Lecture 13

Bayesian Inference (Part 1)

(Based on slides by **Dr. Tom Rainforth**, HT 2020)

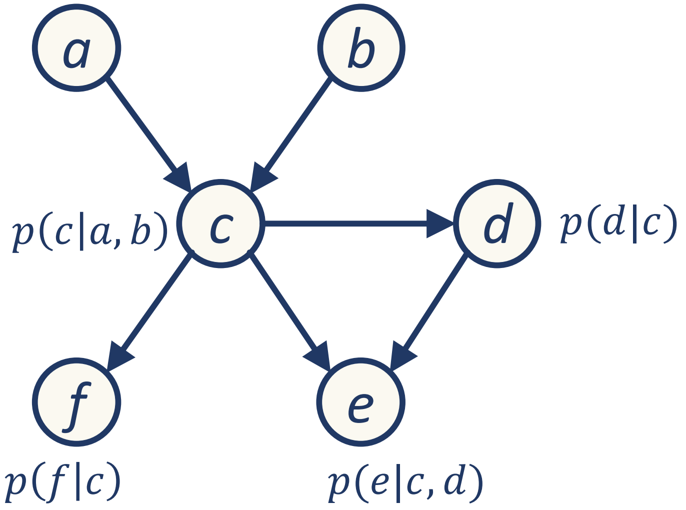
Jiarui Gan

jiarui.gan@cs.ox.ac.uk

Previous Lecture:

Bayesian modeling

- Graphical models



$$
\begin{aligned}
&p(a,b,c,d,e,f) \\
&= p(a) \cdot p(b) \cdot p(c|a,b) \cdot \\
&\quad p(d|c) \cdot p(f|c) \cdot p(e|c,d)
\end{aligned}
$$

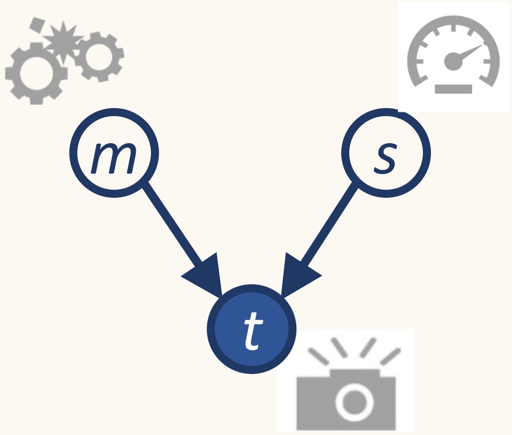
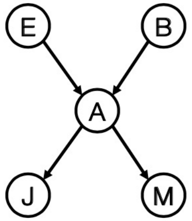
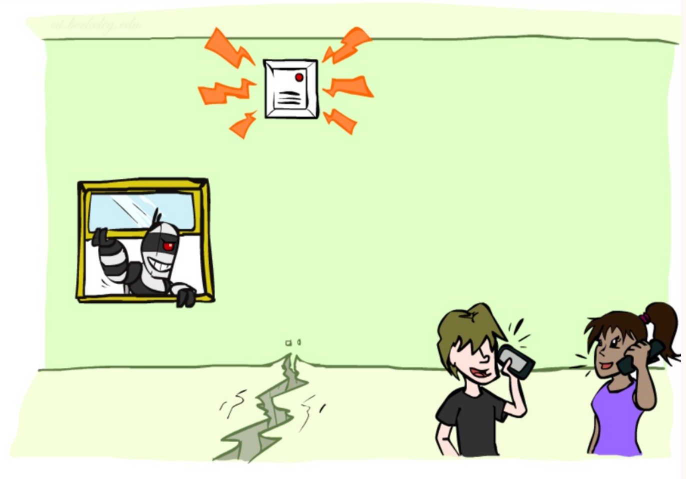


*Image credit: Autocar

Variables

- B: Burglary
- A: Alarm goes off
- M: Mary calls
- J: John calls
- E: Earthquake!





*Example and image credit: Pieter Abbeel

This Lecture

- Estimating and using Bayesian posteriors
- While previous lectures have focused on modeling, we will now be mostly concerned with computation instead; we will generally assume the model is given
- Why is Bayesian inference challenging? And ways to work around:
    - Deterministic Approximations
    - Monte Carlo
    - Rejection sampling
    - Importance sampling

Why is Bayesian inference challenging?

Bayesian Inference is Hard!

- It might at first seem like Bayesian inference is a straightforward problem
  - By Bayes' rule we have that $p(\theta|\mathcal{D}) \propto p(\mathcal{D}|\theta) \cdot p(\theta)$ and so we already know the relative probability of any one value of θ compared to another
- In practice, this could hardly be further from the truth
  - For non–trivial models, Bayesian inference is akin to calculating a high–dimensional integral: the normalization constant $p(\mathcal{D}) = \int p(\mathcal{D}|\theta) \cdot p(\theta)\, d\theta$
  - It is, in general, an NP-hard problem (the examples we have considered so far are special cases)

NP-hardness of Bayesian inference

- We can reduce the NP-complete problem **3SAT** to Bayesian inference

$(x_1 \lor x_2 \lor x_5) \land$

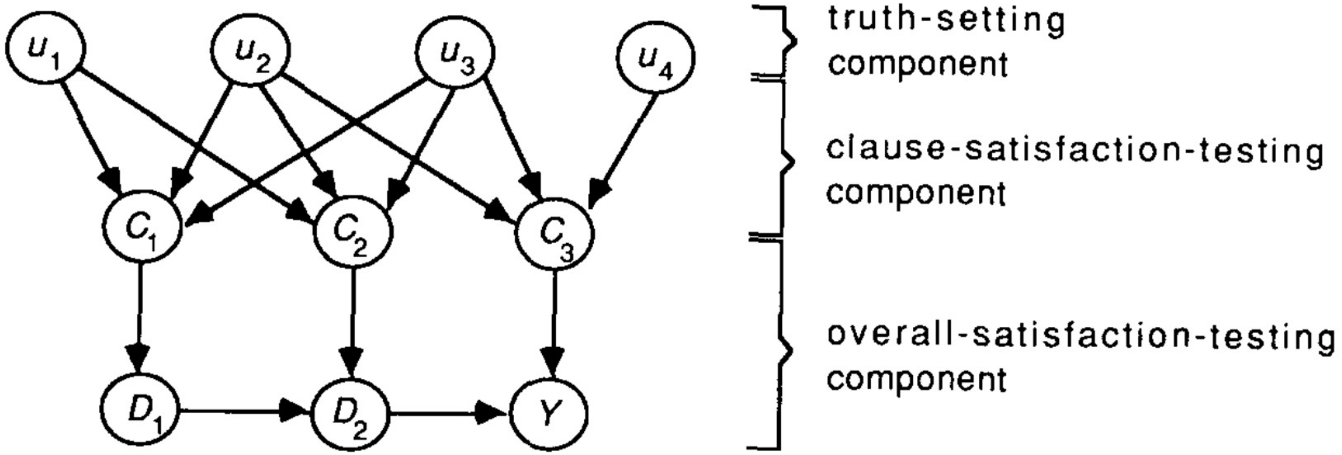
$(\neg x_2 \lor \neg x_7 \lor \neg x_3) \land$

…

…

$(\neg x_9 \lor x_1 \lor x_4)$

3SAT


u_1
u_2
u_3
u_4
C_1
C_2
C_3
D_1
D_2
Y
truth-setting component
clause-satisfaction-testing component
overall-satisfaction-testing component

Bayesian inference

- G. F. Cooper, *The computational complexity of probabilistic inference using Bayesian belief networks,* Artificial Intelligence, *1990*

Why is Bayesian Inference Hard?

- We can break down Bayesian inference into two key challenges:
  - Calculating the normalization constant $p(\mathcal{D}) = \int p(\mathcal{D}|\theta) \cdot p(\theta)\, d\theta$
  - Providing a useful characterization of the posterior $p(\theta|\mathcal{D})$, for example, a set of approximate samples
- Each of these constitutes a somewhat distinct problem
- Many methods sidestep the first problem and directly produce approximate samples

The Normalization Constant (1)

- If $p(\mathcal{D}) = \int p(\mathcal{D}|\theta) \cdot p(\theta)\, d\theta$ is unknown, we lack scaling when evaluating a point
  - We have no concept of how relatively significant that point is compared to the distribution as a whole
  - We don't know how much mass is missing
  - The larger the space of θ, the more difficult this becomes

$$p(\theta|\mathcal{D}) = \frac{p(\theta) \cdot p(\mathcal{D}|\theta)}{p(\mathcal{D})}$$

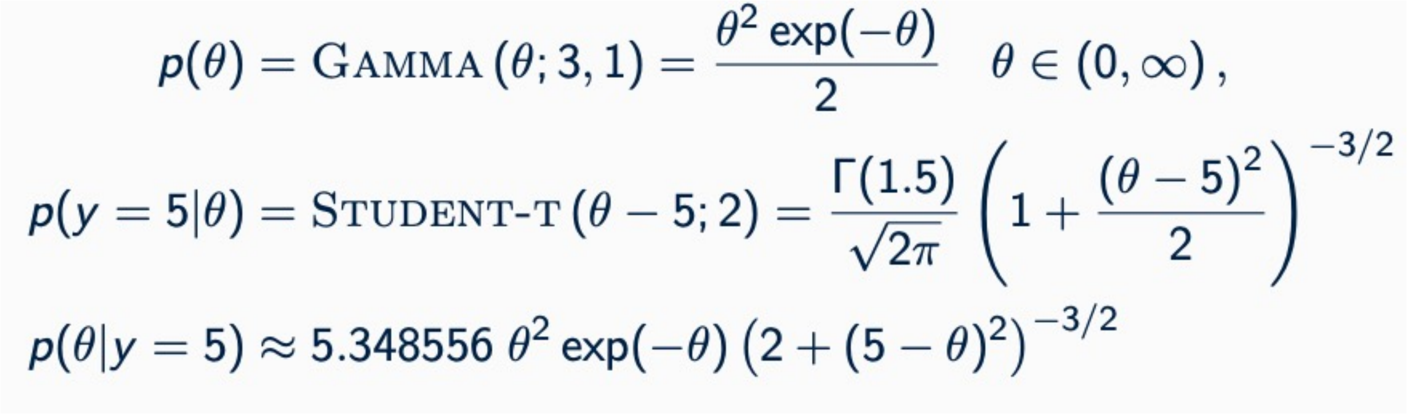
* Image Credit: www.theescapeartist.me

The Normalization Constant (2)

- In practice, even having an exact form for $p(\theta|\mathcal{D})$ is often not enough for many tasks we might want to carry out when θ is continuous or has a very large number of possible values:
  - To make **predictions** using the posterior predictive distribution
  - To calculate the **expected value** of some function, $\mathbb{E}_{p(\theta|\mathcal{D})}\left[f(\theta)\right]$
  - To find the most probable variable values $\theta^* = \arg\max_{\theta} p(\theta|\mathcal{D})$
  - To produce a useful representation of the posterior for passing on to another part of a computational pipeline or to be directly observed by a user
- Knowing is $p(\mathcal{D})$ is only sufficient for this first of these tasks. The others require additional computation of some form

