A Variational Perspective on Noise-Robust Speech Recognition

R. C. van Dalen, M. J. F. Gales

Department of Engineering, Cambridge University
Cambridge, UK
{rcv25,mjfg}@eng.cam.ac.uk

***Abstract*—Model compensation methods for noise-robust speech recognition have shown good performance. Predictive linear transformations can approximate these methods to balance computational complexity and compensation accuracy. This paper examines both of these approaches from a variational perspective. Using a matched-pair approximation at the component level yields a number of standard forms of model compensation and predictive linear transformations. However, a tighter bound can be obtained by using variational approximations at the state level. Both model-based and predictive linear transform schemes can be implemented in this framework. Preliminary results show that the tighter bound obtained from the state-level variational approach can yield improved performance over standard schemes.**

I. Introduction

One way of making speech recognisers more robust to noise is *model compensation*. Rather than enhancing the incoming observations, model compensation techniques modify the HMM's state-conditional distributions so they model the speech in the target environment. Because the interaction between speech and noise is non-linear, the corrupted-speech distribution has no closed form. In particular, even if the speech and noise distributions are Gaussian, the corrupted speech is not. However, the majority of schemes assume the corrupted speech Gaussian-distributed. Additionally, despite correlation changes, the covariance matrices are normally diagonalised. Thus, improvements from different ways of computing the same form of distribution are limited (see e.g. [1]).

Two approaches that remove the Gaussian assumption have been proposed. The "Algonquin" algorithm [2] still uses a Gaussian approximation, but tuned for each clean speech component and observation vector. Thus, the effective distribution over observation vectors is non-Gaussian. Another approach [3] uses a non-parametric distribution, by approximating the integral in the corrupted-speech likelihood expression with a sampling scheme. However, both these approaches require extensive computation for each incoming feature vector. Thus, for most situations they are impractical.

The aim of this work is to find forms of fast likelihood computation that approximate these non-Gaussian distributions. In this work, the framework of *predictive methods* is applied to this problem. Predictive methods approximate a complicated model with a simpler form by minimising the KL divergence between them [4]. How this applies to noise-robust speech recognition can be seen in Fig. 1. The left graphical model represents noise-corrupted speech. The speech is governed by a hidden Markov model, with states θ_t. Associated with each state is a mixture of Gaussians with component indicator k_t and Gaussian component distributions generating speech vectors $\mathbf{x}_t$. The independently and identically distributed noise $\mathbf{n}_t$ corrupts the speech, resulting in noise-corrupted speech $\mathbf{y}_t$.

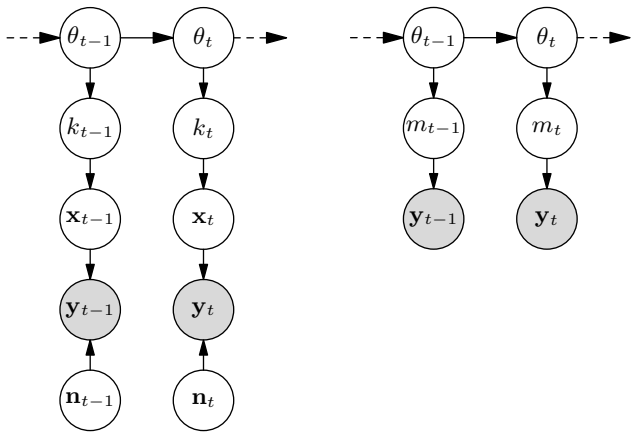

Fig. 1. The noise-corrupted speech p (left), with components k, speech $\mathbf{x}$, noise $\mathbf{n}$, corrupted speech $\mathbf{y}$; its approximation q (right) with components m.

The right graphical model in Fig. 1 approximates the corrupted speech distribution. The Markov model $p(\theta_{t-1}|\theta_t)$, which governs the clean speech, is assumed unchanged. Many compensation methods map each component k of the predicted distribution onto one component m of the approximation and marginalise out only $\mathbf{x}$ and $\mathbf{n}$. This is not a problem if the marginalisation is exact; however, usually each component is approximated with a Gaussian [5], [1], [6], [7]. This *matched-pair approximation* to the KL divergence neglects the potential for a mixture of Gaussians to represent a mixture of non-Gaussian distributions. To approximate a whole state-conditional distribution at once, this paper will sever the hard link between components in the left- and right-hand models. Instead, a variational approach can assign part of the probability mass of component k in the left-hand model to any component m in the right-hand model. This tightens an upper bound on the KL divergence. Within this variational framework, other forms of mixture models can also be estimated. This paper will introduce a new form of predictive CMLLR, which transforms clean speech Gaussians with shared linear transformations. This yields optimal linear transformations for modelling the non-linear effects of the noise.

II. The Influence of Noise on Speech

To make speech recognisers robust to noise, a model of how the noise influences the signal is required. Both noise and speech vectors are usually composed of Mel-frequency cepstral coefficients (MFCCs) and deltas and delta-deltas. The relation between the MFCC vectors of the speech $\mathbf{x}^{\mathsf{s}}$, additive noise $\mathbf{n}^{\mathsf{s}}$, and convolutional (channel) noise $\mathbf{h}^{\mathsf{s}}$ (which is usually assumed fixed, and not shown in Fig. 1), and the corrupted speech $\mathbf{y}^{\mathsf{s}}$, called the mismatch function, is [8], [9]

$$\mathbf{y}^{\mathsf{s}} = \mathbf{f}(\mathbf{x}^{\mathsf{s}}, \mathbf{n}^{\mathsf{s}}, \mathbf{h}^{\mathsf{s}}, \boldsymbol{\alpha}) \triangleq \mathbf{C}\mathbf{log}\Big(\mathbf{exp}\big(\mathbf{C}^{-1}(\mathbf{x}^{\mathsf{s}} + \mathbf{h}^{\mathsf{s}})\big) + \mathbf{exp}\big(\mathbf{C}^{-1}\mathbf{n}^{\mathsf{s}}\big) + 2\boldsymbol{\alpha} \circ \mathbf{exp}\big(\tfrac{1}{2}\mathbf{C}^{-1}\big(\mathbf{x}^{\mathsf{s}} + \mathbf{h}^{\mathsf{s}} + \mathbf{n}^{\mathsf{s}}\big)\big)\Big). \quad (1)$$

Here, $\mathbf{log}(\cdot)$, $\mathbf{exp}(\cdot)$, and $\circ$ denote element-wise logarithm, exponentiation, and multiplication. $\mathbf{C}$ is the truncated DCT matrix and $\mathbf{C}^{-1}$ its pseudo-inverse. $\boldsymbol{\alpha}$ is the *phase factor*, which arises from the interaction between the phase of the speech and noise signal. Since the phase information is discarded to arrive at MFCCs, the effect of this interaction, represented by $\boldsymbol{\alpha}$, is random even if $\mathbf{x}^{\mathsf{s}}, \mathbf{n}^{\mathsf{s}}, \mathbf{h}^{\mathsf{s}}$ are fixed. In this work, as in [3], $\boldsymbol{\alpha}$ is modelled with an independent and identically distributed truncated Gaussian with variances computed according to [10].

Speech recognisers append dynamic (delta and delta-delta) features to the MFCC feature vector. These are computed from a window of per-time slice, static, features. For exposition, assume a window ± 1 and only first-order dynamics. The observation vector $\mathbf{y}_t$ is then computed as

$$\mathbf{y}_t = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ -\frac{\mathbf{I}}{2} & \mathbf{0} & \frac{\mathbf{I}}{2} \end{bmatrix} \begin{bmatrix} \mathbf{y}_{t-1} \\ \mathbf{y}_t \\ \mathbf{y}_{t+1} \end{bmatrix} \triangleq \mathbf{D}\mathbf{y}^{\mathsf{e}}_t, \quad (2)$$

where $\mathbf{y}^{\mathsf{e}}_t$ is the *extended* feature vector comprised of the statics in a window [7]. To compute the corrupted speech vector with statics and dynamics from a window of speech and noise vectors, the mismatch function is applied to each time slice:

$$\mathbf{y}_t = \mathbf{D}\begin{bmatrix} \mathbf{f}(\mathbf{x}^{\mathsf{s}}_{t-1}, \mathbf{n}^{\mathsf{s}}_{t-1}, \mathbf{h}^{\mathsf{s}}, \boldsymbol{\alpha}_{t-1}) \\ \mathbf{f}(\mathbf{x}^{\mathsf{s}}_{t}, \mathbf{n}^{\mathsf{s}}_{t}, \mathbf{h}^{\mathsf{s}}, \boldsymbol{\alpha}_{t}) \\ \mathbf{f}(\mathbf{x}^{\mathsf{s}}_{t+1}, \mathbf{n}^{\mathsf{s}}_{t+1}, \mathbf{h}^{\mathsf{s}}, \boldsymbol{\alpha}_{t+1}) \end{bmatrix} \triangleq \mathbf{D}\mathbf{f}^{\mathsf{e}}(\mathbf{x}^{\mathsf{e}}_t, \mathbf{n}^{\mathsf{e}}_t, \mathbf{h}^{\mathsf{e}}, \boldsymbol{\alpha}^{\mathsf{e}}_t), \quad (3)$$

where extended vectors $\mathbf{x}^{\mathsf{e}}, \mathbf{n}^{\mathsf{e}}, \mathbf{h}^{\mathsf{e}}$, and $\boldsymbol{\alpha}^{\mathsf{e}}$ are defined similarly to $\mathbf{y}^{\mathsf{e}}$. Given distributions over $\mathbf{x}^{\mathsf{e}}, \mathbf{n}^{\mathsf{e}}, \boldsymbol{\alpha}^{\mathsf{e}}$ and a setting for $\mathbf{h}^{\mathsf{e}}$, it is now possible to write the distribution of $\mathbf{y}$ as an integral with a Dirac delta δ.:

$$p(\mathbf{y}) = \iiint \delta_{\mathbf{D}\mathbf{f}^{\mathsf{e}}(\mathbf{x}^{\mathsf{e}}_t, \mathbf{n}^{\mathsf{e}}_t, \mathbf{h}^{\mathsf{e}}, \boldsymbol{\alpha}^{\mathsf{e}}_t)}(\mathbf{y})\, p(\boldsymbol{\alpha}^{\mathsf{e}}) d\boldsymbol{\alpha}^{\mathsf{e}} p(\mathbf{n}^{\mathsf{e}}) d\mathbf{n}^{\mathsf{e}} p(\mathbf{x}^{\mathsf{e}}) d\mathbf{x}^{\mathsf{e}}. \quad (4)$$

However, because of the non-linearity of the mismatch function, this integral has no closed form. In particular, even if the distributions over $\mathbf{x}^{\mathsf{e}}, \mathbf{n}^{\mathsf{e}}, \boldsymbol{\alpha}^{\mathsf{e}}$ are Gaussian, $p(\mathbf{y})$ is not. It is a common approximation to linearise the mismatch function so $p(\mathbf{y})$ drops out as Gaussian [11], [5], [7], but this paper will avoid this. Given a value for $\mathbf{y}$ and distributions for $\mathbf{x}^{\mathsf{e}}, \mathbf{n}^{\mathsf{e}}, \mathbf{h}^{\mathsf{e}}$, and $\boldsymbol{\alpha}^{\mathsf{e}}$, it is possible to approximate the value of the density in the limit exactly with a Monte Carlo approach [3], but this technique is too slow to be used in a speech recogniser.

Though the distribution $p(\mathbf{y})$ has no closed form, it is straightforward to draw samples $\mathbf{y}^{(s)}$ from it. This first draws samples $\mathbf{x}^{\mathsf{e}(s)}, \mathbf{n}^{\mathsf{e}(s)}, \boldsymbol{\alpha}^{\mathsf{e}(s)}$ from their respective distributions. (3) can then be applied to yield a corrupted speech sample:

$$\mathbf{y}^{(s)} = \mathbf{D}\mathbf{f}^{\mathsf{e}}\big(\mathbf{x}^{\mathsf{e}(s)}, \mathbf{n}^{\mathsf{e}(s)}, \mathbf{h}^{\mathsf{e}}, \boldsymbol{\alpha}^{\mathsf{e}(s)}\big). \quad (5)$$

The advantage of representing the distribution $p(\mathbf{y})$ by a set of samples is that it does not assume any properties of the distribution. Section IV will estimate a form appropriate for decoding directly from this Monte Carlo approximation.

Drawing a corrupted-speech sample for a component k requires the component's extended clean speech distribution. This distribution can be found by modifying the last iteration of Baum–Welch so it does not apply the projection to statics and dynamics as it gathers statistics for training the speech recogniser's Gaussians [7]. The parameters of $p(\mathbf{n}_t)$ are often estimated on the data to be recognised using a linearisation of the mismatch function. However, see [12] for a strategy for estimating the noise model without applying this linearisation. To remove the influence of the algorithm for noise model estimation, here the noise distribution will be assumed known.

III. Predictive Methods

For practical speech recognition, it is often important to find approximations that are fast for decoding. Model compensation for noise robustness is a good example: it approximates the predicted corrupted speech with parametric distributions. This paper will present a framework for methods that minimise the Kullback-Leibler (KL) divergence between the predicted distribution p and its approximation q.

The KL divergence is suitable for two reasons. First, it is a well-understood measure of divergence between two distributions that is minimised when the distributions are equal. Second, minimising the KL divergence is equivalent to maximising the expected log-likelihood under the predicted distribution. Many known algorithms that maximise the log-likelihood on data can be turned into predictive variants.

The objective of predictive methods is to find the distribution q that minimises the KL divergence to predicted distribution p. An HMM speech recogniser consists of a distribution $p(\Theta)$ over hidden variables $\Theta = \{\theta_t\}$, where θ_t are the HMM states in Fig. 1, and a distribution $p_\Theta(\mathcal{Y})$ over observed variables $\mathcal{Y} = \{\mathbf{y}_t\}$ given state sequence Θ. If both p and q are HMMs, they therefore factorise as

$$p(\Theta, \mathcal{Y}) = p(\Theta)p_\Theta(\mathcal{Y}); \qquad q(\Theta, \mathcal{Y}) = q(\Theta)q_\Theta(\mathcal{Y}). \quad (6)$$

This paper will not change the state transition model, so that $q(\Theta) := p(\Theta)$. The optimal approximation q then is

$$\begin{aligned} q^* &:= \arg\min_q \mathcal{KL}(p\|q) \\ &= \arg\min_q \sum_\Theta \int p(\Theta)p_\Theta(\mathcal{Y}) \log\left(\frac{p(\Theta)p_\Theta(\mathcal{Y})}{q(\Theta)q_\Theta(\mathcal{Y})}\right) d\mathcal{Y} \\ &= \arg\min_q \sum_\Theta p(\Theta) \int p_\Theta(\mathcal{Y}) \log\left(\frac{p_\Theta(\mathcal{Y})}{q_\Theta(\mathcal{Y})}\right) d\mathcal{Y} \end{aligned}$$

$$= \arg\min_q \sum_\Theta p(\Theta)\mathcal{KL}(p_\Theta \| q_\Theta)\,. \tag{7}$$

In HMMs, $p_\Theta(\mathcal{Y})$ and $q_\Theta(\mathcal{Y})$ represent the output distributions. Since these factorise per time, the optimal setting of q can be expressed in terms of the per-state KL divergence:

$$q^* := \arg\min_q \sum_\theta p(\theta)\mathcal{KL}(p_\theta \| q_\theta) \tag{8a}$$

Here, $p(\theta)$ is the prior distribution for state θ. The maximum-likelihood estimate of $p(\theta)$ is proportional to the occupancy of state θ on the training data.

The KL divergence can be defined in terms of the cross-entropy $\mathcal{H}(p_\theta\|q_\theta) = -\int p_\theta(\mathbf{y}) \log q_\theta(\mathbf{y}) d\mathbf{y}$ and the entropy $\mathcal{H}(p_\theta) = \mathcal{H}(p_\theta\|p_\theta)$:

$$\mathcal{KL}(p_\theta \| q_\theta) \triangleq \mathcal{H}(p_\theta\|q_\theta) - \mathcal{H}(p_\theta). \tag{8b}$$

When minimising the KL divergence with respect to q, the entropy $\mathcal{H}(p_\theta)$ is constant, so that in (8a) only the cross-entropy needs to be minimised:

$$q^* := \arg\min_q \sum_\theta p(\theta)\mathcal{H}(p_\theta\|q_\theta)\,. \tag{8c}$$

In speech recognition, the state-conditional distributions p_θ and q_θ are normally mixtures of Gaussians. Their weights, the component priors, will be written π_k and ω_m:

$$p_\theta(\mathbf{y}) = \sum_{k\in\Omega_\theta} \pi_k p_k(\mathbf{y})\,; \qquad q_\theta(\mathbf{y}) = \sum_{m\in\Omega_\theta} \omega_m q_m(\mathbf{y})\,. \tag{9}$$

The key insight that motivates this paper is that unless it is possible to set q_m exactly equal to p_k, the KL divergence between these mixtures, in (8c), can usually not be analytically minimised exactly. Section IV will introduce a variational upper bound to perform the minimisation. Initially, though, a simpler and commonly-used bound will be considered: the *matched-pair bound* [13], which relates each component k in p_θ to a component in q_θ, in this case $m = k$:

$$\begin{aligned}
&\sum_\theta p(\theta)\mathcal{H}(p_\theta\|q_\theta) = -\sum_\theta p(\theta)\int p_\theta(\mathbf{y}) \log q_\theta(\mathbf{y}) d\mathbf{y} \\
&= -\sum_\theta p(\theta)\sum_{k\in\Omega_\theta}\pi_k \int p_k(\mathbf{y}) \log\left(\sum_{m\in\Omega_\theta}\omega_m q_m(\mathbf{y})\right) d\mathbf{y} \\
&\leq -\sum_\theta p(\theta)\sum_{k\in\Omega_\theta}\pi_k \int p_k(\mathbf{y}) \log(\omega_k q_k(\mathbf{y})) d\mathbf{y} \\
&= \sum_\theta \sum_{k\in\Omega_\theta} p(k)\big(\mathcal{H}(p_k\|q_k) - \log(\omega_k)\big),
\end{aligned} \tag{10a}$$

where the component priors $p(k) \triangleq p(\theta)\pi_k$. The minimisation in (8c) then becomes

$$q^* := \arg\min_q \sum_\theta \sum_{k\in\Omega_\theta} p(\theta)\pi_k\big(\mathcal{H}(p_k\|q_k) - \log(\omega_k)\big). \tag{10b}$$

The optimal setting for the mixture weights is $\omega_k^* := \pi_k$. How to minimise the weighted cross-entropies $\mathcal{H}(p_k\|q_k)$ depends on the parametrisation of q_k. The following will consider two parametrisations: model compensation, which estimates each Gaussian parameter separately; and linear transformations that are shared between clean speech Gaussians.

A. Estimating Model Parameters

A straightforward method for optimising the cross-entropy in (10b) is to estimate the parameters of q_k directly. This can be done for each component k separately. Speech recogniser components are usually Gaussians, with parameters $\mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$, which can be found with

$$\boldsymbol{\mu}_k := \mathcal{E}_{p_k}\{\mathbf{y}\}\,; \qquad \boldsymbol{\Sigma}_k := \mathcal{E}_{p_k}\{\mathbf{y}\mathbf{y}^\mathsf{T}\} - \boldsymbol{\mu}_k\boldsymbol{\mu}_k^\mathsf{T}. \tag{11}$$

This is what model compensation methods approximate. A popular approach is to linearise the influence of the noise in (3), so that the noise-corrupted speech distribution drops out as Gaussian. This is called vector Taylor series (VTS) compensation [11], [5], [9], [7].

To obtain a more accurate estimate, a Monte Carlo approach will be used, called extended DPMC [14], [15]. For each component, S samples $\mathbf{y}^{(s)}$ are drawn as discussed in section II. The expectations in (11) are then approximated with

$$\mathcal{E}_{p_k}\{\mathbf{y}\} \simeq \frac{1}{S}\sum_s \mathbf{y}^{(s)}; \;\; \mathcal{E}_{p_k}\{\mathbf{y}\mathbf{y}^\mathsf{T}\} \simeq \frac{1}{S}\sum_s \mathbf{y}^{(s)}\mathbf{y}^{(s)^\mathsf{T}}. \tag{12}$$

In the limit as the number of samples goes to infinity, extended DPMC yields the optimal Gaussian-for-Gaussian compensation. However, the corrupted speech distribution for one clean speech Gaussian is not Gaussian-distributed.

B. Estimating Linear Transformations

An approach that requires fewer parameters to be estimated is to constrain the approximate distribution q to a transformed clean speech model set. CMLLR [16] (constrained maximum-likelihood linear regression) is a well-known adaptation method that applies one affine transformation to all Gaussian components in a base class. CMLLR is normally estimated on adaptation data with expectation–maximisation. For its predictive variant, predictive CMLLR (PCMLLR) [17], [18], [19], the statistics from adaptation data are replaced by predicted statistics. Good results have been obtained by deriving these statistics from *joint uncertainty decoding* and from vocal tract length normalisation.

CMLLR applies the same linear transformation to each Gaussian in one base class. This is equivalent to applying the inverse transformation to the feature vector. By constraining Gaussians in one base class to share the same transformed feature vector, decoding with CMLLR becomes fast. For clean speech component k with $p_k = \mathcal{N}(\boldsymbol{\mu}_{\mathsf{x}k}, \boldsymbol{\Sigma}_{\mathsf{x}k})$, in base class c, transformation $\{\mathbf{A}^{(c)}, \mathbf{b}^{(c)}\}$ is applied as

$$q_k(\mathbf{y}) \triangleq \left|\mathbf{A}^{(c)}\right| \cdot \mathcal{N}\big(\mathbf{A}^{(c)}\mathbf{y} + \mathbf{b}^{(c)};\, \boldsymbol{\mu}_{\mathsf{x}k},\, \boldsymbol{\Sigma}_{\mathsf{x}k}\big). \tag{13}$$

Since $\mathbf{A}^{(c)}$ can be full, CMLLR allows modelling of feature correlations even when the Gaussian covariances are diagonal with minimal impact on decoding speed.

The original, adaptive, version of CMLLR can be written as minimising a KL divergence to the complete data distribution derived from an empirical distribution representing the adaptation data. There is no space here for the derivation, but [20] gives the details. The only change from adaptive and predictive

CMLLR is the form of the statistics; the algorithm to estimate the transformations from these statistics remains the same. As an example, one of the statistics consists of matrices $\mathbf{G}^{(ci)}$ for each base class c and each dimension i. Expressed in terms of distribution p,

$$\mathbf{G}^{(ci)} \triangleq \sum_{k \in \Omega^{(c)}} p(k) \frac{1}{\sigma^2_{k,i}} \mathcal{E}_{p_k} \left\{ \begin{bmatrix} \mathbf{y}\mathbf{y}^\mathsf{T} & \mathbf{y} \\ \mathbf{y}^\mathsf{T} & 1 \end{bmatrix} \right\}, \tag{14}$$

where $p(k)$ is the component prior, and $\mathcal{E}_{p_k}\{\cdot\}$ denotes the expectation under p_k. It is straightforward to recover the original expression in [16] from this by substituting the distribution over the complete data for p.

$\mathbf{G}^{(ci)}$ can be computed with any method that yields component-dependent first- and second-order statistics in (11). This paper will use the Monte Carlo estimates in (12) (with, for efficiency, the expected number of samples per component equal to its prior; see [12] for details). However, the first- and second-order statistics still make it impossible to model non-Gaussian effects in the distribution p_k. To do that, the next section will introduce an approach to estimating q per state.

IV. Predictive Variational Methods

Minimising the KL divergences between states, instead of their components, makes it possible to approximate the non-Gaussian shape of p. In speech recognition, q_θ is normally a mixture of components q_m, as in (9). This section will first introduce a variational approach to estimating this mixture model. To model the non-Gaussian shape of p_θ, each component in q_θ must be able to represent a specific part of the acoustic space associated with a component in p_θ. Therefore, p_θ is replaced with a Monte Carlo approximation, and responsibilities are assigned per sample. Then, two forms of parametrisation of q_θ will be discussed. The first, variational extended DPMC, estimates the components' weights, means, and covariances. The other, variational PCMLLR, only estimates the parameters of linear transformations shared between multiple clean speech Gaussians across states.

As in section III, p is replaced with a Monte Carlo version by drawing joint samples $\big(\theta^{(s)}, \mathbf{y}^{(s)}\big)$ from $p(\theta, \mathbf{y})$. Section IV-A will discuss the sampling process in more detail. The joint samples are used to at the same time approximate the sum and the integral over the term in square brackets:

$$\begin{aligned} \sum_\theta p(\theta) \mathcal{H}(p_\theta \| q_\theta) &= -\sum_\theta p(\theta) \int p_\theta(\mathbf{y}) \big[\log q_\theta(\mathbf{y}) \big] d\mathbf{y} \\ &\simeq -\sum_s \log q_{\theta^{(s)}}\big(\mathbf{y}^{(s)}\big) \end{aligned} \tag{15a}$$

Since the q_θ are mixture models, this expression cannot be minimised analytically. However, similar to expectation–maximisation, it is possible to minimise a variational upper bound as a proxy for optimising (15a) itself. This introduces an additional set of parameters whose optimisation is interleaved with the optimisation of q. These parameters $r_m^{(s)} \geq 0$ can be seen as assigning a fractional responsibility for sample s to component m. By ensuring that for each sample s the responsibilities are normalised, $\sum_m r_m^{(s)} = 1$, Jensen's inequality can be used (in (15b)) to find an upper bound $\mathcal{F}$ to the weighted cross-entropy in (8c). Substituting (9) into (15a), and inserting variational parameters $r_m^{(s)}$,

$$-\sum_s \log q_{\theta^{(s)}}\big(\mathbf{y}^{(s)}\big) = -\sum_s \log \left(\sum_{m \in \Omega_{\theta^{(s)}}} r_m^{(s)} \frac{\omega_m q_m\big(\mathbf{y}^{(s)}\big)}{r_m^{(s)}} \right)$$

$$\leq -\sum_s \sum_{m \in \Omega_{\theta^{(s)}}} r_m^{(s)} \log \left(\frac{\omega_m q_m\big(\mathbf{y}^{(s)}\big)}{r_m^{(s)}} \right) \tag{15b}$$

$$= -\sum_s \sum_{m \in \Omega_{\theta^{(s)}}} r_m^{(s)} \left(\log q_m\big(\mathbf{y}^{(s)}\big) + \log \frac{\omega_m}{r_m^{(s)}} \right)$$

$$\triangleq \mathcal{F}(p, q, \mathbf{r}). \tag{15c}$$

The minimisation of $\mathcal{F}$ interleaves optimising $\mathbf{r}$, the collection of $r_m^{(s)}$, and q. The responsibilities for each sample can be set to their optimum with $r_m^{(s)} \propto \pi_m q_m\big(\mathbf{y}^{(s)}\big)$. How to optimise the distribution q depends on its form. Sections IV-B and IV-C will discuss two possible forms.

A. Sampling from a Mixture Model

Consider the case where the predicted distribution p_θ for each state is a mixture model. To draw a sample from p, a sample $\theta^{(s)}$ from the state prior $p(\theta)$ is drawn, and then a corresponding component $k^{(s)}$ with probability π_k. Section II has discussed how to draw sample $\mathbf{y}^{(s)}$ from the distribution $p_{k^{(s)}}$.

The resulting joint sample $(\theta^{(s)}, \mathbf{y}^{(s)})$ can contribute to the estimation of any or all components of q_θ: there is no explicit link to the component of p_θ from which the sample was drawn. An important consequence of this is that the components of q_θ are free to move to represent samples drawn from other components in the same state p_θ. This allows the non-Gaussian shape of p_k to be modelled.

Section III has derived a matched-pair bound to the variational approximation discussed in this section. It assigned the responsibility for each sample to the component of q_θ matching the component of p_θ. The variational bound becomes equal to the matched-pair bound when $r_m^{(s)}$ in (15b) is set to 1 for $m = k^{(s)}$ and to 0 otherwise. This upper bound will be used as an initialisation for the variational minimisation.

B. Estimating Model Parameters

As in section III-A, it is possible to estimate the model parameters directly. This has been called "extended iterative DPMC" [14], [20], but here it will be called "variational eDPMC". It is possible to optimise each state-conditional distribution q_θ separately. The update to the parameters of q_θ that optimises (15c) is the same as in expectation–maximisation for mixtures of Gaussians. The component weights are found from the summed responsibilities as

$$\omega_m \propto \sum_s r_m^{(s)}. \tag{16a}$$

The mean for each component is set to

$$\boldsymbol{\mu}_m := \frac{1}{\sum_s r_m^{(s)}} \sum_s r_m^{(s)} \mathbf{y}^{(s)}, \tag{16b}$$

and similarly for the covariance. As in standard expectation–maximisation, this allows components to move to model the shape of the distribution, irrespective of which components the samples were originally drawn from.

C. Estimating Linear Transformations

It is also possible to apply predictive CMLLR in the variational Monte Carlo framework. The only parameters that are estimated are then those of a set of linear transformations that are applied to the Gaussians. Unlike in variational eDPMC, the original clean speech Gaussians are retained.

Note that, as in (13), different components for one state-conditional mixture can be transformed differently if they are in different base classes. However, since a variational approach is used, the optimisation can move Gaussians in the original model and re-use them to model parts of space that are generated by different Gaussians. This has the counter-intuitive effect that linear transformations of Gaussians allow modelling of the non-Gaussian shape of a distribution predicted from essentially the same Gaussians.

The predicted distribution p_k in (14) is approximated with Monte Carlo. There is no space here for the derivation, but see [20] for details. As in (15a), the joint samples approximate both the sum and the integral in the expectation:

$$\mathbf{G}^{(ci)} \simeq \sum_s \sum_{m \in \Omega^{(c)}} r_m^{(s)} \frac{1}{\sigma_{m,i}^2} \begin{bmatrix} \mathbf{y}^{(s)} \mathbf{y}^{(s)\mathsf{T}} & \mathbf{y}^{(s)} \\ \mathbf{y}^{(s)\mathsf{T}} & 1 \end{bmatrix}. \qquad (17)$$

Given these statistics for one base class c, the CMLLR algorithm iteratively finds transformation $\{\mathbf{A}^{(c)}, \mathbf{b}^{(c)}\}$ that yields a local minimum for the variational upper bound in (15c).

However, gathering the statistics cannot be performed per base class, since a sample drawn from state $\theta^{(s)}$ can contribute to any of the components of that state, which can be in any base class. This implies that it is impossible to guarantee that each $\mathbf{G}^{(ci)}$ is estimated on enough samples: the mass of samples drawn from a component in one base class can be assigned to components of another base class. However, this re-assignment is exactly the purpose of the variational scheme. Allowing components to be transformed to model part of the probability mass from other base classes removes the restriction that the noise-corrupted speech for one clean speech Gaussian is modelled by a Gaussian.

To partly mitigate the potential problem that some base classes are trained on too little data, an equal number of samples can be drawn from each base class, and they then be weighted. Additionally, to reduce the variance of the statistics, it is allowable to sample states and components using systematic sampling. See [12] for details and the proof.

V. Experiments

To illustrate the limitations of the matched-bound approximation, the variational predictive methods described in this paper are tested on the 1000-word-vocabulary Resource Management corpus [21]. This task contains 109 training speakers reading 3990 sentences, a total of 3.8 hours of data. All results are averaged over three of the four available test sets, Feb 89, Oct 89, and Feb 91 (Sep 92 is not used), a total of 30 test speakers and 900 utterances. Operations Room noise from the NOISEX-92 database [22] is artificially added at 20 and 14 dB. For a fair comparison, the influence of the noise estimation algorithm should be eliminated. Therefore, a full-covariance Gaussian noise model is extracted directly from the noise audio. (A method that could estimate a noise model for Monte Carlo PCMLLR is proposed in [12].) Because a noise model also implicitly compensates for speaker differences, word error rates will be slightly higher than in [7].

State-clustered triphone models with six components per mixture are built using the HTK RM recipe [23]. The number of components is about 9500. The extended speech statistics are striped as in [7]. The language model is a word-pair grammar.

Table I contains word error rates for model compensation. "VTS" is a standard scheme. It linearises the mismatch function and applies the *continuous-time approximation*, which makes off-diagonal elements of the covariance matrix unreliable [7], so the covariance matrices are (as is usual) diagonalised.

Extended DPMC (eDPMC), discussed in section III-A, estimates means and covariances for each component separately using Monte Carlo. As the number of samples goes to infinity (at 100 000 samples, performance has converged) it yields the optimal Gaussian-for-Gaussian compensation. Compared to VTS, eDPMC removes the linearisation of the mismatch function and the continuous-time approximation (see [7]), and thus yields improved performance. Removing the diagonalisation for eDPMC is especially useful at lower signal-to-noise ratios, when the noise affects feature correlations most. At 14 dB, it yields another 15 % relative increase in performance.

The above results still assume that one clean speech Gaussian generates Gaussian-distributed corrupted speech. The bottom row of Table I shows the effect of removing this constraint. Variational eDPMC uses the variational approach from section IV, which allows corrupted-speech samples to be re-assigned to different components. Because probability mass can be moved to different components, the state-conditional mixture distribution is able to model the non-linear effects of the interaction of the speech and noise. This yields a 10 % relative reduction in word error rate compared to the optimal Gaussian-for-Gaussian compensation.

Table II shows the same contrasts, but estimating only the parameters of linear transformations, by applying predictive CMLLR. As discussed in section III-B, non-variational PCMLLR can be trained from different forms of statistics. The word error rates in the first row use statistics from a VTS-compensated model. The combination of the approximations in VTS and PCMLLR leads to reduced performance. The numbers in the second row are from CMLLR essentially trained from full-covariance eDPMC statistics, with 10 000 samples per base class. Though the clean speech samples can be drawn off-line, collecting the statistics in (17) is still costly. However, for the 16 base class system, the word error rate is the same with 5000 samples per base class, which makes the total number of samples 80 000. The complexity of this is invariant to the number of components, so for larger systems this operating

TABLE I
WORD ERROR RATES FOR MODEL COMPENSATION.

Method	Optimisation	Shape	20 dB	14 dB
VTS	per component	diag	8.6	17.4
eDPMC	per component	diag	7.5	14.9
		full	7.4	13.3
	variational	full	6.9	12.0

TABLE II
WORD ERROR RATES FOR PREDICTIVE CMLLR.

	Base classes			
	16	1024	16	1024
Statistics	20 dB		14 dB	
VTS	10.9	10.0	23.8	20.9
eDPMC	9.2	8.1	19.3	16.4
Variational	8.7	7.9	19.6	15.1

point could yield a reasonable trade-off. With 1024 base classes, PCMLLR with eDPMC does perform better than VTS. Though the full transformation matrix of PCMLLR implicitly performs some compensation for correlation changes, it cannot model the non-Gaussian shape of the distributions.

The bottom row in Table II shows results for variational PCMLLR, which applies a variational approach to estimating linear transformations per base class. It allows the transformations to move components in space to model the non-Gaussian distribution of the corrupted speech. At 20 dB, around 26 % of the probability mass of the samples goes to different components; at 14 dB, 37 %. At 16 base classes this does not consistently yield benefits, but with 128 or more, and especially at lower signal-to-noise ratios, word error rates improve compared to non-variational PCMLLR. This ability to model the non-Gaussian aspect of the corrupted-speech distribution with just linear transformations of clean speech Gaussians shows the power of the variational approach.

VI. CONCLUSION

This paper has viewed model compensation for noise-robustness and predictive linear transformations from a variational perspective. These methods can be seen as minimising an upper bound on the divergence between the corrupted speech and the model for decoding. It is possible to find a tighter bound by considering the divergence between states rather than between components. When applied to eDPMC and predictive CMLLR, this yields reductions in the word error rate. It is possible to model the non-linear impact of the noise with just linear transformations of Gaussians. The variational predictive framework should allow a wide range of schemes to be developed. These could, for example, address the computational cost of the schemes proposed here.

ACKNOWLEDGMENT

The authors like to thank David Barber for suggesting the Monte Carlo approximation to predictive methods. This work was partly supported by EPSRC Project EP/I006583/1 (Generative Kernels and Score Spaces for Classification of Speech) within the Global Uncertainties Programme.

REFERENCES

[1] J. Li, D. Yu, Y. Gong, and L. Deng, "Unscented transform with online distortion estimation for HMM adaptation," in *Proceedings of Interspeech*, 2010, pp. 1660–1663.

[2] T. T. Kristjansson, "Speech recognition in adverse environments: a probabilistic approach," Ph.D. dissertation, University of Waterloo, 2002.

[3] R. C. van Dalen and M. J. F. Gales, "Asymptotically exact noise-corrupted speech likelihoods," in *Proceedings of Interspeech*, Sep 2010, pp. 709–712.

[4] R. C. van Dalen, F. Flego, and M. J. F. Gales, "Transforming features to compensate speech recogniser models for noise," in *Proceedings of Interspeech*, Sep 2009, pp. 2499–2502.

[5] A. Acero, L. Deng, T. Kristjansson, and J. Zhang, "HMM adaptation using vector Taylor series for noisy speech recognition," in *Proceedings of* ICSLP, vol. 3, 2000, pp. 229–232.

[6] M. Seltzer, A. Acero, and K. Kalgaonkar, "Acoustic model adaptation via linear spline interpolation for robust speech recognition," in *Proceedings of* ICASSP, 2010, pp. 4550–4553.

[7] R. C. van Dalen and M. J. F. Gales, "Extended VTS for noise-robust speech recognition," IEEE *Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 733–743, Apr 2011.

[8] A. Acero, "Acoustical and environmental robustness in automatic speech recognition," Ph.D. dissertation, Carnegie Mellon University, 1990.

[9] L. Deng, J. Droppo, and A. Acero, "Enhancement of log mel power spectra of speech using a phase-sensitive model of the acoustic environment and sequential estimation of the corrupting noise," IEEE *Transactions on Speech and Audio Processing*, vol. 12, no. 2, pp. 133–143, 2004.

[10] V. Leutnant and R. Haeb-Umbach, "An analytic derivation of a phase-sensitive observation model for noise robust speech recognition," in *Proceedings of Interspeech*, 2009, pp. 2395–2398.

[11] P. J. Moreno, "Speech recognition in noisy environments," Ph.D. dissertation, Carnegie Mellon University, 1996.

[12] R. C. van Dalen and M. J. F. Gales, "A variational perspective on noise-robust speech recognition," Cambridge University Engineering Department, Tech. Rep. CUED/F-INFENG/TR.670, 2011.

[13] J. R. Hershey and P. A. Olsen, "Approximating the Kullback Leibler divergence between Gaussian mixture models," in *Proceedings of* ICASSP, 2007.

[14] M. J. F. Gales, "Model-based techniques for noise robust speech recognition," Ph.D. dissertation, Cambridge University, 1995.

[15] R. C. van Dalen and M. J. F. Gales, "Covariance modelling for noise robust speech recognition," in *Proceedings of Interspeech*, Sep 2008, pp. 2000–2003.

[16] M. J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech and Language*, vol. 12, no. 2, pp. 75–98, 1998.

[17] M. J. F. Gales and R. C. van Dalen, "Predictive linear transforms for noise robust speech recognition," in *Proceedings of* ASRU, Dec 2007, pp. 59–64.

[18] H. Xu, M. J. F. Gales, and K. K. Chin, "Joint uncertainty decoding with predictive methods for noise robust speech recognition," IEEE *Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 6, pp. 1665–1676, 2011.

[19] C. Breslin, K. K. Chin, M. J. F. Gales, K. Knill, and H. Xu, "Prior information for rapid speaker adaptation," in *Proceedings of Interspeech*, 2010, pp. 1644–1647.

[20] R. C. van Dalen, "Statistical models for noise-robust speech recognition," Ph.D. dissertation, University of Cambridge, Jan 2011.

[21] P. Price, W. M. Fisher, J. Bernstein, and D. S. Pallett, "The DARPA 1000-word Resource Management database for continuous speech recognition," in *Proceedings of* ICASSP, vol. 1, 1988, pp. 651–654.

[22] A. Varga and H. J. M. Steeneken, "Assessment for automatic speech recognition II: NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Communication*, vol. 12, no. 3, pp. 247–251, 1993.

[23] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. A. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, "The HTK book (for HTK version 3.4)," 2006. [Online]. Available: http://htk.eng.cam.ac.uk/docs/docs.shtml