



**DOKUNMATİK YÜZEYLERDE KULLANICI TİPİNİN
ANALİZİ VE DOKUNMA HASSASİYETİNİN
KULLANICI TİPİNE GÖRE BELİRLENMESİ**

Egemen ENGİZEK



T.C.
BURSA ULUDAĞ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**DOKUNMATİK YÜZEYLERDE KULLANICI TİPİNİN ANALİZİ VE
DOKUNMA HASSASİYETİNİN KULLANICI TİPİNE GÖRE BELİRLENMESİ**

Egemen ENGİZEK
502205015

Doç. Dr. Neyir ÖZCAN SEMERCİ
(Danışman)

YÜKSEK LİSANS TEZİ
ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

BURSA – 2024
Her Hakkı Saklıdır

ÖZET

Yüksek Lisans Tezi

DOKUNMATİK YÜZEYLERDE KULLANICI TİPİNİN ANALİZİ VE DOKUNMA HASSASIYETİNİN KULLANICI TİPİNE GÖRE BELİRLENMESİ

Egemen ENGİZEK

Bursa Uludağ Üniversitesi
Fen Bilimleri Enstitüsü
Elektronik Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Neyir ÖZCAN SEMERCİ

Biyometrik veriler kişiye özgü ve değişmeyen özellikler olarak tanımlanır. Kişiye ait bu özellikler fiziksel (parmak izi, avuç içi izi, yüz, iris, retina, kulak, el damarı, vücut kokusu veya DNA bilgisi şeklinde) veya davranışsal (ses, yürüyüş, imza vb.) olabilir. Bu özellikler kişiye özel olduğundan biri ya da birkaçı ile kimlik tanıma ve doğrulama işlemleri yapılabilmektedir. Soft biyometrik veriler ise kişiye ait boy, kilo, göz rengi, saç rengi, saç yoğunluğu, etnik köken ve ırk gibi daha genel özellikleri temsil etmektedir. Bu çalışmada, dokunmatik yüzey (cep telefonu) kullanan kişinin dokunma biçimi, dokunma karakteristik özellikleri gibi soft biyometrik verileri ile cep telefonuna entegre edilen bir devre yardımıyla ölçülen biyoempedansı kullanılmıştır. Yapılan çalışmada yaşları 10 ile 65 arasında değişen 164 (86 erkek, 78 kadın) gönüllüden dokunmatik ekrana 4 farklı çizimi (saat yönünde çember çizimi, saat yönü tersine çember çizimi, saat yönünde üçgen çizimi, saat yönü tersine üçgen çizimi) ve 4 farklı kaydırma (yukarı kaydırma, aşağı kaydırma, sola kaydırma ve sağa kaydırma) işlemlerini 15 defa tekrarlamaları istenmiş, eş zamanlı olarak gönüllülerin biyoempedans ölçümleri telefon kılıfına monte edilmiş devreye bağlı bakır bantlara dokunmaları suretiyle yapılmıştır. Elde edilen veriler makine öğrenmesi yöntemleri (K-NN, DVM ve Mantıksal Regresyon) ile analiz edilerek kişilerin yaş aralığı ve cinsiyeti belirlenmiştir. Yapılan bu sınıflandırma sonucunda oluşturulan profil, kayıtlı kullanıcılar ile karşılaştırılarak kimlik doğrulama için kullanılabilir. Ayrıca belirlenen yaş aralığına göre cihazın kullanımı engellenebilir ya da bazı uygulamalara erişimine kısıtlama getirilebilir. Yapılan bu çalışma ile kişilerin cihaz kullanımlarında düşük güvenli kimlik doğrulama için eşsiz biyometrik verilerini paylaşma gerekliliğinin ortadan kaldırılması amaçlanmaktadır.

Anahtar Kelimeler: Soft Biyometri, Biyoempedans, Kimlik Doğrulama, Makine Öğrenmesi, Biyometrik veri, Destek Vektör Makineleri, K-En yakın komşular, Regresyon.

2024, ix + 93 sayfa

ABSTRACT

MSc Thesis

ANALYSIS OF USER TYPE ON TOUCHPADS AND DETERMINATION OF TOUCH SENSITIVITY BY USER TYPE

Egemen ENGİZEK

Bursa Uludağ University
Graduate School of Natural and Applied Sciences
Department of Electronics Engineering

Supervisor: Assoc. Prof. Dr. Neyir ÖZCAN SEMERCİ

Biometric data is a type of data that defined as person-specific and unchanging characteristics. These characteristics of a person can be physical (in the form of fingerprints, palm prints, face, iris, retina, ears, hand veins, body odor or DNA information) or behavioral (voice, gait, signature, etc.). Physical features are personal; identification and verification can be done with one or more of them. Soft biometric data, on the other hand, representing more general characteristics of the person such as height, weight, eye color, hair color, hair density, ethnicity and race. In this study, soft biometric data such as the touch style and touch characteristics of the person using a touch surface (mobile phone) and the bioimpedance measured with the help of a circuit integrated into the mobile phone were used. In the study, 160 volunteers (80 men, 80 women) aged between 10 and 65 have been asked to make 4 different drawings (clockwise circle drawing, counterclockwise circle drawing, clockwise triangle drawing, counterclockwise triangle drawing) and 4 different swipe movements (swiping left, swiping right, swiping down and swiping up) repeating processes 15 times, and simultaneously, the volunteers' bioimpedance measurements were made by touching the metal copper bands connected to the circuit mounted on the phone case. The obtained data were analyzed with machine learning methods (K-NN, SVM, Linear Regression) and the age range and gender of the people were determined. The profile created because of this classification can be used for authentication by comparing it with registered users. Additionally, depending on the specified age range, the use of the device may be prevented or access to some applications may be restricted. This study aims to eliminate the need for people to share their unique biometric data for low-security authentication when using their devices.

Key words: Soft Biometrics, Bioimpedance, Authentication, Machine Learning, Biometric data, Support Vector Machines, K-Nearest Neighbour, Regression.

2024, ix + 93 pages.

TEŞEKKÜR

Bu uzun süren yolculukta bana ilham veren, rahat bir çalışma ortamı sunan, bu tezin bitirilmesi için en az benim kadar çalışan ve bana destek olan danışman hocam Sn. Doç. Dr. Neyir ÖZCAN SEMERCİ'ye teşekkürlerimi sunarım.

Gönüllü olarak bu çalışmaya katılan kişilere teşekkürlerimi sunarım.

Çalışmalarında bana fikir veren, sorularıma cevap bulan arkadaşlarım Aytunç POLAT ve Selcan GÜNER'e teşekkürlerimi sunarım.

Yüksek lisans öğrenimim boyunca benden desteklerini esirgemeyen, her ihtiyacımda yanımda olan sevgili annem Dilek YORULMAZ'a ve kardeşim Ece ONAT'a teşekkürlerimi sunarım.

Son olarak uzun çalışma saatlerinde yanında olamadığım ve bu uzun süren yüksek lisans yolculuğumda çok büyük bir sabırla beni destekleyen, moral veren sevgili eşim Didem ENGİZEK'e teşekkürlerimi sunarım.

Egemen ENGİZEK
27/06/2024

İÇİNDEKİLER

	Sayfa
ÖZET.....	i
ABSTRACT.....	ii
TEŞEKKÜR.....	iii
SİMGELER ve KISALTMALAR DİZİNİ	v
ŞEKİLLER DİZİNİ.....	vii
ÇİZELGELER DİZİNİ	ix
1. GİRİŞ.....	1
2. KURAMSAL TEMELLER ve KAYNAK ARAŞTIRMASI	4
2.1. Biyoelektrik Empedans	4
2.2. Soft Biyometri.....	7
2.3. Makine Öğrenmesi	10
2.3.1. Denetimli Öğrenme	11
2.3.2. KNN (K-en yakın komşular).....	14
2.3.3. DVM-Destek Vektör Makineleri	17
2.3.4. Aşırı Öğrenme (Overfitting) ve Yetersiz Öğrenme (Underfitting)	32
2.4. Kimlik Doğrulama	45
2.5. Doğruluk, Kesinlik, Duyarlılık, F1-Skor ve Karmaşıklık Matrisi	47
3. MATERYAL ve YÖNTEM.....	50
3.1. Kullanılan Ekipman	50
3.2. Yöntem.....	54
3.3. Öznitelik Seçimi ve Veri Ön işleme.....	67
4. BULGULAR	73
4.1. Sınıflandırma Sonuçları	73
4.2. Regresyon Sonuçları	77
5. TARTIŞMA ve SONUÇ	82
KAYNAKLAR	86
EKLER.....	90
ÖZGEÇMİŞ	93

SİMGELER ve KISALTMALAR DİZİNİ

Simgeler	Açıklama
α	Ölçeklendirme parametresi
A_i	Kosinüs mesafesindeki A vektörün bileşeni
B_i	Kosinüs mesafesindeki B vektörün bileşeni
b	Sapma
C	Düzenleyici parametre
$\cos \theta$	Kosinüs teta
d_{12}	Boşluktaki iki noktanın birbirine olan mutlak uzaklığı
d	Polinom boyutu
e	Eksponansiyel fonksiyon
F_1	F1-Skor performans metriği
Im	Empedans sanal kısım
k	Komşu sayısı
$k\Omega$	Kiloohm (Direnç Birimi)
Ω	Ohm (Direnç Birimi)
θ	Faz Açısı
R_e	Empedans gerçek kısım
σ	Çekirdek ağırlığını belirleyen RTÇ parametresi
$\tan$	Tanjant
V	Volt (Gerilim Birimi)
w^T	Ağırlık vektörü transpozu
x	Girdi vektörü
x_i	Öklid mesafesindeki x düzlemindeki veri değeri
x_k	Minkowski mesafesinde x düzlemindeki veri değeri
$X1$	Polinom çekirdek işlevindeki ilk veri değeri
$X2$	Polinom çekirdek işlevindeki ikinci veri değeri
$\hat{y}$	Hedef etiketler
y_i	Öklid mesafesindeki y düzlemindeki veri değeri
y_k	Minkowski mesafesinde y düzlemindeki veri değeri
Y	Polinom çekirdek işlevi çıktısı
Z	Empedans birimi

Kısaltmalar	Açıklama
ADC	Analog Digital Converter / Analog Dijital Çevirici
BEA	Biyoelektrik Empedans Analizi
DFT	Digital Fourier Transform / Dijital Fourier Dönüşümü
DR	Doğrusal Regresyon
DVM	Destek Vektör Makineleri
ECW	Extracellular Water / Hücre Dışı Sıvı Miktarı
FFM	Fat Free Mass / Yağsız Vücut Kütlesi
FM	Fat Mass / Vücut Yağ Kütlesi
FN	False Neagative / Yanlış Negatif
FP	False Positive / Yanlış Pozitif
IBM	International Business Machines Corporation
ICW	Intracellular Water / Hücre İçi Sıvı Miktarı
IQR	Inter Quartile Range / Çeyrek Açıklığı
I2C	Inter-Integrated Circuit / Entgre Devreler arası
KNN	K-Nearest Neighbour / K-En Yakın Komşular
LBM	Lean Body Mass / Yağsız Vücut Kütlesi
MATLAB	Matris Laboratuvarı
MHz	Mega Hertz
MÖ	Makine Öğrenmesi
MSPS	Megasamples per second / Saniyedeki Mega Örnek
PC	Personal Computer / Kişisel Bilgisayar
PIN	Personal Identification Number /Kişisel Tanımlama Numarası
RBF	Radial Basis Function
RF	Radyo Frekansı
Rfb	Feedback resistor / Geri besleme direnci
RMSE	Root Mean Square Error / Kök Ortalama Kare Hatası
RTÇ	Radyal Tabanlı Çekirdek
SDK	Software Development Kit / Yazılım Geliştirme Seti
SMA	SubMiniature version A / Minyatür altı A versiyon
TBW	Total Body Water / Vücut Su Miktarı
TN	True Negative / Gerçek Negatif
TP	True Positive / Gerçek Pozitif
USB	Universal Serial Bus / Evrensel Seri Veriyolu
YZ	Yapay Zeka
.csv	Comma Seperated Value / Virgülle Ayrılmış Değerler

ŞEKİLLER DİZİNİ

Sayfa

Şekil 2.1.1.	Frekans büyüklüklerine göre hücreden elektrik akım geçişleri. (Canbolat, 2018).....	5
Şekil 2.1.2.	Kompleks düzlemde empedans ve faz açısı gösterimi (Wikipedia, 2009).....	6
Şekil 2.3.1.	Makine öğrenmesi ve ilişkili alanları. (Sindhu, 2020)	11
Şekil 2.3.1.1.	Sınıflandırma ve regresyon (Candan H., 2021).....	12
Şekil 2.3.2.1.	KNN algoritmaları. A) İki boyutlu düzlemde KNN (Mattaparthi, 2023) B) Üç boyutlu düzlemde KNN (Gong, 2022).....	15
Şekil 2.3.3.1.	İki sınıftan örneklerle eğitilmiş bir DVM için maksimum marj hiper düzlemi (Jiaying ve ark. 2018)..	18
Şekil 2.3.3.2.	Sert ve yumuşak marj örnekleri (Kaushal ve ark. 2021).....	20
Şekil 2.3.3.3.	Farklı 'C' değerleri için marj büyüklüğü örneği (Shrimali, 2018).....	22
Şekil 2.3.3.4.	Doğrusal şekilde olmayan veri dağılımı (Karabudak, 2024).....	23
Şekil 2.3.3.5.	İki boyutlu bir dağılımın üç boyutlu hale getirilip sınıflara ayrılması (Mansour ve arkadaşları, 2023).....	24
Şekil 2.3.3.6.	Doğrusal DVM (A), polinom çekirdek işlemli (d=2) (B), ve polinom çekirdek İşlemli (d=5)(C) sınıflandırma (Ben-Hur ve ark., 2008).....	25
Şekil 2.3.3.7.	Hiperbolik tanjant fonksiyon grafiği (Manton, 2023)...	26
Şekil 2.3.3.8.	Sigmoid çekirdeğe sahip bir DVM tarafından oluşturulmuş karar sınırları.....	27
Şekil 2.3.3.9.	Boşluktaki iki noktanın birbirlerine olan uzaklıkları (Sreenivasa, 2020).....	28
Şekil 2.3.3.10.	$\sigma=1$ için gerçekleşen RTÇ grafiği (Sreenivasa, 2020)..	29
Şekil 2.3.3.11.	$\sigma=0.1$ için gerçekleşen RTÇ grafiği (Sreenivasa, 2020).....	30
Şekil 2.3.3.12.	$\sigma=10$ için gerçekleşen RTÇ grafiği (Sreenivasa, 2020).	30
Şekil 2.3.3.13.	Üç sınıflı iki özellikli verilerin RTÇ işlevinde farklı gamma değerleri kullanılarak sınıflandırılması (Akça, 2020).....	32
Şekil 2.3.4.1.	Sınıflandırma algoritması üzerinden öğrenme metotları (Taş, 2020).....	35
Şekil 2.3.4.2.	Regresyon modeli üzerinden öğrenme metotları (Amidi A. ve Amidi S., 2018).....	36
Şekil 2.3.4.3.	Öğrenme eğrileri (Durna, 2020).....	36
Şekil 2.3.4.4.	Varyans ve yanlılık karşılaştırması (Hafeez, 2018)....	37
Şekil 2.3.4.5.	Çapraz doğrulama diyagramı (Peeyada ve ark., 2022)	39
Şekil 2.3.4.6.	Yetersiz ve aşırı öğrenme örneği (Holbrook ve Cook, 2024).....	40

Şekil 2.3.4.7.	Izgara arama örneği (Çavdar ve ark., 2024).....	41
Şekil 2.3.4.8.	Izgara arama ile iki hiper parametrenin bir performansı metriği olan kesinlik kriterini gösteren şema (Joos, 2020).....	43
Şekil 2.3.4.9.	Rastgele arama çapraz doğrulama (Çavdar ve ark., 2024).....	44
Şekil 2.3.4.10.	Izgara arama ile rastgele arama arasındaki hiper parametre kombinasyonu seçimi (Trouvain ve ark., 2020).....	45
Şekil 2.5.1.	Basit yapıli karmaşıklık matrisi.....	47
Şekil 3.1.1.	Çalışmada kullanılan akıllı telefon.....	50
Şekil 3.1.2.	AD5933EBZ geliştirme kartı.....	51
Şekil 3.1.3.	AD5933 elektriksel diyagramı.....	52
Şekil 3.1.4.	AD5933EBZ arayüz yazılımı.....	53
Şekil 3.2.1.	Kullanıcıdan istenilen kaydırma ve çizimler.....	55
Şekil 3.2.2.	Uygulamadan örnek ekran görüntüleri.....	56
Şekil 3.2.3.	Soft Biyometri verilerinin toplandığı örnek tablo.....	56
Şekil 3.2.4.	Uygulamadaki onFling fonksiyonu.....	58
Şekil 3.2.5.	onFling fonksiyonunun çağırdığı liste.....	59
Şekil 3.2.6.	Uygulamadaki onDown fonksiyonu.....	59
Şekil 3.2.7.	Uygulamadaki onScroll fonksiyonu.....	60
Şekil 3.2.8.	handleButtonClick(), startRecording(), stopRecording() fonksiyonları.....	61
Şekil 3.2.9.	Hareketlerin sınıf içine alınıp sıralanması.....	61
Şekil 3.2.10.	Uygulama akış diyagramı.....	62
Şekil 3.2.11.	Deney düzeneği.....	63
Şekil 3.2.12.	Kalibrasyon sonrası doğrulama.....	64
Şekil 3.2.13.	20 kişiden alınan avuç içi direnç değerleri.....	64
Şekil 3.2.14.	16Mhz ile yapılan farklı gerilim çıkışlarının örnek empedans ve faz sonuçları.....	65
Şekil 3.2.15.	4Mhz ile yapılan farklı gerilim çıkışlarının örnek empedans ve faz sonuçları.....	66
Şekil 3.3.1.	Çeyrek açıklık kutu diyagramı.....	69
Şekil 4.1.1.	Sınıflandırma sonuçları.....	74
Şekil 4.1.2.	Gözlem sayısına bağlı Kuadratik DVM sınıflandırıcısına ait karmaşıklık matrisi.....	74
Şekil 4.1.3.	Yüzde başarımlı oranlı Kuadratik DVM sınıflandırıcısına ait karmaşıklık matrisi (Gerçek pozitifler ve yanlış negatifler).....	75
Şekil 4.1.4.	Gözlem sayısına bağlı ince KNN sınıflandırıcısına ait karmaşıklık matrisi.....	75
Şekil 4.1.5.	Yüzde başarımlı oranlı ince KNN sınıflandırıcısına ait karmaşıklık matrisi (Gerçek pozitifler ve yanlış negatifler).....	76
Şekil 4.1.6.	Gözlem sayısına bağlı etkili mantıksal regresyon sınıflandırıcısına ait karmaşıklık matrisi.....	76

Şekil 4.1.7.	Yüzde başarımlı oranlı etkili mantıksal regresyon sınıflandırıcısına ait karmaşıklık matrisi (Gerçek pozitifler ve yanlış negatifler).....	77
Şekil 4.2.1.	Regresyon modelleri başarı oranları.....	78
Şekil 4.2.2.	Orta ölçek Gauss çekirdek işlevli DVR cevap grafiği..	79
Şekil 4.2.3.	Orta ölçek Gauss çekirdek işlevli DVR için beklenen ve gerçekleşen cevaplar.....	79
Şekil 4.2.4.	Etkileşimli doğrusal regresyon grafiği.....	80
Şekil 4.2.5.	Etkileşimli doğrusal regresyon için beklenen ve tahmin edilen cevaplar.....	80
Şekil 4.2.6.	A) Izgara arama çapraz doğrulama olmadan sınıflandırma modelleri başarıları. B) Izgara arama çapraz doğrulama olmadan yapılan regresyon modelleri hata oranı.....	81
Şekil 5.1.	Sınıflandırma modellerinin çapraz doğrulama uygulanmadan ve uygulandıktan sonraki başarı performans grafiği.....	82
Şekil 5.2.	DVR modelinin çapraz doğrulama uygulanmış ve uygulanmamış haldeki hata oranları.....	83
Şekil 5.3.	Doğrusal regresyon modellerinde çapraz doğrulama yöntemi uygulanmış ve uygulanmamış haldeki hata oranları.....	84

ÇİZELGELER DİZİNİ

		Sayfa
Çizelge	2.3.3.1. Sert marj ve yumuşak marj karşılaştırması.....	23
Çizelge	3.2.1. Gönüllülerin yaş ve cinsiyet dağılımı.....	54
Çizelge	3.2.2. Deneye katılan toplam kişi sayısı ve yaş ortalamaları...	54
Çizelge	3.2.3. Arayüz parametre tanımlamaları.....	67
Çizelge	3.3.1. Soft biyometri öznelilikleri.....	71
Çizelge	3.3.2. Empedans ve faz açılarından elde edilen öznelilikler....	72

1. GİRİŞ

Günümüzde kimlik doğrulama birçok farklı yerde ve şekillerde kullanılmaktadır. Kimlik doğrulama, geleneksel olarak anahtar ve şifre onaylayıcılarla dijital olarak ise iris tanımlama, yüz tanımlama ve parmak izi tanımlama şeklinde yapılabilmektedir. Mobil uygulamalarda ise dokunma desenleri çizimleri ya da sesli komutlar kimlik doğrulama için kullanılabilmektedir. Kimlik doğrulama şekilleri çok farklı olmakla beraber, her doğrulama işlemi kullanıcıyı doğrulama ve sisteme erişim izni verip vermeme şeklinde aynı iki işlem adımını takip eder.

Biyometrik verileri kullanarak kimlik doğrulama, teknolojinin de ilerlemesi ile birlikte oldukça popülerleşmiştir. Mobil cihaz kullanımı arttıkça, insan ve makine arasındaki veri alışverişi artmış, bu sayede kullanıcılardan çeşitli verilerin elde edilmesi mümkün olmuştur. Biyometrik veri, dokunmatik ekrana sahip akıllı cep telefonlarını etkinleştirmek, telefona yüklü uygulamaları açmak ve telefonda yapılan çeşitli işlemler için kullanılabilmektedir. Bütün bu işlemler, kimlik doğrulamada biyometrik veri kullanımı ile daha hızlı ve güvenilir olmaktadır. Biyometrik veriler kişiye özgü ve değişmeyen özellikler olarak tanımlanır. Eşsiz biyometrik veriler ise parmak izi, yüz, iris, retina gibi kişiye özel özelliklerdir. Bu eşsiz veriler sadece bu özellikleri sağlayan kişiyi tanımlar ve oldukça güvenilirdir. Bunun yanında soft-biyometrik olarak tanımlanan veriler ise literatürde “Bir kişinin kimliği hakkında bazı bilgiler sağlayan ancak kimliği kesin olarak belirlemek için yeterli kanıt sağlamayan anatomik veya davranışsal özellikler” olarak tanımlanmaktadır (K. Nandakumar ve A.K. Jain, 2009, s. 1235–1239). Yaş, etnik köken, cinsiyet, boy, kilo, yara izleri ve dövme gibi özellikleri içerir (Miguel-Hurtado, Stevenage, Bevan ve Guest, 2016). Literatürde mevcut olan çalışmalar, soft biyometri kullanımının biyometrik sistem performansını artırabildiğini ve büyük veri tabanlarında arama süresini büyük ölçüde azaltabildiğini göstermiştir (K. Nandakumar ve A.K. Jain, 2009). Bu çalışmalarda ilk kimlik doğrulama için eşsiz-biyometri teknikleri kullanılırken, deneğin kimliğinin sürekli olarak doğrulanması için ise soft-biyometri özelliklerinin bir kombinasyonu kullanılmıştır (Miguel-Hurtado ve diğerleri, 2016).

Mobil cihazlarda kimlik doğrulama için ilk oturum açıldığında geleneksel olarak PIN kodu ve şifre istenir. Bu şifre ve PIN kodu girişinden sonra kullanıcının sisteme erişimine izin verilir. Bu tip bir doğrulamanın dezavantajı, bu işlemin sadece bir kere oturum açıldığında yapılmasıdır. Bu da oturum sonlanana kadar cihazın ve içerisindeki önemli bilgilerin savunmasız kalması anlamına gelir. (Sivaz ve Aykut, 2021).

Bunun dışında kimlik doğrulama için yaygın olarak kullanılan yöntemlerden biri olan gizli dokunma desenleri de ekran üzerinde parmaktan geçen yağın iz bırakması ve desenin yapıldığı anda gizlice gözlenmesi gibi güvenlik tehditlerini içermektedir. (Aviv, Gibson, Mossop, Blaze, ve Smith, 2010; Wiedenbeck, Waters, Sobrado, ve Birget, 2006).

Biyoelektrik empedans analizi (BEA) vücuttan zayıf bir elektrik akımının aktığı ve vücudun empedansını (direnç ve reaktans) hesaplamak için voltajın ölçüldüğü, özellikle vücut yağ ve kas kütlelerini tahmin etmeye yönelik bir yöntemdir. Vücutta bulunan kas kütlelerinin artması nedeniyle depolanan su miktarının artması empedansın düşmesine neden olur. BEA vücut yağ oranını tahmin etmek, bazı hastalıkların ön tanısını oluşturmak ve çeşitli destek sağlık programları oluşturmak için tüketici pazarlarında sıklıkla kullanılır.

Bu tez çalışmasında kişilerin cinsiyet ve yaş sınıflandırması soft biyometrik veriler kullanılarak makine öğrenmesi yöntemleri ile yapılmıştır. Kullanılacak olan soft biyometrik veriler, Bursa Uludağ Üniversitesi, Tıp Fakültesi Klinik Araştırmalar Etik Kurulunun 13.06.2023 tarih, 2023-13/4 nolu kararında vermiş olduğu izinle gönüllü bireylerden toplanmıştır. Toplanan bu veriler bir cep telefonu uygulaması ve cep telefonu kılıfına monte edilmiş bir elektronik devre kullanılarak elde edilmiştir. Yapılan çalışmada yaşları 10 ile 65 arasında değişen 164 (86 erkek, 78 kadın) gönüllüden dokunmatik ekrana 4 farklı çizimi (saat yönünde çember çizimi, saat yönü tersine çember çizimi, saat yönünde üçgen çizimi, saat yönü tersine üçgen çizimi) ve 4 farklı kaydırma (yukarı kaydırma, aşağı kaydırma, sola kaydırma ve sağa kaydırma) işlemlerini 15 defa tekrarlamaları istenmiş, eş zamanlı olarak gönüllülerin biyoelektrik

empedans ölçümleri telefon kılıfına monte edilmiş devreye bağlı bakır bantlara dokunmaları suretiyle yapılmıştır. Toplanan bu veriler makine öğrenmesi sınıflandırma ve regresyon modelleri ile işlenmiş, kullanılan modellerin sınıflandırma başarımları ölçülmüştür.

Şu ana kadar eşsiz ve soft biyometrik verilerin birlikte kullanılması görülmüş olsa da biyoelektrik empedans ve soft biyometrik verilerin birlikte kullanıldığı çalışmaya bu tezin yazıldığı zaman itibariyle henüz rastlanılmamıştır. Çalışma, bu yönüyle, problem olarak yenilikçidir.

2. KURAMSAL TEMELLER ve KAYNAK ARAŞTIRMASI

2.1. Biyoelektrik Empedans

Biyoelektrik empedans analizi (BEA), en basit ifadesi ile bir kişinin elektrik empedansını yani vücut dokuları boyunca bir elektrik akımının akışına olan karşıtlığını belirler; bu veri daha sonra dokuların yapısını, yağ su dengesini ve kütlelerini tahmin etmek için kullanılabilir ve vücut ağırlığına, vücut yağına göre farklılık gösterir.

Vücut empedans ölçümü sırasında elde edilebilecek veriler Yağsız Vücut Kütlesi (FFM), Vücut Su Miktarı (TBW), Hücre Dışı Sıvı Miktarı (ECW), Hücre içi Sıvı Miktarı (ICW) şeklindedir. Vücudun toplam ağırlığı bilindiği takdirde bu veriler doğrultusunda Vücut Yağ Kütlesi (FM), Yağsız Vücut Yüzdesi (LBM%) de elde edilebilir.

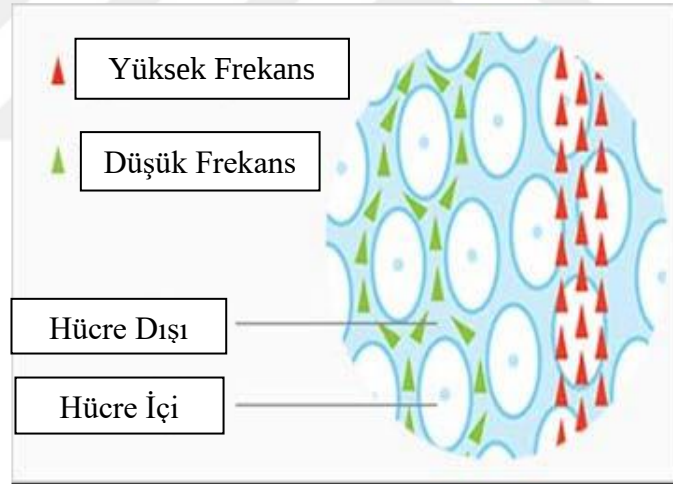
Vücudun mineral yapısı ve su oranına bağlı olarak, yüksek ve düşük frekansın geçirgenliği farklıdır. Düşük frekanslarda hücre zarları kapasitif etki gösterir ve elektrik akımının geçişine engel oluştururlar. Düşük frekanslarda ECW hakkında bilgi sahibi olunabilir. Yüksek frekanslarda ise elektrik akımının hücre zarının kapasitif etkisi yok olur ve elektrik akımı hem hücre içi hem de hücre dışı sıvıdan geçer. Bu şekilde TBW ölçülebilir.

Biyoelektrik empedans analizi (BEA), güvenli ve hızlı olmasının yanı sıra vücut bütünlüğüne zarar vermemesi ve nispeten düşük maliyetli olması nedeniyle, kliniklerde hastaların vücut kompozisyonlarını değerlendirmek için yaygın bir şekilde kullanılmaktadır. Sadece empedans değerleri değil ayrıca faz açısı da hastalıkların analizi açısından önemli görülüp kullanılmaya başlanmıştır. BEA analizinden elde edilen empedans bileşenleri ile hesaplanan faz açısı, beslenme durumu ve morbidite riskinin değerlendirilmesinde objektif ve hızlı bir yöntem olarak büyük ilgi çekmektedir. Faz açısının, kötüleşen klinik durumlar ve çeşitli hastalıkların ölüm

oranları için önemli bir parametre olduğu vurgulanmaktadır. Literatürde, faz açısının hastalıklar, fiziksel aktivite düzeyi ve beslenme durumu gibi birçok faktörle olan ilişkisini inceleyen pek çok çalışma bulunmaktadır. Ancak, belirli hastalıklar, bireyler veya diğer faktörler için referans değerler henüz kesinleşmemiştir. (Canbolat E., 2018).

BEA'nın frekans, empedans ve faz açısı olmak üzere üç bileşeni mevcuttur.

Uygulamalarda değişen akımın frekansına bağlı olarak, hücre zarları iletken ya da kapasitif etki gösterirler. Düşük frekanslarda sadece hücre dışı sıvı hakkında bilgi sahibi olunabilirken, yüksek frekanslarda hücre zarının kaybolan kapasitif etki ile akım hem hücre içi hem hücre dışından geçerek toplam sıvı miktarı hakkında bilgi edinilmesini sağlar. Frekansa göre hücre zarı geçirgenliği Şekil 2.1.1'de gösterilmiştir.



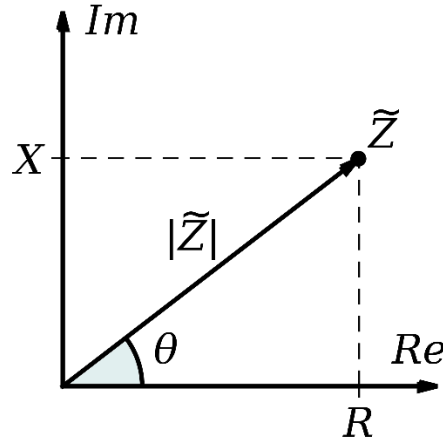
Şekil 2.1.1. Frekans büyüklüklerine göre hücreden elektrik akım geçişleri (Canbolat 2018'den değiştirilerek alınmıştır)

Kullanılan frekans değerlerine göre iki tip BEA sistemi vardır. Tek frekanslı BEA yöntemi 'Klasik BEA' yöntemi olarak bilinmektedir. Dört elektrot ile yapılan bu tip BEA'da sabit bir frekansta akım vücut içerisinde ilerlemektedir. Tek frekanslı BEA, TBW ve LBM analizinde en sık kullanılan yöntem olmakla beraber, TBW'nin hücre içi ve dışını kısımlarını ayırt etmede yetersizdir. Tek frekanslı BEA ölçümleri hidrasyon durumu belirgin bir şekilde değişmiş kişilerde anlamlı değildir ancak normal hidrat

kişilerde, yağsız kitle veya toplam vücut suyunu kolaylıkla hesaplayabilir (Canbolat E., 2018).

