



T.C.
KAHRAMANMARAŞ ST İMAM NİVERSİTESİ
FEN BİLİMLERİ ENSTİTS

İMMN PLAZMA ALGORİTMASI TEMELLİ YENİ
BİR ZNİTELİK SEİMİ YNTEMİNİN
GELİŞTİRİLMESİ

ZER OĖUZ

YKSEK LİSANS TEZİ
ENFORMATİK ANABİLİM DALI

KAHRAMANMARAŞ 2023

T.C.
KAHRAMANMARAŞ SÜTÇÜ İMAM ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

İMMÜN PLAZMA ALGORİTMASI TEMELLİ YENİ
BİR ÖZNİTELİK SEÇİMİ YÖNTEMİNİN
GELİŞTİRİLMESİ

ÖZER OĞUZ

YÜKSEK LİSANS TEZİ
Enformatik Ana Bilim Dalı

KAHRAMANMARAŞ 2023

TEZ BİLDİRİMİ

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

Özer OĞUZ



Not: Bu tezde kullanılan özgün ve başka kaynaktan yapılan bildirişlerin, çizelge, şekil ve fotoğrafların kaynak gösterilmeden kullanımı, 5846 sayılı Fikir ve Sanat Eserleri Kanunundaki hükümlere tabidir.

İMMÜN PLAZMA ALGORİTMASI TEMELLİ YENİ BİR ÖZNİTELİK SEÇİMİ YÖNTEMİNİN GELİŞTİRİLMESİ (YÜKSEK LİSANS TEZİ)

ÖZER OĞUZ

ÖZET

Geleneksel yaklaşımda makine öğrenmesi yöntemlerinde veri ön işleme tabii tutularak sınıflandırılmaktadır. Bu yaklaşımda uzman tasarım özniteliklerin problem temsil etme kabiliyeti her zaman arzu edildiği gibi olmayabilir. Çağdaş yaklaşımda bu sınırlılığın ortadan kaldırılabilmesi için sınıflandırmadan önce veri öznitelik seçim yöntemleri kullanılmaktadır. Öznitelik seçimi için istatistiksel yöntemler ve küresel arama yeteneğine sahip olan optimizasyon algoritmaları kullanılmaktadır. Optimizasyon algoritmalarının istatistiksel yöntemlere göre nispeten daha başarılı sonuçlar ürettiği bilinmektedir. Yakın zamanda COVID pandemisi sürecinde hastalara uygulanan plazma tedavisi sürecini modelleyen ve nümerik optimizasyon problemlerinde oldukça başarılı olduğu kanıtlanan yeni İmmün Plazma Algoritması (IPA) sunulmuştur. Bu nedenle öznitelik seçim problemlerinde IPA'nın yenilikçi sonuçlar vereceği kabul edilebilir. Bu tezde IPA temelli yeni bir öznitelik seçimi yöntemi IPAbFS önerilmiştir. Önerilen IPAbFS düşük-yüksek öznitelikli ve düşük-yüksek örneklemli 39 veriseti üzerinden en yeni 11 farklı rakip algoritma ile karşılaştırılmıştır. Önerilen IPAbFS ile gerçekleştirilen deneysel sonuçlar IPAbFS'nin problem temsil etme kabiliyeti yüksek verileri seçmekte oldukça başarılı olduğunu göstermiştir.

Anahtar Kelimeler: Öznitelik seçimi, İmmün Plazma Algoritması, makine öğrenmesi, KNN, sınıflandırma

Kahramanmaraş Sütçü İmam Üniversitesi
Fen Bilimleri Enstitüsü
Enformatik Ana Bilim Dalı, Mayıs/ 2023

Danışman: Dr. Öğr. Üyesi Hasan BADEM

Sayfa sayısı: 78

**DEVELOPMENT OF A NEW FEATURE SELECTION METHOD BASED ON
IMMUN PLASMA ALGORITHM
(M.Sc. THESIS)**

ÖZER OĞUZ

ABSTRACT

In the conventional approach, data is classified by preprocessing in machine learning methods. In this approach, the ability of expert design features to represent problems may not always be as desired. In order to remove this limitation, data feature selection methods are utilized before classification in the contemporary approach. Statistical methods and optimization algorithms with global search capability are used for feature selection. It is known that optimization algorithms produce relatively more successful results than statistical methods. More recently, a new Immune Plasma Algorithm (IPA) which models the plasma treatment process applied to patients during the COVID pandemic and proved to be very successful in numerical optimization problems has been presented. Therefore, it is acknowledged that IPA will yield innovative results in feature selection problems. In this study, a new IPA-based feature selection method IPAbFS was proposed. The proposed IPAbFS was compared with the latest 11 different competitor algorithms over 39 low-high-attribute and low-high-sampling datasets. The results of the experimental study conducted with the proposed IPAbFS showed that IPAbFS is quite successful in selecting data with high problem-representation capability.

Keywords: Feature selection, Immune Plasma Algorithm, machine learning, KNN, classification

Kahramanmaraş Sütçü İmam University
Graduate School of Natural and Applied Sciences
Department of Informatics, May / 2023

Supervisor: Assist. Prof. Dr. Hasan BADEM

Page number: 78

TEŞEKKÜR

Yüksek lisans öğrenimim süresince danışmanlığımı yapan ve çalışmalarım sırasında desteğini esirgemeyen, değerli danışman hocam Dr. Öğr. Üyesi Hasan BADEM'e çok teşekkür ederim. Tecrübelerinizle her zaman yolumu aydınlattınız hocam.

Bu yoğun süreçte her zaman destekçim olan sevgili eşim Sunay OĞUZ'a teşekkür ederim. Yoğun ve stresli zamanlarımda hep yanımda oldun. İyi ki varsın.

Çalışmalarım sırasında zamanından çalmak zorunda kaldığım sevgili oğlum Buğra OĞUZ'a teşekkür ederim. Her şey sana örnek bir baba olabilmek için...

Yakın zaman önce kaybettiğim sevgili dostum, abim, rehberim Mehmet DİVİLİ'ye teşekkür ederim. Senden öğrendiklerim birçok şeyi mümkün kıldı ve öyle olmaya devam edecek. Işıklar içinde uyu.

Yine tecrübeleriyle her zaman yanımda olan ve desteğini esirgemeyen değerli dostlarım Mehmet EMİR, Onur YOLCU, Melek YOLCU, Taner KILIÇ, Serkan KOLUÇOLAK, Umut KOÇOĞLU, Gülşah KOÇOĞLU, Suat BAYIR, Hacer OĞUZ TUNÇ, Ebubekir TUNÇ, Dr. Halil DUYMUŞ ve Ebru DUYMUŞ'a teşekkür ederim.

“Hayatta en hakiki mürşit ilimdir” sözüyle ve diğer tüm söz ve ilkeleriyle yoluma ışık tutan Ulu Önder MUSTAFA KEMAL ATATÜRK'e teşekkür ederim. Olmasaydın olmazdık....

Özer OĞUZ

İÇİNDEKİLER

Sayfa No

ÖZET	i
ABSTRACT	ii
TEŞEKKÜR	iii
İÇİNDEKİLER	iv
ŞEKİLLER DİZİNİ	vi
ÇİZELGELER DİZİNİ	vii
SİMGELER VE KISALTMALAR DİZİNİ	viii
1. GİRİŞ	1
2. GENEL BİLGİLER VE LİTERATÜR DERLEMESİ	4
2.1. Kuramsal Temeller	4
2.1.1. Yapay Arı Kolonisi Algoritması	4
2.1.2. Genetik Algoritmalar	5
2.1.3. Parçacık Sürü Optimizasyonu Algoritması	7
2.2. Literatür Derlemesi	8
3. MATERYAL ve METOT	12
3.1. Makine Öğrenmesi Tabanlı Sınıflandırma Yöntemleri	12
3.1.1. Karar ağacı	12
3.1.2. k-en yakın komşuluk	14
3.1.3. Destek vektör makineleri	15
3.2. İmmün Plazma Algoritması	16
3.2.1. Başlangıç popülasyonunun oluşturulması fazı	18
3.2.2. Enfeksiyon yayılımı fazı	18
3.2.3. Plazma transferi fazı	19
3.2.4. Donörlerin bağışıklık hafızasının kontrolü fazı	19
3.3. Önerilen İmmün Plazma Algoritması Temelli Öznitelik Seçimi Yöntemi	20
4. DENEYSEL SONUÇLAR ve TARTIŞMA	24
5. SONUÇLAR VE ÖNERİLER	57
KAYNAKLAR	58
ÖZGEÇMİŞ	66

ŞEKİLLER DİZİNİ

Şekil 3.1. Farklı karar ağaçları.....	12
Şekil 3.2. KNN algoritması için örnek.....	14
Şekil 3.3. Bir veri kümesinin (a) doğrusal ve (b) doğrusal olmayan doğrular ile birbirinden ayrılması.....	15
Şekil 3.4. Hiperdüzlemin belirlenmesi.....	16
Şekil 3.5. Genel sistem şeması.....	21
Şekil 3.6. Önerilen IPAbFS yönteminde özenitelik seçim modeli.....	23
Şekil 4.1. Önerilen IPAbFS-KNN yönteminin kullanılan verisetleri için tüm IPA parametlerindeki doğruluk değerleri.....	27
Şekil 4.2. Önerilen IPAbFS modelinin verisetlerindeki yakınsama grafikleri.....	46

ÇİZELGELER DİZİNİ

Çizelge 2.1. ABC temel adımları.....	5
Çizelge 2.2. GA temel adımları.....	7
Çizelge 2.3. PSO temel adımları.....	8
Çizelge 3.1. KNN temel adımları	14
Çizelge 3.2. IPA temel algoritması.....	17
Çizelge 4.1. Verisetlerine air özellikler.....	25
Çizelge 4.2. Makine öğrenmesi modelleri ile elde edilen deneysel sonuçlar	34
Çizelge 4.3. Her bir dataset üzerinde IPAbFS ile elde edilen yöntemlerin 10 çalıştırmada elde edilen ortalama çalışma süreleri.....	37
Çizelge 4.4. 10 bağımsız çalıştırma sonucunda elde edilen öznitelik sayıları	38
Çizelge 4.5. Önerilen IPAbFS yönteminde kullanılan KNN, DT, SVM sınıflandırıcıları arasında Wilcoxon istatistiksel analiz sonuçları.....	41
Çizelge 4.6. Önerilen IPAbFS ve rakip yöntemlere ait parametre ayarlama değerleri.....	43
Çizelge 4.7. Önerilen IPAbFS yönteminin literatür ile karşılaştırılması.....	52

SİMGELER VE KISALTMALAR DİZİNİ

Simgeler

s

Açıklama

: Saniye

Kısaltmalar

Açıklama

AAA	: Yapay Alg Algoritması
ABC	: Yapay Arı Koloni Algoritması
ACO	: Yapay Karınca Koloni Optimizasyon Algoritması
bAAA	: ikili Yapay Alg Algoritması
BEO	: İkili Denge Optimizasyonu
BEOSA	: İkili Denge Optimizasyonu Algoritması
BinHOA	: İkili At Sürüsü Algoritması
BMOChOA	: İkili Çok Amaçlı Şempanze Algoritması
BSOA	: İkili Martı Optimizasyon Algoritması
BWO	: Karadul Optimizasyon Algoritması
cSG	: Çapraz Salp Sürüsü ile Çekirge Optimizasyonu
DBOA	: Dinamik Kelebek Optimizasyon Algoritması
DE	: Diferansiyel Gelişim
FS	: Öznitelik Seçimi
GA	: Genetik Algoritma
GOA	: Çekirge Optimizasyon Algoritması
GWO	: Gri Kurt Algoritması
HBGWOHHO	: İkili Gri Kurt ve Harris Şahinleri Algoritması
HHO	: Harris Şahinleri Algoritması
HOA	: At Sürüsü Algoritması
IBEO	: Geliştirilmiş Denge Optimizasyonu
IPA	: İmmün Plazma Algoritması
IPAbFS	: İmmün Plazma Algoritması Temelli Öznitelik Seçimi
IRSA	: Geliştirilmiş Sürünge Arama Algoritması
ML	: Makine Öğrenmesi
MOCDE	: Çok Amaçlı Diferansiyel Gelişim
MPA	: Deniz Yırtıcıları Algoritması
MPPSO	: Çoklu Populasyon Temelli Parçacık Sürü Optimizasyonu
NoD	: Donör Sayısı
NoR	: Alıcı Sayısı
NRSIPSO	: Komşuluk Kaba Set ile Geliştirilmiş Parçacık Sürü Optimizasyonu
PS	: Popülasyon Sayısı
PSO	: Parçacık Sürü Optimizasyonu
RH-BPSO	: Kaba Hiperküoid Yaklaşımlı İkili Parçacık Sürü Optimizasyonu
RSA	: Sürünge Arama Algoritması
SA	: Tavlama Benzetimi
SDABWO	: Geliştirilmiş Versiyonlu Kara Dul Algoritması

SGOA
SOA

: Spiral Hareket Modlu Optimizasyon Algoritması
: Martı Optimizasyon Algoritması



1.GİRİŞ

Büyük verilerinden anlamlı örüntülerin çıkarılması için genellikle makine öğrenmesi (Machine Learning - ML) yöntemleri kullanılmaktadır. Fakat veri hacminin hızla artması verilerden anlamlı örtülerin çıkarılmasını zorlaştırmaktadır [1]. Bu durum büyük verilerin aynı zamanda ilgisiz ve gereksiz verileri de içermesinden kaynaklanmaktadır. Problemi güçlü şekilde temsil edemeyen ilgisiz veriler, makine öğrenmesi yöntemlerinin başarısını düşürmektedir. Sınıflandırma problemlerinde ise doğruluk değerlerini negatif yönde etkileyebilmektedir [2]. Öznitelik seçme (Feature Selection - FS) yöntemleriyle ilgisiz, gereksiz ve problemi güçlü bir şekilde temsil etmeyen verileri ayrıştırılabilmektedir [3]. FS ile ML yöntemlerinin etkinliği artırılabilir. FS ile verisetlerinin öznitelik boyutu azaldığı için ML yöntemlerinin hesaplama zamanları azalmaktadır [4].

FS yöntemlerinde yaygın olarak filtre ve sarmalayıcı temelli yöntemler kullanılmaktadır. Filtreleme temelli FS işlemi, temelde verilerin istatistiksel özelliklerinden faydalanmaktadır [5]. Dolayısıyla filtreleme yönteminde makine öğrenmesi yöntemleri kullanılmadığından hesaplama maliyetleri düşüktür [6]. Fakat, filtreleme temelli FS ile problemi ayrıştırabilir özniteliklerinde kaybedilmesine neden olabilmektedir. Bu nedenle, ML yöntemlerinin başarısının kısmen düşük kalabilmesine neden olabilir. Sarmalayıcı temelli FS yöntemlerinde temsil gücü en yüksek öznitelikleri elde edebilmek için makine öğrenmesi yöntemleri ile eğitim işlemi gerçekleştirilir. Makine öğrenmesi yöntemleri, en iyi sonuç elde edilene kadar farklı öznitelik kümeleriyle sürekli eğitilir [6], [7]. Bu nedenle sarmalayıcı temelli FS yöntemleri yüksek hesaplama maliyeti gerektirmektedir [8]. Sarmalayıcı temelli FS yöntemlerinde verisetindeki öznitelik sayılarını minimuma indirgeyip maksimum doğruluk değerine ulaşmak için bir optimizasyon süreci gerçekleştirir [9]. Dolayısıyla FS problemi bir optimizasyon problem olarak kabul edilebilir.

Etkili arama stratejilerine sahip olmalarından dolayı metasezgisel arama algoritmaları, FS problemlerinde sarmalayıcı yöntem olarak sıklıkla kullanılmaktadır. Literatürde Genetik Algoritma (Genetic Algorithm - GA) [10], Parçacık Sürü Optimizasyon Algoritması (Particle Swarm Optimization - PSO) [11], Yapay Arı Koloni Algoritması (Artificial Bee Colony Algorithm - ABC) [2],[12], Diferansiyel Gelişim Algoritması (Diferential Evolution Algorithm - DE) [13], Gri Kurt Optimizasyon Algoritması (Grey Wolf Optimization Algorithm - GWO) [14], Karınca Koloni Optimizasyon Algoritması (Ant Colony Optimization Algorithm - ACO) [15], Çekirge Optimizasyon Algoritması (Grasshopper Optimization Algorithm - GOA) [16] gibi metasezgisel algoritmaların FS problemlerinde

başarıyla uygulandığı rapor edilmiştir. Bu algoritmalarda benzer işlem basamakları ile FS işlemi gerçekleştirmektedir. Metasezgisel optimizasyon algoritmaları rastlantısal olarak belirli bir popülasyon büyüklüğünde çözüm oluştururlar. Bu çözümleri kendi işlem süreçlerinden geçirerek sürekli olarak geliştirirler. Algoritmaların ürettikleri çözümlerin kaliteleri fitness değerlerine göre belirlenmektedir. En iyi fitness değeri elde edildiğinde arama süreci durdurulur ve en iyi çözüm uzayı elde edilmiş olur. Optimizasyon algoritmalarının bu işlem adımları ile problemi en iyi temsil edilen öznitelikler elde edilmektedir.

Arslan ve Demirci tarafından 2020 yılında geliştirilerek literatüre sunulan İmmün Plazma Algoritmasının (IPA), nümerik optimizasyon problemlerinde yaygın kullanılan rakip metasezgisel algoritmalara göre oldukça başarılı sonuçlar verdiği deneysel olarak rapor edilmiştir [17]. IPA’da, tüm dünyayı etkisi altına alan Covid-19 salgını tedavisinde sıklıkla kullanılan immün plazma tedavisinden esinlenilmiştir [18]. IPA geliştirildiği günden bugüne kadar insansız hava araçlarının yol planlaması [19], bilişsel radyo ağlarında çok kullanıcı kanalı atama problemi [20], zaman serisi tahmin problemi [21] gibi problemlere başarıyla uygulanmıştır. Fakat gerçekleştirdiğimiz literatür taramalarında öznitelik seçimi problemlerinde IPA temelli bir yaklaşımın henüz geliştirilmediği görülmüştür. IPA’nın diğer problemlerin çözümündeki başarısı, IPA temelli FS yönteminin geliştirilmesiyle ML yöntemlerinin kullanıldığı non-lineer problemlerin daha etkin modellenebileceğini göstermektedir. Bu durum çalışmamızın ana motivasyonunu oluşturmuştur.

Bu tezde FS problemleri için IPA [17] temelli yeni bir yöntem önerilmiş (IPAbFS) ve simülasyon sonuçları sunulmuştur. Çalışmamızda FS problemlerinin çözümünde IPA’nın çözüm arama stratejisi kullanılmıştır.

Önerilen yöntem IPA’nın farklı parametreleriyle geliştirilen çözümler K En Yakın Komşuluk (K-nearest neighbor- KNN) makine öğrenmesi yöntemi ile sınıflandırılmıştır. Önerilen IPAbFS yöntemi, farklı makine öğrenmesi yöntemleri kullanıldığında oluşan performans durumlarını gözlemleyebilmek için KNN dışında Karar Ağaçları (Decision Tree - DT) ve Destek Vektör Makineleri (Support Vector Machine - SVM) makine öğrenmesi yöntemleriyle de sınıflandırma gerçekleştirilmiştir. Bu sınıflandırma işlemleri KNN ile elde edilen en iyi çözümlerin popülasyon büyüklüğü (PS), donör sayısı (NoD) ve alıcı sayısı (NoR) olmak üzere en iyi performansın elde edildiği IPAbFS parametre sayıları kullanılarak gerçekleştirilmiştir. Gerçekleştirilen bu sınıflandırmalara ait tüm sonuçlar çalışmamızda detaylıca rapor edilmiştir.

Bu tezin literatüre katkıları sırasıyla şu şekilde özetlenebilir:

Orijinal IPA sürekli değerler için geliştirilerek literatüre sunulmuştur. IPAbFS ile yeni bir ikili versiyon geliştirilmiştir.

Literatüre ilk defa IPA temelli FS yöntemi önerilmiştir.

Önerilen IPAbFS yöntemi farklı makine öğrenmesi yöntemleri ile test edilerek performansları karşılaştırılmıştır. Böylece IPA temelli FS yönteminde her bir veriseti için kullanılabilecek en iyi makine öğrenmesi yöntemi belirlenmiştir.

IPA'da PS, NoD ve NoR parametleri IPA'nın performansını etkilemektedir. Bu nedenle çalışmamızda her veriseti için en iyi parametrelerde belirlenerek rapor edilmiştir.

Deneysel çalışmada kullanılan verisetleri sağlık, fizik, biyoloji gibi farklı alanları kapsamaktadır. Verisetleri belirlenirken, önerilen IPAbFS başarımını değerlendirebilmek için düşük öznitelikli-yüksek öznitelikli ve düşük örnekleme-yüksek örnekleme sahip verisetleri kullanılmıştır. Bu sayede, önerilen IPAbFS'nin farklı alanlarda ve farklı boyutlardaki verilerdeki performansı belirlenmiş ve rapor edilmiştir.

Önerilen IPAbFS, literatürdeki farklı yaygın kullanılan rakip yöntemleri ile karşılaştırılarak IPAbFS'nin literatürdeki üstün ve zayıflıkları belirlenmiştir.

Bu tezin 2. bölümünde öznitelik seçimi ile ilgili literatür bilgisine yer verilmiştir. Önerilen IPAbFS yöntemi 3. bölümde detaylıca açıklanmıştır. Elde edilen deneysel sonuçlar ise 4. Bölümde tartışılmış ve sunulmuştur. Son olarak çalışmanın genel bir değerlendirmesinin verildiği sonuç bölümü 5. Bölümde yer verilmiştir.

2. GENEL BİLGİLER VE LİTERATÜR DERLEMESİ

2.1. Kuramsal temeller

Bu bölümde en popüler ve rakip metasezgisel algoritmalarından üç tanesi olan ABC, GA ve PSO detaylıca açıklanmaktadır.

2.1.1. Yapay arı kolonisi algoritması

Yapay Arı Koloni Algoritması (ABC) bal arılarının sürü sürüler halinde yiyecek arama davranışlarından ilham alınarak geliştirilmiş olan önemli metasezgisel optimizasyon algoritmalarından biridir. ABC, mühendislik [22], sağlık [23], robotik [24] vb. problemlerine uygulanmış ve başarılı sonuçlar verdiği rapor edilmiştir. ABC Derviş Karaboğa tarafından 2005 yılında geliştirilmiştir literatürde yerini almıştır [25]. Bir arı kolonisi işçi arı, kaşif arı ve gözcü arılardan oluşmaktadır. İşçi arı daha önce keşfedilen bir besin kaynağından yiyecek taşımakla görevlidir. Bir arı kolonisinde yeni bir yiyecek kaynağının bulunmasıyla yeni yiyecek kaynağını bulan arılar diğer arılara yiyecek kaynağının yerini dans etme davranışlarıyla bildirirler. Gözcü arılar dans edilen alanda bekleyerek yeni yiyecek kaynağını seçmekle görevlidirler. Kaşif arı ise koloni çevresinde rastgele gezinerek besin aramakla görevlidir. Her besin kaynağında sadece bir tane görevli arı bulunduğundan arı kolonisinde besin kaynakları sayısı olası çözümleri ifade etmektedir. Arı kolonilerinde nüfusun yarısı görevli arı iken diğer yarısı da gözcü arı olmaktadır. Besin kaynağını tüketen bir görevli arı kaşif arıya dönüşerek yeni besin kaynağı aramaya başlamaktadırlar. Besin kaynağının bulunduğu konum optimize edilecek problem için bir çözümü belirtir. Besin kaynağında bulunan yiyecek miktarı ise olası çözümün uygunluk değerini ifade etmektedir [26].

ABC’de ilk olarak başlangıç popülasyonu ile görevli arıların başlangıç konumları belirlenir. Görevli arılar tarafından bulunan besin kaynaklarında bulunan nektar miktarları ve besin kaynaklarının çevresinde bulunan olası besin kaynaklarında bulunan nektar miktarları hesaplanır. Hesaplanan nektar miktarlarına göre en çok nektar bulunan besin kaynağını görevli arılar gözcü arılara bildirirler. Bildirilen besin kaynaklarının bir tanesini gözcü arılar olasılık hesaplamalarına göre seçerek bu besin kaynağına giderler. Tüm gözcü arıların yiyecek bulamadığı durumda kendisine yiyecek bulan komşu gözcü arıların komşuluk durumları belirlenerek komşu bölgelerdeki besin miktarları hesaplanmaktadır. Tespit edilen yeni besin kaynakları gözcü arılara yeniden bildirilir ve böylece diğer gözcü arıların besin

kaynağı bulmalarına imkân tanınır. Besin arama işlemi tüm gözcü arıların besin kaynağı bulmaları sağlanana kadar devam etmektedir [25].

Bütün gözcü arıların yiyecek kaynaklarına dağıtılmasının ardından en uygun besin kaynağının bulunduğu konum hafızada tutularak besin kalitesiyle besinin miktarı yeterli olmayan besin kaynakları bırakılır. En uygun besin kaynağı optimizasyon probleminin çözümüne en yakın olan değerdir. Yetersiz görülerek bırakılan besin kaynaklarıysa problemin çözümüne en uzak olan değerlerdir [25].

Bırakılan besin kaynaklarındaki işçi arılar kâşif arılara dönüşerek besin kaynağı aramaya başlar ve kendisine besin kaynağı bulur. Bulunan tüm besin kaynaklarının uygunlukları değerlendirilerek durdurma koşulunun sağlanıp sağlanmadığı kontrol edilir. Durdurma kriterinin sağlanması durumunda işlemler tamamlanarak bitirilir. Durdurma koşulu sağlanmıyorsa belirlenen besin kaynaklarının komşuluk durumları taranarak yeni besin kaynakları bulunur. Algoritma bir döngü halinde bu adımlarla devam eder [25]. ABC'nin diğer rakip optimizasyon algoritmaları karşısında, basit, esnek ve kararlı olması, az sayıda parametreye ihtiyaç duyması, diğer algoritmalarla hibritleştirilmeye uygun olması nedeniyle nispeten daha avantajlı olduğu söylenebilir [26]. ABC'nin temel adımları Çizelge 2.1'de sunulmuştur.

Çizelge 2.1. ABC Temel Adımları [27]

Adım 1: Başlangıçta besin kaynağı olabilecek yerleri rastgele üret
REPEAT
Adım 2: Görevli arıları yiyecek kaynaklarına gönder
Adım 3: Gözcü arılar tarafından seleksiyon için kullanılacak olasılıkları, gözcü arılar tarafından elde edilen bilgilere göre hesapla
Adım 4: Gözcü arılar tarafından olasılıklara göre besin kaynağı bölgelerini seç
Adım 5: Tüketilen kaynakları bırak ve kâşif arıları üret
UNTIL maximum çevirim sayısına ulaşılması.

2.1.2. Genetik algoritmalar

Genetik Algoritmada (GA) doğadaki evrimsel mekanizma ve işleyiş model alınarak John H. Holland tarafından 1975 yılında geliştirilmiştir [28]. Genetik algoritmada evrim teorisi temel alınmaktadır ve en iyi çözüm ya da çözümler için evrim teorisinden esinlenilmiştir [29]. Bu yöntem evrim teorisindeki doğadaki güçlü canlıların hayatta kalması ve güçsüz canlıların ve özelliklerin elenmesi esasına dayanmaktadır. GA doğanın bu düzeninden faydalanarak

problemin çözümünü optimize etmeye çalışır [30]. Genler bir çift kromozom sarmalı halinde bulunur. Kromozom yapıları canlıların özelliklerini belirler. Bireylerde allellerden biri baskın iken diğeri pasif olabilmektedir. Popülasyonda yeni bir bireyin meydana gelebilmesi için rastgele iki tane birey seçilerek eşlenir. Eşlenme sırasında anneden ve babadan gelen genler çaprazlanarak yeni yavru bireyler meydana getirilir. Bu şekilde aynı türdeki bireylerin kromozom sayısının nesiller boyunca eşit kalması sağlandığı gibi birbirlerinden farklı yavru bireyler de oluşturulur. Bu yeni yavru bireyin güçlü bir birey olması beklenir. Güçlü bireyler güçsüz bireylerin yerini alarak özelliklerin diğer nesillere aktarılması sağlanır. Bu döngü en bireyin bulunmasına ya da durdurma koşulu sağlanana kadar devam etmektedir [31].

GA'da genler genetik algoritmanın parametre setini oluşturmaktadır [32]. Bu parametre setleri biyolojideki genleri ifade etmektedir. Tüm kromozomlar GA'da popülasyonu oluşturur. GA ile popülasyonun uygunluk değeri optimize edilmeye çalışılır. Daha güçlü olan yani uygunluk değeri daha yüksek olan genlerin bir sonraki nesile aktarılması sağlanmaktadır. Bu süreçte devamlı olarak en iyi çözümler tutularak diğer nesile aktarıldığı için problemin küresel en iyi çözümüne yakın bir çözüm bulunabilmektedir [29] [31]. Rastgele seçimle birleştirilen yeni bireylerden elde edilen yavru bireyler sayesinde popülasyondaki çeşitliliğin devamı sağlanmaktadır. Populasyondaki çeşitlilik kromozomların değişmesiyle de sağlanmaktadır [33]. Kromozomların değişmesi mutasyon olarak isimlendirilir.

GA'da her çözüm bir başka çözümle uygunluk değerlerine göre karşılaştırılmaktadır [34]. Doğada uygunluk değeri bir bireyin hayatta kalmak için en kadar güçlü olduğunu göstermektedir. Bir bireyin doğaya adaptasyonu ne kadar yüksekse hayatını devam ettirmesi ve neslini sürdürmesi şansı o kadar yüksek olmaktadır [28]. GA bu güçlü bireylerin özelliklerini diğer nesillere aktararak en iyi çözüme ulaşmayı hedefler. Uygunluk değeri yüksek olan bireyler popülasyonda bulunan diğer bireyler ile eşleşir. Böylece yeni ve daha güçlü yavru bireylerin meydana gelmesi sağlanmaktadır [28]. Uygunluk değeri düşük olan yani doğaya uyum sağlama yetenekleri düşük olan bireylerin eşleşme ihtimalleri düşük olduğundan, güçsüz bireylerin özelliklerinin diğer nesillere aktarılma şansları da düşük olmaktadır [35].

Temel GA yapısı ve temel adımları Çizelge 2.2'de sunulmuştur.

Çizelge 2.2. GA Temel Adımları [36]

Adım 1: Olası çözümleri içeren başlangıç popülasyonunu üret.

Adım 2: Popülasyondaki her çözüme karşılık gelen uygunluk değerini hesap et.

Adım 3: Durdurma koşulunu sağlıyorsa aramayı durdur.

Durdurma koşulu sağlanmıyorsa aşağıdaki adımları gerçekleştir.

3.1: Doğal seleksiyon işlemi gerçekleştir. (Uygunluk değerleri daha yüksek olan bireylerin (çözümlerin) yeni popülasyonda daha çok temsil edilmektedirler)

3.2: Çaprazlama işlemi gerçekleştir. (Mevcut iki yeni kromozomdan yeni bir kromozom üretilir)

3.3: Mutasyon işlemini gerçekleştir. (Çözümlerde rastgele olarak değişim meydana getirme işlemidir.)

Adım 4: Adım 2'ye git.

2.1.3. Parçacık sürü optimizasyon algoritması

Parçacık Sürü Optimizasyonu (PSO) algoritması, popülasyon temelli ve sürü zekâsına dayalı optimizasyon algoritmasıdır [27],[37]. PSO yöntemini Kennedy ve Eberhart, 1995-1996 yıllarında geliştirerek literatüre kazandırmışlardır [38]. PSO doğadaki canlıların özelliklerinden olan sosyal zekâ temel alınarak geliştirilmiştir [39].

PSO'da her bir parça popülasyonda bir bireyi temsil eder ve her bir parçanın hızı bulunmaktadır. Yeni nesillerdeki bireyler sahip oldukları hızlarını, en iyi bireyi baz alarak güncellerler. Bu şekilde yeni oluşan nesil önceki nesle göre daha güçlü duruma gelmektedir. Popülasyondaki tüm bireyleri en iyi bireye yakınsarken rastgele operatörler ile sonraki nesil bir önceki neslin en iyi bireyini geçme şansı yakalarlar [40].

PSO'da popülasyon bireylerden yani parçacıklardan meydana gelir, her bir birey yani parçanın çözüm uzayında hareket ettiği kabul edilir. Bu şekilde her bir parça bir çözümü temsil eder. Parçalar en iyi durumu hafızasında tutarak sonra hatırlayabilir ayrıca bu parçalar birbirleri arasında bilgi paylaşabilmektedirler [40]. Literatür incelendiğinde PSO'nun araştırmacılar tarafından birçok problemi çözmek için kullanıldığı görülmektedir [41],[42],[43],[44],[45]. PSO'da her bir parçacık diğer parçacıklardan aldıkları bilgilere göre optimum sonuç seviyesine doğru hızlandırmaktadırlar. Her yeni nesilde mevcut hız önceki neslin en iyi sonuçlarına göre güncellenerek yeniden hesap edilmektedir. Bu şekilde popülasyonu oluşturan parçacıklar (bireyler) iterasyon süresince daha iyi bir duruma ulaşırlar [46].

Temel PSO yapısı ve temel adımları Çizelge 2.3’de sunulmuştur.

Çizelge 2.3. PSO Temel Adımları [47]

- Adım 1: Başlangıç sürüsü ile başlangıç pozisyonlarını rastgele bir şekilde üret.
- Adım 2: Sürüdeki tüm parçacıkların fitness değerlerini hesapla.
- Adım 3: Mevcut jenerasyon içerisindeki en iyi parçacığı (pbest) bul.
- Adım 4: Mevcut jenerasyon içerisindeki yerel en iyi parçacıklar arasından küresel en iyi parçacığı (gbest) seç.
- Adım 5: Pozisyonlar ile hızları yenile.
- Adım 6: Durdurma şartı sağlana kadar 2,3,4 ve 5. adımlarını tekrarla.

2.2. Literatür derlemesi

Literatürde metasezgisel algoritmaların öznitelik seçme yöntemi olarak kullanılmasına yönelik çok sayıda çalışma bulunmaktadır. Bu çalışmalarda metasezgisel algoritmalar öznitelik seçim yöntemi olarak önerilmiş ve test edilmiştir. Test çalışmaları metasezgisel algoritmaların öznitelik seçim yöntemi olarak başarılı sonuçlar verdiğini göstermektedir. Bu bölümde, metasezgisel algoritmaların öznitelik seçme yöntemi olarak önerilen ve yakın zamanda literatüre kazandırılan çalışmalara yer verilmektedir.

Dhiman ve arkadaşları., martıların göç ve avlanma davranışlarından esinlenerek Martı Optimizasyon Algoritmasını (Seagull Optimization Algorithm – SOA) geliştirmişlerdir [48]. Araştırmacılar, geliştirdikleri bu yeni algoritmayı öznitelik seçme problemleri üzerinde test etmişlerdir. Bu çalışmada SOA’nın öznitelik seçme problemlerine uygulanabilmesi için İkili Martı Optimizasyon Algoritması (Binary Seagull Optimization Algorithm - BSOA) önerilmiştir [49]. BSOA’da öznitelikleri seçmek ya da elemek için transfer fonksiyonu yöntemi önerilmiştir. BSOA 14 farklı veriseti için öznitelik seçme yöntemi olarak uygulanarak sonuçlar elde edilmiştir. Elde edilen sonuçlar BSOA ile öznitelik seçimi için oldukça başarılı sonuçlara ulaşıldığını göstermektedir. Ayrıca BSOA, 7 farklı metasezgisel optimizasyon algoritması ile karşılaştırılmıştır. Karşılaştırma sonucunda BSOA ile 14 datasetten 12’sinde diğer metasezgisel yöntemlere göre daha iyi sonuçlar elde edildiği rapor edilmiştir.

At Sürüsü Optimizasyon Algoritması (Horse Herd Optimization Algorithm - HOA), MiarNaeimi ve arkadaşları tarafından 2021 yılında geliştirilen metasezgisel bir algoritmadır [50]. HOA, sürü zekasına dayalı bir algoritmadır. At sürülerindeki farklı yaşlardaki atların sosyal yaşamları taklit edilerek geliştirilmiştir. Elmanakhly vd., 2022 yılında

gerçekleştirdikleri çalışmalarında öznitelik seçme yöntemi olarak HOA temelli bir model önermişlerdir [51]. Araştırmacılar bu çalışmalarında öznitelik seçme ya da elemek için 8 farklı transfer fonksiyonu ile modellerini test etmişlerdir. Bu sayede araştırmacılar sürekli değerleri binary değerlere dönüştürebilmişlerdir. Bu 8 transfer fonksiyonundan en başarılı sonuçlara ulaşılan transfer fonksiyonu belirlenmiş ve raporlanmıştır. Ayrıca araştırmacılar HOA'nın arama kabiliyetini arttırmak için temel algoritmaya Levy Uçuşu operatörünü eklemişlerdir. Çalışmada önerilen yöntem diğer optimizasyon algoritması temelli öznitelik seçme modelleri ile karşılaştırılmıştır. Buna göre BinHOA ile test ve karşılaştırma için kullanılan 30 datasetten 27'sinde diğer yöntemlere göre daha başarılı sonuçlar elde edildiği rapor edilmiştir.

Deniz Yırtıcıları Algoritması (Marine Predators Algorithm - MPA), Faramarzi ve arkadaşları tarafından 2020 yılında önerilen sürü temelli yeni bir metasezgisel algoritmadır [52]. MPA deniz yırtıcılarının avlanma stratejilerinden ilham alınarak geliştirilmiştir. Elminaam vd., 2021 yılında yayınladıkları çalışmalarında öznitelik seçimi için MPA temelli yeni bir model önermişlerdir [53]. Araştırmacılar önerdikleri modelleri datasetler üzerinde test edebilmek için sınıflayıcı olarak KNN yöntemini kullanmışlardır. MPA-KNN yöntemi sağlık alanında iyi bilinen 18 farklı dataset üzerinde test edilmiştir. Çalışmada MPA-KNN yöntemi ile karşılaştırma için kullanılan diğer öznitelik seçme yöntemlerine göre daha başarılı sonuçlar elde edildiği rapor edilmiştir

Ghosh vd., 2020 yılında yayınladıkları çalışmalarında, öznitelik seçme problemleri için İkili Denge Optimizasyon Algoritması (Binary Equilibrium Optimization Algorithm -BEO) [54] ve Tavlama Benzetimi (Simulated Annealing – SA) [55] temelli yeni bir İkili Denge Optimizasyonu ve Tavlama Benzetimi isminde model (BEOSA) önermişlerdir [56]. Önerilen bu yöntemde öznitelik eleme işlemi için sürekli değerler V-şekilli transfer fonksiyonu ile ikili değerlere dönüştürülerek seçilen özniteliklerle KNN ile sınıflama işlemi gerçekleştirilmiştir. Araştırmacılar modellerini test etmek için 18 farklı dataset üzerinde BEOSA'yı uygulamışlardır. Elde edilen sonuçlar BEOSA'nın karşılaştırma için kullanılan diğer öznitelik seçme yöntemlerine göre başarılı sonuçlar verdiğini göstermektedir.

Karadul Optimizasyon Algoritması (Black Widow Optimization Algorithm - BWO) Hayyalolam ve Kazem tarafından 2020 yılında geliştirilen yeni bir metasezgisel algoritmadır [57]. BWO algoritması kara dul örümceklerinin çiftleşme davranışları taklit edilerek geliştirilmiştir. Hu vd., 2022 yılında yayınladıkları çalışmalarında BWO temelli yeni bir öznitelik seçme yöntemi önermişlerdir[58]. Araştırmacılar BWO'nun yerel minimuma erken

yakalanma gibi eksikliklerinden kurtulmak için BWO üzerinde bazı değişiklikler yaparak yöntemlerini SDABWO olarak adlandırmışlardır. Çalışmada SDABWO UCI'dan alınan 12 adet dataset üzerinde test edilmiştir. Test doğruluğu için KNN kullanılmış ve 5-Kat çapraz doğrulama ile daha güvenilir sonuçlar elde edilmeye çalışılmıştır. Çalışma sonucunda öznitelik seçme yöntemi olarak önerilen SDABWO'nun oldukça başarılı sonuçlar verdiği rapor edilmiştir.

Samemi vd., 2017 yılında doğadaki çekirge sürülerinin sürü zekasına dayanan davranışlarını matematiksel olarak modelleyip ve taklit ederek yeni bir metasezgisel algoritma olan Çekirge Optimizasyon Algoritması (GOA) geliştirmişlerdir [59]. Xu ve arkadaşları ise 2021 yılında GOA üzerinde yeni bir versiyon olan Spiral Hareket Modlu Çekirge Optimizasyon Algoritmasını (SGOA) geliştirmişlerdir[60]. Xu ve arkadaşları SGOA'yı öznitelik seçme problemleri üzerinde test etmişlerdir. Araştırmacılar daha güvenilir ve doğru sonuçlara ulaşabilmek için 10-kat çapraz doğrulama tekniğini kullanmışlardır. Yöntemde KNN ile sınıflanadırma işlemi gerçekleştirmişlerdir. Yine araştırmada sürekli değerlerin binary değerlere çevrilebilmesi için eşik değer 0.5 olarak belirlenmiştir. SGOA, 21 farklı dataset üzerinde uygulanmış ve sonuçlar analiz edilmiştir. SGOA ile 20 dataset üzerinde karşılaştırma için kullanılan diğer öznitelik seçme yöntemlerine göre daha başarılı sonuçlar elde edildiği rapor edilmiştir.

Kılıç vd., 2021 yılında yayınladıkları çalışmalarında yaygın kullanımı olan PSO algoritmasını yeni bir versiyonunu geliştirerek Öznitelik seçimi için Çoklu Populasyon Temelli İkili Paraçacık Sürü Optimizasyon algoritmasını (MPPSO) önermişlerdir [61]. Bu yöntemde çoklu populasyon rastgele ve Relieff tabanlı başlatma işlemleriyle üretilen başlangıç çözümleri ile başlamaktadır. Her iki populasyon da bu yöntemde çözüm uzayını aramakta kullanılmaktadır. Araştırmacılar öznitelik seçimi için önerdikleri MPPSO'yu test etmek için 29 farklı dataset üzerinde uygulamışlardır. Sınıflandırma için KNN'yi kullanarak, 10-kat çapraz doğrulama ile model test edilmiştir. Yöntem, literatürde aynı alanda yapılan diğer çalışmalardan elde edilen sonuçlarla karşılaştırılmıştır. Çalışma sonucunda MPPSO'nun öznitelik seçme problemlerinde 14 datasette literatürdeki diğer çalışmalara göre daha iyi sonuç elde ettiği rapor edilmiştir.

Yukarıda detaylı olarak açıklanan çalışmalara ek olarak Wang ve arkadaşları tarafından 2021 yılında Diferansiyel Gelişim Algoritması temelli Çok Amaçlı Diferansiyel Gelişim Algoritması (MOCDE) [62], Al-Wajih ve arkadaşları tarafından 2021 yılında Gri Kurt Algoritması (GWO) ve Harris Şahinleri Algoritması [63] (HHO) temelli hibrit bir Hibrit

İkili Gri Kurt ve Harris Şahinleri Algoritması [64] (HBGWOHHO) , Elgamal ve arkadaşları tarafından 2022 yılında Sürüngen Arama Algoritması [65] (RSA) bir versiyonu olan Geliştirilmiş Sürüngen Arama Algoritması [66] (IRSA),Turkoglu ve arkadaşları tarafından 2022 yılında Yapay Alg Algoritması [67] (AAA) temelli ikili Yapay Alg Algoritması [68] (bAAA) yöntemleri öznitelik problemlerine başarıyla uygulanmıştır. Bu nispeten yeni metasezgisel algoritmaları ile öznitelik seçme çalışmalarında oldukça başarılı sonuçlar alınmış ve çalışmalar literatüre kazandırılmıştır.

Bu tezde en yeni metasezgisel algoritmalarından biri olan IPA temelli yeni bir öznitelik seçimi modeli önerilmiştir. Önerdiğimiz modelde, sürekli değerler eşik değer olarak belirlediğimiz 0.5 sayısına göre ikili değerlere dönüştürülerek öznitelik seçme ve eleme işlemleri gerçekleştirilmektedir. Çalışma sonuçları detaylı bir şekilde analiz edilip raporlanarak literatüre kazandırılmıştır.

3.MATERYAL ve METOT

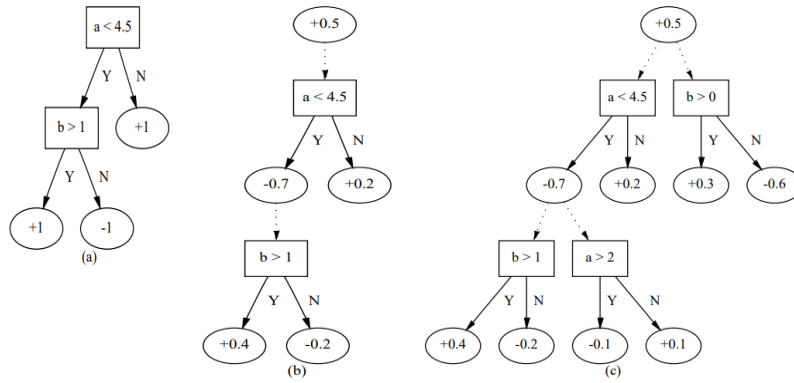
Bu bölümde öncelikle önerilen yöntemin arama işlemini gerçekleştirilen meta-sezgisel İmmün Plazma Algoritmasına yer verilmektedir. Sonrasında ise önerilen İmmün Plazma Algoritması temelli öznelik seçim yöntemi sunulmaktadır.

3.1. Makine Öğrenmesi Tabanlı Sınıflandırma Yöntemleri

Bu bölümde IPAbFS yönteminde kullanılan makine öğrenmesi yöntemlerine ait bilgiler sunulmaktadır.

3.1.1. Karar ağacı

Karar ağacı (Decision Tree – DT), ağaç yapısında bir makine öğrenmesi yöntemidir [69]. DT’de her bir öznelik düğüm biçimindeki yapılarla ifade edilmiştir. En sık kullanılan DT algoritmaları C4.5 [70] ve ID3 [71] algoritmalarıdır. Bu iki algoritma arasındaki en önemli fark C4.5 algoritması nümerik veriler kullanılırken, ID3 algoritmasında nümerik veriler kullanılmamaktadır[72].



Şekil 3.1. Farklı karar ağaçları [69]

Şekil 3.1’de 3 farklı karar ağacı gösterilmektedir. Şekil 3’te görülebileceği gibi ağaç yapısında düğümler öznelikleri temsil etmektedir. Ağaç yapısının diğer kısımlarını yapraklar ve dallar oluşturmaktadır. Ağaç yapısının en üst kısmı kök olarak ifade edilirken en son kısım yaprak olarak ifade edilmektedir. Kök yaprakların arasında kalan diğer kısımlar da dalları oluşturmaktadır [69]

Karar ağaçları oluşturulurken ağaçtaki dallanmanın hangi kriter ya da değerlere göre yapılacağı dikkat edilmesi gereken en önemli konuların başında gelmektedir. Bu problemi

çözmek için geliştirilen çeşitli yöntemler bulunmaktadır. Bu yöntemlerden bazıları bilgi kazanç oranı [69], Towing kuralı [73], Gini indeksi [74] ve Ki-Kare olasılık [75] yöntemleri olarak sıralanabilir. Bilgi kazanç oranı kuramında karar ağacı dallanma yapısının en verimli bir biçimde yapılması önem arz eder. Bu kuramda dallanma yapısının nasıl oluşturulacağının belirlenmesinde entropi kuralları kullanılmaktadır. Entropi bir sistemin rastgelelik ve düzensizliği şeklinde tanımlanabilir.

Bir verisetinin $C_1, C_2, \dots, C_n$ olarak birkaç sınıf etiketinden oluştuğunu ve T değerinin sınıf etiketlerine ait değerleri belirttiği düşünülürse, bir sınıfın olasılığı $P_i = C_i/T$ olur. Sınıfların entropisi ise Eşitlik 3.1 kullanılarak hesaplanmaktadır.

$$Entropi(T) = - \sum_{i=1}^n p_i \log_2 p_i \quad (3.1)$$

Verisetinin B özniteliğinin T sınıf değerleri $T_1, T_2, T_3, \dots, T_n$ olarak ayrıldığı varsayılırsa, B öznitelik değerleri ile T sınıf değerlerinin bölündüğünde sonuçta elde edilecek olan kazanç Eşitlik 3.2 ile hesaplanmaktadır.

$$Kazanç(B, T) = Entropi(T) - \sum_{i=1}^n \frac{|T_i|}{|T|} Entropi(T_i) \quad (3.2)$$

Ayrıca, T kümesindeki B özniteliğinin değerini belirlemek için Eşitlik 3.3 kullanılmaktadır.

$$Bölünme Bilgisi(B) = - \sum_{i=1}^k \frac{|T_i|}{|T|} \log_2 \left(\frac{|T_i|}{|T|} \right) \quad (3.3)$$

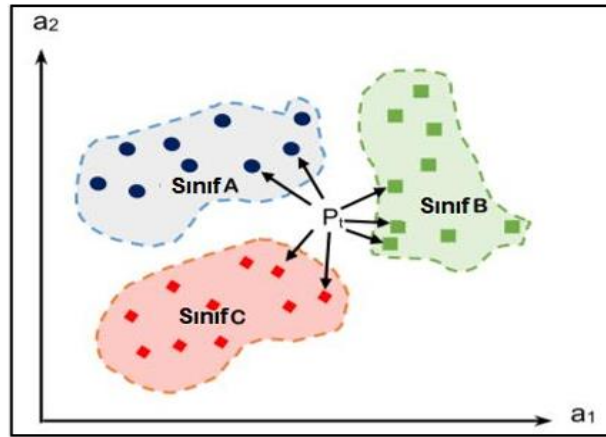
Böylelikle kazanç oranı değeri Eşitlik 3.4 kullanılarak hesaplanan ve sınıflandırma işleminde kullanılacak olan ayırma ile alınan bilgi oranı değerini verecektir. Bu eşitliğin kullanılmasıyla, ağacın tüm düğümlerinin kazanç oranının değeri en yüksek olacak şekilde T eğitim kümesi tekrarlı bir biçimde ayrılmaktadır. Bu işleme tüm yaprak düğümlerinin bir sınıfın gözlem değerini alması sağlandığında sonlandırılır [75].

$$Kazanç Oranı = \frac{Kazanç(B, T)}{Bölünme Bilgisi(B)} \quad (3.4)$$

3.1.2. k-en yakın komşuluk

k-en yakın komşuluk yöntemi, makine öğreniminde en çok kullanılan yöntemlerden biridir. Bir öğrenme aşaması gerektirmeyen örneklere dayanan sınıflandırma yöntemidir [76]. Örnek arasındaki mesafeyi ölçerek sınıflandırma yapar. Sınıfı bilinmeyen verinin, diğer verilere uzaklıkları hesaplanarak en yakın komşuların sınıflarına göre seçim fonksiyonu geliştirilen bir modeldir [77]. Yeni bir eleman sınıflandırılmadan önce, benzerlik ölçüsü kullanılarak, diğer öğelerle karşılaştırılır. Daha sonra bu elemanın k-en yakın komşularına bakılarak bir sınıfa dahil edilir [78].

