UNIVERSIDADE CATÓLICA PORTUGUESA

BRAGA

Machine Learning

Session 9 - PL

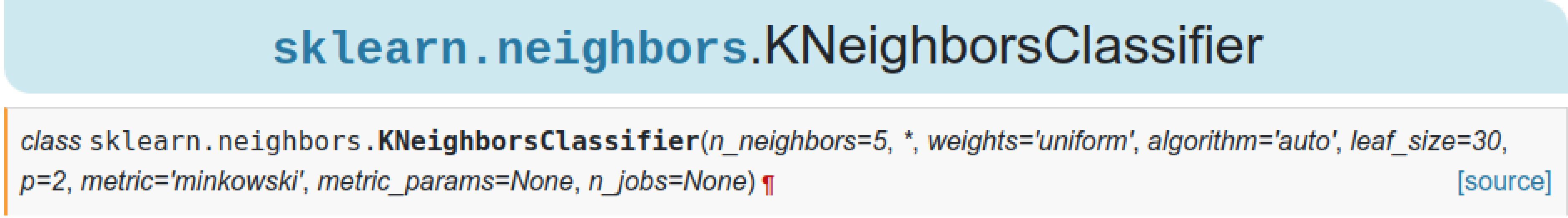
Instance-based and Probabilistic Models

Ciência de Dados Aplicada

2023/2024

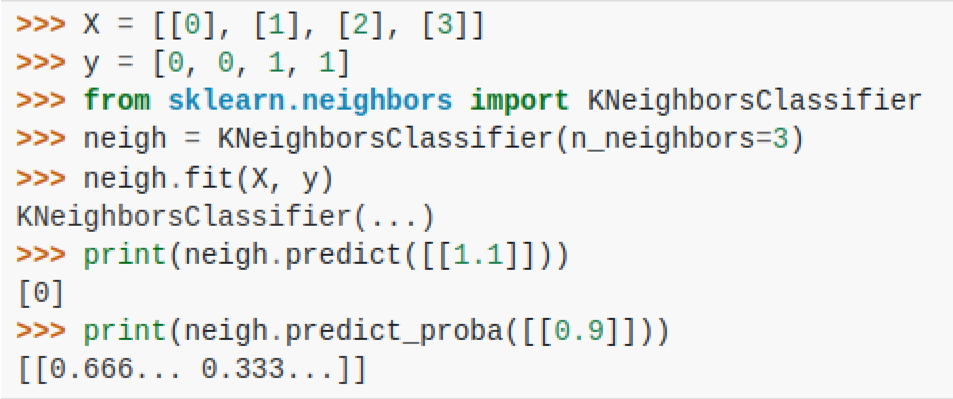
KNN with Scikit-Learn

- https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html



sklearn.neighbors.KNeighborsClassifier

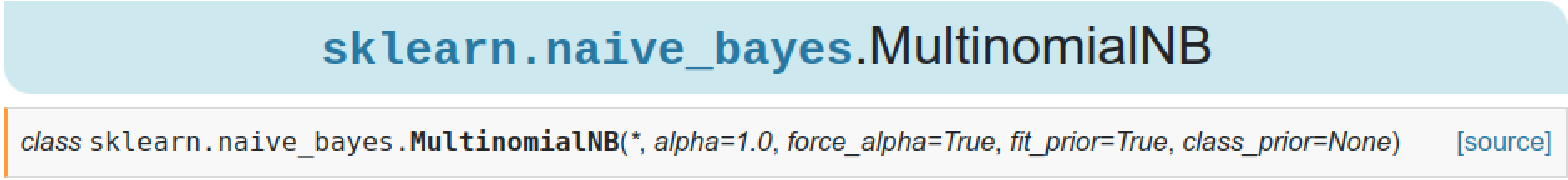
class `sklearn.neighbors.`**`KNeighborsClassifier`**(*n_neighbors=5*, ***, *weights='uniform'*, *algorithm='auto'*, *leaf_size=30*, *p=2*, *metric='minkowski'*, *metric_params=None*, *n_jobs=None*) ¶ [source]



```
>>> X = [[0], [1], [2], [3]]
>>> y = [0, 0, 1, 1]
>>> from sklearn.neighbors import KNeighborsClassifier
>>> neigh = KNeighborsClassifier(n_neighbors=3)
>>> neigh.fit(X, y)
KNeighborsClassifier(...)
>>> print(neigh.predict([[1.1]]))
[0]
>>> print(neigh.predict_proba([[0.9]]))
[[0.666... 0.333...]]
```

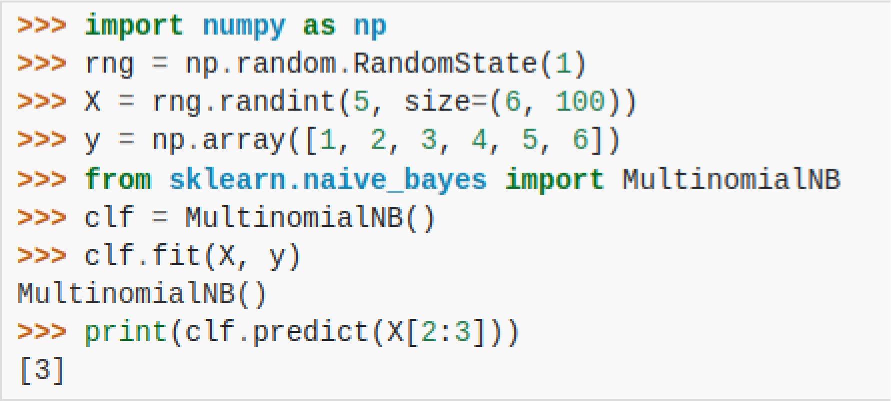
Naive Bayes with Scikit-Learn

- https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html#sklearn-naive-bayes-multinomialnb



sklearn.naive_bayes.MultinomialNB

class `sklearn.naive_bayes.`**`MultinomialNB`**(**, alpha=1.0, force_alpha=True, fit_prior=True, class_prior=None*) [source]



```
>>> import numpy as np
>>> rng = np.random.RandomState(1)
>>> X = rng.randint(5, size=(6, 100))
>>> y = np.array([1, 2, 3, 4, 5, 6])
>>> from sklearn.naive_bayes import MultinomialNB
>>> clf = MultinomialNB()
>>> clf.fit(X, y)
MultinomialNB()
>>> print(clf.predict(X[2:3]))
[3]
```

Exercises:

- Notebooks on the github repository:
  - Notebook with examples:
    - notebooks/session9/examples.ipynb
  - Notebook with exercises:
    - notebooks/session9/exercises.ipynb