apts.ac.uk

Academy for PhD Training in Statistics

Computer Intensive Statistics

Richard Everitt
richard.everitt@warwick.ac.uk

10th–14th July, 2023

Compiled: July 11, 2023

Part 1

Introduction, Motivation & Basics

What *is* Computer Intensive Statistics

Computer, *n.* A device or machine for performing or facilitating calculation.
Compare Middle French computeur person who makes calculations (1578).

Intensive, *adj.* Of very high degree or force, vehement.
French intensif, -ive (14–15th cent. in Hatzfeld & Darmesteter).

Statistics, *n.* The systematic collection and arrangement of numerical facts or data of any kind; (also) the branch of science or mathematics concerned with the analysis and interpretation of numerical data and appropriate ways of gathering such data.
In early use after French statistique and German Statistik.

What Makes Statistics Computer Intensive?

Some *good* reasons for using computer-intensive methods:

Complexity Complex models cannot often be dealt with analytically.

Intractability Models which are not available analytically.

Laziness Computer time is cheap; human time isn't.

Scale Large data sets bring fresh challenges.

We won't address the *bad* reasons here. . .

Vevox.app 170–356–838

What is your familiarity with Computer Intensive Statistics?

What Makes Statistics Computer Intensive?

Some *good* reasons for using computer-intensive methods:

Complexity Complex models cannot often be dealt with analytically.

Intractability Models which are not available analytically.

Laziness Computer time is cheap; human time isn't.

Scale Large data sets bring fresh challenges.

We won't address the *bad* reasons here. . .

Vevox.app 170–356–838

What is your familiarity with Computer Intensive Statistics?

Part 1— Section 1

Motivation

Motivating Problem: Population genetics I

What shapes genetic variation?

```
AACGAGTACTGGCTAAAGCTCGACTCGCTTACGTCAGTCTCTTT
AACGAGTACTGGCTAAAGCTCGACTCGCTTACGTCAGTCTCTTT
AACGGGTACTGGCTAAAGCTCGACTCGCTTACGTCAGTCTCTTT
AACGGGTACTGGCTAAAGCTCGACTCGCTTACGTCAGTCTCTTT
AACGGGTACTGGCTAAAGCTCGACTCGCTTACGTCAGTCTCTTT
AACGGGTACTGGCTAAAGCTCGACTCGCCTACGTCAGTCTCTTT
AACGGGTACTGGCTAAAGCTCGACTCGCCTACGTCAGTCTCTTT
AACGGGTACTGGCTAAAGCTCGACTCGCCTACGTCAGTCTCCTT
AACGAGTACTGGCTAAAGCTCGACTCGCTTACGTCAGTCTCTTT
AACGGGTACTGGCTAAAGCTCGACTCGCCTACGTCAGTCTCCTT
```

Motivating Problem: Population genetics II

Population genetics models

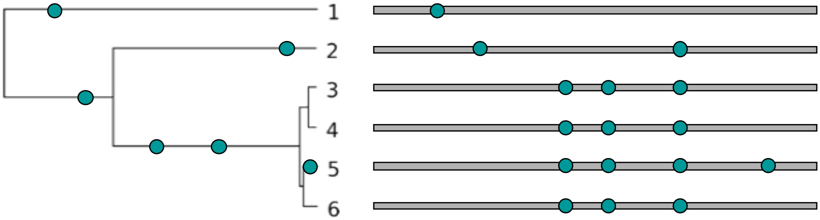
A generative model for DNA sequence data should account for

- Mutation
- Recombination
- Natural selection
- Genetic drift
- Demographic history
  (population expansion, contraction, bottlenecks, . . .)
- Population structure
- . . .

All of these processes are captured through their effects on the *gene genealogy* of a sample.

Motivating Problem: Population genetics III

The genealogy is a latent / hidden / unobserved variable; we need to integrate over it.



For a model with parameters $\boldsymbol{\theta}$ we want to compute

$$L(\boldsymbol{\theta}) = \mathbb{P}(D; \boldsymbol{\theta}) = \int \mathbb{P}(\mathcal{G})\mathbb{P}(D|\mathcal{G}; \boldsymbol{\theta}) \, d\mathcal{G}.$$

Motivating Problem: Hypothesis Testing

Testing Example: Chi-Squared Test of goodness of fit

- $T = \sum_{k=1}^{K} \frac{(O_k - E_k)^2}{E_k}$
- Asymptotic argument: $T \overset{d}{\approx} \chi^2_{K-1}$ under regularity conditions.

What if we *don't* have many observations of every category?

What if we want to know whether the *medians* of two populations are *significantly different*?

What if we don't know the form of their distributions?

Motivating Problem: Confidence Intervals

Constructing confidence intervals requires knowledge of sampling distributions.

Confidence Interval: Medians

- $X_1, X_2, \ldots, X_n \overset{\text{iid}}{\sim} f_X$.
- $X_{[1]} \leq X_{[2]} \leq \cdots \leq X_{[n]}$ are the associated order statistics.
- $T = X_{[(n+1)/2]}$ is the sample median.
- How can we construct a confidence interval for the median of f_X?
- What if we don't even know the form of f_X?

Motivating Problem: Bayesian Inference

Bayesian statistics

- Data $\mathbf{y}_1, \ldots, \mathbf{y}_n$ and model $f(\mathbf{y}_i|\boldsymbol{\theta})$ where $\boldsymbol{\theta}$ is some parameter of interest.

  Likelihood $L(\boldsymbol{\theta}; \mathbf{y}_1, \ldots, \mathbf{y}_n) = \prod_{i=1}^{n} f(\mathbf{y}_i|\boldsymbol{\theta})$

- In the Bayesian framework $\boldsymbol{\theta}$ is a random variable with prior distribution $f^{\mathrm{prior}}(\boldsymbol{\theta})$. After observing $\mathbf{y}_1, \ldots, \mathbf{y}_n$, the posterior density of f is

$$\begin{aligned} f^{\mathrm{post}}(\boldsymbol{\theta}) &= f(\boldsymbol{\theta}|\mathbf{y}_1, \ldots, \mathbf{y}_n) \\ &= \frac{f^{\mathrm{prior}}(\boldsymbol{\theta}) L(\boldsymbol{\theta}; \mathbf{y}_1, \ldots, \mathbf{y}_n)}{\int_{\Theta} f^{\mathrm{prior}}(\boldsymbol{\vartheta}) L(\boldsymbol{\theta}; \mathbf{y}_1, \ldots, \mathbf{y}_n)\, d\boldsymbol{\vartheta}} \end{aligned}$$

- Often this is intractable—we need an approximation.

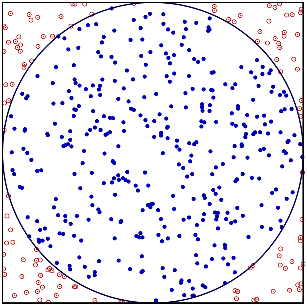
Simulation-based Methods

- Doing statistics backwards:

  Representing the solution of a problem as a parameter of a hypothetical population, and using a random sequence of numbers to construct a sample of the population, from which statistical estimates of the parameter (p values, confidence intervals, or other quantities of interest) can be obtained.

Preliminary Example: Raindrop experiment for π

- Consider "uniform rain" on the square $[-1,1] \times [-1,1]$, i.e. the two coordinates $X, Y \overset{\text{iid}}{\sim} \mathrm{U}[-1,1]$.



- Probability that a rain drop falls in the circle is

$$\mathbb{P}(\text{drop within circle}) = \frac{\text{area of the unit circle}}{\text{area of the square}} = \frac{\iint_{\{x^2+y^2\le 1\}} 1\,dxdy}{\iint_{\{-1\le x,y\le 1\}} 1\,dxdy} = \frac{\pi}{2\cdot 2} = \frac{\pi}{4}.$$

Preliminary Example: Raindrop experiment for π

- Given π, we can compute $\mathbb{P}(\text{drop within circle}) = \frac{\pi}{4}$.
- Given n independent raindrops, the number of rain drops falling in the circle, Z_n is a binomial random variable:

$$Z_n \sim \mathrm{Bin}\left(n, p = \frac{\pi}{4}\right).$$

- So we can estimate p with

$$\widehat{p} = \frac{Z_n}{n},$$

- and π by

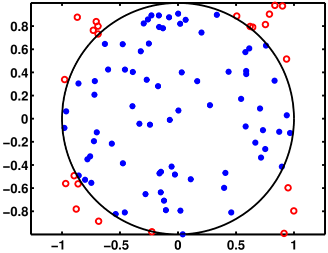
$$\widehat{\pi} = 4\widehat{p} = 4 \cdot \frac{Z_n}{n}.$$

Preliminary Example: Raindrop experiment for π

- Result obtained for
  $n = 100$ raindrops:
  77 points inside the circle.
- Resulting estimate of π is

$$\widehat{\pi} = \frac{4 \cdot Z_n}{n} = \frac{4 \cdot 77}{100} = 3.08,$$

(rather poor estimate).

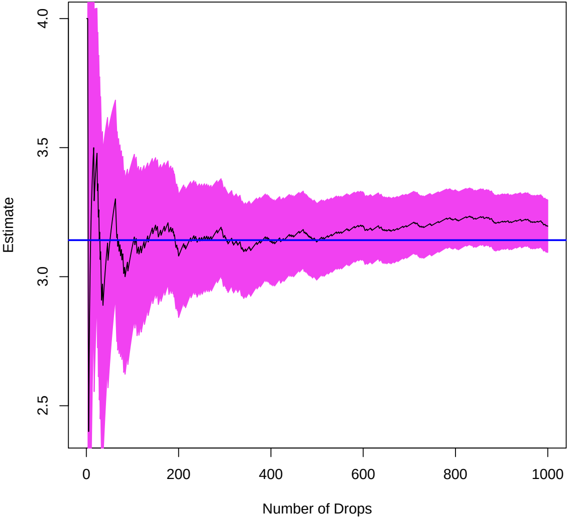


- However: the *law of large numbers* **guarantees** that

$$\widehat{\pi}_n = \frac{4 \cdot Z_n}{n} \rightarrow \pi$$

almost surely for $n \rightarrow \infty$.

Preliminary Example: Raindrop experiment for π



Preliminary Example: Raindrop experiment for π

- How fast does $\widehat{\pi}$ converge to π?
  Central limit theorem gives the answer.
- $(1-2\alpha)$ confidence interval for p ($\widehat{p}_n = Z_n/n$):

$$\left[\widehat{p}_n - z_{1-\alpha}\sqrt{\frac{\widehat{p}_n(1-\widehat{p}_n)}{n}}, \widehat{p}_n + z_{1-\alpha}\sqrt{\frac{\widehat{p}_n(1-\widehat{p}_n)}{n}}\right]$$

- $(1-2\alpha)$ confidence interval for π ($\widehat{\pi}_n = 4\widehat{p}_n$):

$$\left[\widehat{\pi}_n - z_{1-\alpha}\sqrt{\frac{\widehat{\pi}_n(4-\widehat{\pi}_n)}{n}}, \widehat{\pi}_n + z_{1-\alpha}\sqrt{\frac{\widehat{\pi}_n(4-\widehat{\pi}_n)}{n}}\right]$$

- Width of the interval is $O(n^{-1/2})$, thus speed of convergence $O_{\mathbb{P}}(n^{-1/2})$.

Preliminary Example: Raindrop experiment for π

Recall the two core elements of this example:

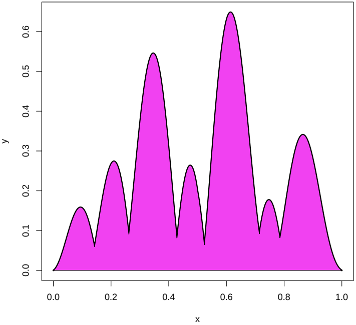
1. Write the quantity of interest (here π) as an expectation:

$$\pi = 4\mathbb{P}(\text{drop within circle}) = \mathbb{E}\left(4 \cdot \mathbb{I}_{\{\text{drop within circle}\}}\right)$$

2. Replace this algebraic representation with a sample approximation.
   - SLLN guarantees that the sample approximation converges to the algebraic representation.
   - CLT gives information about the speed of convergence.

The Generalisation to Monte Carlo Integration

$$f\ :\ [0,1] \to [0,1]$$



$$\int_0^1 f(x)\ dx = \int_0^1 \int_0^{f(x)} 1\ dt\ dx = \iint\limits_{\{(x,t):t\leq f(x)\}} 1 dt\ dx = \frac{\iint\limits_{\{(x,t):t\leq f(x)\}} 1\ dt\ dx}{\iint\limits_{\{0\leq x,t\leq 1\}} 1 dt\ dx}.$$

Comparison of the speed of convergence

- Monte Carlo integration is $O_{\mathbb{P}}(n^{-1/2})$.
- Numerical integration of a *one-dimensional* function by Riemann sums is $O(n^{-1})$.
- Monte Carlo does not compare favourably for one-dimensional problems.
- However:
  - Monte Carlo estimates are often *unbiased*.
  - Order of convergence of Monte Carlo integration is *independent* of dimension.
  - Order of convergence of numerical integration techniques deteriorates with increasing dimension.

Monte Carlo methods can be a good choice for high-dimensional integrals.

Views of Simulation-based Inference

Direct approximation of a quantity of interest.

- Careful construction of random experiment for particular task at hand.
- Justify with a dedicated argument in each case.

Approximation of *integrals* of interest.

- Represent quantity of interest as expectation w.r.t. some f.
- Use sample average to approximate expectation.
- Appeal to SLLN and CLT.

Approximation of *distributions* of interest.

- Represent quantity of interest as a function of distribution f.
- Use empirical measure of sample to approximate f.
- Appeal to Glivenko–Cantelli theorem.

Theoretical Motivation of Sample Approximation

Theorem (Strong Law of Large Numbers)

Let $X_1, X_2, \ldots \overset{iid}{\sim} f$*, and let* $\varphi : E \to \mathbb{R}$ *with* $\mathbb{E}\left[|\varphi(X_1)|\right] < \infty$*.*
Then:

$$\frac{1}{n}\sum_{i=1}^{n}\varphi(X_i) \xrightarrow{a.s.} \mathbb{E}\left[\varphi(X_1)\right].$$

Theorem (Central Limit Theorem)

Let $X_1, \ldots \overset{iid}{\sim} f_X$ *and let* $\varphi : E \to \mathbb{R}^k$ *with* $\Sigma = \mathbb{V}ar\left[\varphi(X)\right] < \infty$*.*
Then as $n \to \infty$*:*

$$\sqrt{n}\left[\frac{1}{n}\sum_{i=1}^{n}\varphi(X_i) - \mathbb{E}\left[\varphi(X_1)\right]\right] \overset{\mathcal{D}}{\to} N(\mathbf{0}, \Sigma).$$

Theoretical Motivation of Sample Approximation

Theorem (Glivenko–Cantelli)

Let $X_1, \ldots \overset{iid}{\sim} f_X$ *have cdf* F_X*.*
Let

$$F_n(x) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}_{(-\infty,x]}(X_i).$$

Then as $n \to \infty$*:*

$$\sup_x |F_n(x) - F(x)| \xrightarrow{a.s.} 0.$$

Part 1— Section 2

Randomized Testing

Randomized Testing

- One simple example of computer intensive statistics.
- We'll revisit *how* we can implement these things later.
- Art of testing: find a set R_α such that

$$\mathbb{P}\left(T \in R_\alpha; H_0\right) = \alpha$$

and

$$\mathbb{P}\left(T \in R_\alpha; H_1\right) > \alpha.$$

- What if we don't know the distribution of the test statistic, f_T?

Is a Die Fair?

- Given n rolls of a die, we want to establish whether it's fair.
- Canonical example of a χ^2-test...
- Compute

$$T = \sum_{k=1}^{K} \frac{(O_k - E_k)^2}{E_k}$$

- $T \overset{\text{approx}}{\sim} \chi^2_{K-1}$ by asymptotic arguments.
- What if the asymptotics don't hold?

A Randomized Goodness of Fit Test

- Imagine we have 9 measured rolls (and can't easily obtain more):

Value	1	2	3	4	5	6
Count	0	1	0	2	2	4

- If the die is fair we *expect* 1.5 observations of each value.
- The test statistic is:

$$T = \frac{1.5^2 + 0.5^2 + 1.5^2 + 0.5^2 + 0.5^2 + 2.5^2}{1.5} = 7\frac{2}{3}$$

- The asymptotics *certainly* don't hold:

$$(O_k - E_k)^2 \in \{0.5^2, 1.5^2, 2.5^2, 3.5^2, 4.5^2, 5.5^2, 6.5^2, 7.5^2\}.$$

- But we can *simulate* from H_0.

An R Implementation

Randomized Goodness of Fit Testing: Setup

```
p  <- 1/6 * c(1,1,1,1,1,1)
n  <- 9
r  <- 10000
ob <- rmultinom(r,n,p)
ex <- n*p
T  <- colSums((ob - ex)^2/ex)
```

How many elements in T are larger than the observed value?

Randomized Goodness of Fit Testing: Comparison

```
t <- 23/3
m <- sum(T >= (t - 1E-9)) #T discrete
print(m/r)
```

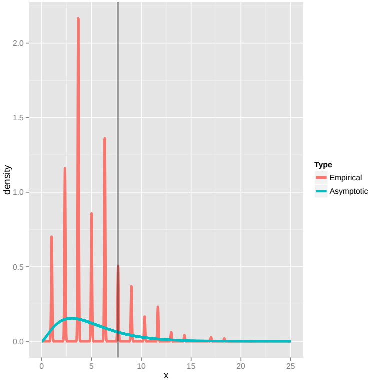
Randomized testing: results

Does this look fair? Vote! Vevox.app 170–356–838

Value	1	2	3	4	5	6
Count	0	1	0	2	2	4

Randomized testing: results

Empirical p-value:
0.1848
Asymptotic p-value:
0.1860



Randomized Test in General

- Given a hypothesis, H_0 and an alternative, H_1, and data $\boldsymbol{x}$ which realises $\boldsymbol{X}$ under H_0:
  - Obtain a realisation $\boldsymbol{u}$ of $\boldsymbol{U}$ ($\boldsymbol{U}|\boldsymbol{X} \sim f_{\boldsymbol{U}|\boldsymbol{X}}$ from some known distribution).
  - Compute R_α such that $\mathbb{P}\left((\boldsymbol{X}, \boldsymbol{U}) \in R_\alpha; H_0\right) = \alpha$.
  - Reject H_0 if $(\boldsymbol{x}, \boldsymbol{u}) \in R_\alpha$.

Goodness of Fit Test in General Form

- Let $f_{\boldsymbol{U}|\boldsymbol{X}}(\boldsymbol{u}|\mathbf{x}) = \prod_{i=1}^{r} f_{T(\boldsymbol{X})}(u_i; H_0)$.

  In practice: sample $\boldsymbol{Z}_i \overset{\text{iid}}{\sim} f_{\boldsymbol{X}}(\cdot; H_0)$ and set $U_i = T(\boldsymbol{Z}_i)$, where $T(\boldsymbol{X})$ is a real-valued summary of $\boldsymbol{X}$.
- Let $R_\alpha = \{(\mathbf{x}, \boldsymbol{u}) : T(\mathbf{x}) > u_{[r(1-\alpha)]}\}$, where $u_{[i]}$ is the i^{th} order statistic.

Are Those Medians Different (Part I)?

- Consider testing for different medians:

$$\begin{aligned} H_0: &\quad X_1, \ldots, X_{n_X} \overset{\text{iid}}{\sim} f_X(\cdot; m) \quad Y_1, \ldots, Y_{n_Y} \overset{\text{iid}}{\sim} f_Y(\cdot; m) \\ H_1: &\quad X_1, \ldots, X_{n_X} \overset{\text{iid}}{\sim} f_X(\cdot; m) \quad Y_1, \ldots, Y_{n_Y} \overset{\text{iid}}{\sim} f_Y(\cdot; m') \end{aligned}$$

- And we'll assume a particular example for the form of the two distributions:

$$f_X(x; m) = f_Y(x; m) = \frac{1}{2} \exp(-|x - m|)$$

- Letting $\widetilde{X} = X_{[(n_X+1)/2]}$ and $\widetilde{Y} = Y_{[(n_Y+1)/2]}$:

$$\begin{aligned} \widetilde{X} - \widetilde{Y} &= (\widetilde{X} - m) - (\widetilde{Y} - m) \\ &= (X - m)_{[(n_X+1)/2]} - (Y - m)_{[(n_Y+1)/2]} \end{aligned}$$

- So the distribution of $\widetilde{X} - \widetilde{Y}$ is *independent* of $m | H_0$.

- A Randomized test:
  - Let $T = \widetilde{X} - \widetilde{Y}$.
  - Draw $i = 1, \ldots, r$ copies of $\boldsymbol{X}$ and $\boldsymbol{Y}$ with $m = 0$:

$$X'^{,j}_{1,\ldots,n_X} \overset{\text{iid}}{\sim} f_X(\cdot;0),$$
$$Y'^{,j}_{1,\ldots,n_Y} \overset{\text{iid}}{\sim} f_Y(\cdot;0).$$

  - Compute the difference between their medians:

$$i = 1, \ldots, r: \qquad T'_i = X'^{,i}_{[(n_X+1)/2]} - Y'^{,i}_{[(n_Y+1)/2]}.$$

  - Let $p = (1 + |\{i : T'_i \geq T\}|)/(r+1)$.
  - Reject H_0 if $p < \alpha$ (a one-sided test; $H_1 :\ m' < m$).

But surely this is cheating: what if we *don't* know so much (like f_X and f_Y)?

Permutation Tests

- Consider the hypotheses:

$$H_0: \quad X_1, \ldots, X_{n_X} \overset{\text{iid}}{\sim} f_X(\cdot) \qquad Y_1, \ldots, Y_{n_Y} \overset{\text{iid}}{\sim} f_Y(\cdot)$$
$$F_X^{-1}(0.5) = F_Y^{-1}(0.5)$$
$$H_1: \quad X_1, \ldots, X_{n_X} \overset{\text{iid}}{\sim} f_X(\cdot) \qquad Y_1, \ldots, Y_{n_Y} \overset{\text{iid}}{\sim} f_Y(\cdot)$$
$$F_X^{-1}(0.5) \neq F_Y^{-1}(0.5)$$

  where f_X and f_Y are *unknown*.
- Here, F_X^{-1} and F_Y^{-1} are assumed to exist.
- Sample medians are natural test statistics, but:
  - We don't know their distribution under H_0.
  - And can't sample from that distribution.
- What can we do?

Permutation Tests

- Let $\boldsymbol{Z} = (X_1, \ldots, X_{n_X}, Y_1, \ldots, Y_{n_Y})$ be an $n = n_X + n_Y$ vector.
- Now let

$$T(\boldsymbol{Z}) = \mathrm{median}(Z_1, \ldots, Z_{n_X}) - \mathrm{median}(Z_{n_X+1}, \ldots, Z_n)$$

- And let $\pi \in \mathcal{P} \subseteq \{1, \ldots, n\}^n$ denote a permutation, writing:

$$\pi\boldsymbol{Z} := (Z_{\pi_1}, Z_{\pi_2}, \ldots, Z_{\pi_n})$$

- Now, under H_0:

$$\forall \pi \in \mathcal{P} : \qquad T(\pi\boldsymbol{Z}) \overset{\mathcal{D}}{=} T(\boldsymbol{Z})$$

- So if $T(\boldsymbol{Z}) > T(\pi\boldsymbol{Z})$ for $100(1-\alpha)\%$ of π we can reject H_0.
- We *just* need to compute $T(\pi\boldsymbol{Z})$ for every $\pi \in \mathcal{P}\ldots$

A Randomized Permutation Test

- We can sample elements uniformly from $\mathcal{P}$:
  - Sample $\pi_1 \sim \mathrm{U}\left(1, \ldots, n\right)$.
  - Sample $\pi_2 \sim \mathrm{U}\left(\{1, \ldots, n\} \setminus \{\pi_1\}\right)$.

    $$\vdots$$
  - Sample $\pi_n \sim \mathrm{U}\left(\{1, \ldots, n\} \setminus \{\pi_1, \ldots, \pi_{n-1}\}\right)$.
- We can do this many times to approximate the law of $T(\pi \boldsymbol{z})$ when $\pi \sim \mathrm{U}(\mathcal{P})$:
  - Sample $\boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_k \overset{\text{iid}}{\sim} \mathrm{U}(\mathcal{P})$.
  - Compute $T_1 = T(\boldsymbol{\pi}_1 \boldsymbol{z}), \ldots, T_k = T(\boldsymbol{\pi}_k \boldsymbol{z})$.
  - Use the empirical distribution of $(T_1, \ldots, T_k)$ to approximate the law of $T(\boldsymbol{\pi z})$.
- This provides a general strategy for nonparametric testing.

Part 1— Section 3

Bootstrap Methods

Bootstrap Methods

- Randomized tests: use empirical distribution of T.
- Permutation tests: use *resampling*-based empirical distribution of T.
- Bootstrap methods: use *resampling*-based empirical distribution of $\hat{\theta}$ to characterise the sampling distribution of $\hat{\theta}$.

The Bootstrap Ansatz

If $X_1, \ldots, X_n \overset{\text{iid}}{\sim} F_X$ and n is large then "$\hat{F}_X^n \approx F$"
$\Longrightarrow$ sampling from $\hat{F}_X^n$ is "close" to sampling from F
$\Longrightarrow$ samples from $\hat{F}_X^n$ might be suitable for approximating F!

The Basis of the Bootstrap

- Given a simple random sample $X_1, \ldots, X_n$
- Repeat the following for $b = 1, \ldots, B$:
  - Sample n times from $\hat{F}^n_X(x)$ i.e. sample n times uniformly *with replacement* from $X_1, \ldots, X_n$ to obtain $\hat{X}^b_1, \ldots, \hat{X}^b_n$.
- For a function of interest $g : E^n \to \mathbb{R}$, approximate the distribution of g under F using the sample $g(\hat{X}^1_1, \ldots, \hat{X}^1_n), \ldots, g(\hat{X}^B_1, \ldots, \hat{X}^B_n)$.
- Glivenko–Cantelli (and extensions) tells us that $\hat{F}^n_X(x) \xrightarrow{a.s.} F_X(x)$.

N.B. Regularity conditions must hold in order for this to work.

Approximating the Sampling Distribution of the Median

- Given $X_1, \ldots, X_n$ a simple random sample:
- Compute $T = \text{median}(X_1, \ldots, X_n)$.
- For $b = 1, \ldots, B$:
    - Sample n times with replacement from $X_1, \ldots, X_n$ to obtain $\hat{X}_1^b, \ldots, \hat{X}_n^b$.
    - Compute $\hat{T}^b = \text{median}(\hat{X}_1^b, \ldots, \hat{X}_n^b)$.
- Treat the empirical distribution of $\hat{T}^1, \ldots, \hat{T}^B$ as a proxy for the sampling distribution of T.

Bootstrap Bias Correction

- Given $x_1, \ldots, x_n$ and,
- estimator $T : E^n \to \mathbb{R}$ of θ,
- compute $t = T(x_1, \ldots, x_n)$.
- For $b = 1, \ldots, B$
  - Sample n times with replacement from $X_1, \ldots, X_n$ to obtain $\hat{X}_1^b, \ldots, \hat{X}_n^b$.
  - Compute $\hat{T}^b = T(\hat{X}_1^b, \ldots, \hat{X}_n^b)$.
- Treat the empirical distribution of $\hat{T}^1 - t, \ldots, \hat{T}^B - t$ as a proxy for the sampling distribution of $T(X_1, \ldots, X_n) - \theta$.
- Obtain *bias-corrected* estimate:

$$t - \frac{1}{B}\sum_{b=1}^{B}(\hat{T}^b - t) = 2t - \frac{1}{B}\sum_{b=1}^{B}\hat{T}^b.$$

Naïve Bootstrap Confidence Intervals 1: The Asymptotic Approach

- For some T we might expect T to have an asymptotically normal distribution.
- So, estimate its variance:

$$\hat{\sigma}_T^2 = \frac{1}{B-1} \sum_{b=1}^{B} \left(\hat{T}^b - \frac{1}{B} \sum_{b=1}^{B} \hat{T}^b \right)^2$$

- And use the normal confidence interval:

$$\left[T - z_{\alpha/2} \hat{\sigma}_T , T + z_{\alpha/2} \hat{\sigma}_T \right]$$

  with approximate coverage α.
- Depends on asymptotic normality.
- Further approximation for finite samples.

Naïve Bootstrap Confidence Intervals 2: Bootstrap Percentile Confidence Intervals

- We could use the bootstrap distribution of T directly:

$$[\hat{T}^{[B(\alpha/2)]}, \hat{T}^{[B(1-\alpha/2)]}]$$

- These are known as *bootstrap percentile confidence intervals*.
- Depend on the *bootstrap* approximation; no additional approximations.

Bootstrap "pivotal" Confidence Intervals

- Using bootstrap approximations of (approximate) pivots can be more elegant.
- Assume that T is an estimator of some real population parameter, θ.
- Define $R = T - \theta$.
- Let F_R denote the cdf of R, then:

$$\begin{aligned} \mathbb{P}(L \leq \theta \leq U) &= \mathbb{P}(L - T \leq \theta - T \leq U - T) \\ &= \mathbb{P}(T - U \leq R \leq T - L) \\ &= F_R(T - L) - F_R(T - U). \end{aligned}$$

Suggests using:

$$[T - F_R^{-1}(1 - \alpha/2), T - F_R^{-1}(\alpha/2)]$$

- We can't use this interval directly because we don't know F_R and we certainly don't know F_R^{-1}.

Bootstrap "pivotal" Confidence Intervals

- We can invoke the bootstrap idea again:
- Compute $T = g(X_1, \dots, X_n)$.
- For $b = 1, \dots, B$:
  - Sample n times with replacement from $X_1, \dots, X_n$ to obtain $\hat{X}_1^b, \dots, \hat{X}_n^b$.
  - Compute $\hat{T}^b = g(\hat{X}_1^b, \dots, \hat{X}_n^b)$.
- Claim that "$\hat{T}^1, \dots, \hat{T}^B$ are to T as T is to θ".
- Set $\hat{R}^b = \hat{T}^b - T$.
- Use the empirical distribution, $\hat{F}_R$, of $\hat{R}^1, \dots, \hat{R}^B$ instead of F_R:

$$[T - \hat{F}_R^{-1}(1 - \alpha/2), T - \hat{F}_R^{-1}(\alpha/2)]$$

Summary of Part 1

- Motivation: Bayesian inference, Fisherian inference, . . .
- Towards simulation-based inference (see later).
- Randomized Tests
- Permutation Tests
- Bootstrap Characterisation of Estimators.
- Bootstrap Confidence Intervals.
- Young, G. A. (1994) Bootstrap: More than a stab in the dark? Statistical Science, 9, 382–395.
- Davison, A. C., Hinkley, D. V. and Young, G. A. (2003) Recent developments in bootstrap methodology. Statistical Science, 18, 141–157.

Part 2

Simulation and the Monte Carlo Method

Simulation

- We've seen *motivation* of simulation for inference.
- We've seen *examples* of simulation-based methods.
- Now we need methods for simulation.

Part 2— Section 4

The Monte Carlo Method

Monte Carlo Method

- A generic scheme for approximating expectations.
- To approximate $I = \mathbb{E}_f\left[\varphi(X)\right]$,
- Draw $X_1, \ldots, X_n \overset{\text{iid}}{\sim} f$,
- Use $\hat{I}_{\text{mc}} = \frac{1}{n}\sum_{i=1}^{n}\varphi(X_i)$.
- Convergence follows from SLLN, CLT, . . .

Recall: The Three Views of the Monte Carlo Method

Direct Approximation Design an experiment such that:

$$\varphi(X) \sim f_{\varphi(X)}$$

constructed such that it has the expectation of interest.

Integral Approximation We're interested in

$$\mathbb{E}_f\left[\varphi(X)\right]$$

and know how to approximate such.

Distributional Approximation We're interested in

$$\mathbb{E}_f\left[\varphi(X)\right]$$

so obtain an approximation of f with respect to which we can compute expectations.

Contrasting Views of Monte Carlo

- Usual explanation of the Monte Carlo Method, with $X_1, \dots \overset{\text{iid}}{\sim} f$ approximating the integral:

$$\frac{1}{n}\sum_{i=1}^{n} \varphi(X_i) \xrightarrow{a.s.} \mathbb{E}_f\left[\varphi(X)\right]$$

- Another perspective, approximate the distribution:
  - let $\hat{f}^n = \frac{1}{n}\sum_{i=1}^{n} \delta_{X_i}$
  - if $\hat{f}^n \Rightarrow f$
  - then we automatically have that

$$\mathbb{E}_{\hat{f}^n}\left[\varphi(X)\right] \to \mathbb{E}_f\left[\varphi(X)\right]$$

for every continuous bounded φ.

Part 2— Section 5

PRNGs

Problem: (how) can computers produce random numbers?

von Neumann's perspective

> *Any one who considers artithmetical methods of reproducing random digits is, of course, in a state of sin. . . . there is no such thing as a random number—there are only methods of producing random numbers, and a strict arithmetic procedure is of course not such a method.*

As in so many other areas, von Neumann was completely correct.

Three Resolutions of this Philosophical Paradox

1. Use Exogeneous Randomness (TRNGs)
   See `www.random.org` or `http://en.wikipedia.org/wiki/Hardware_random_number_generator`.
2. Pseudorandom Number Generators (PRNGs; c.f. *Statistical Computing* module)
   Sacrifice randomness whilst mimicking its *relevant statistical properties*.
3. Quasirandom Number Sequences (QRNSs)
   Sacrifice randomness in exchange for *minimising discrepancy*.

All have advantages and disadvantages; we'll focus on PRNGs.

Part 2— Section 6

Sampling From Distributions

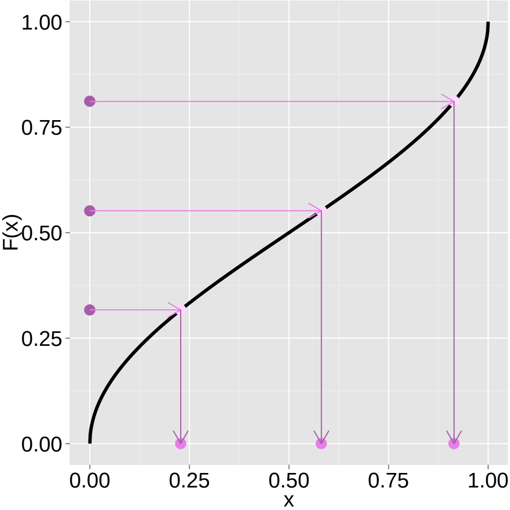
Transformation Methods

- Assume we have a *good* PRNG.
- How can we obtain (pseudo)samples from other distributions?
- General framework:
  - Treat output of PRNG as a stream of iid $U[0, 1]$ RVs.
  - Use laws of probability to transform these to obtain RVs with other distributions.
  - Treat transformed PRNG output as RVs of the target distribution.
- But, how?

Inversion Sampling

The Inversion method

Let $U \sim \mathrm{U}[0,1]$ and
let F be an invertible CDF.
Then $F^{-1}(U)$ has the CDF F.



Inversion Sampling

The Inversion method

Let $U \sim \mathrm{U}[0,1]$ and F be an invertible CDF.
Then $F^{-1}(U)$ has the CDF F.

Inversion Sampling: A simple algorithm for drawing $X \sim F$

1. Draw $U \sim \mathrm{U}[0,1]$.
2. Set $X = F^{-1}(U)$.

Example: Exponential distribution

The exponential distribution with rate $\lambda > 0$ has the CDF $(x \geq 0)$

$$\begin{aligned} F_\lambda(x) &= 1 - \exp(-\lambda x) \\ F_\lambda^{-1}(u) &= -\log(1-u)/\lambda. \end{aligned}$$

So we have a simple algorithm for drawing $X \sim \operatorname{Exp}(\lambda)$:

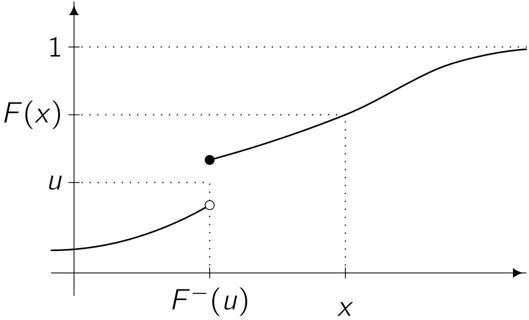
1. Draw $U \sim \mathrm{U}[0,1]$.
2. Set $X = -\dfrac{\log(1-U)}{\lambda}$.

Actually, setting $X = -\dfrac{\log(U)}{\lambda}$ makes more sense.

The Generalised Inverse of the CDF

Generalised inverse of the CDF

$$F^{-}(u) := \inf\{x : \; F(x) \geq u\}$$



Replacing F^{-1} with F^{-} yields a generally-applicable inversion sampling algorithm — key is $F^{-}(u) \leq x \Leftrightarrow u \leq F(x)$.

Box–Muller: Fast Normally-Distributed Random Variables

- Consider (X_1, X_2) their polar representation (R, θ):

$$X_1 = R \cdot \cos(\theta), \qquad X_2 = R \cdot \sin(\theta)$$

- The following equivalence holds (with θ, R independent): $X_1, X_2 \overset{\text{iid}}{\sim} \mathrm{N}(0,1) \iff \theta \sim \mathrm{U}[0, 2\pi]$ and $R^2 \sim \mathrm{Expo}(1/2)$
- Given $U_1, U_2 \overset{\text{iid}}{\sim} \mathrm{U}[0,1]$ set

$$R = \sqrt{-2\log(U_1)}, \qquad \theta = 2\pi U_2.$$

- By substitution

$$X_1 = \sqrt{-2\log(U_1)} \cdot \cos(2\pi U_2),$$
$$X_2 = \sqrt{-2\log(U_1)} \cdot \sin(2\pi U_2).$$

Box–Muller: Algorithm

Box–Muller method

1. Draw
$$U_1, U_2 \overset{\text{iid}}{\sim} \mathrm{U}[0,1].$$
2. Set
$$\begin{aligned} X_1 &= \sqrt{-2\log(U_1)} \cdot \cos(2\pi U_2), \\ X_2 &= \sqrt{-2\log(U_1)} \cdot \sin(2\pi U_2). \end{aligned}$$
3. Output $X_1, X_2 \overset{\text{iid}}{\sim} \mathrm{N}(0,1)$.

The Limitations of Simple Transformations. . .

- When F^{-} is available and cheap to evaluate, inversion sampling is very efficient. But:
    - We often don't have access to F;
    - even if we do, F^{-} may be difficult/impossible to obtain.
    - The multivariate case can be even harder.
- Clever custom transformations:
    - are costly to develop,
    - require considerable ingenuity,
    - are completely infeasible in complicated scenarios.
- We need alternatives.

The Fundamental Theorem of simulation

Fundamental Theorem of Simulation

Sampling from a density f is equivalent to sampling uniformly from the area between f and the ordinal axes and discarding the "vertical" component.

- Follows from the identity

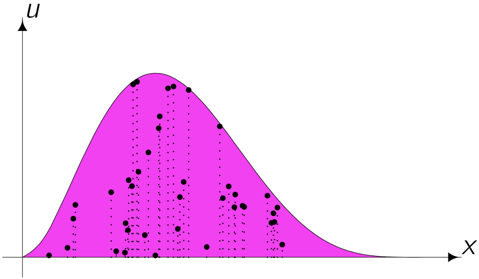
$$f(x) = \int_0^{f(x)} 1 \; du = \int_0^{\infty} \underbrace{1_{0<u<f(x)}}_{=f(x,u)} du.$$

- i.e. $f(x)$ can be interpreted as the marginal density of a uniform distribution on the area under the density $f(x)$:

$$\{(x, u) : \; 0 \leq u \leq f(x)\}.$$

First element of rejection sampling

- We can sample from f by sampling from the area under the density.



- If $(X, U) \sim \mathrm{U}\left(\{(x, u) : \ 0 \leq u \leq f(x)\}\right)$ then $X \sim f$.

Second Element of Rejection Sampling

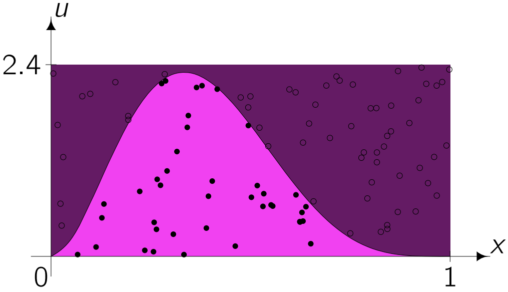
- Generally $\mathcal{G} = \{(x, u) : \ 0 \leq u \leq f(x)\}$ is complicated: we can't sample uniformly from it—at least not directly.
- Idea: Instead:
  - Sample from some $\mathcal{A} \supseteq \mathcal{G}$.
  - Keep only those points which lie within $\mathcal{G}$.
  - *Reject* the rest.

Example: Sampling from a Beta$(3, 5)$ distribution (1)

1. Draw (X, U) from the dark rectangle, i.e.:

$$X \sim \mathrm{U}(0,1) \qquad U \sim \mathrm{U}(0,2.4) \qquad X \perp U.$$

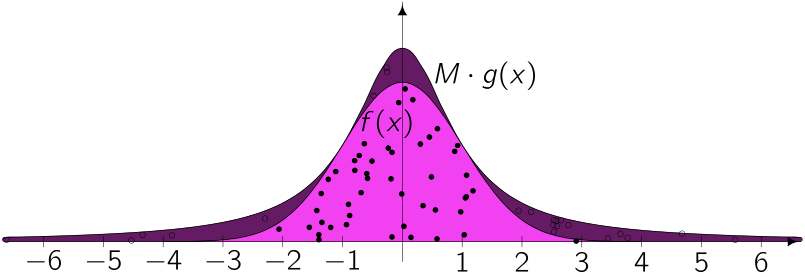
2. Accept X as a sample from f if (X, U) lies under the density.



Step 2 is equivalent to: Accept X if $U \leq f(X)$,
i.e. accept X with probability $\mathbb{P}(U \leq f(X) | X = x) = f(X)/2.4$.

Example: Sampling from a Beta(3, 5) distribution (2)

- Algorithm:
  1. Draw $X \sim \mathrm{U}(0, 1)$.
  2. Accept X as a sample from $\mathrm{Beta}(3, 5)$ w.p. $f(X)/2.4$.
- Not every density can be bounded by a box.
- Natural generalisation: replace M times $\mathrm{U}[0, 1]$ with M times another density g.



A General Algorithm

Algorithm: Rejection sampling

Given two densities f, g with $f(x) \leq M \cdot g(x)$ for all x, we can generate a sample from f by

1. Draw $X \sim g$.
2. Accept X as a sample from f with probability

$$\frac{f(X)}{M \cdot g(X)},$$

otherwise go back to step 1.

For $f(x) \leq M \cdot g(x)$ to hold for all x, f *cannot* have heavier tails than g.

A Useful Trick

Avoiding Unknown Constants

If we know only $\tilde{f}(x)$ and $\tilde{g}(x)$, where $f(x) = C \cdot \tilde{f}(x)$, and $g(x) = D \cdot \tilde{g}(x)$, we can carry out rejection sampling using acceptance probability

$$\frac{\tilde{f}(X)}{M \cdot \tilde{g}(X)}$$

provided $\tilde{f}(x) \leq M \cdot \tilde{g}(x)$ for all x.

Can be useful in Bayesian statistics:

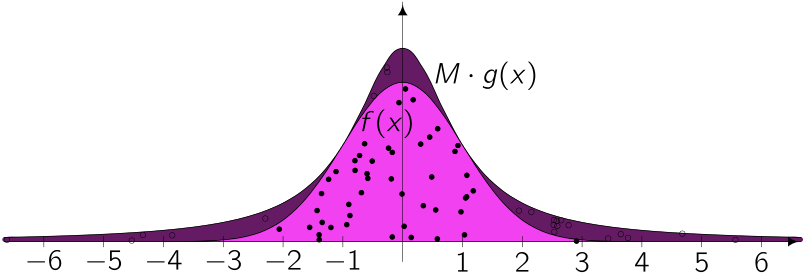
$$\begin{aligned} f^{\text{post}}(\theta) &= \frac{f^{\text{prior}}(\theta) L(\theta; \mathbf{y}_1, \ldots, \mathbf{y}_n)}{\int_\Theta f^{\text{prior}}(\vartheta) L(\vartheta; \mathbf{y}_1, \ldots, \mathbf{y}_n)\, d\vartheta} \\ &= C \cdot f^{\text{prior}}(\theta) L(\theta; \mathbf{y}_1, \ldots, \mathbf{y}_n). \end{aligned}$$

Example: Sampling from N(0, 1)

- Recall the N(0, 1) and Cauchy densities:

$$f(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right), \qquad g(x) = \frac{1}{\pi(1+x^2)}.$$

- For $M = \sqrt{2\pi} \cdot \exp(-1/2)$ we have that $f(x) \leq Mg(x)$. So we can use rejection sampling targeting f using g as proposal.



Non-example: Sampling from a Cauchy Distribution

- We cannot sample the other way round: from a Cauchy distribution using a Normal as proposal distribution.
- The Cauchy distribution has heavier tails than the Normal distribution: there is no $M \in \mathbb{R}$ such that

$$\frac{1}{\pi(1+x^2)} \leq M \cdot \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x^2}{2}\right).$$

Vevox.app 170–356–838

How would you sample from a Cauchy distribution?

Non-example: Sampling from a Cauchy Distribution

- We cannot sample the other way round: from a Cauchy distribution using a Normal as proposal distribution.
- The Cauchy distribution has heavier tails than the Normal distribution: there is no $M \in \mathbb{R}$ such that

$$\frac{1}{\pi(1+x^2)} \leq M \cdot \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x^2}{2}\right).$$

Vevox.app 170–356–838

How would you sample from a Cauchy distribution?

An Alternative to Rejection

- Rejection sampling discards many samples.
- This seems wasteful.
- Couldn’t we, instead, *weight* samples based on the acceptance probability?

The fundamental identities behind importance sampling

Assume that $g(x) > 0$ for (almost) all x with $f(x) > 0$:

$$\mathbb{P}(X \in \mathcal{X}) = \int_{\mathcal{X}} f(x)\,dx = \int_{\mathcal{X}} g(x)\underbrace{\frac{f(x)}{g(x)}}_{=:w(x)}\,dx = \int_{\mathcal{X}} g(x)w(x)\,dx.$$

Assume that $g(x) > 0$ for (almost) all x with $f(x) \cdot \varphi(x) \neq 0$

$$\mathbb{E}_f(\varphi(X)) = \int f(x)\varphi(x)\,dx = \int g(x)\underbrace{\frac{f(x)}{g(x)}}_{=:w(x)}\varphi(x)\,dx$$

$$= \int g(x)w(x)\varphi(x)\,dx = \mathbb{E}_g(w(X)\cdot\varphi(X)).$$

The fundamental identities behind importance sampling

- Consider $X_1, \ldots, X_n \sim g$ and $\mathbb{E}_g|w(X) \cdot \varphi(X)| < \infty$. Then

$$\frac{1}{n}\sum_{i=1}^{n} w(X_i)\varphi(X_i) \overset{a.s.}{\underset{}{\xrightarrow{n\to\infty}}} \mathbb{E}_g(w(X) \cdot \varphi(X))$$
$$\Longrightarrow \frac{1}{n}\sum_{i=1}^{n} w(X_i)\varphi(X_i) \overset{a.s.}{\underset{}{\xrightarrow{n\to\infty}}} \mathbb{E}_f(\varphi(X)).$$

- Thus we can estimate $\mu := \mathbb{E}_f(\varphi(X))$ by
  1. Sample $X_1, \ldots, X_n \sim g$,
  2. $\tilde{\mu} := \frac{1}{n}\sum_{i=1}^{n} w(X_i)\varphi(X_i)$.

The importance sampling algorithm

Algorithm: Importance Sampling

Choose g such that $\mathrm{supp}(g) \supseteq \mathrm{supp}(f \cdot \varphi)$.

1. For $i = 1, \ldots, n$:
    1. Generate $X_i \sim g$.
    2. Set $w(X_i) = \frac{f(X_i)}{g(X_i)}$.
2. Return

$$\tilde{\mu} = \frac{\sum_{i=1}^{n} w(X_i)\varphi(X_i)}{n}$$

as an estimate of $\mathbb{E}_f(\varphi(X))$.

- Importance sampling does not yield realisations from f, but a *weighted sample* (X_i, W_i), which can be used for estimating expectations $\mathbb{E}_f(\varphi(X))$, or approximating f itself.

Basic properties of the importance sampling estimate

- We have already seen that $\tilde{\mu}$ is consistent if $\operatorname{supp}(g) \supseteq \operatorname{supp}(f \cdot \varphi)$ and $\mathbb{E}_g|w(X) \cdot \varphi(X)| < \infty$, as

$$\tilde{\mu} := \frac{1}{n} \sum_{i=1}^{n} w(X_i)\varphi(X_i) \xrightarrow[n\to\infty]{a.s.} \mathbb{E}_f(\varphi(X))$$

- The expected value of the weights is $\mathbb{E}_g(w(X)) = 1$.
- $\tilde{\mu}$ is unbiased (see theorem below)

Theorem 2.2: Bias and Variance of Importance Sampling

$$\begin{aligned} \mathbb{E}_g(\tilde{\mu}) &= \mu, \\ \operatorname{Var}_g(\tilde{\mu}) &= \frac{\operatorname{Var}_g(w(X) \cdot \varphi(X))}{n}. \end{aligned}$$

Optimal proposals

Theorem (Optimal proposal)

The proposal distribution g that minimises the variance of $\tilde{\mu}$ is

$$g^*(x) = \frac{|\varphi(x)|f(x)}{\int |\varphi(t)|f(t)\;dt}.$$

- Theorem of little practical use: the optimal proposal involves $\int |\varphi(t)|f(t)\;dt$, which is the integral we want to estimate!
- Practical relevance:
  Choose g such that it is close to $|\varphi(x)| \cdot f(x)$.

Super-efficiency of importance sampling

- For the optimal g^* we have that

$$\mathrm{Var}_f\left(\frac{\varphi(X_1)+\cdots+\varphi(X_n)}{n}\right) > \mathrm{Var}_{g^\star}(\tilde{\mu}),$$

  if φ is not almost surely constant.

Superefficiency of importance sampling

The variance of the importance sampling estimate can be *less* than the variance obtained by sampling directly from the target f.

- Intuition: Importance sampling allows us to choose a g that focuses on areas which contribute most to $\int \varphi(x) f(x)\, dx$.
- Even sub-optimal proposals can be super-efficient.

Importance Sampling Example 1: Setup

Compute $\mathbb{E}_f|X|$ for $X \sim \mathrm{t}_3$ by ...

- (a) sampling directly from t_3.
- (b) using a t_1 distribution as proposal distribution.
- (c) using a $\mathrm{N}(0, 1)$ distribution as proposal distribution.

Vevox.app 170–356–838

Which of these methods is best?

Reminder:

$$g_{t_3}(x) = \frac{2}{\pi\sqrt{3}} \cdot \frac{1}{\left(1 + \frac{x^2}{3}\right)^2}, \qquad g_{t_1}(x) = \frac{1}{\pi} \cdot \frac{1}{1 + x^2}.$$

Importance Sampling Example 1: Setup

Compute $\mathbb{E}_f|X|$ for $X \sim \mathrm{t}_3$ by ...

- (a) sampling directly from t_3.
- (b) using a t_1 distribution as proposal distribution.
- (c) using a $\mathrm{N}(0,1)$ distribution as proposal distribution.

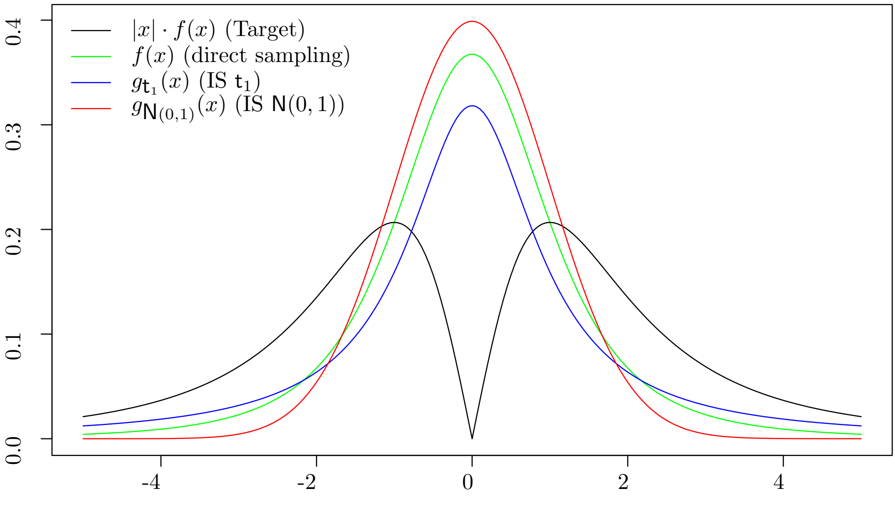
Vevox.app 170–356–838

Which of these methods is best?

