# Top Match

March 7, 2019

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [3]:
```
rating_data=pd.read_csv('/Users/ziranmin/Desktop/Sophia/ml-latest-small/ratings.csv')
rating_data.shape
```

Out[3]: (100836, 4)

In [4]:
```
rating_data.head()
```

Out[4]:

| | userId | movieId | rating | timestamp |
|---|---|---|---|---|
| 0 | 1 | 1 | 4.0 | 964982703 |
| 1 | 1 | 3 | 4.0 | 964981247 |
| 2 | 1 | 6 | 4.0 | 964982224 |
| 3 | 1 | 47 | 5.0 | 964983815 |
| 4 | 1 | 50 | 5.0 | 964982931 |

In [112]:
```
def pearson(rating_data, id_1, id_2):
    # find movies that both users have rated
    a = rating_data.loc[rating_data['userId'] == id_1][["movieId","rating"]]
    a = a.rename(columns ={"rating":"rating_one"})
    b = rating_data.loc[rating_data['userId'] == id_2][["movieId","rating"]]
    b = b.rename(columns ={"rating":"rating_two"})
    combined =  pd.merge(a,b)

    #special case
    if len(combined) == 0:
        return 0

    # rating list of user 1
    A = list(combined.rating_one)
    # rating list of user 2
    B = list(combined.rating_two)

    sum1 = 0
    sum2 = 0
```

```
    sum1sq = 0
    sum2sq = 0
    psum = 0
    n = len(A)
    for i in range(n):
        sum1 += A[i]
        sum2 += B[i]
        sum1sq += A[i]**2
        sum2sq += B[i]**2
        psum += A[i] * B[i]
    num = psum - (sum1 * sum2/n)
    den =((sum1sq - sum1**2 / n)*(sum2sq - sum2**2 / n))**0.5

    #special case
    if den == 0:
        return 0

    return num/den
```

In [113]:
```
def cosine(rating_data, id_1, id_2):
    # find movies that both users have rated
    a = rating_data.loc[rating_data['userId'] == id_1][["movieId","rating"]]
    a = a.rename(columns ={"rating":"rating_one"})
    b = rating_data.loc[rating_data['userId'] == id_2][["movieId","rating"]]
    b = b.rename(columns ={"rating":"rating_two"})
    combined =  pd.merge(a,b)

    #special case
    if len(combined) == 0:
        return 0

    # rating list of user 1
    A = list(combined.rating_one)
    # rating list of user 2
    B = list(combined.rating_two)

    dot_product = np.dot(A, B)
    norm_a = np.linalg.norm(A)
    norm_b = np.linalg.norm(B)
    return dot_product / (norm_a * norm_b)
```

In [109]:
```
def topMatch(rating_data, id_1, sim_function):
    best_id = 0
    best_sim = -10
    for i in rating_data['userId'].unique():
        if i != id_1:
            current_score = sim_function(rating_data, id_1, i)
            if current_score > best_sim:
```

```
                        best_sim = current_score
                        best_id = i
            return best_id
```

```
In [143]: topMatch(rating_data, 3, pearson)
```

```
Out[143]: 95
```

```
In [142]: topMatch(rating_data, 3, cosine)
```

```
Out[142]: 495
```

# 1 Example in video

https://www.bing.com/videos/search?q=recommendation+systems+collaborative+filtering+university+of+wa

```
In [130]: df=pd.read_csv('/Users/ziranmin/Desktop/Sophia/ml-latest-small/example.csv')
```

# 2 Got same result at 11:53 in video

```
In [137]: for i in range(1,7):
              print(pearson(df, 7, i))
```

```
0.9912407071619299
0.38124642583151164
-1.0
0.8934051474415647
0.9244734516419049
0.66284898035987
```