cosine_similarity_cosine_distance

In [1]:
```
#cosine distance is not negative
from sklearn.metrics.pairwise import cosine_similarity,cosine_distances
```

In [3]:
```
cosine_similarity([[3,1]],[[6,2]]) #tells how similar they are ratio:3;1
```

Out[3]:
```
array([[1.]])
```

In [4]:
```
cosine_similarity([[3,1]],[[3,2]]) #3 iphones 1 galaxy  cosine distance = 1 - cosine similarity
```

Out[4]:
```
array([[0.96476382]])
```

In [5]:
```
cosine_distances([[3,1]],[[6,2]]) #e^-16 ~0 which means they are almost similar
```

Out[5]:
```
array([[1.11022302e-16]])
```

In [6]:
```
import pandas as pd #iphone and galaxy are coloumns
df = pd.DataFrame([
    {'iPhone': 3,'galaxy':1},
    {'iPhone': 2,'galaxy':0},
    {'iPhone': 1,'galaxy':3},
    {'iPhone': 1,'galaxy':2},
],
index=["doc1",
       "doc2",
      "doc3",
      "doc4"])
df
```

Out[6]:

	iPhone	galaxy
doc1	3	1
doc2	2	0
doc3	1	3
doc4	1	2

In [9]:
```
cosine_similarity(df.loc["doc1":"doc1"],df.loc["doc2":"doc2"])#comparing from table
#comparing 1 with 2 #94%similar
```

Out[9]:
```
array([[0.9486833]])
```

In [10]:
```
cosine_distances(df.loc["doc1":"doc1"],df.loc["doc2":"doc2"])
```

Out[10]:
```
array([[0.0513167]])
```

In [11]:
```
cosine_similarity(df.loc["doc1":"doc1"],df.loc["doc3":"doc3"])
#6% similar
```

Out[11]:
```
array([[0.6]])
```

In [12]:
```
cosine_distances(df.loc["doc1":"doc1"],df.loc["doc3":"doc3"])
```

Out[12]:
```
array([[0.4]])
```

In [14]:
```
cosine_similarity(df.loc["doc3":"doc3"],df.loc["doc4":"doc4"])
```

Out[14]:
```
array([[0.98994949]])
```

In [16]:
```
cosine_distances(df.loc["doc3":"doc3"],df.loc["doc4":"doc4"])
```

Out[16]:
```
array([[0.01005051]])
```

In [17]:
```
import pandas as pd
import numpy as np
```

In [18]:
```
df = pd.read_csv("movie_revenues.csv")
```

```
df.head()
```

Out[18]:

	budget	genres	homepage	id	keywords	original_language	original_title	overview	popularity	produ
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.avatarmovie.com/	19995	[{"id": 1463, "name": "culture clash"}, {"id":...	en	Avatar	In the 22nd century, a paraplegic Marine is di...	150.437577	[{"n F
1	300000000	[{"id": 12, "name": "Adventure"}, {"id": 14, "...	http://disney.go.com/disneypictures/pirates/	285	[{"id": 270, "name": "ocean"}, {"id": 726, "na...	en	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...	139.082615	[{"na Pic
2	245000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.sonypictures.com/movies/spectre/	206647	[{"id": 470, "name": "spy"}, {"id": 818, "name...	en	Spectre	A cryptic message from Bondâ€™s past sends him...	107.376788	[{"
3	250000000	[{"id": 28, "name": "Action"}, {"id": 80, "nam...	http://www.thedarkknightrises.com/	49026	[{"id": 849, "name": "dc comics"}, {"id": 853,...	en	The Dark Knight Rises	Following the death of District Attorney Harve...	112.312950	[{"n Pictu
4	260000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://movies.disney.com/john-carter	49529	[{"id": 818, "name": "based on novel"}, {"id":...	en	John Carter	John Carter is a war-weary, former military ca...	43.926995	[{"na

df.revenue.describe()

In [19]:
```
df['revenue_mln']= df['revenue'].apply(lambda x: x/1000000)
```

```
df.revenue_mln.describe() #revenue was an absolute dollar value.
#To avoid a large scale a new col is added revenue
```

Out[19]:
```
count    4803.000000
mean       82.260639
std       162.857101
min         0.000000
25%         0.000000
50%        19.170001
75%        92.917187
max      2787.965087
Name: revenue_mln, dtype: float64
```

In [20]:
```
_,mean,std,*_ = df.revenue_mln.describe() #*_ to remove everything after std
#_ to remove count
```

In [21]:
```
mean
```

Out[21]:
```
82.26063865167605
```

In [22]:
```
std
```

Out[22]:
```
162.85710094282982
```

Probability

In [23]:
```
#len is used to count the number of rows in a dataset
import pandas as pd
import numpy as np
```

In [24]:
```
df = pd.read_csv('student-mat.csv')
df.head(3)
```

Out[24]:

	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	Fjob	...	famrel	freetime	goout	Dalc	Walc	health	absences	G1	G2	G
0	GP	F	18	U	GT3	A	4	4	at_home	teacher	...	4	3	4	1	1	3	6	5	6	
1	GP	F	17	U	GT3	T	1	1	at_home	other	...	5	3	3	1	1	3	4	5	5	
2	GP	F	15	U	LE3	T	1	1	at_home	other	...	4	3	2	2	3	3	10	7	8	

3 rows × 33 columns

In [26]:
```
len (df)
```

Out[26]: 395

In [27]:
```
df['grade_A'] = np.where(df['G3']*5>=80,1,0) #1 means if the statement inside where is true
df
```

Out[27]:

	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	Fjob	...	freetime	goout	Dalc	Walc	health	absences	G1	G2	G3	g
0	GP	F	18	U	GT3	A	4	4	at_home	teacher	...	3	4	1	1	3	6	5	6	6	
1	GP	F	17	U	GT3	T	1	1	at_home	other	...	3	3	1	1	3	4	5	5	6	
2	GP	F	15	U	LE3	T	1	1	at_home	other	...	3	2	2	3	3	10	7	8	10	
3	GP	F	15	U	GT3	T	4	2	health	services	...	2	2	1	1	5	2	15	14	15	
4	GP	F	16	U	GT3	T	3	3	other	other	...	3	2	1	2	5	4	6	10	10	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
390	MS	M	20	U	LE3	A	2	2	services	services	...	5	4	4	5	4	11	9	9	9	
391	MS	M	17	U	LE3	T	3	1	services	services	...	4	5	3	4	2	3	14	16	16	
392	MS	M	21	R	GT3	T	1	1	other	other	...	5	3	3	3	3	3	10	8	7	
393	MS	M	18	R	LE3	T	3	2	services	other	...	4	1	3	4	5	0	11	12	10	
394	MS	M	19	U	LE3	T	1	1	other	at_home	...	2	3	3	3	5	5	8	9	9	

395 rows × 34 columns

In [28]:
```
df['high_absences'] = np.where(df['G3']>10,1,0)
df
```

Out[28]:

	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	Fjob	...	goout	Dalc	Walc	health	absences	G1	G2	G3	grade_A	h
0	GP	F	18	U	GT3	A	4	4	at_home	teacher	...	4	1	1	3	6	5	6	6	0	
1	GP	F	17	U	GT3	T	1	1	at_home	other	...	3	1	1	3	4	5	5	6	0	
2	GP	F	15	U	LE3	T	1	1	at_home	other	...	2	2	3	3	10	7	8	10	0	
3	GP	F	15	U	GT3	T	4	2	health	services	...	2	1	1	5	2	15	14	15	0	
4	GP	F	16	U	GT3	T	3	3	other	other	...	2	1	2	5	4	6	10	10	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
390	MS	M	20	U	LE3	A	2	2	services	services	...	4	4	5	4	11	9	9	9	0	
391	MS	M	17	U	LE3	T	3	1	services	services	...	5	3	4	2	3	14	16	16	1	
392	MS	M	21	R	GT3	T	1	1	other	other	...	3	3	3	3	3	10	8	7	0	
393	MS	M	18	R	LE3	T	3	2	services	other	...	1	3	4	5	0	11	12	10	0	
394	MS	M	19	U	LE3	T	1	1	other	at_home	...	3	3	3	5	5	8	9	9	0	

395 rows × 35 columns

In [29]:
```
df['count']=1
```

In [30]:
```
df = df[['grade_A','high_absences','count']]
df.head()
```

Out[30]:

	grade_A	high_absences	count
0	0	0	1
1	0	0	1
2	0	0	1
3	0	1	1
4	0	0	1

In [34]:
```
pd.pivot_table(df,
               values='count',
              index = ['grade_A'],
              columns=['high_absences'],
              aggfunc=np.size,
              fill_value=0) #grade A with high absence
#grade A with no absence
```

Out[34]:

high_absences	0	1
grade_A		
0	186	169
1	0	40

P(A) = having >=80 P(B)= probability of having missed more than 10 classes P(A and B)= 5/sum of all values P(A|B)= p(A and B)/p(B)

In [54]:
```
import random
heads =0
tails =0
```

In [55]:
```
def coin_flips(trails):#defining a function
    global heads
    global tails
    for i in range(trails):
        flip = random.randint(1,2)
        if flip==1:
            heads +=1
```

```
        else:
            tails +=1
        print(flip)
    print("Heads:" + str(heads))
    print("Tails:" + str(tails))

coin_flips(10) #passing the number of trials here
```

```
2
2
2
1
2
1
1
1
1
1
Heads:6
Tails:4
```

SKEWNESS,KURTOSIS

In [56]:
```
import scipy
from scipy.stats import skew
from scipy.stats import kurtosis
```

In [57]:
```
dataset = [88,85,82,97,67,77,74,86,
           81,95,77,88,85,76,81]
```

In [59]:
```
from scipy.stats import skew
print(skew(dataset, axis=0, bias=True)) #peak
#It signifies that the distribution is positively skewed
```

```
0.029331688766181797
```

In [60]:
```
print(kurtosis(dataset, axis=0, bias=True))
#it signifies that the distribution has more values in the tails compared to a normal distribution.
```

```
-0.29271198374234686
```

skewness=0 means normally distributed skewness>0 = inclined towards left skewness<0 = inclined towards right

kurtosis > 3--> it is trying to identify more outliers kurtosis <3 --> playkurtic normal distribution comes under +ve kurtosis

general guideline for modified z score is to use 3.5 as a threshold...i.e anything

that has a mod z score of 3.5 or more is an outlier