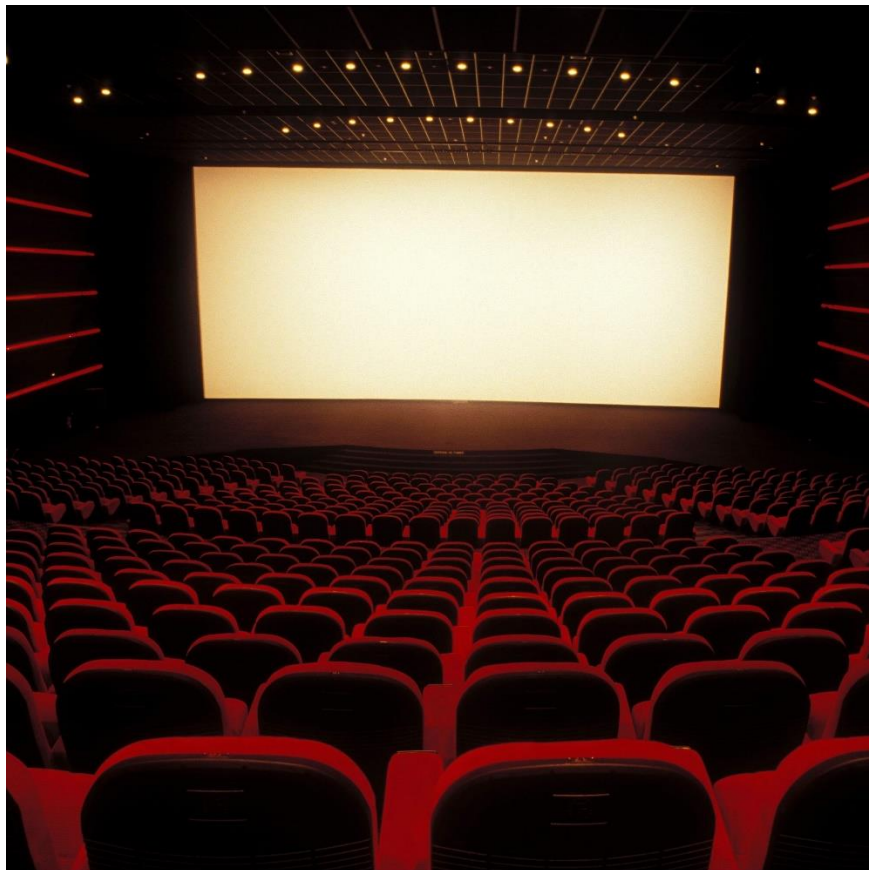


Le Boulch Quentin

Master EE

Python Project

Top 25 movies (by production cost) analysis



Présentation du sujet :

L'industrie cinématographique est l'une des plus influentes et rentables au monde, produisant chaque année des centaines de films dans différents genres et budgets. Dans ce projet, nous allons plonger dans l'univers du cinéma en utilisant Python pour analyser un jeu de données sur les films. Ce jeu de données contient des informations essentielles sur divers films, notamment leur titre, leur coût de production, leurs recettes nationales (US) et mondiales leur genre, le nombre de salles de cinéma, et l'année de sortie.

Le jeu de données sur lequel nous allons travailler regroupe le top 25 des films classé par leur budget de production. (24 j'ai supprimé pour vous un film où il y avait des données manquantes)

Voici sous quelle forme est représenté notre jeu de données :

rank	title	production_cost	domestic_gross	worldwide_gross	genre	year
------	-------	-----------------	----------------	-----------------	-------	------

rank = classement en fonction du budget de production

title = titre du film (pour soucis de présentation je mettrai les noms de chaque film complet dans une annexe)

production_cost = coût de production en USD

domestic_gross = Recettes nationales sur la vente de ticket (USA) en USD

worldwide_gross = recettes mondial sur la vente de ticket

genre = type de film ou catégorie

years = année de sortie du film

Le but de ce projet d'explorer ce jeu de données, regarder quels films ont été les plus rentables et au contraire ceux qui l'ont été un peu moins.

ouvrir le fichier .txt à disposition

Vous pouvez vous servir du fichier csv si vous le souhaitez.

Questions :

1. Créez une liste contenant uniquement les titres des films présents dans le tableau de données et affichez celle-ci dans un fichier **list_movies**
2. Créez un dictionnaire en utilisant la colonne 'Rank' comme clé et la colonne 'Title' comme valeur à partir des données fournies. Vous pourrez l'utiliser par la suite si vous le souhaitez

3.1 *Affichez les résultats dans un fichiers « result.txt » avec une phrase. *

Définissez une fonction appelée **average_cost** qui prend en entrée une liste de coûts de production (production_cost) et renvoie la moyenne de ces coûts.

3.2 Faire de même pour average_domestic_gross et average_worldwide_gross.

3.3 Faire une fonction max_cost et max_gross qui affiche le film qui a coûté le plus cher à produire et celui qui a eu le plus de recettes. Faire la même chose avec min_cost et max_cost.

4.1 Top N films les plus rentables :

Créez une fonction (vous la nommerez **top_profitable_films**) qui permet à l'utilisateur de spécifier un nombre (N) et qui renvoie les N films les plus rentables en fonction du bénéfice brut. Affichez ces films dans un tableau ou un fichier.

Par exemple l'utilisateur demande entre N=5 et la fonction renvoie les 5 films les plus rentables (avec la plus grande valeur de **worldwide_gross**)

4.2 Faire la même chose avec les Top N films les plus rentables aux USA (domestic_gross)

5.1 Regroupez les films par leur année de sortie

5.2 Comptez combien il y en a pour chaque année à l'aide d'une fonction movie_per_year

5.3 Afficher le résultat de l'année qui compte le plus de film produit dans ce top 24 dans le fichier « result.txt » avec une phrase.

6.1 Faire le même travail de groupement et de comptage en regroupant les films par genre.

Affichez le genre le plus représenté dans le fichier « result.txt »

6.2 Calculez la moyenne des coûts de production, des recettes nationales et des recettes mondiales pour chaque genre de film. Affichez ces moyennes dans un tableau ou un graphique pour comparer les performances des genres.

7. Calculez pour chaque film, lequel a le rendement/ratio ou multiplicateur le plus élevé entre **worldwide_gross** et **production_cost**. Par exemple Avengers a fait une recette mondiale sur les tickets de 2 797 800 564 \$ pour un cout de production de 400 000 000 \$. Avengers : End Game à multiplier par 6,99450141 ~ 7 ses couts de production pour avoir un worldwide_gross de près de 2.8 Milliard \$

Et classez-les par ordre descendant dans un tableau, liste ou directement dans la console avec une phrase. Vous pourrez aussi afficher le résultat dans le fichier « result.txt »

ANNEXE :

Noms de tous les films en détail (vous pouvez vous arrangez dans le fichier movies.txt pour remplacez leur vrai nom par des initiales ou autre si ça vous arrange)

Title

Avengers: Endgame

Pirates of the Caribbean: On Stranger Tides

Avengers: Age of Ultron

Star Wars Ep. VII: The Force Awakens

Avengers: Infinity War

Pirates of the Caribbean: At World's End

Justice League

Spectre

~~Mission: Impossible - Dead Reckoning Part One~~

Star Wars: The Rise of Skywalker

Solo: A Star Wars Story

John Carter

Batman v Superman: Dawn of Justice

Star Wars Ep. VIII: The Last Jedi

The Lion King

Tangled

Spider-Man 3

Captain America: Civil War

Thor: Love and Thunder

Harry Potter and the Half-Blood Prince

The Hobbit: The Desolation of Smaug

The Hobbit: The Battle of the Five Armies

The Fate of the Furious

No Time to Die

Avatar

On enlève Mission impossible du dataframe car il y a des données manquantes

Le Boulch Quentin

Master EE

Python Project

Top 25 movies (by production cost) analysis



Presentation of the topic:

The film industry is one of the most influential and profitable industries in the world, producing hundreds of films in various genres and budgets each year. In this project, we will delve into the world of cinema using Python to analyze a small dataset of films. This dataset contains essential information about various movies, including their title, production cost, domestic gross, worldwide gross, genre, the number of movie theaters, and the release year.

The dataset we will work with compiles the top 25 movies ranked by their production budget. (I have removed one film with missing data for you.) Here is how our dataset is represented:

rank	title	production_cost	domestic_gross	worldwide_gross	genre	year
------	-------	-----------------	----------------	-----------------	-------	------

rank = ranking based on production budget

title = movie title (for presentation purposes, I will include the full names of each film in an appendix)

production_cost = production cost in USD

domestic_gross = domestic box office gross (USA) in USD

worldwide_gross = worldwide box office gross

genre = film type or category

years = film release year

The goal of this project is to explore this dataset, identify the most profitable films, and those that were less so.

**open the available .txt file* with all the data*

Questions:

1. Create a list containing only the titles of the films present in the data table and display it in a file named **"list_movies"**
2. Create a dictionary using the 'Rank' column as the key and the 'Title' column as the value from the provided data. You can use this dictionary later if you wish.

3.1 Display the results in a "result.txt" file with a sentence.

Define a function called **"average_cost"** that takes a list of production costs (production_cost) as input and returns the average of these costs.

3.2 Do the same for "average_domestic_gross" and "average_worldwide_gross."

3.3 Create **"max_cost"** and **"max_gross"** functions that display the film that had the highest production cost and the one with the highest gross revenue. Do the same with **"min_cost"** and **"min_gross."**

4.1 Top N most profitable films: Create a function (you will name it **"top_profitable_films"**) that allows the user to specify a number (N) and returns the N most profitable films based on worldwide gross revenue. Display these films in a table or a file. For example, if the user requests N=5, the function should return the 5 most profitable films (with the highest **worldwide_gross** value).

4.2 Do the same for the Top N most profitable films in the USA (**domestic_gross**).

5.1 Group the films by their release year.

5.2 Count how many films there are for each year using a **"movie_per_year"** function.

5.3 Display the result for the year with the most films produced in this top 24 in the "result.txt" file with a sentence.

6.1 Do the same grouping and counting work by grouping the films by genre.

Display the most represented genre in the "result.txt" file.

6.2 Calculate the average production costs, domestic gross, and worldwide gross for each film genre. Display these averages in a table or graph to compare genre performance.

7. Calculate, for each film, which one has the highest return/multiplier ratio between worldwide gross and production cost. For example, if "Avengers" had a worldwide ticket sale of \$2,797,800,564 with a production cost of \$400,000,000, its multiplier would be approximately $6.99450141 \sim 7$. Sort them in descending order in a table, list, or directly in the console with a sentence. You can also display the result in the "result.txt" file.

APPENDIX:

Names of all the films in detail (you can modify the movies.txt file to replace their real names with initials or other if it suits you).

title

Avengers: Endgame

POC: On Stranger Tides

Avengers: Age of Ultron

Star Wars Ep. VII

Avengers: Infinity War

POC: At World's End

Justice League

Spectre

Star Wars: The Rise of Skywalker

Solo: A Star Wars Story

John Carter

Batman v Superman: Dawn of Justice

Star Wars Ep. VIII:

The Lion King

Tangled

Spider-Man 3

Captain America: Civil War

Thor: Love and Thunder

Harry Potter and the Half-Blood Prince

The Hobbit: The Desolation of Smaug

The Hobbit: The Battle of the Five

Armies

The Fate of the Furious

No Time to Die

Avatar

(POC = Pirates the Caribbean)