KNN yönteminde öğrenme, eğitim setindeki diğer verilere göre gerçekleşir. Eğitim setine yeni bir veri örneği geldiği zaman, eğitim setinde bulunan diğer veri örnekleriyle yeni veri örneği arasındaki benzerlik durumuna göre sınıflandırma yapılmaktadır [79]. Bir verisetindeki verilere göre KNN yöntemi ile sınıflandırma işleminin temel adımları Çizelge 3.1’de, KNN algoritması için örnek bir figür ise Şekil 3.4’te sunulmuştur:



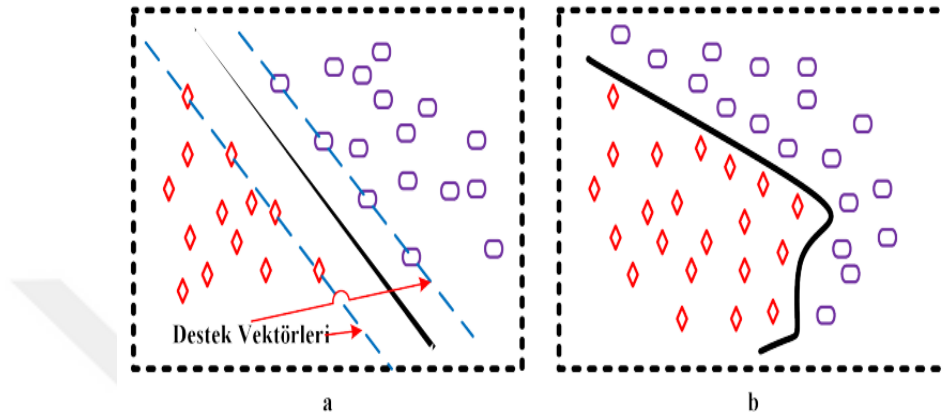
Şekil 3.1. KNN algoritması için örnek [80].

Çizelge 3.1. KNN temel adımları [81]

- | |
|---|
| <p>Adım 1: K değerini belirle.</p> <p>Adım 2: Sınıflandırılmak istenilen bir örneğin diğer veri örneği ile mesafesini uzaklık hesaplama fonksiyonuna göre hesap et.</p> <p>Adım 3: Hesap edilen uzaklıkları sırala, minimum mesafeye göre en yakın komşuları bul.</p> <p>Adım 4: En yakın komşuluk kategorilerini topla.</p> <p>Adım 5: En uygun olan kategoriye belirle.</p> |
|---|

3.1.3. Destek vektör makineleri

Destek vektör makineleri en sık kullanılan makine öğrenmesi yöntemlerinden biridir [82]. SVM ile sınıflandırma yapılabildiği gibi regresyon analizi de yapılabilmektedir [83]. SVM bir veri kümesindeki verileri doğrusal olan ya da doğrusal olmayan doğrular ile birbirinden ayırır. Bir veri kümesinin SVM ile birbirinden ayrılması işlemi Şekil 3.3'te gösterilmiştir.



Şekil 3.2. Bir veri kümesinin (a) doğrusal ve (b) doğrusal olmayan doğrular ile birbirinden ayrılması.

SVM ile sınıflandırma yapılırken, $[-1, +1]$ şeklinde sınıf etiketleri ile belirtilen iki sınıfa ait örnekler bulunur. SVM'de örnekler, eğitim verisi oluşturulan bir karar fonksiyonu yardımıyla birbirinden ayrılmaktadır. SVM ile sınıflandırma yaparken eğitim verilerini en doğru bir şekilde birbirinden ayırmak için bir hiperdüzlem oluşturulur. SVM'de bu işlem, bir karar fonksiyonu aracılığıyla yapılmaktadır. Hiperdüzlem oluşturulurken SVM'lerin amacı farklı veri kümelerindeki noktaların birbirleri arasındaki mesafeyi maksimum değere çıkarmaktır. Elde edilen bu hiperdüzlem optimum hiperdüzlem olarak adlandırılmaktadır[83].

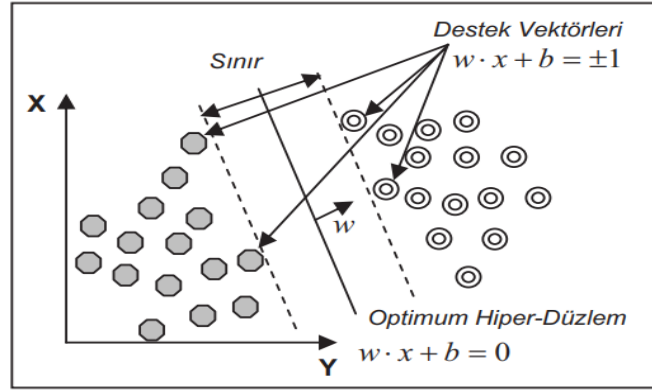
SVM için k adet örneklem içeren bir eğitim verisi $[x_i, y_i]$, $i=1, \dots, k$ olarak alındığında, en uygun hiperdüzlemin denklemleri, Eşitlik 3.5 ve Eşitlik 3.6 ile belirtilir:

$$wx_i + b \geq +1 \text{ her } y_i = +1 \text{ için} \quad (3.5)$$

$$wx_i + b \leq -1 \text{ her } y_i = -1 \text{ için} \quad (3.6)$$

Burada x reel bir sayı olup N -boyutlu bir uzayı göstermektedir. $y \in [-1, +1]$ ise sınıf etiket verilerini, b eğilim değerini, w ise ağırlık vektörünü belirtmektedir. Optimum hiperdüzlemin bulunabilmesi için Şekil 3.4'te görüldüğü gibi bu düzleme paralel ve sınırları

oluşturacak iki hiper-düzlemin belirlenmesi gerekir. Bu hiper-düzlemleri oluşturan kısımlar destek vektörleri olarak isimlendirilir ve bu düzlemler $w x_i + b = \pm 1$ şeklinde gösterilir [84].



Şekil 3.4. Hiperdüzlemin belirlenmesi [85].

SVM’de en uygun hiperdüzlemi bulabilmek için Eşitlik 3.11 kullanılmaktadır.

$$\min \left[\frac{1}{2} \|w\|^2 \right] \quad (3.7)$$

Eşitlik 3.7 için $y_i(w x_i + b) - 1 \geq 0$ ve $y_i \in \{1, -1\}$ olacak şekilde belirtilen kısıtların dikkate alınması gerekmektedir. Bu kısıtlamalara dikkat edilerek Lagrange yöntemi ile Eşitlik 3.8’de verildiği gibi düzenlenebilir [83].

$$L(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^k \alpha_i y_i (w x_i + b) + \sum_{i=1}^k \alpha_i \quad (3.8)$$

Sonuç olarak iki sınıftan oluşan bir problemde karar fonksiyonu olarak Eşitlik 3.9 ile hesaplanır [86].

$$f(x) = \text{sign} \left(\sum_{i=1}^k \lambda_i y_i (x \cdot x_i) + b \right) \quad (3.9)$$

3.2. İmmün Plazma Algoritması

2020 yılında Çin’in Vuhan kentinde ortaya çıkan koronavirüs hastalığı kısa sürede tüm dünyayı etkisi altına alarak küresel bir salgın haline gelmiştir. Hastalığa yakanan biryerlerin bağışıklık sistemlerinin güçlendirilmesi için en etkili yollardan biri immün plazma tedavisi olduğu görülmüştür. İmmün plazma tedavisinde daha önce koronavirüs hastalığı geçirmiş

kişinin kanından alınan antikorlar hasta bireye aktarılır. Böylece hastanın koronavirüs ile daha güçlü bir şekilde mücadele edebilmesi ve hastalığı daha kolay atlatması imkânını sağlar.

Aslan ve Demirci immün plazma tedavisi süreçlerinden esinlenerek IPA'yı geliştirmişlerdir [17]. IPA, immün plazma tedavisi aşamalarının modellenmesine dayalı metasezgisel bir küresel optimizasyon algoritmasıdır [17]. IPA algoritmasının kaba kodu Çizelge 3.2'de sunulmuştur. Çizelge 3.2 incelendiğinde IPA algoritmasının enfeksiyon yayılımı, plazma transferi ve donör güncellemesi fazlarından oluştuğu görülmektedir. IPA'nın 4 ana fazı aşağıda alt bölümlerde detaylıca açıklanmıştır.

Çizelge 3.2. IPA temel algoritması

Başlangıç :

PS, NoD, NoD ve t_{max} kontrol parametreleri ata.
Başlangıç popülasyonunu Eşitlik (3.10)'a göre oluştur.
Başlangıç popülasyonundan x_{best} yi seç.

REPEAT

Enfeksiyon Yayılımı:

Her birey için Eşitlik (3.11)'e göre x_k^{inf} , yi oluştur. t_{cr} yi bir artır.
Eğer $f(x_k^{inf}) < f(x_k)$ ise x_k yi x_k^{inf} ile güncelle.
Eğer $f(x_k^{inf}) < f(x_{best})$ ise x_{best} yi x_k^{inf} ile. güncelle

Plazma Transferi:

Popülasyondaki bireylerin uygunluk değerlerine göre alıcıları ve donörleri seç

Alıcılar için doz kontrolü vektörünü 1'e ayarla.

Her alıcı (x_k^{rcv}) için Eşitlik (3.13)'e göre x_k^{rcv-p} , yi oluştur. t_{cr} yi bir artır.

Eğer alıcının dozu 1 ise

Eğer $f(x_k^{rcv-p}) < f(x_m^{dnr})$ ise x_k^{rcv} yi x_k^{rcv-p} ile güncelle ve dozu 1 artır.

Değilse x_k^{rcv} yi x_m^{dnr} ile güncelle ve dozu 0 yap.

Değilse

Eğer $f(x_k^{rcv-p}) < f(x_m^{dnr})$ ise x_k^{rcv} yi x_k^{rcv-p} ile güncelle.

Değilse Dozu 0 yap.

Eğer $f(x_k^{rcv}) < f(x_{best})$ ise x_{best} yi x_k^{rcv} ile. güncelle.

Bireylerin Bağışıklık Hafızasının Kontrol Edilmesi:

Her donör için

Eğer $(t_{cr}/t_{max}) < rand(0,1)$ ise x_m^{dnr} 'i Eşitlik (3.14) ile güncelle, değilse Eşitlik (3.1) ile güncelle.

Eğer $f(x_m^{dnr}) < f(x_{best})$ ise x_{best} yi x_m^{dnr} ile. Güncelle.

UNTIL ($t_{cr} < t_{max}$)

3.2.1. Başlangıç popülasyonunun oluşturulması fazı

IPA’da popülasyondaki her birey problem için bir aday çözüme karşılık gelmektedir. Çözölmek istenilen problemin D adet farklı karar parametresi olduğu varsayıldığında PS popülasyonu, Eşitlik 3.10’da gösterilen denkleme göre oluşturulur. Bu eşitlikte x_k bir bireyi, x_{kj} , x_k bireyinin j. parametresini x_j^{min} ve x_j^{max} ise bireyin j’ninci parametresinin alt ve üst sınırlarının ifade etmektedir. $rand(0,1)$ bir ise 0 ile +1 arasında rastgele belirlenen bir sayıyı ifade etmektedir [17].

$$\begin{aligned} x_{kj} &= x_j^{low} + rand(0,1)(x_j^{high} - x_j^{low}) \\ k &= \{1,2, \dots, PS\} \text{ ve } j = \{1,2, \dots, D\} \end{aligned} \quad (3.10)$$

3.2.2. Enfeksiyon yayılımı ve bağışıklık sistemi yanıtı fazı

IPA’da bireylerin enfekte edilmesi aşamasında Eşitlik 3.11’e göre bireyler güncellenmektedir. Bu fazda IPA’nın araştırma alanı artırabilmek için enfeksiyonun popülasyondaki her bireye yayılımı gerçekleştirilmektedir. Eşitlikte, x_{kj}^{inf} enfekte edilen x_k bireyinin rastgele olarak seçilen j. parametresinin değerini ifade etmektedir. x_{mj} , popülasyondan rastgele olarak seçilen x_m bireyinin j’ninci parametresinin değeridir. IPA, x_k ile x_m bireyleri farklı bireyler olacak şekilde tasarlanmıştır. $rand(-1,1)$ bir ise -1 ile +1 arasında rastgele üretilen bir sayıyı ifade etmektedir [17].

$$x_{kj}^{inf} = x_{kj} + rand(-1,1)(x_{kj} - x_{mj}) \quad (3.11)$$

Enfekte olan bireylerin bağışıklık sistemi, bireyi hastalığa karşı korumak amacıyla bir bağışıklık sistemi geliştirir. Enfekte edilen bireyin bağışıklık cevabı uygunluk fonksiyonu ile değerlendirilir. x_k^{inf} bireyinin bağışıklık cevabı $f(x_k^{inf})$ ve x_k bireyinin bağışıklık cevabı $f(x_k)$ olmak üzere bağışıklık hafızası Eşitlik 3.12’ye göre güncellenir. Burada x_k^{inf} bireyinin enfeksiyona karşı verdiği cevap x_k bireyinin enfeksiyona karşı verdiği cevaptan daha küçük ise x_k bireyinin enfeksiyon yayılımı aşamasında değiştirilen parametre değeri güncellenir. Diğer durumda x_k bireyinin enfeksiyon öncesindeki parametre değeri aynı kalmaktadır [19].

$$x_{kj} = \begin{cases} x_{kj}^{inf} & \text{if } f(x_k^{inf}) < f(x_k) \\ x_{kj} & \text{if } f(x_k^{inf}) \geq f(x_k) \end{cases} \quad (3.12)$$

3.2.3. Plazma transferi fazı

İmmün plazma tedavisinde hastalığı geçirmiş güçlü bireylerden alınan antikorca zengin plazma sıvısı bağışıklığı zayıf olan hasta bireylere aktarılmaktadır. Böylece hastaların hastalıkla daha güçlü bir şekilde mücadele etmesi sağlanmaktadır. IPA’da bu süreç plazma tedavisi fazında modellenmiştir. Enfeksiyon yayılımı fazında bireyler enfekte edildikten sonra, IPA’da bağışıklık cevabı en iyi olan *NoD* (*Number of Donor*) adet bireyi donör olarak ve bağışıklık cevabı en kötü olan *NoR* (*Number of Receiver*) adet bireyi alıcı olarak belirlenir. *NoD* ve *NoR* sayıları IPA’nın başlangıç fazında sabit değer alan bir kontrol parametrelerdir. Donör ve alıcı bireyler belirlendikten sonra, plazma transferi aşaması gerçekleştirilir. Plazma transferi aşaması Eşitlik 3.13’de gösterilmektedir. Eşitlik 3.13’te x_k^{rcv} rastgele seçilen bir alıcıyı, x_m^{dnr} , rastgele seçilen bir donörü ve x_{kj}^{rcv-p} ise x_k^{rcv} bireyinin plazma tedavisi sonrası j’ninci parametresinin aldığı yeni değeri ifade etmektedir. Ayrıca x_{kj}^{rcv} ve x_{mj}^{dnr} sırasıyla x_k^{rcv} alıcısı ile x_m^{dnr} bireylerinin j’ninci parametre değerlerini ifade etmektedir. Plazma transferi aşamasında bütün alıcılara donörler arasından rastgele seçilen bir donörle plazma transfer edilmektedir [17].

$$x_{kj}^{rcv-p} = x_{kj}^{rcv} + rand(-1,1)(x_{kj}^{rcv} - x_{mj}^{dnr}) \quad (3.13)$$

Bir doz plazma transferi gerçekleştikten sonra (x_k^{rcv-p}) alıcı ve (x_m^{dnr}) donörünün bağışıklık cevaplarına bakılır. (x_k^{rcv-p}) bireyinin bağışıklık cevabı, (x_m^{dnr}) donörünün bağışıklık cevabından daha iyiye (x_k^{rcv}) bireyi elenerek yerine (x_k^{rcv-p}) bireyi geçirilir. Daha sonra plazma transferi 2,3,4,...,n kadar Eşitlik 3.13’e göre gerçekleştirilir. (x_k^{rcv-p}) bireyinin bağışıklık cevabı, (x_m^{dnr}) donörünün bağışıklık cevabından daha kötü ise (x_k^{rcv}) bireyinin yerine (x_m^{dnr}) bireyi gelerek transfer fazı sonlandırılır. IPA, plazma transferi aşamasının sonlandırılmasına (x_k^{rcv-p}) ile (x_k^{rcv}) bireylerinin bağışıklık cevabına göre karar verilecek şekilde tasarlanmıştır [18].

3.2.4. Donörlerin bağışıklık hafızasının kontrol aşaması fazı

Plazma tedavisinde donörlerin enfeksiyona karşı ürettikleri bağışıklık cevabı zamanla değişebilmektedir. Bu değişim IPA’da donörlerin bağışıklık hafızasının kontrol aşaması fazında modellenmektedir. IPA, donör güncelleme fazında donörlerin enfeksiyona karşı verdikleri bağışıklık cevaplarının zamana bağlı değişimi, toplam değerlendirme sayısı (t_c) ve maksimum değerlendirme sayısı (t_{max}) değerlerine göre belirlenir. Buna göre (t_c)/(t_{max}) değeri 0 ile 1 arasında rastgele olarak üretilen ve seçilen sayıdan küçük ise (x_m^{dnr}) donörü

Eşitlik 10'a göre tamamen güncellenir. Diğer durumda (x_m^{dnr}) donörün Eşitlik 3.14'e göre bütün parametreleri güncellenerek donörün bağışıklık hafızası kontrol edilir [17].

$$x_{mj}^{dnr} = x_{mj}^{dnr} + rand(-1,1)x_{mj}^{dnr} \quad (3.14)$$

Aslan ve Demirci, IPA'yı diğer popüler optimizasyon algoritmaları ile karşılatırmışlardır. Yapılan analizler IPA'nın Genetik Algoritma, , Yapay Arı Koloni Algoritması, Diferansiyel Gelişim Algoritması, Güve Alevi Optimizasyonu, Yerçekimi Arama Algoritması, Sinüs Kosinüs Algoritması, Salp Sürüsü Algoritması, Biyo-coğrafya Temelli Optimizasyon, Guguk Kuşu Arama Algoritması, Ateşböceği Algoritması, Yarasa Algoritması, Çiçek Tozlaşma Algoritması, Hayvan Göç Optimizasyonu, Karınca Aslanı Optimizasyonu, Martı Optimizasyon Algoritması, Benekli Sırtlan Algoritmas, Gri Kurt Optimizasyon Algoritması, Çoklu Evren Optimizasyonu ve Madde Arama Algoritması olmak üzere 19 iyi bilinen. optimizasyon algoritmalarına göre başarılı sonuçlar üretebildiğini göstermiştir. IPA'nın test edilen bu problemlerde oldukça başarılı sonuçlar elde ettiği rapor edilmiştir [17]. Fakat literatürde henüz IPA temelli bir öznitelik seçme versiyonu önerilmemiştir. IPA'nın nümerik optimizasyon problemlerindeki göstermiş olduğu başarı, öznitelik seçme probleminde de etkin sonuçlar üretebileceğini göstermektedir. Bu durum bizi IPA'yı bir öznitelik seçme yöntemi olarak test etmeye motive etmiştir. Bu çalışmada IPA'yı bir öznitelik seçme yöntemi olarak test edebilmek için İkili İmmün Plazma Algoritması önerilmiştir.

3.3. Önerilen İmmün Plazma Algoritması Temelli Öznitelik Seçim Yöntemi

Geleneksel olarak bir sınıflandırma problemini modellemek için geliştirilmiş veriseti, verinin durumuna göre gürültü azaltma, veri normalizasyonu işlemlerine tabi tutunularak sınıflandırma yöntemine uygulanmaktadır. Bu yaklaşımda, veri seti tasarımı uzman deneyimine bağlı olarak problemi temsil ettiği düşünülen tüm özellikler birer öznitelik olarak kullanılmaktadır. Fakat, uzmanın belirlediği özniteliklerin bazıları verinin ayrıştırılabilirliğine katkı sunmazken, bazı özniteliklerin varlığı ise model başarısını düşürebilmektedir. Özniteliklerden problemini temsil edenlere karar verilebilmesi için FS sürecinin geliştirilmesini gerekmektedir. FS sürecini içeren makine öğrenmesi temelli sınıflandırma modellerinin genel blok şeması Şekil 3.5'de görülmektedir.



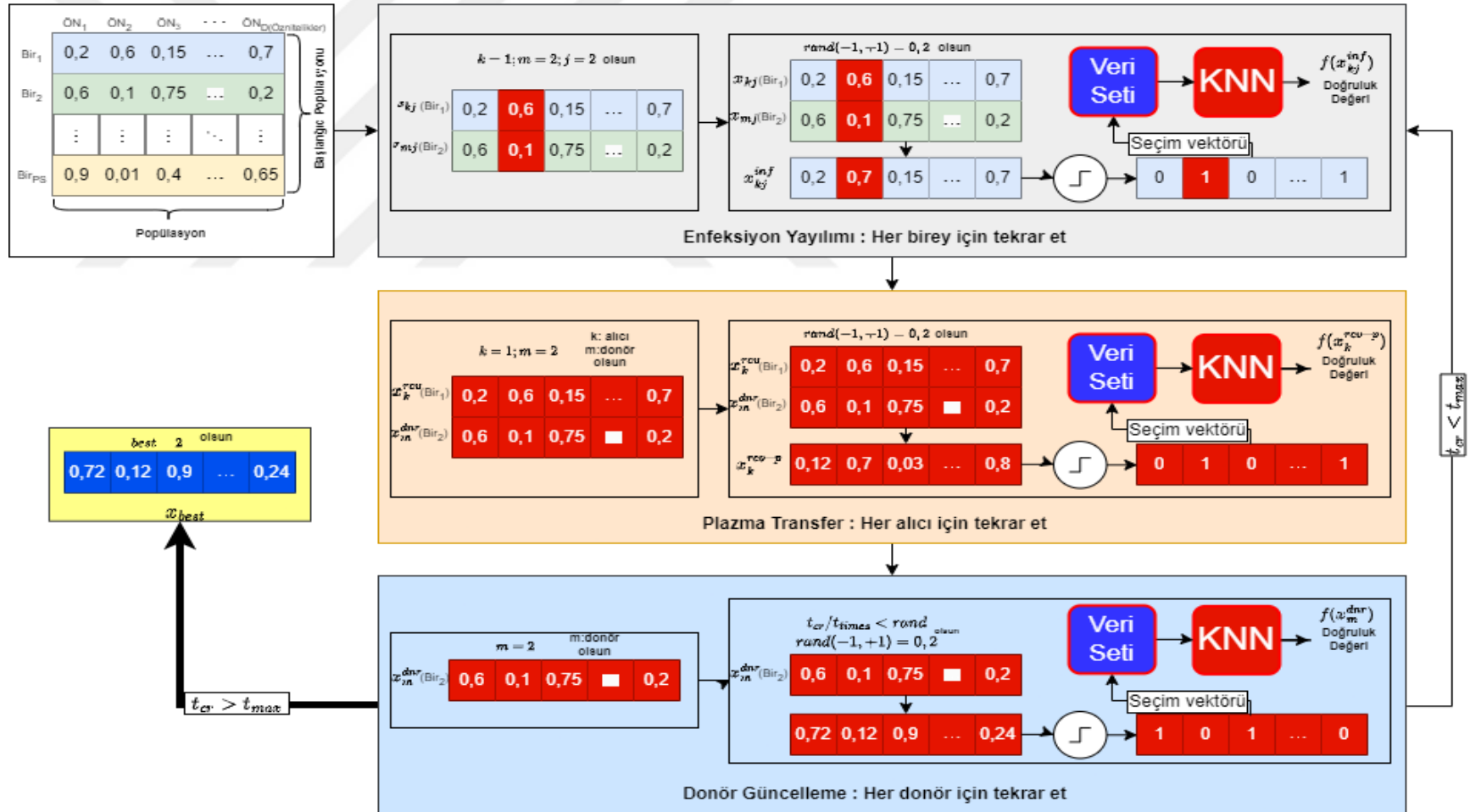
Şekil 3.5. Genel sistem şeması

Şekil 3.5’de görüldüğü üzere, ham veriseti, önışleme tabi tutulduktan sonra tüm veri FS süreci uygulanmaktadır. Seçilen öznitelikler ile sınıflandırma gerçekleştirilmektedir. Bu sayede, problemi temsil kabiliyeti olan öznitelikle ile daha düşük hesaplama maliyetli sistem elde edilmektedir. Bu çalışmada öznitelik seçim sürecinin performansını artırabilen IPA temelli bir FS yöntemi önerilmiştir. Önerilen İmmün Plazma Algoritması Temelli Öznitelik Seçimi Yöntemi (Immune Plasma Algorithm based Feature Selection – IPAbFS) yöntemi IPA algoritmasının nümerik optimizasyon problemlerindeki üstün başarısını, FS sürecine aktarmaktadır.

