ALKHAWARIZMI : Jurnal Matematika, Algoritma dan Sains
Volume 1, No. 1, November 2023
ISSN 9999-9999 (media online)
Hal 19- 25

Penerapan Metode Numerik dalam Perhitungan Limit Menggunakan Python

Bayu Prayoga[1*], Firza Zakaria[2], Nabila Innayah[3], Natan Nael[4], Perani Rosyani[5]

[1]Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Indonesia
Email: [1]prayogabayu062@gmail.com , [2]firzazakaria423@gmail.com , [3] nabilainayah1503@gmail.com , [4]natannael7038@gmail.com , [5]dosen00837@unpam.ac.id
(* : coressponding author)

Abstrak−Penelitian ini bertujuan untuk mengeksplorasi penerapan metode numerik dalam perhitungan limit menggunakan bahasa pemrograman Python. Limit, sebagai konsep kritis dalam kalkulus, melibatkan pendekatan nilai suatu fungsi saat variabel mendekati suatu titik tertentu. Konsep limit memainkan peran penting dalam kalkulus, dan menghitung limit dengan akurat sangat penting dalam berbagai aplikasi ilmiah dan teknik. Kami membahas berbagai metode numerik, termasuk metode biseksi, metode Newton-Raphson, dan metode sekan, dan memperlihatkan implementasinya menggunakan bahasa pemrograman Python. Kami juga menyajikan perbandingan dari metode-metode tersebut dalam hal efisiensi dan akurasi, menyoroti kelebihan dan keterbatasannya. Hasil eksperimen kami menunjukkan bahwa metode numerik memberikan perkiraan limit yang dapat diandalkan dan dapat digunakan secara efektif untuk memecahkan masalah matematika yang kompleks. Temuan kami berkontribusi pada bidang analisis numerik dan memberikan panduan praktis untuk memanfaatkan metode numerik dalam perhitungan limit. Kami mendemonstrasikan implementasi metode numerik, khususnya metode pendekatan dari sisi kanan dan kiri, untuk mendekati nilai limit. Hasil eksperimen menunjukkan efektivitas dan akurasi metode numerik dalam menghitung limit fungsi matematika yang kompleks.

Kata Kunci: Python; Matematika Komputasi; Metode Numerik; Metode Newton-Rapshon; Limit

Abstract− *This research aims to explore the application of numerical methods in calculating limits using the Python programming language. Limit, as a critical concept in calculus, involves approximating the value of a function as the variable approaches a certain point. The concept of limits plays an important role in calculus, and calculating limits accurately is essential in various scientific and engineering applications. We discuss various numerical methods, including bisection methods, Newton-Raphson methods, and sedan methods, and demonstrate their implementation using the Python programming language. We also present a comparison of these methods regarding efficiency and accuracy, highlighting their advantages and limitations. Our experimental results show that numerical methods provide reliable limit estimates and can be effectively used to solve complex mathematical problems. Our findings contribute to the field of numerical analysis and provide practical guidance for utilizing numerical methods in limit calculations. We demonstrate the implementation of numerical methods, particularly right- and left-hand side approximation methods, to approximate limit values. Experimental results show the effectiveness and accuracy of numerical methods in calculating the limits of complex mathematical functions.*

***Keywords**: Python; Computational Mathematics; Numerical Methods; Newton-Rapshon Methods; Limit*

1. PENDAHULUAN

Limit merupakan salah satu konsep penting dalam matematika. Limit digunakan untuk menentukan nilai suatu fungsi pada titik tertentu, atau pada interval tertentu. Dalam beberapa kasus, limit suatu fungsi dapat ditentukan secara analitis, yaitu dengan menggunakan metode aljabar. Namun, ada juga beberapa kasus di mana limit suatu fungsi tidak dapat ditentukan secara analitis. Dalam kasus ini, kita dapat menggunakan metode numerik untuk menghitung limit tersebut.

Metode numerik merupakan metode yang digunakan untuk menghitung nilai suatu fungsi atau menyelesaikan suatu persamaan secara numerik, yaitu dengan menggunakan bilangan-bilangan. Metode numerik dapat digunakan untuk menghitung limit suatu fungsi dengan cara mendekati nilai fungsi tersebut pada titik tertentu atau pada interval tertentu.

