Correlations between programming languages and beliefs about programming

Pierce Edmiston
pedmiston@wisc.edu

My research questions

How is our ability to solve problems affected by the tools we use to solve them?
How do people solve the same problems using different programming languages?

This data blitz:

- Explore survey responses given by professional and academic programmers.
- We asked them what languages they knew and their beliefs about programming.

Who cares about programming languages?

Hundreds of different programming languages, but all are formally equivalent.

Computer scientists think a lot about the differences between languages.

Can we measure the *impact* of different programming languages on individual computer scientists?

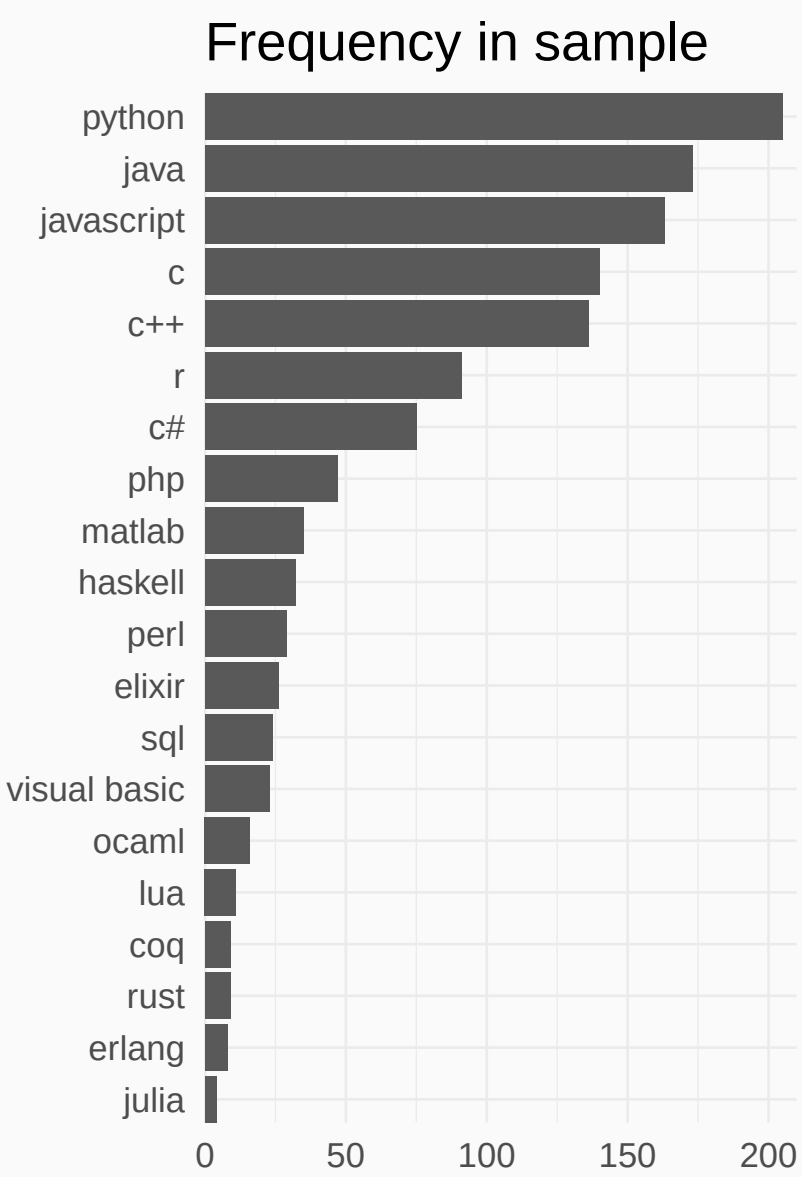
Can we measure differences between programmers?

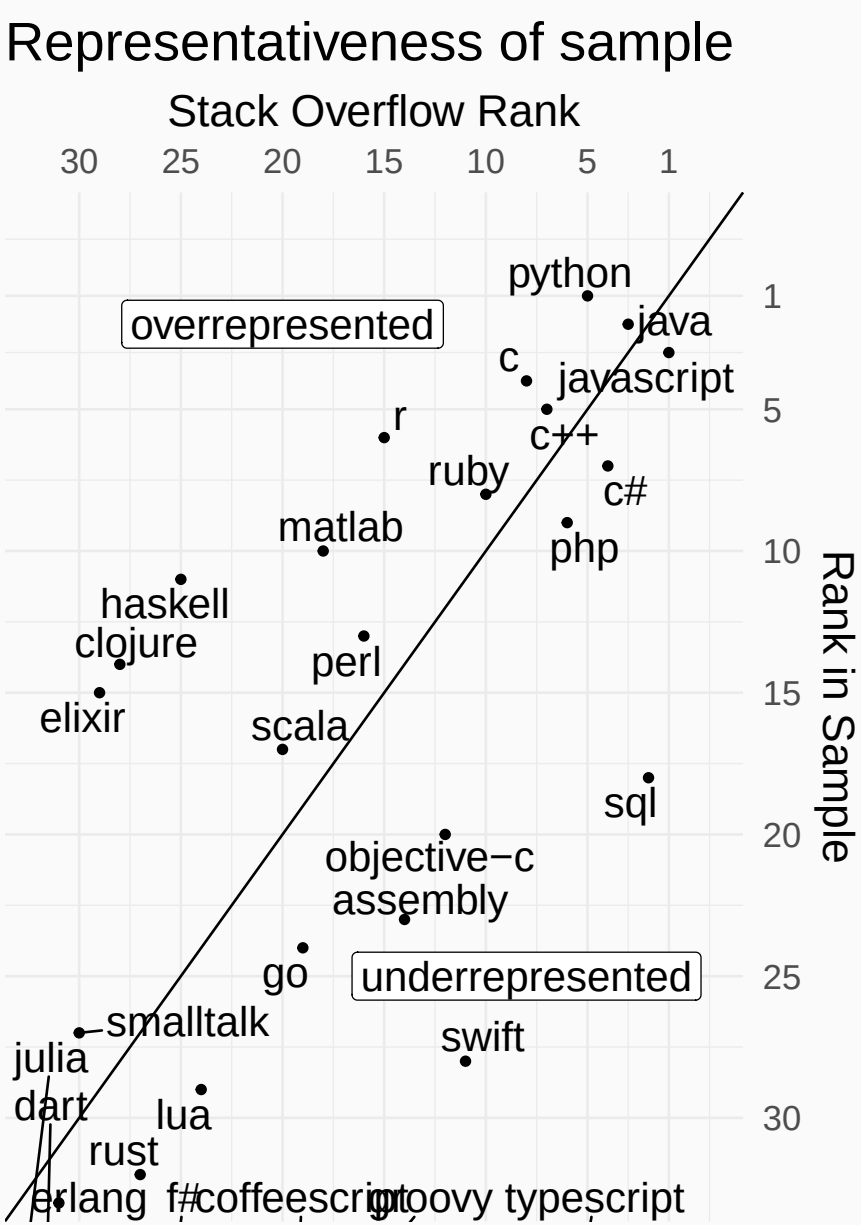
Of course! What should we predict and why does it matter?

H1 Functional languages are more transformative than other language paradigms.

H2 Languages with more paradigms are easier to master.

What languages did they know?

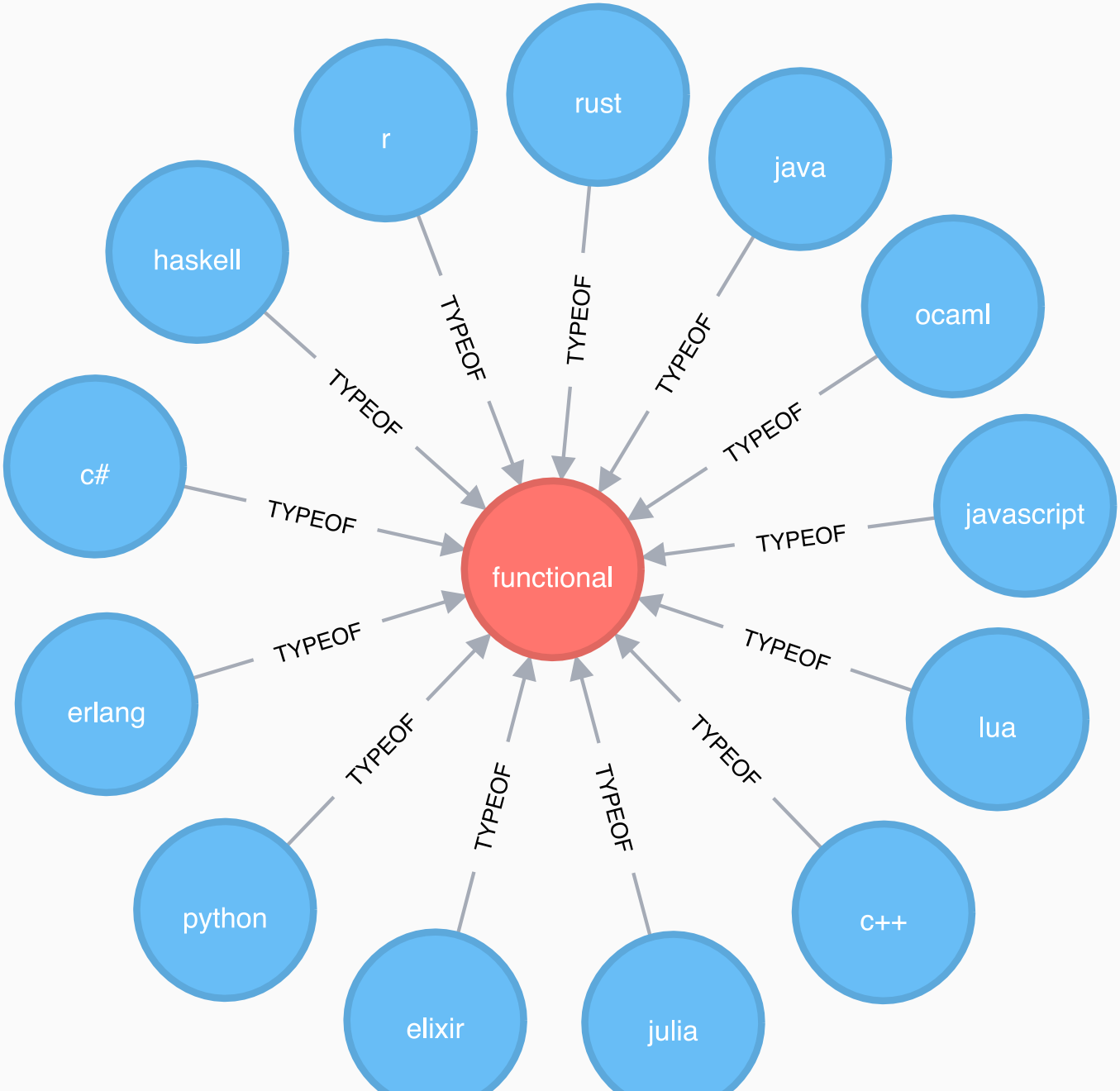




Language paradigms

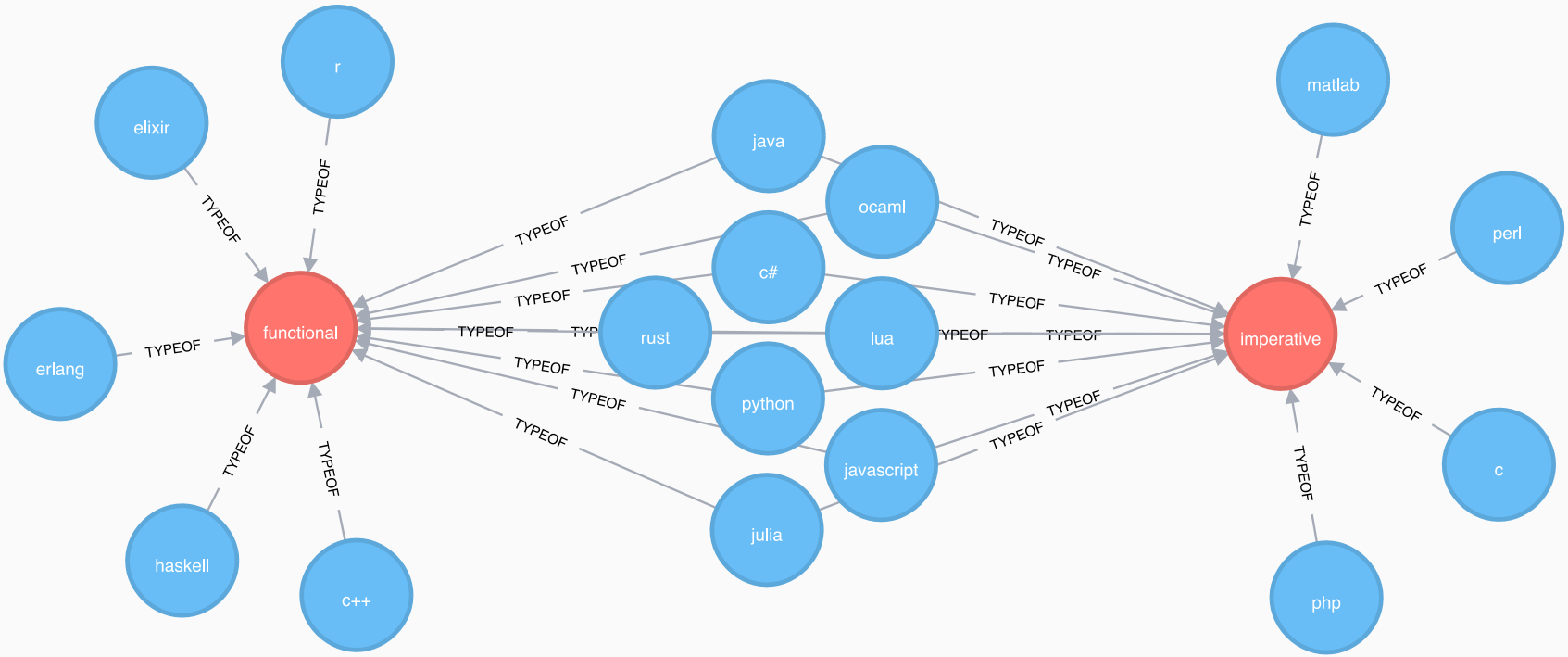
[Lisp] has assisted a number of our most gifted fellow humans in thinking previously impossible thoughts.
– Dijkstra, 1972

Lisp is a type of functional language



Programming languages have many paradigms

Functional versus imperative languages

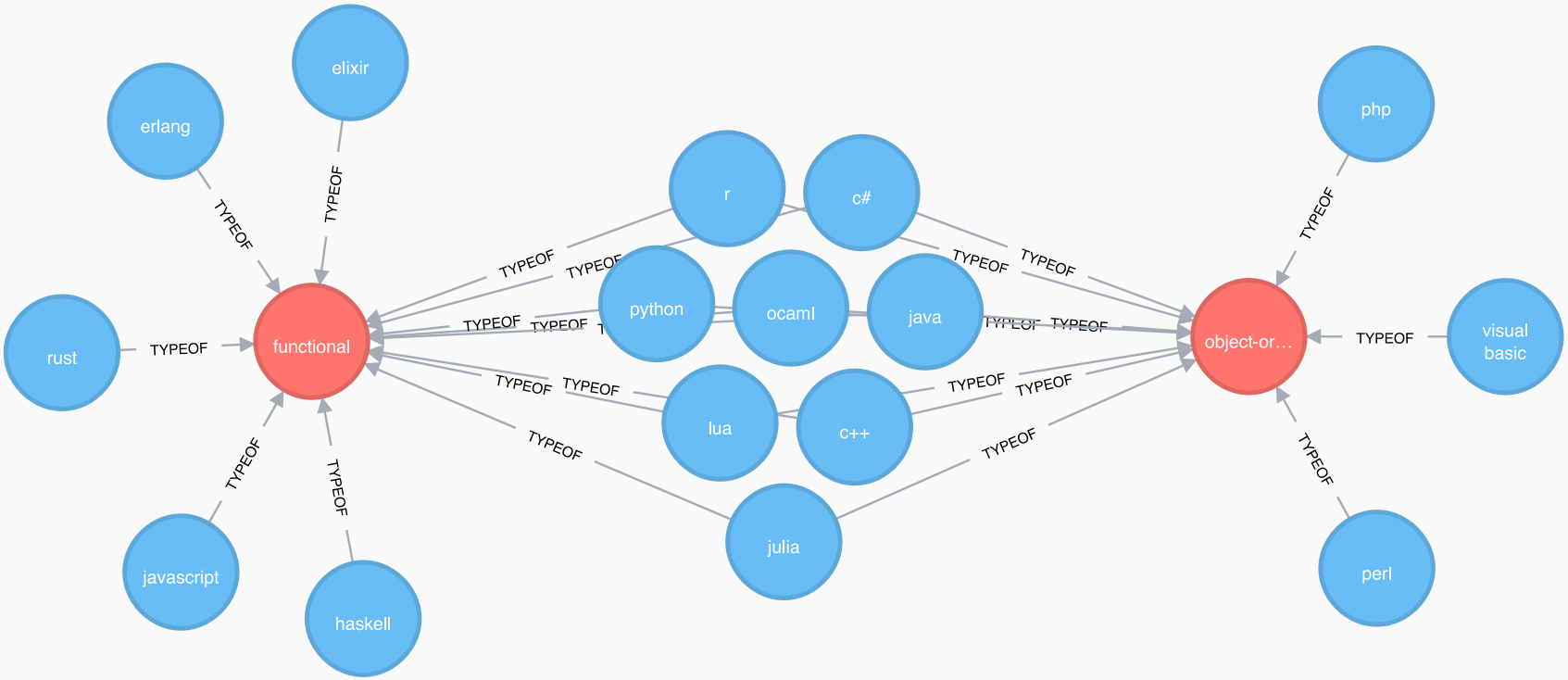


Tower of Hanoi: Imperative versus functional

```c
void hanoi(int n, int a, int b, int c)
{
  if (n > 0) {
    hanoi(n - 1, a, c, b);
    printf("Move disk from pole %d to pole %d\n", a, b);
    hanoi(n - 1, c, b, a);
  }
}
```

```haskell
hanoi :: Integer -> a -> a -> a -> [(a, a)]
hanoi 0 _ _ _ = []
hanoi n a b c = hanoi (n-1) a c b ++ [(a,b)] ++ hanoi (n-1)
```

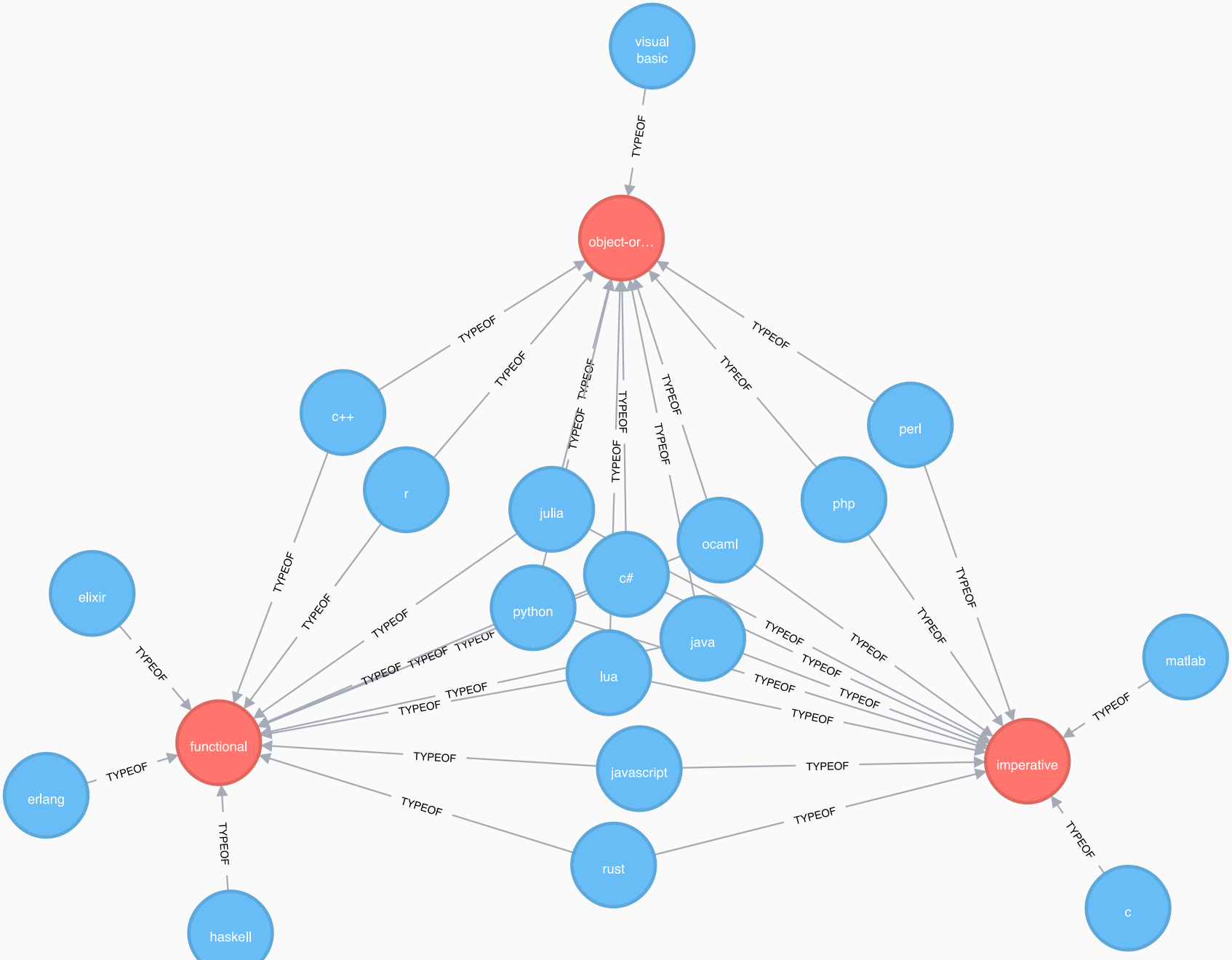
Functional versus object-oriented languages



