İZMİR KÂTİP ÇELEBİ ÜNİVERSİTESİ
2010
FEN BİLİMLERİ ENSTİTÜSÜ

Amazon.com'da Bulunan Film ve Videoların Veri Analizi ve Görselleştirmesi

Yazılım Mühendisliği Ana Bilim Dalı

Dönem Projesi

Sercan YILDIZ

Y220234112

Proje Danışmanı: Dr. Öğr. Üyesi Osman GÖKALP

Ocak 2024

Amazon.com ’da Bulunan Film ve Videoların Veri Analizi ve Görselleştirmesi

Öz

Sayın Okuyucular,

Bu dönem, Amazon.com üzerinde bulunan film ve videoların veri analizi ve görselleştirmesi üzerine bir proje yürüttüm. Bu projede, Amazon'un sunduğu geniş film ve video kütüphanesini inceleyerek, kullanıcı yorumları, derecelendirmeleri ve ilgili veriler üzerinde derinlemesine bir analiz gerçekleştirdim.

Projenin odak noktası, Amazon.com'un film ve video platformunda yer alan içeriklerin kullanıcılar üzerindeki etkilerini ve bu içeriklerin popülerlik düzeyini belirlemekti. Bu amaçla, Python gibi programlama dilleri ve veri analizi araçlarını kullanarak büyük miktarda veriyi işledim ve analiz ettim. Kullanıcı yorumlarından duygusal analizler yaparak, hangi tür içeriklerin daha fazla ilgi gördüğünü ve nedenini anlamaya çalıştım.

Ayrıca, görselleştirmeler aracılığıyla elde ettiğim verileri grafikler, tablolar ve haritalarla sunarak, Amazon.com üzerindeki film ve video içeriklerinin genel eğilimlerini ve kullanıcı tercihlerini daha açık bir şekilde ortaya koydum. Bu analizler, platformun gelecekteki içerik seçimleri veya kullanıcı deneyimi üzerindeki potansiyel etkilerini anlamak için önemli bir bakış açısı sağladı.

Bu proje, veri analizi ve görselleştirme becerilerimi geliştirme fırsatı sundu ve büyük veri kümeleriyle çalışma yeteneğimi artırdı. Aynı zamanda, gerçek dünya verileri üzerinde yapılan bu tür çalışmaların, dijital platformların içerik yönetimi ve kullanıcı memnuniyeti konularında ne kadar önemli olduğunu da kavrama fırsatı buldum. Umarım bu proje, Amazon.com gibi büyük platformların içerik yönetimi ve kullanıcı tercihlerine dair daha derinlemesine anlaşılmasına katkı sağlamıştır.

Saygılarımla,

Anahtar Sözcükler: Amazon, IMBD,Veri Analizi, Veri Görselleştirmesi

Data Analysis of Movies and Videos Available on Amazon.com

Abstract

Dear Readers,

During this term, I conducted a project focused on the data analysis and visualization of films and videos available on Amazon.com. In this project, I delved into the vast film and video library offered by Amazon, examining user reviews, ratings, and associated data through a comprehensive analysis.

The focal point of the project was to determine the impact of the content available on Amazon.com's film and video platform on users and to assess the popularity levels of this content. Utilizing programming languages like Python and various data analysis tools, I processed and analyzed substantial amounts of data. By conducting sentiment analysis on user reviews, I aimed to understand which genres of content garnered more attention and the reasons behind such trends.

Furthermore, I presented the data obtained through visualizations, such as graphs, tables, and maps, to illustrate the general trends and user preferences for film and video content on Amazon.com more explicitly. These analyses provided valuable insights into understanding the potential impacts on future content selection or user experience on the platform.

This project offered an opportunity to enhance my skills in data analysis and visualization and improved my ability to work with large datasets. Additionally, it provided insights into the significance of such studies on real-world data concerning content management and user satisfaction on digital platforms.I hope this project contributes to a deeper understanding of content management and user preferences on major platforms like Amazon.com.Best regards,

Keywords: Amazon, IMDB, Data Visualization, Data Analysis

Canım abim Umut YILDIZ,

Bu projeyi, bu yolculuk boyunca sağladığın destek ve ilham için içten bir minnet ve hayranlıkla sana ithaf ediyorum. Bu projeye yaklaşımımı ve anlayışımı şekillendirmede sağladığın rehberlik paha biçilmezdi.

Bu okulu okumamda, yaptığım ve yapacağım tüm çalışmalarda bana ilham kaynağı olduğun için teşekkür ederim. En derin minnettarlığımla,

Sercan YILDIZ

Teşekkür

Proje çalışmama katkılarından dolayı proje danışmanım Dr. Öğr. Üyesi Osman Gökalp'e teşekkür ederim.

İçindekiler

Şekiller Listesi

1.Giriş

Veri Biliminin Önemi:

Veri bilimi, dijital çağın en önemli disiplinlerinden biri olarak ön plana çıkmaktadır. Büyük verinin artan önemi ve karmaşıklığı, iş dünyasından akademiye, hükümet politikalarından günlük yaşam kararlarına kadar birçok alanda veri biliminin etkisini artırmaktadır. Bu disiplin, çeşitli veri kaynaklarından elde edilen büyük miktarda veriyi analiz ederek değerli bilgiler ve anlayışlar sağlar. Veri biliminin amacı, veriden anlam çıkarmak ve bu bilgileri stratejik karar alma, tahminleme ve optimizasyon süreçlerinde kullanmaktır.

Python'un Veri Bilimindeki Yeri:

Python, veri bilimi alanında en popüler programlama dillerinden biridir. Bu popülerliğin arkasındaki temel nedenler; Python'un okunabilirliği, esnekliği ve geniş kütüphane ekosistemi içerisinde veri analizi ve makine öğrenimi için güçlü araçlar sunmasıdır. Python, veri manipülasyonu (Pandas), sayısal hesaplamalar (NumPy), istatistiksel analiz (SciPy), veri görselleştirme (Matplotlib, Seaborn) ve makine öğrenimi (Scikit-Learn) gibi çeşitli alanlarda etkin çözümler sunar. Bu özellikler, Python'u hem amatörler hem de profesyoneller için tercih edilen bir dil haline getirmiştir.

Jupyter Notebook'un Rolü:

Jupyter Notebook, veri bilimciler arasında yaygın olarak kullanılan etkileşimli bir geliştirme ortamıdır. Kod yazma, not almak, görselleştirme yapmak ve sonuçları paylaşmak gibi işlemleri bir arada sunarak veri analizi sürecini kolaylaştırır ve daha etkileşimli hale getirir. Jupyter, analiz sürecini adım adım belgelemeye ve sonuçları anlaşılır bir şekilde sunmaya olanak tanır. Bu, özellikle akademik araştırmalar, veri bilimi eğitimi ve iş dünyasındaki raporlamalar için büyük bir avantajdır.

Bu Çalışmanın Amacı:

Bu makale, Python ve Jupyter Notebook kullanılarak yapılan veri analizinin temel adımlarını ve metodolojilerini, örnek bir Jupyter notebook üzerinden ayrıntılı bir şekilde ele almaktadır. Çalışma, veri yükleme, temizleme, analiz, görselleştirme ve sonuç çıkarma gibi temel süreçleri kapsamakta ve bu süreçlerin nasıl uygulanacağını göstermektedir. Ayrıca, kullanılan her bir kütüphane ve metodun veri bilimi pratiklerindeki önemi ve rolü detaylandırılmaktadır.

2. Metod

2.1.Verilerde Python Kütüphaneleri

2.1.1 Matplotlib ve Seaborn (Görselleştirme):

Matplotlib, Python'da statik, etkileşimli ve animasyonlu grafikler oluşturmak için kullanılan bir kütüphanedir. Bu kütüphane, veri görselleştirme için geniş bir fonksiyon seti sunar. Seaborn ise, Matplotlib tabanlı, daha yüksek seviyeli bir görselleştirme kütüphanesidir ve çekici, anlamlı istatistiksel grafikler oluşturmayı kolaylaştırır. Bu iki kütüphane, veri setlerinin görsel analizinde kullanılmakta ve veri hikayelerinin anlatılmasında etkili araçlar olarak ön plana çıkmaktadır.