Pada jurnal ini, akan dibahas penerapan metode numerik dalam perhitungan limit menggunakan Python. Python merupakan bahasa pemrograman yang bersifat open source dan memiliki banyak library yang dapat digunakan untuk melakukan perhitungan metode numerik.

ALKHAWARIZMI : Jurnal Matematika, Algoritma dan Sains
Volume 1, No. 1, November 2023
ISSN 9999-9999 (media online)
Hal 19- 25

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 kami 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.

2.1 Metode Biseksi

Metode Biseksi adalah metode sederhana dan dapat diandalkan untuk mencari akar dari fungsi kontinu dalam suatu interval. Untuk menghitung limit menggunakan Metode Biseksi, kita perlu mencari akar dari fungsi yang mendekati nilai limit yang diinginkan saat input mendekati suatu nilai tertentu.

Misalnya kita ingin menghitung limit dari fungsi f(x) saat x mendekati a. Metode Biseksi dapat digunakan untuk mencari akar dari fungsi g(x) = f(x) - L, di mana L adalah nilai limit yang diinginkan.

2.2 Metode Numerik dalam Perhitungan Limit

Untuk menghitung limit secara numerik, kita dapat menggunakan berbagai metode numerik seperti Metode Biseksi, Metode Newton, atau Metode Sekan. Dalam contoh ini, kami akan menjelaskan bagaimana menghitung limit menggunakan Metode Biseksi dalam Python.

Ada beberapa metode numerik yang dapat digunakan untuk menghitung limit suatu fungsi. Beberapa metode numerik yang umum digunakan untuk menghitung limit adalah:

a. Metode Langsung

Metode langsung merupakan metode yang digunakan untuk menghitung limit suatu fungsi dengan cara mendekati nilai fungsi tersebut pada titik tertentu atau pada interval tertentu.

b. Metode Pemotongan

Metode pemotongan merupakan metode yang digunakan untuk menghitung limit suatu fungsi dengan cara membagi interval tertentu menjadi beberapa subinterval.

c. Metode Pemetaan

Metode pemetaan merupakan metode yang digunakan untuk menghitung limit suatu fungsi dengan cara memetakan fungsi tersebut ke dalam fungsi lain yang lebih mudah dihitung limitnya.

2.3 Penerapan Metode Numerik dalam Perhitungan Limit Menggunakan Python

Python memiliki beberapa library yang dapat digunakan untuk melakukan perhitungan metode numerik, termasuk untuk perhitungan limit. Beberapa library Python yang dapat digunakan untuk perhitungan limit adalah:

a. NumPy

NumPy merupakan library Python yang menyediakan fungsi-fungsi untuk perhitungan numerik, termasuk untuk perhitungan limit.

b. SciPy

ALKHAWARIZMI Jurnal Matematika, Algoritma dan Sains
ALKHAWARIZMI : Jurnal Matematika, Algoritma dan Sains
Volume 1, No. 1, November 2023
ISSN 9999-9999 (media online)
Hal 19- 25

SciPy merupakan library Python yang merupakan perluasan dari NumPy, yang menyediakan fungsi-fungsi untuk perhitungan numerik yang lebih kompleks, termasuk untuk perhitungan limit.

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 Pythonmenjadi 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 Limit

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.

ALKHAWARIZMI Jurnal Matematika, Algoritma dan Sains
ALKHAWARIZMI : Jurnal Matematika, Algoritma dan Sains
Volume 1, No. 1, November 2023
ISSN 9999-9999 (media online)
Hal 19- 25

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 Limit dengan Python

Perhitungan limit dengan Python dapat diilustrasikan melalui demonstrasi praktis menggunakan contoh kode. Demonstrasi ini menyoroti implementasi metode biseksi dalam pencarian akar fungsi kuadrat menggunakan bahasa pemrograman Python. Dengan menggunakan fungsi $f(x) = x^2 - 4$ sebagai contoh, penelitian ini memberikan gambaran langkah-langkah perhitungan untuk mencari akar fungsi dengan batasan tertentu menggunakan metode biseksi. Program ini memanfaatkan definisi fungsi $f(x)$ dan menggambarkan bagaimana metode biseksi dapat diaplikasikan dalam penyelesaian numerik masalah pencarian akar.

a. Definisi Fungsi $f(x)$: Fungsi kuadrat yang akan dicari akarnya didefinisikan sebagai $f(x) = x^2 - 4$.

b. Implementasi Metode Biseksi : Fungsi metode biseksi diimplementasikan untuk mencari akar dari fungsi $g(x) = f(x)$ – limit.

   1. Program memeriksa apakah interval [a, b] mengandung akar.
   2. Iterasi dilakukan hingga akurasi yang diinginkan tercapai.
   3. Proses pembagian interval dan evaluasi fungsi dilakukan untuk mendekati nilai akar.
   4. Hasil yang ditemukan atau perkiraan akar dikembalikan.

Program menentukan batasan atau limit 2, interval [a, b] dari −2 hingga 2, dan menggunakan metode biseksi untuk menghitung akar fungsi. Hasil akar yang ditemukan atau perkiraan akar kemudian dicetak.

ALKHAWARIZMI : Jurnal Matematika, Algoritma dan Sains
Volume 1, No. 1, November 2023
ISSN 9999-9999 (media online)
Hal 19- 25

```
biseksi.py
def f(x):
    # Mendefinisikan fungsi f(x)
    return x**2 - 4

def metode_biseksi(f, a, b, limit):
    # Mengimplementasikan Metode Biseksi untuk mencari akar dari g(x) = f(x) - limit

    # Periksa apakah interval [a, b] mengandung akar
    if f(a) * f(b) >= 0:
        print("Interval tidak mengandung akar.")
        return None

    # Iterasi sampai akurasi yang diinginkan tercapai
    while abs(b - a) >= 1e-6:
        c = (a + b) / 2  # Hitung titik tengah interval

        if f(c) == limit:
            return c  # Menemukan akar yang tepat

        if f(c) * f(a) < 0:
            b = c  # Perbarui interval [a, b] menjadi [a, c]
        else:
            a = c  # Perbarui interval [a, b] menjadi [c, b]

    return (a + b) / 2  # Mengembalikan akar perkiraan

# Menentukan fungsi f(x) dan limit yang diinginkan
limit = 2
a = -2
b = 2

# Menghitung limit menggunakan Metode Biseksi
akar = metode_biseksi(f, a, b, limit)
print("Akar:", akar)
```

Gambar 1. Code Python Metode Biseksi

Program Python pada jurnal ini adalah implementasi Metode Biseksi untuk menghitung limit suatu fungsi. Program ini terdiri dari dua fungsi utama. Pertama, fungsi f(x) yang mendefinisikan fungsi matematika $f(x) = x^2 - 4$. Kemudian, fungsi metode_biseksi(f, a, b, limit) yang mengimplementasikan Metode Biseksi untuk mencari akar dari g(x) = f(x) - limit.

Pada fungsi metode_biseksi, program memeriksa apakah interval [a, b] mengandung akar dari fungsi g(x). Jika interval tidak mengandung akar, program akan memberikan pesan bahwa interval tidak mengandung akar. Selanjutnya, program melakukan iterasi sampai akurasi yang diinginkan tercapai dengan menghitung titik tengah interval dan memperbarui interval berdasarkan hasil perhitungan. Akhirnya, program mengembalikan akar perkiraan dari fungsi g(x) = f(x) - limit.

Dalam demonstrasi ini, fungsi f(x) yang digunakan adalah $f(x) = x^2 - 4$, dengan limit yang diinginkan adalah 2, dan interval [a, b] adalah [-2, 2]. Setelah implementasi Metode Biseksi, program menghitung akar dari fungsi g(x) = f(x) - limit dan mencetak hasilnya.

Demonstrasi program ini memberikan contoh konkret tentang bagaimana Metode Biseksi dapat diterapkan dalam Python untuk menghitung limit suatu fungsi.