Çoklu frekans BEA uygulaması ise farklı frekanslar kullanarak sadece hücre dışı değil hücre içindeki sıvı miktarını da belirleyebilir. Yüksek frekansta akım hücre içi sıvısından geçmeye başlar. Çoklu frekans BEA, sıvı değişimlerini ve sıvı dengesini daha iyi açıklamakta olup hidrasyon seviyesindeki değişimleri incelemede de kullanışlıdır. Yağ kütlesi hakkında bilgi sağlamanın yanında çoklu frekans BEA'nın tek frekanslı BEA'ya göre, ekstremite iskelet kaslarını değerlendirme avantajı vardır (Nalçacıoğlu, 2014; Özçetin ve Khalilova, 2017; Aydın 2004).

Empedans, bir malzeme içinden geçen alternatif akıma karşı frekansa bağlı olan elektriksel dirençtir. Empedans 'Z' ile ifade edilir ve birimi (Ω) Ohm'dur. Bu tezde empedans biyolojik maddelerin üzerinden değerlendirildiği için 'Biyoempedans' olarak tanımlanmıştır. Şekil 2.1.2'de empedans ve faz açısına bağlı tüm değerler gösterilmiştir.



Şekil 2.1.2. Kompleks düzlemde empedans ve faz açısı gösterimi (Wikipedia, 2009)

$$|\tilde{Z}|^2 = X^2 + R^2 \quad (2.1)$$

Denklem 2.1'de gösterilen empedans değerinin karesi rezistans ve reaktansın karelerinin toplamıdır. Frekans değişimi ile reaktans kısım artabilir ya da azalabilir. Eğer faz açısı

pozitif ise malzeme indüktif, eğer faz açısı negatif ise malzeme kapasitif etki gösteriyor demektir.

Düşük faz açısı değeri veya düşük reaktans, hücre ölümü ya da hücre zarının seçici geçirgenliğinin bozulmasını ifade eder. Yüksek faz açısı değeri ise düşük faz açısındaki durumun tersine, daha büyük miktarlarda bozulmamış hücre zarlarını ifade eder. Yüksek faz açısı en iyi hücre fonksiyonu ile birlikte en güçlü hücre zarının durumunu gösterir. Bundan dolayı faz açısı hücre sel sağlık durumunun en iyi tanımlayıcı olduğu ifade edilmektedir. Yapılan çalışmalar incelendiğinde kadınlardaki faz açısının erkeklere göre; yaşlı bireylerdekini de genç bireylere kıyasla daha düşük olduğu görülmektedir. Bu durum erkek ve genç bireylerin, kadınlara ve yaşlı bireylere kıyasla daha yüksek kas kütlesine ve daha az yağ kütlesine sahip olması ile açıklanabilir.

2.2. Soft Biyometri

Soft Biyometri özellikleri, “Bir kişinin kimliği hakkında bazı bilgiler sağlayan ancak kimliği kesin olarak belirlemek için yeterli kanıt sağlamayan anatomik veya davranışsal özellikler” olarak tanımlanmaktadır. Bunlar, yaş, etnik köken, cinsiyet, boy, kilo, yara izleri ve dövme ler gibi özellikleri içerir. Bu özellikler, biyometri dağıtımında, parmak izi (Ailisto ve arkadaşları, 2006), iris (Zewail ve arkadaşları 2004) ve yüz (Park ve Jain, 2010) gibi katı biyometri yöntemleriyle kombinasyon halinde kullanılmıştır. Çalışmalar, soft biyometri kullanımının biyometri sistem performansını artırabildiğini ve büyük veri tabanlarında arama süresini büyük ölçüde azaltabildiğini göstermiştir. Soft biyometri yüksek seviyeli ipuçları olmasına rağmen, sürekli kimlik doğrulama senaryolarında kullanıma sunulma konusunda büyük ölçüde yetersizdir (Niinuma ve arkadaşları, 2010). Bu durumlarda, ilk kimlik doğrulama için eşsiz biyometrik teknikleri kullanılırken, deneğin sürekli olarak kimliğini doğrulamak için soft biyometri özelliklerinin bir kombinasyonu kullanılır (Miguel-Hurtado ve diğerleri, 2016).

Soft biyometri uygulamalarının, dokunmatik ekrana veya yüzeye sahip cihazların kullanımının artması ile alanları genişlemiş ve makine öğrenmesi ile popülerleşmiştir.

Bu noktada kişinin tanımlanması için gerekli olan fiziksel ayırt edici özelliklerinin dışına çıkılarak, çok fazla miktarda veri içeren yeni tanımlayıcı setler oluşturulabilmektedir. Cep telefonu ve tablet gibi günlük hayatta sıkça kullanılan cihazlar sayesinde kişilerin parmakları ile yaptıkları tarama, sürükleme, dokunma gibi davranışları rahatlıkla gözlemlenebilir, analiz edilebilmektedir.

Angulo ve Wästlund'un (2011) 32 gönüllü ile yaptığı çalışmada bir Android işletim sistemine sahip cep telefonlar üzerinde gönüllülerin ekran kilidini açmak için kullanılan dokunma desenleri çizimleri istenmiştir. Üç farklı desenin her biri 50 kez tekrarlanmıştır. Desenlerde bulunan 9 farklı noktadan belirlenmiş olanların üzerinde parmağın durma süresi ve noktaları tararken geçen süre hesaplanıp daha sonrasında bu verileri makine öğrenmesi yöntemleri ile değerlendirilmiştir. Sonradan bu veriler ile kişi tahmini yapılmıştır. Hata oranı %10,39 olarak hesaplanmıştır. Altı noktalı bir deseni doğru tahmin etme oranı herhangi biri için $1/16032 = 0,00006$ olarak hesaplanmıştır. Hata oranına istinaden uyumlu kişi olmasa bile deseni doğru yapan ve sisteme erişme başarısı 0,05 olarak hesaplanmıştır. Dolayısıyla deseni bilmeyip ve veri setindeki kişilerden olmayan birisinin deseni doğru yapıp sisteme erişme şansı $0,000006 \times 0,05 = 0,000003$ olarak bulunmuştur. Burada farklı kimlik doğrulama sistemleri ve biyometrik verilerin birlikte kullanıldığında daha güvenli olduğu görülmüştür.

Acien, Morales, Fierrez, Vera-Rodriguez, Hernandez Ortega (2018) yaptıkları araştırmada 3 ile 6 yaş arasındaki 89 çocuk ile yaş tahminlerinde farklı sınıflandırma teknikleri kullanarak %90 ile %96 arasında bir başarı oranı yakalamışlardır. Yapılan çalışmada kullanıcılara akıllı cep telefonları ile dokunma ve kaydırma hareketleri yapmaları istenmiştir. Kaydırma hareketi sırasında ortalama hız, maksimum ivme ve kaydırma toplam süreleri dikkate alınmıştır. Dokunma esnasında ise dokunulması istenilen bölge ile gerçekten dokunulan bölge arasındaki mesafe ve dokunma süresi elde edilmiştir. Tek bir makine öğrenme algoritması kullanarak çocukların nöromotor becerilerinin giderek arttığı saptanabilmiştir. Bu da tehlikeli durumlarda küçük çocukların sisteme erişim izni sağlanamayacağı bir faktör olarak dikkate alınmıştır.

Soft Biyometri sadece hareketler ile tanımlanan bir olgu değildir. Bevan ve Fraser (2016) yaptıkları çalışmada farklı fiziksel özelliklere sahip kişilerin bu fiziksel özellikleri ile ayrıştırılabilir olup olmadıklarını araştırmışlardır. Bu kapsamda 178 kişiden, tasarlanmış bir cep telefonu uygulaması aracılığıyla, 4 farklı sürüklemeye hareketi (yukarıdan aşağıya, aşağıdan yukarıya, soldan sağa ve sağdan sola) yapılması istenmiştir. Her hareket 30 defa tekrarlanmış ve hareketlerden 6 farklı karakteristik özellik çıkarılmıştır. Bunlar kaydırma uzunluğu, kaydırma zamanı, ortalama kaydırma temas genişliği, ortalama basınç, maksimum hız ve maksimum ivmedir. Buradaki özgün yaklaşım, hareketlerin sadece baş parmak ile yapılmasıdır. Kullanıcıların hareketleri yapmadan önce bilekten baş parmak ucuna kadar olan uzunluğu ölçülmüş ve sol elini kullanan kullanıcıların sağdan sola kaydırmada daha hızlı oldukları; sağ elini kullanan kullanıcıların soldan sağa kaydırmada daha hızlı oldukları sonucuna ulaşılmıştır. Burada bileğin ve elin doğal uzanma ve kapanma hareketi de göz önünde bulundurulmuştur. Daha uzun baş parmak uzunluğuna sahip olanlar, bütün kaydırma işlemlerinde daha kısa baş parmağa sahip olanlardan daha hızlı olmuşlardır.

Miguel-Hurtado, Stevenage, Bevan ve Guest'in (2016) yaptığı çalışmada hareket biçimlerinin, farklı karakteristik özellikleri ile çeşitli makine öğrenmesi algoritmalarını ayrı ayrı ve birlikte karşılaştırarak cinsiyet tahmininde %78 doğruluk oranı yakalanmıştır. 116 kişiyle yapılan bu çalışmada hareketlerin sadece bir defa yapılması (yukarıdan aşağıya, aşağıdan yukarıya, soldan sağa ve sağdan sola) karakteristik özellik ve makine öğrenmesi yöntemlerinin ne kadar önemli olduğunu vurgulamaktadır. Burada dikkat edilen nokta her bir hareket karşısında cinsiyet tahminine en çok yarayan karakteristik özellik ve algoritma kullanımıdır. Böylece her özellik, her hareket ve algoritma için geçerli olmayıp modelde ağırlığı azaltılmıştır.

Rzecki Pławiak, Niedźwiecki ve Sośnicki (2017) desenli örüntü tekniğini kullanarak, 50 kişiyle, 9 farklı hareket ve her hareket için 10 tekrar ile yaptıkları çalışmada daha önceki çalışmalardan farklı olarak kişi tanımlamada %99 gibi büyük bir başarı yakalamışlardır. Buradaki en önemli faktör toplanan verilerin herhangi bir makine öğrenme metoduna girmeden önce ön işlem yapılmasıdır. Bu işlemler sırasıyla; yeniden örnekleme,

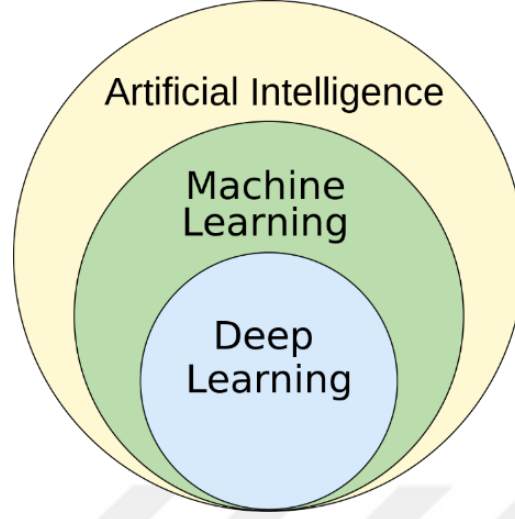
standartlaştırma, temel bileşen analizi, boyut indirgeme, birleştirme, normalizasyon ve veri uzatımıdır. Bu yöntemle yüksek başarı oranı sağlanmıştır; ancak bu işlemler oldukça maliyetli ve dikkatli yapılması gereken işlemlerdir.

2.3. Makine Öğrenmesi

Makine öğrenimi (MÖ), bilgisayarların verileri analiz ederek insanlar gibi öğrenmesini sağlayan bir bilgisayar bilimidir. MÖ, bir bilgisayarın açık ve doğrudan talimatlar olmadan matematiksel modellerin yardımı ile öğrenebilir. Bu durum yapay zekanın (YZ) bir alt kümesi olarak kabul edilir. Verilerdeki kalıpları tanımlamak için MÖ algoritmaları kullanır. Bu kalıplar, tahmin yapabilen bir veri modeli oluşturmak için kullanılır. İnsanların pratik yaparak kendilerini geliştirdiği gibi veri havuzu ve deneyim arttıkça makine öğrenmesi de daha doğru sonuçlar verir (Microsoft Azure, 2024).

İlk olarak 1952 yılında ifade edilen MÖ, resmi olarak Arthur Samuel tarafından tasarlanmıştır. Tasarlanan model IBM laboratuvarlarında dama oynaması için denenmiştir. Arthur Samuel'in tasarladığı bu programın dama oynadığı her seferde kendisini geliştirmesi, hatalarını belirleyip düzeltmesi ve elde ettiği verilerle oyunu kazanmanın yollarını keşfetmesini amaçlanmıştır. Bu kendiliğinden öğrenme programı, makine öğrenmesinin ilk örneklerinden biri olmuştur. Kendi kendine öğrenme ve makine öğrenmesi gibi terimler de bu dönemde kullanılmaya başlanmıştır.

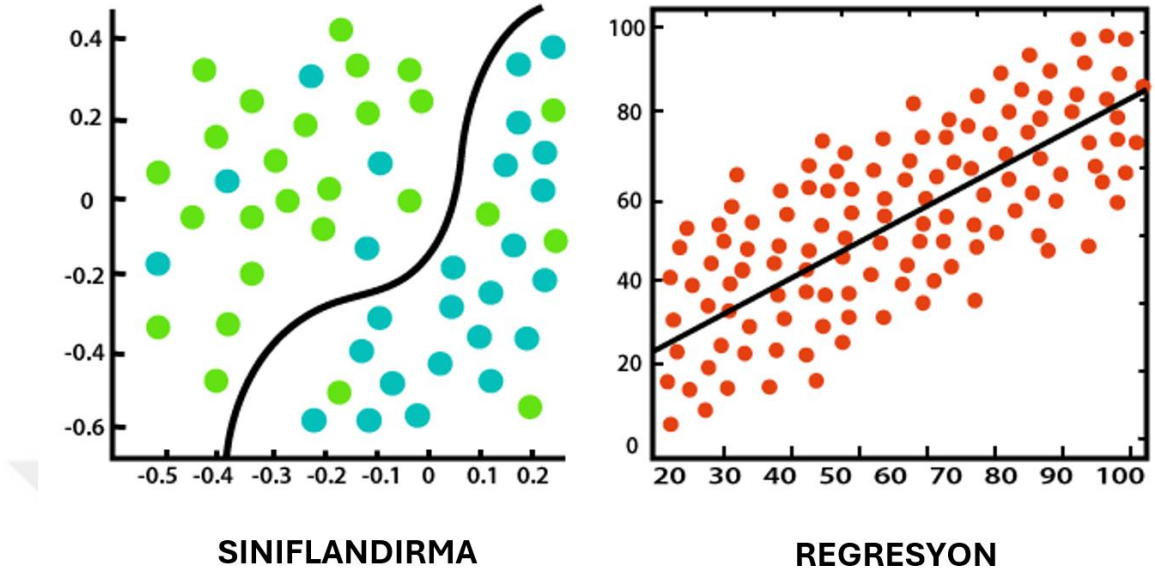
MÖ birçok farklı alanlar ile bütünleşik olarak çalışmaktadır. Örnek olarak MÖ, YZ'nin bir alt kümesidir ve YZ'nin çıktılarını kullanır. YZ insan zekasını taklit eden daha geniş kapsamlı bir kümedir. MÖ ve YZ sıklıkla birbirinin yerine kullanılır; ancak gerçekte aynı anlama gelmezler. MÖ'nün bir alt kümesi ve özel bir biçimi olan 'Derin Öğrenme' ise doğruluğu kendi başına sağlayabilen bir sinir ağı kullanır. Burada kullanılan derin sıfatı öğrenme aşamasında kullanılan farklı katmanlardan meydana gelmektedir. Her katmanın öğrenme şekli, katsayısı ve öğrenme çıktıları farklıdır. Dolayısıyla DÖ, MÖ'ne göre daha fazla veriye ihtiyaç duyar. Bu olguların birbirleriyle olan ilişkileri Şekil 2.3.1'de verilmiştir.



Şekil 2.3.1. Makine öğrenmesi ve ilişkili alanları (Sindhu, 2020)

2.3.1. Denetimli Öğrenme

Denetimli Öğrenme MÖ'de kullanılan bir yöntemdir. Bu yöntemde kullanıcı veri kümelerini etiketleyerek kullanır. Veri seti içerisindeki verilerin (öznitelikler) karşılığı hangi verinin geldiği bu etiketleme yoluyla belirtilir ve öğrenme metoduna bu şekilde girdi verilir. Algoritma bu etiketli veriler üzerinden en uygun fonksiyonu bulmaya çalışır. Girdi ve çıktı arasındaki örüntüyü en iyi tanımlayacak olan fonksiyon daha sonra etiketlenmemiş bir veri geldiğinde aynı fonksiyonu uygulayarak, çıkışı tahmin etmeye çalışır. Denetimli öğrenme algoritması eğitim verilerini inceler ve yeni girdileri eşlemek için kullanabilecek bir fonksiyon üretir. Denetimli Öğrenme sınıflandırma ve regresyon problemlerine uygulanır. Şekil 2.3.1.1'de bu iki temel yöntem gösterilmiştir.



Şekil 2.3.1.1. Sınıflandırma ve regresyon (Candan H., 2021)

Regresyon

Regresyon bağımlı değişken (sonuç veya etiket) ile bir veya daha fazla bağımsız değişken arasındaki ilişkiyi tahmin etmeye yarayan istatistiksel bir durumdur. Bir regresyon probleminde, sürekli değerli bir çıktıyı tahmin etmek hedeflenmektedir. Regresyon analizinin en yaygın biçimi, verilerin dağılımını en yakın şekilde uyan çizginin bulunduğu doğrusal regresyondur (DR).

Belirli matematiksel nedenlerden dolayı (bkz. doğrusal regresyon), bu, bağımsız değişkenler belirli bir değer kümesini aldığı anda araştırmacının bağımlı değişkenin koşullu beklentisini (veya popülasyon ortalama değerini) tahmin etmesine olanak tanır.

Regresyon analizi tahmin yapabilmek için kullanılabilir. Bazı durumlarda bağımsız ve bağımlı değişkenler arasındaki nedensel ilişkileri ortaya çıkarmak için de kullanılabilir. Burada önemli olan nedensel ilişkilerin kullanılması ve modelin bu yönde matematiksel olarak ifade edilebilmesidir. Regresyon biçimleri şu şekildedir:

- Basit Doğrusal Regresyon
- Çoklu Doğrusal Regresyon
- Polinom Regresyon
- Mantıksal (Lojistik) Regresyon
- Karar Ağacı Regresyon
- Rastgele Orman Regresyon

Sınıflandırma

Sınıflandırma, veri kümesini farklı parametrelere dayalı olarak farklı kümelere ayırmayı hedefleyen bir algoritmadır. Girdi verilerini ayrık değerdeki çıktı verisine eşleyen problemlere uygulanmaktadır; yani çıktı verileri kategorik tiptedir. Sınıflandırmada model, eğitim verileri kullanılarak tamamen eğitilir ve daha sonra yeni görünmeyen veriler üzerinde tahmin yapmak için kullanılmadan önce test verileri üzerinde değerlendirilir.

Makine öğrenimi sınıflandırmasında iki tür öğrenci vardır: tembel ve istekli öğrenenler. İstekli öğrenenler, gelecekteki veri kümeleri hakkında herhangi bir tahminde bulunmadan önce ilk olarak eğitim veri kümesinden bir model oluşturan makine öğrenimi algoritmalarıdır. Eğitim sırasında ağırlıkları öğrenerek daha iyi bir genelleme yapma istekleri nedeniyle eğitim sürecinde daha fazla zaman harcarlar, ancak tahmin yapmak için daha az zamana ihtiyaç duyarlar. Çoğu makine öğrenimi algoritması hevesli öğrencilerdir ve aşağıda bazı örnekler verilmiştir:

- Mantıksal (Lojistik) Regresyon
- Destek Vektör Makineleri
- Karar Ağaçları
- Yapay Sinir Ağları

Tembel öğrenenler veya örnek tabanlı öğrenenler ise eğitim verilerinden hemen herhangi bir model oluşturmazlar ve tembellik yönü de buradan gelir. Sadece eğitim verilerini ezberlerler ve her tahmin yapılması gerektiğinde tüm eğitim verisinden en yakın komşuyu ararlar, bu da tahmin sırasında çok yavaş olmalarına neden olur. Bu türden bazı örnekler şunlardır:

- K-En Yakın Komşu
- Vakaya veya olaya bağlı mantık-akıl yürütme

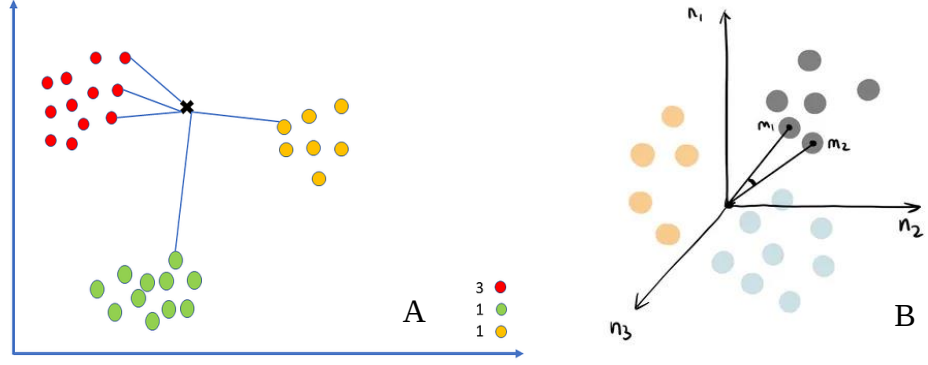
2.3.2. KNN (K-en yakın komşular)

K-en yakın komşu (KNN), sınıflandırma ve regresyon problemlerini çözmek için yaygın olarak kullanılan basit ama etkili bir parametrik olmayan denetimli öğrenme algoritmasıdır. Hem sınıflandırma hem de regresyon amaçlı kullanılabilir. Her iki durumda da giriş, veri setindeki en yakın 'k' eğitim örneğinden oluşur. Çıktı, KNN'nin sınıflandırma veya regresyon için kullanılıp kullanılmadığına göre farklılık gösterir. KNN sınıflandırmasında çıktı, bir sınıf üyeliğidir. Bir nesne, komşularının oylarıyla sınıflandırılır; en yakın k komşusu arasında en yaygın olan sınıfa atanır (k pozitif bir tam sayıdır ve genellikle küçük bir değerdedir). Eğer $k = 1$ ise, nesne en yakın komşusunun sınıfına atanır.

KNN regresyonunda ise çıktı, nesnenin özellik değeridir ve bu değer, k en yakın komşunun değerlerinin ortalaması olarak hesaplanır. Eğer $k = 1$ ise, çıktı en yakın komşunun değeri olur. KNN, veri noktasını en yakın 'k' komşusuna göre sınıflandırma prensibiyle çalışır. Algoritma, mevcut tüm verileri depolar ve yeni bir veri noktası için bu veriler arasından mesafe açısından en yakın k veri noktasını bularak tahmin yapar. Bu tahmin, çözülmesi gereken probleme bağlı olarak, k en yakın komşunun çoğunluk sınıfına veya değerlerinin ortalamasına dayanır.

KNN'nin uygulanması kolaydır ve nispeten düşük hesaplama maliyetlerine sahiptir; bu da onu görüntü ve konuşma tanıma, tıbbi teşhis, finans ve birçok diğer alandaki

uygulamalar için cazip bir seçenek yapar. Ancak, doğruluğu 'k' seçiminden ve kullanılan mesafe ölçütünden etkilendiği için dikkatli olunmalıdır.



Şekil 2.3.2.1. KNN algoritmaları. **A)** İki boyutlu düzlemde KNN (Mattaparthi, 2023)
B) Üç boyutlu düzlemde KNN (Gong, 2022)

Yeni eklenen bir verinin çıktısının hangi etikete dahil olması gerekliliği veri noktasının diğer kümelere olan uzaklığı ve komşu sayısı belirler. Farklı düzlemlerdeki KNN sınıflandırması Şekil 2.3.2.1’de belirtilmiştir.

KNN’de komşu sayısı belirlenirken farklı mesafe metrikleri kullanılır.

Öklid Mesafesi: Düzlemde bulunan iki veri noktası arasındaki en kısa mesafeyi gösteren uzunluktur.

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2.2)$$

Burada ‘n’ boyut sayısını temsil ederken, x_i ve y_i veri noktalarını temsil ederler.

Manhattan Mesafesi: Bütün düzlemlerdeki noktaların mutlak değerlerinin farklarının toplamı olarak ifade edilir.

$$d(x, y) = \sum_{i=1}^n |x_i - y_i| \quad (2.3)$$

Burada da x_i ve y_i veri noktalarını temsil ederler.

Minkowski Mesafesi: Öklid ve Manhattan mesafelerinin genelleştirilmiş bir formudur.

$$d(x, y) = \left(\sum_{k=1}^n |x_k - y_k|^r \right)^{\frac{1}{r}} \quad (2.4)$$

Burada da x_k ve y_k veri noktalarını temsil ederler.

Kosinüs benzerliği: Bir iç çarpım uzayının sıfır olmayan iki vektörü arasındaki benzerliği ölçer. Genellikle metin analizlerinde belge benzerliğini ölçmede kullanılır.

Formülü: Kosinüs mesafesi = $1 - \cos \theta$

$$\frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \cdot \sqrt{\sum_{i=1}^n B_i^2}} \quad (2.5)$$

Burada A_i ve B_i , A ve B vektörlerinin i .ci bileşenleridir.

KNN'deki komşuların sayısını tanımlamak için belirtilen 'k' değerini tahmin etmenin uygun bir yöntemi yoktur. Hiçbir yöntem genelleme olarak kullanılamaz ancak bazı yöntemler 'k' değeri seçiminde yardımcı olabilir.

Sınıflandırmalarda bağlardan kurtulmak ve bir sınıfın baskınlığını engellemek için 'k' tek sayı seçilmelidir. 'k' değeri ne kadar büyük olursa doğruluk da o kadar yüksek olur. Farklı 'k' değerlerini test etmek ve modelin performansını en üst düzeye çıkaranı seçmek için çapraz doğrulama gibi tekniklerden yararlanılır. "N-kuralı karekökü" olarak adlandırılan eğitim veri setindeki toplam veri noktası sayısının karekökü alınarak

belirlenen k sayısı yaygın olarak kullanılan ve doğruya yakın bir sonuç veren bir hesaplama tekniğidir. Yine de sorunun bağlamı ve işlemi yapan kişinin deneyimine bağlı olarak farklı teknikler denenmeli ve yorumlanmalıdır.

2.3.3. DVM-Destek Vektör Makineleri

Denetimli öğrenmenin bir modeli olan Destek Vektör Makineleri (DVM) hem doğrusal hem de doğrusal olmayan problemlere, iki sınıf arasında bir boşluk yaratacak şekilde oluşturulan hiper düzlemleri bulmak üzere geliştirilen bir algoritma türüdür. DVM hem regresyon hem de sınıflandırma problemleri için kullanılabilir ancak genel olarak sınıflandırma problemlerinde en iyi şekilde çalışırlar. Regresyon modeli olarak kullanımına ‘Destek Vektör Regresyon’ (DVR) denir. DVR, verilerin doğrusal olmayan özelliklerini yakalamak için çekirdek işlev yöntemlerini kullanarak bağımlı ve bağımsız değişkenler arasındaki ilişkiyi modellemek için kullanılır. Bu tezde hem DVM hem de DVR kullanılmıştır. Destek Vektör Makineleri algoritmaları iki çeşit olarak tanımlanır:

Doğrusal DVM: Veri noktaları doğrusal bir çizgiyle ayrılabilirse ‘Doğrusal DVM’ kullanılır.

Doğrusal olmayan DVM: Veriler doğrusal olarak ayrılamadığında ‘Doğrusal Olmayan DVM’ kullanılır; bu, veri noktalarının bir sınıfa ayrılamadığı anlamına gelir. Birçok uygulamada doğrusal olarak ayrılabilir veri noktaları elde edilemediğinden bu yöntem başvurulmaktadır. Bu yöntem ‘çekirdek’ teknikleri kullanarak veriyi istenilen şekilde sınıflandırmaya yarar.

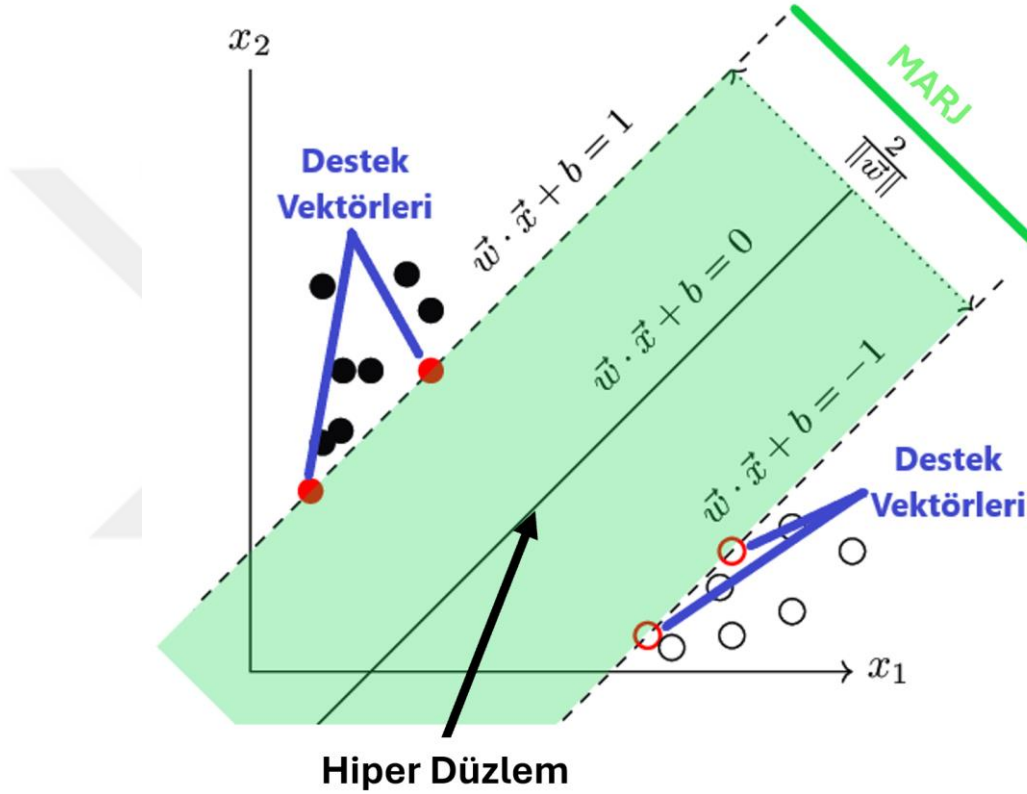
DVM’nin nasıl çalıştığına bakmadan önce bazı kavramları anlamak önemlidir.

Marj: Marj, hiper düzlem ve destek vektörler arasındaki boşluk tanımıdır.

Hiper düzlem: Hiper düzlemler, veri noktalarının sınıflandırılmasına yardımcı olan karar sınırlarıdır.

Destek Vektörler: Hiper düzlemin üzerinde veya ona en yakın olan ve hiper düzlemin konumunu etkileyen veri noktalarıdır.

Çekirdek İşlevi (Kernel Fonksiyonu): Hiper düzlemin şeklini ve karar sınırını belirlemek için kullanılan işlevlerdir.



Şekil 2.3.3.1. İki sınıftan örneklerle eğitilmiş bir DVM için maksimum marj hiper düzlemi (Jiaying ve arkadaşları 2018’den değiştirilerek alınmıştır)

Marjın sınırlarını belirleyen örnekler destek vektörleridir.

İki sınıf arasındaki maksimum mesafede (marj) olması istenen hiper düzlem, en uygun hiper düzlem olarak tanımlanır. Bu hiper düzlem, veri noktalarını sınıflandırmada en uygun performansı sağlar. Marj adı verilen bu boşluk, sınıflar arasındaki uzaklığı temsil eder ve en geniş marjın, modelin genelleme yeteneğini artırır.

Birden fazla hiper düzlem olabilir ancak marjın maksimum olduğu hiper düzlemin en uygun hiper düzlem olduğunu bulunabilir. DVM'nin temel amacı, veri noktalarını yüksek hassasiyetle sınıflandırabilen hiper düzlemleri bulmaktır.

Destek vektörler, hiper düzlem üzerindeki sınıf sınırlarına en yakın veri noktalarını temsil eder. DVM'nin gücü, bu destek vektörlerin belirlenmesi ve sınıflandırma sürecindeki etkisiyle ortaya çıkar. Optimizasyon süreci, marjını en üst düzeye çıkaran hiper düzlemi bulurken destek vektörleri de belirler (Karabudak, 2024).

Şekil 2.3.3.1'deki örneğe bakıldığında oluşturulan hiper düzlemin her iki taraftaki sınır destek vektörlerden uzaklığı en fazla olacak biçimde optimize edildiği görülmektedir. Matematiksel ifade ile:

$$\hat{y} = \begin{cases} 0 & \text{eğer } w^T \cdot x + b < 0 \\ 1 & \text{eğer } w^T \cdot x + b \geq 0 \end{cases} \quad (2.6)$$

w^T : Ağırlık vektörü transpozu

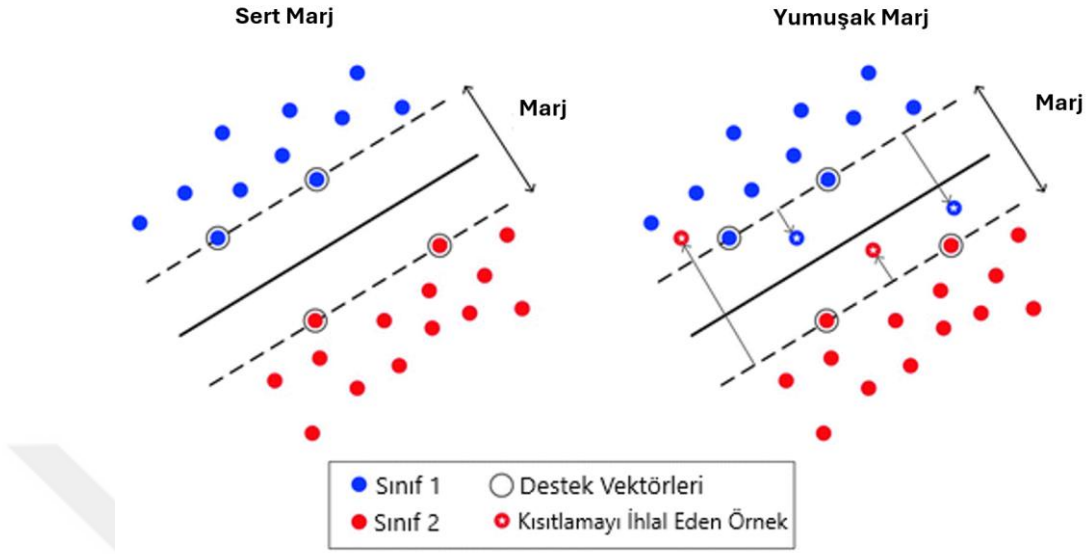
x : Girdi vektörü

b : Sapma

$\hat{y}$: Hedef etiket(ler)

Bir değer için sonuç 0'dan küçükse hiper düzlemin altında kalan beyaz noktalar kümesine dahil olacaktır. Tam tersi, sonuç 0'dan büyükse hiper düzlemin üstünde kalan siyah noktalar kümesine dahil olacaktır.

DVM'de marj her zaman Şekil 2.3.3.1'deki gibi net olmayabilir. Bazen destek vektör noktaları marjın içerisine girebilir ve diğer sınıfa dahil olabilirler. Buna 'yumuşak marj' denilir. Eğer veriler doğrusal olarak ayrılabiliriyorsa, aykırı değerlere karşı çok duyarlıdır. Bu durumda olan marjlara da 'sert marj' denilir. Gerçek hayat uygulamalarında çoğunlukla doğrusal olarak verileri ayırmak mümkün olmadığından yumuşak marj kullanılır.



Şekil 2.3.3.2. Sert ve yumuşak marj örnekleri (Kaushal ve arkadaşları 2021'den değiştirilerek alınmıştır)

Yumuşak marj, bazı marj ihlallerinin; yanlış sınıflandırmaların verilerin çok iyi ayrılmadığı durumlarda devreye girerek, esneklik sağlar. Genelde gürültü veya aykırı değerler olduğunda kullanılır. Bu şekilde yanlış sınıflandırmanın önüne geçilmiş olur.

Burada kullanılan bazı parametreler mevcuttur. Eğer DVM'de yumuşak marj kullanılırsa bir düzenleme parametresi olarak 'C' adı verilen bir faktör bunu düzenler. Bu faktör boşluğu olabildiğince büyütme ile veri noktalarının gruplandırmadaki hatalarını azaltmak arasında bir denge bulmaya yardımcı olur. Şekil 2.3.3.2'de bazı veri noktalarının marj boşluğunun içerisinde ve ayrıca hiper düzlemin yanlış tarafında olduğu gözükmemektedir. Bu ihlaller yumuşak marjin ayarında izin verilmektedir. Buna rağmen marjın kenarındaki noktalar önemini korumakta ve destek vektörleri olmaya devam etmektedirler.

Matematiksel olarak ifade edilmek istenirse:

$$w^T \cdot x_i + b \geq 1 - \xi_i \quad (2.7)$$

Denklem (2.7)'de görüldüğü üzere marjı ve hiper düzlemi tanımlayan denkleminin sağ tarafına marj ihlalini temsil eden değişken, ξ , tanımlanmıştır. $(1 - \xi_i)$ olarak tanımlanan terim her veri noktası için gerekli olan en az marjı temsil eder.

Yumuşak marj, marj maksimizasyonu ile marj ihlallerini için bir ceza terimini birleştirir ve aşağıda belirtilen formül çıktı değerini en aza indirmeyi hedefler.

$$\frac{1}{2} ||w||^2 + C \sum_{i=1} N_{i=1} \xi_i \quad (2.8)$$

Burada “ C ”, DVM’de kullanılan düzenleyici bir parametredir; marj genişliği ve sınıflandırma hatasının toleransı arasında denge kurar.

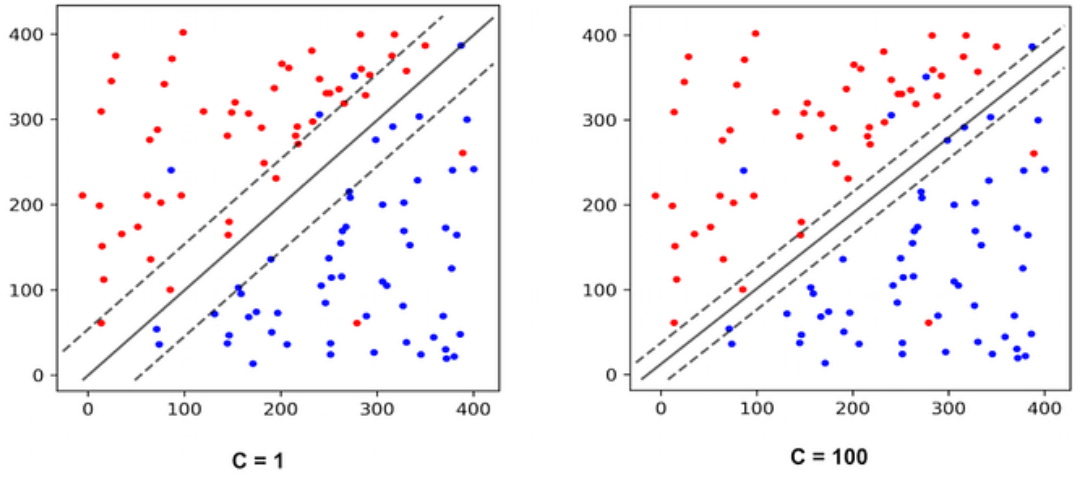
Daha yüksek bir C değeri, yanlış sınıflandırmalara izin vere bile daha geniş bir marja öncelik verir. Daha düşük bir C değeri, daha fazla marj ihlaline karşılık oldukça yumuşak bir karar çizgisi belirler.

Yumuşak Marj Artıları

Yumuşak marj aykırı değerleri ve gürültülü verileri daha etkili bir şekilde işler. Bu görünmeyen verilere daha iyi genelleme yapan daha sağlam bir karar sınırıyla sonuçlanır. Sert marjdan farklı olarak, yumuşak marj doğrusal olmayan ayrılabilen verilerle, çekirdek işlevlerini (fonksiyonlarını) kullanarak daha yüksek boyutlu bir alana örtülü olarak eşleyerek işleyebilir. Bu karmaşık sınırların daha verimli bir biçimde yakalanmasını sağlar.

Yumuşak Marj Eksileri

Yumuşak marjın performansı düzenleyici parametre olan ' C ' ye oldukça bağımlıdır. Bu parametreyi seçmek bazen oldukça zor olabilir; özellikle çok büyük veri setlerinde bu çok zaman alıcı ve maliyetli bir uğraştır. Bazı hesaplamalarda eğer ' C ' değeri çok yüksek olursa model aşırı öğrenme olabilir. Bu durumda gereğinden fazlaca marj ihlali izni olduğu görülür. Farklı ' C ' değerleri için genişlik örnekleri Şekil 2.3.3.3'te verilmiştir.



Şekil 2.3.3.3. Farklı ' C ' değerleri için marj büyüklüğü örneği (Shrimali, 2018)

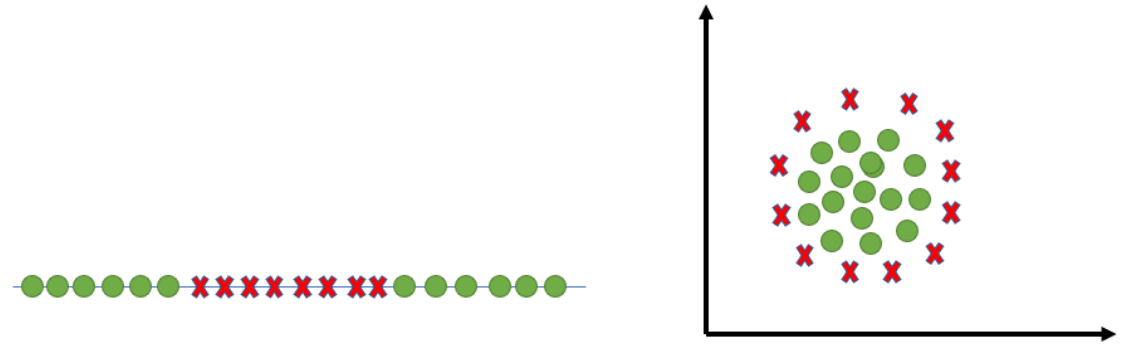
Seçilen marj yöntemleri için belirlenen kriterler Çizelge 2.3.3.1'de verilmiştir.

Çizelge 2.3.3.1. Sert marj ve yumuşak marj karşılaştırması

Kriter	Sert Marj	Yumuşak Marj
Amaç Fonksiyonu	Maksimum marj	Maksimum marj, marj ihlallerinin en aza indirgenmesi
Gürültüyü Yönetme	Gürültüye Duyarlı, kusursuz bir biçimde doğrusal olarak ayrılabilen veriye ihtiyaç duyar.	Sağlamdır, gürültüden etkilenmez, gürültülü verileri işler.
Düzenleme	Düzenleme parametresi yok	Düzenleştirme parametresi C tarafından kontrol edilir.
Karmaşıklık	Basit, hesaplama için verimli	Hesaplama için daha fazla kaynağa ihtiyaç duyar.

Çekirdek işlemleri (Kernel Trick)

Bazı veri setlerini, Şekil 2.3.3.3'teki gösterildiği şekilde tek bir çizgi içeren hiper düzlem ile sınıflandırma yapmak mümkün değildir. Buradaki her bir veri noktası iki boyutlu düzlemde ideal bir biçimde ayrılmamıştır.



Şekil 2.3.3.4. Doğrusal şekilde olmayan veri dağılımı (Karabudak, 2024)

Eğer veriler Şekil 2.3.3.4'teki gibi dağılmışsa bazı çekirdek fonksiyonları (işlemleri) kullanarak boyut artırma yoluna gidilir. Burada bazı ikinci dereceden formüller kullanılarak, karar sınırı belirlenmeye çalışılır. Bu fonksiyonlara çekirdek fonksiyonu ya da çekirdek işlemleri denir.

Polinom Çekirdek

Veri dağılımlarının iki boyutta çözümlenemediği zamanlarda, problemleri çözmek için üç veya daha fazla boyuta taşıyarak yapılan işlemlerdir.

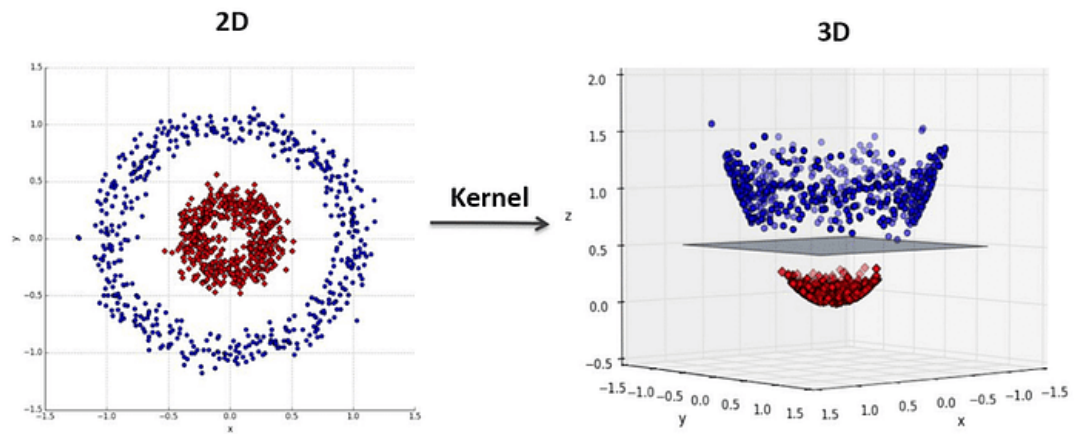
$$f(X1, X2) = (X1^T \cdot X2 + c)^d \quad (2.9)$$

Burada ‘d’ polinomdaki boyutu ifade eder. $X1$ ve $X2$ ise veri tabanındaki veri noktalarıdır. Eğer ‘c’ sıfır ise çekirdek homojendir.

Fonksiyonun çıktısını ‘Y’ olarak ele alırsak;

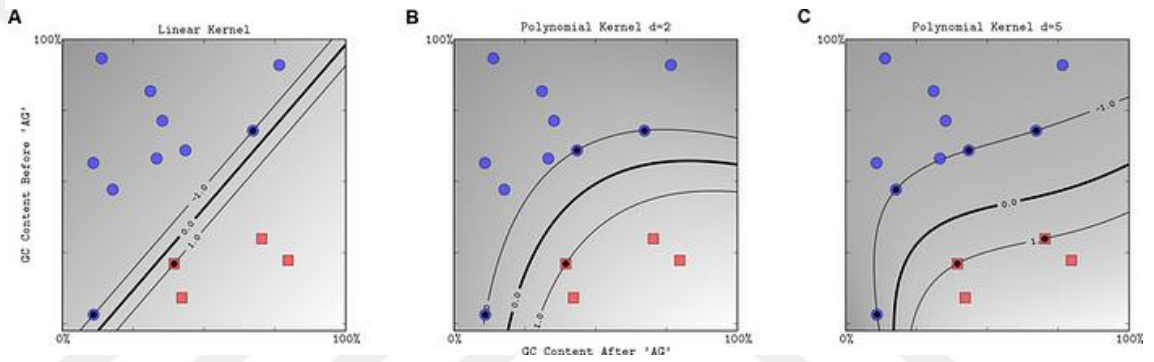
$$X1^T \cdot X2 = \begin{bmatrix} X1 \\ X2 \end{bmatrix} \cdot [X1 \ X2] = \begin{bmatrix} X1^2 & X1 \cdot X2 \\ X1 \cdot X2 & X2^2 \end{bmatrix} \quad (2.10)$$

Önceden iki boyutla değerlendirilen denklem, $X1^2, X2^2$ ve $X1 \cdot X2$ bilinmezleri ile beş boyuta yükseltilmiştir. İki boyutlu düzlemde belirlenemeyen hiper düzlemin üç boyutlu düzlemde belirlenebildiği Şekil 2.3.3.5’te gösterilmiştir.



Şekil 2.3.3.5. İki boyutlu bir dağılımın üç boyutlu hale getirilip sınıflara ayrılması (Mansour ve arkadaşları, 2023)

İşlemi uygularken belirli bir ‘ d ’ derecesi kullanılmaz. Veri dağılımlarına, çeşidine ve büyüklüğüne göre bu değişkenlik gösterebilir. Şekil 2.3.3.6’da B ile gösterilen veri sınıflandırma grafiğinde $d = 2$ alınmıştır. Bu değerde üç adet mavi çember ve bir adet kırmızı kare destek vektörü olarak tanımlanmıştır. Burada verilen örnekteki şekilde yeterli gibi görünse de daha fazla veri noktası olan problemlerde bu yeterli olmayabilir. Şekil 2.3.3.6’da C ile gösterilen veri sınıflandırma grafiğinde ise $d = 5$ alınmıştır ve destek vektör sayısı arttırılmıştır. Böylece daha az hata ile daha kesin bir ayırım yapılmış ve hiper düzlem daha net belirlenmiştir.



Şekil 2.3.3.6. Doğrusal DVM (A), polinom çekirdek işlemli ($d=2$)(B), ve polinom çekirdek işlemli ($d=5$)(C) sınıflandırma (Ben-Hur ve arkadaşları, 2008)

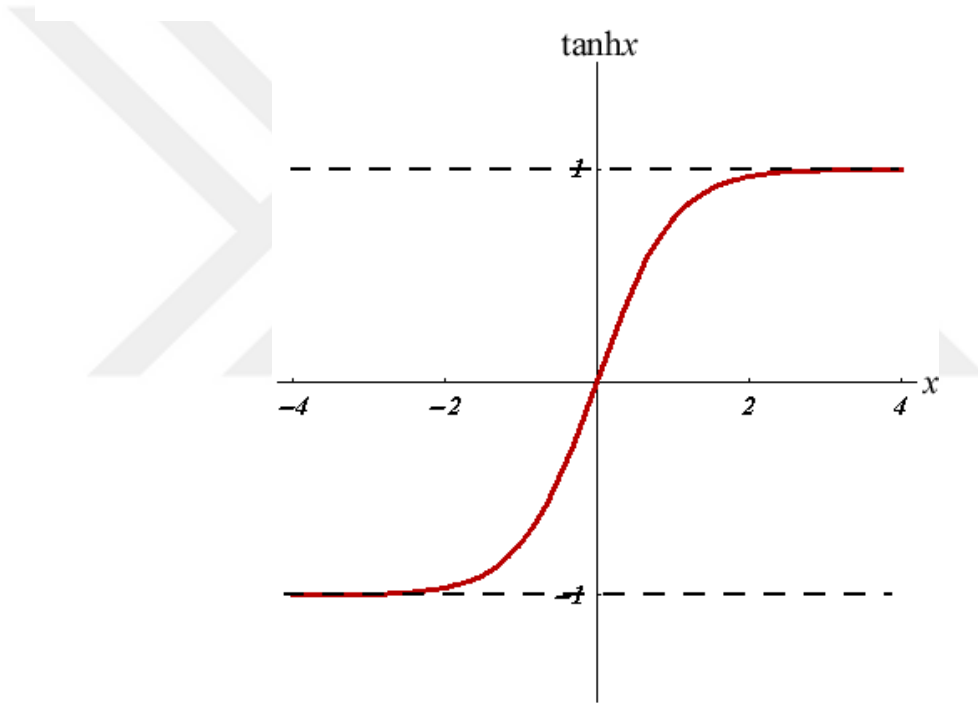
‘ d ’ katsayısını belirlerken modelin yetersiz öğrenmede veya aşırı öğrenmede olup olmadığına dikkat edilmelidir. Model yetersiz öğrenmede ise, katsayı arttırılmalı, eğer model aşırı öğrenmede ise, katsayı azaltılmalıdır.

Sigmoid Çekirdek

Sigmoid çekirdeği DVM ve diğer makine öğrenimi algoritmalarında yaygın olarak kullanılan, sabit olmayan bir çekirdek işlevidir. Diğer çekirdek türlerinin aksine belirli veri türleri için uygun kılan benzersiz özelliklere sahiptir.

$$f(x, y) = \tan(ax^T y + c) \quad (2.11)$$

Denklem (2.11)'de verilen x ve y giriş veri noktalarıdır, α ise bir ölçeklendirme parametresidir ve c sabittir. Hiperbolik tanjant fonksiyonu, sinir ağlarında kullanılan etkinleştirme fonksiyonuna benzer şekilde çekirdek fonksiyonunun çıktısının $[-1,1]$ aralığına düşmesini sağlar. Bu yöntem nöral ağlar için vekil bir kullanım olarak da nitelendirilir. Hiperbolik tanjant fonksiyonu, genel tabir ile 'Sigmoid İşlevi' olarak bilinen genel bir fonksiyon kümesinin alt dalıdır. Şekil 2.3.3.7'de hiperbolik tanjant fonksiyon grafiği gösterilmiş olsa da 'mantıksal fonksiyon' biçimi regresyon modellerinde kullanılır.



Şekil 2.3.3.7. Hiperbolik tanjant fonksiyon grafiği (Manton, 2023'ten değiştirilerek alınmıştır).

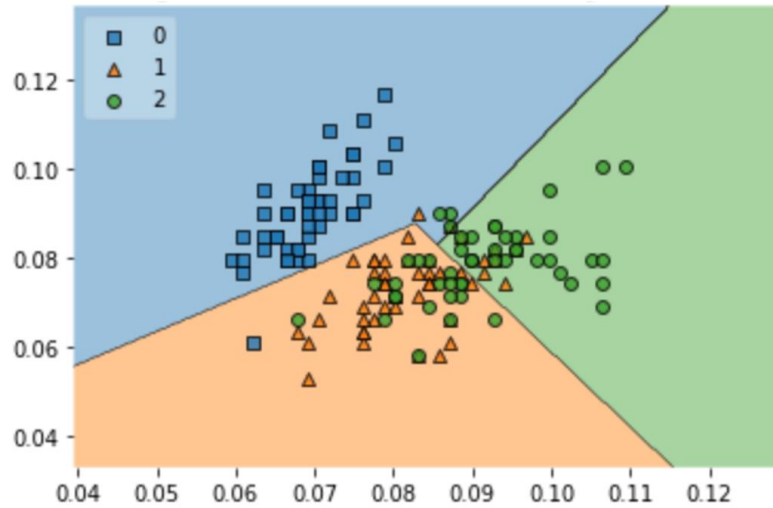
Sigmoid çekirdeğin diğer çekirdeklerden ayrıldığı başlıca noktalar şunlardır:

Doğrusal Olmama: Diğer çekirdek işlevleri gibi sigmoid çekirdek de DVM'lerin veriler içindeki doğrusal olmayan ilişkileri yakalamasını sağlar. Hiperbolik tanjant fonksiyonunu uygulayarak karar sınırına doğrusal olmamayı getirerek daha esnek ve anlamlı modellere olanak tanır.

Sinir Ağlarına Benzerlik: Sigmoid çekirdeğin şekli, yapay sinir ağlarında kullanılan etkinleştirme fonksiyonuna benzer. Bu benzerlik, özellikle sinir ağlarının da iyi performans gösterebildiği senaryolarda, sigmoid çekirdeğin karmaşık veri modellerini etkili bir şekilde modelleyebildiğini göstermektedir.

Metin Sınıflandırmaya Uygunluk: Sigmoid çekirdeği, metin sınıflandırma görevlerinde özel kullanım alanı olan bir işlevdir. Metin verileri genellikle karmaşık, doğrusal olmayan ilişkiler sergiler ve sigmoid çekirdeğin bu tür ilişkileri yakalama yeteneği, onu duygu analizi veya belge sınıflandırması gibi görevler için uygun bir seçim haline getirir.

Sınırları İyi Tanımlanmış Veriler: Verinin iyi tanımlanmış sınıf sınırlarına sahip olduğu durumlarda sigmoid çekirdeği çok iyi bir performans gösterebilir. Veriler nispeten ayrılabilir olduğunda ve karar sınırı aşırı karmaşık olmadığında iyi çalışma eğilimindedir.



Şekil 2.3.3.8. Sigmoid çekirdeğe sahip bir DVM tarafından oluşturulmuş karar sınırları

Şekil 2.3.3.8’de üç farklı sınıfın iki ayrı öznelik üzerinden sınıflandırma yapılmıştır. Görüldüğü üzere sınırlar doğrusal değildir.

Radyal Tabanlı Çekirdek (RTÇ, Gaussian RBF)

DVM sınıflandırma problemlerinin çözmek için kullanılan en yaygın çekirdek işlevidir. Literatürde yer almış birçok radyal tabanlı işlev olmakla beraber, DVM için en yaygın Gauss metodudur. Gauss dağılımına oldukça benzediğinden bu şekilde kullanılmaktadır. Veri seti çok büyük değilse genelde bu çekirdek işlevi tercih edilir.

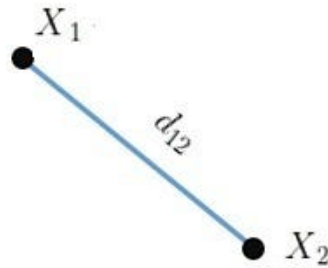
$$K(X_1X_2) = e^{-\frac{\|X_1-X_2\|^2}{2\sigma^2}} \quad (2.12)$$

Formül (2.12)'de gösterilen RTÇ denkleminde $\|X_1X_2\|$ iki farklı öz nitelik uzayında X_1 ve X_2 arasındaki Öklid Uzaklığıdır ve sigma (σ) çekirdek ağırlığını belirleyen RTÇ çekirdek parametresidir. DVM'de doğru sınıflandırma yapabilmek için sigmanın iyi ayarlanması gerekmektedir. Sigma parametresinin varsayılan değeri 1'dir.

RTÇ çekirdek işlevi formül (2.13)'de bahsi geçen X_1 ve X_2 veri noktaları için benzerliklerini veya birbirlerine ne kadar yakın olduklarını hesaplar. ' σ ' varyansı temsil eder ve her ne kadar parametre genellemesi yapılsa da bir hiper parametre olarak tanımlanır.

Boşluktaki X_1 ve X_2 noktalarının arasındaki uzunluğa d_{12} dersek;

$$d_{12} = \|X_1 - X_2\|^2 \quad (2.13)$$

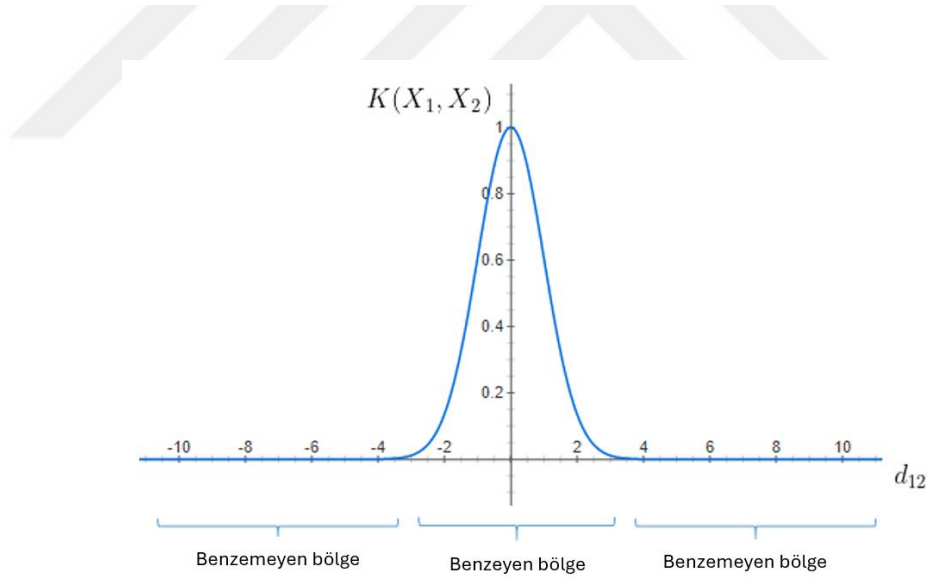


Şekil 2.3.3.9. Boşluktaki iki noktanın birbirlerine olan uzaklıkları (Sreenivasa, 2020).

RTÇ çekirdeğinin alabileceği en yüksek değer 1'dir ve bu ancak $||X_1 - X_2||^2 = 0$ olduğunda gerçekleşir. Noktalar aynı olduğunda aralarındaki mesafe yoktur ve birbirlerine benzerler. Bu durum Şekil 2.3.3.9'daki d_{12} uzaklığının sıfıra eşit olduğu durumda gerçekleşir.

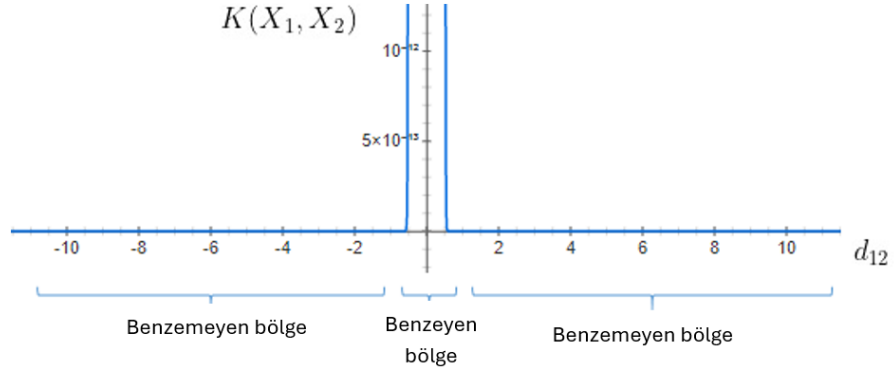
Noktalar büyük bir mesafe ile birbirlerinden ayrıldığında çekirdek değeri 1'den küçük ve 0'a yakındır, bu da noktaların farklı olduğu anlamına gelir. Mesafe, farklılığa eşdeğer olarak düşünülebilir çünkü noktalar arasındaki mesafe arttıkça benzerliklerinin azaldığı gözlemlenebilir (Sreenivasa, 2020).

Hangi noktaların benzer olarak değerlendirilmesi gerektiğine karar vermek için doğru sigma (σ) değerini bulmak önemlidir ve bu çözülmek istenilen probleme göre değişiklik gösterebilir.



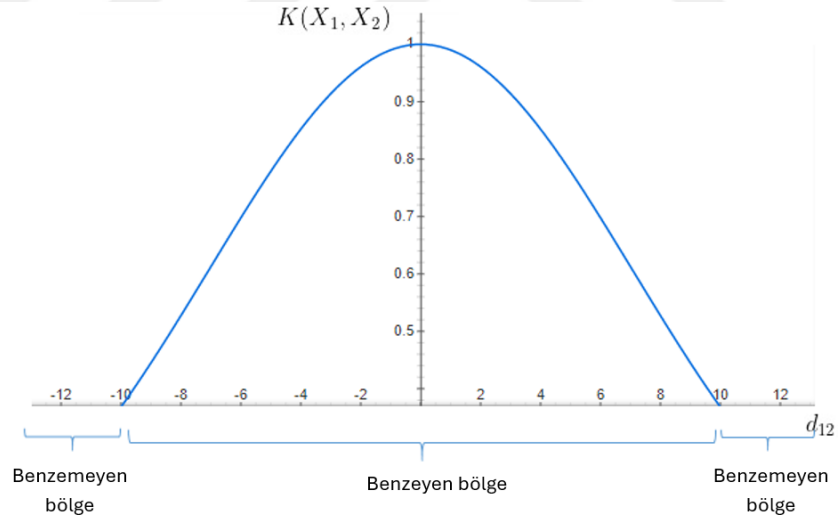
Şekil 2.3.3.10. $\sigma = 1$ için gerçekleşen RTÇ grafiği (Sreenivasa, 2020'den değiştirilerek alınmıştır)

Şekil 2.3.3.10'da görüldüğü gibi mesafe '0' olduğunda benzerlik '1' olurken, dört birim uzaklaşıldığında benzerlik '0'a geliyor.



Şekil 2.3.3.11. $\sigma = 0.1$ için gerçekleşen RTC grafiği (Sreenivasa, 2020'den değiştirilerek alınmıştır)

Şekil 2.3.3.11'de görüldüğü gibi mesafesi 0.2'den küçük olan noktalar birbirine benzerdir. Mesafe 0.2'den fazla olan noktalar birbirlerine benzemeyen noktalar olarak tanımlanırlar.



Şekil 2.3.3.12. $\sigma = 10$ için gerçekleşen RTC grafiği (Sreenivasa, 2020'den değiştirilerek alınmıştır)

Şekil 2.3.3.12'de görüldüğü gibi sigma değeri arttıkça benzerlik alanı da oldukça genişlemiştir. Merkezden 10 birim ve daha uzakta kalan yerler benzemeyen alan olarak tanımlanmıştır.

