



THE UNIVERSITY
of EDINBURGH

Monte Carlo methods for Bayesian inference and beyond (L2)

Víctor Elvira
School of Mathematics
University of Edinburgh
(victor.elvira@ed.ac.uk)

PhD course on Bayesian filtering and Monte Carlo methods
NTU, Singapore, March–April, 2026

Outline

Bayesian inference

- Motivation: the bent coin example
- The Bayesian approach
- Probabilistic regression
- Analytically-tractable Bayesian models
- Intractable integrals

Monte Carlo methods

- Basics
 - Mini-project 1: MC
- Simulation of random variables
- Inverse transform sampling
- Importance Sampling
 - Mini-project 1 (cont)
- Rejection sampling
- Markov Chain Monte Carlo (MCMC)

Outline

Bayesian inference

- Motivation: the bent coin example
- The Bayesian approach
- Probabilistic regression
- Analytically-tractable Bayesian models
- Intractable integrals

Monte Carlo methods

- Basics
 - Mini-project 1: MC
- Simulation of random variables
- Inverse transform sampling
- Importance Sampling
 - Mini-project 1 (cont)
- Rejection sampling
- Markov Chain Monte Carlo (MCMC)

Outline

Bayesian inference

Motivation: the bent coin example

The Bayesian approach

Probabilistic regression

Analytically-tractable Bayesian models

Intractable integrals

Monte Carlo methods

Basics

Mini-project 1: MC

Simulation of random variables

Inverse transform sampling

Importance Sampling

Mini-project 1 (cont)

Rejection sampling

Markov Chain Monte Carlo (MCMC)

Coin example: Maximum Likelihood (ML) Estimation

Motivating example on the importance of being Bayesian

- You toss a (maybe bent) coin $N = 3$ times with the observed result of $Y = \mathcal{S}_3 \equiv \{H, H, T\}$ (the order is unimportant).
- What is the probability of getting H in the next toss? Denoted as $h \triangleq P(H)$
- Binomial likelihood: $p(Y|h) = h^{N_H}(1-h)^{N_T}$, with $N = N_H + N_T$
 - Maximum likelihood (ML) estimate:

$$\begin{aligned}\hat{h} &= \arg \max_h p(Y = \mathcal{S}_3|h) \\ &= \arg \max_h h^2(1-h) = \frac{2}{3}\end{aligned}$$

- What if $N = 1$, only one observation $Y = \mathcal{S}_1 \equiv \{H\}$
 - ML estimate:

$$\begin{aligned}\hat{h} &= \arg \max_h p(Y = \mathcal{S}_1|h) \\ &= \arg \max_h h = 1\end{aligned}$$

- Would you bet all your money (or honor) to a H in next toss?

Coin example: Maximum Likelihood (ML) Estimation

Motivating example on the importance of being Bayesian

- You toss a (maybe bent) coin $N = 3$ times with the observed result of $Y = \mathcal{S}_3 \equiv \{H, H, T\}$ (the order is unimportant).
- What is the probability of getting H in the next toss? Denoted as $h \triangleq P(H)$
- Binomial likelihood: $p(Y|h) = h^{N_H}(1-h)^{N_T}$, with $N = N_H + N_T$
 - Maximum likelihood (ML) estimate:

$$\begin{aligned}\hat{h} &= \arg \max_h p(Y = \mathcal{S}_3|h) \\ &= \arg \max_h h^2(1-h) = \frac{2}{3}\end{aligned}$$

- What if $N = 1$, only one observation $Y = \mathcal{S}_1 \equiv \{H\}$
 - ML estimate:

$$\begin{aligned}\hat{h} &= \arg \max_h p(Y = \mathcal{S}_1|h) \\ &= \arg \max_h h = \mathbf{1}\end{aligned}$$

- Would you bet all your money (or honor) to a H in next toss?

Coin example: Maximum a Posteriori (MAP) Estimation

- We need to add prior information: “You can’t do inference without making assumptions” [MacKay13]
- Posterior of the parameter as

$$\underbrace{p(h|Y)}_{\text{posterior}} = \frac{\underbrace{p(Y|h)}_{\text{likelihood}} \underbrace{p(h)}_{\text{prior}}}{\underbrace{p(Y)}_{\text{marginal likelihood}}} = \frac{p(Y|h)p(h)}{\int_0^1 p(Y|h)p(h)dh}$$

- With a **uniform prior** of the parameter $p(h) = \mathcal{U}([0, 1])$, the maximum a posteriori (MAP):

$$\hat{h} = \arg \max_h p(Y = \mathcal{S}_3|h)p(h) = \frac{2}{3} \quad (N = 3, \mathcal{S}_3 = \{H, H, T\})$$

$$\hat{h} = \arg \max_h p(Y = \mathcal{S}_1|h)p(h) = 1 \quad (N = 1, \mathcal{S}_1 = \{H\})$$

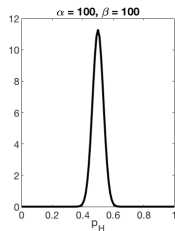
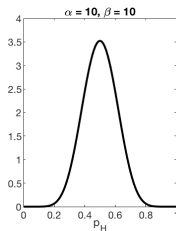
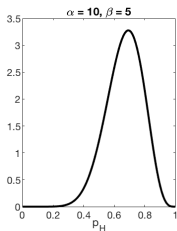
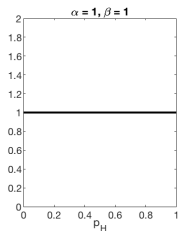
- We still obtain the same results (uninformative prior)

Coin example: The Importance of a Good Prior

- The Beta distribution is a flexible prior for parameters in $[0, 1]$.

$$p(h) = \text{Beta}(h|\alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} h^{\alpha-1} (1-h)^{\beta-1}$$

- $\alpha > 0$ and $\beta > 0$ are hyper-parameters of the prior and correspond to “one plus the pseudo-counts”
- Beta is conjugate to the binomial likelihood:
 - posterior has the same form as the prior
 - avoids intractable expressions



Coin example: Estimations are good, distributions are better

- If prior is $p(h) = \text{Beta}(h|\alpha, \beta)$ and likelihood is $B(N_H; N, h)$, the posterior is:

- $p(h|Y) = \text{Beta}(h|\alpha + N_H, \beta + N_T)$

(recall: Y contains the outcomes of tossing the coin N times)

- We set the prior $p(h) = \text{Beta}(h|10, 10)$

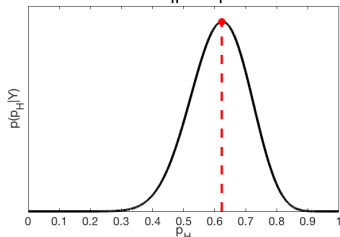
- Case 1: $N = 6$, $N_H = 6$, and $N_T = 0$

$$p(h|Y) = \text{Beta}(16, 10) \quad \Rightarrow \hat{h}^{*(MAP)} \approx 0.62$$

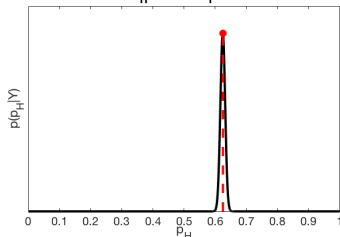
- Case 2: $N = 4000$, $N_H = 2500$, and $N_T = 1500$

$$p(h|Y) = \text{Beta}(2510, 1510) \quad \Rightarrow \hat{h}^{*(MAP)} \approx 0.62$$

$\alpha = 10, \beta = 10, N_H = 6, N_T = 0 \quad (N = 6)$



$\alpha = 10, \beta = 10, N_H = 2500, N_T = 1500 \quad (N = 4000)$



- In Case 1 (left), the coin can still be fair ($h = 0.5$).
 - MAP estimator is blind to uncertainty in the estimation

Solution \Rightarrow work with the full posterior \Rightarrow be fully Bayesian!

