Why think step by step? Reasoning emerges from the locality of experience

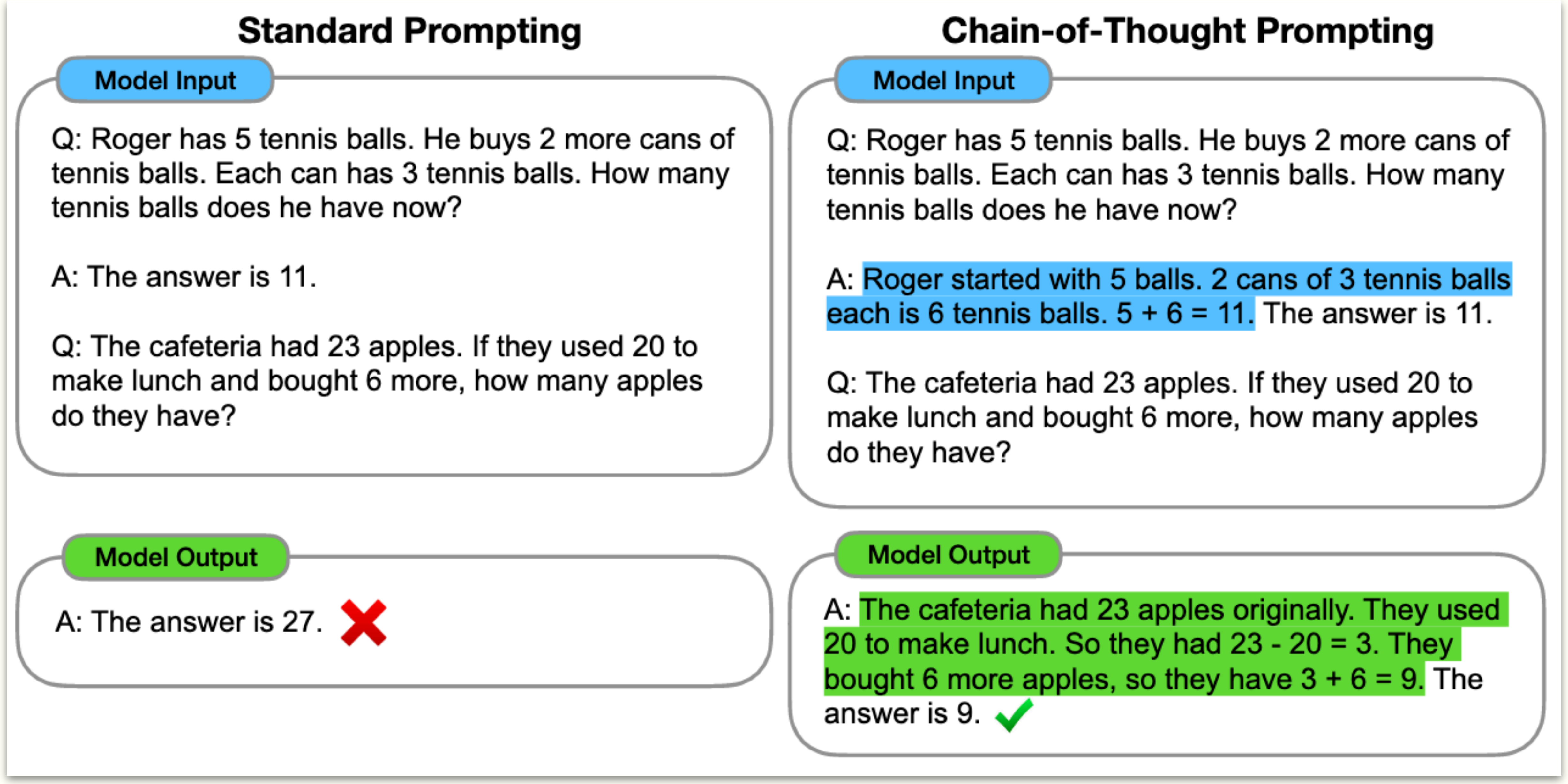
Ben Prystawski, Michael Y. Li, Noah D. Goodman

Why does reasoning work?

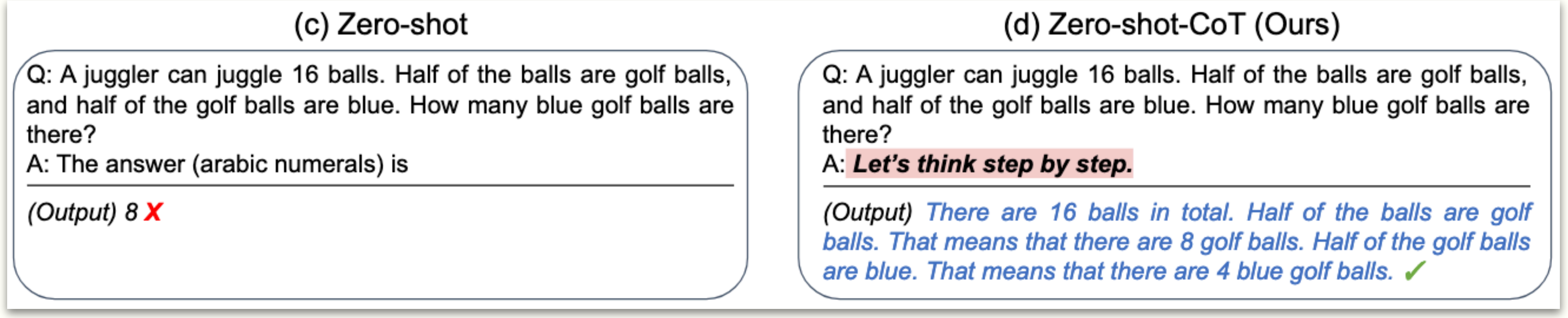
- We can enhance our inferences by working through a series of steps
- But that doesn't give us any new data
- So how does it make our inferences better?

Let's think step by step

- We can get language models to do better on lots of tasks by "chain-of-thought" prompting



(Wei et al., 2022)



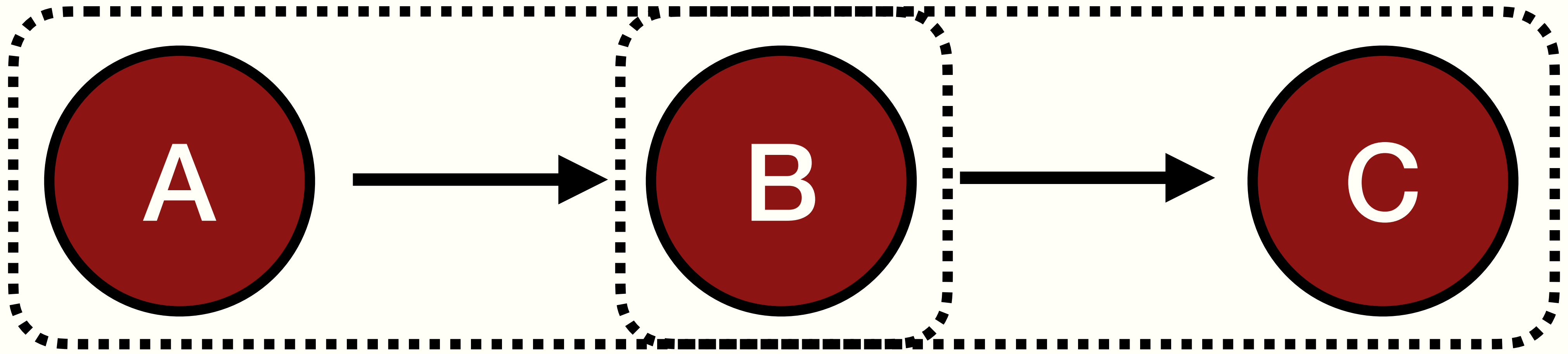
(Kojima et al., 2022)

The "step to rationality"

- Shepard (2008): thought experiments let us apply internalized intuitive knowledge of principles and symmetries
- Do heavier objects fall faster?

Our hypothesis

Step-by-step reasoning lets (humans/LMs) chain together local inferences between variables they have seen together a lot in order to support longer-distance inferences



$$P(C|A) = \sum_{B} P(C|B)P(B|A)$$

Probabilistic inference as language modeling

- We can just write a sample from a Bayes net as a string
- All variables are Boolean-valued

target: X3	target: X1	target: X3
X0=1	X2=1	X1=0
X1=0	X0=1	X2=1
X2=1	X3=1	X0=1
X3=1	X1=0	X3=1

