T.C.
İSTANBUL ÜNİVERSİTESİ-CERRAHPAŞA
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

ÇOCUKLARIN ANTROPOMETRİK VE MOTORİK ÖZELLİKLERİNDEN YAPAY ZEKA KULLANARAK ÖZELLİK SINIFLANDIRMASI VE ÇIKARIMI

Uğur ÖZBALKAN

DANIŞMAN
Dr. Öğr. Üyesi Özgür Can TURNA

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

İSTANBUL-2020

Bu çalışma, 17.07.2020 tarihinde aşağıdaki jüri tarafından Bilgisayar Mühendisliği Anabilim Dalı, Bilgisayar Mühendisliği Programında Yüksek Lisans tezi olarak kabul edilmiştir.

Tez Jürisi

Dr. Öğr. Üyesi Özgür Can TURNAI(Danışman)
İstanbul Üniversitesi-Cerrahpaşa
Mühendislik Fakültesi

Doç. Dr. Muhammed Ali AYDINI
İstanbul Üniversitesi-Cerrahpaşa,
Mühendislik Fakültesi

Dr. Öğr. Üyesi Vecdi Emre LEVENT
Fenerbahçe Üniversitesi
Mühendislik ve Mimarlık Fakültesi

Unvan Bir öğe seçin.
Bir öğe seçin.
Bir öğe seçin.

Unvan Bir öğe seçin.
niversite
Fakülte

20.04.2016 tarihli Resmî Gazete'de yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince; Bu Lisansüstü teze, İstanbul Üniversitesi-Cerrahpaşanın abonesi olduğu intihal yazılım programı kullanılarak Lisansüstü Eğitim Enstitüsünün belirlemiş olduğu ölçütlere uygun rapor alınmıştır.

ÖN SÖZ

Tez çalışmamda değerli bilgi, birikim ve tecrübeleriyle destek olan danışman hocam sayın Dr. Öğretim Üyesi Özgür Can TURNA'ya sonsuz teşekkür ve saygılarımı sunuyorum.

Tez konumu belirlerken ve tez çalışmalarım boyunca yardımlarını esirgemeyen değerli arkadaşım Araştırma Görevlisi Taylan Hayri BALCIOĞLU'na, tezde kullandığım verilerin alınmasında yardımcı olan Doç. Dr. Nuri TOPSAKAL hocama, tez çalışmam boyunca desteklerini ve önerilerini göstermekten kaçınmayan Dr. Öğretim Üyesi Vecdi Emre LEVENT, Dr. Öğretim Üyesi Ümit TERZİ, Öğretim Görevlisi Egemen Ozan SÖNMEZ ve Murat SÜKUTİ'ye teşekkürü bir borç bilirim.

Tez çalışmam boyunca maddi manevi destekleriyle beni yalnız bırakmayan değerli aileme de sonsuz teşekkürlerimi sunuyorum.

Haziran 2020 Uğur ÖZBALKAN

İÇİNDEKİLER

Sayfa No

ÖN SÖZ ..... **iv**
İÇİNDEKİLER ..... **v**
ŞEKİL LİSTESİ ..... **vi**
TABLO LİSTESİ ..... **viii**
SİMGE VE KISALTMA LİSTESİ ..... **xi**
ÖZET ..... **xii**
SUMMARY ..... **xiv**
1. GİRİŞ **1**
2. GENEL KISIMLAR **4**
2.1. YAPAY ZEKA 4
2.2. MAKİNE ÖĞRENMESİ 6
2.3. KARAR AĞACI ALGORİTMALARI 7
2.4. DESTEK VEKTÖR MAKİNESİ 10
2.5. YAPAY SİNİR AĞLARI 12
2.6. PYTHON 14
2.7. ÖZELLİKLER 14
3. MALZEME VE YÖNTEM **17**
4. BULGULAR **24**
5. TARTIŞMA VE SONUÇ **45**
KAYNAKLAR **47**
EKLER **51**
ÖZ GEÇMİŞ **52**

ŞEKİL LİSTESİ

Sayfa No

Şekil 2.1: Veri Bilimindeki Kavramlar [20]. ……..7

Şekil 2.2: İki Dallı Karar Ağacı [23]. ……8

Şekil 2.3: Çok Dallı Karar Ağacı [23]. ……8

Şekil 2.4: Doğrusal Hiper Düzlem. ……...11

Şekil 2.5: Doğrusal Olmayan Hiper Düzlem. ……...11

Şekil 2.6: İnsandaki Sinir Hücresi Yapısı [37]. ……13

Şekil 2.7: Yapay Nöron [37]. …… 13

Şekil 3.1: Weka Açılış Ekranı. ……17

Şekil 3.2: Weka Explorer Penceresi. ……18

Şekil 3.3: Arff Uzantılı Dosya. …….. 18

Şekil 3.4: Dosya Yüklendikten Sonra Oluşan Ekran. ……...19

Şekil 3.5: Classify Penceresine Tıklanıldığında Açılan Ekran. ……20

Şekil 4.1: Cinsiyet İçin Karar Ağacı Örneği. ……38

Şekil 4.2: Sağ Omuz Esnekliği İçin Karar Ağacı Örneği. ……39

Şekil 4.3: Sol Omuz Esnekliği İçin Karar Ağacı Örneği. ……..39

Şekil 4.4: 20m Sürat İçin Karar Ağacı Örneği. ……41

Şekil 4.5: Yaş İçin Karar Ağacı Örneği. ……...41

Şekil 4.6: 10m Sürat İçin Karar Ağacı Örneği. ……42

Şekil 4.7: Python'da Yapılan Korelasyon Sonucu. ……...42

TABLO LİSTESİ

Sayfa No

Tablo 2.1: Yapay Zeka Devrim Çizelgesi [15]. …….5

Tablo 2.2: Sağ Omuz Esnekliği, Sol Omuz Esnekliği ve Cinsiyetin Veri Setindeki İfadeleri.15

Tablo 2.3: Özelliklerin Değerleri. ……16

Tablo 4.1: Sağ Omuz Esnekliği ve Sol Omuz Esnekliği İçin Karar Ağaçlarıyla Elde Edilen Sonuçlar. ……...24

Tablo 4.2: J48 Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi. …25

Tablo 4.3: J48 Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Oranlar. ……...25

Tablo 4.4: Decision Stump Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi. …….25

Tablo 4.5: Decision Stump Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Oranlar. ...26

Tablo 4.6: Hoeffding Tree Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi. …….26

Tablo 4.7: Hoeffding Tree Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Oranlar. ….26

Tablo 4.8: LMT Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi. 27

Tablo 4.9: LMT Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Oranlar. ……27

Tablo 4.10: Random Forest Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi. …….27

Tablo 4.11: Random Forest Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Oranlar. ..28

Tablo 4.12: Random Tree Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi. …….28

Tablo 4.13: Random Tree Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Oranlar. …28

Tablo 4.14: REPTree Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi. …….29

Tablo 4.15: REPTree Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Oranlar. …….29

Tablo 4.16: J48 Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi. ...29

Tablo 4.17: J48 Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Oranlar. …….30

Tablo 4.18: Decision Stump Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi. …….30

Tablo 4.19: Decision Stump Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Oranlar. .30

Tablo 4.20: Hoeffding Tree Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi. …….31

Tablo 4.21: Hoeffding Tree Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Oranlar. …31

Tablo 4.22: LMT Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi.31

Tablo 4.23: LMT Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Oranlar. ……....32

Tablo 4.24: Random Forest Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi. …….32

Tablo 4.25: Random Forest Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Oranlar. …32

Tablo 4.26: Random Tree Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi. …….33

Tablo 4.27: Random Tree Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Oranlar. …33

Tablo 4.28: REPTree Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi. …….33

Tablo 4.29: REPTree Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Oranlar. ……..34

Tablo 4.30: Cinsiyet İçin Karar Ağacı Algoritması Sonuçları. …….34

Tablo 4.31: J48 Algoritmasıyla Cinsiyet İçin Elde Edilen Karmaşıklık Matrisi. …….35

Tablo 4.32: J48 Algoritmasıyla Cinsiyet İçin Elde Edilen Oranlar. ……..35

Tablo 4.33: Hoeffding Tree Algoritmasıyla Cinsiyet İçin Elde Edilen Karmaşıklık Matrisi. ...35

Tablo 4.34: Hoeffding Tree Algoritmasıyla Cinsiyet İçin Elde Edilen Oranlar. ……..36

Tablo 4.35: LMT Algoritmasıyla Cinsiyet İçin Elde Edilen Karmaşıklık Matrisi. ……..36

Tablo 4.36: LMT Algoritmasıyla Cinsiyet İçin Elde Edilen Oranlar. ……..36

Tablo 4.37: Random Forest Algoritmasıyla Cinsiyet İçin Elde Edilen Karmaşıklık Matrisi. ..37

Tablo 4.38: Random Forest Algoritmasıyla Cinsiyet İçin Elde Edilen Oranlar. ……..37

Tablo 4.39: REPTree Algoritmasıyla Cinsiyet İçin Elde Edilen Karmaşıklık Matrisi. …….37

Tablo 4.40: REPTree Algoritmasıyla Cinsiyet İçin Elde Edilen Oranlar. ……38

Tablo 4.41: Cinsiyet, Sağ Omuz Esnekliği ve Sol Omuz Esnekliği İçin Destek Vektör Makinesi ve Yapay Sinir Ağı İle Elde Edilen Sonuç. ……...40

Tablo 4.42: 20m Sürat İçin Karar Ağacı Algoritmaları ve Destek Vektör Makinesiyle Elde Edilen Sonuçlar. ……40

Tablo 4.43: 10m Sürat ve Yaş İçin Korelasyon Öncesi Elde Edilen Sonuçlar. ……43

Tablo 4.44: 10m Sürat ve Yaş İçin Korelasyon Sonrası Elde Edilen Sonuçlar. ……43

Tablo 4.45: Cinsiyet, Vücut Ağırlığı, Durarak Uzun Atlama ve Mekik Verildiğinde 10m Sürat İçin Elde Edilen Karar Ağacı Algoritmaları ve Destek Vektör Makinesi Sonucu. ……44

SİMGE VE KISALTMA LİSTESİ

Kısaltmalar	Açıklama
cm	: Santimetre
FN	: Hatalı Negatifler
FP	: Hatalı Pozitifler
kg	: Kilogram
LMT	: Logitic Model Trees (Mantıksal Model Ağaçları)
m	: Metre
REPTree	: Reduced Error Pruning Tree (Azaltılmış Hata Budama Ağacı)
sn	: Saniye
TN	: Toplam Negatifler
TP	: Toplam Pozitifler
Weka	: Waikato Environment for Knowledge Analysis

ÖZET

YÜKSEK LİSANS TEZİ

ÇOCUKLARIN ANTROPOMETRİK VE MOTORİK ÖZELLİKLERİNDEN YAPAY ZEKA KULLANARAK ÖZELLİK SINIFLANDIRMASI VE ÇIKARIMI

Uğur ÖZBALKAN

İstanbul Üniversitesi-Cerrahpaşa

Lisansüstü Eğitim Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

Danışman : Dr. Öğr. Üyesi Özgür Can TURNA

Sporun dünyadaki popülerliğinin artması, topluma ve kişiye sağladığı faydalar dikkate alındığında ilginin sporcuların ortaya koydukları sportif performansa doğru yönelmesini sağlamıştır. Sportif performans göz önüne alındığında çocukların büyümesinin ve gelişiminin, takip edilebilmesi için antropometrik ve motorik testler yapılmaktadır. Bu çalışmada kuvvet (pençe kuvveti, mekik), denge (flamingo), sürat (10m - 20m), çeviklik (zigzag), esneklik (otur uzan, omuz esnekliği), güç (durarak uzun atlama) gibi motorik özellikler ve vücut ağırlığı, oturma yüksekliği, boy uzunluğu, kulaç uzunluğu gibi antropometrik özellikler için bu testler yapılmıştır. 6-14 yaşları arasında 7505 çocuğa (4552 erkek - 2953 kız) ait ölçümler, makine öğrenmesi yöntemleriyle işlenerek sonuçlar elde edilmiştir. Weka, Python gibi araçlar kullanılarak Yapay Sinir Ağı, Destek Vektör Makinesi ve Karar Ağaçları yöntemleriyle de veriler işlenerek özellik çıkarımı yapılmıştır. Bir özelliğe ilişkin kestirim yapılabilmesi için en az sayıda hangi özelliklerin kullanılabileceğini saptamak hedeflenmiştir. Cinsiyet, Sağ Omuz Esnekliği, Sol Omuz Esnekliği, 10m Sürat, 20m Sürat ve Yaş özellikleri diğer özellikler verildiğinde %74 - %88 aralığında doğruluk oranı elde edilmiştir. 10m Sürat özelliği durarak uzun atlama, mekik, vücut ağırlığı ve cinsiyet özellikleri verilerek %75-%78 doğruluk oranıyla kestirilebildiği gözlemlenmiştir. Bu bilgiler ışığında 10m sürat özelliği başta olmak üzere elde edilen diğer sonuçlarla bu verilerden çıkarım yapacak ve faydalanacak kişiler için sportif değerlendirme açısından kolaylık, objektiflik ve zaman tasarrufu sağlayacağı düşünülmektedir.

Haziran 2020, 51 sayfa.

Anahtar kelimeler: Weka, Python, Yapay Sinir Ağı, Destek Vektör Makinesi, Karar Ağaçları, Motorik, Antropometrik

SUMMARY

M.Sc. THESIS

FEATURE CLASSIFICATION AND INFERENCE BY USİNG ARTIFICIAL INTELLIGENCE FROM ANTHROPOMETRIC AND MOTORIC CHARACTERISTIC OF CHILDREN

Uğur ÖZBALKAN

Istanbul University-Cerrahpasa

Institute of Graduate Studies

Department of Computer Engineering

Supervisor : Assist. Prof. Dr. Özgür Can TURNA

The increasing popularity of sport in the World, and benefits it provides to the society and the person have brought an interest in sportive performance of athletes. When sportive performance is considered, anthropometric and motoric tests are carried out in order to follow up growth and evolution of children. In this study, these tests were carried out for motoric properties such as strength (hand grip, sit-up), balance (flamingo), speed (10m and 20m), agility (zigzag), flexibility (sit and reach, shoulder flexibility), power (standing long jump), and anthropometric properties such as weight, seating height, height, arm span. The results are obtained by processing measurements of 7505 children (4552 boys - 2953 girls) among 6-14 ages with machine learning method. By using tools such as Weka, Python, data is processed by Artificial Neural Network, Support Vector Machine and Decision Trees methods, and feature extraction is done. The aim is to determine minimum number of feature usage to predict a feature. When Gender, Right Shoulder Flexibility, Left Shoulder Flexibility, 10m Speed, 20m Speed and Age features are given, 74% - 88% accuracy rate is obtained. It has been observed that the 10m speed feature can be estimated with 75% to 78% accuracy by giving standing long jump, sit-up, weight and features characteristics. Considering this information, it is thought that it will provide ease, objectivity and time saving to the coach in order to make deductions from the data in terms of sportive evaluation with other results, especially 10m speed feature.

