Hidden Markov Models (HMMs)

From Probability Foundations to Advanced Theory

Donald (Donnie)

Table of contents

1. Hidden Markov Models (HMMs)

From Probability Foundations to Advanced Theory

Welcome to the **Hidden Markov Models Course**.

This site presents a **calm, rigorous, graduate-level treatment of HMMs**:

- From measure-theoretic probability and Markov chains
- Through inference algorithms (forward–backward, Viterbi, EM)
- To asymptotic theory and advanced variants (switching, nonparametric, state-space models)

> **i** Start learning
>
> Start with foundations View full curriculum
> This course is designed for students who want **proofs and derivations**, not just code snippets.
> Use the buttons above to either dive straight into the notes or skim the overall structure first.

1.1. Course modules at a glance

1.1.1. Module A – Foundations (Sections 0–1)

Mathematical background for a rigorous HMM course:

- Probability spaces, conditional expectations, basic measure-theoretic language.
- Stochastic matrices, Perron–Frobenius theory, spectral gap and mixing.
- Finite-state Markov chains: ergodicity, stationary laws, reversibility, and non-homogeneous chains.

1.1.2. Module B – Building HMMs (Sections 2–3)

Construction of HMMs as probabilistic graphical models:

- Conditional independence structure and joint factorization of $S_{1:T}, Y_{1:T}$.
- Observation models: discrete, continuous, and exponential-family emissions.
- Formal HMM definition $(\delta, \Gamma, f_1, \dots, f_K)$ and likelihood in matrix form.

1.1.3. Module C – Inference Algorithms (Section 4)

Core algorithms for posterior computation and decoding:

- Forward filtering and numerically stable log / scaled implementations.
- Forward–backward smoothing and pairwise state probabilities.
- Viterbi decoding, dynamic programming optimality, and max-product semiring.

1.1.4. Module D – Parameter Estimation & Identifiability (Section 5)

How to fit HMMs from data:

- Maximum likelihood estimation and non-convex log-likelihood geometry.
- EM / Baum–Welch as coordinate ascent on an evidence lower bound.
- Identifiability up to label switching and structural pathologies.

1.1.5. Module E – Statistical Theory & Advanced Models (Sections 6–9)

Asymptotics and extensions beyond basic finite-state HMMs:

- Consistency and asymptotic normality of the MLE; Fisher information for dependent data.
- Continuous-state / state-space models and the Kalman filter as an HMM.
- Nonparametric and infinite-state HMMs; online and decision-theoretic perspectives.

1.1.6. Module F – Applications & Proof-Based Problems (Sections 10–11)

Connecting theory to practice and consolidating understanding:

- Applications in speech recognition, bioinformatics, finance, epidemiology, and more.
- Proof-based problem sets covering Markov chains, inference algorithms, EM, identifiability, and asymptotics.

1.2. Get Started with HMMs Today!

Start with foundations View full curriculum

Use the **sidebar** to jump directly to individual sections (0–11), or read them linearly as a graduate course. Mathematics is rendered directly in the browser, and all derivations are written to be compatible with the notation in Zucchini, MacDonald & Langrock and the more theoretical treatments of Capp'e–Moulines–Ryd'en and Douc–Moulines–Stoffer.

Part I.

Overview

2. Hidden Markov Models (HMMs): A Rigorous, Mathematically Heavy Course

This project is a **full, proof-oriented course on Hidden Markov Models (HMMs)**, designed at the level of a serious graduate or early PhD sequence.

The course emphasizes:

- **Probability theory and stochastic processes** (measure-theoretic where needed)
- **Inference and algorithms** (forward–backward, Viterbi, EM/Baum–Welch) with **full derivations and proofs of correctness**
- **Statistical theory** (consistency, asymptotic normality, identifiability)
- **Advanced variants** (continuous-state, nonparametric, switching models)

The materials are organized into **12 sections (0–11)**. Each section lives in its own directory, with a dedicated `README.md` containing **detailed notes, theorems, and proof sketches**.

2.1. Primary References (Used for Notation and Examples)

The exposition and notation lean heavily on:

- **Zucchini, MacDonald, Langrock** – *Hidden Markov Models for Time Series: An Introduction Using R* (2nd ed.).
  This is the **main guiding reference** for finite-state HMMs, likelihoods, algorithms, and many examples.
- **Rabiner (1989)** – *A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition.*
  Classic algorithmic exposition (forward–backward, Viterbi, Baum–Welch).
- **Cappé, Moulines, Rydén (2005)** – *Inference in Hidden Markov Models.*
  Deep, rigorous treatment of HMM inference and statistical properties.
- **Douc, Moulines, Stoffer (2014)** – *Nonlinear Time Series: Theory, Methods and Applications.*
  Asymptotic theory and ergodic properties for dependent data, including HMMs.
- **Murphy (2012)** – *Machine Learning: A Probabilistic Perspective.*
  Broad probabilistic graphical model framing.

Unless otherwise noted, **notation follows Zucchini et al.** where feasible:

- Hidden state process: $(S_t)_{t \geq 1}$, taking values in a finite set $\{1, \dots, K\}$

- Observation process: $(Y_t)_{t\geq 1}$
- Initial distribution: $\delta = (\delta_i)_{i=1}^{K}$
- Transition probability matrix: $\Gamma = (\gamma_{ij})_{i,j=1}^{K}$
- State-dependent (emission) densities or pmfs: $f_i(\cdot)$ for state i

---

2.2. Course Structure (Section Index)

Each bullet links to a folder containing a **section-specific `README.md`**.

- **0. Mathematical Prerequisites**
  Measure-theoretic probability (light but precise), linear algebra and spectral theory for stochastic matrices, convexity and information geometry (KL divergence as a Bregman divergence).

- **1. Markov Chains (Fully Rigorous)**
  Finite-state Markov chains, Chapman–Kolmogorov equations, stationary and invariant distributions, reversibility, ergodic theory (irreducibility, aperiodicity, mixing times, spectral gaps), and non-homogeneous chains.

- **2. Observation Models and Emission Processes**
  Graphical model formulation of HMMs, conditional independence structure, factorization of joint distributions, discrete/continuous/exponential-family emissions, and identifiability issues.

- **3. Hidden Markov Models: Formal Definition**
  Generative definition of HMMs, formal state and observation spaces, initial distribution, transition kernel, emission kernel, and rigorous derivation of the joint and marginal likelihood.

- **4. Inference in HMMs (Core Algorithms)**
  Filtering (forward algorithm), smoothing (forward–backward), and decoding (Viterbi). Includes dynamic programming derivations, correctness proofs, and numerical stability considerations.

- **5. Parameter Estimation**
  Maximum likelihood estimation, EM/Baum–Welch algorithm (as coordinate ascent on an evidence lower bound), monotonicity and convergence guarantees, and identifiability theory.

- **6. Asymptotics and Statistical Theory**
  Consistency and asymptotic normality of MLE in ergodic HMMs, pseudo-true parameters under misspecification, Fisher information for dependent data.

- **7. Non-Standard and Advanced HMMs**
  Continuous-state HMMs (including linear Gaussian / Kalman models), nonparametric HMMs (e.g. Dirichlet process HMMs), and switching state-space models.

- **8. Computational and Numerical Issues**
  Scaling and log-domain implementations, underflow and overflow analysis, complexity of exact inference (time and space), and approximate methods.

- **9. Alternative Foundations**
  Online and distribution-free perspectives, prediction with expert-advice style losses, regret bounds for HMM-like models, decision-theoretic framing via POMDPs.
- **10. Applications**
  Full mathematical mapping of real applications: speech recognition, bioinformatics, finance, epidemiology, and more, always phrased as precise HMMs.
- **11. Proof-Based Problem Sets**
  Collections of theorem-level exercises: proving algorithm correctness, constructing counterexamples, identifiability and stability proofs, and asymptotic bounds.

2.3. How to Use This Course

- **Read Sections 0–1 carefully** if your background in probability or Markov chains is not fully measure-theoretic.
- **Work through the proofs** in Sections 3–5; they are central to a deep understanding of HMMs. Zucchini et al. provide many of the key derivations, which are expanded here.
- **Use Sections 6–9** as advanced material or for a second pass when you care about asymptotics, nonparametric models, or decision-theoretic views.
- **Attempt the problem sets in Section 11** as if they were exam or qualifying questions.

Roughly:

- **70%** of the course is probability and inference theory
- **20%** is algorithms with correctness proofs
- **10%** is applications and modeling case studies

2.4. Deployment and Final Site Build

To build and deploy the course website (a Quarto book with output in `_site/`):

- **1. Render the full site locally**

  ```
  quarto render
  ```

  This generates the static HTML site into the `_site/` directory as configured in `_quarto.yml`.
- **2. Preview locally (optional)**

  ```
  quarto preview
  ```

  This starts a local web server so you can inspect the site before publishing.

- **3. Deploy to GitHub Pages (recommended if using GitHub)**

  From the project root:

  ```
  quarto publish gh-pages
  ```

  This will:

  - Build the site
  - Push the rendered `_site/` contents to the `gh-pages` branch
  - Configure it for GitHub Pages hosting

- **4. Deploy to any static host (Netlify, Vercel, custom server)**

  - Configure your host to use the project root as the build directory
  - Set the **build command** to:

    ```
    quarto render
    ```

  - Set the **publish directory** (or equivalent) to:

    ```
    _site
    ```

  Any static host that can serve a folder of HTML/JS/CSS files can use the contents of `_site/` as the final deployed site.

Part II.

Foundations

3. Section 0 – Mathematical Prerequisites for Hidden Markov Models

This section collects the **mathematical foundations** required for a rigorous treatment of Hidden Markov Models (HMMs). The goal is **not** to teach full measure-theoretic probability from scratch, but to make precise the pieces that will be used repeatedly later.

Throughout, we aim to be compatible with the notation and level of **Zucchini, MacDonald, Langrock** ("Zucchini et al.") while pushing the theory somewhat further when needed for Sections 4–6.

3.1. 0.1 Measure-Theoretic Probability (Light but Precise)

3.1.1. 0.1.1 Probability Spaces

A **probability space** is a triple $(\Omega, \mathcal{F}, \mathbb{P})$ where

- Ω is the **sample space** (set of outcomes);
- $\mathcal{F} \subseteq 2^\Omega$ is a **σ-algebra** of events (closed under complements and countable unions);
- $\mathbb{P} : \mathcal{F} \to [0, 1]$ is a **probability measure** with $\mathbb{P}(\Omega) = 1$ and countable additivity.

For HMMs we usually work with products of measurable spaces, e.g. sequences of states and observations. The relevant product σ-algebras and measures are:

- For a measurable space $(S, \mathcal{S})$, the **countable product** $(S^{\mathbb{N}}, \mathcal{S}^{\otimes\mathbb{N}})$ is defined via the smallest σ-algebra making all coordinate projections measurable.
- For a Markov chain $(S_t)_{t\geq 1}$, the joint law of the whole sequence lives on such a product space.

In finite-state HMMs, $S = \{1, \dots, K\}$ with the **discrete σ-algebra** (all subsets), so measurability is trivial; nevertheless, the measure-theoretic formulation clarifies **conditional expectations** and **ergodic theorems** later.

3.1.2. 0.1.2 Random Variables and Distributions

A **random variable** with values in a measurable space $(S, \mathcal{S})$ is a measurable map

$$X : (\Omega, \mathcal{F}) \to (S, \mathcal{S}).$$

The **distribution** (or law) of X is the pushforward measure $\mathbb{P}_X$ on $(S, \mathcal{S})$:

$$\mathbb{P}_X(A) = \mathbb{P}(X \in A), \quad A \in \mathcal{S}.$$

In HMMs we will consider random variables S_t (hidden states) and Y_t (observations). Their joint distribution factorizes in a special way due to the **Markov property** and **conditional independence**, which we will formalize later.

3.1.3. 0.1.3 Expectation and Conditional Expectation

For an integrable real-valued random variable X, its **expectation** is

$$\mathbb{E}[X] = \int_\Omega X(\omega)\, \mathbb{P}(d\omega),$$

or equivalently, if X takes values in $\mathbb{R}$ with distribution $\mu = \mathbb{P}_X$,

$$\mathbb{E}[X] = \int_{\mathbb{R}} x\, \mu(dx).$$

For a sub-σ-algebra $\mathcal{G} \subseteq \mathcal{F}$, the **conditional expectation** of X given $\mathcal{G}$ is a $\mathcal{G}$-measurable random variable $\mathbb{E}[X \mid \mathcal{G}]$ such that

$$\int_G \mathbb{E}[X \mid \mathcal{G}]\, d\mathbb{P} = \int_G X\, d\mathbb{P}, \quad \forall G \in \mathcal{G}.$$

Key properties (used constantly in HMM derivations):

- **Linearity:** $\mathbb{E}[aX + bY \mid \mathcal{G}] = a\,\mathbb{E}[X \mid \mathcal{G}] + b\,\mathbb{E}[Y \mid \mathcal{G}]$.
- **Tower property:** If $\mathcal{H} \subseteq \mathcal{G} \subseteq \mathcal{F}$, then

$$\mathbb{E}[\mathbb{E}[X \mid \mathcal{G}] \mid \mathcal{H}] = \mathbb{E}[X \mid \mathcal{H}].$$

- **Taking out what is known:** If Z is $\mathcal{G}$-measurable and integrable,

$$\mathbb{E}[ZX \mid \mathcal{G}] = Z\,\mathbb{E}[X \mid \mathcal{G}].$$

In HMMs, filtering and smoothing can be viewed as **computing conditional expectations** like $\mathbb{E}[g(S_t) \mid Y_{1:T}]$ for suitable functions g. The forward–backward algorithms are efficient implementations of these operations.

3.1.4. 0.1.4 Regular Conditional Probabilities

Given random variables X and Y on a probability space, a **regular conditional probability** of X given $Y = y$ is a family of probability measures $\{\mathbb{P}(X \in \cdot \mid Y = y)\}$ such that

- For each measurable A, the map $y \mapsto \mathbb{P}(X \in A \mid Y = y)$ is measurable;
- For each measurable B,

$$\mathbb{P}(X \in B, Y \in C) = \int_C \mathbb{P}(X \in B \mid Y = y)\, \mathbb{P}_Y(dy).$$

On **standard Borel spaces** (Polish spaces with their Borel σ-algebra), regular conditional probabilities always exist and are unique up to $\mathbb{P}_Y$-null sets. This justifies writing objects like

$$\mathbb{P}(S_t = i \mid Y_{1:T} = y_{1:T})$$

rigorously, which is what the forward–backward algorithms compute.

3.1.5. 0.1.5 Modes of Convergence

We briefly recall three notions of convergence for a sequence of random variables (X_n):

- **Almost sure (a.s.) convergence:** $X_n \to X$ a.s. if

$$\mathbb{P}\big(\{\omega : X_n(\omega) \to X(\omega)\}\big) = 1.$$

- **Convergence in probability:** $X_n \to X$ in probability if, for all $\varepsilon > 0$,

$$\lim_{n \to \infty} \mathbb{P}(|X_n - X| > \varepsilon) = 0.$$

- **L^p convergence:** $X_n \to X$ in L^p (for $p \geq 1$) if

$$\lim_{n \to \infty} \mathbb{E}[|X_n - X|^p] = 0.$$

For asymptotic theory in HMMs (Section 6), we will need **laws of large numbers** and **central limit theorems** for functionals of an ergodic Markov chain. These are typically stated in terms of convergence in probability or distribution, and proved using almost sure convergence plus dominated convergence.

3.2. 0.2 Linear Algebra and Spectral Theory

3.2.1. 0.2.1 Probability Vectors and the Simplex

For a finite state space of size K, a **probability vector** is

$$\mu = (\mu_1, \dots, \mu_K)^\top, \quad \mu_i \geq 0, \quad \sum_{i=1}^{K} \mu_i = 1.$$

The set of all such vectors is the **probability simplex**

$$\Delta^{K-1} = \Big\{\mu \in \mathbb{R}^K : \mu_i \geq 0, \sum_i \mu_i = 1\Big\}.$$

We measure distances on Δ^{K-1} using norms:

- ℓ^1 **norm:** $\|\mu - \nu\|_1 = \sum_i |\mu_i - \nu_i|$ (twice the total variation distance);
- ℓ^2 **norm:** $\|\mu - \nu\|_2 = (\sum_i (\mu_i - \nu_i)^2)^{1/2}$.

Both will appear in mixing-time and stability results for Markov chains and filters.

3.2.2. 0.2.2 Stochastic Matrices

A **row-stochastic matrix** is a $K \times K$ matrix $\Gamma = (\gamma_{ij})$ with

$$\gamma_{ij} \geq 0, \quad \sum_{j=1}^{K} \gamma_{ij} = 1 \quad \text{for all } i.$$

In finite-state HMMs (following Zucchini et al.), Γ denotes the **transition matrix** of the hidden Markov chain (S_t):

$$\gamma_{ij} = \mathbb{P}(S_{t+1} = j \mid S_t = i).$$

Given a probability vector μ, the product $\mu^\top \Gamma$ is again a probability vector, representing the distribution of S_{t+1} if μ is the distribution of S_t.

3.2.3. 0.2.3 Perron–Frobenius Theory

For a **non-negative matrix** $A \in \mathbb{R}^{K \times K}$ (i.e. $A_{ij} \geq 0$), the Perron–Frobenius theorem gives powerful spectral properties. In particular, if A is **irreducible**, then

- There exists a **positive eigenvalue** $\rho(A) > 0$ (the spectral radius) with a corresponding **positive eigenvector** $v > 0$.
- $\rho(A)$ is **simple** (algebraic multiplicity 1), and no other eigenvector with non-negative entries exists for a different eigenvalue.

For a **stochastic matrix** Γ:

- Its spectral radius satisfies $\rho(\Gamma) = 1$, since $\Gamma \mathbf{1} = \mathbf{1}$.
- If Γ is irreducible and aperiodic, the **left eigenvector** corresponding to eigenvalue 1, normalized to sum to 1, is the **unique stationary distribution** π:

$$\pi^\top \Gamma = \pi^\top.$$

This provides the spectral foundation for **ergodicity** of finite-state Markov chains, and later for stability of HMM filters.

3.2.4. 0.2.4 Spectral Gap and Convergence Rates

Let the eigenvalues of a stochastic matrix Γ be ordered as

$$1 = \lambda_1 > |\lambda_2| \geq ... \geq |\lambda_K|.$$

The **spectral gap** is

$$\gamma := 1 - |\lambda_2|.$$

For many chains (especially reversible ones), the convergence of $\mu_0^\top \Gamma^t$ to the stationary distribution $\pi^\top$ in ℓ^2 or total variation can be bounded in terms of γ. Roughly,

$$\|\mu_0^\top \Gamma^t - \pi^\top\|_2 \leq C(1-\gamma)^t.$$

More precise inequalities follow from the spectral decomposition of Γ and, in the reversible case, from its self-adjointness in $L^2(\pi)$.

These ideas will underpin **mixing-time** and **filter stability** results (Sections 1.2 and 4.1).

3.3. 0.3 Optimization and Information Geometry

3.3.1. 0.3.1 Convexity on the Probability Simplex

A function $f : \Delta^{K-1} \to \mathbb{R}$ is **convex** if

$$f(\theta\mu + (1-\theta)\nu) \leq \theta f(\mu) + (1-\theta) f(\nu)$$

for all $\mu, \nu \in \Delta^{K-1}$ and $\theta \in [0,1]$.

Many information-theoretic functionals are convex or strictly convex on Δ^{K-1}. Examples:

- Negative entropy $H(\mu) = -\sum_i \mu_i \log \mu_i$ is **strictly concave**;
- The **Kullback–Leibler divergence** (KL) is jointly convex in (p, q).

Convexity is central in understanding **EM updates**, **variational approximations**, and the geometry of the **log-likelihood surface** in HMMs.

3.3.2. 0.3.2 Kullback–Leibler Divergence as a Bregman Divergence

For two discrete distributions $p, q \in \Delta^{K-1}$ with full support ($p_i, q_i > 0$), the **KL divergence** is

$$\mathrm{KL}(p\|q) = \sum_{i=1}^{K} p_i \log \frac{p_i}{q_i}.$$

KL divergence can be written as a **Bregman divergence** associated with the **negative entropy** function

$$\phi(p) = \sum_i p_i \log p_i.$$

The Bregman divergence generated by ϕ is

$$D_\phi(p, q) = \phi(p) - \phi(q) - \langle \nabla\phi(q), p - q\rangle,$$

where $\langle \cdot, \cdot \rangle$ is the usual inner product on $\mathbb{R}^K$. A straightforward calculation shows

$$D_\phi(p, q) = \mathrm{KL}(p\|q).$$

This interpretation highlights several facts:

- $\mathrm{KL}(p\|q) \geq 0$ with equality iff $p = q$ (strict convexity of ϕ);
- KL is **asymmetric**, unlike a metric, which shapes the geometry of likelihood-based optimization.

In HMMs, KL divergence arises when analyzing **consistency** and **information projections**, and in understanding why the EM algorithm can be seen as **coordinate ascent on a lower bound** involving KL terms.

3.3.3. 0.3.3 Duality and Entropy-Regularized Problems

Given a convex function ϕ, its **convex conjugate** ϕ^* is

$$\phi^*(y) = \sup_x \{\langle x, y\rangle - \phi(x)\}.$$

For $\phi(p) = \sum_i p_i \log p_i$ (negative entropy), ϕ^* is the log-partition function

$$\phi^*(\eta) = \log \sum_i e^{\eta_i}.$$

This duality underlies the **exponential family** structure of many emission distributions (Section 2.2) and appears in **variational formulations** of inference in HMMs:

- Entropy-regularized objectives of the form

  $$\max_q \Big\{ \mathbb{E}_q[\log p(Y, S)] + H(q) \Big\}$$

  lead to exponential-family solutions for the optimal q.

In the context of Zucchini et al., this background explains why **log-sum-exp** expressions appear in marginal likelihoods and why certain optimization problems have tractable, closed-form updates.

3.4. 0.4 Summary and Connection to Later Sections

After this section, you should be comfortable with:

- **Probability spaces, random variables, and conditional expectations** in a measure-theoretic language;
- **Finite-dimensional linear algebra** for stochastic matrices, including Perron–Frobenius theory and spectral gaps;
- **Basic convex analysis** on probability simplices, and the interpretation of **KL divergence as a Bregman divergence**.

These tools will be used heavily in:

- **Section 1:** rigorous Markov chain theory (ergodicity, mixing);
- **Section 3–4:** derivation and correctness proofs of forward–backward and Viterbi algorithms;
- **Section 5–6:** EM algorithm analysis, identifiability, consistency, and asymptotic normality.

For a softer introduction, you may cross-reference:

- Zucchini et al., Chapters 1–2, for probabilistic notation and basic Markov chain ideas;
- Murphy (2012), Chapters 2–3, for probability and exponential families;
- Cappé, Moulines, Rydén (2005), Chapter 1, for a more advanced measure-theoretic setup.

4. Section 1 – Markov Chains (Fully Rigorous)

This section develops the **Markov chain theory** that underlies finite-state HMMs. We focus on:

- Finite-state **homogeneous Markov chains** and their invariant distributions;
- **Ergodic properties**: irreducibility, aperiodicity, mixing, spectral gap;
- A brief look at **non-homogeneous** chains, which appear in some generalized HMMs.

Zucchini et al. treat finite-state Markov chains at an applied level; here we give a more rigorous account compatible with their notation.

4.1. 1.1 Finite-State Markov Chains

4.1.1. 1.1.1 Definition and Transition Kernels

Let the **state space** be $E = \{1, \dots, K\}$. A stochastic process $(S_t)_{t \geq 1}$ with values in E is a (time-homogeneous) **Markov chain** with transition matrix $\Gamma = (\gamma_{ij})$ if

$$\mathbb{P}(S_{t+1} = j \mid S_1, \dots, S_t) = \mathbb{P}(S_{t+1} = j \mid S_t) = \gamma_{S_t j}, \quad \forall t \geq 1.$$

Equivalently, for any sequence $i_1, \dots, i_T$ in E, the joint probability is

$$\mathbb{P}(S_1 = i_1, \dots, S_T = i_T) = \delta_{i_1} \prod_{t=1}^{T-1} \gamma_{i_t i_{t+1}},$$

where $\delta = (\delta_i)$ is the **initial distribution** $\delta_i = \mathbb{P}(S_1 = i)$.

This is exactly the hidden-state dynamics that Zucchini et al. use to define finite-state HMMs; the HMM adds an **observation process** on top of this chain.

4.1.2. 1.1.2 Chapman–Kolmogorov Equations

Let $\Gamma^{(n)}$ denote the n-step transition matrix, with entries

$$\gamma_{ij}^{(n)} = \mathbb{P}(S_{t+n} = j \mid S_t = i).$$

Then the **Chapman–Kolmogorov equations** state that for all $m, n \geq 0$,

$$\Gamma^{(m+n)} = \Gamma^{(m)} \Gamma^{(n)}.$$

In particular, $\Gamma^{(n)} = \Gamma^n$ (the usual matrix power). This ties Markov chain evolution directly to the spectral properties of Γ.

4.1.3. 1.1.3 Stationary and Invariant Distributions

A probability vector $\pi \in \Delta^{K-1}$ is a **stationary distribution** for Γ if

$$\pi^\top \Gamma = \pi^\top.$$

Interpretation:

- If $S_1 \sim \pi$, then $S_t \sim \pi$ for all t; the chain is **in equilibrium**.
- If the chain is **irreducible and aperiodic**, π is **unique**, and the distribution of S_t converges to π for any initial distribution δ.

The existence and uniqueness of π are guaranteed by **Perron–Frobenius theory** (Section 0.2) for irreducible, aperiodic stochastic matrices.

4.1.4. 1.1.4 Reversibility and Detailed Balance

A Markov chain with transition matrix Γ and stationary distribution π is **reversible** if it satisfies the **detailed balance equations**

$$\pi_i \, \gamma_{ij} = \pi_j \, \gamma_{ji}, \quad \forall i, j.$$

Intuitively, under stationarity, the probability flow from i to j equals that from j to i.

Consequences:

- In the inner product space $L^2(\pi)$, Γ is **self-adjoint**:

  $$\langle f, \Gamma g \rangle_\pi = \langle \Gamma f, g \rangle_\pi$$

  for functions $f, g : E \to \mathbb{R}$, where $\langle f, g \rangle_\pi = \sum_i \pi_i f(i) g(i)$.
- Hence, the spectrum of Γ is **real**, and spectral analysis is particularly transparent.

In HMMs, even if the hidden chain is not assumed reversible, reversible chains are a useful class for **examples**, **counterexamples**, and **mixing-time calculations**.

4.2. 1.2 Ergodic Theory of Markov Chains

4.2.1. 1.2.1 Irreducibility and Communication Classes

For states $i, j \in E$, write $i \rightsquigarrow j$ if there exists $n \geq 0$ such that $\gamma_{ij}^{(n)} > 0$ (a path of positive probability from i to j). We say i **communicates with** j, written $i \leftrightarrow j$, if both $i \rightsquigarrow j$ and $j \rightsquigarrow i$ hold.

This is an equivalence relation, partitioning E into **communicating classes**. A chain is **irreducible** if it has a single communicating class (every state communicates with every other).

In HMMs, irreducibility of the hidden chain ensures that every state can eventually be reached from any other, which is important for:

- Existence and uniqueness of a stationary distribution;
- Identifiability and mixing assumptions in asymptotic theory (Section 6).

4.2.2. 1.2.2 Periodicity and Aperiodicity

The **period** of a state i is

$$\operatorname{per}(i) = \gcd\{n \geq 1 : \gamma_{ii}^{(n)} > 0\}.$$

In an irreducible chain, all states share the same period, so we can speak of **the** period of the chain. A chain is **aperiodic** if $\operatorname{per}(i) = 1$ for some (hence all) i.

Aperiodicity rules out deterministic cycles and is necessary for convergence of $\mathbb{P}(S_t = \cdot)$ to π in total variation.

4.2.3. 1.2.3 Ergodic Theorem for Finite-State Markov Chains

Let (S_t) be irreducible and aperiodic with stationary distribution π. Then for any bounded function $f : E \to \mathbb{R}$,

$$\frac{1}{T}\sum_{t=1}^{T} f(S_t) \xrightarrow[T\to\infty]{\text{a.s.}} \sum_{i=1}^{K} \pi_i f(i) =: \mathbb{E}_\pi[f(S)].$$

This is the **ergodic theorem**: time averages converge almost surely to space averages under π. It is a Markov-chain version of the **strong law of large numbers**.

In HMMs, ergodic theorems are used to prove **consistency of estimators** and to analyze limiting behavior of likelihoods per unit time.

4.2.4. 1.2.4 Mixing Times and Total Variation Distance

For a probability vector μ on E, the **total variation distance** to π is

$$\|\mu - \pi\|_{\mathrm{TV}} = \frac{1}{2}\sum_{i=1}^{K} |\mu_i - \pi_i|.$$

Let $\mu_t = \delta^\top \Gamma^t$ be the distribution of S_t starting from δ. The **mixing time** $t_{\mathrm{mix}}(\varepsilon)$ is

$$t_{\mathrm{mix}}(\varepsilon) = \min\Big\{t : \sup_{\delta} \|\mu_t - \pi\|_{\mathrm{TV}} \leq \varepsilon\Big\}.$$

In finite-state irreducible aperiodic chains, $t_{\mathrm{mix}}(\varepsilon) < \infty$ for all $\varepsilon > 0$. Spectral methods and coupling (next subsection) give quantitative bounds.

4.2.5. 1.2.5 Spectral Gap and Convergence Rates

Suppose the chain is reversible with respect to π, with eigenvalues of Γ ordered as

$$1 = \lambda_1 > \lambda_2 \geq ... \geq \lambda_K > -1.$$

The **spectral gap** is $\gamma = 1 - \lambda_2$. One can show (see e.g. books on Markov chain mixing) that

$$\|\mu_t - \pi\|_{\mathrm{TV}} \leq C\,(1-\gamma)^t$$

for some constant C depending on δ. Thus, a larger spectral gap implies faster convergence to stationarity.

In HMMs, these spectral-gap-based bounds transfer to **stability of the filtering distribution**: the distribution of S_t given observations becomes asymptotically independent of the initial distribution.

4.2.6. 1.2.6 Coupling Arguments (Sketch)

A powerful probabilistic technique for bounding mixing times is **coupling**: construct two copies of the chain, (S_t) and (S'_t), possibly dependent, such that

- Marginally, each evolves according to Γ;
- They eventually **coalesce**: $S_t = S'_t$ for all sufficiently large t.

Define the **coupling time**

$$T_c = \inf\{t \geq 0 : S_t = S'_t\}.$$

Then for any initial distributions δ, δ',

$$\|\mu_t - \mu'_t\|_{\mathrm{TV}} \leq \mathbb{P}(T_c > t).$$

Hence, controlling $\mathbb{P}(T_c > t)$ yields mixing bounds. The idea of coupling will reappear implicitly in **filter stability** results in HMMs.

4.3. 1.3 Non-Homogeneous Markov Chains

In some extensions of HMMs, the hidden state process may have **time-varying transitions**, represented by a sequence of stochastic matrices (Γ_t). Then

$$\mathbb{P}(S_{t+1} = j \mid S_t = i) = (\Gamma_t)_{ij}.$$

4.3.1. 1.3.1 Product of Time-Varying Kernels

Define the n-step transition kernel from time t to $t+n$ as

$$\Gamma_{t,t+n} = \Gamma_t \Gamma_{t+1} \cdots \Gamma_{t+n-1}.$$

The analog of Chapman–Kolmogorov holds in the obvious way:

$$\Gamma_{t,t+m+n} = \Gamma_{t,t+m} \Gamma_{t+m,t+m+n}.$$

4.3.2. 1.3.2 Stability Conditions

Without time-homogeneity, there may be **no stationary distribution**. Instead, one studies **stability** and **ergodicity** via conditions such as:

- Uniform **Doeblin conditions** (lower bounds on transition probabilities);
- **Dobrushin contraction coefficients** ensuring that products of kernels contract distances between probability distributions.

These ideas become particularly relevant when considering **non-stationary HMMs** or **online learning** settings (see Section 9).

4.4. 1.4 Connection to HMMs and Zucchini et al.

In Zucchini et al., the hidden process (S_t) of an HMM is always a **finite-state Markov chain** with transition matrix Γ and initial distribution δ. The properties introduced here feed directly into later sections:

- **Section 3:** Uses the Markov property to factorize the joint HMM likelihood;
- **Section 4:** Forward–backward and Viterbi algorithms exploit Γ as the transition kernel;
- **Section 6:** Ergodicity and mixing of (S_t) underpin **consistency** and **CLTs** for estimators.

For more detailed Markov chain theory in a measure-theoretic style, see:

- Cappé, Moulines, Rydén (2005), Chapters 1–2;
- Douc, Moulines, Stoffer (2014), Chapters 2–3.

Part III.

Model & Inference

5. Section 2 – Observation Models and Emission Processes

In an HMM, the hidden Markov chain $(S_t)_{t\geq 1}$ is not observed directly. Instead, we observe a process $(Y_t)_{t\geq 1}$, whose distribution is conditionally independent **given the hidden states**.

This section formalizes:

- The **graphical model** structure of HMMs;
- The **factorization** of the joint distribution;
- Classes of **emission distributions** (discrete, continuous, exponential family);
- Basic **identifiability** issues arising from emissions.

We follow the high-level view in Zucchini et al. (Chapter 2), but state the conditional independence structure more explicitly.

5.1. 2.1 Conditional Independence Structure

5.1.1. 2.1.1 Graphical Model Representation

An HMM with T time steps consists of:

- Hidden states $S_1, \dots, S_T$ forming a Markov chain on $\{1, \dots, K\}$;
- Observations $Y_1, \dots, Y_T$ taking values in some space $\mathcal{Y}$.

The **directed graphical model** has edges

- $S_t \to S_{t+1}$ (hidden Markov chain);
- $S_t \to Y_t$ (emission at each time).

The critical conditional independence assumptions are:

1. Given S_t, the observation Y_t is **independent of all other states and observations**:

$$Y_t \perp\!\!\!\perp \{S_s : s \neq t\}, \{Y_s : s \neq t\} \mid S_t.$$

2. The hidden chain is first-order Markov:

$$S_{t+1} \perp\!\!\!\perp \{S_1, \dots, S_{t-1}\} \mid S_t.$$

Together, these imply a specific **factorization** of the joint distribution.

5.1.2. 2.1.2 Factorization of the Joint Distribution

Let $s_{1:T} = (s_1, \dots, s_T)$ and $y_{1:T} = (y_1, \dots, y_T)$. The joint distribution of states and observations factorizes as

$$\mathbb{P}(S_{1:T} = s_{1:T}, Y_{1:T} = y_{1:T}) = \delta_{s_1} f_{s_1}(y_1) \prod_{t=2}^{T} \gamma_{s_{t-1}, s_t} f_{s_t}(y_t),$$

where

- $\delta = (\delta_i)$ is the initial distribution $\mathbb{P}(S_1 = i)$;
- $\Gamma = (\gamma_{ij})$ is the transition matrix $\mathbb{P}(S_t = j \mid S_{t-1} = i)$;
- $f_i(\cdot)$ is the **emission density or mass function** for state i.

This is the basic factorization that Zucchini et al. use throughout their book; it underlies all efficient algorithms (forward–backward, Viterbi, EM).

5.1.3. 2.1.3 d-Separation and Conditional Independences

The graphical structure immediately yields many conditional independences via **d-separation**:

- Given S_t, the past and future observations are conditionally independent:

$$Y_{1:t-1} \perp\!\!\!\perp Y_{t+1:T} \mid S_t.$$

- Given the full state sequence $S_{1:T}$, the observations are conditionally independent across time:

$$Y_t \perp\!\!\!\perp Y_s \mid S_{1:T}, \quad t \neq s.$$

- Given **all observations** $Y_{1:T}$, the hidden states form a **Markov random field** (an undirected chain), but conditional dependences become more complex.

Understanding these independences helps in designing **approximate inference algorithms** and **variational factorizations**.

5.2. 2.2 Emission Distributions

5.2.1. 2.2.1 Discrete Emissions

If Y_t takes values in a finite or countable set $\mathcal{Y} = \{1, \dots, M\}$, each state i has a probability mass function

$$\mathbb{P}(Y_t = y \mid S_t = i) = b_i(y), \quad y \in \mathcal{Y},$$

with $b_i(y) \geq 0$ and $\sum_y b_i(y) = 1$.

Collect b_i into an **emission matrix B** of size $K \times M$, where $B_{iy} = b_i(y)$. Discrete-emission HMMs are the classical setting in **speech recognition** and many applications in **bioinformatics**.

In Zucchini et al., discrete emissions appear in introductory examples and in categorical time series modeling.

5.2.2. 2.2.2 Continuous Emissions

If Y_t takes values in $\mathbb{R}^d$ (or a subset), each state i has a **density** (with respect to Lebesgue measure) $f_i(y)$, so that

$$\mathbb{P}(Y_t \in A \mid S_t = i) = \int_A f_i(y)\, dy.$$

Common parametric choices:

- **Gaussian emissions:** $f_i(y) = \mathcal{N}(y; \mu_i, \Sigma_i)$;
- **Mixtures of Gaussians:** to increase flexibility;
- Other **exponential family** densities (see next subsection).

Continuous-emission HMMs are heavily treated in Zucchini et al. for modeling **time series of real-valued measurements** (e.g. environmental data, financial returns).

5.2.3. 2.2.3 Exponential Family Emissions

Many emission models fall into the **exponential family**. A density (or mass function) $f(y; \eta)$ is in an exponential family if it can be written as

$$f(y; \eta) = h(y) \exp\{\langle \eta, T(y) \rangle - A(\eta)\},$$

where

- $T(y)$ is the vector of **sufficient statistics**;
- η is the **natural parameter**;
- $A(\eta)$ is the **log-partition function** ensuring normalization;
- $h(y)$ is the base measure or carrier density.

In an HMM with exponential-family emissions, each state i has its own natural parameter η_i, and thus its own emission distribution $f_i(y)$. This structure simplifies:

- Derivation of **EM (Baum–Welch) updates** for emission parameters;
- Computation of gradients and Fisher information.

The connection to **information geometry** (Section 0.3) arises because the log-partition function $A(\eta)$ is the **convex conjugate** of negative entropy, and KL divergence between two exponential-family members has a natural Bregman form.

5.2.4. 2.2.4 Identifiability Issues

Identifiability asks whether the parameter θ of an HMM (transition matrix, emissions, etc.) is uniquely determined by the distribution of $Y_{1:T}$ (for all T large enough), up to label permutations of the hidden states.

Even with rich emission families, several issues arise:

- **Label switching:** If we permute state indices, say swap states 1 and 2, and correspondingly permute rows/columns of Γ and emission parameters, the distribution of $Y_{1:T}$ is unchanged. Thus, identifiability is at best **up to permutation**.
- **Overlapping emissions:** If two states share identical emission distributions (e.g. $f_1 = f_2$) and transition rows, they may be **indistinguishable**.
- **Non-identifiability in mixtures:** In some cases, different combinations of transition probabilities and emission parameters can yield the same observed process distribution.

The formal theory of identifiability in HMMs is nontrivial (see Section 5.3 and references there). Zucchini et al. discuss practical implications: e.g., in estimation, one must be aware that state labels are arbitrary and that some parameter settings may be weakly identified.

5.3. 2.3 Observation Models in Practice (Zucchini et al.)

Zucchini et al. provide many concrete observation models:

- **Count data:** Poisson or negative binomial emissions for counts (e.g. number of events per time unit);
- **Continuous data:** Gaussian or t-distributed emissions for real-valued series;
- **Circular data:** von Mises or wrapped distributions for angles;
- **Multivariate data:** multivariate normal or copula-based constructions.

In each case, the key is to specify, for each state i, a parametric family

$$\{f_i(\cdot; \phi_i) : \phi_i \in \Phi_i\}$$

and then estimate ϕ_i jointly with δ and Γ (typically by maximum likelihood using EM).

5.4. 2.4 Summary and Outlook

By now you should understand:

- How the **conditional independence structure** of HMMs induces a specific **factorization** of the joint distribution;
- The role of **emission distributions** in shaping the model's expressiveness;
- Basic **identifiability concerns** arising from overlapping or non-distinct emissions.

These ideas feed directly into:

- **Section 3:** Formal definition of HMMs and likelihood factorization;
- **Section 4:** Algorithms for computing marginal and conditional distributions over states given observations;
- **Section 5:** Parameter estimation (MLE, EM) and identifiability theory.

For additional reading:

- Zucchini et al., Chapters 2–3 (construction of HMMs and emission models);
- Cappé, Moulines, Rydén (2005), Chapters 1–2 (measure-theoretic HMM definition and basic properties).

6. Section 3 – Hidden Markov Models: Formal Definition and Likelihood

We now give a **fully formal definition** of finite-state Hidden Markov Models (HMMs) and derive the **joint** and **marginal (observed)** likelihoods.

This section closely follows the notation of **Zucchini, MacDonald, Langrock**, while making all probabilistic assumptions explicit and preparing the ground for algorithmic and statistical analysis in later sections.

6.1. 3.1 Generative Definition of a Finite-State HMM

6.1.1. 3.1.1 Components of the Model

Fix:

- A finite **state space** $E = \{1, \dots, K\}$;
- An **observation space** $(\mathcal{Y}, \mathcal{B}_y)$, e.g. $\mathbb{R}^d$ with the Borel σ-algebra;
- An **initial distribution** $\delta = (\delta_i)_{i=1}^K$, a probability vector on E;
- A **transition matrix** $\Gamma = (\gamma_{ij})_{i,j=1}^K$ with

$$\gamma_{ij} = \mathbb{P}(S_{t+1} = j \mid S_t = i), \quad \sum_j \gamma_{ij} = 1;$$

- A collection of **emission distributions** $\{F_i : i \in E\}$ on $(\mathcal{Y}, \mathcal{B}_y)$, with densities f_i (with respect to a common dominating measure, often Lebesgue or counting measure).

6.1.2. 3.1.2 Hidden State Process

On a probability space $(\Omega, \mathcal{F}, \mathbb{P})$, define a stochastic process $(S_t)_{t \geq 1}$ with values in E such that

- $\mathbb{P}(S_1 = i) = \delta_i$;
- For all $t \geq 1$,

$$\mathbb{P}(S_{t+1} = j \mid S_1, \dots, S_t) = \mathbb{P}(S_{t+1} = j \mid S_t) = \gamma_{S_t j}.$$

Thus, (S_t) is a **time-homogeneous finite-state Markov chain** as in Section 1.

6.1.3. 3.1.3 Observation Process

Given the hidden process (S_t), define an observation process $(Y_t)_{t\geq 1}$ taking values in $\mathcal{Y}$ such that

- Conditional on $S_t = i$, Y_t is drawn from F_i with density f_i;
- Conditional on **all states**, observations are independent across time:

$$\mathbb{P}(Y_{1:T} \in A_{1:T} \mid S_{1:T} = s_{1:T}) = \prod_{t=1}^{T} F_{s_t}(A_t).$$

Equivalently, with densities,

$$\mathbb{P}(Y_{1:T} \in dy_{1:T} \mid S_{1:T} = s_{1:T}) = \prod_{t=1}^{T} f_{s_t}(y_t)\, dy_t.$$

The pair (S_t, Y_t) defines the **Hidden Markov Model**.

6.2. 3.2 Joint Likelihood Factorization

Fix a time horizon T. For a realizations $s_{1:T} \in E^T$ and $y_{1:T} \in \mathcal{Y}^T$, the joint density (or mass function) of $(S_{1:T}, Y_{1:T})$ is

$$\begin{aligned}
&\mathbb{P}(S_{1:T} = s_{1:T}, Y_{1:T} = y_{1:T}) \\
&= \mathbb{P}(S_1 = s_1)\, \mathbb{P}(Y_1 = y_1 \mid S_1 = s_1) \prod_{t=2}^{T} \mathbb{P}(S_t = s_t \mid S_{t-1} = s_{t-1})\, \mathbb{P}(Y_t = y_t \mid S_t = s_t) \\
&= \delta_{s_1} f_{s_1}(y_1) \prod_{t=2}^{T} \gamma_{s_{t-1}, s_t}\, f_{s_t}(y_t).
\end{aligned}$$

This is the fundamental factorization used throughout Zucchini et al. It mirrors Equation (2.1) in their book (up to notation differences).

The **complete-data log-likelihood** (if we knew the states) is

$$\log L_c(\delta, \Gamma, f; s_{1:T}, y_{1:T}) = \log \delta_{s_1} + \sum_{t=2}^{T} \log \gamma_{s_{t-1}, s_t} + \sum_{t=1}^{T} \log f_{s_t}(y_t).$$

This form is crucial for the **EM/Baum–Welch algorithm** (Section 5.2).

6.3. 3.3 Marginal Likelihood of the Observations

In practice, the states $S_{1:T}$ are unobserved. The **observed data likelihood** is the marginal of the joint distribution over all possible state sequences:

$$L(\delta, \Gamma, f; y_{1:T}) = \mathbb{P}(Y_{1:T} = y_{1:T}) = \sum_{s_{1:T} \in E^T} \mathbb{P}(S_{1:T} = s_{1:T}, Y_{1:T} = y_{1:T}).$$

Substituting the joint factorization,

$$L(\theta; y_{1:T}) = \sum_{s_{1:T}} \delta_{s_1} f_{s_1}(y_1) \prod_{t=2}^{T} \gamma_{s_{t-1}, s_t} f_{s_t}(y_t),$$

where θ denotes the collection of all parameters.

6.3.1. 3.3.1 Naïve Computation is Exponential

There are K^T **terms** in the sum over state sequences. Direct evaluation is computationally infeasible even for moderate T and K.

Example: with $K = 5$ states and $T = 100$, 5^{100} is astronomically large.

Thus, we need to exploit the **Markov and conditional independence structure** to compute this marginal efficiently. This leads to the **forward algorithm** (Section 4.1), which runs in $\mathcal{O}(K^2 T)$ time.

6.3.2. 3.3.2 Matrix-Product Representation (Zucchini's Notation)

Zucchini et al. express the likelihood using **matrix products**. Define

- A diagonal matrix of emission densities at time t:

$$\mathbf{Q}(y_t) = \text{diag}(f_1(y_t), \dots, f_K(y_t)).$$

Then one can show that

$$L(\theta; y_{1:T}) = \delta^\top \mathbf{Q}(y_1) \Gamma \mathbf{Q}(y_2) \cdots \Gamma \mathbf{Q}(y_T) \mathbf{1},$$

where $\mathbf{1}$ is the column vector of ones.

Derivation (sketch): each matrix multiplication corresponds to summing over an intermediate state index. The product $\mathbf{Q}(y_t)\Gamma\mathbf{Q}(y_{t+1})$ encodes the contribution of transitions from time t to $t+1$ and emissions at both times.

This matrix formulation is central in Zucchini et al. and will match the **forward variable recursion** in Section 4.

6.4. 3.4 Log-Likelihood and Its Geometry

The **log-likelihood** is

$$\ell(\theta; y_{1:T}) = \log L(\theta; y_{1:T}).$$

Properties:

- ℓ is typically **non-convex** in θ due to hidden states and combinatorial symmetries (label switching);
- It is, however, **smooth** in the interior of the parameter space (for regular emission families);
- Gradient and Hessian can be expressed in terms of **forward–backward quantities** and **conditional expectations**.

These observations motivate the **EM algorithm**: instead of maximizing ℓ directly, one maximizes a **lower bound** (Section 5.2), whose geometry is often easier.

6.5. 3.5 Parameter Space and Constraints

The parameter space naturally decomposes as

$$\Theta = \Delta^{K-1} \times \mathcal{G} \times \Phi,$$

where

- Δ^{K-1} is the simplex for the initial distribution δ;
- $\mathcal{G}$ is the set of $K \times K$ row-stochastic matrices Γ;
- Φ is the product of emission parameter spaces $\Phi_1 \times \cdots \times \Phi_K$.

Constraints:

- $\delta_i \geq 0, \sum_i \delta_i = 1$;
- $\gamma_{ij} \geq 0, \sum_j \gamma_{ij} = 1$ for each i;
- Emission parameters must keep f_i valid probability distributions.

Optimization (MLE, EM) must respect these constraints; many algorithms use **reparameterizations** (e.g. softmax/logistic transforms) to enforce them automatically.

6.6. 3.6 Summary

In this section we:

- Formally defined a finite-state HMM as a pair of processes (S_t, Y_t) with a Markov hidden chain and conditionally independent emissions;
- Derived the **joint** likelihood of states and observations;
- Obtained the **marginal** likelihood as a sum over K^T state sequences;
- Introduced the **matrix-product representation** of the likelihood used extensively by Zucchini et al.

This sets the stage for:

- **Section 4:** Efficient inference algorithms (forward–backward, Viterbi) that compute various conditional probabilities and the likelihood in $\mathcal{O}(K^2T)$;
- **Section 5:** Parameter estimation via maximum likelihood and EM/Baum–Welch.

For a detailed treatment closely aligned with this notation, see Zucchini et al., **Chapter 2 (The HMM)**.

7. Section 4 – Inference in Hidden Markov Models

This section develops the **core inference algorithms** for finite-state HMMs:

- **Filtering (forward algorithm)** – computing $\mathbb{P}(S_t = i \mid Y_{1:t})$;
- **Smoothing (forward–backward)** – computing $\mathbb{P}(S_t = i \mid Y_{1:T})$;
- **Decoding (Viterbi)** – computing the most probable state sequence $\arg\max_{s_{1:T}} \mathbb{P}(S_{1:T} = s_{1:T} \mid Y_{1:T})$.

We emphasize **recursive structure**, **dynamic programming**, **proofs of correctness**, and **numerical stability**.

The treatment aligns with **Zucchini et al.**, Chapters 2–3, and **Rabiner (1989)**, but is more explicit about the probabilistic underpinnings.

Throughout, $\theta = (\delta, \Gamma, f_1, \dots, f_K)$ denotes the HMM parameters, and we condition implicitly on θ when unambiguous.

7.1. 4.1 Filtering – The Forward Algorithm

7.1.1. 4.1.1 Filtering and Predictive Distributions

Given observations $Y_{1:t} = y_{1:t}$, define

- The **filtering distribution** (posterior over states):

$$\alpha_t(i) := \mathbb{P}(S_t = i \mid Y_{1:t} = y_{1:t}), \quad i = 1, \dots, K.$$

- The **one-step predictive distribution**:

$$\mathbb{P}(Y_{t+1} \in A \mid Y_{1:t} = y_{1:t}) = \sum_{i=1}^{K} \mathbb{P}(S_t = i \mid Y_{1:t}) \sum_{j=1}^{K} \gamma_{ij} F_j(A).$$

The forward algorithm computes all α_t **recursively in** t, in $\mathcal{O}(K^2 T)$ time.

7.1.2. 4.1.2 Unnormalized Forward Variables

Define the **unnormalized forward variables**

$$\tilde{\alpha}_t(i) := \mathbb{P}(S_t = i, Y_{1:t} = y_{1:t}).$$

Then

$$\alpha_t(i) = \frac{\tilde{\alpha}_t(i)}{\sum_{j=1}^{K} \tilde{\alpha}_t(j)}.$$

The forward recursion is most naturally stated for $\tilde{\alpha}_t(i)$.

7.1.3. 4.1.3 Derivation of the Recursion

Initialization (t = 1).

$$\tilde{\alpha}_1(i) = \mathbb{P}(S_1 = i, Y_1 = y_1) = \mathbb{P}(S_1 = i)\,\mathbb{P}(Y_1 = y_1 \mid S_1 = i) = \delta_i f_i(y_1).$$

Induction step. For $t \geq 1$,

$$\begin{aligned}
\tilde{\alpha}_{t+1}(j) &= \mathbb{P}(S_{t+1} = j, Y_{1:t+1} = y_{1:t+1}) \\
&= \sum_{i=1}^{K} \mathbb{P}(S_t = i, S_{t+1} = j, Y_{1:t+1} = y_{1:t+1}) \\
&= \sum_{i=1}^{K} \mathbb{P}(S_t = i, Y_{1:t} = y_{1:t}) \\
&\quad \mathbb{P}(S_{t+1} = j \mid S_t = i) \\
&\quad \mathbb{P}(Y_{t+1} = y_{t+1} \mid S_{t+1} = j) \\
&= \sum_{i=1}^{K} \tilde{\alpha}_t(i)\,\gamma_{ij}\,f_j(y_{t+1}).
\end{aligned}$$

The key step uses:

- The **Markov property** for S_t;
- Conditional independence of Y_{t+1} from the past given S_{t+1}.

Thus the recursion is

$$\boxed{\tilde{\alpha}_{t+1}(j) = f_j(y_{t+1}) \sum_{i=1}^{K} \tilde{\alpha}_t(i)\,\gamma_{ij}.}$$

7.1.4. 4.1.4 Matrix Formulation (Zucchini's Notation)

Let

- $\tilde{\alpha}_t$ be the row vector with entries $\tilde{\alpha}_t(i)$;
- $\mathbf{Q}(y_t) = \text{diag}(f_1(y_t), \dots, f_K(y_t))$ as before.

Then

$$\tilde{\alpha}_1 = \delta^\top \mathbf{Q}(y_1),$$
$$\tilde{\alpha}_{t+1} = \tilde{\alpha}_t \, \Gamma \, \mathbf{Q}(y_{t+1}).$$

This matches precisely the likelihood expression in Section 3.3: the marginal likelihood is

$$L(\theta; y_{1:T}) = \sum_{i=1}^{K} \tilde{\alpha}_T(i) = \tilde{\alpha}_T \mathbf{1}.$$

Zucchini et al. use this matrix-product viewpoint extensively; the forward algorithm is exactly this recursion plus normalization at each step.

7.1.5. 4.1.5 Proof of Correctness by Induction

We show that the recursion indeed computes $\tilde{\alpha}_t(i) = \mathbb{P}(S_t = i, Y_{1:t} = y_{1:t})$ for all t.

- **Base case:** Already verified for $t = 1$.
- **Induction step:** Assume formula holds for t. Then using only the model assumptions (Markov property and conditional independence), we derived the recursion, which equals by definition
$$\mathbb{P}(S_{t+1} = j, Y_{1:t+1} = y_{1:t+1}).$$

Hence, by induction, the recursion is correct for all t. This is the standard argument also given in Zucchini et al. (with lighter measure-theoretic detail).

7.1.6. 4.1.6 Numerical Stability: Scaling and Log-Domain

Direct computation of $\tilde{\alpha}_t(i)$ leads to **underflow**, since they involve products of T probabilities. Two standard cures:

1. **Scaling:** At each step define a scaling constant
$$c_t = \sum_{i=1}^{K} \tilde{\alpha}_t(i), \quad \hat{\alpha}_t(i) = \frac{\tilde{\alpha}_t(i)}{c_t}.$$
Then $\hat{\alpha}_t$ is the normalized filtering distribution, and
$$L(\theta; y_{1:T}) = \prod_{t=1}^{T} c_t, \quad \ell(\theta; y_{1:T}) = \sum_{t=1}^{T} \log c_t.$$
This is exactly the implementation recommended in Zucchini et al.

2. **Log-domain forward algorithm:** Work with

$$a_t(i) = \log \tilde{\alpha}_t(i),$$

and use the **log-sum-exp** trick for the recursion:

$$a_{t+1}(j) = \log f_j(y_{t+1}) + \log\Big(\sum_{i=1}^{K} e^{a_t(i)+\log \gamma_{ij}}\Big).$$

Numerically, compute

$$\log \sum_i e^{z_i} = m + \log \sum_i e^{z_i - m}, \quad m = \max_i z_i,$$

to avoid overflow and underflow.

7.2. 4.2 Smoothing – Forward–Backward Algorithm

Filtering uses observations up to time t. For many tasks (e.g. EM, state decoding), we need **smoothing distributions** that use the **entire sequence** $Y_{1:T}$.

7.2.1. 4.2.1 Smoothing Distributions and Backward Variables

Define the **smoothing distribution** at time t:

$$\gamma_t(i) := \mathbb{P}(S_t = i \mid Y_{1:T} = y_{1:T}).$$

Introduce **backward variables**

$$\beta_t(i) := \mathbb{P}(Y_{t+1:T} = y_{t+1:T} \mid S_t = i).$$

Intuitively, $\beta_t(i)$ is the probability of observing the future $y_{t+1:T}$ if we know the current state is i.

7.2.2. 4.2.2 Backward Recursion

Initialization: At time T, there are no future observations, so by convention

$$\beta_T(i) = 1, \quad i = 1, \dots, K.$$

Induction step: For $t = T-1, \dots, 1$,

$$\begin{aligned}
\beta_t(i) &= \mathbb{P}(Y_{t+1:T} = y_{t+1:T} \mid S_t = i) \\
&= \sum_{j=1}^{K} \mathbb{P}(S_{t+1} = j, Y_{t+1:T} = y_{t+1:T} \mid S_t = i) \\
&= \sum_{j=1}^{K} \gamma_{ij} f_j(y_{t+1}) \beta_{t+1}(j).
\end{aligned}$$

Hence the **backward recursion** is

$$\boxed{\beta_t(i) = \sum_{j=1}^{K} \gamma_{ij} f_j(y_{t+1}) \beta_{t+1}(j).}$$

7.2.3. 4.2.3 Two-Filter Formula: Combining Forward and Backward

We have

$$\begin{aligned}\mathbb{P}(S_t = i, Y_{1:T} = y_{1:T}) &= \mathbb{P}(S_t = i, Y_{1:t} = y_{1:t}) \\ &\quad \mathbb{P}(Y_{t+1:T} = y_{t+1:T} \mid S_t = i, Y_{1:t} = y_{1:t}) \\ &= \tilde{\alpha}_t(i)\, \beta_t(i),\end{aligned}$$

since **future observations are conditionally independent of the past given** S_t.

Thus the smoothing distribution is

$$\gamma_t(i) = \mathbb{P}(S_t = i \mid Y_{1:T} = y_{1:T}) = \frac{\tilde{\alpha}_t(i)\, \beta_t(i)}{L(\theta; y_{1:T})}.$$

In scaled form, using $\hat{\alpha}_t(i)$ and scaled $\hat{\beta}_t(i)$, the denominator cancels nicely (see Zucchini et al. for implementation details):

$$\gamma_t(i) \propto \hat{\alpha}_t(i)\, \hat{\beta}_t(i),$$

with proportionality factors determined by normalization.

7.2.4. 4.2.4 Pairwise Smoothing Probabilities

For EM/Baum–Welch, we also need

$$\xi_t(i,j) := \mathbb{P}(S_t = i, S_{t+1} = j \mid Y_{1:T} = y_{1:T}).$$

Using similar reasoning,

$$\xi_t(i,j) = \frac{\tilde{\alpha}_t(i)\, \gamma_{ij} f_j(y_{t+1}) \beta_{t+1}(j)}{L(\theta; y_{1:T})}.$$

The arrays $\gamma_t(i)$ and $\xi_t(i,j)$ are exactly what EM uses as **expected sufficient statistics** for state occupancies and transitions.

7.3. 4.3 Decoding – The Viterbi Algorithm

Filtering and smoothing give **marginal posterior distributions** over states at each time. In many applications, one wants a **single state sequence estimate** $\hat{s}_{1:T}$.

The most common choice is the **maximum a posteriori (MAP) path**:

$$\hat{s}_{1:T}^{\mathrm{MAP}} \in \arg\max_{s_{1:T}} \mathbb{P}(S_{1:T} = s_{1:T} \mid Y_{1:T} = y_{1:T}).$$

Equivalently,

$$\hat{s}_{1:T}^{\mathrm{MAP}} \in \arg\max_{s_{1:T}} \mathbb{P}(S_{1:T} = s_{1:T}, Y_{1:T} = y_{1:T}),$$

since the denominator $\mathbb{P}(Y_{1:T} = y_{1:T})$ does not depend on $s_{1:T}$.

7.3.1. 4.3.1 Dynamic Programming Formulation

Define

$$\delta_t(j) := \max_{s_{1:t-1}} \mathbb{P}(S_t = j, S_{1:t-1} = s_{1:t-1}, Y_{1:t} = y_{1:t}),$$

and the **backpointer**

$$\psi_t(j) \in \arg\max_i \delta_{t-1}(i)\gamma_{ij}.$$

Then the Viterbi recursion is:

- **Initialization:**
$$\delta_1(j) = \delta_j f_j(y_1), \quad \psi_1(j) \text{ arbitrary.}$$
- **Recursion:** for $t = 2, \dots, T$,
$$\delta_t(j) = f_j(y_t) \max_i \delta_{t-1}(i)\gamma_{ij},$$
$$\psi_t(j) \in \arg\max_i \delta_{t-1}(i)\gamma_{ij}.$$
- **Termination:**
$$\hat{s}_T \in \arg\max_j \delta_T(j).$$
- **Backtracking:** For $t = T-1, \dots, 1$,
$$\hat{s}_t = \psi_{t+1}(\hat{s}_{t+1}).$$

7.3.2. 4.3.2 Proof of Correctness

The Viterbi algorithm is an instance of **dynamic programming** over a chain:

- For each t, j, $\delta_t(j)$ is the **maximum joint probability** over all paths ending in state j at time t;
- The optimal path to j at time t must pass through some i at time $t-1$, and that prefix must be optimal for reaching i at time $t-1$.

Formally, one proves by induction:

1. **Optimal substructure:** if $s^*_{1:T}$ maximizes $\mathbb{P}(S_{1:T}, Y_{1:T})$, then for each t, the prefix $s^*_{1:t}$ must maximize $\mathbb{P}(S_{1:t}, Y_{1:t})$ among all paths ending in s^*_t;
2. The recursion above computes exactly these maxima.

See Zucchini et al., Chapter 3, and Rabiner (1989) for standard textbook proofs.

7.3.3. 4.3.3 Max-Product Semiring Perspective

The Viterbi algorithm can be seen as a **max-product message passing** on the chain factor graph:

- Replace summation (as in forward algorithm) by maximization;
- Replace probabilities by their **logarithms**, turning products into sums:

$$v_t(j) = \log \delta_t(j) = \log f_j(y_t) + \max_i \{v_{t-1}(i) + \log \gamma_{ij}\} + \log \delta_j \mathbf{1}_{t=1}.$$

This semiring viewpoint is useful when generalizing to other objectives (e.g. **min-sum** for costs).

7.3.4. 4.3.4 Complexity and Path Properties

- Time complexity is $\mathcal{O}(K^2 T)$, same order as forward–backward;
- Memory complexity is $\mathcal{O}(KT)$ if all $\psi_t(j)$ are stored; can be reduced with more complex techniques.

Importantly, the **Viterbi path is not obtained by taking the most likely state at each time** (that would use $\gamma_t(i)$), because the most likely joint path is not obtained by locally maximizing each marginal.

7.4. 4.4 Other Inference Quantities

From filtering and smoothing, one can derive many other useful quantities:

- **Predictive distribution:**

$$\mathbb{P}(Y_{t+1} \in A \mid Y_{1:t}) = \sum_{i,j} \alpha_t(i)\gamma_{ij}F_j(A).$$

- **State occupancy expectations:** $\mathbb{E}[\mathbf{1}_{\{S_t=i\}} \mid Y_{1:T}] = \gamma_t(i)$.
- **Expected transition counts:** $\mathbb{E}[\mathbf{1}_{\{S_t=i,S_{t+1}=j\}} \mid Y_{1:T}] = \xi_t(i,j)$.

These are central to **parameter estimation** (Section 5) and to interpreting HMMs in applications (Section 10).

7.5. 4.5 Summary and References

We have developed:

- The **forward algorithm** for filtering, with rigorous derivation and scaling for numerical stability;
- The **backward recursion** and the forward–backward method for **smoothing** and pairwise probabilities;
- The **Viterbi algorithm** for MAP path decoding, with a dynamic programming interpretation and proof sketch.

These algorithms are the computational workhorses of HMM inference. Zucchini et al., **Chapters 2–3**, provide code-oriented explanations (often in R), while the more formal treatment here is aligned with **Cappé, Moulines, Rydén (2005)** and **Rabiner (1989)**.

8. Section 5 – Parameter Estimation in Hidden Markov Models

This section studies **parameter estimation** for finite-state HMMs, focusing on:

- **Maximum likelihood estimation (MLE)** and its properties;
- The **EM / Baum–Welch algorithm**, including derivation and monotonicity;
- **Identifiability theory** and label switching.

We follow the structure of **Zucchini et al.**, Chapters 3–4, and the rigorous development in **Cappé, Moulines, Rydén (2005)**.

Let $\theta = (\delta, \Gamma, \phi_1, \dots, \phi_K)$ collect all parameters (initial distribution, transition matrix, emission parameters). Given data $y_{1:T}$, we aim to estimate θ.

8.1. 5.1 Maximum Likelihood Estimation

8.1.1. 5.1.1 Definition

Given observed data $y_{1:T}$, the **likelihood function** is

$$L_T(\theta) := L(\theta; y_{1:T}) = \mathbb{P}_\theta(Y_{1:T} = y_{1:T}),$$

with log-likelihood

$$\ell_T(\theta) = \log L_T(\theta).$$

A **maximum likelihood estimator** (MLE) $\hat{\theta}_T$ is any point in

$$\hat{\theta}_T \in \arg\max_{\theta \in \Theta} \ell_T(\theta).$$

Because Θ is constrained (simplices, stochastic matrices), many implementations reparameterize (e.g. via logits) to perform unconstrained optimization.

8.1.2. 5.1.2 Non-Convexity and Local Maxima

The log-likelihood $\ell_T(\theta)$ for HMMs is typically **non-convex**:

- Hidden states introduce **latent-variable structure**;
- Symmetries (permutations of states) yield **multiple equivalent maxima**;
- There may be **spurious local maxima** unrelated to the true parameter.

Consequences:

- Gradient-based methods can get trapped in local optima;
- EM (below) converges to a **local stationary point**, not necessarily a global maximum;
- Good **initialization** (e.g. k-means clustering on observations, or simpler models) is critical in practice (as emphasized by Zucchini et al.).

8.1.3. 5.1.3 Label Switching and Equivalence Classes

For any permutation σ of $\{1, \dots, K\}$, define a **permuted parameter** θ^σ by

- $\delta_i^\sigma = \delta_{\sigma^{-1}(i)}$;
- $\gamma_{ij}^\sigma = \gamma_{\sigma^{-1}(i),\sigma^{-1}(j)}$;
- Emission parameters re-labeled: $\phi_i^\sigma = \phi_{\sigma^{-1}(i)}$.

Then

$$L_T(\theta^\sigma) = L_T(\theta)$$

for all T and all data sequences. Thus, parameters are at best **identifiable up to permutation** of hidden states.

This **label switching** means:

- The MLE is only unique up to permutation;
- Post-processing (e.g. ordering states by mean of emissions) is often used to select a canonical labeling (as in Zucchini et al.).

8.2. 5.2 EM / Baum–Welch Algorithm

8.2.1. 5.2.1 General EM Framework

Suppose Y is observed data and S is latent/hidden data. The EM algorithm iteratively maximizes the log-likelihood $\ell(\theta) = \log p_\theta(Y)$ via:

1. **E-step:** Compute
$$Q(\theta \mid \theta^{(k)}) = \mathbb{E}_{\theta^{(k)}}[\log p_\theta(Y, S) \mid Y].$$
2. **M-step:** Set
$$\theta^{(k+1)} \in \arg\max_\theta Q(\theta \mid \theta^{(k)}).$$

EM guarantees **non-decreasing likelihood**: $\ell(\theta^{(k+1)}) \geq \ell(\theta^{(k)})$.

In HMMs, $S = S_{1:T}$ (hidden states) and $Y = Y_{1:T}$ (observations).

8.2.2. 5.2.2 Complete-Data Log-Likelihood for HMMs

Recall (Section 3.2) that the **complete-data log-likelihood** is

$$\log p_\theta(S_{1:T}, Y_{1:T}) = \log \delta_{S_1} + \sum_{t=2}^{T} \log \gamma_{S_{t-1}, S_t} + \sum_{t=1}^{T} \log f_{S_t}(y_t; \phi_{S_t}).$$

Thus,

$$\begin{aligned} Q(\theta \mid \theta^{(k)}) &= \mathbb{E}_{\theta^{(k)}}[\log p_\theta(S_{1:T}, Y_{1:T}) \mid Y_{1:T} = y_{1:T}] \\ &= \sum_i \mathbb{E}[\mathbf{1}_{\{S_1 = i\}} \mid Y] \log \delta_i \\ &\quad + \sum_{t=2}^{T} \sum_{i,j} \mathbb{E}[\mathbf{1}_{\{S_{t-1}=i, S_t=j\}} \mid Y] \log \gamma_{ij} \\ &\quad + \sum_{t=1}^{T} \sum_i \mathbb{E}[\mathbf{1}_{\{S_t = i\}} \mid Y] \log f_i(y_t; \phi_i). \end{aligned}$$

Define the **expected sufficient statistics** under $\theta^{(k)}$:

$$\gamma_t^{(k)}(i) = \mathbb{P}_{\theta^{(k)}}(S_t = i \mid Y_{1:T}),$$

$$\xi_t^{(k)}(i,j) = \mathbb{P}_{\theta^{(k)}}(S_{t-1} = i, S_t = j \mid Y_{1:T}).$$

These are computed using the **forward–backward algorithm** (Section 4.2).

Then

$$\begin{aligned} Q(\theta \mid \theta^{(k)}) &= \sum_i \gamma_1^{(k)}(i) \log \delta_i \\ &\quad + \sum_{t=2}^{T} \sum_{i,j} \xi_t^{(k)}(i,j) \log \gamma_{ij} \\ &\quad + \sum_{t=1}^{T} \sum_i \gamma_t^{(k)}(i) \log f_i(y_t; \phi_i). \end{aligned}$$

8.2.3. 5.2.3 M-Step Updates

Maximizing Q over θ subject to the usual constraints yields closed-form updates for δ and Γ, and often for ϕ_i (for exponential-family emissions).

- **Initial distribution:**

$$\delta_i^{(k+1)} = \gamma_1^{(k)}(i).$$

- **Transition probabilities:** for each i,

$$\gamma_{ij}^{(k+1)} = \frac{\sum_{t=2}^{T} \xi_t^{(k)}(i,j)}{\sum_{t=2}^{T} \sum_{j'} \xi_t^{(k)}(i,j')}.$$

For **emission parameters** (e.g. Gaussian), the M-step corresponds to a **weighted maximum likelihood** with weights $\gamma_t^{(k)}(i)$. For instance, if f_i is normal $\mathcal{N}(\mu_i, \sigma_i^2)$,

$$\mu_i^{(k+1)} = \frac{\sum_{t=1}^{T} \gamma_t^{(k)}(i) y_t}{\sum_{t=1}^{T} \gamma_t^{(k)}(i)},$$

$$(\sigma_i^2)^{(k+1)} = \frac{\sum_{t=1}^{T} \gamma_t^{(k)}(i)(y_t - \mu_i^{(k+1)})^2}{\sum_{t=1}^{T} \gamma_t^{(k)}(i)}.$$

Zucchini et al. work out these updates for many common emission families (Poisson, normal, etc.).

8.2.4. 5.2.4 EM as Coordinate Ascent on an Evidence Lower Bound

Define a distribution $q(S_{1:T})$ over state sequences. Then

$$\log p_\theta(Y) = \mathcal{F}(q, \theta) + \mathrm{KL}(q(S_{1:T}) \| p_\theta(S_{1:T} \mid Y)),$$

where the **variational free energy** (or ELBO) is

$$\mathcal{F}(q, \theta) = \mathbb{E}_q[\log p_\theta(S_{1:T}, Y)] + H(q),$$

with entropy $H(q) = -\mathbb{E}_q[\log q(S_{1:T})]$.

Since KL is non-negative,

$$\mathcal{F}(q, \theta) \le \log p_\theta(Y),$$

with equality iff $q = p_\theta(S_{1:T} \mid Y)$.

EM alternates:

- **E-step:** Set $q^{(k)} = p_{\theta^{(k)}}(S_{1:T} \mid Y)$, which maximizes $\mathcal{F}(q, \theta^{(k)})$ over q;
- **M-step:** Maximize $\mathcal{F}(q^{(k)}, \theta)$ over θ, which is equivalent to maximizing $Q(\theta \mid \theta^{(k)})$.

Thus EM is **coordinate ascent** on $\mathcal{F}$, and therefore

$$\ell(\theta^{(k+1)}) \ge \ell(\theta^{(k)}).$$

8.2.5. 5.2.5 Convergence Properties

Under mild conditions (continuity of ℓ, compactness of parameter space or coercivity), the EM sequence $\{\theta^{(k)}\}$:

- Has **non-decreasing likelihood**;
- Every **limit point** is a **stationary point** of the likelihood (satisfies first-order conditions);
- Global convergence to the **global maximum** is not guaranteed.

Cappé, Moulines, Rydén (2005) provide detailed convergence results for HMM-EM; Zucchini et al. emphasize practical convergence diagnostics.

8.3. 5.3 Identifiability Theory

8.3.1. 5.3.1 Definition of Identifiability

Let $\mathcal{P}_\theta$ be the joint distribution of $Y_{1:\infty}$ under parameter θ. The HMM is **(strictly) identifiable** if

$$\mathcal{P}_\theta = \mathcal{P}_{\theta'} \implies \theta' \in \mathcal{E}(\theta),$$

where $\mathcal{E}(\theta)$ is the **equivalence class** of θ under state permutations (label switching).

Intuitively, **up to permutation of states**, the parameter is uniquely determined by the distribution of the observed process.

8.3.2. 5.3.2 Simple Non-Identifiability Examples

- If two states have identical rows in Γ and identical emission parameters, merging them yields another parameter with the same observed distribution.
- If emission distributions are **linearly dependent** in certain ways (e.g. deterministic relationships), different combinations of transition probabilities and emissions can produce the same marginal process.

These examples show that identifiability requires **structural conditions**.

8.3.3. 5.3.3 Sufficient Conditions for Finite-State HMMs (High-Level)

A line of work (e.g. Allman, Matias, Rhodes; Hsu, Kakade, Zhang; and results cited in Cappé et al.) gives sufficient conditions for identifiability of finite-state HMMs, typically requiring:

- The transition matrix Γ to be of **full rank** and ergodic;
- Emission distributions f_i to be **distinct** and to span a sufficiently rich function space (e.g. a linearly independent set in L^2);
- Enough lags of the observed process to be considered.

Under such conditions, the joint distribution of (Y_t, Y_{t+1}, Y_{t+2}) (or higher blocks) contains enough information to recover θ up to permutation.

8.3.4. 5.3.4 Practical Implications (Zucchini et al.)

In practice, Zucchini et al. stress that:

- One should avoid models where two states are effectively **indistinguishable** (same emissions, similar rows in Γ);
- **Overly complex models** (too many states) can lead to weak identifiability and unstable estimates;
- State labels are arbitrary; interpretability often requires **post hoc ordering** or constraints.

8.4. 5.4 Summary

In this section we:

- Defined MLE for HMMs and highlighted non-convexity and label switching;
- Derived the **Baum–Welch (EM) algorithm** from the complete-data likelihood, including explicit update formulas;
- Interpreted EM as **coordinate ascent** on an evidence lower bound, giving monotonicity and convergence to stationary points;
- Discussed **identifiability** and practical issues with overlapping states.

These results, together with the **asymptotic theory** in Section 6, provide a rigorous foundation for statistical inference in HMMs.

Part IV.

Theory & Advanced Models

9. Section 6 – Asymptotics and Statistical Theory for HMMs

This section treats the **large-sample behavior** of estimators in Hidden Markov Models, focusing on:

- **Consistency** of the maximum likelihood estimator (MLE);
- **Asymptotic normality** and Fisher information;
- **Misspecification** and pseudo-true parameters.

The development is inspired by **Cappé, Moulines, Rydén (2005)** and **Douc, Moulines, Stoffer (2014)**, who provide a rigorous ergodic-theoretic foundation. Zucchini et al. present the main ideas informally; here we state them more precisely.

We mainly consider **finite-state HMMs** with emission densities f_i that are smooth in parameters.

9.1. 6.1 Setup and Regularity Conditions

Let $\{(S_t, Y_t)\}_{t\geq 1}$ be an HMM with true parameter θ^*. Assume:

1. The hidden chain (S_t) is **irreducible and aperiodic**, with unique stationary distribution π^*;
2. Under θ^*, the joint process (S_t, Y_t) is **stationary and ergodic** (true if we start from stationarity or after a transient);
3. The parameter space Θ is compact or the log-likelihood is **coercive**;
4. The emission densities $f_i(y; \phi_i)$ and transition probabilities are **smooth** in θ;
5. The model is **identifiable up to permutation** (Section 5.3).

We observe $Y_{1:T}$ and compute the MLE $\widehat{\theta}_T$.

9.2. 6.2 Consistency of the MLE

9.2.1. 6.2.1 Log-Likelihood per Observation

Define the **average log-likelihood**

$$\bar{\ell}_T(\theta) = \frac{1}{T}\ell_T(\theta) = \frac{1}{T}\log p_\theta(Y_{1:T}).$$

A key result: for each fixed θ, the limit

$$\ell_\infty(\theta) = \lim_{T\to\infty} \bar{\ell}_T(\theta)$$

exists **almost surely** (and in L^1), and can be expressed as an expectation under the stationary distribution of the hidden chain and emissions.

This follows from subadditive ergodic theorems or from explicit Markov chain arguments (see Cappé et al., Chapter 9).

9.2.2. 6.2.2 Identification of the Limit

Under stationarity, one can show that

$$\ell_\infty(\theta) = \mathbb{E}_{\theta^*}\big[\log p_\theta(Y_0 \mid Y_{-\infty:-1})\big],$$

where $Y_{-\infty:0}$ denotes the infinite past.

Intuitively, $\ell_\infty(\theta)$ is the **expected log predictive likelihood** of Y_0 given the entire past, under the true parameter θ^*, but evaluated at a candidate parameter θ.

9.2.3. 6.2.3 Consistency under Correct Specification

If the model is correctly specified and identifiable (up to permutation), then

$$\ell_\infty(\theta) \le \ell_\infty(\theta^*)$$

with equality **only** if θ belongs to the permutation-equivalence class of θ^*.

Under mild regularity conditions, we can show that

$$\sup_{\theta\in\Theta} \bar{\ell}_T(\theta) \xrightarrow[T\to\infty]{\text{a.s.}} \sup_{\theta\in\Theta} \ell_\infty(\theta) = \ell_\infty(\theta^*).$$

If the argmax of ℓ_∞ is unique up to permutation, then **any sequence of MLEs** $\hat{\theta}_T$ converges almost surely to the equivalence class of θ^*. This is **strong consistency** (modulo label switching).

9.2.4. 6.2.4 Misspecification and Pseudo-True Parameters

If the true data-generating process is **not** in the model class, there is no θ^* such that $\mathcal{P}_\theta = \mathcal{P}_{\text{true}}$. Instead, we define a **pseudo-true parameter**:

$$\theta^\circ \in \arg\min_{\theta\in\Theta} \mathrm{KL}(\mathcal{P}_{\text{true}} \| \mathcal{P}_\theta),$$

where $\mathcal{P}_\theta$ is the distribution of $Y_{1:\infty}$ under θ.

Under general conditions, $\hat{\theta}_T$ converges almost surely to θ°. Thus, the MLE approximates the best-fitting model in the Kullback–Leibler sense.

9.3. 6.3 Asymptotic Normality and Fisher Information

9.3.1. 6.3.1 Score Function and Information

The **score function** is

$$U_T(\theta) = \nabla_\theta \ell_T(\theta).$$

The **Fisher information matrix** at θ is

$$I_T(\theta) = -\mathbb{E}_\theta[\nabla^2_\theta \ell_T(\theta)] = \mathbb{E}_\theta[U_T(\theta) U_T(\theta)^\top].$$

For large T, it is natural to study **per-observation** quantities:

$$\bar{U}_T(\theta) = \frac{1}{\sqrt{T}} U_T(\theta), \quad \bar{I}_T(\theta) = \frac{1}{T} I_T(\theta).$$

Under stationarity and ergodicity, one can show that

$$\bar{I}_T(\theta^*) \xrightarrow[T\to\infty]{} I(\theta^*),$$

where $I(\theta^*)$ is the **limiting Fisher information per time step**.

9.3.2. 6.3.2 Central Limit Theorem for the Score

Under appropriate **mixing conditions** (e.g. geometric β-mixing) for the observed process (Y_t), the normalized score satisfies a **central limit theorem**:

$$\bar{U}_T(\theta^*) = \frac{1}{\sqrt{T}} \nabla_\theta \ell_T(\theta^*) \xrightarrow{d} \mathcal{N}(0, I(\theta^*)).$$

The proof typically relies on:

- Writing $U_T(\theta^*)$ as a sum of a **stationary, martingale difference** sequence plus negligible terms;
- Applying a martingale CLT or a mixing CLT.

9.3.3. 6.3.3 Asymptotic Normality of the MLE

Assuming:

- $\hat{\theta}_T \to \theta^*$ almost surely (consistency);
- $I(\theta^*)$ is **non-singular**;
- Regularity conditions for Taylor expansions;

we expand the score around θ^*:

$$0 = U_T(\hat{\theta}_T) = U_T(\theta^*) + \nabla^2_\theta \ell_T(\tilde{\theta}_T)(\hat{\theta}_T - \theta^*),$$

for some $\tilde{\theta}_T$ between $\hat{\theta}_T$ and θ^*.

Divide by $\sqrt{T}$:

$$0 = \bar{U}_T(\theta^*) + \Big(\frac{1}{T}\nabla^2_\theta \ell_T(\tilde{\theta}_T)\Big)\sqrt{T}(\hat{\theta}_T - \theta^*).$$

As $T \to \infty$, the second factor converges to $-I(\theta^*)$, and $\bar{U}_T(\theta^*)$ converges in distribution to $\mathcal{N}(0, I(\theta^*))$. Hence

$$\sqrt{T}(\hat{\theta}_T - \theta^*) \xrightarrow{d} \mathcal{N}(0, I(\theta^*)^{-1}).$$

This is the **asymptotic normality** of the MLE.

9.3.4. 6.3.4 Computing the Information in HMMs

In HMMs, $I(\theta^*)$ can be computed using **forward–backward quantities** and expectations under the stationary distribution.

One approach:

- Express the score as
$$U_T(\theta) = \sum_{t=1}^{T} u_t(\theta),$$
where $u_t(\theta)$ depends on local conditional distributions (e.g. $p_\theta(S_t, S_{t+1} \mid Y_{1:T})$);
- Compute $\mathbb{E}_{\theta^*}[u_t(\theta^*)u_s(\theta^*)^\top]$ and sum over lags.

Douc, Moulines, Stoffer provide explicit formulas and practical approximations.

9.4. 6.4 Model Selection and Information Criteria

Given a family of HMMs with different numbers of states K, we may select K using **information criteria** such as AIC or BIC.

The **Bayesian Information Criterion (BIC)** is

$$\mathrm{BIC} = -2\ell_T(\hat{\theta}_T) + d \log T,$$

where d is the number of free parameters in θ.

Under regularity conditions, BIC is an approximation to -2 **times the log marginal likelihood** (integrated over a prior), and tends to favor the **true model order** when it is among the candidates.

In HMMs, some regularity assumptions may fail (e.g. at parameter boundaries), but BIC is widely used and discussed by Zucchini et al. as a practical guide.

9.5. 6.5 Summary

We have sketched the main elements of **asymptotic theory** for HMMs:

- Existence of a limiting average log-likelihood $\ell_\infty(\theta)$ under ergodicity;
- **Consistency** of MLEs under identifiability and regularity;
- **Asymptotic normality** with covariance given by the inverse **Fisher information**;
- Behavior under **misspecification**, leading to pseudo-true parameters.

These results justify the use of MLE and information criteria in large-sample regimes, and they underpin more advanced methods such as **online estimation** and **sequential Monte Carlo** for HMMs.

10. Section 7 – Non-Standard and Advanced Hidden Markov Models

This section surveys important **extensions and generalizations** of the basic finite-state HMM:

- **Continuous-state HMMs / state-space models** (Kalman filter as a linear-Gaussian HMM);
- **Nonparametric HMMs** with (theoretically) infinitely many states;
- **Switching state-space models** and regime-switching processes.

These models are beyond the core scope of Zucchini et al., but are natural continuations of the HMM framework.

10.1. 7.1 Continuous-State HMMs and State-Space Models

10.1.1. 7.1.1 General State-Space Models

A **state-space model** (SSM) generalizes finite-state HMMs by allowing the hidden state to live in a **continuous space**, typically $\mathbb{R}^d$:

- Hidden process (X_t) on $\mathbb{R}^d$ with transition density

$$p_\theta(x_{t+1} \mid x_t);$$

- Observation process (Y_t) with conditional density

$$g_\theta(y_t \mid x_t).$$

The Markov and conditional independence assumptions are analogous to HMMs:

- $X_{t+1} \perp\!\!\!\perp X_{1:t-1} \mid X_t$;
- $Y_t \perp\!\!\!\perp (X_{1:t-1}, X_{t+1:\infty}, Y_{1:t-1}, Y_{t+1:\infty}) \mid X_t$.

The joint density over $X_{1:T}, Y_{1:T}$ factorizes as

$$\mu(x_1)g(y_1 \mid x_1)\prod_{t=2}^{T} p(x_t \mid x_{t-1})g(y_t \mid x_t),$$

mirroring the finite-state HMM.

10.1.2. 7.1.2 Linear-Gaussian State-Space Models (Kalman Filter)

A particularly important class is the **linear-Gaussian state-space model**:

$$X_{t+1} = FX_t + W_t, \quad W_t \sim \mathcal{N}(0, Q),$$

$$Y_t = HX_t + V_t, \quad V_t \sim \mathcal{N}(0, R),$$

where F, H are matrices, and Q, R are covariance matrices.

Here, $X_t \in \mathbb{R}^d$ is a hidden **continuous state**, and $Y_t \in \mathbb{R}^m$ is observed. The model is Gaussian and Markov; the **Kalman filter** provides exact filtering distributions

$$\mathcal{L}(X_t \mid Y_{1:t}) = \mathcal{N}(m_t, P_t)$$

via recursive updates of the mean m_t and covariance P_t.

This is the continuous analog of the forward algorithm; see Douc, Moulines, Stoffer for a rigorous treatment.

10.1.3. 7.1.3 Relation to Finite-State HMMs

Both finite-state HMMs and linear-Gaussian SSMs share:

- Markovian hidden dynamics;
- Conditional independence structure for observations;
- Recursive inference via **filtering/smoothing algorithms**.

Finite-state HMMs can be seen as a **discrete-state** special case of SSMs, while linear-Gaussian SSMs can be thought of as having a **continuous hidden state** with Gaussian transitions and emissions.

10.2. 7.2 Nonparametric HMMs and Infinite-State Models

10.2.1. 7.2.1 Motivation

Standard HMMs assume a **fixed number of states** K. In some applications, choosing K is difficult or arbitrary. **Nonparametric HMMs** aim to allow a **potentially infinite** number of states, with the data effectively using only finitely many.

10.2.2. 7.2.2 Dirichlet Process HMMs (Informal)

A **Dirichlet process (DP)** is a distribution over probability measures. In an HMM context, one can place a DP prior on the **rows** of the transition matrix, yielding a **DP-HMM**:

- Each row $\Gamma_{i,\cdot}$ is drawn from a DP centered on a base distribution over states;
- Posterior inference encourages **sparse** transition structures and can infer an effective number of states from data.

More structured models such as the **Hierarchical Dirichlet Process HMM (HDP-HMM)** share transition distributions across states and time.

The resulting posterior is supported on **countably infinite state spaces**, but in any finite dataset only a finite number of states have significant posterior mass.

10.2.3. 7.2.3 Inference Challenges

Posterior inference in nonparametric HMMs typically requires:

- **Markov chain Monte Carlo (MCMC)** methods (Gibbs sampling, beam sampling);
- Or **variational inference** (truncating the infinite state space at a large $K_{\max}$).

While Zucchini et al. focus on finite-state models, the same **forward–backward structure** underlies these more complex Bayesian procedures.

10.3. 7.3 Switching State-Space Models and Regime-Switching

10.3.1. 7.3.1 Model Structure

A **switching state-space model** combines discrete regimes with continuous dynamics:

- Discrete hidden regime $S_t \in \{1, \dots, K\}$ evolving as a Markov chain with transition matrix Γ;
- Continuous hidden state $X_t \in \mathbb{R}^d$ with **regime-dependent dynamics**:

$$X_{t+1} = F_{S_t} X_t + W_t, \quad W_t \sim \mathcal{N}(0, Q_{S_t});$$

- Observations

$$Y_t = H_{S_t} X_t + V_t, \quad V_t \sim \mathcal{N}(0, R_{S_t}).$$

This yields a very flexible model where each regime has its own linear-Gaussian dynamics and observation structure.

10.3.2. 7.3.2 Inference

Exact inference is generally **intractable** due to the exponential number of possible regime sequences and continuous states. Approaches include:

- **Approximate dynamic programming** (e.g. Gaussian sum approximations);
- **Particle filters** and **Rao–Blackwellized particle filters** that sample regime sequences while integrating over continuous states using Kalman filters;
- **EM-like algorithms** using approximate E-steps.

10.3.3. 7.3.3 Applications

Switching and regime-switching models are common in:

- **Econometrics** (e.g. Markov-switching autoregressions for business cycles);
- **Signal processing** (systems with mode changes);
- **Engineering** (fault detection, hybrid systems).

They sit at the intersection of HMMs, state-space models, and control theory.

10.4. 7.4 Summary

This section sketched several important generalizations of HMMs:

- **Continuous-state models** (state-space models) with Kalman filtering as a canonical example;
- **Nonparametric HMMs** with an unbounded number of states via Dirichlet process priors;
- **Switching state-space models** blending discrete regimes with continuous dynamics.

While Zucchini et al. primarily focus on finite-state HMMs, many of the **conceptual tools** carry over: Markov structure, conditional independence, and recursive inference algorithms.

11. Section 8 – Computational and Numerical Issues in HMMs

The previous sections described the **theoretical** and **algorithmic** aspects of HMMs. This section focuses on

- **Numerical stability** (underflow, overflow, log-domain computations);
- **Time and memory complexity** of inference and learning;
- **Approximate inference** when exact methods are too expensive.

Zucchini et al. devote substantial attention to **implementation details** (especially in R code); here we formalize and extend those considerations.

11.1. 8.1 Numerical Stability

11.1.1. 8.1.1 Underflow in the Forward Algorithm

Recall the unnormalized forward variables

$$\tilde{\alpha}_t(i) = \mathbb{P}(S_t = i, Y_{1:t} = y_{1:t}).$$

For moderate T, these values can be extremely small:

- If typical emission probabilities are around 10^{-2}, then $\prod_{t=1}^{T} 10^{-2} = 10^{-2T}$ quickly underflows in double precision.

Therefore, naive implementations of the forward recursion lead to **numerical zeros**, even when the true probability is non-zero.

11.1.2. 8.1.2 Scaling Strategy

A standard solution (used systematically in Zucchini et al.) is to **renormalize** at each time step.

Define scaling constants

$$c_t = \sum_{i=1}^{K} \tilde{\alpha}_t(i),$$

and scaled forward variables

$$\hat{\alpha}_t(i) = \frac{\tilde{\alpha}_t(i)}{c_t}.$$

Then

$$\sum_i \hat{\alpha}_t(i) = 1, \quad \hat{\alpha}_t(i) = \mathbb{P}(S_t = i \mid Y_{1:t} = y_{1:t}).$$

Moreover,

$$L(\theta; y_{1:T}) = \prod_{t=1}^{T} c_t,$$

so the log-likelihood is

$$\ell(\theta; y_{1:T}) = \sum_{t=1}^{T} \log c_t.$$

This approach keeps all computations in a numerically safe range while preserving the **exact values** of probabilities (up to floating-point rounding).

11.1.3. 8.1.3 Log-Domain Computations

An alternative is to work entirely in the **log domain**. Let

$$a_t(i) = \log \tilde{\alpha}_t(i).$$

Then the recursion becomes

$$a_{t+1}(j) = \log f_j(y_{t+1}) + \log \sum_{i=1}^{K} e^{a_t(i) + \log \gamma_{ij}}.$$

To compute $\log \sum_i e^{z_i}$ stably, use the **log-sum-exp** identity:

$$\log \sum_i e^{z_i} = m + \log \sum_i e^{z_i - m}, \quad m = \max_i z_i.$$

This avoids overflow/underflow as long as z_i are in representable range. Similar tricks apply in backward, Viterbi, and EM computations.

11.1.4. 8.1.4 Backward and Viterbi Stability

- **Backward recursion**: Use either scaling synchronized with forward scaling or log-domain operations to avoid accumulation of tiny values.
- **Viterbi algorithm**: Since it already works with **max-products**, it is natural to convert to **max-sum** in log space, which improves stability and interpretability (additive costs).

11.2. 8.2 Computational Complexity

11.2.1. 8.2.1 Inference for a Single Sequence

Let K be the number of states and T the sequence length.

- **Forward algorithm:** For each t, computing $\tilde{\alpha}_{t+1}(j)$ requires a sum over $i = 1, \dots, K$, so the cost per time step is $\mathcal{O}(K^2)$. Total cost is $\mathcal{O}(K^2 T)$.
- **Backward algorithm:** Same complexity as forward.
- **Viterbi algorithm:** Also $\mathcal{O}(K^2 T)$ due to the max over i for each j, t.

Memory usage:

- Forward alone can be done with $\mathcal{O}(K)$ memory if only the likelihood is needed;
- Forward–backward typically stores $\mathcal{O}(KT)$ values (e.g. $\hat{\alpha}_t$, $\hat{\beta}_t$) unless one uses streaming or **checkpointing** strategies.

11.2.2. 8.2.2 EM / Baum–Welch Complexity

Each EM iteration involves:

- A full **forward–backward pass** per sequence: $\mathcal{O}(K^2 T)$;
- Simple **M-step updates** costing $\mathcal{O}(K^2 T)$ for transitions and $\mathcal{O}(KT)$ for emissions.

If there are N independent sequences of average length T, the per-iteration cost is $\mathcal{O}(NK^2 T)$.

Zucchini et al. highlight that, for moderate K (say $K \leq 10$) and reasonably long time series, EM is typically very fast on modern hardware.

11.2.3. 8.2.3 Scalability Considerations

For large-scale problems:

- Reducing **state space size** or enforcing **sparsity** in Γ (many zeros) can reduce the K^2 factor;
- Parallelization over sequences is straightforward;
- GPU implementations can exploit the regular structure of matrix–vector products.

11.3. 8.3 Approximate Inference Methods

When exact $\mathcal{O}(K^2 T)$ inference is too costly or when the model is more complex (e.g. continuous-state or nonparametric HMMs), **approximate methods** are used.

11.3.1. 8.3.1 Truncated and Beam Search for Viterbi

For very large K or long sequences, one can approximate Viterbi by:

- **Beam search:** At each time step, keep only the top B partial paths (states) according to their scores; complexity becomes $\mathcal{O}(BKT)$ with trade-off between accuracy and speed.

11.3.2. 8.3.2 Particle Filters (Sequential Monte Carlo)

For continuous-state models, **particle filters** approximate filtering distributions by a weighted set of particles $\{(X_t^{(n)}, w_t^{(n)})\}$. For finite-state HMMs, particle filters are not usually necessary, but similar ideas can be applied to **very large or structured state spaces**.

11.3.3. 8.3.3 Variational Inference

In complex HMM variants (e.g. nonparametric HMMs, switching SSMs), one often uses **variational approximations**:

- Posit a factorized form for the posterior over states (e.g. mean-field or structured);
- Optimize an **ELBO**, similar in spirit to EM but with additional approximations;
- Retain forward–backward-like updates, but in an approximate model.

11.3.4. 8.3.4 Online and Streaming Algorithms

For streaming data, one can use:

- **Online EM**: update parameter estimates incrementally using stochastic approximation to the E-step statistics;
- **Recursive maximum likelihood** methods (e.g. gradient ascent with step sizes η_t).

These algorithms rely heavily on **ergodic and mixing properties** discussed in Section 6.

11.4. 8.4 Implementation Notes (Zucchini et al.)

Zucchini et al. provide practical guidance on implementing HMMs, including:

- Careful use of **scaling** in forward–backward algorithms;
- Vectorized operations (e.g. in R or MATLAB) to exploit matrix structures;
- Diagnostics for **convergence** and **numerical issues** (e.g. checking that filtering probabilities remain normalized).

These considerations are essential for turning theoretical algorithms into **robust software**.

11.5. 8.5 Summary

This section covered the **algorithmic engineering** side of HMMs:

- Handling **underflow and overflow** via scaling and log-domain computations;
- Understanding the **time and space complexity** of inference and EM;
- Employing **approximate methods** when exact inference is infeasible.

These issues are critical in real-world applications, even though the mathematical structure of HMMs remains the same.

12. Section 9 – Alternative Foundations for HMMs

This section explores **non-standard perspectives** on HMMs that go beyond classical likelihood-based estimation:

- **Online and distribution-free viewpoints**, including prediction with expert advice and regret bounds;
- **Decision-theoretic framing** of HMMs as partially observable control problems (POMDPs).

These perspectives are not central in Zucchini et al., but are powerful for understanding HMMs in **sequential decision-making** and **adversarial or non-stationary environments**.

12.1. 9.1 Online Prediction and Regret

12.1.1. 9.1.1 Prediction Problem Setup

Consider a sequence of observations $Y_1, Y_2, \ldots$ taking values in a measurable space $\mathcal{Y}$. At each time t:

1. The forecaster outputs a predictive distribution q_t over Y_t based on $Y_{1:t-1}$;
2. The true outcome Y_t is revealed;
3. The forecaster incurs a loss $\ell(q_t, Y_t)$, often **log-loss**:

$$\ell(q_t, Y_t) = -\log q_t(Y_t).$$

An HMM with parameter θ induces a natural predictive distribution

$$q_t^\theta(\cdot) = p_\theta(\cdot \mid Y_{1:t-1}).$$

The question: how do such predictors perform in an **online** or **adversarial** setting?

12.1.2. 9.1.2 Regret Against a Class of HMMs

Fix a class of HMMs $\{p_\theta : \theta \in \Theta\}$. The **cumulative log-loss** of predictor q up to time T is

$$L_T(q) = \sum_{t=1}^{T} -\log q_t(Y_t).$$

The **regret** against the best HMM in hindsight is

$$R_T(q) = L_T(q) - \inf_{\theta \in \Theta} L_T(q^\theta).$$

One can design online algorithms (e.g. mixture-based or Bayesian) whose regret grows **sublinearly** in T, ensuring that the average additional loss **vanishes** asymptotically.

This connects to the **universal prediction** literature, where HMMs serve as a rich, structured class of experts.

12.1.3. 9.1.3 Bayesian Mixture over HMMs

Consider a prior Π over Θ, and define the **Bayesian mixture predictor**

$$q_t^{\text{mix}}(\cdot) = \int p_\theta(\cdot \mid Y_{1:t-1})\, \Pi(d\theta \mid Y_{1:t-1}),$$

where $\Pi(\cdot \mid Y_{1:t-1})$ is the posterior over θ.

Under log-loss, such mixture predictors achieve near-optimal regret bounds against the best θ in Θ. This is an example of **distribution-free performance guarantees** — no assumptions are made on how Y_t are generated.

12.2. 9.2 Decision-Theoretic Framing and POMDPs

12.2.1. 9.2.1 HMMs as Partially Observable Markov Decision Processes

A **Partially Observable Markov Decision Process (POMDP)** consists of:

- Hidden states S_t in a set E;
- Actions A_t in an action set $\mathcal{A}$;
- Observations Y_t in $\mathcal{Y}$;
- Transition probabilities $p(s_{t+1} \mid s_t, a_t)$;
- Observation probabilities $p(y_t \mid s_t)$;
- Reward (or cost) function $r(s_t, a_t)$.

An HMM is a **degenerate POMDP** with **no actions** (or a single trivial action) and no explicit rewards. Nevertheless, framing HMMs as POMDPs is useful:

- The **belief state** $b_t(i) = \mathbb{P}(S_t = i \mid Y_{1:t})$ is a sufficient statistic for the history;
- Filtering (forward algorithm) is exactly the **belief update** in a POMDP.

12.2.2. 9.2.2 Control and Decision Problems with HMMs

In many applications, we do not only wish to **infer** the hidden states but also to perform **actions** based on our beliefs:

- **Maintenance / reliability:** hidden state models system health; actions trigger inspections/repairs;
- **Finance:** hidden regimes guide trading decisions;
- **Medicine:** hidden disease states guide treatment decisions.

Formally, we want to choose policies π mapping belief states (or observation histories) to actions, to maximize expected cumulative reward:

$$\max_{\pi} \mathbb{E}\left[\sum_{t=1}^{T} r(S_t, A_t)\right].$$

12.2.3. 9.2.3 Dynamic Programming in Belief Space

In a POMDP, the optimal policy can be obtained by dynamic programming on the space of **beliefs** (probability distributions over states). For finite-state HMMs, the belief space is the simplex Δ^{K-1}.

The value function $V_t(b)$ satisfies a **Bellman equation** of the form

$$V_t(b) = \max_{a \in \mathcal{A}} \Big\{ r(b,a) + \mathbb{E}[V_{t+1}(b') \mid b, a] \Big\},$$

where b' is the updated belief after taking action a and receiving observation Y_{t+1}.

The belief update is exactly the **Bayesian filtering step**, which for HMM-like POMDPs is a linear-fractional map on Δ^{K-1}, followed by normalization.

12.2.4. 9.2.4 Risk-Sensitive and Robust Objectives

Beyond expected reward, one can study **risk-sensitive** or **robust** criteria:

- **Exponential utility:** maximize $-\frac{1}{\lambda}\log \mathbb{E}[e^{-\lambda \sum r_t}]$, linking to KL-regularized control;
- **Minimax regret:** choose policies that minimize the worst-case regret relative to a class of models.

These formulations often involve **entropy** and **KL divergence**, connecting back to Section 0.3 and EM-style variational principles.

12.3. 9.3 Summary

This section reframed HMMs in two broader contexts:

- As **online predictors** within a regret-minimization framework, where their performance can be compared against the best model in hindsight without assuming a true generative distribution;
- As special cases of **POMDPs**, where belief updates (filtering) are combined with **decision-making** and **control**.

These perspectives link the probabilistic foundations of HMMs (as in Zucchini et al.) with modern work in **online learning**, **reinforcement learning**, and **robust control**.

Part V.

Applications & Problem Sets

13. Section 10 – Applications of Hidden Markov Models

This section sketches **major application domains** of HMMs, emphasizing **precise mathematical formulations** rather than informal stories. For each domain we describe:

- The **state space** and its interpretation;
- The **observation model** (emissions);
- The **transition structure** and its constraints;
- The **inference or decision problem** being solved.

Zucchini et al. provide many application examples (e.g. animal movement, environmental data). Here we emphasize a few canonical areas.

13.1. 10.1 Speech Recognition

13.1.1. 10.1.1 Model Structure

In classical **speech recognition**, an HMM is used to model the mapping from hidden linguistic units to acoustic features:

- Hidden states S_t: phonetic units (phones), context-dependent phones, or sub-phonetic states;
- Observations Y_t: short-time acoustic feature vectors (e.g. MFCCs) in $\mathbb{R}^d$;
- Transition matrix Γ: encodes allowed transitions between phones (including self-transitions for duration modeling);
- Emission distributions $f_i(y)$: often Gaussian mixtures or more complex distributions over acoustic features.

13.1.2. 10.1.2 Inference Tasks

- **Likelihood computation:** $p_\theta(Y_{1:T})$ for a given sequence of acoustic features and a candidate word sequence;
- **Decoding:** find the most likely sequence of phones or words given observations (Viterbi);
- **Training:** MLE of HMM parameters via EM/Baum–Welch, often embedded inside larger systems (e.g. with language models).

Rabiner (1989) remains a classic reference for this application, describing HMMs as the central modeling tool for early speech systems.

13.2. 10.2 Bioinformatics

13.2.1. 10.2.1 CpG Island Detection

In genomics, HMMs can model regions with different **nucleotide composition**, such as **CpG islands**.

- Hidden states: $S_t \in \{\text{island}, \text{non-island}\}$;
- Observations: nucleotides $Y_t \in \{\text{A}, \text{C}, \text{G}, \text{T}\}$;
- Emissions: state-dependent multinomial distributions over nucleotides;
- Transitions: probabilities governing the length and frequency of CpG islands.

Inference tasks:

- **Decoding:** identify which positions belong to islands vs background (Viterbi or posterior decoding);
- **Parameter estimation:** learn emission probabilities and transition rates from annotated or unannotated sequences.

13.2.2. 10.2.2 Sequence Alignment and Profile HMMs

Profile HMMs generalize simple HMMs for **multiple sequence alignment**:

- States represent positions in an alignment (match, insert, delete);
- Emissions correspond to amino acids or nucleotides;
- Transitions model gaps and alignment patterns.

While structurally more complex, they are still HMMs with specialized topology.

13.3. 10.3 Finance and Econometrics

13.3.1. 10.3.1 Regime-Switching Models

In finance, HMMs model **regime changes** in returns (e.g. bull vs bear markets):

- Hidden states: $S_t \in \{1, \dots, K\}$ representing regimes (e.g. low-volatility vs high-volatility);
- Observations: asset returns $Y_t \in \mathbb{R}$ or $\mathbb{R}^d$;
- Emissions: state-dependent distributions, often Gaussian with mean μ_i and variance σ_i^2 per state i;
- Transitions: Markov matrix encoding persistence of regimes.

The model is

$$Y_t \mid S_t = i \sim \mathcal{N}(\mu_i, \sigma_i^2),$$

with (S_t) as in Section 1.

Inference tasks:

- **Filtering / smoothing:** posterior probabilities of regimes given returns, for risk management and forecasting;
- **Parameter estimation:** MLE via EM;
- **Regime-dependent decision-making:** portfolio allocation or hedging strategies that depend on inferred regimes.

13.3.2. 10.3.2 Markov-Switching Autoregressions

More generally, one can have **Markov-switching AR models** where

$$Y_t = \mu_{S_t} + \phi_{S_t} Y_{t-1} + \varepsilon_t,$$

with regime-dependent AR coefficients. This is an HMM in an extended state space and is closely related to **switching state-space models** (Section 7.3).

13.4. 10.4 Epidemiology and Latent Disease States

13.4.1. 10.4.1 Disease Progression Models

In epidemiology and biostatistics, HMMs can model **disease progression** where the true disease state is partially observed:

- Hidden states: discrete health states (e.g. healthy, infected, recovered) or stages (e.g. early, advanced);
- Observations: noisy test results, symptoms, biomarkers;
- Transitions: disease progression probabilities influenced by covariates (e.g. age, treatment).

The HMM structure is:

- S_t evolves as a Markov chain with transition matrix possibly depending on covariates;
- Y_t arises from state-dependent emission distributions (e.g. logistic regression for test outcomes).

Inference tasks:

- Estimating **transition probabilities** and **state occupancy** probabilities over time;
- Designing **screening and treatment policies** based on inferred states.

13.5. 10.5 General Modeling Pattern (Zucchini et al.)

Zucchini et al. emphasize a common pattern across applications:

1. **Choose a number of states** K and interpret them substantively (e.g. behavior modes, regimes);
2. Specify a **state process** (transition matrix, possibly with covariates);
3. Choose **emission distributions** compatible with the data type (discrete, continuous, circular, multivariate);
4. Fit the model via **MLE/EM** and evaluate via likelihood-based criteria and diagnostics;
5. Use decoding and posterior state probabilities for **interpretation** and **decision-making**.

13.6. 10.6 Summary

This section highlighted how the **abstract HMM framework** is instantiated in:

- **Speech recognition** (linguistic units → acoustic features);
- **Bioinformatics** (genomic regions, alignment profiles);
- **Finance** (market regimes and volatility states);
- **Epidemiology** (latent disease progression).

In all cases, the core mathematical machinery — **Markov chains**, **emission models**, and **inference algorithms** — is exactly that developed in Sections 1–5, as presented systematically in Zucchini et al.

14. Section 11 – Proof-Based Problem Sets for HMMs

This section provides **proof-oriented exercises** designed to consolidate a rigorous understanding of HMMs. Problems range from foundational probability to advanced asymptotic theory.

They are grouped by topic; many are inspired by or extend derivations in **Zucchini et al.**, **Cappé, Moulines, Rydén**, and **Douc, Moulines, Stoffer**.

No solutions are included here; these are intended for coursework, qualifying exams, or self-study at a graduate/PhD level.

14.1. 11.1 Probability and Markov Chains

1. **Sigma-algebras and conditional expectations.**
   Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space and X an integrable random variable. Show that the conditional expectation $\mathbb{E}[X \mid \mathcal{G}]$ with respect to a sub-σ-algebra $\mathcal{G} \subseteq \mathcal{F}$ is unique up to almost sure equality. Prove the tower property.

2. **Ergodic theorem for finite Markov chains.**
   Let (S_t) be an irreducible, aperiodic Markov chain on a finite state space with stationary distribution π. Prove that for any bounded function f,

   $$\frac{1}{T}\sum_{t=1}^{T} f(S_t) \xrightarrow{\text{a.s.}} \sum_i \pi_i f(i).$$

   (Hint: use coupling or spectral methods.)

3. **Spectral gap and mixing.**
   For a reversible Markov chain, prove that the total variation distance between $\mathbb{P}(S_t \in \cdot \mid S_0 = i)$ and π decays at least geometrically with rate determined by the spectral gap $\gamma = 1 - \lambda_2$.

14.2. 11.2 Inference Algorithms

4. **Forward algorithm correctness.**
   Starting from the HMM factorization, prove by induction that the forward recursion computes $\tilde{\alpha}_t(i) = \mathbb{P}(S_t = i, Y_{1:t} = y_{1:t})$.

5. **Forward–backward and smoothing.**
   Derive the backward recursion and show that the smoothing probabilities satisfy

$$\gamma_t(i) = \frac{\tilde{\alpha}_t(i)\beta_t(i)}{\sum_j \tilde{\alpha}_T(j)}.$$

6. **Viterbi optimality.**
   Prove rigorously that the Viterbi path is a maximizer of the joint probability $\mathbb{P}(S_{1:T}, Y_{1:T})$ by showing that the dynamic programming recursion satisfies the Bellman optimality principle.

7. **Comparison of path and marginal modes.**
   Construct an explicit example of a 2-state HMM and a short observation sequence where the sequence of marginally most probable states differs from the Viterbi path.

14.3. 11.3 EM, MLE, and Identifiability

8. **EM monotonicity.**
   Show that the EM update step satisfies

$$\ell(\theta^{(k+1)}) \geq \ell(\theta^{(k)}),$$

   by expressing the log-likelihood as the sum of an ELBO and a KL divergence (Section 5.2.4).

9. **Complete-data sufficient statistics.**
   For a finite-state HMM with discrete emissions, identify the complete-data sufficient statistics for δ, Γ, and emission probabilities. Derive EM update formulas starting from the exponential-family structure.

10. **Label switching.**
    Prove that permuting state labels in an HMM (and correspondingly permuting rows/columns of Γ and emission parameters) yields the same distribution for $Y_{1:T}$. Show that this is the only symmetry for generic parameter values.

11. **Non-identifiability example.**
    Construct a simple 2-state HMM with emission distributions and transition matrix such that two distinct parameter values (not related by permutation) induce the same distribution over $Y_{1:T}$ for all T.

14.4. 11.4 Asymptotics and Information

12. **Existence of limiting log-likelihood.**
    For a stationary ergodic HMM, show (under suitable conditions) that $\bar{\ell}_T(\theta) = T^{-1}\ell_T(\theta)$ converges almost surely to a limit $\ell_\infty(\theta)$ for each fixed θ.

13. **Consistency of MLE.**
    Outline a proof that $\hat{\theta}_T$ converges to the true parameter (up to permutation) by showing that $\ell_\infty(\theta)$ is uniquely maximized at θ^* and using uniform convergence of $\bar{\ell}_T$ to ℓ_∞.

14. **Asymptotic normality.**
    Derive the asymptotic distribution of $\sqrt{T}(\hat{\theta}_T - \theta^*)$ by applying a Taylor expansion to the score and invoking a central limit theorem for $U_T(\theta^*)$.

14.5. 11.5 Advanced and Alternative Perspectives

15. **Kalman filter as linear-Gaussian HMM.**
    Show that the Kalman filter recursion can be derived as the solution to the filtering problem in a linear-Gaussian state-space model, and compare it formally to the discrete-state forward algorithm.

16. **Nonparametric HMM identifiability (sketch).**
    Discuss conditions under which a nonparametric HMM with infinitely many states may still be identifiable from data (e.g. via finite-rank assumptions on certain operator kernels).

17. **POMDP belief MDP.**
    For a finite-state POMDP, prove that the process of belief states b_t forms a Markov decision process on the simplex, and write down the Bellman equations.

18. **Regret bounds for HMM predictors (conceptual).**
    Consider the class of HMM predictors under log-loss. Formulate the notion of regret against the best fixed HMM in hindsight and outline how a Bayesian mixture or online algorithm can achieve sublinear regret.

14.6. 11.6 Using These Problems

These problems are intended to be used alongside the main sections:

- 1–3 pair naturally with **Sections 0–1** (foundations and Markov chains);
- 4–7 with **Section 4** (inference algorithms);
- 8–11 with **Sections 5–6** (EM, identifiability, asymptotics);
- 15–18 with **Sections 7–9** (advanced models and alternative foundations).

Instructors can tailor subsets of these problems to build a full **graduate-level HMM course**, with Zucchini et al. as the primary applied reference and Cappé, Moulines, Rydén and Douc, Moulines, Stoffer providing the theoretical backbone.

Part VI.

Resources & Help

15. About This HMM Course

16. About the Hidden Markov Models (HMMs) Course

This site presents a **rigorous, proof-oriented course on Hidden Markov Models (HMMs)**. It is designed for:

- Graduate students in statistics, machine learning, or applied mathematics
- Researchers and advanced practitioners who want a mathematically honest treatment of HMMs
- Instructors who need a reference set of lecture-style notes and problem sets

The course emphasizes **calm clarity** over flash: clean typography, high-contrast math, and a modular layout so you can move between foundations, algorithms, theory, and applications without friction.

16.1. Pedagogical Philosophy

- **Theory-first, but example-driven.** Core results are stated and proved, with pointers to Zucchini et al. and more advanced monographs.
- **Separation of concerns.**
  - Section 0–1: probability and Markov chains
  - Section 2–3: model construction and likelihoods
  - Section 4–5: algorithms and estimation
  - Section 6–9: asymptotic theory and advanced variants
  - Section 10–11: applications and proof-based problem sets
- **Notation stability.** Notation is aligned as much as possible with Zucchini, MacDonald & Langrock to make cross-reading easy.

16.2. Who Should Use This Material

You will benefit most if you:

- Are comfortable with undergraduate probability and linear algebra
- Are willing to engage with proofs and derivations (not just code)
- Want to connect HMM algorithms to broader ideas in stochastic processes and statistical inference

If your background is lighter, start with **Section 0 (Mathematical Prerequisites)** and use the references to fill any gaps.

16.3. How This Site Is Structured

- **Home page:** Quick overview, value proposition, and module view of the course.
- **Overview (HMM.md):** A textual syllabus with references and links to all sections.
- **Sections 0–11:** Each section is a self-contained set of notes with definitions, theorems, and proof sketches.
- **Resources & Help:**
  - `About` (this page): context and intended audience
  - `FAQ`: practical questions on using the notes
  - `Contact`: how to suggest corrections or improvements

16.4. Primary References

This course is intentionally compatible with:

- Zucchini, MacDonald, Langrock – *Hidden Markov Models for Time Series: An Introduction Using R*.
- Cappé, Moulines, Rydén – *Inference in Hidden Markov Models*.
- Douc, Moulines, Stoffer – *Nonlinear Time Series: Theory, Methods and Applications*.
- Rabiner (1989) – *A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition*.

You can treat these notes as a **bridge** between the applied style of Zucchini et al. and the more measure-theoretic style of Cappé–Moulines–Rydén.

17. HMM Course FAQ

18. Frequently Asked Questions

18.1. What background do I need?

You should be comfortable with:

- Undergraduate probability (random variables, conditional probability, basic limit theorems)
- Linear algebra (eigenvalues, eigenvectors, basic spectral theory)
- Basic calculus and real analysis

Section 0 is designed to refresh **measure-theoretic language** just enough to make later sections precise.

18.2. Is this course focused on R code?

No. While the **notation** is aligned with Zucchini et al. (who use R for examples), these notes are **language-agnostic**. The emphasis is on:

- Mathematical formulation of HMMs
- Algorithms (forward–backward, Viterbi, EM) at the level of formulas and proofs
- Statistical theory (consistency, asymptotic normality, identifiability)

You can implement the algorithms in any language (R, Python, Julia, C++, etc.).

18.3. How should I study using this site?

A suggested path:

1. Read the **home page** and **HMM overview** to understand the big picture.
2. Work through **Sections 0–1** carefully if you are not fully comfortable with Markov chains.
3. Read **Sections 2–3** to understand the formal HMM model and likelihood.
4. Spend time with **Sections 4–5**, doing the derivations and proofs yourself.
5. Use **Sections 6–9** on a second pass for deeper statistical theory and advanced models.
6. Attempt problems from **Section 11** as if they were exam questions.

18.4. Are there solutions to the problem sets?

No solutions are included here. The problems in Section 11 are intended for:

- Graduate coursework and qualifying exams
- Reading groups and self-study

Instructors can prepare their own solution sets or ask students to present solutions.

18.5. How long does the course take?

As a rough guide:

- A 12–14 week semester course could spend **1–2 weeks per major block** (Foundations, Model & Inference, Estimation, Theory, Advanced Models, Applications/Problems).
- An intensive reading course could compress the material into **8–10 weeks** for well-prepared students.

18.6. Can I use these notes for teaching?

Yes, subject to whatever license you choose when publishing the repository. Typical uses:

- As a core set of lecture notes, supplemented with your own examples and code.
- As a reading list for graduate seminars.
- As background material for research students working on time-series or latent variable models.

If you use the notes in a course, consider adding a short remark in your syllabus pointing students to the site and to the primary references.

18.7. How do I report errors or suggest improvements?

See the **Contact** page for how to propose corrections or enhancements once the site is hosted (e.g., via GitHub issues or a simple contact form).

19. Contact & Feedback

20. Contact & Feedback

This HMM course site is designed as a living set of notes. Care has been taken to keep the mathematics and notation consistent, but **typos and gaps can still occur**.

20.1. How to Provide Feedback

Because this project is intended to be hosted from a version-controlled repository (e.g., GitHub), the recommended feedback channels are:

- **Issues:** Open an issue on the course repository describing:
  - The section (e.g., "Section 4 – Inference"),
  - The line or equation where the problem occurs,
  - A brief description of the error or suggested clarification.
- **Pull requests (advanced users):** If you are comfortable editing Markdown/Quarto, you can propose a fix directly and submit a pull request for review.

If you are using these notes in a private setting (not yet on GitHub), you can adapt this page with your preferred contact method (e.g., an academic email address or institutional LMS).

20.2. What Kind of Feedback Is Most Helpful?

- **Mathematical corrections:** Incorrect statements, missing assumptions, or unclear proofs.
- **Notation issues:** Inconsistencies with Zucchini et al. or between sections.
- **Clarity improvements:** Places where a short additional remark, example, or reference would significantly help understanding.
- **Typos and formatting:** Misrendered equations, broken links, or layout glitches.

20.3. A Note on Response Times

This site is intended as a resource rather than a commercial platform. Response times for issues or pull requests may vary. When hosted publicly, you can check the repository's issue tracker to see the status of open items.

If you are adapting these notes for your own course, feel free to fork the repository and modify them to suit your audience, while keeping appropriate attribution to the original references.