Alexandria University
Faculty of Computers and Data Science
Data Science Methodology
Hagar Hazem Mohammed: 20191482504
Hady Ihab Ahmed: 20191450774
Mohammed Amr Shehata: 20191565917

Act Report

Visualization during assessment:

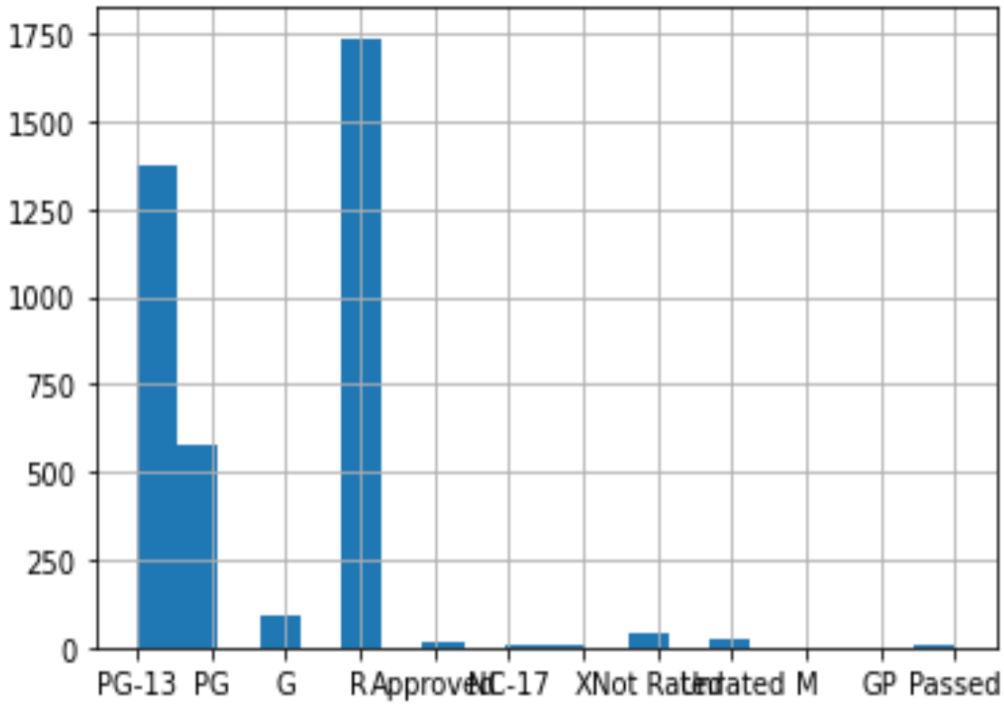
By using isnull().sum() functions we allow to know that the non numeric columns content_rating, color and language in movies dataset has 297,17 and 10 null values respectively, so we determine to solve this missing value problem by filling the null values of the column by the most frequency element in it so we used histogram shape to know the that element by using .hist() function from matplotlib in pandas package, we used this way again to solve this problem in plot_keyword_1 and plot_keyword_2 during tidiness stage.

1. content_rating:

As it is shown in the picture R is the most repeated content rating so we fill the 297 null values with it

```
In [77]: movies_clean.content_rating.hist(bins=20)
```

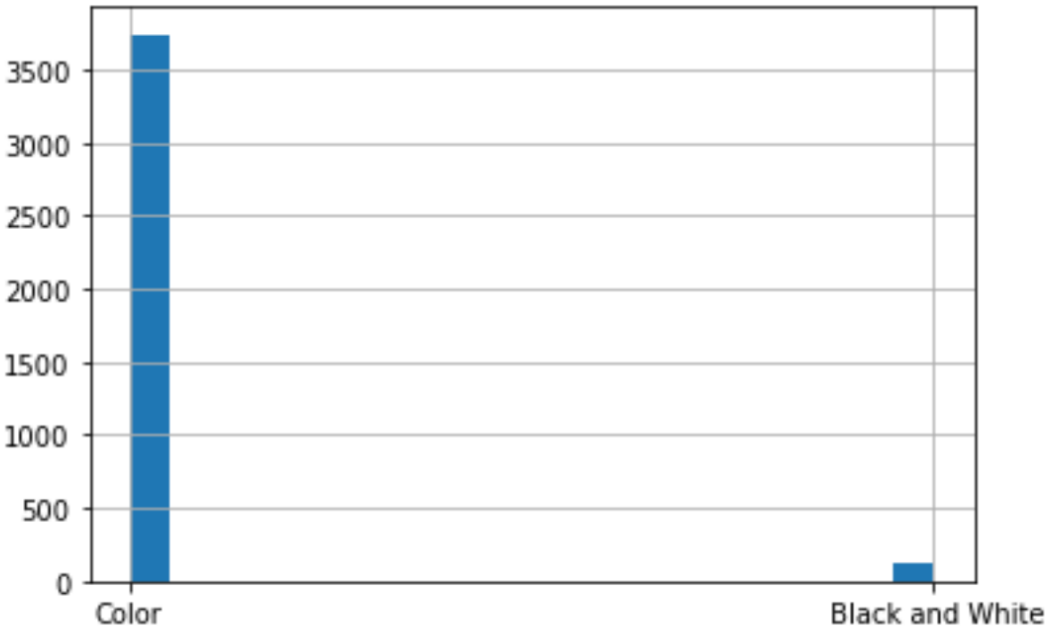
```
Out[77]: <AxesSubplot:>
```



2. Color:

As it is shown in the picture Color is the most repeated color so we fill the 17 null values with it.

