Structural Bayesian Linear Regression for Hidden Markov Models

Watanabe, S.; Nakamura, A.; Juang, B-H

TR2013-071 August 2013

Abstract

Linear regression for Hidden Markov Model (HMM) parameters is widely used for the adaptive training of time series pattern analysis especially for speech processing. The regression parameters are usually shared among sets of Gaussians in HMMs where the Gaussian clusters are represented by a tree. This paper realizes a fully Bayesian treatment of linear regression for HMMs considering this regression tree structure by using variational techniques. This paper analytically derives the variational lower bound of the marginalized log-likelihood of the linear regression. By using the variational lower bound as an objective function, we can algorithmically optimize the tree structure and hyper parameters of the linear regression rather than heuristically tweaking them as tuning parameters. Experiments on large vocabulary continuous speech recognition confirm the generalizability of the proposed approach, especially when the amount of adaptation data is limited.

Journal of Signal Processing Systems

Journal of Signal Processing Systems manuscript No.
(will be inserted by the editor)

STRUCTURAL BAYESIAN LINEAR REGRESSION FOR HIDDEN MARKOV MODELS

Shinji Watanabe · Atsushi Nakamura · Biing-Hwang (Fred) Juang

Received: date / Accepted: date

Abstract Linear regression for Hidden Markov Model (HMM) parameters is widely used for the adaptive training of time series pattern analysis especially for speech processing. The regression parameters are usually shared among sets of Gaussians in HMMs where the Gaussian clusters are represented by a tree. This paper realizes a fully Bayesian treatment of linear regression for HMMs considering this regression tree structure by using variational techniques. This paper analytically derives the variational lower bound of the marginalized log-likelihood of the linear regression. By using the variational lower bound as an objective function, we can algorithmically optimize the tree structure and hyperparameters of the linear regression rather than heuristically tweaking them as tuning parameters. Experiments on large vocabulary continuous speech recognition confirm the generalizability of the proposed approach, especially when the amount of adaptation data is limited.

Keywords Hidden Markov Model · Linear Regression · Variational Bayes · Structural Prior

Shinji Watanabe
Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA, USA
E-mail: watanabe@merl.com
The work was mostly done while Shinji Watanabe was working at NTT Communication Science Laboratories.

Atsushi Nakamura
NTT Communication Science Laboratories, NTT Corporation, Kyoto, Japan
E-mail: nakamura.atsushi@lab.ntt.co.jp

Biing-Hwang (Fred) Juang
Center for Signal and Image Processing, Georgia Institute of Technology, Atlanta, GA, USA
E-mail: juang@ece.gatech.edu

1 Introduction

Hidden Markov Models (HMM) have been widely used for time series analysis (e.g., speech, text, and image processing). HMM parameters can be estimated by statistical methods, effectiveness of which depends on the quality and quantity of available data that should distribute according to the statistical feature of intended signal space or conditions. As there is no sure way of collecting sufficient data to cover all conditions, adaptive training of HMM parameters from a set of previously obtained parameters to a new set that befits a specific environment with a small amount of new data is an important research issue.

In speech recognition, one approach is to view the adaptation of model parameters to new data as a transformation problem; that is, the new set of model parameters is a transformed version of the old set: $\lambda_{n+1} = f(\lambda_n, \{x\}_n)$, where $\{x\}_n$ denotes the new set of data available at moment n for the existing model parameters λ_n to adapt to. Most frequently and practically, the function f is chosen to be of an affine transformation type [1, 2]: $\boldsymbol{\lambda}_{n+1} = \mathbf{A}\boldsymbol{\lambda}_n + \mathbf{b}$, when various parts of the model parameters, e.g., the mean vectors or the variances, are envisaged in a vector space. The adaptation algorithm therefore involves deriving the affine map components, $\mathbf{A}$ and $\mathbf{b}$, from the adaptation data $\{x\}_n$. A number of algorithms have been proposed for this purpose. (See [3, 4] for detail. There are many variants of transformation types for HMMs, e.g., [5–9]). Some techniques bear the name ”linear regression”, and our paper also uses this name by convention. In addition to speech recognition, there are many other applications of the adaptive training of HMMs than speech recognition (e.g., speech synthesis [10], speaker recog-

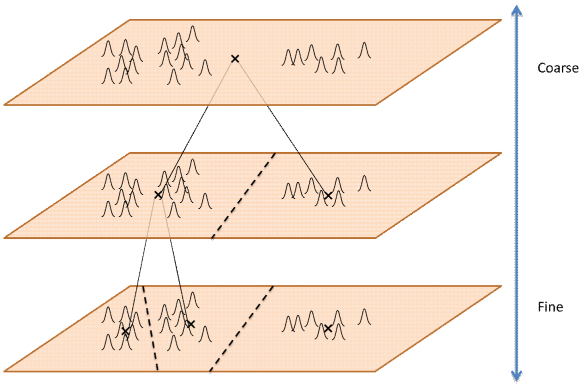

Fig. 1 Gaussian tree representation of liner regression parameters.

nition [11], face recognition [12] and activity recognition [13]).

The linear regression method for HMM parameters estimates the affine transformation parameters from a set of adaptation data, usually limited in size. The transformation with the estimated parameters is then applied to the previously trained HMMs, resulting in the set of "adapted models". Note that for automatic speech recognition, the number of the Gaussian distributions or simply Gaussians, which are used as component distributions in forming state-dependent mixture distributions, is typically in the thousands or more. If each mean vector in the set of Gaussians is to be modified by a unique transformation matrix, the number of "adaptation parameters" can therefore be quite large. The main problem of this method is thus how to improve "generalization capability" by avoiding the overtraining problem when the amount of adaptation data is small. To solve the problem, there are mainly two approaches: 1) model selection and 2) prior knowledge utilization.

The model selection approach is originally proposed within the estimation of linear transformation parameters by using the maximum likelihood EM algorithm (called Maximum Likelihood Linear Regression (MLLR)). MLLR proposes to share one linear transformation in a cluster of many Gaussians in the HMM set, thereby effectively reducing the number of free parameters that can then be trained with a small amount of adaptation data. The Gaussian clusters are usually constructed as a tree structure, as shown in Figure 1, which is predetermined and fixed throughout adaptation. This tree (called regression tree) is constructed based on a centroid splitting algorithm, described in [14]. This algorithm first makes two centroid vectors from a random perturbation of the global mean vector computed from Gaussians assigned to a target leaf node. Then, it splits a set of these Gaussians according to the Euclidean distance between Gaussian mean vectors and two centroid vectors. Obtained two sets of Gaussians are assigned to child nodes, and this procedure is continued to finally build a tree.

The utility of the tree structure is commensurate with the amount of adaptation data; namely, if we have a small amount of data, it uses only coarse clusters (e.g., the root node of a tree in the top layer of Figure 1) where the number of free parameters in the linear transformation matrices is small. On the other hand, if we have a sufficiently large amount of data, it can use fine clusters where the number of free parameters in the linear transformation matrices is large, potentially improving the precision of the estimated parameters. This framework needs to select appropriate Gaussian clusters according to the amount of data, i.e., it needs an appropriate model selection function. Usually, the model selection is performed by setting a threshold value manually (e.g., the total number of speech frames assigned to a set of Gaussians in a node)

While the regression tree in MLLR can be considered one form of prior knowledge, i.e., how various Gaussian distributions are related, another approach is to explicitly construct and use prior knowledge of regression parameters in an approximated Bayesian paradigm. For example, Maximum A Posteriori Linear Regression (MAPLR) [15] and quasi-Bayes linear regression [16] replace the ML criterion with the MAP and quasi-Bayes criteria, respectively, in the estimation of regression parameters. With the explicit prior knowledge acting as a regularization term, MAPLR appears to be less susceptible to the problem of over-fitting. The MAPLR is extended to the structural MAP (SMAP) [17] and the structural MAPLR (SMAPLR) [18], both of which fully utilize the Gaussian tree structure used in the model selection approach to efficiently set the hyper-parameters in prior distributions. In SMAP and SMAPLR, the hyperparameters in the prior distribution in a target node are obtained by the statistics in its parent node. Since the total number of speech frames assigned to a set of Gaussians in the parent node is always larger than that in the target node, the obtained statistics in the parent node is more reliable than that in the target node, and these can be good prior knowledge for transformation parameter estimation in the target node. Another extension of MAPLR is to replace MAP approximation to a fully Bayesian treatment of latent models, called variational Bayes (VB). VB has been developed in the machine learning field based on a variational technique [19–23], and has been successfully applied to HMM training in speech recognition [24–31]. VB is also applied to the estimation of the linear transformation parameters of HMMs [32, 33] to achieve further generalization capabilities.

This paper also employs VB for the linear regression problem, but we focus on the model selection and efficient prior utilization at the same time, in addition to the estimation of the linear transformation parameters of HMMs proposed in previous work [32, 33]. In particular, we consistently use the variational lower bound as the optimization criterion for the model structure and hyper-parameters, in addition to the posterior distributions of the transformation parameters and the latent variables[1]. Since this optimization leads the approximated variational posterior distributions to the true posterior distributions theoretically in the sense of minimizing Kullback Leibler divergence between them, the above consistent approach yields to improve the generalization capability [20, 22, 23]. To do this, this paper provides an analytical solution to the variational lower bound by marginalizing all possible transformation parameters and latent variables introduced in the linear regression problem. The solution is based on a variance-normalized representation of Gaussian mean vectors to simplify the solution as normalized domain MLLR. As a result of variational calculation, we can marginalize the transformation parameters in all nodes used in the structural prior setting. This is a part of the solution of the variational message passing algorithm [34], which is a general framework of variational inference in a graphical model. Furthermore, the optimization of the model topology and hyper-parameters in the proposed approach yields an additional benefit to the improvement of the generalization capability. For example, the proposed approach infers the linear regression without controlling the Gaussian cluster topology and hyper-parameters as the tuning parameters. Thus linear regression for HMM parameters is accomplished without excessive parameterization in a Bayesian sense.

This paper is organized as follows. It first introduces the conventional MLLR framework in Section 2. Then, we provide a formulation of the Bayesian linear regression framework in Section 3. Based on the formulation, Section 4 introduces a practical model selection and hyper-parameter optimization scheme in terms of optimizing the variational lower bound. Section 5 reports unsupervised speaker adaptation experiments for a large vocabulary continuous speech recognition task, and confirms the effectiveness of the proposed approach. The mathematical notations used in this paper are summarized in Table 1.

2 Linear regression for hidden Markov models based on variance normalized representation

This section briefly explains a solution for the linear regression parameters for HMMs within a maximum likelihood EM algorithm framework. This paper uses a solution based on a variance normalized representation of Gaussian mean vectors to simplify the solution[2]. In this paper, we only focus on the transformation of Gaussian mean vectors in HMMs.

2.1 Maximum likelihood solution based on EM algorithm and variance normalized representation

First, we explain the basic EM algorithm of the conventional HMM parameter estimation to set the notational convention and to align with the subsequent development of the proposed approach. Let $\mathbf{O} \triangleq \{\mathbf{o}_t \in \mathbb{R}^D | t = 1, \cdots, T\}$ be a sequence of D dimensional feature vectors for T speech frames. The latent variables in a continuous density HMM are composed of HMM states and mixture components of GMMs. A sequence of HMM states is represented by $\mathbf{S} \triangleq \{s_t | t = 1, \cdots, T\}$, where the value of s_t denotes an HMM state index at frame t. Similarly, a sequence of mixture components is represented by $\mathbf{Z} \triangleq \{z_t | t = 1, \cdots, T\}$, where the value of z_t denotes a mixture component index at frame t. The EM algorithm deals with the following auxiliary function as an optimization function instead of directly using the model likelihood:

$$Q(\boldsymbol{\Theta}; \hat{\boldsymbol{\Theta}}) \triangleq \langle \log p(\mathbf{O}, \mathbf{S}, \mathbf{Z} | \boldsymbol{\Theta}) \rangle_{p(\mathbf{S}, \mathbf{Z} | \mathbf{O}; \hat{\boldsymbol{\Theta}})} , \quad (1)$$

where $\boldsymbol{\Theta}$ is a set of HMM parameters. The brackets $\langle\rangle$ denote the expectation i.e. $\langle g(y) \rangle_{p(y)} \equiv \int g(y) p(y) dy$ for a continuous random variable y and $\langle g(n) \rangle_{p(n)} \equiv \sum_n g(n) p(n)$ for a discrete random variable n. $p(\mathbf{O}, \mathbf{S}, \mathbf{Z} | \boldsymbol{\Theta})$ is a complete data likelihood given $\boldsymbol{\Theta}$. $p(\mathbf{S}, \mathbf{Z} | \mathbf{O}; \hat{\boldsymbol{\Theta}})$ is the posterior distribution of the latent variables given

[1] Strictly speaking, since transformation parameters are not observables and are marginalized in this paper, these can be regarded as latent variables in a broad sense, similar to HMM states and mixture components of Gaussian Mixture Models (GMMs). However, these have different properties, e.g., transformation parameters can be integrated out in the VB-M step, while HMM states and mixture components are computed in the VB-E step, as discussed in Section 3. Therefore, to distinguish transformation parameters from HMM states and mixture components clearly, this paper only treats HMM states and mixture components as latent variables, which follows a terminology in variational Bayes framework [22]

[2] This is first described in [35] as normalized domain MLLR. The structural Bayes approach [17] for bias vector estimation in HMM adaptation also uses this normalized representation.

Table 1 Notation list

t	:	Speech frame index
T	:	The number of speech frames
$\mathbf{o}_t \in \mathbb{R}^D$	:	D dimensional feature vector at t
$\mathbf{O} = \{\mathbf{o}_t \vert t = 1, \cdots, T\}$	:	Sequence of feature vectors for T frames
$\mathbf{S} = \{s_t \vert t = 1, \cdots, T\}$	:	Sequence of HMM states for T frames
$\mathbf{Z} = \{z_t \vert t = 1, \cdots, T\}$	:	Sequence of mixture components in a GMM for T frames
$\mathbf{V} = \{\{s_t, z_t\} \vert t = 1, \cdots, T\}$	:	Joint event sequence of s and v
$Q(\cdot;\cdot)$	:	Auxiliary function used in the EM algorithm
$\boldsymbol{\Theta}$	:	Set of HMM parameters
m	:	Model structure index of a pruned Gaussian tree
$\mathcal{J}_m$	:	Set of leaf nodes with m
j	:	leaf node index
$\mathbf{W}_j \in \mathbb{R}^{D \times (D+1)}$	:	Regression matrix at j
$\boldsymbol{\Lambda}_{\mathcal{J}_m} = \{W_j \vert j = 1, \cdots, \vert\mathcal{J}_m\vert\}$	:	Subset of regression matrices for leaf nodes with m
i	:	node index
$\mathcal{I}_m$	:	Set of nodes with m
$\boldsymbol{\Lambda}_{\mathcal{I}_m} = \{W_i \vert i = 1, \cdots, \vert\mathcal{I}_m\vert\}$	:	Subset of regression matrices for nodes with m
$\mathsf{p}(i)$	:	Parent node of i
$\mathsf{l}(i)$	:	Left child node of i
$\mathsf{r}(i)$	:	Right child node of i
k	:	mixture component index for all Gaussians
$\zeta_{k,t} \in [0, 1]$	:	Posterior probability of mixture component k at t
$\boldsymbol{\mu}_k \in \mathbb{R}^D$	:	Gaussian mean vector at k
$\boldsymbol{\Sigma}_k \in \mathbb{R}^{D \times D}$	:	Gaussian covariance matrix at k
$\mathcal{N}(\cdot \vert \boldsymbol{\mu}, \boldsymbol{\Sigma})$	:	Gaussian distribution with $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$
$\boldsymbol{\mu}_k^{ad}$	:	Transformed mean vector
$\mathbf{C}_k \in \mathbb{R}^{D \times D}$	:	Cholesky decomposition matrix of $\boldsymbol{\Sigma}_k$
$\boldsymbol{\xi}_k \in \mathbb{R}^{D+1}$	:	Augmented normalized vector at k
$\mathcal{K}_i$	:	Set of Gaussians in node i
$\boldsymbol{\Xi}_j \in \mathbb{R}^{(D+1) \times (D+1)}$	:	0th order sufficient statistics for $\mathbf{W}_j$
$\mathbf{Z}_j \in \mathbb{R}^{D \times D}$	:	1st order sufficient statistics for $\mathbf{W}_j$
$\zeta_k \in \mathbb{R}_{>0}$	:	0th order sufficient statistics for kth Gaussian
$\boldsymbol{\nu}_k \in \mathbb{R}^D$	:	1st order sufficient statistics for kth Gaussian
$\mathbf{S}_k \in \mathbb{R}^{D \times D}$	:	2nd order sufficient statistics for kth Gaussian
$\boldsymbol{\Psi}$	:	Set of hyper-parameters
$\mathcal{F}(m, \boldsymbol{\Psi})$	:	Variational lower bound given $\boldsymbol{\Psi}$ and m
$q(\cdot)$	:	Variational posterior distribution
$\mathbf{M}_j \in \mathbb{R}^{D \times (D+1)}$	:	location matrix of matrix variate normal distribution at j
$\boldsymbol{\Omega}_j \in \mathbb{R}^{(D+1) \times (D+1)}$	:	scale matrix of matrix variate normal distribution at j
$\boldsymbol{\Phi}_j \in \mathbb{R}^{D \times D}$	:	scale matrix of matrix variate normal distribution at j
$\mathbf{I}_D$	:	$D \times D$ identity matrix
$\rho_j \in \mathbb{R}_{>0}$	:	precision parameter at j
$g(\cdot)$	:	normalization factor of Gaussian distribution
$h(\cdot)$	:	normalization factor of matrix variate normal distribution

the previously estimated HMM parameters $\hat{\boldsymbol{\Theta}}$. Eq. (1) is an expected value, and is efficiently computed by using the forward-backward algorithm as the E-step of the EM algorithm.

The M-step of the EM algorithm estimates HMM parameters, as follows:

$$\bar{\boldsymbol{\Theta}} = \underset{\boldsymbol{\Theta}}{\operatorname{argmax}}\, Q(\boldsymbol{\Theta}; \hat{\boldsymbol{\Theta}}). \tag{2}$$

The E-step and the M-step are performed iteratively until convergence, and finally we obtain the HMM parameters as a close approximate of the stationary point solution.

Now we focus on the linear transformation parameters within the EM algorithm. We prepare a transformation parameter matrix $\mathbf{W}_j$ for each leaf node j in a Gaussian tree. Here, we assume that the Gaussian tree is pruned by a model selection approach as a model structure m, and the set of leaf nodes in the pruned tree is represented as $\mathcal{J}_m$. Hereinafter, we use $\mathbf{V}$ to denote a joint event of $\mathbf{S}$ and $\mathbf{Z}$ (i.e., $\mathbf{V} \triangleq \{\mathbf{S}, \mathbf{Z}\}$). This will much simplify the following development pertaining to the adaptation of the mean and the covariance parameters. Similar to Eq. (1), the auxiliary function with respect to a set of transformation parameters $\boldsymbol{\Lambda}_{\mathcal{J}_m} = \{\mathbf{W}_j \vert j = 1, \cdots, \vert\mathcal{J}_m\vert\}$ can be represented as follows:

$$
\begin{aligned}
Q(\mathbf{\Lambda}_{\mathcal{J}_m};\hat{\mathbf{\Lambda}}_{\mathcal{J}_m}) &= \langle \log p(\mathbf{O},\mathbf{V}|\mathbf{\Lambda}_{\mathcal{J}_m};\boldsymbol{\Theta})\rangle_{p(\mathbf{V}|\mathbf{O};\boldsymbol{\Theta},\hat{\mathbf{\Lambda}}_{\mathcal{J}_m})} \\
&= \sum_{k=1}^{K}\sum_{t=1}^{T}\zeta_{k,t}\log\mathcal{N}(\mathbf{o}_t|\boldsymbol{\mu}_k^{ad},\mathbf{\Sigma}_k),
\end{aligned}
\tag{3}
$$

k denotes a unique mixture component index of all Gaussians in the target HMMs (for all phoneme HMMs in a speech recognition case), and K is the total number of Gaussians. $\zeta_{k,t} \triangleq p(v_t = k|\mathbf{O};\boldsymbol{\Theta},\hat{\mathbf{\Lambda}}_{\mathcal{J}_m})$ is the posterior probability of mixture component k at t, derived from the previously estimated transformation parameters $\hat{\mathbf{\Lambda}}_{\mathcal{J}_m}$[3]. $\boldsymbol{\mu}_k^{ad}$ is a transformed mean vector with $\mathbf{\Lambda}_{\mathcal{J}_m}$, and the concrete form of this vector is discussed in the next paragraph. In the Q function, we disregard the parameters of the state transition probabilities and the mixture weights since they do not depend on the optimization with respect to $\mathbf{\Lambda}_{\mathcal{J}_m}$. $\mathcal{N}(\cdot|\boldsymbol{\mu},\mathbf{\Sigma})$ denotes a Gaussian distribution with mean parameter $\boldsymbol{\mu}$ and covariance matrix parameter $\mathbf{\Sigma}$, and is defined as follows:

$$
\begin{aligned}
&\mathcal{N}(\mathbf{o}_t|\boldsymbol{\mu}_k^{ad},\mathbf{\Sigma}_k) \\
&\triangleq g(\mathbf{\Sigma}_k)\exp\left(-\frac{1}{2}\mathrm{tr}\left[(\mathbf{\Sigma}_k)^{-1}(\mathbf{o}_t-\boldsymbol{\mu}_k^{ad})(\mathbf{o}_t-\boldsymbol{\mu}_k^{ad})'\right]\right),
\end{aligned}
\tag{4}
$$

where tr$[\cdot]$ and $'$ mean the trace and transposition operations of a matrix, respectively. $g(\mathbf{\Sigma}_k)$ is a normalization factor, and is defined as follows:

$$
g(\mathbf{\Sigma}_k) \triangleq (2\pi)^{-\frac{D}{2}}|\mathbf{\Sigma}_k|^{-\frac{1}{2}}. \tag{5}
$$

In the following paragraphs, we derive Eq. (3) as a function of $\mathbf{\Lambda}_{\mathcal{J}_m}$ to optimize $\mathbf{\Lambda}_{\mathcal{J}_m}$, similar to Eq. (2).

We consider the concrete form of the transformed mean vector $(\boldsymbol{\mu}_k^{ad})$ based on the variance normalized representation. We first define Cholesky decomposition matrix $\mathbf{C}_k$ as follows:

$$
\mathbf{\Sigma}_k \triangleq \mathbf{C}_k(\mathbf{C}_k)'. \tag{6}
$$

$\mathbf{C}_k$ is a $D \times D$ triangular matrix. If the Gaussian k is included in a set of Gaussians $\mathcal{K}_j$ in leaf node j (i.e., $k \in \mathcal{K}_j$), the affine transformation of a Gaussian mean vector in a covariance normalized space $(\mathbf{C}_k)^{-1}\boldsymbol{\mu}_k^{ad}$ is represented as follows:

$$
\begin{aligned}
&(\mathbf{C}_k)^{-1}\boldsymbol{\mu}_k^{ad} = \mathbf{W}_j\begin{pmatrix}1\\(\mathbf{C}_k)^{-1}\boldsymbol{\mu}_k^{ini}\end{pmatrix}. \\
&\Rightarrow \boldsymbol{\mu}_k^{ad} = \mathbf{C}_k\mathbf{W}_j\begin{pmatrix}1\\(\mathbf{C}_k)^{-1}\boldsymbol{\mu}_k^{ini}\end{pmatrix} \triangleq \mathbf{C}_k\mathbf{W}_j\boldsymbol{\xi}_k.
\end{aligned}
\tag{7}
$$

$\boldsymbol{\xi}_k$ is an augmented normalized vector of an initial (non-adapted) Gaussian mean vector $\boldsymbol{\mu}_k^{ini}$. $\mathbf{W}_j$ is a $D \times (D+1)$ affine transformation matrix. j is a leaf node index that holds a set of Gaussians. Namely, transformation parameter $\mathbf{W}_j$ is shared among a set of Gaussians $\mathcal{K}_j$. The clustered structure of the Gaussians is usually represented as a binary tree where a set of Gaussians belongs to each node.

The Q function of $\mathbf{\Lambda}_{\mathcal{J}_m}$ is represented by substituting Eqs. (7) and (4) into Eq. (3) as follows:

$$
\begin{aligned}
&Q(\mathbf{\Lambda}_{\mathcal{J}_m};\hat{\mathbf{\Lambda}}_{\mathcal{J}_m}) \\
&= \sum_{j\in\mathcal{J}_m}\sum_{k\in\mathcal{K}_j}\sum_{t=1}^{T}\zeta_{k,t}\log\mathcal{N}(\mathbf{o}_t|\mathbf{C}_k\mathbf{W}_j\boldsymbol{\xi}_k,\mathbf{\Sigma}_k) \\
&= \sum_{j\in\mathcal{J}_m}\left(\sum_{k\in\mathcal{K}_j}\zeta_k\log g(\mathbf{\Sigma}_k) - \frac{1}{2}\mathrm{tr}\left[\mathbf{W}_j'\mathbf{W}_j\mathbf{\Xi}_j\right.\right. \\
&\qquad\qquad \left.\left. - 2\mathbf{W}_j'\mathbf{Z}_j + \sum_{k\in\mathcal{K}_j}\mathbf{\Sigma}_k^{-1}\mathbf{S}_k\right]\right),
\end{aligned}
\tag{8}
$$

where $\mathbf{\Xi}_j$ and $\mathbf{Z}_j$ are 0th and 1st order statistics of linear regression parameters defined as:

$$
\begin{cases}
\mathbf{\Xi}_j \triangleq \displaystyle\sum_{k\in\mathcal{K}_j}\boldsymbol{\xi}_k(\boldsymbol{\xi}_k)'\zeta_k. \\
\mathbf{Z}_j \triangleq \displaystyle\sum_{k\in\mathcal{K}_j}(\mathbf{C}_k)^{-1}\boldsymbol{\nu}_k(\boldsymbol{\xi}_k)'.
\end{cases}
\tag{9}
$$

Here $\mathbf{Z}_j$ is a $D \times (D+1)$ matrix and $\mathbf{\Xi}_j$ is a $(D+1) \times (D+1)$ symmetric matrix. ζ_k, $\boldsymbol{\nu}_k$, and $\mathbf{S}_k$ are defined as follows:

$$
\begin{cases}
\zeta_k = \displaystyle\sum_{t=1}^{T}\zeta_{k,t} \\
\boldsymbol{\nu}_k = \displaystyle\sum_{t=1}^{T}\zeta_{k,t}\mathbf{o}_t \\
\mathbf{S}_k = \displaystyle\sum_{t=1}^{T}\zeta_{k,t}\mathbf{o}_t\mathbf{o}_t'
\end{cases}
\tag{10}
$$

These are the 0th, 1st, and 2nd order sufficient statistics of Gaussians in HMMs, respectively.

Since Eq. (8) is represented as a quadratic form with respect to $\mathbf{W}_j$, we can obtain the optimal $\bar{\mathbf{W}}_j$, similar to Eq. (2). By differentiating the Q function with re-

[3] k denotes a combination of all possible HMM states and mixture components. In the common HMM representation, k can be represented by these two indexes.

spect to $\mathbf{W}_j$, we can derive the following equation[4]

$$\frac{\partial}{\partial \mathbf{W}_j} Q(\boldsymbol{\Lambda}_{\mathcal{J}_m}; \hat{\boldsymbol{\Lambda}}_{\mathcal{J}_m}) = 0. \Rightarrow \mathbf{Z}_j - \bar{\mathbf{W}}_j \boldsymbol{\Xi}_j = 0. \tag{11}$$

Thus, we can obtain the following analytical solution:

$$\bar{\mathbf{W}}_j = \mathbf{Z}_j \boldsymbol{\Xi}_j^{-1}. \tag{12}$$

Therefore, the optimized mean vector parameter is represented as:

$$\boldsymbol{\mu}_k^{ad} = \mathbf{C}_k \mathbf{Z}_j \boldsymbol{\Xi}_j^{-1} \boldsymbol{\xi}_k. \tag{13}$$

Therefore, $\boldsymbol{\mu}_k^{ad}$ is analytically obtained by using the statistics ($\mathbf{Z}_j$ and $\boldsymbol{\Xi}_j$ in Eq. (9)) and initial HMM parameters ($\mathbf{C}_k$ and $\boldsymbol{\xi}_k$). This solution corresponds to the M-step of the EM algorithm, and the E-step is performed by the forward-backward algorithm, similarly to that of HMMs, to compute these statistics.

3 Bayesian Linear Regression

This section provides an analytical solution for Bayesian linear regression by using a variational lower bound. The previous section only considers a regression matrix in leaf node $j \in \mathcal{J}_m$, we also consider a regression matrix in leaf or non-leaf node $i \in \mathcal{I}_m$ in the Gaussian tree given model structure m. Then, we focus on a set of regression matrices in all nodes $\boldsymbol{\Lambda}_{\mathcal{I}_m} = \{\mathbf{W}_i | i = 1, \cdots, |\mathcal{I}_m|\}$, instead of $\boldsymbol{\Lambda}_{\mathcal{J}_m}$, and marginalize $\boldsymbol{\Lambda}_{\mathcal{I}_m}$ in a Bayesian manner. This extension involves the structural prior setting as proposed in SMAP and SMAPLR [17, 18].

In this section, we mainly deal with:

- the prior distribution of model parameters $p(\boldsymbol{\Lambda}_{\mathcal{I}_m}; m, \boldsymbol{\Psi})$
- the true posterior distribution of model parameters and latent variables $p(\boldsymbol{\Lambda}_{\mathcal{I}_m}, \mathbf{V}|\mathbf{O}; m, \boldsymbol{\Psi})$
- the variational posterior distribution of model parameters and latent variables $q(\boldsymbol{\Lambda}_{\mathcal{I}_m}, \mathbf{V}|\mathbf{O}; m, \boldsymbol{\Psi})$
- the output distribution $p(\mathbf{O}, \mathbf{V}|\boldsymbol{\Lambda}_{\mathcal{I}_m}; \boldsymbol{\Theta})$

For simplicity, we omit some conditional variables in these distribution functions, as follows:

$$\begin{array}{lcl} p(\boldsymbol{\Lambda}_{\mathcal{I}_m}; m, \boldsymbol{\Psi}) & \rightarrow & p(\boldsymbol{\Lambda}_{\mathcal{I}_m}) \\ p(\boldsymbol{\Lambda}_{\mathcal{I}_m}, \mathbf{V}|\mathbf{O}; m, \boldsymbol{\Psi}) & \rightarrow & p(\boldsymbol{\Lambda}_{\mathcal{I}_m}, \mathbf{V}|\mathbf{O}) \\ q(\boldsymbol{\Lambda}_{\mathcal{I}_m}, \mathbf{V}|\mathbf{O}; m, \boldsymbol{\Psi}) & \rightarrow & q(\boldsymbol{\Lambda}_{\mathcal{I}_m}, \mathbf{V}) \\ p(\mathbf{O}, \mathbf{V}|\boldsymbol{\Lambda}_{\mathcal{I}_m}; \boldsymbol{\Theta}) & \rightarrow & p(\mathbf{O}, \mathbf{V}|\boldsymbol{\Lambda}_{\mathcal{I}_m}) \end{array}$$

3.1 Variational lower bound

With regard to the variational Bayesian approaches, we first focus on the following marginalized log-likelihood $p(\mathbf{O}; \boldsymbol{\Theta}, m, \boldsymbol{\Psi})$ with a set of HMM parameters $\boldsymbol{\Theta}$, a set of hyper-parameters $\boldsymbol{\Psi}$, and a model structure[5,6].

$$\begin{aligned} &\log p(\mathbf{O}; \boldsymbol{\Theta}, m, \boldsymbol{\Psi}) \\ &= \log \left(\int \sum_{\mathbf{V}} p(\mathbf{O}, \mathbf{V}|\boldsymbol{\Lambda}_{\mathcal{I}_m}; \boldsymbol{\Theta}) p(\boldsymbol{\Lambda}_{\mathcal{I}_m}; m, \boldsymbol{\Psi}) d\boldsymbol{\Lambda}_{\mathcal{I}_m} \right). \end{aligned} \tag{14}$$

where $p(\mathbf{O}, \mathbf{V}|\boldsymbol{\Lambda}_{\mathcal{I}_m}; \boldsymbol{\Theta})$ is the output distribution of the transformed HMM parameters with transformed mean vectors $\boldsymbol{\mu}_k^{ad}$. $p(\boldsymbol{\Lambda}_{\mathcal{I}_m}; m, \boldsymbol{\Psi})$ is a prior distribution of transformation matrices $\boldsymbol{\Lambda}_{\mathcal{I}_m}$. In the following explanation, we omit $\boldsymbol{\Theta}$, m, and $\boldsymbol{\Psi}$ in the prior distribution and output distribution for simplicity, i.e., $p(\boldsymbol{\Lambda}_{\mathcal{I}_m}; m, \boldsymbol{\Psi}) \rightarrow p(\boldsymbol{\Lambda}_{\mathcal{I}_m})$, and $p(\mathbf{O}, \mathbf{V}|\boldsymbol{\Lambda}_{\mathcal{I}_m}; \boldsymbol{\Theta}) \rightarrow p(\mathbf{O}, \mathbf{V}|\boldsymbol{\Lambda}_{\mathcal{I}_m})$.

The variational Bayesian approach focuses on the lower bound of the marginalized log likelihood $\mathcal{F}(m, \boldsymbol{\Psi})$ with a set of hyper-parameters $\boldsymbol{\Psi}$ and a model structure m, as follows:

$$\begin{aligned} &\log p(\mathbf{O}; \boldsymbol{\Theta}, m, \boldsymbol{\Psi}) \\ &= \log \left(\int \sum_{\mathbf{V}} \frac{p(\mathbf{O}, \mathbf{V}|\boldsymbol{\Lambda}_{\mathcal{I}_m}) p(\boldsymbol{\Lambda}_{\mathcal{I}_m})}{q(\boldsymbol{\Lambda}_{\mathcal{I}_m}, \mathbf{V})} q(\boldsymbol{\Lambda}_{\mathcal{I}_m}, \mathbf{V}) d\boldsymbol{\Lambda}_{\mathcal{I}_m} \right) \\ &\geq \underbrace{\left\langle \log \frac{p(\mathbf{O}, \mathbf{V}|\boldsymbol{\Lambda}_{\mathcal{I}_m}) p(\boldsymbol{\Lambda}_{\mathcal{I}_m})}{q(\boldsymbol{\Lambda}_{\mathcal{I}_m}, \mathbf{V})} \right\rangle_{q(\boldsymbol{\Lambda}_{\mathcal{I}_m}, \mathbf{V})}}_{\triangleq \mathcal{F}(m, \boldsymbol{\Psi})}. \end{aligned} \tag{15}$$

The inequality in Eq. (15) is supported by the Jensen's inequality: $\log(\langle \mathbf{X} \rangle_{p(\mathbf{X})}) \geq \langle \log(\mathbf{X}) \rangle_{p(\mathbf{X})}$. $q(\boldsymbol{\Lambda}_{\mathcal{I}_m}, \mathbf{V})$ is an arbitrary distribution, and is optimized by using a variational method to be discussed later. For simplicity, we omit m, $\boldsymbol{\Psi}$, and $\mathbf{O}$ from the distributions. The variational lower bound is a better approximation of the marginalized log likelihood than the auxiliary functions of maximum likelihood EM and maximum a posteriori

[4] We use the following matrix formulate for the derivation:

$$\frac{\partial}{\partial \mathbf{X}} \mathrm{tr}[\mathbf{X}' \mathbf{A}] = \mathbf{A}$$

$$\frac{\partial}{\partial \mathbf{X}} \mathrm{tr}[\mathbf{X}' \mathbf{X} \mathbf{A}] = 2\mathbf{X}\mathbf{A} \quad (\mathbf{A} \text{ is a symmetric matrix})$$

[5] $\boldsymbol{\Psi}$ and m can also be marginalized by setting their distributions. This paper point-estimates $\boldsymbol{\Psi}$ and m by a MAP approach.

[6] We can also marginalize the HMM parameters $\boldsymbol{\Theta}$. This corresponds to jointly optimize HMM and linear regression parameters.

EM algorithms that point-estimate model parameters, especially for small amount of training data [21–23]. Therefore, the variational Bayes can mitigate the sparse data problem that the conventional approaches must confront with.

The variational Bayes regards the variational lower bound $\mathcal{F}(m, \mathbf{\Psi})$ as an objective function for the model structure and hyper-parameter, and an objective functional for the joint posterior distribution of the transformation parameters and latent variables [22, 23]. In particular, if we consider the true posterior distribution $p(\mathbf{\Lambda}_{\mathcal{I}_m}, \mathbf{V}|\mathbf{O})$ (we omit conditional variables m and $\mathbf{\Psi}$ for simplicity), we obtain the following relationship:

$$\begin{aligned} &\mathrm{KL}\left[q(\mathbf{\Lambda}_{\mathcal{I}_m}, \mathbf{V})||p(\mathbf{\Lambda}_{\mathcal{I}_m}, \mathbf{V}|\mathbf{O})\right] = \\ &\qquad \log p(\mathbf{O}; \boldsymbol{\Theta}, m, \mathbf{\Psi}) - \mathcal{F}(m, \mathbf{\Psi}) \end{aligned} \tag{16}$$

This equation means that maximizing the variational lower bound $\mathcal{F}(m, \mathbf{\Psi})$ with respect to $q(\mathbf{\Lambda}_{\mathcal{I}_m}, \mathbf{V})$ corresponds to minimizing the Kullback-Leibler (KL) divergence between $q(\mathbf{\Lambda}_{\mathcal{I}_m}, \mathbf{V})$ and $p(\mathbf{\Lambda}_{\mathcal{I}_m}, \mathbf{V}|\mathbf{O})$ indirectly. Therefore, this optimization yields to find $q(\mathbf{\Lambda}_{\mathcal{I}_m}, \mathbf{V})$, which approaches to the true posterior distribution[7].

Thus, in principle, we can straightforwardly obtain the (sub) optimal model structure, hyper-parameters, and posterior distribution, as follows:

$$\begin{aligned} \tilde{m} &= \underset{m}{\mathrm{argmax}}\, \mathcal{F}(m, \mathbf{\Psi}). \\ \tilde{\mathbf{\Psi}} &= \underset{\mathbf{\Psi}}{\mathrm{argmax}}\, \mathcal{F}(m, \mathbf{\Psi}). \\ \tilde{q}(\mathbf{\Lambda}_{\mathcal{I}_m}, \mathbf{V}) &= \underset{q(\mathbf{\Lambda}_{\mathcal{I}_m}, \mathbf{V})}{\mathrm{argmax}}\, \mathcal{F}(m, \mathbf{\Psi}). \end{aligned} \tag{17}$$

This optimization steps are performed alternately, and finally lead to local optimum solutions, similar to the EM algorithm. However it is difficult to deal with the joint distribution $q(\mathbf{\Lambda}_{\mathcal{I}_m}, \mathbf{V})$ directly, and we propose to factorize them by utilizing a Gaussian tree structure. In addition, we also set a conjugate form of the prior distribution $p(\mathbf{\Lambda}_{\mathcal{I}_m})$. This procedure is a typical recipe of VB to make a solution mathematically tractable similar to that of the classical Bayesian adaptation approach.

3.2 Structural prior distribution setting in a binary tree

We utilize a Gaussian tree structure to factorize the prior distribution $p(\mathbf{\Lambda}_{\mathcal{I}_m})$. We consider a binary tree structure, but the formulation is applicable to a general non-binary tree. We define the parent node of i as $\mathsf{p}(i)$, the left child node of i as $\mathsf{l}(i)$, and the right child node of i as $\mathsf{r}(i)$, as shown in Figure 2, where a transformation matrix is prepared for each corresponding node i. If we define $\mathbf{W}_1$ as the transformation matrix in the root node, we assume the following factorization for the prior distribution $p(\mathbf{\Lambda}_{\mathcal{I}_m})$,

$$\begin{aligned} p(\mathbf{\Lambda}_{\mathcal{I}_m}) &= p(\mathbf{W}_1, \cdots, \mathbf{W}_{|\mathcal{I}_m|}) \\ &= p(\mathbf{W}_1) p(\mathbf{W}_{\mathsf{l}(1)}|\mathbf{W}_1) p(\mathbf{W}_{\mathsf{r}(1)}|\mathbf{W}_1) \\ &\quad\, p(\mathbf{W}_{\mathsf{l}(\mathsf{l}(1))}|\mathbf{W}_{\mathsf{l}(1)}) p(\mathbf{W}_{\mathsf{r}(\mathsf{l}(1))}|\mathbf{W}_{\mathsf{l}(1)}) \\ &\quad\, p(\mathbf{W}_{\mathsf{l}(\mathsf{r}(1))}|\mathbf{W}_{\mathsf{r}(1)}) p(\mathbf{W}_{\mathsf{r}(\mathsf{r}(1))}|\mathbf{W}_{\mathsf{r}(1)}) \cdots \\ &= \prod_{i \in \mathcal{I}_m} p(\mathbf{W}_i|\mathbf{W}_{\mathsf{p}(i)}). \end{aligned} \tag{18}$$

To make the prior distribution a product form in the last line of Eq. (18), we define $p(\mathbf{W}_1) \triangleq p(\mathbf{W}_1|\mathbf{W}_{\mathsf{p}(1)})$. As seen, the effect of the transformation matrix in a target node propagates to its child nodes.

This prior setting is based on an intuitive assumption that the statistics in a target node is highly correlated with the statistics in its parent node. In addition, since the total number of speech frames assigned to a set of Gaussians in the parent node is always larger than that in the target node, the obtained statistics in the parent node is more reliable than that in the target node, and these can be good prior knowledge for the transformation parameter estimation in the target node.

With a Bayesian approach, we need to set a practical form of the above prior distributions. A conjugate distribution is preferable as far as obtaining an analytical solution is concerned, and we set a matrix variate normal distribution similar to Maximum A Posteriori Linear Regression (MAPLR [15]). A matrix variate normal distribution is defined as follows:

$$\begin{aligned} p(\mathbf{W}_i) &= \mathcal{N}(\mathbf{W}_i|\mathbf{M}_i, \mathbf{\Phi}_i, \mathbf{\Omega}_i) \\ &\triangleq \frac{\exp\left(-\frac{1}{2}\mathrm{tr}\left[(\mathbf{W}_i - \mathbf{M}_i)' \mathbf{\Phi}_i^{-1} (\mathbf{W}_i - \mathbf{M}_i) \mathbf{\Omega}_i^{-1}\right]\right)}{(2\pi)^{D(D+1)/2} |\mathbf{\Omega}_i|^{D/2} |\mathbf{\Phi}_i|^{(D+1)/2}}, \end{aligned} \tag{19}$$

where $\mathbf{M}_i$ is a $D \times (D+1)$ location matrix, $\mathbf{\Omega}_i$ is a $(D+1) \times (D+1)$ symmetric scale matrix, and $\mathbf{\Phi}_i$ is a $D \times D$ symmetric scale matrix. $\mathbf{\Omega}_i$ represents correlation of column vectors, and $\mathbf{\Phi}_i$ represents correlation

[7] The following sections assume factorization forms of $q(\mathbf{\Lambda}_{\mathcal{I}_m}, \mathbf{V})$ to make solutions mathematical tractable. However, this factorization assumption weakens the relationship between the KL divergence and the variational lower bound. For example, if we assume $q(\mathbf{\Lambda}_{\mathcal{I}_m}, \mathbf{V}) = q(\mathbf{\Lambda}_{\mathcal{I}_m})q(\mathbf{V})$, and focus on the KL divergence between $q(\mathbf{\Lambda}_{\mathcal{I}_m})$ and $p(\mathbf{\Lambda}_{\mathcal{I}_m}|\mathbf{O})$, we obtain the following inequality:

$$\mathrm{KL}\left[q(\mathbf{\Lambda}_{\mathcal{I}_m})||p(\mathbf{\Lambda}_{\mathcal{I}_m}|\mathbf{O})\right] \leq \log p(\mathbf{O}; \boldsymbol{\Theta}, m, \mathbf{\Psi}) - \mathcal{F}(m, \mathbf{\Psi}).$$

Compared with Eq. (16), the relationship between the KL divergence and the variational lower bound are less direct due to the inequality relationship. In general, the factorization assumption distances optimal variational posteriors from the true posterior within the VB framework.

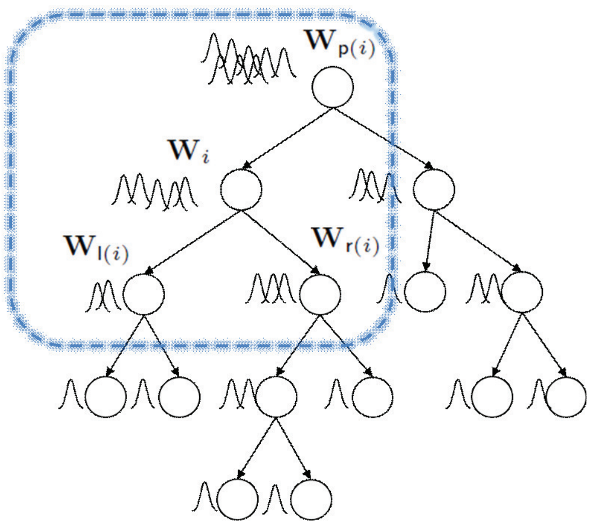
$\mathbf{W}_{\mathsf{p}(i)}$
$\mathbf{W}_i$
$\mathbf{W}_{\mathsf{l}(i)}$
$\mathbf{W}_{\mathsf{r}(i)}$

Fig. 2 Binary tree structure with transformation matrices. If we focus on node i, the transformation matrices in the parent node, left child node, and right child node are represented as $\mathbf{W}_{\mathsf{p}(i)}$, $\mathbf{W}_{\mathsf{l}(i)}$, and $\mathbf{W}_{\mathsf{r}(i)}$, respectively.

of raw vectors. These are hyper-parameters of the matrix variate normal distribution. There are many hyper-parameters to be set, and this makes the implementation complicated. In this paper, we try to find another conjugate distribution with fewer hyper-parameters than Eq. (19). To obtain a simple solution for the final analytical results, we use a spherical Gaussian distribution that has the following constraints on $\mathbf{\Omega}_i$ and $\mathbf{\Phi}_i$:

$$\begin{aligned} &\mathbf{\Phi}_i \approx \mathbf{I}_D, \\ &\mathbf{\Omega}_i \approx \rho_i^{-1}\mathbf{I}_{D+1}, \end{aligned} \tag{20}$$

where $\mathbf{I}_D$ is the $D \times D$ identity matrix. ρ_i indicates a precision parameter. Then, Eq. (19) can be rewritten as follows:

$$\begin{aligned} &\mathcal{N}(\mathbf{W}_i|\mathbf{M}_i, \mathbf{I}_D, \rho_i^{-1}\mathbf{I}_{D+1}) \\ &\quad = h(\rho_i^{-1}\mathbf{I}_{D+1}) \exp\left(-\frac{1}{2}\mathrm{tr}\left[\rho_i(\mathbf{W}_i - \mathbf{M}_i)'(\mathbf{W}_i - \mathbf{M}_i)\right]\right), \end{aligned} \tag{21}$$

where $h(\rho_i^{-1}\mathbf{I}_{D+1})$ is a normalization factor, and defined as

$$h(\rho_i^{-1}\mathbf{I}_{D+1}) \triangleq \left(\frac{\rho_i}{2\pi}\right)^{\frac{D(D+1)}{2}}. \tag{22}$$

This approximation means that matrix elements do not have any correlation each other. This can produce simple solutions for Bayesian linear regression[8]

Based on the spherical matrix variate normal distribution, the conditional prior distribution $p(\mathbf{W}_i|\mathbf{W}_{\mathsf{p}(i)})$ in Eq. (18) is obtaining by setting the location matrix as the transformation matrix $\mathbf{W}_{\mathsf{p}(i)}$ in the parent node with the precision parameter ρ_i as follows:

$$p(\mathbf{W}_i|\mathbf{W}_{\mathsf{p}(i)}) = \mathcal{N}(\mathbf{W}_i|\mathbf{W}_{\mathsf{p}(i)}, \mathbf{I}_D, \rho_i^{-1}\mathbf{I}_{D+1}) \tag{23}$$

Note that in the following sections $\mathbf{W}_i$ and $\mathbf{W}_{\mathsf{p}(i)}$ are marginalized. In addition, we set the location matrix in the root node as the deterministic value of $\mathbf{W}_{\mathsf{p}(1)} = [\mathbf{0}, \mathbf{I_D}]$. Since $\boldsymbol{\mu}_k^{ad} = \mathbf{C}_k\mathbf{W}_{\mathsf{p}(1)}\boldsymbol{\xi}_k = \boldsymbol{\mu}_k^{ini}$ from Eq. (7), this hyper-parameter setting means that the initial mean vectors are not changed if we only use the prior knowledge. This makes sense in the case of small amount of data by fixing the HMM parameters as their initial values; this in a sense also inherits the philosophical background of Bayesian adaptation, although the objective function has been changed from a posteriori probability to a lower bound of the marginalized likelihood. Therefore, we just have $\{\rho_i | i = 1, \cdots, |\mathcal{I}_m|\}$ as a set of hyper-parameters $\mathbf{\Psi}$, which will be also optimized in our framework.

3.3 Variational calculus

In VB, we also assume the following factorization form to the posterior distribution $q(\mathbf{V}, \mathbf{\Lambda}_{\mathcal{I}_m})$:

$$q(\mathbf{V}, \mathbf{\Lambda}_{\mathcal{I}_m}) = q(\mathbf{V})q(\mathbf{\Lambda}_{\mathcal{I}_m}) = q(\mathbf{V}) \prod_{i \in \mathcal{I}_m} q(\mathbf{W}_i) \tag{24}$$

Then, from the variational calculation for $\mathcal{F}(m, \mathbf{\Psi})$ with respect to $q(\mathbf{W}_i)$, we obtain the following (sub) optimal solution for $q(\mathbf{W}_i)$:

$$\begin{aligned} &\log \tilde{q}(\mathbf{W}_i) \\ &\propto \left\langle \langle \log p(\mathbf{O}, \mathbf{V}|\mathbf{\Lambda}_{\mathcal{I}_m})\rangle_{q(\mathbf{V})}\, p(\mathbf{\Lambda}_{\mathcal{I}_m}) \right\rangle_{\prod_{i' \neq i \in \mathcal{I}_m} q(\mathbf{W}_{i'})} \\ &\propto \sum_{i' \in \mathcal{I}_m} \left\langle \log p(\mathbf{W}_{i'}|\mathbf{W}_{\mathsf{p}(i')}) \right\rangle_{\prod_{i' \neq i \in \mathcal{I}_m} q(\mathbf{W}_{i'})} \\ &\quad + \left\langle \langle \log p(\mathbf{O}, \mathbf{V}|\mathbf{\Lambda}_{\mathcal{I}_m})\rangle_{q(\mathbf{V})} \right\rangle_{\prod_{i' \neq i \in \mathcal{I}_m} q(\mathbf{W}_{i'})}, \end{aligned} \tag{25}$$

[8] Matrix variate normal distribution in Eq. (19) is also represented by the following multivariate normal distribution [36]:

$$\begin{aligned} &\mathcal{N}(\mathbf{W}_i|\mathbf{M}_i, \mathbf{\Phi}_i, \mathbf{\Omega}_i) \\ &\quad \propto \exp\left(-\frac{1}{2}\,\mathrm{vec}(\mathbf{W}_i - \mathbf{M}_i)'(\mathbf{\Omega}_i \otimes \mathbf{\Phi}_i)^{-1}\,\mathrm{vec}(\mathbf{W}_i - \mathbf{M}_i)^{-1}\right), \end{aligned}$$

where $\mathrm{vec}(\mathbf{W}_i - \mathbf{M}_i)$ is a vector formed by the concatenation of the columns of $(\mathbf{W}_i - \mathbf{M}_i)$, and $\otimes$ denotes the Kronecker product. Based on this form, a VB solution in this paper could be extended without considering the variance normalized representation used in this paper according to [16].

where we use Eqs. (18) and (24) to rewrite the equation. Operation $\propto$ denotes the proportional relationship between the left and the right hand sides of the probabilistic distribution functions. It is a useful expression since we do not have to write normalization factors explicitly, which are disregarded in the following calculations. In Eq. (25), $\propto$ is also used in the logarithmic domain where normalization factors can be represented as constant terms.

In this expectation, we can consider the following two cases of variational posterior distributions:

1) Leaf node

We first focus on the prior term of Eq. (25). If i is a leaf node, we can disregard the expectation with respect to $\prod_{i' \neq i \in \mathcal{I}_m} q(\mathbf{W}_{i'})$ in the other nodes than the parent node $\mathsf{p}(i)$ of the target leaf node. Thus, we obtain the following simple solution:

$$\log \tilde{q}(\mathbf{W}_i) \propto \left\langle \log p(\mathbf{W}_i | \mathbf{W}_{\mathsf{p}(i)}) \right\rangle_{q(\mathbf{W}_{\mathsf{p}(i)})} + \left\langle \langle \log p(\mathbf{O}, \mathbf{V} | \mathbf{\Lambda}_{\mathcal{I}_m}) \rangle_{q(\mathbf{V})} \right\rangle_{\prod_{i' \neq i \in \mathcal{I}_m} q(\mathbf{W}_{i'})} \tag{26}$$

2) Non-leaf node (with child nodes)

Similarly, if i is a non-leaf node, in addition to the parent node $\mathsf{p}(i)$ of the target node, we also have to consider the child nodes $\mathsf{l}(i)$ and $\mathsf{r}(i)$ of the target node for the expectation, as follows:

$$\log \tilde{q}(\mathbf{W}_i) \propto \left\langle \log p(\mathbf{W}_i | \mathbf{W}_{\mathsf{p}(i)}) \right\rangle_{q(\mathbf{W}_{\mathsf{p}(i)})} \tag{27-1}$$
$$+ \left\langle \log p(\mathbf{W}_{\mathsf{l}(i)} | \mathbf{W}_i) \right\rangle_{q(\mathbf{W}_{\mathsf{l}(i)})} \tag{27-2}$$
$$+ \left\langle \log p(\mathbf{W}_{\mathsf{r}(i)} | \mathbf{W}_i) \right\rangle_{q(\mathbf{W}_{\mathsf{r}(i)})} \tag{27-3}$$
$$+ \left\langle \langle \log p(\mathbf{O}, \mathbf{V} | \mathbf{\Lambda}_{\mathcal{I}_m}) \rangle_{q(\mathbf{V})} \right\rangle_{\prod_{i' \neq i \in \mathcal{I}_m} q(\mathbf{W}_{i'})} \tag{27-4}$$

In both cases, the posterior distribution of the transformation matrix in the target node depends on those in the parent and child nodes. Therefore, the posterior distributions are iteratively calculated. This inference is known as a variational message passing algorithm [34], and Eqs. (26) and (27) are specific solutions of the variational message passing algorithm to a binary tree structure. The next section provides a concrete form of the posterior distribution of the transformation matrix.

3.4 Posterior distribution of transformation matrix

We first focus on Eq. (27), which is a general equation of Eq. (26) that has additional terms based on child nodes to Eq. (26). Eq. (27-4) is based on the expectation with respect to $\prod_{i' \neq i \in \mathcal{I}_m} q(\mathbf{W}_{i'})$ and $q(\mathbf{V})$. The term with $q(\mathbf{V})$ is represented as the following expression similar to Eq. (8):

$$\begin{aligned} &\langle \log p(\mathbf{O}, \mathbf{V} | \mathbf{\Lambda}_{\mathcal{I}_m}) \rangle_{q(\mathbf{V})} \\ &= \sum_{i \in \mathcal{I}_m} \left(\sum_{k \in \mathcal{K}_i} \zeta_k \log g(\mathbf{\Sigma}_k) \right. \\ &\qquad \left. - \frac{1}{2} \mathrm{tr} \Big[\mathbf{W}_i' \mathbf{W}_i \mathbf{\Xi}_i - 2 \mathbf{W}_i' \mathbf{Z}_i + \sum_{k \in \mathcal{K}_i} \mathbf{\Sigma}_k^{-1} \mathbf{S}_k \Big] \right). \end{aligned} \tag{28}$$

Here sufficient statistics (ζ_k, $\mathbf{S}_k$, $\mathbf{\Xi}_i$, and $\mathbf{Z}_i$ in Eqs. (9) and (10)) are computed by the VB-E step (e.g., $\zeta_{k,t} = q(v_t = k)$), which is described in the next section. This equation form means that the term can be factorized by node i. This factorization property is important for the following analytic solutions and algorithm. Actually, by considering the expectation with respect to $\prod_{i' \neq i \in \mathcal{I}_m} q(\mathbf{W}_{i'})$, we can integrate out the terms that do not depend on $\mathbf{W}_i$, as follows:

$$\begin{aligned} &\left\langle \langle \log p(\mathbf{O}, \mathbf{V} | \mathbf{\Lambda}_{\mathcal{I}_m}) \rangle_{q(\mathbf{V})} \right\rangle_{\prod_{i' \neq i \in \mathcal{I}_m} q(\mathbf{W}_{i'})} \\ &\propto -\frac{1}{2} \mathrm{tr} \left[\mathbf{W}_i' \mathbf{W}_i \mathbf{\Xi}_i - 2 \mathbf{W}_i' \mathbf{Z}_i \right]. \end{aligned} \tag{29}$$

Next, we consider Eq. (27-1). Since we use a conjugate prior distribution, $q(\mathbf{W}_{\mathsf{p}(i)})$ is also represented by the following matrix variate normal distribution as the same distribution family with the prior distribution.

$$q(\mathbf{W}_{\mathsf{p}(i)}) = \mathcal{N}(\mathbf{W}_{\mathsf{p}(i)} | \mathbf{M}_{\mathsf{p}(i)}, \mathbf{I}_D, \mathbf{\Omega}_{\mathsf{p}(i)}) \tag{30}$$

Note that the posterior distribution has a unique form that the first covariance matrix is an identity matrix while the second one is a symmetric matrix. We discuss about this form with the analytical solution, later.

By substituting Eqs. (18) and (30) into Eq. (27-1), Eq. (27-1) is represented as follows:

$$\begin{aligned} &\left\langle \log p(\mathbf{W}_i | \mathbf{W}_{\mathsf{p}(i)}) \right\rangle_{q(\mathbf{W}_{\mathsf{p}(i)})} \\ &= \int \left(\log \mathcal{N}(\mathbf{W}_i | \mathbf{W}_{\mathsf{p}(i)}, \mathbf{I}_D, \rho_i^{-1} \mathbf{I}_{D+1}) \right) \\ &\qquad \mathcal{N}(\mathbf{W}_{\mathsf{p}(i)} | \mathbf{M}_{\mathsf{p}(i)}, \mathbf{I}_D, \mathbf{\Omega}_{\mathsf{p}(i)}) d\mathbf{W}_{\mathsf{p}(i)} \end{aligned} \tag{31}$$

To solve the integral, we use the following matrix distribution formula:

$$\int \mathcal{N}(\mathbf{W}_{\mathsf{p}(i)} | \mathbf{M}_{\mathsf{p}(i)}, \mathbf{I}_D, \mathbf{\Omega}_{\mathsf{p}(i)}) d\mathbf{W}_{\mathsf{p}(i)} = 1$$
$$\int \mathbf{W}_{\mathsf{p}(i)} \mathcal{N}(\mathbf{W}_{\mathsf{p}(i)} | \mathbf{M}_{\mathsf{p}(i)}, \mathbf{I}_D, \mathbf{\Omega}_{\mathsf{p}(i)}) d\mathbf{W}_{\mathsf{p}(i)} = \mathbf{M}_{\mathsf{p}(i)}$$

(32)

Then, by disregarding the terms that do not depend on $\mathbf{W}_i$, Eq. (31) can be solved as the logarithmic function of the matrix variate normal distribution that has the posterior distribution parameter $\mathbf{M}_{\mathsf{p}(i)}$ as a hyperparameter.

$$\begin{aligned}
&\left\langle \log p(\mathbf{W}_i|\mathbf{W}_{\mathsf{p}(i)}) \right\rangle_{q(\mathbf{W}_{\mathsf{p}(i)})} \\
&\quad \propto \rho_i \int \mathrm{tr}\left[\mathbf{W}_i'\mathbf{W}_{\mathsf{p}(i)}\right] \mathcal{N}(\mathbf{W}_{\mathsf{p}(i)}|\mathbf{M}_{\mathsf{p}(i)}, \mathbf{I}_D, \boldsymbol{\Omega}_{\mathsf{p}(i)}) d\mathbf{W}_{\mathsf{p}(i)} \\
&\qquad - \frac{\rho_i}{2} \int \mathrm{tr}\left[\mathbf{W}_i'\mathbf{W}_i\right] \mathcal{N}(\mathbf{W}_{\mathsf{p}(i)}|\mathbf{M}_{\mathsf{p}(i)}, \mathbf{I}_D, \boldsymbol{\Omega}_{\mathsf{p}(i)}) d\mathbf{W}_{\mathsf{p}(i)} \\
&\quad \propto \rho_i \mathrm{tr}\left[\mathbf{W}_i'\mathbf{M}_{\mathsf{p}(i)}\right] - \frac{\rho_i}{2}\mathrm{tr}\left[\mathbf{W}_i'\mathbf{W}_i\right] \\
&\quad \propto \log \mathcal{N}(\mathbf{W}_i|\mathbf{M}_{\mathsf{p}(i)}, \mathbf{I}_D, \rho_i^{-1}\mathbf{I}_{D+1})
\end{aligned} \tag{33}$$

Similarly, Eqs. (27-2) and (27-3) are solved as follows:

$$\begin{aligned}
&\left\langle \log p(\mathbf{W}_{\mathsf{l}(i)}|\mathbf{W}_i) \right\rangle_{q(\mathbf{W}_{\mathsf{l}(i)})} \\
&\quad \propto \log \mathcal{N}(\mathbf{W}_i|\mathbf{M}_{\mathsf{l}(i)}, \mathbf{I}_D, \rho_{\mathsf{l}(i)}^{-1}\mathbf{I}_{D+1}) \\
&\left\langle \log p(\mathbf{W}_{\mathsf{r}(i)}|\mathbf{W}_i) \right\rangle_{q(\mathbf{W}_{\mathsf{r}(i)})} \\
&\quad \propto \log \mathcal{N}(\mathbf{W}_i|\mathbf{M}_{\mathsf{r}(i)}, \mathbf{I}_D, \rho_{\mathsf{r}(i)}^{-1}\mathbf{I}_{D+1})
\end{aligned} \tag{34}$$

Thus, the expected value terms of the three prior distributions in Eq. (27) are represented as the following matrix variate normal distribution:

$$\begin{aligned}
&\left\langle \log p(\mathbf{W}_i|\mathbf{W}_{\mathsf{p}(i)}) \right\rangle_{q(\mathbf{W}_{\mathsf{p}(i)})} \\
&\quad + \left\langle \log p(\mathbf{W}_{\mathsf{l}(i)}|\mathbf{W}_i) \right\rangle_{q(\mathbf{W}_{\mathsf{l}(i)})} \\
&\quad + \left\langle \log p(\mathbf{W}_{\mathsf{r}(i)}|\mathbf{W}_i) \right\rangle_{q(\mathbf{W}_{\mathsf{r}(i)})} \\
&\propto \log \mathcal{N}(\mathbf{W}_i|\mathbf{M}_{\mathsf{p}(i)}, \mathbf{I}_D, \rho_i^{-1}\mathbf{I}_{D+1}) \\
&\quad + \log \mathcal{N}(\mathbf{W}_i|\mathbf{M}_{\mathsf{l}(i)}, \mathbf{I}_D, \rho_{\mathsf{l}(i)}^{-1}\mathbf{I}_{D+1}) \\
&\quad + \log \mathcal{N}(\mathbf{W}_i|\mathbf{M}_{\mathsf{r}(i)}, \mathbf{I}_D, \rho_{\mathsf{r}(i)}^{-1}\mathbf{I}_{D+1}) \\
&\propto \log \mathcal{N}\left(\mathbf{W}_i \left| \frac{\rho_i\mathbf{M}_{\mathsf{p}(i)} + \rho_{\mathsf{l}(i)}\mathbf{M}_{\mathsf{l}(i)} + \rho_{\mathsf{r}(i)}\mathbf{M}_{\mathsf{r}(i)}}{\rho_i + \rho_{\mathsf{l}(i)} + \rho_{\mathsf{r}(i)}}, \right.\right. \\
&\qquad\qquad \left. \mathbf{I}_D, (\rho_i + \rho_{\mathsf{l}(i)} + \rho_{\mathsf{r}(i)})^{-1}\mathbf{I}_{D+1} \right)
\end{aligned} \tag{35}$$

It is an intuitive solution, since the location parameter of $\mathbf{W}_i$ is represented as a linear interpolation of the location values of the posterior distributions in the parent and child nodes. The precision parameters control the linear interpolation ratio.

Similarly, we can also obtain the expected value term of the prior term in Eq. (26), and we summarize the prior terms of the non-leaf and leaf node cases as follows:

$$\hat{q}(\mathbf{W}_i) = \mathcal{N}(\mathbf{W}_i|\hat{\mathbf{M}}_i, \mathbf{I}_D, \hat{\rho}_i^{-1}\mathbf{I}_{D+1}) \tag{36}$$

where

$$\begin{aligned}
\hat{\mathbf{M}}_i &= \begin{cases} \frac{\rho_i\mathbf{M}_{\mathsf{p}(i)} + \rho_{\mathsf{l}(i)}\mathbf{M}_{\mathsf{l}(i)} + \rho_{\mathsf{r}(i)}\mathbf{M}_{\mathsf{r}(i)}}{\rho_i + \rho_{\mathsf{l}(i)} + \rho_{\mathsf{r}(i)}} & \text{Non-leaf node} \\ \mathbf{M}_{\mathsf{p}(i)} & \text{Leaf node} \end{cases} \\
\hat{\rho}_i &= \begin{cases} \rho_i + \rho_{\mathsf{l}(i)} + \rho_{\mathsf{r}(i)} & \text{Non-leaf node} \\ \rho_i & \text{Leaf node} \end{cases}
\end{aligned} \tag{37}$$

Thus, the effect of prior distributions becomes different depending on whether the target node is a non-leaf node or leaf node. The solution is different from our previous solution [37] since the previous solution does not marginalize the transformation parameters in non-leaf nodes. In the Bayesian sense, this solution is stricter than the previous solution.

Based on Eqs. (28) and (36), we can finally derive the quadratic form of $\mathbf{W}_i$ as follows:

$$\begin{aligned}
&\log(\tilde{q}(\mathbf{W}_i)) \\
&\propto -\frac{1}{2}\mathrm{tr}\left[\hat{\rho}_i\mathbf{W}_i'\mathbf{W}_i + \mathbf{W}_i'\mathbf{W}_i\boldsymbol{\Xi}_i - 2\hat{\rho}_i\mathbf{W}_i'\hat{\mathbf{M}}_i - 2\mathbf{W}_i'\mathbf{Z}_i\right] \\
&= -\frac{1}{2}\mathrm{tr}\left[\mathbf{W}_i'\mathbf{W}_i(\hat{\rho}_i\mathbf{I}_{D+1} + \boldsymbol{\Xi}_i) - 2\mathbf{W}_i'(\hat{\rho}_i\hat{\mathbf{M}}_i + \mathbf{Z}_i)\right],
\end{aligned} \tag{38}$$

where we disregard the terms that do not depend on $\mathbf{W}_i$. Thus, by defining the following matrix variables

$$\begin{aligned}
\tilde{\boldsymbol{\Omega}}_i &= (\hat{\rho}_i\mathbf{I}_{D+1} + \boldsymbol{\Xi}_i)^{-1}, \\
&= \begin{cases} \left((\rho_i + \rho_{\mathsf{l}(i)} + \rho_{\mathsf{r}(i)})\mathbf{I}_{D+1} + \boldsymbol{\Xi}_i\right)^{-1} & \text{Non-leaf node} \\ (\rho_i\mathbf{I}_{D+1} + \boldsymbol{\Xi}_i)^{-1} & \text{Leaf node} \end{cases} \\
\tilde{\mathbf{M}}_i &= \left(\hat{\rho}_i\hat{\mathbf{M}}_i + \mathbf{Z}_i\right)\tilde{\boldsymbol{\Omega}}, \\
&= \begin{cases} \left(\rho_i\mathbf{M}_{\mathsf{p}(i)} + \rho_{\mathsf{l}(i)}\mathbf{M}_{\mathsf{l}(i)} + \rho_{\mathsf{r}(i)}\mathbf{M}_{\mathsf{r}(i)} + \mathbf{Z}_i\right)\tilde{\boldsymbol{\Omega}} \\ \qquad \text{Non-leaf node} \\ \left(\rho_i\mathbf{M}_{\mathsf{p}(i)} + \mathbf{Z}_i\right)\tilde{\boldsymbol{\Omega}} \\ \qquad \text{Leaf node} \end{cases}
\end{aligned} \tag{39}$$

we can derive the posterior distribution of $\mathbf{W}_i$ analytically. The analytical solution is expressed as

$$\begin{aligned}
&\tilde{q}(\mathbf{W}_i) = \mathcal{N}(\mathbf{W}_i|\tilde{\mathbf{M}}_i, \mathbf{I}_D, \tilde{\boldsymbol{\Omega}}_i) \\
&= h(\tilde{\boldsymbol{\Omega}}_i)\exp\left(-\frac{1}{2}\mathrm{tr}\left[(\mathbf{W}_i - \tilde{\mathbf{M}}_i)'(\mathbf{W}_i - \tilde{\mathbf{M}}_i)\tilde{\boldsymbol{\Omega}}_i^{-1}\right]\right),
\end{aligned} \tag{40}$$

where

$$h(\tilde{\boldsymbol{\Omega}}_i) \triangleq (2\pi)^{-\frac{D(D+1)}{2}}|\tilde{\boldsymbol{\Omega}}_i|^{-\frac{D}{2}}. \tag{41}$$

The posterior distribution also becomes a matrix variate normal distribution since we use a conjugate prior distribution for $\mathbf{W}_i$. From Eq. (39), $\tilde{\mathbf{M}}_i$ are linearly interpolated by hyper-parameter $\hat{\mathbf{M}}_i$ and the 1st order statistics of the linear regression matrix $\mathbf{Z}_i$. $\hat{\rho}_i$ controls the balance between the effects of the prior distribution and adaptation data. This solution is the M-step of the VB EM algorithm and corresponds to that of the ML EM algorithm in Section 2.1.

Compared with Eq. (21), Eq. (40) keeps the first covariance matrix as a diagonal matrix, while the second covariance matrix $\tilde{\mathbf{\Omega}}$ has off diagonal elements. This means that the posterior distribution only considers the correlation between column vectors in $\mathbf{W}$. This unique property comes from the variance normalized representation introduced in Section 2, which makes multivariate Gaussian distributions in HMMs uncorrelated, and this relationship is taken over to the VB solutions.

Although the solution for a non-leaf node would make the prior distribution robust by taking account of the child node hyper-parameters, this structure makes the dependency of the target node with the other linked nodes complex. Therefore, in the implementation step, we approximate the hyper-parameters of the posterior distribution for a non-leaf node to those for a leaf node by $\hat{\mathbf{M}}_i \approx \mathbf{M}_{\mathsf{p}(i)}$ and $\hat{\rho}_i \approx \rho_i$ in the Eq. (37), to make an algorithm simple. We would evaluate the effect of the non-leaf node solution in future work.

Next section explains the E-step of the VB EM algorithm, which computes sufficient statistics ζ_k, $\mathbf{S}_k$, $\mathbf{\Xi}_i$, and $\mathbf{Z}_i$ in Eqs. (9) and (10). These are obtained by using $\tilde{q}(\mathbf{W}_i)$, of which mode $\tilde{\mathbf{M}}_i$ is used for the Gaussian mean vector transformation.

3.5 Posterior distribution of latent variables

From the variational calculation of $\mathcal{F}(m, \mathbf{\Psi})$ with respect to $q(\mathbf{V})$, we also obtain the following posterior distribution:

$$\log \tilde{q}(\mathbf{V}) \propto \langle \log p(\mathbf{O}, \mathbf{V} | \mathbf{\Lambda}_{\mathcal{I}_m}) \rangle_{q(\mathbf{\Lambda}_{\mathcal{I}_m})} . \tag{42}$$

By using the factorization form of the variational posterior (Eq. (24)), we can disregard the expectation with respect to the other variational posteriors than that of the target node i. Therefore, to obtain the above VB posteriors of latent variables, we have to consider the following integral.

$$\int q(\mathbf{W}_i) \log \mathcal{N}(\mathbf{o}_t | \mathbf{C}_k \mathbf{W}_i \boldsymbol{\xi}_k, \mathbf{\Sigma}_k) d\mathbf{W}_i. \tag{43}$$

Since the Gaussian mean vectors are only updated in the leaf nodes, node i in this section is regarded as a leaf node. By substituting Eqs. (40) and (4) into Eq. (43), the equation is represented as (see Appendix A):

$$\begin{aligned} &\int q(\mathbf{W}_i) \log \mathcal{N}(\mathbf{o}_t | \mathbf{C}_k \mathbf{W}_i \boldsymbol{\xi}_k, \mathbf{\Sigma}_k) d\mathbf{W}_i \\ &= \log \mathcal{N}(\mathbf{o}_t | \tilde{\boldsymbol{\mu}}_k, \mathbf{\Sigma}_k) - \frac{1}{2} \mathrm{tr} \left[\boldsymbol{\xi}_k \boldsymbol{\xi}_k' \tilde{\mathbf{\Omega}}_i \right] . \end{aligned} \tag{44}$$

where

$$\tilde{\boldsymbol{\mu}}_k = \mathbf{C}_k \tilde{\mathbf{M}}_i \boldsymbol{\xi}_k \tag{45}$$

The analytical result is almost equivalent to the E-step of conventional MLLR, which means that the computation time is almost the same as that of the conventional MLLR E-step.

Note that the Gaussian mean vectors are updated in the leaf nodes in this result, while the posterior distributions of the transformation parameters are updated for all nodes.

3.6 Variational lower bound

By using the factorization form (Eq. (24)) of the variational posterior distribution, the variational lower bound defined in Eq. (15) is decomposed as follows:

$$\begin{aligned} &\mathcal{F}(m, \mathbf{\Psi}) \\ &\quad = \underbrace{\left\langle \log \frac{p(\mathbf{O}, \mathbf{V} | \mathbf{\Lambda}_{\mathcal{I}_m}) p(\mathbf{\Lambda}_{\mathcal{I}_m})}{\prod_{i \in \mathcal{I}_m} q(\mathbf{W}_i)} \right\rangle_{\substack{\prod_{i \in \mathcal{I}_m} q(\mathbf{W}_i) \\ q(\mathbf{V})}}}_{\triangleq \mathcal{L}(m, \mathbf{\Psi})} \\ &\qquad - \langle \log q(\mathbf{V}) \rangle_{q(\mathbf{V})} . \end{aligned} \tag{46}$$

The second term, which consists of $q(\mathbf{V})$, is an entropy value and is calculated at the E-step in the VB EM algorithm. The first term ($\mathcal{L}(m, \mathbf{\Psi})$) is a logarithmic evidence term for m and $\mathbf{\Psi} = \{\rho_i | i = 1, \cdots |\mathcal{I}_m|\}$ and we can obtain an analytical solution of $\mathcal{L}(m, \mathbf{\Psi})$. Because of the factorization forms in Eqs. (24), (18), and (28), $\mathcal{L}(m, \mathbf{\Psi})$ can be represented as the summation over i, as follows:

$$\mathcal{L}(m, \mathbf{\Psi}) = \sum_{i \in \mathcal{I}_m} \mathcal{L}_i(\rho_i, \rho_{\mathsf{l}(i)}, \rho_{\mathsf{r}(i)}), \tag{47}$$

where

$$\begin{aligned} &\mathcal{L}_i(\rho_i, \rho_{\mathsf{l}(i)}, \rho_{\mathsf{r}(i)}) \\ &\quad \triangleq \sum_{i \in \mathcal{I}_m} \left\langle \log \frac{p(\mathbf{O}, \mathbf{V} | \mathbf{W}_i) p(\mathbf{W}_i | \mathbf{W}_{\mathsf{p}(i)})}{q(\mathbf{W}_i)} \right\rangle_{\substack{q(\mathbf{W}_i) \\ q(\mathbf{V})}} \end{aligned} \tag{48}$$

Note that this factorization form has some dependencies from parent and child node parameters through Eqs. (37) and (39). To derive an analytical solution, we first consider the expectation with respect to only $q(\mathbf{V})$ for cluster i. By substituting Eqs. (8), (21), and (40) into $\mathcal{L}_i(\rho_i, \rho_{\mathsf{l}(i)}, \rho_{\mathsf{r}(i)})$, and by using Eq. (39), the expectation can be rewritten, as follows:

$$\begin{aligned}&\left\langle \log \frac{p(\mathbf{O}, \mathbf{V}|\mathbf{W}_i)p(\mathbf{W}_i|\mathbf{W}_{\mathsf{p}(i)})}{q(\mathbf{W}_i)} \right\rangle_{q(\mathbf{V})} \\ &= \sum_{k \in \mathcal{K}_i} \zeta_k \log g(\mathbf{\Sigma}_k) + \log \frac{g(\hat{\rho}_i^{-1}\mathbf{I}_{D+1})}{g(\tilde{\mathbf{\Omega}}_i)} \\ &- \frac{1}{2}\mathrm{tr}\left[\hat{\rho}_i \hat{\mathbf{M}}_i' \hat{\mathbf{M}}_i - \tilde{\mathbf{M}}_i' \tilde{\mathbf{M}}_i \tilde{\mathbf{\Omega}}_i^{-1} + \sum_{k \in \mathcal{K}_i} \mathbf{\Sigma}_k^{-1}\mathbf{S}_k\right].\end{aligned} \tag{49}$$

The obtained result does not depend on $\mathbf{W}_i$. Therefore, the expectation with respect to $q(\mathbf{W}_i)$ can be disregarded in $\mathcal{L}_i(\rho_i, \rho_{\mathsf{l}(i)}, \rho_{\mathsf{r}(i)})$. Consequently, we can obtain the following analytical result for the lower bound:

$$\begin{aligned}&\mathcal{L}_i(\rho_i, \rho_{\mathsf{l}(i)}, \rho_{\mathsf{r}(i)}) \\ &= -\frac{D}{2}\log(2\pi)\sum_{k \in \mathcal{K}_i} \zeta_k - \frac{1}{2}\sum_{k \in \mathcal{K}_i} \zeta_k \log|\mathbf{\Sigma}_k| \\ &\quad + \frac{D(D+1)}{2}\log \hat{\rho}_i + \frac{D}{2}\log|\tilde{\mathbf{\Omega}}_i| \\ &\quad - \frac{1}{2}\mathrm{tr}\left[\hat{\rho}_i \hat{\mathbf{M}}_i' \hat{\mathbf{M}}_i - \tilde{\mathbf{M}}_i' \tilde{\mathbf{M}}_i \tilde{\mathbf{\Omega}}_i^{-1} + \sum_{k \in \mathcal{K}_i} \mathbf{\Sigma}_k^{-1}\mathbf{S}_k\right].\end{aligned} \tag{50}$$

The first line of the obtained result corresponds to the likelihood value given the amount of data and the covariance matrices of the Gaussians. The other terms consider the effect of the prior and posterior distributions of the model parameters. This is used as an optimization criterion with respect to the model structure m and the hyper-parameters $\mathbf{\Psi}$.

Note that the objective function can be represented as a summation over i because of the factorization form of the prior and posterior distributions. This representation property is used for our model structure optimization in Section 4.2 for a binary tree structure representing a set of Gaussians used in the conventional MLLR.

4 Optimization of hyper-parameters and model structure

In this section, we describe how to optimize hyper-parameters $\mathbf{\Psi}$ and model structure m by using the variational lower bound as an objective function. Once we obtain the variational lower bound, we can obtain an appropriate model structure and hyper-parameters at the same time that maximize the lower bound as follows:

$$\{\widetilde{\mathbf{\Psi}}, \widetilde{m}\} = \underset{m, \mathbf{\Psi}}{\mathrm{argmax}}\, \mathcal{F}(m, \mathbf{\Psi}) \tag{51}$$

In this paper, we use two approximations for the variational lower bound to make the inference algorithm practical. First, we fix latent variables $\mathbf{V}$ during the above optimization. Then, $\langle \log q(\mathbf{V})\rangle_{q(\mathbf{V})}$ in Eq. (46) is also fixed for m and $\mathbf{\Psi}$, and can be disregarded in the objective function. Thus, we can only focus on $\mathcal{L}(m, \mathbf{\Psi})$ in the optimization step, which reduces computational cost greatly, as follows:

$$\{\widetilde{\mathbf{\Psi}}, \widetilde{m}\} \approx \underset{m, \mathbf{\Psi}}{\mathrm{argmax}}\, \mathcal{L}(m, \mathbf{\Psi}) \tag{52}$$

This approximation is widely used in acoustic model selection (likelihood criterion [38] and Bayesian criterion [26]). Second, as we discussed in Section 3.4, the solution for a non-leaf node (Eq. (36)) makes the dependency of the target node with the other linked nodes complex. Therefore, we approximate $\mathcal{L}_i(\rho_i, \rho_{\mathsf{l}(i)}, \rho_{\mathsf{r}(i)}) \approx \mathcal{L}_i(\rho_i)$ by $\hat{\rho}_i \approx \rho_i$ and so on, where $\mathcal{L}_i(\rho_i)$ is defined in the next section. Therefore, in the implementation step, we approximate the posterior distribution for a non-leaf node to that for a leaf node to make an algorithm simple.

4.1 Hyper-parameter optimization

Even though we marginalize all transformation matrix ($\mathbf{W}_i$), we still have to set the precision hyper-parameters ρ_i for all nodes. Since we can derive the variational lower bound, we can optimize the precision hyper-parameter, and can remove the manual tuning of the hyper-parameters with the proposed approach. This is an advantage of the proposed approach with regard to SMAPLR [18], since SMAPLR has to hand-tune its hyper-parameters corresponding to $\{\rho_i\}_i$.

Based on the leaf node approximation for variational posterior distributions, in addition to the fixed latent variable approximation ($\mathcal{F}(m, \mathbf{\Psi}) \approx \mathcal{L}(m, \mathbf{\Psi})$), in this paper the method we implement approximately optimize the precision hyper-parameter as follows:

$$\begin{aligned}\tilde{\rho}_i &= \underset{\rho_i}{\mathrm{argmax}}\, \mathcal{L}(m, \mathbf{\Psi}) \\ &= \begin{cases} \mathrm{argmax}_{\rho_i}\left(\mathcal{L}_i(\rho_i, \rho_{\mathsf{l}(i)}, \rho_{\mathsf{r}(i)}) + \mathcal{L}_{\mathsf{p}(i)}(\rho_{\mathsf{p}(i)}, \rho_i, \rho_{\mathsf{r}(\mathsf{p}(i))})\right) \\ \qquad i \text{ is a left child node of } \mathsf{p}(i) \\ \mathrm{argmax}_{\rho_i}\left(\mathcal{L}_i(\rho_i, \rho_{\mathsf{l}(i)}, \rho_{\mathsf{r}(i)}) + \mathcal{L}_{\mathsf{p}(i)}(\rho_{\mathsf{p}(i)}, \rho_{\mathsf{l}(\mathsf{p}(i))}, \rho_i)\right) \\ \qquad i \text{ is a right child node of } \mathsf{p}(i) \end{cases} \\ &\approx \underset{\rho_i}{\mathrm{argmax}}\, \mathcal{L}_i(\rho_i),\end{aligned}$$

$$\tag{53}$$

where

$$\begin{aligned}\mathcal{L}_i(\rho_i) \triangleq \frac{D(D+1)}{2}\log\rho_i + \frac{D}{2}\log|\tilde{\mathbf{\Omega}}_i| \\ - \frac{1}{2}\mathrm{tr}\left[\rho_i \mathbf{M}'_{\mathsf{p}(i)}\mathbf{M}_{\mathsf{p}(i)} - \tilde{\mathbf{M}}'_i\tilde{\mathbf{M}}_i\tilde{\mathbf{\Omega}}_i^{-1}\right].\end{aligned} \tag{54}$$

This approximation makes the algorithm simple because we can optimize the precision hyper-parameter within the target and parent nodes, and do not have to consider the child nodes. Since we only have one scalar parameter for this optimization step, we simply used a line search algorithm to obtain the optimal precision hyper-parameter. If we consider a more complex precision structure (e.g., a precision matrix instead of a scalar precision parameter in the prior distribution setting Eq. (20)), the line search algorithm may not be adequate. In that case, we need to update hyper-parameters by using some other optimization technique (e.g., gradient decent).

4.2 Model selection

The remaining tuning parameter in the proposed approach is how many clusters we prepare. This is a model selection problem, and we can also automatically obtain the number of clusters by optimizing the variational lower bound. In the binary tree structure, we focus on a subtree composed of a target non-leaf node i and its child nodes $\mathsf{l}(i)$ and $\mathsf{r}(i)$. We compute the following difference based on Eq. (54) of the parent and that of the child nodes[9]

$$\Delta\mathcal{L}_i(\rho_i) \triangleq \mathcal{L}_{\mathsf{l}(i)}(\rho_{\mathsf{l}(i)}) + \mathcal{L}_{\mathsf{r}(i)}(\rho_{\mathsf{r}(i)}) - \mathcal{L}_i(\rho_i). \tag{55}$$

This difference function is used for a stopping criterion in a top-down clustering strategy. Then, if the sign of $\Delta\mathcal{L}$ is negative, the target non-leaf node is regarded as a new leaf node determined by the model selection in terms of optimizing the lower bound. Then, we prune the child nodes $\mathsf{l}(i)$ and $\mathsf{r}(i)$. By checking the signs of $\Delta\mathcal{L}_i$ for all possible nodes, and pruning the child nodes when $\Delta\mathcal{L}_i$ have negative signs, we can obtain the pruned tree structure, which corresponds to maximizing the variational lower bound locally. This optimization is efficiently accomplished by using a depth-first search.

Algorithm 1 Structural Bayesian linear regression.

```
Prepare an initial Gaussian tree with a set of nodes $\mathcal{I}$
Initialize $\tilde{\mathbf{\Psi}} = \{\tilde{\rho}_i, \tilde{\mathbf{M}}_i | i = 1, \cdots |\mathcal{I}|\}$
repeat
    VB E-step
    $\mathcal{L}(m, \mathbf{\Phi})$ = Prune_tree(root node) // prune a tree by model selection
    # of leaf nodes = Transform_HMM(root node) // Transform HMMs in the pruned tree
until Total lower bound is converged or a specified number of iterations has been reached.
```

Algorithm 2 Prune_tree(node i)

```
if First iteration then
    $\tilde{\rho}_i = \mathrm{argmax}_{\rho_i} \mathcal{L}_i(\rho_i)$ // These are used as
    Update $\tilde{q}(\mathbf{W}_i)$ // hyper-parameters of parent nodes
end if
if Node $i$ has child nodes then
    $\tilde{\rho}_i = \mathrm{argmax}_{\rho_i} \mathcal{L}_i(\rho_i)$
    Update $\tilde{q}(\mathbf{W}_i)$
    $\Delta\mathcal{L}$ = Prune_tree(node $left(i)$) + Prune_tree(node $right(i)$) $-\mathcal{L}_i(\tilde{\rho}_i)$
    if $\Delta\mathcal{L} < 0$ then
        Prune child nodes // this node becomes a leaf node
    end if
    return $\mathcal{L}_i(\tilde{\rho}_i)$
else
    $\tilde{\rho}_i = \mathrm{argmax}_{\rho_i} \mathcal{L}_i(\rho_i)$
    Update $q(\mathbf{W}_i)$
    return $\mathcal{L}_i(\tilde{\rho}_i)$
end if
```

Algorithm 3 Transform_HMM(node i)

```
if Node $i$ has child nodes then
    return Transform_HMM(node $left(i)$) + Transform_HMM(node $right(i)$)
else
    Update $\tilde{\boldsymbol{\mu}}_k = C_k \tilde{\mathbf{M}}_i \boldsymbol{\xi}_k$
    return 1
end if
```

This approach is similar to the tree-based triphone clustering based on VB [26].

Thus, by optimizing the hyper-parameters and model structure, we can avoid setting any tuning parameters. We summarize this optimization in Algorithm 1, 2, and 3. Algorithm 1 prepares a large Gaussian tree with a set of nodes $\mathcal{I}$, prunes a tree based on the model selection (Algorithm 2), and transforms HMMs (Algorithm 3). Algorithm 2 first optimizes the precision hyper-parameters $\mathbf{\Psi}$, and then the model structure m. Algorithm 3 transforms Gaussian mean vectors in HMMs at the new root nodes in the pruned tree $\mathcal{I}_m$ obtained by Algorithm 2.

[9] Since we approximate the posterior distribution for a non-leaf node to that for a leaf node, the contribution of the variational lower bounds from the non-leaf nodes to the total lower bounds can be disregarded, and Eq. (55) is used as a pruning criterion. If we don't use this approximation, we just compare the difference between the values $\mathcal{L}_i(\rho_i, \rho_{\mathsf{l}(i)}, \rho_{\mathsf{r}(i)})$ of the leaf and non-leaf node cases in Eq. (50).

Table 2 Experimental setup for CSJ

Sampling rate	16 kHz
Feature type	MFCC + Energy $+\Delta + \Delta\Delta$ (39 dim.)
Frame length	25 ms
Frame shift	10 ms
Window type	Hamming
# of categories	43 phonemes
Context-dependent	5,000 HMM states
HMM topology	(3-state left to right) 32 GMM components
Training method	Discriminative training (MCE)
Language model	3-gram (Good-Turing smoothing)
Vocabulary size	100,808
Perplexity	82.4
OOV rate	2.3 %

5 Experiments

This section shows the effectiveness of the proposed approach through experiments with large vocabulary continuous speech recognition. We used a Corpus of Spontaneous Japanese (CSJ) task [39].

5.1 Experimental condition

The training data for constructing the initial (non-adapted) acoustic model consists of 961 talks from the CSJ conference presentations (234 hours of speech data), and the training data for the language model construction consists of 2,672 talks from the complete CSJ speech data (6.8M word transcriptions). The test set consists of 10 talks (2.4 hours, 26,798 words). Table 2 provides information on acoustic and language models used in the experiments [40]. We used context-dependent models with continuous density HMMs. The HMM parameters were estimated based on a discriminative training (Minimum Classification Error: MCE) approach [41]. Lexical and language models were also obtained by employing all the CSJ speech data. We used a 3-gram model with a Good-Turing smoothing technique. The OOV rate was 2.3 % and the test set perplexity was 82.4. The acoustic model construction, LVCSR decoding, and the following acoustic model adaptation procedures were performed with the NTT speech recognition platform SOLON [42].

5.2 Experimental result

To check whether the proposed approach steadily increase the variational lower bound for each optimization in Section 4, Figure 3 examines the values of the variational lower bound for each condition. Namely, we compare the proposed approach that optimizes both model structure and hyper-parameters, as discussed in Section 4 with those did not optimize each or any of them, in terms of the $\mathcal{L}(m, \mathbf{\Psi})$ value. Figure 3 shows that the proposed approach indeed steadily increases the $\mathcal{L}(m, \mathbf{\Psi})$ value. Therefore, this result indicates that the optimization works well by obtaining appropriate hyper-parameters and model structure.

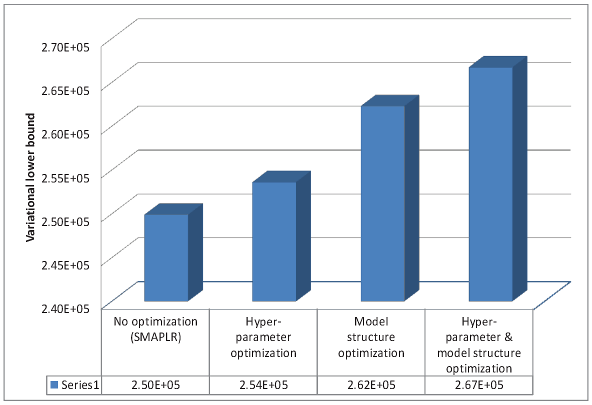

Fig. 3 Variational lower bound for each optimization.

Next, Figure 4 compares the proposed approach with MLLR based on the maximum likelihood estimation, and SMAPLR based on the approximate Bayesian estimation, as regards the Word Error Rate (WER) for various amounts of adaptation data. With a small amount of adaptation data, the proposed approach outperforms the conventional approaches by about 1.0 %, while with a large amount of adaptation data, the accuracies of all approaches are comparable. This property is theoretically reasonable since the variational lower bound would be tighter than the EM-based objective function for a small amount of data, while would approach it for a large amount of data asymptotically. Therefore, we conclude that this improvement comes from the optimization of the hyper-parameters and the model structure of the proposed approach, in addition to the mitigation of sparse data problem based on the Bayesian approach.

Thus, from the values of the lower bound and the recognition result, we show the effectiveness of the proposed approach.

6 Summary and future work

This paper presents a fully Bayesian treatment of linear regression for HMMs by using variational techniques. The derived lower bound of the marginalized log-likelihood

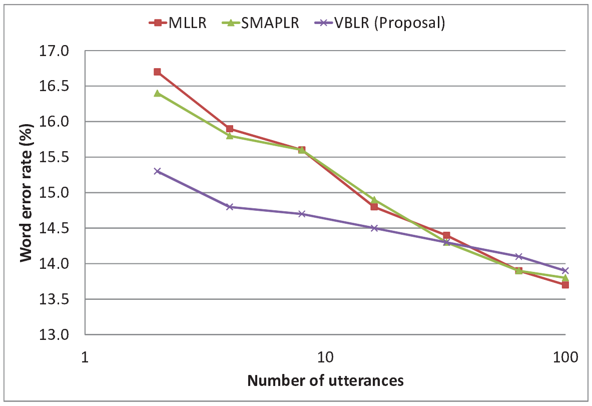

Fig. 4 Word error rates of conventional MLLR, SMAPLR, and the proposed Bayesian Linear Regression (VBLR) for various amounts (utterances) of adaptation data. The word error rate of the non-adapted (speaker independent) model was 17.9%.

can be used for optimizing the hyper-parameters and model structure, which was confirmed by speech recognition experiments. One promising extension is to apply the proposed approach to advanced adaptation techniques. Actually, [43, 44] provide a fully Bayesian solution for standard transformation parameters (not variance normalize representation in this paper), and apply it to both the feature space and model parameter transformations. The model structure and hyper-parameters are also optimized automatically during adaptation. Thus, feature space normalization and model space adaptation are consistently performed based on a variational Bayesian approach without tuning any parameters.

Another important plan for the future work is joint optimization of HMM parameters and linear regression parameters in a Bayesian framework. This paper assumes that the HMM parameters are fixed during the estimation process of linear regression parameters. These parameters depend on each other, and the variational approximation can deal with the problem (in the sense of local optimum solutions). However, to consider the model selection in this joint optimization, we have to think of many possible combinations of HMM and linear regression topologies. One promising approach to this problem is to consider a non-parametric Bayesian approach (e.g., variational inference for Dirichlet process mixtures [45] in the VB framework), which can efficiently search an appropriate model structure in the many possible combinations.

Finally, how to integrate Bayesian approaches with discriminative approaches theoretically and practically is also important future work. One promising approach for this direction is the marginalization of model parameters and margin variables to provide Bayesian interpretations with discriminative methods [46]. However applying [46] to acoustic models requires some extensions to deal with large-scale structured data problems [47]. This extension enables the more robust regularization of discriminative approaches, and allows structural learning by combining Bayesian and discriminative criteria.

References

1. C. J. Leggetter and P. C. Woodland. Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models. *Computer Speech and Language*, 9:171–185, 1995.
2. V. Digalakis, D. Ritischev, and L. Neumeyer. Speaker adaptation using constrained reestimation of Gaussian mixtures. *IEEE Transactions on Speech and Audio Processing*, 3:357–366, 1995.
3. C.-H. Lee and Q. Huo. On adaptive decision rules and decision parameter adaptation for automatic speech recognition. In *Proceedings of the IEEE*, volume 88, pages 1241–1269, 2000.
4. Koichi Shinoda. Acoustic model adaptation for speech recognition. *IEICE transactions on information and systems*, 93(9):2348–2362, 2010.
5. Ananth Sankar and Chin-Hui Lee. A maximum-likelihood approach to stochastic matching for robust speech recognition. *Speech and Audio Processing, IEEE Transactions on*, 4(3):190–202, 1996.
6. Jen-Tzung Chien, Chin-Hui Lee, and Hsiao-Chuan Wang. Improved bayesian learning of hidden markov models for speaker adaptation. In *Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE International Conference on*, volume 2, pages 1027–1030. IEEE, 1997.
7. Kuan-ting Chen, Wen-wei Liau, Hsin-min Wang, and Lin-shan Lee. Fast speaker adaptation using eigenspace-based maximum likelihood linear regression. In *Proc. ICSLP*, volume 3, pages 742–745, 2000.
8. B. Mak, J.T. Kwok, and S. Ho. Kernel eigenvoice speaker adaptation. *Speech and Audio Processing, IEEE Transactions on*, 13(5):984–992, 2005.
9. Marc Delcroix, Tomohiro Nakatani, and Shinji Watanabe. Static and dynamic variance compensation for recognition of reverberant speech with dereverberation preprocessing. *Audio, Speech, and Language Processing, IEEE Transactions on*, 17(2):324–334, 2009.
10. M. Tamura, T. Masuko, K. Tokuda, and T. Kobayashi. Adaptation of pitch and spectrum for HMM-based speech synthesis using MLLR. In *ICASSP'01*, volume 2, pages 805–808, 2001.
11. A. Stolcke, L. Ferrer, S. Kajarekar, E. Shriberg, and A. Venkataraman. MLLR transforms as features in speaker recognition. In *Interspeech'05*, pages 2425–2428, 2005.
12. C. Sanderson, S. Bengio, and Y. Gao. On transforming statistical models for non-frontal face verification. *Pattern Recognition*, 39(2):288–302, 2006.
13. T. Maekawa and S Watanabe. Unsupervised Activity Recognition with User's Physical Characteristics Data. In *Proc. of International Symposium on Wearable Computers (ISWC 2011)*, pages 89–96, 2011.
14. S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, et al.

The HTK book (for HTK version 3.4). *Cambridge University Engineering Department*, 2006.

15. C. Chesta, O. Siohan, and C.-H. Lee. Maximum a posteriori linear regression for hidden Markov model adaptation. In *Proc. Eurospeech1999*, volume 1, pages 211–214, 1999.
16. Jen-Tzung Chien. Quasi-Bayes linear regression for sequential learning of hidden Markov models. *Speech and Audio Processing, IEEE Transactions on*, 10(5):268–278, 2002.
17. K. Shinoda and C.-H. Lee. A structural Bayes approach to speaker adaptation. *IEEE Transactions on Speech and Audio Processing*, 9:276–287, 2001.
18. O. Siohan, T.A. Myrvoll, and C.H. Lee. Structural maximum a posteriori linear regression for fast HMM adaptation. *Computer Speech & Language*, 16(1):5–24, 2002.
19. D.J.C. MacKay. Ensemble learning for hidden Markov models. Technical report, Technical report, Cavendish Laboratory, University of Cambridge, 1997.
20. R.M. Neal and G.E. Hinton. A view of the EM algorithm that justifies incremental, sparse, and other variants. *Learning in Graphical Models*, pages 355–368, 1998.
21. M.I. Jordan, Z. Ghahramani, T.S. Jaakkola, and L.K. Saul. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.
22. H. Attias. Inferring parameters and structure of latent variable models by variational Bayes. In *Proc. Uncertainty in Artificial Intelligence (UAI) 15*, 1999.
23. N. Ueda and Z. Ghahramani. Bayesian model search for mixture models based on optimizing variational bounds. *Neural Networks*, 15:1223–1241, 2002.
24. S. Watanabe, Y. Minami, A. Nakamura, and N. Ueda. *Application of variational Bayesian approach to speech recognition*. NIPS 2002, MIT Press, 2002.
25. F. Valente and C. Wellekens. Variational Bayesian GMM for speech recognition. In *Proc. Eurospeech2003*, pages 441–444, 2003.
26. S. Watanabe, Y. Minami, A. Nakamura, and N. Ueda. Variational Bayesian estimation and clustering for speech recognition. *IEEE Transactions on Speech and Audio Processing*, 12:365–381, 2004.
27. P. Somervuo. Comparison of ML, MAP, and VB based acoustic models in large vocabulary speech recognition. In *Proc. ICSLP2004*, volume 1, pages 830–833, 2004.
28. T. Jitsuhiro and S. Nakamura. Automatic generation of non-uniform HMM structures based on variational Bayesian approach. In *Proc. ICASSP2004*, volume 1, pages 805–808, 2004.
29. K. Hashimoto, H. Zen, Y. Nankaku, A. Lee, and K. Tokuda. Bayesian context clustering using cross valid prior distribution for HMM-based speech recognition. In *Proc. Interspeech'08*, 2008.
30. A. Ogawa and S. Takahashi. Weighted distance measures for efficient reduction of Gaussian mixture components in HMM-based acoustic model. In *Proc. ICASSP'08*, pages 4173–4176, 2008.
31. N. Ding and Z. Ou. Variational nonparametric Bayesian hidden Markov model. In *Proc. ICASSP'10*, pages 2098–2101, 2010.
32. S. Watanabe and A. Nakamura. Acoustic model adaptation based on coarse/fine training of transfer vectors and its application to a speaker adaptation task. In *Proc. ICSLP*, pages 2933–2936, 2004.
33. K. Yu and M. J. F. Gales. Incremental adaptation using Bayesian inference. In *Proceedings of IEEE International Conference on Acoustics, Speech, & Signal Processing (ICASSP 2006)*, volume 1, pages 217–220, 2006.
34. J. Winn and C.M. Bishop. Variational message passing. *Journal of Machine Learning Research*, 6(1):661, 2006.
35. M. J. F. Gales and P. C. Woodland. Variance compensation within the MLLR framework. Technical Report 242, Cambridge University Engineering Department, 1996.
36. A Philip Dawid. Some matrix-variate distribution theory: notational considerations and a Bayesian application. *Biometrika*, 68(1):265–274, 1981.
37. S. Watanabe, A. Nakamura, and B.H. Juang. Bayesian linear regression for hidden Markov model based on optimizing variational bounds. In *Proc. MLSP 2011*, pages 1–6, 2011.
38. J. J. Odell. *The use of context in large vocabulary speech recognition*. PhD thesis, Cambridge University, 1995.
39. K. Maekawa, H. Koiso, S. Furui, and H. Isahara. Spontaneous speech corpus of Japanese. In *Proceedings of LREC2000*, volume 2, pages 947–952, 2000.
40. A. Nakamura, T. Oba, S. Watanabe, K. Ishizuka, M. Fujimoto, T. Hori, E. McDermott, and Y. Minami. Evaluation of the SOLON speech recognition system : 2006 benchmark using the Corpus of Spontaneous Japanese. *IPSJ SIG Notes*, 2006(136):251–256, 2006. (in Japanese).
41. E. McDermott, T.J. Hazen, J. Le Roux, A. Nakamura, and S. Katagiri. Discriminative training for large-vocabulary speech recognition using minimum classification error. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(1):203–223, 2007.
42. T. Hori. NTT Speech recognizer with OutLook On the Next generation: SOLON. In *Proc. NTT Workshop on Communication Scene Analysis*, volume 1, SP-6, 2004.
43. S. J. Hahm, A. Ogawa, M. Fujimoto, T. Hori, and A. Nakamura. Speaker adaptation using variational Bayesian linear regression in normalized feature space. In *Proc. of Interspeech'12*, 2012.
44. S. J. Hahm, A. Ogawa, M. Fujimoto, T. Hori, and A. Nakamura. Feature space variational Bayesian linear regression and its combination with model space VBLR. In *Proc. of ICASSP'13*, 2013.
45. D.M. Blei and M.I. Jordan. Variational inference for Dirichlet process mixtures. *Bayesian Analysis*, 1(1):121–144, 2006.
46. T. Jebara. *Machine learning: discriminative and generative*, volume 755. Springer, 2004.
47. Y. Kubo, S. Watanabe, A. Nakamura, and T. Kobayashi. A regularized discriminative training method of acoustic models derived by minimum relative entropy discrimination. In *Proc. Interspeech 2010*, pages 2954–2957, 2010.

A Derivation of posterior distribution of latent variables

This section derives the posterior distribution of latent variables $\tilde{q}(\mathbf{V}_i)$, introduced in Section 3.5, based on the VB framework. To obtain VB posteriors of latent variables, we consider the following integral (this is the same equation as Eq. (43)).

$$\int \tilde{q}(\mathbf{W}_i) \log \mathcal{N}(\mathbf{o}_t | \mathbf{C}_k \mathbf{W}_i \boldsymbol{\xi}_k, \boldsymbol{\Sigma}_k) d\mathbf{W}_i \tag{A.1}$$

In this derivation, we omit indexes i, k, and t for simplicity. By substituting the concrete form (Eq. (4)) of the multivariate Gaussian distribution into Eq. (A.1), the equation is represented as:

$$\begin{aligned}\text{Eq. (A.1)} &= -\frac{D}{2}\log(2\pi|\mathbf{\Sigma}_k|)\\ &\quad -\frac{1}{2}\int \tilde{q}(\mathbf{W})\underbrace{\left((\mathbf{o}-\mathbf{C}\mathbf{W}\boldsymbol{\xi})'(\mathbf{\Sigma})^{-1}(\mathbf{o}-\mathbf{C}\mathbf{W}\boldsymbol{\xi})\right)}_{(*1)}d\mathbf{W}\end{aligned}\tag{A.2}$$

where we use

$$\int \tilde{q}(\mathbf{W})d\mathbf{W} = 1.\tag{A.3}$$

Now, we focus on the quadratic form $(*1)$ of the third line of Eq. (A.2). By considering $\mathbf{\Sigma} = \mathbf{C}(\mathbf{C})'$ in Eq. (6), $(*1)$ is rewritten as follows:

$$\begin{aligned}(*1) &= ((\mathbf{C})^{-1}\mathbf{o}-\mathbf{W}\boldsymbol{\xi})'((\mathbf{C})^{-1}\mathbf{o}-\mathbf{W}\boldsymbol{\xi})\\ &= \mathrm{tr}\left[((\mathbf{C})^{-1}\mathbf{o}-\mathbf{W}\boldsymbol{\xi})((\mathbf{C})^{-1}\mathbf{o}-\mathbf{W}\boldsymbol{\xi})'\right]\\ &= \mathrm{tr}\left[\boldsymbol{\varGamma}\mathbf{W}'\mathbf{W}-2\mathbf{W}\mathbf{Y}'+\mathbf{U}\right]\end{aligned}\tag{A.4}$$

where we use the cyclic and transpose properties of the trace, as follows:

$$\mathrm{tr}[\mathbf{A}_1\mathbf{A}_2\mathbf{A}_3] = \mathrm{tr}[\mathbf{A}_2\mathbf{A}_3\mathbf{A}_1])\tag{A.5}$$

$$\mathrm{tr}[\mathbf{A}'] = \mathrm{tr}[\mathbf{A}]\tag{A.6}$$

We also define $(D+1)\times(D+1)$ matrix $\boldsymbol{\varGamma}$, $D\times(D+1)$ matrix $\mathbf{Y}$, and $D\times D$ matrix $\mathbf{U}$ in Eq. (A.4) as follows:

$$\begin{aligned}&\boldsymbol{\varGamma} \triangleq \boldsymbol{\xi}\boldsymbol{\xi}'\\ &\mathbf{Y} \triangleq (\mathbf{C})^{-1}\mathbf{o}\boldsymbol{\xi}'\\ &\mathbf{U} \triangleq (\mathbf{\Sigma})^{-1}\mathbf{o}\mathbf{o}'\end{aligned}\tag{A.7}$$

The integral of Eq. (A.4) over $\mathbf{W}$ can be decomposed into the following three terms:

$$\begin{aligned}&\int \tilde{q}(\mathbf{W})\mathrm{tr}\left[\boldsymbol{\varGamma}\mathbf{W}'\mathbf{W}-2\mathbf{W}\mathbf{Y}'+\mathbf{U}\right]d\mathbf{W}\\ &= \underbrace{\int \tilde{q}(\mathbf{W})\mathrm{tr}\left[\boldsymbol{\varGamma}\mathbf{W}'\mathbf{W}\right]d\mathbf{W}}_{(*2)} -2\underbrace{\int \tilde{q}(\mathbf{W})\mathrm{tr}\left[\mathbf{W}\mathbf{Y}'\right]d\mathbf{W}}_{(*3)} +\mathrm{tr}\left[\mathbf{U}\right]\end{aligned}\tag{A.8}$$

where we use the following property:

$$\mathrm{tr}[\mathbf{A}_1+\mathbf{A}_2] = \mathrm{tr}[\mathbf{A}_1]+\mathrm{tr}[\mathbf{A}_3]\tag{A.9}$$

and use Eq. (A.3) in the third term of the second line in Eq. (A.8).

We focus on the integrals $(*2)$ and $(*3)$. Since $\tilde{q}(\mathbf{W})$ is a scalar value, $(*3)$ is rewritten as follows:

$$\begin{aligned}(*3) &= \int \mathrm{tr}\left[\tilde{q}(\mathbf{W})\mathbf{W}\mathbf{Y}'\right]d\mathbf{W}\\ &= \mathrm{tr}\left[\int \tilde{q}(\mathbf{W})\mathbf{W}\mathbf{Y}'d\mathbf{W}\right].\end{aligned}\tag{A.10}$$

Here, we use the following matrix properties:

$$\mathrm{tr}[a\mathbf{A}] = a\ \mathrm{tr}[\mathbf{A}]\tag{A.11}$$

$$\int \mathrm{tr}[f(\mathbf{A})]d\mathbf{A} = \mathrm{tr}\left[\int f(\mathbf{A})d\mathbf{A}\right]\tag{A.12}$$

Thus, the integral is finally solved as

$$\begin{aligned}(*3) &= \mathrm{tr}\left[\left(\int \tilde{q}(\mathbf{W})\mathbf{W}d\mathbf{W}\right)\mathbf{Y}'\right]\\ &= \mathrm{tr}\left[\tilde{\mathbf{M}}\mathbf{Y}'\right]\end{aligned}\tag{A.13}$$

where we use

$$\int \tilde{q}(\mathbf{W})\mathbf{W}d\mathbf{W} = \tilde{\mathbf{M}}.\tag{A.14}$$

Similarly, we also rewrite $(*2)$ in Eq. (A.8) based on Eqs. (A.11) and (A.12), as follows:

$$\begin{aligned}(*2) &= \int \mathrm{tr}\left[\tilde{q}(\mathbf{W})\boldsymbol{\varGamma}\mathbf{W}'\mathbf{W}\right]d\mathbf{W}\\ &= \mathrm{tr}\left[\int \tilde{q}(\mathbf{W})\boldsymbol{\varGamma}\mathbf{W}'\mathbf{W}d\mathbf{W}\right]\\ &= \mathrm{tr}\left[\boldsymbol{\varGamma}\int \tilde{q}(\mathbf{W})\mathbf{W}'\mathbf{W}d\mathbf{W}\right].\end{aligned}\tag{A.15}$$

Thus, the integral is finally solved as

$$(*2) = \mathrm{tr}\left[\boldsymbol{\varGamma}\left(\tilde{\mathbf{\Omega}}+\tilde{\mathbf{M}}'\tilde{\mathbf{M}}\right)\right],\tag{A.16}$$

where we use

$$\int \tilde{q}(\mathbf{W})\mathbf{W}'\mathbf{W}d\mathbf{W} = \tilde{\mathbf{\Omega}}+\tilde{\mathbf{M}}'\tilde{\mathbf{M}}.\tag{A.17}$$

Thus, we solve the all integrals in Eq. (A.8).

Finally, we substitute the integral results of $(*2)$ and $(*3)$ (i.e., Eqs. (A.16) and (A.16)) into Eq. (A.8), and rewrite Eq. (A.8) based on the concrete forms of $\boldsymbol{\varGamma}$, $\mathbf{Y}$, and $\mathbf{U}$ defined in Eq. (A.7) as follows:

$$\begin{aligned}&\text{Eq. (A.8)}\\ &= \mathrm{tr}\left[\boldsymbol{\varGamma}\left(\tilde{\mathbf{\Omega}}+\tilde{\mathbf{M}}'\tilde{\mathbf{M}}\right)-2\tilde{\mathbf{M}}\mathbf{Y}'+\mathbf{U}\right]\\ &= \mathrm{tr}\left[\boldsymbol{\xi}\boldsymbol{\xi}'(\tilde{\mathbf{\Omega}}+\tilde{\mathbf{M}}'\tilde{\mathbf{M}})-2\tilde{\mathbf{M}}\boldsymbol{\xi}\mathbf{o}'((\mathbf{C})^{-1})'+(\mathbf{\Sigma})^{-1}\mathbf{o}\mathbf{o}'\right]\end{aligned}\tag{A.18}$$

Then, by using the cyclic property in Eq. (A.5) and $\mathbf{\Sigma} = \mathbf{C}(\mathbf{C})'$ in Eq. (6), we can further rewrite Eq. (A.8) as follows:

$$\begin{aligned}&\text{Eq. (A.8)}\\ &= \mathrm{tr}\left[\boldsymbol{\xi}\boldsymbol{\xi}'\tilde{\mathbf{\Omega}}+(\mathbf{\Sigma})^{-1}\left(\mathbf{\Sigma}\tilde{\mathbf{M}}\boldsymbol{\xi}\boldsymbol{\xi}'\tilde{\mathbf{M}}'-2\mathbf{C}\tilde{\mathbf{M}}\boldsymbol{\xi}\mathbf{o}'+\mathbf{o}\mathbf{o}'\right)\right]\\ &= \mathrm{tr}\left[\boldsymbol{\xi}\boldsymbol{\xi}'\tilde{\mathbf{\Omega}}+(\mathbf{\Sigma})^{-1}\left(\mathbf{o}-\mathbf{C}\tilde{\mathbf{M}}\boldsymbol{\xi}\right)\left(\mathbf{o}-\mathbf{C}\tilde{\mathbf{M}}\boldsymbol{\xi}\right)'\right]\end{aligned}\tag{A.19}$$

Thus, we obtain the quadratic form with respect to $\mathbf{o}$, which becomes a multivariate Gaussian distribution form. By recovering the omitted indexes i, k, and t, and substituting integral result in Eq. (A.19) into Eq. (A.2), we finally solve Eq. (43) as:

$$\begin{aligned}&\int \tilde{q}(\mathbf{W}_i)\log\mathcal{N}(\mathbf{o}_t|\mathbf{C}_k\mathbf{W}_i\boldsymbol{\xi}_k,\mathbf{\Sigma}_k)d\mathbf{W}_i\\ &= -\frac{D}{2}\log(2\pi|\mathbf{\Sigma}_k|)\\ &\quad -\frac{1}{2}\mathrm{tr}\left[\boldsymbol{\xi}\boldsymbol{\xi}'\tilde{\mathbf{\Omega}}+(\mathbf{\Sigma})^{-1}\left(\mathbf{o}-\mathbf{C}\tilde{\mathbf{M}}\boldsymbol{\xi}\right)\left(\mathbf{o}-\mathbf{C}\tilde{\mathbf{M}}\boldsymbol{\xi}\right)'\right]\\ &= \log\mathcal{N}(\mathbf{o}_t|\mathbf{C}_k\tilde{\mathbf{M}}_i\boldsymbol{\xi}_k,\mathbf{\Sigma}_k)-\frac{1}{2}\mathrm{tr}\left[\boldsymbol{\xi}_k\boldsymbol{\xi}_k'\tilde{\mathbf{\Omega}}_i\right].\end{aligned}\tag{A.20}$$

Here, we use the concrete form of the multivariate Gaussian distribution in Eq. (4).