June 2020, 51. pages.

Keywords: Weka, Python, Artificial Neural Network, Support Vector Machine, Decison Trees, Motoric, Anthropomethric

1. GİRİŞ

Sporun dünyada popülerliğinin artması sonucu ilgi durumunu da sporcuların ortaya koydukları performansa doğru yönlendirmiştir. Yetenek söz konusu olduğunda kalıtsal veya sonradan kazanılmış davranış koşullarından dolayı sporda verimlilik için özel bir yatkınlık veya yüksek düzeyde yatkınlık olduğu düşünülen bireyler anlaşılmaktadır. Bu durumda, yetenekli çocukları seçmek için kişinin bedeni, güç ve yeteneğini, karmaşık nitelikteki motorik becerilerini belirleyen ögeler ve insan vücudunun boyutlarıyla ilgilenen özel bir bilim dalı olan antropometrik ölçüler kullanılarak sistematik bir çalışma yapılmalıdır. 6-14 yaş aralığında çocuklardan antropometrik ve motorik özelliklerinin alınması yapılacak işlemler için önem taşımaktadır. Bu çalışmadaki veriler çeşitli amatör ve profesyonel gruplarda ölçümler gerçekleştiren bir performans ölçüm ekibinden sağlanmıştır.

Sporun kişiye ve topluma sağladıkları göz önüne alındığında, süreklilik gösteren sosyal bir ihtiyaç olgusu haline geldiği yadsınamaz bir gerçeklik oluşturmaktadır. Spor, bireyi toplumsal açıdan üretken hâle getirirken uluslararası şampiyonalar da toplumları ortak duygu ve düşünce temellerinde birleştirir. Sporda başarıyı yakalamış toplumlar kendilerinden söz ettirmeyi de başarmışlardır [1]. Bu başarının oluşabilmesi için sporda performans takibi gereklidir. Özellikle çocuklarda gelişimin takip edilebilmesi için antropometrik ve motorik testler yapılmaktadır. Çocuklardan -sporcu olsun ya da olmasın- koşma, sıçrama, tırmanma yuvarlanma gibi becerileri etkin olarak gerçekleştirmeleri beklenir. Bu beceriler kuvvet, esneklik, dayanıklılık, sürat, koordinasyon gibi temel motorik özellikler ile gerçekleştirilmektedir. Sporcularda formal eğitimin etkisi ile bu beceriler daha çok gelişir. Sözü edilen özellikler tek parametreyi temsil ediyor gibi görünse de karmaşık bir yapı olan sportif performans kapsamında bunu değerlendirmek için bütüncül yaklaşım gereklidir. Bu durumda her bir özellik diğer özellikler ile ilintilidir.

Su ve buhar gücünün kullanılmasıyla başlayan endüstri devrimi, elektrik-elektroniğin makineyle buluşmasıyla, bilgisayarın hayatımıza girmesiyle ve Endüstri 4.0 kavramıyla birlikte de dijitalleşme kavramı ve bu dijitalleşmeyle yüksek teknolojilerin hayata dahil olmasıyla gelişim kazanmıştır [2].

Teknolojinin daha yaygın olduğu günümüzde bilgisayar biliminde yaşanan gelişmeler büyük boyutlu verilerin işlenmesini kolaylaştırmış ve bilgi çıkarımını hızlandırmıştır [1]. Bu çalışma kapsamında istatiksel öğrenme teorisine dayalı bir gözetimli öğrenme algoritması olan "Destek Vektör Makineleri", yaygın bir kullanım alanına sahip, karmaşık problemleri kısa zaman diliminde çözmeye yardımcı bir sistem olan Yapay Sinir Ağları ve ağaç yapısı şeklinde bir sınıflandırma algoritması olan Karar Ağaçları kullanılarak sınıflandırma işlemleri ve özellik çıkarımları yapılacaktır.

Yapay Sinir Ağlarının [3] tarihine bakıldığı zaman 1970 yılı dönüm noktası olmuştur. Öncesinde bilgisayarların öğrenebileceğini kimse kabul etmezken araştırmacıların yaptığı çalışmalar sonucunda Yapay Sinir Ağlarına olan ilgi artmış olup bu teknolojiden faydalanılmak istenilmiştir. Sinir Ağları desen tanıma, sınıflandırma ve tahmin için pek çok iş uygulamasında kullanılır ve sinir ağlarının insandan alınan ölçümlerin modellenmesinde kullanımı yaygınlaşmaktadır. Yapay Sinir Ağları, deneyimlerden öğrenen insan beynin sinir yapısına dayanan modellerdir.

Karar Ağacının anlaşılır olması kullanılma sebeplerindendir. Karar Ağacı tahmin ve sınıflandırma yapılarak kullanılan bir dizi ilgili seçimlerin olası sonuçlarının haritasıdır [4]. Karar Ağaçlarındaki ana fikir, veri kümesinin ögelerini gruplara ayırmaya dayanmaktadır. Amaç, grubun tüm unsurları aynı sınıf etiketine sahip olana kadar sürece devam etmektir. Veri kümesinden bir karar ağacı oluşturmak ve belirli durumlarda ne yapılacağını belirlemek için kullanılır [5].

Destek Vektör Makineleri yüz tanımadan ses analizi yapmaya kadar sınıflandırma problemlerinde kullanılan, istatistiksel öğrenmeye dayalı bir algoritmadır.

Veri setinde bulunan özellikler; Cinsiyet, Yaş, Vücut Ağırlığı, Boy Uzunluğu, Kulaç Uzunluğu, Oturma Yüksekliği, Kormik İndeks, Çeviklik (ZigZag), Sürat (10m, 20m), Denge (Filamingo Denge Testi), Esneklik (Otur Uzan, Sağ Omuz, Sol Omuz), Güç (Durarak Uzun Atlama), Kuvvet (Sağ Pençe Kuvveti, Sol Pençe Kuvveti, Mekik), Ay.

Bu yöntemler kullanılarak gerçekleştirilen işlemler sonucunda sportif verim açısından antrenöre verilerden çıkarım yapabilme konusunda kolaylık oluşturabilecek, zaman tasarrufu sağlayabilecek ve objektif yaklaşım konusunda destek olabilecek bir algoritma önerisi oluşturulabilir.

Tezin ikinci bölümünde literatürde yapılan çalışmaların yanında, Yapay Zeka, Makine Öğrenmesi, Karar Ağacı Algoritmaları, Destek Vektör Makinesi, Yapay Sinir Ağı ve Python kavramlarına yer verilmiştir. Tezin üçüncü bölümde kullanılan yöntemlerin neler olduğuna, dördüncü bölümde elde edilen bulgulara, son bölümde ise sonuç kısmına yer verilmiştir.

2. GENEL KISIMLAR

Literatürde yapılan çalışmalara bakıldığında, Güler ve diğ. [6] tarafından 8, 9, 10 yaşlarındaki erkek çocuklarının Mekik testi, Otur-Eriş testi gibi ölçümleri alınarak sağlık açısından fiziksel uygunluğu ve ulusallığı temsil edip edemeyeceği irdelenmiştir. Çamlıgüney [7] tarafından 8-10 yaş aralığındaki kız çocuklarının Filamingo Denge Testi, Durarak Uzun Atlama, Mekik gibi fiziksel aktivitelerle fiziksel uygunluklarının incelenerek beslenme ile ilişkisi ele alınmıştır. Sayın [8] tarafından 20m Sürat, Otur-Uzan, Mekik gibi fiziksel aktivitelerle 15-17 yaş aralığındaki gençlerin belirtilen aktivitelerle uygunlukları, Uçan ve diğ. [9] tarafından 7-10 yaş aralığındaki çocukların Mekik, Filamingo Denge Testi gibi fiziksel aktivitelerin spor yapan ve yapmayan çocuklar arasındaki ilişki incelenmiştir.

Veri Madenciliğinin sporla alakalı verilerle işlenmesi giderek yaygınlaşmaktadır. Karar Ağaçları, Destek Vektör Makinesi ve Yapay Sinir Ağları elde edilen verilerin işlenmesinde kullanılan yaygın algoritmalardır. Usta [1] tarafından fiziksel ve motorik ölçümleri yapılan kişilerin, uzman kişilerinde görüşlerine başvurularak yatkın oldukları spor dalının belirlenmesine yönelik bir sınıflama sistemi sunulmuştur. Stadudenmayer ve diğ. [10] tarafından yaygın olarak kullanılan tek eksenli bir ivmeölçerle toplanan fiziksel aktivite verilerine uygulanacak iki Yapay Sinir Ağı geliştirilip test edilmiştir**.**

Silve ve diğ. [11] tarafından Yapay Sinir Ağı ile yüzme ve çok çeşitli sporlarda performans modellemesi ve yetenek tanımlaması gibi karmaşık sorunların çözümünde bir yaklaşım ortaya konulmuştur. Ruiz ve diğ. [12] tarafından genç insanlarda 20m Sürat Mekik koşusu testi performansı, Cinsiyet, Yaş, Ağırlık ve Boydan maksimum oksijen alımını tahmin etmek için Yapay Sinir Ağı denklemi geliştirilmiştir. Woods ve diğ. [13] tarafından takım sporunda, oyun pozisyonunu bir oyuncunun ifade edilen beceri setlerine göre sınıflandırıp oyun pozisyonuna göre performans özelliklerinin tanınmasını sağlayarak yetenek tanımlaması yapılmıştır. Eskoifer ve diğ. [14] tarafından Destek Vektör Makinesi kullanılarak, işaretçiler tarafından nicelenen segment hareketinin tam mekânsal ve zamansal bilgisini kullanarak yürüyüş modelindeki grup farklılıklarını sınıflandırmak için bir yöntem ortaya koymuştur.

2.1. YAPAY ZEKA

Teknolojinin devrimiyle birlikte günlük hayatımızda karşılaştığımız sıkıntıların çözümüne yardımcı olacak sistemler ve cihazlar geliştirildi. Bu teknolojik devrimler toplumun, yaşamın ve firmaların üzerinde son derece önemli bir hâle geldi [15].

Teknolojik devrimlerin en büyük gelişimi Yapay Zeka ile oldu. Alan Turing bir makinenin akıllı olup olamayacağını belirlemek için "Turing Testi" adlı bir test geliştirdi [16]. Alan Turing'in bu testinden sonra "Yapay Zeka" ifadesi 1956 yılında "Dartmouth Konferansı'nda" alan adı olarak sunuldu [16]. **"**Yapay Zeka" terimi resmî olarak 1969 yılında ilk Uluslararası Bilgisayar Bilimi Ortak Konferansında Washington'da tanıtıldı [17]. Yapay Zekanın bugüne kadar olan zaman çizelgesine Tablo 2.1'de kısaca bakıldığında Yapay Zeka teknolojisinin nerelere geldiği hakkında bir fikir sahibi olunmasına yardımcı olacaktır.

Tablo 2.1: Yapay Zeka Devrim Çizelgesi [15].

1990	Sinir Ağı cihazı el yazısıyla yazılan rakamları okudu.
1993	Robot Polly görüntü kullanarak hareket etti.
1997	Deep Blue, dünya satranç şampiyonunu yendi.
1998	Robot oyuncak Furby nasıl konuşulacağını öğrendi.
2005	Robot Asimo restoran müşterilerine hizmet verdi.
2009	Google'ın ilk kendi kendine giden arabası.
2011	Watson bilgisayar Jeopardy'i yendi.
2016	AlphaGo, GO şampiyonlarını yendi.

Tablo 2.1'e bakıldığında Sinir Ağı Cihazının El Yazısıyla yazılan rakamları okumasıyla başlayan Yapay Zeka gelişimi, Robot Asimo'nun restoran müşterilerine hizmet vermesine, Alpha Go'nun Go şampiyonlarını yenmesine kadar geldi.

Bugün Yapay Zeka, bilgisayar bilimi, hastaneler, telekomünikasyon, ulaşım, spor, müzik gibi pek çok alanda kullanılmaktadır.

2.2. MAKİNE ÖĞRENMESİ

Su ve buhar gücünün kullanılmasıyla başlayan endüstri devrimi, elektrik-elektroniğin makineyle buluşmasıyla, bilgisayarın gündelik hayata girmesiyle ve Endüstri 4.0 kavramıyla birlikte de dijitalleşme kavramı ve bu dijitalleşmeyle yüksek teknolojilerin hayatımıza girmesiyle gelişim kazandı [2]. Büyük veri denilen kavramın toplumsal hayatta yer alması, bu büyük verinin anlamlandırılıp işlenmesi durumunu da ön plana çıkardı. Makine Öğrenmesi algoritmalarının kullanılmasıyla bu büyük verilerin işlenerek sonuçlar üretilmesine başlandı. Eğitim verileriyle makinenin öğrenmesi sağlanıp test verileriyle de sonuçlar elde edildi. Makine Öğrenmesiyle ilgilide birbirinden farklı tanımlamalar ortaya çıktı.

Makine Öğrenmesi, verilerden otomatik modeller oluşturmak için algoritmalar veya yöntemler içeren yapay zekanın bir dalıdır [18]. Makine Öğrenmesi, insanların tasarlayamayacakları kadar karmaşık görevleri yerine getirmeye olanak sağlar. Başka bir tanım ise Makine Öğrenimini "bilgisayarları insanlar gibi öğrenip harekete geçirme bilimi ve veri ve bilgileri gözlemler ve gerçek dünya etkileşimleri şeklinde besleyerek zaman içinde öğrenimlerini geliştirir." olarak tanımlar [19]. Yapay Zeka, Makine Öğrenmesi, Veri Madenciliği, büyük veri gibi kavramların birbirleriyle ilişkisi Şekil 2.1 'de gösterilmiştir.



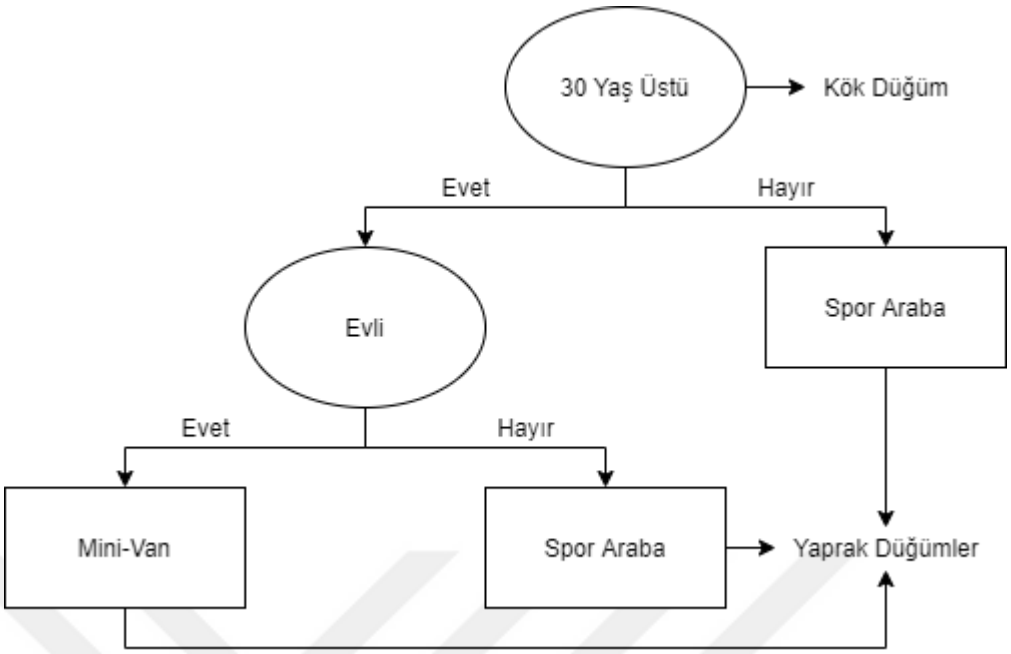
Şekil 2.1: Veri Bilimindeki Kavramlar [20].

Bu tezde Makine Öğrenmesi algoritmalarından Karar Ağacı Algoritmalarının, Yapay Sinir Ağlarının ve Destek Vektör Makinelerinin 7-14 yaş aralığında çocuklardan antropometrik ve motorik özelliklerinin alınmasıyla oluşan veri seti üzerinde çalıştırılmasıyla elde edilen sonuçları değerlendirilmiştir.

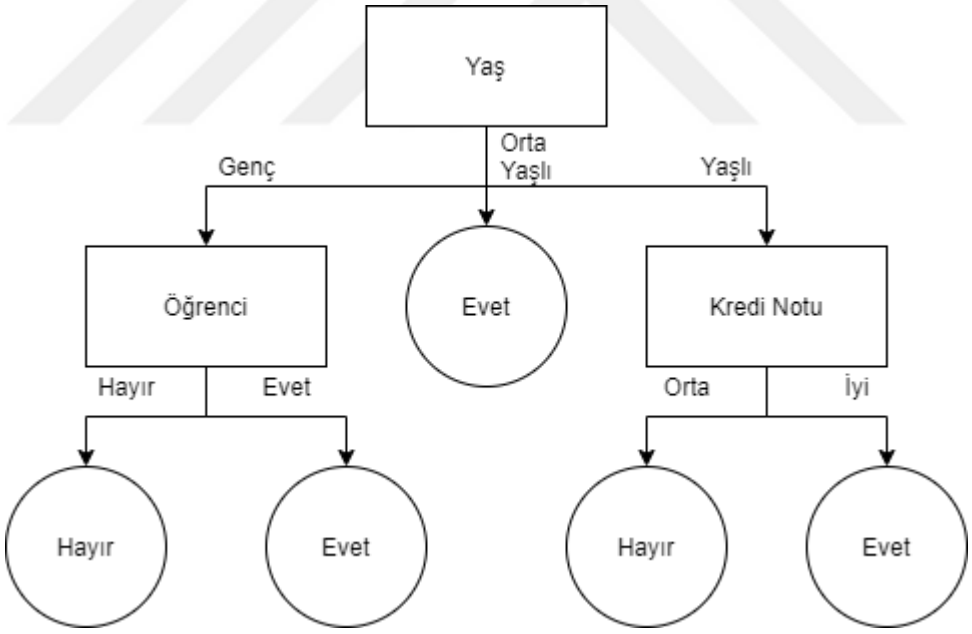
2.3. KARAR AĞACI ALGORİTMALARI

Karar Ağacı yöntemi, sınıflandırma ve tahmin için daha çok kullanılan önemli Veri Madenciliği yöntemleri arasında yer almaktadır. Sınıflandırmada Karar Ağaçlarının kullanılmasının temel sebebi basit bir yapıya, anlaşılır olmasına sahip olmasıdır. Bir diğer avantajı da diğer algoritmalara göre oluşturulma süresinin hızlı olmasıdır. Karar Ağaçlarında eğitim verileri ile ağaçlar oluşturulup eğitime göre oluşturulan bu ağaçta daha sonra test verilerinin kullanılarak değerlendirilmesi, gelecekte yapılacak çalışmalar için elde edilen bu değerlerle doğru tahminlerin yapılması işlenmektedir [21] [22].

İki Dallı ve Çok Dallı Karar Ağaçları Şekil 2.2 ve Şekil 2.3'te gösterilmiştir.



Şekil 2.2: İki Dallı Karar Ağacı [23].



Şekil 2.3: Çok Dallı Karar Ağacı [23].

Decision Stump Algoritması, tek seviyeli bir Karar Ağacı oluşturur. Oluşturulan tek seviyeli bu ağaçta kök düğüm, yaprak düğümlere direkt olarak bağlıdır. Decision Stump Algoritması, sınıflandırma işlemini direkt olarak tek girdi düğüme göre gerçekleştirir [4] [24].

Decision Stump algoritması büyük veri için nüfus segmentasyonunda ve basit evet/hayır modellerinde karar vermeye yardımcı olmak için daha küçük veri setlerinde kullanılırlar [25].

Hoeffding Tree algoritması [26], Karar Ağacının yapımı ve analizi için Hoeffding sınırını kullanır. Belli bir güven seviyesine ulaşmak için çalıştırılacak örnek sayısına karar vermek için kullanılan bir sınırdır. Bir Hoeffding ağacı, dağıtım üreten örneklerin zaman içinde değişmediği varsayılarak büyük veri akışlarından öğrenebilir. Hoeffding Tree algoritması nitelikleri diğer algoritmalardan daha iyi karşılaştırır. Ayrıca bellek tüketimi daha azdır ve verilerin örneklenmesi ile gelişmiş kullanım sağlar ancak bağların olup olmadığını denetlemek için çok zaman harcayan bir algoritmadır.

LMT (Logitic Model Trees) algoritması, karar ağacı öğrenme yöntemlerini ve lojistik regresyonu birleştiren bir sınıflandırma modelidir. LMT, eğitim verilerinin aşırı sığmasını önlemek için bir dizi LogitBoost yinelemesini bulmak için çapraz doğrulama kullanır [27]. Budama işlemleri gerçekleştirilerek sadeleştirme yapılır ve böylelikle performans artırımı sağlanır. [4]

Random Forest Algoritması [28], eğitim verisindeki örneklerin rastgele seçilmesi ile oluşturulan ve hem sınıflandırma hem de regresyon problemleri için model toplama fikirlerine dayanan popüler ve çok verimli bir algoritmadır. Rastgele Ormanlar, Bayes sınıflandırıcısının tahminlerini sağlayan, sınıflandırma hatasını en aza indiren haritalama veya regresyon işlevini sağlayan bir model oluşturma stratejisidir. Random Forest algoritması matematiksel bir bakış açısından açıkça aydınlatılmasa bile birçok farklı uygulamada çok güçlü görünmektedir.

Random Tree Algoritması [29] denetimli bir sınıflandırıcıdır. Rastgele Ağaçlar, orman olarak adlandırılan ağaç yordayıcılarının bir koleksiyonudur. Sınıflandırma şu şekilde çalışır: Rastgele Ağaçlar sınıflandırıcısı girdi özelliği vektörünü alır, ormandaki her ağaçla sınıflandırır ve "oyların" çoğunluğunu alan sınıf etiketini çıkarır. Bir gerileme durumunda, sınıflandırıcı yanıtı ormandaki tüm ağaçlara verilen yanıtların ortalamasıdır. Rastgele Ağaçlar temelde Makine Öğreniminde mevcut iki algoritmanın birleşimidir, tek model ağaçlar Rastgele Orman fikirleriyle birleştirilir.

REPTree algoritması [29], regresyon ağacı mantığını kullanır ve farklı yinelemelerde birden çok ağaç oluşturur. Bundan sonra üretilen tüm ağaçlardan en iyisini seçer. Ağacı budarken kullanılan ölçü, ağaç tarafından yapılan tahminlerdeki ortalama kare hatasıdır.

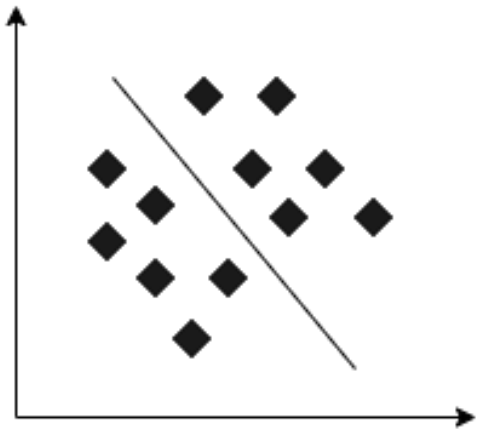
J48 algoritması [30], yukarıdan aşağıya bir yaklaşım izlemektedir yani mevcut bir veri setini parçalara bölerken ilk olarak üst seviyeleri daha sonra alt seviyeleri incelemektedir.

J48 algoritmasının temel adımları şu şekildedir [30]:

- ✓ Örneklerin aynı sınıfa ait olması durumunda ağaç bir yaprağı temsil eder, böylece yaprak aynı sınıfla etiketlenerek döndürülür.
- ✓ Potansiyel bilgiler, özellik üzerindeki bir test tarafından verilen her özellik için hesaplanır. Daha sonra, özellik üzerindeki bir testten kaynaklanacak bilgi kazancı hesaplanır.
- ✓ Daha sonra en iyi özellik, mevcut seçim kriteri ve dallanma için seçilen özellik temelinde bulunur.

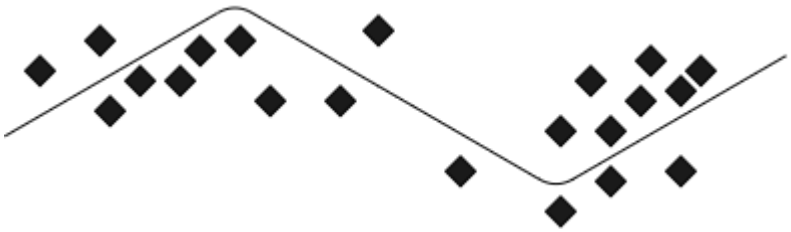
2.4.DESTEK VEKTÖR MAKİNESİ

Destek Vektör Makinesi, Denetimli Öğrenme algoritmalarından birisidir. Destek Vektör Makinesi 1960'lı yılların sonuna doğru Vladimir Vapnik ve Alexey Chervonenkis tarafından temelleri atılmıştır [31]. Bir Destek Vektör Makinesi, soyut bir alanda belirli bir sınıfın üyeleri ile üye olmayanlar arasında optimum bir ayırma köprüsü bulur. Destek Vektör Makinelerinde, n bağımsız değişken sayısı olarak belirlendiğinde, veri noktaları koordinatlarından her biri n-boyutlu uzayda değişken değerlere sahip olarak belirlenebilmektedir. İki sınıfı birbirinden ayıran Hiper Düzlem bulunarak sınıflama işlemi yapılır [32]. Doğrusal bir Hiper Düzlem örneği Şekil 2.4'te gösterilmiştir.



Şekil 2.4: Doğrusal Hiper Düzlem.

Hiper Düzlem noktalar arasındaki uzaklıkları maksimize etme yoluyla belirlenir. Bütün veriler doğrusal bir Hiper Düzlem olarak ayrılmamaktadır. Şekil 2.5'te doğrusal olmayan Hiper Düzlem örneği bulunmaktadır.



Şekil 2.5: Doğrusal Olmayan Hiper Düzlem.

Destek Vektör Makinesinin avantajları [33]**:**

- ✓ Yüksek Boyutlu alanlarda etkilidir.
- ✓ Boyut sayısının numune sayısından fazla olduğu durumlarda hâlâ etkilidir.
- ✓ Karar işlevinde eğitim noktalarının bir alt kümesini kullanır, bu nedenle bellek verimlidir.
- ✓ Karar işlevi için farklı çekirdek işlevleri belirtilebilir. Ortak çekirdekler sağlanır ancak özel çekirdekler belirtmek de mümkündür.

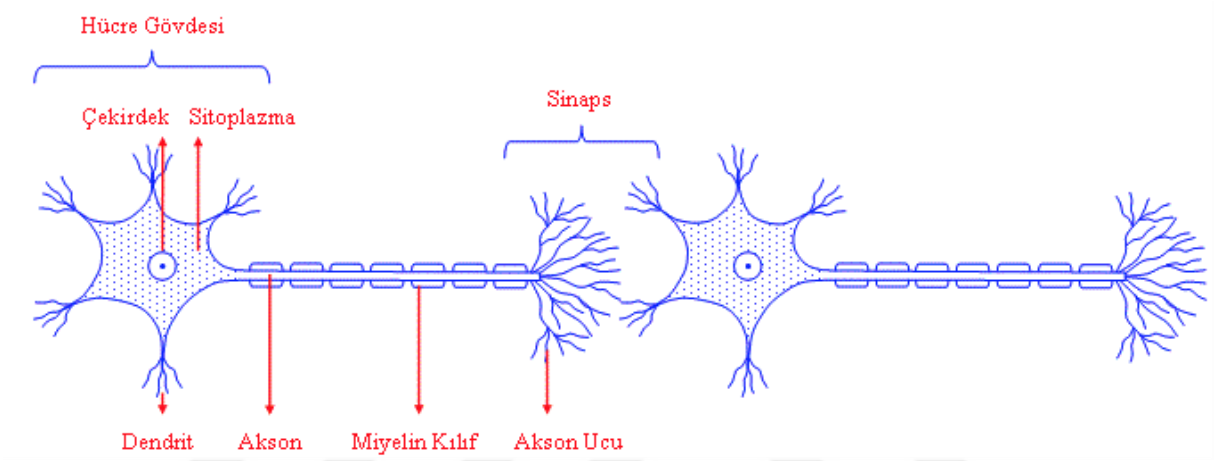
2.5.YAPAY SİNİR AĞLARI

Yapay Sinir Ağları [34], insan beyninin temel özelliği olan öğrenme işlevini yerine getiren bilgisayar sistemleridir. Yapay Sinir Ağları öğrenme sürecini örneklerle yapar. Yapay Sinir Ağlarının geliştirilmesi diğer sistemlere göre daha kolaydır.

Yapay Sinir Ağı [35] temel olarak, her biri bir dizi gerçek değerli aktivasyon üreten nöron adı verilen bir dizi basit birbirine bağlı işlemciden oluşan bir sistemdir. Bu tür yapay nöronlar, gerçek biyolojik nöronların çalışma modunu çoğaltmak için tasarlandı, böylece giriş nöronları çevreyi algılayan sensörler aracılığıyla aktive olur. Yapay bir nöron için, böyle bir ortam n-m-m görüntü matrisi veya sıkıştırılmış ses içeren bir matris olabilir. Diğer ara nöronlar, önceden bağlanmış nöronlardan giriş nöronları yoluyla belirlenmiş bir çıkış nöronlarına ağırlıklı bağlantılar yoluyla aktive olur.

Bu tür sistemler, tüm nöronların ve aralarındaki ağırlıklı bağlantıların, tüm ağın çok sayıda görüntü arasında bir görüntünün belirli bir özelliğini tanımlaması gibi istenen bir davranışı ifade edebilmesi için değerlerini ayarlayacağı şekilde tasarlanmıştır. Bu nedenle Yapay Sinir Ağları, sağlanan görüntülerden özellikleri çıkarma kabiliyetine sahip yapay nöron katmanları arasında bağlantılar kurarak nesne algısını taklit etmiştir [36].

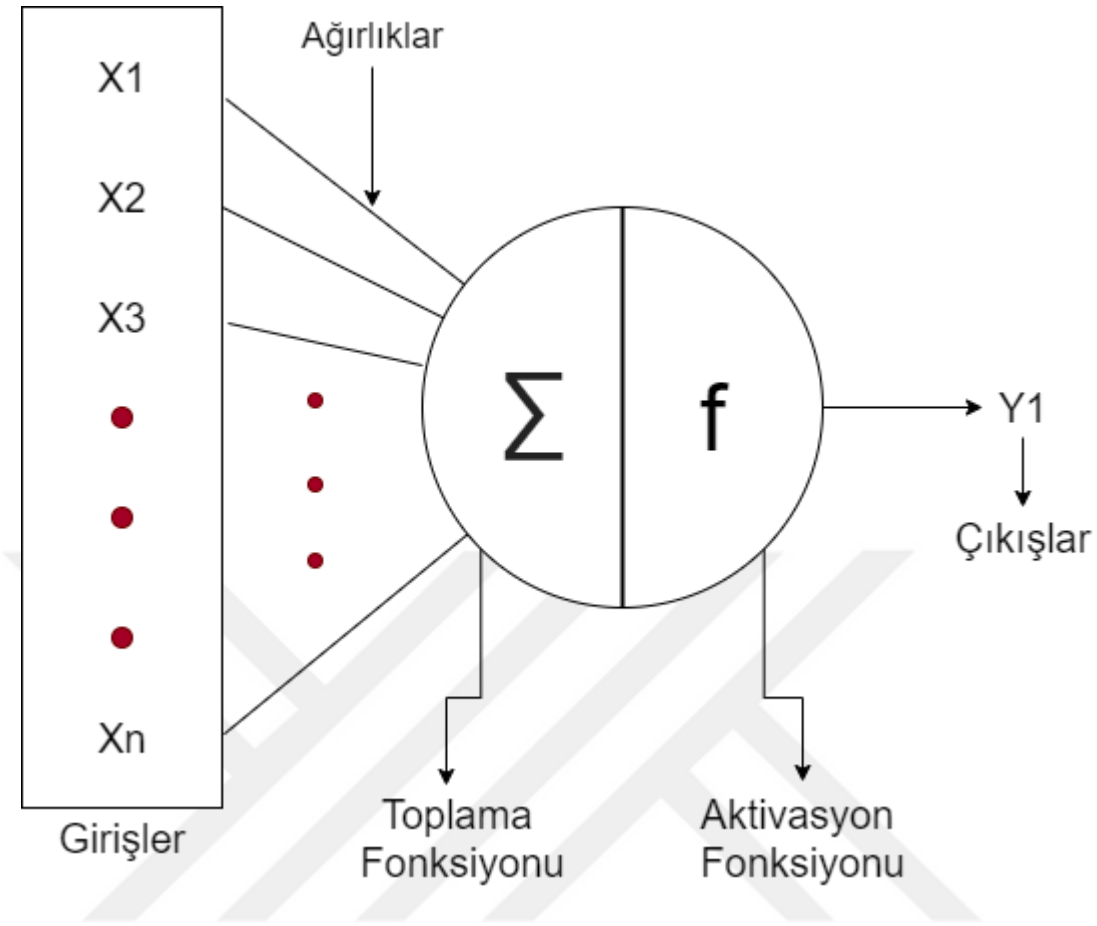
İnsandaki bir sinir hücresinin yapısı aşağıdaki Şekil 2.6 ‘da gösterilmiştir.



Şekil 2.6: İnsandaki Sinir Hücresi Yapısı [37].

Biyolojik bir nöronda gövde, gelen sinyalleri toplar ve işler. Dentrit, diğer nöronlardan gelen sinyalleri gövdeye taşır. Akson, nöronun çıkış mekanizmasıdır ve girdi sinyallerini çıktı sinyallere çevirir. Sinaps ise Akson ve Dentritler arasındaki bağlantı boşluklarıdır ve nöronlar arasındaki elektrokimyasal teması sağlar. Bir nöronun çalışmasına bakıldığında, Sinapslar aracılığı ile sinyaller hücrenin Dentritlerine gelir, tüm Dentritlerdeki sinyaller hücre gövdesinde toplanır ve toplam değer bir eşik değerinden yüksekse aksonlar aracılığıyla diğer nöronlara uyarı sinyali gönderir.

Yapay bir nöronun yapısı ise Şekil 2.7’de gösterilmiştir.



Şekil 2.7: Yapay Nöron [37].

Yapay bir sinir hücresinin girişleri dış ortamdan gelebileceği gibi diğer hücrelerden de gelebilir. Burada yer alan ağırlıklar yapay hücreye gelen girişin önemini ve etkisini gösterir. Toplam fonksiyonu ise net girdi hesabını yapar. Bu işlemi yaparken x girdileriyle w Sinaptik ağırlıklarını çarpıp toplayarak gerçekleştirir. Yapılan işlem aşağıdaki gibidir [37]:

Toplama Fonksiyonu = $\sum_{i=1}^{n} \boldsymbol{X_i} * \boldsymbol{W_i}$ **(2.1)**

Aktivasyon fonksiyonu ise toplama fonksiyonundan elde edilen girdiye karşılık bir çıktı meydana getirir. Aktivasyon fonksiyonu olarak kullanılacak pek çok fonksiyon mevcuttur. Bunlardan bazıları; Sinüs Fonksiyonu, Eşik Değer Fonksiyonu, Sigmoid Fonksiyonu.

2.6. PYTHON

Her programlama dilinin kendine ait özellikleri vardır. Python da bu programlama dillerinden birisidir. Makine Öğrenmesi gibi konularda son yıllarda Python sıklıkla kullanılmaya başlanmıştır. Basit bir faktöriyel hesabı yapan Python kod örneği aşağıdaki gibidir:

Yukarıda basit bir şekilde def ile fonksiyon tanımı yapıldı ve döngü içerisinde verilen sayıya kadar sayıların çarpımını hesaplayıp döngü bitince bu faktöriyelin sonucunu ekrana yazdıran örnek bir kod çalıştırıldı. Bu kodun sonucu print ifadesi ile ekrana yazdırıldı.

7 != 5040

Bu tez içinde Python programlama dili üzerinde korelasyon işlemi ve Yapay Sinir Ağı modeli kullanıldı.

2.7. ÖZELLİKLER

6-14 yaş aralığında çocuklardan alınan antropometrik veriler ve motorik performans verileri bu tez çalışmasında kullanılan sınıflandırma algoritmalarında "özellik" olarak yer almaktadır. Bu çalışmadaki veriler çeşitli amatör ve profesyonel gruplarda ölçümler gerçekleştiren bir performans ölçüm ekibinden sağlanmıştır. Veri setinde bulunan özellikler; Cinsiyet, Yaş, Vücut Ağırlığı, Boy Uzunluğu, Kulaç Uzunluğu, Oturma Yüksekliği, Kormik İndeks, Çeviklik (ZigZag), Sürat (10m, 20m), Denge (Filamingo Denge Testi), Esneklik (Otur Uzan, Sağ Omuz, Sol Omuz), Güç (Durarak Uzun Atlama), Kuvvet (Sağ Pençe Kuvveti, Sol Pençe Kuvveti, Mekik), Ay. Tablo 2.2 ve Tablo 2.3'de özelliklerle ilgili bilgilere yer verilmiştir.

Tablo 2.2: Sağ Omuz Esnekliği, Sol Omuz Esnekliği ve Cinsiyetin Veri Setindeki İfadeleri.

Sağ Omuz Esnekliği	E (EVET)	H (HAYIR)
Sol Omuz Esnekliği	E (EVET)	H (HAYIR)
Cinsiyet	ERKEK	KIZ

Tablo 2.3: Özelliklerin Değerleri.

Özellikler	Minimum	Maksimum	Ortalama	Standart Sapma
Yaş	7	14	10,45	1,82
Vücut Ağırlığı (kg)	17,3	104	38,89	11,85
Boy Uzunluğu (cm)	112	190	141,71	11,71
Kulaç Uzunluğu (cm)	105	192	140,58	12,71
Oturma Yüksekliği (cm)	51	99	75,25	6,09
Kormik İndeks	31,93	77,34	53,13	1,67
Çeviklik (ZigZag) (sn)	5,76	15	8,16	0,92
10m Sürat (sn)	1,7	3,66	2,30	0,19
20m Sürat (sn)	3,01	6,80	4,20	0,39
Filamingo Denge Testi (Çıkış Sayısı)	1	15	10,41	4,83
Otur Uzan Esneklik Testi (cm)	-23	30	2,05	7,70
Durarak Uzun Atlama (cm)	30	229	125,33	23,47
Sağ Pençe Kuvveti (kg)	5	49,1	17,49	6,13
Sol Pençe Kuvveti (kg)	5	48,5	16,90	5,81
Mekik (Adet)	0	32	13,77	5,25

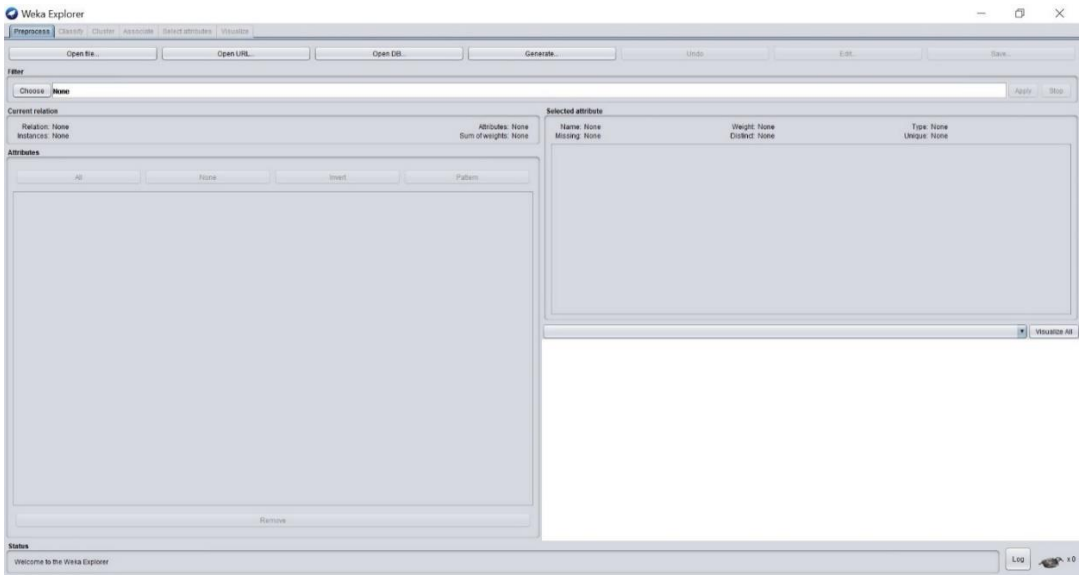
3. MALZEME VE YÖNTEM

Weka**,** Waikato Environment for Knowledge Analysis kelimelerinin baş harflerinden oluşan ve Veri Madenciliği konularında çokça kullanılan paket programlardan birisidir. Açık kaynak kodlu olup Java dili üzerine geliştirilmiştir [38]. Weka programının açılış ekranı Şekil 3.1'de gösterilmiştir.



Şekil 3.1: Weka Açılış Ekranı.

Sınıflandırma, kümeleme gibi işlemleri yapabilmek için Explorer seçeneği üzerinden Weka programı açılır. Explorer seçeneğine tıkladıktan sonraki Weka programının ekranı Şekil 3.2'de olduğu gibi açılacaktır.



Şekil 3.2: Weka Explorer Penceresi.

Açılan bu ekrandan open file seçeneği ile Weka'da çalıştırılması istenilen dosya açılacaktır. Açılacak dosyanın uzantısı .arff uzantılı olmalıdır. Arff uzantılı dosyanın içeriği de Şekil 3.3'de gösterildiği gibi olmalıdır.

```
@relation Cinsiyet_Label

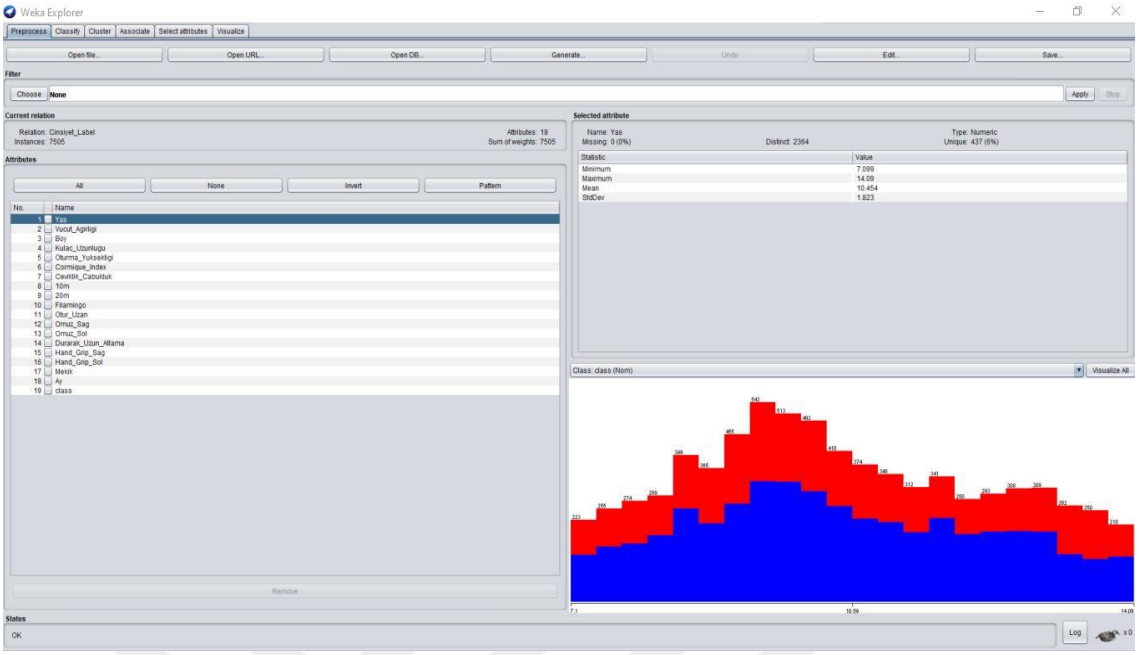
@attribute Yas numeric
@attribute Vucut_Agirligi numeric
@attribute Boy numeric
@attribute Kulac_Uzunlugu numeric
@attribute Oturma_Yuksekligi numeric
@attribute Cormique_Index numeric
@attribute Ceviklik_Cabukluk numeric
@attribute 10m numeric
@attribute 20m numeric
@attribute Filamingo numeric
@attribute Otur_Uzan numeric
@attribute Omuz_Sag {E,H}
@attribute Omuz_Sol {E,H}
@attribute Durarak_Uzun_Atlama numeric
@attribute Hand_Grip_Sag numeric
@attribute Hand_Grip_Sol numeric
@attribute Mekik numeric
@attribute Ay numeric
@attribute class {ERKEK,KIZ}

@data
7.465753425,17.3,113,105,61,53.98,9.093,2.683,5.016,3,-9,H,H,85,8,6.3,5,89,ERKEK
7.756164384,17.5,119,117,62.5,52.52,9.386,2.547,4.763,5,2,H,H,86,8.1,6.5,17,93,KIZ
8.208219178,17.6,112,109,61,54.46,9.483,2.608,4.743,4,12,E,E,103,7.6,6.1,1,98,KIZ
7.706849315,18,114,110,62,54.39,8.7,2.687,5.29,15,11,H,H,111.5,10.3,8.1,12,92,KIZ
7.515068493,18.9,116.5,112,63,54.08,8.988,2.506,4.682,11,12,H,E,100,8.4,9.5,11,90,ERKEK
7.868493151,19,115,111,60,52.17,7.646,2.408,4.452,15,2,E,E,95,8.4,8.9,15,94,KIZ
8.460273973,19,117,113,62,52.99,8.58,2.565,4.899,5,7,H,H,93,11.8,8.6,14,101,ERKEK
7.893150685,19.1,121.5,118,63,51.85,9.569,2.518,4.791,15,6,E,E,125,6.7,7.1,11,94,KIZ
```

Şekil 3.3: Arff Uzantılı Dosya.

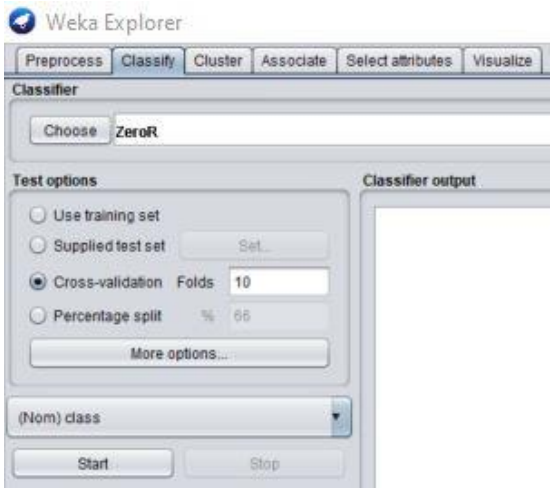
Şekil 3.3'de görüldüğü gibi ilk başta @relation ile elde ettiğimiz veri setinin başlığı yazılacaktır. Ardından @attribute ile verilerin özellikleri ve o özelliğin veri tipi belirtilecektir. Sayısal veriyse özelliğin yanına numeric ifadesi, sözel bir veriyse özelliğin yanına {} parantez içerisine sözel olan ifadeleri virgüllerle ayırarak yazılmalıdır. En son veriyi ifade etmek için @data ifadesi yazılacaktır. Verileri excel dosyasından aldıktan sonra ;(noktalı virgül) olarak ayrılan verileri arff uzantısı olarak Weka'da çalıştırmak için ,(virgül) ile ayırmak gerekmektedir. Notepad++ ile açılan arff uzantılı dosyada ;(noktalı virgül) olarak ayrılan yerileri ,(virgül)'e çevirerek bu işlem gerçekleştirilir.

Weka'da arff uzantılı dosya açıldığında Şekil 3.4'de olduğu gibi bir ekran oluşacaktır.



Şekil 3.4: Dosya Yüklendikten Sonra Oluşan Ekran.

Oluşan ekranda @attribute ile belirtilen özellikler, sağ tarafta select attribute kısmında soldaki özelliklerden birine tıklandığında ekrana gelen verilerin minimum, maximum ve ortalama değerleri vardır. Sağ alt tarafta ise sınıf olarak belirtilen özelliğin diğer özelliklerdeki dağılımını göstermektedir. Açılan ekranın üst menü bölümünde veriyi çalıştırmak için sınıflandırma algoritmaları veya yapılmak istenirse kümeleme işlemi yapılmaktadır. Classify seçeneği seçildikten sonra Şekil 3.5'teki gibi bir ekran gelmektedir.



Şekil 3.5: Classify Penceresine Tıklanıldığında Açılan Ekran.

Choose seçeneği ile birlikte sınıflandırma algoritmalarından biri seçilerek start düğmesine basılır ve algoritma çalıştırılarak elde edilen sonuçlar ekranda gösterilir.

Algoritmaları çalıştırdıktan sonra elde edilen sonuçlardan birisi Karmaşıklık Matrisidir. Karmaşıklık Matrisinde yer alan değerler TP, TN, FP, FN olarak yer almaktadır [32].

Toplam Pozitifler (TP) : Gerçek durum doğru olduğunda kestirilen durumun doğru olmasıdır.

Toplam Negatifler (TN) : Gerçek durum doğru olduğunda kestirilen durumun yanlış olmasıdır.

Hatalı Pozitifler (FP) : Gerçek durum yanlış olduğunda kestirilen durumun doğru olmasıdır.

Hatalı Negatifler (FN) : Gerçek durum yanlış olduğunda kestirilen durumunda yanlış olmasıdır.

Buradaki durumlardan TP ve FN ne kadar yüksek olursa doğruluk değeri de bir o kadar yüksek olacaktır.

Karmaşıklık Matrisi dışında diğer performans ölçütleri ve formülleri aşağıdaki gibidir [32].

Bir sınıflandırma algoritmasının ne kadar doğru çalıştığı ve netliği Doğruluk (Accuracy) ile hesaplanır:

Hata Oranı: (FP+FN)/(P+N) **(3.1)**

Doğruluk: (TP+TN)/(P+N) veya 1-Hata Oranı **(3.2)**

Hata oranı, 1 sayısından doğruluk oranı çıkarılarak elde edilebileceği gibi yukarıdaki formülde de yer aldığı üzere yanlış olan durumların tüm durumlara oranı ile de elde edilebilir.

Pozitif olarak elde edilen durumun test sonucunun da pozitif çıkması durumuna Duyarlılık ile bakılır.

Duyarlılık: TP/(TP+FP) **(3.3)**

Test sonucu pozitif elde edildiğinde gerçek durumunda pozitif olmasına Kesinlik ile bakılır.

Kesinlik: TP/(TP+FN) **(3.4)**

F Ölçütü Kesinlik ve Duyarlılık durumlarının harmonik ortalaması ile elde edilen başarıyı ölçmek için kullanılan bir kriterdir.

F-Ölçütü:$(1+\beta^2)$*Kesinlik*Duyarlılık/β^2(Kesinlik+Duyarlılık) **(3.5)**

Buradaki β değerleri 0,5 ,1 veya 2 olarak alınır. Beta değerinin büyümesiyle de kesinliğin etkisi daha da çok olacaktır.

Olasılık eğrisi olarak bilinen Roc Eğrisi, sınıflama modelinin farklı sınıflama eşiklerindeki performansını gösterir. Gerçek pozitif ve yanlış pozitif olmak üzere iki parametresi vardır:

Gerçek Pozitif Oranı: TP/(TP+FN) **(3.6)**

Yanlış Pozitif Oranı: FP/(FP+TN) **(3.7)**

Roc Eğrisi, Yanlış Pozitif Oranı x eksenini, Gerçek Pozitif Oranı y eksenini belirterek çizilen eğridir.

7505 verinin incelenen özelliklerinden hangilerinin birbiriyle daha yakından ilişkisi olduğuna Python kodu yazılarak korelasyonuna bakıldı. Python'da bulunan pandas kütüphanesi kullanılarak csv dosyası yüklendi. Numpy, Seaborn kütüphanelerinin kullanılmasıyla korelasyon işlemi yapıldı. Korelasyon sonucunu daha iyi görebilmek için çizim kütüphanesi olan matplotlib kullanıldı.

Python'da yapılan korelasyon işlemi aşağıdaki gibidir [39]:

Korelasyon:

```
import color as color
import optional as optional
import pandas as pd
from sklearn.datasets import load_boston
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

veriler = pd.read_csv("verisetim.csv")
cor_mat = veriler.corr()
mask = np.zeros_like(cor_mat)
mask[np.triu_indices_from(mask)] = True
print (mask)
```

```
sns.heatmap(cor_mat,
   xticklabels=cor_mat.columns.values,
   yticklabels=cor_mat.columns.values,
   cbar=True,
   linewidths=0.00009,
   annot=True, fmt=".2g",
   mask=mask)
plt.show()
```

Yapay Sinir Ağı işlemlerinin yapılıp sonuç alınması için Python'da Yapay Sinir Ağı kodu yazılarak sonuçlar elde edildi. Yapay Sinir Ağı işlemini yapabilmek için Python'da yer alan keras, sklearn, pandas kütüphaneleri kullanıldı.

Yapay Sinir Ağı Modeli [40]:

```
from sklearn.preprocessing import LabelEncoder
from sklearn import preprocessing
from sklearn.model_selection import train_test_split
from keras.models import Sequential
from keras.layers import Dense
from keras import metrics
from sklearn.metrics import confusion_matrix
import pandas as pd

df=pd.read_csv("verisetim.csv")
dataset=df.values
X=dataset[:,0:18]
Y=dataset[:,18]
print(Y)

le=LabelEncoder()
X[:,0]=le.fit_transform((X[:,0]))

le2=LabelEncoder()
X[:,11]=le2.fit_transform(X[:,11])

le3=LabelEncoder()
X[:,12]=le3.fit_transform(X[:,12])

x_train,x_test,y_train,y_test=train_test_split(X,Y,test_size=0.33,random_state=0)

sc=preprocessing.MinMaxScaler()
X_train=sc.fit_transform(x_train)
X_test=sc.fit_transform(x_test)

classifier=Sequential()
classifier.add(Dense(12,kernel_initializer='uniform',activation='relu',input_dim=18))
```

```
classifier.add(Dense(12,kernel_initializer='uniform',activation='relu'))
classifier.add(Dense(1,kernel_initializer='uniform',activation='sigmoid'))
classifier.compile(optimizer='Adam',loss='binary_crossentropy',metrics=['accuracy'])
classifier.fit(X_train,y_train,epochs=50)
```

4. BULGULAR

Diğer özellikler verildiğinde Sağ Omuz Esnekliği ve Sol Omuz Esnekliği de sınıf olarak belirtilmesi ile karar ağaçlarında çalıştırılması sonucu elde edilen doğruluk oranı sonuçları Tablo 4.1'de gösterilmiştir.

Tablo 4.1: Sağ Omuz Esnekliği ve Sol Omuz Esnekliği İçin Karar Ağaçlarıyla Elde Edilen Sonuçlar.

Karar Ağaçları	Sağ Omuz Esnekliği	Sol Omuz Esnekliği
J48	%83,51	%82,75
Decision Stump	%84,53	%84,53
Hoefding Tree	%84,17	%84,50
REPTree	%84,05	%83,85
Random Tree	%78,77	%77,02
Random Forest	%84,90	%84,54
Lmt	%84,63	%84,49

Tablo 4.1'e bakıldığında Sağ Omuz Esnekliği ve Sol Omuz Esnekliğinin %85 oranında doğru sonuç verdiği gözükmektedir. Karar Ağaçlarıyla elde edilen doğruluk oranlarının yanında elde edilen Karmaşıklık Matrisi ve bu matrise göre elde edilen oranların Sağ Omuz Esnekliği için J48 algoritmasıyla elde edilen sonuçları Tablo 4.2 ve Tablo 4.3'te gösterilmiştir.

Tablo 4.2: J48 Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi.

		Gerçek Durum	
		Doğru	Yanlış
Kestirilen	Doğru	3760	893
	Yanlış	344	2508

Tablo 4.3: J48 Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Oranlar.

	TP Rate	FP Rate	Kesinlik (Precison)	Duyarlılık (Recall)	F-Ölçütü	ROC Area
J48 Algoritması	0,83	0,14	0,84	0,83	0,83	0,87

Sağ Omuz Esnekliği için Decision Stump algoritmasıyla elde edilen sonuçlar Tablo 4.4 ve Tablo 4.5'te gösterilmiştir.

Tablo 4.4: Decision Stump Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi.

		Gerçek Durum	
		Doğru	Yanlış
Kestirilen	Doğru	3592	1061
	Yanlış	100	2752

Tablo 4.5: Decision Stump Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Oranlar.

	TP Rate	FP Rate	Kesinlik (Precison)	Duyarlılık (Recall)	F-Ölçütü	ROC Area
Decision Stump Algoritması	0,84	0,10	0,87	0,84	0,84	0,86

Sağ Omuz Esnekliği için Hoeffding Tree algoritmasıyla elde edilen sonuçlar Tablo 4.6 ve Tablo 4.7'de gösterilmiştir.

Tablo 4.6: Hoeffding Tree Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi.

		Gerçek Durum	
		Doğru	Yanlış
Kestirilen	Doğru	3634	1019
	Yanlış	169	2683

Tablo 4.7: Hoeffding Tree Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Oranlar.

	TP Rate	FP Rate	Kesinlik (Precison)	Duyarlılık (Recall)	F-Ölçütü	ROC Area
Hoeffding Tree Algoritması	0,84	0,12	0,86	0,84	0,84	0,87

Sağ Omuz Esnekliği için LMT algoritmasıyla elde edilen sonuçlar Tablo 4.8 ve Tablo 4.9'da gösterilmiştir.

Tablo 4.8: LMT Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi.

