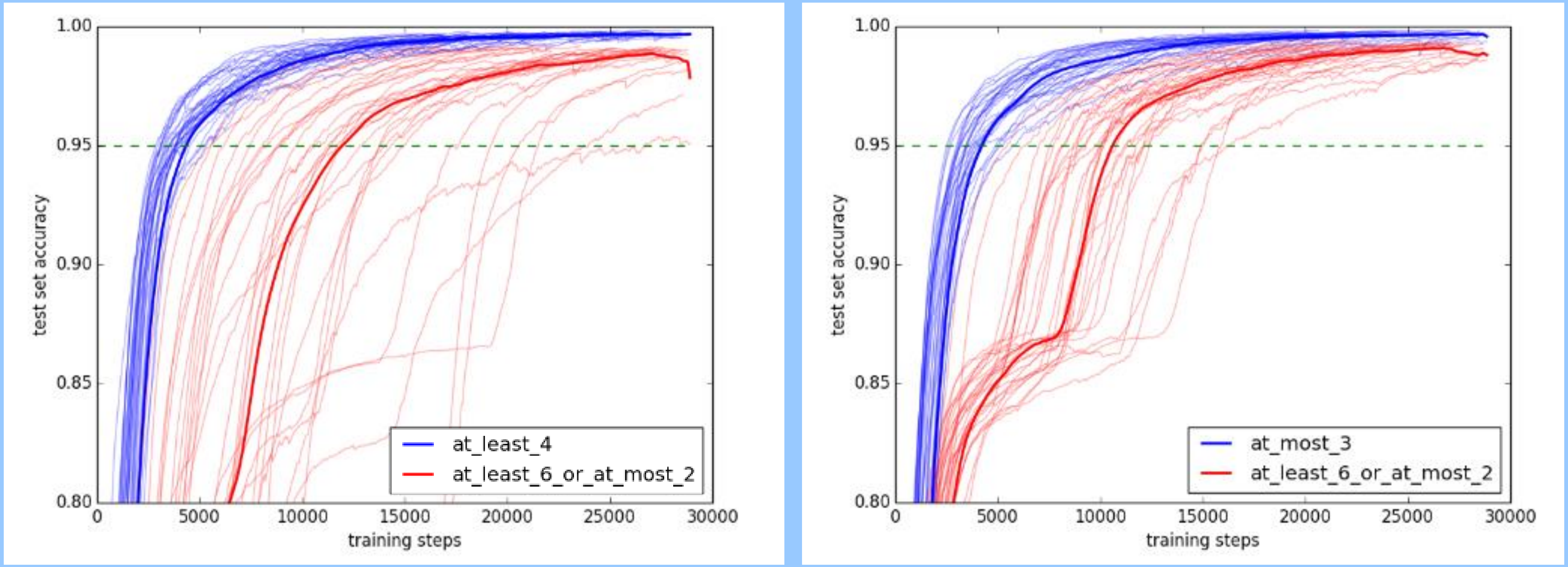
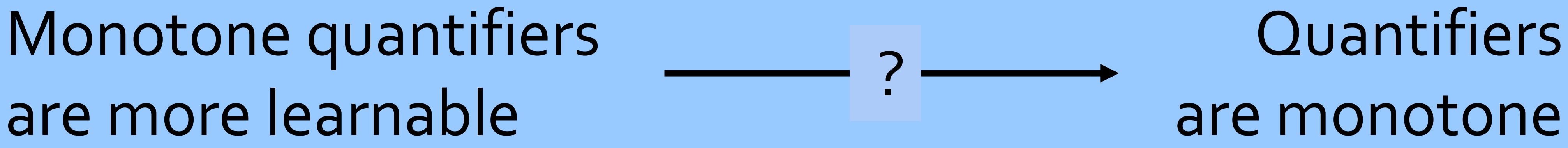
# Cultural Evolution

Evolving a language

ESSLLI 2022 – Fausto Carcassi & Jakub Szymanik



A neural network can learn monotone quantifiers faster than non-monotone quantifiers.



Monotone quantifiers
are more learnable

?

Quantifiers
are monotone

Problem of Linkage
(Kirby 1999)

# Iterated Learning:
# The idea

# Cultural evolution

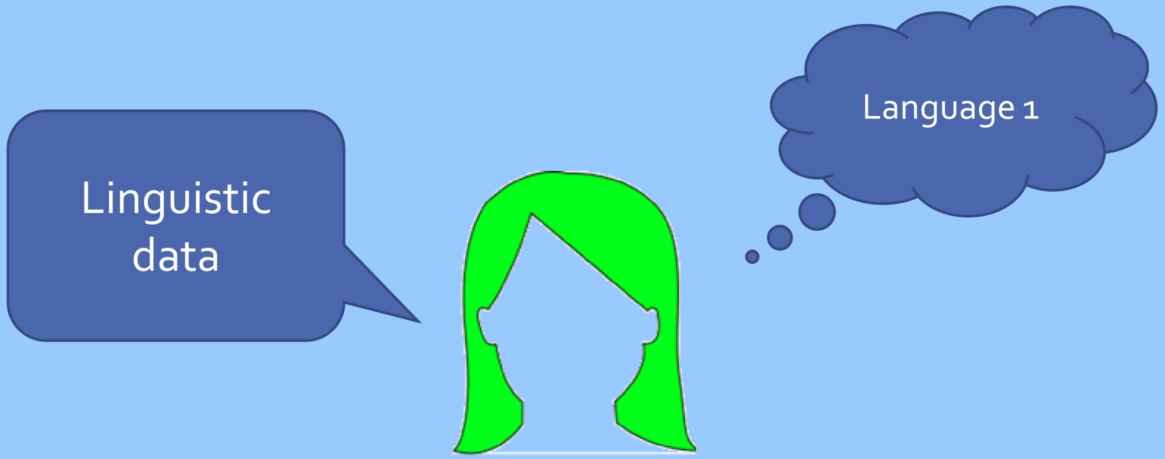
- Culture is hard to define
  - One sense just includes music, art, and films.
  - Our sense wider: roughly includes everything humans *learn* in virtue of belonging to a certain community.
  - This includes: how to sit, eat,  play, who Joanna Newsom is, and *language*
- *Cultural evolution*
  - What are the rules that govern the way culture changes?

"The structure of a language is under intense selection because in its reproduction from generation to generation, it must pass through a narrow bottleneck: children's minds"

- Deacon (1997: 110)

