

Simulation-based Inference

Inverting Simulators

Talk by Stefan Wezel

mlcolab @ Tübingen University Cluster of Excellence

April 30, 2021

Overview

- The problem setting
- Traditional approaches and their issues
- Using Bayesian Neural Nets to alleviate them
- A worked example
- A more interesting, real-world example
- Outlook

What and Why?

An Example

- Inverting simulators - so first let's find out what a simulator is
- Imagine you are a physicist
- You have also highly sophisticated experiments that can produce observations
- You have highly sophisticated models, theories that can produce observations like from your experiment, that were developed over centuries
- But what do we actually care about? How do you perform inference?
 - What were the parameters for your observation

What and Why?

An Example

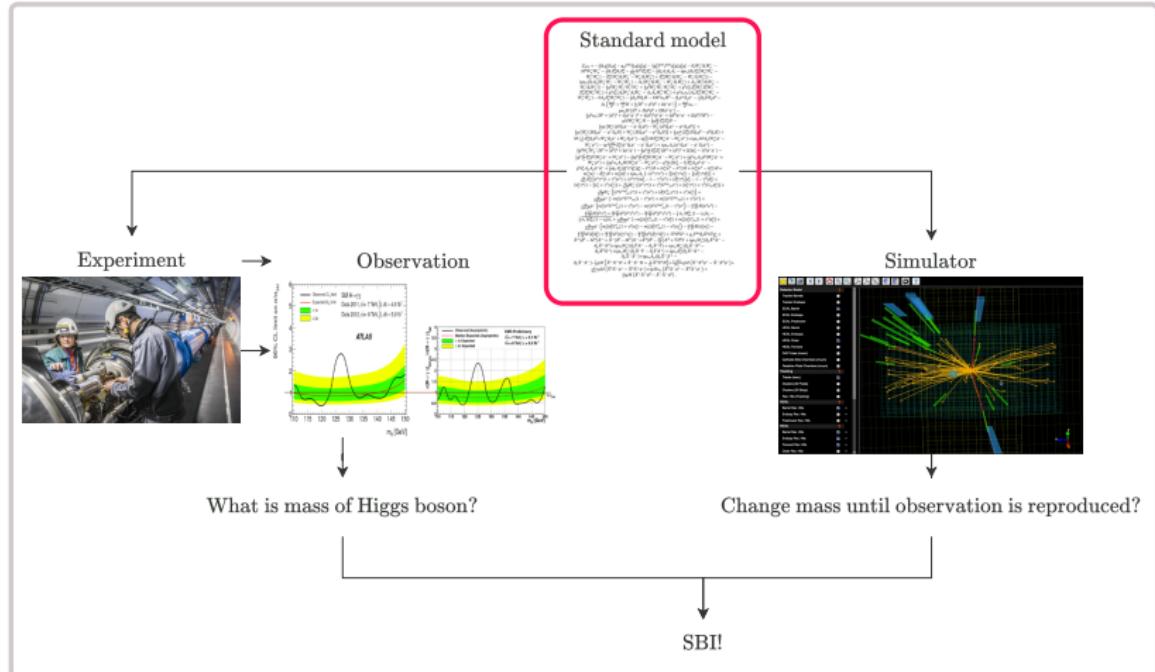


Figure derived from cern.ch, Achintya Rao and Tom McCauley, sciencealert.com

What and Why?

An Example

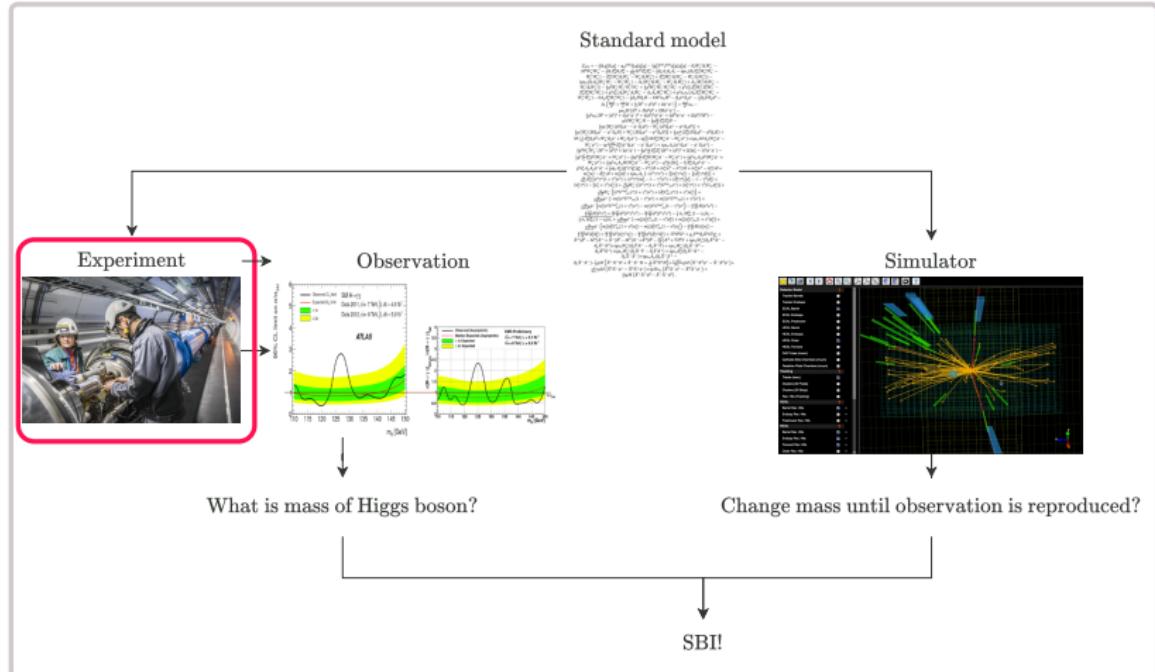


Figure derived from cern.ch, Achintya Rao and Tom McCauley, sciencealert.com

What and Why?

An Example

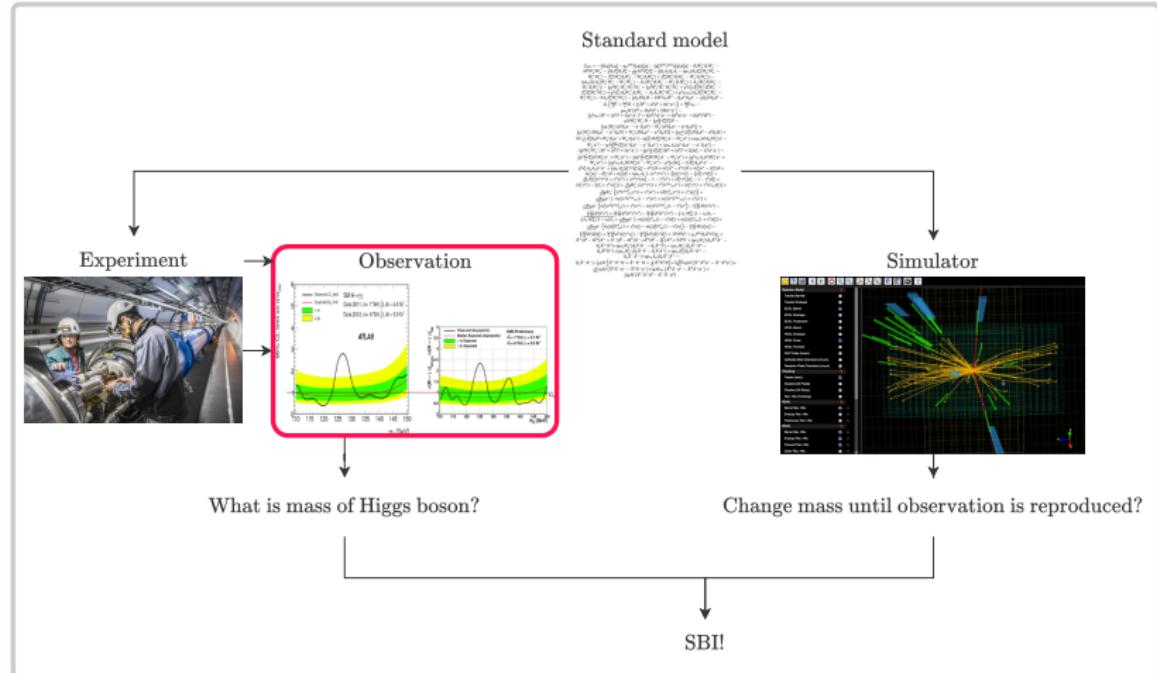


Figure derived from cern.ch, Achintya Rao and Tom McCauley, sciencealert.com

What and Why?

An Example

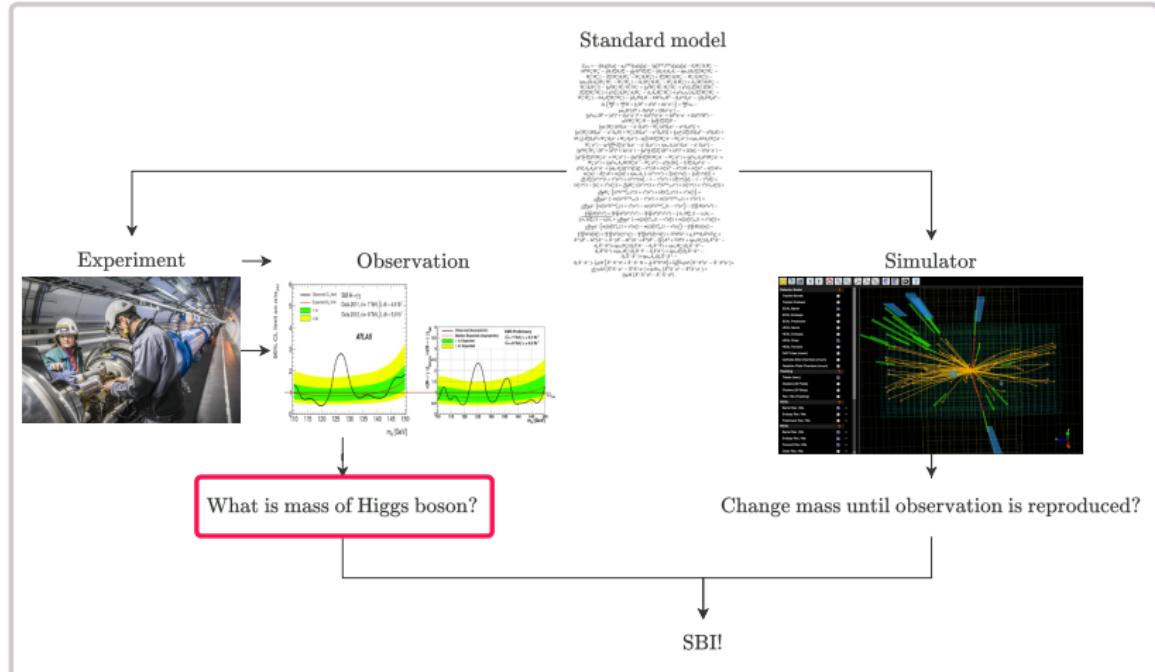


Figure derived from cern.ch, Achintya Rao and Tom McCauley, sciencealert.com

What and Why?

An Example

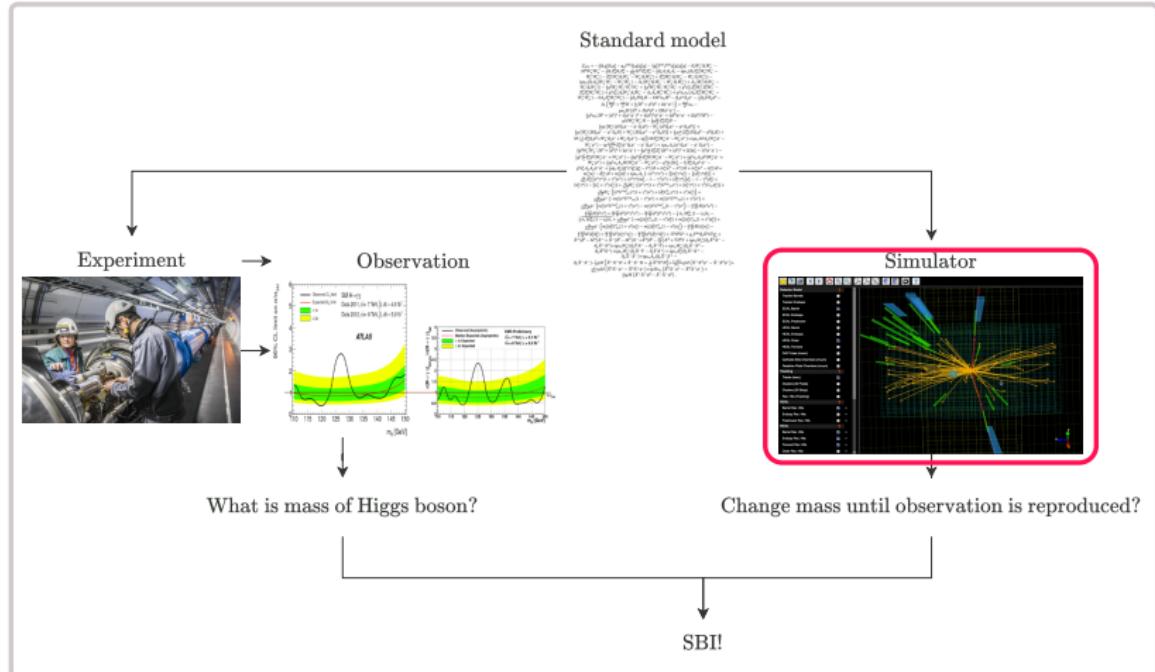


Figure derived from cern.ch, Achintya Rao and Tom McCauley, sciencealert.com

What and Why?

An Example

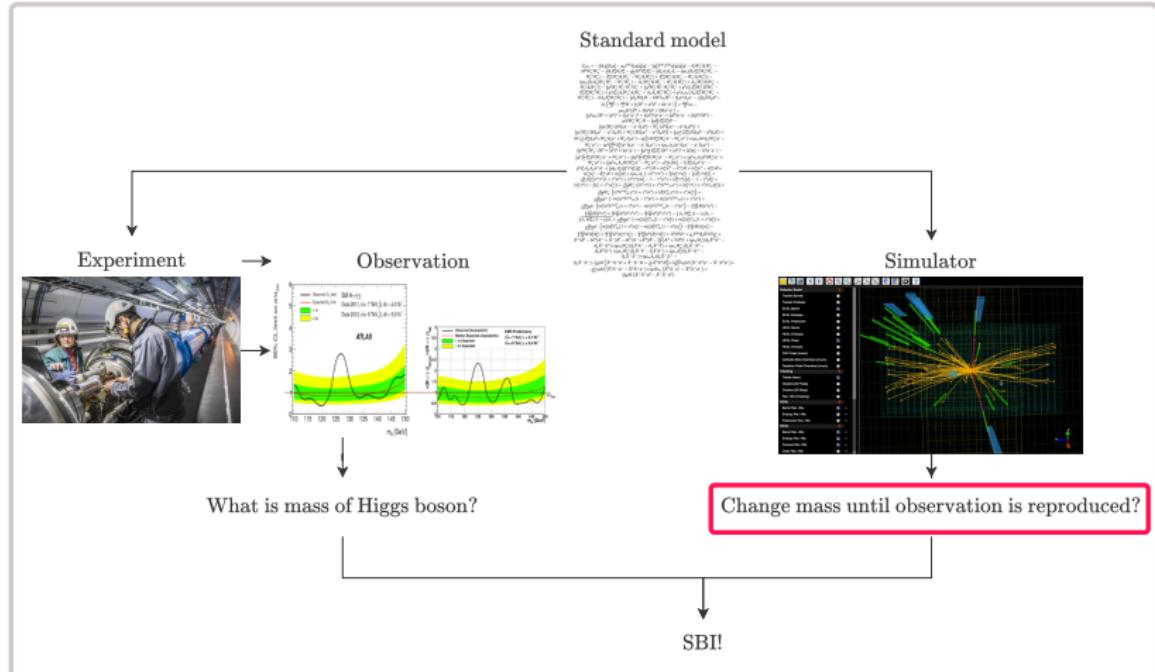


Figure derived from cern.ch, Achintya Rao and Tom McCauley, sciencealert.com

What and Why?

An Example

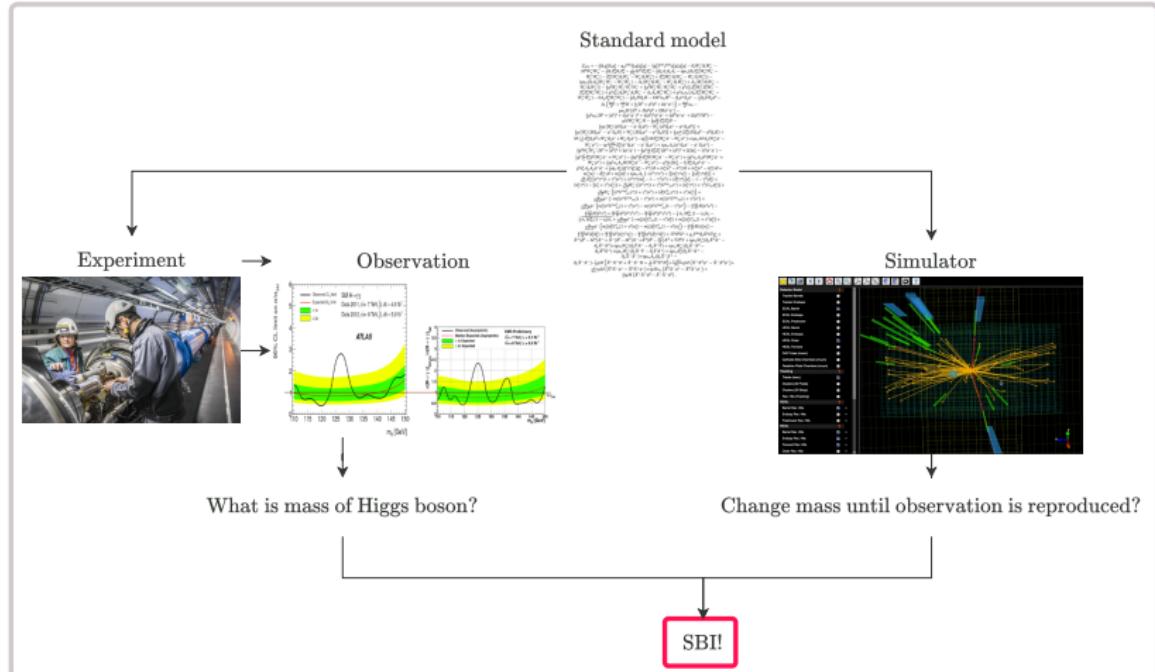


Figure derived from cern.ch, Achintya Rao and Tom McCauley, sciencealert.com

What and Why?

Formalizing our Example

- Before we go into SBI, let's take a minute to formalize our setting
- Simulator has parameters θ , nuisance parameter z and produces data \hat{x}
 - It implicitly defines $p(\hat{x}, z|\theta)$ (it yields samples from this distribution)
- Given a (real) observation x_0 , we would be interested in the parameters that produced it

$$p(\theta|x=x_0) = \frac{p(x|\theta)p(\theta)}{p(x)} = \frac{\overbrace{\int p(x, z|\theta)dz}^{intractable} p(\theta)}{\underbrace{\int p(x|\theta)p(\theta)d\theta}_{intractable}}, \quad (1)$$

- $q_\phi(\theta|x) = \Sigma_k \alpha_k \mathcal{N}(\theta|m_k, S_k) \sim$

What and Why?

Simulation-based Inference to the rescue

- This is exactly a problem setting that scientists often deal with
 - They have a sophisticated model (the simulator) that encapsulates a lot of prior knowledge
 - But inference in this setting often boils down to: What were parameters that produced this observation
- Simulation-based Inference tries to solve this problem
 - by inverting the simulator

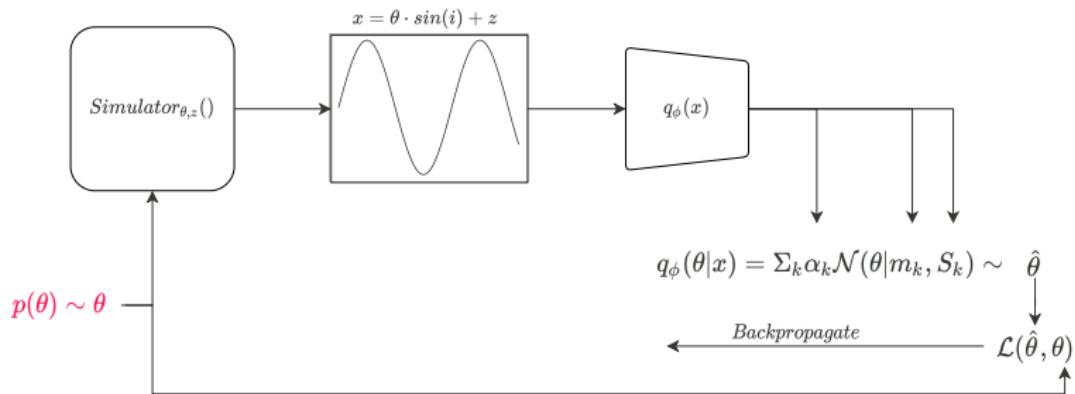
Simulation-based Inference

Traditional Approach

- Broader term: Approximate Bayesian Computation (ABC)
 - Rejection ABC
 - Sampling ABC (perturb initial params)
 - Sequential ABC
 - But gives only point estimates, not full posterior
 - within ϵ
- Some improvements are MCMC ABC and Sequential ABC but they only improve sample efficiency but still produce merely point estimate

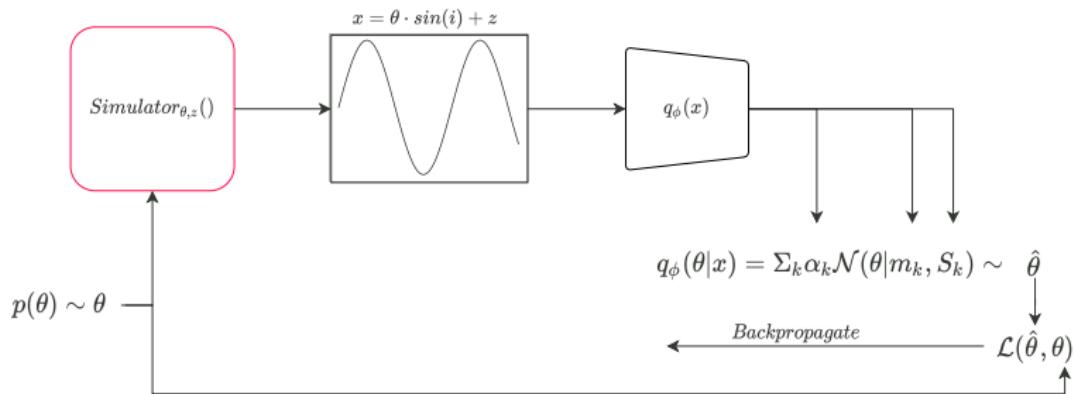
Learning a Posterior

from Simulations



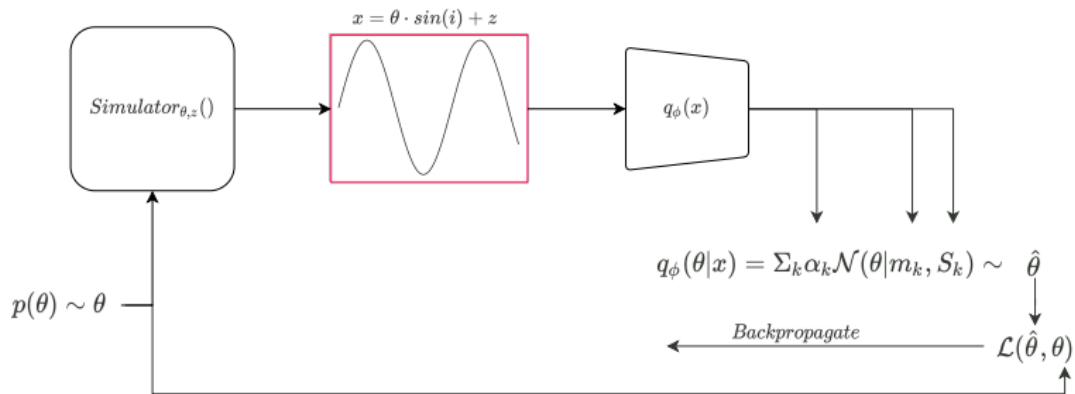
Learning a Posterior

from Simulations



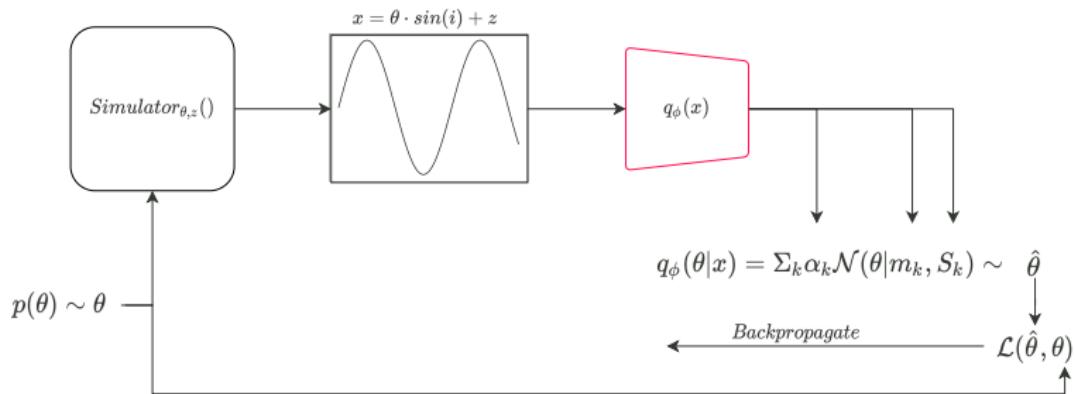
Learning a Posterior

from Simulations



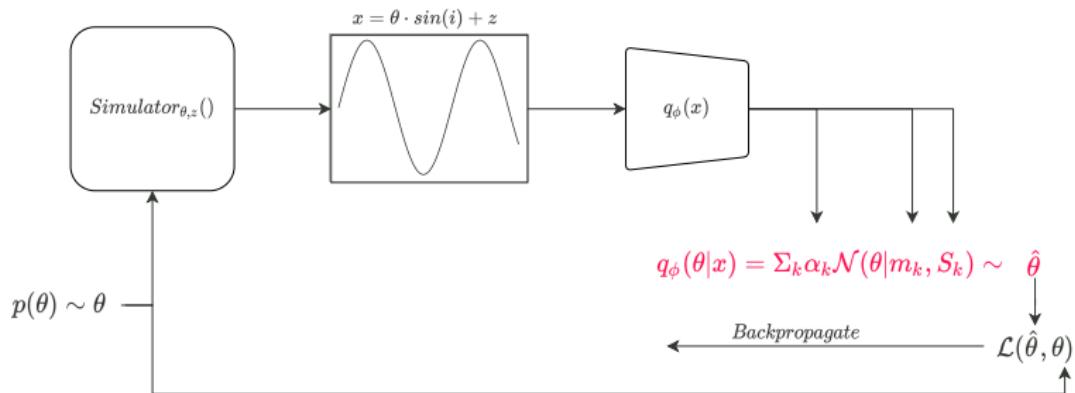
Learning a Posterior

from Simulations



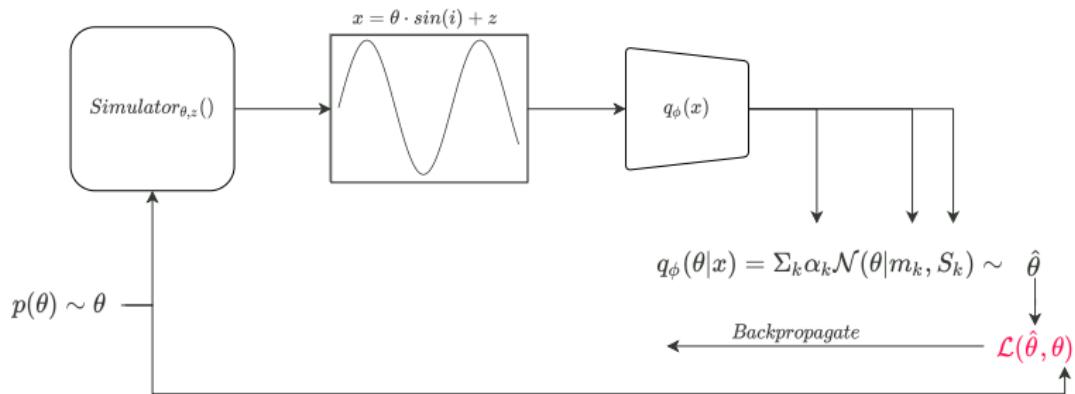
Learning a Posterior

from Simulations



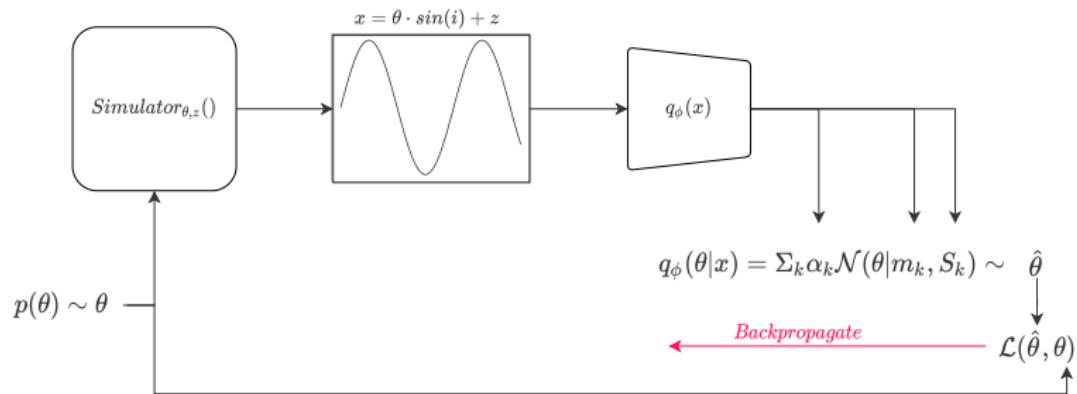
Learning a Posterior

from Simulations



Learning a Posterior

from Simulations



How to use SBI?

Some advertisement

- Now that you are pretty excited about SBI and its applications
- mlcolab and mackelab is developing a python (and eventually Julia) library (that I've used earlier)
- If you are interested: try it out and give us your feedback
- link to colab
- link to github



References

○

How

Approximate Bayesian Computation

- Finding parameters/posterior over parameters with
 - Rejection ABC
 - Markov Chain Monte Carlo ABC
 - Sequential Monte Carlo ABC
 - ...
- Problems
 -

What

is Simulation-based Inference?

- If we know generating factors, we can build a simulator
- But we want to constrain simulator output on observations from real world
- Thus we need realistic values for simulator parameters
- -> inverse problem
- SBI solvers this inverse problem using Bayesian inference

How

does Simulation-based Inference work?