Здравсвуйте, мы продолжаем цикл лекций по библиотекам для машинного обучения. И сегодня мы поговорим о библиотеке SciPy.

Слайд 2

SciPy — это библиотека для языка Python, основанная на расширении NumPy, но для более глубоких и сложных научных вычислений, анализа данных и построения графиков.

SciPy в основном написана на Python и частично на языках C, C++ и Fortran, поэтому отличается высокой производительностью и скоростью работы.

**Слайд 3**

**Для чего нужна SciPy**

SciPy используют специалисты по Data Science, Big Data, аналитики данных , а также математики и ученые в следующих целях:

* для сложных математических расчетов, которые тяжело произвести вручную или с помощью калькулятора;
* проведения научных исследований, где требуется использование продвинутой математики;
* глубокого анализа данных, интерполяции и других методов работы с информацией;
* машинного обучения и создания моделей искусственного интеллекта, прогнозирования и построения моделей;
* формирования двумерных и трехмерных графиков, которые можно потом визуализировать (уже при помощи других библиотек).

**Слайд 4**

**Отличия SciPy от NumPy**

NumPy — это еще одна научная библиотека, на базе которой реализована SciPy.

Ее особенность — использование специальных структур данных , а именно многомерных массивов, в которых хранится информация.

С этим типом данных также работает SciPy.

NumPy содержит данные массивов и такие операции, как индексация, сортировка , а SciPy состоит из числового кода.

Давайте рассмотрим основные различия между этими библиотеками:

* в SciPy гораздо больше функций и методов, чем в NumPy;
* NumPy ориентирована на базовые вычисления и простую работу с матрицами, SciPy предназначена для глубокого научного анализа;
* NumPy не имеет дополнительных зависимостей, вместе с библиотекой не нужно ничего устанавливать. SciPy требует установки NumPy для корректной работы.

SciPy расширяет возможности NumPy, а некоторые частые действия в ней реализованы как отдельные функции, поэтому библиотека сильно упрощает работу со сложными задачами с использованием продвинутой математики.

При этом для ряда более простых задач она избыточна. Будет достаточно NumPy или других библиотек.

**С какими программами используют SciPy**

SciPy используют совместно с [NumPy](https://blog.skillfactory.ru/glossary/numpy/)и библиотекой Matplotlib, для построения графиков и работы с ними.

Часто вместе со SciPy применяют программное обеспечение IPython -это интерактивная оболочка и интерфейс для работы с кодом.

Существуют «научные дистрибутивы» для Python, в которые входит интерпретатор для языка и набор библиотек для математических и инженерных вычислений. В дистрибутивы включается и SciPy. Примеры таких пакетов — Anaconda и Canopy

**Возможности SciPy**

Библиотека позволяет выполнять операции высшей математики — математического анализа и линейной алгебры, различные преобразования, инженерные расчеты. В пакете есть функции для обработки изображений и построения графиков. Отличительная черта SciPy — использование эффективных структур данных, которые создаются с помощью NumPy.

**Работа с продвинутыми математическими операциями**

В SciPy есть функции из линейной алгебры, вычисления интегралов и решения дифференциальных уравнений, реализация различных сложных алгоритмов. Операций и методов больше, чем в NumPy, они более трудоемки. Зато их написание в виде кода очень простое.

**Математическая физика**

Так называют науку на стыке двух дисциплин, которая описывает физические явления с помощью математических моделей. В SciPy входят специальные функции, уравнения и константы, использующиеся в физике. В реальной работе это позволяет моделировать различные физические явления и эффекты с помощью программирования.

**Обработка сигналов**

SciPy позволяет обрабатывать, фильтровать и преобразовывать полученные на вход сигналы: различные волны (чаще всего звуковые), поток света и т.д. Фото и видео тоже относятся к сигналам. Например, с помощью SciPy можно очистить от шумов речь или обработать изображение для последующего распознавания. Это важно в машинном обучении и смежных отраслях.

**Ввод и вывод файлов**

В библиотеке SciPy есть свои методы для работы с данными. Они позволяют загружать информацию, редактировать, сохранять в переменную или в файл. SciPy отображает сложные объекты как многомерные структуры данных — это эффективное и удобное для научной работы представление. Библиотека поддерживает форматы данных MATLAB, поэтому специалист может работать сразу с несколькими стеками технологий.

**Работа с изображениями и графиками**

Возможности для построения красивых графиков в SciPy не такие обширные, как в Matplotlib или Seaborn, но для простейшей визуализации результата их бывает достаточно. Библиотека умеет работать и с многомерными изображениями, а не только с 2D.

**Пакеты SciPy**

Возможности библиотеки распределены по нескольким модулям, или пакетам, которые объединены назначением. Это нужно, чтобы специалист мог подключить вместо полной библиотеки только требуемую часть. Так код будет эффективнее, а его написание — удобнее.

Основные модули SciPy для научных вычислений:

* scipy.special — специальные функции, возможности и понятия из математической физики;
* scipy.integrate — функции для численного интегрирования и решения обыкновенных дифференциальных уравнений;
* scipy.optimize — алгоритмы оптимизации (в т.ч. минимизации математических функций);
* scipy.interpolate — методы для интерполяции, т.е. для приближенного нахождения какой-либо величины по уже известным отдельным ее значениям;
* scipy.fft — преобразования Фурье (операции, сопоставляющие одной функции вещественной переменной другую функцию вещественной переменной);
* scipy.signal — методы для обработки и преобразования сигналов;
* scipy.linalg — операции линейной алгебры. Модули с этим названием есть и в NumPy, и в SciPy, но их возможности различаются. В SciPy больше продвинутых функций и методов;
* scipy.sparse.csgraph — методы для работы с разреженными графами, особыми структурами данных;
* scipy.spatial — функции для работы с пространственными структурами данных и алгоритмами;
* scipy. stats — операции для статистических расчетов;
* scipy.ndimage — обработка многомерных изображений;
* scipy.io — ввод и вывод, загрузка и сохранение файлов.

Пакеты необходимо импортировать отдельно.

**Особенности SciPy**

**Бесплатное распространение**

Исходный код библиотеки открытый, она распространяется бесплатно по лицензии BSD. Это значит, что SciPy можно скачивать, пользоваться в личных или профессиональных целях, просматривать исходный код функций. Модули, написанные с помощью SciPy, можно встраивать в проприетарное, закрытое и платное программное обеспечение.

**Низкий порог входа**

Работать со SciPy сможет новичок в Python. Для использования библиотеки не обязательно быть профессиональным разработчиком, но важно иметь хороший математический аппарат и разбираться в научных вычислениях. На официальном сайте представлена [документация](https://docs.scipy.org/doc/) по SciPy на английском языке.

**Быстрое исполнение кода**

Python критикуют за медленную работу в части трудоемких вычислений. Поэтому разработчики SciPy написали библиотеку не только на нем. Сложные вычисления, которые требуют высокой скорости работы, реализованы на Fortran, C и C++. Поэтому код на SciPy быстро исполняется и не отнимает много ресурсов.

**Как начать пользоваться SciPy**

Установить библиотеку можно несколькими способами:

* Скачать научный дистрибутив. Этот способ рекомендуют разработчики.
* Скачать и установить IDE, затем загрузить через нее SciPy. Среды для программирования на Python позволяют легко подключать библиотеки. Примеры IDE — Pyzo или PyCharm.
* Воспользоваться менеджером pip. Это программа для загрузки, обновления и удаления официальных пакетов Python. Она встроена в язык программирования и автоматически устанавливается вместе с ним.
* Установить SciPy через Anaconda — дистрибутив языков программирования Python и R с набором свободных библиотек для Data Science и машинного обучения
* Для пользователей Linux или MacOS — использовать встроенный файловый менеджер. В Ubuntu и Debian это apt-get, для Fedora — dnf, в MacOS — brew.

После этого достаточно подключить библиотеку к Python-файлу с помощью команды *import scipy* в начале кода. Если из SciPy нужен только один пакет, можно воспользоваться командой *from scipy import pack\_name.* Таким же образом можно подключить несколько пакетов — их надо будет перечислить через запятую.