# Systematic Trading Strategies with Machine Learning Algorithms

## Latent Variable Models
## in Financial Asset Regime Detection

Imperial College Business School

May 1, 2025

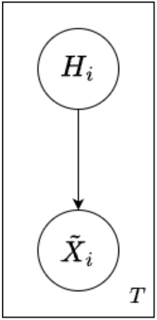
# Outline

# Outline

# Outline

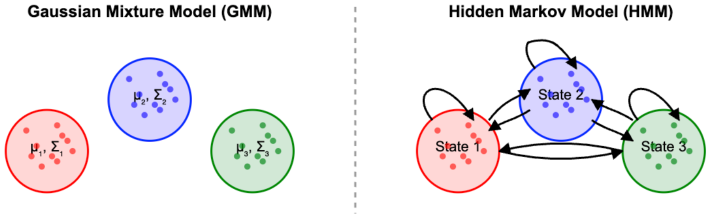
# Gaussian Mixture Models (GMMs)

- Goal: Represent complex, multimodal distributions using a mixture of Gaussians
- Latent variable: $H_i \sim \mathcal{M}(1, \pi_1, \ldots, \pi_M)$
- Observation: $\tilde{X}_i \mid H_i = m \sim \mathcal{N}(\mu_m, \Sigma_m)$
- EM algorithm used to estimate parameters $\theta = (\pi, \mu, \Sigma)$



Imperial College Business School

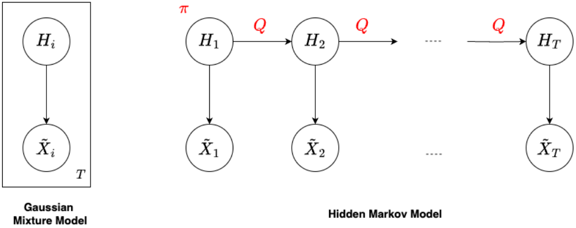
- GMMs assume hidden variables $H_1, \ldots, H_T$ are i.i.d.
- Ignores temporal structure in sequential data
- Need a model where:
  - Hidden states evolve over time
  - Observations depend on the current hidden state
- This leads to **Hidden Markov Models (HMMs)**



# Outline

Imperial College Business School

- Latent states: $H_1, \ldots, H_T$ with M possible hidden states.
- Observations: $\tilde{X}_1, \ldots, \tilde{X}_T \in \mathbb{R}^d$ or $\mathcal{O} = \{o_1, \ldots, o_D\}$
- **Markov property (transition model):**
  - The hidden state sequence forms a first-order Markov chain.
- **Emission independence:**
  - Observations are conditionally independent given the current hidden state:

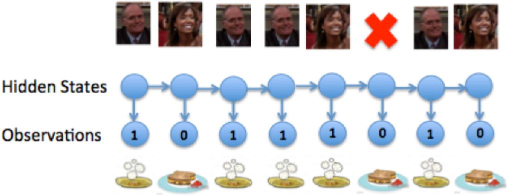


# Parameterization (Discrete Emissions)

- Observation space: $\mathcal{O} = \{o_1, \ldots, o_D\}$
- **Hidden state dynamics:**
  - Initial distribution: $\pi_m = P(H_1 = m)$
  - Transition matrix: $Q_{ij} = P(H_{t+1} = j \mid H_t = i)$
- **Emission distribution:**
  - Emission matrix: $O_{mj} = P(\tilde{X}_t = o_j \mid H_t = m)$
- Parameter set:

$$\theta = (\pi, Q, O)$$

- Example from Programming Session 3:



- Number of hidden states: $M$
- **Hidden state dynamics:**
  - Initial distribution: $\pi_m = P(H_1 = m)$
  - Transition matrix: $Q_{ij} = P(H_{t+1} = j \mid H_t = i)$
- **Emission model (continuous):**

$$\tilde{X}_t \mid H_t = m \sim \mathcal{N}_d(\mu_m, \Sigma_m)$$

- Parameter set:

$$\theta = (\pi, Q, \mu, \Sigma)$$

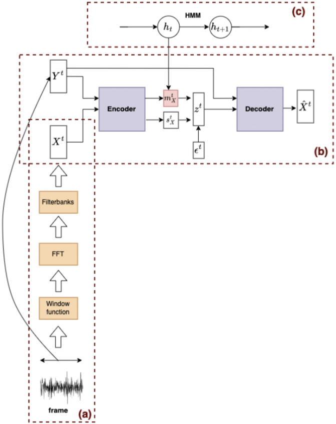



Figure: Forecasting Market Turbulence Regimes

# Outline

**Who Ate Ross's Sandwich?**
Dr. Ross Geller, a paleontologist at New York University, faces a peculiar dilemma.

- Every day, he brings a sandwich and stores it in the department refrigerator.
- His sandwich frequently disappears before lunch.
- He records what he can observe:
  - **0**: Sandwich is safe
  - **1**: Sandwich is missing

# An example - Emission Probabilities

Imperial College Business School

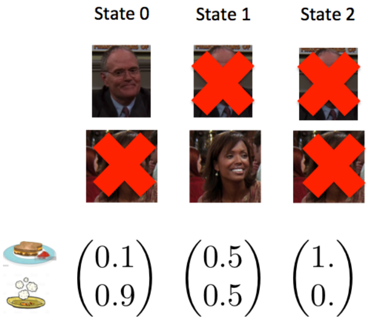
**Hidden States (Unobserved):**

- **State 0**: Dr. Donald is present — 90% chance sandwich is eaten
- **State 1**: Dr. Charlie is present — 50% chance sandwich is eaten
- **State 2**: Neither is present — 0% chance sandwich is eaten

**Observation space:**
$\mathcal{O} = \{0 : \text{Safe}, 1 : \text{Missing}\}$

**Emission matrix $O$:**

$$O = \begin{bmatrix} 0.1 & 0.9 \\ 0.5 & 0.5 \\ 1.0 & 0.0 \end{bmatrix}$$



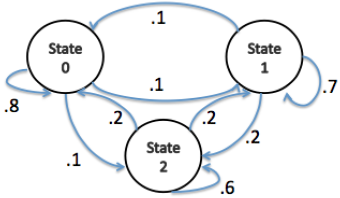
Imperial College Business School

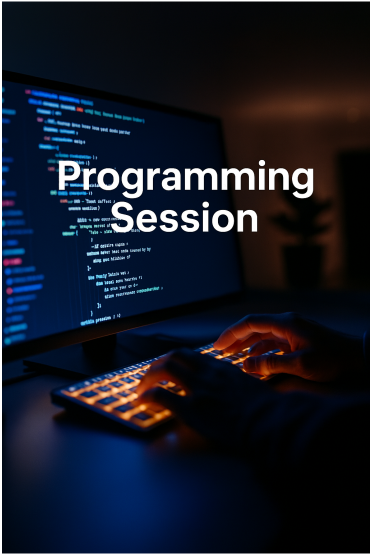
**Initial state distribution (uniform):**

$$\pi = \left[\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right]$$

**Transition matrix $Q$:**

$$Q = \begin{bmatrix} 0.8 & 0.1 & 0.1 \\ 0.1 & 0.7 & 0.2 \\ 0.2 & 0.2 & 0.6 \end{bmatrix}$$






**Programming Session 3: Section 1**

- Section 1: Create the Synthetic Data
- *Click here to access the programming session*

**Solution will be posted tonight on the GitHub page.**

- *Click here to access ccess the GitHub Page*


**Click here to participate in the poll**

# Outline

# Outline

# Objective 1: Inference Problem

**Goal:** Compute the likelihood of the observed sequence $\tilde{\mathbf{x}} = (\tilde{x}_1, \ldots, \tilde{x}_T)$ given the model parameters $\theta$.

We illustrate this in the case of **continuous emissions**, where each hidden state emits a multivariate Gaussian.

$$p_\theta(\tilde{\mathbf{x}}) = \sum_{h_1} \sum_{h_2} \cdots \sum_{h_T} \pi_{h_1} \prod_{t=1}^{T-1} Q_{h_t, h_{t+1}} \prod_{t=1}^{T} \mathcal{N}(\tilde{x}_t; \mu_{h_t}, \Sigma_{h_t})$$

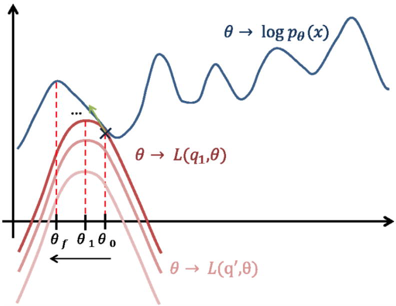
This computation is intractable due to the exponential number of hidden state paths ($M^T$ sequences).

**Approach:** Use a recursive and efficient approach (**The Forward Algorithm**).

Imperial College Business School

**Goal:** Estimate the model parameters $\theta = (\pi, Q, \mu, \Sigma)$ from a sequence of observations $\tilde{\mathbf{x}}$.

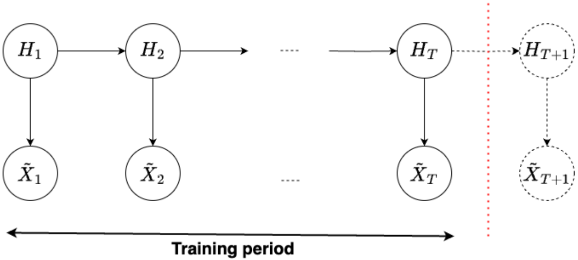
**Approach:** Expectation-Maximization (EM) algorithm:



**Goal:** Compute the distribution over hidden states at time $T+1$ given past observations and learned parameters.

**Approach:** Once the model is trained using the EM algorithm, we can use the filtering probabilities $[\xi(T,h')]_{1\leq h'\leq M}$ (which will be introduced later) to make the prediction:

$$p(H_{T+1} = h \mid \tilde{X}_1 = \tilde{x}_1, \ldots, \tilde{X}_T = \tilde{x}_T) \quad \forall h \in [\![1, M]\!]$$



# Outline

To solve these problems, we introduce the following probabilities:

- **Filtering probabilities:** $\xi \in \mathbb{R}^{T \times M}$

$$\forall (t, h) \in [\![1, T]\!] \times [\![1, M]\!],$$
$$\xi(t, h) := p(H_t = h \mid \tilde{X}_1, \dots, \tilde{X}_t)$$

- **Smoothing probabilities:**
  - $\psi \in \mathbb{R}^{T \times M}$

$$\forall (t, h) \in [\![1, T]\!] \times [\![1, M]\!],$$
$$\psi(t, h) := p(H_t = h \mid \tilde{X}_1, \dots, \tilde{X}_T)$$

  - $\phi \in \mathbb{R}^{T-1 \times M \times M}$

$$\forall (t, h, h') \in [\![1, T-1]\!] \times [\![1, M]\!]^2,$$
$$\phi(t, h, h') := p(H_t = h, H_{t+1} = h' \mid \tilde{X}_1, \dots, \tilde{X}_T)$$

# Forward Propagation: Filtering Probabilities

- **Goal:** Compute filtering probabilities:

$$\forall (t,h) \in [\![1,T]\!] \times [\![1,M]\!], \quad \xi(t,h) := p(H_t = h \mid \tilde{X}_1, \dots, \tilde{X}_t)$$

- We introduce the **alpha variables**, which correspond to the joint probability of the observed sequence up to time $t$ and the hidden state at time $t$:

$$\forall (t,h) \in [\![1,T]\!] \times [\![1,M]\!], \quad \alpha(t,h) := p(\tilde{X}_1, \dots, \tilde{X}_t, H_t = h)$$

- We also introduce the **emission tensor** $\Gamma(t)$, a diagonal matrix encoding the likelihood of the current observation conditioned on each possible hidden state:

$$\forall t \in [\![1,T]\!] : \Gamma(t) := \begin{pmatrix} p(\tilde{X}_t \mid H_t = 1) & & \\ & \ddots & \\ & & p(\tilde{X}_t \mid H_t = M) \end{pmatrix} \in \mathbb{R}^{M \times M}$$

Imperial College Business School

- **Vector notation:**

$$\alpha_t = (\alpha(t,1), \dots, \alpha(t,M))^T \in \mathbb{R}^M$$
$$\xi_t = (\xi(t,1), \dots, \xi(t,M))^T \in \mathbb{R}^M$$

- **Key results**: *Click here for the detailed calculations*
  - Forward Propagation: Recursive calculation of $\alpha$ variables:

$$\boxed{\alpha_1 = \Gamma(1)\pi, \quad \forall t \geq 2 : \alpha_t = \Gamma(t) Q^T \alpha_{t-1}}$$

  - Filtering probabilities from alpha:

$$\boxed{\forall t \in [\![1, T]\!] \quad \xi_t = \frac{\alpha_t}{\mathbb{1}_M^T \alpha_t}}$$

  - Solving Objective 1: $\boxed{p(\tilde{\mathbf{x}}) = \mathbb{1}_M^T \alpha_T}$

# Backward Propagation: Smoothing Probabilities

- **Goal:** Compute smoothing probabilities:

$$\forall (t,h) \in [\![1,T]\!] \times [\![1,M]\!], \quad \psi(t,h) := p(H_t = h \mid \tilde{X}_{1:T})$$
$$\forall (t,h,h') \in [\![1,T]\!] \times [\![1,M]\!]^2, \quad \phi(t,h,h') := p(H_t = h, H_{t+1} = h' \mid \tilde{X}_{1:T})$$

- We introduce the **beta variables**, which represent the likelihood of future observations given the current state:

$$\forall (t,h) \in [\![1,T]\!] \times [\![1,M]\!], \quad \beta(t,h) := p(\tilde{X}_{t+1:T} \mid H_t = h)$$

- Vector / Matrix notation:

$$\beta_t = (\beta(t,1), \ldots, \beta(t,M))^T \in \mathbb{R}^M$$
$$\psi_t = [\psi(t,h)]_h \in \mathbb{R}^M, \quad \phi_t = [\phi(t,h,h')]_{h,h'} \in \mathbb{R}^{M \times M}$$

- **Key results**: *Click here for the detailed calculations*
  - Backward Propagation: Recursive calculation of $\beta$ variables:

$$\boxed{\beta_T = \mathbb{1}_M, \quad \forall t \leq T-1 : \beta_t = Q\Gamma(t+1)\beta_{t+1}}$$

  - Smoothing probability using $\alpha$ and $\beta$:

$$\boxed{\forall t \in [\![1, T]\!] \quad \psi_t = \frac{\alpha_t \circ \beta_t}{\mathbb{1}_M^T \alpha_T}}$$

  - Joint smoothing probability:

$$\boxed{\forall t \in [\![1, T]\!] \quad \phi_t = \frac{\text{diag}(\alpha_t) Q\Gamma(t+1)\text{diag}(\beta_{t+1})}{\mathbb{1}_M^T \alpha_T}}$$

# Outline

Imperial College Business School

**Algorithm EM Algorithm**

**Require:** Observations $\tilde{X} = \{\tilde{x}_1, \ldots, \tilde{x}_T\}$
**Ensure:** Optimal parameters $\theta$
1: **Initialization:** Choose initial parameters $\theta^{(0)}$.
2: **while** not converged **do**
3: **E-step:** Update $q$ to maximize the lower bound with respect to $q$

$$q_{t+1} \in \arg\max_q \left(\mathcal{L}(q, \theta_t)\right)$$

4: **M-step:** Update $\theta$ to maximize the lower bound with respect to $\theta$

$$\theta_{t+1} \in \arg\max_\theta \left(\mathcal{L}(q_{t+1}, \theta)\right)$$

5: **end while**
6: **return** Optimized parameters $\theta^*$

Imperial College Business School

- We apply EM to estimate parameters of an HMM given observations $\tilde{X}_{1:T}$.
- The E-step computes the expected complete log-likelihood with respect to the posterior distribution over hidden states:

$$\mathbb{E}_{\mathbf{h}|\tilde{\mathbf{x}}}[\log p_\theta(\mathbf{h}, \tilde{\mathbf{x}})]$$

- At iteration $i$, we decompose the complete log-likelihood as:

$$\log(p_{\theta^{(i)}}(\mathbf{h}, \tilde{\mathbf{x}})) = \log(\pi^{(i)}_{h_1}) + \sum_{t=1}^{T-1} \log(Q^{(i)}_{h_t, h_{t+1}}) + \sum_{t=1}^{T} \log(\mathcal{N}(\tilde{X}_t; \mu^{(i)}_{h_t}, \Sigma^{(i)}_{h_t}))$$

# EM for HMMs: The E-Step

The expected log-likelihood is computed with respect to the posterior distribution:

$$\mathbb{E}_{H|\tilde{\mathbf{x}}}[\log(\pi_{h_1})] = \sum_{h=1}^{M} \log(\pi_h) p(H_1 = h \mid \tilde{\mathbf{x}})$$

$$\mathbb{E}_{H|\tilde{\mathbf{x}}}[\log(Q_{h_t,h_{t+1}})] = \sum_{h=1}^{M} \sum_{h'=1}^{M} \log(Q_{hh'}) p(H_t = h, H_{t+1} = h' \mid \tilde{\mathbf{x}})$$

$$\mathbb{E}_{H|\tilde{\mathbf{x}}}[\log(\mathcal{N}(\tilde{X}_t; \mu_{h_t}, \Sigma_{h_t}))] = \sum_{h=1}^{M} \log(\mathcal{N}(\tilde{X}_t; \mu_h, \Sigma_h)) p(H_t = h \mid \tilde{\mathbf{x}})$$

# EM for HMMs: The E-Step

Imperial College Business School

- We use the smoothing probabilities $\psi$ and $\phi$, which are computed using the **Forward-Backward algorithm**.
- This allows us to compute the expected complete log-likelihood w.r.t the posterior distribution:

$$\mathbb{E}_{\mathbf{h}|\tilde{\mathbf{x}}}[\log p_\theta(\mathbf{h}, \tilde{\mathbf{x}})] = \sum_{h=1}^{M} \log(\pi_h)\psi(1, h) + \sum_{t=1}^{T-1} \sum_{h,h'} \log(Q_{hh'})\phi(t, h, h') + \sum_{t=1}^{T} \sum_{h=1}^{M} \log(\mathcal{N}(\tilde{X}_t; \mu_h, \Sigma_h))\psi(t, h)$$

- *Click here for the detailed calculations*

# EM for HMMs: The M-Step

- Maximize the expected log-likelihood w.r.t. parameters $\theta$.
- Parameter update rules:

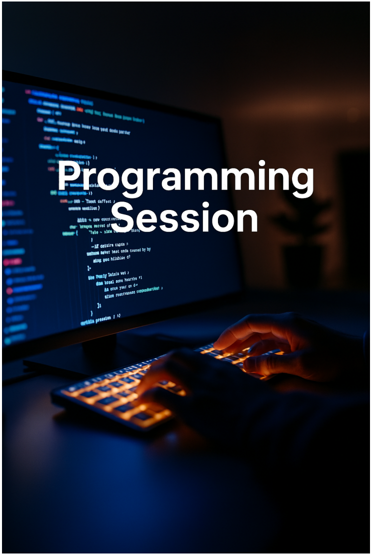
$$\pi_h^{(i+1)} = \psi(1, h)$$

$$Q_{h,h'}^{(i+1)} = \frac{\sum_{t=1}^{T-1} \phi(t, h, h')}{\sum_{t=1}^{T-1} \psi(t, h)}$$

$$\mu_h^{(i+1)} = \frac{\sum_{t=1}^{T} \psi(t, h)\tilde{X}_t}{\sum_{t=1}^{T} \psi(t, h)}$$

$$\Sigma_h^{(i+1)} = \frac{\sum_{t=1}^{T} \psi(t, h)(\tilde{X}_t - \mu_h^{(i)})(\tilde{X}_t - \mu_h^{(i)})^T}{\sum_{t=1}^{T} \psi(t, h)}$$

- *Click here for the detailed calculations*




**Programming Session 3: Section 2**

- Section 2: The Learning Problem - EM Algorithm
- *Click here to access the programming session*

**Solution will be posted tonight on the GitHub page.**

- *Click here to access ccess the GitHub Page*

# Feedback Poll

**Click here to participate in the poll**

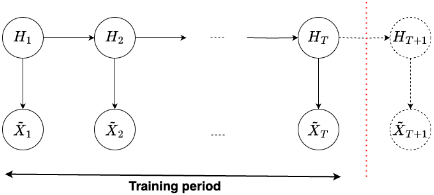
Imperial College Business School

# Predicting the Next Hidden State

- After training the model with the EM algorithm, we want to predict the distribution over hidden states at the next time step $T+1$.
- Objective: Compute

$$\forall h \in [\![1, M]\!], \quad p(H_{T+1} = h \mid \tilde{X}_1 = \tilde{x}_1, \ldots, \tilde{X}_T = \tilde{x}_T)$$

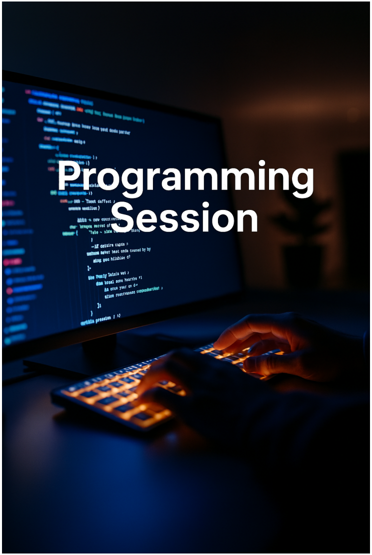
- This is useful for forecasting future regimes such as market turbulence.



- Using the filtering probabilities $\xi(T, \cdot)$, we derive for all $h \in [\![1, M]\!]$:

$$
\begin{aligned}
p(H_{T+1} = h \mid \tilde{X}_{1:T}) &= \sum_{h'=1}^{M} p(H_{T+1} = h, H_T = h' \mid \tilde{X}_{1:T}) \\
&= \sum_{h'=1}^{M} \underbrace{p(H_{T+1} = h \mid H_T = h')}_{= Q_{h'h}} \cdot \underbrace{p(H_T = h' \mid \tilde{X}_{1:T})}_{= \xi(T, h')}
\end{aligned}
$$

- The filtering probabilities are computed using the Forward algorithm introduced earlier.




**Programming Session 3: Section 3**

- Section 3: Predicting the next Hidden State / Next Observation
- *Click here to access the programming session*

**Solution will be posted tonight on the GitHub page.**

- *Click here to access ccess the GitHub Page*

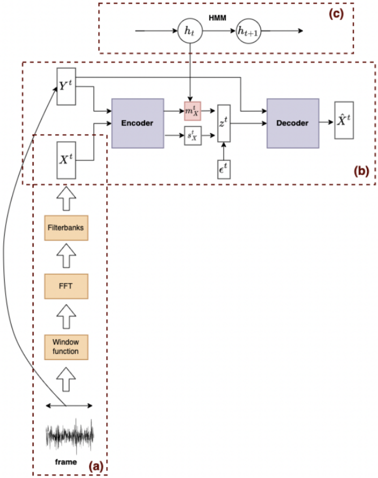
# Outline

**Summary:** The Low Turbulence Model processes return data to detect stable market regimes:

- Extract spectral information (Step a). [3]
- Learn low-dimensional structure (Step b). [1]
- Forecast regimes with HMMs (Step c). [4]

The model outputs regime probabilities used in asset allocation.

See [2] for full methodology and results.




**Click here to participate in the poll**

# Quiz Time!

**Click here to take the quiz**

**Thank you for your attention**

# References I

[1] Diederik P Kingma, Max Welling, et al. *Auto-encoding variational bayes*. 2013.

[2] Hachem Madmoun. “Creating Investment Strategies Based on Machine Learning Algorithms”. PhD thesis. École des Ponts ParisTech, 2022.

[3] Stéphane Mallat. *A wavelet tour of signal processing*. Elsevier, 1999.

[4] Lawrence R Rabiner. “A tutorial on hidden Markov models and selected applications in speech recognition”. In: *Proceedings of the IEEE* 77.2 (1989), pp. 257–286.