Characterizing the Posterior: Example

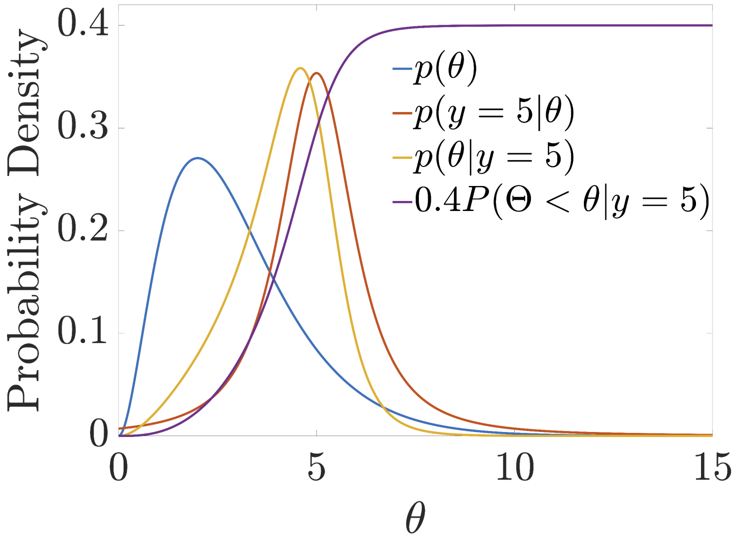
- Lets consider a simple example where we can easily calculate $p(\mathcal{D})$, and thus $p(\theta|\mathcal{D})$, numerically:



$$p(\theta) = \textsc{Gamma}(\theta; 3, 1) = \frac{\theta^2 \exp(-\theta)}{2} \quad \theta \in (0, \infty),$$

$$p(y = 5|\theta) = \textsc{Student-t}(\theta - 5; 2) = \frac{\Gamma(1.5)}{\sqrt{2\pi}} \left(1 + \frac{(\theta - 5)^2}{2}\right)^{-3/2}$$

$$p(\theta|y = 5) \approx 5.348556\, \theta^2 \exp(-\theta) \left(2 + (5 - \theta)^2\right)^{-3/2}$$



- Even though we have the posterior in closed form, it is not a standard distribution and so we don't know how to sample from it
  - Even more difficult for higher dimensional problems

Deterministic Approximations

Point Estimates

- One of the simplest approaches is to effectively ignore the posterior computation problem completely and instead resort to a **heuristic approximation**
- The simplest such approach is to take a **point estimate** $\tilde{\theta}$ for θ (i.e., no uncertainty) and then approximate the posterior predictive distribution using only this value:

$$p(\mathcal{D}^*|\mathcal{D}) \approx p\left(\mathcal{D}^*\middle|\tilde{\theta}\right).$$

$$p(\mathcal{D}^*|\mathcal{D}) \coloneqq \int p(\mathcal{D}^*|\theta) \cdot p(\theta)\, d\theta$$

Point Estimates

- One of the simplest approaches is to effectively ignore the posterior computation problem completely and instead resort to a **heuristic approximation**
- The simplest such approach is to take a **point estimate** $\tilde{\theta}$ for θ (i.e., no uncertainty) and then approximate the posterior predictive distribution using only this value:

$$p(\mathcal{D}^*|\mathcal{D}) \approx p\left(\mathcal{D}^*\middle|\tilde{\theta}\right).$$

- Finding $\tilde{\theta}$ requires only an **optimization** problem to be solved
  - This is far easier than the **integration** problem posed by full posterior inference

Maximum Likelihood

- **Maximum likelihood (MLH)** is a non-Bayesian, frequentist, approach for calculating a $\tilde{\theta}$ based on maximizing the likelihood:

$$\tilde{\theta}_{\mathrm{MLH}} = \mathrm{argmax}_{\theta}\, p(\mathcal{D}|\theta)$$

☹ This can be prone to overfitting and **does not incorporate prior** information, leading to a host of issues we previously discussed (see the lecture notes: Bayesian vs frequentist)

Maximum a Posteriori (MAP)

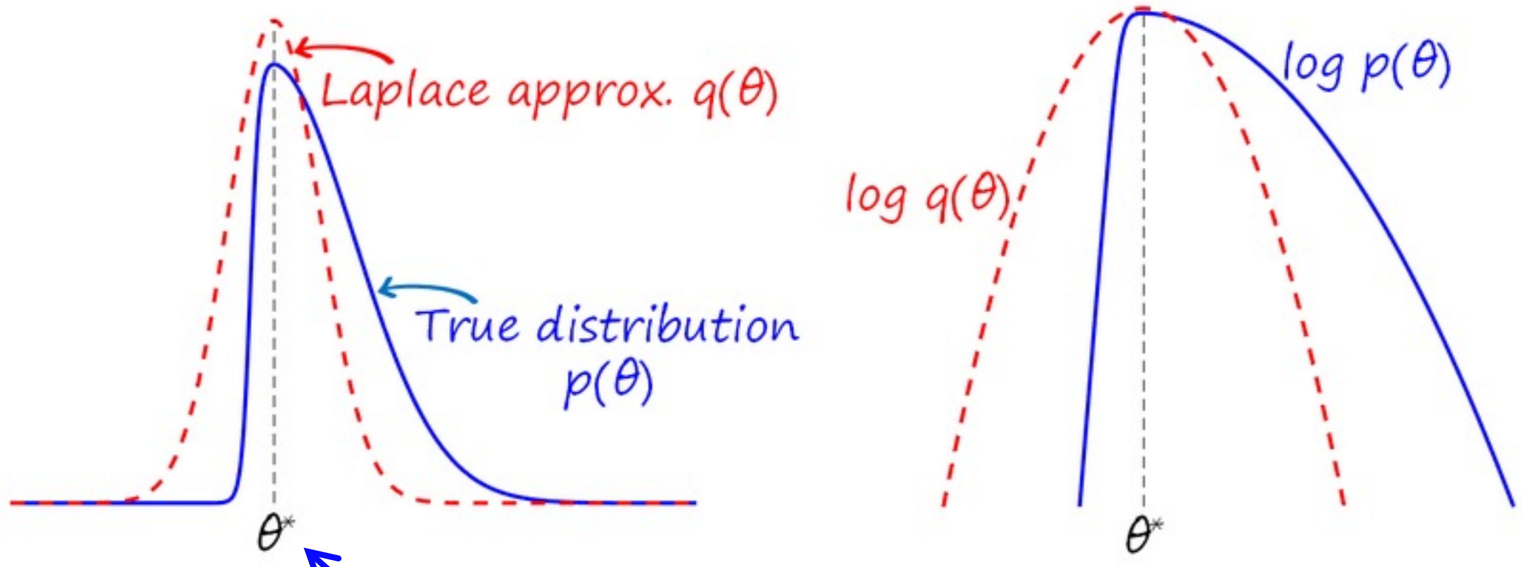
- **Maximum a Posteriori (MAP)** estimation corresponds to choosing $\tilde{\theta}$ to maximize the posterior probability:

$$\tilde{\theta}_{\mathrm{MAP}} = \mathrm{argmax}_{\theta}\, p(\mathcal{D}|\theta) \cdot p(\theta) \equiv \mathrm{argmax}_{\theta}\, p(\theta|\mathcal{D})$$

- ☺ This provides regularization compared to MLH estimation
- ☹ but still has a number of drawbacks compared to **full inference**:
  - It incorporates less information into the predictive distribution
  - The position of the MAP estimate is dependent of the parametrization of the problem (see notes on change of variables)

Laplace Approximation (1)

- The **Laplace approximation** refines the MAP estimate by approximating the full posterior with a Gaussian centered at the **MAP estimate** and covariance dictated by the curvature of the log density around this point

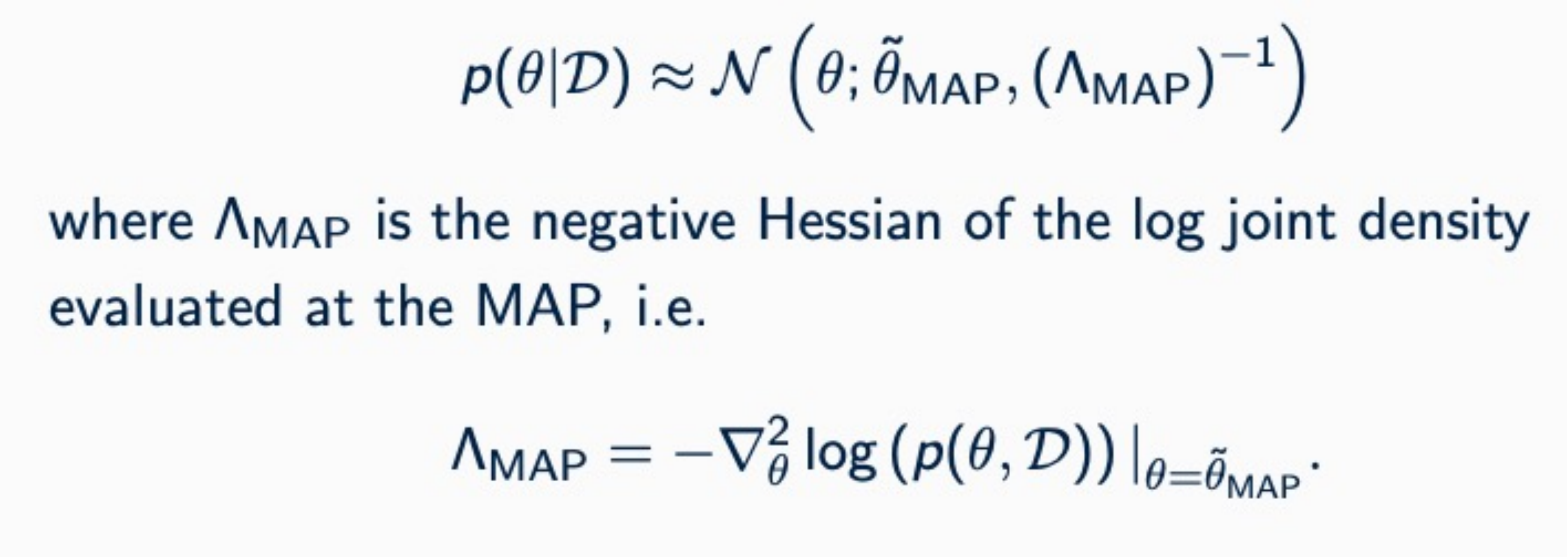


MAP estimate in our case

*Images Credit: Luis Herranz

Laplace Approximation (2)

- More formally, the Laplace approximation is given by



$$p(\theta|\mathcal{D}) \approx \mathcal{N}\left(\theta; \tilde{\theta}_{\text{MAP}}, (\Lambda_{\text{MAP}})^{-1}\right)$$

where Λ_{MAP} is the negative Hessian of the log joint density evaluated at the MAP, i.e.

$$\Lambda_{\text{MAP}} = -\nabla_\theta^2 \log\left(p(\theta, \mathcal{D})\right)\big|_{\theta=\tilde{\theta}_{\text{MAP}}}.$$

Monte Carlo

Monte Carlo

DEFINITION

Monte Carlo is the characterization of a probability distribution through **random sampling**.

- It forms the underlying principle for all stochastic computation
  - The foundation for a huge array of methods for numerical integration, optimization, and Bayesian inference
- It provides us with a means of dealing with complex models and problems in a statistically principled manner

Monte Carlo Estimators

- Consider the problem of calculating the expectation of some function $f(\theta)$ under the distribution $\theta \sim \pi(\theta)$:

$$I := \mathbb{E}_{\pi(\theta)}\left[f(\theta)\right] = \int f(\theta) \cdot \pi(\theta) d\theta$$

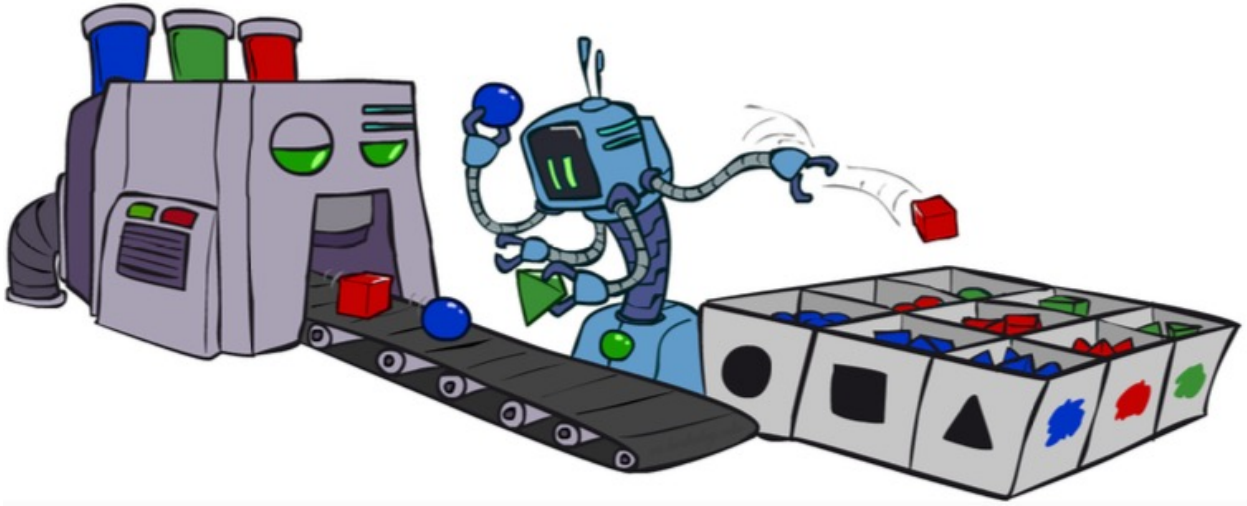
  This can be approximated using the **Monte Carlo estimator** I_N:

$$I \approx I_N := \frac{1}{N} \sum_{n=1}^{N} f(\hat{\theta}_n) \quad \text{where} \quad \hat{\theta}_n \sim \pi(\theta)$$

  are independent draws from $\pi(\theta)$

- Most of the tasks we laid out for Bayesian inference can be formulated as some form of (potentially implicit) expectation

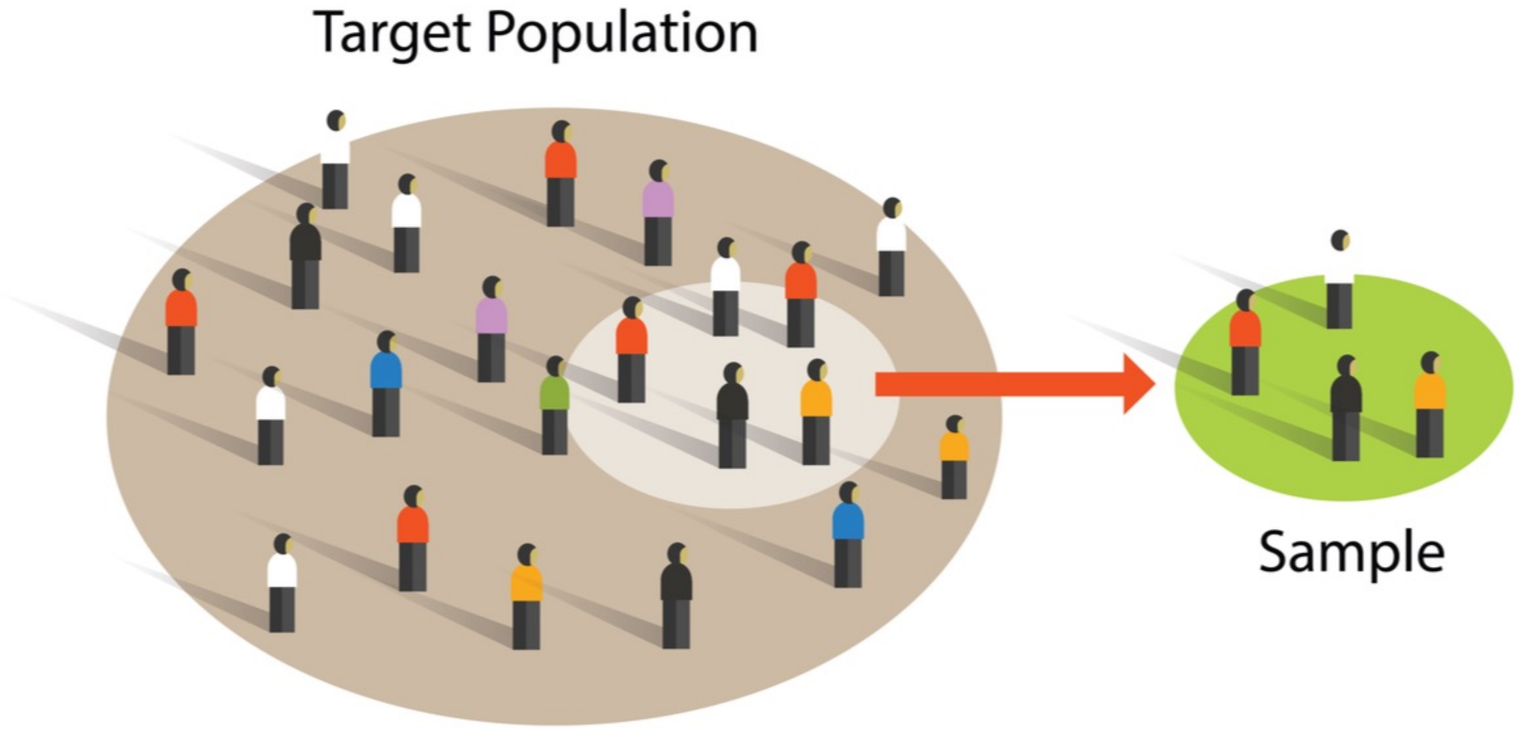
Example: Production Line



*Images Credit: Pieter Abbeel

- The production machine randomly generates colored shapes from some distribution, a robot sorts them into bins
- The production machine is performing Monte Carlo sampling, the robot is constructing a Monte Carlo estimate

Example: Election Polling



*Images Credit: Anthony Figueroa

Unbiasedness (1)

- The Monte Carlo estimate is unbiased (for fixed N), i.e. $\mathbb{E}[I_N] = I$

$$\mathbb{E}[I_N] = \mathbb{E}\left[\frac{1}{N}\sum_{n=1}^{N} f(\hat{\theta}_n)\right]$$

$$= \frac{1}{N}\sum_{n=1}^{N} \mathbb{E}[f(\hat{\theta}_n)]$$

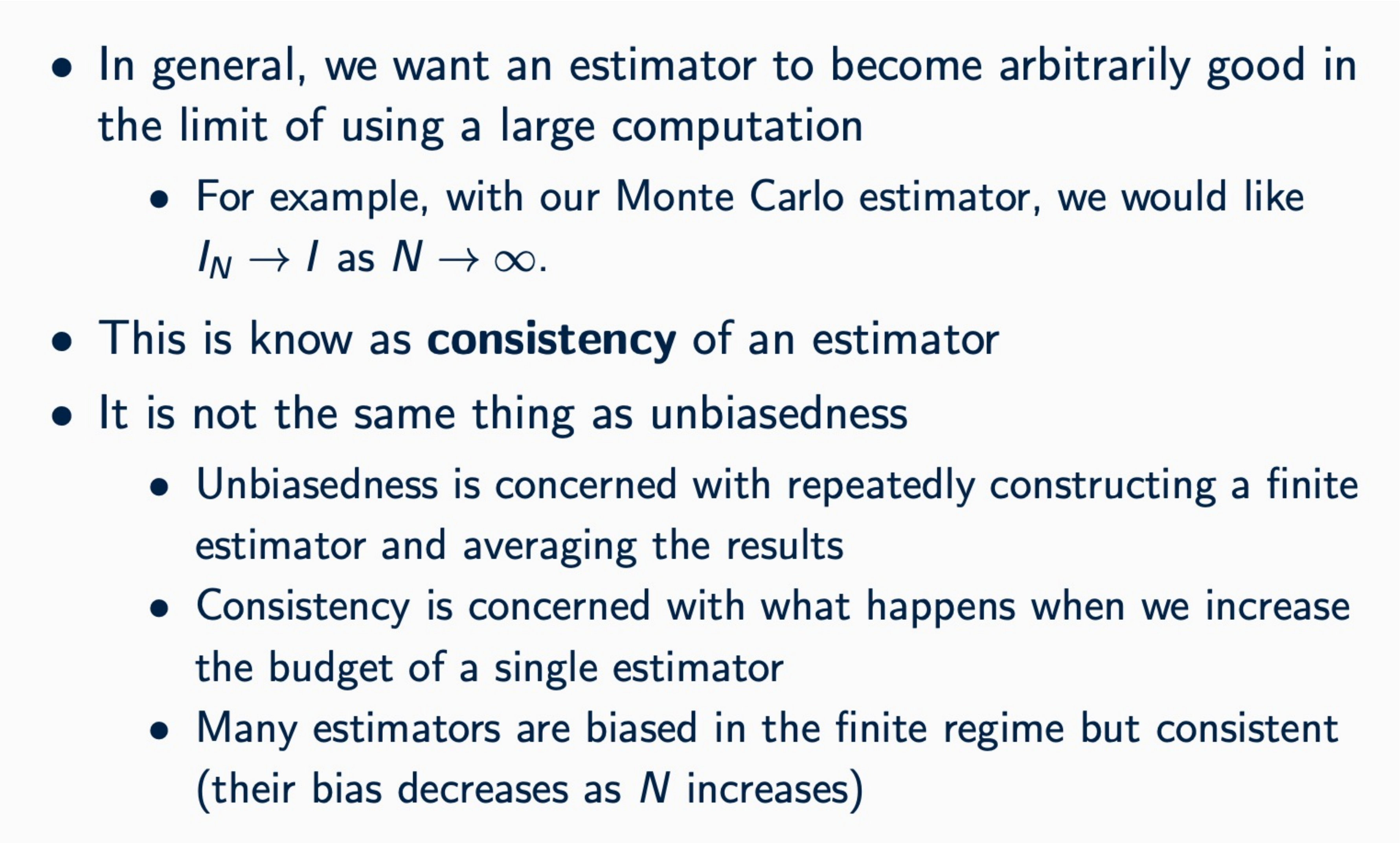
$$= \frac{1}{N}\sum_{n=1}^{N} \mathbb{E}[f(\hat{\theta}_1)] = \mathbb{E}[f(\hat{\theta}_1)] \ = I$$

Unbiasedness (1)

What exactly does unbiasedness mean?

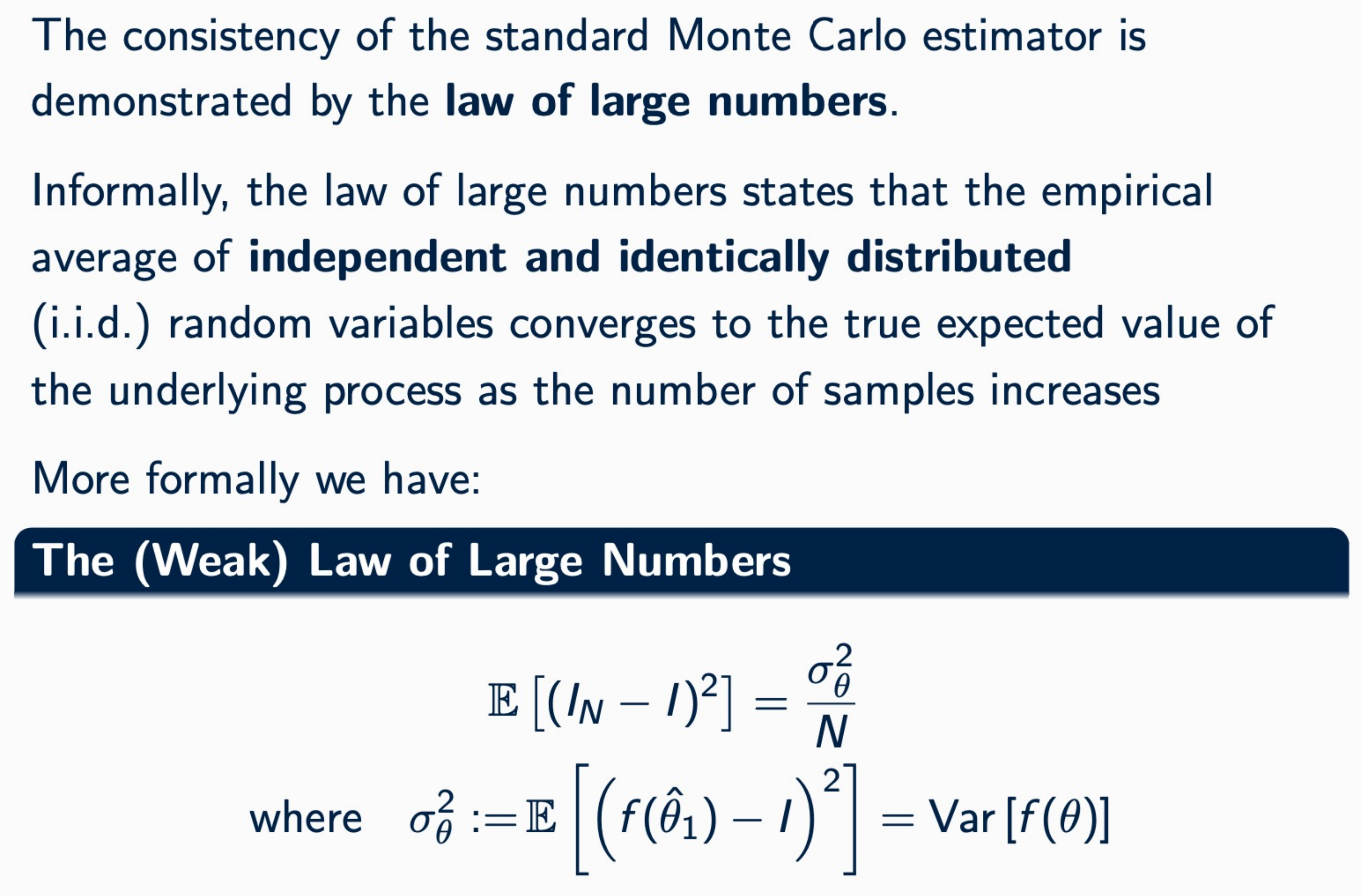
- It means that Monte Carlo does not introduce any systematic error, i.e. **bias**, into the approximation
  - In expectation, it does not overestimate or underestimate the target
  - A biased estimator $\tilde{I}$ would have $\mathbb{E}[\tilde{I}] = I + B$ for some $B \neq 0$
  - Here we are implicitly using the frequentist definition of probability: the expectation is defined through repeating the sampling infinitely often
- It does **not** mean that it is equally likely to overestimate or underestimate
  - It may, for example, typically underestimate by a small amount and then rarely overestimate by a large amount

Consistency of an Estimator



- In general, we want an estimator to become arbitrarily good in the limit of using a large computation
  - For example, with our Monte Carlo estimator, we would like $I_N \to I$ as $N \to \infty$.
- This is know as **consistency** of an estimator
- It is not the same thing as unbiasedness
  - Unbiasedness is concerned with repeatedly constructing a finite estimator and averaging the results
  - Consistency is concerned with what happens when we increase the budget of a single estimator
  - Many estimators are biased in the finite regime but consistent (their bias decreases as N increases)

The Law of Large Numbers



The consistency of the standard Monte Carlo estimator is demonstrated by the **law of large numbers**.

Informally, the law of large numbers states that the empirical average of **independent and identically distributed** (i.i.d.) random variables converges to the true expected value of the underlying process as the number of samples increases

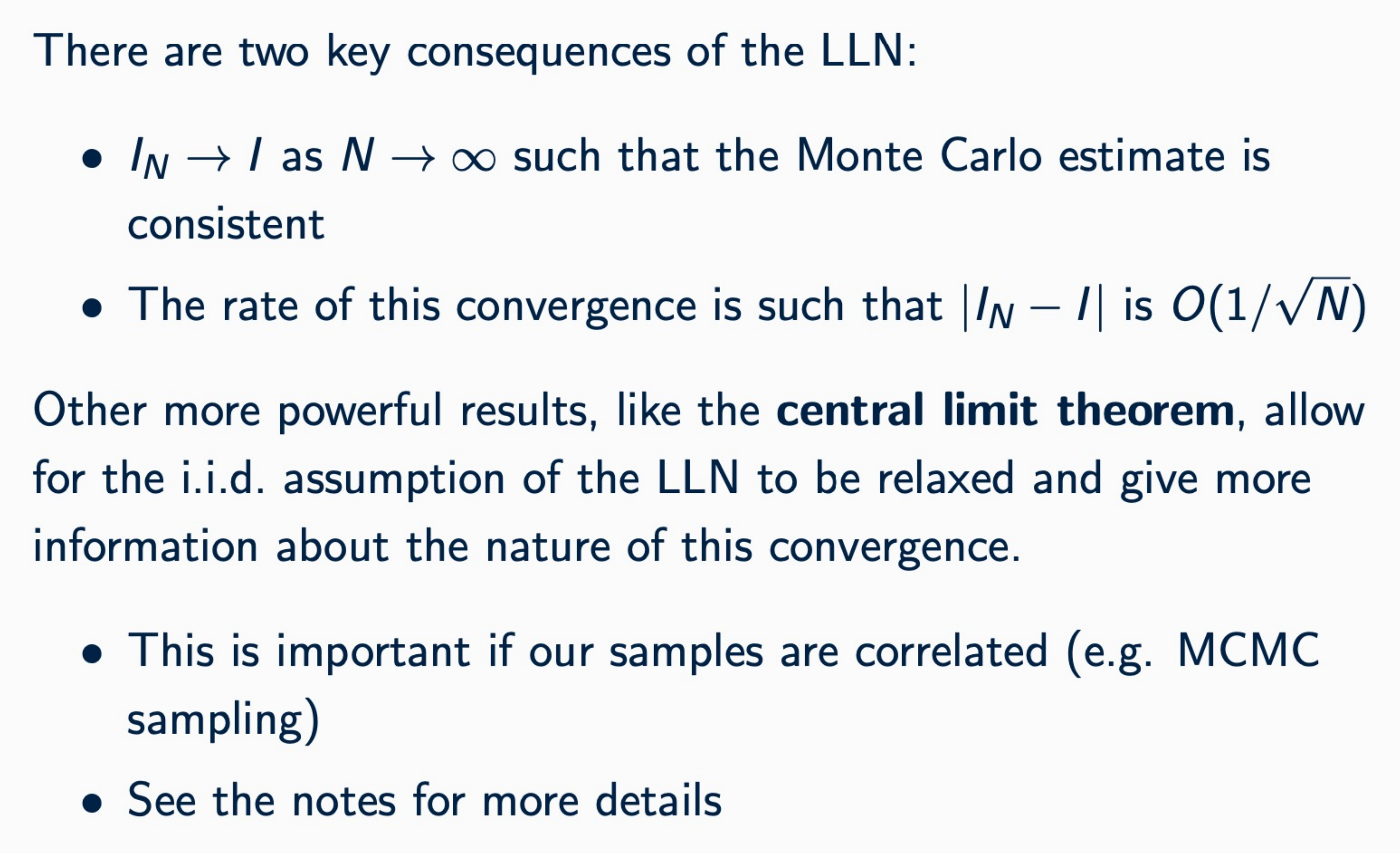
More formally we have:

The (Weak) Law of Large Numbers

$$\mathbb{E}\left[(I_N - I)^2\right] = \frac{\sigma_\theta^2}{N}$$

$$\text{where} \quad \sigma_\theta^2 := \mathbb{E}\left[\left(f(\hat{\theta}_1) - I\right)^2\right] = \mathrm{Var}\left[f(\theta)\right]$$

The Law of Large Numbers (2)



There are two key consequences of the LLN:

- $I_N \to I$ as $N \to \infty$ such that the Monte Carlo estimate is consistent
- The rate of this convergence is such that $|I_N - I|$ is $O(1/\sqrt{N})$

Other more powerful results, like the **central limit theorem**, allow for the i.i.d. assumption of the LLN to be relaxed and give more information about the nature of this convergence.

- This is important if our samples are correlated (e.g. MCMC sampling)
- See the notes for more details

Monte Carlo vs Classical Integration Schemes

- Classical integration approaches like Simpson's rule can offer far better convergence rates in low dimensions that the $O(1/\sqrt{N})$ of Monte Carlo

- But these rates break down (typically exponentially) as the dimension increases

- In high–dimensions, Monte Carlo estimates are one of the only approaches that can remain accurate

Drawing Samples

Drawing Samples

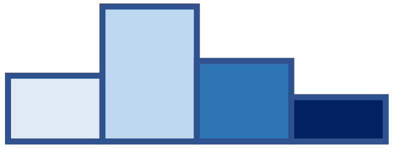
- We have shown how to use samples to characterize distributions and estimate expectations
- But how to we draw these samples in the first place?
- We'll now introduce a number of sampling schemes
- Note that most (with the exception of our first example) will not require us to know the normalization constant $p(\mathcal{D})$: they can operate on $p(\theta, \mathcal{D})$ directly

Sampling Using the Inverse CDF

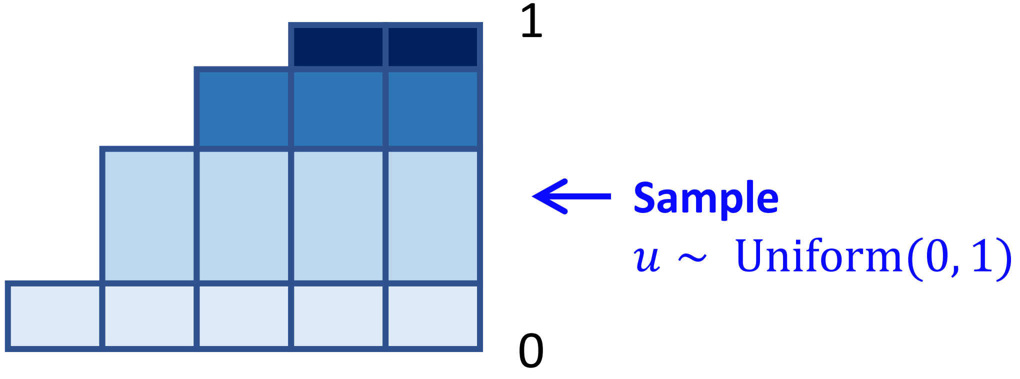
- If we know the cumulative density function (CDF) of the posterior

$$P(\theta \leq x \mid \mathcal{D}) \coloneqq \int_{\theta'=-\infty}^{\theta'=x} p(\theta = \theta' | \mathcal{D})\, d\theta'$$

along with its inverse P^{-1} (we rarely do in practice), then we can draw exact samples by first sampling $u \sim \mathrm{Uniform}(0,1)$ and then taking $x = P^{-1}(u)$, i.e. we have $u = P(\theta \leq x \mid \mathcal{D})$



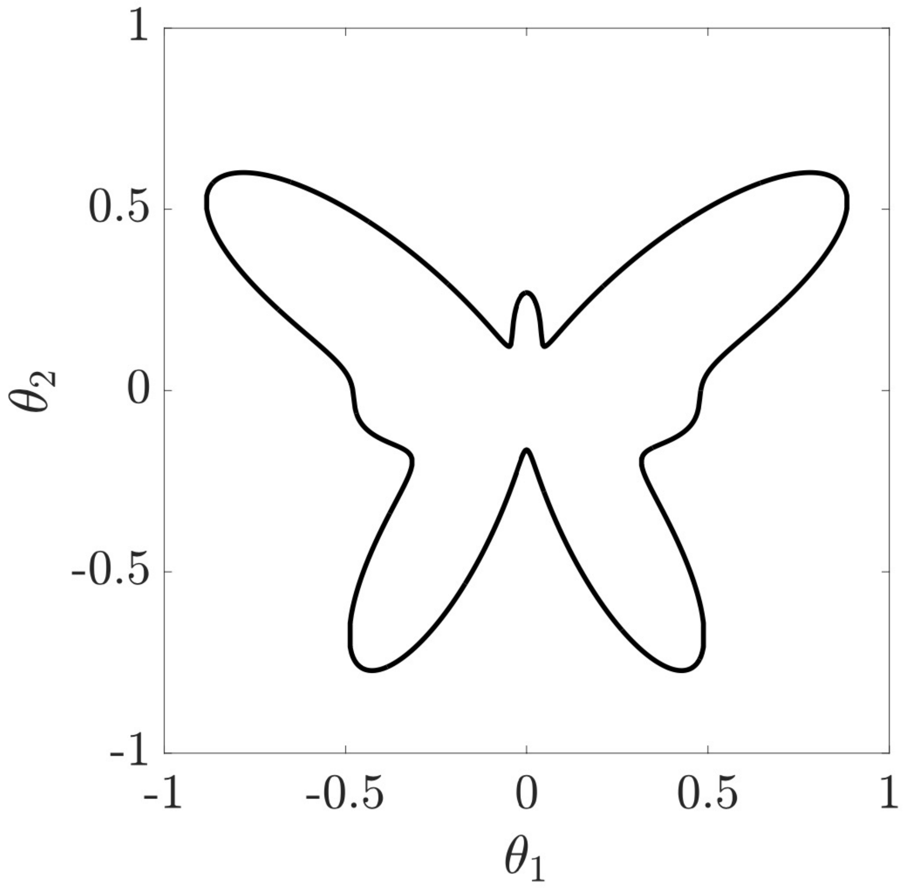
pdf



cdf

Sampling by Rejection

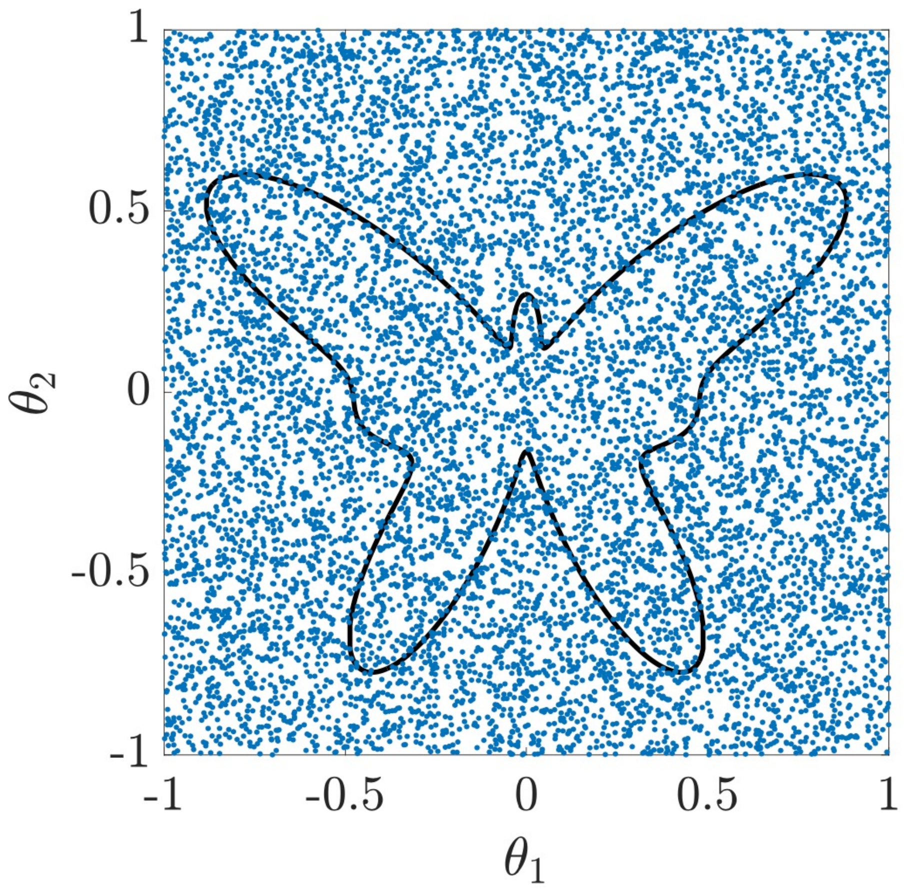
- How might we draw samples uniformly from within this butterfly shape?



Sampling by Rejection

- How might we draw samples uniformly from within this butterfly shape?

- We can draw samples uniformly from a surrounding box

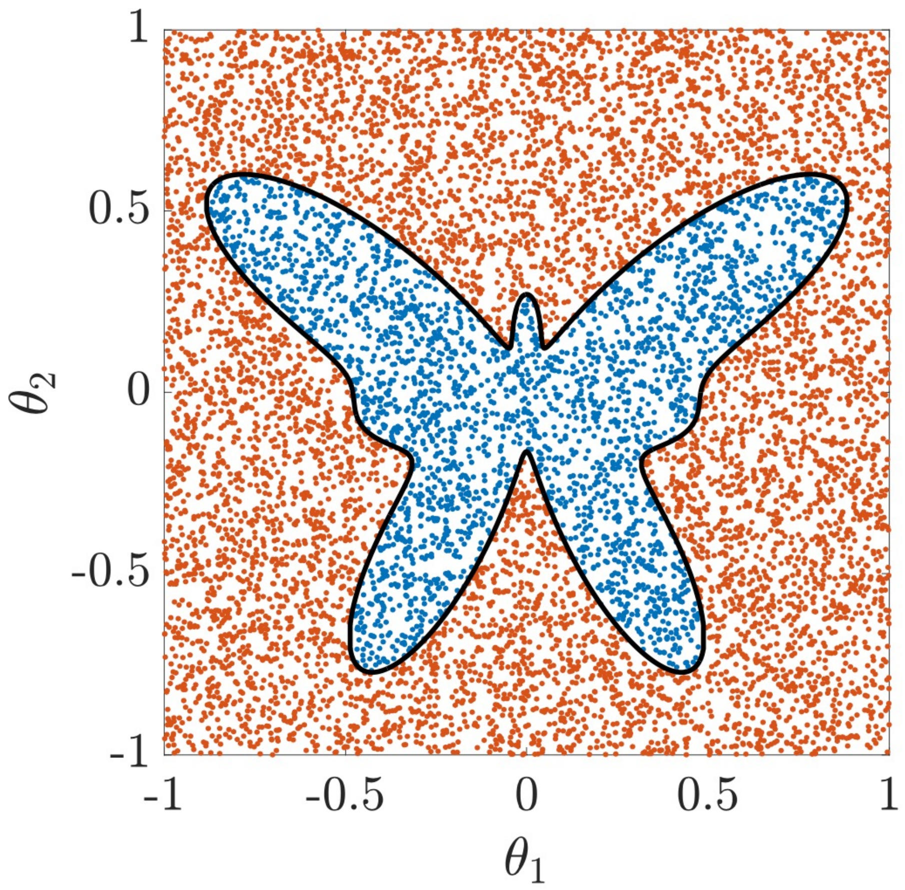


Sampling by Rejection

- How might we draw samples uniformly from within this butterfly shape?

- We can draw samples uniformly from a surrounding box

- Then reject those not falling within the shape



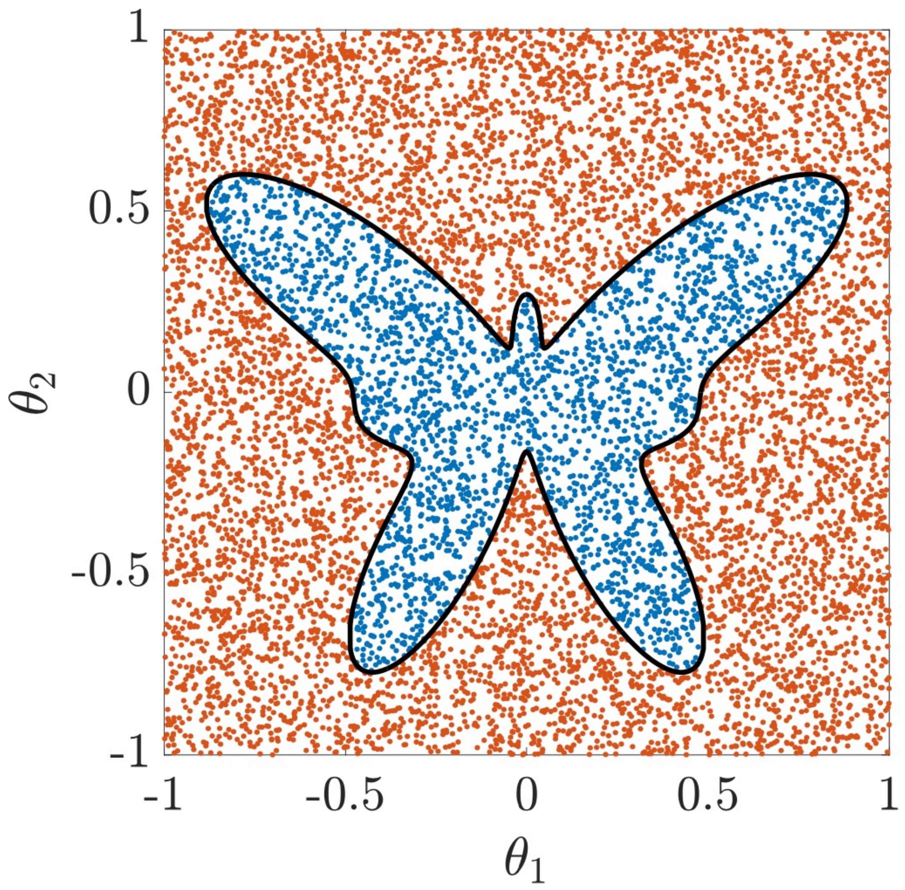
Sampling by Rejection: Estimate Shape Area

☺ The probability of any one sample falling within the shape is equal to the ratio of the areas of the shape and bounding box

$$A_{\text{shape}} = A_{\text{box}} \cdot P(\theta \in \text{shape})$$

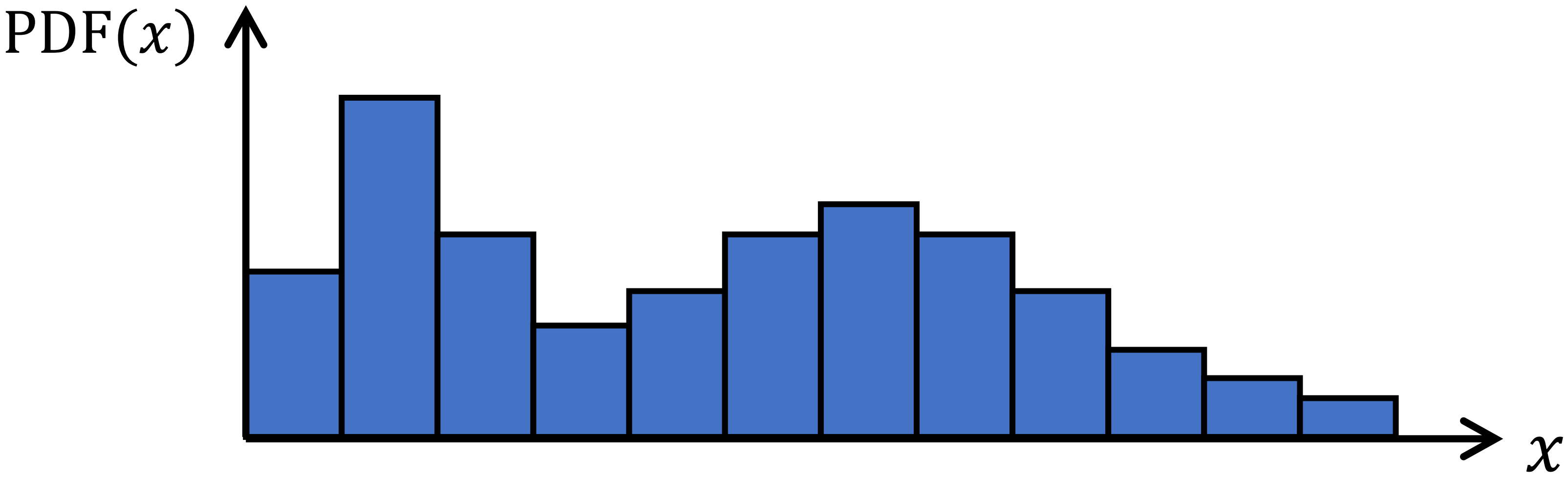
$$\approx \frac{A_{\text{box}}}{N} \sum_{n=1}^{N} \mathbb{I}(\hat{\theta}_n \in \text{shape})$$

- Here we have used a Monte Carlo estimator for $P(\theta \in \text{shape})$
- Note that the value of $P(\theta \in \text{shape})$ will dictate the efficiency of our estimation as it represents the **acceptance rate** of our samples



Sampling from Area Under Density

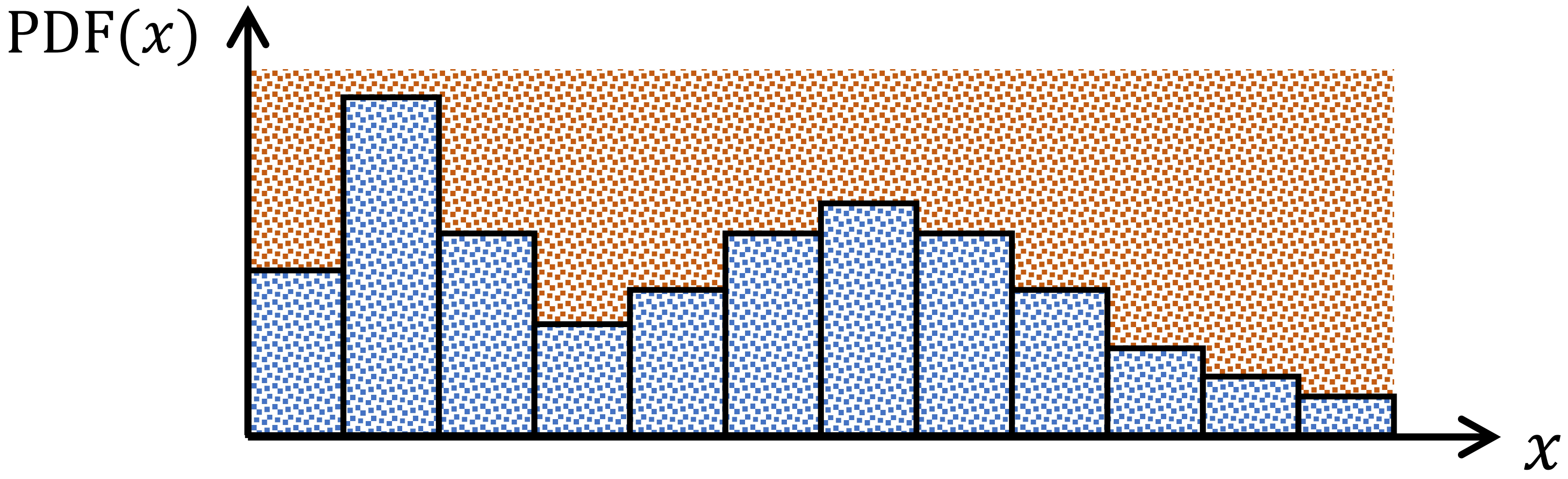
- Sampling from the area under a density function is equivalent to sampling from that density itself



Think about sampling from a histogram with even width bins and then take the width of these bins to zero

Sampling from Area Under Density

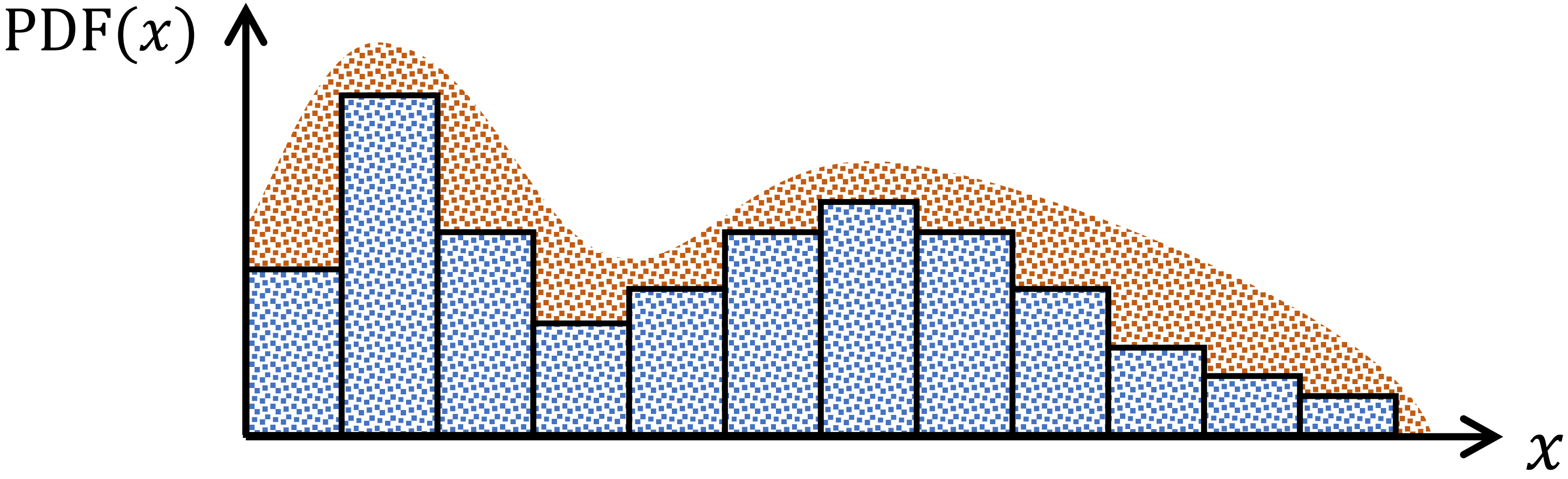
- Sampling from the area under a density function is equivalent to sampling from that density itself



Think about sampling from a histogram with even width bins and then take the width of these bins to zero

Sampling from Area Under Density

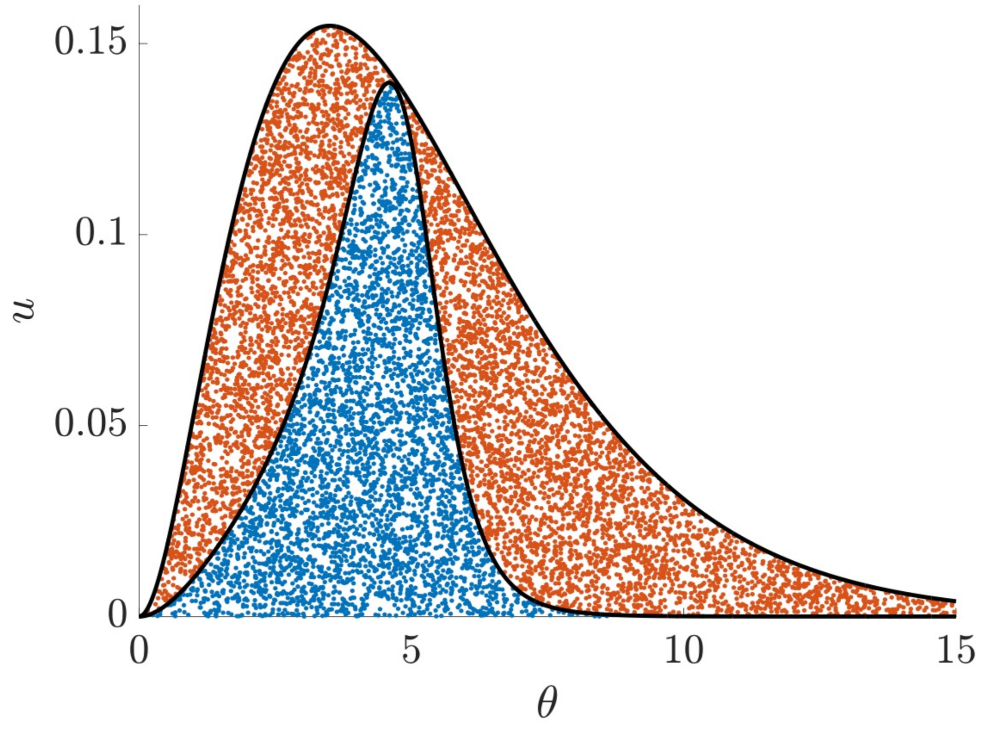
- Sampling from the area under a density function is equivalent to sampling from that density itself



Think about sampling from a histogram with even width bins and then take the width of these bins to zero

Rejection Sampling (1)

- Rejection sampling uses this idea to draw samples from a target by drawing samples from an area **enveloping** its density using an **auxiliary variable** u



Rejection Sampling (2)

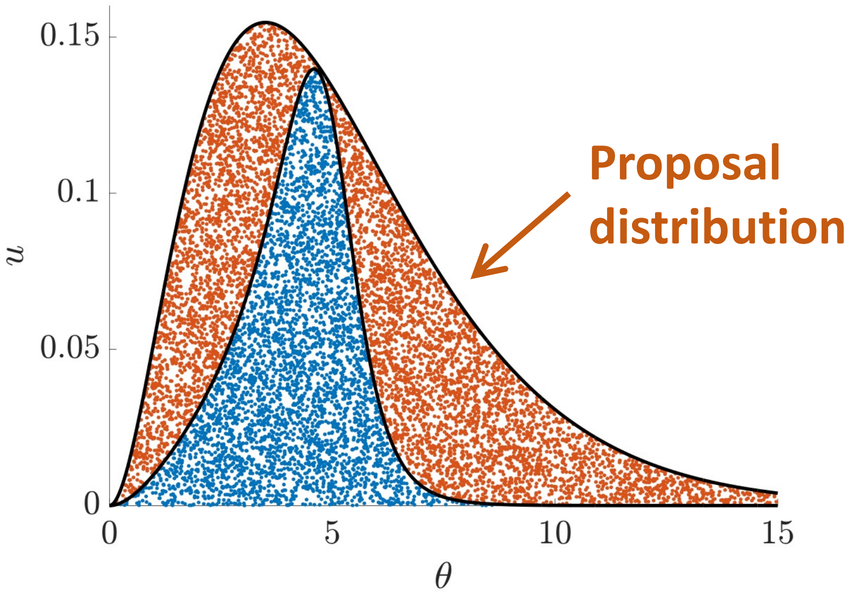
- More formally, we define a **proposal distribution** $q(\theta)$ which completely envelopes a scaled version of the unnormalized target distribution $C \cdot p(\theta)$ for some fixed C, such that $q(\theta) \geq C \cdot p(\theta|\mathcal{D})$ for all values of θ

- We then sample a pair $\{\hat{\theta}, u\}$ by first sampling $\hat{\theta} \sim q$ and then $u \sim \mathrm{Uniform}(0, q(\theta))$. Accept the sample if

$$u \leq C \cdot p(\hat{\theta}|\mathcal{D})$$

  in which case $\hat{\theta}$ is an exact sample from $p(\theta|\mathcal{D})$

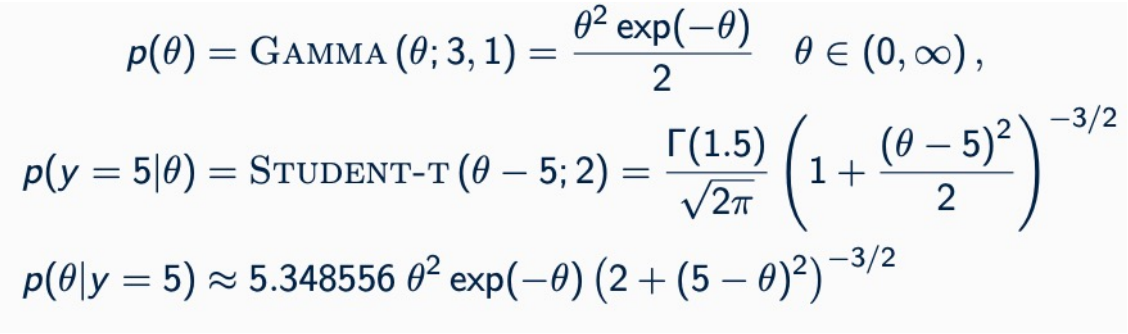
- The acceptance rate of samples is $C \cdot p(\mathcal{D})$, which thus provides an estimate for $p(\mathcal{D})$ by dividing through by C



Proposal distribution $q(\theta)$

Rejection Sampling (3)

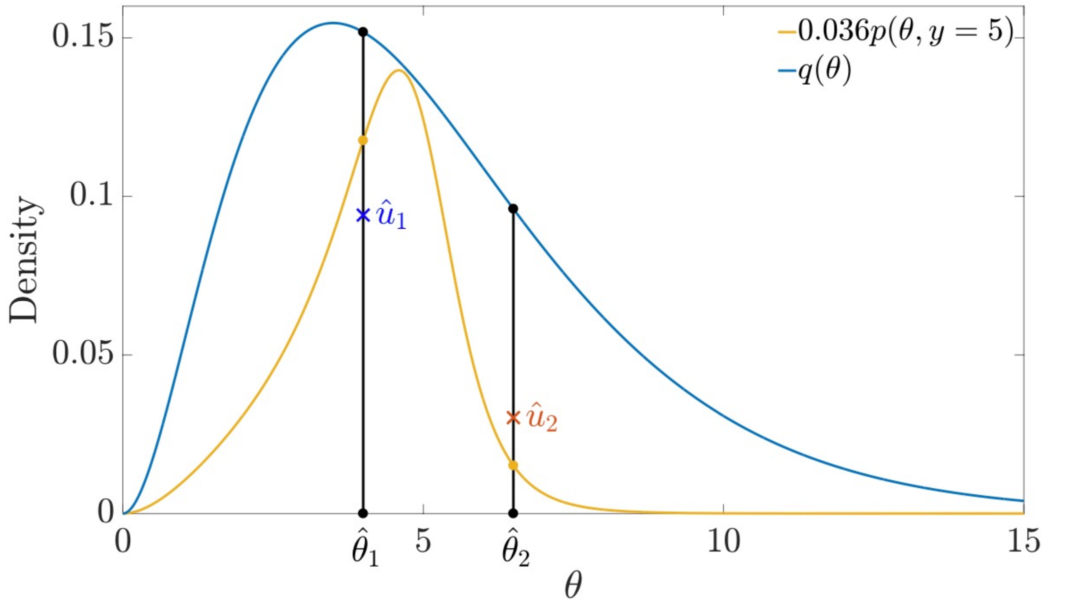
- Rejection sampling in action for our earlier example:



$$p(\theta) = \textsc{Gamma}\left(\theta; 3, 1\right) = \frac{\theta^2 \exp(-\theta)}{2} \quad \theta \in (0, \infty)\,,$$

$$p(y = 5|\theta) = \textsc{Student-t}\left(\theta - 5; 2\right) = \frac{\Gamma(1.5)}{\sqrt{2\pi}} \left(1 + \frac{(\theta - 5)^2}{2}\right)^{-3/2}$$

$$p(\theta|y = 5) \approx 5.348556\, \theta^2 \exp(-\theta) \left(2 + (5 - \theta)^2\right)^{-3/2}$$



Rejection Sampling: Pros and Cons

Pros

- One of the only inference methods to produce exact samples
- Can be highly effective in low dimensions
- Works equally well for unnormalized targets (i.e. we there is no need to know $p(\mathcal{D})$
- Provides a marginal likelihood estimate via the acceptance rate

Cons

- Scales poorly to higher dimensions (more on this later)
- Requires carefully designed proposals
- Very dependent on the value of C
- Finding a valid C requires significant knowledge about the target density

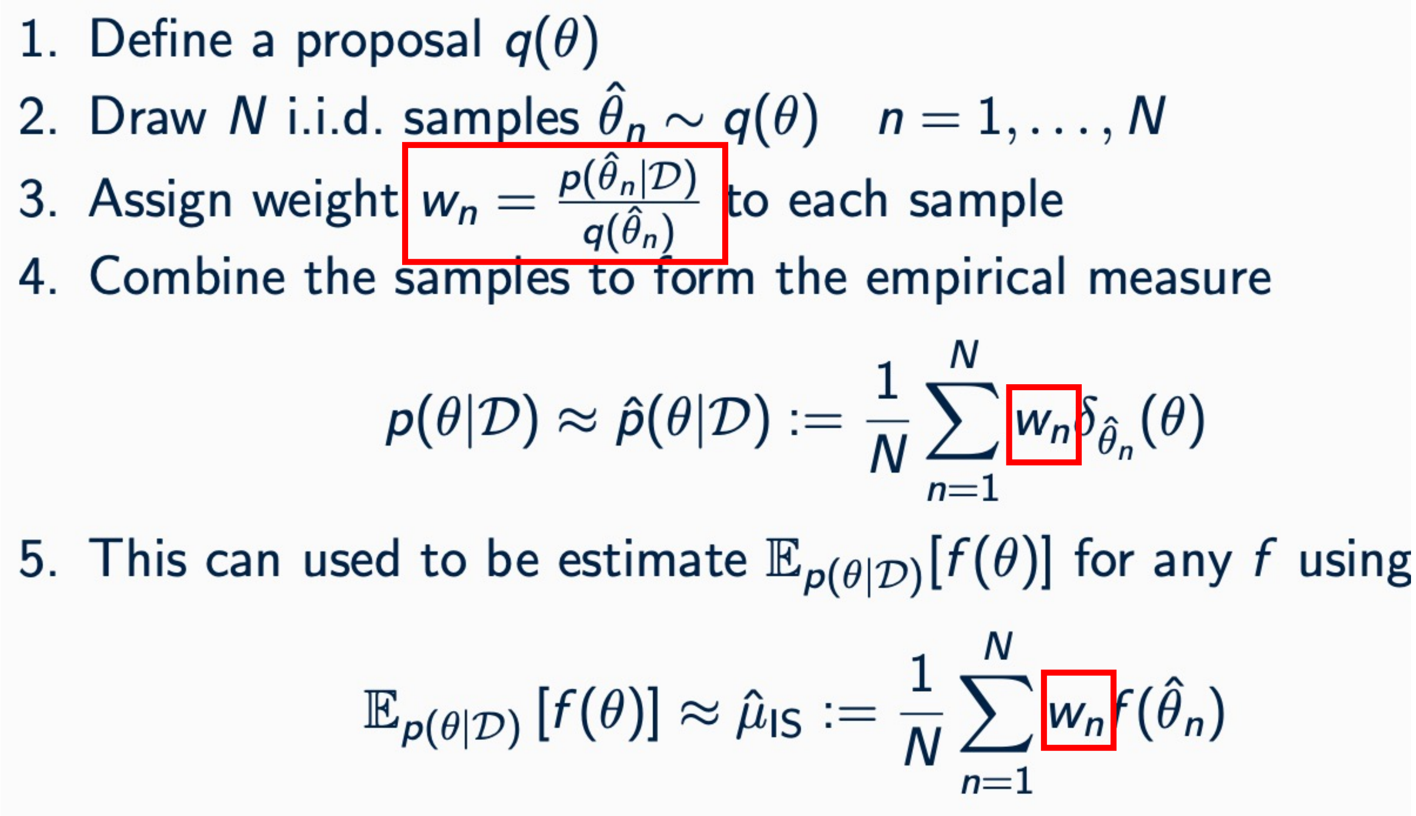
Importance Sampling

Importance Sampling

- **Importance sampling** is a common sampling method that is also the cornerstone for many more advanced inference schemes
- It is closely related to rejection sampling in that it uses a proposal, i.e. $\hat{\theta} \sim q(\theta)$
- Instead of having an accept–reject step, it assigns an **importance weight** to each sample
- These importance weights act like correction factors to account for the fact that we sampled from $q(\theta)$ rather than our target $p(\theta|\mathcal{D})$

Importance Sampling Algorithm

- Assume for now that we can evaluate $p(\theta|\mathcal{D})$ exactly. Here the algorithm is as follows:



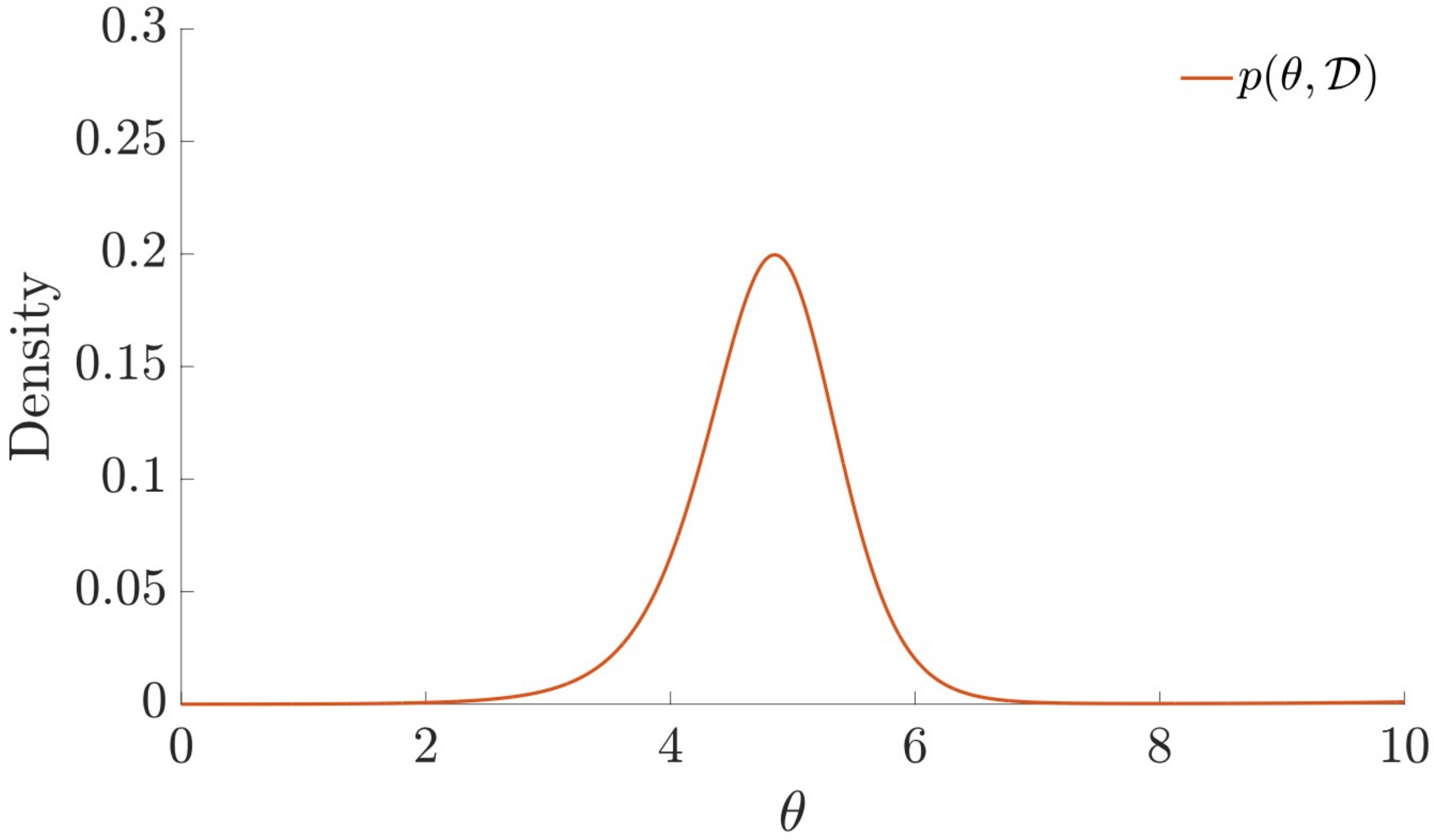
1. Define a proposal $q(\theta)$
2. Draw N i.i.d. samples $\hat{\theta}_n \sim q(\theta) \quad n = 1, \ldots, N$
3. Assign weight $\boxed{w_n = \frac{p(\hat{\theta}_n|\mathcal{D})}{q(\hat{\theta}_n)}}$ to each sample
4. Combine the samples to form the empirical measure

$$p(\theta|\mathcal{D}) \approx \hat{p}(\theta|\mathcal{D}) := \frac{1}{N}\sum_{n=1}^{N} \boxed{w_n}\delta_{\hat{\theta}_n}(\theta)$$

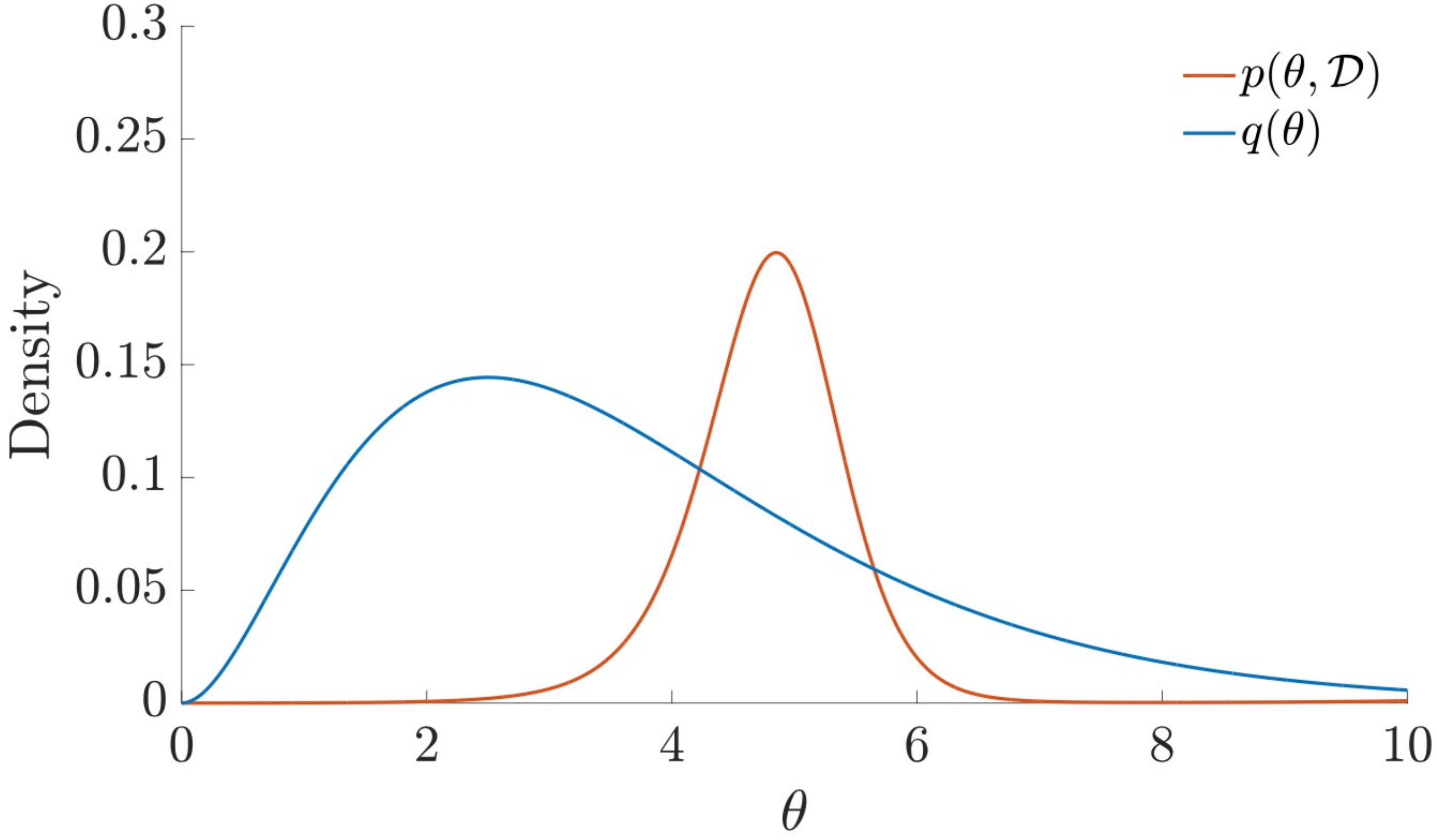
5. This can used to be estimate $\mathbb{E}_{p(\theta|\mathcal{D})}[f(\theta)]$ for any f using

$$\mathbb{E}_{p(\theta|\mathcal{D})}\left[f(\theta)\right] \approx \hat{\mu}_{\text{IS}} := \frac{1}{N}\sum_{n=1}^{N} \boxed{w_n} f(\hat{\theta}_n)$$

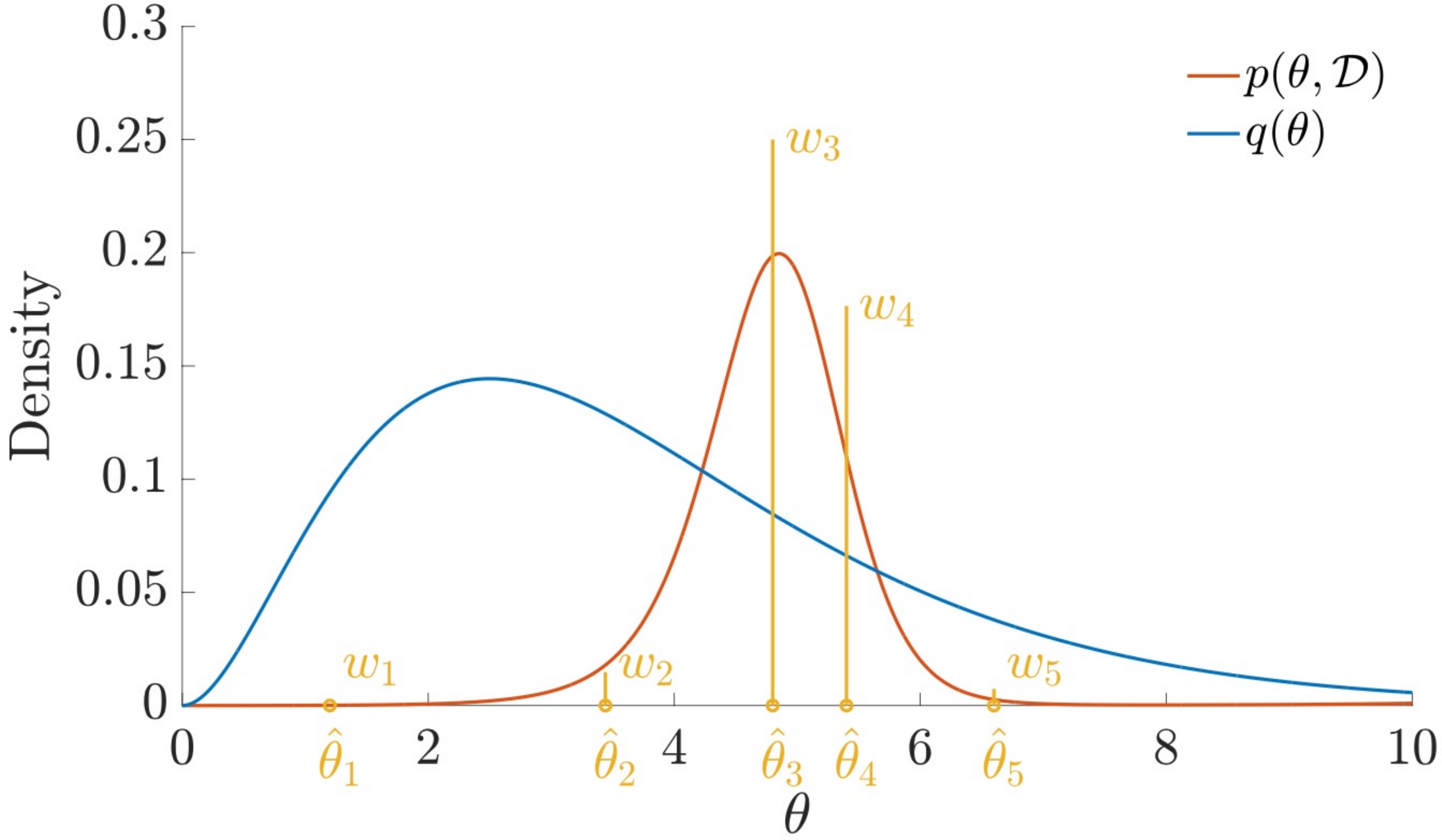
Importance Sampling Example



Importance Sampling Example

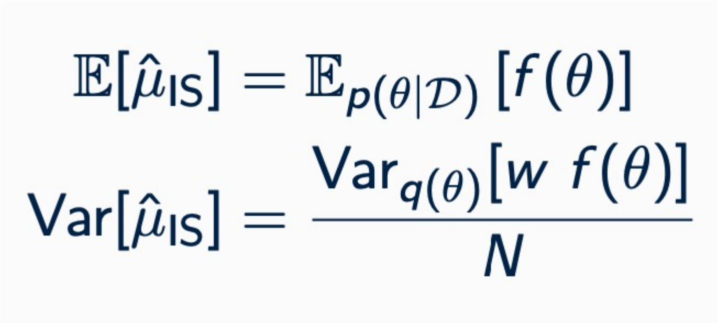


Importance Sampling Example



Importance Sampling Properties

- Provided that $q(\theta)$ has **lighter tails** than $p(\theta|\mathcal{D})$, i.e. $\frac{q(\theta)}{p(\theta|\mathcal{D})} < \varepsilon, \forall\theta$ for some $\varepsilon > 0$, then importance sampling provides an unbiased and consistent estimator for any integrable target function $f(\theta)$:



$$\mathbb{E}[\hat{\mu}_{\mathsf{IS}}] = \mathbb{E}_{p(\theta|\mathcal{D})}\left[f(\theta)\right]$$

$$\mathsf{Var}[\hat{\mu}_{\mathsf{IS}}] = \frac{\mathsf{Var}_{q(\theta)}[w\ f(\theta)]}{N}$$

(see the lecture notes for more details)

Pros and Cons of Importance Sampling

Pros

- By using all the samples from the proposal, can achieve **lower variance** estimates than rejection sampling from the same cost
- No need to find a constant scaling to bound the target (i.e. the C in rejection sampling)
- Can also be **highly effective** in low dimensions
- Provides an unbiased marginal likelihood estimate by taking the average of the weights

Cons

- Also scales poorly to higher dimensions (more on this next lecture)
- Also requires a carefully designed proposals
- Samples are not exact

Summary

- Bayesian inference is hard!
- Even if we can directly evaluate the posterior (which is rare), this may not be enough to characterize it and estimate expectations
- Monte Carlo methods give us a mechanism of representing distributions through samples
- Rejection sampling samples from an envelope of the target than only takes the samples that fall within it
- Importance sampling samples from a proposal and then assigns weights to the samples to account for them not being from the target

Next lecture: MCMC and variational methods

Further Reading

- The notes quite closely match the lecture with some extra details
- Chapters 1, 2, 7, and 9 of Art Owen's online book on Monte Carlo: https://statweb.stanford.edu/~owen/mc/
- Chapter 23 of K P Murphy. *Machine learning: a probabilistic perspective*. 2012
- M F Bugallo et al. "Adaptive importance sampling: the past, the present, and the future". In: IEEE Signal Processing Magazine (2017)
- David MacKay on Monte Carlo methods http://videolectures.net/mackay_course_12/