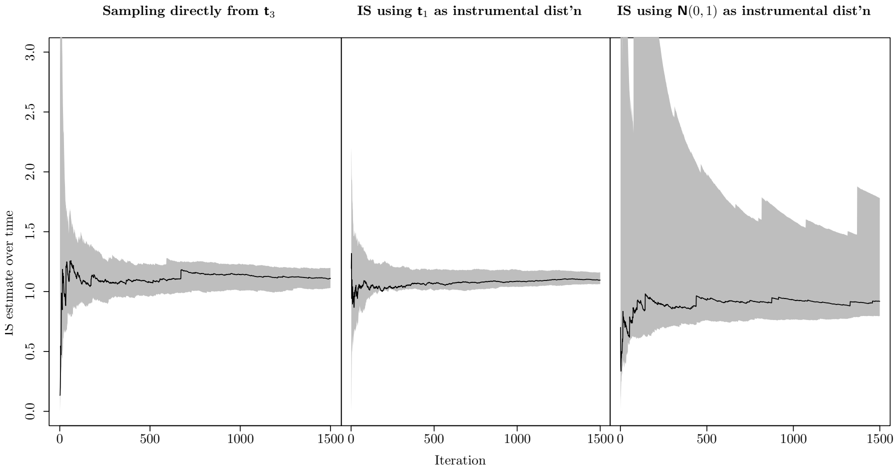
Reminder:

$$g_{t_3}(x) = \frac{2}{\pi\sqrt{3}} \cdot \frac{1}{\left(1+\frac{x^2}{3}\right)^2}, \qquad g_{t_1}(x) = \frac{1}{\pi} \cdot \frac{1}{1+x^2}.$$

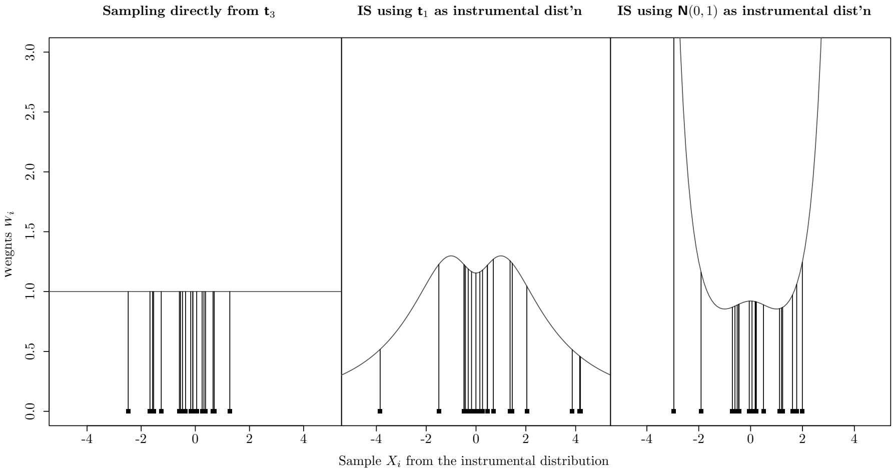
IS Example: Densities



IS Example: Estimates obtained



IS Example: Weights



Another Example: Rare Events (1)

Consider

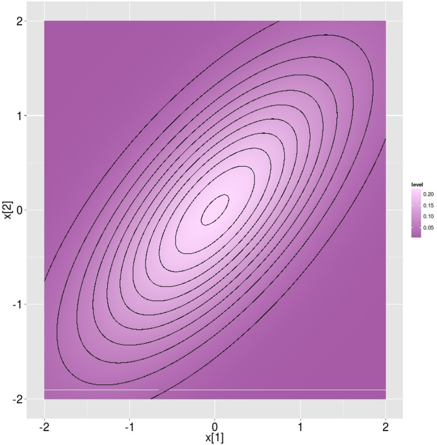
$$f(x,y) = \mathrm{N}\left(\begin{pmatrix} x \\ y \end{pmatrix}; \mu, \Sigma\right),$$

where

$$\mu = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad \Sigma = \begin{bmatrix} 1 & 0.7 \\ 0.7 & 1 \end{bmatrix}.$$

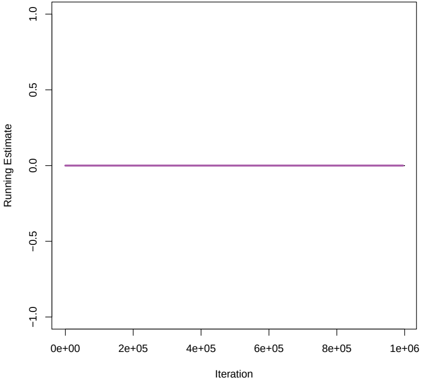
Consider

$$\varphi(x,y) = \mathbb{I}_{[4,\infty)}(x)\mathbb{I}_{[4,\infty)}(y).$$



Another Example: Rare Events (2)

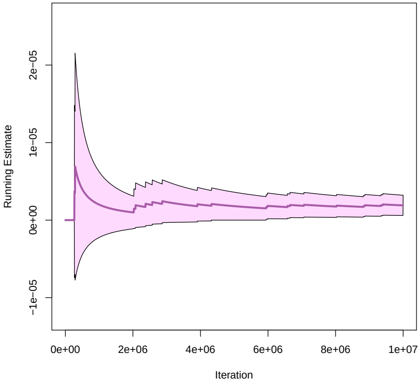
Using simple Monte Carlo with 1,000,000 samples from f:



shaded region shows *estimated* 99.7% confidence interval.

Another Example: Rare Events (3)

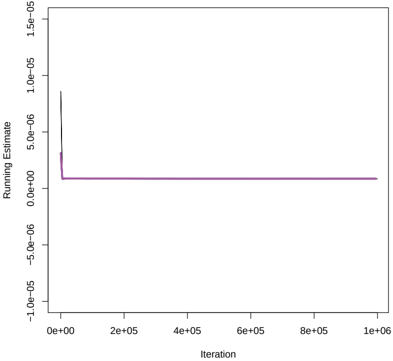
Using simple Monte Carlo with 10,000,000 samples from f:



shaded region shows *estimated* 99.7% confidence interval.

Another Example: Rare Events (4)

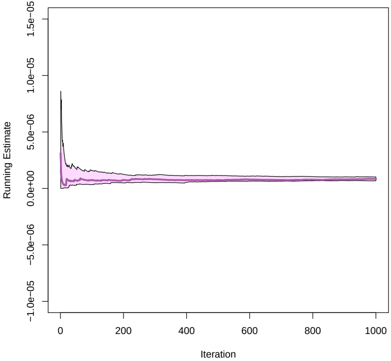
Using importance sampling with 1,000,000 samples from $g(x, y) = \exp(-(x-4)-(y-4))\mathbb{I}_{x\geq 4}\mathbb{I}_{y\geq 4}$:



shaded region shows range of 100 replications.

Another Example: Rare Events (5)

Using importance sampling with 1,000 samples from $g(x, y) = \exp(-(x-4)-(y-4))\mathbb{I}_{x\geq 4}\mathbb{I}_{y\geq 4}$:

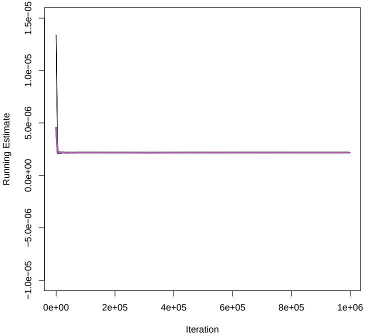


shaded region shows range of 100 replications.

Another Example: Rare Events (6)

Using importance sampling with 1,000,000 samples from

$$g(x,y) = \mathrm{N}\left(\left(\begin{array}{c} x \\ y \end{array}\right); \left(\begin{array}{c} 4 \\ 4 \end{array}\right), \Sigma \;\middle|\; x \geq 4,\ y \geq 4\right):$$

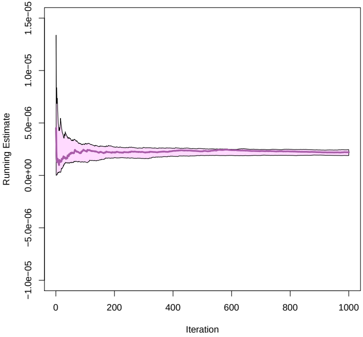


shaded region shows range of 100 replications.

Another Example: Rare Events (7)

Using importance sampling with 1,000 samples from

$$g(x,y) = \mathrm{N}\left(\begin{pmatrix} x \\ y \end{pmatrix}; \begin{pmatrix} 4 \\ 4 \end{pmatrix}, \Sigma \,\middle|\, x \geq 4,\ y \geq 4\right):$$



shaded region shows range of 100 replications.

We only need f up to a multiplicative constant.

- Assume $f(x) = C\tilde{f}(x)$. Then
  $$\tilde{\mu} = \frac{1}{n}\sum_{i=1}^{n} w(X_i)\varphi(X_i) = \frac{1}{n}\sum_{i=1}^{n} \frac{C\tilde{f}(X_i)}{g(X_i)}\varphi(X_i)$$
  C does not cancel out. Knowing $\tilde{f}(\cdot)$ is not enough.
- Idea: Estimate C using the sample, via $\sum_{i=1}^{n} w(X_i)$, i.e. consider the *self-normalised estimator*
  $$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} w(X_i)\varphi(X_i) \Big/ \frac{1}{n}\sum_{i=1}^{n} w(X_i)1$$
- Now we have that $\hat{\mu}$ does not depend on C:
  $$\hat{\mu} = \frac{\sum_{i=1}^{n} w(X_i)\varphi(X_i)}{\sum_{i=1}^{n} w(X_i)} = \frac{\sum_{i=1}^{n} \frac{\tilde{f}(X_i)}{g(X_i)}\varphi(X_i)}{\sum_{i=1}^{n} \frac{\tilde{f}(X_i)}{g(X_i)}},$$

The importance sampling algorithm (2)

Algorithm: Importance Sampling using self-normalised weights

Choose g such that $\mathrm{supp}(g) \supseteq \mathrm{supp}(f)$.

1. For $i = 1, \dots, n$:
    1. Generate $X_i \sim g$.
    2. Set $w(X_i) = \frac{f(X_i)}{g(X_i)}$.
2. Return

$$\hat{\mu} = \frac{\sum_{i=1}^{n} w(X_i)\varphi(X_i)}{\sum_{i=1}^{n} w(X_i)}$$

as an estimate of $\mathbb{E}_f(\varphi(X))$.

Basic properties of the self-normalised estimate

- $\hat{\mu}$ is consistent as

$$\hat{\mu} = \underbrace{\frac{\sum_{i=1}^{n} w(X_i)\varphi(X_i)}{n}}_{=\tilde{\mu}\longrightarrow\mathbb{E}_f(\varphi(X))} \underbrace{\frac{n}{\sum_{i=1}^{n} w(X_i)}}_{\longrightarrow 1} \overset{a.s.}{\underset{n\to\infty}{\longrightarrow}} \mathbb{E}_f(\varphi(X)),$$

(provided $\mathrm{supp}(g) \supseteq \mathrm{supp}(f)$ and $\mathbb{E}_g|w(X)\cdot\varphi(X)| < \infty$).

Theorem: Bias and Variance (ctd.)

$$\begin{aligned}
\mathbb{E}_g(\hat{\mu}) &= \mu + \frac{\mu \mathrm{Var}_g(w(X)) - \mathbb{C}ov_g\left[w(X), w(X)\cdot\varphi(X)\right]}{n} + O(n^{-2}) \\
\mathrm{Var}_g(\hat{\mu}) &= \frac{\mathrm{Var}_g(w(X)\cdot\varphi(X)) - 2\mu\mathbb{C}ov_g\left[w(X), w(X)\cdot\varphi(X)\right]}{n} \\
&\quad + \frac{\mu^2\mathrm{Var}_g(w(X))}{n} + O(n^{-2})
\end{aligned}$$

Finite variance estimators

- Importance sampling estimates are consistent for many choices of g.
- More important in practice: we want *finite variance estimators*:

$$\mathrm{Var}(\tilde{\mu}) = \mathrm{Var}\left(\frac{\sum_{i=1}^{n} w(X_i)\varphi(X_i)}{n}\right) < \infty$$

- Sufficient (albeit restrictive) conditions for finite variance of $\tilde{\mu}$:
  - $f(x) \leq M \cdot g(x)$ and $\mathrm{Var}_f(\varphi(X)) < \infty$, or
  - E is compact, f is bounded above on E, and g is bounded below on E.
- Note: If f has heavier tails then g, then the weights may have *infinite* variance!

Summary of Part 2

- Transformation: Inversion sampling
- Transformation: Case-specific methods such as Box–Muller
- Rejection Sampling
- Importance Sampling

Part 3

Markov chain Monte Carlo

Part 3— Section 7

Motivation and Basics

Why do we need other, more complicated methods?

- Transformation's great when it works.
- Rejection sampling's good when M is small.
- Importance sampling works well with good proposals.
- What do we do when we can't meet any of these requirements?

One Approach

Markov Chain Monte Carlo methods (MCMC)

- Key idea: Create a *dependent* sample, i.e. $X^{(t)}$ depends on the previous value $X^{(t-1)}$.
  Allows for "local" updates.
- Yields an "approximate sample" from the target distribution.
- More mathematically speaking: yields a Markov chain with the target distribution f as stationary distribution.
- Under conditions, the realised chain provides approximations of $\mathbb{E}_f[\varphi(X)]$ and of f itself.

Markov Chains

Markov Chain (N.B. Terminology varies)

A *discrete time* Markov process taking values in a *general space*:

$$X^{(0)} \sim \mu_0 \qquad \text{Initial Dist.}$$

$$X^{(t)} | \left(X^{(0)} = x^{(0)}, \ldots, X^{(t-1)} = x^{(t-1)} \right) \sim K(x^{(t-1)}, \cdot) \quad \text{Kernel}$$

Stationary Distribution

f is a *stationary* or *invariant* distribution for a Markov Chain on E with kernel K if

$$\int_A \int_E f(x) K(x, y) dx dy = \int_A f(y) dy$$

for all measurable sets A [or $\int f(x) K(x, y) dx = f(y)$].

Heuristically Motivating MCMC

- If $X^{(0)}, \ldots$ is an f-invariant Markov chain and $X^{(t)} \sim f$ for some t then $X^{(t+s)} \sim f \quad \forall s \in \mathbb{N}$.
- So if $X^{(t)}$ is "approximately independent" of $X^{(t+s)}$ for large enough s then
  - $X^{(t)}, X^{(t+s)}, \ldots, X^{(t+ks)}, \ldots$ is approximately $\overset{\text{iid}}{\sim} f$,
  - $X^{(t+1)}, X^{(t+s+1)}, \ldots, X^{(t+ks+1)}, \ldots$ is approximately $\overset{\text{iid}}{\sim} f$,

    $\vdots$
  - $X^{(t+s-1)}, X^{(t+2s-1)}, \ldots, X^{(t+ks-1)}, \ldots$ is approximately $\overset{\text{iid}}{\sim} f$.
- We might conjecture that for such a chain, for some large s:

$$\frac{1}{n}\sum_{k=1}^{n} \varphi(X^{(t+ks)}) \to \mathbb{E}_f\left[\varphi(X)\right] \text{ and } \frac{1}{n}\sum_{k=1}^{n} \varphi(X^{(k)}) \to \mathbb{E}_f\left[\varphi(X)\right].$$

Some Questions to Answer

- Can we formalise this heuristic argument?

  $\rightsquigarrow$ ergodic theory
- How can we construct f-invariant Markov kernels?

  $\rightsquigarrow$ various types of sampler
- What properties of these kernels are important?

  $\rightsquigarrow$ more ergodic theory
- How do we initialise the chain?

  $\rightsquigarrow$ transient phases and burn-in
- How do we know if it's working?

  $\rightsquigarrow$ ergodic theory and convergence diagnostics

Aperiodicity

Definition: Period

A Markov chain has a period d if there exists some partition of the state space, $E_1, \dots, E_d$ with the properties that:

- $\forall i \neq j: \; E_i \cap E_j = \emptyset$,
- $\bigcup\limits_{i=1}^{d} E_i = E$,
- The chain moves deterministically between elements of the partition:

$$\forall i, j, t, s : \mathbb{P}\left(X_{t+s} \in E_j | X_t \in E_i\right) = \begin{cases} 1 & j = i + s \mod d \\ 0 & \text{otherwise.} \end{cases}$$

A Markov chain is *aperiodic* if its period is 1.

Irreducibility

Definition: Irreducibility

Given a distribution, f, over E, a Markov chain is said to be *f-irreducible* if for all points $x \in E$ and all measurable sets A such that $f(A) > 0$ there exists some t such that:

$$\int_A K^t(x,y)dy > 0.$$

If this condition holds with $t = 1$, then the chain is said to be *strongly f-irreducible*.

$$K^t(x,y) := \int K(x,z)K^{t-1}(z,y)dz, \quad K^1(x,y) = K(x,y).$$

Transience and Recurrence I

Consider sets $A \subseteq E$ for f-irreducible Markov chains.
Let $\eta_A := \sum_{k=1}^{\infty} \mathbb{I}_A(X^{(k)})$.

Transience and Recurrence of Sets

A set A is *recurrent* if:

$$\forall x \in A : \mathbb{E}_x [\eta_A] = \infty.$$

A set is *uniformly transient* if there exists some $M < \infty$ such that:

$$\forall x \in A : \quad \mathbb{E}_x [\eta_A] \leq M.$$

A set, $A \subseteq E$, is *transient* if it may be expressed as a countable union of uniformly transient sets.

Transience and Recurrence II

Transience and Recurrence of Markov Chains

A Markov chain is *recurrent* if the following hold:

- The chain is f-irreducible for some distribution f.
- For every measurable set $A \subseteq E$ such that $\int_A f(y)dy > 0$, $\mathbb{E}_x[\eta_A] = \infty$ for every $x \in A$.

It is *transient* if it is f-irreducible for some distribution f and the entire space is transient.

In the case of irreducible chains, transience and recurrence are properties of the chain rather than individual states.

A Motivating Convergence Result

Theorem (A Simple Ergodic Theorem)

If $(X_i)_{i\in\mathbb{N}}$ is an f-irreducible, f-invariant, recurrent $\mathbb{R}^d$-valued Markov chain, then the following strong law of large numbers holds for any integrable function $\varphi : \mathbb{R}^d \to \mathbb{R}$:

$$\lim_{t\to\infty} \frac{1}{t}\sum_{i=1}^{t} \varphi(X_i) \overset{a.s.}{=} \int \varphi(x) f(x) dx.$$

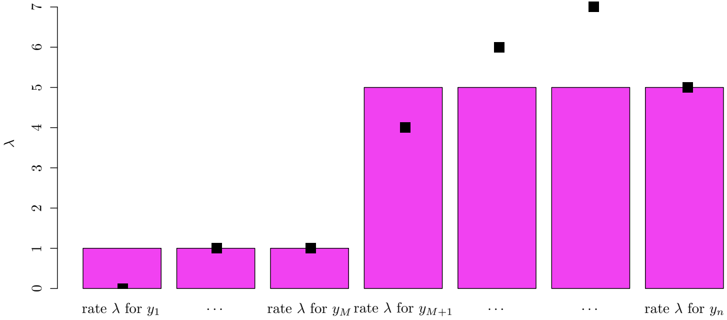
for almost every starting value x.

Note: this gives no *rate* of convergence.

Part 3— Section 8

The Gibbs Sampler

Example: Poisson change point model I



$$Y_i \sim \mathrm{Poi}(\lambda_1) \quad \text{for} \quad i = 1, \ldots, M,$$

$$Y_i \sim \mathrm{Poi}(\lambda_2) \quad \text{for} \quad i = M+1, \ldots, n.$$

Example: Poisson change point model II

Objective: (Bayesian) inference about the parameters λ_1, λ_2, and M given observed data $y_1, \ldots, y_n$.

- Prior distributions: $\lambda_j \sim \mathsf{Gamma}(\alpha_j, \beta_j)$ $(j = 1, 2)$, i.e.

$$f(\lambda_j) = \frac{1}{\Gamma(\alpha_j)} \lambda_j^{\alpha_j - 1} \beta_j^{\alpha_j} \exp(-\beta_j \lambda_j).$$

(discrete uniform prior on M, i.e. $p(M) \propto 1$).

- Likelihood: $L(\lambda_1, \lambda_2, M; y_1, \ldots, y_n)$

$$= \left(\prod_{i=1}^{M} \frac{\exp(-\lambda_1)\lambda_1^{y_i}}{y_i!} \right) \cdot \left(\prod_{i=M+1}^{n} \frac{\exp(-\lambda_2)\lambda_2^{y_i}}{y_i!} \right).$$

Example: Poisson change point model III

- Joint distribution $f(y_1, \ldots, y_n, \lambda_1, \lambda_2, M)$

$$\begin{aligned} &= L(\lambda_1, \lambda_2, M; y_1, \ldots, y_n) \cdot f(\lambda_1) \cdot f(\lambda_2) \cdot p(M) \\ &\propto \left(\prod_{i=1}^{M} \frac{\exp(-\lambda_1)\lambda_1^{y_i}}{y_i!}\right) \cdot \left(\prod_{i=M+1}^{n} \frac{\exp(-\lambda_2)\lambda_2^{y_i}}{y_i!}\right) \\ &\quad \cdot \frac{1}{\Gamma(\alpha_1)} \lambda_1^{\alpha_1 - 1} \beta_1^{\alpha_1} \exp(-\beta_1 \lambda_1) \cdot \frac{1}{\Gamma(\alpha_2)} \lambda_2^{\alpha_2 - 1} \beta_2^{\alpha_2} \exp(-\beta_2 \lambda_2) \end{aligned}$$

- Joint posterior distribution $f(\lambda_1, \lambda_2, M | y_1, \ldots, y_n)$

$$\begin{aligned} \propto \; & \lambda_1^{\alpha_1 - 1 + \sum_{i=1}^{M} y_i} \exp(-(\beta_1 + M)\lambda_1) \\ & \cdot \lambda_2^{\alpha_2 - 1 + \sum_{i=M+1}^{n} y_i} \exp(-(\beta_2 + n - M)\lambda_2) \end{aligned}$$

Example: Poisson change point model IV

- Conditional on M (i.e. if M was known) we have

$$f(\lambda_1|y_1,\ldots,y_n,M) \propto \lambda_1^{\alpha_1-1+\sum_{i=1}^M y_i}\exp(-(\beta_1+M)\lambda_1),$$

i.e.

$$\begin{aligned}\lambda_1|Y_1,\ldots Y_n,M &\sim \text{Gamma}\left(\alpha_1+\sum_{i=1}^{M}y_i,\beta_1+M\right),\\ \lambda_2|Y_1,\ldots Y_n,M &\sim \text{Gamma}\left(\alpha_2+\sum_{i=M+1}^{n}y_i,\beta_2+n-M\right).\end{aligned}$$

- $p(M|\ldots) \propto \lambda_1^{\sum_{i=1}^M y_i}\cdot\lambda_2^{\sum_{i=M+1}^n y_i}\cdot\exp((\lambda_2-\lambda_1)\cdot M).$

Example: Poisson change point model V

This suggests an iterative algorithm:

1. Draw λ_1 from $\lambda_1 | Y_1, \ldots, Y_n, M$, i.e. draw
$$\lambda_1 \sim \text{Gamma}\left(\alpha_1 + \sum_{i=1}^{M} y_i, \beta_1 + M\right).$$
2. Draw λ_2 from $\lambda_2 | Y_1, \ldots, Y_n, M$, i.e. draw
$$\lambda_2 \sim \text{Gamma}\left(\alpha_2 + \sum_{i=M+1}^{n} y_i, \beta_2 + n - M\right).$$
3. Draw M from $M | Y_1, \ldots, Y_n, \lambda_1, \lambda_2$, i.e. draw
$$p(M) \propto \lambda_1^{\sum_{i=1}^{M} y_i} \cdot \lambda_2^{\sum_{i=M+1}^{n} y_i} \cdot \exp((\lambda_2 - \lambda_1) \cdot M).$$

The systematic scan Gibbs sampler

Algorithm: (Systematic scan) Gibbs sampler

Starting with $(X_1^{(0)}, \dots, X_p^{(0)})$ iterate for $t = 1, 2, \dots$

1. Draw $X_1^{(t)} \sim f_{X_1|X_{-1}}(\cdot|X_2^{(t-1)}, \dots, X_p^{(t-1)})$.

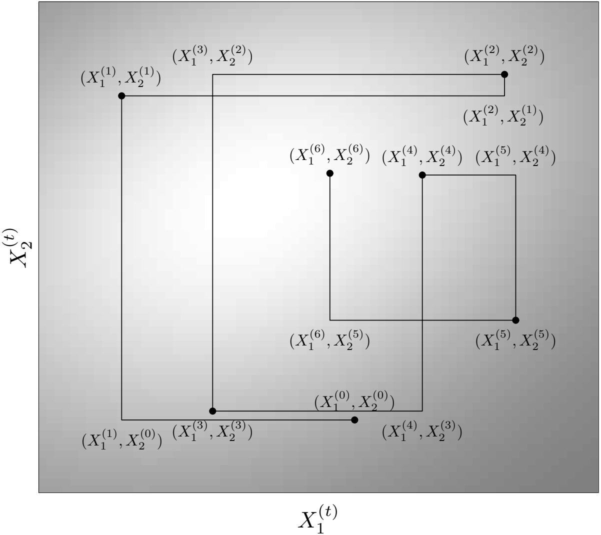
. . .

j. Draw $X_j^{(t)} \sim f_{X_j|X_{-j}}(\cdot|X_1^{(t)}, \dots, X_{j-1}^{(t)}, X_{j+1}^{(t-1)}, \dots, X_p^{(t-1)})$.

. . .

p. Draw $X_p^{(t)} \sim f_{X_p|X_{-p}}(\cdot|X_1^{(t)}, \dots, X_{p-1}^{(t)})$.

Illustration of the systematic scan Gibbs sampler



The random scan Gibbs sampler

Algorithm: (Random scan) Gibbs sampler

Starting with $(X_1^{(0)}, \dots, X_p^{(0)})$ iterate for $t = 1, 2, \dots$

1. Draw an index j from a distribution on $\{1, \dots, p\}$ (e.g. uniform).
2. Draw $X_j^{(t)} \sim f_{X_j|X_{-j}}(\cdot | X_1^{(t-1)}, \dots, X_{j-1}^{(t-1)}, X_{j+1}^{(t-1)}, \dots, X_p^{(t-1)})$, and set $X_\iota^{(t)} := X_\iota^{(t-1)}$ for all $\iota \neq j$.

Invariant distribution

Lemma (Kernel)

The transition kernel of the systematic scan Gibbs sampler is

$$\begin{aligned} K(\mathbf{x}^{(t-1)}, \mathbf{x}^{(t)}) \quad = \quad & f_{X_1|X_{-1}}(x_1^{(t)}|x_2^{(t-1)}, \dots, x_p^{(t-1)}) \\ & \cdot f_{X_2|X_{-2}}(x_2^{(t)}|x_1^{(t)}, x_3^{(t-1)}, \dots, x_p^{(t-1)}) \\ & \cdot \dots \\ & \cdot f_{X_p|X_{-p}}(x_p^{(t)}|x_1^{(t)}, \dots, x_{p-1}^{(t)}). \end{aligned}$$

Proposition (Invariance)

The joint distribution $f(x_1, \dots, x_p)$ is indeed the invariant distribution of the Markov chain $(\mathbf{X}^{(0)}, \mathbf{X}^{(1)}, \dots)$ generated by the Gibbs sampler.

Proof (outline) I

Assume that $\mathbf{X}^{(t-1)} \sim f$, then

$$\mathbb{P}(\mathbf{X}^{(t)} \in \mathcal{X}) = \int_{\mathcal{X}} \int f(\mathbf{x}^{(t-1)}) K(\mathbf{x}^{(t-1)}, \mathbf{x}^{(t)}) \, d\mathbf{x}^{(t-1)} \, d\mathbf{x}^{(t)}.$$

We can expand the $K(\mathbf{x}^{(t-1)}, \mathbf{x}^{(t)})$ of the integrand, and compute the $x_1^{(t-1)}$-integral:

$$\underbrace{\underbrace{\int f(x_1^{(t-1)}, \ldots, x_p^{(t-1)}) \, dx_1^{(t-1)}}_{=f(x_2^{(t-1)}, \ldots, x_p^{(t-1)})} f_{X_1|X_{-1}}(x_1^{(t)} | x_2^{(t-1)}, \ldots, x_p^{(t-1)}) \cdot}_{=f(x_1^{(t)}, x_2^{(t-1)}, \ldots, x_p^{(t-1)})}$$

$$f_{X_2|X_{-2}}(x_2^{(t)} | x_1^{(t)}, \ldots, x_p^{(t-1)}) \cdots f_{X_p|X_{-p}}(x_p^{(t)} | x_1^{(t)}, \ldots, x_{p-1}^{(t)}).$$

Proof (outline) II

And we can then compute the $x_2^{(t-1)}$ integral:

$$\underbrace{\int \underbrace{\int f(x_1^{(t)}, x_2^{(t-1)}, \dots, x_p^{(t-1)})\ dx_2^{(t-1)}}_{=f(x_1^{(t)}, x_3^{(t-1)}, \dots, x_p^{(t-1)})} f_{X_2|X_{-2}}(x_2^{(t)}|x_1^{(t)}, x_3^{(t-1)}, \dots, x_p^{(t-1)})}_{=f(x_1^{(t)}, x_2^{(t)}, x_3^{(t-1)}, \dots, x_p^{(t-1)})}$$

$$f_{X_3|X_{-3}}(x_3^{(t)}|x_1^{(t)}, \dots, x_p^{(t-1)}) \cdots f_{X_p|X_{-p}}(x_p^{(t)}|x_1^{(t)}, \dots, x_{p-1}^{(t)}).$$

And so on until the $x_p^{(t-1)}$-integral:

$$\underbrace{\underbrace{\int f(x_1^{(t)}, \dots, x_{p-1}^{(t)}, x_p^{(t-1)})\ dx_p^{(t-1)}}_{=f(x_1^{(t)}, \dots, x_{p-1}^{(t)})} f_{X_p|X_{-p}}(x_p^{(t)}|x_1^{(t)}, \dots, x_{p-1}^{(t)})}_{=f(x_1^{(t)}, \dots, x_p^{(t)})}.$$

Proof (outline) III

This just leaves the $\mathbf{x}^{(t)}$-integrals:

$$\mathbb{P}(\mathbf{X}^{(t)} \in \mathcal{X}) = \int_{\mathcal{X}} f(x_1^{(t)}, \ldots, x_p^{(t)})\, d\mathbf{x}^{(t)}.$$

Thus f is the density of $\mathbf{X}^{(t)}$ (if $\mathbf{X}^{(t-1)} \sim f$). □

Recall our Poisson Changepoint Model

- Joint posterior distribution $f(\lambda_1, \lambda_2, M | y_1, \ldots, y_n)$

$$\begin{aligned} \propto \quad & \lambda_1^{\alpha_1 - 1 + \sum_{i=1}^{M} y_i} \exp(-(\beta_1 + M)\lambda_1) \\ & \cdot \lambda_2^{\alpha_2 - 1 + \sum_{i=M+1}^{n} y_i} \exp(-(\beta_2 + n - M)\lambda_2) \end{aligned}$$

- Full Posterior Distributions

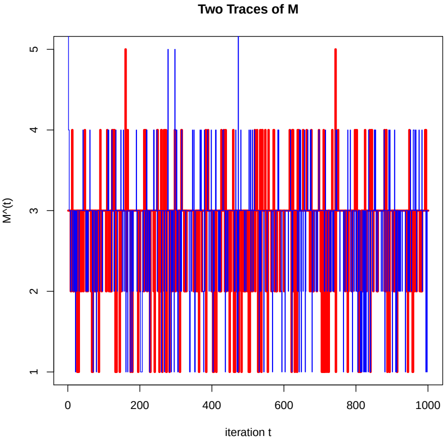
$$\begin{aligned} \lambda_1 | Y_1, \ldots Y_n, M \quad & \sim \quad \text{Gamma}\left(\alpha_1 + \sum_{i=1}^{M} y_i, \beta_1 + M\right), \\ \lambda_2 | Y_1, \ldots Y_n, M \quad & \sim \quad \text{Gamma}\left(\alpha_2 + \sum_{i=M+1}^{n} y_i, \beta_2 + n - M\right). \end{aligned}$$

- and $p(M | \ldots) \propto \lambda_1^{\sum_{i=1}^{M} y_i} \cdot \lambda_2^{\sum_{i=M+1}^{n} y_i} \cdot \exp((\lambda_2 - \lambda_1) \cdot M)$.

An R Implementation

```
cdist.M <- function(lambda1,lambda2) {
    dist.M.log <- cumsum(y[1:n-1]) * log(lambda1) +
        (sum(y)-cumsum(y[1:n-1]))*log(lambda2) +
        (lambda2-lambda1) * (1:(n-1))
    dist.M <- exp(dist.M.log - mean(dist.M.log))
    dist.M <- dist.M / sum(dist.M)
}

pmix.gibbs <- function(M,lambda1,lambda2,t) {
 r <- array(NA,c(t+1,3))
 r[1,] <- c(M,lambda1,lambda2)
 for (i in 1:t) {
  #lambda1
  r[i+1,2] <- rgamma(1,a1+sum(y[1:r[i,1]]), b1+r[i,1])
  #lambda2
  r[i+1,3] <- rgamma(1,a2+sum(y[(r[i,1]+1):n]), b2+n-r[i,1])
  #M
  r[i+1,1] <- sample.int(n-1,1,prob=cdist.M(r[i+1,2],r[i+1,3]))
 }
 r
}
```

Traces and Estimates: M



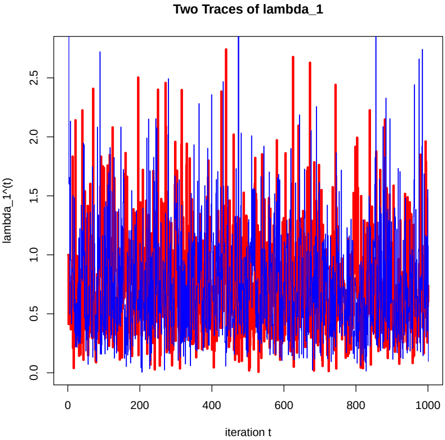
Consider two differently-initialised chains.

Chain 1:
$(M, \lambda_1, \lambda_2)^{(0)} = (3, 1, 2)$

Chain 2:
$(M, \lambda_1, \lambda_2)^{(0)} = (6, 4, \frac{1}{2})$

Estimated Posterior *Modes*:
Chain 1: 3
Chain 2: 3

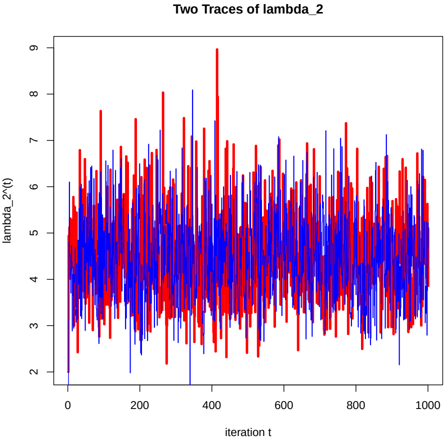
Traces and Estimates: λ_1



Estimated Posterior *Means*:

Chain 1: 0.76
Chain 2: 0.78

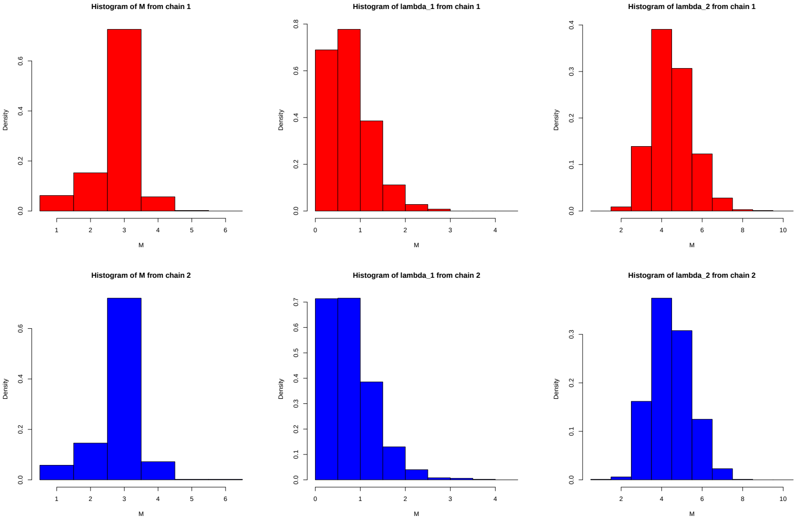
Traces and Estimates: λ_2



Estimated Posterior *Means*:

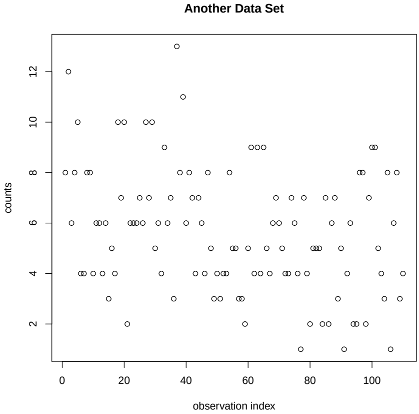
Chain 1: 4.51
Chain 2: 4.47

Histograms: Approximations of the Posterior



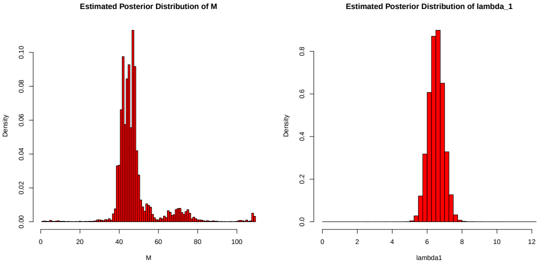
Poisson Change-Point Model: More Challenging Data I

Consider the more realistic data:



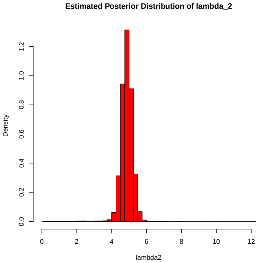
Poisson Change-Point Model: More Challenging Data II

From a chain of length 100,000 we obtain the following



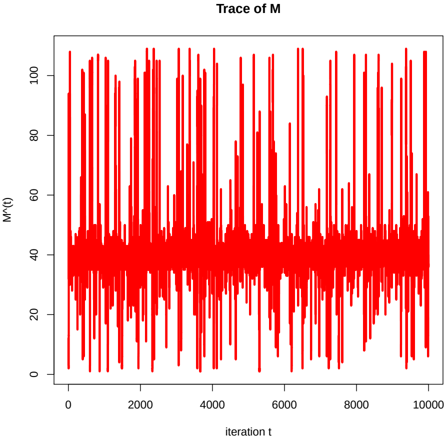
histograms:

Poisson Change-Point Model: More Challenging Data III

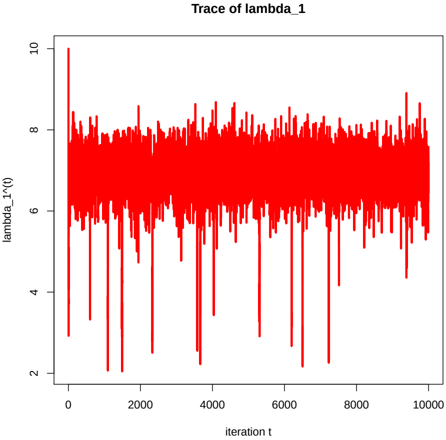


Data was generated with: `y <- c(rpois(40,7),rpois(70,5))`

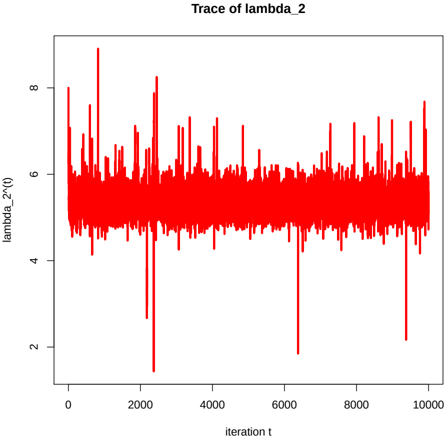
Poisson Change-Point Model: More Challenging Data IV



Poisson Change-Point Model: More Challenging Data V

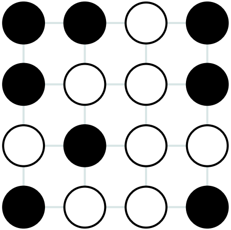


Poisson Change-Point Model: More Challenging Data VI



Example: The Ising Model

The Ising model on $(\mathcal{V},\mathcal{E})$ each $v_i \in \mathcal{V}$ has an associated $x_i \in \{-1,+1\}$:



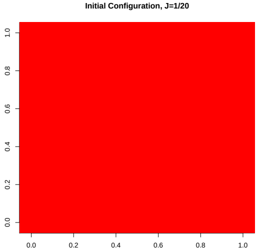
$$\begin{aligned}
&\pi(x_1,\ldots,x_m)\\
&=\frac{1}{Z}\exp\left(J\sum_{(i,j)\in\mathcal{E}} x_i\cdot x_j\right)\\
&=\frac{1}{Z}\exp(-J|\mathcal{E}|)\exp\left(2J\sum_{(i,j)\in\mathcal{E}}\mathbb{I}(x_i=x_j)\right)\\
&=\frac{1}{Z'}\exp\left(2J\sum_{(i,j)\in\mathcal{E}}\mathbb{I}(x_i=x_j)\right).
\end{aligned}$$

$$\pi(x_j|x_{-j})=\exp\left(J\sum_{i:(i,j)\in\mathcal{E}}x_ix_j\right)\Big/\left[\exp\left(-J\sum_{i:(i,j)\in\mathcal{E}}x_i\right)+\exp\left(J\sum_{i:(i,j)\in\mathcal{E}}x_i\right)\right].$$

The Core Logic in R

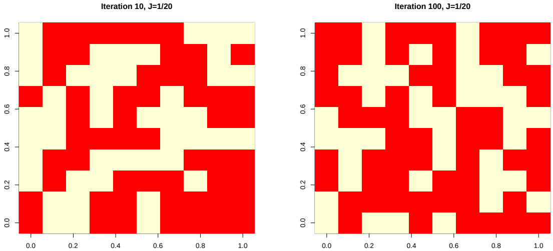
```
tr <- list()
tr[[1]] <- x <- array(0,c(m,n))

for (t in 1:100) {
    for(i in 1:m) {
        for(j in 1:n) {
            ns <- neighbours(m,n,i,j)
            p1 <- 0
            for(k in 1:length(ns)) {
                p1 <- p1 + x[(ns[[k]])[1],(ns[[k]])[2]]
            }
            p0 <- length(ns) - p1
            pp <- c(exp(J*p0),exp(J*p1))
            pp <- pp / sum(pp)
            x[i,j] <- sample(c(0,1),1,prob=pp)
        }
    }
    tr[[t+1]] <- x
}
```

The Gibbs Sampler for Ising Models I



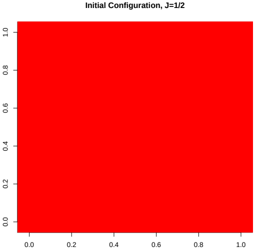
Samples 1, 10, and 100 with $J = 0.05$:

The Gibbs Sampler for Ising Models II

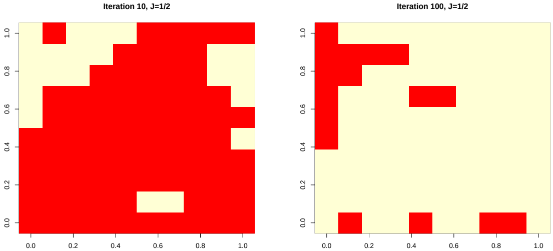


The Gibbs Sampler for Ising Models III

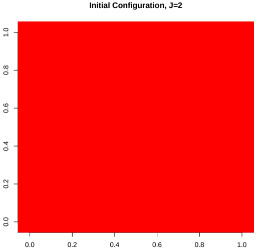
Samples 1, 10, and 100 with $J = 0.50$:



The Gibbs Sampler for Ising Models IV

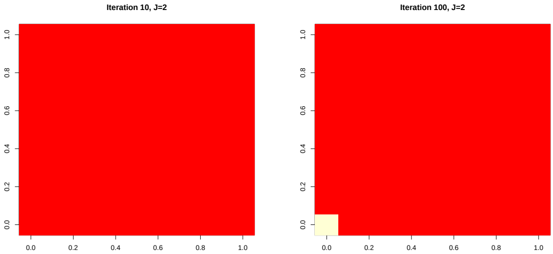


The Gibbs Sampler for Ising Models V



Samples 1, 10, and 100 with $J = 1.00$:

The Gibbs Sampler for Ising Models VI



Solutions include the *Swendsen-Wang* algorithm (c.f. assessment) or *perfect simulation*. . .

The Ising Model and Image Reconstruction

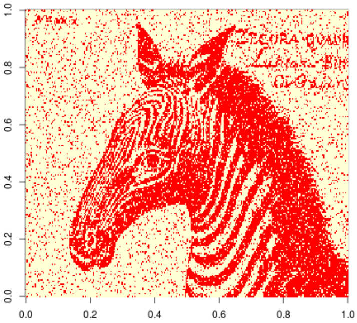
The Ising Model is widely used in statistics as a prior distribution.

- Consider image denoising: x an $m \times n$ image on $\mathcal{V} \subseteq \mathbb{Z}^2$ with obvious neighbourhood structure $\mathcal{E}$:
- Observe y where $y_v = x_v$ wp $1 - \epsilon$.
- Prior: $X \sim \text{Ising}(J, \mathcal{V}, \mathcal{E})$.
- Likelihood: $L(x; y) = \prod_{v \in \mathcal{V}} [(1-\epsilon)\mathbb{I}\{y_v = x_v\} + \epsilon \mathbb{I}\{y_v \neq x_v\}]$.
- Posterior:

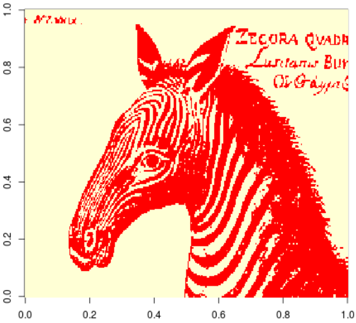
$$p(x|y) \propto \exp\left(2J \sum_{(i,j)\in\mathcal{E}} \mathbb{I}(x_i = x_j)\right) \cdot \prod_{v \in \mathcal{V}} [(1-\epsilon)\mathbb{I}\{y_v = x_v\} + \epsilon \mathbb{I}\{y_v \neq x_v\}]$$

Ludolphus' Zebra

`https://upload.wikimedia.org/wikipedia/commons/a/af/ZebraLudolphus.jpg`



Noisy Image / Samples

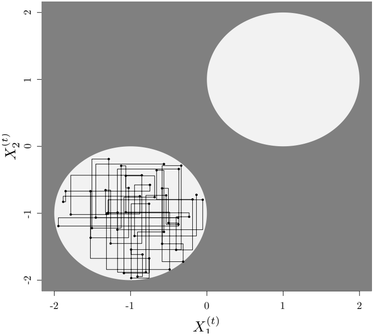


Ground Truth

A Pathological Example: The Reducible Gibbs sampler

Consider Gibbs sampling from the uniform distribution

$$f(x_1, x_2) = \frac{1}{2\pi} \mathbb{I}_{C_1 \cup C_2}(x_1, x_2),$$



$$\begin{aligned} C_1 &:= \{(x_1, x_2) : \|(x_1, x_2) - (1, 1)\| \leq 1\} \\ C_2 &:= \{(x_1, x_2) : \|(x_1, x_2) + (1, 1)\| \leq 1\} \end{aligned}$$

The resulting Markov chain is *reducible*:

It stays forever in either C_1 or C_2.

Part 3— Section 9

The Metropolis–Hastings Algorithm

The Metropolis–Hastings algorithm

Algorithm: Metropolis–Hastings

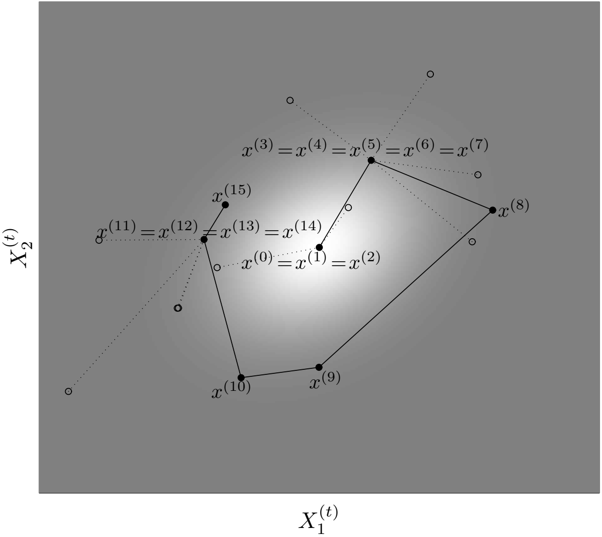
Starting with $\mathbf{X}^{(0)} := (X_1^{(0)}, \dots, X_p^{(0)})$ iterate for $t = 1, 2, \dots$

1. Draw $\mathbf{X} \sim q(\cdot|\mathbf{X}^{(t-1)})$.
2. Compute

$$\alpha(\mathbf{X}|\mathbf{X}^{(t-1)}) = \min\left\{1, \frac{f(\mathbf{X}) \cdot q(\mathbf{X}^{(t-1)}|\mathbf{X})}{f(\mathbf{X}^{(t-1)}) \cdot q(\mathbf{X}|\mathbf{X}^{(t-1)})}\right\}.$$

3. With probability $\alpha(\mathbf{X}|\mathbf{X}^{(t-1)})$ set $\mathbf{X}^{(t)} = \mathbf{X}$, otherwise set $\mathbf{X}^{(t)} = \mathbf{X}^{(t-1)}$.

Illustration of the Metropolis–Hastings method


$x^{(3)} = x^{(4)} = x^{(5)} = x^{(6)} = x^{(7)}$
$x^{(15)}$
$x^{(8)}$
$x^{(11)} = x^{(12)} = x^{(13)} = x^{(14)}$
$x^{(0)} = x^{(1)} = x^{(2)}$
$x^{(10)}$
$x^{(9)}$
$X_2^{(t)}$
$X_1^{(t)}$

Basic properties of the Metropolis–Hastings algorithm

- The probability that a newly proposed value is accepted given $\mathbf{X}^{(t-1)} = \mathbf{x}^{(t-1)}$ is

$$a(\mathbf{x}^{(t-1)}) = \int \alpha(\mathbf{x}|\mathbf{x}^{(t-1)}) q(\mathbf{x}|\mathbf{x}^{(t-1)})\, d\mathbf{x}.$$

- The probability of remaining in state $\mathbf{X}^{(t-1)}$ is

$$\mathbb{P}(\mathbf{X}^{(t)} = \mathbf{X}^{(t-1)}|\mathbf{X}^{(t-1)} = \mathbf{x}^{(t-1)}) = 1 - a(\mathbf{x}^{(t-1)}).$$

- The probability of acceptance does not depend on the normalisation constant: If $f(\mathbf{x}) = C \cdot \tilde{f}(\mathbf{x})$, then

$$\alpha(\mathbf{X}|\mathbf{X}^{(t-1)}) = \min\left(1, \frac{\tilde{f}(\mathbf{X}) \cdot q(\mathbf{X}^{(t-1)}|\mathbf{X})}{\tilde{f}(\mathbf{X}^{(t-1)}) \cdot q(\mathbf{X}|\mathbf{X}^{(t-1)})}\right)$$

Transition Kernel

Lemma (Transition Kernel of Metropolis–Hastings)

The transition kernel of the Metropolis–Hastings algorithm is

$$K(\mathbf{x}^{(t-1)},\mathbf{x}^{(t)}) = \alpha(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)})q(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}) \\ + (1 - a(\mathbf{x}^{(t-1)}))\delta_{x^{(t-1)}}(\mathbf{x}^{(t)}),$$

Lemma (Detailed Balance and Metropolis Hastings)

The Metropolis–Hastings kernel satisfies the detailed balance condition

$$K(\mathbf{x}^{(t-1)},\mathbf{x}^{(t)})f(\mathbf{x}^{(t-1)}) = K(\mathbf{x}^{(t)},\mathbf{x}^{(t-1)})f(\mathbf{x}^{(t)}).$$

f-invariance of Metropolis–Hastings

Proposition (Detailed Balanced implies Invariance)

Any K which satisfies the detailed balance condition with respect to f,

$$K(\mathbf{x}^{(t-1)}, \mathbf{x}^{(t)}) f(\mathbf{x}^{(t-1)}) = K(\mathbf{x}^{(t)}, \mathbf{x}^{(t-1)}) f(\mathbf{x}^{(t)}),$$

is f-invariant.

Proof

Integrate both sides wrt $\mathbf{x}^{(t-1)}$.

Hence the Metropolis–Hastings algorithm is f-invariant.

Random-walk Metropolis: Idea

- In the Metropolis–Hastings algorithm the proposal is from $\mathbf{X} \sim q(\cdot|\mathbf{X}^{(t-1)})$.
- A popular choice for the proposal is $q(\mathbf{x}|\mathbf{x}^{(t-1)}) = g(\mathbf{x} - \mathbf{x}^{(t-1)})$ with g *symmetric*, thus

$$\mathbf{X} = \mathbf{X}^{(t-1)} + \boldsymbol{\varepsilon}, \qquad \boldsymbol{\varepsilon} \sim g.$$

- Probability of acceptance becomes

$$\min\left\{1, \frac{f(\mathbf{X}) \cdot g(\mathbf{X} - \mathbf{X}^{(t-1)})}{f(\mathbf{X}^{(t-1)}) \cdot g(\mathbf{X}^{(t-1)} - \mathbf{X})}\right\} = \min\left\{1, \frac{f(\mathbf{X})}{f(\mathbf{X}^{(t-1)})}\right\}.$$

- We accept . . .
  - every move to a more probable state with probability 1.
  - moves to less probable states with a probability $f(\mathbf{X})/f(\mathbf{x}^{(t-1)}) < 1$.

Random-walk Metropolis: Algorithm

Random-Walk Metropolis

Starting with $\mathbf{X}^{(0)} := (X_1^{(0)}, \ldots, X_p^{(0)})$ and using a symmetric random walk proposal g, iterate for $t = 1, 2, \ldots$

1. Draw $\varepsilon \sim g$ and set $\mathbf{X} = \mathbf{X}^{(t-1)} + \varepsilon$.
2. Compute

$$\alpha(\mathbf{X}|\mathbf{X}^{(t-1)}) = \min\left\{1, \frac{f(\mathbf{X})}{f(\mathbf{X}^{(t-1)})}\right\}.$$

3. With probability $\alpha(\mathbf{X}|\mathbf{X}^{(t-1)})$ set $\mathbf{X}^{(t)} = \mathbf{X}$, otherwise set $\mathbf{X}^{(t)} = \mathbf{X}^{(t-1)}$.

Popular choices for g are (multivariate) Gaussians or t-distributions (the latter having heavier tails)

Example 3.4: Bayesian probit model (1)

- Medical study on infections resulting from birth by Cæsarean section.
- 3 influence factors:
  - indicator whether the Cæsarian was planned or not (z_{i1}),
  - indicator of whether additional risk factors were present at the time of birth (z_{i2}), and
  - indicator of whether antibiotics were given as a prophylaxis (z_{i3}).
- Response variable: number of infections Y_i that were observed amongst n_i patients having the same covariates.

# births		planned	risk factors	antibiotics
infection	total			
y_i	n_i	z_{i1}	z_{i2}	z_{i3}
11	98	1	1	1
1	18	0	1	1
0	2	0	0	1
23	26	1	1	0
28	58	0	1	0
0	9	1	0	0
8	40	0	0	0

Example 3.4: Bayesian probit model (2)

- Model for Y_i:

$$Y_i \sim \mathrm{Bin}(n_i, \pi_i), \qquad \pi = \Phi(\mathbf{z}_i' \boldsymbol{\beta}),$$

where $\mathbf{z}_i = (1, z_{i1}, z_{i2}, z_{i3})$ and $\Phi(\cdot)$ being the CDF of a $\mathrm{N}(0, 1)$.

- Prior on the parameter of interest $\boldsymbol{\beta}$: $\boldsymbol{\beta} \sim N(\mathbf{0}, \mathbb{I}/\lambda)$.
- The posterior density of $\boldsymbol{\beta}$ is

$$\begin{aligned} f(\boldsymbol{\beta}|y_1, \ldots, y_n) \;\; \propto \;\; & \left(\prod_{i=1}^{N} \Phi(\mathbf{z}_i' \boldsymbol{\beta})^{y_i} \cdot (1 - \Phi(\mathbf{z}_i' \boldsymbol{\beta}))^{n_i - y_i} \right) \\ & \cdot \exp\left(-\frac{\lambda}{2} \sum_{j=0}^{3} \beta_j^2 \right) \end{aligned}$$

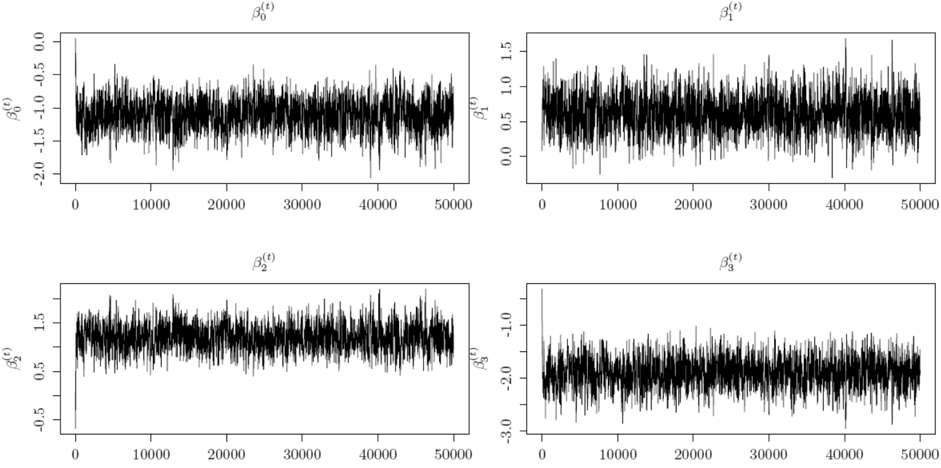
Example 3.4: Bayesian probit model (3)

Use the following “random walk Metropolis” algorithm.
Starting with any $\boldsymbol{\beta}^{(0)}$ iterate for $t = 1, 2, \ldots$:

1. Draw $\boldsymbol{\varepsilon} \sim N(\mathbf{0}, \boldsymbol{\Sigma})$ and set $\boldsymbol{\beta} = \boldsymbol{\beta}^{(t-1)} + \boldsymbol{\varepsilon}$.
2. Compute
$$\alpha(\boldsymbol{\beta}|\boldsymbol{\beta}^{(t-1)}) = \min\left\{1, \frac{f(\boldsymbol{\beta}|Y_1, \ldots, Y_n)}{f(\boldsymbol{\beta}^{(t-1)}|Y_1, \ldots, Y_n)}\right\}.$$
3. With probability $\alpha(\boldsymbol{\beta}|\boldsymbol{\beta}^{(t-1)})$ set $\boldsymbol{\beta}^{(t)} = \boldsymbol{\beta}$, otherwise set $\boldsymbol{\beta}^{(t)} = \boldsymbol{\beta}^{(t-1)}$.

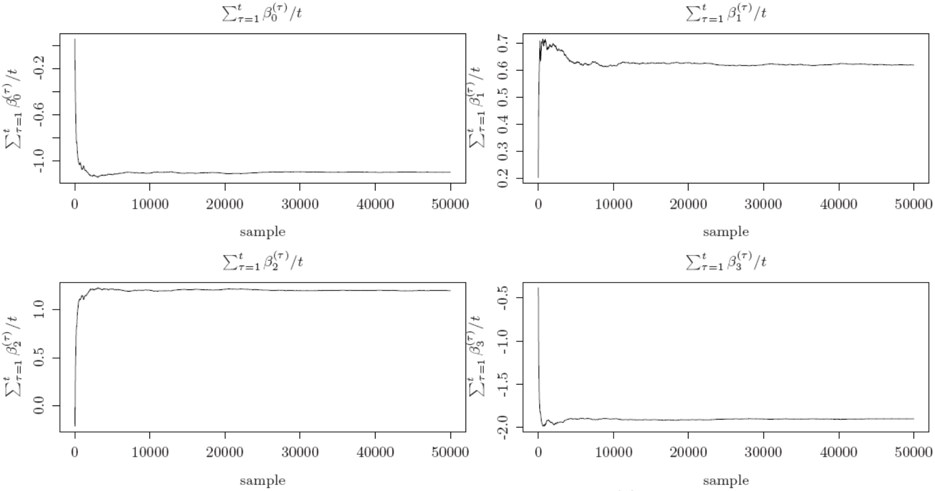
(for the moment we use $\boldsymbol{\Sigma} = 0.08 \cdot \mathbb{I}$, and $\lambda = 10$).

Example 3.4: Bayesian probit model (4)



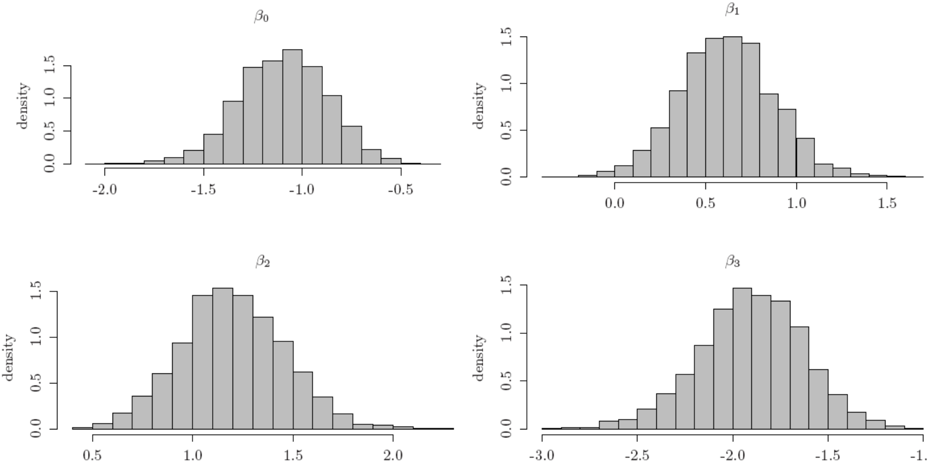
Convergence of the $\beta_j^{(t)}$ is to a distribution, not a value!

Example 3.4: Bayesian probit model (5)



Convergence of cumulative averages $\sum_{\tau=1}^t \beta_j^{(\tau)}/t$ is to a value.

Example 3.4: Bayesian probit model (6)



Example 3.4: Bayesian probit model (7)

		Posterior mean	95% credible interval	
intercept	β_0	-1.0952	-1.4646	-0.7333
planned	β_1	0.6201	0.2029	1.0413
risk factors	β_2	1.2000	0.7783	1.6296
antibiotics	β_3	-1.8993	-2.3636	-1.471

Choosing a good proposal distribution

- Ideally: Markov chain with small correlations $\rho(\mathbf{X}^{(t-1)}, \mathbf{X}^{(t)})$. Yields fast exploration of the support of the target f.
- Two sources for this correlation:
  - correlation between current state $\mathbf{X}^{(t-1)}$ and newly proposed value $\mathbf{X} \sim q(\cdot|\mathbf{X}^{(t-1)})$
    (can be reduced using a proposal with high variance),
  - correlation introduced by retaining a value $\mathbf{X}^{(t)} = \mathbf{X}^{(t-1)}$ because the proposal $\mathbf{X}$ has been rejected
    (can be reduced using a proposal with small variance).
- Trade-off for finding compromise between:
  - fast exploration of the space (good mixing behaviour),
  - obtaining a large probability of acceptance.
- For multivariate distributions: covariance of proposal should reflect the covariance structure of the target.

Example: Choice of proposal (1)

- Target distribution: $\mathrm{N}(0,1)$ (i.e. $f(\cdot) = \phi_{(0,1)}(\cdot)$).
- We want to use a random walk Metropolis algorithm with

$$\varepsilon \sim \mathrm{N}(0, \sigma^2).$$

- What is the optimal choice of σ^2?
- We consider four choices $\sigma^2 = 0.01, 1, 5, 100$.

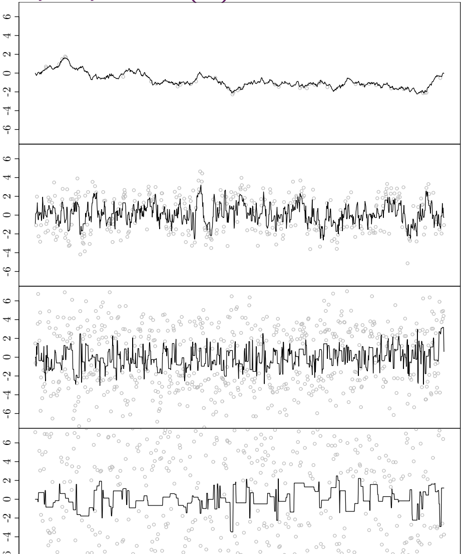
Example 5.3: Choice of proposal (2)

$\sigma^2 = 0.01$

$\sigma^2 = 1$

$\sigma^2 = 5$

$\sigma^2 = 100$

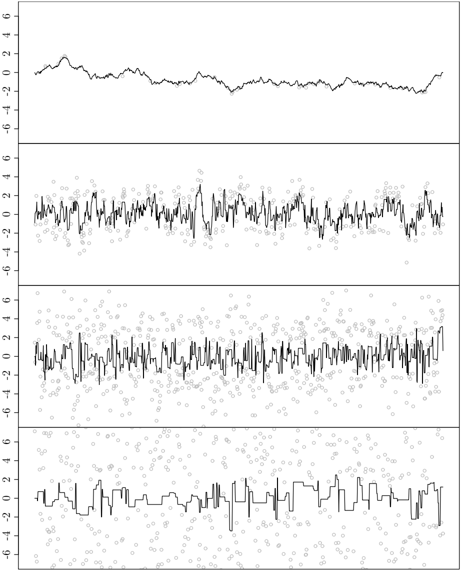


$\sigma^2 = 0.01$

$\sigma^2 = 1$

$\sigma^2 = 5$

$\sigma^2 = 100$



Which proposal looks best?

Vevox.app 170–356–838

Example 5.3: Choice of proposal (4)

	Autocorrelation $\rho(X^{(t-1)}, X^{(t)})$		Probability of acceptance $\alpha(X, X^{(t-1)})$	
	Mean	95% CI	Mean	95% CI
$\sigma^2 = 0.1^2$	0.9901	(0.9891,0.9910)	0.9694	(0.9677,0.9710)
$\sigma^2 = 1$	0.7733	(0.7676,0.7791)	0.7038	(0.7014,0.7061)
$\sigma^2 = 2.38^2$	0.6225	(0.6162,0.6289)	0.4426	(0.4401,0.4452)
$\sigma^2 = 10^2$	0.8360	(0.8303,0.8418)	0.1255	(0.1237,0.1274)

Suggests: Optimal choice is $\sigma^2 = 2.38^2 = 5.66 > 1$.

Example 5.4: Bayesian probit model (revisited)

- So far we used: $\mathrm{Var}(\boldsymbol{\varepsilon}) = 0.08 \cdot \mathbb{I}$.
- Better choice: Let $\mathrm{Var}(\boldsymbol{\varepsilon})$ reflect the covariance structure
- Frequentist asymptotic theory: $\mathrm{Var}(\hat{\boldsymbol{\beta}}^{\text{m.l.e}}) = (\mathbf{Z}'\mathbf{D}\mathbf{Z})^{-1}$, $\mathbf{D}$ is a suitable diagonal matrix.
- Better choice: $\mathrm{Var}(\boldsymbol{\varepsilon}) = 2 \cdot (\mathbf{Z}'\mathbf{D}\mathbf{Z})^{-1}$.
- Increases rate of acceptance from 13.9% to 20.0% and reduces autocorrelation:

$\boldsymbol{\Sigma} = 0.08 \cdot \mathbf{I}$	β_0	β_1	β_2	β_3
Autocorrelation $\rho(\beta_j^{(t-1)}, \beta_j^{(t)})$	0.9496	0.9503	0.9562	0.9532
$\boldsymbol{\Sigma} = 2 \cdot (\mathbf{Z}'\mathbf{D}\mathbf{Z})^{-1}$	β_0	β_1	β_2	β_3
Autocorrelation $\rho(\beta_j^{(t-1)}, \beta_j^{(t)})$	0.8726	0.8765	0.8741	0.8792

(In this example $\det(0.08 \cdot \mathbb{I}) = \det(2 \cdot (\mathbf{Z}'\mathbf{D}\mathbf{Z})^{-1})$.)

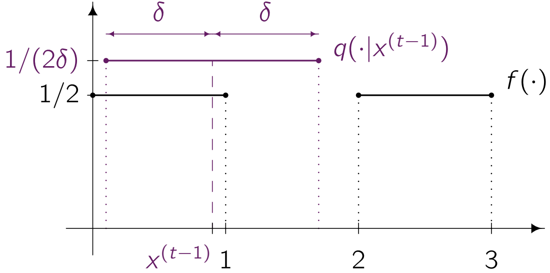
Pathological Example: Reducible Metropolis–Hastings

Consider the target distribution

$$f(x) = (\mathbb{I}_{[0,1]}(x) + \mathbb{I}_{[2,3]}(x))/2.$$

and the proposal distribution $q(\cdot|\mathbf{x}^{(t-1)})$:

$$X|X^{(t-1)} = x^{(t-1)} \sim \mathrm{U}[x^{(t-1)} - \delta, x^{(t-1)} + \delta]$$



Reducible if $\delta \leq 1$: the chain stays either in $[0, 1]$ or $[2, 3]$.

The Metropolised Independence Sampler

Independent proposals: choose $q(\cdot|x) = q(\cdot)$.

Algorithm 5.3 The Independence Sampler

Starting with $\mathbf{X}^{(0)} := (X_1^{(0)}, \dots, X_p^{(0)})$ iterate for $t = 1, 2, \dots$

1. Draw $\mathbf{X} \sim q(\cdot)$.
2. Compute

$$\alpha(\mathbf{X}|\mathbf{X}^{(t-1)}) = \min\left\{1, \frac{f(\mathbf{X}) \cdot q(\mathbf{X}^{(t-1)})}{f(\mathbf{X}^{(t-1)}) \cdot q(\mathbf{X})}\right\}.$$

3. With probability $\alpha(\mathbf{X}|\mathbf{X}^{(t-1)})$ set $\mathbf{X}^{(t)} = \mathbf{X}$, otherwise set $\mathbf{X}^{(t)} = \mathbf{X}^{(t-1)}$.

Acceptance Rate

Proposition (Acceptance Rate of Independence Sampler)

If $f(\mathbf{x})/q(\mathbf{x}) \leq M < \infty$ the acceptance rate of the independence sampler is at least as high as that of the corresponding rejection sampler.

Gibbs Samplers Revisited

What about full conditionals as MH proposals?

- For $\mathbf{X} = (X_1, \ldots, X_p)$:
- Consider $q(\mathbf{X}|\mathbf{x}^{(t-1)}) = \delta_{x_{-p}^{(t-1)}}(X_{-p}) f_{X_p|X_{-p}}(X_p|X_{-p})$.

Remark

A Gibbs sampler step is a special case of the Metropolis–Hastings algorithm.

Part 3— Section 10

Simulated Annealing

Finding the mode of a distribution

- Our objective so far: estimate $\mathbb{E}(h(\mathbf{X}))$.
- A new objective: estimate (global) mode(s) of a distribution:

$$\{\boldsymbol{\xi} : \ f(\boldsymbol{\xi}) \geq f(\mathbf{x}) \ \forall \mathbf{x}\}$$

- Naïvely: Choose the $\mathbf{X}^{(t)}$ with maximal density $f(\mathbf{X}^{(t)})$.

Example: Naïvely Finding The Mode of a Normal Density

- Consider $f(\mathbf{x}) = \phi(\mathbf{x})$
- Use a Random Walk proposal $\mathbf{X} \sim \mathrm{N}(\mathbf{X}^{(t-1)}, \sigma^2)$ with $\sigma^2 = 0.1^2, 1, 2.38^2, 10^2$.
- Run chains for various T, and pick for each: $\mathbf{X}^{\max} = \arg\max_{X \in (X^{(t)})_{t=1}^{T}} f(\mathbf{X})$

$N \vert \sigma^2$	0.1^2	1.0^2	2.38^2	10^2
10	0.906	0.091	0.609	0.623
100	0.315	0.020	-0.063	-0.033
100b	-0.033	0.007	0.065	0.005
1000	0.001	0.001	-0.002	-0.002
1000b	0.015	0.001	-0.001	-0.001

- This approach seems to work here. . .

More Efficiently Finding the Mode

- Idea: Transform distribution such that it is more concentrated around the mode(s).
- Consider
  $$f_{(\beta)}(x) \propto (f(x))^{\beta}$$
  for very large values of β.
- For $\beta \to \infty$ the distribution $f_{(\beta)}(\cdot)$ will be concentrated on the (global) modes.

Example: Normal distribution (1)

- Consider the $\mathsf{N}(\mu, \sigma^2)$ distribution with density

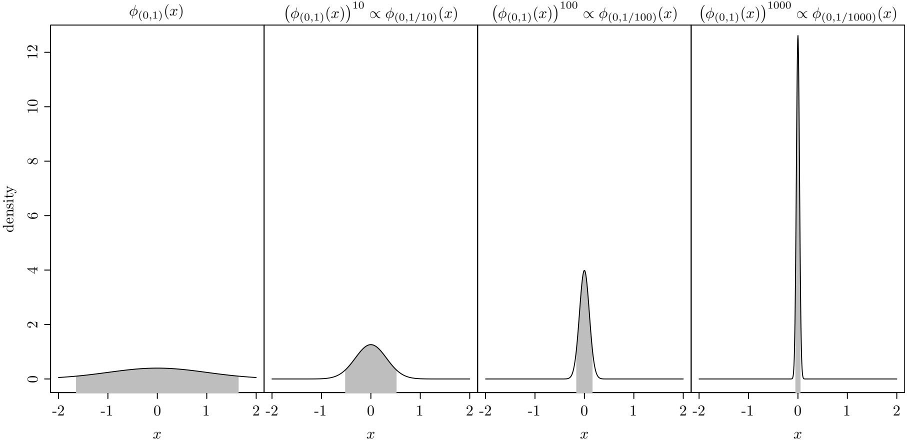
$$f_{(\mu,\sigma^2)}(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \propto \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right).$$

- Mode of the $\mathsf{N}(\mu, \sigma^2)$ distribution is μ.
- For increasing β the distribution is more and more concentrated around its mode μ, as

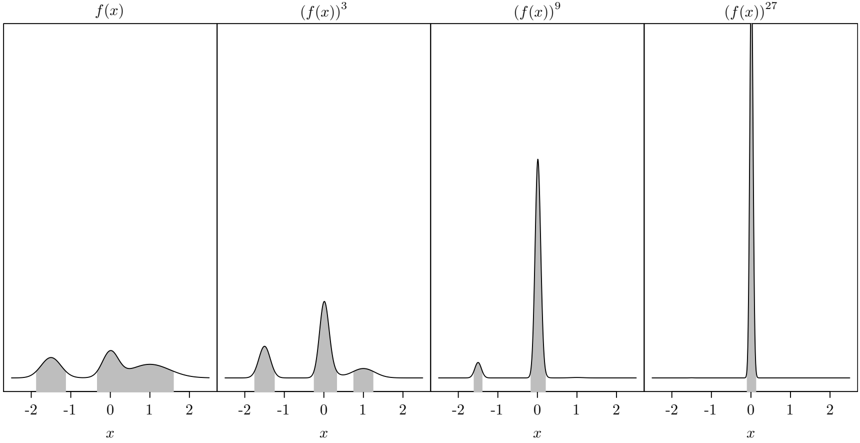
$$\begin{aligned} \left(f_{(\mu,\sigma^2)}(x)\right)^\beta &\propto \left(\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)\right)^\beta \\ &= \exp\left(-\frac{(x-\mu)^2}{2\sigma^2/\beta}\right) \propto f_{(\mu,\sigma^2/\beta)}(x). \end{aligned}$$

- Increasing β corresponds to reducing the variance.

Example: Normal distribution (2)



Another example



Sampling from $f_{(\beta)}(\cdot)$

- We can sample from $f_{(\beta)}(\cdot)$ using a random walk Metropolis algorithm.
- Probability of acceptance becomes

$$\min\left\{1, \frac{f_{(\beta)}(\mathbf{X})}{f_{(\beta)}(\mathbf{X}^{(t-1)})}\right\} = \min\left\{1, \left(\frac{f(\mathbf{X})}{f(\mathbf{X}^{(t-1)})}\right)^{\beta}\right\}.$$

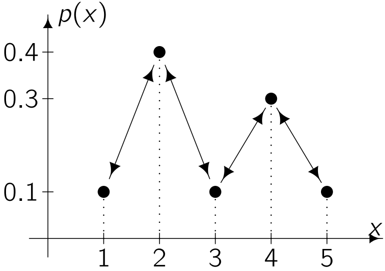
- For $\beta \to \infty$ the probability of acceptance converges to...
  - 1 if $f(\mathbf{X}) \geq f(\mathbf{X}^{(t-1)})$, and
  - 0 if $f(\mathbf{X}) < f(\mathbf{X}^{(t-1)})$.
- For large β the chain $(\mathbf{X}^{(t)})_t$ converges to a local maximum of $f(\cdot)$.
- Whether the chain can escape from local maxima of the density depends on whether it can reach the (global) mode within a single step.

Another Example

Assume we want to find the mode of

$$p(x) = \begin{cases} 0.4 & \text{for } x = 2 \\ 0.3 & \text{for } x = 4 \\ 0.1 & \text{for } x = 1, 3, 5. \end{cases}$$

using a random walk Metropolis algorithm that can only move one to the left or one to the right.



For $\beta \to \infty$ the probability for accepting a move from 4 to 3 converges to 0, as $p(4) > p(3)$, thus the chain cannot escape from the local maximum at 4.

Sampling from $f_{(\beta)}(\cdot)$ is difficult

- For large β the distribution $f_{(\beta)}(\cdot)$ is increasingly concentrated around its modes.
- For large β sampling from $f_{(\beta)}$ gets increasingly difficult.
- Remedy: Start with a small β_0 and let β_t slowly increase.
- The sequence β_t determines whether local extrema are escaped.

Simulated Annealing: Minimising an arbitrary function

- More general objective: find global minima of a function $H : E \to \mathbb{R}_+$.
- Idea: Consider a distribution

$$f(x) \propto \exp(-H(x)) \text{ for } x \in E,$$

yielding

$$f_{(\beta_t)}(x) = (f(x))^{\beta_t} \propto \exp(-\beta_t \cdot H(x)) \text{ for } x \in E.$$

$\rightsquigarrow$ back to the framework of the previous slides.

- In this context β_t is often referred to as *inverse temperature*.

Simulated Annealing: Algorithm

Algorithm: Simulated Annealing

Starting with $\mathbf{X}^{(0)} := (X_1^{(0)}, \dots, X_p^{(0)})$ and $\beta^{(0)} > 0$ iterate for $t = 1, 2, \dots$

1. Increase β_{t-1} to β_t.
2. Draw $\mathbf{X} \sim q(\cdot|\mathbf{X}^{(t-1)})$.
3. Compute

$$\alpha(\mathbf{X}|\mathbf{X}^{(t-1)}) = \min\left\{1, \exp\left(-\beta_t\left(H((\mathbf{X}) - H(\mathbf{X}^{(t-1)})\right)\right) \cdot \frac{q(\mathbf{X}^{(t-1)}|\mathbf{X})}{q(\mathbf{X}|\mathbf{X}^{(t-1)})}\right\}.$$

4. With probability $\alpha(\mathbf{X}|\mathbf{X}^{(t-1)})$ set $\mathbf{X}^{(t)} = \mathbf{X}$, otherwise set $\mathbf{X}^{(t)} = \mathbf{X}^{(t-1)}$.

Annealing schedules

- As before $\mathbf{X}^{(t)}$ converges for $\beta_t \to \infty$ to a *local* minimum of $H(\cdot)$.
- Convergence to a *global* minimum depends on annealing schedule:

  Logarithmic tempering $\beta_t = \frac{\log(1+t)}{\beta_0}$.
  Good theoretical properties; practically irrelevant.

  Geometric tempering $\beta_t = \alpha^t \cdot \beta_0$ for some $\alpha > 1$. Popular choice, no theoretical convergence results.
- In practice: expect simulated annealing to find a "good" *local* minimum, but don't expect it to find the *global* minimum!

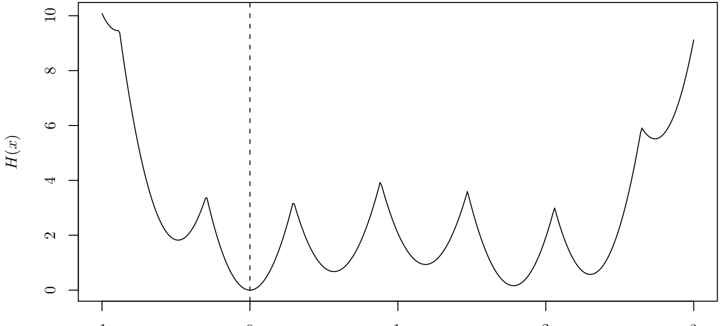
SA Example (1)

Minimise

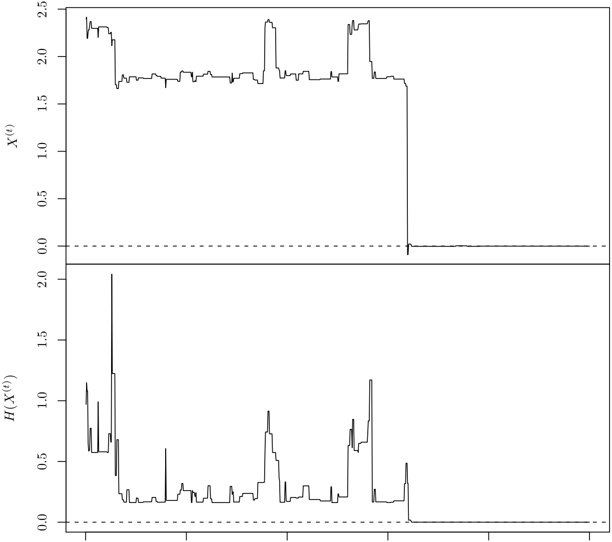
$$H(x) = \left((x-1)^2 - 1\right)^2 + 3 \cdot s(11.56 \cdot x^2)$$

with

$$s(x) = \begin{cases} |x| \mod 2 & \text{for } 2k \leq |x| \leq 2k+1,\ k \in \mathbb{N}_0 \\ 2 - |x| \mod 2 & \text{for } 2k+1 \leq |x| \leq 2(k+1),\ k \in \mathbb{N}_0 \end{cases}$$



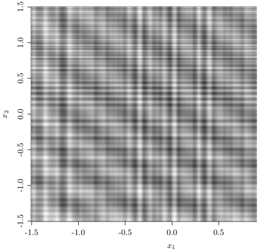
SA Example (2)



A More Challenging Example

- Consider:

$$f(x_1, x_2) = \\ \exp(\sin(50x_1)) + \sin(60\exp(x_2)) + \\ \sin(70\sin(x_1)) + \sin(\sin(80x_2)) - \\ \sin(10(x_1+x_2)) + \frac{1}{4}(x_1^2+x_2^2)$$



- What is its minimum?
- This question was part of SIAM's 2002 hundred-dollar, hundred-digit challenge (*SIAM News*, Volume 35, Number 1).
- It is on the assessment.

Summary of Part 3

- Motivation
- MCMC
- Gibbs Samplers
- Metropolis–Hastings-type Algorithms
- Simulated Annealing

Part 4

Theory and Practice

Part 4— Section 11

Theoretical Considerations and Convergence Results

Irreducibility and recurrence of Gibbs Samplers

Proposition

If the joint distribution $f(x_1, \ldots, x_p)$ satisfies the positivity condition, the Gibbs sampler yields an f-irreducible, recurrent Markov chain.

Outline Proof

Given an $\mathcal{X}$ such that $\int_{\mathcal{X}} f(x_1^{(t)}, \ldots, x_p^{(t)}) d(x_1^{(t)}, \ldots, x_p^{(t)}) > 0$.

$$\int_{\mathcal{X}} K(\mathbf{x}^{(t-1)}, \mathbf{x}^{(t)}) d\mathbf{x}^{(t)} = \int_{\mathcal{X}} \underbrace{f_{X_1|X_{-1}}(x_1^{(t)}|x_2^{(t-1)}, \ldots, x_p^{(t-1)})}_{>0} \cdots \underbrace{f_{X_p|X_{-p}}(x_p^{(t)}|x_1^{(t)}, \ldots, x_{p-1}^{(t)})}_{>0} \, d\mathbf{x}^{(t)}$$

Ergodic theorem

Theorem (Ergodicity of the Gibbs Sampler)

If the Markov chain generated by the Gibbs sampler is irreducible and recurrent (which is e.g. the case when the positivity condition holds), then for any integrable function $\varphi : E \rightarrow \mathbb{R}$

$$\lim_{n\to\infty} \frac{1}{n}\sum_{t=1}^{n} \varphi(\mathbf{X}^{(t)}) \overset{a.s.}{=} \mathbb{E}_f\left(\varphi(\mathbf{X})\right)$$

for almost every starting value $\mathbf{X}^{(0)}$.

Thus we can approximate expectations $\mathbb{E}_f\left(\varphi(\mathbf{X})\right)$ by their empirical counterparts using *a single* Markov chain.

A Simple Example

- Consider

$$\begin{pmatrix} X_1 \\ X_2 \end{pmatrix} \sim \mathrm{N}_2\left(\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \begin{pmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{12} & \sigma_2^2 \end{pmatrix}\right)$$

- Associated marginal distributions

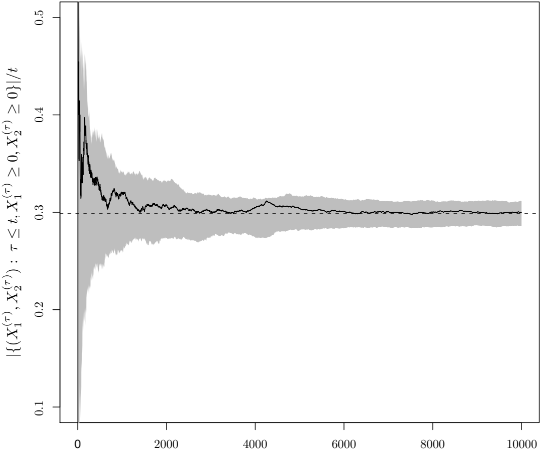
$$X_1 \sim \mathrm{N}(\mu_1, \sigma_1^2),$$
$$X_2 \sim \mathrm{N}(\mu_2, \sigma_2^2)$$

- Associated full conditionals

$$(X_1|X_2 = x_2) \sim \mathrm{N}(\mu_1 + \sigma_{12}/\sigma_2^2(x_2 - \mu_2), \sigma_1^2 - (\sigma_{12})^2\sigma_2^2)$$
$$(X_2|X_1 = x_1) \sim \mathrm{N}(\mu_2 + \sigma_{12}/\sigma_1^2(x_1 - \mu_1), \sigma_2^2 - (\sigma_{12})^2\sigma_1^2)$$

- Gibbs sampler consists of iterating for $t = 1, 2, \ldots$
  1. Draw $X_1^{(t)} \sim \mathrm{N}(\mu_1 + \sigma_{12}/\sigma_2^2(X_2^{(t-1)} - \mu_2), \sigma_1^2 - (\sigma_{12})^2\sigma_2^2)$.
  2. Draw $X_2^{(t)} \sim \mathrm{N}(\mu_2 + \sigma_{12}/\sigma_1^2(X_1^{(t)} - \mu_1), \sigma_2^2 - (\sigma_{12})^2\sigma_1^2)$.

Using the ergodic theorem we can estimate $\mathbb{P}(X_1 \geq 0, X_2 \geq 0)$ by the proportion of samples $(X_1^{(t)}, X_2^{(t)})$ with $X_1^{(t)} \geq 0$ and $X_2^{(t)} \geq 0$:



Theoretical properties of Metropolis–Hastings

- The Markov chain $(\mathbf{X}^{(0)}, \mathbf{X}^{(1)}, \dots)$ is (strongly) irreducible if $q(\mathbf{x}|\mathbf{x}^{(t-1)}) > 0$ for all $\mathbf{x}, \mathbf{x}^{(t-1)} \in \mathrm{supp}(f)$.
  (See, e.g., Roberts & Tweedie, 1996, for weaker conditions.)
- Such a chain is recurrent if it is irreducible.
  (See e.g., Tierney, 1994.)
- The chain is aperiodic if there is positive probability that the chain remains in the current state, i.e. $\mathbb{P}(\mathbf{X}^{(t)} = \mathbf{X}^{(t-1)}) > 0$ (for a suitable group of "current states").

Theorem (A Simple Ergodic Theorem)

If $(X_i)_{i\in\mathbb{N}}$ is an f-irreducible, f-invariant, recurrent $\mathbb{R}^d$-valued Markov chain then the following strong law of large numbers holds for any integrable function $\varphi : \mathbb{R}^d \to \mathbb{R}$:

$$\lim_{t\to\infty} \frac{1}{t} \sum_{i=1}^{t} \varphi(X_i) \overset{a.s.}{=} \int \varphi(x) f(x) dx.$$

for almost every starting value x.

Theorem (A Central Limit Theorem)

Under technical regularity conditions the following CLT holds for a recurrent, f-invariant Markov chain, and a function $\varphi : E \to \mathbb{R}$ which has at least two finite moments:

$$\lim_{t\to\infty} \sqrt{t}\left[\frac{1}{t}\sum_{i=1}^{t}\varphi(X_i) - \int \varphi(x) f(x) dx\right] \overset{\mathcal{D}}{=} N\left(0, \sigma^2(\varphi)\right),$$

$$\sigma^2(\varphi) = \mathbb{E}\left[(f(X_1) - \bar{\varphi})^2\right] + 2\sum_{k=2}^{\infty}\mathbb{E}\left[(\varphi(X_1) - \bar{\varphi})(\varphi(X_k) - \bar{\varphi})\right],$$

where $\bar{\varphi} = \int \varphi(x) f(x) dx$.

Optimal Scaling

Much effort has gone into determining optimal scaling rules:

Diffusion Limits Under strong assumptions:

$$\lim_{p\to\infty} \frac{X_1^{(\lfloor tp \rfloor)}}{\sqrt{p}} \xrightarrow{d} \text{Diffusion}$$

where p is *dimension* and the *speed* of the diffusion depends upon proposal scale.

ESJD Seek to maximise:

$$\int f(x)K(x,y;\theta)(y-x)^2 dxdy$$

Rule of Thumb Optimal RWM Scaling depends upon dimension:

$p = 1$ Acceptance rate of around 0.44.

$p \geq 5$ Acceptance rate of around 0.234.

Part 4— Section 12

Convergence Diagnostics

The need for convergence diagnostics

- Theory guarantees (under certain conditions) the convergence of the Markov chain $\mathbf{X}^{(t)}$ to the desired distribution.
- This does not imply that a *finite* sample from such a chain yields a good approximation to the target distribution.
- Validity of the approximation must be confirmed in practice.
- Convergence diagnostics help answering this question.
- Convergence diagnostics are *not* perfect and should be treated with a good amount of scepticism.

Different diagnostic tasks

Convergence to the target distribution Does $\mathbf{X}^{(t)}$ yield a sample from the target distribution?

- Has reached $(\mathbf{X}^{(t)})_t$ a stationary regime?
- Does $(\mathbf{X}^{(t)})_t$ cover the support of the target distribution?

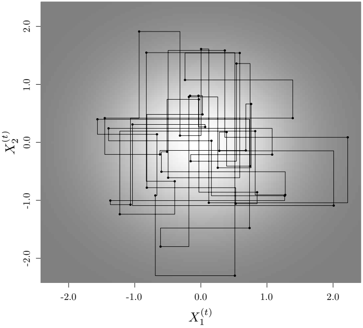
Convergence of averages Is $\sum_{t=1}^{T} \varphi(\mathbf{X}^{(t)})/T \approx \mathbb{E}_f(\varphi(\mathbf{X}))$?

Comparison to i.i.d. sampling How much information is contained in the sample from the Markov chain compared to an i.i.d. sample?

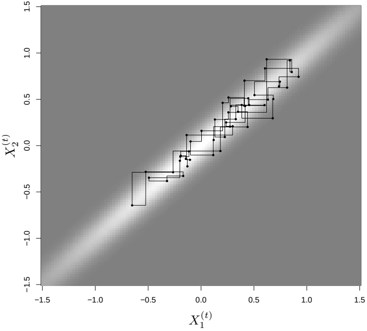
Pathological example 1: potentially slowly mixing

Gibbs sampler from a bivariate Gaussian with correlation $\rho(X_1, X_2)$

$\rho(X_1, X_2) = 0.3$



$\rho(X_1, X_2) = 0.99$



For correlations $\rho(X_1, X_2)$ close to ±1 the chain mixes poorly.

Pathological example 2: no central limit theorem

The following MCMC algorithm has the Beta$(\alpha, 1)$ distribution as stationary distribution:

Starting with any $X^{(0)}$ iterate for $t = 1, 2, \ldots$

1. With probability $1 - X^{(t-1)}$, set $X^{(t)} = X^{(t-1)}$.
2. Otherwise draw $X^{(t)} \sim \text{Beta}(\alpha + 1, 1)$.

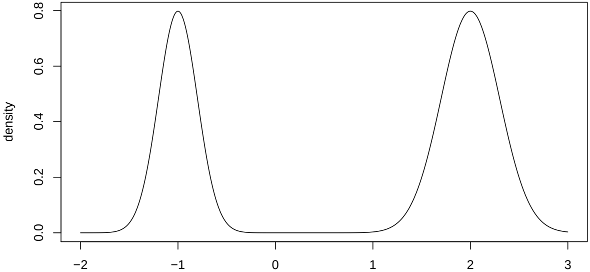
Markov chain converges very slowly (no central limit theorem applies).

Pathological example 3: nearly reducible chain

Metropolis–Hastings sample from a mixture of two well-separated Gaussians, i.e. the target is

$$f(x) = 0.4 \cdot \phi_{(-1,0.2^2)}(x) + 0.6 \cdot \phi_{(2,0.3^2)}(x).$$

If the variance of the proposal is too small, the chain cannot move from one population to the other.



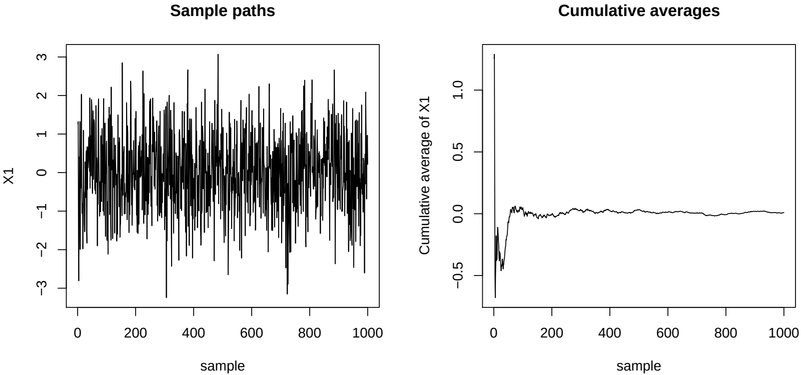
Basic plots

- Plot the sample paths $(X_j^{(t)})_t$.

  should be oscillating very fast and show very little structure.
- Plot the cumulative averages $(\sum_{\tau=1}^{t} \varphi(X_j^{(\tau)})/t)_t$.

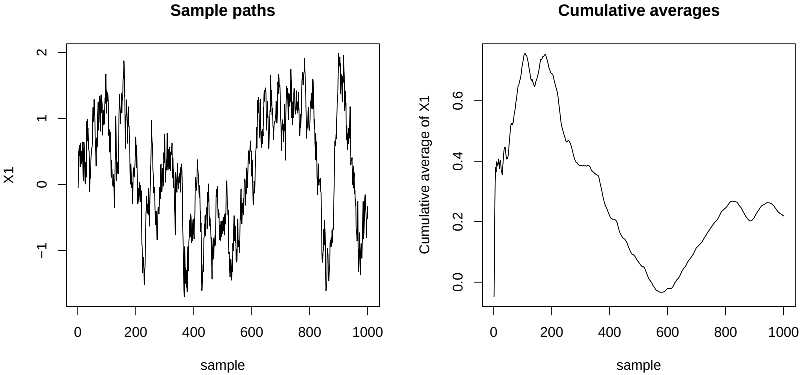
  should be converging to a value.
- Only very obvious problems visible in these plots.
- Difficult to assess multivariate distributions from univariate projections.

Plots for pathological example 1 ($\rho(X_1, X_2) = 0.3$)



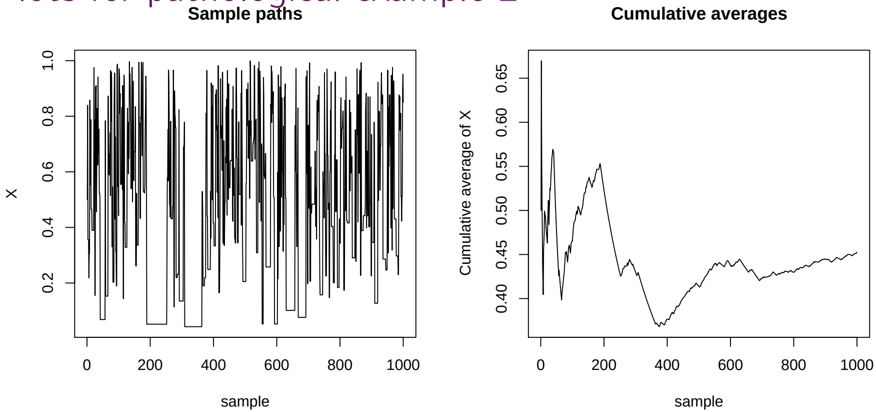
Looks OK.

Plots for pathological example 1 ($\rho(X_1, X_2) = 0.99$)



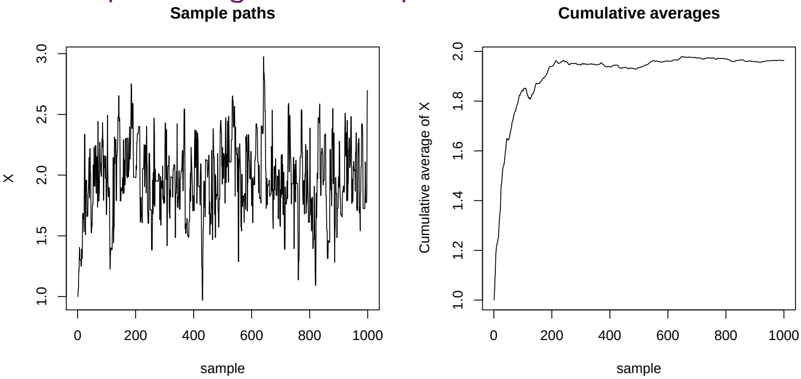
Slow mixing speed can be detected.

Plots for pathological example 2



Slow convergence of the mean can be detected.

Plots for pathological example 3

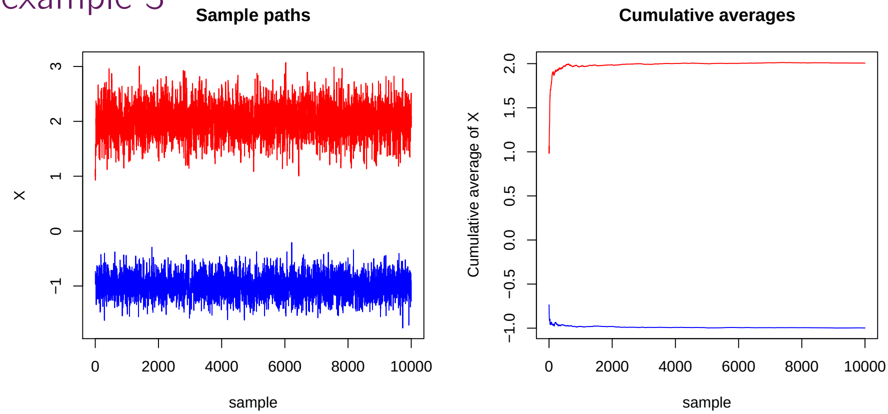


We *cannot* detect that the sample only covers one part of the distribution.
("you've only seen where you've been")

Comparing multiple chains

- Compare $L > 1$ chains $(\mathbf{X}^{(1,t)})_t, \ldots, (\mathbf{X}^{(L,t)})_t$.
- Initialised using overdispersed values $\mathbf{X}^{(1,0)}, \ldots, \mathbf{X}^{(L,0)}$.
- Idea: Variance and range of each chain $(\mathbf{X}^{(l,t)})_t$ should equal the range and variance of all chains pooled together.
- Compare basic plots for the different chains.
- Quantitative measure:
  - Compute distance $\delta_\alpha^{(l)}$ between α and $(1-\alpha)$ quantile of $(X_k^{(l,t)})_t$.
  - Compute distance $\delta_\alpha^{(\cdot)}$ between α and $(1-\alpha)$ quantile of the pooled data.
  - The ratio $\hat{S}_\alpha^{\text{interval}} = \dfrac{\sum_{l=1}^{L} \delta_\alpha^{(l)}/L}{\delta_\alpha^{(\cdot)}}$ should be around 1.
- Alternative: compare variance within each chain with the pooled variance estimate.
- Choosing suitable initial values $\mathbf{X}^{(1,0)}, \ldots, \mathbf{X}^{(L,0)}$ difficult.

Comparing multiple chains plots for pathological example 3



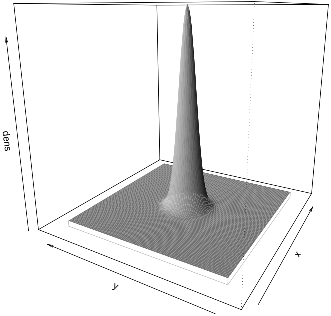
$\hat{S}_{\alpha}^{\text{interval}} = 0.2703 \ll 1$; we can detect that the sample only covers one part of the distribution.

Comparing multiple chains: A warning

- Consider the Witch's hat distribution:

$$f(x_1,x_2) \propto \begin{cases} (1-\delta)\phi_{(\boldsymbol{\mu},\sigma^2\cdot\mathbb{I})}(x_1,x_2)+\delta & \text{if } x_1,x_2 \in (0,1) \\ 0 & \text{otherwise.} \end{cases}$$

- Assume we want to estimate $\mathbb{P}(0.49 < X_1, X_2 \leq 0.51)$ for $\delta = 10^{-3}$, $\boldsymbol{\mu} = (0.5, 0.5)'$, and $\sigma = 10^{-5}$.



Comparing multiple chains: A warning (II)

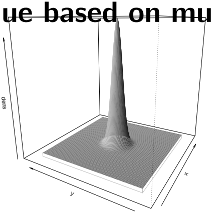
- We can use a Gibbs sampler. Conditional distribution:

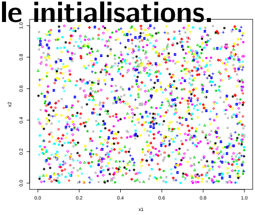
$$f(x_1|x_2) \propto \begin{cases} (1-\delta)\phi_{(\boldsymbol{\mu},\sigma^2.\mathbb{I})}(x_1,x_2)+\delta & \text{for } x_1 \in (0,1) \\ 0 & \text{otherwise.} \end{cases}$$

- But on average only 0.04% of the sampled values lie in $(0.49, 0.51) \times (0.49, 0.51)$ yielding an estimate of:

$$\widehat{\mathbb{P}}(0.49 < X_1, X_2 \leq 0.51) = 0.0004.$$

- **It is close to impossible to detect this problem with any technique based on multiple initialisations.**





Riemann sums and control variates

- Consider order statistic $X^{[1]} \leq \cdots \leq X^{[T]}$.
- Provided $(X^{[t]})_t = 1 \ldots, T$ covers the support of the target, the Riemann sum

$$\sum_{t=2}^{T} (X^{[t]} - X^{[t-1]}) f(X^{[t]})$$

converges to

$$\int f(x) dx = 1.$$

- Thus if $\sum_{t=2}^{T} (X^{[t]} - X^{[t-1]}) f(X^{[t]}) \ll 1$, the Markov chain has failed to explore all the support of the target.
- Requires that target density f is available inclusive of normalisation constants.
- Only effective in 1D.

Riemann sums for pathological example 3

For the chain stuck in the population with mean 2 we obtain

$$\sum_{t=2}^{T}(X^{[t]}-X^{[t-1]})f(X^{[t]})=0.598 \ll 1,$$

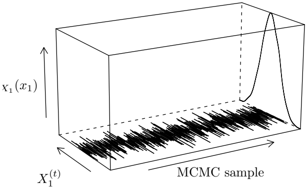
so we can detect that we have not explored the whole distribution.

Effective sample size

- MCMC algorithms yield a positively correlated sample $(\mathbf{X}^{(t)})_{t=1,\ldots,T}$.
- How much less useful is an MCMC sample of size T than an i.i.d. sample of size T?
- Approximate $(\varphi(\mathbf{X}^{(t)}))_{t=1,\ldots,T}$ by an $AR(1)$ process, i.e.:
$$\rho(\varphi(\mathbf{X}^{(t)}), \varphi(\mathbf{X}^{(t+\tau)})) = \rho^{|\tau|}.$$
- Variance of the estimator is
$$\mathrm{Var}\left(\frac{1}{T}\sum_{t=1}^{T}\varphi(\mathbf{X}^{(t)})\right) \approx \frac{1+\rho}{1-\rho}\cdot\frac{1}{T}\mathrm{Var}\left(\varphi(\mathbf{X}^{(t)})\right)$$
- Same variance as an i.i.d. sample of the size $T\cdot\dfrac{1-\rho}{1+\rho}$.
- Thus define $T\cdot\dfrac{1-\rho}{1+\rho}$ as *effective sample size*.

Effective sample for pathological example 1

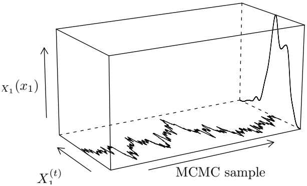
Rapidly mixing chain
($\rho(X_1, X_2) = 0.3$)
10,000 samples



$\rho(X_1^{(t-1)}, X_1^{(t)}) = 0.078$

ESS for estimating $\mathbb{E}_f(X_1)$ is 8,547.

Slowly mixing chain
($\rho(X_1, X_2) = 0.99$)
10,000 samples



$\rho(X_1^{(t-1)}, X_1^{(t)}) = 0.979$

ESS for estimating $\mathbb{E}_f(X_1)$ is 105.

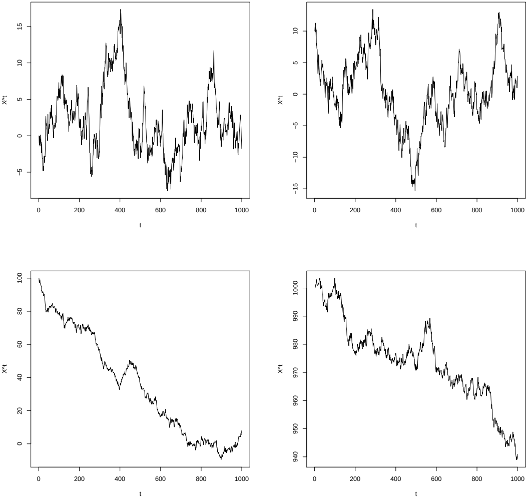
What Else Can We Do?

1. More sophisticated convergence diagnostics:
   - Geweke's method based on spectral analysis
   - Raftery's binary-chain method
   - $\vdots$
2. Theoretical Computations
   - Convergence rates
   - Mixing times
   - Confidence intervals
3. Perfect Simulation
   - Processes with "ordered transitions".
   - Certain spatial processes.

Part 4— Section 13

Practical Considerations

Where do we start?



RWM Traces.

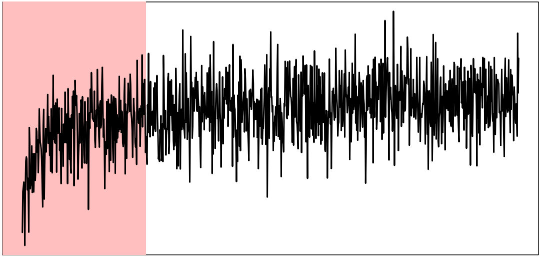
Target:
$f(x) = \mathrm{e}^{-|x|/5}/10$

Starting values:

- $X^{(1)} = 0$
- $X^{(1)} = 10$
- $X^{(1)} = 100$
- $X^{(1)} = 1,000$

Practical considerations: Burn-in period

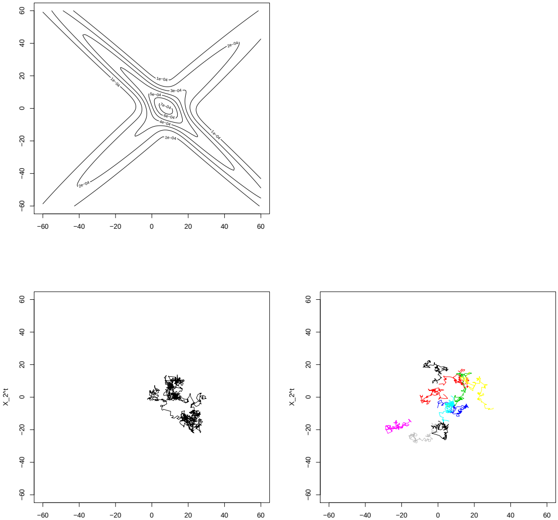
- Theory (ergodic theorems) allows for the use of the entire chain $(\mathbf{X}^{(0)}, \mathbf{X}^{(1)}, \dots)$.
- However distribution of $(\mathbf{X}^{(t)})$ for small t might still be far from the stationary distribution f.
- Can be beneficial to discard the first iterations $\mathbf{X}^{(t)}$, $t = 1, \dots, T_0$ (*burn-in period*).
- Optimal T_0 depends on mixing properties of the chain.



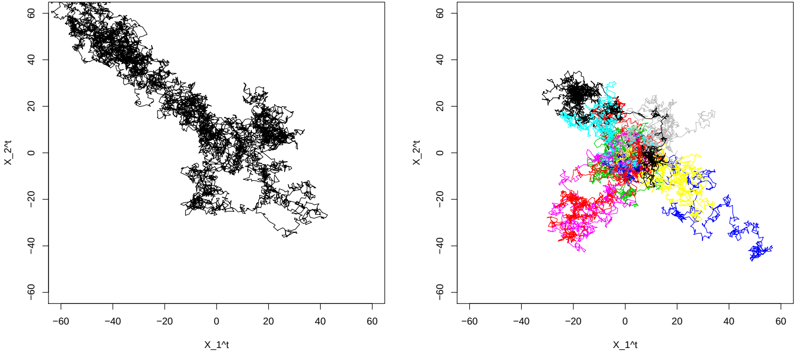
Practical considerations: Multiple Starts?

- Should we use “multiple overdispersed initialisations”?
- Advantages:
  - Exploring different parts of the space.
  - May be useful for assessing convergence.
  - Trivial to parallelize.
- Disadvantages:
  - We need to specify many starting values.
  - What does overdispersed mean, anyway?
  - Every chain needs to reach stationarity.
  - Multiple burn-in periods may be expensive.

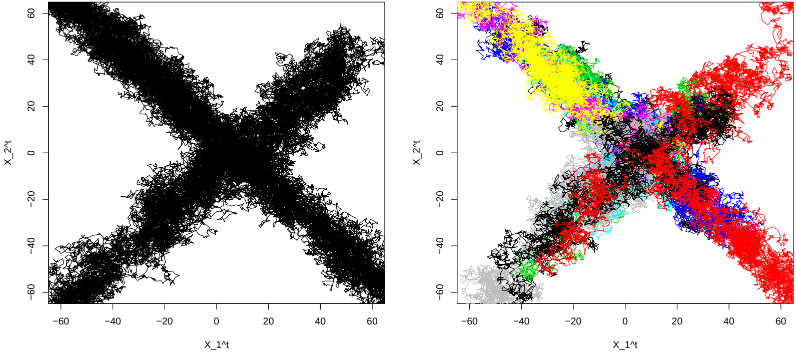
One Chain vs. Many: 1000 or 10 × 100



One Chain vs. Many: $10,000$ or 10×1000



One Chain vs. Many: $100,000$ or $10 \times 10,000$



Practical considerations: Thinning (1)

- MCMC methods typically yield positively correlated chain: $\rho(\mathbf{X}^{(t)}, \mathbf{X}^{(t+\tau)})$ large for small τ.
- Idea: keeping only every m-th value: $(\mathbf{Y}^{(t)})_{t=1,\ldots,\lfloor T/m \rfloor}$ with $\mathbf{Y}^{(t)} = \mathbf{X}^{(m \cdot t)}$ instead of $(\mathbf{X}^{(t)})_{t=1,\ldots,T}$ (*thinning*).
- $(\mathbf{Y}^{(t)})_t$ exhibits less autocorrelation than $(\mathbf{X}^{(t)})_t$, i.e.

$$\rho(\mathbf{Y}^{(t)}, \mathbf{Y}^{(t+\tau)}) = \rho(\mathbf{X}^{(t)}, \mathbf{X}^{(t+m\cdot\tau)}) < \rho(\mathbf{X}^{(t)}, \mathbf{X}^{(t+\tau)}),$$

  if the correlation $\rho(\mathbf{X}^{(t)}, \mathbf{X}^{(t+\tau)})$ decreases monotonically in τ.
- Price: length of $(\mathbf{Y}^{(t)})_{t=1,\ldots,\lfloor T/m \rfloor}$ is only $(1/m)$-th of the length of $(\mathbf{X}^{(t)})_{t=1,\ldots,T}$.

Practical considerations: Thinning (2)

- If $\mathbf{X}^{(t)} \sim f$ and corresponding variances exist,

$$\operatorname{Var}\left(\frac{1}{T}\sum_{t=1}^{T}\varphi(\mathbf{X}^{(t)})\right) \leq \operatorname{Var}\left(\frac{1}{\lfloor T/m \rfloor}\sum_{t=1}^{\lfloor T/m \rfloor}\varphi(\mathbf{Y}^{(t)})\right),$$

  i.e. thinning cannot be justified when objective is estimating $\mathbb{E}_f(\varphi(\mathbf{X}))$.
- Thinning can be a useful concept
  - if computer has insufficient memory.
  - for convergence diagnostics: $(\mathbf{Y}^{(t)})_{t=1,\dots,\lfloor T/m \rfloor}$ is closer to an i.i.d. sample than $(\mathbf{X}^{(t)})_{t=1,\dots,T}$.

Part 5

Alternative approaches

Part 5— Section 14

Augmentation

Augmentation

- "Making the *space* bigger to make the problem easier."
- To target a distribution $f_X(\boldsymbol{x})$:
    - Construct some $f_{X,Z}(\boldsymbol{x}, \boldsymbol{z})$ on $\mathcal{X} \otimes \mathcal{Z}$
    - such that

$$f_X(\boldsymbol{x}) = \int_{\mathcal{Z}} f_{X,Z}(\boldsymbol{x}, \boldsymbol{z}) d\boldsymbol{z}$$

    - and $f_{X,Z}$ is easy to sample from (when f_X is not).
- Versatile technique with many applications.

A Generic Augmentation Scheme

- Given any density $f(\mathbf{x})$, define

$$f(\mathbf{x}, u) := f(\mathbf{x}) \cdot f_{U|X}(u|\mathbf{x})$$

- with

$$f_{U|X}(u|\mathbf{x}) = \frac{1}{f(\mathbf{x})} \mathbb{I}_{[0,f(x)]}(u)$$

- Then

$$f(\mathbf{x}, u) = \mathbb{I}_{[0,f(x)]}(u).$$

Rejection Sampling Revisited

Proposition (Rejection Sampling Equivalence)

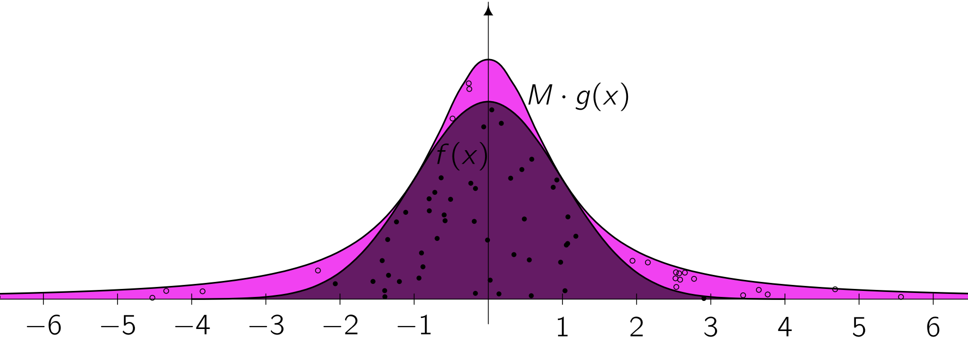
- *Given* $f(\mathbf{x})$, *define*

$$f(\mathbf{x}, u) = \mathbb{I}_{[0,f(\mathrm{x})]}(u).$$

- *Given proposal* $g(\mathbf{x})$ *and* $M \geq \sup_{\mathrm{x}} f(\mathbf{x})/g(\mathbf{x})$, *define*

$$g(\mathbf{x}, u) = \frac{1}{M}\mathbb{I}_{[0,M\cdot g(\mathrm{x})]}(u).$$

- *Let* $w(\mathbf{x}, u) = f(\mathbf{x}, u)/g(\mathbf{x}, u)$
- *The associated self-normalised importance sampling estimator of* $\mathbb{E}_f[\varphi(\mathbf{X})]$ *is the rejection sampling estimator.*



Sample uniformly and weight. . .

Slice Sampling

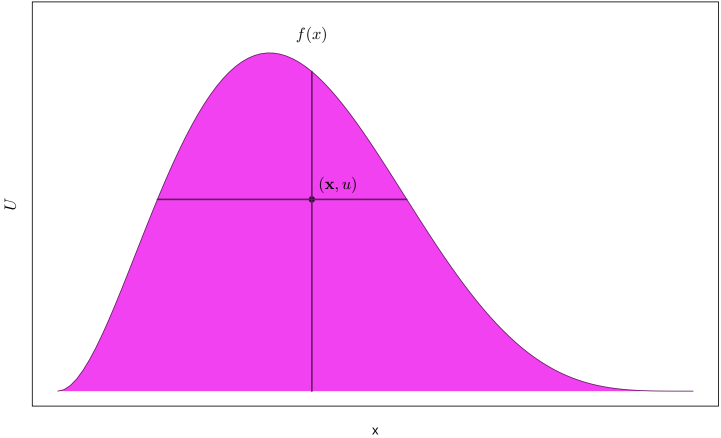
- Rejection sampling can be viewed as importance sampling with an extended target distribution. . .
- so can we apply other algorithms to that extended distribution?

Algorithm: The Slice Sampler

Starting with $(\mathbf{X}^{(0)}, U^{(0)})$ iterate for $t = 1, 2, \ldots$

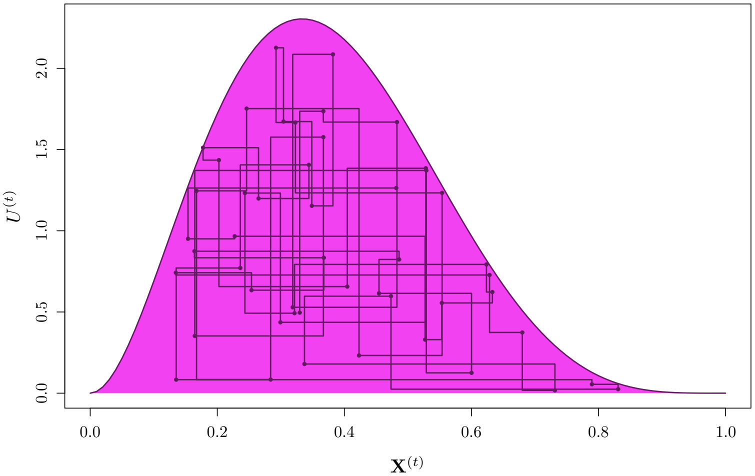
1. Draw $\mathbf{X}^{(t)} \sim f_{\mathrm{X}|U}(\cdot|U^{(t-1)})$.
2. Draw $U^{(t)} \sim f_{U|\mathrm{X}}(\cdot|\mathbf{X}^{(t)})$.

An Illustration of the Conditional Distributions



A Slice-Sampler Trajectory

Example: Sampling from a Beta$(3, 5)$ **distribution**



How Practical Is This?

- Sampling $U \sim \mathrm{U}[0, f(\mathbf{X})]$ is *easy*.
- Sampling $\mathbf{X} \sim \mathrm{U}(L(U))$ where

$$L(u) := \{\mathbf{x} : f(\mathbf{x}) \geq u\}$$

can be easy. . .

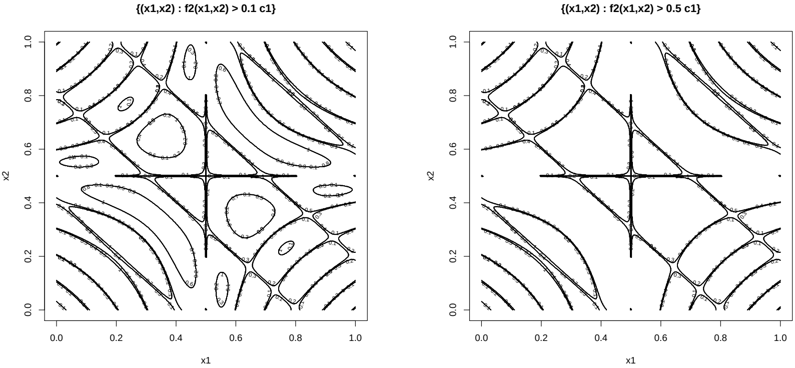
- but it might not be.
- Consider the bivariate density:

$$f_2(x_1, x_2) = c_1 \cdot \sin^2(x_1 \cdot x_2) \cdot \cos^2(x_1 + x_2) \cdot \exp(-\frac{1}{2}(|x_1| + |x_2|)).$$

The Trouble with Slice Sampling

Level sets of:

$$f_2(x_1, x_2) = c_1 \cdot \sin^2(x_1 \cdot x_2) \cdot \cos^2(x_1 + x_2) \cdot \exp(-\frac{1}{2}(|x_1| + |x_2|)).$$



Here we could use rejection.

Algorithm: The Co-ordinate-wise Slice Sampler

Starting with $(X_1^{(0)}, \dots, X_p^{(0)}, U^{(0)})$ iterate for $t = 1, 2, \dots$

1. Draw $X_1^{(t)} \sim f_{X_1|X_{-1},U}(\cdot|X_{-1}^{(t-1)}, U^{(t-1)})$.
2. Draw $X_2^{(t)} \sim f_{X_2|X_{-2},U}(\cdot|X_1^{(t)}, X_3^{(t-1)}, \dots, X_p^{(t-1)}, U^{(t-1)})$.

$$\vdots$$

p. Draw $X_p^{(t)} \sim f_{X_p|X_{-p},U}(\cdot|X_{-p}^{(t)}, U^{(t-1)})$.

p+1. Draw $U^{(t)} \sim f_{U|X}(\cdot|\mathbf{X}^{(t)})$.

Algorithm: The Metropolised Slice Sampler

Starting with $(\mathbf{X}^{(0)}, U^{(0)})$ iterate for $t = 1, 2, \ldots$

1. Draw $\mathbf{X} \sim q(\cdot|\mathbf{X}^{(t-1)}, U^{(t-1)})$.
2. With probability

$$\min\left(1, \frac{f(\mathbf{X}, U^{(t-1)})q(\mathbf{X}^{(t-1)}|\mathbf{X}, U^{(t-1)})}{f(\mathbf{X}^{(t-1)}, U^{(t-1)})q(\mathbf{X}|\mathbf{X}^{(t-1)}, U^{(t-1)})}\right)$$

   accept and set $\mathbf{X}^{(t)} = \mathbf{X}$.
   Otherwise, set $\mathbf{X}^{(t)} = \mathbf{X}^{(t-1)}$.

2. Draw $U^{(t)} \sim f_{U|X}(\cdot|\mathbf{X}^{(t)})$.

Data Augmentation I

- *Latent variable models* are common: statistical models with:
  - parameters $\boldsymbol{\theta}$,
  - observations $\boldsymbol{y}$, and
  - latent variables, $\boldsymbol{z}$.
- Typically, the joint distribution, $f_{\boldsymbol{Y},\boldsymbol{Z},\boldsymbol{\theta}}$, is known,
- but integrating out the latent variables to get $f_{\boldsymbol{Y},\boldsymbol{\theta}}$ is not feasible.
- Without $f_{\boldsymbol{Y},\boldsymbol{\theta}}$ we can't implement an MCMC algorithm targeting $f_{\boldsymbol{\theta}|\boldsymbol{Y}}$.
- The basis of data augmentation is to *augment* $\boldsymbol{\theta}$ with $\boldsymbol{z}$ and to run an MCMC algorithm which targets $f_{\boldsymbol{\theta},\boldsymbol{Z}|\boldsymbol{Y}}$.
- This distribution has the correct marginal in $\boldsymbol{\theta}$.

Data Augmentation and Gibbs Samplers

- Gibbs sampling is only feasible when we can sample easily from the full conditionals.
- A technique that can help achieving full conditionals that are easy to sample from is *demarginalisation*:
  Introduce a set of auxiliary random variables $Z_1, \dots, Z_r$ such that f is the marginal density of $(X_1, \dots, X_p, Z_1, \dots, Z_r)$, i.e.

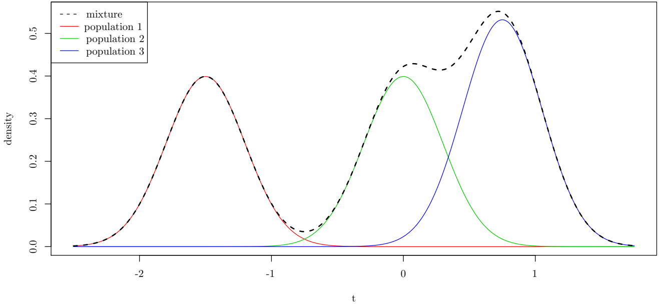
$$f(x_1, \dots, x_p) = \int f(x_1, \dots, x_p, z_1, \dots, z_r)\, d(z_1, \dots, z_r).$$

- In many cases there is a "natural choice" of the *completion* $(Z_1, \dots, Z_r)$.

Example: Mixture of Gaussians — Model

Consider the following K population mixture model for data $Y_1, \ldots, Y_n$:

$$f(y_i) = \sum_{k=1}^{K} \pi_k \phi_{(\mu_k, 1/\tau)}(y_i)$$



Objective: Bayesian inference for $(\pi_1, \ldots, \pi_K, \mu_1, \ldots, \mu_K)$.

Example: Mixture of Gaussians — Priors

- The number of components K is assumed to be known.
- The precision parameter τ is assumed to be known.
- $(\pi_1, \ldots, \pi_K) \sim \mathrm{Dirichlet}(\alpha_1, \ldots, \alpha_K)$, i.e.

$$f_{(\alpha_1,\ldots,\alpha_K)}(\pi_1, \ldots, \pi_K) = \frac{\Gamma(\sum_{k=1}^K \alpha_k)}{\prod_{k=1}^K \Gamma(\alpha_k)} \prod_{k=1}^K \pi_k^{\alpha_k - 1}.$$

- $(\mu_1, \ldots, \mu_K) \sim \mathrm{N}(\mu_0, 1/\tau_0)$, i.e.

$$f_{(\mu_0,\tau_0)}(\mu_k) \propto \exp\left(-\tau_0(\mu_k - \mu_0)^2/2\right).$$

Example: Mixture of Gaussians — Joint distribution

$$f(\mu_1,\dots,\mu_K,\pi_1,\dots,\pi_K,y_1,\dots,y_n) \propto \left(\prod_{k=1}^{K} \pi_k^{\alpha_k-1}\right) \cdot \left(\prod_{k=1}^{K} \exp\left(-\tau_0(\mu_k-\mu_0)^2/2\right)\right) \cdot \left(\prod_{i=1}^{n}\sum_{k=1}^{K} \pi_k \exp\left(-\tau(y_i-\mu_k)^2/2\right)\right)$$

The full conditionals do not seem to come from "nice" distributions.

Use data augmentation: include auxiliary variables $Z_1,\dots Z_n$ which indicate which population the i-th individual is from, i.e.

$$\mathbb{P}(Z_i = k) = \pi_k \qquad \text{and} \qquad Y_i|Z_i = k \sim N(\mu_k, 1/\tau).$$

The marginal distribution of Y is as before, so $Z_1,\dots Z_n$ are indeed a completion.

Example: Mixture of Gaussians — Joint distribution

The joint distribution of the augmented system is

$$
\begin{aligned}
& f(y_1, \ldots, y_n, z_1, \ldots, z_n, \mu_1, \ldots, \mu_K, \pi_1, \ldots, \pi_K) \\
\propto \quad & \left(\prod_{k=1}^{K} \pi_k^{\alpha_k - 1} \right) \cdot \left(\prod_{k=1}^{K} \exp\left(-\tau_0 (\mu_k - \mu_0)^2 / 2 \right) \right) \\
& \cdot \left(\prod_{i=1}^{n} \pi_{z_i} \exp\left(-\tau (y_i - \mu_{z_i})^2 / 2 \right) \right) .
\end{aligned}
$$

The full conditionals now come from “nice” distributions.

Example: Mixture of Gaussians — Full conditionals

$$\mathbb{P}(Z_i = k|Y_1, \dots, Y_n, \mu_1, \dots, \mu_K, \pi_1, \dots, \pi_K)$$
$$= \frac{\pi_k \phi_{(\mu_k, 1/\tau)}(y_i)}{\sum_{\iota=1}^{K} \pi_\iota \phi_{(\mu_\iota, 1/\tau)}(y_i)},$$

$$\mu_k|Y_1, \dots, Y_n, Z_1, \dots, Z_n, \pi_1, \dots, \pi_K$$
$$\sim \mathrm{N}\left(\frac{\tau\left(\sum_{i:\ Z_i=k} Y_i\right) + \tau_o\mu_0}{|\{i:\ Z_i = k\}|\tau + \tau_0}, \frac{1}{|\{i:\ Z_i = k\}|\tau + \tau_0}\right),$$

$$\pi_1, \dots, \pi_K|Y_1, \dots, Y_n, Z_1, \dots, Z_n, \mu_1, \dots, \mu_K$$
$$\sim \mathrm{Dirichlet}\left(\alpha_1 + |\{i:\ Z_i = 1\}|, \dots, \alpha_K + |\{i:\ Z_i = K\}|\right).$$

Example: Mixture of Gaussians — Gibbs sampler

Starting with initial values $\mu_1^{(0)}, \dots, \mu_K^{(0)}, \pi_1^{(0)}, \dots, \pi_K^{(0)}$ iterate for $t = 1, 2, \dots$

1. For $i = 1, \dots, n$:

   Draw $Z_i^{(t)}$ from the discrete distribution on $\{1, \dots, K\}$

$$\mathbb{P}(Z_i^{(t)} = k | Y_1, \dots, Y_n, \mu_1^{(t-1)}, \dots, \mu_K^{(t-1)}, \pi_1^{(t-1)}, \dots, \pi_K^{(t-1)}) = \frac{\pi_k \phi_{(\mu_k^{(t-1)}, 1/\tau)}(y_i)}{\sum_{\iota=1}^K \pi_\iota^{(t-1)} \phi_{(\mu_\iota^{(t-1)}, 1/\tau)}(y_i)}.$$

2. For $k = 1, \dots, K$:

   Draw $\mu_k^{(t)} \sim$

$$\mathrm{N}\left(\frac{\tau \left(\sum_{i:\, Z_i^{(t)} = k} Y_i \right) + \tau_o \mu_0}{|\{i :\ Z_i^{(t)} = k\}| \tau + \tau_0}, \frac{1}{|\{i :\ Z_i^{(t)} = k\}| \tau + \tau_0} \right).$$

3. Draw

$$(\pi_1^{(t)}, \dots, \pi_K^{(t)}) \sim \text{Dirichlet}\left(\alpha_1 + |\{i :\ Z_i^{(t)} = 1\}|, \dots, \alpha_K + |\{i :\ Z_i^{(t)} = K\}| \right)$$

Towards approximate Bayesian computation

- Consider a target distribution $\pi(\theta|y)$ written as:

$$\pi(\theta|y) = \frac{f(y|\theta)p(\theta)}{p(y)}.$$

- If both $p(\theta)$ and $f(y|\theta)$ can be evaluated we're done.
- If we *cannot* evaluate $f(y|\cdot)$ even pointwise, then we *can't* directly use the techniques which we've described previously.
- Consider the case in which y is *discrete*.
- We can invoke a clever data augmentation trick which requires only that we can *sample* from $f(\cdot|\theta)$.

- We can define an extended distribution:

$$\pi(\theta, u|y) \propto f(u|\theta)p(\theta)\delta_{y,u}$$

  and note that it has, as a marginal distribution, our target:

$$\sum_u \pi(\theta, u|y) \propto \sum_u f(u|\theta)p(\theta)\delta_{y,u} = f(y|\theta)p(\theta).$$

- We can sample $(\theta, u) \sim f(u|\theta)p(\theta)$ and use this as a rejection sampling proposal for our target distribution, keeping samples with probability proportional to

$$\frac{\pi(\theta, u|y)}{f(u|\theta)p(\theta)} \propto \delta_{y,u}.$$

Approximate Bayesian Computation

- When data is not discrete / takes many values, exact matches have no or negligible probability.
- Instead, we keep samples for which $||u - y|| \leq \epsilon$.
- This leads to a *different* target distribution:

$$\pi^{\mathrm{ABC}}_{\theta,u|y}(\theta, y|u) \propto f(u|\theta)p(\theta)\mathbb{I}_{B(y,\epsilon)}(u),$$

where $B(y,\epsilon) := \{u : |u - y| \leq \epsilon\}$, so

$$\begin{aligned}\pi^{\mathrm{ABC}}_{\theta|y} \propto &\int f(u|\theta)p(\theta)\mathbb{I}_{B(y,\epsilon)}(u)du\\ \propto &p(\theta)\int f(u|\theta)\mathbb{I}_{B(y,\epsilon)}(u)du\\ \propto &p(\theta)\int_{u\in B(y,\epsilon)} f(u|\theta)du.\end{aligned}$$

This approximation amounts to a *smoothing* of the likelihood.

Even More Approximate Bayesian Computation

- Often a further approximation is introduced by considering not the data itself but some low dimensional summary of the data: This leads to a *different* target distribution:

$$\pi^{\mathrm{ABC}}_{\theta,u|y}(\theta, u|y) \propto f(u|\theta)p(\theta)\mathbb{I}_{B(s(y),\epsilon)}(s(u)).$$

- Unless the summary is a sufficient statistic (which it probably isn't) this introduces a difficult to understand approximation.
- Be very careful.

Exact-approximate methods

- Suppose that, for any θ, it is possible to compute an unbiased estimate $\widehat{f}\,(y|\theta)$ of $f\,(y|\theta)$. Then...

1. Using the acceptance probability

$$\alpha\left(\theta^{(i)},\theta^{*}\right)=\min\left\{1,\frac{\widehat{f}(y|\theta^{*})p(\theta^{*})q(\theta^{(i)}|\theta^{*})}{\widehat{f}(y|\theta^{(i)})p(\theta^{(i)})q(\theta^{*}|\theta^{(i)})}\right\}$$

yields an MCMC algorithm with target distibution $\pi\,(\theta|y)$.

2. Using the weight

$$w^{(i)}=\frac{\widehat{f}(y|\theta^{(i)})p(\theta^{(i)})}{q(\theta^{(i)})}$$

yields an importance sampling algorithm with target distribution $\pi\,(\theta|y)$.

Beaumont (2003), Andrieu and Roberts (2009), Fearnhead et al. (2010).

Why is this true?

- Write down the joint distrubution of *all* of the variables that are being used
$$\widehat{f}(y|\theta, u)p(u|\theta)p(\theta)$$
where u are the random variables used to generate the estimate $\widehat{f}$.
- An algorithm that simulates from $\pi(\theta, u|y)$ has the correct marginal
$$\begin{aligned}\int_u \pi(\theta, u|y)du &\propto \int_u \widehat{f}(y|\theta, u)p(u|\theta)p(\theta)du \\ &= p(\theta)\int_u \widehat{f}(y|\theta, u)p(u|\theta)du \\ &= p(\theta)f(y|\theta) \\ &\propto \pi\left(\theta|y\right).\end{aligned}$$

Why is this true?

- Using $q\left((\theta^*, u^*) \mid (\theta^{(i)}, u^{(i)})\right) = q(\theta^*|\theta^{(i)})p(u^*|\theta^*)$ as a proposal within a Metropolis-Hastings algorithm yields the desired acceptance probability.

$$\min\left\{1, \frac{\widehat{f}(y|\theta^*, u^*)p(u^*|\theta^*)p(\theta^*)}{\widehat{f}(y|\theta^{(i)}, u^{(i)})p(u^{(i)}|\theta^{(i)})p(\theta^{(i)})} \frac{q(\theta^{(i)}|\theta^*)p(u^{(i)}|\theta^{(i)})}{q(\theta^*|\theta^{(i)})p(u^*|\theta^*)}\right\}$$

$$= \min\left\{1, \frac{\widehat{f}(y|\theta^*, u^*)p(\theta^*)}{\widehat{f}(y|\theta^{(i)}, u^{(i)})p(\theta^{(i)})} \frac{q(\theta^{(i)}|\theta^*)}{q(\theta^*|\theta^{(i)})}\right\}.$$

- A similar extended space representation may be used in importance sampling.

Part 5— Section 15

Sequential Monte Carlo

Returning to importance sampling

- Recall the self-normalised importance sampling estimate of $\mathbb{E}_\pi[\theta]$

  $$\sum_{i=1}^{N} \theta^{(i)} \frac{\tilde{w}^{(i)}}{\sum_{j=1}^{N} \tilde{w}^{(j)}}$$

  where

  $$w^{(i)} = \tilde{w}\left(\theta^{(i)}\right) = \frac{p(\theta^{(i)}) f(y|\theta^{(i)})}{q(\theta^{(i)})}$$

  and $\left\{\theta^{(i)}\right\}_{i=1}^{N}$ are independent points simulated from $q(\theta)$.
- The variance of these estimators depends on the “distance” between π and q.
- To control the variance of the estimates, we should choose q to have heavier tails than π.

Returning to importance sampling

- Compared to MCMC:
  - a bit simpler
  - obtain estimates of the marginal likelihood, where MCMC doesn't
  - the proposal is our only way of exploring the space - we cannot use local moves as in MCMC.

Improving IS

- Can we improve on the weaknesses of IS?
  - can we construct a q that is close to π?
- Idea:
  - introduce intermediate distributions between q and π, and perform importance sampling sequentially.
- What are "intermediate" distributions?
- One idea is to use tempering of the likelihood. Choose

$$\pi_t\left(\theta \mid y\right) = p\left(\theta\right) f\left(y \mid \theta\right)^{\gamma_t}$$

for $0 = \gamma_0 \leq \gamma_1 \leq ... \leq \gamma_T$.

A sequential importance sampling approach

- Suppose we draw points from $\pi_0 = q$, the original proposal we used in IS.
- Then use IS with proposal π_0 and target π_1:
  - weight the points using unnormalized weights $\frac{\pi_1(\theta_1)}{\pi_0(\theta_1)}$.
- We then wish to somehow use these weighted points to help us sample from π_2.
- Suppose we just use them directly:
  - there is no gain, since nothing changes that they are simply sampled from q!