2.1.2 NumPy (Sayısal Hesaplamalar):

NumPy, büyük, çok boyutlu diziler ve matrisler için destek sunan, Python programlama dilindeki temel paketlerden biridir. Bu kütüphane, veri bilimi uygulamalarında hızlı ve etkili sayısal hesaplamalar için yaygın olarak kullanılır. NumPy, veri analizinde temel algoritma ve fonksiyonlarıyla, özellikle büyük veri setleri üzerinde çalışırken hız ve verimlilik sağlar.

2.1.3 Pandas (Veri Manipülasyonu ve Analizi):

Pandas, veri manipülasyonu ve analizi için Python'da geliştirilmiş açık kaynak bir kütüphanedir. Pandas, DataFrame adı verilen etkili veri yapıları üzerinde çalışır. Bu kütüphane, veri okuma, temizleme, dönüştürme ve analiz işlemlerini kolaylaştırır. Pandas'ın sunduğu işlevsellik, veri bilimindeki temel görevler için vazgeçilmezdir ve genellikle veri setlerinin ilk işlenmesi ve keşfedici veri analizi aşamalarında kullanılır.

2.1.4 Jupyter Notebook'un Kullanımı:

Jupyter Notebook, veri bilimi projelerinde kod yazımı, not almak, sonuçları görselleştirmek ve paylaşmak için kullanılan etkileşimli bir web uygulamasıdır. Bu

ortam, araştırmacılara ve veri bilimcilere kodları hücreler halinde düzenleme ve çalıştırma imkânı sunar, böylece analiz süreçlerinin adım adım izlenmesini ve dokümante edilmesini sağlar. Jupyter, hem eğitim hem de profesyonel çalışmalarda tercih edilen bir araçtır, çünkü karmaşık veri analizlerini daha erişilebilir ve anlaşılır hale getirir.

2.2. Verileri Anlama ve İnceleme

2.2.1.Kütüphane İçe Aktarımı ve Ortam Ayarları:

İlk adımda, veri görselleştirme için Matplotlib ve Seaborn kütüphaneleri içe aktarılır. set_style ve set_palette metodları, grafiklerin görsel stilini belirlemek için kullanılır. NumPy ve Pandas, veri manipülasyonu ve analizi için temel araçlardır(Şekil2.1). Bu adım, veri analizi için gerekli kütüphanelerin kurulumunu ve yapılandırılmasını kapsar.

```
[1]: #öncelikle kullanacağımız kütüphaneleri çağırmak ile başlayalım.

import matplotlib.pyplot as plt
import seaborn as sns
sns.set_style("whitegrid")
sns.set_palette("Set2")
import numpy as np
import pandas as pd
import os
```

Şekil2.1.: Kullanılan kütüphanelerin listesi

2.2.2.Veri Yükleme ve İlk Gözlem:

Bu adımda, pd.read_csv fonksiyonu ile veri seti yüklenir ve head() metodu ile veri setinin ilk beş satırı gösterilir. Bu, veri setinin yapısını ve içeriğini ilk kez gözlemlemek için kritik bir adımdır(Şekil2.2).

```
[2]: #pd.read kodu ile datasetimizi okuyup df.head ile ilk 5 satırı gözlemleyelim.

df = pd.read_csv("C:\\Users\\Sercan\\Desktop\\Bitirmeprojesi\\dataset.csv")
df.head()
```

[2]:

	Unnamed: 0	title	Movie_Rating	No_of_Ratings	Format	ReleaseYear	MPAA_Rating	Directed_By	Starring	Price
0	0	Totally Killer	4.3	323	Prime Video	2023.0	R	Nahnatchka Khan	Kiernan Shipka,Olivia Holt,Julie Bowen	NaN
1	1	Guy Ritchie's The Covenant	4.7	13268	Prime Video	2023.0	R	Guy Ritchie	Jake Gyllenhaal,Dar Salim,Antony Starr	5.99
2	2	A Million Miles Away	4.9	1126	Prime Video	2023.0	PG	Alejandra Márquez Abella	Michael Peña,Rosa Salazar	NaN
3	3	Kelce	5.0	570	Prime Video	2023.0	NaN	Don Argott	Jason Kelce,Travis Kelce,Kylie Kelce,Connor Ba...	NaN
4	4	Despicable Me 3	4.8	31813	Prime Video	2017.0	PG	Pierre Coffin,Kyle Balda	Steve Carell,Kristen Wiig,Trey Parker	NaN

Şekil2.2.: pd.read_csv kodu ile datesetin okunması ve ilk 5 satırın görüntülenmesi çıktısı

2.2.3.Veri Boyutunun İncelenmesi:

Shape özelliği, veri setinin satır ve sütun sayısını gösterir. Bu, veri setinin genel yapısı hakkında hızlı bir fikir edinmek için kullanılır. shape özelliği, veri setinin satır ve sütun sayısını gösterir(Şekil2.3). Bu, veri setinin genel yapısı hakkında hızlı bir fikir edinmek için kullanılır.

Datayı okuyalım ve işlemlere başlayalım

```
[3]: df.shape
     #df.shape ile satır ve sütün sayısını öğreniyoruz.
[3]: (2108, 10)
```

Şekil2.3.: df.shape kodu ile satır ve sütun sayısını öğrenme

2.2.4. Veri Ön İşleme – Sütunların çıkarılması

Bu adımda, analizde kullanılmayacak sütunlar çıkarılır. drop metodu, belirtilen sütunları DataFrame'den kaldırır. inplace=True parametresi, bu değişikliklerin orijinal DataFrame üzerinde kalıcı olmasını sağlar(Şekil2.4). Bu işlem, veri setini temizlemek ve analiz için daha uygun hale getirmek adına önemlidir.

```
[4]: df.drop(columns=['Unnamed: 0','Directed_By','Starring'],inplace=True)
     #işlem yapmayacağımız sütünları düşürüp işlemlerimize devam ediyoruz
```

Şekil2.4.: df.drop kodu ile işlem yapılmayan sütünların çıkarılması

2.2.5. Kategorik Veri Analizi – Format Sütunu ve MPAA Derecelendirmesi

Value_counts() metodunun kullanımı, belirli bir sütundaki farklı değerlerin frekanslarını hesaplamak için kullanılır. Bu kod parçası, 'Format' sütunundaki değerlerin dağılımını gösterir. Yazar, bu sütunun nominal kategorik veri içerdiğini ve dolayısıyla bu veriyi modellemek için One-Hot Encoding (OHE) yönteminin kullanılacağını belirtir. Nominal verilerde sıralama olmadığı için, OHE, bu tür verileri başarılı bir şekilde modellemek için uygun bir tekniktir.

Aynı şekilde, bu kod parçası 'MPAA_Rating' sütunundaki değerlerin frekansını hesaplamak için kullanılır. Yazar, bu sütundaki değerlerin sıralı olduğunu ve bu nedenle Ordinal Encoder (OE) kullanılacağını açıklar. Sıralı kategorik veriler, değerler arasında belirli bir sıralamanın olduğu durumlarda kullanılır (Şekil2.5.) ve OE, bu tür verileri etkili bir şekilde işlemek için kullanılan bir yöntemdir.

```
[6]: df['Format'].value_counts()
     #Bu kategorik sütunda herhangi bir sıralama olmadığından bu değişkende Nominal (OHE) Encoder kullanacağız

[6]: Format
     Prime Video    2086
     4K               13
     Blu-ray           7
     DVD               2
     Name: count, dtype: int64

[7]: df['MPAA_Rating'].value_counts()

     #G/PG/PG-13/R
     # Bu kategorik sütunda bir sıralama olduğundan bu değişken üzerinde Ordinal Encoder (OE) kullanacağız

[7]: MPAA_Rating
     R        647
     PG-13    457
     PG       247
     G         27
     Name: count, dtype: int64
```

Şekil2.5.: value_counts kodu ile farklı değerlerin frekanslarının hesaplanması

2.2.6.Veri Seti Bilgisi:

Info() metodu, veri seti hakkında genel bilgiler sağlar, bu da veri setinin yapısını, sütun isimlerini ve veri tiplerini anlamak için kullanışlıdır(Şekil2.6). Ayrıca, her sütunda kaç adet non-null değer olduğunu da gösterir, bu da eksik verilerin tespitinde önemlidir.

```
[8]: df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2108 entries, 0 to 2107
Data columns (total 7 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   title          2108 non-null   object
 1   Movie_Rating   2108 non-null   float64
 2   No_of_Ratings  2108 non-null   int64
 3   Format         2108 non-null   object
 4   ReleaseYear    2069 non-null   float64
 5   MPAA_Rating    1378 non-null   object
 6   Price          1011 non-null   float64
dtypes: float64(3), int64(1), object(3)
memory usage: 115.4+ KB
```

Şekil2.6.: df.info kodu ile veri setinden bilgi alınması

2.2.7. Eksik Veri Analizi:

Bu kod, veri setindeki her sütunda bulunan eksik değerlerin (null) sayısını hesaplar. isnull() metodu ile her değerin null olup olmadığı kontrol edilir ve sum() metodu ile 11her sütundaki null değerlerin toplam sayısı hesaplanır(Şekil2.7). Eksik veri analizi, veri temizliği ve ön işleme sürecinin önemli bir parçasıdır.

```
[9]: df.isnull().sum()
     #her sütundaki boş değerleri gözlemleyelim.

[9]: title              0
     Movie_Rating       0
     No_of_Ratings      0
     Format             0
     ReleaseYear       39
     MPAA_Rating      730
     Price           1097
     dtype: int64
```

Şekil2.7.: df.isnull kodu ile veri setinin eksik veri analizi

2.2.8 İstatistiksel Özet

Describe() metodu, sayısal sütunlar için temel istatistiksel özetleri (ortalama, standart sapma, min, max vb.) sağlar(Şekil2.8). Bu istatistikler, veri setinin genel dağılımını ve eğilimlerini anlamak için yararlıdır ve analiz sürecinde önemli bir rol oynar.

```
[10]: df.describe()
      # Veriler matematiksel olarak nasıl görünüyor?
```

[10]:

	Movie_Rating	No_of_Ratings	ReleaseYear	Price
count	2108.000000	2108.000000	2069.000000	1011.000000
mean	4.484677	8090.982448	2008.199613	4.830465
std	0.255954	16157.475033	14.802774	6.717304
min	4.000000	1.000000	1931.000000	0.890000
25%	4.300000	302.000000	2002.000000	3.590000
50%	4.500000	2042.500000	2013.000000	3.790000
75%	4.700000	7717.000000	2019.000000	3.990000
max	5.000000	142807.000000	2023.000000	119.990000

Şekil2.8.: df.describe kodu ile veri setinin matematiksel görünümü

2.2.9. Yinelenen Veri Kontrolü

Bu kod, veri setindeki yinelenen satırların (duplicates) sayısını hesaplar. duplicated() metodu, her satırın yinelenip yinelenmediğini kontrol eder, ve sum() ile bu yinelenen satırların toplam sayısı bulunur(Şekil2.9). Yinelenen veriler, analizin doğruluğunu etkileyebilir, bu yüzden bu adım, veri setinin temizlenmesinde önemli bir rol oynar.

```
[11]: df.duplicated().sum()
      # Kaç tane yinelenen değerin mevcut olduğunu kontrol edelim
```

```
[11]: 219
```

Şekil2.9.: df.duplicated kodu ile yenilenen değerlerin kontrolü

2.2.10. Yinelenen Verilerin Çıkarılması

Bu kod, veri setinden yinelenen satırları çıkarır. drop_duplicates() metodu, yinelenen satırları tanımlar ve çıkarır. keep='first' argümanı, her yinelenen grup içindeki ilk satırın tutulmasını sağlar, diğerleri çıkarılır. inplace=True argümanı, bu değişikliklerin orijinal DataFrame üzerinde kalıcı olmasını sağlar(Şekil2.10). Bu işlem, veri setinin daha temiz ve doğru bir analiz için hazırlanmasında kritik bir adımdı

```
[12]: df.drop_duplicates(keep='first',inplace=True)
      # Kopyaları çıkartalım
```

Şekil2.10.: df.drop_duplicates kodu ile veri setindeki kopyaların çıkarılması

2.2.11 Yinelenen Verilerin Kontrolü (Tekrar)

Bu kod, önceki yinelenen veri kontrolüne benzer bir işlemi tekrarlamaktadır. duplicated().sum() ifadesi, veri setindeki yinelenen (duplicate) satırların sayısını hesaplar(Şekil2.11). Bu, özellikle büyük veri setlerinde, ön işleme aşamalarının doğru bir şekilde gerçekleştirilip gerçekleştirilmediğini kontrol etmek için tekrarlanan bir adım olabilir. Yinelenen verilerin varlığı, analizin doğruluğunu ve güvenilirliğini etkileyebileceği için, bu tür kontroller veri temizliği sürecinin önemli bir parçasıdır.

```
[13]: df.duplicated().sum()

[13]: 0
```

Şekil2.11.: df.duplicated kodu ile yenilenen değerlerin kontrolü(tekrar)

2.3. Eksik Değerleri Değiştirme

2.3.1. 'ReleaseYear' Sütunu için Eksik Değerlerin Doldurulması

Bu kod, 'ReleaseYear' sütunundaki eksik değerleri (NaN) bu sütunun modu (en sık rastlanan değer) ile doldurur. fillna metodu, eksik değerleri belirli bir değerle doldurmak için kullanılır. df['ReleaseYear'].mode()[0] ifadesi, 'ReleaseYear' sütunundaki en sık rastlanan değeri bulur. Bu yöntem, özellikle kategorik veya sıralı verilerde eksik değerlerin doldurulmasında yaygın olarak kullanılır. Burada, en sık rastlanan yıl, eksik yıllar için mantıklı bir yerine koyma olarak seçilmiştir.

2.3.2 'MPAA_Rating' Sütunu için Eksik Değerlerin Doldurulması

Bu satır, 'MPAA_Rating' sütunundaki eksik değerleri, bu sütunun modu ile doldurur. Bu durumda, en sık rastlanan MPAA derecesi, eksik derecelendirmeler için kullanılır. 'MPAA_Rating' gibi kategorik sütunlarda, mod değeri, eksik verileri doldurmak için sıkça tercih edilen bir yöntemdir, çünkü bu değer, mevcut veri dağılımını yansıtan temsili bir değer sağlar.

2.3.3. 'Price' Sütunu için Eksik Değerlerin Doldurulması:

Bu kod, 'Price' sütunundaki eksik değerleri bu sütunun ortalaması ile doldurur. fillna metodunun kullanımı, bu kez sayısal bir sütun için yapılmaktadır ve eksik değerler, sütunun ortalama (mean) değeri ile doldurulur(Şekil2.12). Sayısal verilerde ortalama değer, eksik değerler için yaygın olarak kullanılan bir doldurma yöntemidir, çünkü bu, veri setinin genel eğilimini korumaya yardımcı olabilir.

```
[14]: df['ReleaseYear'] = df['ReleaseYear'].fillna(df['ReleaseYear'].mode()[0])
      df['MPAA_Rating'] = df['MPAA_Rating'].fillna(df['MPAA_Rating'].mode()[0])
      df['Price'] = df['Price'].fillna(df['Price'].mean())


      # Eksik değerlerin adım adım değiştirilmesi
      # 1. ReleaseYear'da en sık görülen değerleri değiştirmek istiyoruz çünkü değerlerin yalnızca %2'si eksik
      # 2. Fiyat'ta eksik değerleri Fiyatın ortalama değerleriyle değiştirmek istiyoruz
      # 3. MPAA_Rating sütununda eksik değerleri en sık görülenlerle değiştirmek istiyoruz
```

Şekil2.12.: .fillna metodunun kullanımı ile eksik değerlerin doldurulması

2.4. Eksik Veri Analizi

Bu kod, veri setindeki her bir sütunda bulunan eksik değerlerin (null) sayısını hesaplar. isnull() metodu, DataFrame'deki her bir değerin eksik (null) olup olmadığını kontrol eder ve boolean bir sonuç döndürür. Sonrasında, sum() metodu kullanılarak, her sütundaki True değerleri (yani eksik değerler) toplanır(Şekil2.13). Bu, veri setindeki eksik değerlerin genel bir özetini sağlar ve hangi sütunlarda eksik değerlerin olduğunu ve bu eksik değerlerin miktarını gösterir.

Eksik veri analizi, veri bilimi projelerinde önemli bir adımdır. Eksik veriler, analiz sonuçlarını etkileyebilecek önemli faktörlerdendir. Bu nedenle, veri setindeki eksik değerlerin tespit edilmesi, bu eksikliklerin uygun bir şekilde ele alınması ve doldurulması veya çıkarılması gerekmektedir. Eksik verilerin doğru bir şekilde yönetilmesi, veri setinin kalitesini ve dolayısıyla analiz sonuçlarının güvenilirliğini artırır.

```
[15]: df.isnull().sum()
```

```
[15]: title           0
      Movie_Rating    0
      No_of_Ratings   0
      Format          0
      ReleaseYear     0
      MPAA_Rating     0
      Price           0
      dtype: int64
```

Şekil2.13.: df.isnull().sum() kodu ile eksik değerlerin toplanması

3. Veriler ile Çalışma

3.1. Veri Gruplandırma

İlk satır, groupby('Format') metoduyla, 'Format' sütunundaki benzersiz değerler bazında veri setini gruplandırır (Şekil3.1.). Bu, veri setini 'Format' sütunundaki her bir benzersiz değere göre ayırır ve bu değerlere göre gruplanmış bir nesne oluşturur.

groupby işlemi, veri setini belirli bir kriter (bu durumda 'Format') etrafında segmentlere ayırmak için kullanılır. Bu, belirli bir sütuna göre verilerin toplu analizi için son derece yararlıdır.

Oluşturulan mformat nesnesi, gruplandırılmış veri üzerinde çeşitli işlemler yapmak için kullanılabilir. Bu işlemler arasında toplama, ortalama alma, maksimum veya minimum değerleri bulma gibi istatistiksel hesaplamalar yer alabilir.

Kodun ikinci satırı (mformat), oluşturulan gruplandırma nesnesini gösterir. Ancak, bu nesne doğrudan verileri göstermez; bunun yerine, veri setinin nasıl gruplandığını temsil eden bir nesnedir.

Gruplandırma, veri analizinde kritik bir rol oynar. Özellikle büyük veri setlerinde, belirli kategorilere göre verilerin ayrıştırılması ve bu kategorilerin her birinin ayrı ayrı incelenmesi, daha detaylı ve özelleştirilmiş analizler yapılmasını sağlar. 'Format' gibi kategorik sütunlar, gruplandırma için sıklıkla tercih edilen sütunlardır.

```
[17]: mformat = df.groupby('Format')
      mformat

[17]: <pandas.core.groupby.generic.DataFrameGroupBy object at 0x00000188C5646A10>
```

Şekil3.1.: df.groupby kodu ile verilerin guruplandırılması

3.2.Gruplandırılmış Veri Üzerinde Toplama İşlemi:

Bu kod, groupby metodunu kullanarak oluşturulan mformat nesnesi üzerinde sum() işlevini uygular(Şekil3.2). Burada, 'Format' sütununa göre gruplandırılmış veri setindeki sayısal sütunlar için toplam değerler hesaplanır.

sum() fonksiyonu, her gruptaki sayısal değerlerin toplamını alır. Bu, gruplandırılmış verilerin sayısal özelliklerini anlamak için kullanışlı bir yöntemdir ve her gruptaki toplam değerleri gösterir.

Bu işlem, veri setindeki farklı 'Format' kategorilerinin sayısal özelliklerine, özellikle de toplam değerlerine ilişkin önemli içgörüler sağlar. Örneğin, eğer 'Price' veya 'No_of_Ratings' gibi sütunlar varsa, bu kod her bir 'Format' kategorisi için bu değerlerin toplamını verecektir.

Sonuç, her 'Format' kategorisine ait sayısal verilerin toplamını içeren bir DataFrame'dir. Bu, belirli bir 'Format'ın veri setindeki diğer kategorilere göre nasıl bir ağırlığa sahip olduğunu gösterir ve analize farklı boyutlar ekleyebilir.

Gruplandırma ve toplama işlemleri, veri analizinde güçlü araçlardır. Bu tür işlemler, veri setinin belirli bölümlerini daha detaylı analiz etmek, özellikle de kategorik sütunlara göre sayısal verilerin nasıl dağıldığını görmek için kullanılır.

```
[18]: mformat.sum()
```

[18]:

Format	title	Movie_Rating	No_of_Ratings	ReleaseYear	MPAA_Rating	Price
4K	The Lord of the Rings: The Motion Picture Tril...	60.7	283406	26299.0	RRRRRRRRRRRR	521.560000
Blu-ray	The Super Mario Bros. Movie (Blu-Ray + DVD + D...	32.2	111232	14161.0	RRRRRRR	210.570000
DVD	SOUND OF FREEDOMHarry Potter: The Complete 8-F...	8.9	86573	4046.0	RR	69.950000
Prime Video	Totally KillerGuy Ritchie's The CovenantA Mill...	8370.7	15150000	3749713.0	RRPGRPGRRPG-13PG-13RPGPG-13PGPG-13PG-13RRPG-13...	8442.078749

Şekil3.2.: sum() kodu ile verilerin toplanması ve çıktısı

3.3. Gruplandırılmış Veri Üzerinde Minimum ve Maksimum Değerlerin Hesaplanması

3.3.1. Minimum Değerlerin Hesaplanması

Bu kod, mformat adı verilen, 'Format' sütununa göre gruplandırılmış veri setindeki her gruptaki minimum değerleri hesaplar. min() fonksiyonu, her gruptaki sayısal ve bazı durumlarda kategorik sütunlar için minimum değerleri bulur.

Özellikle, bu kod her bir 'Format' kategorisinin içerdiği sayısal ve kategorik sütunlardaki en küçük değerleri belirler. Örneğin, eğer veri setinde 'Price', 'ReleaseYear' gibi sayısal sütunlar veya kategorik sütunlar varsa, bu işlem her bir 'Format' kategorisi için bu sütunların minimum değerlerini verecektir.

Bu işlem, her bir 'Format' kategorisindeki minimum değerler hakkında içgörüler sağlar ve kategorilere özgü en düşük değerleri gösterir.

3.3.2. Maksimum Değerlerin Hesaplanması

Bu kod, 'Format' sütununa göre gruplandırılmış veri setindeki her gruptaki maksimum değerleri hesaplar. max() fonksiyonu, her gruptaki sayısal ve bazı durumlarda kategorik sütunlar için maksimum değerleri bulur.

Benzer şekilde, bu kod her bir 'Format' kategorisinin içerdiği sayısal ve kategorik sütunlardaki en büyük değerleri belirler (Şekil3.3). Bu, her bir 'Format' kategorisinin üst sınırlarını gösterir ve kategorilere özgü en yüksek değerleri ortaya çıkarır.

```
[19]: mformat.min()
```

[19]:

Format	title	Movie_Rating	No_of_Ratings	ReleaseYear	MPAA_Rating	Price
4K	Barbie (4K Ultra HD + Digital) [4K UHD]	4.4	183	2023.0	R	15.59
Blu-ray	Barbie (Blu-Ray + Digital)	4.1	61	2023.0	R	12.99
DVD	Harry Potter: The Complete 8-Film Collection [...	4.1	61	2023.0	R	19.96
Prime Video	#Unfit: The Psychology of Donald Trump	4.0	1	1931.0	G	0.89

```
[20]: mformat.max()
```

[20]:

Format	title	Movie_Rating	No_of_Ratings	ReleaseYear	MPAA_Rating	Price
4K	WARNER BROS Live Die Repeat: Edge of Tomorrow ...	4.8	86512	2023.0	R	119.99
Blu-ray	Transformers: Rise of the Beasts [Blu-ray]	4.9	43543	2023.0	R	54.99
DVD	SOUND OF FREEDOM	4.8	86512	2023.0	R	49.99
Prime Video	¿Quieres ser mi hijo?	5.0	142807	2023.0	R	24.99

Şekil3.3.: .min() ve .max() kodu ile gruplandırılmış verilerin değerlerinin hesaplanması

3.4. 'Format' Gruplarına Göre Sıralanması ve İlk Beş Değerin Gösterilmesi:

3.4.1. Fiyata Göre Toplamlarının Sıralanması

df.groupby('Format').sum()['Price'] ile, veri setindeki 'Format' sütununa göre gruplandırma yapılır ve her grup için 'Movie_Rating' sütununun ortalama değerleri hesaplanır(Şekil3.4.).

```
[21]: df.groupby('Format').sum()['Price'].sort_values(ascending=False).head()
```

```
[21]: Format
      Prime Video    8442.078749
      4K              521.560000
      Blu-ray         210.570000
      DVD              69.950000
      Name: Price, dtype: float64
```

Şekil3.4.: df.groupby('Format').sum()['Price'] kodu ile 'Format' grubunun fiyata göre toplamlarının sıralanması

3.4.2. Derecelendirme Sayısına Göre Toplamlarının Sıralanması

df.groupby('Format')['Movie_Rating'].mean() ile, veri setindeki 'Format' sütununa göre gruplandırma yapılır ve her grup için 'Movie_Rating' sütununun ortalama değerleri hesaplanır(Şekil3.5.).

```
[22]: df.groupby('Format')['Movie_Rating'].mean().sort_values(ascending=False).head(1)
```

```
[22]: Format
      4K    4.669231
      Name: Movie_Rating, dtype: float64
```

Şekil3.5.: df.groupby('Format')['Movie_Rating'].mean() kodu ile 'Format' grubunun derecelendirme sayısına göre toplamlarının sıralanması

3.4.3. Film Derecelendirmesine Göre Toplamlarının Sıralanması

df.groupby('Format')['No_of_Ratings'].mean() ile, 'Format' sütununa göre gruplandırılan veri setinde 'No_of_Ratings' sütununun her gruptaki ortalama değerlerini hesaplamaktır(Şekil3.6.).

```
[23]: df.groupby('Format')['No_of_Ratings'].mean().sort_values(ascending=False).head(1)
```

```
[23]: Format
      DVD    43286.5
      Name: No_of_Ratings, dtype: float64
```

Şekil3.6.: df.groupby('Format')['No_of_Ratings'].mean() kodu ile 'Format' grubunun film derecelendirmesine göre toplamlarının sıralanması

3.5. 'Title' Gruplarına Göre Sıralanması ve İlk Beş Değerin Gösterilmesi:

3.5.1. Fiyata Göre Toplamlarının Sıralanması

df.groupby('title')['Price'].sum() ile, veri setindeki 'title' sütununa göre gruplandırma yapılır ve her grup için 'Price' sütununun toplam değerleri hesaplanır(Şekil3.7.).

```
[24]: df.groupby('title')['Price'].sum().sort_values(ascending=False).head(10)
```

```
[24]: title
      Game of Thrones: The Complete Collection [4K UHD]                                                                  119.99
      Harry Potter: 8-Film Collection [4K Ultra HD + Blu-ray] [4K UHD]                                                    95.99
      The Lord of the Rings: The Motion Picture Trilogy (Extended & Theatrical)(4K Ultra HD)                              76.99
      The James Bond Collection [Blu-ray]                                                                                 54.99
      Batman: The Complete Animated Series [Blu-ray]                                                                      52.75
      Harry Potter: The Complete 8-Film Collection [DVD]                                                                  49.99
      Universal Classic Monsters: Icons of Horror Collection [4K UHD]                                                     29.99
      Universal Classic Monsters: Icons of Horror Collection (The Mummy / The Bride of Frankenstein / Phantom of the Opera / Creature from the Black Lagoon) [4K UHD]    29.99
      Barbie (4K Ultra HD + Digital) [4K UHD]                                                                             29.96
      Barbie                                                                                                              24.99
      Name: Price, dtype: float64
```

Şekil3.7.: df.groupby('title')['Price'].sum() kodu ile 'Title' grubunun fiyata göre toplamlarının sıralanması

3.5.2. Film Derecelendirmesine Göre Toplamlarının Sıralanması

df.groupby('title')['Movie_Rating'].sum() ile, veri setini 'title' sütunu bazında gruplandırır ve her grup için 'Movie_Rating' sütununun toplam değerlerini hesaplar(Şekil3.8).

```
[26]: df.groupby('title')['Movie_Rating'].sum().sort_values(ascending=False).head()

[26]: title
      Legally Blonde               9.6
      Tim Burton's Corpse Bride    9.6
      Puss in Boots                9.5
      From Russia with Love        9.4
      Interstellar                 9.4
      Name: Movie_Rating, dtype: float64
```

Şekil3.8.:df.groupby('title')['Movie_Rating'].sum() ile 'title' grubunun film derecelendirmesine göre toplamlarının sıralanması

3.5.3. M.P.A.A Derecelendirmesine Göre Toplamlarının Sıralanması

Bu kod, veri setini 'MPAA_Rating' sütunu bazında gruplandırır. 'MPAA_Rating', Motion Picture Association of America tarafından belirlenen film derecelendirme sisteminin bir parçasıdır ve filmlerin yaş sınıflandırmasını ifade eder. Bu sütun, filmlerin derecelendirmesini içerir(Şekil3.9.).

```
mpaarating = df.groupby('MPAA_Rating')
mpaarating

<pandas.core.groupby.generic.DataFrameGroupBy object at 0x00000188C56985B0>
```

Şekil3.9.: Veri setinin 'MPAA_Rating' sütunu bazında gruplandırılması

3.5.4. G Derecesine Sahip Film Derecelendirmesine Göre Toplamlarının Sıralanması

df[df['MPAA_Rating'] == 'G'] kodu ile, 'MPAA_Rating' sütunu 'G' (Genel İzleyici) derecesine sahip olan filmleri filtreler(Şekil3.10.).

```
[27]: grated = df[df['MPAA_Rating'] == 'G']
      grated.groupby('title')['Movie_Rating'].sum().sort_values(ascending=False).head(3)
```

```
[27]: title
      Romance in Hawaii                 5.0
      Spirit: Stallion of the Cimarron  4.8
      Charlotte's Web                   4.8
      Name: Movie_Rating, dtype: float64
```

Şekil3.10.: 'MPAA_Rating' sütununun Genel İzleyici derecesinin filtrelenmesi

3.6. Filmlerin 'Movie_Rating' Sıralamalarının Belirlenmesi

Fonksiyonun ana işlevi, veri grubundaki 'Movie_Rating' sütununu kullanarak her film başlığının sıralamasını belirlemektir. Bu işlem group['Movie_Rating'].rank(ascending=False) ifadesi ile gerçekleştirilir. 'ascending=False' argümanı, büyükten küçüğe doğru sıralama yapılmasını sağlar, yani en yüksek puanı alan film başlığı en yüksek sıraya sahip olur.

Sıralama sonuçları, 'title' adında yeni bir sütunda saklanır. Bu sütun, her film başlığının sıralamasını içerir(Şekil3.11.).

```
[28]: def ranking(group):
          group['title'] = group['Movie_Rating'].rank(ascending=False)
          return group
```

Şekil3.11.: 'Movie_Rating' sütunun büyükten küçüğe sıralanarak 'title' adında yeni bir sütunda saklanması

3.7. 'Ranking' İşleminin 'mpaarating' Grubuna Uygulanması

Bu işlem, farklı 'MPAA_Rating' kategorilerine sahip filmlerin sıralamalarını belirlemek ve bu kategoriler arasındaki film sıralamalarını karşılaştırmak için kullanılır(Şekil3.12.).

```
[29]: mpaarating.apply(ranking)
```

[29]:

MPAA_Rating		title	Movie_Rating	No_of_Ratings	Format	ReleaseYear	MPAA_Rating	Price
G	**90**	5.0	4.8	20083	Prime Video	2021.0	G	3.890000
	91	21.0	4.4	938	Prime Video	2017.0	G	4.893679
	115	21.0	4.4	117	Prime Video	2019.0	G	4.893679
	118	25.5	4.0	6	Prime Video	2023.0	G	4.893679
	252	16.0	4.6	4305	Prime Video	2015.0	G	4.893679
...	...	...	...	...	...	...	...	...
R	**2099**	387.5	4.6	36	Prime Video	1970.0	R	4.893679
	2101	697.0	4.4	31	Prime Video	2012.0	R	4.893679
	2103	91.5	4.8	30	Prime Video	2023.0	R	2.990000
	2104	16.0	5.0	2	Prime Video	2023.0	R	4.893679
	2106	697.0	4.4	1338	Prime Video	2019.0	R	4.893679

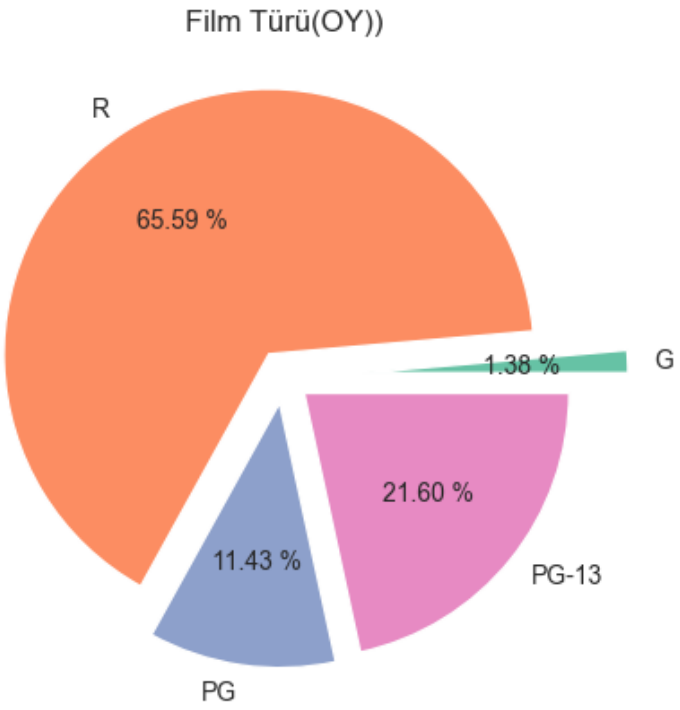
1889 rows × 7 columns

Şekil3.12.: 'Ranking' İşleminin ‘mpaarating’ Grubuna Uygulanması

4. Grafik İnceleme ve Değerlendirme

4.1.MPAA Derecelendirmesine Göre Film Türleri

Bu grafik, veri setindeki filmlerin MPAA derecelendirmelerine göre dağılımını gösterir (G, R, PG, PG-13). Patlama efekti eklenerek, her bir derecelendirmenin oransal temsilini vurgular(Şekil4.1.). Bu grafikten, hangi derecelendirmenin en yaygın veya en nadir olduğu gibi eğilimler çıkarılabilir.

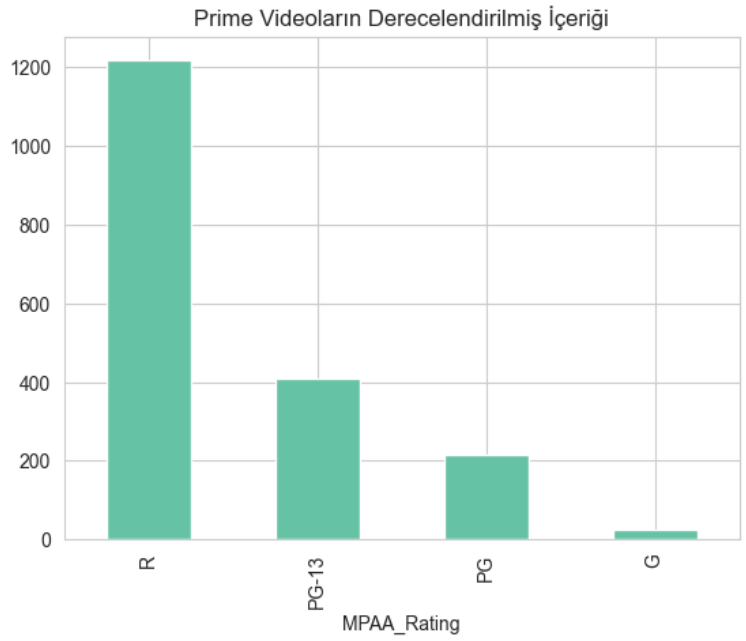


Şekil4.1.: Veri setindeki filmlerin MPAA derecelendirmelerine göre dağılımı grafiği

Bu grafik, belirli bir derecelendirmenin ne kadar yaygın olduğunu gösterir. Örneğin, 'G' derecelendirmesi aile dostu filmleri temsil ederken, 'R' derecelendirmesi yetişkinlere yönelik içerikleri temsil eder. Patlama efektiyle vurgulanan bu oranlar, dağıtım platformlarının veya film yapımcılarının hedef kitlelerini anlamalarına yardımcı olabilir.

4.2. Amazon Prime Video İçeriğinin MPAA Derecelendirmesi

Amazon Prime Video üzerindeki filmlerin MPAA derecelendirmelerine göre dağılımını gösteren bir çubuk grafiği(Şekil4.2.). Bu, platformun demografik yapısını ve hedef kitlesini anlamak için yararlı olabilir.

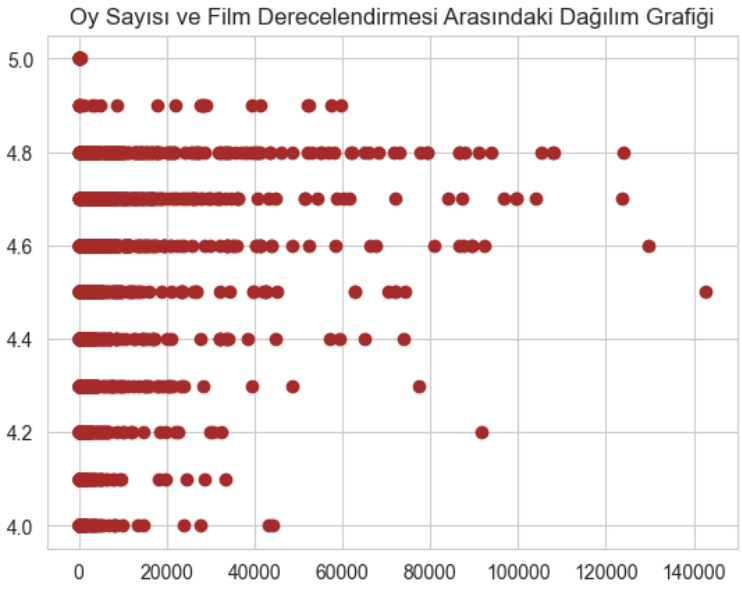


Şekil4.2.: Prime Videoların Derecelendirilmiş İçeriğinin grafiği

Bu grafik, Prime Video'nun hedef kitlesini ve içerik stratejisini anlamak için önemli bir gösterge. Örneğin, daha fazla 'PG' veya 'PG-13' içeriğe sahip olması, platformun genç yetişkinlere veya ailelere hitap etmeye çalıştığını gösterebilir.

4.3. Oy Sayısı ve Film Derecelendirmesi Arasındaki İlişki

Bu saçılım grafiği, filmlerin aldığı toplam oy sayısı ile bu filmlerin ortalama derecelendirmeleri arasındaki ilişkiyi gösterir (Şekil4.3.). Grafikten, daha yüksek veya düşük derecelendirmelerin oy sayılarına etkisinin gözlemlenmesi mümkündür.

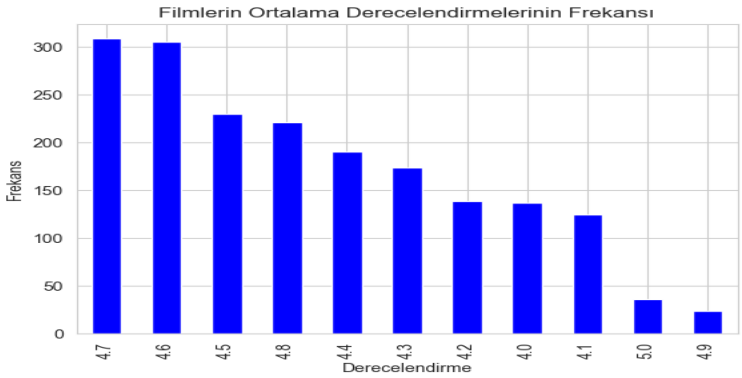


Şekil4.3.: Oy Sayısı ve Film Derecelendirmesi Arasındaki Dağılım Grafiği

Bu grafik, Yüksek oy sayılarına sahip filmlerin derecelendirmeleri, popülerliğin ve kalitenin bir göstergesi olabilir. Bu grafik, izleyici kitlesinin tercihleri ve eğilimleri hakkında fikir verir.

4.4. Film Derecelendirmelerinin Frekans Dağılımı

Filmlerin ortalama derecelendirmelerinin frekansını gösteren bir çubuk grafiği (Şekil4.4.). Bu, veri setindeki filmlerin genel kalitesi hakkında fikir verir.

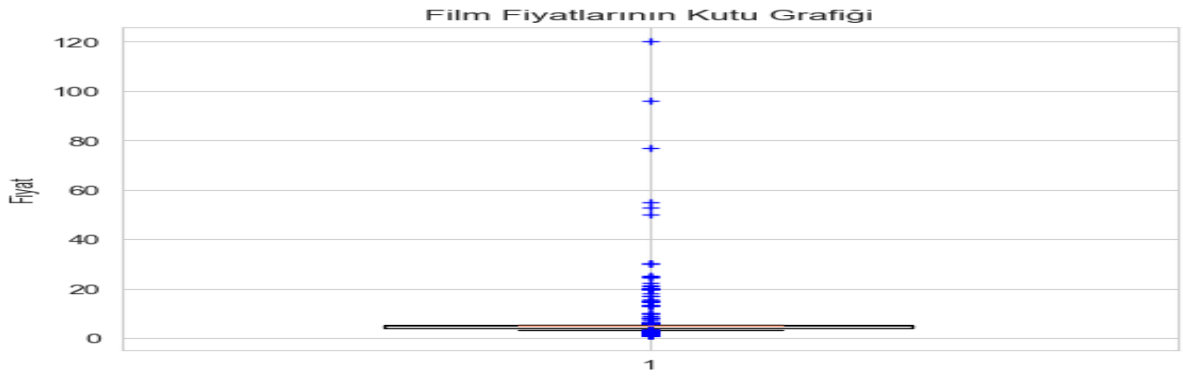


Şekil4.4.: Filmlerin ortalama derecelendirmelerinin frekansını gösteren bir çubuk grafiği

Bu grafik, veri setindeki filmlerin genel kalitesi hakkında fikir verir. Örneğin, yüksek derecelendirmeli filmlerin sayısı, veri setindeki kaliteli içeriğin bir göstergesi olabilir.

4.5. Film Fiyatlarının Kutu Grafiği

Bu kutu grafiği, film fiyatlarının dağılımını gösterir (Şekil4.5.). Medyan, çeyreklikler ve aykırı değerler bu grafikle net bir şekilde görülebilir.

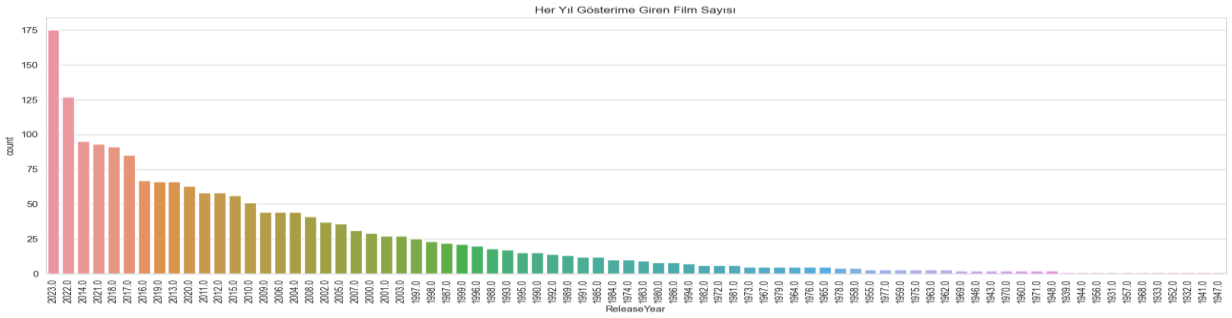


Şekil4.5.: Film fiyatlarının kutu grafiği

Bu grafik, fiyatlandırma stratejileri ve pazar dinamikleri hakkında bilgi sağlar. Örneğin, fiyat dağılımı, belirli fiyat noktalarında yoğunlaşma veya geniş bir fiyat aralığı gösterebilir.

4.6. Yıllara Göre Film Çıkış Sayıları

Her yıl piyasaya sürülen filmlerin sayısını gösteren bir çubuk grafiği (Şekil4.6.). Bu, zaman içinde film üretiminin nasıl değiştiğini gösterir.

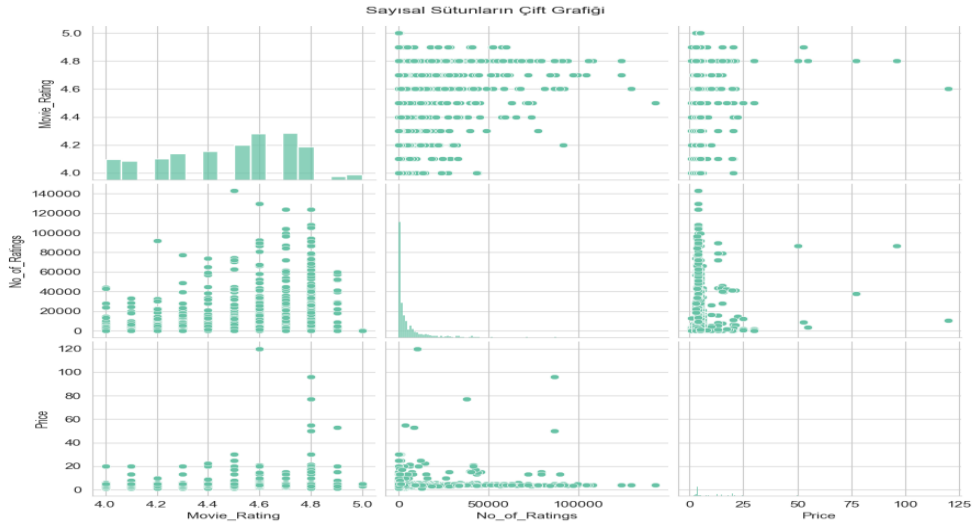


Şekil4.6.: Her yıl gösterime giren film sayısı grafiği

Bu grafik, zaman içinde film üretiminin nasıl değiştiğini gösterir. Bu, sektörel trendler ve teknolojik gelişmelerle ilişkilendirilebilir.

4.7. Sayısal Değişkenlerin Çift Grafiği

Film derecelendirmesi, oy sayısı ve fiyat gibi sayısal değişkenler arasındaki ilişkileri gösteren çift grafikler(Şekil4.7.). Bu, değişkenler arasındaki korelasyonları ve ilişkileri keşfetmek için kullanılır.

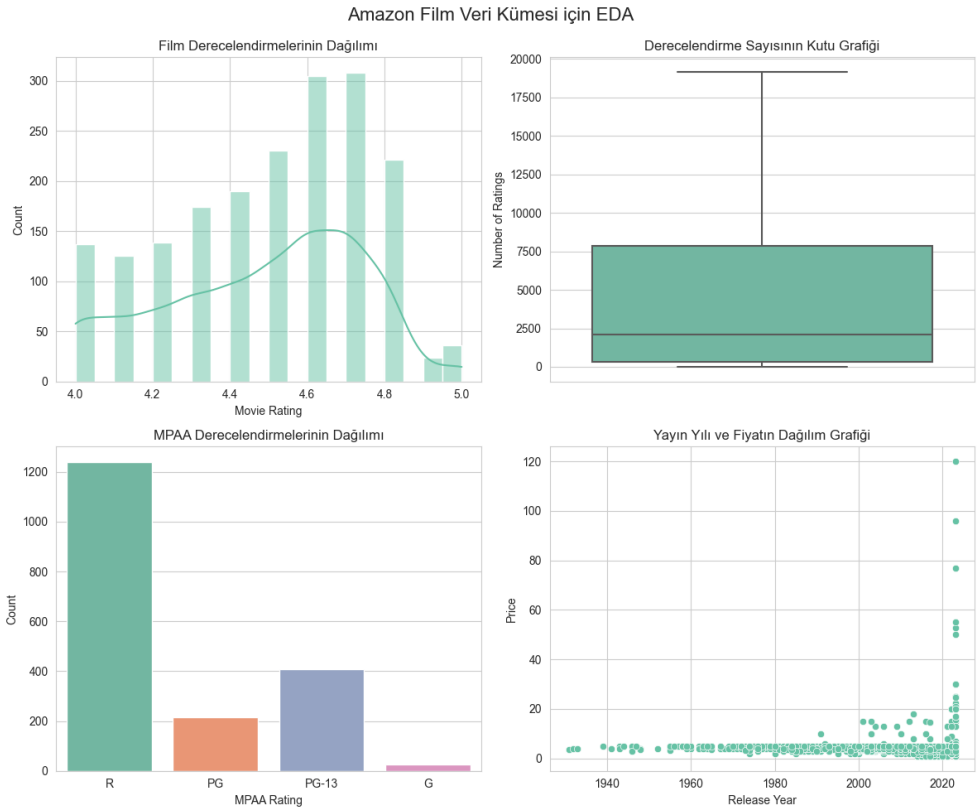


Şekil4.7.: Sayısal Değişkenlerin Çift Grafiği

Bu grafik, farklı değişkenlerin birbirleriyle nasıl ilişkili olduğunu gösterir. Örneğin, yüksek fiyatlı filmlerin daha fazla oy alıp almadığını veya belli bir derecelendirmeye sahip filmlerin ortalama fiyatlarının nasıl olduğunu anlamak mümkündür.

4.8. Çeşitli Grafiklerle EDA(Keşifsel Veri Analizi)

Birden fazla alt grafik içeren bir figür(Şekil4.8.). Bu grafikler, film derecelendirmelerinin dağılımı, derecelendirme sayılarının kutu grafiği, MPAA derecelendirmelerinin dağılımı ve yayın yılı ile fiyatın dağılım grafiği gibi farklı analizleri bir arada sunar. Bu çok yönlü yaklaşım, veri setinin kapsamlı bir şekilde incelenmesini sağlar ve farklı açılardan anlayış kazandırır.



Şekil4.8.: Amazon film veri kümesi için EDA

Bu kapsamlı grafik, veri setinin çeşitli yönlerini aydınlatır. Her grafik, filmler hakkında farklı bir hikâye anlatır ve bu hikayelerin birleşimi, veri setinin genel bir resmini çizer.

5. Sonuç

Bu analizler, film endüstrisi, içerik dağıtım platformları ve pazarlama stratejileri hakkında değerli öngörüler sunar. Daha bilinçli kararlar almalarına yardımcı olur. Her bir grafik, pazarın farklı bir yönünü aydınlatarak, sektördeki dinamiklerin daha iyi anlaşılmasına katkıda bulunur. Bu analizler aynı zamanda, izleyici davranışlarını ve beklentilerini anlamak için de kullanılabilir, böylece daha etkili ve hedef odaklı içerik stratejileri geliştirilebilir. Daha detaylı olarak açıklamak gerekirse;

5.1. Pazarlama ve Hedef Kitle Stratejileri:

MPAA derecelendirmelerine göre dağılım ve Amazon Prime Video'nun içerik stratejisi, pazarlamacıların ve içerik üreticilerinin hedef kitlelerine daha iyi hitap etmeleri için yol gösterici olabilir. Hangi türdeki içeriklerin daha popüler olduğu veya hangi yaş gruplarının daha çok hedef alındığı bu analizlerle anlaşılabilir.

5.2. Kalite ve Popülerlik İlişkisi:

Oy sayısı ve film derecelendirmesi arasındaki ilişki, popülerliğin kaliteye etkisini gösterir. Bu, filmlerin tanıtım ve dağıtım stratejilerinin planlanmasında kullanılabilir.

5.3.Fiyatlandırma Stratejileri:

Film fiyatlarının kutu grafiği, sektördeki fiyatlandırma trendlerini ve müşteri beklentilerini yansıtır. Bu bilgiler, fiyat belirleme ve rekabet stratejilerinde kullanılabilir.

5.4. Sektörel Trendler ve Teknolojik Gelişmeler:

Yıllara göre film çıkış sayıları, sektördeki büyüme eğilimlerini ve teknolojik gelişmelerin etkilerini gösterir. Dijital dağıtım platformlarının yükselişi ve geleneksel sinema salonlarının değişen rolü gibi trendler bu analizle ortaya çıkabilir.

5.5. Ürün Geliştirme ve İçerik Üretimi:

Sayısal değişkenlerin çift grafiği ve çeşitli grafiklerle yapılan EDA, içerik üreticilerine hangi tür filmlerin daha iyi performans gösterdiğini ve izleyicilerin ne tür içerikleri tercih ettiğini gösterir. Bu bilgiler, gelecekteki film projelerinin planlanmasında ve hedef kitlenin daha iyi anlaşılmasında önemli rol oynar.

6.Kaynakça

Amazon- Movies and Films January 23 from

https://www.kaggle.com/datasets/muhammadawaistayyab/amazon-movies-and-films/data

7.Ekler

Bu ek, çalışmada kullanılan Python kodlarını içermektedir. Kodlar, veri analizi ve veri görselleştirmesinin nasıl yapıldığını adım adım göstermektedir. Kodlar ayrıca, veri temizleme ve veri işleme süreçlerini de kapsamaktadır. Kullanılan kodların tamamına, aşağıdaki bağlantıdan erişilebilir:

https://drive.google.com/file/d/1VPwLLTIS2QwTCfL3oLn6TWTAW5mvvnzu/view?usp=sharing