Summary of Hidden Markov Models (written by Peter Sarvari, based on slides of Dr. Stefanos Zafeiriou, Imperial College London)

Note: for the notation, see the document 'Derivations.pdf'.

<u>No scaling</u>

$$p(x_t|z_t) = b_{:,x_t}$$

Forward recursion

$$\alpha(z_t) = p(z_t, x_1, \dots, x_t)$$

$$\alpha(z_t) = b_{:,x_t} * \sum_{z_{t-1}} p(z_t|z_{t-1}) * \alpha(z_{t-1})$$

$$\alpha(z_t) = b_{:,x_t} .* \left(A' * \alpha(z_{t-1})\right)$$

$$\alpha(z_1) = p(z_1, x_1) = p(x_1|z_1) * p(z_1) = b_{:,x_1} .* \pi$$

Backward recursion

$$\beta(z_t) = p(x_{t+1}, \dots, x_T|z_t)$$

$$\beta(z_t) = \sum_{z_{t+1}} p(x_{t+1}|z_{t+1}) * p(z_{t+1}|z_t) * \beta(z_{t+1})$$

$$\beta(z_t) = A * (b_{:,x_{t+1}} .* \beta(z_{t+1}))$$

$$\beta(z_T) = p(z_T|x_1, \dots, x_T) ./ \alpha(z_T) = p(z_T|x_1, \dots, x_T) ./ (z_T|x_1, \dots, x_T) = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$$

Expectations: Smoothing and Smoothed transition

$$E_{p(Z|X)}\{z_{t,k}\} = \sum_k z_{t,k} * p\left(z_{t,k}|x_1, \dots, x_T\right) = p\left(z_{t,k}|x_1, \dots, x_T\right) = \frac{\alpha\left(z_{t,k}\right) * \beta\left(z_{t,k}\right)}{p(x_1, \dots, x_T)} = \frac{\alpha\left(z_{t,k}\right) * \beta\left(z_{t,k}\right)}{\sum_k \alpha\left(z_{T,k}\right)}$$

$$p(z_t|x_1, \dots, x_T) = \frac{\alpha(z_t) .* \beta(z_t)}{\sum_k \alpha\left(z_{T,k}\right)}$$

$$E_{p(Z|X)}\{z_{t-1,j}, z_{t,k}\} = \sum_j \sum_k z_{t-1,j} * z_{t,k} * p\left(z_{t-1,j}, z_{t,k}|x_1, \dots, x_T\right) = p\left(z_{t-1,j}, z_{t,k}|x_1, \dots, x_T\right) =$$

$$\frac{\alpha\left(z_{t-1,j}\right) * p(x_t|\, z_{t,k}) * p(z_{t,k}|z_{t-1,j}) * \beta\left(z_{t,k}\right)}{p(x_1, \dots, x_T)}$$

$$p(z_{t-1}, z_t|x_1, \dots, x_T) = \frac{A .* \left(\alpha\left(z_{t-1,j}\right) * \left(b_{:,x_t} .* \beta(z_t)\right)'\right)}{p(x_1, \dots, x_T)}$$

Scaling

$$c_m = p(x_m|x_1,\dots,x_{m-1}) = \sum\nolimits_{z_m} \alpha(z_m)$$

$$p(x_1,\dots,x_t) = p(x_1) * \prod_{m=2}^{t} c_m$$

Forward recursion

$$\hat{\alpha}(z_t) = p(z_t|x_1,\dots,x_t) = \frac{\alpha(z_t)}{\prod_{m=1}^{t} c_m}$$

$$c_t * \hat{\alpha}(z_t) = b_{:,x_t} * \sum\nolimits_{z_{t-1}} p(z_t|z_{t-1}) * \hat{\alpha}(z_{t-1})$$

$$c_t * \hat{\alpha}(z_t) = b_{:,x_t} .* \left(A' * \hat{\alpha}(z_{t-1})\right)$$

$$\alpha(z_1) = p(z_1|x_1) = \frac{p(x_1,z_1)}{p(x_1)} = \frac{\alpha(z_1)}{\sum_{z_1} \alpha(z_1)}$$

Backward recursion

$$\hat{\beta}(z_t) = \frac{\beta(z_t)}{\prod_{m=t+1}^{T} c_m} = \frac{p(x_{t+1},\dots,x_T|z_t)}{p(x_{t+1},\dots,x_T|x_1,\dots,x_t)}$$

$$c_{t+1} * \hat{\beta}(z_t) = \sum\nolimits_{z_{t+1}} b_{:,x_{t+1}} * p(z_{t+1}|z_t) * \hat{\beta}(z_{t+1})$$

$$c_{t+1} * \hat{\beta}(z_t) = A * \left(b_{:,x_{t+1}} .* \ \hat{\beta}(z_{t+1})\right)$$

$$\hat{\beta}(z_T) = \beta(z_T) = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$$

Expectations: Smoothing and smoothed transition

$$p(z_t|x_1,\dots,x_T) = \hat{\alpha}(z_t) .* \hat{\beta}(z_t)$$

$$p\left(z_{t-1,j}, z_{t,k}\middle|x_1,\dots,x_T\right) = c_t^{-1} * \hat{\alpha}\left(z_{t-1,j}\right) * p\left(x_t\middle|z_{t,k}\right) * p\left(z_{t,k}\middle|z_{t-1,j}\right) * \hat{\beta}\left(z_{t,k}\right)$$

$$p(z_{t-1}, z_t|x_1,\dots,x_T) = c_t^{-1} * A .* \left(\hat{\alpha}(z_{t-1}) * \left(b_{:,x_t} .* \hat{\beta}(z_t)\right)'\right)$$

Viterbi algorithm

The purpose of the algorithm is to find the sequence of states that are together the most probable given the observed data. The key idea is that the most probable path to state i is the most probable path to state j followed by a transition from j to i, where the probability of the transition depends on the transition matrix and the ith observation. As a result, it is impossible to tell the nth state in the most probable sequence of states until the recursion reached the last state. (For example, consider the following situation. One step before the termination, the most probable state in the T-1th place in the sequence seems to be $z_{T-1,3} = 1$. However, the last observation X_T turns out to be a value that can only be generated by the fourth hidden state. Also let's suppose that it is only possible to transition to the fourth hidden state from the first hidden state. Then we know that the one to last hidden state must be $Z_{T-1,1} = 1$.) Then we can be sure that the last state that maximized the probability is the last state in the most probable sequence. After that we need to do traceback to determine previous states in the most probable sequence. Mathematically, this looks the following way:

<u>Method I</u>

$$\delta_t(i) = \max_{z_1,\dots,z_{t-1}} p(z_1, \dots, z_{t-1}, z_{t,i} = 1 | x_1, \dots, x_t)$$

The recursion:

$$\delta_t(i) = c_t^{-1} * b_{i,x_t} * \max_j \delta_{t-1}(j) * a_{j,i}$$

$\alpha_t(i)$ stores the previous state in the most likely sequence given that the following state in the most likely sequence is i.

$$\alpha_t(i) = c_t^{-1} * b_{i,x_t} * arg \max_j \delta_{t-1}(j) * a_{j,i}$$

<u>Method II</u> (no need to use c_t – algorithm suggested in Murphy: Machine Learning)

$$\delta_t(i) = \max_{z_1,\dots,z_{t-1}} p(z_1, \dots, z_{t-1}, z_{t,i} = 1, x_1, \dots, x_t)$$

The recursion:

$$\delta_t(i) = \max_j \delta_{t-1}(j) * a_{j,i} * b_{i,x_t}$$

$$\alpha_t(i) = arg \max_j \delta_{t-1}(j) * a_{j,i} * b_{i,x_t}$$

The start of the recursion:

$$\delta_1(j) = b_{j,x_1} * \pi_j$$

Termination of recursion:

$$z_T{}^* = arg \max_i \delta_T(i)$$

Traceback:

$$z_{t-1}{}^* = \alpha_t(z_t{}^*)$$

For the vectorized implementation, see 'ViterbiDecode.m'.