Computation Methods for Numerical Analysis Using SageMath

MAT251 PROJECT

Lecturer: Dr. D. A. Dikko

University of Ibadan, Ibadan, Oyo State, Nigeria

October, 2023

Group Members

1. 231020 Babalola Isa Gbolahan
2. 231007 Adenekan Damilola Omotoyosi
3. 236634 Owolabi Odunayo Joseph
4. 231224 Oladokun Shukroh Tolani
5. 231010 Agunbiade Olamide Roimot
6. 231235 Raji Mueez Olanrewaju
7. 232089 Mudasiru Samad Olamide
8. 231023 Chukwudi Emmanuel Emeka
9. 231024 Daramola Similoluwa
10. 231068 Shoyemi Opeyemi Samuel
11. 231008 Adeyemi Ridwan Oluwaseyi
12. 231191 Afolabi Nathanael Ademola
13. 231190 Adetoye Inumidun Adekoyejo

Introduction to SageMath

SageMath is a powerful open-source mathematics software system that integrates various mathematical tools, packages, and programming languages into a single, cohesive environment. It was developed to provide a unified platform for mathematical research, exploration, and teaching.

SageMath offers a wide array of features, including symbolic and numerical mathematics, data visualization, and support for various programming languages such as Python. Researchers, mathematicians, educators, and students use SageMath for a range of mathematical tasks, from algebra and calculus to number theory and cryptography.

One of SageMath's distinguishing characteristics is its open-source nature, allowing users to access and modify its source code freely. This fosters collaboration, customization, and the development of mathematical tools that cater to specific needs.

SageMath's versatility and the support of a robust community of users and developers make it a valuable resource in the world of mathematics and computational science. Whether you are a seasoned mathematician or a student learning mathematics, SageMath provides a comprehensive and flexible environment to explore and solve mathematical problems.

The Jupyter Notebook

Jupyter Notebook is a web-based computational environment for creating documents interacting with Python code. It is intended to present code in a nice way and is also capable of LaTeXformatting. Jupyter Notebook is part of the open-source project Jupyter[2], and is already included in Sage.

In order to start Jupyter, we type the following command into the Sage command line:

```
sage: !sage -n jupyter
```

A browser window pops up with the Jupyter web-interface.

SageMath as a Calculator

SageMath offers a wide range of mathematical functions and capabilities, making it a versatile tool for various mathematical calculations and problem-solving tasks. We can perform both basic arithmetic and complex mathematical operations using SageMath as a calculator.

Henceforth, all commands and calculations in Sage are presented as follows:

```
sage: 2+3
5
```

Here, the prefix sage indicates that we are using the Sage command line. If we are working on Jupyter instead, all we just type in the subsequent code and the kernel will do its job.

Notice that all common operations such as +, -, *, / and parentheses are carried out as usual. To operate with exponents, we use ^ or **, the following are some other basic operations, predefined functions amd constants.

General Arithmetics	
Binary Operations	`a+b, a-b, a*b, a/b`
Exponent	`a^b` or `a**b`
Square Root	`sqrt(a)`
n-th Root	`a^(1/n)`
Integer Arithmetics	
Floor-Division	`a // b`
Remainder	`a % b`
Floor-Division & Remainder	`divmod(a,b)`
Factorial $n!$	`factorial(n)`
Binomial Coefficient $\binom{n}{k}$	`binomial (n,k)`

(a) Basic operations in sage

Predefined Functions	
Exponential, Natural Logarithm	`exp, log`
Logarithm w.r.t. Base b	`log(a,b)`
Trig. Functions	`sin, cos, tan`
Inverse Trig. Functions	`arcsin, arccos, arctan`
Hyp. Trig. Functions	`sinh, cosh, tanh`
Inverse Hyp. Trig. Functions	`arcsinh, arccosh, arctanh`
Absolute Value / Modulus	`abs(a)`
Special Values / Constants	
Imaginary Unit i	`I` or `i`
Plus / Minus Infinity	$\pm$`Infinity` or $\pm$`oo`
π	`pi`
Euler's Constant	`e`
Euler-Mascheroni Constant γ	`euler_gamma`
Golden Ratio $\varphi = (1+\sqrt{5})/2$	`golden_ratio`

(b) Predefined functions and constants

Plotting Graphs in Sagemath

Plotting graphs in SageMath is a fundamental and powerful feature. We can create various types of graphs, including functions, parametric plots, polar plots, and more. We shall consider simple plot and 3D plot.

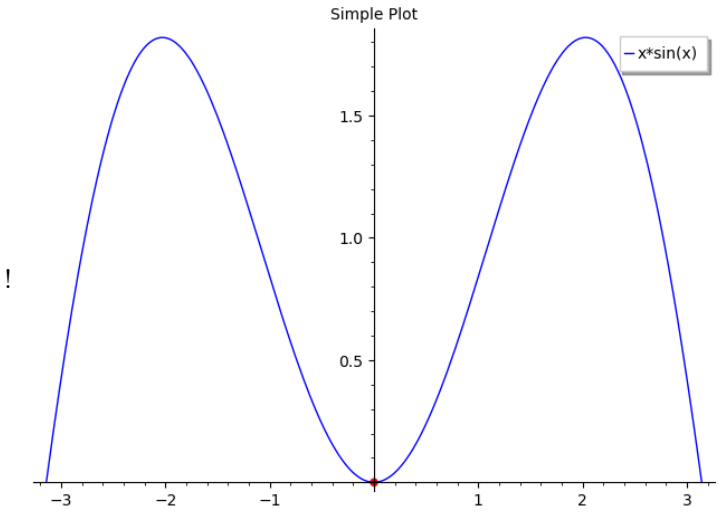
2D Plots

The code to plot 2D(or simple plot) is given in the diagram below:

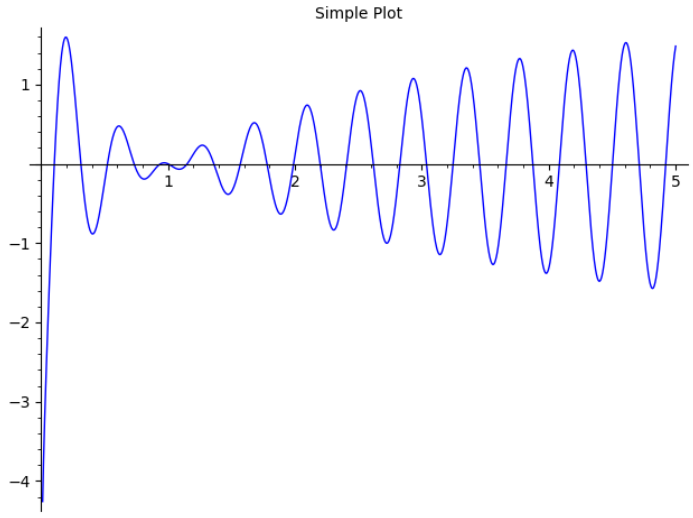
```
# Define a mathematical function or surface
f(x, y) = x*sin(y)


# Create a 3D plot of the function
plot3d(f, (x,x_0,x_1), (y,y_1,y_2))

```

We now draw the graph of the following functions using the above code

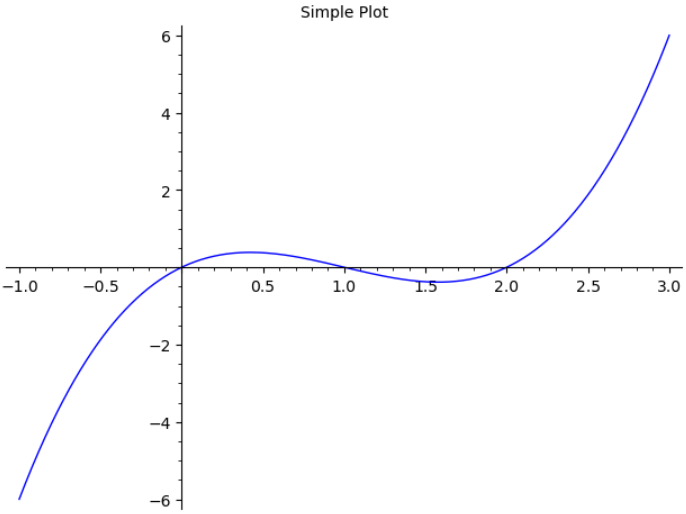
$$y = x\sin(x), \quad y = \ln(x)\cos(15x), \quad y = x^3 - 3x^2 + 2x, \quad y = x^2(x-1)^2, \quad y = -x^3 + 2x^2 + 5x - 30,$$



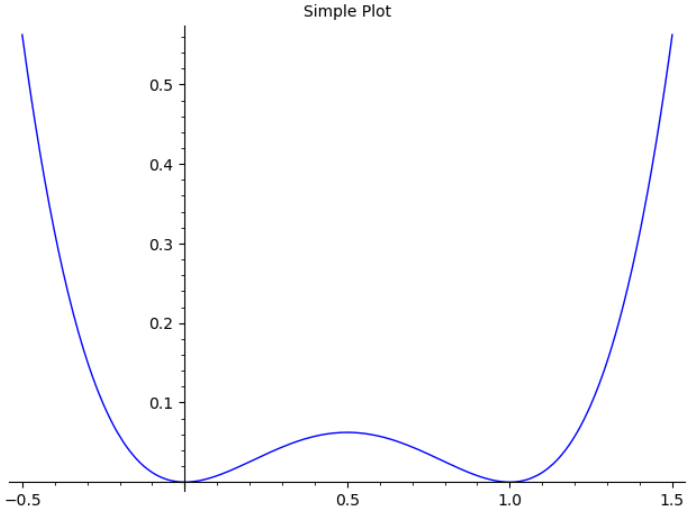
(a) Graph of $y = x\sin(x)$



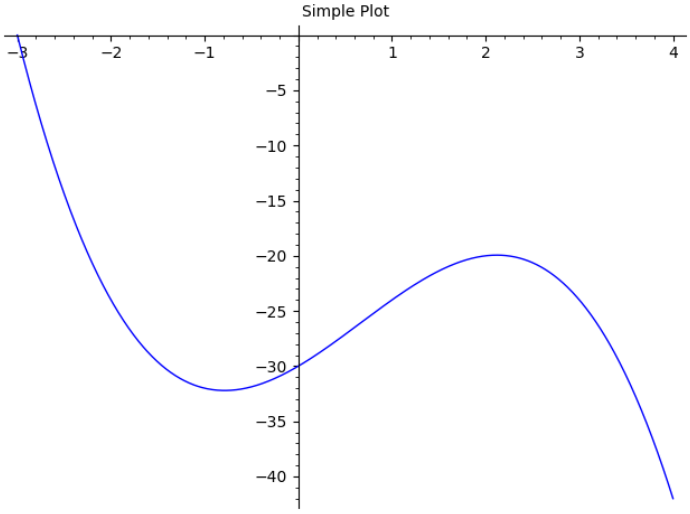
(b) Graph of $y = \ln(x)\cos(15x)$



(c) Graph of $y = x^3 - 3x^2 + 2x$



(d) Graph of $y = x^2\,(x-1)^2$



(e) Graph of $y = -x^3 + 2x^2 + 5x - 30$

Figure 2: Graphs of functions by SageMath

3D Graphs

SageMath is a powerful open-source mathematics software system that offers a wide range of features, including 3D plotting capabilities. 3D plotting allows you to visualize mathematical functions and surfaces in a three-dimensional space.

It's a valuable tool to gain insights into complex mathematical relationships and data.
In this example, we'll demonstrate how to create a 3D plot of a simple function using SageMath, where we can use this as a starting point to explore and visualize our own 3D mathematical expressions. The SageMath code to plot 3D graph is given by:

Figure 3: Sage code to plot in 3D