

# Python and Julia for economic modeling: developments and trends

John Stachurski

October 2023

## Topics

- Trends in scientific computing
- Likely future directions
- Python and Julia as MATLAB replacements

# A (very) short history of scientific computing

General purpose scientific computing environments:

1. Fortran & C / C++
2. MATLAB & (Python + NumPy)
3. Julia & (Python + Numba)
4. Python + Google JAX

## Fortran & C — static types and AOT compilers

```
#include <stdio.h>

int main() {
    int x = 1 + 1;
    printf("1 + 1 = %d\n", x);
    return 0;
}
```

```
PROGRAM ONE_PLUS_ONE
INTEGER :: X = 1 + 1
PRINT *, '1 + 1 = ', X
END PROGRAM ONE PLUS ONE
```

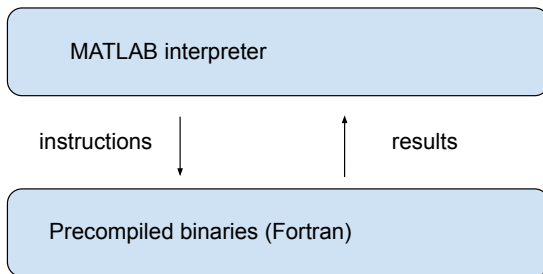
## Pros

- fast — on a single thread

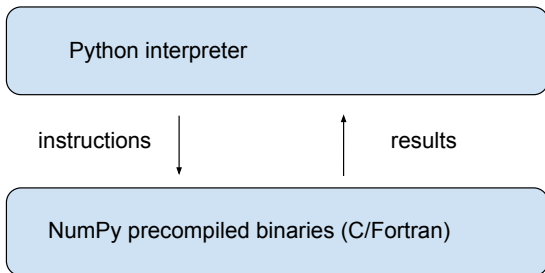
## Cons

- tedious to write
- lack of portability
- hard to debug
- hard to parallelize
- low interactivity

## Phase 2: MATLAB



## Phase 2A: Python + NumPy



## Phase 3: Julia — rise of the JIT compilers

---

```
function quad(x0,  $\alpha$ , n)
    x = x0
    for i in 1:(n-1)
        x =  $\alpha$  * x * (1 - x)
    end
    return x
end
```

```
quad(0.2, 4.0, 10_000_000)
```

---



## Phase 3 continued: Python + Numba copy Julia

---

```
from numba import jit

@jit
def quad(x0,  $\alpha$ , n):
    x = x0
    for i in range(n-1):
        x =  $\alpha$  * x * (1 - x)
    return x

quad(0.2, 4.0, 10_000_000)
```

---

## Phase 4: AI-driven scientific computing

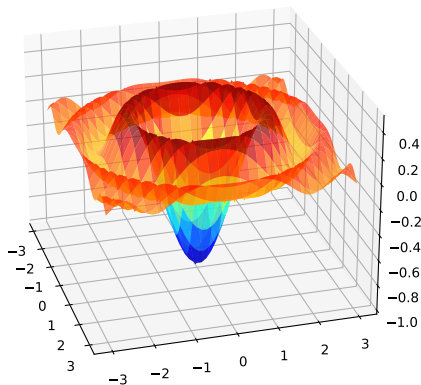
### Core elements

- JIT-compilers
- automatic differentiation
- parallelization (CPUs / GPUs / TPUs)

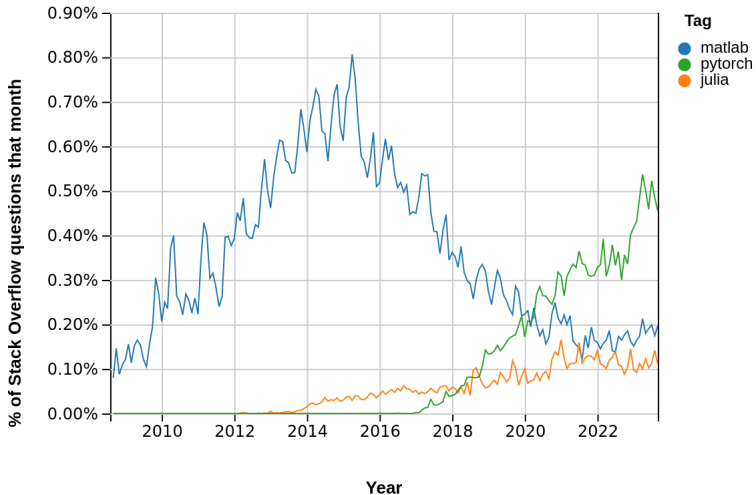
### Key players

- PyTorch
- Google JAX
- Mojo?

AI / machine learning: minimizing differentiable loss functions



# Stack Overflow Trends



## Stack Overflow 2023 developer survey (50 languages)

*Rust is the most admired language, more than 80% of developers that use it want to use it again next year.*

*Compare this to the least admired language: MATLAB. Less than 20% of developers who used this language want to use it again next year.*

— <https://survey.stackoverflow.co/2023/>

- Python: 65.5%
- Julia: 62.7%
- MATLAB: 18.3%

# Forecasts

Julia is valuable

- close to maths
- multiple dispatch, built for JIT, etc.
- Dynare Julia rewrite

But Python will continue increasing dominance

- easy to learn, massive ecosystem
- the language of AI
- backed by Google, Meta, Microsoft, Modular, etc.
- interoperable with Excel / STATA / etc.

# Forecasts

Julia is valuable

- close to maths
- multiple dispatch, built for JIT, etc.
- Dynare Julia rewrite

But Python will continue increasing dominance

- easy to learn, massive ecosystem
- the language of AI
- backed by Google, Meta, Microsoft, Modular, etc.
- interoperable with Excel / STATA / etc.