5. SVM

```python
import numpy as np
import matplotlib.pyplot as plt
from sklearn import svm
from sklearn.datasets import  make_classification

X,Y = make_classification(n_classes=2,n_samples=300,n_clusters_per_class=1,random_s

model = svm.SVC(kernel='linear')
model.fit(X, Y)

w = model.coef_[0]
a = -w[0] / w[1]
xx = np.linspace(-4, 3)
yy = a * xx - (model.intercept_[0]) / w[1]


b = model.support_vectors_[0]
yy_down = a * xx + (b[1] - a * b[0])
b = model.support_vectors_[-1]
yy_up = a * xx + (b[1] - a * b[0])

plt.plot(xx, yy, 'k-')
plt.plot(xx, yy_down, 'k--')
plt.plot(xx, yy_up, 'k--')

plt.scatter(X[:, 0], X[:, 1], c=Y, cmap=plt.cm.Paired)

plt.show()
```
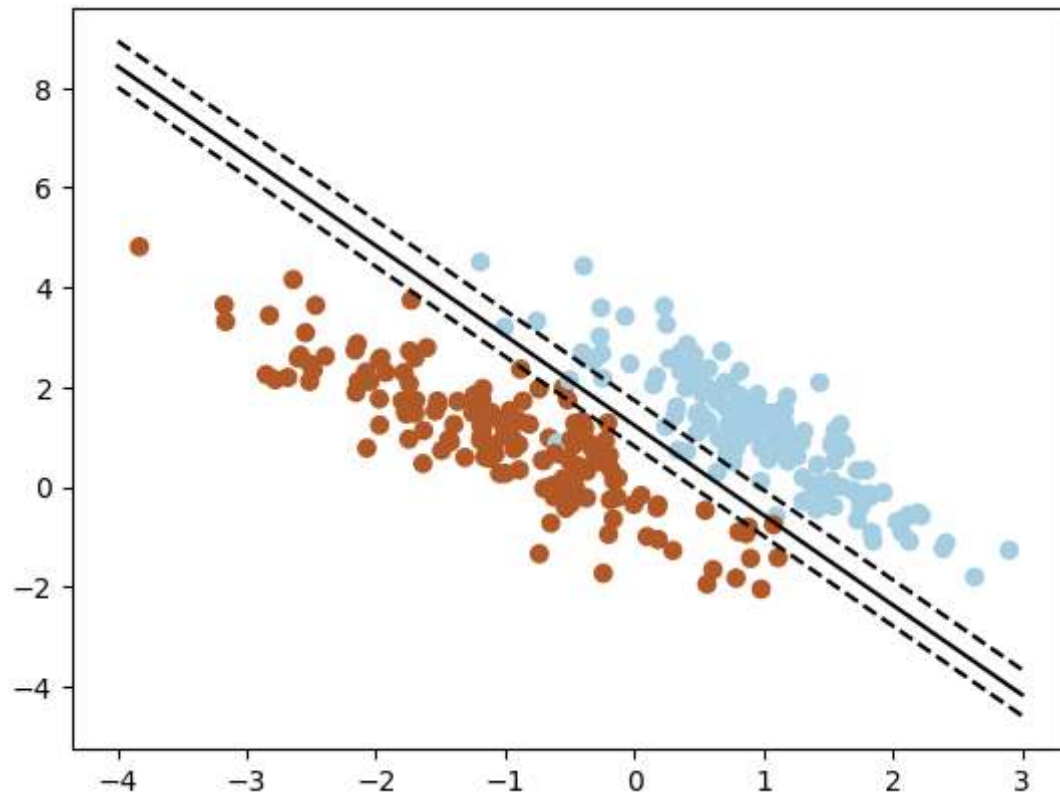
In [ ]: