Spinning Up with

BAYESIAN INFERENCE FOR NEURAL NETWORKS

Piotr Januszewski, 20.07.20

Spinning Up with

BAYESIAN INFERENCE FOR NEURAL NETWORKS

Introducing Bayesian Inference

•

Bayesian perspective on Neural Networks training

•

Variational Inference to the rescue!

•

Variational Bayes in code 🥩

INTRODUCING BAYESIAN INFERENCE

$$P(H \mid E) = \frac{P(E \mid H)P(H)}{P(E)}$$

Bayes' Theorem

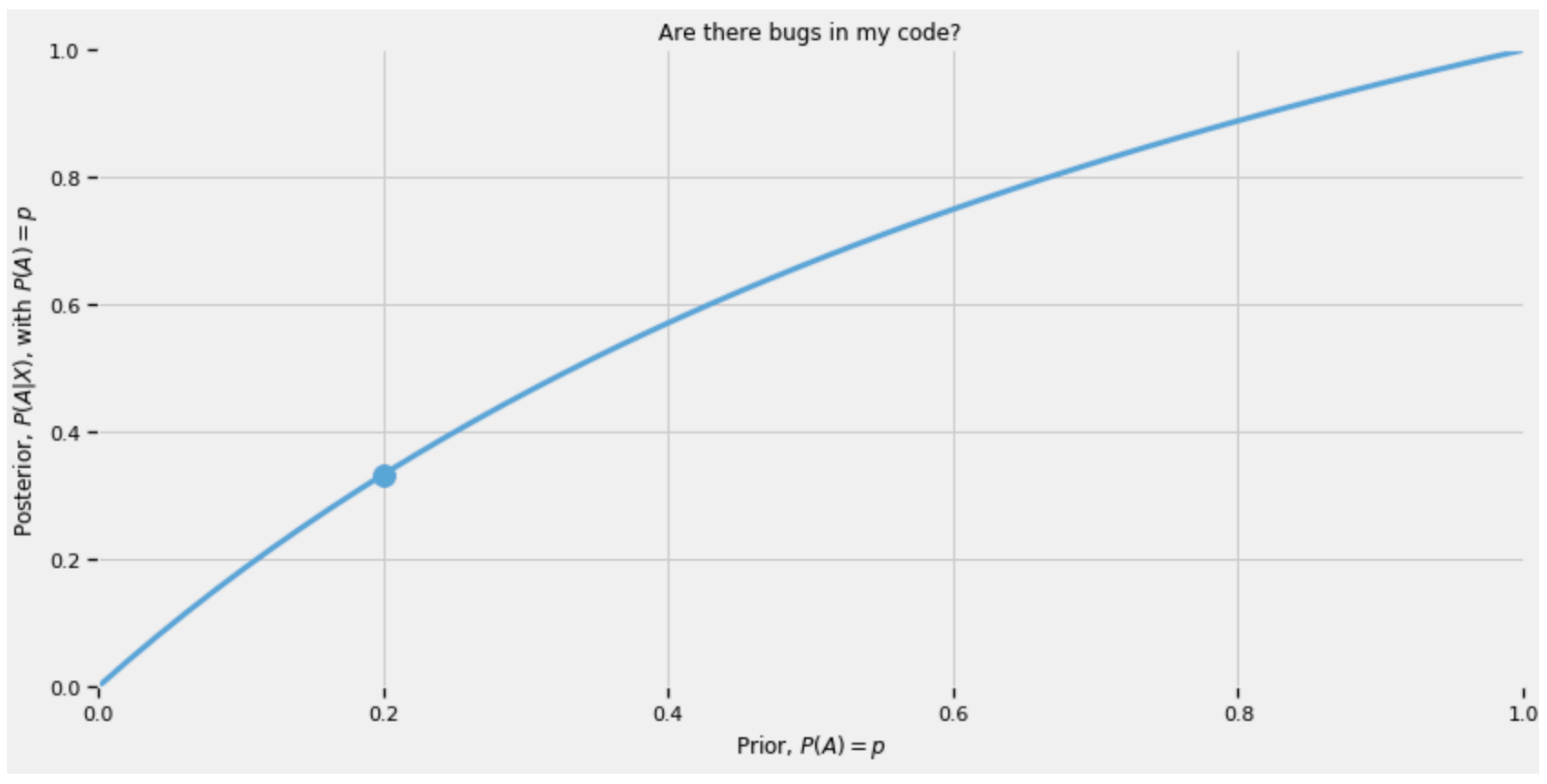
ABC EXAMPLE

- **I want to know if my code is bug-free, an event F, after I run tests that pass, an event T.**
- **I update, infer a posteriori, my believe that the code is bug-free in the face of new evidence, tests run.**
- $P(F) = x$ **is a prior believe in my code that it's bug-free. I parametrise it with** $x \in [0,1]$**.**
- $P(T|F) = 1$ **is a likelihood that the code is bug-free, given tests pass. It's one, "of course".**
- $P(T|\neg F) = 1/2$ **is a likelihood that the code has bugs, given tests still pass. Let's assume it's 50:50.**
- **Moreover, we will need a normalisation constant** $P(T) = P(T|F)P(F) + P(T|\neg F)P(\neg F)$**.**

$$P(F|T) = \frac{P(T|F)P(F)}{P(T|F)P(F) + P(T|\neg F)P(\neg F)}$$

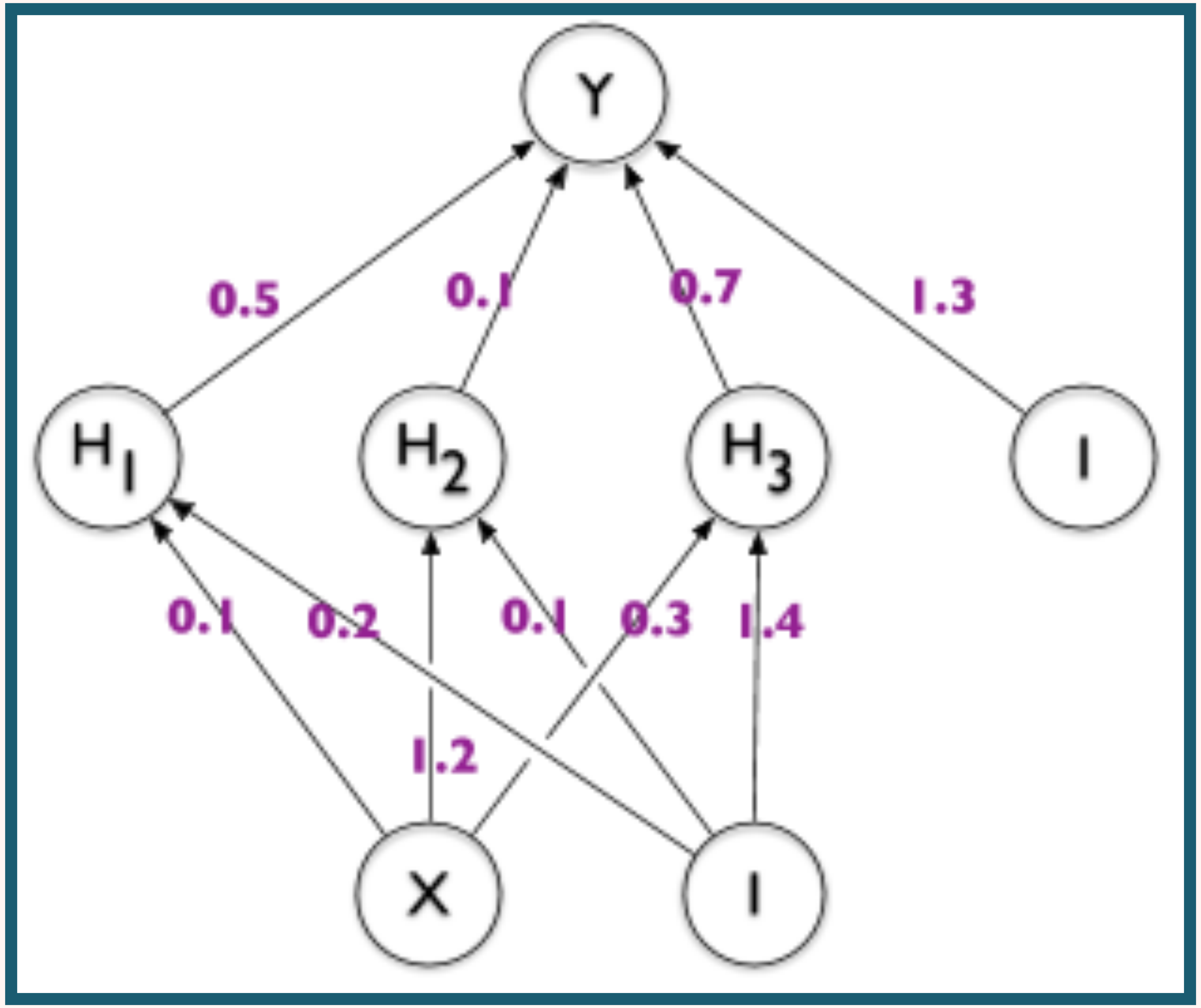
$$P(F|T) = \frac{x}{x + \frac{1}{2}(1-x)}$$

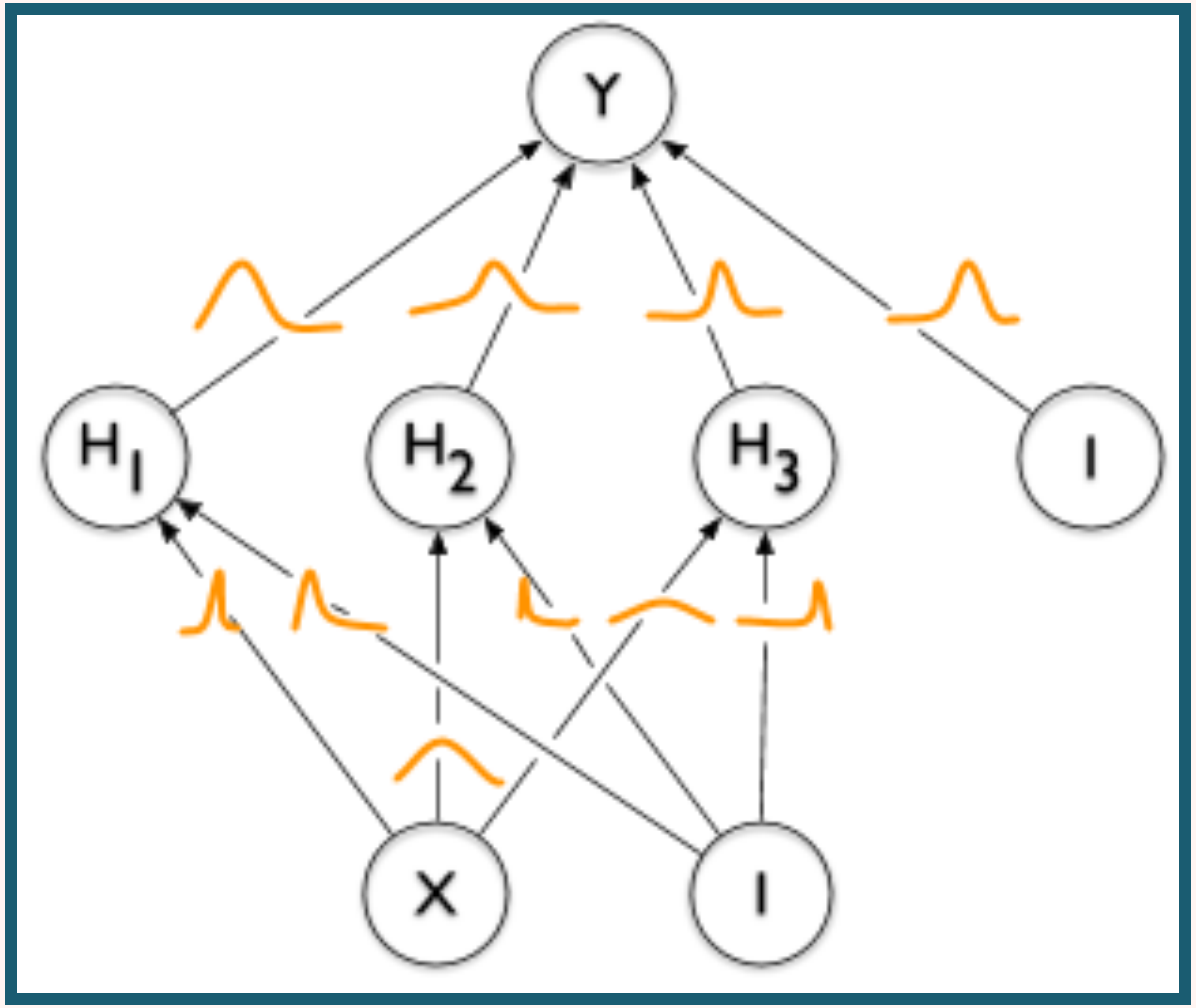
$$P(F|T) = \frac{2x}{1+x}$$



Probabilistic Programming & Bayesian Methods for Hackers

BAYESIAN PERSPECTIVE ON NEURAL NETWORKS TRAINING





The very Basics of Bayesian Neural Networks

BAYESIAN NEURAL NETWORK

- **Dataset $\mathscr{D}$ consisting of predictors X, e.g. images, and labels Y, e.g. classes.**
- **Likelihood $P(\mathscr{D} \mid \theta)$ or $P(Y \mid X, \theta)$ represented with a categorical softmax distribution on logits calculated by a Neural Network parametrised with θ, e.g. MLP.**
- **Prior on Neural Network parameters $P(\theta)$ represented with a Normal distribution.**
  - **It encodes our prior (lack of) knowledge what the parameter values could be. However, we suspect these are some small values around zero.**
- **Posterior $P(\theta \mid \mathscr{D})$ is... well... some distribution too, of course.**
  - **We will calculate it using the Bayes' Theorem!... or we would hope to do so.**

$$P(\theta \mid \mathscr{D}) = \frac{P(\mathscr{D} \mid \theta)P(\theta)}{\int P(\mathscr{D} \mid \theta)P(\theta)d\theta}$$

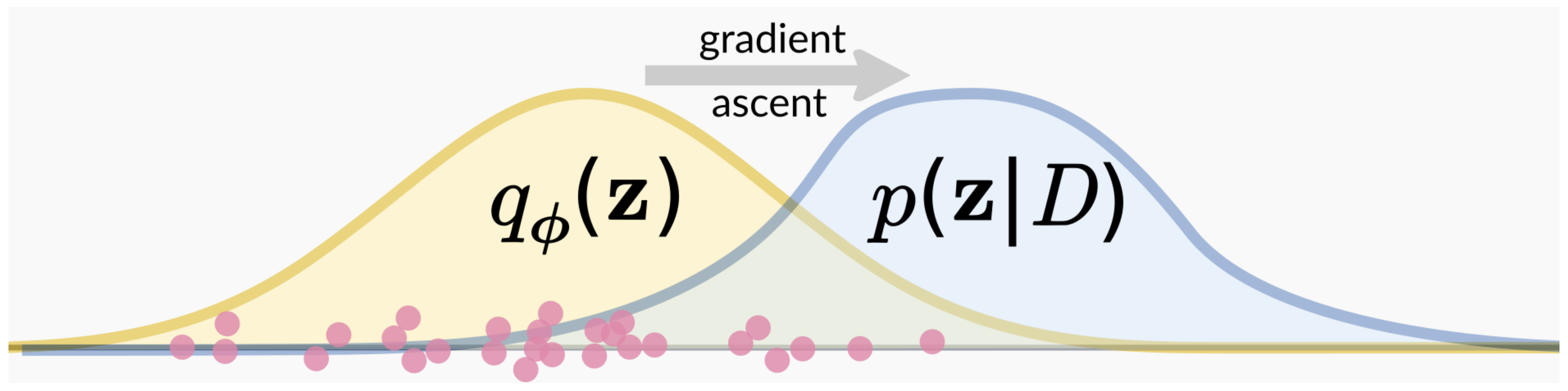
This integral is intractable :/

VARIATIONAL INFERENCE TO THE RESCUE!

CAN'T CALCULATE?
THEN APPROXIMATE!

$$D_{\mathrm{KL}}\left(Q(\theta)\;\|\;P(\theta\,|\,\mathscr{D})\right)$$

VARIATIONAL INFERENCE



[Variational Bayesian phylogenetic inference]

“IF YOU’RE THINKING HOW DO WE KNOW IF RESULTING DISTRIBUTION IS CLOSER TO THE POSTERIOR IF POSTERIOR IS EXACTLY WHAT WE WANT TO CALCULATE, YOU’VE UNDERSTOOD THE IDEA.”

Making Your Neural Network Say “I Don’t Know”

$$\log P(\mathscr{D}) - D_{\mathrm{KL}}[Q(\theta) \| P(\theta \mid \mathscr{D})] = \underbrace{\mathbb{E}_{\theta \sim Q}[\log P(Y \mid X, \theta)] - D_{\mathrm{KL}}[Q(\theta) \| P(\theta)]}_{\textbf{Evidence Lower Bound or ELBO}}$$

VARIATIONAL BAYES IN CODE

JAX

- "Composable transformations of Python+NumPy programs: differentiate, vectorize, JIT to GPU/TPU, and more." ~ JAX Documentation
- `jit` (just-in-time compilation) - speeds up your code by running all the ops inside the jit-ed function as a fused op; it compiles the function when it's called the first time and uses the compiled (optimised) version from the second call onwards.
- `grad` - returns derivatives of function with respect to the model weights passed as parameters.
- `vmap` - automatic batching; returns a new function that can apply the original (per-sample) function to a batch.

Great introduction to JAX and Haiku from EEML practical sessions here.

DIVE INTO CODE!

You can find the code here on GitLab

WHAT NEXT?

- **A great post on Bayesian Inference — Intuition and Example.**
- **Learn about conjugate prior and why it's important, read here.**
  $P(\theta)$ **such that** $P(\theta \mid X)$ **is from the same probability distribution family as** $P(\theta)$**.**
- **Derive and understand ELBO, read here.**
- A Beginner's Guide to Variational Methods: Mean-Field Approximation.
- The problem of approximate inference in Variational Inference: A Review for Statisticians.
- Making Your Neural Network Say "I Don't Know" — Bayesian NNs using Pyro.

THANK YOU!