In [5]:

```python
import pandas as pd
import numpy as np
import keras
import tensorflow
import seaborn as sns
```

In [6]:

```python
df = pd.read_csv("forestfires (1).csv")
```

In [7]:

```python
df
```

Out[7]:

	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain	...	monthfeb	monthjan	monthjul	monthjun	monthmar	monthmay	month
0	mar	fri	86.2	26.2	94.3	5.1	8.2	51	6.7	0.0	...	0	0	0	0	1	0	
1	oct	tue	90.6	35.4	669.1	6.7	18.0	33	0.9	0.0	...	0	0	0	0	0	0	
2	oct	sat	90.6	43.7	686.9	6.7	14.6	33	1.3	0.0	...	0	0	0	0	0	0	
3	mar	fri	91.7	33.3	77.5	9.0	8.3	97	4.0	0.2	...	0	0	0	0	1	0	
4	mar	sun	89.3	51.3	102.2	9.6	11.4	99	1.8	0.0	...	0	0	0	0	1	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
512	aug	sun	81.6	56.7	665.6	1.9	27.8	32	2.7	0.0	...	0	0	0	0	0	0	
513	aug	sun	81.6	56.7	665.6	1.9	21.9	71	5.8	0.0	...	0	0	0	0	0	0	
514	aug	sun	81.6	56.7	665.6	1.9	21.2	70	6.7	0.0	...	0	0	0	0	0	0	
515	aug	sat	94.4	146.0	614.7	11.3	25.6	42	4.0	0.0	...	0	0	0	0	0	0	
516	nov	tue	79.5	3.0	106.7	1.1	11.8	31	4.5	0.0	...	0	0	0	0	0	0	

517 rows × 31 columns

In [8]:

```python
df.isnull().sum()
```

Out[8]:

```
month            0
day              0
FFMC             0
DMC              0
DC               0
ISI              0
temp             0
RH               0
wind             0
rain             0
area             0
dayfri           0
daymon           0
daysat           0
daysun           0
daythu           0
daytue           0
daywed           0
monthapr         0
monthaug         0
monthdec         0
monthfeb         0
monthjan         0
monthjul         0
monthjun         0
monthmar         0
monthmay         0
monthnov         0
monthoct         0
monthsep         0
size_category    0
dtype: int64
```

In [9]:

```python
df.describe()
```

Out[9]:

	FFMC	DMC	DC	ISI	temp	RH	wind	rain	area	dayfri	...	monthde
count	517.000000	517.000000	517.000000	517.000000	517.000000	517.000000	517.000000	517.000000	517.000000	517.000000	...	517.00000
mean	90.644681	110.872340	547.940039	9.021663	18.889168	44.288201	4.017602	0.021663	12.847292	0.164410	...	0.01740
std	5.520111	64.046482	248.066192	4.559477	5.806625	16.317469	1.791653	0.295959	63.655818	0.371006	...	0.13091
min	18.700000	1.100000	7.900000	0.000000	2.200000	15.000000	0.400000	0.000000	0.000000	0.000000	...	0.00000

											...	
25%	90.200000	68.600000	437.700000	6.500000	15.500000	33.000000	2.700000	0.000000	0.000000	0.000000	...	0.00000
50%	91.600000	108.300000	664.200000	8.400000	19.300000	42.000000	4.000000	0.000000	0.520000	0.000000	...	0.00000
75%	92.900000	142.400000	713.900000	10.800000	22.800000	53.000000	4.900000	0.000000	6.570000	0.000000	...	0.00000
max	96.200000	291.300000	860.600000	56.100000	33.300000	100.000000	9.400000	6.400000	1090.840000	1.000000	...	1.00000

8 rows × 28 columns

In [10]:

```
x = df.iloc[:,:-1]
y = df.iloc[:,-1]
```

In [11]:

```
from sklearn.preprocessing import LabelEncoder
label_encoder_x=LabelEncoder()
x=x.apply(LabelEncoder().fit_transform)
x
```

Out[11]:

	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain	...	monthdec	monthfeb	monthjan	monthjul	monthjun	monthmar	monthmay
0	7	0	28	37	41	29	12	34	14	0	...	0	0	0	0	0	1	0
1	10	5	56	49	144	42	85	16	1	0	...	0	0	0	0	0	0	0
2	10	2	56	56	156	42	55	16	2	0	...	0	0	0	0	0	0	0
3	7	0	67	48	33	64	13	72	8	1	...	0	0	0	0	0	1	0
4	7	3	46	66	46	68	30	73	3	0	...	0	0	0	0	0	1	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
512	1	3	9	71	141	7	172	15	5	0	...	0	0	0	0	0	0	0
513	1	3	9	71	141	7	123	54	12	0	...	0	0	0	0	0	0	0
514	1	3	9	71	141	7	116	53	14	0	...	0	0	0	0	0	0	0
515	1	2	92	168	122	80	156	25	8	0	...	0	0	0	0	0	0	0
516	9	5	7	2	48	4	34	14	9	0	...	0	0	0	0	0	0	0

517 rows × 30 columns

In [12]:

```
y = pd.DataFrame(y)

label_encoder_y = LabelEncoder()
y = y.apply(LabelEncoder().fit_transform)
y
```

Out[12]:

	size_category
0	1
1	1
2	1
3	1
4	1
...	...
512	0
513	0
514	0
515	1
516	1

517 rows × 1 columns

In [13]:

```
seed = 7
np.random.seed(seed)
```

In [14]:

```
from keras.models import Sequential
from keras.layers import Dense
model = Sequential()
model.add(Dense(12, input_dim=30,  kernel_initializer='uniform', activation='relu'))
model.add(Dense(8,  kernel_initializer='uniform', activation='relu'))
model.add(Dense(1,  kernel_initializer='uniform', activation='sigmoid'))
```

In [15]:

```
model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])
```

In [16]:

```
history= model.fit(x, y, validation_split=0.33, epochs=150, batch_size=10)
```

```
Epoch 1/150
35/35 [==============================] - 2s 8ms/step - loss: 0.6521 - accuracy: 0.7543 - val_loss: 0.6009 - val_a
ccuracy: 0.6784
Epoch 2/150
35/35 [==============================] - 0s 2ms/step - loss: 0.4523 - accuracy: 0.7572 - val_loss: 0.4260 - val_a
ccuracy: 0.6784
Epoch 3/150
35/35 [==============================] - 0s 2ms/step - loss: 0.3066 - accuracy: 0.7572 - val_loss: 0.3767 - val_a
ccuracy: 0.6784
Epoch 4/150
35/35 [==============================] - 0s 2ms/step - loss: 0.2459 - accuracy: 0.7572 - val_loss: 0.3930 - val_a
ccuracy: 0.6784
Epoch 5/150
35/35 [==============================] - 0s 2ms/step - loss: 0.2252 - accuracy: 0.7572 - val_loss: 0.3377 - val_a
ccuracy: 0.6784
Epoch 6/150
35/35 [==============================] - 0s 2ms/step - loss: 0.2071 - accuracy: 0.9451 - val_loss: 0.2934 - val_a
ccuracy: 0.9064
Epoch 7/150
35/35 [==============================] - 0s 2ms/step - loss: 0.1316 - accuracy: 0.9769 - val_loss: 0.1367 - val_a
ccuracy: 0.9357
Epoch 8/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0798 - accuracy: 0.9711 - val_loss: 0.1236 - val_a
ccuracy: 0.9474
Epoch 9/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0637 - accuracy: 0.9827 - val_loss: 0.1633 - val_a
ccuracy: 0.9240
Epoch 10/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0656 - accuracy: 0.9740 - val_loss: 0.1714 - val_a
ccuracy: 0.9181
Epoch 11/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0565 - accuracy: 0.9827 - val_loss: 0.2052 - val_a
ccuracy: 0.9006
Epoch 12/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0608 - accuracy: 0.9769 - val_loss: 0.1479 - val_a
ccuracy: 0.9298
Epoch 13/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0525 - accuracy: 0.9855 - val_loss: 0.1223 - val_a
ccuracy: 0.9532
Epoch 14/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0555 - accuracy: 0.9798 - val_loss: 0.1566 - val_a
ccuracy: 0.9298
Epoch 15/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0535 - accuracy: 0.9769 - val_loss: 0.1190 - val_a
ccuracy: 0.9532
Epoch 16/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0493 - accuracy: 0.9798 - val_loss: 0.1973 - val_a
ccuracy: 0.8947
Epoch 17/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0494 - accuracy: 0.9827 - val_loss: 0.1725 - val_a
ccuracy: 0.9298
Epoch 18/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0500 - accuracy: 0.9740 - val_loss: 0.2139 - val_a
ccuracy: 0.8947
Epoch 19/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0518 - accuracy: 0.9769 - val_loss: 0.1286 - val_a
ccuracy: 0.9474
Epoch 20/150
35/35 [==============================] - 0s 8ms/step - loss: 0.0556 - accuracy: 0.9769 - val_loss: 0.1823 - val_a
ccuracy: 0.9240
Epoch 21/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0559 - accuracy: 0.9740 - val_loss: 0.1261 - val_a
ccuracy: 0.9474
Epoch 22/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0437 - accuracy: 0.9855 - val_loss: 0.1614 - val_a
ccuracy: 0.9298
Epoch 23/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0414 - accuracy: 0.9855 - val_loss: 0.1819 - val_a
ccuracy: 0.9240
Epoch 24/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0433 - accuracy: 0.9740 - val_loss: 0.1303 - val_a
ccuracy: 0.9415
Epoch 25/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0493 - accuracy: 0.9798 - val_loss: 0.1217 - val_a
ccuracy: 0.9357
Epoch 26/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0417 - accuracy: 0.9884 - val_loss: 0.2168 - val_a
ccuracy: 0.9181
Epoch 27/150
```

```
35/35 [==============================] - 0s 4ms/step - loss: 0.0370 - accuracy: 0.9855 - val_loss: 0.2496 - val_a
ccuracy: 0.8947
Epoch 28/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0372 - accuracy: 0.9855 - val_loss: 0.1174 - val_a
ccuracy: 0.9415
Epoch 29/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0392 - accuracy: 0.9855 - val_loss: 0.1425 - val_a
ccuracy: 0.9357
Epoch 30/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0407 - accuracy: 0.9827 - val_loss: 0.1582 - val_a
ccuracy: 0.9357
Epoch 31/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0425 - accuracy: 0.9740 - val_loss: 0.1814 - val_a
ccuracy: 0.9240
Epoch 32/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0369 - accuracy: 0.9827 - val_loss: 0.1425 - val_a
ccuracy: 0.9357
Epoch 33/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0346 - accuracy: 0.9884 - val_loss: 0.2348 - val_a
ccuracy: 0.9181
Epoch 34/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0349 - accuracy: 0.9827 - val_loss: 0.1452 - val_a
ccuracy: 0.9357
Epoch 35/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0363 - accuracy: 0.9827 - val_loss: 0.1958 - val_a
ccuracy: 0.9240
Epoch 36/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0453 - accuracy: 0.9769 - val_loss: 0.1409 - val_a
ccuracy: 0.9357
Epoch 37/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0388 - accuracy: 0.9855 - val_loss: 0.1500 - val_a
ccuracy: 0.9357
Epoch 38/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0308 - accuracy: 0.9855 - val_loss: 0.1297 - val_a
ccuracy: 0.9240
Epoch 39/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0379 - accuracy: 0.9798 - val_loss: 0.1590 - val_a
ccuracy: 0.9357
Epoch 40/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0326 - accuracy: 0.9855 - val_loss: 0.3729 - val_a
ccuracy: 0.8889
Epoch 41/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0415 - accuracy: 0.9855 - val_loss: 0.2880 - val_a
ccuracy: 0.9123
Epoch 42/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0454 - accuracy: 0.9855 - val_loss: 0.1311 - val_a
ccuracy: 0.9298
Epoch 43/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0379 - accuracy: 0.9855 - val_loss: 0.2282 - val_a
ccuracy: 0.9240
Epoch 44/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0336 - accuracy: 0.9827 - val_loss: 0.2277 - val_a
ccuracy: 0.9181
Epoch 45/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0468 - accuracy: 0.9769 - val_loss: 0.2082 - val_a
ccuracy: 0.9298
Epoch 46/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0467 - accuracy: 0.9769 - val_loss: 0.2208 - val_a
ccuracy: 0.9181
Epoch 47/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0277 - accuracy: 0.9855 - val_loss: 0.1352 - val_a
ccuracy: 0.9240
Epoch 48/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0465 - accuracy: 0.9769 - val_loss: 0.1383 - val_a
ccuracy: 0.9240
Epoch 49/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0333 - accuracy: 0.9884 - val_loss: 0.1660 - val_a
ccuracy: 0.9415
Epoch 50/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0282 - accuracy: 0.9884 - val_loss: 0.2921 - val_a
ccuracy: 0.9298
Epoch 51/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0339 - accuracy: 0.9827 - val_loss: 0.1605 - val_a
ccuracy: 0.9415
Epoch 52/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0258 - accuracy: 0.9884 - val_loss: 0.1489 - val_a
ccuracy: 0.9357
Epoch 53/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0291 - accuracy: 0.9913 - val_loss: 0.1772 - val_a
ccuracy: 0.9415
Epoch 54/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0309 - accuracy: 0.9855 - val_loss: 0.1555 - val_a
ccuracy: 0.9298
Epoch 55/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0198 - accuracy: 0.9913 - val_loss: 0.2885 - val_a
ccuracy: 0.9357
Epoch 56/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0219 - accuracy: 0.9884 - val_loss: 0.1624 - val_a
ccuracy: 0.9298
```

```
Epoch 57/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0275 - accuracy: 0.9884 - val_loss: 0.2140 - val_a
ccuracy: 0.9415
Epoch 58/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0220 - accuracy: 0.9913 - val_loss: 0.2770 - val_a
ccuracy: 0.9415
Epoch 59/150
35/35 [==============================] - ETA: 0s - loss: 0.0016 - accuracy: 1.00 - 0s 2ms/step - loss: 0.0207 - a
ccuracy: 0.9913 - val_loss: 0.1820 - val_accuracy: 0.9415
Epoch 60/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0336 - accuracy: 0.9913 - val_loss: 0.1984 - val_a
ccuracy: 0.9415
Epoch 61/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0236 - accuracy: 0.9942 - val_loss: 0.2534 - val_a
ccuracy: 0.9357
Epoch 62/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0214 - accuracy: 0.9913 - val_loss: 0.1874 - val_a
ccuracy: 0.9415
Epoch 63/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0209 - accuracy: 0.9942 - val_loss: 0.2404 - val_a
ccuracy: 0.9415
Epoch 64/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0239 - accuracy: 0.9913 - val_loss: 0.2705 - val_a
ccuracy: 0.9357
Epoch 65/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0270 - accuracy: 0.9913 - val_loss: 0.1910 - val_a
ccuracy: 0.9298
Epoch 66/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0171 - accuracy: 0.9942 - val_loss: 0.2956 - val_a
ccuracy: 0.9415
Epoch 67/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0231 - accuracy: 0.9913 - val_loss: 0.3214 - val_a
ccuracy: 0.9357
Epoch 68/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0244 - accuracy: 0.9913 - val_loss: 0.2492 - val_a
ccuracy: 0.9357
Epoch 69/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0296 - accuracy: 0.9884 - val_loss: 0.1944 - val_a
ccuracy: 0.9298
Epoch 70/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0292 - accuracy: 0.9913 - val_loss: 0.3320 - val_a
ccuracy: 0.9357
Epoch 71/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0213 - accuracy: 0.9971 - val_loss: 0.3088 - val_a
ccuracy: 0.9298
Epoch 72/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0203 - accuracy: 0.9942 - val_loss: 0.1981 - val_a
ccuracy: 0.9357
Epoch 73/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0235 - accuracy: 0.9884 - val_loss: 0.1997 - val_a
ccuracy: 0.9357
Epoch 74/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0278 - accuracy: 0.9913 - val_loss: 0.2020 - val_a
ccuracy: 0.9357
Epoch 75/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0250 - accuracy: 0.9942 - val_loss: 0.2918 - val_a
ccuracy: 0.9357
Epoch 76/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0305 - accuracy: 0.9913 - val_loss: 0.2090 - val_a
ccuracy: 0.9357
Epoch 77/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0174 - accuracy: 0.9942 - val_loss: 0.3706 - val_a
ccuracy: 0.9357
Epoch 78/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0188 - accuracy: 0.9913 - val_loss: 0.2231 - val_a
ccuracy: 0.9415
Epoch 79/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0186 - accuracy: 0.9942 - val_loss: 0.2943 - val_a
ccuracy: 0.9357
Epoch 80/150
35/35 [==============================] - 0s 1ms/step - loss: 0.0259 - accuracy: 0.9971 - val_loss: 0.2791 - val_a
ccuracy: 0.9357
Epoch 81/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0252 - accuracy: 0.9913 - val_loss: 0.2006 - val_a
ccuracy: 0.9240
Epoch 82/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0199 - accuracy: 0.9913 - val_loss: 0.2828 - val_a
ccuracy: 0.9357
Epoch 83/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0206 - accuracy: 0.9971 - val_loss: 0.2941 - val_a
ccuracy: 0.9357
Epoch 84/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0200 - accuracy: 0.9971 - val_loss: 0.4255 - val_a
ccuracy: 0.9240
Epoch 85/150
35/35 [==============================] - 0s 1ms/step - loss: 0.0259 - accuracy: 0.9884 - val_loss: 0.2284 - val_a
ccuracy: 0.9415
Epoch 86/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0185 - accuracy: 0.9971 - val_loss: 0.2181 - val_a
```

```
ccuracy: 0.9357
Epoch 87/150
35/35 [==============================] - 0s 1ms/step - loss: 0.0292 - accuracy: 0.9913 - val_loss: 0.2408 - val_a
ccuracy: 0.9474
Epoch 88/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0289 - accuracy: 0.9942 - val_loss: 0.3620 - val_a
ccuracy: 0.9357
Epoch 89/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0185 - accuracy: 0.9942 - val_loss: 0.2183 - val_a
ccuracy: 0.9298
Epoch 90/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0326 - accuracy: 0.9913 - val_loss: 0.2126 - val_a
ccuracy: 0.9298
Epoch 91/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0303 - accuracy: 0.9884 - val_loss: 0.2486 - val_a
ccuracy: 0.9474
Epoch 92/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0182 - accuracy: 0.9971 - val_loss: 0.3360 - val_a
ccuracy: 0.9298
Epoch 93/150
35/35 [==============================] - 0s 1ms/step - loss: 0.0188 - accuracy: 0.9913 - val_loss: 0.2662 - val_a
ccuracy: 0.9357
Epoch 94/150
35/35 [==============================] - 0s 1ms/step - loss: 0.0222 - accuracy: 0.9913 - val_loss: 0.2170 - val_a
ccuracy: 0.9298
Epoch 95/150
35/35 [==============================] - 0s 1ms/step - loss: 0.0211 - accuracy: 0.9942 - val_loss: 0.2514 - val_a
ccuracy: 0.9474
Epoch 96/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0271 - accuracy: 0.9855 - val_loss: 0.2064 - val_a
ccuracy: 0.9298
Epoch 97/150
35/35 [==============================] - 0s 1ms/step - loss: 0.0230 - accuracy: 0.9913 - val_loss: 0.2496 - val_a
ccuracy: 0.9474
Epoch 98/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0219 - accuracy: 0.9942 - val_loss: 0.2728 - val_a
ccuracy: 0.9357
Epoch 99/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0224 - accuracy: 0.9971 - val_loss: 0.4341 - val_a
ccuracy: 0.9357
Epoch 100/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0317 - accuracy: 0.9913 - val_loss: 0.3248 - val_a
ccuracy: 0.9357
Epoch 101/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0170 - accuracy: 0.9913 - val_loss: 0.3248 - val_a
ccuracy: 0.9357
Epoch 102/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0189 - accuracy: 0.9913 - val_loss: 0.2453 - val_a
ccuracy: 0.9474
Epoch 103/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0175 - accuracy: 0.9942 - val_loss: 0.2619 - val_a
ccuracy: 0.9474
Epoch 104/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0157 - accuracy: 0.9942 - val_loss: 0.3169 - val_a
ccuracy: 0.9357
Epoch 105/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0145 - accuracy: 0.9942 - val_loss: 0.2420 - val_a
ccuracy: 0.9415
Epoch 106/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0254 - accuracy: 0.9884 - val_loss: 0.2524 - val_a
ccuracy: 0.9474
Epoch 107/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0187 - accuracy: 0.9913 - val_loss: 0.2927 - val_a
ccuracy: 0.9357
Epoch 108/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0220 - accuracy: 0.9913 - val_loss: 0.3727 - val_a
ccuracy: 0.9298
Epoch 109/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0166 - accuracy: 0.9913 - val_loss: 0.2912 - val_a
ccuracy: 0.9357
Epoch 110/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0238 - accuracy: 0.9942 - val_loss: 0.4472 - val_a
ccuracy: 0.9298
Epoch 111/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0245 - accuracy: 0.9942 - val_loss: 0.3480 - val_a
ccuracy: 0.9298
Epoch 112/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0176 - accuracy: 0.9942 - val_loss: 0.3166 - val_a
ccuracy: 0.9357
Epoch 113/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0182 - accuracy: 0.9913 - val_loss: 0.3944 - val_a
ccuracy: 0.9298
Epoch 114/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0167 - accuracy: 0.9913 - val_loss: 0.3048 - val_a
ccuracy: 0.9357
Epoch 115/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0185 - accuracy: 0.9913 - val_loss: 0.2384 - val_a
ccuracy: 0.9298
Epoch 116/150
```

```
35/35 [==============================] - 0s 3ms/step - loss: 0.0224 - accuracy: 0.9913 - val_loss: 0.3073 - val_a
ccuracy: 0.9357
Epoch 117/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0186 - accuracy: 0.9913 - val_loss: 0.2760 - val_a
ccuracy: 0.9474
Epoch 118/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0182 - accuracy: 0.9942 - val_loss: 0.3096 - val_a
ccuracy: 0.9357
Epoch 119/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0167 - accuracy: 0.9913 - val_loss: 0.2875 - val_a
ccuracy: 0.9474
Epoch 120/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0224 - accuracy: 0.9913 - val_loss: 0.3591 - val_a
ccuracy: 0.9298
Epoch 121/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0158 - accuracy: 0.9971 - val_loss: 0.3825 - val_a
ccuracy: 0.9298
Epoch 122/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0198 - accuracy: 0.9913 - val_loss: 0.2685 - val_a
ccuracy: 0.9474
Epoch 123/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0228 - accuracy: 0.9942 - val_loss: 0.6179 - val_a
ccuracy: 0.9006
Epoch 124/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0299 - accuracy: 0.9884 - val_loss: 0.3901 - val_a
ccuracy: 0.9357
Epoch 125/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0323 - accuracy: 0.9855 - val_loss: 0.3976 - val_a
ccuracy: 0.9357
Epoch 126/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0217 - accuracy: 0.9913 - val_loss: 0.2489 - val_a
ccuracy: 0.9415
Epoch 127/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0256 - accuracy: 0.9913 - val_loss: 0.2698 - val_a
ccuracy: 0.9415
Epoch 128/150
35/35 [==============================] - 0s 4ms/step - loss: 0.0210 - accuracy: 0.9913 - val_loss: 0.2451 - val_a
ccuracy: 0.9415
Epoch 129/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0318 - accuracy: 0.9913 - val_loss: 0.4863 - val_a
ccuracy: 0.9123
Epoch 130/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0212 - accuracy: 0.9942 - val_loss: 0.3074 - val_a
ccuracy: 0.9357
Epoch 131/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0213 - accuracy: 0.9913 - val_loss: 0.2839 - val_a
ccuracy: 0.9474
Epoch 132/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0219 - accuracy: 0.9913 - val_loss: 0.2725 - val_a
ccuracy: 0.9474
Epoch 133/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0155 - accuracy: 0.9942 - val_loss: 0.4233 - val_a
ccuracy: 0.9298
Epoch 134/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0141 - accuracy: 0.9913 - val_loss: 0.2733 - val_a
ccuracy: 0.9415
Epoch 135/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0175 - accuracy: 0.9942 - val_loss: 0.3133 - val_a
ccuracy: 0.9357
Epoch 136/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0152 - accuracy: 0.9942 - val_loss: 0.2606 - val_a
ccuracy: 0.9415
Epoch 137/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0129 - accuracy: 0.9971 - val_loss: 0.3565 - val_a
ccuracy: 0.9357
Epoch 138/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0254 - accuracy: 0.9884 - val_loss: 0.2216 - val_a
ccuracy: 0.9240
Epoch 139/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0161 - accuracy: 0.9942 - val_loss: 0.3342 - val_a
ccuracy: 0.9298
Epoch 140/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0154 - accuracy: 0.9942 - val_loss: 0.3243 - val_a
ccuracy: 0.9298
Epoch 141/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0177 - accuracy: 0.9913 - val_loss: 0.3756 - val_a
ccuracy: 0.9357
Epoch 142/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0332 - accuracy: 0.9855 - val_loss: 0.2494 - val_a
ccuracy: 0.9357
Epoch 143/150
35/35 [==============================] - 0s 3ms/step - loss: 0.0205 - accuracy: 0.9913 - val_loss: 0.2486 - val_a
ccuracy: 0.9357
Epoch 144/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0214 - accuracy: 0.9913 - val_loss: 0.3850 - val_a
ccuracy: 0.9357
Epoch 145/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0178 - accuracy: 0.9942 - val_loss: 0.3130 - val_a
ccuracy: 0.9415
```

```
Epoch 146/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0164 - accuracy: 0.9942 - val_loss: 0.3321 - val_a
ccuracy: 0.9415
Epoch 147/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0182 - accuracy: 0.9913 - val_loss: 0.2463 - val_a
ccuracy: 0.9298
Epoch 148/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0264 - accuracy: 0.9884 - val_loss: 0.3168 - val_a
ccuracy: 0.9357
Epoch 149/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0249 - accuracy: 0.9913 - val_loss: 0.3840 - val_a
ccuracy: 0.9357
Epoch 150/150
35/35 [==============================] - 0s 2ms/step - loss: 0.0135 - accuracy: 0.9971 - val_loss: 0.3299 - val_a
ccuracy: 0.9357
```

In [17]:

```
scores = model.evaluate(x, y)
print("%s: %.2f%%" % (model.metrics_names[1], scores[1]*100))
```

```
17/17 [==============================] - 0s 921us/step - loss: 0.1168 - accuracy: 0.9749
accuracy: 97.49%
```

In [18]:

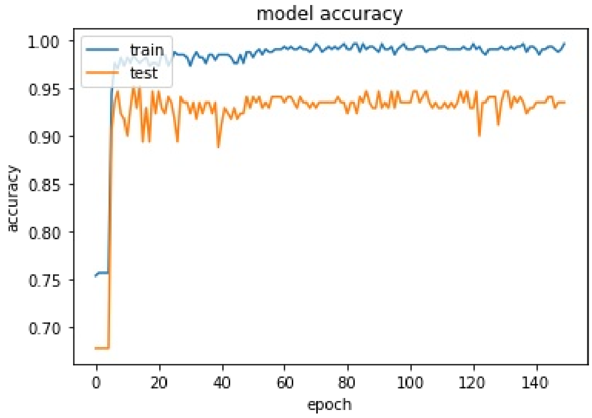
```
history.history.keys()
```

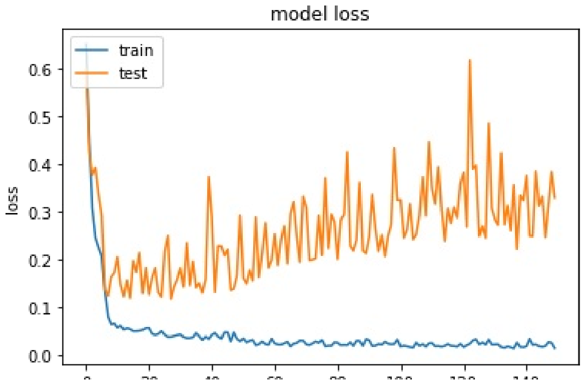
Out[18]:

```
dict_keys(['loss', 'accuracy', 'val_loss', 'val_accuracy'])
```

In [20]:

```
# summarize history for accuracy
import matplotlib.pyplot as plt
%matplotlib inline
plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])
plt.title('model accuracy')
plt.ylabel('accuracy')
plt.xlabel('epoch')
plt.legend(['train', 'test'], loc='upper left')
plt.show()
# summarize history for loss
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('model loss')
plt.ylabel('loss')
plt.xlabel('epoch')
plt.legend(['train', 'test'], loc='upper left')
plt.show()
```





In [24]:

```
from sklearn.preprocessing import StandardScaler
from keras.wrappers.scikit_learn import KerasClassifier
from sklearn.model_selection import GridSearchCV, KFold
#from keras.optimizers import Ada
```

In [25]:

```
a = StandardScaler()
a.fit(x)
X_standardized = a.transform(x)
```

In [26]:

```
pd.DataFrame(X_standardized).describe()
```

Out[26]:

