

**T.C.  
YILDIZ TEKNİK ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**YAPAY BAĞIŞIKLIK SİSTEM TABANLI ALGORİTMA İLE AYKIRI  
DEĞER TESPİTİ**

**MEHMET GÜÇLÜ**

**YÜKSEK LİSANS TEZİ  
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI**

**DANIŞMAN  
DOÇ. DR. BANU DİRİ**

**İSTANBUL, 2012**

**T.C.**  
**YILDIZ TEKNİK ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**

**YAPAY BAĞIŞIKLIK SİSTEM TABANLI ALGORİTMA İLE**  
**AYKIRI DEĞER TESPİTİ**

Mehmet GÜÇLÜ tarafından hazırlanan tez çalışması 28.11.2012 tarihinde aşağıdaki jüri tarafından Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

**Tez Danışmanı**

Doç. Dr. Banu DİRİ

Yıldız Teknik Üniversitesi

**Jüri Üyeleri**

Doç. Dr. Banu DİRİ

Yıldız Teknik Üniversitesi

\_\_\_\_\_

Yrd. Doç. Dr. Songül ALBAYRAK

Yıldız Teknik Üniversitesi

\_\_\_\_\_

Yrd. Doç. Dr. Çağatay ÇATAL

İstanbul Kültür Üniversitesi

\_\_\_\_\_

## ÖNSÖZ

---

Benden yardımlarını, desteğini, sabrını ve bilgisini esirgemeyen değerli hocam Doç. Dr. Banu DİRİ' ye, bugünkü bilgilerimi elde etmede katkısı olan tüm hocalarıma ve her koşulda benden desteğini esirgemeyen aileme teşekkürü bir borç bilirim.

Kasım, 2012

Mehmet GÜÇLÜ

## İÇİNDEKİLER

|                                                              | Sayfa |
|--------------------------------------------------------------|-------|
| SİMGE LİSTESİ .....                                          | viii  |
| KISALTMA LİSTESİ .....                                       | ix    |
| ŞEKİL LİSTESİ.....                                           | x     |
| ÇİZELGE LİSTESİ .....                                        | xii   |
| ÖZET .....                                                   | xiv   |
| ABSTRACT .....                                               | xvi   |
| BÖLÜM 1                                                      |       |
| GİRİŞ .....                                                  | 1     |
| 1.1    Literatür Özeti .....                                 | 1     |
| 1.2    Tezin Amacı .....                                     | 1     |
| 1.3    Hipotez .....                                         | 2     |
| BÖLÜM 2                                                      |       |
| AYKIRILIK .....                                              | 3     |
| 2.1    Örnek .....                                           | 4     |
| 2.2    Aykırı Değer Tespiti ile İlgili Bazı Çalışmalar ..... | 6     |
| BÖLÜM 3                                                      |       |
| YAPAY BAĞIŞIKLIK SİSTEMİ VE SİSTEM TABANLI ALGORİTMASI.....  | 8     |
| 3.1    Bağışıklık Sistemi .....                              | 8     |
| 3.1.1    Doğal Bağışıklık .....                              | 8     |
| 3.1.2    Edinsel (Kazanılmış) Bağışıklık .....               | 8     |
| 3.2    Bağışıklık Algoritmaları .....                        | 9     |
| 3.2.1    Klonal Seçme Prensibi ve Algoritması .....          | 9     |
| 3.2.1.1    Klonal Seçme Prensibi .....                       | 10    |

|                                                        |                                                                      |    |
|--------------------------------------------------------|----------------------------------------------------------------------|----|
| 3.2.1.2                                                | Klonal Seçme Algoritması .....                                       | 10 |
| 3.2.2                                                  | Negatif Seçim Algoritması .....                                      | 12 |
| 3.2.3                                                  | Yapay Bağışıklık Tanıma Sistemi (AIRS) Algoritması .....             | 13 |
| 3.3                                                    | Bağışıklık Sistemleri ile Modellenmiş Bazı Çalışmalar .....          | 14 |
| 3.4                                                    | Yapay Bağışıklık Sistemi ve Sistem Tabanlı Algoritması.....          | 16 |
| 3.4.1                                                  | Yapay Bağışıklık Sistem Algoritması Katmanları .....                 | 17 |
| 3.4.2                                                  | Yapay Bağışıklık Sistem Algoritması ve Aykırı Değer Tespit Modeli .. | 17 |
| 3.4.2.1                                                | Yapay Bağışıklık Tabanlı Sistem Algoritması ve Adımları .....        | 18 |
| 3.4.2.2                                                | YBA ile Aykırı Değer Tespiti .....                                   | 20 |
| 3.4.2.2.1                                              | YBA ile Aykırı Değer Tespitinin Şekiller ile Açıklanması .....       | 20 |
| 3.4.2.2.2                                              | YBA ile Aykırı Değer Tespit Modeli .....                             | 25 |
| 3.4.2.2.3                                              | Tipik YBA ile Modellenen YBA Arasındaki Farklar .....                | 31 |
| <b>BÖLÜM 4</b>                                         |                                                                      |    |
| SINIFLANDIRMA YÖNTEMLERİ ve DEĞERLENDİRME ÖLÇÜTÜ ..... |                                                                      | 33 |
| 4.1                                                    | K – En Yakın Komşuluk Algoritması .....                              | 33 |
| 4.2                                                    | Kutu Grafik Yöntemi .....                                            | 35 |
| 4.3                                                    | Uzaklık Tabanlı Aykırı Değer Tespit Algoritması .....                | 36 |
| 4.4                                                    | Değerlendirme Ölçütü: Hata Matrisi .....                             | 38 |
| <b>BÖLÜM 5</b>                                         |                                                                      |    |
| DENEYSEL SONUÇLAR .....                                |                                                                      | 39 |
| 5.1                                                    | Veri Kümeleri .....                                                  | 39 |
| 5.1.1                                                  | Pima Veri Kümesi .....                                               | 39 |
| 5.1.2                                                  | Blood Veri Kümesi .....                                              | 41 |
| 5.1.3                                                  | Yapay Veri Kümesi .....                                              | 43 |
| 5.2                                                    | Deneyisel Analiz Sonuçları .....                                     | 46 |
| 5.2.1                                                  | Deneyisel Analizler .....                                            | 46 |
| 5.2.1.1                                                | Aykırı Veri Sayısı Baz Alınarak Yapılan Analiz .....                 | 46 |
| 5.2.1.2                                                | Eşik Değer İndeksi Baz Alınarak Yapılan Analiz .....                 | 46 |
| 5.2.2                                                  | Pima Veri Kümesi ile Aykırı Değer Analizi .....                      | 47 |
| 5.2.2.1                                                | Aykırı Veri Sayısı Baz Alınarak Yapılan Analiz (Pima) .....          | 47 |
| 5.2.2.2                                                | Eşik Değer İndeksi Baz Alınarak Yapılan Analiz (Pima) .....          | 51 |
| 5.2.3                                                  | Blood Veri Kümesi ile Aykırı Değer Analizi .....                     | 53 |
| 5.2.3.1                                                | Aykırı Veri Sayısı Baz Alınarak Yapılan Analiz (Blood) .....         | 53 |
| 5.2.3.2                                                | Eşik Değer İndeksi Baz Alınarak Yapılan Analiz (Blood) .....         | 57 |
| 5.2.4                                                  | Yapay Veri Kümesi ile Aykırı Değer Analizi .....                     | 59 |
| 5.2.4.1                                                | Aykırı Veri Sayısı Baz Alınarak Yapılan Analiz (Yapay) .....         | 60 |
| 5.2.4.2                                                | Eşik Değer İndeksi Baz Alınarak Yapılan Analiz (Yapay) .....         | 63 |
| 5.2.4.3                                                | Örnek ve Özellik Sayılarına Göre Yapılan Analiz (Yapay) .....        | 66 |
| 5.2.4.3.1                                              | Analiz 1 .....                                                       | 66 |
| 5.2.4.3.2                                              | Analiz 2 .....                                                       | 68 |
| <b>BÖLÜM 6</b>                                         |                                                                      |    |
| SONUÇ VE ÖNERİLER .....                                |                                                                      | 71 |

|                |    |
|----------------|----|
| KAYNAKLAR..... | 73 |
| ÖZGEÇMİŞ.....  | 77 |

## SİMGE LİSTESİ

---

|                 |                                        |
|-----------------|----------------------------------------|
| o               | Normal veri                            |
| •               | Aykırı veri                            |
| AE              | Afinite (Benzeşme) eşiği               |
| Ag <sub>i</sub> | Ag antijen kümesindeki i. antijen      |
| Ag <sub>j</sub> | Ag antijen kümesindeki j. antijen      |
| B               | Bağışıklık Hücresi                     |
| D               | Eğitim kümesi                          |
| d               | Eşik değer mesafesi                    |
| FN              | Yanlış negatif                         |
| FP              | Yanlış pozitif                         |
| k               | En yakın komşu sayısı                  |
| KA              | Atanacak kaynak sayısı                 |
| KD              | Kontrol değeri                         |
| KS              | Klon sayısı                            |
| n               | Hücre / Örnek / Özellik sayısı         |
| p               | Verilerin aykırılık ilişki sayısı      |
| Q <sub>1</sub>  | Birinci dörtte birlik                  |
| Q <sub>2</sub>  | Medyan - Ortanca değer                 |
| Q <sub>3</sub>  | Üçüncü dörtte birlik                   |
| round()         | En yakın tamsayıyı yuvarlayan operatör |
| T               | Bağışıklık Hücresi                     |
| TN              | Doğru negatif                          |
| TP              | Doğru pozitif                          |
| UE              | Uyarılma eşiği                         |
| X               | Test kümesi                            |
| β               | Çarpım vektörü                         |

## KISALTMA LİSTESİ

---

|      |                                                 |
|------|-------------------------------------------------|
| ABH  | Aday Bellek Hücresi                             |
| AIRS | Yapay Bağışıklık Tanıma Sistemi                 |
| AR   | Otoregresif                                     |
| ARB  | Artificial Recognition Ball                     |
| ARMA | Otoregresif - Hareketli Ortalama                |
| BP   | Kutu Grafik                                     |
| CART | Classification and Regression Trees             |
| DBO  | Uzaklık Tabanlı Aykırı Değer Tespit Algoritması |
| DD   | Depth - Versus - Depth                          |
| EDİ  | Eşik Değer İndeksi                              |
| EKK  | En Küçük Kareler                                |
| EKY  | En Kısa Yol                                     |
| GA   | Genetik Algoritması                             |
| GSPR | Genelleştirilmiş Standart Pearson Artıkları     |
| IQR  | Inter Quartile Range                            |
| IRM  | Immune Recruitment Mechanism                    |
| ITS  | Intelligent transportation system               |
| KNN  | K - En Yakın Komşuluk Algoritması               |
| KSA  | Klonal Seçme Algoritması                        |
| LÖH  | Laplasyen Özharitalar                           |
| MA   | Hareketli Ortalama                              |
| NSA  | Negatif Seçim Algoritması                       |
| RR   | Ridge Regresyon                                 |
| SDM  | Sparse Distributed Memory                       |
| TBÇ  | Temel Bileşenler Çözümlemesi                    |
| UBH  | En İyi Uyuşan Bellek Hücresi                    |
| UE   | Uyarılma Eşiği                                  |
| YBA  | Yapay Bağışıklık Sistem Tabanlı Algoritma       |
| YBS  | Yapay Bağışıklık Sistemi                        |



## ŞEKİL LİSTESİ

|                                                                                                          | Sayfa |
|----------------------------------------------------------------------------------------------------------|-------|
| Şekil 2. 1 Aykırı test verisi (Kutu Grafik) .....                                                        | 5     |
| Şekil 3. 1 Bağışıklık Tipleri.....                                                                       | 9     |
| Şekil 3. 2 Bağışıklık sisteminin klonal seçme mekanizması .....                                          | 10    |
| Şekil 3. 3 Klonal seçme algoritması akış diyagramı .....                                                 | 11    |
| Şekil 3. 4 Negatif Seçim algoritması eğitim aşaması .....                                                | 12    |
| Şekil 3. 5 Negatif Seçim algoritması test aşaması .....                                                  | 13    |
| Şekil 3. 6 Yapay Bağışıklık Sisteminin katmanları .....                                                  | 17    |
| Şekil 3. 7 Yapay Bağışıklık algoritması genel adımları .....                                             | 19    |
| Şekil 3. 8 Bellek hücre havuzunun ve ARB havuzunun rastgele oluşturulması.....                           | 21    |
| Şekil 3. 9 Antijenin bellek hücre havuzuna sunulması .....                                               | 21    |
| Şekil 3. 10 En çok uyarılmış bellek hücresinin belirlenmesi .....                                        | 21    |
| Şekil 3. 11 En çok uyarılmış bellek hücresinin mutasyonla çoğaltılarak ARB havuzuna eklenmesi.....       | 22    |
| Şekil 3. 12 Antijenin ARB havuzuna sunulması .....                                                       | 22    |
| Şekil 3. 13 En az uyarılmış hücrelerin ARB havuzundan çıkartılması .....                                 | 23    |
| Şekil 3. 14 Aday bellek hücresinin bulunması ve en iyi uyuşan bellek hücresi ile karşılaştırılması ..... | 23    |
| Şekil 3. 15 Aday bellek hücresinin bellek havuzuna alınması .....                                        | 24    |
| Şekil 3. 16 Veri kümesinin bellek hücre havuzuna sunulması.....                                          | 24    |
| Şekil 3. 17 En çok uyarılan hücreden en az uyarılan hücreye göre sıralama yapılması                      | 24    |
| Şekil 3. 18 İlk adımın şematik görünümü (YBA) .....                                                      | 25    |
| Şekil 3. 19 İkinci adımı özetleyen şematik görünüm (YBA) .....                                           | 27    |
| Şekil 3. 20 Üçüncü adımı özetleyen şematik görünüm (YBA) .....                                           | 28    |
| Şekil 3. 21 Dördüncü adımı özetleyen şematik görünüm (YBA).....                                          | 29    |
| Şekil 3. 22 Beşinci ve son adımı özetleyen şematik görünüm (YBA) .....                                   | 30    |
| Şekil 3. 23 YBA ile tespit edilen aykırı değerlerin doğruluk analizi.....                                | 31    |
| Şekil 4. 1 KNN algoritmasından bir örnek .....                                                           | 34    |
| Şekil 4. 2 Kutu Grafik Yöntemi.....                                                                      | 35    |
| Şekil 5. 1 Pima veri kümesi .....                                                                        | 40    |
| Şekil 5. 2 Pima aykırı değer veri kümesi .....                                                           | 41    |
| Şekil 5. 3 Blood veri kümesi.....                                                                        | 42    |
| Şekil 5. 4 Blood aykırı değer veri kümesi.....                                                           | 43    |
| Şekil 5. 5 Yapay veri kümesi özelliklerinin değer aralıkları.....                                        | 43    |

|             |                                                                      |    |
|-------------|----------------------------------------------------------------------|----|
| Şekil 5. 6  | Yapay veri kümesi özelliklerinin sapma oranları.....                 | 44 |
| Şekil 5. 7  | Yapay veri kümesi .....                                              | 45 |
| Şekil 5. 8  | Yapay aykırı veri kümesi.....                                        | 45 |
| Şekil 5. 9  | Uyarılmaya göre aykırı tespit başarı oranı (Pima veri kümesi) .....  | 49 |
| Şekil 5. 10 | Uyarılmaya göre aykırı tespit başarı oranı (Blood veri kümesi).....  | 55 |
| Şekil 5. 11 | Uyarılmaya göre aykırı tespit başarı oranı (Yapay veri kümesi) ..... | 62 |

## ÇİZELGE LİSTESİ

|               | Sayfa                                                                           |
|---------------|---------------------------------------------------------------------------------|
| Çizelge 2. 1  | Aykırı test verisi ..... 4                                                      |
| Çizelge 2. 2  | Sıralanmış aykırı test verisi ..... 4                                           |
| Çizelge 4. 1  | Benzerlik ölçüm tablosu ..... 37                                                |
| Çizelge 4. 2  | Aykırılık ilişki sayısı ..... 37                                                |
| Çizelge 4. 3  | Hata Matrisi..... 38                                                            |
| Çizelge 5. 1  | Aykırı veri sayısı baz alınarak yapılan aykırı değer tespiti (YBA - Pima) .. 48 |
| Çizelge 5. 2  | Analiz 1-YBA için Hata matrisi (Pima)..... 48                                   |
| Çizelge 5. 3  | Analiz 1-KNN için Hata matrisi (Pima) ..... 50                                  |
| Çizelge 5. 4  | Analiz 1-DBO için Hata matrisi (Pima) ..... 50                                  |
| Çizelge 5. 5  | Analiz 1-BP için Hata matrisi (Pima) ..... 50                                   |
| Çizelge 5. 6  | Pima veri kümesi için YBA performans gösterimi (Analiz 1) ..... 51              |
| Çizelge 5. 7  | Analiz 2-YBA için Hata matrisi (Pima)..... 51                                   |
| Çizelge 5. 8  | Analiz 2-KNN için Hata matrisi (Pima) ..... 52                                  |
| Çizelge 5. 9  | Analiz 2-DBO için Hata matrisi (Pima) ..... 52                                  |
| Çizelge 5. 10 | Analiz 2-BP için Hata matrisi (Pima) ..... 53                                   |
| Çizelge 5. 11 | Pima veri kümesi için YBA performans gösterimi (Analiz 2) ..... 53              |
| Çizelge 5. 12 | Aykırı veri sayısı baz alınarak yapılan aykırı değer tespiti (YBA – Blood) 54   |
| Çizelge 5. 13 | Analiz 1-YBA için Hata matrisi (Blood) ..... 54                                 |
| Çizelge 5. 14 | Analiz 1-KNN için Hata matrisi (Blood)..... 56                                  |
| Çizelge 5. 15 | Analiz 1-DBO için Hata matrisi (Blood)..... 56                                  |
| Çizelge 5. 16 | Analiz 1-BP için Hata matrisi (Blood)..... 56                                   |
| Çizelge 5. 17 | Blood veri kümesi için YBA performans gösterimi (Analiz 1)..... 57              |
| Çizelge 5. 18 | Analiz 2-YBA için Hata matrisi (Blood) ..... 57                                 |
| Çizelge 5. 19 | Analiz 2-KNN için Hata matrisi (Blood)..... 58                                  |
| Çizelge 5. 20 | Analiz 2-DBO için Hata matrisi (Blood)..... 58                                  |
| Çizelge 5. 21 | Analiz 2-BP için Hata matrisi (Blood)..... 59                                   |
| Çizelge 5. 22 | Blood veri kümesi için YBA performans gösterimi (Analiz 2) ..... 59             |
| Çizelge 5. 23 | Aykırı veri sayısı baz alınarak yapılan aykırı değer tespiti (YBA – Yapay) 60   |
| Çizelge 5. 24 | Analiz 1-YBA için Hata matrisi (Yapay) ..... 61                                 |
| Çizelge 5. 25 | Analiz 1-KNN için Hata matrisi (Yapay) ..... 62                                 |
| Çizelge 5. 26 | Analiz 1-DBO için Hata matrisi (Yapay) ..... 63                                 |
| Çizelge 5. 27 | Analiz 1-BP için Hata matrisi (Yapay) ..... 63                                  |
| Çizelge 5. 28 | Yapay veri kümesi için YBA performans gösterimi (Analiz 1)..... 63              |

|               |                                                                    |    |
|---------------|--------------------------------------------------------------------|----|
| Çizelge 5. 29 | Analiz 2-YBA için Hata matrisi (Yapay) .....                       | 64 |
| Çizelge 5. 30 | Analiz 2-KNN için Hata matrisi (Yapay) .....                       | 64 |
| Çizelge 5. 31 | Analiz 2-DBO için Hata matrisi (Yapay) .....                       | 65 |
| Çizelge 5. 32 | Analiz 2-BP için Hata matrisi (Yapay) .....                        | 65 |
| Çizelge 5. 33 | Yapay veri kümesi için YBA performans gösterimi (Analiz 2).....    | 66 |
| Çizelge 5. 34 | Analiz 1-YBA için Hata Matrisi (500 örnek) .....                   | 66 |
| Çizelge 5. 35 | Analiz 1-YBA için Hata Matrisi (750 örnek) .....                   | 67 |
| Çizelge 5. 36 | Örnek sayısındaki değişime bağlı analiz sonuçları (Analiz 1) ..... | 67 |
| Çizelge 5. 37 | Analiz 1-YBA için Hata Matrisi (Özellik çıkarımı) .....            | 68 |
| Çizelge 5. 38 | Analiz 2-YBA için Hata Matrisi (500 örnek) .....                   | 68 |
| Çizelge 5. 39 | Analiz 2-YBA için Hata Matrisi (750 örnek) .....                   | 69 |
| Çizelge 5. 40 | Örnek sayısındaki değişime bağlı analiz sonuçları (Analiz 2) ..... | 69 |
| Çizelge 5. 41 | Analiz 2-YBA için Hata Matrisi (Özellik çıkarımı) .....            | 70 |

## YAPAY BAĞIŞIKLIK SİSTEM TABANLI ALGORİTMA İLE AYKIRI DEĞER TESPİTİ

Mehmet GÜÇLÜ

Bilgisayar Mühendisliği Anabilim Dalı

Yüksek Lisans Tezi

Tez Danışmanı: Doç. Dr. Banu DİRİ

Aykırı değer tespiti; kredi kartı sahtekarlık tespiti, ağ saldırı tespiti, çok boyutlu verilerde hataların ayıklanması, veri trafiğinde sistem yükünün hafifletilmesi gibi birçok alanda başarıyla uygulanan metince zengin bilgice fakir mantığını referans alan bir veri madenciliği metodolojisidir.

Çok boyutlu veri kümelerinde aykırı değerleri tespit etmek oldukça zordur. Aykırı değerleri teşhis etmede etkili olabilecek birçok algoritma olmasına rağmen, bu algoritmalarından birçoğu geniş ve büyük boyutlu veri kümeleri için kullanışsız olmaktadır. Verilerdeki örnek ve özellik sayısı arttıkça veri madenciliği metotlarının aykırı veri tespit etmedeki başarıları da düşmektedir. Bu tez çalışmasında, veri kümelerinde aykırı değerleri büyük bir oranda teşhis edebilen ve diğer metotlara göre daha etkin çözümler üretebilen Yapay Bağışıklık Sistem Paradigmasından yararlanılan bir algoritma kullanılmıştır. Algoritma, ikisi gerçek hayattan alınmış, diğeri rastgele değerlerle oluşturulmuş üç veri kümesi üzerine uygulanmıştır. Algoritmanın performansı, k - En Yakın Komşuluk Algoritması, Uzaklık Tabanlı Aykırı Değer Tespit Algoritması ve Kutu Grafik yöntemlerinin aykırı değer tespit etmedeki başarıları ile karşılaştırılmıştır. Elde edilen sonuçlar kıyaslandığında, aykırı değer tespit etmede Yapay Bağışıklık Sistem Tabanlı algoritmasının daha iyi sonuçlar verdiği ve daha düşük hata oranı ile çalıştığı görülmüştür.

**Anahtar Kelimeler:** Yapay Bağışıklık Sistem Tabanlı Algoritması, Veri Madenciliğı, Aykırı Değer Tespiti, k - En Yakın Komşuluk Algoritması, Uzaklık Tabanlı Aykırı Değer Tespit Algoritması, Kutu Grafik.

**OUTLIER DETECTION BY USING AN ARTIFICIAL IMMUNE SYSTEM-BASED  
ALGORITHM**

Mehmet GÜÇLÜ

Department of Computer Engineering

MSc. Thesis

Advisor: Doç. Dr. Banu DİRİ

Outlier detection, which is applied successfully in many fields such as credit card fraud detection, network intrusion detection, extracting errors from multi-dimensional data and easing the burden of big data systems, is a data mining methodology which follows the principle of textually rich but information poor approach.

It is very difficult to realize the outlier detection in multi-dimensional data sets. Although there are many algorithms that may be effective in outlier detection, many of these algorithms are impractical for big and large-sized data sets. As the number of features and samples in the data increases, the success of the data mining methods in outlier detection decreases. In this dissertation, the artificial immune system-based algorithm that can realize outlier detection in data sets easily and produce solutions that are more effective than other methods was used. The algorithm was applied to the three data sets two of which are taken from two real-life data sets and one of which is made up of artificial data set. The performance of the algorithm was compared with the performances of the k-Nearest Neighbour Algorithm, Distance-Based Outlier Detection Algorithm and Box Plot method in outlier detection. When the results were compared, it was found out that Artificial Immune System-Based Algorithm gives better results and works with a lower error rate than the other methods used in outlier detection.

**Key words:** Artificial Immune System-Based Algorithm, Data Mining, Outlier Detection, k-Nearest Neighbour Algorithm, Distance- Based Outlier Detection Algorithm, Box Plot



### GİRİŞ

#### 1.1 Literatür Özeti

Veri kümelerindeki tüm veriler, belirlenmiş bir sınıflandırılmada tutulmakta ve belirli kurallar sonucunda elde edilebilmektedir. Belirli bir kural ile oluşturulmamış, belirli bir sınıfa ait olmayan, farklı bir şekilde türetilerek elde edilmiş veriler aykırı olarak adlandırılmaktadır. Aykırı olarak adlandırılan bu tip verilerin tespiti üzerine yapılan tez çalışması içerisinde, giriş kısmı dışında ikinci bölümde aykırılıktan bahsedilmiş, aykırılığı daha iyi kavrayabilmek için rastgele değerlerle üretilmiş bir veri kümesi üzerinde aykırılık tespitini gösteren bir örnek verilmiştir. Bu bölümde aykırı değer analizi ile ilgili geçmişte farklı alanlarda farklı metotlar kullanılarak yapılmış çalışmalardan da bahsedilmiştir. Üçüncü bölümde bağışıklık sisteminin temel özelliklerine değinilmiş, Yapay Bağışıklık Sistem Tabanlı Algoritmanın (YBA) adımları ve bu çalışmada modellenen YBA'nın çalışma prensipleri üzerinde durulmuş, YBA'nın farklı kullanım alanlarından bahsedilmiştir. Dördüncü bölümde YBA'nın performansını diğer metotlarla karşılaştırmak için çalışmada kullanılmış K - En Yakın Komşuluk algoritması, Uzaklık Tabanlı Aykırı Değer Tespit algoritması, Kutu Grafik ve Hata Matrisi gibi tekniklerin özelliklerine ve adımlarına yer verilmiştir. Beşinci bölümde veri kümeleri üzerine uygulanan YBA ve diğer metotların deneysel sonuçlarına yer verilerek ayrıntılı bir analiz yapılmıştır. Son bölüm olan altıncı bölümde ise çalışmanın deneysel sonuçları değerlendirilmiş, gelecek çalışmalar ile ilgili önerilerde bulunulmuştur.

#### 1.2 Tezin Amacı

Aykırı verilerin tespitini yapmak, düzeltilebilme şansı varsa düzeltmek veya veri kümesinden çıkarmak için veri madenciliğine ve istatistik bilimine ihtiyaç vardır.

