TS : HIDDEN MARKOV MODELS

Consider a system which may be described at any time as being in one of K distinct states $\{s_1, \dots, s_K\}$. At regularly spaced discrete times, the state of the system changes according to a set of probabilities associated with the state.

→ Denote the time index as $n = 1, 2, \dots$, and the state of the system at time n using a 1-of-K coding scheme :

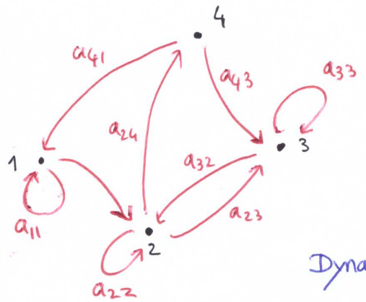
$z_n \in \{0,1\}^K$; $z_n = (z_{n1}, \dots, z_{nK})$

where $z_{nj} = \begin{cases} 1 & \text{if system is in state } s_j \\ 0 & \text{otherwise.} \end{cases}$

→ In full generality, the description of the state of the system at time n requires the knowledge of the state at times $1, \dots, n-1$. We assume that the state evolves according to a first order Markov Chain (MC), so that

$$P(z_{n i_n} = 1 \mid z_{1 i_1} = 1, \dots, z_{n-1, i_{n-1}} = 1) = \underline{P(z_{n i_n} = 1 \mid z_{n-1, i_{n-1}} = 1)}$$

We suppose also that this quantity does not depend on the time index n. This probability is known as the TRANSITION PROBABILITY, and we write



$$a_{ij} = P(z_{nj} = 1 \mid z_{n-1,i} = 1)$$

Dynamics of a MC

The a_{ij} are such that $a_{ij} \geq 0$ and $\sum_{j=1}^{K} a_{ij} = 1$

Put $\underset{(k \times k)}{A} = (a_{ij})_{\substack{1 \le i \le K \\ 1 \le j \le K}}$

→ The initial state z_1 does not have a parent state ; it has a marginal distribution $p(z_1)$ represented by a vector of probabilities $\pi = (\pi_1, \dots, \pi_K)$; where $\pi_k = P(z_{1k} = 1)$; with $\sum_{k=1}^{K} \pi_k = 1$.

(2)

The state of the system is rarely directly observable. Hidden Markov Models (HMM) extend the concept of a Markov Chain to include cases where observations are a probabilistic function of the state variable ≡ noisy measurements.

↳ Denote them x_n (at time n).

One assumes that given z_n ; observation x_n at time n is independent of all other variables in the model, so that

$$P(X_n = x_n \mid Z_1 = z_1, \dots, Z_n = z_n, X_1 = x_1, \dots, X_{n-1} = x_{n-1})$$
$$= P(X_n = x_n \mid Z_n = z_n, \theta)$$

The probabilistic relationship between the state variable (hidden) and the observation is governed by the EMISSION PROBABILITIES.

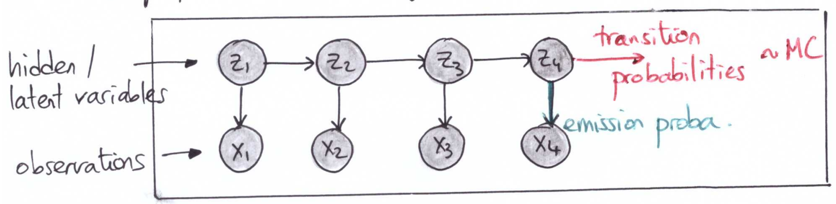
set of parameters governing the distribution

They can be represented in the form

$$P(X_n = x_n \mid Z_n = z_n, \theta) = \prod_{k=1}^{K} \{P(X_n = x_n \mid \theta_k)\}^{z_{nk}}$$

Can be discrete or continuous : work with probabilities or densities, no big deal.

The graphical structure of an HMM looks like this :



Same graphical structure as for linear dynamical systems → Kalman filtering.
Main difference : in Kalman filtering, the latent variable is continuous ; and transition + emission probabilities are gaussian.

Ex of HMMs.

(i) Binomial observations : $X_j \mid z_{jk} = 1 \sim Bi(n_j, p_k)$

number of trials may change at each time index ; while the probability of success is time independent

Using notation from page 2, $\theta = \{p_1, \dots, p_K\}$, and $\theta_k = p_k$; so that

$$P(X_j = x_j \mid z_j = z_j, \theta) = \prod_{k=1}^{K} \{Bi(x_j \mid n_j, p_k)\}^{z_{jk}}$$

(ii) Poisson observations : $X_j \mid z_{jk} = 1 \sim \mathcal{P}(\lambda_k)$

(iii) Normal observations : $X_j \mid z_{jk} = 1 \sim \mathcal{N}(\mu_k, \Sigma_k)$.

→ Applications of HMM :
- Speech Recognition
- Analysis of biological sequences (proteins, DNA)
- On-line character recognition

→ [REF].
- W. Zucchini and I.L. MacDonald. Hidden Markov Models for Time Series. An introduction using R.
- O. Cappé, E. Moulines, T. Ryden. Inference in Hidden Markov Models.

A consequence of the graphical structure of HMM is the factorization of the joint distribution over the latent and observed variables :

$$p(\underline{X}, \underline{Z} \mid \theta) = p(z_1) \left[\prod_{j=2}^{n} p(z_j \mid z_{j-1}) \right] \prod_{\ell=1}^{n} p(x_\ell \mid z_\ell)$$

$$= p(z_1 \mid \pi) \left[\prod_{j=2}^{n} p(z_j \mid z_{j-1}, A) \right] \prod_{\ell=1}^{n} p(x_\ell \mid z_\ell, \theta)$$

$\underline{X} = \{x_1, \dots, x_n\}$
$\underline{Z} = \{z_1, \dots, z_n\}$

emphasize the dependence of the trans & emission proba on the model parameters

Compact representation :
$p(z_j \mid z_{j-1}, A) = P(Z_j = z_j \mid Z_{j-1} = z_{j-1}, A)$ etc.

Several challenges arise :

- How to compute efficiently the likelihood $p(x_1, \dots, x_n)$?
- Given observations $x_1, \dots, x_n$ and the model parameters $\{\pi, A, \theta\}$, find a sequence $z_1, \dots, z_n$ of latent variables that best explain the observations → 'decoding' in speech processing
  → VITERBI ALGORITHM
- How to fit the model ? → BAUM-WELCH ALGORITHM (EM algo) (training)

I. LIKELIHOOD IN AN HMM.

I.1. A direct approach.

The likelihood function $p(\underline{X}) = p(x_1, \dots, x_n)$ can be obtained from the joint distribution derived on page 3 by marginalizing over the latent variables $z_1, \dots, z_n$:

$$p(\underline{X} \mid \theta) = \sum_{z} p(\underline{X}, \underline{Z} \mid \theta)$$

$$= \sum_{z_1, \dots, z_n} \underbrace{p(z_1) \left[\prod_{j=2}^{n} p(z_j \mid z_{j-1}) \right] \prod_{\ell=1}^{n} p(x_\ell \mid z_\ell)}_{O(n) \text{ calculations}}.$$

$\in \{0,1\}^K$

⇒ K^n terms in the summation

⇒ Total of $O(nK^n)$ calculations.

The number of computations needed to evaluate the likelihood grows exponentially with n ⇒ becomes quickly infeasible.

Consequences
- Need an alternative approach to evaluate it (section I.2)
- Direct maximization is also intractable → EM algorithm will save us. (see section II)

I.2. Forward & Backward Variables

→ We first turn our attention to the posterior probability $p(z_j \mid \underline{X})$; where $\underline{X} = (x_1, .., x_n)$; $j \in \{1, .., n\}$

$!! \ \gamma(z_j)$

(this quantity will be useful later when deriving the EM algorithm ⇒ we also need convenient / efficient ways to evaluate it).

Bayes ⇒ $\gamma(z_j) = \dfrac{p(\underline{X} \mid z_j)\, p(z_j)}{p(\underline{X})}$ cf Appendix page 17

Cf also pages 10-13 in the Chapter on Kalman Filtering

$$= \frac{p(x_1, .., x_j \mid z_j)\, p(x_{j+1}, .., x_n \mid z_j)\, p(z_j)}{p(\underline{X})}$$

$$= \frac{p(x_1, .., x_j, z_j)\, p(x_{j+1}, .., x_n \mid z_j)}{p(\underline{X})}$$

$$=: \frac{\alpha(z_j)\, \beta(z_j)}{p(\underline{X})}$$

where $\boxed{\begin{aligned} \alpha(z_j) &= p(x_1, .., x_j, z_j) \\ \beta(z_j) &= p(x_{j+1}, .., x_n \mid z_j) \end{aligned}}$

→ We establish recurrence relations for the variables α and β.
(again, compare with the relations obtained in the context of Kalman filtering : it is the same — except that the latent variable is continuous there ; so we just need to replace integrals with summations)

Idea: go from step $j-1$ to step j : multiply the LHS by trans & emission proba.

• $\alpha(z_{j-1}) = p(x_1, .., x_{j-1}, z_{j-1})$

$$\alpha(z_{j-1}) \underbrace{p(z_j \mid z_{j-1})}_{\text{trans. proba}} = p(x_1, .., x_{j-1}, z_{j-1})\, p(z_j \mid z_{j-1})$$

$$= \underline{p(z_{j-1} \mid x_1, .., x_{j-1})\, p(x_1, .., x_{j-1})}\, \underline{p(z_j \mid z_{j-1})}$$

$$= p(z_j, z_{j-1} \mid x_1, .., x_{j-1})\, p(x_1, .., x_{j-1})$$

(conditionally on z_{j-1}, z_j is independent of $x_1, .., x_{j-1}$)

$$\alpha(z_{j-1})\, p(z_j \mid z_{j-1}) \underbrace{p(x_j \mid z_j)}_{\text{emission proba}} = \underline{p(z_j, z_{j-1} \mid \underline{X}_{j-1})\, p(\underline{X}_{j-1})}\, \underline{p(x_j \mid z_j)}$$

$$= p(z_j, z_{j-1}, x_j \mid \underline{X}_{j-1})\, p(\underline{X}_{j-1})$$

(conditionally on z_j, x_j is independent of $z_{j-1}, x_1, .., x_{j-1}$)

⇒ Marginalize over z_{j-1} to get

$$\sum_{z_{j-1}} \alpha(z_{j-1})\, p(z_j \mid z_{j-1})\, p(x_j \mid z_j) = \sum_{z_{j-1}} p(z_j, z_{j-1}, \underline{X}_j)$$

$$= p(z_j, \underline{X}_j)$$

$$= \alpha(z_j)$$

$$\boxed{\alpha(z_j) = \sum_{z_{j-1}} \alpha(z_{j-1}) \underbrace{p(z_j \mid z_{j-1})}_{\text{transition}} \underbrace{p(x_j \mid z_j)}_{\text{emission}}}$$

$2 \leq j \leq n$

FORWARD message passing from time $j-1$ to time j.

α = FORWARD VARIABLE.

• Similarly, we obtain a recurrence relation for β :

$$\beta(z_j) = p(x_{j+1}, .., x_n \mid z_j) \qquad 1 \leq j \leq n-1$$

$$= \sum_{z_{j+1}} p(x_{j+1}, .., x_n, z_{j+1} \mid z_j)$$

$$= \sum_{z_{j+1}} p(x_{j+1}, .., x_n \mid z_{j+1}, \cancel{z_j})\, p(z_{j+1} \mid z_j)$$

$$\beta(z_j) = \sum_{z_{j+1}} \underbrace{p(x_{j+2},\dots,x_n \mid z_{j+1})}_{\beta(z_{j+1})}\, p(x_{j+1}\mid z_{j+1})\, p(z_{j+1}\mid z_j)$$

$$\boxed{\beta(z_j) = \sum_{z_{j+1}} \beta(z_{j+1})\, \underbrace{p(x_{j+1}\mid z_{j+1})}_{\text{emission}}\, \underbrace{p(z_{j+1}\mid z_j)}_{\text{transition}}}$$

$1 \leq j \leq n-1$

BACKWARD message passing from time $j+1$ to time j.

β = BACKWARD VARIABLE.

→ We usually work with scaled versions of the fwd and bwd variables; to avoid numerical issues.

Specifically,

- $\hat{\alpha}(z_j) = p(z_j \mid x_1,\dots,x_j) = \dfrac{\alpha(z_j)}{p(x_1,\dots,x_j)}$

Introducing $c_j = p(x_j \mid x_1,\dots,x_{j-1})$, we see that

$$p(x_1,\dots,x_j) = p(x_j \mid x_1,\dots,x_{j-1})\, p(x_{j-1}\mid x_1,\dots,x_{j-2}) \times \dots \times p(x_2\mid x_1)\, p(x_1)$$
$$= \prod_{m=1}^{j} c_m$$

so that $\boxed{\alpha(z_j) = \left(\prod_{m=1}^{j} c_m\right) \hat{\alpha}(z_j)}$

unscaled fwd variable (bwd) — scaled fwd variable (bwd)

- likewise, define $\boxed{\beta(z_j) = \left(\prod_{m=j+1}^{n} c_m\right) \hat{\beta}(z_j)}$,

so that $\hat{\beta}(z_j) = \dfrac{\beta(z_j)}{p(x_{j+1},\dots,x_n \mid x_1,\dots,x_j)}$ ← since $\underbrace{p(\underline{x})}_{p(\underline{x}_n \mid \underline{x}_j)\, p(\underline{x}_j)} = \underbrace{\prod_{m=1}^{j} c_m}_{p(\underline{x}_j)} \prod_{m=j+1}^{n} c_m$

The scaled fwd and bwd variables also satisfies recurrence relations:

$$\boxed{\begin{aligned} c_j\, \hat{\alpha}(z_j) &= p(x_j\mid z_j) \sum_{z_{j-1}} \hat{\alpha}(z_{j-1})\, p(z_j \mid z_{j-1}) \\ c_{j+1}\, \hat{\beta}(z_j) &= \sum_{z_{j+1}} \hat{\beta}(z_{j+1})\, p(x_{j+1}\mid z_{j+1})\, p(z_{j+1}\mid z_j) \end{aligned}}$$

Rk: Since the $\hat{\alpha}(z_j)$ sum to 1, c_j is a renorm. factor ⇒ easy to compute.

Note that $\gamma(z_j) = p(z_j \mid x_1,\dots,x_n)$ introduced on page 5 can

$$= \frac{\alpha(z_j)\,\beta(z_j)}{p(\underline{x})}$$

easily be re-expressed in terms of the scaled variables:

$$\gamma(z_j) = \frac{\alpha(z_j)\,\beta(z_j)}{\left(\prod_{m=1}^{j} c_m\right)\left(\prod_{m=j+1}^{n} c_m\right)} = \hat{\alpha}(z_j)\,\hat{\beta}(z_j).$$

→ Back to our original goal: computing the likelihood efficiently.

Well,

$$p(x_1,\dots,x_n) = \sum_{z_n} p(x_1,\dots,x_n,z_n) = \sum_{z_n} \alpha(z_n).$$

⇒ To compute the likelihood, we must complete a FORWARD PASS through the data:

$$\alpha(z_1) \rightarrow \alpha(z_2) \rightarrow \alpha(z_3) \rightarrow \dots / \dots \rightarrow \alpha(z_n)$$

Use recurrence relation established on page 6

Then sum over the z_n variable

COMPUTING THE LIKELIHOOD

So, have we gained anything in terms of computational cost?

- Updating the forward variable is relatively cheap: summation over z_{j-1} requires $O(K)$ operations.
- To get $\alpha(z_n)$, the operation needs to be repeated n times
- Computational cost is $O(nK)$ $\equiv$ linear in the number of observations.

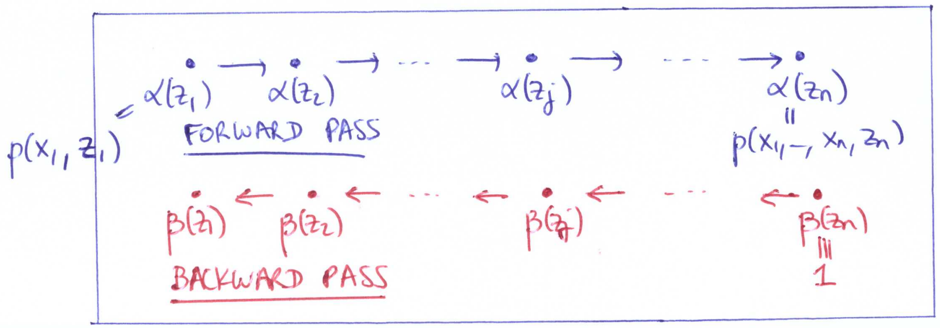
That's a huge improvement

Remark: Alternatively, since $\gamma(z_j) = p(z_j \mid x_1, .., x_n) = \dfrac{\alpha(z_j)\beta(z_j)}{p(x_1, .., x_n)}$,

summing over z_j yields $\sum_{z_j} \gamma(z_j) = 1 \Rightarrow$

$$\boxed{p(x_1, .., x_n) = \sum_{z_j} \alpha(z_j)\,\beta(z_j)}$$

An expression involving the FWD and BWD variables. If you want to make use of this expression, a fwd pass followed by a bwd pass through the data must be completed



Remark: Initialization.

- $\alpha(z_1) = p(x_1, z_1) = p(x_1 \mid z_1)\,\underbrace{p(z_1)}_{\sim \pi \text{ (page 1)}}$
- $\beta(z_n) = 1$. since looking back at the derivation of the recurrence relation for the backward variable,

$$\beta(z_m) = p(x_n \mid z_{n-1}) = \sum_{z_n} p(x_n, z_n \mid z_{n-1})$$
$$= \sum_{z_n} p(x_n \mid z_n)\, p(z_n \mid z_{n-1})$$
$$= \sum_{z_n} \underset{\substack{|||\\ 1}}{\beta(z_n)}\, p(z_n \mid z_{n-1})\, p(x_n \mid z_n).$$

→ Note that the likelihood can also be expressed as:

$p(x_1, .., x_n) = \left(\prod_{j=1}^{n} c_j\right)$. Rk: It will be important to monitor the value of the likelihood during the EM optimization

& $\log lik = \sum_j \log c_j$. Good.

II. EM ALGORITHM FOR HMM

We make use of the EM algorithm to find an efficient way for maximizing the likelihood.

Step I: Complete log-likelihood.

Recall the expression of the joint density established on page 3:

$$\mathcal{L}_c = \log p(\underline{X}, \underline{Z} \mid \theta)$$
$$= \log \left\{ p(z_1 \mid \pi) \left[\underbrace{\prod_{j=2}^{n} p(z_j \mid z_{j-1}, A)}_{} \right] \prod_{\ell=1}^{n} p(x_\ell \mid z_\ell, \theta) \right\}$$

$$\prod_{k=1}^{K} \prod_{m=1}^{K} a_{mk}^{1(z_{j-1,m}=1)1(z_{j,k}=1)} = \prod_{k,m} a_{mk}^{z_{j-1,m} z_{j,k}}$$

$= a_{mk}$ if from time $j-1$ to time j, there is a transition from state m to state k.

Similarly, $p(x_\ell \mid z_\ell, \theta) = \prod_{s=1}^{K} \left[p(x_\ell \mid \theta_s)\right]^{z_{\ell s}}$

$\Downarrow$

$$\mathcal{L}_c = \log\left\{ \underbrace{p(z_1 \mid \pi)}_{= \prod_{k=1}^{K} \pi_k^{z_{1k}}} \left[\prod_{j=2}^{n} \prod_{k=1}^{K} \prod_{m=1}^{K} a_{mk}^{1(z_{j-1,m}=1)\,1(z_{j,k}=1)} \right] \prod_{\ell=1}^{n} \prod_{s=1}^{K} \left[p(x_\ell \mid \theta_s)\right]^{z_{\ell s}} \right\}$$

$$\boxed{\mathcal{L}_c = \sum_{k=1}^{K} z_{1k} \log \pi_k + \sum_{j=2}^{n} \sum_{k=1}^{K} \sum_{\ell=1}^{K} z_{j-1,\ell}\, z_{j,k} \log a_{\ell k} + \sum_{\ell=1}^{n} \sum_{k=1}^{K} z_{\ell k} \log p(x_\ell \mid \theta_k)}$$

Step II. E-step.

We derive the expected value of $\mathcal{L}_c$ with respect to the latent variables $z_1, \dots, z_n$, conditionally on $x_1, \dots, x_n$, and the current model parameter estimates $\theta^{(m)} = \{\pi^{(m)}, A^{(m)}, \varphi^{(m)}\}$:

$$Q(\theta, \theta^{(m)}) = E_{\underline{Z}}\left\{ \mathcal{L}_c \mid \underline{X} = \underline{x},\ \theta = \theta^{(m)} \right\}.$$

→ We need to compute

- $E(z_{jk} \mid x_1, \dots, x_n, \theta^{(m)}) = P(Z_{jk} = 1 \mid x_1, \dots, x_n, \theta^{(m)})$ $\quad 1 \le j \le n,\ 1 \le k \le K$

  ↳ Knowledge of the posterior distribution $p(z_j \mid x_1, \dots, x_n)$ required.

- $E(z_{j-1,\ell}\, z_{j,k} \mid x_1, \dots, x_n, \theta^{(m)}) = P(Z_{j-1,\ell} = 1, Z_{j,k} = 1 \mid x_1, \dots, x_n, \theta^{(m)})$

We need the joint distribution $p(z_{j-1}, z_j \mid x_1, \dots, x_n)$

- Half of the work is done. Indeed, $p(z_j \mid x_1, \dots, x_n)$ can be expressed in terms of the (scaled) forward and backward variables:

$$\boxed{p(z_j \mid x_1, \dots, x_n)} = \gamma(z_j) = \hat{\alpha}(z_j)\, \hat{\beta}(z_j)$$ (see page 8)

these are OK to compute.

- Second half of the job is computing $p(z_{j-1}, z_j \mid x_1, \dots, x_n)$. Fortunately, this joint probability can be expressed in terms of the fwd and bwd variables as well. Indeed,

$$p(z_{j-1}, z_j \mid x_1, \dots, x_n) = \frac{p(x_1, \dots, x_n \mid z_{j-1}, z_j)\, p(z_{j-1}, z_j)}{p(x_1, \dots, x_n)}$$

cf App. page 18

$$= \frac{p(\underline{x}_{j-1} \mid z_{j-1})\, p(x_j \mid z_j)\, p(x_{j+1} \dots x_n \mid z_j)\, p(z_j \mid z_{j-1})\, p(z_{j-1})}{p(\underline{x}_n)}$$

$$= \frac{\alpha(z_{j-1})\, p(x_j \mid z_j)\, p(z_j \mid z_{j-1})\, \beta(z_j)}{p(\underline{x}_n)}$$

$$\boxed{p(z_{j-1}, z_j \mid x_1, \dots, x_n)} = c_j^{-1}\, \hat{\alpha}(z_{j-1})\, \underbrace{p(x_j \mid z_j)}_{\text{emission}}\, \underbrace{p(z_j \mid z_{j-1})}_{\text{transition}}\, \hat{\beta}(z_j)$$

everything here is easily computable.

→ Putting things together, we see that computing the expected value of the complete log-likelihood is tractable once the following quantities are computed (from a fwd + bwd pass through the data)

$$\hat{p}_{jkm} := P(Z_{jk} = 1 \mid x_1, \dots, x_n, \theta^{(m)})$$

$$\tilde{p}_{j\ell km} := P(Z_{j-1,\ell} = 1,\ Z_{j,k} = 1 \mid x_1, \dots, x_n, \theta^{(m)})$$

Summarizing,

$$\boxed{\begin{aligned} Q(\theta,\theta^{(m)}) = &\sum_{k=1}^{K} \hat{p}_{1km} \log \pi_k \\ &+ \sum_{j=2}^{n}\sum_{k=1}^{K}\sum_{\ell=1}^{K} \tilde{p}_{j\ell km} \log a_{\ell k} \\ &+ \sum_{j=1}^{n}\sum_{k=1}^{K} \hat{p}_{jkm} \log p(x_j \mid \theta_k). \end{aligned}}$$

Step III. M-step.

Maximization with respect to π_k, $a_{\ell k}$ and θ_k can be done separately. Details are omitted and left as an exercise. We get:

- $$\pi_k^{(m+1)} = \frac{\hat{p}_{1km}}{\sum_{k=1}^{K} \hat{p}_{1km}}, \quad 1 \le k \le K$$

- $$a_{\ell k}^{(m+1)} = \frac{\sum_{j=2}^{n} \tilde{p}_{j\ell km}}{\sum_{j=2}^{n}\sum_{k'=1}^{K} \tilde{p}_{j\ell k'm}}, \quad 1 \le \ell, k \le K$$

Standard, use Lagrange multipliers for example.

Note that indeed, $\sum_{k=1}^{K} a_{\ell k}^{(m+1)} = 1$

- Maximization with respect to θ_k depends on the particular emission probability considered (Binomial / Poisson / Normal / ...).

Ex: $p(x_j \mid \theta_k) = \mathcal{N}(x_j \mid \mu_k, \Sigma_k)$, we get

$$\mu_k^{(m+1)} = \frac{\sum_{j=1}^{n} \hat{p}_{jkm} x_j}{\sum_{j=1}^{n} \hat{p}_{jkm}}$$

and

$$\Sigma_k^{(m+1)} = \frac{\sum_{j=1}^{n} \hat{p}_{jkm} (x_j - \mu_k^{(m+1)})(x_j - \mu_k^{(m+1)})^t}{\sum_{j=1}^{n} \hat{p}_{jkm}}.$$

+ initialization required.

III. VITERBI ALGORITHM.

We turn our attention to the problem of decoding: given observations $x_1, \dots, x_n$, determine the states of the Markov Chain which are most likely.

- **LOCAL DECODING**: Given $x_1, \dots, x_n$, what is the most likely state at time j, $1 \le j \le n$?
  To answer this question, we need to compute $p(z_j \mid x_1, \dots, x_n)$.
  We have already solved this problem! Indeed, the posterior distribution can be expressed in terms of $\hat{\alpha}$ and $\hat{\beta}$:
  $$p(z_j \mid x_1, \dots, x_n) = \gamma(z_j) = \hat{\alpha}(z_j)\,\hat{\beta}(z_j)$$ (page 8)

- **GLOBAL DECODING**: Given $x_1, \dots, x_n$, what is the most likely sequence of states $z_1, \dots, z_n$? We want to solve:
  $$\arg\max_{z_1,\dots,z_n} p(z_1, \dots, z_n \mid x_1, \dots, x_n)$$

  Proceed recursively:

$$\begin{aligned} \arg\max_{z_1,\dots,z_n} p(z_1,\dots,z_n \mid x_1,\dots,x_n) &= \arg\max_{z_1,\dots,z_n} p(z_1,\dots,z_n, x_1,\dots,x_n) \\ &= \arg\max_{z_n} \underbrace{\max_{z_1,\dots,z_{n-1}} p(z_1,\dots,z_n, x_1,\dots,x_n)}_{\shortparallel \atop \delta_n(z_n)} \end{aligned}$$

Recurrence relation for $\delta_n(z_n)$:

- $\delta_n(z_n) = \max_{z_1,\dots,z_{n-1}} p(z_n, \underline{x}_n)$

$$= \max_{z_1,\dots,z_{n-1}} \left\{ p(z_n|z_{n-1})\, p(x_n|z_n)\, p(z_{n-1}, \underline{x}_{n-1}) \right\}$$

$$= \max_{z_{n-1}} \left\{ p(z_n|z_{n-1})\, p(x_n|z_n) \max_{z_1,\dots,z_{n-2}} p(z_{n-1}, \underline{x}_{n-1}) \right\}$$

$$\delta_n(z_n) = \max_{z_{n-1}} \left\{ p(z_n|z_{n-1})\, p(x_n|z_n)\, \delta_{n-1}(z_{n-1}) \right\} \quad \text{for } n \geq 2.$$

- For $n=1$, initialization is $\delta_1(z_1) = p(z_1, x_1) = p(x_1|z_1)\,p(z_1)$.

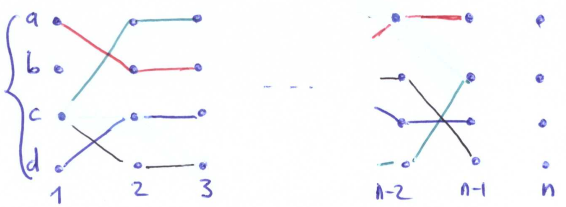
↳ We actually want the maximizing sequence, i.e. the argmax, not the max → keep track of the maximizing sequence at each step.

- $\hat{z}_n = \arg\max_{z_n} \delta_n(z_n)$
- But we need to compute $\delta_n(z_n)$ from its definition:

$$\delta_n(z_n) = \max_{z_1,\dots,z_{n-1}} p(z_n, \underline{x}_n)$$

So, if you keep track of the maximizing sequence up to step $n-1$; ie $\hat{z}_1,\dots,\hat{z}_{n-1}$, you can easily select $\hat{z}_n$ maximizing δ_n:

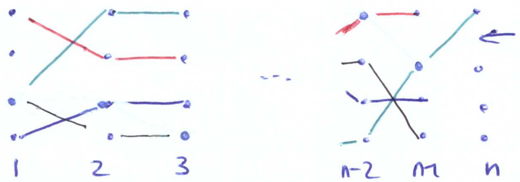
MC with 4 possible states a, b, c, d



are the 4 sequences $z_1,\dots,z_{n-1}$ corresponding to the 4 possible states of the MC maximizing $p(z_{n-1}, \underline{x}_{n-1})$ over $z_1,\dots,z_{n-2}$, and terminating at a, b, c and d; ie sequences leading to $\delta_{n-1}(a), \delta_{n-1}(b), \delta_{n-1}(c), \delta_{n-1}(d)$

At step n, you chain each state a, b, c, d with one of the 4 existing path:

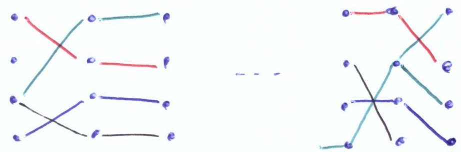
To compute $\delta_n(a)$, select the red, green, black or blue path such that $p(a|z_{n-1})\,p(x_n|a)\,\delta_{n-1}(z_{n-1})$ is maximized



The green path won!

And repeat the procedure, to compute $\delta_n(b), \delta_n(c), \delta_n(d)$. You may end up with some thing like that:

gain, one
as to go find
o derive all
hese paths, & then
nd to select the appropriate path.



Then select the most likely path; ie the terminal value z_n that maximizes $\delta_n(z_n)$

What was said going from $(n-1)$ to n holds for step $(j-1)$ to j.

(i) Initialization: $\delta_1(z_1) = p(x_1|z_1)\,p(z_1)$; $\quad \psi_1(z_1) = 0$

(ii) For $j = 2,\dots,n$:

$$\delta_j(z_j) = p(x_j|z_j) \max_{z_{j-1}} \left\{ p(z_j|z_{j-1})\, \delta_{j-1}(z_{j-1}) \right\}$$

$$\psi_j(z_j) = \arg\max_{z_{j-1}} \left\{ p(z_j|z_{j-1})\, \delta_{j-1}(z_{j-1}) \right\}$$

(iii) Termination. $\hat{z}_n = \arg\max_{z_n} \delta_n(z_n)$

(iv) Backtracking: for $j = n-1, \dots, 1$: $\hat{z}_j = \psi_{j+1}(\hat{z}_{j+1})$

VITERBI ALGORITHM

Remark: To prevent underflow, it is preferable to work on a log scale:

(i) $\bar{\xi}_1(z_1) = \log p(z_1) + \log p(x_1 | z_1)$
$\bar{\varphi}_1(z_1) = 0$

(ii) $\bar{\xi}_j(z_j) = \log p(x_j | z_j) + \max_{z_{j-1}} \{ \log p(z_j | z_{j-1}) + \bar{\xi}_{j-1}(z_{j-1}) \}$
$\bar{\varphi}_j(z_j) = \arg\max_{z_{j-1}} \{ \log p(z_j | z_{j-1}) + \bar{\xi}_{j-1}(z_{j-1}) \}$.

(iii) $\hat{z}_n = \operatorname{argmax}_{z_n} \bar{\xi}_n(z_n)$

(iv) $\hat{z}_j = \bar{\varphi}_{j+1}(\hat{z}_{j+1})$, $j = n-1, \dots, 1$.

VITERBI (log scale)

IV – APPENDIX

We derive in this appendix some straightforward (but tiring) useful expressions for conditional probabilities in the HMM model (applies also to Kalman filtering).

(i) $p(x_1 \dots x_n | z_j) = p(x_1, \dots, x_j | z_j) \, p(x_{j+1}, \dots, x_n | z_j)$

Recall that the joint density is given by

$$p(\underline{x}_n, \underline{z}_n) = p(z_1) \left[\prod_{j=2}^{n} p(z_j | z_{j-1}) \right] \prod_{j=1}^{n} p(x_j | z_j)$$

Marginalize over all variables except z_j:

$$p(\underline{x}_n, z_j) = \sum_{z_1, \dots z_{j-1}, z_{j+1}, \dots z_n} p(\underline{x}_n, \underline{z}_n)$$

$$= \left[\sum_{z_1, \dots z_{j-1}} p(z_1) \prod_{k=2}^{j} p(z_k | z_{k-1}) \prod_{\ell=1}^{j} p(x_\ell | z_\ell) \right] \times \left[\sum_{z_{j+1}, \dots z_n} \prod_{k=j+1}^{n} p(z_k | z_{k-1}) \prod_{\ell=j+1}^{n} p(x_\ell | z_\ell) \right]$$

– The first factor is $\sum_{z_1, \dots z_{j-1}} p(x_1, \dots x_j, z_1, \dots z_j)$
$= p(x_1, \dots x_j, z_j)$.

– The second factor is given by $p(x_{j+1}, \dots x_n | x_1, \dots x_j, z_j)$ since we have that

$$p(x_{j+1}, \dots, x_n | x_1, \dots, x_j, z_j) = \frac{p(x_1, \dots, x_n, z_j)}{p(x_1, \dots x_j, z_j)}.$$

So that

$$p(x_1, \dots, x_n, z_j) = p(x_1, \dots, x_j, z_j) \, p(x_{j+1}, \dots, x_n | \cancel{x_1, \dots, x_j}, z_j)$$

Divide both sides by $p(z_j)$ to get:

$$p(x_1, \dots, x_n | z_j) = p(x_1, \dots, x_j | z_j) \, p(x_{j+1}, \dots, x_n | z_j) \quad \blacksquare$$

(ii) $p(x_1, \dots, x_n | z_{j-1}, z_j) = p(x_1, \dots, x_{j-1} | z_{j-1}) \, p(x_j | z_j) \, p(x_{j+1}, \dots x_n | z_j)$

Starting point is the same: marginalize the joint density over all variables except z_{j-1}, z_j:

$$p(\underline{x}_n, z_{j-1}, z_j) = \left[\sum_{z_1, \dots z_{j-2}} p(z_1) \prod_{k=2}^{j-1} p(z_k | z_{k-1}) \prod_{\ell=1}^{j-1} p(x_\ell | z_\ell) \right]$$

$$\times \, p(z_j | z_{j-1}) \, p(x_j | z_j)$$

$$\times \left[\sum_{z_{j+1}, \dots z_n} \prod_{k=j+1}^{n} p(z_k | z_{k-1}) \prod_{\ell=j+1}^{n} p(x_\ell | z_\ell) \right]$$

– The first term is $\sum_{z_1, \dots z_{j-2}} p(x_1, \dots x_{j-1}, z_1, \dots z_{j-1})$
$= p(x_1, \dots x_{j-1}, z_{j-1})$

– The second term is $\sum_{z_{j+1}, \dots z_n} \left\{ \frac{1}{p(z_j)} p(x_{j+1}, \dots x_n, z_j, \dots, z_n) \right\}$
$= p(x_{j+1}, \dots x_n | z_j)$

19

$\Rightarrow$ We get that

$$p(\underline{X}_n, z_{j-1}, z_j) = p(x_1 \to x_{j-1}, z_{j-1})\, p(z_j | z_{j-1})\, p(x_j | z_j)\, p(x_{j+1} \to x_n | z_j)$$

Divide both sides by $p(z_{j-1}, z_j) = p(z_j | z_{j-1})\, p(z_{j-1})$ to get

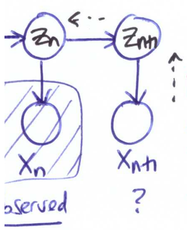
$$p(\underline{X}_n | z_{j-1}, z_j) = p(\underline{X}_{j-1} | z_{j-1})\, p(x_j | z_j)\, p(x_{j+1} \to x_n | z_j) \quad \blacksquare$$

OK, there are easier ways to get to the result $\rightarrow$ d-separation.

V- PREDICTION USING HMM

- You may then use the HMM for prediction, which requires the computation of the PREDICTIVE DISTRIBUTION $p(x_{n+1} | \underline{x}_n)$, which can be computed, making use of the Markovian properties of the model. Indeed,

$$p(x_{n+1} | x_1, \dots, x_n) = \sum_{z_{n+1}} p(x_{n+1}, z_{n+1} | \underline{x}_n)$$



Backtrack: Artificially introduce z_{n+1} and z_n

$$= \sum_{z_{n+1}} p(x_{n+1} | z_{n+1})\, p(z_{n+1} | \underline{x}_n)$$

$$= \sum_{z_{n+1}} p(x_{n+1} | z_{n+1}) \sum_{z_n} p(z_n, z_{n+1} | \underline{x}_n)$$

$$= \sum_{z_{n+1}} p(x_{n+1} | z_{n+1}) \sum_{z_n} p(z_{n+1} | z_n)\, p(z_n | \underline{x}_n)$$

$$= \underbrace{\sum_{z_{n+1}} \underbrace{p(x_{n+1} | z_{n+1})}_{\text{emission}} \sum_{z_n} \underbrace{\hat{\alpha}(z_n)}_{\text{fwd var.}} \underbrace{p(z_{n+1} | z_n)}_{\text{transition}}}_{\text{everything is available. Good.}}$$

20

The predictive distribution can be rewritten in a compact form using matrix multiplications.

↘ Recall : • $A = \begin{pmatrix} a_{11} & \text{—} & a_{1K} \\ | & & | \\ a_{K1} & \text{—} & a_{KK} \end{pmatrix}$ where $a_{ij} = P(z_{n,j} = 1 | z_{n-1,i} = 1)$ (transition proba)

↘ Introduce : • $\hat{\alpha}_n = \begin{pmatrix} p(z_{n1} = 1 | x_1 \dots x_n) \\ \vdots \\ p(z_{nK} = 1 | x_1 \dots x_n) \end{pmatrix}$ (fwd variable)

• $p(x_{n+1}) = \begin{pmatrix} p(x_{n+1} | z_{n+1,1} = 1) \\ \vdots \\ p(x_{n+1} | z_{n+1,K} = 1) \end{pmatrix}$ (emission proba)

Then $\boxed{p(x_{n+1} | x_1, \dots, x_n) = \hat{\alpha}_n^t\, A\, p(x_{n+1})}$

- Higher order predictive distributions can be obtained similarly:

$$\boxed{p(x_{n+k} | x_1, \dots, x_n) = \hat{\alpha}_n^t\, A^k\, p(x_{n+k}) \quad (k \geq 1)}$$

- In the derivation page 19, we also get for free the distribution $p(z_{n+1} | x_1, \dots, x_n)$ since:

$$p(z_{n+1} | \underline{X}_n) = \sum_{z_n} p(z_{n+1}, z_n | \underline{X}_n)$$

$$= \sum_{z_n} p(z_{n+1} | z_n)\, p(z_n | \underline{X}_n)$$

$$= \sum_{z_n} p(z_{n+1} | z_n)\, \hat{\alpha}(z_n).$$