Final Project-1

IMDB Movie Analysis

Please note that the sheets in the .xlsx file are named according to the solution of the task that sheet contains. Here is the link to the excel file: <(paste your excel sheet link)>

# Project Description:

The Final Project-1 is based on the IMDB movie review data collected over the years 1923 and 2016. This project is based on the concept of data analysis and pattern recognition. IMDB as we all know is one of the most trusted movie reviews which has been in the system for a long time now. Furthermore, in the given dataset the movies are compared using different fields which help us draw conclusions from the data and solve the questions asked.

# Approach:

The project started with framing a problem statement by ourselves using our analytical skills and then developing a solution for it, hence the approach is simple, learning the cause of the problem in the first place. It is quite obvious that as we dig deeper into the problem the difficulty to answer increases exponentially. Since our target is to find the root cause of the problem, this brute force approach where we need to understand the data completely suits the best. All the given tasks can be solved one after the other, understanding the problem statement.

# Tech Stack:

Software used: Microsoft Excel

Version: 2103 (16.0.13901.20400)

Developer: Microsoft

Latest Stable Release: April 13, 2021

Purpose: To perform data analysis.

Written in: C++(back-end)

# Task/Solution:

**Some of the resultant data is represented here; remaining data can be found in the .xlsx file in the same drive link as this .docx. Eg: Sheet ‘B’ of “Final Project-1.xlsx” represents Task B**

**A: Clean the data**

Sol: the following techniques can be used to clean the data

* Remove Duplicates: All the duplicate values or repeated records are removed
* Data Parsing from Text to Column: Compound words which are appended using a particular delimiter can be split into separate columns using this feature. In our data set attributes ‘plot\_keywords’ and ‘genre’ has different data elements separated using ‘|’ which can be split into different columns but it would increase the number of columns which makes it less readable, and also increases the number of blanks. Hence I chose not to.
* Delete All Formatting: A professional should have minimal data formatting and no conditional formatting to increase its durability. Hence avoid unnecessary formatting.
* Spell Check: Spell check is one other amazing built-in feature of Excel which I have utilized to correct errors in spellings.
* Change Case - Lower/Upper/Proper: Not used in our task but can be used if needed.
* Highlight Errors: The conditional formatting can be used to show specific exceptions or errors made in the data collected based on a particular condition.
* TRIM Function: Trim() built-in function can be used to remove unnecessary spaces between words.
* Find and Replace: Find and Replace was used to remove ‘Â’ that showed up automatically after each data element in the ‘movie\_title’ field when the dataset got downloaded.
* Handle Missing Data: Here since it is a large dataset, some of the records have more than two missing values ergo chose to remove all such records from our cleaned dataset.

**B: Movies with highest profit:**

Create a new column called profit which contains the difference of the two columns: gross and budget. Sort the column using the profit column as reference. Plot profit (y-axis) vs budget (x- axis) and observe the outliers using the appropriate chart type.  
Find the movies with the highest profit?

Sol:

**These are the movies with highest profit:**

| movie\_title | gross | budget | **profit** |
| --- | --- | --- | --- |
| Avatar | 760505847 | 237000000 | 523505847 |
| Jurassic World | 652177271 | 150000000 | 502177271 |
| Titanic | 658672302 | 200000000 | 458672302 |
| Star Wars: Episode IV - A New Hope | 460935665 | 11000000 | 449935665 |
| E.T. the Extra-Terrestrial | 434949459 | 10500000 | 424449459 |
| The Avengers | 623279547 | 220000000 | 403279547 |
| The Lion King | 422783777 | 45000000 | 377783777 |
| Star Wars: Episode I - The Phantom Menace | 474544677 | 115000000 | 359544677 |
| The Dark Knight | 533316061 | 185000000 | 348316061 |

**C: Top 250:**

Create a new column IMDb\_Top\_250 and store the top 250 movies with the highest IMDb Rating (corresponding to the column: imdb\_score). Also make sure that for all of these movies, the num\_voted\_users is greater than 25,000. Also add a Rank column containing the values 1 to 250 indicating the ranks of the corresponding films.  
Extract all the movies in the IMDb\_Top\_250 column which are not in the English language and store them in a new column named Top\_Foreign\_Lang\_Film. You can use your own imagination also!  
Find IMDB Top 250

Sol: The IMDB top 250 movies are:

| Rank | IMDb\_Top\_250 | imdb\_score | language | num\_voted\_users |
| --- | --- | --- | --- | --- |
| 1 | The Shawshank Redemption | 9.3 | English | 1689764 |
| 2 | The Godfather | 9.2 | English | 1155770 |
| 3 | The Godfather: Part II | 9 | English | 790926 |
| 4 | The Dark Knight | 9 | English | 1676169 |
| 5 | The Good, the Bad and the Ugly | 8.9 | Italian | 503509 |
| 6 | Schindler's List | 8.9 | English | 865020 |
| 7 | The Lord of the Rings: The Return of the King | 8.9 | English | 1215718 |
| 8 | Pulp Fiction | 8.9 | English | 1324680 |
| 9 | Star Wars: Episode V - The Empire Strikes Back | 8.8 | English | 837759 |
| 10 | The Lord of the Rings: The Fellowship of the Ring | 8.8 | English | 1238746 |

Best foreign language movies are:

| Rank | Top\_Foreign\_Lang\_Film | Language |
| --- | --- | --- |
| 5 | The Good, the Bad and the Ugly | Italian |
| 14 | City of God | Portuguese |
| 20 | Spirited Away | Japanese |
| 43 | Oldboy | Korean |
| 45 | AmÃ©lie | French |
| 71 | Pan's Labyrinth | Spanish |

**D: Best Directors:**

Group the column using the director\_name column.  
Find out the top 10 directors for whom the mean of imdb\_score is the highest and store them in a new column top10director. In case of a tie in IMDb score between two directors, sort them alphabetically.  
Find the best directors

Sol: The top 10 directors are:

| **director\_name** | **IMDB\_mean** |
| --- | --- |
| Akira Kurosawa | 8.7 |
| Christopher Nolan | 8.425 |
| David Fincher | 7.75 |
| Francis Ford Coppola | 7.655556 |
| Frank Darabont | 7.975 |
| Irvin Kershner | 7.5 |
| Martin Scorsese | 7.675 |
| Peter Jackson | 7.654545 |
| Quentin Tarantino | 8.2 |
| Sergio Leone | 8.433333 |

**E: Popular Genres:**

Perform this step using the knowledge gained while performing previous steps.  
Find popular genres:

Sol: Most popular genres are:

| **Top\_genres** |
| --- |
| Comedy |
| Action |
| Drama |
| Adventure |
| Crime |
| Biography |
| Horror |

**F: Charts:**

Create three new columns namely, Meryl\_Streep, Leo\_Caprio, and Brad\_Pitt which contain the movies in which the actors: 'Meryl Streep', 'Leonardo DiCaprio', and 'Brad Pitt' are the lead actors. Use only the actor\_1\_name column for extraction. Also, make sure that you use the names 'Meryl Streep', 'Leonardo DiCaprio', and 'Brad Pitt' for the said extraction.  
Append the rows of all these columns and store them in a new column named Combined.  
Group the combined column using the actor\_1\_name column.  
Find the mean of the num\_critic\_for\_reviews and num\_users\_for\_review and identify the actors which have the highest mean.  
Observe the change in number of voted users over decades using a bar chart. Create a column called decade which represents the decade to which every movie belongs to. For example, the title\_year year 1923, 1925 should be stored as 1920s. Sort the column based on the column decade, group it by decade and find the sum of users voted in each decade. Store this in a new data frame called df\_by\_decade.  
  
Find the critic-favorite and audience-favorite actors

Sol: The actor with highest numbers is clearly **Leonardo DiCaprio**

**Actor wise movies**

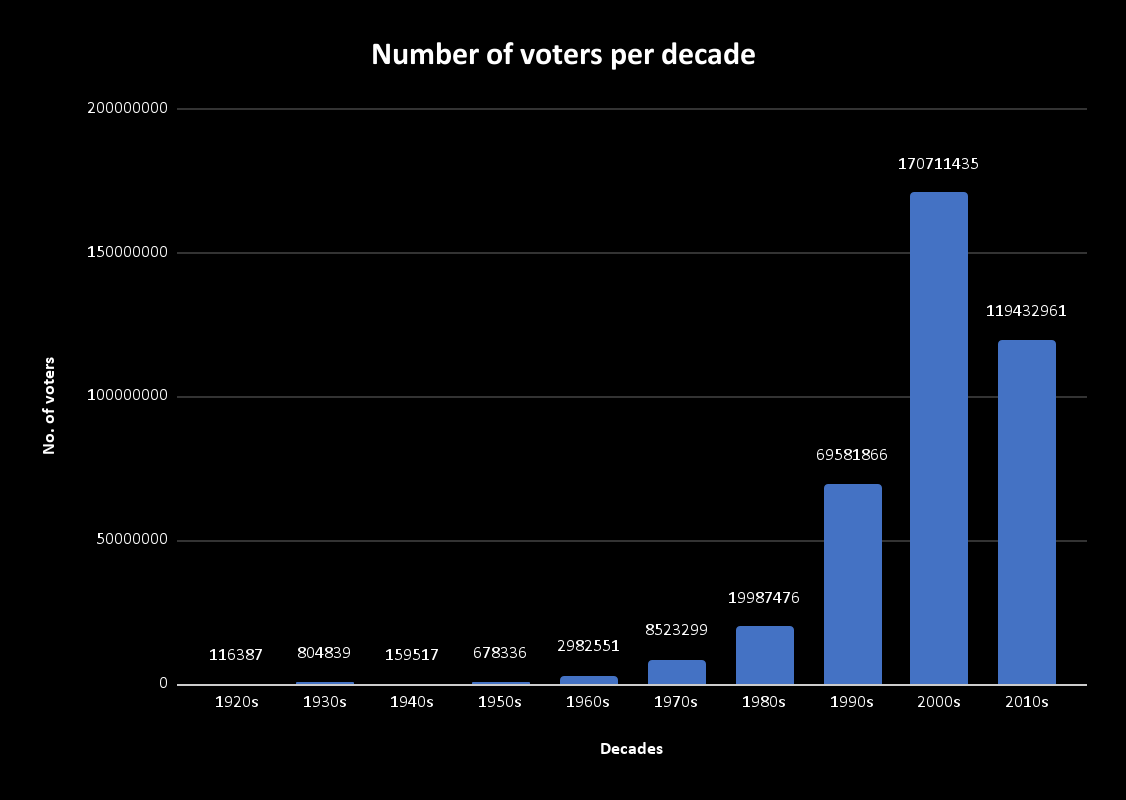
| Meryl\_Streep | Leo\_Caprio | Brad\_Pitt | Combined |
| --- | --- | --- | --- |
| It's Complicated | Titanic | The Curious Case of Benjamin Button | It's Complicated |
| The River Wild | The Great Gatsby | Troy | The River Wild |
| Julie & Julia | Inception | Ocean's Twelve | Julie & Julia |
| The Devil Wears Prada | The Revenant | Mr. & Mrs. Smith | The Devil Wears Prada |
| Lions for Lambs | The Aviator | Spy Game | Lions for Lambs |
| Out of Africa | Django Unchained | Ocean's Eleven | Out of Africa |
| Hope Springs | Blood Diamond | Fury | Hope Springs |
| One True Thing | The Wolf of Wall Street | Seven Years in Tibet | One True Thing |
| The Hours | Gangs of New York | Fight Club | The Hours |

**Average of overall critic and user review count**

| **Average of num\_critic\_for\_reviews** | **167.0405587** |
| --- | --- |
| **Average of num\_user\_for\_reviews** | **335.8976632** |

**Decade wise voted users:**

| Decade | df\_by\_decade |
| --- | --- |
| 1920s | 116387 |
| 1930s | 804839 |
| 1940s | 159517 |
| 1950s | 678336 |
| 1960s | 2982551 |
| 1970s | 8523299 |
| 1980s | 19987476 |
| 1990s | 69581866 |
| 2000s | 170711435 |
| 2010s | 119432961 |

****

# Insights:

Insights refers to accurate understanding of something. These points helps in an insightful understanding of the problems:

* All the problems refer to real-life situations which any data analyst would face while dealing with data. The attributes may differ but the application or approach will not change.
* Here we have only one dataset which shows different features of IMDB movie reviews.
* There happen to be around 5044 records categorized under various columns.
* The dataset is raw or not previously processed. It is basically unclean and needs to be pre-processed and cleaned thoroughly.
* This project has taught me how to vividly analyze the given data and use excel built-in functions with ease

# Result:

To recapitulate, the results are elaborately discussed above, moreover this project/task helped me in better understanding of Excel formulae and working with Excel sheets. It also enhanced my Critical Thinking and Problem-Solving skills. (I could not solve all the questions by using joins. However I managed to draw conclusions using other concepts which are hopefully right).

Thank You.