



# Computer code comprehension shares neural resources with formal logic and math

Yun-Fei Liu<sup>1</sup>, Marina Bedny<sup>1</sup>, Colin Wilson<sup>2</sup>

<sup>1</sup>Department of Psychological and Brain Sciences, Johns-Hopkins University; <sup>2</sup>Department of Cognitive Science, Johns-Hopkins University



## Questions

What's the neural basis underlying code comprehension?

- Idiosyncratic recruitment of neural circuits?
- Recycling **language** network?
- Domain-general **executive control** system?
- Sharing resource with other culturally derived symbol manipulation system?
- **Symbolic math**, **formal logical inference**

## Methods

### Participants:

15 programmers, at least 3 years of experience in Python

### Experiments:

#### Main task:

Real code	<pre>def fun(input):     result=["input"]     for ii in input:         result += ["%s"%ii]     return result</pre>	<pre>def fun(input):     result=["input"]     for ii in input:         result += ["%s"%ii]     return result</pre>	TRUE?
	+	+	#output# ["input", True, True, True]
			#input# "654"
	Presentation (24s)	"Input" (6s)	Response (6s)
Fake code	<pre>input :def fun(     result=input""[]) for: input ii in     result)%s%i+= " [%] result return</pre>	<pre>input :def fun(     result=input""[]) for: input ii in     result)%s%i+= " [%] result return</pre>	SAME?
	+	+	for: input ii in
			#input# "654"

#### "Sentence" Judgement task:

<b>Language</b>	The monkey that the fox fought stole the food.
Same meaning?	The food was stolen by the monkey that the fox fought.
<b>Mathematics</b>	X plus twenty-one equals fifty-eight
Same X value?	X plus twenty-two equals fifty-six
<b>Formal logic</b>	If not Y then both Z and X
Consistent?	If either not Z or not X then Y

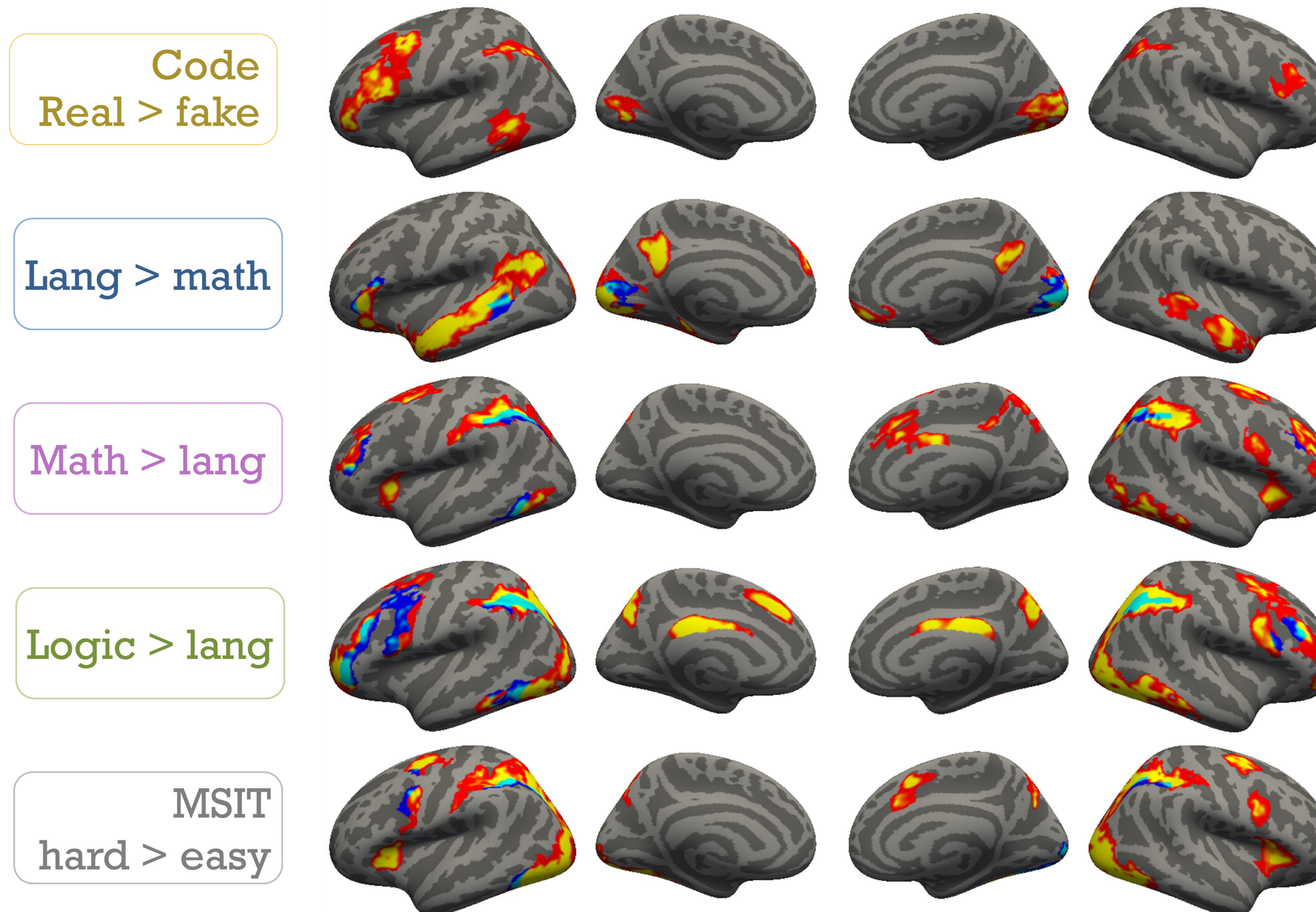
#### Multi-source interference task:



