**INFO 5709 Project 2**

**Multi-Dimensional Data Dimension Design**

**Srinivas Rao PrithviNag Kolla**

**11357028**

In my project, I used python programming in the platform Google Colab on the data sets

“Top 1000 Movie sales information from Box office Mojo Website”, “Top 250 Movies Rating from IMDB”, Ratings and Duration form IMDB Website.

I used python programming and MS-Excel for data transformation and Python Programming for Data Visualization.

**Part 1:**

**Data Scrapping**:

I used “BeautifulSoup Package” to scrape data from the websites.

Example code:

Graphical user interface, text, application

Description automatically generated

**Data Transformation**:

I used “Pandas Package” to transform the data.

* In order to merge the three data sets on the common column Movie Titles: “Title” and remove the rows having null values, Merge() method was used here.

Example code:



* Life time Gross: “Life\_Time\_Gross” column had symbols like ‘$’ and punctuation like ‘,’. In order to remove them I used replace(), pop() methods.
* I used “NumPy package” to arrange the ID after merging the data sets.

**Movie\_Data\_Set:**

Table

Description automatically generated

**Problem Statements:**

1. What is the common duration of movies?
2. What type of movies are made more often?
3. Will IMDB Score impact Life Time Gross of a movie?

**Data Visualization:**

* For Visualization I have used Matplotlib Package.

Chart, box and whisker chart

Description automatically generated

(Figure 1: Box Plot)

* In order to answer the first problem, I used “Box Plot” to show the distribution of duration of movies and I found that most of the movies lie in between 118 to 158 min.
* The median of most of the movies is 138 min.
* I used “Box Plot” instead of “Histogram” because the median is clearly given in “Box Plot” rather than “Histogram”.

“Color Orange Represents the Median, Circle represents the outlier in data.”

Code:

A picture containing text

Description automatically generated

Chart

Description automatically generated

(Figure 2: “Bar Plot”)

* In order to answer problem 2, I used “ Bar Plot” .
* Here, I used count method to count number of occurrences of related Rated movies in the data set and with the help of Matplotlib package, I made “Bar Plot for Ratings”.
* From the figure 2, we infer that ‘R’ rated movies are made often.
* Color blue and different bar sizes represent numbers of movies having that Rating.

Code:

A picture containing diagram

Description automatically generated

Chart, scatter chart

Description automatically generated

(Figure 3: “Scatter Plot”)

* In order to effectively represent the represent the relation between “IMDB Score” and “Life Time Gross”, we took “Scatter Plot” to solve problem 3.
* I have taken top 10 gross movies to find the relation.
* Here, I got to know that the “IMDB Score” didn’t play much in movies Gross Revenue over Life time.
* Color is blue and I haven’t used different colors as I thought it is unnecessary, as we are representing only two axes. The size of circles are default plot sizes.

Code:

Logo, company name

Description automatically generated

**Part 2:**

**Critique the Visualization Tool:**

I found Python programming for data visualization very efficient, simple and dynamic to use. Though, when compared with Tableau. I found that Tableau has no easy scrapping and transformation and it is confusing at times. Tableau is good to integrate large data from servers and analyze and has ready to implement visualizations, which come in handy to save time.

**Tool Strength and Weakness:**

* **Strengths:**

1. Matplotlib has coherent vision of focusing on 2D cross-platform graphics in Python environment and is similar to MATLAB.
2. Matplotlib.pyplot enables complex visualization strategies, like subplots, data transformations and annotations, and has built-in instructions to complete these tasks.

* **Weakness:**

1. It can be difficult to visualize complex data or make complex, multi-field charts using this tool.
2. Multiple backends work well for static images, but can be unwieldly and unpredictable for more dynamic, communicating plots.
3. The ggplot combination in R is one characteristic that users call a clear improvement over Python. Yes, matplotlib is powerful enough to allow the implementation of some of these ideas, but its plotting commands remain rather verbose, and its no-frills, default output looks much more like Excel circa 1993 than ggplot circa 2013.

**Matplotlib Uses:**

Matplotlib is a plotting library used to make different plots like scatter plot, bar plot, box plot, line graph etc. This uses many general-purpose GUI toolkits, such as wxPython, Tkinter, QT, etc., in order to provide object-oriented APIs for embedding plots into applications. Matplotlib is widely used in SciPy as most scientific calculations require plotting of graphs and diagrams.

**Characteristics of UI:**

* User Experience is essential in creating the continuation, this is thoroughly utilized, and the user has simple course learning to apply it, as this will improve the importance of the extension, the content in the extension should explicitly intimate to the user how to communicate with it.
* UI designs come in helpful as they are drawn from most beneficial practices within UI plan and they are reusable answer to commonly happening queries.
* Color is necessary to be used with a scope and accuracy, we have to use it minimally to highlight functionality in the extension applying useful colors will relate to the user what to concentrate on. The font should be a distinct hierarchy of content identified by using various font sizes, weight, color and spacing.