

GYAK05

Az órán elkészített függvényekből készíts egy osztályt KNNClassifier néven. A konstruktor 2 paramétert várjon:

- `-k :int`
- `-test_split_ratio :float` (hány százaléka legyen a test set a train-nek)

Ezt a két paramétert tedd példányszintű változónak a konstruktoron belül.

Az órán megvalósított összes függvény szerepeljen az osztályban, viszont párat módosítani kell:

- `train_set_split`: - a `test_split_ratio` értékét ne paraméterként várja, hanem a már konstruktorban átadott példányszintű tagot használja
-ne térjen vissza az `x_train, y_train, x_test, y_test` változókkal hanem tegye példányszintűvé őket
- `euclidean`: -paraméterként csak az `element_of_x`-et várja
-a `self.x_train`-t példányszintű változóból érje el
- `predict`: -paraméterként csak az `x_test`-et várja
-az `x_train, y_train, k` értékét elérjük már belülről az példányszintű változókból
- az eredményt `self.y_preds`-be mentse
- `accuracy`: -ne várjon semmit paraméterként
-az `y_test`-et és az `y_preds`-et a belső példányszintű változókból érje el
- `confusion_matrix`: -ne várjon semmit paraméterként
-az `y_test`-et és az `y_preds`-et a belső példányszintű változókból érje el
-ez legyen a neve a függvénynek

Adjunk az osztályhoz még egy property-t is ami visszaadja a konstruktorban megadott `k` értékét. Neve legyen: `k_neighbors`

A `load_csv` függvény ugyanúgy kívülről várja a `csv_path`-et viszont mivel ebben a függvényben nem használunk semmilyen példány szintű változót vagy függvényt, ezért ezt a függvényt tegyük statikussá.

A shuffle miatt az elején, figyeljünk az `np.random.seed(42)` beállítására, enélkül nem lesz determinisztikus a feladat és ne fogjuk tudni tesztelni.