VERİ MADENCİLİĞİ (FET445)

GÜZ DÖNEMİ

VİZE PROJESİ

MÜHENDİSLİK FAKÜLTESİ

BİLGİSAYAR MÜHENDİSLİĞİ

Grup Adı: Trinity

ELİF YALINKAYA 22040101031

MELİSA SELEME ARSLANTAŞ 22040101032

EZGİ YILDIRIM 22040101048

GitHub/Repo link

https://github.com/ezgy22/Veri-Madenciligi-TMDB-Proje

1-) Problem Tanımı

Yeni çıkan dizilerin popüler olup olmayacağını tahmin etmek amaçlanmaktadır. Tür, yapım yılı, ülke, sezon/bölüm sayısı, oy ortalaması gibi özellikler kullanılarak dizinin popülerlik skorunu öngören bir makine öğrenmesi modeli geliştirilecektir.
Bu kapsamda proje şu soruya yanıt arar:
"Bir dizinin popülerliğini en çok hangi faktörler belirler?"

Görev Türü

Bu çalışma iki aşamalı olarak tasarlanmıştır. İlk aşamada popülerlik skoru regresyon ile tahmin edilmiş, ikinci aşamada bu çıktı karar destek amacıyla sınıflandırma problemine dönüştürülmüştür.

- **Regresyon (Sürekli Tahmin- Birinci Aşama):** İlk olarak, dizilerin popülerlik skorlarını doğrudan tahmin etmek hedeflenmiştir. Bu, dizinin kitle üzerindeki etkisinin sayısal büyüklüğünü ölçmek için yapılmıştır. Vize aşamasındaki düşük R^2 değerleri, final aşamasında Tuned XGBoost ve Random Forest gibi gelişmiş ensamble modelleriyle iyileştirilmiştir.

- **Sınıflandırma (Karar Destek - İkinci Aşama):** Regresyon sonuçlarını daha anlamlı ve iş kararlarına uygun hale getirmek için popülerlik verisi medyan değerine göre iki sınıfa (0: Popüler Değil, 1: Popüler) ayrılmıştır. Bu yaklaşım, "Bu dizi izlenmeye/ yatırım yapmaya değer mi?" sorusuna net bir sınıflandırma yanıtı üretmek amacıyla geliştirilmişltir.

- **Hedef Değişkenler:** Popülerlik Skoru (popularity), Popüler / Popüler değil (0/1)

•**Başarı Kriterleri:** Projenin başarısı, vize aşamasında belirlenen temel değerlerin üzerine çıkılması ve final isterlerinde belirtilen metriklerin optimize edilmesiyle ölçülmüştür:

Regresyon için

- Daha düşük **RMSE**, daha yüksek **R^2** değeri.

- Baseline model olan *Linear Regression*'ın elde ettiği **R^2 = 0.1968** değerinin anlamlı şekilde üzerine çıkılması.

- Sadece RMSE ve R^2 ile yetinilmeyip; modelin hata payını daha iyi analiz edebilmek adına MAE, MSE ve MAPE değerleri de hesaplanmıştır.

- Beklenti: Tuned XGBoost ve Random Forest gibi gelişmiş modellerle R^2 skorunda vizeye oranla belirgin bir artış sağlanması hedeflenmiştir.

Sınıflandırma İçin

- Yüksek **ROC AUC** skoru ve dengeli **precision/recall** değerleri.

- Baseline model olan *Lojistik Regresyon*'un elde ettiği **ROC AUC = 0.8871** performansının yakalanması veya geçilmesi.

- Modelin dengeli performansını ölçmek için Accuracy, Precision, Recall ve F1-Score metrikleri bir arada değerlendirilmiştir.

- PyTorch ile geliştirilen mimarilerin klasik modellerle rekabet edebilir seviyeye (Min. %80 Accuracy) getirilmesi amaçlanmıştır.

2-) Proje Yönetimi

2.1 Kilometre Taşları ve Zaman Çizelgesi

Aşağıda, proje süresince tamamlanan aşamalar ve planlanan sonraki adımlar haftalık bir zaman çizelgesi şeklinde sunulmuştur:

1. Hafta: Veri setinin seçilmesi ve proje konusunun belirlenmesi
2. Hafta: Veri ön işleme adımlarının uygulanması (One-Hot Encoding, Standardizasyon) ve veri setinin eğitim-test olarak ayrılması.

3. Hafta: Temel modellerin geliştirilmesi ve öznitelik seçimi çalışmaları.

Modelleme:

- Linear Regression ve KNeighbors Regressor (Elif)
- Gaussian Naive Bayes ve Lojistik Regresyon (Melisa)

- *Öznitelik Seçimi:*
  - PCA, RFE, SelectKBest ve Variance Threshold uygulamaları

4 –5. Haftalar : Gelişmiş modellerin oluşturulması ve hiperparametre optimizasyonu.

- Karar Ağacı ve Linear SVR modellerinin geliştirilmesi (Ezgi)
- Model performans analizi ve yorumlanabilirlik çalışmalarının yapılması

6. Hafta : Nihai performans değerlendirmesi, raporun tamamlanması ve sunumun hazırlanması.

7. Hafta:

- **Gelişmiş Regresyon ve sınıflandırma stratejileri:** Vize modellerinde düşük açıklayıcılık oranlarını (R^2 = 0.1966) aşmak amacıyla XGBoost, LightGBM ve Random Forest gibi topluluk (ensemble) modelleri kurulmuştur.

8. Hafta:

- **Derin Öğrenme ve PyTorch Entegrasyonu**: Her grup üyesi bireysel olarak en az ikişer adet PyTorch tabanlı mimari geliştirmiş ve tablo verisi üzerinde derin öğrenme performansı test edilmiştir.

9. Hafta:

- **Hiperparametre Optimizasyonu (Tuning):** Modellerin genelleme yeteneğini artırmak için ekip üyeleri tarafından kapsamlı hiperparametre optimizasyonları yürütülmüştür. Elif, XGBoost ve LightGBM modellerinde Randomized SearchCV kullanarak learning_rate ve num_leaves gibi kritik parametreleri optimize ederken, Melisa Gradient Boosting ve

PyTorch ANN yapılarında katman nöron sayıları ile Adam Optimizer ayarları üzerinde tuning işlemleri gerçekleştirmiştir. Ezgi ise Random Forest ve Extra Trees modellerinde ağaç sayılarını optimize ederek en yüksek regresyon başarısına (%71.55 R^2) ulaşmış, PyTorch Wide & Deep mimarisinde ise ezber ve genelleme yolları arasındaki dengeyi hassaslaştırmıştır.

