

Recommender System

```
In [1]: import pandas as pd
import numpy as np
```

```
In [2]: ratings_df = pd.read_csv('/Users/theranmeadows/Desktop/Bellevue University/dsc630predict
titles_df = pd.read_csv('/Users/theranmeadows/Desktop/Bellevue University/dsc630predict
```

```
In [3]: ratings_df.head()
```

```
Out[3]:
```

	userId	movieId	rating	timestamp
0	1	296	5.0	1147880044
1	1	306	3.5	1147868817
2	1	307	5.0	1147868828
3	1	665	5.0	1147878820
4	1	899	3.5	1147868510

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

```
Out[4]:
```

	movieId	title	genres
0	1	Toy Story (1995)	Adventure Animation Children Comedy Fantasy
1	2	Jumanji (1995)	Adventure Children Fantasy
2	3	Grumpier Old Men (1995)	Comedy Romance
3	4	Waiting to Exhale (1995)	Comedy Drama Romance
4	5	Father of the Bride Part II (1995)	Comedy

```
In [5]: movies_df = ratings_df.merge(titles_df, on = 'movieId', how = 'left')
```

```
In [6]: movies_df.head()
```

```
Out[6]:
```

	userId	movieId	rating	timestamp	title	genres
0	1	296	5.0	1147880044	Pulp Fiction (1994)	Comedy Crime Drama Thriller
1	1	306	3.5	1147868817	Three Colors: Red (Trois couleurs: Rouge) (1994)	Drama
2	1	307	5.0	1147868828	Three Colors: Blue (Trois couleurs: Bleu) (1993)	Drama
3	1	665	5.0	1147878820	Underground (1995)	Comedy Drama War
4	1	899	3.5	1147868510	Singin' in the Rain (1952)	Comedy Musical Romance

```
In [7]: # keep the useful columns
movies_df = movies_df[['userId', 'title', 'rating']]
```

```
In [8]: movies_df.head()
```

```
Out[8]:
```

	userId	title	rating
0	1	Pulp Fiction (1994)	5.0
1	1	Three Colors: Red (Trois couleurs: Rouge) (1994)	3.5
2	1	Three Colors: Blue (Trois couleurs: Bleu) (1993)	5.0
3	1	Underground (1995)	5.0
4	1	Singin' in the Rain (1952)	3.5

```
In [9]: # get the count of rating each movie has
movie_ratings = pd.DataFrame(movies_df.groupby('title')['rating'].count())
```

```
In [10]: movie_ratings = movie_ratings.reset_index()
```

```
In [12]: movie_ratings.head()
```

```
Out[12]:
```

	title	rating
0	"BLOW THE NIGHT!" Let's Spend the Night Togeth...	1
1	"Great Performances" Cats (1998)	179
2	#1 Cheerleader Camp (2010)	9
3	#Captured (2017)	2
4	#Female Pleasure (2018)	3

```
In [13]: # merge the rating back into the original data
movies_df = movies_df.merge(movie_ratings, on = 'title', how = 'left')
```

```
In [14]: #change the names of some columns for clarity
movies_df = movies_df.rename(columns = {'rating_y':'ratings count'})
movies_df = movies_df.rename(columns = {'rating_x':'rating'})
```

```
In [15]: # drop movies with less than 100 ratings
movies_df = movies_df[movies_df['ratings count'] > 100]
```

```
In [16]: movies_df['rating'] = movies_df['rating'].astype(np.float32)
```

```
In [17]: total_ratings = ''
titles_df = ''
```

```
In [18]: user_df = movies_df.pivot_table(index = 'userId', columns = 'title', values = 'rating')
```

```
In [29]: movie = 'Terminator, The (1984)'
```

```
In [30]: # Calculate correlations
```

```
# Calculate correlations
correlations = user_df.corrwith(user_df[movie]).sort_values(ascending=False)

# Print movie suggestions
print('Your movie suggestions are:\n')
for i in range(1,11):
    print(correlations.index[i])
```

Your movie suggestions are:

Terminator 2: Judgment Day (1991)
The Bremen Town Musicians (1969)
Too Many Cooks (2014)
Three from Prostokvashino (1978)
Dead Snow 2: Red vs. Dead (2014)
Winnie the Pooh Goes Visiting (1971)
What We Did on Our Holiday (2014)
Captain Underpants: The First Epic Movie (2017)
George Carlin: Jammin' in New York (1992)
Winnie the Pooh and the Day of Concern (1972)