	0	1	2	3	4	5	6	7	8	
count	5.170000e+02	5.170000e+02	5.170000e+02	5.170000e+02	5.170000e+02	5.170000e+02	5.170000e+02	5.170000e+02	5.170000e+02	5
mean	-4.874674e-17	5.110891e-17	-9.019220e-17	2.594099e-16	6.442300e-17	-8.718579e-17	-7.816657e-17	6.485249e-17	4.724353e-18	
std	1.000969e+00	1.000969e+00	1.000969e+00	1.000969e+00	1.000969e+00	1.000969e+00	1.000969e+00	1.000969e+00	1.000969e+00	1
min	-1.317959e+00	-1.423121e+00	-2.755520e+00	-2.134531e+00	-2.119754e+00	-2.133725e+00	-2.036890e+00	-1.713964e+00	-2.004018e+00	
25%	-1.089076e+00	-9.031536e-01	-5.025653e-01	-8.010724e-01	-7.605602e-01	-6.928003e-01	-7.181571e-01	-7.060079e-01	-7.499909e-01	
50%	5.533922e-02	1.367805e-01	1.039993e-01	1.234588e-01	1.959092e-01	-4.438437e-02	4.755898e-02	-1.390326e-01	2.425585e-03	
75%	1.199754e+00	6.567476e-01	6.672378e-01	8.168572e-01	7.999952e-01	6.400547e-01	7.494654e-01	5.539372e-01	5.040366e-01	
max	1.199754e+00	1.696682e+00	1.793715e+00	1.670271e+00	1.538322e+00	2.117002e+00	2.025659e+00	2.947833e+00	3.012092e+00	1

8 rows × 30 columns

In [28]:

```
from tensorflow.keras.optimizers import Adam, SGD, RMSprop
```

In [29]:

```
def create_model():
    model = Sequential()
    model.add(Dense(12, input_dim=30, kernel_initializer='uniform', activation='relu'))
    model.add(Dense(8, kernel_initializer='uniform', activation='relu'))
    model.add(Dense(1, kernel_initializer='uniform', activation='sigmoid'))\

    adam=Adam(lr=0.01)
    model.compile(loss='binary_crossentropy', optimizer=adam, metrics=['accuracy'])
    return model
model = KerasClassifier(build_fn = create_model,verbose = 0)
# Define the grid search parameters
batch_size = [10,20,40]
epochs = [10,50,100]
# Make a dictionary of the grid search parameters
param_grid = dict(batch_size = batch_size,epochs = epochs)
# Build and fit the GridSearchCV
grid = GridSearchCV(estimator = model,param_grid = param_grid,cv = KFold(),verbose = 10)
grid_result = grid.fit(X_standardized,y)
```

```
Fitting 5 folds for each of 9 candidates, totalling 45 fits
[CV 1/5; 1/9] START batch_size=10, epochs=10....................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 1/5; 1/9] END .....batch_size=10, epochs=10;, score=1.000 total time=   1.0s
[CV 2/5; 1/9] START batch_size=10, epochs=10....................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 2/5; 1/9] END .....batch_size=10, epochs=10;, score=0.942 total time=   0.8s
[CV 3/5; 1/9] START batch_size=10, epochs=10....................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
```

```
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 3/5; 1/9] END .....batch_size=10, epochs=10;, score=0.990 total time=   0.8s
[CV 4/5; 1/9] START batch_size=10, epochs=10...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 4/5; 1/9] END .....batch_size=10, epochs=10;, score=0.951 total time=   0.8s
[CV 5/5; 1/9] START batch_size=10, epochs=10...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 5/5; 1/9] END .....batch_size=10, epochs=10;, score=0.922 total time=   1.0s
[CV 1/5; 2/9] START batch_size=10, epochs=50...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 1/5; 2/9] END .....batch_size=10, epochs=50;, score=1.000 total time=   2.1s
[CV 2/5; 2/9] START batch_size=10, epochs=50...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 2/5; 2/9] END .....batch_size=10, epochs=50;, score=0.962 total time=   2.1s
[CV 3/5; 2/9] START batch_size=10, epochs=50...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 3/5; 2/9] END .....batch_size=10, epochs=50;, score=0.990 total time=   2.1s
[CV 4/5; 2/9] START batch_size=10, epochs=50...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 4/5; 2/9] END .....batch_size=10, epochs=50;, score=0.971 total time=   2.2s
[CV 5/5; 2/9] START batch_size=10, epochs=50...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 5/5; 2/9] END .....batch_size=10, epochs=50;, score=0.932 total time=   2.7s
[CV 1/5; 3/9] START batch_size=10, epochs=100..................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 1/5; 3/9] END ....batch_size=10, epochs=100;, score=1.000 total time=   3.8s
[CV 2/5; 3/9] START batch_size=10, epochs=100..................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 2/5; 3/9] END ....batch_size=10, epochs=100;, score=0.952 total time=   3.7s
[CV 3/5; 3/9] START batch_size=10, epochs=100..................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 3/5; 3/9] END ....batch_size=10, epochs=100;, score=0.971 total time=   4.4s
[CV 4/5; 3/9] START batch_size=10, epochs=100..................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
```

```
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 4/5; 3/9] END ....batch_size=10, epochs=100;, score=0.951 total time=   3.9s
[CV 5/5; 3/9] START batch_size=10, epochs=100..................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 5/5; 3/9] END ....batch_size=10, epochs=100;, score=0.951 total time=   4.7s
[CV 1/5; 4/9] START batch_size=20, epochs=10...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 1/5; 4/9] END .....batch_size=20, epochs=10;, score=1.000 total time=   0.6s
[CV 2/5; 4/9] START batch_size=20, epochs=10...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 2/5; 4/9] END .....batch_size=20, epochs=10;, score=0.952 total time=   0.9s
[CV 3/5; 4/9] START batch_size=20, epochs=10...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 3/5; 4/9] END .....batch_size=20, epochs=10;, score=0.981 total time=   0.7s
[CV 4/5; 4/9] START batch_size=20, epochs=10...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 4/5; 4/9] END .....batch_size=20, epochs=10;, score=0.942 total time=   0.7s
[CV 5/5; 4/9] START batch_size=20, epochs=10...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 5/5; 4/9] END .....batch_size=20, epochs=10;, score=0.932 total time=   1.0s
[CV 1/5; 5/9] START batch_size=20, epochs=50...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 1/5; 5/9] END .....batch_size=20, epochs=50;, score=1.000 total time=   1.2s
[CV 2/5; 5/9] START batch_size=20, epochs=50...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 2/5; 5/9] END .....batch_size=20, epochs=50;, score=0.904 total time=   1.4s
[CV 3/5; 5/9] START batch_size=20, epochs=50...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 3/5; 5/9] END .....batch_size=20, epochs=50;, score=0.981 total time=   1.3s
[CV 4/5; 5/9] START batch_size=20, epochs=50...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 4/5; 5/9] END .....batch_size=20, epochs=50;, score=0.961 total time=   1.5s
[CV 5/5; 5/9] START batch_size=20, epochs=50...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 5/5; 5/9] END .....batch_size=20, epochs=50;, score=0.932 total time=   1.4s
[CV 1/5; 6/9] START batch_size=20, epochs=100..................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 1/5; 6/9] END ....batch_size=20, epochs=100;, score=1.000 total time=   2.4s
[CV 2/5; 6/9] START batch_size=20, epochs=100...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 2/5; 6/9] END ....batch_size=20, epochs=100;, score=0.971 total time=   2.3s
[CV 3/5; 6/9] START batch_size=20, epochs=100...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 3/5; 6/9] END ....batch_size=20, epochs=100;, score=0.981 total time=   2.0s
[CV 4/5; 6/9] START batch_size=20, epochs=100...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 4/5; 6/9] END ....batch_size=20, epochs=100;, score=0.932 total time=   2.2s
[CV 5/5; 6/9] START batch_size=20, epochs=100...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 5/5; 6/9] END ....batch_size=20, epochs=100;, score=0.942 total time=   2.3s
[CV 1/5; 7/9] START batch_size=40, epochs=10....................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 1/5; 7/9] END .....batch_size=40, epochs=10;, score=1.000 total time=   1.3s
[CV 2/5; 7/9] START batch_size=40, epochs=10....................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 2/5; 7/9] END .....batch_size=40, epochs=10;, score=0.962 total time=   1.0s
[CV 3/5; 7/9] START batch_size=40, epochs=10....................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 3/5; 7/9] END .....batch_size=40, epochs=10;, score=0.981 total time=   0.6s
[CV 4/5; 7/9] START batch_size=40, epochs=10....................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
WARNING:tensorflow:5 out of the last 16 calls to <function Model.make_test_function.<locals>.test_function at 0x0
0000251B3644CA0> triggered tf.function retracing. Tracing is expensive and the excessive number of tracings could
be due to (1) creating @tf.function repeatedly in a loop, (2) passing tensors with different shapes, (3) passing
Python objects instead of tensors. For (1), please define your @tf.function outside of the loop. For (2), @tf.fun
ction has experimental_relax_shapes=True option that relaxes argument shapes that can avoid unnecessary retracing
. For (3), please refer to https://www.tensorflow.org/guide/function#controlling_retracing and https://www.tensor
flow.org/api_docs/python/tf/function for  more details.
[CV 4/5; 7/9] END .....batch_size=40, epochs=10;, score=0.932 total time=   0.6s
[CV 5/5; 7/9] START batch_size=40, epochs=10....................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
WARNING:tensorflow:5 out of the last 13 calls to <function Model.make_test_function.<locals>.test_function at 0x0
0000251B3763AF0> triggered tf.function retracing. Tracing is expensive and the excessive number of tracings could
be due to (1) creating @tf.function repeatedly in a loop, (2) passing tensors with different shapes, (3) passing
Python objects instead of tensors. For (1), please define your @tf.function outside of the loop. For (2), @tf.fun
ction has experimental_relax_shapes=True option that relaxes argument shapes that can avoid unnecessary retracing
. For (3), please refer to https://www.tensorflow.org/guide/function#controlling_retracing and https://www.tensor
flow.org/api_docs/python/tf/function for  more details.
[CV 5/5; 7/9] END .....batch_size=40, epochs=10;, score=0.883 total time=   0.7s
```

```
[CV 1/5; 8/9] START batch_size=40, epochs=50....................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 1/5; 8/9] END .....batch_size=40, epochs=50;, score=1.000 total time=   0.9s
[CV 2/5; 8/9] START batch_size=40, epochs=50....................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 2/5; 8/9] END .....batch_size=40, epochs=50;, score=0.942 total time=   1.0s
[CV 3/5; 8/9] START batch_size=40, epochs=50....................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 3/5; 8/9] END .....batch_size=40, epochs=50;, score=0.961 total time=   0.9s
[CV 4/5; 8/9] START batch_size=40, epochs=50....................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 4/5; 8/9] END .....batch_size=40, epochs=50;, score=0.922 total time=   1.0s
[CV 5/5; 8/9] START batch_size=40, epochs=50....................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 5/5; 8/9] END .....batch_size=40, epochs=50;, score=0.913 total time=   1.0s
[CV 1/5; 9/9] START batch_size=40, epochs=100...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 1/5; 9/9] END ....batch_size=40, epochs=100;, score=1.000 total time=   1.2s
[CV 2/5; 9/9] START batch_size=40, epochs=100...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 2/5; 9/9] END ....batch_size=40, epochs=100;, score=0.962 total time=   2.1s
[CV 3/5; 9/9] START batch_size=40, epochs=100...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 3/5; 9/9] END ....batch_size=40, epochs=100;, score=0.990 total time=   1.4s
[CV 4/5; 9/9] START batch_size=40, epochs=100...................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 4/5; 9/9] END ....batch_size=40, epochs=100;, score=0.951 total time=   1.5s
[CV 5/5; 9/9] START batch_size=40, epochs=100...................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 5/5; 9/9] END ....batch_size=40, epochs=100;, score=0.922 total time=   1.3s
```