Coin example: Estimations are good, distributions are better

- If prior is $p(h) = \text{Beta}(h|\alpha, \beta)$ and likelihood is $B(N_H; N, h)$, the posterior is:

- $p(h|Y) = \text{Beta}(h|\alpha + N_H, \beta + N_T)$

(recall: Y contains the outcomes of tossing the coin N times)

- We set the prior $p(h) = \text{Beta}(h|10, 10)$

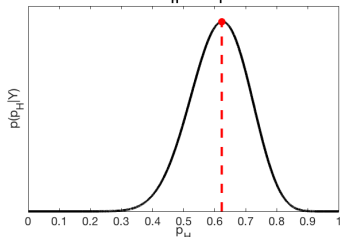
- Case 1: $N = 6$, $N_H = 6$, and $N_T = 0$

$$p(h|Y) = \text{Beta}(16, 10) \quad \Rightarrow \hat{h}^{*(MAP)} \approx 0.62$$

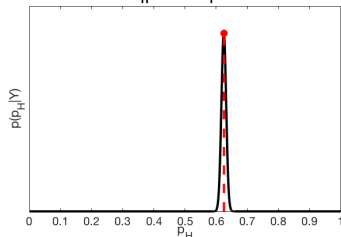
- Case 2: $N = 4000$, $N_H = 2500$, and $N_T = 1500$

$$p(h|Y) = \text{Beta}(2510, 1510) \quad \Rightarrow \hat{h}^{*(MAP)} \approx 0.62$$

$\alpha = 10, \beta = 10, N_H = 6, N_T = 0$ ($N = 6$)



$\alpha = 10, \beta = 10, N_H = 2500, N_T = 1500$ ($N = 4000$)



- In Case 1 (left), the coin can still be fair ($h = 0.5$).
 - MAP estimator is blind to uncertainty in the estimation

Solution \Rightarrow work with the full posterior \Rightarrow be fully Bayesian!

Coin example: Estimations are good, distributions are better

- If prior is $p(h) = \text{Beta}(h|\alpha, \beta)$ and likelihood is $B(N_H; N, h)$, the posterior is:

- $p(h|Y) = \text{Beta}(h|\alpha + N_H, \beta + N_T)$

(recall: Y contains the outcomes of tossing the coin N times)

- We set the prior $p(h) = \text{Beta}(h|10, 10)$

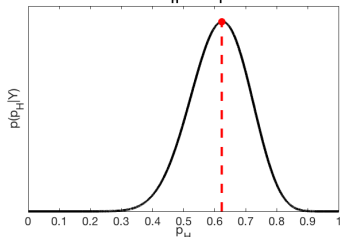
- Case 1: $N = 6$, $N_H = 6$, and $N_T = 0$

$$p(h|Y) = \text{Beta}(16, 10) \quad \Rightarrow \hat{h}^{*(MAP)} \approx 0.62$$

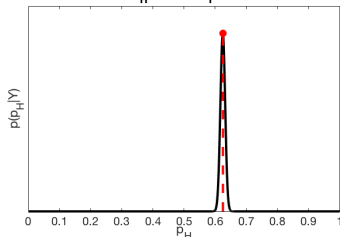
- Case 2: $N = 4000$, $N_H = 2500$, and $N_T = 1500$

$$p(h|Y) = \text{Beta}(2510, 1510) \quad \Rightarrow \hat{h}^{*(MAP)} \approx 0.62$$

$\alpha = 10, \beta = 10, N_H = 6, N_T = 0 \quad (N = 6)$



$\alpha = 10, \beta = 10, N_H = 2500, N_T = 1500 \quad (N = 4000)$



- In Case 1 (left), the coin can still be fair ($h = 0.5$).
 - MAP estimator is blind to uncertainty in the estimation

Solution \Rightarrow work with the full posterior \Rightarrow be **fully Bayesian!**

Outline

Bayesian inference

Motivation: the bent coin example

The Bayesian approach

Probabilistic regression

Analytically-tractable Bayesian models

Intractable integrals

Monte Carlo methods

Basics

Mini-project 1: MC

Simulation of random variables

Inverse transform sampling

Importance Sampling

Mini-project 1 (cont)

Rejection sampling

Markov Chain Monte Carlo (MCMC)

Bayesian inference

- We model uncertainty about parameters β probabilistically
- Combine prior knowledge and data:

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

- Ingredients:
 - likelihood $p(\mathcal{D}|\beta)$: how the data is generated given the parameters
 - prior $p(\beta)$: beliefs about plausible parameter values before seeing data
- The marginal likelihood (evidence):

$$p(\mathcal{D}) = \int p(\mathcal{D}|\beta)p(\beta) d\beta$$

- normalizes the posterior
 - key for model comparison
 - often intractable
- Workflow:
 - specify model and prior
 - observe data
 - compute posterior
 - use it for prediction and decisions

Bayesian inference: challenges

- In simple cases, the posterior is available in closed form
- In most realistic models:
 - the evidence $p(\mathcal{D}) = \int p(\mathcal{D}|\beta)p(\beta)d\beta$ is intractable
 - expectations w.r.t. the posterior are intractable
- Example:

$$\mathbb{E}[\beta|\mathcal{D}] = \int \beta p(\beta|\mathcal{D}) d\beta$$

- Solution: approximate the posterior
 - variational methods
 - Monte Carlo methods

Bayesian inference: challenges

- In realistic models, the posterior is rarely available in closed form
- Main difficulties:
 - marginal likelihood (evidence)

$$p(\mathcal{D}) = \int p(\mathcal{D}|\beta)p(\beta) d\beta$$

is typically intractable

- expectations w.r.t. the posterior are also intractable

$$\int f(\beta) p(\beta|\mathcal{D}) d\beta$$

- As a result, exact Bayesian inference is usually impossible
- Alternatives:
 - MAP: point estimate (no uncertainty)
 - **Monte Carlo**: approximate the posterior with samples
- With Monte Carlo, we can:
 - approximate expectations
 - quantify uncertainty
 - make predictions

Outline

Bayesian inference

Motivation: the bent coin example

The Bayesian approach

Probabilistic regression

Analytically-tractable Bayesian models

Intractable integrals

Monte Carlo methods

Basics

Mini-project 1: MC

Simulation of random variables

Inverse transform sampling

Importance Sampling

Mini-project 1 (cont)

Rejection sampling

Markov Chain Monte Carlo (MCMC)

Probabilistic view of regularized regression

- Linear regression model:

$$y_i \mid \mathbf{x}_i, \boldsymbol{\beta} \sim \mathcal{N} \left(\boldsymbol{\beta}^T \boldsymbol{\phi}(\mathbf{x}_i), \sigma^2 \right)$$

- Likelihood:

$$p(\mathcal{D} \mid \boldsymbol{\beta}) = \prod_{i=1}^N \mathcal{N} \left(y_i \mid \boldsymbol{\beta}^T \boldsymbol{\phi}(\mathbf{x}_i), \sigma^2 \right)$$

- Place a prior on the parameters:
 - Gaussian prior $\Rightarrow p(\boldsymbol{\beta}) \propto \exp(-\lambda \|\boldsymbol{\beta}\|_2^2)$
 - Laplace prior $\Rightarrow p(\boldsymbol{\beta}) \propto \exp(-\lambda \|\boldsymbol{\beta}\|_1)$
- MAP estimator:

$$\hat{\boldsymbol{\beta}} = \arg \max_{\boldsymbol{\beta}} p(\boldsymbol{\beta} \mid \mathcal{D}) = \arg \max_{\boldsymbol{\beta}} p(\mathcal{D} \mid \boldsymbol{\beta}) p(\boldsymbol{\beta})$$

- Equivalent optimization problem (and solution):

$$\hat{\boldsymbol{\beta}} = \arg \min_{\boldsymbol{\beta}} -\log p(\mathcal{D} \mid \boldsymbol{\beta}) + \lambda \|\boldsymbol{\beta}\|_{\ell}$$

- $\ell = 2 \Rightarrow$ ridge
 - $\ell = 1 \Rightarrow$ LASSO
- Not fully Bayesian: we keep a single point instead of the full posterior

Predictive posterior distribution

- Point estimates (e.g., MAP) do not quantify uncertainty
- Bayesian prediction for a new input \mathbf{x}^* :

$$p(y^* | \mathbf{x}^*, \mathcal{D}) = \int p(y^* | \mathbf{x}^*, \boldsymbol{\beta}) p(\boldsymbol{\beta} | \mathcal{D}) d\boldsymbol{\beta}$$

with $p(\boldsymbol{\beta} | \mathcal{D}) = \frac{p(\mathcal{D} | \boldsymbol{\beta}) p(\boldsymbol{\beta})}{p(\mathcal{D})}$ obtained at training.

- We average over all plausible parameter values with associated probabilities
- This provides:
 - uncertainty quantification
 - full probabilistic predictions

Inference at a Higher Level: Comparing Models

- Let us re-think our assumptions:
 - We've assumed a model M_1 (e.g., polynomial linear regression with $m = 5$).
 - What if we're unsure which model is right? (**remember, all are wrong**)
- We can compare models based on the *marginal likelihood* (aka, the evidence) for each model, which is the probability the model assigns to the observed data (denominator in Bayes).

$$p(\mathcal{D}|M_1) = \int p(\mathcal{D}|\boldsymbol{\theta}, M_1)p(\boldsymbol{\theta}|M_1)d\boldsymbol{\theta}$$

- Here, M_1 represents the condition that model M_1 is the correct one (which previously we silently assumed).
- Similarly, we can compute $p(\mathcal{D}|M_k)$, for some other K models (which may have a different parameter space).
 - **Option 1.** Choose the model that gives higher probability to the data
 - **Option 2.** Average predictions from all models with weights proportional to the marginal likelihood times the model prior $p(M_j)$ (preference we have for each model).

$$\begin{aligned} p(y^*|\mathbf{x}^*, \mathcal{D}) &= \sum_{i=1}^K p(y^*, M_i|\mathbf{x}^*, \mathcal{D}) = \sum_{i=1}^K p(y^*|\mathbf{x}^*, \mathcal{D}, M_i)p(M_i|\mathbf{x}^*, \mathcal{D}) \\ &= \sum_{i=1}^K \underbrace{p(y^*|\mathbf{x}^*, \mathcal{D}, M_i)}_{\text{predictive of } i\text{-th model}} \cdot \underbrace{p(M_i|\mathcal{D})}_{\text{posterior of } i\text{-th model}} \end{aligned}$$

with

$$p(M_i|\mathcal{D}) = \frac{p(\mathcal{D}|M_i)p(M_i)}{\sum_{j=1}^K p(\mathcal{D}|M_j)p(M_j)}$$

Outline

Bayesian inference

Motivation: the bent coin example

The Bayesian approach

Probabilistic regression

Analytically-tractable Bayesian models

Intractable integrals

Monte Carlo methods

Basics

Mini-project 1: MC

Simulation of random variables

Inverse transform sampling

Importance Sampling

Mini-project 1 (cont)

Rejection sampling

Markov Chain Monte Carlo (MCMC)

Analytically-tractable Bayesian models

- In some cases, Bayesian inference is analytically tractable
- This typically happens with **conjugate priors**:
 - posterior has the same form as the prior
- Examples:
 1. Gaussian likelihood with Gaussian prior on the mean (known variance, or inverse-Gamma prior on the variance)
 2. Linear regression with Gaussian prior on coefficients and Gaussian noise (known variance, or inverse-Gamma prior)
- These cases are useful, but limited
- In general, we need approximate methods (e.g., Monte Carlo)
- References:
 - Robert, C. (2007). The Bayesian choice: from decision-theoretic foundations to computational implementation. Springer Science & Business Media.
 - Murphy, K. P. (2012). Machine learning: a probabilistic perspective. MIT
 - Barber, D. (2012). Bayesian reasoning and machine learning. Cambridge University Press.

Gaussian properties

- **Univariate Gaussian distribution:**

$$f_X(x) = \mathcal{N}(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

- **Multivariate Gaussian distribution:** Example in 2D, with 2 Gaussian r.v.'s x_1 and x_2 in $\mathbf{x} = [x_1, x_2]^T \in \mathbb{R}^{2 \times 1}$

$$f_X(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\det(2\pi\boldsymbol{\Sigma})^{-\frac{1}{2}}} e^{(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}$$

where $\boldsymbol{\mu} \in \mathbb{R}^{2 \times 1}$ is the mean and $\boldsymbol{\Sigma} \in \mathbb{R}^{2 \times 2}$ is the covariance matrix.

1. **Marginalization** of a joint Gaussian distribution is still **Gaussian**:

$$p(x_1) = \int p(x_2, x_1) dx_2,$$

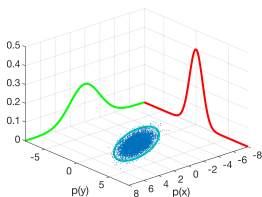
marginalizing x_2 , $p(x_1)$ is also Gaussian

2. **Conditional** of a joint Gaussian distribution is still **Gaussian** (equivalent to first point):

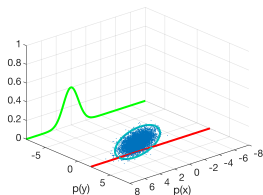
$$p(x_1|x_2) = \frac{p(x_1, x_2)}{p(x_2)}.$$

Gaussian properties (cont)

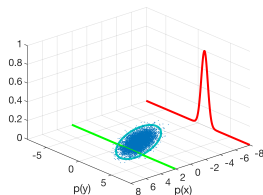
1. Marginals of a bi-variate Gaussian distribution are **Gaussian**:



2. Conditionals of a bi-variate Gaussian distribution are **Gaussian**:



$$p(x_1|x_2 = 2)$$



$$p(x_2|x_1 = 2)$$

Gaussian Example: Tractable Bayesian Linear Regression

- Linear-Gaussian model:

$$y_i | \mathbf{x}_i; \boldsymbol{\beta} \sim \mathcal{N} \left(y; \mathbf{x}_i^T \boldsymbol{\beta}, \sigma^2 \right), \quad i = 1, \dots, N$$

$$\boldsymbol{\beta} \sim \mathcal{N} (\boldsymbol{\beta}; m_0, S_0)$$

- we consider σ^2 , m_0 , and S_0 as known.
- The observation model is equivalent to the previous one (**very important to see this duality**), i.e., $y_i = \mathbf{x}_i^T \boldsymbol{\beta} + \varepsilon$, with $\varepsilon \sim \mathcal{N}(0, \sigma^2)$.
- This Gaussian prior model will turn out to be conjugate, which means that **the posterior is analytical/exact/tractable**

$$p(\boldsymbol{\beta} | \mathcal{D}) = \mathcal{N} (\boldsymbol{\beta}; m_P, S_P)$$

with

$$m_P = S_0 \mathbf{X}^T \left(\mathbf{X} S_0 \mathbf{X}^T + \sigma^2 \mathbf{I} \right)^{-1} (y - \mathbf{X} m_0)$$

$$S_P = S_0 - S_0 \mathbf{X}^T \left(\mathbf{X} S_0 \mathbf{X}^T + \sigma^2 \mathbf{I} \right)^{-1} \mathbf{X} S_0$$

- Exercise: compare this solution with the one of linear regression in L1, with $m_0 = 0$ and identity S_0 .

Outline

Bayesian inference

- Motivation: the bent coin example
- The Bayesian approach
- Probabilistic regression
- Analytically-tractable Bayesian models
- Intractable integrals**

Monte Carlo methods

- Basics
 - Mini-project 1: MC
- Simulation of random variables
- Inverse transform sampling
- Importance Sampling
 - Mini-project 1 (cont)
- Rejection sampling
- Markov Chain Monte Carlo (MCMC)

Intractable Integrals

- The previous examples are ideal, but uncommon
 - the posterior often has no closed-form expression
 - the normalizing constant $p(\mathcal{D})$ is intractable
- The full Bayesian approach requires integration over the posterior:
 - e.g., posterior mean

$$\mu_{\beta|\mathcal{D}} = \int \beta p(\beta|\mathcal{D}) d\beta$$

- these integrals are typically intractable
- Approximation methods:
 - grid-based methods
 - * evaluate the integral at deterministic points
 - * limited to low-dimensional problems
 - variational methods
 - * approximate the posterior within a parametric family
 - * optimization-based
 - * not asymptotically exact
 - Monte Carlo methods
 - * sampling-based (stochastic)
 - * asymptotically exact

Outline

Bayesian inference

- Motivation: the bent coin example
- The Bayesian approach
- Probabilistic regression
- Analytically-tractable Bayesian models
- Intractable integrals

Monte Carlo methods

- Basics
 - Mini-project 1: MC
- Simulation of random variables
- Inverse transform sampling
- Importance Sampling
 - Mini-project 1 (cont)
- Rejection sampling
- Markov Chain Monte Carlo (MCMC)

Outline

Bayesian inference

- Motivation: the bent coin example
- The Bayesian approach
- Probabilistic regression
- Analytically-tractable Bayesian models
- Intractable integrals

Monte Carlo methods

Basics

- Mini-project 1: MC
- Simulation of random variables
- Inverse transform sampling
- Importance Sampling
 - Mini-project 1 (cont)
- Rejection sampling
- Markov Chain Monte Carlo (MCMC)

Monte Carlo methods: a bit of history

- A methodology that comes to the rescue for solving most difficult problems of inference is based on **drawing/simulation** of samples.
 - e.g., in Bayesian inference, for most of models of interest, it is usually **impossible** to find posteriors distributions nor simulate from them.
- The Monte Carlo methods were born in Los Alamos, New Mexico (USA) in the 1940s around the Manhattan project
 - associated to the Electronic Numerical Integrator and Computer (**ENIAC**), one of the first electronic general-purpose computers.
- Foundational works of Stanislaw Ulam (1909-1984) and John von Neumann (1903-1957)
 - they invented the **inversion** and **accept-reject** techniques
 - independently developed by Enrico Fermi (1901-1954)
 - beautiful story at¹
- Led to the **Metropolis algorithm** by Nicolas Metropolis in 1953
 - the first **Markov chain Monte Carlo (MCMC)** algorithm
 - listed among the “10 algorithms with the greatest influence on the development of science and engineering in the 20th century”, by the American Institute of Physics and the IEEE Computer Society in 2000.

<https://poole.ncsu.edu/thought-leadership/article/oppenheimer-ulam-and-risk-analytics-the-legacy-of-wwii-scientists-on-contemporary-computing/>

¹N. Metropolis et al. “The beginning of the Monte Carlo method”. In: *Los Alamos Science* 15.584 (1987), pp. 125–130.

Monte Carlo basics

- **Disclaimer:** in Monte Carlo methods, \mathbf{x} represents the variable to be integrated (not the observed input of supervised learning).
- In real-world problems, we need to solve integrals that have **no closed-form solution**:

$$I(f) \equiv \mathbb{E}_{\tilde{\pi}(\mathbf{x})}[f(\mathbf{x})] = \int f(\mathbf{x})\tilde{\pi}(\mathbf{x})d\mathbf{x}$$

- Monte Carlo methods:
 - approximate distributions by N random samples (or particles)
 - in that way, the **integrals** are approximated by **sums**
- Suppose we can draw N samples from $\tilde{\pi}(\mathbf{x})$, i.e.

$$\mathbf{x}^{(n)} \sim \tilde{\pi}(\mathbf{x}), \quad n = 1, \dots, N.$$

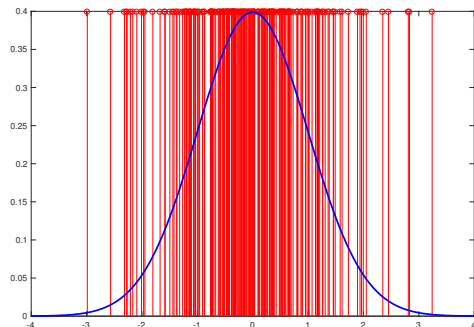
- Then, we can approximate the targeted distribution as

$$\tilde{\pi}(\mathbf{x}) \approx \tilde{\pi}^N(\mathbf{x}) = \frac{1}{N} \sum_{n=1}^N \delta_{\mathbf{x}^{(n)}}(\mathbf{x})$$

Monte Carlo basics

- $\tilde{\pi}(\mathbf{x}) \approx \tilde{\pi}^N(\mathbf{x}) = \frac{1}{N} \sum_{n=1}^N \delta_{\mathbf{x}^{(n)}}(\mathbf{x})$
where $\mathbf{x}^{(n)} \sim \tilde{\pi}(\mathbf{x}), n = 1, \dots, N$.

- Example: $\tilde{\pi}(\mathbf{x}) \equiv \mathcal{N}(0, 1) \approx \tilde{\pi}^N(\mathbf{x})$ with $N = 200$.



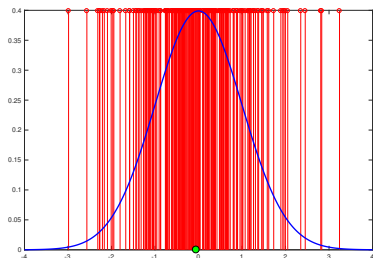
Monte Carlo basics

- We can approximate any integral of $\tilde{\pi}(\mathbf{x})$: **integral + dirac = sum!**

$$\begin{aligned} I(f) \equiv \mathbb{E}_{\tilde{\pi}(\mathbf{x})}[f(\mathbf{x})] &= \int f(\mathbf{x})\tilde{\pi}(\mathbf{x})d\mathbf{x} \approx \int f(\mathbf{x})\tilde{\pi}^N(\mathbf{x})d\mathbf{x} \approx \int f(\mathbf{x}) \left(\frac{1}{N} \sum_{n=1}^N \delta_{\mathbf{x}^{(n)}}(\mathbf{x}) \right) d\mathbf{x} \\ &\approx \frac{1}{N} \sum_{n=1}^N \int f(\mathbf{x})\delta_{\mathbf{x}^{(n)}}(\mathbf{x})d\mathbf{x} \\ &= \frac{1}{N} \sum_{n=1}^N f(\mathbf{x}^{(n)}) \equiv \bar{I}^N(f) \end{aligned}$$

- e.g. the mean of the distribution ($f(\mathbf{x}) = \mathbf{x}$) approximated by $N = 200$.

$$\hat{\mathbf{x}} = \mathbb{E}_{\tilde{\pi}(\mathbf{x})}[\mathbf{x}] = \int \mathbf{x}\tilde{\pi}(\mathbf{x})d\mathbf{x} \approx \frac{1}{N} \sum_{n=1}^N \mathbf{x}^{(n)} = -0.0561$$



- Summary: the integral

$$I(f) \equiv \mathbb{E}_{\tilde{\pi}(\mathbf{x})}[f(\mathbf{x})] = \int f(\mathbf{x})\tilde{\pi}(\mathbf{x})d\mathbf{x}$$

is approximated by

$$\bar{I}_N(f) = \frac{1}{N} \sum_{n=1}^N f(\mathbf{x}^{(n)})$$

- It converges to the right quantity, and with a known rate $1/\sqrt{N}$:
 1. Due to the strong law of large numbers:

$$\bar{I}_N(f) \rightarrow I(f), \quad N \rightarrow \infty$$

2. Due to the central limit theorem:

$$\frac{\sqrt{N}(\bar{I}_N(f) - I(f))}{\sigma_f} \rightarrow \mathcal{N}(0, 1), \quad N \rightarrow \infty$$

Mini-project 1: basic/direct/raw Monte Carlo

- Simulate $N = 20$ samples from a standard normal (zero mean, unit variance) distribution (you can later play with different N).
 - estimate the mean ($f(x) = x$), and the $\Pr(X > \gamma)$, e.g., with $\gamma = 3$ ($f(x) = \mathbb{I}_{|x| > \gamma}(x)$).
 - characterize the estimator (variance, bias, and MSE): you will have to run the estimator *many* times and then take empirical means.

Outline

Bayesian inference

- Motivation: the bent coin example
- The Bayesian approach
- Probabilistic regression
- Analytically-tractable Bayesian models
- Intractable integrals

Monte Carlo methods

Basics

Mini-project 1: MC

Simulation of random variables

Inverse transform sampling

Importance Sampling

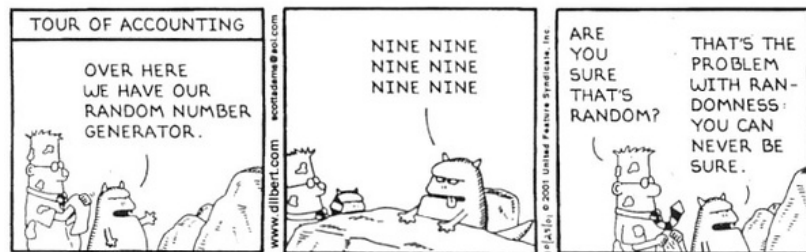
Mini-project 1 (cont)

Rejection sampling

Markov Chain Monte Carlo (MCMC)

Random numbers

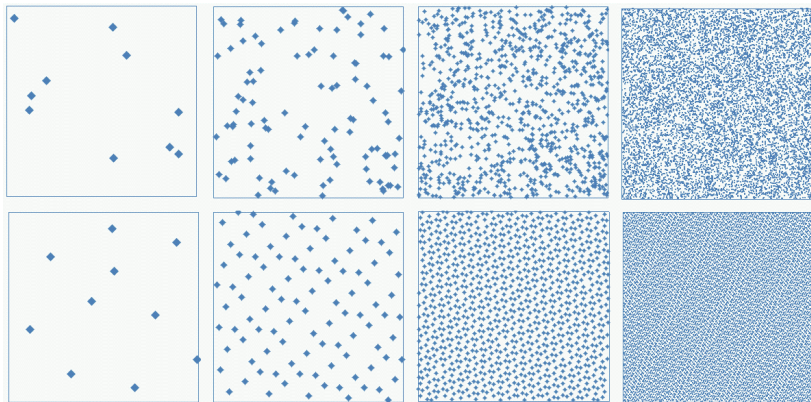
DILBERT By SCOTT ADAMS



Simulation of random variables

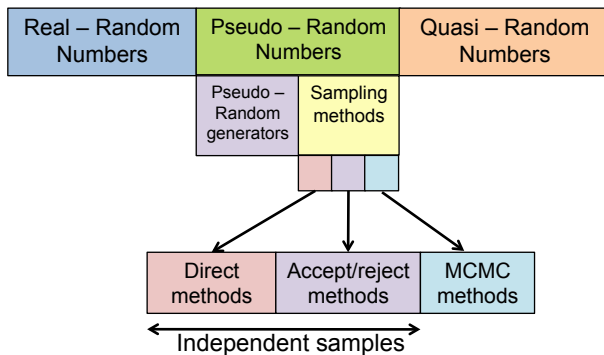
- Solving the previous problem requires the simulation of random numbers according to some distribution (we called it $\tilde{\pi}(\mathbf{x})$).
1. **True** random numbers are generated using a (well-studied) physical device as the source of randomness.
 - e.g., coin flipping, roulette wheels, white noise, and the count of particles emitted by a radioactive source.
 2. **Pseudo**-random numbers correspond to a deterministic sequence that passes tests of randomness, generally generated by a computer. In two steps:
 - 2.1 generation of imitations of independent and identically distributed (i.i.d.) random numbers having a uniform distribution, and
 - 2.2 application of some transformation and/or selection techniques such that these i.i.d. uniform samples are converted into variates from the target probability distribution.
 3. **Quasi**-random numbers correspond to a deterministic sequence that **does not** pass tests of randomness.

Quasi-random numbers



Up. Uniform random numbers. **Bottom.** Uniform quasi random numbers. [Source: Wikipedia]

Random numbers



Outline

Bayesian inference

- Motivation: the bent coin example
- The Bayesian approach
- Probabilistic regression
- Analytically-tractable Bayesian models
- Intractable integrals

Monte Carlo methods

- Basics
 - Mini-project 1: MC
- Simulation of random variables
- Inverse transform sampling**
- Importance Sampling
 - Mini-project 1 (cont)
- Rejection sampling
- Markov Chain Monte Carlo (MCMC)

Inverse transform sampling

- The **inverse transform sampling** method uses uniform samples for simulating the r.v. $X \sim p_X(x)$, with CDF $F_X(x)$
- The method reads as follows:
 1. generate a random number $U \sim \mathcal{U}(0, 1)$
 2. find the inverse of the CDF $F_X(x)$, i.e., $F_X^{-1}(x)$.
 3. compute $X = F_X^{-1}(U)$, which follows $p_X(x)$
- the only requirement is that the CDF $F_X(x)$ must be invertible
- Proof: the CDF of X obtained as in point 3 is:

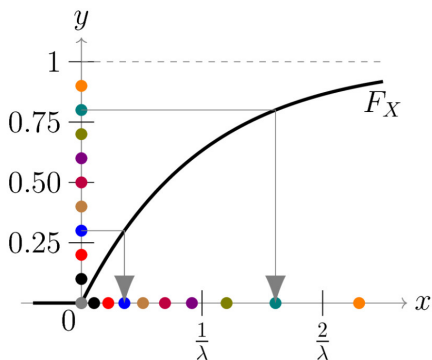
$$\begin{aligned}\Pr(X \leq x) &= \Pr(F_X^{-1}(U) \leq x) \\ &= \Pr(U \leq F_X(x)) \\ &= F_X(x),\end{aligned}$$

since $\Pr(U \leq y) = y$.



Inverse transform sampling: Example

- Example: We want samples from the exponential distribution $p_X(x) = \lambda e^{-\lambda x}$, which yields $F_X(x) = 1 - e^{-\lambda x}$.
 1. generate a random number $U \sim \mathcal{U}(0, 1)$
 2. the inverse of $F_X(x)$ is $x = F_X^{-1}(u) = -\frac{1}{\lambda} \log(1 - u)$
 3. $X = F_X^{-1}(U) = -\frac{1}{\lambda} \log(1 - U)$.



[Source: Wikipedia]

Outline

Bayesian inference

- Motivation: the bent coin example
- The Bayesian approach
- Probabilistic regression
- Analytically-tractable Bayesian models
- Intractable integrals

Monte Carlo methods

- Basics
 - Mini-project 1: MC
- Simulation of random variables
- Inverse transform sampling
- Importance Sampling**
 - Mini-project 1 (cont)
- Rejection sampling
- Markov Chain Monte Carlo (MCMC)

Importance Sampling basics

- Another method for drawing samples or approximating integrals when raw Monte Carlo is not possible
 - **Importance sampling (IS)** is a Monte Carlo method that allows to approximate integrals over complicated distributions.
- Same problem:

$$I(f) \equiv \mathbb{E}_{\tilde{\pi}(\mathbf{x})}[f(\mathbf{x})] = \int f(\mathbf{x})\tilde{\pi}(\mathbf{x})d\mathbf{x} = \int f(\mathbf{x})\frac{\tilde{\pi}(\mathbf{x})}{q(\mathbf{x})}q(\mathbf{x})d\mathbf{x}$$

where $q(\mathbf{x})$ is the proposal density where the samples are now drawn

$$\text{Raw MC estimator: } \bar{I}_N(f) = \frac{1}{N} \sum_{n=1}^N f(\mathbf{x}^{(n)}), \quad \mathbf{x}^{(n)} \sim \tilde{\pi}(\mathbf{x})$$

$$\text{Basic IS estimator: } \hat{I}_N(f) = \frac{1}{N} \sum_{n=1}^N f(\mathbf{x}^{(n)}) \frac{\tilde{\pi}(\mathbf{x}^{(n)})}{q(\mathbf{x}^{(n)})}, \quad \mathbf{x}^{(n)} \sim q(\mathbf{x})$$

Importance Sampling basics

- Basic or aka unnormalized IS (UIS) estimator:

$$\hat{I}_N(f) = \frac{1}{N} \sum_{n=1}^N f(\mathbf{x}^{(n)}) \frac{\tilde{\pi}(\mathbf{x}^{(n)})}{q(\mathbf{x}^{(n)})}, \quad \mathbf{x}^{(n)} \sim q(\mathbf{x})$$

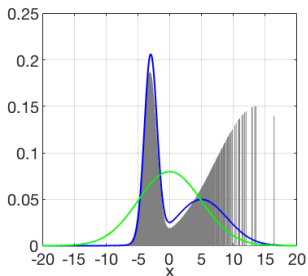
- $w^{(n)} = \frac{\tilde{\pi}(\mathbf{x}^{(n)})}{q(\mathbf{x}^{(n)})}$, $n = 1, \dots, N$, are the **importance weights**
- only constraint: $\tilde{\pi}(\mathbf{x})$ must be evaluated
- Unfortunately, sometimes we have only access to $\pi(\mathbf{x})$, where $\tilde{\pi}(\mathbf{x}) = \frac{\pi(\mathbf{x})}{Z}$, and Z is the normalizing constant (marginal likelihood / model evidence in Bayesian inference) \Rightarrow basic IS estimator is not possible.
 - Alternative, $w^{(n)} = \frac{\pi(\mathbf{x}^{(n)})}{q(\mathbf{x}^{(n)})}$
- Self-normalized IS (SNIS) estimator:

$$\tilde{I}_N(f) = \sum_{n=1}^N f(\mathbf{x}^{(n)}) \bar{w}^{(n)}, \quad \mathbf{x}^{(n)} \sim q(\mathbf{x})$$

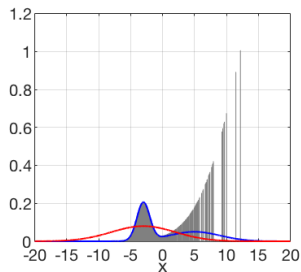
where $\bar{w}^{(n)} = \frac{w^{(n)}}{\sum_{j=1}^N w^{(j)}}$ are the normalized weights ($\sum_{j=1}^N \bar{w}^{(j)} = 1$).

Importance Sampling: example

- IS simulation:
 - Target: bi-modal density (sum of two Gaussians)
 - Proposal density
 - $N = 1000$ weighted samples



Good proposal $q_1 = \mathcal{N}(0, 25)$



Bad proposal $q_2 = \mathcal{N}(-3, 25)$

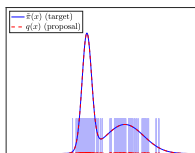
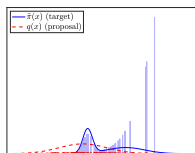
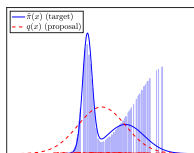
- The approximation converges in both cases, but a good proposal is key for the efficiency of IS.

