# Chapter 11

# Monte Carlo Integration

High–dimensional integrals appear everywhere in theoretical physics. One clear example where high-dimensional integrals are needed is the case of Statistical Mechanics. In that case, the classical partition function

$$Z \;=\; \int \mathrm{e}^{-\beta H(\mathbf{p},\mathbf{q})} \mathrm{d}\mathbf{p}\mathrm{d}\mathbf{q}$$

already sits in a $6N$–dimensional phase space for $N$ particles.

When the dimensionality of integrals $d$ is large, deterministic quadrature collapses because, as we show below, the number of function evaluations needed scales exponentially with $d$, soon becoming impractical for computers. Monte Carlo (MC) methods instead replace the grid by $N$ random points; as we will show, the approximation error (the root–mean–square error) is $\mathcal{O}(N^{-1/2})$ and—most importantly—*independent of* $d$. That dimensional immunity makes MC essential once $d \gtrsim 5$.

## 11.1 Reference Integral

For *any* of the applications mentioned above, we can—without loss of generality—focus on evaluating the integral

$$I = \int_{[0,1]^d} f(\mathbf{x})\, \mathrm{d}\mathbf{x}, \qquad \mathbf{x} = (x_1, \ldots, x_d), \tag{11.1.1}$$

because more complicated domains can be mapped onto the unit cube by elementary transformations. We give below a few specific examples of such transformations.

### 11.1.1 Finite but non-unit intervals

For a one dimensional integral, a general finite integration interval in $[a, b]$ is reduced by the affine map $u = (x-a)/(b-a) \in [0, 1]$ to belong in the unit region as defined above. With $\mathrm{d}x = (b-a)\, \mathrm{d}u$

we obtain

$$\int_a^b g(x)\,\mathrm{d}x \;=\; (b-a)\int_0^1 g\big(a+(b-a)u\big)\,\mathrm{d}u.$$

For a box $\prod_{k=1}^d [a_k, b_k]$ apply the same map on each coordinate:

$$x_k = a_k + (b_k - a_k)u_k, \quad u_k \in [0,1].$$

The Jacobian is $\prod_k (b_k - a_k)$, so

$$\int_{\prod[a_k,b_k]} f(\mathbf{x})\,\mathrm{d}\mathbf{x} = \Big[\prod_{k=1}^d (b_k - a_k)\Big] \int_{[0,1]^d} f\big(\mathbf{a} + B\mathbf{u}\big)\,\mathrm{d}\mathbf{u},$$

with $B = \mathrm{diag}(b_1 - a_1, \ldots, b_d - a_d)$. We have thus reduced a general integral with finite intervals to the form presented in Eq. (11.1.1).

### 11.1.2 Unbounded domains

Consider a one-dimensional integral $\int_0^\infty g(x)\,\mathrm{d}x$. The substitution $u = \frac{x}{1+x} \in [0,1)$ gives $x = u/(1-u)$ and $\mathrm{d}x = \mathrm{d}u\,(1-u)^{-2}$:

$$\int_0^\infty g(x)\,\mathrm{d}x = \int_0^1 g\Big(\tfrac{u}{1-u}\Big)\,\frac{\mathrm{d}u}{(1-u)^2}.$$

The multi-dimensional analogue applies the map independently on each infinite axis. Other choices—$\tanh^{-1}$, arctan—are equally valid; all produce a finite-volume Jacobian.

## 11.2 Tensor–Product Grids

The simplest approach to numerical integration is to divide the integration interval into an equispaced grid. In higher dimension, this means that each axis is divided into $n$ subintervals of length $h = 1/n$. Each axis has $n+1$ nodes, thus the total number of grid points is $N_{\text{grid}} = (n+1)^d \approx h^{-d}$.

Given the grid, one can apply any integration rule of choice. For example, it is possible to apply a $p$-th order Newton–Cotes rule along every axis:

$$I_h = \sum_{\mathbf{j}} w_{\mathbf{j}}\, f(\mathbf{x}_{\mathbf{j}}), \quad \mathbf{x}_{\mathbf{j}} = (j_1 h, \ldots, j_d h). \tag{11.2.1}$$

**Deterministic error**

If all mixed partials of order $p$ are continuous, $|I - I_h| \le C h^p$. Choosing $h \asymp \varepsilon^{1/p}$ gives cost $N_{\text{grid}} \asymp \varepsilon^{-d/p}$ — exponential in $d$.

Given the exponential cost of this approach, we seek now an alternative way, based instead on randomly choosing the grid points.

## 11.3 Integral as Expectation

Select a probability density $p(\mathbf{x}) > 0$ on the cube with $\int_{[0,1]^d} p(\mathbf{x})\,\mathrm{d}\mathbf{x} = 1$. Writing $1 = p(\mathbf{x})/p(\mathbf{x})$ under the integral converts the *deterministic* problem of integrating $f$ into the *statistical* problem of averaging a random variable:

$$I = \int_{[0,1]^d} \frac{f(\mathbf{x})}{p(\mathbf{x})}\, p(\mathbf{x})\,\mathrm{d}\mathbf{x} = \mathbb{E}_{\mathbf{X}\sim p}[Y(\mathbf{X})], \qquad Y(\mathbf{X}) = \frac{f(\mathbf{X})}{p(\mathbf{X})}. \tag{11.3.1}$$

Here $\mathbb{E}_{\mathbf{X}\sim p}[Y(\mathbf{X})]$ denotes the expected value of the random variable $Y$ over the probability distribution $p(\mathbf{X})$. The freedom to choose $p$ lets us steer sampling effort toward regions where $f$ is most influential.