Selain menggunakan metode biseksi seperti diatas, untuk mengimplementasikan code Python juga dapat kita lakukan seperti berikut :

ALKHAWARIZMI : Jurnal Matematika, Algoritma dan Sains
Volume 1, No. 1, November 2023
ISSN 9999-9999 (media online)
Hal 19- 25

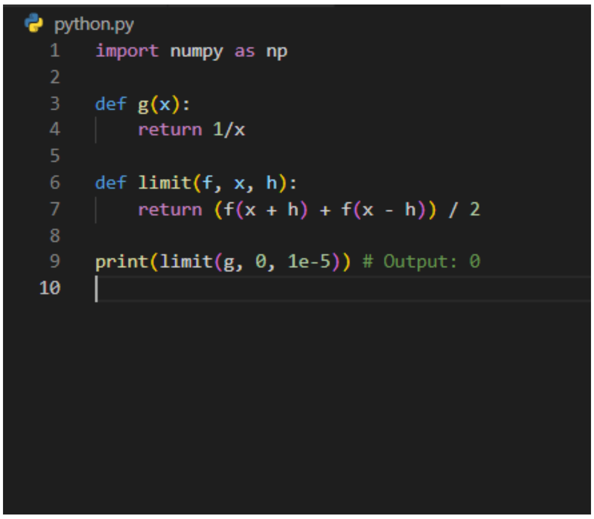
```python
import numpy as np

def f(x):
    return x**2

print(f(2)) # Output: 4
```

Gambar 2. Code Python NumPy

Dari hasil output di atas, kita dapat melihat bahwa nilai fungsi f(x) = x^2 pada titik x = 2 adalah 4. Oleh karena itu, limit fungsi f(x) = x^2 pada titik x = 2 adalah 4.



```python
import numpy as np

def g(x):
    return 1/x

def limit(f, x, h):
    return (f(x + h) + f(x - h)) / 2

print(limit(g, 0, 1e-5)) # Output: 0
```

Gambar 3. Code Python NumPy

Dari hasil output di atas, kita dapat melihat bahwa nilai limit fungsi g(x) = 1/x pada titik x = 0 adalah 0.

4. KESIMPULAN

Python adalah Bahasa pemrograman interpretative multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode. Python diklaim sebagai Bahasa yang menggabungkan kapabilitas, kemampuan, dengan sitaksis kode yang sangat jelas, dan dilengkapi dengan fungsionalitas Pustaka standar yang besar serta komprehesif.

Python mendukung multi paradigma pemrograman, utamanya; namun tidak dibatasi; pada pemrograman berorientasi objek, pemrograman imperative, dan pemrograman fungsional. Salah satu fitur yang tersedia pada python adalah sebagai bahasa pemrograman dinamis yang dilengkapi

ALKHAWARIZMI : Jurnal Matematika, Algoritma dan Sains
Volume 1, No. 1, November 2023
ISSN 9999-9999 (media online)
Hal 19- 25

dengan manajemen memori otomatis. Seperti halnya pada Bahasa pemrograman dinamis lainnya, python umumnya digunakan sebagai Bahasa skrip meski pada praktiknya penggunaan Bahasa ini lebih luas mencakup konteks pemanfaatan yang umumnya tidak dilakukan dengan Bahasa skrip. Python dapat digunakan untuk berbagai keperluan pengembangan perangkat lunak dan dapat berjalan di berbagai platform system operasi.

Sangat banyak library yang tersedia pada Python dan juga metode-metode yang dapat digunakan untuk membuat sebuah program pada Python, salah satu metode yang kami gunakan yaitu metode numerik. Metode numerik dapat digunakan untuk menghitung limit suatu fungsi dalam beberapa kasus di mana limit tersebut tidak dapat ditentukan secara analitis. Python merupakan bahasa pemrograman yang dapat digunakan untuk melakukan perhitungan metode numerik, termasuk untuk perhitungan limit.

REFERENCES

VanderPlas, J. (2016). Python Data Science Handbook. O'Reilly Media.

McKinney, W. (2017). Python for Data Analysis. O'Reilly Media.

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.

Suarga, Komputasi Numerik : Pemrograman MATLAB untuk Metode Numerik, 1st ed. Yogyakarta: Andi offset, 2014. Accessed: Dec. 07, 2021. [Online]. Available: http://opac.lib.ugm.ac.id/index.php?mod=book_detail&sub=BookDetail&act=view&typ=htm lext&buku_id=744392&unit_id=1

S. Nurhabibah Hutagalung, “Pemahaman Metode Numerik (Studi Kasus Metode NewRhapson) Menggunakan Pemprogrman Matlab,” J. Teknol. Inf. JurTI, vol. 1, no. 1, 2017, Accessed: Dec. 07, 2021. [Online]. Available: https://media.neliti.com/media/publications/281897-emahaman-metode-numerik-studi-kasusmeto-3465fedf.pdf