In [1]:
```
import pandas as pd
import numpy as np
import warnings
warnings.filterwarnings("ignore")
```

In [2]:
```
drug=pd.read_csv(r"C:\Users\reshma_koduri\Downloads\drug discovery.csv")
drug
```

Out[2]:

	id	gen	smile	source	score	P_value	Conn
0	AAAA	0	COc1cccc(NC(=O)Cc2ccc(NC(=O)N3CCCC3)cc2)c1	generated	99.9	0.003320	-0
1	AAAB	0	C=CCNC(=O)CNc1cccc(C(=O)N(C)CCc2ccccc2)c1	generated	99.9	0.003384	-0
2	AAAC	0	CC(=O)Nc1ccc(S(=O)(=O)Nc2ccc(C)c(C)c2)cc1	training	99.9	0.003397	-0
3	AAAD	0	CCOC(=O)C1=C(C(=O)OCC)C(c2cccc(Cl)c2)NC(=O)N1	generated	99.9	0.003427	-0
4	AAAE	0	NC(=O)c1ccc(NC(=O)C(CC(=O)O)NC(=O)c2cc(-c3cccc...	generated	99.9	0.003468	-0
...	...	...	...	...	...	...	
1172	ABTC	0	CCCC1(CCc2ccccc2)CC(O)=C(C(CC)c2cccc(NS(=O)(=O...	manual	99.9	1.000000	0
1173	ABTD	0	O=C1Nc2ccc(Cl)cc2C(C#CC2CC2)(C(F)(F)F)O1	manual	99.9	1.000000	0
1174	ABTE	0	CC(C)(C)NC(=O)C1CN(Cc2cccnc2)CCN1CC(O)CC(Cc1cc...	manual	99.9	1.000000	0
1175	ABTF	0	CCOP(=O)(COc1ccc(CC(NC(=O)OC2COC3OCCC23)C(O)CN...	manual	99.9	1.000000	0
1176	ABTG	0	COC(=O)NC(C(=O)NCCCCC(CO)N(CC(C)C)S(=O)(=O)c1c...	manual	99.9	1.000000	0

1177 rows × 8 columns

In [3]:
```
drug.describe()
```

Out[3]:

	gen	score	P_value	Connectivity
count	1177.0	1.177000e+03	1177.000000	1177.000000
mean	0.0	9.990000e+01	0.677346	-0.049811
std	0.0	2.004582e-12	0.392285	0.163636
min	0.0	9.990000e+01	0.003320	-0.809330
25%	0.0	9.990000e+01	0.257861	-0.209375
50%	0.0	9.990000e+01	1.000000	0.000000
75%	0.0	9.990000e+01	1.000000	0.000000
max	0.0	9.990000e+01	1.000000	0.723920

In [4]:
```
drug.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1177 entries, 0 to 1176
Data columns (total 8 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   id            1177 non-null   object 
 1   gen           1177 non-null   int64  
 2   smile         1177 non-null   object 
 3   source        1177 non-null   object 
 4   score         1177 non-null   float64
 5   P_value       1177 non-null   float64
 6   Connectivity  1177 non-null   float64
 7   name          1177 non-null   object 
dtypes: float64(3), int64(1), object(4)
memory usage: 73.7+ KB
```

In [7]:
```
drug['P_value'].unique()
```

Out[7]:
```
array([0.00331968, 0.00338356, 0.00339726, 0.00342658, 0.0034682 ,
       0.00353087, 0.00355484, 0.00357524, 0.00358432, 0.00363285,
       0.0037343 , 0.00389023, 0.00396885, 0.0042186 , 0.00450414,
       0.00486791, 0.00550357, 0.00628477, 0.00643704, 0.00645932,
       0.00664478, 0.00697121, 0.00697142, 0.00757783, 0.00895931,
       0.00914021, 0.00915257, 0.00949567, 0.00980125, 0.01052681,
       0.01165854, 0.01199205, 0.01328552, 0.01357921, 0.01416115,
       0.01450921, 0.01463088, 0.01463118, 0.01520905, 0.01579844,
       0.01631903, 0.01670527, 0.01700551, 0.01744755, 0.01788815,
       0.01826373, 0.0196488 , 0.01981558, 0.02005149, 0.02089379,
       0.02101087, 0.02143682, 0.02161217, 0.0219501 , 0.02234632,
       0.02278351, 0.02366028, 0.023799  , 0.02386443, 0.02525801,
       0.02753619, 0.02795078, 0.02799259, 0.02833159, 0.02851516,
       0.02862323, 0.0286239 , 0.0289735 , 0.03181352, 0.03252018,
       0.03329107, 0.03354188, 0.03489295, 0.03526253, 0.03633253,
       0.03719375, 0.03806932, 0.0385287 , 0.03897426, 0.03898927,
       0.03990402, 0.04046838, 0.04186649, 0.0418879 , 0.04202482,
       0.04278029, 0.04308126, 0.04330748, 0.04456587, 0.04456773,
       0.04492331, 0.04518693, 0.04588357, 0.04598648, 0.04660964,
       0.04676691, 0.04695373, 0.04730705, 0.04757408, 0.04768286,
       0.04870295, 0.05051713, 0.05053752, 0.05088461, 0.05182202,
       0.05275996, 0.05312042, 0.05427295, 0.05438078, 0.05514716,
       0.0552889 , 0.05568718, 0.05570817, 0.05605124, 0.05631784,
       0.05651632, 0.05871457, 0.06042298, 0.06051576, 0.06052005,
       0.06195518, 0.06382694, 0.06484602, 0.06511162, 0.06520212,
       0.06801367, 0.06811696, 0.06867639, 0.06892979, 0.06939979,
       0.06946658, 0.06976797, 0.06985624, 0.0704378 , 0.07101772,
       0.0715195 , 0.07235723, 0.07247133, 0.07256171, 0.07294612,
       0.07348202, 0.07354344, 0.07386633, 0.07582825, 0.07832784,
       0.07905461, 0.07907366, 0.07934654, 0.07973238, 0.07980567,
       0.07985288, 0.08081229, 0.08163699, 0.08181505, 0.08269558,
       0.08323564, 0.08490506, 0.08534807, 0.08605934, 0.08635457,
       0.08788697, 0.08806246, 0.08879437, 0.08895638, 0.09034253,
       0.09052705, 0.09104235, 0.09104315, 0.09104351, 0.09164863,
       0.09238933, 0.09306361, 0.09512018, 0.09555039, 0.09599071,
       0.09605064, 0.09713684, 0.0985899 , 0.09952477, 0.10055522,
       0.10233047, 0.10413599, 0.10451931, 0.10484266, 0.1062647 ,
       0.10697024, 0.10719087, 0.10858597, 0.11185228, 0.11190014,
       0.11269503, 0.11280344, 0.11365458, 0.11459782, 0.11519795,
       0.11654348, 0.11750316, 0.11889455, 0.11979169, 0.12018904,
       0.12029071, 0.12276693, 0.12353802, 0.12390583, 0.12528238,
       0.12540111, 0.12793708, 0.12933567, 0.12954606, 0.12988079,
```

```
0.13063647, 0.13096454, 0.13260898, 0.13355352, 0.1357788 ,
0.1361732 , 0.13671895, 0.13886054, 0.13927272, 0.14051   ,
0.14070866, 0.14088994, 0.14303162, 0.14806286, 0.14975012,
0.15256049, 0.15369048, 0.153924  , 0.15497247, 0.15562023,
0.15601202, 0.159757  , 0.16224638, 0.16280553, 0.16292021,
0.16302445, 0.16377314, 0.16875503, 0.1704928 , 0.17167054,
0.17234037, 0.17408346, 0.17500657, 0.17746683, 0.17785168,
0.17914667, 0.18192299, 0.18262087, 0.18344105, 0.18454918,
0.18738514, 0.18805847, 0.18832182, 0.19356944, 0.19369538,
0.19398345, 0.19751145, 0.19822424, 0.2005027 , 0.20072904,
0.20129993, 0.2031834 , 0.2045194 , 0.20949127, 0.21140648,
0.2126268 , 0.21262762, 0.21613169, 0.21780821, 0.21813464,
