svm-kernels

December 28, 2023

1 Day - 24___________#100DaysOfML

1.0.1 Kernels In SVM

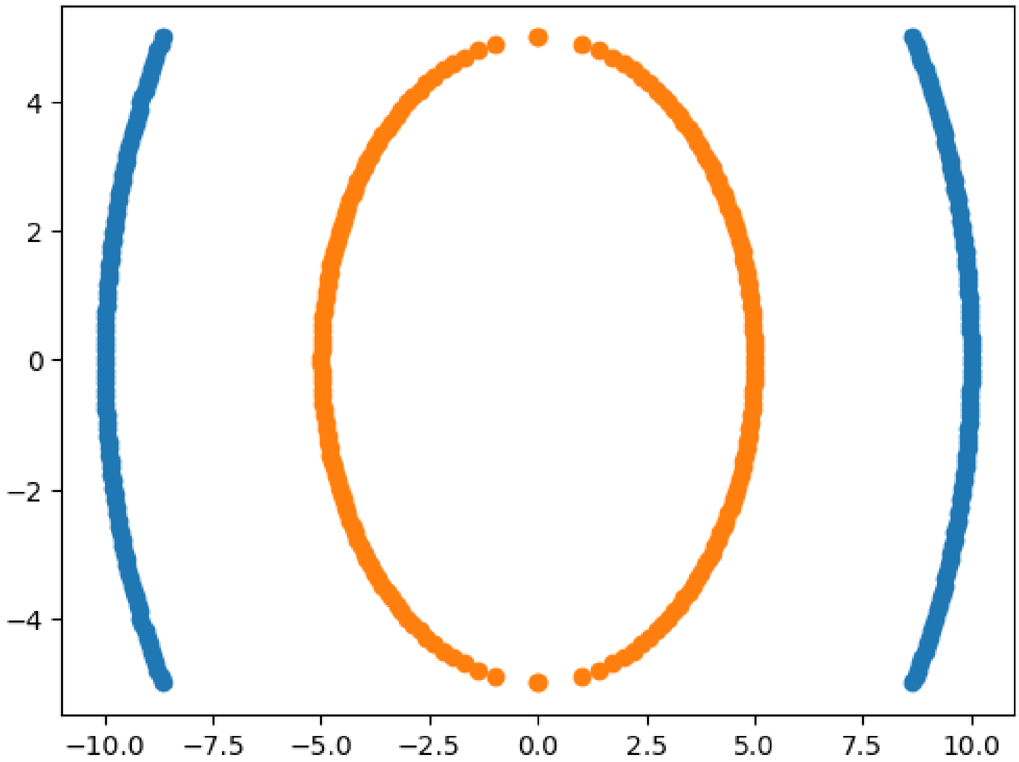
```
[1]: import pandas as pd
     import numpy as np
     import matplotlib.pyplot as plt
     import seaborn as sns
     %matplotlib inline
```

```
[6]: 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])
```

```
[7]: 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])
```

```
[8]: plt.scatter(y,x)
     plt.scatter(y1,x1)
```

```
[8]: <matplotlib.collections.PathCollection at 0x25370cea2c0>
```



[11]:
```
df1 = pd.DataFrame(np.vstack([y,x]).T,columns= ['X1','X2'])
df1['Y'] = 0
df2 = pd.DataFrame(np.vstack([y1,x1]).T,columns= ['X1','X2'])
df2['Y'] = 1
df = df1.append(df2)
df.head()
```

```
C:\Users\Sameer\AppData\Local\Temp\ipykernel_10976\972131391.py:5:
FutureWarning: The frame.append method is deprecated and will be removed from
pandas in a future version. Use pandas.concat instead.
  df = df1.append(df2)
```

[11]:
```
         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
```

[12]:
```
df.tail()
```

[12]:

```
           X1       X2  Y
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
```

[17]:

```
df['X1_Square'] = df['X1']**2
df['X2_Square'] = df['X2']**2
df['X1*X2'] = df['X1']*df['X2']
df.columns
```

[17]:

```
Index(['X1', 'X2', 'Y', 'X1_Square', 'X2_Square', 'X1 * X2', 'X1*X2'],
dtype='object')
```

[20]:

```
X = df[['X1_Square', 'X2_Square', 'X1 * X2']]
y = df['Y']
```

[21]:

```
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)
```

[23]:

```
import plotly.express as px
fig = px.scatter_3d(df,x ='X1_Square', y='X2_Square',z= 'X1 * X2',color= y)
fig.show()
```

[32]:

```
x = np.array([0,1])
y = np.array([1,1])
z = np.array([1,1])
df = pd.DataFrame({"x": x, "y":y, "z":z})
```

[24]:

```
from sklearn.metrics import accuracy_score
from sklearn.svm import SVC
clf = SVC(kernel="linear")
clf.fit(X_train,y_train)
y_pred = clf.predict(X_test)
print(accuracy_score(y_test,y_pred))
```

```
1.0
```

[25]:

```
from sklearn.metrics import accuracy_score
from sklearn.svm import SVC
clf = SVC(kernel="rbf")
clf.fit(X_train,y_train)
y_pred = clf.predict(X_test)
print(accuracy_score(y_test,y_pred))
```

```
1.0
```

[27]:

```
from sklearn.metrics import accuracy_score
from sklearn.svm import SVC
clf = SVC(kernel="sigmoid")
clf.fit(X_train,y_train)
y_pred = clf.predict(X_test)
print(accuracy_score(y_test,y_pred))
```

```
0.76
```

[28]:

```
from sklearn.metrics import accuracy_score
from sklearn.svm import SVC
clf = SVC(kernel="poly")
clf.fit(X_train,y_train)
y_pred = clf.predict(X_test)
print(accuracy_score(y_test,y_pred))
```

```
1.0
```

[37]:

```
import plotly.io as pio
pio.renderers.default = "notebook+pdf"  #
```

[ ]:

```

```