In [30]:

```
print('Best : {}, using {}'.format(grid_result.best_score_,grid_result.best_params_))
means = grid_result.cv_results_['mean_test_score']
stds = grid_result.cv_results_['std_test_score']
params = grid_result.cv_results_['params']
for mean, stdev, param in zip(means, stds, params):
  print('{},{} with: {}'.format(mean, stdev, param))
```

```
Best : 0.9709484577178955, using {'batch_size': 10, 'epochs': 50}
0.9612770676612854,0.029374952296690925 with: {'batch_size': 10, 'epochs': 10}
0.9709484577178955,0.02375141024628043 with: {'batch_size': 10, 'epochs': 50}
0.9651418924331665,0.018959270800235323 with: {'batch_size': 10, 'epochs': 100}
0.9612583994865418,0.02528215974732713 with: {'batch_size': 20, 'epochs': 10}
0.9555265188217164,0.03423454311444474 with: {'batch_size': 20, 'epochs': 50}
0.9651045560836792,0.02503096129938652 with: {'batch_size': 20, 'epochs': 100}
0.9515309929847717,0.04074850959636234 with: {'batch_size': 40, 'epochs': 10}
0.9476848483085633,0.031047757361927466 with: {'batch_size': 40, 'epochs': 50}
0.9651232123374939,0.027859313612066533 with: {'batch_size': 40, 'epochs': 100}
```

In [31]:

```
from keras.layers import Dropout
def create_model(learning_rate,dropout_rate):
    model = Sequential()
    model.add(Dense(8,input_dim = 30,kernel_initializer = 'normal',activation = 'relu'))
    model.add(Dropout(dropout_rate))
    model.add(Dense(4,input_dim = 30,kernel_initializer = 'normal',activation = 'relu'))
    model.add(Dropout(dropout_rate))
    model.add(Dense(1,activation = 'sigmoid'))

    adam = Adam(lr = learning_rate)
    model.compile(loss = 'binary_crossentropy',optimizer = adam,metrics = ['accuracy'])
    return model
```

In [32]:

```
model = KerasClassifier(build_fn = create_model,verbose = 0,batch_size = 40,epochs = 10)

# Define the grid search parameters

learning_rate = [0.001,0.01,0.1]
dropout_rate = [0.0,0.1,0.2]

# Make a dictionary of the grid search parameters

param_grids = dict(learning_rate = learning_rate,dropout_rate = dropout_rate)

# Build and fit the GridSearchCV
```

```
grid = GridSearchCV(estimator = model,param_grid = param_grids,cv = KFold(),verbose = 10)
grid_result = grid.fit(X_standardized,y)
```

```
Fitting 5 folds for each of 9 candidates, totalling 45 fits
[CV 1/5; 1/9] START dropout_rate=0.0, learning_rate=0.001.......................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 1/5; 1/9] END dropout_rate=0.0, learning_rate=0.001;, score=1.000 total time=   0.8s
[CV 2/5; 1/9] START dropout_rate=0.0, learning_rate=0.001.......................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 2/5; 1/9] END dropout_rate=0.0, learning_rate=0.001;, score=0.750 total time=   0.6s
[CV 3/5; 1/9] START dropout_rate=0.0, learning_rate=0.001.......................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 3/5; 1/9] END dropout_rate=0.0, learning_rate=0.001;, score=0.524 total time=   0.6s
[CV 4/5; 1/9] START dropout_rate=0.0, learning_rate=0.001.......................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 4/5; 1/9] END dropout_rate=0.0, learning_rate=0.001;, score=0.680 total time=   0.5s
[CV 5/5; 1/9] START dropout_rate=0.0, learning_rate=0.001.......................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 5/5; 1/9] END dropout_rate=0.0, learning_rate=0.001;, score=0.699 total time=   0.6s
[CV 1/5; 2/9] START dropout_rate=0.0, learning_rate=0.01........................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 1/5; 2/9] END dropout_rate=0.0, learning_rate=0.01;, score=1.000 total time=   0.9s
[CV 2/5; 2/9] START dropout_rate=0.0, learning_rate=0.01........................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 2/5; 2/9] END dropout_rate=0.0, learning_rate=0.01;, score=0.962 total time=   0.5s
[CV 3/5; 2/9] START dropout_rate=0.0, learning_rate=0.01........................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 3/5; 2/9] END dropout_rate=0.0, learning_rate=0.01;, score=0.961 total time=   0.6s
[CV 4/5; 2/9] START dropout_rate=0.0, learning_rate=0.01........................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 4/5; 2/9] END dropout_rate=0.0, learning_rate=0.01;, score=0.922 total time=   0.5s
[CV 5/5; 2/9] START dropout_rate=0.0, learning_rate=0.01........................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 5/5; 2/9] END dropout_rate=0.0, learning_rate=0.01;, score=0.893 total time=   0.6s
[CV 1/5; 3/9] START dropout_rate=0.0, learning_rate=0.1.........................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 1/5; 3/9] END dropout_rate=0.0, learning_rate=0.1;, score=1.000 total time=   0.7s
[CV 2/5; 3/9] START dropout_rate=0.0, learning_rate=0.1.........................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 2/5; 3/9] END dropout_rate=0.0, learning_rate=0.1;, score=0.933 total time=   0.6s
[CV 3/5; 3/9] START dropout_rate=0.0, learning_rate=0.1.........................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 3/5; 3/9] END dropout_rate=0.0, learning_rate=0.1;, score=0.913 total time=   0.6s
[CV 4/5; 3/9] START dropout_rate=0.0, learning_rate=0.1.........................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 4/5; 3/9] END dropout_rate=0.0, learning_rate=0.1;, score=0.932 total time=   0.6s
[CV 5/5; 3/9] START dropout_rate=0.0, learning_rate=0.1.........................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 5/5; 3/9] END dropout_rate=0.0, learning_rate=0.1;, score=0.699 total time=   0.6s
[CV 1/5; 4/9] START dropout_rate=0.1, learning_rate=0.001.......................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 1/5; 4/9] END dropout_rate=0.1, learning_rate=0.001;, score=1.000 total time=   0.8s
[CV 2/5; 4/9] START dropout_rate=0.1, learning_rate=0.001.......................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 2/5; 4/9] END dropout_rate=0.1, learning_rate=0.001;, score=0.750 total time=   1.0s
[CV 3/5; 4/9] START dropout_rate=0.1, learning_rate=0.001.......................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 3/5; 4/9] END dropout_rate=0.1, learning_rate=0.001;, score=0.524 total time=   0.7s
[CV 4/5; 4/9] START dropout_rate=0.1, learning_rate=0.001.......................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 4/5; 4/9] END dropout_rate=0.1, learning_rate=0.001;, score=0.680 total time=   0.5s
[CV 5/5; 4/9] START dropout_rate=0.1, learning_rate=0.001.......................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 5/5; 4/9] END dropout_rate=0.1, learning_rate=0.001;, score=0.699 total time=   0.6s
[CV 1/5; 5/9] START dropout_rate=0.1, learning_rate=0.01........................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 1/5; 5/9] END dropout_rate=0.1, learning_rate=0.01;, score=1.000 total time=   0.6s
[CV 2/5; 5/9] START dropout_rate=0.1, learning_rate=0.01........................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 2/5; 5/9] END dropout_rate=0.1, learning_rate=0.01;, score=0.971 total time=   0.6s
[CV 3/5; 5/9] START dropout_rate=0.1, learning_rate=0.01........................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 3/5; 5/9] END dropout_rate=0.1, learning_rate=0.01;, score=1.000 total time=   0.5s
[CV 4/5; 5/9] START dropout_rate=0.1, learning_rate=0.01........................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 4/5; 5/9] END dropout_rate=0.1, learning_rate=0.01;, score=0.932 total time=   0.7s
[CV 5/5; 5/9] START dropout_rate=0.1, learning_rate=0.01........................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 5/5; 5/9] END dropout_rate=0.1, learning_rate=0.01;, score=0.903 total time=   0.8s
[CV 1/5; 6/9] START dropout_rate=0.1, learning_rate=0.1.........................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 1/5; 6/9] END dropout_rate=0.1, learning_rate=0.1;, score=1.000 total time=   1.1s
[CV 2/5; 6/9] START dropout_rate=0.1, learning_rate=0.1.........................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 2/5; 6/9] END dropout_rate=0.1, learning_rate=0.1;, score=0.923 total time=   1.4s
[CV 3/5; 6/9] START dropout_rate=0.1, learning_rate=0.1.........................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 3/5; 6/9] END dropout_rate=0.1, learning_rate=0.1;, score=0.932 total time=   0.8s
[CV 4/5; 6/9] START dropout_rate=0.1, learning_rate=0.1.........................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 4/5; 6/9] END dropout_rate=0.1, learning_rate=0.1;, score=0.951 total time=   0.6s
[CV 5/5; 6/9] START dropout_rate=0.1, learning_rate=0.1.........................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 5/5; 6/9] END dropout_rate=0.1, learning_rate=0.1;, score=0.922 total time=   0.6s
[CV 1/5; 7/9] START dropout_rate=0.2, learning_rate=0.001.......................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 1/5; 7/9] END dropout_rate=0.2, learning_rate=0.001;, score=1.000 total time=   0.8s
[CV 2/5; 7/9] START dropout_rate=0.2, learning_rate=0.001.......................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 2/5; 7/9] END dropout_rate=0.2, learning_rate=0.001;, score=0.913 total time=   0.7s
[CV 3/5; 7/9] START dropout_rate=0.2, learning_rate=0.001.......................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 3/5; 7/9] END dropout_rate=0.2, learning_rate=0.001;, score=0.524 total time=   0.7s
[CV 4/5; 7/9] START dropout_rate=0.2, learning_rate=0.001.......................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 4/5; 7/9] END dropout_rate=0.2, learning_rate=0.001;, score=0.777 total time=   0.6s
[CV 5/5; 7/9] START dropout_rate=0.2, learning_rate=0.001.......................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 5/5; 7/9] END dropout_rate=0.2, learning_rate=0.001;, score=0.699 total time=   0.6s
[CV 1/5; 8/9] START dropout_rate=0.2, learning_rate=0.01........................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 1/5; 8/9] END dropout_rate=0.2, learning_rate=0.01;, score=1.000 total time=   0.6s
[CV 2/5; 8/9] START dropout_rate=0.2, learning_rate=0.01........................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 2/5; 8/9] END dropout_rate=0.2, learning_rate=0.01;, score=0.962 total time=   1.2s
[CV 3/5; 8/9] START dropout_rate=0.2, learning_rate=0.01........................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 3/5; 8/9] END dropout_rate=0.2, learning_rate=0.01;, score=0.971 total time=   0.5s
[CV 4/5; 8/9] START dropout_rate=0.2, learning_rate=0.01........................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 4/5; 8/9] END dropout_rate=0.2, learning_rate=0.01;, score=0.961 total time=   0.6s
[CV 5/5; 8/9] START dropout_rate=0.2, learning_rate=0.01........................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 5/5; 8/9] END dropout_rate=0.2, learning_rate=0.01;, score=0.942 total time=   0.7s
[CV 1/5; 9/9] START dropout_rate=0.2, learning_rate=0.1.........................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 1/5; 9/9] END dropout_rate=0.2, learning_rate=0.1;, score=1.000 total time=   0.6s
[CV 2/5; 9/9] START dropout_rate=0.2, learning_rate=0.1.........................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 2/5; 9/9] END dropout_rate=0.2, learning_rate=0.1;, score=0.952 total time=   0.6s
[CV 3/5; 9/9] START dropout_rate=0.2, learning_rate=0.1.........................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 3/5; 9/9] END dropout_rate=0.2, learning_rate=0.1;, score=0.971 total time=   0.6s
[CV 4/5; 9/9] START dropout_rate=0.2, learning_rate=0.1.........................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 4/5; 9/9] END dropout_rate=0.2, learning_rate=0.1;, score=0.883 total time=   0.7s
[CV 5/5; 9/9] START dropout_rate=0.2, learning_rate=0.1.........................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 5/5; 9/9] END dropout_rate=0.2, learning_rate=0.1;, score=0.893 total time=   0.6s
```