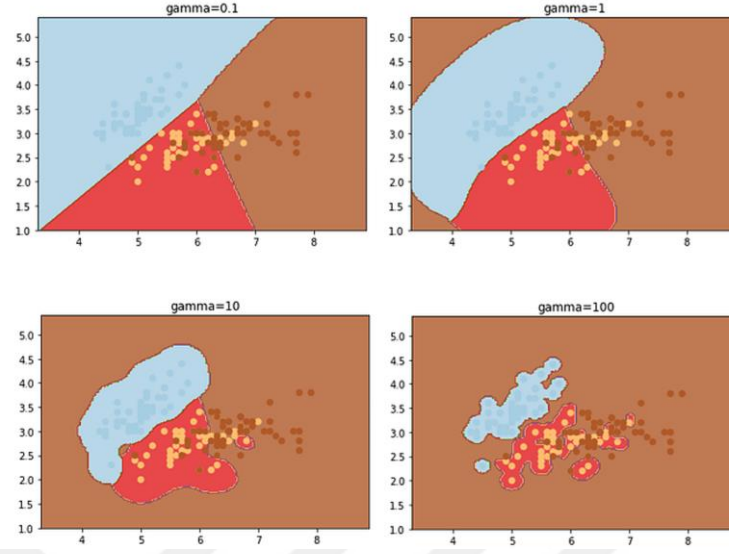
Benzerlik bölgesi Şekil 2.3.3.10, Şekil 2.3.3.11 ve Şekil 2.3.3.12’de olduğu gibi σ değiştikçe değiştiği açıktır. Belirli bir veri kümesi için doğru sigmayı (σ) bulmak önemlidir. Izgara Arama Çapraz Doğrulama (Grid Search Cross Validation) ve Rastgele Arama Çapraz Doğrulama (Random Search Cross Validation) gibi hiper parametre ayarlama teknikleri kullanılarak sigma değeri bulunabilir.

RTÇ, DVM’nin tüm veri kümesini değil, yalnızca eğitim sırasında destek vektörlerini saklaması gerektiğinden alan karmaşıklığı sorununun üstesinden gelir. Bu noktada K-En Yakın Komşular Algoritması ile de benzerlik göstermektedir.

RTÇ’de tanımlanan bir başka hiper parametre ise gamma parametresidir.

$$\gamma = \frac{1}{(2\sigma)^2} \quad (2.14)$$

Gamma değeri ne kadar düşükse, dağılım o kadar geniş olur. C hiper parametresinde olduğu gibi, eğer model aşırı öğrenme gösteriyorsa gamma değeri azaltılmalı, model yetersiz öğrenme gösteriyorsa gamma değeri artırılmalıdır. Gamma değerinin öğrenme modeli üzerindeki performansı Şekil 2.3.3.13’te gösterilmiştir.



Şekil 2.3.3.13. Üç sınıflı iki özellikli verilerin RTÇ işlevinde farklı gamma değerleri kullanılarak sınıflandırılması. (Akça, 2020)

2.3.4. Aşırı Öğrenme (Overfitting) ve Yetersiz Öğrenme (Underfitting)

MÖ’de her ne kadar uygun model ve algoritma seçilmiş olursa olsun, modelin başarısını değerlendirirken bazı sorunlar ortaya çıkmaktadır. Yaygın olarak bu sorun eğitim verisi üzerinde değil, test verisinin düşük performans vermesi ile fark edilir. Modeli oluştururken belirlenen parametreler bazen aşırı öğrenme ya da yetersiz öğrenmeye sebebiyet verebilirler.

Aşırı Öğrenme

Eğer model, eğitim için kullanılan veri setine aşırı odaklanıp ezberlemeye başlarsa ya da eğitim seti homojense, aşırı öğrenme riski artar. Bu tür bir model, eğitim setinde yüksek bir başarı elde etse de test verisiyle karşılaştığında çok düşük bir performans sergiler; çünkü model eğitim setindeki örnekleri ezberlemiş ve test verisinde de aynı örnekleri aramaktadır. En ufak bir farklılıkta, ezberlenen örnekler bulunamayacağı için test verisinde çok kötü tahmin sonuçları alınabilir. Aşırı öğrenme sorunu olan modellerde yüksek varyans ve düşük sapma görülür.

Bu genelde modelin çok karmaşık olması nedeniyle verilerdeki gürültüyü veya rastgele dalgalanmaları yakalamaya başlaması ve bu nedenle modelin daha önce karşılaşmadığı yeni verilere genelleme yaparken kötü performans göstermesi olarak da tanımlanabilir. (Taraflı, 2023)

Aşırı Öğrenmeyi engellemek adına çeşitli tekniklere başvurulur.

Düzenleme (Regularization): Modelin karmaşıklığını azaltarak aşırı öğrenmeyi önlemeye yardımcı olan bir tekniktir. L1 (Lasso) ve L2 (Ridge) düzenleme gibi çeşitli yöntemler vardır. L1 düzenlemesi, modeldeki bazı özelliklerin ağırlıklarını sıfıra yaklaştırarak önemsiz özelliklerin etkisini ortadan kaldırır ve böylece modelin genelleme yeteneğini artırır. L2 düzenlemesi ise tüm özelliklerin katsayılarını küçülterek hepsinin modele katkıda bulunmasını sağlar.

Bagging (bootstrap aggregating): Modelin genelleme yeteneğini artırmak için, rastgele örnekleme oluşturulan yeni veri kümeleri üzerinde birden fazla temel öğrenici kullanarak eğitim yapar ve sonuçları birleştirir. Bu yöntem, özellikle karar ağaçları gibi yüksek varyansa sahip modellerde etkilidir. Rassal orman (random forest) bu tekniğin en bilinen örneklerinden biridir.

Erken durdurma (early stopping): Eğitim sürecini, test hatası artmaya başladığında durdurarak aşırı öğrenmeyi önleyen bir tekniktir. Bazı uygulamalarda yinelemeli eğitim sırasında, ilk başlarda modelin performansı artar, ancak yinelemeler devam ettikçe performans düşmeye başlayabilir. Modelin aşırı yüklenmeden eğitim tekrarını durdurmak, veri kümesinin aşırı uyum yapmasını önler ve modelin daha önce karşılaşmadığı verilere karşı kötü performans göstermesini engeller.

Çapraz doğrulama (cross-validation): Veri setini birden fazla parçaya bölerek her bir parçayı sırayla test seti olarak kullanır ve modelin başarımını bu şekilde değerlendirir. Bu yöntem, modelin farklı veri alt kümeleri üzerindeki performansını ölçer ve aşırı öğrenme riskini azaltır. K-katlı çapraz doğrulama en yaygın kullanılan yöntemlerdendir.

Daha fazla veri eklemek: Örnek sayısını artırmak verideki hedef ile öznitelik arasındaki ilişkiyi çok daha rahat anlayabilmeyi sağlamaktadır. Daha fazla veriye sahip olan yöntemin, daha fazla veri türünü dikkate alarak daha iyi genelleme olasılığı daha yüksektir.

Birleştirmek (Ensembling): Farklı MÖ modellerini bir arada kullanılmasını sağlar. Daha karmaşık örnekler aşırı öğrenmeyi engellemeye yardımcı olur.

Öznitelik sayısını azaltmak: Birbirleriyle yüksek korelasyon gösteren öznitelikler, bulundukları veri setinde aynı bilgiyi taşırlar ve yanlılık gösterirler. Dolayısıyla yüksek korelasyon gösteren özniteliklerin ayıklanması aşırı öğrenmeyi engelleyebilir.

Yetersiz Öğrenme

Yetersiz öğrenme, matematiksel modelin, verinin temel yapısını yakalayamadığı durumdur. Yetersiz öğrenmede model, karmaşık veri noktalarını yakalamak için oldukça basittir. Temel örüntü yakalanamaz ve bu nedenle kötü bir performans çıktısı oluşur. Yetersiz öğrenmede hem eğitim hem de test veri setinde hata oranı yüksektir. Yetersiz öğrenmeyi gidermek için çeşitli tekniklere başvurulur.

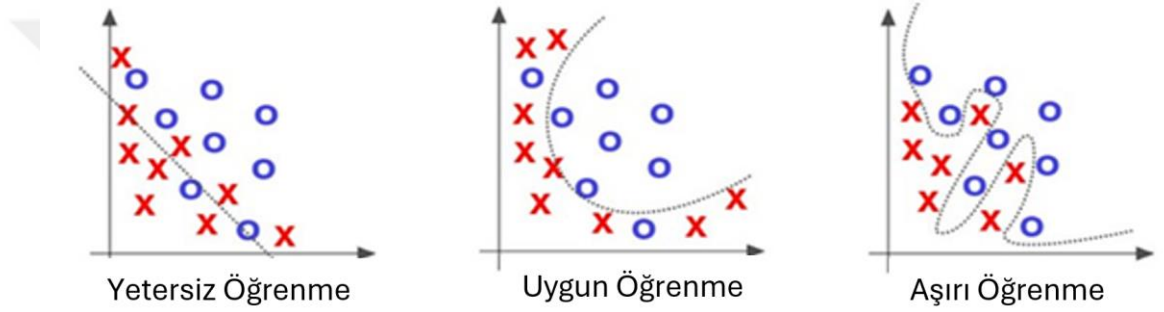
Model karmaşıklığının arttırılması: Eğer oluşturulan model çok basit ise girdi ve çıktılar arasındaki ilişkiyi öğrenemez. Burada yüksek eğitim hatası oluşur. Model tekrar düzenlenerek daha karışık hale getirilerek bu problem giderilmeye çalışılır.

Öznitelik mühendisliği uygulanması: Farklı özniteliklerin çeşitli tekniklerle oluşturulup daha fazla veri noktası ile yetersiz öğrenmenin önüne geçilmeye çalışılır.

Veriden gürültünün çıkarılması: Eğer veriler çok gürültülü ise model öğrenme güçlüğü çeker; çünkü gürültü eğitim verilerini farklı değerler üzerine taşır ve verilerin rastgele olduğunu varsayar; bu da eğitim verisini istenilen derecede ya da yanlış öğreneceğini gösterir.

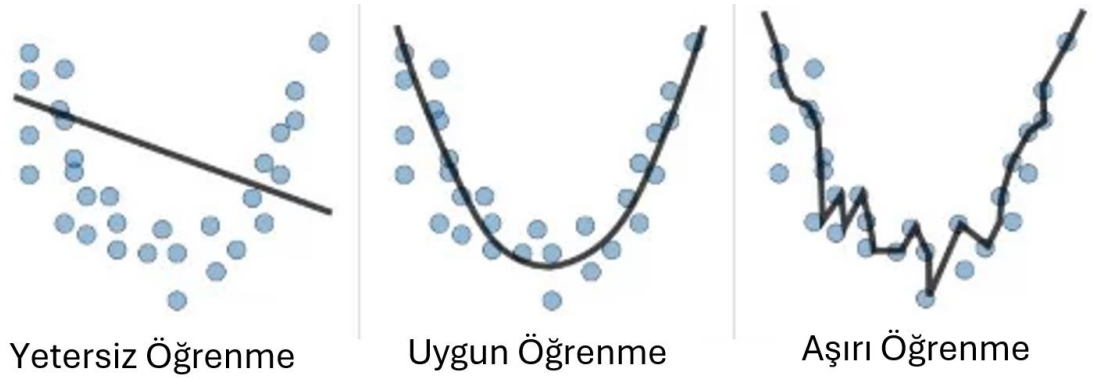
Eğitim süresi veya dönem sayısının arttırılması: Daha fazla dönem sayısı (iterasyon) parametre optimizasyonu, karmaşık veri kümelerini işleme, yakınsama izleme gibi avantajlar sağlar. Bu, modeli daha uzun sürede eğitilmesine olanak tanır ve performansı arttırır.

Eksik veri: Sonuçlar birden fazla değişkene bağlı iken model sadece bir değişken üzerinden yapılandırılırsa eğitim performansı düşük olur.



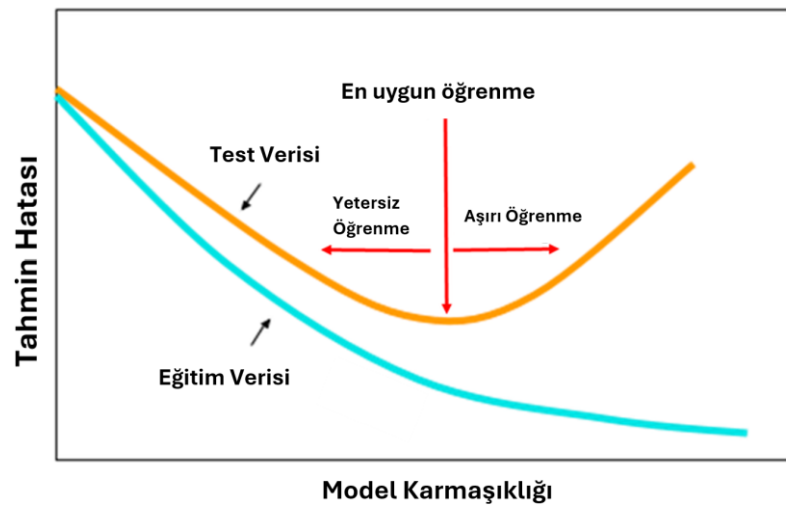
Şekil 2.3.4.1. Sınıflandırma algoritması üzerinden öğrenme metotları (Taş, 2020)

Şekil 2.3.4.1’de görüldüğü üzere sol taraftaki grafikte kesik çizgi ile ifade edilen sınıflandırma çizgisi verileri sınıflandırmada yetersizdir. Burada model oldukça basit kalmaktadır. Sağ taraftaki grafikte ise aşırı öğrenme örneği verilmiştir. Burada model verilen verileri düzgün ayırmışsa da gelecekteki verinin nasıl olacağını tahmin etmekte zorlanır; çünkü sınıflandırma için kullanılan sınır çizgisinin ideal bir matematiksel modeli yoktur. Ayrıca gürültü ve aykırı olan tüm noktaları da kapsama eğilimindedir. Ortada yer alan grafik ise uygun öğrenme ile verilerin iyi bir çizgi ile ayrıldığını gösteriyor. Başarı oranı yüksek ve tahmin edilebilir bir model olduğu için tercih edilmesi en mümkün modeldir.



Şekil 2.3.4.2. Regresyon modeli üzerinden öğrenme metotları (Amidi A. ve Amidi S., 2018)

Şekil 2.3.4.2’de aynı sınıflandırmada olduğu gibi regresyon modelinde de regresyon çizgisinin çeşitli öğrenme yöntemlerine göre çizildiği görülmektedir. Yetersiz öğrenmede yüksek yanlılık, yüksek eğitim öğrenme hatası bulunmaktadır. Aşırı öğrenmede ise düşük eğitim öğrenme hatası olmasına karşılık, yüksek test öğrenme hatası ve yüksek varyans mevcuttur. Model karmaşıklığı ve tahmin hatası üzerinden öğrenme eğrileri Şekil 2.3.4.3’te gösterilmiştir.



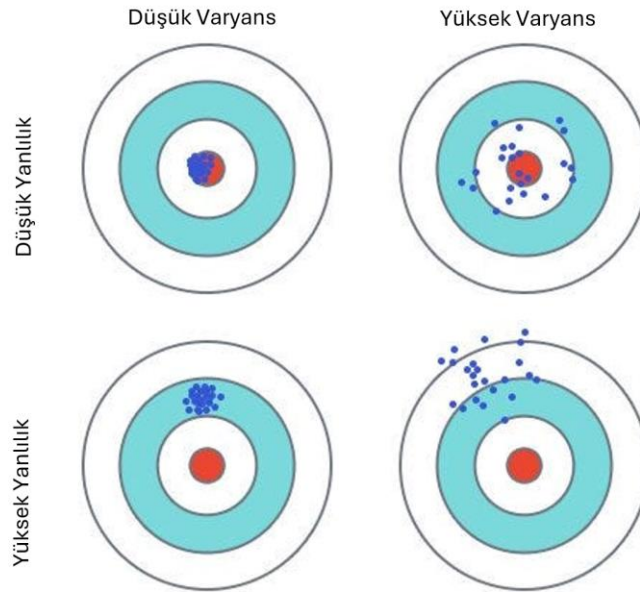
Şekil 2.3.4.3. Öğrenme eğrileri (Durna 2020’den değiştirilerek alınmıştır)

Varyans ve Yanlılık

Yanlılık, öğrenme algoritmasındaki basit varsayımlardan kaynaklanan hatalardır. Normalde varsayımlar modelin anlaşılmasını ve modelin öğrenmesini kolaylaştırır; ancak verilerin karmaşıklığını çözemeyebilir. Modelin girdileriyle çıktısı arasındaki ilişkiyi doğru tahmin edememesiyle sonuçlanır. Bir modelin hem eğitim hem de test verilerinde düşük performansa sahip olması, basit modelden dolayı yüksek yanlılık anlamına gelir ve bu da yetersiz öğrenmeyi işaret eder.

Varyans ise modelin eğitim verilerindeki dalgalanmaları takip ederek, bu dalgalanmalara karşı duyarlılığından oluşan bir hatadır. Eğer bir model yüksek varyansa sahipse verilerin rastgele dalgalanması, gürültüden oldukça etkilenmesi riski oluşur ve test verilerinde zayıf performans gösterir.

Bu açıklamalarla birlikte yanlılığın yetersiz öğrenmeye, varyansın ise aşırı öğrenmeye sebep olduğu görülmektedir. Varyans ve yanlılık etkilerinin tümü Şekil 2.3.4.4'te verilmiştir.



Şekil 2.3.4.4. Varyans ve yanlılık karşılaştırması (Hafeez, 2018)

Çapraz Doğrulama

Bir makine öğrenimi modelinin performansını değerlendirmenin daha doğru ve güvenilir bir yöntemi çapraz doğrulamadır. Çapraz doğrulama, veri setini eğitim ve test setlerine bölerek modeli birçok yönden test eder ve genel performansını değerlendirir.

Eğitim ve test veri setleri, ayrılmalarını kolaylaştırmak için model oluşturulurken ayrılmaktadır. Eğitim seti, modeli oluşturmak için kullanılır ve test seti modelin performansını değerlendirmek için kullanılır. Bu yaklaşımla, farklı eğitim-test bölümleri farklı doğruluk puanları verir ve bu da modelin performansının objektif olarak değerlendirilmesini imkânsız hale getirir. Eğitim-test ayırma metodolojisi bu aşamada henüz tamamlanamaz ve bir çapraz doğrulama tekniği gereksinimi duyar.

En temel çapraz doğrulama teknikleri arasında yer alan K-Fold tekniğinin işleyişi incelendiğinde;

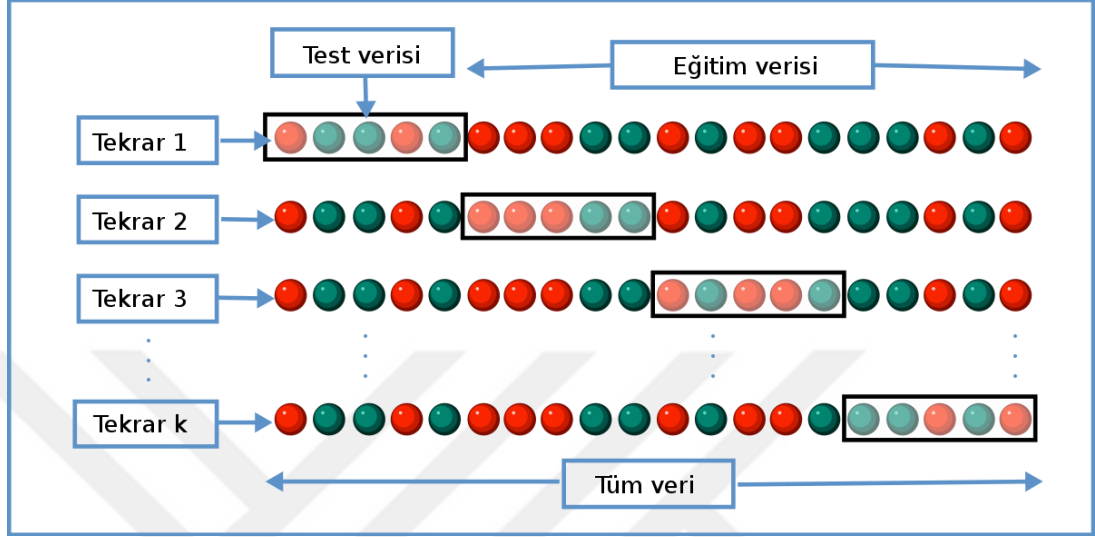
- Veri seti rastgele karıştırılır.
- Veri seti içerisinde k adet grup tanımlanır.

Daha sonra oluşturulan her grup için belli prosedürler uygulanır:

- Doğrulama seti seçilen gruptan oluşur.
- Eğitim seti olarak diğer tüm gruplar (k-1 grupları) kullanılır.
- Eğitim seti, modeli oluşturmak için kullanılır ve doğrulama seti onu değerlendirmek için kullanılır.
- Bir liste modelin değerlendirme puanını içerir.
- Değerlendirme sonuçlarının istatistiksel özeti incelenir. Bu incelemede ortalama, standart sapma, maksimum, minimum vb. değerler bu istatistiksel özete dahil edilmiştir.

Modelin performansının yanlış değerlendirilmesi, K değerinin veri miktarına uygun olarak seçilmesinden kaynaklanabilir. Kesin bir kural olmamakla beraber uygulamalarda genellikle 5 ya da 10 olarak ayarlanabilir (Durna, 2020). Şekil 2.3.4.5'te

model her bir tekrarda test verisi belirlenen ‘K’ sayısı üzerinden kaydırılarak ‘K’ tekrara ulaşıncaya kadar değişik eğitim verileri ve test verileri üzerinden öğrenme yapar.



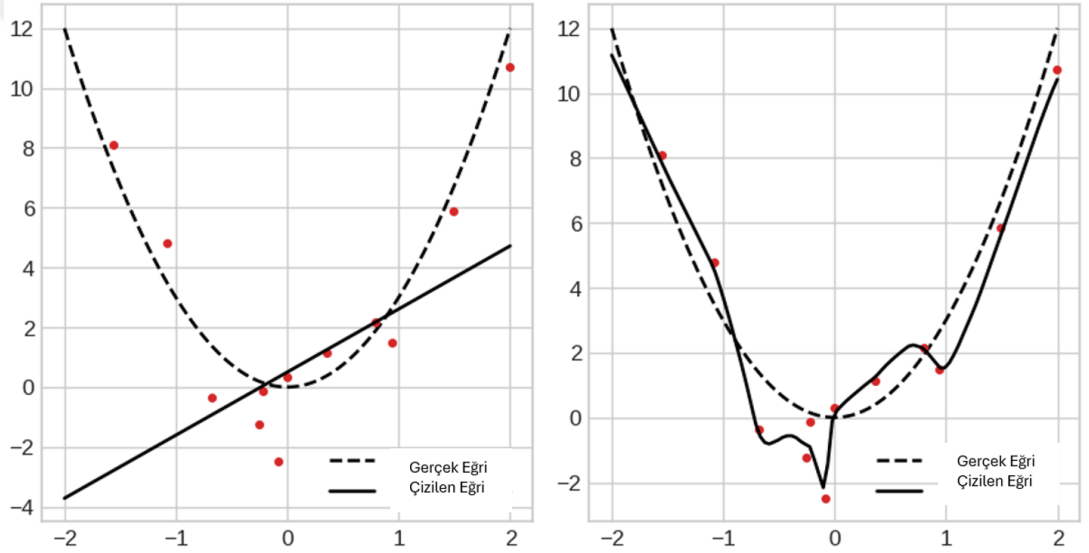
Şekil 2.3.4.5. Çapraz doğrulama diyagramı (Peeyada ve arkadaşları 2022’den değiştirilerek alınmıştır)

Çapraz doğrulamanın ikinci yöntemi ise hiper parametre optimizasyonudur. Makine öğrenimi modellerinde bağımlı ve bağımsız değişkenler arasındaki ilişkiler, hiper parametrelerle temsil edilir. Bu hiper parametreler belirlenen modelde sabit değildir. Başka bir ifadeyle, model eğitimi sırasında kazanılmaz. Araştırmacı bu belirlemeyi modelleme aşamasından önce yapar. Parametrik olmayan sınıflandırma yöntemlerinden biri olan KNN algoritması, örneğin tahmin edilmesi gereken değere en yakın k değerini inceleyerek sınıflandırma yapar. Burada modelin performansını etkileyen ve modelleme öncesinde veri bilimci tarafından ayarlanması gereken hiper parametreler k sayısı ve kullanılacak mesafe ölçüsüdür (Durna, 2020).

Başarı metriğine dayalı olarak bir makine öğrenimi algoritması için en iyi hiper parametre setini belirleme süreci, hiper parametre optimizasyonu olarak bilinir. Aşırı öğrenme ve yetersiz öğrenme arasında bir uzlaşmaya, hiper parametre ayarlama yoluyla model karmaşıklığının ayarlanmasıyla ulaşılabilir. Yine hiper parametreler tarafından verilen kısıtlamalar, modelin esnekliğinden kaynaklanan aşırı öğrenme sorununun çözülmesine yardımcı olabilir. Hiper parametre optimizasyonu güçlü bir temel model

gerektirir. Bu noktada özellik mühendisliğine ve veri ön işlemesine daha fazla yatırım yapmak optimizasyonu iyileştirir (Durna, 2020).

Herhangi bir algoritma için çok fazla sayıda hiper parametre ve bu hiper parametrelerin alabileceği çok sayıda değer olduğundan, her değeri tek tek denemek mümkün olmayacaktır. Bu nedenle hiper parametre optimizasyonu için farklı yöntemler geliştirilmiştir. Şekil 2.3.4.6'da modellerin yetersiz ve aşırı öğrenme sonucu belirledikleri eğriler karşılaştırılmış biçimde gösterilmektedir.



Şekil 2.3.4.6. Yetersiz ve aşırı öğrenme örneği (Holbrook ve Cook, 2024)

Izgara Arama Çapraz Doğrulama (Grid Search Cross Validation)

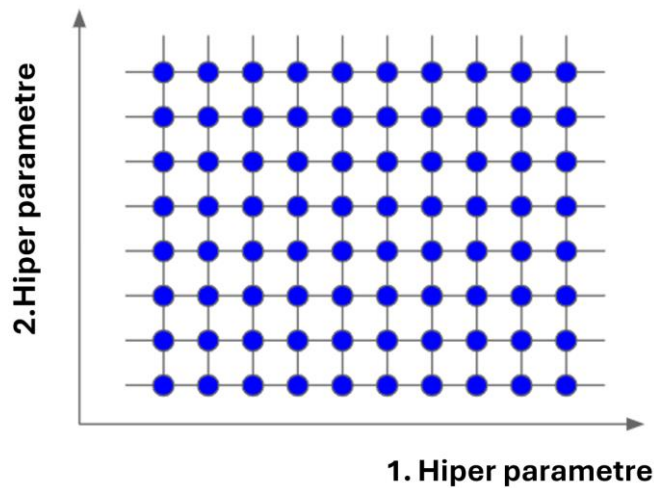
Modelde test edilmesi amaçlanan hiper parametreler ve değerlerinin olası her kombinasyonu yeni bir model oluşturmak için kullanılır ve verilen metrik kullanılarak en etkili hiper parametre seti belirlenir.

Bu teknik, modelin etkinliğini değerlendirmek amacıyla verilen parametrelerin değerlerinden oluşan "kafes" veya "ızgara" üzerindeki her noktayı ziyaret eder. (Bulut 2024).

Izgara arama çapraz doğrulamanın birincil amacı, modelin performansını optimize etmek için kullanılan hiper parametre kombinasyonlarını metodik olarak değerlendirmektir. Bu teknik, belirli bir hiper parametre alanını doldurmak için akla gelebilecek her türlü değer kombinasyonunu dener ve en yüksek performansı sağlayan kümeyi bulur. Izgara arama, bu tarama faaliyetini organize bir şekilde gerçekleştirerek etkili bir optimizasyon yöntemi sağlar (Bulut 2024).

Izgara Arama, belirli bir hiper parametre alanı içindeki değer kombinasyonlarını bir kafes veya ızgara üzerinde düzenler. Her ızgara noktası bir dizi hiper parametre kombinasyonudur. Hiper parametre uzayındaki her nokta, Şekil 2.3.4.7'de gösterildiği gibi diğer tüm noktalardan eşit aralıktır. Daha sonra modeller bu kombinasyonlarla eğitilir ve belirli bir performans kriterine göre değerlendirilir. Izgara araması, bu işlemi mümkün olan her kombinasyon için tekrarlayarak en yüksek performansı sağlayan hiper parametre konfigürasyonunu bulur (Bulut 2024).

Tanımlanan performans ölçümleri her modeli değerlendirmek için kullanılır. Bu göstergeler modelin ne kadar iyi çalıştığını ölçmek için iyi bir yöntemdir. Örneğin F1 puanı, doğruluk ve kesinlik gibi ölçümler sınıflandırma konularına uygulanabilir. (Bulut 2024).



Şekil 2.3.4.7. Izgara arama örneği (Çavdar ve arkadaşları, 2024)

Bunun yararı, hiper parametrelerin her potansiyel kombinasyonunun test edilmesinden dolayı, en iyi performansı gösteren kümenin tanımlanmasına olanak sağlamasıdır. Yalnızca birkaç hiper parametre test edilmek istenildiğinde ve küçük veri kümelerine sahip olduğunda, çok iyi bir performans sergiler. Anlaşılması kolay bir optimizasyon seçeneğidir. Bunu, performansı en üst düzeye çıkarmak amacıyla parametre alanındaki her bir değerin üzerinden metodik olarak teker teker geçerek yapar. İşlevselliği nedeniyle deneyimsiz kullanıcılar için en iyi seçenektir. Belirlenen hiper parametre alanındaki her potansiyel kombinasyon rutin olarak değerlendirilir. Bu düzenlilik sayesinde hiper parametrelerin her kombinasyonu eşit olarak değerlendirilir. Sık tarama, modelin performansını etkileyen birçok hususu kapsayarak genel bir optimizasyon sağlar (Bulut 2024). En iyi hiper parametre değerlerinin bulunduğu örnek Şekil 2.3.4.8’de verilmiştir.

Dezavantajı, büyük veri kümeleriyle çalışırken veya keşfedilecek hiper parametrelerin sayısı ve değeri arttıkça kombinasyon sayısının katlanarak artmasıdır. Oluşturulan her modelin çapraz doğrulama testine tabi tutulduğu göz önüne alındığında masraf her zaman artacaktır; sonuç olarak, bunun yerine “Rastgele Arama Çapraz Doğrulama” yaklaşımı seçilebilir. Bu yaklaşım, belirli bir performans ölçütüne göre en uygun kombinasyonu seçer ancak bu kombinasyonun genel olarak en iyi seçenek olmasını garanti edemez. Özellikle verilen veri seti ve sorun için en iyi cevabı bulma garantisi verilememektedir. Farklı bir ifadeyle, ızgara araması tarafından seçilen kombinasyonlar belirli bir istatistik için en büyük kombinasyonlar olsa da modelin genel performansını etkileyen diğer yönleri gözden kaçırabilirler (Bulut 2024).

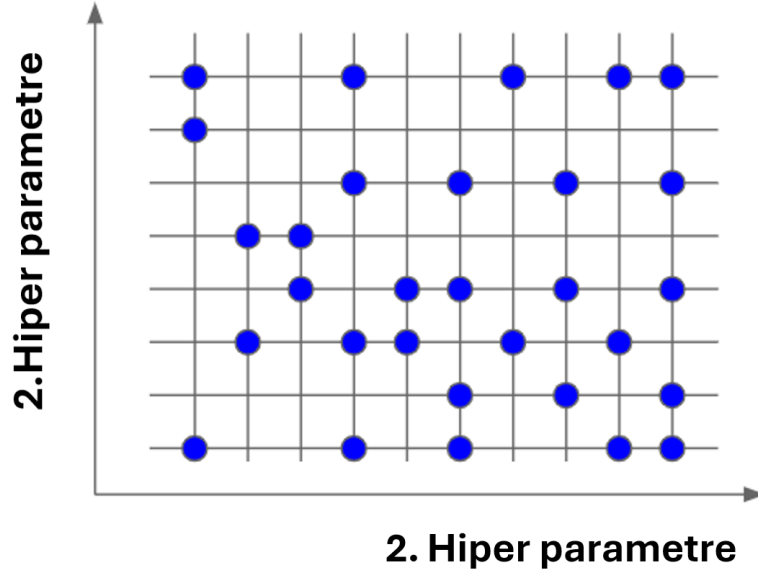
	Alpha 0.1	0.01	0.001	0.0001
Beta 0.1	Found Accuracy: 0.716	Found Accuracy: 0.76	Found Accuracy: 0.81	Found Accuracy: 0.78
0.01	Found Accuracy: 0.721	Found Accuracy: 0.81	Found Accuracy: 0.94	Found Accuracy: 0.86
0.001	Found Accuracy: 0.709	Found Accuracy: 0.88	Found Accuracy: 0.80	Found Accuracy: 0.73
0.0001	Found Accuracy: 0.72	Found Accuracy: 0.69	Found Accuracy: 0.71	Found Accuracy: 0.70

Şekil 2.3.4.8. Izgara arama ile iki hiper parametrenin bir performansı metriği olan kesinlik kriterini gösteren şema (Joos, 2020)

Rastgele Arama Çapraz Doğrulama (Randomized Search Cross Validation)

Bu doğrulama tekniği, çapraz doğrulama oluşturmayı ve modeli test etmek için bir dizi hiper parametrenin rastgele seçilmesini içerir. Bu prosedürler, kullanıcı önceden belirlenen yinleme sayısına veya hesaplama süresi sınırına ulaşana kadar devam eder. (Durna, 2020).