Önerilen IPAbFS yönteminde her birey $[0,1]$ arası rastgele D adet değerlerden oluşan bir başlangıç popülasyonu ile başlatılmaktadır. Burada D verisetinin öznitelik sayısına karşılık gelmektedir. Başlangıç popülasyonundaki her bireyin parametre değerlerinden, eşik eğer $0,5$ ’den büyük olanlara karşılık gelen öznitelikler ile alt bir veri kümesi elde edilir. Alt küme veri setleri, 10-kat çapraz doğrulama üzerinden KNN sınıflandırıcısı ile arama gerçekleştirilir. Sınıflandırma doğruluk değeri bireylerin uygunluk değerine karşılık gelmektedir. Sonra, IPA’nın enfeksiyon fazına karşılık gelen her birey (x_k) için popülasyondan rastgele bir birey seçilerek (x_m), rastgele seçilen bir parametresi arasında Eşitlik 3.11’e göre bir aday çözüm (x_k^{inf}) oluşturulmaktadır. Oluşturulan aday çözümün her bir parametresi yeniden $0,5$ değeri ile karşılaştırılır. $0,5$ değerinden büyük olan parametrelere karşılık gelen öznitelikler seçilir ve 10-kat çapraz doğrulama tekniği ve KNN sınıflandırıcısı ile bir arama işlemi gerçekleştirilir. Bu arama sonucu elde edilen sınıflandırma değeri enfeksiyona karşı bağışıklık sisteminin ürettiği cevabı yani uygunluk değerini temsil eder. Uygunluk değerlerine göre x_k^{inf} ile x_k arasında aç gözlü seçim gerçekleştirilmektedir. Daha sonra, enfeksiyon yayılımından gelen popülasyondaki alıcı olarak tanımlanan uygunluk değeri en düşük olan NoR adet birey (x_k^{rcv}) için donör olarak tanımlanan en iyi NoD adet bireyden x_{mj}^{dnr} Eşitlik 3.13’e göre aday çözüm x_k^{rcv-p} oluşturulur. Bu oluşturulan aday çözüm için de başlangıç popülasyonunun oluşturulması ve enfeksiyon yayılım fazlarındaki öznitelik seçimi yöntemi üretilen aday çözüme aynı şekilde uygulanır. Elde edilen uygunluk değerlerine göre bu çözümler arasında aç gözlü seçim uygulanmaktadır. Plazma transferi

işleminde tüm parametre uzayı arasında Eşitlik 3.13'e göre işlem yapılmaktadır. Daha sonra, plazma transferinden gelen popülasyondaki her bir x_m^{dnr} Eşitlik 3.14'e göre güncellenir ve öznitelik seçimi yöntemi işleminden aynı şekilde geçerek uygunluk değeri hesaplanır. En sonunda, eğer maksimum değerlendirme sayısına ulaşılmamışsa, donör güncelleme fazından gelen popülasyon tekrardan enfeksiyon yayılımına giriş olarak uygulanmaktadır. Tüm bu süreç maksimum değerlendirme sayısına ulaşılan kadar tekrar edilir. Önerilen IPAbFS işlem adımları Şekil 3.6'da görselleştirilmiştir. IPbFS ile elde edilen en iyi bireyin temsil ettiği öznitelik seçim vektörü üzerinden verisetinin etkin özniteliklerine karar verilmektedir.





Şekil 3.6. Önerilen IPAbFS yönteminde öz nitelik seçim modeli

4. DENEYSEL SONUÇLAR VE TARTIŞMA

Tüm deneysel çalışmalar, AMD Ryzen 5 2600X Six-Core Processor 3.60 GHz işlemci ve 16 GB Ram'a sahip bir sistemde gerçekleştirilmiştir. Önerilen IPAbFS, MATLAB programlama dilinde, MATLAB R2021a ortamında geliştirilmiştir. Önerilen IPAbFS modeli için maksimum değerlendirme sayısı 10.000 alınmıştır. Daha kararlı ve güvenilir sonuçlar elde edebilmek için deneysel çalışmalar 10 defa tekrar edilmiştir. Önerilen IPAbFS-KNN, IPAbFS-DT ve IPAbFS-SVM için elde edilen ortalama doğruluk değerleri, seçilen ortalama öznitelik sayıları, çalışma zamanları ve bu değerlerin standart sapma değerleri deneysel sonuçlar olarak rapor edilmiştir. Önerilen IPAbFS ve literatürdeki diğer rakip çalışmalarla karşılaştırmalı olarak sunulmuştur.

Bu bölümde Önerilen IPAbFS'nin deneysel sonuçları detaylı sunulmaktadır. Önerilen yöntemin başarısını doğrulamak için düşük özniteliğe-yüksek özniteliğe ve düşük örnekleme-büyük örnekleme sahip, ikili ve çoklu sınıf etiketli ve farklı problem türlerini temsil edebilecek datasetler tercih edilmiştir. Önerilen IPAbFS, bahsedilen kriterleri sağlayacak 39 adet veriseti (BreastCancer, TicTacToe, HeartDisease, HeartEW, Exactly, Exactly2, MofN, Vote, CongressEW, Hepatitis, German, SpectD, BreastEW, WDBC, Ionosphere, KRVsKpEw, Qsar, Spambase, Hillvalley, Musk1, Musk2, PenglunEW, Madelon, Colon Cancer, Leukemia, Vowel, Wine, Zoo, Letter, Lymphography, Vehicle, Segmentation, CTG, Dermatology, Waveform, Libras, Arrhythmia, Isolet, TOX-171) üzerinden test edilmiştir. Deneysel sonuçlar için kullanılan verisetlerine ait özellikler Çizelge 4.1'de rapor edilmiştir. Tabloda görüldüğü üzere, tüm datasetler Binary, Multiclass, Low-Dimensional ve High-Dimensional olarak kategori edilmiştir. Kullanılan datasetlerin 24 tanesi ikili, 15 tanesi çoklu sınıflı, 32 tanesi düşük boyutlu ve 7 tanesi yüksek boyutlu veriseti olarak tanımlanmıştır. Seçilen verisetlerinin öznitelik sayıları 10 ile 7129 arasında dağılım gösterirken sınıf etiket sayıları 2 ile 26 sayıları arasında dağılım göstermektedir.

Önerilen IPAbFS'nin başarısının adil olarak değerlendirilebilmesi için 10-kat çapraz doğrulama tekniği kullanılmaktadır. 10-kat çapraz doğrulama tekniğinde veriseti 10 parçaya bölünerek bir veri kümesi oluşturulur. Bu veri kümesinin 9 tanesi modelin eğitim verisi olarak kullanılırken 1 tanesi test verisi olarak kullanılır. Bu işlem 10 kere tekrarlanarak modelin eğitim ve test aşamasında veri setindeki bütün verilerin kullanılması sağlanır. Bu şekilde daha kararlı ve doğru sonuçlara elde edilebilmektedir. Modelimizin test çalışmasında her bir iterasyonda 10-kat çapraz doğrulama tekniği uygulanmış ve sonuçlar raporlanmıştır.

Çizelge 4.1. Verisetlerine ait özellikler

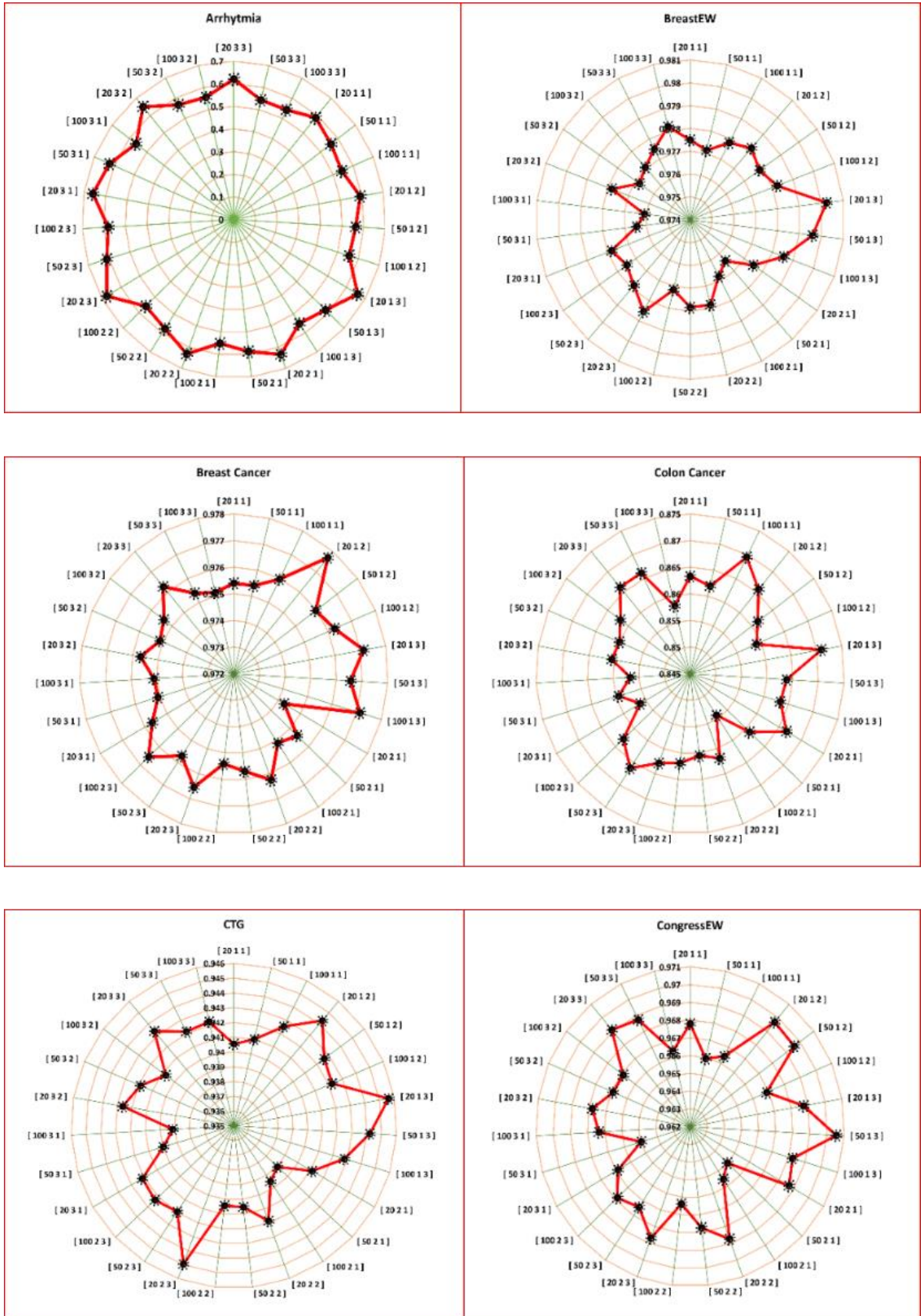
Sıra	Veriseti	Öznitelik Sayısı	Örneklem Sayısı	Sınıf Sayısı	Sınıf Tipi	Veriseti Türü
1	BreastCancer	10	699	2	İkili	Düşük Boyutlu
2	TicTacToe	10	958	2	İkili	Düşük Boyutlu
3	HeartDisease	14	270	2	İkili	Düşük Boyutlu
4	HeartEW	14	270	2	İkili	Düşük Boyutlu
5	Exactly	14	1000	2	İkili	Düşük Boyutlu
6	Exactly2	14	1000	2	İkili	Düşük Boyutlu
7	MofN	14	1000	2	İkili	Düşük Boyutlu
8	Vote	17	300	2	İkili	Düşük Boyutlu
9	CongressEW	17	435	2	İkili	Düşük Boyutlu
10	Hepatitis	20	155	2	İkili	Düşük Boyutlu
11	German	20	1000	2	İkili	Düşük Boyutlu
12	SpectD	23	267	2	İkili	Düşük Boyutlu
13	BreastEW	31	569	2	İkili	Düşük Boyutlu
14	WDBC	31	569	2	İkili	Düşük Boyutlu
15	Ionosphere	35	351	2	İkili	Düşük Boyutlu
16	KrVsKpEW	37	3196	2	İkili	Düşük Boyutlu
17	Qsar	42	1055	2	İkili	Düşük Boyutlu
18	Spambase	58	4601	2	İkili	Düşük Boyutlu
19	Hillvalley	101	1212	2	İkili	Düşük Boyutlu
20	Musk1	167	476	2	İkili	Düşük Boyutlu
21	Musk2	167	476	2	İkili	Düşük Boyutlu
22	PenglungEW	327	73	2	İkili	Yüksek Boyutlu
23	Madelon	501	2600	2	İkili	Yüksek Boyutlu
24	Colon Cancer	2001	62	2	İkili	Yüksek Boyutlu
25	Leukemia	7129	72	2	İkili	Yüksek Boyutlu
26	Vowel	11	990	10	Çoklu	Düşük Boyutlu
27	Wine	14	178	3	Çoklu	Düşük Boyutlu
28	Zoo	17	101	7	Çoklu	Düşük Boyutlu
29	Letter	17	20000	26	Çoklu	Düşük Boyutlu
30	Lymphography	19	148	4	Çoklu	Düşük Boyutlu
31	Vehicle	19	846	4	Çoklu	Düşük Boyutlu
32	Segmentation	20	2310	7	Çoklu	Düşük Boyutlu
33	CTG	22	2126	3	Çoklu	Düşük Boyutlu
34	Dermatology	35	366	6	Çoklu	Düşük Boyutlu
35	Waveform	41	5000	4	Çoklu	Düşük Boyutlu
36	Libras	91	360	15	Çoklu	Düşük Boyutlu
37	Arrhythmia	280	452	16	Çoklu	Yüksek Boyutlu
38	Isolet	618	1559	26	Çoklu	Yüksek Boyutlu
39	TOX-171	5748	171	4	Çoklu	Yüksek Boyutlu

Önerilen IPAbFS ile sınıflandırma işleminde KNN, DT ve SVM makine öğrenmesi yöntemleri kullanılmıştır. Modelin temel sınıflandırıcısı olarak KNN seçilmiştir. KNN yöntemi uygulamasının kolay olması ve hesaplama süresinin görece düşük olması neden ile

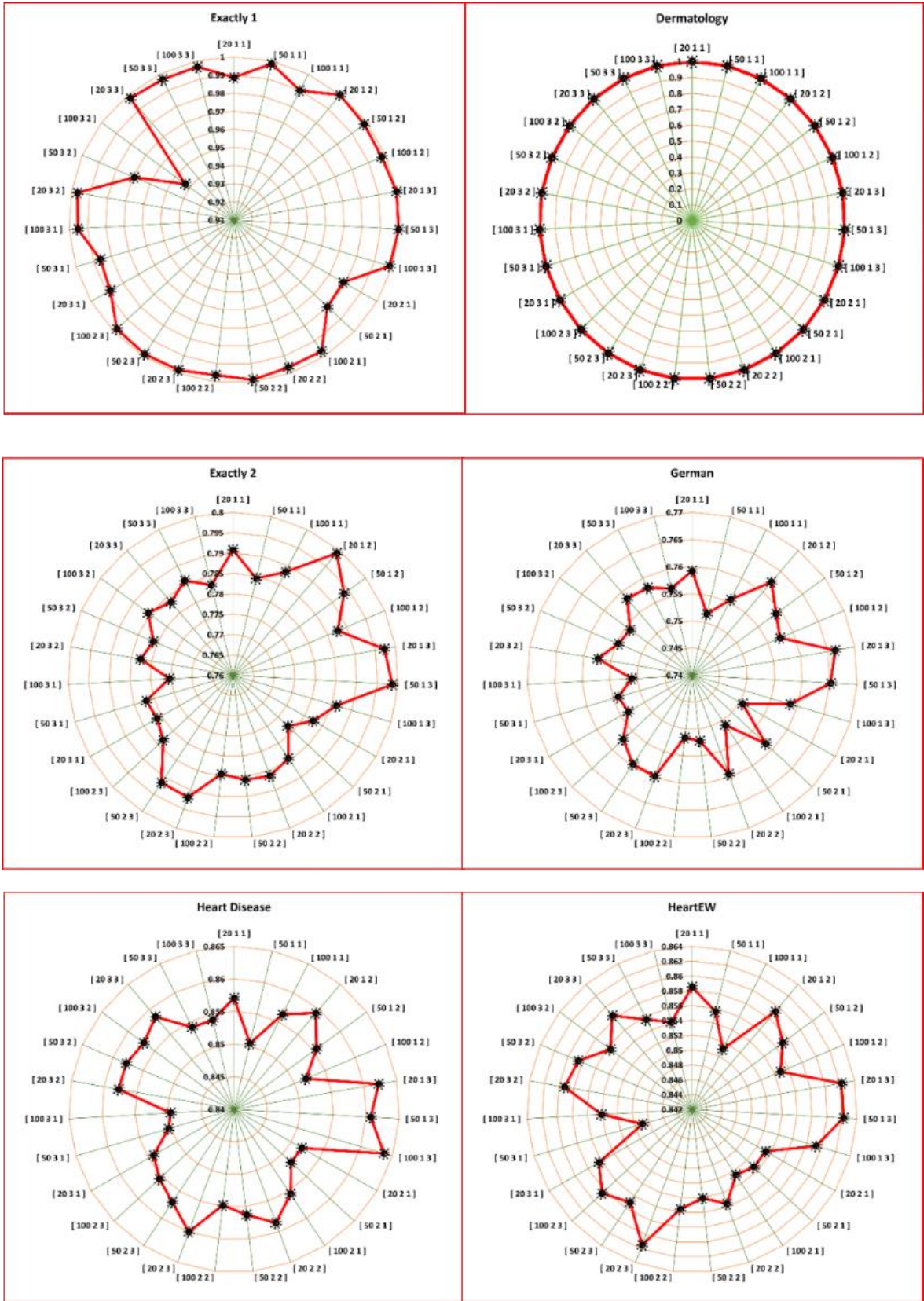
temel sınıflandırıcı olarak kullanılmıştır. KNN algoritmasında k komşuluk değerinin model başarısında büyük önemi bulunmaktadır. Bu çalışmada en uygun k komşuluk değerini belirlemek için farklı kategorilerdeki verisetlerinden 3'er tane seçilerek farklı k komşuluk değerleri ile IPAbFS-KNN koşturulmuştur. Elde edilen sonuçlara göre en uygun k komşuluk değerinin 5 olduğu görülmüştür. KNN temelli tüm IPAbFS sonuçları için $k=5$ üzerinden elde edilmiştir.

Önerilen IPAbFS nin kontrol parametreleri olan PS, NoD ve NoR model başarısına etki edip etmediği test edilmiştir. Bu amaç için [20, 50, 100] PS değerleri, [1, 2, 3] NoD değerleri ve [1, 2, 3] NoR değerlerinin tüm kombinasyonları ile IPAbFS-KNN yöntemiyle sonuçlar alınmıştır. Her bir veri seti için kontrol parametrelerin kombinasyonları üzerinden elde edilen doğruluk değerleri Şekil 4.1'de örümcek ağı grafikleri ile gösterilmiştir. Bu figürler incelendiğinde her bir veriseti için önerilen IPAbFS ile elde edilen en iyi sonuçlar için kontrol parametreleri [PS, NoD, NoR] Çizelge 4.2'de sunulmuştur.

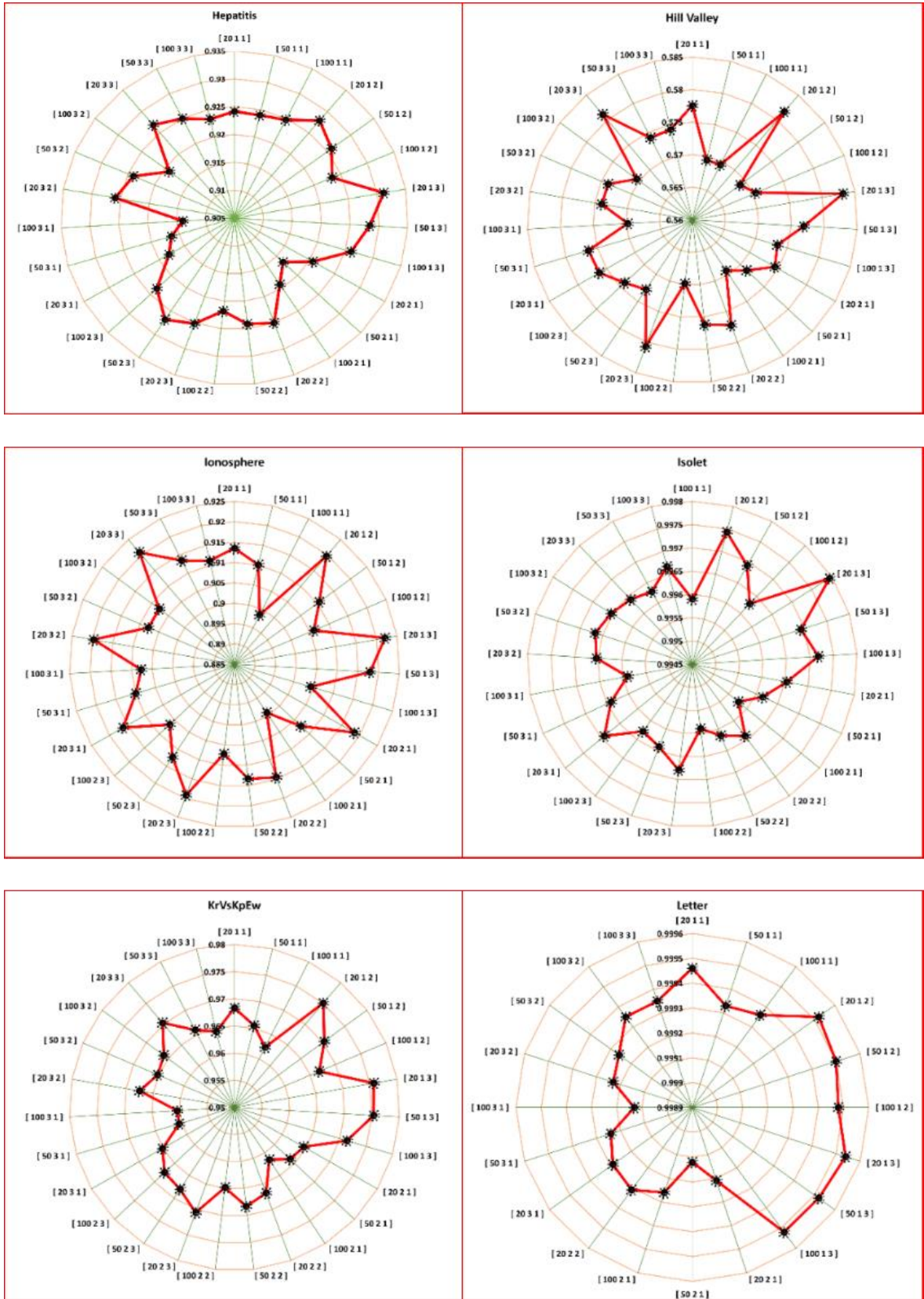
Önerilen IPAbFS yönteminin başarısının KNN'den bağımsız olarak gözlemlenebilmesi için SVM ve DT makine öğrenmesi yöntemleri için de kullanılarak simülasyonlar tekrar edilmiştir. IPAbFS'nin sınıflandırıcısı olan KNN ile elde edilen en iyi [PS, NoD, NoR] parametreleriyle IPAbFS-DT ve IPAbFS-SVM sınıflandırmaları gerçekleştirilmiştir. Elde edilen sonuçlara göre IPAbFS'nin diğer makine öğrenmesi modelleri ile performansları karşılaştırılmıştır. Elde edilen deneysel sonuçlar Çizelge 4.2'de sunulmuştur. Çizelge 4.2'de en iyi değerler kalın punto ile gösterilmiştir.



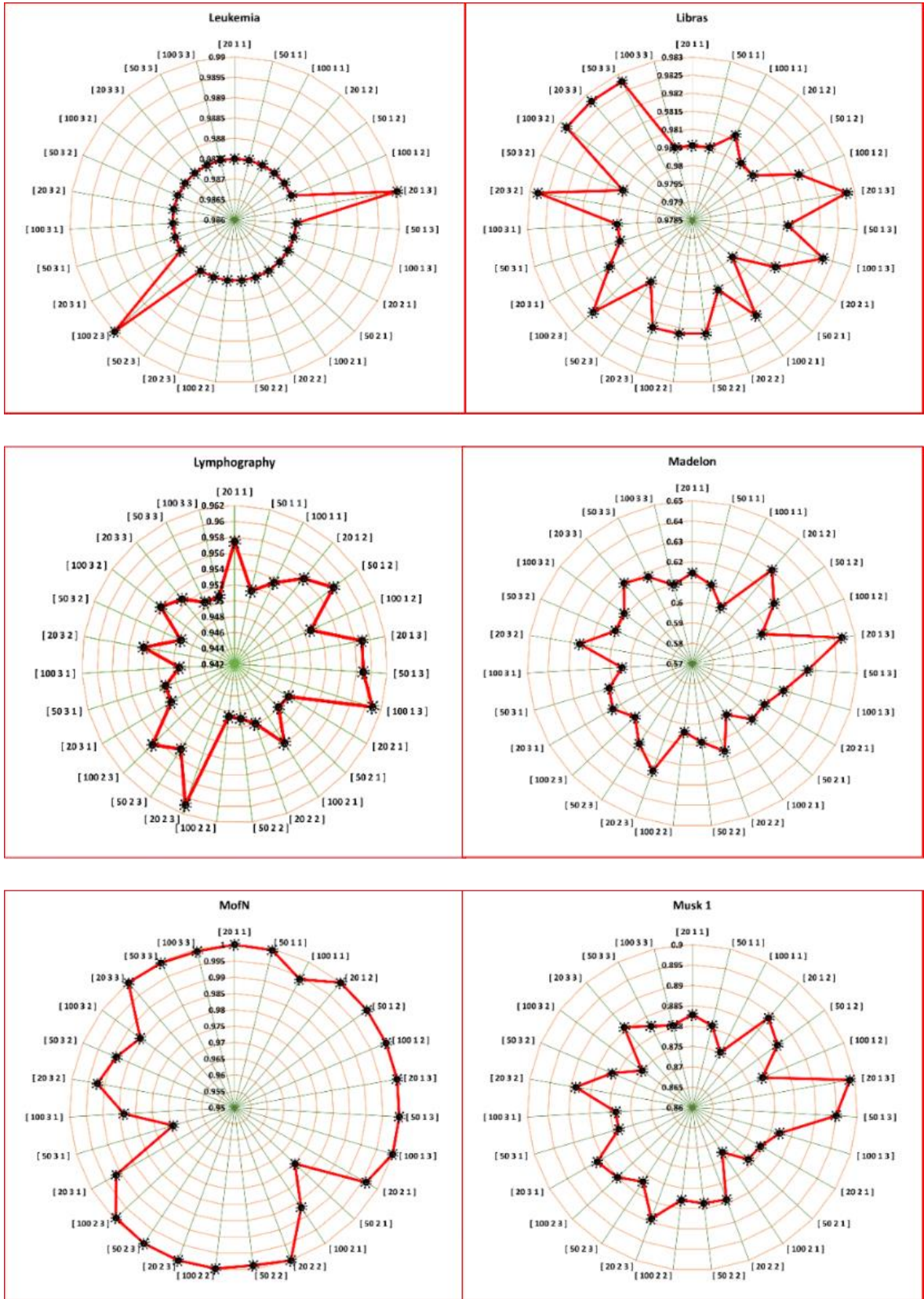
Şekil 4.1. Önerilen IPAbFS-KNN yönteminin kullanılan verisetleri için tüm IPA kontrol parametreleri için doğruluk değerleri



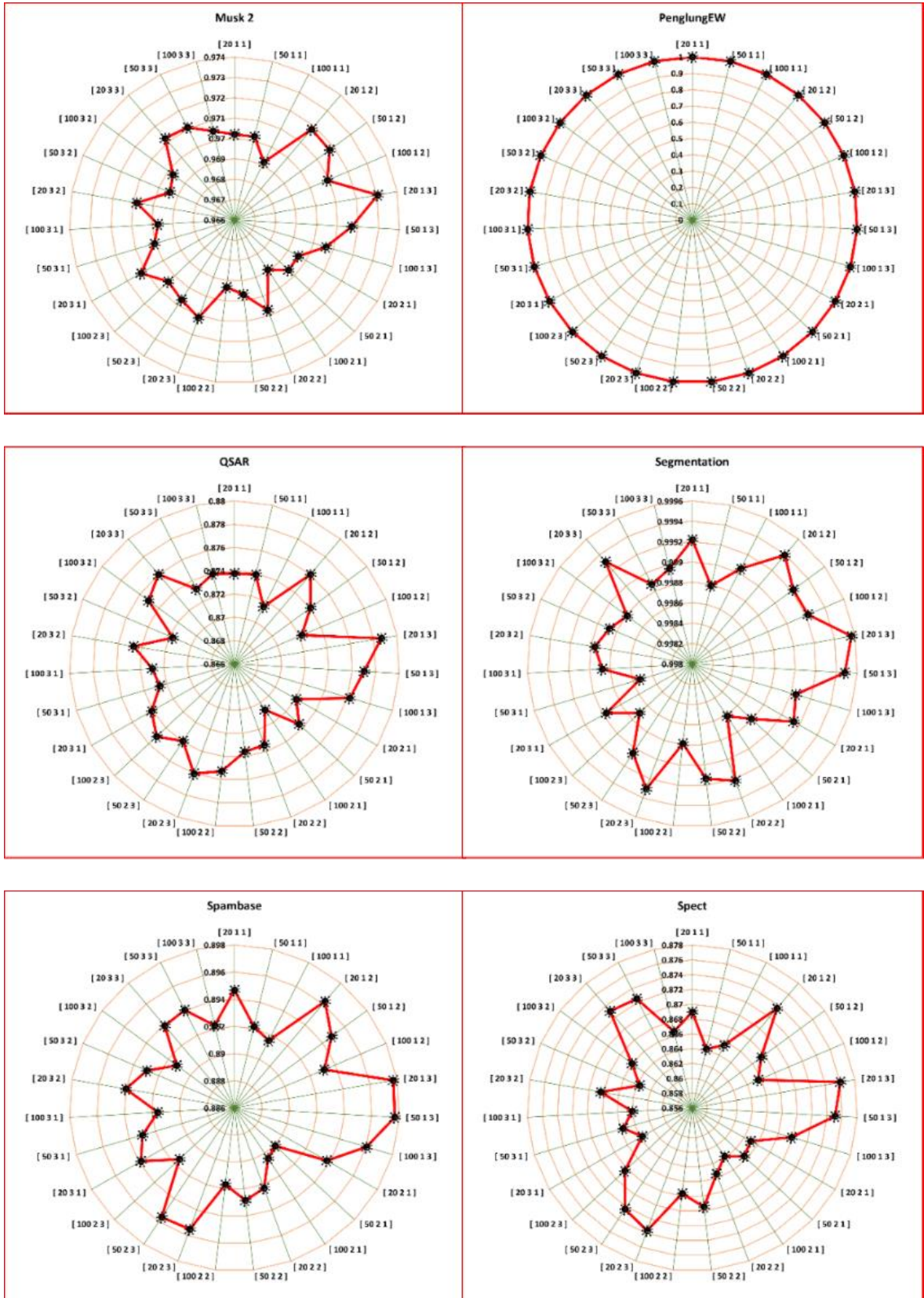
Şekil 4.1. (Devamı) Önerilen IPAbFS-KNN yönteminin kullanılan verisetleri için tüm IPA kontrol parametreleri için doğruluk değerleri



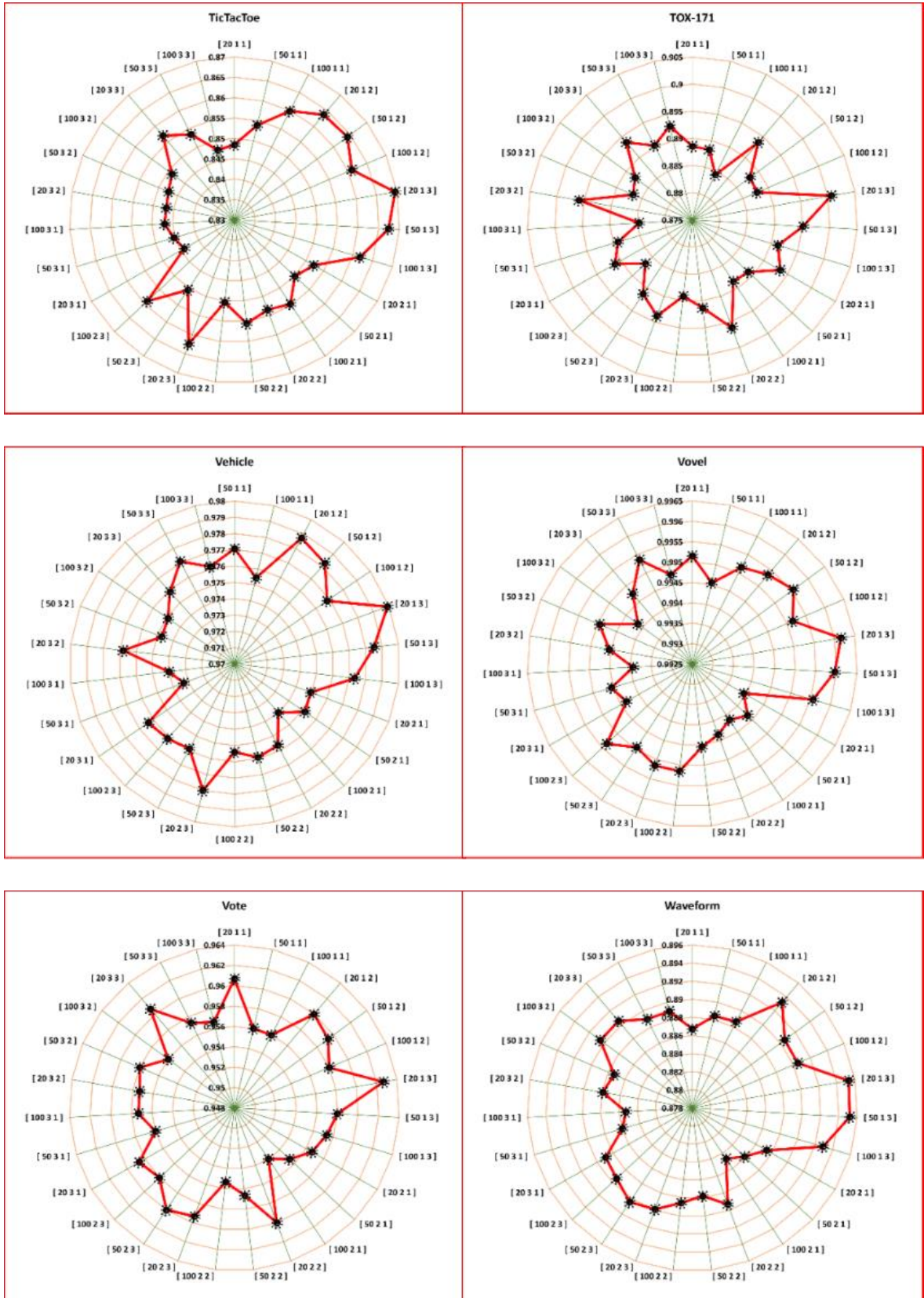
Şekil 4.1. (Devamı) Önerilen IPAbFS-KNN yönteminin kullanılan verisetleri için tüm IPA kontrol parametreleri için doğruluk değerleri



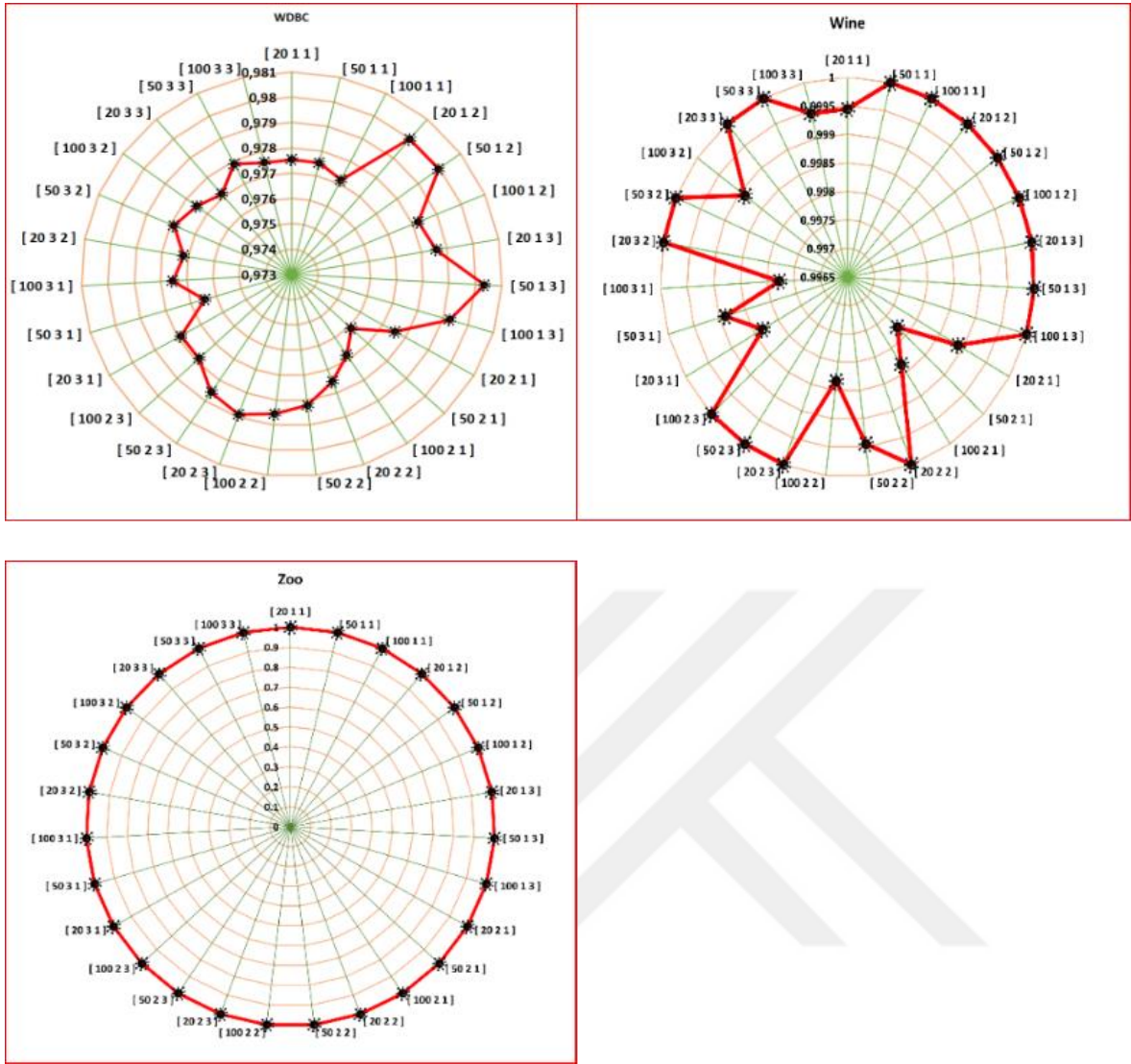
Şekil 4.1. (Devamı) Önerilen IPAbFS-KNN yönteminin kullanılan verisetleri için tüm IPA kontrol parametreleri için doğruluk değerleri



Şekil 4.1. (Devamı) Önerilen IPAbFS-KNN yönteminin kullanılan verisetleri için tüm IPA kontrol parametreleri için doğruluk değerleri



Şekil 4.1. (Devamı) Önerilen IPAbFS-KNN yönteminin kullanılan verisetleri için tüm IPA kontrol parametreleri için doğruluk değerleri



Şekil 4.1. (Devamı) Önerilen IPAbFS-KNN yönteminin kullanılan verisetleri için tüm IPA kontrol parametreleri için doğruluk değerleri

Çizelge 4.2. Makine öğrenmesi modelleri ile elde edilen deneysel sonuçlar

Veriseti	<u>IPA</u>			<u>KNN</u>		<u>DT</u>		<u>SVM</u>	
	<u>PARAMETRELERİ</u>								
	Pop.	NoD	NoR	ORT	STD	ORT	STD	ORT	STD
Arrhythmia	20	3	1	0,676	0,0230	0,8138	0,0019	0,6921	0,0095
BreastCancer	20	1	2	0,977	0,0008	0,9677	0,0016	0,9714	0,0000
BreastEW	20	3	3	0,979	0,0023	0,9609	0,0014	0,9845	0,0008
Colon Cancer	20	1	3	0,871	0,0031	0,945	0,0068	0,94	0,0056
CongressEW	50	1	3	0,970	0,0010	0,9724	0,0001	0,9586	0,0001
CTG	20	1	3	0,945	0,0008	0,9550	0,0013	0,9148	0,0008
Dermatology	20	1	3	1	0,0000	1	0,0000	1	0,0000
Exactly1	20	1	3	1	0,0000	1	0,0000	0,688	0,0000
Exactly2	50	1	3	0,798	0,0023	0,8024	0,0044	0,758	0,0000
German	20	1	3	0,766	0,0018	0,7544	0,0027	0,7762	0,0011
HeartDisease	100	1	3	0,863	0,0048	0,8652	0,0020	0,8637	0,0031
HeartEW	20	1	3	0,862	0,0031	0,8637	0,0017	0,863	0,0000
Hepatitis	20	1	3	0,931	0,0031	0,9989	0,0016	0,8916	0,0039
Hillvalley	20	1	3	0,583	0,0047	0,6072	0,0024	0,7629	0,0035
Ionosphere	20	1	3	0,922	0,0043	0,972	0,0012	0,9145	0,0021
Isolet	20	1	3	0,997	0,0007	0,9920	0,0006	0,9964	0,0003
KrVsKpEW	20	1	3	0,975	0,0033	0,9905	0,0018	0,9503	0,0023
Letter	20	1	3	0,999	0,0000	0,9968	0,0002	0,9924	0,0001
Leukemia	20	1	3	0,99	0,0056	0,9895	0,0000	1	0,0000
Libras	20	1	3	0,982	0,0012	0,9844	0,0023	0,9789	0,0015
Lymphographyh	100	1	3	0,960	0,0039	0,9445	0,0101	0,9420	0,0038
Madelon	20	1	3	0,643	0,0045	0,7894	0,0012	0,61	0,0031
MofN	20	1	3	1	0,0000	1	0,0000	1	0,0000
Musk1	20	1	3	0,898	0,0035	0,8681	0,0035	0,8446	0,0001
Musk2	20	1	3	0,973	0,0008	0,9777	0,0004	0,8765	0,0027
PenglungEW	20	1	3	1	0,0000	1	0,0000	1	0,0000
Qsar	20	1	3	0,878	0,0020	0,8567	0,0024	0,8542	0,0004
Segmentation	20	1	3	0,999	0,0000	0,9981	0,0002	0,9970	0,0000
Spambase	20	1	3	0,923	0,0014	0,9336	0,0007	0,8914	0,0065
SpectD	20	1	3	0,876	0,0038	0,9108	0,0046	0,841	0,0039
TicTacToe	20	1	3	0,869	0,0050	0,8866	0,0020	0,6537	0,0000
TOX-171	20	1	3	0,900	0,0071	0,9056	0,0073	0,9147	0,0033
Vehicle	20	1	3	0,979	0,0009	0,9797	0,0005	0,9203	0,0026
Vote	20	1	3	0,962	0,0028	0,9667	0,0000	0,9553	0,0018
Vowel	20	1	3	0,996	0,0008	0,9808	0,0007	0,9679	0,0017
Waveform	20	1	3	0,895	0,0023	0,8570	0,0022	0,9142	0,0042
WDBC	50	1	3	0,980	0,0015	0,9599	0,0023	0,9835	0,0010
Wine	20	1	3	1	0,0000	0,9944	0,0000	1	0,0000
Zoo	20	1	3	1	0,0000	1	0,0000	1	0,0000
Ort, Doğruluk				0,918	0,0027	0,9293	0,0016	0,8986	0,0018
En İyi Sonuç Sayısı				15		22		12	
Toplam Rank				2		1		3	

Çizelge 4.2 incelendiğinde, ortalama doğruluk değerinde sırasıyla 0,9293 ile IPAbFS-DT birinci, 0,9188 ile IPAbFS-KNN ikinci ve 0,8986 ile IPAbFS-SVM üçüncü olduğu görülmektedir. Fakat, IPAbFS-DT ile IPAbFS-KNN arasında nispeten çok az bir fark vardır. IPAbFS-SVM ise 3 model arasından en kötü performansı gösteren model olmuştur. Ayrıca IPAbFS-DT 39 datasetten 22'sinde en iyi sonucu elde eden yöntem olmuştur. Bunu yanında IPAbFS-KNN modelinin 14 ve IPAbFS-SVM modelinin 12 datasette en iyi sonucu elde eden yöntem olduğu görülmektedir.

Önerilen IPAbFS-KNN yöntemi Breast Cancer, Dermatology, Exactly1, Exactly2, Isolet, Letter, Lymphography, MofN, Musk1, PenglungEW, Qsar, Segmentation, Vehicle, Vowel, Wine ve Zoo verisetlerinde önerilen diğer IPA temelli yöntemlere göre daha başarılı ya da eşit sonuçlar elde etmiştir. Ayrıca önerilen IPAbFS-KNN yöntemiyle bütün verisetleri üzerinde 10 bağımsız çalıştırma sonucunda ortalama 0,0027 standart sapma değeri elde edilmiştir. Önerilen IPAbFS-DT yöntemiyle Arrhythmia, Colon Cancer, CongressEW, CTG, Dermatology, Exactly1, Exactly2, Heart Disease, HeartEW, Hepatitis, Ionosphere, KrVsKpEW, Libras, Madelon, MofN, Musk2, PenglungEW, Spambase, SpectD, TicTacToe, Vote ve Zoo verisetlerinde önerilen diğer IPA temelli yöntemlere göre daha başarılı ya da eşit sonuçlar elde edilmiştir. Ayrıca önerilen IPAbFS-DT yöntemiyle bütün verisetleri üzerinde 10 bağımsız çalıştırma sonucunda ortalama 0,0016 standart sapma değeri elde edilmiştir. Bu değer, önerilen diğer iki yöntemden daha düşük olduğundan dolayı IPAbFS-DT ile elde edilen değerlerin standart sapma değerlerine bakıldığında diğer önerilen diğer iki yöntemle göre nispeten daha tutarlı sonuçlar verdiği söylenebilir. IPAbFS-SVM ile BreastEW, Dermatology, German, Hillvalley, Leukemia, MofN, PenglungEW, Tox-171, Waveform, WDBC, Wine ve Zoo verisetlerinde önerilen diğer IPA temelli yöntemlere göre daha başarılı ya da eşit sonuçlar elde edildiği görülebilmektedir. Ayrıca önerilen IPAbFS-SVM yöntemiyle bütün

verisetleri üzerinde 10 bağımsız çalıştırma sonucunda ortalama 0,0018 standart sapma değeri elde edilmiştir. IPAbFS-SVM ile elde edilen ortalama standart değeri önerilen IPAbFS-DT yönteminden nispeten daha büyükken önerilen IPAbFS-KNN yönteminden nispeten daha küçük olduğu görülebilmektedir.

Öznitelik seçimi yöntemlerinde, doğruluk değerleri kadar kullanılan yöntemlerin çalışma zamanları da önemli performans göstergelerinden biridir. Bu tezde her bir dataset üzerinde test edilen yöntemlerin 10 çalıştırmada elde edilen ortalama çalıştırma süreleri elde edilerek Çizelge 4.3’de sunulmuştur.

Çizelge 4.3’e göre en hızlı yöntem IPAbFS-KNN olduğu görülmektedir. IPAbFS-KNN modelinin hızının temel nedeni KNN makine öğrenmesi yönteminin basit ve hızlı olmasından kaynaklanmaktadır. DT ve SVM makine öğrenmesi yöntemlerinin temel işlem adımları KNN makine öğrenmesi modeline göre daha karmaşıktır. Bu nedenle bu yöntemlerin sınıflandırma süreleri KNN makine öğrenmesi yöntemine göre daha uzun sürebilmektedir. Çizelge 4.3 incelendiğinde IPAbFS-DT, IPAbFS-KNN modeline göre uzun sürede sınıflandırma işlemi gerçekleştirmiştir. IPAbFS-SVM ise önerilen diğer 3 modele göre en yavaş sınıflandırma işlemi gerçekleştiren modeldir.