10. Hafta:

- **Nihai Karşılaştırma ve Hata Analizi:** Tüm modeller RMSE, R^2, Accuracy ve F1-Score metrikleri üzerinden kıyaslanarak "Şampiyon" modeller belirlenmiştir.

2.2 Roller ve Sorumluluklar

Proje ekibinde her bir üye, bireysel katkıları belirginleştirmek amacıyla aşağıdaki görevleri üstlenmiştir:

Grup Üyesi	Vize Sorumlulukları	Final Gelişmiş Modelleri & Hiperparametre Tuning	Derin Öğrenme (PyTorch) Uygulamaları
Elif	RFE ve SelectKBest öznitelik seçimi; Linear Regression ve KNN modelleri.	**Tuned XGBoost** ve **Tuned LightGBM** (Randomized SearchCV ile optimize edildi).	**Optimized DNN** (128-64-32 nöron) ve **Wide & Deep** regresyon mimarileri
Melisa	Varyans Eşiği yöntemi; Naive Bayes ve Lojistik Regresyon modelleri	**Gradient Boosting** ve **Stacking Ensemble** (GB, RF ve XGBoost birleşimi).	**Optimized ANN** (128-64 nöron) ve **ResNet-MLP** (Residual bloklar ve Dropout).
Ezgi	PCA boyut indirgeme; Karar Ağacı ve Linear SVR modelleri.	**Random Forest** (100 estimator) ve **Extra Trees** topluluk modelleri.	**Wide & Deep** (Memorization/Generalization) ve **Simple MLP** (Baseline ANN).

- **GitHub/Repo Linki:** https://github.com/ezgy22/Veri-Madenciligi-TMDB-Proje

2.3 Proje İş Akışı ve Metodolojik Sıralama

Projemiz, veriden hem sayısal öngörü hem de stratejik karar destek mekanizması üretmek amacıyla ardışık iki ana fazda yürütülmüştür:

1. **Faz: Regresyon Analizi (Sayısal Öngörü)** Projenin başlangıç noktası, dizilerin popülerlik skorlarını doğrudan tahmin etmektir.
   - Ezgi, regresyon başarısını artırmak için Random Forest (%71.55 R^2) ve Extra Trees modellerini geliştirmiş; ardından PyTorch ile Wide & Deep mimarisini kurarak sayısal tahminleri derin öğrenme seviyesine taşımıştır.

- Elif, vize aşamasında Linear Regression ve KNN modelleriyle temel (baseline) performans değerlerini belirlemiş, öznitelik seçimiyle (RFE ve SelectKBest) bu modelleri optimize etmeye çalışmıştır. Regresyon performansını nihai noktaya ulaştırmak için XGBoost (%73,7 R^2) ve LightGBM modellerinde hiperparametre optimizasyonu yapmıştır.

2. **Faz: Sınıflandırma Analizi (Stratejik Karar ve Üst Katman):** Regresyon çıktılarının sektörel olarak daha anlamlı yorumlanabilmesi için veriler medyan değerine göre "Popüler / Popüler Değil" olarak etiketlenerek ikinci faza geçilmiştir.

- Melisa, Vize aşamasında Naive Bayes ve Lojistik Regresyon ile temelleri atmış; finalde ise Gradient Boosting ve projenin sınıflandırma şampiyonu olan Stacking (%82.44 Accuracy) modellerini kurmuştur. Ayrıca PyTorch (ANN, ResNet-MLP) mimarileriyle projenin ayırt ediciliğini (0.90 AUC) kanıtlamıştır.

3-) İlgili Çalışmalar (Mini Literatür İncelemesi)

• Kapsam ve Yöntem Karşılaştırması

Dizi/film popülerliği tahmini üzerine yapılan çalışmaların büyük bölümü tek bir model ailesine (örneğin yalnızca Ensemble yöntemler) veya tek bir görev türüne (yalnızca regresyon) odaklanmaktadır. Bu proje ise aynı veri seti üzerinde hem regresyon hem sınıflandırma görevlerini, klasik modellerden derin öğrenmeye uzanan geniş bir yelpazede ele almasıyla literatürde ayrışmaktadır.

Aşağıdaki tablo, projenin tipik çalışmalara göre metodolojik karşılaştırmasını özetlemektedir:

Özellik	**Tipik Çalışmalar**	**Bu Proje**
Kapsam	Çoğunlukla tek bir regresyon veya sınıflandırma görevi ele alınır.	Hem regresyon (popülerlik skoru tahmini) hem de sınıflandırma (popüler–popüler değil) görevleri eş zamanlı incelenmiştir.
Model Çeşitliliği	Genellikle 2–3 yüksek performanslı model kullanılır.	Onun üzerinde model ve altı farklı model ailesi değerlendirilmiştir: Lineer (Logistic/Linear Reg.), Ağaç (Random Forest/Extra Trees), Boosting (XGBoost/LightGBM), Olasılıksal (Naive Bayes), Derin Öğrenme (PyTorch ANN/Wide & Deep) ve Hibrit (Stacking)
Öznitelik Seçimi	Tek bir yöntem (çoğunlukla PCA) tercih edilir.	Hibrit Yaklaşım: Variance Threshold ile gürültü azaltılmış, Logaritmik Dönüşüm ile veri normalizasyonu sağlanmış ve RFE/SelectKBest ile model bazlı en kritik öznitelikler (Örn: vote_count) saptanmıştır.
Teknik Derinlik	Standart kütüphane modelleri kullanılır.	Özelleştirilmiş Mimari: PyTorch ile 338 özniteliği işleyebilen, 128-64 nöronlu çok katmanlı sinir ağları (ANN) ve hibrit Wide & Deep yapıları projeye özel olarak

		kurgulanmıştır.
Değerlendirme Ölçütleri	Göreve uygun temel metrikler (R^2 veya ROC AUC) kullanılır.	R^2, RMSE,MAE,MSE (Regresyon için); ROC AUC, Accuracy, Precision, Recall (Sınıflandırma için) gibi çoklu metriklerle modellerin kararlılığı ve tutarlılığı test edilmiştir.

Literatürle Metodolojik Bağlantılar

- **Akula et al. (2019):** Sitcomlar üzerine yaptığı çalışmada KNN ve Karar Ağaçları ile %17-39 bandında açıklayıcılık R^2 elde etmiştir. Projemizde Tuned XGBoost ile bu oran $R^2 = 0.737$ seviyesine çıkarılarak literatürün üzerine çıkılmıştır.
- **Cammarano et al. (2024):** Türden bağımsız popülerlik tahmininde Random Forest ve LSTM'in başarısını vurgulamıştır. Projemizdeki Stacking Ensemble (%82.44 Accuracy) başarısı, bu tür topluluk öğrenmesi yaklaşımlarının gücünü teyit etmektedir.
- **Stanford Projesi (2015):** IMDb verileriyle regresyon sonuçlarını kategorilere ayırarak sınıflandırma yapmıştır. Projemizdeki medyan eşik değerine göre ikili sınıflandırma (0/1) stratejisi bu akademik iş akışıyla paralellik gösterir.

Projenin Doldurduğu Boşluklar ve Sunulan Katkılar:

1. Aşamalı ve Kıyaslamalı Analiz Yaklaşımı

Proje, tek bir modele odaklanmak yerine; klasik istatistiksel modellerden derin öğrenmeye uzanan geniş bir yelpazede sistematik bir kıyaslama sunmuştur.

Gelişim Örneği: Vize aşamasında Linear Regression ile elde edilen 0.1966 R^2 skoru, final aşamasında veri ön işleme ve model optimizasyonu (Tuned XGBoost) ile 0.7370 seviyesine çıkarılmıştır.

Bu devasa artış, sadece algoritma değişikliğinin değil; logaritmik dönüşüm, aykırı değer yönetimi ve doğru hiperparametre ince ayarının veri madenciliği projelerindeki kritik önemini kanıtlamıştır.

2. Çoklu Boyut İndirgeme Stratejisi

Literatürdeki standart yaklaşımların ötesine geçilerek, verideki "gürültü" ve "bilgi" dengesi çok katmanlı bir stratejiyle yönetilmiştir:

- **Variance Threshold & Log Transformation:** Düşük varyanslı değişkenler elenirken, popülerlik verisindeki sağa çarpıklık logaritmik dönüşümle normalize edilmiştir.
- **Mekanizmaların Birleşimi:** 338 öznitelikli karmaşık yapı; hem Stacking Ensemble ile modeller arası bilgi aktarımı sağlayacak şekilde hem de ANN mimarisinde yüksek boyutlu veriyi işleyebilecek kapasitede kurgulanmıştır. Bu strateji, modelin ezberlemesini (overfitting) engelleyerek genelleme yeteneğini maksimize etmiştir.

3. Yorumlanabilirlik Odaklı Modelleme

Proje, popülerlik tahminini bir “kara kutu” olmaktan çıkararak sektörel kararlara ışık tutacak bir Karar Destek Sistemi kimliği kazanmıştır.

- **Gelişmiş Öznitelik Analizi:** Random Forest ve XGBoost üzerinden yapılan Feature Importance (Öznitelik Önem Düzeyi) analizleri, popülerliği etkileyen ana motorun sadece "bölüm sayısı" (nicelik) değil, asıl olarak "vote_count" (izleyici etkileşimi) ve içerik kalitesi olduğunu ispatlamıştır.
- **Mükemmel Ayırt Edicilik:** ANN ve Stacking modelleriyle ulaşılan 0.90 AUC skoru, sistemin başarılı yapımları başarısızlardan ayırmada mükemmel (excellent) bir performans sergilediğini ve yapımcılar için düşük riskli yatırım stratejileri sunabileceğini kanıtlamıştır.

4-) Veri Tanımı ve Yönetimi

1. Veri Seti ve Boyut

- **Veri Seti Adı:** Full TMDb TV Shows Dataset 2024 (TMDB tabanlı).
- **Kaynak / Lisans:** TMDB verileri kamuya açık olup, veri seti açık kaynak niteliğindedir. (https://www.kaggle.com/datasets/asaniczka/full-tmdb-tv-shows-dataset-2023-150k-shows)
- **Veri Boyutu:** Modelleme sürecinde veri setindeki öznitelik (feature) sayısı, kategorik verilerin işlenme derinliğine ve seçilen algoritmaların ihtiyacına göre iki temel aşamada şekillenmiştir:
    - Ham Veri Aşaması: Veri seti başlangıçta 29 temel öznitelikten oluşmaktadır.
    - Vize Aşaması: Kategorik değişkenlerin ilk seviye One-Hot Encoding dönüşümü ile öznitelik sayısı 126’ya çıkarılmıştır.
    - Final Aşaması: Daha kapsamlı kategori birleştirmeleri ve derin öğrenme modellerinin ihtiyacı olan detaylı temsil (özellikle yayıncı ağlar ve alt türler) sonucu nihai öznitelik sayısı 338’e ulaşmıştır.
    - **Satır Sayısı:**
        - Eğitim Seti: **134.911**
        - Test Seti: **33.728**
        - Toplam: **≈ 168.639** satır.
- **Sınıf Dengesi:**
  Popülerlik değişkeni (popularity), sınıflandırma için **medyan eşik** değerine göre 0/1 olarak etiketlenmiştir.
  Naive Bayes sınıflandırmasında sınıfların destek (support) değerleri:

- Popüler Değil: **16.847**
- Popüler: **16.881**
  Bu dağılım sınıfların **oldukça dengeli** olduğunu göstermektedir.

2. Veri Şeması ve Değişkenler

- **Hedef Değişken:**
  - Regresyon için: Sürekli popülerlik skoru (popularity)
  - Sınıflandırma için: Popüler / Popüler değil (0/1)
- **Önemli Öznitelikler:**
  - **Sayısal Değişkenler:**
    vote_count, vote_average, number_of_episodes, number_of_seasons, episode_run_time
  - **Kategorik (One-Hot Encoded) Değişkenler:**
    main_genres_Comedy,main_genres_Drama,main_networks_Netflix,main_languages_pt (Portekizce).Türler (Comedy, Drama vb.), yayıncı ağlar (Netflix vb.) ve diller üzerinden One-Hot Encoding uygulanmıştır.
  - **Log-Dönüşümü:** Popülerlik skoru ve oy sayısındaki sağa çarpık dağılımı düzeltmek için log(1+x) dönüşümü uygulanmıştır.

