**Makine Öğrenme Teknikleriyle Yenilebilir ve Zehirli Mantar Sınıflandırması**

**Classification of Edible and Poisonous Mushrooms Using Machine Learning Techniques**

*Alican İnan, İrem Kılıçarslan, Gülhan Çalışır*

[*alican.inan@gazi.edu.tr*](mailto:alican.inan@gazi.edu.tr)*,* [*23181616022@gazi.edu.tr*](mailto:23181616022@gazi.edu.tr)*,* [*gulhan.calisir@gazi.edu.tr*](mailto:gulhan.calisir@gazi.edu.tr)

*Bilgisayar Mühendisliği, Teknoloji Fakültesi, Gazi Üniversitesi, Ankara, Türkiye*

**Öz:**

Dünya genelinde milyonlarca farklı mantar türü yetişmekte olup, bu türlerin yenilebilir ya da zehirli olup olmadığını belirlemek, morfolojik benzerlikler nedeniyle hâlâ çözülmesi güç bir problem olarak karşımıza çıkmaktadır. Yanlış sınıflandırmalar, ciddi zehirlenmelere ve hatta ölümle sonuçlanabilecek durumlara yol açabilmektedir. Mantarların sınıflandırılması için çeşitli yöntemler geliştirilmiş olsa da, bu yöntemlerin büyük çoğunluğu manuel süreçlere dayandığından, özellikle geniş ölçekli uygulamalarda zaman ve emek açısından verimsiz kalmaktadır. Bu bağlamda, makine öğrenmesi algoritmaları, otomatik ve yüksek doğruluklu sınıflandırmalar gerçekleştirme potansiyeli sayesinde önemli bir alternatif sunmaktadır. Bu çalışmada, UCI veri deposunda yer alan *Secondary Mushroom* veri seti kullanılarak 61.068 örnek üzerinde analiz gerçekleştirilmiştir. Her biri 20 kategorik özelliğe sahip bu veriler, yedi farklı makine öğrenme algoritması (Decision Tree, k-NN, Naive Bayes, SVM, Random Forest, ANN, Logistic Regression) ile işlenmiş ve elde edilen modellerin sınıflandırma başarıları karşılaştırılmıştır. Deneysel sonuçlara göre, **%99.66** doğruluk ile en başarılı yöntem **k-NN** algoritmasıolmuş ve böylece bu yöntemin mantar türlerinin sınıflandırılmasında etkili bir araç olabileceği ortaya konmuştur.

**Anahtar Kelimeler:** Mantar Sınıflandırması, Makine Öğrenmesi, Yenilebilir ve Zehirli Mantarlar, Secondary Mushroom Veri Seti, Denetimli Öğrenme

**Abstract:**  
Millions of different mushroom species grow across the world, yet determining whether these species are edible or poisonous remains a challenging problem due to their morphological similarities. Misclassification can lead to serious poisoning incidents or even fatal outcomes. Although various classification methods have been developed, most are manual and become inefficient in large-scale applications due to their time and labor demands. In this context, machine learning algorithms offer a significant alternative by enabling automatic and highly accurate classification. In this study, 61,068 samples from the *Secondary Mushroom* dataset available in the UCI Machine Learning Repository were analyzed. These data, each with 20 categorical attributes, were processed using seven different machine learning algorithms, and the classification performances of the resulting models were compared. According to the experimental results, the **k-NN** algorithm achieved the highest classification accuracy of **%99.66**, indicating its potential as an effective tool for the classification of mushroom species.

**Keywords:** Mushroom Classification, Machine Learning, Edible vs. Poisonous Mushrooms, Secondary Mushroom Dataset, Supervised Learning

**1.Giriş:**

Mantarlar insan hayatının birçok alanında ihtiyaç duyulan ve farklı amaçlarla tüketilen maddelerden biri olarak karşımıza çıkmaktadır. Mantarlar üç ana gruba ayrılır: yenilebilir, tıbbi ve zehirli. Yenilebilir mantarlar, genellikle yenilebilir etleriyle tüketilirken, tıbbi mantarlar tedavi amaçlı biyolojik bileşenler içerir. Zehirli mantarlar ise insan sağlığına zarar verebilecek toksik bileşikler içerir. Hem yenilebilir hem de tıbbi mantarların insan sağlığı üzerinde faydalı etkileri olabilir, ancak bazı zehirli türler yanlışlıkla yenilebilir mantarlarla karıştırılabilir ve bu da sağlık sorunlarına yol açabilir [1].