In [33]:

```
def create_model(activation_function,init):
    model = Sequential()
    model.add(Dense(8,input_dim = 30,kernel_initializer = init,activation = activation_function))
    model.add(Dropout(0.1))
    model.add(Dense(4,input_dim = 30,kernel_initializer = init,activation = activation_function))
    model.add(Dropout(0.1))
    model.add(Dense(1,activation = 'sigmoid'))

    adam = Adam(lr = 0.001)
    model.compile(loss = 'binary_crossentropy',optimizer = adam,metrics = ['accuracy'])
    return model
```

In [34]:

```
model = KerasClassifier(build_fn = create_model,verbose = 0,batch_size = 40,epochs = 10)

# Define the grid search parameters
activation_function = ['softmax','relu','tanh','linear']
init = ['uniform','normal','zero']

# Make a dictionary of the grid search parameters
param_grids = dict(activation_function = activation_function,init = init)

# Build and fit the GridSearchCV

grid = GridSearchCV(estimator = model,param_grid = param_grids,cv = KFold(),verbose = 10)
grid_result = grid.fit(X_standardized,y)
```

```
Fitting 5 folds for each of 12 candidates, totalling 60 fits
[CV 1/5; 1/12] START activation_function=softmax, init=uniform.................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 1/5; 1/12] END activation_function=softmax, init=uniform;, score=1.000 total time=   1.2s
```

```
[CV 2/5; 1/12] START activation_function=softmax, init=uniform.................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 2/5; 1/12] END activation_function=softmax, init=uniform;, score=0.250 total time=   0.8s
[CV 3/5; 1/12] START activation_function=softmax, init=uniform.................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 3/5; 1/12] END activation_function=softmax, init=uniform;, score=0.524 total time=   0.7s
[CV 4/5; 1/12] START activation_function=softmax, init=uniform.................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 4/5; 1/12] END activation_function=softmax, init=uniform;, score=0.680 total time=   0.7s
[CV 5/5; 1/12] START activation_function=softmax, init=uniform.................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 5/5; 1/12] END activation_function=softmax, init=uniform;, score=0.699 total time=   0.6s
[CV 1/5; 2/12] START activation_function=softmax, init=normal..................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 1/5; 2/12] END activation_function=softmax, init=normal;, score=1.000 total time=   0.7s
[CV 2/5; 2/12] START activation_function=softmax, init=normal..................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 2/5; 2/12] END activation_function=softmax, init=normal;, score=0.750 total time=   0.7s
[CV 3/5; 2/12] START activation_function=softmax, init=normal..................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 3/5; 2/12] END activation_function=softmax, init=normal;, score=0.476 total time=   0.6s
[CV 4/5; 2/12] START activation_function=softmax, init=normal..................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 4/5; 2/12] END activation_function=softmax, init=normal;, score=0.320 total time=   0.7s
[CV 5/5; 2/12] START activation_function=softmax, init=normal..................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 5/5; 2/12] END activation_function=softmax, init=normal;, score=0.699 total time=   0.7s
[CV 1/5; 3/12] START activation_function=softmax, init=zero....................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 1/5; 3/12] END activation_function=softmax, init=zero;, score=1.000 total time=   1.0s
[CV 2/5; 3/12] START activation_function=softmax, init=zero....................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 2/5; 3/12] END activation_function=softmax, init=zero;, score=0.750 total time=   0.7s
[CV 3/5; 3/12] START activation_function=softmax, init=zero....................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 3/5; 3/12] END activation_function=softmax, init=zero;, score=0.524 total time=   0.6s
[CV 4/5; 3/12] START activation_function=softmax, init=zero....................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
```

```
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 4/5; 3/12] END activation_function=softmax, init=zero;, score=0.680 total time=   0.6s
[CV 5/5; 3/12] START activation_function=softmax, init=zero.....................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 5/5; 3/12] END activation_function=softmax, init=zero;, score=0.699 total time=   0.6s
[CV 1/5; 4/12] START activation_function=relu, init=uniform.....................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 1/5; 4/12] END activation_function=relu, init=uniform;, score=1.000 total time=   0.7s
[CV 2/5; 4/12] START activation_function=relu, init=uniform.....................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 2/5; 4/12] END activation_function=relu, init=uniform;, score=0.750 total time=   0.7s
[CV 3/5; 4/12] START activation_function=relu, init=uniform.....................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 3/5; 4/12] END activation_function=relu, init=uniform;, score=0.524 total time=   0.6s
[CV 4/5; 4/12] START activation_function=relu, init=uniform.....................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 4/5; 4/12] END activation_function=relu, init=uniform;, score=0.680 total time=   0.6s
[CV 5/5; 4/12] START activation_function=relu, init=uniform.....................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 5/5; 4/12] END activation_function=relu, init=uniform;, score=0.699 total time=   0.6s
[CV 1/5; 5/12] START activation_function=relu, init=normal......................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 1/5; 5/12] END activation_function=relu, init=normal;, score=1.000 total time=   0.9s
[CV 2/5; 5/12] START activation_function=relu, init=normal......................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 2/5; 5/12] END activation_function=relu, init=normal;, score=0.750 total time=   0.8s
[CV 3/5; 5/12] START activation_function=relu, init=normal......................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 3/5; 5/12] END activation_function=relu, init=normal;, score=0.524 total time=   0.7s
[CV 4/5; 5/12] START activation_function=relu, init=normal......................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 4/5; 5/12] END activation_function=relu, init=normal;, score=0.680 total time=   0.6s
[CV 5/5; 5/12] START activation_function=relu, init=normal......................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 5/5; 5/12] END activation_function=relu, init=normal;, score=0.699 total time=   0.6s
[CV 1/5; 6/12] START activation_function=relu, init=zero........................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
```

```
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 1/5; 6/12] END activation_function=relu, init=zero;, score=1.000 total time=   0.7s
[CV 2/5; 6/12] START activation_function=relu, init=zero........................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 2/5; 6/12] END activation_function=relu, init=zero;, score=0.750 total time=   0.7s
[CV 3/5; 6/12] START activation_function=relu, init=zero........................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 3/5; 6/12] END activation_function=relu, init=zero;, score=0.524 total time=   0.8s
[CV 4/5; 6/12] START activation_function=relu, init=zero........................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 4/5; 6/12] END activation_function=relu, init=zero;, score=0.680 total time=   0.7s
[CV 5/5; 6/12] START activation_function=relu, init=zero........................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 5/5; 6/12] END activation_function=relu, init=zero;, score=0.699 total time=   0.7s
[CV 1/5; 7/12] START activation_function=tanh, init=uniform.....................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 1/5; 7/12] END activation_function=tanh, init=uniform;, score=0.990 total time=   1.0s
[CV 2/5; 7/12] START activation_function=tanh, init=uniform.....................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 2/5; 7/12] END activation_function=tanh, init=uniform;, score=0.798 total time=   0.9s
[CV 3/5; 7/12] START activation_function=tanh, init=uniform.....................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 3/5; 7/12] END activation_function=tanh, init=uniform;, score=0.748 total time=   0.6s
[CV 4/5; 7/12] START activation_function=tanh, init=uniform.....................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 4/5; 7/12] END activation_function=tanh, init=uniform;, score=0.854 total time=   0.6s
[CV 5/5; 7/12] START activation_function=tanh, init=uniform.....................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 5/5; 7/12] END activation_function=tanh, init=uniform;, score=0.845 total time=   0.8s
[CV 1/5; 8/12] START activation_function=tanh, init=normal......................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 1/5; 8/12] END activation_function=tanh, init=normal;, score=0.962 total time=   0.9s
[CV 2/5; 8/12] START activation_function=tanh, init=normal......................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 2/5; 8/12] END activation_function=tanh, init=normal;, score=0.769 total time=   0.6s
[CV 3/5; 8/12] START activation_function=tanh, init=normal......................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 3/5; 8/12] END activation_function=tanh, init=normal;, score=0.767 total time=   0.8s
[CV 4/5; 8/12] START activation_function=tanh, init=normal......................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 4/5; 8/12] END activation_function=tanh, init=normal;, score=0.874 total time=   0.7s
[CV 5/5; 8/12] START activation_function=tanh, init=normal......................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 5/5; 8/12] END activation_function=tanh, init=normal;, score=0.777 total time=   0.7s
[CV 1/5; 9/12] START activation_function=tanh, init=zero........................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 1/5; 9/12] END activation_function=tanh, init=zero;, score=1.000 total time=   0.6s
[CV 2/5; 9/12] START activation_function=tanh, init=zero........................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 2/5; 9/12] END activation_function=tanh, init=zero;, score=0.750 total time=   1.2s
[CV 3/5; 9/12] START activation_function=tanh, init=zero........................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 3/5; 9/12] END activation_function=tanh, init=zero;, score=0.524 total time=   0.9s
[CV 4/5; 9/12] START activation_function=tanh, init=zero........................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 4/5; 9/12] END activation_function=tanh, init=zero;, score=0.680 total time=   0.8s
[CV 5/5; 9/12] START activation_function=tanh, init=zero........................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 5/5; 9/12] END activation_function=tanh, init=zero;, score=0.699 total time=   0.5s
[CV 1/5; 10/12] START activation_function=linear, init=uniform..................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 1/5; 10/12] END activation_function=linear, init=uniform;, score=0.971 total time=   0.6s
[CV 2/5; 10/12] START activation_function=linear, init=uniform..................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 2/5; 10/12] END activation_function=linear, init=uniform;, score=0.808 total time=   0.6s
[CV 3/5; 10/12] START activation_function=linear, init=uniform..................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 3/5; 10/12] END activation_function=linear, init=uniform;, score=0.699 total time=   0.8s
[CV 4/5; 10/12] START activation_function=linear, init=uniform..................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 4/5; 10/12] END activation_function=linear, init=uniform;, score=0.864 total time=   0.7s
[CV 5/5; 10/12] START activation_function=linear, init=uniform..................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 5/5; 10/12] END activation_function=linear, init=uniform;, score=0.864 total time=   0.6s
[CV 1/5; 11/12] START activation_function=linear, init=normal...................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 1/5; 11/12] END activation_function=linear, init=normal;, score=1.000 total time=   0.9s
[CV 2/5; 11/12] START activation_function=linear, init=normal...................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 2/5; 11/12] END activation_function=linear, init=normal;, score=0.837 total time=   0.9s
[CV 3/5; 11/12] START activation_function=linear, init=normal..................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 3/5; 11/12] END activation_function=linear, init=normal;, score=0.748 total time=   0.9s
[CV 4/5; 11/12] START activation_function=linear, init=normal..................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 4/5; 11/12] END activation_function=linear, init=normal;, score=0.825 total time=   0.7s
[CV 5/5; 11/12] START activation_function=linear, init=normal..................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 5/5; 11/12] END activation_function=linear, init=normal;, score=0.728 total time=   0.6s
[CV 1/5; 12/12] START activation_function=linear, init=zero....................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 1/5; 12/12] END activation_function=linear, init=zero;, score=1.000 total time=   0.5s
[CV 2/5; 12/12] START activation_function=linear, init=zero....................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 2/5; 12/12] END activation_function=linear, init=zero;, score=0.750 total time=   0.5s
[CV 3/5; 12/12] START activation_function=linear, init=zero....................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 3/5; 12/12] END activation_function=linear, init=zero;, score=0.524 total time=   0.5s
[CV 4/5; 12/12] START activation_function=linear, init=zero....................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 4/5; 12/12] END activation_function=linear, init=zero;, score=0.680 total time=   0.5s
[CV 5/5; 12/12] START activation_function=linear, init=zero....................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 5/5; 12/12] END activation_function=linear, init=zero;, score=0.699 total time=   0.5s
```

In [35]:

```
print('Best : {}, using {}'.format(grid_result.best_score_,grid_result.best_params_))
means = grid_result.cv_results_['mean_test_score']
stds = grid_result.cv_results_['std_test_score']
params = grid_result.cv_results_['params']
for mean, stdev, param in zip(means, stds, params):
    print('{},{} with: {}'.format(mean, stdev, param))
```

```
Best : 0.847012710571289, using {'activation_function': 'tanh', 'init': 'uniform'}
0.6305825233459472,0.24482772813004766 with: {'activation_function': 'softmax', 'init': 'uniform'}
0.6490291297435761,0.23405711374304639 with: {'activation_function': 'softmax', 'init': 'normal'}
0.7305825233459473,0.15435061319000673 with: {'activation_function': 'softmax', 'init': 'zero'}
0.7305825233459473,0.15435061319000673 with: {'activation_function': 'relu', 'init': 'uniform'}
0.7305825233459473,0.15435061319000673 with: {'activation_function': 'relu', 'init': 'normal'}
0.7305825233459473,0.15435061319000673 with: {'activation_function': 'relu', 'init': 'zero'}
0.847012710571289,0.0811149266298149 with: {'activation_function': 'tanh', 'init': 'uniform'}
0.8296489834785461,0.077101370840235 with: {'activation_function': 'tanh', 'init': 'normal'}
0.7305825233459473,0.15435061319000673 with: {'activation_function': 'tanh', 'init': 'zero'}
0.8412061333656311,0.08862290833905523 with: {'activation_function': 'linear', 'init': 'uniform'}
0.8275018572807312,0.09602161856687748 with: {'activation_function': 'linear', 'init': 'normal'}
0.7305825233459473,0.15435061319000673 with: {'activation_function': 'linear', 'init': 'zero'}
```

In [36]:

```
def create_model(neuron1,neuron2):
    model = Sequential()
    model.add(Dense(neuron1,input_dim = 30,kernel_initializer = 'uniform',activation = 'tanh'))
    model.add(Dropout(0.2))
    model.add(Dense(neuron2,input_dim = neuron1,kernel_initializer = 'uniform',activation = 'tanh'))
    model.add(Dropout(0.1))
    model.add(Dense(1,activation = 'sigmoid'))

    adam = Adam(lr = 0.001)
    model.compile(loss = 'binary_crossentropy',optimizer = adam,metrics = ['accuracy'])
    return model
```