Estimating using a trained model

The direct prediction estimator

target: X2

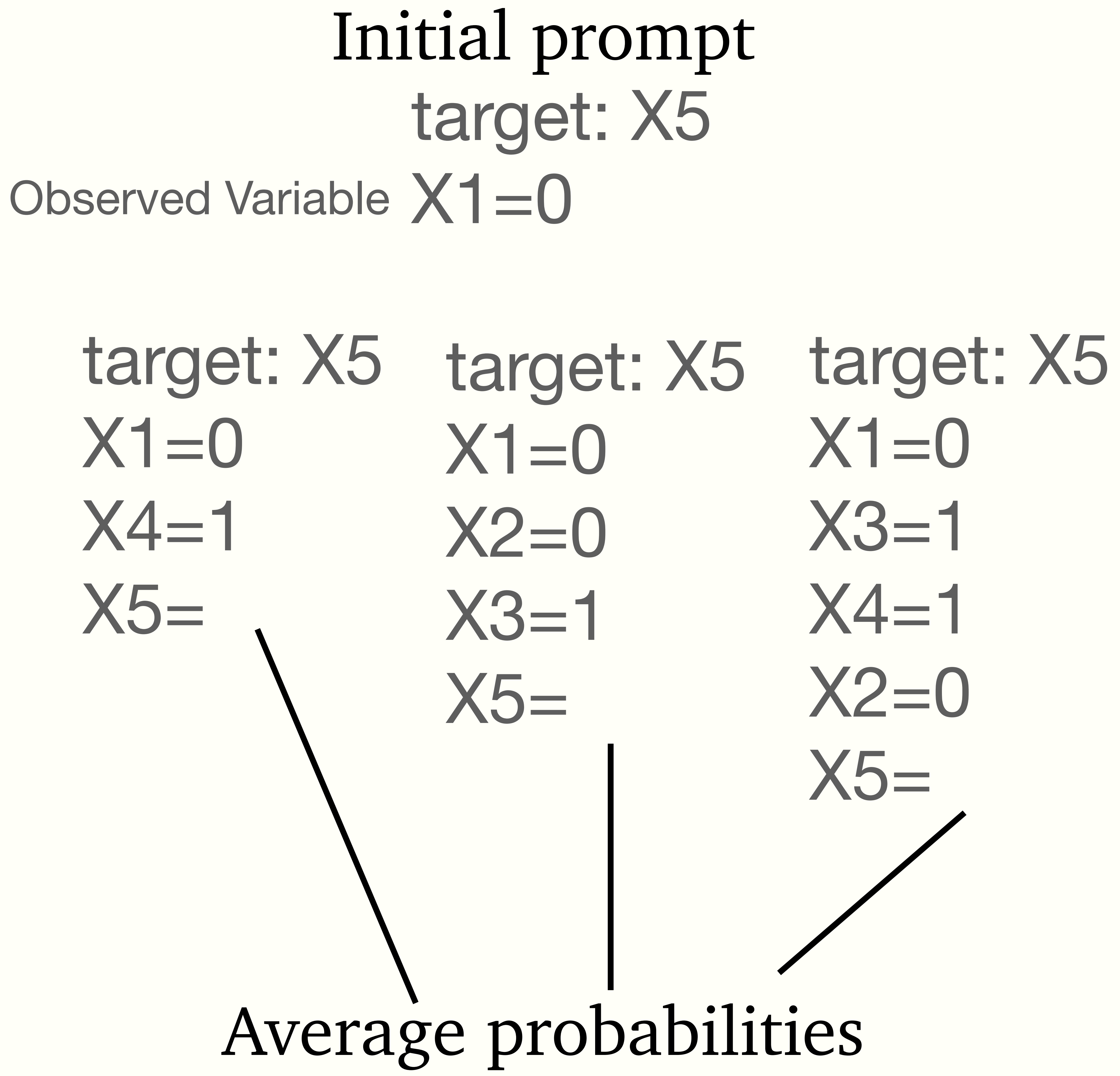
Observed Variable X1=0

Target Variable X2=

Get probabilities of 1 and 0 going here, then normalize

Reasoning as free generation

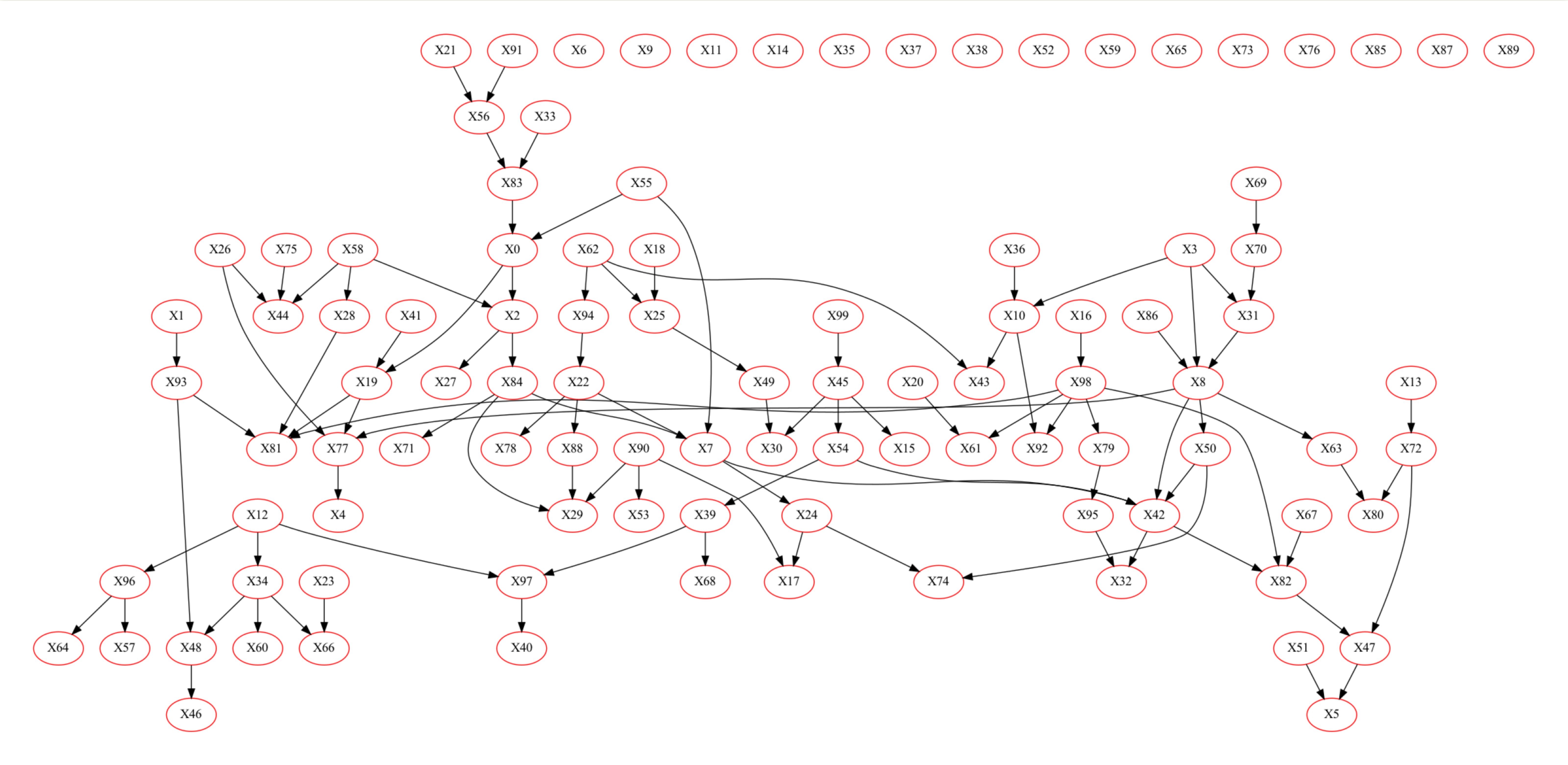
- Run the model forward, generating names and values for intermediate variables
- Compute the probability of the target variable when the model generates its name
- Resample intermediate variables / values 10 times, averaging probabilities



What training conditions lead free generation to outperform direct prediction?

Generating the training data

- We randomly generate Bayes nets
  - 100 nodes
  - 100 random edges
  - Conditional probability tables $\sim \text{Beta}(\frac{1}{5}, \frac{1}{5})$

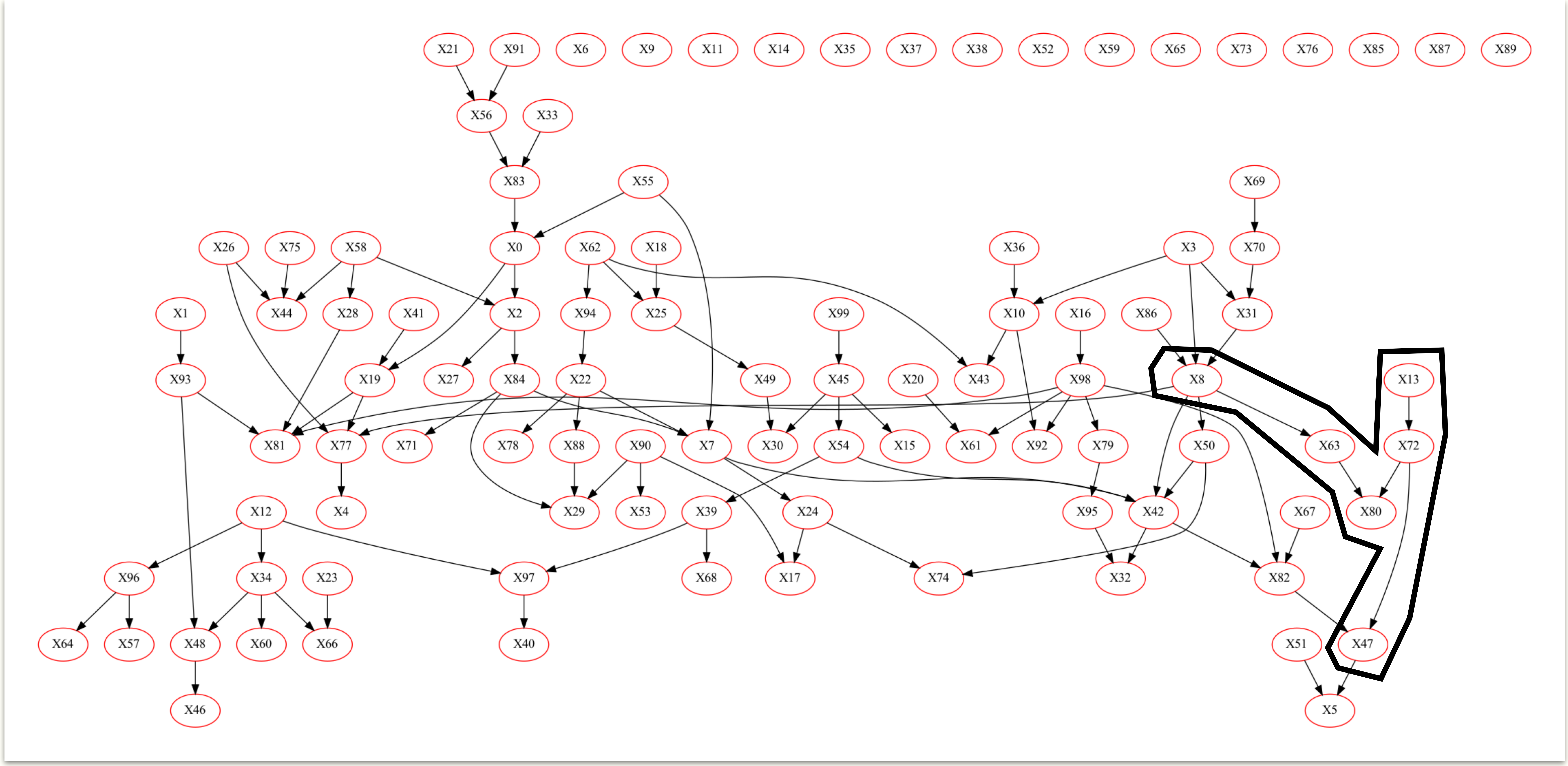


Holding out pairs

- We select pairs of variables to never co-occur with each other in training
    - Distance at least two
    - High mutual information
- All training conditions use the same set of held-out pairs
- Key metric: **how well can a trained language model infer conditional probabilities for held-out pairs? (MSE)**

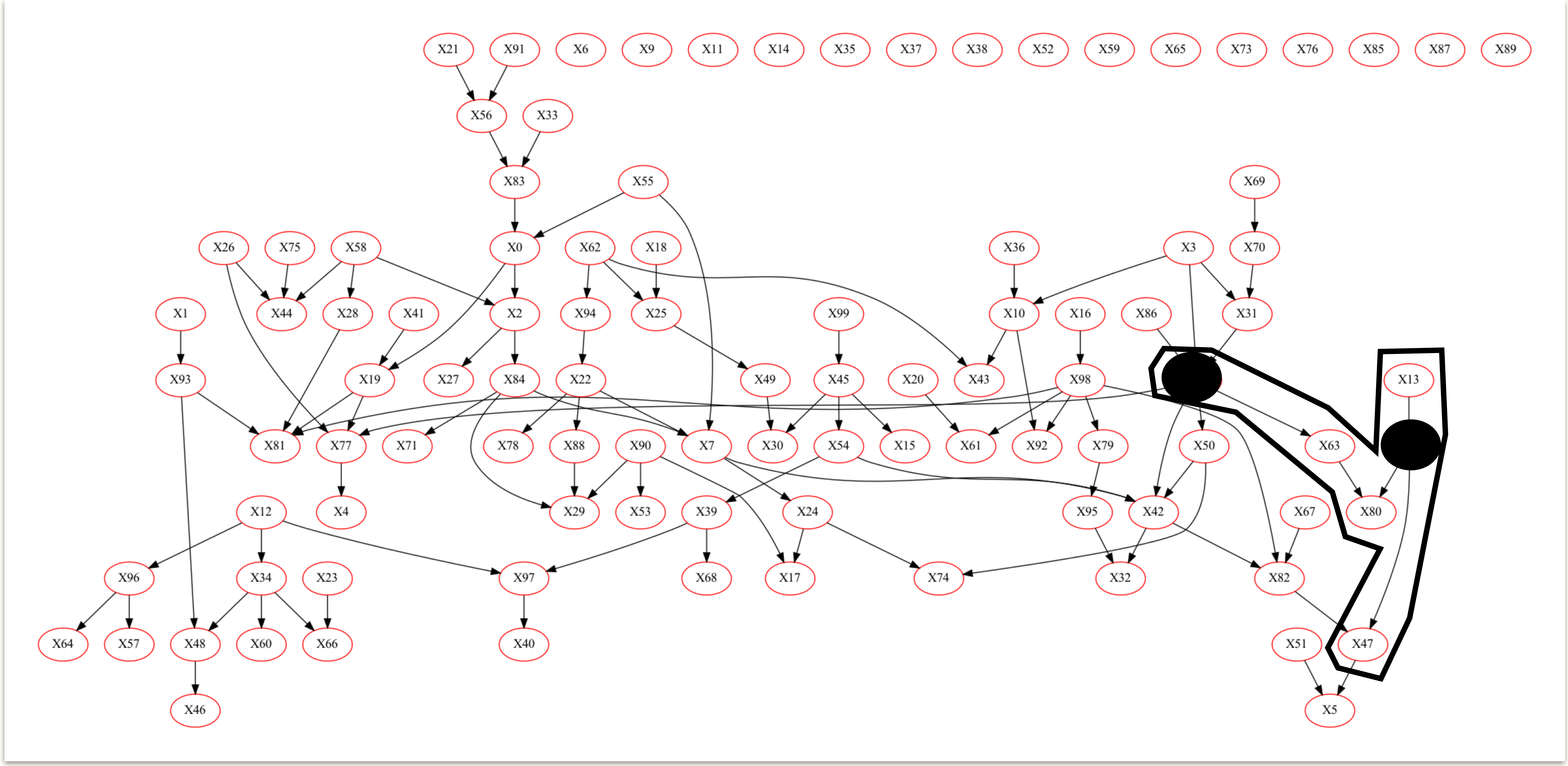
Training data - local neighborhoods

- Each training sample includes only a *local neighborhood* of size $k \sim \text{Geom}(0.5)$



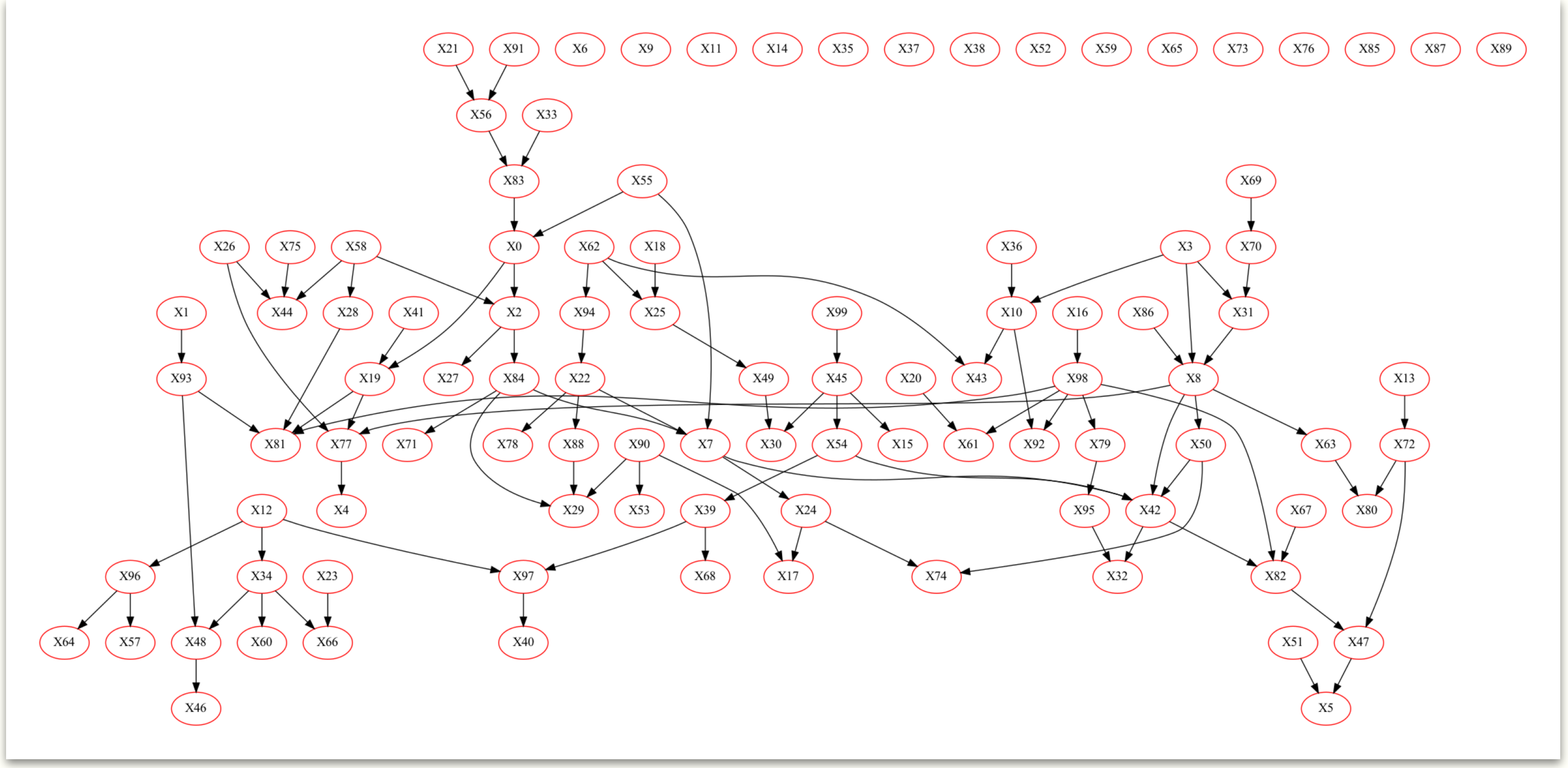
Training data - variable dropout

- We remove a random subset of the variables in the local neighborhood



Training

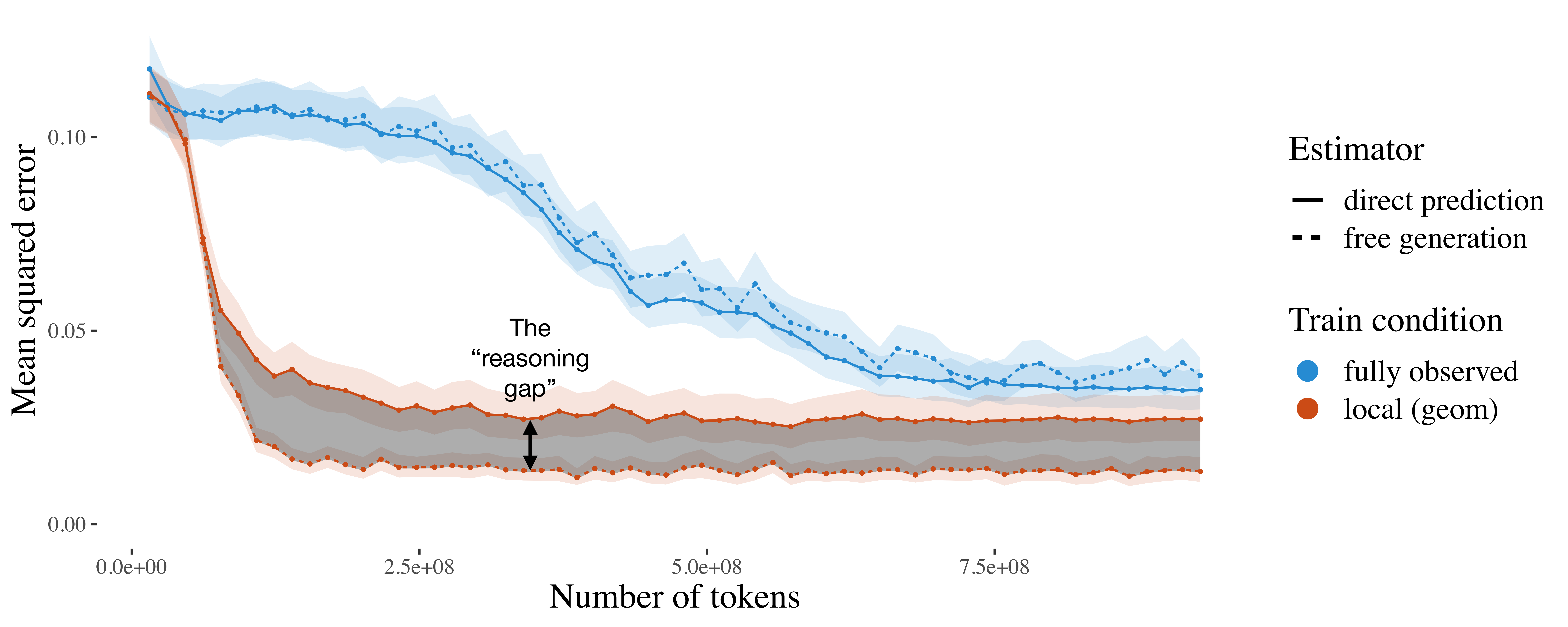
- Concatenate 1 million samples, showing only the selected variables
- 10 different Bayes nets, separate transformer for each
- 300k gradient steps



target: X63
X80=0
X13=1
X47=1
X63=0

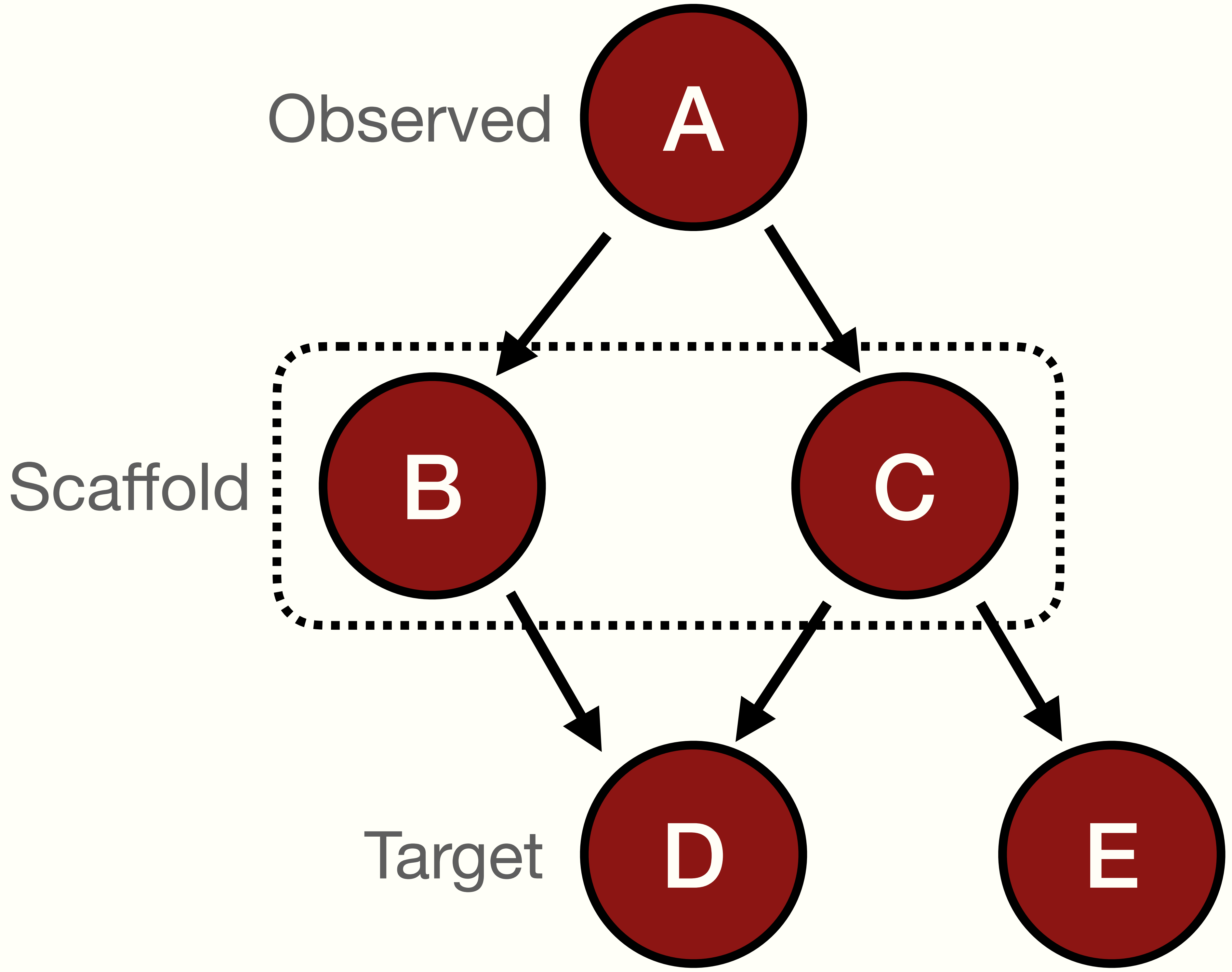
GPT-2

The reasoning gap emerges over training



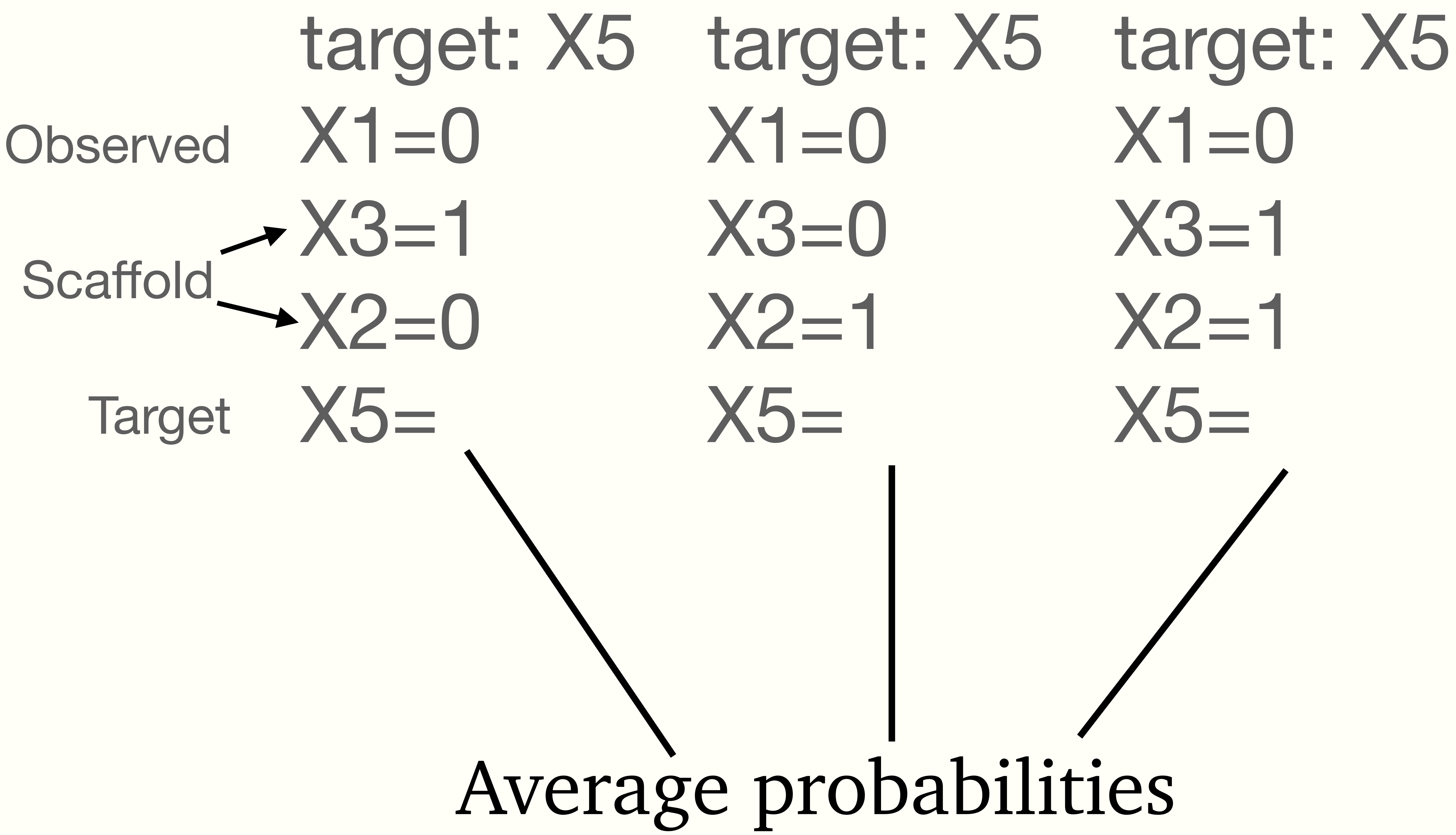
An alternative estimator: scaffolded generation

- We generate the smallest set of variables that *d*-separate the observed and target variables
  - Ordered from closest to the observed variable to farthest (in practice they're generally 1-2 variables)



Using scaffolds

- Sample the values of the scaffold variables, then get the target variable probability
- Resample 10 scaffolds