0.219131  , 0.21973215, 0.22184002, 0.22306153, 0.2255502 ,
0.22735973, 0.22909195, 0.22978112, 0.23050537, 0.23165731,
0.23432149, 0.23781397, 0.23836461, 0.23956356, 0.24059466,
0.2416225 , 0.24277041, 0.24295499, 0.24399913, 0.24697031,
0.25342106, 0.25381506, 0.25786067, 0.25936012, 0.2618198 ,
0.2629654 , 0.26451257, 0.26835093, 0.27131117, 0.27422719,
0.27746106, 0.28264132, 0.28696908, 0.28860334, 0.28989767,
0.29014537, 0.29041303, 0.29251472, 0.29309081, 0.29326431,
0.29479569, 0.2954055 , 0.29569829, 0.29959839, 0.30115981,
0.30311155, 0.30610146, 0.30647185, 0.31038972, 0.31091985,
0.31130012, 0.31566784, 0.31799599, 0.31969131, 0.32697298,
0.33026311, 0.33493444, 0.33584273, 0.33698531, 0.33813792,
0.33906063, 0.3397717 , 0.34077915, 0.3415142 , 0.34348328,
0.34451839, 0.34566832, 0.35022996, 0.35345171, 0.35524473,
0.35578981, 0.35606765, 0.3565432 , 0.35925243, 0.35978151,
0.36077026, 0.3611143 , 0.3645551 , 0.36466251, 0.36488603,
0.36599735, 0.36632923, 0.36732451, 0.37056154, 0.37266574,
0.37691524, 0.3793924 , 0.382819  , 0.38502115, 0.38670005,
0.38963828, 0.3904334 , 0.39046557, 0.39338427, 0.39496421,
0.39533117, 0.39597373, 0.39611395, 0.39717108, 0.40041048,
0.40283063, 0.40335082, 0.40725778, 0.40862107, 0.41547754,
0.41691513, 0.41779437, 0.41875114, 0.42050279, 0.42392585,
0.42457283, 0.43150115, 0.43405619, 0.4345    , 0.43491885,
0.43743994, 0.44159002, 0.44313879, 0.44344144, 0.44504064,
0.44840355, 0.45309878, 0.4532101 , 0.45540034, 0.45643784,
0.45777869, 0.46177079, 0.46387273, 0.46650919, 0.46773491,
0.47100626, 0.47891295, 0.48408713, 0.4890688 , 0.49065342,
0.49091957, 0.49403979, 0.49502733, 0.50120567, 0.51428287,
0.51527591, 0.51545886, 0.51561018, 0.51607102, 0.51700611,
0.51828149, 0.53071362, 0.53259767, 0.53337217, 0.53477984,
0.53610198, 0.53855403, 0.54437613, 0.54716412, 0.54831384,
0.54929137, 0.55035048, 0.55075521, 0.55199519, 0.55631202,
0.55880135, 0.55986261, 0.56173782, 0.56272893, 0.56286225,
0.56591826, 0.56785224, 0.56978596, 0.57093103, 0.57407658,
0.57999508, 0.58535212, 0.58550422, 0.58626813, 0.59541269,
0.59746188, 0.59822391, 0.60353854, 0.60386134, 0.60742148,
0.60751863, 0.60976589, 0.61140965, 0.62799904, 0.62815295,
0.62953156, 0.63818857, 0.64599034, 0.64846181, 0.66346141,
0.66994783, 0.67341921, 0.67359182, 0.67436709, 0.67515299,
0.67690288, 0.68476138, 0.68754834, 0.69626519, 0.70578261,
0.71110034, 0.71729496, 0.72085179, 0.72629987, 0.72695961,
0.72907016, 0.72952725, 0.73260065, 0.73553037, 0.74223549,
0.74764648, 0.74881536, 0.75298128, 0.75976866, 0.75990987,
0.76268508, 0.76287822, 0.766514  , 0.76814667, 0.77541254,
0.77667636, 0.78000203, 0.78502197, 0.78651408, 0.78684912,
0.79494148, 0.79848434, 0.80228855, 0.80485427, 0.8056745 ,
0.80638183, 0.80671752, 0.80794889, 0.80882712, 0.81621584,
0.81930391, 0.82023755, 0.82785384, 0.8282647 , 0.83005382,
0.83728053, 0.83813994, 0.83853873, 0.84430938, 0.84688197,
0.84764454, 0.84921788, 0.85083233, 0.85232688, 0.85643685,
0.86003014, 0.86004176, 0.8608426 , 0.86191448, 0.86540266,
0.87008488, 0.8782218 , 0.88538814, 0.88558038, 0.89178938,
```

```
0.89319651, 0.89734209, 0.89865245, 0.9007128 , 0.90102661,
0.90178276, 0.90324523, 0.90827515, 0.90889474, 0.90925129,
0.91212533, 0.91380576, 0.91990037, 0.92793323, 0.92866678,
0.93086134, 0.93188156, 0.9376778 , 0.94442947, 0.94475727,
0.94673358, 0.95926652, 0.96551891, 0.96554576, 0.96982546,
0.97057751, 0.97200691, 0.97305079, 0.97315653, 0.97444229,
0.97716526, 0.98104994, 0.98491947, 0.99460722, 0.99893939,
1.        ])
```

