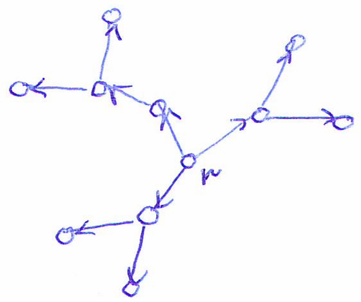
10. Extensions of Markov models and HMMs: acyclic graphs, uncountable feature and state spaces

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 node $r\in V$, we denote by $\vec{E}_r$ 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

$$p(s) = p(s_r)\prod_{ij\in\vec{E}_r} p(s_j\,|\,s_i)$$

holds for <u>any</u> 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

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

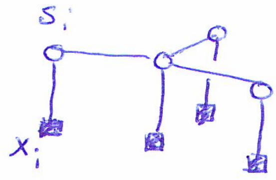
with some functions $g_{ij}\colon K^2\to\mathbb{R}_+$. In particular, $p(s)$ can be written as

$$p(s) = \prod_{\{i,j\}\in E} p(s_i,s_j) \Big/ \prod_{i\in V} p^{n_i-1}(s_i)$$

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 $s \in K^V$, $x \in F^V$ (F is a feature space) s.t.

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



Let us consider ~~in~~ inference tasks for HMMs on trees.

Given an observation field $x \in F^V$ compute

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

Substituting the model, we get

$$p(x) = \sum_{s \in K^V} p(s_r)\, p(x_r \mid s_r) \prod_{ij \in \vec{E}_r} p(s_j \mid s_i)\, p(x_j \mid s_j)$$

which has the form (fixed observation)

$$\sum_{s \in K^V} \prod_{i \in V} \varphi_i(s_i) \prod_{ij \in \vec{E}_r} f_{ij}(s_i, s_j)$$

with

$$f_{ij}(s_i, s_j) = p(s_j \mid s_i)\, p(x_j \mid s_j) \quad \text{and} \quad \varphi_i(s_i) = \begin{cases} p(s_r)\, p(x_r \mid s_r) & \text{if } i = r \\ 1 & \text{otherwise} \end{cases}$$

The algorithm recomputes the φ-s starting from an arbitrary <u>leaf</u> $j \in V$. Let $ij \in \vec{E}_r$ be its only incoming edge.

$$\varphi_i(s_i) := \varphi_i(s_i) \sum_{s_j \in K} f_{ij}(s_i, s_j)\, \varphi_j(s_j)$$

The leaf is removed thereafter. This is repeated until only the root r remains. Finally $p(x) = \sum_{s_r \in K} \varphi_r(s_r)$

<u>Complexity</u>: $|K|^2 |E|$

Remark 1 The same approach is used for solving the task

$$s_* \in \arg\max_{s \in K^V} \log p(x,s)$$

simply by replacing operations $\times \mapsto +$, $+ \mapsto \max$ ■

Computing marginal probabilities: Given an observation field $x \in F^V$, compute $p(x, s_i)$ $\forall i \in V$, $\forall s_i \in K$. Recall the algorithm for computing marginals of an HMM on a chain (Sec. 5).

Here: from Def. 16 follows that

$$p(x, s_i) = p(s_i)\, p(x_i \mid s_i) \prod_{j \in \mathcal{N}_i} p(x_{T_{ij}} \mid s_i),$$

where T_{ij} denotes the subtree given by

$$V(T_{ij}) = \{ m \in V \mid j \in \mathrm{path}(i,m) \}$$

Let us denote $\varphi_{ij}(s_i) := p(x_{T_{ij}} \mid s_i)$. They fulfil the following system of equations

$$\varphi_{ij}(s_i) = \sum_{s_j \in K} p(s_j \mid s_i)\, p(x_j \mid s_j) \prod_{\substack{\ell \in \mathcal{N}_j \\ \ell \neq i}} \varphi_{j\ell}(s_j)$$

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

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: $s_i \in \mathbb{R}^n$, $x_i \in \mathbb{R}^m$

$$s_1 \sim \mathcal{N}(\mu_1, Q)$$

$$s_i | s_{i-1} \sim \mathcal{N}(A s_{i-1}, Q')$$

$$x_i | s_i \sim \mathcal{N}(H s_i, R)$$

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

Observation:

- product of two normal p.d.f.s

$$\mathcal{N}(\mu, A) \cdot \mathcal{N}(\nu, B) = \mathcal{N}(\xi, C)$$

where $\xi = C(A^{-1}\mu + B^{-1}\nu)$, $C = (A^{-1} + B^{-1})^{-1}$

- convolution of two normal p.d.f.s

$$\int_{\mathbb{R}^n} \mathcal{N}(x; \mu, A)\, \mathcal{N}(y - x; \nu, B) = \mathcal{N}(y; \xi, C)$$

where $C = A + B$, $\xi = \mu + \nu$

Hence, e.g. $p(s_n | x_{1:n})$ is normally distributed. Its p.d.f. can be computed recursively $\Rightarrow$ 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
$\hat{=}$ sequential Monte Carlo sampling

1. Generate an i.i.d. sample s_1^ℓ, $\ell = 1,..,L$ using
$$p(s_1|x_1) \sim p(s_1)\, p(x_1|s_1)$$

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

The finally obtained sample s_n^ℓ, $\ell=1,..,L$ estimates $p(s_n|x_{1:n})$ and can be used to estimate the posterior expectation of a random variable $f(s_n)$
$$\mathbb{E}(f|x_{1:n}) = \int_{\mathbb{R}^n} f(s_n)\, p(s_n|x_{1:n})\, ds_n \approx \frac{1}{L}\sum_{\ell=1}^{L} f(s_n^\ell)$$