## 11.4 Monte Carlo Estimator

Draw $N$ independent samples $\mathbf{X}_1, \ldots, \mathbf{X}_N \sim p$ and define

$$\boxed{\hat{I}_N \;=\; \frac{1}{N}\sum_{i=1}^{N} Y_i}, \qquad Y_i \;=\; \frac{f(\mathbf{X}_i)}{p(\mathbf{X}_i)}. \tag{11.4.1}$$

**Unbiasedness.** Because each $Y_i$ has mean

$$\mathbb{E}[Y_i] \;=\; \int_{[0,1]^d} \frac{f(\mathbf{x})}{p(\mathbf{x})}\, p(\mathbf{x})\,\mathrm{d}\mathbf{x} = I,$$

linearity of expectation gives

$$\mathbb{E}[\hat{I}_N] = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}[Y_i] = I.$$

**Why averaging cuts noise.** For *independent, identically distributed* (IID) variables, variances add. Hence

$$\mathrm{Var}(\hat{I}_N) = \mathrm{Var}\Big(\frac{1}{N}\sum_{i=1}^{N} Y_i\Big) = \frac{1}{N^2}\sum_{i=1}^{N} \mathrm{Var}(Y_i) = \frac{\mathrm{Var}(Y)}{N}.$$

The factor $1/N$ is the statistical reward for averaging $N$ copies.

**Variance formula & root-mean-square (RMS) error**

$$\boxed{\mathrm{Var}(\hat{I}_N) \;=\; \frac{1}{N}\Big(\int_{[0,1]^d} \frac{f(\mathbf{x})^2}{p(\mathbf{x})}\,\mathrm{d}\mathbf{x} - I^2\Big)} \quad\Longrightarrow\quad \text{RMS error} = \frac{\sigma}{\sqrt{N}}, \quad \sigma^2 = \mathrm{Var}(Y).$$

The prefactor $\sigma$ measures the *intrinsic roughness* of the weighted integrand $f/p$; choosing a better sampling density $p$ means shrinking $\sigma$ before the $N^{-1/2}$ law goes to work.

**Central Limit Theorem (CLT) and error bars.** If $\sigma^2 < \infty$ then

$$\sqrt{N}\,(\hat{I}_N - I) \;\xrightarrow{\mathcal{D}}\; \mathcal{N}(0, \sigma^2),$$

i.e. the *scaled* error becomes Gaussian for large $N$. The CLT therefore turns a qualitative statement ("errors decay like $N^{-1/2}$") into a quantitative tool for stopping rules and uncertainty quantification. A practical consequence is, for example, that if we can establish the result of the integral with a 95 % confidence interval

$$I \;\in\; \big[\,\hat{I}_N \;\pm\; 1.96\,\widehat{\sigma}/\sqrt{N}\big],$$

where $\widehat{\sigma}$ is the sample standard deviation of the weights $Y_i$. Other confidence intervals can be chosen depending on the accuracy one seeks.

## 11.5 Choice of the distribution and Importance Sampling

The simplest choice is the *uniform* density

$$p(\mathbf{x}) \equiv 1, \qquad \mathbf{x} \in [0,1]^d.$$

Every Monte-Carlo sample is then a point $\mathbf{X} \sim U([0,1]^d)$ drawn with equal probability anywhere in the cube, and each weight reduces to a bare function evaluation:

$$Y \;=\; \frac{f(\mathbf{X})}{p(\mathbf{X})} \;=\; f(\mathbf{X}).$$

**Explicit variance computation.** With $p = 1$ the variance of a single weight is

$$\sigma^2 \;=\; \mathrm{Var}_U(f) = \mathbb{E}_U\big[f(\mathbf{X})^2\big] - \Big(\mathbb{E}_U\big[f(\mathbf{X})\big]\Big)^2 \tag{11.5.1}$$

$$\;=\; \int_{[0,1]^d} f(\mathbf{x})^2\,\mathrm{d}\mathbf{x} \;-\; \big(I\big)^2, \tag{11.5.2}$$

so the Monte-Carlo estimator satisfies

$$\mathrm{Var}\big(\hat{I}_N\big) \;=\; \frac{\sigma^2}{N} \;=\; \frac{1}{N}\Big(\int_{[0,1]^d} f(\mathbf{x})^2\,\mathrm{d}\mathbf{x} - I^2\Big).$$

**When is uniform sampling good?**

- *Smooth, slowly varying integrands.* If $f$ is nearly flat, the difference $\int f^2 - I^2$ is small, so uniform MC works fine.

- *Diagnostics are easy.* No need to evaluate $p(\mathbf{x})$; each sample costs a single call to $f$.

**When is uniform sampling bad?**

- *Sharp peaks or heavy tails.* If $f(\mathbf{x})$ is large only on a tiny fraction of the cube, most samples contribute almost nothing, inflating $\sigma^2$.

- *Oscillatory integrands.* Alternating signs lead to severe cancellation, so the numerator in (11.5.2) can be huge even though the integral $I$ is modest.

These shortcomings motivate *importance sampling*, where a carefully chosen density $p(\mathbf{x})$ places samples precisely where $f$ has the largest effect—shrinking both $\sigma^2$ and the overall error bar.

Reducing the *variance prefactor* is the main leverage to speed up convergence, which brings us to importance sampling. Goal: choose $p$ to shrink the integral in the variance formula.

**Optimal density $p^*$**

We minimise

$$J[p] = \int_{[0,1]^d} \frac{f(\mathbf{x})^2}{p(\mathbf{x})}\,\mathrm{d}\mathbf{x}$$

subject to the constraint $\int_{[0,1]^d} p(\mathbf{x})\,\mathrm{d}\mathbf{x} = 1$. Introduce a Lagrange multiplier $\lambda$ and form

$$\mathcal{L}[p] = J[p] - \lambda\Big(\int p - 1\Big).$$

Take the functional derivative and set it to zero:

$$\frac{\delta \mathcal{L}}{\delta p} = -\frac{f(\mathbf{x})^2}{p(\mathbf{x})^2} - \lambda = 0 \quad\Longrightarrow\quad p(\mathbf{x}) = \frac{|f(\mathbf{x})|}{\sqrt{\lambda}}.$$

Enforce normalisation to determine $\sqrt{\lambda}$:

$$\sqrt{\lambda} = \int_{[0,1]^d} |f(\mathbf{u})|\,\mathrm{d}\mathbf{u}.$$

Hence the variance-minimising importance density is

$$p^*(\mathbf{x}) = \frac{|f(\mathbf{x})|}{\displaystyle\int_{[0,1]^d} |f(\mathbf{u})|\,\mathrm{d}\mathbf{u}}.$$

With $p^*$ every weight $Y$ has constant magnitude, and $\mathrm{Var}(\hat{I}_N)$ achieves its minimum.

In practice working with the optimal importance sampling distribution is not possible, and we approximate $p^*$ with a tractable family, sample, weight, average, and quote the CLT error bar.

## 11.6 Example: Estimating $\pi$ via a Finite Integral

We can estimate $\pi$ by evaluating the following integral, which has an analytical result:

$$\int_0^1 \frac{x^4(1-x)^4}{1+x^2}dx = \frac{22}{7} - \pi. \tag{11.6.1}$$

Defining

$$I = \int_0^1 \frac{x^4(1-x)^4}{1+x^2}dx, \tag{11.6.2}$$

we can recover an estimate for $\pi$ as:

$$\pi = \frac{22}{7} - I. \tag{11.6.3}$$

### 11.6.1 Simple Monte Carlo for the $\pi$ Integral

The process for the Monte Carlo estimation is:

1. Generate $N$ random samples $x_i \sim U(0, 1)$.
2. Compute:
$$I \approx \frac{1}{N} \sum_{i=1}^{N} \frac{x_i^4 (1 - x_i)^4}{1 + x_i^2}. \tag{11.6.4}$$
3. Estimate $\pi$ by:
$$\pi \approx \frac{22}{7} - I. \tag{11.6.5}$$

### 11.6.2 Importance Sampling with a Gaussian Proposal

When the integrand is sharply peaked within the integration interval, importance sampling can reduce the variance further. For instance, choosing a Gaussian sampling distribution centered at $\mu = 0.5$ with standard deviation $\sigma = 0.2$:

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x - 0.5)^2}{2\sigma^2}\right), \tag{11.6.6}$$

we modify our estimator as follows:

1. Generate $N$ samples $x_i$ from the Gaussian distribution $p(x)$.
2. Compute the estimate, including the indicator function $\mathbb{1}_{[0,1]}(x_i)$ to restrict samples to the integration interval:
$$I \approx \frac{1}{N} \sum_{i=1}^{N} \frac{x_i^4 (1 - x_i)^4}{1 + x_i^2} \cdot \frac{1}{p(x_i)} \mathbb{1}_{[0,1]}(x_i). \tag{11.6.7}$$
3. Then, estimate $\pi$ by:
$$\pi \approx \frac{22}{7} - I. \tag{11.6.8}$$

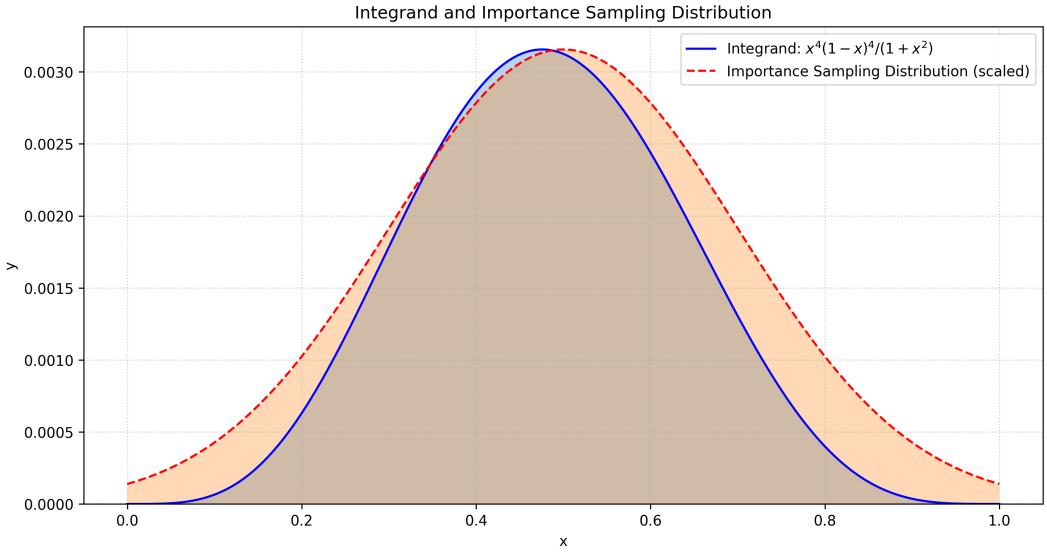



Figure 11.6.1: Visualization of the integrand (blue solid line) along with the Gaussian sampling distribution (red dashed line) used for importance sampling.

# Chapter 12

# Markov Chains

Markov chains, named after the mathematician Andrey Markov, provide a mathematical framework for modelling systems that hop randomly between a collection of states. They will be instrumental later on when we devise general algorithms—such as Metropolis–Hastings—to draw representative configurations from a high-dimensional probability distribution $P(x)$. Markov chains provide a general mathematical framework for modeling systems that transition between different states over time, where the future state depends only on the current state and not on the past history.

## 12.1 State space: labels and examples

We shall work with a (physical, or not) system described by some state variable $X \in S$, where $S$ is a *finite* state space,

$$S = \{1, 2, \ldots, m\}. \qquad m \in \mathbb{N}, \tag{12.1.1}$$

Each element is merely a *label* for a possible configuration of the system. In general, the choice of labels is arbitrary: our notation favours integers because it makes matrix expressions compact. Whenever helpful for intuition, we will write the corresponding verbal description in parentheses.

Besides the state of the system, we further introduce a collection (a sequence, in fact) of states. We denote this sequence of states $X_0, X_1, X_2 \ldots$. We assume that the system has some intrinsic probabilistic dynamics, meaning that it transits from a state $X_{n-1}$ to its subsequent state $X_n$ via a random process.

**Example: Weather** A canonical three-state model for daily weather would use

$$S = \{1, 2, 3\}, \quad 1 = \text{Sunny}, \ 2 = \text{Cloudy}, \ 3 = \text{Rainy}.$$

Thus, in this notation, $X = 1$ means that the weather is sunny, and so on. The sequence of states $X_n$ here could specify, for example, the weather at a given day $n$. For instance, $X_5 = 3$ indicates that it is rainy on day 5.

**Example: Coin-toss** Consider a biased coin that shows Heads (H) with probability $p$ and Tails (T) with probability $1-p$. We label:

$$S = \{1, 2\}, \quad 1 = \mathrm{H},\ 2 = \mathrm{T}.$$

If the coin is tossed once per second, then in this case the sequence $X_n$ is simply the outcome of the $n$th toss.

## 12.2 Definition: the Markov property

A sequence of random variables $\{X_n\}_{n\geq 0}$ with values in $S$ is called a *time-homogeneous Markov chain* if, for every $k \geq 0$ and any states $X_0, \ldots, X_1, X_{k+1} \in S$,

$$\Pr\big(X_{k+1} \mid X_k, \ldots, X_0\big) = \Pr\big(X_{k+1} \mid X_k\big). \tag{12.2.1}$$

In words: *once we know the current state, all information about previous states is irrelevant for predicting the next one.* This "memoryless" feature distinguishes Markov chains from general stochastic processes.

Key consequences of (12.2.1):

- **Discrete time.** Evolution occurs at integer steps $n = 0, 1, 2, \ldots$.
- **Memorylessness.** The future state $X_{n+1}$ depends probabilistically only on $X_n$, it is instead independent of $X_0, \ldots, X_{n-1}$.
- **Time-homogeneity.** The transition mechanism does not depend explicitly on $n$; we use the same rule at each step.

## 12.3 Transition probabilities and matrix

For any pair of states $i, j \in S$, define the *one-step transition probability*

$$T_{ij} = \Pr\big(X_{n+1} = j \mid X_n = i\big), \tag{12.3.1}$$

which by time-homogeneity is independent of $n$. Collecting these into an $m \times m$ array yields the *transition matrix*

$$\hat{T} \equiv \big(T_{ij}\big)_{i,j=1}^{m}, \tag{12.3.2}$$

where the $i$th row lists the probabilities of leaving state $i$.

The transition matrix $\hat{T}$ is therefore an $m \times m$ matrix where each element $T_{ij}$ represents the probability of transitioning from state $i$ to state $j$:

$$T = \begin{pmatrix} T_{11} & T_{12} & \cdots & T_{1m} \\ T_{21} & T_{22} & \cdots & T_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ T_{m1} & T_{m2} & \cdots & T_{mm} \end{pmatrix} \tag{12.3.3}$$

Properties of the transition matrix:

- All elements are non-negative: $T_{ij} \geq 0$ for all $i, j$
- Each row sums to 1: $\sum_{j=1}^{m} T_{ij} = 1$ for all $i$
- The $(i, j)$-th entry of $T^n$ gives the probability of going from state $i$ to state $j$ in exactly $n$ steps

## 12.4 Classification of States

States in a Markov chain can be classified based on their long-term behavior:

- **Recurrent State**: A state $i$ is recurrent if, starting from $i$, the probability of eventually returning to $i$ is 1.
- **Transient State**: A state $i$ is transient if, starting from $i$, there is a non-zero probability that the chain will never return to $i$.
- **Absorbing State**: A state $i$ is absorbing if, once entered, it is impossible to leave (i.e., $T_{ii} = 1$).
- **Periodic State**: A state $i$ has period $d > 1$ if $d$ is the greatest common divisor of all $n > 0$ such that $T_{ii}^n > 0$.
- **Aperiodic State**: A state is aperiodic if its period is 1.

## 12.5 Ergodicity and Limiting Behavior

A Markov chain is said to be ergodic if it is both irreducible (it is possible to get from any state to any other state in a finite number of steps) and aperiodic. Ergodic Markov chains have a unique stationary distribution, which we will discuss in more detail later.

The limiting behavior of a Markov chain as $n \to \infty$ is of particular interest in many applications. For an ergodic Markov chain, the $n$-step transition probabilities converge to the stationary distribution regardless of the initial state.

## 12.6 Weather Model: A Simple Markov Chain Example

To illustrate the concepts of Markov chains, let's consider a simple weather model. We'll model the weather as a Markov chain with three states: Sunny (S), Cloudy (C), and Rainy (R).

The transition probabilities are given as follows:

- If it's sunny today:
  - 70% chance of being sunny tomorrow
  - 20% chance of being cloudy tomorrow
  - 10% chance of being rainy tomorrow
- If it's cloudy today:
  - 30% chance of being sunny tomorrow
  - 40% chance of being cloudy tomorrow
  - 30% chance of being rainy tomorrow
- If it's rainy today:
  - 20% chance of being sunny tomorrow
  - 30% chance of being cloudy tomorrow
  - 50% chance of being rainy tomorrow

We can represent this Markov chain with the following transition matrix:

$$T = \begin{pmatrix} 0.7 & 0.2 & 0.1 \\ 0.3 & 0.4 & 0.3 \\ 0.2 & 0.3 & 0.5 \end{pmatrix} \tag{12.6.1}$$

Where the rows represent the current state (S, C, R) and the columns represent the next state (S, C, R).

Using this transition matrix, we can answer various questions about the weather model:

- What's the probability of having three sunny days in a row?

$$P(\text{SSS}) = 0.7 \times 0.7 = 0.49$$

- If it's rainy today, what's the probability of it being sunny two days from now?

$$P(\text{S—R}) = 0.2 \times 0.7 + 0.3 \times 0.3 + 0.5 \times 0.2 = 0.31$$

- What's the long-term probability of a sunny day? (This involves finding the stationary distribution, which we'll cover in the next Chapter)

### 12.6.1 Graph Representation

A graph representation of this Markov chain can help visualize the states and transitions:

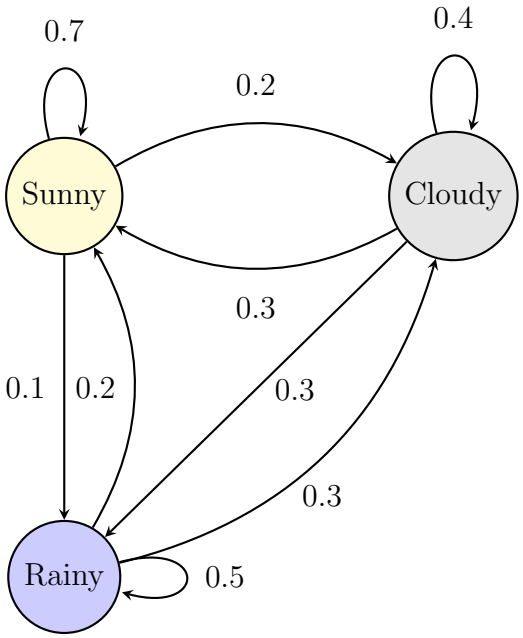



Figure 12.6.1: Graph representation of the weather model Markov chain

In this graph:

- Each node represents a state (Sunny, Cloudy, or Rainy).
- Each edge represents a possible transition between states.
- The numbers on the edges represent the transition probabilities.
- Self-loops represent the probability of staying in the same state.

This visual representation makes it easy to see the possible state transitions and their associated probabilities. For example, we can quickly see that the highest probability is to stay in the Sunny state if it's already sunny (0.7), while the lowest probability is transitioning from Sunny to Rainy (0.1).

# Chapter 13

# Stationary Distributions and Detailed Balance

A cornerstone of Markov-chain theory is the *stationary distribution*, which captures the long-time behavior of the chain irrespective of its initial state.

## 13.1 Definition

Let $T \in \mathbb{R}^{m\times m}$ be the transition matrix of a finite Markov chain with state space $S = \{1, 2, \ldots, m\}$. We wish to characterize the probability distribution of finding, at long times, the system in a given state $i \in S$, namely the probability $P_i = \Pr\big(X_{n\to\infty} = i\big)$. We arrange these probabilities in a column-vector

$$\boldsymbol{P} = \begin{pmatrix} P_1 \\ P_2 \\ \cdots \\ P_m \end{pmatrix} \qquad P_i \geq 0,\ \sum_{i=1}^{m} P_i = 1,$$

and we state that this is a *stationary distribution* (or *equilibrium distribution*) if and only if it satisfies

$$T^{\mathsf{T}}\boldsymbol{P} = \boldsymbol{P}. \tag{13.1.1}$$

In components,

$$P_i = \sum_{j=1}^{m} T_{ji}\, P_j, \qquad i = 1, \ldots, m, \tag{13.1.2}$$

i.e. the probability flowing *into* state $i$ balances the probability already present there.

### 13.1.1 Properties of Stationary Distributions

1. **Existence.** Every finite Markov chain has at least one stationary distribution (Kakutani–Markov–Kolmogorov theorem).

2. **Uniqueness.** If the chain is *irreducible* and *aperiodic* (*ergodic*), the stationary distribution is unique and strictly positive.

3. **Convergence.** For an ergodic chain the distribution after $k$ steps converges to $\boldsymbol{P}$ as $k \to \infty$, regardless of the initial distribution.

4. **Spectral characterisation.** $\boldsymbol{P}$ is a right eigenvector of $T^{\mathsf{T}}$ (or a left eigenvector of $T$) with eigenvalue 1.

### 13.1.2 Computing a Stationary Distribution

Equation (13.1.1) is the homogeneous linear system

$$\big(T^{\mathsf{T}} - I\big)\boldsymbol{P} = \mathbf{0}, \quad \sum_{i} P_i = 1,$$

which can be solved analytically for very small $m$ or numerically (power iteration, *etc.*) for larger chains.

### 13.1.3 Example: Three-State Weather Model

For the weather transition matrix

$$T = \begin{pmatrix} 0.7 & 0.2 & 0.1 \\ 0.3 & 0.4 & 0.3 \\ 0.2 & 0.3 & 0.5 \end{pmatrix}, \qquad \text{(Sunny, Cloudy, Rainy)},$$

solve $T^{\mathsf{T}}\boldsymbol{P} = \boldsymbol{P}$ and $P_S + P_C + P_R = 1$ to obtain

$$\boldsymbol{P} \approx \begin{pmatrix} 0.4545 \\ 0.3182 \\ 0.2273 \end{pmatrix},$$

so in the long run roughly 45% of the days are Sunny, 32% Cloudy, and 23% Rainy.

## 13.2 Detailed Balance

Directly solving (13.1.1) can be cumbersome for large or structured chains. A widely used *sufficient*[1] condition for stationarity is *detailed balance* (also called *reversibility*).

### 13.2.1 Definition

A distribution $\boldsymbol{P}$ and transition matrix $T$ *satisfy detailed balance* if

$$P_i\,T_{ij} = P_j\,T_{ji} \quad \text{for all } i, j \in \{1, \dots, m\}. \tag{13.2.1}$$

Equation (13.2.1) states that, in equilibrium, the probability flow from state $i$ to state $j$ is exactly cancelled by the reverse flow.

### 13.2.2 Detailed Balance Implies Stationarity

**Theorem 13.2.1.** *If $\boldsymbol{P}$ satisfies (13.2.1) with $T$, then $\boldsymbol{P}$ is a stationary distribution; i.e. $T^{\mathsf{T}}\boldsymbol{P} = \boldsymbol{P}$.*

*Proof.* For a fixed state $i$,

$$\left(T^{\mathsf{T}}\boldsymbol{P}\right)_i = \sum_{j=1}^{m} T_{ji}\,P_j.$$

Apply (13.2.1) : $T_{ij}P_i = P_jT_{ji}$. Hence

$$\sum_{j=1}^{m} T_{ji}P_j = \sum_{j=1}^{m} T_{ij}P_i = P_i \sum_{j=1}^{m} T_{ij}.$$

Because $T$ has the meaning of a transition probability, each *row* sums to unity, $\sum_{j=1}^{m} T_{ij} = 1$, so we obtain

$$\left(T^{\mathsf{T}}\boldsymbol{P}\right)_i = P_i \cdot 1 = P_i.$$

Since this holds for every $i$, we conclude $T^{\mathsf{T}}\boldsymbol{P} = \boldsymbol{P}$. □

### 13.2.3 Example: Lazy Symmetric Random Walk on a Ring

Let $m$ sites $\{0, 1, \dots, m-1\}$ be arranged on a circle (indices $\bmod m$). Define a "lazy" nearest-neighbour walk by

$$T_{ij} \;=\; \begin{cases} \tfrac{1}{2}, & j = i, \\ \tfrac{1}{4}, & j \equiv i \pm 1 \pmod{m}, \\ 0, & \text{otherwise.} \end{cases}$$

[1] It is not necessary: there exist stationary, non-reversible chains.

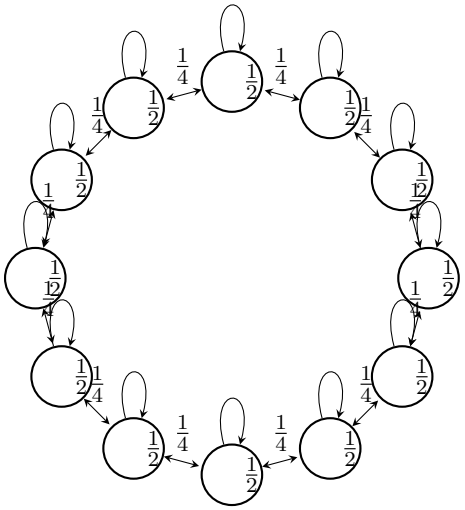
The idea of this transition probability is that at each step the walker can either stay where it is (with probability 1/2) or go on its left or right, with probability 1/4, respectively. Since everything is uniform (there is no privileged site in the ring) we will verify that the uniform distribution $P_i = \frac{1}{m}$ for all $i$ is stationary, i.e. $T^{\mathsf{T}}\,\boldsymbol{P} = \boldsymbol{P}$.

**Verification.** For each fixed $i$,

$$\big(T^{\mathsf{T}}P\big)_i = \sum_{j=0}^{m-1} T_{ji}\,P_j = \frac{1}{m}\big[T_{i,i} + T_{i-1,i} + T_{i+1,i}\big] = \frac{1}{m}\Big(\tfrac{1}{2} + \tfrac{1}{4} + \tfrac{1}{4}\Big) = \frac{1}{m} = P_i.$$

Hence $T^{\mathsf{T}}P = P$, so $P$ is indeed stationary.

**Ergodicity.** Irreducibility is immediate (every site can reach every other by steps of $\pm 1$), and the "stay-put" probability 1/2 breaks the period to 1, making the chain aperiodic.



Lazy symmetric random walk on a 12-site ring

Figure 13.2.1: Undirected edges denote $T(i \to j) = T(j \to i) = 1/4$; loops denote $T(i \to i) = 1/2$.

# Chapter 14

# The Metropolis-Hastings Algorithm

The key challenge that the Metropolis-Hastings algorithm addresses is sampling from complex probability distributions, especially in high-dimensional spaces. This is particularly useful in statistical mechanics, where we often need to sample from the Boltzmann distribution:

$$P(x) = \frac{1}{Z} e^{-\beta E(x)} \tag{14.0.1}$$

where $Z$ is the partition function, $\beta = 1/(k_B T)$, and $E(x)$ is the energy of state $x$. Here $x$ is typically a high dimensional variable collecting the degrees of freedom of our system. For example, it can be a collection of positions for a system in continuous space, or a collection of spin variables, for a discrete magnetic system. Because of this high dimensionality, direct sampling from this distribution is often infeasible due to the unknown normalization constant $Z$. In fact, the problem of computing $Z$ is the central problem of Statistical Physics.

In the previous Chapter we saw how detailed balance guarantees that a distribution $\boldsymbol{P}$ is stationary under a Markov transition. Here we develop the Metropolis–Hastings algorithm, which constructs such a transition matrix explicitly so that one can *sample* from an arbitrary target distribution. This is the heart of Markov Chain Monte Carlo (MCMC), bridging the gap between pure Monte Carlo integration and Markov processes.

## 14.1 From Monte Carlo to Markov Chains

As discussed previously, Monte Carlo methods approximate integrals or expectations

$$\mathbb{E}_P[O] \;=\; \sum_{i \in S} O(i)\, P_i$$

by drawing *independent* samples $i^{(1)}, \ldots, i^{(N)} \sim P$ and computing

$$\frac{1}{N} \sum_{k=1}^{N} O\big(i^{(k)}\big).$$

However, when $P$ is known only up to a normalization constant (as in statistical physics), direct sampling is infeasible. The Metropolis–Hastings method cleverly uses a Markov chain whose *transitions* depend only on ratios of $P_i$, thereby avoiding the unknown normalizing constant. Once the chain equilibrates, its *dependent* samples can still be used for Monte Carlo estimates, often far more efficiently in high dimensions.

## 14.2 Notation and Goals

Let

$$S = \{1, 2, \ldots, m\}, \quad \boldsymbol{P} = (P_1, \ldots, P_m)^{\mathsf{T}}, \quad P_i \geq 0, \; \sum_i P_i = 1,$$

be the discrete target distribution. We choose any proposal matrix

$$Q_{ij}, \quad Q_{ij} \geq 0, \; \sum_j Q_{ij} = 1,$$

that is easy to sample from (e.g. local moves on a lattice, flipping spins for the Ising model, etc.). Our aim is to turn $Q$ into a Markov transition matrix $T$ satisfying detailed balance with respect to $\boldsymbol{P}$.

## 14.3 The Metropolis–Hastings Procedure

1. **Initialize.** Set $X_0 = i_0 \in S$ arbitrarily.
2. **For** $n = 0, 1, 2, \ldots$**:**
   (a) **Propose:** draw $j \sim Q_{i \to j}$ from the current state $i = X_n$.
   (b) **Compute acceptance:**
   $$\alpha_{ij} \;=\; \min\Bigl(1, \; \frac{P_j \, Q_{ji}}{P_i \, Q_{ij}}\Bigr).$$
   This ratio involves only $P_j/P_i$, so any normalizing constant cancels.
   (c) **Accept or reject:** draw $u \sim \mathrm{Unif}[0, 1]$ and set
   $$X_{n+1} = \begin{cases} j, & u < \alpha_{ij}, \\ i, & \text{otherwise.} \end{cases}$$
3. **Repeat** until the chain has mixed ("burn-in") and thereafter collect samples $\{X_n\}$.

Thus each step consists of a *cheap* proposal plus an *inexpensive* accept/reject decision. Over time, the chain concentrates on high-probability regions of $P$.

### 14.3.1 Resulting Transition Matrix and the Diagonal Entry

From the accept/reject rule in the algorithm one reads off directly that, for $j \neq i$,

$$T_{ij} = \Pr\{\text{propose } j \mid i\}\ \Pr\{\text{accept}\} = Q_{ij}\,\alpha_{ij}.$$

Thus $T_{ij}$ is simply the probability of proposing a jump $i \to j$ and then accepting it.

The case $j = i$ requires a bit more care: one can remain in state $i$ either by proposing to stay and accepting, or by proposing some other $j \neq i$ and then rejecting. In full,

$$T_{ii} = \underbrace{Q_{ii}\,\alpha_{ii}}_{\substack{\text{proposal to stay}\\ \text{and accept}}} + \sum_{j \neq i} \underbrace{Q_{ij}\,(1-\alpha_{ij})}_{\substack{\text{proposal to } j\\ \text{and reject}}}.$$

By definition one has $\alpha_{ii} = 1$ (always accept a "proposal" to stay), so

$$T_{ii} = Q_{ii} + \sum_{j \neq i} Q_{ij}\,(1-\alpha_{ij}) = 1 \;-\; \sum_{j \neq i} Q_{ij}\,\alpha_{ij},$$

where in the last step we used $\sum_j Q_{ij} = 1$.

Hence the compact definition for the transition matrix

$$T_{ij} = \begin{cases} Q_{ij}\,\alpha_{ij}, & j \neq i, \\ 1 - \sum_{k \neq i} Q_{ik}\,\alpha_{ik}, & j = i \end{cases}$$

automatically collects *all* ways to remain in $i$.

## 14.4 Detailed Balance and Stationarity

We verify that $T$ satisfies *detailed balance* with respect to $P$, i.e.

$$P_i\,T_{ij} = P_j\,T_{ji}, \quad \forall\, i, j \in S.$$

Indeed, for $i \neq j$:

$$P_i\,Q_{ij}\,\alpha_{ij} = \min\big(P_i Q_{ij}, P_j Q_{ji}\big) = P_j\,Q_{ji}\,\alpha_{ji}.$$

Hence $T^t P = P$, making $P$ the unique stationary distribution when the chain is irreducible and aperiodic. This implies that the Metropolis-Hastings algorithm has as stationary distribution an arbitrary distribution $P$, thus the resulting Markov Chain samples are distributed according to $P$, which was our original goal.

## 14.5 Choice of Proposal and Ergodicity

**Symmetric proposals.** If $Q_{ij} = Q_{ji}$ (e.g. random walk proposals), then

$$\alpha_{ij} = \min\bigl(1, \tfrac{P_j}{P_i}\bigr)$$

depends only on the ratio of target densities. This version of the algorithm is the one originally published by Arianna Rosenbluth, Marshall Rosenbluth, Augusta Teller, and Edward Teller in 1953. It is historically known as the Metropolis algorithm. In 1970, Hastings extended the original algorithm to the more general case of non-symmetric proposals.

**Irreducibility and aperiodicity.** To ensure that $X_n$ converges in distribution to $P$, the chain must be:

- *Irreducible*: every state reachable from every other in finitely many steps.
- *Aperiodic*: not confined to cycles (e.g. allow self-transitions by ensuring $T_{ii} > 0$ or partial rejection).

These conditions are typically met by common proposals (e.g. Gaussian or uniform jumps).

## 14.6 Averages and Estimating Error Bars

When using the Metropolis algorithm to sample from a probability distribution $P(x)$, we often want to estimate the expectation value of some observable $O(x)$:

$$\langle O \rangle = \sum_{x \in S} O(x) P(x). \tag{14.6.1}$$

The Metropolis algorithm allows us to estimate this expectation value by generating a Markov chain of states $\{X_t\}$ that converges to the desired distribution $P(x)$. We then estimate $\langle O \rangle$ as:

$$\bar{O} = \frac{1}{T} \sum_{t=1}^{T} O(X_t), \tag{14.6.2}$$

where $T$ is the total number of steps in the Markov chain (after discarding the burn-in period, namely a few initial steps in which the Markov Chain has not reached a stationary state yet).

To estimate the error bars on $\bar{O}$, we can use the method of independent Markov chains. This approach is straightforward and avoids the complications of dealing with autocorrelations within a single chain. Here's how to proceed:

- **Run Multiple Independent Chains**:

  - Perform $M$ independent Metropolis simulations, each starting from a different initial state and using completely independent random numbers for the transition probabilities and for the acceptance probabilities.
  - For each chain $i$, calculate the average $\bar{O}_i = \frac{1}{T}\sum_{t=1}^{T} O(X_t^{(i)})$, where $X_t^{(i)}$ is the state at step $t$ in chain $i$.

- **Calculate the Overall Average**:

$$\bar{O} = \frac{1}{M}\sum_{i=1}^{M} \bar{O}_i \tag{14.6.3}$$

- **Estimate the Standard Error** of the Mean:

$$\sigma_{\bar{O}} = \sqrt{\frac{1}{M(M-1)}\sum_{i=1}^{M}(\bar{O}_i - \bar{O})^2} \tag{14.6.4}$$

- **Report the Result**:

$$\langle O \rangle \approx \bar{O} \pm \sigma_{\bar{O}} \tag{14.6.5}$$

This method treats each chain's average as an independent estimate of $\langle O \rangle$, which is valid if the chains are truly independent and have been run for a sufficiently long time to converge to the target distribution $\pi(x)$. In this respect, one has to pay attention to the fact that the initial samples of each chain may not be representative of the target distribution, especially if the starting point is in a low-probability region. It's common to discard an initial set of samples, known as the *burn-in* period. The averages above are then compute on the subset of samples that exclude the burn-in period.