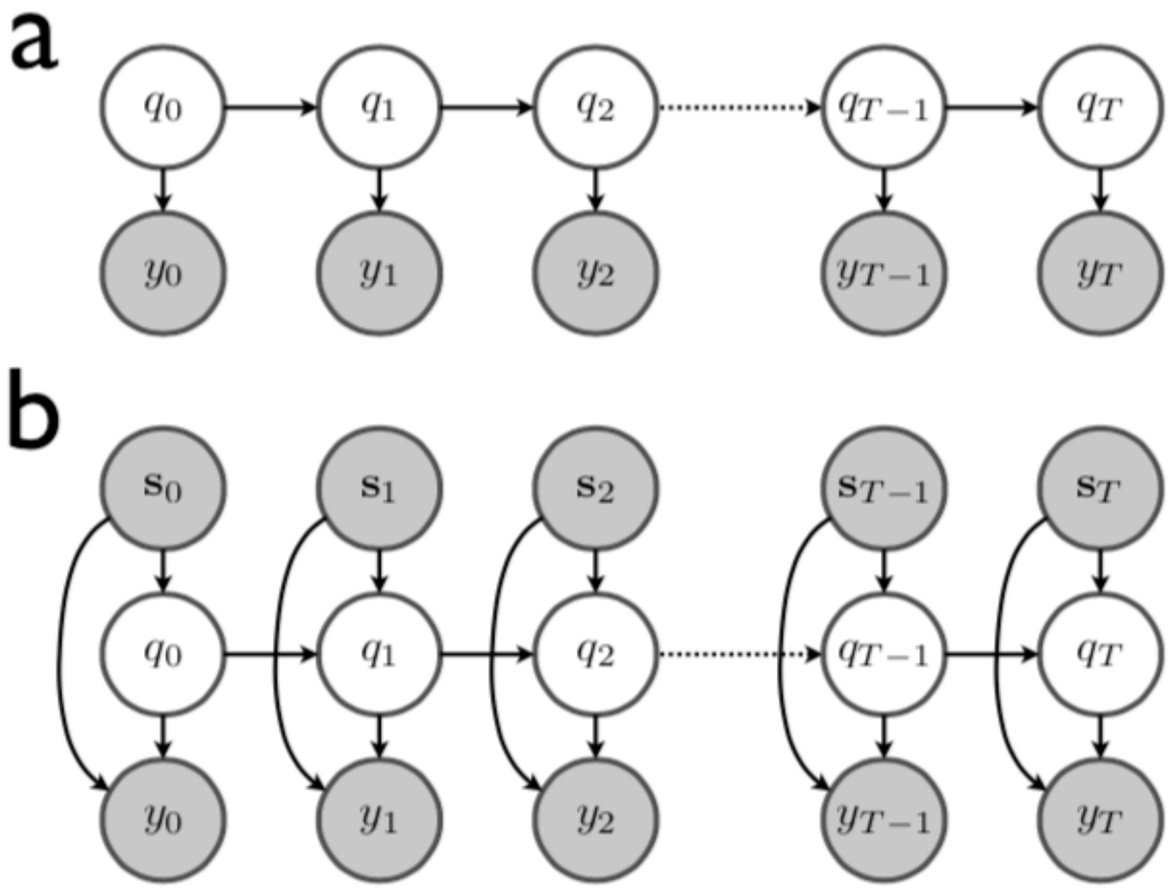
Stimulus-driven hidden Markov model

Introductions

a. Typical hidden Markov model

b. Stimulus-driven hidden Markov model



Notations

Variable/parameters	Mathematical expression	Comment
Stimulus	$\mathbf{s}_t$	size: [num_feature, num_time],

		continuous value
Hidden states	q_t	size: [1, num_time], discrete value, ranges in 1~num_state (might need a q_0 ?)
Output	y_t	size: [1, num_time], discrete value, ranges in 1~num_out (might need a y_0 ?)
Transition filter	$\mathbf{F}_{m,n}$	size :[num_state, num_state, num_feature], continuous value, element (m,n): m to n
Transition matrix	$\alpha_{m,n,t} = \frac{\exp(\mathbf{F}_{m,n}\mathbf{s}_t)}{\sum_l \exp(\mathbf{F}_{m,l}\mathbf{s}_t)}$	Size: [num_state, num_state, num_time], time-varying transition matrix, $\sum_n \alpha_{m,n,t} = 1$
Emission filter	$\mathbf{G}_{m,i}$	Size: [num_state, num_out, num_feature]
Emission matrix	$\eta_{m,i,t} = \frac{\exp(\mathbf{G}_{m,i}\mathbf{s}_t)}{\sum_j \exp(\mathbf{G}_{m,j}\mathbf{s}_t)}$	Size: [num_state, num_out, num_time], time-varying emission matrix

Expectation-maximization

Expected complete log-likelihood (ECLL)

$$\begin{aligned}\langle L(\boldsymbol{\theta}|\mathbf{y},\mathbf{q},\mathbf{S})\rangle_{\hat{p}(\mathbf{q})} &= \sum_{n=1}^{N} \hat{p}\left(q_0 = n\right) \log \pi_n + \sum_{t=1}^{T}\sum_{n=1}^{N}\sum_{m=1}^{N} \hat{p}\left(q_{t-1} = n, q_t = m\right) \log \alpha_{nm,t} \\ &+ \sum_{t=0}^{T}\sum_{n=1}^{N} \hat{p}\left(q_t = n\right) \log \eta_{ny_t,t} \quad (1)\end{aligned}$$

To optimize ECLL, we can seperately optimize

$$\sum_{t=1}^{T}\sum_{m=1}^{N} \hat{p}(q_{t-1} = n, q_t = m) \log \alpha_{n,m,t} \quad (2)$$

$$\sum_{t=1}^{T} \hat{p}(q_t = n) \log \eta_{n,y_t,t} \quad (3)$$

Denote: $U_{nmt} = \hat{p}\left(q_{t-1} = n, q_t = m\right)$, and $V_{nt} = \hat{p}\left(q_t = n\right)$, they are computed as follows

$$U_{nmt} = \hat{p}(q_{t-1} = n, q_t = m) = p(q_{t-1} = n, q_t = m|\mathbf{y}, \theta, \mathbf{S}) = \frac{a_{n,t}\alpha_{nmt}\eta_{my_t}b_{m,t+1}}{p(\mathbf{y}|\theta, \mathbf{S})} \quad (4)$$

$$V_{nt} = \hat{p}(q_t = n) = p(q_t = n|\mathbf{y}, \theta, \mathbf{S}) = \frac{a_{n,t}b_{n,t}}{p(\mathbf{y}|\theta, \mathbf{S})} \quad (5)$$

$$p(\mathbf{y}|\theta, \mathbf{S}) = \sum_{n=1}^{N} a_{n,T} \quad (6)$$

For (2),

$$\begin{aligned}
&\sum_{t=1}^{T}\sum_{m=1}^{N}\hat{p}\left(q_{t-1} = n, q_t = m\right)\log\alpha_{n,m,t} \\
&= \sum_{t=1}^{T}\sum_{m=1}^{N} U_{nmt}\log\left(\frac{\exp(\mathbf{F}_{n,m}\mathbf{s}_t)}{\sum_l \exp(\mathbf{F}_{n,l}\mathbf{s}_t)}\right) \\
&= \sum_{t=1}^{T}\sum_{m=1}^{N} U_{nmt}\mathbf{F}_{n,m}\mathbf{s}_t - \sum_{t=1}^{T}\sum_{m=1}^{N} U_{nmt}\log(\sum_l \exp(\mathbf{F}_{n,l}\mathbf{s}_t)) \quad (7)
\end{aligned}$$

The gradient with regard to $\mathbf{F}_{n,j}$ is,

$$\begin{aligned}
\frac{\partial(2)}{\partial\mathbf{F}_{nj}} &= \sum_{t=1}^{T}\left(U_{njt} - (\sum_{m=1}^{N} U_{nmt})\alpha_{njt}\right)\mathbf{s}_t \\
&= \sum_{t=1}^{T}\left(U_{njt} - (\sum_{m=1}^{N} U_{nmt})\frac{\exp(\mathbf{F}_{n,j}\mathbf{s}_t)}{\sum_l \exp(\mathbf{F}_{n,l}\mathbf{s}_t)}\right)\mathbf{s}_t \quad (8)
\end{aligned}$$

The second derivative is,

$$\frac{\partial^2(2)}{\partial\mathbf{F}_{nj}^2} = -\sum_{t=1}^{T}(\sum_{m=1}^{N} U_{nmt})\frac{\partial\alpha_{njt}}{\partial\mathbf{F}_{nj}}\mathbf{s}_t$$

(In fact you need to compute Hessian matrix ... Stop it...)

Similarly, (3) is

$$\sum_{t=1}^{T} V_{nt} \log \eta_{ny_t,t}$$

$$= \sum_{t=1}^{T} V_{nt} \log \left(\frac{\exp(\mathbf{G}_{ny_t} \mathbf{s}_t)}{\sum_{j=1}^{M} \exp(\mathbf{G}_{nj} \mathbf{s}_t)} \right)$$

$$= \sum_{t=1}^{T} V_{nt} \mathbf{G}_{ny_t} \mathbf{s}_t - \sum_{t=1}^{T} V_{nt} \log \left(\sum_{j=1}^{M} \exp(\mathbf{G}_{nj} \mathbf{s}_t) \right)$$

(M is the total number of output choices, i.e. num_out in the previous table)

The gradient with regard to $\mathbf{G}_{ni}$ is,

$$\frac{\partial(3)}{\partial \mathbf{G}_{ni}} = \sum_{t=1}^{T} V_{nt} \delta_{y_t,i} s_t - \sum_{t=1}^{T} V_{nt} \frac{\exp(\mathbf{G}_{ni} \mathbf{s}_t)}{\sum_{j=1}^{M} \exp(\mathbf{G}_{nj} \mathbf{s}_t)} \mathbf{s}_t$$

$$= \sum_{t=1}^{T} (\delta_{y_t,i} - \eta_{ni,t}) V_{nt} \mathbf{s}_t \quad (9)$$

$\delta_{y_t,i}$ is an identity function, i.e. = 0 if $y_t \neq i$, = 1, if $y_t = i$

Training with data from multiple trials

The ECLL is the sum of the trial-specific ECLLs:

$$\mathsf{ECLL} = < L(\theta | \mathbf{Y}^1, \mathbf{q}^1, \ldots, \mathbf{Y}^R, \mathbf{q}^R >_{\hat{p}(\mathbf{q}^1, \ldots, \mathbf{q}^R)}$$

$$= \sum_{r=1}^{R} \sum_{n=1}^{N} V_{r,n,0} \log \pi_n + \sum_{r=1}^{R} \sum_{t=1}^{T} \sum_{n=1}^{N} \sum_{m=1}^{N} U_{r,n,m,t} \log \alpha_{r,n,m,t} + \sum_{r=1}^{R} \sum_{t=1}^{T} \sum_{n=1}^{N} V_{r,n,t} \log \eta_{r,n,y_t^r,t}$$

The gradients with regard to filters are

$$\nabla \mathbf{F}_{ni} = \sum_{r=1}^{R} \sum_{t=1}^{T} U_{r,n,i,t} \mathbf{s}_t - \sum_{r=1}^{R} \sum_{t=1}^{T} U_{r,n,i,t} \delta_{ni} \mathbf{s}_t - \sum_{t=1}^{T} \sum_{m=1}^{N} U_{nmt} \alpha_{nit} \left(1 - \delta_{ni}\right) \mathbf{s}_t$$

Regularization

The objective function

$$H = \mathsf{ECLL} - \lambda_F \sum_{m,n,l} F_{mnl}^2 - \lambda_G \sum_{p,q,r} G_{p,q,r}^2$$

The gradients with regard to filters are

$$\nabla \mathbf{F}_{ni} = \sum_{r=1}^{R}\sum_{t=1}^{T} U_{r,n,i,t}\mathbf{s}_t^r - \sum_{r=1}^{R}\sum_{t=1}^{T} U_{r,n,i,t}\delta_{ni}\mathbf{s}_t^r - \sum_{r=1}^{R}\sum_{t=1}^{T}\sum_{m=1}^{N} U_{r,n,m,t}\alpha_{r,nit}\left(1-\delta_{ni}\right)\mathbf{s}_t^r \\ - 2\lambda_F \mathbf{F}_{ni}$$

$$\nabla \mathbf{G}_{mj} = \sum_{r=1}^{R}\sum_{t=0}^{T}\sum_{n=1}^{N} V_{r,n,t}\Big(\delta_{y_t{}^r,j}(1-\delta_{1,y_t{}^r}) - \eta_{r,m,j}(1-\delta_{1,j})\Big)\mathbf{s}_t^r \\ - 2\lambda_G \mathbf{G}_{mj}$$

Trouble shooting

Non converging ECLL for long time series / α almost 1 in diagonal elements

First try: get diagonal elements in trans_filter 0.

Now if $m=n$, $\alpha_{nmt} = \frac{1}{1+\sum_{l\neq n}\exp(\mathbf{F}_{nl}\mathbf{s}_t)}$, else, $\alpha_{nmt} = \frac{\exp(\mathbf{F}_{nm}\mathbf{s}_t)}{1+\sum_{l\neq n}\exp(\mathbf{F}_{nl}\mathbf{s}_t)}$

Use the Kronecker delta function, we get one expression:

$$\alpha_{nmt} = \frac{\exp\Big(\mathbf{F}_{nm}\mathbf{s}_t(1-\delta(m,n))\Big)}{\sum_l \exp\Big(\mathbf{F}_{nl}\mathbf{s}_t(1-\delta(l,n))\Big)}$$

The gradients of Eq.(2) turn into:

$$\begin{aligned}
&\frac{\partial}{\partial \mathbf{F}_{ni}}\Big(\sum_{t=1}^{T}\sum_{m=1}^{N}\hat{p}\left(q_{t-1}=n, q_t=m\right)\log\alpha_{n,m,t}\Big) \\
&= \sum_{t=1}^{T}\sum_{m=1}^{N} U_{nmt}\Big(\frac{\partial}{\partial \mathbf{F}_{ni}}(\mathbf{F}_{nm}\mathbf{s}_t(1-\delta(m,n))) - \frac{\partial}{\partial \mathbf{F}_{ni}}(\log(\sum_l \exp(\mathbf{F}_{nl}\mathbf{s}_t(1-\delta(l,n)))))\Big) \\
&= \sum_{t=1}^{T}\sum_{m=1}^{N} U_{nmt}\Big(\delta_{im}(1-\delta_{mn})\mathbf{s}_t - \frac{\exp(\mathbf{F}_{ni}\mathbf{s}_t(1-\delta(i,n)))\mathbf{s}_t(1-\delta(i,n))}{\sum_l \exp(\mathbf{F}_{nl}\mathbf{s}_t(1-\delta(l,n)))}\Big) \\
&= \sum_{t=1}^{T}\sum_{m=1}^{N} U_{nmt}\delta_{im}\mathbf{s}_t - \sum_{t=1}^{T}\sum_{m=1}^{N} U_{nmt}\delta_{im}\delta_{mn}\mathbf{s}_t - \sum_{t=1}^{T}\sum_{m=1}^{N} U_{nmt}\alpha_{nit}(1-\delta_{ni})\mathbf{s}_t
\end{aligned}$$

Applying the properties of Kronecker delta function,

we have

$$\frac{\partial}{\partial \mathbf{F}_{ni}}(2) = \sum_{t=1}^{T} U_{ni,t}\mathbf{s}_t - \sum_{t=1}^{T} U_{ni,t}\delta_{ni}\mathbf{s}_t - \sum_{t=1}^{T}\sum_{m=1}^{N} U_{nmt}\alpha_{nit}\left(1-\delta_{ni}\right)\mathbf{s}_t$$

If $n = i$, it's zero!

for other i, it's $\sum_{t=1}^{T} U_{ni,t}\mathbf{s}_t - \sum_{t=1}^{T}\sum_{m=1}^{N} U_{nmt}\alpha_{nit}\mathbf{s}_t$

After this, the trained α is close to the true one.Nice!

Let's do the similar thing on the lovely η, i.e. select one output as base, say, output 1,

then we get

$$\eta_{m,i,t} = \frac{\exp\Big(\mathbf{G}_{mi}\mathbf{s}_t(1-\delta_{1,i})\Big)}{\sum_l \exp\Big(\exp(\mathbf{G}_{ml}\mathbf{s}_t(1-\delta_{1,l}))\Big)}$$

The gradients of Eq.(3) turn into:

$$\frac{\partial}{\partial \mathbf{G}_{mj}}(3) = \sum_{t=0}^{T}\sum_{n=1}^{N} V_{nt}\Big(\delta_{y_t,j}(1-\delta_{1,y_t}) - \eta_{m,j}(1-\delta_{1,j})\Big)\mathbf{s}_t$$

Easily trapped in local minima

Train multiple times with different random initializations. And feed more data.

Not working : (