

# RESTAURANT RATING RECOMMENDATION ENGINE

Created by: Ali Ahad

## DATA AND TOOLS:

The data used for this project is contained in a CSV file which comprises of columns for user id, restaurant id, overall rating, rating based on food and rating based on service. The recommendation engine is built on python 3.7.1 and uses libraries such as numpy, pandas, matplotlib, scikit-learn and scipy etc.

## EVALUATION METRICS:

To quantify the model performances, I have used the method of root mean squared error (RSME), a method used for standard regression problems, which tells us how far our predictions are from true labels. Given  $n$  pairs of true labels  $t$  and predicted labels  $p$ , the RSME equation is:

$$RSME = \sqrt{\frac{\sum_{i=1}^n (p_i - t_i)^2}{n}}$$

## DATA FLOW:

- I start with the processing of data. After opening the file, load the user ids and restaurant ids in a list making sure that they are all unique.
- Next, I initialize the user-restaurant pair matrix using a pandas data frame and allocate the values inside them through a for loop over the entire file.
- I split the dataset by allocating 70% of the data to training set and 30% of the data to testing set.
- Next, I created a benchmark model using the evaluation matrix which later helps us to see how well we have optimized our data. This benchmark model is created for overall ratings, food ratings, restaurant service ratings and total average restaurant ratings.
- Once, the benchmark model is completed, I now use the matrix factorization technique to optimize our model. First, a cost function is created using a vectorized implementation to predict the rating which we will minimize using gradient descent technique.
- Once, I have the optimized result, I use that to find the optimized ratings and calculate the RSME for our optimized data.
- Top get the recommendation list, I used the predicted ratings and create a dictionary with keys being the restaurant id and values being their corresponding average predicted rating for that feature. I have used all four features stated above to print the recommendation list for each of them.

## RESULTS:

- The results of this project are a dictionaries of Top 10 recommended ratings based on different types of services.
- Additionally, I have also given box-plots of benchmark models and optimized models in the folders "benchmark\_images" and "optimized\_images".