In [8]:
```
drug['source'].unique()
```

Out[8]:
```
array(['generated', 'training', 'hiv', 'manual'], dtype=object)
```

In [9]:
```
drug['gen'].unique()
```

Out[9]:
```
array([0], dtype=int64)
```

In [10]:
```
drug_sorted =  drug.sort_values(by = 'P_value', ascending = True)
drug_sorted
```

Out[10]:

	id	gen	smile	source	score	P_value	C
0	AAAA	0	COc1cccc(NC(=O)Cc2ccc(NC(=O)N3CCCC3)cc2)c1	generated	99.9	0.003320	
1	AAAB	0	C=CCNC(=O)CNc1cccc(C(=O)N(C)CCc2ccccc2)c1	generated	99.9	0.003384	
2	AAAC	0	CC(=O)Nc1ccc(S(=O)(=O)Nc2ccc(C)c(C)c2)cc1	training	99.9	0.003397	
3	AAAD	0	CCOC(=O)C1=C(C(=O)OCC)C(c2cccc(Cl)c2)NC(=O)N1	generated	99.9	0.003427	
4	AAAE	0	NC(=O)c1ccc(NC(=O)C(CC(=O)O)NC(=O)c2cc(-c3cccc...	generated	99.9	0.003468	
...	...	...	...	...	...	...	
772	ABDS	0	COc1ccc2c(c1)OC(CNC(=O)NCc1ccco1)CC2	generated	99.9	1.000000	
773	ABDT	0	COC(=O)C1COC(=O)N(c2ccc(NS(C)(=O)=O)cc2)C1	generated	99.9	1.000000	
774	ABDU	0	O=C(COc1ccc2cc(OCCCN3CCOCC3)ccc2c1)NCC(c1ccccc...	generated	99.9	1.000000	
767	ABDN	0	Cc1cccc(C)c1N(C)S(=O)(=O)c1cnn(C)c1	generated	99.9	1.000000	
1176	ABTG	0	COC(=O)NC(C(=O)NCCCCC(CO)N(CC(C)C)S(=O)(=O)c1c...	manual	99.9	1.000000	

1177 rows × 8 columns

In [11]:
```
drug_sorted = drug_sorted[['smile', 'P_value']]
drug_sorted
```

Out[11]:

	smile	P_value
0	COc1cccc(NC(=O)Cc2ccc(NC(=O)N3CCCC3)cc2)c1	0.003320
1	C=CCNC(=O)CNc1cccc(C(=O)N(C)CCc2ccccc2)c1	0.003384
2	CC(=O)Nc1ccc(S(=O)(=O)Nc2ccc(C)c(C)c2)cc1	0.003397
3	CCOC(=O)C1=C(C(=O)OCC)C(c2cccc(Cl)c2)NC(=O)N1	0.003427

	smile	P_value
4	NC(=O)c1ccc(NC(=O)C(CC(=O)O)NC(=O)c2cc(-c3cccc...	0.003468
...	...	...
772	COc1ccc2c(c1)OC(CNC(=O)NCc1ccco1)CC2	1.000000
773	COC(=O)C1COC(=O)N(c2ccc(NS(C)(=O)=O)cc2)C1	1.000000
774	O=C(COc1ccc2cc(OCCCN3CCOCC3)ccc2c1)NCC(c1ccccc...	1.000000
767	Cc1cccc(C)c1N(C)S(=O)(=O)c1cnn(C)c1	1.000000
1176	COC(=O)NC(C(=O)NCCCCC(CO)N(CC(C)C)S(=O)(=O)c1c...	1.000000

1177 rows × 2 columns

In [12]:
```
drug_sorted.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1177 entries, 0 to 1176
Data columns (total 2 columns):
 #   Column   Non-Null Count  Dtype  
---  ------   --------------  -----  
 0   smile    1177 non-null   object 
 1   P_value  1177 non-null   float64
dtypes: float64(1), object(1)
memory usage: 27.6+ KB
```

In [13]:
```
drug['source']=drug['source'].map({'generated':1,'training':0,'hiv':2,'manual':3})
```

In [14]:
```
drug
```

Out[14]:

	id	gen	smile	source	score	P_value	Connect
0	AAAA	0	COc1cccc(NC(=O)Cc2ccc(NC(=O)N3CCCC3)cc2)c1	1	99.9	0.003320	-0.37
1	AAAB	0	C=CCNC(=O)CNc1cccc(C(=O)N(C)CCc2ccccc2)c1	1	99.9	0.003384	-0.26
2	AAAC	0	CC(=O)Nc1ccc(S(=O)(=O)Nc2ccc(C)c(C)c2)cc1	0	99.9	0.003397	-0.31
3	AAAD	0	CCOC(=O)C1=C(C(=O)OCC)C(c2cccc(Cl)c2)NC(=O)N1	1	99.9	0.003427	-0.32
4	AAAE	0	NC(=O)c1ccc(NC(=O)C(CC(=O)O)NC(=O)c2cc(-c3cccc...	1	99.9	0.003468	-0.28
...	...	...	...	...	...	...	
1172	ABTC	0	CCCC1(CCc2ccccc2)CC(O)=C(C(CC)c2cccc(NS(=O)(=O...	3	99.9	1.000000	0.00
1173	ABTD	0	O=C1Nc2ccc(Cl)cc2C(C#CC2CC2)(C(F)(F)F)O1	3	99.9	1.000000	0.00
1174	ABTE	0	CC(C)(C)NC(=O)C1CN(Cc2cccnc2)CCN1CC(O)CC(Cc1cc...	3	99.9	1.000000	0.00
1175	ABTF	0	CCOP(=O)(COc1ccc(CC(NC(=O)OC2COC3OCCC23)C(O)CN...	3	99.9	1.000000	0.00
1176	ABTG	0	COC(=O)NC(C(=O)NCCCCC(CO)N(CC(C)C)S(=O)(=O)c1c...	3	99.9	1.000000	0.00

1177 rows × 8 columns

In [15]:
```
drug1=drug.drop(['id','gen','name'],axis=1)
drug1
```

Out[15]:

	smile	source	score	P_value	Connectivity
0	COc1cccc(NC(=O)Cc2ccc(NC(=O)N3CCCC3)cc2)c1	1	99.9	0.003320	-0.376625
1	C=CCNC(=O)CNc1cccc(C(=O)N(C)CCc2ccccc2)c1	1	99.9	0.003384	-0.269090
2	CC(=O)Nc1ccc(S(=O)(=O)Nc2ccc(C)c(C)c2)cc1	0	99.9	0.003397	-0.318895
3	CCOC(=O)C1=C(C(=O)OCC)C(c2cccc(Cl)c2)NC(=O)N1	1	99.9	0.003427	-0.329905
4	NC(=O)c1ccc(NC(=O)C(CC(=O)O)NC(=O)c2cc(-c3cccc...	1	99.9	0.003468	-0.288555
...	...	...	...	...	...
1172	CCCC1(CCc2ccccc2)CC(O)=C(C(CC)c2cccc(NS(=O)(=O...	3	99.9	1.000000	0.000000
1173	O=C1Nc2ccc(Cl)cc2C(C#CC2CC2)(C(F)(F)F)O1	3	99.9	1.000000	0.000000
1174	CC(C)(C)NC(=O)C1CN(Cc2cccnc2)CCN1CC(O)CC(Cc1cc...	3	99.9	1.000000	0.000000
1175	CCOP(=O)(COc1ccc(CC(NC(=O)OC2COC3OCCC23)C(O)CN...	3	99.9	1.000000	0.000000
1176	COC(=O)NC(C(=O)NCCCCC(CO)N(CC(C)C)S(=O)(=O)c1c...	3	99.9	1.000000	0.000000

1177 rows × 5 columns

In [17]:
```
drug1
```

Out[17]:

	smile	source	score	P_value	Connectivity
0	COc1cccc(NC(=O)Cc2ccc(NC(=O)N3CCCC3)cc2)c1	1	99.9	0.003320	-0.376625
1	C=CCNC(=O)CNc1cccc(C(=O)N(C)CCc2ccccc2)c1	1	99.9	0.003384	-0.269090
2	CC(=O)Nc1ccc(S(=O)(=O)Nc2ccc(C)c(C)c2)cc1	0	99.9	0.003397	-0.318895
3	CCOC(=O)C1=C(C(=O)OCC)C(c2cccc(Cl)c2)NC(=O)N1	1	99.9	0.003427	-0.329905
4	NC(=O)c1ccc(NC(=O)C(CC(=O)O)NC(=O)c2cc(-c3cccc...	1	99.9	0.003468	-0.288555
...	...	...	...	...	...
1172	CCCC1(CCc2ccccc2)CC(O)=C(C(CC)c2cccc(NS(=O)(=O...	3	99.9	1.000000	0.000000
1173	O=C1Nc2ccc(Cl)cc2C(C#CC2CC2)(C(F)(F)F)O1	3	99.9	1.000000	0.000000
1174	CC(C)(C)NC(=O)C1CN(Cc2cccnc2)CCN1CC(O)CC(Cc1cc...	3	99.9	1.000000	0.000000
1175	CCOP(=O)(COc1ccc(CC(NC(=O)OC2COC3OCCC23)C(O)CN...	3	99.9	1.000000	0.000000
1176	COC(=O)NC(C(=O)NCCCCC(CO)N(CC(C)C)S(=O)(=O)c1c...	3	99.9	1.000000	0.000000

1177 rows × 5 columns

In [18]:
```
drug1.isna().sum()
```

Out[18]:
```
smile           0
source          0
score           0
P_value         0
Connectivity    0
dtype: int64
```

In [23]:
```
x=drug1.drop(['smile', 'P_value'],axis=1)
y=drug1['P_value']
```

In [24]:
```
x
```

Out[24]:

	source	score	Connectivity
0	1	99.9	-0.376625
1	1	99.9	-0.269090
2	0	99.9	-0.318895
3	1	99.9	-0.329905
4	1	99.9	-0.288555
...	...	...	...
1172	3	99.9	0.000000
1173	3	99.9	0.000000
1174	3	99.9	0.000000
1175	3	99.9	0.000000
1176	3	99.9	0.000000

1177 rows × 3 columns

In [25]:
```
y
```

Out[25]:
```
0       0.003320
1       0.003384
2       0.003397
3       0.003427
4       0.003468
          ...
1172    1.000000
1173    1.000000
1174    1.000000
1175    1.000000
1176    1.000000
Name: P_value, Length: 1177, dtype: float64
```

In [26]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.30,random_state=42)
```

In [51]:
```
from sklearn.linear_model import LinearRegression
regressor=LinearRegression()
regressor.fit(x_train, y_train)
```

Out[51]:
```
▾ LinearRegression

LinearRegression()
```

In [54]:
```
y_pred1=regressor.predict(x_test)
y_pred1
```

Out[54]:
```
array([ 0.4792085 ,  0.73645914,  0.44729332,  0.73645914,  0.73645914,
        0.73645914,  0.73645914,  0.73645914,  0.47052954,  0.54183498,
        0.46870273,  0.50462075,  0.73645914,  0.73645914,  0.46724534,
        0.96909842,  0.56132863,  0.73645914,  0.98740331,  0.73645914,
        0.38314728,  0.51886069,  0.51886572,  0.73645914,  0.42158891,
        0.94517211,  0.97044837,  1.04037868,  0.95571984,  0.52452744,
        0.45530233,  0.73645914,  0.73645914,  0.44926283,  0.46404339,
        0.73645914,  0.73645914,  0.54334108,  0.50082277,  0.73645914,
        0.73645914,  0.57090419,  0.95277312,  0.64107794,  0.73645914,
        0.51679043,  0.73645914,  0.73645914,  0.73645914,  0.53255157,
        0.64107794,  0.49644551,  0.5045452 ,  0.98257271,  0.73645914,
        0.73645914,  0.73645914,  0.73645914,  0.83184034,  0.73645914,
        0.73645914,  0.73645914,  0.52508656,  0.73645914,  0.73645914,
        0.73645914,  0.73645914,  0.91823361,  0.73645914,  0.84751917,
        0.92460556,  0.73645914,  0.73645914,  0.46433389,  0.73645914,
        0.73645914,  0.51671487,  0.51224191,  0.73645914,  0.73645914,
        0.99499425,  0.94534338,  0.48778671,  0.94969041,  0.9707103 ,
        0.51177346,  0.40662025,  0.73645914,  0.73645914,  0.73645914,
        0.94381209,  0.73645914,  0.73645914,  0.73645914,  0.73645914,
        0.52552983,  0.54759744,  0.96161327,  0.73645914,  0.73645914,
        0.73645914,  0.43915837,  0.73645914,  0.73645914,  0.73645914,
        0.45683866,  0.73645914,  0.46081798,  0.73645914,  0.73645914,
        0.94979619,  0.40853773,  0.92722154,  0.51685088,  0.73645914,
        0.94446188,  0.54730025,  0.52515205,  0.73645914,  0.51154679,
        0.73645914,  0.73645914,  0.48728803,  0.96581926,  0.93918298,
        0.49885325,  0.96139163,  0.47794418,  0.73645914,  0.92703345,
        0.48527319,  0.73645914,  0.73645914,  0.54621224,  0.9807795 ,
        0.73645914,  0.73645914,  0.56129337,  0.48520267,  0.47579333,
        0.47254943,  0.73645914,  0.52556509,  0.88306611,  0.73645914,
        0.450779  ,  0.73645914,  1.0587289 ,  0.73645914,  0.73645914,
        0.95339773,  0.49856614,  0.73645914,  0.55751553,  0.48842138,
        0.83184034,  0.46947678,  0.73645914,  0.64107794,  0.73645914,
        0.73645914,  0.73645914,  0.73645914,  0.49485882,  0.55462422,
        0.73645914,  0.73645914,  0.73645914,  0.73645914,  0.41264802,
        0.44922923,  0.92722154,  0.73645914, -0.07887897,  0.73645914,
        0.73645914,  0.73645914,  0.73645914,  0.73645914,  0.73645914,
        0.93360185,  0.73645914,  0.73645914,  0.90923228,  0.53781536,
        0.73645914,  0.51156694,  0.9337983 ,  0.73645914,  0.73645914,
        0.55669448,  0.64107794,  1.01191392,  0.73645914,  0.73645914,
        0.73645914,  0.73645914,  0.545502  ,  0.73645914,  0.54906324,
        0.64107794,  0.42048578,  0.73645914,  0.73645914,  0.73645914,
        0.50107462,  0.73645914,  0.73645914,  0.52525279,  0.46433892,
        0.91042607,  0.73645914,  0.73645914,  0.73645914,  0.54689728,
        0.46459582,  0.73645914,  0.3996237 ,  0.49422414,  0.73645914,
        0.73645914,  0.49246115,  0.95949768,  0.9313553 ,  0.54105926,
        0.49426948,  0.4651499 ,  0.52360565,  0.73645914,  0.50987951,
        0.48183787,  0.73645914,  0.41524717,  0.92722154,  0.93672487,
        0.73645914,  0.73645914,  1.00077686,  0.53646038,  0.559344  ,
        0.93377815,  0.47260483,  0.73645914,  0.94553479,  0.73645914,
        1.02893434,  0.73645914,  0.98018512,  0.73645914,  0.99875194,
        0.73645914,  0.52377188,  0.97063474,  0.93013128,  1.06892906,
        0.73645914,  0.53384611,  0.73645914,  0.47154704,  0.73645914,
        0.73645914,  0.50492298,  0.41596244,  0.99759844,  0.95059205,
        0.94154539,  0.97490622,  0.73645914,  0.73645914,  0.40410507,
        0.53643015,  0.73645914,  0.73645914,  0.73645914,  0.52176207,
```

```
0.73645914,  0.73645914,  0.73645914,  0.47339566,  0.73645914,
0.73645914,  0.52731801,  0.73645914,  0.56646649,  0.73645914,
0.73645914,  0.4913278 ,  0.73645914,  0.73645914,  0.73645914,
0.52178221,  0.99907431,  0.46127636,  0.73645914,  0.47170823,
0.4141239 ,  0.73645914,  0.73645914,  0.73645914,  0.73645914,
0.73645914,  0.73645914,  0.92722154,  0.44845855,  0.73645914,
0.73645914,  0.55084638,  0.73645914,  0.5601449 ,  0.73645914,
0.51391927,  0.73645914,  0.44810933,  0.91208329,  0.3198156 ,
0.27815864,  0.50809133,  0.73645914,  0.95625377,  0.73645914,
0.50654493,  0.92920949,  0.43175381,  0.73645914,  0.94555997,
0.52610406,  0.73645914,  0.73645914,  0.47404545,  0.73645914,
0.73645914,  0.73645914,  0.53078354,  0.51814542,  0.42319237,
0.30838134,  0.94641628,  1.01619044,  0.95181607,  1.03537178,
0.45971485,  0.73645914,  0.38081344,  0.49300516,  0.73645914,
0.73645914,  0.73645914,  0.48277981,  0.73645914,  0.73645914,
0.92722154,  0.73645914,  0.73645914,  0.73645914])
```

In [55]:
```
from sklearn.metrics import r2_score
r2_score(y_test,y_pred1)
```

Out[55]:
```
0.07536894330655464
```

In [56]:
```
from sklearn.metrics import mean_squared_error
mean_squared_error(y_test,y_pred1)
```

Out[56]:
```
0.14430117799862377
```

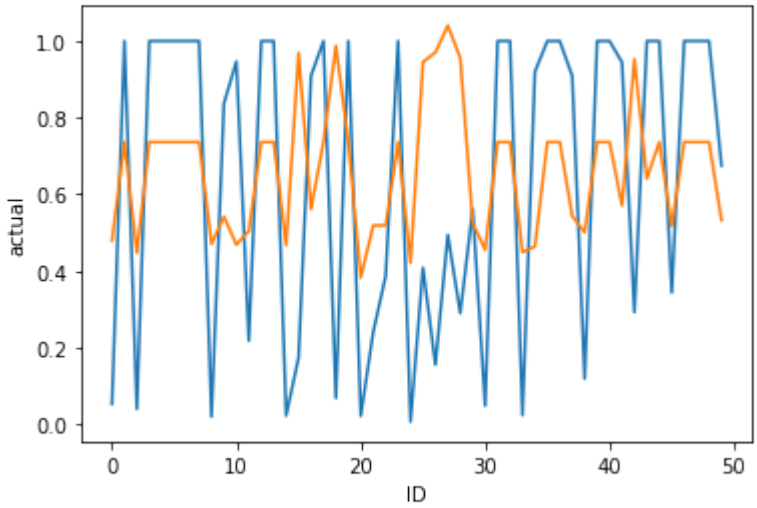
In [57]:
```
results=pd.DataFrame(columns=['actual','predicted'])
results['actual']=y_test
results['predicted']=y_pred1
results=results.reset_index()
results['ID']=results.index
results.head(5)
```

Out[57]:

	index	actual	predicted	ID
0	107	0.053120	0.479208	0
1	774	1.000000	0.736459	1
2	81	0.039904	0.447293	2
3	787	1.000000	0.736459	3
4	665	1.000000	0.736459	4

In [58]:
```
import seaborn as sns
import matplotlib.pyplot as plt
sns.lineplot(x='ID',y='actual',data=results.head(50))
sns.lineplot(x='ID',y='predicted',data=results.head(50))
plt.plot()
```

Out[58]:
```
[]
```



In [59]:
```
from sklearn.model_selection import GridSearchCV
from sklearn.ensemble import RandomForestRegressor
reg=RandomForestRegressor()
n_estimators=[25,50,75,100,125,150,175,200]
criterion=['squared_error']
max_depth=[3,5,10]
parameters={'n_estimators': n_estimators,'criterion':criterion,'max_depth':max_depth
rfc_reg = GridSearchCV(reg, parameters)
rfc_reg.fit(x_train,y_train)
```

Out[59]:
```
▸ GridSearchCV
▸ estimator: RandomForestRegressor
    ▸ RandomForestRegressor
```

In [34]:
```
rfc_reg.best_params_
```

Out[34]:
```
{'criterion': 'squared_error', 'max_depth': 3, 'n_estimators': 100}
```

In [60]:
```
reg=RandomForestRegressor(n_estimators=100,criterion='squared_error',max_depth=3)
```

In [61]:
```
reg.fit(x_train,y_train)
```

Out[61]:
```
▾ RandomForestRegressor
RandomForestRegressor(max_depth=3)
```

In [63]:
```
y_pred2=reg.predict(x_test)
y_pred2
```

Out[63]:
```
array([0.1105979 , 0.98309181, 0.07864243, 0.98309181, 0.98309181,
       0.98309181, 0.98309181, 0.98309181, 0.08299423, 0.68306458,
       0.9810374 , 0.18038375, 0.98309181, 0.98309181, 0.08213904,
       0.2403811 , 0.97888836, 0.98309181, 0.2403811 , 0.98309181,
       0.1105979 , 0.37034336, 0.37034336, 0.98309181, 0.07864243,
       0.45420826, 0.2403811 , 0.2403811 , 0.24385797, 0.39389183,
       0.08213904, 0.98309181, 0.98309181, 0.07864243, 0.97888836,
       0.98309181, 0.98309181, 0.68306458, 0.17107253, 0.98309181,
```

```
0.98309181, 0.98309181, 0.24385797, 0.98309181, 0.98309181,
0.34893965, 0.98309181, 0.98309181, 0.98309181, 0.54666365,
0.98309181, 0.16780143, 0.18038375, 0.2403811 , 0.98309181,
0.98309181, 0.98309181, 0.98309181, 0.98309181, 0.98309181,
0.98309181, 0.98309181, 0.39477277, 0.98309181, 0.98309181,
0.98309181, 0.98309181, 0.97623168, 0.98309181, 0.45639779,
0.97623168, 0.98309181, 0.98309181, 0.08213904, 0.98309181,
0.98309181, 0.30080766, 0.2395556 , 0.98309181, 0.98309181,
0.2403811 , 0.45420826, 0.15384059, 0.31964021, 0.2403811 ,
0.23492148, 0.20182701, 0.98309181, 0.98309181, 0.98309181,
0.45639779, 0.98309181, 0.98309181, 0.98309181, 0.98309181,
0.3942879 , 0.68306458, 0.24385797, 0.98309181, 0.98309181,
0.98309181, 0.07864243, 0.98309181, 0.98309181, 0.98309181,
0.08213904, 0.98309181, 0.08213904, 0.98309181, 0.98309181,
0.31964021, 0.07864243, 0.98309181, 0.34893965, 0.98309181,
0.45639779, 0.68306458, 0.39477277, 0.98309181, 0.23492148,
0.98309181, 0.98309181, 0.15384059, 0.24197299, 0.46376708,
0.16780143, 0.24385797, 0.1105979 , 0.98309181, 0.97623168,
0.15384059, 0.98309181, 0.98309181, 0.68306458, 0.2403811 ,
0.98309181, 0.98309181, 0.97888836, 0.15384059, 0.1105979 ,
0.08649073, 0.98309181, 0.3942879 , 0.2403811 , 0.98309181,
0.07864243, 0.98309181, 0.24505316, 0.98309181, 0.98309181,
0.24385797, 0.16780143, 0.98309181, 0.97354632, 0.15384059,
0.98309181, 0.08213904, 0.98309181, 0.98309181, 0.98309181,
0.98309181, 0.98309181, 0.98309181, 0.15623551, 0.92607113,
0.98309181, 0.98309181, 0.98309181, 0.98309181, 0.07864243,
0.68306458, 0.98309181, 0.98309181, 0.12908683, 0.98309181,
0.98309181, 0.98309181, 0.98309181, 0.98309181, 0.98309181,
0.84597302, 0.98309181, 0.98309181, 0.98309181, 0.66494968,
0.98309181, 0.23492148, 0.84597302, 0.98309181, 0.98309181,
0.97354632, 0.98309181, 0.2403811 , 0.98309181, 0.98309181,
0.98309181, 0.98309181, 0.68306458, 0.98309181, 0.70364411,
0.98309181, 0.07864243, 0.98309181, 0.98309181, 0.98309181,
0.17107253, 0.98309181, 0.98309181, 0.39477277, 0.08213904,
0.98088658, 0.98309181, 0.98309181, 0.98309181, 0.68306458,
0.08213904, 0.98309181, 0.15623551, 0.15623551, 0.98309181,
0.98309181, 0.15384059, 0.24385797, 0.92390784, 0.68306458,
0.15623551, 0.08213904, 0.38186201, 0.98309181, 0.18038375,
0.15280524, 0.98309181, 0.07864243, 0.98309181, 0.68060794,
0.98309181, 0.98309181, 0.2403811 , 0.62137309, 0.97888836,
0.84597302, 0.0911396 , 0.98309181, 0.45420826, 0.98309181,
0.2403811 , 0.98309181, 0.2403811 , 0.98309181, 0.2403811 ,
0.98309181, 0.39098956, 0.2403811 , 0.95361485, 0.24505316,
0.98309181, 0.5516065 , 0.98309181, 0.08427123, 0.98309181,
0.98309181, 0.18038375, 0.07864243, 0.2403811 , 0.25937109,
0.45639779, 0.2403811 , 0.98309181, 0.98309181, 0.07864243,
0.62137309, 0.98309181, 0.98309181, 0.98309181, 0.38010798,
0.98309181, 0.98309181, 0.98309181, 0.10638875, 0.98309181,
0.98309181, 0.4087991 , 0.98309181, 0.98309181, 0.98309181,
0.98309181, 0.15384059, 0.98309181, 0.98309181, 0.98309181,
0.38010798, 0.2403811 , 0.08213904, 0.98309181, 0.08649073,
0.07864243, 0.98309181, 0.98309181, 0.98309181, 0.98309181,
0.98309181, 0.98309181, 0.98309181, 0.68306458, 0.98309181,
0.98309181, 0.70989336, 0.98309181, 0.97888836, 0.98309181,
0.26472263, 0.98309181, 0.07864243, 0.98088658, 0.07864243,
0.11383012, 0.18038375, 0.98309181, 0.24385797, 0.98309181,
0.18038375, 0.95361485, 0.07864243, 0.98309181, 0.45420826,
0.3942879 , 0.98309181, 0.98309181, 0.10638875, 0.98309181,
0.98309181, 0.98309181, 0.42831935, 0.37076762, 0.36476196,
0.07864243, 0.45420826, 0.2403811 , 0.24610428, 0.2403811 ,
0.08213904, 0.98309181, 0.11383012, 0.15384059, 0.98309181,
0.98309181, 0.98309181, 0.15384059, 0.98309181, 0.98309181,
0.98309181, 0.98309181, 0.98309181, 0.98309181])
```

In [64]:
```
from sklearn.metrics import r2_score
r2_score(y_test,y_pred2)
```

Out[64]:
```
0.9381965864950121
```

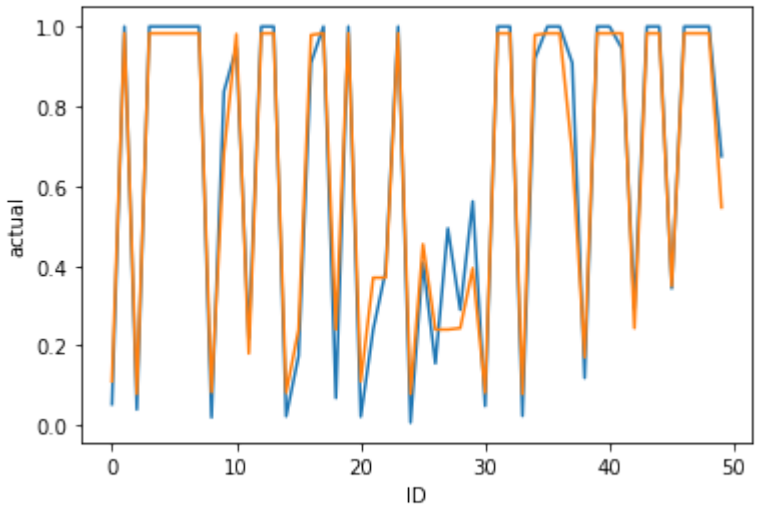
In [65]:
```
results=pd.DataFrame(columns=['actual','predicted'])
results['actual']=y_test
results['predicted']=y_pred2
results=results.reset_index()
results['ID']=results.index
results.head(5)
```

Out[65]:

	index	actual	predicted	ID
0	107	0.053120	0.110598	0
1	774	1.000000	0.983092	1
2	81	0.039904	0.078642	2
3	787	1.000000	0.983092	3
4	665	1.000000	0.983092	4

In [66]:
```
import seaborn as sns
import matplotlib.pyplot as plt
sns.lineplot(x='ID',y='actual',data=results.head(50))
sns.lineplot(x='ID',y='predicted',data=results.head(50))
plt.plot()
```

Out[66]:
```
[]
```



In [ ]:
```
```