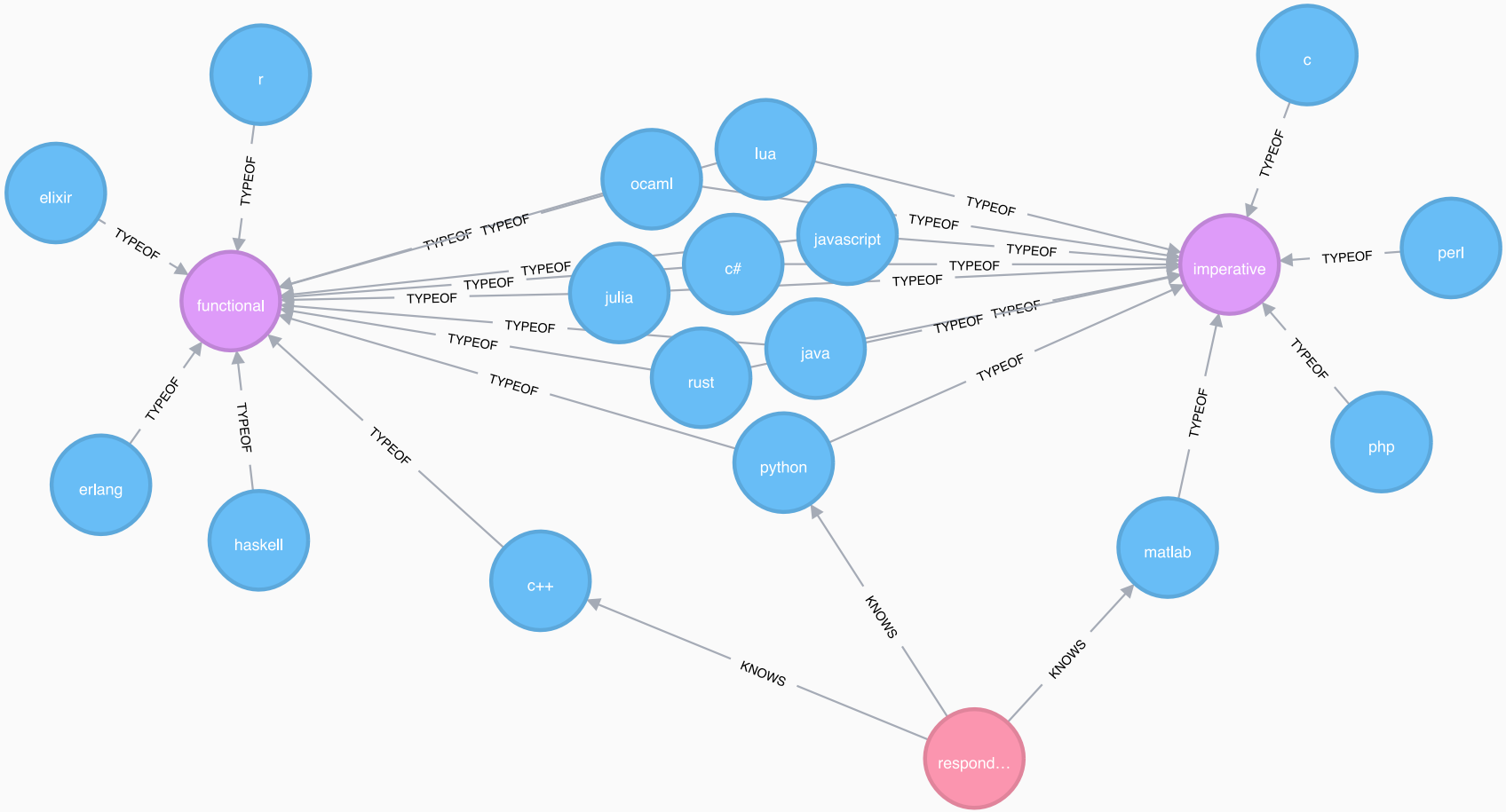
Tower of Hanoi: Object-oriented

```
#import <Foundation/NSObject.h>

@interface TowersOfHanoi: NSObject {
    int pegFrom;
    int pegTo;
    int pegVia;
    int numDisks;
}

-(void) setPegFrom: (int) from andSetPegTo: (int) to andSet
-(void) movePegFrom: (int) from andMovePegTo: (int) to andM
@end
```

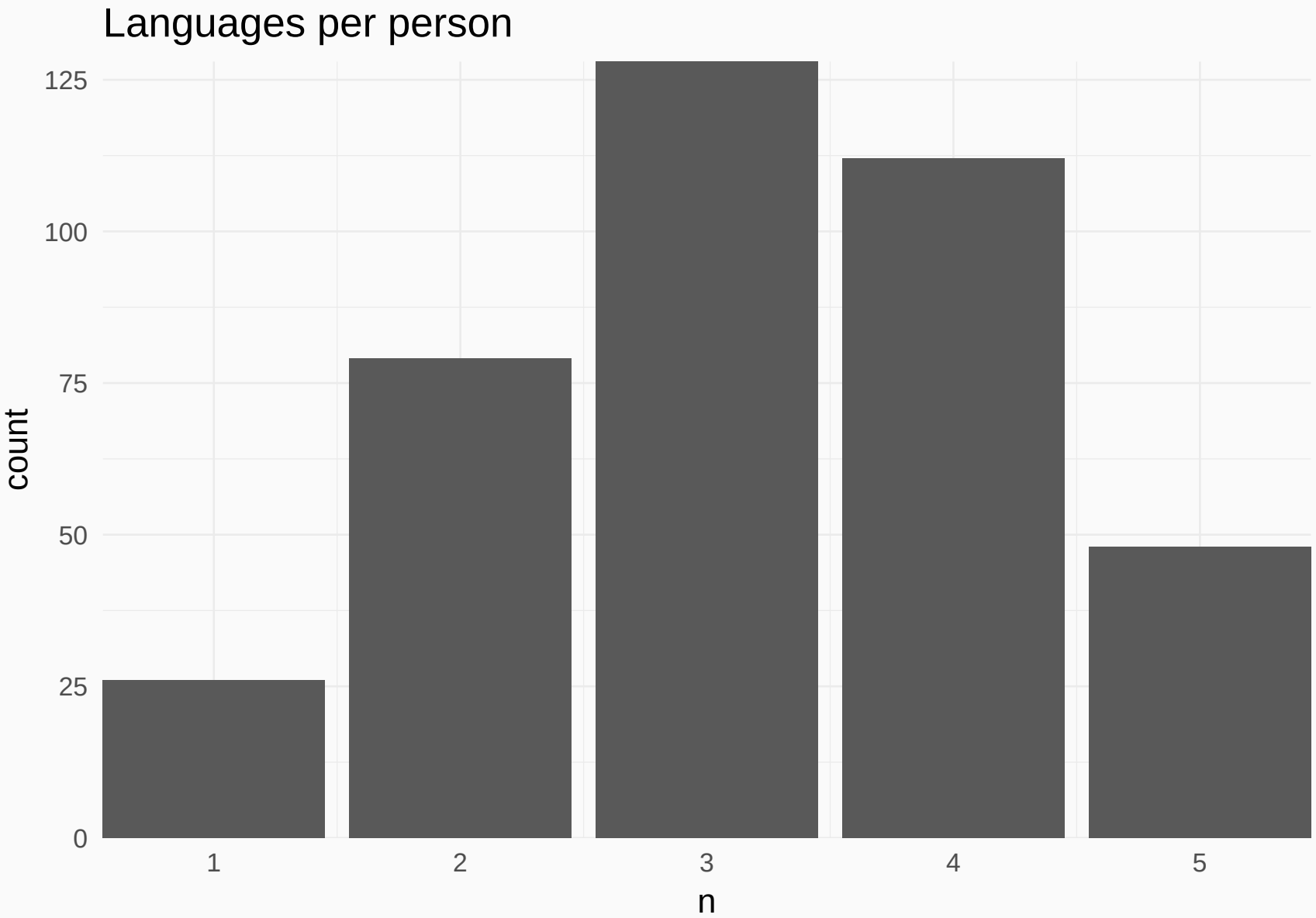
Programming paradigm venn



Programmers know multiple languages



Languages per programmer

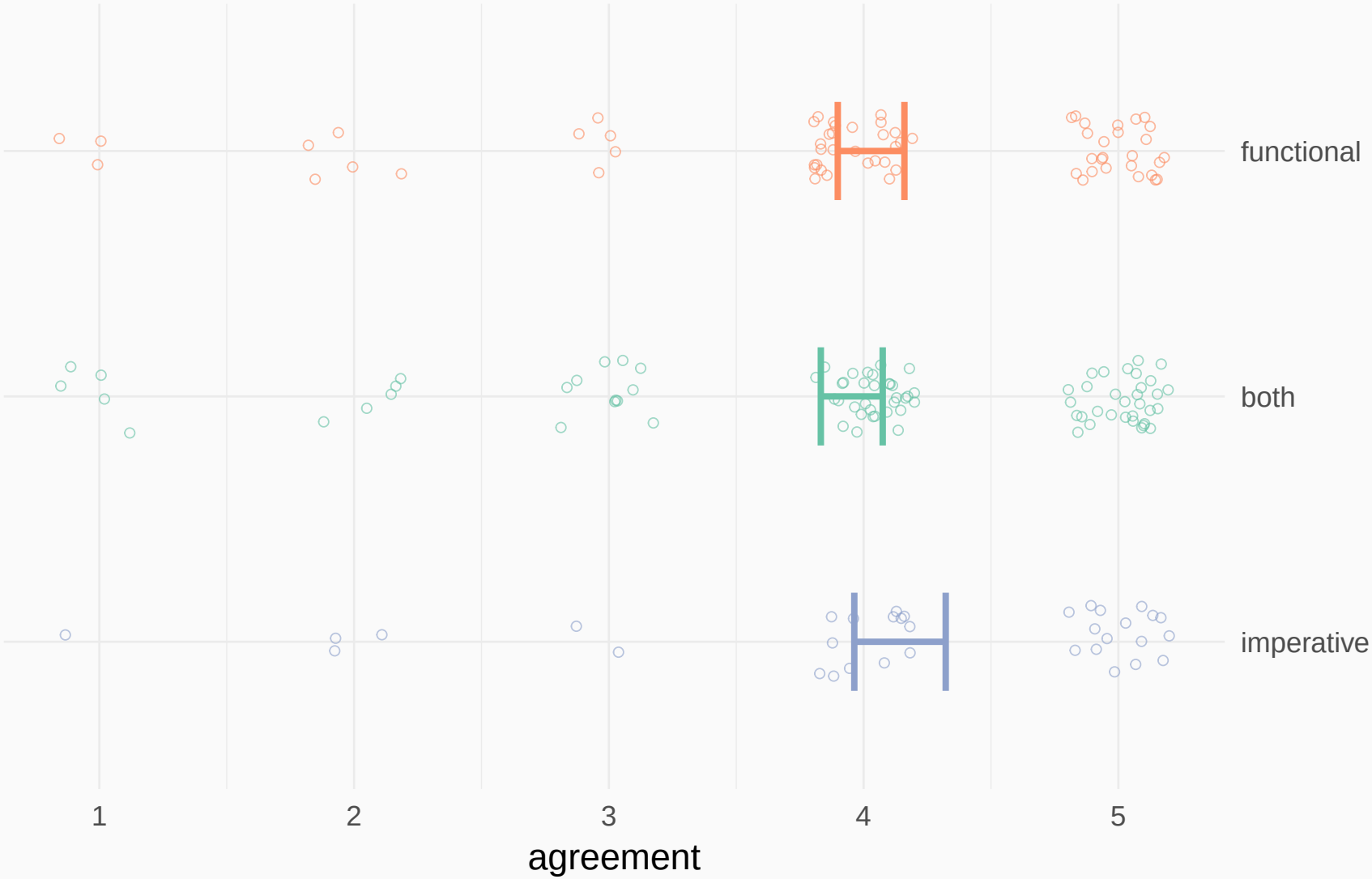


Solutions to the multiple languages problem

- Top language only.
- Count all reported languages.
- *Weigh languages based on self reported proficiency.*

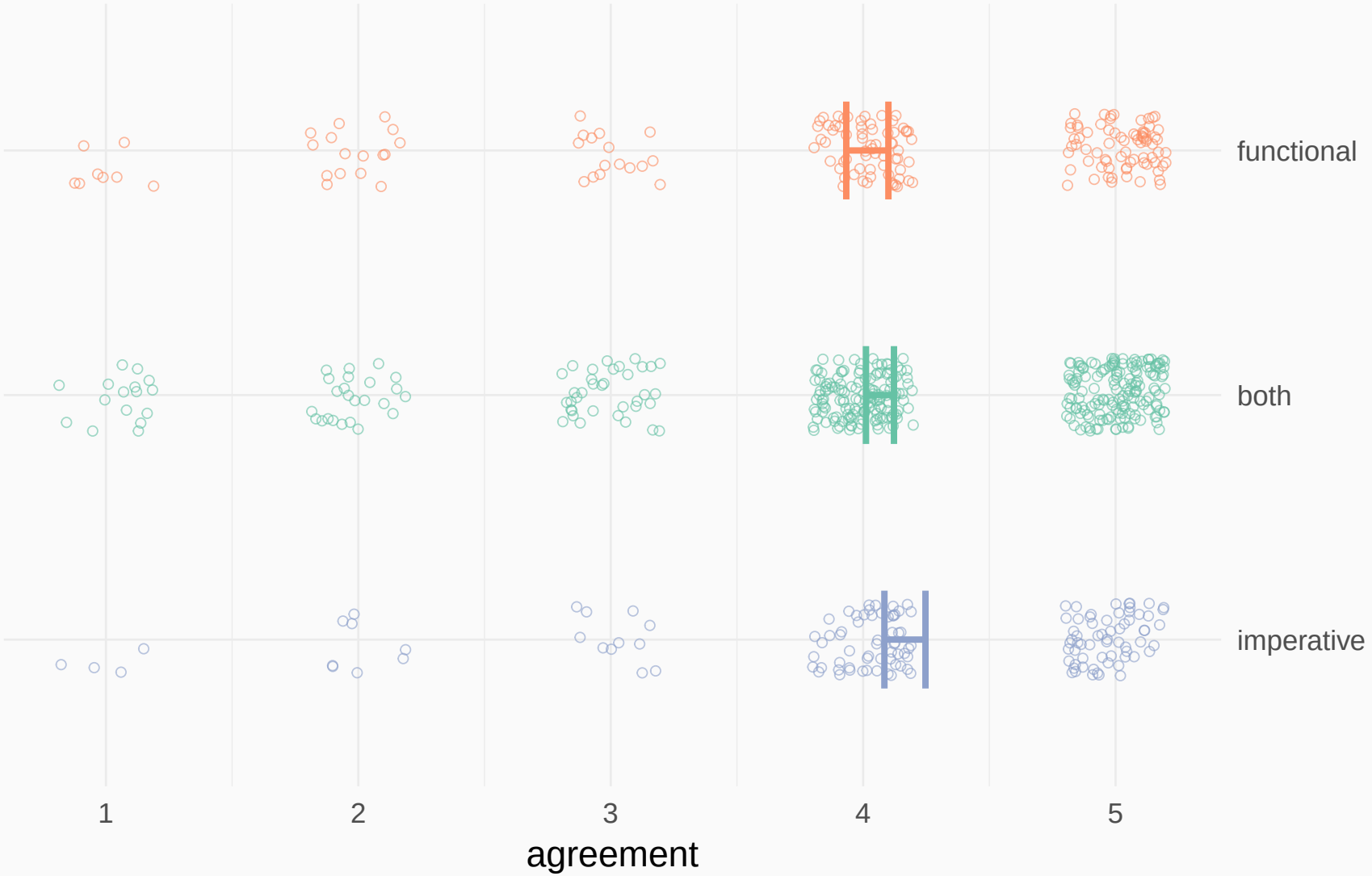
Changed reasoning: Top language

Learning to program has changed how I reason about things outside of coding



Changed reasoning: All languages

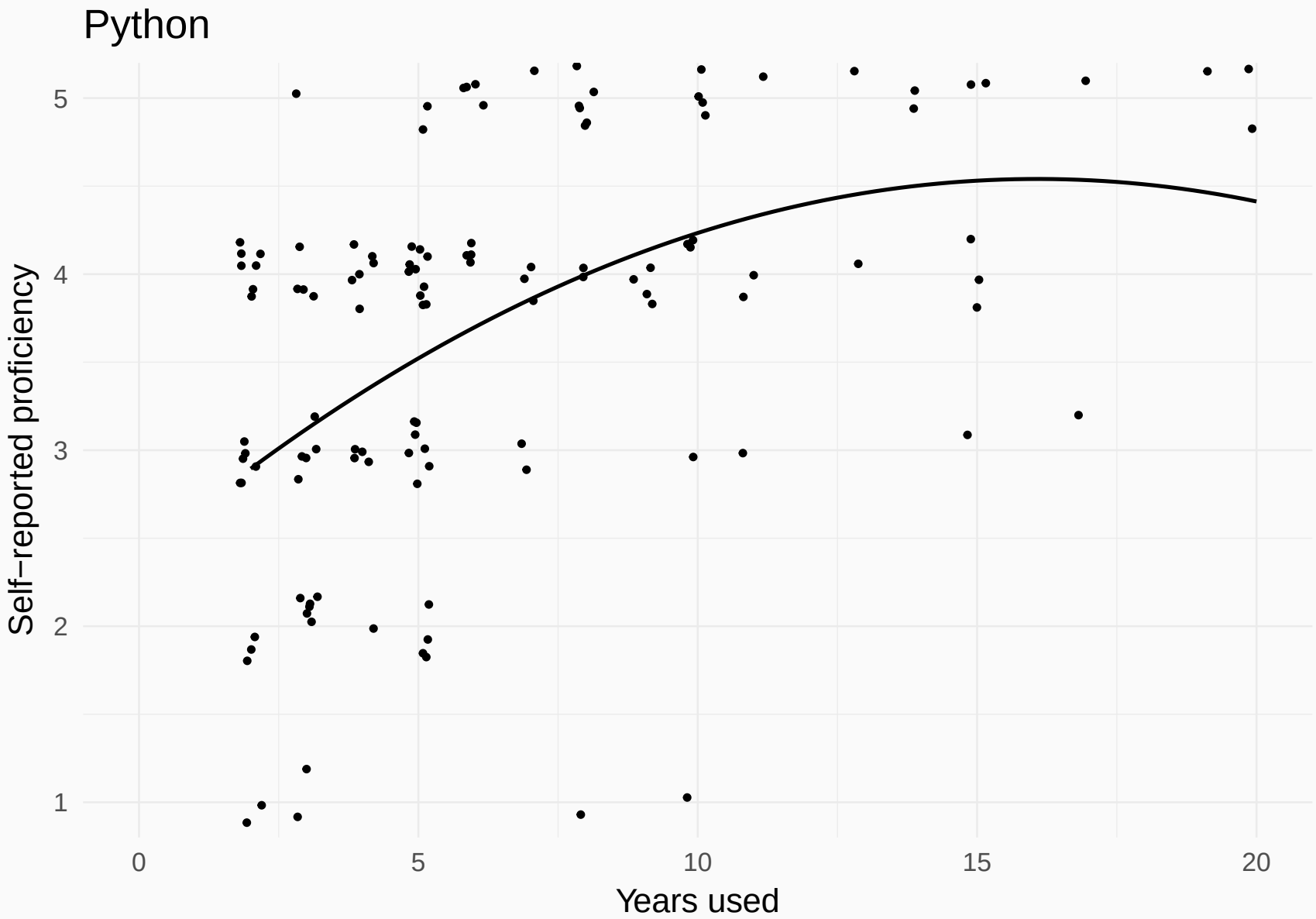
Learning to program has changed how I reason about things outside of coding



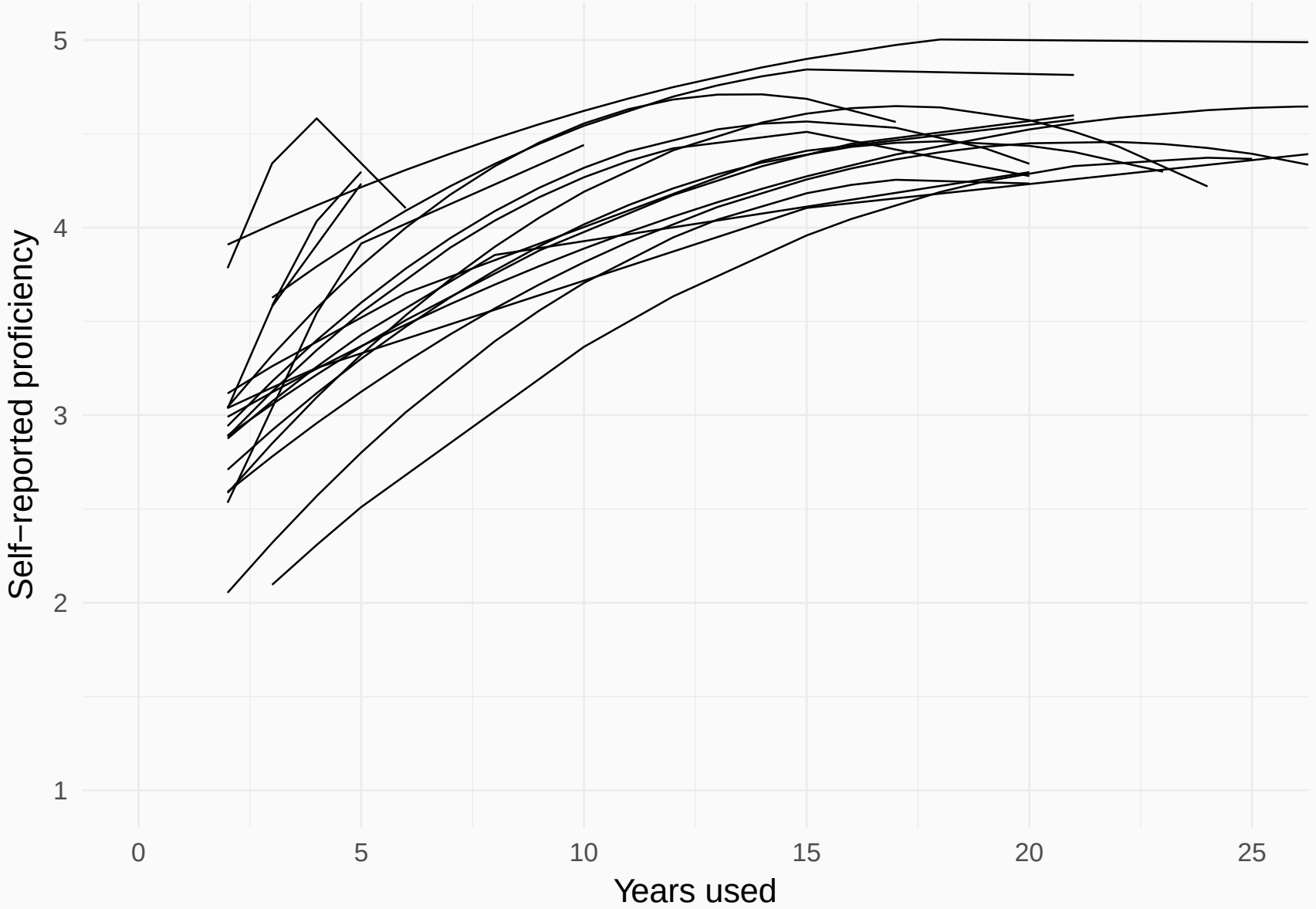
Changed reasoning: All languages

Learning to program has changed how I reason about things outside of coding

Language paradigms and years to proficiency



Experience and proficiency



Conclusions

Programming languages are a test bed for questions about human problem solving.
Pithy intuitions from computer scientists need to be tested with experimental work.
Maybe the differences between programming languages are overblown.