

**T.C.**  
**PAMUKKALE ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**  
**BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI**

**ELECTROENCEPHALOGRAPHY (EEG) SİNYAL**  
**SINIFLANDIRILMASINDA SİNİR-EVRİMİ YAKLAŞIMI**

**YÜKSEK LİSANS TEZİ**

**ERDEM AYBEK**

**DENİZLİ, KASIM - 2024**

**T.C.**  
**PAMUKKALE ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**  
**BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI**



**ELECTROENCEPHALOGRAPHY (EEG) SİNYAL**  
**SINIFLANDIRILMASINDA SİNİR-EVRİMİ YAKLAŞIMI**

**YÜKSEK LİSANS TEZİ**

**ERDEM AYBEK**

**DENİZLİ, KASIM - 2024**

**Bu tezin tasarımı, hazırlanması, yürütülmesi, araştırmalarının yapılması ve bulgularının analizlerinde bilimsel etiğe ve akademik kurallara özenle riayet edildiğini; bu çalışmanın doğrudan birincil ürünü olmayan bulguların, verilerin ve materyallerin bilimsel etiğe uygun olarak kaynak gösterildiğini ve alıntı yapılan çalışmalara atfedildiğine beyan ederim.**

**ERDEM AYBEK**

**İMZA**

# ÖZET

## ELECTROENCEPHALOGRAPHY (EEG) SİNYAL SINIFLANDIRILMASINDA SİNİR-EVRİMİ YAKLAŞIMI YÜKSEK LİSANS TEZİ

ERDEM AYBEK

PAMUKKALE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ  
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI  
(TEZ DANIŞMANI: DOÇ. DR. MERİÇ ÇETİN)

DENİZLİ, KASIM - 2024

Elektroensefalografi (EEG), beyindeki nöronların elektriksel uyarılarını ölçmede ve bu sinyalleri kaydetmede kullanılan bir yöntemdir. Bu sinyaller beyin fonksiyonlarının incelenmesi, epilepsi gibi nörolojik rahatsızlıkların teşhisi, nöropsikolojik rahatsızlıkların sınıflandırılması ve uyku bozukluklarının değerlendirilmesi gibi alanlarda kullanılır. Sağlık alanında sıkça kullanılmakta olan EEG sinyalleri oldukça karmaşık ve çok kaynaklı elektrik sinyalleridir. Anlamalı bir sınıflandırma çabası insan emeğini azalttığı gibi insan hatasını da azaltarak sağlık alanında avantaj sağlar ve iş gücünün verimli kullanımına destek olur. Klinik karar destek sistemleri ile doğru bir şekilde sınıflandırılan EEG sinyalleri, doktorların daha hızlı ve doğru teşhis koymalarına yardımcı olur, bu da uygulamaların doğruluğunu ve güvenilirliğini artırır. Bu çalışmada, epilepsi nöbetlerine ait sinyalleri de içeren CHB-MIT Scalp EEG veri seti kullanılarak EEG sinyallerinin sınıflandırılması ve epilepsi nöbeti tespiti gerçekleştirilmiştir. Çeşitli veri ön işleme yöntemleri kullanılarak veri seti üzerinde varyasyonlar oluşturulmuş ve benzetim çalışmaları iki senaryo üzerinden ele alınmıştır. Bu tezde, EEG sinyallerini sınıflandırma amacıyla, genellikle derin öğrenme algoritmaları gibi geleneksel yöntemlerle optimize edilmesi zor olan karmaşık yapay sinir ağlarını optimize etmek için kullanılan sinir evrimi yaklaşımı tercih edilmiştir. Bu alandaki en popüler yaklaşımlardan biri olan NEAT (Neuro Evolution of Augmented Topologies) algoritması ile elde edilen benzetim sonuçları diğer popüler derin öğrenme modellerinden olan evrimsel sinir ağları (Convolutional Neural Network-CNN) sonuçları ile karşılaştırılmıştır. Sonuçlar tam anlamıyla iki modelin performans karşılaştırmaları şeklinde düşünülmemelidir. Kullanılan her iki yapay zeka yöntemi farklı veri ön işleme ve modelleme yaklaşımları açısından değerlendirilmiştir. Elde edilen sonuçlarda, sinir evrimi yaklaşımının CNN modeline kıyasla daha düşük değerlendirme metriklerine sahip olduğu gözlenmiştir. Literatürde ilk kez karmaşık ve çok kaynaklı EEG sinyallerinin sınıflandırılmasında kullanılan sinir evrimi yaklaşımının, en az CNN modelleri kadar başarılı olduğu ancak, daha yoğun hiperparametre optimizasyonu çalışmaları ile daha yüksek başarımlarına çıkarılabileceği söylenebilir.

**ANAHTAR KELİMELER:** Electroensefalografi, Sinir Evrimi, Evrimsel Sinir Ağları, Epilepsi, Epileptik Nöbet Saptama, Yapay Zeka

## **ABSTRACT**

### **NEUROEVOLUTIONARY APPROACH TO ELECTROENCEPHALOGRAPHY (EEG) SIGNAL CLASSIFICATION MSC THESIS**

**ERDEM AYBEK**

**PAMUKKALE UNIVERSITY INSTITUTE OF SCIENCE**

**COMPUTER ENGINEERING**

**(SUPERVISOR: ASSOC. PROF. DR. MERİÇ ÇETİN)**

**DENİZLİ, NOVEMBER 2024**

Electroencephalography (EEG) is a method used to measure and record electrical signals from neurons in the brain. These signals are used in areas such as examining brain functions, diagnosing neurological disorders such as epilepsy, classifying neuropsychological disorders, and evaluating sleep disorders. EEG signals, which are frequently used in the field of health, are quite complex and multi-source electrical signals. A meaningful classification effort reduces human labor and human error, thus providing advantages in the field of health and supporting the efficient use of labor. EEG signals correctly classified by clinical decision support systems help doctors make faster and more accurate diagnoses, which increases the accuracy and reliability of applications. In this study, the classification of EEG signals and detection of epileptic seizures were performed using the CHB-MIT Scalp EEG dataset, which also includes signals belonging to epileptic seizures. Variations were created on the dataset using various data preprocessing methods, and simulation studies were conducted over two scenarios. In this thesis, the neural evolution approach, which is generally used to optimize complex artificial neural networks that are difficult to optimize with traditional methods such as deep learning algorithms, was preferred for the purpose of classifying EEG signals. The simulation results obtained with the NEAT (Neuro Evolution of Augmented Topologies) algorithm, one of the most popular approaches in this field, were compared with the results of convolutional neural networks (CNN), which are other popular deep learning models. The results should not be considered as a performance comparison of the two models. Both artificial intelligence methods used were evaluated in terms of different data preprocessing and modeling approaches. In the obtained results, it was observed that the neural evolution approach has lower evaluation metrics compared to the CNN model. It can be said that the neural evolution approach, which was used for the first time in the literature in the classification of complex and multi-source EEG signals, is at least as successful as CNN models, but can be increased to higher success rates with more intensive hyperparameter optimization studies.

**KEYWORDS:** Electroencephalography, Neural Evolution, Convolutional Neural Networks, Epilepsy, Epileptic Seizure Detection, Artificial Intelligence

# İÇİNDEKİLER

## Sayfa

|                                                                 |             |
|-----------------------------------------------------------------|-------------|
| <b>ÖZET.....</b>                                                | <b>i</b>    |
| <b>ABSTRACT .....</b>                                           | <b>ii</b>   |
| <b>İÇİNDEKİLER .....</b>                                        | <b>iii</b>  |
| <b>ŞEKİL LİSTESİ.....</b>                                       | <b>v</b>    |
| <b>TABLO LİSTESİ .....</b>                                      | <b>vi</b>   |
| <b>KISALTMALAR LİSTESİ.....</b>                                 | <b>viii</b> |
| <b>ÖNSÖZ.....</b>                                               | <b>ix</b>   |
| <b>1. GİRİŞ.....</b>                                            | <b>1</b>    |
| 1.1 Tezin Akışı .....                                           | 2           |
| <b>2. ELEKTROENSAFALOGRAFI.....</b>                             | <b>3</b>    |
| 2.1 EEG Sinyali .....                                           | 4           |
| 2.2 EEG Sinyali Sınıflandırması .....                           | 4           |
| 2.3 Frekans Bazlı Dalgalar .....                                | 5           |
| 2.3.1 Alfa Dalgası .....                                        | 6           |
| 2.3.2 Teta Dalgası .....                                        | 6           |
| 2.3.3 Beta Dalgası .....                                        | 7           |
| 2.3.4 Delta Dalgası.....                                        | 8           |
| 2.4 Morfolojik Dalga Yapıları.....                              | 8           |
| 2.4.1 Mu Dalgası.....                                           | 8           |
| 2.4.2 Lamda Dalgası .....                                       | 9           |
| 2.4.3 Verteks Keskin Dalgası .....                              | 9           |
| 2.4.4 Pozitif Oksipital Keskin Geçiciler .....                  | 9           |
| 2.4.5 Uyku İçiciği .....                                        | 10          |
| 2.4.6 k Kompleksi.....                                          | 10          |
| 2.4.7 Uykuda Delta .....                                        | 11          |
| 2.4.8 REM Hızlı Göz Hareketleri .....                           | 11          |
| 2.4.9 Yavaş Göz Hareketleri.....                                | 11          |
| 2.5 Epilepsi .....                                              | 11          |
| 2.6 EEG Sinyal Sınıflandırması ile Epilepsi Nöbeti Saptama..... | 12          |
| <b>3. PROBLEM TANIMI .....</b>                                  | <b>13</b>   |
| 3.1 Problem Tanımı .....                                        | 13          |
| 3.2 Sinir Evrimi Yaklaşımı ile EEG Sınıflandırması .....        | 13          |
| 3.3 Evrişimli Sinir Ağları ile EEG Sınıflandırması.....         | 15          |
| <b>4. YÖNTEM.....</b>                                           | <b>17</b>   |
| 4.1 Yapay Zeka.....                                             | 17          |
| 4.2 Yapay Sinir Ağları.....                                     | 17          |
| 4.3 Sinir Evrimi Yaklaşımı.....                                 | 18          |
| 4.4 Sinir Evrimi Algoritmaları .....                            | 19          |
| 4.4.1 NEAT Algoritması.....                                     | 22          |
| 4.5 Evrişimli Sinir Ağları .....                                | 24          |
| 4.6 Phisionet Veri Kümesi.....                                  | 24          |
| 4.7 Veri Ön İşleme Süreci .....                                 | 25          |
| 4.8 Öznitellik Çıkarımı .....                                   | 25          |

|           |                                                                    |           |
|-----------|--------------------------------------------------------------------|-----------|
| 4.9       | Pencere Bazlı Kanal Ortalamaları ile Eğitim (Senaryo-1) .....      | 27        |
| 4.10      | Pencere Bazlı Tüm Kanal Değerleri ile Eğitim (Senaryo 2) .....     | 28        |
| 4.11      | Hiperparametre Optimizasyonu.....                                  | 28        |
| 4.12      | Karmaşıklık Matrisi.....                                           | 29        |
| <b>5.</b> | <b>UYGULAMA SONUÇLARI .....</b>                                    | <b>31</b> |
| 5.1.      | Evrişimli Sinir Ağları Temelli EEG Sinyali Sınıflandırma Sonuçları | 32        |
| 5.2.      | NEAT Temelli EEG Sinyali Sınıflandırma Sonuçları.....              | 33        |
| 5.3.      | Hiperparametre Optimizasyon Sonuçları .....                        | 35        |
| 5.3.1     | Kontrol Konfigürasyonu .....                                       | 37        |
| 5.3.2     | Popülasyon Büyüklüğü (PB) Parametresi.....                         | 39        |
| 5.3.3     | Bağlantı Ekleme Olasılığı (BEO) Parametresi .....                  | 42        |
| 5.3.4     | Düğüm Ekleme Olasılığı (DEO) Parametresi.....                      | 46        |
| 5.3.5     | Uyumluluk Eşiği (UE) Parametresi .....                             | 49        |
| 5.3.6     | Ağırlık Mutasyon Gücü (AMG) Parametresi .....                      | 53        |
| <b>6.</b> | <b>SONUÇ VE ÖNERİLER .....</b>                                     | <b>57</b> |
| <b>7.</b> | <b>KAYNAKLAR.....</b>                                              | <b>60</b> |
| <b>8.</b> | <b>ÖZGEÇMİŞ .....</b>                                              | <b>65</b> |

## ŞEKİL LİSTESİ

### Sayfa

|                                                                                                   |    |
|---------------------------------------------------------------------------------------------------|----|
| Şekil 2.1: 22 kanal ve 256 örnekle 1 saniyelik EEG verisi örneği .....                            | 4  |
| Şekil 2.2: Frekans bazlı çeşitli dalga örnekleri .....                                            | 5  |
| Şekil 2.3: saniyede 10 tepe yapan bir sinüs dalgası .....                                         | 6  |
| Şekil 2.4: Saniyede 5 tepe yapan bir sinüs dalgası.....                                           | 7  |
| Şekil 2.5: Saniyede 15 tepe yapan bir sinüs dalgası.....                                          | 7  |
| Şekil 2.6: Saniyede 3 tepe yapan bir sinüs dalgası.....                                           | 8  |
| Şekil 2.7: Lamda ve mu dalgaları örnekleri .....                                                  | 9  |
| Şekil 2.8: Uyku iğciği, keskin dalga, dalga ve pozitif oksipital keskin geçici<br>örnekleri ..... | 10 |
| Şekil 4.1: Biyolojik nöron ile yapay nöron arasındaki ilişki.....                                 | 18 |
| Şekil 4.2: Nöbet içermeyen işlenmemiş EEG verisi örneği.....                                      | 26 |
| Şekil 4.3: Nöbet içeren işlenmemiş EEG verisi örneği.....                                         | 26 |
| Şekil 4.4: Kanalların PDF değerleri .....                                                         | 27 |
| Şekil 4.5: Kanalların Fourier transform değerleri .....                                           | 27 |
| Şekil 5.1: Popülasyon büyüklüğüne göre performans değişim grafiği .....                           | 39 |
| Şekil 5.2: Bağlantı ekleme olasılığına göre performans değişim grafiği .....                      | 43 |
| Şekil 5.3: Düğüm ekleme olasılığına göre performans değişim grafiği.....                          | 46 |
| Şekil 5.4: Uyumluluk eşliğine göre performans değişim grafiği .....                               | 49 |
| Şekil 5.5: Ağırlık mutasyon gücüne göre performans değişim grafiği .....                          | 53 |

## TABLO LİSTESİ

### Sayfa

|                                                                                |    |
|--------------------------------------------------------------------------------|----|
| Tablo 4.1: Hiperparametre optimizasyonu için test edilen parametre setleri.... | 29 |
| Tablo 4.2. Karmaşıklık matrisi gösterimi .....                                 | 30 |
| Tablo 5.1: 1. Veri seti CNN karmaşıklık matrisi .....                          | 32 |
| Tablo 5.2: 1. Veri seti CNN ek metrikler .....                                 | 32 |
| Tablo 5.3: 2. Veri seti CNN karmaşıklık matrisi .....                          | 33 |
| Tablo 5.4: 2. Veri seti CNN karmaşıklık matrisi .....                          | 33 |
| Tablo 5.5: 1. Veri seti NEAT karmaşıklık matrisi .....                         | 33 |
| Tablo 5.6: 1. Veri seti NEAT ek metrikler .....                                | 34 |
| Tablo 5.7: 2. Veri seti NEAT karmaşıklık matrisi .....                         | 34 |
| Tablo 5.8: 2. Veri seti NEAT ek metrikler .....                                | 34 |
| Tablo 5.9: Her iki yapay zeka modeline ait karmaşıklık matrisleri .....        | 35 |
| Tablo 5.10: Her iki yapay zeka modeline ait ek metrikler .....                 | 35 |
| Tablo 5.11: Tüm konfigürasyonların karmaşıklık matrisi sonuçları .....         | 36 |
| Tablo 5.12: Tüm konfigürasyonlara ait performans sonuçları .....               | 36 |
| Tablo 5.13: Kontrol konfigürasyonuna ait hiperparametreler .....               | 38 |
| Tablo 5.14: Kontrol konfigürasyonuna ait karmaşıklık matrisi .....             | 38 |
| Tablo 5.15: Kontrol konfigürasyonu için performans sonuçları .....             | 38 |
| Tablo 5.16: PB ile ilgili 1. konfigürasyona ait karmaşıklık matrisi .....      | 40 |
| Tablo 5.17: PB ile ilgili 1. konfigürasyona ait performans sonuçları .....     | 40 |
| Tablo 5.18: PB ile ilgili 2. konfigürasyona ait karmaşıklık matrisi .....      | 40 |
| Tablo 5.19: PB ile ilgili 2. konfigürasyona ait performans sonuçları .....     | 41 |
| Tablo 5.20: PB ile ilgili 3. konfigürasyona ait karmaşıklık matrisi .....      | 41 |
| Tablo 5.21: PB ile ilgili 3. konfigürasyona ait performans sonuçları .....     | 41 |
| Tablo 5.22: PB ile ilgili 4. konfigürasyona ait karmaşıklık matrisi .....      | 42 |
| Tablo 5.23: PB ile ilgili 4. konfigürasyona ait performans sonuçları .....     | 42 |
| Tablo 5.24: BEO ile ilgili 1. konfigürasyona ait karmaşıklık matrisi .....     | 43 |
| Tablo 5.25: BEO ile ilgili 1. konfigürasyona ait performans sonuçları .....    | 43 |
| Tablo 5.26: BEO ile ilgili 2. konfigürasyona ait karmaşıklık matrisi .....     | 44 |
| Tablo 5.27: BEO ile ilgili 2. konfigürasyona ait performans sonuçları .....    | 44 |
| Tablo 5.28: BEO ile ilgili 3. konfigürasyona ait karmaşıklık matrisi .....     | 44 |
| Tablo 5.29: BEO ile ilgili 3. konfigürasyona ait performans sonuçları .....    | 45 |
| Tablo 5.30: BEO ile ilgili 4. konfigürasyona ait karmaşıklık matrisi .....     | 45 |
| Tablo 5.31: BEO ile ilgili 4. konfigürasyona ait performans sonuçları .....    | 45 |
| Tablo 5.32: DEO ile ilgili 1. konfigürasyona ait karmaşıklık matrisi .....     | 46 |
| Tablo 5.33: DEO ile ilgili 1. konfigürasyona ait performans sonuçları .....    | 47 |
| Tablo 5.34: DEO ile ilgili 2. konfigürasyona ait karmaşıklık matrisi .....     | 47 |
| Tablo 5.35: DEO ile ilgili 2. konfigürasyona ait performans sonuçları .....    | 47 |
| Tablo 5.36: DEO ile ilgili 3. konfigürasyona ait karmaşıklık matrisi .....     | 48 |
| Tablo 5.37: DEO ile ilgili 3. konfigürasyona ait performans sonuçları .....    | 48 |
| Tablo 5.38: DEO ile ilgili 4. konfigürasyona ait karmaşıklık matrisi .....     | 48 |
| Tablo 5.39: DEO ile ilgili 4. konfigürasyona ait performans sonuçları .....    | 49 |
| Tablo 5.40: UE ile ilgili 1. konfigürasyona ait karmaşıklık matrisi .....      | 50 |
| Tablo 5.41: UE ile ilgili 1. konfigürasyona ait performans sonuçları .....     | 50 |
| Tablo 5.42: UE ile ilgili 2. konfigürasyona ait karmaşıklık matrisi .....      | 51 |
| Tablo 5.43: UE ile ilgili 2. konfigürasyona ait performans sonuçları .....     | 51 |

|                                                                            |    |
|----------------------------------------------------------------------------|----|
| Tablo 5.44: UE ile ilgili 3. konfigürasyona ait karmaşıklık matrisi.....   | 51 |
| Tablo 5.45: UE ile ilgili 3. konfigürasyona ait performans sonuçları.....  | 51 |
| Tablo 5.46: UE ile ilgili 4. konfigürasyona ait karmaşıklık matrisi.....   | 52 |
| Tablo 5.47: UE ile ilgili 4. konfigürasyona ait performans sonuçları.....  | 52 |
| Tablo 5.48: AMG ile ilgili 1. konfigürasyona ait karmaşıklık matrisi ..... | 54 |
| Tablo 5.49: AMG ile ilgili 1. konfigürasyona ait performans sonuçları..... | 54 |
| Tablo 5.50: AMG ile ilgili 2. konfigürasyona ait karmaşıklık matrisi ..... | 54 |
| Tablo 5.51: AMG ile ilgili 2. konfigürasyona ait performans sonuçları..... | 54 |
| Tablo 5.52: AMG ile ilgili 3. konfigürasyona ait karmaşıklık matrisi ..... | 55 |
| Tablo 5.53: AMG ile ilgili 3. konfigürasyona ait performans sonuçları..... | 55 |
| Tablo 5.54: AMG ile ilgili 4. konfigürasyona ait karmaşıklık matrisi ..... | 56 |
| Tablo 5.55: AMG ile ilgili 4. konfigürasyona ait performans sonuçları..... | 56 |

## KISALTMALAR LİSTESİ

|             |   |                                         |
|-------------|---|-----------------------------------------|
| <b>CNN</b>  | : | Convolutional Neural Network            |
| <b>NEAT</b> | : | Neuro Evolution of Augmented Topologies |
| <b>REM</b>  | : | Rapid Eye Movement                      |
| <b>NREM</b> | : | Non-Rapid Eye Movement                  |
| <b>EEG</b>  | : | Elektroensefalografi                    |
| <b>FFT</b>  | : | Fast Fourier Transform                  |
| <b>PDF</b>  | : | Probability Distribution Function       |
| <b>TP</b>   | : | True Positive                           |
| <b>TN</b>   | : | True Negative                           |
| <b>FP</b>   | : | False Positive                          |
| <b>FN</b>   | : | False Negative                          |
| <b>PB</b>   | : | Popölasyon Büyüklüğü                    |
| <b>BEO</b>  | : | Bağlantı Ekleme Olasılığı               |
| <b>DEO</b>  | : | Düğüm Ekleme Olasılığı                  |
| <b>UE</b>   | : | Uyumluluk Eşiğı                         |
| <b>AMG</b>  | : | Ağırlık Mutasyon Gücü                   |
| <b>GPU</b>  | : | Graphics Processing Unit                |
| <b>MEG</b>  | : | Manyetoensefalografi                    |
| <b>fMRI</b> | : | functional Magnetic Resonance Imaging   |
| <b>YSA</b>  | : | Yapay Sinir Ağı                         |
| <b>DPA</b>  | : | Doğrudan Politika Araması               |
| <b>KMUE</b> | : | Kovaryans Matrisi Uyarlama Evrim        |
| <b>STEE</b> | : | Sinirsel Topolojilerin Evrensel Edinimi |
| <b>KSSE</b> | : | Kooperatif Sinaps Sinir Evrimi          |

## ÖNSÖZ

Bu çalışmanın her aşamasında desteğinden vazgeçmeyen sayın Doç. Dr. Meriç ÇETİN'e, tüm eğitim hayatımın arkasında durmuş olan anne ve babam Sevinç ve Durmuş AYBEK'e içtenlikle teşekkürlerimi sunarım.

## 1. GİRİŞ

Elektroensefalografi (EEG) sinyalleri, beynin elektriksel aktivitelerini ölçmek ve anlamak için kullanılan araçlardan biridir. Bu sinyaller kullanılarak beyin fonksiyonlarının incelenmesinden nörolojik ve nöropsikolojik hastalıkların teşhisine, epilepsi nöbeti saptamasından uyku bozukluklarının değerlendirilmesine kadar birçok farklı alanda inceleme yapılabilir. Özellikle epilepsi gibi nörolojik durumların tespitinde kritik öneme sahip olan EEG sinyalleri, yüksek boyutlu ve karmaşık bir yapıya sahiptir. Bu nedenle, bu sinyallerin doğru bir şekilde sınıflandırılması, bilimsel araştırmaların yanı sıra tıbbi tanı süreçlerinde daha etkili kararlar alınmasına olanak sağlar.

Bu tez çalışmasının amacı, daha önce EEG sinyallerinin sınıflandırılması ve epilepsi nöbetlerinin tespitinde kullanılmamış olan sinir evrimi algoritmasını (NEAT-Neuro Evolution of Augmented Topologies) kullanmak ve bu yöntemi, literatürde yaygın şekilde kullanılan Evrişimli Sinir Ağları (CNN-Convolutional Neural Networks) ile karşılaştırmaktır. Çalışmada, Massachusetts Teknoloji Enstitüsü (MIT) ve Boston Çocuk Hastanesi tarafından oluşturulan CHB-MIT Scalp EEG veri seti kullanılmış ve iki farklı senaryo üzerinden benzetim çalışmaları gerçekleştirilmiştir. Bu tezde elde edilen sonuçlar, yalnızca farklı yapay zeka modellerinin performanslarının birbiriyle doğrudan karşılaştırılması amacıyla değerlendirilmemelidir. Yapay zeka alanında geliştirilen algoritmalar genellikle belirli veri türleri veya problemler için özelleştirilmiştir. NEAT ve CNN modelleri de, farklı yapay zeka mimarileri ve optimizasyon süreçlerine dayanan iki ayrı yaklaşımdır. Bu yöntemler, EEG gibi karmaşık ve çok boyutlu sinyaller üzerinde çalışırken farklı avantajlar ve dezavantajlar sunmaktadır. NEAT algoritması, başlangıçta basit yapılarla çalışmaya başlayarak sinir ağlarını evrimsel bir süreçle optimize ederken, CNN gibi derin öğrenme modelleri önceden tanımlanmış bir mimari çerçevesinde çalışır ve genellikle yüksek doğruluk oranlarına ulaşır. Ancak, CNN'in başarısı, güçlü donanım gereksinimleri ve detaylı hiperparametre ayarlamaları gibi faktörlere bağlıdır. Öte yandan, NEAT algoritması da çok boyutlu ve zamana göre değişen dinamiklere sahip veri türlerinde performans açısından sınırlamalar gösterebilir.

Bu alıřmada elde edilen bulgular, her iki yntemin farklı tasarımı yaklařımlarına dayanması nedeniyle, yalnızca başarı metrikleri zerinden dođrudan bir stnlk deđerlendirmesi olarak dřnlmemelidir. Bunun yerine, her iki modelin gl ve zayıf ynlerinin karřılařtırmalı bir analizinin yapılması ve sinir evrimi yaklařımının EEG sınıflandırma problemlerindeki uygulanabilirliđinin deđerlendirilmesi olarak dřnlmemelidir. Bu tez alıřması, literatrde ilk kez EEG sinyallerinin sınıflandırılmasında ve epilepsi nbeti saptamasında sinir evrimi yaklařımını uygulayarak bu yntemin potansiyelini gstermektedir. Aynı zamanda, yapay zeka ve sinir evrimi alanlarının farklı zelliklerini bir araya getiren hibrit model yaklařımlarının gelecekte EEG analizine nasıl katkı sađlayabileceđi konusunda bir referans noktası oluřtırmaktadır. Bu kapsamda, alıřmada elde edilen bulgular yalnızca sađlık alanında dođrudan fayda sađlamakla kalmayıp, bu alanda devam eden ve gelecekteki alıřmalara da yn verebilir.

## **1.1 Tezin Akıřı**

Bu tezin ieriđinde giriř blmnde tezin amacından, neminden ve akıřından bahsedilmiřtir. İkinci blmde EEG, EEG sinyal sınıflandırması, epilepsi ve EEG sinyallerinden epilepsi nbeti saptama konularına deđerilmiřtir. nc blm ierisinde problem tanımı yapılmıř ve sinir evrimi yaklařımı ile evriřimli sinir ađları temelli EEG sinyali sınıflandırma konusu, bu alanda sunulan farklı literatr alıřmaları zerinden incelenmiřtir. Drdnc blmde arařtırma yntemine ve zerinde alıřılan veri kmelerine yer verilmiřtir. Beřinci blmde, uygulama sonuları sunulmuř, son olarak altıncı blmde genel bir deđerlendirme yapılarak sonu ve neriler aktarılmıřtır.

## 2. ELEKTROENSEFALOGRAFI

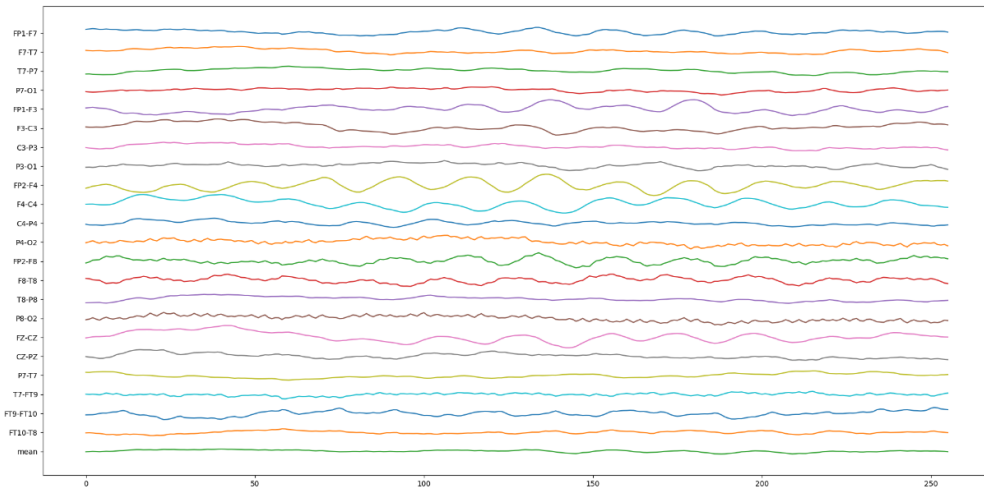
Elektroensefalografi beyin yapılarından kaynaklanan elektriksel aktiviteyi kaydeden bir tekniktir (Bora ve Yeni, 2020). Bu tekniğin günümüzdeki başlıca kullanım alanları nörolojik araştırmalardır. EEG dalgaları sürekli olmayan yapıları ve birçok gürültü kaynağı sebebiyle sınıflandırılmaları oldukça zor olan gerçek-zamanlı elektrik sinyalleridir. Bir EEG kaydı gözlemlendiğinde dalgaların patolojik bir duruma işaret edip etmediğini anlamak dahi oldukça fazla eğitim gerektiren bir durumdur. Ek olarak, herhangi bir duruma işaret etse dahi patolojik durumun varlığını ispatlamış sayılmaz.

İlk EEG kayıt cihazı dünyaya 1929'da Hans Berger tarafından tanıtılmıştır (Collura 1993). Elektroensefalografinin teknik kullanımında, kafa derisine takılan elektrotlar sayesinde beynin elektriksel aktivitesinin kaydedilmesi sağlanmaktadır. Beynin zaman içinde nasıl işlediğine dair kanıt sağlayan, erişilebilir bir test olan Elektroensefalografi insan beyninin elektriksel aktivitesini yansıtan potansiyellerin ölçümüdür. Yüzey elektrotları 6 cm<sup>2</sup> ve bir kaç mm derinliğe kadar yük ölçümü yapabilmektedir. Sinir hücreleri arasındaki bilgi aktarımı aksiyon potansiyeli ile sağlanmaktadır. Bu şekilde sinyaller alınmakta ve EEG sinyali elde edilmektedir. Doktorlar ve bilim insanları tarafından yaygın olarak kullanılmakta olan EEG, beynin elektriksel aktivitesinin incelenmesini, epilepsi, beyin tümörleri, kafa travması, uyku bozuklukları, bunama ve ameliyat sırasında anestezi derinliğinin izlenmesi gibi nörolojik hastalıkların teşhisi için oldukça önemli araçlardan biridir (Hazarika ve diğ. 1997). Bunun yanında anormallikler, davranış bozuklukları (otizm gibi), dikkat bozuklukları, öğrenme güçlükleri ve dil gecikmesinin tedavisinde de faydalıdır. Epilepsi sınıflandırması, nöbet tahmini ve takibi sürecinde de kullanılan EEG, beyindeki sinir hücre gruplarının elektriksel aktivitesini grafiksel olarak gösterir. Bu şekilde, epilepsinin nedenlerinin araştırılması, yorumlanması, teşhis ve tedavi sürecinin geliştirilmesinde büyük yol kat edilir (Guo ve diğ. 2010). Epilepsi nöbeti saptama çalışmalarında EEG, kullanılabilecek diğer teknik ve yaklaşımlara göre hız, maliyet ve erişilebilirlik açısından daha verimli olduğundan araştırmalarda gittikçe daha fazla popülerlik kazanmaktadır (Nigam ve Graupe 2004). Bu tez çalışması, EEG sinyallerinin sınıflandırılmasını ve aynı zamanda EEG sinyallerinin epilepsi

hastalığına işaret edip etmediğini evrimsel bir yapay zeka algoritması ile tespit etmeyi konu almaktadır.

## 2.1 EEG Sinyali

Elektroensefalografi insan beyninde oluşan elektriksel potansiyel farklarını kaydetmeye yarar. Kaydedilen bu değerler beynin genel durumu hakkında bize fikir sağlayabilmektedir. EEG kayıtları kafanın çeşitli bölgelerine yerleştirilmiş olan elektrotlar sayesinde kaydedilmektedir. Her elektrot kendi bölgesindeki potansiyel farkları sürekli ölçmekte olduğu için sonuç olarak elimize elektrot sayısı kadar zaman sinyali geçmektedir. Ayrıca elektrotların bağlanma şekilleri değişiklik gösterebilmektedir. Bu çalışmada kullanılan EEG sinyalleri saniyede 256 örnek 20-28 arasında kanal içeren yarım saat ila kırk dakikalık veri setlerinden oluşmaktadır. Şekil 2.1’de 22 kanal ve bu kanalların ortalamasını içeren 256 örnekli 1 saniyelik bir EEG verisi örneği görülmektedir.



Şekil 2.1: 22 kanal ve 256 örnekli 1 saniyelik EEG verisi örneği

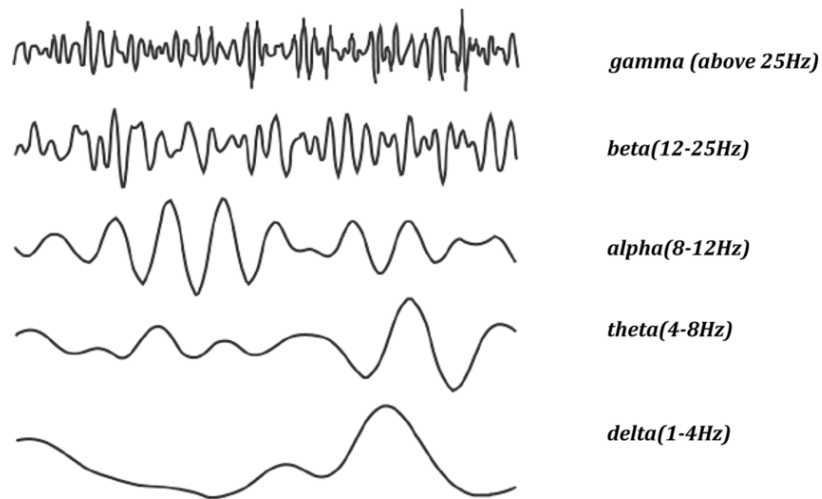
## 2.2 EEG Sinyali Sınıflandırması

Sınıflandırma denildiğinde, bir veri setinin çeşitli kavramsal olarak benzer elemanlarını içeren gruplara bölme işlemi anlaşılmaktadır. Bu tezde üzerinde çalışılan veri setinin “Epilepsi İçeren” ve “Epilepsi İçermeyen” kavramsal gruplarına bir evrimsel sinir ağı geliştirme tekniği olan NEAT ile ayrıştırılabilirliğini ölçmek

amaçlanmıştır. Bu sınıflandırmayı yapabilmek için öncelikle her örneğin sınıflardan biri ile önceden işaretlenmiş olması gerekmektedir. Kullanılan veri setinde EEG sinyalinin hangi saniye aralıklarında epilepsi içermekte olduğu bilinmektedir. Bu nedenle, veri seti epilepsi içeren ve epilepsi içermeyen olarak işaretlenmiştir. Bu aşamadan sonra sınıflandırma işlemi için veri setinin kendisini doğrudan kullanmak oldukça verimsiz olacağından veri setinin çeşitli anlamlı özellikleri hesaplanmıştır. Bu aşama özellik çıkarımı olarak da adlandırılmaktadır. Sınıflandırma işlemi bu ilgili öznitelikler üzerinden gerçekleştirilmektedir. Bunun yanında, veri seti daha kolay anlaşılır çeşitli alt gruplara da ayrılmıştır, bu anlamda özellikle frekans bazlı dalgalardan özellik çıkartırken faydalanılmıştır. Teknik olarak, EEG sinyalleri çeşitli özelliklerine ve çeşitli olaylarla korelasyonlarına göre belirgin bir sınıflandırmaya tabi tutulmuştur. Devam eden alt bölümlerde anlatılan EEG sinyalleri ve bu sinyallerin özellikleri için EEG Atlası (Bora ve Yeni 2020) temel alınmıştır.

### 2.3 Frekans Bazlı Dalgalar

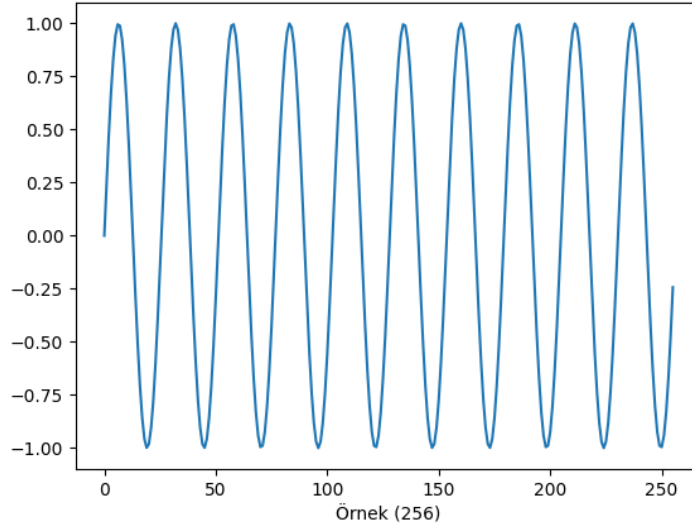
EEG dalgaları birden fazla kaynağın birleşmesiyle son halini alan karmaşık zaman dalgaları olarak incelenmek istendiğinde Fourier analizi ile frekanslarının ayrıştırılarak incelenmesi en popüler yöntemlerden biridir (Strayhorn 2024). Bu ayrıştırma sonrasında zaman dalgaları frekans değerlerine göre beş temel grupta incelenirler. Bunlar alfa, beta, gama, teta ve delta dalgalarıdır. Şekil 2.2’de frekans bazlı çeşitli dalga örnekleri sunulmuştur (Tiwari ve diğ., 2022).



Şekil 2.2: Frekans bazlı çeşitli dalga örnekleri

### 2.3.1 Alfa Dalgası

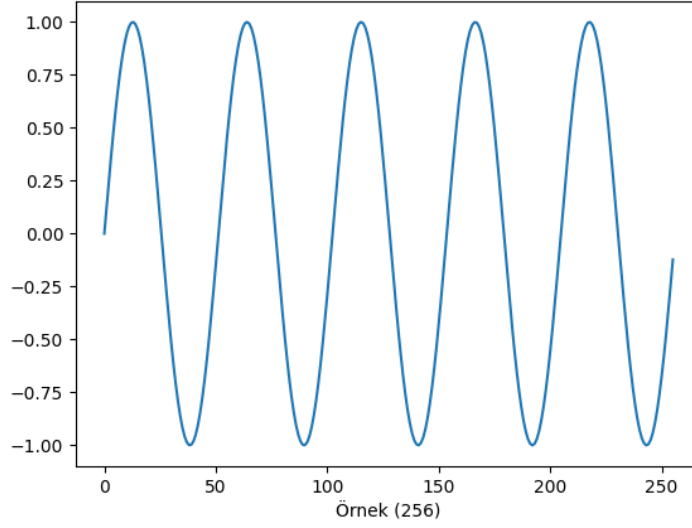
8 - 12 Hz frekans aralındaki dalgalara alfa dalgaları ismi verilmektedir. Bu dalgalar genellikle uyanık olma durumunda görülen sinüzoidal, düzenli dalgalardır. Bu dalgalar normal konumda en sık görülen dalgalardır (Strayhorn 2024). Örneğin zihnimizin çok yorulmadığı ancak dinlenme konumunda da olmadığımız zamanlarda bu dalga türü görülmektedir. Alfa dalgaları ölçülürken, diğer dalgalara geçiş olması bizlere bilgi sunabilir. Örneğin, alfa durumundan beta durumuna geçişin olduğu bir EEG grafiğinde bireyin problem çözmeye başladığı anlaşılabilirken, teta durumuna olan bir değişiklik dinlenmeye geçildiğini gösterebilir. Şekil 2.3'te saniyede 10 tepe yapan bir sinüs dalgası görülmektedir. Bir alfa dalgasının da benzer genliğe sahip olması beklenir.



Şekil 2.3: saniyede 10 tepe yapan bir sinüs dalgası

### 2.3.2 Teta Dalgası

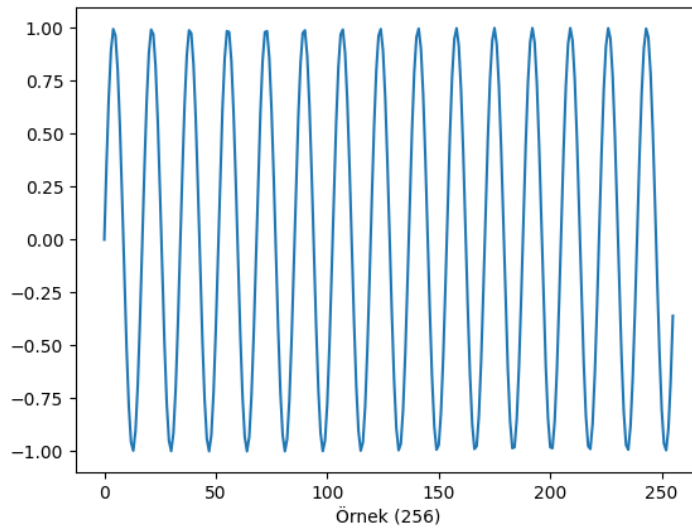
4 - 7 Hz aralığındaki dalgalara teta dalgaları denilmektedir. Teta dalgaları uyku ve uyuşma hali ile ilişkilendirilen yavaş dalgalardır. Örneğin, uyurken beynimizin ürettiği dalgalardır. Teta durumunda bir zihnin alfa durumuna geçiş yapması normal karşılanabilecek bir uyanma durumu olarak adlandırılabilir (Strayhorn 2024). Şekil 2.4'te saniyede 5 tepe yapan bir sinüs dalgası görülmektedir ve teta dalgalarının da benzer genliğe sahip olması beklenmektedir.



**Şekil 2.4:** Saniyede 5 tepe yapan bir sinüs dalgası

### 2.3.3 Beta Dalgası

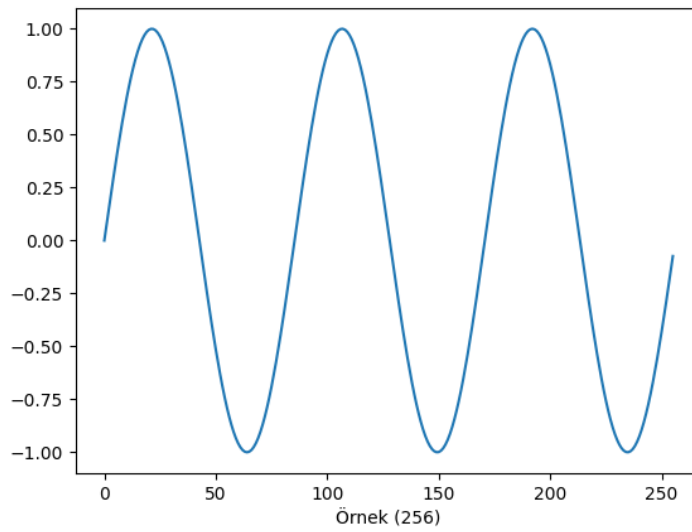
12 Hz'den daha büyük frekans değerlerine beta dalgaları denilmektedir. Hızlı aktivite olarak da nitelendirilen bu dalga, beynin aktif olduğu zamanlarda ortaya çıkmaktadır (Strayhorn 2024). Aynı alfa gibi tüm sağlıklı bireylerde kolayca görülebilen bir beyin dalgasıdır. Alfadan farklı olarak, problem çözme ve çalışma ile ilişkilidir. Şekil 2.5'te saniyede 15 tepe yapan bir sinüs dalgası görünmektedir ve beta dalgalarının da benzer genliğe sahip olmaları beklenmektedir.



**Şekil 2.5:** Saniyede 15 tepe yapan bir sinüs dalgası

### 2.3.4 Delta Dalgası

0 - 3.5 Hz aralığındaki dalgalara delta dalgaları denilmektedir. Bu dalgaların genel olarak hakim olduğu zaman dilimi sağlıklı bireylerde koma gibi bir derin uyku dönemi olabilir (Strayhorn 2024). Alfa dalgasına göre çok daha yavaş kalan bu evrede elektriksel potansiyel oldukça sakinleşmiş bir noktaya taşınmıştır. Şekil 2.6'da saniyede 3 tepe yapan bir sinüs dalgası görünmektedir. Delta dalgalarının da benzer genlikte olmaları beklenir.



Şekil 2.6: Saniyede 3 tepe yapan bir sinüs dalgası

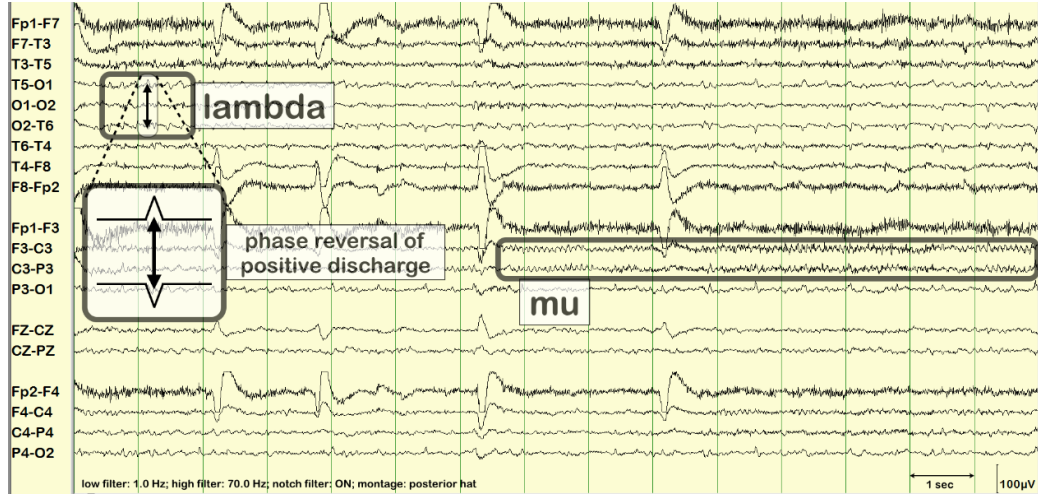
## 2.4 Morfolojik Dalga Yapıları

### 2.4.1 Mu Dalgası

Mu dalgası, ark ya da kemer biçimini alan 8 - 11 Hz aralığındaki sinyal aktivitesidir. Santral bölgelerde lokalizedir. Gözlerin açılıp alfa dalgalarının bloke olması ile ortaya çıkabilir ve bir hareketin düşünülmesi ile kaybolur. Şekil 2.7'de F3 - C3 ve C3 - P3 hattının sonuna doğru mu dalgası örneği görülmektedir (Strayhorn 2024).

### 2.4.2 Lamda Dalgası

Gözler açık iken oksipital bölgede görülen tarak dişi şeklinde difazik dalgalardır. Şekil 2.7’de T5 - O1, O1 -O2 ve O2 - T6 hattının başında lamda dalgası örneği görülmektedir (Strayhorn 2024).



Şekil 2.7: Lamda ve mu dalgaları örnekleri

### 2.4.3 Verteks Keskin Dalgası

En yüksek kuvvetin Cz (Central Verteks) üzerinde bulunduğu monofazik, bifazik, trifazik dalga özelliği gösterebilen uyuklama ve NREM (Non-Rapid Eye Movement) uykusunda görünen bifazik dalgalardır. Fz (Frontal Verteks) ve Pz (Parietal Verteks) kanallarına da yayılabilen bu dalga türünün kuvveti yaş ile azalmaktadır. Küçük çocuklarda Cz üzerinde en yüksek kuvvette görülen verteks keskin dalgası bazen asimetrik yayılım gösterebilir, ancak asimetrik yayılımın sürekliliği bir patolojiye işaret ediyor olabilir. Şekil 2.8 üzerinde C elektrotlarında daha güçlü olacak şekilde işaretli bölgede vertex dalgası görülebilir (Strayhorn 2024).

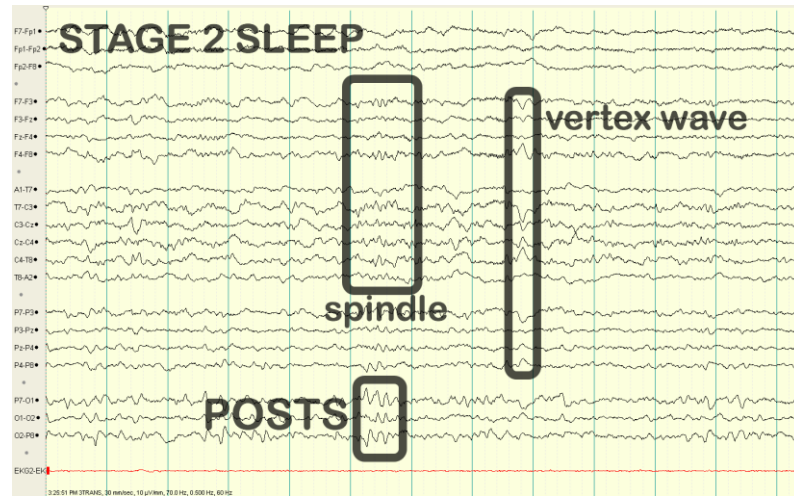
### 2.4.4 Pozitif Oksipital Keskin Geçiciler

Monofazik üçgen biçiminde dalgalardır. 4 yaş itibariyle normal bireylerin çoğunda görülen 15 yaşa kadar daha da sık görülmeye başlayan 35 yaşa kadar aynı

seviyede devam eden ve 50 yaş sonrası sıklığı azalmaya başlayan epileptiform görünümüne benzer, genellikle düzensiz frekanslarda görülen iyi huylu deşarjlardır. REM (Rapid Eye Movement) uykusunda görülmesi nadir olan bu dalga türü düzensiz frekanslara sahiptir. Gün içerisindeki kısa uykularda ya da uykunun erken evrelerinde oksipital bölge elektrotlarında gözlenir. Şekil 2.8 üzerindeki işaretli bölgede (posts bölgesi) pozitif oksipital keskin geçiciler dalgasına bir örnek görülmektedir (Strayhorn 2024).

#### 2.4.5 Uyku İğciği

Uyku iğciği ritmik monomormik 10-14 Hz ve 20-100  $\mu$ V aralıklarında seyreden 2. evre uykuda görülmesi olası daha derin evrelerde görülmesi mümkün 0.5 saniye süreli sıklıkla vertex dalga ile birlikte gözlemlenen dalga türüdür. Bu dalgaya örnek Şekil 2.8 üzerindeki işaretli bölgede (spindle bölgesi) sunulmuştur (Strayhorn 2024).



Şekil 2.8: Uyku iğciği, keskin dalga, dalga ve pozitif oksipital keskin geçici örnekleri

#### 2.4.6 k Kompleksi

Başlangıç keskin sonra yavaş devam eden genellikle uyku iğciklerini takip eden dalga formudur. Temel aktiviteden bariz yüksek seyretmektedir. Yüzeysel uyku ya da derin uyku sırasında görülmeleri mümkündür.

#### **2.4.7 Uykuda Delta**

Geniş dağılımlı ve simetrik olmasına karşın senkron olmak zorunda olmayan delta dalgaları uykunun derin evreleri olan 3. ve 4. evresinde uyku sırasında gözlemlenen diğer dalgalarla birlikte gözlemlenir. Özellikle 4 Hz değerinin altında kalan, kuvveti en az 75  $\mu$ V değerinde olan dalgalar gözlemlendiğinde delta dalgaları olarak değerlendirilir.

#### **2.4.8 REM Hızlı Göz Hareketleri**

REM bir EEG dalga formu değil, bir biyolojik artefaktır. 300 ms'den küçük yukarı doğru seyreden bir defleksiyon hareketi ile yavaş göz hareketlerinden ayrılır.

#### **2.4.9 Yavaş Göz Hareketleri**

Uykunun derin fazından ziyade uyuklama ve yüzeysel uyku sırasında görülen dağılım açısından REM ile benzer olan 500 milisaniyeden daha büyük yukarı göz hareketleridir.

### **2.5 Epilepsi**

Beyindeki ani elektriksel deşarjlara nöbet, nöbetlerin tekrarlı şekilde yaşandığı duruma da epilepsi denmektedir. Epilepsi her yaşta tanısı konulabilen ve uzun süreli tedavi ve takip gerektiren bir hastalık olup yaşam kalitesini önemli ölçüde etkileyen doğrudan ve sosyal etkileri olan akademik başarıyı etkileyebilen, kaygı, duygusal strese bağlı sorunlar oluşmasına sebep olabilen, hastalar yanında hasta ailelerini de etkileyebilen bir hastalıktır (Görgülü ve Fesci 2011). Sürekli nöbetlerle tekrarlı seyreden bir hastalık olan epilepsi beynin bir bölümünde veya tamamında meydana gelen ani ve düzensiz elektriksel boşalma sonucu ortaya çıkan bir klinik durumdur (Li ve diğ. 2016). Epilepsi nöbetleri kasılma, bilinç kaybı, titreme, ağızda köpürme, kısa süreli göz dalması, ağız şapırdatma, sorulara kısa süreli yanıt verememe gibi belirtilerle kendini gösterebilmektedir. Özellikle çocuklar ve yetişkinlerde yaygın

olarak görülen bu hastalık, dünya nüfusunun %1'inde gözlenmektedir (Nicolau ve George 2012).

## **2.6 EEG Sinyal Sınıflandırması ile Epilepsi Nöbeti Saptama**

EEG, beyindeki elektriksel sinyalleri ölçmekte olan bir cihaz olduğu için nöbet yatkınlığı olan bireylerin tespitinde de kullanılmaktadır. Günümüzde, yapay zeka teknikleri ile EEG sinyal sınıflandırması ve epilepsi nöbet tespiti, otomatik, hızlı ve yüksek doğruluk içeren bir yöntem olarak tercih edilmektedir. Bu araştırmalardan elde edilen sonuçlar klinik karar destek sistemlerinde kullanılmaktadır. EEG sinyallerinden epilepsi saptayabilmek için öncelikle epileptik ve normal durumları içeren EEG verisinin uzmanlar tarafından epileptik nöbetlerin olduğu ve olmadığı zaman bölümleri olarak etiketlenmesi gerekir. Veri ön işleme adımlarından gürültü giderme, EEG sinyallerinin belirli zaman dilimlerine bölünmesi (segmentasyon) ve normalizasyon işlemlerinin ardından karmaşık bir yapıya sahip olan EEG sinyallerinin ayırt edici özelliklerin çıkarılması süreci gerçekleştirilir. EEG sinyallerinin özniteliklerini çıkarırken epilepsi ile anlamlı şekilde ilişkilendirilebilecek özellikler tercih edilmesi önemlidir. Özellik çıkarımı uygulamaları için genellikle işaretin ortalama, varyans, enerji değeri gibi zamana dayalı bileşenleri, hızlı Fourier dönüşümü, güç spektrumu, delta, teta, alfa, beta, gama bant güç değerleri gibi frekans bağımlı bileşenleri veya Dalgacık Dönüşümü (Wavelet Transform), entropi gibi farklı öznitelikleri kullanılabilir. Uygun bir model seçiminden sonra etiketlenen EEG verisi kullanılarak model eğitimi gerçekleştirilir. Son olarak, tasarlanan model daha önce görülmemiş veri üzerinde test edilerek modele ait doğruluk ve hassasiyet metrikleri değerlendirilir. Literatürde makine öğrenimi algoritmaları kullanılarak nöbet saptama, epilepsi lateralizasyonu, nöbet durumlarının ayırt edilmesi ve lokalizasyonu için beyin veri setlerinde de uygulamaları görülmektedir. Yaygın olarak kullanılan modeller arasında yapay sinir ağları, destek vektör makineleri, KNN, Naive Bayes, öğrenme makineleri, karar ağaçları ve rastgele orman gibi çeşitli makine öğrenimi sınıflandırıcıları sayılabilir (Siddiqui ve diğ. 2020).

### **3. PROBLEM TANIMI**

#### **3.1 Problem Tanımı**

EEG sinyalleri, özellikle epilepsi gibi nörolojik hastalıkların tespiti ve yönetimi için kritik öneme sahiptir, ancak, bu sinyallerinin yüksek boyutlu ve karmaşık yapısı, doğru şekilde analiz edilmesini zorlaştırmaktadır. Sınıflandırma için kullanılan mevcut analiz yöntemlerinde genellikle belirli veri türlerine göre özelleşmiş algoritmalar tercih edilmektedir. Bu yöntemlerin başarısı, donanım gereksinimlerinin iyileştirilmesi ve/veya parametre optimizasyonu gibi faktörlere bağımlı olabilir. Bu tez çalışmasında, EEG sinyallerinin doğru ve etkin bir şekilde sınıflandırılması için sinir evrimi yaklaşımının uygulanabilirliğini araştırmak ve bunun geleneksel derin öğrenme yöntemleriyle karşılaştırılmasındaki güçlü ve zayıf yönleri ortaya koymak hedeflenmiştir. Ek olarak, NEAT algoritması ile epilepsi nöbetlerinin tespiti yapılmaya çalışılmıştır. Bu araştırmalar gerçekleştirilirken tasarlanan yapay zeka modellerinin donanım gereksinimleri ve parametre optimizasyonu ihtiyaçlarının ne olacağı hesaplanmış ve elde edilen sonuçlar üzerinden genel bir değerlendirme yapılmıştır.

#### **3.2 Sinir Evrimi Yaklaşımı ile EEG Sınıflandırması**

Bu bölümde, NEAT algoritması, NEAT algoritmasının öncülü olan algoritmalara nasıl üstün geldiği, alternatif algoritmalara karşı hangi noktalarda avantaj sağladığı ve NEAT varyasyonları gibi konular aktarılmıştır.

L. Chen ve D. Alahakoon, yayınlarında veri sınıflandırma problemi için NEAT'in hibriti olan L-NEAT (Learning NEAT) metodunu önermişler ve L-NEAT yönteminin NEAT'e göre doğruluk açısından üstün geldiğine dikkat çekmişlerdir (Chen ve Alahakoon 2006). Xin Yao, evrimsel metotların gradyan temelli algoritmaların lokal minimuma sıklıkla takıldığı durumlarda, gradyan temelli algoritmalara iyi bir alternatif olabileceğine değinmiştir (Yao 1999). A.B.M. Wijaya vd. yayınlarında diyabet tespitinde NEAT algoritmasını çok katmanlı bir perseptron ile iş birliği yaparak %75,5 güven sağlanarak çözülebileceğine değinmiştir (Wiyaja ve

diğ. 2021). Papavasileiou ve diğerleri derlemelerinde NEAT algoritmasının çeşitli varyasyonlarına değinmiş, 60 geçerli metot üzerinde kalite değerlendirmesi yapmış ve haklarında özet bilgiler vermiştir (Papavasileiou ve diğ. 2021). K. O. Stanley ve R. Miikkulainen, yayınında NEAT algoritmasının öncesinde gelen nöro-evrim metotlarından daha iyi sonuçlar elde ettiğini ve önceki metotların eksikliklerini nasıl kapattıklarını açıklamıştır (Stanley ve Miikkulainen 2002). Szegedi, Gabor ve diğerleri çalışmalarında gradyan temelli metotlardaki gibi geriye yayılım olmadığından sinir evrimi yöntemlerini avantajlı bulmuş ve alternatif bir teknik tanıtmışlardır (Szegedi ve diğ. 2019). Dweiri ve diğerleri çalışmalarında önerilen metot ile 84% doğruluk oranı yakalamış ve NEAT algoritmasının EEG verisi için verimli olabileceğine dikkat çekmiştir (Dweiri ve diğ. 2023). Bird, J.J. ve diğerleri çalışmalarında düşme telafisi tespiti için EEG üzerinde sinir evrimi algoritmalarını kullanmış ve 90,52% ye kadar doğruluk elde etmiştir (Bird ve diğ. 2023). Raj, P. ve diğerleri yayınlarında 90,06% doğruluk elde etmiş ve model iyileştirilmesi için popülasyon büyüklüğü ile nesil sayısına dikkat çekmiştir (Raj ve diğ. 2024). Risto Miikkulainen, çalışmasında sinir evrimi yaklaşımını belirli bir görevi öğrenmek için sinir ağı tasarımının detaylarını değiştirmeye yönelik bir yöntem olarak tanımlamaktadır (Lehman ve Miikkulainen 2013).

Bunlarla birlikte NEAT algoritmasının günümüzde kullanım alanları çeşitlenmeye devam etmektedir ve çeşitli senaryolara adapte edilebilirliği gösterilmiştir. Pei-Chi Huang ve diğerleri çalışmalarında bir robot kol kontrolcüsü olarak NEAT algoritmasını kullanmayı denemişler ve insan kontrolünün azaltılarak robot kontrolcüsünün daha otonom olmasını başarmışlardır (Huang ve diğ. 2014). Matthew Hausknecht ve diğerleri çalışmalarında bir NEAT varyasyonu olan HyperNEAT ile atari oyunlarının oynanmasını denemişler ve HyperNEAT ile birden fazla görev arasında sorunsuz geçiş sağlamaya çalıştıkları ilk adım olarak nitelendirmişlerdir (Hausknecht ve diğ. 2012). Timothy Andersen çalışmasında Othello oyunu üzerinden NEAT algoritmasını denemiş ve rakiplerine üstünlük kurmayı bir noktaya kadar başarabilmiştir. Oyunlar üzerindeki bu başarısını çeşitli problem çözme becerilerini soyutlamayı örneklendirdiği için de değerli görmektedir (Andersen 2002). Faustino J. Gomez çalışmasında sinir evriminin pratik anlamda kontrolcü olarak kullanılabilmesi amacıyla geliştirdiği yöntemden bahsetmiş ve sinir evriminin pratik kontrolcü olarak kullanılabileceğini de göstermiştir (Gomez 2003).

Kenneth O. Stanley ve diğerleri NERO oyununu anlattıkları çalışmalarında gerçek zamanlı eğitilen bir robot ekinin yer aldığı yeni bir oyun türü tanıtmışlardır (Stanley ve diğ. 2005).

### 3.3 Evrişimli Sinir Ağları ile EEG Sınıflandırması

EEG sinyal sınıflandırma problemi üzerinden tarama yapıldığında çokça CNN ve CNN varyasyonu ile karşılaşılmasına karşın, yerel maksimumdan yapısı gereği çıkabilen NEAT algoritmasıyla karşılaşılmamıştır.

Yongjie Xu ve diğerleri, EEG verisinin otizm spektrumuna göre sınıflandırmasını CNN temelli bir yaklaşım ile denemiş ve %81.08 doğruluk payı ile sınıflandırmayı başarmıştır (Xu ve diğ. 2024). Acharya ve diğerleri EEG sınıflandırma probleminde ilk defa CNN yaklaşımını kullandıkları çalışmalarında doğruluk, özgüllük ve duyarlılığı sırasıyla %88,67, %90,00 ve %95,00 olarak bulmuşlardır (Acharya ve diğ. 2018). San-Segundo, Ruben ve diğerleri EEG sınıflandırma sırasında Fourier transformunu denedikleri yayında nöbetli ve nöbetsiz durumları sınıflandırmada %99,5 a kadar bir doğruluk elde ettiklerini söylemiştir (San-Segundo ve diğ. 2019). Takahashi, Hirokazu ve diğerleri oto-enkoder ile CNN birleştirdikleri çalışmada anormal ancak nöbet içermeyen örneklerin tespitini yapabildiklerini söylemiş ve oto-enkoder kullanmanın faydalı olduğuna vurgu yapmıştır (Takahashi ve diğ. 2020). Rahul Sharma ve diğerleri derin nöral ağlar ile nöbet sınıflandırması yaptıkları çalışmalarında doğruluk değerini %100'e kadar taşıdıklarını söylemişlerdir (Sharma ve diğ. 2020). Zuochen Wei ve diğerleri CHB-MIT verisi üzerinde yaptıkları iyileştirmeli CNN ağı çalışmasında iyileştirmeler olmaksızın %70,68 duyarlılık ve %92,30 özgüllüğe ulaştığını söylemiştir (Wei ve diğ. 2019). Abdulnasir Yıldız ve diğerleri Fourier, Wavelet ve Hilbert Huang dönüşümü kullandıkları 4 örnek uzayda yaptıkları araştırmada örnek uzay sayısı 1024'ü geçtiğinde %100 doğruluk elde ettiklerini söylemişlerdir (Yıldız ve diğ. 2021). Mingkan Shen ve diğerleri CHB-MIT veri tabanı kullanılarak yaptıkları araştırmada istatistiksel momentler ve spektral bant gücü kullanarak CNN ile %97,57 doğruluk, %98,90 duyarlılık, %2,13 yanlış pozitiflik oranı ve 10,46 saniyelik gecikme elde etmiştir (Shen ve diğ. 2023). Yazan Zaid ve diğerleri ön işleme yaparak EEG nöbet sınıflandırması yaptıkları çalışmalarında en iyi

sonuçları orijinal veri ve FFT'yi birlikte kullandıklarında elde ettiklerini söylemiştir (Zaid ve diğ. 2023). Lu ve Triesch epilepsi ve EEG sinyal sınıflandırması üzerine yaptıkları çalışmada modern derin öğrenme yöntemlerinin en gelişmiş performansa ham veri ile ulaşabilecekleri sonucuna ulaşmışlardır (Lu ve Triesch 2019). Rashed-Al-Mahfuz ve diğerleri nöbet tespiti için yaptıkları çalışmada %99,21 doğruluk değerine ulaşabilmiş bir CNN EEG sınıflandırıcı önermektedir (Rashed-Al-Mahfuz ve diğ. 2021).

## 4. YÖNTEM

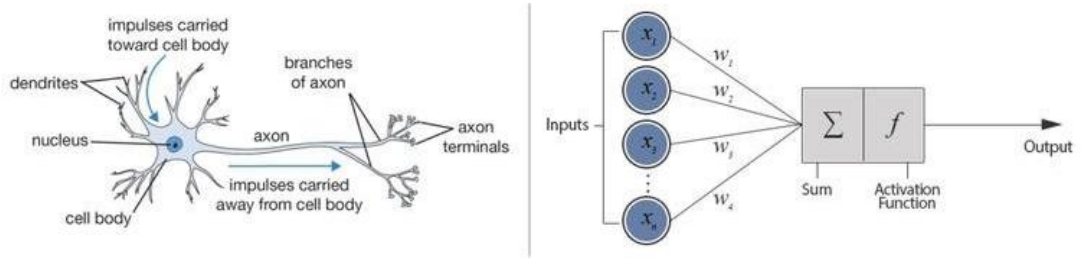
### 4.1 Yapay Zeka

Günümüzde veri uzayının çok yüksek sayılara ulaştığı problemlerin çözülmesinde, problemin çözümü için soruyu değiştiren bir yaklaşım getirerek asıl problemi çözmektense "problemi çözebilecek bağıntıları bulmak" problemini çözmeye odaklanmış adeta problemi çözmeye çalışan insan davranışını modellemekte kullanılan bir bilim dalıdır. Tanımın kendisi oldukça geniş ve insan öğrenimini, insan çözümlemesini özetle insan zekasını taklit etmeye çalışan her türlü sistemi kapsar.

### 4.2 Yapay Sinir Ağları

Yapay Sinir Ağları (YSA), biyolojik beynin çalışma mekanizmasından esinlenerek geliştirilmiş, karmaşık problemlerin çözümüne yönelik bir yapay zeka modelidir. Matematiksel bir altyapı ve hesaplamalı algoritmalar üzerine kurulu olan bu popüler model, özellikle sınıflandırma, tahmin, veri kümesinden anlam çıkarma ve öğrenme problemlerinde oldukça etkilidir. Biyolojik nöron yapısından esinlenerek tasarlanmış olmasına rağmen, yapay bir nörona veya daha karmaşık yapılar olan yapay sinir ağlarında işleyiş mekanizmaları açısından farklılıklar veya benzerlikler mevcuttur. Tıpkı biyolojik sinirlerin aksiyon potansiyeli, akson ve dendritleri olduğu gibi yapay sinirlerin de aktivasyon fonksiyonları, çıkış bağlantıları ve giriş bağlantıları vardır. Yapay bir nörona bu şeklindeki bir yapı, öğrenme, giriş ve çıkış bağlantıları üzerindeki katsayılar üzerinde gerçekleşmekte ve aksiyon potansiyeli de aktivasyon fonksiyonu ile sağlanmaktadır. Bilgi işleme sürecinde biyolojik bir nörona dendritler aracılığıyla sinyaller alınır ve aksonlar aracılığıyla sinyaller iletilir. Sinapslar arasındaki bağlantının şiddeti, nöronların ilettiği sinyallerin etkisini belirler. Belirli bir eşik değerine ulaşıldığında bir aksiyon potansiyeli oluşur ve sinyal diğer nöronları uyarır. Yapay bir nörona bir giriş katmanından veri alınır ve bir çıktı katmanına bilgi gönderilir. Ağırlıklar, girişlerin derecesini belirler ve modelin öğrenme süreci boyunca optimize edilir. Aktivasyon fonksiyonu girişleri belirli bir eşik değerine göre işleyerek

ağ çıktısını üretir. Katmanlı bir yapıya sahip olan YSA'da giriş katmanı, verilerin modele girdiği katmandır. Gizli katmanlar, verinin işlenip anlamlı hale getirildiği katmanlardır. Derin öğrenme modelleri pek çok gizli katman ile oluşturulmuş karmaşık ve güçlü birer YSA modelleridir. Çıkış katmanı ise işlenen verinin sonucunun temsil edildiği katmandır. Biyolojik bir nöron yapısı ile yapay nöron modeli arasındaki benzerlik ilişkisi Şekil 4.1'de görülmektedir (Mwandau 2018).



**Şekil 4.1:** Biyolojik nöron ile yapay nöron arasındaki ilişki

### 4.3 Sinir Evrimi Yaklaşımı

Evrimsel hesaplama, karmaşık problemleri çözmek için evrimsel yöntemlerin hesaplama süreci olarak kullanıldığı bir yaklaşımdır (De Jong 2017). Evrimsel algoritmalar, evrimsel makine öğrenimi yöntemleri içerisinde en çok çalışılan algoritmalar olduğundan sinir ağlarının farklı yönlerini ortaya çıkarmak için sıklıkla tercih edilmektedir. Bu yapı, literatürde nöroevrim veya sinir evrimi şeklinde bilinmektedir (Floreano ve diğ. 2008). Evrimsel algoritmalar, bir sinir ağı yapısının çoklu olasılıklarını yüksek boyutlu arama uzaylarında makul çözümlerle bulmak için son derece uygundur. Bununla birlikte, sinir evrimi yaklaşımı, stokastik gradyan iniş gibi daha geleneksel yaklaşımlarda bulunmayan önemli yeteneklere sahiptir. Önceden yapılandırılmış ağ topolojilerine ağırlık atama sürecinin ötesinde, ideal hiperparametrelerin, yapısal parçaların ve/veya öğrenme kurallarının kendisinin aranmasını içeren bir otomasyon düzeyi sunar (De Jong 2017).

Sinir evrimi yaklaşımının çeşitli problemler üzerinde başarıyla uygulanmasına bir örnek NEAT algoritmasının kullanımıdır. NEAT'de mutasyonlar ve çaprazlamalar ile farklı sinir ağlarının benzer parçaları ilişkilendirilerek sinir ağları oluşturulmaktadır. Aynı zamanda, olası ağ topolojilerinin daha geniş popülasyonlar yerine öncelikle

kendi içinde rekabet etmeleri sağlanır. Böylece, giderek daha karmaşıklaşan topolojilerin evrimini mümkün kılan bir çeşitlilik oluşturulur. NEAT, daha küçük yapıların öncelikli olduğu durumlar için etkili bir yöntemdir. Bunun yanı sıra, büyük veri setlerine sahip problemlerde derin öğrenme modelleri ile kıyaslandığında, orijinal NEAT algoritması tarafından üretilen ağlar, boyut ve etkinlik açısından geride kalabilmektedir.

Sinir ağları ve türlerinin sıkça kullanılması, sinir evrimi alanında da daha etkili çözümler üretmek için yeni teknikler denenmesine yol açmaktadır. Geleneksel sinir evrimi algoritmalarında, değişken topoloji gereksinimlerine ve olasılıksal farklılıklara uyum sağlamak için pek çok uyarılama yapılmıştır (Stanley 2019). Bu uyarlamalar arasında, yüksek boyutlu veri kümelerinde ağ oluşturma, özellik seçimi, gen ifade desenlerinin tanımlanması, pekiştirmeli öğrenme görevleri, veri artırma için öğrenme politikaları sayılabilir (De Jong 2017). Zaman içerisinde NEAT algoritmasının da HyperNEAT (Stanley 2019), DeepNEAT ve CoDeepNEAT (Miikkulainen 2024) olmak üzere çeşitli varyasyonları önerilmiştir.

Literatürde daha önce NEAT algoritmasının EEG sinyallerini sınıflandırma ve epilepsi nöbetlerini saptama uygulamalarına uygun olup olmadığı veya basit donanım ortamlarında kullanılıp kullanılamayacağı ile ilgili bir araştırmaya rastlanmamıştır. Bu nedenle, bu tezde, bahsedilen boşluğu gidermeye yönelik bir uygulama gerçekleştirilmiştir. Sinir evrimi yaklaşımı ile bahsedilen EEG sinyallerini sınıflandırma ve epilepsi nöbetlerini saptama probleminin testine yönelik doğrulamaların yapılması, algorithmadaki farklı çaprazlama ve mutasyon işlemlerinin olası kombinasyonlar için denenmesi ve gerekli iyileştirmelerin belirlenmesi sağlanmaya çalışılmıştır.

#### **4.4 Sinir Evrimi Algoritmaları**

Yapay zekanın bir alt dalı olan yapay sinir ağlarının geliştirilmesi hakkındaki önemli çalışmalardan bir tanesi de sinir-evrim olarak anılan evrimsel hesaplama yöntemi ile yapay sinir ağı geliştirilmesi amaçlanan çalışmalardır. Sinir evrimi yaklaşımları, sinir ağlarının evrimsel algoritmalar kullanılarak oluşturulması ve iyileştirilmesine dayanmaktadır (Stanley 2019). Geleneksel olarak nöron ağırlıklarının

evrim yoluyla oluşturulmasıyla ilgilenir. Buna ek olarak, aktivasyon fonksiyonlarının, hiperparametrelerin, mimarilerin (katman başına düşen nöron sayısı, katman sayısı ve hangi katmanların birbirine bağlanacağı) ve hatta öğrenme kurallarının kendisinin öğrenilmesi gibi unsurlarla da yakından ilgilidir. Son zamanlarda, özellikle takviyeli öğrenme problemlerinde genetik algoritmalar ile yapay olarak evrim sürecine tabi bırakılması popülerdir (Stanley ve Miikkulainen 2002). Sinir evrimi alanındaki algoritmalarından bazıları şu şekilde özetlenebilir.

Doğrudan Politika Araması (DPA) yoluyla sinir evrim yönteminde sinir ağı doğrudan politika uzayında evrim sürecine tabi tutulur. DPA algoritmaya ait sözde-kod şu şekildedir (Salimans ve diğ. 2017):

- Rastgele politikalarla sinir ağlarının popülasyonunu tanımla
- Her politikanın uygunluğunu değerlendir
- Durdurma kriteri sağlanmadığı sürece:
  - En iyi performans gösteren politikaları seç
  - Seçilen politikalara mutasyon uygulayarak yeni politikalar oluştur
  - Yeni politikaların uygunluğunu değerlendir
  - En kötü performans gösteren politikaları yeni politikalarla değiştir

Kovaryans Matrisi Uyarlama Evrim (KMUE) stratejisi yöntemi içerisinde optimize edilen parametrelerin dağılımını evrimleştirir ve sürekli optimizasyon problemlerinde kullanılmaktadır. KMUE algoritmasına ait sözde-kod aşağıdaki gibidir (Hansen 2001):

- Rastgele politikalarla sinir ağlarının popülasyonunu başlat
- Her politikanın uygunluğunu değerlendir
- Karşılanmayan kriteri durdururken:
  - En iyi performans gösteren politikaları seç
  - Seçilen politikalara mutasyon uygulayarak yeni politikalar oluştur
  - Yeni politikaların uygunluğunu değerlendir
  - En kötü performans gösteren politikaları yeni politikalarla değiştir

Sinirsel Topolojilerin Evrensel Edinimi (STEE) yönteminde sinir ağlarının hem ağırlıkları hem de topolojileri aşamalı olarak geliştirilerek ortaya evrilmiş bir sinir ağı çıkartılır. STEE için sözde-kod aşağıdaki gibidir (Hausknecht ve diğ. 2014):

1. Minimal sinir ağlarından oluşan bir popülasyonu başlat (genellikle tek katmanlı bir ağ)
2. Sonlandırma kriteri karşılanmadığında şunları yap:
  - a. Popülasyondaki her ağın uygunluğunu değerlendir
  - b. Popülasyonun bir alt kümesini uygunluğa göre ebeveyn olarak seç
  - c. Alt kümedeki her ebeveyn için:
    - i. Bir yavru oluşturmak için ebeveyn ağını klonla
    - ii. Yavruya yapısal mutasyon uygula:
      1. Belirli bir olasılıkla ağa yeni bir düğüm ekle
      2. Belirli bir olasılıkla mevcut düğümler arasında yeni bir bağlantı ekle
    - iii. Yavruya ağırlık mutasyonu uygula:
      1. Mevcut bağlantıların ağırlıklarını ayarla
  - d. Yeni yavrunun uygunluğunu değerlendir
  - e. Popülasyondaki en az uygun ağları yeni yavruyla değiştir
3. Bitir.
4. En iyi performans gösteren ağı döndür

Kooperatif Sinaps Sinir Evrimi (KSSE) yönteminde sabit bir topolojide sinaptik ağırlıkları evrime tabi tutarak çalışılır. KSSE için sözde-kod şu şekilde verilebilir (Gomez 2003):

1. Sinir ağındaki her sinaps (bağlantı) için bir tane olmak üzere N alt popülasyonu başlat.
2. Sonlandırma kriteri karşılanmadığında şunları yap:
  - a. Her alt popülasyondaki her birey için:
    - i. Bireyin sinaps ağırlıklarını diğer alt popülasyonlardakilerle birleştirerek eksiksiz bir sinir ağı oluştur
  - b. Her eksiksiz ağın uygunluğunu değerlendir
  - c. Her alt popülasyondan, katkıda bulundukları ağların uygunluğuna göre ebeveynleri seç

- d. Her alt popülasyondaki seçili ebeveynlere mutasyon uygula
    - i. Seçilen bireylerin sinaps ağırlıklarını mutasyona tabi tut
  - e. Her alt popülasyondaki en az uygun bireyleri mutasyona uğramış yavrularla değiştir
3. Bitir
  4. En iyi performans gösteren ağı döndür

#### 4.4.1 NEAT Algoritması

Sinir ağı topolojisi oluşturmak için tasarlanan NEAT, genetik bir algoritma yapısı kullanarak başlangıçta küçük ağlar oluşturur. Bu ağlar, algoritma temelindeki nesiller boyunca nöronlar ve bağlantılar ekleyerek ağırlıklarını ayarlar. Diğer sinir evrimi algoritmalarından farklı olarak NEAT, yalnızca ağ performansı aktif olarak etkileyen nöronları veya bağlantıları eklemeye odaklanarak evrilir. NEAT'te mutasyon süreci, düğümlerin ve bağlantıların eklenmesi veya mevcut bağlantıların devre dışı bırakılması ile gerçekleşir. Bu şekilde, ağın işlevselliğini etkileyen değişikliklerden kaçınır. Çaprazlama süreci ise daha karmaşıktır. Çaprazlama esnasında genetik bilginin değişimi ağların düzgün çalışmamasına neden olabilir. Algoritma, ortak yapıları basit bir şekilde tanıyabilir ve genetik bilgiyi kusurlu ağlar oluşturmadan değiştirebilir. Çaprazlama işleminden önce NEAT, seçilen ağların belirli bir dereceye kadar uyumlu olduğundan emin olmak için çözüm popülasyonuna bir türleşme (speciation) tekniği uygular (De Jong 2017). Popülasyon benzerliklerine göre farklı türlere ayrılır. Böylece büyük bir popülasyon yerine öncelikli olarak kendi içinde bir rekabet yoluna gidilir. Bu sayede, farklı ağ topolojileri kendi hızlarında evrimleşme şansı bulur ve erken nesillerde daha iyi sonuçlar elde eden, hızlı yakınsayan ağlar tarafından anında yerlerinden edilmezler.

Geliştirilen topolojiler yoluyla sinir evrimi yaklaşımının diğer çalışmalardan en önemli farkı şudur: diğer çalışmalarda genel olarak tam bağlı bir ağdan bağlantıların zayıflatılması yolu izlenirken NEAT yaklaşımında tek bir sinir hücresinden çok hücreli yapıya bağlantılar kurularak büyümekte olan bir yapıdan bahsedilir. Bu sayede gerektiği ölçüde büyüeyebilen bir yapay zekâ ile karşılaşılır (Stanley ve Miikkulainen 2002). Geliştirilen topolojiler yoluyla sinir evrimi algoritmasında sinir ağı

topolojilerini de evrimleştiren popüler bir sinir evrim yaklaşımıdır (Papavasileiou ve diğ. 2021). NEAT algoritmasına ait genel sözde-kod prosedürü şu şekildedir:

En az yapıya sahip sinir ağlarının popülasyonunu başlat

- Popülasyondaki her ağın uygunluğunu değerlendir
- Kriter karşılanmadığında durdurma:
  - Uygunluğa göre ebeveyn ağları seç
  - Ebeveyn ağlarına çaprazlama uygula
  - Yavru ağlara mutasyon uygula:
    - Ağırlıkları değiştir
    - Yapıyı değiştir (düğümler/bağlantılar ekle)
  - Yavru ağların uygunluğunu değerlendir
  - Popülasyondaki en az uygun bireyleri yavrularla değiştir

Bu tez konusunun araştırma içeriğinde yer alan olan NEAT algoritması için NEAT Python Kütüphanesi kullanılmıştır. Bu kütüphanede yapay zeka modelinin parametrelerini içeren bir konfigürasyon dosyası ve kütüphanenin kullanacağı uygunluk fonksiyonunun yer aldığı ana kaynak kod dosyası bulunmaktadır. NEAT algoritması evrimsel hesaplamayla üretilen bir yapay zeka ağı olduğu için içerisinde yer almakta olan bazı öncül kavramlara karar verilmesi gerekmektedir. NEAT Python kütüphanesinin kullanıcıdan özellikle istediği kavramlar; üretilen modelin giriş sayısı, üretilen modelin çıkış sayısı, bir jenerasyonda yer alacak olan popülasyon sayısı, algoritmanın kaç jenerasyon devam etmesi gerektiği ve her jenerasyon için popülasyonun tamamını değerlendirmeye yönelik bir uygunluk fonksiyonudur.

Algoritmanın içerisinde bulunan popülasyon ve jenerasyon kavramları veri setindeki örnek sayısı ve girdi sayısı ile çarpılarak harcanan zamanı etkilediği için veri setindeki örnek sayısı ve girdi sayısı arttıkça modelin öğrenme süreci de artmaktadır. Ayrıca girdilerin epilepsi ile bağlantısı olabilecek anlamlı bilgiler olması öğrenme sürecini kolaylaştıracağından veri seti üzerine çeşitli özellik çıkarım algoritmaları uygulanmıştır. Algoritma 500 jenerasyon boyunca 75 popülasyonla 8934 örnek veri üzerinde 114 girdi ile çalıştırılmıştır.

## 4.5 Evriřimli Sinir Ağları

Evriřimli sinir ağları, özellikle görüntü işleme ve büyük veri uygulamalarında son derece etkili olan yapay zekanın derin öğrenme alanında kullanılan mimarilerden biridir. Özellikle görsel veriler üzerinde çalışan CNN'ler, mimarileri sayesinde özellik çıkarımı ve sınıflandırmayı otomatikleştirmektedir. CNN ağlarının temel yapısını giriş katmanı, evriřim katmanı, havuzlama katmanı, tam bağlantı katmanı ve çıkış katmanı oluşturmaktadır. Derin öğrenmenin bu türünde girdiler resim (matris) veya zaman serisi olarak ele alınır ve evriřim katmanlarına tabi tutulur. Bu katman çekirdek veya filtre adı verilen küçük matrisler kullanarak veri üzerinde evriřim işlerini gerçekleştirir. Çekirdek (kernel) denen yapılar veri üzerinde kayan bir pencere gibi çalışır ve belirli özellikleri çıkarır. Evriřim katmanları görselin özelliklerini netleştirmekle sorumludur. Genellikle netleştirilen öznitelikleri sayıca azaltmak için evriřim katmanı, sonrasında evriřim katmanından elde edilen çıktıyı daha küçük matrislere dönüřtürmekle sorumlu havuzlama katmanlarına gönderir (Purwono ve diğ. 2022). Havuzlama katmanı özellik haritalarının boyutunu küçülterek modelin hesaplama maliyetini azaltır ve öğrenilen özelliklerin özetlenmesini sağlar. Tam bağlantı katmanı ise çıkarılan öznitelikleri kullanarak sınıflandırma yapar. Bu katman genellikle son katmandır ve çıktı boyutu sınıf sayısına göre değıřkendir. Çıkış katmanında kullanılan aktivasyon fonksiyonunun türü bunu belirler. Çıkış katmanı, girdinin hangi sınıfa ait olduđunu tahmin eder. Tasarımda, üzerinde çalışılan probleme göre evriřim katmanı ve havuzlama katmanı tekrarları farklılaşabilir.

NEAT algoritmasının tek hücreden kendini büyütmesine karşılık evriřimli sinir ağları başlangıç noktasında oldukça fazla bağlantı ile gelebilir. Kendisini geri yayılım algoritmalarıyla güncelleyen evriřimli ağlar bu yönleri ile de evrimsel hesaplama kullanan NEAT ile ayrıřmaktadır.

## 4.6 Phisionet Veri Kümesi

Bu tezde, EEG sinyal sınıflandırmasında sinir evrimi yaklaşımını incelemek üzere phisio.net sitesinden alınan ve herkese açık olarak erişilebilir durumdaki CHB-MIT Scalp EEG veri seti (Guttag 2010) kullanılmıştır. Boston Çocuk Hastanesi ile Massachusetts Teknoloji Enstitüsü (CHB-MIT) tarafından hazırlanan veri seti,

CHB'deki her hasta için nöbet ve nöbet dışı EEG kayıtlarını içermektedir. Veri seti, yaşları 3-22 arasında değişen 5 erkek ve yaşları 1,5-19 arasında değişen 17 kadın olmak üzere toplam 23 hastadan oluşmaktadır. Her hastaya ait Avrupa veri formatında (.edf) kaydedilmiş birden fazla nöbet ve nöbet dışı kayıt dosyası bulunmaktadır. Birincil veri setleri, beynin yüzeyine 10-20 Uluslararası Sisteme göre yerleştirilen farklı türdeki kanallar aracılığıyla elde edilen EEG sinyallerini içeren 1-B formatındadır. Veri setindeki tüm bu sinyaller, 256 Hz frekansında örneklenmiştir. Veri setinin doğrudan eğitime konu olması verimli olmayacağı için aşağıda tanımlanan öznitelikleri çıkartılarak bir veri seti oluşturulmuştur. Hangi özelliklerin neden üzerinde durulduğu devam eden alt bölüm başlıklarında sunulmuştur.

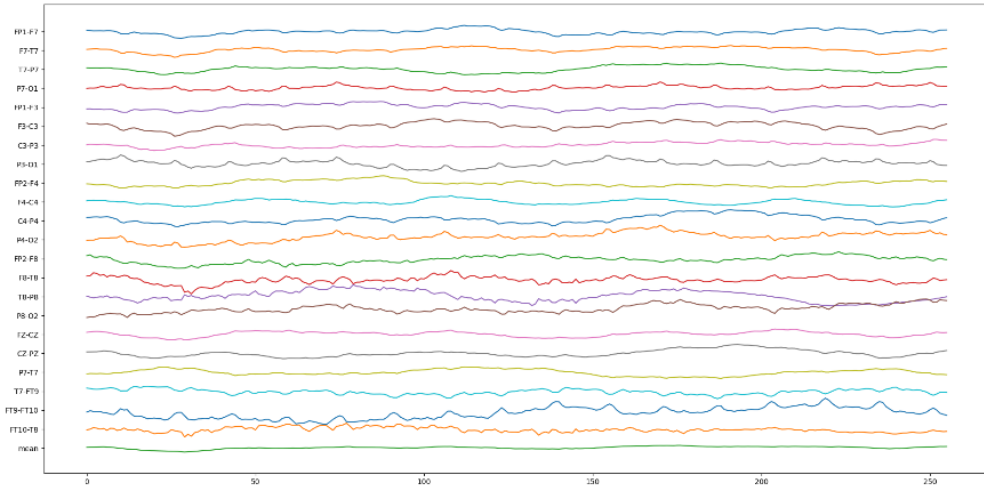
#### 4.7 Veri Ön İşleme Süreci

Bir EEG kaydının bir saniyesi özellikleri çıkartılacak ana örnek olarak değerlendirilmiş ve 1 saniyenin her kanalının özellikleri çıkartıldıktan sonra ortalamaları alınarak “epilepsi içermeyen” “epilepsi içeren” kayıtlar üzerinde işlemler uygulanmıştır.

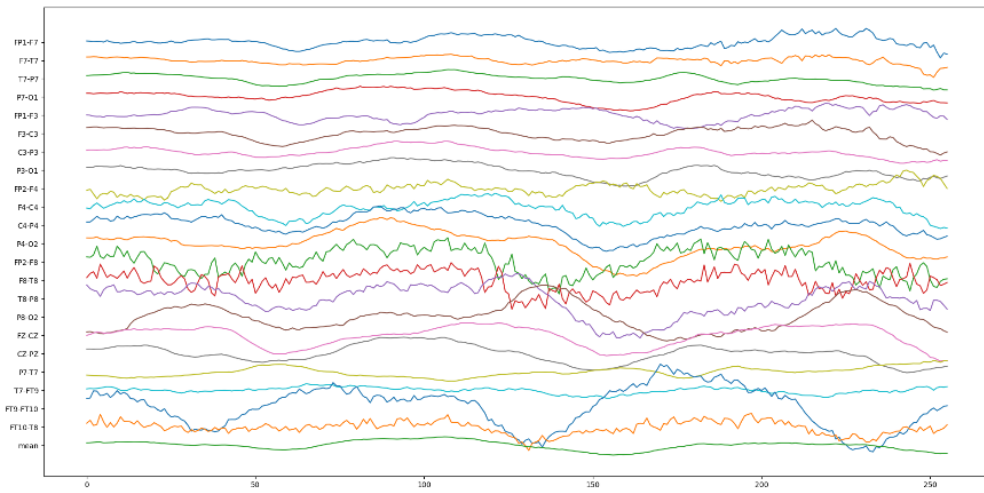
#### 4.8 Öznitelik Çıkarımı

**Kuvvet özniteliği:** Kuvvet değeri zaman sinyalinin normalizasyon katsayısıdır. Sadece bu bile bazı durumlarda anlamlı bir farklılık olabileceğine işaret edebilmektedir. Ancak farklılığın gürültü kaynaklı mı yoksa patolojik bir sebepten mi olduğu kesin değildir.

**Nöbet bilgisi özniteliği:** Bu özellik veri setinin kendisinden gelmekte olan 1 saniyelik verinin nöbet içerip içermediği bilgisidir. Sistemde çıktı olarak kullanılacaktır. Nöbet verisi bulunan ve nöbet verisi bulunmayan EEG sinyal örnekleri Şekil 4.2 ve Şekil 4.3’de sunulmuştur.

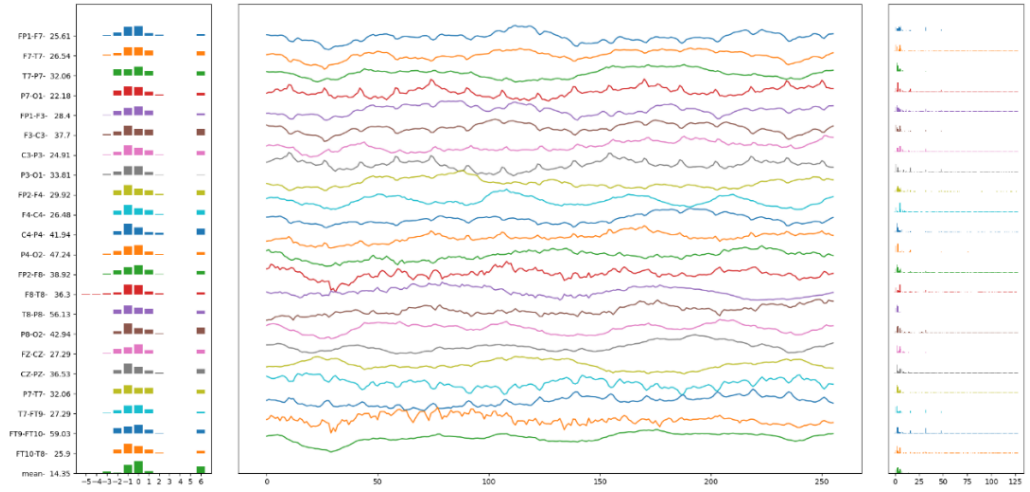


**Şekil 4.2:** Nöbet içermeyen işlenmemiş EEG verisi örneği



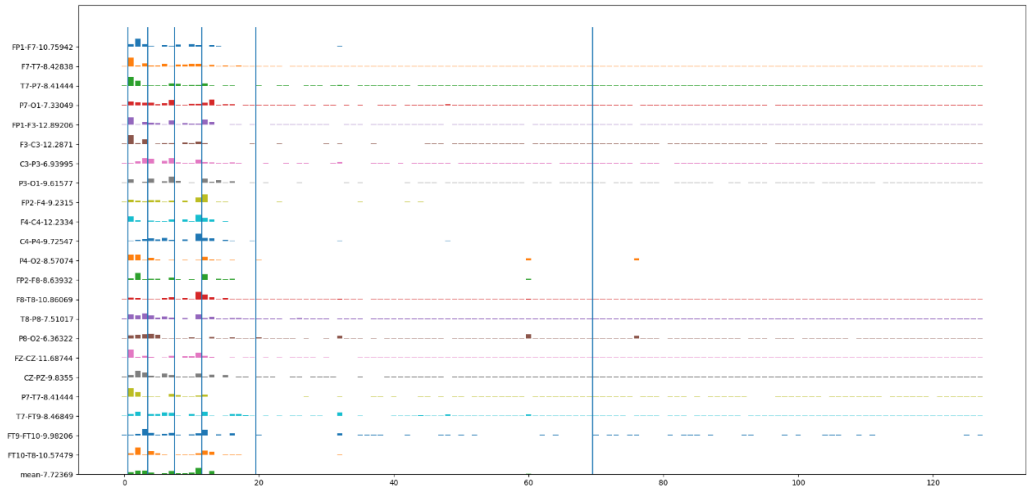
**Şekil 4.3:** Nöbet içeren işlenmemiş EEG verisi örneği

**PDF (Probability Distribution Function) farkı özneliği:** Olasılık dağılım fonksiyonu zaman sinyalinin y eksenini sınırlı sayıda bölgelere ayırarak grafiğin bölgelerde bulunma ihtimalini hesaplayan bir araçtır. Bu sayede eksi bölgedeki ve artı bölgedeki grafik şiddeti düşük çözünürlükte fark edilir hale gelmektedir negatif bölgelerdeki değerleri -1 ile çarparak tüm bölgelerdeki değerlerin toplanması da grafik olasılıkları farkını verir. Tam dengeli bir sinüs dalgasında bu değer 0 çıkması beklenecektir ve grafik dengeli olmaktan uzaklaştıkça bu değer artış gösterecektir. Bu fark patoloji tespiti konusunda anlamlı bir farklılık olabilir. Kanalların PDF değerlerine ait bir EEG sinyal örneği Şekil 4.4’de görülmektedir.



Şekil 4.4 Kanalların PDF değerleri

**Frekans değerleri özneteliği:** Bu özellikler sırasıyla Delta: [1-4), Teta: [4-8), Alpha: [8-13), Beta: [13-21), Spike: [21-71), High: [71-128) Hertz aralıklarındaki frekansların toplam değerlerini vermektedir. Kanalların Fourier transform değerlerine ait bir EEG sinyal örneği Şekil 4.5'te görülmektedir.



Şekil 4.5: Kanalların Fourier transform değerleri

#### 4.9 Pencere Bazlı Kanal Ortalamaları ile Eğitim (Senaryo-1)

Yukarıda bahsedilen toplam veri seti kullanılarak oluşturulmuş olan birinci varyasyonda 22 kanalın sekizer özelliği önce belirlenmiş sonra da 22 kanala ortalaması alınarak toplamda 8 girdi değerine yer verilmiştir.

#### 4.10 Pencere Bazlı Tüm Kanal Değerleri ile Eğitim (Senaryo 2)

Yukarıda bahsedilen toplam veri seti kullanılarak oluşturulmuş olan ikinci varyasyon da ise 22 kanalın sekizer özelliği önce belirlenmiş sonra da her örnekte geçmekte olan kanalların özellikleri yan yana gelecek şekilde bir csv dosyasına kolon olarak yerleştirilmiştir.

#### 4.11 Hiperparametre Optimizasyonu

NEAT'te dahil olmak üzere derin öğrenme algoritmalarında hiperparametre adı verilen konfigürasyon seçenekleri bulunmaktadır. Çeşitli hiperparametreler değiştirilerek elde edilen yeni konfigürasyonlarla probleme daha fazla uyum sağlayan modeller üretilmesi mümkün olabilir. Bu bölümde seçili 4 parametre için 4'er farklı, 1 adet ise parametrelerin medyan değerleri kullanılarak oluşturulmuş kontrol konfigürasyonu olmak üzere 21 farklı konfigürasyon hazırlanmış ve çalıştırılmıştır.

**Popülasyon Büyüklüğü:** Bu parametre ile algoritmanın popülasyon büyüklüğü belirlenmekte olup 5 değer ile denemeler gerçekleştirilmiştir bu değerler sırasıyla 25, 50, 100, 200 ve 300 dür. **Uyumluluk Eşiği:** Bu parametre ile tür çeşitliliğini korumak amaçlanmaktadır. Düşük değerler ile farklılıkları az olan çok sayıda tür oluşturulabilirken değer yükseldikçe çeşitlilik azalmaktadır. Bu çalışmada uyumluluk eşiği için denenen değerler sırasıyla 0.5, 1.5, 2.5, 3.5 ve 4.5 şeklindedir. **Bağlantı Ekleme Olasılığı:** Sinir ağına yeni bağlantı eklenme olasılığını belirten parametredir. Bu parametre ile sinir ağlarının karmaşılaşmasına olanak tanınır. Bağlantı ekleme olasılığı ile ilgili olarak 0.05, 0.1, 0.2, 0.3 ve 0.4 değerleri kullanılmıştır. **Düğüm Ekleme Olasılığı:** Sinir ağına yeni bir düğüm eklenme olasılığını kontrol eden parametredir. Yapısal karmaşıklığı arttırmak için önemlidir. Düğüm ekleme olasılığı için 0.03 0.1 0.15 0.2 ve 0.3 değerleri kullanılmıştır. **Ağırlık Mutasyon Gücü:** Sinir ağı mutasyona uğradığında ne kadar değişeceğini belirleyen parametredir. Gücün yükselmesi mutasyonları rastgeleliğini artırarak ağın dengesini bozabilir. Bu parametre için 0.03 0.1 0.15 0.2 ve 0.3 değerleri kullanılmıştır.

Tablo 4.1'in ilk satırında kontrol konfigürasyonu, diğer satırlarında ise birer parametresi değiştirilmiş olan diğer konfigürasyonlar yer almaktadır.

**Tablo 4.1:** Hiperparametre optimizasyonu için test edilen parametre setleri

| No | Popülasyon<br>Büyüküğü | Uyumluluk<br>Eşüğü | Baęlantı<br>Ekleme<br>Olasılıęı | Düęüm<br>Ekleme<br>Olasılıęı | Aęırlık<br>Mutasyon<br>Gücü |
|----|------------------------|--------------------|---------------------------------|------------------------------|-----------------------------|
| 1  | 100                    | 2,5                | 0,2                             | 0,15                         | 0,15                        |
| 2  | 25                     | 2,5                | 0,2                             | 0,15                         | 0,15                        |
| 3  | 50                     | 2,5                | 0,2                             | 0,15                         | 0,15                        |
| 4  | 200                    | 2,5                | 0,2                             | 0,15                         | 0,15                        |
| 5  | 300                    | 2,5                | 0,2                             | 0,15                         | 0,15                        |
| 6  | 100                    | 0,5                | 0,2                             | 0,15                         | 0,15                        |
| 7  | 100                    | 1,5                | 0,2                             | 0,15                         | 0,15                        |
| 8  | 100                    | 3,5                | 0,2                             | 0,15                         | 0,15                        |
| 9  | 100                    | 4,5                | 0,2                             | 0,15                         | 0,15                        |
| 10 | 100                    | 2,5                | 0,05                            | 0,15                         | 0,15                        |
| 11 | 100                    | 2,5                | 0,1                             | 0,15                         | 0,15                        |
| 12 | 100                    | 2,5                | 0,3                             | 0,15                         | 0,15                        |
| 13 | 100                    | 2,5                | 0,4                             | 0,15                         | 0,15                        |
| 14 | 100                    | 2,5                | 0,2                             | 0,03                         | 0,15                        |
| 15 | 100                    | 2,5                | 0,2                             | 0,1                          | 0,15                        |
| 16 | 100                    | 2,5                | 0,2                             | 0,2                          | 0,15                        |
| 17 | 100                    | 2,5                | 0,2                             | 0,3                          | 0,15                        |
| 18 | 100                    | 2,5                | 0,2                             | 0,15                         | 0,03                        |
| 19 | 100                    | 2,5                | 0,2                             | 0,15                         | 0,1                         |
| 20 | 100                    | 2,5                | 0,2                             | 0,15                         | 0,2                         |
| 21 | 100                    | 2,5                | 0,2                             | 0,15                         | 0,3                         |

#### 4.12 Karmaşıklık Matrisi

Karmaşıklık matrisi istatistiksel sınıflandırma ile çalışıldığında sınıflandırma algoritmasının ne kadar verimli olduğunu gösteren özet bir tablodur. Test sonuçları ve

incelenen örneklerin gerçek sonuçları karşılaştırıldığında ortaya çıkan sonuçları göz önüne sermektedir. Tablo 4.2’de benzetim sonuçlarının karmaşıklık matrisi üzerine gösterimine ait notasyon verilmiştir.

**Tablo 4.2:** Karmaşıklık matrisi gösterimi

| Örnek Sayısı: N | Doğru | Yanlış |
|-----------------|-------|--------|
| Pozitif         | TP    | FP     |
| Negatif         | TN    | FN     |

Gerçekte pozitif test sonucunda pozitif çıkmış değerlere ***Gerçek Pozitif (True Positive)***, gerçekte pozitif test sonucunda negatif çıkmış değerlere ***Yanlış Negatif (False Negative)***, gerçekte negatif test sonucunda negatif çıkmış değerlere ***Gerçek Negatif (True Negative)*** ve gerçekte negatif test sonucunda ise pozitif çıkmış değerlere ise ***Yanlış Pozitif (False Positive)*** denilmektedir.

## 5. UYGULAMA SONUÇLARI

Benzetim çalışmaları, belirtilen senaryolar dahilinde olayları anlamlı şekilde sonuçlandırmayı amaçlamaktadır. Bu amaçla karmaşıklık matrisi kullanılmaktadır. Karmaşıklık matrisinin temelinde dört önemli kavram vardır gerçek pozitif (TP), yapay zekanın olumlu sonuç bulduğunu ve örneğin gerçekten de olumlu olduğunu ifade eder. Gerçek negatif (TN) bu kavram yapay zekanın negatif sonuç bulduğunu ve örneğin gerçekten negatif olduğunu ifade etmektedir. Hatalı pozitif (FP) yapay zekanın pozitif sonuç bulduğunu ancak örneğin aslında negatif olduğunu dolayısıyla yapay zekanın pozitif çıkarımında hatalı olduğunu ifade eder ve benzer şekilde hatalı negatif (FN) ile de yapay zekanın negatif sonuç bulduğunu ve örneğin pozitif olması dolayısıyla yapay zekanın hatalı sonuç verdiğini ifade eder. Bu tez çalışmasında, benzetim sonuçları karmaşıklık matrisinin yorumlanması sırasında beş farklı metrik üzerinden incelenmiştir. Bu metrik değerlerinden doğruluk değeri,

$$Doğruluk = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

şeklindedir. Bu değer, yapay zekanın doğru sonuçlarının toplama oranıdır. Bir diğer metrik olan kesinlik değeri,

$$Kesinlik = \frac{TP}{FP + TP} \quad (2)$$

şeklinde ifade edilir. Bu metrik, yapay zekanın gerçek pozitif tahminlerinin tüm pozitif tahminlerine oranıdır. Hassasiyet/Duyarlılık metrik değeri,

$$Hassasiyet/Duyarlılık = \frac{TP}{FN + TP} \quad (3)$$

şeklindedir. Bu değer, yapay zekanın pozitif örnek uzayını ne kadar kapsayabildiğini göstermektedir. Seçicilik metriği,

$$Seçicilik = \frac{TN}{TN + FP} \quad (4)$$

şeklinde ifade edilir. Bu metrik, yapay zeka modelinin negatif örnek uzayını ne kadar kapsayabildiği göstermektedir. F1 skoru,

$$F1 \text{ skoru} = \frac{2 * Kesinlik * Hassasiyet}{Kesinlik + Hassasiyet} \quad (5)$$

olarak ifade edilir ve bu metrik, kesinlik ve hassasiyet değerleri arasındaki dengeyi gösterir.

### 5.1. Evrişimli Sinir Ağları Temelli EEG Sinyali Sınıflandırma Sonuçları

Benzetim çalışmalarında ilk senaryo olarak hazırlanan 1. veri setinde sadece 8 öznitelik ön plana çıkmaktadır. Bu 8 özellik ile model öğrenimi gerçekleştirildiğinde CNN algoritması % 0.85844 doğruluk değeri ile eğitimi tamamlamaktadır. Değerler gözlemlendiğinde anlaşılmaktadır ki model pozitif sınıflandırmalarında % 0.7753 oranında doğru sonuç üretmiş, ancak gerçek pozitiflerin % 0.6108'ini doğru tanımlayabilmiştir. Tablo 5.1'de birinci veri seti için CNN karmaşıklık matrisi görülmektedir. Tablo 5.2'de ise ilk senaryoya ait birinci veri seti için CNN modeli ile elde edilen nümerik sonuçlar listelenmiştir.

**Tablo 5.1:** 1. Veri seti CNN karmaşıklık matrisi

| Örnek Sayısı: 9332 | Doğru | Yanlış |
|--------------------|-------|--------|
| Pozitif            | 1425  | 413    |
| Negatif            | 6586  | 908    |

**Tablo 5.2:** 1. Veri seti CNN ek metrikler

|            |         |
|------------|---------|
| Doğruluk   | 0.85844 |
| Kesinlik   | 0.7753  |
| Hassasiyet | 0.6108  |
| Seçicilik  | 0.94099 |
| F1 skoru   | 0.68329 |

Benzetim çalışmalarında ikinci senaryo olarak hazırlanan 2. veri setinde sadece 114 öznitelik ön plana çıkmaktadır. Bu 114 özellik ile birlikte öğrenim gerçekleştirildiğinde CNN algoritması % 0.91795 doğruluk değeri ile eğitimi tamamlamaktadır. Değerler gözlemlendiğinde anlaşılmaktadır ki model pozitif

sınıflandırmalarında %0.87394 oranında doğru olduğu, ancak gerçek pozitiflerin % 0.78514'ini doğru tanımlayabildiği görülmektedir. Tablo 5.3'te ikinci senaryodaki veri seti için CNN karmaşıklık matrisi görülmektedir. Tablo 5.4'de ise ilk senaryodaki veri seti için CNN modeli ile elde edilen nümerik sonuçlar listelenmiştir.

**Tablo 5.3:** 2. Veri seti CNN karmaşıklık matrisi

| Örnek Sayısı: 8934 | Doğru | Yanlış |
|--------------------|-------|--------|
| Pozitif            | 1754  | 253    |
| Negatif            | 6447  | 480    |

**Tablo 5.4:** 2. Veri seti CNN karmaşıklık matrisi

|            |         |
|------------|---------|
| Doğruluk   | 0.91795 |
| Kesinlik   | 0.87394 |
| Hassasiyet | 0.78514 |
| Seçicilik  | 0.96224 |
| F1 skoru   | 0.82716 |

## 5.2. NEAT Temelli EEG Sinyali Sınıflandırma Sonuçları

Benzetim çalışmalarında ilk senaryo olarak hazırlanan 1. veri setinde 8 özellik ile birlikte öğrenim gerçekleştirildiğinde NEAT algoritması % 0.76618 doğruluk değeri ile eğitimi tamamlamaktadır. Buradan da anlaşılacağı üzere sadece 8 değer kullanıldığında iki algoritma da zorlanmaktadır, ancak NEAT algoritması CNN'e göre daha düşük doğruluk sonuçları vermektedir. Tablo 5.5'te ilk senaryodaki veri seti için NEAT karmaşıklık matrisi görülmektedir. Tablo 5.6'da ilk senaryodaki 1. veri seti için NEAT yaklaşımı ile elde edilen nümerik sonuçlar listelenmiştir.

**Tablo 5.5:** 1. Veri seti NEAT karmaşıklık matrisi

| Örnek Sayısı: 23330 | Doğru | Yanlış |
|---------------------|-------|--------|
| Pozitif             | 1563  | 1187   |
| Negatif             | 16312 | 4268   |

**Tablo 5.6:** 1. Veri seti NEAT ek metrikler

|            |         |
|------------|---------|
| Doğruluk   | 0.76618 |
| Kesinlik   | 0.56836 |
| Hassasiyet | 0.26805 |
| Seçicilik  | 0.93217 |
| F1 skoru   | 0.36429 |

Benzetim çalışmalarında ikinci senaryo olarak hazırlanan veri setinde 114 özellik ile öğrenim gerçekleştirildiğinde NEAT algoritması %0.81598 doğruluk değeri ile eğitimi tamamlamaktadır. 114 değer kullanıldığında daha iyi sonuçlar elde edilmesine karşın NEAT yaklaşımı CNN'e göre daha düşük sonuçlar vermektedir. Değerler gözlemlendiğinde anlaşılmaktadır ki model pozitif sınıflandırmalarında %0.795 oranında doğru olduğu, ancak gerçek pozitiflerin %0.35586'ını doğru tanımlayabildiği anlaşılmaktadır. Tablo 5.7'de ikinci senaryodaki veri seti için NEAT karmaşıklık matrisi görülmektedir. Tablo 5.8'de ikinci senaryodaki veri seti için NEAT yaklaşımı ile elde edilen nümerik sonuçlar listelenmiştir.

**Tablo 5.7:** 2. Veri seti NEAT karmaşıklık matrisi

| Örnek Sayısı: 8934 | Doğru | Yanlış |
|--------------------|-------|--------|
| Pozitif            | 795   | 205    |
| Negatif            | 6495  | 1439   |

**Tablo 5.8:** 2. Veri seti NEAT ek metrikler

|            |         |
|------------|---------|
| Doğruluk   | 0.81598 |
| Kesinlik   | 0.795   |
| Hassasiyet | 0.35586 |
| Seçicilik  | 0.9694  |
| F1 skoru   | 0.49165 |

2 veri seti ve 2 yapay zeka modeli ile yapılan uygulama sonuçlarında CNN algoritması NEAT algoritmasına karşı 2. veri setinin seçiciliği hariç her durumda daha iyi sonuçlar elde etmiştir. Tablo 5.9'da tüm veri setleri ve tüm modeller için

karmaşıklık matrisleri, Tablo 5.10’da ise tüm veri setleri ve tüm modeller için elde edilen elde edilen nümerik sonuçlar verilmiştir.

**Tablo 5.9:** Her iki yapay zeka modeline ait karmaşıklık matrisleri

|    | 1.Veri Seti<br>CNN | 1.Veri Seti<br>NEAT | 2.Veri Seti<br>CNN | 2.Veri Seti<br>NEAT |
|----|--------------------|---------------------|--------------------|---------------------|
| TP | 1425               | 1563                | 1754               | 795                 |
| TN | 6586               | 16312               | 6447               | 6495                |
| FP | 413                | 1187                | 253                | 205                 |
| FN | 908                | 4268                | 480                | 1439                |

**Tablo 5.10:** Her iki yapay zeka modeline ait ek metrikler

|            | 1.Veri Seti<br>CNN | 1.Veri Seti<br>NEAT | 2.Veri Seti<br>CNN | 2.Veri Seti<br>NEAT |
|------------|--------------------|---------------------|--------------------|---------------------|
| Doğruluk   | 0.85844            | 0.76618             | 0.91795            | 0.81598             |
| Kesinlik   | 0.7753             | 0.56836             | 0.87394            | 0.795               |
| Hassasiyet | 0.6108             | 0.26805             | 0.78514            | 0.35586             |
| Seçicilik  | 0.94099            | 0.93217             | 0.96224            | 0.9694              |
| F1 skoru   | 0.68329            | 0.36429             | 0.82716            | 0.49165             |

### 5.3. Hiperparametre Optimizasyon Sonuçları

Çalışma sonuçları değerlendirildiğinde NEAT algoritmasının şu anki durumuyla karmaşıklık matrisi metriklerine göre hesaplanan sonuçlarında istenilen verim elde edilememiştir. Bu durum gözetilerek NEAT parametreleri üzerinde yapılacak değişikliklerin sonuca ne derece etki edeceğini gözlemlemek için hiperparametre optimizasyonu yoluna gidilmiş ve Tablo 5.11 ve Tablo 5.12’deki karmaşıklık matrisi sonuçları ve metrik değerleri verilmiştir.

**Tablo 5.11:** Tüm konfigürasyonların karmaşıklık matrisi sonuçları

| Konfigürasyon Numarası | Çalışma Zamanı (sn.) | Örnek Sayısı | Doğru Pozitif | Doğru Negatif | Yanlış Pozitif | Yanlış Negatif |
|------------------------|----------------------|--------------|---------------|---------------|----------------|----------------|
| 1                      | 701457,1811          | 8934         | 739           | 6567          | 133            | 1495           |
| 2                      | 171226,9271          | 8934         | 793           | 6501          | 199            | 1441           |
| 3                      | 351533,5641          | 8934         | 743           | 6554          | 146            | 1491           |
| 4                      | 1469853,5614         | 8934         | 790           | 6550          | 150            | 1444           |
| 5                      | 1865279,297          | 8934         | 824           | 6519          | 181            | 1410           |
| 6                      | 970406,927           | 8934         | 536           | 6526          | 174            | 1698           |
| 7                      | 642057,4807          | 8934         | 780           | 6554          | 146            | 1454           |
| 8                      | 614984,9522          | 8934         | 766           | 6539          | 161            | 1468           |
| 9                      | 707978,2346          | 8934         | 831           | 6481          | 219            | 1403           |
| 10                     | 651107,9981          | 8934         | 806           | 6522          | 178            | 1428           |
| 11                     | 687151,9743          | 8934         | 832           | 6510          | 190            | 1402           |
| 12                     | 626436,5305          | 8934         | 797           | 6513          | 187            | 1437           |
| 13                     | 619689,3931          | 8934         | 809           | 6509          | 191            | 1425           |
| 14                     | 607869,4124          | 8934         | 785           | 6533          | 167            | 1449           |
| 15                     | 638431,0796          | 8934         | 840           | 6491          | 209            | 1394           |
| 16                     | 650238,1288          | 8934         | 836           | 6424          | 276            | 1398           |
| 17                     | 657881,7206          | 8934         | 801           | 6528          | 172            | 1433           |
| 18                     | 679063,3753          | 8934         | 837           | 6465          | 235            | 1397           |
| 19                     | 593945,9809          | 8934         | 794           | 6557          | 143            | 1440           |
| 20                     | 576316,655           | 8934         | 810           | 6502          | 198            | 1424           |
| 21                     | 673148,5837          | 8934         | 820           | 6526          | 174            | 1414           |

**Tablo 5.12:** Tüm konfigürasyonlara ait performans sonuçları

| Konfigürasyon Numarası | Doğruluk | Kesinlik | Hassasiyet | Seçicilik | F1 Skoru |
|------------------------|----------|----------|------------|-----------|----------|
| 1                      | 0,81777  | 0,84748  | 0,3308     | 0,98015   | 0,4759   |
| 2                      | 0,81643  | 0,7994   | 0,35497    | 0,9703    | 0,4916   |
| 3                      | 0,81677  | 0,83577  | 0,33259    | 0,97821   | 0,4758   |
| 4                      | 0,82158  | 0,84043  | 0,35363    | 0,97761   | 0,4978   |
| 5                      | 0,82192  | 0,8199   | 0,36885    | 0,97299   | 0,5088   |
| 6                      | 0,79046  | 0,75493  | 0,23993    | 0,97403   | 0,3641   |

|    |         |         |         |         |        |
|----|---------|---------|---------|---------|--------|
| 7  | 0,82091 | 0,84233 | 0,34915 | 0,97821 | 0,4937 |
| 8  | 0,81766 | 0,82632 | 0,34288 | 0,97597 | 0,4847 |
| 9  | 0,81845 | 0,79143 | 0,37198 | 0,96731 | 0,5061 |
| 10 | 0,82024 | 0,81911 | 0,36079 | 0,97343 | 0,5009 |
| 11 | 0,8218  | 0,81409 | 0,37243 | 0,97164 | 0,5111 |
| 12 | 0,81822 | 0,80996 | 0,35676 | 0,97209 | 0,4953 |
| 13 | 0,81912 | 0,809   | 0,36213 | 0,97149 | 0,5003 |
| 14 | 0,81912 | 0,82458 | 0,35139 | 0,97507 | 0,4928 |
| 15 | 0,82057 | 0,80076 | 0,37601 | 0,96881 | 0,5117 |
| 16 | 0,81263 | 0,7518  | 0,37422 | 0,95881 | 0,4997 |
| 17 | 0,82035 | 0,82323 | 0,35855 | 0,97433 | 0,4995 |
| 18 | 0,81733 | 0,78078 | 0,37466 | 0,96493 | 0,5064 |
| 19 | 0,82281 | 0,84739 | 0,35542 | 0,97866 | 0,5008 |
| 20 | 0,81845 | 0,80357 | 0,36258 | 0,97045 | 0,4997 |
| 21 | 0,82225 | 0,82495 | 0,36705 | 0,97403 | 0,5081 |

### 5.3.1 Kontrol Konfigürasyonu

Bu konfigürasyon için tüm parametrelerin medyan değerleri kullanılmış ve orta durumda nasıl bir sonuç alınacağı araştırılmıştır. Hesaplama süresi ~701457 saniye, 8 günden biraz daha fazla, sürmüştür. Kullanılan parametrelerden *Popülasyon Büyüklüğü* parametresi 100 birey olarak belirlenmiştir, birey sayısı arttıkça çeşitlilik ve optimumu bulma ihtimali artacaktır ancak eğitim süresi de artmaktadır. *Uyumluluk Eşiği* parametresi 2,5, *Uyumluluk Eşiği* parametresi arttıkça Türlerin sayısı azalır ve 2 bireyin aynı türden sayılması ihtimali artar bu da yerel minimumlara takılma riskini arttırır. *Bağlantı Ekleme Olasılığı* 0,2, *Bağlantı Ekleme Olasılığı* arttıkça ağın karmaşıklığı da artacaktır bu da ihtiyaç halinde daha karmaşık yapılarla çalışabilmesini sağlayabilir ancak ağın gereksiz karmaşıklığı daha yavaş çalışmaya neden olacaktır. *Düğüm Ekleme Olasılığı* 0,15, *Düğüm Ekleme Olasılığı* arttıkça ağın karmaşıklığı da artacaktır ve sistem daha fazla konuda değerlendirme yapabilecektir ve türleşmeyi destekleyecektir ancak fazla artması zaman yönünden maliyetli olacaktır. *Ağırlık Mutasyon Gücü* ise 0,15 olarak belirlenmiştir. *Ağırlık Mutasyon Gücü* arttıkça daha fazla yeni olasılık denenecek yerel minimumlara takılma riski

azalacak ancak fazla olması durumunda ise sistemde gelişigüze artarak sistemden beklenen fayda alınamayacaktır. Tablo 5.13’de karşılaştırmalarda kullanılan temel kontrol konfigürasyonuna ait hiperparametreler listelenmiştir.

**Tablo 5.13:** Kontrol konfigürasyonuna ait hiperparametreler

|                           |      |
|---------------------------|------|
| Popülasyon Büyüklüğü      | 100  |
| Uyumluluk Eşiği           | 2,5  |
| Bağlantı Ekleme Olasılığı | 0,2  |
| Düğüm Ekleme Olasılığı    | 0,15 |
| Ağırlık Mutasyon Gücü     | 0,15 |

Tezde geliştirilen sinir evrimi modeli yukarıdaki hiperparametre değerleri ile çalıştırıldığında aşağıdaki Tablo 5.14’te görülen karmaşıklık matrisi değerleri elde edilmiştir.

**Tablo 5.14:** Kontrol konfigürasyonuna ait karmaşıklık matrisi

|                   |       |        |
|-------------------|-------|--------|
| Örnek Sayısı:8934 | Doğru | Yanlış |
| Pozitif           | 739   | 133    |
| Negatif           | 6567  | 1495   |

Sonuçlar incelendiğinde doğruluk değerinin 0,81777, Kesinlik değerinin 0,84748, Seçicilik değerinin 0,98015 olduğu dolayısıyla genel performans açısından makul bir seviyede olduğu, Yanlış Pozitifler konusunda makul bir performans gösterdiği gözlenmiştir. Bununla birlikte Hassasiyet değerinin 0,3308 olduğunu ve Yanlış Negatifler konusunda modelin zayıf kaldığı görülmektedir. Sonuç olarak F1 skoru ancak 0,47585 değerinde kalabilmiştir. Temel kontrol konfigürasyonundaki hiperparametre değerleri için elde edilen metrik değerleri Tablo 5.15’te görülmektedir.

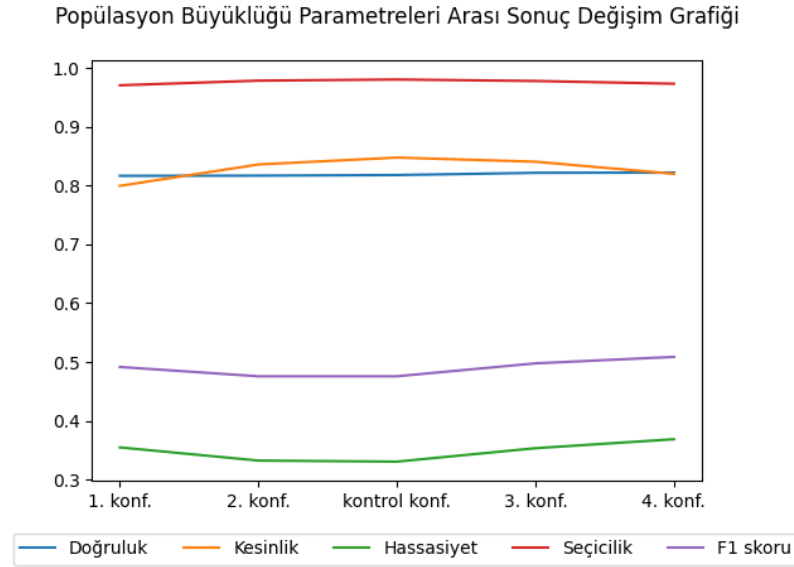
**Tablo 5.15:** Kontrol konfigürasyonu için performans sonuçları

|          |         |
|----------|---------|
| Doğruluk | 0,81777 |
|----------|---------|

|            |         |
|------------|---------|
| Kesinlik   | 0,84748 |
| Hassasiyet | 0,3308  |
| Seicilik  | 0,98015 |
| F1 Skoru   | 0,47585 |

### 5.3.2 Popölasyon Büyüklüğü (PB) Parametresi

Popölasyon boyutu modelin özüm uzayında daha geniş alanı taramasına yardımcı olacaktır ancak sayı arttıka alışma zamanı da durumdan olumsuz şekilde etkilenecektir. Bu başlık altında kullanılan diğerk değerler kontrol konfigürasyonu ile bir tutulmuş ve ilgili konfigürasyonlarda farklı popölasyon boyutları denenmiştir. Kontrol konfigürasyonu popölasyon büyüklüğü farklı konfigürasyonlarla karşılaştırıldığında F1 skoru açısından 300 birey içeren konfigürasyon diğerk değerlere göre daha iyi sonuç üretmiş olduğu Şekil 5.1 de görülebilir.



Şekil 5.1: Popölasyon büyüklüğüne göre performans değışim grafiğı

**1. Konfigürasyon:** Bu konfigürasyonda popölasyon 25 birey olarak ayarlanmış ve algoritma bu şekilde alıştırılmıştır. Diğerk hiperparametreler temel kontrol konfigürasyonundaki değerkleriyle kullanılmıştır. Algoritmanın alışma zamanı ~171226 saniye, yaklaşık 2 gün, sürmüştür ve bu yönüyle kontrol konfigürasyonun

önüne anlamlı şekilde geçmiştir, ancak Şekil 5.1 de de görüldüğü üzere diğer yönleriyle elde edilen sonuçlar kontrol konfigürasyonuna kıyasla anlamlı farklar oluşturmamıştır. 1. Konfigürasyon Hassasiyet ve F1. Skoru göz önüne alındığında kontrol konfigürasyonunun önüne geçmiş, Seçicilik ve Kesinlik metriklerinde ise kontrol konfigürasyonuna göre geride kalmıştır. Doğruluk metriğinde Popülasyon büyüklüğü ile ilgili 1. konfigürasyona ait karmaşıklık matrisi ve metrik değerleri sırasıyla Tablo 5.16 ve Tablo 5.17’de verilmiştir.

**Tablo 5.16:** PB ile ilgili 1. konfigürasyona ait karmaşıklık matrisi

| Örnek Sayısı:8934 | Doğru | Yanlış |
|-------------------|-------|--------|
| Pozitif           | 793   | 199    |
| Negatif           | 6501  | 1441   |

**Tablo 5.17:** PB ile ilgili 1. konfigürasyona ait performans sonuçları

|            |         |
|------------|---------|
| Doğruluk   | 0,81643 |
| Kesinlik   | 0,7994  |
| Hassasiyet | 0,35497 |
| Seçicilik  | 0,9703  |
| F1 Skoru   | 0,49163 |

**2. Konfigürasyon:** İkinci konfigürasyonda popülasyona verilmiş olan değer 50 olarak belirlenmiş ve bu şekilde çalıştırılmıştır. Diğer hiperparametreler temel kontrol konfigürasyonundaki değerleriyle kullanılmıştır. Eğitim sürecinin çalışma zamanı ~351533 saniye, yaklaşık 4 gün, sürmüştür ve bu yönüyle kontrol konfigürasyonun önüne anlamlı bir değerle geçmiştir ancak sonuç değerleri söz konusu olduğunda kontrol konfigürasyonu ile benzer değerlerde görülmektedir. Popülasyon büyüklüğü ile ilgili 2. konfigürasyona ait karmaşıklık matrisi ve metrik değerleri sırasıyla Tablo 5.18 ve Tablo 5.19’de verilmiştir.

**Tablo 5.18:** PB ile ilgili 2. konfigürasyona ait karmaşıklık matrisi

| Örnek Sayısı:8934 | Doğru | Yanlış |
|-------------------|-------|--------|
|-------------------|-------|--------|

|         |      |      |
|---------|------|------|
| Pozitif | 743  | 146  |
| Negatif | 6554 | 1491 |

**Tablo 5.19:** PB ile ilgili 2. konfigürasyona ait performans sonuçları

|            |         |
|------------|---------|
| Doğruluk   | 0,81677 |
| Kesinlik   | 0,83577 |
| Hassasiyet | 0,33259 |
| Seçicilik  | 0,97821 |
| F1 Skoru   | 0,47582 |

**3. Konfigürasyon:** Bu konfigürasyonda 200 birey sayısı belirlenerek kontrol konfigürasyonunun 2 katına çıkılmıştır. Diğer hiperparametreler kontrol konfigürasyonundaki değerleriyle kullanılmıştır. Sürecin çalışma zamanı ~1469853 saniye, yaklaşık 17 gün, olmuştur. Popülasyon büyüklüğü ile ilgili 3. konfigürasyona ait karmaşıklık matrisi ve metrik değerleri sırasıyla Tablo 5.20 ve Tablo 5.21’de verilmiştir.

**Tablo 5.20:** PB ile ilgili 3. konfigürasyona ait karmaşıklık matrisi

| Örnek Sayısı:8934 | Doğru | Yanlış |
|-------------------|-------|--------|
| Pozitif           | 790   | 150    |
| Negatif           | 6550  | 1444   |

**Tablo 5.21:** PB ile ilgili 3. konfigürasyona ait performans sonuçları

|            |         |
|------------|---------|
| Doğruluk   | 0,82158 |
| Kesinlik   | 0,84043 |
| Hassasiyet | 0,35363 |
| Seçicilik  | 0,97761 |
| F1 Skoru   | 0,49779 |

**4. Konfigürasyon:** Bu konfigürasyonda birey sayısı 300 olarak belirlenmiştir. Diğer hiperparametreler kontrol konfigürasyonundaki değerleriyle kullanılmıştır. Çalışma zamanı ~1865279 saniye, yaklaşık 21,5 gün, olmuştur. Bu sonuçla da görüldüğü üzere popülasyonun çalışma zamanı üzerinde doğrusal bir etkisi olduğu bulunmuştur. Popülasyon büyüklüğü ile ilgili 4. konfigürasyona ait karmaşıklık matrisi ve metrik değerleri sırasıyla Tablo 5.22 ve Tablo 5.23'te verilmiştir.

**Tablo 5.22:** PB ile ilgili 4. konfigürasyona ait karmaşıklık matrisi

| Örnek Sayısı:8934 | Doğru | Yanlış |
|-------------------|-------|--------|
| Pozitif           | 790   | 150    |
| Negatif           | 6550  | 1444   |

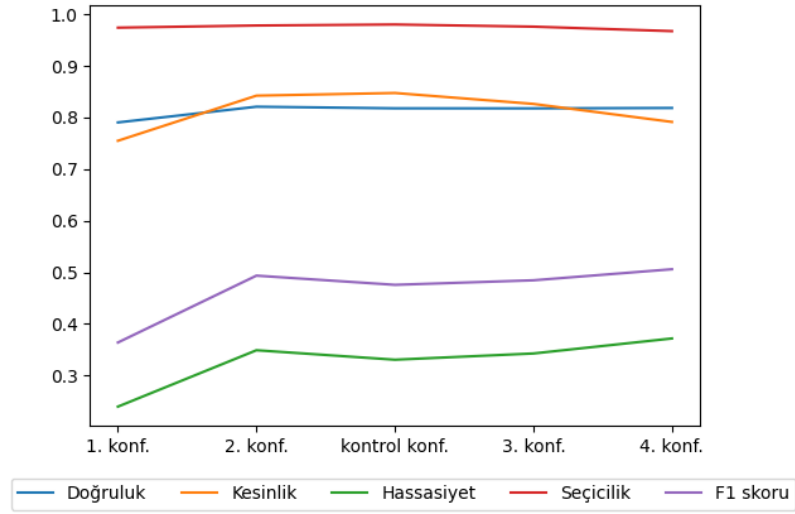
**Tablo 5.23:** PB ile ilgili 4. konfigürasyona ait performans sonuçları

|            |         |
|------------|---------|
| Doğruluk   | 0,82158 |
| Kesinlik   | 0,84043 |
| Hassasiyet | 0,35363 |
| Seçicilik  | 0,97761 |
| F1 Skoru   | 0,49779 |

### 5.3.3 Bağlantı Ekleme Olasılığı (BEO) Parametresi

Bağlantı ekleme olasılığı parametresi sistem karmaşıklığını arttırarak daha kompleks aramalar gerçekleştirilmesine imkan tanıyan bir ayar olarak karşımıza çıkmaktadır. Bu parametrenin denemeleriyle ilgili sonuçlar, diğer parametreler kontrol konfigürasyonu ile eş tutularak yapılmış ve aşağıdaki tablolarda aktarılmıştır. Kontrol konfigürasyonu ve diğerleri karşılaştırıldığında 2. Konfigürasyon F1 değeri açısından 0.51106 ile en iyi konfigürasyon olduğu Şekil 5.2 de görülebilir.

Bağlantı Ekleme Olasılığı Parametreleri Arası Sonuç Değişim Grafiği



Şekil 5.2: Bağlantı ekleme olasılığına göre performans değişim grafiği

**1. Konfigürasyon:** 1. Konfigürasyon 0,05 değeri ile bağlantı ekleme olasılığını oldukça düşük bir noktadan başlatmış olup ~651107 saniye, 7,5 gün, çalışmış ve sonuçları kontrol konfigürasyonu göz önüne alındığında F1 skoru üzerinde 0.035 iyileşme görülmüştür. Diğer hiperparametreler kontrol konfigürasyonundaki değerleriyle kullanılmıştır. BEO ile ilgili 1. konfigürasyona ait karmaşıklık matrisi ve metrik değerleri sırasıyla Tablo 5.24 ve Tablo 5.25’de verilmiştir.

**Tablo 5.24:** BEO ile ilgili 1. konfigürasyona ait karmaşıklık matrisi

| Örnek Sayısı:8934 | Doğru | Yanlış |
|-------------------|-------|--------|
| Pozitif           | 806   | 178    |
| Negatif           | 6522  | 1428   |

**Tablo 5.25:** BEO ile ilgili 1. konfigürasyona ait performans sonuçları

|            |         |
|------------|---------|
| Doğruluk   | 0,82024 |
| Kesinlik   | 0,81911 |
| Hassasiyet | 0,36079 |
| Seçicilik  | 0,97343 |
| F1 Skoru   | 0,50093 |

**2. Konfigürasyon:** 2. Konfigürasyon değeri olarak 0,1 değeri kullanılmış ve tüm konfigürasyonlar arasındaki en yüksek F1 skoru olan 0,51106 değerine ~687151 saniyede, yaklaşık 8 günde, ulaşılmıştır. Diğer hiperparametreler temel kontrol konfigürasyonundaki değerleriyle kullanılmıştır. Bağlantı ekleme olasılığı ile ilgili 2. konfigürasyona ait karmaşıklık matrisi ve metrik değerleri sırasıyla Tablo 5.26 ve Tablo 5.27’de verilmiştir.

**Tablo 5.26:** BEO ile ilgili 2. konfigürasyona ait karmaşıklık matrisi

| Örnek Sayısı:8934 | Doğru | Yanlış |
|-------------------|-------|--------|
| Pozitif           | 832   | 190    |
| Negatif           | 6510  | 1402   |

**Tablo 5.27:** BEO ile ilgili 2. konfigürasyona ait performans sonuçları

|            |         |
|------------|---------|
| Doğruluk   | 0,8218  |
| Kesinlik   | 0,81409 |
| Hassasiyet | 0,37243 |
| Seçicilik  | 0,97164 |
| F1 Skoru   | 0,51106 |

**3. Konfigürasyon:** Bu konfigürasyonda bağlantı ekleme olasılığı 0,3’e ayarlanmıştır. Süreç ~626436 saniyede, yaklaşık 7 günde, tamamlanmış olup F1 skor değerindeki iyileşme 0,01949 olarak gözlemlenmiştir. Diğer hiperparametreler kontrol konfigürasyonundaki değerleriyle kullanılmıştır. BEO ile ilgili 3. konfigürasyona ait karmaşıklık matrisi ve metrik değerleri sırasıyla Tablo 5.28 ve Tablo 5.29’te verilmiştir.

**Tablo 5.28:** BEO ile ilgili 3. konfigürasyona ait karmaşıklık matrisi

| Örnek Sayısı:8934 | Doğru | Yanlış |
|-------------------|-------|--------|
| Pozitif           | 797   | 187    |
| Negatif           | 6513  | 1437   |

**Tablo 5.29:** BEO ile ilgili 3. konfigürasyona ait performans sonuçları

|            |         |
|------------|---------|
| Doğruluk   | 0,81822 |
| Kesinlik   | 0,80996 |
| Hassasiyet | 0,35676 |
| Seçicilik  | 0,97209 |
| F1 Skoru   | 0,49534 |

**4. Konfigürasyon:** Bağlantı eklemek olasılığı parametresi için hazırlanmış olan 4. konfigürasyon için 0,4 değeri ile süreç başlatılmıştır. Kontrol konfigürasyonunun 2 katı olan bu değer ile 687151 saniyede, yaklaşık 8 günde, süreç sonuçlanmış F1 skorunda 0,02446 artış gözlenmiştir. Diğer hiperparametreler temel kontrol konfigürasyonundaki değerleriyle kullanılmıştır. Bağlantı ekleme olasılığı ile ilgili 4. konfigürasyona ait karmaşıklık matrisi ve metrik değerleri sırasıyla Tablo 5.30 ve Tablo 5.31’de verilmiştir.

**Tablo 5.30:** BEO ile ilgili 4. konfigürasyona ait karmaşıklık matrisi

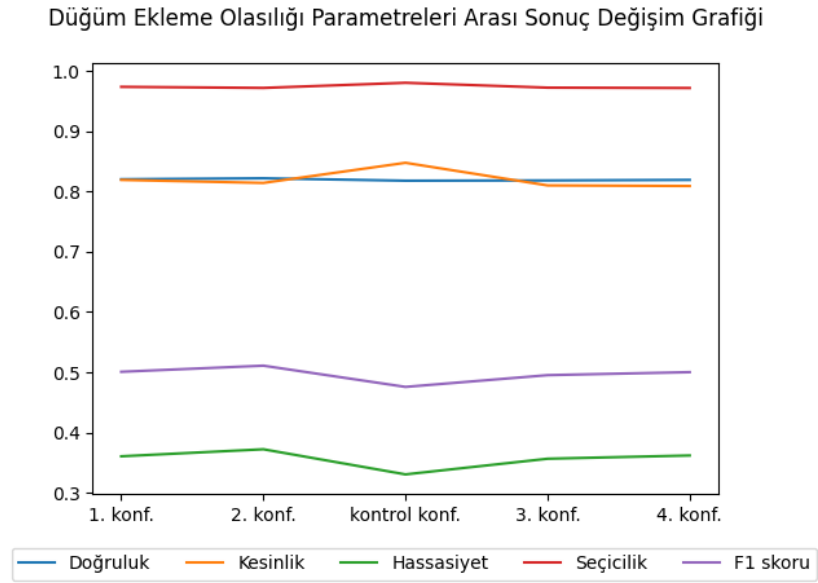
| Örnek Sayısı:8934 | Doğru | Yanlış |
|-------------------|-------|--------|
| Pozitif           | 809   | 191    |
| Negatif           | 6509  | 1425   |

**Tablo 5.31:** BEO ile ilgili 4. konfigürasyona ait performans sonuçları

|            |         |
|------------|---------|
| Doğruluk   | 0,81912 |
| Kesinlik   | 0,809   |
| Hassasiyet | 0,36213 |
| Seçicilik  | 0,97149 |
| F1 Skoru   | 0,50031 |

### 5.3.4 Düğüm Ekleme Olasılığı (DEO) Parametresi

Düğüm ekleme olasılığı parametresi ile popülasyondaki bireylerin karmaşıklıklarını arttırma ihtimali artırılır. Aşağıdaki bölümlerde kontrol parametreleri sabit tutularak denemeler gerçekleştirilmiştir. Kontrol konfigürasyonu ile karşılaştırıldığında F1 skoru açısından bakıldığında en iyi alternatifi 2. Konfigürasyon olduğu Şekil 5.3’ te görülmüştür.



Şekil 5.3: Düğüm ekleme olasılığına göre performans değişim grafiği

**1. Konfigürasyon:** İlk konfigürasyon ile 0,03 değeri denenerek ~607869 saniyede, yaklaşık 7 günde, çalışmasını tamamlayabilmiş, kontrol konfigürasyonuna göre hassasiyet değerindeki 0,03’lük artış ile dikkat çekmiştir. Diğer hiperparametreler kontrol konfigürasyonundaki değerleriyle kullanılmıştır. Düğüm ekleme olasılığı ile ilgili 1. konfigürasyona ait karmaşıklık matrisi ve metrik değerleri sırasıyla Tablo 5.32 ve Tablo 5.33’da verilmiştir.

Tablo 5.32: DEO ile ilgili 1. konfigürasyona ait karmaşıklık matrisi

| Örnek Sayısı:8934 | Doğru | Yanlış |
|-------------------|-------|--------|
| Pozitif           | 785   | 167    |
| Negatif           | 6533  | 1449   |

**Tablo 5.33:** DEO ile ilgili 1. konfigürasyona ait performans sonuçları

|            |         |
|------------|---------|
| Doğruluk   | 0,81912 |
| Kesinlik   | 0,82458 |
| Hassasiyet | 0,35139 |
| Seçicilik  | 0,97507 |
| F1 Skoru   | 0,49278 |

**2. Konfigürasyon:** 2. Konfigürasyon içerisinde düğüm ekleme olasılığı parametresi için 0.1 değeri kullanılmış ve çalışma zamanı ~638431 saniye, yaklaşık 7 gün sürmüştür. F1 skoru sonucu 0,51173 olarak bulunmuştur ve kontrol konfigürasyonundan 0,3588 daha iyi bulunmuştur. Diğer hiperparametreler temel kontrol konfigürasyonundaki değerleriyle kullanılmıştır. Düğüm ekleme olasılığı ile ilgili 2. konfigürasyona ait karmaşıklık matrisi ve metrik değerleri sırasıyla Tablo 5.34 ve Tablo 5.35’de verilmiştir.

**Tablo 5.34:** DEO ile ilgili 2. konfigürasyona ait karmaşıklık matrisi

| Örnek Sayısı:8934 | Doğru | Yanlış |
|-------------------|-------|--------|
| Pozitif           | 840   | 209    |
| Negatif           | 6491  | 1394   |

**Tablo 5.35:** DEO ile ilgili 2. konfigürasyona ait performans sonuçları

|            |         |
|------------|---------|
| Doğruluk   | 0,82057 |
| Kesinlik   | 0,80076 |
| Hassasiyet | 0,37601 |
| Seçicilik  | 0,96881 |
| F1 Skoru   | 0,51173 |

**3. Konfigürasyon:** Bu konfigürasyon ile kontrol konfigürasyonu değerinden büyük bir değer olarak 0,2 denenmiş ve ~650238 saniye, yaklaşık 7,5 gün, sürmüştür. Bu konfigürasyonun hassasiyet değerinde bir miktar iyileşme görüldüyse de kesinlik

değerindeki kötüleşme ile F1 skoru 0,4997 olarak gözlemlenmiştir. Diğer hiperparametreler temel kontrol konfigürasyonundaki değerleriyle kullanılmıştır. Düğüm ekleme olasılığı ile ilgili 3. konfigürasyona ait karmaşıklık matrisi ve metrik değerleri sırasıyla Tablo 5.36 ve Tablo 5.37’te verilmiştir.

**Tablo 5.36:** DEO ile ilgili 3. konfigürasyona ait karmaşıklık matrisi

| Örnek Sayısı:8934 | Doğru | Yanlış |
|-------------------|-------|--------|
| Pozitif           | 836   | 276    |
| Negatif           | 6424  | 1398   |

**Tablo 5.37:** DEO ile ilgili 3. konfigürasyona ait performans sonuçları

|            |         |
|------------|---------|
| Doğruluk   | 0,81263 |
| Kesinlik   | 0,7518  |
| Hassasiyet | 0,37422 |
| Seçicilik  | 0,95881 |
| F1 Skoru   | 0,4997  |

**4. Konfigürasyon:** 4. Düğüm ekleme olasılığı konfigürasyonunda kullanılan değer 0,3 olarak ayarlanmıştır. Çalışma zamanı ~657881 saniye, yaklaşık 7,6 gün, sürmüştür. Çalışma sonucunda ortaya çıkan F1 değeri 0,02368 daha iyi olmuştur. Diğer hiperparametreler temel kontrol konfigürasyonundaki değerleriyle kullanılmıştır. Düğüm ekleme olasılığı ile ilgili 4. konfigürasyona ait karmaşıklık matrisi ve metrik değerleri sırasıyla Tablo 5.38 ve Tablo 5.39’te verilmiştir.

**Tablo 5.38:** DEO ile ilgili 4. konfigürasyona ait karmaşıklık matrisi

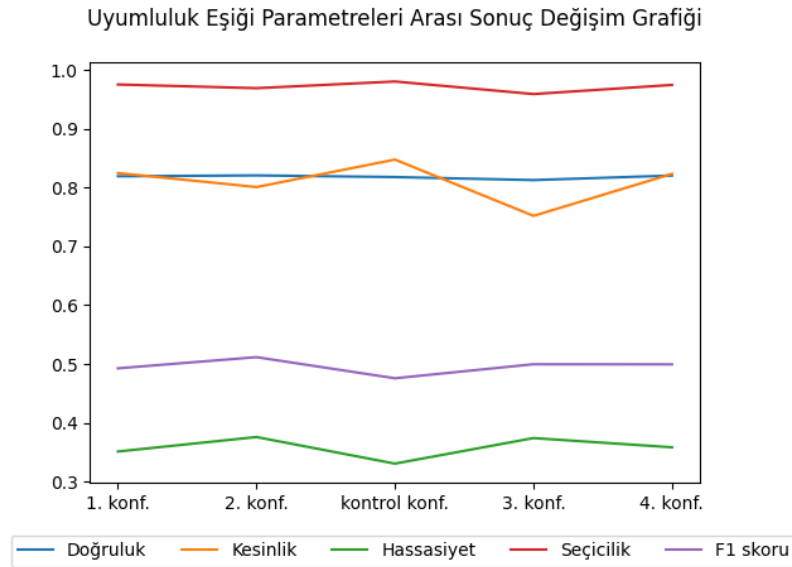
| Örnek Sayısı:8934 | Doğru | Yanlış |
|-------------------|-------|--------|
| Pozitif           | 801   | 172    |
| Negatif           | 6528  | 1433   |

**Tablo 5.39:** DEO ile ilgili 4. konfigürasyona ait performans sonuçları

|            |         |
|------------|---------|
| Doğruluk   | 0,82035 |
| Kesinlik   | 0,82323 |
| Hassasiyet | 0,35855 |
| Seçicilik  | 0,97433 |
| F1 Skoru   | 0,49953 |

### 5.3.5 Uyumluluk Eşiği (UE) Parametresi

Uyumluluk eşiği parametresi ile popülasyondaki türlerin sayısı azalmaktadır ve bireyler arasındaki farklılıklar ile türleşmenin sınırları genişlemiş olur bu değer artırıldıkça yerel minimumlara takılma riski artar algoritmanın derinleşmesi engellenir ancak adaptasyon anlamında daha etkili olabilir. Aşağıdaki bölümlerde kontrol parametreleri sabit tutularak denemeler gerçekleştirilmiştir. Kontrol konfigürasyonu ile karşılaştırıldığında F1 skoru açısından bakıldığında en iyi alternatifi 4. konfigürasyon olduğu Şekil 5.4'te görülmüştür.



**Şekil 5.4:** Uyumluluk eşiğine göre performans değişim grafiği

**1. Konfigürasyon:** Bu konfigürasyon uyumluluk eşiğinin en düşük olduğu konfigürasyon olarak karşımıza çıkmaktadır. Konfigürasyon içinde parametre değeri olarak 0.5 kullanılmış ve ~970406 saniye, yaklaşık 11 gün, süren bir süreç ardından aşağıdaki değerler elde edilmiştir. Diğer hiperparametreler temel kontrol konfigürasyonundaki değerleriyle kullanılmıştır. Uyumluluk Eşiği ile ilgili 1. konfigürasyona ait karmaşıklık matrisi ve metrik değerleri sırasıyla Tablo 5.40 ve Tablo 5.41’de verilmiştir. Sonuçlarda da görüldüğü üzere F1 skoru kontrol konfigürasyonundan 0,11172 geride kalmıştır ve 0.5 değerinin verim sağlayamadığı görülmüştür.

**Tablo 5.40:** UE ile ilgili 1. konfigürasyona ait karmaşıklık matrisi

| Örnek Sayısı:8934 | Doğru | Yanlış |
|-------------------|-------|--------|
| Pozitif           | 536   | 174    |
| Negatif           | 6526  | 1698   |

**Tablo 5.41:** UE ile ilgili 1. konfigürasyona ait performans sonuçları

|            |         |
|------------|---------|
| Doğruluk   | 0,79046 |
| Kesinlik   | 0,75493 |
| Hassasiyet | 0,23993 |
| Seçicilik  | 0,97403 |
| F1 Skoru   | 0,36413 |

**2. Konfigürasyon:** Bu konfigürasyon içerisinde ise uyumluluk eşiği değeri 1,5 olarak ayarlanmış ve ~642057 saniye, yaklaşık 7,5 gün, sonunda aşağıdaki değerler elde edilmiştir. Diğer hiperparametreler temel kontrol konfigürasyonundaki değerleriyle kullanılmıştır. Uyumluluk Eşiği ile ilgili 2. konfigürasyona ait karmaşıklık matrisi ve metrik değerleri sırasıyla Tablo 5.42 ve Tablo 5.43’da verilmiştir. F1 skoru olarak 0,01782 daha iyi bir sonuç elde edilen konfigürasyon diğer bütün metriklerde de kontrol konfigürasyonuna üstünlük sağlamıştır.

**Tablo 5.42:** UE ile ilgili 2. konfigürasyona ait karmaşıklık matrisi

| Örnek Sayısı:8934 | Doğru | Yanlış |
|-------------------|-------|--------|
| Pozitif           | 780   | 146    |
| Negatif           | 6554  | 1454   |

**Tablo 5.43:** UE ile ilgili 2. konfigürasyona ait performans sonuçları

|            |         |
|------------|---------|
| Doğruluk   | 0,82091 |
| Kesinlik   | 0,84233 |
| Hassasiyet | 0,34915 |
| Seçicilik  | 0,97821 |
| F1 Skoru   | 0,49367 |

**3. Konfigürasyon:** Bu konfigürasyon içerisinde uyumluluk eşiği 3,5 olarak ayarlanmış ve sistem ~614984 saniyede, yaklaşık 7 günde, tamamlanmıştır. Diğer hiperparametreler temel kontrol konfigürasyonundaki değerleriyle kullanılmıştır. Uyumluluk Eşiği ile ilgili 3. konfigürasyona ait karmaşıklık matrisi ve metrik değerleri sırasıyla Tablo 5.44 ve Tablo 5.45’de verilmiştir. Bu konfigürasyon sonuçlar anlamında kontrol konfigürasyonuna benzerliği ile dikkat çekmiştir. F1 skorları arasındaki fark sadece 0,00881 olabilmektedir.

**Tablo 5.44:** UE ile ilgili 3. konfigürasyona ait karmaşıklık matrisi

| Örnek Sayısı:8934 | Doğru | Yanlış |
|-------------------|-------|--------|
| Pozitif           | 766   | 161    |
| Negatif           | 6539  | 1468   |

**Tablo 5.45:** UE ile ilgili 3. konfigürasyona ait performans sonuçları

|          |         |
|----------|---------|
| Doğruluk | 0,81766 |
|----------|---------|

|            |         |
|------------|---------|
| Kesinlik   | 0,82632 |
| Hassasiyet | 0,34288 |
| Seicilik  | 0,97597 |
| F1 Skoru   | 0,48466 |

**4. Konfigürasyon:** 4. Konfigürasyon ayarlamasında uyumluluk eşıęi 4,5 olarak ayarlanmış ve süreç ~707978 saniye, yaklaşık 8 gün, alıştıktan sonra sonuçlanmıştır. Diğer hiperparametreler temel kontrol konfigürasyonundaki deęerleriyle kullanılmıştır. Uyumluluk Eşıęi ile ilgili 4. konfigürasyona ait karmaşıklık matrisi ve metrik deęerleri sırasıyla Tablo 5.46 ve Tablo 5.47’de verilmiştir. Bu konfigürasyon F1 skorları karşılaştırıldığında kontrol konfigürasyonuna kıyasla 0,03024 daha iyi bir sonuç elde etmiştir.

**Tablo 5.46:** UE ile ilgili 4. konfigürasyona ait karmaşıklık matrisi

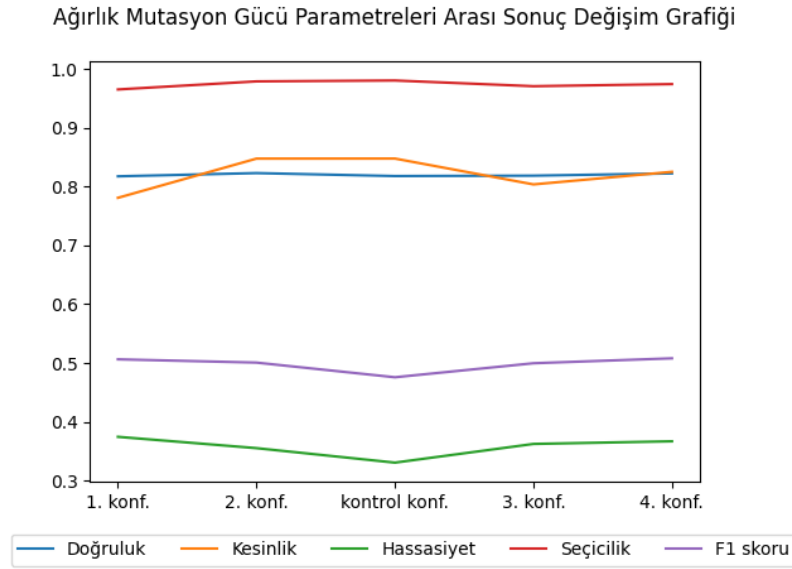
| Örnek Sayısı:8934 | Doęru | Yanlış |
|-------------------|-------|--------|
| Pozitif           | 831   | 219    |
| Negatif           | 6481  | 1403   |

**Tablo 5.47:** UE ile ilgili 4. konfigürasyona ait performans sonuçları

|            |         |
|------------|---------|
| Doęruluk   | 0,81845 |
| Kesinlik   | 0,79143 |
| Hassasiyet | 0,37198 |
| Seicilik  | 0,96731 |
| F1 Skoru   | 0,50609 |

### 5.3.6 Ağırlık Mutasyon Gücü (AMG) Parametresi

Ağırlık mutasyon gücü parametresinin artışı ile genelleme ve aşırı öğrenme gibi problemlere karşı önlem olarak konulmuş olan mutasyon mekanizmasının sürece dahil olma miktarı değiştirilebilir bu değerin yüksek olması öğrenmek konusunda sıkıntılara yol açabilecekken çok düşük olması ise ezberlemek gibi istenmeyen davranışlara yol açabilir. Aşağıdaki bölümlerde kontrol parametreleri sabit tutularak ağırlık mutasyon gücü parametresi ile denemeler gerçekleştirilmiştir. Kontrol konfigürasyonu ile karşılaştırıldığında F1 skoru açısından bakıldığında en iyi alternatifi 4. Konfigürasyon olduğu Şekil 5.5'te görülmüştür.



Şekil 5.5: Ağırlık mutasyon gücüne göre performans değişim grafiği

**1. Konfigürasyon:** 1. konfigürasyon içerisinde ağırlık mutasyon gücü olarak 0,03 değeri ile işlem yapılmış süreç ~679063 saniye, yaklaşık 8 gün sürmüştür. Diğer hiperparametreler temel kontrol konfigürasyonundaki değerleriyle kullanılmıştır. Ağırlık Mutasyonu ile ilgili 1. konfigürasyona ait karmaşıklık matrisi ve metrik değerleri sırasıyla Tablo 5.48 ve Tablo 5.49'de verilmiştir. Bu konfigürasyonun sonuç değerleri kontrol konfigürasyonu ile kıyaslandığında F1 skorunun 0,0305 daha iyi olduğu dikkat çekmektedir.

**Tablo 5.48:** AMG ile ilgili 1. konfigürasyona ait karmaşıklık matrisi

| Örnek Sayısı:8934 | Doğru | Yanlış |
|-------------------|-------|--------|
| Pozitif           | 837   | 235    |
| Negatif           | 6465  | 1397   |

**Tablo 5.49:** AMG ile ilgili 1. konfigürasyona ait performans sonuçları

|            |         |
|------------|---------|
| Doğruluk   | 0,81733 |
| Kesinlik   | 0,78078 |
| Hassasiyet | 0,37466 |
| Seçicilik  | 0,96493 |
| F1 Skoru   | 0,50635 |

**2. Konfigürasyon:** Ağırlık mutasyon gücü 2. konfigürasyonu içerisinde 0,1 değeri kullanılmış ve sistem ~593945 saniyede, yaklaşık 7 günde tamamlanmıştır. Diğer hiperparametreler temel kontrol konfigürasyonundaki değerleriyle kullanılmıştır. Ağırlık Mutasyonu ile ilgili 2. konfigürasyona ait karmaşıklık matrisi ve metrik değerleri sırasıyla Tablo 5.50 ve Tablo 5.51’de verilmiştir. Bu konfigürasyon sonuçları kontrol konfigürasyonu ile kıyaslandığında F1 skorunun 0,02494 daha iyi olduğu görülmüştür.

**Tablo 5.50:** AMG ile ilgili 2. konfigürasyona ait karmaşıklık matrisi

| Örnek Sayısı:8934 | Doğru | Yanlış |
|-------------------|-------|--------|
| Pozitif           | 794   | 143    |
| Negatif           | 6557  | 1440   |

**Tablo 5.51:** AMG ile ilgili 2. konfigürasyona ait performans sonuçları

|          |         |
|----------|---------|
| Doğruluk | 0,82281 |
|----------|---------|

|            |         |
|------------|---------|
| Kesinlik   | 0,84739 |
| Hassasiyet | 0,35542 |
| Seicilik  | 0,97866 |
| F1 Skoru   | 0,50079 |

**3. Konfigürasyon:** Ağırlık mutasyon gücü 3. konfigürasyonu içerisinde kullanılan deęer 0,2 ile kontrol konfigürasyonunu aşmış ve sistem ~576316 saniyede, yaklaşık 6,5 günde, tamamlanmıştır. Dięer hiperparametreler temel kontrol konfigürasyonundaki deęerleriyle kullanılmıştır. Ağırlık Mutasyonu ile ilgili 3. konfigürasyona ait karmaşıklık matrisi ve metrik deęerleri sırasıyla Tablo 5.52 ve Tablo 5.53’te verilmiştir. Bu konfigürasyon sonuçları kontrol konfigürasyonu ile kıyaslandığında F1 skorunun 0,02384 daha iyi olduęu görülmüştür.

**Tablo 5.52:** AMG ile ilgili 3. konfigürasyona ait karmaşıklık matrisi

| Örnek Sayısı:8934 | Doęru | Yanlış |
|-------------------|-------|--------|
| Pozitif           | 810   | 198    |
| Negatif           | 6502  | 1424   |

**Tablo 5.53:** AMG ile ilgili 3. konfigürasyona ait performans sonuçları

|            |         |
|------------|---------|
| Doęruluk   | 0,81845 |
| Kesinlik   | 0,80357 |
| Hassasiyet | 0,36258 |
| Seicilik  | 0,97045 |
| F1 Skoru   | 0,49969 |

**4. Konfigürasyon:** Ağırlık mutasyon gücü 4. konfigürasyonu içerisinde ise deęer olarak 0,3 kullanılmış ve sistem ~673148 saniyede, yaklaşık 8 günde, tamamlanmıştır. Dięer hiperparametreler temel kontrol konfigürasyonundaki

değerleriyle kullanılmıştır. Ağırlık Mutasyonu ile ilgili 4. konfigürasyona ait karmaşıklık matrisi ve metrik değerleri sırasıyla Tablo 5.54 ve Tablo 5.55’te verilmiştir. Bu konfigürasyon sonuçları kontrol konfigürasyonu ile kıyaslandığında F1 skorunun 0,0322 daha iyi olduğu görülmüştür.

**Tablo 5.54:** AMG ile ilgili 4. konfigürasyona ait karmaşıklık matrisi

| Örnek Sayısı:8934 | Doğru | Yanlış |
|-------------------|-------|--------|
| Pozitif           | 820   | 174    |
| Negatif           | 6526  | 1414   |

**Tablo 5.55:** AMG ile ilgili 4. konfigürasyona ait performans sonuçları

|            |         |
|------------|---------|
| Doğruluk   | 0,82225 |
| Kesinlik   | 0,82495 |
| Hassasiyet | 0,36705 |
| Seçicilik  | 0,97403 |
| F1 Skoru   | 0,50805 |

## 6. SONUÇ VE ÖNERİLER

Elektroensefalografi sinyalleri beyin fonksiyonlarının incelenmesi, belirli nörolojik veya nöropsikolojik rahatsızlıkların teşhisi, epilepsi sınıflandırma ve nöbet saptama, uyku bozukluklarının değerlendirilmesi gibi pek çok farklı alanda kullanılmaktadır. Karmaşık ve çok kaynaklı elektriksel verilerden oluşan EEG sinyallerinin anlamlı bir şekilde sınıflandırılması, bu verilerle çalışan kişilerin daha doğru karar analizi yapmasını sağlamaktadır. Bu tez çalışmasında, epilepsi nöbetlerine ait sinyalleri de içeren CHB-MIT Scalp EEG veri seti kullanılarak EEG sinyallerinin sınıflandırılması ve ek olarak epilepsi nöbeti tanınması gerçekleştirilmiştir. Bu amaçla veri seti üzerinde çeşitli veri ön işleme yöntemleri kullanılmıştır. Oluşturulan veri varyasyonları ile iki farklı senaryo üzerinden benzetim çalışmaları ele alınmıştır. CHB-MIT Scalp EEG veri seti kullanılarak oluşturulan ilk senaryoda 22 kanalın sekizer özniteliği belirlenerek pencere bazlı 22 kanalın ortalama değerlerine göre model eğitimleri gerçekleştirilmiştir. CHB-MIT Scalp EEG veri seti kullanılarak oluşturulan ikinci senaryoda önce pencere bazlı 22 kanala ait sekizer öznitelik belirlenmiş, sonra her örnekteki kanal öznitelikleri yan yana gelecek şekilde yeni bir veri sütunu oluşturulmuş ve bu değerler yeni bir girdi olarak tanımlanmıştır.

Bu tezde, EEG sinyallerini sınıflandırma ve epilepsi nöbeti saptama amacıyla yöntem olarak sinir evrimi yaklaşımı kullanılmıştır. Bu yaklaşım genellikle derin öğrenme algoritmaları gibi geleneksel yöntemlerle optimize edilmesi zor olan karmaşık yapay sinir ağlarını optimize etmek için tercih edilebilir. Sinir evrimi yaklaşımı ile EEG sinyallerinin analizine ek olarak, bu çalışmada literatürde çokça kullanılan derin öğrenme modellerinden evrimsel sinir ağları ile de çeşitli analizler gerçekleştirilmiştir. Elde edilen sonuçlar tam anlamıyla model performanslarının karşılaştırmaları şeklinde düşünülmemelidir. Çünkü her iki modelin tasarımları tamamen farklı mimarilere dayanarak geliştirilmektedir ve farklı yapay zeka ve makine öğrenimi ihtiyaçlarını karşılamak üzere kullanılmaktadır. NEAT, sinir ağlarını evrimsel bir süreçle eğitir yani başlangıçta basit ağlarla başlar. Gerekliğinde yeni düğümler ya da bağlantılar ekleyerek topolojisini koşullara göre optimize eder yani evrim geçirir. CNN ise çoğunlukla görüntü ve video işleme uygulamaları için tercih edilmektedir ve katmanlarında evrim geçirmez. CNN modellerinde belirli bir yapı tasarımı (evrim, havuzlama ve tam bağlantılı katmanlar) önceden belirlenir ve bu

tasarım eğitim boyunca başlangıçta karar verilen şekliyle ağı eğitir. Bu nedenle, bu çalışmadaki nümerik sonuçlar hem modele özgü, hem de derin öğrenme modellerine göre sinir evrimi yaklaşımının avantaj ve dezavantajları dikkate alınarak değerlendirilmektedir.

Beyin dalgalarının zamana bağlı değişimini ölçen elektroensefalografi yüksek boyutlu ve karmaşık bir sinyaldir. Sinir evrimi yaklaşımı içerisinde değerlendirilen NEAT algoritması büyük veri setlerini veya çok boyutlu görüntü verilerini işlerken zorlanmaktadır. 5. Bölüm’de de sunulduğu üzere, elektroensefalografi gibi çok değişkenli analog zaman serileri için NEAT algoritması üzerinden model eğitim süreleri CNN algoritmasının model eğitim süreleri yanında daha yüksek değerlerde elde edilmiştir. Bu anlamda, NEAT algoritmasının daha az karmaşık problemlerde daha hızlı sonuç verme özelliğinin bu çalışmadaki nümerik sonuçlarla doğrulandığı söylenebilir. Elde edilen benzetim sonuçlarına göre, sinir evrimi yaklaşımının CNN modellerine kıyasla daha düşük değerlendirme metriklerine sahip olduğu gözlenmiştir. NEAT algoritmasının daha farklı sonuçlar elde edebilmesi için çeşitli popülasyonlarla ve çeşitli parametrelerle çalışması sağlansa da sonuç herhangi bir metrik yararına anlamlı şekilde değişmemiştir. Bu durum, NEAT algoritmasının çok değişkenli zaman serilerine uygulanması için daha farklı yaklaşımlarla denenmesi gerektiğini ortaya koymuştur. Zira NEAT açısından bakıldığında evrimsel gelişim süreci eninde sonunda doğru bir ağ mimarisi geliştirecektir, ancak bu mimarinin jenerasyonlar arasında gelişimi yeterli kapasitede bir ağın bağları kopararak eğitilmesine kıyasla çok uzun zamanlarda gerçekleşecektir. Bu çalışma içerisinde NEAT algoritmasına çalışma başına 1000 jenerasyon imkânı tanınmış ve  $\sim 0,5$  F1 skoru civarında sonuçlara ancak yaklaşılmıştır. Dolayısıyla, EEG’in çok boyutlu ve zaman bağımlı dinamikleri için NEAT algoritmasının ağırlık ayarları ve yapısal mimari değişimleri CNN algoritması karşısında bu çalışmadaki özellikler ile kullanıldığında yavaş kalmıştır. Elde edilen benzetim sonuçlarına göre, NEAT ve zaman serileri ile çalışılırken girdi setinin ve popülasyonun olabildiğince küçük olması gerektiği belirlenmiştir. Diğer parametrelerin ise ortalama değerlerde tutularak yüksek zaman kazancı sağlanarak jenerasyon sayısının artırılmasının önemi ortaya çıkarılmıştır. Buna karşın, büyük veri setleri için CNN modelleri yüksek hesaplama gücüne ihtiyaç duyar. Yüksek doğruluk gerektiğinde GPU ya da benzeri güçlü donanımlar gerektirir. Eğitim sırasında belirli bir veri kümesine bağlı olduğu için yeni tür EEG verileriyle doğrudan adaptasyonu

NEAT algoritmasına göre zorlaşır. Ayrıca, epileptik nöbet gibi durumları analiz etmek için ayrıntılı hiperparametre ayarlamalarına ihtiyaç duyar.

Bu çalışmanın en önemli katkısı; literatürde ilk kez karmaşık ve çok kaynaklı EEG sinyallerinin sınıflandırılmasında ve epilepsi nöbeti saptamada sinir evrimi yaklaşımının kullanılmasıdır. Elde edilen sonuçlar, sinir evrimi yaklaşımının en az CNN modelleri kadar başarılı olduğunu, aynı zamanda da hiperparametre optimizasyonu çalışmaları ile daha yüksek başarımlarına çıkarılabileceğini göstermiştir. Ek olarak, epilepsi teşhisinde EEG verisi kullanıldığında CNN mimarileri ile veri paternleri çıkarmak NEAT'e kıyasla genellikle daha etkilidir. NEAT, küçük veri setlerinde daha az karmaşık görevler için uygun olsa da, EEG'nin karmaşık ve yüksek boyutlu yapısı CNN modelleri ile eğitim için daha uygundur. CNN modelleri ile bu tarz yoğun veriler için sınıflandırma görevlerinde yüksek doğruluk sağlanabilir, bunun yanı sıra donanım ve eğitim süreleri maliyetli olabilir. Belirtilen hususlar dikkate alındığında, bu tez kapsamında araştırılan konu özelinde gelecekte yapılabilecek çalışmalar şu şekilde özetlenebilir:

- Bu çalışmada kullanılan yapay zeka modellerinin güçlü yanlarını birleştiren hibrit bir model tasarlanarak sinir evrimi ile optimize edilen CNN mimarilerinin EEG analizinde kullanılması düşünülebilir.
- Paralel veri işleme tekniklerinden yararlanarak eğitim süreci hızlandırılabilir.
- Bu çalışmada CHB-MIT Scalp EEG veri seti kullanılmıştır. Farklı yaş grupları, uyku evreleri veya diğer nörolojik bozuklukları içeren veri setleri üzerinde analizler yapılarak tasarlanan yapay zeka modellerinin geliştirilebilirliği değerlendirilebilir.
- EEG sinyallerine ek olarak çok-modlu veri analiz yöntemleri (fMRI, MEG veya göz hareketleri vb. içeren) geliştirilerek epilepsi gibi rahatsızlıklar daha kapsamlı analiz edilebilir.
- EEG sinyallerinden daha nitelikli özniteliklerin çıkarılması için dalgacık dönüşümü gibi ileri sinyal işleme teknikleri uygulanabilir.

## 7. KAYNAKLAR

Acharya, U.R., Oh, S.L., Hagiwara, Y., Tan, J.H., and Adeli, H, “Deep convolutional neural network for the automated detection and diagnosis of seizure using EEG signals”, *Computers in Biology and Medicine*, 100, 270-278, (2018).

Andersen T, “Neuroevolution through augmenting topologies applied to evolving neural networks to play othello”, Undergraduate Honors Thesis, *Technical Report HR-02-01*, Department of Computer Sciences, The University of Texas, Austin, (2002).

Bird, J.J., and Lotfi, A, “Fall compensation detection from EEG using neuroevolution and genetic hyperparameter optimisation”, *Genetic Programming Evolvable Machines*, 24(1), 6, (2023).

Bora, İ., and Yeni, N, “EEG Atlası”, *Nobel Tıp Kitapevleri*, (2020).

Chen, L., and Alahakoon, D. “Neuroevolution of augmenting topologies with learning for data classification”, *International Conference on Information and Automation*, Colombo, Sri Lanka, 367-371, (2006).

Collura, T.F, “History and evolution of electroencephalographic instruments and techniques”, *Journal of Clinical Neurophysiology*, 10(4), 476-504, (1993).

De Jong, K, “Evolutionary computation: a unified approach”, *In Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pp. 373-388, (2017).

Dweiri, Y., Hajjar, Y., and Hatahet, O, “A novel neuroevolution model for emg-based hand gesture classification”, *Neural Computing and Application*, 35, 10621–10635, (2023).

Floreano, D., Dürr, P., and Mattiussi, C, “Neuroevolution: from architectures to learning”, *Evolutionary Intelligence*, 1, 47-62, (2008).

Gomez, F. J., and Miikkulainen, R, “Active guidance for a finless rocket using neuroevolution”, *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 2084-2095, (2003).

Gomez, F.J, “Robust non-linear control through neuroevolution”, *PhD Thesis*, University of Texas, (2003).

Görgülü, Ü., ve Fesci, H, “Epilepsi ile yaşam: Epilepsinin psikososyal etkileri”, *Göztepe Tıp Dergisi*, 26, 1, 27-32, (2011).

Guo, L., Rivero, D., and Pazos, A, “Epileptic seizure detection using multiwavelet transform based approximate entropy and artificial neural networks”, *Journal of Neuroscience Methods*, 193, 156-63, (2010).

Guttag, J, “CHB-MIT Scalp EEG Database (version 1.0.0).” *PhysioNet*. (2010).

Hansen, N., and Ostermeier, A, “Completely derandomized self-adaptation in evolution strategies”, *Evolutionary Computation*, 9(2), 159-195, (2001).

Hausknecht, M., Khandelwal, P., Miikkulainen, R., and Stone, P, “HyperNEAT-GGP: a hyperNEAT-based atari general game player”, *Proceedings of the 14th International Conference on Genetic and Evolutionary Computation*, (2012).

Hausknecht, M., Lehman, J., Miikkulainen, R., and Stone, P, “A neuroevolution approach to general atari game playing”, *IEEE Transactions on Computational Intelligence and AI in Games*, 6(4), 355-366, (2014).

Hazarika, N., Chen, J.Z., Tsoi, A.C., and Sergejew, A, “Classification of EEG signals using the wavelet transform”, *Signal Processing*, 59(1), 61-72, (1997).

Huang, P., Lehman, J., Mok, A., Miikkulainen, R., and Sentis, L, “Grasping novel objects with a dexterous robotic hand through neuroevolution”, *IEEE Symposium on Computational Intelligence in Control and Automation*, (2014).

Lehman, J., and Miikkulainen, R, “Neuroevolution”, *Scholarpedia*, 8(6), 30977, (2013).

Li, D., Xie, Q., Jin, Q., and Hirasawa, K, “A sequential method using multiplicative extreme learning machine for epileptic seizure detection”, *Neurocomputing*, 214, 692-707, (2016).

Lu, D., & Triesch, J, “Residual deep convolutional neural network for eeg signal classification in epilepsy”, *Arxiv Preprint*, arXiv:1903.08100, (2019).

Miikkulainen, R., Liang, J., Meyerson, E., Rawal, A., Fink, D., Francon, O., and Hodjat, B, “Evolving deep neural networks”, *In Artificial Intelligence in the Age of Neural Networks and Brain Computing*, 269-287, (2024).

Mwandau, B.N, “Investigating keystroke dynamics as a two-factor biometric security”, *Master Thesis*, Strathmore University, (2018).

Nicolaou, N., and Georgiou, J, “Detection of epileptic electroencephalogram based on permutation entropy and support vector machines”, *Expert Systems with Applications*, 39, 1, 202-209, (2012).

Nigam, V., and Graupe, D, “A neural-network-based detection of epilepsy.”, *Neurological Research*, 26, 55-60, (2004).

Papavasileiou, E., Cornelis, J., and Jansen, B, “A systematic literature review of the successors of neuroevolution of augmenting topologies”, *Evolutionary Computation*, 29(1), 1-73, (2021).

Purwono, P., Ma'arif, A., Rahmانيar, W., Fathurrahman, H.I.K., Frisky, A.Z.K., and ul Haq, Q.M, “Understanding of convolutional neural network: a review”, *International Journal of Robotics and Control Systems*, 2(4), 739-748, (2022).

Raj, P., Edla, D.R., and Parne, S.R, “Concealed information test using neuro evolution of augmenting topologies”, *Wireless Personel Communications*, 135, 1135–1148, (2024).

Rashed-Al-Mahfuz, M., Moni, M.A., Uddin, S., Alyami, S.A., Summers, M.A., and Eapen, V, “A deep convolutional neural network method to detect seizures and characteristic frequencies using epileptic electroencephalogram (EEG) data” *IEEE Journal of Translational Engineering in Health and Medicine*, 9, 2000112, (2021).

Salimans, T., Ho, J., Chen, X., Sidor, S., and Sutskever, I, “Evolution strategies as a scalable alternative to reinforcement learning”, *Arxiv Preprint arXiv:1703.03864*, (2017).

San-Segundo, R., Gil-Martín, M., D'Haro-Enriquez, L.F., and Pardo, J.M, “Classification of epileptic EEG recordings using signal transforms and convolutional neural networks”, *Computers in Biology and Medicine*, 109, 148-158, (2019).

Sharma, R., Pachori, R.B., and Sircar, P, “Seizures classification based on higher order statistics and deep neural network”, *Biomedical Signal Processing and Control*, 59, 101921, (2020).

Shen, M., Wen, P., Song, B., and Li, Y, “Real-time epilepsy seizure detection based on EEG using tunable-Q wavelet transform and convolutional neural network”, *Biomedical Signal Processing and Control*, 82, 104566, (2023).

Siddiqui, M. K., Morales-Menendez, R., Huang, X., and Hussain, N, “A review of epileptic seizure detection using machine learning classifiers”, *Brain Informatics*, 7(1), 5, (2020).

Stanley, K., and Miikkulainen, R, “Evolving neural networks through augmenting topologies”, *Evolutionary Computation*, 10, 99-127, (2002).

Stanley, K., Bryant, B.D., and Miikkulainen, R, “Real-time neuroevolution in the NERO video game”, *IEEE Transactions Evolutionary Computation*, 9(6), 653-668, (2005).

Stanley, K.O., Clune, J., Lehman, J., and Miikkulainen, R, “Designing neural networks through neuroevolution”, *Nature Machine Intelligence*, 1(1), 24-35, (2019).

Strayhorn D. “EEG Atlas [online]”, (1 December 2024), <https://eegatlas-online.com/>, (2024).

Szegedi, G., Kiss, P., and Horvath, T, “Evolutionary federated learning on EEG-data”, *Conference on Theory and Practice of Information Technologies*, pp. 71-78, (2019).

Takahashi, H., Emami, A., Shinozaki, T., Kunii, N., Matsuo, T., and Kawai, K, “Convolutional neural network with autoencoder-assisted multiclass labelling for seizure detection based on scalp electroencephalography”, *Computers in Biology and Medicine*, 125, 104016, (2020).

Tiwari, S., Goel, S., and Bhardwaj, A, “MIDNN- a classification approach for the EEG based motor imagery tasks using deep neural network”, *Applied Intelligence*, 52,4824-4843, (2022).

Wei, Z., Zou, J., Zhang, J., and Xu, J, “Automatic epileptic EEG detection using convolutional neural network with improvements in time-domain”, *Biomedical Signal Processing and Control*, 53, 101551, (2019).

Wijaya, A.B.M., Ikawahyuni, D.S., Gea, R., and Maedjaja, F, “Role comparison between deep belief neural network and neuroEvolution of augmenting topologies to detect diabetes”, *International Journal on Informatics Visualization*, 5(2), 156-161, (2021).

Xu, Y., Yu, Z., Li, Y., Liu, Y., Li, Y., and Wang, Y, “Autism spectrum disorder diagnosis with EEG signals using time series maps of brain functional connectivity and a combined CNN–LSTM model”, *Computer Methods and Programs in Biomedicine*, 250, 108196, (2024).

Yao X. “Evolving artificial neural networks”, *Proceedings of the IEEE*, 87, 9, 1423-1447, (1999).

Yildiz, A., Zan, H., and Said, S, “Classification and analysis of epileptic EEG recordings using convolutional neural network and class activation mapping”, *Biomedical Signal Processing and Control*, 68, 102720, (2021).

Zaid Y., Sah, M., and Direkoglu, C, “Pre-processed and combined EEG data for epileptic seizure classification using deep learning”, *Biomedical Signal Processing and Control*, 84, 104738, (2023).