Extensions of Markov models and HMMs:
acyclic graphs, uncountable feature and state spaces

Markov models and HMMs discussed so far:
- graph: chain or "comb-like" (HMM)
- state / feature space: finite

A. Hidden Markov models on acyclic graphs

Let $T = (V,E)$ be an undirected, connected acyclic graph. When fixing an arbitrary mode $i \in V$, we denote by $\overrightarrow{E_i}$ the edge set of the corresponding rooted digraph.

**Definition 1a** Let $T = (V,E)$ be an undirected tree and $s_i, i \in V$ be $K$-valued random variables. A p.d. for the random field $S \in K^V$ is a Markov model on $T$ if

$$p(s) = p(s_r) \prod_{ij \in \overrightarrow{E_r}} p(s_j | s_i)$$

holds for any choice $r \in V$ (root).

**Definition 1b** A p.d. $p(s)$ for $S \in K^V$ is a Markov model on $T$ if $p$ can be written as

$$p(s) = \prod_{ij \in E} G_{ij}(s_i, s_j)$$

with some functions $G_{ij} : K^2 \rightarrow \mathbb{R}_+$. In particular, $p(s)$ can be written as

$$p(s) = \prod_{ij \in \overrightarrow{E}} p(s_i, s_j) / \prod_{i \in V} p^{n_i}(s_i)$$

where $n_i$ denotes the degree of the node $i \in V$. 
An HMM on an undirected tree $T = (V,E)$ is a $p.d.$ for pairs $s \in \mathbb{K}^V$, $x \in \mathbb{F}^V$ ($\mathbb{F}$ is a feature space) s.t.

- $p(s)$ is a Markov model on $T$
- $p(x|s) = \prod_{i \in V} p(x_i|s_i)$

Let us consider inference tasks for HMMs on trees. Given an observation $\mathbf{x}$, $\mathbf{x} \in \mathbb{F}^V$ compute

$$p(x) = \sum_{s \in \mathbb{K}^V} p(x,s)$$

Substituting the model, we get

$$p(x) = \sum_{s \in \mathbb{K}^V} p(s_r) p(x_r|s_r) \prod_{i \in V} p(s_i|s_{i-}) p(x_i|s_i)$$

which has the form (fixed observation)

$$\sum_{s \in \mathbb{K}^V} \prod_{i \in V} \psi_i(s_i) \prod_{i \in \overline{E}} \phi_{ij}(s_i,s_j)$$

with

$$\phi_{ij}(s_i,s_j) = p(s_j|s_i) p(x_j|x_i) \quad \text{and} \quad \psi_i(s_i) = \left\{ \begin{array}{ll}
p(s_i) p(x_r|x_i) & \text{if } i = r \\
1 & \text{otherwise}
\end{array} \right.$$
Remark 1. The same approach is used for solving the task

\[ S^*_x \in \arg \max_{S \in \mathcal{K}} \log p(x, S) \]

Simply by replacing operations \( x \mapsto + \), \( + \mapsto \max \).

Computing marginal probabilities: Given an observation \( x \in F \), compute \( p(x, S_i) \) for \( i \in V, S_i \in \mathcal{K} \). Recall the algorithm for computing marginals of an HMM on a chain (Sec. 5).

Here: from Def. 16 follows that

\[ p(x, S_i) = p(S_i) p(x, S_i) \prod_{j \in \text{path}(i, m)} p(x_{T_{ij}}, S_i) \]

where \( T_{ij} \) denotes the subtree given by

\[ V(T_{ij}) = \{ m \in V / j \in \text{path}(i, m) \} \]

Let us denote \( \psi_{ij}(S_i) := p(x_{T_{ij}}, S_i) \). They fulfill the following system of equations

\[ \psi_{ij}(S_i) = \sum_{S_j \in \mathcal{K}} p(S_j | S_i) p(x_j | S_j) \prod_{l \in \text{path}(j, l)} \psi_{lj}(S_j) \]

Two passes through all edges of \( T \) suffice to compute all of them and, consequently all marginals. \( \Rightarrow \) Complexity \( 2 | \mathcal{K} |^2 |E| \)

B. Uncountable feature space

All inference & learning algorithms discussed so far can be applied also if the feature space \( F \) is uncountable infinite, provided that
• the conditional distribution (densities) \( p(x_i | s_i) = p_{\theta_i}(x_i | s_i) \) are given in some parametric model (e.g. normal distribution)
• their parameters can be estimated from corresponding samples

C. Uncountable state spaces

**Special case:** \( s_i \in \mathbb{R}^n, \ x_i \in \mathbb{R}^m \)

\[
\begin{align*}
s_i &\sim \mathcal{N}(\mu_i, \Sigma_i) \\
S_i | S_{i-1} &\sim \mathcal{N}(A S_{i-1}, \Sigma') \\
x_i | S_i &\sim \mathcal{N}(H S_i, \Gamma)
\end{align*}
\]

where \( A : \mathbb{R}^n \to \mathbb{R}^n \) and \( H : \mathbb{R}^m \to \mathbb{R}^m \) are linear mappings, \( \Sigma, \Sigma' \) and \( \Gamma \) are covariance matrices and \( \mathcal{N}(\mu, \Sigma) \) denotes a multivariate normal distribution.

**Observation:**
• product of two normal p.d.f.s

\[
\mathcal{N}(\mu, A) \cdot \mathcal{N}(\gamma, B) = \mathcal{N}(\gamma, C)
\]

where \( \gamma = C(A^{-1}\mu + B^{-1}\gamma) \), \( C = (A^{-1} + B^{-1})^{-1} \)
• convolution of two normal p.d.f.s

\[
\int_{\mathbb{R}^n} \mathcal{N}(x; \mu, A) \mathcal{N}(y-x; \gamma, B) = \mathcal{N}(y; \gamma, C)
\]

where \( C = A + B \), \( \gamma = \mu + y \)

Hence, e.g. \( P(S_n | X_1:n) \) is normally distributed. Its p.d.f. can be computed recursively => Kalman filter
General case:

Typical application example: SLAM (simultaneous localisation and mapping)

Approach for computing e.g. $p(s_n | x_{1:n})$: particle filters & sequential Monte Carlo sampling

1. Generate an i.i.d. sample $s^e_1, e = 1, \ldots, L$ using
   $$p(s_1 | x_1) \sim p(s_1) p(x_1 | s_1)$$

2. Iterate: given a sample $s^e_{i-1}, e = 1, \ldots, L$ generated by $p(s_{i-1} | x_{1:i-1})$, sample $s^e_i$ as follows
   $$s^e_i \sim p(x_i | s_i) p(s_i | s_{i-1} = s^e_{i-1})$$

The finally obtained sample $s^e_{1:L}, e = 1, \ldots, L$ estimates $p(s_n | x_{1:n})$ and can be used to estimate the posterior expectation of a random variable $f(s_n)$

$$E(f | x_{1:n}) = \int f(s_n) p(s_n | x_{1:n}) ds_n \approx \frac{1}{L} \sum_{e=1}^{L} f(s^e_{1:L})$$