Literatürde farklı birçok aykırı değer tespit yöntemi vardır. Bunun nedeni, veri kümelerinde farklı tipte ve sayılarda görülen aykırılığı tespit etmede farklı çözümlere ihtiyaç duyulmasıdır. Her yöntemin aykırı değer tespit etmede farklı çözümleri, avantajları veya dezavantajları vardır. Bazı metotlar tek boyutlu verilerde daha iyi ve hızlı çözümler üretebilirken bazı metotlar ise çok boyutlu verilerde daha etkili ve hızlı çözümler elde edebilmektedir.

Geçmiş araştırma çalışmalarıyla yapılan aykırı değer analizlerinde tatmin edici başarıya ulaşılmış olsa da, özellik sayısı fazla olan veri kümelerinde çoğu analizler yetersiz kalabilmektedir. Özellikle çok boyutlu veri kümelerinde aykırı değer tespit etmede daha etkin ve hızlı çözümler sunmak için bu tez çalışmasında Yapay Zeka Optimizasyon Algoritmaları'ndan Yapay Bağışıklık Sistem Tabanlı Algoritması kullanılmıştır. Çalışma, algoritma adımlarının gerçekleştirmesini kendine özgü modellemiş, aykırı değer tespit etme alanına farklı bir bakış açısı getirmiştir.

### **1.3 Hipotez**

Son yıllarda yapılan çalışmalarda Yapay Zeka metotlarının sınıflandırma problemlerinde kullanılmaya başlandığı görülmektedir. Aykırı değerlerin tespiti için yapılan çalışmalarda ise Yapay Zeka tekniklerine çok fazla rastlanılmamaktadır. Tez çalışması ile ortaya konulan modellemede, diğer kullanılan makine öğrenmesi yöntemlerine göre daha yüksek performanslı sonuçlar elde edilmiştir. Bu çalışma ile elde edilen kazançlardan biri başarı oranının yüksek olması, diğer kazanç ise sistemin hızlı yanıt vermesi olmuştur. Daha etkili ve hızlı aykırı değer tespit analizi yapabilecek bir model tasarlanabilmesi için Evrimsel Algoritmaların kullanılması önerilebilir.

## BÖLÜM 2

---

### AYKIRILIK

Aykırı değer, bir kümedeki verilerin geneline uymayan ve verinin genel dağılımından sapmış olan anormal değerlerdir. Veri kümelerinde görülen aykırılıklar, hatalı olabileceği gibi gerçeği de yansıtabilir. Bu noktada dikkat edilmesi gereken ilk şey verinin gerçeği yansıtip yansıtmadığına karar vermektir. Çok boyutlu verilerde bir örneğin aykırı olup olmadığını belirleyebilmek oldukça zordur. Nitekim bir örneğin bir özelliğinde görülen aykırılık diğer özelliklerin normal dağılım içerisinde olması durumunda o örneğin normalmiş gibi algılanmasına neden olabilir. Çünkü aykırı değerler ve veriler birbirlerini etkileyebilme hatta gizleyebilme özelliğine sahiptir. Örneğin, bir vektör uzayındaki en küçük ve en büyük örnekler aykırı veriyi temsil ediyor olsun. En büyük verinin aşırı şekilde diğer verilerden sapmış olması durumunda standart sapma yükselir ve sonucunda en küçük değere sahip aykırı veri sanki normal değere sahipmiş gibi işlem görür. Ancak büyük değere sahip aykırı verinin tespit edilmesi ve vektör uzayından çıkartılması ile standart sapma düşer ve küçük aykırı değerlerin sıra dışı olduğu görülür.

Aykırı değerler, tek boyutlu bir küme içerisinde bir tane olabileceği gibi birden fazla da olabilir ya da çok boyutlu verilerde bir örneğin tek bir özelliğinde olabileceği gibi birden fazla özelliğinde de aykırılık görülebilir. Aykırı değerlerin anlaşılmasının zor olduğu, aynı zamanda kötü veriler olduğu ve bu yüzden araştırmacıları yanılttığı söylene de aykırılığı belirleyebilmek amacıyla geliştirilmiş birçok çalışma mevcuttur. Kümeleme Tabanlı, Sınıf Tabanlı, En Yakın Komşuluk Tabanlı, İstatistiksel Tabanlı ve Spektral Ayırma Tabanlı yöntemleri gibi birçok algoritmalar bu çalışmalara örnek gösterilebilir

[1]. Ancak literatürde Yapay Zeka Optimizasyon tekniklerinin aykırı değer tespit çalışmalarında uygulandığına rastlanılmamıştır. Bu yüzden çalışmada, aykırılık tespitinde ilk kez uygulanacak olan Yapay Zeka tekniklerinden Yapay Bağışıklık Sistem Tabanlı Algoritmasına yer verilecektir.

## 2.1 Örnek

Aykırı değer tanımını kavramak için basit bir aykırı test analizi yapalım. [40, 60] değer aralığından oluşan ve rastgele değerlerle oluşturulmuş bir veri kümesine değer aralığından sapmış iki aykırı veri yerleştirelim (Çizelge 2. 1).

Çizelge 2. 1 Aykırı test verisi

|    |    |           |    |           |
|----|----|-----------|----|-----------|
| 45 | 48 | 40        | 55 | 44        |
| 57 | 42 | 53        | 56 | <b>75</b> |
| 48 | 54 | 56        | 57 | 40        |
| 41 | 50 | 58        | 53 | 59        |
| 53 | 47 | 45        | 60 | 47        |
| 50 | 40 | 51        | 43 | 41        |
| 49 | 44 | 48        | 51 | 50        |
| 41 | 59 | <b>18</b> | 46 | 45        |

Kutu Grafiği (Box Plot) yöntemi kullanılarak aykırı test verisine aşağıdaki adımları uygulayalım:

a) Veriler küçükten büyüğe doğru sıralanır (Çizelge 2. 2).

Çizelge 2. 2 Sıralanmış aykırı test verisi

|           |    |    |    |           |
|-----------|----|----|----|-----------|
| <b>18</b> | 43 | 47 | 51 | 56        |
| 40        | 44 | 48 | 51 | 57        |
| 40        | 44 | 48 | 53 | 57        |
| 40        | 45 | 48 | 53 | 58        |
| 41        | 45 | 49 | 53 | 59        |
| 41        | 45 | 50 | 54 | 59        |
| 41        | 46 | 50 | 55 | 60        |
| 42        | 47 | 50 | 56 | <b>75</b> |

b) Verilerin ortalama değerleri alınır. Aykırı test verisi için bu değer 48.975'tir.

c) Ortanca değer (Median –  $Q_2$ ) bulunur. Veri sayısı tek ise verilerin ortasında yer alan değer alınır. Eğer veri sayısı çift ise, ortanca değer bir alt veri ile bir üst

verinin ortalama değeri alınarak bulunur. Bu test verisi için ortanca değeri 48.5'tir.

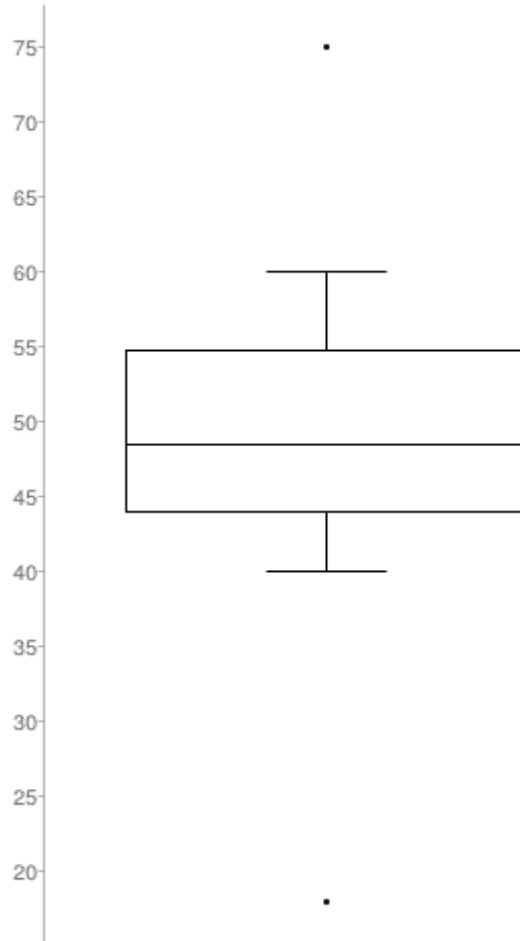
**d)** Veri kümesindeki minimum ve maksimum değerler bulunur. Test verisi için bu değerler minimum için 18 ve maksimum için 75'tir.

**e)** Veri kümesindeki ilk 20 verinin ( $Q_1$ ) ve son 20 verinin ortanca değeri ( $Q_3$ ) bulunur. Örnek veri kümesi için bu değerler sırasıyla 44 ve 54.5'tir.

**f)** İki çeyrek aralığı (Inter Quartile Range IQR) bulunur. Bu değer 10.5'tir.

**g)** Alt ve üst limitler bulunur. Bu değerler; alt limit için 28.25, üst limit için ise 70.25'tir.

Hesaplanan bu değerlere göre çizilmiş kutu grafiğimiz Şekil 2. 1'de gösterilmiştir.



Şekil 2. 1 Aykırı test verisi (Kutu Grafiği)

Şekil 2. 1’de yer alan iki uç nokta aykırı değerleri göstermektedir. Bu veri kümesi için hesaplanan aykırı veriler 18 ve 75’tir. Yani, [28.25, 70.25] aralığında yer alan değerler dışındaki tüm örnekler bu veri kümesi için aykırılığı temsil edecektir.

Bu bölümde kullanılan kutu grafiği tekniği dördüncü bölümde detaylı bir şekilde anlatılmıştır.

## **2.2 Aykırı Değer Tespiti ile İlgili Bazı Çalışmalar**

Aykırı değerlerin modellenmesi ve bu değerlerin tespiti ile ilgili farklı alanlarda geçmiş yıllarda yapılmış birçok çalışma mevcuttur. Kaya (1995), Kimyasal üretim sürecinden iki saatte bir elde edilmiş 197 gözlem üzerinde kalite kontrol işlemi yapmış, yapılan işlem sonucunda 2 aykırı gözlem tespit etmiştir. Aykırı gözlem değerlerini çıkarttıktan sonra geri kalan gözlemler üzerinden tekrar kalite kontrol işlemi yapmış, aykırı gözlemlere rastlanılmadığını göstermiştir. Aykırı gözlemlerin parametre değerleri ile sunulan modelin parametre değerlerini karşılaştırarak, kalite kontrol yorumları yapmıştır [2]. Bulut (2001), Aykırı değer kavramına ve aykırı verilerle karşılaşıldığında yapılması gerekli olan adımlara değinmiş, Robust Temel Bileşen analizini aykırı değerlerin tespitinde kullanmıştır. İstatistiksel yöntemlerle elde edilmiş veri kümesine Robust Temel Bileşen analizini uygulayarak aykırı değer tespiti yapmıştır [3]. Aşıkil (2006), Çoklu Doğrusal Regresyon kullanarak aykırı gözlemleri incelemiştir. Veri kümeleri üzerine çeşitli metotlar uygulayarak, yöntemlerin aykırı değer başarılarını araştırmıştır [4]. Vural (2007), aykırı değer tespitinde farklı metotları uygulamış, En Küçük Kareler (EKK) yöntemine göre çalışmasında kullandığı yöntemlerin üstünlüklerini göstermiştir [5]. Gürünlü Alma (2009), Genetik Algoritma Tabanlı Çoklu Regresyon modellerinde, bilgi kriterlerini kullanarak aykırı değerlerin belirlenmesi üzerine çalışma yapmıştır [6]. Doğan (2009), aykırı durum tespiti, kümeleme ve sınıflandırma gibi veri madenciliği paradigmalarını kullanarak ağ güvenliği alanına yeni bir yaklaşım getirmiştir. Kümeleme ve Sınıflandırma için En Yakın K Komşuluk (KNN) algoritmasını kullanmıştır. Ağdaki kullanıcılar, k en yakın komşu ve eşik değer parametresinin hesaplanmasıyla kümelendirilmiştir. Bu kümeler üzerinde yapılan işlemler sonucunda yeni isteklerin bir aykırı durum mu yoksa normal mi oldukları ayırt edilmiştir [7]. Chen, Wang ve Van Zuylen (2010), farklı boyutlarda olan iki trafik veri kümesine istatistiksel tabanlı, uzaklık

tabanlı ve yoğunluk tabanlı kümeleme metotlarını uygulayarak Akıllı Ulaşım Sistemleri (ITS) için aykırı tespit çalışması yapmıştır [8]. Yıldırım (2010), çoklu bağlantı problemi ve aykırı değerlerin tespiti üzerine çalışma yapmıştır. Çoklu bağlantı problemi için Ridge Regresyon (RR) yöntemini, aykırı değerlerin tespitinde ise hata hesaplama yöntemlerini kullanmıştır [9]. Turgut (2010), En Küçük Hacimli Elipsoid, En Küçük Kovaryans Determinantı ve Hızlı En Küçük Kovaryans Determinantı gibi yöntemlere alternatif olarak geliştirilmiş En Küçük Vektör Varyansı yöntemini çok boyutlu veri kümelerinde aykırı değer tespitinde kullanmıştır. En Küçük Vektör Varyansı yöntemini diğer yöntemlerle karşılaştırmış, çok boyutlu veri kümelerindeki aykırı tespit başarısını ve hesaplama hızını göstermiştir [10]. Koğar (2010), Tek boyutlu veriler için aykırı değer belirlemede; Z puanına dönüştürme, Kutu Grafiği, normal dağılım eğrisine sahip Histogram Grafiği ve Normal Olasılık Grafiği yöntemlerini kullanmıştır. Çok boyutlu verilerde aykırı değer tespiti için ise Mahalanobis Uzaklığı, Cook Uzaklığı ve Kaldıraç Noktası yöntemlerini çalışmasında kullanmıştır [11]. Chen, Miao ve Zhang (2010), Komşuluk tabanlı algoritma kullanarak aykırı değer tespit çalışması yapmış; deneysel sonuçlarını, Sinir Ağları (RNN), En Yakın K Komşuluk (KNN) ve Uzanlık Tabanlı (DBO) aykırı tespit algoritmalarıyla karşılaştırmıştır. Önerdiği komşuluk modeli diğer metotlara kıyasla daha iyi bir performans göstermiştir [12]. Yavuzkanat (2011), 2009 Elit Bayan Voleybol liginde ve firma iflaslarına ilişkin verilere Genelleştirilmiş Standart Pearson Artıkları (GSPR) yöntemini, çoklu aykırı gözlemlerin belirlenmesi amacıyla uygulamış; önerdiği yaklaşımın başarılarını ortaya koymuştur [13]. Erdoğan (2012), Etkin-Aykırı, Yerel Aykırılık Etkeni, Tek Sınıflı Karar Vektör Makineleri ve Parzen Pencere yöntemi yöntemlerini aykırı değer analizinde kullanılmış ve bu yöntemleri Temel Bileşenler Çözümlemesi (TBÇ), Laplasyen Özharitalar (LÖH) ve Çok Boyutlu Ölçekleme teknikleriyle birleştirerek farklı veri kümeleri üzerinde aykırı değer tespit çalışması yapmıştır. Deney sonuçlarında, LÖH izgesel yönteminin diğer yöntemlere göre daha başarılı performans sergilediğini göstermiştir [14]. Wong, Ray, Stephens ve Lewis (2012), güvenlik yönetimi üzerine yaptıkları çalışmada, Yapay Bağışıklık Sistemini (YBS) kredi kartı dolandırıcılık tespiti için modellemişlerdir. Önerdikleri bu sistem güvenlik yönetimi uygulamalarına yeni bir yaklaşım getirmiştir [15].

### YAPAY BAĞIŞIKLIK SİSTEMİ VE SİSTEM TABANLI ALGORİTMASI

Bu bölümde, Bağışıklık Sistemine kavramsal bir giriş yapılacaktır, devamında Yapay Bağışıklık Sistem Tabanlı algoritma ve algoritmanın adımları detaylı bir şekilde anlatılacaktır.

#### 3.1 Bağışıklık Sistemi

Bağışıklık, insan vücuduna zarar verebilecek her türlü etkenlere karşı vücudu korur ve ona direnç kazandırır. Bağışıklık sisteminin 2 temel savunma sistemi vardır: Doğal Bağışıklık ve Edinsel (Kazanılmış) Bağışıklık.

##### 3.1.1 Doğal Bağışıklık

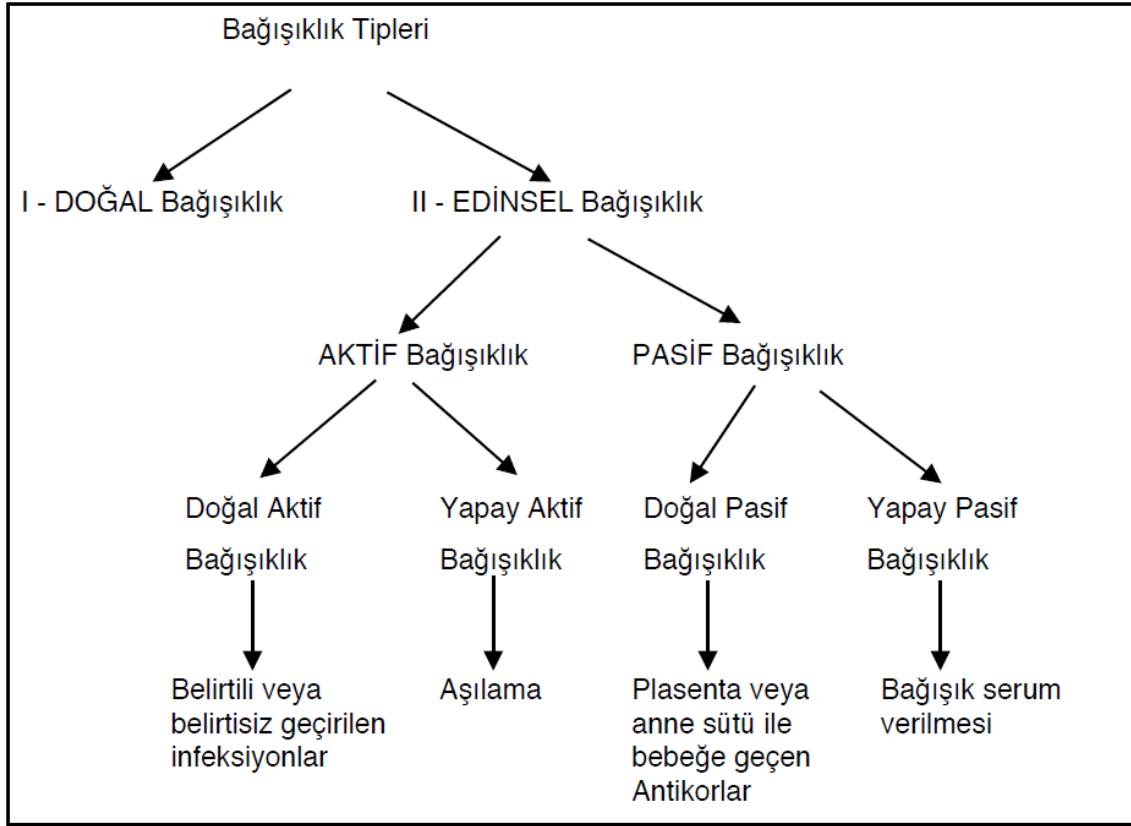
Yapısal ve genetik özelliklerine göre canlılarda doğal olarak bulunan, vücuda giren mikroorganizmaları ayırt etmeden bütünsel olarak değerlendiren ve vücudu hastalıklara karşı koruyan dirençtir [16].

##### 3.1.2 Edinsel (Kazanılmış) Bağışıklık

Vücuda giren mikroplara karşı özel olarak üretilen savunma direncidir. Edinsel bağışıklık, doğuştan edinilmiş bir mekanizma değildir, tam aksine sonradan kazanılmış bir savunma mekanizmasıdır. Edinsel bağışıklık, aktif ve pasif bağışıklık olmak üzere iki farklı şekilde kazanılır. Aktif bağışıklık, ya daha önce geçirilmiş hastalıklardan dolayı doğal yolla veya mikroplardan alınan örneklerden özel olarak hazırlanmış aşılardan uygulanması sonucu yapay yolla kazanılır. Pasif bağışıklık, daha önce vücuda giren mikroplara karşı üretilmiş savunma yanıtının korunmak istenen başka bir canlıya



enjekte edilmesi sonucu dolaylı yoldan kazanılmış bağışıklık türüdür. Şekil 3. 1’de bağışıklık mekanizması gösterilmektedir.



Şekil 3. 1 Bağışıklık Tipleri [16]

Sonuç olarak, bağışıklık sistemi, doğal da olsa sonradan kazanılmış da olsa vücudu yabancı mikroorganizmalara karşı korumak ve vücuda direnç kazandırmak için bir bütün içinde çalışır [16].

### 3.2 Bağışıklık Algoritmaları

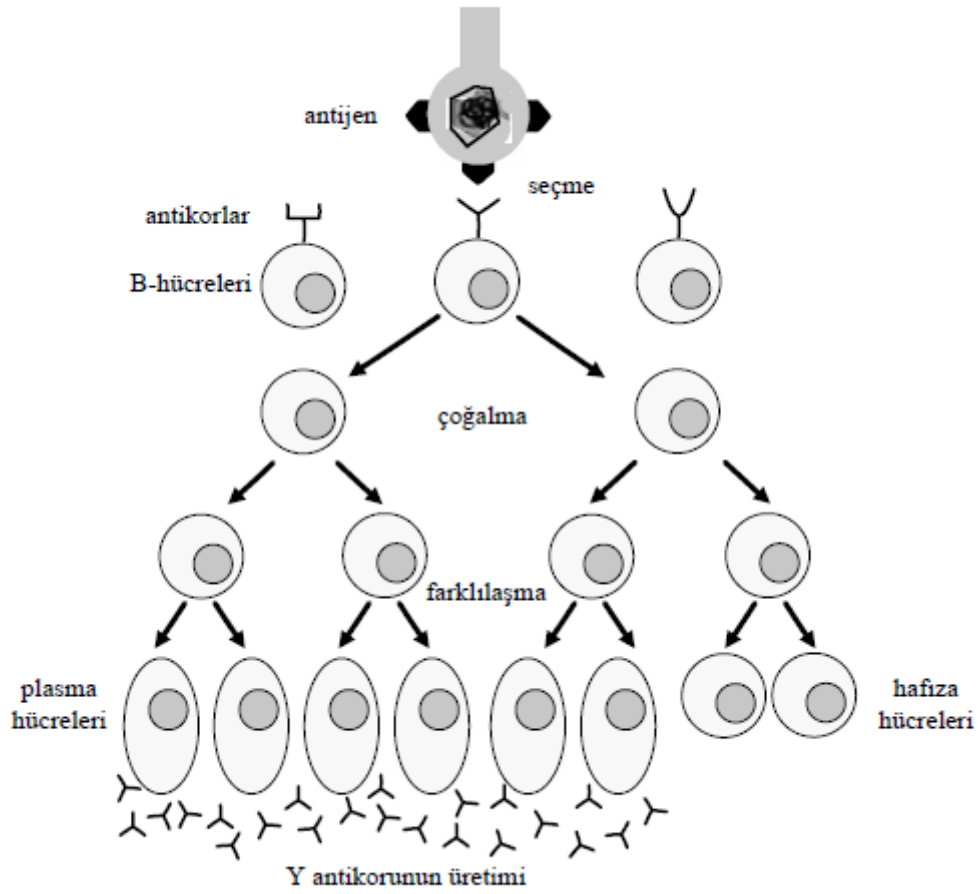
Bu bölümde üç farklı bağışıklık algoritması üzerinde durulmuş, bu algoritmaların çalışma prensipleri çok fazla detaya girilmeden anlatılmıştır.

#### 3.2.1 Klonal Seçme Prensibi ve Algoritması

Bağışıklılığın sürekliliğini sağlamak için hücre ve organların yenilenmesine ihtiyaç vardır. Hücrelerin yenilenip daha güçlü bir savunma mekanizması oluşturulabilmesi ancak çoğalma işlemi ile gerçekleştirilir [17].

### 3.2.1.1 Klonal Seçme Prensibi

Bir antijen tespit edildiğinde antijeni en iyi tanıyacak olan B hücreleri uyarılır. Uyarılan B hücreleri bölünerek çoğalır. Antijeni en iyi tanıyan hücrelerden daha fazla kopya hücreler oluşturulur. Yeni hücreler hipermutasyona uğratılır ve böylece antijenle daha çok uyum sağlayacak olan hücreler üretilmiş olur. Şekil 3. 2’de klonal seçme mekanizması gösterilmektedir.



Şekil 3. 2 Bağışıklık sisteminin klonal seçme mekanizması [18]

Sonuçta, bağışıklık sistemi tarafından antijeni en iyi tanıyacak olan güçlü ve savunma mekanizması yüksek olan hücreler geliştirilir [17].

