**T.C.**

**FIRAT ÜNİVERSİTESİ**

**MÜHENDİSLİK FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**EEG TABANLI VERİLERDEN YAŞ VE CİNSİYET ÇIKARIMI İÇİN DERİN ÖĞRENME TABANLI YENİ BİR YÖNTEM**

Abdulkadir Berat BUYRUK -205260025

Şevki Alperen ATAKAN -200260072

Tez Danışmanı

Prof. Dr. Mehmet KARAKÖSE

**BİTİRME TEZİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**ELAZIĞ - 2024**

**T.C.**

**FIRAT ÜNİVERSİTESİ**

**MÜHENDİSLİK FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**EEG TABANLI VERİLERDEN YAŞ VE CİNSİYET ÇIKARIMI İÇİN DERİN ÖĞRENME TABANLI YENİ BİR YÖNTEM**

Abdulkadir Berat BUYRUK - 205260025

Şevki Alperen ATAKAN - 200260072

**BİTİRME TEZİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

Bu tez …../…../2024 tarihinde aşağıda belirtilen jüri tarafından oybirliği/oyçokluğu ile başarılı/başarısız olarak değerlendirilmiştir.

Prof. Dr. Mehmet KARAKÖSE

# ÖZGÜNLÜK BİLDİRİMİ

Bu çalışmada başka kaynaklardan yapılan tüm alıntıların, ilgili kaynaklar referans gösterilerek açıkça belirtildiğini, alıntılar dışındaki bölümlerin, özellikle projenin ana konusunu oluşturan teorik çalışmaların ve yazılım/donanımının tarafımızdan yapıldığını bildiririm.

Fırat Üniversitesi 28/08/2024

Bilgisayar Mühendisliği Abdulkadir Berat BUYRUK

23119 Elazığ Şevki Alperen ATAKAN

# BENZERLİK BİLDİRİMİ

# TEŞEKKÜR

Bitirme projemiz, Fırat Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü’nde, Sayın Prof. Dr. Mehmet KARAKÖSE’nin yönlendirmesi ve gözetimi altında hazırlanmıştır. Projemizin hazırlaması sürecinde bilgi, görüş ve eleştirilerinden yararlandığımız hocamız Sayın Prof. Dr. Mehmet KARAKÖSE’ye en içten duygularımızla teşekkür ederiz.

Abdulkadir Berat BUYRUK

Şevki Alperen ATAKAN

**İÇİNDEKİLER**

[ÖZGÜNLÜK BİLDİRİMİ I](#_Toc176297700)

[BENZERLİK BİLDİRİMİ II](#_Toc176297701)

[TEŞEKKÜR III](#_Toc176297702)

[ŞEKİLLER LİSTESİ VII](#_Toc176297703)

[TABLOLAR LİSTESİ IX](#_Toc176297704)

[KISALTMALAR LİSTESİ X](#_Toc176297705)

[ÖZET XI](#_Toc176297706)

[ABSTRACT XII](#_Toc176297707)

[1. GİRİŞ 1](#_Toc176297708)

[2. EEG BEYİN DALGALARI 2](#_Toc176297709)

[2.1. Beyin Dalgaları ve Frekans Bantları 2](#_Toc176297710)

[2.2. EEG Kullanım Alanları 3](#_Toc176297711)

[2.2.1. Psikolojik Araştırmalarda EEG 3](#_Toc176297712)

[2.2.2. Beyin Bilgisayar Arayüzleri 4](#_Toc176297713)

[2.2.3. Yaşam Kalitesini İyileştirme 4](#_Toc176297714)

[2.2.4. Kullanıcı Deneyimi Araştırmaları 5](#_Toc176297715)

[2.3. Yaş ve Cinsiyet Çıkarımı 6](#_Toc176297716)

[2.4. Hedefler 7](#_Toc176297717)

[2.5. Yöntemler 7](#_Toc176297718)

[2.5.1. Destek Vektör Makineleri (SVM) 7](#_Toc176297719)

[2.5.2. Yapay Sinir Ağları (ANN) 8](#_Toc176297720)

[2.5.3. Convolutional Neural Networks (CNN) 9](#_Toc176297721)

[2.5.4. Recurrent Neural Networks (RNN) 9](#_Toc176297722)

[2.5.5. Long Short-Term Memory Networks (LSTM) 10](#_Toc176297723)

[2.5.6. Principal Component Analysis (PCA) 10](#_Toc176297724)

[2.5.7. Support Vector Regression (SVR) 11](#_Toc176297725)

[2.6. Avantaj ve Dezavantajlar 12](#_Toc176297726)

[2.6.1. BLSTM-LSTM Modeli 12](#_Toc176297727)

[2.6.2. CNN Mimarileri 12](#_Toc176297728)

[2.6.3. Derin CNN Modeli 12](#_Toc176297729)

[2.6.4. RNN ile Yaş Tahmini 13](#_Toc176297730)

[2.6.5. EEGNet ile Yaş Tahmini 13](#_Toc176297731)

[3. LİTERATÜR TARAMASI 14](#_Toc176297732)

[3.1. Yaş Tahmini 14](#_Toc176297733)

[3.2. Cinsiyet Tahmini 14](#_Toc176297734)

[3.3. Yaş ve Cinsiyet Çıkarımının Kombine Edilmesi 15](#_Toc176297735)

[3.4. Yapılmış Çalışmalar 15](#_Toc176297736)

[3.4.1. Kaushik’in BLSTM-LSTM Modeli ile Yaş ve Cinsiyet Tahmini 15](#_Toc176297737)

[3.4.2. Truong’un CNN Mimarileri ile EEG Verilerinden Cinsiyet Sınıflandırması 16](#_Toc176297738)

[3.4.3. Derin CNN ile Yaş Tahmini 16](#_Toc176297739)

[3.4.4. RNN ile Yaş Tahmini 17](#_Toc176297740)

[3.4.5. EEGNet ile Yaş Tahmini 17](#_Toc176297741)

[4. PROJE 19](#_Toc176297742)

[4.1. Veri Seti Tanımı 19](#_Toc176297743)

[4.2. Verilerin Toplanması 19](#_Toc176297744)

[4.3. Veri Setinin İçeriği 19](#_Toc176297745)

[4.3.1. EEG Elektrot Sütunları 19](#_Toc176297746)

[4.3.2. Demografik Bilgi Sütunları 20](#_Toc176297747)

[4.3.3. Deneklerin Yaş ve Cinsiyet Verileri 21](#_Toc176297748)

[4.4. Grafiksel Analiz Ve Dağılımlar 22](#_Toc176297749)

[4..1. EEG Verilerinin İşlenmesi ve Analizinin Özeti 27](#_Toc176297750)

[4.4.2. EDF Dosyalarının CSV Formatına Dönüştürülmesi: 27](#_Toc176297751)

[4.4.3. Veri Setinin Optimize Edilmesi: 28](#_Toc176297752)

[4.4.4. Denek Bilgileriyle Eşleştirme: 29](#_Toc176297753)

[4.4.5. Tek Bir CSV Dosyası Haline Getirme: 30](#_Toc176297754)

[4.4.6. Cinsiyet Verilerinin Sayısal Hale Getirilmesi: 31](#_Toc176297755)

[4.5. Gerekli Kütüphaneler 31](#_Toc176297756)

[4.5.1. NumPy 31](#_Toc176297757)

[4.5.2. Pandas 32](#_Toc176297758)

[4.5.3. Scikit-Learn 33](#_Toc176297759)

[4.5.4. TensorFlow ve Keras 33](#_Toc176297760)

[4.5.5. Joblib 34](#_Toc176297761)

[4.6. Gerçekleştirme Aşamaları 35](#_Toc176297762)

[4.6.1. Veri Setini Yükleme 35](#_Toc176297763)

[4.6.2. Özellik ve Hedef Değişkenlerin Ayrılması 36](#_Toc176297764)

[4.6.3. Verilerin Standardizasyonu 37](#_Toc176297765)

[4.6.4. Verilerin Yeniden Şekillendirilmesi 37](#_Toc176297766)

[4.6.5. Modelin Tanımlanması 37](#_Toc176297767)

[4.6.6. Sistemin Çalışma Prensibi 39](#_Toc176297768)

[4.6.7. Katmanlar Arasındaki İlişkiler 39](#_Toc176297769)

[4.6.8. Son Katman (Dense Katmanı) 39](#_Toc176297770)

[4.6.9. Eğitim ve Öğrenme Süreci 39](#_Toc176297771)

[4.6.10. Yaş Tahmini Modeli 40](#_Toc176297772)

[4.6.11. Cinsiyet Tahmini Modelinin Oluşturulması 41](#_Toc176297773)

[4.7. Modele Kullanıcı Girişiyle Veri Verilerek Test Edilmesi 41](#_Toc176297774)

[5. SONUÇLAR 47](#_Toc176297775)

[KAYNAKÇA: 58](#_Toc176297776)

[ÖZGEÇMİŞ 60](#_Toc176297777)

# ŞEKİLLER LİSTESİ

**Şekil 2.1.** Beyin Dalgaları 3

**Şekil 2.2.** Beyim Dalgaları 2 3

**Şekil 2.3.** EEG Bağlantısı 4

**Şekil 2.4**. EEG Tasviri 5

**Şekil 2.5.** EEG Okunması 6

**Şekil 2.6.** SVM. 8

**Şekil 2.7.** ANN 8

**Şekil 2.8.** CNN 9

**Şekil 2.9.** RNN 10

**Şekil 2.10**. LTSM 10

**Şekil 2.11.** PCA 11

**Şekil 2.12.** SVR. 11

**Şekil 3.1.** EEGNet Mimarisi 18

**Şekil 4.1.**Veris setinin ilk 5 satırı 21

**Şekil 4.2.** Veri seti cinsiyet dağılımı 22

**Şekil 4.3**. Veri seti yaş dağılımı 23

**Şekil 4.4.** Veri seti correlation matrix 24

**Şekil 4.5.** Veri seti sütun dağılımları 1 25

**Şekil 4.6.** Veri seti sütun dağılımları 2 25

**Şekil 4.7.** Veri seti sütun dağılımları 3 26

**Şekil 4.8.** Veri seti sütun dağılımları 4 26

**Şekil 4.9.** Numpy 32

**Şekil 4.10.** Pandas 32

**Şekil 4.11.** Scikit-Learn 33

**Şekil 4.12**. TensorFlow 34

**Şekil 4.13.** Keras 34

**Şekil 4.14.** Joblib 35

**Şekil 4.15.** Örnek 1 45

**Şekil 4.16.** Örnek 2 46

**Şekil 4.17.** Örnek 3 46

**Şekil 4.18.** Örnek 4 46

**Şekil 5.1.** Formüller 48

**Şekil 5.2.** Accuracy vs Threshold 48

**Şekil 5.3.** Precision vs Threshold 49

**Şekil 5.4.** Recall vs Threshold 50

**Şekil 5.5.** F1-Score vs Threshold 51

**Şekil 5.6.** Recall vs Threshold 52

**Şekil 5.7.** Precision, Recall, and F1-Score vs Threshold 52

**Şekil 5.8.** Precision, Recall, and F1-Score by Age Groups 54

**Şekil 5.9.** Yaş Modeli Eğitim ve Doğrulama Kaybı 55

**Şekil 5.10.** Yaş Modeli Eğitim ve Doğrulama RMSE 55

**Şekil 5.11.** Gerçek Yaş ve Tahmin Edilen Yaş 56

**Şekil 5.12.** Cinsiyet Modeli Eğitim ve Doğrulama Kaybı 57

**Şekil 5.13.** Cinsiyet Modeli Eğitim ve Doğrulama Doğruluğu 57

**Şekil 5.14.** Cinsiyet Modeli Karışıklık Matrisi 58

**Şekil 5.15.** Yaş Modeli Karışıklık Matrisi 58

# TABLOLAR LİSTESİ

**Tablo 1.** Veri setindeki deneklerin yaşları ve cinsiyetleri tablosu 21

**Tablo 2.** Test Girdi Verisi Örneği 41

**Tablo 3.** Sınıflandırma Metrikleri 47

# 

# KISALTMALAR LİSTESİ

**EEG:** Electroencephalography (Elektroensefalografi)

**BCI:** Brain-Computer Interface (Beyin-Bilgisayar Arayüzü)

**CNN:** Convolutional Neural Network (Konvolüsyonel Sinir Ağı)

**LSTM:** Long Short-Term Memory (Uzun Kısa Dönem Hafıza)

**SVM:** Support Vector Machine (Destek Vektör Makinesi)

**ANN:** Artificial Neural Network (Yapay Sinir Ağı)

**RNN:** Recurrent NeuralNetwork (Tekrarlayan Sinir Ağı)

**PCA:** Principal Component Analysis (Temel Bileşen Analizi)

**SVR:** Support Vector Regression (Destek Vektör Regresyonu)

**DCNN:** Deep Convolutional Neural Network (Derin Konvolüsyonel Sinir Ağı)

**BLSTM:** Bidirectional Long Short-Term Memory (Çift Yönlü Uzun Kısa Süreli Bellek)

**MSE:** Mean Squared Error (Ortalama Hata Karesi)

# ÖZET

**EEG TABANLI VERİLERDEN YAŞ VE CİNSİYET ÇIKARIMI İÇİN DERİN ÖĞRENME TABANLI YENİ BİR YÖNTEM**

EEG (elektroensefalografi) verileri kullanılarak bireylerin yaş ve cinsiyet tahminini gerçekleştirmek amacıyla derin öğrenme tabanlı bir sistem geliştirilmiştir. EEG, beyin aktivitelerinin elektriksel potansiyellerini ölçen ve nörofizyolojik durumlar hakkında bilgi sunan bir yöntemdir. Bu çalışmanın temel amacı, EEG sinyallerinin bireylerin yaş ve cinsiyetine dair ipuçları içerip içermediğini araştırmak ve bu bilgileri kullanarak yaş ve cinsiyet tahmin modeli geliştirmektir.

Çalışma, yaş ve cinsiyet tahminleri için aynı kod yapısı içinde birbirinden bağımsız iki modelin oluşturulmasını içermektedir. İlk aşamada, yaş tahminine yönelik bir model geliştirilmiştir. Geliştirilen derin öğrenme modeli, veri seti üzerinden eğitilerek, bireylerin yaş gruplarını doğru bir şekilde tahmin edebilme kapasitesine ulaşmıştır. Modelin performansı, çeşitli yaş gruplarına yönelik doğru tahmin oranları ile değerlendirilmiş ve elde edilen sonuçlar, yaş tahmininde yüksek doğruluk oranlarına işaret etmiştir. İkinci aşamada, aynı kod yapısı içinde cinsiyet tahmini yapmak üzere ikinci bir model oluşturulmuştur. Bu model de EEG sinyallerini kullanarak bireylerin cinsiyetini tahmin etmeyi amaçlamaktadır. Cinsiyet farklılıklarının EEG sinyallerindeki yansımalarını tespit edebilmek için özellik çıkarımı ve model eğitimi süreçleri optimize edilmiştir. Derin öğrenme modeli, cinsiyete özgü sinyalleri öğrenmek için eğitilmiş ve yüksek doğruluk oranlarıyla cinsiyet tahmini yapabilmiştir. Bu sonuçlar, EEG verilerinin cinsiyet farklılıklarını belirlemede etkili olduğunu göstermektedir.

Çalışmanın sonuçları, EEG verilerinin biyometrik tanımlama süreçlerinde kullanılabilirliğini ortaya koymaktadır. Gelecekte bu çalışmanın daha geniş veri setleri ve daha karmaşık modeller ile geliştirilmesi, EEG tabanlı biyometrik tahmin sistemlerinin daha da ileriye taşınmasına olanak tanıyacaktır.

**Anahtar Kelimeler:** Derin Öğrenme, EEG, LTSM, Yaş-Cinsiyet Tahmini

# ABSTRACT

**A NEW DEEP LEARNING-BASED METHOD FOR AGE AND GENDER INFERENCE FROM EEG DATA**

A deep learning-based system has been developed to predict individuals' age and gender using EEG (electroencephalography) data. EEG is a method that measures the electrical potentials of brain activities and provides insights into neurophysiological states. The primary goal of this study is to investigate whether EEG signals contain clues related to individuals' age and gender, and to develop age and gender prediction models based on this information.

The study involves the creation of two independent models within the same code structure, one for age prediction and the other for gender prediction. In the first stage, a model was developed for age prediction. The deep learning model was trained on a dataset to accurately predict the age groups of individuals. The model's performance was evaluated based on its accuracy in predicting various age groups, and the results indicated high accuracy in age estimation. In the second stage, a separate model was created within the same code structure to predict gender. This model aimed to predict individuals' gender using EEG signals. The feature extraction and model training processes were optimized to detect the reflections of gender differences in EEG signals. The deep learning model was trained to learn gender-specific signals and achieved high accuracy in gender prediction. These results demonstrate that EEG data can effectively identify gender differences.

The study's findings highlight the potential of EEG data in biometric identification processes. In the future, expanding this study with larger datasets and more complex models will enable further advancements in EEG-based biometric prediction systems.

**Keywords:** Deep Learning, EEG, LSTM, Age-Gender Estimation

# 

# 1. GİRİŞ

EEG, yani elektroensefalografi, beynin elektriksel aktivitelerini ölçen bir tekniktir. Beyin, nöronlar adı verilen milyarlarca hücreden oluşur ve bu hücreler arası iletişim, elektriksel sinyaller aracılığıyla gerçekleşir. EEG, bu sinyalleri kaydederek beynin farklı bölgelerindeki elektriksel aktiviteleri inceler.

EEG kaydı, kafa derisine yerleştirilen bir dizi küçük metal elektrot aracılığıyla gerçekleştirilir. Bu elektrotlar, beyindeki elektriksel aktiviteleri toplar ve bu sinyaller bir cihaz tarafından kaydedilir. Ortaya çıkan kayıt, zaman içinde dalgalanan bir dizi elektriksel sinyal veya "beyin dalgası" olarak gösterilir. Bu beyin dalgaları, genellikle saniyede meydana gelen elektriksel dalgalanma sayısına (frekans) ve bu dalgaların kuvvetine göre sınıflandırılır. Farklı beyin dalgaları, kişinin bilinç durumu, uyanıklık seviyesi, zihinsel aktivite veya duygusal durumu hakkında bilgi verebilir.

EEG, beyin aktivitelerini incelemek için kullanılan bir yöntemdir; yani, vücuda herhangi bir cerrahi müdahale yapılmadan uygulanır. EEG, beynin genel elektriksel durumunu değerlendirmek için çok hızlı ve etkili bir yol sunar, bu da onu birçok alanda önemli bir araç haline getirir. Beynin belirli bir uyarana nasıl tepki verdiğini veya belirli bir durum sırasında nasıl çalıştığını anlamak için kullanılır.

# 2. EEG BEYİN DALGALARI

EEG, birçok farklı alanda kullanılır ve her bir kullanım alanı, beynin işleyişi hakkında benzersiz bilgiler sunar. Klinik alanda EEG, en çok epilepsi gibi nörolojik rahatsızlıkların tanısında kullanılır. Epilepsi hastalarında, beynin belirli bölgelerinde anormal elektriksel aktiviteler meydana gelir ve bu aktiviteler EEG ile tespit edilebilir. EEG, epileptik nöbetlerin yerini, sıklığını ve şiddetini belirlemek için vazgeçilmez bir araçtır. Ayrıca, EEG, beyin hasarı, uyku bozuklukları ve hatta komadaki hastaların bilinç durumunun değerlendirilmesinde de kullanılır. Bu durumlar, beynin elektriksel aktivitesinde belirgin değişiklikler yaratır ve bu değişiklikler EEG ile izlenebilir.

EEG, kafatası üzerine yerleştirilen elektrotlar aracılığıyla beyinden yayılan elektrik sinyallerini algılar. Bu sinyaller, nöronların toplu aktivitesinden kaynaklanır ve mikrovolt (µV) düzeyinde ölçülür. EEG verileri, özellikle zaman serisi analizi ve sinyal işleme teknikleri kullanılarak analiz edilir, bu da beynin farklı fonksiyonel durumları hakkında bilgi sağlar.

## 2.1. Beyin Dalgaları ve Frekans Bantları

Beyin dalgaları, frekanslarına göre beş ana kategoriye ayrılır: delta, theta, alfa, beta ve gamma dalgaları. Bu dalgalar, farklı zihinsel durumlar ve beyin aktiviteleri ile ilişkilidir.

**Delta Dalgaları (0.5 - 4 Hz)**

Delta dalgaları, en düşük frekansa sahip beyin dalgalarıdır ve genellikle derin uyku veya bilinçsizlik durumunda baskındır. Uyku çalışmaları ve bilinç düzeyi değerlendirmelerinde önemli bir rol oynarlar.

**Theta Dalgaları (4 - 8 Hz)**

Theta dalgaları, hafif uyku, meditasyon veya derin düşünce durumlarında ortaya çıkar. Ayrıca, yaratıcı düşünme ve öğrenme süreçlerinde de etkili olabilirler.

**Alfa Dalgaları (8 - 12 Hz)**

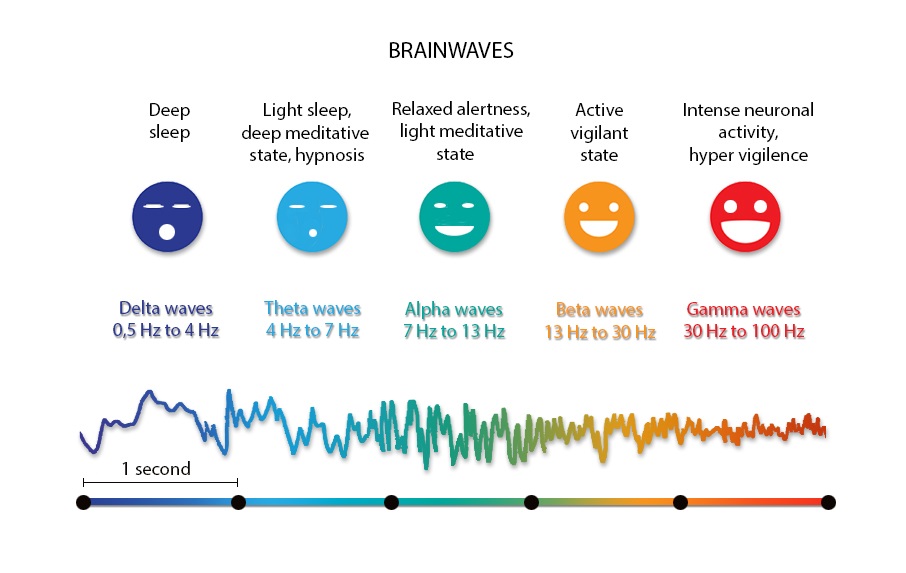
Alfa dalgaları, dinlenme, gevşeme ve sakin zihin durumlarıyla ilişkilidir. Gözler kapalıyken, özellikle uyanıklık ve uyku arasındaki geçiş sırasında belirgindir. Alfa aktivitesi, bilişsel durumun değerlendirilmesinde yaygın olarak kullanılır.

**Beta Dalgaları (12 - 30 Hz)**

Beta dalgaları, aktif düşünme, problem çözme, karar verme ve dikkat gerektiren durumlarda baskındır. Yüksek mental aktivite, kaygı veya stres durumlarında artış gösterebilirler.

**Gamma Dalgaları (30 - 100 Hz)**

Gamma dalgaları, yüksek frekanslıdır ve bilinçli algı, dikkat ve bilinç durumlarıyla ilişkilidir. Zihinsel işlemler ve bilişsel görevlerde yoğunlaşma sırasında ortaya çıkar.

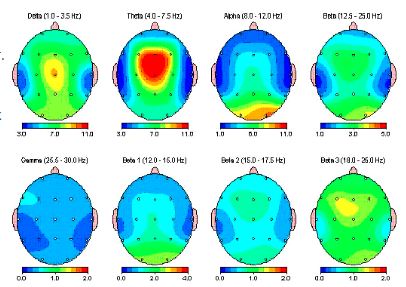


**Şekil 2.1.** Beyin Dalgaları

## 2.2. EEG Kullanım Alanları

### 2.2.1. Psikolojik Araştırmalarda EEG

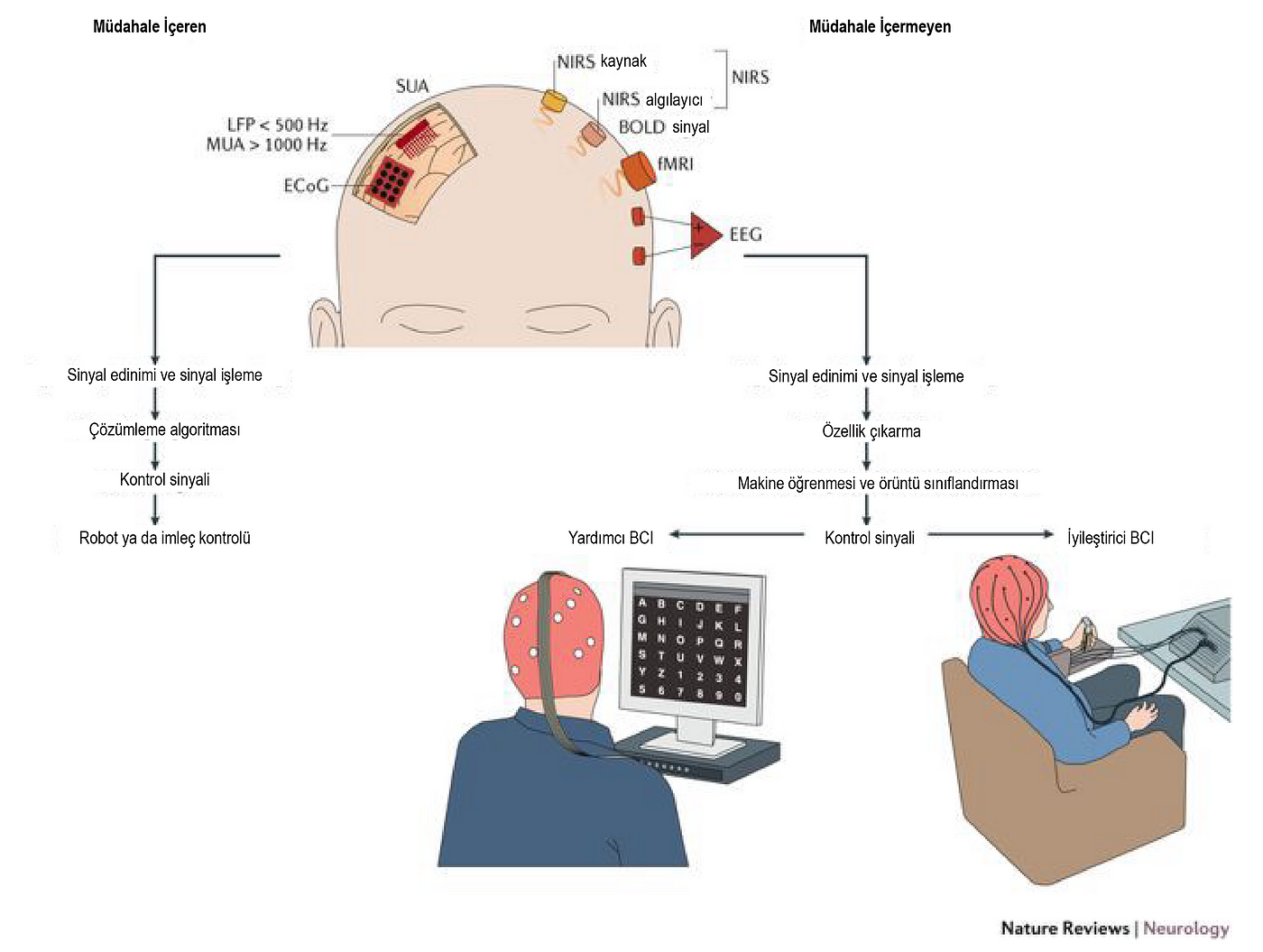
EEG, psikolojik araştırmalarda da geniş bir uygulama yelpazesi sunar. Örneğin, dikkat, hafıza, algı ve karar verme gibi bilişsel süreçler, EEG ile incelenebilir. EEG, bu süreçlerin zamanla nasıl geliştiğini ve beynin hangi bölgelerinin bu süreçlere katıldığını ortaya koyar. Özellikle, bilişsel nörobilim araştırmalarında, EEG’nin milisaniye düzeyindeki zaman çözünürlüğü, beynin anlık tepkilerini incelemek için ideal bir araçtır. Bu, araştırmacıların, örneğin bir görsel uyarana beynin nasıl ve ne kadar hızlı tepki verdiğini anlamalarını sağlar. Aynı şekilde, EEG, duygusal tepkilerin, stresin, anksiyetenin ve diğer psikolojik durumların beyin üzerindeki etkilerini anlamak için de kullanılır.



**Şekil 2.2.** Beyin Dalgaları 2

### 2.2.2. Beyin Bilgisayar Ara yüzleri

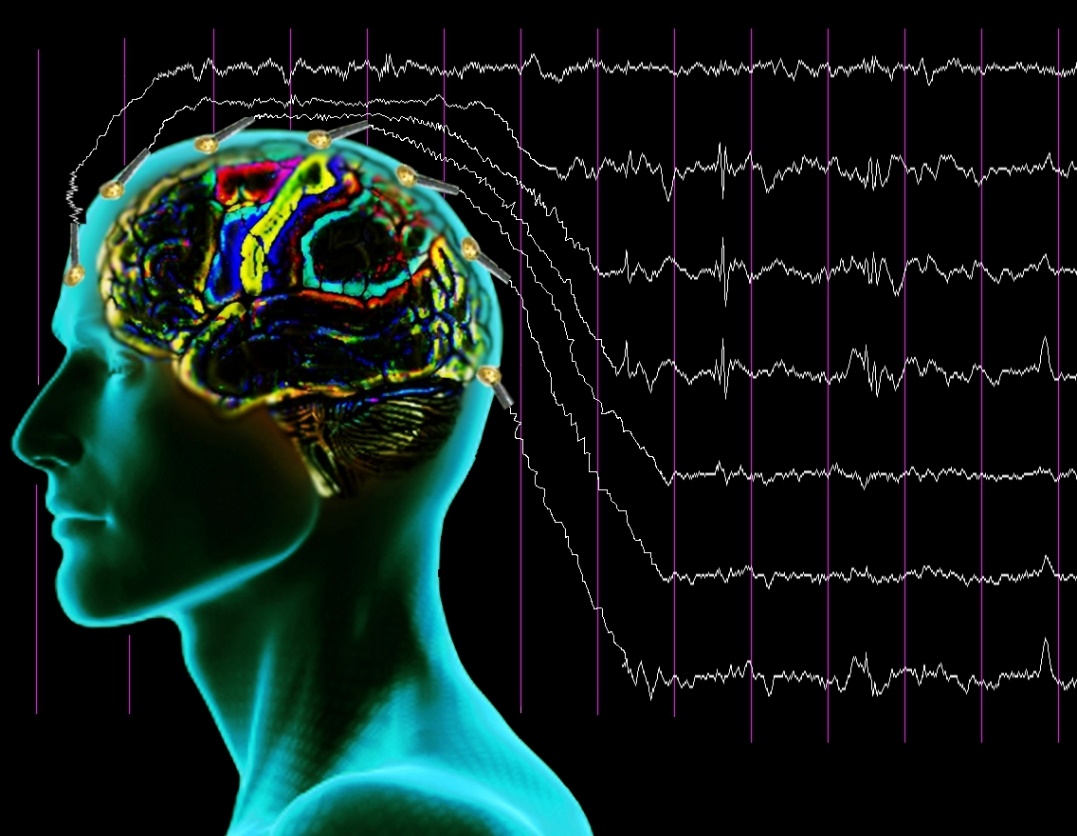
EEG’nin bir diğer önemli kullanım alanı ise beyin-bilgisayar ara yüzleridir (BCI). BCI, beynin elektriksel aktivitelerini okuyarak, bu sinyalleri bir bilgisayar veya başka bir cihazı kontrol etmek için kullanır. Bu teknoloji, felçli hastalar veya motor fonksiyon kaybı yaşayan bireyler için devrim niteliğindedir. EEG tabanlı BCI sistemleri, bireylerin beyin dalgaları aracılığıyla bilgisayarları, protezleri veya diğer cihazları kontrol etmelerini sağlar. Bu, bireylere kaybettikleri hareket yetilerini geri kazandırmakla kalmaz, aynı zamanda günlük yaşamlarında bağımsızlıklarını artırır.



**Şekil 2.3.** EEG Bağlantısı

### 2.2.3. Yaşam Kalitesini İyileştirme

EEG tabanlı cihazlar aynı zamanda meditasyon, stres yönetimi ve uyku kalitesini artırmak gibi amaçlarla kullanılmaktadır. Bu cihazlar, kullanıcıların beyin aktivitelerini izlemelerine ve bu aktivitelerle ilgili geri bildirim almalarına olanak tanır. Örneğin, bir kullanıcı meditasyon yaparken EEG sinyalleri ile zihinsel durumunu izleyebilir ve daha derin bir rahatlama durumu elde edebilir. Bu tür teknolojiler, zihinsel sağlığı ve genel yaşam kalitesini iyileştirmek için EEG’yi günlük yaşama entegre eder.



**Şekil 2.4.** EEG Tasviri

### 2.2.4. Kullanıcı Deneyimi Araştırmaları

EEG’nin bir başka dikkat çekici kullanım alanı ise nöropazarlama ve kullanıcı deneyimi araştırmalarıdır. EEG, bir kişinin bir ürüne, reklama veya bir web sitesine nasıl tepki verdiğini incelemek için kullanılabilir. Beynin elektriksel aktivitelerini izleyerek, hangi reklamların daha dikkat çekici olduğunu veya hangi kullanıcı ara yüzlerinin daha etkili olduğunu belirlemek mümkündür. Bu, pazarlama stratejilerinin daha hedefli ve etkili olmasını sağlar.



**Şekil 2.5.** EEG Okunması

## 2.3. Yaş ve Cinsiyet Çıkarımı

Yaş ve cinsiyet, bireylerin beyin yapısı ve fonksiyonları üzerinde belirgin etkiler yaratır. Yaşlandıkça beynin belirli bölgelerinde elektriksel aktivitede değişiklikler gözlemlenebilir, kadınlar ve erkekler arasındaki hormonal farklılıklar beyin aktivitelerinde çeşitliliklere yol açabilir. EEG sinyalleri, bu tür demografik farkları yansıtabilir ve bu da yaş ve cinsiyetin, EEG verilerinden tahmin edilebilmesine olanak sağlayabilir. Böyle bir tahmin, sadece bireyin kimliğini tanımada değil, aynı zamanda yaşa veya cinsiyete özgü rahatsızlıkların erken teşhisinde de kritik rol oynayabilir. Bu bağlamda, EEG verilerinden yaş ve cinsiyet tahmini yapmak, klinik nörolojide devrim niteliğinde bir adım olabilir. Yaşa bağlı hastalıkların erken teşhisi, bireylerin yaşam kalitesini artırmak için önemlidir. EEG verilerini kullanarak bu tür hastalıkların erken belirtilerini belirlemek, önleyici tedbirlerin alınmasında hayati rol oynayabilir. Aynı şekilde, cinsiyete özgü beyin aktivitelerinin anlaşılması, depresyon veya anksiyete gibi zihinsel rahatsızlıkların cinsiyete göre farklı tedavi yaklaşımlarının geliştirilmesine olanak tanır. Bunun yanı sıra, EEG verilerinden yaş ve cinsiyet tahmini yapma süreci, veri bilimi ve yapay zeka alanlarında da önemli katkılar sunar. Derin öğrenme algoritmaları, büyük ve karmaşık veri setlerinden anlamlı bilgileri çıkarmada eşsiz bir yeteneğe sahiptir. EEG verilerinin analizi, bu algoritmaların sınırlarını zorlayarak, hem teorik hem de pratik zorluklarıyla büyüleyici bir çalışma alanı sunar. Bir derin öğrenme modeli, EEG sinyallerinden yaşı ve cinsiyeti yüksek doğrulukla tahmin edebiliyorsa, bu teknoloji, diğer biyomedikal sinyallerin analizi için de genişletilebilir.

Bu tür bir araştırma, toplumsal etkileriyle de dikkat çeker. Yaş ve cinsiyet tahmini, bireysel sağlık izleme cihazları, kişisel sağlık yönetimi ve özelleştirilmiş tıbbi hizmetler gibi alanlarda devrim yaratabilir. Özellikle yaşlanan nüfus ve cinsiyet temelli sağlık farklılıkları göz önüne alındığında, EEG verilerinden elde edilecek demografik bilgilerin doğruluğu, sağlık sistemlerinde verimliliği ve kişiselleştirilmiş sağlık hizmetlerini artırabilir.

## 2.4. Hedefler

EEG verilerinden yaş ve cinsiyeti yüksek doğrulukla tahmin edebilen bir derin öğrenme modeli geliştirmektir. Beynin farklı bölgelerindeki elektriksel aktivite, bireyin yaşına ve cinsiyetine bağlı olarak değişebilir. Geleneksel makine öğrenmesi yaklaşımları, EEG verilerinin yüksek boyutluluğu ve karmaşıklığı karşısında sınırlı kalabilir. Derin öğrenme modelleri ise, büyük veri setlerinden karmaşık kalıpları çıkarma kapasitesine sahiptir ve bu nedenle EEG verilerinden anlamlı özelliklerin otomatik olarak çıkarılmasında daha etkilidir.

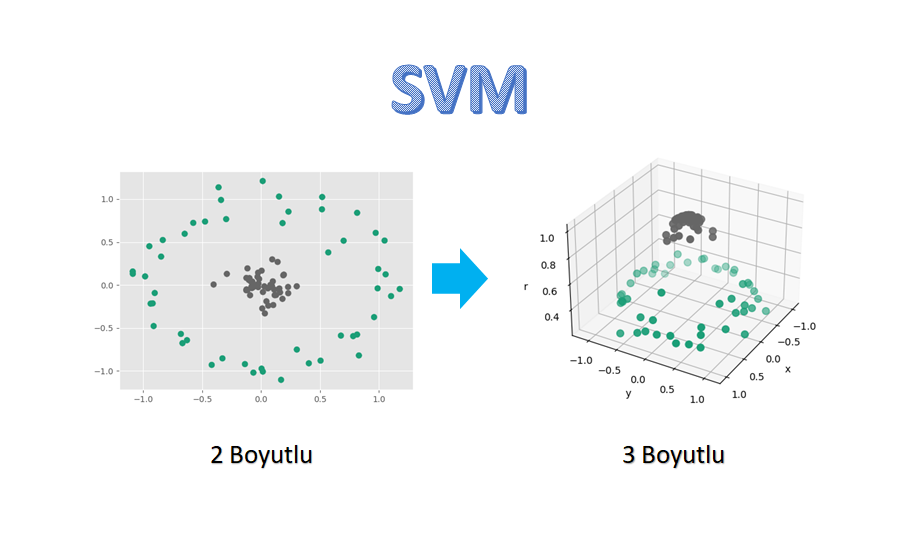
EEG verilerinden yaş ve cinsiyet tahmini, bireyselleştirilmiş tıp, nöroloji, psikiyatri ve nöromarketing gibi alanlarda devrim niteliğinde uygulamalara sahip olabilir. Örneğin, yaşa veya cinsiyete özgü nörolojik hastalıkların erken teşhisi, EEG sinyallerinden çıkarılan bilgilerle daha hassas hale gelebilir. Ayrıca, EEG verilerine dayalı yaş ve cinsiyet tahmini, nörobilim araştırmalarında bireyler arası varyasyonları anlamak ve bu varyasyonların bilişsel süreçlerle nasıl ilişkili olduğunu keşfetmek için yeni bir araç sağlayabilir.

Modelin doğruluğu, hassasiyeti ve genelleme yeteneği gibi metriklerle değerlendirilmesi, modelin pratik uygulamalardaki geçerliliğini belirleyecektir. Bu karşılaştırmalar, derin öğrenme modelinin EEG verilerinden yaş ve cinsiyet tahmini yapmada ne kadar etkili olduğunu gösterecek ve bu alanda gelecekteki araştırmalar için bir temel oluşturacaktır.

## 2.5. Yöntemler

### 2.5.1. Destek Vektör Makineleri (SVM)

Veri noktalarını yüksek boyutlu bir uzaya haritalayarak, iki sınıf arasında en iyi ayırımı sağlayacak bir sınır oluşturur. İki boyutlu veriler üç boyutlu verilere dönüştürülür. EEG verilerinde yaş ve cinsiyet tahmini için özel özellikler kullanılarak SVM modelleri oluşturulabilir. Küçük veri setleri üzerinde daha iyi çalışır. Büyük veri setlerinde veya çok sayıda özellik içeren durumlarda eğitim süresi uzayabilir [1].

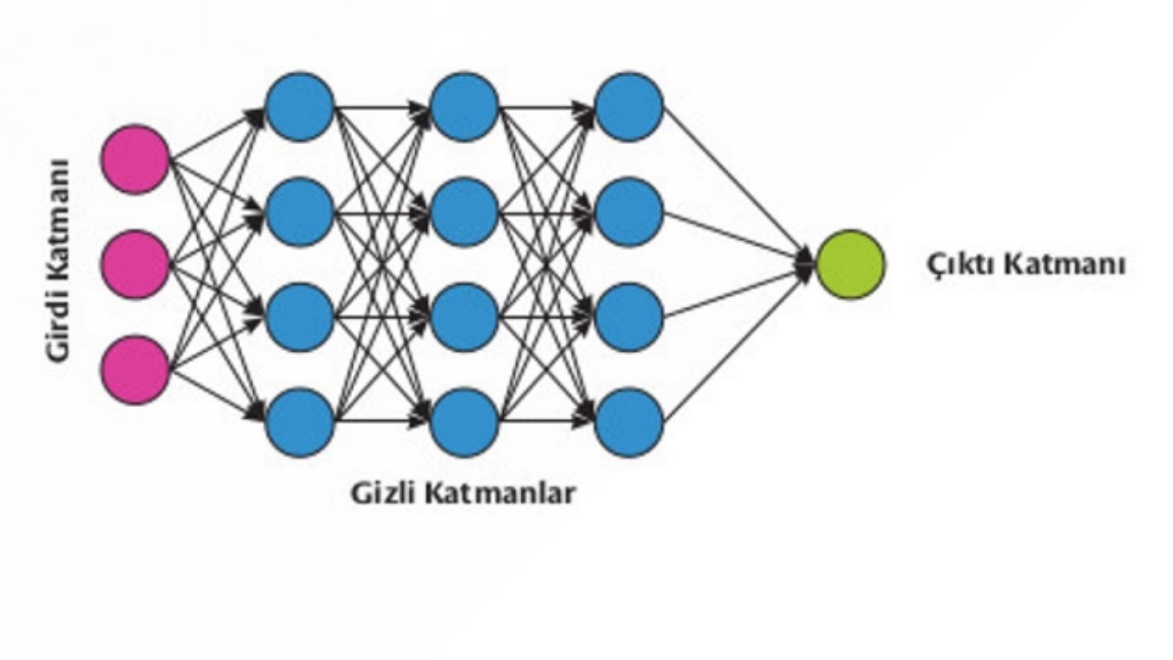


**Şekil 2.6.** SVM

### 2.5.2. Yapay Sinir Ağları (ANN)

Biyolojik sinir ağlarını taklit eden bir yapıdadır. Giriş katmanı, bir veya daha fazla gizli katman ve çıkış katmanından oluşur. EEG sinyalleri doğrudan veya işlenmiş özellikler olarak giriş katmanına verilir. Karmaşık ilişkileri öğrenebilir ve yüksek doğruluk oranlarına ulaşabilir.

Büyük veri setleri ve optimize edilmesi gereken hiperparametreler gerektirir [2].

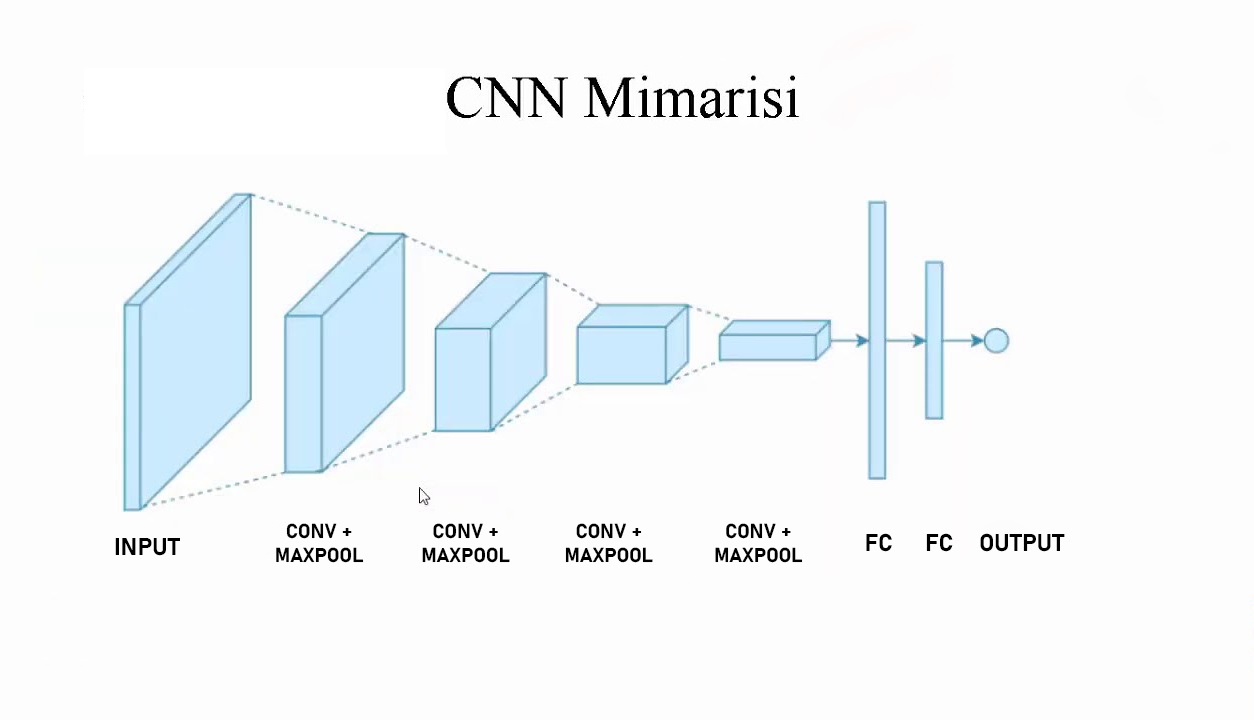


**Şekil 2.7.** ANN

### 2.5.3. Convolutional Neural Networks (CNN)

Görüntü ve zaman serisi verileri (EEG gibi) için uygundur. Veriler üzerinde yerel özelliklerin çıkarılmasına odaklanan konvolüsyon katmanları ve özelliklerin genelleştirilmesi için havuzlama katmanları içerir. Otomatik özellik çıkarımı yapabilir ve karmaşık yapıları tanıyabilir.

Daha büyük veri setlerine ihtiyaç duyar ve eğitim süresi uzundur [3].

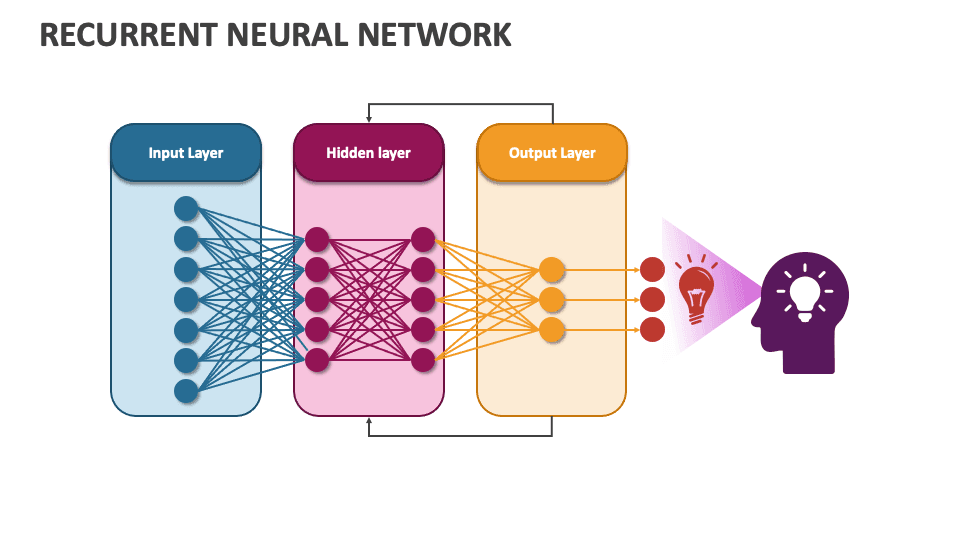


**Şekil 2.8.** CNN

### 2.5.4. Recurrent Neural Networks (RNN)

Zaman serisi verileri ile çalışmak için tasarlanmıştır ve geçmiş zaman adımlarını hatırlayabilir. EEG sinyalleri gibi sıralı verilerdeki bağımlılıkları öğrenebilir. Zaman serisi verilerindeki uzun vadeli bağımlılıkları öğrenebilir.

Uzun sekanslar üzerinde öğrenme zorluğu olabilir, ancak bu sorun Long Short-Term Memory (LSTM) hücreleri gibi özel RNN türleriyle aşılabilir [4].



**Şekil 2.9.** RNN

### 2.5.5. Long Short-Term Memory Networks (LSTM)

Uzun vadeli bağımlılıkları öğrenmekte başarılıdır. EEG verilerindeki zaman bağımlılıklarını öğrenmek için kullanılabilir. Uzun sekanslar üzerinde daha etkili öğrenme sağlar.

Daha karmaşık bir yapı ve daha uzun eğitim süresi gerektirir [5].

