*“We are leaving the age of information and entering the age of recommendation.”*

A recommender system makes predictions based on users’ historical behaviors.

Specifically, it’s to predict user preference for a set of items based on past experience.

To build a recommender system, the most two popular approaches are Content-based and Collaborative Filtering.

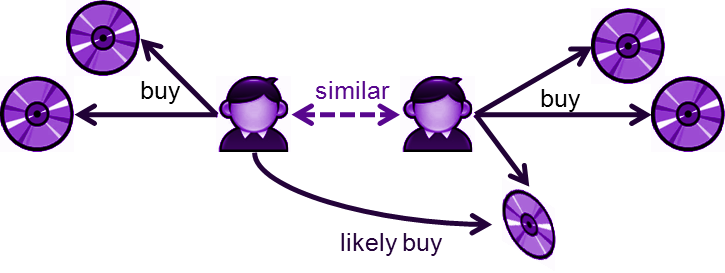
Content-based approach requires a good amount of information about items' own features, rather than using users’ interactions and feedback. For example, it can be movie attributes such as genre, year, director, actor 0 etc., or textual content of articles that can be extracted by applying Natural Language Processing.

Collaborative Filtering, on the other hand, doesn’t need anything else except users’ historical preference on a set of items. Because it’s based on historical data, the core assumption here is that the users who have agreed in the past tend to also agree in the future. In terms of user preference, it is usually expressed by two categories. Explicit and Implicit Rating

Explicit Rating, is a rate given by a user to an item on a sliding scale, like 5 stars for Titanic.This is the most direct feedback from users to show how much they like an item.

Implicit Rating, suggests users preference indirectly, such as page views, clicks, purchase records, whether or not listen to a music track, and so on

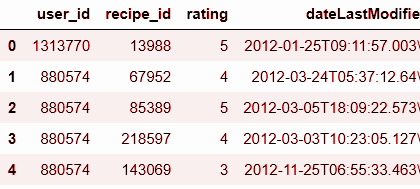
We have used explicit collaborative filtering.

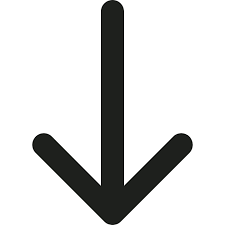


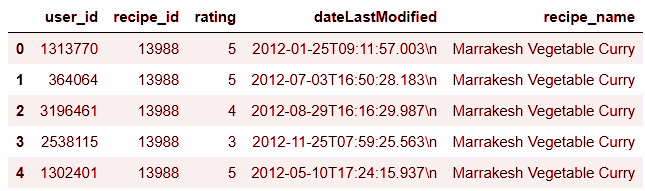
Collaborative Filtering

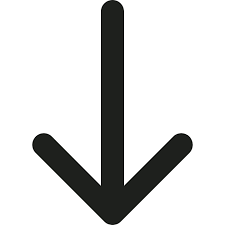
An extensive dataset is used from kaggle having user id, recipe ids, rating given by the user.

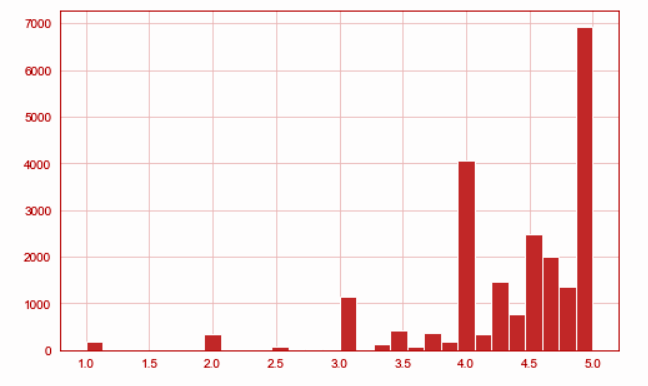
**Explicit Rating Collaborative Filtering**



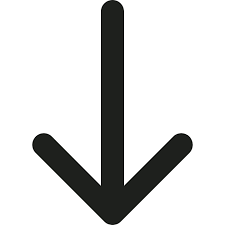


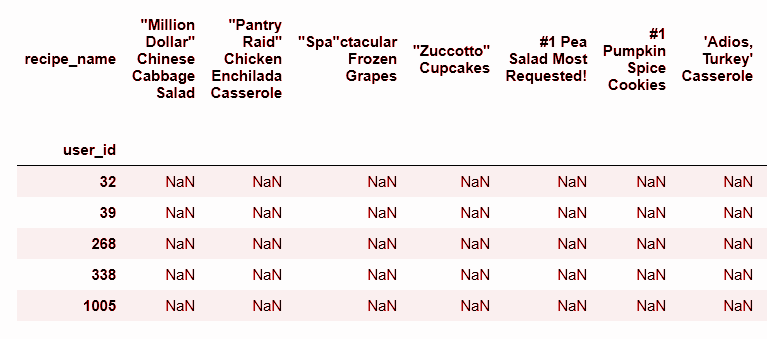




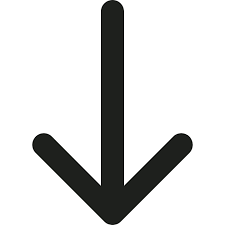


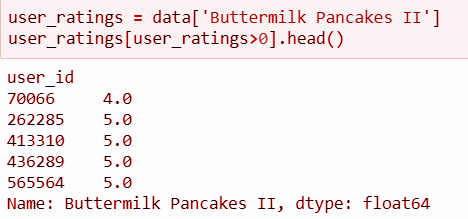
Histogram for ratings, so the majority of the ratings fall in 3 to 5 region



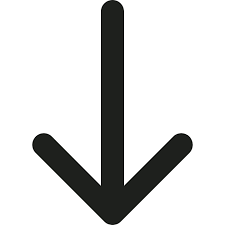


Pivot Table



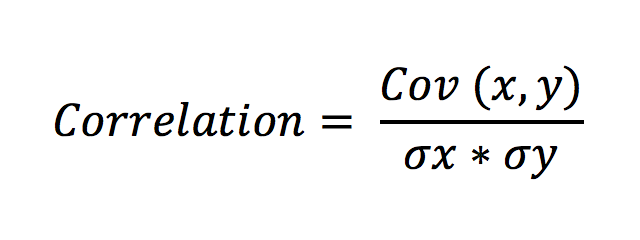


Dish data

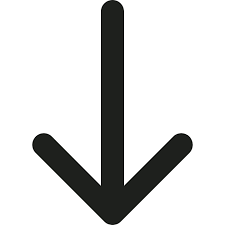




**Correlation** is a statistical technique that can show whether and how strongly pairs of variables are related.

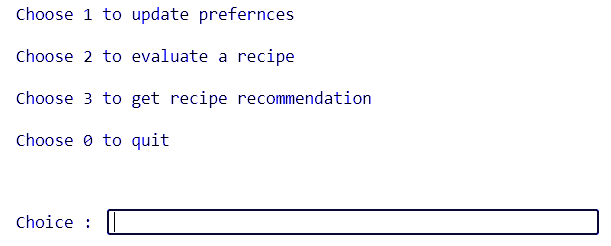


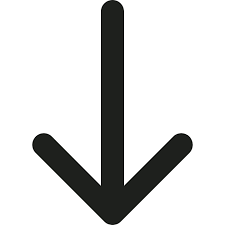
In simple words, both the terms measure the relationship and the dependency between two variables. “**Covariance**” indicates the direction of the linear relationship between variables. “**Correlation**” on the other hand measures both the strength and direction of the linear relationship between two variables



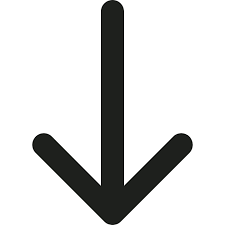


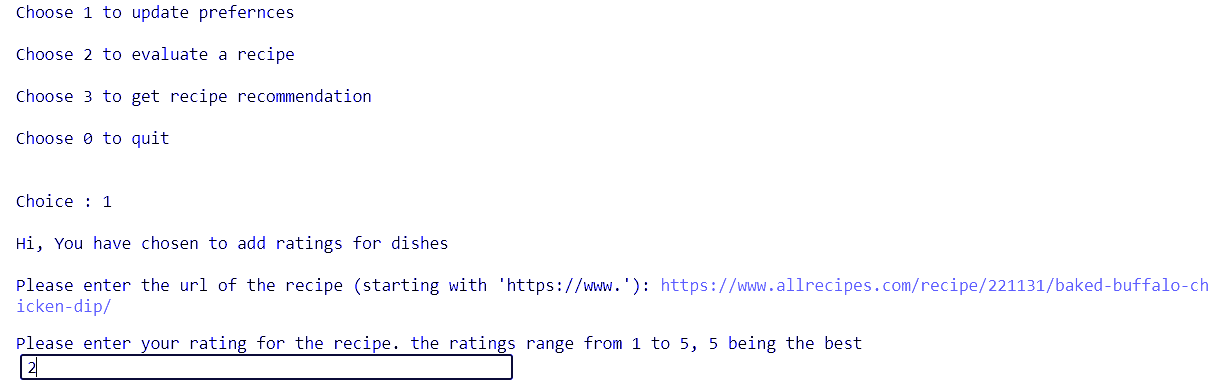
Multinomial Naive Bayes & Cosine Similarity Recommender

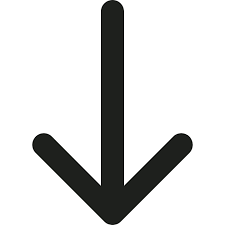


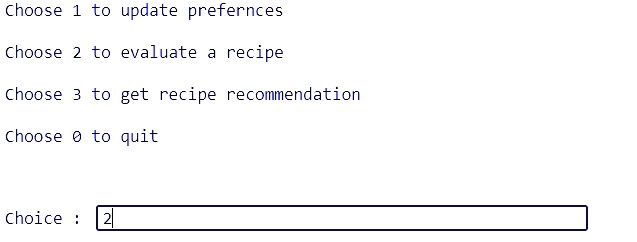


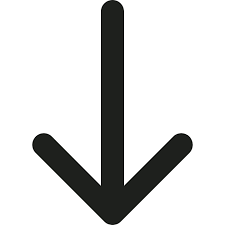




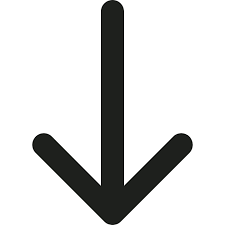


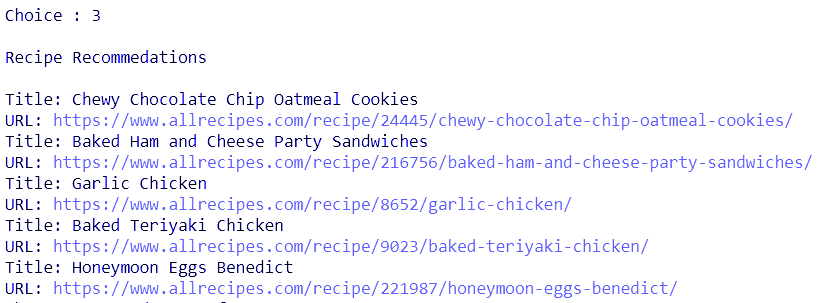












**Evaluating a dish**

Some codes are meant to transform features- normalise numericals or turn text into vectors, or fill up missing data, they are **transformers**. Other codes are meant to predict variables by fitting an algorithm such as a random forest or support vector machine (SVM), they are **estimators.**

So, in a pipeline, we first sequentially apply a list of transformers (data modelling) and then a final estimator (ML model).

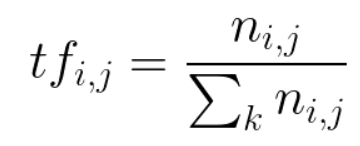
The transform steps must implement fit() and transform().

The final step, estimator, should implement fit() and predict().

**Transformer used if TFIDF**

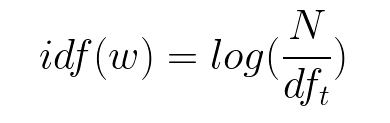
**Term Frequency (TF)**

The number of times a word appears in a document divided by the total number of words in the document. Every document has its own term frequency.

****

**Inverse Data Frequency (IDF)**

The log of the number of documents divided by the number of documents that contain the word ***w***. Inverse data frequency determines the weight of rare words across all documents in the corpus.



Multinomial NB Classifier : Estimator

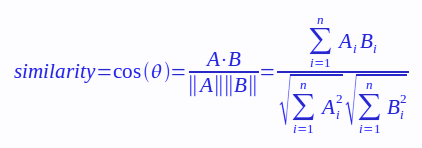
Preferences for few dishes are known.

For each such dish, the ingredient list gives a particular rating.

We need a rating of x dish with y list of ingredients.

**Recommendation**

Cosine Similarity between highly rated dishes and not rated dishes is used to find best recommendations

****