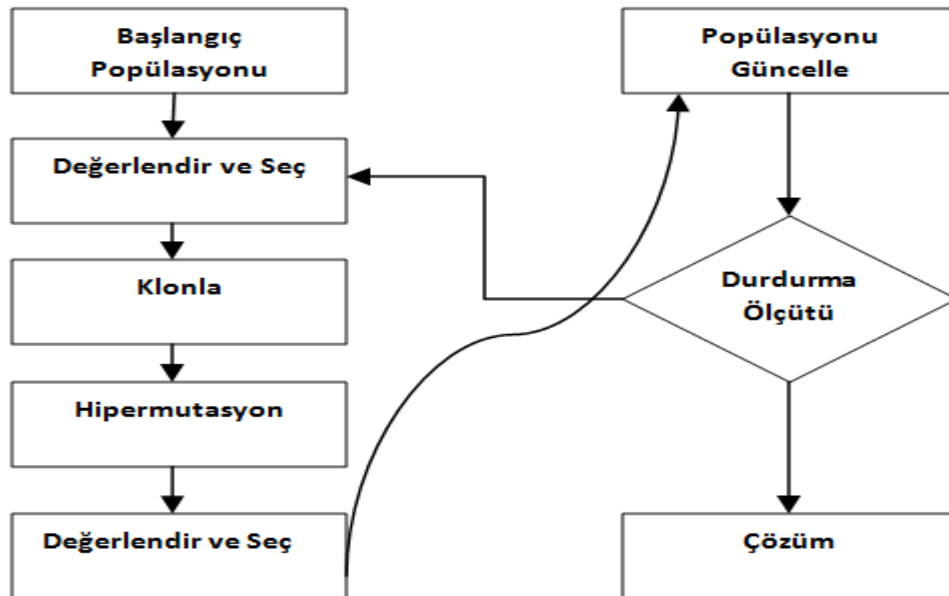
### 3.2.1.2 Klonal Seçme Algoritması

Bağışıklık sisteminden esinlenerek geliştirilen Klonal Seçme Algoritması (KSA), bağışıklık sisteminin önemli bir adımını oluşturmaktadır. Algoritmanın çalışma prensibini şu şekilde açıklayabiliriz (Şekil 3. 3): İlk olarak rastgele değerler alınarak başlangıç

popülasyonu oluşturulur. Popülasyondaki her bir hücre için uygunluk değeri hesaplanır. Uygunluk değeri en yüksek olan hücrelerden n tanesi alınarak hücrelerin uygunluk değerleri ile orantılı olacak şekilde klonlar oluşturulur. Oluşturulacak olan klon sayısı,

$$Nc = \sum_{i=1}^n \text{round}\left(\frac{\beta Np}{i}\right) \quad (3.1)$$

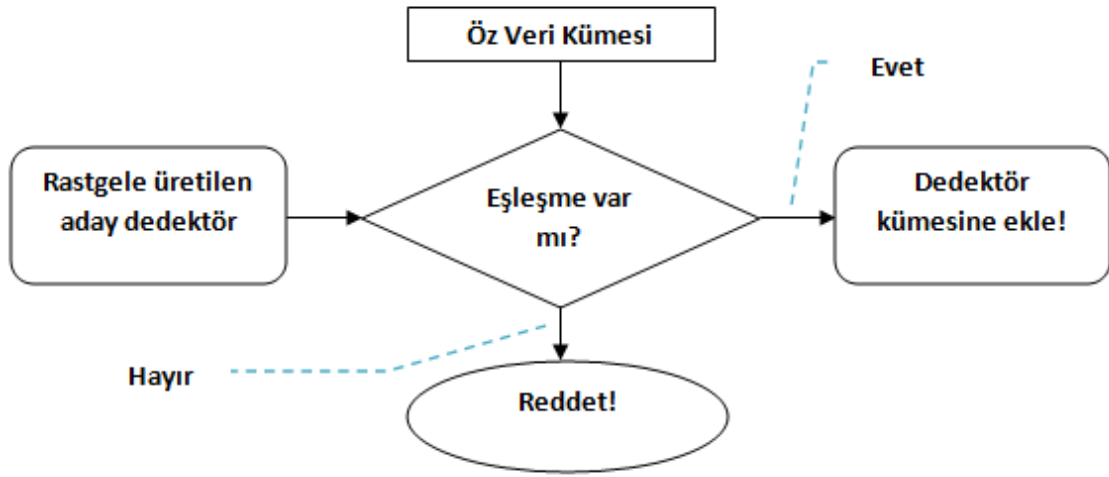
(3.1) gibi hesaplanır.  $\beta$  sabit çarpım katsayısını,  $\text{round}()$  fonksiyonu ise en yakın tamsayıyı veren operatörü gösterir. Klonlar uygunluk değerleriyle ters orantılı olacak şekilde mutasyona uğrattılır. Yani, uygunluk değeri yüksek olan yeni hücreler daha az mutasyona uğrattılır. Mutasyon işleminden sonra klonların uygunluk değerleri hesaplanır. Her bir hücre ve bu hücreden oluşturulmuş klonlar bir alt popülasyonu simgeler. Alt popülasyonda uygunluk değeri yüksek olan hücrenin yaşamasına izin verilir. Uygunluk değeri düşük olan hücreler ise rastgele hücrelerle değiştirilerek sistemden diskalifiye edilirler. İlk adım dışındaki tüm bu işlemler durdurma ölçütü sağlanana kadar devam eder [17]. Tüm adımların tamamlanması ile birlikte popülasyondaki farklılaşma sağlanmış olur.



Şekil 3. 3 Klonal seçme algoritması akış diyagramı [19]

### 3.2.2 Negatif Seçim Algoritması

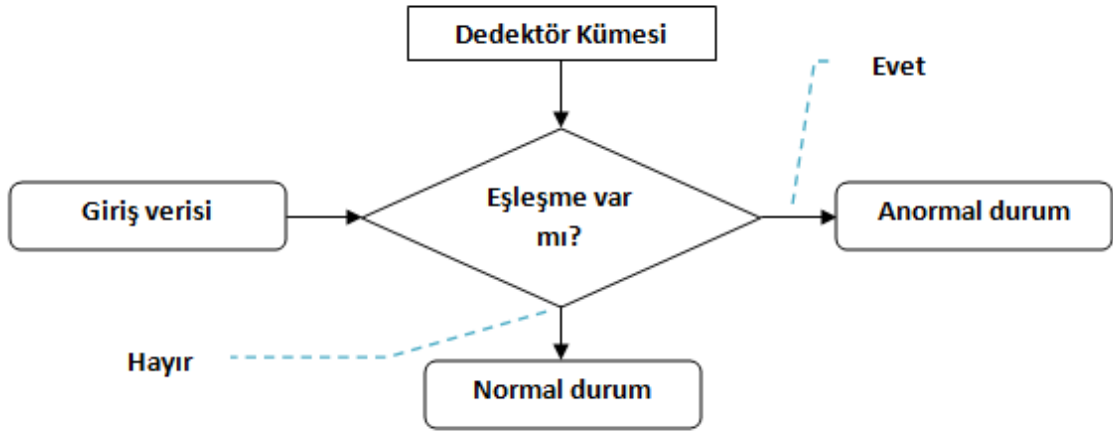
Algoritma, bir antijenin vücuttaki antijen mi yoksa yabancı bir antijen mi olduğuna karar vermek için negatif seçim yapar [20]. Negatif Seçim algoritmasında ilk adım olarak öz veri kümesi oluşturulur. Öz hücreleri belirledikten sonra, rastgele alınan hücrelerden aday dedektör hücreleri belirlenir. Aday dedektör hücreleri ile öz hücreler karşılaştırılır. Karşılaştırma sonucunda eşleşen hücreler çıkartılır, eşleşmeyen hücreler ise dedektör kümesine eklenir [21]. Eğitim aşamasındaki bu adımlar şekil 3. 4'te gösterilmiştir.



Şekil 3. 4 Negatif Seçim algoritması eğitim aşaması

Bir aday dedektör hücresi ile öz hücre arasındaki benzerlik hesaplanır. Benzerlik ölçümü iki hücre arasındaki uzaklık mesafesi alınarak yapılır.

Antijen kümesi, öz veri kümesi ile eşleştirilir. Eğer eşleşme olmaz ise antijen kümesi dedektör kümesine eklenir. Dedektör kümesine belirli sayıda hücre eklenir ve eklemenin tamamlanmasının ardından test aşamasına geçilir. Test aşamasında, gelen bir örnek küme dedektör kümesi ile eşleştirilir. Gelen örnek kümeden herhangi bir dedektör hücresi eşleşiyor ise anormal durum tespit edilir [17]. Test aşamasındaki bu adımlar da Şekil 3. 5'te gösterilmiştir.



Şekil 3. 5 Negatif Seçim algoritması test aşaması

### 3.2.3 AIRS Algoritması

Yapay Bağışıklık Tanıma Sistemi (AIRS) sınıflandırılma çalışmalarında kullanılmış ve birçok araştırmalarda iyileştirilmiş bir optimizasyon algoritmasıdır. Watkins (2001), bağışıklık sisteminden esinlenerek bir öğrenme algoritması geliştirmiştir. Yapay Bağışıklık Tanıma Sistemini gerçek hayattan alınmış veri kümeleri üzerine uygulamış, önerdiği yaklaşımın performansını literatürde kullanılan sınıflandırıcı metotlarının performansıyla karşılaştırmıştır [22]. Watkins ve Timmis (2002), Yapay Bağışıklık Tanıma Sistemi üzerinde iyileştirmeler yaparak, yeni versiyon ile eski versiyon karşılaştırılmıştır. Çalışmada bellek hücre sayısında azalma görülmüş olsa da sınıflandırılma başarısında çok fazla değişme olmamıştır [23]. Kodaz (2007), doktora tezi çalışmasında, AIRS tabanlı bir sınıflandırma yöntemi geliştirmiştir. Çalışmasında standart veri kümelerini ve gerçek hayattan alınmış veri kümelerini kullanmış, önerdiği yaklaşımın sınıflandırma doğruluğunu AIRS algoritması ile karşılaştırmıştır [24].

AIRS algoritması temelde 5 adımdan oluşmaktadır: İklendirme, antijenlerin eğitimi, kaynak için yarışma, bellek hücrenin oluşturulması ve sınıflandırılma. 2., 3., ve 4. adımlar bellek hücre havuzuna sunulan her bir antijen için ayrı ayrı uygulanırken, 1. ve 5. adımlar sadece bir kez uygulanır.

Algoritmanın adımlarını aşağıdaki gibi açıklayabiliriz [25]:

- 1. İklendirme:** Başlangıç popülasyonu oluşturulur. Veri kümeleri [ 0,1 ] değer aralığına normalize edilir ve iki veri arasındaki uzaklık [ 0,1 ] aralığında tutulur.

**II. Antijen Eğitimi:** Antijenler bellek hücre havuzuna ayrı ayrı sunulur. Bellek hücre havuzuna sunulan bir antijenin, havuzdaki hücrelerle olan uyarılma değeri hesaplanır. En yüksek uyarılmaya sahip hücre bulunur ve bu hücreden klonlar oluşturulur. Yeni oluşturulan hücreler mutasyona tabi tutulur. Son olarak elde edilen yeni hücreler ile en iyi uyuşan bellek hücresi Yapay Tanıma Havuzuna (ARB) eklenir.

**III. Kaynak için yarışma:** Aynı antijen ARB havuzuna sunulur. Antijenin havuzdaki hücrelerle olan uyarılma değeri hesaplanır. Uyarılma değerine göre sınırlı sayıda kaynak ataması yapılır. Kaynak sayısındaki artışın belirlenecek sınırı aşması durumunda yeterli kaynağa sahip olamayan hücreden başlanarak havuzdan atılma işlemi yapılır. Yeterli kaynak üretilip ya da üretilmediği durma kriterinin sağlanmasına bakılarak belirlenir. Durma kriteri sağlanıyorsa diğer adıma geçilir, aksi halde yapılan kaynak atamalarından mutasyona uğramış yeni hücreler oluşturulur. Bu işlemler durma kriteri sağlanıncaya kadar devam eder.

**IV. Bellek Hücre Oluşumu:** En yüksek uyarılma değerine sahip ARB, aday bellek hücresi olarak belirlenir. Aday bellek hücresinin uyarılma değeri en iyi uyuşan bellek hücresinin uyarılma değeri ile karşılaştırılır. Aday bellek hücresinin uyarılma değeri yüksek ise, bellek hücre havuzuna eklenir, aksi halde en iyi uyuşan bellek hücresi, bellek hücre havuzunda kalıcı olarak kalır. Tüm antijenler için 2., 3. ve 4. adımın uygulanmasının ardından bellek hücresi oluşturulmuş olur.

**V. Sınıflandırma:** Veri kümesindeki tüm antijenler bellek hücresine sunulur. K En Yakın Komşuluk Algoritması yöntemine göre sınıflandırma işlemi yapılır.

### **3.3 Bağışıklık Sistemleri ile Modellenmiş Bazı Çalışmalar**

Yapay Bağışıklık Sistem Tabanlı algoritması ile ilgili yapılan çalışmalara baktığımızda, farklı bakış açıları ile bağışıklık sistemini modelleyen yaklaşımlar mevcuttur. Smith, Forrest ve Perelson (1999), bağışıklık hafızaları ile ayrık dağıtılmış hafızaların (Sparse Distributed Memory - SDM) aynı sınıfa ait olduğunu açıklamıştır. Yani SDM'deki kuvvetli bölgeler ile bağışıklık sistemindeki B ve T hücreleri arasındaki uyum gösterilmiştir [26]. Forrest (1994), Doğal bağışıklık sisteminden esinlenerek, T hücrelerinin negatif seçimine dayalı bir tespit algoritması geliştirmiştir. Bir hücrenin,

vücudun kendi hücrelerine mi ait olduğu yoksa yabancı bir hücreye mi ait olduğuna karar vermede kullanılan negatif seçim algoritması, rastgele algılayıcılar üreterek bunlardan öz olanları çıkartma işlemini yapar. Böylece, kalan T hücreleri öz olmayan hücreyi rahatlıkla algılayabilir [27].

Bağışıklık sisteminin problem çözme tekniklerini geliştirmedeki başarıları büyük bir ilgi odağı haline gelmiş, çalışmaların birkaçı bağışıklık ilkelerine dayandırılarak ortaya konulmuştur. Fakat bu çalışmalarda, bağışıklık sistemi esin kaynağı alınırken sistemin belirli işlevsel bölgeleri kullanılmıştır. Bazı çalışmalarda ise, algoritmalar çoğalmaya ve yenilenmeye model oluşturmak için geliştirilmiştir. Diğer araştırmalar bağışıklık sistemini merkezi bir sistem olarak ele almış, bağışıklık sistemi ile diğer sistemler arasındaki benzerlikleri sergilemiştir [28]. Dapgusta (1998), bağışıklık sistemini çok temsilcili bir sistem olarak tanımlamıştır. Çok temsilcili sistemin fonksiyonlarını ve işlevselliğini göstermiştir. Temsilciler bağışıklık cevabı oluşturmak için serbestçe hareket eder, birbirlerini etkiler ve yabancı antijenlere karşı maksimum korumayı sağlamak için özel cevaplar üretirler [29].

Hightower, Forrest ve Perelson (1996), antikorların gelişiminde kullanılan somatik mutasyon mekanizmasını modellerken Genetik Algoritmaları kullanmıştır. Genetik Algoritmalar antikorların gelişiminde kullanılmış, sağlam ve etkili antikorların geliştirilmesi ile ilgili yaklaşımlar ortaya atılmıştır [30]. Farmer, Packard ve Perelson (1986), bağışıklık sistemi ile sınıflandırıcı makine öğrenmesi yöntemlerini karşılaştırmıştır [31]. Bersini ve Varela (1991), paralel tepe tırmanma ve paralel ivmelenme için bağışıklık sisteminin yeni katılım mekanizmasını (Immune Recruitment Mechanism-IRM) kullanmışlardır. IRM ve genetik IRM, şekil uzayında herhangi bir popülasyondan aday katmak için geliştirilmiştir [32]. Forest (1993), gürültülü bir ortamda desen algılama için bağışıklık algoritmalarını kullanmıştır. Çalışma sayesinde çeşitli desen sınıflarının kapsamı ve devamlılığı sağlanmıştır [33].

Son yıllarda yapılan çalışmalara baktığımızda ise, Genetik Algoritmalarla birlikte Yapay Bağışıklık Sistem Algoritmasının da araştırma çalışmalarında önemli bir rol aldığını görmekteyiz. Greensmith (2003), sınıf sayısı çok olan internet dökümanlarını sınıflandırmak için Yapay Bağışıklık Tanıma Sistemini kullanmıştır [34]. Hamaker ve

Boggess (2004), sınıflandırma problemlerinde, öklid uzaklık ölçütü dışındaki diğer uzaklık ölçütlerini kullanarak Yapay Bağışıklık Tanıma Sisteminin performansını araştırmıştır [35]. Parmaksızoğlu (2005), Yapay Bağışıklık Sistem Tabanlı Algoritmasını lineer sistemlerin kimliklendirilmesinde kullanmıştır. Lineer sistemlerden AR (Otoregresif), MA (Hareketli Ortalama) ve ARMA (Otoregresif - Hareketli Ortalama) sistemler YBA kullanılarak kimliklendirilmiştir. Bu kimliklendirme sistemlerde hem gürültüsüz hem de gürültülü giriş işareti kullanılarak yapılmıştır [36]. Watkins (2005), doktora tezi çalışmasında, bağışıklık sistemi üzerine detaylı araştırmalar yapmış, AIRS ve klonal seçme algoritmasının yeni versiyonlarını geliştirmiştir [37]. Kaymaz (2007), tıbbi verilerin sınıflandırılmasında Yapay Bağışıklık Sistemini ve KNN sınıflandırma algoritmasını kullanmıştır. Çalışmasında, Yapay Bağışıklık Sistemini kullanarak verilerin özellikleri seçilmiş, k-NN yöntemini kullanarak da verilerden indirgenen özellikleri sınıflandırmıştır [38]. Çatal (2008), yazılım kusur kestirimi problemini, Yapay Bağışıklık Sistem Paradigması Tabanlı Algoritmalarla çözmeye çalışmış ve literatürde var olan YBS tabanlı algoritmaları analiz ederek yeni modeller sunmuştur. Yazılım kusur kestirimi için YBS tabanlı sınıflandırma algoritmalarının performanslarını analiz etmiştir. Özellik azaltma yöntemlerini de kullanarak “Korelasyon Tabanlı Özellik Seçimi” metodunun uygulandığı AIRS tabanlı bir model tasarlamıştır [39]. Karlı (2010), farklı büyüklükteki ağlar için Genetik Algoritma ve Yapay Bağışıklık Sistem Tabanlı algoritmasını kullanmıştır. Önerdiği Genetik Algoritma ve YBS modelinin stokastik En Kısa Yol (EKY) problemlerini çözmede etkili olduklarını ve EKY problemini, YBS modelinin GA’ya göre daha kısa sürede çözdüğünü göstermiştir [40].

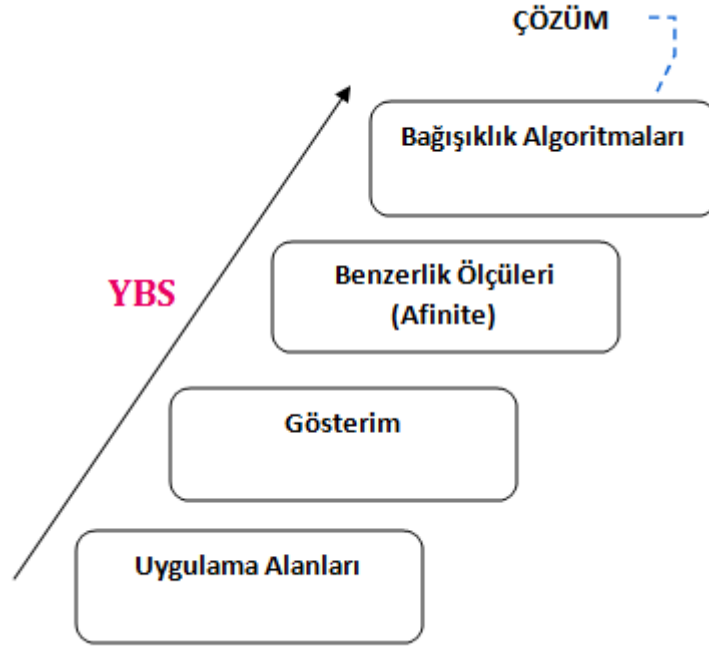
### **3.4 Yapay Bağışıklık Sistemi ve Sistem Tabanlı Algoritması**

Bağışıklık sistemine ait fonksiyonların irdelenmesi, açıklanması ve sistemin model olarak alınması ile yapay zeka tekniklerine verilen önemin artması sonucu Yapay Bağışıklık Sistemi (YBS) ortaya çıkmıştır. YBS, karmaşık problemlere uygulanabilen, bağışıklık sisteminden esinlenerek sistemin fonksiyonlarını model olarak kullanan ve onları taklit eden bir sistem olarak karşımıza çıkar [41].



### 3.4.1 Yapay Bağışıklık Sistem Algoritması Katmanları

Karmaşık bir yapıya sahip olan bağışıklık sistemini tasarlayabilmek için bağışıklık sisteminin hangi elemanlarına ya da fonksiyonel yapısına ihtiyaç duyulacağını bilmesi gerekir. Model alınacak bağışıklık elamanlarının sistemin diğer elemanlarıyla veya fonksiyonlarıyla olan etkileşimleri hesap edilmeli ve olası tüm durumlara karşı göstereceği davranışlar önceden kestirilmelidir.



Şekil 3. 6 Yapay Bağışıklık Sisteminin katmanları

Örneğin, yeni bir Yapay Bağışıklık Tabanlı Sistem tasarlamak için öncelikle uygulama alanı oluşturulmalıdır. Sistemin sahip olduğu birimlerin nasıl etkileşim halinde olacağı açıklanmalı, son olarak da kullanılacak olan benzerlik ölçütlerine göre uzaklık metodu seçilmelidir. Uzaklık ölçümü yaparken verinin niteliği ve boyutu mutlaka göz önünde bulundurulmalıdır. Çünkü kullanılacak metot, hızlı ve kolay şekilde hesaplanabilir olmalıdır [38].

### 3.4.2 Yapay Bağışıklık Sistem Algoritması ve Aykırı Değer Tespit Modeli

Algoritmanın adımları bire bir bu çalışmamızda uygulanmış olup, bazı adımlar aykırı değer tespitine özgü geliştirilerek farklı bir model oluşturulmaya çalışılmıştır. YBA, temelde beş adımla modellenmesine rağmen, bazı çalışmalarda çalışmanın

gereksinimlerine göre adımların deęiřtięi g r lmektedir. Aykırı deęer tespit modelinde algoritmanın temel olan adımları kullanılmış olsa da, bařlangıçtaki verilerin eęitilmesi ve algoritmanın son adımında yer alan sınıflandırmanın yerine aykırılıkların farklı metotlarla çıkarımının s z konusu olmasından dolayı ayrı bir model oluřturduęumuzu s yleyebiliriz.

Bu b l mde Yapay Baęıřıklık Sistem algoritması hakkında temel bilgiler verilecek olup, aykırı deęer tespitine dayalı algoritmanın adımları detaylı bir řekilde a ıklanacaktır.

#### **3.4.2.1 Yapay Baęıřıklık Tabanlı Sistem Algoritması ve Adımları**

Yapay Baęıřıklık Sistem Tabanlı algoritmasında antijen, veri k memizdeki her bir  rneęe karřılıklı gelmektedir. Antikorlar ise, veri k mesinden rastgele se ilmiş bir vekt r olarak tanımlanır. Rastgele se ilmiş bu vekt rler, veri k mesinden se ilmiş herhangi bir satır olup, bu satır sayısı  alıřmada veri k mesindeki  rnek sayısının belirli bir oranı alınarak belirlenmiřtir. Antikorları i eren bu vekt r k mesi eęitim ařamasında bellek h cre havuzu olarak adlandırılmakta, eęitim ařamasından sonra test ařamasında kullanılacak olan bellek h creleri olarak sistemde yer almaktadır. Bellek h creleri, bellek h cre havuzunun evrimleřtirilmiř ve geliřtirilmiř řeklini g sterir. Evrimleřtirmede kullanılan ARB (Artificial Recognition Ball) havuzu, yapay tanıma k resi, antijenle ile en iyi uyulařan antikorun belirlenmesinde ve bu antikorun bellek h cre havuzuna alınmasında rol oynar. Bir antijen i in geliřtirilmiř saęlam antikorlardan o antijen ile en uyulařan antikorun bellek h cre havuzuna yerleřik olarak alınması s z konusudur [39]. ARB havuzu, bellek h cre havuzunda olduęu gibi, veri k mesindeki satırlardan rastgele  rneklerin alınmasıyla oluřturulur. B ylece eęitim ařaması tamamlanır ve algoritmanın adımları uygulanmaya bařlanır.

Yapay Baęıřıklık Sistem algoritması temel olarak 5 ařama ile ger ekleřtirilir. Bu ařamalardan ilki eęitim s recini ve son adımı ise test ařamasını kapsadıęından bu adımlar bir kez uygulanır. Geri kalan adımlar ise, veri k mesindeki her bir  rnek i in ayrı ayrı ger ekleřtirilir.

Algoritmanın adımlarını Şekil 3. 7'deki gibi sıralayabiliriz:

**Adım 1:**

Başlangıç popülasyonunu oluşturma  
(Bellek hücre havuzu ve ARB havuzu)

**Adım 2:**

Bellek hücresinin ve ARB havuzunun üretilmesi  
(Uyarılma, mutasyon, v.s)

**Adım 3:**

ARB havuzunda aday bellek hücresinin geliştirilmesi  
(Kaynak yarışması)

**Adım 4:**

Bellek hücresinin oluşturulması ve sisteme tanıtılması

**Adım 5:**

Aykırı değer tespitinin yapılması

Şekil 3. 7 Yapay Bağışıklık Sistem algoritması genel adımları

Aykırı değer tespitinde kullandığımız modelde, bellek havuzu bir çözüm kümesini temsil etmektedir. İlk adımda, veri kümemizden belirli bir oranda alınan rastgele örneklerle bellek hücre havuzu ve ARB havuzu oluşturulur. İkinci adımda, popülasyonda her bir bellek hücresine ait uygunluk değerleri hesaplanır. Tüm hücreler uygunluk değerlerine göre sıralanarak antijenle en çok benzerliğe sahip antikor bulunur. Sonrasında, antijene en çok benzerlik gösteren antikorun mutasyona maruz bırakılması sonucu oluşan yeni antikor hücreleri ARB havuzuna eklenir. Böylece ARB havuzu üretilir ve bir sonraki adıma geçilir. Üçüncü adımda, ARB havuzunda var olan antikorlar ile mutasyonla oluşturulmuş yeni nesil antikorlar antijene maruz bırakılır. Uyarılma değerine göre ARB havuzundaki hücrelerden kaynak atamaları yapılır ve yeterince kaynağa sahip olamayan antikorlar ARB havuzundan çıkartılır. Yani uyarılma değeri belirli bir eşik değerinin üzerinde olan antikor mutasyona uğratılarak daha fazla kaynak üretilmesi sağlanmış olur. Bu işlemler ARB havuzunun ilgili antijen tarafından belirli bir eşik değerinin üzerinde uyarılması yapılanaya kadar devam etmektedir. Dördüncü adımda, en yüksek uyarılma değerine sahip olan ARB, aday bellek hücresi olarak atanır. Aday bellek

hücresinin uyarılma değeri, ikinci adımda belirlenmiş en iyi uyarılma değerine sahip antikoron uyuşma derecesi ile kıyaslanır. Bu kıyaslama sonucunda, aday bellek hücresinin uyarılma değeri en iyi uyarılma değerine sahip bellek hücresinin uyuşma derecesinden yüksek ise, aday bellek hücresi bellek havuzuna alınır. Aksi takdirde ikinci adımdaki uyarılma değeri yüksek olan antikoron, bellek havuzunda yerleşik olarak kalır ve aday bellek hücresi diskalifiye edilerek sistemden atılır. Bu işlemler veri kümesindeki her bir satır için ayrı ayrı uygulanır ve bellek hücre havuzunun tamamlanması gerçekleşir [39]. Son adımda ise, test edilecek verilerin tümü bellek hücresine sunulur. Her bir hücre için bellek hücresindeki ortalama uyarılma değeri hesaplanır. Ortalama uyarılma değeri en yüksek olan örnekten en az olan örneğe doğru bir sıralama yapılır. Sıralanmış bu örnekler analiz edilerek algoritma tarafında tespit edilen aykırı veriler çıkartılmış olur.

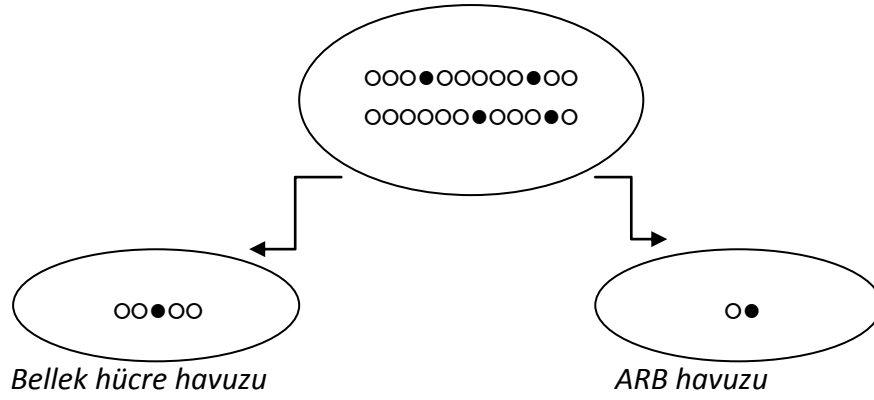
#### **3.4.2.2 Yapay Bağışıklık Tabanlı Sistem Algoritması ile Aykırı Değer Tespiti**

Bu bölümde, Yapay Bağışıklık Sistem Tabanlı Algoritması kullanılarak geliştirilen ve aykırı değerlerin tespitinde kullanılan modelin adım adım detaylı açıklamasına yer verilecektir. İlk olarak tespit modeli, Şekil 3. 8 ile Şekil 3. 17 arasındaki şekiller üzerinde görsel olarak canlandırılmış, sonrasında modelimizin algoritmik adımları detayları ile verilerek sistemin adım adım nasıl çalıştığı anlatılmıştır.

##### **3.4.2.2.1 YBA ile Aykırı Değer Tespitinin Şekiller ile Açıklanması**

Algoritmanın tüm adımları temsili olarak bu bölümde açıklanacaktır. Tüm şekillerde, veri kümemizde yer alan “o” ile gösterilen veriler normal verilerimizi, “●” ile gösterilen veriler ise aykırı verilerimizi temsil etmektedir. Bu verilerinin normal ya da aykırı olmasının eğitim sürecinde hiçbir önemi yoktur, sadece gerçeğe uygun bir soyutlama yapılmak istenmiştir.

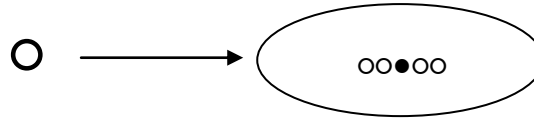
• **Adım 1: Başlangıç popülasyonunu oluşturma**



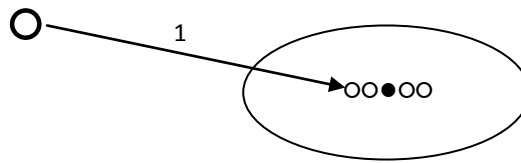
Şekil 3. 8 Bellek hücre havuzunun ve ARB havuzunun rastgele oluşturulması

Veri kümemizden rastgele alınmış örneklerden bellek hücre havuzu ve ARB havuzu oluşturulur. Geri kalan verilerimiz ise antijen olarak bellek hücre havuzuna tek tek sunulur (Şekil 3. 8).

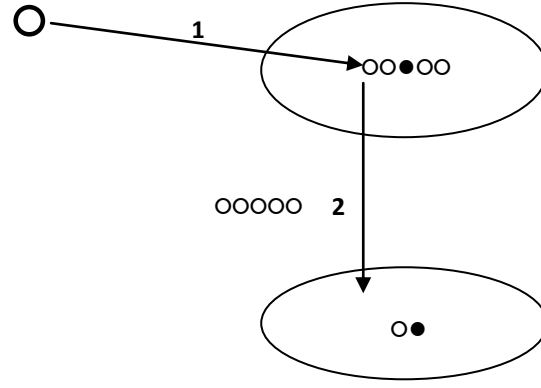
• **Adım 2: Bellek hücresinin ve ARB havuzunun üretilmesi**



Şekil 3. 9 Antijenin bellek hücre havuzuna sunulması



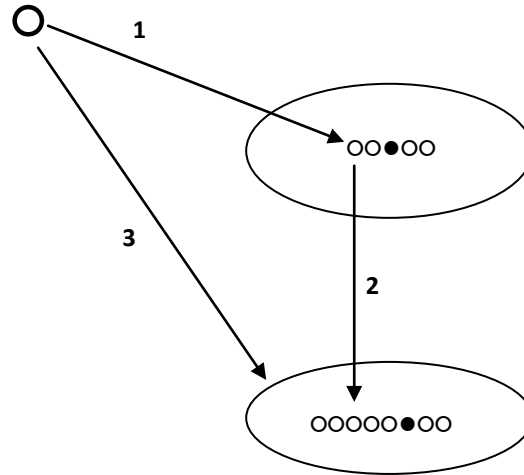
Şekil 3. 10 En çok uyarılmış bellek hücresinin belirlenmesi



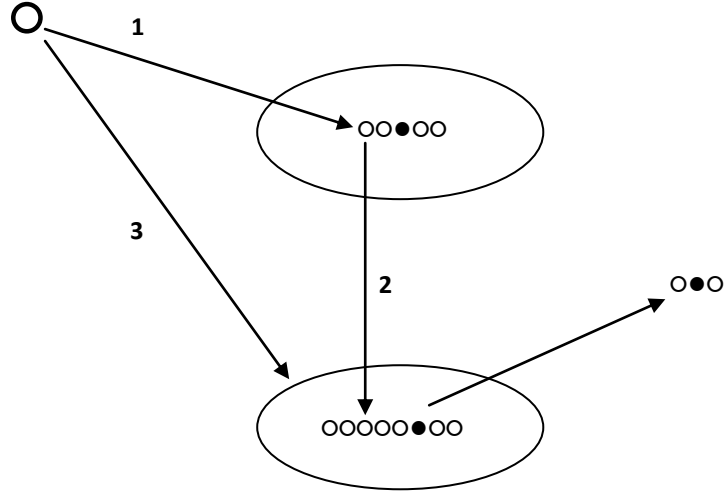
Şekil 3. 11 En çok uyarılmış bellek hücresinin mutasyonla çoğaltılarak ARB havuzuna eklenmesi

Bu bölümde, veri kümesinden alınan bir örnek antijenin bellek hücre havuzuna sunulması, bu antijenle en çok eşleşen bellek hücresinin bulunması ve bu hücrenin mutasyonla çoğaltılarak ARB havuzuna eklenmesi temsili olarak gösterilmiştir (Şekil 3. 9 - Şekil 3. 11).

• **Adım 3: ARB havuzunda aday bellek hücresinin geliştirilmesi**



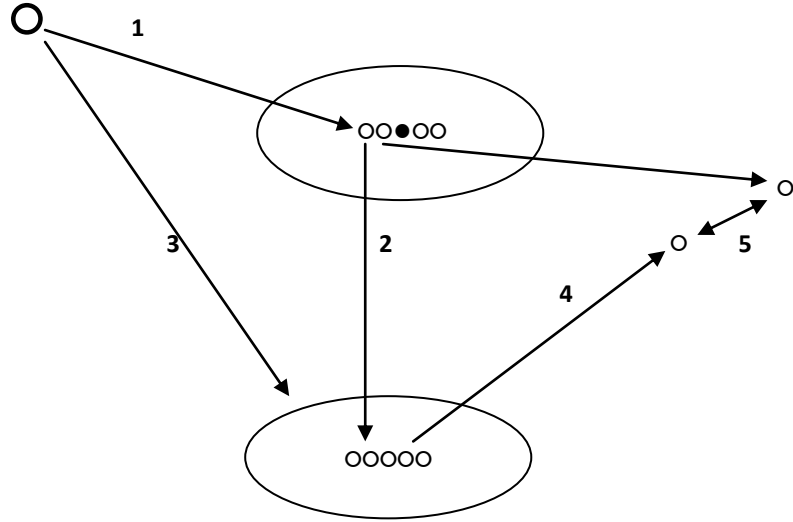
Şekil 3. 12 Antijenin ARB havuzuna sunulması



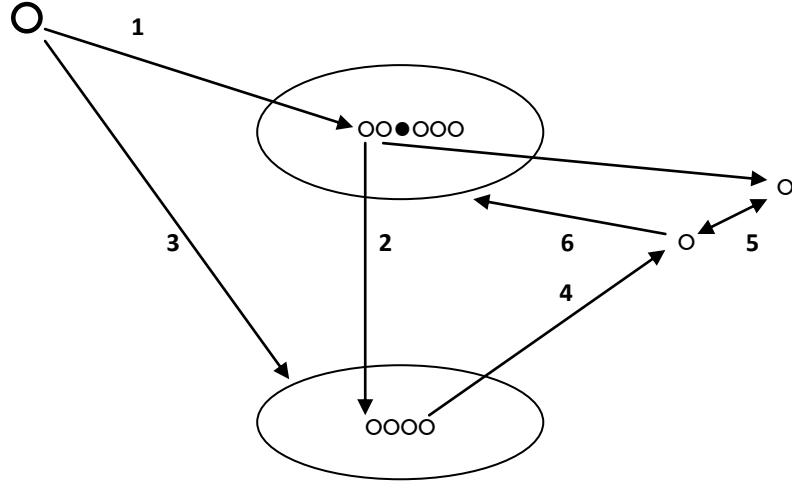
Şekil 3. 13 En az uyarılmış hücrelerin ARB havuzundan çıkartılması

Bu bölümde, antijen bu kez ARB havuzuna sunulur ve ARB havuzundaki tüm hücrelerle arasındaki uyarılma değeri hesaplanır (Şekil 3. 12). Uyarılma değeri düşük olan hücreler havuzdan çıkartılır (Şekil 3. 13). Çıkartılacak hücre sayısı, ARB havuzunda belirlenen kaynak sayısı ile ilgilidir. Bu sayının üzerine çıkıldığında, uyarılma değeri en küçük hücrelerden başlanmak üzere havuz dışına atılır.

• **Adım 4: Bellek hücresinin oluşturulması ve sisteme tanıtılması**



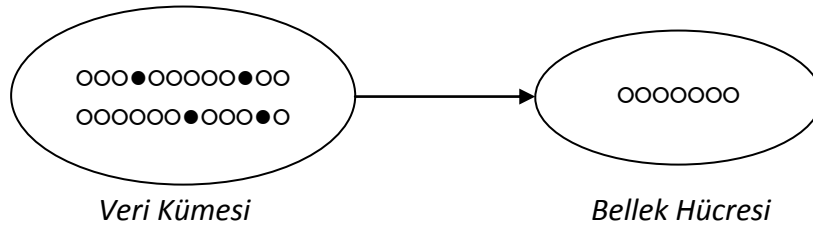
Şekil 3. 14 Aday bellek hücresinin bulunması ve en iyi uyuşan bellek hücresi ile karşılaştırılması



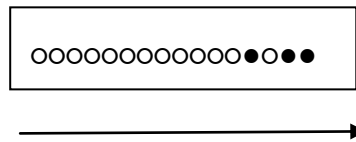
Şekil 3. 15 Aday bellek hücresinin bellek havuzuna alınması

Bu bölümde ise, antijen ile en iyi uyuşan aday bellek hücresi belirlenir ve ikinci adımda bulunan en iyi uyuşan hücre ile karşılaştırılır (Şekil 3. 14). Aday bellek hücresinin uyarılma değeri en iyi uyuşan hücrenin uyarılma değerinden büyükse, aday bellek hücresi bellek hücre havuzuna alınır ve en iyi uyuşan hücre ise belirli kontroller yapılarak bellek hücre havuzunda kalır ya da buradan atılır (Şekil 3. 15).

• **Adım 5: Aykırı değer tespitinin yapılması**



Şekil 3. 16 Veri kümesinin bellek hücresine sunulması



Şekil 3. 17 En çok uyarılan hücreden en az uyarılan hücreye göre sıralama yapılması

Son bölümde, veri kümemizdeki her bir örnek tek tek bellek hücresine sunulur (Şekil 3. 16). En çok uyarılan örnekten en az uyarılan örneğe doğru bir sıralama yapılır (Şekil 3. 17). Sıralanan verilerin analiz edilmesiyle algoritma tarafından tespit edilen aykırı örnekler bulunur.

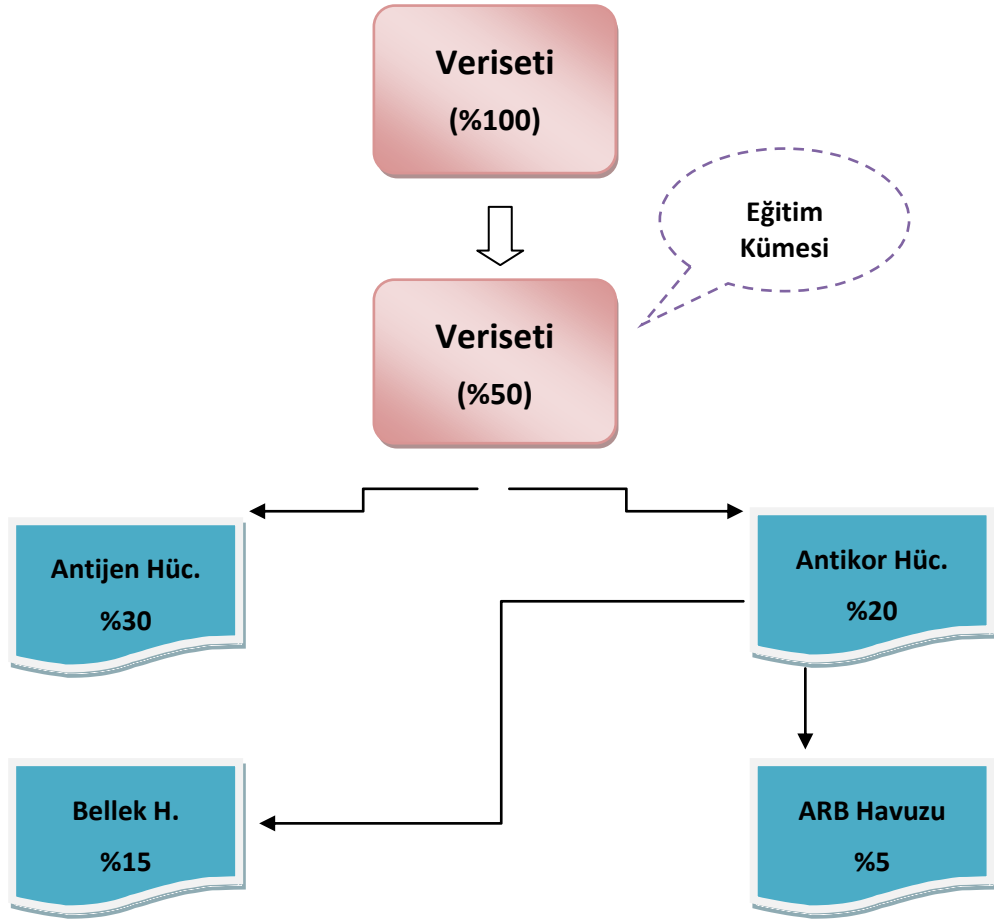


### 3.4.2.2.2 YBA ile Aykırı Değer Tespit Modeli

Bu bölümde, sistemin nasıl modellendiği ile ilgili akış şemasına yer verilerek Yapay Bağışıklık Sistem algoritmasının aykırı değer tespit etmedeki adımları anlatılacaktır.

#### • *Başlangıç popülasyonunu oluşturma*

Çalışmada kullanılan veri kümelerinin çok sayıda örnek içermesinden dolayı ve sistemin daha hızlı yanıt verebilmesini sağlamak açısından test aşamasında kullanılacak olan veri kümesi %50 oranında azaltılmıştır. Veri indirgenmesinde herhangi bir metot kullanılmamış ve veri kümesinden rastgele örnek seçimi yapılarak eğitim verisi oluşturulmuştur.



Şekil 3. 18 İlk adımın şematik görünümü

Oluşturulan eğitim verisinin bir bölümü antijen olarak sınıflandırılan ve sisteme sunulan örnekleri içerirken, diğer bir bölümü ise bu antijenleri tanıyacak ve savaşçı antikorların üretimini sağlayacak örneklerden oluşmaktadır. Antijen hücrelerini tanıyacak olan

antikor hücrelerinden rastgele alınan örnekler ile de Bellek Hücre Havuzu ve ARB havuzu oluşturulur. Şekil 3. 18’de antijen, antikor, Bellek Hücre Havuzu ve ARB havuzunun yüzde kaç oranında örnekten oluştukları gösterilmektedir.

Başlangıç adımında veriler Bellek Hücre Havuzuna sunulmadan önce bu veriler ile ilgili iki tür işlem yapılır. Bunlardan ilki normalizasyon işlemidir [39]. Veri kümesindeki tüm veriler normalize edilir. Normalize edilmiş bu verilerin her bir özelliği [0, 1] arasında olur. Diğer işlem ise test kümesindeki tüm veriler için bir Afinite (Benzeşme) Eşiği’nin hesaplanmasıdır (3.2) [39]. Bu eşik değeri, test kümesindeki tüm veriler arasındaki Öklid uzaklığının ortalamasından oluşur.

$$AE = \frac{\sum_{i=1}^n \sum_{j=i+1}^n \text{Öklid}(Agi, Agj)}{\frac{n(n-1)}{2}} \quad (3.2)$$

Öklid fonksiyonu burada iki antijen arasındaki Öklid uzaklığını verir. Agi ve Agj, antijen kümesindeki i. ve j. antijenleridir. n ise veri kümesindeki tüm örneklerin sayısını yani antijenleri gösterir.

Son işlem olarak da uyarılma eşiği hesaplanır. Uyarılma Eşiği, bir sayısından Afinite Eşiği’nin çıkartılması ile bulunur (3.3) [39].

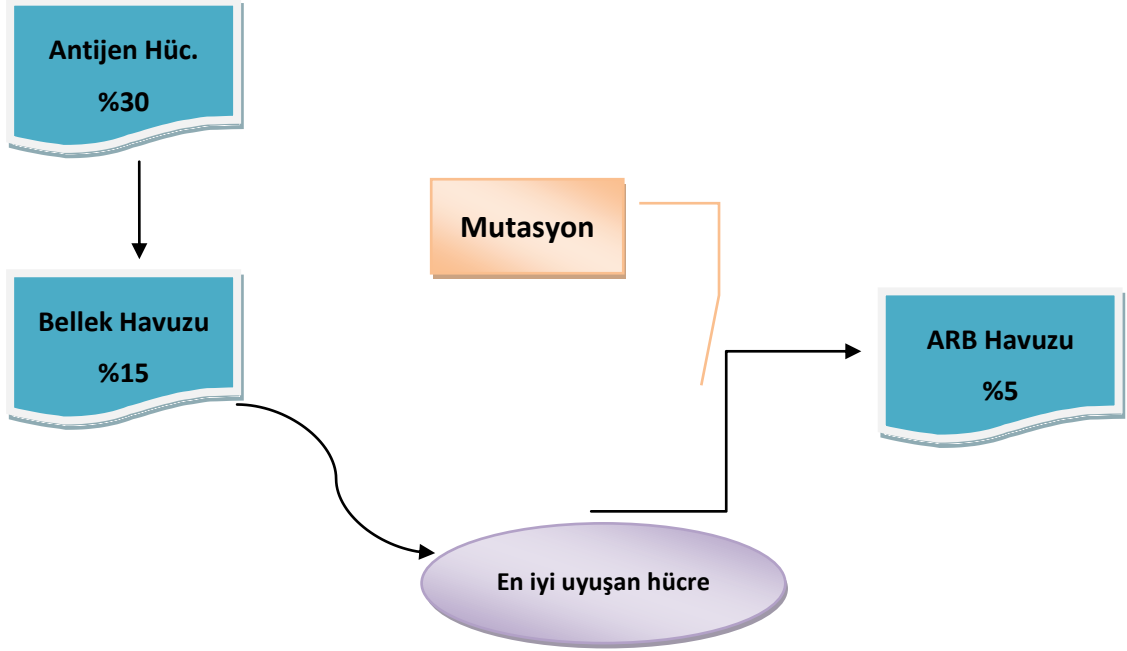
$$UE = 1 - AE \quad (3.3)$$

#### • *En iyi uyuşan hücrenin belirlenmesi ve ARB üretimi*

Antijen hücre havuzundan Bellek Hücre Havuzuna bir antijen sunulur. Bellek Hücre Havuzu’ndaki her hücrenin bu antijenle olan benzerliği hesaplanır. Bu benzerlik, antijen ile bellek hücresi arasındaki Öklid uzaklığı ile hesaplanır. Bu iki hücre arasındaki Öklid uzaklığının [0, 1] aralığında olması için, bu iki veri arasındaki Öklid uzaklık değeri  $\sqrt{n}$  ile bölünür [39] ve böylece normalize edilmiş uzaklık değeri bulunmuş olur. Buradaki n sayısı verimizin özellik sayısını gösterir.

Her bir bellek hücresinin antijenle olan bir uyarılma değeri vardır. Bu değer, 1 sayısından Öklid uzaklık değerinin çıkartılması ile bulunur. Bu değer 0 – 1 değer aralığında yer alır. Uyarılma 1’e ne kadar yakınsa bu etkileşimin ve benzerliğin o kadar yüksek olduğu anlamına gelir. Uyarılma değeri tüm bellek hücreleri için hesaplandıktan

sonra, uyarılma değeri büyük olan bellek hücresinden uyarılma değeri küçük olana doğru bir sıralama yapılır. En çok uyarılan bellek hücresi en iyi uyuşan bellek hücresi olarak seçilir.



Şekil 3. 19 ikinci adımı özetleyen şematik görünüm

En iyi uyuşan bellek hücresinin uyarılma değerin 1'e eşit olduğu durumlarda bire bir eşleşme söz konusu olacağından dolayı bu hücreden kopyalama ile ARB havuzuna ekleme yapılmasına gerek kalmaz. Çünkü maksimum uyarılma o anki antijen için eğitim verisini en iyi tanıyan bellek hücresi olarak belirlenmiş olur [39]. Şekil 3. 19'da uyarılma ve ARB havuzuna eklenme gösterilmiştir.

En iyi uyuşan hücreden oluşturulacak kopya sayısı, o hücrenin uyarılma değeri ile orantılıdır. Bir hücre için üretilecek toplam Kopya Sayısı,

$$KS = \text{Uyarılma değeri} * \text{Mutasyon oranı} * \text{Hipermutasyon oranı} \quad (3.4)$$

(3.4) gibi hesaplanır [39]. Çalışmada mutasyon oranı 20, hipermutasyon oranı ise 2 olarak alınmış ve parametre olarak sisteme girilmiştir.

Oluşturulan klonlar mutasyona tabi tutulur. Uyarılma değeri 0.9 ve üzerinde olan hücreler için oluşturulan klonların sadece bir özelliğinde, uyarılma değeri 0.5 ile 0.9 (0.5 dahil) değerleri arasında olan klonların iki özelliğinde, uyarılma değeri 0 ile 0.5 (0 dahil) değerleri arasında olan klonların ise üç özelliğinde [0, 1] arasında rastgele bir değer

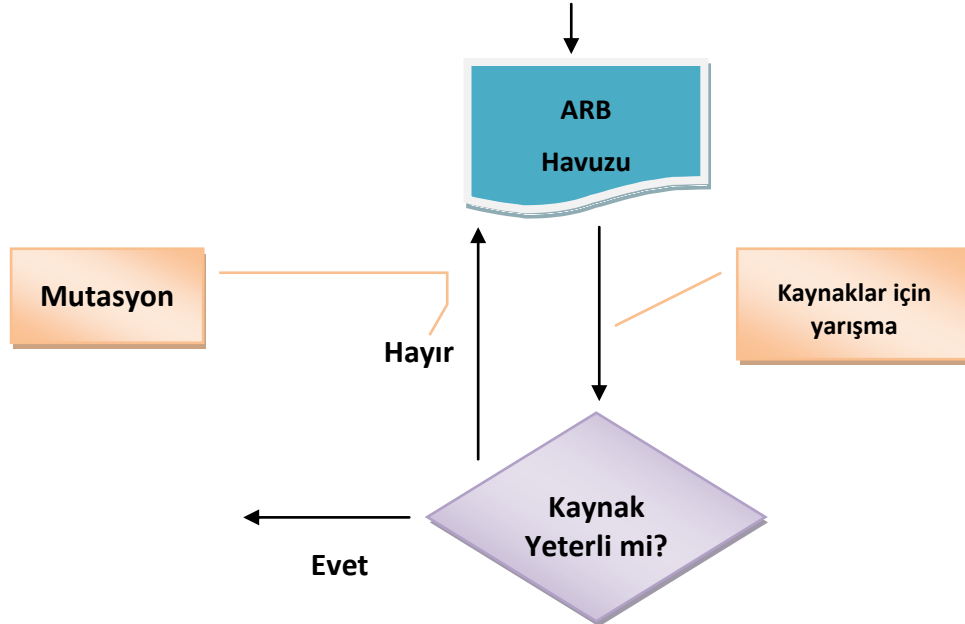
oluşturularak mutasyon gerçekleştirilir. Tüm klonların mutasyona uğrama işlemi bittikten sonra oluşan yeni nesiller ve en iyi uyuşan bellek hücresi ARB havuzuna eklenir.

- **ARB havuzunda aday bellek hücresinin geliştirilmesi**

ARB havuzundaki her hücre eğitim aşamasında o anki antijene sunulur ve her hücrenin uyarılma seviyesi belirlenir. Uyarılma seviyesine göre her hücre için Kaynak Ataması yapılır (3.5) [39]. Uyarılma seviyesi yüksek olan hücreden daha fazla sayıda kaynak atanır.

$$KA = \text{Uyarılma değeri} * 10 \quad (3.5)$$

Çalışmada ARB havuzunda bulunabilecek maksimum kaynak sayısı, veri kümesindeki örnek sayısının %50'si kadar olduğundan, bu sayının geçilmesi halinde en az uyarılan ARB'den başlanmak üzere kaynaklar silinir. Kaynak atamaları yapıldıktan sonra durma kriteri sağlanıyorsa süreç tamamlanır, durma kriteri sağlanmıyorsa kaynaklardan mutasyona uğramış klonlar oluşturulur.



Şekil 3. 20 Üçüncü adımı özetleyen şematik görünüm

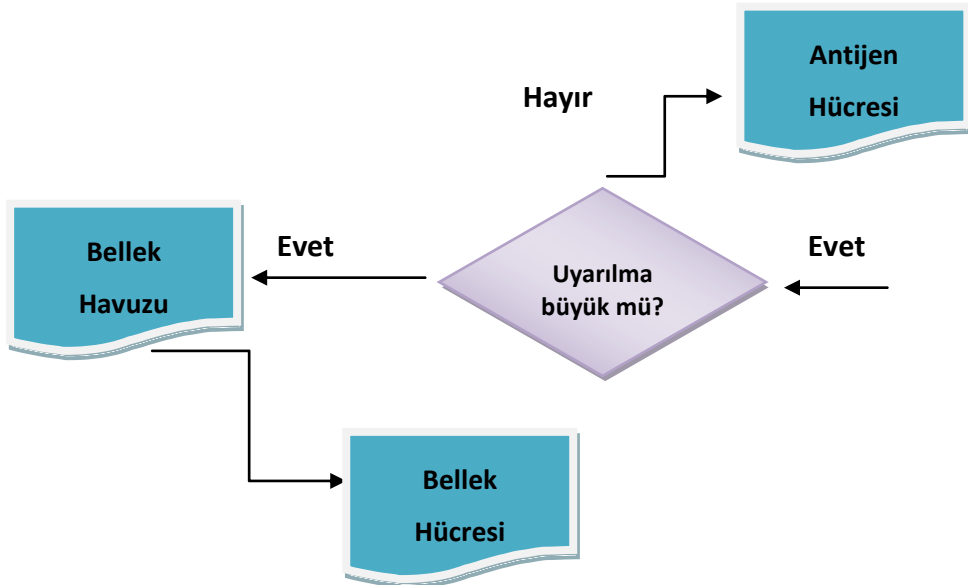
Durma kriteri, ARB hücrelerinin antijen tarafından uyarılma seviyelerinin ortalaması alınarak belirlenir. Bulunan ortalama uyarılma değeri, Uyarılma Eşiği ile karşılaştırılır. İlk adımda hesaplanan Uyarılma Eşik değeri 0.85 değerinin altında ise parametre değeri

0.85 olarak alınır, aksi taktirde ilk adımda hesaplanan Uyarılma Eşik değeri doğrudan parametre değeri olarak alınmış olur. Durma kriterinin sağlanması için ortalama uyarılma değerinin bu eşik değerinden yüksek olması gerekir. Yüksek olamaması durumunda kalan ARB havuzundaki hücrelerden mutasyona uğramış yeni klonlar oluşturulur. Oluşturulan bu yeni klonlar ARB havuzuna eklenir ve kaynaklar için yeniden yarışma gerçekleştirilir. Şekil 3. 20’de ARB havuzunda gerçekleştirilen bu işlemler gösterilmektedir.