Controls

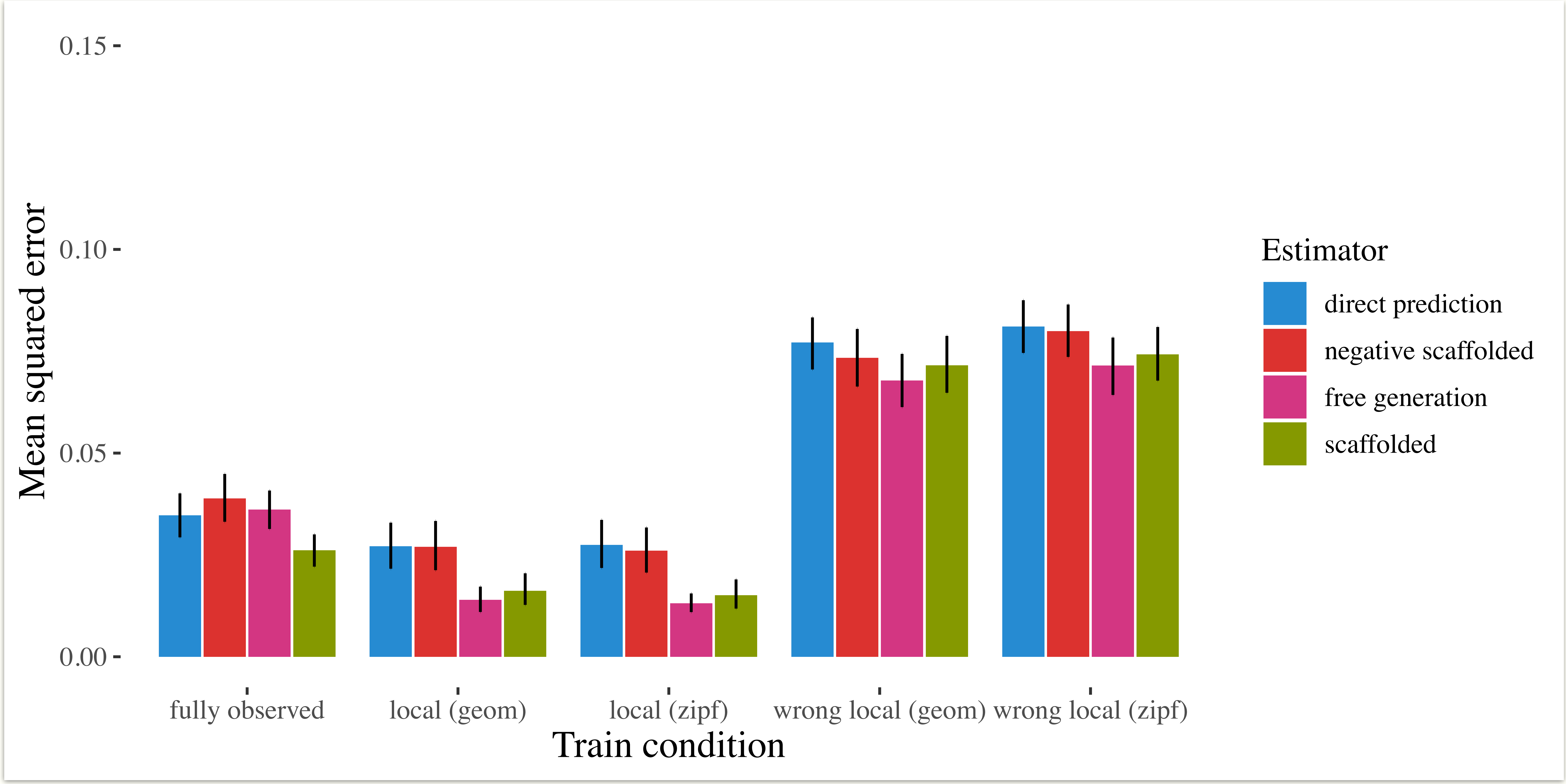
Training Conditions

- Fully-observed: complete samples from the Bayes net (except for held-out pairs)
- Wrong local neighborhood: local neighborhoods from a Bayes net other than the one the samples are drawn from

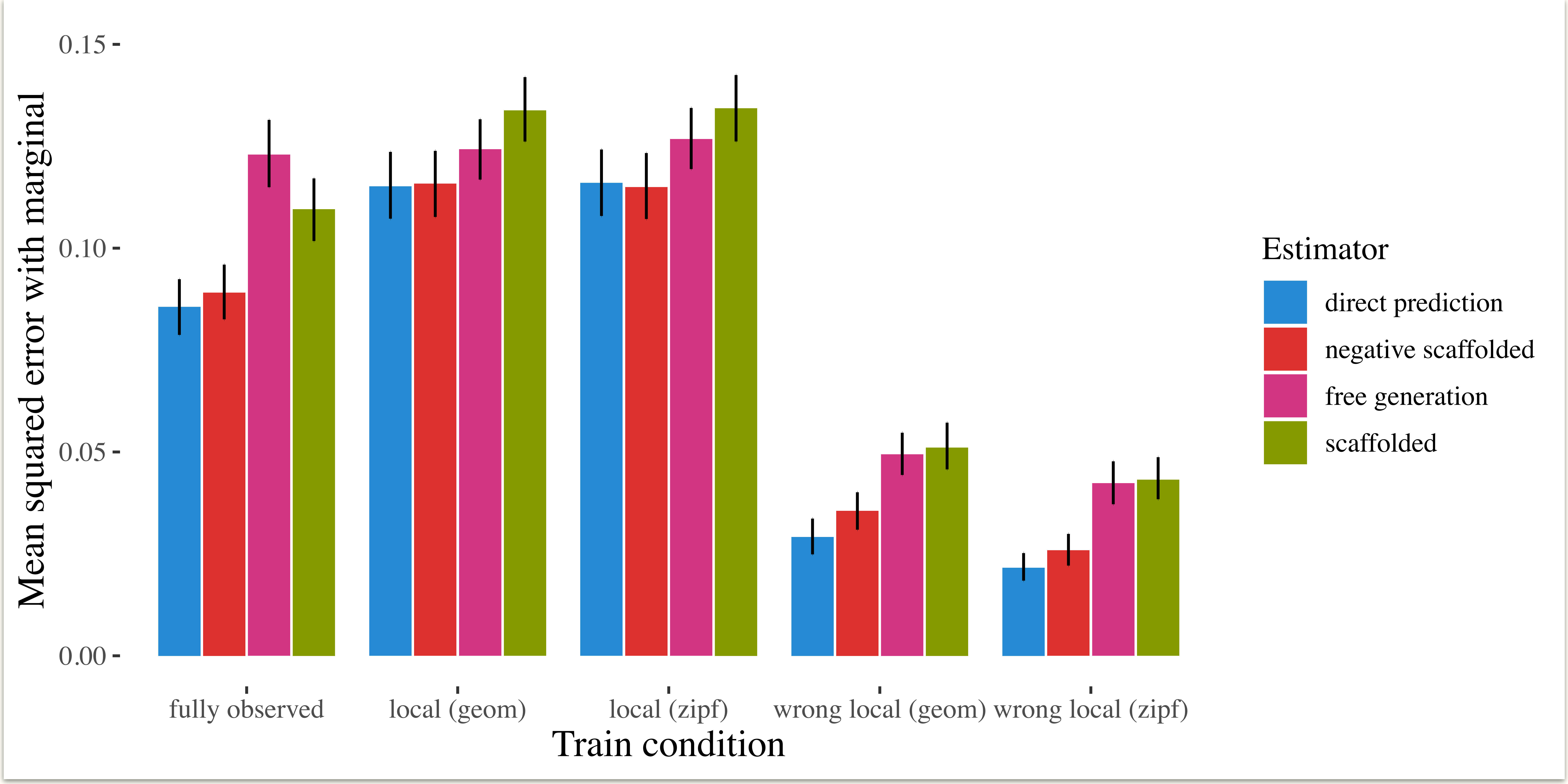
Estimators

- Negative scaffolded generation: reason through random variables that are not in the scaffold

Comparison across all training conditions /estimators



Falling back on the marginal



Conclusions

- Reasoning through intermediate variables improves estimation when data is structured locally
- Locally structured data + reasoning might help explain the gap in data efficiency between humans and machine learning models

Acknowledgements

Computation & Cognition Lab

Stanford | NLP

Questions?

benpry@stanford.edu

Paper: