

PROJECT

**IMDb Score Prediction
With Gradient Boosting
And Neural Network**



Develop a machine learning model to predict the IMDb Scores of the movies available on Films Based in their genre, premiere date, runtime and language . The model aims to accurately finds the popularity of the movies to the assist users in discovering highly rated films that align with their preferences

Design the project based on :

- * Data Source
- * Data Preprocessing
- * Feature Engineering
- * Model Selection
- * Model Training
- * Evaluation

Main algorithms & ML are :

linear Regression ,Random Forest Regression to Predict IMDb Scores .Train the selected model using preprocessing data Regression metrics like MAE ,MSE,R-squared,Gradient Boosting and Neural Network.

DATA SOURCE & DATA PERPROCESSING (used for find the missing values)

File Edit Selection View Go Run Terminal Help

IMDb-Scores

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib as plt
df=pd.read_csv("imdb.csv",encoding="unicode_escape")
```

df

	Title	Genre	Premiere	Runtime	IMDB Score	Language
0	Enter the Anime	Documentary	August 5, 2019	58	2.5	English/Japanese
1	Dark Forces	Thriller	August 21, 2020	81	2.6	Spanish
2	The App	Science fiction/Drama	December 26, 2019	79	2.6	Italian
3	The Open House	Horror thriller	January 19, 2019	94	3.2	English
4	Kaal Khuli	Mystery	October 30, 2020	90	3.4	Hindi
...
579	Taylor Swift: Reputation Stadium Tour	Concert Film	December 31, 2018	125	8.4	English
