletmek için uygulanmaktadır. Yaptığı çalışmada bu konuda yapılan çalışmalar genel olarak incelenmiş ve işlem yönetiminde genetik algoritma kullanımının genel prensiplerini açıklamışlardır (Lee 2018).

Mota ve arkadaşları genetik algoritma kullanarak yük optimizasyonu problemini çözmeye çalışmışlardır. Yapıkları çalışmada enerji maliyetlerinin düşürülmesi için genetik algoritma temelli sistelli sistem kullanmışlardır. Yenilenebilir enerji kaynaklarının dünyada elektrik enerji üretiminde oldukça yüksek bir yer tutacağı ve bu yüzden enerji verimliliğinin sağlanması için genetik algoritma temelli yük yönetiminin önemini vurgulamışlardır (Mota ve Ark 2022).

3. MATERYAL VE YÖNTEM

3.1 Üroflowmetrik ve Rezidüel ölçümler ve Kullanılan Veri Tabanları

Tez çalışmasında indirgeme sistemi için 20 girişli ürolojik veri tabanının yanısıra, 12 girişli ürolojik veri tabanı ve “UCP” de yer alan dermatolojik verileri işleyen veri tabanı kullanılmıştır. 12 girişli ürolojik sistem performans testleri için kullanılmıştır. Üroflowmetrik ölçümler, birçok ürolojik rahatsızlığın tanısında oldukça önem taşımaktadır (Tanagho ve Deng 2008). Ürolojik veri tabanlı uroflowmetrik değerler kullanılarak Üroloji Uzmanı yardımıyla oluşturulmuştur. Üroflowmetrik ölçümlerden maksimum ve ortalama akış hızı, çok düşük, düşük, orta ve yüksek olarak dört adet dilsel ifade ile rezidü ise yok, orta, yüksek ve çok yüksek olarak 4 adet dilsel ifade ile belirtilmiştir. a_1 maksimum akış hızını (idrar akış hızı), a_2 , ortalama akış hızını, a_3 rezidü miktarını, $a_4, a_5, \dots, a_{20}$ ise $T/4$ ile $3T/4$ arasında alınan üroflowmetrik değerleri gösterir. $a_4, a_5, \dots, a_{20}$ ölçüm değerleri ortalama akış hızı referans alınmıştır. Çıkış dilsel ifadeleri ise çok riskli, riskli ve sağlıklı olarak 3 adet dilsel ifade ile yazılmıştır. Üroflowmetrik referans değerler literatüre göre alınmıştır (Vesna ve ark., 2010). Tabloda 20 giriş ve 120 satıra sahip olan sisteme ait olan bazı satırlar ve üroflowmetrik ve rezidüel değerler verilmiştir. Tablo 3.1 de 20 girişli, 1 çıkışlı ve 120 satırlı ürolojik veri tabanına ait olan bazı giriş değerleri gösterilmiştir. Tablo 3.2’ de 12 girişe sahip olan ürolojik giriş değerleri tablosu görülmektedir. 12 giriş değerine sahip olan tablo sistemin diğer sistemlerle test edilebilmesi için eklenmiştir. 20 girişli sisteme benzer şekilde a_1 maksimum akış hızını (idrar akış hızı), a_2 , ortalama akış hızını, a_3 rezidü miktarını göstermekte, $a_4, a_5, \dots, a_{12}$ ise $T/4$ ile $3T/4$ arasındaki üroflowmetrik ölçüm değerlerini göstermektedir. $a_4, a_5, \dots, a_{12}$ üroflowmetrik erleri T , toplam akış zamanını (voiding time) göstermekte olup, $T/4$ ile $3T/4$ arasında alınan veriler o ana denk gelen üroflowmetrik akış hızını göstermektedir. Örneğin voiding time (İdrar Boşaltım Zamanı) = 120 sn olduğu zaman $T/4$, 30. sn deki ve $3T/8$ ise 90. sn deki akış hızını vermektedir. " a_8 " ise $T/2$ de akış hızını ml/sn cinsinden göstermektedir. Akış hızı boşaltım zamanı'nın farklı anları için periyodik olarak hesaplanarak aktarılmaktadır. Test amacıyla 10 girişli ürolojik veride sisteme eklenmiştir. 10 girişli ürolojik veride de 12 girişli ürolojik veriden farklı olarak, $a_4, a_5, a_6, \dots, a_{10}$ ise sırasıyla $T/4$ ve $3T/8$ arasında kalan üroflowmetrik değerleri göstermektedir.

Tablo 3.2 de 12 girişli ürolojik veri tabanına ait girişler bulunmaktadır. Tablo 3.10 'da ise 10 girişli uroflovetrik ölçümlerin olduğu sistem gösterilmiştir. 10 girişli sistem, karşılaştırma kullanılan bazı algoritmaların çok giriş sayısını desteklememelerinden dolayı kullanılmıştır. 10 girişli 2. Tabloda x_1 maksimum akış hızını (idrar akış hızı), x_2 , ortalama akış hızını, x_3 rezidü miktarını, $x_4, x_5, \dots, x_{10}$ ise $T/4$ ile $3T/4$ arasında alınan üroflovetrik değerleri göstermektedir.

Aşağıda referans verilen bazı değerler bulunmaktadır.

Maksimum Akış Hızı (mlt/s)

Çok Düşük.....	$0 \text{ mlt} / s \leq x < 10 \text{ mlt} / s$	1
Düşük.....	$10 \text{ mlt} / s \leq x < 19 \text{ mlt} / s$	2
Orta.....	$19 \text{ mlt} / s \leq x < 30 \text{ mlt} / s$	3
Yüksek.....	$30 \text{ mlt} / s \leq x \leq 40 \text{ mlt} / s$	4

Ortalama Akış Hızı (mlt/s)

Çok Düşük.....	$0 \text{ mlt} / s \leq x < 7 \text{ mlt} / s$	1
Düşük.....	$7 \text{ mlt} / s \leq x < 14 \text{ mlt} / s$	2
Orta.....	$14 \text{ mlt} / s \leq x < 25 \text{ mlt} / s$	3
Yüksek.....	$25 \text{ mlt} / s \leq x \leq 40 \text{ mlt} / s$	4

Rezidü Miktarı (ml)

Yok.....	0 mlt	1
Orta.....	$0 \text{ mlt} < x < 50 \text{ mlt}$	2
Yüksek.....	$50 \text{ mlt} \leq x < 150 \text{ mlt}$	3
Çok Yüksek.....	$150 \text{ mlt} \leq x < 500 \text{ mlt}$	4

Üroflovetri, zamana karşı idrar akış hızının ölçümünü yapan ve ürolojik rahatsızlıkların teşhisinde önemli bir rol oynayan bir testtir. Test invazif olamayan mesane ve sfinkter fonksiyonu ve üretral ve böbrek rahatsızlıklarını değerlendirmede kullanılan bir testtir.

Üroflovetride ölçüm cihazı, idrarın akış hızını saniyeye bağlı olarak hesaplamaktadır. Ölçüm cihazı idrarın boşaltımı için geçen toplam zamanı, toplam akış zamanını ölçer. Bu bilgiler cihazdan grafiksel olarak alınır ve normal idrar akış hızını engelleyen bir durum olup olmadığına bakılır. Üroflovetride ilk önce akış hızı

düşüktür sonradan yükselir. Sağlıklı veri alınması için uroflowmetri grafiğinin $T/4$ ve $3T/4$ arasında kalan aralığı alınmıştır. Ayrıca üroflowmetrik değerler idrara sıkışma miktarına göre de değişkenlik göstermektedir. Üroflowmetri değerlerinin normal çıkmasını engelleyen bazı rahatsızlıklar aşağıda verilmiştir. Bu rahatsızlıklar aşağıda ifade edilen rahatsızlıklarla sınırlı olmayıp şu şekilde sıralanabilir. Rezidü test sonrasında mesanede kalan idrar miktarını ifade eder.

Prostat Hipertrofisi: Prostat dokusunun genişleyerek yayılması ve kontrolsüz büyümesi ile ilgilidir. Prostatın büyümesi ve özellikle üretraya bağlı yapacak şekilde içe doğru büyümesi, idrar akış hızını azaltmaktadır.

Prostat Kanseri ve Tümörü : Üriner Blokaj ve üriner sistemi tıkalı oluşumlar. Bu tür rahatsızlıklar böbrek dahil tüm idrar yolunu etkileyebilmekte ve idrar akış hızını düşürmektedir.

Nörojenik mesane sendromu da mesanenin görevini tam olarak yapmasını engellemektedir. Üroflowmetrik ölçümler aşağıdaki rahatsızlıklar ve semptomlar için önemlidir. Mesane kapasitesinin etkilendiği durumlar, kapasitenin arttığı ve azaldığı durumların tespiti bulunmaktadır. Hiperaktif mesane, detrusor instabilitesi sayılabilir. Üriner sistem obstrüksiyonu nedeni ile tedavi ve operasyon geçiren hastaların takibinde üroflowmetrik ölçüm sonuçlarının önemi büyüktür.

Üroflowmetri (idrara akış hızı, idrar boşaltım zamanı vb ölçüm) testin kullanıldığı rahatsızlıklar yukarıda sayılanlarla sınırlı değildir. Bununla birlikte üroflowmetri sonuçlarını etkileyen bazı etkenler de bulunmaktadır. İlaç tedavisi ve mesane üzerinde etkili olan bazı ilaç yan etkileri bunların arasında gösterilebilir. Test öncesinde mesanenin tamamına yakının dolu olması ve mesanenin test yapılmadan önce boşaltılmaması gerekmektedir. Bu test tek başına ya da sistometri ile birlikte kullanılabilir. Üroflowmetriden elde edilen değerler:

- 1) Maksimum akış hızı (maksimum flow rate, F_{max})
- 2) Ortalama akış hızı (mean flow rate, F_{mean})
- 3) Akış Süresi (Voiding Time, V_{time})
- 4) Residüel idrar (post-voiding residual, PVR)
- 5) Anlık Grafikselleştirme ölçüm değerleri

Tablo 3.1 20 Girişli Üroflovetrik ve rezidüel değer tablosu

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}	a_{14}	a_{15}	a_{16}	a_{17}	a_{18}	a_{19}	a_{20}	d
1	1	1	4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	2	1	3	1	2	1	1	1	2	1	2	2	1	1	1	2	1	1	1	1	1
3	1	1	2	1	1	1	1	2	1	2	1	1	1	1	1	1	1	1	1	1	1
4	2	1	1	1	2	1	1	1	2	1	1	1	1	1	1	2	1	1	2	1	1
5	1	1	3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
6	2	1	2	1	2	1	1	2	1	1	1	1	2	1	1	2	1	1	2	1	1
7	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
8	2	2	3	1	2	1	2	1	1	1	2	2	2	1	1	1	1	2	1	1	1
9	3	2	2	2	2	2	2	1	2	2	2	3	2	3	2	1	2	2	2	2	2
10	2	2	3	2	2	2	1	2	2	2	2	2	2	1	2	1	2	1	2	1	1
11	2	2	2	1	2	1	2	1	2	1	2	2	2	1	1	1	2	1	2	1	1
12	2	2	4	1	1	1	1	1	1	2	1	2	2	1	2	1	1	1	1	1	1
13	3	2	1	1	2	2	1	2	2	2	3	2	2	2	2	2	2	2	2	2	2
14	3	2	2	1	3	1	2	2	2	2	2	2	2	2	2	3	1	2	2	2	2
15	3	2	2	2	2	2	1	2	3	2	2	3	1	2	3	2	2	1	2	2	2
16	3	2	2	2	2	2	2	2	2	2	2	2	3	1	2	2	2	2	3	1	2
17	2	2	3	1	2	2	1	2	1	1	2	2	1	1	2	1	1	1	1	1	1
18	3	3	1	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	2	3
19	3	2	2	2	2	3	2	3	2	2	3	2	1	2	2	2	2	2	2	2	2
20	3	2	1	2	2	2	2	2	3	2	2	3	2	1	2	2	2	2	2	2	2
21	3	2	1	2	3	2	2	2	3	2	2	2	3	2	2	3	2	1	2	2	2
22	3	2	2	1	2	2	2	2	3	2	2	3	2	2	1	2	3	2	2	2	2
23	3	2	1	2	2	2	2	2	1	2	3	2	2	2	3	2	2	2	2	2	2
24	3	2	2	2	3	2	2	2	2	2	2	2	3	2	2	2	2	2	2	2	2
25	2	2	2	2	2	2	2	2	2	2	2	3	2	2	2	2	2	2	2	2	2
26	2	2	2	2	2	2	1	2	3	1	2	2	1	1	1	2	2	2	1	1	1
27	2	2	2	1	1	2	2	2	2	2	1	1	2	2	2	1	1	1	2	1	1
28	2	2	3	1	2	2	1	1	2	2	2	2	1	1	2	1	1	2	1	1	1
29	2	2	1	2	2	3	2	2	2	2	2	2	3	2	1	2	2	2	2	2	2
30	2	2	3	1	1	2	2	2	1	1	2	2	2	1	1	1	2	2	1	1	1

....

Tablo 3.2 12 Girişli Üroflovetrik ve Rezidüel değerler tablosu

	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	d
1	1	1	4	1	1	1	1	1	1	1	1	1	1
2	2	1	3	1	2	1	2	2	1	1	1	2	1
3	1	1	2	2	1	2	1	1	1	1	1	1	1
4	2	1	1	1	2	1	1	1	1	1	1	2	1
5	1	1	3	1	1	1	1	1	1	1	1	1	1
6	2	1	2	2	1	1	1	1	2	1	1	2	1
7	1	1	1	1	1	1	1	1	1	1	1	1	1
8	2	2	3	1	1	1	2	2	2	1	1	1	1
9	3	2	2	1	2	2	2	3	2	3	2	1	2
10	2	2	3	2	2	2	2	2	2	1	2	1	1
11	2	2	2	1	2	1	2	2	2	1	1	1	1
12	2	2	4	1	1	2	1	2	2	1	2	1	1
13	3	2	1	2	2	2	3	2	2	2	2	2	2
14	3	2	2	2	2	2	2	2	2	2	2	3	2
15	3	2	2	2	3	2	2	3	1	2	3	2	2
16	3	2	2	2	2	2	2	2	3	1	2	2	2
17	2	2	3	2	1	1	2	2	1	1	2	1	1
18	3	3	1	3	3	3	3	3	3	3	3	3	3
19	3	2	2	3	2	2	3	2	1	2	2	2	2
20	3	2	1	2	3	2	2	3	2	1	2	2	2
21	3	2	1	2	3	2	2	2	3	2	2	3	2
22	3	2	2	2	3	2	2	3	2	2	1	2	2
23	3	2	1	2	1	2	3	2	2	2	3	2	2
24	3	2	2	2	2	2	2	2	3	2	2	2	2
25	2	2	2	2	2	2	2	3	2	2	2	2	2
26	3	3	1	3	3	3	4	4	3	3	4	3	3
27	3	3	1	3	3	4	3	3	4	4	3	3	3
28	3	3	1	3	3	4	3	3	3	4	3	4	3
29	4	3	1	3	3	3	3	4	3	3	4	3	3
30	4	3	1	3	3	3	3	4	3	3	3	3	3

...

3.2 İşlenen Standart (UCI) Dermatolojik Veri Tabanı

Geliştirilen modifiye suni seçim mekanizmalı indirgeme yazılımında ürolojik veri tabanları dışında UCI standart veri tabanı da kullanılmıştır. Kullanılan dermatolojik veri tabanına ait bazı girişler aşağıda verilmiştir. Veri Tabanı olarak UCI (Makine öğrenmesi Veri Deposu) veritabanı kullanılmıştır. Veri tabanında 33 giriş 1 çıkış bulunmaktadır.

Veritabanında 366 satır bulunmaktadır. Yaş girişli eksik veri bulunması nedeniyle ihmal edilmiştir, böylece 34 giriş yerine 33 giriş kullanılmıştır. Çıkış değişkeni başlangıç aşamasında birbirine benzeyip kolay ayırt edilemeyen 6 adet dermatolojik rahatsızlığı sınıflandırmaktadır. Sınıflandırmadaki “d” çıkış değişkeni 6 adet çıkış değişkenine sahiptir. a_{11} giriş değişkeni ailede aynı rahatsızlığın görülüp görülmemesini gösterir ve 0 ile 1 değişkenleriyle gösterilir. Klinik giriş değerleri $a_1, \dots, a_{11}$ değişkenleriyle, histopatolojik (dokusal patolojik) değerler ise $a_{12}, \dots, a_{33}$ değerleriyle gösterilir. Bu değerler ve karşılıkları aşağıda gösterilmiştir. Giriş değerlerindeki 0, özelliğin bulunmadığını, 3 ise en yüksek oranda bulunduğunu, 1 ve 2 ise dereceli olarak orta derecede ölçülen giriş değerinin bulunduğunu göstermektedir. Çıkış ise dermatolojik rahatsızlığı sınıflandırmaktadır. Çıkış 1 ise hastalık Sedef, 2 ise Seboreik Dermatit 3, 4 ya da 5 olduğunda ise sırasıyla Liken Planus, Piiriasis Rosea (Gül Hastalığı), Kronik Dermatit (chronic dermatitis), Pitriasis Rubra Pilaris, (pityriasis rubrapilaris) olduğunu göstermektedir (Güvenir, H. A., ve ark 1998).

Birçok dermatolojik rahatsızlığın teşhisi, benzer semptomlar gözlemlendiğinden dolayı oldukça zor olabilmekte ve birçok nitelik değerinin değerlendirilmesi gerekmektedir. Bu yüzden veri işlenmesinde baskın giriş değerlerinin tespit edilmesi oldukça önemlidir.

İşlenen dermatolojik veri tabanı UCI den alınmış olup ait bazı giriş değerleri aşağıdaki Tablo 3.3 de verilmiştir (Güvenir, H. A ve ark., 1998).

$a_1, \dots$ Kızarıklık “erythema, eritem”

$a_2, \dots$ Skaling

$a_3, \dots$ Kesin Sınırlar

$a_4, \dots$ Kaşıntı

.....

$a_7, \dots$ folüküler papüller,

$a_8, \dots$ ağız içi (oral) durum (yaraların ilerlemesi),

$a_9, \dots$ diz ve dirsekteki durum (ilerlemesi),

$a_{10}, \dots$ kafa derisinde olup olmadığı,

$a_{11}, \dots$ Rahatsızlığın ailede olup olmadığı, (0 or 1)”

.....

a₂₁.....Epidermal Kalınlaşmanın Uzaması

a₃₁.....Perifolüküler Parakeratoz,

.....

a₃₃.....Band Şeklinde İnfiltrat (dokunun içine nüfuz etme), band-likeinfiltrate

d.....Sınıflandırma ve Karar Değişkeni

1.....Sedef, (Psoriasis)

2.....Seboreik Dermatit,

.....

5.....Kronik Dermatit

6.....Pitriasis Rubra Pilaris

3.2.1 Dermatolojik terminoloji hakkında açıklamalar

Aşağıda giriş değişkeni olarak verilen değişkenlerin tıbbi açıklamaları bulunmaktadır.

Eritem (Kızarıklık) Eritem, kızarıklık ve döküntü ile kendini gösterebilen rahatsızlıktır. Bir çok çeşidi olmakla birlikte ana çeşitleri arasında ışık duyarlılığına bağlı olan şekli sayılabilir. Genellikle tıbbi işlemlere reaksiyondan, ilaç tedavilerine oluşan yan etkilerden ya da enfeksiyonlardan olabilmektedir.

(<https://saglik.wiki/eritem>, 2022)

Koebner Fenomeni: Koebner fenomenine aynı zamanda koebner tepkisi ya da izomorfik tepki de denilmektedir. Genellikle terim travma çizgilerinde gözüken deri lezyonlarını açıklar. Bu durum, Derinin irritasyona uğraması veya doğrusal temas sonrası oluşabilmektedir.

Melanin İnkontinansı: Melanin pigmentindeki anormal değişimleri açıklamak için kullanılmaktadır. Melanin aslen latince olan melas kelimesinden türetilmiştir. Deride renk değişiminin insanı güneşin zararlı ışınlarından koruduğu bilinmektedir.

Papüller: Sınırları belirgin halde olan, kabarık, genellikle 1 cm'den daha küçük çaplı lezyonlardır. Kahverengi, mor, pembe ya da kırmızı renkte olabilir ve papüller

döküntüye de sebep olabilir. Papüller kazındığı ya da çıkarılmaya çalışıldığında enfekte olup, kabuk bağlayabilirler.

Sedef (Psoriasis) : Psoriasis (Sedef) , deri yüzeyinde görülen bir hastalıktır. Bu deri rahatsızlığında deri hücrelerinin çok hızlı bölünmesi söz konusudur. Normalde deri hücrelerinin yenilenmesi üç dört hafta sürerken psoriasisli hastalarda bu yenilenme süreci çok daha hızlı olmaktadır. Bulaşıcı bir rahatsızlık olmadığı düşünülmektedir. Sedef hastalığında düzgün olmayan deri dokusunun yenilenme işleminde hem ölü hem de canlı hücreler deri yüzeyinde birikmektedir. Gümüş görünümlü pullanmalar görülmektedir.

Plak Psoriasisi (Plak Sedefi): Psoriasis hastalarının çoğu plak psoriasisi taşımaktadır. Genellikle kabarmış, kırmızı lezyonlar gümüşsü beyaz tabaklarla kaplıdır. Dirseklerde, dizlerde, baş derisinde ve derinin herhangi bir yerinde görülebilir.

Eritrodermik Psoriasis: Eritrodermik Psoriasis, vücudun bir çok yüzeyini etkileyen bir rahatsızlıktır. Kararsız Plak Psoriasisi olan hastalarda genellikle görülmektedir. Ayak bileklerinde ödem oluşturabilmektedir. Vücudun sıcaklık düzeninde değişimler oluşur

Gutat Psoriasisi (sedefi): Küçük kırmızı lekeler şeklinde genellikle başlamaktadır. İlk dönemlerde küçük kırmızı ve birbirinden bağımsız lekeler şeklinde başlayıp, normal olarak plak psoriasisi kadar kalın tabakalar şekline gelebilir. Gutat sedefi tetikleyen birçok şart vardır. Bunlar arasında, bazı mikrobiyel enfeksiyonlar, stres ve deri yaralanmaları gözlemlenmektedir.

Pustular Psoriasis: Genellikle yetişkinlerde görülen sedef şeklidir ve beyaz sivilcelerle karakterize edilir. İltihabi kökenli ve bulaşıcı değildir. Vücudun çeşitli bölgelerini özellikle el ve ayakları etkilemektedir.

Seboreik Dermatit: Seborreik dermatit, bir deri hastalığıdır. Seborreik Dermatit belirtileri arasında deri lezyonları, büyük bir alana yayılmış plaklar bulunur. Kaşıntı, hafif kızarıklık ve saç kaybı da gözlemlenmektedir.

Liken Planus : Liken Planus, enflamatuvar bir deri hastalığıdır. Küçük ve çok parçacıklı düz ve mor bölgeler şeklinde kendini gösterir. Kollarda ve bacaklarda ağızda ve tırnaklarda ve baş derisinde görülebilir. Baş derisinde ilerlemesi saç dökülmesine neden olur. Sebepleri arasında bazı ilaçların bilinçsiz kullanımı hastalığı arttırmaktadır. Oto bağışıklık hastalığıdır ve bayanlarda daha sık görülmektedir.

Gül Hastalığı (Piiriasis Rosea): Çok bilinen ve görülen bir deri hastalığı olup, deride döküntüye sebep olmaktadır. Mantar ve bakterilere karşı vücudun alerjik bir tepkisidir. Hastalığın yaptığı derideki bozulma sonucu, döküntülere sebep olmaktadır. Döküntüler, tek oval ve pembe lezyonlar görünümündedir.

Kronik Dermatit: Deride döküntü şeklinde görülür. Genellikle geçici olanları alerjik ya da kimyasal bir rahatsız edici maddeler tarafından tetiklenebilir. Başka bir dermatolojik rahatsızlığın beraberinde de gelişebilir ya da gözlemlenebilir. Sedef rahatsızlığıyla birlikte görülebilir.

Pityriasis rubrapilaris: Birbirine benzeyen bir dermatolojik rahatsızlık grubunu ifade etmektedir. Bu hastalık grubu, kabarıp pullanarak dökülme, kızıllaşma ve saç folüküllerinle olan ilişkisi gibi bazı ortak belirtiler göstermektedir. Bu hastalık gurubunun farklı şekilleri sedef hastalığına benzemektedir. Teşhis ve tedavinin için biyopsi gerekebilmektedir. Tipine göre tedavisi de değişmektedir. Dermatolojik ve daha birçok medikal sistemde baskın nitelik(giriş) değerlerinin tespiti oldukça önemlidir. Veri madenciliği rahatsızlıklar ve rahatsızlıklar arasındaki ilişkilerin ortaya çıkarılması ve veri keşfinde çok önemlidir.

Tablo 3.3 Dermatolojik veri tabanı (UCI)

	a_1	a_2	a_3	a_4	a_5	...	a_{25}	a_{26}	a_{27}	a_{28}	a_{29}	a_{30}	a_{31}	a_{32}	a_{33}	d
1	2	2	0	3	0	...	0	0	0	3	0	0	0	1	0	2
2	3	3	3	2	1	...	0	0	0	0	0	0	0	1	0	1
3	2	1	2	3	1		2	0	2	3	2	0	0	2	3	3
4	2	2	2	0	0		0	3	0	0	0	0	0	3	0	1
5	2	3	2	2	2		2	2	3	2	3	0	0	2	3	3
6	2	3	2	0	0		0	0	0	2	0	0	0	1	0	2
7	2	1	0	2	0		0	0	0	0	0	0	0	2	0	5
8	2	2	3	3	3		0	2	2	3	2	0	0	3	3	3
9	2	2	1	0	2		0	0	0	2	0	0	0	2	0	4
10	2	2	1	0	1		0	0	0	2	0	0	0	2	0	4

11	3	3	2	1	1		0	0	0	0	0	0	1	0	1
12	2	2	0	3	0		0	0	0	3	0	0	1	0	2
13	3	3	1	2	0		0	0	0	2	0	0	1	0	2
14	2	3	3	0	0		0	0	0	0	0	0	2	0	1
15	2	2	3	3	0		2	0	3	0	3	0	1	3	3
16	1	1	0	1	3		0	0	0	2	0	0	2	0	4
17	2	2	1	3	0		0	0	0	0	0	0	1	0	2
18	3	3	3	0	0		0	3	0	0	0	0	2	0	1
19	2	1	3	3	3		3	0	2	0	3	0	2	3	3
20	1	1	0	3	0		0	0	0	1	0	0	2	0	5
21	2	1	1	2	0		0	0	0	0	0	1	2	1	0
22	3	2	2	0	0		0	0	0	3	0	0	2	0	2
23	2	2	0	2	0	...	0	0	0	1	0	1	0	2	5

.....

	a_1	a_2	a_3	a_4	a_5	...	a_{25}	a_{26}	a_{27}	a_{28}	a_{29}	a_{30}	a_{31}	a_{32}	a_{33}	d
24	2	2	2	3	2	...	3	0	3	0	2	0	0	2	3	3
25	2	0	0	3	0	...	0	0	0	0	0	0	0	2	0	5
26	2	1	1	0	1		0	2	0	0	0	0	0	2	0	1
27	1	1	0	1	0		1	0	0	1	0	2	2	1	0	6
28	1	2	2	3	0		0	0	0	1	0	0	0	3	0	5
29	3	2	2	2	0		0	0	0	2	0	1	1	2	0	2
30	1	1	2	3	2		3	0	3	0	3	1	0	2	3	3
31	3	2	1	2	0		0	2	0	0	0	0	0	1	0	1
32	3	2	0	2	0		0	0	0	0	0	0	0	1	0	2
33	2	3	3	3	3		0	0	0	0	0	0	0	1	0	1
34	2	2	1	0	0		0	0	0	0	0	0	0	0	0	1
35	2	1	0	0	2		0	0	0	0	0	0	0	0	0	4
36	2	2	1	2	0		0	0	0	0	0	0	0	0	0	2
37	2	1	2	3	2		1	0	2	0	2	0	0	0	3	3
38	2	1	1	1	0		0	0	0	2	0	0	0	1	0	2
39	2	1	2	3	2		2	0	1	0	3	0	0	2	3	3
40	3	3	2	0	0		0	0	0	0	0	0	0	1	0	1
41	1	1	1	0	0		0	0	0	3	0	0	0	1	0	2
42	1	1	1	0	0		0	0	0	2	0	0	0	2	0	4
43	3	2	1	1	0		0	0	0	0	0	0	0	1	0	1
44	2	1	1	0	0		0	0	0	2	0	0	0	2	0	2
45	0	1	0	3	0	...	0	0	0	0	0	0	0	1	0	5

.....

	a_1	a_2	a_3	a_4	a_5	...	a_{25}	a_{26}	a_{27}	a_{28}	a_{29}	a_{30}	a_{31}	a_{32}	a_{33}	d
46	2	1	1	1	1		2	0	2	0	2	0	0	3	3	3
47	2	1	1	3	3	...	0	0	0	1	0	0	0	2	0	4
48	2	1	1	1	0	...	0	0	0	0	0	0	1	1	1	6
49	1	2	1	1	0		0	0	0	2	0	0	0	0	2	2
50	2	0	1	0	0		1	0	3	0	2	0	0	2	2	3

51	3	1	1	2	2	1	0	2	0	3	0	0	2	3	3
52	1	1	1	0	2	0	0	0	0	1	0	0	1	0	4
53	2	2	1	0	0	0	0	0	0	0	0	0	1	0	1
54	3	1	1	1	1	0	0	0	0	0	0	0	2	0	1
55	0	1	0	2	0	0	0	0	0	0	0	0	3	0	5
56	2	2	2	0	0	0	0	0	0	0	0	0	1	0	1
57	1	1	1	2	0	0	0	0	2	0	0	0	2	0	2
58	2	1	1	3	0	1	0	3	0	1	0	0	2	2	3
59	1	1	1	0	0	0	1	0	1	0	0	0	1	0	4
60	3	3	0	3	0	0	0	0	3	0	0	0	2	0	2
61	2	2	1	1	0	0	0	0	1	0	1	1	1	0	6
62	2	1	1	0	0	0	1	0	0	0	0	0	1	0	1
63	1	1	0	1	0	0	0	0	0	0	0	0	1	0	5
64	3	3	2	1	1	0	0	0	0	0	0	0	1	0	1
65	2	2	0	3	0	0	0	0	2	0	0	0	2	0	2
66	3	2	2	2	0	2	0	2	0	2	0	0	1	3	3
67	2	2	2	0	0	0	3	0	0	0	0	0	3	0	1
68	1	1	0	1	3	0	0	0	2	0	0	0	2	0	4
69	3	3	3	2	0	0	0	0	0	0	0	0	3	0	5
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...

