GÜNCEL HESAPSAZ ZEKÂ ALGORİTMALARIYLA BÜYÜK SOSYAL VERİ ANALİZİ

Ümit CAN

Doktora Tezi
Yazılım Mühendisliği Anabilim Dalı
Danışman: Prof. Dr. Bilal ALATAŞ
AĞUSTOS-2019
T.C.
FIRAT ÜNİVERİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

GÜNCEL HESAPSAL ZEKA ALGORİTMALARIYLA BÜYÜK SOSYAL VERİ ANALİZİ

DOKTORA TEZİ

Yük. Müh. Ümit CAN
(141137201)

Tezin Enstitüye Verildiği Tarih : 15 Ağustos 2019
Tezin Savunulduğu Tarih : 20 Eylül 2019

Tez Danışmanı : Prof. Dr. Bilal ALATAŞ (F.Ü.)
Diğer Jüri Üyeleri : Prof. Dr. Nizamettin AYDIN (Y.T.Ü.)
Prof. Dr. Ahmet Bedri ÖZER (M.T.Ü.)
Doç. Dr. Ertan ONUR (ODTÜ)
Dr. Öğr. Üyesi Mustafa ULAŞ (F.Ü.)

EYLÜL-2019
ÖNSÖZ


Doktora çalışmanın süresince, değerli görüş ve katkılarıyla beni yönlendiren, her konuda önerileri ile desteğini esirgemeyen, bilimsel çalışma disiplinini bize kazandıran ve kıymetli tecrübelerinden faydalanıyorum teşhımanım Sayın Prof. Dr. Bilal ALATAŞ’a ve çalışmalarım sırasında benden desteği ve görüşlerini esirgemeyen Dr. Öğr. Üyesi Soner KIZILOLUK’a ve Dr. Sinem AKYOL’a teşekkür ederim. Ayrıca zaman varlıklarıyla bana destek olan sevgili eşim Nihal’e, oğlum Uras’a, anneme, babama, kız kardeşime ve sevgili dostlarına sonsuz teşekkürler.

Ümit CAN
ELAZIĞ-2019
İÇİNDEKİLER

ÖNSÖZ ......................................................................................................................... II
İÇİNDEKİLER .............................................................................................................. III
ÖZET ............................................................................................................................... VI
SUMMARY .................................................................................................................. VII
ŞEKİLLER LİSTESİ ......................................................................................................... VIII
TABLOLAR LİSTESİ ...................................................................................................... XIII
KISALTMALAR .............................................................................................................. XXVIII
SEMBOLLER LİSTESİ ...................................................................................................... XVIII

1. GİRİŞ ......................................................................................................................... 1
  1.1. Literatür Taraması .................................................................................................. 3
  1.2. Tezin Amacı .......................................................................................................... 7
  1.3. Tezin Yapısı .......................................................................................................... 8
2. SOSYAL AĞ ANALİZİ ............................................................................................. 10
  2.1. Sosyal Ağ Analizi Nedir ....................................................................................... 10
3. ÇEVRİMİÇİ SOSYAL AĞ ANALİZİ PROBLEMLERİ ............................................. 13
  3.1. Topluluk Tespiti .................................................................................................... 14
  3.1.1. İlgili Madenciliği ............................................................................................ 18
  3.2. Tutum Tespiti ....................................................................................................... 20
  3.3. Gizlilik Hakkını Koruyan Yakınlık Tespiti .......................................................... 22
  3.4. Anomali Tespiti .................................................................................................... 25
  3.4.1. Siber Suçlar ..................................................................................................... 27
  3.4.1.1. Siber Zorbalık Tespiti .................................................................................. 28
  3.4.1.2. Spamlama Tespiti ....................................................................................... 32
  3.4.1.3. Zararlı Yazılım Tespiti ................................................................................. 33
  3.4.1.4. Sahtekarlık Tespiti ...................................................................................... 36
  3.4.2. Söylenti Tespiti ............................................................................................... 42
  3.5. İryony ve Kinaye Tespiti ...................................................................................... 45
  3.6. Rol Madenciliği .................................................................................................... 47
  3.6.1. Fikir Lideri Tespiti ......................................................................................... 48
  3.7. Konu ve Olay Tespiti ............................................................................................ 51
  3.7.1. Tartışmalı Konuların Tespiti .......................................................................... 54
3.8. Nedenselik Tespiti
BÜYÜK VERİ
4.1. Büyük Veri ve ÖZELLİKLERİ
4.2. Büyük Veri Analiz Araçları
5. OPTİMİZASYON
5.1. Optimizasyon NEDİR
5.2. Sezgisel Algoritmalar
UYGULAMA ESNASINDA KULLANILAN ALGORİTMALAR,
YÖNTEM VE VERİ SETLERİ
6.1. Gri Kurt Optimizasyon Algoritması
6.1.1. Algoritmanın İlham Kaynağı
6.1.2. Matematiksel Model ve Algoritma
6.1.2.1. Sosyal Hiyerarşi
6.1.2.2. Avın Çevresini Sarma
6.1.2.3. Avlanma
6.1.2.4. Ava Saldırı Sömürme
6.1.2.5. Av Arama (Keşif)
6.2. Balina Optimizasyon Algoritması
6.2.1. Algoritmanın İlham Kaynağı
6.2.2. Matematiksel Model ve Optimizasyon Algoritması
6.2.2.1. Avın Çevrelenmesi
6.2.2.2 Kabarcık-ağ Saldırı Yöntemi
6.2.2.3. Av Arama
6.3. Büyük Patlama BüyüK Çöküş Algoritması
6.4. Hibrit Balina Optimizasyonu-Büyük Patlama BüyüK Çöküş Algoritması
6.5. Veri Ön İşleme Adımları
6.5.1. KNIME’da Veri Ön İşleme Aşamaları
6.6. Tutum Veri Setleri
6.6.1. Feminist Hareket Veri Seti
6.6.2. Ateizm Veri Seti
6.6.3. Hillary Clinton Veri Seti
6.6.4. Kürtaj Veri Seti
6.7. Kodlama Aşaması
6.7.1. Algoritmanın Başlangıç Popülasyonu ve Sonlandırma Şartı Değerleri
6.7.2. Jaccard Benzeriliği ve Uygunluk Fonksiyonu
ÖZET


Bu tez çalışmasında çevrimiçi sosyal ağlar, sosyal ağ analizi ve literatürde ilk defa yıllar bir adet çevrimiçi sosyal ağ analizi problemi tek bir çalışma altında detaylı bir şekilde ele alınmıştır. Çevrimiçi sosyal ağ analizi problemlerinden oldukça popüler ve ilgi çekici biri olan Tutum Tespiti problemi ilk defa bu çalışmada bir optimizasyon problemi olarak ele alınmıştır. Optimizasyon çalışması sırasında biri ilk defa burada önerilen yeni bir hibrit algoritma olmak üzere üç adet sezgisel optimizasyon algoritması kullanılmış ve literatürde kullanılan otuz iki adet sınıflandırma algoritması ile tutum sınıflandırması sonuçları kıyaslanmıştır. Kullanılan yöntem optimizasyon tekniğini kullanılarak sosyal ağ verilerinin doğru sınıflandırılması sağlanamaya dayanmaktadır. Önerilen yöntem henüz çok yeni olması rağmen umut verici sonuçlar elde edilmiştir.

Anahtar Kelimeler: Sosyal Ağlar, Sosyal Ağ Analizi, Sosyal Ağ Problemleri, Sosyal Ağ Analizi Yöntemleri
SUMMARY

Big Social Data Analysis with Computational Intelligent Algorithms

Social networks have long been used by various disciplines that are interested in social science or who feel the need to use this concept in their own right. This concept has undergone qualitative and quantitative changes in the field of application in the new century. New information technologies have led to the rapid and effective growth of social networks. These platforms, originally intended for communication between individuals or groups, have now evolved into a platform where millions of people reflect their political views, personal sentiments, or economic preferences. The amount of instant data they produce has reached incredible proportions. These data can be called social network data. The analysis of this data is within the scope of social network analysis. There are currently many social network analysis problems and their solution methods. But nowadays new problems arise with online social networks and various methods are used to solve these problems. These methods provide a solution to a certain extent but still need to be developed.

In this thesis, online social networks, social network analysis and twenty-one online social network analysis problems are discussed in detail in a single study for the first time in the literature. Stance Detection problem, which is one of the most popular and interesting online social network analysis problems, was first considered as an optimization problem in this study. During the optimization study, three heuristic optimization algorithms were used, one of which was proposed for the first time, a new hybrid algorithm and the results of thirty-two classification algorithms used in the literature were compared. The method used is based on ensuring accurate classification of social network data using optimization technique. Although the proposed method is very new, promising results have been achieved.

Key Words: Social Network, Social Network Analysis, Social Network Problems, Social Network Analysis Methods.
ŞEKİLLER LİSTESİ

Sayfa No

Şekil 1.1. Basit bir sosyal ağ diyagramı .................................................................1
Şekil 2.1. Zachary Karate Kulübü ağı.................................................................11
Şekil 2.2. Haziran 2018 Sosyal Ağ sitelerinin aktif kullanıcı sayıları ..................12
Şekil 3.1. Sosyal ağ analizi problemleri ..............................................................13
Şekil 3.2. Sahtekarlık alanlarının genel taksonomisi ..............................................37
Şekil 4.1. Büyük veri ve 7V ..................................................................................62
Şekil 4.2. Büyük veri ve birleştirilmiş bilgi yönetimi ..............................................66
Şekil 5.1. Optimizasyon için matematiksel modeller ..............................................68
Şekil 5.2. Sezgisel yöntemler .................................................................................72
Şekil 6.1. Gri kurt hiyerarşisi (baskılık yukarıdan aşağıya doğru azalır) ..........74
Şekil 6.2. Gri kurtların avlanma davranışları: (A) Avı takip etme, kovalama ve yaklaşma. (B-D) Avın peşinde olma, etrafını çevirme ve taciz etme. (E) Sabit konum alma ve ava saldırdı .........................................................75
Şekil 6.3. 2D ve 3D konum vektörleri ve bir sonraki muhtemel konumları ..........78
Şekil 6.4. GKO’da konum güncelleme ................................................................78
Şekil 6.5. Ava saldıri ile av arayışının karşılaştırılması.......................................79
Şekil 6.6. GKO algoritmasının sözde kodu ............................................................81
Şekil 6.7. Kambur balinanın beslenme modeli .......................................................83
Şekil 6.8. Aramaajanlarının 2 ve 3 boyutlu pozisyon vektörleri ve olası diğer pozisyonları .................................................................85
Şekil 6.9. (a) Daralan çevrelene mekanizması ve (b) Spiral hareket ....................85
Şekil 6.10. Kambur balinanın beslenme modeli ....................................................87
Şekil 6.11. BOA algoritmasının sözde-kodu ..........................................................87
Şekil 6.12. Rosenbrock Fonksiyonu için X1 ve X2 parametreleri ile gösterilen iki boyutlu durumda aday çözümlerin dağılımı ve onların ‘o’ ile gösterilen kütle merkezi. ..................................................................................................................89
Şekil 6.13. İterasyondan sonra Rosenbrock Fonksiyonu için X1 ve X2 parametreleri ile gösterilen iki boyutlu durumda aday çözümlerin dağılımı ve onların ‘o’ ile
gösterilen kütle merkezi.................................................................91
Şekil 6.14. 500 iterasyondan sonra aday çözümlerin kütle merkezinde toplanması. 91
Şekil 6.15. HBO-BPBÇ algoritmasonun aksi diyagramı.................................................................93
Şekil 6.16. KNIME’da yapılan işlemlerin aksi şeması .................................................................95
Şekil 6.17. Doküman oluşturma aşamaları.....................................................................................96
Şekil 6.18. KNIME programındaki metin ön işleme basamakları.................................................97
Şekil 6.19. Feminizm hakkındaki tweetlerinin eğitim ve test verileri. .................................101
Şekil 6.20. Ateizm tweetlerinin eğitim ve test verileri. .................................................................101
Şekil 6.21. Hillary Clinton hakkındaki tweetlerin eğitim ve test verileri. .................................102
Şekil 6.22. Kürtajın yasallaştırılması hakkındaki tweetlerin eğitim ve test verileri ..............103
Şekil 7.1. Feminist Hareket Veri Seti’ni kullanılarak GKO, BOA ve HBO-BPBÇ
algoritmalarının 100 popülasyon 100 iterasyon parametrelerindeki uygunluk
degeri disimi .................................................................................................109
Şekil 7.2. Feminist Hareket Veri Seti’ni kullanılarak GKO, BOA ve HBO-BPBÇ
algoritmalarının 100 popülasyon 150 iterasyon parametlerindeki uygunluk
degeri disimi .................................................................................................111
Şekil 7.3. Feminist Hareket Veri Seti’ni kullanılarak GKO, BOA ve HBO-BPBÇ
algoritmalarının 100 popülasyon ve 200 iterasyon parametrelerindeki uygunluk
degeri disimi .................................................................................................112
Şekil 7.4. Feminist Hareket Veri Seti’ni kullanılarak GKO, BOA ve HBO-BPBÇ
algoritmalarının 200 popülasyon 100 iterasyon parametrelerindeki uygunluk
degeri disimi .................................................................................................114
Şekil 7.5. Feminist Hareket Veri Seti’ni kullanılarak GKO, BOA ve HBO-BPBÇ
algoritmalarının 200 popülasyon ve 150 iterasyon parametrelerindeki uygunluk
degeri disimi .................................................................................................116
Şekil 7.6. Feminist Hareket Veri Seti’ni kullanılarak GKO, BOA ve HBO-BPBÇ
algoritmalarının 200 popülasyon 200 iterasyon parametrelerindeki uygunluk
degeri disimi .................................................................................................118
Şekil 7.7. Ateizm Veri Seti’ni kullanılarak GKO, BOA ve HBO-BPBÇ algoritmalarının
100 popülasyon 100 iterasyon parametrelerindeki uygunluk değer değişimi.127
Şekil 7.8. Ateizm Veri Seti’ni kullanılarak GKO, BOA ve HBO-BPBÇ algoritmalarının
100 popülasyon 150 iterasyon parametrelerindeki uygunluk değer değişimi
Sekil 7.9. Ateizm Veri Seti'ni kullanılarak GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 200 iterasyon parametrelerindeki uygunluk değeri değişimi

Sekil 7.10. Ateizm Veri Seti'ni kullanılarak GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 100 iterasyon parametrelerindeki uygunluk değeri değişimi

Sekil 7.11. Ateizm Veri Seti'ni kullanılarak GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 150 iterasyon parametrelerindeki uygunluk değeri değişimi

Sekil 7.12. Ateizm Veri Seti'ni kullanılarak GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 200 iterasyon parametrelerindeki uygunluk değeri değişimi

Sekil 7.13. Kürtajın Yasallaşması Veri Seti'ni kullanılarak GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 100 iterasyon parametrelerindeki uygunluk değeri değişimi

Sekil 7.14. Kürtajın Yasallaşması Veri Seti'ni kullanılarak GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 150 iterasyon parametrelerindeki uygunluk değeri değişimi

Sekil 7.15. Kürtajın Yasallaşması Veri Seti'ni kullanılarak GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 200 iterasyon parametrelerinde uygunluk değeri değişimi

Sekil 7.16. Kürtajın Yasallaşması Veri Seti'ni kullanılarak GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 100 iterasyon parametrelerinde uygunluk değeri değişimi

Sekil 7.17. Kürtajın Yasallaşması Veri Seti'ni kullanılarak GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 150 iterasyon parametrelerinde uygunluk değeri değişimi

Sekil 7.18. Kürtajın Yasallaşması Veri Seti'ni kullanılarak GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 200 iterasyon parametrelerinde uygunluk değeri değişimi

Sekil 7.19. Hillary Clinton Veri Seti'ni kullanılarak GKO, BOA ve HBO-BPBÇ
algoritmalarının 100 popülasyon 100 iterasyon parametrelerinde uygunluk değeri değişimi .............................................................. 155

Şekil 7.20. Hillary Clinton Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 150 iterasyon parametrelerinde uygunluk değeri değişimi .............................................................. 157

Şekil 7.21. Hillary Clinton Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 200 iterasyon parametrelerinde uygunluk değeri değişimi .............................................................. 159

Şekil 7.22. Hillary Clinton Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 100 iterasyon parametrelerinde uygunluk değeri değişimi .............................................................. 161

Şekil 7.23. Hillary Clinton Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 150 iterasyon parametrelerinde uygunluk değeri değişimi .............................................................. 163

Şekil 7.24. Hillary Clinton Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 200 iterasyon parametrelerinde uygunluk değeri değişimi .............................................................. 165

Şekil 7.25. Feminist Hareket Veri Seti 100 Popülasyon 200 İterasyon Ortalama Uygunluk Değerlerinin Ortalama Grafiği .............................................................. 166

Şekil 7.26. Ateizm Veri Seti 100 Popülasyon 200 İterasyon Ortalama Uygunluk Değerlerinin Ortalama Grafiği .............................................................. 168

Şekil 7.27. Kürtajın Yasallaşması Veri Seti 100 Popülasyon 200 İterasyon Ortalama Uygunluk Değerlerinin Ortalama Grafiği .............................................................. 169

Şekil 7.28. Hillary Clinton Veri Seti 100 Popülasyon 200 İterasyon Ortalama Uygunluk Değerlerinin Ortalama Grafiği .............................................................. 170

Şekil 7.29. Feminist Hareket Veri Seti 200 Popülasyon 200 İterasyon Ortalama Uygunluk Değerlerinin Ortalama Grafiği .............................................................. 170

Şekil 7.30. Ateizm Veri Seti 200 Popülasyon 200 İterasyon Ortalama Uygunluk Değerlerinin Ortalama Grafiği .............................................................. 171

Şekil 7.31. Kürtajın Yasallaşması Veri Seti 200 Popülasyon 200 İterasyon Ortalama Uygunluk Değerlerinin Ortalama Grafiği .............................................................. 172

Şekil 7.32. Hillary Clinton Veri Seti 200 Popülasyon 200 İterasyon Ortalama Uygunluk
Değerlerinin Ortalaması Grafiği

172
### TABLOLAR LİSTESİ

<table>
<thead>
<tr>
<th>Sayfa No</th>
<th>Tablo No</th>
<th>Başlık</th>
<th>Sayfa Numaraları</th>
</tr>
</thead>
<tbody>
<tr>
<td>17</td>
<td>Tablo 3.1</td>
<td>ÇSA’da Topluluk Tespiti ile ilgili yapılan diğer çalışmalar</td>
<td>17</td>
</tr>
<tr>
<td>19</td>
<td>Tablo 3.2</td>
<td>ÇSA’da İlgı Madenciliği ile ilgili yapılan diğer çalışmalar</td>
<td>19</td>
</tr>
<tr>
<td>21</td>
<td>Tablo 3.3</td>
<td>ÇSA’da Tutum Tespiti ile ilgili yapılan diğer çalışmalar</td>
<td>21</td>
</tr>
<tr>
<td>25</td>
<td>Tablo 3.4</td>
<td>ÇSA’da Gizlilik Hakkını Koruyan Yakınlık Tespiti ile ilgili yapılan diğer çalışmalar</td>
<td>25</td>
</tr>
<tr>
<td>31</td>
<td>Tablo 3.5</td>
<td>ÇSA’da Siber Zorbalık Tespiti ile ilgili yapılan diğer çalışmalar</td>
<td>31</td>
</tr>
<tr>
<td>33</td>
<td>Tablo 3.6</td>
<td>ÇSA’da Spamlama Tespiti ile ilgili yapılan diğer çalışmalar</td>
<td>33</td>
</tr>
<tr>
<td>35</td>
<td>Tablo 3.7</td>
<td>ÇSA’da Zararlı Yazılım Tespiti ile ilgili yapılan diğer çalışmalar</td>
<td>35</td>
</tr>
<tr>
<td>39</td>
<td>Tablo 3.8</td>
<td>ÇSA’da Kimlik Avı Tespiti ile ilgili yapılan diğer çalışmalar</td>
<td>39</td>
</tr>
<tr>
<td>41</td>
<td>Tablo 3.9</td>
<td>ÇSA’da Profil Klonlama Tespiti ile ilgili yapılan diğer çalışmalar</td>
<td>41</td>
</tr>
<tr>
<td>44</td>
<td>Tablo 3.10</td>
<td>ÇSA’da Söylenti Tespiti ile ilgili yapılan diğer çalışmalar</td>
<td>44</td>
</tr>
<tr>
<td>47</td>
<td>Tablo 3.11</td>
<td>ÇSA’da İroni ve Kinaye Tespiti ile ilgili yapılan diğer çalışmalar</td>
<td>47</td>
</tr>
<tr>
<td>50</td>
<td>Tablo 3.12</td>
<td>ÇSA’da Fikir Lideri Tespiti ile ilgili yapılan diğer çalışmalar</td>
<td>50</td>
</tr>
<tr>
<td>53</td>
<td>Tablo 3.13</td>
<td>ÇSA’da Konu ve Olay Tespiti ile ilgili yapılan diğer çalışmalar</td>
<td>53</td>
</tr>
<tr>
<td>56</td>
<td>Tablo 3.14</td>
<td>ÇSA’da Tartışmalı Konuların Tespiti ile ilgili yapılan diğer çalışmalar</td>
<td>56</td>
</tr>
<tr>
<td>59</td>
<td>Tablo 3.15</td>
<td>ÇSA’da Nedensellik Tespiti ile ilgili yapılan diğer çalışmalar</td>
<td>59</td>
</tr>
<tr>
<td>106</td>
<td>Tablo 6.1</td>
<td>Manuel olarak belirlenen, tutumu belirli ve tutumu belirsiz hashtag örnekleri</td>
<td>106</td>
</tr>
<tr>
<td>108</td>
<td>Tablo 7.1</td>
<td>Kullanılan sınıflandırma ölçütleri</td>
<td>108</td>
</tr>
<tr>
<td>109</td>
<td>Tablo 7.2</td>
<td>Weka programı kullanılarak Feminist hareket veri setinden elde edilen sınıflandırma sonuçları</td>
<td>109</td>
</tr>
<tr>
<td>107</td>
<td>Tablo 7.3</td>
<td>Feminizm veri seti için kullanılan katsayılar</td>
<td>107</td>
</tr>
<tr>
<td>111</td>
<td>Tablo 7.4</td>
<td>GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 100 iterasyon parametreleri ile feminist hareket veri setine uygulanması sonucunda elde edilen başarı sırası</td>
<td>111</td>
</tr>
<tr>
<td>110</td>
<td>Tablo 7.5</td>
<td>GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 150 iterasyon parametreleri ile feminist hareket veri setine uygulanması sonucunda elde edilen başarı sırası</td>
<td>110</td>
</tr>
<tr>
<td>111</td>
<td>Tablo 7.6</td>
<td>GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 200 iterasyon parametreleri ile feminist hareket veri setine uygulanması sonucunda elde edilen başarı sırası</td>
<td>111</td>
</tr>
</tbody>
</table>
parametreleri ile feminizm veri setine uygulanması sonucunda elde edilen başarı sırası. 

Tablo 7.7. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 100 iterasyon parametreleri ile feminist hareket veri setine uygulanması sonucunda elde edilen başarı sırası. 

Tablo 7.8. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 150 iterasyon parametreleri ile feminist hareket veri setine uygulanması sonucunda elde edilen başarı sırası. 

Tablo 7.9. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 200 iterasyon parametreleri ile feminist hareket veri setine uygulanması sonucunda elde edilen başarı sırası. 

Tablo 7.10. BOA algoritması için $p$ parametrinin 0.25 alınması ile elde edilen başarı sırası. 

Tablo 7.11. BOA algoritması için $p$ parametrinin 1 alınması ile elde edilen başarı sırası. 

Tablo 7.12. GKO algoritması için $t$ parametrinin 2 alınması ile elde edilen başarı sırası. 

Tablo 7.13. GKO algoritması için $t$ parametrinin 4 alınması ile elde edilen başarı sırası. 


Tablo 7.15. Ateizm veri seti için kullanılan katsayılar. 

Tablo 7.16. Weka programı kullanılarak Ateizm veri setinden elde edilen sınıflandırma sonuçları. 

Tablo 7.17. GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 100 iterasyon parametreleri ile ateizm veri setine uygulanması sonucunda elde edilen başarı sırası. 

Tablo 7.18. GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 150 iterasyon parametreleri ile ateizm veri setine uygulanması sonucunda elde edilen başarı sırası. 

Tablo 7.19. GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 200 iterasyon parametreleri ile ateizm veri setine uygulanması sonucunda elde edilen başarı sırası.
Tablo 7.20. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 100 iterasyon parametreleri ile ateizm veri setine uygulanması sonucunda elde edilen başarı sırası .................................................................131
Tablo 7.21. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 150 iterasyon parametreleri ile ateizm veri setine uygulanması sonucunda elde edilen başarı sırası .................................................................133
Tablo 7.22. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 200 iterasyon parametreleri ile ateizm veri setine uygulanması sonucunda elde edilen başarı sırası .........................................................................................135
Tablo 7.23. BOA ve HBO-BPBÇ algoritmaların ateizm veri setinde uygunluk fonksiyonlarının eşit sayıda çağrılması sonuçları.........................................................138
Tablo 7.24. Kürtajın yasallaşması veri seti için kullanılan katsayılar ..................................................139
Tablo 7.25. Weka programı kullanılarak kürtajın yasallaşması veri setinden elde edilen sınıflandırma sonuçları..................................................................................................................139
Tablo 7.26. GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 100 iterasyon parametreleri ile kürtajın yasallaşması veri setine uygulanması sonucunda elde edilen başarı sırası .........................................................................................141
Tablo 7.27. GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 150 iterasyon parametreleri ile kürtaj veri setine uygulanması sonucunda elde edilen başarı sırası .........................................................................................143
Tablo 7.28. GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 200 iterasyon parametreleri ile kürtajın yasallaşması veri setine uygulanması sonucunda elde edilen başarı sırası .........................................................................................145
Tablo 7.29. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 100 iterasyon parametreleri ile kürtaj veri setine uygulanması sonucunda elde edilen başarı sırası .........................................................................................147
Tablo 7.30. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 150 iterasyon parametreleri ile kürtaj veri setine uygulanması sonucunda elde edilen başarı sırası .........................................................................................149
Tablo 7.31. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 200 iterasyon parametreleri ile kürtajın yasallaşması veri setine uygulanması sonucunda
elde edilen başarı sırası. .................................................................151
Tablo 7.32. BOA ve HBO-BPBÇ algoritmalarının kurtajın yasallaşması veri setinde uygunluk fonksiyonlarının eşit sayıda çağrılması sonuçları .........................152
Tablo 7.33. Hillary Clinton veri seti için kullanılan katsayılar .................................154
Tablo 7.34. Weka programı kullanılarak Hillary Clinton veri setinden elde edilen sınıflandırma sonuçları.................................................................154
Tablo 7.35. GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 100 iterasyon parametreleri ile Hillary Clinton veri setine uygulanması sonucunda elde edilen başarı sırası .................................................................156
Tablo 7.36. GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 150 iterasyon parametreleri ile Hillary Clinton veri setine uygulanması sonucunda elde edilen başarı sırası .................................................................157
Tablo 7.37. GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 200 iterasyon parametreleri ile Hillary Clinton veri setine uygulanması sonucunda elde edilen başarı sırası .................................................................159
Tablo 7.38. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 100 iterasyon parametreleri ile Hillary Clinton veri setine uygulanması sonucunda elde edilen başarı sırası .................................................................161
Tablo 7.39. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 150 iterasyon parametreleri ile Hillary Clinton veri setine uygulanması sonucunda elde edilen başarı sırası .................................................................163
Tablo 7.40. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 200 iterasyon parametreleri ile Hillary Clinton veri setine uygulanması sonucunda elde edilen başarı sırası .................................................................165
Tablo 7.41. BOA ve HBO-BPBÇ algoritmaların Hillary Clinton veri setinde uygunluk fonksiyonlarının eşit sayıda çağrılması sonuçları .................................166
<table>
<thead>
<tr>
<th>KISAŁTMALAR</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ÇSA</td>
<td>Çevrimiçi Sosyal Ağlar</td>
</tr>
<tr>
<td>SAA</td>
<td>Sosyal Ağ Analizi</td>
</tr>
<tr>
<td>DVM</td>
<td>Destek Vektör Makineler</td>
</tr>
<tr>
<td>NER</td>
<td>Adlandırılmış Varlık Tanma</td>
</tr>
<tr>
<td>HTA</td>
<td>Hızlı Topluluk Adaptasyonu</td>
</tr>
<tr>
<td>ART</td>
<td>Uyarlamalı Rezonans Teorisi</td>
</tr>
<tr>
<td>LDA</td>
<td>Latent Dirichlet Allocation (Gizli Dirichlet Tahsisı)</td>
</tr>
<tr>
<td>BTO</td>
<td>Biyocoğrafya Tabanlı Optimizasyon</td>
</tr>
<tr>
<td>MSN</td>
<td>Mobil Sosyal Ağlar</td>
</tr>
<tr>
<td>SAS</td>
<td>Sosyal Ağ Servisi</td>
</tr>
<tr>
<td>YTS</td>
<td>Yerel Tabanlı Sistem</td>
</tr>
<tr>
<td>KSSAS</td>
<td>Konumu Saklayan Sosyal Ağ Sunucusu</td>
</tr>
<tr>
<td>HK</td>
<td>Hücresel Kule</td>
</tr>
<tr>
<td>AGRQ-P</td>
<td>Geometric Range Query for Polygons (Çokgenler için Geometrik Arağ Sorgusu)</td>
</tr>
<tr>
<td>AGRQ-C</td>
<td>Geometric Range Query for Circles (Daireler için Geometrik Arağ Sorgulaması)</td>
</tr>
<tr>
<td>OBM</td>
<td>Ortalama Boşluk Merkezi</td>
</tr>
<tr>
<td>SFGMM</td>
<td>Sonsuz Faktörlü Gizli Markov Modeli</td>
</tr>
<tr>
<td>SAH</td>
<td>Sosyal Ağ Hizmetleri</td>
</tr>
<tr>
<td>İPS</td>
<td>İlan Panosu Sistemleri</td>
</tr>
<tr>
<td>KYK</td>
<td>K-en Yakın Komşu</td>
</tr>
<tr>
<td>NB</td>
<td>Naïve Bayes</td>
</tr>
<tr>
<td>DARPA</td>
<td>US Department of Defense Advanced Research Projects Agency (ABD Savunma Bakanlığı İleri Araştırma Projeleri Ajansı)</td>
</tr>
<tr>
<td>KTİ</td>
<td>Konu Algılama ve İzleme</td>
</tr>
<tr>
<td>GLKM</td>
<td>Global ve Lokal Konu Modeli</td>
</tr>
<tr>
<td>Symbol</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>$\vec{a}$</td>
<td>İkiden sıfıra doğru olarak azaldığı yer</td>
</tr>
<tr>
<td>$\vec{A}$</td>
<td>Katsayı vektörü</td>
</tr>
<tr>
<td>$\vec{C}$</td>
<td>Katsayı vektörü</td>
</tr>
<tr>
<td>$D$</td>
<td>Boyut</td>
</tr>
<tr>
<td>$L$</td>
<td>$[-1,1]$'de rastgele bir sayı</td>
</tr>
<tr>
<td>$\vec{r}$</td>
<td>Rastgele bir vektör</td>
</tr>
<tr>
<td>$p$</td>
<td>$[0,1]$'de rastgele bir sayı</td>
</tr>
<tr>
<td>$t$</td>
<td>Mevcut yineleme</td>
</tr>
<tr>
<td>$x_{rand}$</td>
<td>Mevcut popülasyondan seçilen rastgele bir konum vektörü</td>
</tr>
<tr>
<td>$X^*$</td>
<td>Elde edilen en iyi çözümün pozisyon vektörü</td>
</tr>
</tbody>
</table>
1. **GİRİŞ**


![Şekil 1.1. Basit bir sosyal ağ diyagramı](image-url)
Fakat biz burada bu tanımdan öte teknoloji ile gündemimize girmiş çevrimiçi sosyal siteler ile yeni sosyal ağ kavramını kullanacağız. Bu sosyal ağlar bir tür sosyal medyadır, sosyal paylaşım sitelerini içinde barındırır ve internet kullanımcılarının insanlarla bağlantı kurmalarına ve bilgi oluşturmalarına olanak tanır (Yazdanifard ve Yee, 2014). ÇSA sitelerinin toplu olarak benimsenmesinin ortaya çıkması, insanların bilgiyi nasıl ilettilerini ve paylaştıklarını, işletmelerin nasıl faaliyet gösterdiğini ve rekabet ettiğini ve politikalara için nasıl yarıştığını ve etkilediğini değiştirmiştir. Bu sosyal ağlar sayesinde, her hangi bir hastalıkın yayılmasından ve önemli siyasi görüşlerden yola çıkarak yeni suç odaklarının kim olacağı gibi gizli bilgilerin ortaya çıkarılması analiz edilebilir hale gelmiştir (Sapountzi ve Psannis, 2016).

ÇSA verilerinin geniş kullanımı ile beraber çeşitli uygulamalar giderek artan bir oranda ortaya çıkılmış ve ÇSA’lı büyük miktarlarda bir görüş ifade eden büyük miktarlı metin ve multimedya içeriği üretmektedir. Sosyal medya analizinde ÇSA sosyal medya kanallarından elde edilen yapısal ve yapısalsal olmayan verilerin analiz edilip çeşitli sonuçlar çıkarılması işine verilen addır. Sosyal medya, Sosyal ağlar (Facebook, LinkedIn), Bloglar (BlogSpot, Wordpress), Microbloglar (Twitter, Tumblr), Sosyal haberler (Digg, Reddit), Sosyal bookmarklar (Delicious, StumbleUpon), Medya paylaşımı (Instagram, YouTube), Wiki (Wikipedia, Wikihow), Soru cevap siteleri (Yahoo! Answers, Ask.com), ve Yorum siteleri (Yelp, TripAdvisor) olarak kategorize edilebilir (Özköse vd., 2015; Li vd., 2015). Bu sitelerin birçoğu doğrudan ya da en direkt olarak bir sosyal ağ ifade etmektedir.

Bu ağların analizi Sosyal Ağ Analizi (SAA)’nın alanı girmiştir. SAA bireyler, nesneler ya da kurumlar arasındaki ilişkilerin anlamlı sayısal verilere çevrilerek ortaya konuşmasını sağlayan bir yöntemdir. Antropoloji, eğitim, sosyoloji ve ekonomi gibi farklı birçok alan kendi kendine uygulama alanı bulan SAA ile toplulukların yapısı incelenmekte ve topluluklar arasındaki gizli kalan ilişkiler bulunarak bunların arasındaki bağlantıları ortaya çıkarmaktadır (Argan, 2014). SAA araştırmalarında topluluk içindeki bağlantı ve ilişkiler kullanılarak çeşitli problemlere çözümler bulunmaya çalışılmaktadır. SAA yöntemi ve uygulama alanı bakımdan oldukça zengin bir çerçeve sahiptir. Literatürde topluluk keşfi, anomali keşfi, konu keşfi, olay keşfi ve tutum keşfi gibi onlarca sosyal ağ analizi problemi mevcuttur. Bu problemlerin bir kısmı nispeten eski olsa da birçoğu yeni ve popülerdir. Tüm bu problemler ÇSA’ın yaygınlaşması ile daha önce düşünülemediyce kadar geniş alanlarda uygulanma alanı bulunmuştur. Günümüzde hayata her alanında kullanılan ve önumsüzdeki yüz yılda ise muhtemelen temel bir yaşam paradigması haline gelecek olan çevrimiçi iletişim
platformları, bu platformlardaki ilişki biçimleri ve bilgi paylaşımA şekilleri insanlığın olmazsa olmazı haline gelecektir. Bu bakımdan bilişim alanındaki imkânları kullanarak SAA yöntemlerini kullanmak, yeni veriler üzerinden yeni bilgilere erişmek üretim ve fikir dünyamızın gelişimi için vazgeçilemez olacaktır.

1.1. Literatür Taraması


Murakami ve Raymond (2010), yaptıkları çalışmada çevrimiçi tartışmalarda kullanıcılın genel konumlarını belirleme, yanı tartışmanın içerisindeki açıklamalarında
yerel bilgileri kullanarak çevrimiçi bir tartışmaın ana konusunu destekleme veya karşı koyma görevi için bir yöntem önermişlerdir. Çevrimiçi tartışma, her kullanıcının belirli bir konu hakkında bir görüş bildirdiği, diğer kullanıcıların da görüşlerini tartışmaya dâhil ederek konumlarını belirtilkleri bir forumdur. Destekleyici veya muhalif ifadeler, doğrudan cevap verilerek ya da dolaylı olarak diğer görüşler (yerel anlaşmayı ya da anlaşmazlığı ifade etmek için) kastedilerek yapılır, bu durum kullanıcıların genel konumlarını tanımlamayı zorlaştırır. Önceden yapılmış başka bir çalışmada, açıklamaların içeriğini tamamen yok sayan bağlantı tabanlı bir yöntem, tanımlama ifadesi için yalnızca açıklamaların içeriğine dayalı yöntemlerden daha yüksek doğruluk sağlayabileceği göstermiştir. Bu çalışmada, açıklamaların metin içeriğinin bağlantiya dayalı yöntemle kullanılamasının, tanımlama görevinde daha yüksek doğruluk sağlayabileceği göstermiştir.

Başka bir çalışmada ise ideolojik tartışmalardaki tutumları sınıflandırmak için fikirlerin ve fikirlerin tartışılmasını faydasi araştırılmıştır. İdeolojik duruşta tartışılan fikirleri yakalamak için, el ile açıklanmış bir Korpus‘tan otomatik olarak tartışlan bir söz dizimi oluşturulmuştur. Duyguları kullanarak ve fikirleri ve hedefleri özellik olarak savunan denetimli sistemler kurulmuştur. Bu sistemler, dağıtım tabanlı bir temelden daha iyi performans göstermektedir. Ayrıca, her iki düşünme türü kullanılarak, performans tekstenedi bir sistemden daha iyi olmuştur (Somasundaran ve Wiebe, 2010).

Bir diğer çalışmada ise bir konuşmacının belirli bir konuya yönelik tutumu, bütünsel bir özen eğilimi tanımlamayı içeren bir görev belirleme zorlukları vurgulanmıştır. Oyunculukta ideolojidey kadar uzanan 14 konu için, ConvinceMe.net tartışma sitesinde 4731 yayın organı üzerinde tutum sınıflandırması incelemiştir. İdeolojik tartışmaların dürüstüçü yayınların daha büyük bir paya sahip olduğunu ve dürüstüçü gönderilerin hem insanlar hem de eğitimi sınıflandırıcılar için tutum sınıflandırılması çok daha zor olduğunu göstermiştir. Ayrıca, özen ifadelerin sayısının, tartışmalara göre değişen sistemlerin performansıyla ilişkili bir olgu olmuştur. %47 ile %66 arasında değişen unigram taban çizgileri ile karşılaştırıldığında, konu başlığına dayanan tutumun %60 ila %75 aralığında sınıflandırılması sonuçları sunulmuştur. Sonuçlar bu tür mesajların diyalog bağlamını dikkate alan özelliklerin ve yöntemlerin doğruluğunu artırdığını göstermektedir (Walker vd., 2010).

Bir çevrimiçi tartışma forumunda iki taraflı bir tartışma için yazılan bir yazıla ifade edilen tutumun belirlenmesi, görüş (düşünce) madenciliğinde nispeten yeni ve zorlu bir sorundur. Hasan ve Ng (2013) özellikle öğrenme temelli bir tutum sınıflandırma sisteminin
performansının eğitim verilerinin miktari ve kalitesi, alçta yatan model, özellik kümesinin zenginliği ve dil dışi kısıtlamaların uygulanması ile nasıl değiştiği incelenerek, ideolojik tartışmaların görüş sınıflandırmasına yönelik makine öğrenimi yaklaşımlarının nasıl geliştirileceği üzerinde çalışmışlardır.


Ebrahimi vd., (2016) tweetlerdeki tutum sınıflandırmasını, tutumun amacı, tutumun hedefini ve tweet’in ortaklaşa düşüncesini modelleyen olasılıksal bir yaklaşım önermişlerdir. Duygu veya hedef değişkenlerini özellik uzayına ekstra değişkenler olarak birleştirmek yerine, duygudurum-giriş değişkenleri ve hedef-durum-girdi değişkenleri arasında üç yönlü etkileşimleri birleştirmek için özgün bir formülasyon kullanmıştır. Önerilen tanımlama
sezgisel olarak, durum sınıflandırması için hedef özelliklerden duygusal özelliklerini ayırt etmeyi amaçlamaktadır. Ek olarak, tüm hedefleri ele alan tek bir durum sınıflandırıcısının düzenlenmesi, yumuşak bir ağırlık paylaşımı görevi görevi görür. Bu modelin ayırt edici eğitiminin, denetimli duruş sınıflandırmasında son teknoloji sonuçları sağladığı ve üretken eğitiminin zayıf denetlenen ortamda rekabetçi sonuçlar elde ettiği göstermiştir.


Adlandırılmış varlık tanıma (NER), onlarca yıldır üzerinde çalışılmış, iyi bilinen bir bilgi çıkarma görevidir. Son zamanlarda, NER deneyseliler sosyal medya metinlerine üzerine rapor eden çalışmalar ortaya çıkmıştır. Öte yandan, tutum algılama genellikle duygusal analizi kapsamında ele alınan oldukça yeni bir araştırma konusudur. Tutum tespit çalışmalarını çoğulukla, metin sahibinin metinde açıkça veya örtülü olarak belirtildi belirli bir hedefe yönelik tutumunun araştırıldığı çevrimiçi metinlerine uygulanır. Yapılan bu çalışmada, tweet’lerde adlandırma teşebbüsü durum tespit görevine olası kastını araştırılmıştır. NER deneyselilerin değerlendirmeye sonuçlarının yanı sıra, adlandırılmış varlıklar kullanarak takip eden tutum algılama deneyseleri, tweetlerin halka açık bir tutum açıklamalı veri seti üzerinde rapor edilmiştir. Elde edilen sonuçlar, yüksek performanslı bir NER sistemi ile elde edilen adlandırılmış varlıkların tweet’lerde tutum algılama performansına katkıda bulunabileceğini göstermektedir (Küçük 2017).

tespiti alt görevinde ilk sıraya ve durum algılama alt görevinde dördüncü sıraya yerleşilmiştir.


Küçük ve Can (2018), sporla ilgili tweet'lerde tutum algılamayı hedeflemişlerdir. DVM tabanlı tutum sınıflandırmalarının bu tür tweetler üzerindeki performans sonuçları sunulmuştur. Öncelikle, her bir araştırma amaçlı olarak kamuya açık hale getirilen, özel tweet veri setinin tutum bilgilerine açık olan üç sürümü verilmiştir. Daha sonra, bu veri setinde tweet tespiti için farklı özellik setleri kullanılarak DVM sınıflandırıcıları değerlendirilmiştir. Kullanılan özellikler, unigramlar, bigramlar, hashtag'ler, dış bağlantılar, ifadeler ve son olarak adlandırılmış varlıklar bazlıdır. Sonuçlar, DVM sınıflandırıcıları ile unigram, hashtag ve adlandırılmış varlıklar temel olan özelliklerin ortak kullanımının, sporla ilgili tweet'lerde tutum algılama problemi için makul bir yaklaşım olduğunu göstermektedir.

1.2. Tezin Amacı

Bu tez çalışmasında öncelikle sosyal ağların yeni yüzyılda bilgisayar teknolojisi ile vücut bulmuş halı olan ÇSA’lar ve bu ağların bu durumunda yardımcı olabilecek ve bu ağların düğümleri arasındaki ilişkileri analiz ederek işe yarar sonuçlar elde etmemizi sağlayan sosyal ağ analizini anlatılmaya çalışılmıştır. Daha sonra bu tez çalışmalarının temel konularından biri olan ÇSA’lara ele alınmıştır. Ayrıca çalışmadan literatürden çok adaledeki çalışmaların farklı olarak 21 adet sosyal ağ analizi problemi ÇSA’lar özelinde anlatılırlar, bu problemlerin çözümünde kullanlan yöntemler belli oranda anlatılmış, geniş bir literatür taraması yapılmış ve böylece zengin bir kaynak oluşturulması hedeflenmiştir. Bu açıdan özgün bir çalışma amaçlanmıştır. Yapılan çalışmanın sonuçu olarak sosyal ağ analizi problemi konusunda yapılan çalışmalar hakkında daha derin bir analiz imkanı amaçlanmıştır. Literatürde yapılan

7
çalışmalarda kümeleme, sınıflandırma ve birliktelik kuralları gibi birçok veri madenciliği yöntemi kullanılmıştır. Bu çalışmada ise popüler ve daha çok yeni olan bir ÇSA problemi olan “Tutum Tespiti” problemi ilk defa bir optimizasyon problemi olarak ele alınmış ve bu yöntem kullanılarak daha iyi sınıflandırma sonuçları elde edilmeye çalışılmıştır. Ayrıca bu tez kapsamında yeni bir hibrit optimizasyon algoritması önerilmiştir.

Bu tezin literatürde katkıları genel olarak şu şekilde özetlenebilir:

- Sosyal ağlar ve sosyal ağ analizi ile ilgili temel bilgiler sunulmuştur.
- Bu çalışmada literatürde ilk defa yirmiden fazla çevrimiçi sosyal ağ analizi problemi derlenmiş ve ÇSA problemleri kavramı ortaya konularak yeni bir kategorilendirme yapılmış ve ÇSA problemleri özelinde analiz edilmiştir.
- Optimizasyon, sezgisel yöntemler ve büyük veri hakkında temel bilgiler sunulmuştur.
- Bu çalışmada sosyal ağ analizi problemlerinin sınıflandırılması kullanılan optimizasyon yöntemi kullanılmıştır.
- Bu tez kapsamında ayrıca yeni bir hibrit optimizasyon algoritması önerilmiştir.
- Tez kapsamında geliştirilen yöntem daha çok yeni olması rağmen umut verici sonuçlar elde edilmiştir.

1.3. Tezin Yapısı

Bu tez çalışması sekiz temel bölümden oluşmaktadır. Birinci bölümde, sosyal ağlar ve çevrimiçi sosyal ağlar hakkında temel bilgiler verilmiş literatürde bulunan Tutum Tespiti hakkında yapılan çalışmalara değinilmiş ve tezin amacı açıklanmıştır.

İkinci bölümde, sosyal ağ analizi hakkında genel bilgiler verilmiştir.

Üçüncü bölümde, çevrimiçi sosyal ağ analizi problemleri detaylı bir şekilde anlatılarak bu problemler hakkında geniş bir kaynak verilmiştir.

Dördüncü bölümde, Beşinci bölümde, büyük veri ve büyük veri araçları hakkında bilgiler verilmiştir. Daha sonra ise optimizasyon hakkında bilgi verilmiştir.

Altıncı bölümde, deneySEL çalışmalarda kullanılan yöntem, veri ön işleme adımları ve sınıflandırma veri setleri hakkında ayrıntılı bilgi verilmiştir.

Yedinci bölümde, önerilen yöntemin sınıflandırma veri setlerinden elde edilen karşılaştırmalı test sonuçları verilmiştir.
Sekizinci bölümde, tezde yapılan çalışmalar ve elde edilen sonuçlar değerlendirilmiş, ileriye dönük çalışmalar için önerilerde bulunulmuştur.
2. SOSYAL AĞ ANALİZİ

2.1. Sosyal Ağ Analizi Nedir


2018 Haziran ayı itibari ile en popüler sosyal ağ platformlarının kullanıcı sayıları Şekil 2.2’deki grafik ile verilmiştir. Bu grafikteki veriler çevrimiçi sosyal ağların kapsamının ne kadar geniş olduğunu çarpmış bir şekilde ortaya koymaktadır. Bu alanda piyasının lideri konumunda bulunan Facebook 2.1 milyar aylık aktif kullanıcı sayışı ile lider konumundadır. Altıncı sıradaki fotoğraf paylaşım sitesi olan Instagram ise 1 milyardan fazla aktif kullanıcısı sayısına erişmiştir. Ayrıca bir mesajlaşma servisi olan QQ ise 806 milyon aylık aktif kullanıcısı sayısına erişmiştir.

