

Podstawy zastosowania NUMPY

Streszczenie

Moduł NUMPY jest zestawem narzędzi umożliwiającym zaawansowane obliczenia matematyczne. Rozszerza on możliwości PYTHON o nowe typy danych, operacje na nich oraz funkcje przyspieszające obliczenia. Jednym z podstawowych obiektów jest `array`, który stanowi kontener dla macierzy i tablic, ułatwiający m.in. obliczenia na dużych zbiorach danych.

1 Cel

Zapoznanie się z pakietem NUMPY oraz jego zastosowaniem w analizach.

Ważniejsze funkcje: `arange`, `linspace`, `random`, `zeros`, `ones`, `reshape`, `sort`, `argsort`, `dot`, `strides`.

Wykonaj zadania i wyświetl wyniki. Kod zapisuj w jednym pliku do wglądu dla prowadzącego.

Sprawozdanie z zajęć zawierające wykorzystane fragmenty programu, wyniki działań itp. umieść na portalu `elerning` w zakładce `PiAD/Laboratoria/Lab1.elerning`

Proszę się zalogować poświadczeniami używanymi w aplikacji `edziekanat` i potwierdzić udział w kursie.

- **Tablice.**

Tablica jednowymiarowa: `a=np.array([1,2,3,4,5,6,7])`.

Tablica dwuwymiarowa: `b=np.array([[1,2,3,4,5], [6,7,8,9,10]])`.

Wykonaj transpozycję tablicy `b` za pomocą funkcji `transpose`.

Utwórz i wyświetl tablicę składającą się ze 100 elementów za pomocą funkcji `arange`.

Utwórz i wyświetl tablicę składającą się z 10 liczb w zakresie od 0 do 2. Użyj funkcji `linspace`.

Za pomocą funkcji `arange` utwórz tablicę wartości od 0 do 100 ze skokiem wartości co 5.

- **Liczby losowe.**

Za pomocą funkcji `random` utwórz tablicę z 20 liczb losowych rozkładu normalnego, zaokrąglonych do dwu miejsc po przecinku.

Wygeneruj losowo 100 liczb całkowitych w zakresie od 1 do 1000.

Za pomocą funkcji `zeros` i `ones` wygeneruj dwie macierze o rozmiarze 3×2 .

Utwórz macierz losową złożoną z liczb całkowitych z zakresu od 0 do 100 o rozmiarze 5×5 i nadaj jej typ `32bit`.

Zadania.

Wygeneruj tablicę złożoną z losowo wybranych liczb dziesiętnych z zakresu od 0 do 10 (tablica `a`).

Zamień wartości na `integer` i wstaw w nową tablicę (tablicę `b`).

Znajdź funkcję `numpy`, która zaokrągli tablicę `a` do liczb całkowitych. Zamień je następnie na typ `integer`.

Porównaj wyniki z `a` i `b` i wyświetl wniosek za pomocą polecenia `print`.

- **Selekcja danych.**

wprowadź tablicę `b=np.array([[1,2,3,4,5], [6,7,8,9,10]],dtype=np.int32)`.

Za pomocą funkcji `ndim` sprawdź ile wymiarów ma tablica `b`.

Za pomocą funkcji `size`, sprawdź z ilu elementów składa się tablica `b`.

Wybierz wartości dwa i cztery z tablicy `b`.

Wybierz pierwszy wiersz tablicy `b`.

Wybierz wszystkie wiersze z kolumny pierwszej tablicy `b`.

Wygeneruj macierz losową o rozmiarze 20×7 , złożoną z liczb całkowitych w przedziale od 0 do 100. Wyświetl wszystkie wiersze dla czterech pierwszych kolumn.

- **Działania matematyczne i logiczne.**

Stwórz dwie macierze liczb całkowitych z przedziału od 1 do 10 o rozmiarach 3×3 (`a` i `b`).

Dodaj (za pomocą `+` oraz funkcji `add`), odejmij (za pomocą `-` oraz funkcji `subtract`), pomnóż (za pomocą `*` oraz funkcji `multiply`, `dot`, `matmul`), podziel (za pomocą `/` oraz funkcji `divide`), spotęguj (za pomocą `**` oraz funkcji `power`) macierzy `a` i `b` przez siebie.

Sprawdź czy wartości macierzy `a` są większe lub równe 4.

Sprawdź czy wartości macierzy `a` są większe bądź równe 1 i mniejsze bądź równe 4.

Znajdź funkcję w `numpy` do obliczenia sumy głównej przekątnej macierzy `b`.

- **Dane statystyczne.**

Oblicz sumę, wartość minimum, maksimum, odchylenie standardowe macierzy `b`.

Oblicz średnią dla wierszy w macierzy `b`.

Oblicz średnią dla kolumn macierzy `b`.

- **Rzutowanie wymiarów za pomocą `rehape` lub `resize`.**

Utwórz tablicę składającą się z 50 liczb (np., za pomocą funkcji `arange`).

Za pomocą funkcji `reshape` utwórz z tej tablicy macierz o wymiarach 10×5 .

Zrób to samo za pomocą funkcji `resize`.

Sprawdź do czego służy komenda `ravel`. Napisz wniosek.

Stwórz dwie tablice o rozmiarach 5 i 4 (np., za pomocą funkcji `arange`) i dodaj je do siebie. W tym celu sprawdź do czego służy funkcja `newaxis` i wykorzystaj ją. Napisz wniosek.

- **Sortowanie danych.**

Sprawdź składnie funkcji `sort` i `argsort`.

Wprowadź macierz `a=np.random.randint(0, 100, size(5,5))`.

Posortuj wiersze rosnąco.

Posortuj kolumny malejąco.

Zadania.

Wprowadź macierz

```
b=np.array([(1,'MZ','mazowieckie'),  
(2,'ZP','zachodniopomorskie'), (3,'ML','małopolskie')])
```

Posortuj dane rosnąco po kolumnie 2.

Wyświetl nazwę województwa Zachodniopomorskiego.

2 Zadania podsumowujące

1. Utwórz macierz składającą się z pięciu kolumn i 10 wierszy losowo wybranych liczb całkowitych z zakresu od 0 do 100 i policz sumę głównej przekątnej tej macierzy, używając funkcji `trace` a następnie wyświetl wartości używając funkcji `diag`.
2. Utwórz dwie tablice wymiaru 5×5 z losowo wybranych liczb dziesiętnych z rozkładu normalnego i przemnoż je przez siebie.

3. Utwórz dwie tablice z losowo wybranych liczb całkowitych w zakresie od 1 do 100. Stwórz z nich macierze o 5 kolumnach i dodaj te macierze do siebie.
4. Stwórz dwie macierze: jedną o 5 kolumnach i 4 wierszach oraz drugą o 4 kolumnach i 5 wierszach. Dodaj je do siebie używając transformacji wymiarów za pomocą jednego ze znanych sposobów.
5. Pomnóż kolumny 3 i 4, stworzonych przez siebie macierzy.
6. Wygeneruj dwie macierze o rozkładzie normalnym (`np.random.normal`) i jednostajnym(`np.random.uniform`).
Policz wartości średnie, odchylenie standardowe, wariancje, sumy, wartości minimalne i maksymalne. Wyniki wyświetl.
7. Wygeneruj dwie macierze kwadratowe **a** i **b** (o wymiarach zdecyduj się samodzielnie), pomnóż je przez siebie używając (`a*b`) oraz funkcji `dot`. zobacz Jaka jest różnica? Napisz kiedy warto wykorzystać funkcję `dot`?
8. Sprawdź funkcję `strides` oraz `as_strided`. Zastosuj je do wyboru danych z macierzy np. 5 kolumn z trzech pierwszych wierszy.
9. Wygeneruj dwie tablice **a** i **b**.
Połącz je z użyciem funkcji `vstack` i `stack`. Czym one się różnią? Zastanów się i napisz, w jakich przypadkach warto je zastosować?
10. Użyj funkcji `strides` oraz `as_strided` do obliczenia wartości maksymalnej bloków danych z macierzy (zob. rysunek)

0	1	2	3	4	5
6	7	8	9	10	11
12	13	14	15	16	17
18	19	20	21	22	23

Korzystano:

<https://numpy.org/>

<https://github.com/yongtwang/engineering-python>