

```
In [62]: import pandas as pd
import numpy as np
from scipy import sparse
from sklearn.metrics.pairwise import cosine_similarity
```

```
In [63]: ratings = pd.read_csv('courses.csv')
movies = pd.read_csv('courseID.csv')
scores = pd.concat([movies, ratings] , axis=1, sort=False)
scores.head()
```

Out[63]:

	course ID	Name	userId	Course	scores
0	1.0	Robotics 1	1	1	3
1	2.0	Robotics 2	1	10	4
2	3.0	Robotics 3	1	9	4
3	4.0	AI 1	1	4	2
4	5.0	AI 2	2	4	2

```
In [64]: user_scores = scores.pivot_table(index = ['userId'], columns = ['Name'], values = 'scores')
user_scores.head()
```

Out[64]:

Name	AI 1	AI 2	AI 3	Electronics 1	Electronics 2	Mathematics 1	Mathematics 2	Robotics 1	Robotics 2	Robotics 3
------	------	------	------	---------------	---------------	---------------	---------------	------------	------------	------------

userId

1	2.0	NaN	NaN	NaN	NaN	NaN	NaN	3.0	4.0	4.0
2	NaN	2.0	5.0	5.0	NaN	5.0	3.0	NaN	NaN	NaN
3	NaN	NaN	NaN	NaN	5.0	NaN	NaN	NaN	NaN	NaN

```
In [65]: user_scores = user_scores.fillna(0)
user_scores = user_scores.replace(np.nan, 0)
user_scores.head(10)
```

Out[65]:

	Name	AI 1	AI 2	AI 3	Electronics 1	Electronics 2	Mathematics 1	Mathematics 2	Robotics 1	Robotics 2	Robotics 3
userId											
1	2.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.0	4.0	4.0
2	0.0	2.0	5.0	5.0	0.0	5.0	3.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0	5.0	0.0	0.0	0.0	0.0	0.0

```
In [72]: course_similarity_df = user_scores.corr(method='pearson')
course_similarity_df.head(100)
```

Out[72]:

	Name	AI 1	AI 2	AI 3	Electronics 1	Electronics 2	Mathematics 1	Mathematics 2	Robotics 1	Robotics 2	Robotics 3
--	------	------	------	------	---------------	---------------	---------------	---------------	------------	------------	------------

Name												
AI 1	1.0	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	1.0	1.0	1.0	
AI 2	-0.5	1.0	1.0	1.0	-0.5	1.0	1.0	1.0	-0.5	-0.5	-0.5	
AI 3	-0.5	1.0	1.0	1.0	-0.5	1.0	1.0	1.0	-0.5	-0.5	-0.5	
Electronics 1	-0.5	1.0	1.0	1.0	-0.5	1.0	1.0	1.0	-0.5	-0.5	-0.5	
Electronics 2	-0.5	-0.5	-0.5	-0.5	1.0	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	
Mathematics 1	-0.5	1.0	1.0	1.0	-0.5	1.0	1.0	1.0	-0.5	-0.5	-0.5	
Mathematics 2	-0.5	1.0	1.0	1.0	-0.5	1.0	1.0	1.0	-0.5	-0.5	-0.5	
Robotics 1	1.0	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	1.0	1.0	1.0	
Robotics 2	1.0	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	1.0	1.0	1.0	
Robotics 3	1.0	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	1.0	1.0	1.0	

```
In [98]: def get_similar_courses(course_name, user_score):
similar_score = course_similarity_df[course_name]*(user_score - 2.5)
similar_score = similar_score.sort_values(ascending=False)

return similar_score
```



```
In [98]: def get_similar_courses(course_name, user_score):  
        similar_score = course_similarity_df[course_name]*(user_score - 2.5)  
        similar_score = similar_score.sort_values(ascending=False)  
  
        return similar_score  
  
print(get_similar_courses("Robotics 2",5))
```

```
Name  
Robotics 3      2.50  
Robotics 2      2.50  
Robotics 1      2.50  
AI 1            2.50  
Mathematics 1   -1.25  
Electronics 2   -1.25  
Electronics 1   -1.25  
AI 3            -1.25  
AI 2            -1.25  
Mathematics 2   -1.25  
Name: Robotics 2, dtype: float64
```

```
In [92]: Robotics_lover = [("Robotics 1",1),("Robotics 3",2)]
similar_scores = pd.DataFrame()

for course,score in Robotics_lover:
    similar_scores = similar_scores.append(get_similar_courses(course,score),ignore_index = True)

similar_scores.head()
similar_scores.sum().sort_values(ascending = False)
```

```
Out[92]: AI 2          1.0
Mathematics 2      1.0
Electronics 2      1.0
Mathematics 1      1.0
Electronics 1      1.0
AI 3              1.0
Robotics 3        -2.0
Robotics 2        -2.0
Robotics 1        -2.0
AI 1             -2.0
dtype: float64
```

```
In [ ]:
```