Şekil 2.1. Zachary Karate Kulübü ağı (Zarchary, 1977)
Şekil 2.2. Haziran 2018 sosyal ağı sitelerinin aktif kullanıcı sayıları (URL-1, 2018)

Çevrimiçi sosyal ağların çözümlemesi bize insan davranışlarının çok yönlü bilgisini verebilir (Bendel ve Henryk, 2008). Ağdaki aktörler arasındaki olası bağlantı tahmini, kullanıcıların davranışlarından örüntü çıkarımı, topluluk keşfi sosyal ağ analizi sosyal ağ analiz edilirken kullanılan yöntemlerden bazılarıdır. Bir sosyal ağda birbirleri ile etkileşen varlıkların sosyal ağ yapısı içerisinde gerçekleştirdikleri her türlü etkileşimin detaylı olarak incelenmesi ve analiz edilmesi sonucunda söz konusu yapıyla ve içerdeği düğümlerle ilgili olarak ilk bakısta fark edilemeyen çok kıymetli bilgiler elde edilebilmektedir.
3. ÇEVİRİMİÇİ SOSYAL AĞ ANALİZİ PROBLEMLERİ

Hotmail, Yahoo gibi e-mail servisleri ile Facebook, Twitter, YouTube gibi sosyal ağ servislerinden oluşan çevrimiçi sosyal ağlar düğümlerden ve bu düğümlerin birbirleri ile olan etkileşimlerini, etkilerini ve işbirliklerini gösteren bağlardan oluşan karmaşık yapılar bulunmaktadır. Bu yapılar bilginin paylaşılamasını ve yayılmasını görevini görmektedir (Kröl 2014). Bugün bu yapıların geldikleri nokta bakımından sosyal ağ analizi problemlerinin ilk uygulanma alanlarına göre nicel açıdan büyük farklılıklar göstermektedir. Çok önceleri hatta bu yüzyılın başlarına kadar kısıtlı sayıda üyeden oluşan çeşitli sosyal ağ gruplarına uygulanan bu problemler şimdi milyonlarca insandan oluşan sosyal ağlara uygulanabilme olanağı yakalanmıştır. Bu olanak beraberinde onlarca sosyal ağ analizi yöntemimini bu ahlarda uygulama imkanını getirmiştir.


![Şekil 3.1. Sosyal ağ analizi problemleri](image-url)
3.1. Topluluk Keşfi


Neden Topluluk keşfi?

- Bir alıntı ağındaki topluluklar bir konu ile alakalı makaleleri gosterebilirler.
- İlişkili konular ile ilgili genel ağ sayfalarını gosterebilirler.
- Bütün bir aği bize özetlemeye ve onu anlamamızı yardımcı olabilir (URL-2).


Nguyen vd., (2011) yaptıkları çalışmada dinamik ÇSA’ın topluluk yapısını tanımlamak ve işlemek için uyaranabilir modüller bazı bir yöntem olan Hızlı Topluluk


algoritmasının zaman karmaşıklığı konusunda yeni bir yol açması amacıyla tasarlanmıştır. Bir başka avantajı da yöntemden, küme sayısını belirtmeyi gerektirmemesidir.


olasılığını dinamik olarak ayarlamak için koatik bir mekanizma kullanılmıştır. Ek olarak, ağırlıksız ve ağırlıklı ağlarda mutasyon için iki yeni strateji önerilmiştir. Önerilen yöntemin nihai çıktısı bir dizi baskı olmayan (Pareto-optimal) çözüm olduğundan, bu baskı olmayan çözümler arasında en iyi uzaşımayı belirlemek için alpha_SAM adı verilen bir ölçüm önerilmiştir. 14 gerçek hayat örneklerinden alınan veri seti üzerinde yapılan kapsamlı deneylere dayanan nicel değerlendirmeler, bu çalışma sunulan yöntemin, literatürdeki diğer benzer algoritmalarından oldukça üstün olan olumlu sonuçlara ulaştığını ortaya koymaktadır.

Topluluk Keşfi ile ilgili diğer çalışmaları Tablo 3.1’dede gösterilmiştir.

<table>
<thead>
<tr>
<th>YIL</th>
<th>ÖNEMLİ KAYNAKLAR</th>
</tr>
</thead>
</table>
| 2007 | • Karmaşık ağlarda topluluk tespiti için yeni bir evrimsel teknik önerilmiştir (Gog vd., 2007).
|      | • LDA (Latent Dirichlet Allocation=Gizli Dirichlet Paylaştırması) tabanlı hıyerarşik Bayes algoritmaları ile sağlanan SN-LDA (basit sosyal ağ LDA) olarak adlandırılan bir yöntem topluluk tespiti için önerilmiştir (Zhang vd., 2007). |
| 2008 | • Bağlantıların kullanıcı sayfalara arasındaki karşılık “arkadaşlık” olduğunu gösteren Facebook ağının topluluk yapısı araştırılmıştır (Traud vd., 2008). |
| 2009 | • Yerel topluluk yapısının yeni bir ölçümlü önerilmiştir (Chen vd., 2009). |
| 2010 | • Çakışan toplulukların tespiti için, sosyal medya kullanıcılar ve etiketler arasında ağ bilgilerini kullanılarak yeni bir ortak kümeleme çerçevesi önerilmiştir (Wang vd., 2010). |
|      | • Dinamik sosyal ağlarda topluluk algılama için yeni bir algoritma (ILCD) önerilmiştir (Cazabat vd., 2010). |
| 2011 | • YouTube içerikindeki toplulukları tespit etmek için yerel kümeleme dayalı çok aşamalı bir algoritma önerilmiştir (Gargi vd., 2011). |
| 2012 | • Sosyal medya ağlarda topluluk algılamanın etkinliği, kenar içeriğinin kullanılaması yöndede çalışılmıştır (Qi vd., 2012). |
|      | • Sosyal medya topluluk tespiti için bir anket sunulmuştur (Papadopoulos vd., 2012). |
| 2013 | • Topluluk saptaması için grafik kümeleme yöntemlerine dayanan yeni bir yaklaşım önerilmiştir (Giatsoglou vd., 2013). |
| 2014 | • Dinamik sosyal ağlarda topluluk algılaması için Hızlı Topluluk Adaptasyonu (Quick Community Adaptation, QCA) adlı yeni bir çerçeve önerilmiştir (Nguyen vd., 2014). |
| 2015 | • Sosyal medya toplulukları tespiti için bir koalisyon oluşturma oyunu teorisi tabanlı yöntem önerilmiştir (Zhou vd., 2015). |
| 2016 | • Sosyal ağlarda tespit tespit tespit etmeye yönelik çeşitli modülerlik temelli yaklaşımlar tartsılışmıştır (Devi and Poovamal 2016). |
| 2017 | • Karmaşık sosyal ağlarda yüksek kaliteli toplulukları tespit etme için yeni karma (paylaşımlı + dağıtılması belir) paralel algoritma önerilmiştir (Sharma and Oliveira 2017). |
| 2018 | • Sosyal ağlarda duygualan topluluk tespiti çalışılmıştır (Kanavos vd., 2018). |
|      | • Twitter’ın ağı için yeni bir topluluk algılama ve bilgi yayma yaklaşıımı önerilmiştir (Abdelsadek vd., 2018). |
|      | • Twitter’da tinspiration ve fitspiration çevrimiçi topluluklarının karşılaştırılması ele alınmıştır (Tiggemann vd., 2018). |
18

3.1.1. İlgi Madenciliği


İlgi Madenciliği ile ilgili diğer çalışmalar Tablo 3.2’de gösterilmiştir.

Tablo 3.2. ÇSA’ da İlgili Madenciliği ile ilgili yapılan diğer çalışmalar

<table>
<thead>
<tr>
<th>Yılı</th>
<th>Çalışma</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007</td>
<td>Bloglarda kullanıcıların ilgi alanlarını modellemeye yönelik yeni bir sistem sunulmuştur (Liu vd., 2007).</td>
</tr>
<tr>
<td>2008</td>
<td>Blogların ilgilerini moddleyen bir yaklaşım önerilmiştir (Cheng vd., 2008)</td>
</tr>
<tr>
<td>2009</td>
<td>Yenilikçi konulardaki tespiti için kullanıcı ilgisi ontolojisine dayanan yeni bir yöntem önerilmiştir (Nakatsuji vd., 2009)</td>
</tr>
<tr>
<td>2010</td>
<td>Belirli Twitter kullanıcılardan ilgisini çeken konular tespit edilmiştir (Michelson ve Macskassy, 2010). Twitter kullanıcısının ilgi alanlarını ve endişelerini etiketleme için kişiselleştirilmiş ek açıklama bilgilerini otomatik olarak oluşturma üzere tasarlanmış bir sistem tanıtılmıştır (Wu vd., 2010)</td>
</tr>
<tr>
<td>2011</td>
<td>Kullanıcılardan ilgi Alanlarını’nı, kişisel ontoloji tabanlı profilerde kullanıcı tarafından ifade edilen zevk ve tercihlerden otomatik olarak belirleyen bir sistem önerilmiştir (Cantador ve Castells, 2011)</td>
</tr>
<tr>
<td>2012</td>
<td>Kullanıcılardan ilgi alanlarını dikkate alarak eğilimleri ve konuları tespit etmek için istatistiksel bir model önerilmiştir (Tang ve Yang, 2012).</td>
</tr>
<tr>
<td>2013</td>
<td>Kullanıcının ilgi alanları için KAURI olarak bilinen grafik tabanlı bir çerçeve önerilmiştir (Shen vd., 2013). Twitter mesajlarından kullanıcı çıkarlarını elde etmek için yeni bir sistem önerilmiştir (Vu ve Perez, 2013)</td>
</tr>
<tr>
<td>2014</td>
<td>Mikrobloglar kullanıcılardan kullanıcı ilgisini bulmak için bir Kullanıcı-Konu modeli (User-Topic model, UTM) olarak adlandırılan yeni bir konu tespit modeli önerilmiştir (He vd., 2014). İlk sosyal önerisi olarak adlandırılan bir öneri sistemi modeli önerilmiştir (Li vd., 2014)</td>
</tr>
<tr>
<td>2015</td>
<td>İlgili Alanları için yeni bir yöntem geliştirilmiştir (Li vd., 2015). İlk sosyal önerisi olarak adlandırılan bir öneri sistemi modeli önerilmiştir (Li vd., 2014)</td>
</tr>
</tbody>
</table>

**Not:** **ÇSA’ da:** Çevrimiçi Sosyal Analiz. **UTM:** Üç boyutlu model. **SNA:** Sosyal Ağ Analizi.
Kullanıcı çıkarlarını tanımlamak için semantik bir zenginleştirme yöntemi önerilip sonuç olarak kullanıcı ilgilerinin zaman içinde değişmesini ve ilgi hiyerarşisini belirlemek amacı ile konu hiyerarşi ağacı modelinin kullanılması önerilmiştir (Jiang ve Sha, 2015). Kullanıcıların coğrafi mesafeleri gibi çeşitli sosyal özellikleri dikkate alınarak, bazı kullanıcıların ilgi alanları bilinmemesine rağmen diğer kullanıcılarla benzer ilgi alanlarının benzerliği araştırılmıştır (Han vd., 2015).

2016 • Metinsel-coğrafi-sosyal farklılık olasılığı matris faktörizasyonu yöntemi ilgi noktasi (POI point of interest) önerisi yöntemi sunulmuştur (Xingyi vd., 2016)
• Kullanıcının mevcut coğrafi konumlarına göre ilgi bölgelerini keşfedebilen Bölge Bülme Sorusunu adı verilmiştir (Shang vd., 2016).

2017 • İlgili alanların hiyerarşik bir temsiliyle desteklenen Twitter kullanıcılarının yüksek kapsama modellemesi için Twixonomy olarak adlandırılan yeni bir yöntem sunulmuştur (Faralli vd., 2017)
• Modelleden güven ve ilgiye dayalı, topluluk kümenlemesi için yeni bir yöntem önerilmiştir (Ullah ve Lee, 2017)

2018 • Sina Weibo’daki kullanıcı konularını tespit etmek için etiketlere ve iki yönlü etkileşimlere dayalı bir algoritma önerilmiştir (Deng vd., 2018)
• Kullanıcının ilgisi saptanır ve kullanıcıların çeşitli etiketlerle etiketlenmesi için yeni bir yöntem önerilmiştir (Li vd., 2018)

3.2. Tutum Tespiti

Tutum Tespiti problemi belirli bir hedef hakkında görüş belirten kişinin o hedefin yandaşı, karşıtı ya da hedef hakkında taarruz olmamış tespit etmeye yarayan sosyal ağ analizi problemdir. Hedef, bir kişi, herhangi bir organizasyon, bir devlet politikası ya da bir ürün olabilir. İnsanlar çeşitli hedeflere karşı olan tutumlarını çevrimiçi forumlar, Twitter, Youtube ve Instagram gibi sosyal ağlar aracılığı ile ifade edebilir (Sobhani vd., 2016). TutumTespiti fikir madenciliğinin bir alt dalı olarak ya da duygusal analiz olarak düşünülebilir. Duygu analizi ile Tutum Tespiti arasındaki temel fark, duygusal analizde bir metnin pozitif, negatif veya nötr olup olmadığını karar verirken, Tutum Tespiti’nde ise metni yazan kişinin ilgili hedef hakkında tutumunun yandaşlık, karşıtlık ya da nötr olup olmadığını araştırılır. Dahası bu tutum metinde açık bir şekilde belirtilmeyecektir. Hatta metinde bahsedilen konu hakkında metin olumlu görüş bildiriyor görünebilir ancak aynı zamanda yazarın bu konu hakkında karşıt düşüncesi olduğu sonucu çıkarılabilir. Bu açıdan Turuş Tespiti görevi zor bir görevdir (Krejzl ve Steinberger, 2016; Mohammad vd., 2016).

Mohammad vd., (2016) yaptıkları çalışmada ABD’de genel olarak bilinen “Atheism, Climate Change is a Real Concern, Feminist Movement, Hillary Clinton ve Legalization of Abortion” konu başlıklarını hakkında 2 milyon tweet toplayarak tutum tespiti çalışması yapmışlardır. Bu çalışmada tweetlerden her hedefe yönelik duruş, görüşün hedefi ve duygusu etiketleri ile eğitim ve test verileri elle hazırlanmıştır. Daha sonra net etmeli ve yarı-
denetimli olmak üzere iki ayrı çalışma yürütülmüştür. Denetimli çalışmada verilerin %70’i eğitim geri kalanı ise test verisi olarak kullanılmıştır. Yarı-denetimli çalışmada ise eğitim seti kullanılmamış sadece test verilerinden yararlanılmıştır. Yapılan çalışmalar sonucunda denetimli çalışma görevi için en yüksek F-Score 67.82 olarak elde edilmiştir. Yarı-denetimli görev için ise %56.28 F-score elde edilmiştir.


Tutum Tespiti ile ilgili diğer çalışmalar Tablo 3.3’de gösterilmiştir.

Tablo 3.3. ÇSA’ da Tutum Tespiti ile ilgili yapılan diğer çalışmalar

<table>
<thead>
<tr>
<th>Year</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007</td>
<td>Seçimlerin sonuçlarını tahmin etmek için seçimleri tartışan forumlar bir denetsel çerçeve analizi ile incelenmiştir (Kim ve Hovy, 2007).</td>
</tr>
<tr>
<td>2008</td>
<td>Çevrimiçi politik tartışmada kullanıcı sınıflandırması için çevrimiçi tartışmalar incelenmiştir (Malouf ve Mullen, 2008).</td>
</tr>
<tr>
<td>2009</td>
<td>Çevrimiçi tartışmada görüş sınıflandırması için datetimsiz görüş analizi yöntemi ileri sürülmüştür (Somasundaran ve Wiebe, 2009).</td>
</tr>
<tr>
<td>2010</td>
<td>Kullanıcıların yerel bilgilerini kullanarak çevrimiçi tartışmalarında kullanıcıların genel pozisyonlarını (destek veya karşı) sınıflandırmak için bir yöntem önermiştir (Murakami ve Raymond, 2010). İdeolojik tartışmalarındaki tutumları tesp etmek için görüş analizi kullanılmıştır (Somasundaran ve Wiebe, 2010).</td>
</tr>
<tr>
<td>2012</td>
<td>Çevrimiçi forum tartışmaları sonrası görüş sınıflandırması incenmiştir (Walker vd., 2012).</td>
</tr>
<tr>
<td>2013</td>
<td>Çevrimiçi forumlarda ideolojik tartışmalarda görüş sınıflandırılmıştır (Hasan ve Ng, 2013).</td>
</tr>
<tr>
<td>2014</td>
<td>Çevrimiçi tartışma sitelerinde görüş sınıflandırması için yeni bir sınıflandırma yöntemi sunulmuştur (Sridhar vd., 2014).</td>
</tr>
</tbody>
</table>
2015 • Görüş sınıflandırması için yeni bir olasılıksal modelleme çerçevesi önerilmiştir (Sridhar vd., 2015)

2016 • Tweet’lerde tweetin duruşunu, duruş hedefini ve tweetin tutumunu tasarlayan olasılıksal bir yaklaşım önerilmiştir (Ebrahimi vd., 2016).
    • Tweet’lerde duruş tespiti için yeni bir konvolüsyon nöral ağ yöntemi önerilmiştir (Wei vd., 2016)

2017 • Tweet’lerde tanımlı tüzel kişiliklerin tanıma algoritma görevine katkısı araştırılmıştır (Küçük, 2017)
    • Tweet’lerde farklı modeller ve tweet temsilcilerine dayalı çeşitli yaklaşımlar duruş ve cinsiyet tespiti için uygulanmıştır (González vd., 2017).
    • Duruş algılama için yeni bir metot tabanlı contextonyms ve contextssets kavramı önerilmiştir (Gadek vd., 2017)

2018 • Tweet’lerde duruş tespiti için DVM tabanlı bir yaklaşım önerilmiştir (Küçük ve Can, 2018)

3.3. Gizlilik Hakkını Koruyan Yanıllık Tespiti

kesinlik garantisi vermekti, yüksek iletişim maliyetleri çıkarka ve kullanıcı tercihleri açısından esnek bir sistem sağlamak için (Šikšnys vd., 2010). Kullanıcıların gizlilik haklarını koruyarak bir yakınlık algılama sistemi inşa etme noktasında literatürde birçok çalışma mevcuttur.


Li vd., (2011), mobil sosyal ağlarda (MSN) gizliliği koruyan profil eşleştirmesi sorununa üzerine çalışmış ve yüksek derecede kullanıcı gizliliğini korunmayı sağlayan iki somut öneri ortaya koymuşlardır. Hafif protokoller (lightweight protocols) tasarlamak amacıyla, temel güvenlik hesaplama teknigi olarak Shamir’in gizli paylaşım tekniği kullanılmaktadır, ayrıca önerilecek yöntemlerin iletişim maliyetlerini düşürmek amacıyla ek iyileştirmeler önerilmektedir. Kapsamlı güvenlik çözümlemesi ve simülasyon çalışması sonucunda, yöntemlerinin sadece mutlu ve meraklı modeli altında güveni olmadığını, aynı zamanda aktif saldırıları da önlediğini ve yöntemlerinin büyük ağlarda sorgu öznitelikleri sayısının profil özniteliklerinin sayısından daha küçük olduğu MSN'lerde, kullanılan en son teknoloji ürünlerden çok daha verimli olduğunu göstermiştirlerdir.

Başka bir çalışmada ise CenLocShare adı verilen merkezi bir gizlilik korumalı konum paylaşım sistemi önerilmiştir. Bu sistem SNS ve LBS ’yi tek bir sunucuya, yani Konumu Saklayan Sosyal Ağ Sunucusu’na (KSSAS) entegre etmekte ve gizliliği korunmuş konumları paylaşmak için kukla konumları ve KSSAS ile Hücresel Kule (HK) arasındaki özel


Diğer bir çalışmada ise kullanıcılar için gizliliği koruyarak arkadaş tavsiyesi yapma sorunu ele alınmıştır. Önceki çalışmalardan farklı olarak, bir k-derece anonimlikte arkadaş tavsiyesi (KAT) yöntemi tasarlanmıştır. Kullanıcının gizliliğini korumak amacıyla k-derecede anonim hiper- grafik ve anonim verinin kullanılabilirliğini sağlamak amacıyla da bölütleme ağacı kullanılmıştır. Kapsamlı güvenlik çözümlemesi uygulandığında, önerilen yöntemin gizliliği korunur, doğru bir şekilde arkadaş tavsiyesi hizmeti sağladığı gösterilmiştir. Gerçek ÇSA veri setlerinde yapılan deneyler, arkadaş tavsiye yöntemlerinin
etkili olduğunu ve kenar bölütleme (edge segmentation) algoritmasonun ölçeklenebilir olduğunu göstermektedir (Zhang vd., 2018).

Gizlilik Hakkını Koruyan Yakınlık Tespiti ile ilgili diğer çalışmalar Tablo 3.4’ded gösterilmiştir.

Tablo 3.4. ÇSA’da Gizlilik Hakkını Koruyan Yakınlık Tespiti ile ilgili yapılan diğer çalışmalar

<table>
<thead>
<tr>
<th>Yıllar</th>
<th>Açıklama</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007</td>
<td>Çevrimiçi mobil ağlarda konum gizliliğinin korunmasını amaçlayan yeni teknikler önerilmiştir (Ardagna vd., 2007).</td>
</tr>
<tr>
<td>2008</td>
<td>Sosyal ağ verilerinin yayılmamasının gizliliğini korumak için anonimleştirme tekniklerinin kapsamlı incelemesi sunulmuştur (Zhou vd., 2008).</td>
</tr>
<tr>
<td>2009</td>
<td>FriendLocator adlı bir istemci-sunucu, konum-gizlilik farklılığında bir arkadaş bulucu önerilmiştir (Šiksnys vd., 2009).</td>
</tr>
<tr>
<td></td>
<td>Orjinal gizlilik koruma teknikleri önerilmiştir (Mascetti vd., 2009).</td>
</tr>
<tr>
<td>2010</td>
<td>Ortak çevrimiçi topluluk tabanlı uygulamalarda gizliliği korumak için Bağlam Obstrüksiyonu için gizlilik duyarlı yeni bir mimari önerilmiştir (Rahman vd., 2010).</td>
</tr>
<tr>
<td>2011</td>
<td>Yer, yokluk, ortak konum ve kimlik gizliliği konuları ele alınmıştır (Vicente vd., 2011).</td>
</tr>
<tr>
<td>2012</td>
<td>Bir kullanıcının gerçek konumunu ve güvenli el sıkışmalarını açıklamak için gizliliğini koruyan yeni bir yakınlık testi gerçekleştirilmiştir (Zeng vd., 2012).</td>
</tr>
<tr>
<td></td>
<td>PeerSense adlı yeni bir sistem önerilmiştir (Gupta vd., 2012).</td>
</tr>
<tr>
<td>2013</td>
<td>Mobil jeososyal ağlar için yeni bir gizlilik koruyucu yakınlık algılama yöntemi önerilmiştir (Li vd., 2013).</td>
</tr>
<tr>
<td>2014</td>
<td>ÇSA'ların için konum bilgilerinin özel ve güvenli bir şekilde paylaşılmasına izin veren yeni bir hizmet önerilmiştir (Novak ve Li, 2014).</td>
</tr>
<tr>
<td>2015</td>
<td>Belirli bir kullanıcı için, bir arkadaşını korumak adına ortak konumunun yakınlık ölçüsünü kullanarak arkadaşını önermek üzere iki yeni yaklaşım sunulmuştur (Samanthula vd., 2015).</td>
</tr>
<tr>
<td></td>
<td>Kullanıcıların konum bilgilerini açıklamadan yerel tanımlı servis sunucusundan yakınlık testi sonuçlarının güvenirlüğünü doğrulanmasını sağlayan gizliliği koruyan doğrulanabilir bir yakınlık testi önerilmiştir (Zhou vd., 2015).</td>
</tr>
<tr>
<td></td>
<td>Konum gizliliği için InnerCircle adlı yeni bir güvenli çok bölümlü hesaplama protokolü sunulmuştur (Hallgren vd., 2015).</td>
</tr>
<tr>
<td>2016</td>
<td>Konum tabanlı hizmetler için EPPD adlı bir gizlilik koruyucu yakınlık test şeması sunulmuştur (Huang vd., 2016).</td>
</tr>
<tr>
<td>2017</td>
<td>mOSN'ler için yeni bir mimari ve Kullanıcı Tanımlı Gizlilik Yer Paylaşımı (UDPLS) sistemi adı verilen yeni bir program önerilmiştir (Sun vd., 2017).</td>
</tr>
<tr>
<td>2018</td>
<td>Gizliliği koruyan, konum duyarlı bir sosyal POI (ilgi noktası) önerisi sistemi önerilmiştir (Wang vd., 2018).</td>
</tr>
</tbody>
</table>

3.4. Anomali Tespiti

Popüler sosyal ağ analizi problemlerinden biri de anomali tespiti problemidir. Anomali tespiti genel olarak veriler arasında beklenmeyen davranış örtüntülerini bulma işidir. Bu uyumsuz örtüntüler genellikle anormallikler, aykırılıklar, uyumsuz gözlemeler, istisnalar, sapmalar, sürprizler veya kirleticiler olarak adlandırılır. Anomaliler, belirli bireylerin ya


Sun vd., (2017), kullanıcıların sosyal medyadaki duygusal modellemesi ve anormal duygusunu tespiti hakkında bir çalışma gerçekleştirmişlerdir. Çok değişkenli Gauss modeli ve birleşik
olasılık yoğunluğu kullanılarak mikro-bloglarda kullanıcıların anormal duyguları belirlemeneye çalışılmıştır. Bulgular, anormallık tespit kesinliğinin, kullanıcı ve ay bazında sırasıyla %83,49 ve %87,84 olduğunu göstermiştir. Deney aynı zamanda, bireysel kullanımın “nötr, mutluluk, üzünlük” duygularının KS testi ile normal dağılım gösterdiği, “ şaşırtıcı, kızgınlık” duygularının normal dağılım göstermediği ve grup tarafından yayınlanan mikro blogların hareketi kuvvet yasası dağılımına maruz kalırken, bireysel kullanıcı için bunun geçerli olmadığını göstermiştir.

Yasami ve Safaei, (2017) dinamik sosyal ağlarda anomali tespitinde yeni bir istatistiksel yaklaşım ortaya koymuşlardır. Bu yaklaşım, ağ dinamiklerinin, zaman içinde komşu düğümlere dinamik şekilde kademeli olarak eklenen her bir düğümün mikroskobik özellikleri tarafından yönlendirilebileceği gerçeğine dayanmaktadır. Önerilen yaklaşım iki ana bileşenden oluşmaktadır: (1) normal modellene bileşeni ve (2) anomali tespit bileşeni. İlk bileşen, ağ dinamiklerini yöneten üç ana süreçte yer almaktadır. İlk süreç, özelliklerin doğum, ölüm ve yaşam süresidir. İkinci süreç, özellik katmanlarını göz önünde alarak, Sonsuz Faktörlü Gizli Markov Modeli (SFGMM) ile modellenen düğüm özelliklerinin evrimidir. Özellik kaskadı, her düşümün geçmişteki özelliklerinin geleekte komşu düğümlerin özelliklerini nasıl etkileyeceğini açıkça tanımlayan bir olgudur. Çalışmalarında modellenen üçüncü süreç, düğümlerin özellikleri ile dinamik sosyal ağlarda bağlantı oluşturma arasındaki ilişkidir. Önerilen anomali tespit yaklaşıması, sentetik ve gerçek sosyal ağ veri kümeleri üzerinde yapılan deneylerle doğrulanmıştır. Deneysel sonuçlar, bu yaklaşımın, özellikle ikili normal-anormal sınıflama testine uygulandığında bazı istatistiksel performans ölçümleri açısından diğer benzer yaklaşımları nasıl geride bıraktığını göstermektedir.

3.4.1. Siber Suçlar

“Siber suç” terimi, günümüzde ağ teknolojilerinin yarattığı fırsatlardan yararlanılarak işlenecek suçları veya ortaya çıkan zararları tanımlamak için yaygın olarak kullanılmaktadır (Wall, 2008). Siber Suç kavramı bilişim kaynaklarını ve araçlarını kullanarak her türlü yasadışı eylem, her türlü korsan faaliyet, insanların ve kurumların özel hayatlarına müdahale eden, ekonomik anlamda kişi ve kurumları zarara uğratan faaliyetler olarak tanımlanabilir. Bilgi ve iletişim teknolojileri hükümlerle, şirketlere ve birçok kuruma kendi varlıklarını sürdürümek ve geliştirmek için büyük olanaklar sunmaktadır. Ama öbür yandan suç işleme
niyeti olanlara da fırsatlar sunmakta ve bireyleri, toplulukları, birçok alanda faaliyet yürüten örgütleri ve bir ulus yüksek oranda siber saldırı ve siber suç tehditlerine maruz bırakmaktadır (Choo, 2011). Geniş bir faaliyet alanı bulunan siber saldırılar sosyal ağların giderek yaygınlaşması ile bu platformlarda da çok yaygın bir biçimde varlığını göstermektedirler. Siber suçlular, sosyal medya kimlik avı (Aggarwal vd., 2012),  


3.4.1.1. Siber Zorbalık Tespiti


Yapılan başka bir çalışmada küfür ve siber saldırılan içeren önemli sayıda medya oturumunun siber zorbalık olarak etiketlenmediği gözlenmiş ve siber zorbaların tespitinin sadece küfür aramaktan daha karmaşık olması gerektiği ortaya konmuştur. Yüzde 60-70'ın


Ek olarak, başlangıç yöntemi tekrar oluşturulmuş ve performansı önerilen yöntemle karşılaştırılmıştır. Deneyin sonuçları, önerilen yöntemin başlangıç çizgisinden daha iyi sonuçlar elde ettiği göstermiştir. Adil koşullarda (% 50'si zararlı girdilerden oluşan test veri seti) önerilen yöntem ile %10'lık Geri Çıkarma değerinde % 90'ın üzerinde hassasiyet elde edilmiş ve yaklaşık %50 yakının Geri Çağırma değerinde Yüksek Hassasiyet (% 80-% 70) elde edilmiştir. Gerçek dünya koşulları altında (%12'si zararlı girdilerden oluşan test veri seti), yöntem %10'lık Geri Çağırma değerinde %50 hassasiyet elde etmiştir. Uyguluk eğrisi, önerilen yöntem için Geri Çağırmanın artmasıyla yavaş yavaş düşerken, başlangıç için ilgi düzeyi eğrisi, aynı Geri Çağırma oranında, aniden %30'dan %15'e düşmüştür. Önerilen yöntemdeki dezavantajlara gelince, kişisel bilgiler içeren zararlı gönderiler hem zararlı hem de zararlı olmayan gönderilerde sıkılıkla görülen nötr ifadelerin bulunması nedeniyle daha az zararlı olarak değerlendirilmiştir.

Başka bir çalışmada ise birbirinden farklı özellik kombinasyonları test edilmiş ve anlamalı bir ayırma gücü sahip olan ve daha iyi sonuçlar elde edebilen bir kombinasyonu belirlemek
amaçla farklı sayıda özellik tekrarlı olarak seçilmiştir. En anlamlı önerilen özelliklerin belirlenmesinde üç özelliği seçme algoritmaları, yani c2 testi, bilgi kazanımı ve Pearson koreasyonu seçilmiştir. Sentetik azınlık yüksek hızda örnekleme tekniği (SMOTE) yaklaşımı ve ağırlıkları düzenhlenen yaklaşım (maliyete duyarlı) veri kümesindeki sınıfları dengelemek için kullanılmıştır. Daha sonra, önerilen özellikler için en iyi kurgunun seçilmesi amacıyla çok farklı kurgu altında dört sınıflandırıcı, yani, Naive Bayes (NB), destek vektör mekanizması (DVM), rastgele orman ve k-en yakın komşunun (KYK) performansları karşılaştırılmıştır (Al-гарadi vd., 2016).

Siber Zorbalık Tespiti ile ilgili diğer çalışmalar Tablo 3.5’de gösterilmiştir.

Tablo 3.5. ÇSA’ da Siber Zorbalık Tespiti ile ilgili yapılan diğer çalışmalar

<table>
<thead>
<tr>
<th>Yıllar</th>
<th>Çalışma İçeriği</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007</td>
<td>Çevrimiçi sosyal medyada sanal zorbalık hakkında bir çalışma sunulmuştur (McKenna 2017).</td>
</tr>
<tr>
<td>2009</td>
<td>Ortaokul öğrencileri arasında siber zorbalık araştırmaştırılmış (Kift vd., 2009).</td>
</tr>
<tr>
<td>2010</td>
<td>Twitter’da zorbalık tespit etmek için duygu analizi kullanılmıştır (Patchin ve Hinduja 2010).</td>
</tr>
<tr>
<td>2011</td>
<td>• Sosyal medyada ABD’li gençlerin olumsuz davranışlarıyla karşılaşma deneyimi araştırılmıştır (Lenhart vd., 2011).</td>
</tr>
<tr>
<td></td>
<td>• Siber zorbalık tespitinin doğruğunu geliştirmek için bu çalışma kullanıcıların bilgilerini, özelliklerini ve taciz sonrası davranışlarını kullanmıştır (Dadvar ve De Jong 2012).</td>
</tr>
<tr>
<td></td>
<td>• Agresif içerikli tespit etmek ve sosyal medyada potansiyel saldırgan kullanıcıları tespit etmek için Lexical Syntactic Feature (LSF) mimarisi önerilmiştir (Chen vd., 2012).</td>
</tr>
<tr>
<td>2013</td>
<td>• Siber zorbalık olaylarının tespit araçlarının performansını artırmak için kullanılanın yorum geçmişi ve kullanıcı özellikleri gibi kullanıcı bağlı herhangi kullanılmış ve Siber zorbalık tespiti denetimli bir sınıflandırma görevi olarak kabul edilmiştir (Dadvar vd., 2013).</td>
</tr>
<tr>
<td></td>
<td>• Yakin dil analizi siber zorbalıkta kullanılmıştır (Kontostathis vd., 2013).</td>
</tr>
<tr>
<td>2014</td>
<td>• Ask FM sosyal ağında olumsuz kullanıcı davranışa (bazılarının intihar davranışına yol açması) üzerinde çalışılmıştır (Hosseini, 2014).</td>
</tr>
<tr>
<td>2015</td>
<td>• Siber zorbalığı saptamak için Bulanık mantığı ve genetik algoritmayı temel alan algoritma önerilmiştir (Nandhini ve Sheeba, 2015).</td>
</tr>
<tr>
<td>2016</td>
<td>• Siber zorbalık tespitinde orijinal bir çerçeve önerilmiştir (Zhao vd., 2016).</td>
</tr>
<tr>
<td>2017</td>
<td>• Semantik Geliştirilmiş Marjinalize Dengeleme Otomatik Enkoderi (smSDA) adlı bir yöntem, sosyal medyada zorbalık mesajlarının otomatik olarak algılanması için önerilmiştir (Zhao ve Mao 2017).</td>
</tr>
<tr>
<td></td>
<td>• Siber zorbalığın tespiti için Sentiment Informed Cyberbullying Detection (SICD) adlı yeni bir yöntem önerilmiştir (Dani vd., 2017).</td>
</tr>
<tr>
<td></td>
<td>• Ergenlerin sosyal medyayı nasıl kullanıkları, ergenlerin sosyalite ve çevrimiçi saldırıları ilişkili cinsiyet rolleri araştırılmıştır (Young vd., 2017).</td>
</tr>
<tr>
<td>2018</td>
<td>• Metin madenciliği teknigi, hesaplama yöntemleri ve siber zorbalık ağ görselleştirmesi kullanılarak bir siber zorbalık metin tespit çalışması yapılmıştır (Nurrahmi ve Nuranah 2018).</td>
</tr>
<tr>
<td></td>
<td>• Twitter’da sanal zorbalık algılanması için derin bir sinir ağının deneyimi bir yöntem önerilmiştir (Zhao vd., 2018).</td>
</tr>
</tbody>
</table>
3.4.1.2. Spamlama Tespiti


32
Tablo 3.6. ÇSA’da Spamlama Tespiti Tespiti ile ilgili yapılan diğer çalışmalar

<table>
<thead>
<tr>
<th>Yılı</th>
<th>Önemli Çalışma</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007</td>
<td>Çevrimiçi sosyal genel ağı sitelerinde spam'lar ile mücadele hakkında bir anket sunulmuştur (Heymann vd., 2007).</td>
</tr>
<tr>
<td>2008</td>
<td>Çevrimiçi sosyal ağlarda video spam göndericilerinin tespit edilmesi için yeni bir yaklaşım önerilmiştir (Benevenuto, 2008).</td>
</tr>
<tr>
<td>2009</td>
<td>Twitter'ın ve Twitter'ın geliştiricilerinin yapısında aşırı ağ girişimlerinin tespit edilmesi için yeni bir yaklaşım önerilmiştir (Yardi vd., 2009).</td>
</tr>
<tr>
<td>2010</td>
<td>Twitter'daki şüpheli kullanıcılara tespit etmek için bir spam tespit çerçevesi sistemi önerildi (Wang, 2010).</td>
</tr>
<tr>
<td>2011</td>
<td>OSN'lerde hesapları kullanılarak başlatılan spam miktarını belirlemek için bir çözüm önerilmiştir (Gao vd., 2011).</td>
</tr>
<tr>
<td>2012</td>
<td>Gerçek zamanlı kullanıcıların massejlerinin OSN'lerde kontrol edilmesi için bir çevrimiçi spam tespit sistemi önerilmiştir (Gao vd., 2012).</td>
</tr>
<tr>
<td>2014</td>
<td>Hesap bilgilerini ve aksi tweet'leri kullanarak Twitter'da spam gönderenleri tespit edilmesi için bir anormallık algılama sistemi önerilmiştir (Miller vd., 2014).</td>
</tr>
<tr>
<td>2015</td>
<td>Twitter'ın ve Twitter'ın geliştiricilerinin yapısında aşırı ağ girişimlerinin tespit edilmesi için yeni bir çözüm önerilmiştir (Wang vd., 2015).</td>
</tr>
<tr>
<td>2016</td>
<td>Mikroblogda sosyal spam gönderen ve istemeyen kesinlikle algılanması için bir sistem önerilmiştir (Wu vd., 2016).</td>
</tr>
<tr>
<td>2018</td>
<td>Spam gönderen ve spam gönderenlerin Facebook'taki kullanıcı profilerinin ayrıntılı olarak incelenmesi için bir çözüm önerilmiştir (Fu vd., 2018).</td>
</tr>
</tbody>
</table>

3.4.1.3. Zararlı Yazılım Tespiti


Yapılan başka bir çalışmada ise yalnızca kullanıcının oluşturduğu sosyal ağ grafiğine değil, aynı zamanda kullanıcının etkinlik olaylarını de içeren gerçek dünya konum tabanlı bir çevrimiçi sosyal ağdan derlenen bir veri kümesine dayanmaktadır. Bu ağın sosyal yapısı ve kullanıcı etkinlik kalıpları çözümlenmiştir ve bu özel ağdan elde edilecek sonuçların diğer çevrimiçi sosyal ağlar için de geçerli olacağını gösterilmiş tipik bir çevrimiçi sosyal ağ olduğu belirlenmiştir. İlk bulaşmanın, kullanıcının tıklama olasılığının, sosyal yapının ve etkinlik kalıplarının çevrimiçi sosyal ağlarda kötü amaçlı yazılımların yayılımı üzerinde etkisini incelemek üzere kapsamlı bir simülasyon kullanılmıştır. Ayrıca, çevrimiçi sosyal ağlarda kötü amaçlı yazılım yayılmasına karşı kullanıcı odaklı ve sunucu odaklı bazı savunma yöntemlerinin performansını araştırılmış ve bunların etkinliğini etkileyen temel faktörler belirlenmiştir (Yan vd., 2011).


Zararlı Yazılım Tespiti ile ilgili diğer çalışmalar Tablo 3.7’de gösterilmiştir.

Tablo 3.7. ÇSA’ da Zararlı Yazılım Tespiti ile ilgili yapılan diğer çalışmalar

<table>
<thead>
<tr>
<th>Yılı</th>
<th>Çalışma Başlığı</th>
<th>Yazarlar</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007</td>
<td>Çevrimiçi sosyal ağlarda bilgisayar virüs yayılımını incelemek için sosyal ağ analizi tekniklerinden faydalanmıştır (Guo ve Cheng, 2007).</td>
<td></td>
</tr>
<tr>
<td>2008</td>
<td>Çevrimiçi hizmetle ilgili güvenlik sorunları tanıtılmıştır (Mansfield-Devine, 2008).</td>
<td></td>
</tr>
<tr>
<td>2009</td>
<td>ÇSA’larda aktif solucan propagasyonunu çözümlemek ve simüle etmek için genel bir model sunulmuştur (Faghani ve Saidi 2009).</td>
<td></td>
</tr>
<tr>
<td>2010</td>
<td>Erken uyarı ÇSA solucan algılama sistemi sunulmuştur (Xu vd., 2010).</td>
<td></td>
</tr>
<tr>
<td>2011</td>
<td>Dinamik sosyal ağlarda solucan yayımı için yeni bir adaptif yöntem geliştirilmiştir (Nguyen vd., 2010).</td>
<td></td>
</tr>
<tr>
<td>2011</td>
<td>Facebook ağına dayanan bir virüs yayılım çerçevesi önerilmiştir (Fan ve Yeung, 2011).</td>
<td></td>
</tr>
<tr>
<td>2012</td>
<td>Salgın teorisine dayanan Twitter kötü amaçlı yazılım yayılımı için temel bir model önerilmiştir (Sanzgiri vd., 2012).</td>
<td></td>
</tr>
<tr>
<td>2013</td>
<td>Sosyal ağ'an yayılması modellemek için yeni bir analitik model önerilmiştir. (Wen vd., 2013).</td>
<td></td>
</tr>
</tbody>
</table>
• Kullanıcı davranışları, toplulukların ve topluluk boyutlarının yüksek kümelenmiş yapısı, XSS solucanı yayılma hızı üzerinde ana faktörler olarak kabul edilmiş ve tanımlanmıştır (Faghani ve Nguyen, 2013).

2014 • Çevrimiçi sosyal ağlarda virüs yayılımını önleme ve kontrol etmek için yeni bir topluluk tespit algortması önerilmiştir (Liu vd., 2014).
• ÇSA’larda ClickJacking Worm Propagation çalışıldı (Faghani ve Nguyen, 2014).

2015 • İnsan hareketliğinin zamansal karakteristiğine dayalı olarak āhiret ağaçları sosyal solucanların yayılma özelliklerini ortaya koyan yeni bir çerçeve tartışılmiştir (Wang vd., 2015).

2016 •ÇSA’ların platformlarına XSS saldırılarının ayrıntılı olayları tartışmıştır (Chaudhary vd., 2016).

2017 • ÇSA’larda kötü amaçlı yazılım yayılımının performansını çözümleme için yeni bir kötü amaçlı yazılım yayma modeli, bir ileti-yeniden hesaplanabilir mekanizma önerilmiştir (Chen vd., 2017).
• Sosyal ağlarda arama motorunda virüs yayılımı etkilenir ve virüs yayılım sürecinin kararlılığı incelenmiştir (Fu vd., 2017).

2018 • Sosyal ağlarda virüs yayılımını karakterize etmek için HSID (heterojen-duyarlı-bulaşıcı-uyku model) olarak adlandırılan yeni bir model önerilmiştir (Jia vd., 2018).

3.4.1.4. Sahtekârlık Tespiti

Sahtekarlık Tespiti alanında birçok çalışma mevcuttur. Şekil 3.2’de sahtekarlık alanlarını gösteren bir şema bulunmaktadır.

Şekil 3.2. Sahtekarlık alanlarının genel taksonomisi (Abdallah vd., 2016)


A. Kimlik Avı Tespiti

Kimlik avı, saldırımanın yararına belirli eylemleri gerçekleştirmeleri için kullanıcılara ikna etmek amacıyla, kullanıcılara elektronik iletişim kanalları aracılığıyla elektronik mesaj

Aggarwal vd., (2012), yaptıkları çalışmada sundukları PhishAri metodu ile Twitter’de kimlik avı saldırılarını gerçek zamanlı olarak tespit etmiştir. PhishAri, URL içeren bir tweet’in kimlik avı içerip içermediğini tespit etmek için Twitter özelliklerinin yanı sıra URL

Kimlik Avı Tespiti ile ilgili diğer çalışmalar Tablo 3.8’de gösterilmiştir.

Tablo 3.8. ÇSA’ da Kimlik Avı Tespiti ile ilgili yapılan diğer çalışmalar

<table>
<thead>
<tr>
<th>Yılı</th>
<th>onerilen Daha</th>
</tr>
</thead>
<tbody>
<tr>
<td>2009</td>
<td>Sosyal Ağ Sitelerine kimlik avı saldırıları (URL-9, 2018).</td>
</tr>
<tr>
<td>2010</td>
<td>Facebook'taki “duvar” mesajlarının büyük bir veri kumesine dayanarak kötü amaçlı içeriği tespit etme hakkında bir çalışma gösterilmiştir (Gao vd., 2010).</td>
</tr>
<tr>
<td>2014</td>
<td>Çevrimiçi mobil platformlar ağları için MobiFish adlı yeni bir kimlik avı sistemi sunulmuştur (Wu vd., 2014).</td>
</tr>
<tr>
<td>2015</td>
<td>Kimlik avı, kötü amaçlı yazılım veya diğer zararlı içerik barındırıldığında sosyal medya spam URL’leri hakkında bir araştırma sunulmuştur (Cao và Caverlee, 2015). Facebook alışkanlıkları ve kimlik avı saldırıların ilişkisi incelenmiştir (Vishwanath 2015).</td>
</tr>
<tr>
<td>2017</td>
<td>Mızrak avcılığı deneysel çalışması için 1200'den fazla üniversite öğrencisi bir E-posta veya bir Facebook mesajı gönderilmiştir (Benenson, 2017).</td>
</tr>
</tbody>
</table>

### B. Profil Klonlama Tespiti


Profil klonlama ile yapılan başka bir çalışmada ise iki yeni klonlama tespit yöntemi önerilmiştir. İlk yöntem, her iki profilin özellikleri arasındaki benzerliğe, ikinci yöntem ise ilişki ağlarının benzerliğine dayanmaktadır. Önerilen yeni yöntemler yapılan deneylerle test edilmiş ve önerilen yöntemlerin mevcut yöntemlerle kıyasla yararlı ve verimli olduğunu ortaya çıkmıştır. Bu çalışma, ayrıca, Facebook’ta profil klonlamanın sadece mümkün olmakla kalmayıp, aynı zamanda da uygulanması oldukça kolay olarak bir çalışma sunulmuştur.

Profil Klonlama Tespiti ile ilgili diğer çalışmaları Tablo 3.9’da gösterilmiştir.

Tablo 3.9. ÇSA’da Profil Klonlama Tespiti ile ilgili yapılan diğer çalışmalar

<table>
<thead>
<tr>
<th>Yıllar</th>
<th>Çalışmalar</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007</td>
<td>Facebook kullanıcıları kimlik hırsızlığı için kolay hedefler (URL-10, 2018).</td>
</tr>
<tr>
<td>2008</td>
<td>Bloglarda gönüllü kendi kendi işta etmenin faktörleri araştırılmış (Le vd., 2008).</td>
</tr>
<tr>
<td>2009</td>
<td>Çevrimici platformlarda açılan hassas bilgilerin kimlik hırsızlığıyla ilgili, kişisel güvenlik çalışmıştır (Fogel ve Nehmad 2009).</td>
</tr>
<tr>
<td></td>
<td>Facebook ile ilgili güvenlik sorunları sunulmuştur (Leitch ve Warren, 2009).</td>
</tr>
<tr>
<td>2010</td>
<td>Kullanıcının gizlilik ayarını otomatik olarak yapılandırarak için Gizlilik Sihirbazi şablon tasarımı sunulmuştur (Fang ve LeFevre 2010).</td>
</tr>
<tr>
<td></td>
<td>Çevrimiçi ağı profilinde bilgi açıklamayı hedeflemiş (Nosko vd., 2010).</td>
</tr>
<tr>
<td>2011</td>
<td>Şüpheli kimlikleri keşfetmeye ve bünüleri onaylamaya odaklanan yeni bir tespit çerçevesi önerilmiştir (Jin vd., 2011).</td>
</tr>
<tr>
<td>2012</td>
<td>Sybil saldırıları ve Kimlik Klon saldırıları arasındaki farklılıklar tartışmıştır (Jin vd., 2012).</td>
</tr>
<tr>
<td></td>
<td>ÇSA da Şahit Profil Saldırısı için yeni bir çerçeve önerilmiştir (Conti vd., 2012).</td>
</tr>
<tr>
<td>2013</td>
<td>Çalışma Yelp, Flickr ve Twitter’da üç özellik kullanılmış, Sorunun tanımlanması için coğrafi konum, yayınların zaman damgası ve kullanıcı yazım stili,(Goga vd., 2013).</td>
</tr>
<tr>
<td></td>
<td>Şüpheli kimliği tespit etmek için özellik benzerliğine ve arkadaş ağına benzer bir yöntem önerilmiştir (Khayyam bashi ve Rizi, 2013).</td>
</tr>
<tr>
<td>2014</td>
<td>Kullanıcıları Kimlik Hırsızlığı Saldırılara karşı koruyan yeni bir program sunulmuştur (He vd., 2014).</td>
</tr>
</tbody>
</table>
Profil Klonlama Saldırılarını tespit etmek için Facebook, LinkedIn ve Google+'da bir yöntem uygulanmıştır (Devmane ve Rana, 2014).

ÇSA'lerde büyük bir takım kimliğe bürünüme saldırıları oluşturmak için yeni bir teknik önerilmiştir (Goga vd., 2015).

ÇSA'lerde yeni bir profil klonlama tespit konsepti sunulmuştur (Zabielski vd., 2016).

Bayes ağına dayalı olasılıksal bir üretkenlik modeli kullanılan MSN'lerde kimlik hırsızlığını tespit etmek için yeni bir yöntem önerilmiştir (Wang vd., 2017).

Kimlik temelli siber suçlarla ilgili yeni bir inceleme sunulmuştur (Bahri et al. 2018).

Farklı ÇSA sahalarında ele geçirilmiş profiller, klonlanmış profiller ve çevrimiçi botlar (spam botlar, sosyal botlar, botlar ve etkili botlar) üzerinde çalışılmıştır (Wani vd., 2018).

3.4.2. Söylenti Tespiti

gönderenleri çözümelemek ve bulmak için tweetleri postalamaları kullanmışlardır. Tweetleme davranışlarını kontrol ederek, spam göndericilerin yanlış bilgi yayılama ve yarıştırmacı takipçileri bulma olasılığı daha yüksektir (Yang vd., 2015).


Söylenti Tespiti ile ilgili diğer çalışmalar Tablo 3.10’da gösterilmiştir.
<table>
<thead>
<tr>
<th>Yılı</th>
<th>Şöylentenin yayılması için güvenilir bir dedikodu protokolü önerilmiştir (Mitra ve Maheswaran, 2007).</th>
</tr>
</thead>
<tbody>
<tr>
<td>2008</td>
<td>Oyun teorisi ve konum teorisi kavramları sosyal ağlarda dedikodu yayılmasını analiz etmek için kullanılmış (Kosta vd., 2008).</td>
</tr>
<tr>
<td>2009</td>
<td>Sosyal ağ için dedikodu protokolüne dayalı alternatif bir yaklaşım önerilmiştir (Abbas vd., 2009).</td>
</tr>
</tbody>
</table>
| 2010 | • Söylentilerin sosyal ağda yayılması için dedikodu benzeri yöntemler sunulmuştur (Tripathy vd., 2010)
  • Sosyal medyada söyleni difüzyonunun gerçek zamanlı çözümlenmesi sağlayacak yeni bir çerçeve sunulmuştur (Ratkiewicz vd., 2010). |
| 2011 | • Bir deprem felaketinden sonra nasıl söyleniler yayılmıştır ve bunlarla nasıl başa çıkabiliriz konusunu tartışan bir çalışmadan (Takahashi ve Igata, 2012)
  • Sina Weibo üzerindeki söylenilerin otomatik olarak tespit edilmesi çalışılmıştır (Yang vd., 2012).
  • En çok kuşkulanan ve en çok kuşku duyulan kaynakların araştırılması için bir Sosyal Medya Söyleni Bilgisi Saptama (Suspect) sistem önerilmiştir (Nguyen vd., 2012). |
| 2012 | • Sina Weibo ve Sosyal Medya Söyleni Bilgisi Saptama (Suspect) sistem önerilmiştir (Fan vd., 2013). |
| 2013 | • ÇSA'nın yayılmasına yönelik bir platform olan Hoaxy önerilmiştir (Shao vd., 2016).
  • Youtube ve Twitter üzerinden Sosyal Medya Söyleni Bilgisi Saptama (Suspect) sistem önerilmiştir (Liu ve Xu, 2016). |
| 2014 | • ÇSA'nın yayılmasına yönelik bir platform olan Hoaxy önerilmiştir (Shao vd., 2016).
  • Youtube ve Twitter üzerinden Sosyal Medya Söyleni Bilgisi Saptama (Suspect) sistem önerilmiştir (Liu ve Xu, 2016). |
| 2015 | • ÇSA'nın yayılmasına yönelik bir platform olan Hoaxy önerilmiştir (Shao vd., 2016).
  • Youtube ve Twitter üzerinden Sosyal Medya Söyleni Bilgisi Saptama (Suspect) sistem önerilmiştir (Liu ve Xu, 2016). |
| 2016 | • ÇSA'nın yayılmasına yönelik bir platform olan Hoaxy önerilmiştir (Shao vd., 2016).
  • Youtube ve Twitter üzerinden Sosyal Medya Söyleni Bilgisi Saptama (Suspect) sistem önerilmiştir (Liu ve Xu, 2016). |
| 2017 | • Sosyal medyada sahte haber tespiti hakkında bir araştırma önerilmiştir (Shu vd., 2017).
  • Rumence sınıflandırması için, iki farklı metin göstergesi, bir kelime modeli çantası ve sinir ağları dil modeli çözümlenmiştir (Ma vd., 2017).
  • Hillary Clinton ve Donald Trump'un takipçilerinden dedikodu tweetleri çözümlemişi ve karşılaştırmalı bir çalışma için bir platform önerilmiştir (Jin vd., 2017). |
| 2018 | • Söylenti temsili öğrenme ve sınıflandırma için iki özayneleri nöral modeli sunulmuştur (Ma vd., 2018). |
3.5. İroni ve Kinaye Tespiti


İroni ve Kinaye Tespiti ile ilgili diğer çalışmalar Tablo 3.11’de gösterilmiştir.
Tablo 3.11. ÇSA’ da İroni ve Kinaye Tespiti ile ilgili yapılan diğer çalışmalar

<table>
<thead>
<tr>
<th>Yılı</th>
<th>Çalışma-detayları</th>
</tr>
</thead>
<tbody>
<tr>
<td>2010</td>
<td>• Twitter’a ve Amazon ürün incelemeleri veri kümelerine yarı denetimli alaylı tanımlama yöntemleri uygulanmıştır (Davidov vd., 2010).&lt;br&gt;• SASI adı verilen Kinaye tanımlama için yeni bir Yarı Kontrollü Algoritma önerilmiştir (Tsur vd., 2010).</td>
</tr>
<tr>
<td>2011</td>
<td>• Alayçi tespit görevi için alayçi Twitter birliği kurmak için yeni bir yöntem sunulmuştur (González-Ibáñez vd., 2011).</td>
</tr>
<tr>
<td>2012</td>
<td>• Çevrimiçi sosyal medyada ironi ve mizah çalışması yapılmıştır (Reyes vd., 2012).</td>
</tr>
<tr>
<td>2013</td>
<td>• Flemençe tweetlerin kinaye tespiti uygulanmıştır (Liebrecht vd., 2013).&lt;br&gt;• Twitter için bir kinaye tanıycı yöntemi önerildi (Riloff vd., 2013).</td>
</tr>
<tr>
<td>2014</td>
<td>• Kinaye tweetlerde kullanım ve duygusal analize etki incelenmiştir (Maynard ve Greenwood, 2014).</td>
</tr>
<tr>
<td>2015</td>
<td>• Twitter’da bir kinaye algılama yöntemi sunulmuştur (Rajadesingan vd., 2015).</td>
</tr>
<tr>
<td>2016</td>
<td>• Yunan siyasi tweet’lerinde ironi tespiti için bir sınıflandırma şeması sunulmuştur (Charalampakis vd., 2016).&lt;br&gt;• Kinaye algılama için yeni bir sinir ağ modeli önerildi (Charalampakis vd., 2016).</td>
</tr>
<tr>
<td>2017</td>
<td>• Otomatik sarcasm algılama hakkında bir ankет sunulmuştur (Joshi vd., 2017).&lt;br&gt;• Kinaye ve huysuzluk farklılıkları belirleyebilmek adına, Monologic alanda sınıflandırıcıları eğitmek için bir yöntem önerilmiştir (Lukin ve Walker 2017).</td>
</tr>
<tr>
<td>2018</td>
<td>• Twitter için yeni bir ironi tespit yöntemi önerilmiştir (Van Hee vd., 2018).&lt;br&gt;• İngilizce tweetlerde ironinin saptanması için sinir ağına dayalı yeni bir model önerilmiştir (Vu vd., 2018).</td>
</tr>
</tbody>
</table>

3.6. Rol Madenciliği

Sosyal rol keşfi motivasyonu sosyolojideki Role Teorisi’nden almıştır. Rol keşfi insanların toplumsal rollerine dayanarak öngörülebilir şekillerde davranışı gözlemler ve bu bir kişinin rolünün bilinmesinin çevreyle olan etkileşimlerini belirlemeye ya da bunun tam tersinin belirlenmesine yardımcı olabileceği göstermektedir (Biddle, 1986). Rol keşfi ilk olarak sosyolojide bir kişinin bir toplumdaki spesifik işlevini (bir baba, doktor ve bir akademik danışman) açıklamak için kullanılmıştır. Sosyologlar tarafından tanımlanan bu roller sosyal roller olarak tanımlanır. Buradan gidilerek rol keşfi sosyal ağ analizinde önemli bir konu haline gelmiştir. Önceleri rol keşfi özellikle küçük kapalı bir sosyal ağdaki (örneğin on düğümden oluşan bir grafik) aktörlerin rollerini anlamaya çalışan sosyologların ilgilendiği bir alan idi. Son zamanlarda rol keşfi sosyal ağlar, teknolojik ağlar, biyolojik ağlar, genel ağ grafiplerinde ve başka birçok alanda keşfedilmeye başlamıştır. Rol keşfinin genel grafik madenciliğinde ve keşif çözümlenmesinde önemli olması rağmen aynı zamanda başka alanlarda de faydalı olabilir. Örneğin, roller IP-izleri gibi teknolojik ağlardaki anormalliklerin...

3.6.1. Fikir Lideri Tespiti


48


Fikir Lideri Tespiti ile ilgili diğer çalışmalar Tablo 3.12’de gösterilmiştir.

Tablo 3.12. ÇSA’da Fikir Lideri Tespiti ile ilgili yapılan diğer çalışmalar

<table>
<thead>
<tr>
<th>Yılı</th>
<th>Çalışmanın Özeti</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007</td>
<td>Blogosferde fikir liderlerini tanımlamak için InfluenceRank adlı yeni bir algoritma önerilmiştir (Song vd., 2007).</td>
</tr>
<tr>
<td>2008</td>
<td>Blog sisteminde mesaj yayılımı süreci ve fikir lideri masaj difüzyonunun etkisi üzerine bir çalışma sunulmuştur (Ko vd., 2008).</td>
</tr>
<tr>
<td>2009</td>
<td>&quot;Yeni başlayanlar&quot; ve &quot;takipçi&quot; kavramları sosyal medyada formüleleştirilmiştir (Mathioudakis ve Koudas, 2009).</td>
</tr>
<tr>
<td></td>
<td>Twitter'da etkili kullanıcıların sıralı bir listesini tanımlamak için bir Sayfa Sıralaması yöntemi uygulanmıştır (Kwak vd., 2010).</td>
</tr>
<tr>
<td></td>
<td>TwitterRank adı verilen ve Twitter konularını ve tweet'lerden işlenen bilgileri belirli konulardaki uzmanları tanımlamak için kullanılan Sayfa Sıralamasi benzeri bir algoritma önerilmiştir (Weng vd., 2010).</td>
</tr>
<tr>
<td></td>
<td>Tweet etkisinin üç ölçütü: bağımsız (bir kullanıcıyı takip eden kişi sayısı), Twitter'daki kullanıcıların sıralamak için retweetler (başkalarının bir kullanıcının tweet'lerini ileten başkalarının sayısı) ve bahsettikleri (başkalarının bir kullanıcının adına belirten değerlerin sayısı) kullanılmıştır (Cha vd., 2010).</td>
</tr>
</tbody>
</table>
• Bir kullanıcının uzmanlık derecesini elde etmek ve sıralı bir uzman listesi oluşturmak için bir kullanıcının bilgi puanını ve yetki puanını birleştiren bir mezhep yöntem önerilmiştir (Kao vd., 2010).

2011 • Twitter ağındaki kullanıcılarnın etkisini belirlemek için Etki Pasifliği algoritması sunulmuştur (Romero vd., 2011).
• Topluluk soru-cevaplama sitelerinde hem hedef hem de ilgili kategorilerdeki bilgileri kullanarak yeni bir otorite sırallama yaklaşımı önerilmiştir (Zhu vd., 2011).

2012 • Olumlu görüş lideri grubunun tespit edilmesine yönelik bir çalışma 2011 yılında (Romero vd., 2011).
• Twitter'da konu uzmanlarını bulunmak için Congos adlı yeni bir sistem önerilmiştir (Ghosh vd., 2012).
• LinkedIn sosyal ağ, yetki puanlarına göre sıralı bir kullanıcı listesi bulmak için kullanılmıştır (Budalakoti ve Bekkerman, 2012).
• Çevrimiçi topluluk bloglarında etkili kullanıcıların sıralı bir listesi tespit etme konusu üzerinde çalışılmıştır (Agarwal vd., 2012).

2013 • Yeni bir MapReduce tabanlı Microblog tarayıcısı ve düğüm arasındaki ortalama hesaplama yöntemi önerilmiştir (Deng vd., 2013).
• Liderleri ve takipçileri tanımlamak için Boylamlar Kullanıcı Merkezli Etkisi modeli önerilmiştir (Shafiq vd., 2013).


2015 • Fikir liderleri için parametrik olmayan bir karışım modeli temelli yaklaşım önerilmiştir (Bouguessa ve Romdhane 2015).

2016 • Çevrimiçi topluluklarda fikir liderleri tespit etmek için yeni bir çerçeve sunulmuştur (Aghdam ve Navimipour, 2016).
• Çevrimiçi gözden geçirme topluluklarında konuya duyarlı fikir liderlerinin belirlenmesindeki zorluklar üzerinde çalışılmıştır (Miao vd., 2016).
• Sosyal medya kanallarında fikir liderlerinin özellikleri sunulmuştur (Winter ve Neubaum, 2016).

2017 • Sosyal medyada fikir liderleri kişilik özellikleri seyahat endüstrisi bağlamında incelenmiştir (Song vd., 2017).

2018 • Çevrimiçi fikir liderlerinin liderlik işlevleri, bilgi liderliği işlevleri ve aldıkları çoklu roller araştırılmıştır (Lin vd., 2018).
• Bir e-ticaret topluluğunda, bir grubun etkileşim mekanizmasının ve grup liderinin oluşturulması sırasında fikir liderlerinin etki gücü incelenmiştir (Zhao vd., 2018).

3.7. Konu ve Olay Tespiti


Sonuç olarak elde ettikleri sonuçlar diğer klasik metotlarla karşılaştırılmıştır ve iyi sonuçlar elde ettiklerini ortaya koymuşlardır.


Konusu ve Olay Tespiti ile ilgili diğer çalışmalar Tablo 3.13’te gösterilmiştir.

Tablo 3.13. ÇSA’da Konu ve Olay Tespiti ile ilgili yapılan diğer çalışmaları

<table>
<thead>
<tr>
<th>Yılı</th>
<th>Önemli Gelişmeler</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007</td>
<td>• Flickr'deki fotoğrafların etiketleri için yer ve olay semantiklerinin öğrenilmesi çalışılmıştır (Rattenbury vd., 2007).</td>
</tr>
<tr>
<td>2008</td>
<td>• Kleinberg'in algoritması blogosferde sıcak konu tespiti için sunulmuştur (Platakis vd., 2008).</td>
</tr>
<tr>
<td>2009</td>
<td>• Flickr görsel topluluk genel ağı sitesindeki fotoğraflardan olayları tespit etmek için önerilmiştir (Chen ve Roy, 2009).</td>
</tr>
<tr>
<td></td>
<td>• Bloglarda yeni bir olay tespiti yöntemi incelenmiştir (Jurgens ve Stevens, 2009).</td>
</tr>
<tr>
<td>2010</td>
<td>• Twitter kullanarak ilgili etkinliklerin belirlenmesi önerilmiştir (Popescu vd., 2010).</td>
</tr>
<tr>
<td></td>
<td>• Twitter daki olayların gerçek zamanlı etkileşimini, tweetleri takip etmek ve bir hedef olayı tanımlamak için bir algoritma sunulmuştur (Sakaki vd., 2010).</td>
</tr>
<tr>
<td>2011</td>
<td>• Otomatik olay tespiti için NLP tekniklerinin çevrimiçi olarak kullanılması önerilmiştir (Gu vd., 2011).</td>
</tr>
<tr>
<td>2012</td>
<td>• Doküman benzerliği ve kümeleme algoritmaları kullanılarak, tweet içeriklerinin sözlüksel anlamalı genişlemesine dayalı yeni bir olay tespiti yaklaşımı önerilmiştir (Ozdkis vd., 2012).</td>
</tr>
<tr>
<td></td>
<td>• TwiCal - Twitter için ilk açık alanlı olay çarkıma ve sınıflandırma sistemi tanıtılmıştır (Ritter vd., 2012).</td>
</tr>
<tr>
<td></td>
<td>• Twevent olarak adlandırılan tweet'ler için segment tabanlı bir olay tespiti yöntemi önerilmiştir (Li vd., 2012).</td>
</tr>
<tr>
<td></td>
<td>• Twitter tabanlı Olay Algılama ve Analiz Sistemi (TEDAS) önerilmiştir (Li vd., 2012).</td>
</tr>
<tr>
<td>2013</td>
<td>• Doğrulukta önem vermeden, smrsüz metin aksılarına öleçeklenen olay tespiti için bir yaklaşıım önerilmiştir (Petrovic, 2013).</td>
</tr>
<tr>
<td></td>
<td>• Avustralya ve Yeni Zelanda için bir deprem dedektörü Twitter kullanılarak önerildi (Robinson vd., 2013).</td>
</tr>
<tr>
<td>2014</td>
<td>• Yer-zaman kısıtlamalı konu (LTT) adlı bir grafik modeli kullanılarak, Twitter sosyal medya aksıları için yeni bir olay tespit çerçevesi önerilmiştir (Zhou vd., 2014).</td>
</tr>
<tr>
<td></td>
<td>• Çevrimiçi iteratif ileme konusunda konu tespiti için bir çalışma yapılmıştır (Spina vd., 2014).</td>
</tr>
<tr>
<td>2015</td>
<td>• Twitter'da etkinlik tespiti için bir anket sunulmuştur (Atefeh ve Khreich, 2015).</td>
</tr>
<tr>
<td></td>
<td>• Konu tespiti için teknik değer artırılma ve K-n Kümeleme yöntemi kullanılmaktır (Nur'aini vd., 2015).</td>
</tr>
<tr>
<td>2016</td>
<td>• Çevrimiçi sosyal ağ olay tespiti anketi sunulmuştur (Cordeiro ve Gama, 2016).</td>
</tr>
<tr>
<td></td>
<td>• Sosyal medyada kitle kaynak kullanımını kullanılarak gerçek zamanlı kentsel acil durum olayını tespit etmek için yeni bir yöntem önerilmiştir (Xu vd., 2016).</td>
</tr>
</tbody>
</table>

53
2017 • Spesifik bilgilerin tespit edilmesi için Konuma Özgü Bilgi Algılama Modeli önerilmiş ve sosyal medyada kullanılan olay için Stanford CoreNLP'ye dayanmış 4-Tuple Yapısi önerilmiştir (Xu vd., 2017).
   • Weibo'da bir Desen Tabanlı Konu Tespit ve Analiz Sistemi önerilmiştir (Zhang vd., 2017)

2018 • Sıcak konu tespiti için paralel iki fazlı mik-mak sıcak konu tespiti olarak adlandırılan yeni bir yinelemesiz algoritma önerilmiştir (Ai vd., 2017)
   • TUS-LDA yöntemi, gönderilerden duyarlılıkla ilgili konuları araştırmak için kullanılması önerilmiştir (Xu vd., 2018)
   • Olay tespiti için yeni bir olay tespit algoritması önerilmiştir ve tespit edilen olayların hiyerarşik bir şekilde görselleştirilmesi sunulmuştur (Akhtar vd., 2018)
   • Twitter'da etkinlik özet için yeni bir sistem önerilmiştir (Meladianos vd., 2018)

<table>
<thead>
<tr>
<th>Yılı</th>
<th>Gelişme detayı</th>
</tr>
</thead>
</table>

3.7.1. Tartışmalı Konuların Tespiti


Bu çalışmada sosyal medyanın nasılsağ kamusal alanını şekillendirdiği ve farklı politik yönelimlere sahip topluluklar arasındaki iletişimi kolaylaştırdığı araştırılmıştır. Twitter'deki


ÇSA’lardaki tartışmalı konuların tanımlanması amacıyla yerel grafik motiflerine dayalı yeni bir yöntem dilden bağımsızdır ve tartışmalı konular hakkında yapılan tartışma bağlanımlarını tespitinde yerel kullanıcı etkileşim kalıplarını kullanmaktadır. Bir içerik öğesi hakkında, kullanıcılar yanıt grafiğinin farklı konfigürasyonlarını oluşturarak birbirlerine yanıt vermektedir. Bu grafikten ve kullanıcı arkadaşı grafiğinden elde edilen yerel motifler araştırılmıştır. Bu tür motifler, büyük olasılıkla karşılıklı yanıt eylemi ve büyük olasılıkla karşılıklı arkadaşlık ilişkisi ile bağlı sadece kullanıcı arasındaki farklı karşılıklı etkileşim kalıplarına tekabül ederler. Üç kullanıcıın etkileşimi ile ilgili olarak da benzer motifler ele alınmıştır. Kısıt bir Twitter veri kümesi...
kullanarak, bu tür motivlerin, kaskad derinliği gibi sık kullanılan diğer başlangıç grafik özelliklerinden ziyade, tartışmalı konuları tahmin etmede daha güçlü olduğu kantlanmıştır. Tartışmalı konuların tespitinde özellikle iki boyutlu kalıpların yapısal özelliklerden daha uygun olduğu görülmüştür. Çoğu durumda, kullancıların sosyal çevrelerinin dışlarındaki tespitlerinde tartışmalı konuların ortaya çıktığı gözlemlenmiş ve, artırılmışsaража arasında tartışmalı konuların daha az ele alındığı anlamaktadır. Son olarak, önerilen motivler herhangi bir yanıt ağacından veya alt ağacından kolayca elde edilebileceği için, zaman içinde tartışmaların ve alt tartışmaların evrimini gözlemlemede bu tür kalıpların kullanımı denenmiştir. Gerçekten de farklı alt konularda veya dışsal olaylara (örneğin, haberlere) bağlı olarak, konu üzerindeki tartışmanın düzeyi ile birlikte tartışma konusunun da geliştiği gözlenmiştir. Tartışma konusu olmayan bir tweetin doğrudan yanıt alt ağacının yaklaşık % 7'si tartışmalı olarak tespit edilmiş (Coletto vd., 2017).


Tartışmalı Konuların Tespiti ile ilgili diğer çalışmalar Tablo 3.14’te gösterilmiştir.

<table>
<thead>
<tr>
<th>Yıllar</th>
<th>Çalışmalar</th>
</tr>
</thead>
<tbody>
<tr>
<td>2008</td>
<td>Twitter’da etkinlik özetin yeni bir sistem önerilmiştir (Zafiropoulos ve Vrana, 2008).</td>
</tr>
<tr>
<td>2009</td>
<td>Blog yorumlarında anlaşmaya ve anlaşmazlığa bulma konusunda bir çalışma sunulmuştur (Gilbert vd., 2009).</td>
</tr>
</tbody>
</table>
| 2010  | Twitter’da tartışmalı olayların tespitini için üç yeni yöntem önerilmiştir (Popescu ve Pennacchiotti, 2010).  
|      | Çelişki tespit ve yönetimi için yeni bir yaklaşım önerilmiştir(Tsytarsarau vd., 2010).   
<p>|      | Twitter’da grup polarizasyonu üzerine yeni bir çalışma sunulmuştur (Yardı ve Boyd, 2010).   |</p>
<table>
<thead>
<tr>
<th>Year</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012</td>
<td>• Twitter için konuşma grafiği yönteminin yeniden yapılandırılması önerilmiştir (Cogan vd., 2012).</td>
</tr>
<tr>
<td>2013</td>
<td>• Hem polarize hem de polarize olmayan bağlamlardan ortaya çıkan sosyal ağlar arasında bir karşılaştırma gösterilmiştir (Guerra vd., 2013). &lt;br&gt;• Analizazılık içeren konuların tartışıldığı sosyal medyanın rolü üzerinde durulmuştur (Smith vd., 2013). &lt;br&gt;• Siyasi yönetim temelli veri küümeleri ile politik yönetim sınıflandırmaları arasındaki ilişki sunulmuştur (Cohen ve Ruths, 2013).</td>
</tr>
<tr>
<td>2015</td>
<td>• Bir konunun karşı tarafıyla ilgili önyargı içeriği, güçlü tarafların belirlenmesi ve “düzenli” katılımcıların fikir yanlılıklarını çıkarmak için bir çerçeve sunulmuştur (Lu vd., 2015). &lt;br&gt;• Sosyal medya platformları hakkında tartışma çözümlemesi hakkında yeni bir çalışma yapılmıştır (Marres, 2015).</td>
</tr>
<tr>
<td>2016</td>
<td>• Polarize kullanıcıları ve tartışmalı konuları tanımlayan yeni bir yöntem önerilmiştir (Coletto vd., 2016). &lt;br&gt;• Twitter’da trend olan konuların ihtilal seviyesini tanımlayan bir yöntem incelenmiştir (Rabab’ah vd., 2016). &lt;br&gt;• Partizan tutuşmasına sosyal medya platformlarının rolü ele alınmıştır (Hong ve Kim, 2016).</td>
</tr>
<tr>
<td>2017</td>
<td>• Urduca yazılımı tweetlerde tartışmalı konuşma ve kalıpları tespit etmek için bazı sınıflandırma yöntemleri uygulanmıştır (Mustafa vd., 2017). &lt;br&gt;• Sosyal medya adları tepkilere dayanan yeni bir tartışma tespiti yöntemi önerilmiştir (Sriteja vd., 2017).</td>
</tr>
<tr>
<td>2018</td>
<td>• Twitter’ın trend konularındaki tartışmaları tespit etmek için melez bir yaklaşımsı sunulmuştur (Al-Ayyoub vd., 2018).</td>
</tr>
</tbody>
</table>

3.8. Nedensellik Tespiti

nedensellik çözümlemesi aracılığıyla, sistem tasarımının etkisi, kullanıcılarnın birbirleriyle nasıl işbirliği yaptıkları ve birbirleriyle nasıl etkileşimde bulundukları ve kullanıcı katılımını artırmak ve içerik kalitesini iyileştirmek için bireylerin birliktereme ent iyi nasıl teşvik edileceği daha iyi anlaşılabilmesidir.


piyasalarının kapalı olduğu pazar gününün duygusal durumu ile pazartesi gününün borsa özellikleri arasındaki ilişki incelendiğinde ortaya çıkmıştır. Sonuçlar, farklı coğrafi bölgelerde duygu durumunun hisse senedi piyasası özellikleri üzerinde nedensel bir etkisi olduğunu göstermiş ve dünya çapında hisse senedi piyasaları için davranışsal finansmanın önmini vurgulamıştır.


Nedensellik Tespiti ile ilgili diğer çalışmalar Tablo 3.15’de gösterilmiştir.

<table>
<thead>
<tr>
<th>Yılı</th>
<th>Çalışmanın merkezi konusu</th>
<th>Kaynak</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007</td>
<td>Blogların duyguluk bilgisi ile ürün satışlarının tahmin etme bilgisi arasındaki ilişki incelenmiştir (Liu vd., 2007).</td>
<td></td>
</tr>
<tr>
<td>2008</td>
<td>Forumdaki kullanıcıların mesajlarını çözümlenerek film içindeki trendleri ve gerçek dünya olaylarının tahmin etme üzerine bir çalışma ortaya konulmuştur (Krauss vd., 2008).</td>
<td></td>
</tr>
<tr>
<td>2009</td>
<td>Satın alınmadan önce ürünlerin duyguşlarını araştırmak isteyen tüketiciere veya markaların kamuoyu duyarsılığıını izlemek isteyen şirketlere yardımcı olabilecek Twitter mesajlarının duygularını sınıflandırmak için bir çalışma sunulmuştur (Go vd., 2009).</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td></td>
</tr>
</tbody>
</table>
| **2010** | • Twitter'da ifade edilen film duyguları ile gişe makbuzları arasındaki ilişki sunulmuştur (Asur ve Huberman, 2010).
|   | • Gelecek borsa hakkında yeni bilgilerin ilişkilendirilmesi ve genel ağ logolarından duygu (kayıgın, endişe ve korku) analizleri gösterilmiştir (Gilbert ve Karahalios, 2010). |
| **2011** | • Gelecek borsa hakkında yeni bilgilerin ilişkilendirilmesi ve genel ağ logolarından duygu (kayıgın, endişe ve korku) analizleri gösterilmiştir (Gilbert ve Karahalios, 2010). |
|   | • Konu Duyarlılığı Değişim Analizi, duygu değişikliği nedenini belirlemiştir (Jiang vd., 2011). |
| **2012** | • 2010 ABD kongre seçimleri sırasında 61 milyon Facebook kullanıcısına gönderilen siyasi seferberlik mesajlarının sonucu incelenmiştir (Bond vd., 2012). |
|   | • Twitter gönderilerini çözümlerek altın fiyat, ham petrol fiyatı, döviz kurları ve borsa göstergeleri tahmin edilmiştir (Zhang vd., 2012). |
| **2013** | • Online sosyal veri kullanılarak günlük hisse senedi fiyat hareketlerinin tahmin edilmesi incelenmiştir (Nann vd., 2013). |
| **2014** | • Twitter feed'lerinde ifade edilen duygular, belirli şirketler ve onların ürünlerini için çözümlenmiş ve duygular analizleri ve nedensel kural geliştirme yöntemleri kullanılmıştır (Smith vd., 2014). |
| **2015** | • Sosyal medyada kullanıcı tarafından sağlanan içerik verilerine dayanarak uyuşturucu ve olumsuz tepkiler için nedensel sinyalleri tanımlamak için Dernek Kurallı Madenciliği uygulanmıştır (Yang ve Yang, 2015). |
| **2016** | • Atorvastatin (kas ağrısı) ve sibutramin (kardiyovasküler AE'ler) ile ilişkili klinik olarak önemli advers olayların (AE'ler) belirlenmesi ve sosyal medya kayıtlarında, Olumsuz Olay Raporlama Sisteminine (FAERS) karşı kalıpları karşılaştırılmış ve sosyal medya kayıtlarının olup olmadığı değerlendirilmiştir için Granger nedensellik testlerinden yararlanılmıştır (Duh vd., 2016). |
| **2017** | • Araştırmada, Twitter verileri hisse senedi piyasası fiyatlarının hisselerin polaritesini (pozitif veya negatif) kullanarak tahmin etmek veya tanımlamak için kullanılabileceğini durumlarda incelenmiştir (Bernardo vd., 2017). |
| **2018** | • Sosyal medyadan olumsuz ilaç reaksiyonlarını tespit etmek için ilk nedensellik duyarlı yaklaşımlar önerilmiştir (Bollegala vd., 2018). |
|   | • Bir ağ nedensel çarşama çerçevesi kullanılarak sosyal medya ağları üzerindeki tahmini etkilemeye yönelik yeni bir yaklaşım sunulmuştur (Smith vd., 2018). |
4. BÜYÜK VERİ


Yine başka bir çalışmaya göre 2020 yılında üretilecek veri miktarı 2009 yılında üretilmiş veri miktarının tam 44 katı büyüklüğünde olacaktır (URL-16). Her saat, Wal-Mart müşterilerinin işlemlerini şirkete yaklaşık 2,5 petabayt veri sağlamakta ve her hafta sosyal ağlarda yaklaşık 300.000 defa kendisinden bahsedilmektedir (URL-17).

4.1. Büyük Veri ve Özellikleri


Şekil 4.1. Büyük veri ve 7V

- **Hacim:** Hacim, bir kuruluşun veya bir kişinin topladığı ve/veya ürettiği veri miktarını ifade eder. Büyük veri, twitter verileri, genel ağ sayfasındaki tıklamalar, ağ trafiği, ışık hızındaki verileri yakalayan sensör özellikleri ekipmanların elde ettiği yüksek hacimli ve düşük yoğunluklu değeri bilinmeyen verilerin işlenmesini gerektirir (URL 18). Şu anda en az 1
terabayt büyük veri eşiği iken, büyük veri olarak nitelendirilebilecek minimum boyut teknoloji gelişiminin bir fonksiyonu olarak ifade edilebilir. 1 terabayt veri miktarı, tahminlere göre 1.500 CD veya 220 DVD’ye sığacak yaklaşık 16 milyon Facebook fotoğrafını depolayacak bir depolama miktarını ifade etmektedir (Gandomi ve Haider, 2015).


- **Değişkenlik**: Hacimsel verilerde ham verilerin gerçek anlamları ve yorumlamaları bağlamına bağlıdır. Sosyal medyada bir kelimede farklı anlamlara gelebilir. Kelimelerin statik

- **Doğruluk:** Saklanan verilerin hatalı olabileceği ve büyük miktarda işlenmemiş veri olmasının belirsizliği ile ilgilidir. Sonuç veya çözümlemenin doğruluğu kaydedilen verilerin kesinliğine ve doğruluğuna bağlıdır (Pant ve Tanwar, 2016). Yüksek oranlarda toplanan büyük verilerle ilgili temel kararlar alınması gerektiğinde veriler ne ölçüde güvenilir olabilir? Basitçe, verilerin yeterince doğru olduğu, yanılsız yönlendirilmediğini, bozulmadığını veya beklenen bir kaynağın gelmediğini bilmek zordur. Bu büyük veri doğruluğu olarak bilinen önemli bir konudur (Gani vd., 2016).

- **Görüntüleme:** Yüzlerce özellik arasındaki sayısız ilişki ve bağımlılık ile verilerin sunumu çok önemlidir. Görüntüleme, bulguun gösterimini okunabilir ve anlaşılabilir olacak şekilde tanımlar (Pant ve Tanwar, 2016).

4.2. Büyük Veri Analiz Araçları


Hadoop projesi içinde ayrı ayrı birçok proje vardır. Bu projeler sayesinde büyük veriler kolayca işlenip çeşitli sonuçlar çıkarılabilir. Hadoop projesi şu modülleri içerir:

- **Hadoop Common:** Diğer Hadoop modüllerini destekleyen ortak araçları ifade eder [URL 36].
- **Hadoop Distributed File System (HDFS):** Uygulama verisine yüksek verimle erişim sağlayan dağıtık bir dosya sistemidir (URL 19, 2017; Shvachko vd., 2010)
- **Hadoop YARN**: İş planlanması ve küme kaynağı yönetimi için bir çerçeve sağlar (URL 19, 2017).

- **Hadoop MapReduce**: Geniş veri setlerini işlemek ve onlardan bilgi üretmek için kullanılan bir YARN tabanlı paralel işleme ve programlama modelidir (URL 19, 2017; Dean ve Ghemawat, 2010).

Diğer Hadoop projeleri:

- **Ambari**: Hadoop HDFS, Hadoop MapReduce, Hive, HCatalog, HBase, ZooKeeper, Oozie, Pig and Sqoop projeleri için destek içeren, Apache Hadoop kümlerini izlemeyi ve yönetmeyi sağlayan genel ağ tabanlı bir araçtır. Ambari ayrıca ısı haritaları ve MapReduce, Pig ve Hive gibi uygulamaları izlemek ve küme verimliliğini gösterecek tablosu sağlar (URL 19, 2017).

- **Avro**: Bir veri serilizasyon sistemidir (URL 19, 2017).

- **Cassandra**: Facebook tarafından geliştirilen Cassandra, büyük miktarlı verilerin depolanması için geliştirilmiş, açık kaynaklı, ilişkisiz, kolon odaklı, dağıtılmış bir veritabanıdır (Lakshman ve Malik, 2009).

- **Chukwa**: Büyük dağıtık sistemleri yönetmek için kullanılan bir veri toplama sistemidir (URL 19, 2017).

- **HBase**: Büyük tablolar için yazısal veri depolamayı destekleyen ölçeklenebilir, dağıtılmış bir veritabanıdır. Son zamanda geliştirdiğimiz önemli bir Apache Hadoop tabanlı proje olan Google'ın BigTable, HBase kullanılarak modellenmiştir. HBase, verilere rastgele gerçek zamanlı okuma / yazma erişimi olan HDFS dosya sisteminin üzerine kurulu, dağıtılmış, hataya dayanıklı bir ölçeklenebilir veritabanı eklmiştir (URL 19, 2017; Chang vd., 2008).

- **Hive**: Hadoop dosyaları üzerinde depolanmış büyük veri kümlerine; veri yönetme, sorgulama ve çözümleme olanağı sağlayan Hadoop'un üzerine inşa edilmiş bir veri ambarıdır. Gerçek zamanlı sorgular sunmak için tasarlanmıştır, ancak metin dosyalarını ve sıra dosyalarını destekleyebilir (Kulkarni ve Khandewal, 2014).

- **Mahout**: Ölçeklenebilir bir makine öğrenme ve veri madenciliği kütüphanesidir. Mahout'un odakladığı mevcut algoritmaların alanları şunlardır: kümeleme, sınıflandırma, veri madenciliği (sık öğe seti) ve evrimsel programlama (URL 36- Taylor, 2010)

- **Pig**: MapReduce işlerinin Hadoop kümlerinde yürütülmesi için yüksek düzeyde bir paralel mekanizma sağlar. Pig Latin olarak bilinen bir komut dosyası dili kullanır, veri akışı dili paralel bir şekilde verilerin işlenmesine yöneliktir (Kulkarni, ve Khandewal, 2014).

- **Spark**: Hadoop verileri için hızlı ve genel bir hesaplama motorudur. Spark, ETL, makine öğrenimi, aksam işleme ve grafik hesaplama gibi geniş bir uygulama yelpazesini destekleyen basit ve etkileyici bir programlama modeli sağlar (URL 19, 2017).
- **Tez**: Hadoop YARN üzerine kurulmuş genelleştirilmiş bir veri akış programlama çerçevesidir. TEZ, toplu ve interaktif kullanım durumları için verileri işlemek üzere görevlerin keyfi DAG (Directed Acyclic Graph) sini yürütme için güçlü ve esnek bir motor sağlar (URL 19, 2017; Saha vd., 2015)

- **ZooKeeper**: Dağıtık uygulamalar için yüksek performanslı koordinasyon hizmeti sağlar (URL 19, 2017).

Şekil 4.2' de birleştirilmiş bilgi yönetimi platformları'nın genel bir tablosu gösterilmektedir.

![Bilgi Yönetimi Platformları Tablosu](URL:18,2017)


Şekil 3' te görüldüğü gibi birçok kaynaktan elde edilen çeşitli veri türleri alındığında, doğrudan (gerçek zamanlı) bellek süreçlerine yazılabilir veya mesajları, dosyaları veya veritabanı işlemleri olarak diske yazabilir. Bir kez aldıktan sonra, verilerin nerelere kalacağı konusunda birden fazla seçenek vardır. Dosya sistemine, geleneksel bir RDBMS'ye veya NoSQL ve Hadoop Dağıtılmış Dosya Sistemi (HDFS) gibi dağıtılmış kümelemiş sistemlere yazılabilir. Yapılandırılmış verilerin hızlı değerlendirilmesine yönelik temel

Büyük veri yönetim araçları ve çözümlemey yöntemleri günümüzde kullanılması zorunlu hale gelen bir olgudur. Bu araçlar sayesinde daha önceden çözümlemeyen veriler çeşitli boyutları ile ele alınabilir ve yeni sonuçlar elde edilebilir.
5. ÖPTİMİZASYON

5.1. Optimizasyon nedir?


![Optimizasyon Modeli](image)


diagram

Sekil 5.1. Optimizasyon için matematiksel modeller

Eğer karar değişkenleri üzerinde hiçbir sınırlama yoksa sınırlayıcısal, en azından bir sınırlama olması durumunda sınırlayıcısal olur. Eğer problem tek bir dönem için çözülecekse statik model, birden fazla dönem göz önünde alınarak çözüleceksedinamik model kullanılır.
Modelin algoritmada işletilmesi esnasında belirli, kesin parametre veya girdiler kullanılıyor model deterministik, olasılık özelliği varsa model stokastiktir. Eğer birden fazla amaç varsa, problemler çok amaçlıdır. Eğer tüm karar değişkenleri pozitif reel (gerçel) değerler alırsa sürekli optimizasyon problemini söz konusudur. Tüm karar değişkenlerinin tamsayı değerler alması gerekiyorsa kesikli optimizasyon problemi ortaya çıkar.


5.2. Sezgisel Algoritmalar

Problemlerin çözüm uzayı tüm çözümlerin değerlendirilemeyeceği kadar büyük veya sonsuz büyüklükte olabilir. Bu durumda çözümlerin kabul edilebilir bir sürede bulunması


Basitlik ve Analiz Edilebilirlik: Karmaşık algoritmalar, esneklik ve çözüm kalitesi açısından basit algoritmalarдан daha zor çözümlenebilir. Algoritma kolayca çözümlenebilir olmalıdır.


Genel amaçlı sezgisel yöntemler, biyolojik tabanlı, fizik tabanlı, sosyal tabanlı, müzik tabanlı ve kimya tabanlı olmak üzere çeşitli gruplara ayrılmaktadır. Ayrıca bunların birleşimi olan melez yöntemler de vardır. Genetik algoritma (GA), diferansiyel gelişim algoritması, karınca koloni algoritmaları, yapay sinir ağları, arı koloni algoritmaları ve yapay bağışıklık sistemleri biyolojik tabanlı, ısıl işlem ve elektromanyetizma algoritması fizik tabanlı, tabu arama sosyal tabanlı, yapay kymyasal reaksiyon optimizasyon algoritması kimya tabanlı ve armonik algoritma müzik tabanlı algoritma ve modellerdir.

Algoritma tek bir çözümden başlayıp bunu operatörlerle ilerletiyorrsa bu noltalı yöntemler denir ve tabu arama, ısıl işlem gibi bütün yerel arama tabanlı sezgisel algoritmalar bu gruba gider. Çok noktadan yani bir popülasyon üzerinden çözümle başlanıp bu farklı noktalarla optimizasyon yapıyorsa bu noltalı ya da popülasyon tabanlı yöntemlerdir. Bunlara GA, PSO, karınca koloni algoritmaları, arı koloni algoritmaları, yapay bağışıklık sistemleri ve elektromanyetizma algoritması örnek olarak verilebilir.


Çoğunlukla sezgisel yöntemler tek bir komşuluk yapısında çalışır ve tek komşuluk yapıtı olarak sınıflandırılabilir. Bazıları da değişken komşuluk arama algoritmalarında olduğu gibi arama işlemini sistematik bir şekilde değiştirerek birden fazla yerel arama yöntemiyle diğer çözüm alanlarına ulaşmaya çalışır ve değişken komşuluk yapıtı şeklinde sınıflandırılabilir.

Algoritmalar çalışırken daha önceki durumlar ya da en iyi durumlar hatırlanıyor hafızalı, hatırlanmıyorsa hafızasız şeklinde sınıflandırılabilir. Örneğin PSO ve tabu arama hafızalı, GA hafızasızdır (Alataş, 2007).
Şekil 5.2. Sezgisel Yöntemler
6. UYGULAMA ESNASINDA KULLANILAN ALGORİTMALAR, YöNTEM VE VERİ SETLERİ


6.1. Gri Kurt Optimizasyonu Algoritması

6.1.1. Algoritmanın İİham Kaynağı

Şekil 6.1. Gri kurt hiyerarşisi (baskınlık yukarıdan aşağıya doğru azalır)


En düşük seviyedeki gri kurt omegadır. Omegaya günah keçisi rolü düşer. Omegalar, her zaman baskı olan diğer bütün kurtlara boyun eğmek zorundadır. En son sırada yemelerine izin verilir. Omega, sürüde önemli bir üye değildir gibi görünse de, sürüün omega kayına uğraması durumunda tüm sürüün iç kavgaya sürüklendiği ve sorunlar yaşadığı görülmuştur. Bunun nedeni tüm kurtların şiddetinin ve geriliminin omegalar tarafından yatıştırılmasıdır. Bu durum tüm sürüyü tatmin eder ve sürüün baskınlık yapısının
korumasını sağlar. Bazı durumlarda, omega sürüdeki bebek bakıcısı rolünü de üstlenir (Mirjalili vd., 2014).


Sosyal hiyerarşilere ek olarak, grup avı gri kurtlara ait bir diğer ilginç sosyal davranışdır. Muro ve diğerlerine göre (Muro vd., 2011) gri kurt avının ana evreleri aşağıdaki gibidir:

- Avı takip etmek, kovalamak ve yaklaşmak.
- Hareketsiz kalınca kadar avın peşinde olmak, etrafını çevirmek ve taciz etmek.
- Avayı saldırmak.

Bu adımlar Şekil 6.2’de gösterilmektedir.

Şekil 6.2. Gri kurtların avlanma davranışları: (A) Avı takip etme, kovalama ve yaklaşma. (B-D) Avın peşinde olma, etrafını çevirmek ve taciz etme. (E) Sabit konum alma ve avı saldırmak (Muro vd., 2011)
Bu çalışmada, GKO'yı tasarlamak ve optimize etmek amacıyla, bu av tekniği ve gri kurtların sosyal hiyerarşisi matematiksel olarak modellenmiştir.

### 6.1.2. Matematiksel Model ve Algoritma

Bu alt bölümde sosyal hiyerarşi, takip etme, çevrelene ve ava saldırmanın matematiksel modelleri verilmiştir. Ardından ana hatlarıyla GKO algoritması anlatılmıştır (Mirjalili vd., 2014).

#### 6.1.2.1. Sosyal Hiyerarşı

GKO'yu tasarlarken kurtların sosyal hiyerarşisini matematiksel olarak modellemek amacıyla, en uygun çözüm alfa ($\alpha$) olarak isimlendirilmiştir. Buradan hareketle, ikinci ve üçüncü en iyi çözümleri sırasıyla beta ($\beta$) ve delta ($\delta$) olarak adlandırılmıştır. Aday çözümlerin geri kalanını omega ($\omega$) olduğu varsayılmıştır. GKO algoritmasında avlanma (optimizasyon) $\alpha$, $\beta$ ve $\delta$ tarafından yönlendirilir. $\omega$ kurtlar bu üç kurdu takip eder (Mirjalili vd., 2014).

#### 6.1.2.2. Avın Çevresini Sarma

Yukarıda da belirtildiği gibi, gri kurtlar avlanma boyunca avlarının çevresini sararlar. Çevresini sarma davranışını matematiksel olarak modellemek üzere aşağıdaki formlar verilmiştir:

\[
\mathbf{D} = |\mathbf{C} \cdot \mathbf{X}_p(t) - \mathbf{X}(t)| \quad (6.1)
\]

\[
\mathbf{X}(t + 1) = \mathbf{X}_p(t) - \mathbf{A} \cdot \mathbf{D} \quad (6.2)
\]

Burada $t$ mevcut yinelemeyi, $A$ ve $C$ katsayı vektörlerini, $X_p$ avın konum vektörünü ve $X$ ise gri kurdu konum vektörünü gösterir.

$A$ ve $C$ vektörleri şu şekilde hesaplanır:

\[
\mathbf{A} = 2\mathbf{a} \cdot \mathbf{r}_1 - \mathbf{a} \quad (6.3)
\]
\( \vec{\mathcal{C}} = 2. \vec{r}_2 \) \hfill (6.4)

Burada \( a \)'nin bileşenleri, yinelemeler boyunca doğrusal olarak 2'den 0'a düşürülmüş ve \( r_1, r_2 \) ise [0,1] 'deki rastgele vektörlardır.

(6.1) ve (6.2) eşitliklerinin etkilerini görmek için, iki boyutlu konum vektörü ve olası komşularının bazıları Şek. 6.3(a)'da gösterilmektedir. Buradaki şekilde de görülebileceği gibi, \((X, Y)\) konumundaki bir gri kurt, \((X^*, Y^*)\) konumundaki avına göre kendi konumunu güncelleyebilir. \( A \) ve \( C \) vektörlerinin değeri ayarlanarak, mevcut pozisyona göre en iyi ajanın etrafındaki farklı yerlere ulaşılabilir. Örneğin, \((X^* - X, Y^* - Y)\) konumuna, \( A = (1,0) \) ve \( C = (1,1) \) ayarlarıyla ulaşılabilir. Gri bir kurdan 3 boyutlu uzayda muhtemel güncellenmiş konumları, Şekil 6.3(b)'de gösterilmektedir. Rastgele vektörler \( r_1 \) ve \( r_2 \)'nin, kurtların Şekil 6.3'te gösterilen noktalar arasındaki herhangi bir konuma ulaşmasına imkan vermesine dikkat edilmelidir (Mirjalili vd., 2014).

Böylece, gri kurdan herhangi bir rastgele konumda, avın etrafındaki alanda konumu, (6.1) ve (6.2) eşitlikleri kullanılarak güncellenebilir.

Aynı kavram, \( n \) boyutta bir arama alanına genişletilebilir ve gri kurtlar bugüne kadar elde edilen en iyi çözüm etrafında hiper-küpler (veya hiper-küreler) içinde hareket edeceklandır.

6.1.2.3. Avlanma


\[
\begin{align*}
\overline{D}_\alpha &= |\overrightarrow{C_1X_\alpha - \bar{X}}|, \\
\overline{D}_\beta &= |\overrightarrow{C_2X_\beta - \bar{X}}|, \\
\overline{D}_\delta &= |\overrightarrow{C_3X_\delta - \bar{X}}|
\end{align*}
\] \hfill (6.5)
\[
X_1^* = X_{\alpha}^* - A_1^* - (D_\alpha^*), \quad X_2^* = X_{\beta}^* - A_2^* - (D_\beta^*), \quad X_3^* = X_\delta^* - A_3^* - (D_\delta^*)
\]

\[
\dot{X}(t + 1) = \frac{X_1^* + X_2^* + X_3^*}{3}
\]

Şekil 6.3. 2D ve 3D konum vektörleri ve bir sonraki muhtemel konumları

Şekil 6.4. GKO’da konum güncelleme

6.1.2.4. Ava Saldırı (Sömürme)


GKO algoritması, şimdiye kadar önerilen operatörlerle, aramaajanlarının alfa, beta ve deltaharın konumuna göre konumlarını güncellemelerine ve ava saldırılmalarına olanak tanır. GKO algoritması bu operatörlerle yerel çözümlerde hareketsizliğe eğilimlidir. Önerilen çevreleme mekanizmasının belli ölçüde keşif özelliği gösterdiği doğrudur, ancak GKO’nun keşfi vurgulamak için daha fazla operatöre ihtiyaç vardır.

Şekil 6.5. Ava saldırı ile av arayışını karşılaştırılması
6.1.2.5. Av Arama (Keşif)


GKO algoritmasının sözde kodu Şekil 6'da verilmiştir. GKO'nun优化问题問題の理理論を理解するためには、次の点を注意する必要があります:

- Önerilen sosyal hiyerarşi, GKO'nun yineleme süreci boyunca bugüne kadar elde edilen en iyi çözümleri kaydetmesine yardımcı olur.
- Önerilen çevrelere mekanizması, çözümlerin çevresinde hiper-küre olarak daha üst boyutlara genişletilebilecek daire şeklinde bir alanı tanımlar.
- A ve C rastgele parametreleri, farklı rastgele yarıçapları olan hiper-kürelere sahip aday çözümlere yardımcı olur.
- Önerilen avlanma yöntemi, aday çözümlerin arınan konumunu tespit etmesini sağlar.
- Keşif ve sömürme, a ve A'nın uyarlabilir değerleri ile güvenve alta alınmıştır.
- a ve A parametrelerinin uyarlabilir değerleri, GKO'nun keşif ve sömürme arasında sorunsuz geçiş yapmasına izin vermektedir.
- A'nın azalmasıyla birlikte, yinelemelerin yarısı keşfe (|A| ≥ 1) ve diğer yarısı ise sömürmeye (|A| < 1) aittir.
- GKO'nun, (a ve C olmak üzere) ayarlanacak sadece iki ana parametresi vardır. Gri kurtlarının tüm yaşam döngüsünü taklit etmek için mutasyonu ve diğer evrimsel operatörleri entegre etme olasılıkları vardır. Şekil 6.6'da GKO algoritmasının sözde kodu gösterilmiştir.

**Şekil 6.6. GKO algoritmasının sözde kodu**
6.2. Balina Optimizasyonu Algoritması

6.2.1. Algoritmanın İlham Kaynağı


Şekil 6.7. Kambur balinanın beslenme modeli (Mirjalili ve Lewis., 2016)

Burada kabarcık-net besleme yönteminin sadece kambur balinalarda gözlenen eşsiz bir davranış olduğunu bahsetmek kayda değerdir. Bu çalışmada, spiral kabarcık-net besleme manevrası, optimizasyon yapmak için Balina Optimizasyon Algoritması (BOA) için matematiksel olarak modellenmiş (Mirjalili ve Lewis., 2016).

6.2.2. Matematiksel Model ve Optimizasyon Algoritması

Bu bölümde ilk olarak, çevrelenen avın matematiksel modeli, sarmal kabarcık-ağ besleme manevrası ve av arayışı yer almaktadır. BOA algoritması daha sonra teklif edilir (Mirjalili ve Lewis., 2016).

6.2.2.1. Avın Çevrelenmesi

Kambur balinalar avın yerini görebilir ve bunları çevreleyebilir. Arama alanındaki en uygun tasarımın konumu önceden bilinmediğinden, BOA algoritması mevcut en iyi aday çözümün hedef av olduğunu ya da optimum değere yakın olduğunu varsayar. En iyi arama aracı tanımlandıktan sonra, diğer arama araçları, konumlarını en iyi arama aracına doğru uyarlamaya çalışır. Bu davranışı aşağıdaki denklemlerle temsil edilir:

\[
\overline{D} = |\overline{CX^*(t)} - \overline{X(t)}| \tag{6.8}
\]
\( \mathbf{X}(t + 1) = \mathbf{X}^* - \mathbf{A}\mathbf{D} \)  \hspace{1cm} (6.9)

\( t \) mevcut yinelemeyi gösterir, \( \mathbf{A} \) ve \( \mathbf{C} \) katsayı vektörleridir, \( \mathbf{X}^* \) şu ana kadar elde edilen en iyi çözümun pozisyon vektörüdür, \( \mathbf{X} \) pozisyon vektörüdür, \( | | \) mutlak değerdir, ve "." bir elemanın elemanla çarpımıdır. Daha iyi bir çözüm varsa, \( \mathbf{X}^* \) in her yinelemede güncellenmesi gerektiyinden bahsetmek gerekir (Mirjalili ve Lewis., 2016). \( \mathbf{A} \) ve \( \mathbf{C} \) vektörleri aşağıdaki gibi hesaplanır:

\[
\mathbf{A} = 2\mathbf{a} \cdot \mathbf{r} - \mathbf{a} \hspace{1cm} (6.10)
\]

\[
\mathbf{C} = 2 \cdot \mathbf{r} \hspace{1cm} (6.11)
\]

İterasyonlar sırasında \( \mathbf{a} \) 2'den 0'a doğrusal olarak (hem keşif hem de sömürü aşamalarında) azalır ve \( \mathbf{r} \) [0,1] 'de rastgele bir vektördür.

Şekil 6.8 (a) Denklem 6.9’un ardından mantığı göstermektedir. Bir arama aracının konumunu \((X, Y)\) mevcut en iyi kaydın konumuna göre güncellenebilir \((X^*, Y^*)\). \( \mathbf{A} \) ve \( \mathbf{C} \) vektörlerinin değeri ayarlanarak mevcut konuma göre en iyi araç etrafındaki farklı yerler elde edilebilir. Bir arama aracının 3D uzayda muhtemel güncelleme pozisyonunu da Şekil 6.8 (b)’de gösterilmektedir. Belirtilmelidir ki rastgele vektörün \((\mathbf{r})\) tanımlanmasıyla, gösterilen anahtar noktalar arasında bulunan arama alanında herhangi bir konuma ulaşmak mümkündür. Bu nedenle, Denklem 6.9 herhangi bir arama aracının mevcut en iyi çözüm mahallindeki konumunu güncellemesine izin verir ve avın etrafını çevrelemeyi taklit eder (Mirjalili ve Lewis., 2016).

Aynı kavram boyutlar dâhilinde bir arama alanına genişletilebilir ve arama araçları şimdiye kadar elde edilen en iyi çözüm etrafında hiper küplerde hareket edecekdir. Bir önceki bölümde de belirtildiği gibi, kambur balinalar da avlarına kabarcık-ağ stratejisiyle saldırlar (Mirjalili ve Lewis., 2016).

6.2.2. Kabarcık-ağ Saldırı Yöntemi (sömürü aşaması)

Kambur balinaların kabarcık-ağ davranışını matematiksel olarak modellemek için iki yaklaşım tasarlanmıştır:
Daralan çevreleme mekanizması: Bu davranış Denklem'deki ($\tilde{a}$) değerinin düşürülmesiyle elde edilir. (6.10) $\tilde{A}$'in dalgalanma aralığının da azaldığını unutmayın. ($\tilde{a}$)

Diğer bir deyisle, tekrarlamalar boyunca, $a$'nin 2'den 0'a düştüğü yerde $\tilde{A}$, $[-a, a]$ rastgele bir değerdir. $[-1, 1]$'de $\tilde{A}$ için rastgele değerler belirlandığında, bir arama aracının yeni konumu, aracın orijinal konumu ile mevcut en iyi aracının konumu arasında herhangi bir yerde tanımlanabilir. Şekil 6.9'da (a) $(X, Y)'den (X^*, Y^*)'ye doğru mümkün pozisyonlardan gösterir ki bu da $0 \leq A \leq 1$ 2D alanından elde edilebilir (Mirjalili ve Lewis., 2016).

![Şekil 6.8. Arama ajanlarının 2 ve 3 boyutlu pozisyon vektörleri ve olası diğer pozisyonları](image)

![Şekil 6.9. (a) Daralan çevreleme mekanizması ve (b) Spiral hareket](image)
Sarmal güncelleme pozisyonu: Şekil 6.9. (b) ’de görüldüğü gibi, bu yaklaşım öncelikle (X, Y)’deki balinayla (X*, Y*)’da yer alan av arasındaki mesafeyi hesaplar. Balina ve avın konumu arasında kambur balinaların heliks şeklinde hareketini taklit etmek için aşağıdaki gibi spiral bir denklem oluşturulur:

\[
\ddot{X}(t + 1) = \overrightarrow{D} \cdot e^{bl} \cdot \cos(2\pi l) + \overrightarrow{X}(t)
\]

(6.12)

Bu denklem, (t) için esit olduğu ve i inci balinanın avına olan mesafesini gösterdiğiinde (bugüne kadar elde edilen en iyi çözüm), \(b\) logaritmik spiralin şeklini tanımlamak için bir sabittir, \(l\) \([-1,1]\)’de rastgele bir saydır ve elemanın elemanla çarpımıdır (Mirjalili ve Lewis., 2016).

Kambur balinaların daralan bir daire içinde ve aynı anda spiral şekilde bir yol boyunca av etrafında yüzdüğünü unutmayın. Bu eşzamanlı davranışı modelmek için, küçülme çevrimli mekanizma veya optimizasyon sırasında balinaların pozisyonunu güncellemek için螺旋 model arasında seçim yapacak % 50 olasılık olduğunu varsayıyoruz. Matematiksel modeli aşağıdaki gibidir:

\[
\ddot{X}(t + 1) = \begin{cases} 
\overrightarrow{X}(t) - \overrightarrow{A} \cdot \overrightarrow{D}, & i f \ p < 0.5 \\
\overrightarrow{D} \cdot e^{bl} \cdot \cos 2\pi l + \overrightarrow{X}, & i f \ p \geq 0.5
\end{cases}
\]

(6.13)

Burada \(p\), [0,1]’de rastgele bir saydır.

Kabarcık-ağ yönteminin yanı sıra, kambur balinalar avlarını rastgele ararlar. Aramanın matematiksel modeli aşağıdaki gibidir.

6.2.2.3. Av Arama

\(\overrightarrow{A}\) Vektörünün varyasyonuna dayanan aynı yaklaşıma avı (keşif) aramak için kullanılabilir. Aslında kambur balinalar, birbirlerinin pozisyonlarına göre rastgele arama yaparlar. Burada, arama aracını bir referans balinanın yaklaşıması zorlamaya zorlamak için \(\overrightarrow{A}\) değerini l’den büyük veya l’den daha düşük bir değerle kullanırız. Sömürü aşamasının aksine, bir arama aracının keşif aşamasındaki konumunu, şu ana kadar bulunan en iyi arama aracını yerine rastgele seçilen bir arama aracına göre güncelleyoruz. Bu mekanizma ve \(|\overrightarrow{A}| > 1\) keşifleri vurgular ve BOA
algoritmasının global bir arama yapmasına izin verir. Matematiksel modeli aşağıdaki gibidir.

\[
\bar{D} = |\vec{c} \cdot \overline{X_{\text{rand}}} - \overline{X}| \tag{6.14}
\]

\[
\vec{x}(t + 1) = \overline{X_{\text{rand}}} - \vec{A}\bar{D} \tag{6.15}
\]


Başlangıç popülasyonu ayarla \(X_i\) \((i = 1, 2, \ldots, n)\)
Her bir arama ajanının uygunluk değerini hesapla
\(X^*\)=Bilinen en iyi arama ajanı
while \(t < \) maksimum iterasyon sayısı
for (her bir arama ajan için)
    Güncelle \(a, A, C, l\) ve \(p\)
        if \((p<0.5)\)
            if \(|A|<1\)
                Denklem 6.8 ile arama ajanı konumu güncelle
            else if \(|A|<1\)
                Rastgele bir arama ajanı seç \((\overline{X_{\text{rand}}})\)
                Denklem 6.15 ile arama ajanını güncelle
            end if
        else if \((p\geq0.5)\)
            Denklem 6.12 ile arama ajanı konumunu güncelle
        end if
    end if
end for

Şekil 6.11. BOA algoritmasının sözde-kodu
Burada $x_{\text{rand}}$, mevcut popülasyondan seçilen rastgele bir konum vektördür (rastgele bir balıka). Belirli bir çözüm etrafındaki bazı muhtemel pozisyonlar Şekil 6.10’da $\hat{A} > 1$ ile tasvir edilir.

BOA algoritması bir takım rastgele çözümler kümesiyle başlar. Her bir yinelemesde, arama araçları, rastgele seçilen bir arama aracı veya şimdiye kadar elde edilen en iyi çözümlere göre konumlarını günceller. $a$ parametresi, sırasıyla, keşif ve sömürü sağlamak için 2’den 0’a düşürülebilir. Arama araçlarının konumunu güncellemek için $|\hat{A}| < 1$ olduğunda en iyi çözüm seçilirken, $|\hat{A}| > 1$ iken, bir rastgele arama aracı seçilir. P değerine bağlı olarak, BOA, yolda bir sarmal veya dairesel hareket arasında geçiş yapabilir. Son olarak, BOA algoritmasının bir sonlandırma kriterinin tatmini ile sona erdirilir.

BOA algoritmasının sözde kodu Şekil 6.11.’de sunulmuştur.

Teorik bakış açısı göre, BOA küresel bir optimizör olarak düşünülebilir çünkü arama /sömürü yeteneğini içeren. Dahası, önerilen hiper-küp mekanizması, en iyi çözümün bulunduğu mahaldeki bir arama alanı tanımlar ve diğer arama ajanlarının o alandaki mevcut en iyi kaydı kullanmasına izin verir. A arama vektörünün uyarlamalı varyasyonu, BOA algoritmasının, keşif ve sömürü arasında $A'$yi azaltarak sorunsuz geçişine izin verir, bazı tekrarlamlar keşfe adanmıştır. ($|A| \geq 1$) ve geri kalani sömürüye adanmıştır ($|A| < 1$). Dikkat çekici bir şekilde, BOA, ayarlanacak sadece iki ana iç parametreyi içerir ($A$ ve $C$). Mutasyon ve diğer evrimsel operasyonlar, kambur balinalarının davranışını tam olarak yeniden üretmek için BOA formülasyonuna dâhil edilmiş olsa da, sezgiselliğin miktarını ve iç parametrelerin sayısını minimuma indirerek böylece BOA algoritmasının çok basit bir versiyonunu uygulamıştır.

6.3. Büyük Patlama Büyük Çöküş Algoritması

Gelişigüzel, doğada, yerleşimsel bir etkileşim olarak görülülen yerel veya global bir optimum noktaya yaklaşıkça enerji yayılınca eşit olarak görülülebilir. Enerji yayılının düzenli parçacıklardan düzensizlik oluşturmasından dolayı, tümüyle yeni olan çözüm adaylarının (düzünsizlik ve kaos) doğumu için yakınsamış bir çözümün gelişigüzel bir dönüşüm kullanılır (Erol ve Eksin, 2006).

Bu metotta başlangıç popülasyonu gelişigüzel bir şekilde oluşturulur. Başlangıç popülasyonunun gelişigüzel oluşturulmasına Büyük Patlama aşaması denir. Bu aşamada,
aday çözümler tüm arama uzayına benzer bir biçimde dağıtılmıştır. Şekil 6.12’de aday çözümler, 2-bayt uzunlukwadaki değişkenlerle gerçek sayıların sembolize edildiği Rosenbrock fonksiyonunun optimizasyon probleminde, nasıl dağıtıldığına dair bir fikir vermek için oluşturulmuştur (Erol ve Eksin, 2006).

Normal gelişigüzel sayı üreticisinin, birim sayıdan daha büyük sayılar üretmesi sırasında, sayıların değerleri arama uzayında tutması için sınırlanmalıdır. Bu sınırlarının etkisi Şekil 4.15’de adayların sınırlarda birikmesi olarak görülebilir.

Bu örnekte popülasyon sayısı kalite testlerinde 30 olarak korunmuştur. Fakat bu büyüklük yakınsamaya veya iterasyon sayısına bağlı olarak azaltılabilir ya da artırılabilir (Erol ve Eksin, 2006).

Büyük Patlama safhasını Büyük Çöküş safhası izler. Büyük Çöküş, sadece kütte merkezinin hesaplanmasından türetildiginden dolayı ‘kütte’ merkezi olarak adlandırılan birçok girişi ve sadece bir çıkma olan bir yakınsama operatörüdür. Burada kütte terimi uygunluk fonksiyonunun değerinin tersine işaret eder. Kütte merkezini gösteren nokta $x^c$ ile belirtilir ve şu formülü göre hesaplanır:

$$
x^c = \frac{\sum_{i=1}^{N} x_i^i \cdot f_i}{\sum_{i=1}^{N} f_i}$$

(6.16)

$x^i, n$ boyutlu bir arama uzayın da bir noktadır. $f^i$ bu noktannın uygunluk fonksiyonu değeridir. $N$, Büyük Patlama safhasında ki popülasyon sayısıdır. Büyük Çöküş safhasındaki

Şekil 6.12. Rosenbrock Fonksiyonu için $X_1$ ve $X_2$ parametreleri ile gösterilen iki boyutlu durumda aday çözümlerın dağılımı ve onların ‘o’ ile gösterilen kütle merkezi.


K’nın k+1 iterasyon/uzay daki uzay sınırları

\[(k+1)^{\text{ninci iterasyon}>1}\]  \hspace{1cm} (6.17)
İkinci patlamadan sonra kütle merkezi yeniden hesaplanır. Bu başarılı patlama ve daralma basamakları bir durdurma kriterine rastlayana kadar tekrar eder bir şekilde korunur.

![Diagram](image)

Şekil 6.13. 4. iterasyondan sonra Rosenbrock Fonksiyonu için $X_1$ ve $X_2$ parametreleri ile gösterilen iki boyutlu durumda aday çözümlerin dağılımı ve onların ‘o’ ile gösterilen kütle merkezi.

![Diagram](image)


Normal gelişigüzel nokta üreticisine sağlanan iki parametre önceki basamağın kütle merkezi ve standart sapmasıdır. Sapma ayarlanabilir fakat tamamlanan iterasyonlarla beraber azalan değeri Şekil 6.14’de verildiği üzere iyi sonuçlar üretir. Kütle merkezi
etrafındaki birikme görülebilir fakat hala topluluğun dışında az miktardaki nokta var olur (Erol ve Eksin, 2006).


BP-BS’da ki adımları özetlersek (Erol ve Eksin, 2006):

Adım 1: Gelişigüzel bir biçimde N adaydan oluşan başlangıç jenerasyonunu oluştur. Arama uzayının sınırlarını dikkate al.

Adım 2: Tüm aday çözümlerin uygunluk fonksiyon değerlerini hesapla.

Adım 3: Denklem 6.16’ya göre kütte merkezini bul. En iyi uygun birey Denklem 6.16’ya kullanmak yerine kütte merkezi olarak seçilebilir.

Adım 4: Kütte merkezinin etrafındaki yeni adayları, iterasyonlar akıp geçtikçe değeri azalan normal gelişigüzel bir sayı ekleyerek veya çıkartarak hesapla. Bu şöyle formüle edilebilir:

\[ x^{new} = x^c + \frac{lr}{k} \]  

(6.18)

\( x^c \) kütte merkezini belirtir, \( l \) parametrenin üst sınırıdır, \( r \) normal gelişigüzel bir sayı ve \( k \) iterasyon basamağıdır. Sonra yeni nokta, \( x^{new} \) alttan ve üstten sınırlanır.

Adım 5: Durdurma kriterine ulaştana kadar Adım 2’ye dön.

6.4. Hibrit Balina Optimizasyonu-Büyük Patlama Büyük Çöküş Algoritması


Bu tez kapsamında ilk defa önerilen Hibrit Balina Optimizasyonu-Büyük Patlama Büyük Çöküş Algoritması (HBO-BPBÇ) algoritmasının akış şeması aşağıda Şekil 6.15’de verilmiştir.

![Şekil 6.15. HBO-BPBÇ algoritmasının akış diyagrama](image)

Hibrit Balina Optimizasyonu-Büyük Patlama Büyük Çöküş Algoritması (HBO-BPBÇ), BOA’nın belli sayıda ilerleyen iterasyonları boyunca en iyi uygunluğa sahip bireyinin değişmemesi durumunda en iyi uygunluğa sahip bireyi x cüktle merkezi olarak kabul eder. Daha sonra Denklem 6.18 kullanılarak bu birey etrafında büyük patlama işlemi

93

6.5. Veri Ön İşleme Adımları

Bu tez kapsamında verilerin ön işlemesi sırasında takip edilen veri ön işleme aşamaları KNIME programında aşağıdaki şekilde icra edilmiştir.

6.5.1. KNIME’da Veri Ön İşleme Aşamaları

KNIME (Konstanz Information Miner), grafik ara yüzü veri bilimi uygulamaları ve hizmetleri oluşturmak için üretilmiş açık kaynak kodlu ve kullanıcı dostu arayüz sahip bir veri madenciliği uygulama yazılımdır. Sezgisel, açık ve sürekli yeni gelişmeleri entegre edebilen KNIME, verileri anlama, veri bilim iş akışlarını tasarlama ve yeniden kullanılabilir bileşenleri içermesi nedeni ile önemli bir uygulama aracıdır.

a) File Reader

Bu düğüm bir ASCII dosyası veya URL konumundan veri okumak için kullanılabilir. Çeşitli formatları okumak için yapılandırılabilir. Düğümün yapılandırılması iletişim kutusunu açıp bir dosya adı verdiğinizde, dosya okuyucu, dosya içeriğini inceleyerek okuyucunun ayarlarını tahmin etmeye çalışır. Görüntülenen veriler doğru değilse ya bir hata bildiriliyorsa, ayarları manuel olarak ayarlayabilir.

b) Rule-based Row Filter


c) Document Creation

d) Önisleme Adımları

Tez kapsamında kullanılan veri setlerinin KNIME metin işleme uzantısı kullanılarak tasarlanan ön işleme adımları 12 basamaktan oluşmaktadır (Şekil 6.18). Söz konusu ön işleme adımları aşağıda kısaca tanıtılmıştır.

- **Punctuation Erasure**: Bu düğüm, terimlerin noktalama karakterlerini silmenizi sağlar. Önislemeden geçen terimler, çıkan DataTable'da ve bu terimleri içeren belgelerde saklanır.
- **Number Filter**: Sadece sayılardan oluşan terimlerin tümünü filtreler.
- **N Chars Filter**: Belirtilen N sayıda karakterden daha az sayıda sözcükten oluşan tüm terimleri filtreler.
- **Stop Word filter**: Belirtilen geçersiz sözcük dosyasında bulunan tüm terimleri filtreler. Geçersiz sözcükler, her satır sadece bir geçersiz sözcük içerecek şekilde, bir kolonda aralarda yazılıması gerekir.
- **Case Converter:** Bu düğüm, terimleri küçük veya büyük harfe dönüştürmenizi sağlar.

![Case Converter Diagram](image)

- **Snowball Stemmer:** Bu düğüm, Snowball kök çıkarma kütüphanesinden sağlanan kök çıkarma algoritmalarından yararlanarak terimlerin köklerini çıkarmanızı olanak tanır, ayrıntılar için bkz. (Http://snowball.tarantara.org/). İngilizce, Almanca, Fransızca vb. için olmak üzere kullanılabilir durumda çeşitli kök çıkarma algoritmaları uygulanabilir.

- **Bag of Words Creator:** Bu düğüm, bir grup belgenin sözcük torbasını (BoW) oluşturur. BoW, bir belgede iki terimleri içeren ve diğer ise ilgili belgede geçen terimleri içermek üzere en az iki kolondan oluşur. Satırlar, ilgili belgede yer alan çoklu terim demetini ifade eder.

- **Term to String:** Bu düğüm, tanımlı bir kolonun terimlerini dizelere dönüştürmenize ve bu dizeleri içeren yeni bir kolon eklemenize olanak tanır. Yalnızca terimlerin sözcükleri dönüştürülüğünden terimlerin etiketleri kaybolur.

- **TF:** Bir belgedeki terimlerin göreceli terim siklığını (tf) hesaplar ve tf değerini içeren bir kolon ekler. Değer, belgedeki bir terimin mutlak siklığı, o belgenin tüm terimlerin sayısına bölünmesiyle bulunur.
• **GroupBy:** Bir tablonun satırlarını, seçilen grup kolonlarındaki benzersiz değerler bazında gruplandırır. Seçilen grup kolonunun her benzersiz değeri için bir satır oluşturur. Kalan kolonlar, belirtilen gruplama ayarlarına göre kümelenir. Çıkış tablosu, seçilen grup kolonlarının her bir benzersiz değer kombinasyonu için birer satır içerir.

• **Row Filter:** Bu düğüm, belirli ölçütlere göre satır filtrelemeye izin verir. Belirli (satır numarasıyla belirlenen) aralıkları, belirli bir satır kimliğine sahip satırları ve seçilebilir bir sütunda belirli bir değere sahip satırları (öznitelik) eklenebilir veya dışarada bırakabilir. Düğümün yapılandırma iletişim kutusunda nasıl yapılandırma yapılacağını ilişkin adımlar aşağıda verilmiştir. Not: Düğüm, veri tablosunun etki alanı değiştirmez, yanı sınırlardan biri veya bir değer tamamenfiltrelenmiş olsa bile, üst ve alt sınırlar veya tablo özellikindeki olması değerler uyarlanmaz.

• **Reference Row Filter:** Bu düğüm, ikinci bir tabloyu referans olarak kullanarak ilk tablodaki satırların filtrelenmesini sağlar. İletişim kutusu ayarına bağlı olarak, referans tablosundaki satırlar çıktı tablosuna eklenir veya dışlanır. Ekleme/dışlama testi sırasında her iki tablonun seçilen kolonların değerleri karşılaştırılır.

e) **Document vector**

Bu düğüm, her bir belge için terimler alanında o belgeyi temsil eden bir belge vektörü oluşturur. Özellik vektörlerinin değerleri, Boole değerleri veya belirtilen bir kolonun değerleri olarak, yani bir tf * idf sütunu olarak belirtilebilir. Vektörlerin boyutu, BoW'daki birbirinden farklı terimlerin sayısı kadar olaktır.

f) **Category To Class**

g) Column Filter

Bu düğüm, yalnızca kalan kolonlar çıkış tablosuna eklenecek şekilde kolonların giriş tablosundan filtrelenmesine olanak tanır. İletişim kutusundaki kolonlar Eklenen ve Dişlanan listesinden çıkarılabilir.

h) Partitioning

Giriş tablosu, örneği eğitim ve test verileri gibi (satır bazında) iki bölüntüye ayrılır. İki çıkış portunda iki bölüm yer alır.

i) ARFF Yazıcı

Bu düğüm, bir dosyaya veya URL ile belirtilen uzak bir yere ARFF formatında veri kaydeder. Yapılandırma iletişim kutusunda, geçerli bir varış yeri belirtin. Düğüm, yürütüldüğünde giriş portundan gelen verileri belirtilden konuma yazar. Şu anda, sadece seyre olmayan ARFF dosyalarını yazar (yani, değeri sıfır olsa bile, tüm verileri her zaman yazar). Hedef konum bir uzak URL ise, tüm seçeneklerin kullanılabilir durumda olmadığına dikkat edin, çünkü genel olarak uzak konumun var olup olmadığını belirlemek mümkün değildir. Bu durumda her zaman için üzerine yazım yapılacaktır.

6.6. Tutum Veri Setleri

Tutum tespiti, bir metne bakarak, o metnin yazarının bir teklif veya hedefin lehinde mi yoksa aleyhinde mi olduğunu ya da bu teklif ya da hedefe karşı tarafırsa mi olduğunu otomatik olarak karar verme görevidir. Hedef bir kişi, bir kurum, bir hükümet politikası, bir hareket, bir ürün, vb olabilir. Örneğin, Barack Obama’nın konuşmalarına bakarak, onun ABD’de daha katı silah Yasalarının lehinde olduğu ileri sürülebilir. Benzer şekilde, insanlar çevrimiçi forumlarda, bloglarda, Twitter, Youtube, Instagram vb.de gönderiler yoluyla genellikle çeşitli hedef varlıklara karşı tutum sergilerler. Otomatik tutum tespiti, bilgi edinme, metin özetleme ve metinsel çıkarımlarda yaygın bir şekilde uygulanır. Son on yılda, çevrimiçi içi
platformlarda beyan edilen tutumların modellenmesi hakkında aktif araştırmalar yürütülmektedir.

Bu tez kapsamında dört farklı konu başlığı hakkında (Tablo 6.1) iki milyondan fazla tutum içeren tweetten üretilen eğitim ve test verilerinin bulunduğu veri tabanından (Mohammad vd., 2016) dört farklı veri setine ait bilgiler ayrı ayrı çıkarılmıştır. Daha sonra veri ön işleme teknikleri ile veriler sadeleştirilerek (sayısallaştırılmış) optimizasyon sınıflandırması için elverişli hale getirilmiştir.

**Tablo 6.1** Manuel olarak belirlenen, tutumu belirli ve tutumu belirsiz hashtag örnekları (Mohammad vd., 2016)

<table>
<thead>
<tr>
<th>Hedef</th>
<th>Örnek Lehte Hashtag</th>
<th>Örnek Aleyhte Hashtag</th>
<th>Örnek Duruşu belirsiz Hashtag</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ateizm</td>
<td>#DinlereSon</td>
<td>#Tanrınlıradesi</td>
<td>#ateizm</td>
</tr>
<tr>
<td>Feminist</td>
<td>#FeminizmGerekliÇünkü</td>
<td>#FeminizmBerbattur</td>
<td>#Feminizm</td>
</tr>
<tr>
<td>Hillary Clinton</td>
<td>#HAYDIIHARY</td>
<td>#NedenHillary'yeOyVermiyorum</td>
<td>#hillary2016</td>
</tr>
<tr>
<td>Kürtajın Yasallaştırılması</td>
<td>#SeçimHakki</td>
<td>#KürtajınSonBulmasıçinDuaE diyorum</td>
<td>#PlanlıEbeveynlik</td>
</tr>
</tbody>
</table>

### 6.6.1. Feminist Hareket Veri Seti

6.6.2. Ateizm Veri Seti

Ateizm tartışması insanlığın tarihi kadar eskidir. Bu tartışma, çevrimiçi sosyal platformlarda yaygın olarak yer alan tartışmalar arasında. Tez kapsamında Tweeter'da ateizm hakkında devam eden tartışmalardan 935.181 adet tweetden elde edilen eğitim ve test verileri kullanılmıştır (Şekil 6.20).

<table>
<thead>
<tr>
<th>Target</th>
<th>Sentiment</th>
<th>Stance</th>
<th>Train/Test</th>
<th>Tweet</th>
</tr>
</thead>
<tbody>
<tr>
<td>pos</td>
<td>AGAINST</td>
<td>Atheism</td>
<td>Test</td>
<td>Come to me, all you who are weary and burdened, and I will give ye rest. Matthew 11:28</td>
</tr>
<tr>
<td>pos</td>
<td>AGAINST</td>
<td>Atheism</td>
<td>Test</td>
<td>(2/2) the Father's active goodness and unrestricted love. - RUTHLESS</td>
</tr>
<tr>
<td>pos</td>
<td>AGAINST</td>
<td>Atheism</td>
<td>Test</td>
<td>Jer 20:11 But the #OB is with me like a mighty warrior; so my per</td>
</tr>
<tr>
<td>pos</td>
<td>AGAINST</td>
<td>Atheism</td>
<td>Test</td>
<td>What other nation is so great as to have their gods near them the v</td>
</tr>
<tr>
<td>pos</td>
<td>AGAINST</td>
<td>Atheism</td>
<td>Test</td>
<td>We have to #Pray with our eyes on #God, not on the difficulties. re</td>
</tr>
<tr>
<td>pos</td>
<td>AGAINST</td>
<td>Atheism</td>
<td>Test</td>
<td>@evrttbd this is PRECIOUS..omg go get em #Europe #Jesus #awak</td>
</tr>
<tr>
<td>pos</td>
<td>AGAINST</td>
<td>Atheism</td>
<td>Test</td>
<td>@truthway_eng Jesus commands you to follow Acts 2:38-39 to be</td>
</tr>
<tr>
<td>pos</td>
<td>AGAINST</td>
<td>Atheism</td>
<td>Test</td>
<td>Bethany Hamilton Reveals She is Resting in God’s Promises Despite 1</td>
</tr>
<tr>
<td>pos</td>
<td>AGAINST</td>
<td>Atheism</td>
<td>Test</td>
<td>Husbands, love your wives, as Christ loved the church and gave him</td>
</tr>
<tr>
<td>pos</td>
<td>AGAINST</td>
<td>Atheism</td>
<td>Test</td>
<td>When we have done all in our power that the Lord will find a way to</td>
</tr>
<tr>
<td>pos</td>
<td>AGAINST</td>
<td>Atheism</td>
<td>Test</td>
<td>The #humanist church is the new doth. #Christian #test #Se</td>
</tr>
<tr>
<td>pos</td>
<td>AGAINST</td>
<td>Atheism</td>
<td>Test</td>
<td>I pray for the best for everybody!!! #SenSt</td>
</tr>
<tr>
<td>pos</td>
<td>AGAINST</td>
<td>Atheism</td>
<td>Test</td>
<td>The Irish national school system is secular under law. We can reaeffr</td>
</tr>
<tr>
<td>pos</td>
<td>AGAINST</td>
<td>Atheism</td>
<td>Test</td>
<td>Proud to be #TeamJinnah :) No Blutto, no Imran’s, only Jinnah’s Pi</td>
</tr>
<tr>
<td>pos</td>
<td>AGAINST</td>
<td>Atheism</td>
<td>Test</td>
<td>Its really incredible how much this world needs the work of missiona</td>
</tr>
<tr>
<td>pos</td>
<td>AGAINST</td>
<td>Atheism</td>
<td>Test</td>
<td>If the governments of all countries were secular, the world would be</td>
</tr>
<tr>
<td>pos</td>
<td>Favor</td>
<td>Atheism</td>
<td>Test</td>
<td>@MuhammadAli except for your belief in Allah, Gods are imaginary.</td>
</tr>
<tr>
<td>pos</td>
<td>Favor</td>
<td>Atheism</td>
<td>Test</td>
<td>Where will you be when @NeuralHorizons2015 flies by Pluto? #ML</td>
</tr>
</tbody>
</table>

Şekil 6.20. Ateizm tweetlerinin eğitim ve test verileri
6.6.3. Hillary Clinton Veri Seti


![Hillary Clinton hakkındaki tweetlerin eğitim ve test verileri](image)

6.6.4. Kürtaj Veri Seti

Şekil 6.22. Kürtajın yasallaştırılması hakkındaki tweetlerin eğitim ve test verileri

6.7. Kodlama Aşaması

KINME’dan elde edilen doküman matrisinin boyutu popülasyon bireylerinin oluşturulmasında kullanılan öncül kriterdir. Her vektör ayrı bireyleri temsil eder ve bu vektörler GKO, BOA ve HBO-BPBC için 0-1 aralığından rasgele değerlerden oluşmuştur. Popülasyon sayısının \( N \) ön işlemeden elde edilen doküman matrisindeki kelime sayısının \( M \) olduğu bir popülasyon aşağıdaki matris formunda gösterebilir.

\[
X_{N \times M} = \begin{bmatrix}
X_{11} & X_{12} & X_{1M} \\
X_{21} & X_{22} & X_{2M} \\
X_{(N-1)1} & X_{(N-1)2} & X_{(N-1)M} \\
X_{N1} & X_{N2} & X_{NM}
\end{bmatrix}
\]

6.7.1. Algoritmanın Başlangıç Popülasyonu ve Sonlandırma Şartı Değerleri

Üretilen başlangıç popülasyonu 0 ve 1 arasındaki değerlerinden rasgele oluşturulmaktadır. Popülasyon boyutu ve iterasyon sayısı standart olarak 100 alınmıştır.

6.7.2. Jaccard Benzerliği ve Uygunluk Fonksiyonu

Bu tez kapsamında metinler arasındaki ilişkinin ölçümlesi amacıyla geliştirilen Jaccard benzerliğinden faydalanılmıştır. Bu benzerlikte temel olarak iki metin üzerinden özellik
belirlemesi yapıldıktan sonra, ortak olan özelliklerin sayısı, metinlerdeki toplam özellik sayısına oranlanarak elde edilir. Bu işlem aşağıdaki gibi ifade edilebilir.

\[ X_i = X_{i1}, X_{i2}, ..., X_{iM} \] popülasyondaki \( i \). birey, \( D_j = D_{j1}, D_{j2}, ..., D_{jM} \) ise doküman matrisindeki \( j \). yorum satırına karşılık gelen vektör değerleri olarak kabul edilerek, dökümanlar arasındaki Jaccard benzerliği Denklem 6.19 ile ifade edilir.

\[
\text{jaccard\_de\_geri}_{ij} = \frac{|X_i \cap D_j|}{|X_i \cup D_j|} \quad (6.19)
\]

Bununla beraber Denklem 6.9’a tweet yorumunda söz edilen ve işleme tabii tutulan kelimelerin sayısının, doküman matrisindeki kelimelerin toplam sayısına oranı eklenmiştir (Denklem 6.20).

\[
\text{oran}_j = \frac{j\text{yorumdaki ketime sayısı}}{\text{dokümandaki ketime sayısı}} \quad (6.20)
\]

Sonuç olarak Denklem 6.21 ile gösterilen yeni Jaccard değeri üretilmiştir.

\[
\text{benzerlik}_{ij} = k_1 \times \text{jaccard\_de\_geri}_{ij} + k_2 \times \text{oran}_j \quad (6.21)
\]

Denklem 6.21’de \( k_1 \) ve \( k_2 \) toplamları bire eşit olan katsayıları temsil eder.

Popülasyondaki \( X_i \) ile temsil edilen bireylerin veri setinde kayıtlı olan tüm tweetler ile olan benzerliği değerlendirilmiştir. Benzerlik değerinin belirlenen değerden yüksek ya da düşük olmasına göre tweetlerin sınıflaması yapılmıştır.


\[
\text{Uygunluk fonksiyonu} = c_1 \times \frac{DP}{DP+YN} \times \frac{DN}{DP+YP} + c_2 \times \frac{DP}{DP+YP} + c_3 \times \frac{DP}{DP+YN} + c_4 \times \frac{DP}{\text{length (veriset)}} \quad (6.22)
\]

Denklem 6.22’de

DP: Doğru pozitiflerdir ve kuralla aynı sınıf etiketine sahip olan, kural tarafından kapsanan örneklerin sayısıdır.

YP: Yanlış pozitiflerdir ve kuraldan farklı sınıf etiketine sahip olan, kural tarafından kapsanmayan örneklerin sayısıdır.

YN: Yanlış negatiflerdir ve kural tarafından kapsanmayan fakat kuralla aynı sınıf etiketine sahip örneklerin sayısıdır.
DN: Doğru negatiflerdir ve kural tarafından kapsanmayan ve kuralla da aynı etikete sahip olmayan örneklerin sayısıdır. 

$c_1$, $c_2$, $c_3$ ve $c_4$ değerleri ise ağırlıklar ve bu dört değerin toplamı bire eşit olmaktadır. Kullanıcı tanımlıdır ve en iyi sonucu verecek şekilde belirlenebilmektedir.

6.7.3. Hesapsal Yük

Tez kapsamında kullanılan algoritmaların hesapsal karmaşıklıkları uygunluk fonksiyonunun çağrılma sayısı, popülasyon sayısı, veri setlerinde bulunan verilerin boyutu ve veri tabanındaki kayıt sayısı ile orantılı olarak artmaktadır. Algoritmalar için genel bir karmaşıklık tanımlamak gerekirse; bu uygunluk fonksiyonun çağrılma sayısı, popülasyon sayısı, veri setlerinde bulunan verilerin boyutu ve veri tabanındaki kayıt sayısının çarpımından oluşur.
7. UYGULAMA SONUÇLARI

Bu tez çalışmasında bir çevrimiçi sosyal ağ analizi problemi olan Tutum Tespit’i probleminin sınıflandırılmasında ilk defa bir çözüm yöntemi olarak GKO, BOA ve ilk defa bu tez kapsamında yeni bir hibrit algoritma olarak önerilen HBO-BPBÇ sezgisel optimizasyon algoritmaları kullanılmıştır. Bu algoritmalar dört farklı sosyal konudan elde edilen tutum Tweet veri setleri üzerinde uygulanmıştır. İlk olarak bu algoritmalarдан elde edilen sonuçlar ile bu veri setlerinin WEKA yazılımı içerisinde yer alan 32 ayrı sınıflandırma algoritmalarının en iyi sonuc veren parametreleri baz alınarak elde edilen sonuçları sınıflandırma ölçütleri kullanılarak karşılaştırılmıştır. Daha sonraki bölümde ise algoritmaların 20 defa çalıştırılması sonucunda elde edilen ortalama uygunluk değerlerinin ortalama değerlerini temsil eden grafik sonuçları gösterilmiştir.

Sonuçlar karşılaştırılırken aşağıdaki sınıflandırma ölçütleri baz alınmıştır:

<table>
<thead>
<tr>
<th>Ölçüt</th>
<th>Açıklaması</th>
</tr>
</thead>
<tbody>
<tr>
<td>Doğru Etiketlenen Veri Sayısı/ Toplam Veri Sayısı</td>
<td>( \frac{DP + DN}{DP + YP + DN + YN} )</td>
</tr>
<tr>
<td>Doğruluk</td>
<td>( \frac{DP + DN}{DP + YP + DN + YN} )</td>
</tr>
<tr>
<td>Kesinlik</td>
<td>( \frac{DP}{DP + YP} )</td>
</tr>
<tr>
<td>Hassasiyet</td>
<td>( \frac{DP}{DP + YN} )</td>
</tr>
<tr>
<td>F-Ölçütü</td>
<td>( \frac{2 * DP}{2 * DP + YP + YN} )</td>
</tr>
</tbody>
</table>

Tez kapsamında kullanılan veri setlerine ait değerlendirirmeler aşağıda ayrı başlıklar olarak ele alınmıştır.
7.1. Parametre Test Sonuçları

Bu bölümde sırası ile GKO, BOA ve BOA-BPBÇ algoritmalarının popülasyon sayısı, iterasyon sayısı veya algoritma içerisindeki sabit değerler gibi parametre değerlerinin çeşitli aralıklarla değiştirilmesi sonucu elde edilen sonuçlar her algoritma ve uygulandığı veri seti bazında verilmiştir.

7.1.1. Feminist Hareket Veri Seti’nden Elde Edilen Sonuçlar

Bu alt bölümde GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile popülasyon sayıları 100 ve 200 olarak seçilme üzere sabit tutularak, iterasyon sayıları 100, 150 ve 200 olarak sırası ile arttırılmış ve böylece popülasyon sayısının artması ve buna paralel olarak iterasyon sayısının arttırılması sonucunda elde edilen sonuçlar gösterilmiştir. Ayrıca en iyi sonucun elde edildiği popülasyon ve iterasyon sonuçları sabit tutularak diğer parametre değerlerinin başarı sonuçuna etkisi test edilmiştir. Son olarak algoritmalar uyguluk fonksiyonunun eşit sayıda çağrılma sayısına göre karşılaştırılmıştır. Tüm bu sonuçlar Feminist Hareket veri setinin WEKA programı içerisinde bulunan 32 farklı sınıflandırma algoritması ile değerlendirilmesi ile elde edilen sonuçlarla karşılaştırılmıştır. Aşağıda WEKA programından elde edilen sınıflandırma sonuçları Tablo 7.2’de gösterilmiştir. Ayrıca GKO, BOA ve HBO-BPBÇ algoritmalarında kullanılan benzerlik oranı ve uyguluk fonksiyonunda kullanılan katsayı değerleri Tablo 7.3’te gösterilmiştir.

