BLACKBOARD AND CHALK LECTURE NOTES FOR MONTE CARLO METHODS AND APPLICATIONS

GAUTAM IYER

Contents

E-mail address: `gautam@math.cmu.edu`.

Date: Fall, 2024.

1. **Preface.**

These are the notes I used while teaching an undergraduate course on *Monte Carlo Methods and Applications* at Carnegie Mellon University in Fall 2024. These notes only list statements of important results covered in lectures. Motivation, intuition, and proofs will be done on the blackboard, and will not be on these notes.

More information can be found on the class website: `https://www.math.cmu.edu/~gautam/sj/teaching/2024-25/387-montecarlo`.

2. **What is a "Monte Carlo Method", and why is it useful?**

A *Monte Carlo method* is an algorithm that obtains a numerical approximation using repeated random trials. This was originally proposed by *Stanislaw Ulam*, inspired by his uncles gambling habits in the Monte Carlo casino in Monaco.

Example 2.1. The *mean* of a random variable can be estimated by taking an average of independent trials:

$$\boldsymbol{E}X = \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} X_i \,,$$

where $X_1, \dots, X_N$ are N-independent copies of the random variable X. (This follows from the law of large numbers.)

This is useful in practice if the random variable is easy to simulate; but hard to compute analytically.

Example 2.2 (Numerical integration). Let $\Omega \subseteq \mathbb{R}^d$, and $f\colon \Omega \to \mathbb{R}$ be an integrable function, and $X_1, \dots X_N$ be i.i.d. random variables with common distribution $\operatorname{Unif}(\Omega)$. Then

$$\lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} f(X_n) = \int_\Omega f(x)\,dx \,. \tag{2.1}$$

This is a corollary of the law of large numbers.

The advantage of (2.1) is that the error is of order

$$\sqrt{\frac{\operatorname{Var}(f)}{N}}$$

independent of the dimension d. On the other hand, if you use a standard quadrature algorithm the error is of order

$$\frac{\sqrt{d}\max|\nabla f|}{2N^{1/d}} \,.$$

This makes the computational cost of quadrature *exponential* in the dimension d, and is known as the *curse of dimensionality*. On the other hand, the computational cost of Monte Carlo integration is *independent* of the dimension. (See Section 4.)

Example 2.3 (Travelling Salesman). Given N points on the plane (cities), the *travelling salesman problem* is to find a route that travells through each city exactly once, and returns to the starting point. This is a "classic" problem which is known to be NP-hard, and you can read more about it on Wikipedia

This has been extensively studied, and there are several well known combinatorial algorithms that yield results close to the optimal path in practical amounts of time.

We will numerically approximate the solution using an algorithm known as *simulated annealing* (see Section 7.1).

Example 2.4 (Substitution Ciphers). A *substitution cipher* is one where you create a *key* that is a permutation of the alphabet (e.g. $A \mapsto K$, $B \mapsto Z$, etc.). Using this key, you can encode and decode a message. At first sight this might seem uncrackable by brute force – your key is one permutation of 28! (26 letters plus a period and space punctuation).

This is a needle in an enormous haystack. If you could examine 10^{12} keys in a second (which is a generous overestimate), then it would still take you about *a billion years* to crack this code. Nevertheless, if you're sending sufficiently long (few paragraphs) of readable text data, this method is crackable in seconds using simulated annealing (see Section 7.2).

Example 2.5 (Generative Modelling). Generative modelling algorithms are used to (for instance) produce realistic images from user input. They work by training a neural net on a large sample of images and learning a probability distribution associated with these images. Images are generated by using a Monte Carlo method to sample from this distribution.

2.1. Plan of this course.

(1) In order to use Monte Carlo methods, you need to be able to sample from a given distribution. We will start with a quick introduction to basic sampling algorithms.
(2) We will then study the *Metropolis Hastings* algorithm; to understand why this works, we need to understand the basics of the convergence of *Markov Chains* to their stationary distribution.
(3) We will analyze a few applications of the Metropolis Hastings algorithm, and study commonly used numerical diagnostics.
(4) We will study *simulated annealing* and use it to solve a few optimization problems.
(5) Time permitting, I might sketch the algorithms used in generative modelling. To fully understand these we need to understand basics of SDEs and Langevin Monte Carlo and related sampling algorithms (Section 6), which we will not have the time for. We will black-box the required background and obtain some intuition about how generative modelling works.

We will implement many of these algorithms in Python.

3. Basic Sampling Algorithms.

3.1. Uniform sampling.

Our goal is now to build a collection of distributions we can effectively sample from.

Question 3.1. *Suppose you have a random bit generator that returns either* 0 *or* 1 *with probability* 1/2*, and is independent of all previous results.*

(1) How do you generate a uniformly random number $N \in \{0, \dots, 2^N - 1\}$ *(i.e. how do you sample from the uniform distribution on* $\{0, \dots, 2^N - 1\}$
(2) How do you sample from the uniform distribution on $\{0, \dots, M\}$*, where* M *is not necessarily a power of* 2*?*
(3) How do you sample from the uniform distribution on $[0, 1]$*?*

3.2. **Transformation Methods.** The idea behind transformation methods is to start with a random variable X you can effectively simulate, and find a transformation T so that $T(X)$ follows your desired distribution.

Lemma 3.2. *Suppose $\Omega \subseteq \mathbb{R}^d$, and $T : \Omega \to \mathbb{R}^d$ is some C^1, injective transformation for which with $\det DT$ never vanishes. If X is a Ω-valued random variable with probability density function p, then $T(X)$ is an $\mathbb{R}^d$ valued random variable with probability density function q, where*

$$q = p \circ T^{-1} |\det DT^{-1}| \,.$$

Remark 3.3. Here DT is the Jacobian matrix of T.

Remark 3.4. Recall, we say p is the probability density function of a $\mathbb{R}^d$ valued random variable X if for every (nice) set $A \subseteq \mathbb{R}^d$, we have

$$\boldsymbol{P}(X \in A) = \int_A p(x)\, dx \,.$$

When $d = 1$ the above is a 1-dimensional Riemann integral, when $d = 2$, the above is a area integral, and when $d = 3$ the above is a volume integral. In dimensions higher than 3 this is a Lebesgue integral, and can be thought of as d iterated integrals. No matter what the dimension is, we will always only use one integral sign, and never write $\iint$ or $\iiint$.

Proof. Done in class. □

Proposition 3.5 (Box Mueller). *Suppose $U = (U_1, U_2)$ is uniformly distributed on $(0,1)^2$. Set*

$$Z_1 = \sqrt{-2 \ln U_1} \cos(2\pi U_2) \,, \qquad \textit{and} \qquad Z_2 = \sqrt{-2 \ln U_1} \sin(2\pi U_2) \,.$$

Then $Z = (Z_1, Z_2)$ is a standard two dimensional normal.

Proposition 3.6 (Inversion method). *Let F be the CDF of a PDF p. If $U \sim \operatorname{Unif}([0,1])$, then $F^{-1}(U)$ is a random variable with PDF p.*

Proof. Follows from Lemma 3.2. □

Remark 3.7. If p is a PDF for which F^{-1} can be computed easily, then the inversion method is a very efficient method of sampling from p.

Example 3.8. If $X \sim \operatorname{Exp}(\lambda)$, then

$$F_X(x) = \boldsymbol{P}(X \leqslant x) = 1 - e^{-\lambda x} \qquad F_X^{-1}(x) = \frac{-\ln(1-x)}{\lambda}$$

and so $-\ln(1-U)/\lambda \sim \operatorname{Exp}(\lambda)$

Proposition 3.9 (Knothe–Rosenblatt rearrangement). *Let $X = (X_1, X_2)$ be a $\mathbb{R}^2$ valued random variable with PDF p. Let p_1, F_1 be the PDF and CDF of the first marginal X_1. Explicitly,*

$$p_1(x_1) = \int_{\mathbb{R}} p(x_1, x_2)\, dx_2 \qquad \textit{and} \qquad F_1(x_1) = \boldsymbol{P}(X_1 \leqslant x_1) \,.$$

Let F_{2,x_1} be the CDF of X_2 conditioned on $X_1 = x_1$. That is,

$$F_{2,x_1}(x_2) = \text{“}\boldsymbol{P}(X_2 \leqslant x_2 \mid X_1 = x_1)\text{”} = \frac{1}{p_1(x_1)} \int_{-\infty}^{x_2} p(x_1, x_2)\, dx_2 \,.$$

Define the transformation $T\colon (0,1)^2 \to \mathbb{R}^2$ *by*

$$T(x_1, x_2) = (F_1^{-1}(x_1), F_{2,x_1}^{-1}(x_2))\,.$$

If $U = (U_1, U_2)$ *is uniformly distributed random variable on* $(0,1)^2$*, then the PDF of* $T(U)$ *is* p*.*

Remark 3.10. The notation F_{2,x_1}^{-1}, denotes the inverse of the function $y \mapsto F_{2,x_1}(y)$ for a fixed x_1. We also implicitly assume $p_1 \neq 0$, otherwise we restrict the domain accordingly.

Remark 3.11. The Knothe–Rosenblatt rearrangement can be used to efficiently sample from two dimensional distributions, *provided* the inverse CDFs F_1^{-1}, F_{2,x_1}^{-1} can be computed.

Remark 3.12. This can easily be generalized to higher dimensions by setting

$$T(x_1, x_2) = (F_1^{-1}(x_1), F_{2,x_1}^{-1}(x_2), F_{3,x_1,x_2}^{-1}(x_3), \dots)\,.$$

3.3. **Rejection sampling.** Suppose we can draw independent samples from a *proposal distribution* with density p, and the uniform distribution, then we can sample from any *target distribution* q as long as $\max q/p < \infty$.

