Project Title	**Netflix Movies and TV Shows Clustering**
Skills take away From This Project	• **Data Cleaning and Preprocessing** • **Feature Engineering** • **Unsupervised Machine Learning (Clustering)** • **K-Means, Hierarchical, and DBSCAN Clustering** • **Data Visualization (Matplotlib, Seaborn)** • **Principal Component Analysis (PCA) and t-SNE for Dimensionality Reduction** • **Model Evaluation using Silhouette Score and Davies-Bouldin Index** • **Python, Pandas, NumPy, Scikit-Learn**
Domain	**Entertainment, Data Science**

Problem Statement:

The project focuses on clustering Netflix movies and TV shows based on various features like genre, rating, and duration. The goal is to use unsupervised machine learning techniques to identify similar content groups, which can help users discover content based on preferences.

Business Use Cases:

1. Personalized content recommendations for Netflix users based on clustering.
2. Identifying niche content categories to enhance Netflix's recommendation algorithm.
3. Understanding market trends and clustering content for better targeting of advertisements.

4. Assisting production houses in understanding content gaps and demand patterns.

Approach:

1. **Data Collection & Exploration:**
   - Load and inspect the dataset to understand its structure and contents.
   - Identify missing values, duplicate records, and inconsistencies.
   - Perform Exploratory Data Analysis (EDA) to visualize trends and distributions.
2. **Data Preprocessing:**
   - Handle missing values in key columns like `director`, `cast`, and `country` by using imputation strategies or removing irrelevant data.
   - Convert categorical data (`type`, `rating`, `listed_in`) into numerical format using one-hot encoding or label encoding.
   - Standardize numerical features such as `duration` and `release_year` to ensure uniform scaling.
   - Extract relevant features from text columns like `listed_in` and `description` using Natural Language Processing (NLP) techniques such as TF-IDF vectorization if necessary.
3. **Feature Engineering:**
   - Create new meaningful features, such as:
     - Content age: `current_year - release_year`.
     - Genre count: Number of genres associated with each content.
   - Transform categorical variables into numerical representations suitable for clustering algorithms.
4. **Clustering Model Selection:**
   - Choose appropriate clustering techniques such as:
     - **K-Means Clustering:** Suitable for numerical data; requires selecting the optimal number of clusters using the Elbow Method or Silhouette Score.
     - **Hierarchical Clustering:** Provides a tree-like structure to understand relationships between data points.
     - **DBSCAN:** A density-based approach that can help identify noise and anomalies.
   - Experiment with different numbers of clusters and evaluate their performance.
5. **Model Training & Optimization:**
   - Apply the chosen clustering algorithm and fine-tune hyperparameters.
   - Evaluate different distance metrics and linkage criteria (for hierarchical clustering).

    - Use dimensionality reduction techniques like Principal Component Analysis (PCA) or t-SNE to visualize clusters in 2D or 3D.
6. **Visualization & Interpretation:**
    - Generate cluster plots to analyze content similarities.
    - Create heatmaps to show correlations between features and clusters.
    - Present insights derived from clusters, such as the most common genres per group or the distribution of content ratings.
7. **Evaluation & Refinement:**
    - Use metrics such as Silhouette Score, Davies-Bouldin Index, and Inertia to validate clustering effectiveness.
    - Adjust features and preprocessing steps based on evaluation results.
    - Compare different clustering approaches to determine the best model for Netflix content categorization.

Results:

- Successfully clustered Netflix movies and TV shows based on genre, rating, and other attributes.
- Insights into content groupings, allowing for better recommendation strategies.
- Visual representation of clusters to understand content distribution and similarity.

Project Evaluation metrics:

- Silhouette Score for cluster quality assessment.
- Inertia (for K-Means) to determine the optimal number of clusters.
- Davies-Bouldin Index for cluster separation measurement.
- Visualization techniques (e.g., t-SNE, PCA) to validate cluster formations.

Technical Tags:

- Python, Pandas, NumPy, Scikit-Learn
- Machine Learning, Unsupervised Learning
- K-Means, Hierarchical Clustering
- Data Preprocessing, Feature Engineering
- Data Visualization (Matplotlib, Seaborn)

Data Set:

- **Netflix Movies and TV Shows dataset** (7787 entries, 12 columns)
- Format: CSV
- Contains metadata about movies and TV shows available on Netflix
- Dataset Link: NETFLIX MOVIES AND TV SHOWS CLUSTERING

Data Set Explanation:

- `show_id`: **Unique identifier for each title**
- `type`: **Indicates whether the entry is a "Movie" or "TV Show"**
- `title`: **Name of the movie/TV show**
- `director`: **Name of the director(s) (contains missing values)**
- `cast`: **Main actors in the content (contains missing values)**
- `country`: **Country where the content was produced (contains missing values)**
- `date_added`: **Date when the content was added to Netflix**
- `release_year`: **Year the content was released**
- `rating`: **Content rating (e.g., TV-MA, PG-13)**
- `duration`: **Movie runtime or number of TV show seasons**
- `listed_in`: **Genre classification (e.g., Drama, Comedy, Horror)**
- `description`: **Short synopsis of the content**

Project Deliverables:

- Source code for clustering implementation (Python Notebook).
- Processed dataset with cleaned and transformed features.
- Visualizations showcasing clustering results.
- Final project report documenting the approach, results, and insights

Project Guidelines:

- Follow proper data preprocessing and cleaning practices.
- Ensure the clustering approach is justified with metrics.
- Maintain modular code structure for readability.
- Use version control (Git) for code tracking and collaboration.
- Document all steps with clear explanations and justifications.

Reference

Project Live Evaluation	**Project Live Evaluation**

EDA Guide	Exploratory Data Analysis (EDA) G…
Capstone Explanation Guideline	Capstone Explanation Guideline
GitHub Reference	How to Use GitHub.pptx

Timeline:

The project must be completed and submitted within **7 days from the assigned date.**

PROJECT DOUBT CLARIFICATION SESSION (PROJECT AND CLASS DOUBTS)

About Session: The Project Doubt Clarification Session is a helpful resource for resolving questions and concerns about projects and class topics. It provides support in understanding project requirements, addressing code issues, and clarifying class concepts. The session aims to enhance comprehension and provide guidance to overcome challenges effectively.
Note: Book the slot at least before 12:00 Pm on the same day

Timing: Monday-Saturday (4:00PM to 5:00PM)

Booking link :https://forms.gle/XC553oSbMJ2Gcfug9

LIVE EVALUATION SESSION (CAPSTONE AND FINAL PROJECT)

About Session: The Live Evaluation Session for Capstone and Final Projects allows participants to showcase their projects and receive real-time feedback for

improvement. It assesses project quality and provides an opportunity for discussion and evaluation.

Note: This form will Open only on Saturday (after 2 PM) and Sunday on Every Week

Timing: Monday-Saturday (05:30PM to 07:00PM)

Booking link : https://forms.gle/1m2Gsro41fLtZurRA

Created By:	Verified By:	Approved By:
Shadiya P P		