Durma kriteri sağlandığında, en fazla kaynağa sahip en çok uyarılmış ARB, aday bellek hücresi olarak atanır. Böylece aday bellek hücresi bu bölümde belirlenmiş olup, bir sonraki adımda en iyi uyuşan bellek hücresi ile birlikte hangisinin antijen tarafından daha çok uyarılıp uyarılmadığının kontrolü yapılır. Yapılan kontroller sonucunda Bellek Hücre Havuzuna yeni bir hücrenin alınıp alınmamasına karar verilir [39].

#### • Bellek hücresinin oluşturulması

Eğitim aşamasının son adımı olan bu bölümde, bir önceki bölümde belirlenen aday bellek hücresinin uyarılma değeri ile ikinci adımda belirlenen en iyi uyuşan bellek hücresinin uyarılma değeri karşılaştırılır. Şekil 3. 21’de aday bellek hücresi ile en iyi uyuşan bellek hücresi arasındaki karşılaştırma gösterilmiştir.



Şekil 3. 21 Dördüncü adımı özetleyen şematik görünüm

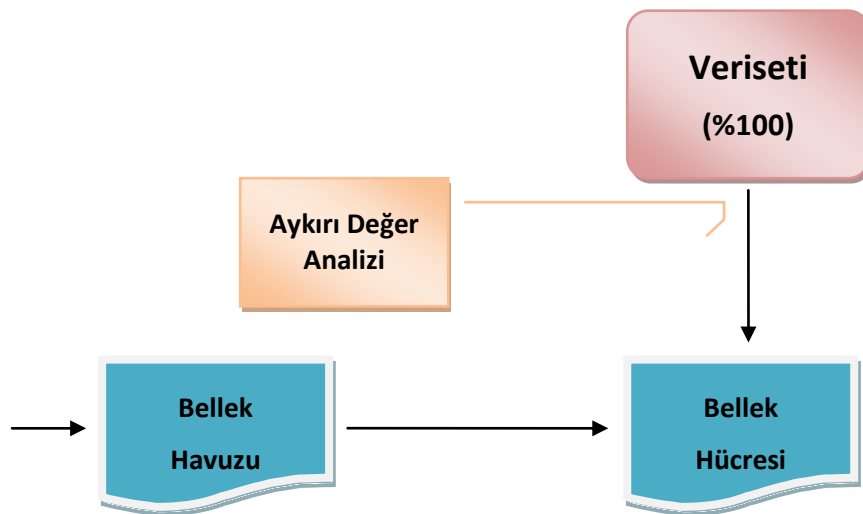
Aday Bellek Hücresi'nin uyarılma değeri en iyi uyuşan bellek hücresinin uyarılma değerinden büyük ise, aday bellek hücresi bellek havuzuna atanır. En iyi uyuşan bellek hücresinin bellek havuzundan çıkarılıp çıkarılmaması durumu ise yapılan kontrol ile belirlenir [39]. Aday Bellek Hücresi ile en iyi uyuşan bellek hücresi arasındaki Afinite Değeri bulunur. Bulunan bu değer hesaplanan kontrol değerinden (3.6) büyük ise en iyi uyuşan bellek hücresi bellek havuzunda kalır. Aksi durumda hücre, bellek havuzundan çıkartılır.

$$KD = \text{Afinite Eşığı} * 0,2 \quad (3.6)$$

Aday bellek hücresinin, bellek havuzuna eklenme durumunun değerlendirilmesi ile eğitim tamamlanmış olur. Tüm bu işlemler (2. , 3. ve 4. adımlar) veri kümesindeki her bir antijen için uygulanır. Veri kümesindeki tüm antijenlerin eğitiminin tamamlanması ile bellek hücresi oluşur ve böylece son adım olan test aşamasına geçilir.

#### • **Aykırı değerlerin belirlenmesi**

Veri kümesindeki tüm örnekler antijen olarak kabul edilip, bu antijenler eğitim aşaması sonunda oluşturulan bellek hücrelerine sunulur (Şekil 3. 22). Her bir antijenin bellek hücresindeki her bir hücre ile arasındaki uyarılma değeri hesaplanır. Bu uyarılma değerlerinin ortalaması alınır ve antijene yeni bir uyarılma değeri atanır. Bu işlemler veri kümesindeki tüm antijenler için yapıldıktan sonra, uyarılma değeri yüksek olan antijenden uyarılma değeri düşük olan antijene doğru bir sıralama yapılır.

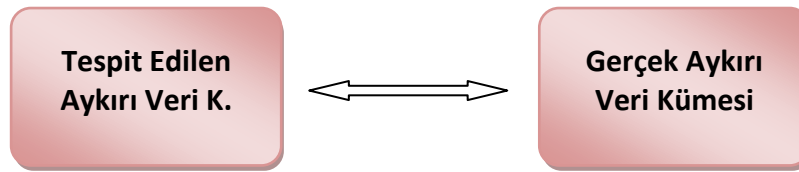


Şekil 3. 22 Son adımı özetleyen şematik görünüm

Algoritma tarafından uyarılma değerlerine göre sıralanmış antijenler kullanılarak iki tür aykırı tespit analizi yapılır. İlk analizde, veri kümesinde yer alan aykırı değerlerin sayısı kadar en düşük uyarılma değerine sahip antijenlerden başlanarak antijen seçilir. Bu antijenler YBA tarafından tespit edilen aykırı değerler olarak alınır. Tespit edilen aykırı değerler ile veri kümesindeki gerçek aykırı değerler karşılaştırılarak doğruluk analizi yapılır (Şekil 3. 23). İkinci ve son analizde ise, antijenlerin uyarılma değerlerinin ortalaması ve standart sapması alınır. Ortalama uyarılma değeri ve standart sapma değeri toplamının, veri kümesindeki toplam örnek sayısı ile çarpılması sonucu bir Eşik Değer İndeksi (EDİ) (3.7) bulunur.

$$\text{Eşik Değer İndeksi} = (\text{Ortalama değer} + \text{Standart Sapma}) * \text{Örnek Sayısı} \quad (3.7)$$

Eşik değerin altında olan örnekler normal veri olarak etiketlenirken, üzerinde olan örnekler ise aykırı veri olarak algoritma tarafından etiketlenmiş olunur.



Şekil 3. 23 YBA ile tespit edilen aykırı değerlerin doğruluk analizi

#### 3.4.2.2.3 Tipik YBA ile Aykırı Değer Tespit Modeli YBA Arasındaki Farklar

Aykırı değer tespiti üzerine yapılan modellemede Yapay Bağışıklık Sistem Tabanlı Algoritmasının genel adımları uygulanmıştır. Ancak, Yapay Bağışıklık Sistem Tabanlı Algoritmasının kullanıldığı çalışmalardaki bazı standart parametreler ve adımlar bu çalışmadaki performans testleri sonucu değiştirilmiş ya da bunlar üzerine ilave işlemler yapılmıştır. Bu işlemleri adım adım aşağıdaki gibi sıralayabiliriz:

- **Birinci adımda:** Çalışmada kullanılan veri kümelerinin çok sayıda örnek içermesinden dolayı eğitim veri kümesi %50 oranında azaltılmıştır. Bu aynı zamanda sistemin daha hızlı yanıt vermesi açısından avantaj sağlamıştır. Ayrıca eğitim veri kümesinden %30 oranında eğitici test verisi, %15 oranında bellek hücre havuzu ve %5 oranında da ARB havuzu oluşturulmuştur. Bu adımdaki son

bir farklılık ise, verilerin tümünün bellek hücre havuzunun ve ARB havuzunun oluşturulmadan önce normalize edilmiş olmasıdır.

- **İkinci adımda:** Klonlardaki çeşitliliğin ve verimin artırılması için üretilecek toplam kopya sayısında mutasyon oranı 20, hipermutasyon oranı ise 2 alınmıştır. Klonların mutasyona uğrama işleminde ise, tek bir özellikte mutasyon yapılmamış, uyarıma göre bir, iki ya da üç özellikte mutasyon işlemi gerçekleştirilmiştir.
- **Üçüncü adımda:** ARB havuzunda bulunabilecek maksimum kaynak sayısı, veri kümesindeki örnek sayısının %50'sini oluşturmaktadır. Uyarılma Eşiği parametresi ise yapılacak kontroller ile en az 0.85 değerinde tutulmuş ve kaynaklar için daha az yarışma yaptırılmıştır, çünkü veriler arasındaki sapmalar anormal sapma olarak görüldüğü için bu değer aşağı çekilmiştir.
- **Dördüncü adımda:** Bu adımda klasik Yapay Bağışıklık Sistem Tabanlı Algoritmasının adımı uygulanmıştır. Bu nedenle çok küçük değişimler dışında herhangi bir farklılık oluşturulmamıştır.
- **Beşinci ve son adımda:** En fazla farklılığın görüldüğü adımdır. Bu adım tamamen aykırı değer tespit etmede çözüm aşaması olduğu için diğer çalışmalardan tamamen ayrılmıştır. Bu adımda, veri kümesindeki tüm örnekler bellek hücresine sunulmuş ve her bir örnek için ortalama uyarılma değeri hesaplanmıştır. Bu uyarılma değeri yüksek olan örnekten düşük olan örneğe doğru bir sıralama yapılmıştır. Bu sıralanmış örnek dağılımı kullanılarak iki farklı aykırı tespit analizi yapılmıştır. Bu analizlerdeki sonuçlar, algoritmanın aykırı değer tespit etmedeki başarısını vermiş ve diğer metotlarla karşılaştırılmıştır.



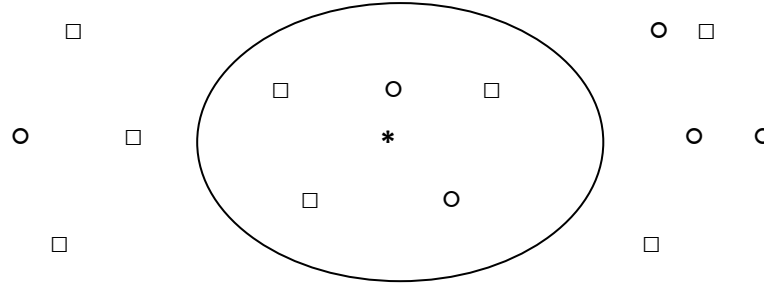
### SINIFLANDIRMA YÖNTEMLERİ ve DEĞERLENDİRME ÖLÇÜTÜ

Bu bölümde, çalışmada kullanılan sınıflandırma yöntemleri ve değerlendirme ölçütü anlatılacaktır.

#### 4.1 K – En Yakın Komşuluk Algoritması

K – En Yakın Komşuluk algoritması (KNN), veriler arasındaki uzaklıkları kullanarak sınıflandırma yapan uzaklık tabanlı bir yöntemdir. Algoritmada geçen  $k$  değeri, test kümesindeki bir örneğin sınıfını belirlemek için eğitim kümesindeki herhangi bir veriye en yakın  $k$  adet örneğin seçilmesi anlamına gelir. KNN, “birbirine yakın olan örnekler muhtemelen aynı kategoriye aittir” mantığı üzerine kurulu olan bir yöntemdir. KNN, veri madenciliği, bilgi güvenliğinin sağlanmasında, saldırı tespit sistemlerinde, genetik ve biyoenformatiğin birçok alanında, örüntü tanıma sistemleri gibi birçok benzeri sistemde kullanılmaktadır [42].

K – En Yakın Komşuluk algoritmasında herhangi bir örneğin sınıflandırılması, sınıf etiketi bilinen örnekler kullanılarak yapılmaktadır. Sınıflandırılacak örnek, eğitim kümesindeki her bir örnek ile ayrı ayrı işleme tabi tutulur. Sınıflandırılacak olan veri için eğitim kümesindeki o örneğe en yakın  $k$  adet örnek seçilir. Küme içerisinde hangi sınıfa ait en çok örnek varsa sınıflandırılması yapılacak örnek de o sınıfa ait olur.



Şekil 4. 1 KNN algoritmasından bir örnek

Şekil 4. 1’de “ \* ” test verisinin hangi sınıfa ait olduğuna karar vermek için  $k$  değeri 5 olan bir örnek gösterilmiştir. “ \* ” simgesine en yakın olan 5 örnek veri, test edilecek olan örneğin “ □ ” simgesine mi yoksa “ ○ ” simgesine mi ait olduğunu belirleyecektir. “\*” simgesine en yakın beş örnek işaretlenen daire ile belirlenmiştir. Bu daire içerisindeki beş örnekten üç tanesi “ □ ” simge sınıfına ait olduğu, geriye kalan iki tanesinin ise “ ○ ” simge sınıfına ait olduğu görülmektedir. Bu sonuçlara göre “ \* ” örneğinin “ □ ” simge sınıfından olduğu tahmin edilerek, hangi sınıfa ait olduğu belirlenir.

K – En Yakın Komşuluk algoritması tarafından sınıflandırılması yapılacak verilerin eğitim kümesindeki verilerle olan benzerlikleri hesaplanır. Test edilen verilerin atamaları, en yakın olduğu düşünülen  $k$  adet verinin ortalamasıyla belirlenen eşik değere göre yapılır [42].

KNN algoritmasının adımları:

1. Test Kümesindeki her bir verinin  $X = \{ X_1, X_2, X_3, \dots, X_n \}$ , eğitim kümesindeki verilere  $D = \{ d_1, d_2, d_3, \dots, d_m \}$  yakınlığı hesaplanır.
2. Her bir verinin eğitim kümesindeki verilere olan yakınlıkları sıralanıp, ilk  $k$  tane yakınlık değeri alınarak ortalama yakınlık değeri hesaplanır. Çalışmada  $k$  değeri 10 olarak alınmıştır.
3. Ortalama yakınlık değeri büyük olan veriden yakınlık değeri küçük olan veriye doğru bir sıralama yapılır.
4. Ortalama yakınlıkları belirlenen test verilerinin ortalama değeri ve standart sapması alınarak eşik değer indeksi (Bkz. Bölüm 3) hesaplanır.

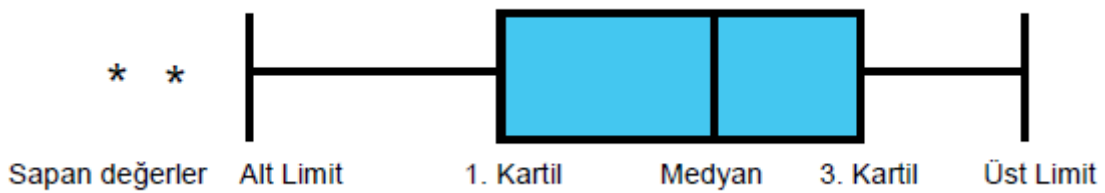
5. Eşik değeri indeksinin altında olan örnekler normal, üstünde olan örnekler ise aykırı olarak etiketlenir.

Algoritmanın performansını etkileyen faktörler [42]:

1. Benzerlik ölçümü için seçilecek  $k$  en yakın komşu sayısı.
2. Verinin aykırı veri ya da normal veri olup olmadığına karar vermek için, verinin  $k$  en yakın komşuya olan benzerliklerinden hesaplanan ve ortalama değerinin kıyaslanmasında kullanılan eşik değeri.
3. Benzerlik ölçümü
4. Eğitim kümesindeki normal verilerin yeterli sayıda olması. Yeterli sayıda ve çeşitlilikte olmayan normal veriler, test kümesinde yer alan normal verileri anormal olarak algılayabilmesi.

#### 4.2 Kutu Grafik Yöntemi

Kutu Grafik yöntemi, bir vektör uzayında bir değişkenin dağılımını göstermek için kullanılır [43]. Veri kümesindeki her bir özellik için minimum (en küçük gözlem değeri), maksimum (en büyük gözlem değeri), medyan (ortanca değeri),  $Q_1$  (birinci dörtebirlik),  $Q_3$  (üçüncü dörtebirlik) değerleri hesaplanır. Bu hesaplamaların yapılabilmesi için veri kümesindeki ilgili özellik değerleri küçükten büyüğe doğru sıralanır. Bu sıralamadan beş sayılı özetleme tablosu çıkartılır (4. 3).



Şekil 4. 2 Kutu Grafik Yöntemi

Şekil 4. 2’de merkez kutu  $Q_1$  ve  $Q_3$  arasında, kutunun ortasında medyan ve sapan değerler ise grafiğin dışında gösterilmiştir. Kutunun sol ve sağ sınırları, sırasıyla  $Q_1$  (%25’lik değeri) ve  $Q_3$ ’ü (%75’lik değeri) temsil eder. Kutunun içindeki çizgi (medyan), %50’lik değerde çizilir. Kutunun sol ve sağ tarafına uzanan çizgiler sınır çizgisi olarak

anılır. Sağ taraftaki sınırın sonunda azami değer işaretlenmiştir. Asgari değer, sol taraftaki sınırın sonunda yer almaktadır. Sol ve sağ sınır dışındaki tüm değerler ise aykırı değer olarak kabul edilir.

Veri kümelerindeki her bir örneğin tüm özellikleri için ayrı ayrı grafik metodu uygulanır. Bir örneğin bir özellik değerinde aykırı olup olmadığına, veri kümesindeki ilgili özelliğin dağılımına bakılarak karar verilir. Veri kümesinden alınan bir örneğin kaç özelliğinde aykırılık var ya da yok, her bir verinin özellik değerlerinin ayrı ayrı değerlendirilmesi ile bulunur. Tüm verilerin özelliklerindeki aykırılık tespiti yapıldıktan sonra, özelliklerinde hiç aykırılık görülmeyen ya da en az aykırılık görülen örnekten, özelliklerinde en çok aykırılık görülen örneğe doğru bir sıralama gerçekleştirilir. Sıralanmış bu örneklerden Kutu Grafik yöntemi tarafından tespit edilen aykırı örnekler çıkartılır ve yöntemin performansı değerlendirilir.

#### **4.3 Uzaklık Tabanlı Aykırı Değer Tespit Algoritması**

Uzaklık Tabanlı Aykırı Değer Tespit Algoritması, veriler arasındaki uzaklığı kullanan bir diğer algoritmadır. Algoritmanın adımları aşağıdaki gibidir:

- Veri kümesindeki tüm veriler normalize edilir. Bu verilerin her bir özelliği [ 0, 1 ] değer aralığında olur.
- İki veri arasındaki uzaklık mesafesi  $\sqrt{n}$  (n değeri bir örneğin özellik sayısını göstermektedir) ile bölünür. Böylece iki veri arasındaki benzerlik ölçüm değeri [ 0 , 1 ] aralığında tutulmuş olur.
- Tüm veriler arasındaki benzerlik ölçüm değerleri hesaplanır. İki veri arasındaki benzerlik ölçütü için Eşik Değer (d) kullanılır. Bu değer çalışmada 0.3 alınmıştır.
- Her bir örnek için veri kümesindeki diğer verilerle olan aykırılık ölçüt sayısı (p) bulunur. Bu sayı, bir verinin veri kümesindeki tüm verilerle olan benzerlik ilişki sayısını göstermektedir. Yani iki örnek arasındaki benzerlik 0.3 değerinin altında ise, bu örnekler birbirleri ile benzerdir. Aksi durumda bu iki örnek arasında aykırılığın olduğu kabul edilir.
- Bir örneğin diğer verilerle olan ilişkide, aykırılık ilişkisi fazla ise o örneğin aykırılığı kuvvetlidir. Bu mantıkla, aykırılık ilişki sayısı az olan örnekten fazla olan örneğe

doğru bir sıralama yapılır. Sıralanmış bu örneklerden DBO tarafından tespit edilen aykırı örnekler çıkartılır ve yöntemin performansı değerlendirilir.

Çizelge 4. 1 Benzerlik ölçüm tablosu

|   | A | B   | C   | d   | e   | f   |
|---|---|-----|-----|-----|-----|-----|
| a | 0 | 0,3 | 0,1 | 0,7 | 0,1 | 0,2 |
| b |   | 0   | 0,2 | 0,1 | 0,2 | 0,4 |
| c |   |     | 0   | 0,3 | 0,5 | 0,1 |
| d |   |     |     | 0   | 0,8 | 0,2 |
| e |   |     |     |     | 0   | 0,3 |
| f |   |     |     |     |     | 0   |

Çizelge 4. 1’de örnek bir veri kümesindeki a, b, c, d, e, f verileri arasındaki benzerlik ölçüm değerleri gösterilmiştir. Her bir örneğin diğer verilerle olan benzerlik ilişkisini [0, 1] aralığında görülmektedir. Örneğin, a verisi ile d verisi arasındaki benzerlik ilişkisi 0.3 değerinin üzerinde olması, bu iki veri arasındaki ilişkinin aykırılık olduğunu göstermektedir. a ile e arasındaki benzerlik ilişkisinin 0.3 değerinin altında olması ise, bu iki veri arasındaki ilişkide benzerlik olduğunu göstermektedir. a verisinin diğer verilerle olan aykırılık ilişki sayısı 2’dir ( $p=2$ ). Çünkü a verisi, b ve d verileri ile aykırılık göstermektedir. Çizelge 4. 2’de aykırılık ilişki sayılarını gösterilmektedir.

Çizelge 4. 2 Aykırılık ilişki sayısı

|          |   |
|----------|---|
| <b>a</b> | 2 |
| <b>b</b> | 2 |
| <b>c</b> | 2 |
| <b>d</b> | 3 |
| <b>e</b> | 3 |
| <b>f</b> | 2 |

Çizelge 4. 2’de görüldüğü gibi, aykırılık ilişki sayısı fazla olan d ve e örneklerinin aykırı olmaları muhtemeldir. Bu örnek veri kümesi içerisinde 2 adet aykırı örnek varsa bu örnekler d ve e verileridir diyebiliriz.

#### 4.4 Değerlendirme Ölçütü: Hata Matrisi

Veri kümelerine uygulanan metotların performanslarını ölçmek için hata kestirim modeli olan hata matrisleri oluşturulmuştur. Matrislerin elemanları kullanılarak F-Ölçüm, Recall (çağrı), Accuracy (doğruluk), Precision (kesinlik) ve True Negative Ratio-TNR değerleri hesaplanmıştır. Çağrı, doğru şekilde sınıflandırılan aykırı verilerin yüzdesidir. Doğruluk, verilerin doğru şekilde kestiriminin yapılma olasılığıdır. TNR, normal verilerin doğru şekilde tespit edilme oranıdır. F-ölçüm ise, kesinlik ve çağrı değerlerinin harmonik ortalaması ile hesaplanmaktadır [39]. Çalışmada yaptığımız analizlerde yöntemlerin performans değerlendirilmesi yapılırken her metodun F-Ölçüm değeri kriter olarak alınmıştır.

Çizelge 4. 3 Hata Matrisi [39]

|                            | <b>Aykırı<br/>(Tespit)</b> | <b>Normal<br/>(Tespit)</b> |
|----------------------------|----------------------------|----------------------------|
| <b>Aykırı<br/>(Gerçek)</b> | Doğru Pozitif<br>(TP)      | Yanlış Pozitif<br>(FP)     |
| <b>Normal<br/>(Gerçek)</b> | Yanlış Negatif (FN)        | Doğru Negatif (TN)         |

$$Recall = TP / ( TP + FN )$$

$$Accuracy = ( TP + TN ) / ( TP + FN + FP + TN )$$

$$Precision = TP / ( TP + FP )$$

$$TNR = TN / ( TN + FP )$$

Çizelge 4. 3'te örnek bir hata matrisi tablosu verilmektedir. Aykırı değerlikli örnekler aykırı ile gösterilirken, aykırı olmayan normal veriler ise normal ile ifade edilmiştir. Sütunlarda tespitler, satırlarda ise gerçek değerler verilmiştir. Matrisin diyagonalindeki değerler doğru yapılan tespitleri gösterirken, diğer değerler hatalı tespit sayılarını göstermektedir.

### DENEYSEL SONUÇLAR

Çalışmada üç farklı veri kümesi kullanılmış, YBA algoritmasının ve diğer yöntemlerin başarısı yapılan iki tür analizle ölçülmüştür.

#### 5.1 Veri Kümeleri

İkisi gerçek diğeri ise yapay olmak üzere üç veri kümesi üzerinde çalışma yapılmıştır. Yapay Bağışıklık Sistem Tabanlı Algoritması (YBA) ile Distance-Based Outlier (DBO), K-Nearest Neighbor (KNN) ve Box Plot metotları; bu veri kümeleri üzerinde ayrı ayrı uygulanmış ve aykırı değer tespit etmedeki başarıları hesaplanarak birbirleri ile karşılaştırılmıştır.

##### 5.1.1 Pima Veri Kümesi

Pima, birçok araştırmada ön işlemden geçirildikten sonra aykırı değerlerin çıkartılması ve özellikle verilerin sınıflandırılması ile ilgili çalışmalarda test edilmiş bir veri kümesidir. Vijayarani ve Nithya (2011), kümeleme algoritmalarını kullanarak Pima Indians Diabetes ve Breast Cancer Wisconsin gibi sağlık alanındaki veri kümelerinde aykırı değerleri tespit etmiş, önerdiği yaklaşımla diğer yöntemlerin aykırı değer tespit etme performanslarını karşılaştırmıştır [44]. Pearson, R. K. (2006), veri kümelerinde sıkça karşılaşılan eksik veriler üzerine yaptığı çalışmasında ön işlemden geçirilmiş Pima veri kümesini kullanmıştır [45]. Breault (2001), diyabetik veri kümelerinden olan Pima veri kümesine Rosetta Yazılımını uygulamış, veri kümesinde diyabetik olmayan analiz değerlerinin tespit edilmesi ile ilgili çalışmalar yapmıştır [46]. Xue, Shang ve Feng

(2010), C-means kümeleme algoritmasını Pima veri kümesine uygulamış, önerdiği metodun aykırı değer tespitindeki performansını analiz etmiştir [47].

Ana kaynağı Ulusal Diyabet Enstitüsü ve UCI olan Pima veri kümesi, 768 örnekten oluşmaktadır. Her bir örnek 8 özellik içermektedir. 21 yaş ve üzeri hasta bayanlarda gözlemlenen değerlerden oluşan bu veri kümesinin içerdiği özellik vektörüne ait bilgiler aşağıdaki gibidir:

- |                                              |                                                                |
|----------------------------------------------|----------------------------------------------------------------|
| <b>1. Kaç kez hamilelik geçirdikleri</b>     | <b>5. İki saatlik serum insülin (mu U/ml)</b>                  |
| <b>2. Glukoz konsantrasyonu</b>              | <b>6. Vücut kitle indeksi (Ağırlık-kg/(Boy-m)<sup>2</sup>)</b> |
| <b>3. Kan basıncı (mm Hg)</b>                | <b>7. Diyabet soyağacı fonksiyonu</b>                          |
| <b>4. Üç başlı kasın deri kalınlığı (mm)</b> | <b>8. Yaş</b>                                                  |

Veri kümesindeki verilerin 500 tanesi normal veri, geri kalan 268 örnek ise aykırı verileri temsil etmektedir. Aykırı değerler, veri kümesinin yaklaşık %34 'lük bir kısmını oluşturmaktadır. Her bir örneğin bir veya daha fazla özelliğinde, o özellik içerisinde anormal şekilde sapmış aykırı değerler vardır. Şekil 5. 1'de Pima veri kümesinden bir örnek gösterilmektedir.

Preview of C:\Users\MasK\Desktop\TEZ\Veriseti\Veriseti\pima.txt

|      |      |      |       |     |        |       |     |
|------|------|------|-------|-----|--------|-------|-----|
| 101  | 148  | 72   | 35    | 0   | 33,60  | 21,60 | 50  |
| 1    | 85   | 66   | 29    | 0   | 26,60  | 0,351 | 31  |
| 8    | 183  | 521  | 0     | 0   | 105,01 | 0,67  | 32  |
| 1    | 89   | 66   | 23    | 94  | 28,10  | 0,167 | 21  |
| 0    | 137  | 210  | 1025  | 168 | 43,10  | 2,29  | 33  |
| 5    | 116  | 74   | 0     | 0   | 25,60  | 0,201 | 30  |
| 1805 | 5    | 50   | 32    | 88  | 31,00  | 7,80  | 26  |
| 10   | 115  | 0    | 0     | 0   | 35,30  | 0,134 | 29  |
| 2    | 197  | 70   | 98    | 543 | 30,50  | 0,16  | 164 |
| 8    | 125  | 96   | 0     | 0   | 0,00   | 0,23  | 54  |
| 4    | 110  | 92   | 0     | 0   | 37,60  | 0,191 | 30  |
| 10   | 168  | 74   | 72    | 0   | 38,00  | 12,30 | 34  |
| 10   | 139  | 80   | 0     | 0   | 27,10  | 1     | 57  |
| 1    | 0    | 121  | 23    | 846 | 30,10  | 0,40  | 71  |
| 5    | 166  | 72   | 40785 | 175 | 25,80  | 16,00 | 51  |
| 91   | 100  | 0    | 0     | 0   | 257,70 | 0,48  | 32  |
| 0    | 1071 | 84   | 47    | 230 | 45,80  | 0,55  | 31  |
| 7    | 107  | 2050 | 0     | 0   | 29,60  | 0,25  | 175 |
| 1    | 103  | 30   | 38    | 83  | 43,30  | 0,183 | 33  |
| 31   | 115  | 70   | 350   | 96  | 34,60  | 0,53  | 32  |
| 3    | 126  | 88   | 41    | 235 | 39,30  | 0,704 | 27  |
| 8    | 99   | 84   | 0     | 0   | 35,40  | 0,388 | 50  |
| 7    | 196  | 90   | 0     | 0   | 120,78 | 0,45  | 41  |
| 9    | 119  | 212  | 101   | 0   | 29,00  | 0,26  | 29  |
| 0    | 143  | 94   | 33    | 146 | 36,60  | 11,00 | 51  |
| 10   | 0    | 70   | 26    | 115 | 181,00 | 0,21  | 41  |
| 87   | 147  | 76   | 0     | 0   | 39,40  | 0,26  | 43  |

Şekil 5. 1 Pima veri kümesi



Veri kümesinde, her bir örneğin en az bir en çok üç özelliğinde aykırı değerler yer almaktadır. Şekil 5. 2’de Pima veri kümesindeki aykırı verilerden bir örnek gösterilmektedir.

Preview of C:\Users\MasK\Desktop\TEZ\Veriseti\Veriseti\pima\_aynk.txt

|      |      |      |       |     |        |       |     |
|------|------|------|-------|-----|--------|-------|-----|
| 101  | 148  | 72   | 35    | 0   | 33,60  | 21,60 | 50  |
| 8    | 183  | 521  | 0     | 0   | 105,01 | 0,67  | 32  |
| 0    | 137  | 210  | 1025  | 168 | 43,10  | 2,29  | 33  |
| 1805 | 5    | 50   | 32    | 88  | 31,00  | 7,8   | 26  |
| 2    | 197  | 70   | 98    | 543 | 30,50  | 0,16  | 164 |
| 8    | 125  | 96   | 0     | 0   | 0,00   | 0,23  | 54  |
| 10   | 168  | 74   | 72    | 0   | 38,00  | 12,30 | 34  |
| 1    | 0    | 121  | 23    | 846 | 30,10  | 0,4   | 71  |
| 5    | 166  | 72   | 40785 | 175 | 25,80  | 16,00 | 51  |
| 91   | 100  | 0    | 0     | 0   | 257,70 | 0,48  | 32  |
| 0    | 1071 | 84   | 47    | 230 | 45,80  | 0,55  | 31  |
| 7    | 107  | 2050 | 0     | 0   | 29,60  | 0,25  | 175 |
| 31   | 115  | 70   | 350   | 96  | 34,60  | 1     | 32  |
| 7    | 196  | 90   | 0     | 0   | 120,78 | 0,45  | 41  |
| 9    | 119  | 212  | 101   | 0   | 29,00  | 0,26  | 29  |
| 0    | 143  | 94   | 33    | 146 | 36,60  | 11,00 | 51  |
| 10   | 0    | 70   | 26    | 115 | 181,00 | 0,21  | 41  |
| 87   | 147  | 76   | 0     | 0   | 39,40  | 0,26  | 43  |
| 3    | 158  | 76   | 171   | 245 | 31,60  | 0,85  | 328 |

Şekil 5. 2 Pima aykırı değer veri kümesi

### 5.1.2 Blood Veri Kümesi

Blood, sınıflandırma ve aykırı tespit çalışmalarında sıkça kullanılmış ve önışlemden geçirilerek uygulamalarda test edilmiş bir diğer veri kümesidir. Singh ve Nagpal (2011), Blood veri kümesinde anormallik gösteren aykırı davranışların bulunması ve karar ağaçlarının oluşturulması gibi çalışmalar yaparak veri kümesi üzerinde bilgi keşfi yapmıştır [48]. Santhanam ve Sundaram (2010), CART (Classification and Regression Trees) sınıflandırma algoritmasını Blood veri kümesine uygulamış, kan bağış davranışlarını dönemsel olarak tanımlamıştır [49]. Li, Cuesta-Albertos ve Liu (2010), Depth - Versus - Depth (DD) sınıflandırma tekniğini Blood veri kümesine uygulayarak, sınıflandırılma işlemini gerçekleştirmiştir [50].

Ana kaynağı Chung-Hua Üniversitesi ve UCI olan Blood veri kümesi, 748 örnekten oluşmaktadır. Her bir örnek 4 özellik içermektedir. Kan bağış hizmet servisi tarafından tutulan istatistiksel değerlerden oluşan bu veri kümesi, farklı dönemlerdeki kan bağış sayısı (ilk bağıştan sonraki dönem, son bağıştan sonraki dönem, v.s) ve toplam bağış sayısı gibi özellikleri içeren değerleri tutmaktadır.

Veri kümesindeki verilerin 570 tanesi normal veri, geri kalan 178 örnek ise aykırı verileri temsil etmektedir. Aykırı değerler, veri kümesinin yaklaşık %23 'lük bir kısmını

oluşturmaktadır. Pima veri kümesindeki gibi her bir örneğin bir veya daha fazla özelliğinde, o özellik içerisinde anormal şekilde sapmış aykırı değerler vardır. Şekil 5. 3'te Blood veri kümesinden bir örnek gösterilmektedir.

Preview of C:\Users\Mask\Desktop\TEZ!\Veriseti\Veriseti\Blood.txt

|     |     |       |     |
|-----|-----|-------|-----|
| 2   | 50  | 0     | 105 |
| 0   | 100 | 3250  | 28  |
| 104 | 16  | 4000  | 35  |
| 2   | 32  | 5000  | 150 |
| 1   | 24  | 6000  | 77  |
| 4   | 4   | 1000  | 4   |
| 2   | 7   | 0     | 14  |
| 1   | 12  | 3000  | 35  |
| 2   | 108 | 2250  | 22  |
| 31  | 46  | 11500 | 0   |
| 4   | 23  | 5750  | 58  |
| 0   | 3   | 750   | 4   |
| 2   | 55  | 40222 | 28  |
| 1   | 13  | 3250  | 47  |
| 2   | 6   | 1500  | 100 |
| 42  | 5   | 1250  | 11  |
| 2   | 45  | 3500  | 48  |
| 154 | 15  | 0     | 475 |
| 2   | 6   | 1500  | 15  |
| 2   | 48  | 750   | 400 |
| 43  | 3   | 10    | 450 |
| 4   | 11  | 2750  | 28  |
| 2   | 22  | 1500  | 16  |
| 145 | 6   | 0     | 16  |
| 9   | 9   | 2250  | 16  |
| 4   | 14  | 3500  | 40  |
| 4   | 6   | 1500  | 14  |
| 4   | 105 | 3000  | 34  |
| 4   | 5   | 1250  | 102 |

Şekil 5. 3 Blood veri kümesi

Veri kümesinde, her bir örneğin en az bir en çok iki özelliğinde aykırı değer yer almaktadır. Şekil 5. 4'te Blood veri kümesindeki aykırı verilerden bir örnek gösterilmektedir.

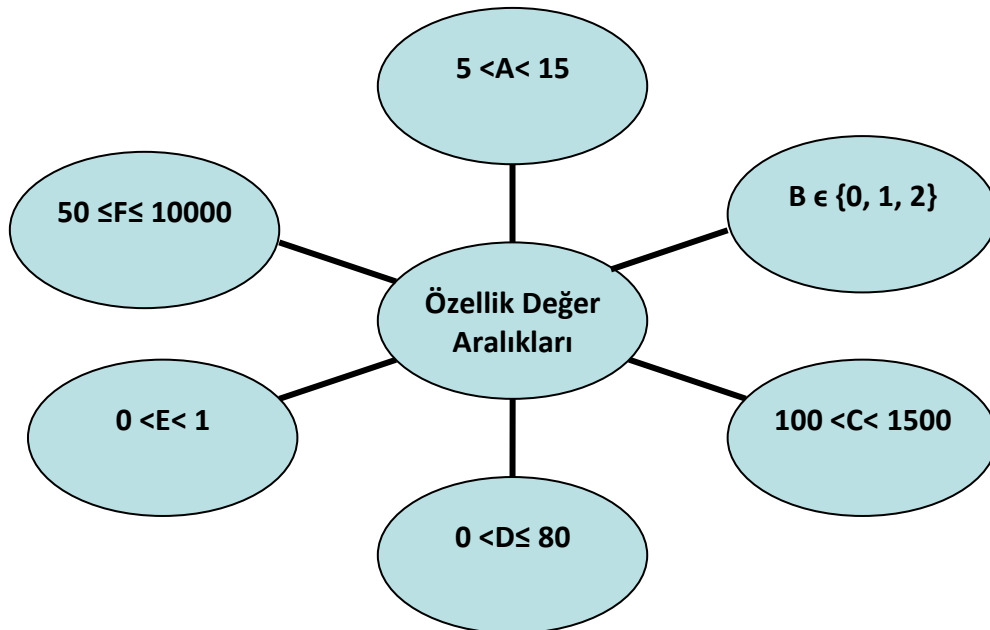
Preview of C:\Users\MasK\Desktop\TEZ!\Veriseti\Veriseti\Blood\_aynık.txt

|     |     |        |      |
|-----|-----|--------|------|
| 2   | 5   | 1250   | 47   |
| 652 | 45  | 2000   | 135  |
| 6   | 2   | 20     | 11   |
| 215 | 1   | 250    | 4    |
| 125 | 1   | 10     | 0    |
| 4   | 452 | 250    | 4    |
| 4   | 1   | 0      | 425  |
| 525 | 1   | 250    | 165  |
| 4   | 1   | 250    | 4    |
| 755 | 10  | 2000   | 52   |
| 4   | 3   | 750    | 25   |
| 565 | 17  | 4250   | 79   |
| 265 | 8   | 106200 | 125  |
| 11  | 3   | 750    | 16   |
| 0   | 14  | 3500   | 325  |
| 311 | 3   | 750    | 38   |
| 4   | 4   | 1000   | 655  |
| 125 | 4   | 1000   | 22   |
| 9   | 45  | 500    | 16   |
| 795 | 5   | 1250   | 28   |
| 14  | 3   | 750    | 250  |
| 985 | 25  | 1000   | 23   |
| 11  | 5   | 1250   | 165  |
| 11  | 9   | 0      | 72   |
| 109 | 5   | 1250   | 1050 |

Şekil 5. 4 Blood aykırı değer veri kümesi

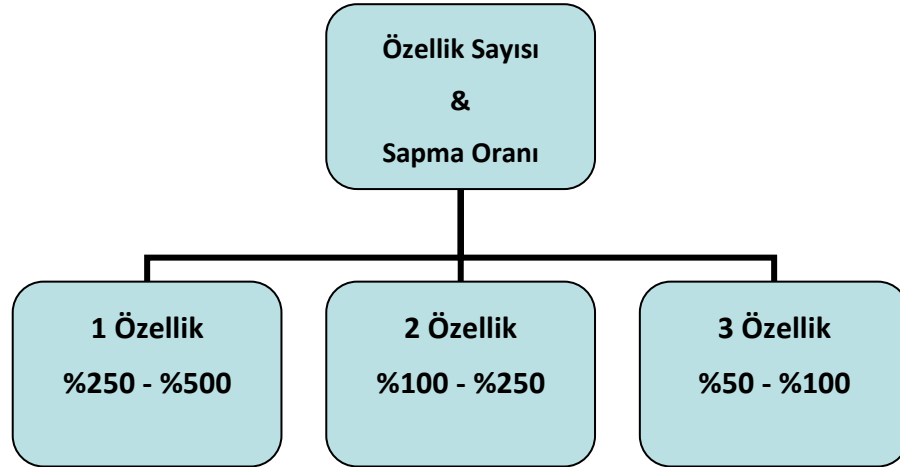
### 5.1.3 Yapay Veri Kümesi

Yapay veri kümesi, rastgele değerlerle oluşturulmuş bir veri kümesidir. Bu veri kümesi 1000 örnekten oluşmaktadır. Her bir örnek 6 özellik içermektedir. Rastgele oluşturulan her bir özellik değeri belirli aralıklarda tutulmuş, bu aralıklar özelliklere göre değişkenlik göstermektedir. Şekil 5. 5'te özellik verilerinin değer aralıkları gösterilmektedir.



Şekil 5. 5 Yapay veri kümesi özelliklerinin değer aralıkları

Bir örnekte, A, B, C, D, E, F özellikleri verilen değer aralıklarında yer almışsa normal veri olarak kabul edilmiştir. Aksi durumda bir örneğin en az bir özelliğinde özellik değer aralıklarından belirli bir oranda saptırılmış değer var ise, o örnek aykırı olarak tanımlanmıştır. Her bir örnek için belirlenen bu sapma oranı, aykırı değer içeren özellik sayısına göre değişmektedir. Şekil 5. 6'da özellik sayılarına göre sapma oranı verilmiştir.



Şekil 5. 6 Yapay veri kümesi özelliklerinin sapma oranları

Özellik sayısındaki artışa bağlı olarak sapma oranının azalmasının nedenini aşağıdaki gibi sıralayabiliriz:

- Tek bir özellikte görülen sapma ne kadar çok yüksek ise ayırt ediciliği o kadar kolay olacaktır. Çünkü çok boyutlu verilerin tek bir özelliğinde görülen aykırılığın ayırt edilmesi güçtür. Bu yüzden tek bir özellikte oluşturulacak aykırı örnek verilerinin sapma oranı yüksek olmalıdır.
- İki veya üç özellikte görülen sapma oranı, tek bir özellikte görülen sapma oranı ile aynı ya da bu orandan daha yüksek olma durumunda değildir. Çünkü sapma görülen özellik sayısı arttıkça aykırılık kuvvetlenir, buna bağlı olarak da ayırt edicilik artmış olur.
- Üçten fazla özelliklerde pozitif ya da negatif saptırılma oranı yüksek de olsa düşük de olsa verinin ayırt edilebilme olasılığı çok yüksek olmaktadır. Bu durumda çok boyutlu verilerde üçten fazla özelliklerde görülen sapmaların çok fazla önemi olmadığını söyleyebiliriz.

Sapma oranları, yapay veri kümesindeki normal veri dağılımı içerisinde yer alan bir ya da birden fazla özelliğin maksimum ya da minimum değeri üzerinden uygulanır. Bu oranlar belirlenen değer aralıklarından rastgele seçilerek yapılır. Örneğin, bir verinin iki özelliğinde sapmış değer yer alacaksa, bu iki özellik için sapma oranı %100 - %250 değer arasında olur ve maksimum veya minimum değer üzerine uygulanır.

Preview of C:\Users\MasK\Desktop\TEZ\Veriseti\Veriseti\Yapay Veri Kümesi.txt

|    |   |      |     |       |       |
|----|---|------|-----|-------|-------|
| 7  | 0 | 120  | 45  | 0,4   | 9000  |
| 12 | 0 | 105  | 30  | 0,9   | 40500 |
| 6  | 2 | 1050 | 71  | 0,5   | 150   |
| 6  | 1 | 900  | 15  | 0,3   | 4750  |
| 11 | 0 | 3150 | 200 | 0,6   | 200   |
| 10 | 2 | 300  | 8   | 0,5   | 1075  |
| 6  | 2 | 1300 | 61  | 0,9   | 85    |
| 14 | 1 | 210  | 25  | 0,3   | 8100  |
| 24 | 1 | 550  | 55  | 1,85  | 17100 |
| 7  | 1 | 480  | 12  | 0,4   | 990   |
| 8  | 0 | 175  | 1   | 0,2   | 2350  |
| 13 | 2 | 1000 | 20  | 0,7   | 195   |
| 56 | 0 | 375  | 78  | 0,8   | 3375  |
| 9  | 0 | 255  | 32  | 0,8   | 75    |
| 13 | 2 | 790  | 14  | 0,3   | 7020  |
| 6  | 1 | 7700 | 42  | 0,8   | 135   |
| 14 | 0 | 865  | 64  | 0,3   | 640   |
| 26 | 2 | 1400 | 140 | 0,4   | 15300 |
| 11 | 2 | 790  | 17  | 0,4   | 1360  |
| 8  | 1 | 450  | 38  | 0,2   | 6650  |
| 8  | 0 | 2520 | 21  | 1,575 | 14400 |
| 6  | 1 | 365  | 26  | 0,6   | 4850  |
| 42 | 1 | 1145 | 8   | 0,2   | 27000 |
| 10 | 0 | 950  | 80  | 0,9   | 5550  |
| 13 | 2 | 620  | 19  | 0,5   | 620   |
| 7  | 2 | 1350 | 5   | 0,4   | 2750  |
| 10 | 0 | 580  | 400 | 0,1   | 4550  |

Şekil 5. 7 Yapay veri kümesi

Şekil 5. 7’de yapay veri kümesinden bir örnek gösterilmektedir. Veri kümesindeki verilerin 900 tanesi normal veri, geri kalan 100 örnek ise aykırı verileri temsil etmektedir. Aykırı değerler, veri kümesinin %10’luk bir kısmını oluşturmaktadır.

Preview of C:\Users\MasK\Desktop\TEZ\Veriseti\Veriseti\Yapay\_aynık.txt

|    |   |       |     |       |       |
|----|---|-------|-----|-------|-------|
| 12 | 0 | 105   | 30  | 0,9   | 40500 |
| 11 | 0 | 3150  | 200 | 0,6   | 200   |
| 14 | 1 | 210   | 25  | 4,95  | 8100  |
| 24 | 1 | 550   | 55  | 1,85  | 17100 |
| 56 | 0 | 375   | 78  | 0,8   | 3375  |
| 6  | 1 | 7700  | 42  | 0,8   | 135   |
| 26 | 2 | 1400  | 140 | 0,4   | 15300 |
| 8  | 1 | 450   | 38  | 0,2   | 6650  |
| 8  | 0 | 2520  | 21  | 1,575 | 14400 |
| 6  | 1 | 365   | 26  | 0,6   | 4850  |
| 42 | 1 | 1145  | 8   | 0,2   | 27000 |
| 13 | 2 | 50040 | 19  | 0,5   | 620   |

Şekil 5. 8 Yapay aykırı veri kümesi

Şekil 5. 8’deki yapay aykırı veri kümesi, 100 örneği temsil etmektedir. Bu örneklerden, tek özelliğinde aykırı değer gözlenen örnek sayısı %20, iki özelliğinde aykırı değer

gözlenen örnek sayısı %30 ve üç özelliğinde aykırı değer gözlenen örnek sayısı ise %50 oranında alınmıştır. Üç özelliğinde sapma gözlenen örnek sayısının fazla alınmasının nedeni, daha gerçekçi analizler yapılarak sapma oranı düşük olan verilerin daha net bir şekilde tespit edilmesini sağlamaktır.

## **5.2. Deneysel Analiz Sonuçları**

Bu bölümde veri kümelerine uygulanan Yapay Bağışıklık Sistem Tabanlı Algoritmasının ve diğer tekniklerin aykırı değer tespit etmedeki performansları iki farklı analiz türünde ayrı ayrı değerlendirilerek karşılaştırılmıştır.

### **5.2.1 Deneysel Analizler**

Veri kümelerine uygulanan yöntemlerden iki farklı analiz sonuçları çıkartılmıştır. Analizler, metotların tespit ettiği, verilerin ortalama uyarılma değerleri kullanılarak yapılmıştır. İlk analiz, bir veri kümesinde aykırı sayısının bilinmesi halinde uygulanması gereken analiz tipini gösterirken; ikinci ve diğer analiz ise, veri kümesindeki aykırı veri sayısının bilinmemesi halinde uygulanması gereken analizi göstermektedir.

#### **5.2.1.1 Aykırı Veri Sayısı Baz Alınarak Yapılan Analiz**

Veri kümelerine uygulanan her bir yöntem için verilerin ortalama uyarılma değeri hesaplanır. Ortalama uyarılma değeri en yüksek olan örneklerden en düşük olan örneklerle doğru bir sıralama yapılır. Veri kümesinde yer alan gerçek aykırı örneklerin sayısı kadar örnekler kümeden alınır ve en az ortalama uyarılma değerine sahip örnekler bu sıralamadan çekilir. Alınan örnekler daha önceden bilinen gerçek aykırı veriler ile karşılaştırılır. Tespit edilen aykırı örneklerin gerçekten aykırı olup olmadığına dair doğruluk analizi yapılır ve her bir yöntem için F-Ölçüm değeri hesaplanır.

#### **5.2.1.2 Eşik Değer İndeksi Baz Alınarak Yapılan Analiz**

Yapılan ilk analizde olduğu gibi, her bir yöntem için verilerin ortalama uyarılma değerleri hesaplanarak, en çok uyarılmış verilerden en az uyarılmış verilere doğru bir sıralama yapılır. Verilerin ortalama uyarılma değerleri kullanılarak, bu değerlerin standart sapması ve ortalaması hesaplanır. Standart sapma ve ortalama değerlerin toplamı alınır ve veri kümesindeki örnek sayısı ile çarpılır. Çarpım sonucunda eşik değer indeksi bulunmuş olur. Eşik değer indeksi kullanılarak ortalama uyarılma değerlerine

göre sıralanan verilerin kaçınıcı örnek indeksinden sonrasının aykırı veriler olduğunu çıkartılır. Yani eşik değeri indeksinin üzerindeki örnekler aykırı verileri temsil ederken, altında yer alan veriler ise normal davranış gösteren örnekleri temsil eder. Son işlem olarak da her bir yöntem için F-Ölçüm değeri hesaplanır ve yöntemlerin performansları karşılaştırılır.

### **5.2.2 Pima Veri Kümesi ile Aykırı Değer Analizi**

İlk deneysel küme olan Pima, en fazla özellik ve aykırı örnek içeren veri kümesidir. Bu küme içerisindeki 768 örneğin her biri antijen sıfatıyla bellek hücrelerine sunulmuştur. Her bir örneğin bellek hücresi ile olan ortalama uyarılma değeri hesaplanmış ve en çok uyarılan örnekten en az uyarılan örneğe doğru bir sıralama yapılmıştır. Uyarılmış örnek sıralamaları kullanılarak “Aykırı değeri sayısı” ve “Eşik değeri indeks sayısı” baz alınarak iki farklı türde analizler yapılmıştır.

#### **5.2.2.1 Aykırı Sayısı Baz Alınarak Yapılan Analiz**

Uyarılma seviyesi en düşük örnekten başlanarak gerçekte aykırı veri sayısı kadar en az uyarılmış örnekler tespit edilmiş, bu örnekler Pima gerçekte aykırı veri kümesi ile karşılaştırılmıştır. Karşılaştırma sonucunda, Yapay Bağışıklık Sistem algoritması ile tespit edilen aykırı örneklere ait analiz sonuçları Çizelge 5. 1’de, Çizelge 5. 2’de ve Şekil 5. 9’da gösterilmiştir.

Çizelge 5. 1’de, ilk sütunda en çok uyarılan örnekten en az uyarılan örneğe doğru yapılan sıralamaya yer verilmiştir. [501, 768] örnek aralığındaki sıralamalarda yer alan tüm veriler, YBA tarafından aykırı değeri olarak tespit edilen örnekleri göstermektedir. İkinci sütunda yer alan değerler, uyarılma sırası içerisindeki gerçekte aykırı örnek sayısını; üçüncü sütunda yer alan değerler, ikinci sütundaki aykırı örnek sayısının uyarılma sırası içerisindeki örnek sayısına oranını; dördüncü sütundaki değerler ise, ikinci sütundaki aykırı sayısının toplam aykırı sayısına oranını göstermektedir.

Çizelge 5. 2’de, hata matrisi tablosu gösterilmektedir. Bu tabloya bakılarak, YBA tarafından tespit edilen aykırı verilerin kaç tanesinin doğru, kaç tanesinin yanlış tespit edildiği sonucunu görebiliriz. Matrisin diyagonalindeki değerler doğru yapılan tespitleri gösterirken, diğer değerler ise yapılan tespit hatalarını gösteren veri sayılarını içermektedir.

Şekil 5. 9'da, Çizelge 5. 1'in birinci ve üçüncü sütununda yer alan örnek sayısı ve uyarılma seviyeleri içerisinde tespit edilmiş gerçek aykırı örnek sayı yüzdesi arasındaki ilişki, x-y koordinatı üzerinde gösterilmiştir.

Çizelge 5. 1 Uyarılma sırasına göre aykırı değer tespiti (YBA)

| Uyarım Örnek Sıralaması | Gerçek Aykırı Sayısı | Aykırı Oranı | Aykırı Yüzdesi |
|-------------------------|----------------------|--------------|----------------|
| 1-75                    | 0                    | 0            | 0              |
| 76-150                  | 0                    | 0            | 0              |
| 151-225                 | 0                    | 0            | 0              |
| 226-300                 | 1                    | 1,33         | 0,37           |
| 301-375                 | 3                    | 4            | 1,12           |
| 376-450                 | 13                   | 17,3         | 4,85           |
| 451-500                 | 15                   | 30           | 5,59           |
| <b>501-600</b>          | 79                   | 79           | 29,4           |
| <b>601-675</b>          | 66                   | 88           | 24,6           |
| <b>676-750</b>          | 73                   | 97,3         | 27,2           |
| <b>751-768</b>          | 18                   | <b>100</b>   | 6,71           |
| <b>Toplam</b>           | <b>268</b>           |              | <b>% 100</b>   |

YBA ile uyarılmış aykırı örnekler sıralamasında, tespit edilen en az uyarılmış 18 [751-768] örneğin hepsi doğru tespit edilmiştir. En az uyarılmış 168 [601-768] örneğin doğru tespit edilme oranı %93'tür.

Çizelge 5. 2 YBA için Hata Matrisi (Pima)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 236    | 32     |
|        | Normal | 32     | 468    |



YBA ile aykırı olarak tespit edilmiş verilerden 236'sının algoritma tarafından doğru tespit edildiği görülmüştür. 32 veri ise normal olmasına rağmen aykırı olarak etiketlenmiştir. Çizelge 5. 2'de yer alan değerler kullanılarak aşağıdaki hesaplamalar yapılmıştır:

- Doğru şekilde tespit edilmiş aykırı verilerin başarı oranı:

$$\text{Aykırı Değer Tespit Yüzdesi} = 236 / (236 + 32) = 0.88$$

- Doğru şekilde tespit edilmiş örneklerin başarı oranı:

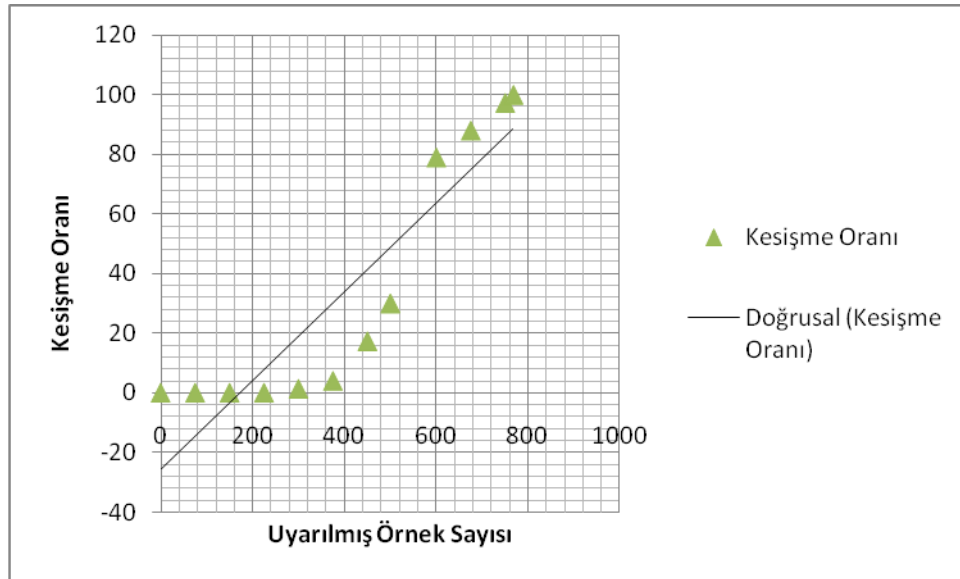
$$\text{Tüm Örnekler için Tespit Yüzdesi} = (236 + 468) / (236 + 32 + 468 + 32) = 0.916$$

- Doğru şekilde tespit edilmiş aykırı olmayan verilerin başarı oranı:

$$\text{Normal Veri Tespit Yüzdesi} = 468 / (468 + 32) = 0.936$$

- Hassasiyet (Precision) =  $236 / (236 + 32) = 0.88$

- F-ölçüm (F-measure) =  $(2 * (0.88 * 0.88)) / (0.88 + 0.88) = \mathbf{0.88}$



Şekil 5. 9 Uyarılmaya göre aykırı tespit başarı oranı

Şekil 5. 9'da, ez az uyarılmaya sahip örneklerin büyük bir çoğunluğunun doğru tespit edildiğini kesişme oranına bakarak söyleyebiliriz. Kesişme oranı yüzdelerinin 600-768 örnek aralığında yaklaşık %80 ve üzerinde olduğu, aynı zamanda örnek sayısındaki artışın 400'den sonraki dilimlerde yaklaşık doğrusal bir kesişmeye karşılık geldiği görülmektedir.

Çizelge 5. 3'te, Çizelge 5. 4'te ve Çizelge 5. 5'te, Pima veri kümesine sunulan sırasıyla, KNN, DBO ve BP tekniklerinin ayrı ayrı performansları gösterilmektedir. Çizelge 5. 6'da ise, YBA ile KNN, DBO ve BP tekniklerinin aykırı örnek tespit etme başarıları karşılaştırmalı olarak verilmiştir.

Çizelge 5. 3 KNN için Hata Matrisi (Pima)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 222    | 46     |
|        | Normal | 46     | 454    |

Çizelge 5. 4 DBO için Hata Matrisi (Pima)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 215    | 53     |
|        | Normal | 53     | 447    |

Çizelge 5. 5 BP için Hata Matrisi (Pima)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 225    | 43     |
|        | Normal | 43     | 457    |

Çizelge 5. 6 Pima veri kümesi için YBA performans gösterimi (Analiz 1)

| Metotlar | Bulunan Gerçek Aykırı Veri Sayısı | F - Ölçüm |
|----------|-----------------------------------|-----------|
| YBA      | 236                               | 0.88      |
| KNN      | 222                               | 0.82      |
| DBO      | 215                               | 0.80      |
| BP       | 225                               | 0.83      |

#### 5.2.2.2 Eşik Değer İndeks Sayısı Baz Alınarak Yapılan Analiz

Uyarılmış örneklerin ortalama değerleri ve standart sapması alınarak eşik değer indeksi hesaplanmıştır (Bkz. Bölüm 3). Bu değer her bir metot için ayrı ayrı hesaplanmış ve eşik değer üzerinde yer alan örnekler aykırı veri olarak tanımlanmıştır. Eşik değer indeksi; YBA için 569, KNN için 548, DBO için 571 ve BP için 539 olarak hesaplanmıştır.

Çizelge 5. 7 YBA için Hata Matrisi (Pima)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 199    | 69     |
|        | Normal | 0      | 500    |

Çizelge 5. 7’de uyarılmış örnek sıralamasında yer alan 569. örnek ve sonrasındaki en az uyarılmış örnekler alınarak YBA tarafından tespit edilen aykırı veriler gösterilmiştir.

$$\text{Recall} = 199 / (199 + 0) = 1.00$$

$$\text{Precision} = 199 / (199 + 69) = 0.74$$

$$\text{F-Ölçüm} = \mathbf{0.85}$$

YBA tarafından tespit edilen gerçek aykırı veri sayısı 199’dur. Gerçek aykırı veri olmasına rağmen algoritmanın normal veri olarak tanımladığı örnek sayısı ise 69’tur. Algoritmanın bu veri kümesi üzerinde aykırı verileri doğru tespit etme başarı oranı, F-Ölçüm değerine bakarak %85’tir diyebiliriz.

Çizelge 5. 8’de, Çizelge 5. 9’da ve Çizelge 5. 10’da KNN, DBO ve BP yöntemlerinin aykırı veri tespitindeki performansları gösterilmiştir. Çizelge 5. 11’de ise YBA ve diğer yöntemlerin Pima veri kümesinde aykırı örnekleri doğru tespit etmedeki başarı oranları gösterilmiştir.

Çizelge 5. 8 KNN için Hata Matrisi (Pima)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 211    | 57     |
|        | Normal | 9      | 491    |

$$\text{Recall} = 211/(211+9) = 0.95$$

$$\text{Precision} = 211/(211+57) = 0.78$$

$$\text{F-Ölçüm} = \mathbf{0.85}$$

Çizelge 5. 9 DBO için Hata Matrisi (Pima)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 195    | 73     |
|        | Normal | 2      | 498    |

$$\text{Recall} = 195/(195+2) = 0.98$$

$$\text{Precision} = 195/(195+73) = 0.72$$

$$\text{F-Ölçüm} = \mathbf{0.83}$$

Çizelge 5. 10 BP için Hata Matrisi (Pima)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
| GERÇEK |        | Aykırı | Normal |
|        | Aykırı | 211    | 57     |
|        | Normal | 18     | 482    |

$$\text{Recall} = 211/(211+18) = 0.92$$

$$\text{Precision} = 211/(211+57) = 0.78$$

$$\text{F-Ölçüm} = \mathbf{0.84}$$

Çizelge 5. 11 Pima veri kümesi için YBA performans gösterimi (Analiz 2)

| Metotlar | Bulunan Gerçek Aykırı Veri Sayısı | F - Ölçüm |
|----------|-----------------------------------|-----------|
| YBA      | 199                               | 0.85      |
| KNN      | 211                               | 0.85      |
| DBO      | 195                               | 0.83      |
| BP       | 211                               | 0.84      |

### 5.2.3 Blood Veri Kümesi ile Aykırı Değer Analizi

Diğer bir deneysel küme olan Blood, en az özellik içeren veri kümesidir. Bu küme içerisindeki 748 örnek üzerinden yapılan çalışmada Yapay Bağışıklık Sistem Tabanlı algoritmasının performansı analiz edilmiştir.

#### 5.2.3.1 Aykırı Sayısı Baz Alınarak Yapılan Analiz

Gerçek aykırı veri sayısı kadar en az uyarılmış örnekler alınarak yapılan, bu kümeye ait performans sonuçları Çizelge 5. 12’de, Çizelge 5. 13’te ve Şekil 5. 10’da gösterilmiştir.

Çizelge 5. 12 Uyarılma sırasına göre aykırı değer tespiti (YBA)

| Uyarım Örnek Sıralaması | Gerçek | Aykırı Sayısı | Aykırı Oranı | Aykırı Yüzdesi |
|-------------------------|--------|---------------|--------------|----------------|
| 1-75                    |        | 0             | 0            | 0              |
| 76-150                  |        | 0             | 0            | 0              |
| 151-225                 |        | 0             | 0            | 0              |
| 226-300                 |        | 0             | 0            | 0              |
| 301-375                 |        | 0             | 0            | 0              |
| 376-450                 |        | 2             | 2,66         | 1,12           |
| 451-500                 |        | 9             | 18           | 5,05           |
| 501-570                 |        | 13            | 18,5         | 7,30           |
| <b>571-650</b>          |        | 64            | 80           | 35,9           |
| <b>651-700</b>          |        | 44            | 88           | 24,7           |
| <b>701-748</b>          |        | 46            | <b>95,8</b>  | 25,8           |
| <b>Toplam</b>           |        | <b>178</b>    |              | <b>% 100</b>   |

Yapay Bağışıklık Sistem Tabanlı algoritması tarafından Blood veri kümesinde yer alan 178 [571-748] aykırı veriden 154 aykırı veri doğru tespit edilmiştir. Doğru tespit başarı oranı %86,5'tir.

Çizelge 5. 13 YBA için Hata Matrisi (Blood)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 154    | 24     |
|        | Normal | 24     | 546    |

Blood veri kümesindeki verilerden yalnızca 24 tanesi aykırı veri olarak yanlış tespit edilmiştir. Çizelge 5. 13'te yer alan değerler kullanılarak aşağıdaki hesaplamalar yapılmıştır:

- Doğru şekilde tespit edilmiş aykırı verilerin başarı oranı:

$$\text{Aykırı Değer Tespit Yüzdesi} = 154 / (154 + 24) = 0.865$$

- Doğru şekilde tespit edilmiş örneklerin başarı oranı:

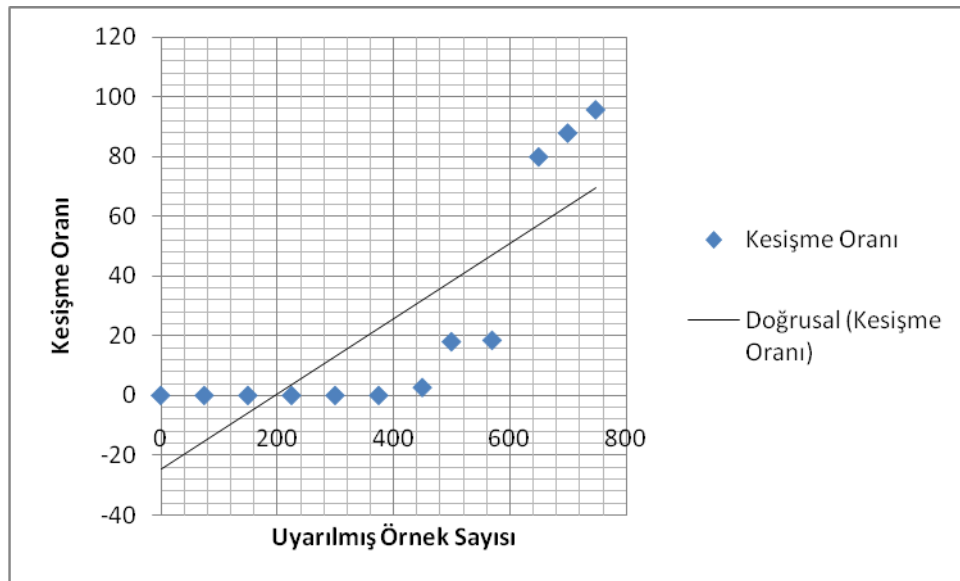
$$\text{Tüm Örnekler için Tespit Yüzdesi} = (154 + 546) / (154 + 24 + 546 + 24) = 0.935$$

- Doğru şekilde tespit edilmiş aykırı olmayan verilerin başarı oranı:

$$\text{Normal Veri Tespit Yüzdesi} = 546 / (546 + 24) = 0.957$$

- Hassasiyet (Precision) =  $154 / (154 + 24) = 0.865$

- F-ölçüm (F-measure) =  $(2 * (0.865 * 0.865)) / (0.865 + 0.865) = \mathbf{0.865}$



Şekil 5. 10 Uyarılmaya göre aykırı tespit başarı oranı

Şekil 5. 10'da, algoritma ile tanımlanan, aykırılığı en yüksek örneklerin büyük bir çoğunluğunun doğru tespit edildiği, kesişme oranının 650-748 örnek aralığında yaklaşık %80 ve üzerinde olduğu görülmektedir. Özellikle son üç kesişme oranına bakıldığında, en az uyarılmış örneklerin doğruluk oranının bir hayli yüksek olduğu gözükmemektedir.

Çizelge 5. 14'de, Çizelge 5. 15'de ve Çizelge 5. 16'da, Blood veri kümesine sunulan sırasıyla, KNN, DBO ve BP metotlarının performansları gösterilmektedir.

Çizelge 5. 17’de ise, Yapay Bağışıklık Sistem Tabanlı algoritması ile KNN, DBO ve BP tekniklerinin aykırı tespit etme başarıları karşılaştırmalı olarak verilmiştir.

Çizelge 5. 14 KNN için Hata Matrisi (Blood)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 145    | 33     |
|        | Normal | 33     | 537    |

Çizelge 5. 15 DBO için Hata Matrisi (Blood)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 149    | 29     |
|        | Normal | 29     | 541    |

Çizelge 5. 16 BP için Hata Matrisi (Blood)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 151    | 27     |
|        | Normal | 27     | 543    |



Çizelge 5. 17 Blood veri kümesi için YBA performans gösterimi (Analiz 1)

| Metotlar | Bulunan Gerçek Aykırı Veri Sayısı | F-Ölçüm |
|----------|-----------------------------------|---------|
| YBA      | 154                               | 0.86    |
| KNN      | 145                               | 0.81    |
| DBO      | 149                               | 0.83    |
| BP       | 151                               | 0.84    |

#### 5.2.3.2 Eşik Değer İndeks Sayısı Baz Alınarak Yapılan Analiz

Her bir metot için eşik değer indeksine göre aykırı değer tespit etme performansları hesaplanmıştır. Blood veri kümesi için eşik değer indeksi; YBA için 518, KNN için 537, DBO için 561 ve BP için 550 olarak hesaplanmıştır.

Çizelge 5. 18 YBA için Hata Matrisi (Blood)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 173    | 5      |
|        | Normal | 57     | 513    |

Çizelge 5. 18’de uyarılmış örnek sıralamasında yer alan 518. örnek ve sonrasındaki en az uyarılmış örnekler alınarak YBA tarafından tespit edilen aykırı veriler gösterilmiştir.

$$\text{Recall} = 173/(173+57) = 0.75$$

$$\text{Precision} = 173/(173+5) = 0.97$$

$$\text{F-Ölçüm} = \mathbf{0.84}$$

Blood veri kümesine uygulanan YBA’nın tespit ettiği gerçek aykırı veri sayısı 173’dür. Gerçek aykırı veri olmasına rağmen algoritmanın normal veri olarak tanımladığı örnek sayısı ise 5’tir. Algoritmanın bu veri kümesi üzerinde aykırı veri tespit etme başarı oranı %84’tür.

Çizelge 5. 19’da, Çizelge 5. 20’de ve Çizelge 5. 21’de KNN, DBO ve BP yöntemlerinin Blood veri kümesinde tespit ettikleri gerçek aykırı veri sayıları ve performansları gösterilmiştir. Çizelge 5. 22’de ise YBA ve diğer metotların veri kümesinde aykırı örnekleri doğru tespit etmedeki başarı oranları gösterilmiştir.

Çizelge 5. 19 KNN için Hata Matrisi (Blood)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 150    | 28     |
|        | Normal | 61     | 509    |

$$\text{Recall} = 150 / (150 + 61) = 0.71$$

$$\text{Precision} = 150 / (150 + 28) = 0.84$$

$$\text{F-Ölçüm} = \mathbf{0.76}$$

Çizelge 5. 20 DBO için Hata Matrisi (Blood)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 153    | 25     |
|        | Normal | 34     | 536    |

$$\text{Recall} = 153 / (153 + 34) = 0.81$$

$$\text{Precision} = 153 / (153 + 25) = 0.85$$

$$\text{F-Ölçüm} = \mathbf{0.82}$$

Çizelge 5. 21 BP için Hata Matrisi (Blood)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 153    | 25     |
|        | Normal | 45     | 525    |

$$\text{Recall} = 153/(153+45) = 0.77$$

$$\text{Precision} = 153/(153+25) = 0.85$$

$$\text{F-Ölçüm} = \mathbf{0.80}$$

Çizelge 5. 22 Blood veri kümesi için YBA performans gösterimi (Analiz 2)

| Metotlar | Bulunan Gerçek Aykırı Veri Sayısı | F-Ölçüm |
|----------|-----------------------------------|---------|
| YBA      | 173                               | 0.84    |
| KNN      | 150                               | 0.76    |
| DBO      | 153                               | 0.82    |
| BP       | 153                               | 0.80    |

#### 5.2.4 Yapay Veri Kümesi ile Aykırı Değer Analizi

Son çalışma kümesi olan yapay, en az aykırı örnek içeren veri kümesidir. Bu küme içerisinde yer alan rastgele oluşturulmuş 1000 örnek üzerinden yapılan çalışmada Yapay Bağışıklık Sistem algoritması ile diğer tekniklerinin performansları analiz edilmiştir. Bu küme üzerinde yapılan analizler özellik ve örnek sayılarına göre değişmektedir. Analizler üç kategori altında yapılmıştır: aykırı sayısı baz alınarak yapılan analiz, eşik değer indeks sayısı baz alınarak yapılan analiz ve yapay veri kümesi üzerinde bir takım indirgemeler ile değiştirilmiş özellik ve örnek sayılarına göre yapılan analizler.

#### 5.2.4.1 Aykırı Sayısı Baz Alınarak Yapılan Analiz

1000 örnek üzerinden gerçek aykırı veri sayısı kadar en az uyarılmış örneklerin alınmasıyla hesaplanan performans sonuçları Çizelge 5. 23'te, Çizelge 5. 24'te ve Şekil 5. 11'de gösterilmiştir.

Çizelge 5. 23 Uyarılma sırasına göre aykırı değer tespiti (YBA)

| Uyarım Örnek Sıralaması | Gerçek Aykırı Sayısı | Aykırı Oranı | Aykırı Yüzdesi |
|-------------------------|----------------------|--------------|----------------|
| 1-75                    | 0                    | 0            | 0              |
| 76-150                  | 0                    | 0            | 0              |
| 151-225                 | 0                    | 0            | 0              |
| 226-300                 | 0                    | 0            | 0              |
| 301-375                 | 0                    | 0            | 0              |
| 376-450                 | 0                    | 0            | 0              |
| 451-525                 | 0                    | 0            | 0              |
| 526-600                 | 0                    | 0            | 0              |
| 601-675                 | 0                    | 0            | 0              |
| 676-750                 | 0                    | 0            | 0              |
| 751-825                 | 1                    | 1,33         | 1              |
| 826-900                 | 3                    | 4            | 3              |
| <b>901-1000</b>         | <b>96</b>            | <b>96</b>    | <b>96</b>      |
| <b>Toplam</b>           | <b>100</b>           |              | <b>% 100</b>   |

1000 veriden oluşan yapay veri kümesinde yer alan 100 aykırı örnekten 96 [901-1000] tanesi algoritma tarafından doğru tespit edilmiş olup, kalan 4 örnek ise yanlış etiketlenmiştir. Yapay veri kümesi için aykırı örneklerin doğru tespit edilme oranı %96 olmuştur.

Çizelge 5. 24 YBA için Hata Matrisi (Yapay)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 96     | 4      |
|        | Normal | 4      | 896    |

Çizelge 5. 24'te yer alan değerler kullanılarak aşağıdaki hesaplamalar yapılmıştır:

- Doğru şekilde tespit edilmiş aykırı verilerin başarı oranı:

$$\text{Aykırı Değer Tespit Yüzdesi} = 96 / (96 + 4) = 0.96$$

- Doğru şekilde tespit edilmiş örneklerin başarı oranı:

$$\text{Tüm Örnekler için Tespit Yüzdesi} = (96 + 896) / (96 + 4 + 896 + 4) = 0.992$$

- Doğru şekilde tespit edilmiş aykırı olmayan verilerin başarı oranı:

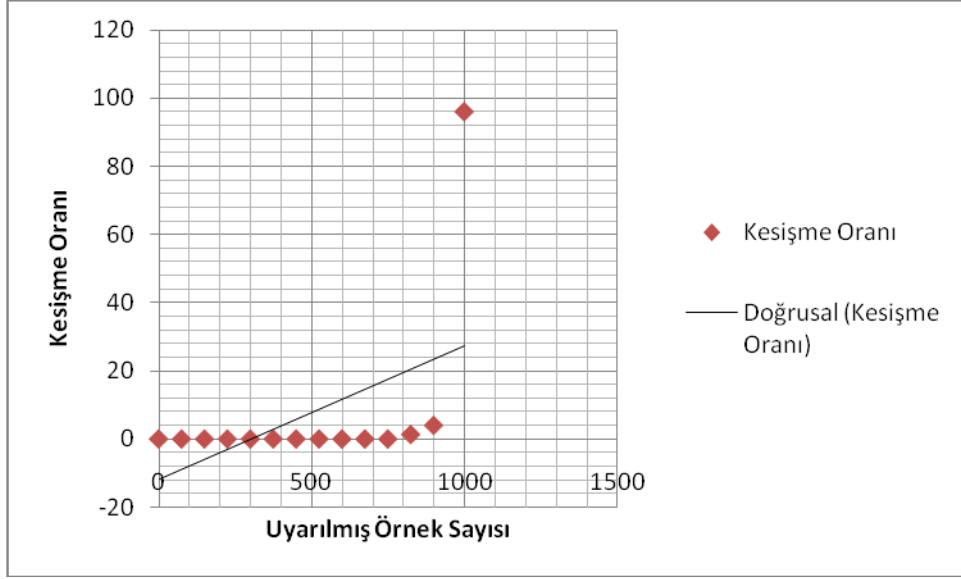
$$\text{Normal Veri Tespit Yüzdesi} = 896 / (896 + 4) = 0.995$$

- Hassasiyet (Precision) =  $96 / (96 + 4) = 0.96$

- F-ölçüm (F-measure) =  $(2 * (0.96 * 0.96)) / (0.96 + 0.96) = \mathbf{0.96}$

Yapay veri kümesinde yapılan deneysel sonuçlar, diğer veri kümelerinde yapılan deneysel sonuçlara göre bu analizde daha başarılı sonuçlar vermiştir. Bu veri kümesindeki sonuçlarda etkili olan nedenleri şöyle sıralayabiliriz:

- Veri kümesinde, aykırı değerlikli özellik sayısına bağlı olarak özellik değerlerinde belirli sapma oranlarının kullanılması.
- Normal verilerde özellik değerlerinin belirli aralıkta tutulması ve değer aralıkları arasında çok büyük farklılıkların oluşturulmaması.
- Tek bir özellikte görülen aykırılığın tespitinin zor olmasından dolayı, bu veri kümesi için oluşturulan sapmanın çok yüksek oranda tutulmasıdır.



Şekil 5. 11 Uyarılmaya göre aykırı tespit başarı oranı

Şekil 5. 11’de, en az uyarılmış 901-1000 aralığındaki örneklerin büyük bir çoğunluğunun diğer uyarılmış örneklerden net bir şekilde ayrıldığı ve kesişme oranının en az uyarılmış örnekler için çok yüksek olduğu bu yüzden doğrusal kesişmenin yatay eksene daha yakın çizildiği görülmektedir.

Çizelge 5. 25’te, Çizelge 5. 26’da ve Çizelge 5. 27’de, yapay veri kümesine sunulan KNN, DBO ve BP metotlarının deneysel sonuçları gösterilmektedir. Çizelge 5. 28’de ise, Yapay Bağışıklık Sistem algoritması ile birlikte KNN, DBO ve BP tekniklerinin aykırı değer tespit etme başarıları verilmiştir.

