da-lab-8

April 28, 2023

[1]:
```
from sklearn.datasets import make_classification
from sklearn.pipeline import Pipeline
from sklearn.model_selection import RepeatedStratifiedKFold
from sklearn.model_selection import cross_val_score
from sklearn.decomposition import PCA
from sklearn.linear_model import LogisticRegression
import matplotlib.pyplot as plt
import matplotlib
import pandas as pd
import numpy as np
```

[2]:
```
def read_data():
␣
 ↪X,y=make_classification(n_samples=1000,n_features=20,n_informative=15,n_redundant=5,random_
 return X,y
```

[3]:
```
data=pd.read_csv('/content/drive/MyDrive/Project/amazon(DA).csv')
data.dropna(inplace=True)
dataset=data.values
X=dataset[:,:-1]
y=dataset[:,-1]
print(X.shape,y.shape)
```

```
(1463, 4) (1463,)
```

[3]:
```
```

[4]:
```
#define steps as a list of tuples
steps=[('pca',PCA(n_components=4)),('m',LogisticRegression())]
models=Pipeline(steps=steps)
```

[5]:
```
#evaluate the model
cv=RepeatedStratifiedKFold(n_splits=9,n_repeats=3,random_state=1)
n_scores=cross_val_score(models,X,y,cv=cv,n_jobs=-1,scoring='accuracy',error_score='raise')
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
```

```
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
```

[6]:
```
from numpy import mean
from numpy import std
```

[7]:
```
print('Accuracy: %.3f (%.3f)' %(mean(n_scores),std(n_scores)))
```

```
Accuracy: 0.021 (0.009)
```

[8]:
```
def create_models():
 models=dict()
 for i in range(0,21):
  steps=[('pca',PCA(n_components=i)),('m',LogisticRegression())]
  models[str(i)]=Pipeline(steps=steps)
 return models
```

[9]:
```
#def evaluate_models(models,X,y):
#if X.shape[1] == 0:
#3 raise ValueError("Input feature array must have at least one column.")
#cv=RepeatedStratifiedKFold(n_splits=9,n_repeats=3,random_state=1)
#scores=cross_val_score(models,X,y,cv=cv,n_jobs=-1,scoring='accuracy')
# return scores
def evaluate_models(models, X, y):
    if X.shape[1] == 0:
        raise ValueError("Input feature array must have at least one column.")
    cv = RepeatedStratifiedKFold(n_splits=9, n_repeats=3, random_state=1)
    results, names = list(), list()
    for i, model in enumerate(models):
        name = f"Model_{i+1}"
        scores = cross_val_score(model, X, y, cv=cv, n_jobs=-1,␣
 ↪scoring='accuracy')
        results.append(scores)
        names.append(name)
        print('#%s %.3f (%.3f)' % (name, mean(scores), std(scores)))
    return results, names
```

[10]:
```
models=create_models()
```

[11]:
```
results,names=list(),list()
for name,model in models.items():
 scores=evaluate_models(model,X,y)
 results.append(scores)
 names.append(name)
print('#%s %.3f (%.3f)' % (name, np.mean(scores), np.std(scores)))
```

```
#Model_1 nan (nan)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
```

```
#Model_2 0.026 (0.015)
#Model_1 nan (nan)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
```

```
#Model_2 0.026 (0.015)
#Model_1 nan (nan)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
```

```
#Model_2 0.026 (0.015)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
```

```
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
```

```
#Model_1 nan (nan)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
```

```
#Model_2 0.026 (0.015)
#Model_1 nan (nan)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
```

```
#Model_2 0.026 (0.015)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=9.
  warnings.warn(
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-11-86ffdfec408c> in <cell line: 2>()
      1 results,names=list(),list()
      2 for name,model in models.items():
----> 3  scores=evaluate_models(model,X,y)
      4  results.append(scores)
      5  names.append(name)

<ipython-input-9-38fb89e0dfa4> in evaluate_models(models, X, y)
     12     for i, model in enumerate(models):
     13         name = f"Model_{i+1}"
---> 14         scores = cross_val_score(model, X, y, cv=cv, n_jobs=-1,␣
 ↪scoring='accuracy')
     15         results.append(scores)
     16         names.append(name)

/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_validation.py␣
 ↪in cross_val_score(estimator, X, y, groups, scoring, cv, n_jobs, verbose,␣
 ↪fit_params, pre_dispatch, error_score)
    513     scorer = check_scoring(estimator, scoring=scoring)
    514
--> 515     cv_results = cross_validate(
    516         estimator=estimator,
    517         X=X,

/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_validation.py␣
 ↪in cross_validate(estimator, X, y, groups, scoring, cv, n_jobs, verbose,␣
 ↪fit_params, pre_dispatch, return_train_score, return_estimator, error_score)
    283     )
    284
--> 285     _warn_or_raise_about_fit_failures(results, error_score)
    286
```

```
    287     # For callabe scoring, the return type is only know after calling.␣
 ↪If the

/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_validation.py␣
 ↪in _warn_or_raise_about_fit_failures(results, error_score)
    365                 f"Below are more details about the failures:
 ↪\n{fit_errors_summary}"
    366             )
--> 367             raise ValueError(all_fits_failed_message)
    368
    369         else:

ValueError:
All the 27 fits failed.
It is very likely that your model is misconfigured.
You can try to debug the error by setting error_score='raise'.

Below are more details about the failures:
--------------------------------------------------------------------------------
27 fits failed with the following error:
Traceback (most recent call last):
  File "/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/
 ↪_validation.py", line 686, in _fit_and_score
    estimator.fit(X_train, y_train, **fit_params)
  File "/usr/local/lib/python3.10/dist-packages/sklearn/decomposition/_pca.py",
 ↪line 435, in fit
    self._fit(X)
  File "/usr/local/lib/python3.10/dist-packages/sklearn/decomposition/_pca.py",
 ↪line 512, in _fit
    return self._fit_full(X, n_components)
  File "/usr/local/lib/python3.10/dist-packages/sklearn/decomposition/_pca.py",
 ↪line 526, in _fit_full
    raise ValueError(
ValueError: n_components=5 must be between 0 and min(n_samples, n_features)=4␣
 ↪with svd_solver='full'
```

[ ]:
```
plt.boxplot(results,labels=names,showmeans=True)
plt.xticks(rotation=45)
plt.show()
```

[ ]:
```
#Making predictions
#define steps with 15 PC
steps=[('pca',PCA(n_components=4)),('m',LogisticRegression())]
models=Pipeline(steps=steps)
```

[ ]:
```
models.fit(X,y)
```

```
[ ]: Pipeline(memory=None,steps=[('pca',PCA(copy=True, iterated_power='auto',␣
      ↪n_components=15,random_state=None, svd_solver='auto', tol=0.0,whiten=False)),
     ('m',LogisticRegression(C=1.0, class_weight=None,␣
      ↪dual=False,fit_intercept=True, intercept_scaling=1,l1_ratio=None,␣
      ↪max_iter=100,multi_class='auto', n_jobs=None,penalty='l2', random_state=None,
     solver='lbfgs', tol=0.0001, verbose=0,
     warm_start=False))],verbose=False)

[ ]: #single test data for prediction
     row = [[0.2929949, -4.21223056, -1.288332, -2.17849815,]]

[ ]: yhat = models.predict(row)
     print('Predicted Class: %d' % yhat[0])

[ ]: #using real data set
     from sklearn.datasets import load_diabetes
     from sklearn.preprocessing import StandardScaler
     import pandas as pd

[ ]: df=pd.read_csv('/content/drive/MyDrive/Project/amazon(DA).csv')
     df.head(6)

[ ]: df.columns

[ ]: dataset=df.values
     X=dataset[:,:-1]
     y=dataset[:,-1]

[ ]: X.shape,y.shape

[ ]: scaler = StandardScaler()

[ ]: scaler.fit(df)

[ ]: Diabetes_scaled = scaler.transform(df)

[ ]: steps=[('norm',StandardScaler()),('pca',PCA(n_components=10)),('m',LogisticRegression())]
     model=Pipeline(steps=steps)

[ ]: def create_models():
      models=dict()
      for i in range(1,11):
      ␣
      ↪steps=[('norm',StandardScaler()),('pca',PCA(n_components=10)),('m',LogisticRegression())]
       models[str(i)]=Pipeline(steps=steps)
      return models
```