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandırılan Örnek Sayısı</th>
<th>Doğruluk Yüzdeleri (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Bagging</td>
<td>163</td>
<td>57,193</td>
<td>0,540</td>
<td>0,572</td>
<td>0,547</td>
</tr>
<tr>
<td>2 LibSVM</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>3 QDA</td>
<td>162</td>
<td>56,8421</td>
<td>0,525</td>
<td>0,568</td>
<td>0,521</td>
</tr>
<tr>
<td>4 Stacking</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>5 MultiScheme</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>6 ClojureClassifier</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>7 REPTree</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>8 Vote</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>9 CSForest</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>10 InputMappedClassifier</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>11 MultilayerPerceptronCS</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>12 MODLEM</td>
<td>159</td>
<td>55,7895</td>
<td>0,535</td>
<td>0,558</td>
<td>0,528</td>
</tr>
<tr>
<td>13 IterativeClassifierOptimizer</td>
<td>158</td>
<td>55,4386</td>
<td>0,431</td>
<td>0,554</td>
<td>0,453</td>
</tr>
<tr>
<td>14 HyperPipes</td>
<td>158</td>
<td>55,4386</td>
<td>0,506</td>
<td>0,554</td>
<td>0,461</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>----</td>
<td>------</td>
<td>----</td>
<td>----</td>
<td>----</td>
</tr>
<tr>
<td></td>
<td>Algoritma</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Benzerlik</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Uygunluk Fonksiyonu</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>NaiveBayes</td>
<td>157</td>
<td>55,0877</td>
<td>0,602</td>
<td>0,551</td>
</tr>
<tr>
<td>16</td>
<td>DecisionTable</td>
<td>156</td>
<td>54,7368</td>
<td>0,407</td>
<td>0,547</td>
</tr>
<tr>
<td>17</td>
<td>WISARD</td>
<td>155</td>
<td>54,386</td>
<td>0,500</td>
<td>0,544</td>
</tr>
<tr>
<td>18</td>
<td>SimpleLogistic</td>
<td>155</td>
<td>54,386</td>
<td>0,494</td>
<td>0,544</td>
</tr>
<tr>
<td>19</td>
<td>SMO</td>
<td>155</td>
<td>54,386</td>
<td>0,523</td>
<td>0,544</td>
</tr>
<tr>
<td>20</td>
<td>OLM</td>
<td>155</td>
<td>54,386</td>
<td>?</td>
<td>0,544</td>
</tr>
<tr>
<td>21</td>
<td>RandomlibKnn</td>
<td>154</td>
<td>54,0351</td>
<td>0,479</td>
<td>0,540</td>
</tr>
<tr>
<td>22</td>
<td>RandomForest</td>
<td>152</td>
<td>53,3333</td>
<td>0,516</td>
<td>0,533</td>
</tr>
<tr>
<td>23</td>
<td>IBk</td>
<td>148</td>
<td>51,9298</td>
<td>0,484</td>
<td>0,519</td>
</tr>
<tr>
<td>24</td>
<td>IB1</td>
<td>147</td>
<td>51,5789</td>
<td>0,461</td>
<td>0,516</td>
</tr>
<tr>
<td>25</td>
<td>RandomTree</td>
<td>147</td>
<td>51,5789</td>
<td>0,507</td>
<td>0,516</td>
</tr>
<tr>
<td>26</td>
<td>OrdinalClassClassifier</td>
<td>142</td>
<td>49,8246</td>
<td>?</td>
<td>0,498</td>
</tr>
<tr>
<td>27</td>
<td>LDA</td>
<td>139</td>
<td>48,7719</td>
<td>0,497</td>
<td>0,488</td>
</tr>
<tr>
<td>28</td>
<td>RBFNetwork</td>
<td>137</td>
<td>48,0702</td>
<td>0,408</td>
<td>0,481</td>
</tr>
<tr>
<td>29</td>
<td>ExtraTree</td>
<td>130</td>
<td>45,614</td>
<td>0,468</td>
<td>0,456</td>
</tr>
<tr>
<td>30</td>
<td>Logistic</td>
<td>122</td>
<td>42,807</td>
<td>0,464</td>
<td>0,428</td>
</tr>
<tr>
<td>31</td>
<td>MultiClassClassifier</td>
<td>121</td>
<td>42,4561</td>
<td>0,452</td>
<td>0,425</td>
</tr>
<tr>
<td>32</td>
<td>VFI</td>
<td>62</td>
<td>21,7544</td>
<td>?</td>
<td>0,218</td>
</tr>
</tbody>
</table>

Tablo 7.3. Feminizm veri seti için kullanılan katsayılar

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Benzerlik</th>
<th>Uygunluk Fonksiyonu</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>benzerlik</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>k₁</td>
</tr>
<tr>
<td>GKO</td>
<td>0,2</td>
<td>0,9</td>
</tr>
<tr>
<td>BOA</td>
<td>0,1</td>
<td>0,9</td>
</tr>
<tr>
<td>HBO-BPBÇ</td>
<td>0,1</td>
<td>0,9</td>
</tr>
</tbody>
</table>

A. GKO, BOA ve HBO-BPBÇ Algoritmalarının 100 Popülasyon 100 İterasyon Parametreleri ile Feminist Hareket Veri Seti’ne Uygulanmasından Elde Edilen Sonuçlar

GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 100 popülasyon ve 100 iterasyon parametreleri ile feminist hareket veri setine uygulanması sonucunda test verilerinden başarılı sınıflandırma sonuçları elde edilmiştir. Tablo 7.4’de GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar WEKA’da yer alan algoritmalar ile elde edilen sonuçlar ile karşılaştırıldığında HBO-BPBÇ %59,2982 doğru sınıflandırma oranı ile birinci, BOA %58,9473 ile ikinci ve GKO ise %58,2456 ile üçüncü sıraday yer almıştır. Şekil 7.1’de algoritmaların çalışmaları sırasında 100 iterasyon boyunca uygunluk değerinin değişimi gösterilmiştir.
Şekil 7.1. Feminist Hareket Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 100 iterasyon parametrelerindeki uygunluk değeri değişimi

<p>| Tablo 7.4. GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 100 iterasyon parametreleri ile feminist hareket veri setine uygulanması sonucunda elde edilen başarı sırası. |
|---|---|---|---|---|---|
| Algoritma | Doğru Sınıflandırılan Örnek Sayısı | Doğruluk Yüzdesi (%) | Kesinlik | Hassasiyet | F-Ölçütü |
| 1 HBO-BPBÇ | 169 | 59,2982 | 0,520 | 0,518 | 0,497 |
| 2 BOA | 168 | 58,9473 | 0,527 | 0,551 | 0,508 |
| 3 GKO | 166 | 58,2456 | 0,489 | 0,476 | 0,450 |
| 4 Bagging | 163 | 57,193 | 0,540 | 0,572 | 0,547 |
| 5 LibSVM | 162 | 56,8421 | 0,518 | 0,568 | 0,518 |
| 6 QDA | 162 | 56,8421 | 0,525 | 0,568 | 0,521 |
| 7 Stacking | 162 | 56,8421 | 0,518 | 0,568 | ? |
| 8 MultiScheme | 162 | 56,8421 | 0,568 | 0,568 | ? |
| 9 ClojureClassifier | 162 | 56,8421 | 0,568 | 0,568 | ? |
| 10 REPTree | 162 | 56,8421 | 0,568 | 0,568 | ? |
| 11 Vote | 162 | 56,8421 | 0,568 | 0,568 | ? |
| 12 CSForest | 162 | 56,8421 | 0,568 | 0,568 | ? |
| 13 InputMappedClassifier | 162 | 56,8421 | 0,568 | 0,568 | ? |
| 14 MultilayerPerceptronCS | 162 | 56,8421 | 0,568 | 0,568 | ? |
| 15 MODLEM | 159 | 55,7895 | 0,535 | 0,558 | 0,528 |
| 16 IterativeClassifierOptimizer | 158 | 55,4386 | 0,431 | 0,554 | 0,453 |
| 17 HyperPipes | 158 | 55,4386 | 0,506 | 0,554 | 0,461 |
| 18 NaiveBayes | 157 | 55,0877 | 0,602 | 0,551 | 0,564 |
| 19 DecisionTable | 156 | 54,7368 | 0,407 | 0,547 | 0,434 |
| 20 WISARD | 155 | 54,386 | 0,500 | 0,544 | 0,498 |
| 21 SimpleLogistic | 155 | 54,386 | 0,494 | 0,544 | 0,503 |
| 22 SMO | 155 | 54,386 | 0,523 | 0,544 | 0,531 |
| 23 OLM | 155 | 54,386 | 0,544 | 0,544 | ? |</p>
<table>
<thead>
<tr>
<th></th>
<th>Algoritma</th>
<th>100 Popülasyon</th>
<th>İterasyon</th>
<th>100 Popülasyon</th>
<th>İterasyon</th>
<th>100 Popülasyon</th>
<th>İterasyon</th>
</tr>
</thead>
<tbody>
<tr>
<td>24</td>
<td>RseslibKnn</td>
<td>54,0351</td>
<td>0,479</td>
<td>54,0351</td>
<td>0,479</td>
<td>54,0351</td>
<td>0,479</td>
</tr>
<tr>
<td>25</td>
<td>RandomForest</td>
<td>53,3333</td>
<td>0,516</td>
<td>53,3333</td>
<td>0,516</td>
<td>53,3333</td>
<td>0,516</td>
</tr>
<tr>
<td>26</td>
<td>IBk</td>
<td>51,9298</td>
<td>0,484</td>
<td>51,9298</td>
<td>0,484</td>
<td>51,9298</td>
<td>0,484</td>
</tr>
<tr>
<td>27</td>
<td>IB1</td>
<td>51,5789</td>
<td>0,461</td>
<td>51,5789</td>
<td>0,461</td>
<td>51,5789</td>
<td>0,461</td>
</tr>
<tr>
<td>28</td>
<td>RandomForest</td>
<td>51,5789</td>
<td>0,507</td>
<td>51,5789</td>
<td>0,507</td>
<td>51,5789</td>
<td>0,507</td>
</tr>
<tr>
<td>29</td>
<td>OrdinalClassClassifier</td>
<td>49,8246</td>
<td>?</td>
<td>49,8246</td>
<td>?</td>
<td>49,8246</td>
<td>?</td>
</tr>
<tr>
<td>30</td>
<td>LDA</td>
<td>48,7719</td>
<td>0,497</td>
<td>48,7719</td>
<td>0,497</td>
<td>48,7719</td>
<td>0,497</td>
</tr>
<tr>
<td>31</td>
<td>RBFNetwork</td>
<td>48,0702</td>
<td>0,408</td>
<td>48,0702</td>
<td>0,408</td>
<td>48,0702</td>
<td>0,408</td>
</tr>
<tr>
<td>32</td>
<td>ExtraTree</td>
<td>45,614</td>
<td>0,468</td>
<td>45,614</td>
<td>0,468</td>
<td>45,614</td>
<td>0,468</td>
</tr>
<tr>
<td>33</td>
<td>Logistic</td>
<td>42,807</td>
<td>0,464</td>
<td>42,807</td>
<td>0,464</td>
<td>42,807</td>
<td>0,464</td>
</tr>
<tr>
<td>34</td>
<td>MultiClassClassifier</td>
<td>42,4561</td>
<td>0,452</td>
<td>42,4561</td>
<td>0,452</td>
<td>42,4561</td>
<td>0,452</td>
</tr>
<tr>
<td>35</td>
<td>VFI</td>
<td>21,7544</td>
<td>?</td>
<td>21,7544</td>
<td>?</td>
<td>21,7544</td>
<td>?</td>
</tr>
</tbody>
</table>

**B. GKO, BOA ve HBO-BPBÇ Algoritmalarının 100 Popülasyon 150 İterasyon Parametreleri ile Feminist Hareket Veri Seti’ne Uygulanmasından Elde Edilen Sonuçlar**

GKO, BOA ve HBO-BPBÇ algoritmalarının sıra ile 100 popülasyon ve 150 iterasyon parametreleri ile feminist hareket veri setine uygulanması sonucunda test verilerinden başarılı sınıflandırma sonuçları elde edilmiştir. Tablo 7.5’de GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar WEKA’da yer alan algoritmalar ile elde edilen sonuçlar ile karşılaştırıldığında GKO %58,5964 doğru sınıflandırma oranı ile birinci, HBO-BPBÇ %58,2456 ile ikinci ve BOA ise %56,8421 ile üçüncü sıradı yer almıştır. Şekil 7.2’de algoritmaların çalışmaları sırasında 150 iterasyon boyunca uygunluk değerinin değişimi gösterilmiştir.
Şekil 7.2. Feminist Hareket Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 150 iterasyon parametlerindeki uygunluk değeri değişimi

Tablo 7.5. GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 150 iterasyon parametreleri ile feminist hareket veri setine uygulanması sonucunda elde edilen başarı sırası.
C. GKO, BOA ve HBO-BPBÇ Algoritmalarının 100 Popülasyon 200 İterasyon Parametreleri ile Feminist Hareket Veri Seti’ne Uygulanmasından Elde Edilen Sonuçlar

GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 100 Popülasyon ve 200 iterasyon parametreleri ile feministizm veri setine uygulanması sonucunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.6’da GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar WEKA’da yer alan algoritmalar ile elde edilen sonuçlar ile karşılaştırıldığında BOA %62,8070 doğru sınıflandırma oranı ile birinci, HBO-BPBÇ %60 ile ikinci ve GKO ise %59,2982 ile üçüncü sıralarda yer almıştır. Ayrıca BOA’nın standart sapması 6,39, HBO-BPBÇ’nin 6,20 ve GKO’nun ise 6,54 olmuştur. Şekil 7.7’te algoritmaların çalışmaları sırasında 200 iterasyon boyunca uygunluk değerinin değişimi gösterilmiştir.

<table>
<thead>
<tr>
<th></th>
<th>Algoritma</th>
<th>Popülasyon</th>
<th>Sınıflandırma</th>
<th>Standard Sapması</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>OLM</td>
<td>155</td>
<td>?</td>
<td>0,544</td>
</tr>
<tr>
<td>24</td>
<td>RseslibKnn</td>
<td>154</td>
<td>54,0351</td>
<td>0,479</td>
</tr>
<tr>
<td>25</td>
<td>RandomForest</td>
<td>152</td>
<td>53,3333</td>
<td>0,516</td>
</tr>
<tr>
<td>26</td>
<td>IBk</td>
<td>148</td>
<td>51,9298</td>
<td>0,484</td>
</tr>
<tr>
<td>27</td>
<td>IBI</td>
<td>147</td>
<td>51,5789</td>
<td>0,461</td>
</tr>
<tr>
<td>28</td>
<td>RandomTree</td>
<td>147</td>
<td>51,5789</td>
<td>0,507</td>
</tr>
<tr>
<td>29</td>
<td>OrdinalClassClassifier</td>
<td>142</td>
<td>49,8246</td>
<td>0,498</td>
</tr>
<tr>
<td>30</td>
<td>LDA</td>
<td>139</td>
<td>48,7719</td>
<td>0,497</td>
</tr>
<tr>
<td>31</td>
<td>RBFNetwork</td>
<td>137</td>
<td>48,0702</td>
<td>0,408</td>
</tr>
<tr>
<td>32</td>
<td>ExtraTree</td>
<td>130</td>
<td>45,614</td>
<td>0,468</td>
</tr>
<tr>
<td>33</td>
<td>Logistic</td>
<td>122</td>
<td>42,807</td>
<td>0,464</td>
</tr>
<tr>
<td>34</td>
<td>MultiClassClassifier</td>
<td>121</td>
<td>42,4561</td>
<td>0,452</td>
</tr>
<tr>
<td>35</td>
<td>VFI</td>
<td>62</td>
<td>21,7544</td>
<td>0,218</td>
</tr>
</tbody>
</table>
Şekil 7.3. Feminist Hareket Veri Seti’ni kullanan GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon ve 200 iterasyon parametrelerindeki uygunluk değeri değişimi

Tablo 7.6. GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 200 iterasyon parametreleri ile feminist hareket veri setine uygulanması sonucunda elde edilen başarı sırası.

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandırılan Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 BOA</td>
<td>179</td>
<td>62,807</td>
<td>0,565</td>
<td>0,569</td>
<td>0,555</td>
</tr>
<tr>
<td>2 HBO-BPBÇ</td>
<td>171</td>
<td>60,000</td>
<td>0,506</td>
<td>0,529</td>
<td>0,491</td>
</tr>
<tr>
<td>3 GKO</td>
<td>169</td>
<td>59,298</td>
<td>0,549</td>
<td>0,589</td>
<td>0,542</td>
</tr>
<tr>
<td>4 Bagging</td>
<td>163</td>
<td>57,193</td>
<td>0,540</td>
<td>0,572</td>
<td>0,547</td>
</tr>
<tr>
<td>5 LibSVM</td>
<td>162</td>
<td>56,842</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>6 QDA</td>
<td>162</td>
<td>56,842</td>
<td>0,525</td>
<td>0,568</td>
<td>0,521</td>
</tr>
<tr>
<td>7 Stacking</td>
<td>162</td>
<td>56,842</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>8 MultiScheme</td>
<td>162</td>
<td>56,842</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>9 ClojureClassifier</td>
<td>162</td>
<td>56,842</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>10 REPTree</td>
<td>162</td>
<td>56,842</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>11 Vote</td>
<td>162</td>
<td>56,842</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>12 CSForest</td>
<td>162</td>
<td>56,842</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>13 InputMappedClassifier</td>
<td>162</td>
<td>56,842</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>14 MultilayerPerceptronCS</td>
<td>162</td>
<td>56,842</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>15 MODLEM</td>
<td>159</td>
<td>55,789</td>
<td>0,535</td>
<td>0,558</td>
<td>0,528</td>
</tr>
<tr>
<td>16 IterativeClassifierOptimize</td>
<td>158</td>
<td>55,438</td>
<td>0,431</td>
<td>0,554</td>
<td>0,453</td>
</tr>
<tr>
<td>17 HyperPipes</td>
<td>158</td>
<td>55,438</td>
<td>0,506</td>
<td>0,554</td>
<td>0,461</td>
</tr>
<tr>
<td>18 NaiveBayes</td>
<td>157</td>
<td>55,087</td>
<td>0,602</td>
<td>0,551</td>
<td>0,564</td>
</tr>
<tr>
<td>19 DecisionTable</td>
<td>156</td>
<td>54,736</td>
<td>0,407</td>
<td>0,547</td>
<td>0,434</td>
</tr>
<tr>
<td>20 WISARD</td>
<td>155</td>
<td>54,386</td>
<td>0,500</td>
<td>0,544</td>
<td>0,498</td>
</tr>
<tr>
<td>21 SimpleLogistic</td>
<td>155</td>
<td>54,386</td>
<td>0,494</td>
<td>0,544</td>
<td>0,503</td>
</tr>
<tr>
<td>22 SMO</td>
<td>155</td>
<td>54,386</td>
<td>0,523</td>
<td>0,544</td>
<td>0,531</td>
</tr>
<tr>
<td>23 OLM</td>
<td>155</td>
<td>54,386</td>
<td>?</td>
<td>0,544</td>
<td>?</td>
</tr>
<tr>
<td>24 RseslibKnn</td>
<td>154</td>
<td>54,035</td>
<td>0,479</td>
<td>0,540</td>
<td>0,489</td>
</tr>
<tr>
<td>25 RandomForest</td>
<td>152</td>
<td>53,333</td>
<td>0,516</td>
<td>0,533</td>
<td>0,522</td>
</tr>
<tr>
<td>26 IBk</td>
<td>148</td>
<td>51,929</td>
<td>0,484</td>
<td>0,519</td>
<td>0,494</td>
</tr>
<tr>
<td>27 IB1</td>
<td>147</td>
<td>51,578</td>
<td>0,461</td>
<td>0,516</td>
<td>0,472</td>
</tr>
<tr>
<td>28 RandomForest</td>
<td>147</td>
<td>51,578</td>
<td>0,507</td>
<td>0,516</td>
<td>0,511</td>
</tr>
<tr>
<td>29 OrdinalClassClassifier</td>
<td>142</td>
<td>49,824</td>
<td>?</td>
<td>0,498</td>
<td>?</td>
</tr>
<tr>
<td>30 LDA</td>
<td>139</td>
<td>48,771</td>
<td>0,497</td>
<td>0,488</td>
<td>0,492</td>
</tr>
<tr>
<td>31 RBFNetwork</td>
<td>137</td>
<td>48,070</td>
<td>0,408</td>
<td>0,481</td>
<td>0,441</td>
</tr>
<tr>
<td>32 ExtraTree</td>
<td>130</td>
<td>45,614</td>
<td>0,468</td>
<td>0,456</td>
<td>0,460</td>
</tr>
<tr>
<td>33 Logistic</td>
<td>122</td>
<td>42,807</td>
<td>0,464</td>
<td>0,428</td>
<td>0,441</td>
</tr>
<tr>
<td>34 MultiClassClassifier</td>
<td>121</td>
<td>42,456</td>
<td>0,452</td>
<td>0,425</td>
<td>0,435</td>
</tr>
<tr>
<td>35 VFI</td>
<td>62</td>
<td>21,754</td>
<td>?</td>
<td>0,218</td>
<td>?</td>
</tr>
</tbody>
</table>
D. GKO, BOA ve HBO-BPBÇ Algoritmalarının 200 Popülasyon 100 İterasyon Parametreleri ile Feminist Hareket Veri Seti’ne Uygulanmasından Elde Edilen Sonuçlar

GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 200 popülasyon ve 100 iterasyon parametreleri ile feminist hareket veri setine uygulanması sonucunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.7’de GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar WEKA’da yer alan algoritmalar ile elde edilen sonuçlar ile karşılaştırıldığında HBO-BPBÇ %58,5964 doğru sınıflandırma oranı ile birinci, GKO %57,1929 ile ikinci ve BOA ise %56,4912 ile üçüncü sıradada yer almıştır. Şekil 7.4’de algoritmaların çalışmaları sırasında 100 iterasyon boyunca uygunluk değerinin değişimi gösterilmiştir

Şekil 7.4. Feminist Hareket Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 100 iterasyon parametrelerindeki uygunluk değeri değişimi

Tablo 7.7. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 100 iterasyon parametreleri ile feminist hareket veri setine uygulanması sonucunda elde edilen başarı sırası.

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandıran Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Olçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 HBO-BPBÇ</td>
<td>167</td>
<td>58,5964</td>
<td>0,528</td>
<td>0,537</td>
<td>0,510</td>
</tr>
<tr>
<td>2 BOA</td>
<td>166</td>
<td>58,2456</td>
<td>0,521</td>
<td>0,572</td>
<td>0,513</td>
</tr>
<tr>
<td>3 GKO</td>
<td>163</td>
<td>57,1929</td>
<td>0,526</td>
<td>0,559</td>
<td>0,537</td>
</tr>
<tr>
<td>4 Bagging</td>
<td>163</td>
<td>57,193</td>
<td>0,540</td>
<td>0,572</td>
<td>0,547</td>
</tr>
<tr>
<td>5 LibSVM</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>6 QDA</td>
<td>162</td>
<td>56,8421</td>
<td>0,525</td>
<td>0,568</td>
<td>0,521</td>
</tr>
</tbody>
</table>
E. GKO, BOA ve HBO-BPBÇ Algoritmalarının 200 Popülasyon 150 İterasyon Parametreleri ile Feminist Hareket Veri Seti’ne Uygulanmasından Elde Edilen Sonuç

GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 200 popülasyon ve 150 iterasyon parametreleri ile feminist hareket veri setine uygulanması sonucunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.8’de GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar WEKA’da yer alan algoritmalar ile elde edilen sonuçlar ile karşılaştırıldığında BOA %58,2456 doğru sınıflandırma oranı ile birinci, GKO %57,8995 ile ikinci ve HBO-BPBÇ ise %57,5439 ile üçüncü sırada yer almıştır. Şekil 7.5’de algoritmaların çalışmaları sırasında 150 iterasyon boyunca uygunluk değerinin değişimi gösterilmiştir.
Şekil 7.5. Feminist Hareket Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 150 iterasyon uygunluk değerini değişimini gösteren grafik.

Tablo 7.8. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 150 iterasyon parametreleri ile feminizm veri setine uygulanması sonucunda elde edilen başarı sırası.

<table>
<thead>
<tr>
<th>Sıra</th>
<th>Algoritma:</th>
<th>Doğru Sınıflandırılan Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Olçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>BOA</td>
<td>166</td>
<td>58,2456</td>
<td>0,510</td>
<td>0,536</td>
<td>0,513</td>
</tr>
<tr>
<td>2</td>
<td>GKO</td>
<td>165</td>
<td>57,8995</td>
<td>0,520</td>
<td>0,512</td>
<td>0,513</td>
</tr>
<tr>
<td>3</td>
<td>HBO-BPBÇ</td>
<td>164</td>
<td>57,5439</td>
<td>0,496</td>
<td>0,535</td>
<td>0,504</td>
</tr>
<tr>
<td>4</td>
<td>Bagging</td>
<td>163</td>
<td>57,193</td>
<td>0,540</td>
<td>0,572</td>
<td>0,547</td>
</tr>
<tr>
<td>5</td>
<td>LibSVM</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>6</td>
<td>QDA</td>
<td>162</td>
<td>56,8421</td>
<td>0,525</td>
<td>0,568</td>
<td>0,521</td>
</tr>
<tr>
<td>7</td>
<td>Stacking</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>8</td>
<td>MultiScheme</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>9</td>
<td>ClojureClassifier</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>10</td>
<td>REPTree</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>11</td>
<td>Vote</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>12</td>
<td>CSForest</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>13</td>
<td>InputMappedClassifier</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>14</td>
<td>MultilayerPerceptronCS</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>15</td>
<td>MODLEM</td>
<td>159</td>
<td>55,7895</td>
<td>0,535</td>
<td>0,558</td>
<td>0,528</td>
</tr>
<tr>
<td>16</td>
<td>IterativeClassifierOptimize r</td>
<td>158</td>
<td>55,4386</td>
<td>0,431</td>
<td>0,554</td>
<td>0,453</td>
</tr>
<tr>
<td>17</td>
<td>HyperPipes</td>
<td>158</td>
<td>55,4386</td>
<td>0,506</td>
<td>0,554</td>
<td>0,461</td>
</tr>
<tr>
<td>18</td>
<td>NaiveBayes</td>
<td>157</td>
<td>55,0877</td>
<td>0,602</td>
<td>0,551</td>
<td>0,564</td>
</tr>
<tr>
<td>19</td>
<td>DecisionTable</td>
<td>156</td>
<td>54,7368</td>
<td>0,407</td>
<td>0,547</td>
<td>0,434</td>
</tr>
<tr>
<td>20</td>
<td>WISARD</td>
<td>155</td>
<td>54,386</td>
<td>0,500</td>
<td>0,544</td>
<td>0,498</td>
</tr>
<tr>
<td>21</td>
<td>SimpleLogistic</td>
<td>155</td>
<td>54,386</td>
<td>0,494</td>
<td>0,544</td>
<td>0,503</td>
</tr>
</tbody>
</table>

116
F. GKO, BOA ve HBO-BPBC Algoritmalarının 200 Popülasyon 200 İterasyon Parametreleri ile Feminist Hareket Veri Seti’ne Uygulanmasından Elde Edilen Sonuç

GKO, BOA ve HBO-BPBC algoritmalarının sırası ile 200 popülasyon ve 200 iterasyon parametreleri ile feminist hareket veri setine uygulanması sonucunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.8’de GKO, BOA ve HBO-BPBC algoritmaları ile elde edilen sonuçlar, WEKA’da yer alan algoritmalar kullanılarak elde edilen sonuçlar ile karşılaştırıldığında BOA %61,0526 doğru sınıflandırma oranı ile birinci, HBO-BPBC %57,8947 ile ikinci ve GKO ise %57,8947 ile üçüncü sırada yer almıştır. Ayrıca BOA’nın standart sapması 7,48, HBO-BPBC’nin 6,33 ve GKO’nun ise 9,00 olmuştur. Şekil 7.6’da algoritmaların çalışmaları sırasında 200 iterasyon boyunca uyguluk değerinin değişimi gösterilmiştir.
Şekil 7.6. Feminist Hareket Veri Seti’ni kullanılarak GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 200 iterasyon parametrelerindeki uygunluk değeri değişimi

Tablo 7.9. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 200 iterasyon parametreleri ile feminist hareket veri setine uygulanması sonucunda elde edilen başarı sırası.
G. BOA Algoritmanın Feminist Hareket Veri Setinde En İyi Sonuç Aldığı 100 Popülasyon 100 İterasyon Parametreleri Sabit Tutarak $p$ Parametresinin Üç Değeri için Test Edilmesi

Yukarda 100 popülasyon 100 iterasyon parametreleri ile yapılan testlerde $p$ parametresi 0,5 olarak alınmıştır. Bundan dolayı aşağıdaki deneylerde bu parametre sırası ile 0,25 ve 1 değerlerinde test edilecektir.

i) BOA Algoritması için $p$ Parametresinin 0,25 Almaası ile Elde Edilen Sonuç

BOA algoritması Feminist hareket veri seti üzerinde en iyi sonucu verdiği diğer parametre değerleri sabit tutulmak sureti ile $p$ parametresi 0,25 alınarak test edilmiştir. Yapılan denemeler sonucunda en iyi sonucun geçilemediği görülmüştür. 20 çalıştırma sonucunda elde edilen en iyi sonuç Tablo 7.11’de gösterilmiştir.

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>163</th>
<th>57,193</th>
<th>0,540</th>
<th>0,572</th>
<th>0,547</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bagging</td>
<td>162</td>
<td>56,8421</td>
<td>0,525</td>
<td>0,568</td>
<td>0,521</td>
</tr>
<tr>
<td>LibSVM</td>
<td>162</td>
<td>56,8421</td>
<td>0,525</td>
<td>0,568</td>
<td>0,521</td>
</tr>
<tr>
<td>QDA</td>
<td>162</td>
<td>56,8421</td>
<td>0,525</td>
<td>0,568</td>
<td>0,521</td>
</tr>
<tr>
<td>Stacking</td>
<td>162</td>
<td>56,8421</td>
<td>0,525</td>
<td>0,568</td>
<td>0,521</td>
</tr>
<tr>
<td>MultiScheme</td>
<td>162</td>
<td>56,8421</td>
<td>0,525</td>
<td>0,568</td>
<td>0,521</td>
</tr>
<tr>
<td>ClojureClassifier</td>
<td>162</td>
<td>56,8421</td>
<td>0,525</td>
<td>0,568</td>
<td>0,521</td>
</tr>
<tr>
<td>REPTree</td>
<td>162</td>
<td>56,8421</td>
<td>0,525</td>
<td>0,568</td>
<td>0,521</td>
</tr>
<tr>
<td>Vote</td>
<td>162</td>
<td>56,8421</td>
<td>0,525</td>
<td>0,568</td>
<td>0,521</td>
</tr>
<tr>
<td>CSForest</td>
<td>162</td>
<td>56,8421</td>
<td>0,525</td>
<td>0,568</td>
<td>0,521</td>
</tr>
</tbody>
</table>
ii) BOA Algoritması için p Parametresinin 1 Alınması ile Elde Edilen Sonuç

BOA algoritması feminist hareket veri seti üzerinde en iyi sonucu verdiği diğer parametre değerleri sabit tutulmak sureti ile p parametresi 1 alınarak test edilmiştir. Yapılan denemeler sonucunda en iyi sonucun geçilemediği görülmüştür. 20 defa çalışmaya sonucunda elde edilen en iyi sonuc Tablo 7.11’de gösterilmiştir.

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandırılan Örnek Sayısı</th>
<th>Doğruluk Yuzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bagging</td>
<td>163</td>
<td>57,193</td>
<td>0,540</td>
<td>0,572</td>
<td>0,547</td>
</tr>
<tr>
<td>LibSVM</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>QDA</td>
<td>162</td>
<td>56,8421</td>
<td>0,525</td>
<td>0,568</td>
<td>0,521</td>
</tr>
<tr>
<td>Stacking</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>MultiScheme</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>ClojureClassifier</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>REPTree</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>Rakam</td>
<td>Ad</td>
<td>Sonuçlar</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>----</td>
<td>----------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Vote</td>
<td>162</td>
<td>56.8421</td>
<td>?</td>
<td>0,568</td>
</tr>
<tr>
<td>9</td>
<td>CSForest</td>
<td>162</td>
<td>56.8421</td>
<td>?</td>
<td>0,568</td>
</tr>
<tr>
<td>10</td>
<td>InputMappedClassifier</td>
<td>162</td>
<td>56.8421</td>
<td>?</td>
<td>0,568</td>
</tr>
<tr>
<td>11</td>
<td>MultilayerPerceptronCS</td>
<td>162</td>
<td>56.8421</td>
<td>?</td>
<td>0,568</td>
</tr>
<tr>
<td>12</td>
<td>BOA</td>
<td>160</td>
<td>56.1404</td>
<td>0,605</td>
<td>0,637</td>
</tr>
<tr>
<td>13</td>
<td>MODLEM</td>
<td>159</td>
<td>55.7895</td>
<td>0,535</td>
<td>0,558</td>
</tr>
<tr>
<td>14</td>
<td>IterativeClassifierOptimizer</td>
<td>158</td>
<td>55.4386</td>
<td>0,431</td>
<td>0,554</td>
</tr>
<tr>
<td>15</td>
<td>HyperPipes</td>
<td>158</td>
<td>55.4386</td>
<td>0,506</td>
<td>0,554</td>
</tr>
<tr>
<td>16</td>
<td>NaiveBayes</td>
<td>157</td>
<td>55.0877</td>
<td>0,602</td>
<td>0,551</td>
</tr>
<tr>
<td>17</td>
<td>DecisionTable</td>
<td>156</td>
<td>54.7368</td>
<td>0,407</td>
<td>0,547</td>
</tr>
<tr>
<td>18</td>
<td>WISARD</td>
<td>155</td>
<td>54.386</td>
<td>0,500</td>
<td>0,544</td>
</tr>
<tr>
<td>19</td>
<td>SimpleLogistic</td>
<td>155</td>
<td>54.386</td>
<td>0,494</td>
<td>0,544</td>
</tr>
<tr>
<td>20</td>
<td>SMO</td>
<td>155</td>
<td>54.386</td>
<td>0,523</td>
<td>0,544</td>
</tr>
<tr>
<td>21</td>
<td>OLM</td>
<td>155</td>
<td>54.386</td>
<td>?</td>
<td>0,544</td>
</tr>
<tr>
<td>22</td>
<td>RseslibKnn</td>
<td>154</td>
<td>54.0351</td>
<td>0,479</td>
<td>0,540</td>
</tr>
<tr>
<td>23</td>
<td>RandomForest</td>
<td>152</td>
<td>53.3333</td>
<td>0,516</td>
<td>0,533</td>
</tr>
<tr>
<td>24</td>
<td>IBk</td>
<td>148</td>
<td>51.9298</td>
<td>0,484</td>
<td>0,519</td>
</tr>
<tr>
<td>25</td>
<td>IB1</td>
<td>147</td>
<td>51.5789</td>
<td>0,461</td>
<td>0,516</td>
</tr>
<tr>
<td>26</td>
<td>RandomTree</td>
<td>147</td>
<td>51.5789</td>
<td>0,507</td>
<td>0,516</td>
</tr>
<tr>
<td>27</td>
<td>OrdinalClassifier</td>
<td>142</td>
<td>49.8246</td>
<td>?</td>
<td>0,498</td>
</tr>
<tr>
<td>28</td>
<td>LDA</td>
<td>139</td>
<td>48.7719</td>
<td>0,497</td>
<td>0,488</td>
</tr>
<tr>
<td>29</td>
<td>RBFNetwork</td>
<td>137</td>
<td>48.0702</td>
<td>0,408</td>
<td>0,481</td>
</tr>
<tr>
<td>30</td>
<td>ExtraTree</td>
<td>130</td>
<td>45.614</td>
<td>0,468</td>
<td>0,456</td>
</tr>
<tr>
<td>31</td>
<td>Logistic</td>
<td>122</td>
<td>42.807</td>
<td>0,464</td>
<td>0,428</td>
</tr>
<tr>
<td>32</td>
<td>MultiClassClassifier</td>
<td>121</td>
<td>42.4561</td>
<td>0,452</td>
<td>0,425</td>
</tr>
<tr>
<td>33</td>
<td>VFI</td>
<td>62</td>
<td>21.7544</td>
<td>?</td>
<td>0,218</td>
</tr>
</tbody>
</table>

H. GKO Algoritmasının Feminist Hareket Veri Setinde En İyi Sonuç Aldığı 100 Popülasyon 100 İterasyon Parametreleri Sabit Tutularak t Parametresinin Üç Değeri için Test Edilmesi

Yukarda 100 popülasyon 100 iterasyon parametreleri ile yapılan testlerde t parametresi 0 olarak alınmıştır. Bundan dolayı aşağıdaki deneylerde bu parametre sırası ile 2 ve 4 değerlerinde test edilecektir.

i) GKO Algoritması için t Parametresinin 2 Alınması ile Elde Edilen Sonuç

GKO algoritması Feminist hareket veri seti üzerinde en iyi sonucu verdiği diğer parametre değerleri sabit tutulmak sureti ile t parametresi 2 alınarak test edilmiştir. Yapılan denemeler sonucunda en iyi sonucun geçmediği görülmuştur. 20 çalıtırma sonucunda elde edilen en iyi sonuç Tablo 7.1’de gösterilmiştir.
<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandıran Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bagging</td>
<td>163</td>
<td>57,193</td>
<td>0,540</td>
<td>0,572</td>
<td>0,547</td>
</tr>
<tr>
<td>GKO</td>
<td>162</td>
<td>56,8421</td>
<td>0,540</td>
<td>0,589</td>
<td>0,545</td>
</tr>
<tr>
<td>LibSVM</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>QDA</td>
<td>162</td>
<td>56,8421</td>
<td>0,525</td>
<td>0,568</td>
<td>0,521</td>
</tr>
<tr>
<td>Stacking</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>MultiScheme</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>ClojureClassifier</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>REPTree</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>Vote</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>CSForest</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>InputMappedClassifier</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>MODLEM</td>
<td>159</td>
<td>55,7895</td>
<td>0,535</td>
<td>0,558</td>
<td>0,528</td>
</tr>
<tr>
<td>IterativeClassifierOptimizer</td>
<td>158</td>
<td>55,4386</td>
<td>0,431</td>
<td>0,554</td>
<td>0,453</td>
</tr>
<tr>
<td>HyperPipes</td>
<td>158</td>
<td>55,4386</td>
<td>0,506</td>
<td>0,554</td>
<td>0,461</td>
</tr>
<tr>
<td>NaiveBayes</td>
<td>157</td>
<td>55,0877</td>
<td>0,602</td>
<td>0,551</td>
<td>0,564</td>
</tr>
<tr>
<td>DecisionTable</td>
<td>156</td>
<td>54,7368</td>
<td>0,407</td>
<td>0,547</td>
<td>0,434</td>
</tr>
<tr>
<td>WISARD</td>
<td>155</td>
<td>54,386</td>
<td>0,500</td>
<td>0,544</td>
<td>0,498</td>
</tr>
<tr>
<td>SimpleLogistic</td>
<td>155</td>
<td>54,386</td>
<td>0,494</td>
<td>0,544</td>
<td>0,503</td>
</tr>
<tr>
<td>SMO</td>
<td>155</td>
<td>54,386</td>
<td>0,523</td>
<td>0,544</td>
<td>0,531</td>
</tr>
<tr>
<td>OLM</td>
<td>155</td>
<td>54,386</td>
<td>?</td>
<td>0,544</td>
<td>?</td>
</tr>
<tr>
<td>RseslibKnn</td>
<td>154</td>
<td>54,0351</td>
<td>0,479</td>
<td>0,540</td>
<td>0,489</td>
</tr>
<tr>
<td>RandomForest</td>
<td>152</td>
<td>53,3333</td>
<td>0,516</td>
<td>0,533</td>
<td>0,522</td>
</tr>
<tr>
<td>IBk</td>
<td>148</td>
<td>51,9298</td>
<td>0,484</td>
<td>0,519</td>
<td>0,494</td>
</tr>
<tr>
<td>IB1</td>
<td>147</td>
<td>51,5789</td>
<td>0,461</td>
<td>0,516</td>
<td>0,472</td>
</tr>
<tr>
<td>RandomTree</td>
<td>147</td>
<td>51,5789</td>
<td>0,507</td>
<td>0,516</td>
<td>0,511</td>
</tr>
<tr>
<td>OrdinalClassClassifier</td>
<td>142</td>
<td>49,8246</td>
<td>?</td>
<td>0,498</td>
<td>?</td>
</tr>
<tr>
<td>LDA</td>
<td>139</td>
<td>48,7719</td>
<td>0,497</td>
<td>0,488</td>
<td>0,492</td>
</tr>
<tr>
<td>RBFNetwork</td>
<td>137</td>
<td>48,0702</td>
<td>0,408</td>
<td>0,481</td>
<td>0,441</td>
</tr>
<tr>
<td>ExtraTree</td>
<td>130</td>
<td>45,614</td>
<td>0,468</td>
<td>0,456</td>
<td>0,460</td>
</tr>
<tr>
<td>Logistic</td>
<td>122</td>
<td>42,807</td>
<td>0,464</td>
<td>0,428</td>
<td>0,441</td>
</tr>
<tr>
<td>MultiClassClassifier</td>
<td>121</td>
<td>42,4561</td>
<td>0,452</td>
<td>0,425</td>
<td>0,435</td>
</tr>
<tr>
<td>VFI</td>
<td>62</td>
<td>21,7544</td>
<td>?</td>
<td>0,218</td>
<td>?</td>
</tr>
</tbody>
</table>

ii) GKO Algoritması için $t$ Parametresinin 4 Alınması ile Elde Edilen Sonuç

GKO algoritması feminist hareket veri seti üzerinden en iyi sonucu verdiği diğer parametre değerleri sabit tutulmak sureti ile $t$ parametresi 4 alınarak test edilmiştir. Yapılan denemeler
sonucunda en iyi sonucun geçilemediği görülmüştür. 20 çalıştırma sonucunda elde edilen en iyi sonuc Tablo 7.13’te gösterilmiştir.

**Tablo 7.13.** GKO algoritması için τ parametresinin 4 alınması ile elde edilen başarı sırası

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğruluk Sınıflandırılan Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasılık</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Bagging</td>
<td>163</td>
<td>57,193</td>
<td>0,540</td>
<td>0,572</td>
<td>0,547</td>
</tr>
<tr>
<td>2 GKO</td>
<td>162</td>
<td>56,8421</td>
<td>0,512</td>
<td>0,535</td>
<td>0,519</td>
</tr>
<tr>
<td>3 LibSVM</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>4 QDA</td>
<td>162</td>
<td>56,8421</td>
<td>0,525</td>
<td>0,568</td>
<td>0,521</td>
</tr>
<tr>
<td>5 Stacking</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>6 MultiScheme</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>7 ClojureClassifier</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>8 REPTree</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>9 Vote</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>10 CSForest</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>11 InputMappedClassifier</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>12 MultilayerPerceptronCS</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>13 MODLEM</td>
<td>159</td>
<td>55,7895</td>
<td>0,535</td>
<td>0,558</td>
<td>0,528</td>
</tr>
<tr>
<td>14 IterativeClassifierOptimizer</td>
<td>158</td>
<td>55,4386</td>
<td>0,431</td>
<td>0,554</td>
<td>0,453</td>
</tr>
<tr>
<td>15 HyperPipes</td>
<td>158</td>
<td>55,4386</td>
<td>0,506</td>
<td>0,554</td>
<td>0,461</td>
</tr>
<tr>
<td>16 NaiveBayes</td>
<td>157</td>
<td>55,0877</td>
<td>0,602</td>
<td>0,551</td>
<td>0,564</td>
</tr>
<tr>
<td>17 DecisionTable</td>
<td>156</td>
<td>54,7368</td>
<td>0,407</td>
<td>0,547</td>
<td>0,434</td>
</tr>
<tr>
<td>18 WISARD</td>
<td>155</td>
<td>54,386</td>
<td>0,500</td>
<td>0,544</td>
<td>0,498</td>
</tr>
<tr>
<td>19 SimpleLogistic</td>
<td>155</td>
<td>54,386</td>
<td>0,494</td>
<td>0,544</td>
<td>0,503</td>
</tr>
<tr>
<td>20 SMO</td>
<td>155</td>
<td>54,386</td>
<td>0,523</td>
<td>0,544</td>
<td>0,531</td>
</tr>
<tr>
<td>21 OLM</td>
<td>155</td>
<td>54,386</td>
<td>?</td>
<td>0,544</td>
<td>?</td>
</tr>
<tr>
<td>22 RseslibKnn</td>
<td>154</td>
<td>54,0351</td>
<td>0,479</td>
<td>0,540</td>
<td>0,489</td>
</tr>
<tr>
<td>23 RandomForest</td>
<td>152</td>
<td>53,3333</td>
<td>0,516</td>
<td>0,533</td>
<td>0,522</td>
</tr>
<tr>
<td>24 IBk</td>
<td>148</td>
<td>51,9298</td>
<td>0,484</td>
<td>0,519</td>
<td>0,494</td>
</tr>
<tr>
<td>25 IB1</td>
<td>147</td>
<td>51,5789</td>
<td>0,461</td>
<td>0,516</td>
<td>0,472</td>
</tr>
<tr>
<td>26 RandomForest</td>
<td>147</td>
<td>51,5789</td>
<td>0,507</td>
<td>0,516</td>
<td>0,511</td>
</tr>
<tr>
<td>27 OrdinalClassClassifier</td>
<td>142</td>
<td>49,8246</td>
<td>?</td>
<td>0,498</td>
<td>?</td>
</tr>
<tr>
<td>28 LDA</td>
<td>139</td>
<td>48,7719</td>
<td>0,497</td>
<td>0,488</td>
<td>0,492</td>
</tr>
<tr>
<td>29 RBFNetwork</td>
<td>137</td>
<td>48,0702</td>
<td>0,408</td>
<td>0,481</td>
<td>0,441</td>
</tr>
<tr>
<td>30 ExtraTree</td>
<td>130</td>
<td>45,614</td>
<td>0,468</td>
<td>0,456</td>
<td>0,460</td>
</tr>
<tr>
<td>31 Logistic</td>
<td>122</td>
<td>42,807</td>
<td>0,464</td>
<td>0,428</td>
<td>0,441</td>
</tr>
<tr>
<td>32 MultiClassClassifier</td>
<td>121</td>
<td>42,4561</td>
<td>0,452</td>
<td>0,425</td>
<td>0,435</td>
</tr>
<tr>
<td>33 VFI</td>
<td>62</td>
<td>21,7544</td>
<td>?</td>
<td>0,218</td>
<td>?</td>
</tr>
</tbody>
</table>
I. Feminist hareket veri seti kullanılarak algoritmaların uygunluk fonksiyonlarının eşit sayıda çağrılma sayısına göre karşılaştırılması

Uygunluk fonksiyonunun 40400 defa çağrılma değeri baz alınarak BOA ve HBO-BPBC algoritmalarının sonuçları WEKA’da yer alan algoritmalar kullanılarak elde edilen sonuçlar ile karşılaştırıldığında BOA %58,9473 doğru sınıflandırma oranı elde etmiştir. HBO-BPBC %58,5965 ile ikinci sıraya yer almıştır. Aşağıda Tablo 7.14’te sıralama sonuçları gösterilmiştir. Bu sonuçlara göre HBO-BPBC algoritmalarının uygunluk fonksiyonun eşit çağrılma sayısı baz alınarak yapılan testlerde de başarılı olduğu görülmüştür.

Tablo 7.14 BOA ve HBO-BPBC algoritmaları feminist hareket veri setinde uygunluk fonksiyonlarının eşit sayıda çağrılması sonuçları

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandıran Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 BOA</td>
<td>168</td>
<td>58,9473</td>
<td>0,518</td>
<td>0,551</td>
<td>0,508</td>
</tr>
<tr>
<td>2 HBO-BPBC</td>
<td>167</td>
<td>58,5965</td>
<td>0,457</td>
<td>0,511</td>
<td>0,449</td>
</tr>
<tr>
<td>3 GKO</td>
<td>166</td>
<td>58,2456</td>
<td>0,489</td>
<td>0,476</td>
<td>0,450</td>
</tr>
<tr>
<td>4 Bagging</td>
<td>163</td>
<td>57,193</td>
<td>0,540</td>
<td>0,572</td>
<td>0,547</td>
</tr>
<tr>
<td>5 LibSVM</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>6 QDA</td>
<td>162</td>
<td>56,8421</td>
<td>0,525</td>
<td>0,568</td>
<td>0,521</td>
</tr>
<tr>
<td>7 Stacking</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>8 MultiScheme</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>9 ClojureClassifier</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>10 REPTree</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>11 Vote</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>12 CSForest</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>13 InputMappedClassifier</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>14 MultilayerPerceptronCS</td>
<td>162</td>
<td>56,8421</td>
<td>?</td>
<td>0,568</td>
<td>?</td>
</tr>
<tr>
<td>15 MODLEM</td>
<td>159</td>
<td>55,7895</td>
<td>0,535</td>
<td>0,558</td>
<td>0,528</td>
</tr>
<tr>
<td>16 IterativeClassifierOptimize</td>
<td>158</td>
<td>55,4386</td>
<td>0,431</td>
<td>0,554</td>
<td>0,453</td>
</tr>
<tr>
<td>17 HyperPipes</td>
<td>158</td>
<td>55,4386</td>
<td>0,506</td>
<td>0,554</td>
<td>0,461</td>
</tr>
<tr>
<td>18 NaiveBayes</td>
<td>157</td>
<td>55,0877</td>
<td>0,602</td>
<td>0,551</td>
<td>0,564</td>
</tr>
<tr>
<td>19 DecisionTable</td>
<td>156</td>
<td>54,7368</td>
<td>0,407</td>
<td>0,547</td>
<td>0,434</td>
</tr>
<tr>
<td>20 WISARD</td>
<td>155</td>
<td>54,386</td>
<td>0,500</td>
<td>0,544</td>
<td>0,498</td>
</tr>
<tr>
<td>21 SimpleLogistic</td>
<td>155</td>
<td>54,386</td>
<td>0,494</td>
<td>0,544</td>
<td>0,503</td>
</tr>
<tr>
<td>22 SMO</td>
<td>155</td>
<td>54,386</td>
<td>0,523</td>
<td>0,544</td>
<td>0,531</td>
</tr>
<tr>
<td>23 OLM</td>
<td>155</td>
<td>54,386</td>
<td>?</td>
<td>0,544</td>
<td>?</td>
</tr>
<tr>
<td>24 RseslibKnn</td>
<td>154</td>
<td>54,0351</td>
<td>0,479</td>
<td>0,540</td>
<td>0,489</td>
</tr>
<tr>
<td>25 RandomForest</td>
<td>152</td>
<td>53,3333</td>
<td>0,516</td>
<td>0,533</td>
<td>0,522</td>
</tr>
<tr>
<td>26 IBk</td>
<td>148</td>
<td>51,9298</td>
<td>0,484</td>
<td>0,519</td>
<td>0,494</td>
</tr>
<tr>
<td>27 IB1</td>
<td>147</td>
<td>51,5789</td>
<td>0,461</td>
<td>0,516</td>
<td>0,472</td>
</tr>
</tbody>
</table>
7.1.2. Ateizm Veri Seti’nden Elde Edilen Sonuçlar


Tablo 7.15. Ateizm veri seti için kullanılan katsayılar

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Benzerlik</th>
<th>k₁</th>
<th>k₂</th>
<th>c₁</th>
<th>c₂</th>
<th>c₃</th>
<th>c₄</th>
</tr>
</thead>
<tbody>
<tr>
<td>GKO</td>
<td>0,01</td>
<td>0,9</td>
<td>0,1</td>
<td>0,01</td>
<td>0,39</td>
<td>0,3</td>
<td>0,3</td>
</tr>
<tr>
<td>BOA</td>
<td>0,02</td>
<td>0,9</td>
<td>0,1</td>
<td>0,01</td>
<td>0,39</td>
<td>0,3</td>
<td>0,3</td>
</tr>
<tr>
<td>HBO-BPBÇ</td>
<td>0,02</td>
<td>0,9</td>
<td>0,1</td>
<td>0,01</td>
<td>0,39</td>
<td>0,3</td>
<td>0,3</td>
</tr>
</tbody>
</table>

Tablo 7.16. Weka programı kullanılarak Ateizm veri setinden elde edilen sınıflandırma sonuçları

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandırılan Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>RandomForest</td>
<td>138</td>
<td>62,7273</td>
<td>0,622</td>
<td>0,627</td>
<td>0,618</td>
</tr>
<tr>
<td>NaïveBayes</td>
<td>135</td>
<td>61,3636</td>
<td>0,647</td>
<td>0,614</td>
<td>0,557</td>
</tr>
</tbody>
</table>
GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 100 popülasyon ve 100 iterasyon parametreleri ile Ateizm Veri Seti’ne uygulanması sonucunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.11’dede GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar, WEKA’dada yer alan algoritmalar kullanılarak elde edilen sonuçlar ile karşılaştırıldığında HBO-BPBÇ %61,8182 doğru sınıflandırma oranı ile ikinci, BOA %61,3636 ile üçüncü ve GKO ise %61,3636 ile dördüncü sıralarda yer almıştır. Şekil 7.7’de algoritmaların çalışmaları sırasında 100 iterasyon boyunca uygunluk değerinin değişimi gösterilmiştir.

