

Ficha laboratorial – Discriminantes Logísticos e Máquinas de Suporte Vetorial

Ex1. Corre o código abaixo.

```
1 import numpy as np
2 from sklearn.linear_model import LogisticRegression
3 from sklearn.datasets import load_breast_cancer
4 from sklearn.model_selection import train_test_split

1 BC=load_breast_cancer()
2 Xtrain,Xtest,ytrain,ytest=train_test_split(BC.data,BC.target)
3 print(ytrain.shape)
4 print(Xtrain.shape)
5 Dlog=LogisticRegression(max_iter=10000).fit(Xtrain, ytrain)
6 print(Dlog.score(Xtest,ytest))
7 print(Dlog.coef_)
8 print(Dlog.intercept_)
9 print(Dlog.n_iter_)
```

Observa e analisa os outputs e depois corre a instrução abaixo e verifica a grande diferença nos outputs:

```
Dlog=LogisticRegression(solver='newton-cg', C=0.1, penalty='l2', max_iter=10000).fit(Xtrain, ytrain)
```

Ex.2 Corre os dois códigos apresentados abaixo.

```
1 from sklearn.datasets import load_iris
2 from sklearn.metrics import confusion_matrix
3 Iris=load_iris()
4 Xtr,Xte,ytr,yte=train_test_split(Iris.data,Iris.target,test_size=1./3,random_state=0)

1 Dlog1=LogisticRegression(solver='saga',C=0.1,penalty='l2', max_iter=1000).fit(Xtr,ytr)
2 print(Dlog1.coef_)
3 print(Dlog1.intercept_)
4 print(Dlog1.n_iter_)
5 yest=Dlog1.predict(Xte)
6 print(confusion_matrix(yte,yest))
```

Analisa os outputs e depois altera a “penalty” para “l1” e observa a diferença.

Ex. 3 Corre o código abaixo e observa o valor obtido no desempenho:

```
1 from sklearn.svm import SVC
2 svm=SVC(kernel='rbf', gamma='auto').fit(Xtrain, ytrain)
3 print(svm.score(Xtest, ytest))
```

A seguir corre o código abaixo e verifica o impacto do pré-processamento nos dados:

```
1 from sklearn.preprocessing import StandardScaler
2 sc=StandardScaler().fit(Xtrain, ytrain)
3 Xtrain1=sc.transform(Xtrain); Xtest1=sc.transform(Xtest)
4 svm1=SVC(kernel='rbf', gamma='auto').fit(Xtrain1, ytrain)
5 print(svm1.score(Xtest1, ytest))
```