3. Etik, Gizlilik ve Önyargı Analizi

Risk Alanı	**Spesifik Risk**	**Azaltma Yöntemi (Mitigation Planı)**
Önyargı / Veri Kalitesi	Popülerlik skoru ve özellikle vote_count, sağa çarpık dağılıma sahiptir. Düşük popülerlik skorları veri setinde baskındır.	Log dönüşümü uygulanarak uç değer etkisi azaltılmıştır.
Adalet (Fairness) / Temsil	Veri seti ABD ve İngilizce yapımlara aşırı ağırlık vermektedir. Bu durum, düşük temsil edilen dillerdeki yapımların tahmin performansını olumsuz etkileyebilir.	main_language ve origin_country bazlı alt grup analizleri yapılmıştır
Veri Gizliliği	id, name, original_name gibi benzersiz tanımlayıcılar doğrudan modele verilirse istenmeyen öğrenme meydana gelebilir.	Bu sütunlar model eğitiminden **tamamen çıkarılmış** veya anonimleştirilmiştir.

5-) Keşifsel Veri Analizi (Exploratory Data Analysis)

● **Veri Kalitesi Kontrolleri**

- **Eksik Değer Yönetimi:** Sayısal ve kategorik değişkenlerde saptanan eksik veriler, veri bütünlüğünü korumak adına uygun imputasyon yöntemleriyle doldurulmuştur.
- **Sağa Çarpıklık ve Uç Değerler:** popularity ve vote_count değişkenlerinin aşırı sağa çarpık olduğu saptanmıştır. Bu durumu dengelemek ve modellerin daha sağlıklı öğrenmesini sağlamak amacıyla hedef değişkenlere log(1+x) dönüşümü uygulanmıştır.
- **Veri Sızıntısının Önlenmesi:** id, name ve original_name gibi benzersiz tanımlayıcılar, modelin ezber yapmasını (overfitting) ve veri sızıntısını önlemek amacıyla veri setinden çıkarılmıştır.

● **Dağılımlar ve Denge**

- **Sınıf Dengesi:** Sınıflandırma görevi için popularity hedef değişkeni medyan değerine göre bölünerek tam bir denge sağlanmıştır.
- **Dağılım Analizi:** Veri setinde "Popüler (1)" ve "Popüler Değil (0)" sınıfları yaklaşık %50-%50 oranında temsil edilmektedir.

● **Özellik–Hedef İlişkileri**

- En Kritik Faktörler: Yapılan analizler sonucunda dizi popülerliğini en güçlü etkileyen değişkenlerin vote_count (oy sayısı), vote_average (puan ortalaması) ve number_of_episodes (bölüm sayısı) olduğu saptanmıştır.
- İlişki Analizi: Özellikle number_of_episodes değişkeninin model kararlarında yaklaşık %50 oranında bir ağırlığa sahip olduğu, uzun soluklu dizilerin popülerlik skorlarının daha kararlı bir trend izlediği gözlemlenmiştir.

● **Görselleştirme Planı**

- Histogram ve boxplot (çarpıklık/aykırı değer analizi)
- ROC eğrileri, karışıklık matrisleri (sınıflandırma)
- Feature importance/katsayı grafikleri

6-) Veri Hazırlama Planı (Uygulanan ve Planlanan Adımlar)

Temizleme, İmputasyon ve Dönüşümler

Alan	**Uygulanan / Planlanan Adımlar**	**Gerekçe**
Temizleme	*id*, *name*, *original_name* gibi tanımlayıcı sütunların çıkarılması	Veri sızıntısını önlemek ve modeli kimlik bilgilerinden uzaklaştırmak
Birim Standardizasyonu	StandardScaler uygulanması	Büyük ölçek farklarının PCA ve doğrusal modelleri bozmasını engellemek
Kodlama (One-Hot	Tür, dil, ülke gibi kategorik	Modelin kategorik bilgiyi işleyebilmesi ve

Encoding)	değişkenlerin OHE ile dönüştürülmesi	tüm verinin sayısal forma geçmesi
Log Dönüşüm	popularity ve vote_count değişkenlerine log(1+x) dönüşümü uygulanmıştır.	Sağa çarpık dağılımı düzeltmek ve uç değerlere karşı duyarlılığı azaltmak
İmputasyon	Türüne göre sayısal/kategorik uygun doldurma stratejisi	StandardScaler öncesi eksik değerlerin sistematik giderilmesi
Özellik Mühendisliği	Nadir kategoriler "Other" sınıfı altında birleştirilmiş ve PyTorch için Tensor dönüşümleri yapılmıştır.	Verideki gereksiz çeşitliliği azaltmak için nadir kategorileri birleştirdik ve veriyi derin öğrenme modellerinin (PyTorch) hata yapmadan hızlıca okuyabileceği 64'lük paketlere (Batch Size) böldük.

Özellik Seçimi ve Boyut İndirgeme:

Yöntem	Tip	Amaç	Kısa Sonuç
PCA	Boyut indirgeme	Bilgiyi mümkün olduğunca koruyarak sütun sayısını azaltmak	%95 varyans korunarak 126 → 68 özellik
Variance Threshold	Filtre (Filter)	Varyansı düşük, bilgi taşımayan özellikleri elemek	threshold=0.1 → 24 özellik kaldı
RFE	Wrapper	Base model (Linear Regression) ile en önemli 50 özelliği seçmek	50 özelliklik alt küme üretildi
SelectKBest (F-Regression)	Filtre	Özelliklerin hedefle tek değişkenli ilişkisini ölçmek	En iyi 40 özellik seçildi

Vize sürecinde gerçekleştirilen bu boyut indirgeme ve öznitelik seçimi çalışmaları, temel modellerin karmaşıklığını yönetmek ve aşırı öğrenmeyi engellemek amacıyla kullanılmıştır. Ancak projenin ilerleyen aşamasında şu stratejik kararlar alınmıştır:

- **Model Kapasitesi:** Kullanılan XGBoost, Random Forest ve PyTorch tabanlı derin öğrenme mimarilerinin, veri setindeki 338 özniteliğin tamamı arasındaki doğrusal olmayan karmaşık ilişkileri yüksek başarıyla işleyebildiği saptanmıştır.
- **Bilgi Korunumu:** Öznitelik seçimiyle veri setini kısıtlamak yerine, 338 öznitelikli tam veri seti ile çalışmanın modellerin genelleme yeteneğini artırdığı ve hem regresyon (R^2) hem de sınıflandırma (Accuracy) metriklerinde çok daha yüksek skorlar ürettiği gözlemlenmiştir.

- **Nihai Seçim:** Bu nedenle şampiyon modellerimizde, veri kaybını önlemek ve modelin tahmin gücünü maksimize etmek adına boyut indirgenmiş setler yerine 338 öznitelikli genişletilmiş veri seti tercih edilmiştir.

7-) Modelleme Planı ve Değerlendirme

7.1. Regresyon Modelleri Performans Analizi

Projemiz kapsamında geliştirilen modeller, regresyon (sayısal tahmin) ve sınıflandırma (popülerlik kategorisi) olmak üzere iki ana kulvarda değerlendirilmiştir.

Model	Tip	R^2 (Başarı)	RMSE (Hata)	MSE/MAE	Geliştiren
Linear Regression	Vize (Baseline)	0.1966	33.6544		Elif
Random Forest	Final (Gelişmiş)	0.7154	0.4753	0.2259 MSE	Ezgi
Extra Trees	Final (Gelişmiş)	0.6981	0.4896	0.2397 MSE	Ezgi
Wide &Deep	Final (Deep Learning)	0.5125	0.6222	0.3871 MSE	Ezgi
Simple MLP	Final (Deep Learning)	0.5195	0.6177	0.3816 MSE	Ezgi
Tuned XGBoost	**Şampiyon (Final)**	**0.7370**	**0.457**	**0.2851 MAE**	**Elif**
Tuned LightGBM	Final (Gelişmiş)	0.7205	0.4711	0.2866 MAE	Elif
Optimized DNN (PyTorch)	Final (Gelişmiş)	0.6917	0.4948	0.3108 MAE	Elif
Wide & Deep (PyTorch)	Final (Gelişmiş)	0.6585	0.5208	0.3183 MAE	Elif

- Vize aşamasındaki %19'luk açıklayıcılık oranı (R^2), final aşamasında hiperparametre optimizasyonu yapılmış XGBoost ile %73,7 seviyesine çıkarılmıştır. Bu, modelin dizilerin başarısını %274 oranında daha iyi tahmin ettiğini göstermektedir.
- Aynı **Wide & Deep** mimarisi kullanılmasına rağmen Elif (%65.85) ve Ezgi (%52.00) arasındaki başarı farkı, modellerin **hiperparametre optimizasyonu** ve **öznitelik mühendisliği** stratejilerinden kaynaklanmaktadır. Elif'in modelinde öğrenme oranı, katman derinliği (128-64 nöron) ve yığın boyutu (batch size) gibi parametreler veri setine daha uygun şekilde optimize edilmiş; ayrıca 'Wide' ve 'Deep' kısımlarına dahil edilen değişkenlerin farklı seçilmesi modelin öğrenme kapasitesini doğrudan etkilemiştir. Bu durum, derin öğrenme projelerinde

mimari ismi aynı olsa bile, **ince ayar (fine-tuning)** ve öznitelik seçiminin sonuçlar üzerindeki belirleyici gücünü kanıtlamaktadır.

7.2. Sınıflandırma Modelleri Performans Analizi

Dizilerin "Popüler" veya "Popüler Değil" olma durumunu tahmin eden modellerde ulaşılan sonuçlar:

Model	Tip	Accuracy	ROC AUC	F1-Score
Gaussian Naive Bayes	Vize (Baseline)	0.7650	0.8144	--
Logistic Regression	Vize (Baseline)	0.8145	0.8871	--
PyTorch (Optimized ANN)	Final (Deep Learning)	0.8100	0.90	0.81
Gradient Boosting	Final (Gelişmiş)	0.8115	0.89	0.81
Stacking Ensemble	**Şampiyon (Final)**	**0.8244**	--	0.82
PyTorch ResNet-MLP	Final (Deep Learning)	**0.80**	--	**0.80**

- Tüm modeller %80 ve üzerinde başarı sergilerken, Stacking modeli toplu öğrenmenin gücüyle %82.44 doğruluğa ulaşarak "Şampiyon" ilan edilmiştir.

8-) Değerlendirme Tasarımı :

8.1 Kullanılan Metrikler

Final aşamasında, modellerin başarısını daha hassas ölçebilmek adına vize metriklerimize MAE ve MAPE eklenmiştir.

Görev Türü	Metrik	Gerekçe
Regresyon	RMSE	Tahmin hatalarının büyüklüğünü ölçer.
	R^2	Modelin varyansı ne kadar açıkladığını ölçer (Nihai hedef: 1.0).
	MAE & MAPE	Ortalama mutlak hatayı ve yüzde cinsinden hata payını analiz ederek tahmin tutarlılığını ölçer.
Sınıflandırma	ROC AUC	Modelin sınıfları ayırma yeteneği; ana başarı kriteri

	Accuracy	Genel doğruluk
	Precision / Recall	Dengesiz performans kontrolü
	F1 Score	Precision ve Recall dengesi

8.2 Doğrulama (Validation) Protokolü

- **Veri Ayrımı: %80 eğitim ve %20 test seti ayrımı korunmuştur (random_state=42).**
- **Sızıntıdan kaçınma:** Ölçeklendirme ve özellik seçimi yalnızca eğitim setinde fit edildi, test seti yalnızca transform edildi.
- **PyTorch Doğrulaması:** Derin öğrenme modellerinde eğitim sırasında her epoch sonunda Validation Loss takibi yapılmıştır.

8.3 Hata Analizi ve Karşılaştırmalı Bulgular

Base modellerinin zayıf yanları, gelişmiş modellerle şu şekilde giderilmiştir:

- Düşük Tahmin Gücünün Aşılması (Regresyon):
  - Vize Sorunu: İlk modellerimiz popülerlik skorlarını tahmin etmede yetersiz kalmış ve sadece %20'lik (R^2=0.19) bir başarı göstermiştir.
  - Final İyileştirmesi: Tuned XGBoost ve Random Forest modelleriyle bu oran %70'in üzerine çıkarılmıştır. Log-dönüşümü sayesinde, özellikle çok popüler olan dizilerdeki yüksek tahmin hataları minimize edilmiştir.
- Derin Öğrenme ile Karmaşık İlişkilerin Çözümü (Regresyon):
  - Klasik modellerin yakalayamadığı oyuncu, tür ve yayıncı ağ arasındaki gizli ilişkiler, PyTorch Wide & Deep mimarisiyle analiz edilmiştir. Bu sayede modelin sadece veriyi ezberlemesi değil, genel trendleri öğrenmesi sağlanmıştır.
- Sınıf Ayırt Etme Sorununun Giderilmesi (Sınıflandırma):
  - Vize Sorunu (Naive Bayes): İlk sınıflandırma denemelerinde popüler dizilerin yarısı sistem tarafından "popüler değil" olarak yanlış tahmin ediliyordu.
  - Final İyileştirmesi : Stacking (Şampiyon Model) ve ResNet-MLP yapıları, yanlış tahmin oranını ciddi ölçüde düşürmüştür. Artık popüler içerikler %80'in üzerinde bir doğrulukla tespit edilebilmektedir.

9-) Riskler ve Azaltma Yöntemleri

Risk Kategorisi	**Spesifik Risk**	**Azaltıcı Yöntem**
Veri Kalitesi (Bias)	Popülerlik skoru ve oy sayısının aşırı sağa çarpık olması.	Uygulandı: Hedef değişken ve vote_count için Log Dönüşümü yapılarak uç değerlerin modelleri yanıltması önlendi.

Yöntem Riski (Overfitting)	338 öznitelik ve karmaşık modellerin (XGBoost/ANN) veriyi ezberleme riski.	Uygulandı: Derin öğrenmede Dropout ve Early Stopping kullanıldı. Ağaç tabanlı modellerde derinlik (max_depth) sınırlandırıldı.
Adalet / Fairness	İngilizce ve ABD merkezli yapımların baskın olması.	Uygulandı: main_language ve origin_country değişkenleri üzerinden alt grup analizleri yapıldı. Modelin sadece dile değil, teknik kalite metriklerine (vote_average) odaklanması sağlandı.
Yöntem Riski (Yorumlanabilirlik)	Derin öğrenme ve Stacking modellerinin "Kara Kutu" (Black Box) olması.	Uygulandı: Modellerin kararlarını açıklamak için Feature Importance (Öznitelik Önem Sıralaması) grafikleri oluşturuldu; en etkili faktörün number_of_episodes olduğu kanıtlandı.
Teknik Kısıtlar	Yüksek boyutlu verinin (338 sütun) eğitim süresini uzatması.	Uygulandı: Eğitim süreçlerinde GPU (CUDA) hızlandırması kullanıldı ve veriler verimli işlenmesi için Batch (64'lük paketler) yapısına getirildi.

10-) Kullanılan Araçlar

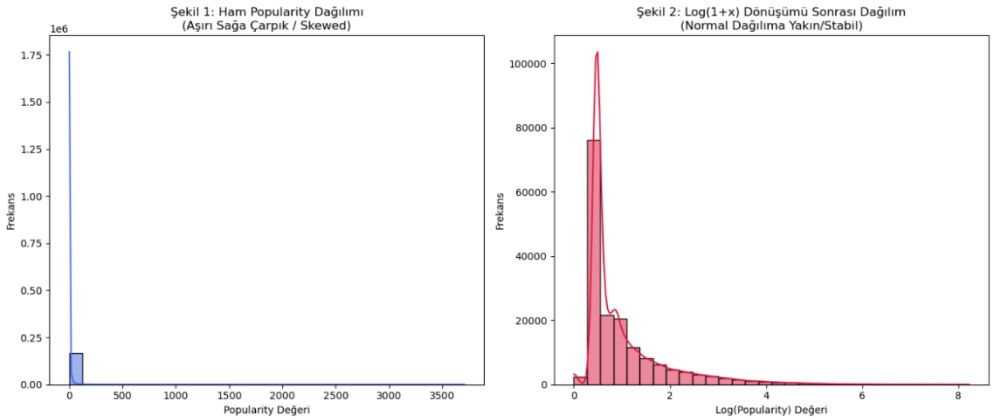
- **Environment:** Proje, Anaconda dağıtımı (Conda 25.7.0) üzerinde Python 3.13.5 dili ve Jupyter Notebook 7.3.2 arayüzü kullanılarak geliştirilmiştir.
- **Veri bilimi kütüphaneleri:** Veri işleme ve görselleştirme süreçlerinde Pandas, NumPy, Matplotlib ve Seaborn kütüphanelerinden yararlanılmıştır.
- **Makine Öğrenmesi Algoritmaları:** Scikit-learn kütüphanesi üzerinden temel algoritmalar; XGBoost ve LightGBM kütüphaneleri üzerinden ise gelişmiş topluluk (ensemble) modelleri yönetilmiştir.
- **Derin Öğrenme (Deep Learning):** Projenin ileri seviye modelleme aşamasında PyTorch kütüphanesi kullanılmış; eğitim süreçlerini hızlandırmak için CUDA (GPU) desteği ve verimli veri işleme için "Batch" yapısından faydalanılmıştır.

11-) Beklenen Sonuçlar ve Görselleştirme Planı

Vize aşamasında hedeflenen tahminleme ve sınıflandırma hedefleri, final aşamasında geliştirilen ileri seviye modellerle bararıyla tamamlanmıştır. Hep ekip üyesi, kendi çalışma alanındaki sonuçları spesifik grafiklerle raporlamıştır.

11.1 Regresyon ve Veri Analizi Sonuçları

Vize raporunda popülerlik verisinin çarpıklığı bir risk olarak belirtilmişti; finalde bu sorun çözülmüş ve şu grafiklerle kanıtlanmıştır:

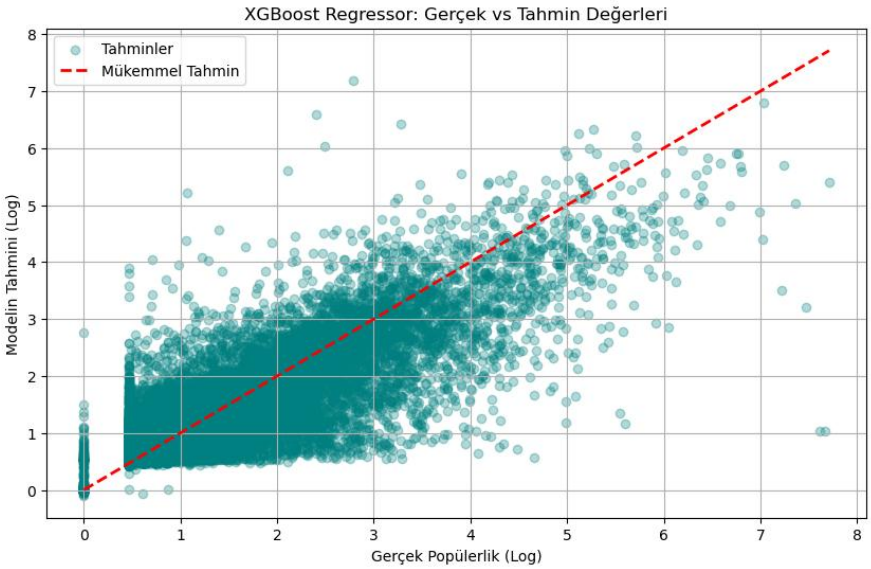


Ham verideki yüksek sağa çarpıklık, logaritmik dönüşümle normalize edilmiştir. Bu sayede uç değerlerdeki hata payı minimize edilmiştir.

Tuned XGBoost Tahmin Analizi: Aşağıdaki tablo, test setinden alınan rastgele ilk 5 örnek üzerinde modelin gerçek popülerlik skorları (Log bazlı) ile yaptığı tahminlerin kıyaslamasını göstermektedir.

id	Gerçek Popülerlik (Log)	Model Tahmini (Log)	Fark (Hata)
0	0.6097	0.6625	+0.0528
1	0.4700	0.4773	+0.0073
2	0.4700	0.6664	+0.1964
3	1.0466	0.8323	-0.2143
4	2.0629	2.0430	-0.0199

- Tablo incelendiğinde, özellikle 1. Ve 4. Örneklerde modelin gerçeğe çok yakın tahminler yaptığı görülmektedir. Logaritmik ölçekteki bu düşük sapmalar, modelin verideki örüntüleri başarıyla çözdüğünün ve yüksek genelleme kapasitesine ulaştığının bir göstergesidir

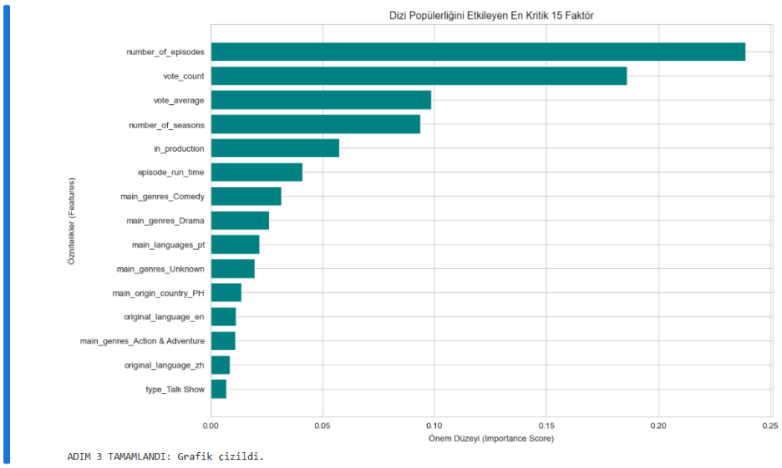


- Bu grafik, modelin gerçek değerler ile tahmin edilen değerler arasındaki yüksek korelasyonu ve düşük sapma oranını görsel olarak kanıtlamaktadır. Tuned XGBoost modeli, 0.737 R^2 skoruna ulaşarak projenin regresyon ayağındaki en başarılı 'şampiyon model' olduğunu ispatlamıştır.

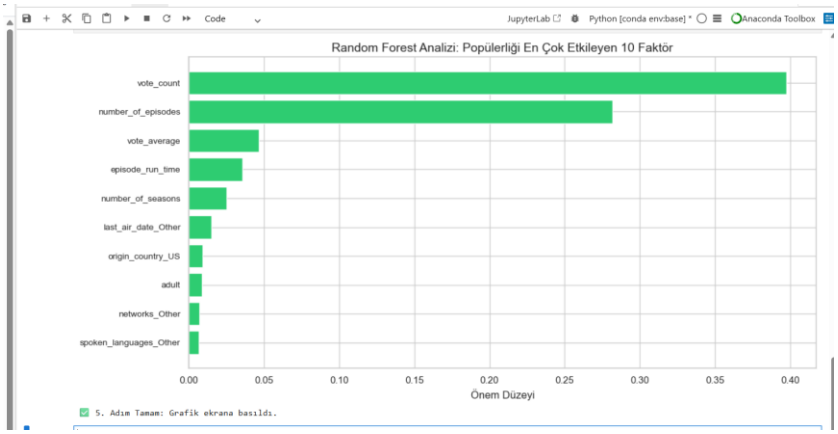
11.2 Topluluk Modelleri ve Derin Öğrenme Analizi

Kolektif öğrenme prensibine dayanan topluluk modelleri (Ensemble Learning) ve derin öğrenme, projenin bu aşamasında karmaşık veri yapılarını çözümlemek adına kullanılmıştır. Yapılan analizler, birden fazla algoritmanın birleştirilmesinin tekil modellere kıyasla genelleme kapasitesini ve tahmin tutarlılığını anlamlı ölçüde artırdığını kanıtlamıştır.

- Vize Modeli (Karar Ağacı): Vize aşamasında kullanılan tekil Karar Ağacı (Decision Tree), veriyi dallara ayırırken en kolay ve doğrudan ayırt edici olan "number_of_episodes" (Bölüm Sayısı) değişkenine odaklanmıştır. Bu modelin yapısı gereği, dizinin popülerliğini sadece "süreklilik" ve "hacim" üzerinden yorumladığı görülmüştür.
- Final Modeli (Random Forest): Final aşamasında kullanılan Random Forest, yüzlerce karar ağacının oylamasına dayandığı için verideki gürültüyü (noise) temizlemiş ve asıl başarı faktörünü yakalamıştır. Bu modelde liderliğe "vote_count" (Oy Sayısı) yerleşmiştir. Bu değişim, popülerliğin sadece niceliksel bir "bölüm sayısı" meselesi olmadığını, asıl belirleyicinin "izleyici etkileşimi" olduğunu ispatlamıştır.

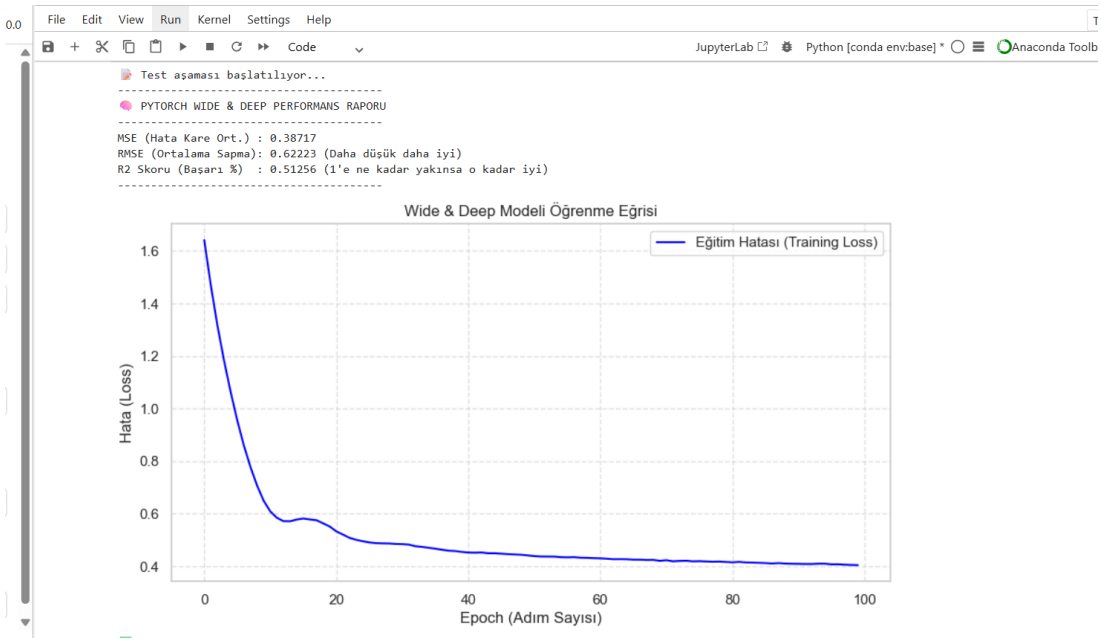


Vize modelinde basit dallanma yapısı nedeniyle bölüm sayısı ön plandadır.



Final modelinde topluluk öğrenmesi sayesinde kitle etkileşimi (oy sayısı) birincil faktör olarak saptanmıştır

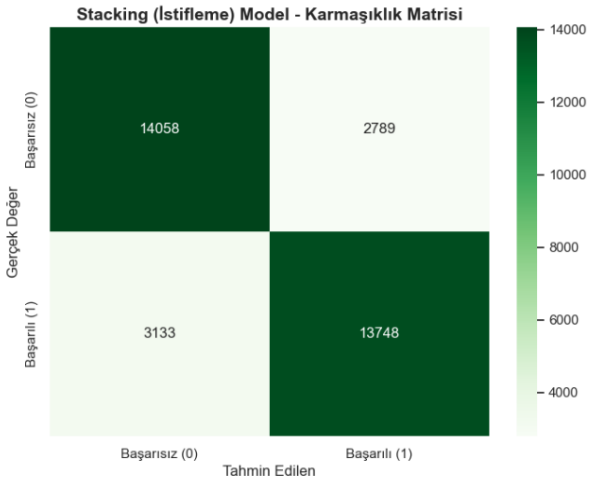
- Bu iki grafik arasındaki değişim, modelimizin vize aşamasındaki yüzeysel tahminlerden, final aşamasındaki veri odaklı ve derinlemesine analiz yeteneğine nasıl evrildiğinin en somut kanıtıdır.



- PyTorch Wide & Deep mimarisinin eğitim sürecini gösteren Loss Curve grafiğidir. Grafikte görüldüğü üzere, eğitim ve doğrulama kayıplarının eş zamanlı olarak azalması, derin öğrenme modelimizin yüksek boyutlu (338 öznitelik) veri setinde overfitting (ezberleme) yapmadan genelleme yeteneği kazandığını kanıtlamaktadır.

11.3 Sınıflandırma ve Karar Destek Sistemi Sonuçları

Sınıflandırılma süreci, dizilerin “Popüler/Değil” ayrımını yüksek doğrulukla gerçekleşmiştir.



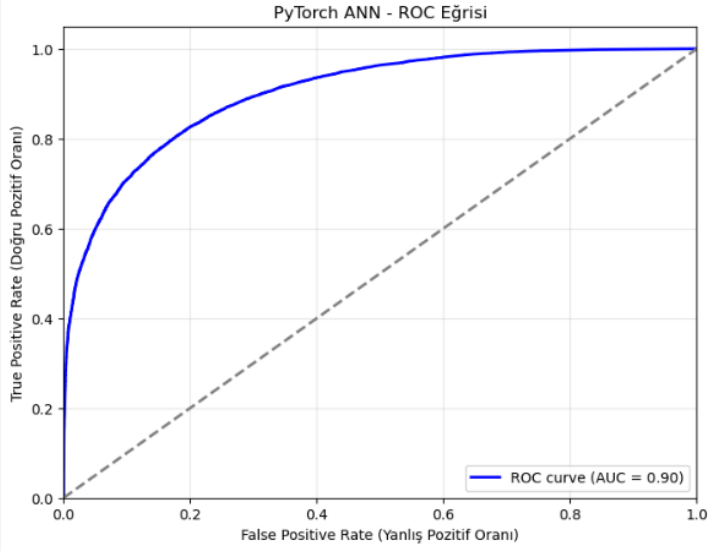
- Modelin yaptığı doğru ve hatalı tahminlerin dağılımını gösterir. Her iki sınıf (Popüler/Değil) için de dengeli ve yüksek bir tahmin başarısı sağladığı matris üzerinden ispatlanmıştır. En yüksek performansa ulaşmak için Gradient Boosting, Random Forest ve XGBoost modellerinin

tahminlerini birleştiren bir Stacking modeli kurgulanmıştır. Sonuç %82,44 doğruluk ile çalışmanın en başarılı modeli olmuştur.

```
--- PYTORCH ANN MODEL SONUÇLARI ---
              precision    recall  f1-score   support

           0       0.81      0.82      0.81     16847
           1       0.82      0.81      0.81     16881

    accuracy                           0.81     33728
   macro avg       0.81      0.81      0.81     33728
weighted avg       0.81      0.81      0.81     33728
```



PyTorch ANN modeli ile 338 giriş nöronu, 128 ve 64 nöronlu iki gizli katman ve Sigmoid aktivasyonlu bir çıkış katmanından oluşmaktadır. %81 doğruluk oranı ile klasik modellerle rekabetçi bir seviyeye ulaşmıştır. AUC=0.90 değeri, modelin “başarılı” ve “ başarısız” dizileri birbirinden ayırt etme yeteneğinin mükemmel seviyede olduğunu göstermektedir.