580	Winter on Fire: Ukraine's Fight for Freedom	Documentary	October 9, 2015	91	8.4	English/Ukrainian/Russian
581	Springsteen on Broadway	One-man show	December 16, 2018	153	8.5	English
582	Emicida: Amarilo - It's All For Yesterday	Documentary	December 8, 2020	89	8.6	Portuguese
583	David Attenborough: A Life on Our Planet	Documentary	October 4, 2020	83	9.0	English

584 rows x 6 columns

OUTLINE TIMELINE 26°C Rain showers

Cell 7 of 9 Go Live 14:33 27-09-2023

File Edit Selection View Go Run Terminal Help

IMDb-Scores

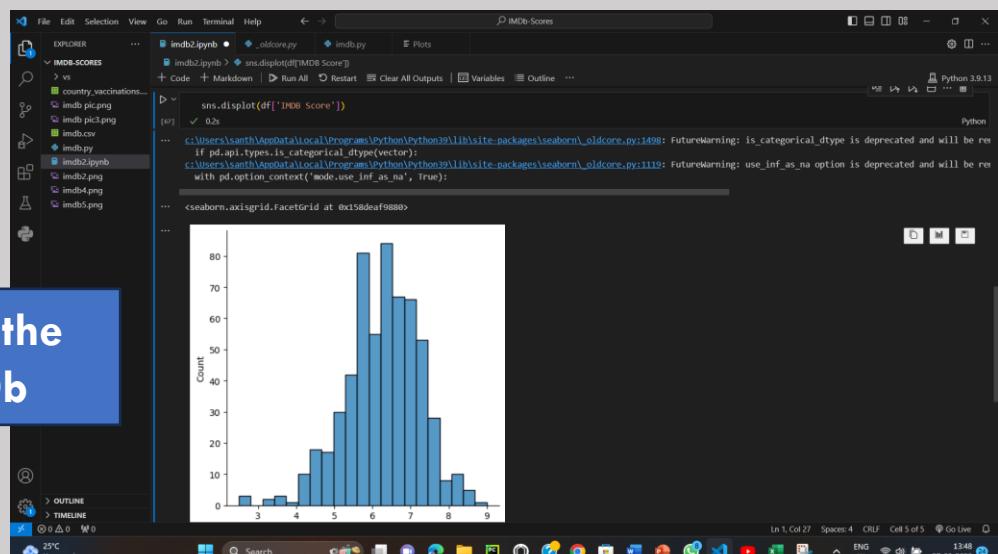
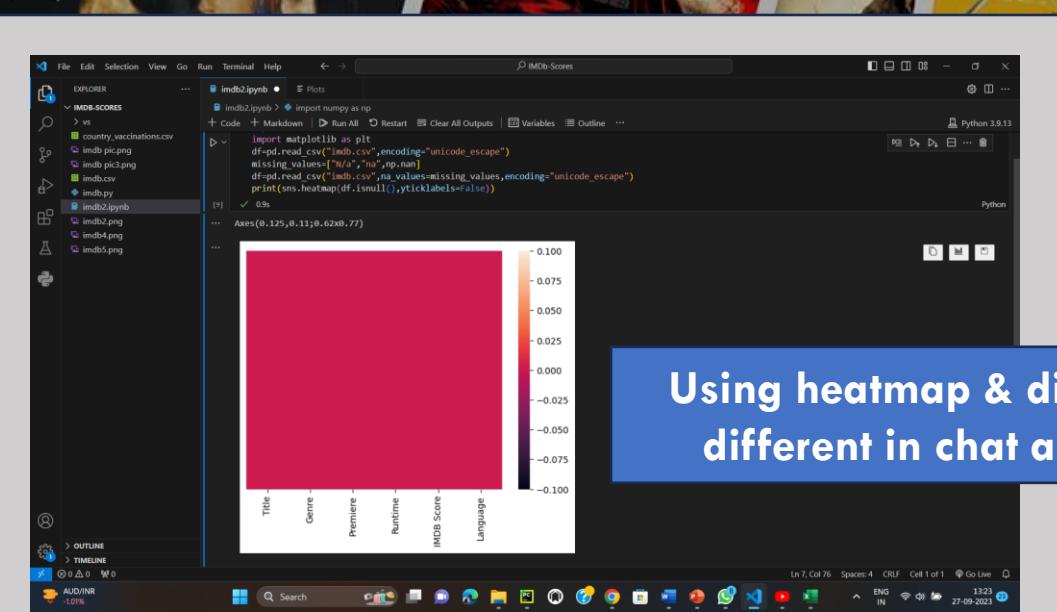
```
import pandas as pd
df=pd.read_csv("imdb.csv",encoding="unicode_escape")
print(df.isnull().sum())
```

Title 0
Genre 0
Premiere 0
Runtime 0
IMDB Score 0
Language 0
dtype: int64

TIMELINE GBP/INR -0.41%

Search Cell 1 of 1 Go Live 13:19 27-09-2023

Using heatmap & displot can find the different in chat and predict IMDb



NEURAL NETWORKING

The screenshot shows a Jupyter Notebook environment with the title bar "IMDb-Scores". The left sidebar displays a file tree with various Python files and configuration files. The main notebook area contains the following Python code:

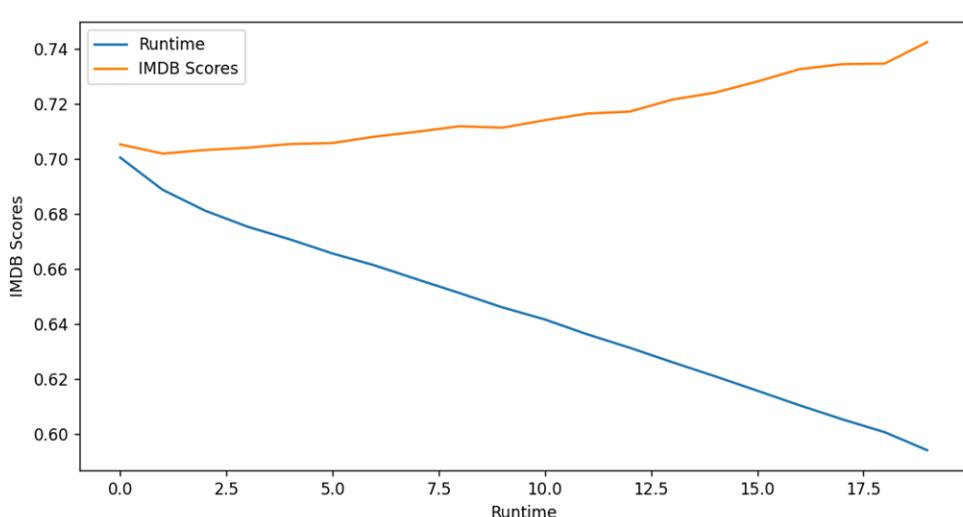
```
File Edit Selection View Go Run Terminal Help ← → ⌘ IMDB-Scores EXPLORER ... Untitled 1.ipynb test.py Keyboard Shortcuts test.py sample.py gradient boosting.py Activate.ps1 ...
```

```
IMDb_Scores.JIMDB SCORE PREDICTION.pdf at main - Praeview593_IMDb Scores_files > test1.py ...
```

```
8
9 # Generate some example data (replace with your own dataset)
10 np.random.seed(0)
11 X = np.random.randn(1000, 20) # 1000 samples with 20 features
12 y = np.random.randint(2, size=1000) # Binary labels (0 or 1)
13
14 # Split the data into training and testing sets
15 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
16
17 # Create a simple neural network model
18 model = keras.Sequential([
19     keras.layers.Dense(32, activation='relu', input_dim=20),
20     keras.layers.Dense(16, activation='relu'),
21     keras.layers.Dense(1, activation='sigmoid') # Binary classification, so using sigmoid activation
22 ])
23
24 # Compile the model
25 model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])
26
27 # Train the model
28 history = model.fit(X_train, y_train, epochs=20, batch_size=32, validation_data=(X_test, y_test))
29
30 # Evaluate the model on the test data
31 test_loss, test_accuracy = model.evaluate(X_test, y_test)
32 print("Test Loss: [test_loss]")
33 print("Test Accuracy: [test_accuracy]")
34
35 # Plot training and validation loss
36 plt.figure(figsize=(10, 5))
37 plt.plot(history.history['loss'], label='Runtime')
38 plt.plot(history.history['val_loss'], label='IMDb Scores')
39 plt.xlabel('Runtime')
40 plt.ylabel('IMDb Scores')
41 plt.legend()
42 plt.show()
43
44 # Make predictions on the test data
```

- **Hyperparameter Tuning:** Proper tuning of hyperparameters is crucial for optimal performance.
 - **Potential for Overfitting:** Care must be taken to avoid overfitting, especially if the weak learners are too complex.

- Neural Networking is an ensemble learning technique that combines the predictions of several weak learners to create a strong predictive model.
 - Neural Networking, with its iterative learning approach, stands as a powerful tool for predictive modeling, providing accurate results across diverse datasets.



Key Components For Gradient Boosting

- **Weak Learners:** Typically shallow decision trees are used.
- **Boosting:** Models are built sequentially, and each subsequent model corrects errors made by the previous ones.

Use Cases For Gradient Boosting

- **Commonly Applied:** Used in various domains, including finance, healthcare, and Kaggle competitions.



Key Components For Neural Network

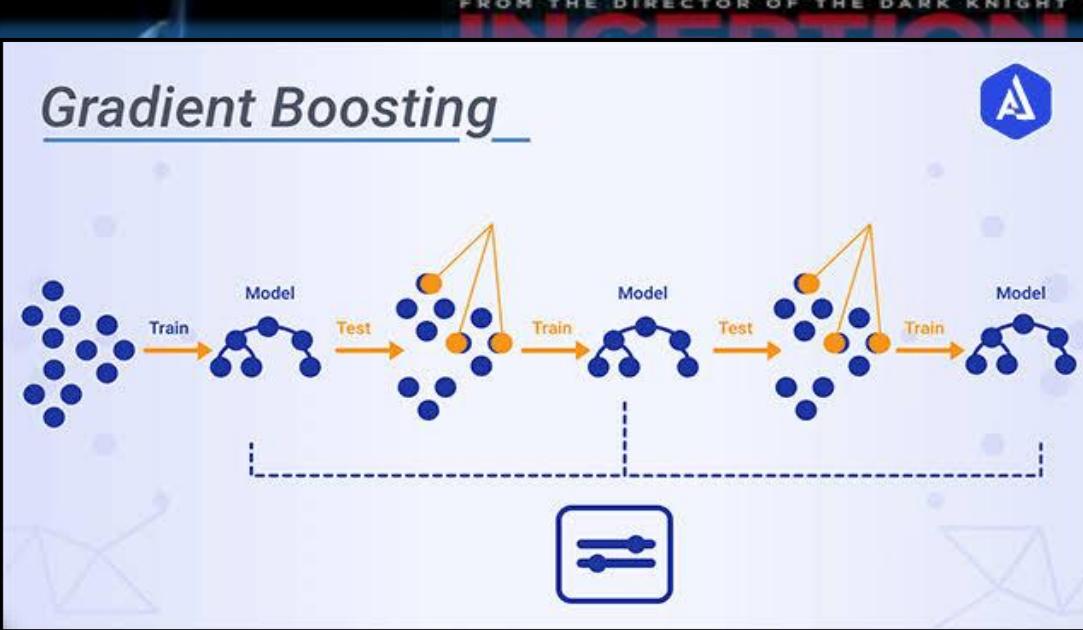
- **Neurons:** Basic computational units that process information.
- **Layers:** Organized in input, hidden, and output layers.
- **Activation Functions:** Determine the strength of connections between neurons.

Use Cases For Gradient Boosting

- **Image and Speech Recognition:** Neural Networks excel in tasks like image and speech recognition.

GRADIENT BOOSTING RISES

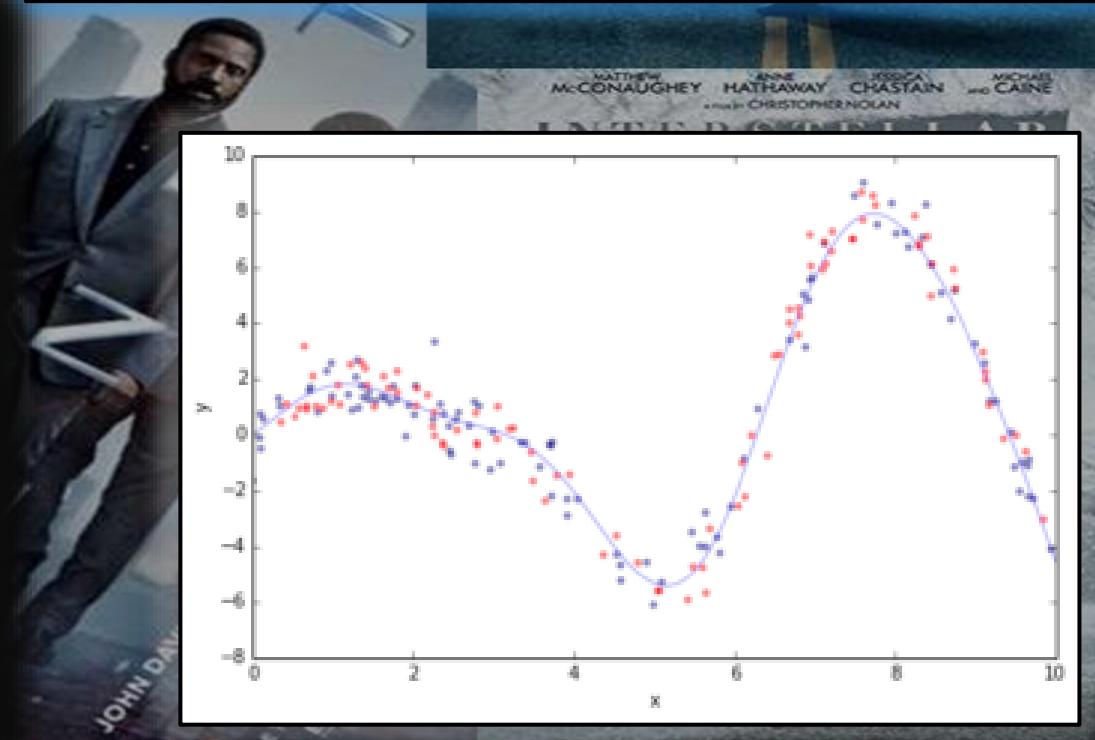
- Gradient Boosting is an ensemble learning technique that combines the predictions of several weak learners to create a strong predictive model.
 - Gradient Boosting, with its iterative learning approach, stands as a powerful tool for predictive modeling, providing accurate results across diverse datasets.



The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a back/forward navigation bar. The title bar says "IMDb-Scores". The left sidebar (Explorer) lists files and folders related to an "IMDb-Scores" project, such as "IMDb-Scores.ipynb", "gradient boosting.py", and various image files like "100.png" and "102.png". The main editor area contains the following Python code:

```
gradient boosting.py > ...
1 import pandas as pd
2 from sklearn.model_selection import train_test_split
3 from sklearn.ensemble import GradientBoostingClassifier
4 from sklearn.metrics import accuracy_score
5
6 # Load your CSV file into a Pandas DataFrame
7 data = pd.read_csv('imdb.csv')
8
9 # Assuming the last column is your target variable and the rest are features
10 X = data.iloc[:, :-1]
11 y = data.iloc[:, -1]
12
13 # Split the data into training and testing sets
14 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
15
16 # Initialize the Gradient Boosting classifier
17 gb_classifier = GradientBoostingClassifier(n_estimators=100, learning_rate=0.1, max_depth=3, random_state=42)
18
19 # Train the classifier
20 gb_classifier.fit(X_train, y_train)
21
22 # Make predictions on the test set
23 predictions = gb_classifier.predict(X_test)
24
25 # Evaluate the accuracy
26 accuracy = accuracy_score(y_test, predictions)
27 print(f'Accuracy: {accuracy}')
```

The status bar at the bottom shows "Ln 21, Col 1" and "Spaces: 4" along with icons for terminal, file, search, and other tools. The bottom right corner displays the date and time as "11/10/2023 14:49".



**THANK
YOU**

Selected sources /

WA_Fn-UseC_-Telco-Cus... + :

WA_Fn-UseC_-Telco-Cus... + :

Search

Navigation paths +

- WA_Fn-UserC_-Telco-Cus... +
- customerID
- gender
- SeniorCitizen
- Partner
- Dependents
- tenure
- PhoneService
- MultipleLines
- InternetService
- OnlineSecurity
- OnlineBackup
- DeviceProtection
- TechSupport

Explore data relationships

WA_Fn-UserC_-Telco-Customer-Churn.csv

Reset to original

Partner

Edit diagram ▾

Relationship diagram ⓘ

```

graph TD
    Partner((Partner)) --- Contract
    Partner --- DeviceProtection
    Partner --- Dependents
    Partner --- OnlineBackup
    Partner --- OnlineSecurity
    Partner --- PaymentMethod
    Partner --- TotalCharges
    Partner --- tenure
    Partner --- Churn
    
```

tenure by Churn colored by Partner

Add +

tenure and MonthlyCharges by Partner

Add +

See more

Explore data relationships

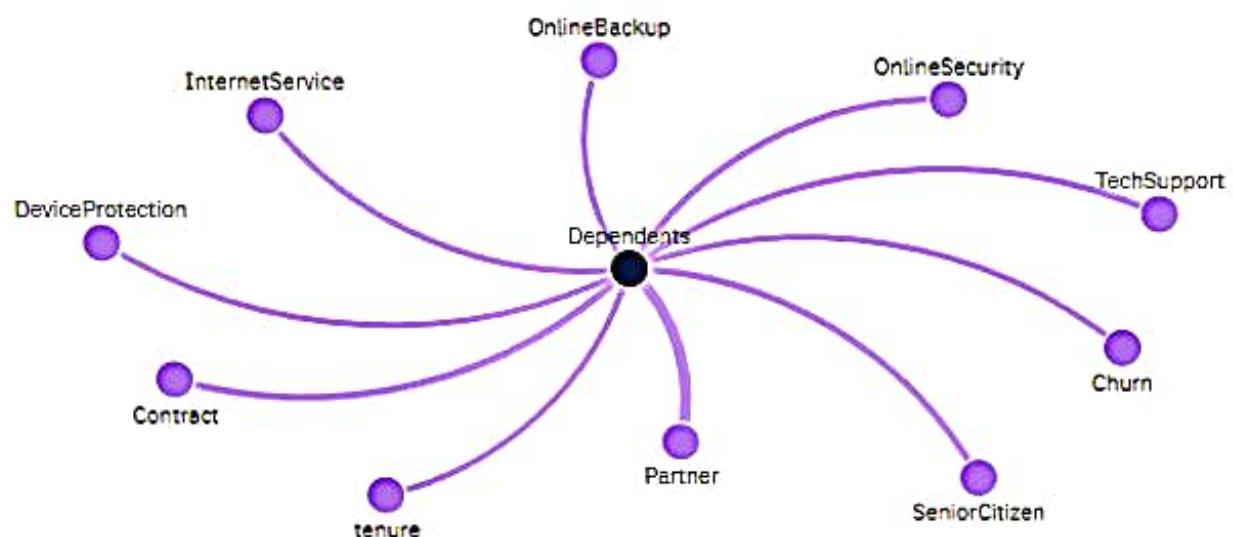
WA_Fn-UseC_Telco-Customer-Churn.csv

[Reset to original](#)

Dependents



[Edit diagram](#) ▾



Relationship diagram ①

10% 100%

Analytics Details Fields Properties

Cards

1 **tenure by Churn... by customerID**

2 **MonthlyCharges...red by gender**

3 **tenure and Mon...ges by Partner**

4 **Data relationships**

tenure by Churn colored by customerID

customerID

9993-HOTOH	9993-LHIEB	9974-JFBHQ	9972-EWR2S	9968-FFVWH	9964-WBQDJ
9969-WOKFT	9958-MEKUC	9966-QCPOY	9953-ZMKSM	9943-VSZUV	9938-TXDGL
9926-RHDQ	9924-JPRMC	9919-FZDED	9906-NHHVC	9896-UYMIE	9885-ZCUMM
9880-TDQAC	9866-OCCKE	9861-POSZP	9848-3QJTX	9844-FELAJ	9838-BFCQT
9835-ZIITK	9823-EALYC	9821-EESNZ	9803-FTCG	9802-CAQUT	9800-OULGR
9795-SHUHB	9795-NREXC	9788-HNGUT	9786-YNNHU	9778-OGKQZ	9777-1OHWP
9776-QUZI	9769-TSBZE	9742-XOKTS	9739-JLPQJ	9716-WZCLW	9680-NIAUV

Details

Over all values of **Churn** and **customerID**, the sum of **tenure** is nearly 87 thousand.

The summed values of **tenure** range from 63 to 72.

For **tenure**, the most significant values of **customerID** are 8809-XKHMD, 8204-YJCLA, 2274-XUATA, 0244-LGNFY, and 9919-FZDED, whose respective **tenure** values add up to 360, or 0.4 % of the total.

For **tenure**, the most significant value of **Churn** is No, whose respective **tenure** values add up over 81 thousand, or 93.8 % of the total.

For **tenure**, the most significant values of **customerID** are 8809-XKHMD, 8204-YJCLA, 2274-XUATA, 0244-LGNFY, and 9919-FZDED, whose respective **tenure** values add up to 360, or 0.4 % of the total.

Analytics Details Fields Properties

Cards

- 1 tenure by Churn... by customerID
- 2 MonthlyCharges...red by gender
- 3 tenure and Mon...ges by Partner
- 4 Data relationships

MonthlyCharges by OnlineSecurity colored by gender

gender
● Female ● Male

OnlineSecurity	Female (MonthlyCharges)	Male (MonthlyCharges)
No	~130,000	~135,000
No internet service	~18,000	~18,000
Yes	~85,000	~80,000

Details

MonthlyCharges is unusually low when **OnlineSecurity** is No internet service.

Across all values of **OnlineSecurity** and **gender**, the sum of **MonthlyCharges** is over 456 thousand.

The summed values of **MonthlyCharges** range from nearly 16 thousand to almost 135 thousand.

MonthlyCharges is unusually low when the combinations of **OnlineSecurity** and **gender** are No internet service and Female and No internet service and Male.

For **MonthlyCharges**, the most significant values of **OnlineSecurity** are No and Yes, whose respective **MonthlyCharges** values add up to nearly 424 thousand, or 92.9 % of the total.

For **MonthlyCharges**, the most significant value of **gender** is Male, whose respective

