metrics_ex_01

July 5, 2024

1 Exercise M7.02

We presented different classification metrics in the previous notebook. However, we did not use it with a cross-validation. This exercise aims at practicing and implementing cross-validation.

We will reuse the blood transfusion dataset.

[3]:
```python
import pandas as pd

blood_transfusion = pd.read_csv("../datasets/blood_transfusion.csv")
data = blood_transfusion.drop(columns="Class")
target = blood_transfusion["Class"]
```

Note

If you want a deeper overview regarding this dataset, you can refer to the Appendix - Datasets description section at the end of this MOOC.

First, create a decision tree classifier.

[4]:
```python
from sklearn.tree import DecisionTreeClassifier

model = DecisionTreeClassifier()
```

Create a `StratifiedKFold` cross-validation object. Then use it inside the `cross_val_score` function to evaluate the decision tree. We will first use the accuracy as a score function. Explicitly use the `scoring` parameter of `cross_val_score` to compute the accuracy (even if this is the default score). Check its documentation to learn how to do that.

[5]:
```python
from sklearn.model_selection import StratifiedKFold, cross_val_score

cv = StratifiedKFold()
cv_results = cross_val_score(model, data, target, cv=cv, scoring='accuracy')
cv_results.mean()
```

[5]:
```
0.611185682326622
```

Repeat the experiment by computing the `balanced_accuracy`.

[6]:

```
cv_results = cross_val_score(model, data, target, cv=cv,␣
 ↪scoring='balanced_accuracy')
cv_results.mean()
```

[6]:

```
0.5062781954887219
```

We will now add a bit of complexity. We would like to compute the precision of our model. However, during the course we saw that we need to mention the positive label which in our case we consider to be the class `donated`.

We will show that computing the precision without providing the positive label will not be supported by scikit-learn because it is indeed ambiguous.

[7]:

```
from sklearn.model_selection import cross_val_score
from sklearn.tree import DecisionTreeClassifier

tree = DecisionTreeClassifier()
try:
    scores = cross_val_score(tree, data, target, cv=10, scoring="precision")
except ValueError as exc:
    print(exc)
```

```
/opt/conda/lib/python3.11/site-
packages/sklearn/model_selection/_validation.py:842: UserWarning: Scoring
failed. The score on this train-test partition for these parameters will be set
to nan. Details:
Traceback (most recent call last):
  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 136, in __call__
    score = scorer._score(
            ^^^^^^^^^^^^^^

  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 355, in _score
    return self._sign * self._score_func(y_true, y_pred, **scoring_kwargs)
                        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 211, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^

  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 2127, in precision_score
    p, _, _, _ = precision_recall_fscore_support(
                 ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 184, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^

  File "/opt/conda/lib/python3.11/site-
```

```
packages/sklearn/metrics/_classification.py", line 1721, in
precision_recall_fscore_support
    labels = _check_set_wise_labels(y_true, y_pred, average, labels, pos_label)
             ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 1507, in
_check_set_wise_labels
    raise ValueError(
ValueError: pos_label=1 is not a valid label. It should be one of ['donated',
'not donated']

  warnings.warn(
/opt/conda/lib/python3.11/site-
packages/sklearn/model_selection/_validation.py:842: UserWarning: Scoring
failed. The score on this train-test partition for these parameters will be set
to nan. Details:
Traceback (most recent call last):
  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 136, in __call__
    score = scorer._score(
            ^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 355, in _score
    return self._sign * self._score_func(y_true, y_pred, **scoring_kwargs)
                        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 211, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 2127, in precision_score
    p, _, _, _ = precision_recall_fscore_support(
                 ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 184, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 1721, in
precision_recall_fscore_support
    labels = _check_set_wise_labels(y_true, y_pred, average, labels, pos_label)
             ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 1507, in
_check_set_wise_labels
    raise ValueError(
ValueError: pos_label=1 is not a valid label. It should be one of ['donated',
'not donated']
```

```
  warnings.warn(
/opt/conda/lib/python3.11/site-
packages/sklearn/model_selection/_validation.py:842: UserWarning: Scoring
failed. The score on this train-test partition for these parameters will be set
to nan. Details:
Traceback (most recent call last):
  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 136, in __call__
    score = scorer._score(
            ^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 355, in _score
    return self._sign * self._score_func(y_true, y_pred, **scoring_kwargs)
                        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 211, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 2127, in precision_score
    p, _, _, _ = precision_recall_fscore_support(
                 ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 184, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 1721, in
precision_recall_fscore_support
    labels = _check_set_wise_labels(y_true, y_pred, average, labels, pos_label)
             ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 1507, in
_check_set_wise_labels
    raise ValueError(
ValueError: pos_label=1 is not a valid label. It should be one of ['donated',
'not donated']

  warnings.warn(
/opt/conda/lib/python3.11/site-
packages/sklearn/model_selection/_validation.py:842: UserWarning: Scoring
failed. The score on this train-test partition for these parameters will be set
to nan. Details:
Traceback (most recent call last):
  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 136, in __call__
    score = scorer._score(
            ^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 355, in _score
    return self._sign * self._score_func(y_true, y_pred, **scoring_kwargs)
                        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 211, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 2127, in precision_score
    p, _, _, _ = precision_recall_fscore_support(
                 ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 184, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 1721, in
precision_recall_fscore_support
    labels = _check_set_wise_labels(y_true, y_pred, average, labels, pos_label)
             ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 1507, in
_check_set_wise_labels
    raise ValueError(
ValueError: pos_label=1 is not a valid label. It should be one of ['donated',
'not donated']

  warnings.warn(
/opt/conda/lib/python3.11/site-
packages/sklearn/model_selection/_validation.py:842: UserWarning: Scoring
failed. The score on this train-test partition for these parameters will be set
to nan. Details:
Traceback (most recent call last):
  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 136, in __call__
    score = scorer._score(
            ^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 355, in _score
    return self._sign * self._score_func(y_true, y_pred, **scoring_kwargs)
                        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 211, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
```

```
packages/sklearn/metrics/_classification.py", line 2127, in precision_score
    p, _, _, _ = precision_recall_fscore_support(
                 ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 184, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 1721, in
precision_recall_fscore_support
    labels = _check_set_wise_labels(y_true, y_pred, average, labels, pos_label)
             ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 1507, in
_check_set_wise_labels
    raise ValueError(
ValueError: pos_label=1 is not a valid label. It should be one of ['donated',
'not donated']

  warnings.warn(
/opt/conda/lib/python3.11/site-
packages/sklearn/model_selection/_validation.py:842: UserWarning: Scoring
failed. The score on this train-test partition for these parameters will be set
to nan. Details:
Traceback (most recent call last):
  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 136, in __call__
    score = scorer._score(
            ^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 355, in _score
    return self._sign * self._score_func(y_true, y_pred, **scoring_kwargs)
                        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 211, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 2127, in precision_score
    p, _, _, _ = precision_recall_fscore_support(
                 ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 184, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 1721, in
precision_recall_fscore_support
```

```
    labels = _check_set_wise_labels(y_true, y_pred, average, labels, pos_label)
             ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 1507, in
_check_set_wise_labels
    raise ValueError(
ValueError: pos_label=1 is not a valid label. It should be one of ['donated',
'not donated']

  warnings.warn(
/opt/conda/lib/python3.11/site-
packages/sklearn/model_selection/_validation.py:842: UserWarning: Scoring
failed. The score on this train-test partition for these parameters will be set
to nan. Details:
Traceback (most recent call last):
  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 136, in __call__
    score = scorer._score(
            ^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 355, in _score
    return self._sign * self._score_func(y_true, y_pred, **scoring_kwargs)
                        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 211, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 2127, in precision_score
    p, _, _, _ = precision_recall_fscore_support(
                 ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 184, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 1721, in
precision_recall_fscore_support
    labels = _check_set_wise_labels(y_true, y_pred, average, labels, pos_label)
             ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 1507, in
_check_set_wise_labels
    raise ValueError(
ValueError: pos_label=1 is not a valid label. It should be one of ['donated',
'not donated']

  warnings.warn(
/opt/conda/lib/python3.11/site-
packages/sklearn/model_selection/_validation.py:842: UserWarning: Scoring
failed. The score on this train-test partition for these parameters will be set
to nan. Details:
Traceback (most recent call last):
  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 136, in __call__
    score = scorer._score(
            ^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 355, in _score
    return self._sign * self._score_func(y_true, y_pred, **scoring_kwargs)
                        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 211, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 2127, in precision_score
    p, _, _, _ = precision_recall_fscore_support(
                 ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 184, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 1721, in
precision_recall_fscore_support
    labels = _check_set_wise_labels(y_true, y_pred, average, labels, pos_label)
             ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 1507, in
_check_set_wise_labels
    raise ValueError(
ValueError: pos_label=1 is not a valid label. It should be one of ['donated',
'not donated']

  warnings.warn(
/opt/conda/lib/python3.11/site-
packages/sklearn/model_selection/_validation.py:842: UserWarning: Scoring
failed. The score on this train-test partition for these parameters will be set
to nan. Details:
Traceback (most recent call last):
  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 136, in __call__
    score = scorer._score(
            ^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 355, in _score
    return self._sign * self._score_func(y_true, y_pred, **scoring_kwargs)
                        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 211, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 2127, in precision_score
    p, _, _, _ = precision_recall_fscore_support(
                 ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 184, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 1721, in
precision_recall_fscore_support
    labels = _check_set_wise_labels(y_true, y_pred, average, labels, pos_label)
             ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 1507, in
_check_set_wise_labels
    raise ValueError(
ValueError: pos_label=1 is not a valid label. It should be one of ['donated',
'not donated']

  warnings.warn(
/opt/conda/lib/python3.11/site-
packages/sklearn/model_selection/_validation.py:842: UserWarning: Scoring
failed. The score on this train-test partition for these parameters will be set
to nan. Details:
Traceback (most recent call last):
  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 136, in __call__
    score = scorer._score(
            ^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-packages/sklearn/metrics/_scorer.py",
line 355, in _score
    return self._sign * self._score_func(y_true, y_pred, **scoring_kwargs)
                        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 211, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 2127, in precision_score
    p, _, _, _ = precision_recall_fscore_support(
                 ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/utils/_param_validation.py", line 184, in wrapper
    return func(*args, **kwargs)
           ^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 1721, in
precision_recall_fscore_support
    labels = _check_set_wise_labels(y_true, y_pred, average, labels, pos_label)
             ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/conda/lib/python3.11/site-
packages/sklearn/metrics/_classification.py", line 1507, in
_check_set_wise_labels
    raise ValueError(
ValueError: pos_label=1 is not a valid label. It should be one of ['donated',
'not donated']

  warnings.warn(
```

Tip

We catch the exception with a try/except pattern to be able to print it.

We get an exception because the default scorer has its positive label set to one (`pos_label=1`), which is not our case (our positive label is “donated”). In this case, we need to create a scorer using the scoring function and the helper function `make_scorer`.

So, import `sklearn.metrics.make_scorer` and `sklearn.metrics.precision_score`. Check their documentations for more information. Finally, create a scorer by calling `make_scorer` using the score function `precision_score` and pass the extra parameter `pos_label="donated"`.

[10]:

```python
from sklearn.metrics import  make_scorer, precision_score

scorer = make_scorer(precision_score, pos_label="donated")
```

Now, instead of providing the string `"precision"` to the `scoring` parameter in the `cross_val_score` call, pass the scorer that you created above.

[17]:

```python
tree = DecisionTreeClassifier()
try:
    scores = cross_val_score(tree, data, target, cv=cv, scoring=scorer)
except ValueError as exc:
    print(exc)
scores.mean()
```

[17]:

```
0.23535508973919422
```

`cross_val_score` will only compute a single score provided to the `scoring` parameter. The function `cross_validate` allows the computation of multiple scores by passing a list of string or scorer to the parameter `scoring`, which could be handy.

Import `sklearn.model_selection.cross_validate` and compute the accuracy and balanced accuracy through cross-validation. Plot the cross-validation score for both metrics using a box plot.

```
[18]: from sklearn.model_selection import cross_validate

      scoring = ["accuracy", "balanced_accuracy"]
      scores = cross_validate(tree, data, target, cv=cv, scoring=scoring)
      scores
```

```
[18]: {'fit_time': array([0.00994825, 0.00421381, 0.00368786, 0.00331712,
      0.00400615]),
       'score_time': array([0.00469351, 0.0037477 , 0.00338626, 0.00377631,
      0.00394154]),
       'test_accuracy': array([0.41333333, 0.61333333, 0.61333333, 0.70469799,
      0.68456376]),
       'test_balanced_accuracy': array([0.4619883 , 0.49853801, 0.43201754,
      0.61892231, 0.49686717])}
```

```
[21]: scores["test_accuracy"]
```

```
[21]: array([0.41333333, 0.61333333, 0.61333333, 0.70469799, 0.68456376])
```

```
[19]: import pandas as pd

      color = {"whiskers": "black", "medians": "black", "caps": "black"}
      metrics = pd.DataFrame(
          [scores["test_accuracy"], scores["test_balanced_accuracy"]],
          index=["Accuracy", "Balanced accuracy"],
      )
      metrics
```

```
[19]:                           0         1         2         3         4
      Accuracy           0.413333  0.613333  0.613333  0.704698  0.684564
      Balanced accuracy  0.461988  0.498538  0.432018  0.618922  0.496867
```

```
[20]: metrics = pd.DataFrame(
          [scores["test_accuracy"], scores["test_balanced_accuracy"]],
          index=["Accuracy", "Balanced accuracy"],
      ).T
      metrics
```

```
[20]:    Accuracy  Balanced accuracy
      0  0.413333           0.461988
      1  0.613333           0.498538
      2  0.613333           0.432018
      3  0.704698           0.618922
      4  0.684564           0.496867
```

[ ]:

```
import matplotlib.pyplot as plt

metrics.plot.box(color=color)
_ = plt.title("Computation of multiple ")
```