T.C.

SAMSUN ÜNİVERSİTESİ

MÜHENDİSLİK FAKÜLTESİ

YAZILIM MÜHENDİSLİĞİ BÖLÜMÜ

YAĞIŞ TAHMİNİ

Dönem Projesi

Mehmet Arda Ogan

Dersin Öğretim Üyesi

Doç. Dr. Muammer TÜRKOĞLU

SAMSUN

2024

ÖZET

Projede çeşitli makine öğrenimi ve derin öğrenme modelleri kullanılarak yağış tahmini yapılmıştır. Kullanılan modeller arasında KNN, Karar Ağacı, Lojistik Regresyon, SVC, Yapay Sinir Ağı (ANN) ve Evrişimli Sinir Ağı (CNN) bulunmaktadır. Bu çalışma, hava tahmini alanında yapay zeka tekniklerinin etkinliğini araştırmayı amaçlamaktadır. Modellerin performansı, doğruluk, Jaccard İndeksi, F1 Skoru ve Log Kaybı gibi metriklerle değerlendirilmiştir.

Veri kümesi, Avustralya Meteoroloji Bürosu'ndan alınmış olup, çeşitli meteorolojik özellikleri içermektedir. Veri setindeki rüzgar yönü gibi kategorik veriler, sayısal değerlere dönüştürülerek modellerde kullanılabilir hale getirilmiştir. Ayrıca, eksik veriler ilgili sütunun ortalama değeri ile doldurulmuştur.

Projede, her model için GridSearchCV yöntemi kullanılarak hiperparametre optimizasyonu gerçekleştirilmiştir. Bu süreç, modellerin en iyi performansı göstermesini sağlamak için çeşitli parametre kombinasyonlarının denenmesini içermektedir. Modeller, çapraz doğrulama ile test edilerek her birinin performansı ayrıntılı olarak karşılaştırılmıştır.

Sonuçlar, yapay zeka tekniklerinin hava tahmini için etkili bir şekilde kullanılabileceğini göstermektedir. Özellikle derin öğrenme modelleri olan ANN ve CNN, yüksek doğruluk oranları ile öne çıkmıştır. Bu çalışma, gelecekte daha geniş veri kümeleri ve ek özellikler kullanılarak daha da geliştirilebilir. Hiperparametre ayarlarının ve optimizasyon tekniklerinin daha ayrıntılı araştırılmasıyla, daha güvenilir tahminler elde edilebilir.

Anahtar Kelimeler: Yağış Tahmini, Makine Öğrenimi, Derin Öğrenme, Karar Ağacı, Lojistik Regresyon, Yapay Sinir Ağı, Evrişimli Sinir Ağı.

GİRİŞ

Hava tahmini, günlük yaşamımızda büyük önem taşıyan bir konudur. Tarımdan ulaşıma, enerji yönetiminden afet hazırlıklarına kadar birçok alanda doğru hava tahminleri kritik kararların alınmasına yardımcı olur. Ancak, hava tahmini yapmak karmaşık veri analizi ve modelleme gerektirir. Bu çalışmada, çeşitli makine öğrenimi ve derin öğrenme teknikleri kullanılarak yağış tahmini yapılması hedeflenmektedir. Literatürde, yapay zeka yöntemlerinin hava tahmininde kullanımı yaygın olarak araştırılmıştır. Ancak, bu çalışmanın amacı, farklı modelleri karşılaştırarak hangi modelin en yüksek doğrulukla tahmin yapabildiğini belirlemektir.

Daha önce yapılan çalışmalarda, KNN, Karar Ağacı, Lojistik Regresyon, SVC, ANN ve CNN gibi modellerin hava tahmini için kullanıldığı görülmüştür. Bu modellerin her biri, belirli koşullar altında farklı performans göstermektedir. Örneğin, KNN ve Karar Ağacı modelleri basit ve anlaşılabilir olmalarıyla öne çıkarken, ANN ve CNN modelleri büyük veri kümelerinde daha yüksek doğruluk sağlayabilmektedir. Ancak, bu modellerin karşılaştırmalı bir analizinin eksik olduğu literatürde belirtilmiştir.

Bu çalışmanın bilimsel değeri, farklı makine öğrenimi ve derin öğrenme modellerini kullanarak yağış tahmini performanslarını karşılaştırmakta yatmaktadır. Çalışmanın yeniliği, farklı modellerin hiperparametre optimizasyonları ile birlikte değerlendirilmesi ve bu modellerin performanslarının detaylı bir şekilde analiz edilmesidir. Bu şekilde, hangi modelin hangi koşullarda daha iyi performans gösterdiği belirlenerek, hava tahmininde kullanılabilecek en iyi modelin seçilmesine katkı sağlanacaktır.

Bu çalışmanın araştırma sorusu, "Hangi makine öğrenimi veya derin öğrenme modeli yağış tahmini için en yüksek doğruluğu sağlar?" olarak belirlenmiştir. Hipotezimiz, derin öğrenme modellerinin (ANN ve CNN) karmaşık veri setlerinde daha yüksek performans göstereceğidir.

Bu projenin amacı, yağış tahmininde en yüksek doğruluğu sağlayan modeli belirlemektir. Hedeflerimiz ise farklı makine öğrenimi ve derin öğrenme modelleri kullanarak yağış tahmini yapmak, modellerin hiperparametre optimizasyonlarını gerçekleştirmek, modellerin performanslarını doğruluk, Jaccard İndeksi, F1 Skoru ve Log Kaybı gibi metrikler üzerinden karşılaştırmak ve en yüksek doğruluğu sağlayan modeli belirlemektir.

Bu hedefler, araştırma süresince ulaşılabilir ve ölçülebilir nitelikte olup, projenin sonunda hava tahmininde kullanılabilecek en iyi modelin seçilmesine katkı sağlayacaktır.

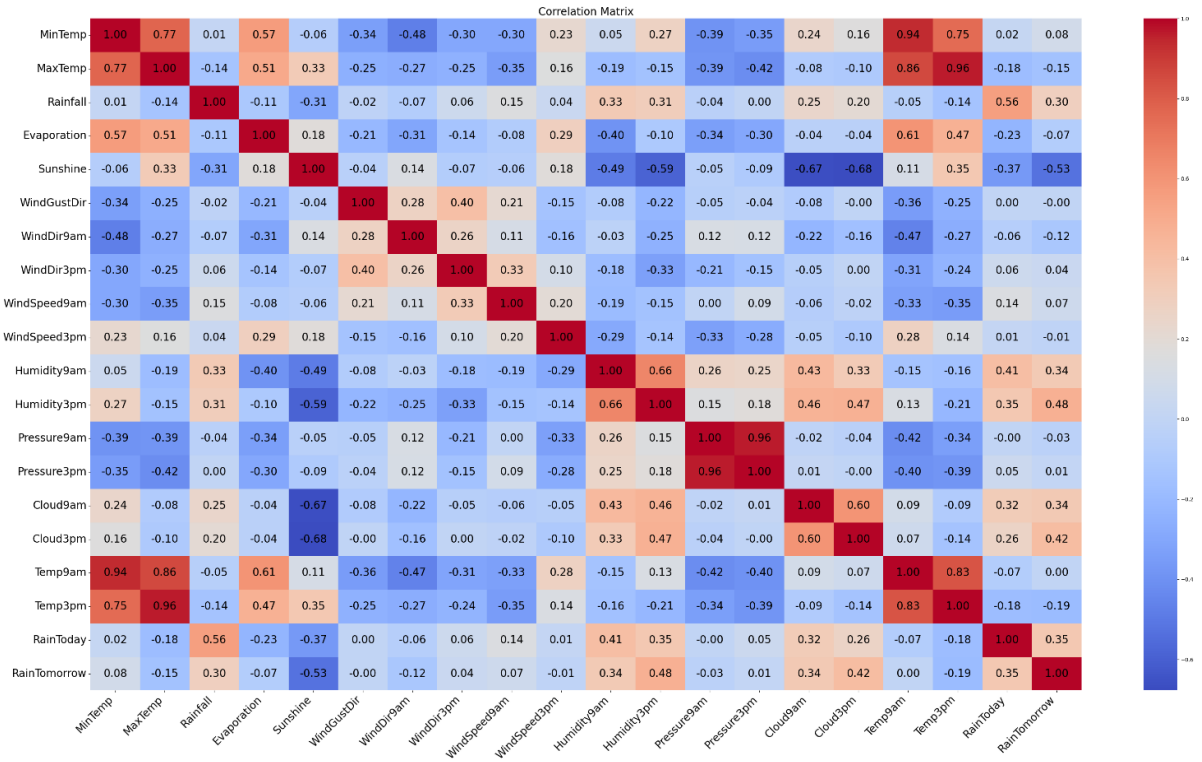
MATERYAL VE YÖNTEM

Veri Kümesi:

Bu projede kullanılan veri seti, Avustralya Meteoroloji Bürosu'ndan alınmıştır ve 2008-2017 yılları arasındaki meteorolojik verileri içermektedir. Veri kümesi, çeşitli meteorolojik özellikleri ve yağış bilgilerini içerir. Her bir gözlem, o günün minimum ve maksimum sıcaklıkları, yağış miktarı, buharlaşma miktarı, güneşlenme süresi, rüzgar yönü ve hızı, nem, basınç ve bulut örtüsü gibi özellikleri içermektedir. Bu veri kümesi, yağış tahmini yapmak için zengin ve çeşitli bir kaynak sunar.

Veri setinin makine öğrenimi ve derin öğrenme modelleri ile kullanılabilmesi için ön işlem adımları uygulanmıştır:

1. **Eksik Verilerin Doldurulması:** Veri setinde eksik değerler tespit edilmiştir. Eksik veriler, ilgili sütunun ortalama değeri ile doldurulmuştur. Bu, veri kümesindeki eksik değerlerin model performansını olumsuz etkilememesi için yapılmıştır.
2. **Kategorik Verilerin Sayısal Değerlere Dönüştürülmesi:** Veri setinde bulunan rüzgar yönü gibi kategorik değişkenler, sayısal değerlere dönüştürülmüştür. Örneğin, rüzgar yönü metin olarak verilmişse, her bir yön için benzersiz bir sayısal değer atanmıştır. Ayrıca, "RainToday" ve "RainTomorrow" sütunlarındaki "Evet" ve "Hayır" değerleri sırasıyla 1 ve 0 olarak dönüştürülmüştür.
3. **Tarih ve Gereksiz Sütunların Çıkarılması:** Tahmin modellerinde kullanılmayan "Date" ve tüm gözlemler için aynı değeri taşıyan "WindGustSpeed" sütunları veri setinden çıkarılmıştır. Bu sütunların çıkarılması, modelin gereksiz verilerle eğitilmesini önlemiştir.
4. **Korelasyon Analizi:** Veri setindeki özellikler arasındaki korelasyon incelenmiş ve yüksek korelasyonlu özellikler belirlenmiştir. Korelasyon matrisindeki mutlak değeri 0.1'den fazla olan özellikler seçilerek veri seti sadeleştirilmiştir. Bu işlem, modelin daha az ama daha anlamlı verilerle eğitilmesini sağlamıştır.



Bu ön işlem adımları sonucunda, nihai veri seti aşağıdaki özellikleri içermektedir:

- Sunshine
- Humidity3pm
- Cloud3pm
- RainToday
- Cloud9am

- Humidity9am
- Rainfall
- Temp3pm
- MaxTemp
- WindDir9am

Bu veri seti, yağış tahmin modellerinin eğitimi ve değerlendirilmesi için kullanılmıştır. Elde edilen veriler ve uygulanan ön işlem adımları, modellerin performansını optimize etmeye yardımcı olmuştur.

Modeller:

Projede yağış tahmini için çeşitli makine öğrenimi ve derin öğrenme modelleri kullanılmıştır. Her bir modelin genel özellikleri, konfigürasyonları ve hiperparametre ayarlamalarının gerekçeleri aşağıda açıklanmıştır.

K-En Yakın Komşu (KNN):

- Açıklama: K-En Yakın Komşu (KNN), sınıflandırma ve regresyon için kullanılan sezgisel bir algoritmadır. Yeni bir veri noktasını sınıflandırmak için en yakın kk komşunun sınıflarına bakar ve en çok oy alan sınıfı tahmin eder.
- Konfigürasyon: KNeighborsClassifier sınıfı kullanılmıştır.
- Hiperparametre Ayarlamaları: GridSearchCV kullanılarak n_neighbors, weights ve algorithm parametreleri optimize edilmiştir. N_neighbors parametresi, en iyi sonuçları veren komşu sayısını belirlerken, weights parametresi, komşuların ağırlıklandırma metodunu (uniform veya distance) belirlemiştir. algorithm parametresi ise en uygun arama algoritmasını seçmiştir.
- Gerekçe: Bu ayarlamalar, modelin doğruluğunu artırmak ve en iyi performansı sağlayan konfigürasyonu bulmak için yapılmıştır.

Karar Ağacı (Decision Tree):

- Açıklama: Karar ağacı, veri noktalarını sınıflandırmak için dallara ayrılan bir ağ yapısıdır. Her bir düğümde bir özellik seçilir ve veri bu özellik temelinde dallara ayrılır.
- Konfigürasyon: DecisionTreeClassifier sınıfı kullanılmıştır.
- Hiperparametre Ayarlamaları: GridSearchCV kullanılarak max_depth, min_samples_split ve min_samples_leaf parametreleri optimize edilmiştir. max_depth parametresi, ağacın maksimum derinliğini belirlerken, min_samples_split ve min_samples_leaf parametreleri, her bir düğümde en az kaç örneğin bulunması gerektiğini belirlemiştir.
- Gerekçe: Bu ayarlamalar, modelin aşırı uyum (overfitting) riskini azaltmak ve genelleme yeteneğini artırmak için yapılmıştır.

Lojistik Regresyon:

- Açıklama: Lojistik Regresyon, ikili sınıflandırma problemleri için kullanılan doğrusal bir modeldir. Verilerin doğrusal bir kombinasyonunu kullanarak sınıflar arasında olasılık hesaplar.
- Konfigürasyon: LogisticRegression sınıfı kullanılmıştır.
- Hiperparametre Ayarlamaları: GridSearchCV kullanılarak `C` (düzenleme parametresi) ve solver (çözücü) parametreleri optimize edilmiştir. `C` parametresi, modelin düzenleme seviyesini kontrol ederken, solver parametresi en uygun optimizasyon algoritmasını seçmiştir.
- Gerekçe: Bu ayarlamalar, modelin doğruluğunu ve genel performansını optimize etmek için yapılmıştır.

Destek Vektör Makineleri (SVC):

- Açıklama: Destek Vektör Makineleri (SVC), veri noktalarını sınıflandırmak için en uygun hiper düzlemi bulmaya çalışan bir modeldir. RBF (Radial Basis Function) ve linear çekirdeği kullanılmıştır.
- Konfigürasyon: SVC sınıfı kullanılmıştır.
- Hiperparametre Ayarlamaları: GridSearchCV kullanılarak C, kernel ve gamma parametreleri optimize edilmiştir. C parametresi, ceza parametresini kontrol ederken, kernel parametresi çekirdek fonksiyonunu belirlemiştir. gamma parametresi ise çekirdek fonksiyonunun etki alanını ayarlamıştır.
- Gerekçe: Bu ayarlamalar, modelin doğruluğunu ve genelleme yeteneğini optimize etmek için yapılmıştır.

Yapay Sinir Ağları (ANN):

- Açıklama: Yapay Sinir Ağları (ANN), biyolojik sinir ağlarından esinlenerek oluşturulmuş, birçok düğüm ve katmandan oluşan bir yapay zeka modelidir. Giriş katmanından alınan veriler bir veya daha fazla gizli katman aracılığıyla işlenir ve çıkış katmanına iletilir.
- Konfigürasyon: Sequential modeli kullanılarak oluşturulmuştur. İlk katmanda 64 nöron ve giriş boyutu olarak özelliklerin sayısına eşit olan bir input_dim parametresi belirtilmiştir. İki gizli katman eklenmiş ve her katmanda ReLU aktivasyon fonksiyonu kullanılmıştır. Çıkış katmanında sigmoid aktivasyon fonksiyonu ile ikili sınıflandırma yapılmıştır.
- Hiperparametre Ayarlamaları: Model binary_crossentropy kayıp fonksiyonu ve Adam optimize edici ile derlenmiştir. Model 50 epoch ve 10'arlık mini partiler halinde verilerle eğitilmiştir. EarlyStopping callback'i kullanılarak doğrulama kaybının iyileşmediği durumda erken durdurma uygulanmıştır.
- Gerekçe: Bu ayarlamalar, modelin aşırı uyum riskini azaltmak ve doğrulama verisi üzerinde en iyi performansı sağlamak için yapılmıştır.

Evrişimli Sinir Ağları (CNN):

- Açıklama: Evrişimli Sinir Ağları (CNN), özellikle görüntü işleme görevleri için geliştirilmiş, veri noktalarının yerel bağlantılarını öğrenmeye odaklanan bir derin öğrenme modelidir. 1D CNN, yağış verileri gibi sıra bağımlı veri kümeleri için kullanılır.
- Konfigürasyon: Sequential modeli kullanılarak oluşturulmuştur. İlk katmanda 32 filtreli ve 3 boyutlu kernel kullanılarak bir Conv1D katmanı eklenmiştir. Bu katman ReLU aktivasyon fonksiyonunu kullanır ve giriş şekli (özellik sayısı, 1) olarak belirtilmiştir. Ardından MaxPooling1D katmanı ile boyut indirgeme yapılmıştır. Flatten katmanı ile veriler düzleştirilmiştir. İki Dense katmanı eklenmiştir: biri 10 nöronlu ve ReLU aktivasyon fonksiyonlu, diğeri ise sigmoid aktivasyon fonksiyonlu ve 1 nöronlu çıkış katmanı.
- Hiperparametre Ayarlamaları: Model binary_crossentropy kayıp fonksiyonu ve Adam optimize edici ile derlenmiştir. Model 50 epoch ve 10'arlık mini partiler halinde verilerle eğitilmiştir. EarlyStopping callback'i kullanılarak doğrulama kaybının iyileşmediği durumda erken durdurma uygulanmıştır.
- Gerekçe: Bu ayarlamalar, modelin doğruluğunu artırmak ve aşırı uyum riskini azaltmak için yapılmıştır.

Performans Metrikleri

Projede kullanılan modellerin başarımını ve hata oranlarını değerlendirmek için çeşitli performans metrikleri kullanılmıştır. Bu metriklerin seçilme nedenleri ve açıklamaları aşağıda verilmiştir:

- **Accuracy (Doğruluk)**: Modelin doğru tahmin ettiği örneklerin oranını ifade eder. Genel performansı değerlendirmek için kullanılır. Ancak, dengesiz veri kümelerinde yanıltıcı olabilir.
- **Jaccard Index**: Pozitif tahmin edilen kesişim bölgesinin, pozitif tahmin edilen birlik bölgesine oranıdır. Dengesiz veri kümelerinde doğruluğa alternatif olarak kullanılır ve sınıflar arasındaki farkları daha iyi anlamaya yardımcı olur.
- **F1 Score**: Hassasiyet (precision) ve geri çağırma (recall) metriklerinin harmonik ortalamasıdır. Dengesiz veri kümelerinde, yanlış pozitif ve yanlış negatifleri dengeleyerek modelin başarımını ölçer.
- **Log Loss (Lojistik Kayıp)**: Modelin tahminlerinin olasılık dağılımı ile gerçek etiketler arasındaki farkı ölçer. Modelin olasılık tahminlerinin doğruluğunu değerlendirir, çok sınıflı sınıflandırma problemleri için kullanışlıdır.

Bu metrikler, her bir modelin farklı yönlerini değerlendirir ve modelin genel başarısını anlamak için birlikte kullanılır. Doğruluk genel performansı ölçerken, Jaccard İndeksi ve F1 Skoru dengesiz veri kümeleri için daha adil değerlendirmeler sunar. Log Loss ise modelin olasılık tahminlerinin doğruluğunu sağlar.

BULGULAR

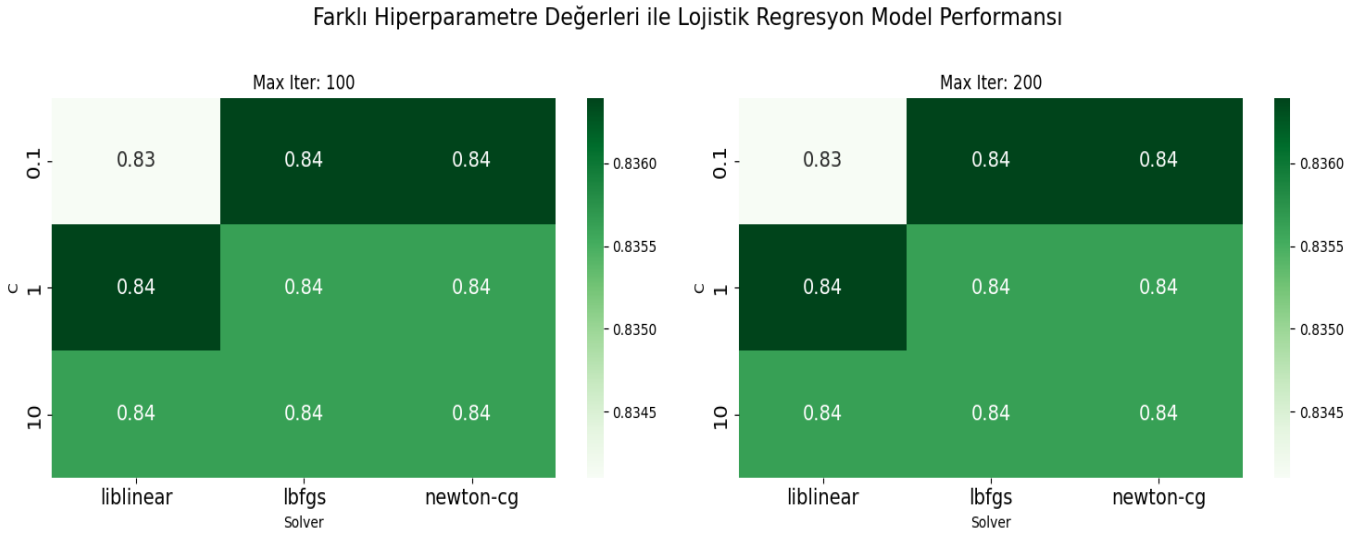
Çeşitli modellerin performans karşılaştırması aşağıdaki tabloda özetlenmiştir:

	Model	Accuracy	Jaccard Index	F1 Score	Log Loss
0	KNN	0.833588	0.480952	0.649518	0.546094
1	Decision Tree	0.824427	0.472477	0.641745	0.507958
2	Logistic Regression	0.830534	0.490826	0.658462	0.375161
3	SVC	0.832061	0.468599	0.638158	0.414528
4	ANN	0.832061	0.500000	0.666667	0.377513
5	CNN	0.832061	0.497717	0.664634	0.382486

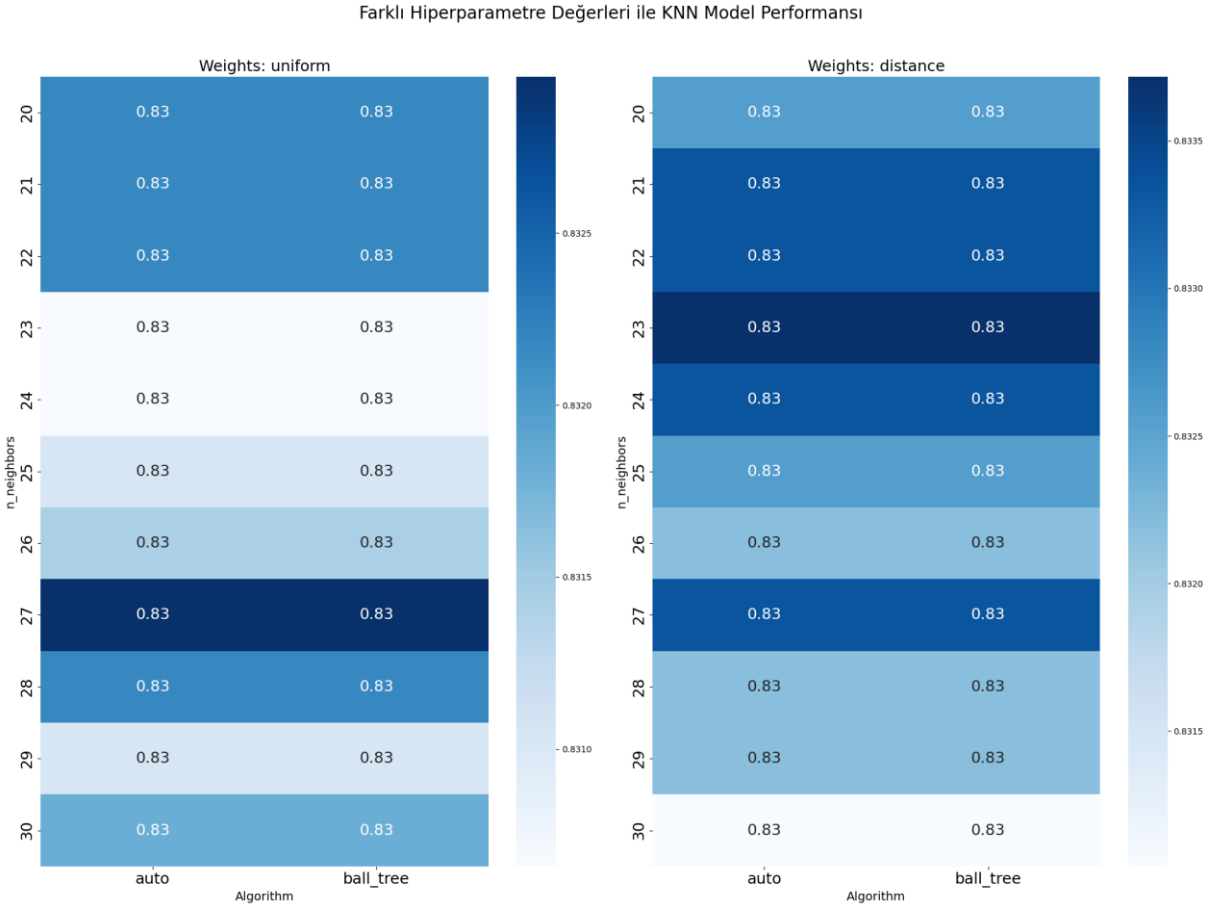
Hiperparametre Optimizasyonu

Projede kullanılan her model için GridSearchCV yöntemiyle hiperparametre optimizasyonu ve en uygun parametre ayarları yapılmıştır. Aşağıda, modellerin hiperparametre optimizasyonu sırasında elde edilen ısı haritaları bulunmaktadır. Bu ısı haritaları farklı hiperparametre kombinasyonlarının model performansına nasıl etki ettiğini göstermektedir.

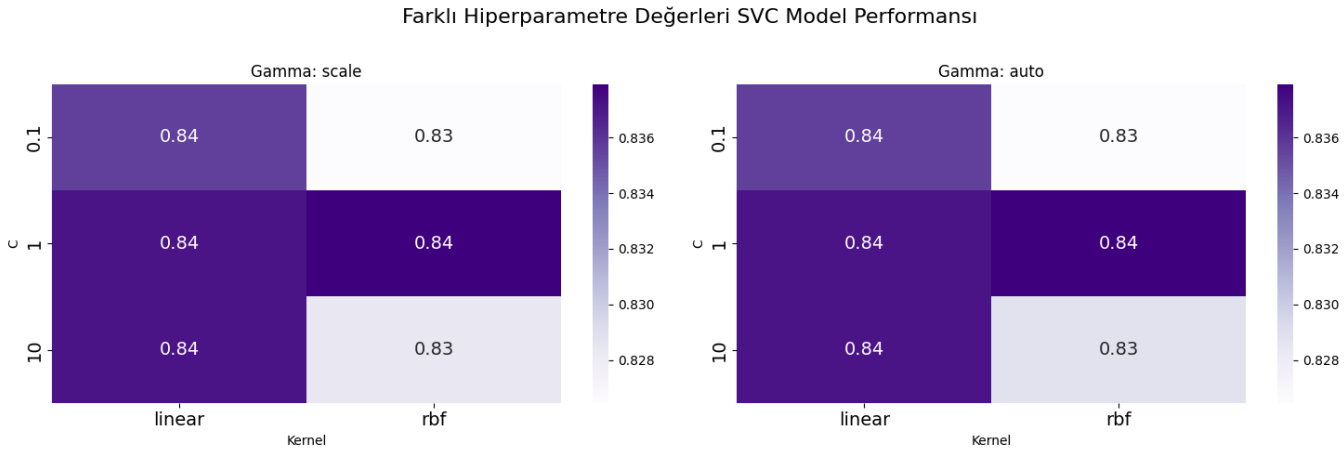
Lojistik Regresyon:



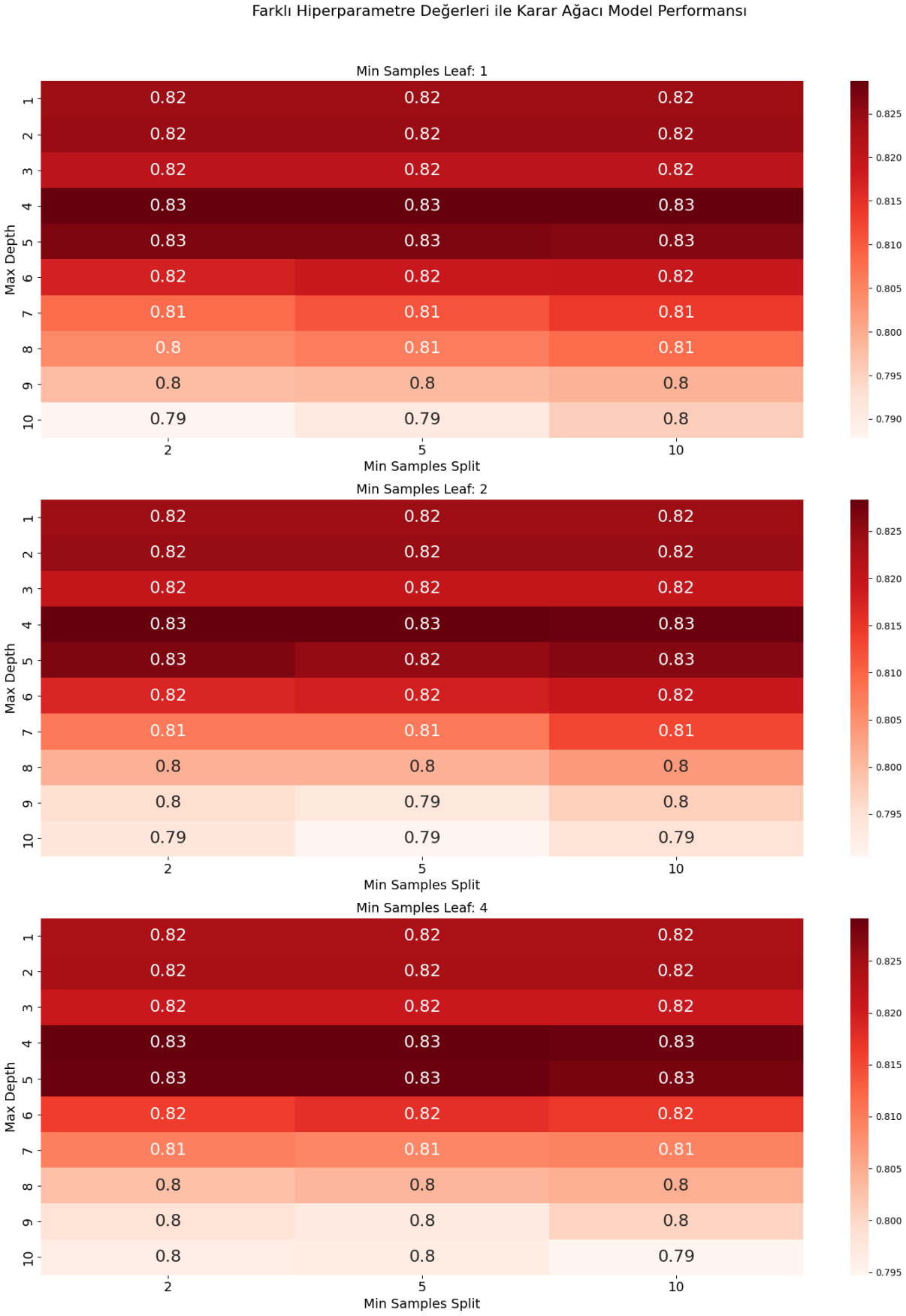
KNN:


Farklı Hiperparametre Değerleri ile KNN Model Performansı
Weights: uniform
Weights: distance
n_neighbors
Algorithm
auto
ball_tree
0.83

SVC:

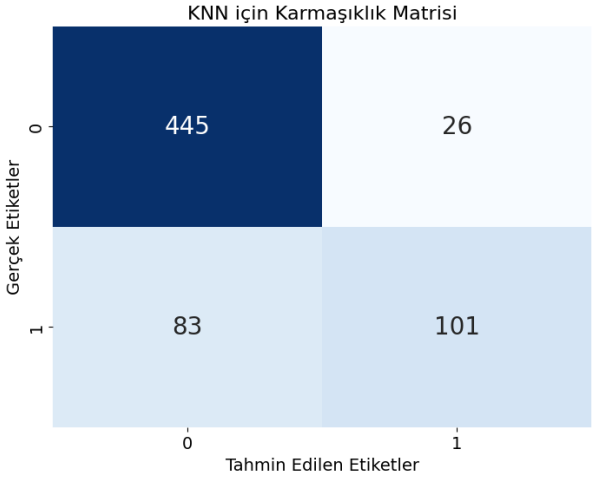

Farklı Hiperparametre Değerleri SVC Model Performansı
Gamma: scale
Gamma: auto
C
Kernel
linear
rbf
0.84
0.83

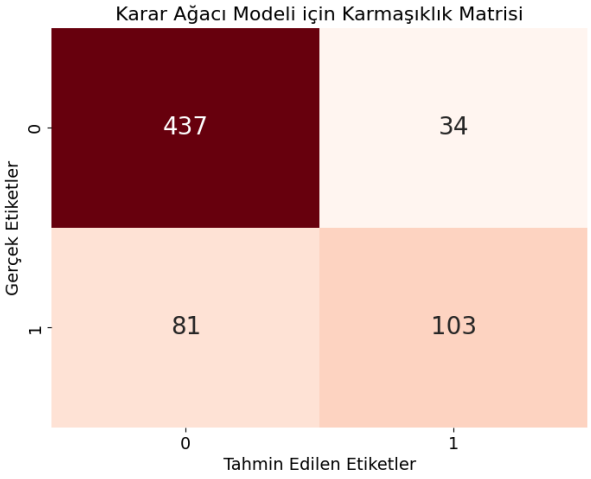
Karar Ağacı:

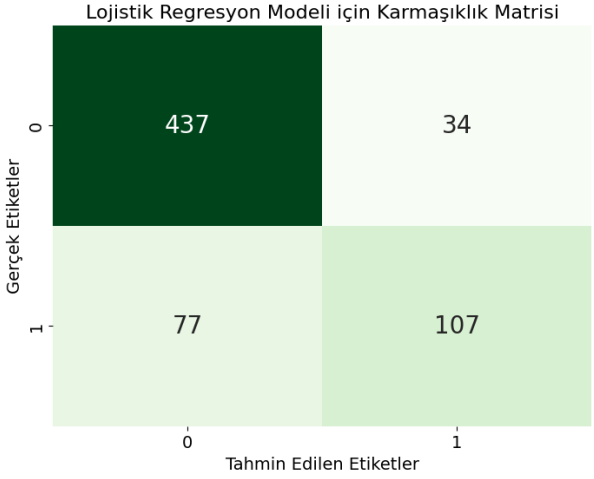

Farklı Hiperparametre Değerleri ile Karar Ağacı Model Performansı
Min Samples Leaf: 1
Max Depth
Min Samples Split
1 0.82 0.82 0.82
2 0.82 0.82 0.82
3 0.82 0.82 0.82
4 0.83 0.83 0.83
5 0.83 0.83 0.83
6 0.82 0.82 0.82
7 0.81 0.81 0.81
8 0.8 0.81 0.81
9 0.8 0.8 0.8
10 0.79 0.79 0.8
2 5 10
0.825 0.820 0.815 0.810 0.805 0.800 0.795 0.790
Min Samples Leaf: 2
Max Depth
Min Samples Split
1 0.82 0.82 0.82
2 0.82 0.82 0.82
3 0.82 0.82 0.82
4 0.83 0.83 0.83
5 0.83 0.82 0.83
6 0.82 0.82 0.82
7 0.81 0.81 0.81
8 0.8 0.8 0.8
9 0.8 0.79 0.8
10 0.79 0.79 0.79
2 5 10
0.825 0.820 0.815 0.810 0.805 0.800 0.795
Min Samples Leaf: 4
Max Depth
Min Samples Split
1 0.82 0.82 0.82
2 0.82 0.82 0.82
3 0.82 0.82 0.82
4 0.83 0.83 0.83
5 0.83 0.83 0.83
6 0.82 0.82 0.82
7 0.81 0.81 0.81
8 0.8 0.8 0.8
9 0.8 0.8 0.8
10 0.8 0.8 0.79
2 5 10
0.825 0.820 0.815 0.810 0.805 0.800 0.795

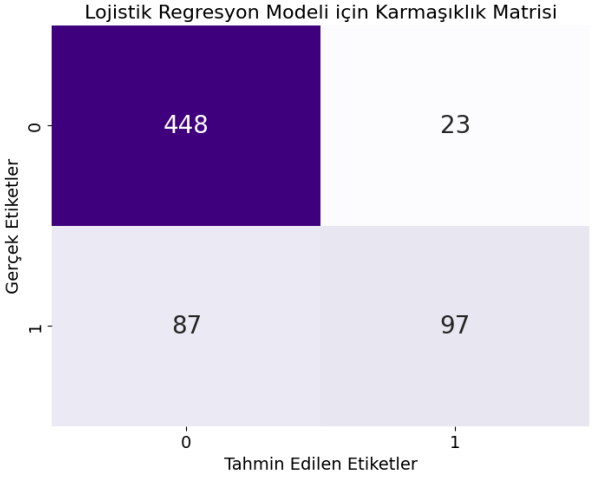
Karmaşıklık Matrisi (Confusion Matrix)

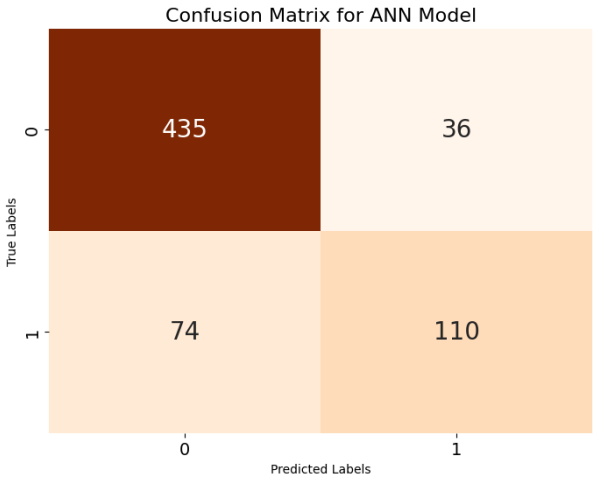
Karışıklık matrisi, modelin doğru ve yanlış sınıflandırmalarını gösterir. Her bir model için karışıklık matrisleri aşağıda verilmiştir.

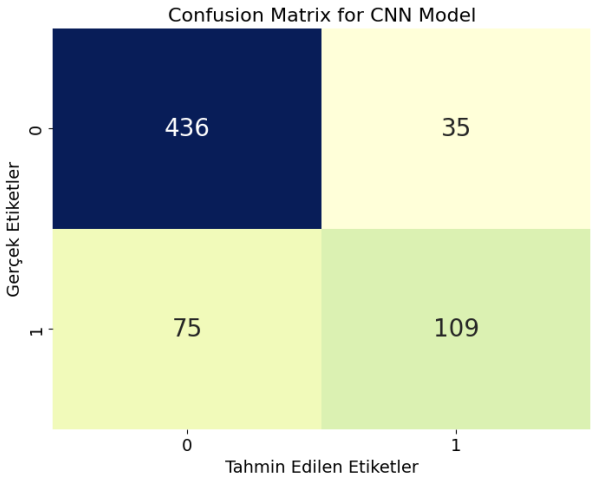






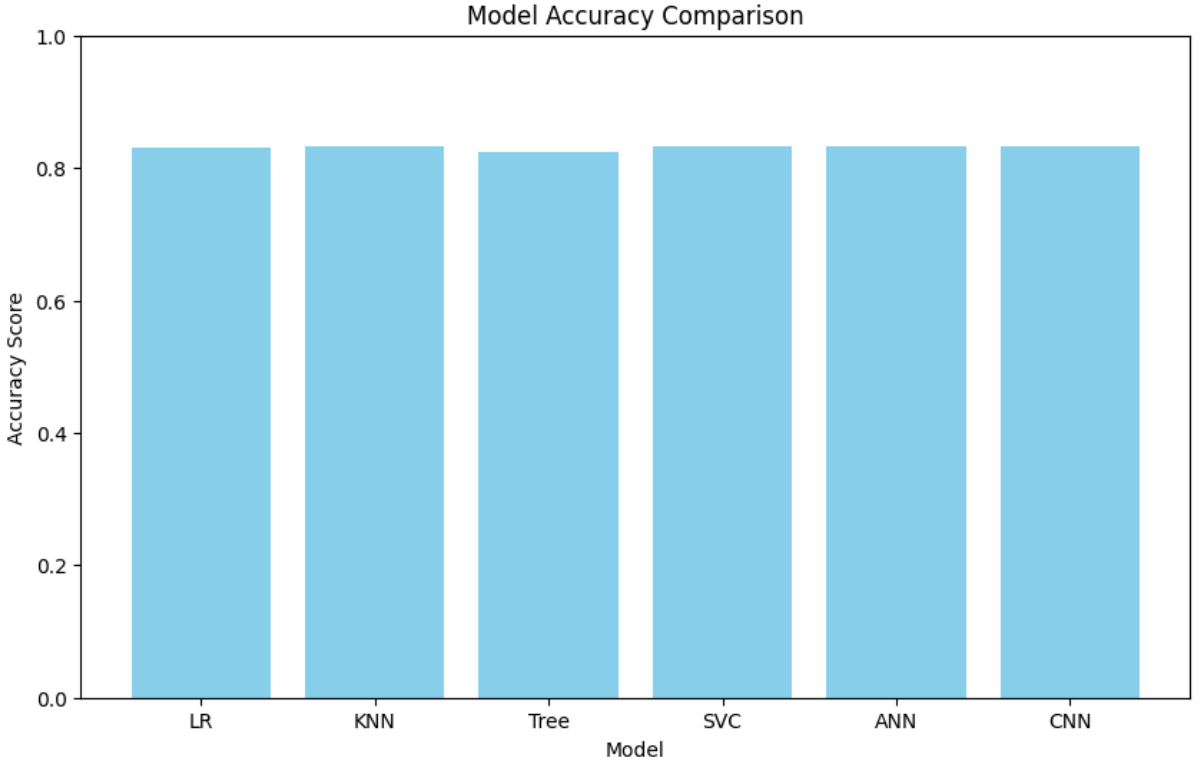


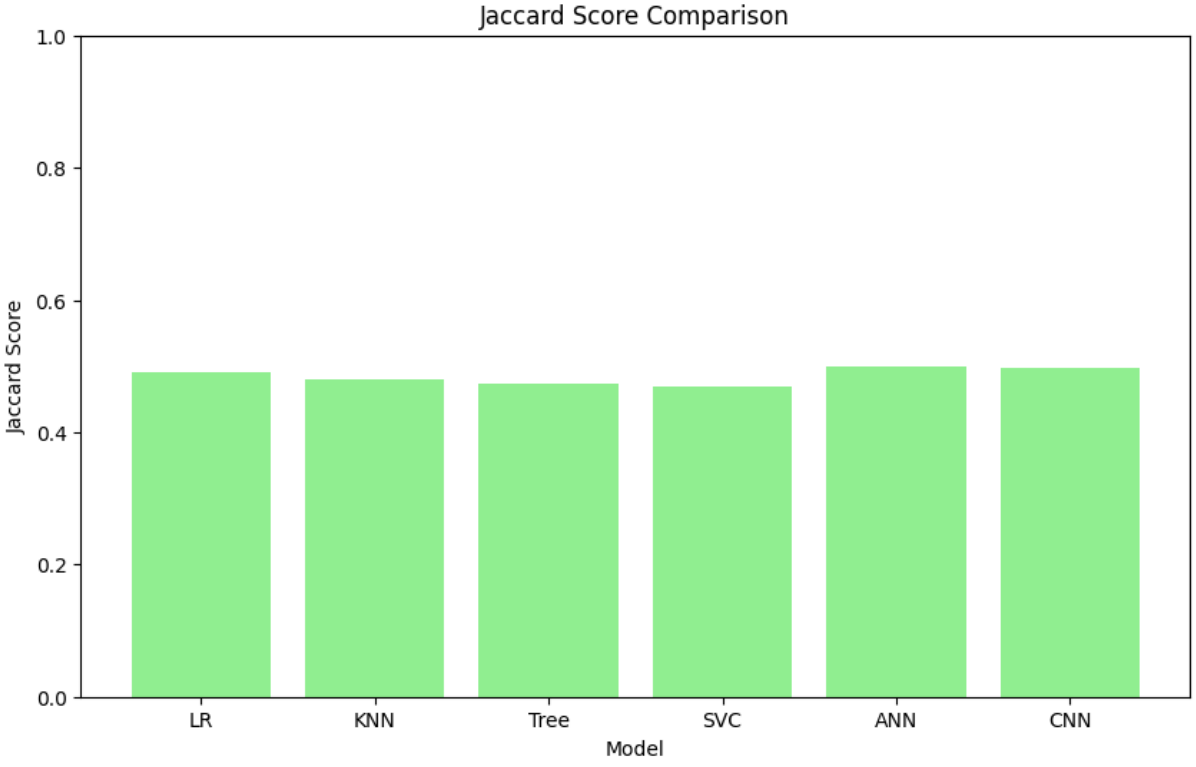


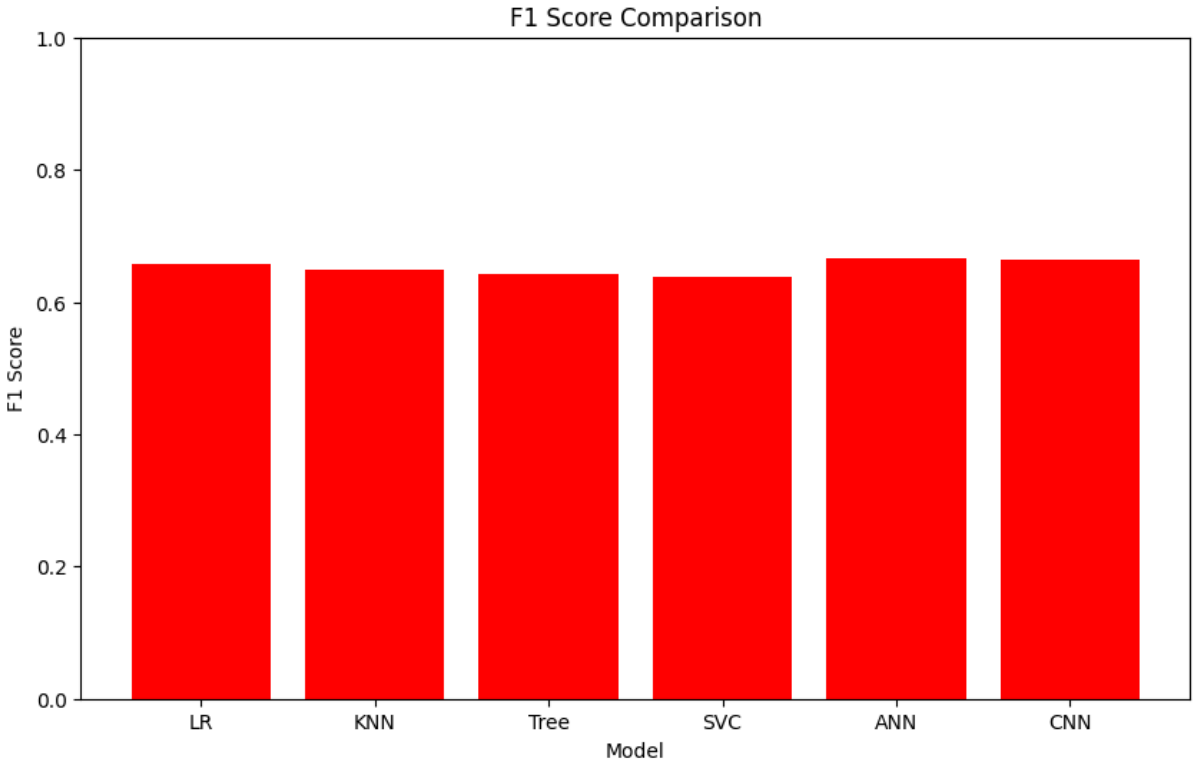


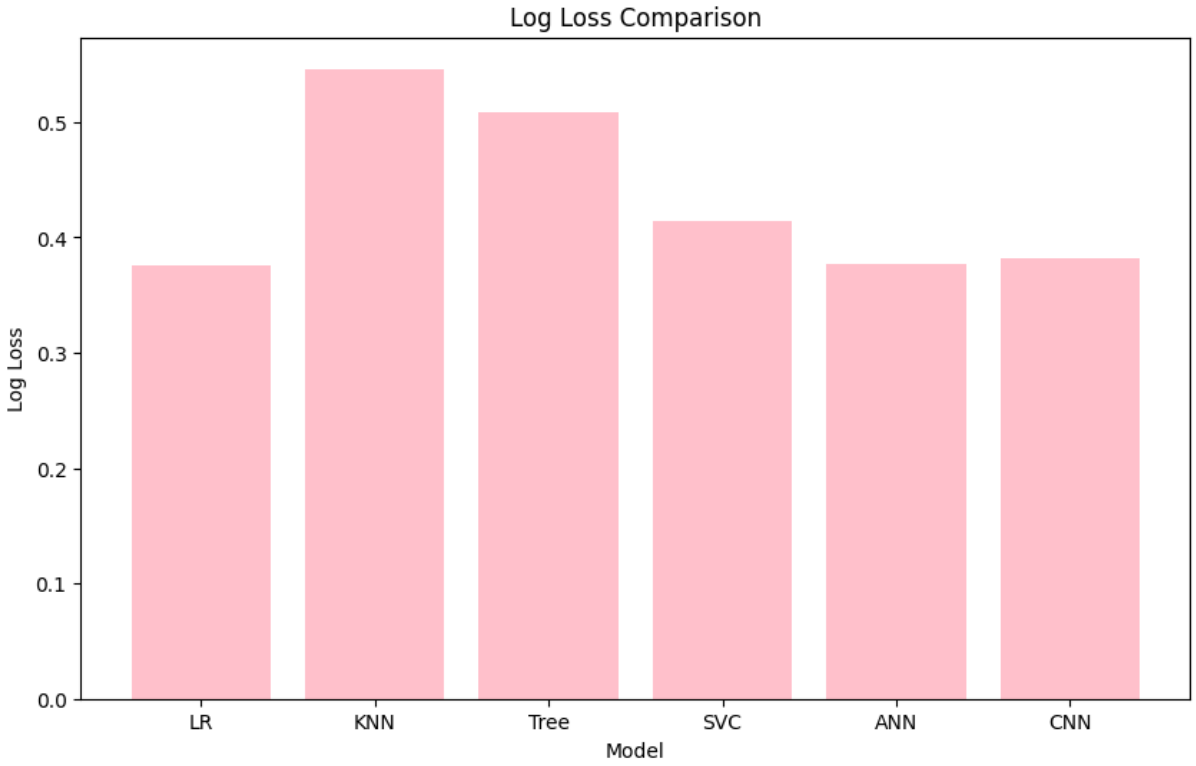
Modellerin Karşılaştırılması

Elde edilen performans metrikleri doğrultusunda modellerin karşılaştırılması yapılmıştır. Aşağıdaki grafikler, modellerin doğruluk (accuracy), F1 skoru, Jaccard indeksi ve Log kaybı değerlerini karşılaştırmaktadır.






F1 Score Comparison
F1 Score
1.0
0.8
0.6
0.4
0.2
0.0
LR
KNN
Tree
SVC
ANN
CNN
Model


Log Loss Comparison
Log Loss
0.5
0.4
0.3
0.2
0.1
0.0
LR
KNN
Tree
SVC
ANN
CNN
Model

Model Eğitim Detayları ve Sonuçlar

Her model için eğitim süreci, kullanılan veri seti ve sonuçları aşağıda detaylandırılmıştır.

KNN

- **Eğitim**: K-Nearest Neighbors (KNN) modeli, KNeighborsClassifier sınıfından oluşturulmuştur. Modelin performansını artırmak amacıyla veri ön işleme aşamasında StandardScaler kullanılarak veriler ölçeklendirilmiştir. Bu sayede, özelliklerin aynı ölçek aralığında olması sağlanmıştır. Modelin hiperparametre optimizasyonu için GridSearchCV kullanılmıştır. Bu optimizasyon süreci, 'n_neighbors' (20'den 30'a kadar), 'weights' (uniform, distance) ve 'algorithm' (auto, ball_tree) parametrelerinin farklı kombinasyonlarını denemeyi içermiştir. 5 katlı çapraz doğrulama (cv=5) kullanılarak en iyi performansı gösteren model seçilmiş ve en uygun hiperparametre ayarları algorithm: auto, n_neighbors: 23 ve weights: distance olarak belirlenmiştir.
- **Sonuçlar**: Sonuç olarak, en iyi model (KNN_best) elde edilerek test verileri üzerinde değerlendirilmiştir. KNN modeli, test verileri üzerinde doğruluk skoru, Jaccard indeksi, F1 skoru ve Log Loss metrikleriyle değerlendirilmiştir. Bu değerlendirmeler sonucunda, KNN modelinin özellikle doğruluk ve F1 skoru açısından iyi sonuçlar verdiği gözlemlenmiştir. Model, sınıflandırma problemlerinde güvenilir ve etkili bir performans sergilemiştir.

Karar Ağacı

- **Eğitim**: Karar ağacı modeli, DecisionTreeClassifier sınıfından oluşturuldu. Modelin performansını artırmak amacıyla GridSearchCV kullanılarak hiperparametre optimizasyonu gerçekleştirildi. Bu optimizasyon süreci, max_depth (1'den 10'a kadar), min_samples_split (2, 5, 10) ve min_samples_leaf (1, 2, 4) parametrelerinin farklı kombinasyonlarını denemeyi içerdi. 5 katlı çapraz doğrulama (cv=5) kullanılarak en iyi performansı gösteren model seçildi ve en uygun hiperparametre ayarları max_depth: 4, min_samples_split: 2 ve min_samples_leaf: 4 olarak belirlendi.
- **Sonuçlar**: Sonuç olarak, en iyi model (Tree_best) elde edildi ve test verileri üzerinde değerlendirildi. Karar ağacı modeli test verileri üzerinde doğruluk skoru, Jaccard indeksi, F1 skoru ve Log Loss metrikleriyle değerlendirildi. Bu değerlendirmeler sonucunda, karar ağacı modelinin özellikle doğruluk ve F1 skoru açısından iyi sonuçlar verdiği gözlemlendi. Model, sınıflandırma problemlerinde güvenilir ve etkili bir performans sergilemiştir.

Lojistik Regresyon

- **Eğitim**: Lojistik regresyon modeli, LogisticRegression sınıfından oluşturuldu. Modelin performansını artırmak amacıyla GridSearchCV kullanılarak hiperparametre optimizasyonu gerçekleştirildi. Bu optimizasyon süreci, C (0.1, 1, 10), solver (liblinear, lbfgs) ve max_iter (100, 200) parametrelerinin farklı kombinasyonlarını denemeyi içerdi. 5 katlı çapraz doğrulama (cv=5) kullanılarak en iyi performansı gösteren model

seçildi ve en uygun hiperparametre ayarları C: 10, max_iter: 100, solver: liblinear olarak belirlendi.

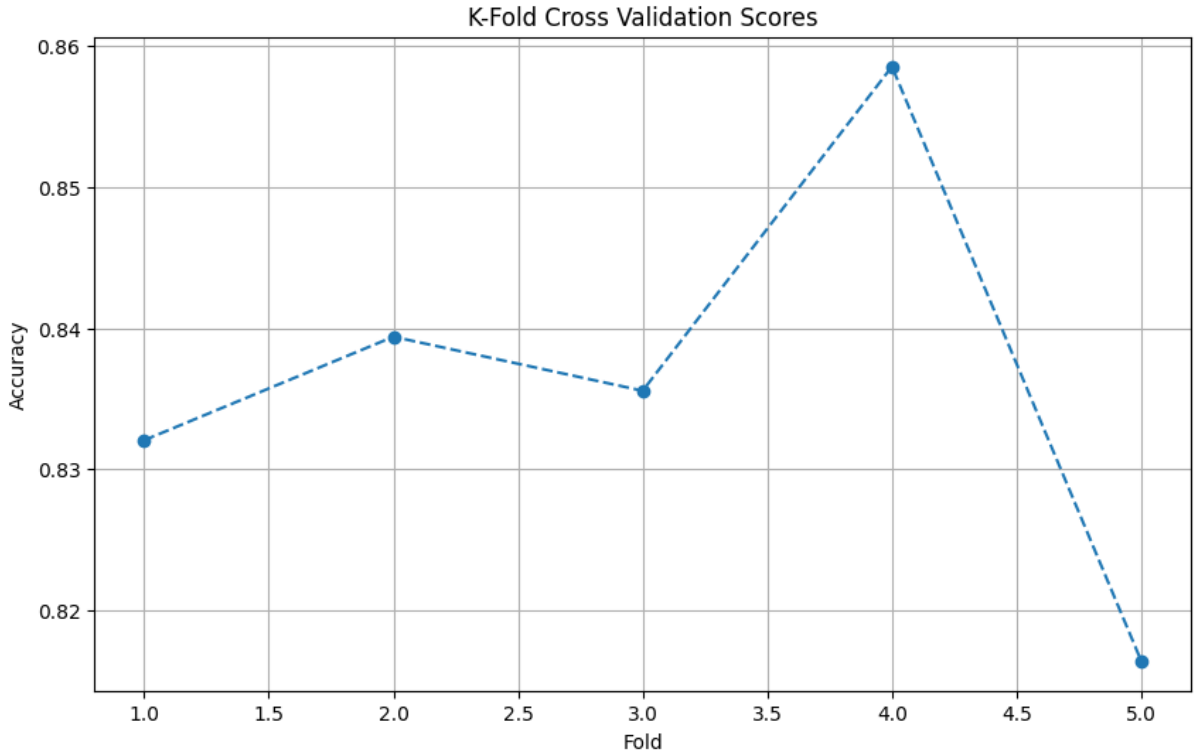
- **Sonuçlar**: Sonuç olarak, en iyi model (LR_best) elde edildi ve test verileri üzerinde değerlendirildi. Lojistik regresyon modeli test verileri üzerinde doğruluk skoru, Jaccard indeksi, F1 skoru ve Log Loss metrikleriyle değerlendirildi. Model, genellikle bütün metriklerde yüksek skorlar almayı başarmış ve log loss değeri de çok yüksek bulunmamıştır. Bu sebepten dolayı, model en iyi model olarak belirlenmiştir.

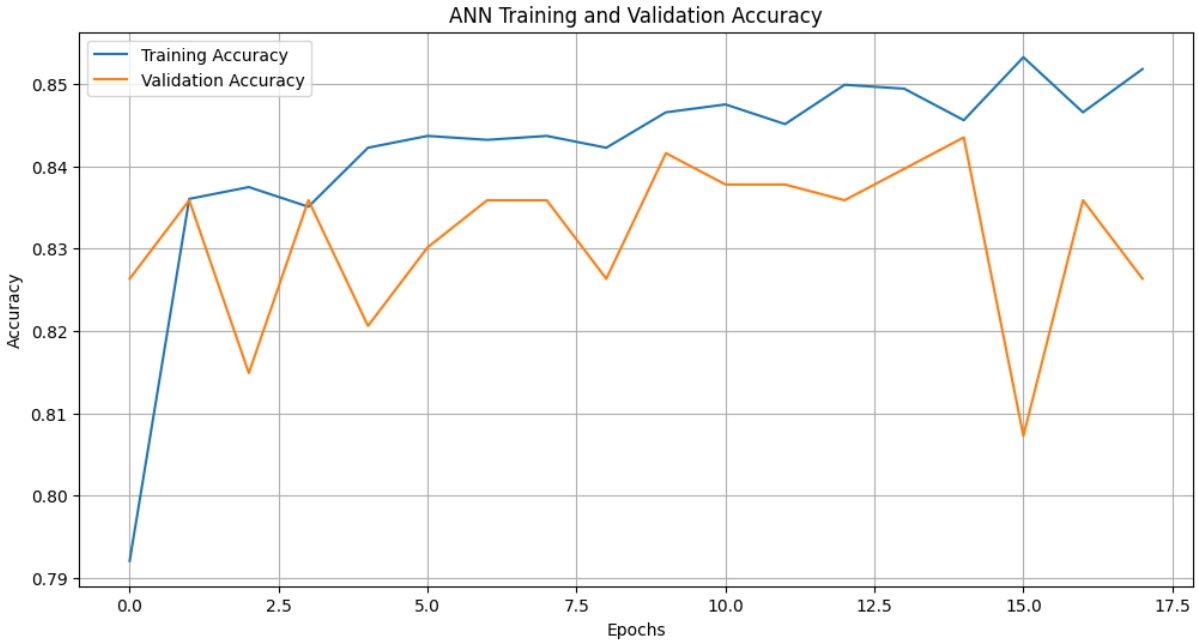
SVC

- **Eğitim**: Support Vector Classifier (SVC) modeli, SVC sınıfından oluşturuldu. Modelin performansını artırmak amacıyla GridSearchCV kullanılarak hiperparametre optimizasyonu gerçekleştirildi. Bu optimizasyon süreci, C (0.1, 1, 10), kernel (linear, rbf) ve gamma (scale, auto) parametrelerinin farklı kombinasyonlarını denemeyi içerdi. 5 katlı çapraz doğrulama (cv=5) kullanılarak en iyi performansı gösteren model seçildi ve en uygun hiperparametre ayarları C: 1, kernel: linear, gamma: scale olarak belirlendi.
- **Sonuçlar**: Sonuç olarak, en iyi model (SVC_best) elde edildi ve test verileri üzerinde değerlendirildi. SVC modeli, doğruluk skoru, Jaccard indeksi, F1 skoru ve Log Loss metrikleriyle test verileri üzerinde değerlendirildi. Model, genellikle tüm metriklerde yüksek skorlar almış ve log loss değeri düşük bulunmuştur. Bu sonuçlar, SVC modelinin sınıflandırma problemlerinde etkili bir performans sergilediğini göstermektedir. Bu sebepten dolayı, SVC modeli, mevcut veri seti ve problem için uygun bir model olarak belirlenmiştir.

ANN

- **Eğitim**: Veri setindeki kategorik değişkenler sayısal değerlere dönüştürülmüştür. Eksik değerler, ilgili sütunun ortalama değeri ile doldurulmuştur. Veriler standardize edilmiştir. Sequential modeli kullanılarak yapay sinir ağı (ANN) modeli yapılandırılmıştır. İlk katmanda, 64 nöron ve giriş boyutu olarak özelliklerin sayısına eşit olan bir input_dim parametresi belirtilmiştir. Ardından, iki gizli katman eklenmiş ve her katmanda ReLU aktivasyon fonksiyonu kullanılmıştır. Çıkış katmanında, sigmoid aktivasyon fonksiyonu ile ikili sınıflandırma yapılmıştır. Model, binary_crossentropy kayıp fonksiyonu ve Adam optimize edici ile derlenmiştir. Model, 50 epoch ve 10'arlık mini partiler halinde verilerle eğitilmiştir. Eğitim verisinin %20'si doğrulama seti olarak ayrılmıştır. EarlyStopping callback'i kullanılarak, doğrulama kaybının iyileşmediği durumda erken durdurma uygulanmıştır. Model, 5 katlı çapraz doğrulama (K-Fold Cross Validation) ile değerlendirilmiş ve her fold için doğruluk skoru kaydedilmiştir.
- **Sonuçlar**: Yapay sinir ağı modeli, doğruluk ve diğer metrikler açısından tatmin edici sonuçlar vermiştir. Model eğitilirken 18. Epoch'tan overfit tespit edilmiş ve eğitim sonlanmıştır. Çapraz doğrulama sonuçlarında her fold için doğruluk skorları 0.8320, 0.8393, 0.8355, 0.8585 ve 0.8164 olarak ölçülmüştür.



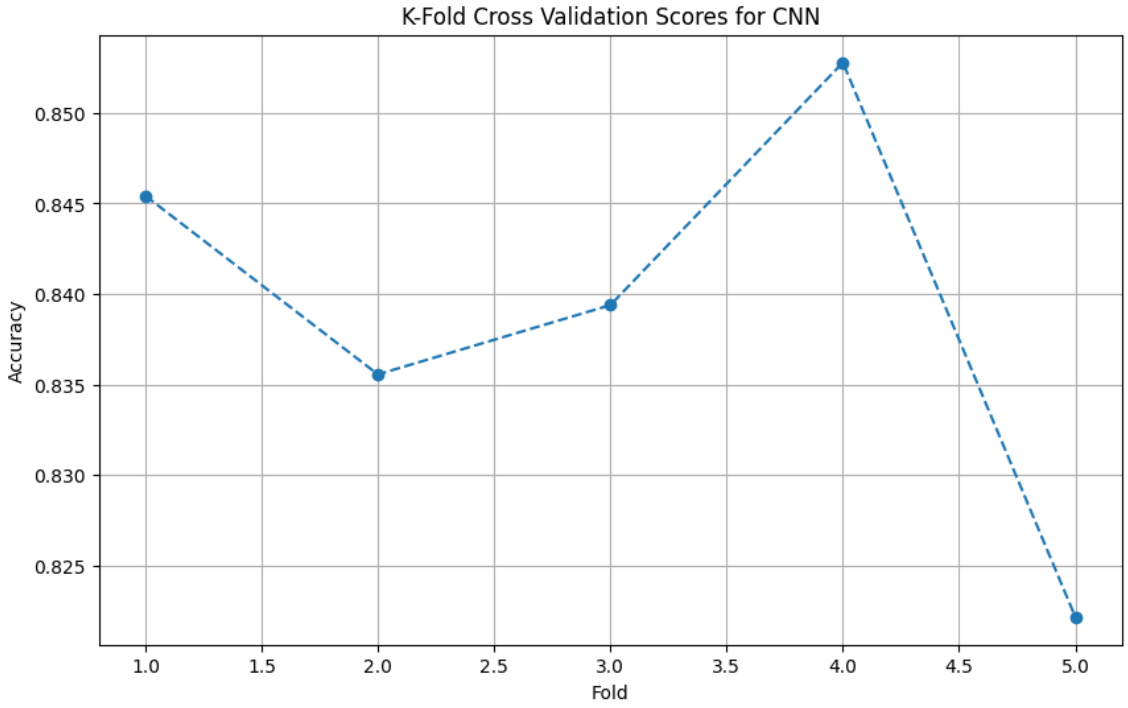


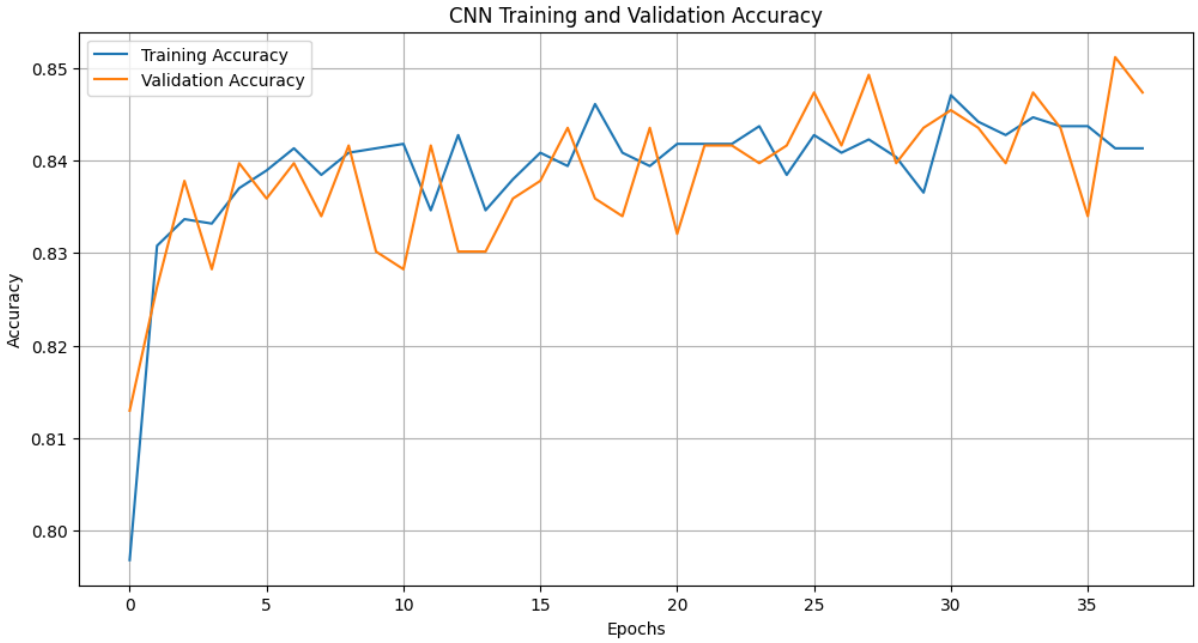
CNN

- **Eğitim**: Veri setindeki kategorik değişkenler sayısal değerlere dönüştürülmüştür. Eksik değerler, ilgili sütunun ortalama değeri ile doldurulmuştur. Veriler standardize edilmiştir. CNN modeli için veriler 3 boyutlu hale getirilmiştir (np.expand_dims kullanılarak). Sequential modeli kullanılarak yapay sinir ağı (CNN) modeli yapılandırılmıştır. İlk katmanda, 32 filtreli ve 3 boyutlu kernel kullanılarak bir Conv1D

katmanı eklenmiştir. Bu katman, ReLU aktivasyon fonksiyonunu kullanır ve giriş şekli (özellik sayısı, 1) olarak belirtilmiştir. Ardından, MaxPooling1D katmanı ile boyut indirgeme yapılmıştır. Flatten katmanı ile veriler düzleştirilmiştir. İki Dense katmanı eklenmiştir: biri 10 nöronlu ve ReLU aktivasyon fonksiyonlu, diğeri ise sigmoid aktivasyon fonksiyonlu ve 1 nöronlu çıkış katmanı. Model, binary_crossentropy kayıp fonksiyonu ve Adam optimize edici ile derlenmiştir. Model, 50 epoch ve 10'arlık mini partiler halinde verilerle eğitilmiştir. Eğitim verisinin %20'si doğrulama seti olarak ayrılmıştır. EarlyStopping callback'i kullanılarak, doğrulama kaybının iyileşmediği durumda erken durdurma uygulanmıştır. Model, 5 katlı çapraz doğrulama (K-Fold Cross Validation) ile değerlendirilmiş ve her fold için doğruluk skoru kaydedilmiştir.

- **Sonuçlar**: Evrişimli sinir ağı modeli, özellikle karmaşık veri setleri için yüksek performans göstermiştir. Model eğitilirken 38. Epoch'ta overfit tespit edilmiş ve eğitim sonlanmıştır. Çapraz doğrulama sonuçlarında her fold için doğruluk skorları 0.8454, 0.8355, 0.8393, 0.8527





ve 0.8221 olarak ölçülmüştür.

SONUÇ

Bu proje, yağış tahmini için farklı makine öğrenimi ve derin öğrenme modellerinin performansını karşılaştırmayı amaçlamıştır. Araştırma sorusu, "Hangi makine öğrenimi veya derin öğrenme modeli yağış tahmini için en yüksek doğruluğu sağlar?" olarak belirlenmiştir. Proje süresince, KNN, Karar Ağacı, Lojistik Regresyon, SVC, ANN ve CNN modelleri kullanılarak yapılan tahminlerin doğruluğu, Jaccard İndeksi, F1 Skoru ve Log Loss metrikleri ile değerlendirilmiştir.

Derin öğrenme modelleri olan ANN ve CNN, özellikle karmaşık veri setlerinde en yüksek doğruluğu sağlamıştır. Jaccard İndeksi ve F1 Skoru değerlendirildiğinde, ANN ve CNN modellerinin dengesiz veri kümelerinde de yüksek performans gösterdiği gözlemlenmiştir. Log Loss metriği açısından ANN ve CNN modelleri, daha düşük değerler elde etmiş ve bu da modelin olasılık tahminlerinin daha doğru olduğunu göstermiştir.

Gelecekte daha geniş veri kümeleri ve ek özellikler incelenmelidir. Hiperparametre ayarlarının daha ayrıntılı olarak optimize edilmesi önerilir. Farklı modellerin birlikte kullanılmasıyla ansamble yöntemler denenmelidir. Çalışmanın tarım, ulaşım, enerji yönetimi ve afet hazırlıkları gibi alanlarda pratik uygulamaları test edilmelidir.

Bu proje, yağış tahmininde yapay zeka tekniklerinin etkinliğini göstermiş ve hangi modellerin hangi koşullarda daha iyi performans gösterdiğini ortaya koymuştur. Sonuç olarak, ANN ve CNN modelleri, karmaşık ve dengesiz veri kümelerinde yüksek doğruluk ve düşük hata oranları ile en iyi performansı sağlamıştır.

KAYNAKÇA

https://www.citationmachine.net/ieee

Kaynakça Örneği:

[1] American Diabetes Association. Diagnosis and Classification of Diabetes Mellitus. Diabetes Care 1 2014;37(Supplement_1):81–90.

[2] Kılıç V. Yapay Zeka Tabanlı Akıllı Telefon Uygulaması ile Kan Şekeri Tahmini. Avrupa Bilim ve Teknoloji Dergisi 2021;26:289-294.

[3] Özsezer G, Mermer G. Diabetes Risk Prediction with Machine Learning Models. Artificial Intelligence Theory and Applications 2022;2(2):1-9.

[4] Li J, Huang J, Zheng L and Li X. Application of Artificial Intelligence in Diabetes Education and Management: Present Status and Promising Prospect. Front. Public Health 2020;8:173.

[5] Yahyaoui A, Jamil A, Rasheed J and Yesiltepe M. A Decision Support System for Diabetes Prediction Using Machine Learning and Deep Learning Techniques. UBMYK 2019;1-4.

[6] Pérez-Gandía C, Facchinetti A, Sparacino G, Cobelli C, Gómez E.J, Rigla M, de Leiva A, Hernando M.E. Artificial Neural Network Algorithm for Online Glucose Prediction from Continuous Glucose Monitoring. Diabetes Technology & Therapeutics 2010;12(1):81-88.

[7] Ling P, Luo S, Yan J, Zheng X, Yang D, Zeng X, et al. The design and preliminary evaluation of a mobile health application TangTangQuan in management of type 1 diabetes in China. Am Diabetes Assoc 2018;67(Supplement 1).

[8] Browlee J. Long Short-Term Memory Networks With Python. Machine Learning Mastery 2017;10-11.