Çizelge 4.3. Her bir dataset üzerinde IPAbFS ile test edilen yöntemlerin 10 çalıştırmada elde edilen ortalama çalışma süreleri (s)

Veriseti	IPAbFS-KNN	IPAbFS-DT	IPAbFS-SVM
Arrhythmia	1872,88	4400,67	45873,95
BreastCancer	1011,36	1059,03	1701,76
BreastEW	732,99	1387,37	1624,65
Colon Cancer	677,65	568,96	3529,16
CongressEW	615,80	1357,38	1451,97
CTG	1459,89	2500,74	1809,05
Dermatology	629,99	948,07	13929,37
Exactly1	768,00	1205,49	4008,64
Exactly2	783,32	1762,61	3951,36
German	634,63	1510,90	9076,31
HeartDisease	578,65	1412,73	4015,36
HeartEW	944,97	1407,06	3106,43
Hepatitis	571,59	1335,62	3288,56
Hillvalley	768,17	5125,61	37954,10
Ionosphere	605,61	1236,89	1365,38
Isolet	6871,29	77915,65	8324,02
KrVsKpEW	778,45	817,74	13103,69
Letter	5711,68	7292,59	9394,86
Leukemia	543,19	1648,02	882,13
Libras	667,83	905,38	20868,88
Lymphography	587,68	513,18	1473,79
Madelon	3213,05	15721,32	9284,27
MofN	1090,29	601,35	1510,38
Musk1	644,62	4235,87	839,66
Musk2	4091,76	8736,42	4212,63
PenglungEW	579,77	602,78	4850,77
Qsar	581,91	927,65	2082,72
Segment	1061,17	1093,64	6195,32
Spambase	1280,87	7272,13	2443,20
SpectD	619,98	1471,68	1280,87
TicTacToe	1046,20	645,37	1647,48
TOX-171	1640,30	9742,97	27858,27
Vehicle	708,19	806,28	1566,48
Vote	578,48	1335,67	1133,82
Vowel	1164,57	2819,36	23770,44
Waveform	1035,66	2955,19	6981,00
WDBC	574,01	1652,84	2924,50
Wine	534,66	1345,53	3976,23
Zoo	959,75	1370,36	8779,06
Ortalama Hesaplama Süresi (Sıra)	1262,07 (1)	4606,36 (2)	7745,40 (3)

Çizelge 4.4. 10 bağımsız koşturma sonucunda elde edilen ortalama öznitelik sayıları

Veriseti	Öznitelik Sayısı	KNN ile Seçilen Ort, Öznitelik Sayısı	DT ile Seçilen Ort, Öznitelik Sayısı	SVM ile Seçilen Ort, Öznitelik Sayısı
Arrhythmia	279	32,5	113	91,8
BreastCancer	9	6	4,2	6,8
BreastEW	30	6	11,4	13,4
Colon	2000	803,4	872,8	903,4
CongressEW	16	5,6	8,4	9,8
CTG	22	8,6	11,6	14
Dermatology	34	18,6	15,6	19,4
Exactly1	13	6,2	7	7,4
Exactly2	13	9	10,6	6,8
German	24	11	10	16,6
HeartDisease	13	5,4	3	9
HeartEW	13	6,2	3,2	7,8
Hepatitis	19	10	10,4	8,6
Hillvalley	100	28,2	33,8	50,2
Ionosphere	34	7,4	15,8	15
Isolet	617	293	236,4	297,4
KrVsKpEW	36	16,4	24,6	19,8
Letter	16	14	11	10
Leukemia	3571	1786,4	1765,4	1797,2
Libras	90	38,2	40,5	47,4
Lymphography	18	11	7,2	12
Madelon	500	157,8	217,2	192,8
MofN	13	6,8	9,6	8,6
Musk1	166	60,2	62,6	77,2
Musk2	166	77,8	76,2	57,6
PenglungEW	325	166	164	153,6
Qsar	41	20	16,6	24,8
Segmentation	19	9,4	8,4	10,2
Spambase	34	17,2	30,4	29,8
SpectD	22	11,6	13,8	15
TicTacToe	9	7	8,4	3,8
TOX-171	5748	1946	2018,2	2651,2
Vehicle	18	8,4	10,2	6,4
Vote	16	4,6	8,2	9
Vowel	10	7,4	7,2	9,2

Waveform	21	15,2	12,6	15,4
WDBC	30	14	13,4	14
Zoo	16	9,6	6,6	8,2
Wine	13	6,8	8,6	11,6
Seilen Ort, znitelik Sayısı	362,41	145,35	151,48	171,23
En İyi Sonu Sayısı		18	14	7
Toplam Rank		1	2	3

Kullanılan verisetleri iin nerilen IPAbFS-KNN, IPAbFS-DT ve IPAbFS-SVM yntemleri ile 10 bağımsız kořma zerinden ortalama seilen znitelik sayıları izelge 4.4'te verilmiřtir. Ayrıca, seilen znitelik sayılarının ortalamaları da hesaplanmıřtır. IPAbFS-KNN ortalamada 145.35, IPAbFS-DT ortalamada 151.48 ve IPAbFS-SVM ortalamada 171.23 znitelik semiřtir. Buna gre btn verisetleri iin boyut indirgeme iřleminde en bařarılı olan yntem IPAbFS-KNN olmuřtur. Ek olarak, IPAbFS-KNN yntemi toplam 39 verisetinin 18'inde diğerk iki ynteme gre daha az sayıda znitelik ile benzer doğruluk değeri elde edilmiřtir.

nerilen IPAbFS ynteminde kullanılan KNN, DT ve SVM sınıflandırıcılarının performansları arasında istatiksel olarak anlamlı farkın gzlemlenebilmesi iin Wilcoxon istatiksel testi uygunlařmıřtır. Elde edilen istatiksel sonular izelge 4.5'te rapor edilmiřtir. izelge 4.5 incelendiğinde, 39 veri seti zerinden elde edilen sonulara gre IPAbFS-KNN yntemi ile IPAbFS-DT yntemi arasında 15 verisetinde IPAbFS-KNN'nin lehine anlamlı fark bulunurken, 13 verisetinde IPAbFS-DT ynteminin lehine anlamlı fark bulunmuřtur ($p<0.05$). Ayrıca, 11 verisetinde ise herhangi bir anlamlı fark bulunmamıřtır. IPAbFS-KNN ile IPAbFS-SVM arasındaki sonular incelendiğinde ise 24 verisetinde IPAbFS-KNN'nin lehine anlamlı fark bulurken, 7 verisetinde IPAbFS-SVM lehine anlamlı fark ıkmıřtır ($p<0.05$). IPAbFS-KNN ile IPAbFS-SVM arasında 8 verisetinde

ise herhangi bir anlamlı fark bulunamamıştır ($p>0.05$). IPAbFS-SVM ile IPAbFS-DT arasındaki istatistiksel analizde ise 18 verisetinde IPAbFS-DT lehine anlamlı fark vardır ($p<0.05$). IPAbFS-SVM yönteminin lehine ise 12 verisetinde anlamlı fark bulunmuşken, IPAbFS-SVM ile IPAbFS-DT arasında 9 verisetinde herhangi bir anlamlı fark bulunamamıştır ($p>0.05$).



Çizelge 4.5. Önerilen IPAbFS yönteminde kullanılan KNN, DT SVM sınıflandırıcıları arasında Wilcoxon istatistiksel analiz sonuçları

Verseti	<u>IPAbFS-KNN-- IPAbFS-DT</u>			<u>IPAbFS-KNN-- IPAbFS-SVM</u>			<u>IPAbFS-SVM-- IPAbFS-DT</u>		
	Ort. Fark,	p	Sig,	Ort. Fark,	p	Sig,	Ort. Fark,	p	Sig,
Arrhythmia	-0,1376	0,0079	DT	-0,0158	0,0635	Not Sig,	-0,1218	0,0079	DT
BreastCancer	0,01	0,0079	KNN	0,0063	0,0079	KNN	0,0037	0,0079	SVM
BreastEW	0,0184	0,0079	KNN	-0,0052	0,0079	SVM	0,0236	0,0079	SVM
Colon	-0,074	0,0079	DT	-0,069	0,0079	SVM	-0,005	0,5238	Not Sig,
CongressEW	-0,0019	0,0159	DT	0,0119	0,0079	KNN	-0,0138	0,0079	DT
CTG	-0,0094	0,0079	DT	0,0309	0,0079	KNN	-0,0403	0,0079	DT
Dermatology	0	1	Not Sig,	0	1	Not Sig,	0	1	Not Sig,
Exactly1	0	1	Not Sig,	0,312	0,0079	KNN	-0,312	0,0079	DT
Exactly2	-0,0037	0,1508	Not Sig,	0,0407	0,0079	KNN	-0,0444	0,0079	DT
German	0,012	0,0079	KNN	-0,0098	0,0079	SVM	0,0218	0,0079	SVM
HeartDisease	-0,0015	0,7302	Not Sig,	0	1	Not Sig,	-0,0015	0,6825	Not Sig,
HeartEW	-0,0015	0,6349	Not Sig,	-0,0008	0,4444	Not Sig,	-0,0007	1	Not Sig,
Hepatitis	-0,0677	0,0079	DT	0,0396	0,0079	KNN	-0,1073	0,0079	DT
Hillvalley	-0,024	0,0079	DT	-0,1797	0,0159	SVM	0,1557	0,0079	SVM
Ionosphere	-0,05	0,0079	DT	0,0075	0,0238	KNN	-0,0575	0,0079	DT
Isolet	0,0059	0,0079	KNN	0,0015	0,0079	KNN	0,0044	0,0079	SVM
KrVsKpEW	-0,0148	0,0079	DT	0,0254	0,0079	KNN	-0,0402	0,0079	DT
Letter	0,0027	0,0476	KNN	0,0071	0,0476	KNN	-0,0044	0,0079	DT
Leukemia	0,0005	0,119	Not Sig,	-0,01	0,0238	SVM	0,0105	0,0079	SVM
Libras	-0,0017	0,0317	DT	0,0039	0,0317	KNN	-0,0056	1	Not Sig,
Lymphography	-0,0091	0,0079	DT	0,0189	0,0079	KNN	-0,028	0,0079	DT
Madelon	-0,1458	1	Not Sig,	0,0336	1	Not Sig,	-0,1794	1	Not Sig,
MofN	0	0,0079	KNN	0	0,0079	KNN	0	0,0079	SVM

Musk1	0,0307	0,0079	KNN	0,0542	0,0079	KNN	-0,0235	0,0079	DT
Musk2	-0,0046	1	Not Sig,	0,0966	1	Not Sig,	-0,1012	1	Not Sig,
PenglungEW	0	0,0079	KNN	0	0,0079	KNN	0	0,0397	SVM
Qsar	0,022	0,0079	KNN	0,0245	0,0079	KNN	-0,0025	0,0079	DT
Segmentation	0,0015	0,0079	KNN	0,0026	0,0079	KNN	-0,0011	0,0079	DT
Spambase	-0,0102	0,0079	DT	0,032	0,0079	KNN	-0,0422	0,0079	DT
SpectD	-0,0348	0,0079	DT	0,035	0,0079	KNN	-0,0698	0,0079	DT
TicTacToe	-0,0171	0,1429	Not Sig,	0,2158	0,0159	KNN	-0,2329	0,0556	Not Sig,
TOX-171	-0,0049	0,9841	Not Sig,	-0,014	0,0079	SVM	0,0091	0,0079	SVM
Vehicle	0,0002	0,0476	KNN	0,0596	0,0079	KNN	-0,0593	0,0079	DT
Vote	-0,004	0,0079	DT	0,0074	0,0079	KNN	-0,0114	0,0079	DT
Vowel	0,0154	0,0079	KNN	0,0283	0,0079	KNN	-0,0129	0,0079	DT
Waveform	0,0384	0,0079	KNN	-0,0188	0,0079	SVM	0,0572	0,0079	SVM
WDBC	0,0204	0,0079	KNN	-0,0032	1	Not Sig,	0,0236	0,0079	SVM
Zoo	0,0056	1	Not Sig,	0	1	Not Sig,	0,0056	1	Not Sig,
Wine	0	0,0079	KNN	0	0,0079	KNN	0	0,0079	SVM
KNN Sayısı			15			24			-
DT Sayısı			13			-			18
SVM Sayısı			-			7			12
Not Sig, Sayısı			11			8			9

Önerilen IPAbFS, literatürdeki diğer metasezgisel algoritma temelli öz nitelik seçimi yöntemlerine göre başarısını değerlendirebilmek için metasezgisel algoritma temelli rakip öz nitelik seçme yöntemi olan 11 tane güncel çalışmanın sonuçları ile karşılaştırılmıştır. Bu tezde önerilen IPAbFS ve literatürde önerilen rakip yöntemlere ait parametre ayarlamalarına ait değerler Çizelge 4.6’da verilmiştir.

Çizelge 4.6. Önerilen IPAbFS ve rakip yöntemlere ait parametre setleri

Öz nitelik Seçimi Yöntemi	Parametre Ayarlama Değerleri
Immune Plasma Algorithm Based Feature Selection (IPAbFS)	Popülasyon: 20 Değerlendirme sayısı: 10,000 Bağımsız çalıştırma: 10 10-kat çarpaz doğrulama
Binary Multi-Objective Chimp Optimizer with Dual Archive (BMOChOA) [87]	Popülasyon: 100 Değerlendirme sayısı: 5000 Bağımsız çalıştırma: 25 10-kat çarpaz doğrulama
Improved Equilibrium Optimizer (IBEO) [88]	Popülasyon: 5 Değerlendirme sayısı: *Yerel Arama Algoritması kullanıldığından iterasyon sayı verilmesine rağmen değerlendirme sayısı belirlenememektedir. Bağımsız çalıştırma: 20 10-kat çarpaz doğrulama
Dynamic Butterfly Optimization Algorithm (DBOA) [89]	Popülasyon: 10 Değerlendirme sayısı: *Yerel Arama Algoritması kullanıldığından iterasyon sayı verilmesine rağmen değerlendirme sayısı belirlenememektedir. Bağımsız çalıştırma: 30 10-kat çarpaz doğrulama

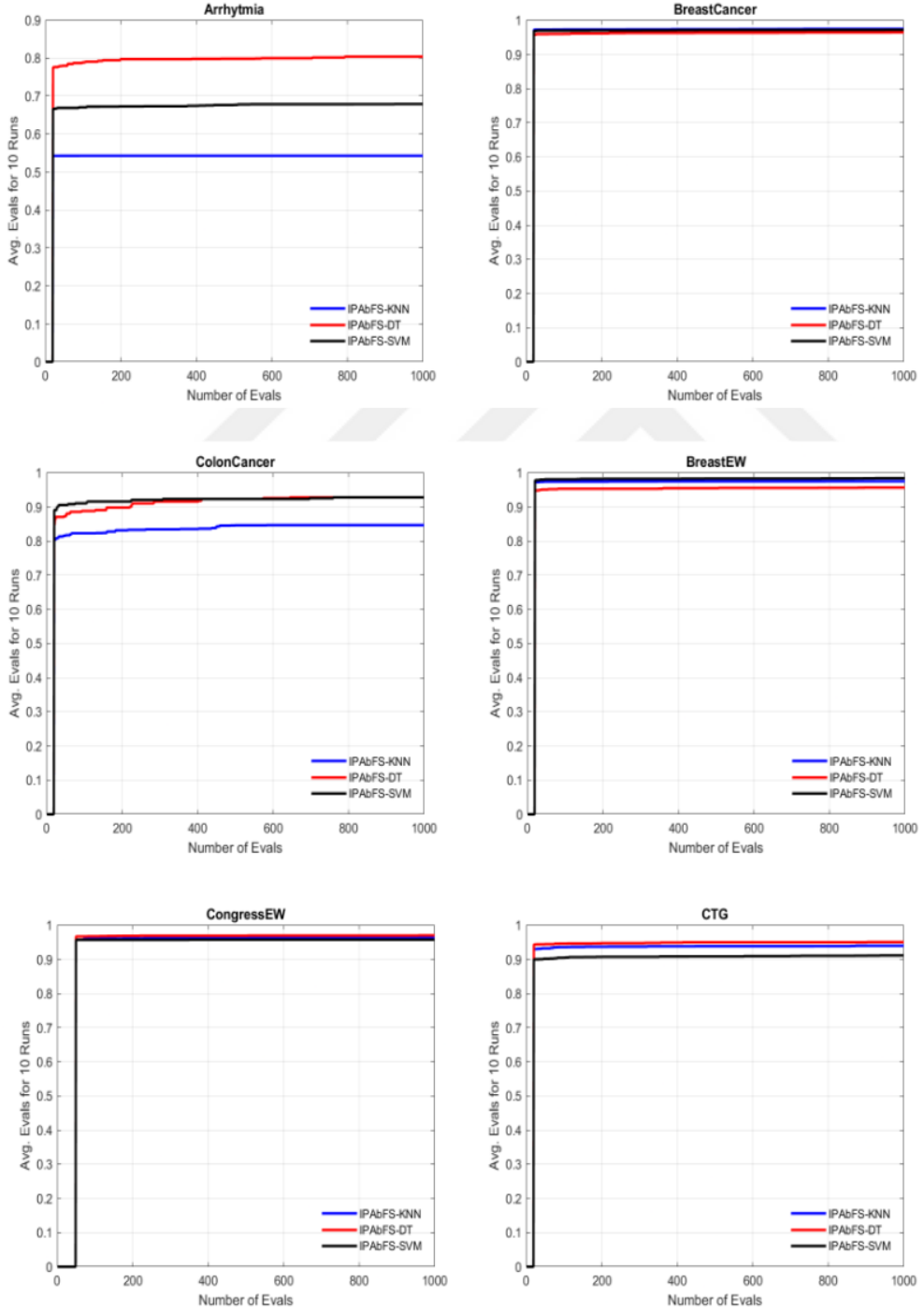
Grasshopper Optimization Algorithm with Crossover Operators (cSG) [90]	Popülasyon: 25 Değerlendirme sayısı: *Yerel Arama Algoritması kullanıldığından iterasyon sayı verilmesine rağmen değerlendirme sayısı belirlenememektedir. Bağımsız çalıştırma: 30 10-kat çapraz doğrulama
Hybrid Binary Grey Wolf With Harris Hawks Optimizer (HBGWOHHO) [64]	Popülasyon: 10 Değerlendirme sayısı: 1000 Bağımsız çalıştırma: 20 10-kat çapraz doğrulama
Improved Reptile Search Optimization Algorithm Using Chaotic Map and Simulated Annealing (IRSA) [66]	Popülasyon: 10 Değerlendirme sayısı: *Yerel Arama Algoritması kullanıldığından iterasyon sayı verilmesine rağmen değerlendirme sayısı belirlenememektedir. Bağımsız çalıştırma: 30 10-kat çapraz doğrulama
Rough Hypercuboid Approach with Binary Particle Swarm Optimization Algorithm (RH-BPSO) [91]	Popülasyon: 30 Değerlendirme sayısı: 9000 10-kat çapraz doğrulama
Multi Population Based Particle Swarm Optimization (MPPSO) [61]	Popülasyon: 20 Değerlendirme sayısı: 2000 Bağımsız çalıştırma: 20 10-kat çapraz doğrulama
Enhanced Black Widow Optimization Algorithm(SDABWO) [58]	Popülasyon: 20 Değerlendirme sayısı: 6000 Bağımsız çalıştırma: 20 5 kat çapraz doğrulama
Neighborhood Rough Set and Improved Particle Swarm Optimization (NRSIPSO) [92]	Popülasyon: 20 Değerlendirme sayısı: 800 Bağımsız çalıştırma: 10 10-kat çapraz doğrulama

Binary Horse Herd Optimization Method (BinHOA) [51]	Popülasyon: 10 Değerlendirme sayısı: *Yerel Arama Algoritması kullanıldığından iterasyon sayı verilmesine rağmen değerlendirme sayısı belirlenememektedir. Bağımsız çalıştırma: 20 5 kat çapraz doğrulama
--	---

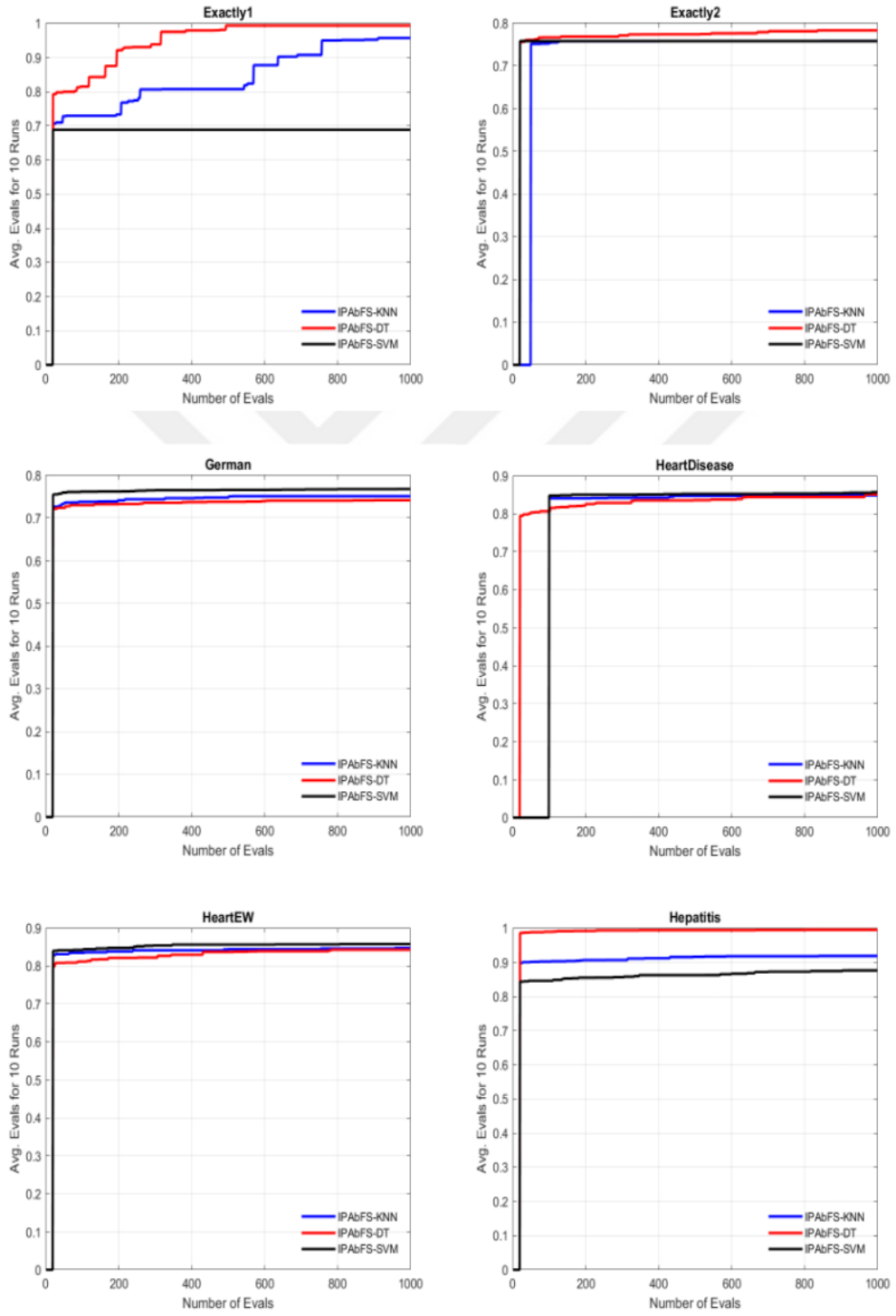
Bu tezde önerilen yöntemin toplam değerlendirme sayısı 10.000 olarak belirlense de tüm verisetlerinde ilk 1000 değerlendirme içinde en iyi sonuçların elde edildiği Şekil 4.2’de sunulan önerilen yönteme ait yakınsama grafiklerinde görülmektedir.

Karşılaştırmalı olarak sonuçlar Çizelge 4.7’de sunulmaktadır. Her dataset için alınan en iyi sonuçlar kalın punto ile belirtilmiştir. Ayrıca her data set için elde edilen sonuçların tüm sonuçlar arasındaki başarı sıra değerleri parantez içerisinde belirtilmiştir. Çizelge 4.7’de görüldüğü gibi IPAbFS 26 dataset için literatürde yer alan diğer rakip yöntemlere göre daha yüksek doğruluk değerli elde edilmiştir. Ortalama sıra da ise en iyi yöntem olduğu görülmektedir. Şekil 4.2’de önerilen IPAbFS-KNN, IPAbFS-DT ve IPAbFS-SVM yöntemlerine ait yakınsama grafikleri verilmiştir. Her 3 yöntemde de %100 ortalama doğruluk değeri elde edilen verisetlerinin yakınsama grafikleri çizdirilmemiştir. Şekil 4.2 incelendiğinde önerilen IPAbFS-KNN yönteminin ilk 1000 evaluation için BreastEW, Musk 1, Qsar, Vowel, KNN, Wine, verisetlerinde en iyi yakınsamayı elde ettiği görülmektedir. IPAbFS yöntemi Arrhythmia, CTG, Exactly1, Exactly2, HeartDisease, Hepatitits, Ionosphere, KrVsKpEW, Madelon, Spambase, SpectEW, Vote ve TicTacToe verisetlerinde en iyi yakınsamayı önerilen IPAbFS-DT göstermiştir. Önerilen IPAbFS-SVM ise TOX-171, Waveform, Colon Cancer, Exactly2, German, HeartEW, HillValley verisetlerinde en iyi yakınsamayı göstermiştir. Breast Cancer, Letter, Libras, Musk2, Leukemia, Segmentation ve CongressEW verisetlerinde önerilen IPAbFS ile sınıflandırmadan bağımsız

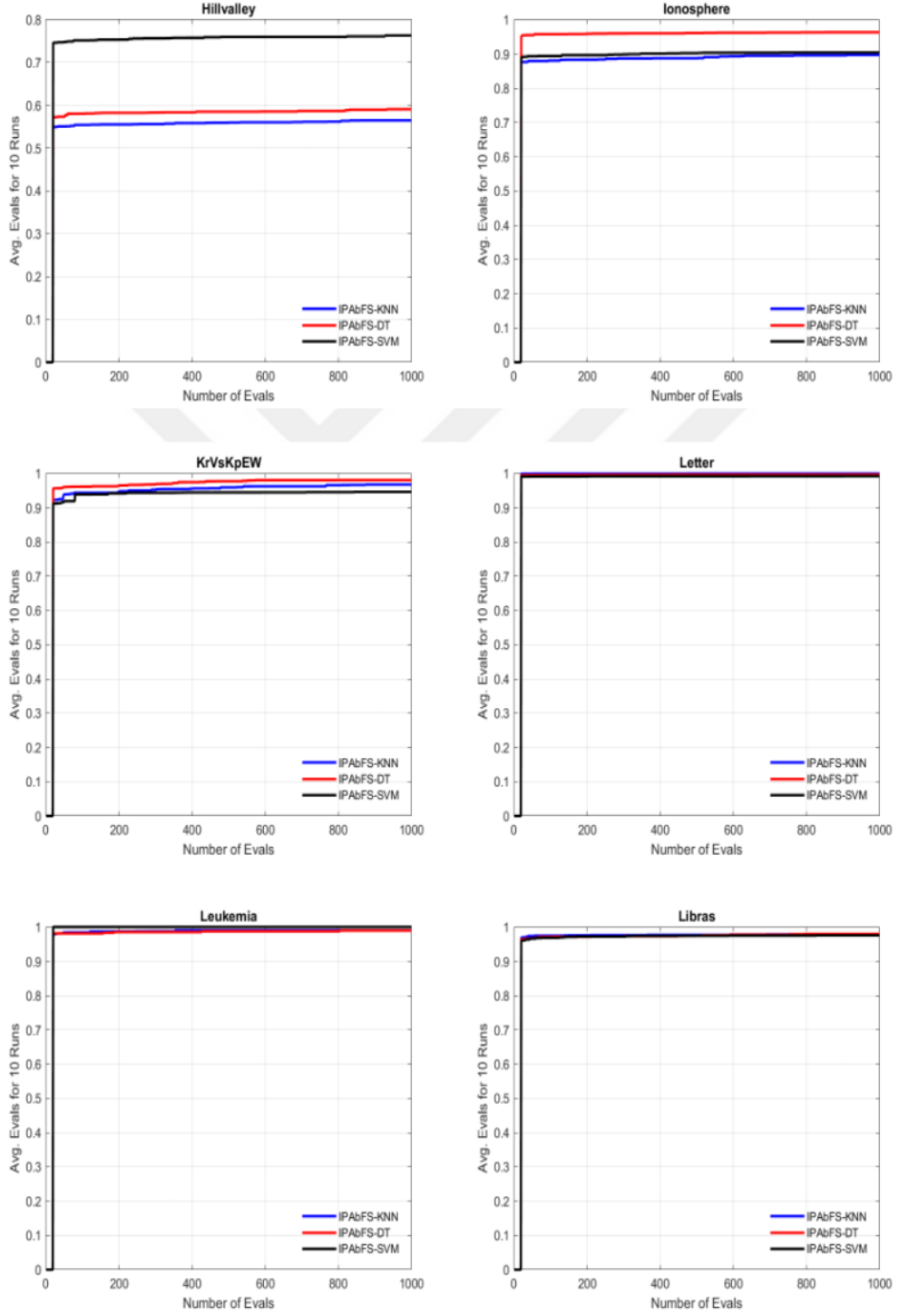
benzer yakınsama görülmüştür. Şekil 4.2 genel olarak değerlendirildiğinde ilk 1000 evaluation için en iyi yakınsamayı önerilen IPAbFS-DT göstermiştir.



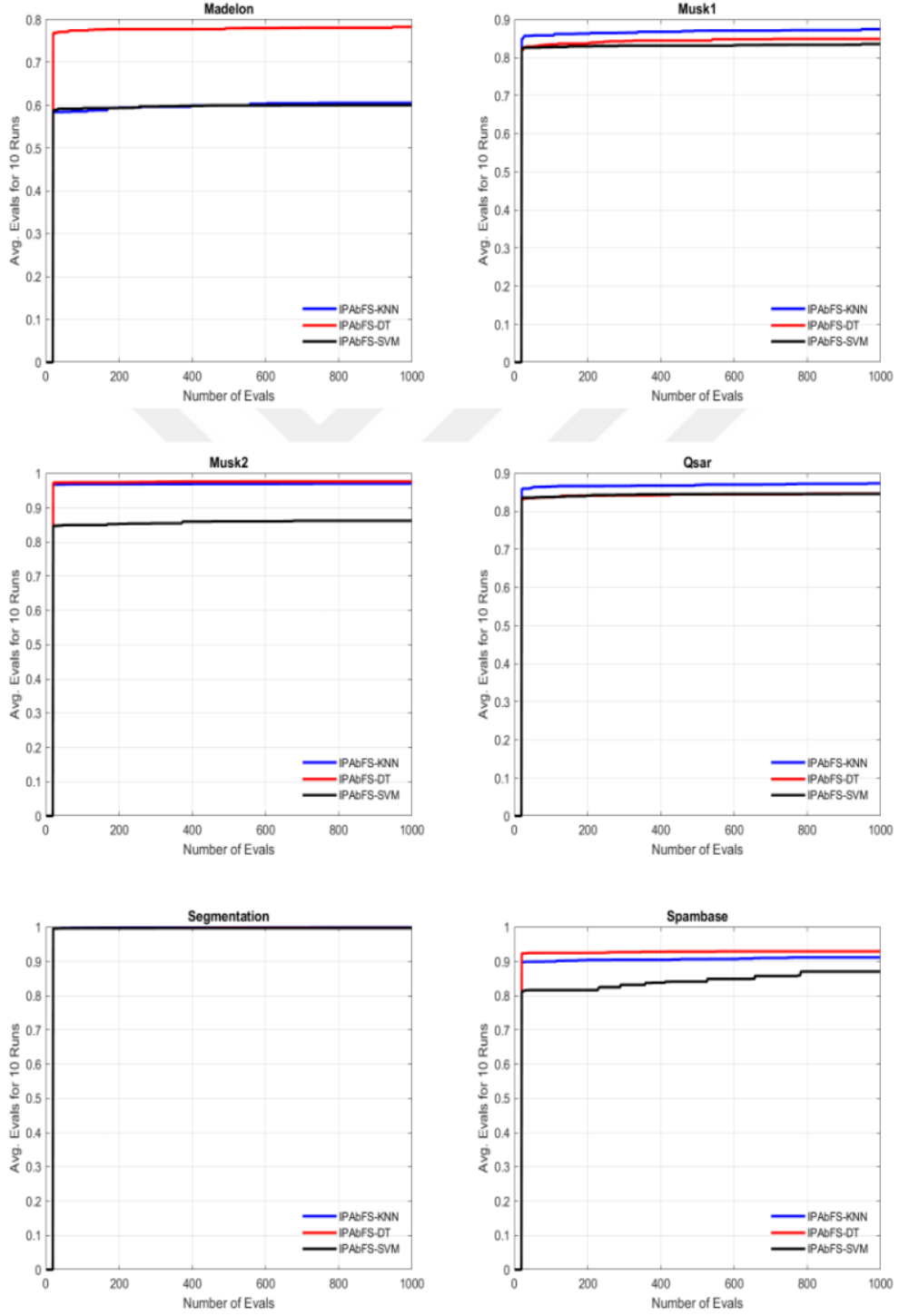
Şekil 4.2. Önerilen IPAbFS modelinin verisetlerindeki yakınsama grafikleri



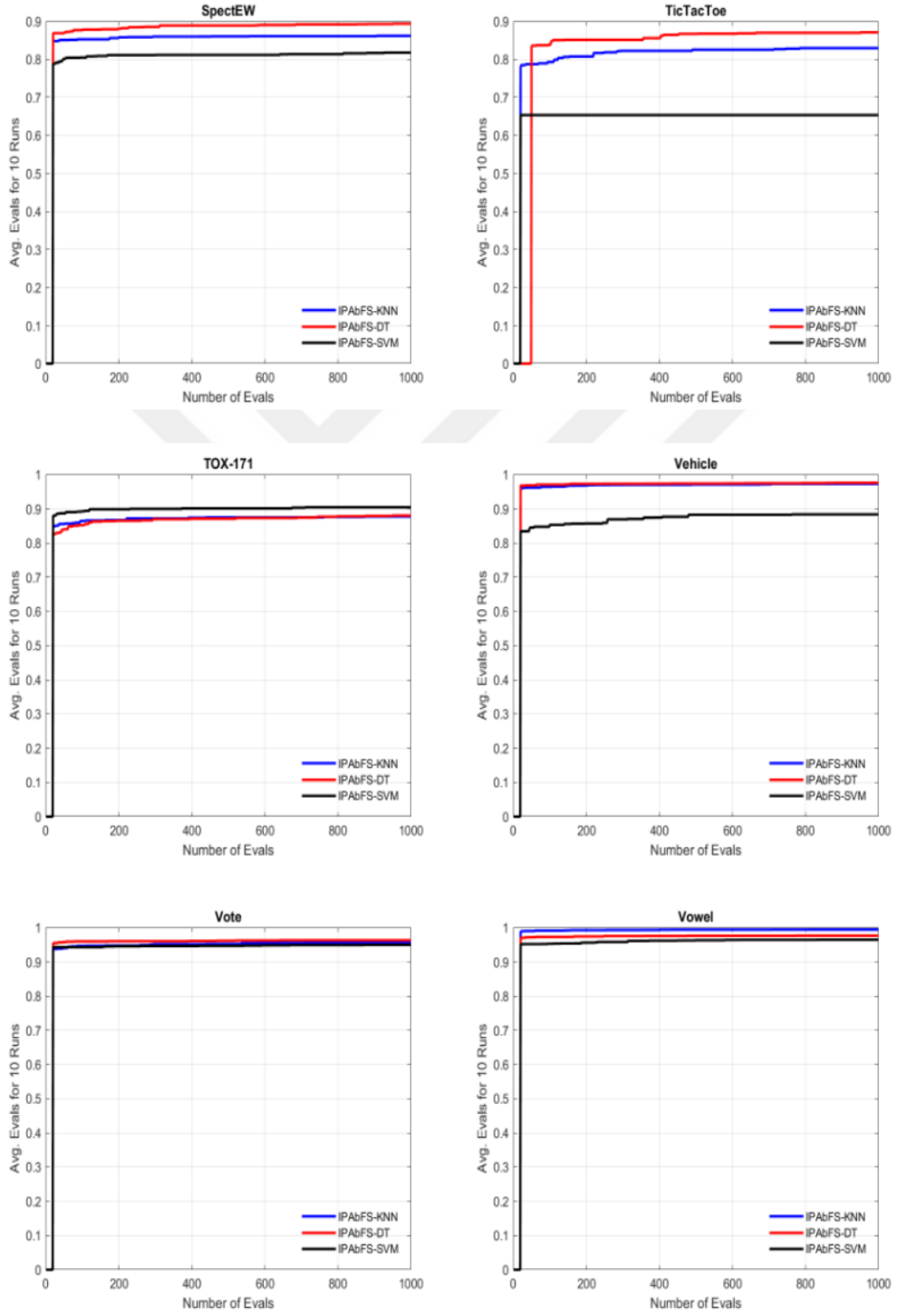
Şekil 4.2. (Devamı) Önerilen IPAbFS modelinin verisetlerindeki yakınsama grafikleri



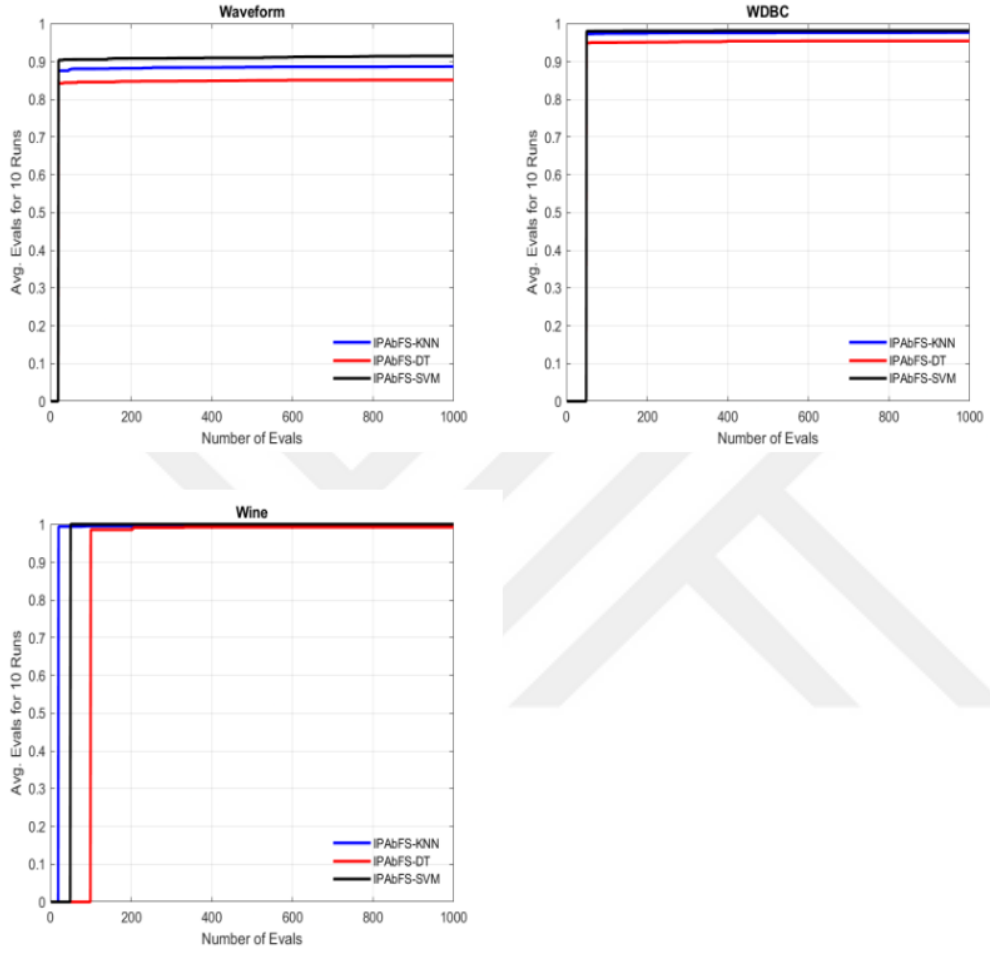
Şekil 4.2. (Devamı) Önerilen IPAbFS modelinin verisetlerindeki yakınsama grafikleri



Şekil 4.2. (Devamı) Önerilen IPAbFS modelinin verisetlerindeki yakınsama grafikleri



Şekil 4.2. (Devamı) Önerilen IPAbFS modelinin verisetlerindeki yakınsama grafikleri



Şekil 4.2. (Devamı) Önerilen IPAbFS modelinin verisetlerindeki yakınsama grafikleri

Çizelge 4.7. Önerilen IPAbFS yönteminin literatürdeki diğer yöntemler ile karşılaştırılması

Datasets	Önerilen IPAbFS			Literatürde Önerilen Diğer FS Modelleri				
	KNN	DT	SVM	BMOChOA [87]	IBEO [88]	DBOA [89]	cSG [90]	HBGWOHHO [64]
Arrhythmia	0,6763(5)	0,8139(1)	0,6921(3)	0,623(6)				
BreastCancer	0,9777(3)	0,9677(9)	0,9714(7)	0,96(10)			0,9705(8)	0,99(1)
BreastEW	0,9793(5)	0,9609(6)	0,9845(3)				0,989(2)	0,99(1)
Colon	0,871(5)	0,945(1)	0,94(2)	0,796(7)				
CongressEW	0,9705(4)	0,9724(3)	0,9586(6)					0,99(2)
CTG	0,9456(3)	0,955(2)	0,9148(5)			0,9846(1)		
Dermatology	1(1)	1(1)	1(1)		0,9702(5)			
Exactly1	1(1)	1(1)	0,688(5)			0,998(2)		0,74(3)
Exactly2	0,7987(2)	0,8024(1)	0,758(5)					0,77(3)
German	0,7664(2)	0,7544(3)	0,7762(1)					
HeartDisease	0,8637(3)	0,8652(2)	0,8637(3)					
HeartEW	0,8622(6)	0,8637(4)	0,863(5)		0,6634(7)			0,9(1)
Hepatitis	0,9312(2)	0,9989(1)	0,8916(3)	0,85(5)				
Hillvalley	0,5832(5)	0,6072(4)	0,7629(1)			0,6398(3)		
Ionosphere	0,922(11)	0,972(2)	0,9145(13))		0,943(7)	0,9548(4)	0,9735(1)	0,93(10)
Isolet	0,9979(1)	0,992(2)	0,98(3)					
KrVsKpEW	0,9757(3)	0,9905(1)	0,9503(5)					0,97(4)
Letter	0,9995(1)	0,9968(2)	0,9924(3)					
Leukemia	0,99(2)	0,9895(3)	1(1)	0,837(6)	0,9534(5)			
Libras	0,9828(2)	0,9844(1)	0,9789(3)			0,8375(5)		
Lymphography	0,9609(2)	0,9445(3)	0,942(4)	0,825(10)	0,8486(8))		0,9305(5)	0,9(6)
Madelon	0,6436(3)	0,7894(1)	0,61(4)					

MofN	1(1)	1(1)	1(1)			0,9976(2)		0,93(3)
Musk1	0,8988(3)	0,8681(4)	0,8446(5)					
Musk2	0,9731(3)	0,9777(2)	0,8765(5)					
PenglungEW	1(1)	1(1)	1(1)			0,9674(2)		1(1)
Qsar	0,8787(2)	0,8567(3)	0,8542(4)			0,9028(1)		
Segmentation	0,9996(1)	0,9981(2)	0,997(3)		0,9614(5)			
Spambase	0,9234(4)	0,9336(3)	0,8914(7)		0,9057(6)	0,9413(1)		
SpectD	0,876(4)	0,9108(1)	0,841(7)		0,7419	0,8654(5)	0,8972(3)	0,91(2)
TicTacToe	0,8695(2)	0,8866(1)	0,6537(5)					0,79(4)
TOX-171	0,9007(3)	0,9056(2)	0,9147(1)			0,8907(4)		
Vehicle	0,9799(1)	0,9797(2)	0,9203(3)		0,706(6)			
Vote	0,9627(5)	0,9667(4)	0,9553(7)			0,9816(2)	0,9833(1)	0,98(3)
Vowel	0,9962(1)	0,9808(2)	0,9679(3)			0,9473(4)		
Waveform	0,8953(2)	0,857(4)	0,9142(1)		0,8066(8)	0,8442(6)	0,7936(9)	0,86(3)
WDBC	0,9803(2)	0,9599(4)	0,9835(1)					
Wine	1(1)	0,9944(3)	1(1)				0,9982(2)	1(1)
Zoo	1(1)	1(1)	1(1)		0,948(5)		1(1)	0,97(3)
Ort,Doğruluk	0,919	0,929	0,899	0,815	0,859	0,911	0,948	0,919
Ort, Rank	2,795	2,395	3,613	7,167	6,1	3	3,444	2,941
En İyi Sonuç Sayısı	11	14	11	0	0	3	2	3

Çizelge 4.7. (Devamı) Önerilen IPAbFS yönteminin literatür ile karşılaştırılması

Datasets	Önerilen IPAbFS				Literatürde Önerilen Diğer FS Modelleri (Devamı)				
	KNN	DT	SVM	IRSA [66]	RH-BPSO [91]	MPPSO [61]	SDABWO [58]	NRSIPSO [92]	BinHOA [51]
Arrhythmia	0,6763(5)	0,8139(1)	0,6921(3)	0,6928(2)					0,6829(4)
BreastCancer	0,9777(3)	0,9677(9)	0,9714(7)	0,9861(2)		0,9722(6)	0,9755(4)		0,9728(5)
BreastEW	0,9793(5)	0,9609(6)	0,9845(3)			0,9822(4)	0,9584(6)		
Colon	0,871(5)	0,945(1)	0,94(2)		0,8288(6)			0,8874(3)	0,879(4)
CongressEW	0,9705(4)	0,9724(3)	0,9586(6)			0,9954(1)			0,9676(5)
CTG	0,9456(3)	0,955(2)	0,9148(5)		0,9288(4)				
Dermatology	1(1)	1(1)	1(1)	0,9963(2)		0,9891(3)			0,9786(4)
Exactly1	1(1)	1(1)	0,688(5)			1(1)			
Exactly2	0,7987(2)	0,8024(1)	0,758(5)			0,7601(4)			
German	0,7664(2)	0,7544(3)	0,7762(1)					0,741(4)	
HeartDisease	0,8637(4)	0,8652(3)	0,8637(4)	0,8944(1)				0,8844(2)	
HeartEW	0,8622(6)	0,8637(4)	0,863(5)			0,8659(4)	0,8694(2)		0,6731(8)
Hepatitis	0,9312(2)	0,9989(1)	0,8916(3)	0,8193(6)					0,8716(4)
Hillvalley	0,5832(5)	0,6072(4)	0,7629(1)						0,7355(2)
Ionosphere	0,922(11)	0,972(2)	0,9145(13)		0,934(9)	0,935(8)	0,9571(3)	0,9202(12)	0,9519(5)
Isolet	0,9979(1)	0,992(2)	0,98(3)					0,9467(4)	
KrVsKpEW	0,9757(3)	0,9905(1)	0,9503(5)			0,9781(2)			
Letter	0,9995(1)	0,9968(2)	0,9924(3)						
Leukemia	0,99(2)	0,9895(3)	1(1)						0,9763(4)
Libras	0,9828(2)	0,9844(1)	0,9789(3)			0,9365(4)	0,8048(6)		0,783(7)
Lymphography	0,9609(2)	0,9445(3)	0,942(4)	0,6299(11)		0,8919(7)	0,9793(1)	0,8444(9)	
Madelon	0,6436(3)	0,7894(1)	0,61(4)						0,7751(2)

MofN	1(1)	1(1)	1(1)			1(1)			
Musk1	0,8988(3)	0,8681(4)	0,8446(5)			0,9666(2)			0,9933(1)
Musk2	0,9731(3)	0,9777(2)	0,8765(5)			0,9857(1)			
PenglungEW	1(1)	1(1)	1(1)			0,9521(3)			
Qsar	0,8787(2)	0,8567(3)	0,8542(4)						
Segmentation	0,9996(1)	0,9981(2)	0,997(3)		0,9707(4)		0,9166(6)		
Spambase	0,9234(4)	0,9336(3)	0,8914(7)			0,9374(2)			0,9203(5)
SpectD	0,876(4)	0,9108(1)	0,841(7)			0,8592(6)			
TicTacToe	0,8695(2)	0,8866(1)	0,6537(5)			0,8528(3)			
TOX-171	0,9007(3)	0,9056(2)	0,9147(1)						
Vehicle	0,9799(1)	0,9797(2)	0,9203(3)			0,7553(4)	0,7328(5)		
Vote	0,9627(5)	0,9667(4)	0,9553(7)			0,9593(6)			
Vowel	0,9962(1)	0,9808(2)	0,9679(3)						0,937(5)
Waveform	0,8953(2)	0,857(4)	0,9142(1)			0,8151(7)			0,853(5)
WDBC	0,9803(2)	0,9599(4)	0,9835(1)	0,9639(3)	0,9442(5)			0,7971(6)	
Wine	1(1)	0,9944(3)	1(1)		0,9544(7)	0,9927(4)	0,98(6)	0,9833(5)	
Zoo	1(1)	1(1)	1(1)			1(1)	0,98(2)	0,9528(4)	
Ort, Doğruluk	0,919	0,929	0,899	0,855	0,927	0,930	0,915	0,884	0,872
Ort, Rank	2,795	2,395	3,613	3,714	5,833	3,609	4,100	5,33	4,375
En İyi Sonuç Sayısı	11	14	11	1	0	3	1	0	1

Çizelge 4.7’de sunulan PenglungEW, Madelon, Colon Cancer, Leukemia, Arrhythmia, Isolet ve TOX-171 olmak üzere toplam 7 büyük boyutlu datasetin tamamında önerilen IPAbFS literatürdeki diğer rakip yöntemlerinden daha yüksek doğruluk değeri elde edilmiştir. Yine Çizelge 4.7’ye göre özellikle çoklu sınıflı verisetlerinde önerilen IPAbFS’nin literatürde yer alan diğer yöntemlere göre oldukça başarılı sonuçlar aldığı anlaşılmaktadır. Ayrıca, önerilen IPAbFS, Vowel, Wine, Zoo, Letter, Lymphography, Vehicle, Segmentation, CTG, Dermatology, Waveform ve Libras çoklu sınıflı 11 datasetin 9’unda rakip yöntemlerden daha başarılı sonuçlar elde etmiştir.

Çizelge 4.7 ‘nin ortalama doğruluk satırında her bir yöntemin verisetlerindeki ortalama değeri sunulmaktadır. Bu değerler incelendiğinde önerilen IPAbFS, 26 verisetinde birinci yöntem olurken kalan 6 verisetinde ise ikinci yöntem olduğu görülmektedir. Bunlara ek olarak, Çizelge 4.7’ye göre önerilen IPAbFS için en başarılı makine öğrenmesi yönteminin DT olduğu görülmektedir. IPAbFS’de DT ile 39 datasetin 14’sinde rakip yöntemlere göre en iyi ya da eşit sonuçlar elde edilirken KNN makine öğrenmesi modeli ile 11 ve SVM ile 11 adet datasette diğer yöntemlere göre en iyi ya da eşit sonuçlar elde edilmiştir. Genel olarak yöntemlerin başarı sıralaması dikkate alındığında en başarılı yöntemin IPAbFS olduğu görülmektedir.

5. SONUÇLAR ve ÖNERİLER

Bu tezde sarmalayıcı temelli FS için ilk kez IPA temelli yeni bir öznitelik seçimi modeli önerilmiştir. Önerilen yöntemde, rastgele üretilen başlangıç çözümleri IPA ile geliştirilerek en iyi öznitelik kümesine ulaşılmaktadır. Önerilen IPAbFS, küçük, orta ve büyük boyutlu 39 adet veriseti üzerinden test edilmiştir. Literatüre sunulmuş rakip algoritmalara göre, önerilen IPAbFS yönteminin 26 verisetinde en başarılı yöntem olduğu deneysel sonuçlar ile görülmüştür. Ayrıca deneysel sonuçlar, IPAbFS'nin verisetlerindeki öznitelik sayılarını oldukça başarılı bir şekilde azaltabildiğini göstermiştir. Önerilen IPAbFS, KNN, DT ve SVM makine öğrenmesi yöntemleri üzerinden sınıflandırma gerçekleştirilerek her dataset için en iyi sınıflandırıcı belirlenmiştir. IPAbFS'nin farklı sınıflandırıcılar üzerinden elde edilen sonuçları istatistiksel olarak karşılaştırılmıştır. Ortalama doğruluk değerine IPAbFS yönteminde en başarılı sınıflandırıcı DT olduğu görülmüştür. Bununla birlikte, ortalama hesaplama zamanlarında en hızlı koşma KNN temelli IPABFS olduğu görülmüştür. Deneysel sonuçlar, Önerilen IPAbFS'nin büyük boyutlu verisetlerin de oldukça başarılı olduğunu ortaya koymuştur. IPAbFS, büyük boyutlu 7 datasetin tamamında karşılaştırılan diğer rakip yöntemlere göre daha başarılı sonuçlar üretmiştir. Tüm sonuçlar genel olarak değerlendirildiğinde IPAbFS modelinin öznitelik seçimi problemlerinin çözümünde oldukça başarılı olduğu görülmektedir.

Gelecekte IPAbFS modeli IPA'nın farklı versiyonları ile güncellenerek daha başarılı sonuçlar elde edilebilir. Ayrıca IPAbFS, diğer FS modelleriyle birleştirilerek hibrit yeni ve daha başarılı bir model geliştirilebilir. IPAbFS derin öğrenme gibi farklı makine öğrenmesi yöntemleriyle performansları değerlendirilerek kullanılabilir.

KAYNAKLAR

- [1] Elminaam, D. S. A., Nabil, A., Ibraheem, S. A., Houssein, E. H. (2021a). An Efficient Marine Predators Algorithm for Feature Selection. *IEEE Access*, 9, 60136-60153. <https://doi.org/10.1109/ACCESS.2021.3073261>
- [2] Badem, H., Turkusagi, D., Caliskan, A., Cil, Z. A. (2019). Parkinson Hastalığı Teşhisi için Yapay Arı Kolonisi Temelli Öznitelik Seçimi. *TIPTEKNO 2019 - Tıp Teknolojileri Kongresi*. <https://doi.org/10.1109/TIPTEKNO.2019.8895090>
- [3] Kira, K., Rendell, L. A. (1992). A Practical Approach to Feature Selection. *Machine Learning Proceedings 1992*, 249-256. <https://doi.org/10.1016/B978-1-55860-247-2.50037-1>
- [4] Elminaam, D. S., Ibrahim, S. A., Houssein, E. H., Elsayed, S. M. (2022). An Efficient Chaotic Gradient-Based Optimizer for Feature Selection. *IEEE Access*, 10, 9271-9286. <https://doi.org/10.1109/ACCESS.2022.3143802>
- [5] Bommert, A., Sun, X., Bischl, B., Rahnenführer, J., Lang, M. (2020). Benchmark for filter methods for feature selection in high-dimensional classification data. *Computational Statistics & Data Analysis*, 143, 106839. <https://doi.org/10.1016/J.CSDA.2019.106839>
- [6] Akcan, U. (2022). *Karınca koloni algoritması ile öznitelik seçimi*. Yüksek Lisans Tezi. Bursa Uludağ Üniversitesi Fen Bilimleri Enstitüsü. Bursa. 53s.
- [7] Li, J., Kang, H., Sun, G., Feng, T., Li, W., Zhang, W., Ji, B. (2020). IBDA: Improved Binary Dragonfly Algorithm with Evolutionary Population Dynamics and Adaptive Crossover for Feature Selection. *IEEE Access*, 8, 108032-108051. <https://doi.org/10.1109/ACCESS.2020.3001204>
- [8] Hu, Z., Bao, Y., Xiong, T., Chiong, R. (2015). Hybrid filter–wrapper feature selection for short-term load forecasting. *Engineering Applications of Artificial Intelligence*, 40, 17-27. <https://doi.org/10.1016/J.ENGAPPAI.2014.12.014>
- [9] Aljarah, I., Faris, H., Heidari, A. A., Mafarja, M. M., Al-Zoubi, A. M., Castillo, P. A., Merelo, J. J. (2021). A robust multi-objective feature selection model based on local neighborhood multi-verse optimization. *IEEE Access*, 9, 100009-100028. <https://doi.org/10.1109/ACCESS.2021.3097206>
- [10] Raymer, M. L., Punch, W. F., Goodman, E. D., Kuhn, L. A., Jain, A. K. (2000). Dimensionality reduction using genetic algorithms. *IEEE Transactions on Evolutionary Computation*, 4(2), 164-171. <https://doi.org/10.1109/4235.850656>
- [11] Cui, X., Beaver, J. M., Charles, J. S., Potok, T. E. (2008). Dimensionality reduction particle swarm algorithm for high dimensional clustering. *2008 IEEE Swarm Intelligence Symposium, SIS 2008*. <https://doi.org/10.1109/SIS.2008.4668309>
- [12] Prasartvit, T., Kaewkamnerdpong, B., Achalakul, T. (2011). Dimensional reduction based on artificial bee colony for classification problems. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and*

- [13] Martinović, G., Bajer, D., Zorić, B. (2014). A differential evolution approach to dimensionality reduction for classification needs. *International Journal of Applied Mathematics and Computer Science*, 24(1), 111-122.
<https://doi.org/10.2478/AMCS-2014-0009>
- [14] Emary, E., Zawbaa, H. M., Grosan, C., Hassenian, A. E. (2015). Feature subset selection approach by gray-wolf optimization. *Advances in Intelligent Systems and Computing*, 334, 1-13. https://doi.org/10.1007/978-3-319-13572-4_1/COVER
- [15] Deriche, M. (2009). Feature selection using ant colony optimization. *2009 6th International Multi-Conference on Systems, Signals and Devices, SSD 2009*.
<https://doi.org/10.1109/SSD.2009.4956825>
- [16] Hichem, H., Elkamel, M., Rafik, M., Mesaaoud, M. T., Ouahiba, C. (2022). A new binary grasshopper optimization algorithm for feature selection problem. *Journal of King Saud University - Computer and Information Sciences*, 34(2), 316-328.
<https://doi.org/10.1016/J.JKSUCI.2019.11.007>
- [17] Aslan, S., Demirci, S. (2020). Immune Plasma Algorithm: A Novel Meta-heuristic for Optimization Problems. *IEEE Access*.
<https://doi.org/10.1109/ACCESS.2020.3043174>
- [18] Erkin, T., Aslan, S. (2022). A New Immune Plasma Algorithm Based on Levy Flight. *2022 30th Signal Processing and Communications Applications Conference, SIU 2022*. <https://doi.org/10.1109/SIU55565.2022.9864831>
- [19] Aslan, S. (2022). An immune plasma algorithm with a modified treatment schema for UCAV path planning. *Engineering Applications of Artificial Intelligence*, 112, 104789. <https://doi.org/10.1016/J.ENGAPPAL.2022.104789>
- [20] Kısa, M., Aslan, S., Demirci, S. (2021). İmmün Plazma Algoritması ile Bilişsel Radyo Ağlarında Çok Kullanıcılı Kanal Atama Probleminin Çözülmesi. *Avrupa Bilim ve Teknoloji Dergisi*, 29(29), 304-310. <https://doi.org/10.31590/EJOSAT.1024761>
- [21] Arslan, S. (2021). Zaman Serisi Tahmin Probleminin İmmün Plazma Programlama Kullanılarak Çözülmesi. *Avrupa Bilim ve Teknoloji Dergisi*, 29(29), 219-224. <https://doi.org/10.31590/EJOSAT.1010078>
- [22] Akay, B., Karaboga, D. (2012). Artificial bee colony algorithm for large-scale problems and engineering design optimization. *Journal of Intelligent Manufacturing*, 23(4), 1001-1014. <https://doi.org/10.1007/S10845-010-0393-4/METRICS>
- [23] Subanya, B., Rajalaxmi, R. R. (2014). Feature selection using artificial bee colony for cardiovascular disease classification. *2014 International Conference on Electronics and Communication Systems, ICECS 2014*.
<https://doi.org/10.1109/ECS.2014.6892729>
- [24] Kumar, S., Sikander, A. (2022). Optimum Mobile Robot Path Planning Using Improved Artificial Bee Colony Algorithm and Evolutionary Programming. *Arabian Journal*

for Science and Engineering, 47(3), 3519-3539. <https://doi.org/10.1007/S13369-021-06326-8/TABLES/14>

- [25] Karaboga, D. (2005). An idea based on honey bee swarm for numerical optimization. http://abc.erciyes.edu.tr/pub/tr06_2005.pdf
- [26] Karaboga, D., Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, 39(3), 459-471. <https://doi.org/10.1007/S10898-007-9149-X/METRICS>
- [27] Derviş Karaboğa. (2018). *Yapay Zeka Optimizasyon Algoritmaları*. 6.Basım . Nobel Yayıncılık. . ISBN: 978-605-133-764-7. Türkiye. 231s.
- [28] John H. Holland. (1975). *Adaptation in Natural and Artificial Systems*. 1. Basım. University of Michigan Press.
- [29] Grefenstette, J. J. (1986). Optimization of Control Parameters for Genetic Algorithms. *IEEE Transactions on Systems, Man and Cybernetics*, 16(1), 122-128. <https://doi.org/10.1109/TSMC.1986.289288>
- [30] Man, K. F., Tang, K. S., Kwong, S. (1996). Genetic algorithms: Concepts and applications. *IEEE Transactions on Industrial Electronics*, 43(5), 519-534. <https://doi.org/10.1109/41.538609>
- [31] Bhoskar, T., Kulkarni, O. K., Kulkarni, N. K., Patekar, S. L., Kakandikar, G. M., Nandedkar, V. M. (2015). Genetic Algorithm and its Applications to Mechanical Engineering: A Review. *Materials Today: Proceedings*, 2(4-5), 2624-2630. <https://doi.org/10.1016/J.MATPR.2015.07.219>
- [32] Holland, J. H. (1984). Genetic Algorithms and Adaptation. *Adaptive Control of Ill-Defined Systems*, 317-333. https://doi.org/10.1007/978-1-4684-8941-5_21
- [33] Angeline, P. J. (1995). Evolution Revolution: An Introduction to the Special Track on Genetic and Evolutionary Programming. *IEEE Intelligent Systems*, 10(03), 6-10. <https://doi.org/10.1109/MIS.1995.10027>
- [34] Galletly, J. E. (1992). An Overview of Genetic Algorithms. *Kybernetes*, 21(6), 26-30. <https://doi.org/10.1108/EB005943/FULL/XML>
- [35] Beyer, H.-G., Beyer, H.-G., Schwefel, H.-P., Schwefel, H. P. (2002). Evolution strategies – A comprehensive introduction. *Natural Computing* 1(1), 3-52. <https://doi.org/10.1023/A:1015059928466>
- [36] Kramer, O. (2017). Genetic algorithms. *Studies in Computational Intelligence*, 679, 11-19. https://doi.org/10.1007/978-3-319-52156-5_2/TABLES/1
- [37] Shi, Y., Eberhart, R. C. (1999). Empirical study of particle swarm optimization. *Proceedings of the 1999 Congress on Evolutionary Computation, CEC 1999*, 3, 1945-1950. <https://doi.org/10.1109/CEC.1999.785511>

- [38] Kennedy, J., Eberhart, R. (t.y.). Particle swarm optimization. *Proceedings of ICNN'95 - International Conference on Neural Networks*, 4, 1942-1948. <https://doi.org/10.1109/ICNN.1995.488968>
- [39] Du, K.-L., Swamy, M. N. S. (2016). Particle Swarm Optimization. *Search and Optimization by Metaheuristics*, 153-173. https://doi.org/10.1007/978-3-319-41192-7_9
- [40] Wang, D., Tan, D., Liu, L. (2018). Particle swarm optimization algorithm: an overview. *Soft Computing*, 22(2), 387-408. <https://doi.org/10.1007/S00500-016-2474-6/FIGURES/1>
- [41] Eberhart, R. C., Hu, X. (1999). Human tremor analysis using particle swarm optimization. *Proceedings of the 1999 Congress on Evolutionary Computation, CEC 1999*, 3, 1927-1930. <https://doi.org/10.1109/CEC.1999.785508>
- [42] Guedria, N. (2016). Improved accelerated PSO algorithm for mechanical engineering optimization problems. *Applied Soft Computing*, 40, 455-467. <https://doi.org/10.1016/J.ASOC.2015.10.048>
- [43] Mahor, A., Prasad, V., Rangnekar, S. (2009). Economic dispatch using particle swarm optimization: A review. *Renewable and Sustainable Energy Reviews*, 13(8), 2134-2141. <https://doi.org/10.1016/J.RSER.2009.03.007>
- [44] Gopakumar, A., Jacob, L. (2008). Localization in wireless sensor networks using particle swarm optimization. *IET Conference Publications*, 535 CP, 227-230. <https://doi.org/10.1049/CP:20080185>
- [45] Cui, X., Potok, T. E., Palathingal, P. (2005). Document clustering using particle swarm optimization. *Proceedings - 2005 IEEE Swarm Intelligence Symposium, SIS 2005*, 191-197. <https://doi.org/10.1109/SIS.2005.1501621>
- [46] Poli, R., Kennedy, J., Blackwell, T. (2007). Particle swarm optimization. *Swarm Intelligence*, 1(1), 33-57. <https://doi.org/10.1007/S11721-007-0002-0>
- [47] Özsağlam, M. Y., Çunkaş, M. (2008). Optimizasyon Problemlerinin Çözümü için Parçacık Sürü Optimizasyonu Algoritması. *Journal of Polytechnic*, 11(4), 299-305. <https://doi.org/10.2339/2008.11.4>
- [48] Dhiman, G., Kumar, V. (2019). Seagull optimization algorithm: Theory and its applications for large-scale industrial engineering problems. *Knowledge-Based Systems*, 165, 169-196. <https://doi.org/10.1016/J.KNOSYS.2018.11.024>
- [49] Kumar, V., Kumar, Di., Kaur, M., Singh, Di., Idris, S. A., Alshazly, H. (2021). A Novel Binary Seagull Optimizer and its Application to Feature Selection Problem. *IEEE Access*, 9, 103481-103496. <https://doi.org/10.1109/ACCESS.2021.3098642>
- [50] MiarNaeimi, F., Azizyan, G., Rashki, M. (2021). Horse herd optimization algorithm: A nature-inspired algorithm for high-dimensional optimization problems. *Knowledge-Based Systems*, 213, 106711. <https://doi.org/10.1016/J.KNOSYS.2020.106711>

- [51] Elmanakhly, D. A., Saleh, M., Rashed, E. A., Abdel-Basset, M. (2022). BinHOA: Efficient Binary Horse Herd Optimization Method for Feature Selection: Analysis and Validations. *IEEE Access*, 10, 26795-26816. <https://doi.org/10.1109/ACCESS.2022.3156593>
- [52] Faramarzi, A., Heidarinejad, M., Mirjalili, S., Gandomi, A. H. (2020). Marine Predators Algorithm: A nature-inspired metaheuristic. *Expert Systems with Applications*, 152, 113377. <https://doi.org/10.1016/J.ESWA.2020.113377>
- [53] Elminaam, D. S. A., Nabil, A., Ibraheem, S. A., Houssein, E. H. (2021b). An Efficient Marine Predators Algorithm for Feature Selection. *IEEE Access*, 9, 60136-60153. <https://doi.org/10.1109/ACCESS.2021.3073261>
- [54] Faramarzi, A., Heidarinejad, M., Stephens, B., Mirjalili, S. (2020). Equilibrium optimizer: A novel optimization algorithm. *Knowledge-Based Systems*, 191, 105190. <https://doi.org/10.1016/J.KNOSYS.2019.105190>
- [55] Van Laarhoven, P. J. M., Aarts, E. H. L. (1987). Simulated annealing. *Simulated Annealing: Theory and Applications*, 7-15. https://doi.org/10.1007/978-94-015-7744-1_2
- [56] Guha, R., Ghosh, K. K., Bera, S. K., Sarkar, R., Mirjalili, S., Kumar Bera, S. (2020). BEO: Binary Equilibrium Optimizer Combined with Simulated Annealing for Feature Selection. <https://doi.org/10.21203/RS.3.RS-28683/V1>
- [57] Hayyolalam, V., Pourhaji Kazem, A. A. (2020). Black Widow Optimization Algorithm: A novel meta-heuristic approach for solving engineering optimization problems. *Engineering Applications of Artificial Intelligence*, 87, 103249. <https://doi.org/10.1016/J.ENGAPPAL.2019.103249>
- [58] Hu, G., Du, B., Wang, X., Wei, G. (2022). An enhanced black widow optimization algorithm for feature selection. *Knowledge-Based Systems*, 235, 107638. <https://doi.org/10.1016/J.KNOSYS.2021.107638>
- [59] Saremi, S., Mirjalili, S., Lewis, A. (2017). Grasshopper Optimisation Algorithm: Theory and application. *Advances in Engineering Software*, 105, 30-47. <https://doi.org/10.1016/J.ADVENGSOFT.2017.01.004>
- [60] Xu, Z., Gui, W., Heidari, A. A., Liang, G., Chen, H., Wu, C., Turabieh, H., Mafarja, M. (2021). Spiral Motion Mode Embedded Grasshopper Optimization Algorithm: Design and Analysis. *IEEE Access*, 9, 71104-71132. <https://doi.org/10.1109/ACCESS.2021.3077616>
- [61] Kılıç, F., Kaya, Y., Yildirim, S. (2021). A novel multi population based particle swarm optimization for feature selection. *Knowledge-Based Systems*, 219, 106894. <https://doi.org/10.1016/J.KNOSYS.2021.106894>
- [62] Wang, P., Xue, B., Liang, J., Zhang, M. (2021). Multiobjective Differential Evolution for Feature Selection in Classification. *IEEE Transactions on Cybernetics*. <https://doi.org/10.1109/TCYB.2021.3128540>

- [63] Heidari, A. A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., Chen, H. (2019). Harris hawks optimization: Algorithm and applications. *Future Generation Computer Systems*, 97, 849-872. <https://doi.org/10.1016/J.FUTURE.2019.02.028>
- [64] Al-Wajih, R., Abdulkadir, S. J., Aziz, N., Al-Tashi, Q., Talpur, N. (2021). Hybrid Binary Grey Wolf with Harris Hawks Optimizer for Feature Selection. *IEEE Access*, 9, 31662-31677. <https://doi.org/10.1109/ACCESS.2021.3060096>
- [65] Abualigah, L., Elaziz, M. A., Sumari, P., Geem, Z. W., Gandomi, A. H. (2022). Reptile Search Algorithm (RSA): A nature-inspired meta-heuristic optimizer. *Expert Systems with Applications*, 191, 116158. <https://doi.org/10.1016/J.ESWA.2021.116158>
- [66] Elgamal, Z., Sabri, A. Q. M., Tubishat, M., Tbaishat, D., Makhadmeh, S. N., Alomari, O. A. (2022). Improved Reptile Search Optimization Algorithm Using Chaotic Map and Simulated Annealing for Feature Selection in Medical Field. *IEEE Access*, 10, 51428-51446. <https://doi.org/10.1109/ACCESS.2022.3174854>
- [67] Uymaz, S. A., Tezel, G., Yel, E. (2015). Artificial algae algorithm (AAA) for nonlinear global optimization. *Applied Soft Computing*, 31, 153-171. <https://doi.org/10.1016/J.ASOC.2015.03.003>
- [68] Turkoglu, B., Uymaz, S. A., Kaya, E. (2022). Binary Artificial Algae Algorithm for feature selection. *Applied Soft Computing*, 120, 108630. <https://doi.org/10.1016/J.ASOC.2022.108630>
- [69] Ross Quinlan, by J., Kaufmann Publishers, M., Salzberg, S. L. (1994). C4.5: Programs for Machine Learning by J. Ross Quinlan. Morgan Kaufmann Publishers, Inc., 1993. *Machine Learning* 1994 16:3, 16(3), 235-240. <https://doi.org/10.1007/BF00993309>
- [70] Quinlan, J. R. (1996). Improved Use of Continuous Attributes in C4.5. *Journal of Artificial Intelligence Research*, 4, 77-90. <https://doi.org/10.48550/arxiv.cs/9603103>
- [71] Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, 1(1), 81-106. <https://doi.org/10.1007/BF00116251>
- [72] Sathyadevan, S., Nair, R. R. (2015). Comparative analysis of decision tree algorithms: Id3, c4.5 and random forest. *Smart Innovation, Systems and Technologies*, 31, 549-562. https://doi.org/10.1007/978-81-322-2205-7_51
- [73] Breiman, L. (1984). *Classification and Regression Trees*. Taylor & Francis. ISBN: 978-0412048418. USA. 368s.
- [74] Chou, P. A. (1991). Optimal Partitioning for Classification and Regression Trees. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(4), 340-354. <https://doi.org/10.1109/34.88569>
- [75] Mingers, J. (1989). An Empirical Comparison of Pruning Methods for Decision Tree Induction. *Machine Learning*, 4(2), 227-243. <https://doi.org/10.1023/A:1022604100933/METRICS>

- [76] Cover, T. M., Hart, P. E. (1967). Nearest Neighbor Pattern Classification. *IEEE Transactions on Information Theory*, 13(1), 21-27. <https://doi.org/10.1109/TIT.1967.1053964>
- [77] Coomans, D., Massart, D. L. (1982). Alternative k-nearest neighbour rules in supervised pattern recognition : Part 1. k-Nearest neighbour classification by using alternative voting rules. *Analytica Chimica Acta*, 136(C), 15-27. [https://doi.org/10.1016/S0003-2670\(01\)95359-0](https://doi.org/10.1016/S0003-2670(01)95359-0)
- [78] Uddin, S., Haque, I., Lu, H., Moni, M. A., Gide, E. (2022). Comparative performance analysis of K-nearest neighbour (KNN) algorithm and its different variants for disease prediction. *Scientific Reports*, 12(1), 1-11. <https://doi.org/10.1038/s41598-022-10358-x>
- [79] Amra, I. A. A., Maghari, A. Y. A. (2017). Students performance prediction using KNN and Naïve Bayesian. *ICIT 2017 - 8th International Conference on Information Technology, Proceedings*, 909-913. <https://doi.org/10.1109/ICITECH.2017.8079967>
- [80] Güvenç, E., Sakal, M., Çetin, G., ÖZKARACA, O. (2022). Öğrencilerin Dersteki Niteliklerinin Makine Öğrenmesi Teknikleri Kullanılarak Sınıflandırılması. *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, 10(3), 1359-1371. <https://doi.org/10.29130/DUBITED.1017202>
- [81] Almomany, A., Ayyad, W. R., Jarrah, A. (2022). Optimized implementation of an improved KNN classification algorithm using Intel FPGA platform: Covid-19 case study. *Journal of King Saud University-Computer and Information Sciences*, 34(6), 3815-3827. <https://doi.org/10.1016/J.JKSUCI.2022.04.006>
- [82] Guyon, I., Weston, J., Barnhill, S., Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine Learning*, 46(1-3), 389-422. <https://doi.org/10.1023/A:1012487302797/METRICS>
- [83] Cortes, C., Vapnik, V., Saitta, L. (1995). Support-vector networks. *Machine Learning*, 20(3), 273-297. <https://doi.org/10.1007/BF00994018>
- [84] Schölkopf, B. (1998). SVMs - A practical consequence of learning theory. *IEEE Intelligent Systems and Their Applications*, 13(4), 18-21. <https://doi.org/10.1109/5254.708428>
- [85] Kavzoğlu, T., Çölkesen, İ. (2010). Destek vektör makineleri ile uydu görüntülerinin sınıflandırılmasında kernel fonksiyonlarının etkilerinin incelenmesi. *Harita Dergisi*, 76(144), 73-82. <http://search/yayin/detay/102501>
- [86] Hearst, M. A., Dumais, S. T., Osuna, E., Platt, J., Scholkopf, B. (1998). Support vector machines. *IEEE Intelligent Systems and their Applications*, 13(4), 18-28. <https://doi.org/10.1109/5254.708428>
- [87] Piri, J., Mohapatra, P., Pradhan, M. R., Acharya, B., Patra, T. K. (2022). A Binary Multi-Objective Chimp Optimizer with Dual Archive for Feature Selection in the Healthcare Domain. *IEEE Access*, 10, 1756-1774. <https://doi.org/10.1109/ACCESS.2021.3138403>

- [88] Elmanakhly, D. A., Saleh, M. M., Rashed, E. A. (2021). An Improved Equilibrium Optimizer Algorithm for Features Selection: Methods and Analysis. *IEEE Access*, 9, 120309-120327. <https://doi.org/10.1109/ACCESS.2021.3108097>
- [89] Tubishat, M., Alswaitti, M., Mirjalili, S., Al-Garadi, M. A., Alrashdan, M. T., Rana, T. A. (2020). Dynamic butterfly optimization algorithm for feature selection. *IEEE Access*, 8, 194303-194314. <https://doi.org/10.1109/ACCESS.2020.3033757>
- [90] Ewees, A. A., Gaheen, M. A., Yaseen, Z. M., Ghoniem, R. M. (2022). Grasshopper Optimization Algorithm with Crossover Operators for Feature Selection and Solving Engineering Problems. *IEEE Access*, 10, 23304-23320. <https://doi.org/10.1109/ACCESS.2022.3153038>
- [91] Luo, C., Wang, S., Li, T., Chen, H., Lv, J., Yi, Z. (2022). Large-Scale Meta-Heuristic Feature Selection Based on BPSO Assisted Rough Hypercuboid Approach. *IEEE Transactions on Neural Networks and Learning Systems*. <https://doi.org/10.1109/TNNLS.2022.3171614>
- [92] Feng, J., Gong, Z. (2022). A Novel Feature Selection Method With Neighborhood Rough Set and Improved Particle Swarm Optimization. *IEEE Access*, 10, 33301-33312. <https://doi.org/10.1109/ACCESS.2022.3162074>