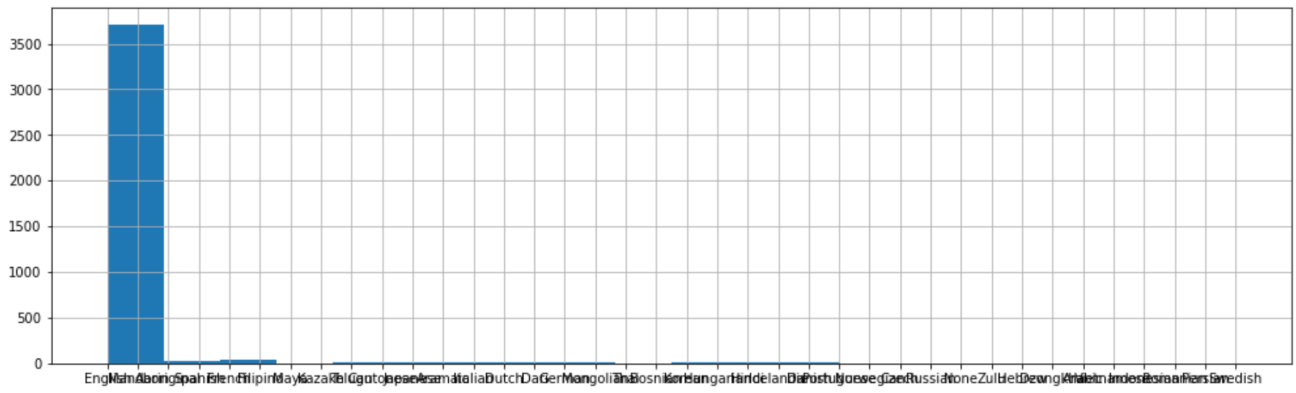
```
In [76]: movies_clean.color.hist(bins=20)
Out[76]: <AxesSubplot:>
```



3. language:

As it is shown in the picture English is the most repeated language so we fill the 10 null values with it.

```
In [78]: movies_clean.language.hist(figsize=(17,5),bins=20)
Out[78]: <AxesSubplot:>
```



4. **plot_keyword_1 and plot_keyword_2:**

As shown in the following two figure (figure 1 and figure 2) the values in plot_keyword_1 and plot_keyword_2 are large, so the most repeated value is not clear, therefore histogram is not the best way to know that value, so we used another function in the pandas called idxmax() that return that values easily as shown in figure 3.

Figure 1:

```
In [80]: movies_clean.plot_keyword_1.hist(figsize=(20,5),bins=20)
Out[80]: <AxesSubplot:>
```

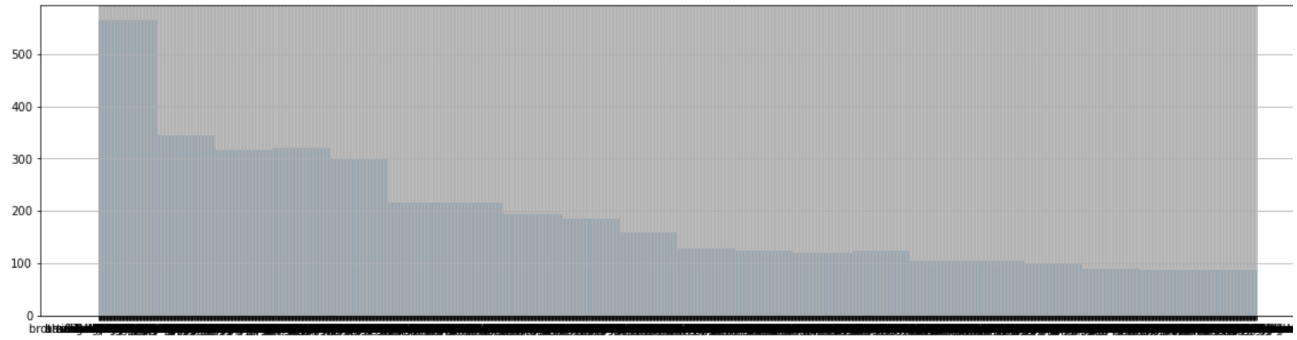


Figure 2:

```
In [81]: movies_clean.plot_keyword_2.hist(figsize=(20,5),bins=20)
Out[81]: <AxesSubplot:>
```

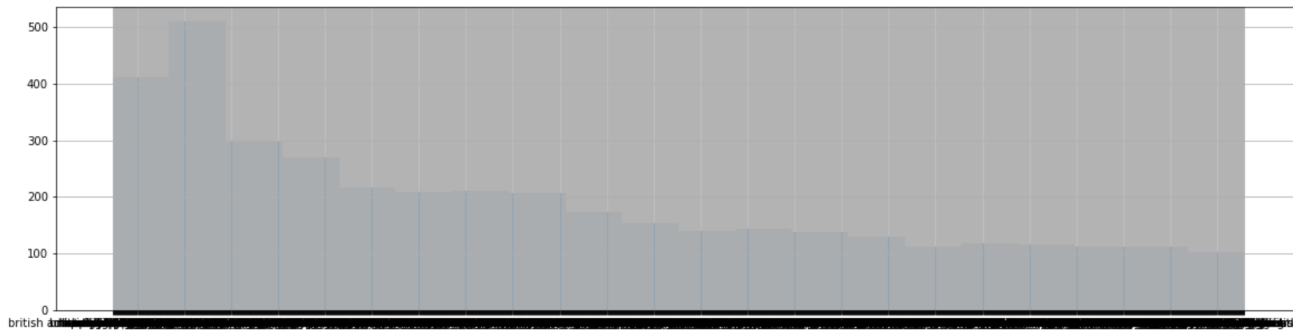


Figure 3:

```
In [83]: x = movies_clean['plot_keyword_1'].value_counts().idxmax()
         y = movies_clean['plot_keyword_2'].value_counts().idxmax()
         x,y

Out[83]: ('alien', 'friend')
```

Visualization after assessment:

By using plot function from matplotlib in pandas dataframe, we allowed to plot the boxplot and know the outliers of each attribute(column). We applied the boxplot on duration attribute as shown in figure 1 and we noticed many outliers and that's because of the real meaning of our dataset. for example, the average duration of short movies is 20 minutes and we have many short movies in the dataset and because of the important of short movies in the real life we can not just erase them. In figure 2 we noticed many outliers in imdb_score as well. Imdb score is the rank of the movie that takes range (0,10) in float and as known, any ranks or grade are normally distributed which means that only small values will be upper and lower outliers.

Figure 1:

```
In [68]: movies_clean.duration.plot(kind='box')
Out[68]: <AxesSubplot:>
```

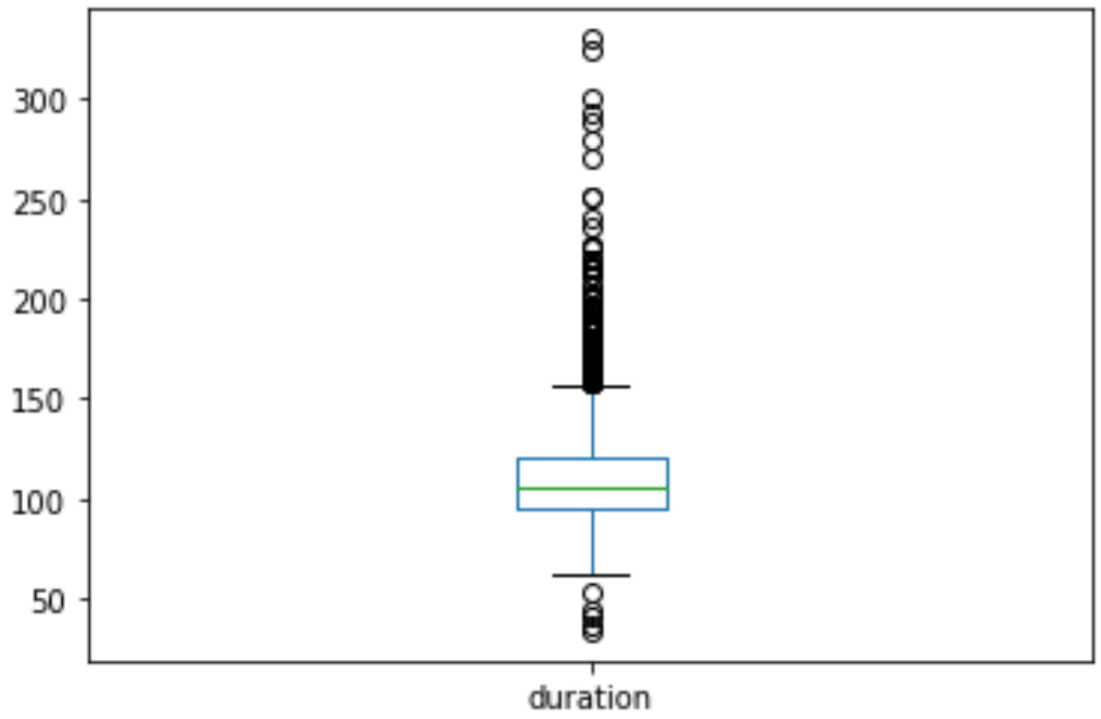
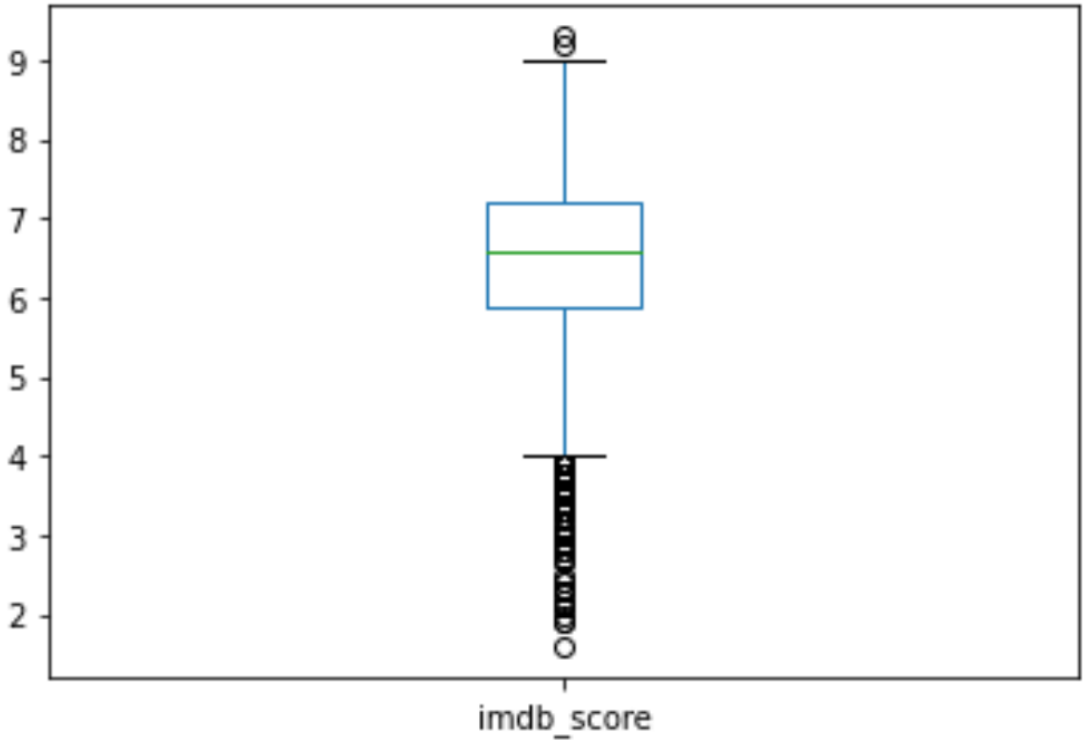


Figure 2:

```
In [67]: movies_clean.imdb_score.plot(kind='box')
```

```
Out[67]: <AxesSubplot:>
```



Also we do two scatter plot to describe the relationship between title_year as a input and budget as a output as in figure 1, and the another relationship between budget as a input and gross as a outputs as in figure 2.

Figure 1:

Not very strong but there is a +ve correlation between title_year and budget.

```
In [69]: #+correlated
         movies_clean.plot(y='budget', x='title_year', kind='scatter',xlim = (1950,2018),ylim = (0,300000000))
```

```
Out[69]: <AxesSubplot:xlabel='title_year', ylabel='budget'>
```

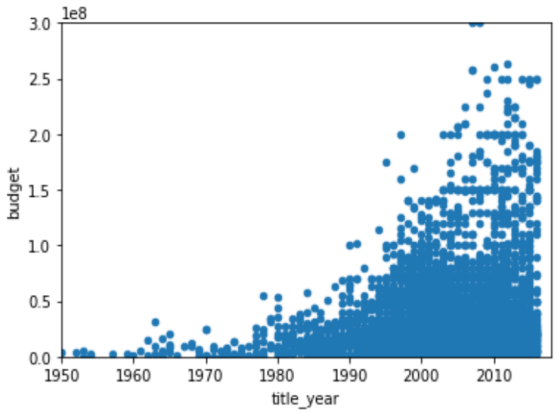


Figure 2:

It is obvious that budget and gross are uncorrelated as shown is the following figure.

```
In [70]: #uncorrelated
         movies_clean.plot(y='gross', x='budget', kind='scatter',xlim = (0,150000000),ylim = (0,80000000))
```

```
Out[70]: <AxesSubplot:xlabel='budget', ylabel='gross'>
```