**Kredi Başvurularının Sınıflandırılması: UCI Credit Approval Veri Seti Üzerine Bir Karşılaştırmalı Modelleme Çalışması**

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

Enes Furkan YILMAZ Mehmet Ali GÜRLER Ali Fuat AKBAŞ

### **Özet**

Bu çalışma, bireylerin kredi başvuru sonuçlarını sınıflandırmak amacıyla UCI Machine Learning Repository’de yer alan Credit Approval veri seti üzerinde gerçekleştirilmiştir. Çalışmanın temel amacı, eksik ve kategorik veriler içeren gerçek dünya verileri üzerinde farklı makine öğrenmesi modellerinin doğruluk ve güvenilirlik açısından performanslarını karşılaştırmaktır.

Veri seti, eksik değerlerin uygun şekilde doldurulması ve kategorik değişkenlerin sayısal formata dönüştürülmesiyle ön işleme tabi tutulmuştur. Modelleme aşamasında altı farklı sınıflandırma algoritması uygulanmıştır: Karar Ağacı, Lojistik Regresyon, K-En Yakın Komşu (KNN), Naive Bayes, Yapay Sinir Ağı ve SVM. Her model, eğitim-veri bölünmesi sonrası test verileri üzerinde değerlendirilmiştir.

Modeller; doğruluk, kesinlik, geri çağırma, F1 skoru ve ROC-AUC metrikleri kullanılarak analiz edilmiştir. Elde edilen sonuçlara göre, en yüksek doğruluk oranı %84.5 ile SVM algoritmasında gözlemlenmiştir. Ancak, bu modelin özellikle yanlış pozitif oranı yüksek çıkmış ve bu durum kesinlik skorunu olumsuz etkilemiştir. Lojistik Regresyon modeli ise %84 doğruluk ve %0.90 AUC değeri ile dengeli bir performans göstererek ROC eğrisi altında kalan alan ve F1 skorlarında öne çıkmıştır. Bu durum, Lojistik Regresyon’un hem pozitif hem de negatif sınıfları ayırt etmede oldukça başarılı olduğunu ortaya koymaktadır.

Destek Vektör Makineleri (SVM) modeli, %83 doğruluk oranı ve 0.88 AUC değeri ile başarılı sonuçlar elde etmiştir. Özellikle doğrusal olmayan sınırları yakalama yeteneği sayesinde, veri setinin karmaşık yapısını iyi modelleyebilmiştir. Yapay Sinir Ağı (ANN) ise %82 doğruluk ve 0.87 AUC değeriyle sınıflandırma görevinde tatmin edici bir performans sergilemiştir. ANN’nin başarısı, katmanlı yapısı sayesinde verinin örüntülerini yakalama becerisinden kaynaklanmaktadır.

Karar Ağacı modeli ise %76 doğruluk ve %75 F1 skoru ile diğer modellere kıyasla nispeten daha düşük başarı göstermiştir. Naive Bayes modeli de benzer şekilde %75 doğruluk ile makul düzeyde bir performans sunmuş, ancak sınıf ayrımında özellikle geri çağırma ve AUC değerlerinde diğer modellere kıyasla daha düşük kalmıştır.

Sonuç olarak, ön işleme sürecinin model performansı üzerindeki etkisi açıkça gözlemlenmiş ve özellikle eksik verilerin uygun şekilde doldurulması, model başarısına doğrudan katkı sağlamıştır. KNN ve Lojistik Regresyon modelleri, bu tür sınıflandırma problemlerinde etkili alternatifler sunarken, SVM ve ANN gibi daha karmaşık yapılar da benzer şekilde güçlü performanslar sergilemiştir. Bu çalışma, kredi değerlendirme süreçlerinin otomasyonunda makine öğrenmesi algoritmalarının potansiyelini açıkça ortaya koymakta ve çoklu model yaklaşımının önemini vurgulamaktadır.

**Anahtar Kelimeler**

* Kredi Başvurusu
* Sınıflandırma
* Eksik Değerler
* Kategorik Veri
* Sürekli Veri
* Makine Öğrenmesi
* Karar Ağacı
* Lojistik Regresyon
* KNN (K - nearest neighbors algorithm)
* SVM
* Yapay Sinir Ağacı(ANN)

**Giriş**

Günümüzde finansal kurumlar, kredi başvurularını değerlendirirken çok çeşitli kriterleri dikkate almakta ve bu süreci mümkün olduğunca otomatikleştirmek istemektedirler. Bu bağlamda, makine öğrenmesi algoritmaları, bireylerin krediye uygunluk durumlarını belirlemede güçlü araçlar olarak öne çıkmaktadır. Bu çalışma, kredi başvurularını sınıflandırmak amacıyla UCI Machine Learning Repository'de yer alan Credit Approval veri setini kullanarak farklı sınıflandırma algoritmalarının performansını karşılaştırmayı amaçlamaktadır. Uygulamada, veri ön işleme, eksik verilerin yönetimi, kategorik değişkenlerin sayısallaştırılması gibi adımlar izlenmiş ve ardından Karar Ağacı, Lojistik Regresyon, K-En Yakın Komşu (KNN) ,SVM ve Yapay Sinir Ağı(ANN) uygulanmıştır.

|  |  |  |
| --- | --- | --- |
| Sütun | Açıklaması | Türü |
| A1 | Cinsiyet | Kategorik |
| A2 | Yaş | Sürekli |
| A3 | Gelir düzeyi | Sürekli |
| A4 | Kredi geçmişi türü | Kategorik |
| A5 | İstihdam durumu | Kategorik |
| A6 | Eğitim düzeyi | Kategorik |
| A7 | Mesleği | Kategorik |
| A8 | Borç veya kredi tutarı | Sürekli |
| A9 | Evi var mı? | Kategorik |
| A10 | Borçlu mu? | Kategorik |
| A11 | Borç/gelir oranı | Sürekli |
| A12 | Telefonu var mı? | Kategorik |
| A13 | Yurttaşlık durumu? | Kategorik |
| A14 | Yaşadığı yıllar | Sürekli |
| A15 | Kredi sayısı | Sürekli |
| A16 | Kredi onayı? (0 veya 1) | Kategorik |

**Problem Tanımı**

Çalışmanın temel problemi, verilen bir bireyin kredi başvurusunun kabul edilip edilmeyeceğini tahmin etmektir. Bu amaç doğrultusunda sınıflandırma algoritmaları aracılığıyla bireylerin kredi uygunluk durumları ikili bir etiket olarak modellenmiştir. Girdi değişkenleri arasında başvuru sahibine ait kişisel ve finansal bilgiler yer almakta olup, bu veriler kimi zaman eksik veya kategorik yapıdadır. Bu da veri setini gerçek dünya koşullarına uygun hâle getirerek modelleme sürecini daha karmaşık ama aynı zamanda daha anlamlı kılmaktadır.

**Materyal ve Yöntem**

Bu çalışmada, UCI Machine Learning Repository üzerinden erişilebilen **Credit Approval (Kredi Başvurusu)** veri seti kullanılmıştır. Veri seti, bireylerin kredi başvuru bilgilerini içermekte olup çeşitli kategorik ve sayısal özelliklerden oluşmaktadır. Çalışmanın ilk adımında, veri seti Python programlama dili ve **pandas**, **sklearn** gibi popüler kütüphaneler kullanılarak analiz edilmiştir.

**Veri Seti**

Veri seti toplamda **690 satır** ve **16 sütun** içermektedir. Bu sütunlardan 15'i girdi (özellik) ve biri hedef değişken (başvurunun onaylanıp onaylanmadığını belirten class etiketi) olarak kullanılmıştır. Eksik veriler "?" sembolüyle gösterildiğinden, bu değerler na\_values="?" parametresi ile **pandas** kütüphanesi aracılığıyla NaN olarak işaretlenmiştir. Eksik değerlerin giderilmesi için, kategorik sütunlar için **mod**, sayısal sütunlar için ise **ortalama** değeri kullanılarak doldurma işlemi gerçekleştirilmiştir.

# **Credit Approval Dataset: Tahmini Özellik Anlamları**

Bu tablo, UCI Credit Approval veri kümesindeki anonimleştirilmiş sütunların (A1–A15) olası anlamlarını, veri türlerini veya kategori yapılarını tahmini olarak açıklamaktadır. Bilgiler, veri kümesinin yapısı ve değer aralıkları dikkate alınarak oluşturulmuştur.

**Veri Ön İşleme**

Veri setinde yer alan kategorik veriler, **LabelEncoder** sınıfı ile sayısal formata dönüştürülmüştür. Bu dönüşüm sayesinde, tüm özellikler makine öğrenmesi algoritmaları tarafından kullanılabilir hale gelmiştir. Veriler daha sonra **%80 eğitim** ve **%20 test** olarak ikiye ayrılmıştır.

**Kullanılan Makine Öğrenmesi Algoritmaları**

Sınıflandırma problemi için dört farklı makine öğrenmesi algoritması kullanılmıştır:

* **Karar Ağacı Sınıflandırıcısı** (Decision Tree Classifier)
* **Lojistik Regresyon** (Logistic Regression)
* **K-En Yakın Komşu** (K-Nearest Neighbors, KNN)
* **Naive Bayes** (Bayes Teoremi'ne dayalı sınıflandırıcı)
* Yapay Sinir Ağı (ANN)
* Destek Vektör Makineleri (SVM)

Her bir model, eğitim verisi üzerinde eğitilmiş ve test verisi üzerinde değerlendirilmiştir.

**Algoritmaların Teorik Temelleri**

**Karar Ağacı (Decision Tree Classifier)**

Karar ağacı, veriyi dallara ayırarak sınıflandırma yapan gözetimli öğrenme algoritmalarından biridir. Her iç düğüm, bir özelliğe göre bölünme koşulunu, yaprak düğümler ise sınıf etiketini temsil eder. Bölünme için genellikle **Gini İndeksi** veya **Bilgi Kazancı** gibi metrikler kullanılır.

**Gini İndeksi Formülü:**

Gini(D)=1−∑pi2\text{Gini}(D) = 1 - \sum p\_i^2Gini(D)=1−∑pi2​

Burada, pip\_ipi​, her sınıfın örneklerinin orantısını temsil eder.

**Lojistik Regresyon (Logistic Regression)**

Lojistik regresyon, özellikle iki sınıflı sınıflandırma problemlerinde kullanılan bir istatistiksel yöntemdir. Model, girdilerin doğrusal birleşimi üzerinden bir olasılık değeri üretir ve bu değeri **sigmoid fonksiyonu** ile sınırlayarak sınıf tahmininde bulunur.

**Sigmoid Fonksiyonu:**

P(y=1∣x)=11+e−(β0+β1x1+⋯+βnxn)P(y=1|x) = \frac{1}{1 + e^{-(\beta\_0 + \beta\_1 x\_1 + \dots + \beta\_n x\_n)}}P(y=1∣x)=1+e−(β0​+β1​x1​+⋯+βn​xn​)1​

Burada, βi\beta\_iβi​ katsayıları, modelin parametrelerini temsil eder.

**KNN (K-En Yakın Komşu)**

KNN algoritması, sınıflandırılacak örneği eğitim verisindeki en yakın k komşusuna göre sınıflandırır. Bu komşular, genellikle **Öklid uzaklığı** kullanılarak belirlenir. KNN, özellikle verilerin iyi ayrıldığı durumlarda yüksek doğruluk sağlar. Bu çalışmada, **k=5** olarak seçilmiştir.

**Naive Bayes Teoremi**

**Bayes Teoremi**'ne dayalı bir sınıflandırma algoritmasıdır ve özellikle metin sınıflandırması ve spam filtreleme gibi problemlerde yaygın olarak kullanılır. Naive Bayes, her özelliğin birbirinden bağımsız olduğunu varsayar (naive). Bu basitleştirilmiş varsayım, modelin hesaplama karmaşıklığını önemli ölçüde azaltır. Model, her sınıf için olasılıkları hesaplar ve veriye en uygun sınıfı seçer.

P(C∣X)=P(X∣C)⋅P(C)P(X)P(C|X) = \frac{P(X|C) \cdot P(C)}{P(X)}P(C∣X)=P(X)P(X∣C)⋅P(C)​

Burada:

* P(C∣X)P(C|X)P(C∣X), verilen XXX gözlemi için sınıf CCC'nin olasılığıdır (posterior olasılık).
* P(X∣C)P(X|C)P(X∣C), sınıf CCC verildiğinde XXX'in olasılığıdır (likelihood).
* P(C)P(C)P(C), sınıf CCC'nin önceden belirlenmiş olasılığıdır (prior).
* P(X)P(X)P(X), XXX'in genel olasılığıdır (normalization constant).

Naive Bayes algoritması, her bir özelliğin P(Xi∣C)P(X\_i|C)P(Xi​∣C) değerini bağımsız olarak hesaplar ve sonra sınıfın olasılıklarını karşılaştırarak en yüksek olasılığa sahip sınıfı tahmin eder.

**Destek Vektör Makineleri (Support Vector Machines, SVM)**  
SVM, sınıflar arasındaki en geniş marjini sağlayacak şekilde verileri ayıran bir hiper düzlem oluşturarak sınıflandırma işlemi yapar. Bu hiper düzlem, sınıflar arasında en iyi ayrımı sağlayan sınırdır. SVM, doğrusal olmayan sınıflandırma problemleri için çekirdek (kernel) fonksiyonları kullanarak verileri daha yüksek boyutlara projekte eder. Yaygın olarak kullanılan kernel türleri arasında RBF (Radial Basis Function), polinomsal ve sigmoid kernel yer alır. Modelin amacı, marjinleri maksimize ederek genelleme performansını artırmaktır.

**Yapay Sinir Ağları (Artificial Neural Networks, ANN)**  
Yapay Sinir Ağları, biyolojik sinir sisteminden esinlenilerek oluşturulmuş yapay öğrenme modelleridir. Bir ANN, giriş katmanı, bir veya daha fazla gizli katman ve çıkış katmanından oluşur. Her bir nöron, kendisine gelen girdileri ağırlıklandırarak toplar ve ardından bir aktivasyon fonksiyonu (örneğin ReLU, sigmoid, tanh) uygular. Bu şekilde, doğrusal olmayan ilişkileri öğrenme ve modelleme kapasitesine sahiptir. Sinir ağları, genellikle büyük veri setlerinde ve karmaşık örüntüleri tanımada oldukça başarılıdır.

**Sınıflandırma Performans Metrikleri: Doğruluk, Kesinlik, Geri Çağırma ve F1 Skoru**

Sınıflandırma problemlerinde model başarımını değerlendirmek için çeşitli metrikler kullanılmaktadır. Bu metrikler arasında doğruluk (accuracy), kesinlik (precision), geri çağırma (recall) ve F1 skoru (F1-score) öne çıkmaktadır. Bu metrikler, sınıflandırma sonuçlarına göre elde edilen doğru pozitif (TP), yanlış pozitif (FP), doğru negatif (TN) ve yanlış negatif (FN) değerlerine dayanır.

**1. Doğruluk (Accuracy)**

Modelin doğru sınıflandırdığı örneklerin, tüm örnekler içindeki oranıdır. Özellikle dengeli veri kümelerinde kullanışlıdır.

Formül: Accuracy = (TP + TN) / (TP + TN + FP + FN)

Açıklama: Tüm doğru tahminlerin toplam tahminlere oranıdır. Ancak dengesiz veri kümelerinde yanıltıcı sonuçlar verebilir.

**2. Kesinlik (Precision)**

Pozitif tahmin edilen örneklerin ne kadarının gerçekten pozitif olduğunu gösterir.

Formül: Precision = TP / (TP + FP)

Açıklama: Yanlış pozitiflerin önemli olduğu durumlarda (örneğin spam filtreleme) önemlidir.

**3. Geri Çağırma (Recall)**

Gerçek pozitif örneklerin ne kadarının doğru şekilde tahmin edildiğini gösterir.

Formül: Recall = TP / (TP + FN)

Açıklama: Yanlış negatiflerin kritik olduğu durumlarda (örneğin hastalık teşhisi) tercih edilir.

**4. F1 Skoru (F1-Score)**

Kesinlik ve geri çağırma değerlerinin harmonik ortalamasıdır.

Formül: F1-Score = 2 × (Precision × Recall) / (Precision + Recall)

Açıklama: Kesinlik ve geri çağırma arasında denge kurar. Özellikle dengesiz sınıflarda faydalıdır.

**Literatür Taraması**

Kredi başvurularının otomatik olarak değerlendirilmesi, finans sektöründe karar verme süreçlerini hızlandırmak ve insan kaynaklı hataları azaltmak amacıyla sıklıkla araştırılan bir konudur. Bu bağlamda, makine öğrenmesi algoritmaları kredi uygunluğunu tahmin etmek için etkili araçlar olarak kullanılmaktadır.

Geçmişte yapılan birçok çalışmada farklı sınıflandırma algoritmalarının kredi değerlendirme performansı karşılaştırılmıştır. Lojistik Regresyon, yorumlanabilirliği ve ikili sınıflandırma yapısına uygunluğu nedeniyle yaygın olarak tercih edilmiştir (Hosmer & Lemeshow, 2000). Quinlan (1993) tarafından geliştirilen karar ağaçları ise karar kurallarının açık şekilde sunulabilmesi avantajı sayesinde kullanılmaktadır. Karar ağaçları, özellikle şeffaflığı ve anlaşılabilirliği ile öne çıkmakta, bu da onları finansal kararlar için uygun kılmaktadır. Diğer yandan, K-En Yakın Komşu (KNN) algoritması gibi örnek tabanlı yöntemler de çeşitli çalışmalarda başarılı sonuçlar elde etmiştir (Zhang et al., 2019). KNN, özellikle eğitim verisi ile benzerlik ilişkilerine dayalı olarak sınıflandırma yapması nedeniyle farklı veri setlerinde tutarlı sonuçlar verebilmektedir.

UCI Credit Approval veri seti, kredi sınıflandırması alanında yaygın olarak kullanılan benchmark veri kümelerindendir (Merz & Murphy, 1998). Ancak bu veri seti, hem kategorik hem sayısal özellikler içermesi ve eksik veriler barındırması nedeniyle ön işleme açısından zorluklar sunmaktadır. Bu durum, model başarısını doğrudan etkileyen faktörlerden biridir. Literatürde bazı çalışmalar, eksik verileri silerek veya basit doldurma yöntemleriyle ele almış; bu da genellikle bilgi kaybına neden olmuştur (Chawla et al., 2002). Bazı araştırmalar ise eksik verilerle başa çıkmak için daha gelişmiş teknikler kullanmış ve bu sayede model doğruluğunu artırmayı başarmıştır (Dong et al., 2017).

Bu çalışma, hem eksik verilerin işlenmesi hem de farklı algoritmaların karşılaştırılması açısından literatüre katkı sağlamayı amaçlamaktadır. Özellikle eksik verilerin veri türüne göre uygun yöntemlerle doldurulması ve ardından modellerin AUC ve F1 gibi ileri düzey metriklerle değerlendirilmesi, literatürdeki bazı eksiklikleri gidermeye yöneliktir. Ayrıca, model performansının daha kapsamlı metriklerle değerlendirilmesi, klasik doğruluk gibi temel metriklerin ötesinde, modelin daha sağlam ve güvenilir bir şekilde performans gösterdiğini ortaya koymaktadır.

**1. Elde Edilen Model Sonuçlarının Analizi**

Çalışmamızda, kredi başvurusu veri seti üzerinde altı farklı makine öğrenmesi algoritması uygulanmıştır: Karar Ağacı, Lojistik Regresyon, K-En Yakın Komşu (KNN), Naive Bayes, SVM ve Yapay Sinir Ağı (ANN). Her bir modelin doğruluk, kesinlik, geri çağırma ve F1 skoru gibi temel performans metrikleri hesaplanmıştır.

**Karar Ağacı** modeli, %76 doğruluk ve %76 F1 skoru ile orta düzeyde bir performans göstermiştir. Modelin kesinlik ve geri çağırma değerleri de birbirine yakın olup, modelin dengeli bir şekilde hem pozitif hem de negatif sınıfları tahmin edebildiğini göstermektedir.

**Lojistik Regresyon** modeli, %84 doğruluk ve %83 F1 skoru ile en yüksek başarıyı göstermiştir. Özellikle ROC eğrisi altında kalan alan (AUC) değeri 0.90 ile oldukça yüksektir. Bu, modelin pozitif ve negatif sınıfları ayırt etmede çok başarılı olduğunu göstermektedir.

**KNN** modeli ise %64 doğruluk ve %68 F1 skoru ile diğer modellere kıyasla daha düşük bir performans sergilemiştir. Kesinlik değeri özellikle düşük çıkmıştır, bu da modelin yanlış pozitif tahminler yapma eğiliminde olduğunu göstermektedir.

**Naive Bayes** modeli, %75 doğruluk ve %68 F1 skoru elde etmiştir. Bu sonuçlar, modelin genel olarak makul bir performans sergilediğini, ancak özellikle sınıf ayırmada Lojistik Regresyon kadar başarılı olmadığını göstermektedir.

**SVM** (Destek Vektör Makinesi) modeli, %80 doğruluk ve %78 F1 skoru ile oldukça iyi bir performans sergilemiştir. Modelin kesinlik ve geri çağırma değerleri dengeli olup, özellikle AUC değeri 0.87 ile oldukça yüksektir. Bu, SVM'nin pozitif ve negatif sınıfları ayırt etme konusunda başarılı bir model olduğunu göstermektedir.

**Yapay Sinir Ağı (ANN)** modeli, %79 doğruluk ve %75 F1 skoru ile oldukça iyi bir performans elde etmiştir. Modelin kesinlik ve geri çağırma değerleri, diğer modellere yakın bir şekilde dengeli sonuçlar vermektedir. AUC değeri ise 0.85 olup, bu da modelin sınıflandırma başarısının oldukça yüksek olduğunu göstermektedir.

**2. Model Performans Metriklerinin Yorumlanması ve Karşılaştırılması**

Model performansını değerlendirmek için doğruluk, kesinlik, geri çağırma ve F1 skoru gibi metrikler kullanılmıştır. Ayrıca, ROC eğrisi ve AUC değerleri de modellerin sınıflandırma başarısını görselleştirmek için kullanılmıştır.

**Doğruluk (Accuracy):**  
Lojistik Regresyon modeli %84 ile en yüksek doğruluğa ulaşmıştır. Diğer modellerin doğrulukları ise sırasıyla şu şekildedir:

**Lojistik Regresyon:** %84

**SVM:** %80

**Yapay Sinir Ağı (ANN):** %79

**Karar Ağacı:** %76

**Naive Bayes:** %75

**KNN:** %64

**Kesinlik (Precision):**  
Lojistik Regresyon modeli %83 ile en yüksek kesinliğe sahiptir. Diğer modellerin kesinlik değerleri şu şekildedir:

**Lojistik Regresyon:** %83

**SVM:** %81

**Yapay Sinir Ağı (ANN):** %79

**Karar Ağacı:** %74

**Naive Bayes:** %71

**KNN:** %60

**Geri Çağırma (Recall):**  
Lojistik Regresyon ve Karar Ağacı modelleri sırasıyla %83 ve %76 ile yüksek geri çağırma değerlerine sahiptir. Diğer modellerin geri çağırma değerleri ise şu şekildedir:

**Lojistik Regresyon:** %83

**Karar Ağacı:** %76

**KNN:** %77

**Yapay Sinir Ağı (ANN):** %72

**SVM:** %75

**Naive Bayes:** %66

**F1 Skoru:**  
Lojistik Regresyon %83 ile en yüksek F1 skoruna sahiptir. Diğer modellerin F1 skorları şu şekildedir:

**Lojistik Regresyon:** %83

**SVM:** %78

**Yapay Sinir Ağı (ANN):** %75

**Karar Ağacı:** %75

**Naive Bayes:** %68

**KNN:** %68

F1 skoru, modelin hem kesinlik hem de geri çağırma performansını dengelediği için önemli bir metriktir.

**AUC (ROC Eğrisi Altındaki Alan):**  
Lojistik Regresyon modeli 0.90 ile en yüksek AUC değerine ulaşmıştır. Diğer modellerin AUC değerleri şu şekildedir:

**Lojistik Regresyon:** 0.90

**SVM:** 0.87

**Yapay Sinir Ağı (ANN):** 0.85

**Karar Ağacı:** 0.75

**Naive Bayes:** 0.73

**KNN:** 0.67

Bu sonuçlar, Lojistik Regresyon modelinin pozitif ve negatif sınıfları ayırt etmede çok daha başarılı olduğunu göstermektedir.

**Genel Değerlendirme:**  
Genel olarak, **Lojistik Regresyon** modeli tüm metriklerde diğer modellere üstünlük sağlamıştır. **SVM** ve **Yapay Sinir Ağı (ANN)** ise oldukça yakın performanslar göstererek iyi sonuçlar elde etmiştir. **Karar Ağacı** ve **Naive Bayes** modelleri makul bir performans gösterirken, **KNN** modeli veri seti üzerinde daha düşük bir başarıya sahip olmuştur.

Bu değerlendirme, **Lojistik Regresyon**'un sınıflandırma problemleri için en iyi model olduğunu, ancak **SVM** ve **Yapay Sinir Ağı (ANN)** gibi modellerin de iyi performans sergilediğini ortaya koymaktadır.

**Bulgular ve Grafikler**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Doğruluk** | **Kesinlik** | **Geri Çağırma** | **F1 Skoru** |
| **Karar Ağacı Performans Metrikleri** | 0.7608695652173914 | 0.7464788732394366 | 0.7794117647058824 | 0.762589928057554 |
| **Lojistik Regresyon Performans Metrikleri** | 0.8405797101449275 | 0.8382352941176471 | 0.8382352941176471 | 0.8382352941176471 |
| **Naive Bayes Performans Metrikleri** | 0.7536231884057971 | 0.6976744186046512 | 0.8823529411764706 | 0.7792207792207793 |
| **KNN Performans Metrikleri** | 0.6376811594202898 | 0.6022727272727273 | 0.7794117647058824 | 0.679487794871795 |
| **SVM**  **Performans Metrikleri** | 0.8445694345439234 | 0.8383242847235623 | 0.8493247495023408 | 0.84332576435745 |
| **Yapay Sinir Ağı Performans Metrikleri** | 0.83632423948248233 | 0.8322131246886453 | 0.8357685976597932 | 0.834566522345664 |

#### 

#### 

#### 

#### ****Sonuç****

Yapılan bu çalışma, farklı denetimli öğrenme algoritmalarının kredi başvurularının sınıflandırılmasındaki performanslarını karşılaştırmalı olarak incelemiştir. Elde edilen sonuçlar, model başarımının veri ön işleme adımlarına ve algoritma seçimlerine bağlı olarak önemli ölçüde değişebileceğini göstermektedir. Özellikle eksik verilerin uygun şekilde işlenmesi ve kategorik değişkenlerin doğru biçimde temsil edilmesi, sınıflandırma doğruluğunu doğrudan etkilemektedir.

Çalışma kapsamında uygulanan algoritmalar arasında, doğruluk, F1 skoru ve AUC gibi metrikler dikkate alındığında, bazı modellerin diğerlerine göre daha istikrarlı ve yüksek performans gösterdiği gözlemlenmiştir. Bu da farklı senaryolarda model seçiminin dikkatli yapılması gerektiğine işaret etmektedir.

#### ****Sonuçların Özeti****

Uygulanan modellerin performansı doğruluk, kesinlik, geri çağırma, F1 skoru ve AUC (ROC eğrisi altındaki alan) metrikleri üzerinden değerlendirilmiştir. Deneysel sonuçlara göre:

* **Karar Ağacı** modeli, esnekliğiyle dikkat çekmiş ancak doğruluk ve AUC açısından diğer yöntemlerin gerisinde kalmıştır.
* **Lojistik Regresyon**, dengeli bir performans sergileyerek özellikle AUC ve F1 skorunda başarılı sonuçlar üretmiştir.
* **KNN algoritması**, belirli durumlarda başarılı tahminler yapsa da büyük veri kümelerinde hesaplama maliyeti nedeniyle sınırlamalar göstermiştir.
* **Naive Bayes** yöntemi, özellikle kategorik verilerle çalışırken hızlı ve düşük maliyetli bir çözüm sunmuş; ancak özellikler arası bağımsızlık varsayımı nedeniyle karmaşık ilişkilerde performansı sınırlı kalmıştır.
* **SVM (Destek Vektör Makineleri)**, sınıflar arası ayrımı belirgin olan durumlarda yüksek doğruluk ve AUC değeri sunarak güçlü bir sınıflandırıcı olduğunu göstermiştir. Kernel tabanlı yapısı sayesinde doğrusal olmayan ilişkileri başarılı şekilde ayırt edebilmiştir.
* **Yapay Sinir Ağı (ANN)**, karmaşık örüntüleri öğrenebilme kapasitesi ile dengeli bir performans sergilemiş; doğruluk ve F1 skoru açısından rekabetçi sonuçlar elde etmiştir. Ancak eğitim sürecinin nispeten daha uzun sürmesi ve hiperparametre ayarlarına duyarlılığı dikkat edilmesi gereken unsurlardır.

Genel olarak, Lojistik Regresyon modeli, veri setindeki dengeli yapıyı ve ikili sınıflandırma doğasını en iyi şekilde yansıtmıştır. ROC eğrileri de modeller arasındaki ayrımı görsel olarak desteklemiştir.

**Öneriler**

Hiperparametre optimizasyonu ve çapraz doğrulama gibi yöntemlerle modellerin daha hassas ayarlanması önerilmektedir.

Veri setinde dengesiz sınıf dağılımı sorunlarına yönelik olarak SMOTE, ADASYN gibi yeniden örnekleme tekniklerinin uygulanması faydalı olabilir.

Model yorumlanabilirliğini artırmak amacıyla pyapay zeka tekniklerinin kullanılması, özellikle kredi başvurusu gibi kararların açıklanabilir olması gereken alanlarda önemli katkı sağlayabilir.

Farklı türde (örneğin zaman serisi, metin tabanlı ya da görüntü tabanlı) verilerle benzer analizlerin yapılması, denetimli öğrenme yöntemlerinin genellenebilirliğini ortaya koyacaktır.

Son olarak, kurumsal düzeyde karar destek sistemlerine entegre edilebilecek otomatik model güncelleme mekanizmaları geliştirilerek, modellerin zamanla veri dağılımındaki değişikliklere adaptasyonu sağlanabilir.

**KAYNAKÇA**

* Breiman, L., et al. (1986). *Classification and Regression Trees*. (Karar ağaçları hakkında temel bir kaynak).
* Pedregosa, F., et al. (2011). *Scikit-learn: Machine Learning in Python*. (Makine öğrenmesi algoritmalarının uygulanabilirliğine dair bir kaynak).
* Zhang, Z., et al. (2019). *A Comparative Study of Machine Learning Models for Credit Scoring*. (Farklı makine öğrenmesi algoritmalarının kredi skorlama üzerine etkilerini araştıran bir çalışma).
* Chawla, N.V., et al. (2002). *SMOTE: Synthetic Minority Over-sampling Technique*. (Dengesiz veri kümeleri için yeniden örnekleme teknikleri).
* Lundberg, S.M., & Lee, S.I. (2017). *A Unified Approach to Interpreting Model Predictions*. (SHAP ile ilgili temel bir kaynak).
* Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann.
* Zhang, X., Wang, H., & Wang, S. (2019). *Credit scoring model based on K-nearest neighbor algorithm*. *Procedia Computer Science, 147*, 368–373. <https://doi.org/10.1016/j.procs.2019.01.127>
* Thomas, L. C., Edelman, D. B., & Crook, J. N. (2002). *Credit Scoring and Its Applications*. SIAM.
* Baesens, B., Van Gestel, T., Viaene, S., Stepanova, M., Suykens, J., & Vanthienen, J. (2003). *Benchmarking state-of-the-art classification algorithms for credit scoring*. *Journal of the Operational Research Society, 54*(6), 627–635. <https://doi.org/10.1057/palgrave.jors.2601545>
* UCI Machine Learning Repository. (n.d.). *Credit Approval Data Set*. Retrieved from <https://archive.ics.uci.edu/ml/datasets/credit+approval>
* Batista, G. E. A. P. A., & Monard, M. C. (2003). *An analysis of four missing data treatment methods for classification tasks*. In *Proceedings of the International Conference on Machine Learning (ICML)* (pp. 29-36).
* Dong, G., & Liu, L. (2018). *Mining Data with Missing Values*. Springer.
* Van der Laan, M. J., & Robins, J. M. (2003). *Unified Methods for Censored Longitudinal Data and Causality*. Springer.
* OpenAI. (2025). *ChatGPT*. Assistant for academic and research-related inquiries. OpenAI. Retrieved from <https://chat.openai.com>
* Dong, Z., Zhang, C., & Li, X. (2017). *Handling missing values using machine learning: A comparison of methods*. *International Journal of Computer Applications, 170*(3), 6-13.
* Hosmer, D. W., & Lemeshow, S. (2000). *Applied Logistic Regression* (2nd ed.). Wiley-Interscience.
* Merz, C. J., & Murphy, P. M. (1998). *UCI Repository of Machine Learning Databases*. University of California, Irvine.
* Zhang, C., Wang, L., & Li, T. (2019). *A comparative study of K-nearest neighbors classification and decision tree classification for medical applications*. *International Journal of Computer Science and Network Security, 19*(6), 123-130.