Rastgele Arama Çapraz Doğrulama, Izgara Arama Çapraz Doğrulama 'ya alternatif olarak kullanılır. Genelde hiper parametre sayısı ve değerleri çok fazla olduğunda daha az işlem yapabilmek adına uzayda rastgele hiper parametre kombinasyonlarını dener. Dolayısıyla tüm noktalar yerine, bu noktaların yalnızca rastgele seçilen bir alt kümesini test eder. Bu alt küme ne kadar küçükse optimizasyon o kadar hızlı ancak doğruluğu az olur. Noktaların örnek dağılımları Şekil 2.3.4.9'da verilmiştir.

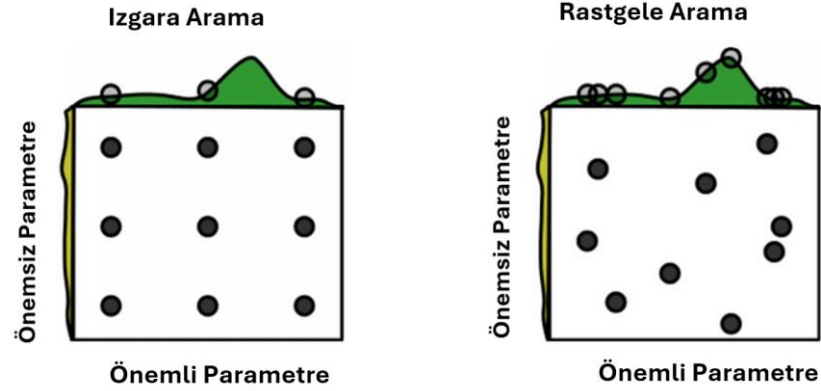


Şekil 2.3.4.9. Rastgele arama çapraz doğrulama (Çavdar ve arkadaşları, 2024)

Avantajı, büyük veri kümelerinde, daha düşük bir maliyetle, ızgara arama yaklaşımıyla elde edilen optimum puanla neredeyse aynı sonuçları verecek hiper parametre kümelerini bulabilmesidir. Bunu kullanarak daha geniş bir hiper parametre aralığı taranabilir.

Dezavantajı ise ideal hiper parametre setine yaklaşırken her potansiyel kombinasyonu denememesi, dolayısıyla en iyi performansa sahip hiper parametre setinin bulunacağının sözünü verememesidir.

Şekil 2.3.4.10'da bir öğrenme modelinden elde edilen önemli ve önemsiz parametrelerin ızgara arama ve rastgele arama yöntemi ile hiper parametre kombinasyonu seçimi karşılaştırılmıştır.



Şekil 2.3.4.10. Izgara arama ile rastgele arama arasındaki hiper parametre kombinasyonu seçimi (Trouvain ve arkadaşları, 2020)

2.4. Kimlik Doğrulama

Makine Öğrenmesinde ve bilgisayar sistemlerinde kimlik doğrulama kullanıcının kimliğini kanıtlamasında kullanılacak bilgilerin tümüdür. Bu şekilde kullanıcının erişmeye çalıştığı sistem, hizmet, uygulama veya belgelere izin veya ret alması sağlanmaktadır.

Kişinin kimlik doğrulaması çeşitli faktörlere bağlıdır. Her kimlik doğrulama faktörü, erişim izni verilmeden önce kişinin kimliğini doğrulamak, işlem talebini onaylamak, bir belgeyi imzalamak ya da başkalarına yetki vermek için kullanılan bir dizi öğeyi kapsar. Bu faktörleri üç farklı kategoride değerlendirebilir. Güvenilir bir kimlik doğrulama için en az iki, tercihen de bu üç faktörün kullanılması gerekmektedir (Federal Financial Institutions Examination Council, 2008). Faktörler şu şekilde sıralanır:

Bilgi: Kullanıcının bildiği bir şey olarak adlandırılır. Parola, PIN numarası, güvenlik sorusu gibi sadece kullanıcının bildiği şeylerdir.

Sahiplik: Kullanıcının sahip olduğu tüm şeyler. Bunlar akıllı saat, kimlik kartı, güvenlik belirteçleri, cep telefonu gibi şeylerdir.

Kalıtım: Kullanıcının kendisine has olan ve tanımlayan tüm şeylerdir. Parmak izi, retina modeli, imza, yüz, ses, biyo-elektrik sinyaller ve diğer biyometrik tanımlayıcılar bu alana girmektedir.

Bahsedilen faktörleri kullanarak, tek faktörlü veya çok faktörlü kimlik doğrulama yapılabilmektedir. En az güvenilir kimlik doğrulama türü, yalnızca bir faktör gerektiren tek faktörlü kimlik doğrulamadır. Genelde iyi bir koruma sağlamaz ve ele geçirilmeye ya da yanlış kullanılmaya karşı savunmasızdır. Bu tip doğrulamalar, hassas bilgi içeren kişisel bilgiye veya finansal bilgiye erişim için tavsiye edilmezler.

Çoklu faktör doğrulama ise iki ya da daha fazla faktör içeren kimlik doğrulama çeşididir. Örneğin banka kartına sahip olmak ve PIN kodunu bilmek iki farklı faktörün bir araya geldiği kimlik doğrulama sistemidir. Bu tip doğrulama sistemi hassas bilgi içeren sistemlere yetki verilmesi için kullanılır. Faktör tanımlamaları dışında ayrıca doğrulama tipi de kimlik doğrulamanın nasıl yapılacağını gösterir. Başlıca doğrulama tipleri şunlardır:

Güçlü Doğrulama: İçinde en az iki faktörü barındıran çok katmanlı bir doğrulama sistemidir. Bu doğrulama tipi çok faktörlü kimlik doğrulamaya benzer ancak daha sıkı gereksinimler de bu doğrulamaya eklenir.

Sürekli Kimlik Doğrulama: Yalnızca ilk oturum açma oturumu sırasında kullanıcıların kimliğini doğrulayan geleneksel bilgisayar sistemlerinde önemli bir güvenlik açığı ortaya çıkabilir. Kullanıcıları bazı biyometrik özelliklerine göre sürekli olarak izleyen ve doğrulayan kimlik doğrulama tekniklerini kullanan sistemlerin bu sorunu çözmesi gerekmektedir. Son araştırmalar, dokunma dinamikleri, tuş vuruş dinamikleri ve yürüyüş tanıma gibi bazı davranışsal özellikleri çıkarmak için akıllı telefon sensörlerini ve aksesuarlarını kullanma olasılığını göstermiştir (De Marsico ve arkadaşları, 2018). Davranışsal biyometri olarak adlandırılan bu özellikler, akıllı telefon kullanıcılarını sürekli ve dolaylı olarak tanımlamak veya doğrulamak için kullanılabilir. Aktif veya

sürekli kimlik doğrulama sistemleri, bu davranışsal biyometrik özellikleri temel olarak oluşturulan sistemlerdir (Mahfouz ve arkadaşları, 2017) (Patel ve arkadaşları, 2016).

Dijital kimlik doğrulama: Bazen "elektronik kimlik doğrulama" veya "e-kimlik doğrulama" olarak da adlandırılan "dijital kimlik doğrulama" ifadesi, bir bilgi sistemine kullanıcı kimliği güvenliğini elektronik olarak oluşturmak ve sağlamak için kullanılan prosedürlerin bir toplamıdır.

Bu tezde bahsedilen çalışmada kalıtım faktörlerinden ikisini (biyo-empedans ve soft biyometri) kullanılarak, sürekli kimlik doğrulamaya benzer bir doğrulama gerçekleştirilmiştir. Bunun nedeni ise daha güvenilir ve az hata yapan bir sistem tasarımı arayışıdır.

2.5. Doğruluk, Kesinlik, Duyarlılık, F1-Skor ve Karmaşıklık Matrisi

Karmaşıklık Matrisi, sınıflandırma algoritmalarının performansını ortaya koyan bir matris tablosudur. Bu tabloda gösterilen bilgiler, modelin performansı, hataları ve zayıflıkları hakkında bilgiler verir.

		GERÇEK SINIF	
		POZİTİF	NEGATİF
TAHMİN SINIF	POZİTİF	TP	FP
	NEGATİF	FN	TN

Şekil 2.5.1. Basit yapıli karmaşıklık matrisi

Şekil 2.5.1’de bulunan kısaltmalar şu şekildedir.

Gerçek Pozitif (TP): Modelin pozitif olarak nitelendirdiğini gerçekte de pozitif olduğunu belirten bölge.

Gerçek Negatif (TN): Modelin negatif olarak nitelendirdiğini gerçekte de negatif olduğunu belirten bölge.

Yanlış Pozitif (FP): Modelin pozitif olarak nitelendirdiğini gerçekte negatif olarak değerlendiren bölge

Yanlış Negatif (FN): Modelin negatif olarak nitelendirdiğini gerçekte pozitif olarak değerlendiren bölge.

Örnek vermek gerekirse, sınıflandırma yaparak kişiye gelen elektronik postaları istenmeyen e-posta olup olmadığına göre ayrılması istenilen bir problem ele alınsın. Model oluşturulduktan sonra hata matrisi üzerinden her bir bölgede yer alan sayılar üzerinden değerlendirme yapılır ve verileri ilgili kutuya taşır. Eğer model istenmeyen bir e-posta tahmin etmişse ve bu gerçekten istenmeyen bir e-posta ise TP kutusuna yerleştirir. Eğer model sıradan bir e-postayı, istenmeyen e-posta olarak değerlendirmiyorsa TN kutusuna yerleştirir. Eğer model sıradan bir e-postayı istenmeyen e-posta olarak değerlendiriyorsa FP kutusuna yerleştirir. Eğer model istenmeyen bir e-postayı sıradan bir e-posta olarak değerlendiriyorsa bu sefer FN kutusuna taşır.

Bu sayılar bulunduktan sonra performans metrikleri bulunur ve kararlar ilgili sonuçlar çıkar. Bu metrikler:

Kesinlik: Pozitif olması beklenen değerlerden ne kadarının pozitif çıktığının ölçüsüdür.

$$Kesinlik = \frac{TP}{TP + FP} \quad (2.15)$$

Duyarlılık: Gerçekte pozitif olan değerlerin bütün tahminlere olan oranını gösteren bir metriktir.

$$Duyarlılık = \frac{TP}{TP + FN} \quad (2.16)$$

F1-Skor: Kesinlik ve Duyarlılık değerlerinin harmonik ortalamasını gösteren metriktir.

$$F_1 = 2 * \frac{Kesinlik * Duyarlilik}{Kesinlik + Duyarlilik} \quad (2.17)$$

Doğruluk: Modeldeki doğru tahmin edilen verilerin toplam veri kümesine olan oranıdır.

$$Doğruluk = \frac{TP + TN}{TP + FP + TN + FN} \quad (2.18)$$

Eğer veriler eşit dağılmamışsa modelin doğruluğu tek başına yeterli olmayacaktır. F1-Skor değerinin kullanılması eşit dağılmayan veri kümelerinde hatalı bir model seçimi yapılmamasını sağlamaktadır. Tüm hata maliyetlerini içeren bir metrik olan F1-Skor, harmonik ortalama alınması sebebiyle uç durumların göz ardı edilmesine de yarar.

3. MATERYAL ve YÖNTEM

3.1. Kullanılan Ekipman

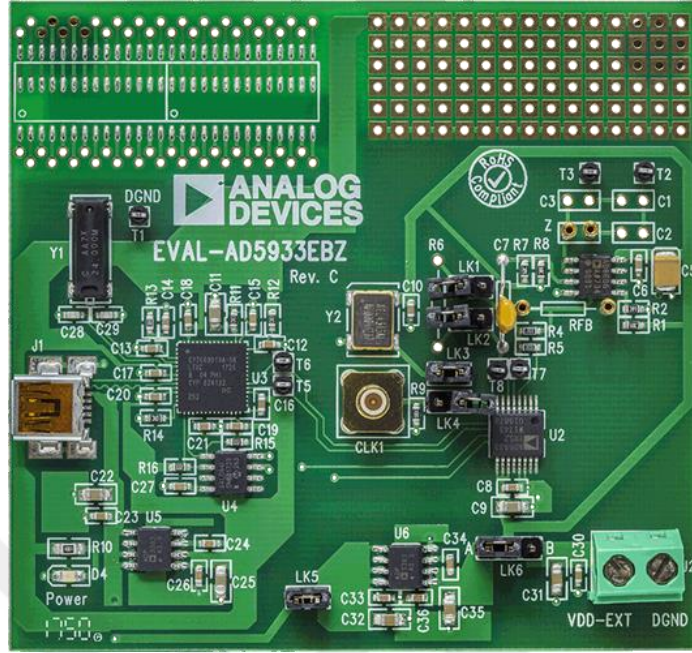
Çalışmada soft biyometrik verilerin toplanabilmesi için dokunmatik ekrana sahip akıllı bir telefon kullanılmıştır. Uygulamada çizilecek olan şekiller ve sürükleme hareketi için oldukça bir geniş ekran alanı sunabilen ‘Samsung Galaxy A51’ model cep telefonu seçilmiştir (Şekil 3.1.1). A51 ekran genişliği 164.0mm (6.5") ve çözünürlüğü 1080 x 2400 pikseldir.



Şekil 3.1.1. Çalışmada kullanılan akıllı telefon

A51 Android işletim sistemine sahip olduğundan, uygulamayı geliştirmek için Android Studio platformu kullanılmıştır. Kullanılan versiyon Android Studio Giraffe | 2022.3.1 Patch 2’dir.

Biyoelektrik empedans verisi toplamak için Analog Devices firmasına ait AD5933EBZ model geliştirme kartı kullanılmıştır (Şekil 3.1.2). Geliştirme kartı üzerinde bulunan AD5933 empedans çevirici ya da daha çok kullanılan ismiyle ağ analizatörü, biyoelektrik empedans uygulamalarında sıklıkla kullanılan bir bütünleşmiş devredir.



Şekil 3.1.2. AD5933EBZ geliştirme kartı

AD5933, yerleşik frekans jeneratörünü 12 bit, 1 MSPS, analogdan dijitale dönüştürücü (ADC) ile birleştiren yüksek hassasiyetli bir empedans dönüştürücü sistem çözümüdür. Frekans üretici, harici bir karmaşık empedansın bilinen bir frekansla uyarılmasına izin verir. Empedanstaki gelen yanıt sinyali, yerleşik ADC tarafından örneklenir ve ayrık bir Fourier dönüşümü (DFT), yerleşik bir DSP motoru tarafından işlenir. DFT algoritması, her çıkış frekansında bir gerçek (R_e) ve sanal (Im) veri sözcüğü döndürür.

AD5933 piyasada çokça bulunan ucuz ve uygulama alanı çok geniş olan bir ağ analizördür. Elektriksel diyagram ve bağlantılar Şekil 3.1.3'te gösterilmiştir.

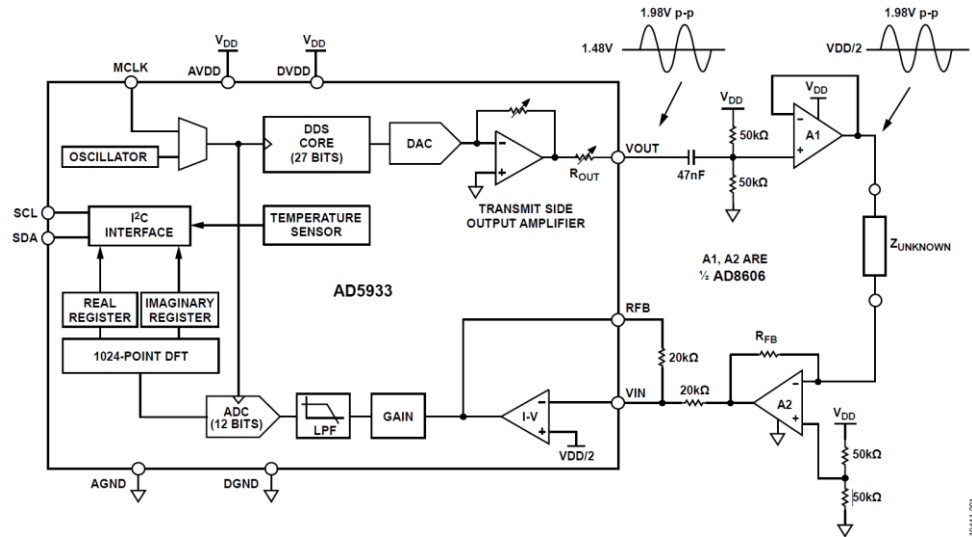
Kalibre edildikten sonra tarama boyunca her frekans noktasında empedansın büyüklüğü ve empedansın bağıl fazı kolayca hesaplanır. Bu, seri I2C arayüzünden okunabilen gerçek ve sanal kayıt içerikleri kullanılarak çip dışında yapılır.

Geliştirme kartında ayrıca gerektiğinde AD5933'e sistem saati görevi görecektir yüksek performanslı kırılmış 16 MHz yüzeye monte kristal bulunur. AD5933'e arayüz oluşturmak, AD5933 ile iletişim kurmak için gerekli I2C sinyallerini üreten bir USB

mikro denetleyici aracılığıyla gerçekleştirilir. USB mikro kontrolcüye arayüz oluşturma, PC'de bulunan ve PC'den çalıştırılan bir Visual Basic grafik kullanıcı arayüzü aracılığıyla yapılır. AD5933 ayrıca 13 bit çözünürlüklü dahili bir sıcaklık sensörü içerir. Parça 2,7 V'den 5,5 V'ye kadar bir beslemeyle çalışır. Diğer yerleşik bileşenler arasında, cihazın ayrı analog ve dijital bölümleri için sabit bir besleme voltajı görevi görececek bir ADR423, AD5933'e 3,0V referansı ve arayüz oluşturan yerleşik evrensel seri veri yolu denetleyicisine bir besleme görevi görececek bir ADP3303 ultra yüksek hassasiyetli regülatör bulunur.

Geliştirme kartı bir USB kablosu yardımıyla bilgisayardan güç alır ve üzerinde gerçek zamanlı empedans ölçümleri yapılabilir.

AD5933EBZ seçilmesinin nedenleri hacmi ve iki adet kablo kullanarak veri toplayabilmesidir. Ayrıca maliyeti ve kolay arayüzü kullanımı tercih etme sebepleri arasındadır.

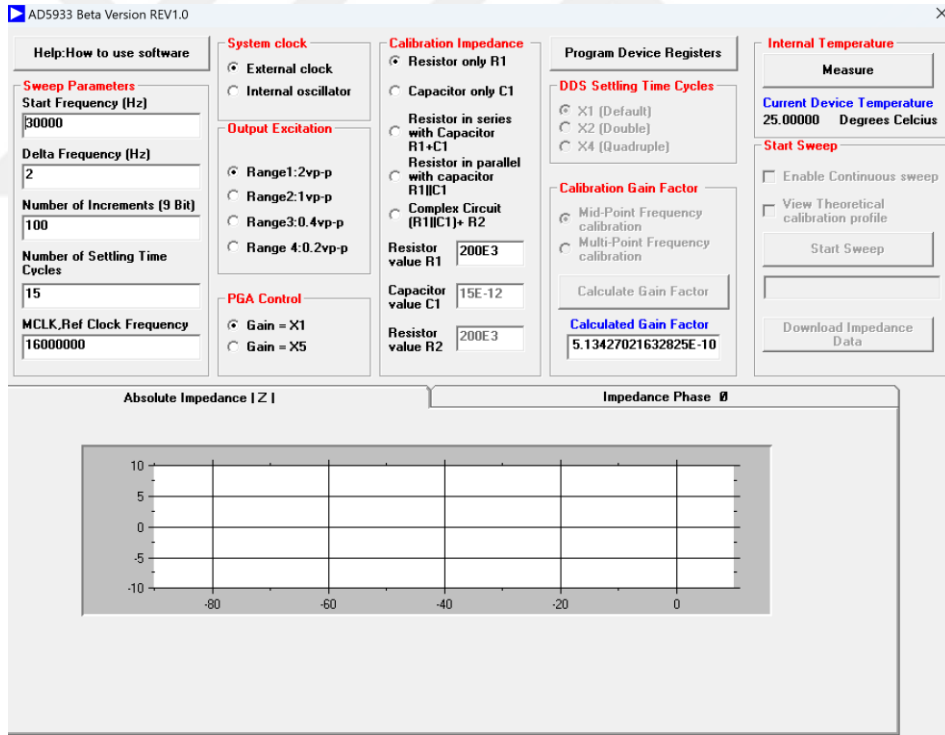


Şekil 3.1.3. AD5933 elektriksel diyagramı

Ölçümlerin istenilen seviyede ve kalitede yapılabilmesi için geliştirme kartının donanımı değiştirilmiştir.

Devrenin düşük frekanslardaki empedans ölçüm performansını bulabilmek için geliştirme kartının geliştirme alanına 4Mhz ve 2Mhz olmak üzere iki ayrı kristal osilatör eklenmiştir. Ayrıca ölçüm yapılacak olan kabloların bağlantısını sağlamak için 2 adet SMA konnektör ve RF kablo geliştirme alanına eklenmiştir. Kristal osilatörler arasında hızlı bir geçiş sağlayabilmek için 2 adet kaydırmalı buton eklenmiştir. Bu sayede hem kalibrasyon hem de ölçüm sırasında kristal frekansı değişimi hızlıca yapılabilmiştir.

Biyoelektrik empedans analizi için çıkış gerilimi, frekans, kalibrasyon empedansı, tarama frekansı, frekans aralıkları ve saat frekansı gibi parametreleri kontrol etmek için AD5933EBZ geliştirme kartı arayüz yazılımı kullanılmıştır (Şekil 3.1.4). Bu yöntem ile hızlıca parametrelerin belirlenip mikrokontrolör adreslerine yazılması sağlanmıştır.



Şekil 3.1.4. AD5933EBZ arayüz yazılımı

3.2. Yöntem

Çalışmaya 164 kişi gönüllü olarak katılmıştır. Gönüllülerden onam formu imzalanmaları istenmiş, çalışmanın detayları anlatılmış, olası yarar ve zararlardan bahsedilmiş, istedikleri zaman deneye son verebilecekleri belirtilmiş ve dilerlerse çalışmadan kendi istekleriyle bütünüyle ayrılarak verilerinin tamamen silineceği taahhüt edilmiştir.

Bu kapsamda çalışmayı yürüten araştırmacı, bütün gönüllülerle tek tek görüşmüş ve rızalarını almıştır. Onam formları 18 yaşından büyükler için tek form olup gönüllünün kendisi tarafından imzalanmış, 18 yaşından küçükler için iki ayrı belge; biri çocuk ve diğeri çocuğun herhangi bir ebeveyni için, imzalanıp bir kopyası gönüllünün kendisine bir kopyası da Uludağ Üniversitesi Etik Kurulu'na teslim edilmiştir. Gönüllülerin yaş ve cinsiyet bilgileri alındıktan sonra sırasıyla biyoelektrik empedans verileri ve soft biyometrik verileri alınmıştır.

Çizelge 3.2.1. Gönüllülerin yaş ve cinsiyet dağılımı

Yaş dağılımı	Kadın	Erkek	Toplam
10-17	3	12	15
17-24	5	14	19
24-31	12	8	20
31-38	21	19	40
38-45	11	14	25
45-52	11	7	18
52-59	6	4	10
59 ve üzeri	9	8	17

Çizelge 3.2.2. Deneye katılan toplam kişi sayısı ve yaş ortalamaları

Cinsiyet	Kişi Sayısı	Yaş Ort.
Kadın	78	40,01
Erkek	86	35,09

Çizelge 3.2.1’de 8 farklı yaş ve 2 farklı cinsiyet olmak üzere 16 etiket üzerinden değerlendirilecek olan yaş ve cinsiyet dağılımı verilmiştir. Çizelge 3.2.2’de ise gönüllülerin cinsiyetlerine göre kişi sayısı ve yaş ortalaması verilmiştir.

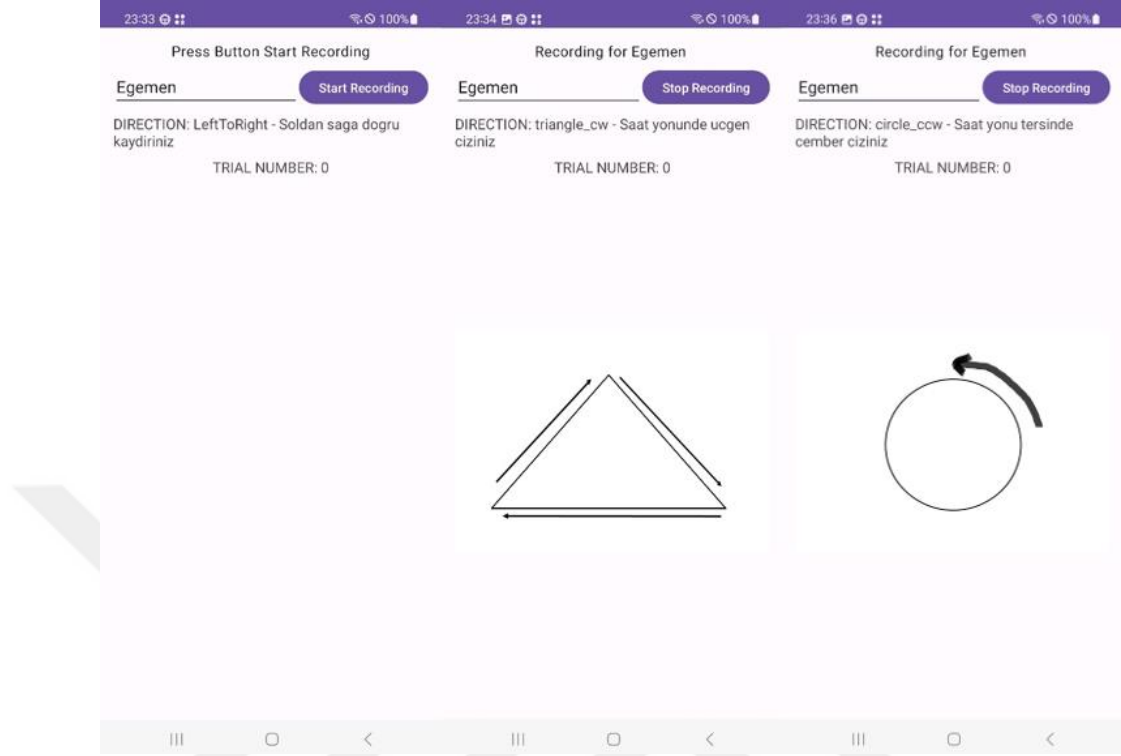
Gönüllülerden uygulamadaki direktifleri sadece sağ elleri ve baş parmakları ile yapmaları istenmiştir. Bu sayede daha kontrollü bir çalışma olması amaçlanmıştır. Cinsiyet tahmininde baş parmak uzunluğu önemli bir parametre olup hem çizimlerin tamamlanma süresi hem de kaydırmadaki sürüklenme mesafesini daha belirgin hale getirmiştir.

Çalışmaya katılan gönüllülerden telefon uygulamasını kullanarak dört farklı kaydırma hareketi yapmaları istenmiş (soldan sağa, sağdan sola, yukarıdan aşağıya ve aşağıdan yukarı şeklinde) sonrasında da saat yönü ve saat yönü tersine göre sırasıyla üçgen ve çember çizimleri istenmiştir (Şekil 3.2.1). Kaydırma işlemi, belirli bir şekle tabi olmadan, kullanıcının isteğine bırakılmıştır. Üçgen ve çember çizimleri ise uygulamada gösterilen şeklin kenarları üzerinden takip edilmesi istenmiştir (Şekil 3.2.2).



Şekil 3.2.1. Kullanıcıdan istenilen kaydırma ve çizimler

Her işlem 15 defa tekrarlanmış ve 164 kişiden toplamda 19680 adet soft biyometrik verisi toplanmıştır.



Şekil 3.2.2. Uygulamadan örnek ekran görüntüleri

Toplanan bu veriler telefon hafızasının içerisinde .csv uzantılı bir Microsoft Excel tablosuna aktarılmıştır. (Şekil 3.2.3)

id	UserId	UserName	swipeDir	repetition	startPoint	endPoint	startTime	endTime	startPress	endTime	size	startTime	endTime	Velocityx	Velocityy	points	Pressure	touchSize
1	-2E-09	movements3	derleme	LeftToRight	1 Point(377, Point(790, 0.011764707	77964601	77964716	7895.426	236.7797	Point(450, Point(532, Point(641, Point(736, Point(790, 1 1 1 1 0.013725 0.013725 0.011								
2	-2E-09	movements3	derleme	LeftToRight	2 Point(394, Point(827, 0.009803922	77965255	77965353	8510.733	820.7736	Point(423, Point(508, Point(623, Point(733, Point(827, 1 1 1 1 0.013725 0.015686 0.015								
3	-2E-09	movements3	derleme	LeftToRight	3 Point(311, Point(779, 0.011764707	77965520	77965611	1100.1	-168.823	Point(358, Point(464, Point(607, Point(748, Point(779, 1 1 1 1 0.015686 0.013725 0.013725 0.013								
4	-2E-09	movements3	derleme	LeftToRight	4 Point(366, Point(726, 0.013725491	77965762	77965844	9621.799	-1015.51	Point(448, Point(556, Point(680, Point(726, 1 1 1 1 0.015686 0.013725 0.011765 0.013725								
5	-2E-09	movements3	derleme	LeftToRight	5 Point(256, Point(707, 0.013725491	77965979	77966079	9755.682	-662.368	Point(318, Point(419, Point(537, Point(667, Point(707, 1 1 1 1 0.015686 0.015686 0.015686 0.015								
6	-2E-09	movements3	derleme	LeftToRight	6 Point(350, Point(711, 0.011764707	77966229	77966311	8128.488	-261.863	Point(311, Point(444, Point(564, Point(679, Point(711, 1 1 1 1 0.015686 0.013725 0.011765 0.011765								
7	-2E-09	movements3	derleme	LeftToRight	7 Point(278, Point(740, 0.013725491	77966421	77966512	9388.672	-1516.93	Point(329, Point(424, Point(552, Point(679, Point(740, 1 1 1 1 0.015686 0.013725 0.013 0.013								
8	-2E-09	movements3	derleme	LeftToRight	8 Point(404, Point(762, 0.013725491	77966528	77966711	9216.209	-1007.05	Point(425, Point(508, Point(626, Point(748, Point(762, 1 1 1 1 0.015686 0.013725 0.011765 0.011765								
9	-2E-09	movements3	derleme	LeftToRight	9 Point(325, Point(791, 0.009803922	77966837	77966919	9875.534	-46.9783	Point(400, Point(510, Point(661, Point(791, Point(792, 1 1 1 1 0.015686 0.015686 0.015686 0.009604								
10	-2E-09	movements3	derleme	LeftToRight	10 Point(363, Point(743, 0.015686275	77967045	77967119	9825.02	-417.565	Point(386, Point(482, Point(607, Point(743, Point(744, 1 1 1 1 0.015686 0.015686 0.015686 0.015686								
11	-2E-09	movements3	derleme	LeftToRight	11 Point(383, Point(778, 0.011764707	77967245	77967318	9811.51	1501.828	Point(415, Point(513, Point(642, Point(778, Point(779, 1 1 1 1 0.013725 0.013725 0.013725 0.011765								
12	-2E-09	movements3	derleme	LeftToRight	12 Point(320, Point(698, 0.011764707	77967549	77967618	10420.04	-1406.76	Point(343, Point(427, Point(548, Point(698, Point(700, 1 1 1 1 0.015686 0.015686 0.011765 0.011765								
13	-2E-09	movements3	derleme	LeftToRight	13 Point(310, Point(802, 0.013725491	77967753	77967834	11230.44	-476.927	Point(385, Point(505, Point(658, Point(802, Point(803, 1 1 1 1 0.013725 0.015686 0.015686 0.013725								
14	-2E-09	movements3	derleme	LeftToRight	14 Point(290, Point(748, 0.011764707	77967984	77968067	10935.64	40.6307	Point(350, Point(461, Point(596, Point(748, Point(749, 1 1 1 1 0.015686 0.015686 0.015686 0.011765								
15	-2E-09	movements3	derleme	LeftToRight	15 Point(325, Point(639, 0.013725491	77968219	77968352	9291.202	842.2922	Point(381, Point(474, Point(600, Point(639, Point(640, 1 1 1 1 0.015686 0.015686 0.013725 0.013725								
16	-2E-09	movements3	derleme	RightToLeft	1 Point(291, Point(698, 0.013725491	77968493	77968576	10124.61	-507.87	Point(318, Point(389, Point(514, Point(654, Point(698, 1 1 1 1 0.015686 0.015686 0.015686 0.015686								
17	-2E-09	movements3	derleme	RightToLeft	2 Point(741, Point(227, 0.013725491	77968976	77969050	-14089.9	6116.077	Point(717, Point(802, Point(414, Point(227, Point(228, 1 1 1 1 0.013725 0.015686 0.015686 0.013725								
18	-2E-09	movements3	derleme	RightToLeft	3 Point(696, Point(135, 0.009803922	77969259	77969342	-10849.9	5500.005	Point(652, Point(503, Point(322, Point(157, Point(135, 1 1 1 1 0.015686 0.013725 0.013725 0.013725								
19	-2E-09	movements3	derleme	RightToLeft	4 Point(376, Point(141, 0.011764707	77969534	77969600	-10220.3	3506.184	Point(441, Point(274, Point(141, Point(142, Point(143, 1 0.015686 0.013725 0.011765 0.011765								
20	-2E-09	movements3	derleme	RightToLeft	5 Point(653, Point(57, 1, 0.011764707	77969766	77969849	-9871.4	4613.301	Point(518, Point(332, Point(144, Point(57, 1, 1 1 1 1 0.015686 0.013725 0.011765 0.011765								
21	-2E-09	movements3	derleme	RightToLeft	6 Point(680, Point(122, 0.011764707	77970008	77970099	-7352.34	4500.149	Point(633, Point(486, Point(331, Point(198, Point(122, 1 1 1 1 0.015686 0.019608 0.017 0.017								
22	-2E-09	movements3	derleme	RightToLeft	7 Point(916, Point(187, 0.013725491	77970258	77970341	-13372.6	3686.305	Point(769, Point(601, Point(403, Point(212, Point(167, 1 1 1 1 0.013725 0.015686 0.015 0.015								
23	-2E-09	movements3	derleme	RightToLeft	8 Point(731, Point(238, 0.011764707	77970568	77970591	-8524.71	2917.844	Point(686, Point(546, Point(386, Point(268, Point(238, 1 1 1 1 0.013725 0.017947 0.015 0.015								
24	-2E-09	movements3	derleme	RightToLeft	9 Point(767, Point(104, 0.009803922	77970742	77970856	-5710.89	2913.983	Point(755, Point(631, Point(441, Point(270, Point(107, Point(104, 1 1 1 1 0.013725 0.015686 0.011765 0.011765								
25	-2E-09	movements3	derleme	RightToLeft	10 Point(747, Point(172, 0.015686275	77970991	77971090	-7325.01	2696.899	Point(712, Point(586, Point(416, Point(268, Point(185, Point(172, 1 1 1 1 0.013725 0.015686 0.011765 0.011765								
26	-2E-09	movements3	derleme	RightToLeft	11 Point(744, Point(72, 1, 0.009803922	77971292	77971332	-8887.68	4125.359	Point(643, Point(465, Point(291, Point(135, Point(72, 1, 1 1 1 1 0.013725 0.013725 0.015 0.015								
27	-2E-09	movements3	derleme	RightToLeft	12 Point(756, Point(156, 0.011764707	77971482	77971564	-11626.9	3216.238	Point(648, Point(469, Point(287, Point(156, Point(157, 1 1 1 1 0.013725 0.015686 0.011765 0.011765								
28	-2E-09	movements3	derleme	RightToLeft	13 Point(719, Point(199, 0.013725491	77971724	77971806	-9418.3	3401.676	Point(675, Point(533, Point(370, Point(238, Point(199, 1 1 1 1 0.015686 0.015686 0.015686 0.015686								
29	-2E-09	movements3	derleme	RightToLeft	14 Point(687, Point(96, 1, 0.007843138	77971974	77972055	-11733.5	4366.452	Point(639, Point(477, Point(296, Point(136, Point(96, 1, 1 1 1 1 0.013725 0.011765 0.013725 0.013725								
30	-2E-09	movements3	derleme	RightToLeft	15 Point(663, Point(279, 0.013725491	77972348	77972355	-13733.1	2958.756	Point(531, Point(338, Point(279, Point(280, Point(281, 1 1 1 1 0.013725 0.015686 0.013725 0.013725								
31	-2E-09	movements3	derleme	down	1 Point(825, Point(668, 0.011764707	77972893	77972942	-1980.202	12979.11	Point(647, Point(668, Point(669, Point(670, Point(671, 1 1 1 1 0.013725 0.011765 0.011765 0.011765								
32	-2E-09	movements3	derleme	down	2 Point(614, Point(646, 0.011764707	77973115	77973173	1314.902	12160.32	Point(618, Point(632, Point(646, Point(647, Point(648, 1 1 1 1 0.013725 0.013725 0.011765 0.011765								
33	-2E-09	movements3	derleme	down	3 Point(588, Point(633, 0.011764707	77973323	77973381	2348.369	15076.7	Point(600, Point(633, Point(647, Point(648, Point(649, 1 0.011764707 0.013725 0.011765 0.011765								
34	-2E-09	movements3	derleme	down	4 Point(581, Point(620, 0.013725491	77973531	77973579	2051.145	12358.91	Point(593, Point(620, Point(633, Point(634, Point(635, 1 1 1 1 0.011765 0.013725 0.011765 0.011765								
35	-2E-09	movements3	derleme	down	5 Point(558, Point(604, 0.011764707	77973730	77973788	2231.33	13112.26	Point(569, Point(584, Point(604, Point(605, Point(606, 1 1 1 1 0.013725 0.011765 0.011765 0.011765								
36	-2E-09	movements3	derleme	down	6 Point(487, Point(513, 0.011764707	77973929	77973970	1645.895	10386.17	Point(504, Point(513, Point(528, Point(539, Point(540, 1 1 1 1 0.013725 0.011765 0.011765 0.011765								
37	-2E-09	movements3	derleme	down	7 Point(495, Point(514, 0.011764707	77974139	77974188	1181.036	13134.45	Point(507, Point(514, Point(529, Point(540, Point(541, 1 1 1 1 0.013725 0.011765 0.011765 0.011765								

