DATA SCIENCE FOR BUSINESS (DATA ANALYTICS)

PREDICTION OF MOVIE SUCCESS USING RANDOM FOREST A MACHINE LEARNING ALGORITHMS AND THEIR COMPARISON

III. PRACTICAL FEASIBILITY

A.

Cost

Python and R both are open-source platforms of programming language and they are cost-free.

Easy to use

Python is an object-oriented programming language that makes it easier to write large-scale, maintainable, and robust code.

Easy to learn

R is less similar to other common programming languages. Python is simple enough to be a good first programming language to learn.

Community support

R and Python both have large open source communities. Libraries/tools are added continuously to their respective catalogs.

Range of libraries

Both analytical tools are enriched with libraries like:

Python libraries - Numpy, SciPy, Pandas, Keras, SciKit-Learn, TensorFlow, Matplotlib, Seaborn, PyTorch, XGbost etc.

R libraries - tidyverse, dplyr, tidyr, stringr, lubridate, ggplot2, grammar of graphes, ggvis, rgl, htmlwidges, leaflet, dygraphs, DT, diagrammeR, network3D, threeJS etc.

Computational speed

It takes about two minutes and two seconds. The Python code was faster than the R alternative. The R code for this pipeline is 3.3 times faster than the Python code.

Memory Consumption

R's limitations make Python better for data science and machine learning. R is not scalable. Python is a single-threaded language that runs in RAM, which means it is memory-constrained.

Visualization Capacity

Machine learning and advanced analytics are helping humans make sense of large amounts of structured and unstructured data using our natural visual learning ability. Visualization is present here.

Python and R are programming languages that help humans to understand vast datasets.

System Integration

Both Python and R have great packages to solve any kind of problem. There are so many different possibilities to choose from. Python is the obvious choice for these jobs because it is very flexible to integrate other systems.

B. Visualization

Raw data analysis, pre-processing with statistical tools, and data visualization. The initial step loads the dataset imdb movie into the R environment through the reading command.

Figure 1. The command names (data frame) lists all the variables in the dataset

Subset variable

The required variables for analysis will be considered and that will be moved into the “myvars”.

The specified data with specified variables will be a subset with the above code.

The new dataset will have the attributes or columns as in Figure 2.

Dimension of Dataset

Now the dataset has 5043 records and 13 attributes

Summary of each attribute in the dataset Figure 3.

Figure 4 shows various dimensions

Data pre-processing

Not null values will be removed from the dataset using the command “na.omit”. Figure 5. After removing the na values the dimensionality of the dataset gets reduced.

IMDB score and gross

The blot shows the relationship between IMDB score and gross Figure 6.

When the IMDB score is increased the number of people watching that movie is also increased until the score 7.5. After the score of 7.5, the number of people watching that movie based on IMDB score is get reduced. Figure 7. shows the negative correlation between gross and IMDB after the threshold value 7.5 [18].

IMDB and number of critic reviews

The high IMDB score has a greater number of critic reviews. The number of critic reviews and IMDB score is positively correlated with each other.

Run the plot command Figure 8.

The movie released year and number of movies

Run the gplot command Figure 9.

From the analysis of movie released year and number of movies released on that, a greater number of movies are released after the year 2000 Figure 10.

Movie release year and gross

Run the plot using the command Figure 11.

The relationship between movie release year and gross, media industry released more number of movies after the year 2000 as well as got more gross income Figure 12.

Correlation analysis

Correlation among the variables such as budget, duration, number of faces in the poster, imdb score, genres, number of critic's reviews, and Facebook likes for director are analyzed.

In the correlation matrix, positive values represent the high positive correlation between the two variables Figure 13.

Negative values indicate a negative correlation between the two analyzed variables. Fewer values indicate that two variables do not correlate that much.

null values will be removed from the dataset using the

Country analysis

While grouping according to the country, Figure 14. indicates that more movies are released from the country USA. Least number of movies released from the country Hong Kong from the top 10 countries

Movie Subset of variable analysis

The relationship between variables such as "num_critic_for_reviews", duration", "facenumber_in_poster","director_facebook_likes","actor_3_facebook_likes","actor_2_facebook_likes","actor_1_facebook_likes","budget","title_year","num_voted_users","imdb_score","gross" are using the scatter plot Figure 15.

The following factors are observed in Figure 16. such as

Positively correlated variables:

number of votes and number of critics

actor 3 Facebook likes, and actor 2 Facebook likes

number of voted and gross [19]

Correlation plot for the analyzed variables

Figure 17. shows the correlation plot for the analyzed variables.

Model build

The model is developed using the R with random regression Figure 18.

PYTHON

The movie data is analyzed in Python. Initially, the dataset will be loaded into an environment similar to R-environment.

The packages such as NumPy and pandas are required to import the dataset into the python environment. The dataset will be read into the python environment.

From the dataset aspect ratio and IMDb link will be removed as well as the dependent variable such as the IMDb score is also dropped.

The column names are displayed while executing the command “column.values. It is illustrated in Figure 19.

Data preprocessing is done using Python packages

The imdb score distribution is illustrated in Figure 20. using the package matplotlib. The below code illustrates the imdb distribution

The graph Figure 21 illustrates that on average, most of the movies got a score of 6.5. Very few movies got a score of 9.

Correlation is done with packages Imputer and StandardScalaer. Perform the Impute for removing the outlier in the dataset. From the scipy the classifier used for modeling.

The correlation coefficient of the numeric features is illustrated in Figure 22. Figure 23. the correlation matrix shows the relationship between each considered variables. The number of movie critics is highly correlated with the number of movie Facebook likes. Duration is not correlated with any other variables. There is a high correlation between the number of voted users and the number of reviews, gross and movie Facebook likes are also highly related. Model validation is illustrated in Figure 24.