Forecast and Simulation **Summer Term 2023**
http://ml3.leuphana.de/lectures/summer23/FS/
Machine Learning Group, Leuphana University of Lüneburg

Ulf Brefeld (brefeld@leuphana.de)
Yannick Rudolph (yannick.rudolph@leuphana.de)

Exercise 8

Due: Thursday, June 22, 2023

Task 1: Baum-Welch algorithm

This is a continuation of Task 1 from Exercise 2.

As in Task 1 from Exercise 2, Alice wants to reestimate her model of Bob's activities. She still believes that his activities are guided by two weather conditions, this time however she only has data for Bob's activities and no access to the corresponding weather data. You want to help her by implementing the Baum-Welch (or equivalently: EM) algorithm for HMM training and fit an HMM to the available data ... However, we have to adapt the vanilla Baum-Welch algorithm for multiple sequences first, since we have access to 1000 (independent) sequences (of same lenght) and want to train on all of them.

For exercise i–iii, assume we observe N sequences $\{\mathbf{x}^n\}_{n=1}^N$ of lenght T each. We further assume, that each latent variable can take one of K values. We thus represent each latent variable $\mathbf{z}_t^n$ with a K dimensional *binary vector* with components $\mathbf{z}_{tk}^n$, for $k = 1, ..., K$ and enforce that $\sum_k \mathbf{z}_{tk}^n = 1$, i.e. only one component equals 1 for each latent variable.

i) Show that in the E-step of the EM algorithm, we simply evaluate posterior probabilities for the latent variables by running the alpha and beta recursions independently for each of the sequences.

ii) With the notation following [Bishop, 2006], show that in the M-step, the initial probability and transition probability parameters can be re-estimated as

$$\pi_k = \frac{\sum_{n=1}^N \gamma(\mathbf{z}_{1k}^n)}{\sum_{n=1}^N \sum_{j=1}^K \gamma(\mathbf{z}_{1j}^n)} = \frac{\sum_{n=1}^N \gamma(\mathbf{z}_{1k}^n)}{N} \tag{1}$$

and

$$A_{jk} = \frac{\sum_{n=1}^N \sum_{t=2}^T \xi(\mathbf{z}_{(t-1)j}^n, \mathbf{z}_{tk}^n)}{\sum_{n=1}^N \sum_{l=1}^K \sum_{t=2}^T \xi(\mathbf{z}_{(t-1)j}^n, \mathbf{z}_{tl}^n)} = \frac{\sum_{n=1}^N \sum_{t=2}^T \xi(\mathbf{z}_{(t-1)j}^n, \mathbf{z}_{tk}^n)}{\sum_{n=1}^N \sum_{t=2}^T \gamma(\mathbf{z}_{(t-1)j}^n)} \tag{2}$$

respectively.

iii) Since our observations are discrete, emission probabilities are modeled by a categorical distribution. Analogously to our representation of the latent variables, we represent each output $\mathbf{x}_t^n$ with a D dimensional variable, with components $\mathbf{x}_{ti}^n$, for $i = 1, ..., D$, where D is the number of possible outcomes and we have the constraint that $\sum_i \mathbf{x}_{ti}^n = 1$, i.e. only one component equals 1. We thus have, that $p(\mathbf{x}_t^n | \mathbf{z}_t^n = k)$ is given by $p(\mathbf{x}_t^n | \mu_k) = \prod_i^D (\mu_{ki})^{\mathbf{x}_{ti}^n}$, where μ_k denote the parameters of the emission probability for a latent variable with active component k (i.e. the kth component equaling 1).

Show that the M-step maximization with respect to μ_k results in

$$\mu_{ki} = \frac{\sum_{n=1}^{N}\sum_{t=1}^{T}\gamma(\mathbf{z}_{tk}^n)\mathbf{x}_{ti}^n}{\sum_{n=1}^{N}\sum_{t=1}^{T}\gamma(\mathbf{z}_{tk}^n)}. \tag{3}$$

Hint: we basically solved part i) in Exercise 7.

Hint 2: for ii) and iii) start with

$$\begin{aligned} Q(\theta, \theta^{\text{old}}) =& \mathbb{E}\mathbf{z}[\ln p(\mathbf{x}, \mathbf{z}|\theta)] && (4)\\ =& \sum_{n=1}^{N}\sum_{k=1}^{K}\gamma(\mathbf{z}_{1k}^n)\ln\pi_k + \sum_{n=1}^{N}\sum_{t=2}^{T}\sum_{j=1}^{K}\sum_{k=1}^{K}\xi(\mathbf{z}_{(t-1)j}^n, \mathbf{z}_{tk}^n)\ln A_{jk} && (5)\\ &+ \sum_{n=1}^{N}\sum_{t=1}^{T}\sum_{k=1}^{K}\gamma(\mathbf{z}_{tk}^n)\ln p(\mathbf{x}_t^n|\mu_k), && (6) \end{aligned}$$

with $\theta = \{\pi, A, \mu\}$ and maximize the expression with appropriate Lagrangian multipliers to take into account the summation constraints.

iv) *Bonus Task: Derive the equation provided in Hint 2.*

Answer

i) First of all, note that for every observed variable there is a corresponding latent variable, and so for every sequence x^n of observed variables there is a corresponding sequence z^n of latent variables. The sequences are assumed to be independent given the model parameters, and so the joint distribution of all latent and observed variables will be given by

$$p(\mathbf{x}, \mathbf{z}|\theta) = \prod_{n=1}^{N} p(\mathbf{x}^n, \mathbf{z}^n|\theta) \tag{7}$$

where $\mathbf{x}$ denotes $\{\mathbf{x}^n\}$ and $\mathbf{z}$ denotes $\{\mathbf{z}^n\}$.

Given our solution to Exercise 7, Task 2, v), we know, that the posterior distribution factorizes over the N, i.e.

$$p(\mathbf{z}|\mathbf{x}, \theta) = \prod_{n=1}^{N} p(\mathbf{z}^n|\mathbf{x}^n, \theta). \tag{8}$$

Thus the evaluation of the posterior distribution of the latent variables, corresponding to the E-step of the EM algorithm, can be done independently for each of the sequences (using the standard alpha-beta recursions).

ii–iv) See classnotes.

Task 2: Baum-Welch algorithm (continues)

This is a continuation of Task 1 and also relates to Task 1 of Exercise 1.

i) Implement the Baum-Welch algorithm for Task 1 (taking the data from *exercise-02-data.npz*).

ii) As in Task 1 of Exercise 1, apply the Viterbi algorithm to find the most probable sequence of states $z = (z_1, z_2, \ldots, z_5)$ given the following sequence of activities reported by Bob $x = (\mathsf{Walk}, \mathsf{Shop}, \mathsf{Clean}, \mathsf{Shop}, \mathsf{Walk})$, and using the HMM learned with the Baum-Welch algorithm. Try to provide an interpretation for the two possible states.

iii) Compare your learned HMM to the model provided in Task 1 of Exercise 1.

Answer

Postponed till next exercise.

Task 3: Revisiting the forward-backward algorithm

This Task relates to Task 2 of Exercise 2.

i) You might have re-used the forward-backward algorithm that we have implemented in Task 2 of Exercise 2. In our implementation of the algoithm, we have used the *log-sum-exp trick* to prevent numerical problems. Both [Rabiner, 1989] (Section V.A.) and [Bishop, 2006] (Section 13.2.4) rather suggest the use of *scaling factors*. What might be the advantage of scaling compared to the *log-sum-exp trick*?

ii) *Bonus Task: Re-implement the forward-backward algorithm with scaling factors.*

Answer

Postponed till next exercise.

References

[Bishop, 2006] Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.

[Rabiner, 1989] Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.