3.3 Kaba Küme Temelli Bilgi (Enformasyon) Sistemleri

Enformasyon sistemlerinde, nitelik (özellik) değerleri, giriş değerlerini temsil ederler. Enformasyon sistemlerinde baskın değerlerin tespit edilmesi ve veri içinde saklı olan bilginin ortaya çıkarılması hem veri madenciliği hem de esnek hesaplama temelli sistemlerin en temel amaçlarından biridir. Medikal enformasyon sistemlerinde her bir satır belirli ölçüm değerlerinde elde edilen durumu belirtmektedir. Sütunlardan her biri hastadan yapılan herhangi bir ölçüm değerini ya da hastada görülen ya da görülmeyen herhangi bir semptomu göstermiştir.

Kaba küme analizi için veriler, her satırı bir nesneyi (veya örneği) her sütunu nesneyle ilgili bir özelliği gösteren bir tablo şeklindedir. Özelliklerin değerleri ya ölçüm ile ya da insan deneyimi ile elde edilir. Böyle bir tabloya, enformasyon sistemi adı verilir. $A, A=(U, A)$ şeklinde bir bilgi (enformasyon) sistemi tanımlanır. U , burada A 'nın evreni adı verilen nesnelerin boş olmayan sonlu kümesini ifade ederken, özelliklerin boş olmayan sonlu kümesi ise A ile gösterilir. Burada herhangi bir a özelliği ise $a:U \rightarrow V_a$ şeklinde tanımlanır. a 'nın değer kümesine V_a , denmektedir. Bir karar sistemi $A=(U, A \cup \{d\})$ biçiminde tanımlanan bir bilgi sistemidir. Burada $d \notin A$,

karar özelliğidir. Burada A 'nın elemanlarına nesnenin özellikleri adı verilir (Pawlak, 1991).

3.4 Ayırt Edilemezlik (Indiscernibility)

Bir $R \subset X \times X$,ikili bağıntısı, yansıyan (yani bir nesne kendisi ile ilişkili xRx), simetrik (xRy ise yRx) ve geçişken (xRy ve yRz ise xRz) ise, denklik bağıntısı adı verilir. Bir $x \in X$ elemanının, denklik sınıfı (equivalence class), xRy olan bütün $y \in X$ nesnelerinden meydana gelmektedir (Pattaraintakorn,2008). Ayırt edilemezlik bağıntısı Denklem 3.1 de gösterilmiştir (Yang, 2009).

$$IND_A(B) = \{(x,y) \in U^2 \mid \forall a \in B \ a(x) = a(y)\} \dots\dots\dots(3.1)$$

Eğer $(x,y) \in IND_A(B)$ ise, x ve y nesneleri B 'ye göre ayırt edilemezdirler.

3.5 Ayırt Edilebilirlik

Bir bilgi sistemi ayırt edilebilirlik matrisi ile de ifade edilebilmektedir. Veri madenciliği üzerine literatürde yeterli minimal özelliklerin bulunması ve indirgeme ile ilgili çeşitli çalışmalar bulunmakla birlikte ayırt edilebilirlik matrisi de ve fonksiyonu da veri keşfi ve baskın özelliklerin tespitinde önemli bir yer tutmaktadır. Ayırt edilebilirlik fonksiyonu da bu yaklaşımlardan biridir. Bu fonksiyonun veriler arasındaki etkin giriş değerlerinin hesaplanmasında veri madenciliğinde uygulamaları bulunmakla birlikte veri işleme zamanı ve giriş sayısının sınırlandırılması vb. dezavantajları da bulunmaktadır (Yang, 2009).

3.6 Karara Bağlı Ayırt Edilebilirlik Matrisi

“ n ” adet nesnesi olan bir enformasyon sistemi olarak “ A ” kabul edildiği zaman, A 'nın ayırt edilebilirlik (discernibility) matrisi, M_A aşağıda verilen

c_{ij} elemanları olan simetrik bir $n \times n$ matristir. Bu matrisin her c_{ij} elemanı, x_i ve x_j nesnelerini farklı kılan özellikler kümesinden oluşmaktadır (Denklem 3.2).

$$c_{ij} = \{a \in A, a(x_i) \neq a(x_j)\} \quad i, j = 1, 2, \dots, n \dots \dots \dots (3,2)$$

(U, CUD) simetrik bir $|U| \times |U|$ matristir. Bir ayırt edici matriste Denklem 3.3 deki gibi ayırt edilebilirlik fonksiyonu aşağıdaki gibi hesaplanır (Denklem 3,3).

$$f_D(a_1^*, \dots, a_m^*) = \bigwedge \{ \bigvee C_{ij} \mid 1 \leq j \leq i \leq |U|, C_{ij} \neq \emptyset \} \dots \dots \dots (3.3)$$

Karara bağlı ayırt edilebilirlik matrisine sadece karar değişkeni farklı olan satırlar değerlendirilir. Her bir c_{ij} hücresi i ve j nesnelerinin birbirinden farklı olan özelliklerini içerir. Karar değişkeni farklı olan satırların birbirinden farklı olan özellikleri ayırt edilebilirlik matrisine yazılırlar. Her bir hücre içindeki elemanlar birbirlerine $\vee$ (birleşim) operatörü ile tüm hücreler ise $\wedge$ (kesişim) operatörüyle bağlanarak karara bağlı ayırt edilebilirlik matrisinin fonksiyonu elde edilir (Yang, 2009). Bu matris oluşturulurken her bir sütun ve satır altındaki verilerden farklı olan özellikler yazılır. Fonksiyon Boole Cebri indirgeme kuralları kullanılarak sadeleştirilir.

Geliştirilen yazılımla sonuçların karşılaştırılması amacıyla karara bağlı ayırt edici matris kullanan yaklaşımın hızlandırılmış versiyonu oluşturulmuş ve geliştirilen indirgeme yazılımı ile karşılaştırılmıştır. Karara bağlı ayırt edici matris yaklaşımını ve fonksiyonunu kullanan yaklaşımda desteklenen giriş sayısı, geliştirilen yeni modifiyeli seçim mekanizması kullanan genetik algoritma ve kaba küme kullanan hibrit sisteme göre daha azdır ve veri işleme zamanları daha fazladır.

Karara bağlı ayırt edici matris kullanan yaklaşım, yazılım ortamına taşındığı zaman daha az giriş sayısını desteklemektedir ve veri işleme zamanı daha uzundur. Bu yaklaşım yazılım ortamında, aşırı hafıza kullanma ihtiyacı bulunduğu için dolaylı, çok yüksek giriş sayılarını desteklememekte ve yüksek giriş sayılarında hafıza hataları vermektedir.

3. 7 Çekirdek İndirgenmiş Özellik Kümesi (The Core Reduct)

Çekirdek indirgenmiş özellik, bulunan bütün indirgenmiş kümelerinde ortak olan nitelik (özellik) ya da özelliklerden oluşur. Bazı bilgi sistemlerinde mevcut olup bazı

bilgi sistemlerinde olmayabilir. Çekirdek indirgenmiş özellik kümesi genellikle tüm indirgenmiş özellik kümelerinde (reduct) ortak olan indirgenmiş özellik kümelerini temsil etmektedir. Çekirdek indirgenmiş özellik kümesi tüm indirgenmiş özellik kümeleri içindeki ortak girişlerdir. Her sistemde çekirdek indirgenmiş özellik kümesi elde edileme bunun sebebi elde edilen reductların hepsinde birden aynı ortak giriş ya da girişler bulunmamasıdır.

3.8 Eşdeğerlik Sınıfları ve Ayırt Edilemezlik

Eşdeğerlik sınıfları, kaba küme teorisinde ele alınan özellikler (girişler) bakımından ayırt edilemeyecek durumları göstermektedir. Eşdeğerlik sınıfları aşağıdaki örnek enformasyon sistemi örnek alındığı zaman ve a_1, a_2, a_3, a_4 özellik değerleri dikkate alındığı zaman, T_1 ve T_3 'ün ve T_5, T_7 aynı eşdeğerlik sınıfına dahildir. Bu özellikler dikkate alındığı zaman oluşan eşdeğerlik sınıfları aşağıdaki gibidir. Aynı durum sadece a_1 ve a_2 değişkenleri dikkate alındığı zamanda aynıdır. Aşağıdaki örnek tabloda eğer sadece a_1, a_2, a_3 , değişkenleri de dikkate alındığı zaman eşdeğerlik sınıfları aynıdır. Eşdeğerlik sınıfları $\{T_1, T_3\}, \{T_5, T_7\}, \{T_2\}, \{T_4\}, \{T_6\}, \{T_8\}, \{T_9\}$ olarak bulunur. S, R, T, M giriş ve çıkışı gösteren dilsel ifadelerdir. Tablo 3.4 de Eşdeğerlik sınıfları için Örnek Enformasyon Sistemi verilmiştir.

Tablo 3.4 Eşdeğerlik Sınıfları İçin Örnek Enformasyon Sistemi

	a_1	a_2	a_3	a_4	a_5
T_1	S	S	T	T	K
T_2	K	T	T	R	R
T_3	S	S	T	T	K
T_4	M	S	R	T	K
T_5	R	T	T	R	R
T_6	S	T	K	R	S
T_7	R	T	T	R	R
T_8	T	R	S	M	R
T_9	K	R	M	R	T

$$\text{Eşdeğerlik Sınıfı}(a_1, a_2, a_3, a_4) = \{T_1, T_3\}, \{T_5, T_7\}, \{T_2\}, \{T_4\}, \{T_6\}, \{T_8\}, \{T_9\}$$

Eşdeğerlik Sınıfı $(a_1, a_2, a_3) = \{T_1, T_3\}, \{T_5, T_7\}, \{T_2\}, \{T_4\}, \{T_6\}, \{T_8\}, \{T_9\}$

3.9 Küme Yaklaşımları

Kaba küme teorisinde çeşitli yaklaşımlar bulunmaktadır. Bu yaklaşımlar alt yaklaşım ve üst yaklaşım şeklinde ifade edilebilmektedir. Kaba küme teorisindeki bazı yaklaşımlar ise ayırt edilemezlik yaklaşımını kullanmaktadır. $IND(B)$, ikili bağıntısını kullanarak kümelerin yaklaşımlarını oluşturmaktadır.

Kaba küme teorisi kullanan bazı yaklaşımlar, $IND(B)$ ikili bağıntısını kullanarak kümelerin yaklaşımlarını oluşturmaktadır. Alt ve üst yaklaşımlar (Denklem 3.4 ve 3.5) de gösterildiği gibi ifade edilebilir.

$$\underline{BX} = \{x \mid [x]B \subseteq X\} \dots \dots \dots (3.4)$$

$$\bar{BX} = \{x \mid [x]B \cap X \neq \emptyset\} \dots \dots \dots (3.5)$$

$\underline{BX}$ 'teki nesneler kesin olarak X 'in elemanları iken, $\bar{BX}$ 'teki nesneler sadece X 'in muhtemel elemanlarıdır. Burada B kümesindeki bilgi temeli de dikkate alınmaktadır. Sınır bölgesi eleman olup olmadığı tam belirli olmayan kısımdan oluşmaktadır. Kesin küme bir kümenin sınır bölgesi boş olduğunda kullanılan bir tabirdir. Elemanların hangi kümeye dahil olduğu ya da olup olmadığı belirlidir. $U - \bar{B}(X)$ kümesine, X 'in B 'nin dış bölgesi denmektedir. Pozitif ve Negatif bölgelerin tanımlarına göre sınır bölgesi ise aşağıdaki denkleme göre ifade edilmektedir (Pattaraintakorn, 2008) (Denklem 3.6).

$$POS(X) = \underline{X}; NEG(X) = U - \bar{X}; BND(X) = \bar{X} - \underline{X} \dots \dots \dots (3.6)$$

3.10 Kaba Küme Yaklaşımlarında Kullanılan Mekanizmalar

Geliştirilen indirgeme yazılımında (GARSBS) özellik bağımlılığı değeri kromozomların uygunluk değerleri olarak kabul edilmiştir. Kaba küme teorisinde çıkış

değerinin iki dilsel ifade ile ifade edildiği sistemlerde örneğin çıkış değeri hasta veya hasta değil gibi, pozitif, negatif ve ara bölge tanımlanmaktadır.

Aşağıda pozitif, negatif ve ara bölge elemanlarından oluşmuş bir küme ifade edilmiştir. Aşağıda kaba küme teorisinde kullanılan ve sınır bölgesini açıklamak verilen bir örnek bulunmaktadır. Aşağıda verilen tabloya göre tek tek giriş değerlerine göre ayırt edilemeyenleri inceleyelim. Aşağıda a_1 , a_2 ve a_3 değişkenleri üç adet medikal giriş değerini simgelemektedir. Bu medikal giriş değerlerinin hastalığın olup olmadığına göre yorumlanması, alt yaklaşım üst yaklaşım ve sınır bölge analizi yöntemine göre aşağıdaki gibidir. Alt yaklaşım ifadeleri için örnek bir veri tabanı Tablo 3.5’de gösterilmiştir.

Tablo 3.5 Alt Yaklaşım, Üst Yaklaşım İfadeleri

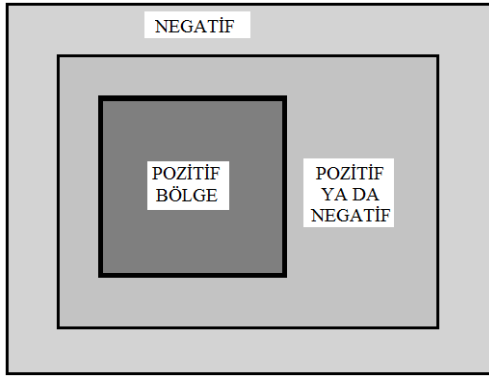
	a_1	a_2	a_3	<i>Çıkış</i>
T_1	S	A	A	2
T_2	S	B	D	1
T_3	T	C	B	1
T_4	T	C	B	1
T_5	K	D	S	2
T_6	S	B	D	2
T_7	M	D	S	2
T_8	T	B	C	1

İki değişkenli çıkış değerlerinde T_2 ve T_6 aynı giriş değerlerine sahip olmasına rağmen farklı çıkış değerlerine sahipler. A_1 , A_2 ve A_3 giriş değerlerine eşdeğerlik sınıfları oluşturulursa o zaman pozitif bölgede kalan elemanlar ve eşdeğerlik sınıfları aşağıdaki grafikte gösterildiği gibidir. Yukarıdaki tabloda, çıkış dilsel ifadesi olan "1" pozitif bölgeyi göstermek üzere, aşağıdaki eşdeğerlik sınıfları elde edilmiştir. T_2 ve T_6 sınır bölgesinde yer almaktadır çünkü aynı giriş değerlerine sahip olmalarına rağmen farklı çıkış değerlerine sahiptirler. Medikal sistemlerde aynı giriş değerine sahip olan sistemlerde farklı çıkış değerleri olabilmektedir. Şekil 3.1’de pozitif, negatif ve sınır bölgesi gösterilmektedir. 3 giriş değerine göre aynı eşdeğerlik sınıfına sahiptir.

$\{T_3, T_4, T_8\}$ Pozitif Bölge

$\{T_2, T_6\}$ Sınır Bölgesi

$\{T_1, T_5, T_7\}$ Negatif Bölge



Şekil. 3.1 Pozitif, Negatif ve Pozitif ya da Negatif Olan Bölge

Kaba küme ve eşdeğerlik bölgeleri, alt yaklaşım üst yaklaşım ve ara bölgeyle ilgili eşitlik ve ifadeler Denklemler 3.7- 3.12 arasında verilmiştir.

$$\underline{B}(X) \subseteq X \subseteq \overline{B}(X) \dots\dots\dots(3.7)$$

$$\overline{B}(X \cup Y) \subseteq \overline{B}(X) \cup \overline{B}(Y) \dots\dots\dots(3.8)$$

$$\underline{B}(X \cap Y) = \underline{B}(X) \cap \underline{B}(Y) \dots\dots\dots(3.9)$$

$$\begin{aligned} \text{Eğer } X \subseteq Y \text{ ise o zaman } \underline{B}(X) \subseteq \underline{B}(Y) \\ \text{ve } \overline{B}(X) \subseteq \overline{B}(Y) \end{aligned} \dots\dots\dots(3.10)$$

$$\underline{B}(X \cup Y) \supseteq \underline{B}(X) \cup \underline{B}(Y) \dots\dots\dots(3.11)$$

$$\overline{B}(X \cap Y) \subseteq \overline{B}(X) \cap \overline{B}(Y) \dots\dots\dots(3.12)$$

3. 11 Tahmin Doğruluğu (Yaklaşım Kesinliği)

Tahmin doğruluğu ya da yaklaşım kesinliği alt yaklaşımın üst yaklaşıma olan oranı ile hesaplanır ve Denklem 3.13 de gösterildiği gibidir.

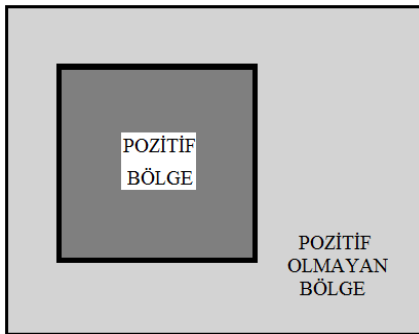
$$\alpha_B(X) = \frac{|BX|}{|X|} \dots\dots\dots(3.13)$$

Burada $|X|$, $X \neq \varnothing$ 'in eleman sayısıdır. $0 \leq \alpha_B(X) \leq 1$ 'dir. $\alpha_B(X) = 1$ ise, X, B 'ye göre kesin, aksi takdirde $\alpha_B(X) < 1$ ise, X, B 'ye göre kaba olarak ifade edilmektedir (Ślęzak 2009).

3.12 Özellik (Giriş Değerlerinin) Bağımlılığı

Veri analizinde önemli bir konu da değişkenler arasındaki bağımlılıkların hesaplanmasıdır. Q ' da yer alan özellik kümeleri P ' de yer alan tüm özellik (giriş) değerlerine bağımlılık derecesi aşağıdaki denklemle hesaplanabilir. Nitelik (Özellik) bağımlılığı pozitif bölgede yer alan elemanların tüm U 'ya oranı olarak ifade edilmektedir (Denklem 3.14). Özellik bağımlılığı değeri geliştirilen indirgeme yazılımına durdurma kriteri olarak kullanılmıştır. Özellik bağımlılığı değerini anlamak için Şekil 3.2'deki grafiğe bakılabilir. Geliştirilen indirgeme yazılımında özellik bağımlılığı değeri her bir kromozomun uygunluk değeri olarak kabul edilmektedir. Kullanıcı tarafından girilen özellik bağımlılık değerine göre yazılım durdurulabilmektedir. Özellik bağımlılığı (Nitelik Bağımlılığı), pozitif bölgede kalan elemanların, evrensel küme olarak kabul edilen U 'ya oranıdır. Şekil 3.2'de özellik bağımlılığı değerinin grafiksel sembolizasyonu gösterilmektedir. Geliştirilen indirgeme yazılımında klavyeden girilen özellik bağımlılığı değerine eşit veya daha yüksek değere sahip olan kromozomlar indirgenmiş özellik kümeleri olarak ifade edilir.

$$k = \gamma_P(Q) = \frac{|POSP(Q)|}{|U|} \dots\dots\dots(3.14)$$



Şekil 3.2 :Özellik Bağımlılığı Değerinin Grafiksel Sembolizasyonu

3.13 Kaba Kümeler ve Temel Elemanları

Kaba kümelerde, bilginin temsil edilmesi enformasyon sistemi, enformasyon sistemleri nesnelerin fonksiyonel temsili ile ifade edilebilir. (U, A) olarak bir bilgi (enformasyon) sistemi ifade edilmektedir. Burada U , bir küme (bir evren), A , bir özellik kümesi ve her bir özellik $a \in A$ a: U ve V_a da, “ a ” değerinin değer kümesidir. Ayırt edilemezlik, bir özellik kümesi içinde belirlenen özellik (giriş) değerlerine göre fonksiyonun ayırt edilemeyeceğini gösterir.

Kaba küme teorisi, veri madenciliği ve giriş özellik değerlerinin indirgenmesi gibi bilgi keşfinin ve değerlendirilmesinin önem arz ettiği durumlarda kullanılır. Veri madenciliği ve veri madenciliği temelli sistemler, bilgi keşfi ve belirtilen tablolarda saklı ya da ortaya çıkarılmamış bazı ilişkileri ortaya çıkarmak için oldukça önemlidir.

Kaba kümelerin veri madenciliğinde de önemli bir yeri bulunmaktadır. Veri madenciliği bir enformasyon sisteminde saklanmış olan bilgiyi keşfetmek ve ortaya çıkarmak için oldukça önemlidir.

Oluşturulan yazılımın desteklediği giriş değerlerinin karşılaştırılması amacıyla karara bağlı ayırt edici matris kullanarak da indirgeme yapan bir yazılım da hazırlanmıştır.

Karara bağlı ayırt edici matris kullanan yaklaşımda karara bağlı ayırt edici fonksiyon kullanılır. Karara bağlı ayırt edici matris benzeri yaklaşımlar, aşırı hafıza kullanımı, desteklenen giriş sayısının fazla olması ve veri işleme zamanının fazla olması gibi bazı dezavantajlara sahiptir.

Karara bağlı ayırt edici matrise ve benzer yaklaşımlar indirgemede karara bağlı ayırt edici fonksiyon kullanılır. Karara bağlı ayırt edici fonksiyonun kullanılan indirgemelerinde, Boolean cebri indirgeme kuralları kullanılır. Boole cebrinde kullanılan bazı indirgeme kuralları aşağıda verilmiştir. Burada $\cup$, birleşme operatörünü, $\cap$ ise kesişim operatörünü göstermektedir. Aşağıdaki denklemlerde (Denklemler 3.15-3.22) kesişim operatörü yerine “.” Kullanılmıştır (Polkowski, 2002).

$$x \cup y = y \cup x \dots \dots \dots (3.15)$$

$$x.y = y.x \dots \dots \dots (3.16)$$

$$(x \cup y) \cup z = x \cup (y \cup z) \dots \dots \dots (3.17)$$

$$(x.y).z = x.(y.z) \dots \dots \dots (3.18)$$

$$x \cup x.y = x \dots \dots \dots (3.19)$$

$$x.(x \cup y) = x \dots \dots \dots (3.20)$$

$$x \cup y.z = (x \cup y).(x \cup z) \dots \dots \dots (3.21)$$

$$x.(y \cup z) = x.y \cup x.z \dots \dots \dots (3.22)$$

Ayrıca, aşağıdaki Boole cebri denklemleri de kaba küme teorisi kullanıldığında aşağıdaki gibi ifade edilmektedir. (Denklem 3.23 - 3.30)

$$x \cup x = x \dots \dots \dots (3.23)$$

$$x.x = x \dots \dots \dots (3.24)$$

$$x \cup 0 = x \dots \dots \dots (3.25)$$

$$(x \cup y)' = x' y' \dots \dots \dots (3.26)$$

$$(xy)' = x' \cup y' \dots \dots \dots (3.27)$$

$$x'' = x \dots \dots \dots (3.28)$$

$$x \cup x' y = x \cup y \dots \dots \dots (3.29)$$

$$x.(x' \cup y) = x.y \dots \dots \dots (3.30)$$

Sistem performansını karşılaştırmak için, karara bağlı ayırt edici matris kullanan ve yazılımsal olarak hızlandırılmış versiyon aşağıda gösterilmiştir. Test yazılımı, Delphi

7 programlama dili kullanarak oluşturulmuştur. Genetik algoritma ve kaba küme temelli sistemle karşılaştırmak için oluşturulan ve karara bağlı ayırt edici ayırt edici matris kullanan yazılım versiyonu, Şekil 3.3’de aşağıda gösterilmiştir.

The screenshot shows a software window titled 'Form1'. On the left, there are several buttons: 'Construct Table', 'Reduction 1', 'Reduction 2', 'Load From File', and 'Save The Changes'. Below these are checkboxes for 'Working Time' and 'Memory Usage'. The main area contains a table with 10 rows and 13 columns. The columns are labeled a1 through a12, followed by a column labeled 's'. The rows are numbered 1 to 10. The table contains binary values (0 and 1). Below the table, there are several input fields and buttons, including 'x1', 'x2', 'x3', 'x4', 'x5', 'x6', and a large text area containing a complex mathematical expression. The right side of the window shows a list of variables and their values.

Şekil. 3.3 Karara Bağlı Ayırt Edici Matris Kullanan Versiyon

Modifiye seçim mekanizmalı genetik algoritma ve kaba küme kullanan idirgeme yazılımı hem fazla giriş sayısını destekleyebilmekte hem de daha efektif hafıza kullanımına sahiptir. Test amacıyla kullanılan, karara bağlı ayırt edici matris kullanan versiyon, yüksek Ram kullanım oranına ve işleme zamanına sahiptir ve bu sebeple de daha az giriş sayısını desteklemektedir.

3.14 Genetik Algoritma ve Genetik Algoritma Tabanlı Sistemler ve Biyolojik Sistemlerle Matematiksel Sistemlerin Benzerliği

Klasik, genetik algoritma temelli sistemler rastsal çözüm kümelerinden başlayarak çaprazlama, mutasyon operatörleriyle yeni nesil oluşturulur. Genetik algoritmalar, genetikten esinlenerek oluşturulmuş ve doğadan ilham alan matematiksel modellerdir. Kromozomlar, çözüm kümelerini belirleyen ve çeşitli şekillerde kodlanmış yapılardır. Genetik algoritmada kullanılan bazı temel bileşenler aşağıda özetlenmiştir. Çevremizde her bir yaşayan organizma, içinde genetik bilgiyi taşıyan kromozomlara sahiptir. Her bir kromozom uzun bir DNA molekülü içermektedir. Aynı biyolojideki genetik biliminde olduğu gibi, genetik algoritmada da çaprazlama ve mutasyon işlemleriyle yeni kromozomlar oluşmaktadır. Çaprazlama işleminde kromozomlar arasında gen alışverişi olur ve yeni kromozomlar oluşturulmuş

olur. Bütün yaşayan organizmalar hücrelerden oluşmuşlardır. Hücreler DNA'yı DNA da hücre için gerekli olan genetik kodu taşımaktadır. Her bir olası farklı kodlama aleller ismini almaktadır. DNA kodu ise her bir canlı türünün kendine özgüdür. Genetik algoritmalar, biyoloji ve genetik biliminden esinlenerek geliştirilmiş olan matematiksel çözüm modelleridir. Genetik ve biyolojide yer alan bazı tanımlamalar genetik algoritma tabanlı çözüm mekanizmalarında da kullanılmaktadır.

Çaprazlama işlemi yeni bireyler meydana gelmeden önce gerçekleşir. Çaprazlama işlemine çoğunlukla biyoloji de olduğu gibi literatürde ve genetik algoritma tabanlı matematiksel modellerde, rekombinasyon da denmektedir. İki ebeveynden gelen gametler, diploit kromozomları oluştururlar. Yeni oluşan bireyler mutasyona tabi olabilirler. DNA diziliminde meydana gelen değişikliklere mutasyon denmektedir. Biyolojide yer alan bu sistem genetik algoritma tabanlı matematiksel sistemler tarafından esinlenilerek, her bir kromozomun birer çözüm adayı olarak gösterilmiştir. Genler kromozom içinde yer alan yani çözüm kümesi adayları içinde yer alan kodlardır (bitlerdir). Genetik algoritmada ifade edilen nesil, tüm aday çözüm kümelerin içinde ifade edildiği topluluktur. Bu terim o anki toplam kromozom sayısı ile de ifade edilebilmektedir.

Genetik algoritmalar rastsal çözüm kümeleri oluşturan algoritmalar ve John Holland bu teoremin kurucularından kabul edilmektedir. Genetik algoritma iki ebeveyn kromozomdan üretilen yeni kromozomlar (yani çözüm kümeleri) üzerine yoğunlaşmıştır. Rastsal çözüm kümelerinden daha iyi çözüm kümelerine ulaşmak genetik algoritmada amaçlanmıştır. Hayatta karşılaştığımız ve matematiksel olarak çözümünün zor ve belirsiz olan birçok problemi çözmek için genetik algoritmalar kullanılmaktadır. Genetik algoritmada başlangıçta bir probleme ait başlangıç çözüm kümeleri önerilir ve önerilen çözüm kümelerine dayanarak yeni çözüm kümeleri oluşturulmaya çalışılır. Bu çözüm kümeleri, önerilen çözüm kümesinden seçilen kromozomların rekombinasyon yani çaprazlama ve mutasyona tabi tutulmasıyla yeni nesil oluşturulur. Genetik algoritmalar, diğer algoritmaların çözemediği durumlarda çözüm öneren algoritmalar olarak kabul edilmişlerdir. Bu algoritmalar daima daha iyi bireyler oluşturmayı amaçlayarak seleksiyon çaprazlama ve mutasyon işlemleriyle yeni bireyler oluştururlar. Genetik algoritma tabanlı sistemlerin genel basamakları aşağıdaki gibi özetlenebilir:

- 1) Mümkün çözümleri kromozom şeklinde kodlayan bir sistem

Kromozomlar, bir problem için muhtemel çözüm kümelerini temsil etmektedirler. Örneğin bir fonksiyon maksimizasyon probleminde kromozomlar, fonksiyon içindeki değişkenleri temsil etmekte ve her bir değişken değeri genlerde saklanmaktadır. Oluşturulan indirgeme yazılımında değişkenler, giriş değerlerini temsil etmektedir.

2) İlk çözüm kümesinin oluşturulması.

İlk çözüm kümesinin oluşturulması, genetik algoritma problemlerine göre değişiklik göstermekle birlikte, ilk çözümü temsil eden ve kromozomlardan oluşan topluluğu ifade etmektedir. Bununla birlikte ilk çözüm kümesi genellikle rastgele oluşturulur ve bu çözüm kümeleri çaprazlama ve mutasyondan oluşan rekombinasyon işlemleriyle yeni nesillerin oluşması sağlanır.

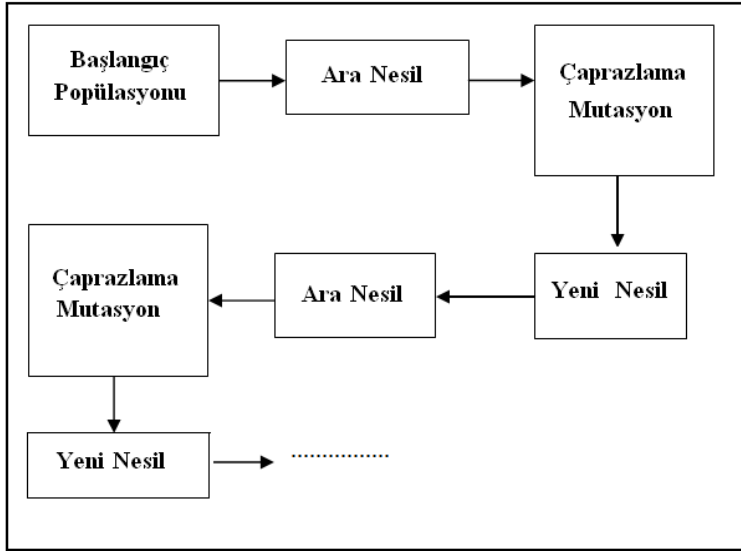
3) Kromozomların uygunluk değerlerini hesaplayan bir fonksiyon. Uygunluk değerlerini hesaplayan fonksiyon genetik algoritma kullanılarak çözülmesi gereken probleme göre değişmektedir. Geliştirilen indirgeme yazılımında uygunluk değeri, kromozomların özellik bağımlılığı değerleri hesaplanarak elde edilir.

4) Yeni aday çözümler oluşturmak için seçim mekanizması uygulanarak çaprazlama ve mutasyona uğrayacak bireyler belirlenir. Aynı zamanda seçim mekanizması, bazı yaklaşımlarda ara neslin oluşturulması için seçilir. Ara nesil, çaprazlama ve mutasyon işlemlerine tabi tutulur ve daha sonra rekombinasyon operatörünün uygulanacağı kromozomları seçmek için uygulanır.

5) Çaprazlama ve mutasyon operatörleriyle yeni nesil oluşturma ve durdurma kriterine ulaşmadıysa 3. adımdan devam eder.

6) Genetik algoritmadaki önerilen çözümlerden herhangi biri durdurma kriterine ulaşırsa, sistem durur. Eğer durdurma kriterine ulaşmazsa devam eder. Bu kriter genelde maksimum nesil sayısına bağlı olarak belirlenebileceği gibi, kullanıcı tarafından belirlenen bir eşik değerine göre de olabilir.

Klasik bir genetik algoritma sisteminin genel yapısı Şekil 3.4'deki gibidir.



Şekil 3.4 Genetik Algoritmanın Sisteminin Genel Yapısı (Ara Nesil Kullanan Vers.)

Genetik algoritma, genelde bir kromozom adı verilen bir çözüm kümesiyle başlar. Kromozomlar, çözümdeki adayları temsil ederler. Popülasyon ise kromozomların oluşturduğu topluluktur. Yeni bir popülasyon oluşturmak için mevcut popülasyondan seçilen bireyler çaprazlama ve mutasyon işlemlerine tabi tutulurlar. Yeni popülasyonun, önceki popülasyondan daha iyi olması amaçlanmaktadır. Yeni çözümü oluşturmak için üreme operatörüne yollanacak bireyler, kromozomların uygunluk değerleri dikkate alınarak yapılır. Genellikle en iyi çözüm bulunana kadar bu işlem devam etmektedir. Genetik algoritma temelli sistemler kullanıcı tarafından seçilen bir durdurma kriterine göre durdurulur. Minimizasyon ve maksimizasyonu problemleri genetik algoritma sistemlerinin çalışmasına örnek olarak gösterilebilir. Bir maksimizasyon probleminde genetik algoritma, her bir değişken için sayısal değerler üretir ve bu değerleri yerine yazarak fonksiyonun değerini hesaplar. Çaprazlama ve mutasyon işlemleriyle yeni nesiller üretilir. Genellikle algoritma belirtilen maksimum nesil sayısı ile ya da kromozomların istenen uygunluk değerlerine ulaşmış olmaları kontrol edilerek durdurulur. Genetik algoritmada çözüm kümelerinin oluşturulması için en önemli mekanizmalardan biri de kromozomların kodlama şeklidir. Kromozom kodlama çeşidi çözülecek probleme göre değişkenlik gösterir. Genetik algoritma, klasik matematiksel yöntemlerle çözülmesi zor olan bazı problemlerin çözümünde esnek hesaplama yöntemlerinde ve yapay zeka teknikleri içinde kullanılmaktadır.

3.15 Genetik Algoritma Temelli Sistemlerin Genel Yapısı ve Terminolojisi

Genetik algoritmalarda en çok kullanılan terimler, kromozom, gen, alel, genotip, fenotip, uygunluk, uygunluk fonksiyonu, şema, ebeveyn seçimi, çaprazlama ve mutasyon işlemleridir. DNA, Deoksiribonükleik asit'in kısaltılmış halidir ve canlılardaki biyolojik işlemlerin şifresini taşımaktadır. DNA, organizmalarda ve canlılarda, biyolojik işlemler için gerekli genetik materyali taşıyan birimdir. DNA'nın en önemli görevi bilginin uzun süreli saklanması, taşınması ve ilerideki jenerasyonlara iletilmesidir (Wikipedia 2015).

Genetik bilgileri içeren DNA'ya ait olan parçalara gen denilmektedir. İlk kez Miesher tarafından 1869 yılında keşfedilmiştir. İlk keşfedildiği zaman beyaz kan hücreleri üzerinde testler yapılmış ve asidik özelliği nedeniyle nükleik asit adı verilmiştir. Hücredeki iplikli yapıya ait olan bölgeye kromatin olarak adlandırılmıştır. Nükleotitler, DNA'nın temel yapı taşlarıdır ve yapısında fosfat, şeker ve organik baz bulunur. Nükleotitler içerdikleri organik Baza göre isimlendirilir. Bu bazlar, Adenin (A), Timin (T), Sitozin (C) ve Guanin (G)'dir. DNA'nın çok uzun bir polimer olduğu 1930'lu yıllarda anlaşılmıştır. Nükleotitler bir araya geldiklerinde Adenin karşısında Timin, Guanin karşısında ise Sitozin bulunur. DNA modellemesinde iki iplik bulunmaktadır. Hücrelerdeki DNA kromozom isimli yapının içinde yer almaktadır. Örneği ökaryot canlılar DNA'sını hücre çekirdeğinde bulundururken, prokaryotlar ise hücre sitoplazmasına yayılmış şekilde bulundurur. Hücre çekirdeği kromozomları, kromozomlar DNA'yı içermektedir. Aşağıda biyoloji ve genetikteki kromozomsal yapının matematiksel çözüme ve genetik algoritma sistemlerine aktarılması anlatılmıştır. Aşağıda genetik algoritma temelli sistemlerin temel terimleri hakkında bilgi verilmiştir.

Kromozom: Kromozom, çözüm kümesi içerisinde önerilen bir çözümü temsil etmektedir. Kromozomlar üreyerek yeni bireyleri (kromozomları) oluştururlar. Her bir kromozom genlerden oluşmaktadır ve genler ebeveynlerden aktarılır (Taşkın ve Emel, 2002).

Gen: Bir kromozom içindeki tek bir nitelik ya da özellik olarak ifade edilir. Klasik bir genetik algoritma sistemine örnek olarak gezgin satıcı problemini verebiliriz. Gezgin satıcı probleminde tüm şehirlere uğrayarak minimum mesafenin kat edilmesi

istenmektedir. Bu durumda uğranacak şehirlerin listesi kromozom olarak ifade edilir ve her bir şehir ise kromozom içinde yer alan birer genle kodlanır (Taşkın ve Emel, 2002).

Alel: Alel bir genin alabileceği değerleri göstermektedir. Saç renginin gende ifade edildiği düşünüldüğünde, aleller saç rengine ait değerlerdir. Bu durumda aleller, kahverengi, sarı, siyah ve kırmızıdır.

Genotip: Genotip biyolojide bir genin genetik yapısını sembolize etmektedir. Genlerin temsil edilişleri ve yapısı genotip ile ilgilidir.

Fenotip: Fenotip, genotipin fiziksel versiyonudur. Her bir genotipin karşılık geldiği fenotip değerleri vardır. Bu değerler biyolojik canlılarda dış görünümü ifade etmektedirler. Fenotip genellikle genlerle ifade edilmektedir. Fenotip, genetik (genotip) ve çevresel etkenlerin, dış görünüşte oluşturduğu tesirlerdir. Genellikle dış fiziksel yapı zaman içinde değişmektedir. Genetik algoritma ve matematiksel modellemelerde genotip, fenotip ve kromozom arasındaki ilişki aşağıdaki gibi ifade edilebilir (Cura, 2008). Örneğin genetik kodlaması ikili sayı sisteminde 1100111 ile kodlanan sayı sisteminin onluk sistemde karşılığı olan 103 değeri, sayının fenotipi olarak kabul edilir.

Uygunluk fonksiyonu: Uygunluk fonksiyonu, bir kromozomun ya da aday çözümünün popülasyon içerisinde ne kadar iyi bir değere sahip olduğunu gösteren fonksiyondur. Çözülecek probleme bağlı olarak uygunluk fonksiyonu değişmektedir. Örneğin bir fonksiyonun maksimum değerinin hesaplanacağı bir maksimizasyon probleminde uygunluk değeri fonksiyonun o kromozomda belirtilen değerlere göre hesaplanan değeri ya da bu değere orantılı bir büyüklük olarak hesaplanabilir. Genetik algoritmayla çözülecek probleme göre uygunluk değeri değişmektedir (Goldberg 1989).

Şema: Kromozomda yer alan genlerin ve hangi gende hangi değer bulunduğunu ile ilgilidir.

Çaprazlama ve Çaprazlama Oranı: Seçilen iki kromozomdan genlerin yeni oluşan bireylere aktarılmasıdır. Matematiksel olarak çaprazlama operatöründen sonra yeni bireyler oluşmaktadır. Çaprazlamanın farklı çeşitleri bulunmaktadır. Çaprazlama işlemine uğrayacak kromozomlar ara nesil kullanan yaklaşımda ara nesilden seçilir, çaprazlamaya uğrar ve yeni nesil oluşur. Çaprazlama oranı bir nesil arasından kaç kromozomun çaprazlamaya uğratılacağını göstermektedir. Örneğin, nesil 20 kromozomdan oluşuyorsa ve çaprazlama oranı %50 ise toplam 10 kromozom (5 çift)

çaprazlamaya girerek yeni kromozomları oluşturur. Diğer 5 çift çaprazlamaya girmeden aynen kalır. Farklı çaprazlama metotları bulunmakla birlikte hepsi yeni nesil ve çözüm kümelerine ulaşmayı amaçlamaktadır (Goldberg 1989).

Mutasyon ve Mutasyon Oranı: Kromozom içinde yer alan bir ya da birden fazla genin yer değiştirmesiyle ifade edilir. Mutasyon oranı bir nesil içinde yer alan kromozomların yüzde kaçının mutasyona uğratılacağını göstermektedir.

Aday Çözüm Kümelerinin Gösterilmesi : Tipik olarak, bit dizisi ya da karakter genelde çözüm kümelerini içermektedir. Bu fonksiyon önerilen çözümün ne derece iyi olduğunu gösteren bir değer geri döndürür. Her bir aday çözümü, çeşitli kodlama şekilleriyle gösterilir.

Kromozomların Yapısı ve Kodlama Çeşitleri : Her bir kromozom bir gen dizisinden oluşmuştur. Örneğin fonksiyon maksimizasyon ve minimizasyon problemlerinde, fonksiyonun maksimum ve minimum değerlerin aranması için her bir kromozom bir değişkenler paketinde yer alan değerleri sembolize ederken, kromozom içinde yer alan her bir gen ise her bir değişken için atanan değeri temsil eder. Çok bilinmeyenli matematiksel bir fonksiyonun maksimizasyonu probleminde fonksiyonun maksimum değerini hesaplamak için her bir değişkene ait olan reel sayı değerleri kromozoma ait olan genlerde ifade edilirken, kromozomda ise bu değerler toplu halde ifade edilir.

Gezgin satıcı problemi ve benzeri yaklaşımlarda ise kromozomlar rakamlar ya da harflerden oluşan bir yapıya sahiptirler. Bu problemin genetik algoritma ile çözümünde her bir harf ya da rakam bir şehri temsil eder.

Kromozomlar sayısal bir dizi olarak ifade edilebildiği gibi harflerle de ifade edilebilir. En çok kullanılan kromozomsal kodlama şekli binary (ikili) kodlama şeklidir. Kodlama, kullanılan problem çeşidine göre değişmektedir ve çeşitli şekilleri bulunmaktadır. Genetik algoritma temelli sistemlerde en çok kullanılan kodlama çeşidi binary kodlama çeşididir. Geliştirilen “GARSBS” indirgeme yazılımında permutasyon kodlama çeşidi kullanılmıştır. Yazılımda baskın girişlerin hesaplanması istendiğinden dolayı, her bir rakam bir giriş değerini temsil etmektedir.

Permutasyon kodlama çeşidi, literatürde gezgin satıcı vb. problemlerde kullanılan bir kodlama çeşididir. Gezgin satıcı probleminin çözümünde, her bir rakam bir şehri ifade etmektedir.

İkili Kodlama (Binary Kodlama) : Kodlama 0 ve 1 sayılarla lojik bitler şeklinde yapılan genetik algorithmada en çok kullanılan kodlama çeşidir. Birçok klasik genetik algoritma tabanlı sistem ikili kodlama sistemini kullanmaktadır. Her bir kromozom 0 ve 1'lerden oluşan bir dizi halinde gösterilmektedir. Aşağıda 2 kromozomun ikili kodlama şekli gösterilmiştir. İkili kodlama genetik algorithmada en çok kullanılan kodlama çeşitlerindendir. İkili kodlama geliştirilen indirgeme yazılımında kullanılmamıştır. İkili kodlama, Tablo 3.6'da gösterilmiştir (Gen ve Cheng, 2000).

Tablo 3.6 İkili Kodlamanın genel yapısı

1	Kromozom ₁	1 0 1 0 1 1 0 0 1 1 1 1 0 0 0 1
2	Kromozom ₂	1 0 0 1 1 1 0 0 1 0 1 0 1 1 1 0 0

İkili kodlamada aynı zamanda karakterlerin kodlanması da mümkündür. Geliştirilen indirgeme yazılımında, ikili kodlama yerine permutasyon kodlama kullanılmıştır. İkili kodlamaya döndürülen sayıların, daha sonra tekrar kodlarının çözülmesi gerekmektedir. Bu kodlama şeklinde, sayılar ya da harfler ikili sayı sistemiyle aşağıdaki gibi kodlanabilirler. Örneğin 0 dan 9'a kadar olan sayılar ikili kodlama kullanıldığında aşağıdaki gibi ifade edilebilir. Kromozomların ikili kodlamadaki durumları Tablo 3.7' de verilmiştir. Kodlama şekilleri genetik algorithma temelli sistemlerde çözülecek problemin şekline göre seçilmektedir. Genetik algorithmada çok çeşitli kodlama çeşitleri bulunmaktadır. Bu tür kodlama işlemi ayrıca decode edilmeye ihtiyaç duymaktadır. Bu yüzden indirgeme yazılımında permutasyon kodlama kullanılmış ve hafıza bu şekilde hem daha etkin bir şekilde kullanılmış hem de daha çok giriş sayısının desteklenmesi amaçlanmıştır. Bununla birlikte, çok girişli olmayan maksimizasyon, minimizasyon ve gezgin satıcı problemi gibi bazı uygulamalarda kullanılabilmektedir.

3.16 Permutasyon Kodlama ya da Sayısal Kodlama

Permutasyon kodlama her bir kromozomun birer sayı ile ifade edildiği durumlarda kullanılır. Permutasyon kodlamada her bir kromozom bir dizi sayı ile ifade edilir. Permutasyon kodlamaya örnek olarak Tablo 3.7'de yer alan kromozomları

gösterebiliriz. Geliştirilen genetik algoritma ve kaba küme temelli hibrit sistem yazılımında permutasyon kodlama kullanılmış olup, permutasyon kodlamada her bir sayı medikal bir giriş değerine ait olan sütununu yani nitelik (girişi) göstermektedir. Literatürde kodlama şekillerine farklı isimler verilmiştir. Örneğin gezgin satıcı problemi gibi sıralamanın kromozom içinde önemli olduğu durumlarda, Permutasyon kodlama şekline sıralı kodlama da denmektedir. Tablo 3.8’de permutasyon kodlama şekli gösterilmiştir (Gen ve Cheng, 2000).

Permutasyon kodlamada kromozom içinde çaprazlama ve mutasyon yapıldıktan sonra çözümlenmesi gereken problemin özelliğine bağlı olarak kromozom içinde aynı genin tekrar etmesinin engellenmesi gerekir. Geliştirilen indirgeme yazılımında aynı giriş değerlerinin tekrarı engellenmiştir.

Tablo 3.7 Genetik Algoritmada Kullanılan İkili Kodlama Sistemi Sayı Kodlanmış Şekli

1	0	0 0 0 0
2	1	0 0 0 1
3	2	0 0 1 0
4	3	0 0 1 1
5	4	0 1 0 0
6	2 4	0 0 1 0 - 0 1 0 0
7	3 3 5 7	0 0 1 1 - 0 0 1 1 - 0 1 0 1 - 0 1 1 1
8	2 2 4 9	0 0 1 0 - 0 0 1 0 - 0 1 0 0 - 1 0 0 1

Tablo 3.8. Permutasyon Kodlama Şekli

1	Kromozom ₁	2 4 9 8 7 1 5 12 6
2	Kromozom ₂	7 4 5 9 2 3 11 4 8
3	Kromozom ₃	8 6 8 11 14 18 5 3 16
4	Kromozom ₄	2 5 10 14 9 8 12 15 17

3.17 Değer Kodlama

Doğrudan değer kodlama birçok problemde kullanılabilir. Binary sayıların kullanılması da bu durumlarda bazen kullanışlı olmayabilir. Değer kodlama yönteminde bazen doğrudan reel sayılar kullanılabilir. (Gen ve Cheng, 2000)

Her bir kromozom değer kodlamada bir değer dizisi şeklinde gösterilebilir. Kullanılan değerler genetik algoritmayla çözülmesi gereken problemlerin cinsine göre değişmektedir. Genetik algortmada kullanılan değer kodlamaya ait kromozomlar aşağıda gösterilmiştir. Değerler çözülecek problemin cinsine bağlı olarak değişmektedir. Bunlara örnek olarak reel sayıları, karakter dizilerini ve bazı karmaşık değerleri gösterebiliriz. Tablo 3.9' da kromozomlar ve değer kodlama şekilleri gösterilmiştir.

Tablo 3.9 Kromozomlar ve Bazı Değer Kodlama Şekilleri

	Kromozom	Değer Kodlamadaki Şekli
1	Kromozom ₁	A B C K M N O S D H E
2	Kromozom ₂	1.49968 1.76553 2.66596 4.56235
3	Kromozom ₃	Kuzeye, Güneye, Güney Batıya, Kuzey Batıya (istikamet belirtildiği zaman)
4	Kromozom ₄	(Yukarıya), (Aşağıya), (Sağa), (Sola)

3. 18 Genetik Algoritmada Ağaç Kodlama

Ağaç kodlama, birçok genetik algoritma tabanlı programlama şekillerinde kullanılmaktadır. Bu tür kodlamada her bir kromozom bir ağaç yapısıyla ifade edilir. Ağaç kodlama şeklinde başlangıç düğümünden itibaren dallanarak ağacın uç noktalarına ulaşılabilir (www.datascienceearth.com, 2022).

Ağaç kodlamaya örnek olarak verilen değerlerden fonksiyonun bulunmasını gösterebilir. Ağaç dalları bu tür problemlerin çözümünde fonksiyondaki değişkenleri göstermektedir.

Genetik algoritma temel olarak 3 tip operatörden oluşmaktadır. Bunlar, seleksiyon (seçim), çaprazlama (crossover) ve mutasyondur. Seleksiyon genel olarak

popülasyondaki kromozomlardan üremeye (çaprazlamaya) girecek olanların seçimi ile ilgili bir terimdir. Bu operatör, genetik algoritma performansı üzerinde önemli bir mekanizmaya sahiptir. Genel olarak çaprazlama operatörü genetik algortmada iki adet ebeveyn kromozom seçer. Mutasyon aynı kromozom içinde bir ya da birden çok bitin yer değiştirmesiyle elde edilir.

3.19 Çaprazlama Operatörleri

3.19.1 Tek noktalı çaprazlama

Tek noktalı çaprazlamada her iki ebeveyn kromozomdan seçilen parçalarla yeni kromozom oluşturulur. Bu oluşturulan yeni çocuk kromozomlar her iki ebeveynin kromozomlarından parçalar taşımaktadır. Tek noktalı çaprazlama operatörü genetik algoritma temelli sistemlerde en çok kullanılan operatörlerden biridir. Çaprazlama için belirlenen kesik noktasının (ayrım) noktasının sağında ya da solunda yer alan genler ebeveyn kromozomlar arasında değiştirilerek iki yeni kromozom oluşturulur. Kromozomların çaprazlamadan önceki durumları aşağıda şekil 3.5 de gösterilmiştir. Sonraki durumları Şekil 3.6' de gösterilmiştir. (<https://user.engineering.uiowa.edu>, 2022).

1	2	3	4	9	10	7	8
3	4	5	6	7	11	12	15

Şekil 3.5 Kromozomların çaprazlama önceki durum

Kromozomların eşleştikten sonraki durumu

1 2 3 4 7 11 12 15
3 4 5 6 9 10 7 8

Şekil. 3.6 Kromozomların tek noktalı çaprazlamadan sonraki durumu

Bu çaprazlama operatörü kullanılarak çözülecek probleme bağlı olarak aynı kromozomun içinde yer alan genlerin tekrarının engellenmesi gerekebilir. Yapılan çalışmada tek noktalı çaprazlamada, permutasyon tip kodlama kullanılmıştır ve her bir rakam giriş değerlerini sembolize etmektedir.

3.19.2 İki noktalı çaprazlama

İki noktalı çaprazlama, genetik algoritma temelli sistemlerde temel çaprazlama türlerinden biridir. İki nokta arasındaki kalan kısımlar, iki yeni kromozom oluşturacak şekilde karşılıklı gen alışverişi yapılır. Genetik algorithmada en çok kullanılan operatörlerden biridir. Genellikle bu tür çaprazlamada 2 kesik noktası arasında kalan genler yer değiştirilir. İki noktalı çaprazlamadan önceki ve sonra kromozomların durumları aşağıda şekil 3.7 ve 3.8 de gösterilmiştir. Yapılan çalışmada iki noktalı çaprazlamada her bir gen giriş değerlerini temsil ettiği için genlerin tekrarı engellenmiştir. Her bir gen giriş değerlerini temsil ettiğinden, aynı giriş değerlerinin tekrarı, tekrar eden gen yerine o kromozom içinde kullanılmayan bir giriş değeri getirilerek engellenmiştir (Goldberg, 1989).

1	2	12	14	11	15
3	4	5	6	9	10

Şekil 3.7 Çaprazlama öncesindeki durum

Çaprazlama gerçekleştirildikten sonraki durum aşağıdaki gibidir.

1	2	5	6	11	15
3	4	12	14	9	10

Şekil 3.8 Çaprazlamadan Sonra Elde Edilen Kromozomlar

3.19.3 Sıraya dayalı çaprazlama operatörü

Sıraya Dayalı çaprazlamada (Order Based Crossover), bir grup nokta çaprazlama için seçilir. Seçilen noktalara karşılık gelen yerler birinci kromozomda yerlerini korur. İkinci Kromozomda ise seçilen noktasındaki karakterler birinci kromozomda anı noktanın yanına taşınırlar (Cheng ve Ark. 1999). Aynı durum ikinci ebeveyn kromozom için de geçerlidir. İkinci ebeveynin seçilen nokta karşısındaki yeri korunurken, ikinci kromozomda seçilen kromozomun yanına ilk kromozomdan eşleşen noktadaki kromozom taşınmaktadır. Bu tür çaprazlama

kromozomu oluşturan karakterlerin sayı ve sıralarının önem taşıdığı durumlarda kullanılır. Şekil 3.9 ve Şekil 3.10 da seçilen kromozomlar ve çaprazlama operatörünün uygulanması gösterilmiştir (Renner, 2003).

1	3	5	6	7	9	12
2	4	9	8	10	11	6

Şekil 3.9 Sıraya Dayalı Çaprazlama İçin Seçilen Kromozomlar

1	2	5	6	7	10	12
2	1	9	8	10	7	6

Şekil 3.10 Çaprazlamadan sonraki durum.

3.19.4 Kısmi planlı çaprazlama (Partially matched crossover)

Kısmi Planlı Çaprazlamada, indirgemedeki aday kromozomlar kısmi planlı çaprazlamada eşleşirler. Çaprazlama için seçilen 2 Aday kromozom içindeki karakterler aşağıdaki gibi çaprazlamaya girerler ve karakter alışverişi yaparlar. Oluşturulan yazılım çaprazlama sonrasındaki çakışan değerleri engellemektedir. Şekil 3.11, Şekil 3.12 ve Şekil 3.13 kısmi planlı çaprazlama anlatılmıştır. (Goldberg 1989).

5	7	9	1	8	12	15
6	3	10	11	4	2	17

Şekil 3.11 Kısmi planlı çaprazlama için seçilen kromozomlar

5	7	10	11	4	4	15
6	3	9	1	8	2	17

Şekil 3.12 Çaprazlamadan sonraki geçici durum

5 7 10 11 4 8 15
6 3 9 1 8 2 17

Şekil 3.13 Çaprazlamadan sonraki durum

3.19.5 Doğrusal Sıralı Çaprazlama

Doğrusal Sıralı Çaprazlama (LOX) ve bu çaprazlamanın adımları aşağıdaki gibidir. Bu çaprazlama metodu, Sıralı Çaprazlamanın (OX) in değiştirilmiş bir versiyonu olarak kabul edilir. İşlem adımları aşağıdaki gibidir. 2 noktalı çaprazlama operatörü vardır. (Renner 2003). Mevcut popülasyon içerisinde rastgele iki ebeveyn seçilir. (Cheng ve Ark 1999). Seçilen alt diziler karşılıklı yer değiştirilir. OX operatöründen farklı olarak sol ve sağ kısımlar değiştirilir

3.19.6 Genetik Algoritma Temelli Sistemlerde Kullanılan Diğer Çaprazlama Operatörleri

Eklemeli Çaprazlama

Bu tür çaprazlamada 1. ebeveyn kromozomdan kesilen kısım, ikinci ebeveyn kromozoma eklenerek sonuca ulaşılır. Çaprazlamadan önceki ebeveyn kromozomlar aşağıda gösterilmiştir. Çaprazlamadan önceki ve sonraki durum Şekil 3.14 ve Şekil 3.15’de verilmiştir. Bu çaprazlama yöntemi sabit uzunlukta kromozomlar üretilmek istendiğinden dolayı indirgeme yazılımında kullanılmamıştır.



Şekil. 3.14 Çaprazlamadan Önceki Durum



Şekil. 3.15 Çaprazlamadan Sonraki Durum

Uniform Çaprazlama

Bu teknikte, çaprazlama için belirli bir yer belirlenmez. Çaprazlama için seçilen iki ebeveyn kromozom için sırayla yeni kromozom oluşturulur. Seçilen iki adet ebeveyn kromozomdan tek çocuk kromozom oluşturulur. Uniform (Tekdüze), çaprazlamada rastgele değerlere göre ya birinci kromozomdan ya da ikinci kromozomdan gen alınır. Şekil 3.16 da üretilen kromozom görülmektedir. İki ebeveynden iki adet yeni kromozom oluşturulan versiyonları da vardır (Söke ve Bingöl 2004).

Ebeveyn Kromozom₁	1	3	5	7	9	12
Ebeveyn Kromozom₂	6	11	12	15	17	16
Yeni Kromozom	6	11	5	15	9	12

Şekil 3. 16 Tekdüze (Uniform) Çaprazlama

Genellikle eşik değer 0,5 seçilir. Eşik değerden büyük seçilirse, ikinci daha küçük ise birinci kromozomdan alınır. Uniform çaprazlamada sayılar gen düzeyinde üretilmektedir. Rastgele üretilen “0,6- 0,7- 0,4 - 0,65- 0,25 -0,1” değerlerine karşılık olarak üretilen kromozom aşağıdadır.

3. 20 Kullanılan Mutasyon Yöntemleri

Kullanılan Mutasyon Yöntemleri, Ters Mutasyon, Komşu iki karakter değiştirme, keyfi iki karakter değiştirme ve keyfi üç karakter değiştirme ve araya karakter ekleme mutasyon yöntemleridir. Mutasyonda genel amaç, seçilen karakter ya da karakterlerin kendi içinde yer değiştirmesidir. Mutasyonunun çaprazlama operatöründen en temel farkı, mutasyon, seçilen kromozom üzerinde yapılmaktadır.

Ters Mutasyon

Seçilen bölgedeki elemanların sıraları tersine çevrilerek gerçekleştirilir. Şekil 3.17 de gösterilmiştir (Schmitt, 1998). Soldan sağa doğru yazılan karakterler sağdan sola doğru yazılarak ters mutasyon gerçekleştirilmiş olur.

2 5 6 | 8 10 12 14 | 15 17
 2 5 6 | 14 12 10 8 | 15 17

Şekil 3.17 Ters Mutasyon

Komşu 2 Karakter (Giriş) Değiştirme

Komşu 2 karakter değiştirmede, 2 adet komşu seçilen bölgedeki karakterlerin değişimidir. Şekil 3.18’de komşu 2 giriş değiştirme gösterilmektedir. Komşu 2 karakter değiştirme, yazılımda komşu 2 giriş değiştirme şeklinde ifade edilir. Komşu 2 girişe ait değiştirilecek genler kromozom uzunluğu dikkate alınarak değiştirilir.

2 5 6 | 8 10 | 12 14 15 17
 2 5 6 | 10 8 | 12 14 15 17

Şekil 3.18 Komşu 2 giriş değiştirme

Keyfi İki Karakteri (Girişi) Değiştirme :

Kromozomda rastgele, (random) seçilen iki karakter (giriş) değiştirilebilir. Şekil 3.19’de verilen örnekte 1. karakter (1) ile 4. karakter (7) karşılıklı yer değiştirmiştir.

1 3 5 7 9 12 15
 7 3 5 1 9 12 15

Şekil 3.19. Keyfi iki karakter değiştirme mutasyon yöntemi

Keyfi iki karakter deęiřtirme ve keyfi üç karakter deęiřtirmede keyfi olarak seilen 2 ve 3 giriřin yerleri deęiřtirilmiř olur (Braaten ve ark 2000). Yapılan alıřmada ve karakter deęerleri yerine giriř deęerleri getirilmiřtir. Yazılımda indirgemed, kromozomlar giriř deęerlerini kodlamalarına raęmen, mutasyon uygulanmasının sebebi, mutasyona uęrayan bireyin aprazlama sonrasında yeni bireyler oluřturmasıdır.

Keyfi Ü Karakteri (Giriři) Deęiřtirme:

Kromozomda rastgele seilen üç karakter (giriř) yer deęiřtirilir. Ařaęıdaki řekil 3.20 de 3. Seilen karakter 1. Karakterin yerine, 1. Karakter 2. Karakterin yerine, 2. Karakter de 3. Karakterin yerine tařınmıřtır. 1. satırda mutasyona uęramamıř kromozom, 2. satırda ise mutasyona uęramıř kromozom gsterilmiřtir. Ařaęıda 9, 1'in yerine 1, 5'in yerine ve 5 te 9'un yerine yazılmıřtır.

1	3	5	7	9
9	3	1	7	5

řekil 3.20 Keyfi üç karakteri deęiřtirme mutasyon yntemi, (mutasyondan nceki ve sonraki durum)

Araya Karakter Ekleme Mutasyonu (Shift) :

Kromozomda seilen bir karakter kaydırılarak bařka bir pozisyona yerleřtirilir. řekil 3.21'de yer alan rnekte 4. Sırada yer alan karakter (7) , 2. karakterin yerine kaydırılmak iin seilmiřtir. 7, 2. karakterin yerine kaydırıldıktan sonra nceden 2. ve 3. karakterin yerinde bulunan 3 ve 5 rakamları birer pozisyon saęa kaydırılmıřtır.

1	3	5	7	9	10	15
1	7	3	5	9	10	15

řekil 3.21 Araya Karakter Ekleme Mutasyonu

3.21 Genetik Algoritma Temelli Sistemlerde Kullanılan Diğer Mutasyon Yöntemleri

Genetik Algoritmada kullanılan bazı mutasyon yöntemleri ikili kodlama şeklinde kullanılan mutasyonu gösterebiliriz. Bu mutasyon yönteminde, rastgele seçilen bir bit mutasyona uğratılmakta ve yeni kromozom oluşturulmaktadır. İkili kodlama şekli permutasyon kodlamadan farklı olarak sadece lojik 1 ve 0 değerlerinden oluşmaktadır.Örneğin aşağıdaki katarde rastgele seçilen bir bit değiştirilerek ikinci kromozom oluşturulur. Bu yöntemde değiştirilen bitin komplementeri alınır. Bu tür mutasyon komplementeri mutasyon adıyla anılmaktadır (Şekil 3.22).

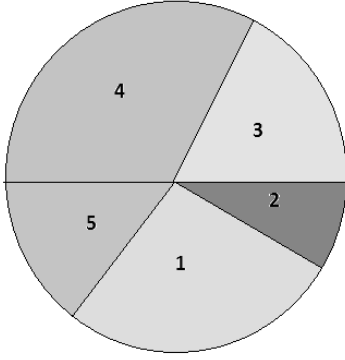
1	1	0	0	1	1	1	1	0
1	0	0	0	1	1	1	1	0

Şekil 3.22. Komplementeri Alan Mutasyondan Sonraki Durum

3.22 Genetik Algoritmada Kullanılan Seleksiyon Yöntemleri

3.22.1 Rulet Çemberi Seçim Sistemi

Genetik algoritma tabanlı sistemlerde yaygın olarak kullanılan bir seçim sistemidir. Seleksiyon yani seçim mekanizması her bir kromozomun kendi uygunluk değerine bağlıdır. Seleksiyon işlemi aynen rulette olduğu gibi dönen bir çemberle temsil edilir. Rulet çemberi seçim mekanizması genetik algoritmada en çok kullanılan seçim mekanizmalarından biridir. Bu seçim mekanizmasında iyi uygunluk değerine sahip olan kromozomların sonraki nesillere aktarılma ihtimalleri daha yüksektir. Her bir kromozoma rulet çemberi çarkında bir alan ayrılır. Uygunluk değeri daha fazla olan kromozomlara daha fazla alan ayrılır. Rulet çemberi seçim mekanizmasının avantajları olduğu gibi bazı dezavantajları da bulunmaktadır. Lokal (yerel) optimuma takılma problemi bu dezavantajlardan biridir. Rulet çemberi seçim mekanizması geliştirilen genetik algoritma ve kaba küme temelli özellik indirgeme yazılımında da kullanılmıştır. Rulet çemberi ve seçim mekanizmasının grafiksel gösterimi aşağıda verilmiştir. (Şekil 3.23).



Şekil 3.23 Rulet çemberi seçim mekanizmasında rulet çemberinin grafiksel gösterimi

Grafiksel gösterimden de görüleceği üzere uygunluk değeri fazla olan kromozomlara çember üzerinde daha fazla yer ayrılmıştır. Uygunluk değeri nispeten daha az olan kromozomlar (aday çözüm kümeleri) çember üzerinde daha az açıyla grafik üzerinde temsil edilmişlerdir. Rulet çemberi mekanizmasında rastgele üretilen değerlerin çember üzerindeki daha geniş olan bölgeye düşme ihtimali daha yüksek olduğu için bu seçim mekanizmasında daha iyi uygunluk değerine sahip olan kromozomların bir sonraki jenerasyona ulaşma ihtimalleri daha yüksektir. Fakat yine de seçim mekanizmasında rastgele değer üretilip (Rulet çemberi sanal olarak döndürülüyor) bu rastgele değerlerin hangi dilime denk geldiğine bakıldığından dolayı, en iyi çözümlerin bir sonraki jenerasyona aktarılma ihtimali yüksek olmasına rağmen her zaman garanti edilemez. Bu seçim mekanizmasının düşük popülasyon büyüklüklerinde lokal optimuma takılma problemi bulunmaktadır.

Genetik algoritma ve kaba küme kullanan medikal giriş değerlerini indirgeyen yazılımda, rulet çemberinde, gen havuzunda yer alan kromozomlar, özellik bağımlılığı değerlerine göre değerlendirilmişlerdir. Geliştirilen yazılımda özellik bağımlılığı fazla olan kromozomsal çözümler çözüm aralığında daha yüksek bir yüzdeyle ifade edilir, az olan çözümler ise çözüm uzayında daha düşük yüzdesel oranla ifade edilir. Rulet çemberinde en iyi çözüme sahip olan kromozomlardan daha iyi değerlere sahip olan kromozomlara ulaşılması amaçlanmıştır. Özellik bağımlılığı yüksek olan kromozomlar yüksek bir yüzdesel oranla ifade edileceğinden dolayı rastgele seçim işleminde seçilme ihtimalleri de o derece yüksek olmaktadır.

Rulet çemberinde seçim için iki temel metod vardır. Çaprazlama ve mutasyon işlemlerinin uygulanacağı bireylerin seçimi ara nesile seçilecek kromozomlar popülasyondaki kromozom sayısı kadar 0 ile 1 arasında tekrar tekrar random (rastgele) sayı üretilmesiyle belirlenir. Her bir kromozom (yani aday çözüm) uygunluk değerlerine

bağlı olarak belirli bir açı ile ifade edilir. İkinci bir yol olarak, rulet çemberi yönteminde, kromozomların uygunluk değerleri art arda eklenir. Yazılımsal olarak sıfırdan uygunluk değerinin toplamına kadar random sayı üretilir. Random üretilen sayının hangi aralığa düştüğü kontrol edilerek o kromozom ara nesile seçilmiş olur. Rulet Çemberi Mekanizmasının en büyük dezavantajı yerel optimum noktalara takılma problemidir. Yerel optimuma takılma problemi 2 adet yeni modifiyesuni seçim mekanizmaları ile yerel optimuma takılma problemi çözülmüştür.

Rulet çemberinin ara seçim mekanizması kullanan versiyonunun çalışma prensibi aşağıda gösterilmiştir. Uygunluk değerinin daha yüksek olduğu durumda seçilme ihtimalinin yüksek olduğu durumda Rulet Çemberinin çalışma prensibi aşağıdaki gibidir.

Uygunluk değerleri kromozomlarla doğru orantılı olan genetik algoritma temelli sistemlerde rulet çemberi seçim mekanizması çalıştırılırken rastgele olarak üretilen sayı hangi aralık değerine denk geliyorsa o kromozomların seçilmesi sağlanır.

Kromozomlar ve Uygunluk Değerleri

<i>Kromozom₁</i>	0,775.....	$0 \leq x < 0,775$
<i>Kromozom₂</i>	0,885.....	$0,775 \leq x < 1,66$
<i>Kromozom₃</i>	0,500.....	$1,66 \leq x < 2,16$
<i>Kromozom₄</i>	0,455.....	$2,16 \leq x < 2,615$
<i>Kromozom₅</i>	0,555.....	$2,615 \leq x < 3,17$
<i>Kromozom₆</i>	0,666.....	$3,17 \leq x \leq 3,836$

0 ile 3,836 arasında üretilcek random sayının uygunluk değeri daha yüksek olan kromozoma gelme ihtimali daha yüksek olduğundan dolayı bu kromozomların seçilme ihtimali daha yüksektir. İkinci bir yöntem olarak da rulet çemberi mekanizmasında 0 ile 1 arasında rastgele sayı üretilir. Üretilen sayının geldiği yüzdelik dilime göre kromozom seçilmiş olur. 6 kromozomdan oluşan neslin uygunluk değerleri toplamına göre aralıklar belirlenir. Böylece uygunluk değerleri toplamı ve aralıklar 0 ile 1 arasında normalizasyona uğratılmış olur. Aşağıda kromozomlar ve yüzdelik dilimleri verilmiştir.

3,836. $k = 1,$

$k = 0, 2606$

Kromozomlar, Uygunluk Değerleri ve Rulet Çemberindeki Yüzdelik Bölgeleri

$Kromozom_1$	0,775.....	$0 \leq x < 0,2019$	$\%0 \leq x < \%20,19$
$Kromozom_2$	0,885.....	$0,2019 \leq x < 0,4325$	$\%20,19 \leq x < \%43,25$
$Kromozom_3$	0,500.....	$0,4325 \leq x < 0,5628$	$\%43,25 \leq x < \%56,28$
$Kromozom_4$	0,455.....	$0,5628 \leq x < 0,6814$	$\%56,28 \leq x < \%68,14$
$Kromozom_5$	0,555.....	$0,6814 \leq x < 0,8261$	$\%68,14 \leq x < \%82,16$
$Kromozom_6$	0,666.....	$0,8261 \leq x \leq 1,0$	$\%82,16 \leq x \leq \%100$

Örneğin 0 ile 1 arasında random olarak 0.3, 0.5 ve 0,6 sayıları üretildiği zaman sırasıyla 2. , 3. ve 4. kromozomlar seçilmiş olur.

3.22.2 Suni seçim Mekanizması

Literatürde çeşitli seçim mekanizmaları vardır. Seçim mekanizmaları farklı problemler için farklı seçim mekanizmaları uygulanabilir. (Gen, Cheng 2000). Bu seçim mekanizmasında son iki nesilden elde edilen kromozomlar bir havuzda toplanırlar ve uygunluk değerlerine göre sıralanırlar. En iyi ve en kötü uygunluk değerine sahip olan kromozomlardan belirlenen oranlarda alınır. Bu yöntemde kromozomlar uygunluk değerlerine göre sıralanırlar.

3.22.3 Rank Temelli Seçim Mekanizması

Rulet Çemberi seçim mekanizmasının özel bir şekli olarak Rank temelli seçim mekanizması görülebilir. Bu seçim mekanizmasında kromozomlar uygunluk değerlerine göre sıralanırlar. Uygunluk değeri en düşük olandan en yüksek olana doğru, 1' den popülasyon sayısı olan N ' ye kadar kromozomlara katsayılar atanır. Hazırlanan yazılımda uygunluk değeri eşit olan kromozomlara eşit uygunluk değerleri verilmiştir. Bu yüzden en yüksek uygunluk değeri N olmayabilir. (Razali ve Geraghty 2011).

İki kromozomun bağımlılık değeri aynı olduğu zaman, en yüksek uygunluk değeri $N-1$ olmaktadır. Örneğin iki kromozomun uygunluk değeri aynı olduğu zaman onlara aynı uygunluk değeri verilmektedir. Aşağıdaki örnekte rank temelli seçim mekanizması sayısal bir örnekle anlatılmıştır. Aşağıdaki örnekte çeşitli kromozomlar ve bu kromozomlara ait olan uygunluk değerleri gösterilmiştir. Tablo 3.10 'de olduğu gibi kromozomların sıra değerlerine göre azdan çoğa doğru artan rank değerleri belirlenir ve

aralıklar ona göre belirlenir. Jenerasyonda 8 adet kromozom olduğu zaman $0 \leq x \leq 32$ aralığında 8 adet sayı üretilir. Üretilen random sayıların aralık değerlerine göre sayılar seçilmiş olur. Örneğin rastgele üretilen sayılar olarak 0,3, 2, 22, 25, 30, 5, 12, 5 üretildiği zaman sırasıyla Kromozom 5, Kromozom1, Kromozom 4, Kromozom 4, Kromozom 6, Kromozom 3, Kromozom7 ve Kromozom 3 seçilir. Sonraki jenerasyonlar da aynı şekilde oluşturulur. Rank temelli seçim mekanizmasında yerel optimum noktalara takılma problemi rulet çemberinden daha az olmakla birlikte rulet çemberi seçim mekanizmasının seçme işlemi değiştirilmiş bir varyantıdır.

Tablo 3. 10. Rank Temelli Seçim Mekanizması ve kromozomların değerleri

		Uygunluk Değeri	Rank	Sınır Değerleri
1	Kromozom 5	0,2	1	$0 \leq x < 1$
2	Kromozom 1	0,3	2	$1 \leq x < 3$
3	Kromozom 3	0,4	3	$3 \leq x < 6$
4	Kromozom 2	0,5	4	$6 \leq x < 10$
5	Kromozom 7	0,5	5	$10 \leq x < 15$
6	Kromozom 8	0,5	5	$15 \leq x < 20$
7	Kromozom 4	0,8	6	$20 \leq x < 26$
8	Kromozom 6	0,8	6	$26 \leq x \leq 32$

3.22.4 Stokastik Evrensel Örnekleme Yöntemi

Stochastic Evrensel örnekleme (Stochastic Universal Sampling) yöntemi, tek fazlı bir örnekleme algoritmasıdır. Kromozom seçiminde, kromozomlara uygunluk değerlerine göre Rulet çemberinde olduğu gibi yerler tahsil edilir. Başlangıç noktası 0 ile belirlenen öteleme noktası arasından rastgele seçilir. Stokastik universal örnekleme yöntemi aşağıda verilen örnekte açıklanmıştır. Her bir kromozoma yani aday çözüm kümesine uygunluk değeriyle orantılı olarak yer tahsil edilir.

Kromozomlar uygunluk değerleri aşağıdaki gibi olmaktadır. Stokastik universal örneklemede, aralık değerleri kromozomların uygunluk değerleri ile orantılıdır. Kromozomların uygunluk değerleri toplamı, popülasyondaki kromozom sayısına bölünerek her bir kromozom arasındaki mesafe bulunur. Stokastik evrensel örnekleme yöntemi aşağıda örnek üzerinde açıklanmıştır. (<https://en.wikipedia.org>, 2022).

Kromozomlar ve kromozomlara ait uygunluk değerleri, Tablo 3.11’de verilmiştir. Denklem 3.31’ de seçim noktaları arasındaki aralık değerin hesaplanması gösterilmiştir.

Tablo 3.11 Stokastik Üniversal Örnekleme Yönteminde Kromozomlar ve Uygunluk

No	Kromozom No	Kromozomların Uygunluk Değeri	Kromozoma Ayrılan Alan
1	Kromozom 4	0,8	$0 \leq x < 0,8$
2	Kromozom 6	0,8	$0,8 \leq x < 1,6$
3	Kromozom 2	0,5	$1,6 \leq x < 2,1$
4	Kromozom 7	0,5	$2,1 \leq x < 2,6$
5	Kromozom 8	0,5	$2,6 \leq x < 3,1$
6	Kromozom 3	0,4	$3,1 \leq x < 3,5$
7	Kromozom 1	0,3	$3,5 \leq x < 3,8$
8	Kromozom 5	0,2	$3,8 \leq x < 4,0$

Yukarıdaki kromozomların uygunluk değerleri toplamı = 4

$$\text{Seçim Noktaları Arasındaki Ar. Değ} = \frac{\text{Kromozomların Toplam Uyg. Değ}}{\text{Nesildeki Toplam Kromozom Sayısı}} \dots (3.31)$$

$$\text{Seçim noktaları arasındaki aralık değeri} = \frac{4}{8} = 0,5 \text{ olarak hesaplanır.}$$

Başlangıç noktası rastgele (random olarak) 0 ile belirlenen aralık değeri arasında seçilir. Örneğin başlangıç noktası rastgele olarak 0.2 olarak belirlendiği zaman 0.2 den sonra sırayla 0.2, 0.7, 1.2, 1.7, 2.2, 2.7, 3.2 ve 3.7 ye denk gelen noktalardan seçilir. Kromozom aralıkları rulet çemberinde olduğu gibi uygunluk değerleriyle orantılıdır. Kromozom 4, Kromozom 4, Kromozom 6, Kromozom 2, Kromozom 7, Kromozom 8, Kromozom 3, Kromozom 1 seçilir. Bir sonraki jenerasyonda benzer şekilde üretilir.

3.22.5 Genetik Algoritma Temelli Sistemlerde Kullanılan Turnuva Tipi Seçim Yöntemi

Turnuva tipi seçim yönteminde, bireylerin popülasyonundan bir birey seçmenin yöntemidir. Turnuva seçim yöntemi popülasyondan seçilen bireyler için turnuva düzenler. Her bir turnuvanın kazanı çaprazlama için seçilmektedir (wikipedia, 2020).

3.23 Genetik Algoritma Kullanmayan ve Alt Kümeleri Hesaplayan Versiyon

Geliştirilen sistemi test etmek için genetik algoritma kullanmayan ve tüm alt kümeleri tek tek hesaplayarak girilen eşik özellik bağımlılık değerinin üzerindeki değerleri indirgenmiş özellik kümesi olarak kabul edilen bir test yazılımı da oluşturulmuştur. Tüm alt kümeleri hesaplayan versiyonunun çok giriş sayısını desteklememekte ve hafıza kullanım ihtiyacı genetik algoritma temelli sisteme göre daha fazla olmaktadır. Eşik özellik bağımlılığı değeri hem değişik veritabanlarına entegrasyonu sağlamak hem de işlem zamanını sınırlandırabilmek için kullanıcı tarafından değiştirilebilmektedir.

3. 24 Diğer Bazı İndirgeme Yaklaşımları

3.24.1 Rölatif (Göreceli) Bağımlılık Metodu

Bu yaklaşımda, karara bağlı ayırt edici matris ya da ayırtedicilik fonksiyonu kullanılmamaktadır. Bununla birlikte, indirgeme için Göreceli Bağımlılık kullanılmaktadır. Bununla birlikte göreceli bağımlılık metodu, aşağıdaki denklemle ifade edilir. Rölatif göreceli bağımlılık metodu çok giriş sayılı sistemleri desteklememektedir. Göreceli bağımlılık aşağıdaki formüle göre ifade edilir. Bu yaklaşımda iki algoritma vardır. Bunlardan ilki, özelliklerin teker teker elimine edilmesini sağlamaktadır. Eğer göreceli bağımlılık 1 olduğu sürece özellikler elimine edilir. Göreceli bağımlılık metodunun denklemi aşağıdaki gibidir. (Denklem 3.32). Göreceli bağımlılık metodunda pay ve paydadaki alt küme eleman sayılarının aynı olmasına bakılmaktadır (Han ve Ark 2005).

$$\kappa_R(D) = \frac{|U / IND(R)|}{|U / IND(R \cup D)|} \dots\dots\dots (3.32)$$

Bu yaklaşımda eşdeğerlik kümelerinin sayısına bakılmakta, oran eşit olduğu zaman alt dallara dallanmaktadır. Yukarıdaki denklemde “D” karar değişkenini göstermektedir. Denklemde eşdeğerlik kümelerinin kardinalitelerine bakılmakta ve pay ve payda arasındaki oran 1 olduğu zaman giriş niteliklerinden biri azaltılarak aynı göreceli bağımlılık değerine ulaşıp ulaşılmadığına bakılmaktadır. Değişkenler tek tek elimine edilip çıkarılmaya çalışılmaktadır.

3.24.2 Hızlı indirgenmiş özellik ve (ters) indirgenmiş küme yaklaşımları

Hızlı İndirgenmiş Özellik kümesi ve Ters İndirgenmiş özellik yaklaşımlarını şu şekilde özetleyebiliriz. Acele indirgeme algoritması boş kümeyle başlar ve her bir defasında kaba küme özellik bağımlılığında en büyük değişimi yapan özelliği eklemeye çalışır. Bu algoritmaya göre her bir özelliğin bağımlılığı hesaplanır ve en iyi aday seçilir. Böylece düğüm düğüm ilerleyerek indirgenmiş özellik kümesi bulunmaya çalışılır. Algoritma az giriş sayısı bulunan sistemlere uygun bir algoritmadır ve fazla giriş sayısına sahip olan sistemler için uygun değildir. Diğer bir algoritma da ters reduct algoritmasıdır. Ters indirgenme algoritmasında, ileriye doğru giden sistemin tersine geriye doğru özellik değerlerinin eliminasyonuna dayanmaktadır. Bu indirgeme yöntemi büyük veri tabanları ve çok girişli sistemlere uygun değildir. En az önemli olan özellik değerlerinin sırasıyla elimine edilmesiyle çalışmaktadır.

3.24.3 Johnson İndirgeyici

Bu algoritma minimal büyüklükte indirgenmiş özellik kümelerini bulmayı amaçlamaktadır. “B” ismi verilen verilen indirgenmiş özellik kümesi aşağıdaki adımlar kullanılarak bulunur (Johnson, 1974). Burada S, ayırt edinilirlik fonksiyonu, w(S) ise “S” için gösterilen ağırlıkları ifade eder. Yaklaşık sonuçlar için destek oranı S’den yeterli sayıda küme ayrıldığı zaman aşağıdaki gibi olur (A. Øhrn, 1998).

$$1.B = \emptyset$$

2. $a, \sum w(s)$ ' yi maksimize eden özellik değerini gösterir. (Toplam S içinde a'yı içeren tüm kümeler üzerinde yapılır).
3. a yı B' 'ye ekle
4. S den a 'yı içeren tüm S kümelerini çıkar
5. Eğer $S = \emptyset$ ise, B' 'yi döndür, aksi halde adım 2'ye git.

3.25 Yapay Sinir Ağı (YSA) ve YSA Tabanlı Sınıflandırıcı

Geliştirilen kaba küme ve genetik algoritma temelli indirgeme yazılımına test amacıyla değişken girişli Yapay Sinir Ağı (YSA) tabanlı sınıflandırıcı eklenmiştir. YSA tabanlı sınıflandırıcı ve test yazılımı Delphi7 programlama dili kullanılarak oluşturulmuştur. Genetik algoritma ve kaba küme kullanan indirgeme yazılımında hesaplanan indirgenmiş özellik kümeleri (reduct) değerleri test amacıyla YSA tabanlı, gizli nöron sayısı kullanıcı tarafından değiştirilebilen yazılıma sokulurlar.

Değişken girişli Yapay Sinir Ağları temelli sınıflandırıcıda gizli nöron sayısı kullanıcı tarafından değiştirilebilmektedir. Yapay Sinir Ağları (YSA) Tabanlı test mekanizmasında klasifikasyon yüzdesi belirlenen sütunlara göre hesaplanmasını sağlar. Hazırlanan yazılım belirlenen hata oranına göre durdurulabildiği gibi aynı zamanda maksimum epok sayısına göre de durdurulabilir. YSA sınıflandırıcıda hem belirlenen giriş değerlerine göre eğitim yapılmaktadır hem de sınıflandırma doğruluğu hesaplanmaktadır. Yazılım ara katmandaki nöron sayısı değişken olacak şekilde tasarlanmıştır. Ara katman nöron sayısındaki değişikliklerin yapay sinir ağları temelli test sistemlerine çeşitli etkileri bulunmaktadır. Ara katman nöron sayısındaki değişiklik YSA temelli sistemlerde özellikle öğrenme üzerine çeşitli etkilere sahiptir.

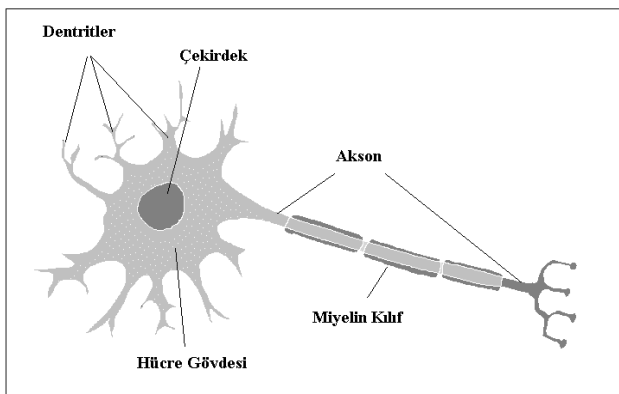
Yapay Sinir Ağları temelli sistemlerde, ara katman nöron sayısının aşırı artması, ezberlemeyi arttırmakta, aşırı azaltılması ise öğrenme ve epoksal hata oranlarını arttırmaktadır. Bu yüzden gizli (ara katman) nöron sayısının değiştirilebilir olması önem taşımaktadır. Çıkış karar değişkenidir ve tekdir.

3.26 Yapay Sinir Ağları Temelli Sistemlere Genel Bakış

Yapay sinir ağları temelli sistemler, nörolojik sistemlerdeki nöronlar arasındaki bağlantılar ve bağlantılar arasındaki bağlantılardan faydalanılarak sinir ağı sistemi kurulmuştur. Yapay sinir ağları temelli sistemler, sinir ağlarının arasındaki bağlantıların

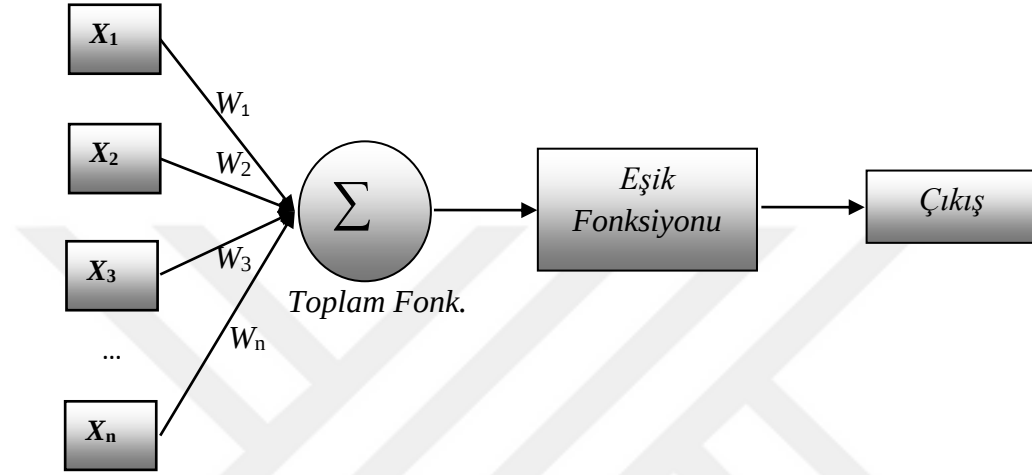
ve bağlantı kuvvetlerinin matematiksel bir düzene adapte edilmesiyle oluşturulmuş sistemlerdir. Yapay Sinir Ağı temelli çalışmalarda iki sinir hücresi ve hücrelerin arasındaki impulsun nasıl iletildiği incelenmiş ve teori bu ilişkiler üzerine oturtulmuştur. Sinir ağları temelli algoritmaların başarı oranları, matematiksel modellemelerin geliştirilmesiyle günden güne artmıştır.

YSA, insan beyninin bilgileri değerlendirme ve işleme sisteminden esinlenerek geliştirilmiş matematiksel modellemeler bütünüdür. Nöronlar arasındaki bağlantıların matematiksel modellemeye dönüştürülmesiyle elde edilmiştir. YSA temelli öğrenme algoritmaları, insanın düşünme sistemini sinir ağlarına taşımaya çalışmaktadır. (Öztemel 2006, Şen 2004). Biyolojik sistemlerde öğrenme, nöronlar arasındaki bağlantılardan ilham alınarak matematiksel modellemelere taşınmıştır. Yapay sinir ağları temelli yapay zeka uygulamalarında sinir hücreleri arasındaki bağlantılar ve arasındaki ağırlıkların ayarlanmasına dayanan matematiksel modellemeler kurulmuştur. Bağlantı ağırlıklarının tekrar tekrar değiştirilmesiyle eğitim gerçekleştirilmiş olur. Yapay sinir ağlarında çok sayıda işlem birimi aynı anda devreye girebilmekte ve istenen hata değeri elde edilene kadar iterasyonlar devam etmektedir. Yapay Sinir Ağı temelli sınıflandırıcıda genellikle geri yayılım metoduyla daha yüksek performanslı sonuçlar elde edilmektedir. Yapay sinir ağı temelli sistemlerde ağda veri alan ve çıktı üreten sinir hücreleri mevcuttur. YSA ve YSA temelli sınıflandırıcıda biyolojik sistemlerden esinlenerek oluşturulmuştur. Şekil 3.24’de bir sinir hücresine ait olan bazı elemanlar gösterilmiştir. Aşağıda biyolojik bir sinir hücresine ait olan temel parçalar gösterilmiştir.



Şekil 3.24. Biyolojik Nörona ait olan temel yapı

Soma, Akson, Sinapsis, sinir hücresinin temel yapıları olarak kabul edilmektedir. Yapay sinir ağlarının matematiksel modellemelerinde, biyolojik sinir hücreleri ve bu hücrelerin çalışma ilkeleri temel alınmıştır. Sinir hücresi arasındaki bağıntılar, yapay sinir ağlarındaki ağırlıkların modellemesiyle açıklanır. Aşağıda bir tek katmanlı algılayıcı modeli gösterilmiştir. Bir Basit Algılayıcının yapısı Şekil 3.25’de gösterilmiştir.



Şekil 3.25 Bir Perseptron Hücresinin Genel Yapısı

YSA temelli sistemlerde, ağırlıkların güncellenmesiyle sonuca ulaşılmaktadır. Hesaplanan ağırlık değerlerine elde edilen çıktılar, tablodaki çıktılar hesaplanan ağırlık değerleriyle karşılaştırılır. Öğrenme YSA temelli sistemlerde ağırlıkların güncellenmesi ve güncellenen ağırlık değerlerine göre üretilen çıktıların eğitim kümesindeki beklenen çıktı değerleriyle karşılaştırılmasıyla yapılmaktadır.

Ağırlıkların ayarlanmasıyla ve güncellenmesi YSA temelli sistemlerde öğrenme olarak tanımlanır. Ağın ürettiği çıktı ile eğitim kümesinde ağdan beklenen çıktının karşılaştırılır.

Yapay sinir ağları, biyolojik sistemlerdeki nöronlar ve nöronlar arasındaki bağlantılardan esinlenerek ve bu sistemin matematiksel modellemeye taşınmasından oluşturulmuştur. Bu tür sistemlere örnek olarak yüz tanıma sistemlerini, yazı tanımlama sistemlerini ve medikal alanda uygulanan birçok sınıflandırma sistemini sayabiliriz. Sinir ağı temelli sistemlere örnek olarak çok katmanlı algılayıcıları verebiliriz.

Perseptron modelinde, sinir hücrelerine gelen sinyallerin toplamı ancak belirli bir eşik değerinden daha yüksek olması durumunda çıkış 1 olmaktadır. Eğer sinyallerin

toplamı, belirlenen bir eşik değerden daha düşükse o zaman nöron çıkışı sıfır yapılır. Bu yaklaşımda ağırlık beklenen çıkış değerini üretememesi durumunda ağırlıklar güncellenir. (Şen, 2004)

Yapay sinir ağları temelli sistemlerde, durdurma kriteri olarak maksimum epok sayısı veya hata oranı kullanılabilir. Ağırlıklar, ağırlık istenen çıkışı üretilmemesine göre güncellenmektedir. (Öztemel, 2006). Perseptrona giriş net değeri Denklem 3.33’de gösterilmiştir. (Anthony ve Barlett 2009).

Perseptron yapay sinir ağı temelli sistemlerde, temel bir eğitim birimi olarak kabul edilmektedir ve çıkışı ise belirlenen bir eşik değeri ile karşılaştırılarak denklem 3.34 ’de ifade edilmiştir.

$$NET = \sum_{l=1}^M w_l \cdot x_l \dots\dots\dots(3.33)$$

$$\text{Çıkış} = \begin{cases} \text{Eğer } NET > \varphi \dots\dots\dots 1 \\ \text{Eğer } NET \leq \varphi \dots\dots\dots 0 \end{cases} \dots\dots(3.34)$$

Bununla birlikte diğer yaklaşımların geliştirilmesine yardımcı olduğundan dolayı danışmanlı öğrenme temelli sistemlerde önemli bir yer tutmaktadır. Öğrenme temelli sistemlerde ağırlıkların güncellenmesi vb. konularında perseptron önemli yer tutmaktadır. Yapay sinir ağlarında kullanılan temel fonksiyonlar aşağıda ifade edilmiştir. Bu fonksiyonların içinde Hard Limiter Fonksiyonu, Eşik Fonksiyonu ve Sigmoid Fonksiyonu verilebilir (E. Oztemel 2008).

- a. Hard Limiter fonksiyonu : Bu fonksiyonda ayrık sonuçlar elde edilir. Tek katmanlı algılayıcıda da bu fonksiyon kullanılmaktadır.
- b. Eşik fonksiyonu: Yapay sinir ağlarında kullanılan bazı eşik fonksiyonları bulunmaktadır. Eşik fonksiyonunun en önemli çeşitleri arasında basamak fonksiyonunun farklı şekilleri sayılabilir. $-T$ ve T aralığında lineer olan ve bu aralıklar dışında sabit değerlere sahip olan çeşitleri de bulunmaktadır. Eşik fonksiyonunun denklemleri aşağıdaki gibi ifade edilebilmektedir. (Denklem 3.35-3.36) Basamak eşik fonksiyonu aşağıdaki gibidir.

$$f(x) = \begin{cases} +T & \text{eğer } x \geq T \text{ ise} \\ x & \text{eğer } |x| < T \text{ ise} \\ -T & \text{eğer } x \leq -T \text{ ise} \end{cases} \dots\dots\dots(3.35)$$

$$f(x) = \begin{cases} +k, & \text{eğer } x > 0 \text{ ise} \\ -b; & \text{diğer durumlarda} \end{cases} \dots\dots\dots(3.36)$$

c. Sigmoid fonksiyonu: Sigmoid fonksiyonu özellikle geriye yayılım vb. yapay sinir ağı temelli yaklaşımlarda oldukça sık kullanılan bir etkinleştirme fonksiyonudur. Türevi alınabilen bir fonksiyon olduğundan birçok geriye yayılım tabanlı YSA sistemlerinde kullanılmaktadır. Sigmoid fonksiyonu yerine yine süreklilik arz eden benzeri fonksiyonlar da kullanılabilir. Ağırlıklar Denklem 3.37 deki denklem ile güncellenir.

$$w_{i,yeni}(t+1) = w_{i,eski}(t) + (\mu * [d(t) - f(y_{in})] * x_i(t)) \dots\dots(3.37)$$

3.37’ deki Bu formül yardımıyla, yapay sinir hücresinin öğrenmek için isleme koyduğu güncelleme süreci açıklanmış olmaktadır (Fausett L., 1994). Öğretmenli öğrenmede, yapay sinir ağıının dışarıdan eğitilmesi durumu söz konusudur. Bu tip öğrenme yönteminde girilen değerlere göre çıkıştan hangi değerlerin elde edileceği bilinmekte ve yapay sinir ağıındaki ağırlıklar buna göre güncellenmektedir. Diğer bir öğrenme yöntemi ise danışmansız öğrenme olarak kabul edilir. Yapay sinir ağıları genellikle bu iki öğrenme metodundan birini kullanır. Güncellemede kullanılan formül (3.37)’de yer alan “μ” ile sembolize edilen katsayı öğrenme katsayısıdır ve öğrenme hızını etkiler. Yapay sinir ağılarının öğrenme algoritmalarında genelde belli bir iterasyon (döngü sayısı) geçtikten sonra, “m” değeri azaltılır. Bunun gerçekleştirilmesi için “m” değeri belli bir sabit katsayı ile çarpılmaktadır (Denklem 3.38) (Kung S.Y.,1993).

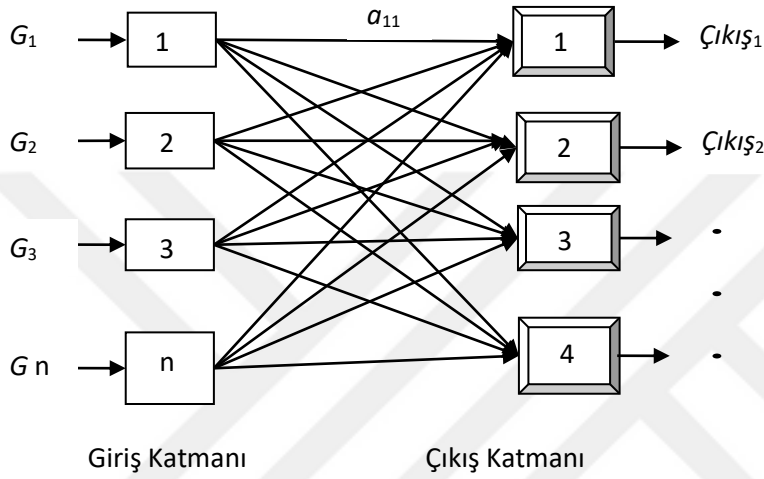
$$m = m * \text{sabit} \dots\dots\dots (3.38)$$

Kullanılacak eğitim setinin büyüklüğü ağı üzerinden elde edilecek sonucun başarısı üzerine etkili olabilir.

İki tabakalı YSA modeli aşağıdaki gibi ifade edilmektedir. (Şekil 3.32) İki tabakalı YSA modelinde giriş ve çıkış katmanları bulunurken ara katmanlar gizli

katman) bulunmaz. Aşağıdaki şekilde girişler G_n harfiyle gösterilmektedir. a_n elemanları ağırlık değerlerini göstermektedir. (Şen, 2004)

Çeşitli YSA mimarileri bulunmakla birlikte eğer gizli katman kullanılacaksa bu katmanda kullanılacak etkinleştirme fonksiyonun belirlenmesi gerekir. Öğrenme oranının kullanıcı tarafından farklı problemler için test edilmesinde fayda vardır. Aktivasyon fonksiyonunun türevi alınabilir bir türden seçilmesi hesaplamalar için faydalıdır. Şekil 3.26'da iki tabakalı YSA modeli gösterilmiştir.



Şekil 3.26 İki tabakalı YSA modelinin genel yapısı

3. 27 Yapay Sinir Ağları Modellemesinde Dikkat Edilecek Noktalar

Yapay sinir ağları modellemesinde dikkat edilmesi gereken noktalardan biri verilerin toplanmasıdır. Girdi ve çıktılar dikkatli bir şekilde değerlendirilmelidir. Girdi ve çıktı katmanlarında kaç adet hücre olacağı belirlenmelidir. Aynı zamanda eğer kullanılan YSA modelinde gizli katman olup olmayacağı da belirlenmelidir. Kullanılacak aktivasyon fonksiyonu da ağın öğrenme performansı ve süresi yönünde etkilidir. Geliştirilen indirgeme yazılımında Sigmoid işlemcisi kullanılmıştır. Fonksiyon S şeklinde olduğundan dolayı çoğu zaman "S" fonksiyonu olarak da isimlendirilmektedir. Sigmoid fonksiyonu, genellikle geri yayılım temelli YSA sistemlerinde kullanılan bir fonksiyondur. Yapay sinir ağı temelli sistemlerde, bundan başka eşik işlemcisi, Sigmoid işlemcisi, Hiperbolik, Gauss ve Rampa işlemcisi de kullanılmaktadır. (Şen, 2004)

3.28 Yapay Sinir Ağlarında Eğitim Kuralları ve Tanımları

Bir YSA'nın eğitimi sırasında faydalanabileceği belirli kurallar bulunmaktadır. Bu kuralların temeli Hebb tarafından kurulmuş ve daha sonra da buna bağlı olarak geliştirilmiştir. Şimdiye kadar geliştirilen birçok öğrenme kuralı olmakla birlikte, en çok kullanılan öğrenme algoritmaları aşağıdaki gibidir (Şen, 2004).

1) Hebb Öğrenme Kuralı : Hebb öğrenme kuralı ilk YSA temelli öğrenme kuralı olarak kabul edilmektedir. Temel olarak bir nöron başka bir nörondan bilgi alıyorsa ve her ikisi de aktif iseler, bu hücrelerdeki bağlantı kuvvetlendirilmelidir bunun zıttı durumda ise bağlantıyı zayıflatmak gerekmektedir. Bu kural özet olarak bağlantı değerlerinin artırılıp azaltılmasına bağlıdır.

2) Hopfield (1982) Kuralı : Bu yaklaşım, Hebb öğrenme kuralına benzetmekle birlikte öncekinden farkı bağlantı değerlerinin hem işaret hem de büyüklüklerinin değiştirilmesidir. Bu bağlantı değerlerinde öğrenme katsayısı belirli bir değerde artırılır. Buradaki temel fark öğrenme katsayısının da dikkate alınmasıdır. Öğrenme katsayısı Hebb Öğrenme kuralında 1'e eşit iken Hopfield Kuralında ise genellikle 0 ile 1 arasında değişen bir katsayı ile ifade edilir.

3) Delta Kuralı : Bu kural yaygın olarak kullanılmakla birlikte Hebb kuralının geliştirilmiş versiyonudur. Bu yaklaşımda, hesaplanan YSA çıktı değerlerinin beklenen çıktı değerlerine eşit olması için ağırlıklar değiştirilmektedir. Bu yöntemde, YSA çıktıların ortalama kare hatalarının en küçüğe ulaşması sağlanmıştır. Geri yayılım temelli algoritmalarda hatanın katman katman geri yayılımı amaçlanmıştır. Hatalar en son tabakadan başlayarak öndeki tabakalara doğru dağıtılır (Şen 2004).

4)Kohonen Kuralı Kohonen kuralında herhangi bir çıktı kümesi olması gerekmez. Bu kuralda danışmansız öğrenme yöntemi kullanılmaktadır. Danışmansız öğrenmede amaç daha farklıdır. Kohonen ağlarının amacı rastgele " N " boyutlu giriş vektörlerini 1 ya da 2 boyutlu ayırık boyuta haritalamaktır. Giriş uzayında yer alan birbirine yakın olan örnekler (örüntüler) harita içinde birbirlerine daha yakın olmalıdırlar. Bir Kohonen ağı çıkış üniteleri ve " N " adet giriş ünitesinden oluşmuştur. Her bir çıkış ünitesinin giriş

ünitelerine bağlanmış ve ağırlıklandırılmıştır. Genellikle bu ağırlıklar rastgele üretilmiş sayılardan oluşmaktadır. Kohonen ağlarında öğrenmenin temel basamakları aşağıdaki gibi ifade edilebilir. Ağırlık değerlerine ilk değerleri atanır. Ağırlık değişimleri ihmal edilebilir kabul edilene kadar döngü devam eder. Her bir örnek (pattern) için giriş örneği sunulur ve kazanan çıkış ünitesi belirlenir. Kazanana komşu olan tüm üniteler bulunur. Ağırlık vektörleri tüm bu üniteler için güncellenir. Komşuluğun boyutu, eğer ihtiyaç duyulursa azaltılır. Kazanan çıkış ünitesi, giriş örneğine en az öklit mesafesinde bulunan ünite dir. Bu yaklaşımda, kazanan ünitenin komşuluğundaki her bir ünitenin ağırlıkları denklem ile güncellenir. (Bkz Denklem 3.39)

$$\vec{w}_i = \vec{w}_i + \alpha(\vec{x}_i - \vec{w}_i) \dots \dots \dots (3.39)$$

3.29 Yapay Sinir Ağlarında Geri Yayılım (Back Propagation) Algoritmasının Temel Özellikleri

Çok katmanlı algılayıcıların öğrenme kuralı en küçük kareler kuralının genelleştirilmiş halidir. Ağın öğrenmesi için eğitim seti adı verilen ve örneklerden oluşan bir tabloya ihtiyaç vardır. (Öztemel 2006). Bu tablo kullanılarak hem giriş değerleri hem de o giriş değerlerine karşılık ağın üretmesi gereken çıkış değerler yer almaktadır (Zadeh, 1994) .

1) İleriye doğru hesaplama

Ağın çıktısını hesaplamada ileri doğru hesaplama kullanılır..

2) Geriye doğru hesaplama

Ağırlıkların değiştirilmesi safhasıdır.

İleriye Doğru Hesaplama

Net ve Çıkış Değerleri, Ara katman nöronları için denklem 3.40 ve 3.41 de gibidir (Fogel, 2000 , Öztemel 2006).

$$NET_j = \sum_{i=1}^{i=n} X_i \cdot W_{ij} \text{ olarak ifade edilir.}$$

$$\zeta_{j} = C_j = \frac{1}{1 + e^{-(NET_j^a + \beta_j^a)}} \dots\dots\dots(3.40 \text{ ve } 3.41)$$

3.30 Geriye Doğru Hesaplama

Geriye yayılım algoritmasında, değiştirilen ağırlık değerleri bir sonraki iterasyona uygulanır. Her bir iterasyon için hata değerleri hesaplanarak sonuca ulaşılır. Ara katmanla Çıktı katmanı arasındaki ağırlıkların güncellenmesinde aşağıdaki denklemler kullanılır (Şen, 2004, Öztemel 2006). Denklem 3.42- Denklem 3.53 arasında geriye yayılım algoritmasında kullanılan basamaklar gösterilmiştir.

$$\Delta A_{jm}^a(t) = \lambda \cdot \delta_m \cdot C_j^a + \alpha \Delta A_{jm}^a(t-1)$$

Burada λ , öğrenme katsayısını α ise momentum katsayısını göstermektedir.(3.42)

$$\Delta A_{jm}^a(t) = \lambda \cdot \delta_m \cdot C_j^a + \alpha \Delta A_{jm}^a(t-1)$$

$$\delta_m = C_m \cdot (1 - C_m) \cdot E_m \text{ olur.} \dots\dots\dots(3.43 \text{ ve } 3.44)$$

formülleri kullanılır.

Değişim hesaplandıktan sonra ağırlıkların yeni değerleri şu şekilde hesaplanır :

$$A_{jm}^a(t) = A_{jm}^a(t-1) + \Delta A_{jm}^a(t) \dots\dots\dots(3.45)$$

$$\Delta \beta_m^c(t) = \lambda \cdot \delta_m + \alpha \cdot \Delta \beta_m^c(t-1)$$

olacaktır. Eşik değer t . iterasyondaki değeri ise

$$\beta_m^c(t) = \beta_m^c(t-1) + \Delta \beta_m^c(t) \text{ şeklinde hesaplanacaktır.} \dots\dots\dots(3.46 \text{ ve } 3.47)$$

Ara katman girdi katmanı arasındaki ağırlıkların değiştirilmesi

$$\Delta A_{kj}^i(t) = \lambda \cdot \delta_j^a \cdot C_k^i + \alpha \Delta A_{kj}^i(t-1)$$

Burada λ , öğrenme katsayısını α ise momentum katsayısını göstermektedir.(3.48)

$$\delta_j^a = f'(NET) \cdot \sum_m \delta_m \cdot A_{jm}^a$$

$$A_{kj}^i(t) = A_{kj}^i(t-1) + \Delta A_{kj}^i(t) \dots\dots\dots(3.49-3.50)$$

$$\Delta\beta_j^a(t) = \lambda.\delta_j^a + \alpha.\Delta\beta_j^a(t-1)$$

olacaktır. Eşik değerin t . iterasyondaki değeri ise

$$\beta_j^a(t) = \beta_j^a(t-1) + \Delta\beta_j^a(t) \text{ şeklinde hesaplanır.} \dots\dots\dots(3.50 \text{ ve } 3.51)$$

Epoksal Hata

$$E_m = B_m - \hat{C}_m \dots\dots\dots(3.52)$$

B_m , beklenen çıkış, $\hat{C}_m$, YSA'nın ürettiği çıkış değerini göstermek üzere toplam hata denklem 3,53 de gösterildiği gibi hesaplanır (Öztemel, 2014, Elmas 2016).

$$TH = \frac{1}{2} \sum_m (E_m)^2 \dots\dots\dots(3.53)$$

Oluşturulan sistemde birden fazla çıkışlı bir sistem test edildiği için

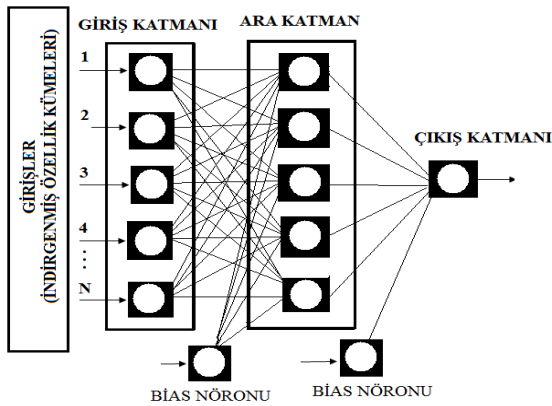
Bütün çıkışlara ait toplam epoksal hata (n adet ayrı çıkışa sahip olan sistem için)

$$E_{total}(\text{çıkış}) = |TH_1| + |TH_2| + \dots + |TH_n| \text{ şeklinde hesaplanmıştır.} \dots\dots\dots(3.54)$$

Bu değer her bir kromozom için uygunluk değeri olarak hesaplanmıştır.

Geriye Yayılımın genel şeması aşağıdaki gibidir (Şekil 3.33).

Şekil 3.27'de geri yayılım algoritmasının genel yapısı gösterilmiştir.



Şekil. 3.27 Geri Yayılım Algoritmasının Genel Yapısı

3.31 Sınıflandırma Doğruluğu Formülü

Yapay Sinir Ağına önceden görmediği veri setinden örnekler gösterilerek ağın performansı aşağıdaki formüle göre ölçülür (Öztemel 2006). Klasifikasyon Doğruluğu Yüzdesi aşağıdaki formüldeki gibi hesaplanır. (Denklem 3.55). Bu denklemde D doğruları, T ise test örneklerinin sayısını göstermektedir.

$$P = \frac{D}{T} * 100 \dots\dots\dots(3.55)$$

3.32 Medikal Alanda Sensitivite, Spesifite ve Kesinlik Kavramları

Medikal sistemlerde sensitivite, spesifite ve kesinlik anlamları aşağıdaki gibi ifade edilir. Aşağıda teşhissel bir teste ait olan çıktı değerleri gösterilmektedir. (Zhu, 2010) Medikal anlamda sensitivite, spesifite ve pozitif prediktif ve negatif prediktif değerler aşağıdaki Tablo 3.12’de gösterilmiştir.

Duyarlılık (Sensitivite): Tıbbi sistemlerde ve medikal sınıflandırmada sensitivite aşağıdaki formüllerle hesaplanır.

Gerçekte sağlık problemine sahip olan bireyler içinde tarama testinin sağlık problemlileri bulabilme özelliği olarak tanımlanmaktadır.

$$\text{Duyarlılık (Sensitivite)} = (GP / (GP + YN)) * 100 \dots\dots\dots(3.56)$$

olarak ve yüzdelik değer olarak ifade edilir.

Duyarlılık kısaca bir tarama testinin hasta bulma yeteneği olarak da tanımlanabilir.

Özgüllük (Spesifite): Gerçekte sağlıklı olan bireyler içinde tarama testinin sağlıklı bireyleri bulabilme özelliği olarak tanımlanmaktadır.

$$\text{Özgüllük (Spesifite)} = (GN / (YP + GN)) * 100 = \% \dots\dots\dots(3.57)$$

olarak ifade edilir.

$$\text{Pozitif Prediktif Değer} = (GP / (GP + YP)) * 100 = \% \dots\dots\dots(3.58)$$

olarak ifade edilir.

$$\text{Negatif Prediktif Değer} = (GN / (YN+GP)) * 100 = \% \dots\dots\dots(3.59)$$

olarak ifade edilir

Tablo 3.12 Sensitivite, Spesivite, Pozitif ve Negatif Prediktif Değerler

<i>Teşhissel Test Sonucu</i>	<i>Pozitif</i>	<i>Negatif</i>	<i>Ham Toplam</i>
<i>Pozitif</i>	Gerçek Pozitif (G. P.)	Yalancı Pozitif (Y. P.)	Gerçek Pozitif + Yalancı Pozitif (Pozitif sonuçlu örnekler toplamı)
<i>Negatif</i>	Yalancı Negatif (Y. N.)	Gerçek Negatif (G. N.)	Yalancı Negatif + Gerçek Negatif (Negatif test sonuçlu örnekler toplamı)
<i>Sutun Toplamı</i>	Gerçek Pozitif + Yalancı Negatif (G. P.+ Y. N.)	Yalancı Pozitif + Gerçek Negatif	N=GP+GN+YP+YN

3.33 Normalizasyon Yaklaşımları: $X_{\min}$ en düşük değerini, $X_{\max}$ ise en yüksek değeri göstermektedir. Klasifikasyon sisteminde testi yapan YSA kısmında $X_{\max}$, o nitelik (giriş) değerine ait olan sutundaki maksimum değeri göstermektedir. $X_{\min}$ ise, o nitelik değerine ait olan minimum değeri göstermektedir.

$$X^* = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \dots\dots\dots(3.60)$$

Örneğin giriş sütununda 4 adet dilsel ifade ile belirtilen bir giriş değeri için “4” ifadesi normalizasyona uğratılınca 1’e, 1 ifadesi ise 0’a dönüşmektedir. Burada sütunsal normalizasyon uygulanmıştır. Yapay Sinir Ağı Temelli Sistemlerde, tüm tablonun maks ve min değerlerine göre de normalizasyon metotları vardır.

Aşağıda minmaks normalizasyon işleminin sütunlara göre örnek bir uygulaması görülmektedir. Ürolojik veriler işlenirken, Min max normalizasyon işlemi ilk önce kaba küme teorisine göre uygun aralıklarla ifade edilen üroflowmetrik değerler dilsel ifadelerle dönüştürülmüştür. Daha sonra bu dilsel ifadeler, yazılımın değişken girişli ve indirgenmiş özellik kümelerini test eden kısmında normalizasyona uğratılmıştır. Normalizasyon için sütun bazlı normalizasyon seçilmiş ve YSA test aşaması kısmı için hem giriş hem de çıkış değerleri kendi ait oldukları sütunda yer alan maksimum dilsel ifadenin matematiksel değerine göre normalizasyona uğratılmıştır. Normalizasyon işlemi, geri beslemeli yapay sinir ağı temelli sistemler için oldukça önemlidir. 1, 2, 3 ve 4 ile ifade edilen maksimum akış hızının dilsel ifadesi olan değerler YSA test aşamasında sutunsal normalizasyona uğratılarak YSA eğitime girerler. Aşağıdaki tabloda bazı normalize değerler gösterilmektedir. Tablo 3.13’de maksimum akış hızına ait bazı giriş değerleri ve normalize edilmiş değerleri gösterilmiştir.

Tablo 3.13 Maksimum akış hızına ait bazı giriş değerleri ve normalize edilmiş değerler

	Giriş Değeri (Maksimum Akış Hızı a_1)	Normalize Değeri
1	1	0
2	2	0,3333
3	1	0
4	2	0,3333
5	1	0
6	2	0,3333
7	1	0
8	2	0,3333
9	3	0,6666
10	2	0,3333
		
n.	4	1

Z Skor (Sonucuna Göre) Standartlaşma : Z Skor standartlaştırma, birçok istatistiksel uygulamalarda sıklıkla uygulanan bir normalizasyon işlemidir. Bu standartlaştırma ve normalizasyon tekniği verilerin ortalaması ve standart hatasına göre değerlerin

hesaplanmasına dayanır. Bu tür bir normalizasyonda ilk önce verilerin standartlaştırılmalarına dikkat edilir (Özkan 2008). Z skor standartlaştırma denklem 3.61 - 3,63 'de verilmiştir.

Z skor Standartlaştırma

$$X^* = \frac{X - \bar{X}}{\sigma_x} \dots\dots\dots(3.61)$$

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i \dots\dots\dots(3.62)$$

$$\sigma_x = \sqrt{\frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n-1}} \dots\dots\dots(3.63)$$

1) Örneklerin Toplanması. Örneklerin toplanması hem eğitim seti için hem de test aşaması için önemlidir. 2) Ağın Yapısının ve Çözülecek Problemin Belirlenmesi 3) Öğrenme Parametrelerinin Belirlenmesi 4) Ağırlık Değerlerinin Atanması 5) Elde edilen değerin beklenen değer ile karşılaştırılması ve ağırlıkların değiştirilerek, yeni ağırlıklarla sistem çıkışının hesaplanması

Oluşturulan YSA test yazılımında ilk ağırlık rasgele (random) olarak atanabildiği gibi sonraki test aşamalı için YSA da hesaplanan ağırlık değerleri bir text dosyasında kaydedilip daha sonra o test dosyasından da kontrol edilebilmektedir.

3.34 Yapay Sinir Ağı Temelli Öğretmensiz Öğrenme Modelleri

Danışmansız öğrenme şekillerine örnek olarak LVQ gösterilebilir. Kullanılan indirgeme yazılımında tablodaki çıkış değerleri de referans alındığından dolayı, bu yöntem kullanılmamıştır. Doğrusal Vektör Parçalı YSA bunlardan biridir. (Öztemel 2006).

3.34.1 Doğrusal Vektör Parçalı YSA Modeli (LVQ)

1) Giriş Katmanı: Sadece veri girişinin yapıldığı katmandır.

2) Kohonen Katmanı: Bu katmanda kalan her bir eleman bir referans vektörünü göstermektedir.

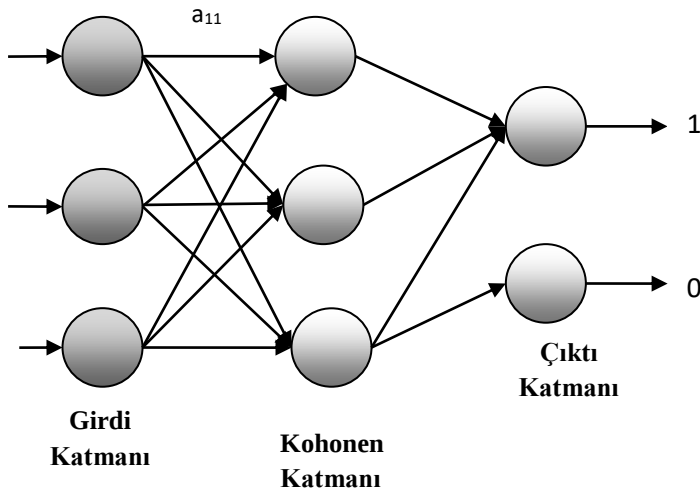
3) Çıktı katmanı: Bu katmanda girdinin ait olduğu sınıf belirlenmektedir. LVQ (DVP YSA) ağının çalışma prosedürü aşağıda verilmiştir.

Bu çalışma prosedüründe,

Aşağıdaki matriste en sağdaki sutunda bulunan i . giriş vektörü, g_{ij} ($j=1, 2, \dots, n$), en solda bulunan i .inci çıkış vektörü ζ_{ij} ($j=1, 2, \dots, n$) matriste yer alan a_{ij} ler ise çıkış vektörlerini göstermektedir.

$$\begin{bmatrix} \zeta_{i1} \\ \zeta_{i2} \\ \vdots \\ \zeta_{im} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{1,2} & \cdot & \cdot & \cdot & a_{1,n} \\ a_{21} & a_{2,2} & \cdot & \cdot & \cdot & a_{2,n} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{m,1} & a_{m,2} & \cdot & \cdot & \cdot & a_{m,n} \end{bmatrix} * \begin{bmatrix} g_{11} \\ g_{12} \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ g_{in} \end{bmatrix} \dots\dots\dots(3.64)$$

LVQ ağının genel yapısı aşağıda verilmiştir(Bkz Şekil 3.28) (Öztemel 2003, Şen 2004)



Şekil 3.28 Doğrusal Vektör Parçalı YSA Modeli

3.35 Veri Madenciliğinde Diğer Bazı Bilgi Keşfi Algoritmaları

3.35.1 Birliktelik Kurallarının Çıkarılması Veri madenciliğinde birbirleriyle ilgisi olan durumların tespit edilmesi birliktelik kurallarının çıkarılması ile ilgili olmaktadır. Birliktelik kuralları, medikal alanda hastalarda bulunan semptomların değerlendirilmesinde de kullanılmaktadır. Birliktelik kuralları, nesneler arasında bilinmeyen korelasyonların ve ilişkilerin hesaplanmasında özellikle bilgi keşfi oldukça önemlidir. Veri madenciliğinde özellikle, satılan ürünlerin analizinde birliktelik kuralları kullanılmaktadır. Aşağıdaki örnekte birliktelik kuralları ve apriori algoritmasının işlem basamakları ayrıntılı bir şekilde anlatılmıştır. Bu algorithmada belirlenen bir güven oranına bağlı olarak birliktelik kuralları oluşturulmaktadır. Aşağıdaki denklemler “destek” ve güven oranlarını ifade etmektedir.

Örneğin, Birliktelik kuralları market analizinde kullanıldığında, her bir satır bir müşteriye satıra ait olan sütunlar ise, müşterilerin satın aldığı alışverişleri gösterebilir. Bu algoritma giriş değişkenlerinin indirgenmesini değil, değişkenler arasındaki ilişkileri ve bağıntıları ortaya çıkarmaktadır. (Bkz. Denklem 3.65 ve 3.66)

$$\text{Destek}(A \Rightarrow B) = P(A \cup B) \dots \dots \dots (3.65)$$

$$\text{Güven}(A \Rightarrow B) = P(B | A) \dots \dots \dots (3.66)$$

1) Algoritmanın ilk iterasyonunda her bir küme tek elemanlı olarak başlar ve tek elemanlı kümelerin tablo içerisinde geçme sayıları belirlenir. Önceden belirlenen minimum destek oranı ile kümelerin tablo içinde geçme sayıları karşılaştırılır. Minimum destek oranını geçen kümeler iki elemanlı kümeleri oluşturmak için seçilirler. Aynı iterasyonlar iki elemanlılardan üç elemanlılar ve daha sonra dört elemanlılar oluşturulana kadar devam eder.

Apriori algoritması iki ve daha fazla elemanlı kümelerin oluşturulmasında aşağıdaki adımlar uygulanmaktadır.

- 1) Birleşme Adımı
- 2) Budama Adımı

3.35.2 Yaygın (Minimum Destek Şartını Sağlayan) Kümelerden Birliktelik Kurallarının Üretilmesi

Yaygın kümelerden birliktelik kurallarının üretilmesi aşağıdaki gibi ifade edilir. Birliktelik kurallarından faydalanılarak aşağıdaki denklem kullanılarak güven şartını sağlayan kurallar elde edilir. Aşağıdaki denklemde destek oranı ifade edilmiştir. Aşağıdaki denklemdeki Birleşme operatörü, her iki durumun da aynı anda görülme durumunu ifade etmektedir. Denklem 3.67 da güven oranı ifade edilmiştir.

$$\text{güven (confidence) } (A \rightarrow B) = P(B|A) = \frac{\text{destek sayısı } (A \cup B)}{\text{destek sayısı } (A)} \dots\dots(3.67)$$

3.35.3 Veri Madenciliğinde Sınıflandırma ve Karar Ağaçları

Veri madenciliğinde çok farklı yaklaşımlar bulunmakla birlikte bunlardan birisi de karar ağacı ve karar ağacı sınıflandırıcılardır. Genellikle karar ağaçları, danışmanlı öğrenme metotları arasında kabul edilmekte ve karar ağacı uygulandıktan sonra yorumlaması “Eğer o halde” kuralları ile olmaktadır. Özellikle kural çıkarımında ve kural tabanının oluşturulması vb. işlemlerde kullanılırlar. Karar ağacı sınıflandırıcısında algoritma ağacın en üst noktasından alt noktalarına doğru ilerleyerek bir sınıflandırma yapmaktadır. Genellikle ağaç oluşturma aşağıdaki adımlarla başlar ve devam eder.

4. ÖNERİLEN YÖNTEM

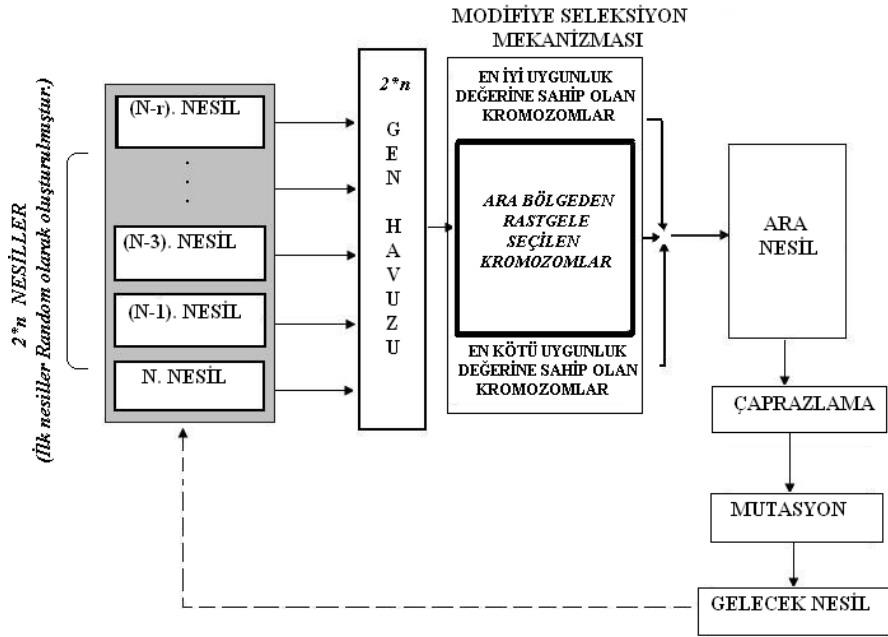
4.1 Genetik Algoritmada Kullanılan Bazı Terimler ve Geliştirilen Yazılım Arayüzü

Popülasyon, genetik algoritmadaki bir nesil içindeki kromozom sayısıdır. Maksimum jenerasyon sayısı, genellikle genetik algoritma temelli sistemlerde durdurma kriteri olarak kullanılır. Jenerasyon (nesil), popülasyonda yer alan kromozomların oluşturduğu çözümler kümesidir. Klasik genetik algoritmada, jenerasyonda yer alan kromozomlar, çaprazlama ve mutasyon işlemlerine tabi tutularak yeni jenerasyon oluşturulur. Çaprazlama oranı kullanılan seleksiyon algoritmasının çeşidine göre çaprazlama yapılacak kromozomların toplam kromozom sayısına oranıdır. Ara nesil kullanan sistemlerde ara nesile seçilen kromozomların belirtilen oranı çaprazlamaya tabi

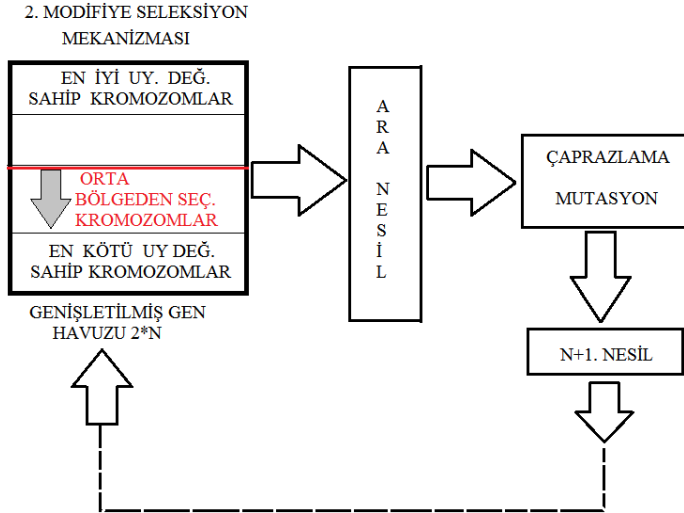
tutulur.Çaprazlama,klasik yöntemlerde seçilen kromozom çiftlerinin, kullanılan yönteme göre birbirlerinden gen alışverişi temeline dayanır.

Genetik Algoritmada kullanılan temel çaprazlama operatörleri aşağıda verilmiştir. Oluşturulan yazılım tek noktalı çaprazlama, kısmi planlı çaprazlama (PMX) ve doğrusal sıralı çaprazlama operatörlerini desteklemektedir. Geliştirilen GARSBS (Genetik Algoritma ve Kaba Küme Temelli Hibrit Sistem Yazılımı), klasik seçim mekanizmalarından rulet çemberi ve suni seçim mekanizmasını da desteklemektedir.

Ayrıca yazılım yeni geliştirilen iki farklı modifiye suni seçim mekanizmasını da desteklemektedir. Geliştirilen modifiye suni seçim mekanizmaları hem işlem zamanını ve performansını arttırmakta hem de genetik algoritma temelli sistemin daha iyi çözüm kümelerine daha kısa sürede ulaşmasını sağlamaktadır. Geliştirilen yazılımın genel görünümü aşağıdaki şekilde gibidir. 1. modifiye suni seçim mekanizması Şekil 4.1’de gösterilmiştir. Ayrıca Şekil 4.2’de, geliştirilen ikinci yeni modifikasyon gösterilmiştir. Yazılım ara yüzünün genel yapısı Şekil 4.3’te, test arayüzü ise Şekil 4.4 ve 4.5 ’de gösterilmiştir.



Şekil 4.1 Modifiye Seçim Mekanizmalı Genetik Algoritma ve Kaba Küme Kullanan İndirgeme Yazılımı



Şekil 4.2 Yeni geliştirilen 2. Modifiye Suni seçim Mekanizması (Kromozomları ortadan ve sonrasında alıyor)

HYBRID APPROACH FOR FEATURE REDUCTION

Selection Method: **Mod. Art. Sel. with Random Sol**

Rates (Best and Worst): **30** **30**

Solution Addition Type: **Random**

The Generations in the Genpool(2n): **n = 3**

Crossover Method: **Order Based Crossover** **50**

Mutation Method: **Invesion Mutation** **50**

Number of Max. Gen.: **1000**

Population: **8**

Stopping Criterion: **0.999**

ARTIFICIAL NEURAL NETWORK PARAMETERS

Input Number: **5**

Hidden Neuron No: **5**

Output No: **1**

Learning Rate: **0.3**

Maks. Epok No: **500000**

Error Rate: **0.0001**

Buttons: Load From File, Run the System, Save The Results, Run The ANN

Working Time: ☐

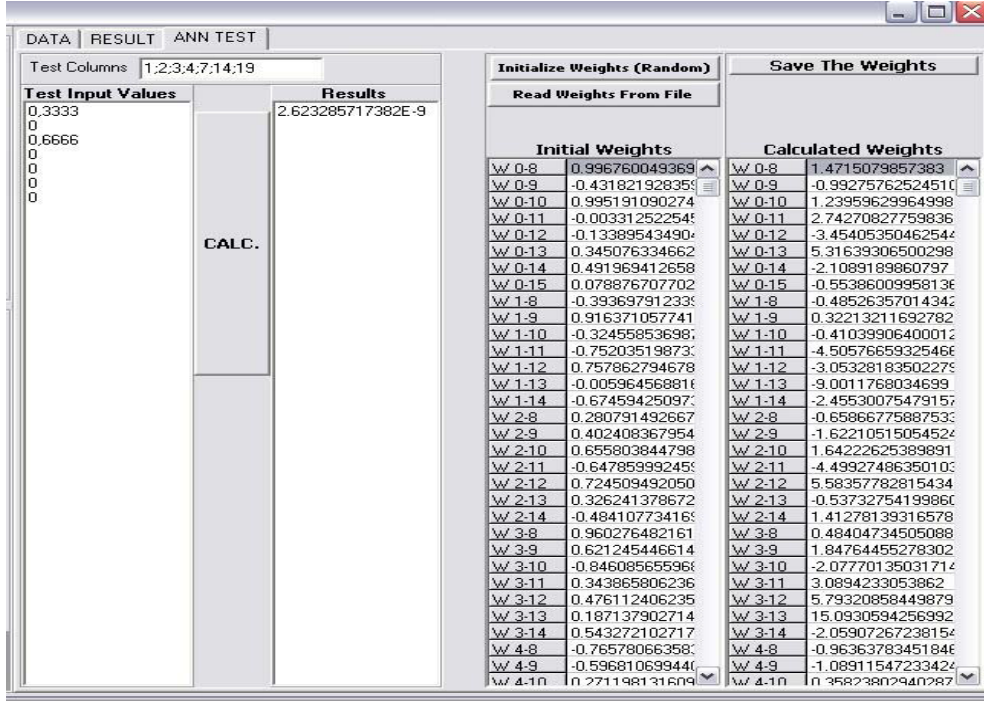
Memory Usage: ☐

DATA RESULT ANN TEST

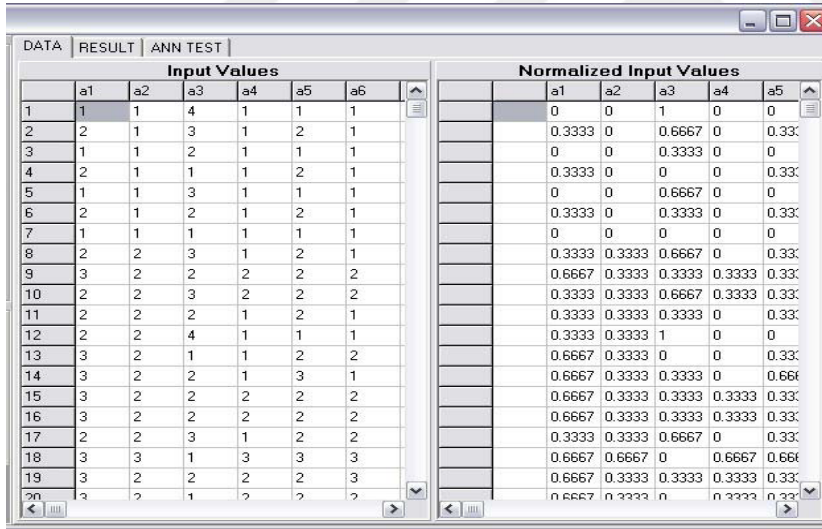
Delete

6. Chromosome: 18, 4, 6, 10, 15, Fitness Value = 0.8833333333333333 ,
 7. Chromosome: 8, 2, 19, 17, 1, Fitness Value = 0.85 ,
 8. Chromosome: 1, 10, 3, 2, 17, Fitness Value = 0.8416666666666667 ,
 =====
 5. Generation Chromosomes
 =====
 1. Chromosome: 13, 4, 12, 7, 15, Fitness Value = 0.9833333333333333 ,
 2. Chromosome: 3, 16, 7, 18, 15, Fitness Value = 0.9833333333333333 ,
 3. Chromosome: 2, 1, 15, 3, 4, Fitness Value = 0.9583333333333333 ,
 4. Chromosome: 10, 15, 2, 17, 1, Fitness Value = 0.7833333333333333 ,
 5. Chromosome: 17, 7, 3, 11, 1, Fitness Value = 0.9166666666666667 ,
 6. Chromosome: 18, 4, 6, 10, 15, Fitness Value = 0.8833333333333333 ,
 7. Chromosome: 2, 10, 17, 15, 1, Fitness Value = 0.7833333333333333 ,
 8. Chromosome: 1, 10, 3, 2, 17, Fitness Value = 0.8416666666666667 ,
 =====
 For The Gene Pool From 5. Gen. and preceding Gen.Chromosomes
 =====
 1. Chromosome: 13, 4, 15, 7, 12, Fitness Value = 0.9833333333333333 ,
 2. Chromosome: 13, 4, 12, 7, 15, Fitness Value = 0.9833333333333333 ,
 3. Chromosome: 2, 10, 17, 15, 1, Fitness Value = 0.7833333333333333 ,
 4. Chromosome: 2, 10, 17, 15, 19, Fitness Value = 0.75 ,
 5. Chromosome: 3, 11, 7, 14, 18, Fitness Value = 0.9833333333333333 ,
 6. Chromosome: 17, 7, 3, 11, 1, Fitness Value = 0.9166666666666667 ,
 7. Chromosome: 10, 15, 2, 17, 1, Fitness Value = 0.7833333333333333 ,
 8. Chromosome: 18, 5, 20, 1, 13, Fitness Value = 0.9 ,
 =====
 6. Generation Chromosomes
 =====
 1. Chromosome: 13, 1, 11, 17, 7, Fitness Value = 0.9166666666666667 ,
 2. Chromosome: 13, 12, 4, 7, 15, Fitness Value = 0.9833333333333333 ,
 3. Chromosome: 2, 1, 15, 17, 10, Fitness Value = 0.7833333333333333 ,
 4. Chromosome: 2, 10, 17, 15, 19, Fitness Value = 0.75 ,
 5. Chromosome: 11, 1, 7, 2, 3, Fitness Value = 0.8833333333333333 ,
 6. Chromosome: 17, 13, 11, 7, 1, Fitness Value = 0.9166666666666667 ,
 7. Chromosome: 10, 15, 2, 17, 1, Fitness Value = 0.7833333333333333 ,
 8. Chromosome: 18, 5, 20, 1, 13, Fitness Value = 0.9 ,
 =====
 For The Gene Pool From 6. Gen. and preceding Gen.Chromosomes
 =====

Şekil 4.3 Genetik Algoritma ve Kaba Küme Temelli İndirgeme Yazılımının Arayüzü



Şekil 4.4 GARSBS' ın YSA yabancı değişken girişli test sistemi



Şekil 4.5 Değişken Girişli YSA sınıflandırıcının normalize etme kısmı

GARSBS' de kullanılan Genetik Durdurma Kriterleri

1-Nesil Sayısına göre durdurma

2-Durdurma Kriteri (Uygunluk Değerine Göre)

Değişken Genetik Giriş Parametreleri

1-Çaprazlama Oranı (%)

2-Mutasyon Oranı (%)

3-Nesil (Jenerasyon) Sayısı

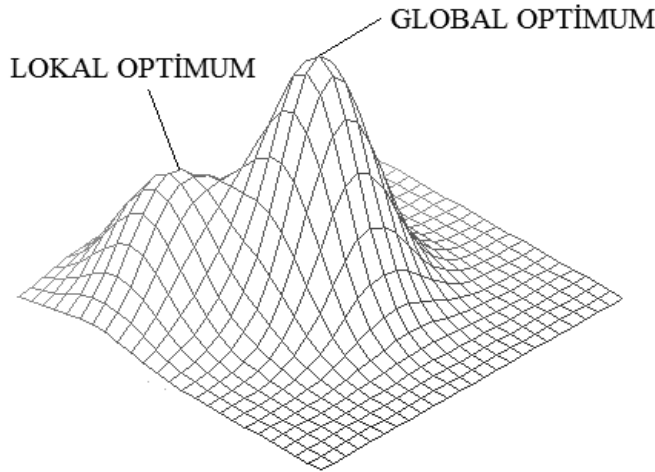
4-Populasyondaki Birey (Kromozom) Sayısı

4.2 Lokal ve Global Optimum Durumları

Genetik algoritma temelli sistemlerde en önemli problemlerden birisi de çözümlerin global optimuma takılma problemidir. Genetik algoritmada özellikle popülasyondaki birey sayısının az olduğu durumlarda, aday çözüm kümelerinin belirli noktalara takılması ve daha farklı çözüm kümelerinin bulamaması yerel optimum noktaya takılmaya örnektir. Genetik algoritma temelli sistemler, lokal optimum noktalarına takıldığı zaman, sonraki nesillerde kromozomların tekrarlanması daha sonraki nesillerde dahi iyi çözüm kümelerinin bulunmasını engellemektedir. Global optimum ise genetik algoritma temelli sistemlerin bütün tüm jenerasyonlarda hesapladığı en iyi uygunluk değerine sahip olan kromozomdur. Lokal optimum, çoğu zaman en iyi çözüm noktasını temsil etmemektedir. Global optimum, literatürde farklı anlamlarda kullanılmakla beraber, çoğu zaman tüm jenerasyonlarda hesaplanan maksimum uygunluk değerine ait olan kromozomun ürettiği çözümü temsil etmektedir.

Yeni geliştirilen modifiye suni seçim mekanizmalarını kullanan genetik algoritma ve kaba küme temelli hibrit özellik indirgeme yazılımı lokal optimuma takılma problemini çözerek, sistemin daha yüksek uygunluk değerine sahip olan çözümleri bulmasını sağlamaktadır. Ayrıca, hafızayı daha verimli kullanmayı sağlayarak daha çok girişli sistemlerin indirgenmesi sağlanmaktadır.

Genetik algoritma temelli sistemlerde lokal optimum ve global optimum problemini açıklamak için aşağıdaki grafik kullanılabilir. Aşağıdaki Şekil 4.6'da genel bir optimizasyon probleminde elde edilen lokal ve global optimum noktaları 2 değişkenli bir sistem için göstermiştir.



Şekil 4.6 Genetik algoritma temelli sistemde Yerel ve Global Optimum noktaları

4.3 GARSBS’de Kullanılan Durdurma Kriteri

Geliştirilen indirgeme sistemi yazılımında, kullanılan durdurma kriteri aşağıda verilmiştir. Kromozom uygunluk değerleri kaba küme teorisindeki özellik (nitelik bağımlılığı) değerine göre hesaplanır. Klavyeden girilen özellik bağımlılığı değerine eşit veya daha yüksek sonuçlar bulunduğu yazılım sonuçları yazdırır.

4.4 Yeni Geliştirilen Modifiye Suni seçim Mekanizması

Çalışmada, yeni geliştirilen seçim mekanizmaları klasik suni seçim mekanizmasının modifiye edilmiş şeklidir. İki adet yeni modifikasyon önerilerek geliştirilmiş ve önerilmiştir ve klasik suni seçim mekanizmasının çalışması ve performansı iletmiştir.

Yeni önerilen ve geliştirilen Modifiye suni seçim mekanizmalarında ara nesil içim kromozom seçim işlemi aşağıdaki gibi yapılmaktadır. Kullanıcı girişine bağlı olarak “2n” adet son üretilen jenerasyonları ara nesil oluşturmak için kullanan ve

geliştirilen modifikasyon, kromozomların seçim mekanizmalarında yapılan 2 modifikasyonla birleştirilerek oluşturulan yazılım daha verimli ve hızlı çalışabilir hale getirilmiştir. Sistem, yapay sinir ağları temelli yazılımla sistem test edilmiş ve yüksek sınıflandırma doğruluğu değerleri elde edilmiştir. Sınıflandırma kısmına eklenen yapay sinir ağları temelli yazılımda gizli nöron sayısı, öğrenme hızı, maksimum epok sayısı vb. parametreler kullanıcı tarafından değiştirilebilmektedir. Yapay sinir ağları temelli indirgeme yazılımı indirgenmiş özellik kümelerini işleyerek farklı örnekler için de klasifikasyon yapabilmektedir. Geliştirilen ve son “ $2*n$ ” jenerasyonu ara nesil olarak kullanan modifikasyon daha hızlı ve etkin reduct bulunmasını sağlayarak sistem performansını arttırmıştır. “ n ” parametresi kullanıcı tarafından girilebilmektedir. Ayrıca başlangıçta ilk $2*n$ adet jenerasyon birbirlerinden bağımsız olarak rastgele yöntemle üretildikten sonra ilk jenerasyon ve son $2*n$ jenerasyon gen havuzunda toplanmaktadır. Örneğin $n=4$ olduğu zaman son 8 kuşak gen havuzunda toplanmaktadır.

Geliştirilen yazılım, farklı çaprazlama ve mutasyon yöntemlerini de desteklemektedir. Sistem, başka sistemlerle karşılaştırılmış ve karşılaştırılan sistemlere göre daha yüksek sınıflandırma doğruluğuna sahip olduğu görüldü. Ayrıca geliştirilen yazılımın performansı, karara bağlı ayırt edici matris kullanan ayrı bir yazılım kullanan ve diğer indirgeme sistemleri kullanılarak karşılaştırılmıştır. Sistem yazılımına eklenen değişken girişe adapte olan YSA yazılımı optimize edilerek hızlandırılmış ve klasifikasyon sistemi daha etkin ve verimli çalışabilecek hale getirilmiştir. Sistem performansı “ $2*n$ ” değişken jenerasyonlu gen havuzu sayesinde arttırılmıştır. Sistem performansı başka algoritmaları kullanan test yazılımları oluşturularak, geliştirilen sistemle karşılaştırılmıştır. Hızlı İndirgenmiş Özellik Değerleri, Ters İndirgenmiş Özellik Değerleri yaklaşımları ve Karara Bağlı Ayırt Edici Matris kullanan yaklaşımlarla karşılaştırılmış ve geliştirilen sistemin daha yüksek bir sınıflandırma doğruluğuna sahip olduğu gözlemlenmiştir.

Rastgele (Random) aday çözüm kümesi eklemede en iyi çözümler, en kötü çözümler ve rastgele seçilen çözümlerden belirlenen oranlarda alınır. Rastgele seçilen kromozomlar, bütün kromozomlar uygunluk değerlerine göre sıralandıktan sonra, en iyi çözüm kümelerinin ve en kötü çözüm kümelerinin arasında kalan alandan seçilir böylece random seçimlerde hem tekrarlanmanın önüne geçilmiş olunur. Orta uygunluk değerli kromozomları ekleyen yeni modifiye versiyona ait denklem aşağıda verilmiştir. İki modifiye suni seçim mekanizmasında da gen havuzunda $2*n$ sayıda jenerasyon toplanmaktadır. İlk $2*n$ jenerasyon rastgele yöntemle üretilmektedir. Orta uygunluk

değerlerini ekleyen modifikasyonda ise, kromozomlar en büyükten küçüğe sıralandıktan sonra, ortada yer alan kromozomdan başlayarak sırasıyla aşağıya doğru gidilerek seçim yapılır. Başlangıç jenerasyonları olarak $2*n$ jenerasyon rastgele (random) olarak oluşturulmuştur.

Yeni geliştirilen modifiye Suni seçim mekanizmasında kromozomların seçim oranını belirleyen formüller aşağıdaki gibidir.

$$\%RSÇ \text{ (Rastgele Seçilen Çözüm Kümeleri)} = \%100 - (\%EİÇK + \%EKÇK)$$

$$\%OBSÇ \text{ (Rastgele Seçilen Çözüm Kümeleri)} = \%100 - (\%EİÇK + \%EKÇK)$$

4.5 Rulet Çemberi Mekanizmasında Eklenen Yeni Modifikasyon

Rulet çemberi mekanizmasının en önemli problemlerinden biri de rulet çemberi mekanizmasının lokal optimum noktalara takılma problemidir. Şekil 4.7’de rulet çemberi mekanizması için yeni geliştirilen ve önerilen mekanizmanın çalışma prensibi gösterilmiştir. Elitizm (seçkinlik) ve benzeri yaklaşımlarda en iyi çözümlerin korunması amaçlanmaktadır. Klasik elitizm versiyonunda en iyi çözümler korunmakta ve hiçbir işleme tabi tutulmadan yeni jenerasyona eklenmektedir. Bu da bazen daha iyi çözüm kümelerini bulmayı engelleyebilmektedir.

Geliştirilen modifikasyonda hem iyi çözümleri koruyan bir versiyonda bulunmaktadır.

Rulet çemberi seçim mekanizması özellikle giriş değişkenlerinin sırasının önem arz etmediği durumlarda lokal optimum noktalara daha çok yakalanmakta ve böylece algoritma daha iyi çözüm kümelerini yakalayamamaktadır. Ara nesilde tekrar eden yerel optimum çözümler global optimuma erişmeyi engellemektedir. Bu problem incelenen genetik algoritma problemine göre değişmektedir. Bu problemin çözümü için önerilen ve indirgeme yazılımına eklenen yeni modifikasyonun çalışma adımları aşağıdaki gibidir. Modifikasyonun rulet çemberi mekanizmasına eklenmesinin sebebi, lokal optimuma takılma probleminin bu seçim mekanizmasında özellikle özellik indirgeme gibi problemlerin çözümünde belirgin bir şekilde görülmesidir.

Modifikasyon kullanıcıya göre seçmeli olarak aşağıdaki bir biçimde yazılıma entegre edilmiştir. Modifikasyon sadece belirlenen jenerasyonda (T) ya da her $n*T$ jenerasyonda dışarıdan çözüm ekleme mekanizmasını kullanmaktadır. Böylece yerel

optimuma takılma problemi çözülmüş ve algoritmanın daha kısa sürede çözüme ulaşması sağlanmış olur.

Eski jenerasyonda yer almayan çözümleri eklemek için yazılım “m” kere kontrol yapmaktadır. Önerilen yeni hibrit modifikasyonun ayrıntıları aşağıdadır.

Dışarıdan çözüm eklemeli yeni modifiye rulet çemberi seçim mekanizmasının çalışma şekli

- 1) Dışarıdan üretilen çözüm kümeleri 1. modifiye rulet çemberi versiyonunda sadece t . jenerasyondaki kromozomları değiştirecek
- 2) 2. modifiye versiyonda her t jenerasyonda bir müdahale edecek. t yazılıma kullanıcı tarafından girilecektir.
- 3) Kullanıcının seçimine bağlı olarak Rulet Çemberi Seçim mekanizmasında
 - 1) Yerine getirilen k adet kromozom, değiştirilen jenerasyonla aynı kromozomlara sahip olup olmadığı kullanıcı tarafından belirlenen parametre ile kontrol edilecek. Yazılım kullanıcı tarafından girilen “ m ” parametresi kadar döngüyle değiştirilecek jenerasyonda olmayan yeni kromozom değerlerini üretmeye çalışır. Örneğin $m=5$ girildiği zaman 5 kere farklı kromozom üretmeye çalışır. Çözüm kümesinin sınırlı olduğu ve farklı üretmediği durumlarda ise son ürettiği kromozomları silinen değerlerin yerine yazar.

Kromozomlar uygunluk değerlerine göre sıralanmaz ve rulet çemberi mekanizması tarafından belirlenen ve liste sonunda yer alan k adet kromozom silinir.

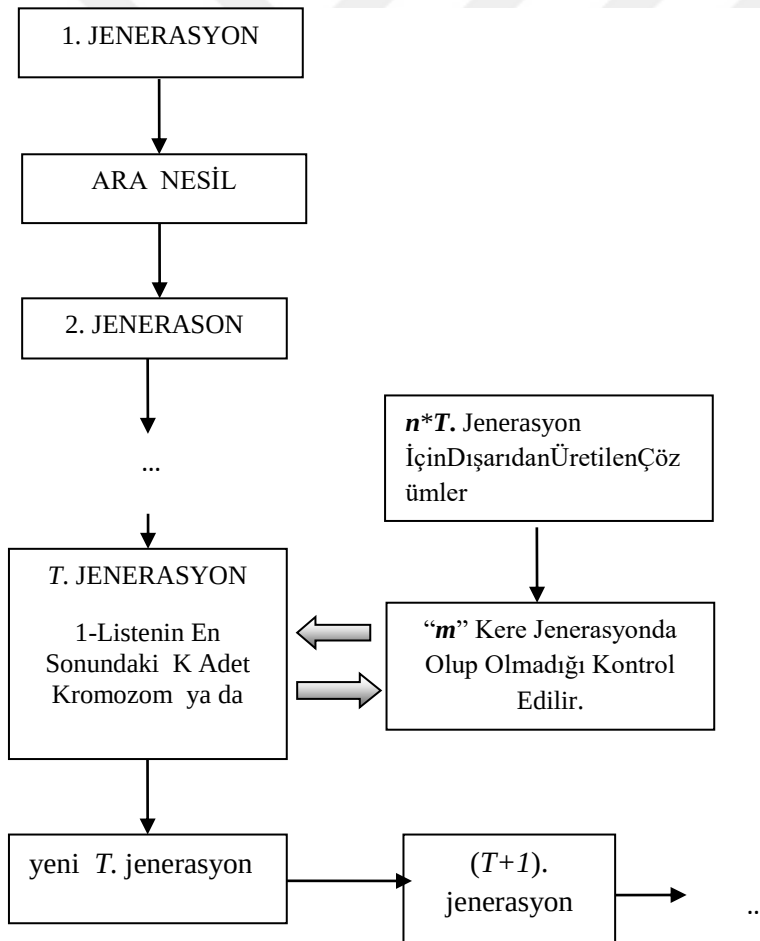
4.6 Geliştirilen Genetik Algoritma ve Kaba Küme Temelli Yazılımda Kullanılan Durdurma Kriterleri

Geliştirilen yazılımda, özellik bağımlılığı değerlerinin farklı değerleri için yazılımın aday çözüm kümeleri üretmesi sağlanarak yazılımın farklı veri tabanlarına uyumluluğu sağlanmıştır. Burada γ , kullanıcı tarafından girilen değerdir. Yazılım, kaba küme metodolojisindeki özellik bağımlılığı değerine göre durdurulabilmektedir. Kullanıcı tarafından girilen farklı özellik bağımlılığı değerlerine göre genetik algoritma

arayüzü durdurulabilmektedir. Özellik Bağımlılığı değeri kullanıcı tarafından “0” ile “1.0” arasındaki değerlerde girilebilmektedir. Böylece hem yazılımın farklı veri tabanlarına uyumluluğu sağlanmış ve kademeli olarak indirgenmiş özellik kümelerinin hesaplanabilmesi sağlanmıştır.

Örneğin özellik bağımlılığı değeri “0,95” olarak belirlendiğinde özellik bağımlılığı 0,95 ve üzerinde olan değerler indirgenmiş giriş değerleri (reduct) olarak ifade edilmektedir. “ θ ” kullanıcı tarafından girilen bir değer olmak üzere, indirgeme yazılımının durdurma kriteri Denklem 4.1’de gösterilmiştir.

$$\text{Durdurma Kriteri eğer } \gamma (\text{Kromozom}) = \frac{|POSP (Q)|}{|U|} \geq \theta \dots\dots\dots(4.1)$$



Şekil 4.7 Rulet Çemberi Seçim mekanizmasına eklenen yeni modifikasyon

Yazılım sistemine eklenen başka seçim mekanizmaları da vardır. Bu seçim mekanizmaları stochastic universal sampling (Tahmini Evrensel Örnekleme) ve rank (Sıra) temelli seçim mekanizmalarıdır. Bu seçim mekanizmalarına da performans artırımı sağlamak için dışarıdan çözüm eklemeli modifikasyon eklenmiştir. Bu seçim mekanizmalarının da çeşitli versiyonları bulunmakla birlikte, indirgeme yazılımında test amaçlı kullanılan versiyonları aşağıda açıklanmıştır. Rank temelli seçim mekanizması ve stokhastik evrensel örnekleme (stochastic universal sampling) metodları aslında rulet çemberi seçim mekanizmasının değiştirilmiş ve ilerletilmiş versiyonları olarak kabul edilirler.



5. DENEYSEL SONUÇLAR

5.1. Geliştirilen İndirgeme Yazılımı ve Diğer Yazılımlarla Karşılaştırılması

Geliştirilen indirgeme yazılımı çok girişli sistemleri desteklemektedir. Geliştirilen modifikasyonlu seçim mekanizması kullanan genetik algoritma ve kaba küme temelli indirgeme yazılımı, hafızayı etkin bir şekilde kullanmaktadır. Geliştirilen indirgeme yazılımı, yüksek klasifikasyon doğruluğunda sonuçlar elde etmektedir.

Birçok indirgeme yazılımı aşırı hafıza kullanımı ve hafıza hataları nedeniyle çok girişli sistemleri desteklememektedir.

Yüksek giriş değerine sahip olan sistemlerin indirgenmesi, sistem tarafından aşırı bellek istendiğinden dolayı birçok indirgeme algoritmasının yazılım versiyonları tarafından gerçekleştirilememektedir. Geliştirilen yazılım 8, 12, 20 girişli (120 satırlı) ve 33 girişli (366 satırlı) veri tabanlarında indirgeme için denenmiştir. Yazılım 100 girişe kadar giriş verilerini desteklemektedir. Ürolojik veri ve dermatolojik veriler üzerinde indirgenmiş özellik kümeleri (reductlar) hesaplanmıştır.

Eklenen yeni modifiye suni seçim mekanizmasıyla algoritmanın lokal ve global optimum noktalara takılma problemi çözülmüştür. Önerilen Modifiye suni seçim mekanizmaları, klasik suni seçim mekanizmasından ve rulet çemberi seçim mekanizmasından daha kısa sürede ve yerel optimum noktalara takılmadan çalışmıştır.

Ayrıca rulet çemberi mekanizmasına eklenen dışarıdan çözüm eklemeli yeni geliştirilen modifikasyon rulet çemberi mekanizmasındaki yerel optimum noktalara takılma problemini çözmüştür. Ayrıca daha iyi özellik bağımlılığına sahip olan özellik kümeleri daha kısa zaman içinde çözümlenmiştir.

Birçok indirgeme algoritması ve yazılımı aşırı hafıza kullanımı nedeniyle yüksek giriş değerlerini desteklememekte ve yüksek giriş değerine sahip sistemlere uyumlu değildir ve yüksek giriş değerlerinde aşırı hafıza kullanımı olmaktadır.

Aşağıda 20 girişli ürolojik sistemin indirgemesinden elde edilen reductlar gösterilmiştir. Geliştirilen indirgeme yazılımında 20 giriş 5 girişe indirgenmiştir. Eşik değerinin üzerinde yer alan 6, 7 ve 8 girişli indirgenmiş bazı özellik kümeleri aşağıda verilmiştir. Belirlenen sabit bir jenerasyondan sonra elde edilen özellik bağımlılığı değerleri aşağıdaki gibidir (Bkz Tablo 5.1).

Tablo 5.1 Genetik Algoritma ve Kaba Küme Temelli İndirgeme Yazılımı Tarafından Hesaplanan İndirgenmiş Özellik Kümeleri ve Özellik (Nitelik) Bağımlılığı Değerleri

Eleman.	İndirgenmiş Özellik Kümeleri	Özellik Bağımlılığı Değeri
8	$a_1, a_2, a_3, a_5, a_7, a_9, a_{12}, a_{15}$	1.0
	$a_1, a_2, a_3, a_4, a_5, a_8, a_{10}, a_{12}$	0,983
	$a_2, a_3, a_4, a_5, a_6, a_8, a_{10}, a_{12}$	0,983
7	$a_1, a_2, a_3, a_4, a_7, a_{14}, a_{19}$	1
	$a_1, a_2, a_3, a_4, a_5, a_{12}, a_{15}$	0,983
6	$a_1, a_2, a_3, a_5, a_8, a_{12}$	0,958
5	$a_1, a_2, a_3, a_4, a_{20}$	1

20 girişli ürolojik sisteme ait test prosedürü, 1. modifiye suni seçim mekanizmasını destekleyen geliştirilen genetik algoritma ve kaba küme temelli indirgeme yazılımı kullanılarak yapılmıştır. Kullanılan RAM oranları ortalama değerlerdir. 20 girişli ürolojik sistemde modifiye suni seçim mekanizmasını kullanan versiyonda YSA tipi sınıflandırıcı %90'nın üzerinde klasifikasyon doğruluğu elde edildi.

Johnson indirgeyicide klasifikasyon doğruluğu hesaplanırken iki giriş ve daha çok girişi içeren indirgenmiş özellik kümeleri alınmıştır. Klasifikasyon doğruluğu hesaplanırken full ve nesne tabanlı ayırt edilemezliğin hesaplandığı değerlerin ortalaması alınmıştır. Johnson reducer ile %60.11'ik bir ortalama klasifikasyon doğruluğu hesaplanmıştır. Klasifikasyon doğruluğu değişken girişli yapay sinir ağı temelli sınıflandırıcı tarafından hesaplanmıştır. Aşağıda Johnson indirgeyici tarafından hesaplanan ve 20 girişli ürolojik veri tabanı kullanıldığı zaman elde edilen indirgenmiş özellik kümeleri gösterilmiştir. Johnson indirgeyici 10 girişli sistemi test için kullanıldığında yaklaşık %60'lık bir klasifikasyon doğruluğu elde edilmiştir. Johnson indirgeyicinin nesne tabanlı indirgeme sisteminde yapılan aşırı indirgeme klasifikasyon doğruluğunu azaltmaktadır.

İşlem zamanlarının testinde Core2Quad 3 GHz işlemci kullanılmıştır.

Johnson İndirgeyici tarafından hesaplanan indirgenmiş özellik kümelerinden bazıları şu şekildedir. $\{ a_1a_3a_4a_{15}a_{20}, a_1a_5a_{20}, a_1a_5a_{20}, \dots, a_4a_5, a_2a_{17}, a_1a_3 \dots vb. \}$ Tablo 5.1 de genetik algoritma ve kaba küme temelli indirgeme yazılımı tarafından hesaplanan indirgenmiş özellik kümeleri ve özellik (nitelik) bağımlılığı değerleri gösterilmiştir.

Tablo 5.2’de Modifiye Suni seçim Mekanizması Kromozomların Belirlenen Jenerasyonda Ortalama Özellik Bağımlılığı değerleri, Tablo 5.3 Test Edilen yazılım ortamına taşınan bazı algoritmalar ve İşlem Zamanları gösterilmiştir.

Tablo 5.4’de Tabloda belirtilen indirgeme yaklaşımlarına ait ortalama sınıflandırma doğruluğu, Tablo 5.5’de 10 Girişli Ürolojik Sistemde Hesaplanan İndirgenmiş Giriş değerleri (Özellik kümeleri) gösterilmiştir. Tablo 5.6’da 12 Girişli ürolojik veri için hesaplanan Bazı Reductlar (Genetik Algoritma ve Kaba Küme Kullanan Yazılımla Hesaplananlar) gösterilmiştir. Tablo 5.7’de Gen. Alg. ve Kaba Küm. Kul. Sistem ver 2, Karara Bağlı Ayırt Edici Matris ve Johnson İndirgeyici Sınıflandırma Doğrulukları, Sensivite ve Spesifte Değerleri (12 girişli ürolojik veri için)

Tablo 5.2 Modifiye Suni seçim Mekanizması Kromozomların Belirlenen Jenerasyonda Ortalama Özellik Bağımlılığı değerleri

Ara Nesildeki Kromozom	ModifiyeSuniseçim Gen Hav. 6 Jen.	Modifiye Suniseçim Gen Hav.8 Jen.	Suni Seçim
1	0,82	0,85	0,6
2	0,78	0,84	0,55
3	0,7	0,75	0,5
4	0,65	0,68	0,4
5	0,55	0,59	0,38
6	0,45	0,47	0,36

Tablo 5.3 Test Edilen Algoritmalar ve İşlem Zamanları

	TEST EDİLEN SİSTEM YAZILIMI		Giriş Sayısı	Ortalama Zaman	İşletim Sist. (En Yüksek Çalışma Küm.)
1	GARSBS (Genetik Algoritma ve Kaba Küme Temelli İndigeme Yaz.)	Suni seçim	20	2 dk- 30 dk.	90-350 MB.
		2 Jenerasyonlu Mod. Suni Seçim	20	1.5 dk -20 dk.	80- 250 MB.
		6 Jenerasyonlu Mod.suni seçim	20	1 dk. – 8 dk.	60 – 150 MB.
		8 Jenerasyonlu Mod. Suni seçim	20	40 sn – 7 dak.	50-150 MB
		Suni seçim	10	1 dk.-6 dk.	40 MB.-80 MB.
		6 Jenerasyonlu Mod Suni seçim	10	30 sn – 4 dak.	35 MB. 70 MB.
2	Karara Bağlı Ayırt Edici Fonksiyon ve Matris Kullanan Yaklaşım		20	4 Saati Geçiyor	800 MB. Tahsis Edilen Değeri Aşıyor. Desteklemiyor.
			10	10 dk.-25 dak.	150 MB.-500 MB.
3	Hızlı İndirgenmiş özellik kümesi yaklaşımı		20	Aşırı Hafıza Kullanımından Dolayı Hafıza Hatasına Sebep Oluyor.	
			10	5 dk.- 20 dk.	380 MB. -500 MB.
4	Ters İndirgenmiş özellik kümesi yaklaşımı		20	Aşırı Hafıza Kullanımından Dolayı Hafıza Hatasına Sebep Oluyor.	
			10	6-20 dk.	370 MB. -550 MB
5	Bağıl Bağımlılık Metodu		20	Aşırı Hafıza Kullanımından Dolayı Hafıza Hatasına Sebep Oluyor.	
			10	10-20 dk.	200 - 350 MB.

TABLO 5.4. Test Edilen Yaklaşımlara Ait Yazılımların Ortalama Sınıflandırma Doğrulukları

	Test Edilen Sistem	Ortalama Sınıflandırma Doğruluğu (20 Girişli Ürolojik Veritabanı 1)	Ortalama Sınıflandırma Doğruluğu (10 Girişli Ürolojik Veri Tabanı 2)
1	GARSBS	%90	%90
2	Johnson İndirgeyici	%60	% 55
3	Karara Bağlı Ayırt Edici Matris	Giriş Sayısını Desteklemiyor	% 75
4	Hızlı Reduct	Giriş Sayısını Desteklemiyor	% 70
5	Ters Reduct	-	% 70
6	Bağıl Bağımlılık Metodu	-	% 65

TABLO 5.5 10 Girişli Ürolojik Sistemde Hesaplanan Reductlar (İnd. Özellik Kümeleri)

	Test Edilen SistemYazılımı	Bazı İndirgenmiş Özellik Kümeleri
1	GARSBS	$X_1X_2X_3X_4X_5, X_1X_2X_3X_6, X_1X_2X_3X_5X_6, \dots, X_1X_3X_5X_6X_8$
2	Karara Bağlı Ayırt Edici Matris Yaklaşımı	$f_d = X_1X_3X_4X_5X_6X_9 + \dots + X_1X_3X_4X_6X_7X_9 + X_3X_4X_7X_8X_9X_{10}$
3	Hızlı Reduct	$X_1X_3X_4X_5X_6X_9$
4	Ters Reduct	$X_2X_3X_4X_5X_7X_9X_{10}$
5	Göreceli Bağımlılık	$X_3X_4X_7X_8X_9X_{10}$
6	Johnson Reducer (Nesne tabanlı ve tam ayırtedilemezlik yaklaşımı)	$X_1X_3X_4X_5X_6X_9, X_1X_7X_{10}, X_6X_9, \dots, X_2X_7, X_3X_5X_8X_{10}$

Tablo 5.6 12 Girişli Sistemde hesaplanan Bazı Reductlar(Genetik Algoritma ve Kaba Küme Kullanan Yazılımla Hesaplananlar

Eleman Sayısı	İndirgenmiş Özellik Kümeleri	Özellik Bağımlılığı Değeri
7	$a_1 a_2 a_3 a_5 a_6 a_{11} a_{12}$	1
	$a_2 a_3 a_4 a_8 a_9 a_{11} a_{12}$	1
6	$a_1 a_2 a_3 a_7 a_9 a_{10}$	1
6	$a_1 a_2 a_3 a_5 a_8 a_{12}$	1
6	$a_1 a_3 a_4 a_5 a_8 a_{10}$	1
6	$a_2 a_3 a_5 a_8 a_9 a_{12}$	1
6	$a_1 a_3 a_6 a_8 a_9 a_{10}$	1
6	$a_1 a_2 a_3 a_7 a_9 a_{10}$	1
6	$a_1 a_2 a_3 a_7 a_9 a_{10}$	1
6	$a_1 a_2 a_3 a_5 a_8 a_{12}$	1
6	$a_1 a_3 a_4 a_5 a_8 a_{10}$	1
6	$a_2 a_3 a_5 a_8 a_9 a_{12}$	1
6	$a_1 a_3 a_6 a_8 a_9 a_{10}$	1
6	$a_2 a_3 a_7 a_8 a_9 a_{10}$	0,975
6	$a_2 a_3 a_4 a_8 a_{10} a_{12}$	0,967
6	$a_2 a_3 a_4 a_8 a_{10} a_{11}$	0,975
6	$a_2 a_3 a_4 a_7 a_9 a_{11}$	0,983
6	$a_2 a_3 a_4 a_8 a_{10} a_{11}$	0,975
5	$a_1 a_2 a_3 a_8 a_{10}$	1

Karara bağlı ayırt edici matris kullanan yaklaşımdaki indirgenmiş özellik kümeleri (reductlar) aşağıdaki gibi hesaplanır. Karara bağlı ayırt edici fonksiyon kullanan yaklaşımda oluşturulan discernibility fonksiyonu aşağıdaki gibidir. Fonksiyon geliştirilen karara bağlı ayırt edici matrisi kullanan test yazılımı ile indirgenmiştir.

Genetik algortima ve Kaba Küme Kullanan Yazılım, ürolojik giriş verilerini indirgemektedir. Burada a_1 , maksimum akış hızı, a_2 , ortalama akış hızı a_3 rezidüyü göstermektedir. $a_1 a_2 a_3 a_8 a_{10}$ 'u içeren indirgenmiş özellik kümesindeki giriş değerleri

olan $a_1, a_2, a_3, a_8, a_{10}$ maksimum akış hızı, ortalama akış hızı, rezidü miktarını a_8 ve a_{10} anlık üroflowmetrik akış değerlerini göstermektedir.

Test işlemlerinde 12 giriş değişkenli ürolojik data kullanılmıştır. Nitelik İndirgeme Sistemlerinde giriş sayısı azaltıldığında YSA temelli işleme zamanları önemli ölçüde azalmaktadır. Yapay Sinir Ağları temelli sistemde işleme zamanı %70 oranında azalmaktadır.

Karara bağlı ayırt edici fonksiyon aşağıda gösterilmiştir ve aşağıdaki gibidir. 12 girişli ürolojik veri tabanının kullanıldığı sistemde +, “ $\cup$ ” birleşim operatörünü, “ $*$ ” ise kesişim operatörünü göstermektedir.

$$f_d = (a_1 + a_2 + a_3 + a_5 + a_6 + a_7 + a_8 + a_9 + a_{10} + a_{11}) * (a_1 + a_2 + a_3 + a_6 + a_8 + a_9 + a_{10} + a_{11} + a_{12}) * (a_1 + a_2 + a_4 + a_5 + a_7 + a_8 + a_9 + a_{10} + a_{11}) * (a_1 + a_2 + a_3 + a_6 + a_7 + a_8 + a_9 + a_{10} + a_{11} + a_{12}) * (a_1 + a_2 + a_3 + a_5 + a_6 + a_7 + a_8 + a_9 + a_{10} + a_{11}) * (a_1 + a_2 + a_4 + a_5 + a_6 + a_7 + a_8 + a_{10} + a_{11} + a_{12}) * (a_1 + a_2 + a_3 + a_5 + a_6 + a_7 + a_8 + a_9 + a_{10} + a_{11}) * (a_1 + a_3 + a_5 + a_6 + a_8 + a_{10} + a_{11}) * (a_1 + a_3 + a_4 + a_8 + a_{10}) * (a_1 + a_6 + a_8 + a_{10} + a_{11}) * (a_1 + a_3 + a_5 + a_7 + a_8 + a_{10}) * (a_1 + a_2 + a_3 + a_4 + a_5 + a_6 + a_7 + a_8 + a_9 + a_{10} + a_{11} + a_{12}) * (a_1 + a_2 + a_3 + a_4 + a_6 + a_7 + a_9 + a_{10} + a_{11}) * (a_1 + a_2 + a_3 + a_5 + a_7 + a_8 + a_9 + a_{10} + a_{11} + a_{12}) * \dots * (a_1 + a_2 + a_4 + a_5 + a_9 + a_{11} + a_{12}) * (a_1 + a_2 + a_3 + a_4 + a_5 + a_8 + a_9 + a_{12})$$

$$f_d = (a_2 + a_4 + a_6 + a_9 + a_{10} + a_{11}) * (a_1 + a_2 + a_5 + a_6 + a_7 + a_8 + a_{10}) * (a_1 + a_2 + a_5 + a_6 + a_7 + a_{10} + a_{11}) * (a_1 + a_4 + a_7 + a_8 + a_{10}) * (a_3 + a_8 + a_{10}) * (a_1 + a_4 + a_6 + a_9 + a_{11} + a_{12}) * (a_1 + a_5 + a_6 + a_8 + a_{11}) * \dots * (a_3 + a_5 + a_{11}) * (a_1 + a_6 + a_7 + a_9) * (a_2 + a_5 + a_7 + a_9 + a_{11}) * (a_1 + a_2 + a_4 + a_6 + a_7 + a_{10} + a_{11}) * (a_1 + a_3 + a_4 + a_6 + a_{11}) * (a_3 + a_{11} + a_{12}) * (a_8 + a_{10} + a_{11}) * (a_1 + a_9 + a_{10}) * (a_5 + a_9 + a_{10})$$

İkinci indirgemededen sona fonksiyon aşağıdaki gibi elde edilmiştir.

$$f_d = a_1 a_2 a_3 a_8 a_{10} + a_1 a_2 a_3 a_7 a_9 a_{10} + a_1 a_2 a_3 a_5 a_8 a_{12} + a_1 a_2 a_3 a_5 a_6 a_{11} a_{12} + a_2 a_3 a_4 a_6 a_{10} a_{12} + a_2 a_3 a_4 a_8 a_9 a_{12} + a_2 a_3 a_4 a_9 a_{11} a_{12} + a_2 a_3 a_6 a_7 a_8 a_{10} + a_2 a_3 a_6 a_7 a_{10} a_{11} + a_2 a_3 a_5 a_7 a_9 a_{10} + a_2 a_3 a_7 a_8 a_9 a_{10} + a_2 a_3 a_7 a_8 a_{10} a_{11} + a_2 a_3 a_6 a_8 a_{10} a_{12} + \dots + a_3 a_7 a_9 a_{10} a_{11} + a_3 a_7 a_9 a_{11} a_{12} + a_3 a_8 a_9 a_{10} a_{12} + a_3 a_8 a_9 a_{11} a_{12} + a_1 a_3 a_5 a_8 a_{11} a_{12} + a_1 a_3 a_5 a_8 a_{10} a_{11} + a_1 a_3 a_5 a_8 a_{10} a_{12} + a_1 a_3 a_8 a_{10} a_{11} a_{12} + a_1 a_3 a_7 a_8 a_{10} a_{12}$$

Johnson indirgeyici (reducer) full ve nesne tabanlı ayırt edilemezlik versiyonunda klasifikasyon doğruluğu %55 olarak test aşamasında elde edilmiştir. Kaba küme temelli temelli sistemlerde ortalama sınıflandırma doğruluğu %95'in üzerinde elde edilmiştir. Johnson indirgeyicide 2 ya da daha fazla indirgenmiş özellik kümesi olan indirgenmiş özellik kümeleri hesaplanmaktadır.

Johnson indirgeyici tarafından bulunan indirgenmiş özellik kümeleri aşağıda verilmiştir.

$a_3a_7a_9a_{10}a_{11}$, $a_1a_2a_9$, $a_9a_{11}a_{12}$, $a_1a_4a_9$, a_2a_{10} , ..., a_3a_4 , a_3a_{10} , a_2a_7 ,
 a_2a_{11}

Kaba küme ve genetik algoritma temelli indirgeme sisteminin hesapladığı, indirgenmiş özellik indirgeme yazılımından elde edilen sonuçlar aşağıdaki tabloda gösterilmiştir.

$a_4a_5a_6a_7... a_{12}$, 12 girişli ürolojik sistemdeki $T/4$ ile $3T/4$ arasındaki üroflowmetrik ölçüm değerlerini göstermektedir.

Tablo 5.7 Gen. Alg. ve Kaba Küm. Kul. Sistem ver 2, Karara Bağlı Ayırt Edici Matris ve Johnson İndirgeyici Sınıflandırma Doğrulukları, Sensivite ve Spesifte Değerleri

Test Edilen Sistem Yazılımı	Ortalama Klasifikasyon Doğruluğu %	Ortalama Duyarlılık (Sensitivite) %	Ortalama Spesifite (%)	PPV (%)	NPV(%)
Gen. Alg. ve Kaba Küm. Kul. Sistem ver 2	95	97	93	95	95
Karara Bağlı Ayırt Edici Matris	80	82	78	84	74
Johnson İndirgeyici	55	52	60	66	45

Lokal (Yerel) Optimuma Takılma Problemi

Rulet çemberi mekanizmasında yerel optimuma takılma problemi aşağıda verilmiştir.

481. GenerationChromosomes

=====

1. Chromosome : 2, 3, 5, 4, 1, Fitness Value = 0,510928961748634 ,
2. Chromosome : 5, 2, 1, 4, 3, Fitness Value = 0,510928961748634 ,
3. Chromosome : 4, 1, 5, 2, 3, Fitness Value = 0,510928961748634 ,
4. Chromosome : 2, 4, 1, 3, 5, Fitness Value = 0,510928961748634 ,
5. Chromosome : 3, 1, 2, 5, 4, Fitness Value = 0,510928961748634 ,
6. Chromosome : 4, 1, 2, 5, 3, Fitness Value = 0,510928961748634 ,
7. Chromosome : 3, 2, 5, 1, 4, Fitness Value = 0,510928961748634 ,
8. Chromosome : 1, 3, 4, 5, 2, Fitness Value = 0,510928961748634 ,

=====

From 481. Gen.

=====

1. Chromosome : 2, 3, 5, 4, 1, Fitness Value = 0,510928961748634 ,
2. Chromosome : 1, 3, 4, 5, 2, Fitness Value = 0,510928961748634 ,
3. Chromosome : 3, 2, 5, 1, 4, Fitness Value = 0,510928961748634 ,
4. Chromosome : 5, 2, 1, 4, 3, Fitness Value = 0,510928961748634 ,
5. Chromosome : 2, 3, 5, 4, 1, Fitness Value = 0,510928961748634 ,
6. Chromosome : 2, 3, 5, 4, 1, Fitness Value = 0,510928961748634 ,
7. Chromosome : 4, 1, 2, 5, 3, Fitness Value = 0,510928961748634 ,
8. Chromosome : 3, 1, 2, 5, 4, Fitness Value = 0,510928961748634 ,

=====

482. GenerationChromosomes

=====

1. Chromosome : 1, 4, 3, 5, 2, Fitness Value = 0,510928961748634 ,
2. Chromosome : 1, 3, 4, 5, 2, Fitness Value = 0,510928961748634 ,
3. Chromosome : 3, 1, 2, 5, 4, Fitness Value = 0,510928961748634 ,
4. Chromosome : 2, 1, 3, 4, 5, Fitness Value = 0,510928961748634 ,
5. Chromosome : 2, 3, 5, 4, 1, Fitness Value = 0,510928961748634 ,

6. Chromosome : 2, 3, 5, 4, 1, Fitness Value = 0,510928961748634 ,

7. Chromosome : 1, 2, 3, 4, 5, Fitness Value = 0,510928961748634 ,

8. Chromosome : 3, 1, 2, 5, 4, Fitness Value = 0,510928961748634 ,

5.2 Yeni Önerilen Modifiye Rulet Seçim Mekanizmasındaki Ekran Çıktısı

(Dışarıdan Aday Çözüm Kümesi Eklemeli)

Yeni Önerilen Modifiye Rulet seçim mekanizması dışarıdan belirtilen jenerasyonda ya da tekrarlamalı olarak belirtilen jenerasyonda bir sisteme dışarıdan (önceki jenerasyonlara ve ara nesile bağlı olmayarak) aday çözüm kümesi eklemeye çalışmakta, ekleme işlemini yaparken kullanıcı isterse kromozomları uygunluk değerlerine göre sıralayıp en kötülerden belirtilen sayıda kromozomu silmekte ve silinen kromozomlar yerine o jenerasyonda bulunmayan kromozomları eklemek için yazılım “n” kere jenerasyonda bulunmayan kromozomu üretmeye çalışmakta, üretmediği az girişli sistemlerin bulunduğu durumlarda n kere denedikten sonra en son ürettiği değeri aynı bile olsa üzerine yazmaktadır. Bu sistemin kilitlenmesini ve verimli çalışmasını sağlamak için eklenmiştir. Aynı işlem kromozomlar uygunluk değerlerine göre sıralanmadan en sondaki kromozomların silinmesiyle de yapılabilmektedir. Aşağıda 20. Jenerasyonda 8 kromozom içeren nesildeki 4 kromozom silinmiştir.

19. GenerationChromosomes

=====

1. Chromosome : 2, 1, 4, 8, 3, Fitness Value = 0,448087431693989 ,

2. Chromosome : 8, 2, 4, 3, 1, Fitness Value = 0,448087431693989 ,

3. Chromosome : 3, 1, 4, 2, 5, Fitness Value = 0,510928961748634 ,

4. Chromosome : 8, 3, 4, 1, 2, Fitness Value = 0,448087431693989 ,

5. Chromosome : 2, 4, 3, 1, 5, Fitness Value = 0,510928961748634 ,

6. Chromosome : 8, 3, 4, 1, 2, Fitness Value = 0,448087431693989 ,

7. Chromosome : 4, 2, 1, 3, 5, Fitness Value = 0,510928961748634 ,

8. Chromosome : 5, 4, 2, 3, 1, Fitness Value = 0,510928961748634 ,

From 19. Gen.

=====

1. Chromosome : 2, 4, 3, 1, 5, Fitness Value = 0,510928961748634 ,

2. Chromosome : 8, 3, 4, 1, 2, Fitness Value = 0,448087431693989 ,

3. Chromosome : 8, 3, 4, 1, 2, Fitness Value = 0,448087431693989 ,

4. Chromosome : 8, 3, 4, 1, 2, Fitness Value = 0,448087431693989 ,

5. Chromosome : 2, 1, 4, 8, 3, Fitness Value = 0,448087431693989 ,

6. Chromosome : 5, 4, 2, 3, 1, Fitness Value = 0,510928961748634 ,

7. Chromosome : 8, 2, 4, 3, 1, Fitness Value = 0,448087431693989 ,

8. Chromosome : 5, 4, 2, 3, 1, Fitness Value = 0,510928961748634 ,

20. GenerationChromosomes

=====

1. Chromosome : 4, 2, 3, 5, 1, Fitness Value = 0,510928961748634 ,

2. Chromosome : 4, 1, 2, 3, 5, Fitness Value = 0,510928961748634 ,

3. Chromosome : 5, 4, 2, 3, 1, Fitness Value = 0,510928961748634 ,

4. Chromosome : 8, 3, 1, 4, 2, Fitness Value = 0,448087431693989 ,

0. Chromosome : 16, 29, 4, 26, 1, Fitness Value = 0,508196721311475 ,

0. Chromosome : 20, 14, 17, 10, 15, Fitness Value = 0,5 ,

0. Chromosome : 6, 24, 22, 16, 29, Fitness Value = 0,483606557377049 ,

0. Chromosome : 31, 29, 19, 27, 16, Fitness Value = 0,314207650273224 ,

=====

From 20. Gen.

=====

1. Chromosome : 16, 29, 4, 26, 1, Fitness Value = 0,508196721311475 ,

2. Chromosome : 8, 3, 1, 4, 2, Fitness Value = 0,448087431693989 ,
3. Chromosome : 4, 2, 3, 5, 1, Fitness Value = 0,510928961748634 ,
4. Chromosome : 6, 24, 22, 16, 29, Fitness Value = 0,483606557377049 ,
5. Chromosome : 6, 24, 22, 16, 29, Fitness Value = 0,483606557377049 ,
6. Chromosome : 4, 1, 2, 3, 5, Fitness Value = 0,510928961748634 ,
7. Chromosome : 31, 29, 19, 27, 16, Fitness Value = 0,314207650273224 ,
8. Chromosome : 31, 29, 19, 27, 16, Fitness Value = 0,314207650273224 ,

=====

.....

25. GenerationChromosomes

=====

1. Chromosome : 1, 2, 4, 16, 29, Fitness Value = 0,519125683060109 ,
2. Chromosome : 29, 2, 24, 1, 16, Fitness Value = 0,483606557377049 ,
3. Chromosome : 1, 4, 3, 5, 2, Fitness Value = 0,510928961748634 ,
4. Chromosome : 16, 29, 2, 1, 4, Fitness Value = 0,519125683060109 ,
5. Chromosome : 16, 29, 4, 1, 2, Fitness Value = 0,519125683060109 ,
6. Chromosome : 16, 1, 3, 2, 4, Fitness Value = 0,655737704918033 ,
7. Chromosome : 29, 1, 4, 2, 3, Fitness Value = 0,461748633879781 ,
8. Chromosome : 16, 29, 4, 1, 2, Fitness Value = 0,519125683060109 ,

=====

UCI Dermatolojik veri tabanındaki veriler işlenmiş ve 33 giriş değeri istenen ve klavyeden girilen eşik değere bağlı olarak özellik bağımlılık değerleri 0,89 ile 1 arasında değişen reductlar üretebilmektedir. 33 giriş değeri 8 girişe indirgenmiştir. Bulunan Reductların YSA Sınıflandırma Doğruluğu 88% ile 98% arasında değişmektedir.

Rulet Çemberine eklenen yeni modifikasyon veri işleme süresini kısaltmakta ve klasik rulet çemberi mekanizmasının lokal optimum çözümlere takılmasını engellemektedir. Yazılıma istenen jenerasyon kadar çalışıp daha sonra otomatik olarak aday çözüm kümesinin giriş sayısını ekleme özelliği de eklenmiştir. Birçok indirgeme yazılımı

yüksek RAM kullanımı ve çok giriş sayısını desteklememektedir. Önerilen modifikasyonlu sistem efektif RAM kullanımı ve çok giriş sayısını destekleme gibi özellikleri ve farklı reduct'lar hesaplayabilme özelliği ile diğer indirgeme algoritmalarına göre oldukça avantajlıdır. Ayrıca sistem kilitlenmesini ve Ram kullanımını daha efektif hale getirebilmek için sisteme eklenen özellik sayesinde giriş sayısı belirli bir jenerasyon çalıştıktan sonra otomatik olarak arttırılmaktadır. Karara Bağlı Ayırt Edici Matris Kullanan Yaklaşım her bir altkümei hesaplayan test yazılımları geliştirilerek sistemle performansı karşılaştırılmıştır.



6. SONUÇLAR

Veri madenciliğinde yüksek giriş sayılarına sahip olan sistemlerin giriş sayılarının indirgenerek baskın girişlerin bulunması çok önemlidir. Baskın girişlerin tespit edilmesi ve çeşitli veri tabanlarında baskın giriş değerlerinin tespit edilmesi hangi ölçüm değerlerinin hangi hastalıkların teşhisi için önemli olduğunun ortaya çıkarılması için veri madenciliği, yapay zaka ve esnek hesaplama sistemlerinin kullanımı gün geçtikçe yaygınlaşmaktadır.

Bu tez çalışmasında geliştirilen genetik algoritma ve kaba küme temelli indirgeme yazılımında önerilen modifiye edilmiş seçim mekanizmaları sayesinde hem çok girişli medikal sistemlerin desteklenmesi sağlanmıştır. Veri madenciliği ve giriş nitelik sayılarını indirgeyen bir çok algoritma çok yüksek hafıza kullanımı nedeniyle yüksek giriş sayılarına sahip olan sistemleri desteklememektedirler. Geliştirilen yazılım sayesinde kullanıcı tarafından girilen özellik bağımlılığı seçim kriterine göre de genetik algoritma temelli indirgeme sistemi durdurtulabilmekte ve bir çok veri tabanına uyumluluk sağlanabilmektedir. Yapılan testlerde yüksek sınıflandırma doğruluğu elde edilmiştir. Etkin ve verimli hafıza kullanımı sağlanmış ve ayrıca yerel optimuma takılma problemi büyük ölçüde çözülmüştür. Geliştirilen yazılımla ve önerilen genetik algoritma seçim modifikasyonlarıyla veriler daha hızlı ve etkin bir şekilde işlenmiş ve çok girişli medikal sistemlerin de indirgemesi başarıyla yapılmıştır. Elde edilen sonuçlar deneysel sonuçlar kısmında verilmiştir.

7. KAYNAKLAR

- Hand D, Manilla H, Symth P, 2001, Principle of Data Mining, MIT Press, 2001, Cambridge England
- Kondo M. 2006, 'On the structure of generalized roughsets' , Information Sciences 176 598-600
- Herbert, J. P. ve Yao, J. 2009 'Criteria for choosing a roughsets model' Computers and Mathematics with Applications 57, 908-918.
- Pattaraintakorn ve Cercone 2008, 'Integrating Rough sets theory and medical applications', Applied Mathematics Letters 21, 400-403
- Guvenir , H. A., Demiroz G., Ilter N., 1998, 'Learning differential diagnosis of erythematous-squamous diseases using voting feature intervals' Artificial Intelligence in Medicine 13(3):147-65
- Shen L., Loh H. T., 2004 'Applying Rough Sets to market timing decisions', Decision Support Systems 37, 583-597
- Yao Y. Y., 1998, 'A comparative study of fuzzy sets and roughsets', Journal of Information Sciences 109, 224-242
- Johnson D. S. 1974, 'Approximation Algorithms for Combinatorial Problems', Journal of Computer and System Sciences 9, 265-378
- Øhrn A, Komorowski J, Skowron A, Synak , 1998, The Design and Implementation of a Knowledge Discovery Toolkit Based on Rough Sets: The ROSETTA System, In Rough Sets in Knowledge Discovery 1: Methodology and Applications, L. Polkowski and A. Skowron (eds.), Studies in Fuzziness and Soft Computing, Vol. 18, 1998, pp. 376-399.
- Øhrn A., Komorowski J., Skowron A., Synak P. 1998, The ROSETTA Software System, In Rough Sets in Knowledge Discovery 2: Applications, Case Studies and Software Systems, L. Polkowski and A. Skowron (eds.), Studies in Fuzziness and Soft Computing, Vol. 19, 1998 ,pp. 572-576.
- Levitin G, Rubinovitz J ve Shnits B., 2006, 'A genetic algorithm for assembly line balancing', Vol 168, Issue 3, 811-825
- Jinn-Yi Yeh*, Wen-Shan Lin, 2007 'Using simulation technique and genetic algorithm to improve the quality care of a hospital emergency department' Expert Systems with Applications 32 1073–1083
- Ghosh P ve Mitchell M. 2006, 'Segmentation of medical images using a genetic algorithm' GECCO (2006), Proceedings of the 8th annual conference on Genetic and evolutionary computation July 2006 Pages 1171–1178

Rennera G, Eka'rt A, 2003 'Genetic algorithms in computer aided design' Computer-Aided Design 35, 709–726

Mitra S, Acharya T., 2003, 'Data Mining Multimedia, Soft Computing and Bioinformatics, WileyInterscience, A John Wiley and Sons Publication, page 1-7

E. Oztemel, Yapay Sinir Aglari, Papatya, Istanbul, Turkey, 2006.

Z Şen 2004, Yapay Sinir Ağı ve İlkeleri

Z. Pawlak, 1997. "Rough set approach to knowledge-based decision support," European Journal of Operational Research, vol. 99, no. 1, pp. 48–57,

https://tr.wikipedia.org/wiki/Cantor_teoremi, 2022

Polkowski L., 2002 Rough Sets, Mathematical Foundations

Taşkın, Çağatan & Emel, Gül. (2002). Genetik Algoritmalar ve Uygulama Alanları. Uludağ Üniversitesi İİBF Dergisi. 21. 129-152.

M. Gen and R. Cheng, Genetic Algorithms & Engineering Optimization, John Wiley&Sons, New York, NY, USA, 2000.

<https://www.datascienceearth.com/genetik-algoritma-i/> erişim 2022.

https://user.engineering.uiowa.edu/~comp/Public/Gen_Alg.pdf, 2022)

Yao Y. Y., "A comparative study of fuzzy sets and rough sets," Information Sciences, vol. 109, no. 1–4, pp. 227–242, 1998.

Goldberg, D, 1989 Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley.

B. Makand T. Munakata, 2002 "Rule extraction from expert heuristics: a comparative study of rough sets with neural networks and ID3," European Journal of Operational Research, vol. 136, no. 1, pp. 212–229

G. Renner and A. Ekart, 2003. "Genetic algorithms in computer aided design," CAD Computer Aided Design, vol. 35, no. 8, pp. 709–726,

P. van Bommel, T. van der Weide, and C. Lucasius, "Genetic algorithms for optimal logical database design," Information and Software Technology, vol. 36, no. 12, pp. 725–732, 1994.

Schmitt, L. M., Nehaniv, C. L. and Fujii R. H., "Linear analysis of genetic algorithms," Theoretical Computer Science, vol. 200, no. 1-2, pp. 101–134, 1998.

T. Murata and H. Ishibuchi, 1994 "Performance evaluation of genetic algorithms for

flowshop scheduling problems,” in Proceedings of the 1st IEEE Conference on Evolutionary Computation, pp. 812–817, June 1994.

Noraini Mohd Razali, John Geraghty, 2011 Proceedings of the World Congress on Engineering 2011 Vol II, WCE 2011, July 6 - 8, London, U.K.

Braaten, Ø.. Rødningen, O. K Nordal, I. and. Leren, T. P 2000. “ The genetic algorithm applied to haplotypedata at the LDL receptor Computational and Mathematical Methods in Medicine 13 locus,” Computer Methods and Programs in Biomedicine, vol. 61, no. 1, pp. 1–9,

https://en.wikipedia.org/wiki/Stochastic_universal_sampling, 2022

E. A. Tanagho and D. Deng, “Smith’s general urology,” in Urodynamic Studies, E. A. Tanagho and J. W. McAninch, Eds., part 28, pp. 455–473, 17th edition, 2008.

Vesna, Z Milica L, Marina V, Andjelka S, and Lidija D, “Correlation between uroflowmetry parameters and treatment outcome in children with dysfunctional voiding,” Journal of Pediatric Urology, vol. 6, no. 4, pp. 396–402, 2010.

Jensen R , Shen Q 2008, Computational Intelligence and Feature Selection, Rough and Fuzzy Approaches, IEE press, 2008

Bhattacharjee K. ve Pant M. 2019, “Hybrid particle swarm optimization-genetic algorithm trained multi-layer perceptron for classification of human glioma from molecular brain neoplasia data” Cognitive Systems Research 58 (2019) 173–194

Ranjna Sirohi, Anupama Singh, Ayon Tarafdar, N.C. Shahi, 2018, “Application of genetic algorithm in modelling and optimization of cellulase production”, Bioresource Technology 270, 751–754

Lee C.K.H., 2018 “A review of applications of genetic algorithms in operations management” Engineering Applications of Artificial Intelligence 76 (2018) 1–12

Bruno Mota, Pedro Faria, Zita Vale 2022 * Residential load shifting in demand response events for bill reduction using a genetic algorithm Energy 260 (2022) 124978

Söke A, Bingül Z “Genetik algoritmaların farklı çaprazlama teknikleriyle iki boyutlu kesme problemlerine uygulanışı” Politeknik Dergisi (Journal of Polytechnic) Cilt: 7 Sayı: 1 s. 1-11, 2004

Han J, Sanches R, Hu X, 2005, Feature selection based on relative attribute dependency: an experimental study, RSFDGrC'05 Proceedings of the 10th international conference on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing - Volume Part I, Pages 214-223, 2005

EK

Ankara Üniversitesi Tıp Fakóltesi İbni Sina Hastanesi Üroloji Bölümüne ait olan ürolojik verilerin bilimsel amaçlar için Kürşat Zühtüoğulları tarafından kullanılmasına izin veriyorum.

A.Ü.Tıp Fak.

Üroloji Böl. Öğretim Üyesi

Prof. Dr. Nihat ARIKAN