Sticky HDP-HMM-VAR Model

By Zhenning Zhao

Sticky HDP-HMM-VAR Model

3.5 - Chinese Restaurant Franchise with Loyal Customer

3.5 - Direct Assignment Rao–Blackwellized Gibbs Sampler for Sticky HDP-HMM

Section 0 - Bayesian Estimation

0.1 - Multivariate Normal Distribution

The **Bayesian estimation** of the **multivariate normal distribution** involves placing a **conjugate prior** over the unknown parameters — typically the mean vector μ and the covariance matrix Σ. Assume we have n observations $x_1, \ldots, x_n$, each $x_i \in \mathbb{R}^d$, and we model:

$$x_i \sim \mathcal{N}(\mu, \Sigma)$$

Conjugate Prior (Normal-Inverse-Wishart)

The conjugate prior for (μ, Σ) is the **Normal-Inverse-Wishart** distribution:

- $\Sigma \sim \mathcal{IW}(\nu_0, \Lambda_0)$ (Inverse-Wishart)
- $\mu \mid \Sigma \sim \mathcal{N}(\mu_0, \Sigma/\kappa_0)$

Where:

- μ_0 is the prior mean,
- κ_0 is the prior strength (pseudo-count),
- ν_0 is the degrees of freedom,
- Λ_0 is the scale matrix.

Posterior Distribution

Given n samples with empirical mean $\bar{x}$ and empirical scatter matrix $S = \sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{x})^\top$, the posterior is:

- $\Sigma \sim \mathcal{IW}(\nu_n, \Lambda_n)$
- $\mu \mid \Sigma \sim \mathcal{N}(\mu_n, \Sigma/\kappa_n)$

Where the posterior hyperparameters are:

- $\kappa_n = \kappa_0 + n$
- $\nu_n = \nu_0 + n$

- $\mu_n = \frac{\kappa_0 \mu_0 + n\bar{x}}{\kappa_0 + n}$
- $\Lambda_n = \Lambda_0 + S + \frac{\kappa_0 n}{\kappa_0 + n}(\bar{x} - \mu_0)(\bar{x} - \mu_0)^\top$

This gives a closed-form Bayesian update for the multivariate normal distribution with unknown mean and covariance.

0.2 - Bayesian Linear Regression

In **Bayesian linear regression**, we place a **conjugate prior** over the regression coefficients β and the noise variance σ^2 in the linear model:

$$y = X\beta + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma^2 I)$$

where:

- $y \in \mathbb{R}^n$ is the response vector,
- $X \in \mathbb{R}^{n \times p}$ is the design matrix,
- $\beta \in \mathbb{R}^p$ are the regression coefficients,
- σ^2 is the error variance.

Conjugate Prior (Normal-Inverse-Gamma)

The conjugate prior for (β, σ^2) is a **Normal-Inverse-Gamma** distribution:

- $\beta \mid \sigma^2 \sim \mathcal{N}(\beta_0, \sigma^2 V_0)$
- $\sigma^2 \sim \text{Inverse-Gamma}(a_0, b_0)$

Where:

- β_0 is the prior mean for β,
- V_0 is the prior covariance (scaled by σ^2),
- a_0, b_0 are the shape and scale parameters of the inverse-gamma prior on σ^2.

Posterior Distribution

Given the data (X, y), the posterior distributions are:

- $\beta \mid \sigma^2, y \sim \mathcal{N}(\beta_n, \sigma^2 V_n)$
- $\sigma^2 \mid y \sim \text{Inverse-Gamma}(a_n, b_n)$

With updated parameters:

- $V_n = (X^\top X + V_0^{-1})^{-1}$
- $\beta_n = V_n(X^\top y + V_0^{-1}\beta_0)$
- $a_n = a_0 + \frac{n}{2}$
- $b_n = b_0 + \frac{1}{2}\left(y^\top y + \beta_0^\top V_0^{-1}\beta_0 - \beta_n^\top V_n^{-1}\beta_n\right)$

This provides a full closed-form Bayesian update for both the regression coefficients and the variance of the noise. The posterior distribution can be used for inference, prediction, and uncertainty quantification.

0.3 Bayesian Estimation of Multivariate Linear Regression

In the **Bayesian estimation of multivariate linear regression**, we model a matrix-valued response $Y \in \mathbb{R}^{n \times k}$ as a linear function of predictors $X \in \mathbb{R}^{n \times m}$ with matrix-valued coefficients $\beta \in \mathbb{R}^{m \times k}$:

$$Y = X\beta + E, \quad E \sim \mathcal{MN}_{n \times k}(0, I_n, \Sigma)$$

Here, the error matrix E is drawn from a **matrix normal distribution** with row covariance I_n and column covariance $\Sigma \in \mathbb{R}^{k \times k}$. This implies:

$$\text{vec}(Y) \sim \mathcal{N}(\text{vec}(X\beta), \Sigma \otimes I_n)$$

Conjugate Prior (Matrix Normal - Inverse-Wishart)

We place conjugate priors over both β and Σ:

- $\Sigma \sim \mathcal{IW}(\nu_0, S_0)$
- $\beta \mid \Sigma \sim \mathcal{MN}_{m \times k}(M_0, V_0, \Sigma)$

Where:

- $M_0 \in \mathbb{R}^{m \times k}$ is the prior mean of β,
- $V_0 \in \mathbb{R}^{m \times m}$ is the row covariance (shared across all columns),
- Σ is the column covariance (shared with likelihood),
- ν_0 is the prior degrees of freedom,

- $S_0 \in \mathbb{R}^{k \times k}$ is the prior scale matrix.

Posterior Distribution

Given data matrices (X, Y) with n samples, define:

- $V_n^{-1} = V_0^{-1} + X^\top X$
- $M_n = V_n(V_0^{-1} M_0 + X^\top Y)$
- $\nu_n = \nu_0 + n$
- $S_n = S_0 + Y^\top Y + M_0^\top V_0^{-1} M_0 - M_n^\top V_n^{-1} M_n$

Then the posterior distributions are:

- $\Sigma \mid Y, X \sim \mathcal{IW}(\nu_n, S_n)$
- $\beta \mid \Sigma, Y, X \sim \mathcal{MN}_{m \times k}(M_n, V_n, \Sigma)$

These formulas provide a closed-form Bayesian update for both the regression coefficients and the covariance of multivariate Gaussian noise.

Section 1 - Dirichlet Process

1.1 - Dirichlet Process

A **Dirichlet Process (DP)** is a distribution over distributions. It is commonly used in **Bayesian nonparametric models** where the number of parameters (e.g., clusters) is not fixed in advance.

Formally, a Dirichlet Process is defined as:

$$G \sim \mathrm{DP}(\gamma, H)$$

- G: A random probability measure.
- γ: Concentration parameter.
- H: Base distribution (mean of the DP).

1.2 - Stick Breaking Process

Imagine a stick of unit length (i.e., total length = 1). We iteratively break off a portion of the stick to assign probabilities. At each step:

1. Draw a value β_k from a distribution, typically $\beta_k \sim \mathrm{Beta}(1, \alpha)$.
2. Break off a fraction β_k of the remaining stick.
3. The length of the broken piece is assigned as the weight (or probability) π_k for the k-th component.
4. The remaining stick is used in the next iteration.

Mathematically, the weights $\{\pi_k\}_{k=1}^{\infty}$ are defined as:

$$\begin{aligned}
\pi_1 &= \beta_1 \\
\pi_2 &= \beta_2(1-\beta_1) \\
\pi_3 &= \beta_3(1-\beta_1)(1-\beta_2) \\
&\vdots \\
\pi_k &= \beta_k \prod_{j=1}^{k-1}(1-\beta_j)
\end{aligned}$$

Each π_k represents the probability of drawing a sample from a corresponding base distribution H.

1.3 - Chinese Restaurant Process

Suppose we are running a Chinese restaurant with an infinite number of tables. When a customer arrives, we label them as customer i. The seating rule is as follows:

- The customer chooses an existing table t with probability proportional to the number of customers already sitting at that table (n_t).
- Alternatively, the customer may start a **new table** with probability proportional to a concentration parameter $\gamma > 0$.

Formally, if there are n customers already seated and table t has n_t customers, then:

- Probability of choosing table t:

$$P(\text{customer } i \text{ sits at table } t) = \frac{n_t}{i - 1 + \gamma}$$

- Probability of starting a new table:

$$P(\text{customer } i \text{ starts a new table}) = \frac{\gamma}{i - 1 + \gamma}$$

Each table t is associated with a unique dish θ_t, drawn from a base distribution H:

$$\theta_t \sim H$$

This process results in customers being grouped into clusters (tables), with the number of clusters growing logarithmically with the number of customers.

Theorem 1.3.1

The distribution over partitions generated by the Chinese Restaurant Process is **marginally equivalent** to sampling from a Dirichlet Process, i.e.

$$\theta_i \sim G, \quad G \sim \mathrm{DP}(\gamma, H)$$

Theorem 1.3.2

As the number of customers $n \to \infty$, the number of occupied tables K_n grows as:

$$\mathbb{E}[K_n] \sim \gamma \log(n)$$

1.4 - Gibbs Sampler

The **Gibbs Sampler** is a Markov Chain Monte Carlo (MCMC) technique used to approximate posterior distributions when direct sampling is difficult. In the context of a Dirichlet Process Mixture Model (DPMM), the Gibbs sampler iteratively assigns data points to clusters (tables), updating assignments based on the current state of the model.

Each iteration involves two steps for each data point x_i:

1. **Remove x_i from its current cluster.**
   Update the counts of the cluster. If this was the only member of its cluster, remove the cluster entirely.
2. **Sample a new cluster assignment for x_i**, based on:
   - The **likelihood** of x_i under existing cluster parameters θ_t.
   - The **prior probability** given by the CRP.

The conditional probability of assigning x_i to an existing cluster t is:

$$P(c_i = t \mid c_{-i}, X) \propto n_t \cdot p(x_i \mid \theta_t)$$

The probability of assigning x_i to a **new cluster** is:

$$P(c_i = \text{new} \mid c_{-i}, X) \propto \gamma \cdot \int p(x_i \mid \theta)\, dH(\theta)$$

After assignments are updated, we resample the cluster parameters θ_t from their posterior:

$$\theta_t \sim p(\theta \mid \{x_i : c_i = t\})$$

This iterative process eventually converges to samples from the posterior distribution over the clusters under the DP mixture model.

1.4 - Prediction

In a Dirichlet Process Mixture Model where each cluster is modeled by a **Multivariate Normal distribution** with a conjugate **Normal-Inverse-Wishart prior**, we can make predictions for new or partially observed data points.

Prediction for Partially Observed Multivariate Normal Variables

Suppose each data point $x_i \in \mathbb{R}^d$ is modeled as:

$$x_i \sim \mathcal{N}(\mu, \Sigma)$$

and we partition the variables into two subsets:

- $x = \begin{bmatrix} x^{(o)} \\ x^{(m)} \end{bmatrix}$, where:
  - $x^{(o)} \in \mathbb{R}^{d_o}$: observed entries
  - $x^{(m)} \in \mathbb{R}^{d_m}$: missing (or predicted) entries

Given:

- Posterior estimates for the mean μ and covariance Σ (possibly from cluster assignment under DPMM),
- The observed values $x^{(o)}$,

We can derive the **posterior predictive distribution** of $x^{(m)}$ as a conditional multivariate normal distribution:

$$x^{(m)} \mid x^{(o)} \sim \mathcal{N}(\mu^{(m|o)}, \Sigma^{(m|o)})$$

Where:

- **Conditional mean**:

$$\mu^{(m|o)} = \mu^{(m)} + \Sigma^{(mo)}(\Sigma^{(oo)})^{-1}(x^{(o)} - \mu^{(o)})$$

- **Conditional covariance**:

$$\Sigma^{(m|o)} = \Sigma^{(mm)} - \Sigma^{(mo)}(\Sigma^{(oo)})^{-1}\Sigma^{(om)}$$

Here:

- $\mu^{(m)}$, $\mu^{(o)}$ are the subcomponents of the mean vector,
- $\Sigma^{(mm)}$, $\Sigma^{(oo)}$, $\Sigma^{(mo)}$, $\Sigma^{(om)}$ are the relevant partitions of the covariance matrix.

Predictive Distribution in DPMMs

In a **Dirichlet Process Gaussian Mixture Model (DP-GMM)** where each data point may be **partially observed**, the predictive distribution for a new point $x_{n+1}^{(o)}$ (only partially observed) must integrate over the **unobserved entries** and the **latent cluster assignment**.

Let:

- $x_{n+1}^{(o)}$: the **observed** part of the new data point,
- $x_{n+1}^{(m)}$: the **missing** part (unobserved),
- $x_{n+1} = \begin{bmatrix} x_{n+1}^{(o)} \\ x_{n+1}^{(m)} \end{bmatrix}$

Then the **predictive distribution** over the observed components is:

$$p(x_{n+1}^{(o)} \mid x_{1:n}) = \sum_{t=1}^{K} \frac{n_t}{n+\gamma} \mathcal{N}\left(x_{n+1}^{(o)} \mid \mu_t^{(o)}, \Sigma_t^{(oo)}\right) + \frac{\gamma}{n+\gamma} \int \mathcal{N}\left(x_{n+1}^{(o)} \mid \mu^{(o)}, \Sigma^{(oo)}\right) dH(\mu, \Sigma)$$

Where:

- $\mu_t^{(o)}$, $\Sigma_t^{(oo)}$: mean and covariance **restricted to the observed dimensions** for cluster t,
- H: the **Normal-Inverse-Wishart prior** over (μ, Σ),
- K: number of current clusters,
- n_t: number of points assigned to cluster t,
- γ: concentration parameter.

Section 2 - Hierarchical Dirichlet Process

2.1 - Hierarchical Dirichlet Process (HDP)

The **Hierarchical Dirichlet Process (HDP)** extends the Dirichlet Process to **grouped data**.

- In many settings (e.g., documents, patients, users), we want to share clusters across groups but still allow each group to have its own mixture of components.
- HDP enables **sharing of topics/clusters** across groups using a hierarchy of DPs.

Let:

- H: base distribution
- γ: concentration for the global DP
- α: concentration for each group-level DP

The HDP is defined as:

1. Global distribution:

$$G_0 \sim \mathrm{DP}(\gamma, H)$$

2. For each group j:

$$G_j \sim \mathrm{DP}(\alpha, G_0)$$

Each G_j is a discrete distribution, **sharing atoms** with G_0, hence inducing shared clusters across groups.

2.2 - Chinese Restaurant Franchise (CRF)

The **Chinese Restaurant Franchise (CRF)** is the metaphor used to describe the sampling process in an HDP.

- Each **restaurant** corresponds to a group (e.g., a document).
- Each **customer** corresponds to an element (e.g., a word).

- Each **table** corresponds to a cluster (e.g., a bunch of words).
- Each restaurant has an **infinite number of tables**.
- Each table serves a **dish** (e.g., a theme), which is shared across restaurants.
- Dishes come from a global menu (distribution over clusters).

For customer i in restaurant j :

1. Choose a **table** t:
    - Probability proportional to number of customers already at table.
    - Or sit at a new table with probability proportional to α.
2. If sitting at a **new table**, choose a **dish** (cluster) k:
    - Probability proportional to the number of tables across all restaurants serving dish k.
    - Or choose a new dish from base distribution H with probability proportional to γ.
    - Dish can repeat within a restaurant at different tables.

2.3 - Gibbs Sampling

We perform **Gibbs sampling** for inference in the HDP using the Chinese Restaurant Franchise (CRF) representation. The key idea is to marginalize out the random measures G_0 and G_j, and sample the latent table and dish assignments for each observation (e.g., each word in a document).

Let:

- x_{ji}: the i-th observation in group j
- t_{ji}: table assignment for x_{ji}
- k_{jt}: dish (cluster) served at table t in group j
- ϕ_k: parameters for dish k, drawn from base distribution H

The sampling steps are:

Step 1: Sample Table Assignments t_{ji}

For each observation x_{ji}, remove it temporarily from the current table and update counts.

- Let n_{jt}^{-ji}: number of customers at table t in restaurant j, excluding x_{ji}
- Let $\phi_{k_{jt}}$: parameter of the dish served at table t

The **posterior probability** of assigning x_{ji} to an existing table t is:

$$P(t_{ji} = t \mid \cdots) \propto n_{jt}^{-ji} \cdot f(x_{ji} \mid \phi_{k_{jt}})$$

The **probability of creating a new table** is:

$$P(t_{ji} = \text{new}) \propto \alpha \cdot (\sum_k \frac{m_{\cdot k}}{m_{\cdot\cdot} + \gamma} f(x_{ji} \mid \phi_k) + \frac{\gamma}{m_{\cdot\cdot} + \gamma} \int f(x_{ji} \mid \phi)\, dH(\phi))$$

Where:

- $m_{\cdot k}$: number of tables (across all restaurants) serving dish k
- $m_{\cdot\cdot}$: total number of tables across all restaurants

If a **new table** is created, we sample its **dish** (cluster) in Step 2.

Step 2: Sample Dish Assignments k_{jt}

For each table t in each restaurant j, given all observations assigned to it:

- Let $x_{jt}^{(1)}, \ldots, x_{jt}^{(n_{jt})}$ be the data assigned to table t

The posterior probability that table t is assigned to **existing dish** k is:

$$P(k_{jt} = k \mid \cdots) \propto m_{\cdot k}^{-jt} \cdot \int \left(\prod_{i=1}^{n_{jt}} f(x_{jt}^{(i)} \mid \phi_k) \right) dH(\phi_k)$$

Where $m_{\cdot k}^{-jt}$ is the number of tables (excluding current table t) assigned to dish k.

The probability of assigning table t to a **new dish** is:

$$P(k_{jt} = \text{new}) \propto \gamma \cdot \int \left(\prod_{i=1}^{n_{jt}} f(x_{jt}^{(i)} \mid \phi) \right) dH(\phi)$$

This integral is the **marginal likelihood**, and if H is conjugate to the likelihood model f, it can be computed analytically.

Step 3: Sample Dish Parameters ϕ_k

For each dish k, given all observations assigned to it, sample the parameter ϕ_k from its posterior:

$$\phi_k \sim P(\phi \mid \{x_{ji} : k_{jt_{ji}} = k\}, H)$$

If H is conjugate to the likelihood model f, this is straightforward. Otherwise, one can use Metropolis-Hastings or other MCMC methods to sample ϕ_k.

Remarks:

- **Collapsed sampling** is often used: marginalize out ϕ_k to sample assignments more efficiently.
- HDP Gibbs sampling is **nonparametric**, and the number of clusters can grow with data.
- Common choices for H and f: in topic models, H is typically a Dirichlet distribution over word probabilities, and f is a multinomial distribution.

2.4 - HDP Document Topic Classifier

Model Structure Recap

Let:

- x_{ji}: the i-th word in document j
- t_{ji}: the table assignment of word x_{ji}
- k_{jt}: topic (dish) assigned to table t in document j
- ϕ_k: the word distribution for topic k, drawn from a base Dirichlet H

Generative process:

1. Draw global topic distribution:

$$G_0 \sim \mathrm{DP}(\gamma, H)$$

2. For each document j:

$$G_j \sim \mathrm{DP}(\alpha, G_0)$$

3. For each word x_{ji}:
    - Choose a table t_{ji} in restaurant j
    - The table is assigned topic k_{jt}
    - Draw word: $x_{ji} \sim f(\cdot \mid \phi_{k_{jt}})$

Inference via Collapsed Gibbs Sampling

To estimate the document-topic structure, we perform **collapsed Gibbs sampling**, which integrates out the random measures G_0 and G_j, and samples the table and topic assignments directly.

Step 1: Initialize

- Assign each word in each document to a random table.
- Assign each table to a random topic.
- Maintain count statistics:
  - n_{jt}: words at table t in document j
  - m_{jk}: number of tables in document j serving topic k
  - $m_{\cdot k}$: total number of tables across all documents serving topic k
  - Word-topic counts for each topic k

Step 2: Sample Table Assignments (t_{ji})

For each word x_{ji}:

1. Remove the current assignment and decrement counts.
2. Compute the probability of joining an existing table t:

$$P(t_{ji} = t) \propto n_{jt}^{-ji} \cdot f(x_{ji} \mid \phi_{k_{jt}})$$

3. Compute the probability of starting a new table:

$$P(t_{ji} = \text{new}) \propto \alpha \cdot \left(\sum_k \frac{m_{\cdot k}}{m_{\cdot\cdot} + \gamma} f(x_{ji} \mid \phi_k) + \frac{\gamma}{m_{\cdot\cdot} + \gamma} \int f(x_{ji} \mid \phi) \, dH(\phi) \right)$$

4. If a new table is created, sample a topic for it (see next step).

Step 3: Sample Topic Assignments for Tables (k_{jt})

For each table t in document j:

1. Collect all words at that table: $\{x_{ji} : t_{ji} = t\}$
2. Compute the posterior probability for assigning the table to topic k:

$$P(k_{jt} = k) \propto m_{\cdot k}^{-jt} \cdot \int \left(\prod_i f(x_{ji} \mid \phi_k) \right) dH(\phi_k)$$

3. Compute probability of assigning the table to a **new** topic:

$$P(k_{jt} = \text{new}) \propto \gamma \cdot \int \left(\prod_i f(x_{ji} \mid \phi) \right) dH(\phi)$$

4. If a new topic is chosen, sample new parameters ϕ_k for it.

Step 4: Sample Topic Parameters (ϕ_k)

For each topic k, collect all words currently assigned to it and sample the topic distribution ϕ_k from the posterior:

$$\phi_k \sim P(\phi \mid \{x_{ji} : k_{jt_{ji}} = k\}, H)$$

If H is Dirichlet and f is multinomial, this posterior is also Dirichlet and easily sampled.

Estimating Document Topic Proportions

After running Gibbs sampling:

- For each document j, the topic proportion θ_j is estimated based on how its words are assigned to topics.

There are two common ways to estimate θ_j:

1. **Table-weighted estimation**:

$$\theta_{jk} = \frac{\sum_t \delta(k_{jt} = k) \cdot n_{jt}}{\sum_t n_{jt}}$$

   where n_{jt} is the number of words at table t and k_{jt} is its assigned topic.

2. **Direct word-topic estimation**:

$$\theta_{jk} = \frac{n_{jk}}{\sum_k n_{jk}}$$

   where n_{jk} is the number of words in document j assigned to topic k.

Section 3 - Sticky HDP-HMM

3.1 - HDP-HMM

The **HDP-HMM (Hierarchical Dirichlet Process Hidden Markov Model)** is a Bayesian nonparametric extension of the standard HMM that allows for an **infinite number of states**.

- Each state has a **state-specific transition distribution** drawn from a Dirichlet Process (DP).
- These transition distributions share a **global base distribution**, encouraging similar transition patterns across states.
- Enables automatic inference of the number of hidden states from data.

Let:

- $\beta \sim \mathrm{GEM}(\gamma)$: global transition weights
- $\pi_j \sim \mathrm{DP}(\alpha, \beta)$: transition distribution for state j
- $z_t \sim \pi_{z_{t-1}}$: latent state sequence
- $y_t \sim F(\theta_{z_t})$: observation emitted from state

However, HDP-HMM lacks an explicit **self-transition bias**, often causing rapid, unrealistic switching between states, which leads to over-segmentation and redundancy.

3.2 - Sticky HDP-HMM

The **Sticky HDP-HMM** improves upon the HDP-HMM by introducing a **self-transition bias**, promoting **temporal persistence** of states.

- Adds a "stickiness" parameter $\kappa > 0$ to boost the prior probability of staying in the same state.
- Reduces redundant states and encourages smoother transitions.

Modified transition prior:

$$\pi_j \sim \mathrm{DP}(\alpha + \kappa, \frac{\alpha\beta + \kappa\delta_j}{\alpha + \kappa})$$

Key advantages:

- Encourages **state persistence**, especially important in time-series tasks like speaker diarization.
- Enables use of **nonparametric emission models** (e.g., Dirichlet Process mixtures).
- Leads to **better mixing** in sampling algorithms and **more accurate** segmentation.

3.3 - HDP-HMM-Multinormal-Emission

The **HDP-HMM-Multinormal-Emission** is a special case of the HDP-HMM where each hidden state emits observations from a **Multivariate Normal (Gaussian) distribution**. This is suitable for modeling i.i.d. multivariate data conditional on the state, with no dependence on past observations (unlike VAR emissions).

Model Specification:

Let $\mathbf{y}_t \in \mathbb{R}^D$ denote the D-dimensional observation at time t. When the latent state is $z_t = k$, the emission is:

$$\mathbf{y}_t \mid z_t = k \sim \mathcal{N}(\mu_k, \Sigma_k)$$

- $\mu_k \in \mathbb{R}^D$: mean vector of state k
- $\Sigma_k \in \mathbb{R}^{D \times D}$: covariance matrix of state k

Full HDP-HMM-Multinormal Specification:

As before, let:

- $\beta \sim \mathrm{GEM}(\gamma)$
- $\pi_j \sim \mathrm{DP}(\alpha + \kappa, \frac{\alpha\beta + \kappa\delta_j}{\alpha + \kappa})$
- $z_t \mid z_{t-1} \sim \pi_{z_{t-1}}$
- $\mathbf{y}_t \mid z_t = k \sim \mathcal{N}(\mu_k, \Sigma_k)$

Let $\Theta = \{(\mu_k, \Sigma_k)\}_{k=1}^{\infty}$ be the set of Gaussian emission parameters. The full joint distribution is:

$$p(\mathbf{y}_{1:T}, z_{1:T}, \{\pi_j\}, \beta, \Theta) = p(\beta) \prod_j p(\pi_j \mid \beta) \prod_k p(\mu_k, \Sigma_k) \\ \times p(z_1) \prod_{t=2}^{T} p(z_t \mid \pi_{z_{t-1}}) \prod_{t=1}^{T} p(\mathbf{y}_t \mid z_t, \mu_{z_t}, \Sigma_{z_t})$$

Emission Prior:

A common conjugate prior for (μ_k, Σ_k) is the **Normal-Inverse-Wishart** (NIW) prior:

$$\Sigma_k \sim \mathcal{IW}(\nu_0, S_0), \quad \mu_k \mid \Sigma_k \sim \mathcal{N}(m_0, \Sigma_k / \kappa_0)$$

This prior allows for efficient marginalization and closed-form updates during inference.

3.4 - HDP-HMM-VAR

The **HDP-HMM-VAR** is an extension of the HDP-HMM where each hidden state governs a **Vector Autoregressive (VAR)** process, allowing the emission distribution to depend on **multiple past observations**. This models complex temporal dependencies in multivariate time series.

Model Specification:

Let $\mathbf{y}_t \in \mathbb{R}^D$ be the observed D-dimensional vector at time t. Given a VAR process of order r, the observation model in state $z_t = k$ is:

$$\mathbf{y}_t = \sum_{i=1}^{r} A_{k,i} \mathbf{y}_{t-i} + \mathbf{e}_t, \quad \mathbf{e}_t \sim \mathcal{N}(0, \Sigma_k)$$

- $A_{k,i} \in \mathbb{R}^{D \times D}$: autoregressive coefficient matrix for lag i in state k
- $\Sigma_k \in \mathbb{R}^{D \times D}$: state-specific noise covariance matrix
- r: VAR order (fixed)

Define $\bar{\mathbf{y}}_t = \begin{bmatrix} \mathbf{y}_{t-1}^\top & \cdots & \mathbf{y}_{t-r}^\top \end{bmatrix}^\top \in \mathbb{R}^{Dr}$ and stack the coefficient matrices:

$$A_k = [A_{k,1} \quad \cdots \quad A_{k,r}] \in \mathbb{R}^{D \times Dr}$$

Then the emission model becomes:

$$\mathbf{y}_t \mid z_t = k, \bar{\mathbf{y}}_t \sim \mathcal{N}(A_k \bar{\mathbf{y}}_t, \Sigma_k)$$

Full HDP-HMM-VAR Specification:

Let $\beta \sim \mathrm{GEM}(\gamma)$ and define:

- Transition distributions:

$$\pi_j \sim \mathrm{DP}(\alpha + \kappa, \tfrac{\alpha\beta + \kappa\delta_j}{\alpha + \kappa})$$

- Latent states:

$$z_t \mid z_{t-1} \sim \pi_{z_{t-1}}$$

- Emissions (VAR(r)):

$$\mathbf{y}_t \mid z_t = k, \bar{\mathbf{y}}_t \sim \mathcal{N}(A_k \bar{\mathbf{y}}_t, \Sigma_k)$$

Let $\Theta = \{A_k, \Sigma_k\}_{k=1}^{\infty}$ denote the set of state-specific VAR parameters.

The joint distribution over observations $\{\mathbf{y}_t\}_{t=1}^T$, latent states $\{z_t\}_{t=1}^T$, and parameters is:

$$\begin{aligned} p(\mathbf{y}_{1:T}, z_{1:T}, \{\pi_j\}, \beta, \Theta) = p(\beta) \prod_j p(\pi_j \mid \beta) \prod_k p(A_k, \Sigma_k) \\ \times p(z_1) \prod_{t=2}^T p(z_t \mid \pi_{z_{t-1}}) \prod_{t=r+1}^T p(\mathbf{y}_t \mid z_t, \bar{\mathbf{y}}_t, A_{z_t}, \Sigma_{z_t}) \end{aligned}$$

Where:

- $p(\beta) = \mathrm{GEM}(\gamma)$
- $p(\pi_j \mid \beta) = \mathrm{DP}(\alpha + \kappa, \frac{\alpha\beta + \kappa\delta_j}{\alpha + \kappa})$
- $p(A_k, \Sigma_k)$ typically uses a conjugate Matrix-Normal Inverse-Wishart prior:

$$A_k \mid \Sigma_k \sim \mathcal{MN}(M_0, \Sigma_k, K_0^{-1}), \quad \Sigma_k \sim \mathcal{IW}(\nu_0, S_0)$$

3.5 - Chinese Restaurant Franchise with Loyal Customer

The **Chinese Restaurant Franchise with Loyal Customer (CRF-LC)** is an intuitive metaphor for the Sticky HDP-HMM.

- Each **restaurant** corresponds to a previous state z_{t-1}.

- Each **customer** is a data point y_t, who chooses a table t and is served a dish (next state z_t).
- Every restaurant has a **specialty dish** (corresponding to self-transition), favored by its "loyal" customers.

Sampling process:

1. Customer y_t enters restaurant $j = z_{t-1}$ — that is, the customer is assigned to the restaurant corresponding to the previous state in the Markov chain.
2. The customer chooses a **table** $t_{ji} \sim \tilde{\pi}_j$:
    - Each table in the restaurant has a number of customers already seated.
    - The probability of choosing a table is **proportional to the number of customers already at that table** (i.e., popularity).
    - Alternatively, the customer may start a **new table** with probability proportional to the concentration parameter $\alpha + \kappa$.
3. If a **new table** is chosen, it selects a **considered dish** (i.e., candidate next state) $\bar{k}_{jt} \sim \beta$:
    - β reflects the **global popularity of dishes** across all restaurants (i.e., how frequently each state has been chosen).
    - The probability of choosing dish k is **proportional to the number of tables across all restaurants that already serve dish** k.
    - Or, a new dish can be chosen with probability proportional to γ.
4. With probability $\rho = \kappa/(\alpha + \kappa)$, the considered dish is **overridden**:
    - The restaurant enforces its **house specialty dish**, which corresponds to a **self-transition** (dish j).
    - Otherwise, the considered dish $\bar{k}_{jt}$ is served as-is.

This process increases the chance of self-transitions by biasing new tables toward reusing the restaurant's own dish.

Variables introduced:

- $w_{jt} \sim \text{Bernoulli}(\rho)$: **override indicator** — equals 1 if the dish is overridden to the specialty, 0 otherwise.
- k_{jt}: **actual dish served** at table t — either the considered dish or the overridden dish j.

- $z_t = k_{jtji}$: **next state assignment** — the customer's final dish determines the next hidden state.

This metaphor models **state persistence** as family loyalty to a restaurant's specialty, reinforcing self-transitions over time.

3.5 - Direct Assignment Rao–Blackwellized Gibbs Sampler for Sticky HDP-HMM

Notation:

- j: id of restraunt
- i: id of customer
- t: id of table
- β: global mixture weights
- t_{ji}: table assignment for each customer
- k_{jt}: served dish assignment
- w_{jt}: dish override variables
- $m_{j,k}$: number of tables in j's restraunt serving dish k
- $\bar{m}_{j,k}$: number of tables in j's restraunt considering dish k

Input: Previous augmented state assignments $(z_{1:T}^{(n-1)}, s_{1:T}^{(n-1)})$ and global transition distribution $\beta^{(n-1)}$

1. **Initialize:** Set $(z_{1:T}, s_{1:T}) = (z_{1:T}^{(n-1)}, s_{1:T}^{(n-1)}), \beta = \beta^{(n-1)}$
2. **For each time step** $t = 1, \ldots, T$:

   (a) **Remove previous assignment** $(z_t, s_t) = (k, j)$:

   - Decrement counts:

$$n_{z_{t-1}z_t} - -, \quad n_{z_t z_{t+1}} - -$$

   - Remove y_t from sufficient statistics:

$$(\hat{\mu}_{k,j}, \hat{\Sigma}_{k,j}) \leftarrow (\hat{\mu}_{k,j}, \hat{\Sigma}_{k,j}) \ominus y_t, \quad \hat{\nu}_{k,j} \leftarrow \hat{\nu}_{k,j} - 1$$

   (b) **For each current HDP-HMM state** k, calculate the probability of the t-th state is k after observing all other states:

$$p(z_t = k|z_{/t}, \beta, \alpha, \kappa) \propto \begin{cases} (\alpha\beta_k + n^{-t}_{z_{t-1}k} + \kappa\delta(z_{t-1}, k)) \\ (\frac{\alpha\beta_{z_{t+1}} + n^{-t}_{kz_{t+1}} + \kappa\delta(k, z_{t+1}) + \delta(z_{t-1}, k)\delta(k, z_{t+1})}{\alpha + n^{-t}_{k\cdot} + \kappa + \delta(z_{t-1}, k)}), & k \in \{1, \dots, K\} \\ \frac{\alpha^2\beta_{\tilde{k}}\beta_{z_{t+1}}}{\alpha+\kappa} & k = K+1 \end{cases}$$

where

- n^{-t}_{jk} denotes the number of transitions from state j to k not counting the transition from z_{t-1} to z_t or from z_t to z_{t+1}.
- β_k: global mixture weights of k for $k < K$.
- $\beta_{\tilde{k}}$: global mixture weights of $K+1, K+2, \dots +\infty$
- $\delta(a, b)$ denote an indicator function where it will be 1 if $a = b$, else 0.

$$f_k(y_t) = p(z_t = k|z_{/t}, \beta, \alpha, \kappa) \cdot postior(observations|z_t = k)$$

(c) **Sample new state assignment** z_t:

$$z_t \sim \sum_{k=1}^{K} f_k(y_t) \cdot \delta(z_t = k) + f_{K+1}(y_t) \cdot \delta(z_t = K+1)$$

(d) **Update statistics:**

- If a new state is chosen, increment K to $K+1$, and split β: Sample $b \sim \text{Beta}(1, \gamma)$, then

  $$\beta_K \leftarrow b \cdot \beta_{\tilde{k}}, \quad \beta_{\tilde{k}} \leftarrow (1-b) \cdot \beta_{\tilde{k}}$$

  which is, to split β for the last state into 2 separate parts.
- Update counts and cached stats:

  $$n_{z_{t-1}z_t} ++, \quad n_{z_t z_{t+1}} ++$$

  $$(\hat{\mu}_{k,j}, \hat{\Sigma}_{k,j}) \leftarrow (\hat{\mu}_{k,j}, \hat{\Sigma}_{k,j}) \oplus y_t, \quad \hat{\nu}_{k,j} \leftarrow \hat{\nu}_{k,j} + 1$$

3. **Fix:** $(z^{(n)}_{1:T}, s^{(n)}_{1:T}) = (z_{1:T}, s_{1:T})$. if there exists a j such that $nj\cdot$ and $n\cdot j$ are both 0 at the same time, remove j and decrement K.
4. **Sample auxiliary variables** m, w, **and** $\bar{m}$

   (a) For each $(j, k) \in \{1, 2, 3, \dots, K\}^2$, set $m_{j,k} = 0$ and $n = 0$. For each customer in restraunt j eating dish k, i.e. for $i = 1, \dots, n_{jk}$, sample

   $$x \sim Ber(\frac{\alpha\beta_k + \kappa\delta(j,k)}{n + \alpha\beta_k + \kappa\delta(j,k)})$$

which is a sampling of whether to incease the number of tables in restraunt j choosing dish k. Increase $m_{j,k}$ by 1 if $x = 1$, otherwise do nothing. Now increase n by 1 and repeat until we go through everyone in this restraunt.

(b) For each $j \in \{1, \ldots, K\}$, sample the number of override variables in restraunt j:

$$w_{j\cdot} \sim Binomial(m_{jj}, \frac{\rho}{\rho + \beta_j(1-\rho)})$$

(c) Set the number of informative tables in restraunt j considering dish k to:

$$\bar{m}_{jk} = \begin{cases} m_{jk}, & j \neq k \\ \frac{\alpha^2 \beta_{\tilde{k}} \beta_{z_{t+1}}}{\alpha + \kappa} & j = k \end{cases}$$

5. **Sample global transition distribution** $\beta^{(n)}$

$$\beta^{(n)} \sim Dir(\bar{m}_{\cdot 1}, \bar{m}_{\cdot 2}, \ldots, \bar{m}_{\cdot k}, \gamma)$$

which will give us a sample for probabilities coresponding to state $k = 1, 2, 3, \ldots, K$, and $\tilde{k}$.