		Gerçek Durum	
		Doğru	Yanlış
Kestirilen	Doğru	3646	1007
	Yanlış	146	2706

Tablo 4.9: LMT Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Oranlar.

	TP Rate	FP Rate	Kesinlik (Precison)	Duyarlılık (Recall)	F-Ölçütü	ROC Area
LMT Algoritması	0,84	0,11	0,87	0,84	0,84	0,90

Sağ Omuz Esnekliği için Random Forest algoritmasıyla elde edilen sonuçlar Tablo 4.10 ve Tablo 4.11'de gösterilmiştir.

Tablo 4.10: Random Forest Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi.

		Gerçek Durum	
		Doğru	Yanlış
Kestirilen	Doğru	3777	876
	Yanlış	257	2595

Tablo 4.11: Random Forest Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Oranlar.

	TP Rate	FP Rate	Kesinlik (Precison)	Duyarlılık (Recall)	F-Ölçütü	ROC Area
Random Forest Algoritması	0,84	0,12	0,86	0,84	0,85	0,91

Sağ Omuz Esnekliği için Random Tree algoritmasıyla elde edilen sonuçlar Tablo 4.12 ve Tablo 4.13'te gösterilmiştir.

Tablo 4.12: Random Tree Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi.

		Gerçek Durum	
		Doğru	Yanlış
Kestirilen	Doğru	3848	805
	Yanlış	788	2064

Tablo 4.13: Random Tree Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Oranlar.

	TP Rate	FP Rate	Kesinlik (Precison)	Duyarlılık (Recall)	F-Ölçütü	ROC Area
Random Tree Algoritması	0,78	0,23	0,78	0,78	0,78	0,77

Sağ Omuz Esnekliği için REPTree algoritmasıyla elde edilen sonuçlar Tablo 4.14 ve Tablo 4.15'te gösterilmiştir.

Tablo 4.14: REPTree Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi.

		Gerçek Durum	
		Doğru	Yanlış
Kestirilen	Doğru	3706	947
	Yanlış	250	2602

Tablo 4.15: REPTree Algoritmasıyla Sağ Omuz Esnekliği İçin Elde Edilen Oranlar.

	TP Rate	FP Rate	Kesinlik (Precison)	Duyarlılık (Recall)	F-Ölçütü	ROC Area
REPTree Algoritması	0,84	0,13	0,85	0,84	0,84	0,88

Sol Omuz Esnekliği için karar ağaçlarıyla elde edilen sonuçlar incelendiğinde J48 Algoritmasıyla elde edilen sonuçlar Tablo 4.16 ve Tablo 4.17'de gösterilmiştir.

Tablo 4.16: J48 Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi.

		Gerçek Durum	
		Doğru	Yanlış
Kestirilen	Doğru	3247	445
	Yanlış	849	2964

Tablo 4.17: J48 Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Oranlar.

	TP Rate	FP Rate	Kesinlik (Precison)	Duyarlılık (Recall)	F-Ölçütü	ROC Area
J48 Algoritması	0,82	0,17	0,83	0,82	0,82	0,83

Sol Omuz Esnekliği için Decision Stump Algoritmasıyla elde edilen sonuçlar Tablo 4.18 ve Tablo 4.19'da gösterilmiştir.

Tablo 4.18: Decision Stump Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi.

		Gerçek Durum	
		Doğru	Yanlış
Kestirilen	Doğru	3592	100
	Yanlış	1061	2752

Tablo 4.19: Decision Stump Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Oranlar.

	TP Rate	FP Rate	Kesinlik (Precison)	Duyarlılık (Recall)	F-Ölçütü	ROC Area
Decision Stump Algoritması	0,84	0,15	0,87	0,84	0,84	0,84

Sol Omuz Esnekliği için Hoeffding Tree Algoritmasıyla elde edilen sonuçlar Tablo 4.20 ve Tablo 4.21'de gösterilmiştir.

Tablo 4.20: Hoeffding Tree Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi.

		Gerçek Durum	
		Doğru	Yanlış
Kestirilen	Doğru	3589	103
	Yanlış	1060	2753

Tablo 4.21: Hoeffding Tree Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Oranlar.

	TP Rate	FP Rate	Kesinlik (Precison)	Duyarlılık (Recall)	F-Ölçütü	ROC Area
Hoeffding Tree Algoritması	0,84	0,15	0,87	0,84	0,84	0,86

Sol Omuz Esnekliği için LMT Algoritmasıyla elde edilen sonuçlar Tablo 4.22 ve Tablo 4.23'de gösterilmiştir.

Tablo 4.22: LMT Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi.

		Gerçek Durum	
		Doğru	Yanlış
Kestirilen	Doğru	3586	106
	Yanlış	1058	2755

Tablo 4.23: LMT Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Oranlar.

	TP Rate	FP Rate	Kesinlik (Precison)	Duyarlılık (Recall)	F-Ölçütü	ROC Area
LMT Algoritması	0,84	0,15	0,86	0,84	0,84	0,88

Sol Omuz Esnekliği için Random Forest Algoritmasıyla elde edilen sonuçlar Tablo 4.24 ve Tablo 4.25'te gösterilmiştir.

Tablo 4.24: Random Forest Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi.

		Gerçek Durum	
		Doğru	Yanlış
Kestirilen	Doğru	3489	203
	Yanlış	957	2856

Tablo 4.25: Random Forest Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Oranlar.

	TP Rate	FP Rate	Kesinlik (Precison)	Duyarlılık (Recall)	F-Ölçütü	ROC Area
Random Forest Algoritması	0,84	0,15	0,86	0,84	0,84	0,90

Sol Omuz Esnekliği için Random Tree Algoritmasıyla elde edilen sonuçlar Tablo 4.26 ve Tablo 4.27'de gösterilmiştir.

Tablo 4.26: Random Tree Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi.

		Gerçek Durum	
		Doğru	Yanlış
Kestirilen	Doğru	2847	745
	Yanlış	879	2934

Tablo 4.27: Random Tree Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Oranlar.

	TP Rate	FP Rate	Kesinlik (Precison)	Duyarlılık (Recall)	F-Ölçütü	ROC Area
Random Tree Algoritması	0,77	0,23	0,77	0,77	0,77	0,77

Sol Omuz Esnekliği için REPTree Algoritmasıyla elde edilen sonuçlar Tablo 4.28 ve Tablo 4.29'da gösterilmiştir.

Tablo 4.28: REPTree Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Karmaşıklık Matrisi.

		Gerçek Durum	
		Doğru	Yanlış
Kestirilen	Doğru	3480	212
	Yanlış	1000	2813

Tablo 4.29: REPTree Algoritmasıyla Sol Omuz Esnekliği İçin Elde Edilen Oranlar.

	TP Rate	FP Rate	Kesinlik (Precison)	Duyarlılık (Recall)	F-Ölçütü	ROC Area
REPTree Algoritması	0,83	0,15	0,85	0,83	0,83	0,87

Cinsiyet sınıf olarak verildiğinde karar ağaçlarından iyi sonuçlar gösteren algoritmalar Tablo 4.30'da gösterilmiştir.

Tablo 4.30: Cinsiyet İçin Karar Ağacı Algoritması Sonuçları.

Karar Ağaçları	Cinsiyet
J48	%67,98
Hoefding Tree	%68,48
REPTree	%68,71
Random Forest	%74,16
Lmt	%72,85

Tablo 4.30'a bakıldığında Cinsiyet ortalama %70 oranında doğru sonuç verdi. Cinsiyet ile ilgili karar ağacı algoritmaları çalıştırıldığında elde edilen Karmaşıklık Matrisi ve bu matrise göre ortaya çıkan oranlar incelendiğinde Cinsiyet için J48 algoritmasıyla elde edilen sonuçlar Tablo 4.31 ve Tablo 4.32'de gösterilmiştir.

Tablo 4.31: J48 Algoritmasıyla Cinsiyet İçin Elde Edilen Karmaşıklık Matrisi.

		Gerçek Durum	
		Doğru	Yanlış
Kestirilen	Doğru	3575	977
	Yanlış	1426	1527

Tablo 4.32: J48 Algoritmasıyla Cinsiyet İçin Elde Edilen Oranlar.

	TP Rate	FP Rate	Kesinlik (Precison)	Duyarlılık (Recall)	F-Ölçütü	ROC Area
J48 Algoritması	0,68	0,37	0,67	0,68	0,67	0,67

Cinsiyet için Hoeffding Tree algoritmasıyla elde edilen sonuçlar Tablo 4.33 ve Tablo 4.34'de gösterilmiştir.

Tablo 4.33: Hoeffding Tree Algoritmasıyla Cinsiyet İçin Elde Edilen Karmaşıklık Matrisi.

		Gerçek Durum	
		Doğru	Yanlış
Kestirilen	Doğru	3738	814
	Yanlış	1551	1402

Tablo 4.34: Hoeffding Tree Algoritmasıyla Cinsiyet İçin Elde Edilen Oranlar.

	TP Rate	FP Rate	Kesinlik (Precison)	Duyarlılık (Recall)	F-Ölçütü	ROC Area
Hoeffding Tree Algoritması	0,68	0,38	0,67	0,68	0,67	0,70

Cinsiyet için LMT algoritmasıyla elde edilen sonuçlar Tablo 4.35 ve Tablo 4.36'da gösterilmiştir.

Tablo 4.35: LMT Algoritmasıyla Cinsiyet İçin Elde Edilen Karmaşıklık Matrisi.

		Gerçek Durum	
		Doğru	Yanlış
Kestirilen	Doğru	3796	756
	Yanlış	1281	1672

Tablo 4.36: LMT Algoritmasıyla Cinsiyet İçin Elde Edilen Oranlar.

	TP Rate	FP Rate	Kesinlik (Precison)	Duyarlılık (Recall)	F-Ölçütü	ROC Area
LMT Algoritması	0,72	0,32	0,72	0,72	0,72	0,78

Cinsiyet için Random Forest algoritmasıyla elde edilen sonuçlar Tablo 4.37 ve Tablo 4.38'de gösterilmiştir.

Tablo 4.37: Random Forest Algoritmasıyla Cinsiyet İçin Elde Edilen Karmaşıklık Matrisi.

		Gerçek Durum	
		Doğru	Yanlış
Kestirilen	Doğru	3868	684
	Yanlış	1255	1698

Tablo 4.38: Random Forest Algoritmasıyla Cinsiyet İçin Elde Edilen Oranlar.

	TP Rate	FP Rate	Kesinlik (Precison)	Duyarlılık (Recall)	F-Ölçütü	ROC Area
LMT Algoritması	0,74	0,31	0,74	0,74	0,73	0,80

Cinsiyet için REPTree algoritmasıyla elde edilen sonuçlar Tablo 4.39 ve Tablo 4.40'ta gösterilmiştir.

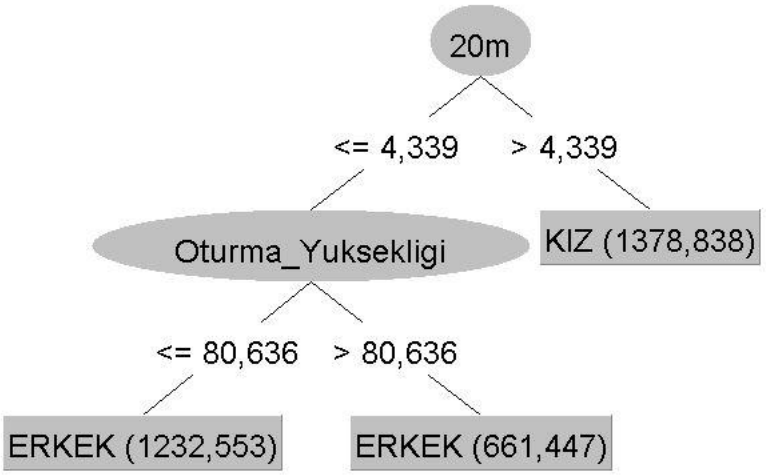
Tablo 4.39: REPTree Algoritmasıyla Cinsiyet İçin Elde Edilen Karmaşıklık Matrisi.

		Gerçek Durum	
		Doğru	Yanlış
Kestirilen	Doğru	3654	898
	Yanlış	1450	1503

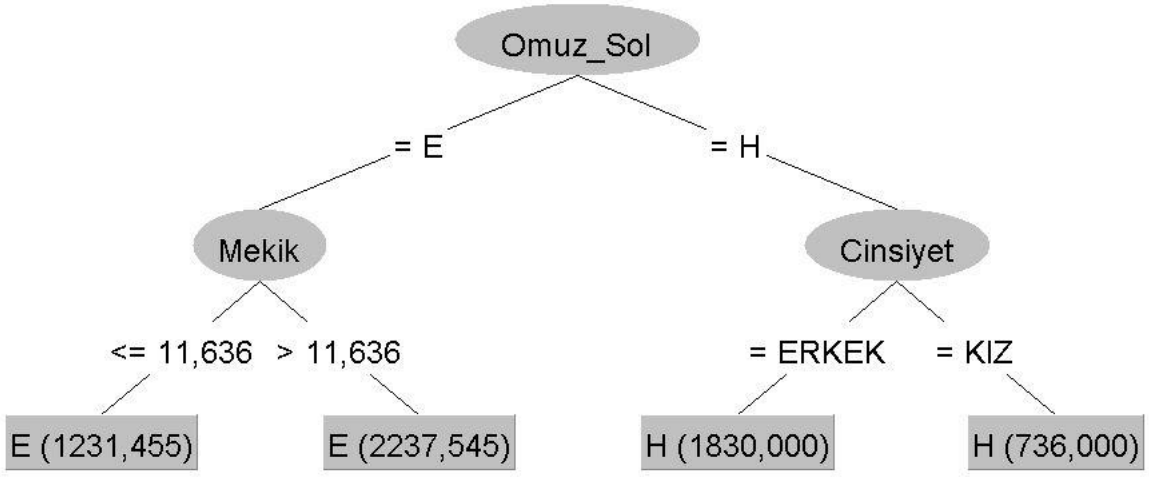
Tablo 4.40: REPTree Algoritmasıyla Cinsiyet İçin Elde Edilen Oranlar.

