

Suggested solution for Assignment 4

Teaching assistant: Wilson Tam (yct@cs.cmu.edu)

Due on:

Instructions: Problem 1 and Problem 2 are compulsory, while you can choose either Problem 3 or Problem 4.

Problem 1: Expectation Maximization

The expectation-maximization (EM) algorithm is a technique for maximum likelihood estimation involving hidden variables. The aim of this problem is to help you understand how the optimization function is established in the context of a simple Gaussian mixture model (GMM) with K mixtures. Assume the GMM generates the observation vectors $X=x_1, x_2, \dots, x_T$ from $t=1$ to $t=T$ independently. However, we do not know which Gaussian mixture was picked to generate the observation at time t . In other words, the hidden variables here are the mixture indices (1.. K) to generate the observation at each time t . Denote the mixture index at time t as q_t . Therefore, the hidden mixture sequence is $Q=q_1, q_2, \dots, q_T$.

For your information, a multivariate Gaussian distribution has the following form:

$$p(x; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^n \cdot |\Sigma|}} \cdot e^{-\frac{1}{2}(x-\mu)^t \Sigma^{-1}(x-\mu)} \quad (1)$$

where μ and Σ denote the mean and the covariance parameters respectively, and n denotes the feature dimension.

1. Derive the log likelihood of X when Q is observed i.e. $\log p(X, Q)$?

Solution:

$$\log p(X, Q) = \sum_{t=1}^T \log p(x_t | q_t) + \log p(q_t) \quad (2)$$

$$= \sum_{t=1}^T \left(-\frac{1}{2}(x_t - \mu_{q_t})^t \Sigma_{q_t}^{-1}(x_t - \mu_{q_t}) - \frac{1}{2} \log |\Sigma_{q_t}| + \log \lambda_{q_t} \right) + Constant \quad (3)$$

2. Hence, write down $\log p(X)$ in terms of $p(X, Q)$?

Solution:

$$\log p(X) = \log \sum_Q p(X, Q) \quad (4)$$

$$= \log \sum_{q_1 q_2 \dots q_T} \prod_{t=1}^T p(x_t | q_t) \cdot p(q_t) \quad (5)$$

$$= \log \prod_{t=1}^T \sum_{k=1}^K \lambda_k \cdot p(x_t | k) \quad (6)$$

$$= \sum_{t=1}^T \log \sum_{k=1}^K \lambda_k \cdot p(x_t|k) \quad (7)$$

3. Using the Jensen's inequality $\log \sum_y p(y) f(y) \geq \sum_y p(y) \log f(y)$ where $p(y)$ is a probability distribution, and the posterior distribution $P^{(j-1)}(q_t|X)$ computed using an old model from the (j-1)th training iteration, show that the lower-bound of $\log p(X)$ denoted as $Q(X; \{\lambda_k, \mu_k, \Sigma_k\})$, known as the EM auxiliary function, is:

$$Q(X; \{\lambda_k, \mu_k, \Sigma_k\}) = \sum_{t=1}^T \sum_{k=1}^K P^{(j-1)}(q_t = k|X) \left(-\frac{1}{2} (x_t - \mu_k)^t \Sigma_k^{-1} (x_t - \mu_k) - \frac{1}{2} \log |\Sigma_k| + \log \lambda_k \right) + \text{Constant}$$

Solution: We introduce the posterior probability $P^{(j-1)}(q_t = k|X)$ into Eqn 7 and apply Jensen's inequality:

$$\begin{aligned} \sum_{t=1}^T \log \sum_{k=1}^K P^{(j-1)}(q_t = k|X) \left(\frac{\lambda_k \cdot p(x_t|k)}{P^{(j-1)}(q_t = k|X)} \right) &\geq \sum_{t=1}^T \sum_{k=1}^K P^{(j-1)}(q_t = k|X) \log \left(\frac{\lambda_k \cdot p(x_t|k)}{P^{(j-1)}(q_t = k|X)} \right) \\ &= \sum_{t=1}^T \sum_{k=1}^K P^{(j-1)}(q_t = k|X) (\log p(x_t|k) + \log \lambda_k) + \text{constant} \end{aligned}$$

4. Hence, compute the derivative of $Q(X; \{\mu_k\})$ with respect to the mean vector of the k-th Gaussian. Show that the reestimation formula for μ_k is:

$$\hat{\mu}_k^{(j)} = \frac{\sum_{t=1}^T P^{(j-1)}(q_t = k|X) \cdot x_t}{T_k}$$

where $T_k = \sum_{t=1}^T P^{(j-1)}(q_t = k|X)$ and $\sum_{k=1}^K T_k = T$

How does the reestimation formula compare with the MLE formula with a single Gaussian (i.e. without hidden variables)?

Solution:

The reestimation formula for GMM is very similar to estimating a single Gaussian but each sample x_t is weighted by the posterior probability over the mixture index k . In other words, fractional counts are used to softly assign x_t to different Gaussian mixtures.

Problem 2: Acoustic Adaptation

A multivariate Gaussian distribution is given with known mean and covariance parameters. Given the adaptation data $X = \{x_1, x_2, \dots, x_T\}$, we want to find a transformation matrix $A_{n \times n}$ such that the likelihood of the *transformed* adaptation data is maximized. An input vector x can be transformed linearly as $y = Ax + b$ where b is a bias vector.

1. Derive the log likelihood of the *transformed* adaptation data.

Solution:

Since a transformed vector is computed as $y = Ax + b$, the transformed log likelihood of the adaptation is:

$$L(A) = \log p(Y) = \sum_{t=1}^T \log p(y_t) \quad (8)$$

$$= \sum_{t=1}^T (Ax_t + b - \mu)^t \Sigma^{-1} (Ax_t + b - \mu) + \text{Constant} \quad (9)$$

2. Derive the maximum likelihood estimation of A via matrix differentiation.

$$\frac{\partial L(A)}{\partial A} = \sum_{t=1}^T x_t (Ax_t + b - \mu)^t \Sigma^{-1} = 0 \quad (10)$$

$$\Rightarrow \left(\sum_{t=1}^T x_t x_t^t \right) A^t = \sum_{t=1}^T x_t (\mu - b)^t \quad (11)$$

$$\Rightarrow V A^t = Z \quad (12)$$

$$\Rightarrow A = Z^t V^{-1} \quad (13)$$

where $Z = \sum_{t=1}^T x_t (\mu - b)^t$ and $V = \sum_{t=1}^T x_t x_t^t$ and V is symmetric and is assumed invertible.

3. Is transformation of the feature vector identical to transformation of the mean vector μ in a single Gaussian distribution? Explain briefly.

Solution:

For a single Gaussian, μ and x are symmetric in terms of likelihood. Therefore, transforming the mean vector or transforming the feature vector should result in the same optimal likelihood. However, transforming a mean vector and transforming a feature vector generally produce different results in terms of likelihood.

4. A GMM is applied with K mixtures. Using the results from Problem 1, write down the EM auxiliary function involving A, i.e. Q(X;A).

Solution:

$$Q(X; A) = \sum_{t=1}^T \sum_{k=1}^K \gamma_{tk} \left(-\frac{1}{2} (Ax_t + b - \mu_k)^t \Sigma_k^{-1} (Ax_t + b - \mu_k) \right) + \text{constant} \quad (14)$$

where $\gamma_{tk} = P^{(j-1)}(q_t = k | X)$.

5. By differentiation, write down the MLE condition for A.

Solution:

$$\frac{\partial Q(X; A)}{\partial A} = \sum_{t=1}^T \sum_{k=1}^K \gamma_{tk} x_t (Ax_t + b - \mu_k)^t \Sigma_k^{-1} = 0 \quad (15)$$

$$\Rightarrow \sum_{t=1}^T \sum_{k=1}^K \gamma_{tk} x_t x_t^t A^t \Sigma_k^{-1} = \sum_{t=1}^T \sum_{k=1}^K \gamma_{tk} x_t (\mu_k - b)^t \Sigma_k^{-1} \quad (16)$$

$$\Rightarrow \sum_{t=1}^T \sum_{k=1}^K \gamma_{tk} \Sigma_k^{-1} A x_t x_t^t = \left(\sum_{t=1}^T \sum_{k=1}^K \gamma_{tk} x_t (\mu_k - b)^t \Sigma_k^{-1} \right)^t \quad (17)$$

$$\Rightarrow \sum_{k=1}^K \Sigma_k^{-1} A V_k = Z^t \quad (18)$$

where $V_k = \sum_{t=1}^T \gamma_{tk} x_t x_t^t$ and is symmetric.

Bonus: If a diagonal covariance of each Gaussian mixture is assumed, derive a reestimation formula for A in closed form.

Solution:

If diagonal covariance is assumed, then we can solve A row by row. Therefore, the q-th row of Eqn 18 is:

$$A_q \sum_{k=1}^K \Sigma_{qq}^{-1} V_k = Z_q^t \quad (19)$$

$$\Rightarrow A_q = Z_q^t G_q^{-1} \quad (20)$$

where $G_q = \sum_{k=1}^K \Sigma_{qq}^{-1} V_k$ and is assumed invertible.

Problem 3: Discriminative training

1. In acoustic modeling using HMM, explain briefly why discriminative training is usually desirable?

Solution:

Since HMM is not a correct model for acoustic speech, the MLE of HMM would be biased. In other words, maximizing the training likelihood does not correspond to minimizing the recognition errors. Therefore, discriminative training is usually desirable.

2. Does discriminative training always help? Give a scenario when discriminative training may hurt.

Solution:

No, especially when there is a mismatch between the training and test data, or the amount of training data is small leading to overfitting problem.

3. The maximum mutual information (MMI) criterion can be expressed as follows:

$$I(X, W) = \log \frac{p(X|W) \cdot P(W)}{\sum_{W' \neq W} p(X|W') \cdot P(W')} \quad (21)$$

Will you employ a (strong) trigram LM or a (weak) unigram LM for P(W)? Explain briefly your choice.

Solution:

A (weak) unigram LM is preferred so that more confusable hypotheses can be used for discriminative training.

4. In MMIE for continuous speech recognition, explain why the competing classes are usually represented either by a word lattice or a N-best list, but not a 1-best competitor.

Solution:

Since the denominator term of MMIE consists of other competing classes, using a 1-best competitor is not enough and thus either N-best list or word lattice should be employed for MMIE.

5. Assume a two-class $\{\omega_1, \omega_2\}$ classification problem with each class modeled by a Gaussian distribution. As a continuation of problem 2, write down the MMI criterion involving the feature transformation matrix A assuming the observation vectors X come from class ω_1 . You can ignore the term P(W) without the loss of generality.

Solution:

$$D(X; A) = \frac{1}{2} \sum_{t=1}^T (Ax_t + b - \mu_2) \Sigma_2^{-1} (Ax_t + b - \mu_2) - \frac{1}{2} \sum_{t=1}^T (Ax_t + b - \mu_1) \Sigma_2^{-1} (Ax_t + b - \mu_1) + \text{constant}$$

6. Compute the derivative of the above MMI criterion with respect to A. Can you obtain a closed-form solution for A?

$$\begin{aligned} \frac{\partial D(X; A)}{\partial A} &= \sum_{t=1}^T x_t (Ax_t + b - \mu_2)^t \Sigma_2^{-1} - \sum_{t=1}^T x_t (Ax_t + b - \mu_1)^t \Sigma_1^{-1} = 0 \\ \Rightarrow \left(\sum_{t=1}^T x_t x_t^t \right) A^t (\Sigma_2^{-1} - \Sigma_1^{-1}) &= \sum_{t=1}^T x_t (\mu_2 - b)^t \Sigma_2^{-1} - \sum_{t=1}^T x_t (\mu_1 - b)^t \Sigma_1^{-1} \\ \Rightarrow BA^t C &= Z \end{aligned}$$

Closed-form solution can be obtained if B and C are invertible.

Problem 4: Search

1. What is the difference between synchronous (e.g. Viterbi) and asynchronous search (e.g. A^*).

Solution:

In Viterbi search, all possible hypotheses with the same length (in terms of number of frames processed) are expanded, while in A^* search only the most promising hypothesis is considered for expansion. Viterbi search doesn't need a heuristic function to estimate the score of the remaining frames while A^* search does.

2. What is inadmissible? How does it happen in A^* search?

Solution:

The search solution is inadmissible when it is suboptimal which can happen in A^* search when the heuristic function overestimates the cost of the remaining frames.

3. Programming exercise: Implement A^* search to extract 3-best hypotheses from a word lattice. Download the lattice file from the course website. Report each extracted word hypothesis with a total score. Explain your choice of the heuristic function $h(\cdot)$. The word lattice has the following format:

```
StartNode EndNode Info(a=acoustic score, l=lmscore, wp=word penalty) Word CombinedScore
.. .. .. ..
FinalNode
..
node t=timestamp
.. ..
```

Each word and the score (in log10) is associated to an edge in the word lattice.

Solution:

Since the word lattice contains all the scores of any subpaths to the ending state, you can compute heuristic function $h(\cdot)$ exactly by running a backward algorithm to get the score from any word node in the lattice to the ending state.