Algorithm 1 Rejection sampling

```
Require: Proposal PDF $p$, target PDF $q$, with $M = \max q/p < \infty$.
  repeat
      Choose independent $X \sim p$, $U \sim \mathrm{Unif}([0,1])$
  until $U < q(X)/(Mp(X))$.
  return $X$
```

The algorithm is easy to understand with a picture: Let $M = \max q/p$. Simulate $X_1, \dots, X_N$ independently from the distribution p. Simulate $U_1, \dots, U_N$ independently from $\mathrm{Unif}([0,1])$. Plot the points $(X_i, U_i M p(X_i))$. Throw away all the points that are above the graph of q. Two examples of this are shown in Figure 1

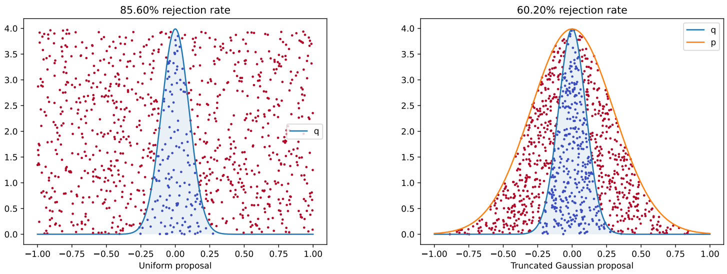

FIGURE 1. Rejection sampling of a truncated Gaussian target q. The points above the graph are rejected.

Proposition 3.13. *Let* X_n *be i.i.d. random variables whose common distribution has density* p*. Let* U_n *be independent,* $\mathrm{Unif}([0,1])$ *distributed random variables. Define*

$$N = \min\Big\{ n \,\Big|\, U_n \leqslant \frac{q(X_n)}{Mp(X_n)} \Big\} \quad \textit{where} \quad M = \max_x \Big\{ \frac{q(x)}{p(x)} \Big\}\,,$$

and set $Y = X_N$. Then the PDF of Y is q.

Proof. Done in class. □

Lemma 3.14. *If N is as in Proposition 3.13, then $\boldsymbol{P}(N = 1) = 1/M$ and $\boldsymbol{E}N = M$. (In other words, to produce one sample from q, you have to draw on average M samples from p.)*

Remark 3.15 (Curse of dimensionality)*.* The computational cost of rejection sampling typically grows exponentially with the dimension. That is if p, q are d-dimensional distributions, you can in general expect $\max q/p$ to be exponentially large in d. A simple illustration is if we try to rejection sample the uniform distribution on the unit ball $B(0,1) \subseteq \mathbb{R}^d$, starting from the uniform distribution on the unit cube $[-1,1]^d$. In this case

$$M = \max \frac{q}{p} = \frac{2^d}{\operatorname{vol}(B(0,1))} = \frac{2^d \Gamma(1 + n/2)}{\pi^{d/2}} \approx \sqrt{d\pi}\Big(\frac{2d}{\pi e}\Big)^{d/2},$$

which grows exponentially with d.

Remark 3.16*.* Rejection sampling can almost always be used; but before using it try and estimate M. If it's too large, rejection sampling might not work in practical amounts of time.

4. Monte Carlo Integration.

4.1. How expensive is quadrature?

Proposition 4.1. *Let $f\colon [0,1]^d \to \mathbb{R}$ be a C^1 function. Divide $[0,1]^d$ into N identical cubes $Q_1, \dots, Q_N$, and let ξ_i denote the center of the i-th cube. Then,*

$$\Big| \int_{[0,1]^d} f(x)\,dx - \sum_{i=1}^{N} f(\xi_i)\operatorname{vol}(Q_i) \Big| \leqslant \frac{\sqrt{d}\max|\nabla f|}{2N^{1/d}}$$

Remark 4.2*.* In order to approximate the integral of f to order ε, you need roughly $N = O(1/\varepsilon^d)$ cubes. For $\varepsilon = 0.01$ and $d = 10$, this is $O(10^{20})$ cubes. If you can examine about 10^{12} a second (a generous overestimate for my computer), it will take you a few years to use quadrature to compute this integral to two decimal places.

4.2. Monte Carlo Integration.

Theorem 4.3. *Let X_n be $\mathbb{R}^d$ valued, i.i.d. random variables with common probability density function p. Let $f\colon \mathbb{R}^d \to \mathbb{R}$ be a function such that $\int_{\mathbb{R}^d} |f(x)| p(x)\,dx < \infty$. Then*

$$\lim_{N\to\infty} \frac{1}{N}\sum_{n=1}^{N} f(X_n) = \int_{\mathbb{R}^d} f(x)\,p(x)\,dx\,, \quad \textit{almost surely}.$$

If further $\int_{\mathbb{R}^d} |f(x)|^2 p(x)\,dx < \infty$, then

$$\operatorname{Var}\Big(\frac{1}{N}\sum_{n=1}^{N} f(X_n)\Big) = \frac{1}{N}\int_{\mathbb{R}^d} |f(x) - \mu|^2 p(x)\,dx\,, \quad \textit{where } \mu = \int_{\mathbb{R}^d} f(x)p(x)\,dx\,.$$

We will prove this using the *law of large numbers* (Theorem 4.6, below).

Remark 4.4. If X_n are mutually independent and uniformly distributed on $[0,1]^d$, then the above implies

$$\lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} f(X_n) = \int_{\mathbb{R}^d} f(x)\,dx\,, \quad \text{almost surely}\,.$$

By the central limit theorem and the three sigma rule

$$\boldsymbol{P}\Big(\Big| \frac{1}{N} \sum_{n=1}^{N} f(X_n) - \int_{[0,1]^d} f(x)\,dx \Big| < \frac{3}{\sqrt{N}} \Big(\int_{[0,1]^d} |f(x)|^2\,dx \Big)^{1/2} \Big) \geqslant 0.997$$

Thus if we want to attain an error of $\varepsilon > 0$ with 99.7% certainty, we need to choose

$$N = \frac{9}{\varepsilon^2} \int_{[0,1]^d} |f(x)|^2\,dx = O\Big(\frac{1}{\varepsilon^2}\Big)\,.$$

Remark 4.5. To approximate the integral of f with accuracy ε you need to:

(1) Choose $N = O(1/\varepsilon^d)$ using quadrature.
(2) Choose $N = O(1/\varepsilon^2)$ using Monte Carlo.

Use quadrature in dimension 1 (and maybe dimension 2). Use Monte Carlo in higher dimensions.

4.3. **Law of Large Numbers.** Theorem 4.3 follows immediately from the *Law of Large Numbers.*

Theorem 4.6 (Law of large numbers)**.** *Let X_n be a sequence of i.i.d. random variables with $\boldsymbol{E}|X_n| < \infty$. Then*

$$\lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} X_n = \boldsymbol{E}X_1\,. \tag{4.1}$$

This is easy to prove if we assume $\boldsymbol{E}X_1^2 < \infty$, and is usually done in every introductory probability course. We will instead prove this without assuming $\boldsymbol{E}X_1^2 < \infty$ using *characteristic functions*, in Section 4.4, below. We were, however, somewhat imprecise when stating (4.1), which involves *convergence of random variables.* This requires measure theory to treat rigorously and goes beyond the scope of this course. Here are two more precise versions of (4.1).

(1) The *weak* law of large numbers says (4.1) holds in probability. That is, for any $\varepsilon > 0$ we have

$$\lim_{N\to\infty} \boldsymbol{P}\Big(\Big| \frac{1}{N} \sum_{n=1}^{N} X_n - \boldsymbol{E}X_1 \Big| > \varepsilon \Big) = 0$$

(2) The *strong* law of large numbers says (4.1) holds almost surely. That is, for any $\varepsilon > 0$ we have

$$\boldsymbol{P}\Big(\lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} X_n = \boldsymbol{E}X_1 \Big) = 1\,.$$

Proof of Theorem 4.3. Follows immediately from Theorem 4.6 and the fact that the variance of independent random variables adds. □

4.4. **Convergence in Distribution.**

Definition 4.7. We say a sequence of random variables X_n converges to a random variable X *in distribution* if the CDF of X_n converges to the CDF of X at all points where the CDF of X is continuous.

For $\mathbb{R}^d$ valued random variables, it is better to define convergence in distribution using bounded continuous test functions instead. However, we're not going to split hairs about this as we will test convergence in distribution using Lévy's continuity theorem.

Theorem 4.8 (Levy's continuity theorem)**.** *A sequence of random variables X_n converge to X in distribution if and only if the characteristic functions of X_n (defined below) converge pointwise to the characteristic function of X. That is X_n converges to X in distribution if and only if*

$$\lim_{n \to \infty} \varphi_{X_n}(\lambda) \to \varphi_X(\lambda) \quad \textit{for every } \xi \in \mathbb{R}^d \,.$$

Proof. The proof goes beyond the scope of this course, but is in every standard measure theory based probability book. □

Definition 4.9. Let X be a $\mathbb{R}^d$ valued random variable. The *characteristic function* of X is the function $\varphi_X \colon \mathbb{R}^d \to \mathbb{C}$ defined by

$$\varphi_X(\lambda) = \boldsymbol{E} e^{i\lambda \cdot X}\,, \quad \text{where } i = \sqrt{-1}\,.$$

Proposition 4.10. *Let X be a random variable.*

(1) φ_X is continuous, and $\varphi_X(0) = 1$.
(2) If $\boldsymbol{E}|X| < \infty$, then φ is differentiable and $\nabla \varphi(0) = -i\boldsymbol{E}X$.

Proof. Proving continuity and differentiability require the dominated convergence theorem, which is beyond the scope of this course. Computing $\varphi_X(0)$ and $\nabla\varphi_X(0)$ is direct, and will be done on the board. □

Proposition 4.11. *If $c \in \mathbb{R}$ and X is a random variable then $\varphi_{cX}(\lambda) = \varphi_X(c\lambda)$.*

Proposition 4.12. *Two random variables X and Y are independent if and only if $\varphi_{(X,Y)}(\lambda, \mu) = \varphi_X(\lambda)\varphi_Y(\mu)$.*

Proof. The reverse direction requires Fourier inversion, and is beyond the scope of this course. The forward direction can be done easily. □

Proof of Theorem 4.6. Will show that the convergence (4.1) holds *in distribution* using Theorem 4.8 □

5. **Markov Chain Monte Carlo**

5.1. **A sampling problem.** Given a *large* state space $\mathcal{X}$ and a weight function $\pi_u \colon \mathcal{X} \to [0, \infty)$, how do you draw samples from $\mathcal{X}$ so that any point $x \in X$ is drawn with probability proportional to $\pi_u(x)$. This is an extremely hard problem that is relevant to many modern applications. Two main difficulties are:

(1) Converting π_u to a probability distribution by normalizing is easier said than done. The probability distribution is clearly $\pi(x) = \pi_u(x)/Z$, where $Z = \sum_{\mathcal{X}} \pi_u(x)$. However, $\mathcal{X}$ is large and computing Z is usually not tractable.

(2) Even if Z is known, need to draw points where π_u is relatively larger more frequently. Can't know where these points are without examining all points, which is not tractable.

5.1.1. *Example: The Ising model.* This arises as a model of *ferromagnetism.*

- Consider a lattice of points, each with a spin of ± 1.
- Neighboring points with equal spins represent neighboring particles whose magnetic fields align.
- Neighboring spins that align lead to an increased overall magnetic field, and a reduced energy of the system.
- Let $\Lambda \subseteq \mathbb{R}^d$ be a finite set (the lattice of particles), and $\sigma\colon \Lambda \to \{\pm 1\}$ be the spins.
- The energy of the configuration σ is to
$$H(\sigma) \stackrel{\text{def}}{=} -J \sum_{\substack{i,j \in \Lambda \\ i \sim j}} \sigma_i \sigma_j - B \sum_{i \in \Lambda} \sigma_i \,,$$
where $i \sim j$ means i and j are nearest neighbors in the lattice Λ.
- Here $J \neq 0$ is the interaction strength ($J > 0$ for ferromagnetic materials, and $J < 0$ for anti-ferromagnetic materials).
- B represents the external magnetic field. Having spins align with the external magnetic field (i.e. $\operatorname{sign}(\sigma_i) = \operatorname{sign}(B)$) reduces the total energy.
- Expect to find configuration σ with probability proportional to
$$\pi_u(\sigma) \stackrel{\text{def}}{=} e^{-\beta H(\sigma)}\,, \quad \text{where } \beta = \frac{1}{k_B T}\,,$$
T is the temperature, and k_B is the Boltzmann constant.
- To normalize π_u you have to compute the *partition function*
$$Z_\beta \stackrel{\text{def}}{=} \sum_{\sigma} \pi_u(\sigma)\,.$$
If there are 100 points in the lattice, then the above sum has 2^{100} terms which is not computationally tractable.
- If $\beta \approx 0$ then π is roughly uniform. When $\beta = O(1)$, there are $O(2^{|\Lambda|/2})$ low energy configurations that the system is typically in. There are $2^{|\Lambda|}$ configurations in total so uniform random sampling will find the typical low energy configurations with probability $2^{-|\Lambda|/2}$.

5.1.2. *Example: Pro bit model.* This arises in machine learning where you want to label a vector of features.

- Want to predict a binary label $Y \in \{0, 1\}$ given a vector of features $z \in \mathbb{R}^d$.
- Predict $\boldsymbol{P}(Y = 1 \mid z) = \Phi(z \cdot \beta)$ where:
  - Let Φ be the CDF of the standard normal.
  - $\beta \in \mathbb{R}^d$ is to be determined later
- Suppose we are given given m labelled data points $\mathcal{L} = \{(z_i, y_i) \mid i \in \{1, \dots, m\}\}$, with z_i i.i.d. with probability density function g.
- Starting from a *prior distribution* p (for β), we compute the *posterior* by
$$\pi(\beta) \stackrel{\text{def}}{=} p(\beta \mid \mathcal{L}) = \frac{p(\beta)}{\boldsymbol{P}(\mathcal{L})} \boldsymbol{P}(\mathcal{L} \mid \beta) = \frac{p(\beta)}{\boldsymbol{P}(\mathcal{L})} \prod_{i=1}^{d} g(z_i) \boldsymbol{P}(Y = y_i \mid z_i) \propto \pi_u(\beta)\,,$$

where

$$\pi_u(\beta) = p(\beta) \prod_{i=1}^{d} g(z_i) \boldsymbol{P}(Y = y_i \mid z_i) = p(\beta) \prod_{i=1}^{d} \Phi(z_i \cdot \beta)^{y_i} (1 - \Phi(z_i \cdot \beta))^{1-y_i} .$$

- Computing $\boldsymbol{P}(\mathcal{L})$ isn't easy – it's a integral in a very high dimensional space. So it's not easy to normalize π_u.
- Typically want to sample from the posterior distribution $\pi(\beta)$. For instance, if we want to choose β to be the *posterior mean*, then we can compute it by sampling N points β_1, ..., β_N from π and then using the Monte Carlo approximation $\frac{1}{N}\sum_{n=1}^{N} \beta_n$.

5.1.3. *Example: Disk packing – phase transitions.*

- Fix N large (between 100 and 10^6), and ε small.
- We want to pack N hard disks of radius ε into the unit square.
- Let $\eta = N\pi\varepsilon^2 < 1$ be the density of the disks.
- Physical motivation – phase transition (e.g. water melting).
- $\mathcal{X} = \mathcal{X}_{N,\varepsilon}$ be the set of all possible configurations where N non-overlapping disks of radius ε are packed into the unit square.
- $\mathcal{X} \subseteq [0,1]^{2N}$, and so it makes sense to draw a point "uniformly randomly" from $\mathcal{X}$.
- *Kirkwood transition.* When $\eta < 0.71\dots$ disks are randomly placed. Above this they are roughly in a hexagonal grid.

  To check this numerically you need:

  (1) An algorithm to randomly sample configurations of hard disks.
  (2) A way to test if a given configuration is roughly hexagonal.

Roughly hexagonal test. Represent a configuration by $x = (x_1, \dots, x_N) \in \mathbb{R}^{2N}$ with each $x_i \in [0,1]^2$ representing the center of the i^{th} disk. Compute

$$F(x) = \frac{1}{N} \Bigl| \sum_{j=1}^{N} \frac{1}{|\mathcal{N}_j|} \sum_{k \in \mathcal{N}_j} e^{6i\theta_{j,k}} \Bigr| ,$$

where $\mathcal{N}_j$ is the set of all disks that are adjacent and $\theta_{j,k}$ is the angle between the line joining the centers of the corresponding disks and the horizontal axis.

(1) If the disks are in a roughly hexagonal lattice, then $\theta_{j,k} \approx \alpha + n\pi/3$, where α is some fixed angle (independent of j, k) and $n \in \{0, 1, 2\}$. In this case $F(x) \approx 1$.
(2) If instead the disk arrangement is random, there will be a lot of cancellation in the sum and we will have $F(x) \ll 1$.

Random sampling algorithm. Start with all disks in a valid configuration (e.g. on a hexagonal grid, or rectangular grid). Fix a small number $\rho > 0$.

(1) Choose one disk, at random with center x_i.
(2) Choose $y \in B(x_i, \rho)$ uniformly randomly.
(3) If moving the i^{th} disks center to y is a valid configuration (i.e. doesn't overlap other disks), then do it.

Repeat these steps for as long as your computational budget allows (see Figure 2). (Note, no matter what ρ is this algorithm will eventually work; however if ρ is too small or too large the algorithm may take too long to converge. One of the questions in the homework is to choose ρ.)

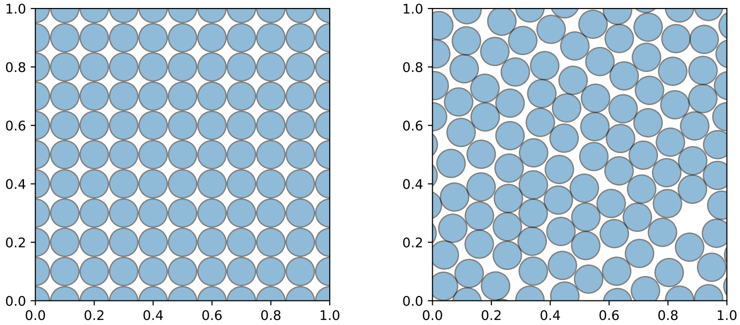

FIGURE 2. Left: 100 disks in a rectangular grid with $\eta = .72$. Right: Same disks after running the Metropolis algorithm for 10^6 iterations. Some hexagonal structure can be visibly observed. The measure of hexagonality increases from 0 to roughly 0.6 in about 500,000 iterations, after which it oscillates around this value.

This was proposed by Metropolis, two Rosenbluth's, and two Tellers. It was later generalized by Hastings resulting in the *Metropolis Hastings* algorithm, which is one of the most fundamental algorithms in sampling today.

5.2. **The Metropolis Hastings algorithm.** This algorithm tells you how to sample from an un-normalized probability distribution π_u on a (possibly very large) state space $\mathcal{X}$. It requires a *proposal mechanism* Q – that is, given $x \in \mathcal{X}$, we are able to choose $y \in \mathcal{X}$ randomly with distribution $Q(x, y)$. Examples of proposition mechanisms are:

- Given a configuration of hard disks, choose one randomly and move it's center as described in Section 5.1.3.
- Given a spin configuration $\sigma\colon \Lambda \to \{\pm 1\}$ in the Ising model, pick a site $i \in \Lambda$ uniformly randomly, and flip the spin. (I.e. choose $i \in \Lambda$ uniformly, and set $\tau(j) = \sigma(j)$ if $j \neq i$ and $\tau(i) = -\sigma(i)$.)

Given a proposal mechanism $Q(x, y)$, the *Metropolis–Hastings* algorithm is as follows:

(1) Choose $X_0 \in \mathcal{X}$ arbitrarily.
(2) Given X_n, choose $y \in \mathcal{X}$ with probability $Q(X_n, y)$ using your proposal mechanism.
(3) Define the *acceptance probability* $A(x, y)$ by

$$A(x,y) \stackrel{\text{def}}{=} \min\Big\{1, \frac{\pi_u(y)Q(y,x)}{\pi_u(x)Q(x,y)}\Big\} \tag{5.1}$$

(4) Flip a coin that lands heads with probability $A(X_n, y)$. If the coin lands heads accept the new state and set $X_{n+1} = y$. If not, reject it and set $X_{n+1} = X_n$.

While the Metropolis–Hastings algorithm is simple to explain, the reason it works is not so obvious. We will soon see that under certain assumptions (*irreducibility* and *aperiodicity*) the distribution of X_n converges to normalised probability distribution $\pi = \pi_u / Z$ (where $Z = \sum_x \pi_u(x)$) as $n \to \infty$. So if you want to sample from π, you only have to simulate X_n for some large n.

A few words of caution:

- Irreducibility and aperiodicity aren't always easy to check; many times the Metropolis–Hastings algorithm is used without checking the required conditions.
- In the continuous space setting, one needs an additional condition to ensure convergence. Many treatments don't even state these conditions. Checking them is usually hard.
- The most important practical consideration is the *rate of convergence.* Even though $\operatorname{dist}(X_n) \to \pi$ as $n \to \infty$, the convergence rate may be extremely slow. You see this in practice if your chain gets stuck – new proposals keep getting rejected.

5.3. **Markov Chains.** The Metropolis–Hastings algorithm describes a *Markov Chain* whose stationary distribution is the desired distribution π. In some situations the distribution of a Markov chain will converge to its stationary distribution, which is why the Metropolis–Hastings algorithm algorithm works. Our aim is to understand what Markov Chains are, and when they converge to their stationary distribution.

Definition 5.1. A Markov chain on $\mathcal{X}$ is a family of random variables X_0, X_1, ...such that for all $n \in \mathbb{N}$ and $x_0, \dots, x_{n+1} \in \mathcal{X}$ we have

$$\boldsymbol{P}(X_{n+1} = x_{n+1} \mid \{X_k = x_k \mid 0 \leqslant k \leqslant n\}) = \boldsymbol{P}(X_{n+1} = x_{n+1} \mid X_n = x_n)\,.$$

Definition 5.2. A *time homogeneous* Markov chain is a Markov chain where

$$\boldsymbol{P}(X_{n+1} = y \mid X_n = x) = \boldsymbol{P}(X_1 = y \mid X_0 = x)\,,$$

for all $x, y \in \mathcal{X}$ and $n \in \mathbb{N}$.

Example 5.3 (Simple random walk)*.* Let $\mathcal{X} = \mathbb{Z}$, ξ_n be i.i.d. ± 1 valued random variables (coin flips) and set $X_{n+1} = X_n + \xi_{n+1}$.

Example 5.4 (Metropolis chain)*.* Given X_n, choose X_{n+1} according to the Metropolis–Hastings rule:

(1) Given $X_n = x \in \mathcal{X}$, propose $y \in \mathcal{X}$ with probability $Q(x, y)$.
(2) Let $X_{n+1} = y$ with probability $A(x, y)$ (defined in (5.1)), and $X_{n+1} = x$ otherwise.

Definition 5.5. The *transition matrix* of a (time homogeneous) Markov chain is

$$P(x, y) = \boldsymbol{P}(X_1 = y \mid X_0 = x)\,.$$

Remark 5.6. Notice $P(x, y) \geqslant 0$ and $\sum_y P(x, y) = 1$. Such matrices are called *stochastic matrices.*

Proposition 5.7. *For any $n \in \mathbb{N}$,*

$$\boldsymbol{P}(X_n = y \mid X_0 = x) = P^n(x, y)\,,$$

where P^n means the n^{th} power of the matrix P.

Proof. Directly multiply. □

Proposition 5.8. *If* $\operatorname{dist}(X_0) = \mu_0$ *(i.e. $\boldsymbol{P}(X_0 = x) = \mu_0(x)$ for all $x \in \mathcal{X}$), then* $\operatorname{dist}(X_n) = \mu_0 P^n$ *(as a matrix product).*

Proof. Directly multiply. □

5.3.1. *Stationary distribution.*

Definition 5.9. We say distribution π is *stationary* for the Markov chain X if $\pi P = \pi$. That is if $X_0 \sim \pi$ then $X_n \sim \pi$ for all $n \in \mathbb{N}$.

Theorem 5.10. *If $|\mathcal{X}| < \infty$, then any Markov chain has a stationary distribution.*

Proof. Frobenius theorem. For a direct probabilistic proof one can construct π by picking $x_0 \in \mathcal{X}$ (arbitrarily), and letting $\pi(y)$ to be the average number of visits to y before returning to x_0. See Proposition 1.14 in [LP17]. □

Remark 5.11. The stationary distribution need not be unique. For example $\mathcal{X} = \{0, 1\}$, and $P = I$ has infinitely many stationary distributions.

Definition 5.12. A Markov chain is called *irreducible* if for any $x, y \in \mathcal{X}$ there exists $n \in \mathbb{N}$ such that $P^n(x, y) > 0$.

Theorem 5.13. *If P is irreducible, the stationary distribution is unique.*

Lemma 5.14. *If P is irreducible, and π is a stationary distribution, then $\pi(x) > 0$ for all $x \in \mathcal{X}$.*

Proof of Theorem 5.13. If π_1 and π_2 are two stationary distributions, choose x_0 that minimizes $\pi_1(x)/\pi_2(x)$. Clearly

$$\pi_1(x_0) = \sum_{x \in \mathcal{X}} \frac{\pi_1(x)}{\pi_2(x)} \pi_2(x) P(x, x_0) \geqslant \frac{\pi_1(x_0)}{\pi_2(x_0)} \pi_2(x_0) = \pi_1(x_0)\,.$$

This implies

$$\sum_{x \in \mathcal{X}} \frac{\pi_1(x)}{\pi_2(x)} \pi_2(x) P(x, x_0) = \sum_{x \in \mathcal{X}} \frac{\pi_1(x_0)}{\pi_2(x_0)} \pi_2(x) P(x, x_0)\,.$$

Since $\pi_1(x)/\pi_2(x) \geqslant \pi_1(x_0)/\pi_2(x_0)$ for all x, the above equality can only hold if

$$\frac{\pi_1(x)}{\pi_2(x)} = \frac{\pi_1(x_0)}{\pi_2(x_0)} \quad \text{for every } x \in \mathcal{X} \text{ such that } P(x, x_0) > 0\,.$$

for all x such that $P(x, x_0) > 0$. Now for any given $x \in \mathcal{X}$, irreducibility implies we can find $N = N(x) \in \mathbb{N}$ such that $P^N(x, x_0) > 0$. Since π_1 and π_2 are also stationary for P^N the above argument will imply

$$\frac{\pi_1(x)}{\pi_2(x)} = \frac{\pi_1(x_0)}{\pi_2(x_0)} \quad \text{for every } x \in \mathcal{X}\,.$$

Since $\sum_{x \in \mathcal{X}} \pi_1(x) = \sum_{x \in \mathcal{X}} \pi_2(x) = 1$, this implies $\pi_1 = \pi_2$, finishing the proof. □

5.3.2. *Convergence to the stationary distribution.*

Example 5.15. Irreducibility alone doesn't guarantee convergence to the stationary distribution. For example the chain with transitions $P(0, 1) = P(1, 0) = 1$ and $P(1, 1) = P(0, 0) = 0$ is irreducible, but the distribution need not converge to the stationary distribution.

Definition 5.16. A Markov chain is called *aperiodic* if for all $x \in \mathcal{X}$,

$$\gcd\{n \geqslant 1 \mid P^n(x, x) > 0\} = 1\,.$$

Example 5.17. The simple random walk (Example 5.3) is irreducible, but not aperiodic. If instead ξ_n are i.i.d. random variables such that $\boldsymbol{P}(\xi_n = 0) = 1/2$ and $\boldsymbol{P}(\xi_n = \pm 1) = 1/4$, then the *lazy* random walk defined by $X_{n+1} = X_n + \xi_{n+1}$ is irreducible and aperiodic.

Example 5.18. The Markov chain with $P = I$ is aperiodic but not irreducible (if $|\mathcal{X}| > 1$).

Remark 5.19. If a Markov chain is irreducible but not aperiodic, one common trick is to introduce lazyness: Flip an independent fair coin. If it lands heads, don't move. If it lands tails, move according to the original transition kernel. That is, define a new transition matrix Q by

$$Q(x,y) = \frac{1}{2}(I + P) = \begin{cases} \frac{1}{2}(1 + P(x,x)) & y = x\,, \\ \frac{1}{2}P(x,y) & y \neq x\,. \end{cases}$$

The new chain will be both irreducible and aperiodic, and have the same stationary distribution.

Theorem 5.20. *If a Markov chain is irreducible and aperiodic, then* $\operatorname{dist}(X_n) \to \pi$ *in* total variation *as* $n \to \infty$.

Definition 5.21. For any two probability distributions μ, ν we define the *total variation* distance between μ and ν by

$$\|\mu - \nu\|_{\mathrm{TV}} \overset{\text{def}}{=} \frac{1}{2} \sum_{x \in \mathcal{X}} |\mu(x) - \nu(x)|\,.$$

Definition 5.22. We say $\operatorname{dist}(X_n) \to \pi$ in *total variation* as $n \to \infty$, if we have $\|\operatorname{dist}(X_n) - \pi\|_{\mathrm{TV}} = 0$ as $n \to \infty$.

Remark 5.23. Notice $\|\operatorname{dist}(X_n) - \pi\|_{\mathrm{TV}} = 0$ as $n \to \infty$ is equivalent to having

$$\lim_{n \to \infty} \frac{1}{2} \sum_{x \in \mathcal{X}} |\boldsymbol{P}(X_n = x) - \pi(x)| = 0\,.$$

Lemma 5.24. *If a Markov chain is aperiodic, then there exists* $N \in \mathbb{N}$ *such that* $P^n(x,x) > 0$ *for all* $x \in \mathcal{X}$ *and* $n \geqslant N$.

Lemma 5.25. *If a Markov chain is irreducible and aperiodic, there exists* $N \in \mathbb{N}$ *such that* $P^N(x,y) > 0$ *for all* $x, y \in \mathcal{X}$.