<table>
<thead>
<tr>
<th>No.</th>
<th>Adı</th>
<th>Popülasyon</th>
<th>1. Oran</th>
<th>2. Oran</th>
<th>3. Oran</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>RandomTree</td>
<td>130</td>
<td>0,585</td>
<td>0,591</td>
<td>0,587</td>
</tr>
<tr>
<td>2</td>
<td>MultilayerPerceptronCS</td>
<td>128</td>
<td>0,620</td>
<td>0,582</td>
<td>0,509</td>
</tr>
<tr>
<td>3</td>
<td>SMO</td>
<td>128</td>
<td>0,575</td>
<td>0,582</td>
<td>0,541</td>
</tr>
<tr>
<td>4</td>
<td>OrdinalClassClassifier</td>
<td>128</td>
<td>0,582</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>QDA</td>
<td>127</td>
<td>0,586</td>
<td>0,577</td>
<td>0,504</td>
</tr>
<tr>
<td>6</td>
<td>LDA</td>
<td>126</td>
<td>0,562</td>
<td>0,573</td>
<td>0,565</td>
</tr>
<tr>
<td>7</td>
<td>ExtraTree</td>
<td>126</td>
<td>0,570</td>
<td>0,573</td>
<td>0,569</td>
</tr>
<tr>
<td>8</td>
<td>Bagging</td>
<td>125</td>
<td>0,732</td>
<td>0,568</td>
<td>0,455</td>
</tr>
<tr>
<td>9</td>
<td>WISARD</td>
<td>125</td>
<td>0,589</td>
<td>0,568</td>
<td>0,492</td>
</tr>
<tr>
<td>10</td>
<td>SimpleLogistic</td>
<td>124</td>
<td>0,594</td>
<td>0,564</td>
<td>0,478</td>
</tr>
<tr>
<td>11</td>
<td>IterativeClassifierOptimizera</td>
<td>122</td>
<td>0,592</td>
<td>0,555</td>
<td>0,446</td>
</tr>
<tr>
<td>12</td>
<td>RBFNetwork</td>
<td>121</td>
<td>0,492</td>
<td>0,550</td>
<td>0,503</td>
</tr>
<tr>
<td>13</td>
<td>MODLEM</td>
<td>121</td>
<td>0,515</td>
<td>0,550</td>
<td>0,464</td>
</tr>
<tr>
<td>14</td>
<td>IBk</td>
<td>119</td>
<td>0,563</td>
<td>0,541</td>
<td>0,532</td>
</tr>
<tr>
<td>15</td>
<td>DecisionTable</td>
<td>119</td>
<td>0,430</td>
<td>0,541</td>
<td>0,422</td>
</tr>
<tr>
<td>16</td>
<td>IB1</td>
<td>117</td>
<td>0,519</td>
<td>0,532</td>
<td>0,232</td>
</tr>
<tr>
<td>17</td>
<td>MultiClassClassifier</td>
<td>117</td>
<td>0,528</td>
<td>0,532</td>
<td>0,529</td>
</tr>
<tr>
<td>18</td>
<td>HyperPipes</td>
<td>117</td>
<td>0,525</td>
<td>0,532</td>
<td>0,406</td>
</tr>
<tr>
<td>19</td>
<td>MultiScheme</td>
<td>116</td>
<td>0,527</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>LibSVM</td>
<td>116</td>
<td>0,527</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>RseslibKnn</td>
<td>116</td>
<td>0,598</td>
<td>0,527</td>
<td>0,429</td>
</tr>
<tr>
<td>22</td>
<td>REPTree</td>
<td>116</td>
<td>0,527</td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>ClojureClassifier</td>
<td>116</td>
<td>0,527</td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>Stacking</td>
<td>116</td>
<td>0,527</td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>Vote</td>
<td>116</td>
<td>0,527</td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>InputMappedClassifier</td>
<td>116</td>
<td>0,527</td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>CSForest</td>
<td>116</td>
<td>0,527</td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>Logistic</td>
<td>117</td>
<td>0,514</td>
<td>0,505</td>
<td>0,508</td>
</tr>
<tr>
<td>29</td>
<td>OLM</td>
<td>97</td>
<td>0,441</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>VFI</td>
<td>57</td>
<td>0,259</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

A. GKO, BOA ve HBO-BPBÇ Algoritmalarının 100 Popülasyon 100 İterasyon Parametreleri ile Ateizm Veri Seti’ne Uygulanmasından Elde Edilen Sonuç
Şekil 7.7. Ateizm Veri Seti’ni kullanılarak GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 100 iterasyon parametrelerindeki uyguluk değeri değişimi

Tablo 7.17. GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 100 iterasyon parametreleri ile ateizm veri setine uygulanması sonucunda elde edilen başarı sırası.

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandırılan Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1  RandomForest</td>
<td>138</td>
<td>62,7273</td>
<td>0,622</td>
<td>0,627</td>
<td>0,618</td>
</tr>
<tr>
<td>2  HBO-BPBÇ</td>
<td>136</td>
<td>61,8182</td>
<td>0,599</td>
<td>0,526</td>
<td>0,538</td>
</tr>
<tr>
<td>3  BOA</td>
<td>135</td>
<td>61,3636</td>
<td>0,614</td>
<td>0,545</td>
<td>0,558</td>
</tr>
<tr>
<td>4  GKO</td>
<td>135</td>
<td>61,3636</td>
<td>0,579</td>
<td>0,567</td>
<td>0,571</td>
</tr>
<tr>
<td>5  NaiveBayes</td>
<td>135</td>
<td>61,3636</td>
<td>0,647</td>
<td>0,614</td>
<td>0,557</td>
</tr>
<tr>
<td>6  RandomTree</td>
<td>130</td>
<td>59,0909</td>
<td>0,585</td>
<td>0,591</td>
<td>0,587</td>
</tr>
<tr>
<td>7  MultilayerPerceptronCS</td>
<td>128</td>
<td>58,1818</td>
<td>0,620</td>
<td>0,582</td>
<td>0,509</td>
</tr>
<tr>
<td>8  SMO</td>
<td>128</td>
<td>58,1818</td>
<td>0,575</td>
<td>0,582</td>
<td>0,541</td>
</tr>
<tr>
<td>9  OrdinalClassClassifier</td>
<td>128</td>
<td>58,1818</td>
<td>0,582</td>
<td>0,582</td>
<td>0,541</td>
</tr>
<tr>
<td>10 QDA</td>
<td>127</td>
<td>57,7273</td>
<td>0,586</td>
<td>0,577</td>
<td>0,504</td>
</tr>
<tr>
<td>11 LDA</td>
<td>126</td>
<td>57,2727</td>
<td>0,562</td>
<td>0,573</td>
<td>0,565</td>
</tr>
<tr>
<td>12 ExtraTree</td>
<td>126</td>
<td>57,2727</td>
<td>0,570</td>
<td>0,573</td>
<td>0,569</td>
</tr>
<tr>
<td>13 Bagging</td>
<td>125</td>
<td>56,8182</td>
<td>0,732</td>
<td>0,568</td>
<td>0,455</td>
</tr>
<tr>
<td>14 WISARD</td>
<td>125</td>
<td>56,8182</td>
<td>0,589</td>
<td>0,568</td>
<td>0,492</td>
</tr>
<tr>
<td>15 SimpleLogistic</td>
<td>124</td>
<td>56,3636</td>
<td>0,594</td>
<td>0,564</td>
<td>0,478</td>
</tr>
<tr>
<td>16 IterativeClassifierOptimize</td>
<td>122</td>
<td>55,4545</td>
<td>0,592</td>
<td>0,555</td>
<td>0,446</td>
</tr>
<tr>
<td>17 RBFNetwork</td>
<td>121</td>
<td>55</td>
<td>0,492</td>
<td>0,550</td>
<td>0,503</td>
</tr>
<tr>
<td>18 MODLEM</td>
<td>121</td>
<td>55</td>
<td>0,515</td>
<td>0,550</td>
<td>0,464</td>
</tr>
<tr>
<td>19 IBk</td>
<td>119</td>
<td>54,0909</td>
<td>0,563</td>
<td>0,541</td>
<td>0,532</td>
</tr>
<tr>
<td></td>
<td>Adı</td>
<td>İkili</td>
<td>Popülasyon</td>
<td>İterasyon</td>
<td>İstest 1</td>
</tr>
<tr>
<td>-----</td>
<td>----------------------------</td>
<td>-------</td>
<td>------------</td>
<td>-----------</td>
<td>----------</td>
</tr>
<tr>
<td>12</td>
<td>DecisionTable</td>
<td>119</td>
<td>54,0909</td>
<td>0,430</td>
<td>0,541</td>
</tr>
<tr>
<td>13</td>
<td>IB1</td>
<td>117</td>
<td>53,1818</td>
<td>0,494</td>
<td>0,532</td>
</tr>
<tr>
<td>14</td>
<td>MultiClassClassifier</td>
<td>117</td>
<td>53,1818</td>
<td>0,528</td>
<td>0,532</td>
</tr>
<tr>
<td>15</td>
<td>HyperPipes</td>
<td>117</td>
<td>53,1818</td>
<td>0,525</td>
<td>0,532</td>
</tr>
<tr>
<td>16</td>
<td>MultiScheme</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>17</td>
<td>LibSVM</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>18</td>
<td>RseslibKnn</td>
<td>116</td>
<td>52,7273</td>
<td>0,598</td>
<td>0,527</td>
</tr>
<tr>
<td>21</td>
<td>REPTree</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>22</td>
<td>ClojureClassifier</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>23</td>
<td>Stacking</td>
<td>116</td>
<td>52,2727</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>24</td>
<td>Vote</td>
<td>116</td>
<td>52,2727</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>25</td>
<td>InputMappedClassifier</td>
<td>116</td>
<td>52,2727</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>26</td>
<td>CSForest</td>
<td>116</td>
<td>52,2727</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>27</td>
<td>Logistic</td>
<td>111</td>
<td>50,4545</td>
<td>0,514</td>
<td>0,505</td>
</tr>
<tr>
<td>28</td>
<td>OLM</td>
<td>97</td>
<td>44,0909</td>
<td>?</td>
<td>0,441</td>
</tr>
<tr>
<td>30</td>
<td>VFI</td>
<td>57</td>
<td>25,9090</td>
<td>?</td>
<td>0,259</td>
</tr>
</tbody>
</table>

**B. GKO, BOA ve HBO-BPBÇ Algoritmalarının 100 Popülasyon 150 İterasyon Parametreleri ile Ateizm Veri Seti’ne Uygulanmasından Elde Edilen Sonuç**

GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 100 popülasyon ve 150 iterasyon parametreleri ile Ateizm veri setine uygulanan sonucunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.18’de GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar, WEKA’da yer alan algoritmalar kullanılarak elde edilen sonuçlar ile karşılaştırıldığında GKO %62,2727 doğru sınıflandırma oranı ile birinci, BOA %61,8181 ile ikinci ve HBO-BPBÇ ise %61,3636 ile üçüncü sırada yer almıştır. Şekil 7.8’de algoritmaların çalışmaları sırasında 150 iterasyon boyunca uygunluk değerinin değişimi gösterilmiştir.
Şekil 7.8. Ateizm Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 150 iteraşyon parametrelerindeki uygunluk değeri değişimi

Tablo 7.18. GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 150 iteraşyon parametreleri ile ateziz veri setine uygulanması sonucunda elde edilen başarı sırası.

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandıran Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 RandomForest</td>
<td>138</td>
<td>62,7273</td>
<td>0,622</td>
<td>0,627</td>
<td>0,618</td>
</tr>
<tr>
<td>2 GKO</td>
<td>137</td>
<td>62,2727</td>
<td>0,626</td>
<td>0,578</td>
<td>0,588</td>
</tr>
<tr>
<td>3 BOA</td>
<td>136</td>
<td>61,8181</td>
<td>0,632</td>
<td>0,518</td>
<td>0,591</td>
</tr>
<tr>
<td>4 HBO-BPBÇ</td>
<td>135</td>
<td>61,3636</td>
<td>0,626</td>
<td>0,509</td>
<td>0,522</td>
</tr>
<tr>
<td>5 NaiveBayes</td>
<td>135</td>
<td>61,3636</td>
<td>0,647</td>
<td>0,614</td>
<td>0,557</td>
</tr>
<tr>
<td>6 RandomTree</td>
<td>130</td>
<td>59,0909</td>
<td>0,585</td>
<td>0,591</td>
<td>0,587</td>
</tr>
<tr>
<td>7 MultilayerPerceptronCS</td>
<td>128</td>
<td>58,1818</td>
<td>0,620</td>
<td>0,582</td>
<td>0,509</td>
</tr>
<tr>
<td>8 SMO</td>
<td>128</td>
<td>58,1818</td>
<td>0,575</td>
<td>0,582</td>
<td>0,541</td>
</tr>
<tr>
<td>9 OrdinalClassClassifier</td>
<td>128</td>
<td>58,1818</td>
<td>?</td>
<td>0,582</td>
<td>?</td>
</tr>
<tr>
<td>10 QDA</td>
<td>127</td>
<td>57,7273</td>
<td>0,586</td>
<td>0,577</td>
<td>0,504</td>
</tr>
<tr>
<td>11 LDA</td>
<td>126</td>
<td>57,2727</td>
<td>0,562</td>
<td>0,573</td>
<td>0,565</td>
</tr>
<tr>
<td>12 ExtraTree</td>
<td>126</td>
<td>57,2727</td>
<td>0,570</td>
<td>0,573</td>
<td>0,569</td>
</tr>
<tr>
<td>13 Bagging</td>
<td>125</td>
<td>56,8182</td>
<td>0,732</td>
<td>0,568</td>
<td>0,455</td>
</tr>
<tr>
<td>14 WISARD</td>
<td>125</td>
<td>56,8182</td>
<td>0,589</td>
<td>0,568</td>
<td>0,492</td>
</tr>
<tr>
<td>15 SimpleLogistic</td>
<td>124</td>
<td>56,3636</td>
<td>0,594</td>
<td>0,564</td>
<td>0,478</td>
</tr>
<tr>
<td>16 IterativeClassifierOptimizer</td>
<td>122</td>
<td>55,4545</td>
<td>0,592</td>
<td>0,555</td>
<td>0,446</td>
</tr>
<tr>
<td>17 RBFNetwork</td>
<td>121</td>
<td>55</td>
<td>0,492</td>
<td>0,550</td>
<td>0,503</td>
</tr>
<tr>
<td>18 MODLEM</td>
<td>121</td>
<td>55</td>
<td>0,515</td>
<td>0,550</td>
<td>0,464</td>
</tr>
<tr>
<td>19 IBk</td>
<td>119</td>
<td>54,0909</td>
<td>0,563</td>
<td>0,541</td>
<td>0,532</td>
</tr>
<tr>
<td>20 DecisionTable</td>
<td>119</td>
<td>54,0909</td>
<td>0,430</td>
<td>0,541</td>
<td>0,422</td>
</tr>
<tr>
<td>21 IB1</td>
<td>117</td>
<td>53,1818</td>
<td>0,494</td>
<td>0,532</td>
<td>0,423</td>
</tr>
<tr>
<td>No.</td>
<td>Algoritma</td>
<td>Popülasyon</td>
<td>Test</td>
<td>Train</td>
<td>DoğruSınıflandırma</td>
</tr>
<tr>
<td>-----</td>
<td>--------------------------</td>
<td>------------</td>
<td>------</td>
<td>-------</td>
<td>-------------------</td>
</tr>
<tr>
<td>22</td>
<td>MultiClassClassifier</td>
<td>117</td>
<td>53,181</td>
<td>0,528</td>
<td>0,532</td>
</tr>
<tr>
<td>23</td>
<td>HyperPipes</td>
<td>117</td>
<td>53,181</td>
<td>0,525</td>
<td>0,532</td>
</tr>
<tr>
<td>24</td>
<td>MultiScheme</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>25</td>
<td>LibSVM</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>26</td>
<td>RseslibKnn</td>
<td>116</td>
<td>52,7273</td>
<td>0,598</td>
<td>0,527</td>
</tr>
<tr>
<td>27</td>
<td>REPTree</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>28</td>
<td>ClojureClassifier</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>29</td>
<td>Stacking</td>
<td>116</td>
<td>52,2727</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>30</td>
<td>Vote</td>
<td>116</td>
<td>52,2727</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>31</td>
<td>InputMappedClassifier</td>
<td>116</td>
<td>52,2727</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>32</td>
<td>CSForest</td>
<td>116</td>
<td>52,2727</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>33</td>
<td>Logistic</td>
<td>111</td>
<td>50,4545</td>
<td>0,514</td>
<td>0,505</td>
</tr>
<tr>
<td>34</td>
<td>OLM</td>
<td>97</td>
<td>44,0909</td>
<td>?</td>
<td>0,441</td>
</tr>
<tr>
<td>35</td>
<td>VFI</td>
<td>57</td>
<td>25,9090</td>
<td>?</td>
<td>0,259</td>
</tr>
</tbody>
</table>

C. GKO, BOA ve HBO-BPBÇ Algoritmalarının 100 Popülasyon 200 İterasyon Parametreleri ile Ateizm Veri Seti’ne Uygulanmasından Elde Edilen Sonuç

GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 100 popülasyon ve 200 iterasyon parametreleri ile Ateizm veri setine uygulanması sonucunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.18’de GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar. WEKA’dan yer alan algoritmalar kullanılarak elde edilen sonuçlar ile karşılaştırıldığında GKO %64,5455 doğru sınıflandırma oranı ile birinci, HBO-BPBÇ %64,0909 ile ikinci, BOA %60,9090 ile beşinci sıradan yer almıştır. Ayrıca BOA’nın standart sapması 3,98, HBO-BPBÇ’inin 4,63 ve GKO’nun ise 7,63 olmuştur. Şekil 7.9’da algoritmaların çalışmaları sırasında 200 iterasyon boyunca uygunluk değerinin değişimi gösterilmiştir.
Şekil 7.9. Ateizm Veri Seti’ni kullanılarak GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 200 iteration parametrelerindeki uygunluk değeri değişimi

Tablo 7.19. GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 200 iteration parametreleri ile ateizm veri setine uygulanması sonucunda elde edilen başarı sırası.

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandırılan Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütı</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 GKO</td>
<td>142</td>
<td>64,5455</td>
<td>0,583</td>
<td>0,563</td>
<td>0,567</td>
</tr>
<tr>
<td>2 HBO-BPBÇ</td>
<td>141</td>
<td>64,0909</td>
<td>0,647</td>
<td>0,564</td>
<td>0,584</td>
</tr>
<tr>
<td>3 BOA</td>
<td>139</td>
<td>63,1818</td>
<td>0,648</td>
<td>0,503</td>
<td>0,511</td>
</tr>
<tr>
<td>4 RandomForest</td>
<td>138</td>
<td>62,7273</td>
<td>0,622</td>
<td>0,627</td>
<td>0,618</td>
</tr>
<tr>
<td>5 NaiveBayes</td>
<td>135</td>
<td>61,3636</td>
<td>0,647</td>
<td>0,614</td>
<td>0,557</td>
</tr>
<tr>
<td>6 RandomTree</td>
<td>130</td>
<td>59,0909</td>
<td>0,585</td>
<td>0,591</td>
<td>0,587</td>
</tr>
<tr>
<td>7 MultilayerPerceptronCS</td>
<td>128</td>
<td>58,1818</td>
<td>0,620</td>
<td>0,582</td>
<td>0,509</td>
</tr>
<tr>
<td>8 SMO</td>
<td>128</td>
<td>58,1818</td>
<td>0,575</td>
<td>0,582</td>
<td>0,541</td>
</tr>
<tr>
<td>9 OrdinalClassClassifier</td>
<td>128</td>
<td>58,1818</td>
<td>?</td>
<td>0,582</td>
<td>?</td>
</tr>
<tr>
<td>10 QDA</td>
<td>127</td>
<td>57,7273</td>
<td>0,586</td>
<td>0,577</td>
<td>0,504</td>
</tr>
<tr>
<td>11 LDA</td>
<td>126</td>
<td>57,2727</td>
<td>0,562</td>
<td>0,573</td>
<td>0,565</td>
</tr>
<tr>
<td>12 ExtraTree</td>
<td>126</td>
<td>57,2727</td>
<td>0,570</td>
<td>0,573</td>
<td>0,569</td>
</tr>
<tr>
<td>13 Bagging</td>
<td>125</td>
<td>56,8182</td>
<td>0,732</td>
<td>0,568</td>
<td>0,455</td>
</tr>
<tr>
<td>14 WISARD</td>
<td>125</td>
<td>56,8182</td>
<td>0,589</td>
<td>0,568</td>
<td>0,492</td>
</tr>
<tr>
<td>15 SimpleLogistic</td>
<td>124</td>
<td>56,3636</td>
<td>0,594</td>
<td>0,564</td>
<td>0,478</td>
</tr>
<tr>
<td>16 IterativeClassifierOptimize</td>
<td>122</td>
<td>55,4545</td>
<td>0,592</td>
<td>0,555</td>
<td>0,446</td>
</tr>
<tr>
<td>17 RBFNetwork</td>
<td>121</td>
<td>55</td>
<td>0,492</td>
<td>0,550</td>
<td>0,503</td>
</tr>
<tr>
<td>18 MODLEM</td>
<td>121</td>
<td>55</td>
<td>0,515</td>
<td>0,550</td>
<td>0,464</td>
</tr>
<tr>
<td>19 IBk</td>
<td>119</td>
<td>54,0909</td>
<td>0,563</td>
<td>0,541</td>
<td>0,532</td>
</tr>
<tr>
<td>20 DecisionTable</td>
<td>119</td>
<td>54,0909</td>
<td>0,430</td>
<td>0,541</td>
<td>0,422</td>
</tr>
<tr>
<td>No.</td>
<td>Algoritma</td>
<td>ביע</td>
<td>% değerleri</td>
<td>% değerleri</td>
<td>% değerleri</td>
</tr>
<tr>
<td>-----</td>
<td>----------------------</td>
<td>----</td>
<td>------------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>21</td>
<td>IB1</td>
<td>117</td>
<td>53,181</td>
<td>0,494</td>
<td>0,532</td>
</tr>
<tr>
<td>22</td>
<td>MultiClassClassifier</td>
<td>117</td>
<td>53,181</td>
<td>0,528</td>
<td>0,532</td>
</tr>
<tr>
<td>23</td>
<td>HyperPipes</td>
<td>117</td>
<td>53,181</td>
<td>0,525</td>
<td>0,532</td>
</tr>
<tr>
<td>24</td>
<td>MultiScheme</td>
<td>116</td>
<td>52,727</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>25</td>
<td>LibSVM</td>
<td>116</td>
<td>52,727</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>26</td>
<td>ReseslibKnn</td>
<td>116</td>
<td>52,727</td>
<td>0,598</td>
<td>0,527</td>
</tr>
<tr>
<td>27</td>
<td>REPTree</td>
<td>116</td>
<td>52,727</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>28</td>
<td>ClojureClassifier</td>
<td>116</td>
<td>52,727</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>29</td>
<td>Stacking</td>
<td>116</td>
<td>52,272</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>30</td>
<td>Vote</td>
<td>116</td>
<td>52,272</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>31</td>
<td>InputMappedClassifier</td>
<td>116</td>
<td>52,272</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>32</td>
<td>CSForest</td>
<td>116</td>
<td>52,272</td>
<td>?</td>
<td>0,527</td>
</tr>
<tr>
<td>33</td>
<td>Logistic</td>
<td>111</td>
<td>50,454</td>
<td>0,514</td>
<td>0,505</td>
</tr>
<tr>
<td>34</td>
<td>OLM</td>
<td>97</td>
<td>44,090</td>
<td>?</td>
<td>0,441</td>
</tr>
<tr>
<td>35</td>
<td>VFI</td>
<td>57</td>
<td>25,909</td>
<td>?</td>
<td>0,259</td>
</tr>
</tbody>
</table>

### D. GKO, BOA ve HBO-BPBÇ Algoritmalarının 200 Popülasyon 100 İterasyon Parametreleri ile Ateizm Veri Seti’ne Uygulanmasından Elde Edilen Sonuç

GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 200 popülasyon ve 100 iterasyon parametreleri ile Ateizm veri setine uygulanması sonucunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.19’da GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar, WEKA’da yer alan algoritmalar kullanılarak elde edilen sonuçlar ile karşlaştırıldığında HBO-BPBÇ %62,7273 doğru sınıflandırma oranı ile birinci, BOA %60,9090 ile dördüncü ve GKO ise %60 ile beşinci sırada yer almıştır. Şekil 7.10’da algoritmaların çalışmaları sırasında 100 iterasyon boyunca uygunluk değerinin değişimi gösterilmiştir.
Tablo 7.20. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 100 iterasyon parametreleri ile ateizm veri setine uygulanması sonucunda elde edilen başarı sırası.
<table>
<thead>
<tr>
<th>No</th>
<th>Algoritma</th>
<th>Popülasyon</th>
<th>Iterasyon</th>
<th>Orijinal Değer</th>
<th>BOA Değer</th>
<th>HBO-BPBÇ Değer</th>
</tr>
</thead>
<tbody>
<tr>
<td>24</td>
<td>MultiScheme</td>
<td>116</td>
<td>52,727</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>25</td>
<td>LibSVM</td>
<td>116</td>
<td>52,727</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>26</td>
<td>RsetlibKnn</td>
<td>116</td>
<td>52,727</td>
<td>0,598</td>
<td>0,527</td>
<td>0,429</td>
</tr>
<tr>
<td>27</td>
<td>REPTree</td>
<td>116</td>
<td>52,727</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>28</td>
<td>ClojureClassifier</td>
<td>116</td>
<td>52,727</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>29</td>
<td>Stacking</td>
<td>116</td>
<td>52,272</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>30</td>
<td>Vote</td>
<td>116</td>
<td>52,272</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>31</td>
<td>InputMappedClassifier</td>
<td>116</td>
<td>52,272</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>32</td>
<td>CSForest</td>
<td>116</td>
<td>52,272</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>33</td>
<td>Logistic</td>
<td>111</td>
<td>50,4545</td>
<td>0,514</td>
<td>0,505</td>
<td>0,508</td>
</tr>
<tr>
<td>34</td>
<td>OLM</td>
<td>97</td>
<td>44,0909</td>
<td>?</td>
<td>0,441</td>
<td>?</td>
</tr>
<tr>
<td>35</td>
<td>VFI</td>
<td>57</td>
<td>25,9090</td>
<td>?</td>
<td>0,259</td>
<td>?</td>
</tr>
</tbody>
</table>

**E. GKO, BOA ve HBO-BPBÇ Algoritmalarının 200 Popülasyon 150 İterasyon Parametreleri ile Ateizm Veri Seti’ne Uygulanmasından Elde Edilen Sonuç**

GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 200 popülasyon ve 150 iterasyon parametreleri ile Ateizm veri setine uygulanması sonucunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.20’de GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar, WEKA’da yer alan algoritmalar kullanılarak elde edilen sonuçlar ile karşılaştırıldığında GKO %61,3636 doğru sınıflandırma oranı ile ikinci, BOA %61,3636 ile üçüncü ve HBO-BPBÇ ise %60,4545 ile beşinci sıradadır yer almıştır. Şekil 7.11’de algoritmaların çalışmaları sırasında 150 iterasyon boyunca uygunluk değeri değişimi gösterilmiştir.

![Şekil 7.11. Ateizm Veri Seti’ni kullanan GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 150 iterasyon parametrelerindeki uygunluk değeri değişimi](image-url)
Tablo 7.21. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 150 iterasyon parametreleri ile ateizm veri setine uygulanması sonucunda elde edilen başarı sırası.

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandırılan Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasyet</th>
<th>Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 RandomForest</td>
<td>138</td>
<td>62,7273</td>
<td>0,622</td>
<td>0,627</td>
<td>0,618</td>
</tr>
<tr>
<td>2 GKO</td>
<td>135</td>
<td>61,3636</td>
<td>0,583</td>
<td>0,557</td>
<td>0,565</td>
</tr>
<tr>
<td>3 BOA</td>
<td>135</td>
<td>61,3636</td>
<td>0,620</td>
<td>0,511</td>
<td>0,498</td>
</tr>
<tr>
<td>4 NaiveBayes</td>
<td>135</td>
<td>61,3636</td>
<td>0,647</td>
<td>0,614</td>
<td>0,557</td>
</tr>
<tr>
<td>5 HBO-BPBÇ</td>
<td>133</td>
<td>60,4545</td>
<td>0,555</td>
<td>0,530</td>
<td>0,519</td>
</tr>
<tr>
<td>6 RandomTree</td>
<td>130</td>
<td>59,0909</td>
<td>0,585</td>
<td>0,591</td>
<td>0,587</td>
</tr>
<tr>
<td>7 MultilayerPerceptronCS</td>
<td>128</td>
<td>58,1818</td>
<td>0,620</td>
<td>0,582</td>
<td>0,509</td>
</tr>
<tr>
<td>8 SMO</td>
<td>128</td>
<td>58,1818</td>
<td>0,575</td>
<td>0,582</td>
<td>0,541</td>
</tr>
<tr>
<td>9 OrdinalClassClassifier</td>
<td>128</td>
<td>58,1818</td>
<td>0,582</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>10 QDA</td>
<td>127</td>
<td>57,7273</td>
<td>0,586</td>
<td>0,577</td>
<td>0,504</td>
</tr>
<tr>
<td>11 LDA</td>
<td>126</td>
<td>57,2727</td>
<td>0,562</td>
<td>0,573</td>
<td>0,565</td>
</tr>
<tr>
<td>12 ExtraTree</td>
<td>126</td>
<td>57,2727</td>
<td>0,570</td>
<td>0,573</td>
<td>0,569</td>
</tr>
<tr>
<td>13 Bagging</td>
<td>125</td>
<td>56,8182</td>
<td>0,732</td>
<td>0,568</td>
<td>0,455</td>
</tr>
<tr>
<td>14 WISARD</td>
<td>125</td>
<td>56,8182</td>
<td>0,589</td>
<td>0,568</td>
<td>0,492</td>
</tr>
<tr>
<td>15 SimpleLogistic</td>
<td>124</td>
<td>56,3636</td>
<td>0,594</td>
<td>0,564</td>
<td>0,478</td>
</tr>
<tr>
<td>16 IterativeClassifierOptimizer</td>
<td>122</td>
<td>55,4545</td>
<td>0,592</td>
<td>0,555</td>
<td>0,446</td>
</tr>
<tr>
<td>17 RBFNetwork</td>
<td>121</td>
<td>55</td>
<td>0,492</td>
<td>0,550</td>
<td>0,503</td>
</tr>
<tr>
<td>18 MODLEM</td>
<td>121</td>
<td>55</td>
<td>0,515</td>
<td>0,550</td>
<td>0,464</td>
</tr>
<tr>
<td>19 IBk</td>
<td>119</td>
<td>54,0909</td>
<td>0,563</td>
<td>0,541</td>
<td>0,532</td>
</tr>
<tr>
<td>20 DecisionTable</td>
<td>119</td>
<td>54,0909</td>
<td>0,430</td>
<td>0,541</td>
<td>0,422</td>
</tr>
<tr>
<td>21 IB1</td>
<td>117</td>
<td>53,1818</td>
<td>0,494</td>
<td>0,532</td>
<td>0,423</td>
</tr>
<tr>
<td>22 MultiClassClassifier</td>
<td>117</td>
<td>53,1818</td>
<td>0,528</td>
<td>0,532</td>
<td>0,529</td>
</tr>
<tr>
<td>23 HyperPipes</td>
<td>117</td>
<td>53,1818</td>
<td>0,525</td>
<td>0,532</td>
<td>0,406</td>
</tr>
<tr>
<td>24 MultiScheme</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>25 LibSVM</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>26 RseslibKnn</td>
<td>116</td>
<td>52,7273</td>
<td>0,598</td>
<td>0,527</td>
<td>0,429</td>
</tr>
<tr>
<td>27 REPTree</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>28 ClojureClassifier</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>29 Stacking</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>30 Vote</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>31 InputMappedClassifier</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>32 CSForest</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>33 Logistic</td>
<td>111</td>
<td>50,4545</td>
<td>0,514</td>
<td>0,505</td>
<td>0,508</td>
</tr>
<tr>
<td>34 OLM</td>
<td>97</td>
<td>44,0909</td>
<td>?</td>
<td>0,441</td>
<td>?</td>
</tr>
<tr>
<td>35 VFI</td>
<td>57</td>
<td>25,9090</td>
<td>?</td>
<td>0,259</td>
<td>?</td>
</tr>
</tbody>
</table>
F. GKO, BOA ve HBO-BPBÇ Algoritmalarının 200 Popülasyon 200 İterasyon Parametreleri ile Ateizm Veri Seti’ne Uygulanmasından Elde Edilen Sonuç

GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 200 popülasyon ve 200 iterasyon parametreleri ile Ateizm veri setine uygulanması sonunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.21’de GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar, WEKA’da yer alan algoritmalar kullanılarak elde edilen sonuçlar ile karşılaştırıldığında HBO-BPBÇ %63,1818 doğru sınıflandırma oranı ile birinci, GKO %61,3636 ile üçüncü ve BOA ise %60,9090 ile beşinci sıradadır yer almıştır. Ayrıca BOA’nın standart sapması 4,83, HBO-BPBÇ’nin 2,80 ve GKO’nun ise 8,81 olmuştur. Şekil 7.12’de algoritmaların çalışmaları sırasında 200 iterasyon boyunca uygunluk değerinin değişimi gösterilmiştir.

![Şekil 7.12. Ateizm Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 200 iterasyon parametrelerindeki uygunluk değerinin değişimi](image)

Tablo 7.22. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 200 iterasyon parametreleri ile ateizm veri setine uygulanması sonucunda elde edilen başarı sırası.

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandıran Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 HBO-BPBÇ</td>
<td>139</td>
<td>63,1818</td>
<td>0,631</td>
<td>0,590</td>
<td>0,605</td>
</tr>
<tr>
<td>2 RandomForest</td>
<td>138</td>
<td>62,7273</td>
<td>0,622</td>
<td>0,627</td>
<td>0,618</td>
</tr>
<tr>
<td>3 BOA</td>
<td>135</td>
<td>61,3636</td>
<td>0,606</td>
<td>0,526</td>
<td>0,545</td>
</tr>
<tr>
<td>4 GKO</td>
<td>135</td>
<td>61,3636</td>
<td>0,575</td>
<td>0,565</td>
<td>0,567</td>
</tr>
<tr>
<td>5 NaiveBayes</td>
<td>135</td>
<td>61,3636</td>
<td>0,647</td>
<td>0,614</td>
<td>0,557</td>
</tr>
<tr>
<td>6 RandomTree</td>
<td>130</td>
<td>59,0909</td>
<td>0,585</td>
<td>0,591</td>
<td>0,587</td>
</tr>
<tr>
<td>No.</td>
<td>Adı</td>
<td>Tweet</td>
<td>40400 Defa Çağrılma Sayısına Göre Karşılaştırılması</td>
<td>Uygunluk Fonksiyonunun Eşit Çağrılma Sayısına Göre Karşılaştırılması</td>
<td></td>
</tr>
<tr>
<td>-----</td>
<td>-------------------------</td>
<td>--------</td>
<td>----------------------------------------------------</td>
<td>---------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>MultilayerPerceptronCS</td>
<td>128</td>
<td>58,1818</td>
<td>0,620</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>SMO</td>
<td>128</td>
<td>58,1818</td>
<td>0,575</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>OrdinalClassClassifier</td>
<td>128</td>
<td>58,1818</td>
<td>?</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>QDA</td>
<td>127</td>
<td>57,7277</td>
<td>0,586</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>LDA</td>
<td>126</td>
<td>57,7277</td>
<td>0,562</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>ExtraTree</td>
<td>126</td>
<td>57,7277</td>
<td>0,570</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Bagging</td>
<td>125</td>
<td>56,8182</td>
<td>0,732</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>WISARD</td>
<td>125</td>
<td>56,8182</td>
<td>0,589</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>SimpleLogistic</td>
<td>124</td>
<td>56,3636</td>
<td>0,594</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>IterativeClassifierOptimizer</td>
<td>122</td>
<td>55,4545</td>
<td>0,592</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>RBFNetwork</td>
<td>121</td>
<td>55</td>
<td>0,492</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>MODLEM</td>
<td>121</td>
<td>55</td>
<td>0,515</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>IBk</td>
<td>119</td>
<td>54,0909</td>
<td>0,563</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>DecisionTable</td>
<td>119</td>
<td>54,0909</td>
<td>0,430</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>IB1</td>
<td>117</td>
<td>53,1818</td>
<td>0,494</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>MultiClassClassifier</td>
<td>117</td>
<td>53,1818</td>
<td>0,528</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>HyperPipes</td>
<td>117</td>
<td>53,1818</td>
<td>0,525</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>MultiScheme</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>LibSVM</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>RseslibKnn</td>
<td>116</td>
<td>52,7273</td>
<td>0,598</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>REPTree</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>ClojureClassifier</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>Stacking</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>Vote</td>
<td>116</td>
<td>52,2727</td>
<td>?</td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>InputMappedClassifier</td>
<td>116</td>
<td>52,2727</td>
<td>?</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>CSForest</td>
<td>116</td>
<td>52,2727</td>
<td>?</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>Logistic</td>
<td>111</td>
<td>50,4545</td>
<td>0,514</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>OLM</td>
<td>97</td>
<td>44,0909</td>
<td>?</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>VFI</td>
<td>57</td>
<td>25,9090</td>
<td>?</td>
<td></td>
</tr>
</tbody>
</table>

G. Ateşiz Veri Seti Kullanılarak Algoritmaların Uygunluk Fonksiyonlarının Eşit Sayıda Çağrılma Sayısına Göre Karşılaştırılması

Uygunluk fonksiyonunun 40400 defa çağrılma değeri baz alınarak BOA ve HBO-BPBÇ algoritmalarının sonuçları WEKA’dan yer alan algoritmalar kullanılarak elde edilen sonuçlar ile karşılaştırıldığında HBO-BPBÇ %62,7272 doğru sınıflandırma oranı ile ikinci, BOA %61,3636 ile üçüncü sıradaki yer almıştır. Aşağıda Tablo 7.22’de sıralama sonuçları gösterilmiştir. Bu sonuçlara göre HBO-BPBÇ algoritmasyonun uygunluk fonksiyonun eşit çağrılma sayısı baz alınarak yapılan testlerde de başarılı olduğu görülmüştür.
Tablo 7.23. BOA ve HBO-BPBÇ algoritmaların ateizm veri setinde uygunluk fonksiyonlarının eşit sayıda çağrılması sonuçları

<table>
<thead>
<tr>
<th></th>
<th>Algoritma</th>
<th>Doğru Sınıflandıran Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>RandomForest</td>
<td>138</td>
<td>62,7273</td>
<td>0,622</td>
<td>0,627</td>
<td>0,618</td>
</tr>
<tr>
<td>2</td>
<td>HBO-BPBÇ</td>
<td>137</td>
<td>62,7272</td>
<td>0,632</td>
<td>0,512</td>
<td>0,532</td>
</tr>
<tr>
<td>3</td>
<td>BOA</td>
<td>135</td>
<td>61,3636</td>
<td>0,614</td>
<td>0,545</td>
<td>0,558</td>
</tr>
<tr>
<td>4</td>
<td>GKO</td>
<td>135</td>
<td>61,3636</td>
<td>0,579</td>
<td>0,567</td>
<td>0,571</td>
</tr>
<tr>
<td>5</td>
<td>NaiveBayes</td>
<td>135</td>
<td>61,3636</td>
<td>0,647</td>
<td>0,614</td>
<td>0,557</td>
</tr>
<tr>
<td>6</td>
<td>RandomTree</td>
<td>130</td>
<td>59,0909</td>
<td>0,585</td>
<td>0,591</td>
<td>0,587</td>
</tr>
<tr>
<td>7</td>
<td>MultilayerPerceptronCS</td>
<td>128</td>
<td>58,1818</td>
<td>0,620</td>
<td>0,582</td>
<td>0,509</td>
</tr>
<tr>
<td>8</td>
<td>SMO</td>
<td>128</td>
<td>58,1818</td>
<td>0,575</td>
<td>0,582</td>
<td>0,541</td>
</tr>
<tr>
<td>9</td>
<td>OrdinalClassClassifier</td>
<td>128</td>
<td>58,1818</td>
<td>?</td>
<td>0,582</td>
<td>?</td>
</tr>
<tr>
<td>10</td>
<td>QDA</td>
<td>127</td>
<td>57,7273</td>
<td>0,586</td>
<td>0,577</td>
<td>0,504</td>
</tr>
<tr>
<td>11</td>
<td>LDA</td>
<td>126</td>
<td>57,2727</td>
<td>0,562</td>
<td>0,573</td>
<td>0,565</td>
</tr>
<tr>
<td>12</td>
<td>ExtraTree</td>
<td>126</td>
<td>57,2727</td>
<td>0,570</td>
<td>0,573</td>
<td>0,569</td>
</tr>
<tr>
<td>13</td>
<td>Bagging</td>
<td>125</td>
<td>56,8182</td>
<td>0,732</td>
<td>0,568</td>
<td>0,455</td>
</tr>
<tr>
<td>14</td>
<td>WISARD</td>
<td>125</td>
<td>56,8182</td>
<td>0,589</td>
<td>0,568</td>
<td>0,492</td>
</tr>
<tr>
<td>15</td>
<td>SimpleLogistic</td>
<td>124</td>
<td>56,3636</td>
<td>0,594</td>
<td>0,564</td>
<td>0,478</td>
</tr>
<tr>
<td>16</td>
<td>IterativeClassifierOptimizer</td>
<td>122</td>
<td>55,4545</td>
<td>0,592</td>
<td>0,555</td>
<td>0,446</td>
</tr>
<tr>
<td>17</td>
<td>RBFNetwork</td>
<td>121</td>
<td>55</td>
<td>0,492</td>
<td>0,550</td>
<td>0,503</td>
</tr>
<tr>
<td>18</td>
<td>MODLEM</td>
<td>121</td>
<td>55</td>
<td>0,515</td>
<td>0,550</td>
<td>0,464</td>
</tr>
<tr>
<td>19</td>
<td>IBk</td>
<td>119</td>
<td>54,0909</td>
<td>0,563</td>
<td>0,541</td>
<td>0,532</td>
</tr>
<tr>
<td>20</td>
<td>DecisionTable</td>
<td>119</td>
<td>54,0909</td>
<td>0,430</td>
<td>0,541</td>
<td>0,422</td>
</tr>
<tr>
<td>21</td>
<td>IB1</td>
<td>117</td>
<td>53,1818</td>
<td>0,494</td>
<td>0,532</td>
<td>0,423</td>
</tr>
<tr>
<td>22</td>
<td>MultiClassClassifier</td>
<td>117</td>
<td>53,1818</td>
<td>0,528</td>
<td>0,532</td>
<td>0,529</td>
</tr>
<tr>
<td>23</td>
<td>HyperPipes</td>
<td>117</td>
<td>53,1818</td>
<td>0,525</td>
<td>0,532</td>
<td>0,406</td>
</tr>
<tr>
<td>24</td>
<td>MultiScheme</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>25</td>
<td>LibSVM</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>26</td>
<td>RseslibKnn</td>
<td>116</td>
<td>52,7273</td>
<td>0,598</td>
<td>0,527</td>
<td>0,429</td>
</tr>
<tr>
<td>27</td>
<td>REPTree</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>28</td>
<td>ClojureClassifier</td>
<td>116</td>
<td>52,7273</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>29</td>
<td>Stacking</td>
<td>116</td>
<td>52,2727</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>30</td>
<td>Vote</td>
<td>116</td>
<td>52,2727</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>31</td>
<td>InputMappedClassifier</td>
<td>116</td>
<td>52,2727</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>32</td>
<td>CSForest</td>
<td>116</td>
<td>52,2727</td>
<td>?</td>
<td>0,527</td>
<td>?</td>
</tr>
<tr>
<td>33</td>
<td>Logistic</td>
<td>111</td>
<td>50,4545</td>
<td>0,514</td>
<td>0,505</td>
<td>0,508</td>
</tr>
<tr>
<td>34</td>
<td>OLM</td>
<td>97</td>
<td>44,0909</td>
<td>?</td>
<td>0,441</td>
<td>?</td>
</tr>
<tr>
<td>35</td>
<td>VFI</td>
<td>57</td>
<td>25,9090</td>
<td>?</td>
<td>0,259</td>
<td>?</td>
</tr>
</tbody>
</table>
7.1.3. Kürtajın Yasallaşması Veri Seti’nden Elde Edilen Sonuçlar

Bu alt bölümde GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile popülasyon sayıları 100 ve 200 olarak seçilerek üzerinde sabit tutularak, iterasyon sayıları 100, 150 ve 200 olarak sırası ile arttırılmış ve böylece popülasyon sayısının artması ve buna paralel olarak iterasyon sayılarının arttırılması sonucunda Kürtajın yasallaşması veri setinden elde edilen sonuçlar gösterilmiştir. Ayrıca en iyi sonucun elde edildiği popülasyon ve iterasyon sonuçları sabit tutularak diğer parametre değerlerinin başarı sonucuna etkisi test edilmiştir. Tüm bu sonuçlar Kürtajın yasallaşması veri setinin WEKA programı içerisinde bulunan 32 farklı sınıflandırma algoritması ile değerlendirilmesi ile elde edilen sonuçlarla karşılaştırılmıştır. Aşağıda WEKA programından elde edilen sınıflandırma sonuçları Tablo 7.24’té gösterilmiştir. Ayrıca GKO, BOA ve HBO-BPBÇ algoritmalarında kullanılan benzerlik oranı ve uygunluk fonksiyonunda kullanılan katsayı değerleri Tablo 7.23’té gösterilmiştir.

Tablo 7.24. Kürtajın yasallaşması veri seti için kullanılan katsayılar

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Benzerlik</th>
<th>Uygunluk Fonksiyonu</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>benzerlik</td>
<td>k₁</td>
</tr>
<tr>
<td>GKO</td>
<td>0,1</td>
<td>0,9</td>
</tr>
<tr>
<td>BOA</td>
<td>0,01</td>
<td>0,9</td>
</tr>
<tr>
<td>HBO-BPBÇ</td>
<td>0,01</td>
<td>0,9</td>
</tr>
</tbody>
</table>

Tablo 7.25. Weka programı kullanılarak kürtajın yasallaşması veri setinden elde edilen sınıflandırma sonuçları

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandırılan Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 SimpleLogistic</td>
<td>201</td>
<td>71,7857</td>
<td>0,736</td>
<td>0,718</td>
<td>0,718</td>
</tr>
<tr>
<td>2 SMO</td>
<td>191</td>
<td>68,2143</td>
<td>0,696</td>
<td>0,682</td>
<td>0,686</td>
</tr>
<tr>
<td>3 RandomForest</td>
<td>190</td>
<td>67,8571</td>
<td>0,689</td>
<td>0,679</td>
<td>0,667</td>
</tr>
<tr>
<td>4 NaiveBayes</td>
<td>188</td>
<td>67,1429</td>
<td>0,689</td>
<td>0,671</td>
<td>0,677</td>
</tr>
<tr>
<td>5 Bagging</td>
<td>186</td>
<td>66,4286</td>
<td>0,697</td>
<td>0,664</td>
<td>0,663</td>
</tr>
<tr>
<td>6 WISARD</td>
<td>185</td>
<td>66,0714</td>
<td>0,630</td>
<td>0,661</td>
<td>0,619</td>
</tr>
<tr>
<td>7 DecisionTable</td>
<td>184</td>
<td>65,7143</td>
<td>0,624</td>
<td>0,657</td>
<td>0,577</td>
</tr>
<tr>
<td>8 ExtraTree</td>
<td>183</td>
<td>65,3571</td>
<td>0,647</td>
<td>0,654</td>
<td>0,641</td>
</tr>
<tr>
<td>9 IterativeClassifierOptimizer</td>
<td>182</td>
<td>65</td>
<td>0,718</td>
<td>0,650</td>
<td>0,544</td>
</tr>
<tr>
<td>10 HyperPipes</td>
<td>181</td>
<td>64,6429</td>
<td>0,624</td>
<td>0,646</td>
<td>0,545</td>
</tr>
<tr>
<td>11 MultilayerPerceptronCS</td>
<td>180</td>
<td>64,2857</td>
<td>0,655</td>
<td>0,643</td>
<td>0,648</td>
</tr>
</tbody>
</table>
A. GKO, BOA ve HBO-BPBÇ Algoritmalarının 100 Popülasyon 100 İterasyon Parametreleri ile Kürtağın Yasallaşması Veri Seti’ne Uygulanmasından Elde Edilen Sonuç

GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 100 popülasyon ve 100 iterasyon parametreleri ile Kürtağın yasallaşması veri setine uygulanması sonucunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.25’de GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar, WEKA’da yer alan algoritmalar kullanarak elde edilen sonuçlar ile karşılaştırıldığında BOA %67,8571 doğru sınıflandırma oranı ile üçüncü, HBO-BPBÇ %67,1429 ile beşinci ve GKO ise %63,5714 ile ondördüncü sırada yer almıştır. Şekil 7.23’te algoritmaların çalışmalarını sırasında 100 iterasyon boyunca uygunluk değerinin değişimi gösterilmiştir.
Şekil 7.13. Kürtağın Yasallaşması Veri Seti’ni kullanarak GKO, BOA ve HBO-BPÇ algoritmalarının 100 popülasyon 100 iterasyon parametrelerindeki uygulan değer değişimi

Tablo 7.26. GKO, BOA ve HBO-BPÇ algoritmalarının 100 popülasyon 100 iterasyon parametreleri ile kürtağın yasallaşması veri setine uygulanması sonucunda elde edilen başarı sırası

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğrulandırılan Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Olçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>SimpleLogistic</td>
<td>201</td>
<td>71,7857</td>
<td>0,736</td>
<td>0,718</td>
<td>0,718</td>
</tr>
<tr>
<td>SMO</td>
<td>191</td>
<td>68,2143</td>
<td>0,696</td>
<td>0,682</td>
<td>0,686</td>
</tr>
<tr>
<td>BOA</td>
<td>190</td>
<td>67,8571</td>
<td>0,625</td>
<td>0,627</td>
<td>0,621</td>
</tr>
<tr>
<td>RandomForest</td>
<td>190</td>
<td>67,8571</td>
<td>0,689</td>
<td>0,679</td>
<td>0,667</td>
</tr>
<tr>
<td>HBO-BPÇ</td>
<td>188</td>
<td>67,1429</td>
<td>0,606</td>
<td>0,604</td>
<td>0,604</td>
</tr>
<tr>
<td>NaiveBayes</td>
<td>188</td>
<td>67,1429</td>
<td>0,689</td>
<td>0,671</td>
<td>0,677</td>
</tr>
<tr>
<td>Bagging</td>
<td>186</td>
<td>66,4286</td>
<td>0,697</td>
<td>0,664</td>
<td>0,663</td>
</tr>
<tr>
<td>WISARD</td>
<td>185</td>
<td>66,0714</td>
<td>0,630</td>
<td>0,661</td>
<td>0,619</td>
</tr>
<tr>
<td>DecisionTable</td>
<td>184</td>
<td>65,7143</td>
<td>0,624</td>
<td>0,657</td>
<td>0,577</td>
</tr>
<tr>
<td>ExtraTree</td>
<td>183</td>
<td>65,3571</td>
<td>0,647</td>
<td>0,654</td>
<td>0,641</td>
</tr>
<tr>
<td>IterativeClassifierOptimizer</td>
<td>182</td>
<td>65</td>
<td>0,718</td>
<td>0,650</td>
<td>0,544</td>
</tr>
<tr>
<td>HyperPipes</td>
<td>181</td>
<td>64,6429</td>
<td>0,624</td>
<td>0,646</td>
<td>0,545</td>
</tr>
<tr>
<td>MultilayerPerceptronCS</td>
<td>180</td>
<td>64,2857</td>
<td>0,655</td>
<td>0,643</td>
<td>0,648</td>
</tr>
<tr>
<td>GKO</td>
<td>178</td>
<td>63,5714</td>
<td>0,469</td>
<td>0,453</td>
<td>0,430</td>
</tr>
<tr>
<td>LDA</td>
<td>178</td>
<td>63,5714</td>
<td>0,657</td>
<td>0,636</td>
<td>0,643</td>
</tr>
<tr>
<td>CSForest</td>
<td>178</td>
<td>63,5714</td>
<td>0,648</td>
<td>0,636</td>
<td>0,618</td>
</tr>
<tr>
<td>QDA</td>
<td>178</td>
<td>63,5714</td>
<td>0,577</td>
<td>0,636</td>
<td>0,579</td>
</tr>
<tr>
<td>IB1</td>
<td>178</td>
<td>63,5714</td>
<td>0,591</td>
<td>0,636</td>
<td>0,575</td>
</tr>
<tr>
<td>MODLEM</td>
<td>177</td>
<td>63,2143</td>
<td>0,593</td>
<td>0,632</td>
<td>0,594</td>
</tr>
<tr>
<td>ClojureClassifier</td>
<td>176</td>
<td>62,8571</td>
<td>?</td>
<td>0,629</td>
<td>?</td>
</tr>
<tr>
<td>InputMappedClassifier</td>
<td>176</td>
<td>62,8571</td>
<td>?</td>
<td>0,629</td>
<td>?</td>
</tr>
<tr>
<td></td>
<td>Algoritma</td>
<td>#</td>
<td>%100 Popülasyon</td>
<td>%150 İterasyon</td>
<td>%100 Popülasyon</td>
</tr>
<tr>
<td>----</td>
<td>-------------------------</td>
<td>----</td>
<td>-----------------</td>
<td>----------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>22</td>
<td>LibSVM</td>
<td>176</td>
<td>62,8571</td>
<td>0,629</td>
<td>62,8571</td>
</tr>
<tr>
<td>23</td>
<td>MultiScheme</td>
<td>176</td>
<td>62,8571</td>
<td>0,629</td>
<td>62,8571</td>
</tr>
<tr>
<td>24</td>
<td>Stacking</td>
<td>176</td>
<td>62,8571</td>
<td>0,629</td>
<td>62,8571</td>
</tr>
<tr>
<td>25</td>
<td>Vote</td>
<td>176</td>
<td>62,8571</td>
<td>0,629</td>
<td>62,8571</td>
</tr>
<tr>
<td>26</td>
<td>MultiClassClassifier</td>
<td>170</td>
<td>60,7143</td>
<td>0,610</td>
<td>60,7143</td>
</tr>
<tr>
<td>27</td>
<td>IBk</td>
<td>169</td>
<td>60,3571</td>
<td>0,604</td>
<td>60,3571</td>
</tr>
<tr>
<td>28</td>
<td>RandomForest</td>
<td>169</td>
<td>60,3571</td>
<td>0,599</td>
<td>60,3571</td>
</tr>
<tr>
<td>29</td>
<td>REPTree</td>
<td>160</td>
<td>57,1429</td>
<td>0,575</td>
<td>57,1429</td>
</tr>
<tr>
<td>30</td>
<td>OrdinalClassClassifier</td>
<td>159</td>
<td>56,7857</td>
<td>0,578</td>
<td>56,7857</td>
</tr>
<tr>
<td>31</td>
<td>RseslibKnn</td>
<td>156</td>
<td>55,7143</td>
<td>0,578</td>
<td>55,7143</td>
</tr>
<tr>
<td>32</td>
<td>RBFNetwork</td>
<td>150</td>
<td>53,5714</td>
<td>0,560</td>
<td>53,5714</td>
</tr>
<tr>
<td>33</td>
<td>OLM</td>
<td>149</td>
<td>53,2143</td>
<td>0,479</td>
<td>53,2143</td>
</tr>
<tr>
<td>34</td>
<td>Logistic</td>
<td>141</td>
<td>50,3571</td>
<td>0,523</td>
<td>50,3571</td>
</tr>
<tr>
<td>35</td>
<td>VFI</td>
<td>53</td>
<td>18,9286</td>
<td>0,189</td>
<td>18,9286</td>
</tr>
</tbody>
</table>

B. GKO, BOA ve HBO-BPBÇ Algoritmalarının 100 Popülasyon 150 İterasyon Parametreleri ile Kürtajın Yasallaşması Veri Seti’ne Uygulanmasından Elde Edilen Sonuç

GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 100 popülasyon ve 150 iterasyon parametreleri ile Kürtajın yaşallaşması veri setine uygulanması sonucunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.26’da GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar, WEKA’da yer alan algoritmalar kullanılarak elde edilen sonuçlar ile karşılaştırıldığında HBO-BPBÇ %67,8571 doğru sınıflandırma oranı ile üçüncü, BOA %66,0714 ile yedinci ve GKO ise %63,2143 ile onsekizinci sıradada yer almıştır. Şekil 7.14’te algoritmaların çalışmaları sırasında 150 iterasyon boyunca uyguluk değerinin değişimi gösterilmiştir.
Şekil 7.14. Kürtajın Yasallaşması Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 150 iterasyon parametrelerindeki uygunluk değeri değişimi

Tablo 7.27. GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 150 iterasyon parametreleri ile kürtaj veri setine uygulanması sonucunda elde edilen başarı sırası.

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandıran Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1  SimpleLogistic</td>
<td>201</td>
<td>71,7857</td>
<td>0,736</td>
<td>0,718</td>
<td>0,718</td>
</tr>
<tr>
<td>2  SMO</td>
<td>191</td>
<td>68,2143</td>
<td>0,696</td>
<td>0,682</td>
<td>0,686</td>
</tr>
<tr>
<td>3  HBO-BPBÇ</td>
<td>190</td>
<td>67,8571</td>
<td>0,614</td>
<td>0,621</td>
<td>0,617</td>
</tr>
<tr>
<td>4  RandomForest</td>
<td>190</td>
<td>67,8571</td>
<td>0,689</td>
<td>0,679</td>
<td>0,667</td>
</tr>
<tr>
<td>5  NaiveBayes</td>
<td>188</td>
<td>67,1429</td>
<td>0,689</td>
<td>0,671</td>
<td>0,677</td>
</tr>
<tr>
<td>6  Bagging</td>
<td>186</td>
<td>66,4286</td>
<td>0,697</td>
<td>0,664</td>
<td>0,663</td>
</tr>
<tr>
<td>7  BOA</td>
<td>185</td>
<td>66,0714</td>
<td>0,575</td>
<td>0,601</td>
<td>0,582</td>
</tr>
<tr>
<td>8  WISARD</td>
<td>185</td>
<td>66,0714</td>
<td>0,630</td>
<td>0,661</td>
<td>0,619</td>
</tr>
<tr>
<td>9  DecisionTable</td>
<td>184</td>
<td>65,7143</td>
<td>0,624</td>
<td>0,657</td>
<td>0,577</td>
</tr>
<tr>
<td>10 ExtraTree</td>
<td>183</td>
<td>65,3571</td>
<td>0,647</td>
<td>0,654</td>
<td>0,641</td>
</tr>
<tr>
<td>11 IterativeClassifierOptimizor</td>
<td>182</td>
<td>65</td>
<td>0,718</td>
<td>0,650</td>
<td>0,544</td>
</tr>
<tr>
<td>12 HyperPipes</td>
<td>181</td>
<td>64,6429</td>
<td>0,624</td>
<td>0,646</td>
<td>0,545</td>
</tr>
<tr>
<td>13 MultilayerPerceptronCS</td>
<td>180</td>
<td>64,2857</td>
<td>0,655</td>
<td>0,643</td>
<td>0,648</td>
</tr>
<tr>
<td>14 LDA</td>
<td>178</td>
<td>63,5714</td>
<td>0,657</td>
<td>0,636</td>
<td>0,643</td>
</tr>
<tr>
<td>15 CSForest</td>
<td>178</td>
<td>63,5714</td>
<td>0,648</td>
<td>0,636</td>
<td>0,618</td>
</tr>
<tr>
<td>16 QDA</td>
<td>178</td>
<td>63,5714</td>
<td>0,577</td>
<td>0,636</td>
<td>0,579</td>
</tr>
<tr>
<td>17 IB1</td>
<td>178</td>
<td>63,5714</td>
<td>0,591</td>
<td>0,636</td>
<td>0,575</td>
</tr>
<tr>
<td>18 GKO</td>
<td>177</td>
<td>63,2143</td>
<td>0,624</td>
<td>0,530</td>
<td>0,485</td>
</tr>
<tr>
<td>19 MODEM</td>
<td>177</td>
<td>63,2143</td>
<td>0,593</td>
<td>0,632</td>
<td>0,594</td>
</tr>
<tr>
<td>20 ClojureClassifier</td>
<td>176</td>
<td>62,8571</td>
<td>?</td>
<td>0,629</td>
<td>?</td>
</tr>
<tr>
<td>21 InputMappedClassifier</td>
<td>176</td>
<td>62,8571</td>
<td>?</td>
<td>0,629</td>
<td>?</td>
</tr>
<tr>
<td>22 LibSVM</td>
<td>176</td>
<td>62,8571</td>
<td>?</td>
<td>0,629</td>
<td>?</td>
</tr>
</tbody>
</table>
C. GKO, BOA ve HBO-BPBÇ Algoritmalarının 100 Popülasyon 200 İterasyon Parametreleri ile Kürtajın Yasallaşması Veri Seti’ne Uygulanmasından Elde Edilen Sonuç

GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 100 popülasyon ve 200 iterasyon parametreleri ile Kürtajın Yasallaşması veri setine uygulanması sonucunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.27’de GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar, WEKA‘da yer alan algoritmalar kullanılarak elde edilen sonuçlar ile karşılaştırıldığında HBO-BPBÇ % 70 doğru sınıflandırma oranı ile ikinci, BOA %64,6429 ile onbirinci ve GKO ise %64,2857 ile on üçüncü sıradada yer almıştır. Ayrıca BOA’nın standart sapması 7,06, HBO-BPBÇ’nin 8,78 ve GKO’nun ise 44,24 olmuştur. Şekil 7.15’de algoritmaların çalışmaları sırasında 200 iterasyon boyunca uygunluk değerinin değişimi gösterilmiştir.
Şekil 7.15. Kürtajın Yasallaşması Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 200 iterasyon parametrelerinde uyguluk değeri değişimi

Tablo 7.28. GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 200 iterasyon parametreleri ile kürtajın yasallaşması veri setine uygulanması sonucunda elde edilen başarı sırası.

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandıran Örnek Sayısı</th>
<th>Doğruluk Yüzdeleri (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 SimpleLogistic</td>
<td>201</td>
<td>71,7857</td>
<td>0,736</td>
<td>0,718</td>
<td>0,718</td>
</tr>
<tr>
<td>2 HBO-BPBÇ</td>
<td>196</td>
<td>70</td>
<td>0,650</td>
<td>0,623</td>
<td>0,623</td>
</tr>
<tr>
<td>3 SMO</td>
<td>191</td>
<td>68,2143</td>
<td>0,696</td>
<td>0,682</td>
<td>0,686</td>
</tr>
<tr>
<td>4 RandomForest</td>
<td>190</td>
<td>67,8571</td>
<td>0,689</td>
<td>0,679</td>
<td>0,667</td>
</tr>
<tr>
<td>5 NaiveBayes</td>
<td>188</td>
<td>67,1429</td>
<td>0,689</td>
<td>0,671</td>
<td>0,677</td>
</tr>
<tr>
<td>6 Bagging</td>
<td>186</td>
<td>66,4286</td>
<td>0,697</td>
<td>0,664</td>
<td>0,663</td>
</tr>
<tr>
<td>7 BOA</td>
<td>185</td>
<td>66,0714</td>
<td>0,608</td>
<td>0,643</td>
<td>0,6169</td>
</tr>
<tr>
<td>8 WISARD</td>
<td>185</td>
<td>66,0714</td>
<td>0,630</td>
<td>0,661</td>
<td>0,619</td>
</tr>
<tr>
<td>9 DecisionTable</td>
<td>184</td>
<td>65,7143</td>
<td>0,624</td>
<td>0,657</td>
<td>0,577</td>
</tr>
<tr>
<td>10 ExtraTree</td>
<td>183</td>
<td>65,3571</td>
<td>0,647</td>
<td>0,654</td>
<td>0,641</td>
</tr>
<tr>
<td>11 IterativeClassifierOptimizer</td>
<td>182</td>
<td>65</td>
<td>0,718</td>
<td>0,650</td>
<td>0,544</td>
</tr>
<tr>
<td>12 HyperPipes</td>
<td>181</td>
<td>64,6429</td>
<td>0,624</td>
<td>0,646</td>
<td>0,545</td>
</tr>
<tr>
<td>13 GKO</td>
<td>180</td>
<td>64,2857</td>
<td>0,589</td>
<td>0,642</td>
<td>0,603</td>
</tr>
<tr>
<td>14 MultilayerPerceptronCS</td>
<td>180</td>
<td>64,2857</td>
<td>0,655</td>
<td>0,643</td>
<td>0,648</td>
</tr>
<tr>
<td>15 LDA</td>
<td>178</td>
<td>63,5714</td>
<td>0,657</td>
<td>0,636</td>
<td>0,643</td>
</tr>
<tr>
<td>16 CSForest</td>
<td>178</td>
<td>63,5714</td>
<td>0,648</td>
<td>0,636</td>
<td>0,618</td>
</tr>
<tr>
<td>17 QDA</td>
<td>178</td>
<td>63,5714</td>
<td>0,577</td>
<td>0,636</td>
<td>0,579</td>
</tr>
<tr>
<td>18 IB1</td>
<td>178</td>
<td>63,5714</td>
<td>0,591</td>
<td>0,636</td>
<td>0,575</td>
</tr>
<tr>
<td>19 MODLEM</td>
<td>177</td>
<td>63,2143</td>
<td>0,593</td>
<td>0,632</td>
<td>0,594</td>
</tr>
<tr>
<td>20 ClojureClassifier</td>
<td>176</td>
<td>62,8571</td>
<td>0,629</td>
<td>0,629</td>
<td>0,629</td>
</tr>
<tr>
<td>21 InputMappedClassifier</td>
<td>176</td>
<td>62,8571</td>
<td>0,629</td>
<td>0,629</td>
<td>0,629</td>
</tr>
<tr>
<td>22 LibSVM</td>
<td>176</td>
<td>62,8571</td>
<td>0,629</td>
<td>0,629</td>
<td>0,629</td>
</tr>
</tbody>
</table>

145
<table>
<thead>
<tr>
<th>23</th>
<th>MultiScheme</th>
<th>176</th>
<th>62.8571</th>
<th>?</th>
<th>0.629</th>
<th>?</th>
</tr>
</thead>
<tbody>
<tr>
<td>24</td>
<td>Stacking</td>
<td>176</td>
<td>62.8571</td>
<td>?</td>
<td>0.629</td>
<td>?</td>
</tr>
<tr>
<td>25</td>
<td>Vote</td>
<td>176</td>
<td>62.8571</td>
<td>?</td>
<td>0.629</td>
<td>?</td>
</tr>
<tr>
<td>26</td>
<td>MultiClassClassifier</td>
<td>170</td>
<td>60.7143</td>
<td>0.610</td>
<td>0.607</td>
<td>0.607</td>
</tr>
<tr>
<td>27</td>
<td>IBk</td>
<td>169</td>
<td>60.3571</td>
<td>0.688</td>
<td>0.604</td>
<td>0.607</td>
</tr>
<tr>
<td>28</td>
<td>RandomTree</td>
<td>169</td>
<td>60.3571</td>
<td>0.607</td>
<td>0.604</td>
<td>0.598</td>
</tr>
<tr>
<td>29</td>
<td>REPTree</td>
<td>160</td>
<td>57.1429</td>
<td>0.647</td>
<td>0.571</td>
<td>0.575</td>
</tr>
<tr>
<td>30</td>
<td>OrdinalClassClassifier</td>
<td>159</td>
<td>56.7857</td>
<td>0.641</td>
<td>0.568</td>
<td>0.578</td>
</tr>
<tr>
<td>31</td>
<td>RseslibKnn</td>
<td>156</td>
<td>55.7143</td>
<td>0.596</td>
<td>0.557</td>
<td>0.521</td>
</tr>
<tr>
<td>32</td>
<td>RBFNetwork</td>
<td>150</td>
<td>53.5714</td>
<td>0.636</td>
<td>0.536</td>
<td>0.560</td>
</tr>
<tr>
<td>33</td>
<td>OLM</td>
<td>149</td>
<td>53.2143</td>
<td>0.435</td>
<td>0.532</td>
<td>0.479</td>
</tr>
<tr>
<td>34</td>
<td>Logistic</td>
<td>141</td>
<td>50.3571</td>
<td>0.559</td>
<td>0.504</td>
<td>0.523</td>
</tr>
<tr>
<td>35</td>
<td>VFI</td>
<td>53</td>
<td>18.9286</td>
<td>?</td>
<td>0.189</td>
<td>?</td>
</tr>
</tbody>
</table>

D. GKO, BOA ve HBO-BPBÇ Algoritmalarının 200 Popülasyon 100 İterasyon Parametreleri ile Kürtajın Yasallaşması Veri Seti’ne Uygulanmasından Elde Edilen Sonuç

GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 200 popülasyon ve 100 iterasyon parametreleri ile Kürtajınyasallaşması veri setine uygulanması sonucunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.28’de GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar, WEKA’da yer alan algoritmalar kullanılarak elde edilen sonuçlar ile karşılaştırıldığında GKO %69,6429 doğru sınıflandırma oranı ile ikinci, HBO-BPBÇ %67,1429 ile beşinci ve BOA ise %67,1429 ile altıncı sıradaki yer almıştır. Şekil 7.16’da algoritmaların çalışmalarını sırasında 100 iterasyon boyunca uygunluk değerinin değişimi gösterilmiştir.
Şekil 7.16. Kürtajın Yasallaşması Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 100 iterasyon parametrelerinde uyguluk değeri değişimi

Tablo 7.29. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 100 iterasyon parametreleri ile kürtaj veri setine uygulanması sonucunda elde edilen başarı sırası.

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Dogru Sınıflandırılan Örnek Sayısı</th>
<th>Doğruluk Yüzd. (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1  SimpleLogistic</td>
<td>201</td>
<td>71.7857</td>
<td>0.736</td>
<td>0.718</td>
<td>0.718</td>
</tr>
<tr>
<td>2  GKO</td>
<td>195</td>
<td>69.6429</td>
<td>0.671</td>
<td>0.542</td>
<td>0.539</td>
</tr>
<tr>
<td>3  SMO</td>
<td>191</td>
<td>68.2143</td>
<td>0.696</td>
<td>0.682</td>
<td>0.686</td>
</tr>
<tr>
<td>4  RandomForest</td>
<td>190</td>
<td>67.8571</td>
<td>0.689</td>
<td>0.679</td>
<td>0.667</td>
</tr>
<tr>
<td>5  HBO-BPBÇ</td>
<td>188</td>
<td>67.1429</td>
<td>0.616</td>
<td>0.584</td>
<td>0.578</td>
</tr>
<tr>
<td>6  BOA</td>
<td>188</td>
<td>67.1429</td>
<td>0.611</td>
<td>0.609</td>
<td>0.608</td>
</tr>
<tr>
<td>7  NaiveBayes</td>
<td>188</td>
<td>67.1429</td>
<td>0.689</td>
<td>0.671</td>
<td>0.677</td>
</tr>
<tr>
<td>8  Bagging</td>
<td>186</td>
<td>66.4286</td>
<td>0.697</td>
<td>0.664</td>
<td>0.663</td>
</tr>
<tr>
<td>9  WISARD</td>
<td>185</td>
<td>66.0714</td>
<td>0.630</td>
<td>0.661</td>
<td>0.619</td>
</tr>
<tr>
<td>10 DecisionTable</td>
<td>184</td>
<td>65.7143</td>
<td>0.624</td>
<td>0.657</td>
<td>0.577</td>
</tr>
<tr>
<td>11 ExtraTree</td>
<td>183</td>
<td>65.3571</td>
<td>0.647</td>
<td>0.654</td>
<td>0.641</td>
</tr>
<tr>
<td>12 IterativeClassifierOptimizer</td>
<td>182</td>
<td>65</td>
<td>0.718</td>
<td>0.650</td>
<td>0.544</td>
</tr>
<tr>
<td>13 HyperPipes</td>
<td>181</td>
<td>64.6429</td>
<td>0.624</td>
<td>0.646</td>
<td>0.545</td>
</tr>
<tr>
<td>14 MultilayerPerceptronCS</td>
<td>180</td>
<td>64.2857</td>
<td>0.655</td>
<td>0.643</td>
<td>0.648</td>
</tr>
<tr>
<td>15 LDA</td>
<td>178</td>
<td>63.5714</td>
<td>0.657</td>
<td>0.636</td>
<td>0.643</td>
</tr>
<tr>
<td>16 CSForest</td>
<td>178</td>
<td>63.5714</td>
<td>0.648</td>
<td>0.636</td>
<td>0.618</td>
</tr>
<tr>
<td>17 QDA</td>
<td>178</td>
<td>63.5714</td>
<td>0.577</td>
<td>0.636</td>
<td>0.579</td>
</tr>
<tr>
<td>18 IB1</td>
<td>178</td>
<td>63.5714</td>
<td>0.591</td>
<td>0.636</td>
<td>0.575</td>
</tr>
<tr>
<td>19 MODLEM</td>
<td>177</td>
<td>63.2143</td>
<td>0.593</td>
<td>0.632</td>
<td>0.594</td>
</tr>
<tr>
<td>20 ClojureClassifier</td>
<td>176</td>
<td>62.8571</td>
<td>?</td>
<td>0.629</td>
<td>?</td>
</tr>
<tr>
<td>21 InputMappedClassifier</td>
<td>176</td>
<td>62.8571</td>
<td>?</td>
<td>0.629</td>
<td>?</td>
</tr>
<tr>
<td>Sıra</td>
<td>Algoritma</td>
<td>Popülasyon</td>
<td>Đ score</td>
<td>1.</td>
<td>2.</td>
</tr>
<tr>
<td>------</td>
<td>---------------------------</td>
<td>------------</td>
<td>---------</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>22</td>
<td>LibSVM</td>
<td>176</td>
<td>62,8571</td>
<td>?</td>
<td>0,629</td>
</tr>
<tr>
<td>23</td>
<td>MultiScheme</td>
<td>176</td>
<td>62,8571</td>
<td>?</td>
<td>0,629</td>
</tr>
<tr>
<td>24</td>
<td>Stacking</td>
<td>176</td>
<td>62,8571</td>
<td>?</td>
<td>0,629</td>
</tr>
<tr>
<td>25</td>
<td>Vote</td>
<td>176</td>
<td>62,8571</td>
<td>?</td>
<td>0,629</td>
</tr>
<tr>
<td>26</td>
<td>MultiClassClassifier</td>
<td>170</td>
<td>60,7143</td>
<td>0,610</td>
<td>0,607</td>
</tr>
<tr>
<td>27</td>
<td>IBk</td>
<td>169</td>
<td>60,3571</td>
<td>0,688</td>
<td>0,604</td>
</tr>
<tr>
<td>28</td>
<td>RandomForest</td>
<td>169</td>
<td>60,3571</td>
<td>0,607</td>
<td>0,604</td>
</tr>
<tr>
<td>29</td>
<td>REPTree</td>
<td>160</td>
<td>57,1429</td>
<td>0,647</td>
<td>0,571</td>
</tr>
<tr>
<td>30</td>
<td>OrdinalClassClassifier</td>
<td>159</td>
<td>56,7857</td>
<td>0,641</td>
<td>0,568</td>
</tr>
<tr>
<td>31</td>
<td>RseslibKnn</td>
<td>156</td>
<td>55,7143</td>
<td>0,596</td>
<td>0,557</td>
</tr>
<tr>
<td>32</td>
<td>RBFNetwork</td>
<td>150</td>
<td>53,5714</td>
<td>0,636</td>
<td>0,536</td>
</tr>
<tr>
<td>33</td>
<td>OLM</td>
<td>149</td>
<td>53,2143</td>
<td>0,435</td>
<td>0,532</td>
</tr>
<tr>
<td>34</td>
<td>Logistic</td>
<td>141</td>
<td>50,3571</td>
<td>0,559</td>
<td>0,504</td>
</tr>
<tr>
<td>35</td>
<td>VFI</td>
<td>53</td>
<td>18,9286</td>
<td>?</td>
<td>0,189</td>
</tr>
</tbody>
</table>

**E. GKO, BOA ve HBO-BPBÇ Algoritmalarının 200 Popülasyon 150 İterasyon Parametreleri ile Kürtajın Yasallaşması Veri Seti’ne Uygulanmasından Elde Edilen Sonuç**

GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 200 popülasyon ve 150 iterasyon parametreleri ile Kürtaj veri setine uygulanması sonucunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.29’da GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar, WEKA’da yer alan algoritmalar kullanılarak elde edilen sonuçlar ile karşılaştırıldığında BOA %67,8571 doğru sınıflandırma oranı ile üçüncü, GKO %67,8571 ile dördüncü ve HBO-BPBÇ ise %66,0714 ile sekizinci sırada yer almıştır. Şekil 7.17’de algoritmaların çalışmaları sırasında 150 iterasyon boyunca uyguluk değerinin değişimi gösterilmiştir.
Şekil 7.17. Kürtajın Yasallaşması Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 150 iterasyon parametrelerinde uygunluk değeri değişimi

Tablo 7.30. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 150 iterasyon parametreleri ile kürtaj veri setine uygulandıktan sonra elde edilen başarı sırası.

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğrulan &quot;{\textsuperscript}{\textdagger}\hspace{1em}&quot; Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>SimpleLogistic</td>
<td>201</td>
<td>71,7857</td>
<td>0,736</td>
<td>0,718</td>
<td>0,718</td>
</tr>
<tr>
<td>SMO</td>
<td>191</td>
<td>68,2143</td>
<td>0,696</td>
<td>0,682</td>
<td>0,686</td>
</tr>
<tr>
<td>BOA</td>
<td>190</td>
<td><strong>67,8571</strong></td>
<td><strong>0,621</strong></td>
<td><strong>0,674</strong></td>
<td><strong>0,633</strong></td>
</tr>
<tr>
<td>GKO</td>
<td>190</td>
<td><strong>67,8571</strong></td>
<td><strong>0,609</strong></td>
<td><strong>0,612</strong></td>
<td><strong>0,610</strong></td>
</tr>
<tr>
<td>RandomForest</td>
<td>190</td>
<td>67,8571</td>
<td>0,689</td>
<td>0,679</td>
<td>0,667</td>
</tr>
<tr>
<td>NaiveBayes</td>
<td>188</td>
<td>67,1429</td>
<td>0,689</td>
<td>0,671</td>
<td>0,677</td>
</tr>
<tr>
<td>Bagging</td>
<td>186</td>
<td>66,4286</td>
<td>0,697</td>
<td>0,664</td>
<td>0,663</td>
</tr>
<tr>
<td>HBO-BPBÇ</td>
<td><strong>185</strong></td>
<td><strong>66,0714</strong></td>
<td><strong>0,607</strong></td>
<td><strong>0,670</strong></td>
<td><strong>0,622</strong></td>
</tr>
<tr>
<td>WISARD</td>
<td>185</td>
<td>66,0714</td>
<td>0,630</td>
<td>0,661</td>
<td>0,619</td>
</tr>
<tr>
<td>DecisionTable</td>
<td>184</td>
<td>65,7143</td>
<td>0,624</td>
<td>0,657</td>
<td>0,577</td>
</tr>
<tr>
<td>ExtraTree</td>
<td>183</td>
<td>65,3571</td>
<td>0,647</td>
<td>0,654</td>
<td>0,641</td>
</tr>
<tr>
<td>IterativeClassifierOptimizer</td>
<td>182</td>
<td>65</td>
<td>0,718</td>
<td>0,650</td>
<td>0,544</td>
</tr>
<tr>
<td>HyperPipes</td>
<td>181</td>
<td>64,6429</td>
<td>0,624</td>
<td>0,646</td>
<td>0,545</td>
</tr>
<tr>
<td>MultilayerPerceptronCS</td>
<td>180</td>
<td>64,2857</td>
<td>0,655</td>
<td>0,643</td>
<td>0,648</td>
</tr>
<tr>
<td>LDA</td>
<td>178</td>
<td>63,5714</td>
<td>0,657</td>
<td>0,636</td>
<td>0,643</td>
</tr>
<tr>
<td>CSForest</td>
<td>178</td>
<td>63,5714</td>
<td>0,648</td>
<td>0,636</td>
<td>0,618</td>
</tr>
<tr>
<td>QDA</td>
<td>178</td>
<td>63,5714</td>
<td>0,577</td>
<td>0,636</td>
<td>0,579</td>
</tr>
<tr>
<td>IB1</td>
<td>178</td>
<td>63,5714</td>
<td>0,591</td>
<td>0,636</td>
<td>0,575</td>
</tr>
<tr>
<td>MODLEM</td>
<td>177</td>
<td>63,2143</td>
<td>0,593</td>
<td>0,632</td>
<td>0,594</td>
</tr>
<tr>
<td>ClojureClassifier</td>
<td>176</td>
<td>62,8571</td>
<td>?</td>
<td>0,629</td>
<td>?</td>
</tr>
<tr>
<td>InputMappedClassifier</td>
<td>176</td>
<td>62,8571</td>
<td>?</td>
<td>0,629</td>
<td>?</td>
</tr>
<tr>
<td>No.</td>
<td>Algoritma</td>
<td>Popülasyon</td>
<td>Sınıflandırma Oranı</td>
<td>Standart Sapması</td>
<td></td>
</tr>
<tr>
<td>-----</td>
<td>------------------------</td>
<td>------------</td>
<td>--------------------</td>
<td>-----------------</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>LibSVM</td>
<td>176</td>
<td>62,8571</td>
<td>0,629</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>MultiScheme</td>
<td>176</td>
<td>62,8571</td>
<td>0,629</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>Stacking</td>
<td>176</td>
<td>62,8571</td>
<td>0,629</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>Vote</td>
<td>176</td>
<td>62,8571</td>
<td>0,629</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>MultiClassClassifier</td>
<td>170</td>
<td>60,7143</td>
<td>0,610</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>IBk</td>
<td>169</td>
<td>60,3571</td>
<td>0,688</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>RandomTree</td>
<td>169</td>
<td>60,3571</td>
<td>0,607</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>REPTree</td>
<td>160</td>
<td>57,1429</td>
<td>0,647</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>OrdinalClassifier</td>
<td>159</td>
<td>56,7857</td>
<td>0,641</td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>RseslibKnn</td>
<td>156</td>
<td>55,7143</td>
<td>0,596</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>RBFNetwork</td>
<td>150</td>
<td>53,5714</td>
<td>0,636</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>OLM</td>
<td>149</td>
<td>53,2143</td>
<td>0,435</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>Logistic</td>
<td>141</td>
<td>50,3571</td>
<td>0,559</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>VFI</td>
<td>53</td>
<td>18,9286</td>
<td>0,189</td>
<td></td>
</tr>
</tbody>
</table>

**F. GKO, BOA ve HBO-BPBÇ Algoritmalarının 200 Popülasyon 200 İterasyon Parametreleri ile Kürtajın Yasallaşması Veri Seti’ne Uygulanmasından Elde Edilen Sonuç**

Şekil 7.18. Kürtajın Yasallaşması Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBC algoritmalarının 200 popülasyon 200 iterasyon parametrelerinde uygunluk değeri değişimini

Tablo 7.31. GKO, BOA ve HBO-BPBC algoritmalarının 200 popülasyon 200 iterasyon parametreleri ile Kürtajın Yasallaşması veri setine uygulanması sonucunda elde edilen başarı sırası.

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandırılan Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 SimpleLogistic</td>
<td>201</td>
<td>71,7857</td>
<td>0,736</td>
<td>0,718</td>
<td>0,718</td>
</tr>
<tr>
<td>2 SMO</td>
<td>191</td>
<td>68,2143</td>
<td>0,696</td>
<td>0,682</td>
<td>0,686</td>
</tr>
<tr>
<td>3 RandomForest</td>
<td>190</td>
<td>67,8571</td>
<td>0,689</td>
<td>0,679</td>
<td>0,667</td>
</tr>
<tr>
<td>4 NaiveBayes</td>
<td>188</td>
<td>67,1429</td>
<td>0,689</td>
<td>0,671</td>
<td>0,677</td>
</tr>
<tr>
<td>5 HBO-BPBC</td>
<td>187</td>
<td>66,7857</td>
<td>0,607</td>
<td>0,635</td>
<td>0,619</td>
</tr>
<tr>
<td>6 Bagging</td>
<td>186</td>
<td>66,4286</td>
<td>0,697</td>
<td>0,664</td>
<td>0,663</td>
</tr>
<tr>
<td>7 BOA</td>
<td>185</td>
<td>66,0714</td>
<td>0,601</td>
<td>0,616</td>
<td>0,601</td>
</tr>
<tr>
<td>8 WISARD</td>
<td>185</td>
<td>66,0714</td>
<td>0,630</td>
<td>0,661</td>
<td>0,619</td>
</tr>
<tr>
<td>9 DecisionTable</td>
<td>184</td>
<td>65,7143</td>
<td>0,624</td>
<td>0,657</td>
<td>0,577</td>
</tr>
<tr>
<td>10 ExtraTree</td>
<td>183</td>
<td>65,3571</td>
<td>0,647</td>
<td>0,654</td>
<td>0,641</td>
</tr>
<tr>
<td>11 IterativeClassifierOptimizer</td>
<td>182</td>
<td>65</td>
<td>0,718</td>
<td>0,650</td>
<td>0,544</td>
</tr>
<tr>
<td>12 HyperPipes</td>
<td>181</td>
<td>64,6429</td>
<td>0,624</td>
<td>0,646</td>
<td>0,545</td>
</tr>
<tr>
<td>13 MultilayerPerceptronCS</td>
<td>180</td>
<td>64,2857</td>
<td>0,655</td>
<td>0,643</td>
<td>0,648</td>
</tr>
<tr>
<td>14 GKO</td>
<td>179</td>
<td>63,9286</td>
<td>0,606</td>
<td>0,635</td>
<td>0,597</td>
</tr>
<tr>
<td>15 LDA</td>
<td>178</td>
<td>63,5714</td>
<td>0,657</td>
<td>0,636</td>
<td>0,643</td>
</tr>
<tr>
<td>16 CSForest</td>
<td>178</td>
<td>63,5714</td>
<td>0,648</td>
<td>0,636</td>
<td>0,618</td>
</tr>
<tr>
<td>17 QDA</td>
<td>178</td>
<td>63,5714</td>
<td>0,577</td>
<td>0,636</td>
<td>0,579</td>
</tr>
<tr>
<td>18 IB1</td>
<td>178</td>
<td>63,5714</td>
<td>0,591</td>
<td>0,636</td>
<td>0,575</td>
</tr>
<tr>
<td>19 MODLEM</td>
<td>177</td>
<td>63,2143</td>
<td>0,593</td>
<td>0,632</td>
<td>0,594</td>
</tr>
<tr>
<td>20 ClojureClassifier</td>
<td>176</td>
<td>62,8571</td>
<td>?</td>
<td>0,629</td>
<td>?</td>
</tr>
<tr>
<td>21 InputMappedClassifier</td>
<td>176</td>
<td>62,8571</td>
<td>?</td>
<td>0,629</td>
<td>?</td>
</tr>
</tbody>
</table>
22 LibSVM 176 62,857 1,8629 1629 23 MultiScheme 176 62,857 1,8629 1629 24 Stack 176 62,857 1,8629 1629 25 Vote 176 62,857 1,8629 1629 26 MultiClassClassifier 170 60,7143 0,610 1607 1607 27 IBk 169 60,357 1,688 1604 1607 28 RandomTree 169 60,357 1,688 1604 1607 29 REPTree 160 57,1429 0,647 1571 1575 30 OrdinalClassClassifier 159 56,7857 0,641 1568 1578 31 RseslibKnn 156 55,7143 0,596 1557 0,521 32 RBFNetwork 150 53,5714 0,636 1536 1560 33 OLM 149 53,2143 0,435 1532 0,479 34 Logistic 141 50,3571 0,559 1504 1523 35 VFI 53 18,9286 0,189 0,189

G. Kürtajın Yasallaşması Veri Seti Kullanılarak Algoritmaların Uyguluk Fonksiyonlarının Eşit Sayıda Çağırılma Sayısına Göre Karşılaştırılması


Tablo 7.32. BOA ve HBO-BPBÇ algoritmalarının kürtajın yasallaşması veri setinde uyguluk fonksiyonlarının eşit sayıda çağırılması sonuçları

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandırılan Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassas-</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>SimpleLogistic</td>
<td>201</td>
<td>71,7857</td>
<td>0,736</td>
<td>0,718</td>
<td>0,718</td>
</tr>
<tr>
<td>SMO</td>
<td>191</td>
<td>68,2143</td>
<td>0,696</td>
<td>0,682</td>
<td>0,686</td>
</tr>
<tr>
<td>BOA</td>
<td>190</td>
<td>67,8571</td>
<td>0,625</td>
<td>0,627</td>
<td>0,621</td>
</tr>
<tr>
<td>RandomForest</td>
<td>190</td>
<td>67,8571</td>
<td>0,689</td>
<td>0,679</td>
<td>0,667</td>
</tr>
<tr>
<td>HBO-BPBÇ</td>
<td>188</td>
<td>67,1429</td>
<td>0,615</td>
<td>0,644</td>
<td>0,625</td>
</tr>
<tr>
<td>NaiveBayes</td>
<td>188</td>
<td>67,1429</td>
<td>0,689</td>
<td>0,671</td>
<td>0,677</td>
</tr>
<tr>
<td>Bagging</td>
<td>186</td>
<td>66,4286</td>
<td>0,697</td>
<td>0,664</td>
<td>0,663</td>
</tr>
<tr>
<td>WISARD</td>
<td>185</td>
<td>66,0714</td>
<td>0,630</td>
<td>0,661</td>
<td>0,619</td>
</tr>
<tr>
<td>DecisionTable</td>
<td>184</td>
<td>65,7143</td>
<td>0,624</td>
<td>0,657</td>
<td>0,577</td>
</tr>
<tr>
<td>ExtraTree</td>
<td>183</td>
<td>65,3571</td>
<td>0,647</td>
<td>0,654</td>
<td>0,641</td>
</tr>
<tr>
<td>IterativeClassifierOptimizer</td>
<td>182</td>
<td>65</td>
<td>0,718</td>
<td>0,650</td>
<td>0,544</td>
</tr>
<tr>
<td>No.</td>
<td>Algoritma</td>
<td>Popülasyon Sayısı</td>
<td>Iterasyon Sayısı</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----</td>
<td>----------------------</td>
<td>-------------------</td>
<td>------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>HyperPipes</td>
<td>181</td>
<td>64,6429</td>
<td>0,624</td>
<td>0,646</td>
</tr>
<tr>
<td>13</td>
<td>MultilayerPerceptronCS</td>
<td>180</td>
<td>64,2857</td>
<td>0,655</td>
<td>0,643</td>
</tr>
<tr>
<td>14</td>
<td>GKO</td>
<td>178</td>
<td>63,5714</td>
<td>0,469</td>
<td>0,453</td>
</tr>
<tr>
<td>15</td>
<td>LDA</td>
<td>178</td>
<td>63,5714</td>
<td>0,657</td>
<td>0,636</td>
</tr>
<tr>
<td>16</td>
<td>CSForest</td>
<td>178</td>
<td>63,5714</td>
<td>0,648</td>
<td>0,636</td>
</tr>
<tr>
<td>17</td>
<td>QDA</td>
<td>178</td>
<td>63,5714</td>
<td>0,577</td>
<td>0,636</td>
</tr>
<tr>
<td>18</td>
<td>IB1</td>
<td>178</td>
<td>63,5714</td>
<td>0,591</td>
<td>0,636</td>
</tr>
<tr>
<td>19</td>
<td>MODLEM</td>
<td>177</td>
<td>63,2143</td>
<td>0,593</td>
<td>0,632</td>
</tr>
<tr>
<td>20</td>
<td>ClojureClassifier</td>
<td>176</td>
<td>62,8571</td>
<td>?</td>
<td>0,629</td>
</tr>
<tr>
<td>21</td>
<td>InputMappedClassifier</td>
<td>176</td>
<td>62,8571</td>
<td>?</td>
<td>0,629</td>
</tr>
<tr>
<td>22</td>
<td>LibSVM</td>
<td>176</td>
<td>62,8571</td>
<td>?</td>
<td>0,629</td>
</tr>
<tr>
<td>23</td>
<td>MultiScheme</td>
<td>176</td>
<td>62,8571</td>
<td>?</td>
<td>0,629</td>
</tr>
<tr>
<td>24</td>
<td>Stacking</td>
<td>176</td>
<td>62,8571</td>
<td>?</td>
<td>0,629</td>
</tr>
<tr>
<td>25</td>
<td>Vote</td>
<td>176</td>
<td>62,8571</td>
<td>?</td>
<td>0,629</td>
</tr>
<tr>
<td>26</td>
<td>MultiClassClassifier</td>
<td>170</td>
<td>60,7143</td>
<td>0,610</td>
<td>0,607</td>
</tr>
<tr>
<td>27</td>
<td>IBk</td>
<td>169</td>
<td>60,3571</td>
<td>0,688</td>
<td>0,604</td>
</tr>
<tr>
<td>28</td>
<td>RandomTree</td>
<td>169</td>
<td>60,3571</td>
<td>0,607</td>
<td>0,604</td>
</tr>
<tr>
<td>29</td>
<td>REPTree</td>
<td>160</td>
<td>57,1429</td>
<td>0,647</td>
<td>0,571</td>
</tr>
<tr>
<td>30</td>
<td>OrdinalClassClassifier</td>
<td>159</td>
<td>56,7857</td>
<td>0,641</td>
<td>0,568</td>
</tr>
<tr>
<td>31</td>
<td>RseslibKnn</td>
<td>156</td>
<td>55,7143</td>
<td>0,596</td>
<td>0,557</td>
</tr>
<tr>
<td>32</td>
<td>RBFNetwork</td>
<td>150</td>
<td>53,5714</td>
<td>0,636</td>
<td>0,536</td>
</tr>
<tr>
<td>33</td>
<td>OLM</td>
<td>149</td>
<td>53,2143</td>
<td>0,435</td>
<td>0,532</td>
</tr>
<tr>
<td>34</td>
<td>Logistic</td>
<td>141</td>
<td>50,3571</td>
<td>0,559</td>
<td>0,504</td>
</tr>
<tr>
<td>35</td>
<td>VFI</td>
<td>53</td>
<td>18,9286</td>
<td>?</td>
<td>0,189</td>
</tr>
</tbody>
</table>

7.1.4. Hillary Clinton Veri Seti’nden Elde Edilen Sonuçlar

Bu alt bölümde GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile popülasyon sayıları 100 ve 200 olarak seçilerek üzere sabit tutularak, iterasyon sayıları 100, 150 ve 200 olarak sırası ile arttırılmış ve böylece popülasyon sayısının artışına paralel olarak iterasyon sayısının arttırılması sonucunda Hillary Clinton veri setinden elde edilen sonuçlar gösterilmiştir. Ayrıca en iyi sonucun elde ettiği popülasyon ve iterasyon sonuçlarını sabit tutularak diğer parametre değerlerinin başarı sonucuna etkisi test edilmiştir. Tüm bu sonuçlar Hillary Clinton veri setinin WEKA programı içerisinde bulunan 32 farklı sınıflandırma algoritmaları ile değerlendirilmesi ile elde edilen sonuçlarla karşılaştırılmıştır. Aşağıda WEKA programından elde edilen sınıflandırma sonuçları Tablo 7.11’de gösterilmiştir. Ayrıca GKO, BOA ve HBO-BPBÇ algoritmalarında kullanılan benzerlik oranı ve uygunluk fonksiyonunda kullanılan katsayı değerleri Tablo 7.10’dan gösterilmiştir.
Tablo 7.33. Hillary Clinton veri seti için kullanılan katsayılar

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Benzerlik</th>
<th>Uygunlık Fonksiyonu</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$k_1$</td>
<td>$k_2$</td>
</tr>
<tr>
<td>GKO</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>BOA</td>
<td>0.01</td>
<td>0.1</td>
</tr>
<tr>
<td>HBO-BPBÇ</td>
<td>0.01</td>
<td>0.1</td>
</tr>
</tbody>
</table>

Tablo 7.34. Weka programı kullanılarak Hillary Clinton veri setinden elde edilen sınıflandırma sonuçları

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğrulan Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 SMO</td>
<td>250</td>
<td>84,7458</td>
<td>0,833</td>
<td>0,847</td>
<td>0,836</td>
</tr>
<tr>
<td>2 IterativeClassifierOptimizer</td>
<td>248</td>
<td>84,0678</td>
<td>0,873</td>
<td>0,841</td>
<td>0,783</td>
</tr>
<tr>
<td>3 REPTree</td>
<td>247</td>
<td>83,7288</td>
<td>0,826</td>
<td>0,837</td>
<td>0,786</td>
</tr>
<tr>
<td>4 SimpleLogistic</td>
<td>246</td>
<td>83,3898</td>
<td>0,869</td>
<td>0,834</td>
<td>0,769</td>
</tr>
<tr>
<td>5 OrdinalClassClassifier</td>
<td>246</td>
<td>83,3898</td>
<td>0,828</td>
<td>0,834</td>
<td>0,774</td>
</tr>
<tr>
<td>6 Bagging</td>
<td>245</td>
<td>83,0508</td>
<td>0,799</td>
<td>0,831</td>
<td>0,777</td>
</tr>
<tr>
<td>7 MODLEM</td>
<td>245</td>
<td>83,0508</td>
<td>0,809</td>
<td>0,831</td>
<td>0,813</td>
</tr>
<tr>
<td>8 LibSVM</td>
<td>244</td>
<td>82,7119</td>
<td>?</td>
<td>0,827</td>
<td>?</td>
</tr>
<tr>
<td>9 CSForest</td>
<td>244</td>
<td>82,7119</td>
<td>?</td>
<td>0,827</td>
<td>?</td>
</tr>
<tr>
<td>10 NaiveBayes</td>
<td>243</td>
<td>82,3729</td>
<td>0,809</td>
<td>0,824</td>
<td>0,815</td>
</tr>
<tr>
<td>11 RseslibKnn</td>
<td>243</td>
<td>82,3729</td>
<td>0,790</td>
<td>0,824</td>
<td>0,791</td>
</tr>
<tr>
<td>12 RandomForest</td>
<td>243</td>
<td>82,3729</td>
<td>0,801</td>
<td>0,824</td>
<td>0,804</td>
</tr>
<tr>
<td>13 DecisionTable</td>
<td>241</td>
<td>81,6949</td>
<td>0,790</td>
<td>0,817</td>
<td>0,765</td>
</tr>
<tr>
<td>14 IBk</td>
<td>237</td>
<td>80,3390</td>
<td>0,778</td>
<td>0,803</td>
<td>0,786</td>
</tr>
<tr>
<td>15 WISARD</td>
<td>237</td>
<td>80,3390</td>
<td>0,789</td>
<td>0,803</td>
<td>0,782</td>
</tr>
<tr>
<td>16 LDA</td>
<td>232</td>
<td>78,6441</td>
<td>0,792</td>
<td>0,786</td>
<td>0,789</td>
</tr>
<tr>
<td>17 ExtraTree</td>
<td>231</td>
<td>78,3051</td>
<td>0,778</td>
<td>0,783</td>
<td>0,780</td>
</tr>
<tr>
<td>18 VFI</td>
<td>229</td>
<td>77,6271</td>
<td>0,816</td>
<td>0,776</td>
<td>0,789</td>
</tr>
<tr>
<td>19 IB1</td>
<td>222</td>
<td>75,2542</td>
<td>0,730</td>
<td>0,753</td>
<td>0,728</td>
</tr>
<tr>
<td>20 RandomTree</td>
<td>217</td>
<td>73,5593</td>
<td>0,725</td>
<td>0,736</td>
<td>0,728</td>
</tr>
<tr>
<td>21 Logistic</td>
<td>215</td>
<td>72,8814</td>
<td>0,784</td>
<td>0,729</td>
<td>0,748</td>
</tr>
<tr>
<td>22 RBFNetwork</td>
<td>215</td>
<td>72,8814</td>
<td>0,730</td>
<td>0,729</td>
<td>0,729</td>
</tr>
<tr>
<td>23 MultiClassClassifier</td>
<td>208</td>
<td>70,5085</td>
<td>0,738</td>
<td>0,705</td>
<td>0,719</td>
</tr>
<tr>
<td>24 QDA</td>
<td>200</td>
<td>67,7966</td>
<td>0,720</td>
<td>0,678</td>
<td>0,634</td>
</tr>
<tr>
<td>25 OLM</td>
<td>196</td>
<td>66,4407</td>
<td>0,807</td>
<td>0,664</td>
<td>0,688</td>
</tr>
<tr>
<td>26 HyperPipes</td>
<td>184</td>
<td>62,3729</td>
<td>0,660</td>
<td>0,624</td>
<td>0,546</td>
</tr>
<tr>
<td>27 ClojureClassifier</td>
<td>172</td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
<td>?</td>
</tr>
<tr>
<td>28 MultiScheme</td>
<td>172</td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
<td>?</td>
</tr>
<tr>
<td>29 Stacking</td>
<td>172</td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
<td>?</td>
</tr>
<tr>
<td>30 Vote</td>
<td>172</td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
<td>?</td>
</tr>
<tr>
<td>31 InputMappedClassifier</td>
<td>172</td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
<td>?</td>
</tr>
</tbody>
</table>
A. GKO, BOA ve HBO-BPBÇ Algoritmalarının 100 Popülasyon 100 İterasyon Parametreleri ile Hillary Clinton Veri Seti’ne Uygulanmasından Elde Edilen Sonuç

GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 100 popülasyon ve 100 iterasyon parametreleri ile Hillary Clinton veri setine uygulanması sonucunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.34’te GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar, WEKA’da yer alan algoritmalar kullanılarak elde edilen sonuçlar ile karşılaştırıldığında BOA %81,0169 doğru sınıflandırma oranı ile ondördüncü, GKO ise %78,9830 ile onyedinci sırada ve HBO-BPBÇ ise %78,3051 ile ondokuzuncu sırada yer almıştır. Şekil 7.19’da algoritmaların çalışmaları sırasında 100 iterasyon boyunca uygunluk değerinin değişimi gösterilmiştir.

Şekil 7.19. Hillary Clinton Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 100 iterasyon parametrelerinde uygunluk değeri değişimi
<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandıran Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 SMO</td>
<td>250</td>
<td>84,7458</td>
<td>0,833</td>
<td>0,847</td>
<td>0,836</td>
</tr>
<tr>
<td>2 IterativeClassifierOptimizer</td>
<td>248</td>
<td>84,0678</td>
<td>0,873</td>
<td>0,841</td>
<td>0,783</td>
</tr>
<tr>
<td>3 REPTree</td>
<td>247</td>
<td>83,7288</td>
<td>0,826</td>
<td>0,837</td>
<td>0,786</td>
</tr>
<tr>
<td>4 SimpleLogistic</td>
<td>246</td>
<td>83,3898</td>
<td>0,869</td>
<td>0,834</td>
<td>0,769</td>
</tr>
<tr>
<td>5 OrdinalClassClassifier</td>
<td>246</td>
<td>83,3898</td>
<td>0,828</td>
<td>0,834</td>
<td>0,774</td>
</tr>
<tr>
<td>6 Bagging</td>
<td>245</td>
<td>83,0508</td>
<td>0,799</td>
<td>0,831</td>
<td>0,777</td>
</tr>
<tr>
<td>7 MODLEM</td>
<td>245</td>
<td>83,0508</td>
<td>0,809</td>
<td>0,831</td>
<td>0,813</td>
</tr>
<tr>
<td>8 LibSVM</td>
<td>244</td>
<td>82,7119</td>
<td>?</td>
<td>0,827</td>
<td>?</td>
</tr>
<tr>
<td>9 CSForest</td>
<td>244</td>
<td>82,7119</td>
<td>?</td>
<td>0,827</td>
<td>?</td>
</tr>
<tr>
<td>10 NaiveBayes</td>
<td>243</td>
<td>82,3729</td>
<td>0,809</td>
<td>0,824</td>
<td>0,815</td>
</tr>
<tr>
<td>11 RsslibKnn</td>
<td>243</td>
<td>82,3729</td>
<td>0,790</td>
<td>0,824</td>
<td>0,791</td>
</tr>
<tr>
<td>12 RandomForest</td>
<td>243</td>
<td>82,3729</td>
<td>0,801</td>
<td>0,824</td>
<td>0,804</td>
</tr>
<tr>
<td>13 DecisionTable</td>
<td>241</td>
<td>81,6949</td>
<td>0,790</td>
<td>0,817</td>
<td>0,765</td>
</tr>
<tr>
<td>14 BOA</td>
<td>239</td>
<td>81,0169</td>
<td>0,742</td>
<td>0,661</td>
<td>0,641</td>
</tr>
<tr>
<td>15 IBk</td>
<td>237</td>
<td>80,3390</td>
<td>0,778</td>
<td>0,803</td>
<td>0,786</td>
</tr>
<tr>
<td>16 WISARD</td>
<td>237</td>
<td>80,3390</td>
<td>0,789</td>
<td>0,803</td>
<td>0,782</td>
</tr>
<tr>
<td>17 GKO</td>
<td>233</td>
<td>78,9830</td>
<td>0,694</td>
<td>0,686</td>
<td>0,684</td>
</tr>
<tr>
<td>18 LDA</td>
<td>232</td>
<td>78,6441</td>
<td>0,792</td>
<td>0,786</td>
<td>0,789</td>
</tr>
<tr>
<td>19 HBO-BPBÇ</td>
<td>231</td>
<td>78,3051</td>
<td>0,737</td>
<td>0,642</td>
<td>0,633</td>
</tr>
<tr>
<td>20 ExtraTree</td>
<td>231</td>
<td>78,3051</td>
<td>0,778</td>
<td>0,783</td>
<td>0,780</td>
</tr>
<tr>
<td>21 VFI</td>
<td>229</td>
<td>77,6271</td>
<td>0,816</td>
<td>0,776</td>
<td>0,789</td>
</tr>
<tr>
<td>22 IB1</td>
<td>222</td>
<td>75,2542</td>
<td>0,730</td>
<td>0,753</td>
<td>0,728</td>
</tr>
<tr>
<td>23 RandomTree</td>
<td>217</td>
<td>73,5593</td>
<td>0,725</td>
<td>0,736</td>
<td>0,728</td>
</tr>
<tr>
<td>24 Logistic</td>
<td>215</td>
<td>72,8814</td>
<td>0,784</td>
<td>0,729</td>
<td>0,748</td>
</tr>
<tr>
<td>25 RBFNetwork</td>
<td>215</td>
<td>72,8814</td>
<td>0,730</td>
<td>0,729</td>
<td>0,729</td>
</tr>
<tr>
<td>26 MultiClassClassifier</td>
<td>208</td>
<td>70,5085</td>
<td>0,738</td>
<td>0,705</td>
<td>0,719</td>
</tr>
<tr>
<td>27 QDA</td>
<td>200</td>
<td>67,7966</td>
<td>0,720</td>
<td>0,678</td>
<td>0,634</td>
</tr>
<tr>
<td>28 OLM</td>
<td>196</td>
<td>66,4407</td>
<td>0,807</td>
<td>0,664</td>
<td>0,688</td>
</tr>
<tr>
<td>29 HyperPipes</td>
<td>184</td>
<td>62,3729</td>
<td>0,660</td>
<td>0,624</td>
<td>0,546</td>
</tr>
<tr>
<td>30 ClojureClassifier</td>
<td>172</td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
<td>?</td>
</tr>
<tr>
<td>31 MultiScheme</td>
<td>172</td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
<td>?</td>
</tr>
<tr>
<td>32 Stacking</td>
<td>172</td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
<td>?</td>
</tr>
<tr>
<td>33 Vote</td>
<td>172</td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
<td>?</td>
</tr>
<tr>
<td>34 InputMappedClassifier</td>
<td>172</td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
<td>?</td>
</tr>
<tr>
<td>35 MultilayerPerceptronCS</td>
<td>168</td>
<td>56,9492</td>
<td>0,557</td>
<td>0,569</td>
<td>0,450</td>
</tr>
</tbody>
</table>
B. GKO, BOA ve HBO-BPBÇ Algoritmalarının 100 Popülasyon 150 İterasyon Parametreleri ile Hillary Clinton Veri Seti’ne Uygulanmasından Elde Edilen Sonuç

GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 100 popülasyon ve 150 iterasyon parametreleri ile Hillary Clinton veri setine uygulandığında sınıflandırma sonuçları elde edilmiştir. Tablo 7.35’de GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar, WEKA’da yer alan algoritmalar kullanılarak elde edilen sonuçlar ile karşılaştırıldığında BOA %81,3559 doğru sınıflandırma oranı ile ondördüncü, HBO-BPBÇ %78,9830 ile onyedinci ve GKO ise %78,6441 ile onsekizinci sıraya yer almıştır. Şekil 7.20’de algoritmaların çalışmaları sırasında 150 iterasyon boyunca uygunluk değeri değişimi gösterilmiştir.

![Diagram](image)

**Şekil 7.20.** Hillary Clinton Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 150 iterasyon parametrelerinde uygunluk değeri değişimi

**Tablo 7.36.** GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 150 iterasyon parametreleri ile Hillary Clinton veri setine uygulanması sonucunda elde edilen başarı sırası

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandırılan Örnek Sayısı</th>
<th>Doğruluk Yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1  SMO</td>
<td>250</td>
<td>84,7458</td>
<td>0,833</td>
<td>0,847</td>
<td>0,836</td>
</tr>
<tr>
<td>2  IterativeClassifierOptimizer</td>
<td>248</td>
<td>84,0678</td>
<td>0,873</td>
<td>0,841</td>
<td>0,783</td>
</tr>
<tr>
<td>3  REPTree</td>
<td>247</td>
<td>83,7288</td>
<td>0,826</td>
<td>0,837</td>
<td>0,786</td>
</tr>
<tr>
<td>4  SimpleLogistic</td>
<td>246</td>
<td>83,3898</td>
<td>0,869</td>
<td>0,834</td>
<td>0,769</td>
</tr>
<tr>
<td>5  OrdinalClassClassifier</td>
<td>246</td>
<td>83,3898</td>
<td>0,828</td>
<td>0,834</td>
<td>0,774</td>
</tr>
<tr>
<td>6  Bagging</td>
<td>245</td>
<td>83,0508</td>
<td>0,799</td>
<td>0,831</td>
<td>0,777</td>
</tr>
<tr>
<td>------</td>
<td>----------------------</td>
<td>------------</td>
<td>-----------</td>
<td>----------</td>
<td>----------</td>
</tr>
<tr>
<td>7</td>
<td>MODLEM</td>
<td>245</td>
<td></td>
<td>83,0508</td>
<td>0,809</td>
</tr>
<tr>
<td>8</td>
<td>LibSVM</td>
<td>244</td>
<td></td>
<td>82,7119</td>
<td>?</td>
</tr>
<tr>
<td>9</td>
<td>CSForest</td>
<td>244</td>
<td></td>
<td>82,7119</td>
<td>?</td>
</tr>
<tr>
<td>10</td>
<td>NaiveBayes</td>
<td>243</td>
<td></td>
<td>82,3729</td>
<td>0,809</td>
</tr>
<tr>
<td>11</td>
<td>ReslibKnn</td>
<td>243</td>
<td></td>
<td>82,3729</td>
<td>0,790</td>
</tr>
<tr>
<td>12</td>
<td>RandomForest</td>
<td>243</td>
<td></td>
<td>82,3729</td>
<td>0,801</td>
</tr>
<tr>
<td>13</td>
<td>DecisionTable</td>
<td>241</td>
<td></td>
<td>81,6949</td>
<td>0,790</td>
</tr>
<tr>
<td>14</td>
<td>BOA</td>
<td>240</td>
<td></td>
<td>81,3559</td>
<td>0,695</td>
</tr>
<tr>
<td>15</td>
<td>IBk</td>
<td>237</td>
<td></td>
<td>80,3390</td>
<td>0,778</td>
</tr>
<tr>
<td>16</td>
<td>WISARD</td>
<td>237</td>
<td></td>
<td>80,3390</td>
<td>0,789</td>
</tr>
<tr>
<td>17</td>
<td>HBO-BPBÇ</td>
<td>233</td>
<td></td>
<td>78,9831</td>
<td>0,709</td>
</tr>
<tr>
<td>18</td>
<td>GKO</td>
<td>232</td>
<td></td>
<td>78,6441</td>
<td>0,669</td>
</tr>
<tr>
<td>19</td>
<td>LDA</td>
<td>232</td>
<td></td>
<td>78,6441</td>
<td>0,792</td>
</tr>
<tr>
<td>20</td>
<td>ExtraTree</td>
<td>231</td>
<td></td>
<td>78,3051</td>
<td>0,778</td>
</tr>
<tr>
<td>21</td>
<td>VFI</td>
<td>229</td>
<td></td>
<td>77,6271</td>
<td>0,816</td>
</tr>
<tr>
<td>22</td>
<td>IBI</td>
<td>222</td>
<td></td>
<td>75,2542</td>
<td>0,730</td>
</tr>
<tr>
<td>23</td>
<td>RandomForest</td>
<td>217</td>
<td></td>
<td>73,5593</td>
<td>0,725</td>
</tr>
<tr>
<td>24</td>
<td>Logistic</td>
<td>215</td>
<td></td>
<td>72,8814</td>
<td>0,784</td>
</tr>
<tr>
<td>25</td>
<td>RBFNetwork</td>
<td>215</td>
<td></td>
<td>72,8814</td>
<td>0,730</td>
</tr>
<tr>
<td>26</td>
<td>MultiClassClassifier</td>
<td>208</td>
<td></td>
<td>70,5085</td>
<td>0,738</td>
</tr>
<tr>
<td>27</td>
<td>QDA</td>
<td>200</td>
<td></td>
<td>67,7966</td>
<td>0,720</td>
</tr>
<tr>
<td>28</td>
<td>OLM</td>
<td>196</td>
<td></td>
<td>66,4407</td>
<td>0,807</td>
</tr>
<tr>
<td>29</td>
<td>HyperPipes</td>
<td>184</td>
<td></td>
<td>62,3729</td>
<td>0,660</td>
</tr>
<tr>
<td>30</td>
<td>ClojureClassifier</td>
<td>172</td>
<td></td>
<td>58,3051</td>
<td>?</td>
</tr>
<tr>
<td>31</td>
<td>MultiScheme</td>
<td>172</td>
<td></td>
<td>58,3051</td>
<td>?</td>
</tr>
<tr>
<td>32</td>
<td>Stacking</td>
<td>172</td>
<td></td>
<td>58,3051</td>
<td>?</td>
</tr>
<tr>
<td>33</td>
<td>Vote</td>
<td>172</td>
<td></td>
<td>58,3051</td>
<td>?</td>
</tr>
<tr>
<td>34</td>
<td>InputMappedClassifier</td>
<td>172</td>
<td></td>
<td>58,3051</td>
<td>?</td>
</tr>
<tr>
<td>35</td>
<td>MultilayerPerceptronCS</td>
<td>168</td>
<td></td>
<td>56,9492</td>
<td>0,557</td>
</tr>
</tbody>
</table>

C. GKO, BOA ve HBO-BPBÇ Algoritmalarının 100 Popülasyon 200 İterasyon Parametreleri ile Hillary Clinton Veri Seti’ne Uygulanmasından Elde Edilen Sonuç

GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 100 popülasyon ve 200 iterasyon parametreleri ile Hillary Clinton veri setine uygulanması sonucunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.36’da GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar, WEKA’da yer alan algoritmalar kullanılarak elde edilen sonuçlar ile karşılaştırıldığında BOA %81,3559 doğru sınıflandırma oranı ile ondördüncü, HBO-BPBÇ %78,9830 ile onbeşinci ve GKO ise %78,6441 ile yirmibeşinci sıradada yer almıştır. Ayrıca BOA’nın standart sapması 26,01, HBO-BPBÇ’nin 20,27 ve GKO’nun ise 41,67 olmuştur.
Şekil 7.21’de algoritmaların çalışmaları sırasında 200 iterasyon boyunca uygunluk değerinin değişimi gösterilmiştir.

Şekil 7.21. Hillary Clinton Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 100 populasyon 200 iterasyon parametrelerinde uygunluk değeri değişimi

Tablo 7.37. GKO, BOA ve HBO-BPBÇ algoritmalarının 100 populasyon 200 iterasyon parametreleri ile Hillary Clinton veri setine uygulanması sonucunda elde edilen başarı sırası.

<table>
<thead>
<tr>
<th>Algoritma</th>
<th>Doğru Sınıflandırılan Örnek Sayısı</th>
<th>Doğruluk Yüzdeleri (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 SMO</td>
<td>250</td>
<td>84,7458</td>
<td>0,833</td>
<td>0,847</td>
<td>0,836</td>
</tr>
<tr>
<td>2 IterativeClassifierOptimizer</td>
<td>248</td>
<td>84,0678</td>
<td>0,873</td>
<td>0,841</td>
<td>0,783</td>
</tr>
<tr>
<td>3 REPTree</td>
<td>247</td>
<td>83,7288</td>
<td>0,826</td>
<td>0,837</td>
<td>0,786</td>
</tr>
<tr>
<td>4 SimpleLogistic</td>
<td>246</td>
<td>83,3898</td>
<td>0,869</td>
<td>0,834</td>
<td>0,769</td>
</tr>
<tr>
<td>5 OrdinalClassClassifier</td>
<td>246</td>
<td>83,3898</td>
<td>0,828</td>
<td>0,834</td>
<td>0,774</td>
</tr>
<tr>
<td>6 Bagging</td>
<td>245</td>
<td>83,0508</td>
<td>0,799</td>
<td>0,831</td>
<td>0,777</td>
</tr>
<tr>
<td>7 MODLEM</td>
<td>245</td>
<td>83,0508</td>
<td>0,809</td>
<td>0,831</td>
<td>0,813</td>
</tr>
<tr>
<td>8 LibSVM</td>
<td>244</td>
<td>82,7119</td>
<td>?</td>
<td>0,827</td>
<td>?</td>
</tr>
<tr>
<td>9 CSForest</td>
<td>244</td>
<td>82,7119</td>
<td>?</td>
<td>0,827</td>
<td>?</td>
</tr>
<tr>
<td>10 NaiveBayes</td>
<td>243</td>
<td>82,3729</td>
<td>0,809</td>
<td>0,824</td>
<td>0,815</td>
</tr>
<tr>
<td>11 RseslibKnn</td>
<td>243</td>
<td>82,3729</td>
<td>0,790</td>
<td>0,824</td>
<td>0,791</td>
</tr>
<tr>
<td>12 RandomForest</td>
<td>243</td>
<td>82,3729</td>
<td>0,801</td>
<td>0,824</td>
<td>0,804</td>
</tr>
<tr>
<td>13 DecisionTable</td>
<td>241</td>
<td>81,6949</td>
<td>0,790</td>
<td>0,817</td>
<td>0,765</td>
</tr>
<tr>
<td>14 BOA</td>
<td>241</td>
<td>81,6949</td>
<td>0,751</td>
<td>0,675</td>
<td>0,665</td>
</tr>
<tr>
<td>15 HBO-BPBÇ</td>
<td>240</td>
<td>81,3559</td>
<td>0,763</td>
<td>0,673</td>
<td>0,662</td>
</tr>
<tr>
<td>16 IBk</td>
<td>237</td>
<td>80,3390</td>
<td>0,778</td>
<td>0,803</td>
<td>0,786</td>
</tr>
<tr>
<td>17 WISARD</td>
<td>237</td>
<td>80,3390</td>
<td>0,789</td>
<td>0,803</td>
<td>0,782</td>
</tr>
</tbody>
</table>
D. GKO, BOA ve HBO-BPBÇ Algoritmalarının 200 Popülasyon 100 İterasyon Parametreleri ile Hillary Clinton Veri Seti’ne Uygulanmasından Elde Edilen Sonuç

GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 200 popülasyon ve 100 iterasyon parametreleri ile Hillary Clinton veri setine uygulanması sonucunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.11’dede GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar, WEKA’da yer alan algoritmalar kullanarak elde edilen sonuçlar ile karşılaştırıldığında BOA %81,3559 doğru sınıflandırma oranı ile on üçüncü, HBO-BPBÇ %80 ile onyedinci ve GKO ise %79,6610 ile onsekizinci sıraday yer almıştır. Şekil 7.7’de algoritmaların çalışmaları sırasında 100 iterasyon boyunca uygunluk değerinin değişimi gösterilmiştir.
**Şekil 7.22.** Hillary Clinton Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 100 iterasyon parametrelerinde uygunluk değeri değişimi

**Tablo 7.38.** GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 100 iterasyon parametreleri ile Hillary Clinton veri setine uygulanması sonucunda elde edilen başarı sırası.
<table>
<thead>
<tr>
<th>No.</th>
<th>Algoritma</th>
<th>Popülasyon</th>
<th>İterasyon</th>
<th>% Tespit</th>
<th>% Tespit</th>
<th>% Tespit</th>
</tr>
</thead>
<tbody>
<tr>
<td>22</td>
<td>IB1</td>
<td>222</td>
<td></td>
<td>0,730</td>
<td>0,753</td>
<td>0,728</td>
</tr>
<tr>
<td>23</td>
<td>RandomTree</td>
<td>217</td>
<td></td>
<td>0,725</td>
<td>0,736</td>
<td>0,728</td>
</tr>
<tr>
<td>24</td>
<td>Logistic</td>
<td>215</td>
<td></td>
<td>0,784</td>
<td>0,729</td>
<td>0,748</td>
</tr>
<tr>
<td>25</td>
<td>RBFNetwork</td>
<td>215</td>
<td></td>
<td>0,730</td>
<td>0,729</td>
<td>0,729</td>
</tr>
<tr>
<td>26</td>
<td>MultiClassClassifier</td>
<td>208</td>
<td></td>
<td>0,738</td>
<td>0,705</td>
<td>0,719</td>
</tr>
<tr>
<td>27</td>
<td>QDA</td>
<td>200</td>
<td></td>
<td>0,720</td>
<td>0,678</td>
<td>0,634</td>
</tr>
<tr>
<td>28</td>
<td>OLM</td>
<td>196</td>
<td></td>
<td>0,807</td>
<td>0,664</td>
<td>0,688</td>
</tr>
<tr>
<td>29</td>
<td>HyperPipes</td>
<td>184</td>
<td></td>
<td>0,660</td>
<td>0,624</td>
<td>0,546</td>
</tr>
<tr>
<td>30</td>
<td>ClojureClassifier</td>
<td>172</td>
<td></td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
</tr>
<tr>
<td>31</td>
<td>MultiScheme</td>
<td>172</td>
<td></td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
</tr>
<tr>
<td>32</td>
<td>Stacking</td>
<td>172</td>
<td></td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
</tr>
<tr>
<td>33</td>
<td>Vote</td>
<td>172</td>
<td></td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
</tr>
<tr>
<td>34</td>
<td>InputMappedClassifier</td>
<td>172</td>
<td></td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
</tr>
<tr>
<td>35</td>
<td>MultilayerPerceptronCS</td>
<td>168</td>
<td></td>
<td>56,9492</td>
<td>0,557</td>
<td>0,569</td>
</tr>
</tbody>
</table>

**E. GKO, BOA ve HBO-BPBÇ Algoritmalarının 200 Popülasyon 150 İterasyon Parametreleri ile Hillary Clinton Veri Seti’ne Uygulanmasından Elde Edilen Sonuç**

GKO, BOA ve HBO-BPBÇ algoritmalarının sırası ile 200 popülasyon ve 150 iterasyon parametreleri ile Hillary Clinton veri setine uygulanması sonucunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.38’de GKO, BOA ve HBO-BPBÇ algoritmaları ile elde edilen sonuçlar, WEKA’dan yer alan algoritmalar kullanılarak elde edilen sonuçlar ile karşılaştırıldığında HBO-BPBÇ %80,3390 doğru sınıflandırma oranı ile ondördüncü, GKO %80 ile onyedinci ve BOA ise %79,6610 ile onsekizinci sırada yer almıştır. Şekil 7.23’teki algoritmaların çalışmaları sırasında 150 iterasyon boyunca uygunluk değerinin değişimi gösterilmiştir.
**Şekil 7.23.** Hillary Clinton Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBC algoritmalarının 200 popülasyon 150 iterasyon parametrelerinde uygunluk değeri değişimi

**Tablo 7.39.** GKO, BOA ve HBO-BPBC algoritmalarının 200 popülasyon 150 iterasyon parametreleri ile Hillary Clinton veri setine uygulanması sonucunda elde edilen başarı sırası.

<table>
<thead>
<tr>
<th>Algoritma</th>
<th><em>Doğru sınıflandırılan örnek sayısı</em></th>
<th>_Doğruluk yüzdesi (%)</th>
<th>Kesinlik</th>
<th>Hassasiyet</th>
<th>F-Ölçütü</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 SMO</td>
<td>250</td>
<td>84,7458</td>
<td>0,833</td>
<td>0,847</td>
<td>0,836</td>
</tr>
<tr>
<td>2 IterativeClassifierOptimizer</td>
<td>248</td>
<td>84,0678</td>
<td>0,873</td>
<td>0,841</td>
<td>0,783</td>
</tr>
<tr>
<td>3 REPTree</td>
<td>247</td>
<td>83,7288</td>
<td>0,826</td>
<td>0,837</td>
<td>0,786</td>
</tr>
<tr>
<td>4 SimpleLogistic</td>
<td>246</td>
<td>83,3898</td>
<td>0,869</td>
<td>0,834</td>
<td>0,769</td>
</tr>
<tr>
<td>5 OrdinalClassClassifier</td>
<td>246</td>
<td>83,3898</td>
<td>0,828</td>
<td>0,834</td>
<td>0,774</td>
</tr>
<tr>
<td>6 Bagging</td>
<td>245</td>
<td>83,0508</td>
<td>0,799</td>
<td>0,831</td>
<td>0,777</td>
</tr>
<tr>
<td>7 MODLEM</td>
<td>245</td>
<td>83,0508</td>
<td>0,809</td>
<td>0,831</td>
<td>0,813</td>
</tr>
<tr>
<td>8 LibSVM</td>
<td>244</td>
<td>82,7119</td>
<td>?</td>
<td>0,827</td>
<td>?</td>
</tr>
<tr>
<td>9 CSForest</td>
<td>244</td>
<td>82,7119</td>
<td>?</td>
<td>0,827</td>
<td>?</td>
</tr>
<tr>
<td>10 NaiveBayes</td>
<td>243</td>
<td>82,3729</td>
<td>0,809</td>
<td>0,824</td>
<td>0,815</td>
</tr>
<tr>
<td>11 RseslibKnn</td>
<td>243</td>
<td>82,3729</td>
<td>0,790</td>
<td>0,824</td>
<td>0,791</td>
</tr>
<tr>
<td>12 RandomForest</td>
<td>243</td>
<td>82,3729</td>
<td>0,801</td>
<td>0,824</td>
<td>0,804</td>
</tr>
<tr>
<td>13 DecisionTable</td>
<td>241</td>
<td>81,6949</td>
<td>0,790</td>
<td>0,817</td>
<td>0,765</td>
</tr>
<tr>
<td>14 HBO-BPBC</td>
<td>237</td>
<td>80,3390</td>
<td>0,735</td>
<td>0,659</td>
<td>0,655</td>
</tr>
<tr>
<td>15 IBk</td>
<td>237</td>
<td>80,3390</td>
<td>0,778</td>
<td>0,803</td>
<td>0,786</td>
</tr>
<tr>
<td>16 WISARD</td>
<td>237</td>
<td>80,3390</td>
<td>0,789</td>
<td>0,803</td>
<td>0,782</td>
</tr>
<tr>
<td>17 GKO</td>
<td>236</td>
<td>80</td>
<td>0,649</td>
<td>0,646</td>
<td>0,611</td>
</tr>
<tr>
<td>18 BOA</td>
<td>234</td>
<td>79,3220</td>
<td>0,716</td>
<td>0,689</td>
<td>0,693</td>
</tr>
<tr>
<td>19 LDA</td>
<td>232</td>
<td>78,6441</td>
<td>0,792</td>
<td>0,786</td>
<td>0,789</td>
</tr>
<tr>
<td>20 ExtraTree</td>
<td>231</td>
<td>78,3051</td>
<td>0,778</td>
<td>0,783</td>
<td>0,780</td>
</tr>
<tr>
<td>21 VFI</td>
<td>229</td>
<td>77,6271</td>
<td>0,816</td>
<td>0,776</td>
<td>0,789</td>
</tr>
<tr>
<td></td>
<td>Algortıım</td>
<td>Amaç Numarı</td>
<td>Öğrenme Pozisyonu</td>
<td>Temizlik Oranı</td>
<td>Senetler Oranı</td>
</tr>
<tr>
<td>---</td>
<td>---------------</td>
<td>-------------</td>
<td>-------------------</td>
<td>----------------</td>
<td>----------------</td>
</tr>
<tr>
<td>22</td>
<td>IB1</td>
<td>222</td>
<td>75,2542</td>
<td>0,730</td>
<td>0,753</td>
</tr>
<tr>
<td>23</td>
<td>Random Tree</td>
<td>217</td>
<td>73,5593</td>
<td>0,725</td>
<td>0,736</td>
</tr>
<tr>
<td>24</td>
<td>Logistic</td>
<td>215</td>
<td>72,8814</td>
<td>0,784</td>
<td>0,729</td>
</tr>
<tr>
<td>25</td>
<td>RBF Network</td>
<td>215</td>
<td>72,8814</td>
<td>0,730</td>
<td>0,729</td>
</tr>
<tr>
<td>26</td>
<td>MultiClassClassifier</td>
<td>208</td>
<td>70,5085</td>
<td>0,738</td>
<td>0,705</td>
</tr>
<tr>
<td>27</td>
<td>QDA</td>
<td>200</td>
<td>67,7966</td>
<td>0,720</td>
<td>0,678</td>
</tr>
<tr>
<td>28</td>
<td>OLM</td>
<td>196</td>
<td>66,4407</td>
<td>0,807</td>
<td>0,664</td>
</tr>
<tr>
<td>29</td>
<td>HyperPipes</td>
<td>184</td>
<td>62,3729</td>
<td>0,660</td>
<td>0,624</td>
</tr>
<tr>
<td>30</td>
<td>ClojureClassifier</td>
<td>172</td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
</tr>
<tr>
<td>31</td>
<td>MultiScheme</td>
<td>172</td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
</tr>
<tr>
<td>32</td>
<td>Stacking</td>
<td>172</td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
</tr>
<tr>
<td>33</td>
<td>Vote</td>
<td>172</td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
</tr>
<tr>
<td>34</td>
<td>InputMappedClassifier</td>
<td>172</td>
<td>58,3051</td>
<td>?</td>
<td>0,583</td>
</tr>
<tr>
<td>35</td>
<td>MultilayerPerceptronCS</td>
<td>168</td>
<td>56,9492</td>
<td>0,557</td>
<td>0,569</td>
</tr>
</tbody>
</table>

**F. GKO, BOA ve HBO-BPBC Algoritmalarının 200 Popülasyon 200 İterasyon Parametreleri ile Hillary Clinton Veri Seti’ne Uygulanmasından Elde Edilen Sonuç**

GKO, BOA ve HBO-BPBC algoritmalarının sırası ile 200 popülasyon ve 200 iterasyon parametrelerini ile Hillary Clinton veri setine uygulanması sonucunda sınıflandırma sonuçları elde edilmiştir. Tablo 7.39'da GKO, BOA ve HBO-BPBC algoritmaları ile elde edilen sonuçlar, WEKA'da yer alan algoritmalar kullanılarak elde edilen sonuçlar ile karşılaştırıldığında BOA %82,3729 doğru sınıflandırma oranı ile onuncu, HBO-BPBC %81,6949 ile ondördüncü ve GKO ise %80,6780 ile onaltıncı sırada yer almıştır. Ayrıca BOA'nın standart sapması 19,91, HBO-BPBC'nin 3,84 ve GKO'nun ise 52,36 olmuştur. Şekil 7.24'te algoritmaların çalışmalarını sırasında 200 iterasyon boyunca uygunluk değerinin değişimi gösterilmiştir.
Şekil 7.24. Hillary Clinton Veri Seti’ni kullanarak GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 200 iterasyon parametrelerinde uygunluk değeri değişimi

Tablo 7.40. GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 200 iterasyon parametreleri ile Hillary Clinton veri setine uygulanması sonucunda elde edilen başarı sırası.
G. Hillary Clinton Veri Seti Kullanılarak Algoritmaların Uygunluk Fonksiyonlarının Eşit Sayıda Çağrılma Sayısına Göre Karşılaştırılması

Uygunluk fonksiyonunun 40400 defa çağrılma değeri baz alınarak BOA ve HBO-BPBC algoritmalarının sonuçları WEKA’dan veya diğer algoritmalar kullanılarak elde edilen sonuçlar ile karşılaştırıldığında HBO-BPBC %82,7119 doğru sınıflandırma oranı ile sekizinci, BOA %81,3559 ile onbeşinci olmuştur. Aşağıda Tablo 7.40’da sıralama sonuçları gösterilmştir. Bu sonuçlara göre HBO-BPBC algoritmanın uygunluk fonksiyonun eşit çağrılma sayısı baz alınarak yapılan testlerde de başarılı olduğu görülmüştür.

**Tablo 7.41.** BOA ve HBO-BPBC algoritmaların Hillary Clinton veri setinde uygunluk fonksiyonlarının eşit sayıda çağrılması sonuçları

<table>
<thead>
<tr>
<th><strong>Algoritma</strong></th>
<th><strong>Giriş Örnek Sayısı</strong></th>
<th><strong>Doğruluğun Oranı</strong> (%)</th>
<th><strong>Kesinlik</strong> (%)</th>
<th><strong>Hassa -süyet</strong> (%)</th>
<th><strong>F-Ölçütü</strong> (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SMO</td>
<td>250</td>
<td>84,7458</td>
<td>0,833</td>
<td>0,847</td>
</tr>
<tr>
<td>2</td>
<td>IterativeClassifierOptimizer</td>
<td>248</td>
<td>84,0678</td>
<td>0,873</td>
<td>0,841</td>
</tr>
<tr>
<td>3</td>
<td>REPTree</td>
<td>247</td>
<td>83,7288</td>
<td>0,826</td>
<td>0,837</td>
</tr>
<tr>
<td>4</td>
<td>SimpleLogistic</td>
<td>246</td>
<td>83,3898</td>
<td>0,869</td>
<td>0,834</td>
</tr>
<tr>
<td>5</td>
<td>OrdinalClassClassifier</td>
<td>246</td>
<td>83,3898</td>
<td>0,828</td>
<td>0,834</td>
</tr>
<tr>
<td>6</td>
<td>Bagging</td>
<td>245</td>
<td>83,0508</td>
<td>0,799</td>
<td>0,831</td>
</tr>
<tr>
<td>7</td>
<td>MODLEM</td>
<td>245</td>
<td>83,0508</td>
<td>0,809</td>
<td>0,831</td>
</tr>
<tr>
<td>8</td>
<td>HBO-BPBC</td>
<td>244</td>
<td><strong>82,7119</strong></td>
<td><strong>0,837</strong></td>
<td><strong>0,669</strong></td>
</tr>
<tr>
<td>9</td>
<td>LibSVM</td>
<td>244</td>
<td>82,7119</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>10</td>
<td>CSForest</td>
<td>244</td>
<td>82,7119</td>
<td>?</td>
<td>?</td>
</tr>
</tbody>
</table>
7.2. Ortalama Uygunlukların Ortalamasının Grafikleri

Bu bölümde GKO, BOA ve HBO-BPBÇ Algoritmalarının 20 defa çalıştırılması sonucu ortamalama uygunluk değerlerinin ortalamalarından elde edilen grafikler her bir veri seti için sırası ile 100, 200 populasyon değerleri ve 200 iterasyon değerleri için sırası ile verilmiştir.

A. GKO, BOA ve HBO-BPBÇ Algoritmalarının 100 Popülasyon 200 İterasyon Parametreleri ile Feminist Hareket Veri Seti Kullanılarak Elde Edilen Ortalama Uygunlukların Ortalamasının Grafik ile Gösterimi

GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 200 iterasyon parameterleri baz alınarak 20 defa çalıştırılması sonucunda el edilen ortalama uygunluk değerlerinin ortamaları Şekil 7.25’de gösterilmiştir.
Şekil 7.25. Feminist Hareket Veri Seti 100 Popülasyon 200 İterasyon Ortalama Uygunluk Değerlerinin Ortalaması Grafiği

B. GKO, BOA ve HBO-BPBÇ Algoritmalarının 100 Popülasyon 200 İterasyon Parametreleri ile Ateizm Veri Seti Kullanılarak Elde Edilen Ortalama Uygunlukların Ortalamasının Grafik ile Gösterimi

GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 200 iteration parameterleri baz alınarak 20 defa çalıştırılması sonucunda el edilen ortalama uygunluk değerlerinin ortamaları Şekil 7.26’da gösterilmiştir.

Şekil 7.26. Ateizm Veri Seti 100 Popülasyon 200 İterasyon Ortalama Uygunluk Değerlerinin Ortalaması Grafiği
C. GKO, BOA ve HBO-BPBÇ Algoritmalarının 100 Popülasyon 200 İterasyon Parametreleri ile Kürtajın Yasallaşması Veri Seti Kullanılarak Elde Edilen Ortalama Uygunlukların Ortalamasını Grafik ile Gösterimi

GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 200 iterasyon parameterleri baz alınarak 20 defa çalıştırılması sonucunda el edilen ortalama uygunluk değerlerinin ortamaları Şekil 7.27’de gösterilmiştir.

![GKO, BOA ve HBO-BPBÇ Algoritmalarının 100 Popülasyon 200 İterasyon Parametreleri ile Kürtajın Yasallaşması Veri Seti Kullanılarak Elde Edilen Ortalama Uygunlukların Ortalamasını Grafik ile Gösterimi](image)

**Şekil 7.27.** Kürtajın Yasallaşması Veri Seti 100 Popülasyon 200 İterasyon Ortalama Uygunluk Değerlerinin Ortalaması Grafiği

**D. GKO, BOA ve HBO-BPBÇ Algoritmalarının 100 Popülasyon 200 İterasyon Parametreleri ile Hillary Clinton Veri Seti Kullanılarak Elde Edilen Ortalama Uygunlukların Ortalamasını Grafik ile Gösterimi**

GKO, BOA ve HBO-BPBÇ algoritmalarının 100 popülasyon 200 iterasyon parameterleri baz alınarak 20 defa çalıştırılması sonucunda el edilen ortalama uygunluk değerlerinin ortamaları Şekil 7.28’de gösterilmiştir.
Şekil 7.28. Hillary Clinton Veri Seti 100 Popülasyon 200 İterasyon Ortalama Uygunluk Değerlerinin Ortalaması Grafiği

E. GKO, BOA ve HBO-BPBÇ Algoritmalarının 200 Popülasyon 200 İterasyon Parametreleri ile Feminist Hareket Veri Seti Kullanılarak Elde Edilen Ortalama Uygunlukların Ortalamasının Grafik ile Gösterimi

GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 200 iterasyon parameterleri baz alınarak 20 defa çalıştırılması sonucunda el edilen ortalama uygunluk değerlerinin ortamları Şekil 7.29’da gösterilmiştir.

Şekil 7.29. Feminist Hareket Veri Seti 200 Popülasyon 200 İterasyon Ortalama Uygunluk Değerlerinin Ortalaması Grafiği
F. GKO, BOA ve HBO-BPBÇ Algoritmalarının 200 Popülasyon 200 İterasyon Parametreleri ile Ateizm Veri Seti Kullanılarak Elde Edilen Ortalama Uygunlukların Ortalamasının Grafik ile Gösterimi

GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 200 iterasyon parameterleri baz alınarak 20 defa çalıştırılması sonucunda el edilen ortalama uygunluk değerlerinin ortamları Şekil 7.30’da gösterilmiştir.

Şekil 7.30. Ateizm Veri Seti 200 Popülasyon 200 İterasyon Ortalama Uygunluk Değerlerinin Ortalaması Grafiği

G. GKO, BOA ve HBO-BPBÇ Algoritmalarının 200 Popülasyon 200 İterasyon Parametreleri ile Kürtajın Yasallaşması Seti Kullanılarak Elde Edilen Ortalama Uygunlukların Ortalamasının Grafik ile Gösterimi

GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 200 iterasyon parameterleri baz alınarak 20 defa çalıştırılması sonucunda el edilen ortalama uygunluk değerlerinin ortamları Şekil 7.31’de gösterilmiştir.
H. GKO, BOA ve HBO-BPBÇ Algoritmalarının 200 Popülasyon 200 İterasyon Parametreleri ile Hillary Clinton Veri Seti Kullanılarak Edilen Ortalama Uygunlukların Ortalamasının Grafik ile Gösterimi

GKO, BOA ve HBO-BPBÇ algoritmalarının 200 popülasyon 200 iterasyon parameterleri baz alınarak 20 defa çalıştırılması sonucunda el edilen ortalama uygunluk değerlerinin ortamaları Şekil 7.32’de gösterilmiştir.
8. SONUÇLAR ve ÖNERİLER

Sosyal ağlar ve bunula ilgili yapılan çalışmalar uzun bir geçmişe sahiptir. Sosyolojiden eğitim ve ekonomiye kadar birçok alanda sosyal ağ kavramı ve bu ağları çözümleme yöntemleri büyük önemine sahiptir. Yirminci yüzyılın sonlarında başlayan ve bu yüzyılın başında ivmelenen internet devrimi, sosyal ağ kavramına ve bunun uygulama alanlarına çeşitli yenilikler getirmiştir. Önceleri daha küçük sosyal ağlar uygulanma imkanı bulunan sosyal ağ analiz yöntemleri, şimdilerde bilişim alanında teknolojik gelişmelerin sunduğu imkanlarla daha öncekine göre çok büyük ağlara uygulanma imkanına kavuşmuştur. Bu, sosyal ağ analizi bakımından büyük imkanlar ortaya çıkması, niceliksel ve niteliksel olarak yeni sonuçlara yol açması anlamını taşımaktadır.


Bu tez kapsamında popüler ve yeni çevrimiçi sosyal ağ analizi problemleri geniş bir şekilde araştırılmıştır. Literatürde ilk defa çevrimiçi sosyal ağ analizi problemleri bir çalışmada bu ölçüde geniş bir şekilde derlenmiş ve araştırlar için bir rehber kaynak oluşturulmaya çalışılmıştır. Bu çalışmada popüler ve çok yeni olan bir ÇSA problemi olan Tutum Tespiti problemi üzerinde durulmuştur. Dünya kamuoyunun ilgisini üzerinde toplayan dört farklı konu ile ilgili sürekli uzun ve göreceli kısa dönemli tartışmalar hakkında beyan

174
KAYNAKLAR


Hof, P. R., ve Van der Gucht, E. 2007. Structure of the cerebral cortex of the humpback whale, Megaptera novaeangliae (Cetacea, Mysticeti, Balaenopteridae). The
Anatomical Record: Advances in Integrative Anatomy and Evolutionary Biology: Advances in Integrative Anatomy and Evolutionary Biology, 290(1), 1-31.


Liebrecht, C., Kunneman, F., ve van Den Bosch, A., 2013. The perfect solution for detecting sarcasm in tweets# not. In Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, Atlanta, Georgia, 14 June.


Ma, H., Ji, Y., Li, X., ve Zhou, R., 2016. A Microblog Hot Topic Detection Algorithm Based on Discrete Particle Swarm Optimization, In PRICAI 2016: Trends in Artificial Intelligence, Springer: Cham, Germany,

Malouf, R., ve Mullen, T., 2008. Taking sides: user classification for informal online political discourse. Internet Res., 18, 177-190,


Mansfield-Devine, S., 2008. Anti-social networking: exploiting the trusting environment of Web 2.0. Network Security. 4-7,


Miao, Q., Meng, Y. ve Sun, J., 2016. Identifying the most influential topic-sensitive opinion leaders in online review communities. In Proceedings of 2016 IEEE International Conference on Cloud Computing and Big Data Analysis (ICCCBDA), Chengdu, China, 5-7 July.


Papka, R. On-line new event detection, clustering, and tracking. PhD, University of Massachusetts Amherst, Amherst, USA, September 1999.


Petrovic, S. 2013. Real-time event detection in massive streams, PhD, University of Edinburgh , Edinburgh, UK.


Ritter, A., Mausam, Etzioni, O., ve Clark, S., 2012. Open domain event extraction from twitter. In Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Beijing, China, August 12 – 16.


ÖZGEÇMİŞ

Ümit CAN
Munzur Üniversitesi
Mühendislik Fakültesi
Bilgisayar Mühendisliği Bölümü

E-Posta: ucan@munzur.edu.tr

1982  Tunceli’de doğdu
1997-2000  Malatya Fen Lisesi’nı tamamladı
2001-2007  Bahçeşehir Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümünden mezun oldu
2010-  Munzur Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümünde Araştırma Görevlisi olarak çalışmaktadır.