In [105…
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

In [ ]:
```
```

In [116…
```
X=np.linspace(-5.0,5.0,100)
y=np.sqrt(10**2-X**2)
y=np.hstack([y,-y])
x=np.hstack([X,-X])
```

In [118…
```
x
```

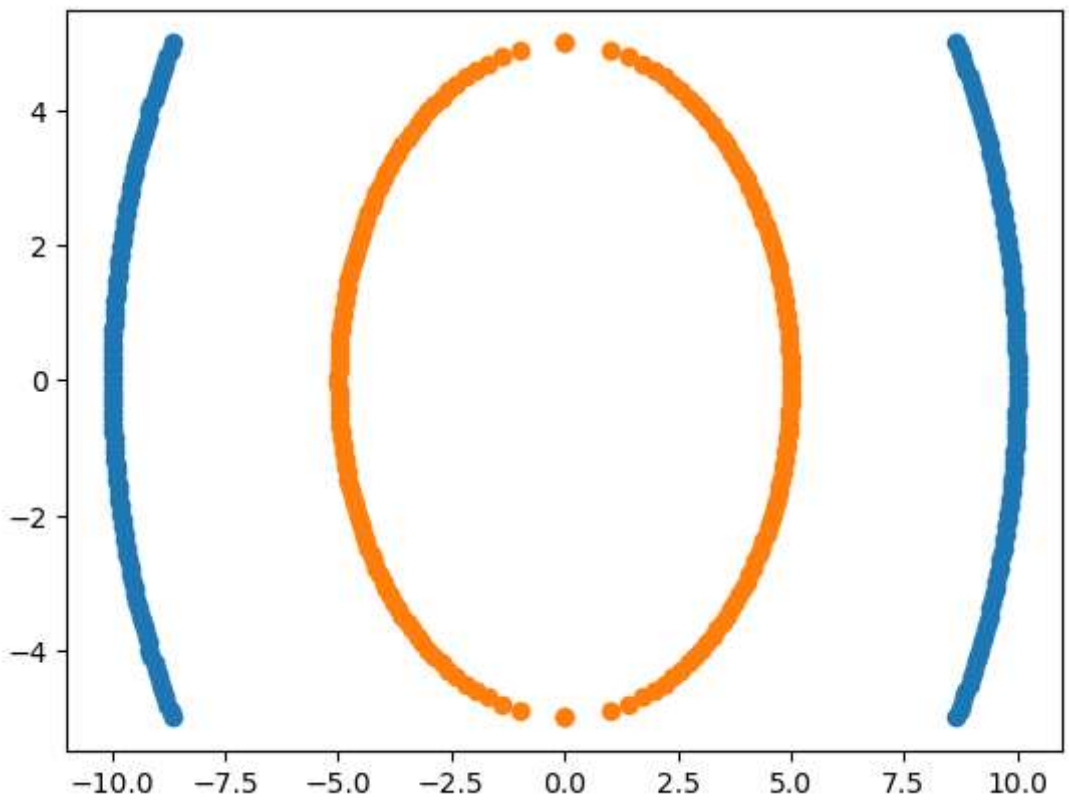
Out[118]:
```
array([-5.        , -4.8989899 , -4.7979798 , -4.6969697 , -4.5959596 ,
       -4.49494949, -4.39393939, -4.29292929, -4.19191919, -4.09090909,
       -3.98989899, -3.88888889, -3.78787879, -3.68686869, -3.58585859,
       -3.48484848, -3.38383838, -3.28282828, -3.18181818, -3.08080808,
       -2.97979798, -2.87878788, -2.77777778, -2.67676768, -2.57575758,
       -2.47474747, -2.37373737, -2.27272727, -2.17171717, -2.07070707,
       -1.96969697, -1.86868687, -1.76767677, -1.66666667, -1.56565657,
       -1.46464646, -1.36363636, -1.26262626, -1.16161616, -1.06060606,
       -0.95959596, -0.85858586, -0.75757576, -0.65656566, -0.55555556,
       -0.45454545, -0.35353535, -0.25252525, -0.15151515, -0.05050505,
        0.05050505,  0.15151515,  0.25252525,  0.35353535,  0.45454545,
        0.55555556,  0.65656566,  0.75757576,  0.85858586,  0.95959596,
        1.06060606,  1.16161616,  1.26262626,  1.36363636,  1.46464646,
        1.56565657,  1.66666667,  1.76767677,  1.86868687,  1.96969697,
        2.07070707,  2.17171717,  2.27272727,  2.37373737,  2.47474747,
        2.57575758,  2.67676768,  2.77777778,  2.87878788,  2.97979798,
        3.08080808,  3.18181818,  3.28282828,  3.38383838,  3.48484848,
        3.58585859,  3.68686869,  3.78787879,  3.88888889,  3.98989899,
        4.09090909,  4.19191919,  4.29292929,  4.39393939,  4.49494949,
        4.5959596 ,  4.6969697 ,  4.7979798 ,  4.8989899 ,  5.        ,
        5.        ,  4.8989899 ,  4.7979798 ,  4.6969697 ,  4.5959596 ,
        4.49494949,  4.39393939,  4.29292929,  4.19191919,  4.09090909,
        3.98989899,  3.88888889,  3.78787879,  3.68686869,  3.58585859,
        3.48484848,  3.38383838,  3.28282828,  3.18181818,  3.08080808,
        2.97979798,  2.87878788,  2.77777778,  2.67676768,  2.57575758,
        2.47474747,  2.37373737,  2.27272727,  2.17171717,  2.07070707,
        1.96969697,  1.86868687,  1.76767677,  1.66666667,  1.56565657,
        1.46464646,  1.36363636,  1.26262626,  1.16161616,  1.06060606,
        0.95959596,  0.85858586,  0.75757576,  0.65656566,  0.55555556,
        0.45454545,  0.35353535,  0.25252525,  0.15151515,  0.05050505,
       -0.05050505, -0.15151515, -0.25252525, -0.35353535, -0.45454545,
       -0.55555556, -0.65656566, -0.75757576, -0.85858586, -0.95959596,
       -1.06060606, -1.16161616, -1.26262626, -1.36363636, -1.46464646,
       -1.56565657, -1.66666667, -1.76767677, -1.86868687, -1.96969697,
       -2.07070707, -2.17171717, -2.27272727, -2.37373737, -2.47474747,
       -2.57575758, -2.67676768, -2.77777778, -2.87878788, -2.97979798,
       -3.08080808, -3.18181818, -3.28282828, -3.38383838, -3.48484848,
       -3.58585859, -3.68686869, -3.78787879, -3.88888889, -3.98989899,
       -4.09090909, -4.19191919, -4.29292929, -4.39393939, -4.49494949,
       -4.5959596 , -4.6969697 , -4.7979798 , -4.8989899 , -5.        ])
```

In [57]:
```
X1=np.linspace(-5.0,5.0,100)
y1=np.sqrt(5**2-X1**2)
y1=np.hstack([y1,-y1])
x1=np.hstack([X1,-X1])
```

In [58]:
```
plt.scatter(y,x)
```

```
plt.scatter(y1,x1)
```

Out[58]:
```
<matplotlib.collections.PathCollection at 0x26679df1a00>
```



In [59]:
```
df1=pd.DataFrame(np.vstack([y,x]).T,columns=['X1','X2'])
df1['Y']=0
```

In [60]:
```
df1
```

Out[60]:

	X1	X2	Y
0	8.660254	-5.00000	0
1	8.717792	-4.89899	0
2	8.773790	-4.79798	0
3	8.828277	-4.69697	0
4	8.881281	-4.59596	0
...	...	...	...
195	-8.881281	-4.59596	0
196	-8.828277	-4.69697	0
197	-8.773790	-4.79798	0
198	-8.717792	-4.89899	0
199	-8.660254	-5.00000	0

200 rows × 3 columns

In [61]:
```
df2=pd.DataFrame(np.vstack([y1,x1]).T,columns=['X1','X2'])
df2['Y']=1
```

In [62]:
```
df2
```

Out[62]:

	X1	X2	Y
0	0.000000	-5.00000	1
1	0.999949	-4.89899	1
2	1.406908	-4.79798	1
3	1.714198	-4.69697	1
4	1.969049	-4.59596	1
...	...	...	...
195	-1.969049	-4.59596	1
196	-1.714198	-4.69697	1
197	-1.406908	-4.79798	1
198	-0.999949	-4.89899	1
199	-0.000000	-5.00000	1

200 rows × 3 columns

In [63]:
```
df=pd.concat([df1,df2])
```

In [64]:
```
df
```

Out[64]:

	X1	X2	Y
0	8.660254	-5.00000	0
1	8.717792	-4.89899	0
2	8.773790	-4.79798	0
3	8.828277	-4.69697	0
4	8.881281	-4.59596	0
...	...	...	...
195	-1.969049	-4.59596	1
196	-1.714198	-4.69697	1
197	-1.406908	-4.79798	1
198	-0.999949	-4.89899	1
199	-0.000000	-5.00000	1

400 rows × 3 columns

In [65]:
```
X=df.iloc[: , : -2]
y=df.Y
```

In [66]:
```
X
```

Out[66]:

	X1
0	8.660254
1	8.717792
2	8.773790
3	8.828277
4	8.881281
...	...
195	-1.969049
196	-1.714198
197	-1.406908
198	-0.999949
199	-0.000000

400 rows × 1 columns

In [67]:
```
y
```

Out[67]:
```
0      0
1      0
2      0
3      0
4      0
      ..
195    1
196    1
197    1
198    1
199    1
Name: Y, Length: 400, dtype: int64
```

In [68]:
```
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.25,random_state=0)
```

In [69]:
```
from sklearn.svm import SVC
model=SVC(kernel='linear')
model.fit(X_train,y_train)
```

Out[69]:
```
SVC(kernel='linear')
```

In [70]:
```
from sklearn.metrics import accuracy_score
y_pred=model.predict(X_test)
accuracy_score(y_test,y_pred)
```

Out[70]:
```
0.45
```

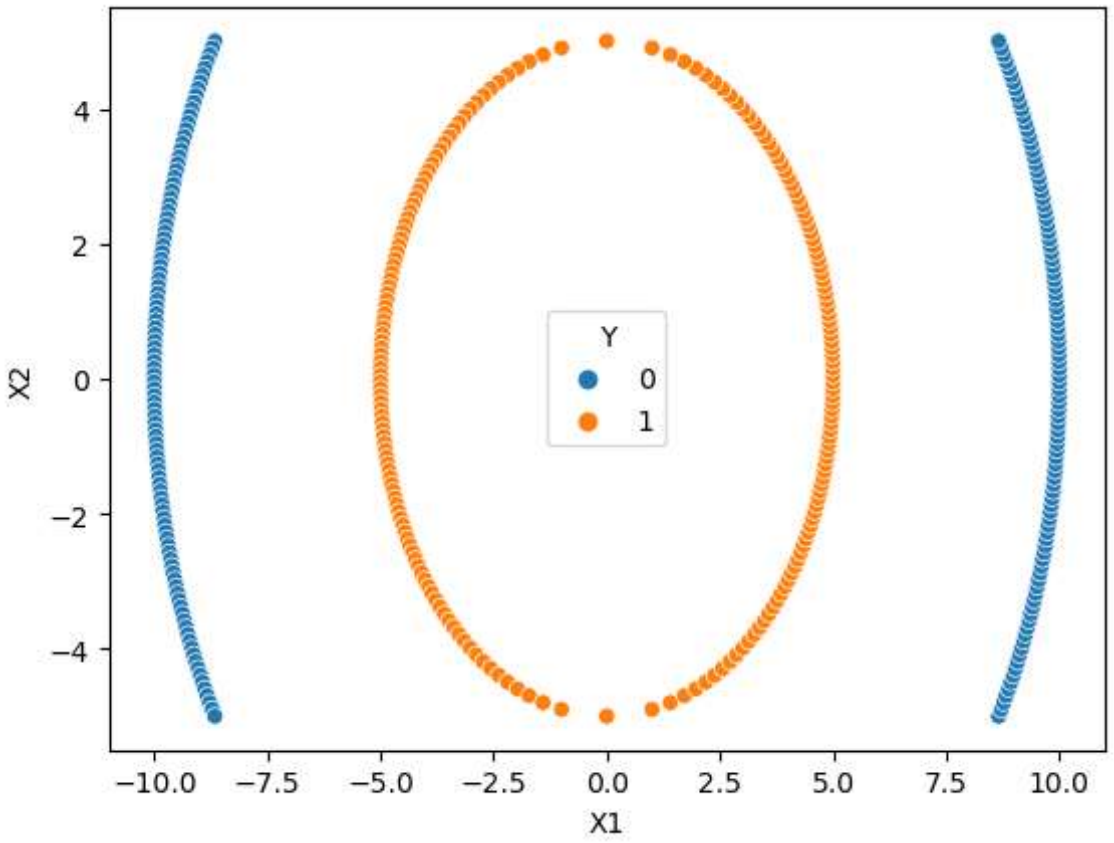
In [119…
```
df.head(10)
```

Out[119]:

	X1	X2	Y	X1_square	X2_square	X1*X2
0	8.660254	-5.000000	0	75.000000	25.000000	-43.301270
1	8.717792	-4.898990	0	75.999898	24.000102	-42.708375
2	8.773790	-4.797980	0	76.979390	23.020610	-42.096467
3	8.828277	-4.696970	0	77.938476	22.061524	-41.466150
4	8.881281	-4.595960	0	78.877155	21.122845	-40.818009
5	8.932829	-4.494949	0	79.795429	20.204571	-40.152614
6	8.982945	-4.393939	0	80.693297	19.306703	-39.470515
7	9.031653	-4.292929	0	81.570758	18.429242	-38.772248
8	9.078976	-4.191919	0	82.427813	17.572187	-38.058336
9	9.124936	-4.090909	0	83.264463	16.735537	-37.329285

In [72]:
```
import seaborn as sns
sns.scatterplot(x=df['X1'],y=df['X2'],hue=df['Y'],data=df)
```

Out[72]:
```
<AxesSubplot:xlabel='X1', ylabel='X2'>
```



In [73]:
```
from sklearn.svm import SVC
model1=SVC(kernel='rbf')
model1.fit(X_train,y_train)
```

Out[73]:
```
SVC()
```

In [77]:
```
from sklearn.metrics import accuracy_score
y_pred1=model1.predict(X_test)
```

```
accuracy_score(y_test,y_pred1)
```

Out[77]:
```
1.0
```

In [75]:
```
from sklearn.svm import SVC
model2=SVC(kernel='poly')
model2.fit(X_train,y_train)
```

Out[75]:
```
SVC(kernel='poly')
```

In [78]:
```
from sklearn.metrics import accuracy_score
y_pred2=model2.predict(X_test)
accuracy_score(y_test,y_pred2)
```

Out[78]:
```
0.45
```

In [79]:
```
df.head()
```

Out[79]:

	X1	X2	Y
0	8.660254	-5.00000	0
1	8.717792	-4.89899	0
2	8.773790	-4.79798	0
3	8.828277	-4.69697	0
4	8.881281	-4.59596	0

In [82]:
```
df['X1_square']=df['X1']**2
df['X2_square']=df['X2']**2
df["X1*X2"]=(df['X1']*df['X2'])
```

In [83]:
```
df.head()
```

Out[83]:

	X1	X2	Y	X1_square	X2_square	X1*X2
0	8.660254	-5.00000	0	75.000000	25.000000	-43.301270
1	8.717792	-4.89899	0	75.999898	24.000102	-42.708375
2	8.773790	-4.79798	0	76.979390	23.020610	-42.096467
3	8.828277	-4.69697	0	77.938476	22.061524	-41.466150
4	8.881281	-4.59596	0	78.877155	21.122845	-40.818009

In [90]:
```
X=df[["X1","X2","X1_square","X2_square","X1*X2"]]
```

In [91]:
```
X
```

Out[91]:

	X1	X2	X1_square	X2_square	X1*X2
0	8.660254	-5.00000	75.000000	25.000000	-43.301270
1	8.717792	-4.89899	75.999898	24.000102	-42.708375
2	8.773790	-4.79798	76.979390	23.020610	-42.096467
3	8.828277	-4.69697	77.938476	22.061524	-41.466150
4	8.881281	-4.59596	78.877155	21.122845	-40.818009
...	...	...	...	...	...
195	-1.969049	-4.59596	3.877155	21.122845	9.049671
196	-1.714198	-4.69697	2.938476	22.061524	8.051537
197	-1.406908	-4.79798	1.979390	23.020610	6.750316
198	-0.999949	-4.89899	0.999898	24.000102	4.898740
199	-0.000000	-5.00000	0.000000	25.000000	0.000000

400 rows × 5 columns

In [92]:
```
X_test
```

Out[92]:

	X1
132	-9.842526
109	-2.874798
141	-4.925731
196	-8.828277
46	4.987486
...	...
146	-9.993749
135	-9.892159
190	-2.874798
64	4.780671
164	-4.780671

100 rows × 1 columns

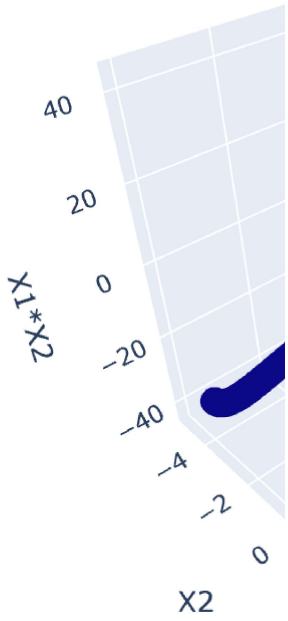
In [93]:
```
X_train
```

Out[93]:

	X1
50	4.999745
63	9.906589
112	-3.263736
159	-9.953852
83	3.680983
...	...
123	-4.223140
192	-9.031653
117	-9.445795
47	9.996811
172	-9.738311

300 rows × 1 columns

In [95]:
```
import plotly.express as px
fig=px.scatter_3d(df,x='X1',y='X2',z='X1*X2',color='Y')
fig.show()
```



In [96]:
```
import plotly.express as px
fig=px.scatter_3d(df,x='X1_square',y='X2_square',z='X1*X2',color='Y')
fig.show()
```