Dünya genelinde yaklaşık 5,1 milyon mantar türü bulunduğu tahmin edilmekte olup, bugüne kadar yalnızca yaklaşık 70.000 tür isimlendirilmiş ve tanımlanmıştır [2]. Tanımlanan bu türlerin ise %1’inden daha azı zehirli olarak sınıflandırılmıştır. Buna rağmen, zehirli türlerin yenilebilir olanlarla morfolojik benzerlik göstermesi nedeniyle, mantar kaynaklı zehirlenme vakaları hem dünya genelinde hem de Türkiye’de yaygın biçimde görülmektedir. Bu vakaların çoğunun ciddi sağlık problemlerine ve bazı durumlarda ölümlere yol açtığı bilinmektedir [3]. Bu durum, insanların mantar tüketimi konusunda daha temkinli olmalarına neden olmaktadır.

Mantarların yenilebilir ya da zehirli olup olmadığını belirlemede kullanılan yöntemler çoğunlukla görsel tanıma teknikleri ile biyokimyasal analizlere dayanmaktadır [4]. Bu sebeple, günlük hayatta uzman olmayan bireylerin yalnızca morfolojik özellikleri değerlendirerek doğru bir sonuca varmaları çoğu zaman mümkün olmamaktadır. Bu nedenle son yıllarda, şapka şekli, koku ve lamel rengi gibi çeşitli morfolojik özelliklere dayanarak mantarların sınıflandırılmasında makine öğrenmesi algoritmaları yaygın biçimde kullanılmaya başlanmıştır [5][6]. Bu sayede, sınıflandırma sürecinin hem daha hızlı hem de daha yüksek doğruluk oranıyla gerçekleştirilmesi mümkün hale gelmiştir.

Bu çalışmada, literatürde mantar sınıflandırması üzerine yapılan önceki araştırmalar taranmış [7][8], kullanılan makine öğrenme algoritmaları ve elde edilen başarı oranları incelenmiştir [9]. Bu doğrultuda, ikili sınıflandırma problemi olarak ele alınan bu çalışmada, 61.068 örnek ve 20 özellikten oluşan veri seti kullanılarak mantarların yenilebilir veya zehirli olup olmadıkları yedi farklı makine öğrenme algoritması karşılaştırmalı olarak test edilmiştir. Elde edilen sonuçlar, modellerin performans ölçütleri bağlamında analiz edilerek en etkili yöntemler belirlenmiştir.

**2.Materyal ve Metot**

Bu bölümde veri seti, karışıklık matrisi, performans metrikleri ve makine öğrenmesi yöntemleri açıklanmıştır.

**2.1.Veri Seti**

Bu çalışmada kullanılan veri seti, mantar türlerinin yenilebilirlik durumuna göre sınıflandırılmasını amaçlayan *Secondary Mushroom Dataset*’tir. Veri seti, her biri "edible" (yenilebilir) veya "poisonous" (zehirli) olarak etiketlenmiş *61.068 gözlem* ve *20 açıklayıcı değişkenden* oluşmaktadır. Değişkenler; mantarların fiziksel ve kimyasal özelliklerini (şapka şekli, rengi, kokusu vb.) içeren kategorik verilerdir ve analiz öncesinde sayısal formata dönüştürülmüştür. Şekil 1.de görülen 18 nominal özelliğin dışında; stem width (sap genişliği) ve stem height (sap yüksekliği) olmak üzere iki adet numerik öznitelik de bulunmaktadır.

Şekil 1.Secondary Mushroom Dataseti için Nominal Öznitelikleri

A table with text on it

AI-generated content may be incorrect.

**2.2.Veri Ön İşleme**

İlk aşama olan veri ön işleme sürecinde, veri setinde bulunan eksik veriler uygun yöntemlerle doldurulmuş, ardından verinin kalitesini artırmak amacıyla aykırı değer analizi gerçekleştirilmiştir. Aykırı değerler, hem genel dağılımı bozarak modelin hatalı öğrenmesine yol açabileceği hem de ROC ve doğruluk skorlarında dengesizlik yaratabileceği için dikkatle incelenmiştir. Ancak, bu değerler doğrudan veri setinden çıkarılmamış, bunun yerine Min-MaxScaler yöntemi kullanılarak veriler belirli bir ölçeğe indirgenmiş ve aşırı değerlerin etkisi minimize edilmiştir. Böylece daha sağlıklı bir öğrenme ortamı sağlanmıştır.

Şekil 2. Aykırı Değer Analizi

A graph with different colored squares

AI-generated content may be incorrect.

Veri temizliğinin ardından, kategorik veriler önce Label Encoding ile sayısal forma dönüştürülmüş ancak bu yaklaşımın bazı modellerde olası bir overfitting durumuna neden olabileceği ihtimali göz önünde bulundurularak One Hot Encoding yöntemine geçilmiştir. Bu sayede olası bir overfitting durumunun önüne geçilmeye çalışılmıştır.

Veri setinin daha anlamlı bir şekilde görselleştirilmesi ve boyutsallığın azaltılması amacıyla *Temel Bileşenler Analizi (PCA)* uygulanmıştır. PCA sonucu elde edilen bileşenler, verideki genel sınıf ayrımını gözlemlemek için iki boyutlu düzlemde görselleştirilmiştir [10].

Şekil 3. PCA ile Görselleştirme

A screen shot of a graph

AI-generated content may be incorrect.

Ayrıca, sınıflandırma modellerinde kullanılacak değişkenlerin istatistiksel önemini değerlendirmek amacıyla *tek yönlü ANOVA testi* uygulanmış olup en anlamlı 15 öznitelik belirlenmiştir. ANOVA testi, her bir değişkenin sınıf etiketiyle olan ilişkisinin anlamlı olup olmadığını belirlemeye yardımcı olmuş ve önemli değişkenlerin seçilmesinde kullanılmıştır [11].

**2.2.Karışıklık Matrisi ve Performans Metrikleri**

Sınıflandırma modelinin performansı, örneklerin doğru ve yanlış sınıflandırılma durumlarının karşılaştırılmasıyla değerlendirilmektedir. Bu değerlendirme sürecinde en yaygın kullanılan araçlardan biri karışıklık matrisidir. Karışıklık matrisi, modelin tahmin ettiği sınıflarla test verisindeki gerçek sınıflar arasındaki ilişkiyi tablo şeklinde sunar. Matristeki satırlar modelin tahmin ettiği sınıfları, sütunlar ise gerçek sınıfları göstermektedir

[12].

Şekil 4. Karışıklık matrisi

A table with text on it

AI-generated content may be incorrect.

Model performansını ölçmek için kullanılan başlıca metrikler; doğruluk (accuracy), duyarlılık (recall/sensitivity), özgüllük (specificity), hassasiyet (precision), ROC (Receiver Operating Characteristic) eğrisi ve AUC (Area Under Curve) değerleridir. Bu metriklerin hesaplanmasında karışıklık matrisinde yer alan dört temel değer kullanılır: TP (True Positive), TN (True Negative), FP (False Positive) ve FN (False Negative).

A table with math equations

AI-generated content may be incorrect.

**2.3.Sınıflandırma Algoritmaları**  
Bu çalışmada, literatürdeki yaygın makine öğrenme algoritmalarından yedi tanesi, veri setimiz üzerinde test edilmek üzere seçilmiştir. Bu algoritmalar arasında Karar Ağaçları, k-NN (K-Nearest Neighbors), Naive Bayes, Destek Vektör Makineleri (SVM), Rastgele Orman, Yapay Sinir Ağları (ANN) ve Lojistik Regresyon yer almaktadır [13].

**2.3.1.Karar Ağaçları**

Karar ağaçları, sınıflandırma ve regresyon problemlerinde yaygın olarak kullanılan ve kolayca oluşturulup yorumlanabilen denetimli öğrenme algoritmalarından biridir. Bu algoritma, veri setinin özelliklerine göre dallanan bir ağaç yapısı oluşturur; kök düğümden başlayıp karar düğümleri ve yaprak düğümleri aracılığıyla sınıflandırma işlemini gerçekleştirir. Karar ağaçları, her düğümde belirli bir özelliğe dayanarak veriyi bölerek sonuca ulaşır. Bu özelliklerin seçimi, genellikle bilgi kazancı (information gain), kazanım oranı (gain ratio) ve Gini indeksi gibi ölçütlerle yapılır.

Veri setinin karmaşıklığına göre ağaç büyüyebilir; ancak en küçük boyutlu ağaç tercih edilir, böylece modelin aşırı uyum yapması engellenir. Karar ağacı modelinin öğrenme süreci, veri setindeki elemanların sınıflara ayrılmasıyla sonlanır. Bu yöntem, özellikle verinin anlaşılabilir ve yorumlanabilir olmasını sağlayarak sınıflandırma süreçlerinde etkin bir çözüm sunar [14].

**2.3.2.k-NN (K-Nearest Neighbors)**

k-En Yakın Komşuluk (k-NN) algoritması, bir gözlemi sınıflandırırken, test verisindeki gözlemin eğitim verisindeki her bir gözlemle olan uzaklığını hesaplar. Ardından, test verisindeki gözlem, en yakın k komşusunun çoğunluğuna göre sınıflandırılır. Bu sınıflandırma, en yakın k komşusunun ait olduğu sınıfı belirleyerek yapılır. k=1 durumunda, sınıflandırma yalnızca en yakın komşuya dayanırken, k arttıkça sınıflandırma daha genelleşir ve daha stabil hale gelir. k'nin tek sayı olarak seçilmesi tercih edilir, çünkü çift sayı durumunda eşitlik durumu ortaya çıkabilir. Çalışmalar, genellikle k=1, k=3 ve k=5 değerlerinde en iyi sonuçların elde edildiğini göstermektedir. Algoritmanın performansı, k değeri kadar, uzaklık hesaplama yöntemine de bağlıdır. Farklı uzaklık hesaplama yöntemleri kullanılarak algoritmanın başarısı karşılaştırılabilir [15].

**2.3.3.Naive Bayes**

Naive Bayes, sınıf üyelik olasılıklarına dayalı bir istatistiksel sınıflandırıcıdır ve sınıf tahminlerini Bayes teoremi kullanarak yapar. Bu algoritmanın temelinde, eğitim verisinin çeşitli kombinasyonlarına ilişkin sıklıkların hesaplanması yer alır. Naive Bayes'in avantajı, kolay uygulanabilir olması ve genellikle başarılı sonuçlar vermesidir. Ancak dezavantajı, sınıf ve özellikler arasındaki bağımsızlık varsayımına dayanmasıdır. Bu algoritma, olayların birbirinden bağımsız olduğu varsayımı altında çalışır ve koşullu olasılık hesaplamalarıyla sınıflandırma yapar [16].

**2.3.4. Destek Vektör Makineleri (SVM)**

Destek Vektör Makineleri (SVM), denetimli öğrenme yöntemlerinden biri olarak sınıflandırma problemlerinde kullanılır. Bu yöntem, veri noktalarının bir doğrusal veya doğrusal olmayan hiper düzlem ile ayrılmasını sağlayarak sınıflandırma yapar. İstatistiksel öğrenme teorisi ve yapısal risk minimizasyonu ilkesine dayalı olarak sınıflandırma ve regresyon problemlerine çözüm sunar. SVM, yüksek boyutlu verilere karşı dayanıklıdır ve genelleme yeteneği oldukça yüksektir. Ancak, eğitim süresi genellikle uzun olup, performansı parametre seçimlerine bağlı olarak değişkenlik gösterebilir [17].

**2.3.5.Rastgele Orman**

Rastgele Orman, birden fazla karar ağacı içeren ve doğru sınıflama oranlarını artıran etkili bir sınıflandırma yöntemidir. Yeni bir örneğin sınıflandırılmasında, her ağaç bağımsız olarak sınıflandırma yapar ve bu süreç "ağaç oylaması" olarak adlandırılır. Rastgele Orman, özellikle çok sayıda girdi değişkenine sahip büyük veri setlerinde yüksek performans sergilemekle birlikte, eksik verilerle de iyi sonuçlar verebilir. Ayrıca, bu algoritma örnekleme boyutunu belirleyerek her sınıftan alınacak örnek sayısını "yerine koyma" yöntemiyle belirler. Bu parametre, rastgeleliğin etkisini artırarak her ağacın verinin farklı yönlerini görmesini sağlar. Dengesiz sınıf dağılımına sahip veri kümelerinde, bu özellik, sınıf dengesizliği sorununu çözmeye ve modelin genel performansını iyileştirmeye yardımcı olur [18].

**2.3.6. Yapay Sinir Ağları (ANN)**

Yapay Sinir Ağları (ANN), biyolojik sinir sistemlerinden esinlenerek geliştirilmiş, çok katmanlı bir yapıya sahip denetimli öğrenme yöntemleridir. Giriş verileri, gizli katmanlar aracılığıyla işlenip, çıkış katmanına iletilir. Öğrenme, ağırlıkların geri yayılım algoritmasıyla güncellenmesiyle gerçekleşir. ANN'ler doğrusal olmayan ilişkileri modelleyebilme yeteneğine sahip olup, karmaşık sınıflandırma ve regresyon problemlerini çözmek için kullanılır. Ancak, büyük veri ve yüksek hesaplama gücü gereksinimi nedeniyle eğitim süreci zaman alıcı olabilir [19].

**2.3.7.Lojistik Regresyon**

Lojistik regresyon, gözlemlerin gruplara atanmasında kullanılan yaygın bir sınıflandırma yöntemidir. Bu analiz, veriler üzerinden sınıflandırma modeli oluşturur ve yeni gözlemler için sınıf tahminleri yapılmasını sağlar. Bağımlı değişkenin kategorik olduğu durumlarda, lojistik regresyon, bağımsız değişkenlerle arasındaki neden-sonuç ilişkisini analiz eder. Bu yöntem, özellikle bağımsız değişkenlerin modeldeki önemini değerlendirir ve modelin doğruluğunu artırır. Lojistik regresyon, tıp alanında sıklıkla, hastalık riski gibi faktörlerin erken tespiti ve önlenmesi amacıyla kullanılır [20].

**3.Bulgular ve Tartışma**

Bu çalışma, UCI Secondary Mushroom veri seti kullanılarak mantar türlerinin yenilebilir veya zehirli olarak sınıflandırılması amacıyla yedi farklı makine öğrenmesi algoritması oluşturmak ve sonuçları incelemek amacıyla uygulanmıştır. Bu algoritmalar şunlardır: Karar Ağacı (Decision Tree), K-En Yakın Komşu (k-NN), Naive Bayes, Destek Vektör Makineleri (SVM), Rastgele Orman (Random Forest), Yapay Sinir Ağı (ANN) ve Lojistik Regresyon.

Modeller eğitildikten sonra bazı modellerde gözlemlenen AUC değerlerinin çok yüksek çıkması dikkat çekmiştir. Özellikle k-NN, ANN ve Random Forest algoritmalarında AUC değerleri 1.00 seviyesine kadar ulaşmıştır. Bu durum, potansiyel bir overfitting (aşırı öğrenme) problemi olabileceğini düşündürmüştür. Aşırı öğrenmeyi önlemek ve modelin genelleme performansını artırmak amacıyla hiperparametre optimizasyonuna gidilmiştir. Bu süreçte; k-NN modelinde komşu sayısı (n\_neighbors) artırılmıştır, Random Forest modelinde ağaç sayısı ve maksimum derinlik sınırlandırılmıştır ve ANN modelinde gizli katman sayısı, nöron sayısı, öğrenme oranı gibi parametrelerde iyileştirmeler yapılmıştır.

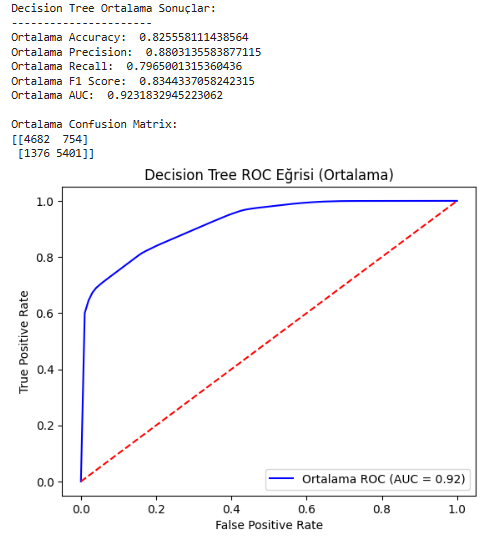
Tüm bu optimizasyonlara rağmen bazı modellerin AUC değerleri yüksek kalmaya devam etmiştir. Bu durumun temel nedeninin, veri setinin yapısından kaynaklandığı değerlendirilmiştir. Özellikle k-NN ve ANN modelleri, veri kümesindeki belirgin sınıf ayrımlarına çok hassas yanıt vererek keskin tahminler yapmıştır. Ancak eğitim ve test sonuçları karşılaştırıldığında, modelin genelleme performansının tatmin edici seviyede olduğu görülmüş ve aşırı öğrenme probleminin minimize edildiği sonucuna varılmıştır.

Aynı zamanda bu çalışmada modellerin değerlendirilmesi aşamasında 5 katlı çapraz doğrulama (5-Fold Cross Validation) yöntemi uygulanmıştır. Bu yöntem sayesinde; tüm veri kümesi rastgele ve eşit şekilde 5 parçaya bölünmüş, modeller bu parçalardan biri test, kalanları eğitim olmak üzere 5 kez çalıştırılmış ve tüm metriklerin ortalaması alınarak daha güvenilir ve genel sonuçlara ulaşılmıştır.

Çapraz doğrulama, özellikle yüksek AUC değerlerinin modelin genelleme başarısını ne derece temsil ettiğini anlamak ve rastgelelik kaynaklı sapmaları önlemek adına kritik öneme sahiptir. Ayrıca ROC eğrilerinde her bir fold sonucu interpolasyon (ara değer bulma) yöntemiyle birleştirilerek ortalama bir ROC eğrisi çizilmiş ve bu sayede görsel yorumlama kolaylaştırılmıştır.

Kullanılan bütün modellerin ROC eğrileri, hesaplanan eğri altındaki alan (AUC), accuracy, precision, recall, F1 score değerleri ve karışıklık matrisleri şekillerde verilmiştir.

Şekil 5. Decision Tree Modeli ROC Eğrisi



Şekil 6. k-NN Modeli ROC Eğrisi

metin, ekran görüntüsü, çizgi, öykü gelişim çizgisi; kumpas; grafiğini çıkarma içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

Şekil 7. Naive Bayes Modeli ROC Eğrisi

metin, ekran görüntüsü, çizgi, öykü gelişim çizgisi; kumpas; grafiğini çıkarma içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

Şekil 8. SVM Modeli ROC Eğrisi

metin, ekran görüntüsü, çizgi, diyagram içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

Şekil 8. Random Forest Modeli ROC Eğrisi

metin, ekran görüntüsü, çizgi, öykü gelişim çizgisi; kumpas; grafiğini çıkarma içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

Şekil 9. ANN Modeli ROC Eğrisi

metin, ekran görüntüsü, çizgi, öykü gelişim çizgisi; kumpas; grafiğini çıkarma içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

Şekil 10. Logistic Regression Modeli ROC Eğrisi

metin, ekran görüntüsü, çizgi, öykü gelişim çizgisi; kumpas; grafiğini çıkarma içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

Bütün modellerin accuracy, precision, recall, F1 score ve AUC değerleri performans özeti Şekil 11.de toplu bir şekilde verilmiştir.

Şekil 11. Modellerin Performans Özeti

A screenshot of a graph

AI-generated content may be incorrect.

**4.Sonuç**

Bu çalışma kapsamında, çeşitli makine öğrenmesi algoritmaları kullanılarak sınıflandırma başarısı detaylı bir şekilde incelenmiştir. Yapılan incelemeler sonucunda, mantar türlerinin yenilebilir veya zehirli olarak sınıflandırılmasında makine öğrenmesi algoritmalarının rahatlıkla kullanılabileceği sonucuna ulaşılmıştır.

Öncelikle veri ön işleme adımlarına büyük önem verilmiş, eksik ve aykırı veriler titizlikle düzeltilmiştir. Özellik seçimi ve boyut indirgeme teknikleri ile veri setinin daha anlamlı hale gelmesi sağlanmıştır.

Elde edilen bulgular ışığında Random Forest, ANN ve k-NN modelleri genel olarak en yüksek doğruluk ve AUC değerlerine ulaşmıştır. Ancak, özellikle k-NN ve ANN modellerinde AUC değerlerinin oldukça yüksek olması modelin overfitting yapmış olabileceğine dair işaretler vermiştir. Bu nedenle hiperparametre ayarlamaları ve K-Fold Cross Validation uygulanarak model genellenebilirliği artırılmıştır. Ayrıca modelin eğitim ve test performanslarının paralel seyretmesi, bu durumun aşırı öğrenmeden (overfitting) ziyade verinin yapısından kaynaklandığını göstermektedir.

Çalışmada kullanılan bütün modellerin accuracy, precision, recall, F1 score ve AUC değerlerinin karşılaştırması, Şekil 12.de açıkça gösterilmiştir.

Şekil 12. Modellerin Accuracy, Precision, Recall, F1 Score ve AUC Değerlerinin Karşılaştırması

A graph of different colored lines

AI-generated content may be incorrect.

%99.66 doğruluk oranı ile en başarılı algoritma k-NN olurken, onu %97.38 doğruluk oranıyla ANN ve %94.37 doğruluk oranı ile Random Forest algoritmaları takip etmektedir. Veriler ışığında, bu üç modelin diğer modellere kıyasla bu veri seti üzerinde daha yüksek performans sergilediği söylenebilir.

SVM ve Decision Tree modelleri de sırasıyla %83.60 ve %82.55 doğruluk oranları ile başarılı modeller arasında yer almıştır.

Doğruluk oranları sırasıyla %63.74 ve %59.38 olan Logistic Regression ve Naive Bayes algoritmalarının bu veri seti üzerinde pek başarılı olmadıkları söylenebilir, ayrıca bu modellerin karmaşıklık matrisleri ve ROC eğrileri de göz önüne alındığında sınıflandırma açısından daha düşük performans sergiledikleri açıkça görülmüştür. Ancak bu durum bir sorun teşkil etmediği gibi, modelin basit yapısı ve veri dağılımına duyarlılığı ile ilişkilendirilebilir.

k-NN, ANN ve Random Forest gibi algoritmaların yüksek doğruluk oranlarına ulaşmasında, veri setindeki örüntülerin belirginliği ve sınıfların ayrılabilirliği etkili olmuştur. k-NN algoritması, komşuluk ilişkileri üzerinden karar verdiği için özellikle benzer örüntülere sahip veriler üzerinde son derece başarılı çalışmaktadır. ANN ve Random Forest algoritmaları ise veri içindeki karmaşık ilişkileri öğrenebilme ve çok sayıda karar sınırı oluşturabilme yetenekleri sayesinde oldukça yüksek performans sergilemiştir.

Diğer taraftan, Naive Bayes ve Logistic Regression gibi algoritmaların daha düşük doğruluk oranlarına sahip olmasının temel nedeni, bu yöntemlerin veri yapısındaki karmaşıklığı ve ilişkileri sınırlı şekilde modelleyebilmesidir. Naive Bayes algoritması, tüm özelliklerin birbirinden bağımsız olduğunu varsayarak işlem yaptığı için kategorik ve karmaşık bağıntılar içeren bu veri setinde sınırlı kalmıştır. Logistic Regression ise doğrusal sınırlara dayalı çalıştığı için, doğrusal olarak ayrılması zor olan sınıflar arasında başarılı bir ayırım yapamamıştır. Bunun sonucunda her iki modelde de ROC eğrisi ve karmaşıklık matrisi analizlerinde nispeten düşük başarı oranları gözlemlenmiştir.

Sonuç olarak, her modelin kendine özgü avantaj ve dezavantajları bulunduğu, tek bir modelin tüm problemler için en iyi sonucu vermediği ve bundan dolayı model seçiminde yalnızca tek bir algoritmaya değil, çoklu model karşılaştırmasına dayalı karar verilmesi gerektiği sonuçlarına ulaşılmıştır. Çalışmada kullanılan cross-validation yöntemi ve hiperparametre ayarlamaları ile model doğruluğu ve kararlılığı artırılmış, aşırı öğrenme riski azaltılmıştır. Bu nedenle sınıflandırma problemlerinde tek bir algoritmaya bağlı kalmak yerine birden çok modelin denenmesi ve çapraz doğrulama gibi yöntemlerle modellerin tutarlılığınınölçülmesi gerekliliği ortaya çıkmıştır. Teorik bilgilerin yanı sıra ROC eğrileri, AUC değerleri ve karmaşıklık matrisleri gibi metriklerin kullanımı ile çok boyutlu bir performans değerlendirmesi yapılmıştır.

**Kaynakça**

[1] Cheung, P. C. (2010). The nutritional and health benefits of mushrooms. *Nutrition Bulletin*, *35*(4), 292-299.

[2] Blackwell, M. (2011). The Fungi: 1, 2, 3… 5.1 million species?. *American journal of botany*, *98*(3), 426-438.

[3] Eren, S. H., Demirel, Y., Ugurlu, S., Korkmaz, I., Aktas, C., & Güven, F. M. K. (2010). Mushroom poisoning: retrospective analysis of 294 cases. *Clinics*, *65*(5), 491-496.

[4] Yan, Z., Liu, H., Li, J., & Wang, Y. (2023). Application of identification and evaluation techniques for edible mushrooms: a review. *Critical Reviews in Analytical Chemistry*, *53*(3), 634-654.

[5] Tutuncu, K., Cinar, I., Kursun, R., & Koklu, M. (2022, June). Edible and poisonous mushrooms classification by machine learning algorithms. In *2022 11th Mediterranean Conference on Embedded Computing (MECO)* (pp. 1-4). IEEE.-\*

[6] Siddique, A. B., Bakar, M. A., Ali, R. H., Arshad, U., Ali, N., Abideen, Z. U., ... & Imad, M. (2023, October). Studying the effects of feature selection approaches on machine learning techniques for mushroom classification problem. In *2023 International Conference on IT and Industrial Technologies (ICIT)* (pp. 1-6). IEEE.

[7] Agarwal, A. K., Khullar, V., & Agrawal, N. K. (2021, August). Scalable Machine learning for Mushroom Dataset Classification. In *2021 International Conference on Simulation, Automation & Smart Manufacturing (SASM)* (pp. 1-4). IEEE.

[8] Morshed, M. S., Ashraf, F. B., Islam, M. U., & Shafi, M. S. R. (2023, January). Predicting Mushroom Edibility with Effective Classification and Efficient Feature Selection Techniques. In *2023 3rd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST)* (pp. 1-5). IEEE.

[9] Zahan, N., Hasan, M. Z., Malek, M. A., & Reya, S. S. (2021, February). A deep learning-based approach for edible, inedible and poisonous mushroom classification. In *2021 international conference on information and communication technology for sustainable development (ICICT4SD)* (pp. 440-444). IEEE.

[10] Abdi, H., & Williams, L. J. (2010). Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, *2*(4), 433-459.

[11] Kim, T. K. (2017). Understanding one-way ANOVA using conceptual figures. *Korean journal of anesthesiology*, *70*(1), 22.

[12] Luque, A., Carrasco, A., Martín, A., & de Las Heras, A. (2019). The impact of class imbalance in classification performance metrics based on the binary confusion matrix. *Pattern Recognition*, *91*, 216-231.

[13] Ayodele, T. O. (2010). Types of machine learning algorithms. *New advances in machine learning*, *3*(19-48), 5-1.

[14] Charbuty, B., & Abdulazeez, A. (2021). Classification based on decision tree algorithm for machine learning. *Journal of applied science and technology trends*, *2*(01), 20-28.

[15] Zhang, S., Li, X., Zong, M., Zhu, X., & Cheng, D. (2017). Learning k for knn classification. *ACM Transactions on Intelligent Systems and Technology (TIST)*, *8*(3), 1-19.

[16] Mitchell, T. M., & Mitchell, T. M. (1997). *Machine learning* (Vol. 1, No. 9). New York: McGraw-hill.

[17] Zhang, Y. (2012). Support vector machine classification algorithm and its application. In *Information Computing and Applications: Third International Conference, ICICA 2012, Chengde, China, September 14-16, 2012. Proceedings, Part II 3* (pp. 179-186). Springer Berlin Heidelberg.

[18] Liu, Y., Wang, Y., & Zhang, J. (2012, September). New machine learning algorithm: Random forest. In *International conference on information computing and applications* (pp. 246-252). Berlin, Heidelberg: Springer Berlin Heidelberg.

[19] Choi, R. Y., Coyner, A. S., Kalpathy-Cramer, J., Chiang, M. F., & Campbell, J. P. (2020). Introduction to machine learning, neural networks, and deep learning. *Translational vision science & technology*, *9*(2), 14-14.

[20] Hosmer Jr, D. W., Lemeshow, S., & Sturdivant, R. X. (2013). *Applied logistic regression*. John Wiley & Sons.

***GitHub Linki:*** https://github.com/Alican504/Veri-Bilimi-Proje