A sequential importance sampling approach

- Suppose we move them a little:
  - for each point, use a "kernel" $K\left(\cdot \mid \theta_1\right)$ centered at the current point.
- For initial point θ_1, we simulate $\theta_2 \sim K\left(\cdot \mid \theta_1\right)$.
- Then use θ_2 points as proposals in an importance sampler.
- What is the distribution of these points?

$$\int_{\theta_1} \pi_0\left(\theta_1\right) K\left(\theta_2 \mid \theta_1\right) d\theta_1$$

- Therefore our importance weight is

$$\frac{\pi_2\left(\theta_2\right)}{\int_{\theta_1} \pi_0\left(\theta_1\right) K\left(\theta_2 \mid \theta_1\right) d\theta_1}$$

Problem and solution

- In general, we cannot analytically evaluate

$$\int_{\theta_1} \pi_0(\theta_1) K(\theta_2 \mid \theta_1) \, d\theta_1$$

- What can we do?
- We cannot marginalize over θ_1, but we can evaluate the joint distribution of the proposal

$$\pi_0(\theta_1) K(\theta_2 \mid \theta_1)$$

  - as long as K is chosen such that we can!
- Can we set up an importance sampler on some joint distribution on θ_1 and θ_2, that has marginal π_2?
- Yes, easily!
  - use $\pi_2(\theta_2) L(\theta_1 \mid \theta_2)$, where L is any normalized distribution on θ_1 given θ_2.

Constructing an SMC sampler

- Simulate $\theta_1 \sim \pi_0$.
- Simulate $\theta_2 \sim K\left(\cdot \mid \theta_1\right)$.
- Find unnormalized weight

$$\frac{\pi_2\left(\theta_2\right) L\left(\theta_1 \mid \theta_2\right)}{\pi_0\left(\theta_1\right) K\left(\theta_2 \mid \theta_1\right)}.$$

- Using self-normalising IS with points weighted in this way allows us to estimate expectations with respect to π_2 since we have correctly weighted points from the joint $\pi_2\left(\theta_2\right) L\left(\theta_1 \mid \theta_2\right)$.
- Note that so far, to keep the notation simple, we are simply seeing the procedure for a single point as in standard IS; we will repeat this N times.

Constructing an SMC sampler

- We would like to implement the approach sequentially, so that:
  - at step 1, we have weighted points from π_1,
  - at step 2, we have weighted points from π_2,
  - etc.
- Use the following approach:
  - Simulate $\theta_0 \sim \pi_0$.
  - Find unnormalized weight

$$w_1 = \frac{\pi_1\left(\theta_1\right)}{\pi_0\left(\theta_1\right)}.$$

Constructing an SMC sampler

- Simulate $\theta_2 \sim K\left(\cdot \mid \theta_1\right)$.
- At step 2, we would like to use a weight "update" that is written in terms of the weight from the previous step:

$$
\begin{aligned}
w_2 &= \frac{\pi_2\left(\theta_2\right) L\left(\theta_1 \mid \theta_2\right)}{\pi_0\left(\theta_1\right) K\left(\theta_2 \mid \theta_1\right)} \\
&= \frac{\pi_1\left(\theta_1\right)}{\pi_0\left(\theta_1\right)} \frac{\pi_2\left(\theta_2\right)}{\pi_1\left(\theta_1\right)} \frac{L\left(\theta_1 \mid \theta_2\right)}{K\left(\theta_2 \mid \theta_1\right)} \\
&= w_1 \frac{\pi_2\left(\theta_2\right)}{\pi_1\left(\theta_1\right)} \frac{L\left(\theta_1 \mid \theta_2\right)}{K\left(\theta_2 \mid \theta_1\right)}.
\end{aligned}
$$

Constructing an SMC sampler

- In general, we have the following steps:
  - Simulate $\theta_t \sim K_t\left(\cdot \mid \theta_{t-1}\right)$.
  - Use a weight "update" that is written in terms of the weight from the previous step

$$w_t = w_{t-1} \frac{\pi_t\left(\theta_t\right)}{\pi_{t-1}\left(\theta_{t-1}\right)} \frac{L_{t-1}\left(\theta_{t-1} \mid \theta_t\right)}{K_t\left(\theta_t \mid \theta_{t-1}\right)}.$$

How to choose K and L?

- K and L can be chosen however we like, and the algorithm is still valid.
- However, some choices are better than others:
  - we want to choose K_t such that it helps us explore the posterior,
  - a useful way of generating new points will help us explore the posterior and give an advantage over importance sampling.
- One idea:
  - choose K_t to be an MCMC kernel with stationary distribution π_t.

How to choose K and L?

- How should we choose L?
  - this will affect the variance of the estimates we get from the algorithm.
- If K_t is an MCMC kernel and π_t is not too far from π_{t+1} for all t, then choosing L_{t-1} to be the time reversal of K_t results in low variance estimates, i.e., choose L_{t-1} such that

$$\pi_t\left(\theta_{t-1}\right) K_t\left(\theta_t \mid \theta_{t-1}\right) = \pi_t\left(\theta_t\right) L_{t-1}\left(\theta_{t-1} \mid \theta_t\right).$$

SMC sampler with MCMC moves

- This results in the weight update

$$w_t = w_{t-1} \frac{\pi_t(\theta_t)}{\pi_{t-1}(\theta_{t-1})} \frac{L_{t-1}(\theta_{t-1} \mid \theta_t)}{K_t(\theta_t \mid \theta_{t-1})} = w_{t-1} \frac{\pi_t(\theta_{t-1})}{\pi_{t-1}(\theta_{t-1})}.$$

Missing detail

- There is a key detail missing that will prevent this from being a successful algorithm.
- The fact that we have written this sequentially has obscured the fact that we are simply sequentially constructing an importance sampler that is on the space of, at iteration t, t copies of θ.
- The target is $\pi_t\left(\theta_t\right) L_{t-1}\left(\theta_{t-1} \mid \theta_t\right) \dots L_1\left(\theta_1 \mid \theta_2\right)$.
- The proposal is $\pi_0\left(\theta_1\right) K_2\left(\theta_2 \mid \theta_1\right) \dots K_t\left(\theta_t \mid \theta_{t-1}\right)$.

IS on path space

- This is an importance sampler on (potentially) a very high-dimensional space:
    - each particle is actually a representation of the entire path that the particle has taken through the steps of the method,
    - we have a fixed number of particles, and we are trying to represent a space of increasing size,
    - we cannot hope to have a good representation of such a high-dimensional space,
    - it will be a disaster!
- What can we do about this?
- Idea:
    - although we are performing IS on the path space, we only need to have a good representation of the marginal distribution of θ_t.

Resampling to the rescue

- The idea is to resample from the population of particles according to their weights:
    - suppose we have N particles,
    - sample N times from a multinomial distribution with N states,
    - this gives the indices of particles we will keep in our resampled population of particles.
- Some particles will die, and we will get duplicates of others.
- Assign all resampled particles a weight of $1/N$.
- Negative effects:
    - we become degenerate (have only one particle representing) states early in the path (although this doesn't matter, since we no longer care about the marginal distribution at these states),
    - the variance of estimates based on our resampled particles will be more than before we did resampling.

Resampling to the rescue

- Positive effect:
  - we concentrate our particles on the regions of mass of π_t,
  - these particles will provide much better proposals for π_{t+1}.
- This turns out to be crucial!
  - the introduction of the resampling step was the key idea in the original particle filter of Gordon et al. (1993).

SMC review

- We explore the target using a population of particles, a sequence of distributions and kernels that move us around the space.
- Using a population of particles has something in common with using multiple MCMC chains.
- Using a sequence of distributions reduces the responsibility of choosing a good importance sampling proposal.
- The kernels can potential use local moves, which allow us to scale to higher dimensions than importance sampling.
- A major advantage is that it is relatively easy to automatically adapt the algorithm as it is running:
    - the sequence of distributions;
    - parameters of the kernels (including the scale of proposals).

Part 5— Section 16

Gradient-based methods

The Metropolis-Adjusted Langevin Algorithm

- Based on the Langevin diffusion:

$$d\mathbf{X}_t = -\frac{1}{2}\nabla \log(f(\mathbf{X}_t))dt + d\mathbf{B}_t$$

  which is f-invariant *in continuous time*.

- Given target f the MALA proposal mechanism samples:

$$\mathbf{X} \leftarrow \mathbf{X}^{(t-1)} + \epsilon$$
$$\epsilon \sim \mathrm{N}\left(-\frac{\sigma^2}{2}\nabla \log f(\mathbf{X}^{(t-1)}), \sigma^2 I_p\right)$$

  at time t.

- Accepts X with the usual MH acceptance probability.

The Metropolis-Adjusted Langevin Algorithm

- Based on the Langevin diffusion:

$$d\mathbf{X}_t = \frac{1}{2}\nabla \log(f(\mathbf{X}_t))dt + d\mathbf{B}_t$$

  which is f-invariant *in continuous time*.

- Given target f the MALA proposal proposes:

$$\mathbf{X} \leftarrow \mathbf{X}^{(t-1)} + \epsilon$$
$$\epsilon \sim \mathrm{N}\left(\frac{\sigma^2}{2}\nabla \log f(\mathbf{X}^{(t-1)}), \sigma^2 I_p\right)$$

  at time t.

- Accepts X with the usual MH acceptance probability.
- Optimal acceptance rate (under similar strong conditions) now 0.574.

MALA Example: Normal (1)

Target $f(x) = \mathrm{N}(0,1)$

Proposal

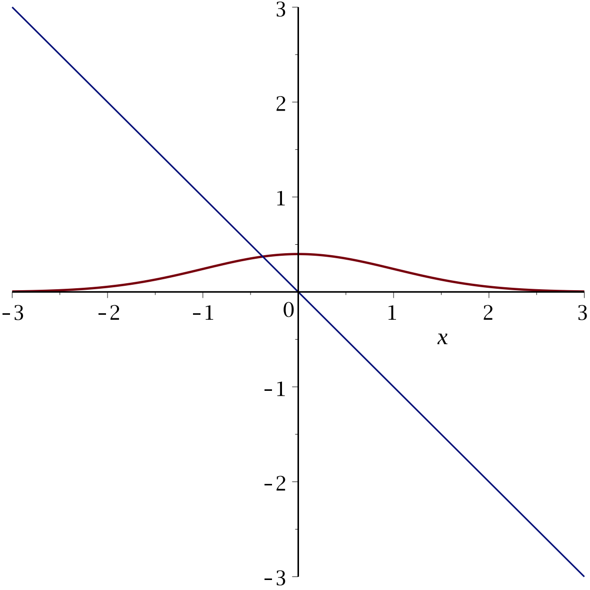
$$q(X^{(t-1)}, X) = \mathrm{N}\left(X^{(t-1)} - \frac{\sigma^2 X^{(t-1)}}{2}, \sigma^2\right)$$

Acceptance Probability

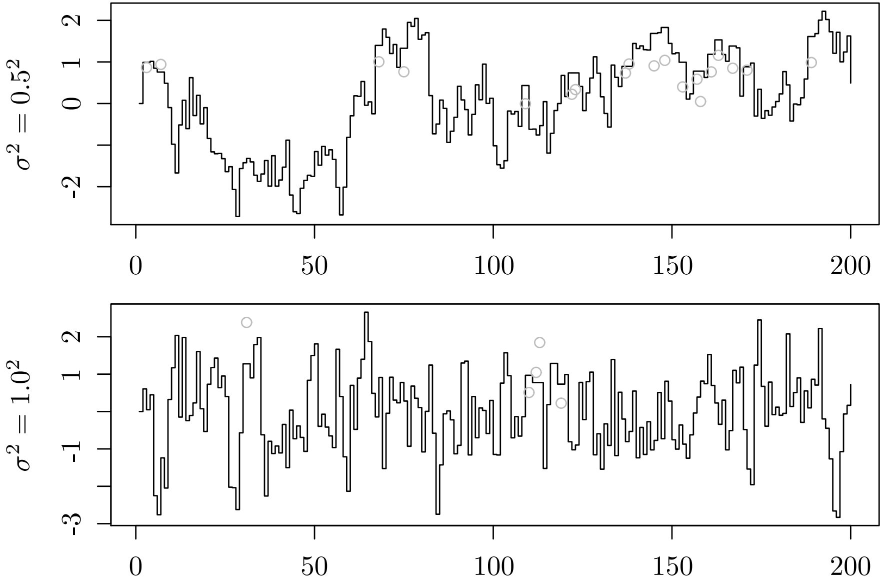
$$\begin{aligned}\alpha(X^{(t-1)}, X) &= 1 \wedge \frac{f(X)}{f(X^{(t-1)})}\frac{q(X, X^{(t-1)})}{q(X^{(t-1)}, X)}\\ &= 1 \wedge \exp\left(\frac{1}{2}\left[(X^{(t-1)})^2 - X^2\right]\right) \times\\ &\exp\left(\frac{1}{2\sigma^2}\left[\left\{X - \mu(X^{(t-1)})\right\}^2 - \left\{X^{(t-1)} - \mu(X)\right\}^2\right]\right)\end{aligned}$$

where $\mu(x) := x - \frac{x\sigma^2}{2}$.

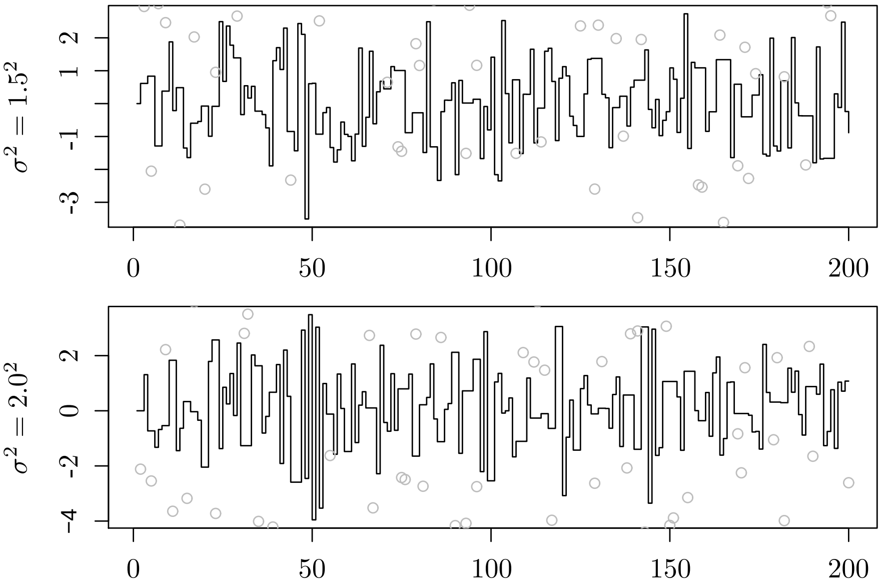
MALA



MALA


$\sigma^2 = 0.5^2$
2
1
0
-2
0
50
100
150
200
$\sigma^2 = 1.0^2$
2
1
-1
-3
0
50
100
150
200

MALA


$\sigma^2 = 1.5^2$
2
1
-1
-3
0
50
100
150
200
$\sigma^2 = 2.0^2$
2
0
-2
−4
0
50
100
150
200

MALA Example: Normal (2)

RWM	Autocorrelation $\rho(X^{(t-1)}, X^{(t)})$	Probability of acceptance $\alpha(X, X^{(t-1)})$	ESJD
$\sigma^2 = 0.1^2$	0.9901	0.9694	0.010
$\sigma^2 = 1$	0.7733	0.7038	0.448
$\sigma^2 = 2.38^2$	0.6225	0.4426	0.742
$\sigma^2 = 10^2$	0.8360	0.1255	0.337

MALA	Autocorrelation $\rho(X^{(t-1)}, X^{(t)})$	Probability of acceptance $\alpha(X, X^{(t-1)})$	ESJD
$\sigma^2 = 0.5^2$	0.898	0.877	0.246
$\sigma^2 = 1$	0.492	0.961	1.293
$\sigma^2 = 1.5^2$	0.047	0.774	2.137
$\sigma^2 = 2.0^2$	0.011	0.631	4.119

Scaling with dimension

- The number of iterations we must run the following algorithms to obtain one effectively independent point is, as a function of the size of the parameter space d:
  - $O(d)$ for random walk Metropolis-Hastings, which gives an overall computational cost of $O\left(d^2\right)$;
  - $O\left(d^{1/3}\right)$ for the Metropolis-adjusted Langevin algorithm, which gives an overall computational cost of $O\left(d^{4/3}\right)$.

Hamiltonian / Hybrid Monte Carlo

- Mimics a conservative physical system by introducing momentum.
- Approximate continuous measure-preserving flow using (symplectic) numerical integration.
- Use Metropolis–Hastings accept/reject correction.
- Can mix *much* faster than random walk algorithms.
- Difficulties with multi-modal targets and can be expensive.

c.f. Neal (2011) MCMC using Hamiltonian dynamics. In Brooks et al., 113–162. [Brooks, Gelman, Jones, and Meng (eds.) (2011) Handbook of Markov Chain Monte Carlo. CRC Press.]

Constructing a proposal: dynamics of a ball

- For random walk, we found that we needed to decrease the proposal variance as the dimension increased.
- We would like to have proposals that move a long way, but still have a good probability of acceptance
  - we need a proposal that follows the mass of the distribution.
- Think of the negative log of the target distribution, and consider the idea of setting a ball rolling around this surface
  - someone with a background in physics could describe the dynamics of this ball.
- Idea:
  - give the ball a push in a random direction
  - follow the dynamics of the ball for a while
  - use this as the proposal.

Hamiltonian dynamics

- Hamiltonian mechanics is an abstract formulation of classical mechanics (i.e. equations of motion, etc).
- It describes a system involving two time-evolving vectors θ and v, each of dimension d.
- The “Hamiltonian” $H(\theta, v)$ describes the time evolution of the system, through Hamilton’s equations

  $$\frac{\mathrm{d}\theta_i}{\mathrm{d}t} = \frac{\partial H}{\partial v_i} \qquad \frac{\mathrm{d}v_i}{\mathrm{d}t} = -\frac{\partial H}{\partial \theta_i}$$

  for $i = 1, \ldots, d$.
- Note that physicists would be very annoyed by the notation here, where the vectors are called q and p instead of θ and v.
- This is very abstract
  - what do these equations mean?

Hamiltonian dynamics: total energy

- In the use of this technique in MCMC, we use these dynamics to describe a frictionless ball rolling around the negative log of the posterior distribution, subject to a gravitational pull.
- The vector θ denotes the position of the ball, and the vector v its momentum
  - recall that momentum is equal to mass times velocity
  - for simplicity we will take the mass of the ball to be 1, which means that momentum equals velocity.
- $H(\theta, v)$ represents the total energy of the ball

$$\underbrace{H(\theta, v)}_{\text{total energy}} = \underbrace{U(\theta)}_{\text{potential energy}} + \underbrace{K(v)}_{\text{kinetic energy}} .$$

Hamiltonian dynamics: potential energy

- Recall from classical mechanics that gravitational potential energy U is equal to mgh, where m is the mass of the ball, g is the gravitational field, and h is the height.
- For simplicity, we simply set m and g to be equal to 1.
- Therefore we simply take $U(\theta)$ to be the height of the ball at θ
$$U(\theta) = -\log(\pi(\theta \mid y)).$$
- For example, $U(\theta) = \theta^2$ would correspond to a Gaussian with zero mean.

Hamiltonian dynamics: kinetic energy

- Recall from classical mechanics that kinetic energy K is equal to a half times mass times velocity squared.
- In our case (with $m = 1$, momentum equals velocity). We obtain, in the univariate case, $K = v^2/2$.
- We are looking at the multivariate case, which gives $K(v) = v^T v/2$.

Hamiltonian dynamics: Hamiltonian

- The Hamiltonian in our case is given by

$$H(\theta, v) = -\log(\pi(\theta \mid y)) + v^T v/2.$$

- Hamilton's equations in our case are given by

$$\frac{\mathrm{d}\theta}{\mathrm{d}t} = v \quad \text{and} \quad \frac{\mathrm{d}v}{\mathrm{d}t} = \nabla \log(\pi(\theta \mid y)).$$

- These make sense!
  - the rate of change of position is given by the velocity
  - the rate of change of velocity is given by the gradient of the surface.
- To construct a proposal for use in MCMC, we will simply simulate forwards from these dynamics for some time t
  - this simulation defines a deterministic function R_t, mapping $(\theta, v) \mapsto (\theta^*, v^*)$.

Hamiltonian dynamics: properties

- What did we gain from the abstract formulation, rather than simply working out this formulation from classical mechanics?
- Hamiltonian dynamics has some nice mathematical properties, that are particularly useful when constructing MCMC updates (here we follow Neal (2011)).
- **Reversibility.** There is an inverse to R_t, and this can be defined in terms of R_t. We have that R_t^{-1} is given by
  - taking the negative of the velocity (to make the ball go backwards)
  - applying R_t (running the dynamics for time t)
  - taking the negative of the velocity of the result (to make the ball "face" back in the direction it was originally)
  - we need this property for the dynamics to have π as the invariant distribution.

Hamiltonian dynamics: properties

- **Conservation of the Hamiltonian.** The dynamics do not change the value of H - the total energy of the ball is conserved.
    - this property is crucial in ensuring that the acceptance probability is high
    - soon we will define the a joint distribution of θ and v in terms of H - the conservation of H under the dynamics will mean that (θ, v) has the same density as (θ^*, v^*).
- **Volume preservation.** Hamiltonian dynamics preserves volume in the space of (θ, v). This means that no Jacobian is needed when calculating the acceptance probability of a move (as it is in some other methods).

Hamiltonian Monte Carlo

- We now have most of the ingredients needed to define Hamiltonian Monte Carlo.
- We proceed as follows
  - define a joint distribution on (θ, v) such that we can run Hamiltonian dynamics on it in order to obtain points from π
  - describe how to deal with the fact that we cannot simulate Hamiltonian dynamics exactly.

Hamiltonian Monte Carlo: joint distribution

- Define a joint distribution on (θ, v) as follows

$$\begin{aligned}\pi_{\theta,v}(\theta, v) &\propto \exp(-H(\theta, v)) \\ &= \exp(-U(\theta))\exp(-K(v)) \\ &= \exp(-(-\log(\pi(\theta \mid y))))\exp\left(-v^T v/2\right) \\ &= \pi(\theta \mid y)\exp\left(-v^T v/2\right).\end{aligned}$$

- We see that the joint distribution on (θ, v) has $\pi(\theta \mid y)$ as its marginal, and that we have a Gaussian distribution on v
    - we could choose a different covariance for this Gaussian distribution on v - this would correspond to using a different mass for the ball in the potential energy.

Using Hamiltonian dynamics as an MCMC move

- “A Note On Metropolis-Hastings Kernels For General State Spaces”, Tierney (1998) gives the Metropolis-Hastings acceptance probability for a volume preserving deterministic move T that is an involution, i.e. where, in our case, $T\left(T\left(\theta, v\right)\right) = \left(\theta, v\right)$. The acceptance probability is given by $\min\left\{1, \frac{\pi(T(\theta,v))}{\pi(\theta,v)}\right\}$.
- We define T to be the composition of applying Hamiltonian dynamics $R_t\left(\theta, v\right)$, then taking the negative of the velocity component.

Using Hamiltonian dynamics as an MCMC move

- Then, using the conservation of the Hamiltonian, the acceptance probability of applying Hamiltonian dynamics to the joint target is given by $\min\left\{1, \frac{\pi_{\theta,v}(T(\theta,v))}{\pi_{\theta,v}(\theta,v)}\right\} = \min\left\{1, \frac{\exp(-H(T(\theta,v)))}{\exp(-H(\theta,v))}\right\} = 1$, which means that we would always accept such a move!
- Potentially make very large moves, as long as we choose appropriately the time for which we simulate the dynamics
  - too short, and we will not move far
  - too long, and it is possible that we end up where we started!
- Alternate the dynamics with simulating a new velocity exactly from the target distribution for v, so that we change the direction of the trajectories at different iterations.

Approximating Hamiltonian dynamics

- We cannot simulate Hamiltonian dynamics exactly
  - we must use some solver, just as we did for the Langevin method.
- We use the "leapfrog" method to approximately simulate the dynamics
  - this produces a discretized trajectory that approximates the continuous dynamics
  - the transformation produced using this approach is also reversible and volume preserving.

Approximating Hamiltonian dynamics

- However, the Hamiltonian is not exactly conserved.
  - This means that the acceptance probability is not 1.
- Let T be the transformation given by the leapfrog method, and $(\theta^*, v^*) = T(\theta, v)$. Then, the acceptance probability is

$$\min\left\{1, \frac{\pi(T(\theta, v))}{\pi(\theta, v)}\right\} = \min\{1, \exp(-H(\theta^*, v^*) + H(\theta, v))\},$$

  - Note that, as in standard Metropolis-Hastings, we can use $p(\theta)\, l(y \mid \theta)$ in place of $\pi(\theta \mid y)$, since the normalizing constant $p(y)$ cancels.
- When implementing the leapfrog method, we need $\nabla \log(\pi(\theta \mid y))$. This is given by $\nabla \log p(\theta_t) + \nabla \log l(y \mid \theta_t)$ as in the previous lecture.

HMC properties

- Dependence on dimension
  - the optimal τ is proportional to $d^{1/4}$
  - $O\left(d^{1/4}\right)$ steps are needed to reach a nearly independent point
  - overall cost is $O\left(d^{5/4}\right)$
  - this beats both random walk and MALA.
- The tuning of HMC makes a big difference to the performance
  - much research is devoted to automating this tuning
  - the "no u-turn sampler" (NUTS), implemented in Stan, is a significant contribution.

HMC in action

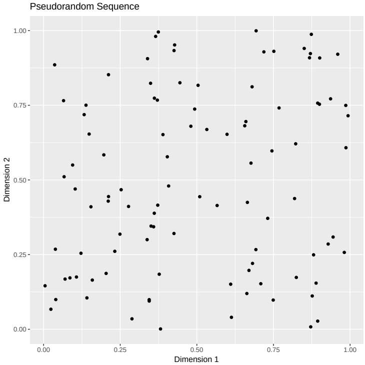
HMC in action

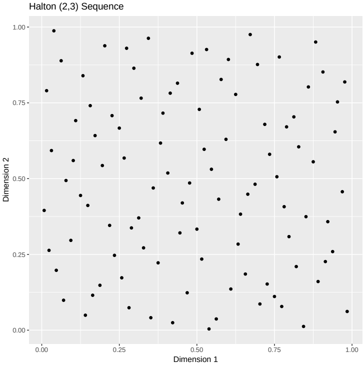
Part 5— Section 17

Other directions

Quasi Monte Carlo

- Why use “random” numbers?
- Wouldn’t “regular” numbers be better?





Low Discrepancy Sequences

Definition (Discrepancy)

Given $P = \{x_1, \dots, x_N\} \subset [0,1]^d$, *the discrepancy and star discrepancy are:*

$$D_N(P) = \sup_{J \in \mathcal{J}} \left| \frac{|P \cap J|}{N} - \lambda(J) \right|$$

$$D_N^\star(P) = \sup_{J \in \mathcal{J}^\star} \left| \frac{|P \cap J|}{N} - \lambda(J) \right|$$

where $\mathcal{J}$ *are sets of the form* $\prod_{i=1}^d [a_i, b_i)$ *and* $\mathcal{J}^\star$ *are* $\prod_{i=1}^d [0, b_i)$.

- QMC: why not approximate integrals with low discrepancy (not random) sequences?
- The *Koksma-Hlawka Inequality* controls approximation error.

Quasi Monte Carlo

Advantages

- Can (dramatically) beat Monte Carlo's $\sqrt{n}$-convergence rate.
- Reduces dependency on random numbers.

Challenges

- Constructing minimum discrepancy sequences.
- Sequence extensibility.
- Transformations (& preserving discrepancy)

c.f. Niederreiter, H. (1992) Random Number Generation and Quasi-Monte Carlo Methods. Society for Industrial and Applied Mathematics.

Dealing with Big Data

- Distribution: sub-posteriors; consensus methods; medians of medians.
- Subsampling: unadjusted Langevin; zig-zag & bouncy particle samplers. Give rise to non-reversible MCMC algorithms that rely heavily on tractable properties of *piecewise deterministic Markov processes*.
- A whole lot of computer science.

c.f. Bardenet, Doucet and Holmes (2017). On Markov chain Monte Carlo methods for tall data. Journal of Machine Learning Research 18:1–43;
Fearnhead et al. (2018). Piecewise deterministic Markov processes for continuous-time Monte Carlo. Statistical Science 33(3): 386–412.

Thank you!