

Библиотека Numpy

№ урока: 5 **Курс:** Python Advanced

Средства обучения: Python3.6, PyCharm

Обзор, цель и назначение урока

Научить студентов использовать библиотеку numpy для решения математических задач, таких как работа с матрицами и векторами. Разобраться в особенностях данной библиотеки.

Изучив материал данного занятия, учащийся сможет:

- Разобраться с назначением и особенностями библиотеки numpy.
- Понимать, что такое ndarray и его отличия от списков (list) Python.
- Разбираться в основных типах данных в numpy.
- Использовать возможности данной библиотеки.

Содержание урока

1. Назначение и особенности библиотеки Numpy.
2. Что такое ndarray и отличие ndarray от списков (list) Python.
3. Примеры создания ndarray, индексация и срезы.
4. Основные операции и возможности библиотеки numpy.
5. Практика.

Резюме

- Библиотека предназначена для математических вычислений для решения научных задач. Она предоставляет ряд удобных интерфейсов для работы с векторами и многомерными массивами и широко используется в Data Science.
- Для установки данной библиотеки используем менеджер пакетов pip (pip install numpy).
- Основным типом данных для работы с N-мерными массивами является тип данных ndarray, который содержит в себе следующие составляющие:
 - Количество измерений (ndim)
 - Форма (shape)
 - Количество всех элементов массива (size)
 - Тип элементов массива (dtype)
 - Размер компонентов массива в байтах (itemsize)
 - Значения элементов массива (data)
- В данном модуле реализован набор типов данных, которые можно использовать для более оптимального решения. Пример основных типов данных:
 - Целые числа (int_: int8, int16, int32, int64, uint8, uint16, uint32, uint64)
 - Числа с плавающей точкой (float_: float16, float32, float64)
 - Комплексные числа (complex_: complex64, complex128)
 - Булевы значения (bool_)
- Типы данных имеют связь с поддерживаемыми типами данных языка Python, а именно возможно конвертировать в обе стороны. Но необходимо понимать, что, выбирая частный случай типа данных из Numpy, в Python будет использоваться обобщенный тип, например при конвертации в список элементов uint8 будет использован int.

- Numpy предоставляет удобный интерфейс для работы с индексами и срезами, который позволяет обращаться к массивам конкретных размерностей и выбирать необходимые данные. В целом, механизм похож на работы со срезами и индексами `list` в Python.
- Здесь же реализован ряд математических операций с многомерными массивами (например матрицами), которые могут упростить вычисления. Нет необходимости выполнять перебор элементов матрицы и итерироваться по ним с использованием циклов (например, для умножения всех элементов на число).
- Широкий набор математических функций из области линейной алгебры также доступен для программиста, что не требует глубокого знания данных алгоритмов. А также в рамках данной библиотеки доступны некоторые математические константы, которые удобно использовать в своих вычислениях (например: π или e)

Закрепление материала

- Что такое библиотека Numpy?
- Что такое `ndarray`?
- Перечислите основные компоненты (атрибуты) типа `ndarray`?
- Какие основные типы данных доступны в библиотеке Numpy для математических операций?
- В чем отличие типов данных `uint8` от `int8`?
- Из чего состоит тип `complex64`?
- Как каждый элемент матрицы возвести в квадрат?
- Как перемножить две матрицы?
- Как найти определитель матрицы?
- Как транспонировать матрицу?
- Как использовать два разных `ndarray`, но ссылаться на одни и те же данные?
- Как создать полный дубликат `ndarray`-массива без каких-либо ссылок?
- Какая функция используется для чтения данных и преобразования их в `ndarray` из файла/буфера данных?
- В чем отличие `zeros` от `zeros_like`?
- В чем отличие `zeros` от `empty`?
- Для каких задач используется суффикс `_like` в именовании функция библиотеки Numpy, какого общее правило?

Дополнительное задание

Задание 1

Создайте двумерный массив размерностью 25 x 25 и умножьте его на единичную матрицу.

Задание 2

Создайте матрицу размерностью 12 x 12, транспонируйте её, возведите каждый элемент матрицы в квадрат и умножьте на 2. Вычитите из данной матрицы единичную матрицу и распечатайте результат на консоль.

Задание 3

Сгенерируйте 100 элементов от 1 до 20 используя средства Numpy.

Задание 4

Сгенерируйте массив значений от 1 до 300 используя средства Numpy. Измените размерность данного массива на 3 x 100, а затем создайте новый массив с теми же значениями, но размерностью 6 x 50.

Самостоятельная деятельность учащегося

Задание 1

Сохраните в файл данные об изменениях температуры в течение трёх месяцев (сгенерируйте случайные значения- 90 элементов) и, используя средства Numpy, загрузите эти данные из файла в ndarray (плоский список, значения расположены по два в одной строке). Преобразуйте получившийся набор данных в плоский массив ndarray. Выведите максимальное, минимальное, среднее значения температур среди существующих значений. Для решения данной задачи используйте оптимальный тип данных Numpy, учитывая, что замеры производились в летнее время и температура не падает ниже +15 С. Разбейте общую последовательность на три равных, чтобы получился двумерный массив 3 x 30 и сохраните последние 5 значений каждого месяца в текстовый файл (массив размерности 3 x 5).

Рекомендуемые ресурсы

Официальный сайт библиотеки Numpy

<https://docs.scipy.org/doc/numpy/>

Линейная алгебра библиотеки Numpy

<https://docs.scipy.org/doc/numpy/reference/routines.linalg.html>