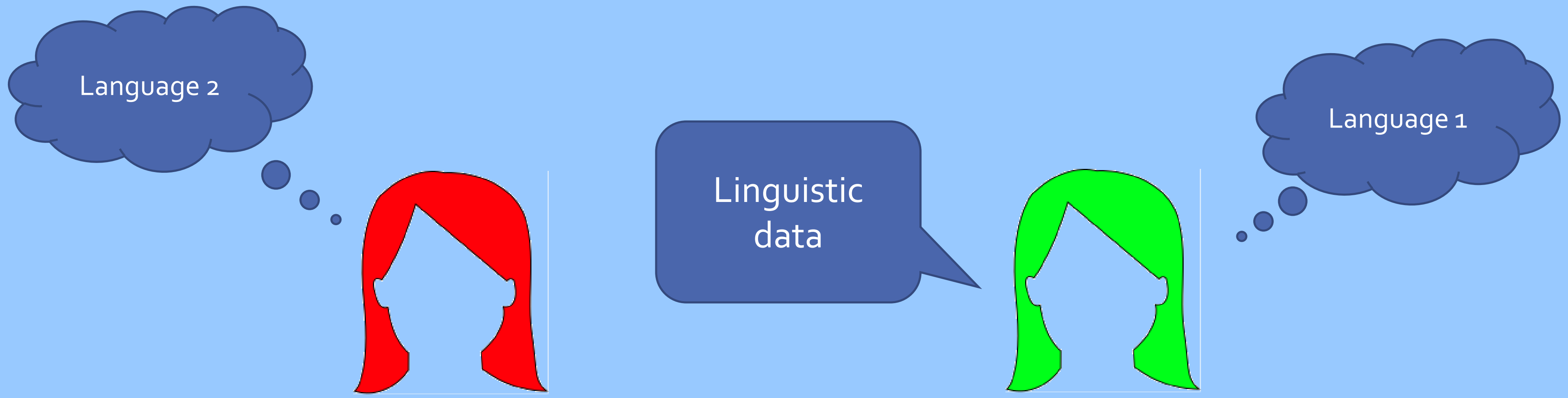
# The iterated learning model

- We can model cultural evolution by an iterated process of change



# The iterated learning model

- We can model cultural evolution by an iterated process of change

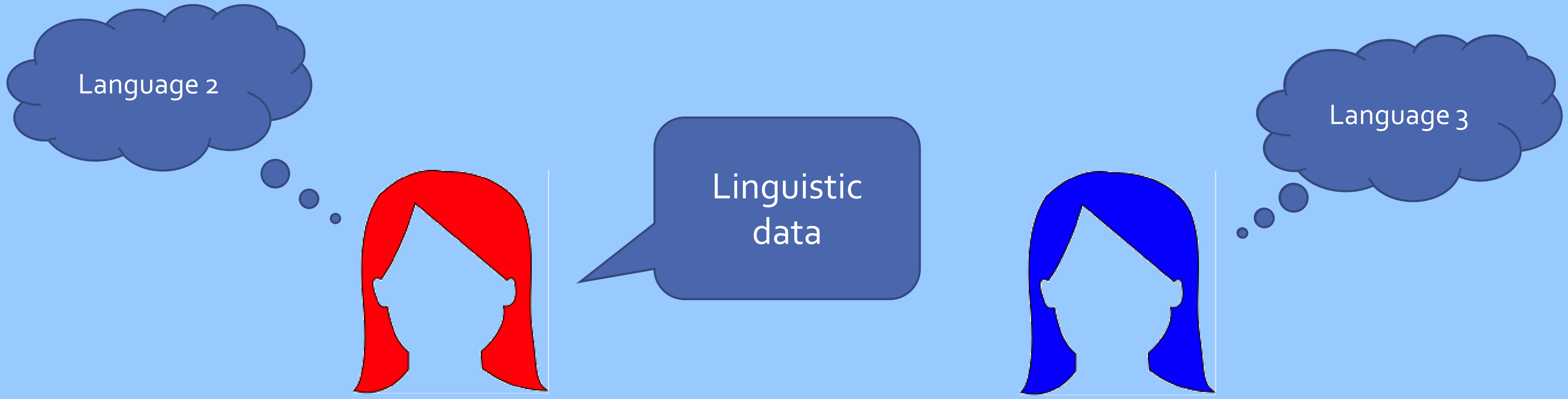


# The iterated learning model

- We can model cultural evolution by an iterated process of change

# The iterated learning model

- We can model cultural evolution by an iterated process of change



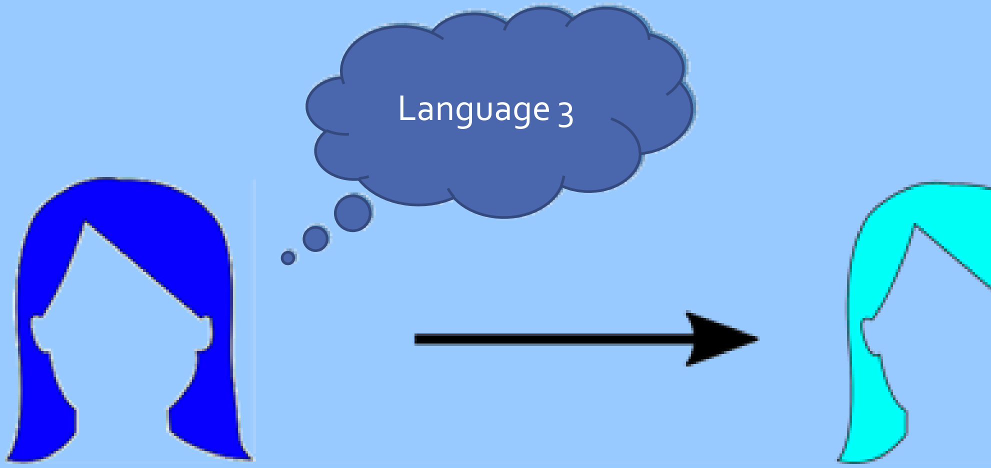
# The iterated learning model

- We can model cultural evolution by an iterated process of change

Language 2
Language 3

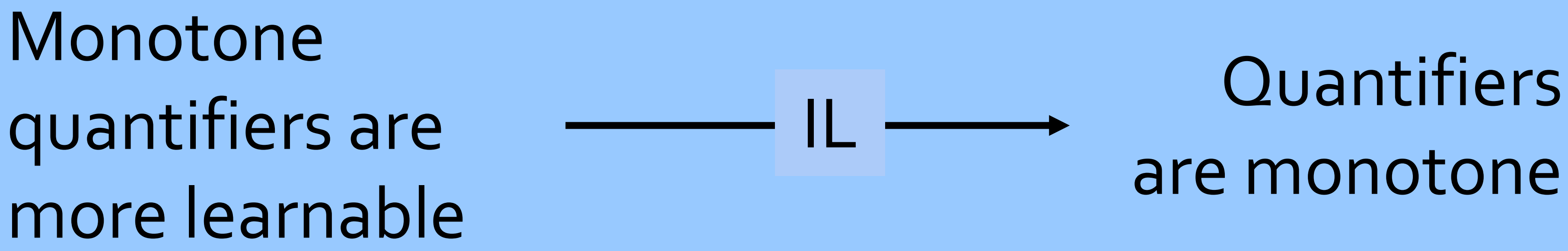
# The iterated learning model

- We can model cultural evolution by an iterated process of change
- Iterated Learning shows the effects of cognitive structure on language structure



# The iterated learning model

Iterated Learning shows the effects of cognitive structure on language structure.



Monotone quantifiers are more learnable

IL

Quantifiers are monotone

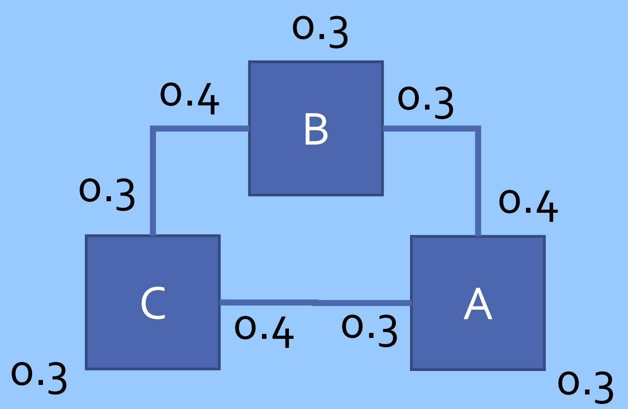
# The iterated learning model

- Some simplifications:
    - Only one cultural parent for each cultural child
    - No horizontal transmission
- Iterated learning *reveals* learning biases!
- Iterated learning is a *mechanism* for learnability to influence language.
- We can explain some universals as the result of IL + certain learning biases.
- We can look at IL through the lens of the theory of *Markov Chains*

# Iterated Learning & Markov Chains

# Markov chains: An example

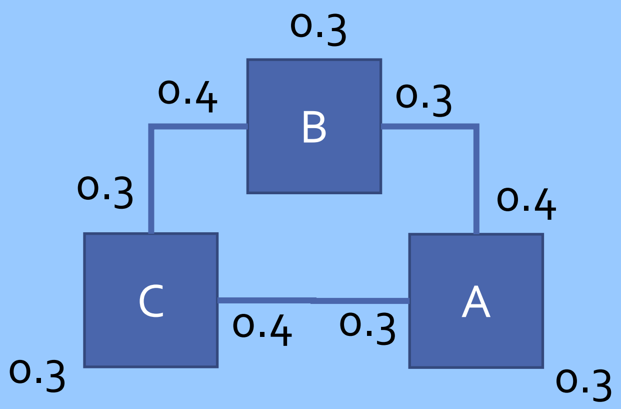
- Suppose there are three rooms connected by corridors, as follows:



- You start in room A, and then move (facing the center) :
  - left with probability 0.3
  - right with probability 0.4
  - stay where you are with prob 0.3
- A *Markov Chain* is (roughly) a (discrete-time) process that
  - Changes state stochastically and
  - Whose state at time *t* only depends on the state at time *t-1* (Markov condition)

# Markov chains: An example

- Question: how do we calculate the probability that you will be in B in three steps?
- We can represent the process with a *transition matrix*:



| | A | B | C |
|---|---|---|---|
| A | 0.3 | 0.4 | 0.3 |
| B | 0.3 | 0.3 | 0.4 |
| C | 0.4 | 0.3 | 0.3 |

- We start with a one-hot vector that indicates we are in B: [0, 1, 0]
- And then do matrix multiplication three times

# Markov chains: An example

| | | |
|---|---|---|
| 0 | 1 | 0 |

| | A | B | C |
|---|---|---|---|
| A | 0.3 | 0.4 | 0.3 |
| B | 0.3 | 0.3 | 0.4 |
| C | 0.4 | 0.3 | 0.3 |

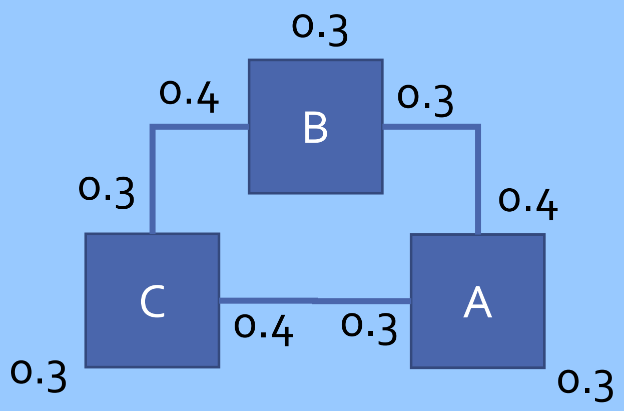
| | | |
|---|---|---|
| 0.3 | 0.3 | 0.4 |

| | A | B | C |
|---|---|---|---|
| A | 0.3 | 0.4 | 0.3 |
| B | 0.3 | 0.3 | 0.4 |
| C | 0.4 | 0.3 | 0.3 |

| | | |
|---|---|---|
| 0.34 | 0.33 | 0.33 |

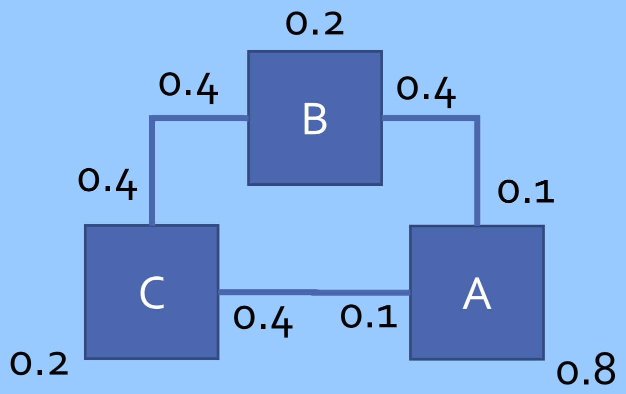
| | A | B | C |
|---|---|---|---|
| A | 0.3 | 0.4 | 0.3 |
| B | 0.3 | 0.3 | 0.4 |
| C | 0.4 | 0.3 | 0.3 |

| | | |
|---|---|---|
| 0.333 | 0.334 | 0.333 |



# Markov chains: An example

- Initial position washed out pretty fast!
  - We are going towards a uniform distribution.
  - The differences are all *relative* to where one is.
  - Rather than ‘pointing’ to a specific place.
- Suppose instead that you have a preference to stay in room A, whenever you end up there:



| | A | B | C |
|---|---|---|---|
| A | 0.8 | 0.1 | 0.1 |
| B | 0.4 | 0.2 | 0.4 |
| C | 0.4 | 0.4 | 0.2 |

# Markov chains: An example

| | | |
|---|---|---|
| 0 | 1 | 0 |

| | A | B | C |
|---|---|---|---|
| A | 0.8 | 0.1 | 0.1 |
| B | 0.4 | 0.2 | 0.4 |
| C | 0.4 | 0.4 | 0.2 |

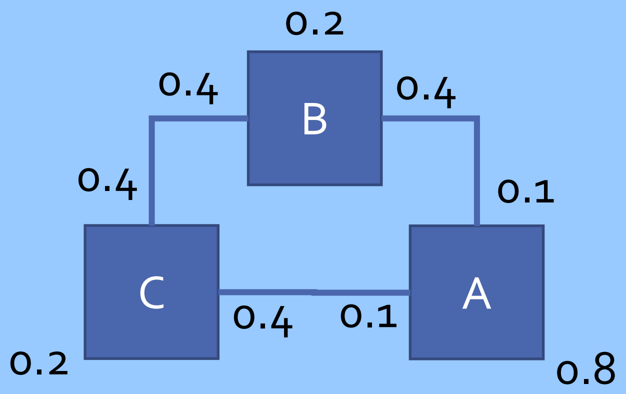
| | | |
|---|---|---|
| 0.4 | 0.2 | 0.4 |

| | A | B | C |
|---|---|---|---|
| A | 0.8 | 0.1 | 0.1 |
| B | 0.4 | 0.2 | 0.4 |
| C | 0.4 | 0.4 | 0.2 |

| | | |
|---|---|---|
| 0.56 | 0.24 | 0.2 |

| | A | B | C |
|---|---|---|---|
| A | 0.8 | 0.1 | 0.1 |
| B | 0.4 | 0.2 | 0.4 |
| C | 0.4 | 0.4 | 0.2 |

| | | |
|---|---|---|
| 0.624 | 0.184 | 0.192 |



# Stationary distribution

- When the situation is not symmetric across rooms, you might always tend to end up in a certain room over time
- Two possible reasons:
    - When we get to A, we stick to it
    - When we are in a different room, we tend to go to A
- We can see time evolution at https://www.mathematik.tu-clausthal.de/en/mathematics-interactive/simulation/markov-chain-discrete/
- Over time, it does not matter where one starts, one reaches a certain probability of being in each room
    - (Note: condition of ergodicity)
- This distribution that we tend towards over time is called *stationary*
- Important insight: time average == space average

- *Where the hell am I going with this?*

# Iterated learning *is* a Markov chain

- Now imagine:
    - Instead of rooms we have all the possible languages (or lang fragments)
    - Instead of ‘timesteps’ we have generations of cultural transmission
    - Instead of ‘moving’ we have ‘acquiring from parent’
        - Conditional distribution over learner’s language given teacher’s
- Iterated Learning as a Markov Chain
    - At each new generation, the learner acquires a language from the teacher
    - How could the amount of data seen by the learner affect the process?
    - Language spoken at each generation only depends on previous one
    - Iterated learning can be thought of as a Markov Chain
    - …and therefore it has a stationary distribution!
        - (With some weak assumptions)

# The stationary distribution of IL

- Deep insight:
    - Under IL, it doesn't matter where we start!
    - (Assuming ergodicity)
    - IL isn't (necessarily) a *diachronic* model
- How do we find the stationary distribution?
- *Model* iterated learning
    - Pick a model of the domain (set of possible languages)
    - Pick a model of learning
    - Run simulation (or find first eigenvalue of transition matrix)
- Let's look at two models of learning: Bayesian learning and neural learning

# Summary of the situation

- We started with the linkage problem: how does learnability influence language?
- This can be answered by iterated learning as a model of cultural evolution
- We saw that iterated learning can be interpreted as a Markov chain
- We need two ingredients for implementing IL as a Markov chain:
  - The space of possible states (langs)
  - A model of transition (learning)

- Bayesian inference is a model of language acquisition to combine with IL!

# Bayesian Iterated Learning

# Bayes' theorem

- Bayes' theorem is easy to prove:

$$
\begin{aligned}
P(H\&D) &= P(H \mid D)P(D) \\
&= P(D \mid H)P(H) \\
P(H \mid D)P(D) &= P(D \mid H)P(H) \\
P(H \mid D) &= \frac{P(D \mid H)P(H)}{P(D)}
\end{aligned}
$$

- And hard to understand!
- Four components: prior, likelihood, posterior, evidence

# Bayesian learning

- Usually, we apply Bayes theorem to calculate P(H | D), where:
  - H is unobservable
  - D is observable
- You can think of an application of Bayes theorem as a way of updating one's model of the world when new data comes in.
- A prior and a posterior then are relative to *one update*
- So we can think of one application of Bayes' theorem as an update in the state of knowledge given some data
- This gives a very natural way of thinking about the way humans could update their picture of unknown quantities given a stream of new evidence.

# Language learning in Bayesian agents

- In language acquisition
    - The hypotheses are possible languages / semantic objects
    - The data is linguistic data produced by the hypotheses
- Ex1
    - H: section of conceptual space (nominal meaning)
    - D: set of objects to which the noun applies
- Ex2
    - H: section of a scale (adjectival meaning)
    - D: set of tuples (degree, truth value)

# Language Learning in Bayesian agents

In Bayesian language acquisition:

- **Prior**
  - Encodes the cognitive biases towards some languages over others
  - NOTE: This need not be language specific!
  - E.g. simplicity bias is not linguistic
- **Likelihood**
  - Encodes the probability that a language user with a specific language would produce each possible utterance in each possible situation
- **Posterior**
  - The probability of each possible language given the observed utterance/situation
- Let's see a simple example of Bayesian language acquisition

# A simple example

- Two objects, two words
- Each word refers to some of the objects
- Possible languages (at least one obj per word and one word per obj):

| L1 | o1 | o2 |
|---|---|---|
| w1 | 1 | 1 |
| w2 | 1 | 1 |

| L2 | o1 | o2 |
|---|---|---|
| w1 | 1 | 0 |
| w2 | 1 | 1 |

| L3 | o1 | o2 |
|---|---|---|
| w1 | 1 | 1 |
| w2 | 1 | 0 |

| L4 | o1 | o2 |
|---|---|---|
| w1 | 1 | 1 |
| w2 | 0 | 1 |

| L5 | o1 | o2 |
|---|---|---|
| w1 | 0 | 1 |
| w2 | 1 | 1 |

| L6 | o1 | o2 |
|---|---|---|
| w1 | 0 | 1 |
| w2 | 1 | 0 |

| L7 | o1 | o2 |
|---|---|---|
| w1 | 1 | 0 |
| w2 | 0 | 1 |

- Suppose that there is a bias against ambiguity, e.g. this prior:

| L1 | L2 | L3 | L4 | L5 | L6 | L7 |
|---|---|---|---|---|---|---|
| 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.25 | 0.25 |

# A simple example

- Suppose the data is generated as follows:
  - The speaker sees one object sampled at random
  - Then they sample among the utterances compatible with the objects
- If we observed one datapoint from L2, (o1, w1), the likelihood would be:

0.5

| L1 | o1 | o2 |
|---|---|---|
| w1 | 1 | 1 |
| w2 | 1 | 1 |

1.

| L2 | o1 | o2 |
|---|---|---|
| w1 | 1 | 0 |
| w2 | 1 | 1 |

0.5

| L3 | o1 | o2 |
|---|---|---|
| w1 | 1 | 1 |
| w2 | 1 | 0 |

0.5

| L4 | o1 | o2 |
|---|---|---|
| w1 | 1 | 1 |
| w2 | 0 | 1 |

0.

| L5 | o1 | o2 |
|---|---|---|
| w1 | 0 | 1 |
| w2 | 1 | 1 |

0.

| L6 | o1 | o2 |
|---|---|---|
| w1 | 0 | 1 |
| w2 | 1 | 0 |

1.

| L7 | o1 | o2 |
|---|---|---|
| w1 | 1 | 0 |
| w2 | 0 | 1 |

- We then apply Bayes theorem and get the (unnormalized) posterior:

| L1 | L2 | L3 | L4 | L5 | L6 | L7 |
|---|---|---|---|---|---|---|
| 0.1 * 0.5 | 0.1 * 1 | 0.1 * 0.5 | 0.1 * 0.5 | 0.1 * 0. | 0.25 * 0 | 0.25 * 1. |

# A simple example

- We get posterior:

| L1 | L2 | L3 | L4 | L5 | L6 | L7 |
|---|---|---|---|---|---|---|
| 0.1 | 0.2 | 0.1 | 0.1 | 0. | 0. | 0.5 |

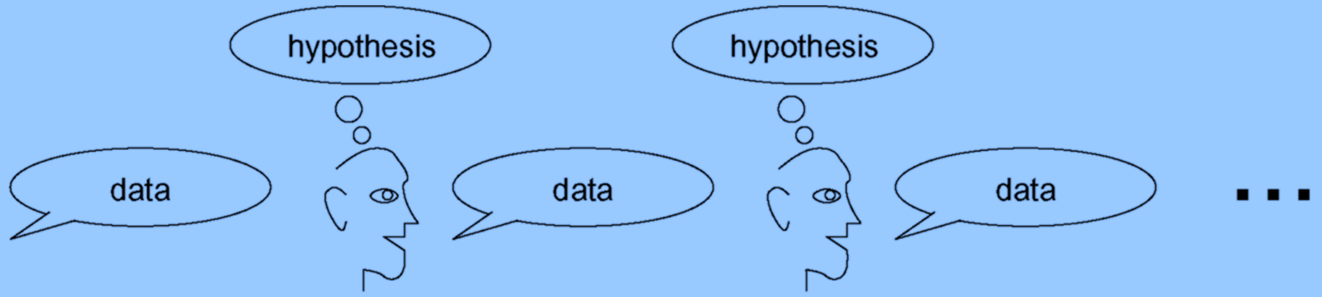
- Even though the likelihood of (o1, w1) is the same for L2 and L7, because of the prior L7 has higher posterior probability.
- Last step: select a language based on the posterior. Two options:
  - They can sample a language from the posterior
  - Or select the language with the highest posterior probability (MAP)
- In this case, they might sample e.g., L2 or take the MAP L7

# Bayesian IL & stationary distribution

- What if we iterate this process?
- This can be thought of as running a Markov chain on the space of 7 languages
  - Where the transition probability x → y is the prob of a learner learning y from a certain number of datapoints produced by true language x
- Question: What is the *stationary distribution* of this chain?
  - I.e. what will be the distribution over languages eventually?
- Surprising answer (assuming sample agents):
  - It doesn't depend on the number of datapoints
  - It doesn't depend on starting language
  - **It's just the prior**! (Griffiths & Kalish 2007)

# Convergence to the prior

Intuition:



- IL
- + Bayesian agents
- + Sampling agents
- Is a type of Gibbs sampling

$$d_0 \xrightarrow{P_{LA}(h|d)} h_1 \xrightarrow{P_{PA}(d|h)} d_1 \xrightarrow{P_{LA}(h|d)} h_2 \xrightarrow{P_{PA}(d|h)} d_2 \xrightarrow{P_{LA}(h|d)} \ldots$$

$$h_1 \xrightarrow{\sum_d P_{PA}(d|h)P_{LA}(h|d)} h_2 \xrightarrow{\sum_d P_{PA}(d|h)P_{LA}(h|d)} \ldots$$

$$d_0 \xrightarrow{\sum_h P_{LA}(h|d)P_{PA}(d|h)} d_1 \xrightarrow{\sum_h P_{LA}(h|d)P_{PA}(d|h)} d_2 \longrightarrow \ldots$$

$$h_1, d_1 \xrightarrow{P_{LA}(h|d)P_{PA}(d|h)} h_2, d_2 \xrightarrow{P_{LA}(h|d)P_{PA}(d|h)} \ldots$$

# Bayesian IL: A temporary conclusion

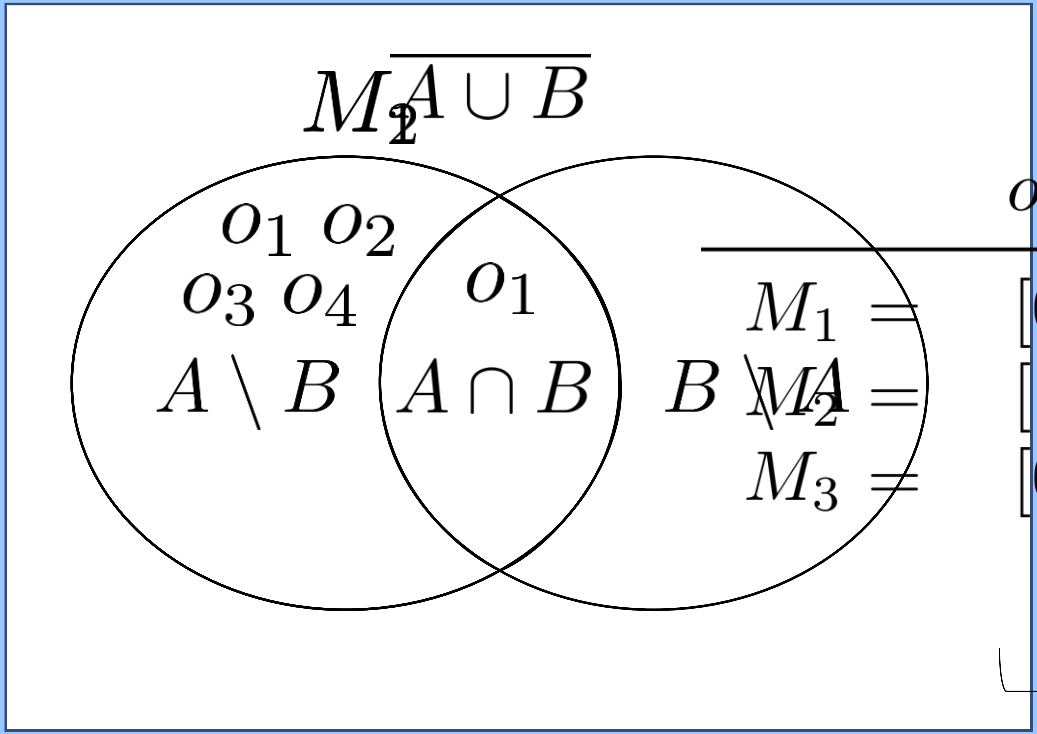
- Temporary conclusion:
  - IL+Bayesian learners alone is somewhat boring
  - …always prior = stationary distribution.
- Two ways to make it interesting:
  - Use not-sampling agents, e.g., MAP or maximum-likelihood agents
    - Hard to study mathematically
  - Combine with other pressures, e.g., communicative accuracy
    - Last lecture!
- Point: IL *reveals* cognitive biases, but Bayesian inference builds them in
- But we don’t know the prior biases of ANNs!

