In [1]:

```
import numpy as np
import pandas as pd
from math import sqrt
from sklearn.metrics import mean_squared_error
from sklearn.metrics.pairwise import pairwise_distances
```

In [4]:

```
# root mean square error

def rmse(pred, test):
    pred = pred[test.nonzero()].flatten()
    test = test[test.nonzero()].flatten()
    return sqrt(mean_squared_error(pred, test))

def predict(ratings, similarity, type='user'):
    if type == 'user':
        mean_user_rating = ratings.mean(axis=1)
        #We use np.newaxis so that mean_user_rating has same format as ratings
        ratings_diff = (ratings - mean_user_rating[:, np.newaxis])
        pred = mean_user_rating[:, np.newaxis] + similarity.dot(
            ratings_diff) / np.array([np.abs(similarity).sum(axis=1)]).T
    elif type == 'book':
        pred = ratings.dot(similarity) / \
            np.array([np.abs(similarity).sum(axis=1)])
    return pred
```

In [12]:

```
#Considering only 10000 rows of data
df_ratings = pd.read_csv("BX-Book-Ratings.csv", encoding="latin1")
df_ratings.sort_values(["user_id", "isbn"], inplace=True)
df_ratings = df_ratings.head(10000)
df_ratings.reset_index()
df_ratings.head()

n_users = df_ratings["user_id"].unique().shape[0]
print(n_users)

n_books = df_ratings["isbn"].unique().shape[0]
print(n_books)

data_matrix = np.zeros((n_users, n_books))
for line in df_ratings.head().itertuples():
    data_matrix[line[1]-1, line[2]-1] = line[3]

print(data_matrix.shape)


user_similarity = pairwise_distances(data_matrix, metric='cosine')
item_similarity = pairwise_distances(data_matrix.T, metric='cosine')

user_prediction = predict(data_matrix, user_similarity, type='user')
book_prediction = predict(data_matrix, item_similarity, type='book')


print("Root mean square error of user prediction")
print(rmse(user_prediction, data_matrix))

print("Root mean square error of book prediction")
print(rmse(book_prediction, data_matrix))
```

```
1323
8742

---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-12-96735a289b05> in <module>
     14 data_matrix = np.zeros((n_users, n_books))
     15 for line in df_ratings.head().itertuples():
---> 16     data_matrix[line[1]-1, line[2]-1] = line[3]
     17 
     18 print(data_matrix.shape)

TypeError: unsupported operand type(s) for -: 'str' and 'int'
```