02471 Machine Learning for Signal Processing

Solution

Exercise 10: State-space models – the Hidden Markov Model

10.1 Probabilities in HMM

Exercise 10.1.1

Use marginalization (sum rule)

$$
\begin{aligned}
P(y_1) &= \sum_{i=1}^{K} P(y_1|x_1 = i)P(x_1 = i) \\
P(y_1 = 2) &= \sum_{i=1}^{K} P(y_1 = 2|x_1 = i)P(x_1 = i) \\
&= P(y_1 = 2|x_1 = 1)P(x_1 = 1) + P(y_1 = 2|x_1 = 2)P(x_1 = 2) \\
&= 0.6 \cdot 0.7 + 0.2 \cdot 0.3 \\
&= 0.48
\end{aligned}
$$

Exercise 10.1.3

Use marginalization (sum rule)

$$
\begin{aligned}
P(y_1, y_2) &= \sum_{i=1}^{K} \sum_{j=1}^{K} P(y_1, y_2|x_1 = i, x_2 = j)P(x_1 = i, x_2 = j) \\
&= \sum_{i=1}^{K} \sum_{j=1}^{K} P(y_2|y_1, x_1 = i, x_2 = j)P(y_1|x_1 = i, x_2 = j)P(x_2 = j|x_1 = i)P(x_1 = i) \\
&= \sum_{i=1}^{K} \sum_{j=1}^{K} P(y_2|x_2 = j)P(y_1|x_1 = i)P(x_2 = j|x_1 = i)P(x_1 = i)
\end{aligned}
$$

where, in the last line, we have used the information from the HMM graph to remove unneeded conditionals.

This operation scales with K^2.

Exercise 10.1.5

$$
\begin{aligned}
P(y_2, y_1) &= \sum_{x_2} P(y_1, y_2, x_2) \\
P(y_1, y_2, x_2) &= \sum_{x_1} P(y_1, y_2, x_1, x_2) \\
&= \sum_{x_1} P(y_2|y_1, x_1, x_2) P(y_1, x_1, x_2) \\
&= \sum_{x_1} P(y_2|y_1, x_1, x_2) P(x_2|y_1, x_1) P(y_1, x_1)
\end{aligned}
$$

Removing the terms that are not needed for the conditionals, we get

$$
\begin{aligned}
P(y_1, y_2, x_2) &= \sum_{x_1} P(y_2|x_2) P(x_2|x_1) P(y_1, x_1) \\
&= P(y_2|x_2) \sum_{x_1} P(x_2|x_1) P(y_1, x_1)
\end{aligned}
$$

Exercise 10.1.6

Using $\alpha(x_n) := P(y_{[1:n]}, x_n)$ we can write

$$
\begin{aligned}
P(y_1, y_2, x_2) = \alpha(x_2) &= P(y_2|x_2) \sum_{x_1} P(x_2|x_1) P(y_1, x_1) \\
&= P(y_2|x_2) \sum_{x_1} P(x_2|x_1) \alpha(x_1)
\end{aligned}
$$

Notice that this case is general, i.e. we could have used n and $n-1$ as time index instead of 2 and 1, and all derivations are still correct. By that, we get the recursive formula:

$$
\alpha(x_n) = P(y_n|x_n) \sum_{x_{n-1}} P(x_n|x_{n-1}) \alpha(x_{n-1})
$$

All of this is derived for the case where y_i is a discrete random variable, but we did not use that property in the derivation, so the result also holds for (multivariate) continuous random variables. In that case, the capital $P(\cdot)$ is replaced with $p(\cdot)$ for the expressions that go over a distribution for $\mathbf{y}$.

This formula contains $K+1$ multiplications and K additions.

Exercise 10.1.7

We can exploit this result further using Bayes formula

$$
\begin{aligned}
P(X|Y) &= \frac{P(Y,X)}{P(Y)} \quad \Rightarrow \\
P(x_n|y_{[1:n]}) &= \frac{P(y_{[1:n]}, x_n)}{P(y_{[1:n]})} \\
&= \frac{\alpha(x_n)}{P(y_{[1:n]})}
\end{aligned}
$$

We get an even more efficient formula since, from the sum formula we have

$$
\begin{aligned}
P(y_{[1:n]}) &= \sum_{x_n} P(y_{[1:n]}, x_n) \\
&= \sum_{x_n} \alpha(x_n)
\end{aligned}
$$

Combining this yields

$$
P(x_n|y_{[1:n]}) = \frac{\alpha(x_n)}{\sum_{x_n} \alpha(x_n)}
$$

$P(y_{[1:n]})$ has K additions, so including the calculations for computing $\alpha(x_n)$, we get $K+n(2K+1) = \mathcal{O}(nK)$ operations, as opposed to the direct implementation that had $\mathcal{O}(K^N)$ operations.

10.2 HMM model formulation and EM updates

There are no explicit solutions for exercise 10.1.1–10.1.3. The book readily derives the expressions. We'll provide reading directions instead.

Exercise 10.2.1

Use the information from how the model parameters are setup (sec 16.5.1, page 847), and then derive eq. (16.32)–(16.35).

Exercise 10.2.2

This derivation is described in sec 16.5.2, page 852, eq (16.52).

Exercise 10.2.3

This derivation is described in sec 16.5.2, page 853, eq (16.53)–(16.55).

Exercise 10.2.4

The maximization step becomes

$$\begin{aligned}\mathcal{Q}\big(\Theta, \Theta^{(t)}\big) &= \sum_{k=1}^{K} \gamma\big(x_{1,k} = 1; \Theta^{(t)}\big) \ln P_k \\ &+ \sum_{n=2}^{N} \sum_{i=1}^{K} \sum_{j=1}^{K} \xi\big(x_{n-1,j} = 1, x_{n,i} = 1; \Theta^{(t)}\big) \ln P_{ij} \\ &+ \text{ constant}\end{aligned}$$

where the constant involves parameters independent of P_k, P_{ij}. Since P_k and P_{ij} are decoupled, they can be solved independently.

Since each row of the matrix containing P_{ij} is a discrete distribution, each row much sum to one. We have K states, hence we will have K rows and thus K constraints

$$\sum_{k=1}^{K} P_{kj} = 1, \qquad j = 1, \cdots, K$$

The Lagrangian then becomes

$$L(P_{ij}, \lambda) = \sum_{n=2}^{N} \sum_{i=1}^{K} \sum_{j=1}^{K} \xi\big(x_{n-1,j} = 1, x_{n,i} = 1; \Theta^{(t)}\big) \ln P_{ij} - \lambda \left(\sum_{k=1}^{K} P_{kj} - 1 \right)$$

Taking the derivative with respect to P_{ij} and equating to zero we get,

$$\frac{1}{\lambda} \sum_{n=2}^{N} \xi\big(x_{n-1,j} = 1, x_{n,i} = 1; \Theta^{(t)}\big) = P_{ij}$$

and plugging into the constraint, in order to compute λ, we obtain

$$\begin{aligned}\sum_{k=1}^{K} \frac{1}{\lambda} \sum_{n=2}^{N} \xi\big(x_{n-1,j} = 1, x_{n,k} = 1; \Theta^{(t)}\big) &= 1 \\ \Rightarrow \frac{1}{\sum_{n=2}^{N} \sum_{k=1}^{K} \xi(x_{n-1,j} = 1, x_{n,k} = 1; \Theta^{(t)})} &= \frac{1}{\lambda}\end{aligned}$$

Substituting $\frac{1}{\lambda}$ and adding the iteration index $(t+1)$ then yields

$$P_{ij}^{(t+1)} = \frac{\sum_{n=2}^{N} \xi\big(x_{n-1,j} = 1, x_{n,i} = 1; \Theta^{(t)}\big)}{\sum_{n=2}^{N} \sum_{k=1}^{K} \xi(x_{n-1,j} = 1, x_{n,k} = 1; \Theta^{(t)})}$$