Çizelge 5. 25 KNN için Hata Matrisi (Yapay)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 94     | 6      |
|        | Normal | 6      | 894    |

Çizelge 5. 26 DBO için Hata Matrisi (Yapay)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
| GERÇEK |        | Aykırı | Normal |
|        | Aykırı | 91     | 9      |
|        | Normal | 9      | 891    |

Çizelge 5. 27 BP için Hata Matrisi (Yapay)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
| GERÇEK |        | Aykırı | Normal |
|        | Aykırı | 94     | 6      |
|        | Normal | 6      | 894    |

Çizelge 5. 28 Yapay veri kümesi için YBA performans gösterimi (Analiz 1)

| Metotlar | Bulunan Gerçek Aykırı Veri Sayısı | F-Ölçüm |
|----------|-----------------------------------|---------|
| YBA      | 96                                | 0.96    |
| KNN      | 94                                | 0.94    |
| DBO      | 91                                | 0.91    |
| BP       | 94                                | 0.94    |

#### 5.2.4.2 Eşik Değer İndeks Sayısı Baz Alınarak Yapılan Analiz

Yapay veri kümesi üzerine uygulanan tüm yöntemler için eşik değer indeksi hesaplanmıştır. Yapay veri kümesi için eşik değer indeksi; YBA için 827, KNN için 751, DBO için 786 ve BP için 720 olarak hesaplanmıştır.

Çizelge 5. 29 YBA için Hata Matrisi (Yapay)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 100    | 0      |
|        | Normal | 73     | 827    |

Çizelge 5. 29'da uyarılmış örnek sıralamasında yer alan 827. örnek ve sonrasındaki en az uyarılmış örnekler alınarak YBA tarafından tespit edilen aykırı veriler gösterilmiştir.

$$\text{Recall} = 100/(100+73) = 0.57$$

$$\text{Precision} = 100/(100+0) = 1.00$$

$$\text{F-Ölçüm} = \mathbf{0.72}$$

YBA, Yapay veri kümesindeki gerçek aykırı verilerin tamamını doğru tahmin etmiştir. Algoritma, normal veri olmasına rağmen aykırı olarak tanımladığı örnek sayısı 73'tür. Algoritmanın bu veri kümesi üzerinde aykırı veri tespit etme başarı oranı %72 olmuştur. Başarı oranının düşmesinde normal verilerin aykırı veri olarak tespit edilmesi etkili olmuştur.

Çizelge 5. 30'da, Çizelge 5. 31'de ve Çizelge 5. 32'de KNN, DBO ve BP yöntemlerinin Yapay veri kümesinde tespit ettikleri gerçek aykırı veri sayıları ve performansları gösterilmiştir. Çizelge 5. 33'te ise YBA ve diğer metotların Yapay veri kümesindeki aykırı örnekleri doğru tespit etmedeki başarıları gösterilmiştir.

Çizelge 5. 30 KNN için Hata Matrisi (Yapay)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 98     | 2      |
|        | Normal | 151    | 749    |



$$\text{Recall} = 98/(98+151) = 0.39$$

$$\text{Precision} = 98/(98+2) = 0.98$$

$$\text{F-Ölçüm} = \mathbf{0.55}$$

Çizelge 5. 31 DBO için Hata Matrisi (Yapay)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 94     | 6      |
|        | Normal | 120    | 780    |

$$\text{Recall} = 94/(94+120) = 0.43$$

$$\text{Precision} = 94/(94+6) = 0.94$$

$$\text{F-Ölçüm} = \mathbf{0.59}$$

Çizelge 5. 32 BP için Hata Matrisi (Yapay)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 97     | 3      |
|        | Normal | 183    | 717    |

$$\text{Recall} = 97/(97+183) = 0.34$$

$$\text{Precision} = 97/(97+3) = 0.97$$

$$\text{F-Ölçüm} = \mathbf{0.50}$$

Çizelge 5. 33 Yapay veri kümesi için YBA performans gösterimi (Analiz 2)

| Metotlar | Bulunan Gerçek Aykırı Veri Sayısı | F-Ölçüm |
|----------|-----------------------------------|---------|
| YBA      | 100                               | 0.72    |
| KNN      | 98                                | 0.55    |
| DBO      | 94                                | 0.59    |
| BP       | 97                                | 0.50    |

#### 5.2.4.3 Yapay Veri Kümesinde Örnek ve Özellik Sayılarına Göre Yapılan Analiz

Yapay veri kümesinde örnek ve özellik indirgemesi yapılarak Yapay Bağışıklık Sistem algoritmasının aykırı değer tespit etme performansı değerlendirilmiştir. Örnek indirgemesinde aykırı örnek sayısı sabit tutulmuştur. Özellik çıkarımında ise aykırı değer içermeyen tek bir özellik çıkarılmış geriye kalan 5 özellik üzerinden analiz yapılmıştır. İndirgeme ve Çıkarma işlemleri sonucu elde edilen veri kümelerinde iki türlü analiz yapılmıştır: aykırı sayısı baz alınarak yapılan analiz ve eşik değer indeks sayısı baz alınarak yapılan analiz.

##### 5.2.4.3.1 Analiz 1

Gerçek aykırı veri sayısı baz alınarak yapılan analiz sonuçları Çizelge 5. 34’te, Çizelge 5. 35’te ve Çizelge 5. 36’da gösterilmiştir.

Çizelge 5. 34 YBA için Hata Matrisi (500 örnek)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 92     | 8      |
|        | Normal | 8      | 392    |

Çizelge 5. 34'te, yapay veri kümesinden 500 normal verinin çıkartılması sonucunda geriye kalan 400 normal ve 100 anormal örnek üzerinde yapılan performans değerlendirilmesi gösterilmektedir.

Çizelge 5. 35 YBA için Hata Matrisi (750 örnek)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 94     | 6      |
|        | Normal | 6      | 644    |

Çizelge 5. 35'te ise, yapay veri kümesinden 250 normal verinin çıkartılması sonucunda geriye kalan 650 normal ve 100 anormal örnek üzerinde yapılan performans değerlendirilmesi gösterilmektedir.

Çizelge 5. 34'te ve Çizelge 5. 35'te yer alan değerler kullanılarak Çizelge 5. 36'da gösterilen sonuçlar elde edilmiştir.

Çizelge 5. 36 Örnek sayısındaki değişime bağlı analiz sonuçları (Analiz 1)

| Önek Sayısı | Bulunan Gerçek Aykırı Veri Sayısı | F-Ölçüm |
|-------------|-----------------------------------|---------|
| 500         | 92                                | 0.92    |
| 750         | 94                                | 0.94    |
| 1000        | 96                                | 0,96    |

Çizelge 5. 36'da görüldüğü gibi örnek sayısındaki düşüşler aykırı değerlerin tespit edilme başarısında çok büyük farklılıklar oluşturmamıştır. Bu durumun nedeni olarak Yapay Bağışıklık Sistem Tabanlı algoritması tarafından test kümesindeki her bir örneğin tanınmasını kolaylaştıran evrimleştirilmiş veri havuzunun oluşturulmuş olmasını gösterebiliriz.

Çizelge 5. 37 YBA için Hata Matrisi (Özellik Çıkarımı)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 91     | 9      |
|        | Normal | 9      | 891    |

Çizelge 5. 37’de, yapay veri kümesinden aykırı değer içermeyen bir özelliğin çıkartılması sonucunda geriye kalan 5 özellikli 1000 örnek üzerinde yapılan performans değerlendirilmesi gösterilmektedir. Aykırı içermeyen özelliğin çıkartılmasındaki ana neden, tek özellikte görülen aykırı örnek sayısının olası kaybını önlemektir. Çizelge 5. 37’de yer alan sonuçlara bakıldığında algoritmanın aykırı tespit etme başarısının %96’dan %91’e düştüğünü görmekteyiz.

#### 5.2.4.3.2 Analiz 2

Özellik çıkarımında ve örnek indirgemesinde, YBA tarafından uyarılan hücreler için eşik değer indeksi kullanılmış, performans değerleri Çizelge 5. 38’de, Çizelge 5. 39’da ve Çizelge 5. 40’ta gösterilmiştir. Eşik değer indeksi; 500 örnekli Yapay veri kümesi için 365, 750 örnekli Yapay veri kümesi için 601 ve 5 özellikli Yapay veri kümesi için ise 795 olarak hesaplanmıştır.

Çizelge 5. 38 YBA için Hata Matrisi (500 örnek)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 95     | 5      |
|        | Normal | 40     | 360    |

$$\text{Recall} = 95/(95+40) = 0.70$$

$$\text{Precision} = 95/(95+5) = 0.95$$

$$\text{F-Ölçüm} = \mathbf{0.80}$$

Çizelge 5. 39 YBA için Hata Matrisi (750 örnek)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 97     | 3      |
|        | Normal | 52     | 598    |

$$\text{Recall} = 97/(97+52) = 0.65$$

$$\text{Precision} = 97/(97+3) = 0.97$$

$$\text{F-Ölçüm} = \mathbf{0.77}$$

Çizelge 5. 40 Örnek sayısındaki değişime bağlı analiz sonuçları (Analiz 2)

| Önek Sayısı | Bulunan Gerçek Aykırı Veri Sayısı | F-Ölçüm |
|-------------|-----------------------------------|---------|
| <b>500</b>  | 95                                | 0.80    |
| <b>750</b>  | 97                                | 0.77    |
| <b>1000</b> | 100                               | 0,72    |

Çizelge 5. 40, Çizelge 5. 38’de ve Çizelge 5. 39 ‘da yer alan sayısal veriler kullanılarak hesaplanmıştır. Örnek sayısında yapılan artış ile birlikte eşik değer indeksinin de kullanılması başarı oranını düşürmüştür. Orandaki bu düşüş, örnek sayısındaki artışa bağlı eşik değer indeks değerinin düşmesinden kaynaklanmaktadır.

Çizelge 5. 41 YBA için Hata Matrisi (Özellik Çıkarımı)

|        |        | TESPİT |        |
|--------|--------|--------|--------|
|        |        | Aykırı | Normal |
| GERÇEK | Aykırı | 98     | 2      |
|        | Normal | 107    | 793    |

$$\text{Recall} = 98/(98+107) = 0.47$$

$$\text{Precision} = 98/(98+2) = 0.98$$

$$\text{F-Ölçüm} = \mathbf{0.63}$$

Çizelge 5. 41’de tek bir özelliğin çıkartılması sonucu eşik değer indeksli analiz sonuçları gösterilmiştir. Algoritmanın başarısı özellik çıkarımından dolayı %63 seviyesine düşmüştür.

Veri kümesinden bir özelliğin çıkartılması, anormal verilerin normal verilerden ayırt edilmesini güçleştirmiştir. Çünkü, özellik sayısındaki azalış evrimleşme sürecini kısaltmıştır. Bu durum daha az kaynak yarışmasının yapıldığını gösterir. Buradan çıkartacağımız sonuç ise, özellik sayısındaki artışların etkin kaynaklara ulaşmayı geciktirdiği, bunun sonucunda da daha fazla evrimleşmenin gerçekleşip daha etkin kaynakların elde edileceğidir.

### SONUÇ ve ÖNERİLER

Yapay Bağışıklık Sistemleri, karmaşık problemleri çözmedeki başarılı performansı ile ilgi odağı haline gelmiştir. YBS, veri madenciliği konusuna giren birçok alanda başarıyla uygulanmıştır. Özellikle son yıllarda Yapay Bağışıklık Sistemleri ile optimizasyon ve sınıflandırma üzerine çok başarılı algoritmalar geliştirilmiştir. Bağışıklık Ağ Teorisi, Klonal Seçim Algoritması, Negatif Seçim Algoritması, Genetik Algoritmalar ve Yapay Bağışıklık Tanıma Sistemi Algoritması gibi farklı algoritmalar modellenmiş ve birçok uygulamada kullanılmıştır.

Yapay Bağışıklık Sistem Tabanlı Algoritması bir yapay zeka tekniği olup, problem çözme metodu olarak günümüzde kullanılmaktadır. Bu tez çalışmasında ise, doğal bağışıklık sisteminden esinlenen YBA, aykırı verilerin tespitinde kullanılmıştır.

Yapay Bağışıklık Sistem Tabanlı Algoritması, çalışmada kullanılan veri kümeleri üzerinde uygulanmış, algoritmanın aykırı değer tespit etme başarısı analiz edilmiştir. Algoritmanın performansı, K - En Yakın Komşuluk algoritması (KNN), Uzaklık Tabanlı Aykırı Değer Tespit algoritması (DBO) ve Kutu Grafik (BP) tekniklerinin performansları ile karşılaştırılmıştır. Çalışmada kullanılan veri kümelerinden ikisi gerçek hayattan alınmış araştırma sonuçlarını gösterirken, diğer veri kümesi ise rastgele değerlerle oluşturulmuş bir yapay kümeyi temsil etmektedir.

İlk veri kümesi olan Pima veri kümesinde yapılan birinci analizde, YBA'ın aykırı değer tespit etme başarısı % 88 olurken; diğer tekniklerin başarı oranları KNN için % 82.8, DBO için % 80.2 ve BP için % 83.9 olmuştur. Pima veri kümesinde yapılan ikinci analizde ise başarı oranları YBA için % 85 iken; KNN için % 85, DBO için % 83 ve BP için % 84

olarak hesaplanmıştır. Bir diğer veri kümesi olan Blood veri kümesinde yapılan ilk deneysel analizde YBA'nın aykırı değer tespit etme başarısı % 86.5 iken; diğer tekniklerin başarı oranları KNN için % 81.4, DBO için % 83.7 ve BP için % 84.8 olarak hesaplanmıştır. Blood veri kümesinde yapılan ikinci analizde ise başarı oranları YBA için % 84 olurken, KNN için % 76, DBO için % 82 ve BP için % 80 olmuştur. Son veri kümesi olan yapay veri kümesinde yapılan çalışmalarda, ilk olarak kümedeki tüm örnekler üzerinden analizler yapılmıştır. İlk analizde YBA'nın aykırı değer tespit etme başarısı % 96 iken, diğer tekniklerin başarı oranları ise KNN için % 94, DBO için %91 ve BP için % 94 olmuştur. İkinci analizdeki başarı oranları ise; YBA için % 72, KNN için % 55, DBO için % 59 ve BP için % 50 olarak hesaplanmıştır. Yapay veri kümesinde örnek azaltma ve özellik çıkarma işlemleri yapılmış, bu işlemlerle ilgili YBA'nın performansı ayrıca analiz edilmiştir. Örnek sayılarına göre bakıldığında, 500 örnek üzerinden yapılan ilk analizde YBA'nın aykırı değer tespit etme başarı oranı %92 iken, 500 örnek üzerinden yapılan ikinci analizde bu oran % 80 olmuştur. 750 örnek üzerinden yapılan ilk analizde YBA'nın başarı oranı %94 iken, aynı sayıda örnek üzerinde yapılan ikinci analizimizde ise bu oran % 77 olmuştur. Son olarak özellik azaltma ile yapılan performans değerlendirilmesinde YBA'nın aykırı değer tespit etme başarısı ilk analizde %91 olarak hesaplanmıştır. Aynı özelliğin çıkartılması ile birlikte Yapay veri kümesine ikinci bir analiz uygulanmış, bu analize göre YBA'nın başarı oranı % 63 olmuştur. Genel olarak tüm metotların ilk analizdeki performans değerleri, ikinci analizde hesaplanan performans değerlerine göre daha yüksek olmuştur.

Sonuç olarak, Yapılan tüm deneysel analizlerde Yapay Bağışıklık Sistem Tabanlı algoritması diğer tekniklere göre daha başarılı bir performans göstermiştir. Bu da Yapay Bağışıklık Sistem algoritmasının aykırı değer tespit etmede etkili bir yöntem olduğunu göstermektedir.

Gelecek çalışmalarda Yapay Bağışıklık Sistem Tabanlı algoritmasının aykırı değer tespitindeki performansı, diğer yapay zeka tekniklerinden olan Genetik Algoritmalar, Parçacık Sürü Optimizasyonu, Diferansiyel Gelişim Algoritması ve Yapay Arı Koloni Algoritmaları gibi tekniklerin performansları ile karşılaştırılabilir.



- [1] Zhang, Y., Meratnia, N. ve Havinga, P., (2010). Outlier Detection Techniques for Wireless Sensor Networks: A Survey, IEEE Communications Surveys & Tutorials, 12(2).
- [2] Kaya, A., (1995). Kalite kontrol amaçlı zaman serilerinde sapan değer (Outlier) analizi, Yüksek Lisans Tezi, Ege Üniversitesi, Fen Bilimleri Enstitüsü, İzmir.
- [3] Bulut, H.E., (2001), Temel bileşenler analizinde aykırı değer sorunu ve aykırı değerlerin Robust temel bileşenler analizi ile belirlenmesi, Yüksek Lisans Tezi, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.
- [4] Aşıkgil, B., (2006), Çoklu doğrusal regresyonda aykırı, etkili değerlerin araştırılması ve bir uygulama, Yüksek Lisans Tezi, MSGSÜ, Fen Bilimleri Enstitüsü, İstanbul.
- [5] Vural, A., (2007), Aykırı değerlerin regresyon modellerine etkileri ve sağlam kestiriciler, Yüksek Lisans Tezi, Marmara Üniversitesi, Sosyal Bilimleri Enstitüsü, İstanbul.
- [6] Gürünlü Alma, Ö., (2009), Genetic algorithm based outlier detection using information criterion, Doktora Tezi, Ege Üniversitesi, Fen Bilimleri Enstitüsü, İzmir.
- [7] Doğan, Y., (2009), Outlier detection with K nearest neighbor clustering, Yüksek Lisans Tezi, Dokuz Eylül Üniversitesi, Fen Bilimleri Enstitüsü, İzmir.
- [8] Chen, S., Wang, W. ve Van Zuylen, H., (2010). A comparison of outlier detection algorithms for ITS data, Transportation College, Southeast University, Elsevier Pres., 37:1169-1178, China.
- [9] Yıldırım, N., (2010), En küçük kareler, Ridge regresyon ve Robust regresyon yöntemlerinde analiz sonuçlarına aykırı değerlerin etkilerinin belirlenmesi, Yüksek Lisans Tezi, Çukurova Üniversitesi, Fen Bilimleri Enstitüsü, Adana.
- [10] Turgut, K., (2010), Çok değişkenli veride aykırı değerlerin tespiti için MVV yöntemi ve diğer yöntemlerle karşılaştırılması, Yüksek Lisans Tezi, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.

- [11] Koğar, H., (2010), Farklı örneklem büyüklüklerinde uç değerlerle baş etme yöntemlerinin puanların geçerlik ve güvenirlik kanıtları üzerindeki etkisi, Yüksek Lisans Tezi, Ankara Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.
- [12] Chen, Y., Miao, D. ve Zhang, H., (2010). Neighborhood Outlier Detection, Department of Computer Science and Technology, Xiamen University of Technology, Elsevier Pres., 37:8745-8749, China.
- [13] Yavuzkanat, M. K., (2011), Lojistik regresyonda çoklu aykırı gözlemlerin belirlenmesi ve etkililiklerinin incelenmesi, Yüksek Lisans Tezi, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.
- [14] Erdoğan, G., (2012). Spectral methods for outlier detection in machine learning, Yüksek Lisans Tezi, Boğaziçi Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- [15] Wong, N., Ray, P., Stephens, G. ve Lewis, L., (2012). Artificial immune systems for the detection of credit card fraud: an architecture, prototype and preliminary results, Information Systems Journal, 22:53-76, USA.
- [16] Akşit, F., Akgün, Y. ve Kiraz, N., (1996). Mikrobiyoloji, T.C. Anadolu Üniversitesi Açıköğretim Fakültesi Yayınları, Yayın No:219, Eskişehir.
- [17] Babayigit, B., ve Güney, K., (2006). Klonal Seçme Algoritması Kullanılarak Genlik ve Faz Kontrolü ile Doğrusal Anten Dizi Diyagramında Sıfırların Üretilmesi, Elektrik-Elektronik-Bilgisayar Mühendisliği Sempozyumu (ELECO'2006), 72-76, Bursa.
- [18] Ada, G.L., ve Nossal, G., (1987). The Clonal Selection Theory, Scientific American, 257:50-57.
- [19] De Castro, L.N. ve Von Zuben, F.J., (2002). Learning and Optimization Using The Clonal Selection Principle, IEEE Pres., 6:239-251.
- [20] De Castro, L.N. ve Timmis, J.I., (2003). Artificial Immune Systems as Novel Soft Computing Paradigm, Soft Computing Journal, 7:1-31.
- [21] Igava, K. ve Ohashi, H., (2009). A Negative Selection Algorithm for Classification and Reduction of the Noise Effect, Applied Soft Computing, 9: 431-438.
- [22] Watkins, A., (2001). AIRS: A Resource Limited Artificial Immune Classifier, Yüksek Lisans Tezi, Mississippi State University, Mississippi, ABD.
- [23] Watkins, A. ve Timmis, J., (2002). Artificial Immune Recognition System (AIRS): Revisions and Refinements, International Conference on Artificial Immune Systems, University of Kent at Canterbury, United Kingdom.
- [24] Kodaz, H., (2007). Bilgi Kazancı Tabanlı Yapay Bağışıklık Tanıma Sistemi, Doktora Tezi, Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Konya.
- [25] Brownlee, J., (2005). Artificial Immune Recognition System (AIRS): A Review and Analysis, Centre for Intelligent Systems and Complex Processes, Faculty of Information and Communication Technologies, Swinburne University of Technology, Victoria, Australia.

- [26] Smith, D.J., Forrest, S. ve Perelson, A.S., (1999). Artificial Immune Systems and Their Application, Chapter 6, Springer-Verlag, Berlin.
- [27] Forrest, S., Perelson, A.S., Allen, L. ve Cherukuri, R., (1994). Self-Nonself Discrimination in a Computer, in Proceeding of the IEEE Symposium on Research in Security and Privacy, Oakland, 202-212.
- [28] Segel, L., (1997). The Immune Systems As a Prototype of Autonomous Decentralized Systems, in Proceeding of the IEEE International Conference on Systems, Man and Cybernetics, 1:375-385.
- [29] Dapagusta, D., (1998). An Artificial Immune Systems as a Multi-Agent Decision System, in Proceeding of the 1998 IEEE International Conference on Systems, Man and Cybernetics, IEEE Press., 3816-3820.
- [30] Hightower, R., Forrest, S. ve Perelson, A.S., (1996). The Baldwin Effect in The Immune Systems: Learning by Somatic Hypermutation, Addison-Wesley, 159-167.
- [31] Farmer, J.D., Packard, N.H. ve Perelson, A.S., (1986). The Immune System, Adaptation, and Machine Learning, Physica, 187-204.
- [32] Bersini, H. ve Varela, F.J., (1991). The Immune Recruitment Mechanism: A Selective Evolutionary Strategy, in Proceeding of the 4th International Conference on Genetic Algorithms, 520-536.
- [33] Forrest, S., Jovarnik, B., Smith, R. ve Perelson, A.S., (1993). Using Genetic Algorithm to Explore Pattern Recognition in The Immune System, Evolutionary Computation 1, 191-211.
- [34] Greensmith, J., (2003). New Frontiers For An Artificial Immune System, MSc Thesis, Multidisciplinary Informatics, University of Leeds, United Kingdom.
- [35] Hamaker, J. ve Boggess, L., (2004). Non-Euclidean Distance Measures in AIRS, an Artificial Immune Classification System, Proceedings of the 2004 Congress on Evolutionary Computation, 1067-1073.
- [36] Parmaksızoğlu, S., (2005). Yapay bağışıklık algoritması kullanılarak lineer sistemlerin kimliklendirilmesi, Yüksek Lisans Tezi, Erciyes Üniversitesi, Fen Bilimleri Enstitüsü, Kayseri.
- [37] Watkins, A., (2005). Exploiting Immunological Metaphors in the Development of Serial, Parallel, and Distributed Learning Algorithms, PhD Thesis, Kent University.
- [38] Kaymaz, E.D., (2007). Yapay bağışıklık sistemi tabanlı k-NN sınıflandırma algoritması ile protein örüntülerinin hücredeki yerleşim yerlerinin belirlenmesi, Yüksek Lisans Tezi, Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elazığ.
- [39] Çatal, Ç., (2008), Yazılım kusur kestirimi probleminde yapay bağışıklık sistemlerinin uygulanması, Doktora Tezi, YTÜ, Fen Bilimleri Enstitüsü, İstanbul.

- [40] Karslı, N., (2010). Akıllı ulaşım sistemleri için yapay bağışıklık sistemleri ve genetik algoritma ile yeni stokastik en kısa yol algoritmalarının geliştirilmesi, Doktora Tezi, Atatürk Üniversitesi, Fen Bilimleri Enstitüsü, Erzurum.
- [41] Şahan, S., Ceylan, R. ve Güneş, S., (2004). PMR Kodlamasında Yapay Bağışıklık Sistemleri ve YSA, SIU'04, 121-124.
- [42] Eren, Ö., (2008). Alerjen Proteinlerin Otomatik Sınıflandırılması, Yüksek Lisans Tezi, Başkent Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.
- [43] Eymen, U.E., (2007). Veri Analiz Yöntemleri, İstatistik Merkezi Yayın No:1, İstanbul.
- [44] Vijayarani, S. ve Nithya, S., (2011). An Efficient Clustering Algorithm for Outlier Detection, International Journal of Computer Applications, 32(7).
- [45] Pearson, R. K., (2006). The Problem of Disguised Missing Data, ACM SIGKDD Explorations, 8(1):83-92, USA.
- [46] Breault, J.L., (2001). Data Mining Diabetic Databases: Are Rough Sets a Useful Addition?, Alton Ochsner Medical Foundation, Tulane University, New Orleans.
- [47] Xue, Z., Shang, Y. ve Feng, A., (2010). Semi-supervised outlier detection based on fuzzy rough C-means clustering, Henan University of Science and Technology, Journal: Mathematics and Computers in Simulation, Elsevier Pres., 80:1911-1921, China.
- [48] Singh, V. ve Nagpal, S., (2011). Interactive Knowledge Discovery in Blood Transfusion Data Set, International Journal of Computer Science & Information Technology, 1(8):541-547, India.
- [49] Santhanam, T. ve Sundaram, S., (2010). Application of CART Algorithm in Blood Donors Classification, Journal of Computer Science 6(5):548-552, India.
- [50] Li, J., Cuesta-Albertos, J.A. ve Liu, R.Y., (2010). DD-Classfier: Nonparametric Classification Procedure Based on DD-plot, National Science Foundation.

## ÖZGEÇMİŞ

---

### KİŞİSEL BİLGİLER

**Adı Soyadı** : Mehmet GÜÇLÜ  
**Doğum Tarihi ve Yeri** : 22 / 03 / 1985 – Zile / TOKAT  
**Yabancı Dili** : İngilizce – Fransızca  
**E-posta** : guclu.mehmet@adalet.gov.tr

### ÖĞRENİM DURUMU

| Derece    | Alan                    | Okul/Üniversite             | Mezuniyet Yılı |
|-----------|-------------------------|-----------------------------|----------------|
| Y. Lisans | Bilgisayar Mühendisliği | Yıldız Teknik Üniversitesi  | 2012           |
| Lisans    | Bilgisayar Mühendisliği | Doğuş Üniversitesi          | 2009           |
| Lisans    | İşletme                 | Anadolu Üniversitesi        | 2010           |
| Lise      | Fen Bilimleri           | İhsan Mermerci Lisesi (YDA) | 2003           |

### İŞ TECRÜBESİ

| Yıl        | Firma/Kurum                         | Görevi   |
|------------|-------------------------------------|----------|
| 2011 - ... | B. Adalet Sarayı / Adalet Bakanlığı | Mühendis |