

ИНТЕГРИРОВАНИЕ УРАВНЕНИЙ В PYTHON С ИСПОЛЬЗОВАНИЕМ БИБЛИОТЕК SYMPY И SCIPY

ЖОРОЕВА Мунара Омурбековна

преподаватель

АБДИРАШИТОВ Чынгыз Абдирашитович

магистрант

Ошский государственный университет

г. Ош, Кыргызстан

В данной статье рассматриваются методы интегрирования уравнений с использованием языка программирования Python и его специализированных библиотек SymPy и SciPy. Символьная библиотека SymPy позволяет выполнять аналитическое (точное) интегрирование функций и уравнений, что делает её удобным инструментом для исследовательских и учебных задач. В то же время библиотека SciPy реализует численные методы интегрирования, обеспечивающие высокую точность и скорость при работе с функциями, не имеющими аналитического решения. В работе представлены теоретические основы символьного и численного интегрирования, сравнительный анализ методов, а также примеры программных реализаций на Python.

Ключевые слова: интегрирование, Python, SymPy, SciPy, численные методы, символьная математика.

Введение. В современном научном и инженерном моделировании задачи интегрирования уравнений занимают центральное место. Интегралы применяются в физике, математике, экономике и информатике для нахождения площадей, объёмов, решения дифференциальных уравнений и моделирования непрерывных процессов.

С развитием вычислительных технологий язык Python стал одним из наиболее популярных инструментов для проведения математических расчётов и численного анализа. Благодаря своей простоте, открытости и широкому набору библиотек, Python позволяет исследователям и инженерам эффективно решать задачи, которые ранее требовали специализированного программного обеспечения.

Особое место среди библиотек Python занимают SymPy и SciPy.

– SymPy – библиотека для символьной математики, предоставляющая возможности аналитического интегрирования, дифференцирования и решения уравнений [1].

– SciPy реализует широкий спектр численных методов, включая функции численного интегрирования и решения обыкновенных дифференциальных уравнений [2].

Таким образом, интеграция возможностей SymPy и SciPy позволяет охватить как теоретические, так и прикладные аспекты вычислительной математики, обеспечивая исследова-

теля универсальными инструментами для решения задач различного уровня сложности (<https://docs.sympy.org/latest/>).

Методы интегрирования в Python.

Интегрирование функций и уравнений в Python может быть реализовано двумя основными подходами (<https://www.python.org/>):

1. Символьным (аналитическим) методом с использованием библиотеки SymPy;
2. Численным методом с использованием библиотеки SciPy.

Символьное интегрирование с использованием библиотеки SymPy

Библиотека SymPy предназначена для символьных вычислений и позволяет получать точное выражение интеграла в виде формулы [1].

Пример неопределённого интеграла:

```
from sympy import symbols, integrate
x = symbols('x')
expr = x**2 + 3*x + 2
result = integrate(expr, x)
print(result)
```

Результат:

$$\int (x^2 + 3x + 2)dx = \frac{x^3}{3} + \frac{3x^2}{2} + 2x + C$$

Численное интегрирование с использованием библиотеки SciPy

Для численного интегрирования, когда аналитическое решение отсутствует, применяется модуль `scipy.integrate` (<https://docs.scipy.org/doc/scipy/>).

Пример использования функции `quad()`:

```
from scipy.integrate import quad
import numpy as np
f = lambda x: x**2
result, error = quad(f, 0, 2)
print(result, error)
```

Результат: $\int_0^2 x^2 dx \approx 2.6666666667$

Метод `quad()` основан на адаптивных квадратичных формулах Гаусса–Кронрода [2], обеспечивая высокую точность вычислений.

Практическая реализация и сравнение методов

Рассмотрим функцию:

$$f(x) = e^{-x^2}$$

Известно, что:

$$\int_0^{\infty} e^{-x^2} dx = \frac{\sqrt{\pi}}{2}$$

```
from sympy import symbols, integrate, exp, oo, sqrt, pi
```

```
x = symbols('x')
```

```
integrate(exp(-x**2), (x, 0, oo))
```

Результат:

$$\frac{\sqrt{\pi}}{2}$$

Решение с SciPy

```
from scipy.integrate import quad
import numpy as np
f = lambda x: np.exp(-x**2)
quad(f, 0, np.inf)
```

Результат:

$$0.8862269255 \approx \frac{\sqrt{\pi}}{2}$$

Обе библиотеки дают одинаковый результат с различными подходами – аналитическим (SymPy) и численным (SciPy) [1; 2].

Заключение. В работе проведено исследование методов интегрирования с использованием Python и библиотек SymPy и SciPy. Библиотека SymPy показала высокую точность аналитических вычислений, а SciPy продемонстрировала надёжность и эффективность численного интегрирования.

Совместное применение этих инструментов позволяет объединить преимущества символического и численного анализа, что делает Python мощным средством для математического моделирования и инженерных расчётов.

ЛИТЕРАТУРА

1. Meurer A., Smith C.P., Paprocki M. и др. SymPy: symbolic computing in Python // PeerJ Computer Science. 2017. № 3: e103.
2. Virtanen P. и др. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python // Nature Methods. 2020. Т. 17. P. 261-272.

INTEGRATION OF EQUATIONS IN PYTHON USING THE SYMPY AND SCIPY LIBRARIES

ZHOROEVA Munara Omurbekovna

Lecturer

ABDIRASHITOV Chyngyz Abdirashitovich

Undergraduate Student

Osh State University

Osh, Kyrgyzstan

This article discusses equation integration methods using the Python and its specialized libraries, SymPy and SciPy. The symbolic SymPy library enables analytical (exact) integration of functions and equations, making it a convenient tool for research and educational purposes. Meanwhile, the SciPy library implements numerical integration methods that ensure high accuracy and speed when working with functions that do not have an analytical solution. The paper presents the theoretical foundations of symbolic and numerical integration, a comparative analysis of the methods, and examples of Python software implementations.

Keywords: integration, Python, SymPy, SciPy, numerical methods, symbolic mathematics.