In [37]:

```
model = KerasClassifier(build_fn = create_model,verbose = 0,batch_size = 40,epochs = 10)

# Define the grid search parameters

neuron1 = [4,8,16]
neuron2 = [2,4,8]

# Make a dictionary of the grid search parameters

param_grids = dict(neuron1 = neuron1,neuron2 = neuron2)

# Build and fit the GridSearchCV

grid = GridSearchCV(estimator = model,param_grid = param_grids,cv = KFold(),verbose = 10)
grid_result = grid.fit(X_standardized,y)
```

```
Fitting 5 folds for each of 9 candidates, totalling 45 fits
[CV 1/5; 1/9] START neuron1=4, neuron2=2........................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 1/5; 1/9] END .........neuron1=4, neuron2=2;, score=1.000 total time=   1.0s
[CV 2/5; 1/9] START neuron1=4, neuron2=2........................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 2/5; 1/9] END .........neuron1=4, neuron2=2;, score=0.750 total time=   0.7s
[CV 3/5; 1/9] START neuron1=4, neuron2=2........................................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 3/5; 1/9] END .........neuron1=4, neuron2=2;, score=0.592 total time=   1.0s
[CV 4/5; 1/9] START neuron1=4, neuron2=2........................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 4/5; 1/9] END .........neuron1=4, neuron2=2;, score=0.709 total time=   0.7s
[CV 5/5; 1/9] START neuron1=4, neuron2=2........................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 5/5; 1/9] END .........neuron1=4, neuron2=2;, score=0.728 total time=   0.8s
[CV 1/5; 2/9] START neuron1=4, neuron2=4........................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 1/5; 2/9] END .........neuron1=4, neuron2=4;, score=1.000 total time=   0.6s
[CV 2/5; 2/9] START neuron1=4, neuron2=4........................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 2/5; 2/9] END .........neuron1=4, neuron2=4;, score=0.769 total time=   0.6s
```

```
[CV 2/5; 2/9] END .........neuron1=4, neuron2=4;, score=0.760 total time=   0.6s
[CV 3/5; 2/9] START neuron1=4, neuron2=4........................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 3/5; 2/9] END .........neuron1=4, neuron2=4;, score=0.680 total time=   0.5s
[CV 4/5; 2/9] START neuron1=4, neuron2=4........................................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 4/5; 2/9] END .........neuron1=4, neuron2=4;, score=0.777 total time=   0.7s
[CV 5/5; 2/9] START neuron1=4, neuron2=4........................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 5/5; 2/9] END .........neuron1=4, neuron2=4;, score=0.699 total time=   1.3s
[CV 1/5; 3/9] START neuron1=4, neuron2=8........................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 1/5; 3/9] END .........neuron1=4, neuron2=8;, score=1.000 total time=   0.6s
[CV 2/5; 3/9] START neuron1=4, neuron2=8........................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 2/5; 3/9] END .........neuron1=4, neuron2=8;, score=0.750 total time=   0.8s
[CV 3/5; 3/9] START neuron1=4, neuron2=8........................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 3/5; 3/9] END .........neuron1=4, neuron2=8;, score=0.699 total time=   0.7s
[CV 4/5; 3/9] START neuron1=4, neuron2=8........................................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 4/5; 3/9] END .........neuron1=4, neuron2=8;, score=0.757 total time=   0.6s
[CV 5/5; 3/9] START neuron1=4, neuron2=8........................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 5/5; 3/9] END .........neuron1=4, neuron2=8;, score=0.709 total time=   0.6s
[CV 1/5; 4/9] START neuron1=8, neuron2=2........................................


C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 1/5; 4/9] END .........neuron1=8, neuron2=2;, score=0.990 total time=   0.8s
[CV 2/5; 4/9] START neuron1=8, neuron2=2........................................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 2/5; 4/9] END .........neuron1=8, neuron2=2;, score=0.788 total time=   0.6s
[CV 3/5; 4/9] START neuron1=8, neuron2=2........................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 3/5; 4/9] END .........neuron1=8, neuron2=2;, score=0.718 total time=   0.5s
[CV 4/5; 4/9] START neuron1=8, neuron2=2........................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 4/5; 4/9] END .........neuron1=8, neuron2=2;, score=0.825 total time=   0.6s
[CV 5/5; 4/9] START neuron1=8, neuron2=2........................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 5/5; 4/9] END .........neuron1=8, neuron2=2;, score=0.796 total time=   1.0s
[CV 1/5; 5/9] START neuron1=8, neuron2=4........................................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 1/5; 5/9] END .........neuron1=8, neuron2=4;, score=1.000 total time=   0.8s
[CV 2/5; 5/9] START neuron1=8, neuron2=4........................................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(


[CV 2/5; 5/9] END .........neuron1=8, neuron2=4;, score=0.817 total time=   0.9s
[CV 3/5; 5/9] START neuron1=8, neuron2=4......................................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 3/5; 5/9] END .........neuron1=8, neuron2=4;, score=0.728 total time=   0.7s
[CV 4/5; 5/9] START neuron1=8, neuron2=4......................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 4/5; 5/9] END .........neuron1=8, neuron2=4;, score=0.835 total time=   0.7s
[CV 5/5; 5/9] START neuron1=8, neuron2=4......................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 5/5; 5/9] END .........neuron1=8, neuron2=4;, score=0.825 total time=   0.7s
[CV 1/5; 6/9] START neuron1=8, neuron2=8......................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 1/5; 6/9] END .........neuron1=8, neuron2=8;, score=0.990 total time=   0.7s
[CV 2/5; 6/9] START neuron1=8, neuron2=8......................................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 2/5; 6/9] END .........neuron1=8, neuron2=8;, score=0.817 total time=   0.9s
[CV 3/5; 6/9] START neuron1=8, neuron2=8......................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 3/5; 6/9] END .........neuron1=8, neuron2=8;, score=0.825 total time=   1.0s
[CV 4/5; 6/9] START neuron1=8, neuron2=8......................................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 4/5; 6/9] END .........neuron1=8, neuron2=8;, score=0.874 total time=   0.9s
[CV 5/5; 6/9] START neuron1=8, neuron2=8......................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 5/5; 6/9] END .........neuron1=8, neuron2=8;, score=0.864 total time=   1.1s
[CV 1/5; 7/9] START neuron1=16, neuron2=2.....................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 1/5; 7/9] END ........neuron1=16, neuron2=2;, score=1.000 total time=   0.8s
[CV 2/5; 7/9] START neuron1=16, neuron2=2.....................................

C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 2/5; 7/9] END ........neuron1=16, neuron2=2;, score=0.837 total time=   0.9s
[CV 3/5; 7/9] START neuron1=16, neuron2=2.....................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 3/5; 7/9] END ........neuron1=16, neuron2=2;, score=0.786 total time=   0.7s
[CV 4/5; 7/9] START neuron1=16, neuron2=2.....................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 4/5; 7/9] END ........neuron1=16, neuron2=2;, score=0.883 total time=   0.9s
[CV 5/5; 7/9] START neuron1=16, neuron2=2.....................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(

[CV 5/5; 7/9] END ........neuron1=16, neuron2=2;, score=0.854 total time=   0.8s
[CV 1/5; 8/9] START neuron1=16, neuron2=4.....................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
[CV 1/5; 8/9] END ........neuron1=16, neuron2=4;, score=1.000 total time=   0.7s
[CV 2/5; 8/9] START neuron1=16, neuron2=4.....................................
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
```

```
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 2/5; 8/9] END ........neuron1=16, neuron2=4;, score=0.827 total time=   0.8s
[CV 3/5; 8/9] START neuron1=16, neuron2=4.......................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 3/5; 8/9] END ........neuron1=16, neuron2=4;, score=0.864 total time=   0.9s
[CV 4/5; 8/9] START neuron1=16, neuron2=4.......................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 4/5; 8/9] END ........neuron1=16, neuron2=4;, score=0.883 total time=   1.0s
[CV 5/5; 8/9] START neuron1=16, neuron2=4.......................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 5/5; 8/9] END ........neuron1=16, neuron2=4;, score=0.864 total time=   2.9s
[CV 1/5; 9/9] START neuron1=16, neuron2=8.......................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 1/5; 9/9] END ........neuron1=16, neuron2=8;, score=1.000 total time=   1.6s
[CV 2/5; 9/9] START neuron1=16, neuron2=8.......................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 2/5; 9/9] END ........neuron1=16, neuron2=8;, score=0.875 total time=   2.2s
[CV 3/5; 9/9] START neuron1=16, neuron2=8.......................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 3/5; 9/9] END ........neuron1=16, neuron2=8;, score=0.893 total time=   2.0s
[CV 4/5; 9/9] START neuron1=16, neuron2=8.......................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 4/5; 9/9] END ........neuron1=16, neuron2=8;, score=0.913 total time=   1.1s
[CV 5/5; 9/9] START neuron1=16, neuron2=8.......................................
```

```
C:\Users\rajesh\anaconda3\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argumen
t is deprecated, use `learning_rate` instead.
  warnings.warn(
```

```
[CV 5/5; 9/9] END ........neuron1=16, neuron2=8;, score=0.903 total time=   1.3s
```

In [38]:

```
print('Best : {}, using {}'.format(grid_result.best_score_,grid_result.best_params_))
means = grid_result.cv_results_['mean_test_score']
stds = grid_result.cv_results_['std_test_score']
params = grid_result.cv_results_['params']
for mean, stdev, param in zip(means, stds, params):
  print('{},{} with: {}'.format(mean, stdev, param))
```

```
Best : 0.9167475700378418, using {'neuron1': 16, 'neuron2': 8}
0.7558252334594726,0.13371650171026972 with: {'neuron1': 4, 'neuron2': 2}
0.7829910278320312,0.1143816453043283 with: {'neuron1': 4, 'neuron2': 4}
0.7830097079277039,0.11082004291898047 with: {'neuron1': 4, 'neuron2': 8}
0.8237304091453552,0.09040687424442402 with: {'neuron1': 8, 'neuron2': 2}
0.8411314368247986,0.08816012538426918 with: {'neuron1': 8, 'neuron2': 4}
0.874159824848175,0.06202426039127104 with: {'neuron1': 8, 'neuron2': 8}
0.8721620559692382,0.07128872944106017 with: {'neuron1': 16, 'neuron2': 2}
0.8877147197723388,0.05905731439446852 with: {'neuron1': 16, 'neuron2': 4}
0.9167475700378418,0.043443214944303685 with: {'neuron1': 16, 'neuron2': 8}
```

In [39]:

```
from sklearn.metrics import classification_report, accuracy_score

def create_model():
    model = Sequential()
    model.add(Dense(16,input_dim = 30,kernel_initializer = 'normal',activation = 'linear'))
    model.add(Dropout(0.1))
    model.add(Dense(8,input_dim = 30,kernel_initializer = 'normal',activation = 'linear'))
    model.add(Dropout(0.1))
    model.add(Dense(1,activation = 'linear'))

    adam = Adam(lr = 0.01) #sgd = SGD(lr=learning_rate, momentum=momentum, decay=decay_rate, nesterov=False)
    model.compile(loss = 'binary_crossentropy',optimizer = adam,metrics = ['accuracy'])
    return model
```

In [40]:

```
model = KerasClassifier(build_fn = create_model,verbose = 0,batch_size = 40,epochs = 100)

# Fitting the model

model.fit(X_standardized,y)

# Predicting using trained model

y_predict = model.predict(X_standardized)

# Printing the metrics
print(accuracy_score(y,y_predict))
```

```
0.9845261121856866
```

In [ ]: