

УДК 004.4

*Проценко А. С., студентка
I курсу спеціальності 122
Горяшин А. С.,
асистент кафедри інформаційних технологій*

ВИКОРИСТАННЯ БІБЛІОТЕКИ SCIPY У НАУКОВИХ ДОСЛІДЖЕННЯХ

Донецький національний університет імені Василя Стуса, м. Вінниця

Бібліотека Python – велика збірка вбудованих модулів, які мають завчасно скомпільований код, включаючи класи і методи, які позбавляють необхідності впроваджувати розробнику код з самого початку. Дієвим для застосування складних алгоритмів робить його неважкий синтаксис, який також скорочує час навчання і полегшує розуміння експертами різного рівня досвіду. Бібліотеки сприяють написанню коду та створенню моделі в Machine Learning (машинне навчання) та Data Science (наука про дані). Ці збірки модулів допомагають користувачу кілька разів застосовувати код. Таким чином, імпортувавши потрібну збірку модулів, він зможе реалізувати необхідну функцію у програмі. Розробники використовують різні збірки Python, що класифіковані на модулі для створення моделей, видобутку, обробки, аналізу чи візуалізації даних.

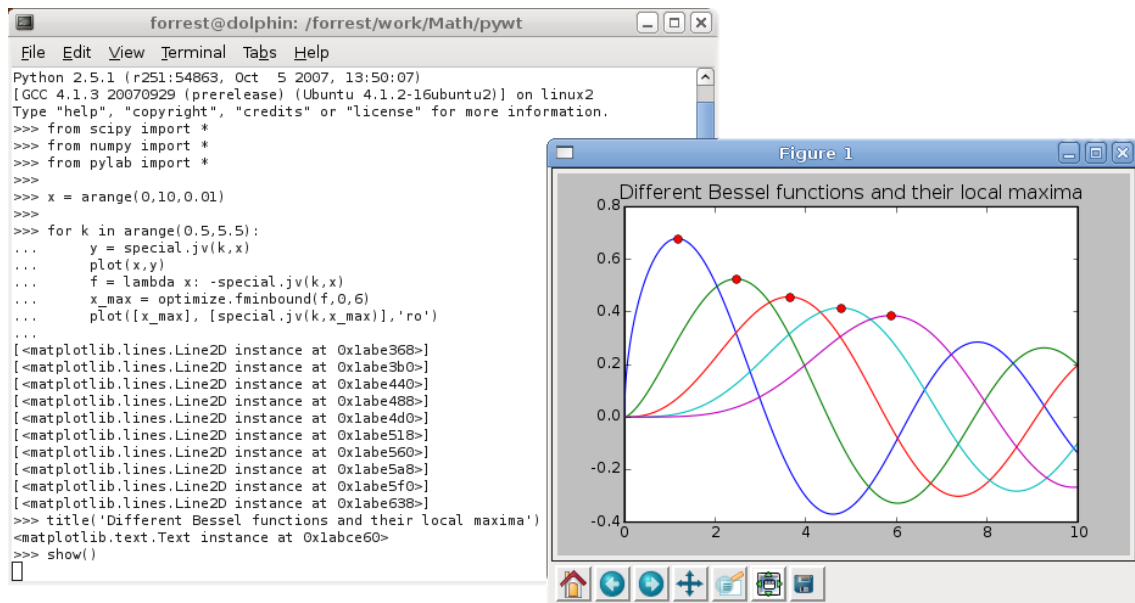
Розглянемо детальніше одну з бібліотек – SciPy (Scientific Python). Вона є відкритою бібліотекою, що містить високоякісні наукові інструменти для мови програмування Python. У своєму складі має модулі для інтегрування, оптимізації, статистики, інтерполяції, обробки сигналів чи зображень, генетичних алгоритмів, а також розв’язування алгебраїчних рівнянь та інших завдань, що вирішуються в науці й інженерній розробці. Ця збірка створюється для тих користувачів, що і MATLAB чи Scilab. Мінусом SciPy є його єдина спрямованість на числові об’єкти та алгоритми, тому він не може надати жодної функції побудови, однак для таких задач можна застосовувати бібліотеку Matplotlib, яка є подібним засобом виведення графіки як у MATLAB. [1]

Загальна організація даних в SciPy – це багатовимірний масив, який реалізується модулем NumPy (старі версії Scientific Python застосовували для цієї мети Numeric).

Оглянемо загальнодоступні підпакети в SciPy:

- «constants»: константи та коефіцієнти перерахунку;
- «cluster»: операції для векторного перетворення (квантування);
- «fftpack»: операції дискретної математики для перетворення Фур'є;
- «integrate»: операції інтегрування;
- «interpolate»: операції інтерполяції;
- «io»: операції введення та виведення даних;

- «lib»: обгортка, що є проміжним шаром для зовнішніх бібліотек;
- «linalg»: операції лінійної алгебри;
- «misc»: різні сервісні програми для керування;
- «ndimage»: інструменти для обробки зображень;
- «optimize»: інструменти оптимізації;
- «sandbox»: експериментальний код;
- «signal»: інструменти для обробки сигналів;
- «sparse»: інструменти підтримки розріджених матриць;
- «spatial»: k-вимірні дерева, метод найближчих k-сусідів, метрики;
- «special»: спеціальні функції;
- «stats»: математично-статистичні операції;
- «weave»: інструменти, що дозволяють включити код C/C++ в код мови програмування Python.



Малюнок 4 – Приклад застосування бібліотеки SciPy для побудови графіка функцій Бесселя і знаходження їхніх локальних максимумів

Розширення функціональності бібліотеки можна здійснити за допомогою деяких інших інструментів:

- База даних: взаємодія бібліотеки SciPy з ієрархічною БД PyTables, створена для оперативного керування масивними об'ємами даних, збережені у форматі HDF5.
- Оптимізація: незважаючи на те, що SciPy містить власний пакет, математичний програмний засіб OpenOpt дає доступ до інших розв'язувачів та пакетів оптимізації.
- Розгорнутий аналіз даних: Scientific Python може забезпечити інтерфейс до статистичного модуля R, що призначений для детального розгляду показників з допомогою RPy.
- Інтерактивна оболонка: бібліотека IPython сприяє налагодженню програми і написанню коду в стилі, схожому на MATLAB.

- Символьна математика: бібліотеки PyDSTool, Symbolic, SymPy, сприяють роботі з даним видом математики.

- Графіка: Matplotlib - рекомендований пакет для створення двовимірної графіки та для 3D-візуалізації засоби: Python Imaging Library і MayaVi. [2]

Отже, можна зробити висновок, що використання бібліотеки SciPy є ефективним при виконанні різних алгоритмів для наукових досліджень, однак не завжди може самостійно функціонувати, тому потребує підключення деяких інших інструментів.

Список літератури:

1. Найкращі бібліотеки Python для спеціалістів із обробки даних. URL: <https://techukraine.net/%D0%BD%D0%B0%D0%B9%D0%BA%D1%80%D0%B0%D1%89%D1%96-%D0%B1%D1%96%D0%B1%D0%BB%D1%96%D0%BE%D1%82%D0%B5%D0%BA%D0%B8-python-%D0%B4%D0%BB%D1%8F-%D1%81%D0%BF%D0%B5%D1%86%D1%96%D0%B0%D0%BB%D1%96%D1%81%D1%82/> (дата звернення: 11.11.2022)
2. SciPy. URL: <https://uk.wikipedia.org/wiki/SciPy> (дата звернення: 11.11.2022)

УДК 004.67

*Семенюк А. М., студент гр. Б21-д/122Б2-II
Горяшин А.С., асистент кафедри інформаційних технологій*

ФОРМУВАННЯ ІНФОРМАЦІЙНОЇ МОДЕЛІ З ВИКОРИСТАННЯМ ООП

Донецький національний університет імені Василя Стуса, м. Вінниця

Використання електронного документообігу, це не лише сучасний рух по зменшенню паперових документів актуальний з екологічної точки зору, а й можливість пришвидчити підготовку документів, наявність завжди актуальної інформації, зменшення (в ідеалі – виключення) дублювання відомостей, розпаралелювання потоків відображення та обробки даних.

Використання останнього методу, особливо в медичній галузі, питання досить актуальне, а саме в теперішній час.

Це пов'язане з тим, що процес прийняття рішень:

- ✓ при лікуванні,
- ✓ експертному обстеженні пацієнтів,
- ✓ підготовці медичних висновків та заключень

в медицині є колективне напрацювання декількох спеціалістів, для унеможливлення хибних результатів. На сучасному етапі розвитку галузі та можливостях комунікації – експертизи, консиліуми, дистанційні огляди, та ін. – це рішення що приймається декількома людьми, які не завжди знаходяться в