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
UPC, Barcelona, July 7-11, 2025

Outline

Outline

Outline

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:

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

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

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

  - 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:

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

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

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

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

Coin example: Maximum a Posteriori (MAP) Estimation

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

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

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

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

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

- We still obtain the same results (uninformative prior)

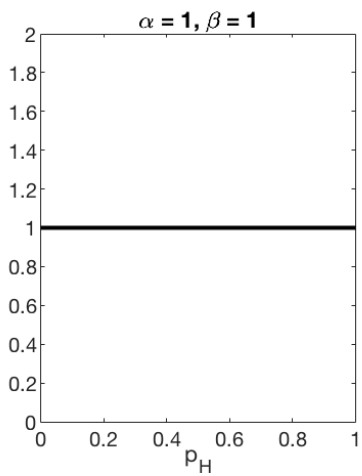
[MacKay13] D. MacKay, "Information theory, inference and learning algorithms." Cambridge university press, 2003.

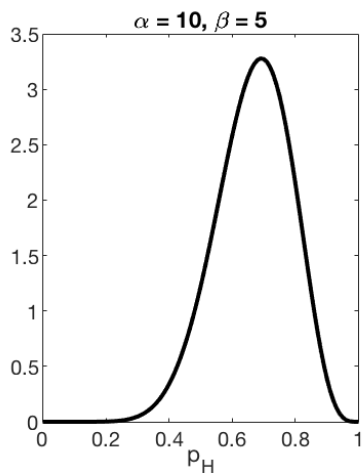
Coin example: The Importance of a Good Prior

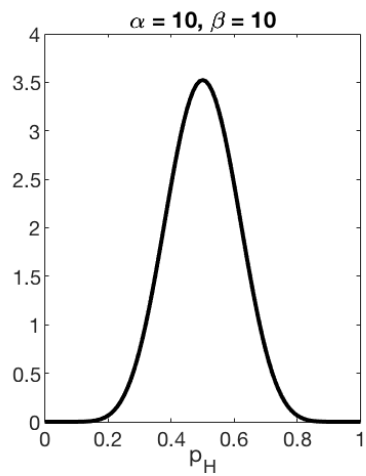
- The Beta distribution is a flexible prior for parameters in $[0, 1]$.

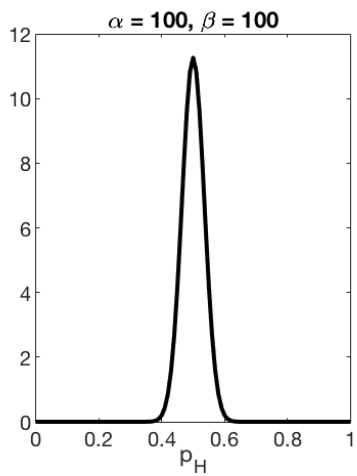
$$p(h) = Beta(h|\alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} h^{\alpha-1}(1-h)^{\beta-1}$$

  - $\alpha > 0$ and $\beta > 0$ are hyper-parameters of the prior and correspond to "one plus the pseudo-counts"
- The Beta distribution is a **conjugate prior** for the binomial likelihood.
  - Conjugate? It means that the posterior is the same family (Beta in this case) than the prior for that likelihood.
  - Is this good news? Yes, it is: for most combinations of likelihood and prior, the posterior has **intractable form** (no analytic solution).









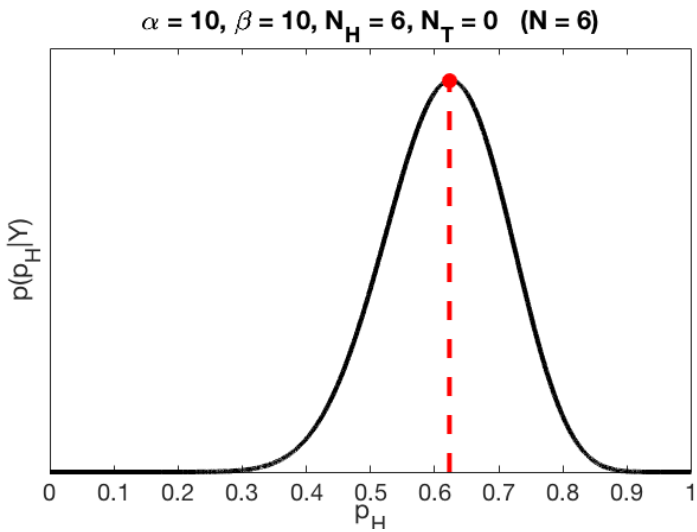
Coin example: Estimations are good, distributions are better

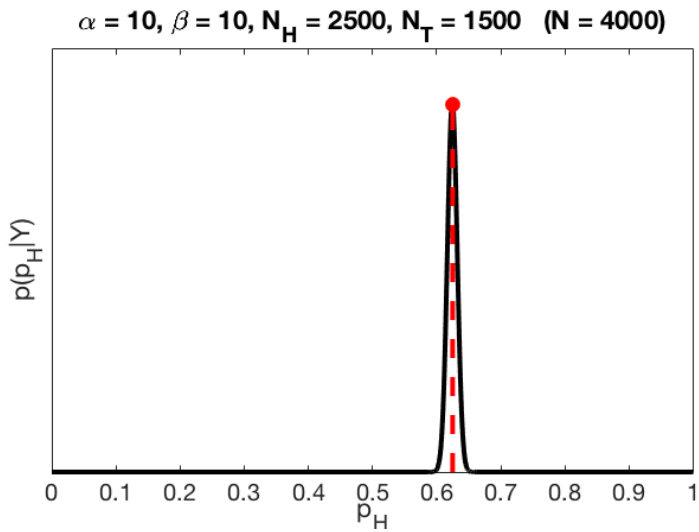
- If prior is $p(h) = Beta(h|\alpha, \beta)$ and likelihood is $\mathrm{B}(N_H; N, h)$, the posterior is:
  - $p(h|Y) = Beta(h|\alpha + N_H, \beta + N_T)$

  (recall: Y contains the outcomes of tossing the coin N times)
- We set the prior $p(h) = Beta(h|10, 10)$
  - Case 1: $N = 6$, $N_H = 6$, and $N_T = 0$

    $p(h|Y) = Beta(16, 10) \quad \Rightarrow \widehat{h}^{*(MAP)} \approx 0.62$
  - Case 2: $N = 4000$, $N_H = 2500$, and $N_T = 1500$

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





- 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!**

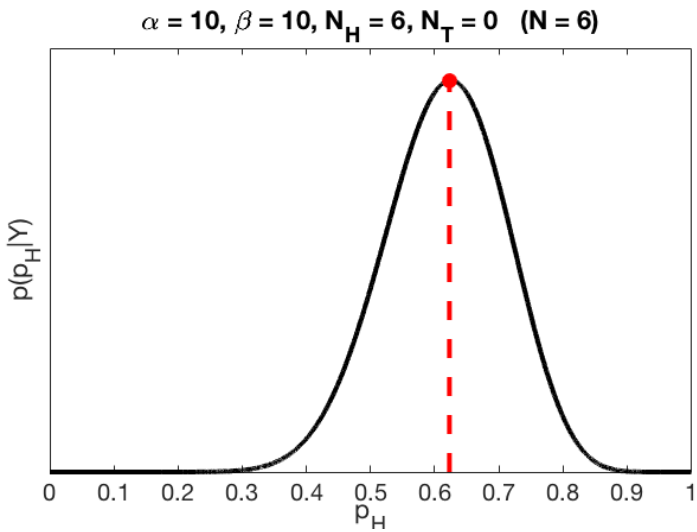
Coin example: Estimations are good, distributions are better

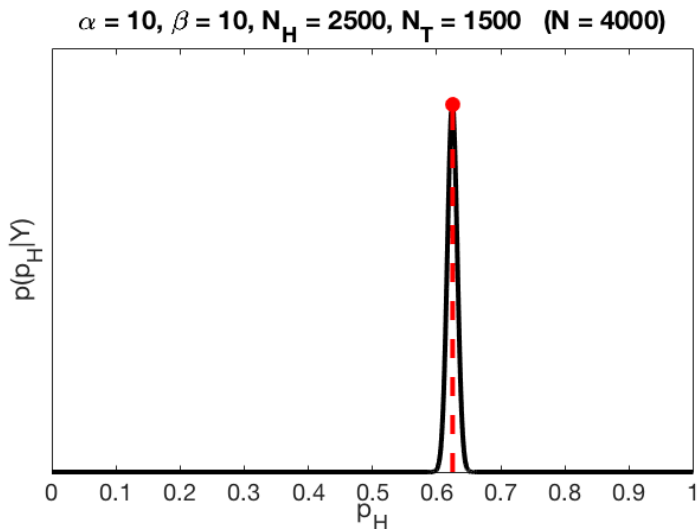
- If prior is $p(h) = Beta(h|\alpha, \beta)$ and likelihood is $\mathrm{B}(N_H; N, h)$, the posterior is:
  - $p(h|Y) = Beta(h|\alpha + N_H, \beta + N_T)$

  (recall: Y contains the outcomes of tossing the coin N times)
- We set the prior $p(h) = Beta(h|10, 10)$
  - Case 1: $N = 6$, $N_H = 6$, and $N_T = 0$

    $p(h|Y) = Beta(16, 10) \quad \Rightarrow \widehat{h}^{*(MAP)} \approx 0.62$
  - Case 2: $N = 4000$, $N_H = 2500$, and $N_T = 1500$

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





- 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!**

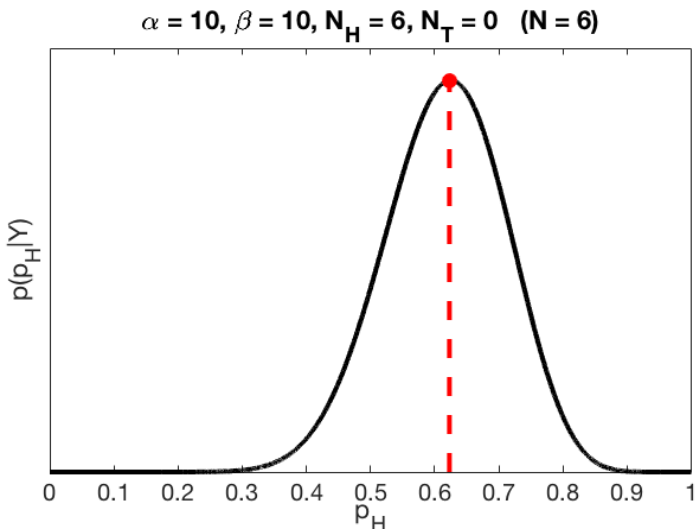
Coin example: Estimations are good, distributions are better

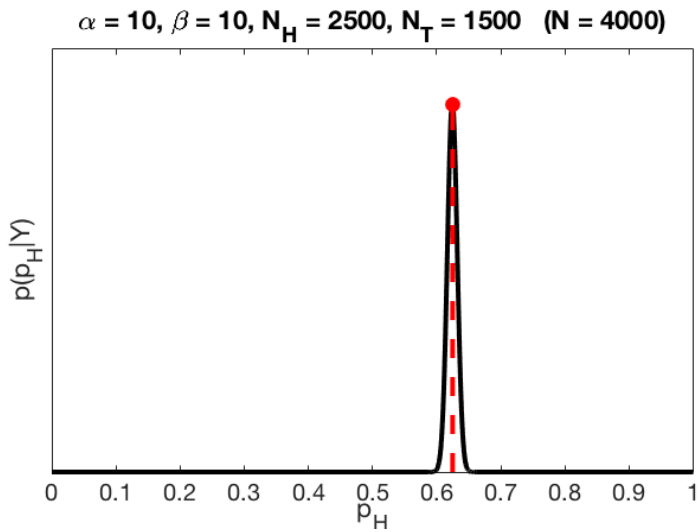
- If prior is $p(h) = Beta(h|\alpha, \beta)$ and likelihood is $\mathrm{B}(N_H; N, h)$, the posterior is:
  - $p(h|Y) = Beta(h|\alpha + N_H, \beta + N_T)$

  (recall: Y contains the outcomes of tossing the coin N times)
- We set the prior $p(h) = Beta(h|10, 10)$
  - Case 1: $N = 6$, $N_H = 6$, and $N_T = 0$

    $p(h|Y) = Beta(16, 10) \quad \Rightarrow \widehat{h}^{*(MAP)} \approx 0.62$
  - Case 2: $N = 4000$, $N_H = 2500$, and $N_T = 1500$

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





- 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

The Bayesian approach: the posterior

- The *posterior distribution* for the model parameters given the observed data, $p(\text{parameters}|\text{data})$, is the goal in Bayesian inference.
- It is obtained applying *Bayes' Rule*:

$$p(\text{parameters}|\text{data}) = \frac{p(\text{data}|\text{parameters})p(\text{parameters})}{p(\text{data})}$$

where

  - $p(\text{data}|\text{parameters})$: **likelihood**, it relates statistically the data with the unknowns of the model. Most of our assumptions are there (e.g., if the relation is linear, if the noise is additive, if the noise is Gaussian, etc)
  - $p(\text{parameters})$: **prior** distribution, it contains all our knowledge about the unknown parameter (similar to the regularization)
  - $p(\text{data})$: **model evidence, marginal likelihood, or normalizing constant** (very hard to compute sometimes)

- Following the previous notation in regression,

$$\overbrace{p(\boldsymbol{\beta}|\mathcal{D})}^{\text{posterior}} = \frac{\overbrace{p(\mathcal{D}|\boldsymbol{\beta})}^{\text{likelihood}}\overbrace{p(\boldsymbol{\beta})}^{\text{prior}}}{\underbrace{p(\mathcal{D})}_{\text{marginal likelihood}}}$$

- We will see later how to use this posterior for predicting the output a new sample in regression.

The Bayesian approach: workflow

- In supervised learning, for a new input $\mathbf{x}^*$, our prediction/estimation is $\hat{y}^* \in \mathbb{R}$ (a number).

  Related questions:
  - How certain are you that the true response is $\hat{y}^*$?
  - How much money would you dare to bet?
  - How much probability would you assign to other responses different from $\hat{y}^*$?

- Why not instead giving a probability associated for each possible outcome?
- The Bayesian/probabilistic approach:
  1. We define a **probabilistic model** that expresses qualitative aspects of our knowledge (e.g., forms of distributions, independence assumptions). The model will have some unknown parameters (also before!).
  2. We specify a **prior probability** distribution for these unknown parameters that expresses our beliefs about which values are more or less likely, before seeing the data.
  3. We **gather data**.
  4. We **compute the posterior pdf** for the parameters, given the observed data.
  5. We **use this posterior** pdf to:
     - Reach scientific conclusions, properly accounting for uncertainty.
     - Make predictions by averaging over the posterior distribution.
     - Make decisions so as to minimize posterior expected loss.

The Bayesian approach: challenges

- Recall the **posterior**:

$$\overbrace{p(\boldsymbol{\beta}|\mathcal{D})}^{\text{posterior}} = \frac{\overbrace{p(\mathcal{D}|\boldsymbol{\beta})}^{\text{likelihood}}\overbrace{p(\boldsymbol{\beta})}^{\text{prior}}}{\underbrace{p(\mathcal{D})}_{\text{marginal likelihood}}}$$

- Complex problems require complex models with complex distributions.
- Even for simple likelihood and simple (but not conjugate!) prior, the posterior might be intractable:
  - The marginal likelihood (denominator) $p(\mathcal{D}) = \int p(\mathcal{D}|\boldsymbol{\beta})p(\boldsymbol{\beta})d\boldsymbol{\beta}$ impossible to obtain
  - integrals on the posterior, $\int f(\boldsymbol{\beta})p(\boldsymbol{\beta}|\mathcal{D})d\boldsymbol{\beta}$, impossible to obtain
- **Monte Carlo methods**: A very general technique to represent the posterior by simulating **random samples** from it. We can then:
  - *Visualize* the distribution by viewing these sample values, or low-dimensional projections of them.
  - Make *Monte Carlo* estimates for probabilities or expectations with respect to the distribution, by taking averages over these sample values.
- **MAP**: *maximum a posteriori* is a point-wise estimate, without uncertainty quantification, as $\widehat{\beta} = \arg\max_{\beta} p(\boldsymbol{\beta}|\mathcal{D})$.
  - It is a popular a popular alternative to Monte Carlo, because it reduces to an optimization problem, for which efficient algorithms often exist.
- **Sampling** from the posterior is usually more difficult, but this is nevertheless the **dominant approach in Bayesian/probabilitc ML**.

Outline

Relation between MAP estimates and regularized regression

- MAP estimation can be also interpreted in non-Bayesian terms, by thinking of the log prior as a regularization, the penalty function added.
- In linear regression, we have training data $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$ for which we fit a linear combination of basis function by using

$$y = f(\mathbf{x}, \boldsymbol{\beta}) + \text{noise}$$

$$f(\mathbf{x}, \beta) = \beta_0 + \sum_{j=1}^{m} \beta_j \phi_j(\mathbf{x}) = \boldsymbol{\beta}^T \phi(\mathbf{x})$$

- Then we have seen that the maximum likelihood (ML) estimator of $\boldsymbol{\beta}$ (with Gaussian assumption of the noise term) was:

$$(\widehat{\boldsymbol{\beta}}, \widehat{\sigma}) = \arg\max_{(\boldsymbol{\beta},\sigma)} L(\boldsymbol{\beta}, \sigma) = \arg\min_{(\boldsymbol{\beta},\sigma)} -\log L(\beta, \sigma)$$

- Then, introduced regularization to avoid overfitting with an extra penalty function $R(\boldsymbol{\beta})$

$$\widehat{\beta} = \arg\min_{\boldsymbol{\beta}} -\log L(\boldsymbol{\beta}, \sigma) + \lambda R(\boldsymbol{\beta})$$

- This penalty function can be interpreted as the negative log prior of $\boldsymbol{\beta}$:
  - LASSO penalty $\Leftrightarrow$ MAP estimate with **Laplace Prior**
  - Squared penalty, $\lambda \sum_{j=1}^{m-1} \beta_j^2$, $\Leftrightarrow$ MAP estimate with **Gaussian Prior**

Predictive Posterior Distribution

- The most obvious drawback of MAP estimation, and indeed of any other point estimate such as the posterior mean or median, is that it **does not provide any measure of uncertainty**.
- In many applications, it is important to know how much one can trust a given estimate (and we had it with the posterior distribution)
- As a consequence, instead of $\hat{y}^*$ (point estimate), we prefer to give the predictive posterior distribution to make prediction at a new input point $\mathbf{x}^*$:

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

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

  - allows to have a complete probabilistic information about the predicted value (mean, variance, higher probability density regions, ...)
  - marginalization of the model parameters (we take into account all possible parameters instead of just $\hat{\boldsymbol{\beta}}$)

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).

$$p(y^*|\mathbf{x}^*, \mathcal{D}) = \sum_{i=1}^{K} p(y^*, M_i|\mathbf{x}^*, \mathcal{D}) = \sum_{i=1}^{K} p(y^*|\mathbf{x}^*, \mathcal{D}, M_i)p(M_i|\mathbf{x}^*, \mathcal{D})$$

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

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

Analytically-Tractable Bayesian Models

- For most Bayesian inference problems, the integrals needed to do inference and prediction are not analytically tractable
  - hence the need for numerical quadrature, Monte Carlo methods, or various approximations.
- Most of the exceptions involve **conjugate priors**, which combine nicely with the likelihood to give a posterior distribution of the same form. Examples:
  1. Independent observations of Gaussian variables with Gaussian prior for the mean, and either known variance or inverse-Gamma prior for the variance.
  2. Linear regression with Gaussian prior for the regression coefficients, and Gaussian noise, with known variance or inverse-Gamma prior if the variance is unknown.
- It is nice when a tractable model and prior are appropriate for the problem.
  - Unfortunately, people are tempted to use such models and priors even when they are not appropriate.
    - Traditionally, Gaussian distribution has been largely (over-)used because of its good properties.
- Good books in Bayesian theory:
  - 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 $\mathbf{x} = [x_1, x_2]^\top \in \mathbb{R}^{2\times 1}$

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

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:

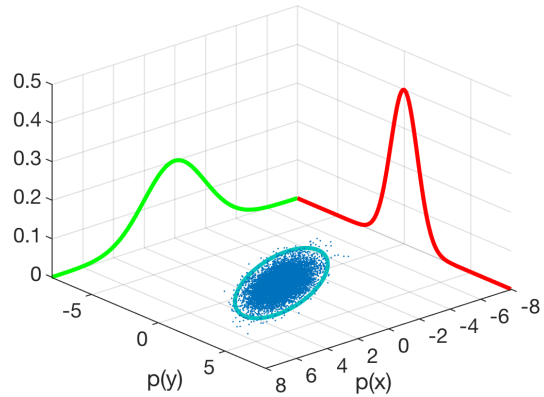
$$p(x_1) = \int p(x_2, x_1) dx_2,$$

   marginalizing x_2, $p(x_1)$ is also Gaussian
2. **Conditional** of a joint Gaussian distribution is still Gaussian (equivalent to first point):

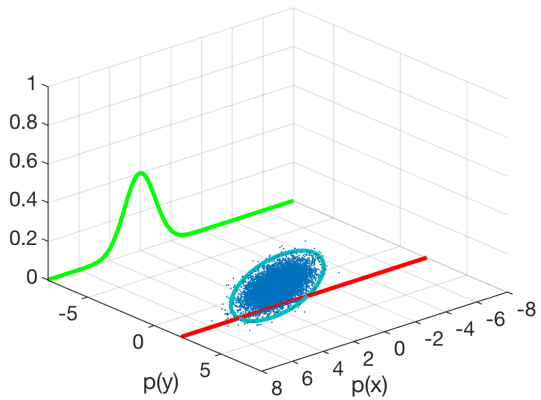
$$p(x_1|x_2) = \frac{p(x_1, x_2)}{p(x_2)}.$$

Gaussian properties (cont)

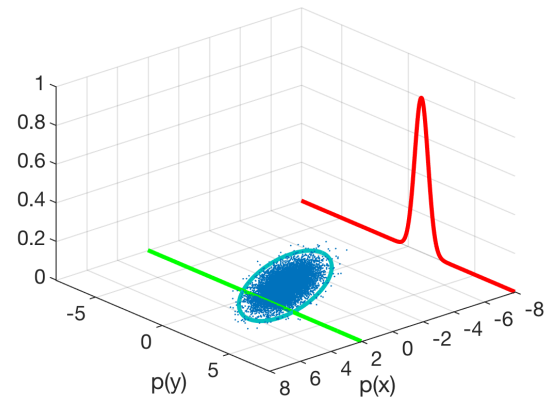
1. Marginals of a bi-variate Gaussian distribution are Gaussian:



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:

$$y_i|\mathbf{x}_i;\boldsymbol{\beta} \sim \mathcal{N}\left(y;\mathbf{x}_i^T\boldsymbol{\beta},\sigma^2\right), \qquad i=1,\ldots,N$$
$$\boldsymbol{\beta} \sim \mathcal{N}\left(\boldsymbol{\beta};m_0,S_0\right)$$

  - 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**

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

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

Intractable Integrals

- The previous example is the dreamed scenario, but it is not usual.
  - Usually, the posterior has not known form (not a known distribution), and the normalizing constant $p(\mathcal{D})$ cannot be computed.
- Moreover, the full Bayesian approach requires to integrate over the posterior distribution,
  - e.g., if we want the posterior mean,

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

  - When the posterior distribution is complex $\Rightarrow$ this integral is intractable.
- There exists many different algorithms to approximate this integral:
  - grid-based method $\Rightarrow$ Complex as the dimension of the unknowns parameters increases,
  - variational algorithms: Approximate posterior distribution as a product of exponential families distributions,
  - Monte-Carlo techniques: one of the most used technique nowadays.

Outline

Outline

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 at[1]
- 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/

[1] N. 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.

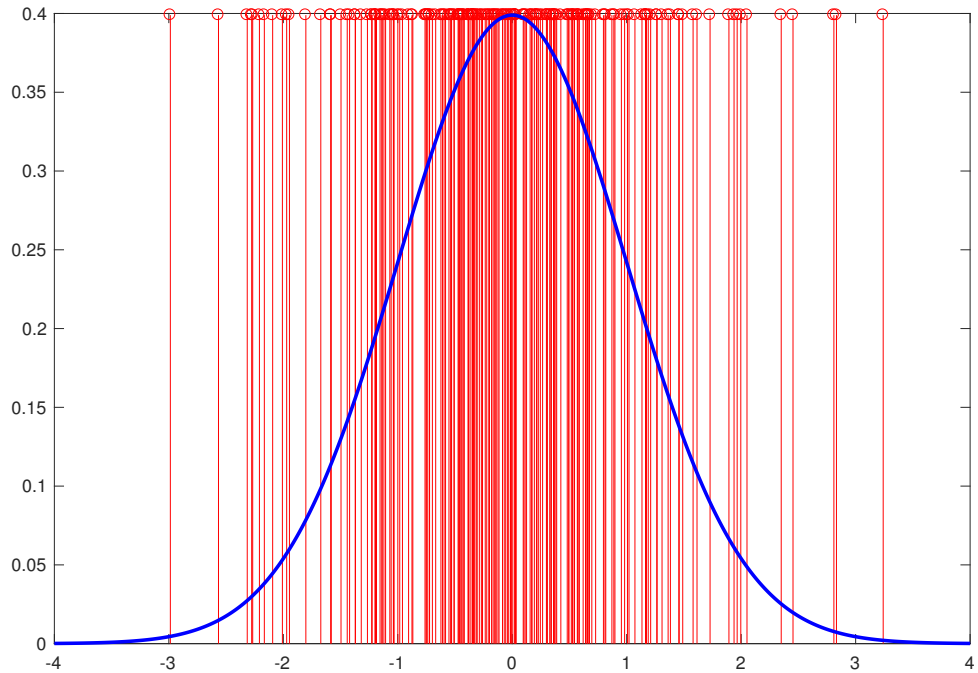
$$\mathbf{x}^{(n)} \sim \tilde{\pi}(\mathbf{x}), \qquad n = 1, ..., N.$$

- Then, we can approximate the targeted distribution as

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

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

- 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!

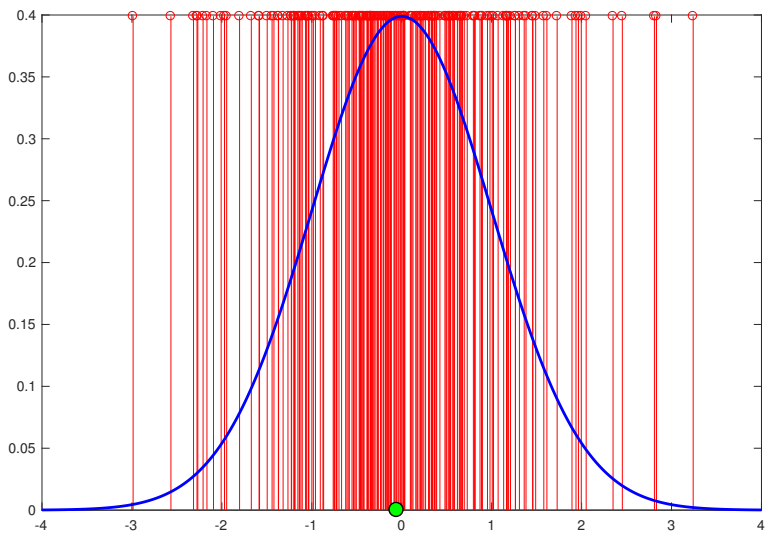
$$I(f) \equiv \mathbb{E}_{\tilde{\pi}(\mathbf{x})}[f(\mathbf{x})] = \int f(\mathbf{x})\tilde{\pi}(\mathbf{x})d\mathbf{x} \approx \int f(\mathbf{x})\tilde{\pi}^N(\mathbf{x})d\mathbf{x} \approx \int f(\mathbf{x}) \left(\frac{1}{N} \sum_{n=1}^{N} \delta_{\mathbf{x}^{(n)}}(\mathbf{x}) \right) d\mathbf{x}$$

$$\approx \frac{1}{N} \sum_{n=1}^{N} \int f(\mathbf{x})\delta_{\mathbf{x}^{(n)}}(\mathbf{x})d\mathbf{x}$$

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

- 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

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

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

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

  2. Due to the central limit theorem:

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

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

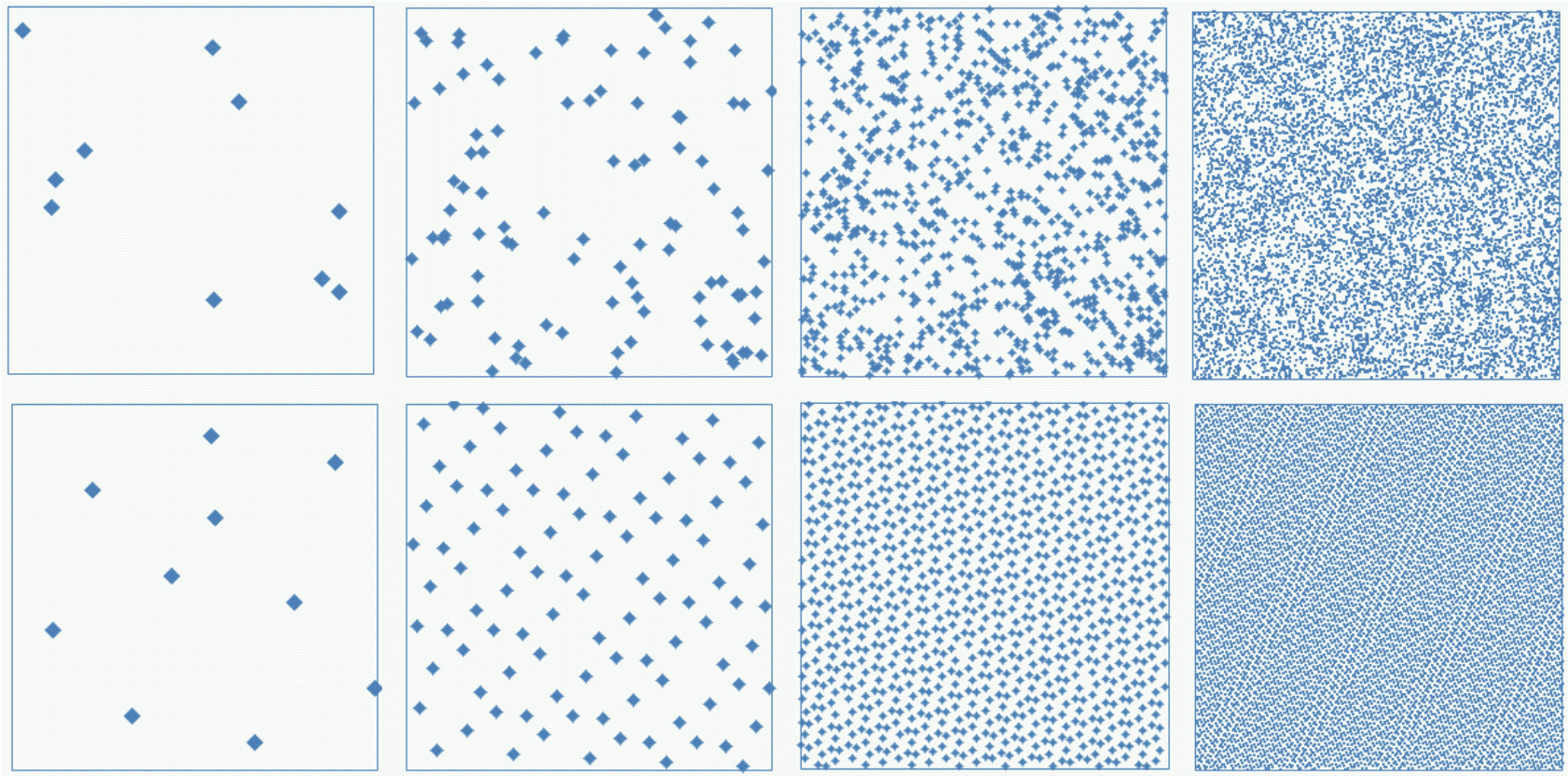
Random numbers

Simulation of random variables

- Solving the previous problem requires the simulation of random numbers according to some distribution (we called it $\tilde{\pi}(\mathbf{x})$).

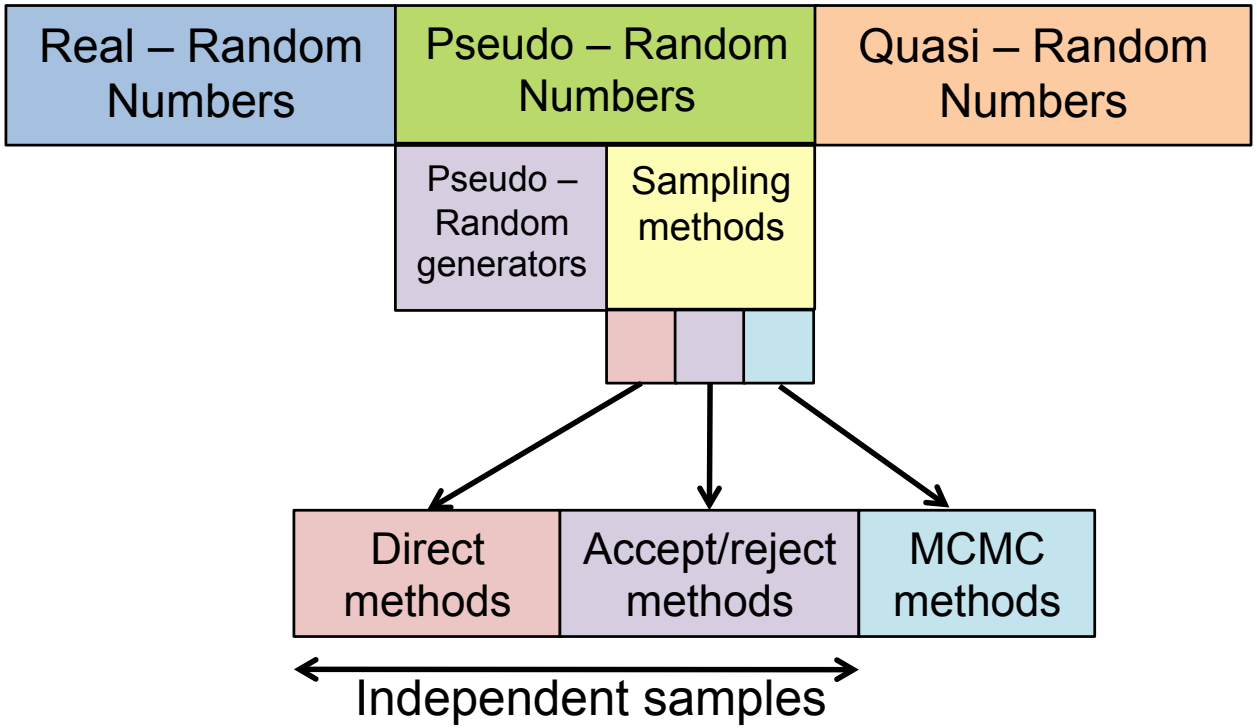
1. **True** random numbers are generated using a (well-studied) physical device as the source of randomness.
   - e.g., coin flipping, roulette wheels, white noise, and the count of particles emitted by a radioactive source.
2. **Pseudo**-random numbers correspond to a deterministic sequence that passes 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

Inverse transform sampling

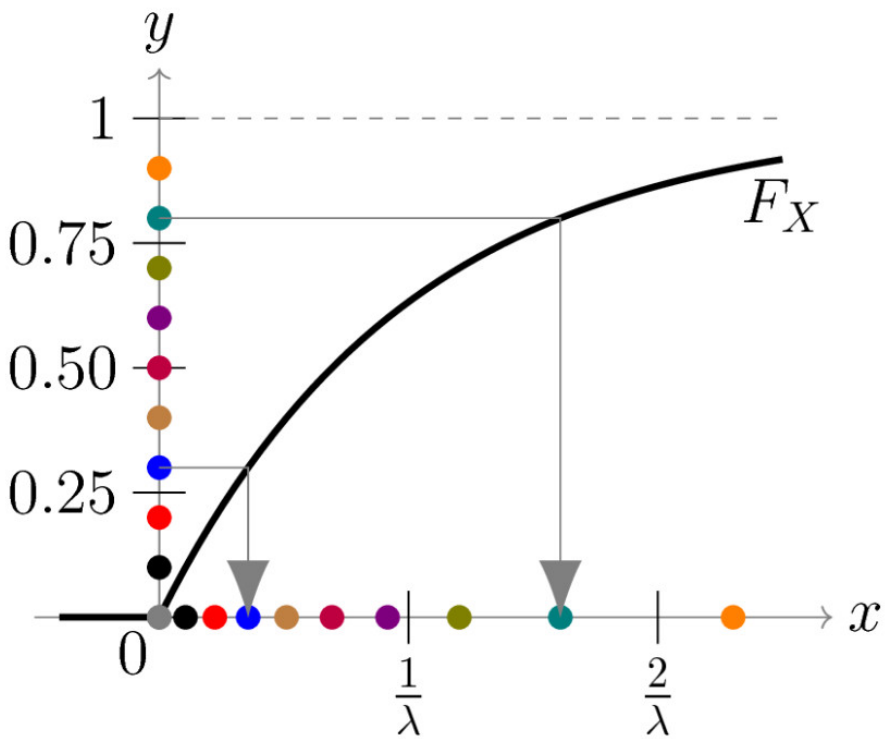
- The **inverse transform sampling** method uses uniform samples for simulating the r.v. $X \sim p_X(x)$, with CDF $F_X(x)$
- The method reads as follows:
  1. generate a random number $U \sim \mathcal{U}(0,1)$
  2. find the inverse of the CDF $F_X(x)$, i.e., $F_X^{-1}(x)$.
  3. compute $X = F_X^{-1}(U)$,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

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

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

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

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

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

Importance Sampling basics

- Basic or aka unnormalized IS (UIS) estimator:

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

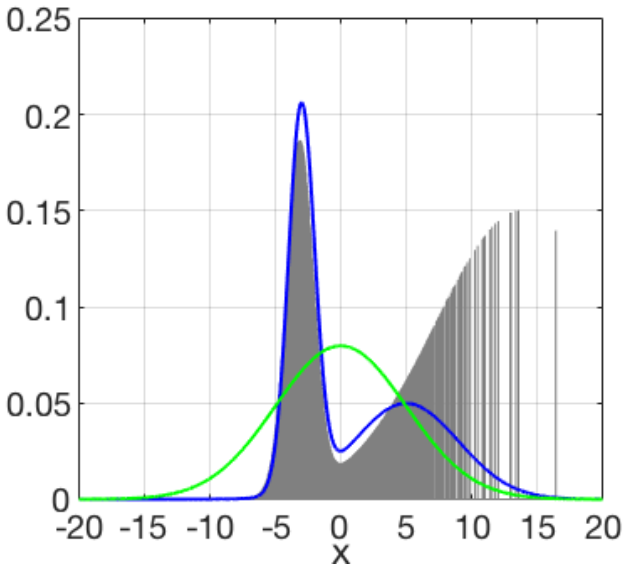
  - $w^{(n)} = \frac{\tilde{\pi}\left(\mathbf{x}^{(n)}\right)}{q\left(\mathbf{x}^{(n)}\right)}$, $n = 1, ..., N$, are the **importance weights**
  - only constraint: $\tilde{\pi}(\mathbf{x})$ must be evaluated
- Unfortunately, sometimes we have only access to $\pi(\mathbf{x})$, where $\tilde{\pi}(\mathbf{x})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

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

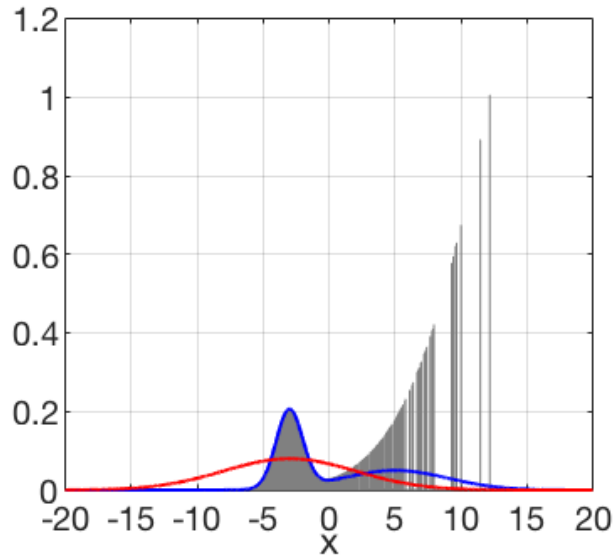
where $\bar{w}^{(n)} = \frac{w^{(n)}}{\sum_{j=1}^{N} w^{(j)}}$ are the normalized weights ($\sum_{j=1}^{N} \bar{w}^{(j)} = 1$).

Importance Sampling: example

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



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



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

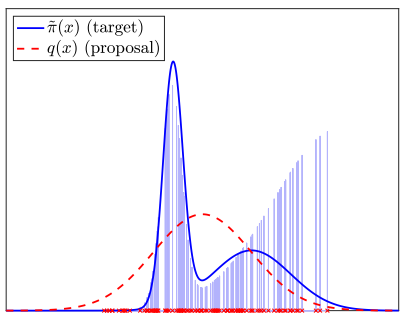
- The approximation converges in both cases, but a good proposal is key for the efficiency of IS.

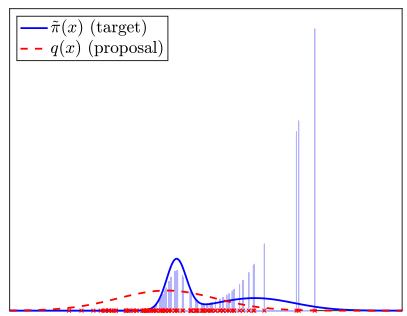
The variance in IS and the need of better proposals

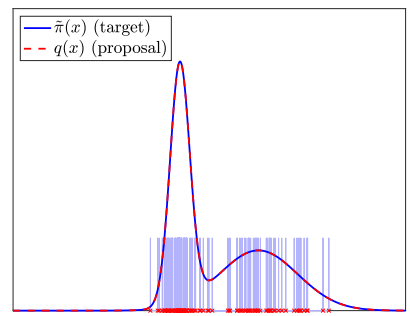
- A good proposal $q(\mathbf{x})$ is key for the efficiency of IS.
- Variance of the UIS estimator of $I = \int f(\mathbf{x})\pi(\mathbf{x})d\mathbf{x}$:

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

- 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: $\widehat{\text{ESS}} = \frac{1}{\sum_{n=1}^{N} \bar{w}_n^2}$, although it presents serious problems[2]
- Use **multiple** proposals (**MIS**) and **explore** the space (**AIS**).

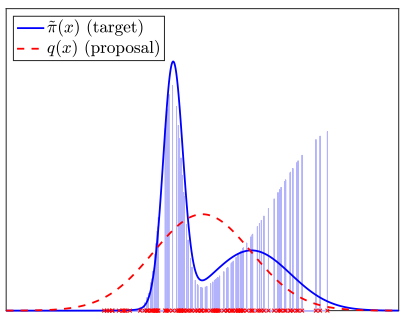
[2] V. Elvira, L. Martino, and C. P. Robert. "Rethinking the effective sample size". In: *International Statistical Review* 90.3 (2022), pp. 525–550.

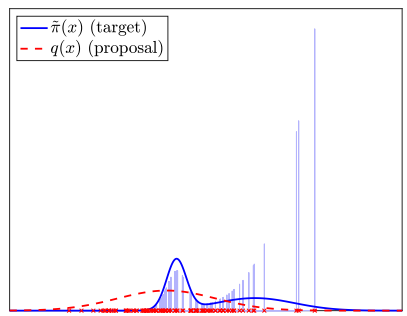
The variance in IS and the need of better proposals

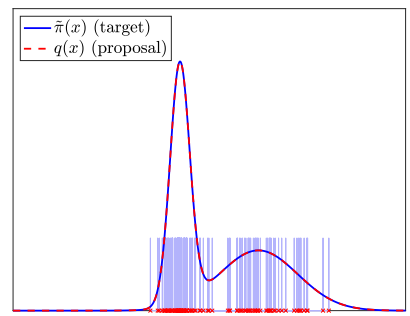
- A good proposal $q(\mathbf{x})$ is key for the efficiency of IS.
- Variance of the UIS estimator of $I = \int f(\mathbf{x})\pi(\mathbf{x})d\mathbf{x}$:

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

- 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 problems[2]
- Use **multiple** proposals (**MIS**) and **explore** the space (**AIS**).

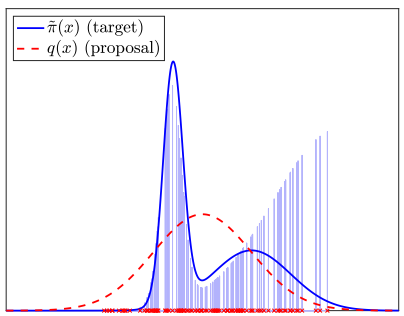
[2] V. Elvira, L. Martino, and C. P. Robert. "Rethinking the effective sample size". In: *International Statistical Review* 90.3 (2022), pp. 525–550.

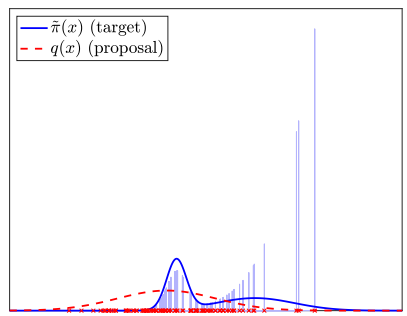
The variance in IS and the need of better proposals

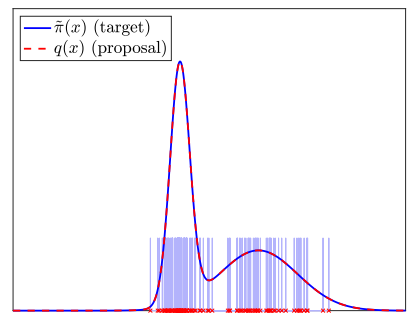
- A good proposal $q(\mathbf{x})$ is key for the efficiency of IS.
- Variance of the UIS estimator of $I = \int f(\mathbf{x})\pi(\mathbf{x})d\mathbf{x}$:

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

- 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 problems[2]
- Use **multiple** proposals (**MIS**) and **explore** the space (**AIS**).

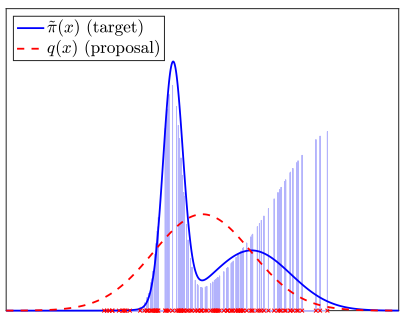
[2] V. Elvira, L. Martino, and C. P. Robert. "Rethinking the effective sample size". In: *International Statistical Review* 90.3 (2022), pp. 525–550.

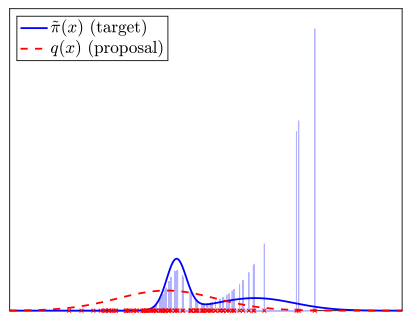
The variance in IS and the need of better proposals

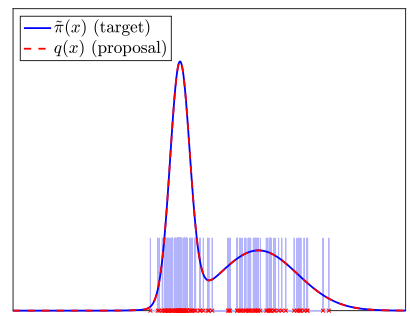
- A good proposal $q(\mathbf{x})$ is key for the efficiency of IS.
- Variance of the UIS estimator of $I = \int f(\mathbf{x})\pi(\mathbf{x})d\mathbf{x}$:

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

- 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 problems[2]
- Use **multiple** proposals (**MIS**) and **explore** the space (**AIS**).

[2] V. 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 $\gamma = 2$ ($f(x) = \mathbb{I}_{|x|>\gamma}(x)$).
  - characterize the estimator (variance, bias, and MSE): you will have to run the estimator *many* times and then take empirical means.
- 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.

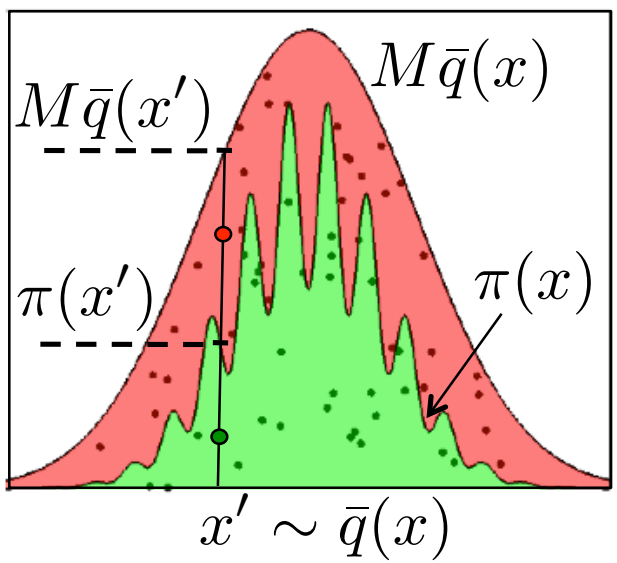
Mini-project: IS in higher dimensions (2/2)

- 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 dx 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

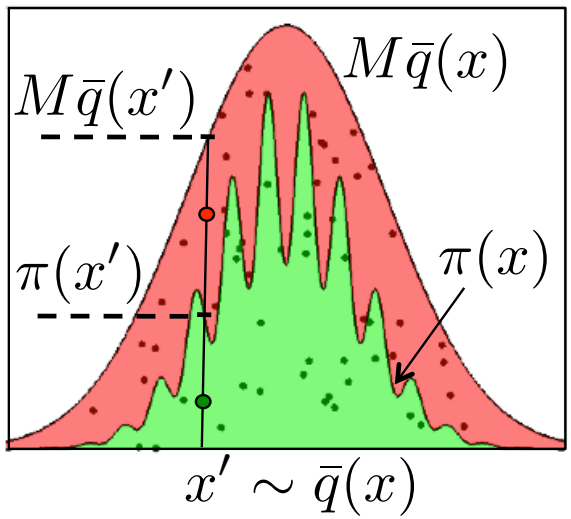
Accept/reject methods: rejection sampling

- W cannot sample from $\tilde{\pi}(\mathbf{x})$ (we do not even know the normalizing constant C_π), so we just know $\pi(\mathbf{x})$.
- 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 $M\bar{q}(\mathbf{x}) > \pi(\mathbf{x})$, for all $\mathbf{x}$.
2. **Challenge 2:** More efficient when you do not reject samples, i.e., when the gap between $M\bar{q}(\mathbf{x})$ and $\pi(\mathbf{x})$ 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

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 equillibrium distribution the targeted distribution (the posterior in Bayesian inference).
  - $x_1 \rightarrow x_2 \rightarrow ... \rightarrow x_t \rightarrow x_{t+1} \rightarrow ...$
- 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:
  1. Initialization:
     - Set $t = 0$
     - Select an initial state x_0
  2. 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)
  3. Compute the acceptance probability

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

  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 $t = t + 1$

Metropolis-Hastings algorithm (with symmetric proposal)

- Example if $q(x|x_{t-1}) = \mathcal{N}\left(x; x_{t-1}, \sigma^2\right)$.
- 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 $t = t + 1$