The variance in IS and the need of better proposals

- A **good proposal** $q(\mathbf{x})$ is key for the efficiency of IS.
- Variance of the UIS estimator of $I = \int f(\mathbf{x})\pi(\mathbf{x})d\mathbf{x}$:

$$\text{Var}_{\pi(\mathbf{x})}(\widehat{I}) = \frac{1}{N} \int \frac{f^2(\mathbf{x})\pi^2(\mathbf{x})}{q(\mathbf{x})} d\mathbf{x} - \frac{I^2}{N}$$

- optimal UIS proposal: $q(\mathbf{x}) \propto |f(\mathbf{x})|\pi(\mathbf{x})$
- for a generic $f(\mathbf{x})$, $q(\mathbf{x})$ should be as *close* as possible to $\pi(\mathbf{x})$



- Very difficult to find a good $q(\mathbf{x})$ a priori:
 - $\pi(\mathbf{x})$ can be only evaluated (up to a normalizing constant)
 - $\pi(\mathbf{x})$ may be multimodal, skewed, heavy tailed
- A posteriori metric: $\widehat{\text{ESS}} = \frac{1}{\sum_{n=1}^N \widehat{w}_n^2}$, although it presents serious problems²
- Use **multiple proposals (MIS)** and explore the space (**AIS**).

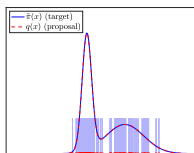
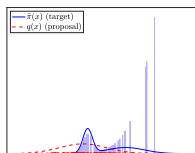
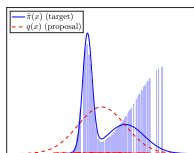
²V. Elvira, L. Martino, and C. P. Robert. "Rethinking the effective sample size". In: *International Statistical Review* 90.3 (2022), pp. 525–550.

The variance in IS and the need of better proposals

- A **good proposal** $q(\mathbf{x})$ is key for the efficiency of IS.
- Variance of the UIS estimator of $I = \int f(\mathbf{x})\pi(\mathbf{x})d\mathbf{x}$:

$$\text{Var}_{\pi(\mathbf{x})}(\widehat{I}) = \frac{1}{N} \int \frac{f^2(\mathbf{x})\pi^2(\mathbf{x})}{q(\mathbf{x})} d\mathbf{x} - \frac{I^2}{N}$$

- optimal UIS proposal: $q(\mathbf{x}) \propto |f(\mathbf{x})|\pi(\mathbf{x})$
- for a generic $f(\mathbf{x})$, $q(\mathbf{x})$ should be as *close* as possible to $\pi(\mathbf{x})$



- Very difficult to find a good $q(\mathbf{x})$ a priori:
 - $\pi(\mathbf{x})$ can be only evaluated (up to a normalizing constant)
 - $\pi(\mathbf{x})$ may be multimodal, skewed, heavy tailed
- A posteriori metric: $\widehat{\text{ESS}} = \frac{1}{\sum_{n=1}^N \widehat{w}_n^2}$, although it presents serious problems²
- Use **multiple proposals (MIS)** and explore the space (**AIS**).

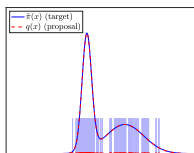
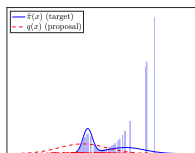
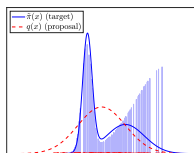
²V. Elvira, L. Martino, and C. P. Robert. "Rethinking the effective sample size". In: *International Statistical Review* 90.3 (2022), pp. 525–550.

The variance in IS and the need of better proposals

- A **good proposal** $q(\mathbf{x})$ is key for the efficiency of IS.
- Variance of the UIS estimator of $I = \int f(\mathbf{x})\pi(\mathbf{x})d\mathbf{x}$:

$$\text{Var}_{\pi(\mathbf{x})}(\widehat{I}) = \frac{1}{N} \int \frac{f^2(\mathbf{x})\pi^2(\mathbf{x})}{q(\mathbf{x})} d\mathbf{x} - \frac{I^2}{N}$$

- optimal UIS proposal: $q(\mathbf{x}) \propto |f(\mathbf{x})|\pi(\mathbf{x})$
- for a generic $f(\mathbf{x})$, $q(\mathbf{x})$ should be as *close* as possible to $\pi(\mathbf{x})$



- Very difficult to find a good $q(\mathbf{x})$ a priori:
 - $\pi(\mathbf{x})$ can be only evaluated (up to a normalizing constant)
 - $\pi(\mathbf{x})$ may be multimodal, skewed, heavy tailed
- A posteriori metric: $\widehat{\text{ESS}} = \frac{1}{\sum_{n=1}^N \bar{w}_n^2}$, although it presents serious problems²
- Use **multiple proposals (MIS)** and explore the space (**AIS**).

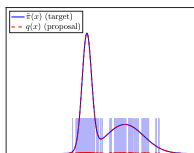
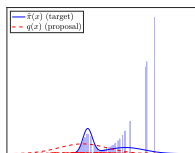
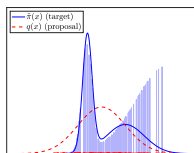
²V. Elvira, L. Martino, and C. P. Robert. "Rethinking the effective sample size". In: *International Statistical Review* 90.3 (2022), pp. 525–550. □ ◀ ▶ ⏪ ⏩ ☰ 🔍 ↻

The variance in IS and the need of better proposals

- A **good proposal** $q(\mathbf{x})$ is key for the efficiency of IS.
- Variance of the UIS estimator of $I = \int f(\mathbf{x})\pi(\mathbf{x})d\mathbf{x}$:

$$\text{Var}_{\pi(\mathbf{x})}(\widehat{I}) = \frac{1}{N} \int \frac{f^2(\mathbf{x})\pi^2(\mathbf{x})}{q(\mathbf{x})} d\mathbf{x} - \frac{I^2}{N}$$

- optimal UIS proposal: $q(\mathbf{x}) \propto |f(\mathbf{x})|\pi(\mathbf{x})$
- for a generic $f(\mathbf{x})$, $q(\mathbf{x})$ should be as *close* as possible to $\pi(\mathbf{x})$



- Very difficult to find a good $q(\mathbf{x})$ a priori:
 - $\pi(\mathbf{x})$ can be only evaluated (up to a normalizing constant)
 - $\pi(\mathbf{x})$ may be multimodal, skewed, heavy tailed
- A posteriori metric: $\widehat{\text{ESS}} = \frac{1}{\sum_{n=1}^N \bar{w}_n^2}$, although it presents serious problems²
- Use **multiple** proposals (**MIS**) and **explore** the space (**AIS**).

²V. Elvira, L. Martino, and C. P. Robert. "Rethinking the effective sample size". In: *International Statistical Review* 90.3 (2022), pp. 525–550. □ ◀ ▶ ⏪ ⏩ ☰ 🔍 ↻

Mini-project MC: IS in one dimension (1/2)

- Option 1: again a standard normal target and a normal proposal with mean $\mu = 0.5$ and $\sigma = 2$
- Option 2: replicate the two examples in the previous slide:
 - you can start with a target with just one Gaussian component (left mode)
- In any case, simulate $N = 20$ samples from the proposal (you can later play with different N).
 - estimate the target mean ($f(x) = x$), the target second moment (the target mean ($f(x) = x^2$), and the $\Pr(X > \gamma)$, e.g., with $\gamma = 2$ ($f(x) = \mathbb{1}_{|x| > \gamma}(x)$).
 - characterize the estimator (variance, bias, and MSE): you will have to run the estimator *many* times and then take empirical means.
- Play with the proposal's mean and variance and display the effect in the performance of both IS estimators (UIS and SNIS). Compare it with the raw/direct MC estimator performance.

Mini-project: IS in higher dimensions (2/2)

- The mismatch between target and proposal gets more problematic in higher dimensions:
 - standard normal target of dimension d_x : we start with $d_x = 2$ but we can play with larger dimensions later
 - select a proposal with also isotropic/diagonal covariance (you can generalize the proposal above)
 - do the sampling and weighting of IS, but select a sampling method in which you sequentially simulate each dimension
 - * if isotropic/diagonal proposal covariance, the conditionals are equal to the marginals
 - * if both proposal and target covariances are isotropic/diagonal, you can compute the weights as the product of all the "dimension weights" (both numerator and denominator factorizes with the marginals)
 - for a particular run, shows the inequality for the IS weights (you can sort them in decreasing order as a function of index)
 - approximate the MSE still for $d_x = 2$
- now play with the dimension, including a plot of MSE vs d_x and plot for some run the IS weights at a large dimension
 - if all is going well, you are experiencing the curse of dimensionality!

Outline

Bayesian inference

- Motivation: the bent coin example
- The Bayesian approach
- Probabilistic regression
- Analytically-tractable Bayesian models
- Intractable integrals

Monte Carlo methods

Basics

Mini-project 1: MC

Simulation of random variables

Inverse transform sampling

Importance Sampling

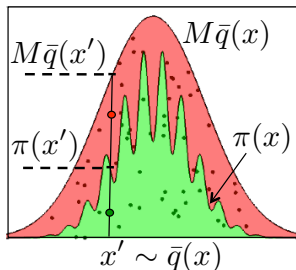
Mini-project 1 (cont)

Rejection sampling

Markov Chain Monte Carlo (MCMC)

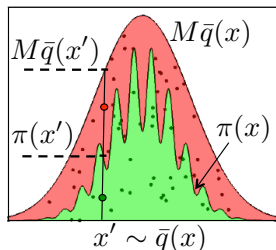
Accept/reject methods: rejection sampling

- We cannot sample from $\tilde{\pi}(x)$ (we do not even know the normalizing constant C_π), so we just know $\pi(x)$.
- We sample from a **proposal distribution** $\bar{q}(x)$ but not all samples are accepted.
- **Rejection sampling** procedure:
 1. **Sampling:** $x' \sim \bar{q}(x)$
 2. **Accepting (or not):**
 - * sample $u' \sim \mathcal{U}(0, 1)$
 - * accept the sample x' only if $u' < \frac{\pi(x')}{M\bar{q}(x')}$.



Accept/reject methods: rejection sampling

1. **Challenge 1:** M must be known so $M\bar{q}(x) > \pi(x)$, for all x .
2. **Challenge 2:** More efficient when you do not reject samples, i.e., when the gap between $M\bar{q}(x)$ and $\pi(x)$ is low.



- We want an **acceptance rate** close to 1.

$$\begin{aligned} a_R = \mathbb{E}_{\bar{q}}[p_a(x)] &= \int_{\mathbb{R}} p_a(x) \bar{q}(x) dx, \\ &= \int_{\mathbb{R}} \frac{\pi(x)}{M\bar{q}(x)} \bar{q}(x) dx, \\ &= \frac{C_\pi}{M}. \end{aligned}$$

Outline

Bayesian inference

- Motivation: the bent coin example
- The Bayesian approach
- Probabilistic regression
- Analytically-tractable Bayesian models
- Intractable integrals

Monte Carlo methods

- Basics
 - Mini-project 1: MC
- Simulation of random variables
- Inverse transform sampling
- Importance Sampling
 - Mini-project 1 (cont)
- Rejection sampling
- Markov Chain Monte Carlo (MCMC)

Markov Chain Monte Carlo (MCMC)

- Markov Chain Monte Carlo (MCMC) is a classic method in Monte Carlo.
- Unfortunately we only know how to draw samples from very easy distributions (e.g., Gaussian).
 - In the general case, we cannot use the “raw” Monte Carlo.
- It builds a Markov Chain that has as equilibrium distribution the targeted distribution (the posterior in Bayesian inference).
 - $x_1 \rightarrow x_2 \rightarrow \dots \rightarrow x_t \rightarrow x_{t+1} \rightarrow \dots$
- The first samples do not follow the true distribution (burn-in period).
- However, the more samples there are, the more closely the distribution of the sample will match the actual desired/targeted distribution.
- Two widely known/used algorithms:
 1. Gibbs sampling algorithm
 2. **Metropolis-Hastings algorithm**

Metropolis-Hastings algorithm

- Remember, the goal is obtaining samples from $\tilde{\pi}(x) = \frac{\pi(x)}{C_\pi}$, where C_π is unknown.
- The algorithm proceeds as follows:
 - Initialization:
 - * Set $t = 0$
 - * Select an initial state x_0
 - Generate a candidate state (sample) $x' \sim q(x|x_{t-1})$
 - * again, we need a proposal distribution $q(x|x_{t-1})$ (we use the previous state as a parameter)
 - Compute the acceptance probability

$$\alpha(x', x_{t-1}) = \min \left(1, \frac{\pi(x')q(x_{t-1}|x')}{\pi(x_{t-1})q(x'|x_{t-1})} \right)$$

- Accept or reject** with probability $\alpha(x', x_{t-1})$
 - sample $u' \sim \mathcal{U}(0, 1)$
 - if $u' < \alpha(x', x_{t-1})$, accept the sample x' , i.e. $x_t = x'$.
 - if $u' \geq \alpha(x', x_{t-1})$, take as sample the previous state, i.e., $x_t = x_{t-1}$ (then it is repeated).
- Set $t = t + 1$

Metropolis-Hastings algorithm (with symmetric proposal)

- Example if $q(x|x_{t-1}) = \mathcal{N}(x; x_{t-1}, \sigma^2)$.
- The acceptance rate is simplified because $q(x'|x_{t-1}) = q(x_{t-1}|x')$
- The algorithm proceeds as follows:
 1. Initialization:
 - * Set $t = 0$
 - * Select an initial state x_0
 2. Generate a candidate state (sample), $x' \sim q(x|x_{t-1})$
 3. Compute the acceptance probability

$$\alpha(x', x_{t-1}) = \min \left(1, \frac{\pi(x')}{\pi(x_{t-1})} \right)$$

4. **Accept or reject** with probability $\alpha(x', x_{t-1})$
 - 4.1 sample $u' \sim \mathcal{U}(0, 1)$
 - 4.2 if $u' < \alpha(x', x_{t-1})$, accept the sample x' , i.e. $x_t = x'$.
 - 4.3 if $u' \geq \alpha(x', x_{t-1})$, take as sample the previous state, i.e., $x_t = x_{t-1}$ (then it is repeated).
5. Set $t = t + 1$