	TP Rate	FP Rate	Kesinlik (Precison)	Duyarlılık (Recall)	F-Ölçütü	ROC Area
REPTree Algoritması	0,68	0,37	0,68	0,68	0,68	0,71

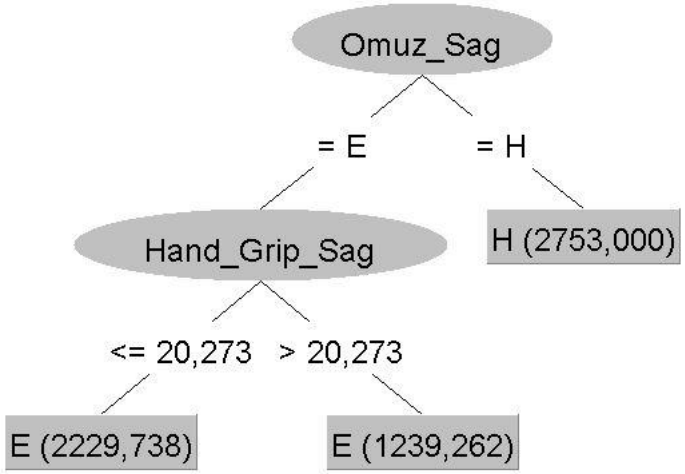
Özellik sayısı ve verilerin fazla olmasından karar ağacı çok büyük çıkmaktadır. Hoeffding Algoritmasıyla bu durum daha basit bir karar ağacı ortaya çıkardığı için Cinsiyet, Sağ Omuz Esnekliği ve Sol Omuz Esnekliği için Karar Ağacı örnekleri Şekil 4.1, Şekil 4.2 ve Şekil 4.3'te gösterilmiştir.



Şekil 4.1: Cinsiyet İçin Karar Ağacı Örneği.



Şekil 4.2: Sağ Omuz Esnekliği İçin Karar Ağacı Örneği.



Şekil 4.3: Sol Omuz Esnekliği İçin Karar Ağacı Örneği.

Karar Ağaçlarıyla Cinsiyet, Sağ Omuz Esnekliği ve Sol Omuz Esnekliği için elde edilen sonuçlara yakın Destek Vektör Makinesi ve Yapay Sinir Ağı ile de elde edildi. Elde edilen doğruluk oranları Tablo 4.41'de gösterilmiştir.

Tablo 4.41: Cinsiyet, Sağ Omuz Esnekliği ve Sol Omuz Esnekliği İçin Destek Vektör Makinesi ve Yapay Sinir Ağı İle Elde Edilen Sonuç.

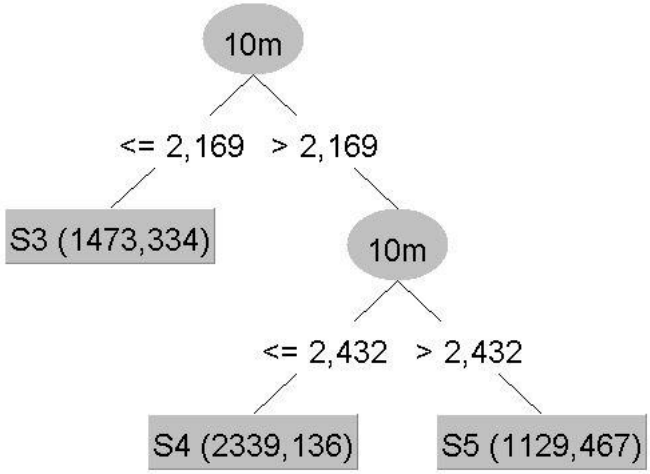
	Cinsiyet	Sağ Omuz Esnekliği	Sol Omuz Esnekliği
Destek Vektör Makinesi	%72,59	%84,53	%84,53
Yapay Sinir Ağı	0,71	0,84	0,84

20m Sürat sınıf olarak verildiğinde Karar Ağaçlarında iyi sonuçlar gösteren algoritmalar ve Destek Vektör Makinesinin sonucu Tablo 4.42'de gösterilmiştir.

Tablo 4.42: 20m Sürat İçin Karar Ağacı Algoritmaları ve Destek Vektör Makinesiyle Elde Edilen Sonuçlar.

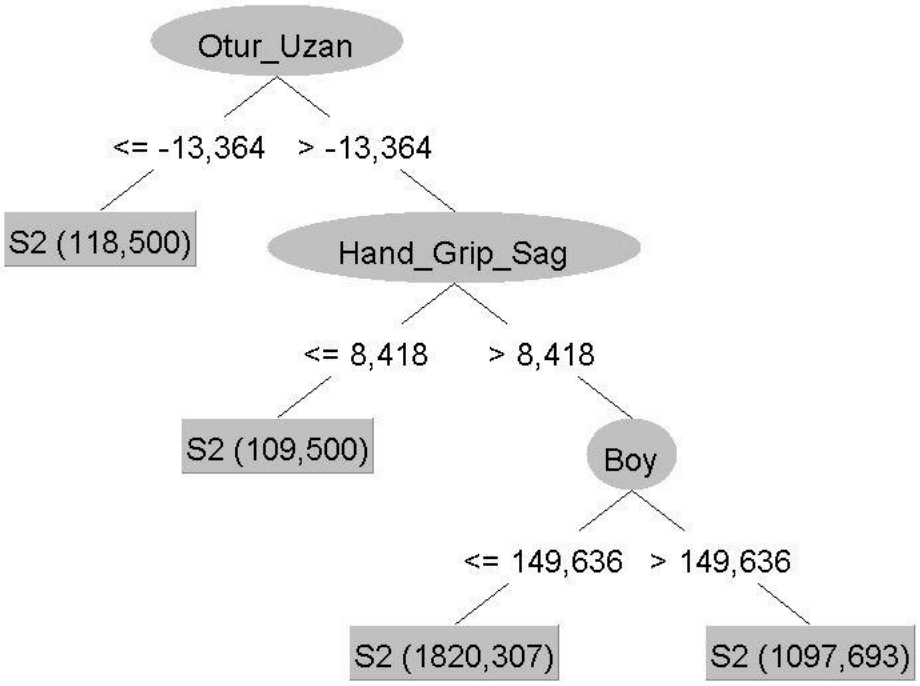
Karar Ağaçları	20m Sürat
J48	%79
Hoefding Tree	%77,85
REPTree	%80,81
Random Tree	%72,85
Random Forest	%81,58
LMT	%82,02
Destek Vektör Makinesi	%80,23

Tablo 4.42'ye bakıldığında 20m Sürat ortalama %80 oranında doğru sonuç verdi. 20m Sürat için de Karar Ağacı örneği Şekil 4.4'te gösterilmiştir.

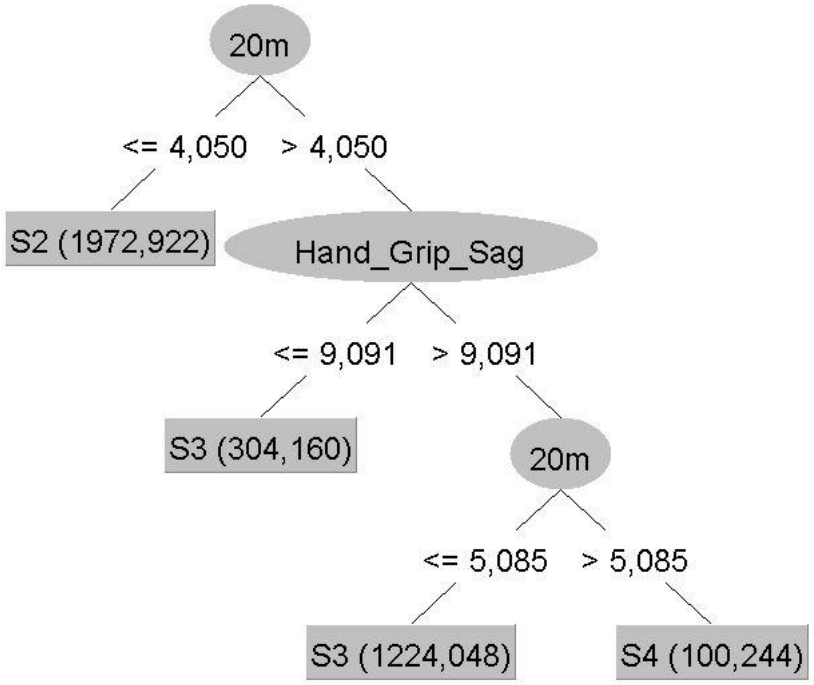


Şekil 4.4: 20m Sürat İçin Karar Ağacı Örneği.

Cinsiyet, Sağ Omuz Esnekliği, Sol Omuz Esnekliği ve 20m Sürat özelliklerinin sonuçları dışında 10m Sürat ve Yaş ile ilgili sonuçlarda korelasyon işlemiyle birlikte incelendi. S1, S2, S3 ifadeleri sınıf olarak belirtilen özelliğe etiket olarak verilmiştir. 10m Sürat ve Yaş ile ilgili de Karar Ağacı örnekleri Şekil 4.5 ve Şekil 4.6'da gösterilmiştir.



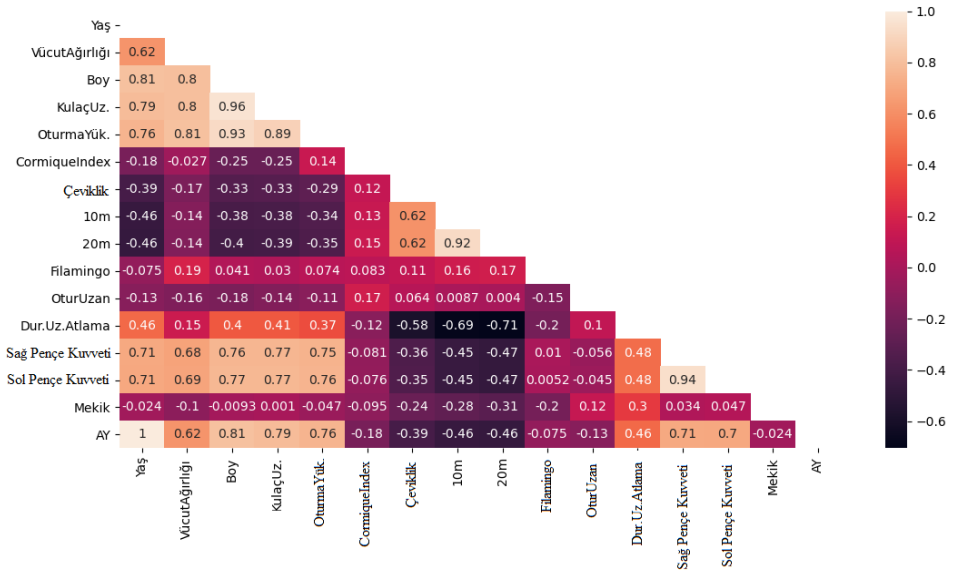
Şekil 4.5: Yaş İçin Karar Ağacı Örneği.



Şekil 4.6: 10m Sürat İçin Karar Ağacı Örneği.

7505 verinin incelenen özelliklerinden hangilerinin birbiriyle daha yakından ilişkili olduğunu tespit etmek için Python kodu yazılarak korelasyonuna bakıldı.

Python'da yapılan korelasyon sonucunda ortaya çıkan sonuç Şekil 4.7'deki gibidir.



Şekil 4.7: Python'da Yapılan Korelasyon Sonucu.

Korelasyon işleminde 0,8'den büyük olan özellikler çıkarıldığında elde edilen sonuçlara bakıldığında 10m Sürat, korelasyon yapılmadan önce verdiği sonuçtan %10 daha düşük olsa da iyi sonuçlar vermiştir. Üçerli olarak gruplandırılan (6-7-8, 9-10-11, 12-13-14) yaş özelliğinde de korelasyon öncesine yakın sonuçlar elde edildi.

Tablo 4.43: 10m Sürat ve Yaş İçin Korelasyon Öncesi Elde Edilen Sonuçlar.

	J48	Decision Stump	Hoefding Tree	REPTree	Random Tree	Random Forest	LMT	Destek Vektör Makinesi
10m Sürat	%89,38	%88,74	%88,59	%89,56	%85,01	%89,84	%89,83	%88,83
Yaş	%69,48	%68,18	%71,83	%72,16	%65,62	%74,43	%74,69	%74,44

Tablo 4.44: 10m Sürat ve Yaş İçin Korelasyon Sonrası Elde Edilen Sonuçlar.

	J48	Decision Stump	Hoefding Tree	REPTree	Random Tree	Random Forest	LMT	Destek Vektör Makinesi
10m Sürat	%75,98	%75,45	%69,04	%78,69	%71,55	%80,31	%80,85	%80,63
Yaş	%67,86	%68,62	%71,76	%72,03	%65,44	%73,91	%74,76	%74,24

Diğer özellikler verildiğinde sınıf olarak belirtilen özelliğin elde edilmesinin, korelasyon öncesi ve sonrası oluşan değişikliklerinin sonuçlarına bakıldı. Daha az sayıda özellikle yine diğer özelliği elde edebilme durumu oluştuğunda ölçüm yapan kişiler için bu daha büyük bir avantaj sağlar. Özellikle sportif verim açısından antrenöre verilerden çıkarım yapabilme konusunda kolaylık oluşturabilecek ve zaman tasarrufu sağlayabilecek durumlar oluşur.

10m Sürat özelliğinde Yaş, Ay, Sağ Pençe Kuvveti, Sol Pençe Kuvveti, Boy Uzunluğu, Çeviklik (ZigZag), Kulaç Uzunluğu, Kormik İndeks, Oturma Yüksekliği, Filamingo Denge Testi, Otur Uzan Esnekliği, Sağ Omuz Esnekliği, Sol Omuz Esnekliği çıkarılıp Cinsiyet, Vücut Ağırlığı, Durarak Uzun Atlama ve Mekik verildiğinde Karar Ağacı Algoritmaları ve Destek Vektör Makinesiyle elde edilen sonuçlar Tablo 4.45'te gösterilmiştir.

Tablo 4.45: Cinsiyet, Vücut Ağırlığı, Durarak Uzun Atlama ve Mekik Verildiğinde 10m Sürat İçin Elde Edilen Karar Ağacı Algoritmaları ve Destek Vektör Makinesi Sonucu.

Karar Ağaçları	10m Sürat
J48	%78,22
Decision Stump	%75,45
Hoefding Tree	%77,92
REPTree	%77,45
Random Tree	%69,03
Random Forest	%75,48
LMT	%78,32
Destek Vektör Makinesi	%78,30

Tablo 4.45'e bakıldığında korelasyon sonrası 10m Sürat için elde edilen oranlara çok yakın sonuçlar elde edildiği gibi sadece 4 özellik ile 10m Süratin bulunması bu testleri gerçekleştiren kişiler için kolaylık sağlayacağı gibi daha hızlı sonuçlar elde etmelerine de yardımcı olacaktır.

5. TARTIŞMA VE SONUÇ

Sporun dünyada popülerliğinin artması sonucu ilgi durumunu da sporcuların ortaya koydukları performansa doğru yönlendirmiştir. Performans sporunda elit düzeye ulaşabilmek için çocukların küçük yaştan itibaren doğru branşlara yönlendirilmek üzere yetenek seçimi kavramı altında performans takibi gereklidir. Yetenek seçimi ve buna bağlı performans takibi büyüme ve fiziksel gelişim aşamalarında çeşitli ölçümlerle gerçekleştirilmektedir.

Bu çalışma kapsamında 6-14 yaş aralığında çocukların antropometrik ve motorik özelliklerine ilişkin veriler sınıflandırma işlemleri için önem taşımaktadır. Bu çalışmada kullanılan veriler profesyonel bir performans ölçüm ekibinden sağlanmıştır.

Çalışmada Kuvvet (Pençe Kuvveti, Mekik), Denge (Flamingo), Sürat (10m - 20m), Çeviklik (Zigzag), Esneklik (Otur Uzan, Omuz Esnekliği), Güç (Durarak Uzun Atlama) gibi motorik özellikler ve Vücut Ağırlığı, Oturma Yüksekliği, Boy Uzunluğu, Kulaç Uzunluğu gibi antropometrik özelliklere ilişkin veriler kullanılmıştır.

Gelişen teknoloji ve dijitalleşme kavramlarıyla Yapay Zeka, Makine Öğrenmesi gibi kavramlar daha da önem kazanmıştır. Karar Ağaçları, Destek Vektör Makinesi ve Yapay Sinir Ağları verilerin işlenerek sonuç elde edilmesinde kullanılan modellerdir. Bu tez kapsamında 7505 çocuğun yukarıda belirtilen parametrelerine ilişkin veriler Karar Ağacı Algoritmalarıyla, Destek Vektör Makinesiyle ve Yapay Sinir Ağı modellemesiyle çalıştırılarak sonuçlar elde edilmiştir.

Karar Ağacı Algoritmaları, Destek Vektör Makinesi , Yapay Sinir Ağı modellemesiyle 10m Sürat özelliğine ilişkin sonuçlar yaklaşık %89 oranında doğrulukla elde edilmiştir. Sağ Omuz Esnekliği ve Sol Omuz Esnekliği özellikleri yaklaşık %84 oranında doğrulukla, 20m Sürat özelliği yaklaşık %80 oranında, Cinsiyet ve Yaş özellikleri yaklaşık %70 oranında doğrulukla sonuçlanmıştır.

Özelliklerin Python kodlamasıyla korelasyonuna bakılmıştır. Korelasyon katsayısı 0,8'in üzerinde olan özellikler veri setinden çıkarıldığında, 10m Sürat özelliği yaklaşık %10 daha düşük doğrulukla bulunsa da %80'e yakın oranda doğruluğa ulaşılmıştır. Yaş özelliği için korelasyon öncesi elde edilen sonuçlar ile korelasyon sonrası elde edilenler sonuçlar birbirine yakın bulunmuştur.

Korelasyonu 0,8 üzerinde bulunan özelliklerin veri setinden çıkarılmasının ardından Yaş, Ay, Sağ Pençe Kuvveti, Sol Pençe Kuvveti, Boy Uzunluğu, Çeviklik (ZigZag), Kulaç Uzunluğu, Kormik İndeks, Oturma Yüksekliği, Filamingo Denge Testi, Otur Uzan Esnekliği, Sağ Omuz Esnekliği, Sol Omuz Esnekliği özelliklerinin de çıkarılmasıyla 10m Sürat özelliği J48, LMT ve Destek Vektör Makinesi algoritmalarıyla %78 doğruluk oranında bulunmuştur. Yalnızca 4 özellikle (Cinsiyet, Vücut Ağırlığı, Durarak Uzun Atlama ve Mekik) yapılan sınıflamada 10m Sürat özelliğinin yüksek doğruluk oranıyla sonuçlar verdiği gözlemlenmiştir.

Daha az sayıda özelliğin yer aldığı veri seti kullanılarak yüksek doğrulukla bir özelliğin elde edilmesi, bu verilerden çıkarım yapacak ve faydalanacak kişiler için sportif değerlendirme açısından zaman, iş gücü tasarrufu ve objektiflik sağlayacaktır.

Spor branşlarında spesifik olarak ihtiyaç duyulan özelliklerin, antrenörlük stratejileri de dikkate alınarak birbirleriyle ilişkisinin incelenip bu özelliklerden branşın ihtiyacı olan tüm özellikleri en iyi temsil eden grubun hassasiyeti yüksek biçimde belirleneceği bir çalışma yapılması spor camiasında antrenörlere, veri toplayan ve işleyen kişilere verilerin değerlendirilmesinde destek olacak, büyük kolaylık ve fayda sağlayacaktır.

KAYNAKLAR

[1]. Kanat Usta, H., 2019, *Yapay sinir ağları temelli sporda yetenek yönlendirme karar destek sistemi tasarımı*, Yüksek Lisans, Sağlık Bilimleri Enstitüsü.

[2]. Fırat, O., Fırat, S., 2017, Endüstri 4.0 yolculuğunda trendler ve robotlar, *Istanbul university journal of the school of business,* 46 (2), 211-223.

[3]. Nabiyev, V., 2010, *Yapay zeka insan-bilgisayar etkileşimi,* Seçkin Yayıncılık, Ankara, ISBN: 978-975-02-1156-0.

[4]. Onan, A., 2014, Şirket iflaslarının tahmin edilmesinde karar ağacı algoritmalarının karşılaştırmalı başarım analizi, *Bilişim teknolojileri dergisi,* 8 (1). 9-19.

[5]. Kuzey, C., 2012, *Veri madenciliğinde destek vektör makinaları ve karar ağaçları yöntemlerini kullanarak bilgi çalışanlarının kurum performansı üzerine etkisinin ölçülmesi ve bir uygulama,* Doktora, Sosyal Bilimler Enstitüsü

[6]. Güler, D., Günay, M., Baltacı, G., Gökdemir, K., 2004, 8-10 yaş grubu türk erkek çocukların sağlıkla ilişkili fiziksel uygunluk normları, Gazi Üniversitesi Kırşehir Eğitim Fakültesi, 5 (2), 157-164.

[7]. Çamlıgüney, A. F., 2010, *8-10 yaş grubu kız çocukların fiziksel aktivite düzeyleri ile fiziksel uygunlukların belirlenmesi ve beslenme profili ile ilişkisinin incelenmesi*, Doktora, Sağlık Bilimleri Enstitüsü.

[8]. Sayın, N., 2014, *15-17 yaş grubu gençlerin fiziksel aktivite düzeyleri ile fiziksel uygunlukları arasındaki ilişki,* Yüksek Lisans, Sağlık Bilimleri Enstitüsü.

[9]. Uçan, İ., Buzdağlı, Y., Ağgön, E., 2018, Çocuklarda sporun fiziksel uygunluk üzerine etkisinin incelenmesi, *Atatürk üniversitesi beden eğitimi ve spor bilimleri dergisi,* 20 (3), 123-133.

[10]. Stadudenmayer, J., Pober, D., Crouter, S., Bassett, D., Freedson, P., 2009, An artificial neural network to estimate physical activity energy expenditure and identify physical activity type from an accelerometer, *Journal of applied physiology*, 107 (4), 1300-1307.

[11]. Silva, A. J., Costa, A. M., Oliveria, P. M., Reis, V. M., Saavedra J., Perl, J., Rouboa, A., Marinho, D.A., 2007, The use of neural network technology to model swimming performance, *Journal of sports science and medicine*, 6 (1), 117-125.

[12]. Ruiz, J. R., Ramirez-Lechuga, J., Ortega, F. B., Castro-Pinero, J., Benitez, J. M., Arauzo-Azofra, A., Sanchez, C., Sjöström, M., Castillo, M. J., Gutierrez, A., Zabala, M., 2008, Artificial neural network-based equation for estimating VO2max from the 20 m shuttle run test in adolescents, *Artificial intelligence in medicine,* 44 (3), 233-245.

[13]. Woods, C. T., Veale, J., Fransen, J., Robertson, S., Collier, N. F., 2018, Classification of playing position in elite junior Australian football using technical skill indicators, *Journal of sports sciences,* 36 (1), 97-103.

[14]. Eskoifer, B. M., Federolf, P., Kugler, P. F., Nigg, B. M., 2011, Marker-based classification of young–elderly gait pattern differences via direct PCA feature extraction and SVMs, *Computer methods in biomechanics and biomedical engineering,* 16 (4), 435-442.

[15]. Makridakis, S., 2017, The forthcoming artificial intelligence (AI) revolution: its impact on society and firms, *Futures,* 90, 46-60.

[16]. Aydın, Ş. E., 2017, *Yapay zeka teknolojisi (yapay zekaların dünü bugünü yarını),* Yüksek Lisans, Sosyal Bilimler Enstitüsü.

[17]. Kornienko, A. A., Kornienko, A. V., Fofanov, O. B., Chubik, M. P., 2015, Knowledge in artificial intelligence systems: searching the strategies for application, *Procedia - Social and behavioral sciences,* 166 (2015), 589-594.

[18]. Heller, M., 2019, *What is machine learning? Intelligence derived from data,* https://www.infoworld.com/article/3214424/what-is-machine-learning-intelligence-derived-from-data.html, [Ziyaret Tarihi: 5 Mayıs 2020]

[19]. Faggella, D., 2020, *What is machine learning?,* https://emerj.com/ai-glossary-terms/what-is-machine-learning/, [Ziyaret Tarihi: 5 Mayıs 2020]

[20]. Singh, K., 2018, *Understanding different components & roles in data science*, https://dimensionless.in/understanding-different-components-roles-in-data-science/, [Ziyaret Tarihi: 6 Mayıs 2020].

[21]. Garcia-Laencina, P. J., Sancho-Gomez, J. L., Figueiras-Vidal, A. R., 2010, Pattern classification with missing data: a review, *Neural computing and applications*, 19 (2), 263-282.

[22]. Birant, D., 2011, Comparison of decision tree algorithms for predicting potential air pollutant emissions with data mining models, *Journal of environmental informatics*, 17 (1), 46-53.

[23]. Nguyen, A., 2017, *Comparative study of C5.0 and cart algorithms*, http://mercury.webster.edu/aleshunas/Support%20Materials/C4.5/Nguyen-Presentation%20Data%20mining.pdf, [Ziyaret Tarihi: 8 Mayıs 2020].

[24]. Witten, I. H., Frank, E., 2005, *Data mining*, Morgan kaufmann publishers, San francisco, ISBN: 0-12-088407-0.

[25]. McClendon, L., Meghanathan, N., 2015, Using machine learning algorithms to analyze crime data, *Machine learning and applications: an international journal*, 2 (1), 1-12.

[26]. Kumar, A., Kaur, P., Sharma P., 2015, A Survey on hoeffding tree stream data classification algorithms, Career point university hamirpur – research journal, 1 (2), 28-32.

[27]. Chen, W., Xie, X., Wang, J., Pradhan, B., Hong, H., Bui, D. T., Duan, Z., Ma. J., 2017, A comparative study of logistic model tree, random forest, and classification and regression tree models for spatial prediction of landslide susceptibility, *Catena*, 151, 147-160.

[28]. Genuer, R., Poggi, J. M., Tuleau-Malot, C., 2010, Variable selection using random forests, *Pattern recognition letters*, 31 (14), 2225-2236.

[29]. Kalmegh, S., 2015, Analysis of weka data mining algorithm reptree, simple cart and randomtree for classification of indian news, *International journal of innovative science, engineering & technology*, 2 (2), 438-446.

[30]. Kaur, G., Chhabra, A., 2014, Improved j48 classification algorithm for the prediction of diabetes, *International journal of computer applications*, 98 (22), 13-17.

[31]. Güven, Z. B., Bilgin, T. T., 2014, Zaman serileri madenciliği kullanılarak nüfus artışı tahmin uygulaması, *Akademik bilişim'14 - XVI. akademik bilişim konferansı bildirileri,* 5-7 Şubat 2014 Mersin Üniversitesi, Mersin, 41-45.

[32]. Gürsakal, N., 2018, *Makine öğrenmesi,* Dora yayıncılık, Bursa, ISBN: 978-605-247-063-3.

[33]. *Support vector machines*, https://scikit-learn.org/stable/modules/svm.html, [Ziyaret Tarihi: 24 Nisan 2020]

[34]. Öztemel, E., 2012, *Yapay sinir ağları,* Papatya yayıncılık, İstanbul, ISBN: 978-975-6797-39-6.

[35]. Schmidhuber, J., 2015, Deep learning in neural networks: an overview, *Neural networks,* 61 (2015), 85-117.

[36]. Cao, C., Liu, F., Tan, H., Song, D., Shu, W., Li, W., Zhou, Y., Bo, X., Xie, Z., 2018, Deep learning and its applications in biomedicine, *genomics, proteomics & bioinformatics,* 16 (1), 17-32.

[37]. Güzel, Y., 2018, *Yapay zeka ders notları 03 | biyolojik sinir sistemi ve yapay sinir ağı hücresi*, https://medium.com/@yasinguzel/yapay-zeka-ders-notlar%C4%B1-03-biyolojik-sinir-sistemi-ve-yapay-sinir-a%C4%9F%C4%B1-h%C3%BCcresi-6555add68d80, [Ziyaret Tarihi: 27 Nisan 2020]

[38]. Bouckaert, R. R., Frank, E., Hall, M., Kirkby, R., Reutemann, P., Seewald, A., Scuse, D., 2013, *WEKA manual for version 3-7-8,* University of waikato, New Zealand.

[39]. Açıl, S., 2017, Python: korelasyon matrisi ve görselleştirilmesi, https://medium.com/@sddkal/python-korelasyon-matrisi-ve-g%C3%B6rselle%C5%9Ftirilmesi-554093ad59, [Ziyaret Tarihi: 30 Nisan 2020].

[40]. Şana, M., 2017, *Pyton ile basit bir ysa uygulaması,* https://www.pythondersleri.com/2017/01/python-ile-basit-bir-ysa-uygulamasi.html, [Ziyaret Tarihi: 6 Mayıs 2020].

ÖZ GEÇMİŞ

Kişisel Bilgiler	
Adı Soyadı	Uğur Özbalkan
Doğum Yeri	İstanbul/Eminönü
Doğum Tarihi	30.09.1995
Uyruğu	☑ T.C. ❒ Diğer:
Telefon	0534 898 45 70
Web Adresi	



Eğitim Bilgileri Lisans	
Üniversite	İstanbul Üniversitesi
Fakülte	Mühendislik Fakültesi
Bölümü	Bilgisayar Mühendisliği
Mezuniyet Yılı	2018

Yüksek Lisans	
Üniversite	İstanbul Üniversitesi-Cerrahpaşa
Enstitü Adı	Lisansüstü Eğitim Enstitüsü
Anabilim Dalı	Bilgisayar Mühendisliği Anabilim Dalı
Programı	Bilgisayar Mühendisliği

Makale ve Bildiriler

1) 2ND INTERNATIONAL CONFERENCE ON LIFE AND ENGINEERING SCIENCES PROCEEDING BOOK

-An Automated Dynamic Analysis System to Characterize Android APPS