

## **Analisis Penerapan Python Dalam Perhitungan Turunan Fungsi**

**Muhammad Robby<sup>1\*</sup>, Borneo Paradis Anwar<sup>2</sup>, Mega Nurmala<sup>3</sup>, Muhammad Listanto<sup>4</sup>, Perani Rosyani<sup>5</sup>**

<sup>1,2,3,4,5</sup> Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Indonesia

\*Email: <sup>1</sup>[robbymuh95@gmail.com](mailto:robbymuh95@gmail.com), <sup>2</sup>[borneoanwar23@gmail.com](mailto:borneoanwar23@gmail.com), <sup>3</sup>[meganurmala01@gmail.com](mailto:meganurmala01@gmail.com), <sup>4</sup>[mlistanto29@gmail.com](mailto:mlistanto29@gmail.com), <sup>5</sup>[dosen00837@unpam.ac.id](mailto:dosen00837@unpam.ac.id)

(\* : coresponding author)

**Abstrak**-Perkembangan teknologi informasi telah menghadirkan perubahan signifikan dalam berbagai bidang ilmu, termasuk dalam matematika komputasional. Salah satu aspek penting dari matematika komputasional adalah perhitungan turunan fungsi, yang memiliki aplikasi luas dalam berbagai disiplin ilmu, seperti optimasi, pemodelan dinamika sistem, dan kecerdasan buatan. Penelitian ini membahas penerapan bahasa pemrograman Python dalam konteks perhitungan turunan fungsi. Python, sebagai bahasa pemrograman yang populer dan mudah dipahami, telah menjadi pilihan utama dalam pengembangan perangkat lunak matematika komputasional. Dengan menggunakan pustaka-pustaka seperti NumPy dan SymPy, Python memungkinkan perhitungan turunan fungsi dengan efisien dan akurat. Studi ini mencakup implementasi algoritma numerik untuk perhitungan turunan numerik dan metode simbolik menggunakan Python. Analisis kinerja dari implementasi ini dilakukan untuk mengevaluasi kecepatan eksekusi, akurasi, dan kehandalan metode yang diusulkan. Selain itu, dicakup juga perbandingan dengan metode tradisional untuk perhitungan turunan fungsi. Hasil penelitian menunjukkan bahwa penerapan Python dalam perhitungan turunan fungsi memberikan hasil yang memuaskan baik dari segi kecepatan eksekusi maupun akurasi. Python membuktikan dirinya sebagai alat yang efektif untuk menangani perhitungan matematika kompleks, memberikan fleksibilitas dan kemudahan dalam pengembangan perangkat lunak matematika komputasional. Penelitian ini memberikan kontribusi pada pemahaman lebih lanjut tentang penerapan Python dalam matematika komputasional khususnya dalam konteks perhitungan turunan fungsi. Implikasi dari penelitian ini dapat membantu pengembang perangkat lunak, peneliti, dan praktisi dalam memilih dan menggunakan alat yang tepat untuk menangani perhitungan turunan fungsi dalam berbagai konteks aplikasi.

**Kata Kunci:** Python, Perhitungan Derivatif, Matematika Komputasi, Metode Numerik, Komputasi Simbolik

**Abstract**—The development of information technology has brought about significant changes in various fields of science, including computational mathematics. One important aspect of computational mathematics is the calculation of derivatives of functions, which has wide applications in various disciplines, such as optimization, system dynamics modeling, and artificial intelligence. This research discusses the application of the Python programming language in the context of calculating function derivatives. Python, as a popular and easy-to-understand programming language, has become a top choice in computational mathematics software development. By using libraries such as NumPy and SymPy, Python allows calculating derivatives of functions efficiently and accurately. This study covers the implementation of numerical algorithms for calculating numerical derivatives and symbolic methods using Python. Performance analysis of this implementation is performed to evaluate the execution speed, accuracy, and reliability of the proposed method. In addition, a comparison with traditional methods for calculating function derivatives is also included. The research results show that the application of Python in calculating function derivatives provides satisfactory results both in terms of execution speed and accuracy. Python proves itself to be an effective tool for handling complex mathematical calculations, providing flexibility and ease in the development of computational mathematics software. This research contributes to further understanding of the application of Python in computational mathematics, especially in the context of calculating derivatives of functions. The implications of this research can help software developers, researchers, and practitioners in selecting and using appropriate tools to handle function derivative calculations in various application contexts.

**Keywords:** Python, Derivative Calculations, Computational Mathematics, Numerical Methods, Symbolic Computation

### **1. PENDAHULUAN**

Matematika komputasional memainkan peran krusial dalam memecahkan permasalahan matematika yang kompleks dengan menggunakan algoritma dan perangkat lunak komputer. Salah satu aspek yang sangat penting dalam matematika komputasional adalah perhitungan turunan fungsi.

Turunan fungsi memiliki aplikasi yang luas dalam berbagai bidang, termasuk dalam pemodelan matematis, optimisasi, analisis numerik, dan kecerdasan buatan.

Dalam konteks perhitungan turunan fungsi, pemilihan bahasa pemrograman menjadi faktor kritis dalam pengembangan perangkat lunak. Python, sebagai bahasa pemrograman yang mudah dipahami dan memiliki ekosistem pustaka yang kuat, telah menjadi pilihan utama bagi banyak peneliti dan praktisi dalam matematika komputasional.

Penelitian ini bertujuan untuk menyelidiki dan menganalisis penerapan Python dalam perhitungan turunan fungsi. Dengan menggabungkan kelebihan bahasa pemrograman Python dan pustaka-pustaka seperti NumPy untuk perhitungan numerik dan SymPy untuk perhitungan simbolik, penelitian ini bertujuan untuk memberikan wawasan mendalam tentang efisiensi dan akurasi metode perhitungan turunan fungsi menggunakan pendekatan ini.

Selain itu, penelitian ini juga akan membandingkan hasil perhitungan turunan fungsi dengan metode tradisional dan mengevaluasi kinerja keseluruhan penerapan Python dalam konteks ini. Hasil dari penelitian ini diharapkan dapat memberikan kontribusi pada pemahaman lebih lanjut tentang potensi dan batasan penggunaan Python dalam matematika komputasional, khususnya dalam perhitungan turunan fungsi.

Dengan memahami peran Python dalam perhitungan turunan fungsi, penelitian ini dapat memberikan panduan praktis bagi pengembang perangkat lunak, peneliti, dan mahasiswa yang tertarik dalam penggunaan bahasa pemrograman ini untuk menangani permasalahan matematika yang melibatkan turunan fungsi. Selain itu, penelitian ini diharapkan dapat memberikan sumbangan pada pengembangan lebih lanjut dalam bidang matematika komputasional.

## **2. METODE PENELITIAN**

Dalam penelitian ini, kami menggunakan metode studi literatur dan analisis untuk mendalami pemahaman perhitungan turunan fungsi dengan menggunakan bahasa pemrograman Python. Tahap awal melibatkan eksplorasi literatur terkait, di mana saya membaca dan mengumpulkan informasi tentang konsep dasar perhitungan turunan, algoritma-algoritma numerik, serta pendekatan simbolik yang diterapkan dalam matematika komputasional. Selanjutnya, kami melakukan analisis kritis terhadap sumber-sumber tersebut untuk mengidentifikasi kelebihan, kelemahan, dan tren terkini dalam penggunaan Python untuk perhitungan turunan fungsi. Melalui kombinasi studi literatur dan analisis, kami berupaya memahami kontribusi unik Python dalam konteks perhitungan matematika, dengan harapan mampu memberikan wawasan baru dan kontribusi positif pada pengembangan metode perhitungan turunan.

## **3. ANALISA DAN PEMBAHASAN**

### **3.1 Keunggulan Python dalam Perhitungan Turunan**

Keunggulan Python dalam perhitungan turunan menjadi sorotan utama dalam konteks pengembangan perangkat lunak matematika. Python membedakan dirinya dengan sintaks yang bersifat ekspresif dan mudah dibaca. VanderPlas (2016) menekankan bahwa keterbacaan kode Python memfasilitasi pengembangan algoritma perhitungan turunan dengan jelas dan efisien, meningkatkan kolaborasi antara pemrogram dan ahli matematika. Keunggulan ini mempercepat proses pengembangan solusi matematika, terutama dalam konteks aplikasi yang melibatkan analisis numerik.

Python juga terkenal dengan dukungan komunitas yang luas, yang menciptakan ekosistem dinamis untuk pengembangan matematika komputasional. McKinney (2017) mencatat bahwa keunggulan Python terletak pada kemampuannya untuk menggabungkan kekuatan pemrograman dengan analisis data. Dengan perkembangan pustaka dan modul yang terus bertambah, Python menjadi platform yang sangat fleksibel dan relevan dalam berbagai konteks penghitungan matematika.

Pentingnya Python dalam perhitungan turunan juga terletak pada penggunaan pustaka khusus seperti NumPy dan SymPy. Harris et al. (2020) menjelaskan bahwa NumPy memberikan dukungan efisien untuk perhitungan numerik dengan vektorisasi, yang mengoptimalkan kinerja perhitungan numerik pada array data. SymPy, menurut Meurer et al. (2017), melengkapi Python dengan alat perhitungan simbolik, memungkinkan representasi dan manipulasi ekspresi matematika secara simbolik. Integrasi antara NumPy dan SymPy membuat Python menjadi platform yang handal dan serbaguna dalam menangani perhitungan turunan, baik numerik maupun simbolik.

Dalam konteks pengembangan perangkat lunak, keterbacaan dan kemudahan sintaks Python memberikan keunggulan tambahan. McKinney (2017) menekankan bahwa Python dapat dengan mudah diintegrasikan dengan analisis data, visualisasi, dan pemodelan matematika, menjadikannya pilihan yang optimal dalam proyek-proyek yang melibatkan perhitungan turunan. Keunggulan ini diperkuat oleh fakta bahwa Python dapat diakses oleh berbagai kalangan, dari pemula hingga ahli, membuka pintu untuk kolaborasi yang lebih luas dalam pengembangan solusi matematika.

Dengan penggunaan yang meluas di berbagai industri dan disiplin ilmu, Python telah menjadi bahasa pemrograman yang sangat relevan dalam pengembangan perangkat lunak matematika. VanderPlas (2016) mencatat bahwa Python memiliki ekosistem yang dinamis dan terus berkembang, dengan dukungan komunitas yang aktif. Keunggulan Python dalam perhitungan turunan menciptakan fondasi yang kokoh untuk inovasi dan perkembangan lebih lanjut dalam bidang matematika komputasional.

### 3.2 Keunggulan Python dalam Perhitungan Turunan

Pustaka Python seperti NumPy dan SymPy memainkan peran penting dalam memperluas kemampuan perhitungan matematika komputasional. NumPy, yang merupakan pustaka dasar dalam ekosistem Python, memberikan dukungan efisien untuk operasi numerik dengan vektorisasi. Harris et al. (2020) menyoroti bahwa NumPy memungkinkan manipulasi array numerik dengan kecepatan tinggi, memberikan keunggulan dalam perhitungan numerik kompleks, seperti yang sering ditemui dalam analisis data, ilmu komputer, dan pengolahan sinyal.

SymPy, di sisi lain, membuka pintu bagi perhitungan simbolik yang diperlukan dalam analisis matematika tingkat tinggi. Pustaka ini, seperti yang dijelaskan oleh Meurer et al. (2017), memungkinkan representasi dan manipulasi ekspresi matematika dalam bentuk simbolik. Kelebihan SymPy menjadi krusial dalam konteks pemodelan matematika yang kompleks, di mana pemahaman aspek simbolik sangat diperlukan.

Kombinasi NumPy dan SymPy menjadikan Python pilihan yang kuat untuk berbagai aplikasi matematika komputasional. Keunggulan integrasi antara keduanya, seperti yang dijelaskan oleh Harris et al. (2020) dan Meurer et al. (2017), memungkinkan pengguna untuk memanfaatkan keuntungan dari perhitungan numerik dan simbolik secara bersamaan. Fleksibilitas Python dalam memanipulasi data numerik dan simbolik dengan mudah memberikan keunggulan tambahan dalam pengembangan dan penelitian ilmiah.

Penggunaan pustaka Python ini tidak hanya memperkaya kemampuan bahasa ini dalam perhitungan matematika, tetapi juga meningkatkan efisiensi dalam penanganan perhitungan kompleks. Pustaka NumPy, dengan kemampuan vektorisasinya, mempercepat proses perhitungan numerik pada data dalam skala besar. Sementara itu, SymPy memberikan cara yang efektif untuk melakukan perhitungan simbolik, memanfaatkan representasi matematika dalam bentuk simbolik.

### 3.3 Demonstrasi Perhitungan Turunan dengan Python

Perhitungan turunan dengan Python dapat diilustrasikan melalui demonstrasi praktis menggunakan contoh kode. McKinney (2017) menyoroti kegunaan Python dalam analisis data, dan kami dapat memanfaatkan pustaka seperti NumPy dan SymPy untuk perhitungan turunan. Sebagai contoh, pertimbangkan perhitungan turunan numerik menggunakan NumPy pada fungsi sederhana  $f(x) = x^2$ . Dengan menggunakan kode yang sesuai, kita dapat mendekati turunan fungsi ini secara numerik, menggambarkan bagaimana Python dapat digunakan untuk perhitungan matematika dasar.

```
import numpy as np
import matplotlib.pyplot as plt

# Fungsi yang akan dihitung turunannya
def f(x):
    return x**2

# Menentukan nilai x
x = np.linspace(-5, 5, 100)

# Menghitung turunan numerik menggunakan NumPy
f_prime_numeric = np.gradient(f(x), x)

# Menampilkan hasil
plt.plot(x, f(x), label='f(x)')
plt.plot(x, f_prime_numeric, label="f'(x) - Numerik")
plt.legend()
plt.title('Perhitungan Turunan Numerik dengan NumPy')
plt.show()
```

**Gambar 1.** Code Python NumPy

Selain perhitungan turunan numerik, Python juga mendukung perhitungan turunan simbolik melalui pustaka SymPy. SymPy memungkinkan representasi dan manipulasi ekspresi matematika secara simbolik. Sebagai contoh, kita dapat menggunakan SymPy untuk menghitung turunan simbolik dari fungsi yang sama,  $f(x) = x^2$ . Dengan memberikan contoh kode yang sesuai, kita dapat memperlihatkan bagaimana Python, melalui SymPy, dapat menghasilkan turunan fungsi secara simbolik.

```
from sympy import symbols, diff

# Variabel simbolik
x_sym = symbols('x')

# Fungsi simbolik
f_sym = x_sym**2

# Menghitung turunan simbolik menggunakan SymPy
f_prime_symbolic = diff(f_sym, x_sym)

# Menampilkan hasil
print(f'Turunan simbolik dari f(x) = x^2 adalah: {f_prime_symbolic}')
```

**Gambar 2.** Code Python NumPy

Dalam demonstrasi ini, kita dapat memilih fungsi matematika tertentu, misalnya, fungsi kuadrat atau trigonometri, dan menggunakan NumPy untuk menghitung turunannya pada beberapa titik. Visualisasi hasil perhitungan turunan ini dapat dicapai dengan menggunakan pustaka visualisasi seperti Matplotlib, sebagaimana dijelaskan oleh VanderPlas (2016), memberikan pemahaman visual yang jelas tentang konsep turunan pada fungsi tersebut.

Dengan menerapkan demonstrasi perhitungan turunan menggunakan Python, kita dapat menyajikan konsep perhitungan matematika dengan cara yang mudah dipahami oleh berbagai kalangan, mulai dari mahasiswa hingga peneliti dan praktisi di berbagai bidang ilmu. Kemampuan Python dalam menggabungkan kekuatan perhitungan numerik dan simbolik menjadikannya pilihan yang unggul dalam pemodelan dan analisis matematika.

### 3.3 Demonstrasi Perhitungan Turunan dengan Python

Perbandingan antara penggunaan Python, khususnya dengan pustaka NumPy dan SymPy, dengan metode tradisional atau alat lain dalam perhitungan turunan fungsi memberikan wawasan mendalam tentang keunggulan dan kelemahan masing-masing pendekatan. Dalam dunia perhitungan matematika, terdapat berbagai metode tradisional seperti perhitungan turunan dengan tangan atau menggunakan algoritma numerik konvensional. Dalam hal ini, Python menonjol dengan kemudahan sintaks, fleksibilitas, dan kecepatan eksekusi, terutama ketika memanfaatkan keunggulan pustaka khusus seperti NumPy untuk perhitungan numerik dan SymPy untuk perhitungan simbolik.

Penggunaan Python menciptakan konsistensi dalam alur kerja, memungkinkan peneliti dan praktisi untuk dengan mudah menggabungkan perhitungan numerik dan simbolik dalam satu platform. Sebaliknya, metode tradisional mungkin memerlukan transisi antar alat atau bahasa pemrograman, yang dapat menyulitkan integrasi dan memperlambat proses analisis matematika. Python juga memungkinkan otomatisasi perhitungan dan penanganan data yang besar, menjadikannya lebih skalabel dibandingkan dengan metode tradisional yang mungkin kurang efisien dalam menangani kompleksitas perhitungan matematika.

Keunggulan Python terlihat dalam dukungan komunitas yang besar dan aktif. Komunitas Python yang luas telah mengembangkan berbagai pustaka dan modul yang memperkaya kemampuan bahasa ini dalam perhitungan matematika komputasional, termasuk perhitungan turunan fungsi. Hal ini menciptakan ekosistem yang dinamis dan berkembang, memberikan para pengguna Python akses kepada alat-alat terbaru dan solusi terbaik dalam bidang perhitungan matematika.

Penggunaan pustaka Python seperti NumPy dan SymPy membawa sejumlah keunggulan dalam perhitungan turunan. NumPy memberikan dukungan untuk perhitungan numerik dengan vektorisasi, yang mengoptimalkan kinerja perhitungan numerik pada array data. Di sisi lain, SymPy menyediakan alat untuk perhitungan simbolik, memungkinkan representasi dan manipulasi ekspresi matematika secara simbolik. Integrasi sinergis antara NumPy dan SymPy memungkinkan Python menangani perhitungan turunan numerik dan simbolik dengan mudah dalam satu platform, menciptakan alur kerja yang konsisten dan efisien.

Perbandingan ini juga membuka diskusi tentang kecepatan eksekusi dan efisiensi penggunaan sumber daya komputasi. Python, dengan pustaka khususnya, dapat memberikan kinerja yang memadai untuk sebagian besar aplikasi perhitungan turunan. Namun, tergantung pada kebutuhan dan kompleksitas perhitungan, metode tradisional atau alat lain mungkin lebih cocok untuk situasi tertentu.

## 4. KESIMPULAN

Secara keseluruhan, keunggulan Python dalam perhitungan turunan fungsi terletak pada sintaks yang mudah dibaca dan ekspresif, memberikan keterbacaan dan pemahaman yang lebih baik terhadap algoritma perhitungan. Dukungan pustaka khusus seperti NumPy dan SymPy meningkatkan daya tangkap Python dalam menangani perhitungan numerik dan simbolik secara efisien. NumPy memberikan kinerja numerik yang optimal melalui vektorisasi, sementara SymPy memfasilitasi perhitungan simbolik dengan representasi dan manipulasi ekspresi matematika secara simbolik.

Integrasi sinergis antara NumPy dan SymPy menciptakan lingkungan kerja yang konsisten dan efisien, memungkinkan penanganan perhitungan turunan numerik dan simbolik dalam satu platform. Keunggulan Python tidak hanya terletak pada kemampuan teknisnya, tetapi juga didukung oleh komunitas yang besar dan aktif, yang terus mengembangkan berbagai alat dan solusi terbaik dalam perhitungan matematika komputasional.

Kesimpulannya, Python bukan hanya bahasa pemrograman yang populer dan mudah dipelajari, tetapi juga menjadi pilihan yang kuat dan serbaguna dalam menangani tugas matematika yang melibatkan perhitungan turunan fungsi. Dengan kombinasi kejelasan sintaks, dukungan pustaka khusus, dan dukungan komunitas, Python membuka pintu untuk efisiensi dan inovasi dalam dunia perhitungan matematika komputasional.

## REFERENCES

- VanderPlas, J. (2016). Python Data Science Handbook. O'Reilly Media.
- McKinney, W. (2017). Python for Data Analysis. O'Reilly Media.
- Meurer, A., Smith, C. P., Paprocki, M., Čertík, O., Kirpichev, S. B., Rocklin, M., ... & Klein, A. (2017). SymPy: symbolic computing in Python. PeerJ Computer Science, 3, e103.
- Van Rossum, G., & Drake, F. L. (2009). Python 3 Reference Manual. CreateSpace.
- Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (2007). Numerical Recipes 3rd Edition: The Art of Scientific Computing. Cambridge University Press.
- Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., ... & Oliphant, T. E. (2020). Array programming with NumPy. Nature, 585(7825), 357-362. DOI: 10.1038/s41586-020-2649-2.