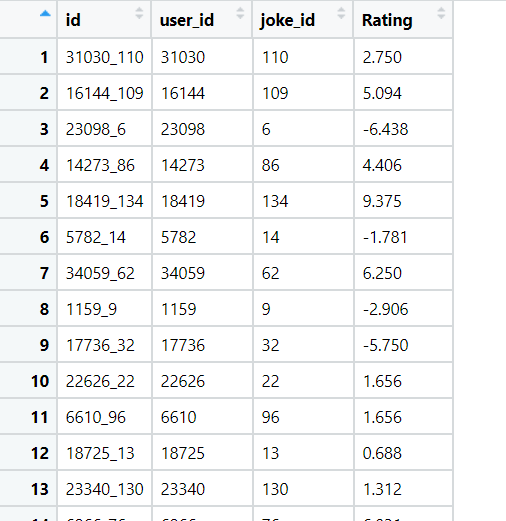
# Topic: Recommendation Engine

Q) Build a recommender system with the given data using UBCF.

**Description of the data**

In this dataset have users on the rows rated the jokes in the columns. The dataset comprises two csv files, Jokes.csv and Rating.csv is formatted as an excel file. The Jokes file consist of joke\_id and Jokes & Rating.csv has the ratings given by the users to the jokes. Each rating is from (-10.00 to +10.00) and 99

corresponds to a null rating (user did not rate that joke). Note that the ratings are real values ranging from -10.00 to +10.00.



**Ans:**

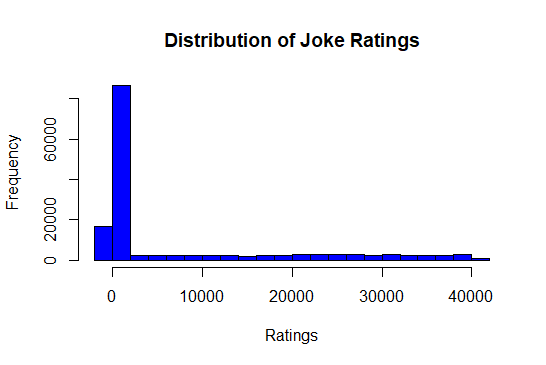
* Loading the dataset using the library readxl

**Data Preprocessing:**

* Removing unnecessary columns like “id”.
* Checking the NA values, as there are no NA values no further imputation is required.
* There are 50000 observations in the dataset with 3 variables

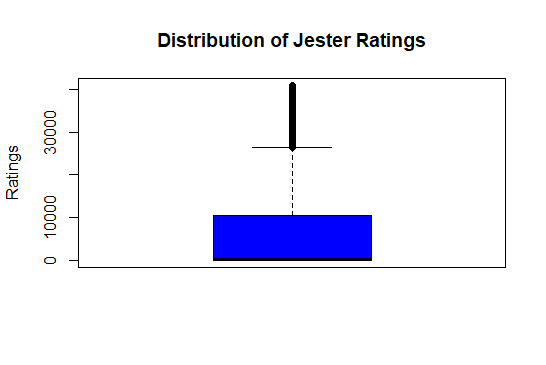
**Graphical Representation:**

**Histogram Representation:**

****

* From the histogram, clearly shows that the data is right skewed in nature

**Boxplot Representation:**



* The above box plot shows that there exist outliers which effects the score of the data.

**Converting data into Matrix:**

* Now converting the data into matrix format after data preprocessing.
* Now the new dimensions of the matrix are [24175, 139].
* Now based on the “recommender lab real rating matrix format”, object is created with the converted matrix.
* Using 5 different recommender functions, 5 different recommendations are built.
* Here are the 5-recommender function: UBCF, IBFC, SVD, POPUALR and binarize functions

**Creating Recommendations for Users:**

* With the help of above constructed recommender system now we can recommend jokes for each individual user.
* Taking no of recommendation = 3
* **For User\_id = 45 (given 4 ratings, given are the below recommendations)**
* With UBCF the jokes recommended are “113”,”3”, and “97”.
* With IBCF the jokes recommended are “46”, “1” and “4”.
* With SVD the jokes recommended are “124”,”117” and “25”.
* With POPULAR the jokes recommended are “79”, “94” and “104”.
* With binarize the jokes recommended are “7” and “8”.
* **For User\_id = 439 (given 5 ratings, given are the below recommendations)**
* With UBCF the jokes recommended are “104”,”77”, and “15”.
* With IBCF the jokes recommended are “2”, “9” and “11”.
* With SVD the jokes recommended are “106”,”117” and “25”.
* With POPULAR the jokes recommended are “79”, “94” and “104”.
* With binarize the jokes recommended are “9” and “13”.

Here 9 repeated twice, so these jokes are recommended most to the user

* **For User\_id = 2634 (given 6 ratings, given are the below recommendations)**
* With UBCF the jokes recommended are no recommendations.
* With IBCF the jokes recommended are no recommendations
* With SVD the jokes recommended are “1”,”2” and “3”.
* With POPULAR the jokes recommended are “140”, “94” and “43”.
* With binarize the jokes recommended are “6” and “8”.

# Hints:

1. Business Problem
   1. Objective
   2. Constraints (if any)
2. Data Pre-processing

2.1 Data cleaning, Feature Engineering, EDA etc.

1. Model Building
   1. Partition the dataset
   2. Model(s) - Reasons to choose any algorithm
   3. Model(s) Improvement steps
   4. Model Evaluation
   5. Python and R codes
2. Deployment

4.1 Deploy solutions using R shiny and Python Flask.

1. Result Share the benefits/impact of the solution - how or in what way the business (client) gets benefit from the solution provided.

**Note:**

1. For each assignment the solution should be submitted in the format
2. Research and Perform all possible steps for improving the model(s) recommendations
3. All the codes (executable programs) are running without errors
4. Documentation of the module should be submitted along with R & Python codes, elaborating on every step mentioned here