Algebra with SymPy

Abstract

Algebra is the unifying thread weaving through almost all of mathematics, seamlessly linking simple equations to complex concepts like matrix operations, polynomial functions, and vector spaces. This one-pager highlights algebraic properties with a SymPy demonstration.

Algebraic Properties

For real numbers a, b and c

Commutative:	$a + b = b + a$	$a \times b = b \times a$
Associative:	$(a + b) + c = a + (b + c)$	$(a \times b) \times c = a \times (b \times c)$
Distributive:	$(a + b) \times c = a \times c + b \times c$	
Inverse:	$a + (-a) = 0$	$a \times \frac{1}{a} = 1$
Negation:	$-(-a) = a$	$(-a) \times b = a \times (-b) = -(a \times b)$
	$(-a) \times (-b) = a \times b$	
Zero Product:	$a \times 0 = 0 \times a = 0$	
Cancellation:	If $a + x = a + y$, then $x = y$	

Python

SymPy is a robust open-source Python library for symbolic computation, offering computer algebra capabilities. It can be used as a standalone application, integrated as a library in other applications, or accessed online through SymPy Live or SymPy Gamma. To use SymPy, first install it using the Python package manager: **pip install sympy**

```
from sympy import symbols, Eq, solve

x, y = symbols("x y")              # Define symbols
equation = Eq(4 + 4 * y, 8)        # Define equation
solution = solve(equation, y)      # Solve equation for y

print(solution)

------------------------------------------------------------
>>> [1]
```

Author: Karl-Fredrik M. Hagman SymPy: https://docs.sympy.org/latest/index.html