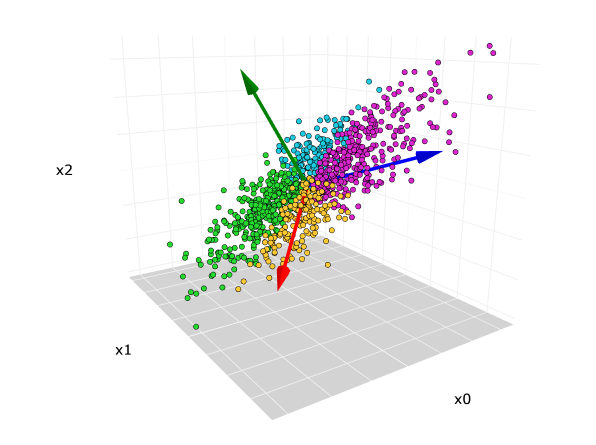


**Şekil 2.10.** LTSM

### 2.5.6. Principal Component Analysis (PCA)

Çok boyutlu verilerdeki temel bileşenleri çıkararak boyut azaltımı yapar. EEG verilerindeki önemli özellikleri çıkarmak için kullanılabilir ve daha sonra bu özellikler diğer makine öğrenmesi modelleri ile işlenebilir. Verilerin boyutunu azaltarak modellerin daha verimli çalışmasını sağlar.

Bilgi kaybına neden olabilir; bu yüzden dikkatli kullanılmalıdır [6].

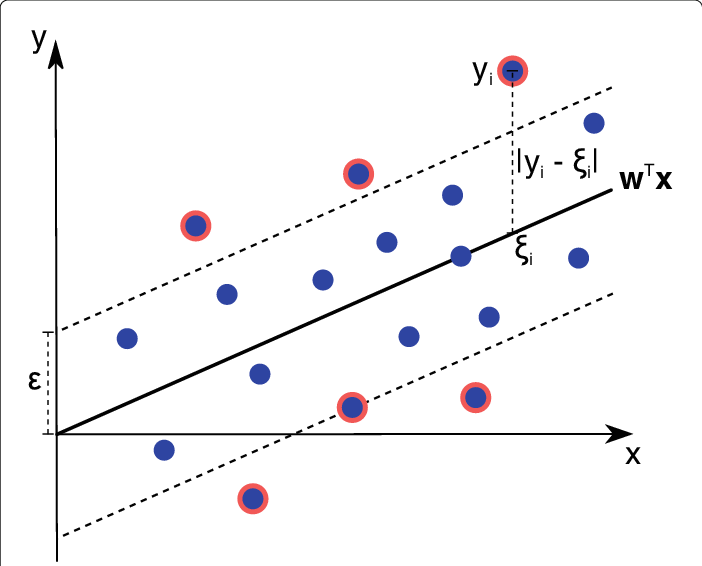


**Şekil 2.11.** PCA

### 2.5.7. Support Vector Regression (SVR)

Yaş tahmini gibi sürekli değişkenlerin tahmini için kullanılabilir. Yüksek performans sağlar ve genellikle doğruluk oranı yüksektir.

Büyük veri setlerinde ve karmaşık yapılarda eğitim süresi uzun olabilir [7].



**Şekil 2.12.** SVR

## 2.6. Avantaj ve Dezavantajlar

### 2.6.1. BLSTM-LSTM Modeli

BLSTM ve LSTM, uzun bağımlılıkları yakalayarak zaman serisi verilerinde başarılıdır. Verileri hem geçmişten hem de gelecekten analiz edebildiği için zaman serisi verilerinde daha derin bilgi çıkarımı sağlar. Veriler üzerinde yüksek performans gösterir ve doğruluk oranı yüksektir.

LSTM’ler genellikle karmaşık olduğundan, eğitimi ve çıkarımı daha fazla hesaplama gücü ve zaman gerektirmesi ve verinin yapısına bağlı olarak aşırı uyum (overfitting) riski bulundurması dezavantajları arasındadır.

Veri setinin zaman serisi ağırlıklıysa ve hem yaş hem de cinsiyet tahmini yapılmak isteniyorsa bu model uygundur.

### 2.6.2. CNN Mimarileri

CNN’ler, verinin uzaysal özelliklerini etkili bir şekilde çıkarabilir, bu da EEG verilerinde önemli olabilir. Görüntü işleme alanında da çok iyi yerleşmiş bir modeldir, genellikle iyi bir başlangıç noktası sunar. GPU ile oldukça iyi bir şekilde paralel işlenebilir, böylelikle eğitim süresi daha az sürmektedir. CNN ,bu özellikleriyle avantaj sağlamaktadır.

Zaman serisi verilerinde (EEG gibi) doğrudan uygulanırken bazı zorluklar yaşayabilir, çünkü bu tür verilerde zaman içindeki bağımlılıkları yakalamak zor olabilir. Genellikle, diğer modellere kıyasla daha fazla parametreye sahiptir ve bu da modeli daha karmaşık hale getirebilir.

EEG verilerinde uzaysal (spatial) özellikler önemliyse ve bu özellikler üzerinden cinsiyet tahmini yapılmak isteniyorsa, CNN mimarileri diğer yöntemlere kıyasla daha iyi performans sergileyecektir.

### 2.6.3. Derin CNN Modeli

Çok katmanlı yapısı sayesinde verilerden karmaşık özellikler çıkarabilir ve farklı katman yapıları ve filtre boyutları ile kolayca modifiye edilebilir.

Büyük veri setleri gerektirir, aksi takdirde model aşırı öğrenmeye eğilimli olabilir. Modelin derinliği arttıkça eğitim süresi de uzar, bu da büyük veri setlerinde dezavantaj olarak karşımıza çıkar.

Büyük veri seti varsa ve bu verilerden yaş tahmini yapılmak isteniyorsa kullanılabilir.

### 2.6.4. RNN ile Yaş Tahmini

Zaman serisi verilerinde başarılıdır, çünkü her adımda önceki bilgilere dayalı tahminler yapar. LSTM veya GRU gibi daha gelişmiş varyantlarına göre daha az hesaplama gücü gerektirir, daha az karmaşık yapıdadır.

Uzun sekanslarda vanishing gradient sorunuyla karşılaşabilir. LSTM ve GRU’ya göre uzun bağımlılıkları yakalama konusunda daha az etkilidir.

Veri setinin daha kısa sekanslardan oluştuğu ve basit bir modelle tahmin yapılacağı durumlarda diğerlerine göre daha uygun olabilir.

### 2.6.5. EEGNet ile Yaş Tahmini

EEGNet, özellikle EEG verileri için tasarlanmış bir CNN mimarisi olduğundan, EEG verilerinin uzaysal ve zamansal özelliklerini iyi bir şekilde yakalayabilir. Diğer derin öğrenme modellerine kıyasla daha hafif bir yapıdadır, bu da daha hızlı eğitim olanağı sağlar.

Çok özelleşmiş olduğundan, başka tür veri setlerine uygulanması zor olabilir. Daha derin ağlara kıyasla sınırlı katman sayısı nedeniyle daha karmaşık özellikleri yakalama kapasitesi sınırlı olabilir. Kullanım alanı diğerlerine oranla geniş değildir.

Sınırlı hesaplama kaynakları varsa EEGNet kullanılabilir.

# 3. LİTERATÜR TARAMASI

## 3.1. Yaş Tahmini

Yaş çıkarımı, EEG sinyalleri kullanılarak yapılan araştırmalarda özellikle dikkat çeken bir konudur. Beynin yaşlanma süreci boyunca geçirdiği yapısal ve işlevsel değişiklikler, EEG sinyallerinde belirgin izler bırakır. Genç ve yaşlı bireylerin beyin aktiviteleri, genellikle farklı frekans bantlarında farklılık gösterir. Örneğin, alfa dalgalarının (8-13 Hz) gücünde yaşla birlikte belirgin bir azalma olduğu gözlemlenmiştir.

Önceki araştırmalara bakıldığında yaş tahmininde EEG sinyallerinin kullanımı konusunda çeşitli yöntemler önerilmiştir. Geleneksel makine öğrenmesi algoritmalarından derin öğrenme modellerine kadar birçok yaklaşım test edilmiştir. Örneğin, destek vektör makineleri (SVM) ve karar ağaçları gibi klasik yöntemler, EEG sinyallerinin özelliklerini kullanarak yaş tahmininde başarılı sonuçlar elde etmiştir [8]. Bununla birlikte, son zamanlarda derin öğrenme modelleri, özellikle sinir ağları, daha yüksek doğruluk oranları sunarak bu alanda ön plana çıkmaktadır.

Derin öğrenme modelleri, EEG sinyallerinin karmaşık yapısını ve zaman içindeki dinamiklerini yakalamada daha başarılı olabilmektedir. Bu modeller, EEG sinyallerinden çıkarılan çok boyutlu özellikleri öğrenerek, yaş tahminini otomatikleştirir ve doğruluğu artırır. Özellikle konvolüsyonel sinir ağları (CNN), EEG sinyallerinin mekânsal ve zamansal özelliklerini aynı anda işleyebilme kapasitesine sahiptir. Bu sayede, farklı yaş gruplarının beyin aktivitelerindeki ince farklılıklar bile tespit edilebilir duruma gelmiştir [9].

## 3.2. Cinsiyet Tahmini

Cinsiyet çıkarımı, EEG sinyalleri kullanılarak yapılan diğer önemli bir çalışmadır. Beyin yapısı ve fonksiyonlarındaki cinsiyete bağlı farklılıklar, EEG sinyallerinde de kendini göstermektedir. Erkekler ve kadınlar arasındaki nörofizyolojik farklılıklar, belirli frekans bantlarında ve beyin bölgelerinde belirgin olabilir. Bu farklılıklar, cinsiyet tahmininde EEG sinyallerinin kullanılmasını mümkün kılar.

Cinsiyet tahminine yönelik araştırmalarda, EEG sinyallerindeki frekans bileşenleri, güç spektrumları ve bağlantısallık ölçütleri gibi özellikler sıklıkla kullanılmaktadır. Örneğin, bazı çalışmalar, erkeklerin EEG sinyallerinde daha yüksek beta dalgası (13-30 Hz) aktivitesine sahip olduğunu belirtmektedir. Bu tür biyomarkerlar, cinsiyet tahmini modellerinin temelini oluşturur. Ancak, cinsiyet farklarının EEG sinyallerinde incelenmesi, sosyal ve kültürel etkilerle karmaşıklaşabilir. Bu doğrultuda edinilen her bilgi ve her çıkarımı doğru olarak kabul etmek hataya yol açabilir. Bu nedenle, modellerin geliştirilmesinde dikkatli bir özellik mühendisliği gereklidir [10].

Cinsiyet çıkarımında bir diğer kullanılan teknik derin öğrenme teknikleridir. Özellikle, LSTM (Long Short-Term Memory) gibi zaman serisi analizi yapabilen modeller, EEG sinyallerinin zaman içindeki değişimlerini etkili bir şekilde yakalayarak cinsiyet tahmininde kullanılmaktadır. Bu modeller, EEG sinyallerindeki cinsiyetle ilgili ince farkları öğrenerek yüksek doğruluk oranları elde edebilir [11].

## 3.3. Yaş ve Cinsiyet Çıkarımının Kombine Edilmesi

EEG sinyalleri kullanılarak hem yaş hem de cinsiyet tahminini aynı anda yapmak, daha karmaşık ancak potansiyel olarak daha bilgilendirici bir yaklaşımdır. Bazı araştırmacılar, bu iki demografik özelliğin birbiriyle etkileşim içinde olduğunu ve birlikte değerlendirildiğinde daha anlamlı sonuçlar verebileceğini savunmaktadır. Örneğin, yaşlanma sürecinin cinsiyetler arasında farklı şekillerde gerçekleşmesi, EEG sinyallerinde belirgin farklılıklar yaratabilir. Bu tür karşılıklı etkilerin modellenmesi, daha doğru tahminler yapmayı mümkün kılabilir [12].

Birlikte kullanılan çalışmalarda genellikle çoklu görev öğrenme yaklaşımları kullanılır. Bu yöntem, aynı anda birden fazla görev (örneğin, yaş ve cinsiyet tahmini) için bir modelin eğitilmesini sağlar. Böylece, her bir görevdeki performans, diğerinden öğrenilen bilgilerle iyileştirilir. Bu tür modeller, EEG sinyallerinden daha zengin ve kapsamlı bilgi çıkarmayı mümkün kılar [13].

## 3.4. Yapılmış Çalışmalar

### 3.4.1. Kaushik’in BLSTM-LSTM Modeli ile Yaş ve Cinsiyet Tahmini

Kaushik ve ekibinin gerçekleştirdiği çalışmada, EEG sinyalleri kullanılarak yaş ve cinsiyet tahmini amacıyla BLSTM-LSTM modeli kullanılmıştır. Bu model, EEG gibi zaman serisi verilerini işleyebilen ve uzun süreli bağımlılıkları öğrenebilen bir sinir ağı yapısıdır. Çalışmada bu model kullanılmıştır, yüksek bir doğruluk oranı elde edilebilmiştir.

Kaushik’in çalışmasında BLSTM ve LSTM modellerini birlikte kullanıp EEG sinyallerinden hem yaş hem de cinsiyet tahmini yapılmıştır.

Kaushik ve ekibi, modelin hem yaş hem de cinsiyet tahmininde yüksek doğruluk oranları elde ettiğini rapor etmiştir. Çalışma sonuçlarına göre:

Yaş Tahmininde Doğruluk Oranı: %87

Cinsiyet Tahmininde Doğruluk Oranı: %92 olarak kayda geçirilmiştir.

Kaushik’in çalışması, EEG sinyallerini kullanarak yaş ve cinsiyet tahmininde BLSTM-LSTM modelinin etkinliğini ortaya koymuştur. Model, elde edilen yüksek doğruluk oranlarıyla, bu alanda önemli bir ilerleme kaydedilmesini sağlamıştır [14].

### 3.4.2. Truong’un CNN Mimarileri ile EEG Verilerinden Cinsiyet Sınıflandırması

Truong ve ekibi, EEG verilerini kullanarak cinsiyet sınıflandırması yapmak amacıyla Convolutional Neural Network (CNN) mimarilerini kullanan bir çalışma gerçekleştirmiştir. CNN, özellikle görüntü işleme ve zaman serisi verilerinin sınıflandırılmasında yaygın olarak kullanılan bir yapay sinir ağı modelidir. EEG verileri gibi çok boyutlu ve karmaşık sinyalleri işleyebilme kapasitesi, CNN'leri bu tür biyomedikal uygulamalar için ideal hale getirmiştir.

Truong ve ekibinin çalışmasında, EEG verileri modelin girişine beslenmiş ve CNN katmanları üzerinden işlem yapılmıştır. Aşamaları şu şekildedir:

Veri Girişi: EEG sinyalleri, çok boyutlu matrisler olarak modele beslenmiştir. Bu veriler, zaman serisi şeklinde düzenlenmiştir. Her bir kanal zaman içindeki sinyalleri temsil etmektedir.

Konvolüsyon Katmanları: Truong ve ekibinin çalışmasında, bu katmanlar sinyallerin farklı frekans bantlarındaki önemli desenleri yakalamada kritik rol oynamıştır.

Pooling (Havuzlama) Katmanları: Havuzlama katmanları, konvolüsyon katmanlarından gelen EEG tabanlı verilerin çıktılarını özetleyerek veri boyutunu küçültür. Bu işlem, modelin hesaplama yükünü azaltmak amacıyla kullanılmıştır.

Çıkış Katmanı: Son olarak, modelin çıkış katmanı cinsiyet sınıflandırmasını gerçekleştirir. Bu katman, her bir EEG sinyalinin kadın veya erkek olarak sınıflandırılmasını sağlar.

Truong ve ekibinin CNN modeli, EEG verilerinden cinsiyet sınıflandırmasında yüksek doğruluk oranlarına ulaşmıştır. Çalışmanın sonuçlarına göre cinsiyet sınıflandırmasında doğruluk oranı: %94 olmuştur.

Bu çalışma EEG verilerinden cinsiyet sınıflandırması yapmada CNN mimarisinin gücünü göstermektedir. Modelin elde ettiği yüksek doğruluk oranları, bu yaklaşımın gelecekteki biyometrik sınıflandırma uygulamalarında geniş bir potansiyele sahip olduğunu ortaya koymaktadır [15].

### 3.4.3. Derin CNN ile Yaş Tahmini

Bu çalışmada, EEG verilerinden yaş tahmini yapmak için Derin Evrişimli Sinir Ağı (DCNN) kullanılmıştır. EEG sinyallerinin karmaşıklığı ve zaman içindeki değişkenliği, yaş tahmini gibi biyometrik analizler için güçlü ve derin öğrenme modellerine ihtiyaç duyar. Bu bağlamda, DCNN’in çok katmanlı yapısı, sinyallerden anlamlı öznitelikleri çıkararak yaş tahmini yapmada etkili bir araç olarak kullanılmıştır.

Çalışma, TD-BRAIN veri tabanından alınan 1,346 EEG verisi içermektedir. Bu veri seti, farklı yaş gruplarına ait kişilerin EEG kayıtlarını içermekte ve modelin geniş bir yaş aralığında doğru tahminler yapmasına fayda sağlamaktadır. Bu çalışmada veri sızıntısını önlemek ve modelin genelleme yeteneğini artırmak amacıyla çapraz doğrulama yöntemi kullanılmıştır. Bu yöntem, modelin aynı konudan gelen eğitim ve test verilerini karıştırmamasını sağlamak amacıyla kullanılmıştır. Böylece, modelin performansı ve gerçek dünya verilerinde nasıl çalışacağı daha iyi anlaşılabilir.

Çalışmanın doğruluk oranı %85’tir. Bu doğruluk oranı, EEG sinyalleri gibi karmaşık biyomedikal verilerden yaş tahmini yapmak için derin öğrenme yaklaşımlarının ne kadar güçlü olduğunu göstermektedir. DCNN’nin çok katmanlı yapısı, yaşa özgü sinyal özelliklerini etkili bir şekilde yakalayabilmiş ve doğru tahminler yapmıştır.

Bu çalışma, EEG verilerinden yaş tahmini yapmak için Derin Evrişimli Sinir Ağı (DCNN) kullanımının etkinliğini gözler önüne sermektedir. TD-BRAIN veri tabanından alınan geniş EEG verileri üzerinde yapılan bu analizler, DCNN’nin biyometrik tahminlerde yüksek doğruluk oranlarına ulaşabileceğini göstermiştir [16].

### 3.4.4. RNN ile Yaş Tahmini

Bu çalışmada, EEG verilerinden yaş tahmini yapmak için RNN mimarisi tercih edilmiştir. RNN’ler, özellikle ardışık veri noktaları arasındaki zaman bağımlılıklarını yakalayabilme açısından etkili olacağı düşünülmüştür.

Veri seti, geniş bir yaş aralığını kapsayan EEG kayıtlarından oluşmaktadır. Modelin eğitim sürecinde, bu EEG sinyalleri zaman serisi olarak modele beslenmiş ve RNN katmanları tarafından işlenmiştir. RNN katmanları, her bir zaman dilimindeki bilgiyi işleyerek, sinyallerde yaşa özgü kalıpları tespit etmeye çalışmıştır.

Yaş tahminlerinde doğruluk oranı %83 olmuştur.

Bu doğruluk oranı, RNN modelinin zaman serisi verilerindeki bağıntıları etkili bir şekilde öğrendiğini göstermektedir. Model, önceki zaman adımlarındaki bilgileri dikkate alarak, EEG sinyallerinden yaş tahmini yapmada güçlü bir performans sergilemiştir.

Bu çalışma, EEG verilerinden yaş tahmini yapmak için Recurrent Neural Networks (RNN) kullanımının etkinliğini ortaya koymaktadır. Modelin elde ettiği yüksek doğruluk oranları, RNN’nin biyometrik analizlerde güçlü bir yaklaşım olduğunu göstermiştir [17].

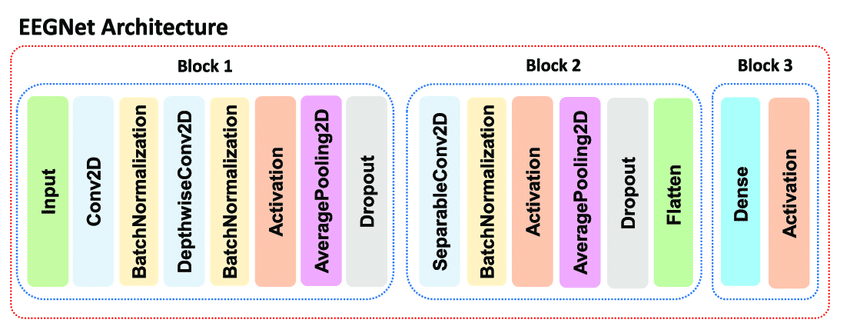
### 3.4.5. EEGNet ile Yaş Tahmini

EEG verilerinden yaş tahmini yapmak için EEGNet kullanımı, sinyal işleme ve derin öğrenme alanında önemli bir gelişme olarak değerlendirilmektedir. EEGNet, EEG sinyallerinden anlamlı öznitelikleri çıkarmak için tasarlanmış özel bir derin öğrenme modelidir. Model, özellikle EEG verilerinin yüksek boyutlu ve karmaşık yapısını işleyebilme yeteneği ile öne çıkar.

Bu çalışmada, yaş tahmini yapmak için EEGNet mimarisi kullanılmıştır. EEGNet, konvolüsyonel sinir ağlarının modifiye edilmiş bir versiyonudur ve EEG sinyallerinin zaman ve frekans alanında etkili bir şekilde işlenmesini sağlar. Model, çeşitli konvolüsyon katmanları ve havuzlama işlemleri aracılığıyla EEG sinyallerindeki yaşa özgü özellikleri öğrenir.

Veri seti, farklı yaş gruplarına ait geniş bir EEG kayıtları koleksiyonundan oluşmaktadır. EEGNet modelinin eğitim sürecinde, bu veriler zaman serisi olarak modele beslenmiş ve modelin derin katmanları sinyallerde yaşla ilişkili kalıpları öğrenmeye çalışmıştır. EEGNet, sinyalin hem zaman hem de frekans özelliklerini analiz ederek yaş tahminini yapar.

EEGNet, EEG verilerinden yaş tahmini yapmada etkili bir model olarak değerlendirilmiştir. Tahminlerde doğruluk oranı %87 olmuştur. Bu yöntem de EEG verilerinden yaş tahmini yapmak için kullanılabilecek güçlü yöntemlerdendir [18].



**Şekil 3.1.** EEGNet Mimarisi

# 4. PROJE

## 4.1. Veri Seti Tanımı

Bu çalışma, dinlenme durumunda kaydedilen elektroensefalografi (EEG) verilerini incelemektedir. Veriler, Norveç Oslo Üniversitesi Psikoloji Bölümü'nde gerçekleştirilmiş olup, SRM projesinin deneysel paradigmalarından elde edilmiştir. Bu veri seti, dinlenme sırasında, gözleri kapalı halde bulunan deneklerin EEG sinyallerini içermektedir ve bu sinyaller, zihinsel durumları anlamak ve bilişsel süreçleri incelemek için önemli bir kaynak sunmaktadır.

Veriler 111 adet bireyin gözleri kapalı durumdayken dinlenme halindeki EEG verilerinden toplanmasından oluşmaktadır. Veriler 69 kadın ve 42 erkek bireyden oluşmakta ve her birey için 20.000 adet EEG verisi bulunmaktadır. Toplam veri sayısı 2.220.000’dir.

## 4.2. Verilerin Toplanması

Bu çalışmada kullanılan EEG verileri, BioSemi ActiveTwo sistemi ile 64 elektrot kullanılarak kaydedilmiştir. Elektrotlar, genişletilmiş 10-20 sistemine (10-10) uygun bir pozisyon şemasına göre yerleştirilmiştir. Her bir EEG kaydı, deneklerin gözleri kapalı halde dinlenme pozisyonunda bulunduğu dört dakikalık bir süre boyunca gerçekleştirilmiştir. Veriler, Avrupa'da kaydedildiği için 50 Hz hat gürültüsü frekansı filtrelenmemiştir. Deneklerin EEG kayıtlarına ait zaman noktaları, \*\_scans.tsv dosyasında detaylı olarak listelenmiştir. Bazı deneklere ait tek bir EEG dosyası varken, diğer denekler için iki ayrı kayıt yapılmış ve bu dosyalar veri setine dahil edilmiştir. Bu kapsamda elde edilen ham EEG verileri, ortalama referans ile yeniden referanslanmış, ancak bunun dışında herhangi bir işlemden geçirilmemiştir. Ham veri dosyalarında olay işaretçileri bulunmamakta olup, tüm kayıt kesintisiz dinlenme durumu verilerini içermektedir. Verilerin kalitesi, veri seti sağlanmadan önce dikkatlice değerlendirilmemiştir. Çoğu veri dosyası yüksek kalitede olmakla birlikte, bazı dosyalar düşük kalitede olabilir. Bu sebeple, veri seti "olduğu gibi" sağlanmakta ve her bir veri dosyasının kalitesini doğrulamak kullanıcıların sorumluluğundadır [19].

## ****4.3. Veri Setinin İçeriği****

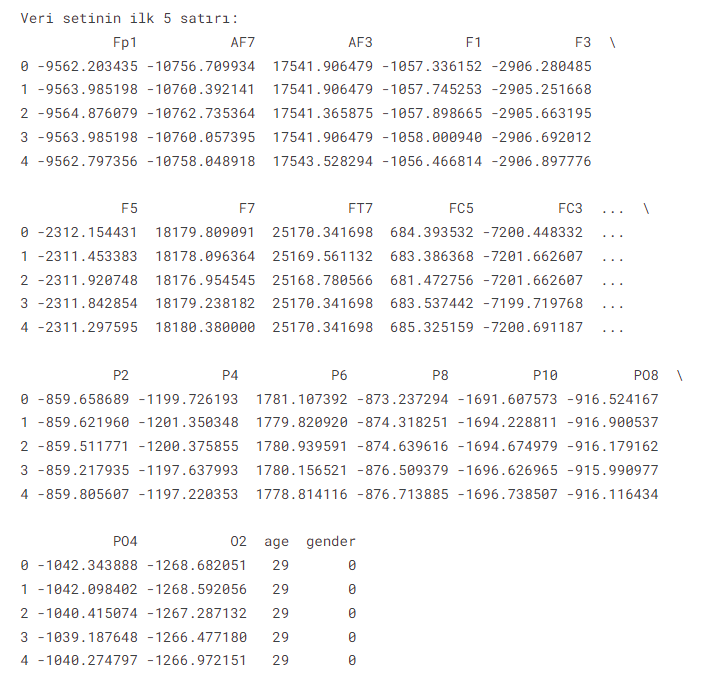
### 4.3.1. EEG Elektrot Sütunları

* **Fp1, Fp2:** Frontal polar (ön frontal) bölgedeki elektrotlar. Bu bölge genellikle duygusal ve kognitif süreçlerle ilişkilendirilir.
* **AF7, AF3, AF4, AF8:** Anterior frontal (ön frontal) bölgeye yerleştirilen elektrotlar. Genellikle dikkat, konsantrasyon ve karar verme süreçleri ile ilişkilidir.
* **F1, F3, F5, F7, F2, F4, F6, F8:** Frontal bölgedeki elektrotlar. Bu bölge motor beceriler, dil ve hafıza işleme gibi yüksek düzeydeki kognitif işlevlerle ilişkilidir.
* **FT7, FT8:** Fronto-temporal bölgeye yerleştirilen elektrotlar. Bu bölge dil işleme ve hafıza ile ilişkilidir.
* **FC5, FC3, FC1, FCz, FC2, FC4, FC6**: Fronto-central bölgeye yerleştirilen elektrotlar. Bu bölge motor kontrol ve hareket planlamasıyla ilgilidir.
* **C1, C3, C5, Cz, C2, C4, C6:** Central bölgeye yerleştirilen elektrotlar. Bu bölge somatosensoryal işleme (vücut duyularının algılanması) ile ilişkilidir.
* **T7, T8:** Temporal (şakak) bölgedeki elektrotlar. Bu bölge hafıza, dil işleme ve duygusal işlemlerle ilişkilidir.
* **TP7, TP8:** Temporo-parietal bölgeye yerleştirilen elektrotlar. Bu bölge mekansal farkındalık ve entegrasyon ile ilişkilidir.
* **CP5, CP3, CP1, CPz, CP2, CP4, CP6:** Centro-parietal bölgeye yerleştirilen elektrotlar. Genellikle duyusal işleme ve motor planlamayla ilişkilidir.
* **P1, P3, P5, P7, P9, P2, P4, P6, P8, P10, Pz:** Parietal bölgeye yerleştirilen elektrotlar. Bu bölge mekansal farkındalık, dokunma algısı ve vücut oryantasyonu ile ilişkilidir.
* **PO7, PO3, PO4, PO8, POz:** Parieto-occipital bölgeye yerleştirilen elektrotlar. Görsel işleme ve mekansal farkındalıkla ilgilidir.
* **O1, O2, Oz:** Occipital (arka) bölgedeki elektrotlar. Görsel işleme ve görsel bilgi algısı ile ilişkilidir.
* **Iz:** İnion, beynin en arka kısmındaki elektrot. Görsel işleme ile ilişkilidir.
* **Fpz:** Fp1 ve Fp2'nin ortasında yer alan, frontal polar bölgedeki elektrot.

### 4.3.2. Demografik Bilgi Sütunları

* **age:** Katılımcının yaşı
* **gender:** Katılımcının cinsiyeti

Veri setinin ilk 5 satırı aşağıda verilmiştir. Son iki sütundaki veriler ön işleme esnasında düşürülmüştür. Giriş ve çıkışlar için farklı değişkenlere atanacaktır.



**Şekil 4.1.** Veri setinin ilk 5 satırı

### 4.3.3. Deneklerin Yaş ve Cinsiyet Verileri

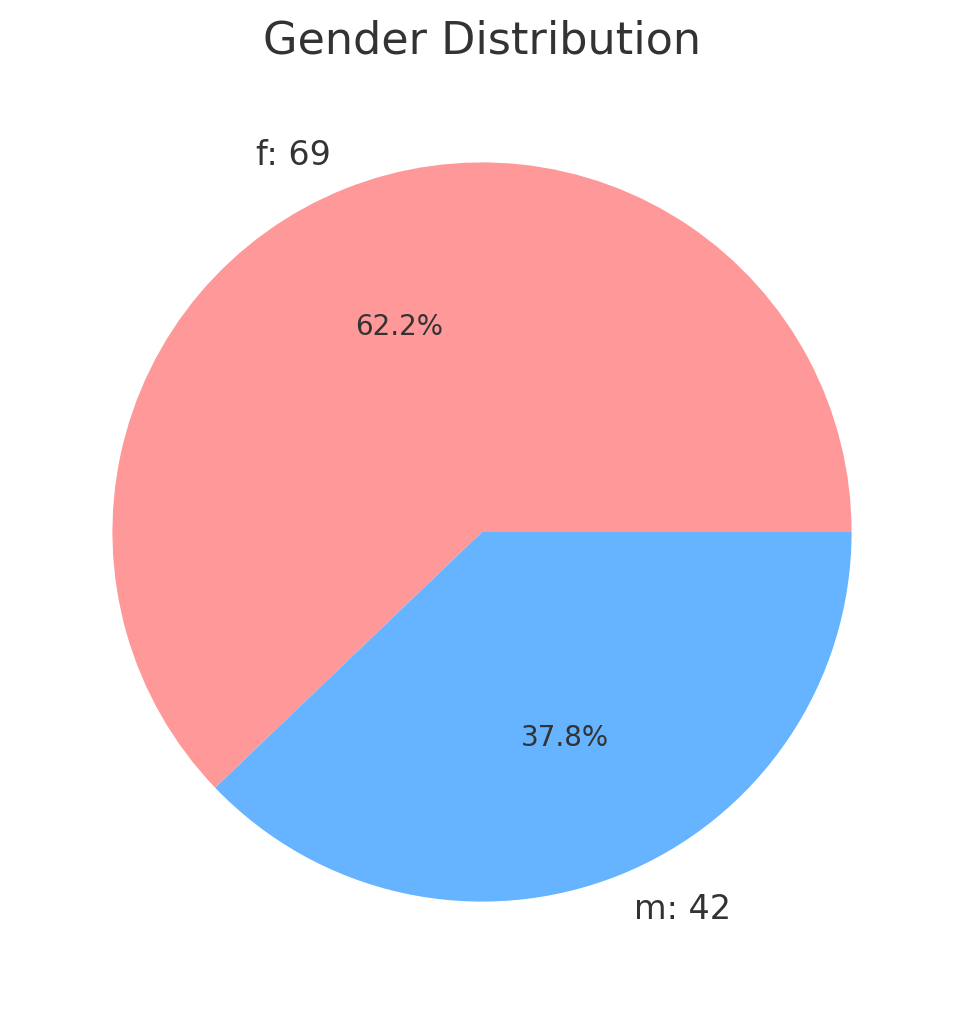
Veri setim, her bir deneğin EEG kanalları (örneğin, 'Fp1', 'AF7', vb.) ve demografik bilgilerini ('age', 'gender') içermektedir. Bu sütunlar, 'sub' başlığı altında birleştirilmiştir. Böylece, her deneğin EEG verilerini ve yaş, cinsiyet gibi bilgileri tek bir yapı içinde incelenmiştir. Deneklerin yaş ve cinsiyet tablosu aşağıda verilmiştir.

|  |  |  |
| --- | --- | --- |
| **participant\_id** | **age** | **gender** |
| sub-001 | 29 | f |
| sub-002 | 29 | f |
| sub-003 | 62 | f |
| sub-004 | 20 | f |
| sub-005 | 32 | f |
| sub-006 | 39 | m |
| sub-007 | 37 | f |
| sub-008 | 34 | f |
| sub-009 | 19 | m |
| sub-010 | 34 | f |
| sub-011 | 46 | m |
| sub-012 | 32 | m |
| ........... | …….. | ….. |
| sub-101 | 63 | f |
| sub-102 | 43 | f |
| sub-103 | 42 | f |
| sub-104 | 46 | m |
| sub-105 | 31 | f |
| sub-106 | 23 | f |
| sub-107 | 18 | f |
| sub-108 | 50 | f |
| sub-109 | 19 | f |
| sub-110 | 39 | f |
| sub-111 | 65 | m |

**Tablo 1.** Veri setindeki deneklerin yaşları ve cinsiyetleri tablosu

## 4.4. Grafiksel Analiz Ve Dağılımlar

Veri setimizde toplam 111 birey bulunmaktadır. Cinsiyet dağılımı şu şekildedir: Kadınlar %62.2 (69 birey) ve Erkekler %37.8 (42 birey). Bu dağılım, modelimizin her iki cinsiyeti de temsil eden dengeli bir veri setine sahip olduğunu göstermektedir.

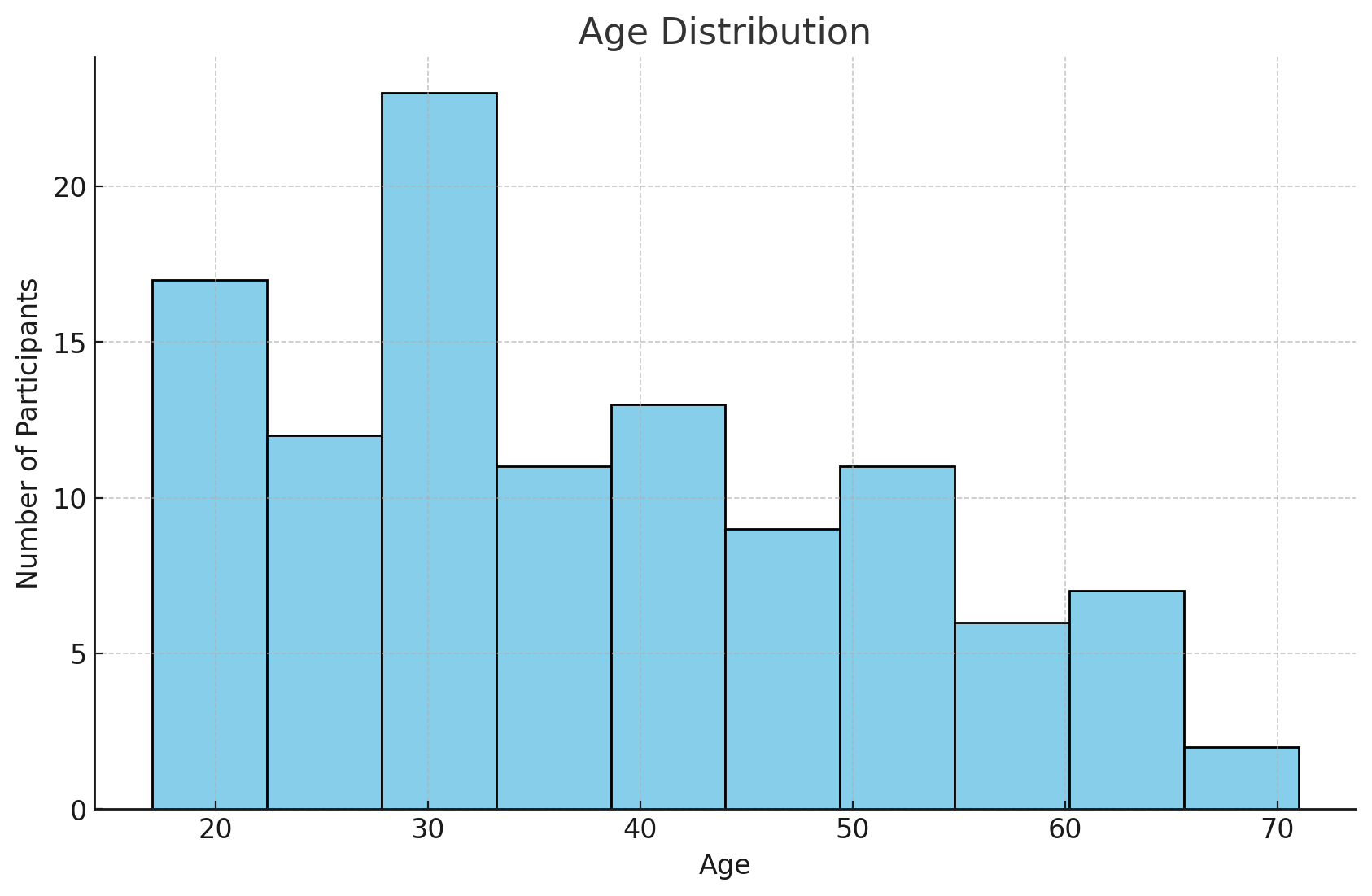


**Şekil 4.2.** Veri seti cinsiyet dağılımı

Veri setimizdeki toplam 111 birey, çeşitli yaş aralıklarını temsil etmektedir. Yaş dağılımı, yaş gruplarının aşağıdaki şekilde dağıldığını göstermektedir:

* 18-24 yaş: %16.2 (18 birey)
* 25-34 yaş: %30.6 (34 birey)
* 35-44 yaş: %21.6 (24 birey)
* 45-54 yaş: %18 (20 birey)
* 55 yaş ve üstü: %13.5 (15 birey)

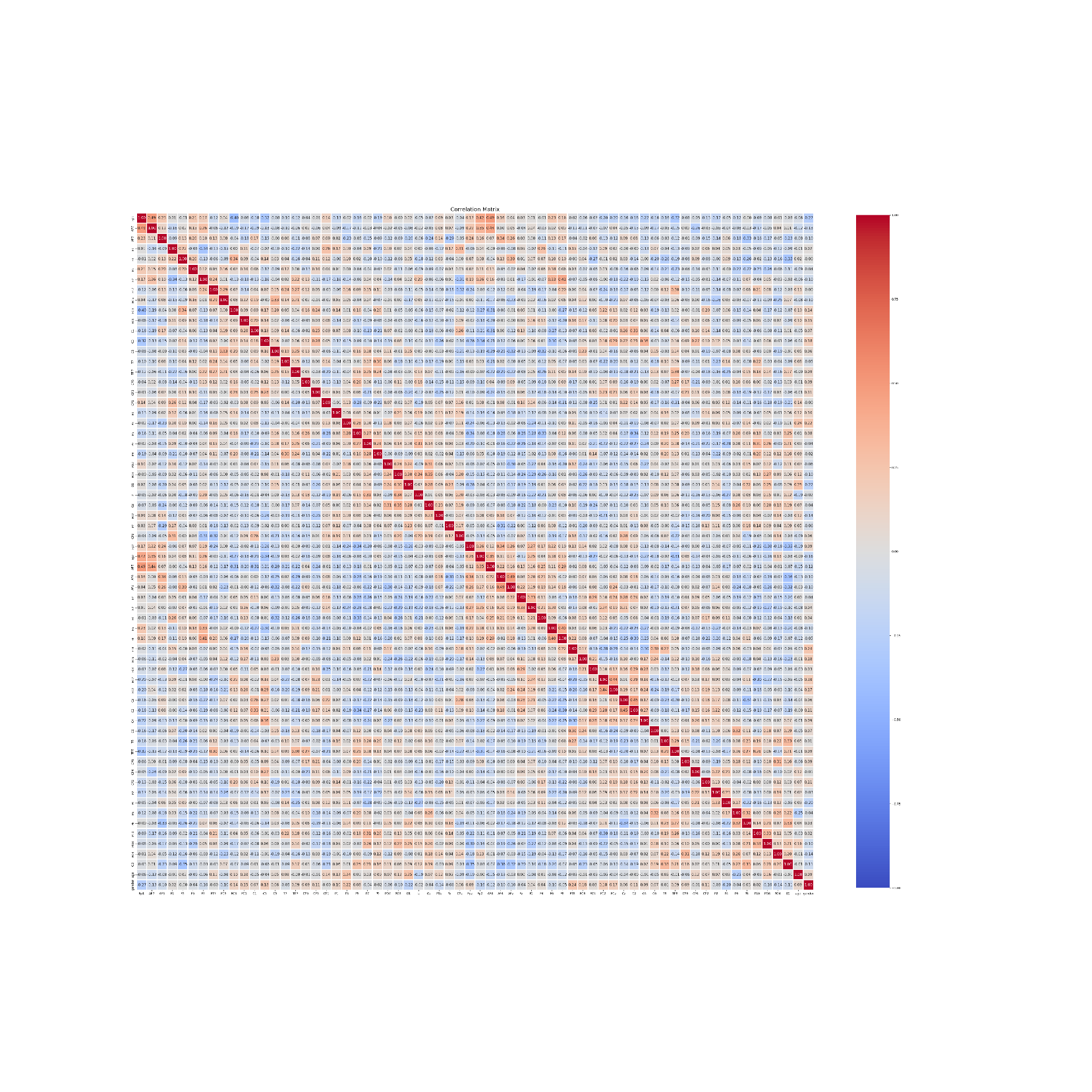
Bu dağılım, farklı yaş gruplarındaki bireylerin dengeli bir şekilde temsil edildiğini ve yaşın modelin performansını nasıl etkileyebileceğini analiz etmek için uygun bir veri seti sunduğunu göstermektedir.



**Şekil 4.3.** Veri seti yaş dağılımı

Veri setinin her bir sütununun birbiriyle ilişkisi correlation matriste gösterilmiştir. Kırmızı tonlar, pozitif korelasyonu temsil ederken, mavi tonlar negatif korelasyonu gösteriyor. Daha koyu kırmızı olan hücreler, iki değişken arasında güçlü bir pozitif ilişki olduğunu, yani biri artarken diğerinin de arttığını ifade ediyor. Öte yandan, koyu mavi hücreler ise negatif bir ilişkiyi, yani biri artarken diğerinin azaldığını gösteriyor.

Ana diagonal boyunca yer alan koyu kırmızı çizgi, her değişkenin kendisiyle olan mükemmel pozitif korelasyonunu (korelasyon katsayısı 1) simgeliyor. Diğer hücreler ise, değişkenler arasındaki farklı korelasyon derecelerini ortaya koyuyor.

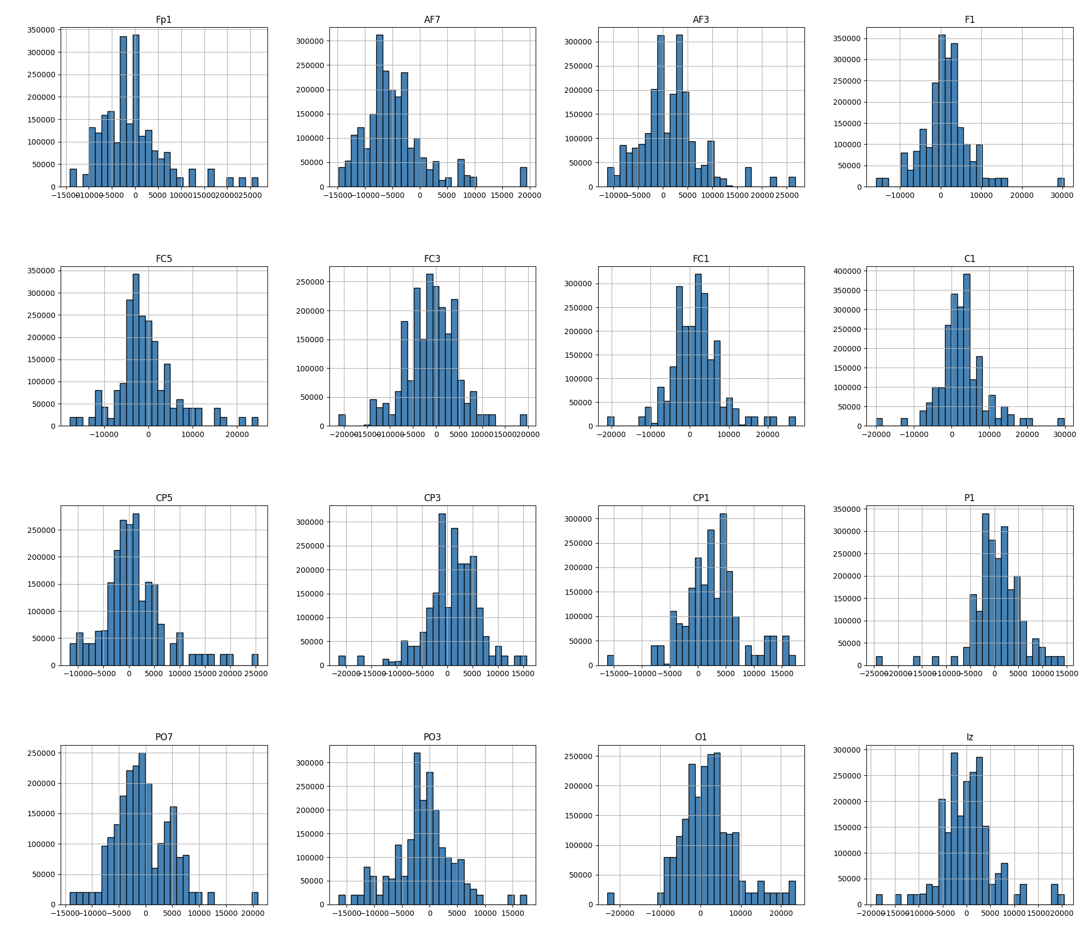


**Şekil 4.4.** Veri seti correlation matrix

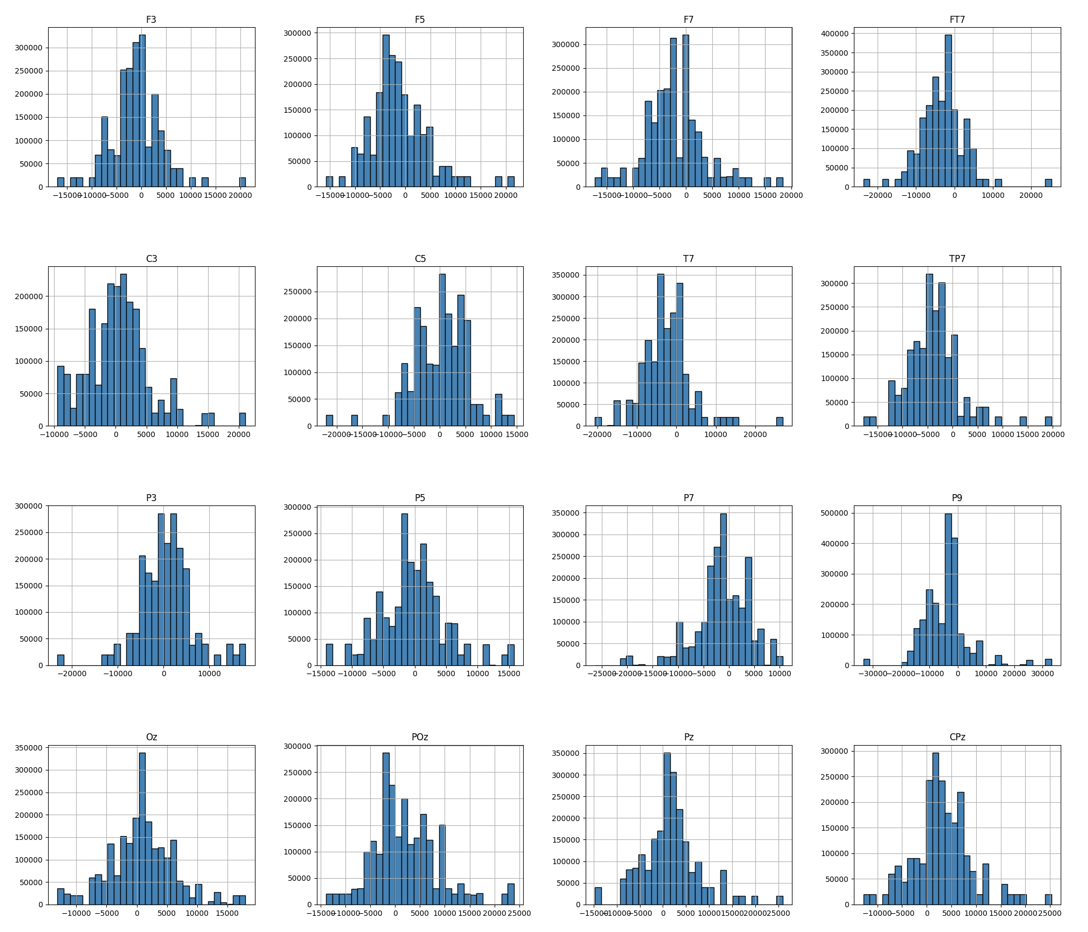
Veri setimizde, her bir birey için çeşitli özelliklerin yer aldığı 1 ana sütun bulunmaktadır. Bu sütun, aşağıdaki bilgileri içermektedir:

* EEG Kanal Verileri: EEG sinyallerini temsil eden kanallar
* Yaş: Bireylerin yaş bilgisi.
* Cinsiyet: Bireylerin cinsiyeti.