# Neural Iterated Learning

# The evolution of monotonicity

- In the first lecture, we looked at monotonicity as a universal of the meaning of simple determiners.
- Yesterday, we saw that ANNs can learn some monotonic quantifiers faster than non-monotonic quantifiers.
- However, *some* non-monotonic quantifiers might still be easier than *some* monotonic ones.
- In Carcassi, Steinert-Threlkeld, & Szymanik (2021), we look at an IL model.
  - We implicitly search a much larger space of quantifiers.
  - We show the *evolution* of quantifier meaning

# The evolution of monotonicity



|  | $o_1$ | $o_2$ | $o_3$ | $o_4$ | $Q(A, B \cap A)$ |
|---|---|---|---|---|---|
| $M_1 =$ | $[0$ | $0$ | $0$ | $0]$ | $1$ |
| $M_2 =$ | $[1$ | $0$ | $0$ | $0]$ | $0$ |
| $M_3 =$ | $[0$ | $1$ | $1$ | $0]$ | $0$ |

$\vdots$

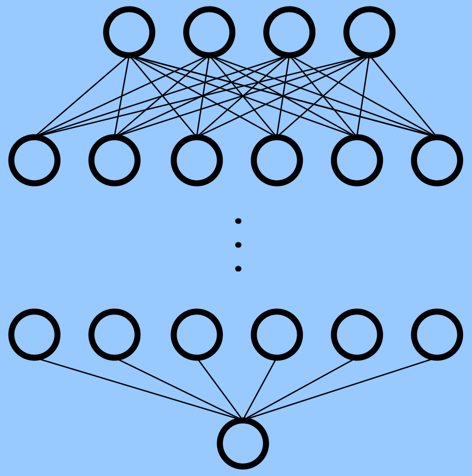
Input

Output

# The evolution of monotonicity

## Cultural parent

$[0, 1, 1, 0]$



$0.32$

## Data

$[0, 0, 1, 0] \quad 0.0$

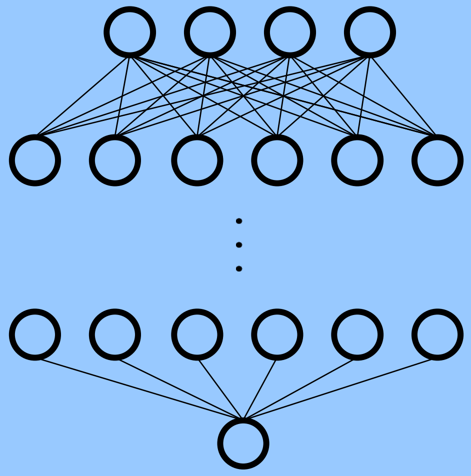
$[1, 0, 1, 0] \quad 1.0$

$[0, 1, 1, 0] \quad 1.0$

$\vdots$

Bottleneck size

## Cultural child

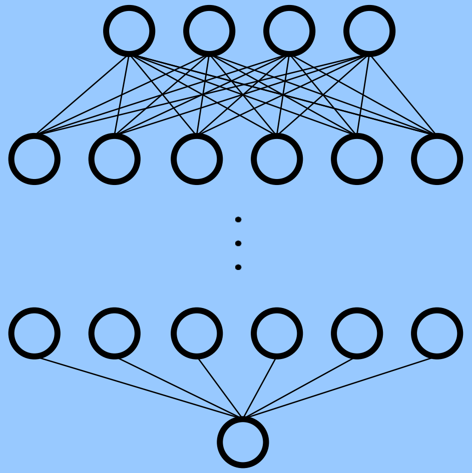


# The evolution of monotonicity

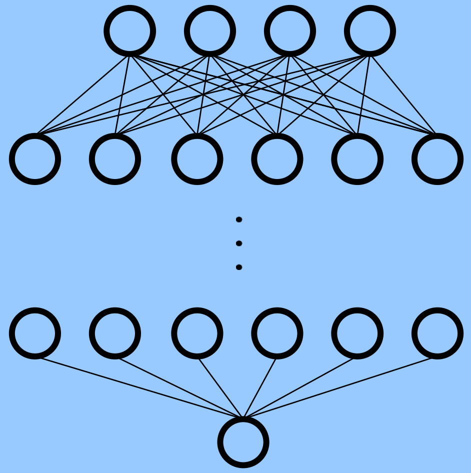
Cultural parent

Data

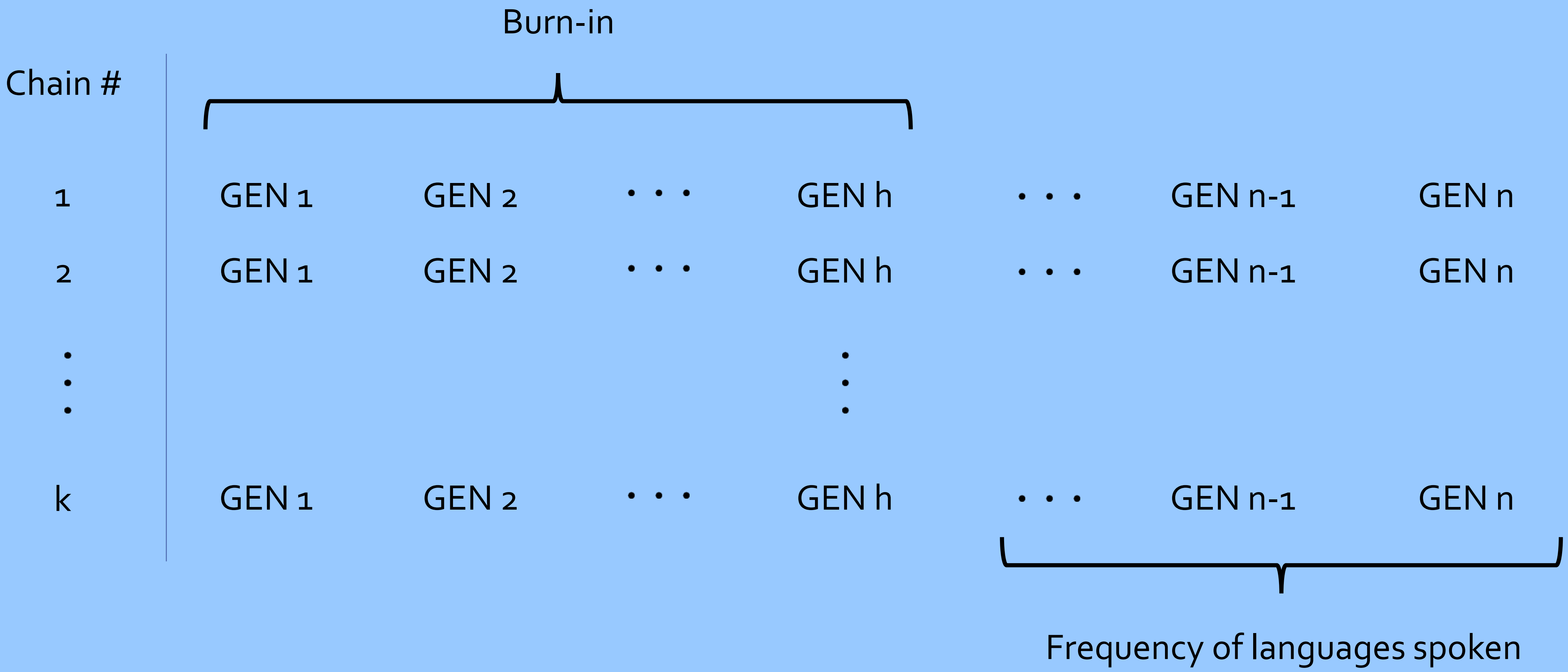
Cultural child



$[0, 0, 1, 0] \quad 0.0$



# The evolution of monotonicity

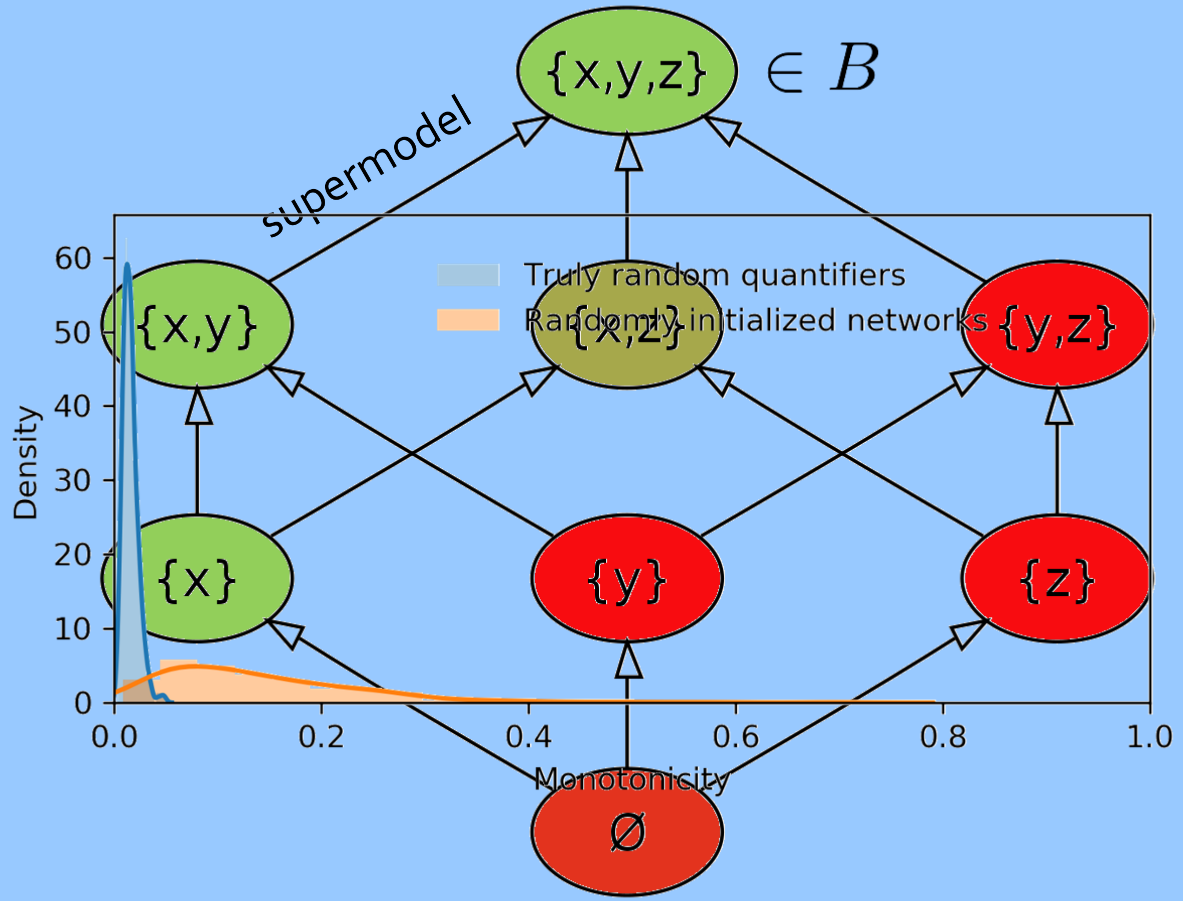


Burn-in

| Chain # | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | GEN 1 | GEN 2 | • • • | GEN h | • • • | GEN n-1 | GEN n |
| 2 | GEN 1 | GEN 2 | • • • | GEN h | • • • | GEN n-1 | GEN n |
| ⋮ | | | | ⋮ | | | |
| k | GEN 1 | GEN 2 | • • • | GEN h | • • • | GEN n-1 | GEN n |

Frequency of languages spoken

# The evolution of monotonicity



$M$ is a random model

$$1_Q = Q(M) = \text{round}(NN(M))$$

$$H(1_Q)$$

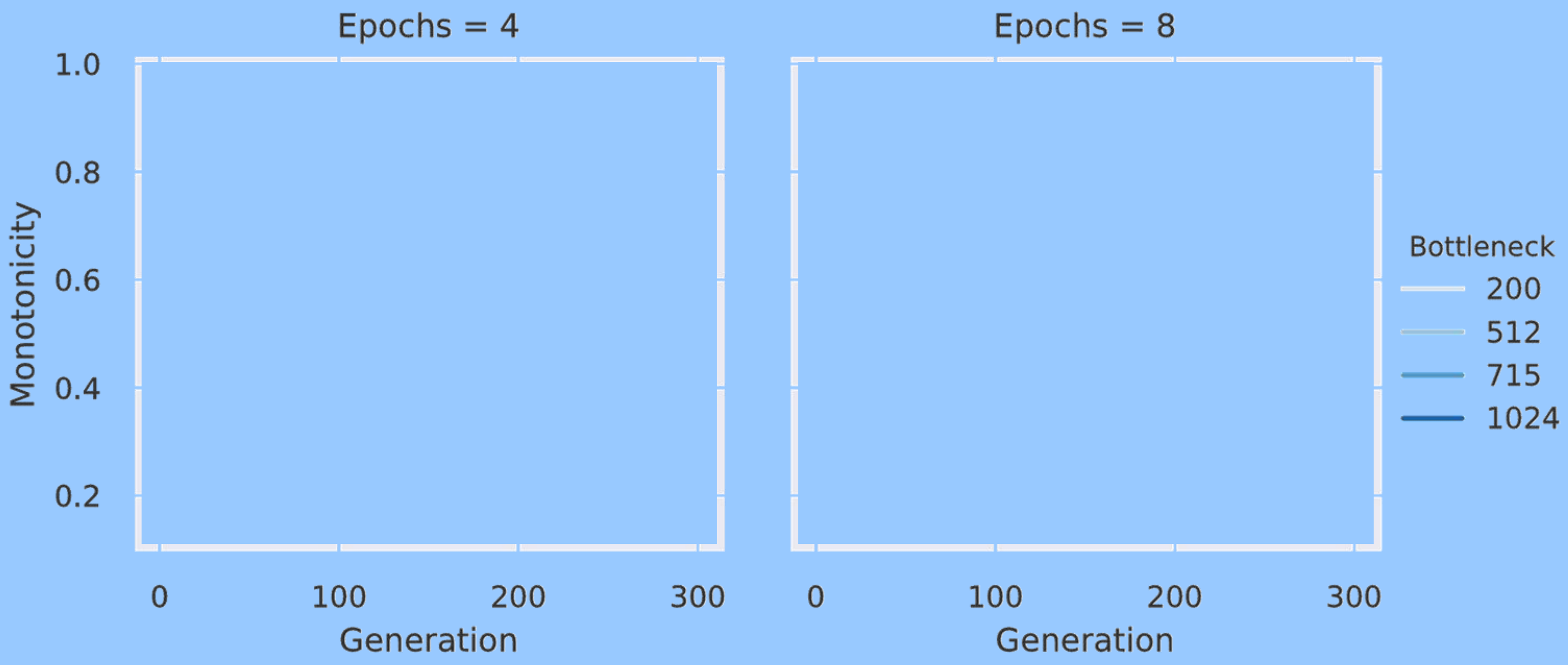
$1_Q^- =$ a submodel of $M$ is true

$$H(1_Q | 1_Q^-)$$

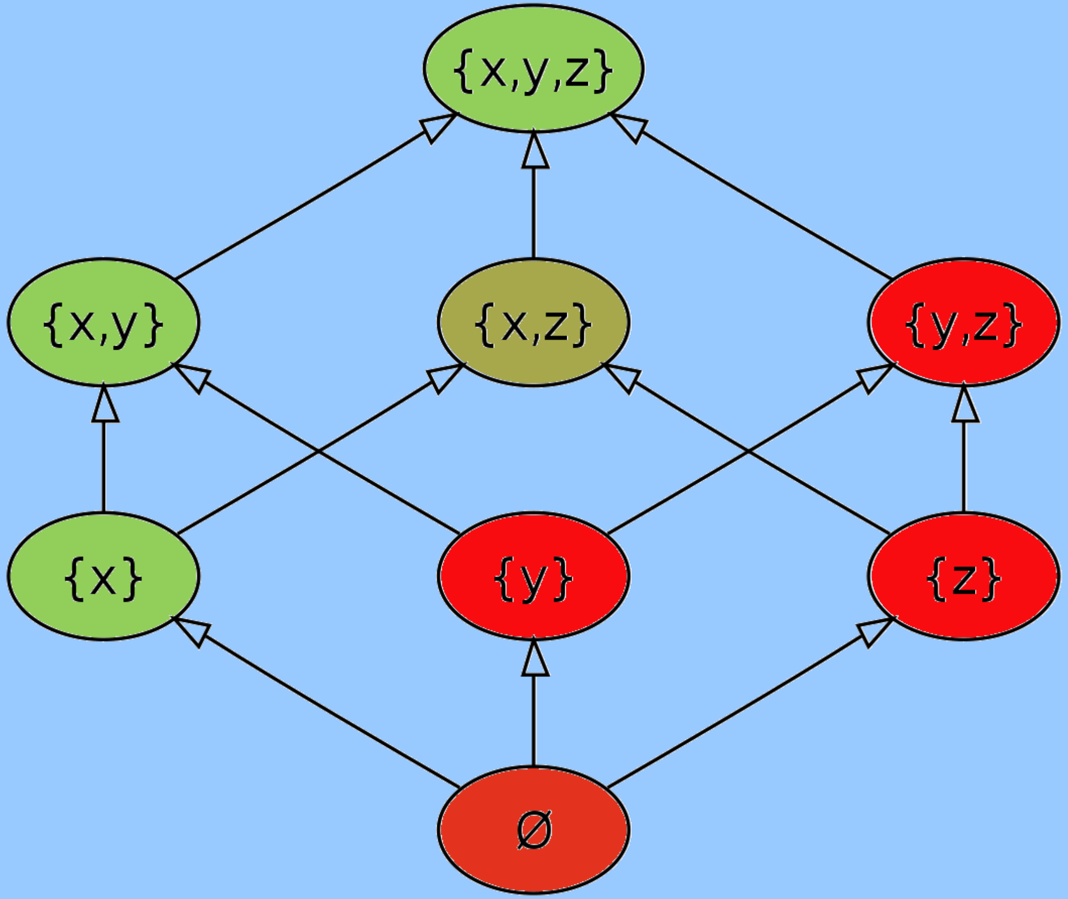
$\frac{H(1_Q|1_Q^-)}{H(1_Q)}$: prop of $H(1_Q)$ left given $1_Q^-$.

$$\text{mon}(Q) := 1 - \frac{H(1_Q \mid 1_Q^-)}{H(1_Q)}$$

# The evolution of monotonicity



# The evolution of monotonicity



$$\exists a \text{ s.t. } \begin{cases} Q(x) = 1 & \text{a} \in \text{x} \\ Q(x) = 0 & \text{otherwise} \end{cases}$$

Proper-noun-like quantifiers evolve in the first model because neural networks find it easy to exploit the identity of individual objects.