Şekil 3.2.3. Soft biyometri verilerinin toplandığı örnek tablo

Uygulamadan 8 farklı veri elde edilmiştir. Bu veriler:

Çizime başlanan koordinat: Kullanıcının ekrana dokunmaya başladığı anda, kayda başlanan ekran düzlemindeki koordinat verisidir.

Çizimin bittiği koordinat: Kullanıcının ekrandan parmağını çektiği anda, kaydedilen son koordinat verisidir.

Başlangıç zamanı: Kullanıcının belirlenen her hareket ve tekrarda ekrana dokunmasıyla başlayan süre verisidir.

Bitiş zamanı: Kullanıcının belirlenen her hareket ve tekrarda ekrandan parmağını kaldırmasıyla kaydedilen süre verisidir.

X eksenindeki hız: Çizim veya tarama sırasında hareketin tamamını kapsayan x eksenindeki ortalama hız verisidir.

Y eksenindeki hız: Çizim veya tarama sırasında hareketin tamamını kapsayan y eksenindeki ortalama hız verisidir.

Basınç: Dokunma süresinde her noktada uygulanan basınç verisidir.

Dokunma Yüzeyi: Ekrandaki hareket esnasında kullanıcının her çizim noktasındaki dokunma alan verisidir.

Uygulamada Android işletim sisteminin kendine ait olan temel fonksiyonlarından yararlanılmıştır. Tek aktiviteli olarak tasarlanan uygulamada, uygulama kaydırma ve çizim hareketleri yapmıştır. Bütün fonksiyonlar Android SDK'da bulunan GestureDetector.OnGestureListener arayüzünün fonksiyonlarıdır. Dolayısıyla bu arayüz içindeki MotionEvent nesnesi ile bağlantı kurulmuş ve parametreler elde edilmiştir.

onFling fonksiyonu

X ve Y düzlemi üzerindeki hızı, saniyede taranan piksel şeklinde alır.

Çalışmada onFling fonksiyonunda, (Şekil 3.2.4) kullanıcı parmağını dokunmaya başladığından kaldırıncaya kadar hız verilerini almış ve harici dosyaya aktarmıştır. Fonksiyon ayrıca verilerin dosyaya yazılması ile ilgili CSVHelper java sınıfından diğer verileri dizi şeklinde almış ve her hareket sonunda harici dosyaya yazmıştır. Bunu kullanıcının her dokunma hareketinden sonra tekrarlamıştır. Her bir kullanıcı için 15 x 8 = 90 defa fonksiyon çalışmıştır.

```
override fun onFling(
    event1: MotionEvent, event2: MotionEvent, velocityX: Float, velocityY: Float
): Boolean {
    if (event2.action == MotionEvent.ACTION_UP && movementStarted) {
        movementStarted = false
        repetitionCount.value = repetitionCount.value?.plus(1)
        endTime = event2.eventTime

        val userName = binding.etUserName.text.toString()
        val userNameHash = userName.hashCode()
        CSVHelper().saveMovement(
            userNameHash,
            userName,
            swipeDirection.value?.direction,
            repetitionCount: repetitionCount.value ?: 0,
            startPoint,
            actionPoints,
            pressureValues,
            sizeValue,
            velocityX,
            velocityY,
            startTime,
            endTime
        )
        actionPoints.clear()
        pressureValues.clear()
        sizeValue.clear()
        if (repetitionCount.value == MAX_REPETITION_COUNT) {
            repetitionCount.value = 0
            swipeDirection.value = swipeDirection.value?.getNextDirection()
        }

        if (swipeDirection.value == SwipeDirection.IDLE) {
            stopRecording()
        }
    }
    return true
}
```

Şekil 3.2.4. Uygulamadaki onFling fonksiyonu

```

void saveMovement(
    int userNameHash,
    String userName,
    String swipeDirection,
    int repetitionCount,
    Point startPoint,|
    ArrayList<Point> actionPoints,
    ArrayList<Float> pressureValues,
    ArrayList<Float> sizeValue,
    Float velocityX,
    Float velocityY,
    Long startTime,
    Long endTime

```

Şekil 3.2.5. onFling fonksiyonunun çağırdığı liste

CSVHelper dosyası her bir hareketi yeni bir satır olarak .csv uzantılı dosyaya ekler.

onDown fonksiyonu

Kullanıcının ekrana dokunulması ile başlangıç noktasını ve başlangıç zamanını yazar (Şekil 3.2.6)

```

override fun onDown(event: MotionEvent): Boolean {
    if (movementStarted || !isRecording) return false
    movementStarted = true
    if (event.action == ACTION_DOWN) {
        startPoint = Point(event.x.toInt(), event.y.toInt())
        startTime = event.eventTime
    }
    return true
}

```

Şekil 3.2.6. Uygulamadaki onDown fonksiyonu

onScroll fonksiyonu

X ve Y eksenini üzerindeki katedilen mesafeyi yazar. Uygulamada kullanıcının parmağıyla dokunduğu bütün noktaları, basınç değerleri ve basılan yüzey alanı da bu fonksiyon elde eder (Şekil 3.2.7).

```

override fun onScroll(
    event1: MotionEvent, event2: MotionEvent, distanceX: Float, distanceY: Float
): Boolean {
    Log.d(DEBUG_TAG, msg: "onScroll: $event1 $event2")
    Log.d(DEBUG_TAG, msg: "onScroll pressure value: ${event2.size}")
    if (event2.action == ACTION_MOVE && movementStarted) {
        val movePoint = Point(event2.x.toInt(), event2.y.toInt())
        actionPoints.add(movePoint)
        pressureValues.add(event2.pressure)
        sizeValue.add(event2.size)
    }
    return true
}

```

Şekil 3.2.7. Uygulamadaki onScroll fonksiyonu

Bu fonksiyonların dışında yazılan ve ana aktivite ekranında bulunan üç ayrı fonksiyon mevcuttur.

handleButtonClick()

Uygulamanın açılış ekranındaki butonu aktifleştirmek için kullanılan fonksiyondur (Şekil 3.2.8).

startRecording()

Uygulamada isim yazıldıktan sonra tarif edilen kayıt başlama butonudur. Basıldıktan sonra buton üzerindeki yazı değişerek ‘kayıt sonlandır’ haline gelmektedir (Şekil 3.2.8).

stopRecording()

Kayıt başladıktan sonra kayıt butonunun yerine çıkan yazı butonudur. Kaydı durdurduktan sonra buton üzerindeki yazı değişerek ‘kayı başlat’ haline gelmektedir (Şekil 3.2.8).

```

private fun handleClick() {
    if (!isRecording) { //Start Recording
        startRecording()
    } else {
        stopRecording()
    }
}

private fun startRecording() {
    isRecording = true

    binding.tvStatus.text = getString(R.string.user_name, binding.etUserName.text)
    binding.btnStartStopRecord.text = "Stop Recording"

    (getSystemService(Context.INPUT_METHOD_SERVICE) as InputMethodManager)
        .hideSoftInputFromWindow(binding.etUserName.windowToken, flags 0)

    swipeDirection.value = SwipeDirection.RIGHT
}

private fun stopRecording() {
    isRecording = false
    repetitionCount.value = 0

    binding.etUserName.text.clear()
    binding.tvStatus.text = "Idle"
    binding.btnStartStopRecord.text = "Start Recording"
}

```

Şekil 3.2.8. handleClick(), startRecording(), stopRecording() fonksiyonları

8 farklı hareketin hepsi birer obje olarak oluşturulmuş ve sıralanmıştır. Enum sınıfı tanımlanmış hareketleri içine almış ve getNextDirection fonksiyonu da hareketleri sıralamıştır (Şekil 3.2.9).

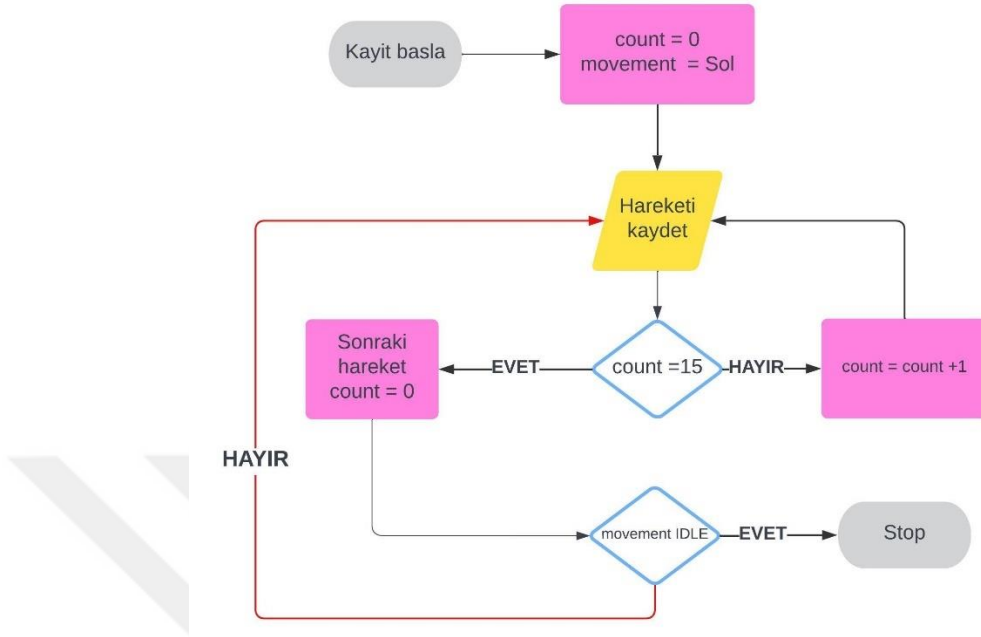
```

private enum class SwipeDirection(val direction: String, val tooltip: String) {
    UP( direction: "up", tooltip: "Yukarı doğru kaydırınız"),
    DOWN( direction: "down", tooltip: "Aşağı doğru kaydırınız"),
    LEFT( direction: "RightToLeft", tooltip: "Sağdan sola doğru kaydırınız"),
    RIGHT( direction: "LeftToRight", tooltip: "Soldan sağa doğru kaydırınız"),
    TRIANGLE_CW( direction: "triangle_cw", tooltip: "Saat yönünde ucgen ciziniz"),
    TRIANGLE_CCW( direction: "triangle_ccw", tooltip: "Saat yönü tersi ucgen ciziniz"),
    CIRCLE_CW( direction: "circle_cw", tooltip: "Saat yönünde cember ciziniz"),
    CIRCLE_CCW( direction: "circle_ccw", tooltip: "Saat yönü tersinde cember ciziniz"),
    IDLE( direction: "idle", tooltip: "no action necessary");

    fun getNextDirection(): SwipeDirection {
        return when (this) {
            RIGHT -> LEFT
            LEFT -> DOWN
            DOWN -> UP
            UP -> TRIANGLE_CW
            TRIANGLE_CW -> TRIANGLE_CCW
            TRIANGLE_CCW -> CIRCLE_CW
            CIRCLE_CW -> CIRCLE_CCW
            CIRCLE_CCW -> IDLE
            else -> IDLE
        }
    }
}

```

Şekil 3.2.9. Hareketlerin sınıf içine alınıp sıralanması



Şekil 3.2.10. Uygulama akış diyagramı

Soft biyometrik veri toplama akış diyagramı Şekil 3.2.10’da gösterilmiştir.

Biyoelektrik empedans verilerini toplama işlemi çalışmada kullanılan cep telefonu kılıfının üzerinde bulunan bakır bantların üzerine lehimlenmiş iki adet koaksiyel kablo ile yapılmıştır. Koaksiyel kablo sinyal gürültüsünden etkilenilmemesi için tercih edilmiştir.

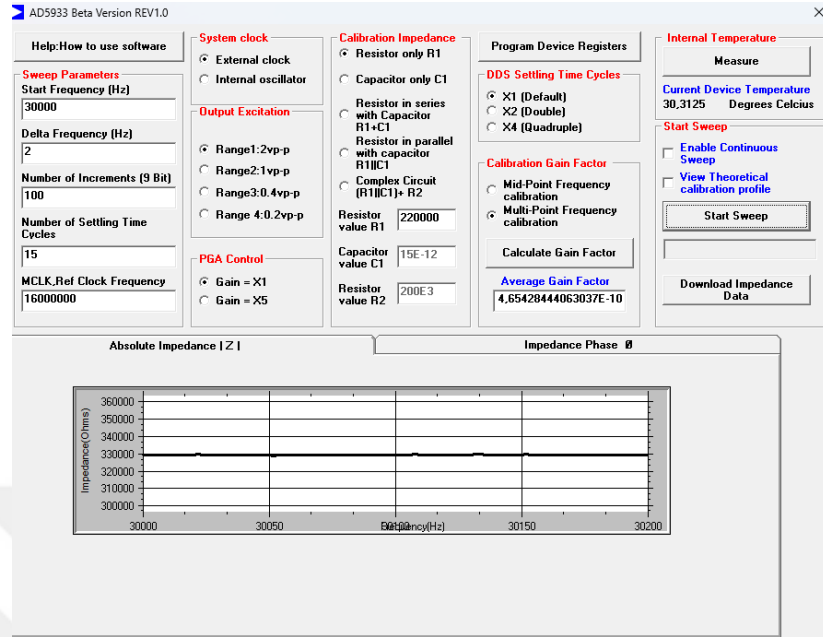
Bakır bantlar birbirlerinden yaklaşık 0,3 cm. ara ile telefon kılıfının enine doğru yapıştırılmıştır. Koaksiyel kabloların her biri bu bakır bantların üzerine lehimlenmiştir (Şekil 3.2.11).

Bunun yapılmasındaki amaç, cep telefonunu tutan kişinin her iki bakır banda dokunması ve devrenin kapalı hale getirilmesidir. Ölçümler farklı frekans genişliklerinde, farklı uyarım gerilimlerinde ve farklı kristal osilatör frekanslarında yapılmıştır.



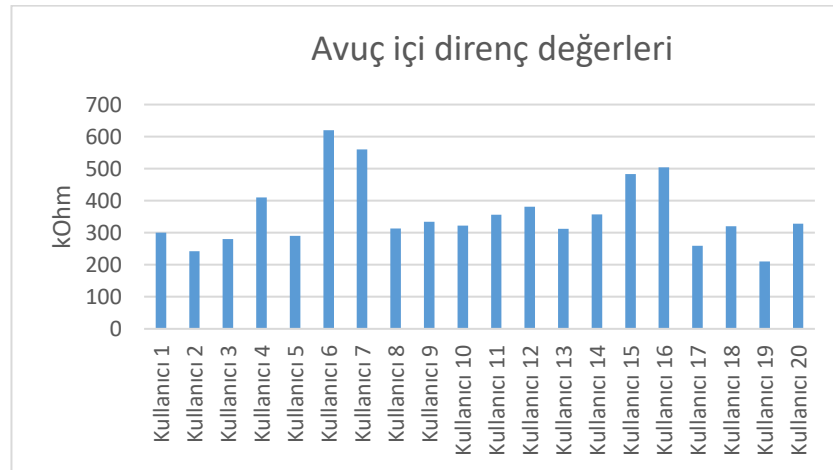
Şekil 3.2.11. Deney düzeneği

AD5933EBZ'nin düzgün çalıştığını anlayabilmek için öncesinde kalibrasyon yapılmıştır. Kalibrasyon, devre üzerinde bulunan Rfb direnç yuvasına kalibre edilmesi istenilen direncin konulması ve ölçüm yapılmak istenen Z yuvasına aynı direnci konulmasıyla yapılmıştır. Kalibrasyon, Analog Devices'ın tavsiye ettiği kullanıcı rehberi (UG-364) kullanılarak belirlenmiştir. Rfb ve Z yuvasına 200 k Ω dirençler bağlanmıştır. Daha sonra kullanıcı rehberindeki adımlar izlenmiştir. Kalibrasyonun doğruluğunu kanıtlamak için daha sonra Z yuvasına değeri bilinen başka bir direnç (330 k Ω) takılmış ve empedans değeri ölçülmüştür (Şekil 3.2.12).



Şekil 3.2.12. Kalibrasyon sonrası doğrulama

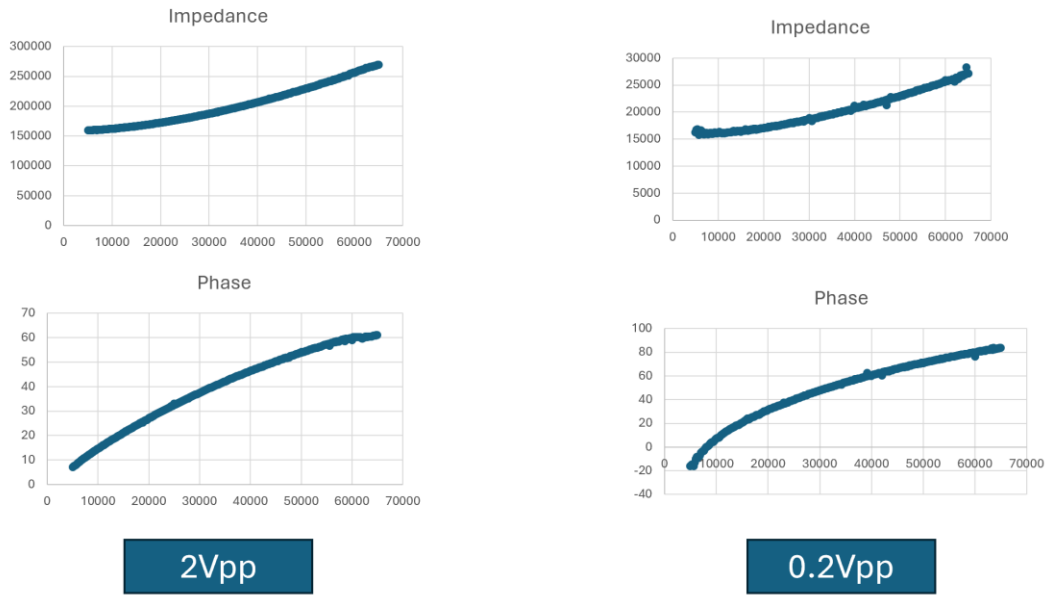
Çalışmada empedans verileri elde edilirken bütün gönüllülerden avuç içi direnç değeri alınmamıştır. Bu yöntem çalışmanın süresini çok uzattığı için araştırma öncesi ortalama bir kalibrasyon değeri bulunmaya çalışılmıştır. 20 kişiden alınan değerlerin sonucu olarak 330 k Ω kalibrasyon direnci uygun görülmüştür (Şekil 3.2.13).



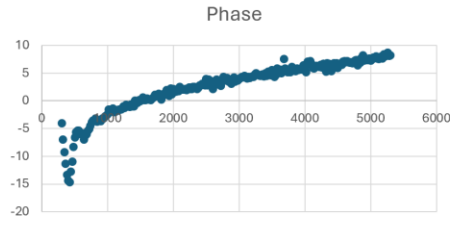
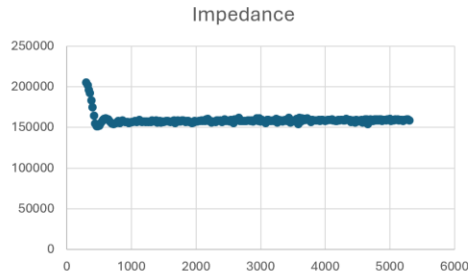
Şekil 3.2.13. 20 kişiden alınan avuç içi direnç değerleri

Nordbotten (2008) çalışmasında düşük frekanslar için AD5933 için kullanılan 16Mhz kristal osilatörün yeterli olmadığını belirtmiştir. Düşük frekanslarda empedans ve faz açısını daha iyi gözlemlene şansı olduğundan geliştirme kartına 4Mhz ve 2Mhz olmak üzere iki adet osilatör eklenmiştir. Ancak 2Mhz osilatör ile yapılan ölçümler oldukça belirsiz ve arayüzün yazılımının tanım aralığında bulunmayan sonuçlar vermiştir. Bu nedenle düşük frekansların ölçümü 4Mhz kristal osilatör ile yapılmıştır.

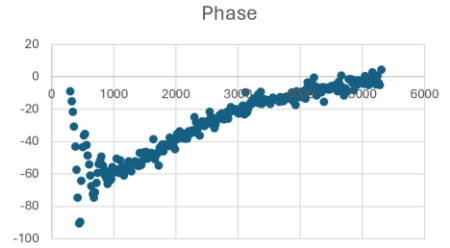
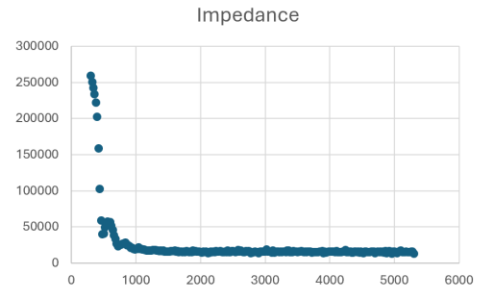
Arayüz yazılımında seçilebilecek en yüksek çıkış tepeden tepeye gerilimi 2V, en düşüğü ise 0.2V'dir. Karşılaştırılacak olursa 2V çıkış tepeden tepeye geriliminde devrenin empedans ve faz cevabı daha düzeltilmiş, 0.2V çıkış tepeden tepeye geriliminde ise ufak empedans ve faz değişiklikleri daha belirgin görülmektedir. 2V tepeden tepeye olan gerilim çıkışında ayrıntılar zor görülmektedir, ayrıca 4Mhz ile yapılan ölçümlerde devre doyuma gitmekte ve bir süre sonra aynı empedans değerlerini vermektedir.



Şekil 3.2.14. 16Mhz ile yapılan farklı gerilim çıkışlarının örnek empedans ve faz sonuçları



2Vpp



0.2Vpp

Şekil 3.2.15. 4Mhz ile yapılan farklı gerilim çıkışlarının örnek empedans ve faz sonuçları

Şekil 3.2.14 ve Şekil 3.2.15'te görüldüğü gibi 2V çıkış gerilimi hem veri azlığı hem de devrenin doyuma girmesine neden olduğu için çalışmadan çıkarılmıştır.

Düşük frekans 300Hz-5300Hz aralığında belirlenmiştir. Yüksek frekans ise 5000Hz-65000Hz aralığındadır. Her empedans ölçümü 200Hz aralığında yapılmıştır. Dolayısıyla 300Hz ile 5300Hz arasında 251 adet empedans ve faz açısı ölçümü yapılmıştır. 5Khz ile 65Khz arasında ise 301 adet empedans ve faz ölçümü yapılmıştır. Frekans tarama parametre ve diğer devre ayarları Çizelge 3.2.3'te gösterilmiştir.

Çizelge 3.2.3. Arayüz parametre tanımlamaları

Tarama Parametreleri							
Başlangıç Frekansı (Hz)	Frekans Aralığı (Hz)	Aralık Sayısı	Saat Frekans (MHz)	Sistem Saati	Uyarıcı Gerilim(V)	Kazanç	Kazanç Faktörü Kalibrasyonu
300	200	250	4	Dışarıdan	0.2	1	Çoklu nokta Frekans Kalibrasyonu
5000	200	300	16	Dışarıdan	0.2	1	Çoklu nokta Frekans Kalibrasyonu

Ölçümler frekans tarama şeklinde olduğundan ‘Kazanç Faktörü Kalibrasyonu’ çoklu nokta şeklinde seçilmiştir. Ayrıca parametre ayarlarında belirtilen ‘Kazanç’ bölümü ‘5’ olarak seçildiğinde, devre her ölçümde doyuma ulaşmıştır. O nedenle ‘Kazanç’ ‘1’ seçilmiştir.

3.3. Öznitelik Seçimi ve Veri Önışleme

Soft biyometrik ve biyoelektrik empedans verileri toplandıktan sonra verilere uç değerler kontrolü yapılmıştır. Kaydedilen empedans ve faz değerlerinin uç değerleri çok fazla olmayıp veri setindeki bir önceki ve bir sonraki değerlerin aritmetik ortalaması ile anlamlı veri noktaları oluşturulmuştur.

Soft biyometrik verilerinde ise çizimlerin hızlı olarak çizilmesinin bir sonucu olarak yanlış yapıldığından dolayı bazı gönüllülerin deneme sayısı 15’ten az olarak değerlendirilmiştir. Bu beklenen bir durumdur ve 15 tekrar sayısının belirlenmesindeki ana sebeplerdendir.

Uç veriler temizlendikten sonra bazı istatistiksel özellikler öznitelik olarak belirlenmiştir. Literatür çalışması yapılırken birçok araştırmada yer alan bu özniteliklerin bir kısmı bu çalışmada da yer almıştır. Bunlar standart sapma, ortalama, medyan, maksimum, minimum değer ve çeyrekler açıklığıdır.

Standart Sapma: Bir rastgele değişkenin ortalama etrafında sahip olacağı tahmin edilen varyasyon miktarı, standart sapması ile ölçülür. Düşük bir standart sapma, değerlerin beklenen değer olarak da bilinen kümenin ortalamasına yakın olma eğilimini belirtirken, büyük bir standart sapma daha geniş bir değer aralığını belirtir. Neyin aykırı değer olarak sayılıp neyin sayılmadığına karar verirken standart sapma sıklıkla kullanılır.

$$s = \sqrt{\frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n - 1}} \quad (3.1)$$

X_i : Veri setindeki her bir veri

$\bar{X}$: Verilerin aritmetik ortalaması

s : Standart sapma

n : Örneklem sayısı

Medyan: Bir popülasyonun, olasılık dağılımının veya bir veri örneğinin üst ve alt yarısını bölen değere medyan denir. Bir veri kümesi için "orta" değer olarak kabul edilebilir.

$$\frac{(n + 1)}{2} \quad (3.2)$$

n : Toplam kişi/veri sayısı

Ortalama: Veri setindeki tüm değerlerin toplanıp veri noktası sayısına verilen isimdir. Aritmetik ortalama şeklinde de ifade edilir.

$$\bar{x} = \frac{1}{n} \left(\sum_{i=1}^n x_i \right) \quad (3.3)$$

$\bar{x}$: Aritmetik ortalama

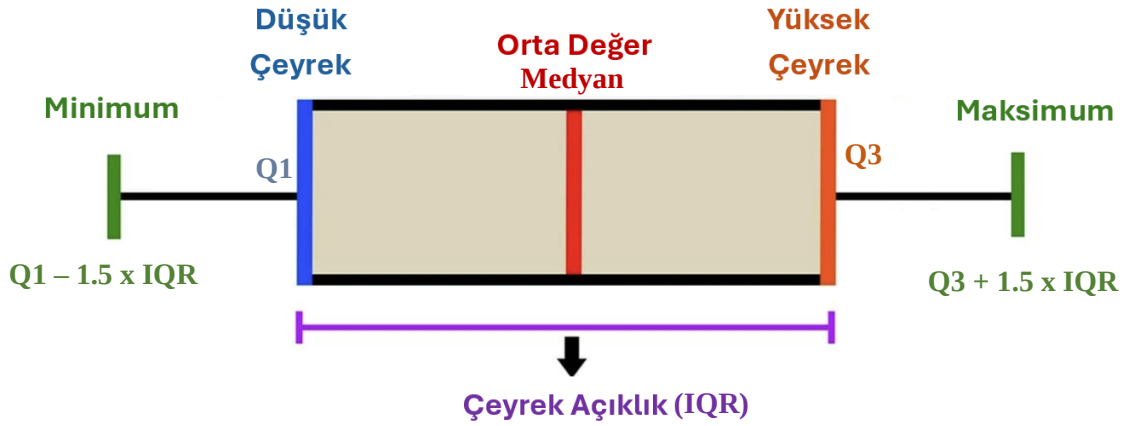
n : Popölasyondaki veri sayısı

x_i : Popölasyondaki her bir veri değeri

Maksimum değeri: Veri noktalarındaki en yüksek değeri.

Minimum değeri: Veri noktalarındaki en düşük değeri.

IQR-Çeyrek açıklığı: Çeyrekler açıklığı veri setinin medyanının her iki tarafından eşit şekilde, veri setinin %50'sini kapsayan ve veri setinin ilk çeyrek ile son çeyrek farkını gösteren istatistiksel bir yayılma ölçüsüdür. Çeyrek açıklık ölçüsü uç değerlerden etkilenmediği için iyi bir istatistiksel yöntemdir.



Şekil 3.3.1. Çeyrek açıklık kutu diyagramı

Şekil 3.3.1.'de ifade edilen 'Maksimum' ve 'Minimum' değerleri, düşük ve yüksek çeyrek değerlerine 1.5 IQR değeri eklenmesi ve çıkarılmasıyla elde edilir. Bu değerlerin daha ötesinde tanımlanan veri değerleri varsa onlar uç değer olarak kabul edilir ve işleme alınmazlar.

Verilerin bu istatistiksel deęerleri hesaplandıktan sonra birbirleriyle uyumlu olabilmesi için bir ön işlem yöntemi olan normalizasyon işlemine tabi tutulmuşlardır.

$$X_{normal} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (3.4)$$

Normalizasyon işlemi sonrası bütün deęerler $[-1,1]$ aralığına gelmiş ve deęerlendirme modelin deęerlendirme işini kolaylaştırmıştır.

Çizelge 3.2.1'den görüleceęi gibi yaş grupları ve cinsiyetlerle birlikte gönüllüler 16 farklı gruba ayrılmış ve denetimli öğrenme yöntemine göre etiketlenmişlerdir.

Bütün bu öznitelik seçimi, etiketleme ve normalizasyon işlemi sonrasında, veriler tek bir Excel belgesine eklenmiş ve analiz edilmeye hazır hale getirilmiştir.

Çizelge 3.3.1. Soft biyometri öznitelikleri

Öznitelik İsmi	Açıklaması
L-R Time	Soldan sağa kaydırma ortalama zamanı
R-L Time	Sağdan sola kaydırma ortalama zamanı
DownTime	Aşağıya doğru kaydırma ortalama zamanı
UpTime	Yukarı doğru kaydırma ortalama zamanı
TricwTime	Saat yönünde üçgen ortalama çizme süresi
TriccwTime	Saat yönünün tersinde üçgen ortalama çizme süresi
CircwTime	Saat yönünde çember ortalama çizme süresi
CirccwTime	Saat yönünün tersinde çember ortalama çizme süresi
L-R Press	Soldan sağa kaydırma ortalama basınç değeri
R-L Press	Sağdan sola kaydırma ortalama basınç değeri
DownPress	Aşağıya doğru kaydırma ortalama basınç değeri
UpPress	Yukarı doğru kaydırma ortalama basınç değeri
TricwPress	Saat yönünde üçgen çizerken ortalama basınç değeri
TriccwPress	Saat yönünün tersinde üçgen çizerken ortalama basınç değeri
CircwPress	Saat yönünde çember çizerken ortalama basınç değeri
CirccwPress	Saat yönünün tersinde çember çizerken ortalama basınç değeri
L-R DistX	Soldan sağa kaydırma x ekseninde ortalama katedilen mesafe
R-L DistX	Sağdan sola kaydırma x ekseninde ortalama katedilen mesafe
DownDistX	Aşağıya doğru kaydırma x ekseninde ortalama katedilen mesafe
UpDistX	Yukarı doğru kaydırma x ekseninde ortalama katedilen mesafe
TricwDistX	Saat yönünde üçgen çizerken x ekseninde ortalama katedilen mesafe
TriccwDistX	Saat yönü tersinde üçgen çizerken x ekseninde ortalama katedilen mesafe
CircwDistX	Saat yönünde çember çizerken x ekseninde ortalama katedilen mesafe
CirccwDistX	Saat yönü tersinde çember çizerken x ekseninde ortalama katedilen mesafe
L-R DistY	Soldan sağa kaydırma x ekseninde ortalama katedilen mesafe
R-L DistY	Sağdan sola kaydırma x ekseninde ortalama katedilen mesafe
DownDistY	Aşağıya doğru kaydırma x ekseninde ortalama katedilen mesafe
UpDistY	Yukarı doğru kaydırma x ekseninde ortalama katedilen mesafe
TricwDistY	Saat yönünde üçgen çizerken x ekseninde ortalama katedilen mesafe
TriccwDistY	Saat yönü tersinde üçgen çizerken x ekseninde ortalama katedilen mesafe
CircwDistY	Saat yönünde çember çizerken x ekseninde ortalama katedilen mesafe
CirccwDistY	Saat yönü tersinde çember çizerken x ekseninde ortalama katedilen mesafe

Çizelge 3.3.2. Empedans ve faz açılarından elde edilen öznitelikler

Özniteliğin Elde Edildiği Veri	Öznitelikler	Açıklama
Düşük Frekans Empedans	STD IMP LOW	Standart sapma değeri
	MEAN IMP LOW	Ortalama değer
	MED IMP LOW	Medyan değeri
	MAX IMP LOW	En yüksek değer
	MIN IMP LOW	En düşük değer
	IQR IMP LOW	Çeyrek açıklık değeri
Düşük Frekans Faz Açısı	STD PH LOW	Standart sapma değeri
	MEAN PH LOW	Ortalama değer
	MED PH LOW	Medyan değeri
	MAX PH LOW	En yüksek değer
	MIN PH LOW	En düşük değer
	IQR PH LOW	Çeyrek açıklık değeri
Yüksek Frekans Empedans	STD IM HIGH	Standart sapma değeri
	MEAN IMP HIGH	Ortalama değer
	MED IMP HIGH	Medyan değeri
	MAX IMP HIGH	En yüksek değer
	MIN IMP HIGH	En düşük değer
	IQR IMP HIGH	Çeyrek açıklık değeri
Yüksek Frekans Faz Açısı	STD PH HIGH	Standart sapma değeri
	MEAN PH HIGH	Ortalama değer
	MED PH HIGH	Medyan değeri
	MAX PH HIGH	En yüksek değer
	MIN PH HIGH	En düşük değer
	IQR PH HIGH	Çeyrek açıklık değeri

Çizelge 3.3.1 ve Çizelge 3.3.2’de gösterildiği gibi, 32 adet soft biyometrik ölçümlerinden, 24 adet biyoelektrik empedans ölçümlerden olmak üzere toplamda 56 adet öznitelik elde edilmiştir.

4. BULGULAR

Her gönüllü için hesaplanmış olan 56 öznitelik .csv uzantılı Excel dosyası ile MATLAB çalışma alanına yüklendikten sonra, ‘sınıflandırma öğrenici’ uygulamasında makine öğrenme performansı ölçülmüştür.

Çapraz Doğrulama parametresi 5 olarak seçilmiştir. Öznitelik kısmından gönüllülerin ismi, yaş ve cinsiyet bilgileri çıkarılmıştır. Gönüllü ismi nümerik bir veri olmadığından, kullanılamamaktadır. Yaş ve cinsiyet de ölçümü olmayan ayrıca etiketleri belirlediğimiz temel veriler olduğundan kaldırılmıştır. Diğer öznitelikler ile etiket sınıflandırılması yapılmıştır.

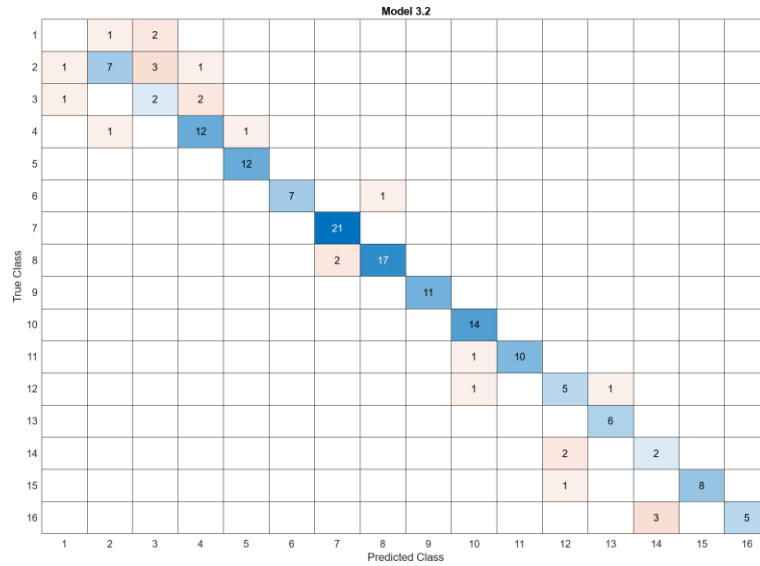
4.1. Sınıflandırma Sonuçları

Sınıflandırma için üç farklı makine öğrenme algoritması (DVM, Mantıksal Regresyon ve K-En yakın komşular) belirlenmiştir. Bu çalışmada ilgili sınıflandırıcı algoritmaları farklı çekirdek fonksiyonları kullanılarak MATLAB ortamında test edilmiş ve sınıflandırma başarıları karşılaştırılmıştır. En etkili hiper parametre setlerini bulabilmek için ızgara arama çapraz doğrulama metodu kullanılmıştır. Şekil 4.1.1’de tanımlanan sınıflandırma modellerinin başarı oranları gösterilmiştir. Gözlem sayılarına bağlı olan karmaşıklık matrisleri DVM için Şekil 4.1.2’de, KNN için Şekil 4.1.4’te ve mantıksal regresyon için Şekil 4.1.6’da gösterilmiştir. Yüzde başarı oranına göre karmaşıklık matrisleri ise DVM için Şekil 4.1.3’te, KNN için Şekil 4.1.5’te ve mantıksal regresyon için Şekil 4.1.7’de gösterilmiştir.

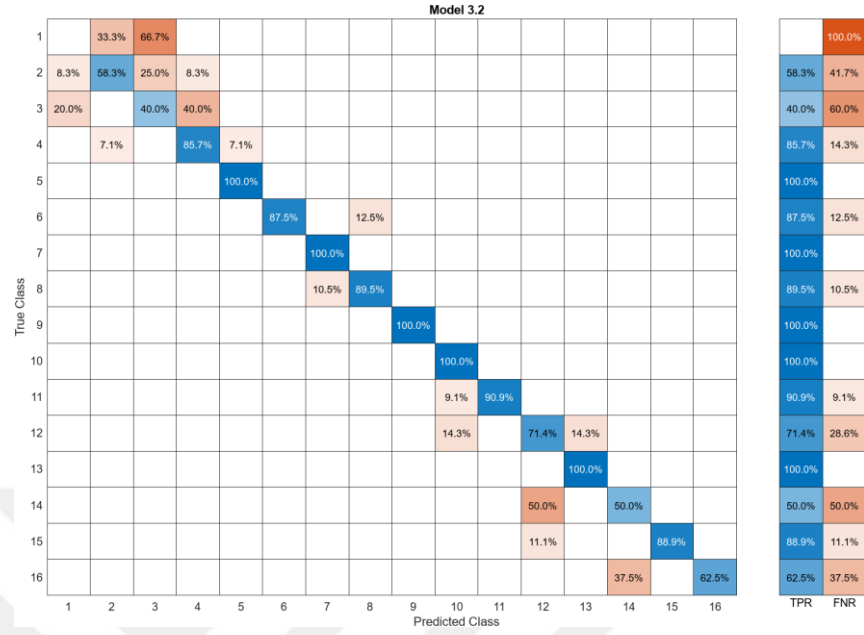
3.2 SVM	Accuracy (Validation): 84.8%
Last change: Quadratic SVM	56/56 features
3.3 SVM	Accuracy (Validation): 82.9%
Last change: Cubic SVM	56/56 features
3.5 SVM	Accuracy (Validation): 79.3%
Last change: Medium Gaussian SVM	56/56 features
3.1 SVM	Accuracy (Validation): 77.4%
Last change: Linear SVM	56/56 features
4.1 KNN	Accuracy (Validation): 63.4%
Last change: Fine KNN	56/56 features
4.6 KNN	Accuracy (Validation): 62.2%
Last change: Weighted KNN	56/56 features
2 Efficient Logistic Regression	Accuracy (Validation): 47.0%
Last change: Efficient Logistic Regression	56/56 features
4.2 KNN	Accuracy (Validation): 45.7%
Last change: Medium KNN	56/56 features
3.6 SVM	Accuracy (Validation): 43.3%
Last change: Coarse Gaussian SVM	56/56 features
4.4 KNN	Accuracy (Validation): 41.5%
Last change: Cosine KNN	56/56 features
4.5 KNN	Accuracy (Validation): 37.8%
Last change: Cubic KNN	56/56 features
3.4 SVM	Accuracy (Validation): 28.7%
Last change: Fine Gaussian SVM	56/56 features
4.3 KNN	Accuracy (Validation): 12.8%
Last change: Coarse KNN	56/56 features

Şekil 4.1.1. Sınıflandırma sonuçları

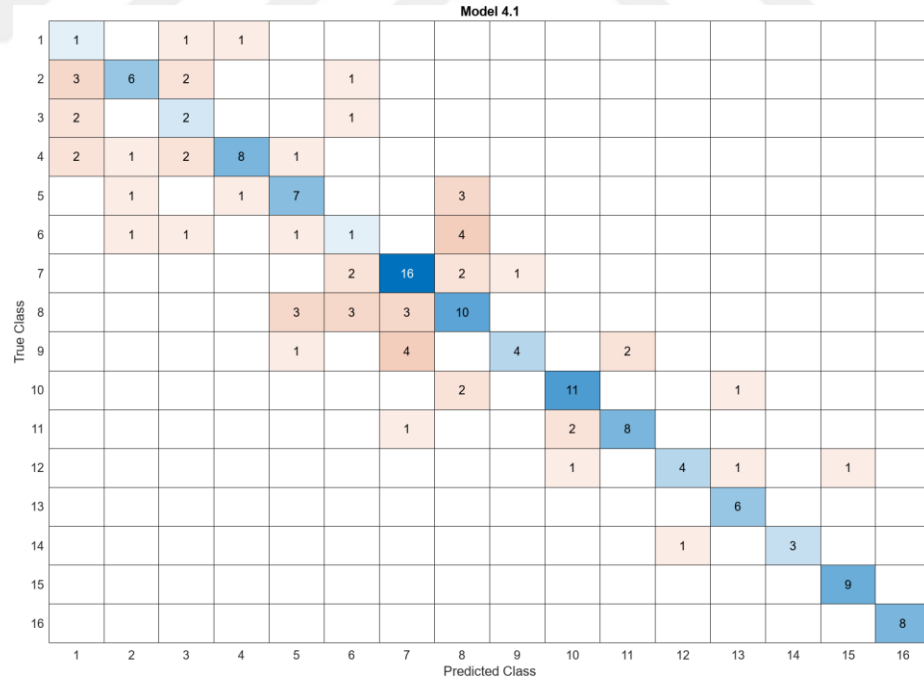
Şekil 4.1.1’de de görüldüğü gibi Kuadratik Çekirdek işlemlili DVM %84,8 doğruluk başarısı yakalamıştır. Bu sonuç sayesinde sınıflandırma arasında en iyi model olmuştur. K-en yakın komşu modellerinden en başarılı olanı %63.4’lük bir başarı oranı ile ‘komşu sayısı’ 1 olan ‘İnce (Fine) KNN’ modeli olmuştur. ‘Etkili Mantıksal Regresyon’ modelinin başarısı %47.0’te kalmıştır.



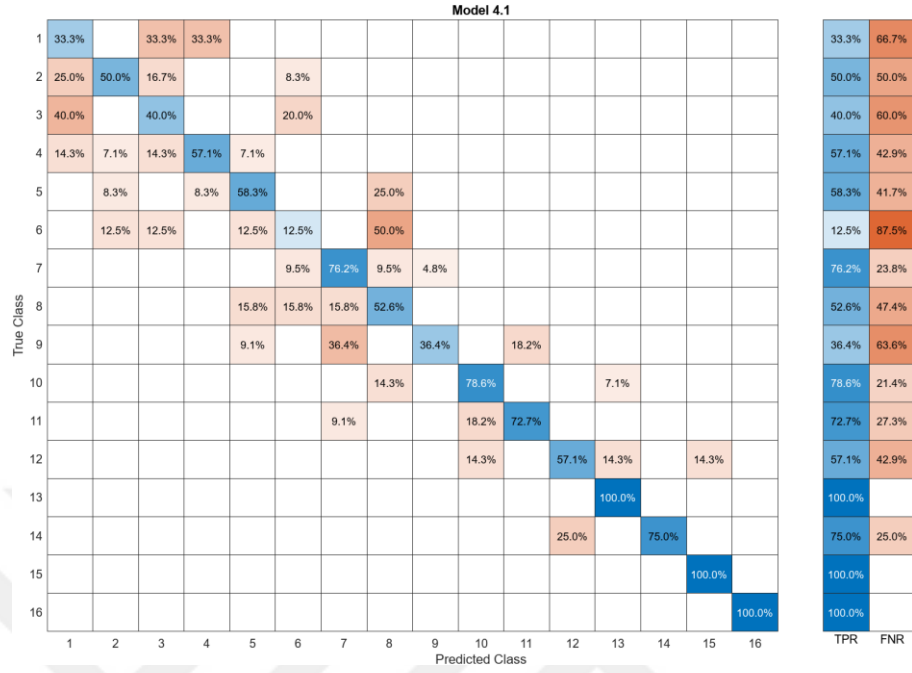
Şekil 4.1.2. Gözlem sayısına bağlı Kuadratik DVM sınıflandırıcısına ait karmaşıklık matrisi



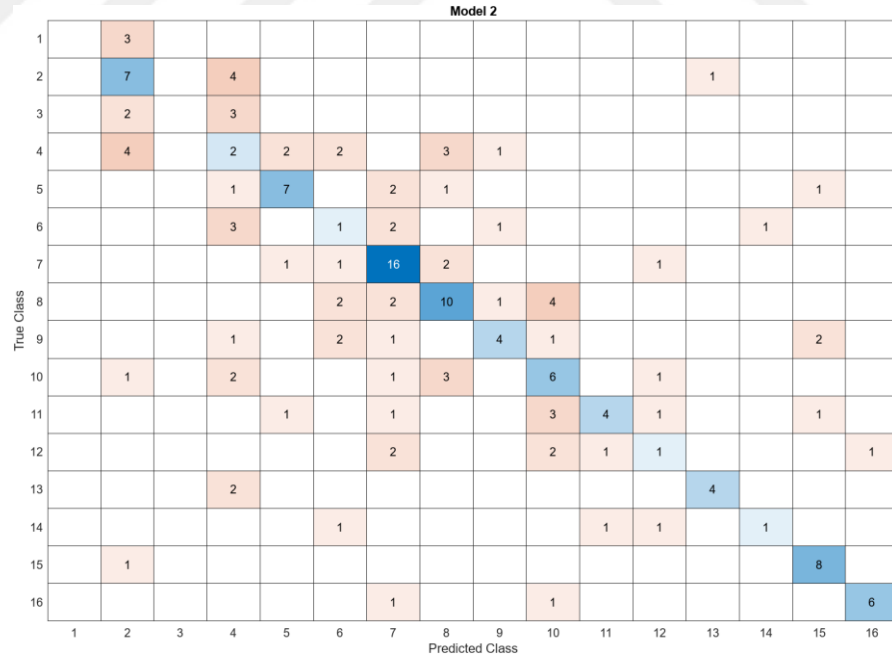
Şekil 4.1.3. Yüzde başarımla orantılı Kuadratik DVM sınıflandırıcısına ait karmaşıklık matrisi (Gerçek pozitifler ve yanlış negatifler)



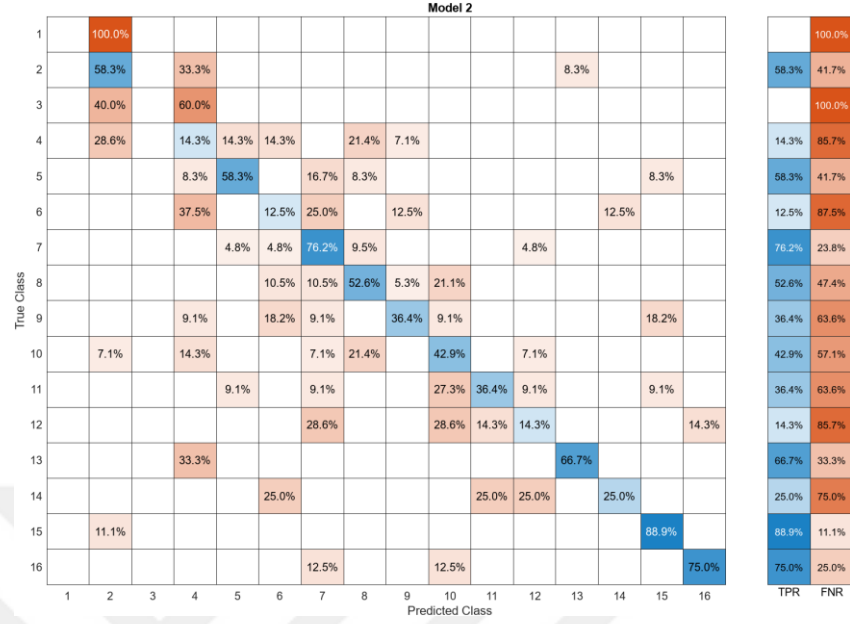
Şekil 4.1.4. Gözlem sayısına bağlı ince KNN sınıflandırıcısına ait karmaşıklık matrisi



Şekil 4.1.5. Yüzde başarımla oranlı ince KNN sınıflandırıcısına ait karmaşıklık matrisi (Gerçek pozitifler ve yanlış negatifler)



Şekil 4.1.6. Gözlem sayısına bağlı etkili mantıksal regresyon sınıflandırıcısına ait karmaşıklık matrisi



Şekil 4.1.7. Yüzde başarımlı etkili mantıksal regresyon sınıflandırıcısına ait karmaşıklık matrisi (Gerçek pozitifler ve yanlış negatifler)

4.2. Regresyon Sonuçları

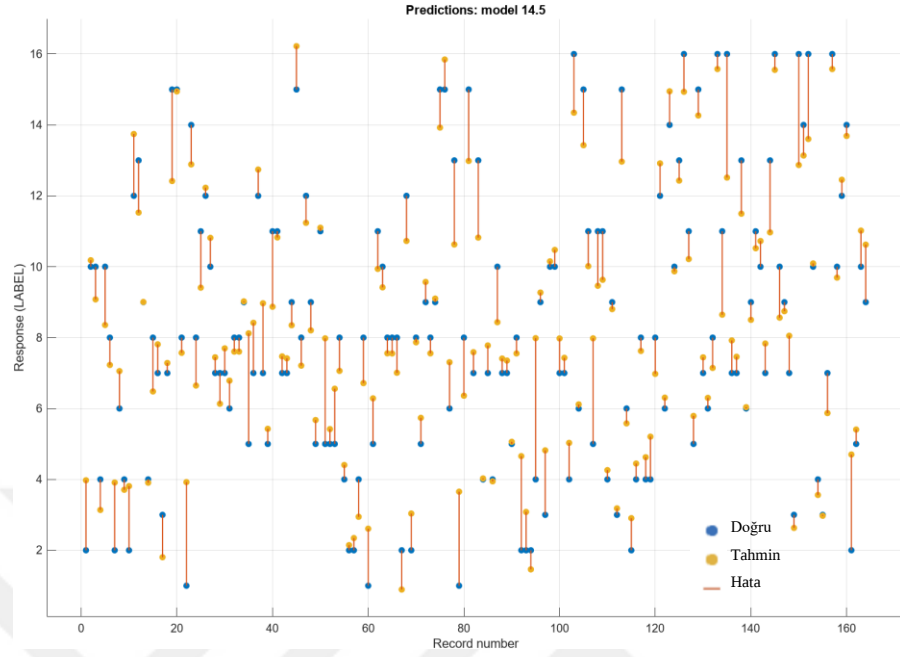
Regresyon yönteminde DVR ve doğrusal regresyon olmak üzere iki farklı model kullanılmıştır. DVR modelinde farklı çekirdek yöntemleri denenmiştir. Doğrusal regresyon modellerinde ise farklı hiper parametrelı regresyon modelleri denenmiştir. Burada da ızgara arama çapraz doğrulama tekniğı kullanılmıştır. Şekil 4.2.1.'de regresyon modellerinin başarımlı oranları gösterilmiştir. Regresyon için cevap grafikleri orta ölçek Gauss çekirdek işlevli DVR yöntemi için Şekil 4.2.2'de, etkileşimli doğrusal regresyon yöntemi için 4.2.4'te gösterilmiştir. Bu iki grafik gerçek sınıfla tahmin edilen sınıf arasındaki mesafeyi ölçerek hata oranını gösterir. Beklenen ve gerçekleşen cevaplar grafikleri ise; orta ölçek Gauss çekirdek işlevli DVR yöntemi için Şekil 4.2.3'te, etkileşimli doğrusal regresyon yöntemi için 4.2.5'te gösterilmiştir. Bu iki grafikte gösterilen 16 gerçek sınıf (x-ekseni) ve 16 tahmin edilen sınıf (y-ekseni) üzerinden tahmin dağılımı gösterilmiştir. Tahminin doğruluğı, model tarafından çizilen kusursuz regresyon çizgisi üzerinden değerlendirilir.

☆ 14.5 SVM	RMSE (Validation): 1.2584
Last change: Medium Gaussian SVM	56/56 features
☆ 14.1 SVM	RMSE (Validation): 1.5267
Last change: Linear SVM	56/56 features
☆ 14.6 SVM	RMSE (Validation): 1.6734
Last change: Coarse Gaussian SVM	56/56 features
☆ 14.2 SVM	RMSE (Validation): 2.4649
Last change: Quadratic SVM	56/56 features
☆ 14.4 SVM	RMSE (Validation): 3.2973
Last change: Fine Gaussian SVM	56/56 features
☆ 14.3 SVM	RMSE (Validation): 11.076
Last change: Cubic SVM	56/56 features
☆ 16 Linear Regression	RMSE (Validation): 30.063
Last change: Interactions Linear	56/56 features
☆ 15 Linear Regression	RMSE (Validation): 213.79
Last change: Linear	56/56 features
☆ 17 Linear Regression	RMSE (Validation): 387.25
Last change: Robust Linear	56/56 features

Şekil 4.2.1. Regresyon modelleri başarı oranları

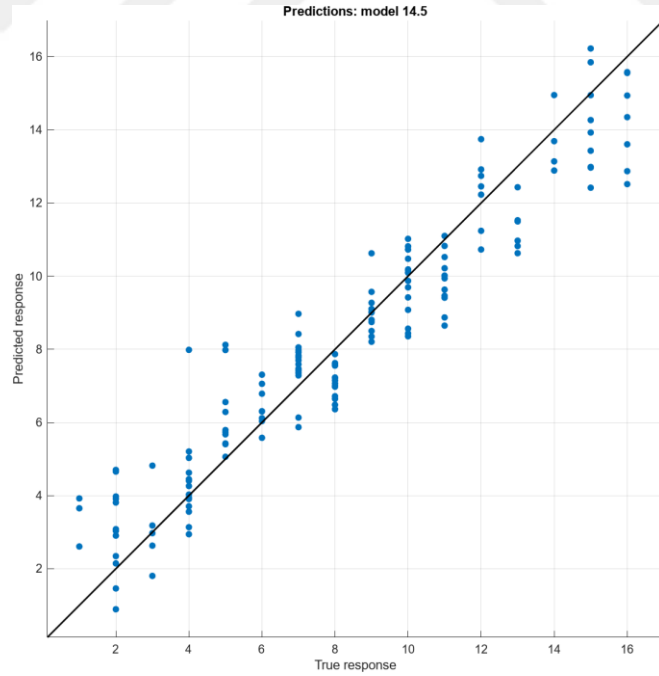
Regresyon modellerinde Şekil 4.2.1.'de görüldüğü üzere başarı oranı RMSE (Root Mean Square Error-Kök Ortalama Kare Hatası) üzerinden değerlendirilir. Bu değer tahmin edilen değer ile gerçek değer arasındaki beklenen ortalama fark olarak yorumlanabilir. Bu değer uç değerlere karşı hassas olduğu için bazı modellerde yüksek çıkmıştır. Ancak bazı modeller, eğitim verisi üzerinden uç verileri baskıladığında bu değer düşük çıkar. RMSE değeri ne kadar düşüğe, seçilen model o kadar iyi çalışıyor demektir. Dolayısıyla regresyon modellerini belirlemede oldukça önemli bir parametredir.

Alınan sonuçlara göre orta ölçekli Gauss çekirdek işlevli DVR regresyon modeli 1.2584 RMSE hatası ile en iyi model olarak belirlenmiştir. Daha sonrasında doğrusal DVR 1.5267 RMSE değeri ile ikinci, kaba ölçekli Gauss çekirdek işlevli DVR 1.6734 RMSE ile üçüncü, kuadratik çekirdek işlevli DVR 2.4649 RMSE ile dördüncü, ince ölçekli Gauss çekirdek işlevli DVR 3.2973 RMSE ile beşinci ve en sonunda kübik çekirdek işlevli DVR 11.076 RMSE ile altıncı en iyi performansa sahip DVR modeli olmuştur. Bu durumda genel olarak DVR modelleri regresyon modelleri içerisinde doğrusal regresyon ve diğer varyans modellerine göre daha başarılıdır.

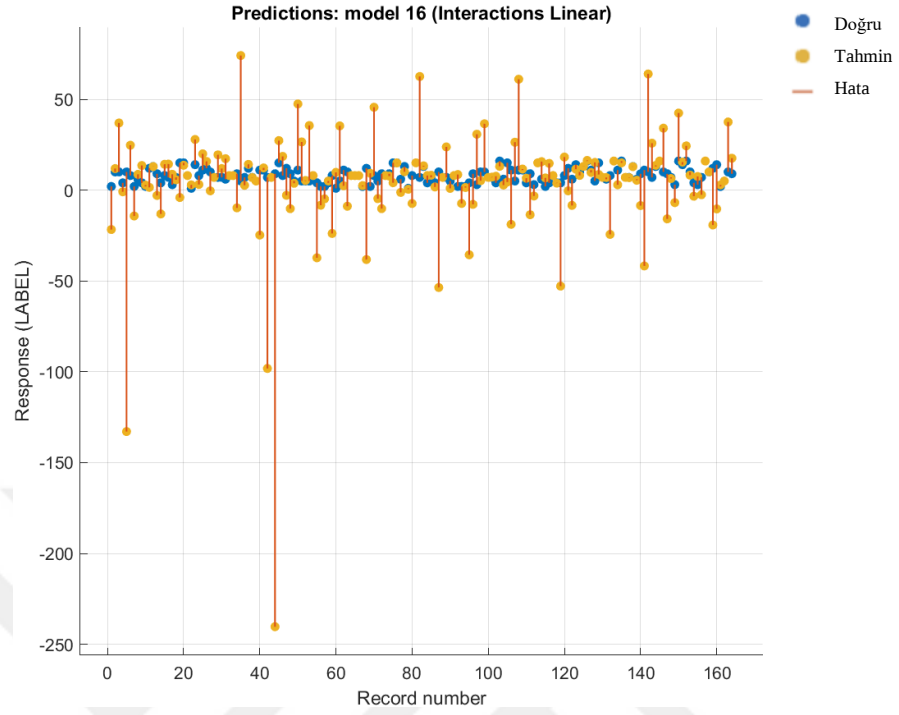


Şekil 4.2.2. Orta ölçek Gauss çekirdek işlevli DVR cevap grafiği

Şekil 4.2.2.'de düşey eksen etiketi temsil ederken yatay eksen kişi sırasını temsil eder.

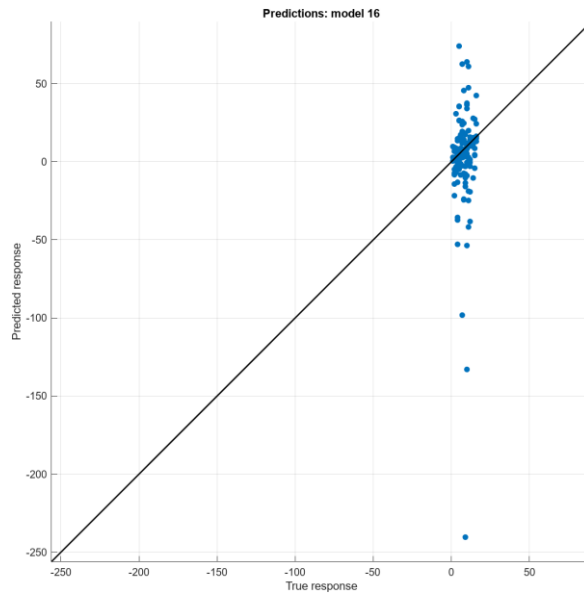


Şekil 4.2.3. Orta ölçek Gauss çekirdek işlevli DVR için beklenen ve gerçekleşen cevaplar



Şekil 4.2.4. Etkileşimli doğrusal regresyon grafiği

Şekil 4.2.4.'te düşey eksen etiketi temsil ederken yatay eksen kişi sırasını temsil eder.



Şekil 4.2.5. Etkileşimli doğrusal regresyon için beklenen ve tahmin edilen cevaplar

Hem sınıflandırma hem de regresyon modelleri için ızgara arama çapraz doğrulama yöntemi kullanılmadan aynı veri seti ile modellerin başarısı ve hata oranı hesaplanmıştır. Şekil 4.2.6.'da sınıflandırma başarıları ve regresyon kök ortalama kare hataları gösterilmiştir.

Models		
Sort by	Accuracy (Validation)	
3.2 SVM	Accuracy (Validation): 81.7%	
Last change: Quadratic SVM	56/56 features	
3.3 SVM	Accuracy (Validation): 81.1%	
Last change: Cubic SVM	56/56 features	
3.5 SVM	Accuracy (Validation): 78.0%	
Last change: Medium Gaussian SVM	56/56 features	
3.1 SVM	Accuracy (Validation): 77.4%	
Last change: Linear SVM	56/56 features	
4.6 KNN	Accuracy (Validation): 63.4%	
Last change: Weighted KNN	56/56 features	
4.1 KNN	Accuracy (Validation): 61.0%	
Last change: Fine KNN	56/56 features	
2 Efficient Logistic Regression	Accuracy (Validation): 43.3%	
Last change: Efficient Logistic Regression	56/56 features	
3.6 SVM	Accuracy (Validation): 42.7%	
Last change: Coarse Gaussian SVM	56/56 features	
4.2 KNN	Accuracy (Validation): 42.1%	
Last change: Medium KNN	56/56 features	
4.5 KNN	Accuracy (Validation): 36.6%	
Last change: Cubic KNN	56/56 features	
4.4 KNN	Accuracy (Validation): 35.4%	
Last change: Cosine KNN	56/56 features	
3.4 SVM	Accuracy (Validation): 26.8%	
Last change: Fine Gaussian SVM	56/56 features	
4.3 KNN	Accuracy (Validation): 12.2%	
Last change: Coarse KNN	56/56 features	A

Models		
Sort by	RMSE (Validation)	
4.5 SVM	RMSE (Validation): 1.2881	
Last change: Medium Gaussian SVM	56/56 features	
4.6 SVM	RMSE (Validation): 1.646	
Last change: Coarse Gaussian SVM	56/56 features	
4.1 SVM	RMSE (Validation): 2.3885	
Last change: Linear SVM	56/56 features	
4.4 SVM	RMSE (Validation): 3.2596	
Last change: Fine Gaussian SVM	56/56 features	
4.2 SVM	RMSE (Validation): 17.976	
Last change: Quadratic SVM	56/56 features	
3.2 Linear Regression	RMSE (Validation): 31.607	
Last change: Interactions Linear	56/56 features	
4.3 SVM	RMSE (Validation): 91.42	
Last change: Cubic SVM	56/56 features	
3.3 Linear Regression	RMSE (Validation): 202.3	
Last change: Robust Linear	56/56 features	
3.1 Linear Regression	RMSE (Validation): 211.74	
Last change: Linear	56/56 features	B

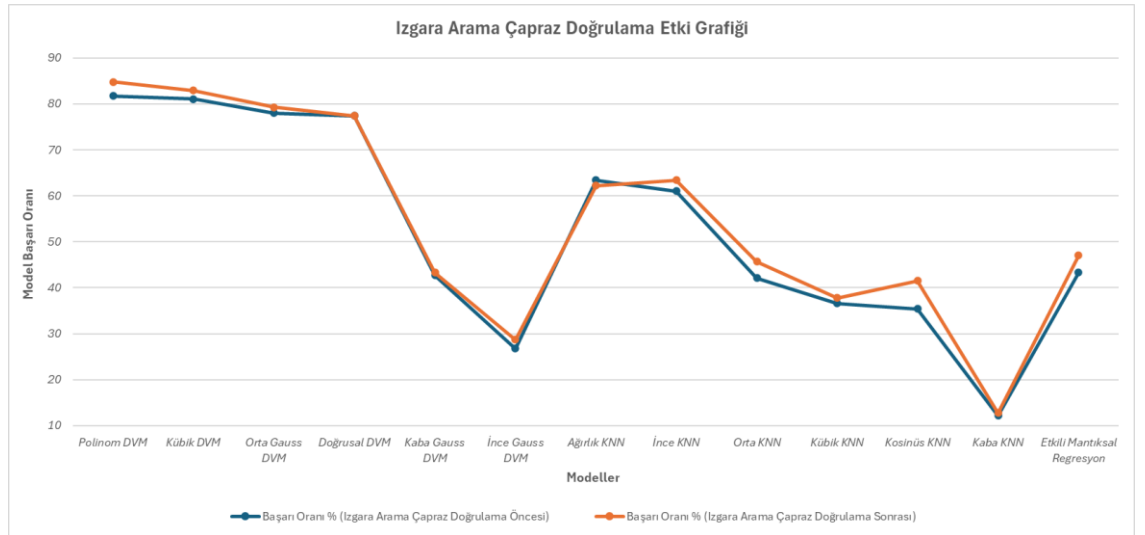
Şekil 4.2.6. A) Izgara arama çapraz doğrulama olmadan sınıflandırma modelleri başarısı. **B)** Izgara arama çapraz doğrulama olmadan yapılan regresyon modelleri hata oranı

5. TARTIŞMA ve SONUÇ

Yapılan testlerden sonra sınıflandırma modellerinden en iyi performansı kuadratik çekirdek işlevli DVM modeli göstermiştir (%84.8). Burada ikinci dereceden belirlenen bir fonksiyon hiper düzlemi belirlemiştir. Bu model uygulama zamanını kısaltılarak, çok fazla tahmin edici ve gözlemci içeren bir veri seti için uygun bir hale gelmiştir.

Regresyon modellerinde ise orta ölçek Gauss çekirdek işlevli regresyon modeli en iyi performansı göstermiştir (1.2584 RMSE). Oluşturulan kusursuz regresyon model çizgisinin etrafında veriler kabaca simetrik olarak dağılmıştır.

Optimizasyon yöntemi olarak kullanılan ızgara arama çapraz doğrulama yöntemi sınıflandırma modelleri için doğruluğu arttırmıştır. Şekil 5.1.'de sınıflandırma modelleri için doğrulama yöntemi uygulanmadan ve uygulandıktan sonraki performans verileri gösterilmiştir.



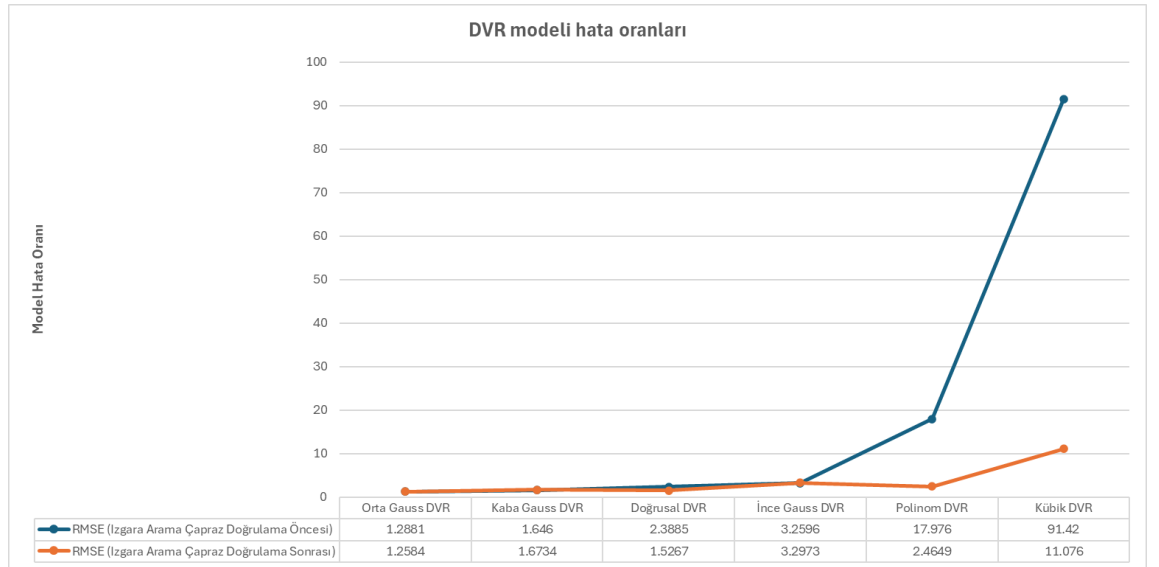
Şekil 5.1. Sınıflandırma modellerinin çapraz doğrulama uygulanmadan ve uygulandıktan sonraki başarı performans grafiği

Şekil 5.1.'den anlaşılacağı gibi ızgara arama çapraz doğrulama modeli sadece 'Doğrusal DVM' ve 'Ağırlık KNN' modellerinde etki gösterememiş; diğer tüm modellerde, modelin başarısını arttırmıştır.

Sınıflandırma modellerinde DVM'in başarısı, diğer modellerle karşılaştırıldığında modelin karmaşıklığı ve öz nitelik sayısının fazla olmasıyla ilişkilidir. Çekirdek yöntemleri modele esneklik sağlamış ve aşırı uyum riskini azaltırken en yüksek doğruluk seviyesine ulaşmıştır. Normalize edilmiş ve farklı katmanlarda bulunan; fakat doğrusal bir şekilde ayıramayan veri setini çekirdek yöntemleriyle başarılı bir şekilde ayırmıştır.

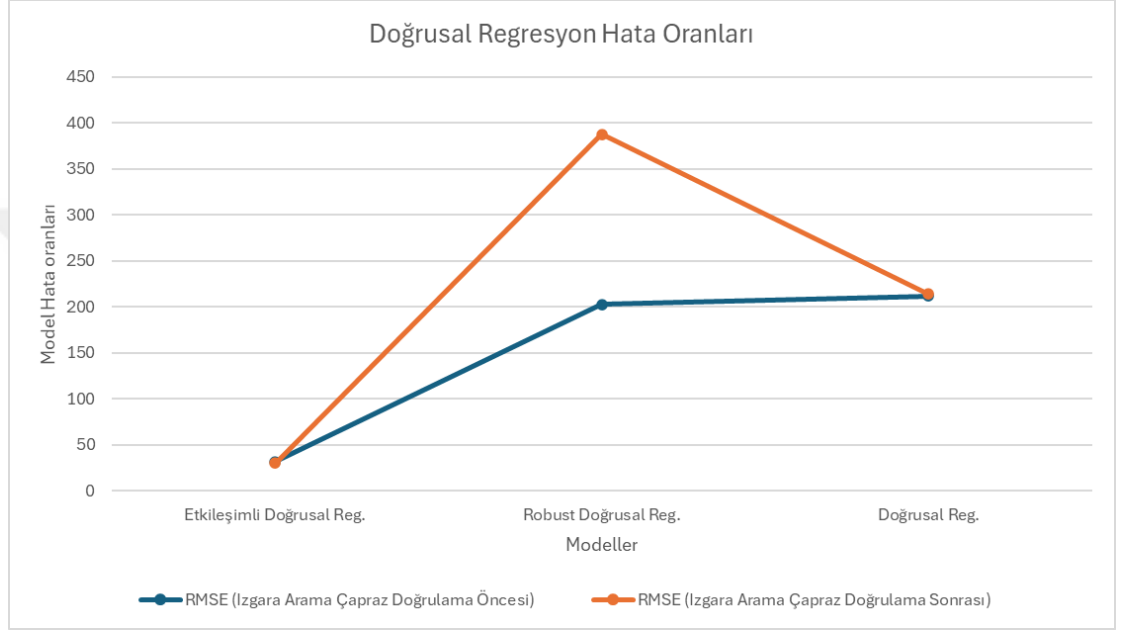
Buna karşılık birbirlerine çok yakın değerlere sahip olan öz nitelikler KNN tarafından, DVM kadar iyi sınıflandırılmamışlardır. Öz nitelik sayısı arttıkça noktalar arası mesafe KNN daha az anlamlı ve daha az ayırt edici olmuştur.

DVR modellerinin başarısı Şekil 5.2.'de karşılaştırılarak gösterilmiştir. Tüm modeller ızgara arama çapraz doğrulama yöntemi DVR için modelin hata oranını düşürmüştür.



Şekil 5.2. DVR modelinin çapraz doğrulama uygulanmış ve uygulanmamış haldeki hata oranları

Doğrusal regresyon modellerinde ise ızgara arama çapraz doğrulama yöntemi sadece etkileşimli doğrusal regresyon modellerinde hata oranını düşürebilmiştir (Şekil 5.3). Diğer regresyon modellerinde ise çapraz doğrulama yöntemi uygulandıktan sonra hata oranları yükselmiştir.



Şekil 5.3. Doğrusal regresyon modellerinde çapraz doğrulama yöntemi uygulanmış ve uygulanmamış haldeki hata oranları

Sonuçlara göre sınıflandırma modellerinde DVM ve regresyon modellerinde DVR sırasıyla en yüksek doğruluk oranı ve en düşük hata oranı sağlayarak, tercih sebebi olmuşlardır.

Bu tez çalışmasında kişilerin cinsiyet ve yaş sınıflandırması soft biyometrik veriler kullanılarak makine öğrenmesi yöntemleri ile yapılmıştır. Yapılan çalışma literatürde eşsiz ve soft biyometrik verilerin birlikte kullanıldığı çalışmalara alternatif biyoelektrik empedans ve soft biyometrik verilerin birlikte kullanıldığı özgün değere sahip bir çalışmadır.

Yapılan bu sınıflandırma ile oluşturulacak profil, kayıtlı kullanıcılar ile karşılaştırılarak kimlik doğrulama için kullanılabilir. Ayrıca belirlenen yaş aralığına göre cihazın kullanımı engellenebilir ya da kullanıcının bazı uygulamalara erişimine kısıtlama getirilebilir. Bu çalışma kişilerin cihaz kullanımlarında düşük güvenli kimlik doğrulama için eşsiz biyometrik verilerini paylaşma gerekliliğinin de soft biyometrik ve empedans veriler kullanarak ortadan kaldırılabilceğini göstermektedir. Aynı zamanda kişiye özgü eşsiz biyometrik verilerin ele geçirilmesi durumunda, ek güvenlik faktörü olarak bu verilerin kullanılabilceği görülmüştür.



KAYNAKLAR

- Acien, A., Aythami M., Fierrez, J., Vera-Rodriguez, R. ve Hernandez-Ortega, J. (2018). Active Detection of Age Groups Based on Touch Interaction. *IET Biometrics*. 8. doi:10.1049/iet-bmt.2018.5003.
- Ailisto H., Vildjiounaite E., Lindholm M., Mäkelä S.M., Peltola J. (2006). Soft biometrics—combining body weight and fat measurements with fingerprint biometrics, *Pattern Recognition Letters*, Volume 27, Issue 5, s. 325-334. doi: <https://doi.org/10.1016/j.patrec.2005.08.018>.
- Akça, M. F. (2020). Nedir Bu Destek Vektör Makineleri? (*Makine Öğrenmesi Serisi-2*) Erişim adresi: <https://medium.com/deep-learning-turkiye/nedir-bu-destek-vektör-makineleri-makine-öğrenmesi-serisi-2-94e576e4223e>
- Amidi A., ve Amidi S. (2018). Machine Learning tips and tricks cheatsheet. Erişim adresi: <https://stanford.edu/~shervine/teaching/cs-229/cheatsheet-machine-learning-tips-and-tricks>
- Angulo, J. ve Wästlund, E. (2012). Exploring Touch-Screen Biometrics for User Identification on Smart Phones. *IFIP Advances in Information and Communication Technology*. 375. s. 130- 143. doi:10.1007/978-3-642-31668-5_10.
- Aviv, A.J., Gibson, K., Mossop, E., Blaze, M., ve Smith, J.M. (2010). Smudge attacks on smartphone touch screens. *Proceedings of the 4th USENIX Conference on Offensive Technologies*, WOOT, s. 1–7. USENIX Association, Berkeley
- Ben-Hur, A., Ong, C. S., Sonnenburg, S., Schölkopf, B., ve Rätsch, G. (2008). Support Vector Machines and Kernels for Computational Biology. *PLoS computational biology*. 4. e1000173. doi: 10.1371/journal.pcbi.1000173.
- Bevan C., ve Fraser, D., (2016). Different Strokes for Different Folks? Revealing the Physical Characteristics of Smartphone Users from their Swipe Gestures. *International Journal of Human-Computer Studies*. 88. doi: 10.1016/j.ijhcs.2016.01.001.
- Bulut, F. (2024). Model Optimizasyonunda Yol Haritası: Grid Search Yöntemi ve Gelecek Perspektifler. Erişim adresi: <https://medium.com/@furkanbt/model-optimizasyonunda-yol-haritasi-grid-search-yöntemi-ve-gelecek-perspektifler-d14166c650b2>
- Canbolat E. (2018). Biyoelektrik İmpedans Analizi Parametrelerinden Faz Açısının, Tanısal Kriter Olarak Olası Rolü. *İnönü Üniversitesi Sağlık Bilimleri Dergisi*; 7 (1): s.58-65
- Candan H., (2021) Adım Adım Makine Öğrenmesi Bölüm 2: Denetimli Öğrenme Nedir? Erişim adresi: Adım Adım Makine Öğrenmesi Bölüm 2: Denetimli Öğrenme Nedir? | by Hatice Candan | Machine Learning Türkiye | Medium
- Çavdar T. Aymaz S., ve Aymaz S., (2024). A Framework for Elephant Flow Detection for SDNs Based on the Classification. *Arabian Journal for Science and Engineering*, Volume: 49, Issue:3. s. 424-4252

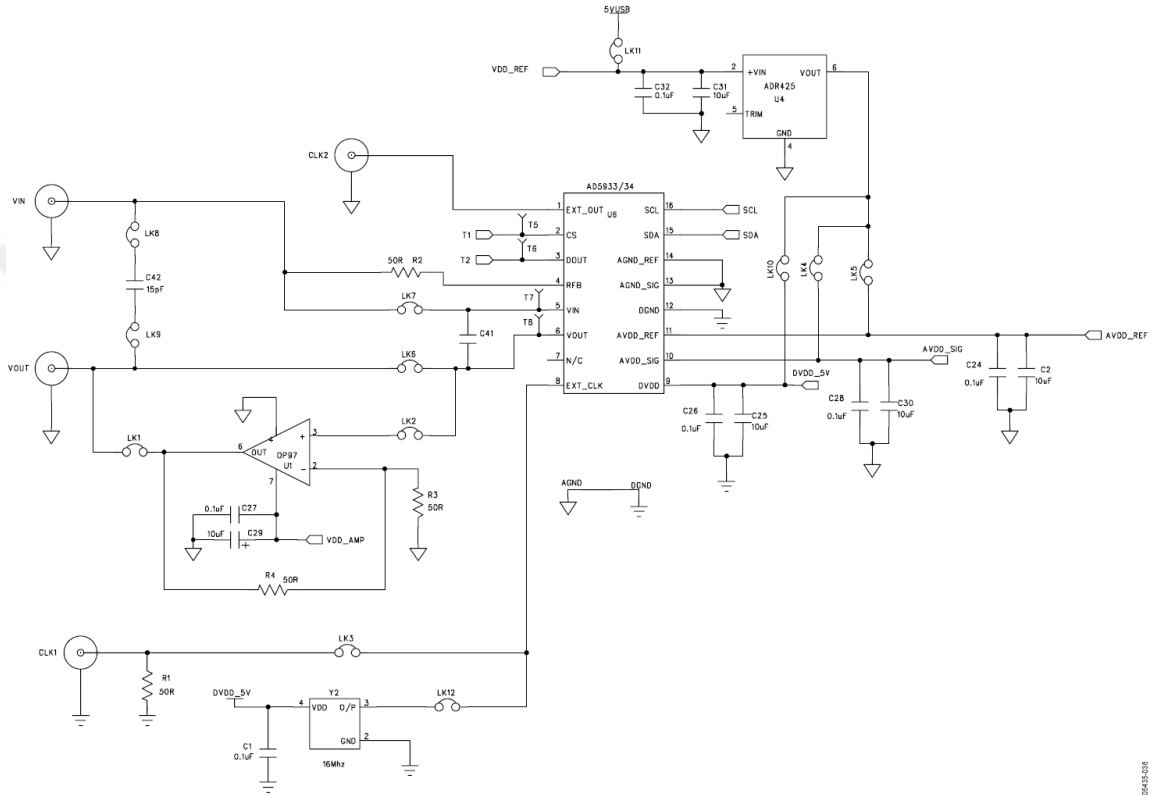
- De Marsico, Maria; Fartade, Eduard Gabriel; Mecca, Alessio (2018). "Feature-based Analysis of Gait Signals for Biometric Recognition - Automatic Extraction and Selection of Features from Accelerometer Signals". *Proceedings of the 7th International Conference on Pattern Recognition Applications and Methods*. SCITEPRESS- Science and Technology Publications. s. 630–637. doi:10.5220/0006719106300637
- Durna, M. B. (2020). Cross-Validation nedir? Nasıl çalışır? Erişim adresi: <https://medium.com/bili%C5%9Fim-hareketi/cross-validation-nedir-nas%C4%B1l-%C3%A7al%C4%B1%C5%9F%C4%B1r-4ec4736e5142>
- Durna, M.B. (2020). Hiperparametre Optimizasyonu. Erişim adresi: <https://medium.com/bili%C5%9Fim-hareketi/hiperparametre-optimizasyonu-9ba0e7f32e6f>
- Electrical Impedance (2009). *Vikipedi* içinde Erişim adresi: https://en.wikipedia.org/wiki/Electrical_impedance#/media/File:Complex_Impedance.svg
- Federal Financial Institutions Examination Council (2008). "Authentication in an Internet Banking Environment"
- Gong D. (2022). Top 6 Machine Learning Algorithms for Classification. Erişim adresi: <https://towardsdatascience.com/top-machine-learning-algorithms-for-classification-2197870ff501>
- Haafez, J. (2018). Regularization: the path to bias-variance trade-off. Erişim adresi: Regularization: the path to bias-variance trade-off | by HAFEEZ JIMOH | Towards Data Science
- Hoolbrok, R., ve Cook, A. (2024). Overfitting and Underfitting. Erişim adresi: <https://www.kaggle.com/code/ryanholbrook/overfitting-and-underfitting>
- Jiaying Y., Javed A. A. ve Gemma V.L. (2019). Support Vector Machines on Noisy Intermediate Scale Quantum Computers. doi:10.13140/RG.2.2.17956.63360.
- Joos, K. (2020). GridSearch: the ultimate Machine Learning Tool. Erişim adresi: <https://towardsdatascience.com/gridsearch-the-ultimate-machine-learning-tool-6cd5fb93d07>
- Karabudak, Ş. (2024). Destek Vektör Makinesi Algoritması. Erişim adresi: <https://suhedakarabudak.com/2024/01/20/support-vector-machine/>
- Kaushal, P., Nithin P., Pranav S., Koushik S., ve Koundinya, A. (2021). Myers-briggs Personality Prediction and Sentiment Analysis of Twitter using Machine Learning Classifiers and BERT. *International Journal of Information Technology and Computer Science*. 13. s.48-60. doi:10.5815/ijitcs.2021.06.04.
- Mahfouz, Ahmed; Mahmoud, Tarek M.; Eldin, Ahmed Sharaf (2017). "A survey on behavioral biometric authentication on smartphones". *Journal of Information Security and Applications*. 37: s.28–37 doi:10.1016/j.jisa.2017.10.002
- Makine öğrenmesi nedir? Erişim adresi: <https://azure.microsoft.com/tr-tr/resources/cloud-computing-dictionary/what-is-machine-learning-platform>
- Mansour, M., Martens, J., ve Blankenbach, J. (2023). Hierarchical SVM for Semantic Segmentation of 3D Point Clouds for Infrastructure Scenes. doi:10.20944/preprints202306.2255.v1.v

- Manton, N. (2023). Integration Theory for Kinks and Sphalerons in One Dimension. *Journal of Physics A: Mathematical and Theoretical*, Volume 57, Number 2 doi: 10.1088/1751-8121/ad14ac
- Mattaparthi G. (2023). K-Nearest Neighbors (KNN): A comprehensive guide. Eriřim adresi: <https://medium.com/@geethasreemattaparthi/k-nearest-neighbors-knn-a-comprehensive-guide-a83857ab2666>
- Miguel-Hurtado, O., V. Stevenage, S., Bevan, C., ve Guest, R. (2016). Predicting sex as a soft-biometrics from device interaction swipe gestures. *Pattern Recognition Letters*, 79, 44-51. doi: <https://doi.org/10.1016/j.patrec.2016.04.024>
- Nalçacıoğlu H. (2014). Çocukluk çağı nefrotik sendromlu hastalarda vücut sıvı hacminin belirlenmesinde biyoelektrik impedans analizi, Nt-probnp ve vena kava inferior sonografi incelemesinin rolü. *Uzmanlık Tezi*. Samsun: Ondokuz Mayıs Üniversitesi Tıp Fakültesi, Çocuk Sağlığı ve Hastalıkları Ana Bilim Dalı, Çocuk Nefroloji Bilim Dalı
- Nandakumar, K. ve Jain, A.K. (2009). Soft biometrics, in: S. Li, A. Jain (Eds.), *Encyclopedia of Biometrics*, Springer US, Boston, MA, s. 1235–1239.
- Niinuma K., Park U. ve Jain A.K., (2010) "Soft Biometric Traits for Continuous User Authentication," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 4, s. 771-780, doi: 10.1109/TIFS.2010.2075927.
- Özçetin M, Khalilova F, Kılıç A. (2017). Beslenme durumunun değerlendirilmesinde sıra dışı bir yöntem: *BİA. Çocuk Dergisi*; 17(2): s.61-6
- Park U., ve Jain A. K., (2010) "Face Matching and Retrieval Using Soft Biometrics," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 3, s. 406-415, doi: 10.1109/TIFS.2010.2049842.
- Patel, Vishal M.; Chellappa, Rama; Chandra, Deepak; Barbello, Brandon (July 2016). "Continuous User Authentication on Mobile Devices: Recent progress and remaining challenges". *IEEE Signal Processing Magazine*. 33 (4): 49–61. doi:10.1109/msp.2016.2555335
- Peeyada, P., Dutta, H., Shiangjen, K., Cholanjiak, W. (2022). A modified forward-backward splitting methods for the sum of two monotone operators with applications to breast cancer prediction. *Mathematical Methods in the Applied Sciences*. 46, n/a-n/a. doi:10.1002/mma.8578.
- Rzecki K., Pławiak P., Niedźwiecki M., Sońnicki T., Leńkow J. ve Ciesielski M., (2017) Person recognition based on touch screen gestures using computational intelligence methods. *Information Sciences*, Volumes 415–416, s. 70-84, ISSN 0020-0255, doi:10.1016/j.ins.2017.05.041.
- Shrimali K. R. (2018). SVM using Scikit-Learn in Python. Eriřim adresi: <https://learnopencv.com/svm-using-scikit-learn-in-python/>
- Sıvaz O., ve Aykut M. (2021). Touchscreen Swipe Based Biometric Authentication Using Spectral Regression - Kernel Discriminant Analysis. *29th Signal Processing and Communications Applications Conference (SIU)*, Istanbul, Turkey, s.1-4, doi: 10.1109/SIU53274.2021.9477815.
- Sindhu V. (2020). AN EMPIRICAL SCIENCE RESEARCH ON BIOINFORMATICS IN MACHINE LEARNING. *JOURNAL OF MECHANICS OF CONTINUA AND MATHEMATICAL SCIENCES*. spl7. 10.26782/jmcms.spl.7/2020.02.00006.

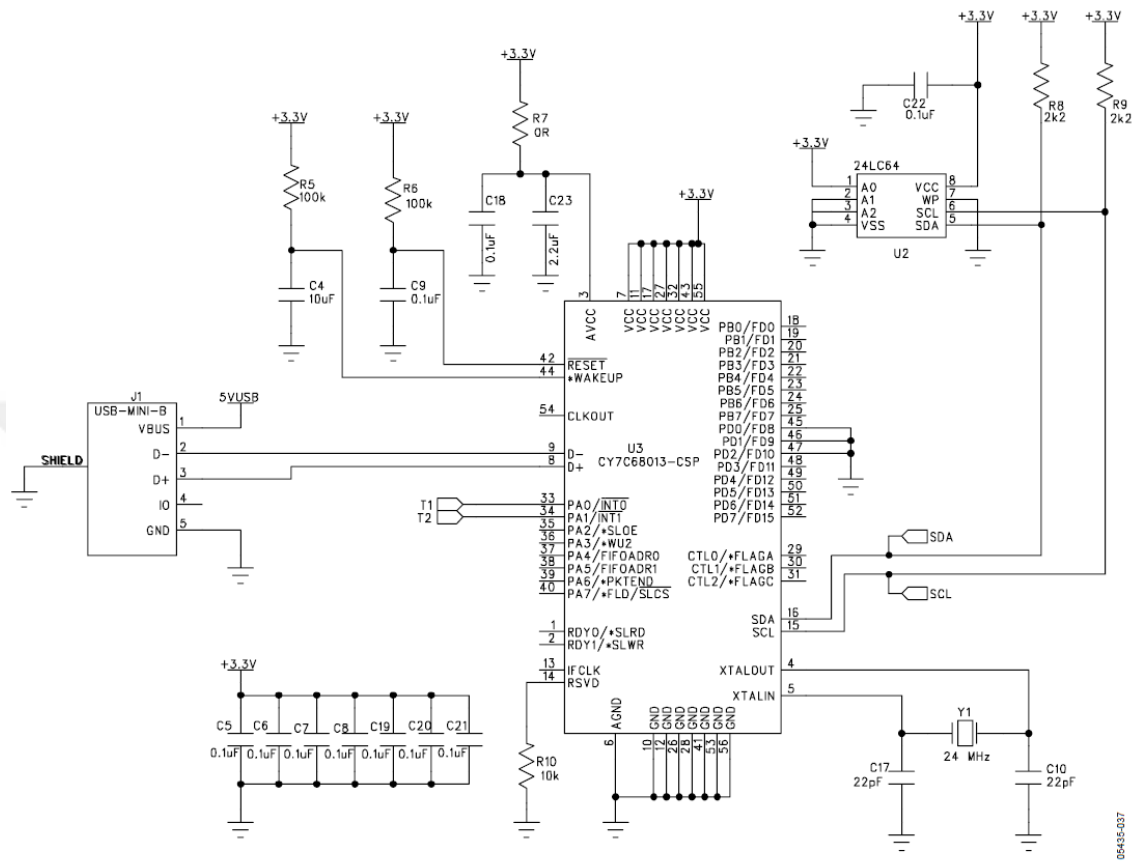
- Sreenivasa, S. (2020). Radial Basis Function (RBF) Kernel: The Go-To Kernel. Eriřim adresi: <https://towardsdatascience.com/radial-basis-function-rbf-kernel-the-go-to-kernel-acf0d22c798a>
- Tafralı, S. (2023). Ařırı Öğrenme Problemleri Nedir ve Nasıl Çözölür. Eriřim adresi: <https://miuul.com/blog/as%C4%B1r%C4%B1-ogrenme-problemleri-nedir-ve-nas%C4%B1l-%C3%A7ozulur>
- Taş, B. (2020). Makine Öğrenmesinde Overfitting / Underfitting / Best Fitting Kavramları. Eriřim adresi: <https://medium.com/kodluyoruz/makine-%C3%B6%C4%9Frenmesinde-overfitting-underfitting-best-fitting-kavramlar%C4%B1-9f80a5d42719>
- Trouvain, N., Pedrelli, L., Dinh, T. ve Hinaut, X., (2020). ReservoirPy: An Efficient and User-Friendly Library to Design Echo State Networks. doi:10.1007/978-3-030-61616-8_40.
- Wiedenbeck, S., Waters, J., Sobrado, L. ve Birget, J.C. (2006). Design and evaluation of a shoulder-surfing resistant graphical password scheme. *Proceedings of the WorkingConference on Advanced Visual Interfaces, AVI*, s. 177–184. ACM, New York
- Zewail R., Elsafi, A., Saeb M., ve Hamdy N., (2004) "Soft and hard biometrics fusion for improved identity verification," *The 2004 47th Midwest Symposium on Circuits and Systems, MWSCAS '04.*, Hiroshima, Japan, s. I-225, doi:10.1109/MWSCAS.2004.1353967.

EKLER

EK 1 AD5933EBZ şeması



04455-208



05435-037

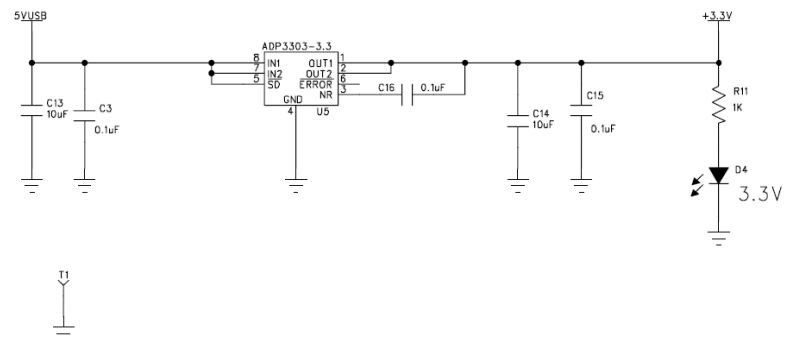


Figure 38.