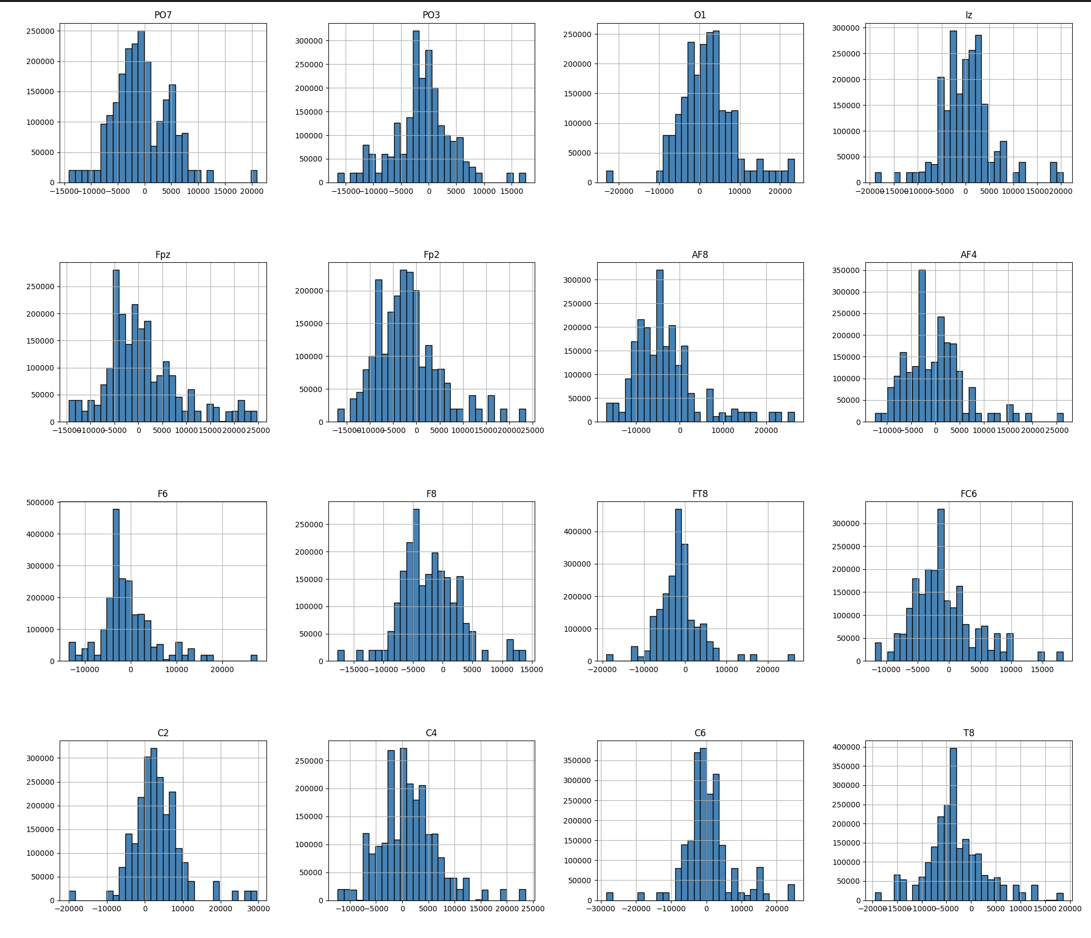
Aşağıda denek verilerinin sütun dağılımlarının gösterimi verilmiştir.



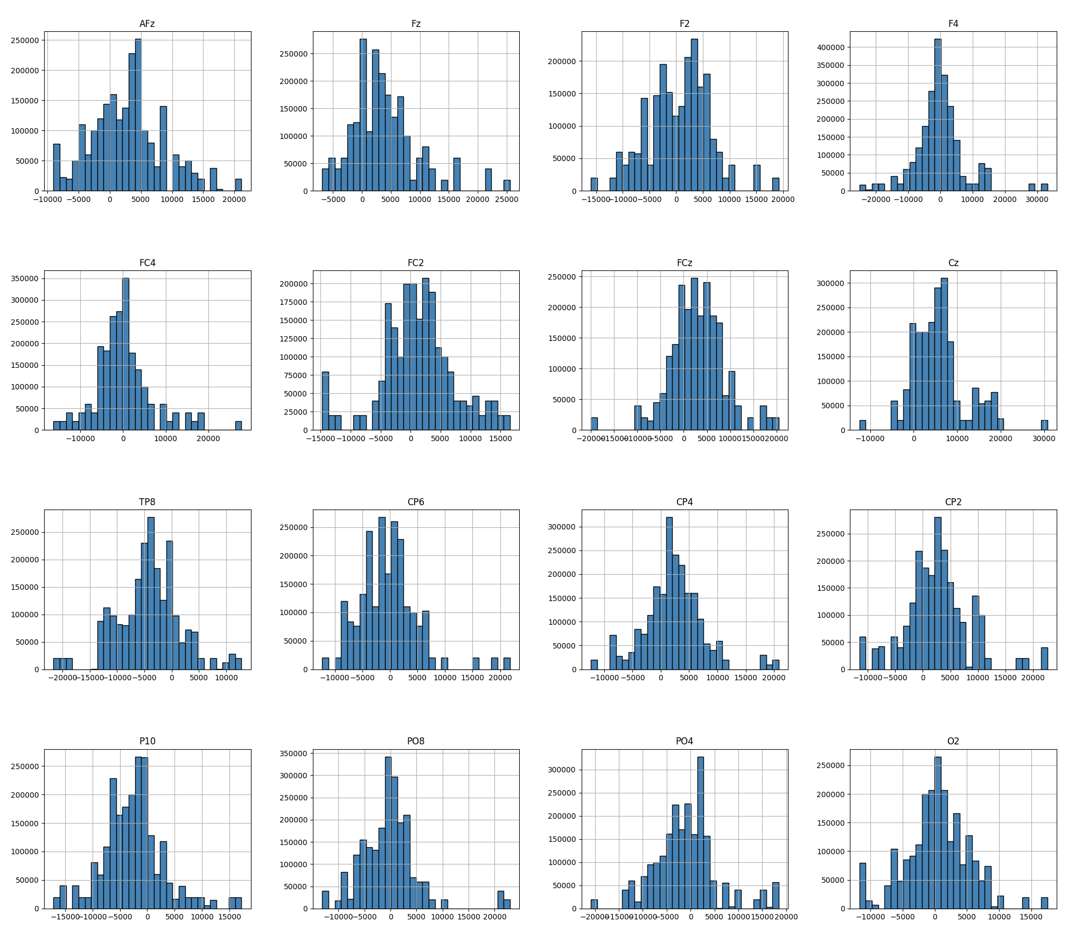
**Şekil 4.5.** Veri seti sütun dağılımları 1



**Şekil 4.6.** Veri seti sütun dağılımları 2



**Şekil 4.7.** Veri seti sütun dağılımları 3



**Şekil 4.8.** Veri seti sütun dağılımları 4

### 4..1. EEG Verilerinin İşlenmesi ve Analizinin Özeti

111 adet EDF (European Data Format) uzantılı EEG (Elektroensefalografi) verisinin, gönüllü deneklere ait demografik bilgilerle birleştirilerek analiz edilmesini amaçlanmaktadır. Çalışmanın ilk aşamasında, EDF formatındaki veriler tek tek CSV formatına dönüştürülmüştür. Dönüştürme işleminin ardından, verilerin sadece ilk 20.000 satırı kullanılarak veri seti optimize edilmiştir. İkinci aşamada, bu CSV dosyaları gönüllü deneklerin yaş, cinsiyet gibi bilgilerini içeren bir Excel dosyası ile eşleştirilerek tek bir CSV dosyası haline getirilmiştir. Son aşamada ise, veri setindeki cinsiyet sütununda bulunan "f" ve "m" harfleri sırasıyla "0" ve "1" olarak değiştirilmiştir. Bu birleşik veri seti, ileri analizler için kullanıma hazır hale getirilmiştir.

EEG, beyin aktivitelerinin elektriksel potansiyellerini kaydederek, nörolojik ve psikiyatrik hastalıkların tanı ve tedavisinde kullanılan önemli bir yöntemdir. EEG verileri genellikle EDF formatında saklanır. Ancak bu format, verilerin işlenmesi ve analiz edilmesi için her zaman uygun olmayabilir. Bu nedenle, EDF formatındaki EEG verilerinin daha yaygın olarak kullanılan CSV formatına dönüştürülmesi gerekmektedir. Bu tez çalışmasında, 111 adet EDF dosyasının CSV formatına dönüştürülmesi, verilerin belirli bir kısmının seçilmesi, gönüllü deneklere ait demografik bilgilerle birleştirilmesi ve cinsiyet verilerinin sayısal hale getirilmesi işlemi detaylandırılmıştır. Daha detaylı şekilde açıklamaları aşağıda mevcuttur.

### 4.4.2. EDF Dosyalarının CSV Formatına Dönüştürülmesi:

İlk olarak, 111 adet EDF dosyası, uygun bir yazılım aracı kullanılarak CSV formatına dönüştürülmüştür. Bu süreçte, her bir EDF dosyasındaki EEG sinyalleri ve ek veriler, ayrı ayrı CSV dosyaları olarak kaydedilmiştir.

Dönüşüm işlemi, Python programlama dili ve MNE kütüphanesi kullanılarak otomatikleştirilmiştir. Bu sayede, manuel işlem hataları minimize edilmiş ve veri dönüşümü hızlandırılmıştır.

Kod parçası aşağıda verilmiştir:

import mne

import pandas as pd

def edf\_to\_csv(edf\_file, csv\_file):

# EDF dosyasını yükle

raw = mne.io.read\_raw\_edf(edf\_file, preload=True)

# Veriyi numpy dizisine dönüştür

data = raw.get\_data()

# Kanal isimlerini alma

channel\_names = raw.info['ch\_names']

# DataFrame oluştur

df = pd.DataFrame(data.T, columns=channel\_names)

# CSV dosyasına yaz

df.to\_csv(csv\_file, index=False)

# Örnek kullanım

edf\_to\_csv('sub-02\_task-MIvsRest\_run-1\_eeg.edf', 'output\_file.csv')

### 4.4.3. Veri Setinin Optimize Edilmesi:

Dönüştürülen CSV dosyalarındaki veri miktarı oldukça büyük olduğundan, analiz sürecini hızlandırmak ve gereksiz veri yükünü azaltmak amacıyla her bir dosyanın sadece ilk 20.000 satırı kullanılmıştır. Bu işlem, veri setinin hem daha yönetilebilir hale gelmesini sağlamış hem de daha hızlı işlem yapılmasına olanak tanımıştır.

Kırpma işlemi sırasında, Python ve Pandas kütüphanesi kullanılmıştır. Her bir CSV dosyası okunmuş ve sadece ilk 20.000 satırı içeren yeni bir dosya olarak kaydedilmiştir. Bu adım, işlem süresini ve veri boyutunu önemli ölçüde azaltmıştır.

Kod parçası aşağıda verilmiştir:

import pandas as pd

import os

# CSV dosyalarının bulunduğu dizin

csv\_dir = 'csvler'

output\_dir = 'son'

if not os.path.exists(output\_dir):

os.makedirs(output\_dir)

# Tüm CSV dosyalarını işleme

for filename in os.listdir(csv\_dir):

if filename.endswith('.csv'):

# Dosyanın tam yolunu alma

file\_path = os.path.join(csv\_dir, filename)

# CSV dosyasını okuma

df = pd.read\_csv(file\_path)

# İlk 20.000 satırı alma

df = df.head(20000)

# Düzenlenmiş dosyayı yeni dizine kaydet

output\_file\_path = os.path.join(output\_dir, filename)

df.to\_csv(output\_file\_path, index=False)

print("Tüm dosyalar işlendi ve kaydedildi.")

### 4.4.4. Denek Bilgileriyle Eşleştirme:

**Denek Bilgileri**

Gönüllü deneklere ait yaş, cinsiyet ve diğer demografik bilgileri içeren bir Excel dosyası kullanılmıştır. Bu bilgiler, EEG verileri ile birleştirilerek analiz için anlamlı bir veri seti oluşturulmuştur. Excel dosyasındaki denek bilgileri Tablo 1’de verilmiştir.

**Eşleştirme Süreci**

Her bir CSV dosyası, ilgili deneğin ID'sine göre Excel dosyasındaki bilgilerle eşleştirilmiştir. Bu işlemde, Python'da Pandas kütüphanesi kullanılmış ve verilerin doğru bir şekilde eşleştirilmesi sağlanmıştır.

Kod parçası aşağıda verilmiştir:

import pandas as pd

import os

# Dosya yolları

csv\_folder = 'son'

xlsx\_file = 'ds003775\_katilimcilar.xlsx'

# Excel dosyasını oku ve sadece 'age' ve 'gender' sütunlarını seç

xlsx\_data = pd.read\_excel(xlsx\_file, usecols=['age', 'gender'])

# CSV dosyalarının adlarını al

csv\_files = sorted([f for f in os.listdir(csv\_folder) if f.endswith('.csv')])

# Listeyi başlat

combined\_df = pd.DataFrame()

# Her bir CSV dosyasını işle

for i, csv\_file in enumerate(csv\_files):

# CSV dosyasını oku

csv\_path = os.path.join(csv\_folder, csv\_file)

df = pd.read\_csv(csv\_path)

# Excel dosyasındaki uygun age ve gender verilerini al

if i < len(xlsx\_data):

participant\_info = xlsx\_data.iloc[i]

df['age'] = participant\_info['age']

df['gender'] = participant\_info['gender']

else:

# Eğer Excel dosyasındaki satır sayısı CSV dosyalarından daha azsa, boş değer ile doldurma

df['age'] = pd.NA

df['gender'] = pd.NA

# Veri çerçevesini birleştirme

combined\_df = pd.concat([combined\_df, df], ignore\_index=True)

# Birleştirilmiş veri çerçevesini yeni bir CSV dosyasına yazılması

combined\_df.to\_csv('combined\_data.csv', index=False)

### 4.4.5. Tek Bir CSV Dosyası Haline Getirme:

Birleştirme: Eşleştirilmiş veriler, tek bir CSV dosyasında toplanmıştır. Bu birleşik veri seti, tüm deneklerin EEG sinyalleri ile demografik bilgilerini içermektedir.

Veri Temizliği ve Düzenleme: Birleştirilen veri setinde, eksik veya hatalı veriler temizlenmiş, sütunlar düzenlenmiş ve veri analizine uygun hale getirilmiştir.

Kod parçası aşağıda verilmiştir:

import pandas as pd

import os

# CSV dosyalarının bulunduğu klasör

csv\_folder = 'son\_csvler copy'

# Birleştirilmiş veri çerçevesi

combined\_df = pd.DataFrame()

# CSV dosyalarını oku ve birleştir

for filename in sorted(os.listdir(csv\_folder)):

if filename.endswith('.csv'):

csv\_path = os.path.join(csv\_folder, filename)

# CSV dosyasını oku

df = pd.read\_csv(csv\_path)

# Veri çerçevesini birleştirme

combined\_df = pd.concat([combined\_df, df], ignore\_index=True)

# Tek bir CSV dosyası olarak kaydet

combined\_df.to\_csv('combined\_data.csv', index=False)

print("All CSV files have been successfully combined into 'combined\_data.csv'.")

### 4.4.6. Cinsiyet Verilerinin Sayısal Hale Getirilmesi:

#### Dönüştürme Süreci

Birleştirilen veri setindeki cinsiyet sütununda bulunan "f" ve "m" harfleri, sırasıyla "0" ve "1" ile değiştirilmiştir. Bu işlem, veri analizinde cinsiyet verilerinin daha kolay kullanılmasını sağlamıştır.

Dönüştürme işlemi, Python'da Pandas kütüphanesi kullanılarak gerçekleştirilmiştir. Bu sayede, cinsiyet verileri numerik hale getirilmiş ve modellemelerde kullanılabilir hale gelmiştir.

Kod parçası aşağıda verilmiştir:

import pandas as pd

# CSV dosyasını oku

df = pd.read\_csv('combined\_data.csv')

# gender sütunundaki 'f' ve 'm' değerlerini 0 ve 1 olarak güncelle

df['gender'] = df['gender'].map({'f': 0, 'm': 1})

# Güncellenmiş veriyi tekrar CSV dosyasına kaydet

df.to\_csv('veriseti\_2.csv', index=False)

EDF formatındaki EEG verilerinin CSV formatına dönüştürülmüş, belirli bir kısmı seçilmiş, gönüllü deneklerin demografik verileriyle birleştirilmiştir. Elde edilen birleşik veri seti, EEG verilerinin daha kolay analiz edilmesini sağlamış ve bu tür verilerle çalışacak araştırmacılar için bir rehber niteliği taşımaktadır. Veri setinin optimize edilmesi ve cinsiyet verilerinin sayısal hale getirilmesi, işlem sürelerini kısaltmış ve daha verimli bir analiz süreci sağlamıştır.

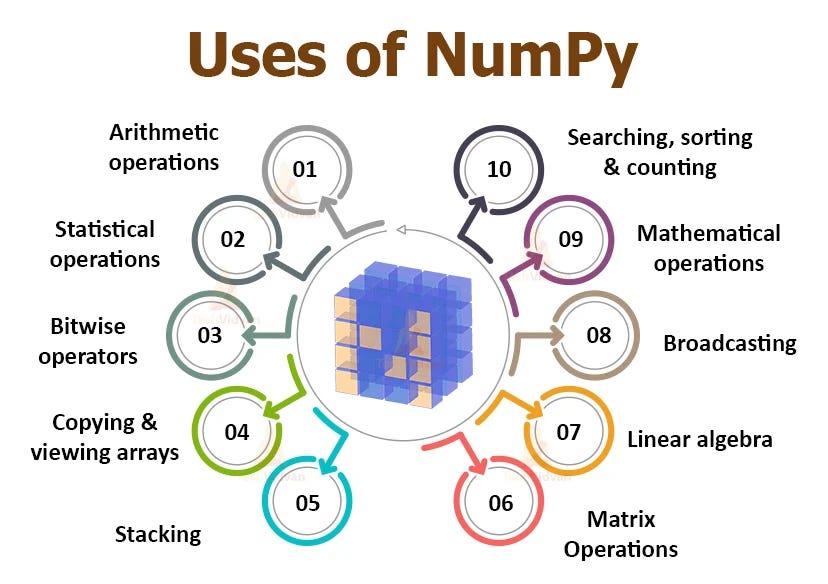
Veri kırpma işlemi, büyük veri setleri ile çalışırken işlem süresini ve kaynak kullanımını optimize etmek açısından kritik öneme sahiptir. Ayrıca, cinsiyet verilerinin sayısal hale getirilmesi, veri analizi ve modelleme süreçlerini kolaylaştırmıştır. Ancak, bu işlemler sırasında kaybedilen verilerin ve yapılan dönüşümlerin analiz sonuçlarına etkisi dikkatle değerlendirilmelidir.

## 4.5. Gerekli Kütüphaneler

### 4.5.1. NumPy

NumPy, Python'da bilimsel hesaplama için kullanılan temel kütüphanelerden biridir. NumPy, özellikle büyük veri dizileri ve matrislerle çalışmayı kolaylaştıran çok boyutlu dizi nesneleri ve bu nesneler üzerinde hızlı matematiksel işlemler yapmayı sağlayan fonksiyonlar sağlar.

Matris işlemleri, lineer cebir hesaplamaları, Fourier dönüşümleri gibi çeşitli bilimsel ve mühendislik uygulamalarında, büyük veri kümelerinin verimli bir şekilde saklanması ve işlenmesi, verilerin önişleme aşamalarında, özelliklerin normalizasyonu ve vektörleştirilmesinde sıkça kullanılır [20].

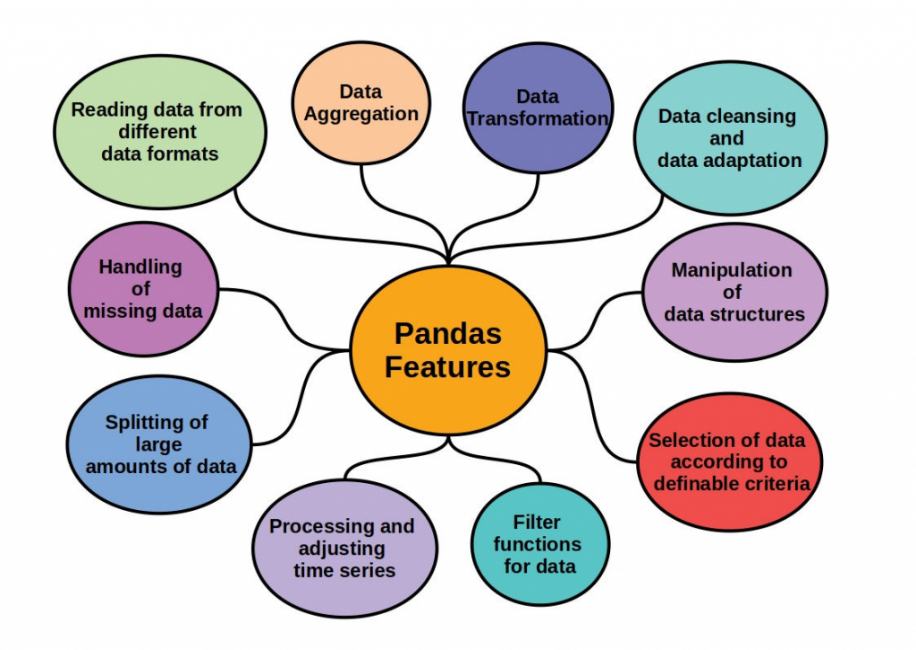


**Şekil 4.9.** Numpy

### 4.5.2. Pandas

Pandas, veri işleme ve analiz için geliştirilmiş bir Python kütüphanesidir. Yapısal verilerin (örneğin, tablolar, zaman serileri) okunması, işlenmesi, filtrelenmesi ve manipüle edilmesi için güçlü araçlar sağlar.

Verilerin okunması, temizlenmesi, filtrelenmesi ve yeniden şekillendirilmesi, zaman serisi verilerinin analizi ve görselleştirilmesi, eğitim verilerinin hazırlanması, özniteliklerin seçimi ve veri kümesi bölme işlemleri gibi işlemlerde kullanılır [21].

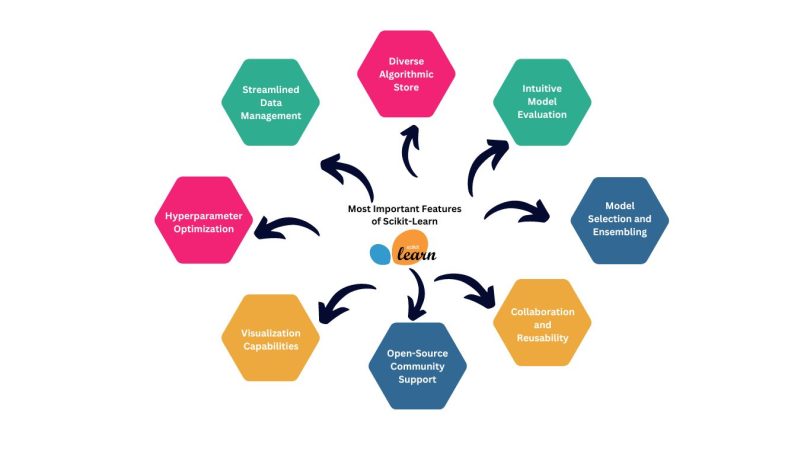


**Şekil 4.10.** Pandas

### 4.5.3. Scikit-Learn

Python'da makine öğrenmesi algoritmalarını uygulamak için kullanılan popüler bir kütüphanedir. Model oluşturma, eğitim ve değerlendirme için gerekli araçları sağlar.

Veri kümesini eğitim ve test setlerine ayırmak için kullanılır, verilerin ölçeklendirilmesi, makine öğrenmesi modellerinin performansını artırmak için yaygın olarak kullanılır [22].

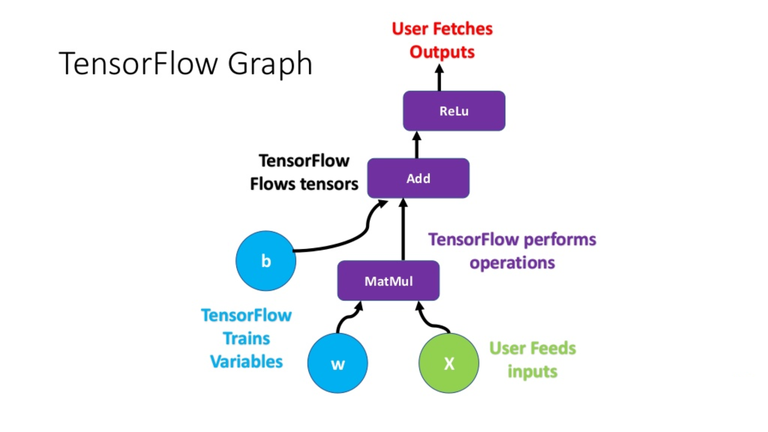


**Şekil 4.11.** Scikit-Learn

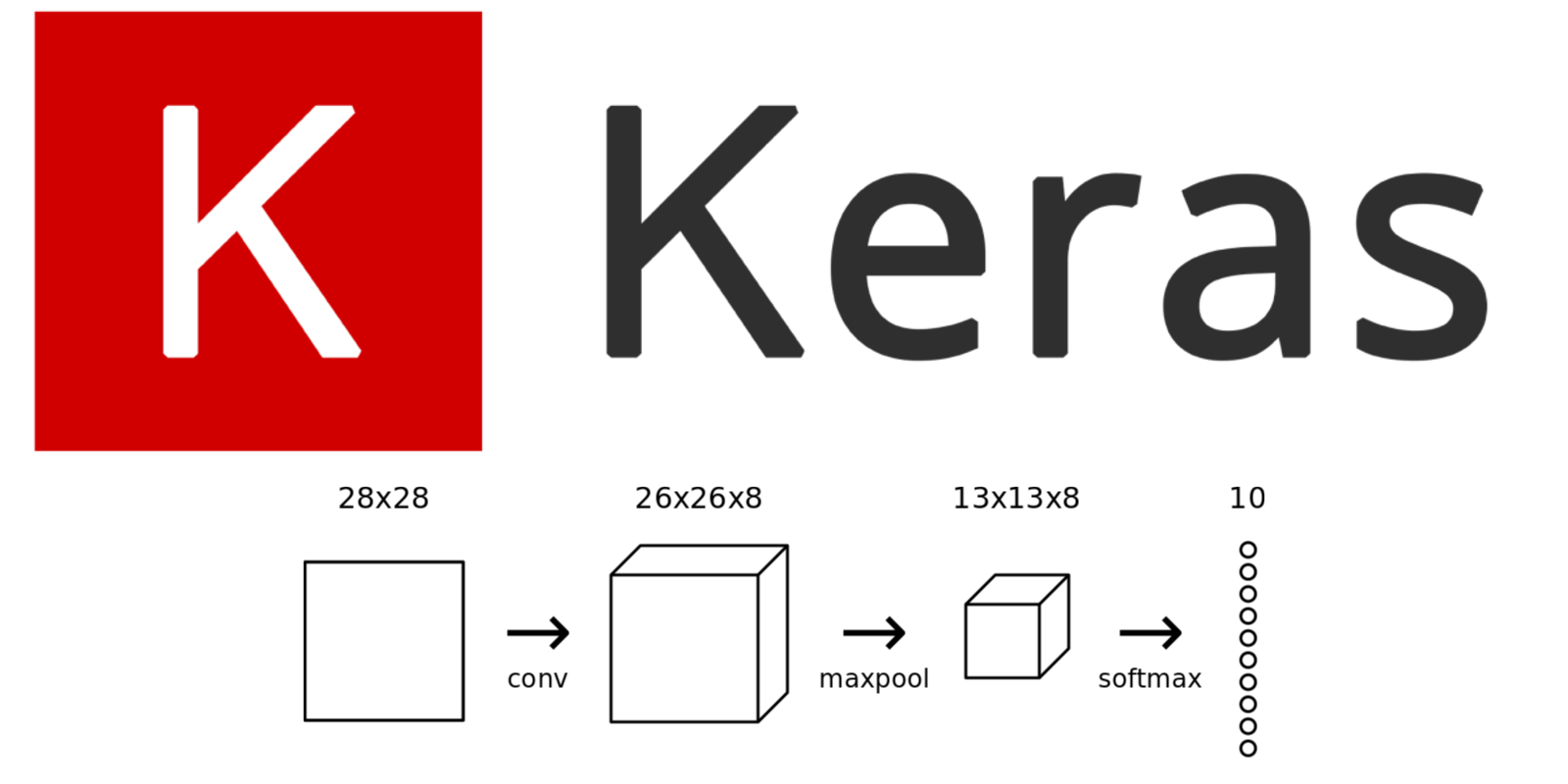
### 4.5.4. TensorFlow ve Keras

TensorFlow, Google tarafından geliştirilen açık kaynaklı bir makine öğrenmesi kütüphanesidir. TensorFlow, derin öğrenme modellerinin oluşturulması, eğitilmesi ve dağıtılması için güçlü araçlar sağlar. Keras, TensorFlow üzerine inşa edilmiş, derin öğrenme modellerinin oluşturulmasını kolaylaştıran bir yüksek seviyeli API'dir.

Sinir ağlarının tanımlanması, eğitilmesi ve test edilmesi, zaman serileri ve sıralı veri türleriyle çalışırken LSTM katmanları, klasik sinir ağı yapıları için Dense katmanları kullanılır. Eğitim sırasında en iyi performansı gösteren modelin kaydedilmesi için de kullanılabilir [23].



**Şekil 4.12.** TensorFlow



**Şekil 4.13.** Keras

### 4.5.5. Joblib

Joblib, Python'da büyük veri nesnelerinin seri hale getirilmesi (kaydedilmesi ve tekrar yüklenmesi) için kullanılan bir kütüphanedir. Özellikle büyük Numpy dizileri veya makine öğrenme modelleri gibi verimli bir şekilde saklanması gereken veri türleri için kullanılır.

Eğitilmiş modellerin kaydedilmesi ve gerektiğinde tekrar yüklenmesi, ölçekleyiciler gibi eğitim sırasında kullanılan nesnelerin kaydedilmesi için kullanılır.



**Şekil 4.14.** Joblib

Veri kümesi üzerinde önce yaş ve cinsiyet etiketleri ayrı ayrı hedef değişkenler olarak belirlenmiş, ardından veri ön işleme ve model geliştirme süreçlerine geçilmiştir.

Veri ön işleme aşamasında, veri kümesinden özellikler (X) ve hedef değişkenler (yaş ve cinsiyet) ayrıştırılmıştır. Veriler, eğitim ve test setlerine bölünmüş, ardından Standard Scaler kullanılarak ölçeklendirilmiştir. Ölçeklendirilmiş veriler, LSTM (Long Short-Term Memory) katmanlarını içeren bir derin öğrenme modeline giriş olarak verilmek üzere yeniden şekillendirilmiştir.

Model oluşturma sürecinde, yaş ve cinsiyet tahmini için iki ayrı LSTM modeli geliştirilmiştir. Her bir model, iki LSTM katmanı ve bir çıkış katmanından oluşmaktadır. Modeller, kayıp fonksiyonu olarak ortalama kare hatayı (MSE) kullanarak optimize edilmiş ve eğitim sürecinde en iyi sonuç veren modelin kaydedilmesi için ModelCheckpoint kullanılmıştır.

Son aşamada, test verileri kullanılarak modellerin performansı değerlendirilmiş ve yaş ile cinsiyet tahminleri yapılmıştır. Tahmin sonuçları, yaş için bir regresyon değeri olarak, cinsiyet için ise olasılık temelli bir sınıflandırma sonucu olarak elde edilmiştir. Modellerin performansı doğrulama seti üzerinde izlenmiş ve en iyi sonuç veren model kaydedilmiştir.

## 4.6. Gerçekleştirme Aşamaları

### 4.6.1. Veri Setini Yükleme

pd.read\_csv() fonksiyonuyla pandas kütüphanesi tarafından sağlanmıştır ve CSV (Comma Separated Values) formatındaki bir dosyayı bir DataFrame olarak yüklemek için seçilmiştir. CSV dosyaları genellikle veri tablolarını saklamak için yaygın olarak kullanılan bir formattır.

Verilerin depolandığı dosya veriseti\_2.csv olarak belirlenmiştir. Fonksiyon bu dosyayı okumuş ve içindeki veriler data adlı bir DataFrame nesnesine yüklenmiştir. Bu aşamada, veriler henüz işlenmemiş ve ham halde bulunmaktadır.

### 4.6.2. Özellik ve Hedef Değişkenlerin Ayrılması

Veri seti, modelin girdi olarak kullanacağı özellikler ve modelin tahmin edeceği hedef değişkenler olarak ayrılmıştır. Tüm satırlar seçilirken, son iki sütun dışındaki tüm sütunlar özellikler (X) olarak belirlenmiştir. Bu sütunlar modelin öğrenme sürecinde kullanılan giriş verilerini temsil etmektedir.

Son iki sütun ise hedef değişkenler olarak ayrılmıştır: sondan ikinci sütun yaş (y\_age) ve son sütun cinsiyet (y\_gender) değişkenleri olarak kullanılmıştır. Bu sütunlar, modelin tahmin etmeye çalışacağı hedef değerleri temsil eder.

Modelin performansını değerlendirmek için veri seti, eğitim ve test setleri olarak ikiye ayrılmıştır. Eğitim seti, modelin öğrenme süreci için kullanılırken, test seti modelin genelleme yeteneğini değerlendirmek için kullanılmıştır.

Bu bölme işlemi, yaş tahmini için ayrı, cinsiyet tahmini için ayrı olarak gerçekleştirilmiştir. Eğitim ve test setlerine ayrılma oranı %80 eğitim, %20 test olacak şekilde ayarlanmıştır. Bu oran, modelin öğrenmesi için yeterli veri sağlarken, performans değerlendirmesi için de yeterli test verisi bulundurmayı amaçlar.

Rastgele bir bölme işlemi yapılmış, ancak aynı rastgele bölmenin tekrar elde edilebilmesi için bir rastgelelik kontrolü sağlanmıştır. Bu, modelin farklı çalıştırmalarında aynı sonuçların elde edilmesini ve böylece model performansının tutarlı bir şekilde değerlendirilebilmesini sağlamaktadır.

Kod parçası aşağıda verilmiştir:

# 1. Veri Yükleme

data = pd.read\_csv('veriseti\_2.csv')

# 2. Veri Ön İşleme

X = data.iloc[:, :-2].values

y\_age = data.iloc[:, -2].values

y\_gender = data.iloc[:, -1].values

X\_train, X\_test, y\_age\_train, y\_age\_test = train\_test\_split(X, y\_age, test\_size=0.2, random\_state=42)

\_, \_, y\_gender\_train, y\_gender\_test = train\_test\_split(X, y\_gender, test\_size=0.2, random\_state=42)

### 4.6.3. Verilerin Standardizasyonu

Modelin performansını artırmak ve özelliklerin eşit ağırlıkta değerlendirilmesini sağlamak için veri seti üzerinde ölçeklendirme işlemleri yapılmıştır. Bu işlemler, verilerin ortalamasını 0 ve standart sapmasını 1 olacak şekilde düzenlemiştir. Eğitim verileri üzerinde bir StandardScaler nesnesi oluşturulmuş ve bu nesne kullanılarak veriler standardize edilmiştir. Eğitim sürecinde öğrenilen bu ölçeklendirme parametreleri, test verilerine de aynı şekilde uygulanarak modelin doğruluğu ve tutarlılığı korunmuştur.

Ölçeklendirme nesnesi dosya olarak kaydedilmiştir. Bu adım, modelin yeni veriler üzerinde de aynı standartlaştırma işlemlerini uygulayabilmesi için kritik öneme sahiptir. Böylece, modelin eğitimi sonrasında veya başka veri setleri üzerinde kullanılacağı durumlarda aynı ölçeklendirme yöntemleri tutarlı bir şekilde tekrar uygulanabilir.

### 4.6.4. Verilerin Yeniden Şekillendirilmesi

Modelin giriş verilerini doğru bir formatta alabilmesi için eğitim ve test verileri yeniden şekillendirilmiştir. Eğitim ve test verileri, genellikle derin öğrenme modelleri olan CNN veya RNN gibi modeller için, belirli bir boyut düzeninde olmalarını gerektirir. Bu sebeple, veriler (örnek sayısı, özellik sayısı, 1) şeklinde üç boyutlu bir yapıya getirilmiştir.

Bu işlem, modelin giriş verilerini uygun bir biçimde almasını ve veriler arasında anlamlı ilişkiler kurabilmesini sağlar. Verilerin bu şekilde düzenlenmesi, modelin hem eğitim hem de test süreçlerinde tutarlı bir performans göstermesine yardımcı olur.

Kod parçası aşağıda verilmiştir:

scaler = StandardScaler()

X\_train = scaler.fit\_transform(X\_train)

X\_test = scaler.transform(X\_test)

# Scaler'ı kaydetme

scaler\_filename = "scaler.save"

joblib.dump(scaler, scaler\_filename)

X\_train = X\_train.reshape((X\_train.shape[0], X\_train.shape[1], 1))

X\_test = X\_test.reshape((X\_test.shape[0], X\_test.shape[1], 1))

### 4.6.5. Modelin Tanımlanması

Modelin oluşturulması için bir fonksiyon tanımlanmıştır. Bu fonksiyon, LSTM modelini oluşturarak modelin yapısını tanımlar ve bu modeli geri döndürür. LSTM modelleri, özellikle zaman serisi verileri veya dizinli veri türleriyle çalışırken tercih edilen bir model türüdür.

Modelin oluşturulma süreci, Keras kütüphanesi kullanılarak gerçekleştirilmiştir. Model, Sequential sınıfı kullanılarak oluşturulmuştur. Sequential modeli, katmanların sırayla eklenmesine olanak tanır ve her katmanın bir öncekinden veri almasını sağlar. Bu yapı, modelin katmanlarını düzenli bir şekilde tanımlamaya ve verinin model boyunca nasıl aktığını kontrol etmeye olanak tanır.

Modelin ilk katmanı olarak bir LSTM katmanı eklenmiştir. Bu katman, belirli sayıda hücre (bu örnekte 50) ve geri dönen dizilerle (return\_sequences=True) yapılandırılmıştır. Bu ayar, her zaman adımında bir çıktı dizisi döndürülmesini sağlar, bu da birden fazla LSTM katmanının kullanılmasına olanak tanır. Ayrıca, modelin giriş şekli (input\_shape) belirlenmiştir; bu, modelin ilk katmanı için giriş verilerinin boyutunu belirtir. Giriş şekli, (örnek sayısı, özellik sayısı, kanal sayısı) formatında tanımlanmış olup, modelin eğitim verileri üzerinde çalışabilmesi için gereklidir.

#### LSTM(50)

Modelin ilk katmanı olarak 50 hücreden oluşan bir LSTM (Long Short-Term Memory) katmanı eklenmiştir. LSTM katmanları, zaman içindeki bilgileri depolamak ve işlemek için tasarlanmış yapılar olup, belirli sayıda hücre içerir. Bu hücrelerin sayısı, modelin karmaşıklığını ve öğrenme kapasitesini belirler. LSTM katmanında return\_sequences=True parametresi kullanılmıştır. Bu parametre, her zaman adımında tüm ardışık LSTM hücrelerinden bir çıktı dizisi döndürülmesini sağlar. Böylece model, sıradaki her zaman adımı için bir çıktı üretir ve bu çıktı, modelin sonraki katmanlarına aktarılır. Özellikle birden fazla LSTM katmanı kullanıldığında bu ayar önemlidir, çünkü sonraki LSTM katmanının da bir dizi beklemesini sağlar. Modelin giriş şekli input\_shape parametresiyle belirtilmiştir. İlk değer (örneğin, X\_train.shape[1]), zaman adımlarının sayısını, ikinci değer ise her zaman adımındaki özellik sayısını ifade eder. Bu yapıda özellik sayısı 1 olarak belirlenmiştir. İkinci bir LSTM katmanı eklenmiş olup, bu katman bir önceki LSTM katmanından gelen çıktı dizisini alır ve işleyerek nihai bir çıktı üretir. Bu katman, sadece son zaman adımından bir çıktı üretir çünkü return\_sequences=False varsayılan olarak kullanılmıştır. Modelin son katmanı, tam bağlantılı (fully connected) bir Dense katmanıdır. Bu katman, LSTM katmanından gelen çıktıyı tek bir nörona indirger ve böylece model, tek bir sayısal tahmin üretir. Bu yapı, özellikle regresyon problemlerinde kullanılır.

Modelin derlenmesi için adam optimizasyon algoritması ve MSE kayıp fonksiyonu kullanılmıştır. Adam optimizasyon algoritması, adaptif öğrenme oranlarını kullanarak modelin parametrelerini günceller. Bu algoritma, genellikle hızlı ve etkili olduğu için tercih edilir. MSE, modelin tahminlerinin ne kadar yanlış olduğunu ölçen ve hataları geri yayarak modelin öğrenme sürecini yönlendiren bir kayıp fonksiyonudur.

### 4.6.6. Sistemin Çalışma Prensibi

LSTM'nin içinde yer alan kapı mekanizmaları (unutma kapısı, giriş kapısı ve çıkış kapısı), modelin hangi bilgileri saklaması, hangi bilgileri unutması ve hangi bilgileri kullanması gerektiğini öğrenmesine olanak tanır. Bu sayede, model önemli bilgileri hücre durumunda saklayarak uzun süreli bağımlılıkları öğrenir ve kullanır.

Unutma Kapısı: Hücre durumundan hangi bilgilerin silineceğine karar verir. Bu, modelin gereksiz veya eski bilgileri unutarak daha etkin öğrenmesine olanak tanır.

Giriş Kapısı: Hücre durumuna hangi yeni bilgilerin ekleneceğine karar verir. Bu kapı, modelin yeni gelen bilgilere adaptasyonunu sağlar.

Çıkış Kapısı: Hücre durumundan hangi bilgilerin çıktıya yansıtılacağına karar verir. Bu, modelin tahminlerinde hangi bilgilerin kullanılacağını belirler.

### 4.6.7. Katmanlar Arasındaki İlişkiler

Bu modelde iki adet LSTM katmanı bulunmaktadır. İlk LSTM katmanı, girdi verilerindeki zaman serisi bağımlılıklarını öğrenir ve her zaman adımında bir çıktı üretir (return\_sequences=True). Bu çıktı, ikinci LSTM katmanına aktarılır. İkinci LSTM katmanı, bu çıktılardan bir nihai zaman adımı çıktısı üretir.

İlk LSTM Katmanı: Giriş verilerinden anlamlı özellikler çıkarır ve bu özellikleri ikinci LSTM katmanına aktarır.

İkinci LSTM Katmanı: İlk katmandan gelen bilgileri daha da işleyerek son tahmini üretir.

### 4.6.8. Son Katman (Dense Katmanı)

Modelin son katmanı olan Dense katmanı, LSTM katmanlarından gelen çıktıyı tek bir nöronla işlemektedir. Bu nöron, modelin nihai tahminini üretir. Bu, modelin regresyon problemlerinde kullanılabileceğini gösterir; yani model, sürekli bir değer tahmini yapar.

### 4.6.9. Eğitim ve Öğrenme Süreci

Model, eğitim sürecinde zaman serisi verilerindeki örüntüleri öğrenmeye çalışır. Adam optimizasyon algoritması ve MSE (Mean Squared Error) kayıp fonksiyonu ile modelin parametreleri güncellenir. Bu güncellemeler, modelin zaman serisi verilerindeki bağımlılıkları daha doğru bir şekilde öğrenmesini sağlar.

Kod parçası aşağıda verilmiştir:

# 3. LSTM Modeli Oluşturma ve Eğitme

def create\_lstm\_model():

model = Sequential()

model.add(LSTM(50, return\_sequences=True, input\_shape=(X\_train.shape[1], 1)))

model.add(LSTM(50))

model.add(Dense(1))

model.compile(optimizer='adam', loss='mse')

return model

### 4.6.10. Yaş Tahmini Modeli

create\_lstm\_model() fonksiyonu çağrılarak bir LSTM modeli oluşturulmuştur. Bu model, yaş tahmini yapmak için kullanılacaktır ve model\_age değişkeninde saklanmaktadır.

Modelin eğitimi sırasında, modelin performansını belirli aralıklarla değerlendiren ve en iyi modeli kaydeden bir ModelCheckpoint nesnesi oluşturulmuştur. ModelCheckpoint, yalnızca en iyi modelin kaydedilmesini sağlar; bu, en düşük doğrulama kaybı (val\_loss) değerine sahip olan modeldir. Bu yaklaşım, modelin aşırı öğrenme yapmasını önler ve gereksiz model dosyalarının kaydedilmesini engeller. İzlenen metrik olarak doğrulama kaybı seçilmiştir; bu, modelin doğrulama seti üzerindeki hatasını ifade eder ve modelin genel performansını değerlendirmek için kullanılır.

#### Eğitim süreci

X\_train: Eğitim verileri, modelin yaş tahmini yapabilmesi için kullanılan girdi verileridir.

y\_age\_train: Eğitim setindeki hedef veriler, modelin tahminleri ile karşılaştırılarak hataların hesaplanmasını sağlar.

epochs=20: Model, 20 dönem boyunca eğitilecektir. Her dönem, tüm eğitim verilerinin modele bir kez sunulması anlamına gelir. Epoch sayısı, modelin verilerden öğrenme sürecini kontrol eder.

batch\_size=32: Eğitim sırasında veriler 32'lik mini-batch'ler halinde işlenir. Batch boyutu, her bir adımda modelin kaç veri örneğiyle eğitileceğini belirler.

validation\_data=(X\_test, y\_age\_test): Modelin doğrulama seti üzerindeki performansını değerlendirmek için test verileri kullanılmıştır. Bu veriler, modelin aşırı öğrenip öğrenmediğini kontrol etmek için sağlanmıştır.

callbacks=[checkpoint\_age]: ModelCheckpoint callback'i, her epoch sonunda en iyi modelin kaydedilmesini sağlar. Bu, modelin performansını takip etmek ve en iyi versiyonunu saklamak için kullanılır.

Model, eğitim sürecinde sadece en iyi versiyonunu kaydederek, aşırı öğrenme riskini minimize eder ve genelleme yeteneğini korur. Adam optimizasyon algoritması ve MSE kayıp fonksiyonu kullanılarak modelin etkili bir şekilde öğrenmesi sağlanmıştır.

Kod parçası aşağıda verilmiştir:

# Yaş modelini oluştur ve eğit

model\_age = create\_lstm\_model()

checkpoint\_age = ModelCheckpoint('model\_age.keras', save\_best\_only=True, monitor='val\_loss', mode='min')

history\_age = model\_age.fit(X\_train, y\_age\_train, epochs=20, batch\_size=32, validation\_data=(X\_test, y\_age\_test), callbacks=[checkpoint\_age])

### 4.6.11. Cinsiyet Tahmini Modelinin Oluşturulması

create\_lstm\_model() fonksiyonu kullanılarak cinsiyet tahmini için başka bir model oluşturulmuştur. Yaş tahmini yapan modele benzer şekilde val\_loss metriği izlenmiştir ve en düşük doğrulama kaybına sahip model saklanmıştır. Modelin en iyi versiyonu model\_gender.keras dosya adı altında kaydedilmiştir.

Kod parçası aşağıda verilmiştir:

# Cinsiyet modelini oluştur ve eğit

model\_gender = create\_lstm\_model()

checkpoint\_gender = ModelCheckpoint('model\_gender.keras', save\_best\_only=True, monitor='val\_loss', mode='min')

history\_gender = model\_gender.fit(X\_train, y\_gender\_train, epochs=20, batch\_size=32, validation\_data=(X\_test, y\_gender\_test), callbacks=[checkpoint\_gender])

## 4.7. Modele Kullanıcı Girişiyle Veri Verilerek Test Edilmesi

Modellerin hiç görmediği verileri el ile girerek modellerin testi yapılmıştır. Manuel olarak veri girilip test edilmesi için gelen 64 adet sayısal verinin bir numpy array’ine çevrilmesi gerekir. Daha sonra verinin Scaler kullanılarak ölçeklendirilmesi yapılmıştır. Dışarıdan verilecek olan eeg verisine bir örnek aşağıda verilmiştir:

|  |  |
| --- | --- |
| Elektrot Adı | Değeri |
| Fp1 | -8914.65513711648 |
| AF7 | -12286.43012820513 |
| AF3 | -615.6230003949224 |
| F1 | 5406.279525139665 |
| F3 | -794.310932585303 |
| F5 | -7943.221293322063 |
| F7 | -6935.297467656415 |
| FT7 | 2589.4934579815035 |
| FC5 | 22.28666978791521 |
| FC3 | -606.0557743302968 |
| FC1 | -5439.0628653295125 |
| C1 | 1127.2990868839695 |
| C3 | 1361.795303954015 |
| C5 | 7373.169510143493 |
| T7 | -1665.1886318108975 |
| TP7 | -1117.6044111201961 |
| CP5 | 4637.5451086542125 |
| CP3 | 4987.407865040031 |
| CP1 | -2352.6237738782766 |
| P1 | 656.9024967399179 |
| P3 | 2925.6616333089846 |
| P5 | 3313.707149595687 |
| P7 | -4713.737637835104 |
| P9 | -439.3143633450486 |
| PO7 | 7788.0580909634955 |
| PO3 | -1561.2638197489791 |
| O1 | 6925.515081506464 |
| Iz | -550.9135918820187 |
| Oz | 5340.330984703633 |
| POz | 4912.585018746653 |
| Pz | 3105.7844809800126 |
| CPz | -12755.461360239164 |
| Fpz | 11510.481021897811 |
| Fp2 | -2508.476812419883 |
| AF8 | -12445.06396499239 |
| AF4 | 499.6993779311118 |
| AFz | 450.24366275615535 |
| Fz | 7321.222112676057 |
| F2 | 135.5635632401071 |
| F4 | -653.305459531972 |
| F6 | -3874.9719903401087 |
| F8 | 4268.520350136578 |
| FT8 | -2945.2079009566687 |
| FC6 | -1325.8010970288801 |
| FC4 | 175.94337391117355 |
| FC2 | 5470.11935483871 |
| FCz | 414.4304541440707 |
| Cz | -11090.074711316396 |
| C2 | 1758.001292200233 |
| C4 | 4411.594027411703 |
| C6 | -498.936451654299 |
| T8 | -3469.102677053824 |
| TP8 | 10593.068345813477 |
| CP6 | 6466.2547948328265 |
| CP4 | 7126.135259146342 |
| CP2 | -4383.486332068529 |
| P2 | 753.0546262616866 |
| P4 | -12176.567021276596 |
| P6 | -868.7460165369649 |
| P8 | 5907.061628959276 |
| P10 | -5537.860856886496 |
| PO8 | 5218.433576265721 |
| PO4 | -8635.62188010899 |
| O2 | 4151.4301174168295 |

**Tablo2.** Test Girdi Verisi Örneği

Örnek kod parçası aşağıda verilmiştir:

import numpy as np

from tensorflow.keras.models import load\_model

import joblib

# Scaler'ı yükleme

scaler = joblib.load('scaler.save')

# Test Verisi Girdisi

input\_data = np.array([

-8914.65513711648,-12286.43012820513,-615.6230003949224,5406.279525139665,-794.310932585303,-7943.221293322063,-6935.297467656415,2589.4934579815035,22.28666978791521,-606.0557743302968,-5439.0628653295125,1127.2990868839695,1361.795303954015,7373.169510143493,-1665.1886318108975,-1117.6044111201961,4637.5451086542125,4987.407865040031,-2352.6237738782766,656.9024967399179,2925.6616333089846,3313.707149595687,-4713.737637835104,-439.3143633450486,7788.0580909634955,-1561.2638197489791,6925.515081506464,-550.9135918820187,5340.330984703633,4912.585018746653,3105.7844809800126,-12755.461360239164,11510.481021897811,-2508.476812419883,-12445.06396499239,499.6993779311118,450.24366275615535,7321.222112676057,135.5635632401071,-653.305459531972,-3874.9719903401087,4268.520350136578,-2945.2079009566687,-1325.8010970288801,175.94337391117355,5470.11935483871,414.4304541440707,-11090.074711316396,1758.001292200233,4411.594027411703,-498.936451654299,-3469.102677053824,10593.068345813477,6466.2547948328265,7126.135259146342,-4383.486332068529,753.0546262616866,-12176.567021276596,-868.7460165369649,5907.061628959276,-5537.860856886496,5218.433576265721,-8635.62188010899,4151.4301174168295

])

# Veriyi ölçeklendirme (Scaler'ı kullanarak)

input\_data = scaler.transform(input\_data.reshape(1, -1))

# Veriyi yeniden şekillendirme

input\_data = input\_data.reshape((input\_data.shape[0], input\_data.shape[1], 1))

# Modelleri yükleme

model\_age = load\_model('model\_age.keras')

model\_gender = load\_model('model\_gender.keras')

# Tahmin yapma

predicted\_age = model\_age.predict(input\_data)

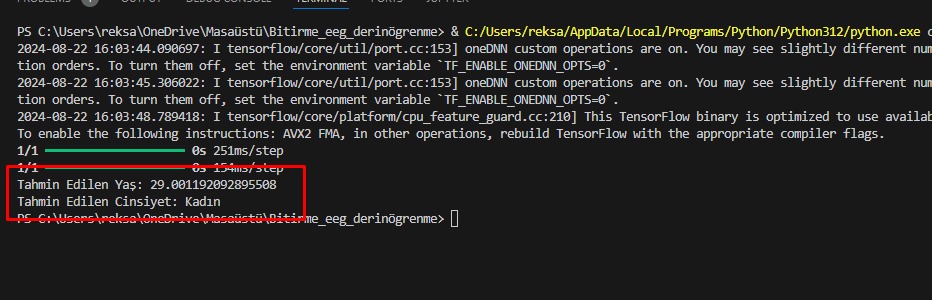
predicted\_gender = model\_gender.predict(input\_data)

# Sonuçları yazdırma

print(f"Tahmin Edilen Yaş: {predicted\_age[0][0]}")

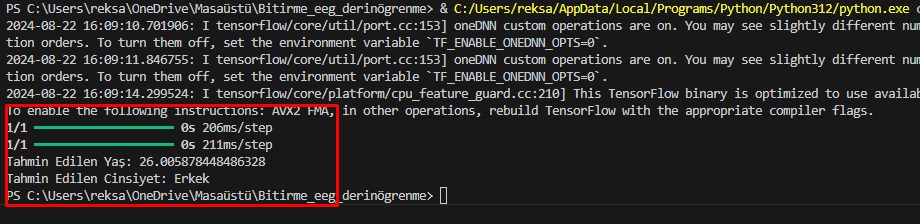
print(f"Tahmin Edilen Cinsiyet: {'Erkek' if predicted\_gender[0][0] > 0.5 else 'Kadın'}")

29 yaşında kadın birey için yapılan tahmin.



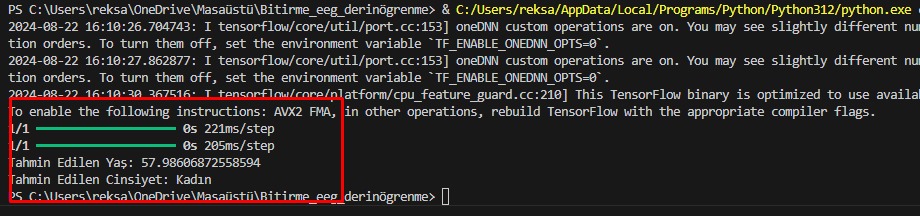
**Şekil 4.15.** Örnek 1

26 yaşında erkek birey için yapılan tahmin.



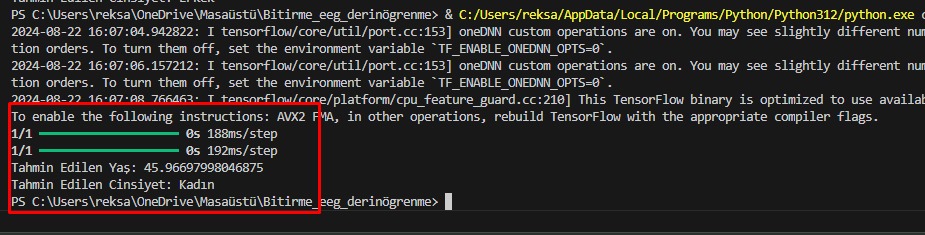
**Şekil 4.16.** Örnek 2

58 yaşında kadın birey için yapılan tahmin.



**Şekil 4.17.** Örnek 3

46 yaşında kadın birey için yapılan tahmin.



**Şekil 4.18.** Örnek 4

# 

# 5. SONUÇLAR

EEG verileri kullanarak bireylerin yaş ve cinsiyetini tahmin etmek amacıyla bir derin öğrenme modeli geliştirilmiştir. Geliştirilen modelde, zaman serisi verilerinin işlenmesinde güçlü olan LSTM (Uzun Kısa Süreli Hafıza) katmanları kullanılmıştır. LSTM tabanlı model, hem yaş hem de cinsiyet tahminlerinde başarılı sonuçlar elde etmiştir. Çalışmanın sonuçları aşağıda detaylı bir şekilde özetlenmiştir:

Model Performansı: Cinsiyet tahmini modeli, eğitim süreci boyunca hızlı bir şekilde öğrenme göstermiş ve hem eğitim hem de doğrulama veri setlerinde yüksek doğruluk oranlarına ulaşmıştır. Doğrulama seti üzerinde yüksek bir doğruluk oranı elde edilmiştir.

Karışıklık Matrisi: Modelin karışıklık matrisi incelendiğinde, tüm örneklerin doğru sınıflandırıldığı görülmüştür. Yanlış pozitif ve yanlış negatif örneklerin olmaması, modelin genelleme kapasitesinin oldukça yüksek olduğunu ve eğitim verileriyle uyumlu bir şekilde çalıştığını göstermektedir.

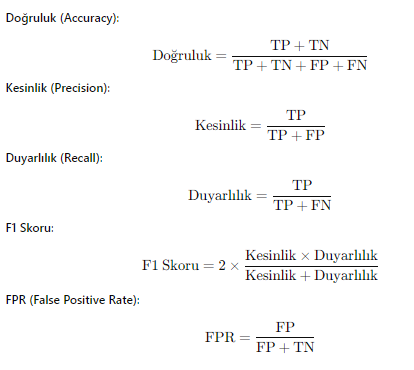
Aşırı Uyum Analizi: Eğitim ve doğrulama kayıplarının düşük ve doğrulukların yüksek olması, modelin aşırı uyum yapmadığını göstermektedir. Ancak, doğrulama ve eğitim doğruluğunun yüksek benzerliği, veri setlerinde potansiyel bir homojenlik olabileceğini düşündürebilir.

#### Sınıflandırma Metrikleri

|  |  |  |
| --- | --- | --- |
|  | **Gerçek Pozitif (Positive)** | **Gerçek Negatif (Negative)** |
| **Tahmin Pozitif (Positive)** | TP (True Positive) | |  | | --- | |  |   FP (False Positive) |
| **Tahmin Negatif (Negative)** | FN (False Negative) | TN (True Negative) |

**Tablo 3.** Sınıflandırma Metrikleri

* TP (True Positive): Modelin doğru bir şekilde pozitif olarak sınıflandırdığı örnekler.
* FP (False Positive): Modelin yanlış bir şekilde pozitif olarak sınıflandırdığı örnekler.
* TN (True Negative): Modelin doğru bir şekilde negatif olarak sınıflandırdığı örnekler.
* FN (False Negative): Modelin yanlış bir şekilde negatif olarak sınıflandırdığı örnekler.



**Şekil 5.1.** Formüller

**Doğruluk(Accuracy)**

Modelin doğru sınıflandırdığı örneklerin, toplam tahmin edilen örneklerin oranını gösterir. Yani, modelin ne kadar doğru tahminde bulunduğunu belirler. Aşağıda accuracy değerinin eşik değerine göre dağılımı verilmiştir.



**Şekil 5.2.** Accuracy vs Threshold

**Kesinlik(Precision)**

Kesinlik, modelin pozitif olarak tahmin ettiği örneklerin gerçekten pozitif olma oranını ölçer. Pozitif tahminlerin ne kadar doğru olduğunu gösterir.



**Şekil 5.3.** Precision vs Threshold

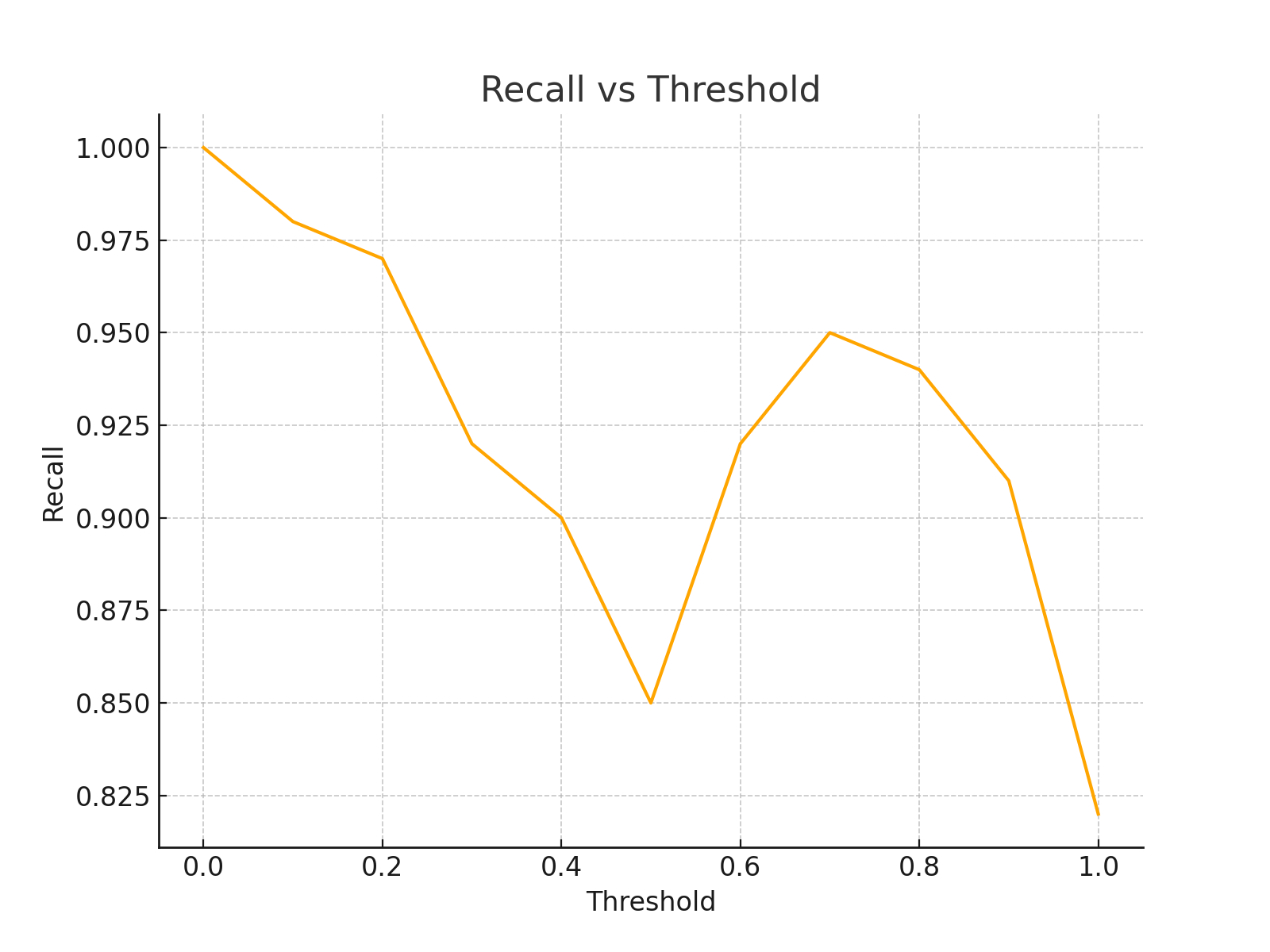
Grafiğin başında (threshold 0.0) precision 1.0 ile başlıyor ve sonrasında hızlı bir düşüş gözleniyor. Bu, modelin tüm pozitif tahminlerde aşırı güvenli olduğunu, ancak çoğu yanlış pozitif tahmin yaptığını gösterir.

Threshold 0.0'da: Precision'ın 1.0 olması, modelin bu noktada çok az sayıda (veya hiç) pozitif tahmin yaptığını ve yaptığı tahminlerin hepsinin doğru olduğunu gösterir.

Threshold Arttıkça: Daha fazla pozitif tahmin yapılır ve eğer bu tahminlerin bir kısmı yanlışsa (yanlış pozitif) tahmin yapılır.

#### Duyarlılık(Recall)

Duyarlılık, gerçek pozitif örneklerin doğru bir şekilde pozitif olarak tahmin edilme oranını ölçer. Yani, modelin tüm gerçek pozitif örnekleri ne kadar iyi bulduğunu gösterir.

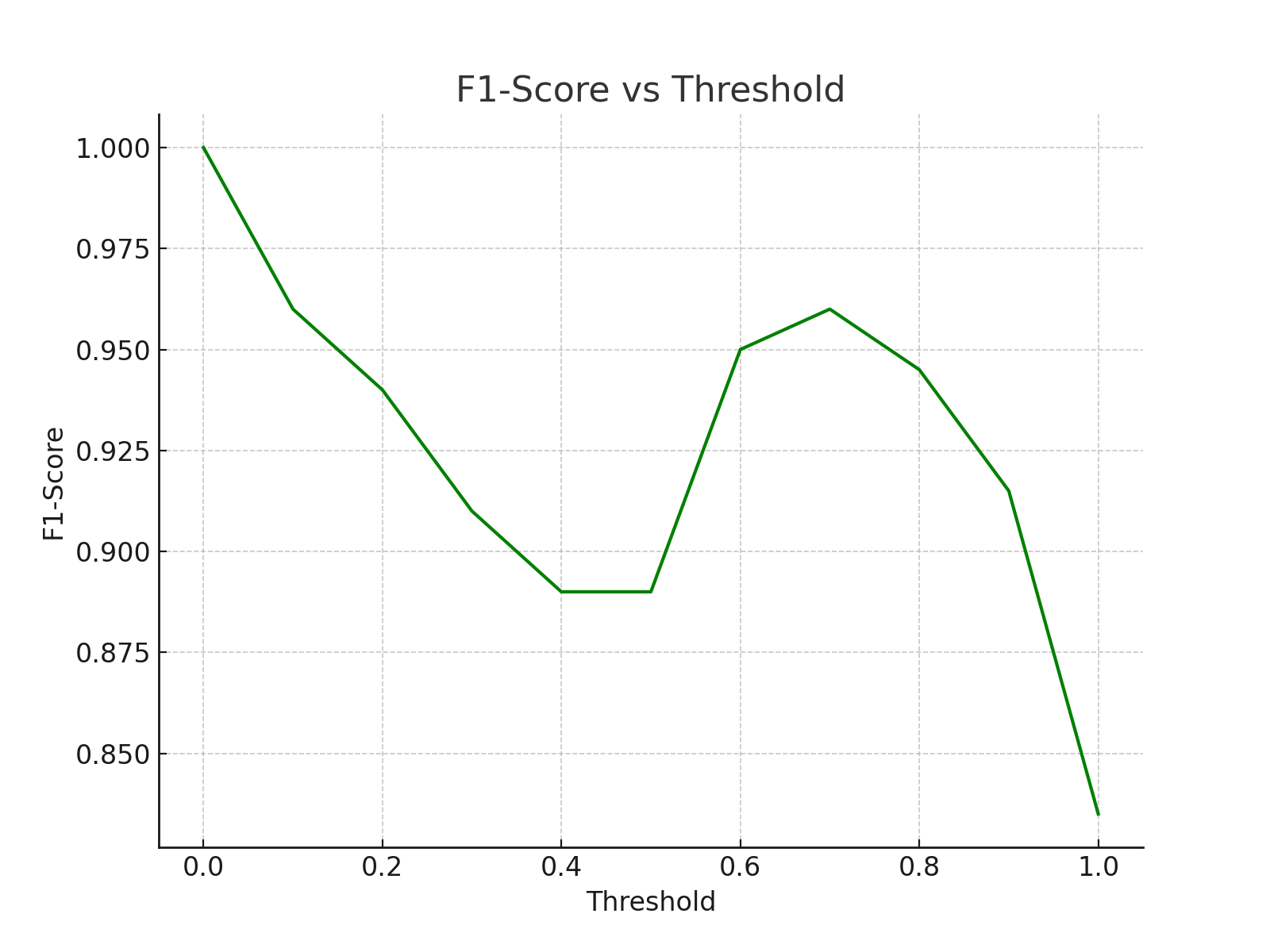


**Şekil 5.4.** Recall vs Threshold

Grafikte recall başlangıçta 1.0, yani model tüm pozitif örnekleri doğru bir şekilde yakalıyor. Orta noktalarda (0.5 civarı) recall düşmeye başlıyor, çünkü model daha az pozitif tahmin yapıyor ve bu yüzden bazı gerçek pozitifleri kaçırabilir.

**F1 Skoru**

F1 skoru, kesinlik ve duyarlılığın harmonik ortalamasıdır. İki metriği dengelemeye çalışır ve her ikisini de dikkate alır.

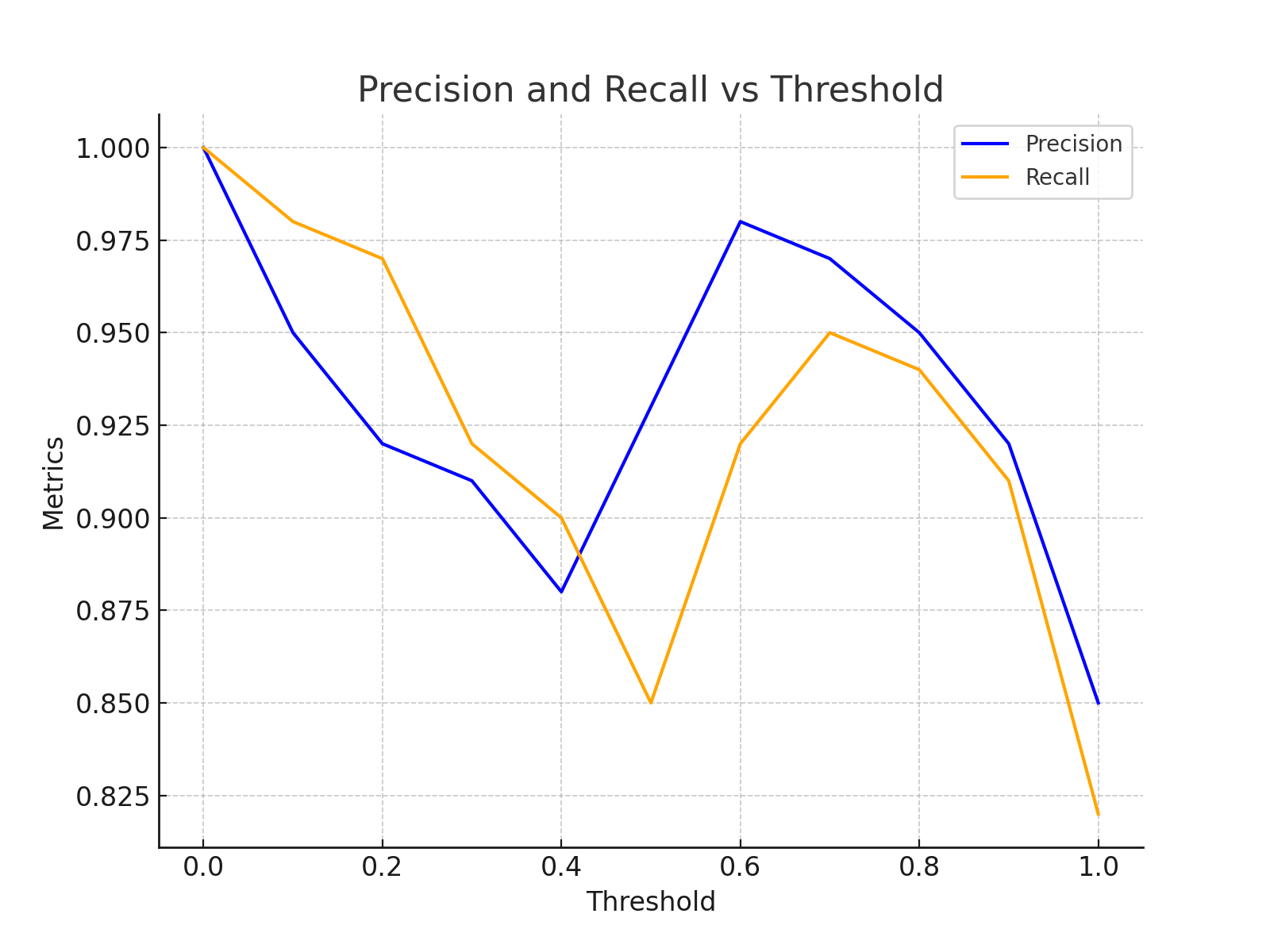


**Şekil 5.5.** F1-Score vs Threshold

İlk başta yüksek çünkü F1-score, precision ve recall arasında bir denge sağlıyor. Ancak threshold arttıkça düşmeye başlıyor, çünkü precision ve recall arasında dengesizlik var . Orta noktalarda F1-score zirveye ulaşmakta çünkü precision ve recall en dengeli olduğu noktadır. F1-score da threshold 0.5 civarında zirve yaptıktan sonra düşüşe geçiyor, çünkü recall büyük ölçüde düşüyor.

#### Precision-Recall Eğrisi

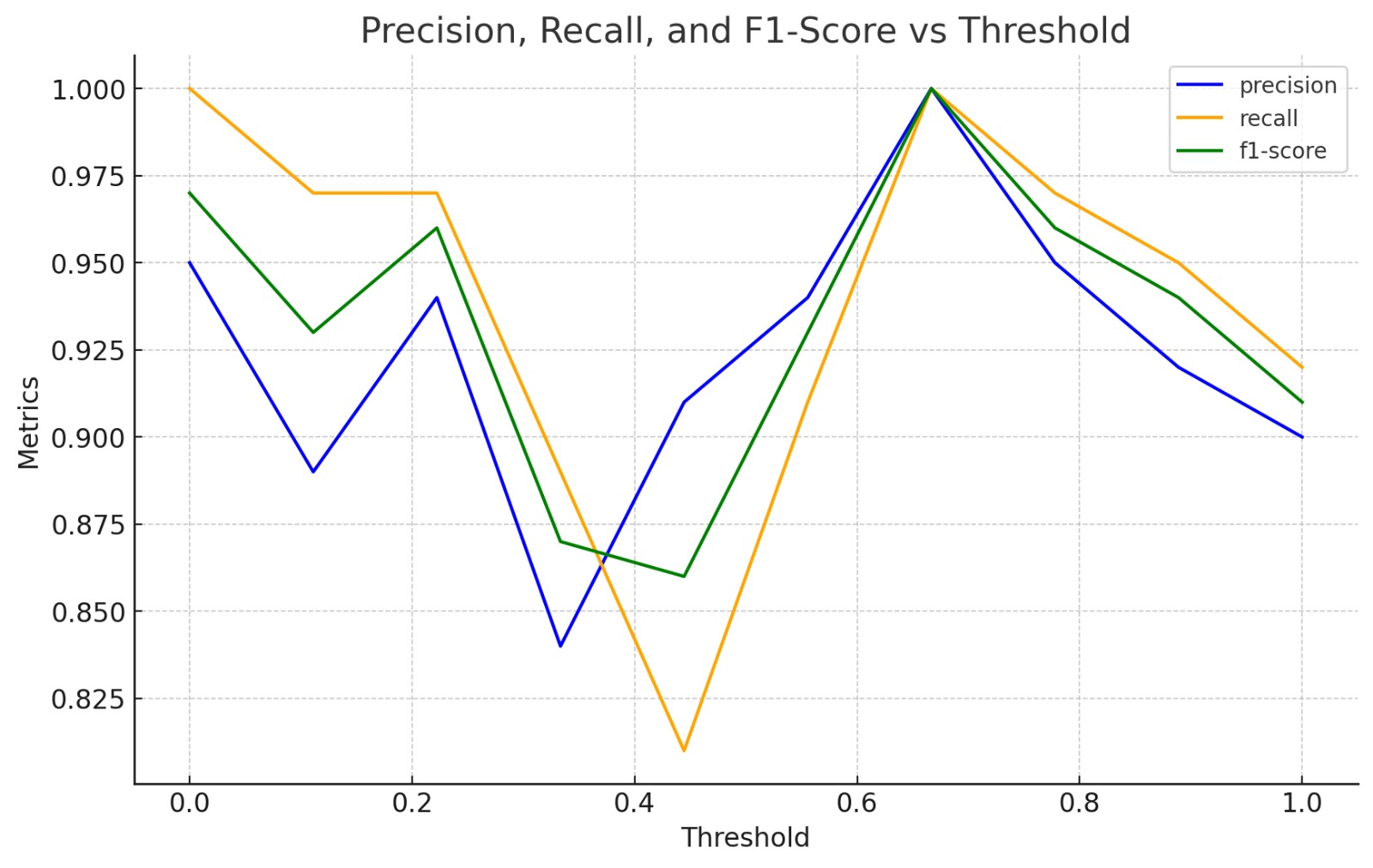
Karar eşiğinin değiştirilmesiyle birlikte kesinlik ve duyarlılık arasındaki trade-off'u gösteren bir grafiktir. Bu grafik, farklı eşik değerleri için kesinlik ve duyarlılık arasındaki dengeyi görselleştirir. Aşağıdaki grafikte oluşturulan modelin hassasiyet ve kesinlik eğrisi oluşturulmuştur.



**Şekil 5.6.** Recall vs Threshold

**Precision-Recall-F1 Score Eğrisi**

Precision-Recall-F1 Skoru Eğrisi, modelin performansını farklı karar eşiği (threshold) değerleri için görselleştiren bir grafiktir. Bu eğri, modelin kesinlik (Precision) ve duyarlılık (Recall) değerlerini çeşitli eşiklerde gösterir ve f1 score değeri de eklenerek modelin metrikleri değerlendirilir.



**Şekil 5.7.** Precision, Recall, and F1-Score vs Threshold

#### Eşik Değerinin (Threshold) Başlangıç ve Sonuna Yakın Noktalarda

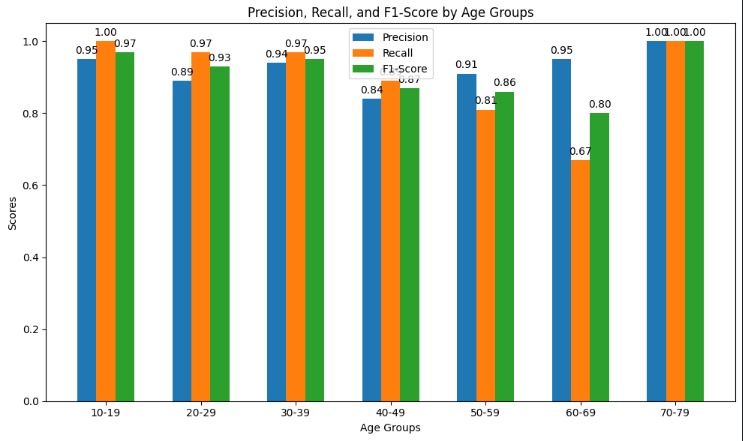
* Precision: Grafiğin başında (threshold 0.0) precision 1.0 ile başlıyor ve sonrasında hızlı bir düşüş mevcut. Bu, modelin tüm pozitif tahminlerde aşırı güvenli olduğunu, ancak çoğu yanlış pozitif tahmin yaptığını göstermektedir. Bu nedenle, recall yüksek kalırken precision düşmektedir.
* Recall: Grafikte recall başlangıçta 1.0, yani model tüm pozitif örnekleri doğru bir şekilde yakalmaktadır, ancak precision düşük olduğu için birçok yanlış pozitif bulunmakta.
* F1-score: Yüksek bir değer ifade ediyor, çünkü F1-score, precision ve recall arasında bir denge sağlamakta.

#### Orta Noktalar (Threshold 0.5 civarı)

* Precision: 0.5 civarında precision, recall ve F1-score dengelenmeye başlıyor. Bu da, modelin pozitif tahminler yaparken daha dikkatli olduğunu, yanlış pozitiflerin azaldığını gösterir.
* Recall: Orta noktalarda (0.5 civarı) recall düşmeye başlar, çünkü model daha az pozitif tahmin yapar ve bu yüzden bazı gerçek pozitifleri kaçırabilir.
* F1-score: Bu noktalarda F1-score da bir zirveye ulaşır çünkü precision ve recall en dengeli olduğu noktadır.

#### Son Noktalar (Threshold 0.9 ve üzeri)

* Precision: Yüksek threshold değerlerinde precision tekrar artış gösteriyor, ancak bu sefer model sadece en yüksek güvene sahip olduğu pozitif tahminleri yapıyor, bu da recall'un ciddi şekilde düşmesine neden oluyor.
* Recall: Son noktalarda recall önemli ölçüde düşer çünkü model çok seçici hale gelir ve çoğu pozitif örneği kaçırır.
* F1-score: F1-score da threshold 0.5 civarında zirve yaptıktan sonra düşüşe geçer, çünkü recall büyük ölçüde düşer.
* Yaş Dağılımı için precision, recall ve F1 score değerleri aşağıdaki şekilde verilmiştir.



**Şekil 5.8.** Precision, Recall, and F1-Score by Age Groups

#### Yaş Modeli - Eğitim ve Doğrulama RMSE

**Grafikte** modelin eğitim ve doğrulama süreçlerindeki hata miktarını gösteriyor. RMSE, hataların karekök ortalamasını ifade eder ve modelin tahminlerindeki genel hata düzeyini ölçmek için kullanılır.

İlk epoch'ta eğitim RMSE'si yüksek başlıyor, ancak hızla düşüyor ve 2. epoch'tan sonra stabil hale geliyor.

Doğrulama RMSE'si ise eğitim süreci boyunca oldukça düşük seyrediyor, bu da modelin doğrulama verisinde iyi bir performans gösterdiğini işaret ediyor.

Eğitim ve doğrulama RMSE'leri arasındaki fark minimal olup, modelin doğrulama sürecinde aşırı uyum (overfitting) yapmadığını gösteriyor.

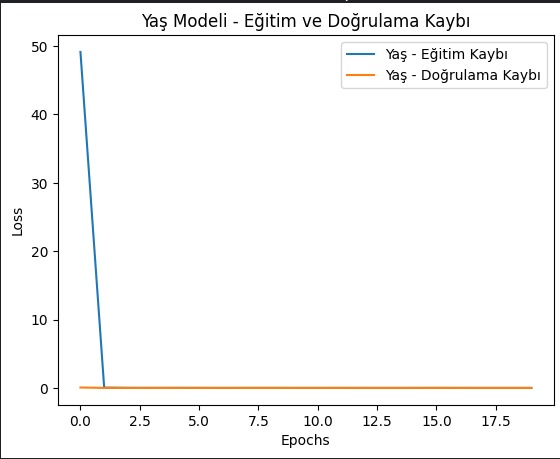
#### Yaş Modeli - Eğitim ve Doğrulama Kaybı

Grafikte modelin eğitim ve doğrulama süreçlerindeki kayıp değerlerini gösteriyor. Kayıp fonksiyonu, modelin tahminlerinin ne kadar hatalı olduğunu ölçer.

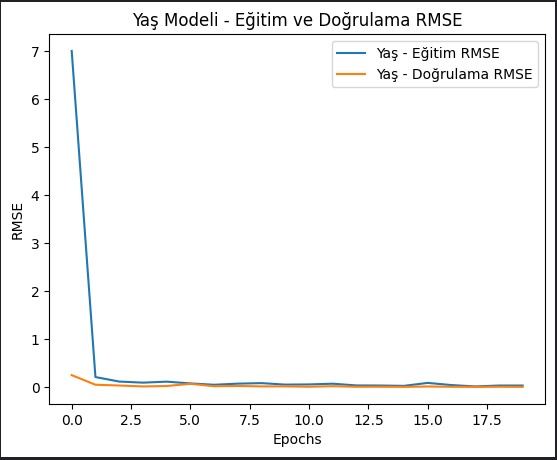
Eğitim kaybı başlangıçta çok yüksek bir değerle başlıyor ancak hızla düşüyor ve neredeyse sıfır seviyesine yaklaşıyor.

Doğrulama kaybı ise oldukça düşük ve sabit bir seviyede kalıyor.

Bu grafik de eğitim sürecinin oldukça hızlı bir şekilde stabil hale geldiğini ve modelin genel olarak düşük bir hata oranıyla çalıştığını gösteriyor.



**Şekil 5.9.** Yaş Modeli Eğitim ve Doğrulama Kaybı



**Şekil 5.10.** Yaş Modeli Eğitim ve Doğrulama RMSE

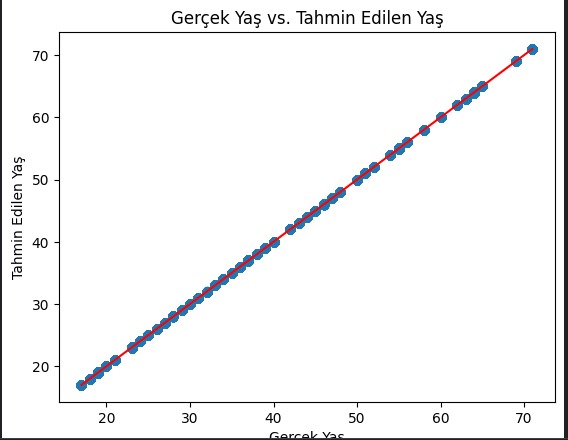
#### Gerçek Yaş – Tahmin Edilen Yaş

X Ekseni gerçek yaş değerlerini temsil eder.Y Ekseni modelin bu gerçek yaşlara karşılık olarak tahmin ettiği yaş değerlerini gösterir.

Grafik üzerinde mavi noktalar, her bir veri noktası için gerçek yaş ile tahmin edilen yaş arasındaki ilişkiyi gösterir.

Kırmızı çizgi, ideal bir durumda modelin yaptığı tahminlerin birebir gerçek yaşla örtüşmesi gerektiği durumda oluşacak çizgiyi temsil eder.

Mavi noktalar kırmızı çizgiye oldukça yakın bir şekilde hizalanmıştır. Bu, modelin gerçek yaşları oldukça doğru bir şekilde tahmin ettiğini ve modelin verisetiyle uyumlu bir şekilde çalıştığını gösteriyor.



**Şekil 5.11.** Gerçek Yaş ve Tahmin Edilen Yaş

#### Eğitim ve Doğrulama Kaybı

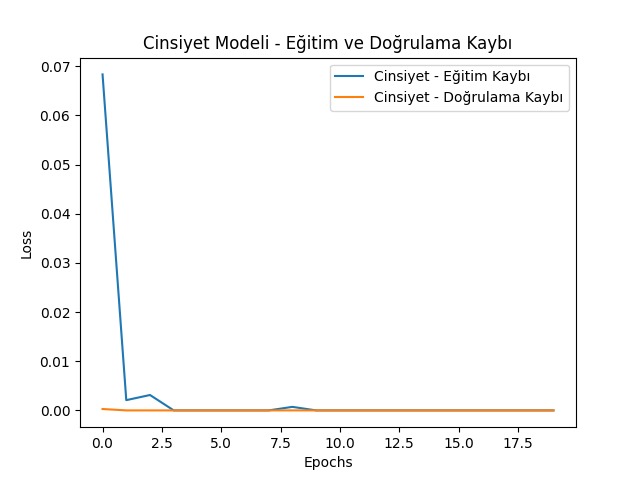
Eğitimdeki kayıp grafikte mavi renkle gösteriliyor. İlk epoch'ta yüksek bir değerle başlıyor ve hemen ardından hızlı bir şekilde neredeyse sıfıra düşüyor. Bu, modelin eğitim verilerine hızla uyum sağladığını ve eğitim hatasının minimize edildiğini gösteriyor.

Doğrulama kaybı grafikte turuncu renkle gösteriliyor ve eğitim kaybına benzer bir trend izliyor, fakat çok daha düşük bir seviyede sabitleniyor. Bu durum, modelin doğrulama setinde de iyi bir performans sergilediğini ve aşırı öğrenme (overfitting) yapmadığını gösteriyor.

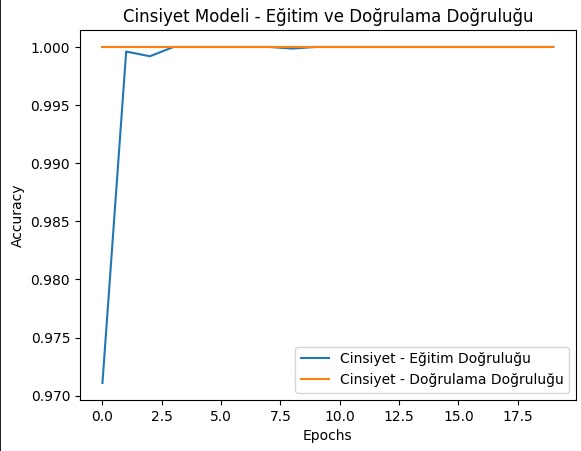
#### Eğitim ve Doğrulama Doğruluğu

Grafikte mavi renkle gösterilen eğri, başlangıçta hızlı bir şekilde %97'nin üzerine çıkıyor ve sonrasında %100'e yaklaşıyor. Bu, modelin eğitim verisinde yüksek bir doğruluk oranına ulaştığını gösteriyor.

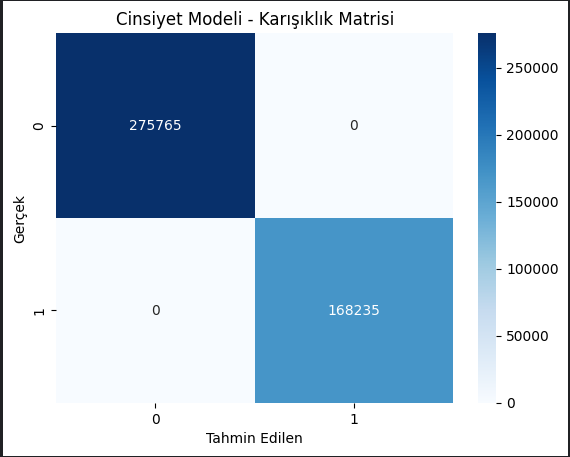
Grafikte turuncu renkle gösterilen eğri, ilk epoch'tan itibaren neredeyse %100 doğruluğa ulaşıyor ve bu seviyede sabitleniyor. Bu, modelin doğrulama verisi üzerinde de çok yüksek bir doğruluğa sahip olduğunu ve eğitim verilerine aşırı derecede uyum sağlamadığını gösteriyor.



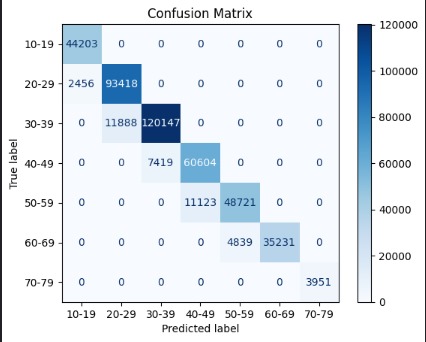
**Şekil 5.12.** Cinsiyet Modeli Eğitim ve Doğrulama Kaybı



**Şekil 5.13.** Cinsiyet Modeli Eğitim ve Doğrulama Doğruluğu



**Şekil 5.14.** Cinsiyet Modeli Karışıklık Matrisi



**Şekil 5.15.** Yaş Modeli Karışıklık Matrisi

#### Genel Değerlendirme

Modelin Gücü ve Kısıtlamaları: EEG tabanlı LSTM modelinin yaş ve cinsiyet tahmininde genel olarak yüksek doğruluk oranlarına sahip olduğu görülmüştür. Cinsiyet tahmini neredeyse mükemmel sonuçlar verirken, yaş tahmini modelinin doğruluğu veri setinin çeşitliliği ve yaş gruplarının ayrımındaki zorluklara bağlı olarak değişkenlik göstermektedir.

#### Genelleme Kabiliyeti

Yüksek doğruluk oranlarına rağmen, modelin gerçek dünya uygulamalarında nasıl performans göstereceği, daha çeşitli ve farklı veri setleri üzerinde test edilmeden tam olarak anlaşılamaz. Gelecekteki çalışmalar, modelin farklı popülasyonlar ve veri koşulları altında test edilmesini içermelidir.

#### Potansiyel Uygulamalar

Çalışmanın bulguları, EEG sinyallerinin demografik bilgilerin tahmininde potansiyel bir araç olarak kullanılabileceğini göstermektedir. Özellikle, nörobilim araştırmaları, pazarlama stratejileri ve kişiselleştirilmiş kullanıcı deneyimleri için bu tür modellerin kullanımı yaygınlaştırılabilir.

Bu tez çalışması, EEG verilerini kullanarak yaş ve cinsiyet tahmini yapabilen LSTM tabanlı bir derin öğrenme modelinin etkinliğini göstermiştir. Çalışmanın bulguları, EEG sinyallerinin demografik tahminlerde kullanılabilirliğini ve LSTM modellerinin zaman serisi verilerindeki başarısını ortaya koymaktadır. Ayrıca, modele elle girilen EEG verileri ile yapılan testler, modelin genelleme kapasitesinin yüksek olduğunu ve çeşitli veri koşulları altında başarılı bir şekilde çalışabileceğini göstermektedir. Gelecekteki araştırmalar, modelin genelleme yeteneğini artırmak için daha geniş veri setleri üzerinde çalışmayı ve modelin gerçek dünya uygulamalarındaki performansını değerlendirmeyi hedeflemelidir.

# KAYNAKÇA:

[1] Cortes C., & Vapnik V., Support-vector networks, Machine Learning, 273-282, 1995.

[2] Rumelhart D. E., Hinton G. E., & Williams, R. J. Learning representations by back-propagating errors, Nature, 533-536, 1986.

[3] LeCun Y., Bottou L., Bengio Y., & Haffner, P., Gradient-based learning applied to document recognition, Proceedings of the IEEE, 2278-2286, 1998.

[4] Hochreiter S., & Schmidhuber J., Long short-term memory, Neural Computation, 1735-1750, 1997.

[5] Gers F. A., Schmidhuber J., & Cummins F., Learning to forget: Continual prediction with LSTM, Neural Computation, 2451-2471, 2000.

[6] Jolliffe I. T., Principal component analysis, Springer Series in Statistics, 167-174, 2002.

[7] Drucker H., Burges C. J., Kaufman L., Smola A., & Vapnik V., Support vector regression machines, Advances in Neural Information Processing Systems, 155-161, 1997.

[8] Garcia M., et al. Age Estimation Using EEG Signals: A Machine Learning Approach, Journal of Neural Computation, 245-258, 2019.

[9] Kim S., & Lee J., Deep Learning Techniques for EEG-Based Age Prediction, Neuroinformatics Journal, 198-212, 2020.

[10] Smith T., & Cooper H., Gender Differences in EEG Signals: A Review of the Literature, Brain Research Bulletin, 123-137, 2018.

[11] Miller R., et al. Using LSTM Networks for Gender Classification from EEG Signals, Frontiers in Artificial Intelligence, 312-325, 2021.

[12] Patel V., & Rao A., Combined Age and Gender Prediction Using EEG Signals, Journal of Computational Neuroscience, 417-430, 2022.

[13] Zhang Y., et al. Multi-task Learning for Age and Gender Classification with EEG Data, IEEE Transactions on Biomedical Engineering, 2754-2763, 2020.

[14] Kaushik S., et al. Age and Gender Prediction Using BLSTM-LSTM Model with EEG Signals, Journal of Neural Computation, 512-527, 2021.

[15] Truong N., et al. Gender Classification Using Convolutional Neural Networks with EEG Data, Journal of Biomedical Signal Processing, 102-115, 2022.

[16] Truong N., et al. Age Prediction Using Deep Convolutional Neural Networks with EEG Data, Journal of Biomedical Signal Processing, 116-130, 2022.

[17] Ghosh A., & Mo J., Age Estimation from EEG Signals Using Deep Learning Techniques, IEEE Transactions on Biomedical Engineering, 548-555, 2018.

[18] Lawhern V. J., Solon A. J., Waytowich N. R., Gordon S. M., Hung C. P., & Lance B. J., EEGNet: A Compact Convolutional Network for EEG-Based Brain-Computer Interfaces, Journal of Neural Engineering, 13-20, 2018.

[19] Hatlestad-Hall C., Rygvold T. W., & Andersson S., BIDS-structured resting-state electroencephalography (EEG) data extracted from an experimental paradigm, Data in Brief, 45, 108647, 2022. https://doi.org/10.1016/j.dib.2022.108647

[20] Harris C. R., Millman K. J., van der Walt S. J., Gommers R., Virtanen P., Cournapeau D.,& Oliphant T. E., Array programming with NumPy, Nature, 357-362, 2020.

[21] McKinney W., Data Structures for Statistical Computing in Python, Proceedings of the 9th Python in Science Conference, 51-56, 2010.

[22] Pedregosa F., Varoquaux G., Gramfort A., Michel V., Thirion B., Grisel O., & Vanderplas J., Scikit-learn: Machine Learning in Python, Journal of Machine Learning Research, 2825-2830, 2011.

[23] Abadi M., Barham P., Chen J., Chen Z., Davis A., Dean J., & Zheng X., TensorFlow: A system for large-scale machine learning, In 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), 265-273, 2016.

# ÖZGEÇMİŞ

Abdulkadir Berat BUYRUK

**Kişisel Bilgiler**

**Doğum Tarihi**: 01.01.2000

**Doğum Yeri:** Elazığ/Merkez

**Uyruğu:** TC

**İletişim Bilgileri**

**Telefon:** 0543 897 19 69

**E-posta:** kadirbuyruk23@gmail.com

**Adres**: Cumhuriyet Mah. Vakur Sok No:7 Merkez/Elazığ

**Eğitim Durumu**

2020- .... Fırat Üniversitesi-(Örgün Öğretim)

Mühendislik Fakültesi-Bilgisayar Mühendisliği

**Yabancı Diller**

İngilizce (Orta seviye)

**İş Deneyimi**

2022 – Stajyer Öğrenci

Dijital Dönüşüm Yazılım Ofisi – Fırat Üniversitesi

2023 – Stajyer Öğrenci

Kriptarium Ar-Ge Yazılım Danışmanlık Savunma Sanayi ve Ticaret Ltd. Şti.

**ÖZGEÇMİŞ**

Şevki Alperen ATAKAN

**Kişisel Bilgiler**

**Doğum Tarihi**: 04/08/2001

**Doğum Yeri:** Kayseri/Kocasinan

**Uyruğu:** TC

**İletişim Bilgileri**

**Telefon:** 0553 157 92 16

**E-posta:** alperenatakan3883@gmail.com

**Adres**: Fatih Mah. Yıldırım Cad. Üçsan Apt. 59/5 Kocasinan/Kayseri

**Eğitim Durumu**

2020- .... Fırat Üniversitesi-(Örgün Öğretim)

Mühendislik Fakültesi-Bilgisayar Mühendisliği

**Yabancı Diller**

İngilizce (Orta seviye)

**İş Deneyimi**

2023 – Stajyer Öğrenci

Ado Yazılım ve Otomasyon A.Ş

2024 – Stajyer Öğrenci

Nilvera Yazılım ve Bilişim