#### Decoding different types of functions:

**FOR** vs **IF** Multi-variate pattern analysis (MVPA) implemented with Support Vector Machine (SVM)

## Results

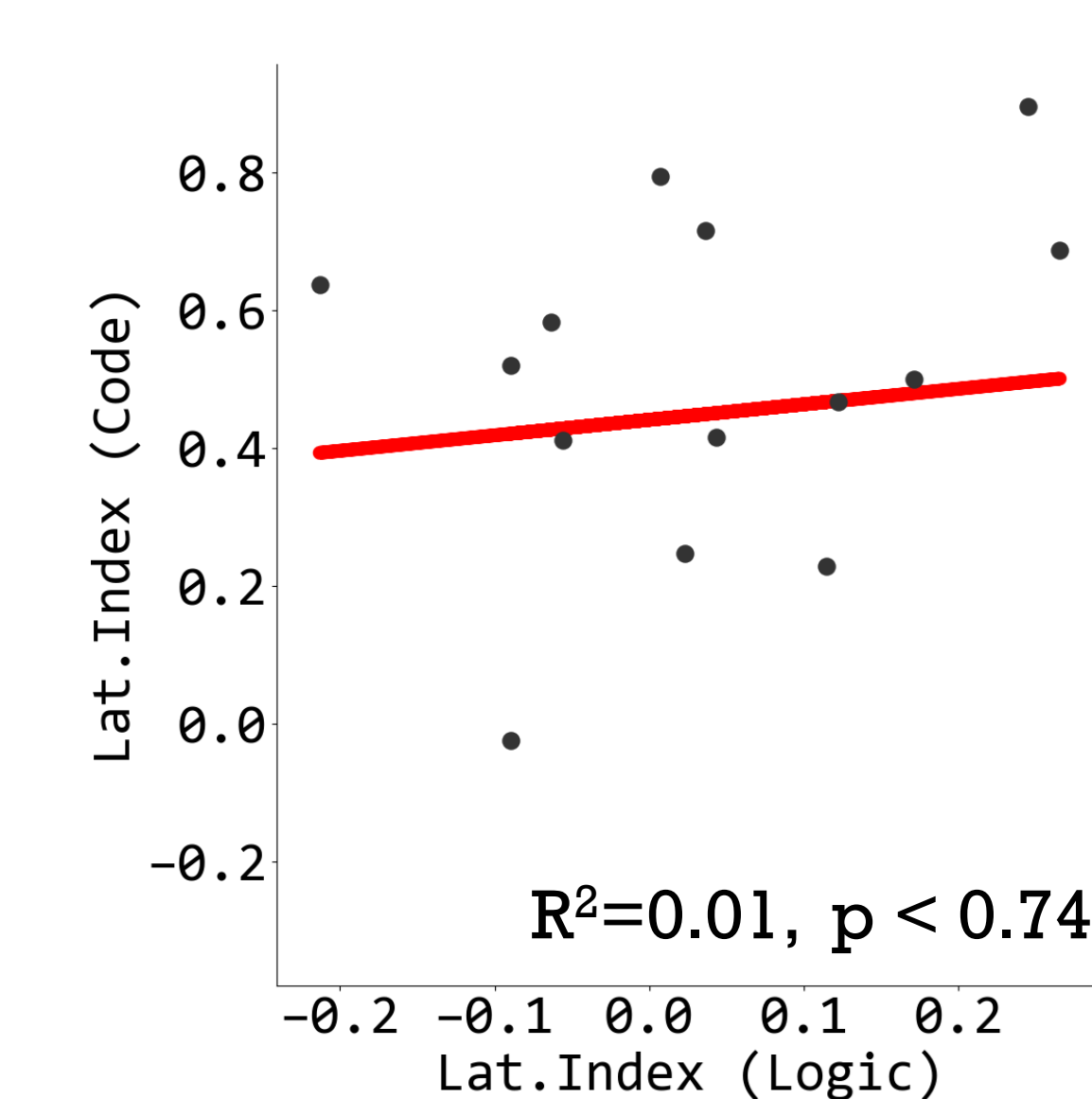
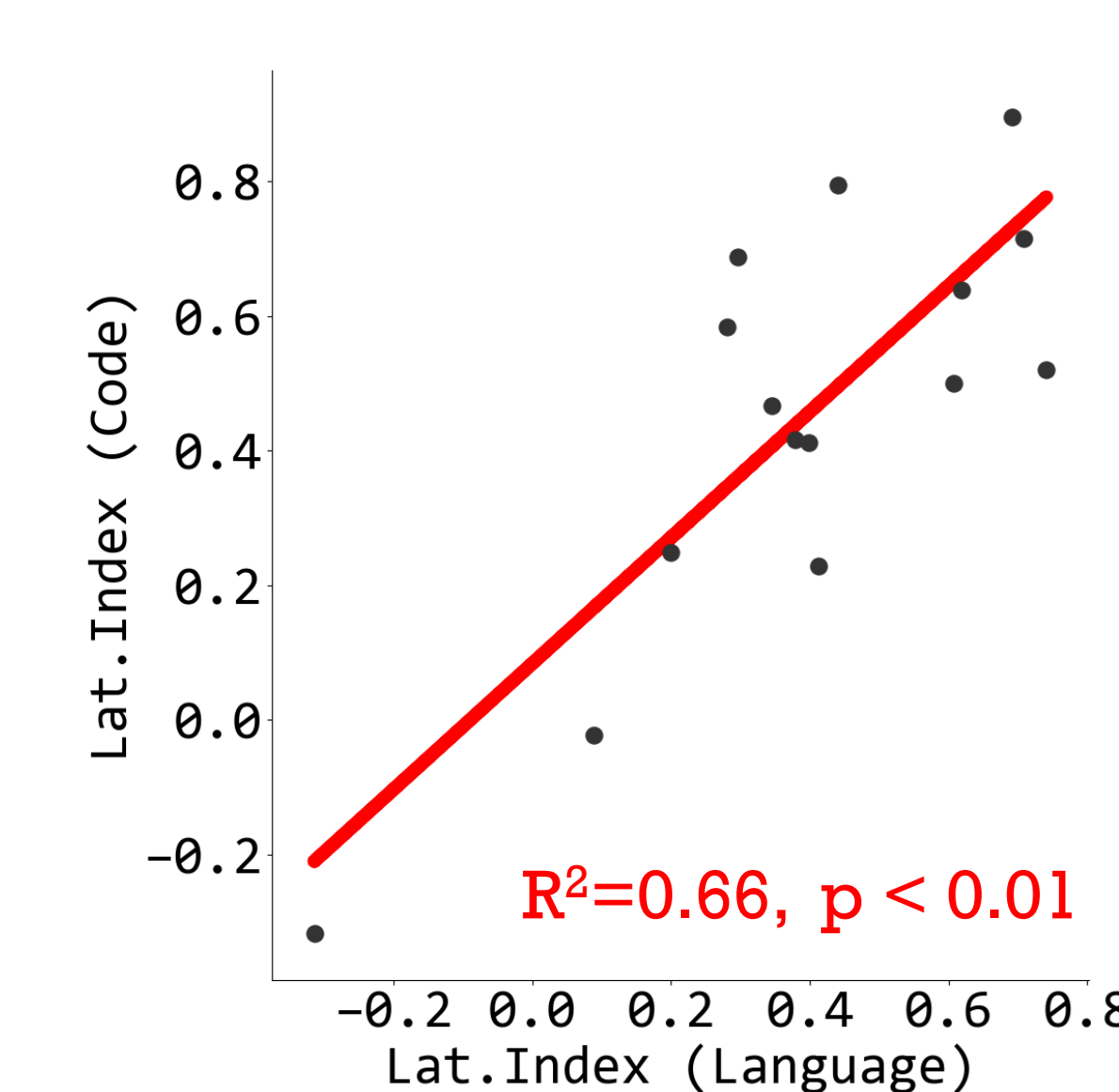
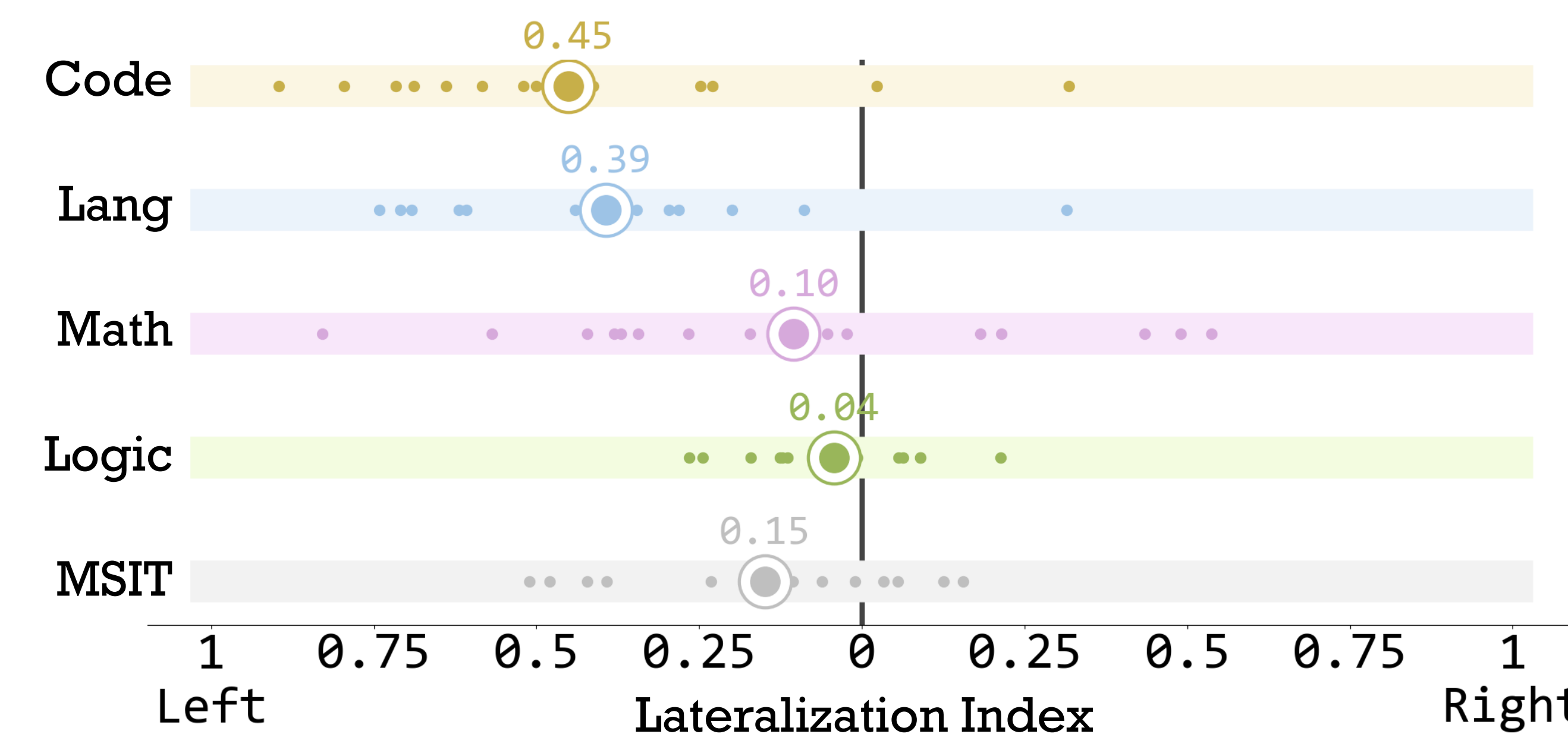
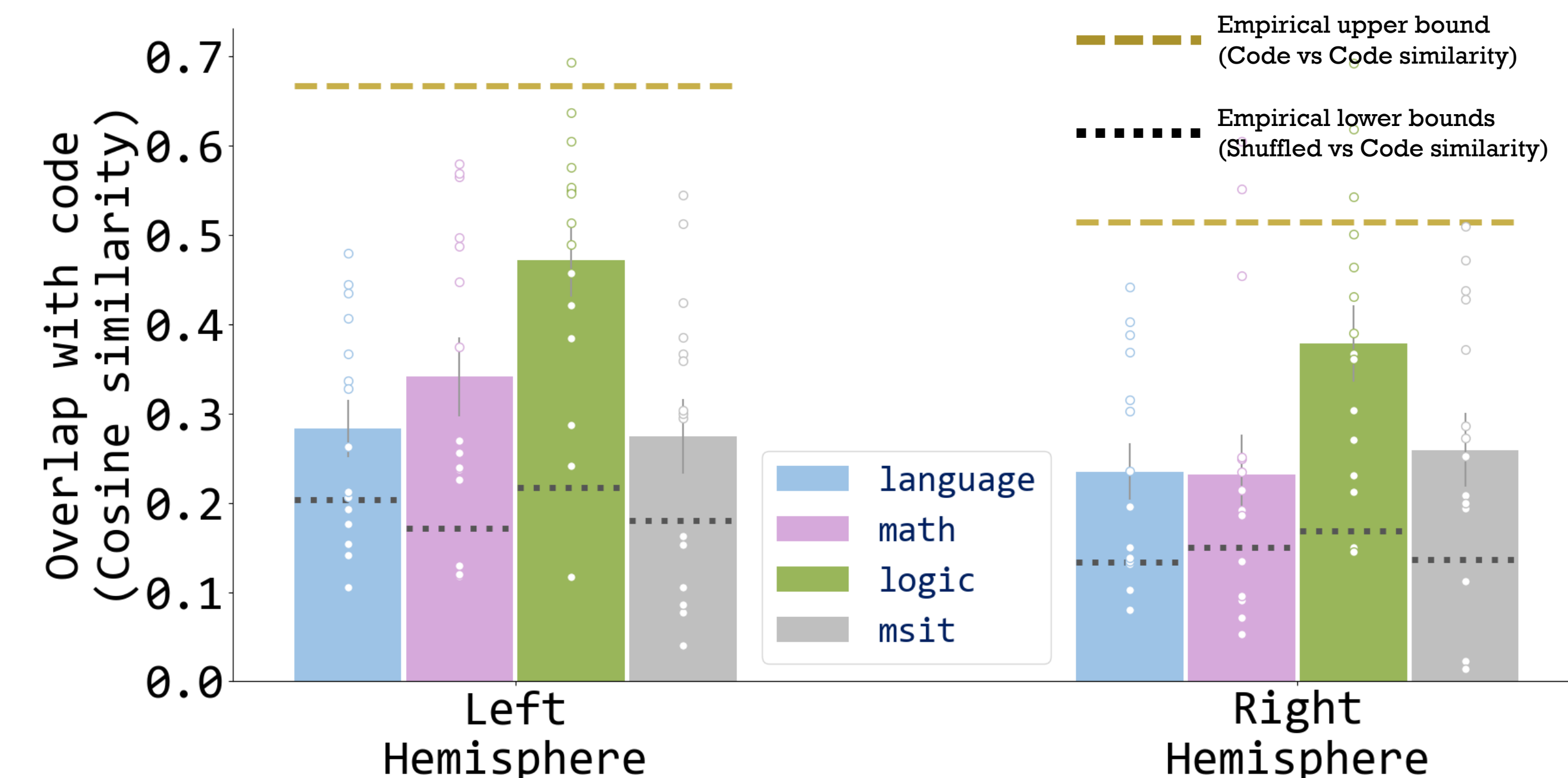
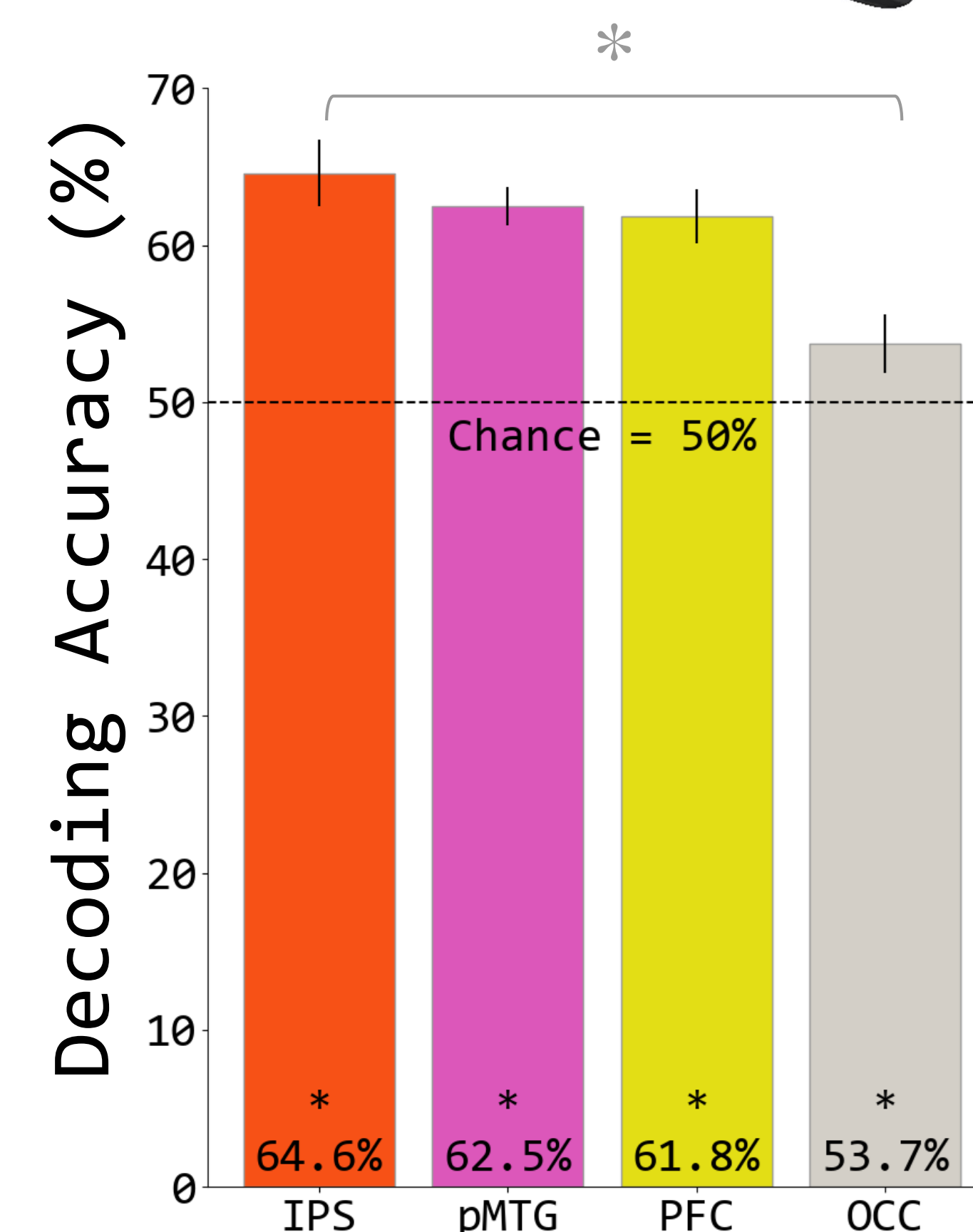


**FOR** → 

```
def fun(input):
    result = ["in"]
    for ii in input.split("="):
        result += [ii.isdigit()]
    return result
```

**IF** → 

```
def fun(input):
    result = "if:"
    if input.isdigit():
        result += input.upper()
    return result
```



## Conclusion

- A consistent left-lateralized network was activated during code comprehension relative to fake code trials.
- The regions activated during code comprehension greatly overlap with the regions activated during formal logical inference.
- The lateralization index for code comprehension is significantly correlated to the index for language comprehension.

**Code comprehension may depend on the same neural substrate as logic, and probably math, but the acquisition of code comprehension may depend on the development of the language network.**

## References

1. Amalric and Dehaene (2016). PNAS
2. Bush and Shin (2006). Nat. Protoc.
3. Dehaene and Cohen (2007). Neuron
4. Fedorenko et al. (2019). Trends Cogn. Sci.
5. Kanjlia, S., et al. (2016). PNAS
6. Monti, et al. (2009). PNAS