Proof of Theorem 5.20. The proof will be done on the board. A rough sketch is:

(1) Choose

$$\delta = \min_{x,y \in \mathcal{X}} \frac{P^N(x,y)}{\pi(y)}$$

which is strictly positive (by Lemma 5.25), and write

$$P^N(x,y) = \delta\pi(y) + (1 - \delta)Q(x,y)\,,$$

For some matrix Q. By choice of δ, Q is a stochastic matrix.

(2) Compute $P^{kN+j}(x,y) = (1 - (1-\delta)^k)\pi(y) + (1-\delta)^k Q^k P^j(x,y)$.

(3) Implies $|P^{kN+j}(x,y) - \pi(y)| \leqslant 2(1-\delta)^k$.

(4) Implies $\|\operatorname{dist}(X_{kN+j}) - \pi\|_{\mathrm{TV}} \leqslant 2(1-\delta)^k$. □

Finally, we mention that the law of large numbers doesn't apply to Markov chains as we typically won't have $X_1, X_2, \dots$ to be independent, or identically distributed. However, the conclusion still holds and is often called the Ergodic theorem instead of the law of large numbers.

Theorem 5.26 (Ergodic theorem)**.** *Let X_n be an irreducible Markov chain with stationary distribution π. Let $f \colon \mathcal{X} \to \mathbb{R}$ be any function. Then*

$$\boldsymbol{P}\Big(\lim_{N \to \infty} \frac{1}{N} \sum_{n=1}^{N} f(X_n) = \sum_{x \in \mathcal{X}} f(x)\pi(x)\Big) = 1\,.$$

Proof. The proof is accessible, but tricky and technical. There's a one page proof in the appendix of [LP17]. □

5.3.3. *Detailed balance.*

Definition 5.27. We say a Markov chain X satisfies the *detailed balance* condition for a distribution μ if

$$\mu(x)P(x,y) = \mu(y)P(y,x)\,. \tag{5.2}$$

Put another way, (5.2) is equivalent to the statement that if $X_0 \sim \mu$, then

$$\boldsymbol{P}(X_0 = x, X_1 = y) = \boldsymbol{P}(X_0 = y, X_1 = x)\,.$$

Proposition 5.28. *If a Markov chain satisfies the detailed balance condition for a distribution π, then π is a stationary distribution for the chain.*

Proof. By detailed balance,

$$\sum_{x \in \mathcal{X}} \pi(x)P(x,y) = \sum_{x \in \mathcal{X}} \pi(y)P(y,x) = \pi(y) \sum_{x \in \mathcal{X}} P(y,x) = \pi(y)$$ □

Remark 5.29. The converse is false. If a Markov chain has stationary distribution π, then it need not satisfy the detailed balance condition (5.2). Example 5.15 is an example where the uniform distribution is the unique stationary distribution, but the detailed balance condition is not satisfied.

Remark 5.30. If a Markov chain satisfies the detailed balance condition with the uniform distribution, then the transition matrix is symmetric and hence doubly stochastic (i.e. both row sums and column sums are 1). There are, of course, many doubly stochastic matrices that are not symmetric.

Proposition 5.31. *The stationary distribution of the Metropolis–Hastings chain (Example 5.4) is π.*

Proof. The transition matrix of the Metropolis chain is given by

$$P(x,y) = \begin{cases} Q(x,y)A(x,y) & y \neq x \\ 1 - \sum_{y' \neq x} Q(x,y')A(x,y') & y = x\,. \end{cases}$$

Pick $x, y \in \mathcal{X}$ and suppose first $y \neq x$, and $\pi(x)Q(y,x) \geqslant \pi(y)Q(x,y)$. In this case

$$A(x,y) = \frac{\pi(y)Q(y,x)}{\pi(x)Q(x,y)} \qquad \text{and} \qquad A(y,x) = 1\,,$$

and so

$$\pi(x)P(x,y) = \pi(x)Q(x,y)A(x,y) = \pi(y)Q(y,x)$$

$$= \pi(y)Q(y,x)A(y,x) = \pi(y)P(y,x)\,,$$

and so the detailed balance condition (5.2) holds in this case. By symmetry, when $\pi(x)Q(y,x) \leqslant \pi(y)Q(x,y)$ we also have (5.2). When $y = x$, the detailed balance condition (5.2) is trivially true. Thus by Proposition 5.28, π is a stationary distribution for the Metropolis–Hastings chain. □

Remark 5.32. There are other choices of the acceptance ratio for which the stationary distribution is π. One choice (Barker '65) is

$$A(x,y) = \frac{1}{1 + \frac{\pi(x)Q(x,y)}{\pi(y)Q(y,x)}} = \frac{1}{1 + \frac{\pi_u(x)Q(x,y)}{\pi_u(y)Q(y,x)}}$$

The advantage of the traditional choice (5.1) is that it minimizes the asymptotic variance

$$\lim_{N\to\infty} N \operatorname{Var}\Big(\frac{1}{N}\sum_{n=1}^{N} f(X_n)\Big)\,.$$

When applying the Metropolis–Hastings algorithm, you want to choose a proposal mechanism to ensure that the chain is *irreducible* and *aperiodic*. If these two conditions are satisfied, then Proposition 5.31 will imply that $\operatorname{dist}(X_n) \to \pi$ as $n \to \infty$. This means if you run the chain long enough, you've generated points that are sampled according to the desired target distribution π.

The practical issue is that the convergence may happen very slowly, and you may have to wait for a very long time before $\operatorname{dist}(X_n)$ is close enough to the stationary distribution. In general you want to choose your proposal mechanism in a manner that improves the rate of convergence of $\operatorname{dist}(X_n)$ to the stationary distribution. There's no silver bullet. Coming up with these mechanisms is usually problem specific, and estimating the rate of convergence for a given proposal mechanism is not easy. If you're interested in learning more, look up mixing times of Markov chains.

6. Stochastic Differential Equations.

6.1. **Motivation.** Very often we want to compute solutions of *partial differential equations* (or PDEs). We will see that for some PDEs, the solution can be written as the expected value of a *Stochastic Differential Equation* (or SDE), and then we can compute the desired solution by Monte Carlo simulation.

The other reason we study this is for sampling. Suppose we want to sample from a distribution with density $p\colon \mathbb{R}^d \to [0,\infty)$. It turns out that if we set $U = -\ln p$, let X be the solution of the SDE

$$dX_t = -\nabla U(X)\,dt + \sqrt{2}\,dW_t\,, \tag{6.1}$$

then the *stationary distribution* of X has density p (we will see why later). Thus simulating the SDE (6.1) will allow you to sample from p.

6.2. **Brownian Motion.** Brownian motion is a continuous time random walk. One way to describe this is by taking a discrete time random walk where coins are flipped every second and rescaling it so that coins are flipped every ε seconds. The limiting process we get as $\varepsilon \to 0$ is called Brownian motion. We will use W_t to denote the Brownian motion process. That is for every $t \geqslant 0$, W_t is the *random variable* that describes the location of the (continuous time) random walk after time t. One can show (using the central limit theorem) that for any $s \leqslant t$, the Brownian increment $W_t - W_s$ is normally distributed with variance proportional to $t-s$, and is *independent* of W_r for all $r \leqslant s$. To standardize notation we will normalize the proportionality constant to be 1, and define a (standard) Brownian motion as follows.

Definition 6.1. We say W is a (standard) Brownian motion if:

(1) For every $t \geqslant 0$, W_t is a random variable. Moreover, for every realization the function $t \mapsto W_t$ is always continuous as a function of t.
(2) For any $0 \leqslant s < t$, $W_t - W_s \sim N(0, t-s)$.
(3) For any $0 \leqslant r \leqslant s < t$, the random variables $W_t - W_s$ and W_r are independent.

Proposition 6.2. *For almost every realization, the function $t \mapsto W_t$ is not differentiable anywhere.*

Proof. We will prove a simpler version on the board. □

6.3. **Itô integrals.** When describing SDEs one usually writes expressions of the form

$$dX_t = b_t\,dt + \sigma_t\,dW_t\,, \tag{6.2}$$

where b and σ are *adapted processes.*

Definition 6.3. We say Y is an adapted process if for every $t \geqslant 0$, Y_t is a random variable that can be expressed in terms of $\{(s, W_s) \mid 0 \leqslant s \leqslant t\}$.

Now to make sense of (6.2) one might be tempted to say

$$X_{t+h} - X_t \approx b_t h + \sigma_t(W_t - W_h) \quad \text{when } h \text{ is small.}$$

However, this doesn't make sense as both sides vanish as $h \to 0$. In regular calculus one gets around this by dividing both sides by h and sending $h \to 0$. This won't work for us as the limit of the last term on the right wont exist (Proposition 6.2). We will, instead, make sense of (6.2) by integrating.

Definition 6.4. We say (6.2) holds if for every $T > 0$ we have

$$X_T - X_0 = \int_0^T b_s\,ds + \int_0^T \sigma_s\,dW_s\,. \tag{6.3}$$

The first term on the right of (6.3) is simply a Riemann integral (with a possibly random integrand). The second term requires more care – it's called an *Itô* integral and *is different from a Riemann integral.*

Definition 6.5. We define the *Itô integral* of σ with respect to Brownian motion by

$$\int_0^T \sigma_s\,dW_s = \lim_{\|P\| \to 0} \sum_{i=0}^{N-1} \sigma_{t_i}(W_{t_{i+1}} - W_{t_i})\,,$$

where $P = \{0 = t_0 < t_1 < \cdots < t_N = T\}$ is a partition of $[0, T]$ and $\|P\|$ (called the *mesh size* of P is the length of the largest subinterval – $\|P\| = \max_i t_{i+1} - t_i$).

Remark 6.6. For Itô integrals, it's important you sample σ at the left endpoint of the interval. That is, we had $\sigma_{t_i}(W_{t_{i+1}} - W_{t_i})$ and not $\sigma_{t_{i+1}}(W_{t_{i+1}} - W_{t_i})$ or $\sigma_{\xi_i}(W_{t_{i+1}} - W_{t_i})$ where $\xi_i = (t_{i+1} - t_i)/2$. For Riemann integrals the position you sample at doesn't matter and won't change the value of the integral. *This is not true for Itô integrals.* If you change the sample points you may change value of the integral.

Proposition 6.7. *If σ is adapted then*

(1) $\boldsymbol{E} \int_0^T \sigma_s dW_s = 0.$

(2) $\boldsymbol{E} \Big(\int_0^T \sigma_s dW_s \Big)^2 = \boldsymbol{E} \int_0^T \sigma_s^2 \, ds.$

Proof. Will be done on the board. □

Remark 6.8. For Riemann integrals we instead have

(1) $\boldsymbol{E} \int_0^T b_s \, ds = \int_0^T \boldsymbol{E} b_s \, ds.$

(2) $\boldsymbol{E} \Big(\int_0^T b_s \, ds \Big)^2 = \int_0^T \int_0^T \boldsymbol{E}(b_r b_s) \, dr \, ds.$

6.4. **Itô formula (statement).** The *chain rule* from multi-variable calculus says that if X, Y are differentiable functions of t, and f is a differentiable function of x, y, t, then

$$\frac{d}{dt}(f(t, X_t, Y_t)) = \partial_t f(t, X_t, Y_t) + \partial_x f(t, X_t, Y_t) \frac{dX_t}{dt} + \partial_y f(t, X_t, Y_t) \frac{dY_t}{dt}$$

Formally multiplying the dt on both sides gives

$$d(f(t, X_t, Y_t)) = \partial_t f(t, X_t, Y_t) \, dt + \partial_x f(t, X_t, Y_t) \, dX_t + \partial_y f(t, X_t, Y_t) \, dY_t \, .$$

The above *requires* X, Y to be differentiable functions of t. *This usually fails for stochastic processes.* The chain rule can be generalized to work for stochastic processes. This generalization is called the *Itô formula*, and *is different* from the standard chain rule. The generalization has an extra term using the *joint quadratic variation* denoted by $[X, Y]$ which will be defined subsequently.

Theorem 6.9 (Two-dimensional Itô formula).

- *Let X, Y be a two Itô process.*
- *Let $f = f(t, x, y)$ be a function that's defined for $t \in \mathbb{R}$, $x, y \in \mathbb{R}$.*
- *Suppose $f \in C^{1,2}$. That is:*
  - ▷ *f is once differentiable in t*
  - ▷ *f is twice in both x and y.*
  - ▷ *All the above partial derivatives are continuous.*

  Then:

$$d(f(t, X_t, Y_t)) = \partial_t f(t, X_t, Y_t) \, dt + \partial_x f(t, X_t, Y_t) \, dX_t + \partial_y f(t, X_t, Y_t) \, dY_t + \frac{1}{2} \Big(\partial_x^2 f(t, X_t, Y_t) \, d[X, X]_t + \partial_y^2 f(t, X_t, Y_t) \, d[Y, Y]_t + 2 \partial_x \partial_y f(t, X_t, Y_t) \, d[X, Y]_t \Big)$$

Remark 6.10. We will often drop the arguments of f and simply write

$$\begin{aligned} d(f(t, X_t, Y_t)) &= \partial_t f\, dt + \partial_x f\, dX_t + \partial_y f\, dY_t \\ &\qquad + \frac{1}{2}\Big(\partial_x^2 f\, d[X,X]_t + \partial_y^2 f\, d[Y,Y]_t + 2\partial_x\partial_y f\, d[X,Y]_t\Big) \end{aligned}$$

Remember the arguments are present. After differentiating f you should substitute $x = X_t$, $y = Y_t$.

Remark 6.11 (Integral form). The integral form of the above is

$$\begin{aligned} f(T, X_T, Y_T) - f(0, X_0, Y_0) &= \int_0^T \partial_t f\, dt + \int_0^T \partial_x f\, dX_t + \int_0^T \partial_y f\, dY_t \\ &\quad + \frac{1}{2}\Big(\int_0^T \partial_x^2 f\, d[X,X]_t + \int_0^T \partial_y^2 f\, d[Y,Y]_t + 2\int_0^T \partial_x\partial_y f\, d[X,Y]_t\Big) \end{aligned}$$

We will return to the Itô formula (and provide some intuition about why it holds) after we describe what joint quadratic variation is.

6.5. Joint quadratic variation.

- Let X and Y be two continuous adapted $\mathbb{R}$ valued processes.
- $P = \{0 = t_1 < t_1 \cdots < t_n = T\}$ is a partition of $[0, T]$.

Definition 6.12. The *joint quadratic variation* of X, Y, is defined by

$$[X, Y]_T = \lim_{\|P\|\to 0} \sum_{i=0}^{n-1} (X_{t_{i+1}} - X_{t_i})(Y_{t_{i+1}} - Y_{t_i})\,,$$

In stochastic calculus we often encounter expressions of the form

$$d[X,Y]_t = b_t\, dt\,. \tag{6.4}$$

Interpret this in the same way we interpreted the SDE (6.2) – the right hand side is the infinitesimal increment of $[X, Y]$, and when integrated gives the total increment. That is (6.4) means

$$[X,Y]_T - [X,Y]_0 = \int_0^T b_t\, dt\,.$$

(Of course, $[X, Y]_0 = 0$, so we can drop it from the left hand side above.)

Proposition 6.13. *If either X or Y is differentiable, then $[X, Y]_t = 0$.*

Remark 6.14. More generally, if either X or Y have *finite first variation* then $[X, Y]_t = 0$.

Remark 6.15. Also remember that both X and Y are already assumed to be continuous processes.

Proof. Will be done on the board. □

Proposition 6.16. *(1)* (Symmetry) $[X, Y] = [Y, X]$
(2) (Bi-linearity) *If $\alpha \in \mathbb{R}$, X, Y, Z are continuous processes,* $[X, Y + \alpha Z] = [X, Y] + \alpha[X, Z]$.

Proof. Direct check. □

Proposition 6.17. *If W is a 1D Brownian motion, then* $[W, W]_t = t$

Proof. Will be done on the board. □

Proposition 6.18. *If X and Y are independent, then $[X, Y]_t = 0$.*

Proof. The general proof requires a few basic facts about martingales. If X and Y are independent Brownian motion's then the proof is simpler, and will be done on the board. □

Proposition 6.19. *Let W^1 and W^2 be two independent Brownian motions, b^1, b^2, σ^1, σ^2 be two processes, and suppose*

$$dX^i_t = b^i_t \, dt + \sigma^i_t \, dW^i_t \, .$$

Then

$$d[X^i, Y^i]_t = \sigma^i_t \, \sigma^j_t \, d[W^i, W^j]_t = \begin{cases} \sigma^i_t \, \sigma^j_t \, dt & i = j \, , \\ 0 \, dt & i \neq j \, , \end{cases}$$

Remark 6.20 (Vector notation). When X is a $\mathbb{R}^d$ valued process, we will write each individual coordinate (at time t) as X^1_t, X^2_t, …, X^d_t. There is no ambiguity with the notation for taking powers:

(1) If $X_t \in \mathbb{R}^d$, then taking powers of X_t doesn't make sense; so X^i_t refers to the i^{th} coordinate.
(2) If $X_t \in \mathbb{R}$, then taking coordinates doesn't make sense; so X^i_t refers to the i^{th} power.

Definition 6.21 (d-dimensional Brownian motion)**.** We say a d-dimensional process $W = (W^1, \dots, W^d)$ is a Brownian motion if:

(1) Each coordinate W^i is a standard 1-dimensional Brownian motion.
(2) For $i \neq j$, the processes W^i and W^j are independent.

Remark 6.22. By proposition W is a d-dimensional Brownian motion then

$$d[W^i, W^j]_t = \begin{cases} dt & i = j \, , \\ 0 \, dt & i \neq j \, . \end{cases}$$

Here's the d-dimensional Itô formula in vector notation:

Theorem 6.23 (Multi-dimensional Itô formula)**.**

- *Let X be a d-dimensional Itô process. $X_t = (X^1_t, \dots, X^d_t)$.*
- *Let $f = f(t, x)$ be a function that's defined for $t \in \mathbb{R}$, $x \in \mathbb{R}^d$.*
- *Suppose $f \in C^{1,2}$. That is:*
  - ▷ *f is once differentiable in t*
  - ▷ *f is twice in each coordinate x_i*
  - ▷ *All the above partial derivatives are continuous. Then:*

$$d(f(t, X_t)) = \partial_t f(t, X_t) \, dt + \sum_{i=1}^{d} \partial_i f(t, X_t) \, dX^i_t + \frac{1}{2} \sum_{i,j} \partial_i \partial_j f(t, X_t) \, d[X^i, X^j]_t$$

Remark 6.24 (Integral form of Itô's formula).

$$f(T, X_T) - f(0, X_0) = \int_0^T \partial_t f(t, X_t) \, dt + \sum_{i=1}^{d} \int_0^T \partial_i f(t, X_t) \, dX^i_t$$

$$+\frac{1}{2}\sum_{i,j}\int_0^T \partial_i\partial_j f(t,X_t)\,d[X^i,X^j]_t$$

Intuition behind Theorem 6.23. Will be done on the board. □

6.6. **Diffusions.** Given $b\colon \mathbb{R}^d \to \mathbb{R}^d$ and $\sigma\colon \mathbb{R}^d \to \mathbb{R}^d \times \mathbb{R}^d$, define the diffusion X by

$$dX_t = b(X_t)\,dt + \sigma(X_t)\,dW_t\,, \tag{6.5}$$

where W is a d-dimensional Brownian motion. For every $x \in \mathbb{R}^d$, let X_t^x denote the solution of (6.5) with initial data $X_0^x = x$. In order to simplify notation, we will often use x as a superscript for $\boldsymbol{E}$ and $\boldsymbol{P}$ instead of X. That is we define

$$\boldsymbol{E}^x f(X_t) \overset{\text{def}}{=} \boldsymbol{E} f(X_t^x) \qquad \text{and} \qquad \boldsymbol{P}^x(X_t \in A) \overset{\text{def}}{=} \boldsymbol{P}(X_t^x \in A)\,.$$

In words, $\boldsymbol{E}^x(f(X_t)$ simply means solve (6.5) for time t with initial data x and compute the expectation of $f(X_t)$.

For any $t > 0$ one can show that X_t^x is a continuous random variable. Let $p_t(x,y)$ denote the density of X_t^x. That is

$$\boldsymbol{P}^x(X_t \in A) = \int_A p_t(x,y)\,dy\,, \quad \boldsymbol{E}^x(f(X_t)) = \int_{\mathbb{R}^d} p_t(x,y) f(y)\,dy\,.$$

Proposition 6.25. *The process X_t is a (continuous time) Markov process, and the density p satisfies the Kolmogorov Chapman equation:*

$$p_t(x,y) = \int_{\mathbb{R}^d} p_s(x,z) p_{t-s}(z,y)\,dz\,, \quad \textit{for any } x,y \in \mathbb{R}^d\,,\ 0 < s < t\,.$$

Proof. A proof of this will follow from some of the results we prove later, and is on your homework. (The results we prove will not rely on this.) □

Proposition 6.26. *Let Y be a solution of (6.5) with initial data Y_0. Let ρ_0 be the density of Y_0. The density of Y_t is given by*

$$\rho_t(y) = \int_{\mathbb{R}^d} \rho_0(x) p_t(x,y)\,dx\,.$$

Proof. This is a continuous version of writing out the distribution of a Markov process in terms of its transition kernel. □

The main results we aim to prove here are the following:

Theorem 6.27. *The density p satisfies*

$$\partial_t p - \mathcal{L}_x p = 0 \quad \textit{(in variables } x,t\textit{)}\,, \tag{6.6}$$

$$\partial_t p - \mathcal{L}_y^* p = 0 \quad \textit{(in variables } y,t\textit{)}\,, \tag{6.7}$$

where $\mathcal{L}, \mathcal{L}^*$ are the differential operators defined by:

$$\mathcal{L} f = b \cdot \nabla f + \sum_{i,j=1}^d a_{i,j} \partial_i \partial_j f$$

$$\mathcal{L}^* f = -\nabla \cdot (bf) + \sum_{i,j=1}^d \partial_i \partial_j (a_{i,j} f)\,.$$

$$\textit{where} \quad a_{i,j} \overset{\text{def}}{=} \frac{1}{2}\sum_{k=1}^d \sigma_{i,k}\sigma_{j,k} = \frac{1}{2}(\sigma\sigma^T)_{i,j}$$

Theorem 6.28 (Kolmogorov backward equation)**.** *Let $f\colon \mathbb{R}^d \to \mathbb{R}$, and define*

$$\theta_t(x) = \boldsymbol{E}^x f(X_t)\,. \tag{6.8}$$

Then θ satisfies the PDE

$$\partial_t \theta = \mathcal{L}\theta \tag{6.9}$$

with initial data f. Conversely if θ satisfies (6.9) *then* (6.8) *holds.*

Remark 6.29. Once you have (6.8), you can solve the PDE (6.9) by Monte Carlo simulation.

Before explaining where $\mathcal{L}, \mathcal{L}^*$ come from, we note that they are are adjoints of each other.

Lemma 6.30. *For any $f, g\colon \mathbb{R}^d \to \mathbb{R}$ define*

$$\langle f, g\rangle = \int_{\mathbb{R}^d} f(x) g(x)\, dx\,.$$

The adjoint operator has the property that $\langle f, \mathcal{L}g\rangle = \langle \mathcal{L}^* f, g\rangle$.

Proof. Will be done on the board. □

The operator $\mathcal{L}$ arises when you apply Itô's formula to X.

Lemma 6.31. *If f is a $C^{1,2}$ function then*

$$df(t, X_t) = (\partial_t f(t, X_t) + \mathcal{L} f(t, X_t))\, dt + \sum_{i,j} \partial_i f(t, X_t) \sigma_{i,j}(X_t)\, dW^j_t$$

Proof. Will be done on the board. □

Remark 6.32. More generally, the operator $\mathcal{L}$ arises as the *generator* of the diffusion X. Explicitly, generator of a diffusion is the operator L defined by

$$Lf(x) = \lim_{t \to 0} \frac{\boldsymbol{E}^x f(X_t) - f(x)}{t}\,.$$

for all functions f where the limit exists. A direct calculation (using Lemma 6.31) shows that for all C^2 functions we have

$$Lf = \mathcal{L} f\,.$$

Proof of Theorem 6.28. The forward direction can be done using the Markov property (but we will not need it). We will prove on the board that the following generalization of the converse holds: For every $0 \leqslant t \leqslant T$, we have

$$\theta_T(x) = \boldsymbol{E}^x \theta_{T-t}(X_t) \tag{6.10}$$

□

Proof of equation (6.6) *in Theorem 6.27.* Will be done on the board using Theorem 6.28. □

Proof of equation (6.7) *in Theorem 6.27.* Differentiating (6.10) we see

$$\begin{aligned}
0 = \partial_t \boldsymbol{E}^x \theta_{T-t}(X_t) &= \partial_t \int_{\mathbb{R}^d} p_t(x, y) \theta_{T-t}(y)\, dy \\
&= \int_{\mathbb{R}^d} (\partial_t p_t(x, y) \theta_{T-t}(y) - p_t(x, y) \partial_t \theta_{T-t}(y))\, dy \\
&= \int_{\mathbb{R}^d} (\partial_t p_t(x, y) \theta_{T-t}(y) - p_t(x, y) \mathcal{L}_y \theta_{T-t}(y))\, dy
\end{aligned}$$

$$= \int_{\mathbb{R}^d} (\partial_t p_t(x,y)\theta_{T-t}(y) - \mathcal{L}_y^* p_t(x,y)\theta_{T-t}(y))\,dy$$
$$= \int_{\mathbb{R}^d} (\partial_t p_t(x,y) - \mathcal{L}_y^* p_t(x,y))\theta_{T-t}(y)\,dy\,.$$

Sending $t \to T$ this will imply

$$\int_{\mathbb{R}^d} (\partial_t p_t(x,y) - \mathcal{L}_y^* p_t(x,y))\theta_0(y)\,dy = 0\,,$$

and since θ_0 is arbitrary this implies $\partial_t p - \mathcal{L}_y^* p = 0$ as claimed. □

Theorem 6.33. *Suppose there exists a probability density function ρ that satisfies*

$$\mathcal{L}^* \rho = 0 \tag{6.11}$$

Then ρ is the stationary distribution of X, and as $t \to \infty$ the distribution of X_t converges to the distribution with density ρ.

Remark 6.34. There may not exist any probability density functions that satisfy (6.11). For instance if $b = 0$, then $\mathcal{L} = \mathcal{L}^* = \frac{1}{2}\Delta$, and the only solutions to (6.11) are constants. This is not a probability density function (on $\mathbb{R}^d$) since it does not integrate to 1.

Proof. A full proof of convergence is beyond the scope of this course; but I'll give some intuition. (Checking that it's the stationary distribution is on the homework.) □

6.7. **Applications to Sampling.** By Theorem 6.33, if we want to sample from a distribution with density p, we just need to find b, σ such that $\mathcal{L}^* p = 0$, and then simulate the SDE (6.5). As $t \to \infty$ the distribution will converge to a distribution with density p.

Theorem 6.35. *Let $U = -\ln p$, and consider the SDE*

$$dX_t = -\nabla U(X_t)\,dt + \sqrt{2}\,dW_t\,. \tag{6.12}$$

(Here W is a d-dimensional Brownian motion.) The stationary distribution of X has probability density function p.

Proof. Direct calculation (and is on your homework). □

The *Langevin Monte Carlo (LMC)* algorithm samples from p by discretizing (6.12) using the Euler Maruyama scheme.

(1) Choose a time step τ
(2) Define

$$X_{n+1} = X_n - \nabla U(X_n)\,\tau + \sqrt{2\,\tau}\,\zeta_{n+1} \tag{6.13}$$

where ζ_{n+1} is an independent d-dimensional standard normal.

Here X_n represents the solution to (6.12) at time $n\tau$. As $\tau \to 0$ and $n \to \infty$, the density of X_n converges to the desired probability density function p.

Remark 6.36. If you know that $p(x) = \frac{1}{Z} p_u(x)$ for some un-normalized probability density function p_u, then LMC can still be used. Notice

$$\nabla \ln p = \nabla \ln p_u\,,$$

and so if we use $U = -\ln p_u$, the stationary distribution of (6.13) will have density p.

Remark 6.37. The LMC algorithm works well if U is convex. If U is not convex, it could take a very long time to converge, and an example was on your homework. However, in many practical situations (even when U is not convex), this algorithm (or some variant) is the only viable option.

The *Metropolis Adjusted Langevin Algorithm (MALA)* combines the LMC and the Metropolis Hastings algorithm. Instead of choosing X_{n+1} according to (6.13), propose it as a new state and accept it according to the Metropolis Hastings algorithm. One can compute the acceptance ratio explicitly, and the MALA can be described as:

(1) Choose a time step τ, and $U = -\ln p$ (or $U = -\ln p_u$).

(2) Propose a new state $\tilde{X}_{n+1}$ according to

$$\tilde{X}_{n+1} = X_n - \nabla U(X_n)\,\tau + \sqrt{2\,\tau}\,\zeta_{n+1}\,.$$

(3) Define the acceptance ratio

$$A(x,y) = \min\Bigl\{1, \frac{p_u(y)Q(y,x)}{p_u(x)Q(x,y)}\Bigr\} \quad \text{where} \quad Q(x,y) \propto \exp\Bigl(\frac{-1}{4\tau}|y-x+\tau\nabla U(x)|^2\Bigr)$$

(4) Flip a coin that lands heads with probability $A(X_n, \tilde{X}_{n+1})$. If the coin lands heads, let $X_{n+1} = \tilde{X}_{n+1}$. Otherwise let $X_{n+1} = X_n$.

Remark 6.38. MALA has similar use cases as LMC; MALA usually moves into regions of high probability faster than LMC.

Example 6.39. For the pro-bit model, the un-normalized density for the posterior distribution for β is given by

$$\pi_u(\beta) = p(\beta) \prod_{i=1}^{d} \Phi(z_i \cdot \beta)^{y_i} (1 - \Phi(z \cdot \beta))^{1-y_i}\,.$$

where p is a prior, Φ is the CDF of the standard normal and $\{(z_i, y_i)\}$ are the labelled points. If we compute the posterior mean $\bar{\beta}$ using MALA / LMC, then given a feature vector $z \in \mathbb{R}^d$ we can predict the label $y = 1$ with probability $\Phi(\bar{\beta} \cdot z)$.

7. **Simulated Annealing.**

Often one wants to maximize a function F in a large (high dimensional) space. A elementary *stochastic hill climb* algorithm for this is:

- Start at a point $x \in \mathcal{X}$.
- Choose a close by point y randomly.
- If $F(y) > F(x)$, move to y. Otherwise stay at x.
- Repeat until your computational budget is exhausted, or until F isn't increasing much.

The drawbacks of this method are that the hill climb may get stuck at local maxima and often takes too long to go up ridges. When you're working with functions whose derivative you can compute easily, then *stochastic gradient descent* may provide better results.

An alternate approach is *simulated annealing* – this may make you down climb at certain points, but avoids getting stuck in local maxima. Conventionally, simulated annealing is always stated to minimize a function; replace the function by its negative if you want to maximize it instead.

The basis simulated annealing is the following elementary observation.

Lemma 7.1. *Let $f\colon \mathcal{X} \to \mathbb{R}$ be some function, $\beta > 0$, and define and un-normalized probability distribution $\pi_u = e^{-\beta f}$. Let $Z = \sum_{\mathcal{X}} \pi_u$ be the normalization constant, and $\pi = \pi_u / Z$ be the normalized probability distribution. When $\beta = 0$, π is uniformly distributed on all of $\mathcal{X}$. As $\beta \to \infty$, π converges to a probability distribution that is uniformly distributed on the* global minima *of the function f.*

Proof. Done on the board. □

Simulated annealing can now be described as follows:

- Fix a sequence of *temperatures* $T_n \to 0$ as $n \to \infty$.
- Fix a proposal mechanism Q, and start with some $X_0 \in \mathcal{X}$.
- Generate X_{n+1} from X_n by using the Metropolis–Hastings to sample from the (un-normalized) probability distribution $\pi_u = e^{-f/T_n}$.
- Repeat for a large number of iterations.

Initially (when the temperature T_n is large), un-normalized probability distribution $\pi_u = e^{-f/T_n}$ is roughly uniform. So the process X explores $\mathcal{X}$ fast. As $T_n \to 0$, π_u starts clustering around the minima of f, and the typical location of X_n should be around the minimum of f.

The choice of temperatures is the *annealing schedule* (or cooling schedule) is situation dependent. In practice one chooses T_0 to be something large (so that $\beta = 1/T_0$ is small), and T_N to be very small, and allows T_n to decrease geometrically from T_0 to T_N.

7.1. **Example: Travelling salesman.** Given N points on the plane (cities), the *travelling salesman problem* is to find a route that travells through each city exactly once, and returns to the starting point. This is a "classic" problem which is known to be NP-hard, and you can read more about it on Wikipedia

This has been extensively studied, and there are several well known combinatorial algorithms that yield results close to the optimal path in practical amounts of time. Simulated annealing was originally proposed in order to solve the traveling salesman problem.

The simplest algorithm for the traveling salesman problem is the *greedy nearest neighbor* algorithm: Start anywhere, and simply travel to the nearest unvisited city. There are certain scenarios where this algorithm performs badly; but in most configurations this gives you a travel distance that is within 25% of the minimum.

We can improve this as follows:

1. Given a tour σ (which is just some permutation of the N cities), let $C(\sigma)$ be the total length (or cost) of the tour. We will use simulated annealing to minimize C over all tours.
2. To use simulated annealing, we need a *proposal mechanism.* Given a tour σ, pick two cities randomly and let τ be a new tour that is the same as σ except it swaps the order in which it visits the two chosen cities. (You can also pick three cities, and cycle the order in which they are visited, or use many other variations on this theme.)
3. Choose a cooling schedule (by experimenting) and run simulated annealing to minimize C with the above proposal mechanism.

A Python implementation of this algorithm is on the class website. (Part of it is redacted, and filling it in is part of your homework.)

7.2. **Example: Cracking substitution ciphers.** A *substitution cipher* is one where you create a *key* that is a permutation of the alphabet (e.g. $A \mapsto K$, $B \mapsto Z$, etc.). Using this key, you can encode and decode a message. At first sight this might seem uncrackable by brute force – your key is one permutation of 28! (26 letters plus a period and space punctuation).

This is a needle in an enormous haystack. If you could examine 10^{12} keys in a second (which is a generous overestimate), then it would still take you about *a billion years* to crack this code. Nevertheless, if you're sending sufficiently long (few paragraphs) of readable text data, this method is crackable in seconds using simulated annealing (or even just a stochastic hill climb).

To crack a substitution cipher, we need to first define a *fitness function.* Download a few long English books, and compute frequencies of letter sequences. That is compute how often 'a' occurs, how often 'as' occurs, how often 'ast' occurs, and so on. Using these frequencies define a *fitness function* F that takes as input a string of symbols (e.g. a message), and outputs a real number. The closer the symbol frequencies match English, the higher the fitness should be. (There's an elegant and clever way to do this, which I'm not describing here as finding it is part of your homework.)

We will now crack substitution ciphers as follows:

- Given a guess at a key σ, define $f(\sigma)$ to be the fitness of the coded message M decoded with key σ. (We want to maximize $f(\sigma)$.)
- Given a (random) key σ, our *proposal mechanism* generates a new key τ by randomly swapping two symbols in σ.
- Run a stochastic hill climb to maximize $f(\sigma)$, or simulated annealing to minimize $-f(\sigma)$. (Both work really well!)

As an example, we took a passage from Arthur Conan Doyle's *Sherlock Holmes*, and coded it up with a randomly chosen key. We then downloaded five different books from Project Gutenberg, and ran simulated annealing. Here's the decoded message after every 100 iterations. The $F = \dots$ shows the fitness of the decoded message with the current guess of the key, and the $D = \dots$ shows the number of symbols where our guessed key and actual key differ. We find the correct key in a few thousand iterations! (A redacted version of this notebook is on the class website, and part of your homework is to find a fitness function and implement simulated annealing to crack substitution ciphers.)

```
F=-470.33, D=27:  . R.SLDBERVW.LREPDS.SLD.GS.FEXFNS. LD.XRPFZO.G.LFAD.SDEMRP.LDFBM.LGP.PDZ GRZ.LDB
F=-374.97, D=24:  STFS OPUEFHJSOFEMP S OPSR SNEANB STOPSAFMNLCSRSONVPS PEIFMSOPNUISORMSMPLTRFLSOPU
F=-328.22, D=24:   RE SANUCEJZ AECMNS SAN IS PCGPHS RAN GEMPOB I APXN SNCTEM ANPUT AIM MNORIEO ANU
F=-301.30, D=21:   CO SANULOVF AOLMNS SAN IS ELHE.S CAN HOMERT I AEZN SNLGOM ANEUG AIM MNRCIOR ANU
F=-294.07, D=17:   CO SANULOVG AOLMNS SAN IS ELHE.S CAN HOMERK I AEQN SNLDOM ANEUD AIM MNRCIOR ANU
F=-290.06, D=18:   CO SANULOFG AOLMNS SAN IS ELVE.S CAN VOMERK I AEZN SNLDOM ANEUD AIM MNRCIOR ANU
F=-246.34, D=13:   DO NSERLOCK SOLMEN NSE IN ALVA.N DSE VOMAUY I SAWE NELTOM SEART SIM MEUDIOU SER
F=-245.93, D=13:   DO NSERLOCK SOLMEN NSE IN ALFA.N DSE FOMAUY I SAWE NELTOM SEART SIM MEUDIOU SER
F=-222.94, D=11:   TO NSERLOCK SOLMEN NSE IN ALUAYN TSE UOMAF. I SAWE NELDOM SEARD SIM MEFTIOF SER
F=-140.26, D= 8:   TO FHERLOCK HOLMEF FHE IF ALWAYF THE WOMAN. I HAUE FELDOM HEARD HIM MENTION HER
F=-141.88, D= 6:   TO FHERLOCK HOLMEF FHE IF ALWAYF THE WOMAN. I HAJE FELDOM HEARD HIM MENTION HER
F=-112.05, D= 4:   TO FHERLOCK HOLMEF FHE IF ALWAYF THE WOMAN. I HAXE FELDOM HEARD HIM MENTION HER
F= -34.58, D= 3:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAZE SELDOM HEARD HIM MENTION HER
F= -26.99, D= 2:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
F= -27.95, D= 3:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
F= -27.64, D= 2:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
F= -27.64, D= 2:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
F= -28.26, D= 3:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
F= -27.64, D= 2:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
F= -28.26, D= 3:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
F= -28.26, D= 3:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
F= -26.99, D= 2:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
F= -25.54, D= 2:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
F= -25.88, D= 2:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
```

```
F= -26.44, D= 3:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
F= -26.26, D= 3:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
F= -25.88, D= 2:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
F= -26.26, D= 3:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
F= -26.26, D= 3:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
F= -26.26, D= 3:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
F= -26.26, D= 3:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
F= -26.26, D= 3:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
F= -25.54, D= 2:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
F= -24.30, D= 0:   TO SHERLOCK HOLMES SHE IS ALWAYS THE WOMAN. I HAVE SELDOM HEARD HIM MENTION HER
```

References

[LP17] D. A. Levin and Y. Peres. *Markov chains and mixing times*. American Mathematical Society, Providence, RI, 2017. doi:10.1090/mbk/107. Second edition of [MR2466937], With contributions by Elizabeth L. Wilmer, With a chapter on “Coupling from the past” by James G. Propp and David B. Wilson.