

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 1433 27-09-2023

File Edit Selection View Go Run Terminal Help

IMDb-Scores

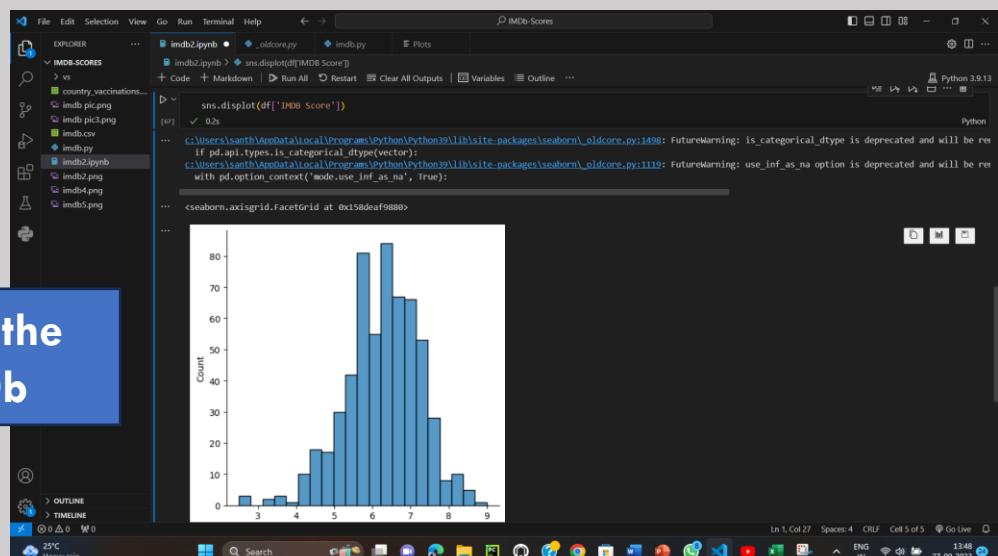
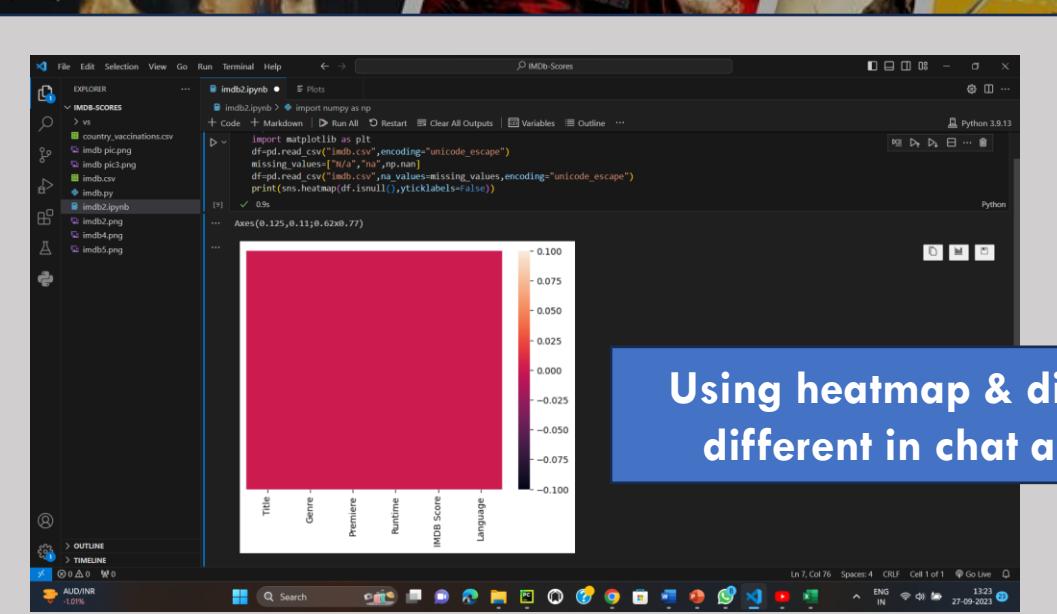
```
import pandas as pd
df=pd.read_csv("imdb.csv",encoding="unicode_escape")
print(df.isnull().sum())
```

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 Ln 3, Col 25 Spaces: 4 CRLF Cell 1 of 1 1319 27-09-2023

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



NEURAL NETWORKING

The screenshot shows a Jupyter Notebook interface with the title bar "IMDb-Scores". The left sidebar displays a tree view of files and a "OUTLINE" section. The main area contains Python code for training and evaluating a neural network model on IMDb movie reviews. The code includes imports, data generation, splitting the dataset, creating a sequential model with three layers, compiling it with Adam optimizer and binary crossentropy loss, training the model, evaluating it on test data, plotting training and validation loss, and finally making predictions on the test data.

```
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.JMDB SCORE PREDICTION.pdf at main - Praeview593_IMDB_Scores_Files > test1.py ...
```

```
diffs-189dcf77e7a4.js.download
element-registry-e9b061451a1e.js.download
environment-509b58e05bf1.js.download
github-a3fc6af5a97.css
github-elements-e7eb7ec8c502.js.download
global-d7555a777bd9.css
light-a09c8f73428.css
notifications-global-f5767007bcf.js.download
optimizable-b8ae60018b11.js.download
primer-047ee6293tcd.css
primer-primitives-e143cb97ed1.css
react-code-view-b23004b3649d.js.download
react-lib-2104b5934c3.js.download
repositories-e0894816616.js.download
test.py
test1.py

topic-suggestions-e57c71e486d0.js.download
ui_packages_react_core_create_browser_history_ts-ui_packages...
ui_packages_react_core_register_app_ts-4c8aa7d9158e.js.download
ui_packages_ref-selector_RefSelector_ts-9cbaf85c1995.js.download
ui_packages_soft_nav_soft-nav_ts-df17d5597d8f8.js.download
vendors-node_modules_alex_crc32_lib_crc32_esm_js-node...
vendors-node_modules_color-convert_index_js-35bae68c408...
vendors-node_modules_delegated-events_dist_index_js-node...
vendors-node_modules_delegated-events_dist_index_js-node...
vendors-node_modules_dominly_dist_purify_js-64d590970fa...
vendors-node_modules_github_clipboard-copy-element_dist...
vendors-node_modules_github_file-attachment-element_dist...
vendors-node_modules_github_file-attachment-element_dist...

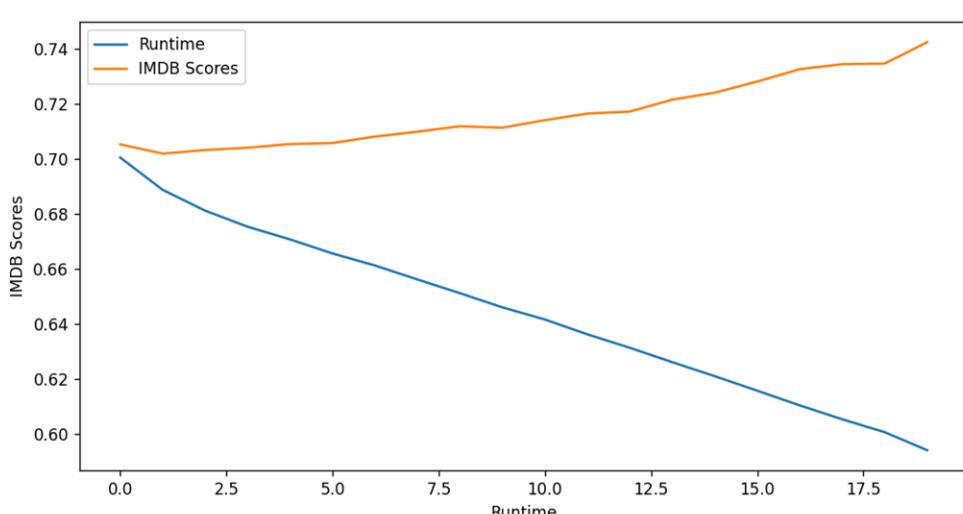
```

```
> OUTLINE
> TIMELINE
X 32°C Mostly sunny
```

```
Ln 13, Col 1 Spaces: 4 UTF-8 CRLF Python 3.11.5 ('myenv': venv) Go Live
```

- **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

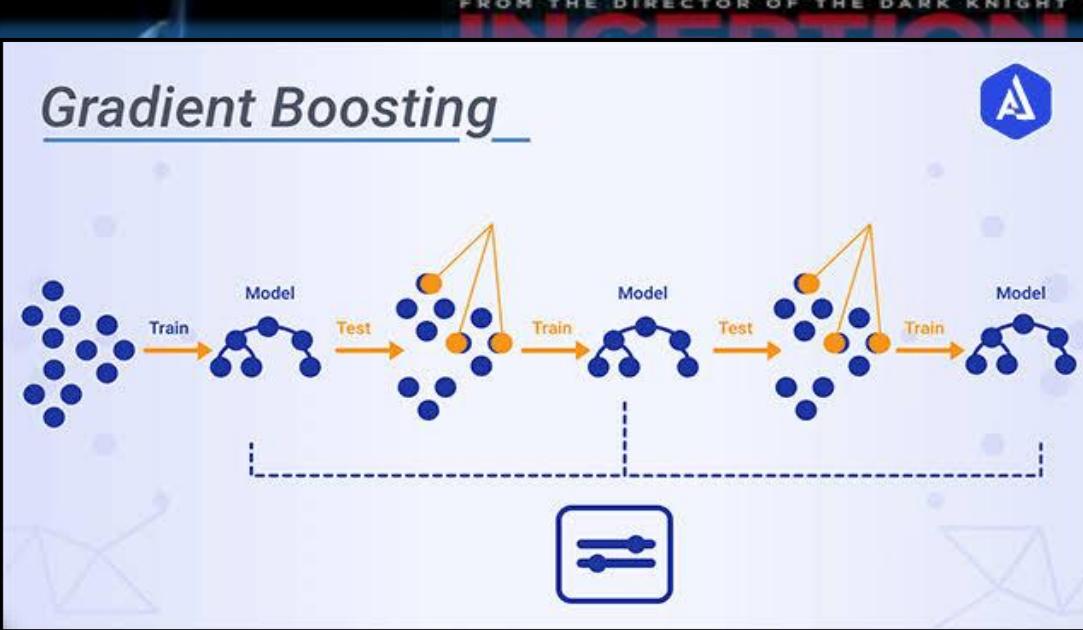
- **Neurons:** Basic computational units that process information.
- **Layers:** Organized in input, hidden, and output layers.
- **Activation Functions:** Determine the strength of connections between neurons.

Use Cases For Gradient Boosting

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

GRADIENT BOOSTING RISES

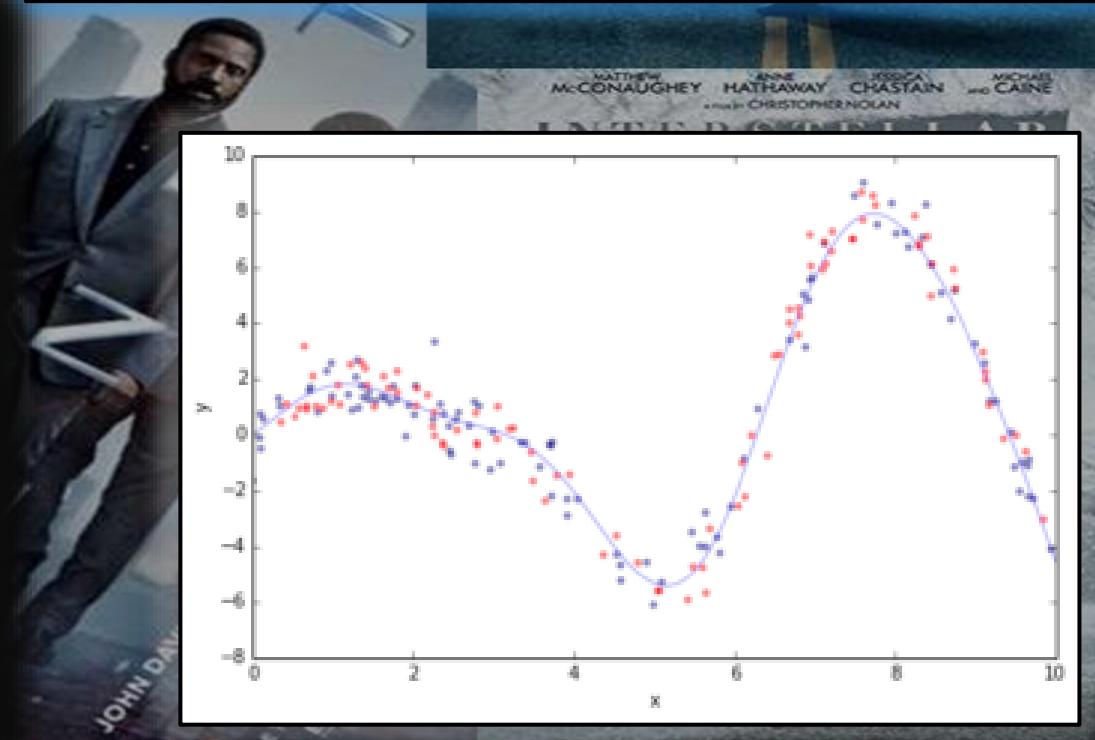
- Gradient Boosting is an ensemble learning technique that combines the predictions of several weak learners to create a strong predictive model.
 - Gradient Boosting, with its iterative learning approach, stands as a powerful tool for predictive modeling, providing accurate results across diverse datasets.



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 set of icons for file operations like Open, Save, and Close. The title bar says "IMDb-Scores". The left sidebar (Explorer) lists files and folders under "IMDB-SCORES", including "gradient boosting.py", "test1.py", "Keyboard Shortcuts", "test.py", "sample.py", and "Activate.ps1". Below this, there are icons for search, file operations, and navigation. 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 other system information like temperature and battery level.



**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

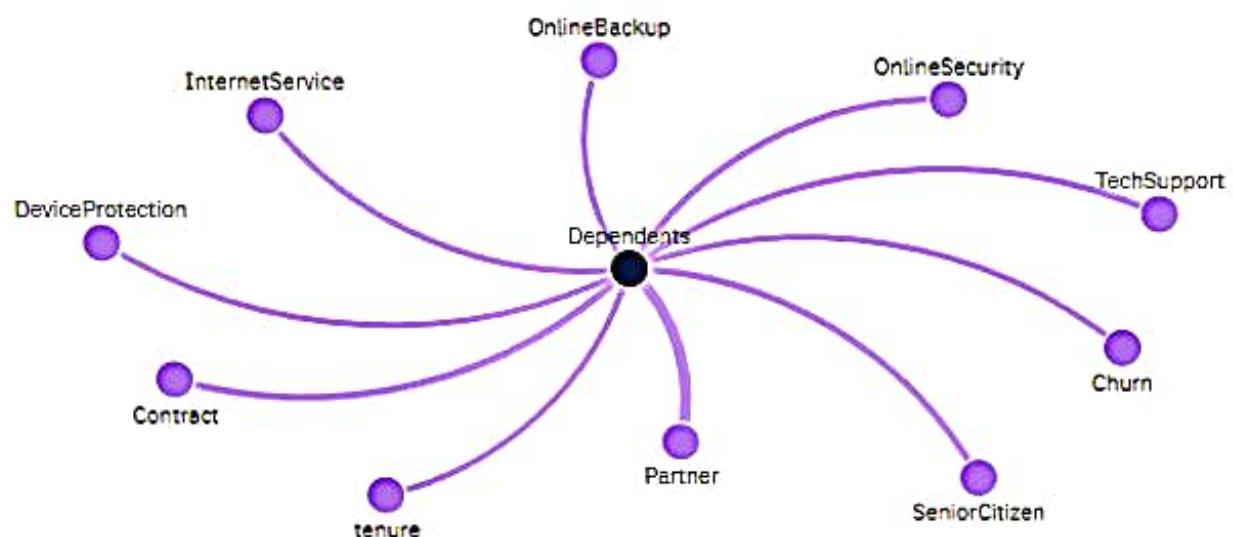
WA_Fn-UseC_Telco-Customer-Churn.csv

[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-OUIGR |
| 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

