

جواب سوال 8 و 9 و 10)

می خواهیم کلاسیفایر SVM را با استفاده از پکیج sklearn بر روی سه فیچر خواسته شده با استفاده از داده های آموزشی ترین کنیم :
می توانیم برای این کار از کرنل های مختلف SVM استفاده کنیم. در صورت استفاده نکردن از kernel به صورت پیش فرض 'rbf' انتخاب می شود.
(default = 'rbf')

kernel{'linear', 'poly', 'rbf', 'sigmoid', 'precomputed'}, default = 'rbf'

برای اطمینان از این موضوع ما یک بار بدون استفاده از کرنل و هم با استفاده از کرنل rbf جواب ها را مقایسه می کنیم.

خروجی در هر دو یکسان خواهد بود :

```
from sklearn import svm
x1 = ds[['trestbps', 'chol', 'thalach']]

# Create a SVM classifier
svclassifier = svm.SVC()
X_train, X_test, y_train, y_test = tts( x1 , y , test_size = 0.2, random_state = 0)
svclassifier.fit(X_train, y_train)
pred = svclassifier.predict(X_test)
print(confusion_matrix(y_test,pred))
print(classification_report(y_test,pred))
```

```
[[18 11]
 [ 4 25]]
precision recall f1-score support

0 0.82 0.62 0.71 29
1 0.69 0.86 0.77 29

accuracy 0.74 58
macro avg 0.76 0.74 0.74 58
weighted avg 0.76 0.74 0.74 58
```

```
# Create a Radial basis function SVM classifier
svclassifier = svm.SVC(kernel='rbf')
X_train, X_test, y_train, y_test = tts( x1 , y , test_size = 0.2, random_state = 0)
svclassifier.fit(X_train, y_train)
pred = svclassifier.predict(X_test)
print(confusion_matrix(y_test,pred))
print(classification_report(y_test,pred))
```

```
[[18 11]
 [ 4 25]]
precision recall f1-score support

0 0.82 0.62 0.71 29
1 0.69 0.86 0.77 29

accuracy 0.74 58
macro avg 0.76 0.74 0.74 58
weighted avg 0.76 0.74 0.74 58
```

اکنون می خواهیم با استفاده از دو کرنل متفاوت دیگر ('linear' , 'poly') خروجی را مقایسه کنیم :

```
# Create a polynomial SVM classifier
```

```
svclassifier = svm.SVC(kernel='poly')
```

```
X_train, X_test, y_train, y_test = tts( x1 , y , test_size = 0.2, random_state = 0)
```

```
svclassifier.fit(X_train, y_train)
```

```
pred = svclassifier.predict(X_test)
```

```
print(confusion_matrix(y_test,pred))
```

```
print(classification_report(y_test,pred))
```

```
[[11 18]
```

```
 [ 4 25]]
```

```
precision recall f1-score support
```

```
0    0.73    0.38    0.50    29
```

```
1    0.58    0.86    0.69    29
```

```
accuracy                0.62    58
```

```
macro avg    0.66    0.62    0.60    58
```

```
weighted avg    0.66    0.62    0.60    58
```

```
# Create a linear SVM classifier
```

```
svclassifier = svm.SVC(kernel='linear')
```

```
X_train, X_test, y_train, y_test = tts( x1 , y , test_size = 0.2, random_state = 0)
```

```
svclassifier.fit(X_train, y_train)
```

```
pred = svclassifier.predict(X_test)
```

```
print(confusion_matrix(y_test,pred))
```

```
print(classification_report(y_test,pred))
```

```
[[18 11]
```

```
 [ 3 26]]
```

```
precision recall f1-score support
```

```
0    0.86    0.62    0.72    29
```

```
1    0.70    0.90    0.79    29
```

```
accuracy                0.76    58
```

```
macro avg    0.78    0.76    0.75    58
```

```
weighted avg    0.78    0.76    0.75    58
```

همان طور که ملاحظه می کنید خروجی های متفاوتی به دست آمد. کرنل poly مقدار خطای کمتری در کلاس 0 نسبت به بقیه کرنل ها دارد. اما میزان صحت سنجی کمتری دارد. کرنل linear بیشترین میزان صحت سنجی دارد اما بیشترین مقدار خطا را در کلاس 1 به خود اختصاص داده است. با مقایسه خروجی کرنل ها می توان به این نتیجه رسید بهترین جواب در این مثال متعلق به کرنل rbf است.

حال می خواهیم کلاسیفایر SVM را روی همه ی فیچرها با استفاده از داده های آموزشی ترین کنیم :

```
# Create a polynomial SVM classifier on all dataset
```

```
svclassifier = svm.SVC()
```

```
X_train, X_test, y_train, y_test = tts( x1 , y , test_size = 0.2, random_state = 0)
```

```
svclassifier.fit(X_train, y_train)
```

```
pred = svclassifier.predict(X_test)
```

```
print(confusion_matrix(y_test,pred))
```

```
print(classification_report(y_test,pred))
```

```
[[18 11]
```

```
[ 4 25]]
```

```
precision  recall f1-score  support
```

```
0    0.82    0.62    0.71    29
```

```
1    0.69    0.86    0.77    29
```

```
accuracy                0.74    58
```

```
macro avg    0.76    0.74    0.74    58
```

```
weighted avg    0.76    0.74    0.74    58
```

با مقایسه خروجی کلاسیفایر SVM روی همه فیچرها و همچنین روی سه فیچر خواسته شده در می یابیم که خروجی در هر دو حالت یکسان است.