Learning Robot Manipulation Tasks with Semi-Tied Gaussian Mixture Models

Ajay Kumar Tanwani

Swiss Machine Learning Day 2015
Application – Skill Acquisition in Teleoperated Robots
Semi-Autonomous Manipulation

Recognition of intentions on teleoperator side

Reproduction of movement on robot side

Subspace Clustering  Task Adaptability  Autonomous Control
Outline

- Semi-tied Gaussian mixture models
- Task-parameterized semi-tied GMMs
- Hidden semi-Markov model encoding
- Linear quadratic tracking control
- Valve opening with Baxter robot
Subspace Clustering

\[ \mathcal{P}(\xi_t | \theta) = \sum_{i=1}^{K} \pi_i \mathcal{N}(\xi_t | \mu_i, \Sigma_i) \quad \theta = \{\pi_i, \mu_i, \Sigma_i\}_{i=1}^{K} \]

- Model over-fitting with \( D \gg T \)
- Need for parsimonious model with fewer parameters and better generalization
- Statistical subspace clustering imposes special structure on the covariance matrix to model the latent space of dimension \( d \) with \( d \ll D \)
  - Isotropic, diagonal, block-diagonal, multiple diagonal, full
Subspace Clustering

Motion segmentation and tracking

3D human motion tracking

[Elhamifar and Vidal, 2013] [Li et al., 2009]
Subspace Clustering

- Mixture of factor analyzers
  \[ \Sigma_i = \Lambda_i \Lambda_i^T + \Psi_i \]

- Probabilistic principal component analysis
  \[ \Sigma_i = \Lambda_i \Lambda_i^T + \sigma^2 I_D \]

\[ P(\xi_t|\theta) = \sum_{i=1}^{K} \pi_i \mathcal{N}(\xi_t|\mu_i, \Sigma_i) \]

- Human movements are spatially and temporally correlated along important synergistic directions
- Need for sharing the parameters across the mixture components
Semi-Tied Gaussian Mixture Models

\[ \Sigma_i = H \Sigma_i^{(\text{diag})} H^\top \]

- \( H \) applies global linear transformation to de-correlate the data, and \( \Sigma_i^{(\text{diag})} \) selects the appropriate subspace.
- Mixture components are aligned along the basis vectors for noisy and/or insufficient training data.

\[ \Sigma_i := \alpha H \Sigma_i^{(\text{diag})} H^\top + (1 - \alpha) S_i \quad \alpha \in (0, 1) \]

\[ P(\xi_t | \theta) = \sum_{i=1}^{K} \pi_i \mathcal{N}(\xi_t | \mu_i, \Sigma_i) \]

- \( H \in \mathbb{R}^{D \times D} \Rightarrow \) common latent basis vectors
- \( \Sigma_i^{(\text{diag})} \in \mathbb{R}^{D \times D} \Rightarrow \) component-specific diagonal matrix
- \( S_i \in \mathbb{R}^{D \times D} \Rightarrow \) empirical covariance matrix

[Gales, 1999]
Semi-Tied Gaussian Mixture Models

- **E-Step:**
  \[
  h_{t,i}^{\hat{\theta}} := \frac{\pi_i \mathcal{N}(\xi_t | \mu_i, \Sigma_i)}{\sum_{k=1}^{K} \pi_k \mathcal{N}(\xi_t | \mu_k, \Sigma_k)}
  \]

- **M-Step:**
  \[
  \pi_i := \frac{\sum_{t=1}^{T} h_{t,i}}{T}
  \]
  \[
  \mu_i := \frac{\sum_{t=1}^{T} h_{t,i} \xi_t}{\sum_{t=1}^{T} h_{t,i}}
  \]
  \[
  S_i := \frac{\sum_{t=1}^{T} h_{t,i} (\xi_t - \mu_i)(\xi_t - \mu_i)^T}{\sum_{t=1}^{T} h_{t,i}}
  \]

\[
\theta = \{ \pi_i, \mu_i, B, \Sigma_i^{(\text{diag})} \}_{i=1}^{K}
\]
\[
B = H^{-1}
\]
\[
\Sigma_i^{(\text{diag})} := \text{diag}(BS_iB^T)
\]
\[
C := B^{-1} |B|
\]

Variational optimisation of \(\Sigma_i^{(\text{diag})}\) and \(B\)

\[
G_d := \sum_{i=1}^{K} \frac{1}{\sum_{i,d}^{(\text{diag})}} S_i \sum_{t=1}^{T} h_{t,i}^{\hat{\theta}}
\]
\[
b_d := c_d G_d^{-1} \sqrt{\sum_{t=1}^{T} \sum_{i=1}^{K} h_{t,i}^{\hat{\theta}}} c_d G_d^{-1} c_d^T
\]
\[
\Sigma_i := \alpha H \Sigma_i^{(\text{diag})} H^T + (1 - \alpha) S_i
\]
Chicken Dance Encoding

- Regenerated movement sequence is shown in green

\[ D = 94, \quad K = 75 \]
Chicken Dance Encoding

- Regenerated movement sequence is shown in green

\[ D = 94, \ K = 75 \]
Chicken Dance Encoding

Semi-Tied GMM components are more correlated than standard GMM components

\[ D = 94, \ K = 75 \]
Chicken Dance Encoding

- Semi-Tied model requires more components but the number of parameters remain less
Outline

- Semi-tied Gaussian mixture models
- **Task-parameterized semi-tied GMMs**
- Hidden semi-Markov model encoding
- Linear quadratic tracking control
- Valve opening with Baxter robot
Task-Parameterized Semi-Tied GMM

- Adopt the model parameters to new environmental situations using frames of reference

- Observe the data from $P$ coordinate systems $\{A_j, b_j\}_{j=1}^P : \{\xi_t^{(j)}\}_{j=1}^P$

\[ \xi_t^{(j)} = A_j^{-1}(\xi_t - b_j) \]

\[ h_{t,i}^{(p)} := \frac{\pi_i \mathcal{N}(\mu_i^{(p)}, \Sigma_i^{(p)})}{\sum_{k=1}^K \pi_k \mathcal{N}(\mu_k^{(p)}, \Sigma_k^{(p)})} \]

\[ \mathcal{N}(\mu_i^{(p)}, \Sigma_i^{(p)}) = \prod_{j=1}^P \mathcal{N}(\xi_t^{(j)} \mid \mu_i^{(j)}, \Sigma_i^{(j)}) \]
Task-Parameterized Semi-Tied GMM

\[ \xi_t^{(j)} = A_j^{-1}(\xi_t - b_j) \]
\[ \theta_p = \{ \pi_i, \{ \mu_i^{(j)}, \Sigma_i^{(j)} \}_{j=1}^P \}_{i=1}^K \]

\[ \Sigma_i^{(j)} = \alpha H^{(j)} \Sigma_i^{(j)(\text{diag})} H^{(j)\top} + (1 - \alpha) S_i^{(j)} \]
Task-Parameterized Semi-Tied GMM $\mathcal{N}(\tilde{\mu}_i, \tilde{\Sigma}_i) \propto \prod_{j=1}^{P} \mathcal{N}(\tilde{A}_j \mu_i^{(j)} + \tilde{b}_j, \tilde{A}_j \Sigma_i^{(j)} \tilde{A}_j^\top)$

- Given the new environmental situation $\{\tilde{A}_j, \tilde{b}_j\}_{j=1}^{P}$, the model parameters are adapted by taking \textit{product of linearly transformed Gaussians}

\[
\tilde{\mu}_i = \tilde{\Sigma}_i \sum_{j=1}^{P} \left( \tilde{A}_j \Sigma_i^{(j)} \tilde{A}_j^\top \right)^{-1} \left( \tilde{A}_j \mu_i^{(j)} + \tilde{b}_j \right) \\
\tilde{\Sigma}_i = \left( \sum_{j=1}^{P} \left( \tilde{A}_j \Sigma_i^{(j)} \tilde{A}_j^\top \right)^{-1} \right)^{-1}
\]
Outline

- Semi-tied Gaussian mixture models
- Task-parameterized semi-tied GMMs
- **Hidden semi-Markov model encoding**
- Linear quadratic tracking control
- Valve opening with Baxter robot
Hidden Semi-Markov Model Encoding

- Recognize the current state of the task and re-plan the movement sequence
- Encapsulate the spatio-temporal information in the model
Hidden Semi-Markov Model Encoding

- Each state output is a single Gaussian representing product of Gaussians

- Self-transition probability is explicitly modeled for state duration by a Gaussian

\[
\theta_h = \left\{ \prod_i, \{a_{i,m}\}_{m=1}^K, \{\mu_i^{(j)}, \Sigma_i^{(j)}\}_{j=1}^P, \mu_i^D, \Sigma_i^D \right\}_{i=1}^K
\]
Hidden Semi-Markov Model Encoding

- Generation of state sequence with datapoint $\xi_t$ to be in state $i$ at time $t$ is computed with forward variable

$$\alpha_{t,i}^{\text{HSMM}} = \sum_{j=1}^{K} \sum_{d=1}^{\min(d_{\text{max}},t-1)} \alpha_{t-d,j}^{\text{HSMM}} a_{j,i} \mathcal{N}(d | \mu_i^D, \Sigma_i^D)$$

- Desired step-wise reference trajectory $\mathcal{N}(\hat{\mu}_t, \hat{\Sigma}_t)$ follows from the forward variable

$$q_t = \arg \max_i \alpha_{t,i}^{\text{HSMM}}, \quad \hat{\mu}_t = \hat{\mu}_{q_t}, \quad \hat{\Sigma}_t = \hat{\Sigma}_{q_t}$$

$$\alpha_{1,i}^{\text{HSMM}} = \frac{\pi_i \mathcal{N}(\xi_1 | \hat{\mu}_i, \hat{\Sigma}_i)}{\sum_{k=1}^{K} \pi_k \mathcal{N}(\xi_1 | \hat{\mu}_k, \hat{\Sigma}_k)}$$
Outline

▪ Semi-tied Gaussian mixture models
▪ Task-parameterized semi-tied GMMs
▪ Hidden semi-Markov model encoding
▪ **Linear quadratic tracking control**
▪ Valve opening with Baxter robot
Linear Quadratic Tracking Control

- Desired step-wise reference trajectory $N(\hat{\mu}_t, \hat{\Sigma}_t)$ is smoothly tracked by minimizing the cost function starting from initial state $\xi_1$

$$c_t(\xi_t, u_t) = \sum_{i=1}^{T}(\xi_t - \hat{\mu}_t)\!
\cdot\! Q_t(\xi_t - \hat{\mu}_t) + u_t\!
\cdot\! R_t u_t \quad Q_t = \hat{\Sigma}_t^{-1} \succeq 0, R_t \succ 0$$

s.t. $\dot{\xi}_t = A_d \xi_t + B_d u_t$

- Optimal control input is obtained by solving a set of differential equations

$$u_t^* = K_t^x (\hat{x}_t - x_t) + K_t^\dot{x} (\hat{\dot{x}}_t - \dot{x}_t) - R_t^{-1} B_d^\top d_t$$

$$\dot{\xi}_t = [x_t^\top \dot{x}_t^\top]^\top$$

$$\dot{\hat{\mu}}_t = [\hat{x}_t^\top \hat{\dot{x}}_t^\top]^\top$$
Task variability is used for adjusting the compliance in following the trajectory
Task variability is used for adjusting the compliance in following the trajectory.
Outline

- Semi-tied Gaussian mixture models
- Task-parameterized semi-tied GMMs
- Hidden semi-Markov model encoding
- Linear quadratic tracking control
- Valve opening with Baxter robot
Valve Opening Experiment with Baxter

- Two frames: \( \{A_1, b_1\} \) for initial configuration of the valve, \( \{A_2, b_2\} \) for desired end configuration of the valve

- Eight demonstrations \( n = 1 \ldots 8 \) downsampled to 200 datapoints, 50-50 training testing ratio, and \( D = 14 \)

\[
A_j^{(n)} = \begin{bmatrix}
R_j^{(n)} & \mathcal{E}_j^{(n)} & 0 \\
0 & R_j^{(n)} & \mathcal{E}_j^{(n)}
\end{bmatrix}, \quad b_j^{(n)} = \begin{bmatrix}
p_j^{(n)} \\
0 \\
0 \\
0
\end{bmatrix}
\]

- \( x_t^p \in \mathbb{R}^3 \) \( \Rightarrow \) Cartesian position
- \( \mathcal{E}_t^q \in \mathbb{R}^4 \) \( \Rightarrow \) Quaternion orientation
- \( \dot{x}_t^p \in \mathbb{R}^3 \) \( \Rightarrow \) Linear velocity
- \( \dot{\mathcal{E}}_t^q \in \mathbb{R}^4 \) \( \Rightarrow \) Quaternion derivative
Valve Opening Experiment with Baxter

- Task parameterized semi-tied mixture components are better aligned and scaled
Valve Opening Experiment with Baxter

- Task parameterized semi-tied mixture components are better aligned and scaled
Valve Opening Experiment with Baxter

<table>
<thead>
<tr>
<th>α</th>
<th>Training MSE</th>
<th>Testing MSE</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>0.0021</td>
<td>0.0146</td>
<td>1470</td>
</tr>
<tr>
<td>TP-HSMM</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.0038</td>
<td>0.0119</td>
<td>-</td>
</tr>
<tr>
<td>1.0</td>
<td>0.0040</td>
<td><strong>0.0119</strong></td>
<td>588</td>
</tr>
<tr>
<td>TP-Semi-Tied HSMM</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Semi-tied model gives better testing accuracy than standard GMM with much less parameters
Valve Opening Experiment with Baxter

- The model exploits variability in the demonstrations to extract invariant patterns
Conclusion

- Semi-tied GMMs encode similar coordination patterns with a set of basis vectors /synergistic directions
- Proposed framework combines parsimonious movement representation, task adaptability and optimal control for learning manipulation tasks
- Task-parameterized semi-tied HSMM enables the robot to autonomously deal with different manipulation scenarios in a task