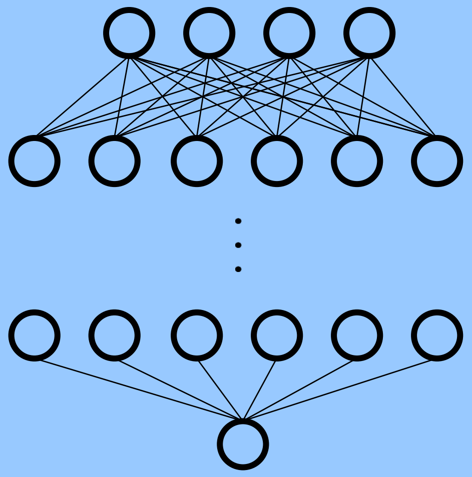
# The evolution of monotonicity

Cultural parent

Data

Cultural child

$[0, 1, 1, 0]$

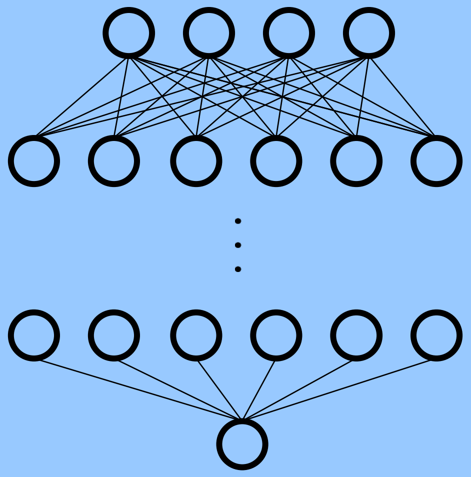


$[0, 0, 1, 0]$ $0.0$

$[1, 0, 1, 0]$ $1.0$

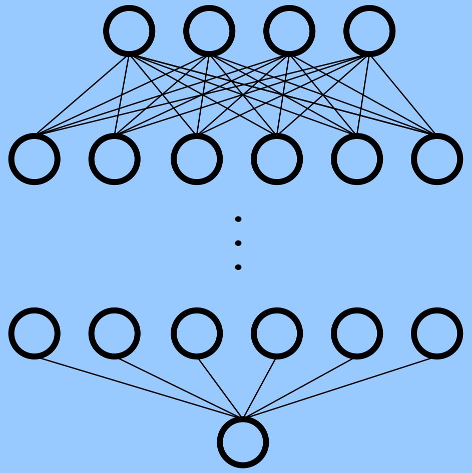
$[0, 1, 1, 0]$ $1.0$

$\vdots$



# The evolution of monotonicity

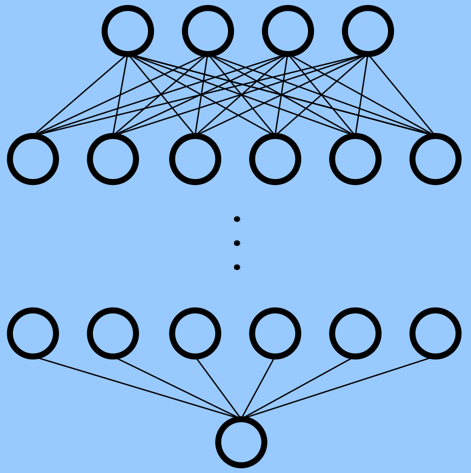
Cultural parent



Data

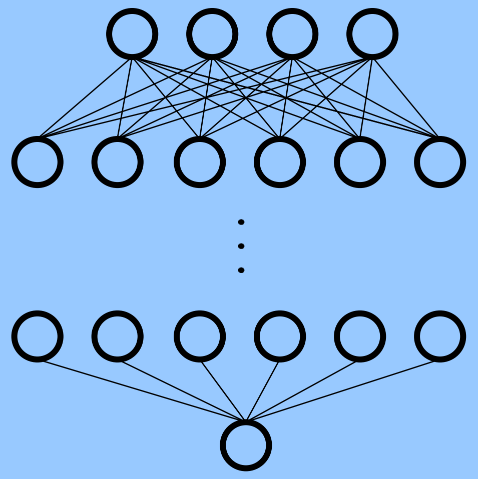
| | |
|---|---|
| $[1, 0, 1, 0]$ | $0.0$ |
| $[1, 0, 0, 0]$ | $0.0$ |
| $[0, 0, 1, 1]$ | $1.0$ |
| $[1, 1, 0, 0]$ | $1.0$ |
| $\vdots$ | |

Cultural child



# The evolution of monotonicity

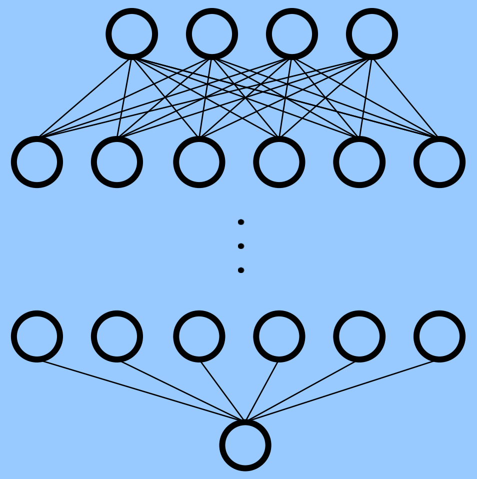
Cultural parent



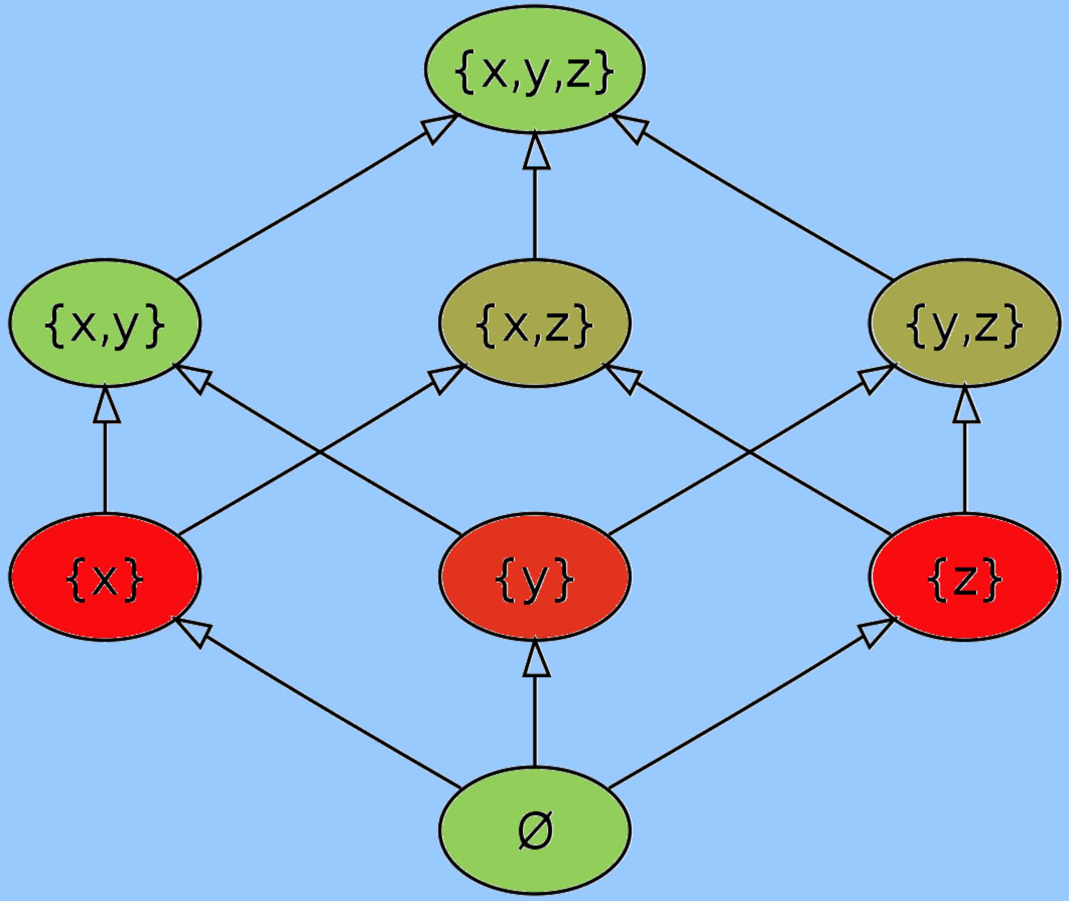
Data

$$[1, 0, 0, 0] \quad 0.0$$

Cultural child



# The evolution of monotonicity

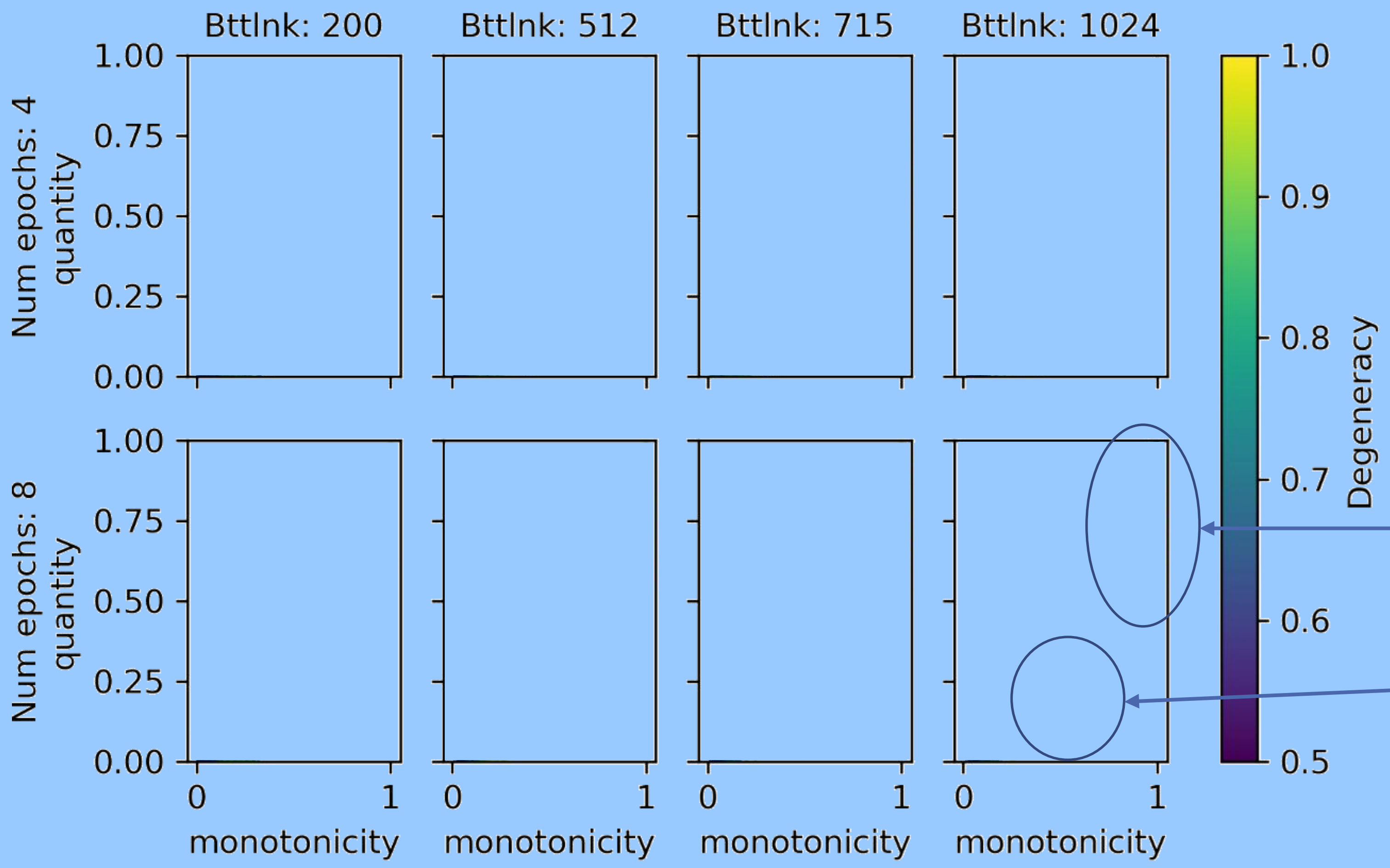


The only quantifiers that are robust across the permutations of the string are the *quantitative* quantifiers.

$\# = \text{size of } A \cap B \text{ in } M$

$H(1_Q|\#)$ $\dfrac{H(1_Q|\#)}{H(1_Q)}$

# The evolution of monotonicity

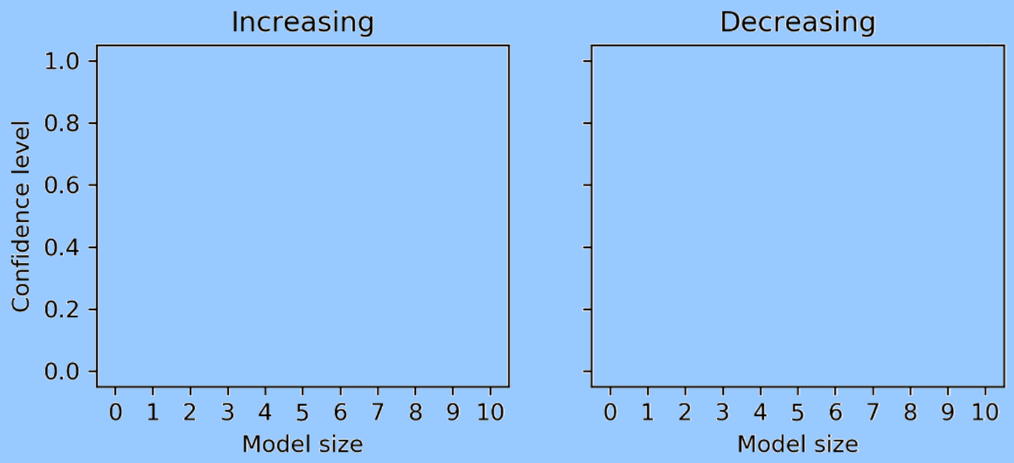


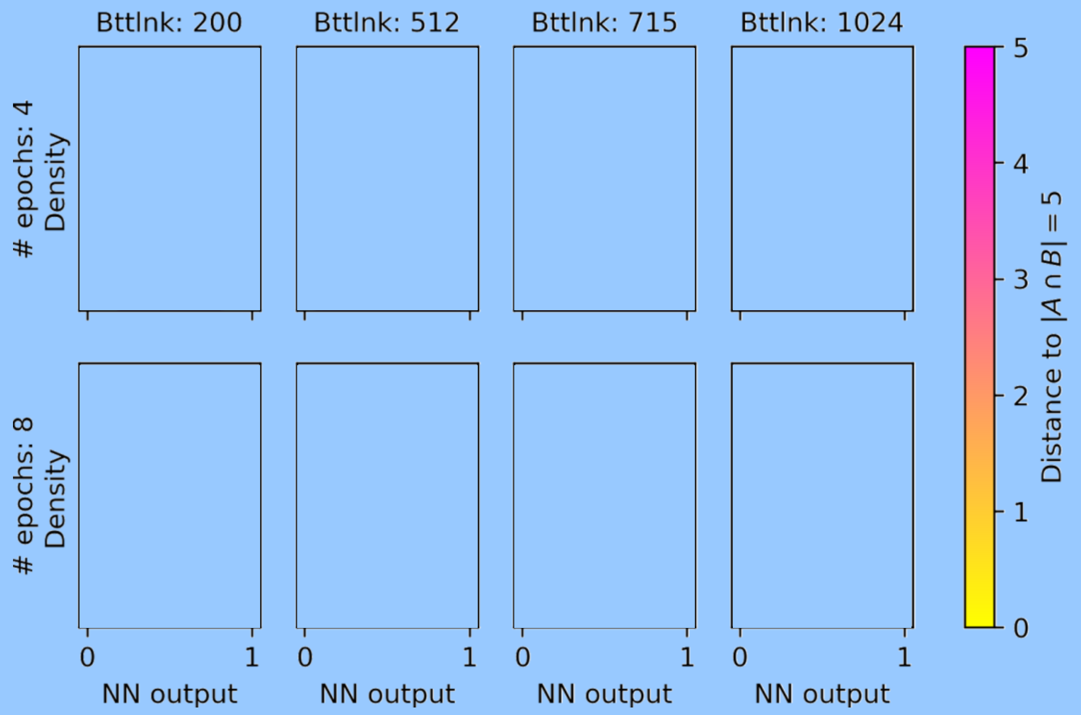
Vague threshold quantifiers

Nearly degenerate quantifiers

# The evolution of monotonicity

- By "threshold quantifier" we mean that the average confidence in its truth is a monotonic function of the model size.
- This is not simply a side effect of the fact that there are more models with middle number of ones.





# Summary

- Iterated Learning model as a way of solving the linkage problem
- IL requires a model of learning, two natural options: Bayes & ANNs
- With sampling Bayesian learners, IL converges to the prior
    - We'll come back to IL on Friday
- With neural learners, we can use IL to reveal biases
    - We used this to reveal the IL preference for monotonicity
    - And for quantity!
- Questions?