kfold cross_val_score

May 11, 2023

[2]:
```
import pandas as pd
from sklearn.datasets import load_iris
```

[10]:
```
df=load_iris()
```

[11]:
```
dir(df)
```

[11]:
```
['DESCR',
 'data',
 'data_module',
 'feature_names',
 'filename',
 'frame',
 'target',
 'target_names']
```

[14]:
```
df1=pd.DataFrame(load_iris().data,columns=load_iris().feature_names)
```

[15]:
```
df1.head()
```

[15]:

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2

[22]:
```
df1['target']=load_iris().target
df1.head()
```

[22]:

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	\
0	5.1	3.5	1.4	0.2	
1	4.9	3.0	1.4	0.2	
2	4.7	3.2	1.3	0.2	
3	4.6	3.1	1.5	0.2	
4	5.0	3.6	1.4	0.2	

	target

```
0       0
1       0
2       0
3       0
4       0
```

[40]:
```
df1['species']=df1.target.apply(lambda x : load_iris().target_names[x])
df1.loc[139:140]
```

[40]:
```
     sepal length (cm)  sepal width (cm)  petal length (cm)  petal width (cm)  \
139                6.9               3.1                5.4               2.1
140                6.7               3.1                5.6               2.4

     target    species
139       2  virginica
140       2  virginica
```

[51]:
```
from sklearn.model_selection import cross_val_score
from sklearn.linear_model import LogisticRegression
```

[52]:
```
logicscore=cross_val_score(LogisticRegression(),load_iris().data, load_iris().
 ↪target)
logicscore
```

```
C:\Users\Rakesh\anaconda3\lib\site-
packages\sklearn\linear_model\_logistic.py:814: ConvergenceWarning: lbfgs failed
to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
  n_iter_i = _check_optimize_result(
```

[52]:
```
array([0.96666667, 1.        , 0.93333333, 0.96666667, 1.        ])
```

[ ]:

[ ]: