論文 / 著書情報
Article / Book Information

Title	Robust Discriminative Training Against Data Insufficiency in PLDA-Based Speaker Verification
Authors	Johan Rohdin, Sangeeta Biswas, Koichi Shinoda
Citation	Elsevier Computer Speech and Language, vol. 35, pp. 32-57
Pub. date	2015, 6
DOI	http://dx.doi.org/10.1016/j.csl.2015.06.003
Creative Commons	See next page.
Note	このファイルは著者（最終）版です。 This file is author (final) version.

License

Robust Discriminative Training Against Data Insufficiency in PLDA-based Speaker Verification

Johan Rohdin, Sangeeta Biswas, and Koichi Shinoda

Tokyo Institute of Technology, Japan

Abstract

Probabilistic linear discriminant analysis (PLDA) with i-vectors as features has become one of the state-of-the-art methods in speaker verification. Discriminative training (DT) has proven to be effective for improving PLDA's performance but suffers more from data insufficiency than generative training (GT). In this paper, we achieve robustness against data insufficiency in DT in two ways. First, we compensate for statistical dependencies in the training data by adjusting the weights of the training trials in order for the training loss to be an accurate estimate of the expected loss. Second, we propose three constrained DT schemes, among which the best was a discriminatively trained transformation of the PLDA score function having four parameters. Experiments on the male telephone part of the NIST SRE 2010 confirmed the effectiveness of our proposed techniques. For various number of training speakers, the combination of weight-adjustment and the constrained DT scheme gave between 7% and 19% relative improvements in $\hat{C}_{\text{llr}}$ over GT followed by score calibration. Compared to another baseline, DT of all the parameters of the PLDA score function, the improvements were larger.

1. Introduction

In recent years, the combination of i-vector (Dehak et al., 2009, 2011) and probabilistic linear discriminant analysis (PLDA) (Ioffe, 2006; Kenny, 2010) has become one of the state-of-the-art systems in speaker verification. In this system, utterances are mapped into low dimensional vectors known as i-vectors. An i-vector contains information related to the speaker identity as well as irrelevant factors such as speaker's emotions, transmission channels, languages, and environmental noise. Given two i-vectors, the PLDA model separates speaker factors from irrelevant factors and provides a log-likelihood ratio (LLR) score for the two i-vectors being from the same speaker or not.

The PLDA parameters are usually optimized by generative training (GT) under the maximum likelihood (ML) criterion. However, several studies have suggested that discriminative training (DT) is beneficial, either as a complement or as an alternative to GT (Brümmer, 2010; Burget et al., 2011; Cumani et al., 2011, 2012, 2013; Borgström and McCree, 2013). In particular, score calibration by means of a discriminatively trained affine transformation (AT-Cal) (Brümmer, 2010), has become popular. AT-Cal only adjusts the scores and can therefore be applied to any speaker verification system. DT schemes that are specific to PLDA have also been proposed. A DT scheme that optimizes all the parameters of the PLDA LLR score function (Scr-UC)[1] was proposed by Burget et al. (2011) and Cumani et al. (2011) and a DT scheme that optimizes the PLDA model parameters instead of its score function, was proposed by Borgström and McCree (2013). However, DT is in general less robust against data insufficiency than GT. For example, in Cumani and Laface (2014), Scr-UC was worse than GT when the number of training speakers was less than around 1600. In this paper, we tackle the data insufficiency problem in two approaches. One is to effectively use the limited amount of training data. The other is to constrain the model parameters to avoid overfitting.

When the amount of training data is limited, each training utterance or speaker is often used in more than one training trial in the model training. Accordingly, the training trials are not statistically independent. As a consequence, the *average loss* of the training trials that we use as training objective is not the best estimate of the *expected loss*. We propose to adjust the weights of the training trials in order to obtain the *best linear unbiased estimator* (BLUE) of the *expected loss*.

In order to find the constraints that best avoid overfitting without constraining the model too much, we propose three discriminative training schemes that are less constrained than Src-UC (Burget et al., 2011; Cumani et al., 2011) but more flexible than AT-Cal (Brümmer, 2010). The first is a transformation of the PLDA LLR score function having four parameters. The second is a scaling of each element in the i-vectors. The third is a training scheme that, like Src-UC, updates all parameters of the PLDA LLR score function but preserves some properties of PLDA that are removed by Scr-UC (Rohdin et al., 2014a). Experiments on the male telephone part of the NIST SRE 2010 confirmed the effectiveness of our proposed techniques.

The remainder of this paper is organized as follows. Section 2 introduces the necessary background including the detection cost function, i-vector and PLDA based speaker-verification and discriminative PLDA training. Section 3 performs an analysis of the discriminative training methods. Based on the conclusions in Section 3, Section 4 presents the compensation for the statistical dependence, and Section 5 presents constrained discriminative PLDA training. Section 6 experimentally evaluates the methods. Finally, Section 7 concludes this paper.

[1] UC refers to *unconstrained*.

2. Background

2.1. Detection cost function

When making a decision based on the score from a speaker verification system, it is typically desired to minimize the expected cost of the decision. This is reflected in the *detection cost function* (DCF) used in the NIST evaluations. When the test and enrollment utterances in a trial are from the same speaker, we refer to the trial as a *target trial*, otherwise we refer to it as a *non-target trial*. The DCF measures the cost for an application with a prior probability of a target trial, P_{tar}, and the costs C_{FR} and C_{FA} for false rejection (FR) and false acceptance (FA) respectively.

$$\text{DCF} = P_{\text{tar}} C_{\text{FR}} P_{\text{FR}} + (1 - P_{\text{tar}}) C_{\text{FA}} P_{\text{FA}}, \tag{1}$$

where $P_{\text{FR}} = P(\text{error}|\text{target})$ and $P_{\text{FA}} = P(\text{error}|\text{non-target})$ are the empirical probabilities for FR and FA respectively estimated in the evaluation database. For the purpose of ranking systems, a scaling of the DCF does not make any difference. Therefore, for system optimization it is equivalent to use

$$\text{DCF}' = P_{\text{eff}} P_{\text{FR}} + (1 - P_{\text{eff}}) P_{\text{FA}}, \tag{2}$$

where

$$P_{\text{eff}} = \frac{P_{\text{tar}} C_{\text{FR}}}{P_{\text{tar}} C_{\text{FR}} + (1 - P_{\text{tar}}) C_{\text{FA}}}, \tag{3}$$

is known as the *effective prior*. In order to minimize the DCF, the decision threshold for the LLR score should be set to

$$\begin{aligned} \tau &= -\left(\log \frac{P_{\text{tar}}}{1 - P_{\text{tar}}} + \log \frac{C_{\text{FR}}}{C_{\text{FA}}} \right) \\ &= -\log \frac{P_{\text{eff}}}{1 - P_{\text{eff}}}. \end{aligned} \tag{4}$$

Therefore, if the speaker verification system outputs scores that can be interpreted as LLRs, the threshold can easily be obtained for any P_{eff}. The cost obtained by using τ as the decision threshold is called the *actual detection cost* (actDCF) and the cost obtained by the optimal threshold for the evaluation set, is called the *minimum detection cost* (minDCF). If actDCF and minDCF are similar, we say that the LLR scores are well *calibrated* for the particular P_{eff}.

2.2. i-vector based system

In the i-vector system (Dehak et al., 2011), it is assumed that the features from an utterance are generated by a *Gaussian Mixture Model* (GMM). It is further assumed that the GMM-*supervector*, $\boldsymbol{\mu}$, corresponding to an utterance can be modeled as

$$\boldsymbol{\mu} = \bar{\boldsymbol{\mu}} + \boldsymbol{T}\boldsymbol{\phi}, \tag{5}$$

where $\boldsymbol{\phi}$ is a random vector, $\boldsymbol{T}$ is a basis matrix for the *total variability space*, and $\bar{\boldsymbol{\mu}}$ is the mean of $\boldsymbol{\mu}$. It is assumed that $\boldsymbol{\phi}$ follows a standard normal distribution and its dimension, d, i.e., the rank of $\boldsymbol{T}$, is lower than the dimension of $\boldsymbol{\mu}$. Given the speech features of an utterance, the *i-vector*, $\boldsymbol{\omega}$, is the *maximum a posteriori* estimate of $\boldsymbol{\phi}$.

An i-vector contains information related to the speaker identity as well as irrelevant *channel* factors such as the speaker's emotions, transmission channels, language, and environmental noise. Channel factors should be removed in order to improve the accuracy of verification. Currently, PLDA has become one of the state-of-the-art channel compensation techniques in i-vector based speaker verification (Kenny, 2010).

2.3. PLDA

PLDA was originally proposed in image processing for object/face recognition (Ioffe, 2006; Prince and Elder, 2007). Kenny (2010) proposed to use it in speaker verification with i-vectors as features. In its most general form, PLDA assumes that the feature vectors (i-vectors), $\boldsymbol{\omega}$, are generated as:

$$\boldsymbol{\omega} = \boldsymbol{m} + \boldsymbol{V}\boldsymbol{y} + \boldsymbol{U}\boldsymbol{x} + \boldsymbol{D}\boldsymbol{z}, \tag{6}$$

where $\boldsymbol{m}$ is the mean of $\boldsymbol{\omega}$, $\boldsymbol{y}$ is a random vector depending on the class, and, $\boldsymbol{x}$ and $\boldsymbol{z}$ are random vectors depending on the *channel*, i.e., they are different from session to session. Contrary to the GMM-supervector, the i-vector is observed, which means that $\boldsymbol{U}$ and $\boldsymbol{D}$ must together span the full i-vector space. Two different PLDA configurations are popular. The configuration suggested by Prince and Elder (2007) constrains both $\boldsymbol{V}$ and $\boldsymbol{U}$ to have a rank lower than d, and $\boldsymbol{D}$ to be diagonal. This configuration is suitable for large d. This PLDA model is very similar to joint factor analysis (JFA) (Kenny, 2005; Kenny et al., 2007). The configuration suggested by Ioffe (2006) skips $\boldsymbol{U}$ but puts no constraints on $\boldsymbol{D}$, i.e.,

$$\boldsymbol{\omega} = \boldsymbol{m} + \boldsymbol{V}\boldsymbol{y} + \boldsymbol{D}\boldsymbol{z}. \tag{7}$$

This is the most popular configuration in speaker verification (Kenny, 2010; Brümmer and de Villiers, 2010) and we will use it in this study. The speaker matrix, $\boldsymbol{V}$, may have a rank lower than d (Kenny, 2010), or equal to d (Brümmer and de Villiers, 2010) in which case the model is known as the *two-covariance model*.

The original PLDA model (Ioffe, 2006; Prince and Elder, 2007) assumes $\boldsymbol{y}$, $\boldsymbol{x}$ and $\boldsymbol{z}$ follow Gaussian distribution (G-PLDA). However, the elements of the i-vector are, in reality, more heavy-tailed than the Gaussian distribution. Therefore, an extension named heavy-tailed PLDA (HT-PLDA), based on t-distributions, has been proposed (Kenny, 2010). HT-PLDA has much better performance than G-PLDA but is much slower both in the training and the testing phase. Later, normalizing the i-vectors to unit length, was shown to greatly improve the Gaussianity of the i-vectors so that G-PLDA provides similar performance as HT-PLDA (Garcia-Romero and Espy-Wilson, 2011). From here on, we only consider G-PLDA and refer to it as PLDA.

Given two i-vectors, $\boldsymbol{\omega_i}$ and $\boldsymbol{\omega_j}$, the LLR score is given by

$$s_{ij} = \log \frac{p\left(\boldsymbol{\omega}_i, \boldsymbol{\omega}_j | \mathcal{H}_s\right)}{p\left(\boldsymbol{\omega}_i, \boldsymbol{\omega}_j | \mathcal{H}_d\right)}, \tag{8}$$

where the hypotheses $\mathcal{H}_s$ and $\mathcal{H}_d$ are the following:

$\mathcal{H}_s$: $\boldsymbol{\omega_i}$ and $\boldsymbol{\omega_j}$ are from the same speaker.

$\mathcal{H}_d$: $\boldsymbol{\omega_i}$ and $\boldsymbol{\omega_j}$ are from different speakers.

According to Eq. (7), two i-vectors are generated by,

$$\begin{bmatrix} \boldsymbol{\omega}_i \\ \boldsymbol{\omega}_j \end{bmatrix} = \begin{bmatrix} \boldsymbol{m} \\ \boldsymbol{m} \end{bmatrix} + \begin{bmatrix} \boldsymbol{V} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{V} \end{bmatrix} \begin{bmatrix} \boldsymbol{y}_i \\ \boldsymbol{y}_j \end{bmatrix} + \begin{bmatrix} \boldsymbol{D} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{D} \end{bmatrix} \begin{bmatrix} \boldsymbol{z}_i \\ \boldsymbol{z}_j \end{bmatrix}, \tag{9}$$

where the speaker factors, $\boldsymbol{y}_i$ and $\boldsymbol{y}_j$ are the same in a target trial but different in a non-target trial. Accordingly, $[\boldsymbol{\omega}_i^T \ \boldsymbol{\omega}_j^T]^T$ follows a multivariate normal distribution. Calculating the mean and covariance of an i-vector pair in a target and a non-target trial based on Eq. (9) and plugging the resulting multivariate normal distributions into Eq. (8) results in a closed-form expression of the LLR given by

$$\begin{aligned} s_{ij} = &\boldsymbol{\omega}_i^T \boldsymbol{P} \boldsymbol{\omega}_j + \boldsymbol{\omega}_j^T \boldsymbol{P} \boldsymbol{\omega}_i + \boldsymbol{\omega}_i^T \boldsymbol{Q} \boldsymbol{\omega}_i + \boldsymbol{\omega}_j^T \boldsymbol{Q} \boldsymbol{\omega}_j \\ &+ (\boldsymbol{\omega}_i + \boldsymbol{\omega}_j)^T \boldsymbol{c} + k, \end{aligned} \tag{10}$$

where

$$\boldsymbol{P} = \frac{1}{2} \Sigma_{\text{tot}}^{-1} \Sigma_{\text{ac}} (\Sigma_{\text{tot}} - \Sigma_{\text{ac}} \Sigma_{\text{tot}}^{-1} \Sigma_{\text{ac}})^{-1}, \tag{11}$$

$$\boldsymbol{Q} = \frac{1}{2} \Sigma_{\text{tot}}^{-1} - (\Sigma_{\text{tot}} - \Sigma_{\text{ac}} \Sigma_{\text{tot}}^{-1} \Sigma_{\text{ac}})^{-1}, \tag{12}$$

$$\boldsymbol{c} = -2(\boldsymbol{P} + \boldsymbol{Q})\boldsymbol{m}, \tag{13}$$

$$\begin{aligned} k = &\frac{1}{2} (\log |\Sigma_{\text{tot}}| - \log |\Sigma_{\text{tot}} - \Sigma_{\text{ac}} \Sigma_{\text{tot}}^{-1} \Sigma_{\text{ac}}|) \\ &+ \boldsymbol{m}^T 2(\boldsymbol{P} + \boldsymbol{Q})\boldsymbol{m}, \end{aligned} \tag{14}$$

and $\Sigma_{\text{ac}} = \boldsymbol{V}\boldsymbol{V}^T$ and $\Sigma_{\text{tot}} = \boldsymbol{V}\boldsymbol{V}^T + \boldsymbol{D}\boldsymbol{D}^T$. Let $\boldsymbol{\gamma} = [\text{vec}(\boldsymbol{P})^T, \text{vec}(\boldsymbol{Q})^T, \boldsymbol{c}^T, k]^T$, where vec$(\cdot)$ stacks the columns of a matrix into a column vector, and let

$$\varphi(\boldsymbol{\omega}_i, \boldsymbol{\omega}_j) = \begin{bmatrix} \text{vec}(\boldsymbol{\omega}_i \boldsymbol{\omega}_j^T + \boldsymbol{\omega}_j \boldsymbol{\omega}_i^T) \\ \text{vec}(\boldsymbol{\omega}_i \boldsymbol{\omega}_i^T + \boldsymbol{\omega}_j \boldsymbol{\omega}_j^T) \\ \boldsymbol{\omega}_i + \boldsymbol{\omega}_j \\ 1 \end{bmatrix}. \tag{15}$$

Then Eq. (10) can be rewritten as (Burget et al., 2011; Cumani et al., 2011)

$$s_{ij} = \boldsymbol{\gamma}^T \varphi(\boldsymbol{\omega}_i, \boldsymbol{\omega}_j). \tag{16}$$

In other words, the PLDA LLR score is a linear function of a non-linear feature expansion $\varphi(\boldsymbol{\omega}_i, \boldsymbol{\omega}_j)$ of the two i-vectors.

Typically, the PLDA parameters are estimated by the generative maximum likelihood (ML) criterion.

2.4. Discriminative training

As mentioned in the previous subsection, normalizing the i-vectors to unit length improves their Gaussianity and substantially improves the performance of PLDA. However, even with length normalization, it is clear that there is still a mismatch between the model assumptions and the training data. Obviously a PLDA model cannot generate i-vectors of a fixed length. Further, it has been shown that for length-normalized i-vectors, the within-class covariance depends strongly on the speaker factors (Bousquet et al., 2014), whereas PLDA assumes the within-class covariance is independent of the speaker factors. If the model assumptions are not accurate, we cannot expect the parameters obtained by GT to be optimal neither for discriminating between speakers nor for providing well-calibrated LLR scores. Therefore, it might be better to use a DT criterion that directly optimizes the model for providing accurate LLR scores.

Discriminative training has been proven very effective for score calibration and fusion (Brümmer et al., 2007; Brümmer, 2010) based on affine functions. Given the scores $s_1, \ldots, s_n$ from n different systems, the fused and/or calibrated score is given by

$$s = w_0 + w_1 s_1 + \ldots + w_n s_n, \tag{17}$$

where the parameters $\boldsymbol{w} = [w_0, \ldots, w_n]$ need to be estimated. Let $t_h \in [-1, 1]$ be the label of trial h, i.e., it equals 1 if the two utterances are from the same speaker and -1 otherwise. Then $\boldsymbol{w}$ can be estimated by minimizing the loss, $\bar{l}(\boldsymbol{w})$,

$$\begin{aligned} \bar{l}(\boldsymbol{w}) = &\sum_{h:t_h=1} \frac{P_{\text{eff}}}{N_1} l\left(t_h, s_h(\boldsymbol{w}), \tau\right) \\ &+ \sum_{h:t_h=-1} \frac{1 - P_{\text{eff}}}{N_{-1}} l\left(t_h, s_h(\boldsymbol{w}), \tau\right), \end{aligned} \tag{18}$$

where N_1 and N_{-1} are the numbers of target and non-target trials respectively, and $l\left(t_h, s_h(\boldsymbol{w}), \tau\right)$ is a *loss function* for a trial. In this study, we follow Brümmer et al. (2007) and use the logistic regression loss function given by

$$l\left(t_h, s_h(\boldsymbol{w}), \tau\right) = \log\left(1 + \exp(-t_h(s_h(\boldsymbol{w}) - \tau))\right). \tag{19}$$

This loss function encourages both good calibration and discrimination (Brümmer and du Preez, 2006). See Brümmer and Doddington (2013) for a comparison of different such loss functions. With $n = 1$, we obtain an affine transformation of the scores which has become the standard approach for calibration (Brümmer et al., 2007; Brümmer, 2010). We refer to this method as AT-Cal. An affine transformation results in a very constrained update of the score function that cannot increase the system's ability to discriminate between target and non-target trials, i.e., to reduce minDCF. On the other hand it can substantially improve calibration, even with quite small amounts of data.

Burget et al. (2011) and Cumani et al. (2011) proposed to optimize the parameters $\boldsymbol{\gamma}$ in Eq. (16), by minimizing the loss in Eq. (18) where s_h is a function of $\boldsymbol{\gamma}$ instead of $\boldsymbol{w}$. Both the logistic regression loss in Eq. (19) and the SVM hinge loss were evaluated. In these studies, all possible i-vector pairs (typically some hundred millions) were used for training, and efficient calculations of the total loss and its gradient with respect to $\boldsymbol{\gamma}$ were presented. However, this method easily overfits to the training data due to the large number of parameters to be estimated. We refer to this method as Scr-UC, where UC refers to *unconstrained*. Scr-UC is similar to a DT scheme for JFA, proposed by Burget et al. (2008).

Borgström and McCree (2013) considered discriminative PLDA training with multiple enrollment sessions. The proposed training scheme applies DT to the model parameters, $\boldsymbol{m}$, $\boldsymbol{V}$ and $\boldsymbol{D}$, rather than the parameters of the LLR score function.

The training trials were from either single or multiple enrollment sessions. Only the eigenvalues (which were floored to be positive) or a scaling factor of the covariance matrices were updated by DT. The number of parameters to be estimated are therefore much fewer than in Scr-UC, which reduces the risk of over-training.

While GT uses utterances as observations and the speaker IDs as classes, DT aims directly at improving the LLR score and uses trials as observations and *same* or *different* speaker as classes. The trials need to be constructed from the available training data, ideally over all possible trials. However, when a training utterance (or just the same speaker) is used in more than one trial, the trials will be statistically dependent. Further, when the parameters $\boldsymbol{\gamma}$ are optimized directly, rather than the original PLDA model parameters, $\boldsymbol{m}$, $\boldsymbol{V}$ and $\boldsymbol{D}$, we may obtain a model with different properties than a PLDA model. Optimizing the parameters $\boldsymbol{m}$, $\boldsymbol{V}$ and $\boldsymbol{D}$ preserves the properties of the PLDA model but is more difficult due to the complex relation between these parameters and $\boldsymbol{P}$, $\boldsymbol{Q}$, $\boldsymbol{c}$ and k, given by Eqs. (11) to (14). These issues will be analyzed in the following section.

3. Analysis of discriminative PLDA training

In this section we carry out two analyses:

1. What is the effect of using statistically dependent trials in DT?
2. What is the effect of training the parameters of the PLDA LLR score directly instead of training the original parameters of the PLDA model?

Analysis 1 leads us to propose a compensation for the statistical dependence in Section 4. Analysis 2 motivates us to propose one of the constrained DT schemes in Section 5. In the remainder of this paper, $\boldsymbol{\theta}$ denotes the parameters to be estimated by DT (e.g., $\boldsymbol{w}$ in AT-Cal or $\boldsymbol{\gamma}$ in Scr-UC).

3.1. The effect of using statistically dependent trials

In this subsection we discuss how DT is affected by the use of statistically dependent trials. We argue that using an equal weight for all target trials and another equal weight for all non-target trials in the training objective is not optimal when the trials are statistically dependent. For example, consider the correlations due the same speakers being used in many training trials. If each trial has equal weight, speakers with many trials will influence the model more than speakers with few trials. In order to avoid this *speaker dependency* in the model and make it good for the general population, the weights for speakers with many trials need to be reduced. The remaining discussion in this subsection does not consider the reason for the statistical dependencies. In section 4 we show how to apply the principles discussed in this subsection specifically to the statistical dependencies that arise when all possible training trials are used, i.e., the same speakers and utterances are used in more than one training trial.

A trial consists of a label $T \in [1, -1]$ and two i-vectors $\boldsymbol{\Omega}_i$ and $\boldsymbol{\Omega}_j$. Here, we use upper case letters to denote that we treat these variables as random variables. We collect the i-vector pair of a trial in a vector denoted $\boldsymbol{\Omega}^{(p)} = [\boldsymbol{\Omega}_i^T, \boldsymbol{\Omega}_j^T]^T$. The loss of a trial, $L(\boldsymbol{\theta}) = l(T, s(\boldsymbol{\theta}, \boldsymbol{\Omega}^{(p)}))$, is then also a random variable. The training trials are observations of these random variables. Analogously, we use $\bar{l}(\boldsymbol{\theta})$ to denote the average loss of an observed set of training trials as in Eq. (18), and $\bar{L}(\boldsymbol{\theta})$ to be the corresponding random variable, i.e., the average loss of a set of trials treated as random variables. The *expected loss* of a single trial is given by

$$\begin{aligned} &\mathrm{E}_{T,\boldsymbol{\Omega}^{(p)}} L(\boldsymbol{\theta}) \\ &\quad = \mathrm{E}_{T,\boldsymbol{\Omega}^{(p)}}(l(\boldsymbol{\theta}, T, \boldsymbol{\Omega}^{(p)})) \\ &\quad = \sum_{T=-1,1} \mathrm{P}(T) \int_{\boldsymbol{R}^{2d}} l(T, s(\boldsymbol{\theta}, \boldsymbol{\Omega}^{(p)})) \mathrm{P}(\boldsymbol{\Omega}^{(p)}|T) \mathrm{d}^{2d}\boldsymbol{\Omega}^{(p)} \end{aligned} \tag{20}$$

where $\mathrm{P}(T = 1) = P_{\text{eff}}$, $\mathrm{P}(T = -1) = 1 - P_{\text{eff}}$ and $P(\boldsymbol{\Omega}^{(p)}|T)$ is the probability density function for the i-vector pair conditioned on the trial label. Discriminative training aims to find the $\boldsymbol{\theta}$ that minimizes $\mathrm{E}_{T,\boldsymbol{\Omega}^{(p)}}(L(\boldsymbol{\theta}))$ by minimizing $\bar{l}(\boldsymbol{\theta})$. In order for this approach to be successful, $\bar{L}(\boldsymbol{\theta})$ must be a good estimator of $E_{T,\boldsymbol{\Omega}^{(p)}}(L(\boldsymbol{\theta}))$ for each value of $\boldsymbol{\theta}$.

Let us generalize the DT objective as

$$\hat{L}(\boldsymbol{\theta}) = \tilde{P}_{\text{eff}} \hat{L}_1(\boldsymbol{\theta}) + (1 - \tilde{P}_{\text{eff}}) \hat{L}_{-1}(\boldsymbol{\theta}), \tag{21}$$

where $0 \leq \tilde{P}_{\text{eff}} \leq 1$,

$$\hat{L}_1(\boldsymbol{\theta}) = \sum_{h:t_h=1} \beta_h l\left(t_h, \boldsymbol{\theta}, \boldsymbol{\Omega}_h^{(p)}\right), \tag{22}$$

$$\hat{L}_{-1}(\boldsymbol{\theta}) = \sum_{h:t_h=-1} \beta_h l\left(t_h, \boldsymbol{\theta}, \boldsymbol{\Omega}_h^{(p)}\right), \tag{23}$$

and

$$\sum_{h:t_h=1} \beta_h = \sum_{h:t_h=-1} \beta_h = 1. \tag{24}$$

Here $\hat{L}_1(\boldsymbol{\theta})$ and $\hat{L}_{-1}(\boldsymbol{\theta})$ are *estimators* of the expected loss of a target and non-target trial respectively, and β_h is the weight for trial h. The expected loss of a trial with label t, $\mathrm{E}_{\boldsymbol{\Omega}^{(p)}|t}(L(\boldsymbol{\theta}))$, is not affected by the fact that the trials are statistically dependent. As long as Eq. (24) is fulfilled, $\tilde{P}_{\text{eff}} = P_{\text{eff}}$ therefore gives an unbiased estimate of the expected loss, $E_{T,\boldsymbol{\Omega}^{(p)}}(L(\boldsymbol{\theta}))$ (for any $\boldsymbol{\theta}$). In addition, we propose to adjust the trial weights, β_h, so that the variances of $\hat{L}_1(\boldsymbol{\theta})$ and $\hat{L}_{-1}(\boldsymbol{\theta})$ is minimized. This gives the best linear unbiased estimator (BLUE) (Kay, 1993, ch. 6) of the expected loss. From here on, we use $t \in [-1, 1]$ also as a suffix to indicate target or non-target trial. Let the i-vector pairs, $\boldsymbol{\Omega}_h^{(p)}$, of the training trials with label t be collected in a vector $\vec{\boldsymbol{\Omega}}_t \in \mathbb{R}^{2dN_t}$. Further, let the weights for the corresponding trials be collected in a vector $\boldsymbol{\beta}_t \in \mathbb{R}^{N_t}$, and let $\Sigma_t \in \mathbb{R}^{N_t \times N_t}$ be the covariance matrix for the losses of these trials.[2] Then,

[2] For simplicity, we do not consider the statistical dependencies between a target trial and a non-target trial in this study.

$\mathrm{var}\left[\hat{L}_t\left(\boldsymbol{\theta}, \vec{\boldsymbol{\Omega}}_t\right)\right]$ is given by

$$\begin{aligned}
&\mathrm{E}_{\vec{\Omega}_t|\mathcal{I}_t}\left(\hat{L}_t(\boldsymbol{\theta}, \vec{\boldsymbol{\Omega}}_t) - \mathrm{E}_{\vec{\Omega}_t|\mathcal{I}_t}\hat{L}_t(\boldsymbol{\theta}, \vec{\boldsymbol{\Omega}}_t)\right)^2 \\
&= \mathrm{E}_{\vec{\Omega}_t|\mathcal{I}_t}\left(\sum_{h:t_h=t}\beta_h l\left(t, \boldsymbol{\theta}, \boldsymbol{\Omega}_h^{(p)}\right) - \mathrm{E}_{\boldsymbol{\Omega}^{(p)}|t}L\left(\boldsymbol{\theta}\right)\right)^2 \\
&= \boldsymbol{\beta}_t^T \Sigma_t \boldsymbol{\beta}_t, \qquad (25)
\end{aligned}$$

where $\mathcal{I}_t$ denotes any information about how $\vec{\boldsymbol{\Omega}}_t$ is generated that affects Σ_t.

Previous studies have set β_h to $1/N_1$ for the target trials and $1/N_{-1}$ for the non-target trials. From Eq. (25) it is clear that when Σ_t is diagonal whose all elements are equal, this choice of β_h is optimal and results in the well-known formula for the variance of the sample mean of IID variables. However, when the trials are correlated, this choice of β_h is not optimal. By using a Lagrange multiplier to enforce the constraint in Eq. (24) it can be shown that, as long as Σ_t is non-singular,[3] the minimizer is given by,

$$\boldsymbol{\beta}_t = \frac{\Sigma_t^{-1}\mathbf{1}}{\mathbf{1}^T\Sigma_t^{-1}\mathbf{1}}, \qquad (26)$$

where $\mathbf{1}$ is a column-vector of length N_t whose all elements equal 1.

According to Eq. (26), the optimal $\boldsymbol{\beta}_t$ is not affected by a scaling of the covariance matrices. Since all target/non-target trial losses have the same variance, we therefore only need to know the correlation between the trials. We denote the correlation matrices $\boldsymbol{R}_t = \Sigma_t/v_t$, where v_1 and v_{-1} are the variances for the target and non-target trial losses respectively. These correlation matrices have one entry per observation, In order to estimate them, it is therefore necessary to impose a structure on them so that they depend on a small number of parameters. In Section 4, we propose how to do this for the correlations that arise from using the same speakers and utterances in several training trials.

It should be noticed that same results can be obtained by regarding the trial losses $\boldsymbol{l}_t(\boldsymbol{\theta}) = [l_1(\boldsymbol{\theta}), \ldots, l_{N_t}(\boldsymbol{\theta})]^T$ as one multivariate observation following normal distribution with mean $\boldsymbol{\eta} = [\eta, \ldots, \eta]^T$ and covariance matrix Σ_t and then using the ML estimate of η as loss estimator, i.e.,

$$\begin{aligned}
\hat{L}_t(\boldsymbol{\theta}) &= \arg\max_{\eta} \frac{1}{\sqrt{(2\pi)^{N_t}|\Sigma_t|}} \exp\left(-\frac{1}{2}(\boldsymbol{l}(\boldsymbol{\theta}) - \boldsymbol{\eta})^T\Sigma_t^{-1}(\boldsymbol{l}(\boldsymbol{\theta}) - \boldsymbol{\eta})\right) \\
&= \boldsymbol{l}_t(\boldsymbol{\theta})^T \frac{\Sigma_t^{-1}\mathbf{1}}{\mathbf{1}^T\Sigma_t^{-1}\mathbf{1}}. \qquad (27)
\end{aligned}$$

3.2. The effect of direct optimization of the PLDA LLR score function

Scr-UC optimizes the parameters of LLR score, $\boldsymbol{\gamma} = [\mathrm{vec}(\boldsymbol{P})^T, \mathrm{vec}(\boldsymbol{Q})^T, \boldsymbol{c}^T, k]^T$, directly, whereas the DT scheme proposed in Borgström and McCree (2013) optimizes the parameters of the PLDA model, $\boldsymbol{m}$, $\boldsymbol{V}$ and $\boldsymbol{D}$. The discriminative training objective in Eq. (18) depends on the scores, s_h, of the training trials. Since the scores according to Eq (16) are given by a linear function of $\boldsymbol{\gamma}$, direct optimization of $\boldsymbol{\gamma}$ is most straight-forward. However, if no constraints are imposed on $\boldsymbol{\gamma}$, this may result in a model with different properties than a PLDA model. The matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$ depends on the PLDA *between-class* covariance matrix, $\boldsymbol{V}\boldsymbol{V}^T$, and the *within-class* covariance matrix, $\boldsymbol{D}\boldsymbol{D}^T$, according to Eqs. (11) and (12). It is however not immediately apparent what constraints that follows on $\boldsymbol{\gamma}$. In this subsection, the constraints on $\boldsymbol{\gamma}$ are presented, as well as an analysis of their impact on the model. The discussion here is similar to the one in Rohdin et al. (2014a).

The matrices, $\boldsymbol{P}$ and $\boldsymbol{Q}$, are symmetric and have the same rank as $\boldsymbol{V}$ (Garcia-Romero and Espy-Wilson, 2011). In addition, it can be shown based on Eq. (11) and (12), that the matrices, $\boldsymbol{P}$ and $\boldsymbol{Q}$, are constrained as follows:

1. $\boldsymbol{P}$ is positive-(semi)definite.
2. $\boldsymbol{Q}$ is negative-(semi)definite.
3. $\boldsymbol{P} + \boldsymbol{Q}$ is positive-(semi)definite.

For these constraints, *semi* applies when the rank of $\boldsymbol{V}$ is smaller than d. The proofs are given in AppendixA.

Scr-UC preserves the symmetry of $\boldsymbol{P}$ and $\boldsymbol{Q}$ but relaxes the definiteness constraints. In the remainder of this subsection, the effects of these constraints on the model are analyzed. In Section 6, the impact of the constraints is evaluated experimentally.

The first constraint leads to a *directional property*. Consider an i-vector, $\boldsymbol{\omega}$, scored against both $\alpha\boldsymbol{\omega}$ and $-\alpha\boldsymbol{\omega}$, where α is a positive constant. That is, in the first trial, $\boldsymbol{\omega}$ is scored against an i-vector pointing in same direction and, in the second trial it is scored against an i-vectors pointing in the opposite direction. Let $s(\boldsymbol{\omega}_i, \boldsymbol{\omega}_j) = s_{ij}$ in Eq. (10). If the i-vectors are centered around $\boldsymbol{m}$, the difference between the scores of these two trials is

$$s(\boldsymbol{\omega}, \alpha\boldsymbol{\omega}) - s(\boldsymbol{\omega}, -\alpha\boldsymbol{\omega}) = 4\alpha\boldsymbol{\omega}^T\boldsymbol{P}\boldsymbol{\omega}. \qquad (28)$$

In other words, the score of the *same direction* trial will be guaranteed to be larger than the score of the *different direction* trial if and only if $\boldsymbol{P}$ is positive definite.

The second constraint leads to a *length property*:

$$s(\boldsymbol{\omega}, \boldsymbol{\omega}) > s(\alpha\boldsymbol{\omega}, \frac{1}{\alpha}\boldsymbol{\omega}). \qquad (29)$$

This property means that two i-vectors of equal length and direction will obtain a higher score than two i-vectors having just equal direction.

From the first and the second constraint, it follows directly that $\boldsymbol{P}-\boldsymbol{Q}$ is positive-definite. Together with the third constraint, this leads to the following properties:

$$s(\boldsymbol{\omega}, \boldsymbol{\omega}) > s(\mathbf{0}, \mathbf{0}), \qquad (30)$$

$$s(\boldsymbol{\omega}, -\boldsymbol{\omega}) < s(\mathbf{0}, \mathbf{0}). \qquad (31)$$

This means that any two i-vectors pointing in the same direction obtain a higher score than any two i-vectors pointing in opposite direction. These two properties are therefore a stronger version of the directional property.

[3] A singular Σ would mean that the losses of two trials have correlation 1 or that the variance of a trial loss is 0. These are not realistic scenarios.

Table 1: Different kinds of trial pairs and the notation of their correlation. Capital letters refer to speakers and their indices refer to utterances. 'Corr' is the notation of the correlation coefficient.

Set	Things in common	Trial pair example	Corr.
Target	1 utt.	(A_1, A_2) - (A_1, A_3)	c_a
	Spk.	(A_1, A_2) - (A_3, A_4)	c_b
	Nothing	(A_1, A_2) - (B_1, B_2)	0
Non-target	1 utt., 1 spk.	(A_1, B_1) - (A_1, B_2)	c_{-a}
	2 spk.	(A_1, B_1) - (A_2, B_2)	c_{-b}
	1 utt.	(A_1, B_1) - (A_1, C_1)	c_{-c}
	1 spk.	(A_1, B_1) - (A_2, C_1)	c_{-d}
	Nothing	(A_1, B_1) - (C_1, D_1)	0

4. Compensation for statistically dependent trials

In Subsection 3.1, we showed that using an equal weight for all target trials and another equal weight for the non-target trials is typically not optimal when the trials are statistically dependent. We argued that it is preferable to adjust the weights of the trials to obtain the BLUE for the loss estimator, $\hat{L}(\boldsymbol{\theta})$, and showed that in order to do this, we need to know the correlation between the losses of the training trials. In this section, we propose practical methods for *weight-adjustment* of training trials that are statistically dependent due to each speaker and utterance being used in more than one trial.

4.1. Weight-adjustment formulas

Let capital letters denote different speakers in our training data. Let N_X be the number of utterances of speaker X, X_i be the i-th utterance of this speaker, and $l(X_i, Y_j, \boldsymbol{\theta})$ be the loss for the trial involving utterance X_i and Y_j. We would like to make some assumptions about Σ_1 and Σ_{-1} that allow us to calculate the optimal weights $\boldsymbol{\beta}_t$ by means of Eq. (26). Since the optimal weight vector only depends on the correlation between the trial losses, we can let the variances of the trial losses be functions of $\boldsymbol{\theta}$. However, if the correlation coefficients depend on $\boldsymbol{\theta}$, the optimal weights will also depend on $\boldsymbol{\theta}$. For simplicity, we therefore assume that the correlation coefficients do not depend on $\boldsymbol{\theta}$, but only on what the trials have in common. For example, we assume:

$$\mathrm{corr}\Big(l(A_1, A_2, \boldsymbol{\theta}), l(A_1, A_3, \boldsymbol{\theta})\Big) = c_a, \tag{32}$$

where the correlation coefficient, c_a, is the same for all target trial pairs that share one utterance. All the possible relations between two trials as well as the notation for the correlation coefficients are given in Table 1. Notice that two trials that have nothing in common are statistically independent so that, e.g., $\mathrm{corr}\left(l(A_1, A_2, \boldsymbol{\theta}), l(B_1, B_2, \boldsymbol{\theta})\right) = 0$.

In this study, we use all the trials that can be constructed from the training data except those where both the utterances are the same. Under the assumptions given in the previous paragraph, the optimal weight for each target trial of speaker A is then given by (see AppendixB.1 for proof)

$$\beta_A = \frac{k_1}{1 + 2(N_A - 2)c_a + (N_A - 2)(N_A - 3)c_b/2}, \tag{33}$$

where k_1 is set so that the sum of the weights equals 1. Since we do not use target trials where both the utterances are the same, a speaker with only one utterance is never used for target trials, i.e., $N_A = 1$ is never used in the above formula. For a speaker with two utterances, there is only one unique target trial so the second and third term in the denominator will be 0. For a speaker with three utterances, we can construct two trials with one shared utterance but not two trials with no shared utterances. In this case, the third term in the denominator will be 0. Notice that if all correlations equals 0, or if each speaker has the same number of utterances, each trial will obtain the same weight. If all correlations equal 1, each speaker will obtain the same weight.

In order to derive the weights for the non-target trials, we do some approximations. The *approximately* optimal weight for each non-target trial of speaker A and B, is then

$$\beta_{AB} \approx \frac{k_{-1}}{W_{AB}} \tag{34}$$

where k_{-1} is set so that the sum of the weights equals 1 and,

$$\begin{aligned} W_{AB} = 1 &+ c_{-a}(N_A + N_B - 2) \\ &+ c_{-b}(N_A - 1)(N_B - 1) \\ &+ (2c_{-c} + c_{-d}(N_A + N_B - 2)) \sum_{I \neq A,B} N_I. \end{aligned} \tag{35}$$

The derivation including the approximations is given in AppendixB.2.

4.2. Estimation of correlation coefficients

Ideally, we would have knowledge about c_a, c_b, c_{-a}, c_{-b}, c_{-c} and c_{-d}. In this study we explore two ways to find their values. The first is to approximate them with functions that depend on one tunable parameter. The second is to estimate them based on sample correlations in the training data.

4.2.1. Estimation by a one-parameter model

Consider first the target trials. We assume that two target trials from the same speaker are correlated, and that two target trials where one utterance is the same are more correlated, so that $0 \leq c_b \leq c_a \leq 1$. In order to obtain only one parameter to tune we set

$$\begin{aligned} c_a &= \alpha_1, \\ c_b &= \alpha_1^2, \end{aligned} \tag{36}$$

where $0 \leq \alpha_1 \leq 1$ will be tuned. The weights according this formula for different values of α_1, are given in Fig. 1. Notice that, even for small values of α_1, the number of utterances of a speaker has large impact on the optimal weights for that speaker.

For the non-target trials, we can apply the same strategy, i.e., assume that a shared speaker makes the trials correlated and a shared utterance makes them more correlated. However, the relation between c_{-b} and c_{-c} is not clear. The former gives the correlation between non-target trial losses where both speakers are the same. The latter gives the correlation between non-target trial losses that has one common utterance. While

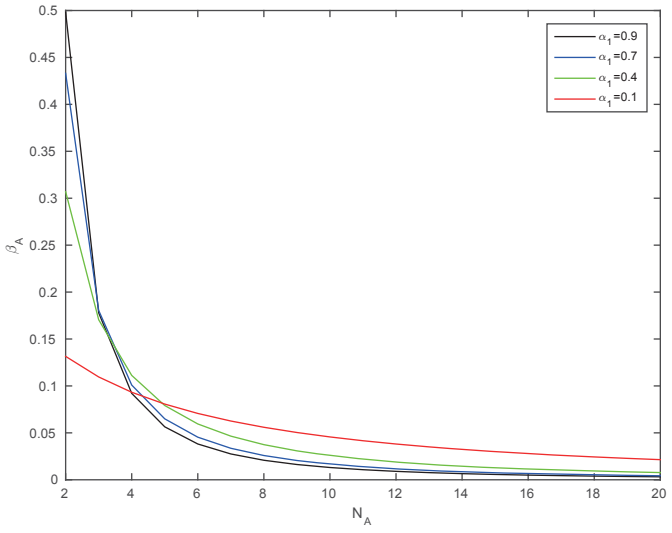

Figure 1: Optimal target trial weights for speaker A. 'N_A' is the number of utterances for the speaker and 'β_A' is the weight according to Eqs. (33) and (36).

it is clear that having one common utterance should give larger correlation than only having one common speaker, we cannot guess the relation between *one same utterance* and *two same speakers*. Therefore, we set

$$c_{-a} = \alpha_{-1},\ c_{-b} = \alpha_{-1}^2,\ c_{-c} = \alpha_{-1}^2,\ c_{-d} = \alpha_{-1}^3, \qquad (37)$$

with $0 \leq \alpha_{-1} \leq 1$. In this study, we will make a further simplification and set, $\alpha_1 = \alpha_{-1}$, denoted α from here on. This parameter will be tuned on a development set which to some extent can compensate for inaccuracies introduced by our assumptions and approximations.

4.2.2. Estimation by sample correlation

Let $\bar{l}_1(\boldsymbol{\theta})$ and $\bar{v}_1(\boldsymbol{\theta})$ be the sample mean and sample variance of the loss of the target trials for the parameter $\boldsymbol{\theta}$. Given N_a target trial pairs with one utterance in common (see Table 1), we calculate the sample correlation for those trials as

$$\bar{c}_a = \frac{1}{\bar{v}_1(\boldsymbol{\theta}) N_a} \sum_{h=1}^{N_a} \left(l(\boldsymbol{\omega}_1^{(h)}, \boldsymbol{\omega}_2^{(h)}, \boldsymbol{\theta}) - \bar{l}_1(\boldsymbol{\theta}) \right) \times \left(l(\boldsymbol{\omega}_1^{(h)}, \boldsymbol{\omega}_3^{(h)}, \boldsymbol{\theta}) - \bar{l}_1(\boldsymbol{\theta}) \right). \qquad (38)$$

The sample correlations for the other correlation coefficients are calculated analogously. Compared to the one-parameter model, this method makes fewer assumptions about the data. It relies on the assumption that the correlation coefficients are independent of $\boldsymbol{\theta}$ and the same for each trial of the same kind (as defined in Table 1). On the other hand, since the correlation coefficients are not tuned for performance, but to fit the data, this method may be more sensitive for incorrect model assumptions. For this method, we estimate the sample correlations based on trial losses calculated with the corresponding DT model without weight-adjustment.

5. Constrained discriminative PLDA training

In order to avoid overfitting, we would like to constrain the PLDA model during DT. One way to do this is to apply L2 regularization. Finding the right regularization is however difficult. Letting the regularization parameter be equal for all model parameters, as is typical, could be far from optimal. On the other hand, tuning many different regularization parameters is complicated. Similarly to Borgström and McCree (2013), we therefore propose several training schemes where a small number of parameters estimated by DT are used to adjust the model estimated by GT. This approach is in the spirit of AT-Cal but the training schemes we propose are less constrained. Based on the discussion in Subsection 3.2, we also propose a DT scheme that preserves the properties of $\boldsymbol{P}$ and $\boldsymbol{Q}$. The gradients for the training schemes we propose can be derived based on the gradients for Scr-UC given in Cumani et al. (2011). These calculations are efficient enough for using all possible combinations of the training utterances as training data. The details of gradient calculations as well as the initializations are given in AppendixC.

5.1. Reducing the number of parameters to be estimated

As an option with $O(1)$ parameters, we propose to scale each part of the PLDA LLR score function:

$$s_{ij} = a_{\mathrm{P}} \boldsymbol{\omega}_i^T \boldsymbol{P} \boldsymbol{\omega}_j + a_{\mathrm{P}} \boldsymbol{\omega}_j^T \boldsymbol{P} \boldsymbol{\omega}_i + a_{\mathrm{Q}} \boldsymbol{\omega}_i^T \boldsymbol{Q} \boldsymbol{\omega}_i + a_{\mathrm{Q}} \boldsymbol{\omega}_j^T \boldsymbol{Q} \boldsymbol{\omega}_j + a_{\mathrm{c}} (\boldsymbol{\omega}_i + \boldsymbol{\omega}_j)^T \boldsymbol{c} + a_{\mathrm{k}} k, \qquad (39)$$

where a_{P}, a_{Q}, a_{c} and a_{k} are trained discriminatively. In other words, we let the discriminative training adjust the weight of each *feature kind* in the original model parameters. The definiteness properties $\boldsymbol{P}$ and $\boldsymbol{Q}$ given in Subsection 3.2 were, in our experiments, almost always satisfied by itself (see Subsection 6.4), so we did not add any other constraints for this purpose. We refer to this method as Scr-4par. It should be noted that if $a_{\mathrm{P}} = a_{\mathrm{Q}} = a_{\mathrm{c}}$ in Eq. (39), we obtain AT-Cal.

As an option with $O(d)$ parameters, we propose to scale all the elements of the i-vector. Either one scaling for each of $\boldsymbol{P}$, $\boldsymbol{Q}$ and $\boldsymbol{c}$, or a common scaling can be used. In either case, we use a scaling of k. Accordingly this gives $3d+1$ or $d+1$ parameters to be estimated. In this study, we use the latter and refer to it as iV-elmnt. Another natural option with $O(d)$ parameters is to scale the eigenvalues of $\boldsymbol{P}$ and $\boldsymbol{Q}$, but the advantage of iV-elmnt is that we do not have to consider whether to preserve the PLDA properties.

For generative ML training, letting $\mathrm{rank}(\boldsymbol{V}) = r < d$ has been reported to be beneficial (Matejka et al., 2011). As explained in Subsection 3.2, this reduces the rank of $\boldsymbol{P}$ and of $\boldsymbol{Q}$ from d to r as well. Based on results in (Bishop, 2006, p. 577) it follows that the number of parameters to be estimated will be reduced from $d^2+2d+1 = O(d^2)$ to $r(2d-r)+d+r+1 = O(dr)$. However, a large reduction of the number of parameters in this way limits the model too much. As an extreme example, if we want the same number of parameters as iV-elmnt, we have to set $r = 1$, which means that the i-vectors are projected into a one-dimensional space.

5.2. Preserve the properties of $\boldsymbol{P}$ and $\boldsymbol{Q}$

As the least constrained option, we propose to just preserve the definiteness constraints of $\boldsymbol{P}$ and $\boldsymbol{Q}$. In this case the number of parameters to be estimated is not reduced, but the values they can take are limited. In this subsection, we propose reparameterizations of of $\boldsymbol{P}$ and $\boldsymbol{Q}$ such that when these parameters are optimized instead of $\boldsymbol{P}$ and $\boldsymbol{Q}$, the definiteness constraints will be preserved.

The matrix $\boldsymbol{P}$ is positive-semidefinite if

$$\boldsymbol{P} = \boldsymbol{P}_A \boldsymbol{P}_A^T, \tag{40}$$

where $\boldsymbol{P}_A$ is a $d \times r$ matrix with real elements. Accordingly, in order to keep $\boldsymbol{P}$ positive-semidefinite, we train $\boldsymbol{P}_A$ instead of $\boldsymbol{P}$. The rank of $\boldsymbol{P}$ is equal to r and can therefore be selected by selecting the number of columns in $\boldsymbol{P}_A$. If we wish to control the rank without keeping $\boldsymbol{P}$ positive definite, we can use $\boldsymbol{P} = \boldsymbol{P}_A \boldsymbol{P}_B$ where $\boldsymbol{P}_B \neq \boldsymbol{P}_A^T$.

In order to keep $\boldsymbol{Q}$ negative-semidefinite, we set

$$\boldsymbol{Q} = -\boldsymbol{Q}_A \boldsymbol{Q}_A^T, \tag{41}$$

and train $\boldsymbol{Q}_A$ instead of $\boldsymbol{Q}$.

In order to enforce the third constraint, we set

$$\boldsymbol{P} + \boldsymbol{Q} = \boldsymbol{R}_A \boldsymbol{R}_A^T, \tag{42}$$

instead of $\boldsymbol{P} = \boldsymbol{P}_A \boldsymbol{P}_A^T$, and as before

$$\boldsymbol{Q} = -\boldsymbol{Q}_A \boldsymbol{Q}_A^T. \tag{43}$$

We then optimize $\boldsymbol{R}_A$ and $\boldsymbol{Q}_A$. In this study, we apply the three definiteness constraints without reducing the rank. We refer to this method as Scr-Def.

5.3. Regularization

For the DT schemes with many parameters, we also apply L2 regularization in order to prevent overfitting, i.e., we add the term $\rho\|\boldsymbol{\theta} - \tilde{\boldsymbol{\theta}}\|^2$ to the training objective in Eq. (21), where $\|\cdot\|$ denotes the Euclidean norm and the regularization parameter, ρ, is tuned on a development set. This forces the parameter vector, $\boldsymbol{\theta}$, to be close to $\tilde{\boldsymbol{\theta}}$. For Scr-UC and Scr-Def, we use either $\mathbf{0}$ or the model from GT. For Scr-Def, we use regularization in terms of $\boldsymbol{P}$ and $\boldsymbol{Q}$ rather than $\boldsymbol{R}_A$ and $\boldsymbol{Q}_A$. For example, the contribution to the regularization term from $\boldsymbol{Q}$ is

$$\rho\|\mathrm{vec}(\boldsymbol{Q} - \tilde{\boldsymbol{Q}})\|^2 = \rho\|\mathrm{vec}(-\boldsymbol{Q}_A \boldsymbol{Q}_A^T - \tilde{\boldsymbol{Q}})\|^2, \tag{44}$$

rather than $\rho\|\mathrm{vec}(\boldsymbol{Q}_A - \tilde{\boldsymbol{Q}}_A)\|^2$, where $\tilde{\boldsymbol{Q}}$ and $\tilde{\boldsymbol{Q}}_A$ denotes either $\mathbf{0}$ or the parameters obtained by GT.

The optimal ρ depends on the size of the training data. Instead of tuning ρ for each training data size, we use a modified training objective given by

$$\hat{L}'(\boldsymbol{\theta}) = \kappa \hat{L}(\boldsymbol{\theta}) + \rho\|\boldsymbol{\theta} - \tilde{\boldsymbol{\theta}}\|^2 \tag{45}$$

where $\kappa = N_1 + N_{-1}$. We then tune ρ for the full training data and use this value also for smaller amounts of training data. This means that the influence of the regularization becomes larger for the smaller training data.

6. Experiments

6.1. Outline

We first evaluated the weight-adjustment proposed in Section 4 applied to AT-cal. We then compared the constrained DT schemes proposed in Section 5 with the two baselines, AT-Cal and Scr-UC. Finally, we compared the combination of weight-adjustment and the best constrained DT scheme with the best baseline for various training data sizes.

We mainly focused on the *logarithmic cost function* ($\hat{C}_{\mathrm{llr}}$) introduced in Brümmer and du Preez (2006) as evaluation metric. In addition, we report results on several other standard evaluation metrics, namely *equal error rate* (EER), *actual* and *minimum detection costs* (actDCF, minDCF) for the effective prior used in the NIST SRE 2008 and the NIST SRE 2010 (NIST, 2008, 2010), as well as minimum $\hat{C}_{\mathrm{llr}}$ ($\hat{C}_{\mathrm{llr}}^{\mathrm{min}}$). Notice that EER, minDCF and $\hat{C}_{\mathrm{llr}}^{\mathrm{min}}$ are *calibration-insensitive* evaluation metrics, i.e., they are not affected by monotonic transformations of the scores such as the affine transformation employed by AT-Cal.

In the next subsection, the details of the experimental set-up are given. In Subsection 6.3, the experimental results are given. Finally, an analysis of the results is given in Subsection 6.4.

6.2. Experimental set-up

We conducted experiments on the male part of three sets, the NIST SRE 2006 core task (SRE06), NIST SRE 2008 core task condition-6 (SRE08) and NIST SRE 2010 core task condition-5 extended (SRE10). We used SRE06 as the development set for tuning the regularization parameter, ρ, and for the weight-adjustment parameter, α. For some experiments (see subsection 6.3.1), we also used SRE06 for calibration. SRE08 and SRE10 were used as the evaluation sets. A few trials in SRE06 and SRE08 were excluded because of their inconsistent meta-data. The number of trials were 22123, 12356 and 179338 for SRE06, SRE08 and SRE10 respectively. It should be noted that SRE08 could be too small to give a reliable estimate of actDCF10. The evaluation metrics were calculated with the BOSARIS toolkit (Brümmer and de Villiers, 2011) which uses the PAV algorithm for calculating the minimum version of the evaluation metrics.

For training the UBM and the $\boldsymbol{T}$ matrix, we used NIST SRE 2004 (SRE04), NIST SRE 2005 (SRE05), Switchboard II Phase 1 (SB2P1), Switchboard II Phase 2 (SB2P2), Switchboard II Phase 3 (SB2P3), Switchboard Cellular Part 1 (SBCP1) and Switchboard Cellular Part 2 (SBCP2). For SRE04, we used speech files included in the training lists of one, three, eight and sixteen single-channel conversation sides and in the test list of one single-channel conversation side. For SRE05, we used speech files included in the training lists of one, three and eight two-channel conversation sides and in the test list of one single-channel conversation side. For the Switchboard datasets, we used all non-empty speech files.

For training PLDA models, we used the same data except SB2P1. In addition, from the Switchboard data, we excluded speech distorted by *echo* or *crosstalk* or *background noise* according to the meta-data in the databases. MIXER PIN and PIN

were used as unique speaker IDs for NIST SRE and Switchboard datasets respectively. For the files whose MIXER PIN were missing, we used model IDs as speaker IDs. This gave 1153 speakers with in total 9152 utterances.

We used 15 PLP coefficients (Hermansky, 1990) along with log-energy and applied feature warping (Pelecanos and Sridharan, 2001). After that, we appended the first-order and second-order derivatives, resulting in 48 elements per frame. Non-speech parts were then removed by using a spectral subtraction-based voice activity detector (Mak and Yu, 2010). Our UBM had 2048 Gaussian components and d, i.e., the rank of $\boldsymbol{T}$, was set to 400. The i-vectors went through the process of centering, whitening, and length-normalization (Garcia-Romero and Espy-Wilson, 2011).

Generative PLDA training was performed with the EM algorithm (Brümmer, 2010). The number of columns of $\boldsymbol{V}$ was set to d. For the discriminative training methods, we used the L-BGFS (Liu and Nocedal, 1989) implementation in Schmidt (2012). We used its default stopping criteria and in addition, we stopped the training if no change in minDCF08 had been observed on the development set for 20 iterations. As in Burget et al. (2011) and Cumani et al. (2011), we used all the trials that could be constructed from the training data, except that we excluded target trials where an utterance is scored against itself. The number of unique target trials in the training data was 52,709 and the number of unique non-target trials was 41,822,267. We used the effective prior of SRE08, $P_{\text{eff}} = 0.0917$, to balance target and non-target trials and for setting τ. The weight-adjustment parameter, α, was optimized over the steps $0, 0.1, \ldots, 1.0$. Sample correlations were estimated based on the losses of 10^6 trial pairs of each kind (sampled with replacement). The regularization parameter, ρ, was optimized over the steps $10^{-3}, 10^{-2}, \ldots, 10^4$.

6.3. Results

6.3.1. Weight-adjustment for AT-Cal

For the initial exploration, we first evaluated the weight-adjustment with the one-parameter model for AT-Cal. We trained a PLDA model with the training data described in Subsection 6.2, and used data from the test set of SRE06 for calibration. We selected the calibration data in a way that the effect of the weight-adjustment should be easily observed, i.e., few speakers with large variation in their number of utterances. Specifically, we randomly selected 11, 14 or 21 speakers, and then for each of them, we randomly selected their number of utterances uniformly in the interval 1 to the number of available utterances (between 1 and 36, around 6 on average). Notice that this choice of calibration data was for demonstrating the effect of weight-adjustment. It is generally better to use all the available data. The actDCFs and $\hat{C}_{\text{llr}}$ for the α that was the optimal on the development set, as well as $\alpha = 0$ which gives the standard equal weight to each trial, are shown in Table 2. For reference, the result in the calibration-insensitive evaluation metrics are given in Table 4. In Figure 2, $\hat{C}_{\text{llr}}$ vs. α is shown. We observed a large improvement in $\hat{C}_{\text{llr}}$ for 11 calibration speakers. The optimal α on the development set was 0.5 but any value in between 0.1 and 0.6 gave similar results. For 14 and 21 speakers, the improvements were marginal. A general rule for the optimal value of α is therefore not possible to infer from this experiment.

Table 2: Calibration results using SRE06 as calibration data. $\alpha = 0$ is the standard approach with equal weight to each trial. A '*' indicates that this value was optimal for $\hat{C}_{\text{llr}}$ on the development set.

Set	#Spkr	actDCF08	actDCF10	$\hat{C}_{\text{llr}}$	α
SRE08	11	0.0270	0.01525	3.660	0
		0.0271	0.01500	2.440	*0.5
	14	0.0343	0.00688	0.256	0
		0.0334	0.00552	0.245	*0.1
	21	0.0316	0.00266	0.215	0
		0.0291	0.00281	0.211	*1.0
SRE10	11	0.0113	0.001556	2.124	0
		0.0110	0.001518	1.363	*0.5
	14	0.0104	0.000468	0.087	0
		0.0104	0.000413	0.085	*0.1
	21	0.0103	0.000462	0.087	0
		0.0102	0.000452	0.082	*1.0

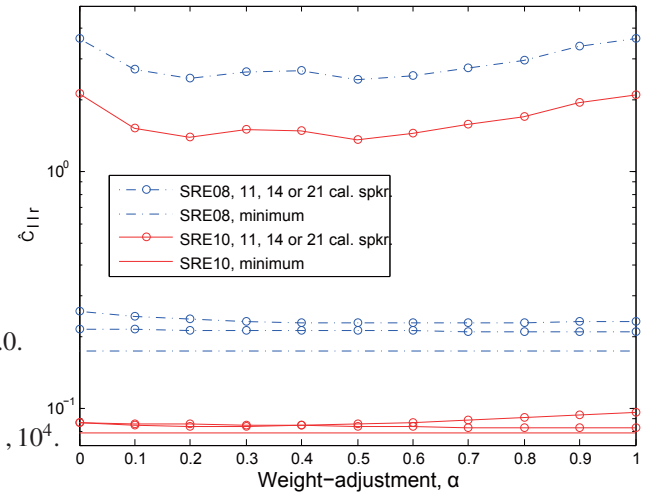

Figure 2: $\hat{C}_{\text{llr}}$ vs. the weight-adjustment parameter α. The lines without circles denote $\hat{C}^{\text{min}}_{\text{llr}}$. Lines with circles denote $\hat{C}_{\text{llr}}$ for 11 (upper), 14 (middle) and 21 (lower) calibration speakers, respectively.

The differences in actDCF08 were insignificant in most cases. Both actDCF08 and actDCF10 consider a small value of P_{eff} (0.0917 and 0.0010 respectively). $\hat{C}_{\text{llr}}$ on the other hand, considers all values of P_{eff} (Brümmer and du Preez, 2006). For the training sets with 11, 14 and 21 speakers, the proportion of target-trials were 0.0774, 0.0736 and 0.0583 respectively. These values are close to P_{eff} of actDCF08. The expected loss is therefore likely to be better estimated for this value of P_{eff} than others, so that the benefit of an improved estimation procedure becomes smaller.

In the above experiment we used additional data for the calibration. Results using only the original training data for weight-adjustment both based on the one-parameter model and based on sample correlations are shown in Table 3. We considered the following training conditions:

1. Use the data from 90% of the training speakers for model

Table 3: Calibration results for three training/calibration conditions. The conditions are described in Subsubsection 6.3.1. 'W.-adj' refers to weight-adjustment, 'sp.' to sample correlation, and 'α' refers to the one-parameter model, tuned for $\hat{C}_{\text{llr}}$ on the development set.

Set	Cond.	actDCF08	actDCF10	$\hat{C}_{\text{llr}}$	W.-adj.
SRE08	1	0.0267	0.00151	0.286	no
		0.0278	0.00146	0.268	$\alpha = 0.2$
	2,3	0.0256	0.00130	0.201	no
	2	0.0251	0.00130	0.199	sp.
	2	0.0253	0.00131	0.197	$\alpha = 1.0$
	3	0.0251	0.00130	0.196	$\alpha = 1.0$
SRE10	1	0.0178	0.000623	0.168	no
		0.0182	0.000658	0.158	$\alpha = 0.2$
	2,3	0.0143	0.000678	0.100	no
	2	0.0141	0.000678	0.098	sp.
	2	0.0141	0.000688	0.098	$\alpha = 1.0$
	3	0.0135	0.000678	0.095	$\alpha = 1.0$

training, and the data from the remaining training speakers for calibration.

2. Use all training data both for model training and for calibration.

3. As Condition 2, but in addition preserve the balance between the NIST SRE and the Switchboard corpora when weight-adjustment is utilized.

Condition 3 was motivated by the fact that the data is made-up by several different corpora and as a side-effect of weight-adjustment, the balance between these corpora may change. In fact, the Switchboard corpora has much fewer utterances per speaker than the NIST SRE corpora. Since the weight-adjustment increases the weights for speakers with fewer trials, the weight for Switchboard is increased. Again, we confirmed that the weight-adjustment is effective, although the effect was smaller than in our previous experiment. We did not see any significant difference between weight-adjustment based on the one-parameter model and weight-adjustment based on sample correlations. It was overall better to use all data both in model training and in calibration, than to split the data. Using calibration trials from i-vectors that have been used in PLDA training is not ideal since it does not resemble the test situation where the trials are from new i-vectors, while in our experiments, the benefit of having more data for training outweighed this problem. Preserving the balance between NIST SRE and Switchboard was useful, in which case we obtained an relative improvement in $\hat{C}_{\text{llr}}$ of 5% by weight-adjustment on SRE10.

6.3.2. Comparison of DT schemes

Next, we evaluated the different discriminative PLDA training schemes without weight-adjustment. Since in the previous experiment, using all training data both for the GT step and the DT step was better in almost all cases, we continued to use this approach. The results are shown in Table 5. If we ignore the methods for which regularization towards the GT model was applied, there is a clear pattern in Table 5. Overall, one of the baselines, AT-Cal, performed best for SRE08 and Scr-4par performed best for SRE10, where the relative improvement over AT-Cal was 14%. The less constrained iV-elmnt performed worse than these two methods but better than Scr-UC which has no constraints except regularization.

Table 4: Results of GT in the calibration insensitive evaluation metrics. '%Spkr' is the percentage of the training speakers used for model training. 'm.' refers to minimum.

Set	% Spkr	m.DCF08	m.DCF10	$\hat{C}_{\text{llr}}^{\text{min}}$	EER
SRE 08	100	0.0250	0.000728	0.175	0.0480
	90	0.0254	0.000713	0.176	0.0497
SRE 10	100	0.0101	0.000385	0.079	0.0198
	90	0.0103	0.000403	0.081	0.0201

For Scr-Def and Scr-UC, we had to apply regularization in order to avoid overfitting. The regularization parameter, ρ was chosen to minimize $\hat{C}_{\text{llr}}$ on the development set, SRE06. In terms of $\hat{C}_{\text{llr}}$, these two methods performed worse than AT-Cal and Scr-4par. Applying regularization towards the GT model, gives mixed results. Both SCR-Def and SCR-UC performed well in actDCF08 but they did not perform well in actDCF10 and $\hat{C}_{\text{llr}}$. This indicates that these systems are only good for the effective prior that has been specified in the training objective. The bad performance for the other effective priors is, however, surprising since the logistic regression loss function emphasizes on a broad range of effective priors (Brümmer and du Preez, 2006). Moreover, the optimal ρ was quite large and the effect on minDCF08 was minor. It should also be noted that the optimal ρ varies depending on which evaluation metric is considered. In particular, this was a problem for Scr-Def with regularization towards $\mathbf{0}$ so we did not include that result in the table. The problem of this method method might be because its objective function is non-convex.

All the results taken into account, AT-cal and Scr-4par seems to be the best methods for this amount of training data or smaller.

6.3.3. Weight-adjustment for Scr-4par and Scr-UC

We have already confirmed that the weight-adjustment improves the performance of At-Cal. In this experiment we explore the effect of weight-adjustment on Scr-4par and Scr-UC with regularization towards $\mathbf{0}$. The former is important because this DT scheme performed the best on SRE10. The latter is interesting since it is the least constrained scheme that does not use the model obtained by GT in any way. For Scr-UC with weight-adjustment, we used the same regularization as for Scr-UC without weight-adjustment (which was tuned on the development set). The results are given in Table 6. For Scr-4par with the one-parameter model, we did not obtain any improvement in $\hat{C}_{\text{llr}}$ on the development set. Therefore, only the $\alpha = 0$ is included. The effect of weight-adjustment based on sample correlations were small. For Scr-UC the effect of weight-adjustment was larger in particular actDCF08 where we observed improvements of around 5% and 8% for SRE08 and SRE10 respectively. The difference between the weight-adjustment based on

Table 5: Discriminative training results. 'Reg. to GT' and 'Reg. to **0**' mean L2 regularization towards the model obtained by GT and regularization towards **0**, respectively. The regularization parameter, ρ, was tuned to optimize for $\hat{C}_{\text{llr}}$ on the development set.

Set	Method	actDCF08	minDCF08	actDCF10	minDCF10	EER	$\hat{C}_{\text{llr}}$	$\hat{C}_{\text{llr}}^{\text{min}}$	ρ
SRE08	AT-Cal	**0.0256**	**0.0250**	0.00130	0.000728	0.0480	**0.201**	**0.175**	-
	Scr-4par	0.0274	0.0254	0.00257	0.000802	0.0471	0.202	0.177	-
	iV-elmnt	0.0269	0.0262	0.00171	**0.000669**	0.0478	0.225	0.182	-
	Scr-Def. Reg. to GT	0.0269	0.0253	0.01278	0.000809	0.0461	1.415	0.178	10^2
	Scr-UC. Reg. to GT	0.0268	0.0253	0.01269	0.000809	**0.0459**	1.416	0.178	10^2
	Scr-UC. Reg. to **0**	0.0334	0.0304	**0.000876**	0.000743	0.0564	0.235	0.212	10^1
SRE10	AT-Cal	0.0143	0.0101	0.000678	0.000385	0.0198	0.100	0.0788	-
	Scr-4par	0.0117	**0.0100**	0.000574	**0.000375**	**0.0188**	**0.086**	**0.0744**	-
	iV-elmnt	0.0146	0.0121	**0.000563**	0.000412	0.0253	0.119	0.0962	-
	Scr-Def. Reg. to GT	**0.0110**	0.0103	0.001523	0.000377	0.0204	0.663	0.0805	10^2
	Scr-UC. Reg. to GT	0.0111	0.0103	0.001495	0.000380	0.0203	0.664	0.0801	10^2
	Scr-UC. Reg. to **0**	0.0304	0.0183	0.000916	0.000598	0.0370	0.180	0.1368	10^1

the one-parameter model and the weight-adjustment based one sample correlations were as in previous experiments small. In general, the minimum costs are much less effected by weight-adjustment than the actual costs. It should be noticed that the training objective aims at reducing actual costs. The minimum costs are only indirectly affected since they cannot be higher than the actual costs.

6.3.4. Different training data sizes

In the final experiment, we evaluated AT-Cal, Scr-4par and SCR-UC for smaller numbers of training speakers, with and without weight-adjustment. For simplicity, we did not preserve the balance between the NIST SRE and the Switchboard corpora. The same training data was used both in the GT step and the DT step. Since previous experiments showed very small differences between weight-adjustment based on the one-parameter model based on sample correlations, we use only the former in this experiment. In Table 7 the results using half of the training speakers are shown. Scr-UC benefited mostly from weight-adjustment where actDCF08 improved around 10% for both SRE08 and SRE10, and $\hat{C}_{\text{llr}}$ improved around 6% for both SRE08 and SRE10.

In Fig. 3, $\hat{C}_{\text{llr}}$ vs. the number of training speakers is shown for SRE10. It is clear that Scr-4par gave better results than AT-cal and that the weight-adjustment in most cases improved the performance of both methods. The relative improvements of Scr-4par with weight-adjustment compared to the baseline, AT-Cal without weight-adjustment, ranged from 7% to 19% for the different training sizes. It is interesting that the gap between the two methods became larger when the amount of training data increased. This is reasonable since more training data is needed in order to take advantage of the extra flexibility of Scr-4par. However, in accordance with the experiment in Subsubsection 6.3.2, AT-Cal was better on SRE08 in most cases. As in the experiments with AT-Cal in Subsubsection 6.3.1, the effect of weight-adjustment disappears when the number of training speakers became large. For other evaluation metrics than $\hat{C}_{\text{llr}}$, the trend was less clear.

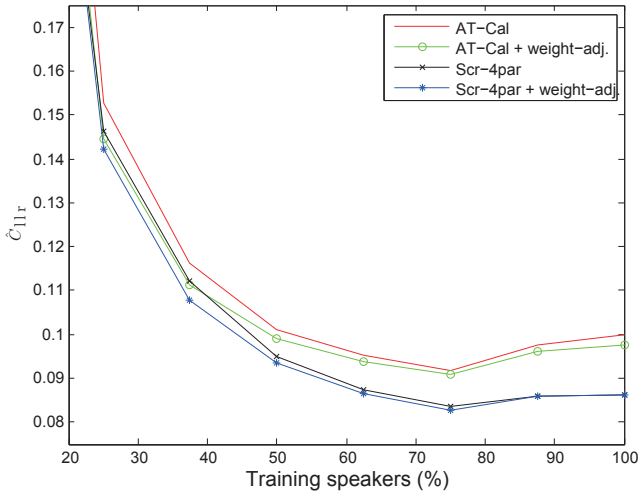

Figure 3: $\hat{C}_{\text{llr}}$ for SRE10 vs. the percentage of training speakers. 100% equals 1152 speakers. The weight-adjustment parameter, α, was chosen to be optimal for the development set for each training-size.

6.4. Analysis

In this subsection, we analyze some of the results more in detail. In particular, we investigate how accurate the assumptions leading to the weight-adjustment formulas in Eqs. (33) and (35) are, and whether the definiteness properties of $\boldsymbol{P}$ and $\boldsymbol{Q}$ discussed in Subsection 3.2 are important.

The weight-adjustment did not always work well. For example, when all training data was used for Scr-4par, the optimal value of α on the development set was 0, which corresponds to not using any weight-adjustment. The fact that the weight-adjustment did not always work may indicate that the assumptions behind it are not accurate enough. Let us recall the assumptions. First, we assumed that the correlation between the losses of two trials does not depend on the model parameters, $\boldsymbol{\theta}$, but only on what the trials have in common, e.g., one utterance might be the same in both trials. For the one-parameter model, we further assumed that all correlations are given by a parameter α which was tuned on the development set. The

Table 6: Results for Scr-4par and Scr-UC with and without weight-adjustment. The weight-adjustment parameter, α was tuned to optimize $\hat{C}_{\text{llr}}$ on the development set. 'Reg. to **0**' means L2 regularization towards **0**.

Set	Method	actDCF08	minDCF08	actDCF10	minDCF10	EER	$\hat{C}_{\text{llr}}$	$\hat{C}_{\text{llr}}^{\text{min}}$	Weight-adj
SRE08	Scr-4par	0.0274	0.0254	0.00257	0.000802	0.0471	0.202	0.177	no, $\alpha = 0$
		0.0276	0.0254	0.00256	0.000804	0.0476	0.204	0.178	Sample corr.
	Scr-UC. Reg. to **0**	0.0334	0.0304	0.000876	0.000743	0.0564	0.235	0.212	no
		0.0317	0.0302	0.000851	0.000739	0.0570	0.231	0.213	$\alpha = 0.2$
		0.0314	0.0300	0.000851	0.000740	0.0572	0.231	0.213	Sample corr.
SRE10	Scr-4par	0.0117	0.0100	0.000574	0.000375	0.0188	0.086	0.0744	no, $\alpha = 0$
		0.0116	0.0099	0.000569	0.000375	0.0188	0.085	0.0742	Sample corr.
	Scr-UC. Reg. to **0**	0.0304	0.0183	0.000916	0.000598	0.0370	0.180	0.137	no
		0.0278	0.0182	0.000888	0.000612	0.0369	0.173	0.137	$\alpha = 0.2$
		0.0275	0.0183	0.000888	0.000616	0.0368	0.173	0.138	Sample corr.

Table 7: Result for three DT schemes, with and without weight-adjustment, using half of the training speakers. The weight-adjustment parameter, α was tuned to optimize $\hat{C}_{\text{llr}}$ on the development set. 'Reg. to **0**' means L2 regularization towards **0**.

Set	Method	actDCF08	minDCF08	actDCF10	minDCF10	EER	$\hat{C}_{\text{llr}}$	$\hat{C}_{\text{llr}}^{\text{min}}$	Weight-adj
SRE08	AT-Cal	0.0281	0.0263	0.00092	0.000793	0.0523	0.201	0.187	no
		0.0286	0.0263	0.00115	0.000793	0.0523	0.202	0.187	$\alpha = 0.2$
	Scr-4par	0.0322	0.0274	0.00177	0.000921	0.0515	0.220	0.190	no
		0.0319	0.0273	0.00217	0.000910	0.0519	0.219	0.189	$\alpha = 0.3$
	Scr-UC. Reg. to **0**	0.0591	0.0340	0.001000	0.000790	0.0724	0.349	0.248	no
		0.0533	0.0329	0.000997	0.000819	0.0722	0.328	0.247	$\alpha = 0.4$
SRE10	AT-Cal	0.0127	0.0111	0.000743	0.000396	0.0229	0.101	0.090	no
		0.0125	0.0111	0.000705	0.000396	0.0229	0.099	0.090	$\alpha = 0.2$
	Scr-4par	0.0116	0.0106	0.000657	0.000394	0.0217	0.0950	0.0848	no
		0.0114	0.0106	0.000607	0.000392	0.0219	0.0934	0.0850	$\alpha = 0.3$
	Scr-UC. Reg. to **0**	0.0646	0.0239	0.001000	0.000695	0.0479	0.327	0.176	no
		0.0593	0.0238	0.001000	0.000737	0.0479	0.302	0.176	$\alpha = 0.4$

Table 8: Estimated correlations for AT-Cal, Scr-4par and Scr-UC. 'c_0' and 'c_{-0}' are the estimated correlations for trials that have nothing in common, and accordingly should be 0.

Set	Corr.	AT-Cal	Scr-4par.	Scr-UC
Target	c_a	0.378	0.377	0.364
	c_b	0.040	0.036	0.108
	c_0	4.79×10^{-4}	4.83×10^{-4}	2.15×10^{-4}
Non-target	c_{-a}	0.768	0.856	0.770
	c_{-b}	0.534	0.623	0.507
	c_{-c}	0.010	0.003	0.006
	c_{-d}	7.75×10^{-4}	2.92×10^{-4}	9.31×10^{-4}
	c_{-0}	-3.15×10^{-4}	-1.97×10^{-4}	3.20×10^{-4}

estimated sample correlations are shown in Table 8. We can see that there is a clear correlation between trials involving the same utterance or speaker. Moreover, the results for the three models are quite similar, which suggests the dependence on $\boldsymbol{\theta}$ may not be large. However, our assumptions about how the correlations depends on α are not that accurate. In particular, it is noticeable that using the same two speakers in both trials causes much more correlation than using only one same utterance, i.e., that $c_{-b} \gg c_{-c}$.

As already revealed, for Scr-4par, the definiteness constraints on $\boldsymbol{P}$ and $\boldsymbol{Q}$ were almost always preserved by itself. $\boldsymbol{P}$ was always kept positive definite and $\boldsymbol{Q}$ was always kept negative definite. Out of the 8 training sizes, it happened once that the matrix $\boldsymbol{P}+\boldsymbol{Q}$ was not positive definite, but in this case, only one of its eigenvalues were negative. We did not see any difference in performance between SCR-Def and Scr-UC but an inspection of the eigenvalues reveals that the definiteness constraints were never fulfilled for Scr-UC, regardless of whether regularization was applied towards the model obtained by GT, or towards **0**. However, we also investigated whether $\boldsymbol{\omega}^T\boldsymbol{P}\boldsymbol{\omega} > 0$, $\boldsymbol{\omega}^T\boldsymbol{Q}\boldsymbol{\omega} < 0$ and $\boldsymbol{\omega}^T(\boldsymbol{P}+\boldsymbol{Q})\boldsymbol{\omega} > 0$ for each i-vector, $\boldsymbol{\omega}$, in the PLDA training set and in the development set, SRE06. When regularization towards the GT model was applied, the number of *violations* was very few. This means that the constraints are, in some sense, practically fulfilled for i-vectors that are normally observed. This may be because the DT model remains close to the GT model and therefore keeps its properties. Interestingly though, when regularization towards **0** was applied, there were many violations against the second constraint, $\boldsymbol{\omega}^T\boldsymbol{Q}\boldsymbol{\omega} < 0$, but no violations against the other two constraints. Recall from Section 3.2 that the second constraint is related to a length property of the model, which is unlikely to be useful when the i-vectors are length-normalized. While this supports our analysis in Sec-

tion 3.2, it also shows that, at least for the training sizes used in our experiment, the training procedure tends to learn the useful properties from the data.

7. Conclusions and future work

We dealt with two issues in order to improve the robustness of discriminative training (DT) against data insufficiency in PLDA based speaker verification. First, we examined how to appropriately use statistically dependent training trials by adjusting the weights of the trials in the training objective. Second, we proposed three new DT schemes and systematically compared them with two existing training schemes, namely generative training (GT) followed by score calibration by means of a discriminatively trained affine transformation, and, DT of all the parameters of the PLDA score function. We evaluated the methods on the male telephone part of the NIST SRE 08 and the NIST SRE10. We confirmed the effectiveness of weight-adjustment when the number of training speakers were few or when DT was only weakly constrained. On SRE08, GT followed by score calibration performed better than any of the proposed DT schemes. However, on SRE10 one of our DT schemes was better. In combination with weight-adjustment it gave improvements in between 7% and 19% in $\hat{C}_{\text{llr}}$ depending on the training data size, compared to GT followed by score calibration, which in turn was better than DT of all the parameters of the PLDA score function.

Future directions are many. There are other phenomena that may cause the training trials to be statistically dependent than common utterances or speakers. For example, when the same microphone is used in more than one training utterance. It would be interesting to apply the weight-adjustment to deal with such dependencies. Although using the best linear unbiased estimator for the expected loss is well motivated and works well, is possible that the results could be improved by some other estimator than the BLUE. For example, a non-linear estimator or an estimator that considers higher moments than the variance. Another issue for future consideration is that there might be a mismatch between the properties of the training trials and the properties of the test trials. Several studies in domain adaptation have shown that the Switchboard and the NIST SRE corpora have different properties (Garcia-Romero and McCree, 2014; Biswas et al., 2015). Furthermore, the target trials in the test sets of the NIST SRE are always from different telephone numbers whereas the majority of the target trials used for DT are from the same telephone number.

8. Acknowledgment

We thank Lukáš Burget as well as one anonymous reviewer of our Odyssey paper (Rohdin et al., 2014b) who suggested us to compensate for the statistical dependence of the training trials.

AppendixA. Constraints on the PLDA LLR score function

In this section, we derive the constraints on $\boldsymbol{P}$ and $\boldsymbol{Q}$ mentioned in Subsection 3.2. In this paper, we use the term *definite* only for symmetric matrices. We use the term *semidefinite* when at least one eigenvalue of the matrix is zero, i.e., it does not have full rank, and the term *nonnegative-definite* for matrices which are either positive-definite or positive-semidefinite.

AppendixA.1. Rank of $\boldsymbol{P}$ and $\boldsymbol{Q}$

In this subsection, we show that both the rank of $\boldsymbol{P}$ and the rank of $\boldsymbol{Q}$ is equal to the rank of $\boldsymbol{V}$.

Let $\boldsymbol{S} = \Sigma_{\text{tot}} - \Sigma_{\text{ac}}\Sigma_{\text{tot}}^{-1}\Sigma_{\text{ac}}$. Then,

$$\text{rank}(\boldsymbol{P}) = \text{rank}(\Sigma_{\text{tot}}^{-1}\Sigma_{\text{ac}}\boldsymbol{S}^{-1}) \leq \text{rank}(\Sigma_{\text{ac}}), \tag{A.1}$$

$$\text{rank}(\Sigma_{\text{ac}}) = \text{rank}(\Sigma_{\text{tot}}\boldsymbol{P}\boldsymbol{S}) \leq \text{rank}(\boldsymbol{P}). \tag{A.2}$$

Hence, $\text{rank}(\boldsymbol{P}) = \text{rank}(\Sigma_{\text{ac}}) = \text{rank}(\boldsymbol{V})$.

Using that $\boldsymbol{S}$ is positive definite (Boyd and Vandenberghe, 2004, Ch. A.5.5) and the Woodbury identity (Petersen and Pedersen, 2012, Eq. (156)) we obtain

$$\begin{aligned}\boldsymbol{Q} &= -\Sigma_{\text{tot}}^{-1}\Sigma_{\text{ac}}\boldsymbol{S}^{-1}\Sigma_{\text{ac}}\Sigma_{\text{tot}}^{-1} \\ &= -\Sigma_{\text{tot}}^{-1}\Sigma_{\text{ac}}\boldsymbol{S}^{-1/2}(\Sigma_{\text{tot}}^{-1}\Sigma_{\text{ac}}\boldsymbol{S}^{-1/2})^T,\end{aligned} \tag{A.3}$$

where $\boldsymbol{S}^{-1/2}$ is the square root of $\boldsymbol{S}^{-1}$. Set $\boldsymbol{M} = \Sigma_{\text{tot}}^{-1}\Sigma_{\text{ac}}\boldsymbol{S}^{-1/2}$. Then,

$$\text{rank}(\boldsymbol{Q}) = \text{rank}(\boldsymbol{M}) \leq \text{rank}(\Sigma_{\text{ac}}), \tag{A.4}$$

$$\text{rank}(\Sigma_{\text{ac}}) = \text{rank}(\Sigma_{\text{tot}}\boldsymbol{M}\boldsymbol{S}^{1/2}) \leq \text{rank}(\boldsymbol{M}). \tag{A.5}$$

Hence, $\text{rank}(\boldsymbol{Q}) = \text{rank}(\Sigma_{\text{ac}}) = \text{rank}(\boldsymbol{V})$.

AppendixA.2. Definiteness of $\boldsymbol{P}$ and $\boldsymbol{Q}$

In this subsection, we derive the following constraints on $\boldsymbol{P}$ and $\boldsymbol{Q}$:

1. $\boldsymbol{Q}$ is negative-(semi)definite.
2. $\boldsymbol{P}$ is positive-(semi)definite.
3. $\boldsymbol{P} + \boldsymbol{Q}$ is positive-(semi)definite.

For these constraints, *semi* applies when $\text{rank}(V) < d$. Constraint 1 follows directly from Eq. (A.3) and the fact that $\text{rank}(\boldsymbol{Q}) = \text{rank}(\boldsymbol{V})$. If $\boldsymbol{SPS}$ is positive-(semi)definite, then $\boldsymbol{S}^{-1}\boldsymbol{SPSS}^{-1} = \boldsymbol{P}$ is positive (semi)definite (Harville, 1997, Thm. 14.2.9). A bit of processing of $\boldsymbol{SPS}$ gives

$$\boldsymbol{SPS} = \Sigma_{\text{ac}} - \Sigma_{\text{ac}}\Sigma_{\text{tot}}^{-1}\Sigma_{\text{ac}} + \Sigma_{\text{ac}}\Sigma_{\text{tot}}^{-1}\Sigma_{\text{wc}}\Sigma_{\text{tot}}^{-1}\Sigma_{\text{ac}}, \tag{A.6}$$

where $\Sigma_{\text{wc}} = \Sigma_{\text{tot}} - \Sigma_{\text{ac}}$. From Eq. (A.6) it is clear that $\boldsymbol{P}$ is symmetric. The last term in Eq. (A.6) is nonnegative-definite. The term $\boldsymbol{Z} = \Sigma_{\text{ac}} - \Sigma_{\text{ac}}\Sigma_{\text{tot}}^{-1}\Sigma_{\text{ac}}$ is a Schur complement of Σ_{tot} in

$$\boldsymbol{M} = \begin{bmatrix}\Sigma_{\text{tot}} & \Sigma_{\text{ac}} \\ \Sigma_{\text{ac}} & \Sigma_{\text{ac}}\end{bmatrix}. \tag{A.7}$$

$\boldsymbol{Z}$ is positive-(semi)definite if $\boldsymbol{M}$ is positive-(semi)definite (Boyd and Vandenberghe, 2004, Ch. A.5.5). By expanding $[\boldsymbol{x}_1^T \ \boldsymbol{x}_2^T]\boldsymbol{M}[\boldsymbol{x}_1^T \ \boldsymbol{x}_2^T]^T$ for two real vectors, $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, it can be verified that $\boldsymbol{M}$ is positive definite if $\text{rank}(\Sigma_{\text{ac}}) = d$, otherwise positive-semidefinite.

Since, $\boldsymbol{SPS}$, is a sum of nonnegative-definite matrices, it is nonnegative definite. Since rank($\boldsymbol{P}$) = rank($\boldsymbol{V}$), Constraint 2 follows.

$\boldsymbol{P} + \boldsymbol{Q}$ can be rewritten as

$$\begin{aligned}\boldsymbol{P} + \boldsymbol{Q} &= \Sigma_{\text{tot}}^{-1}\Sigma_{\text{ac}}\boldsymbol{S}^{-1}(I - \Sigma_{\text{ac}}\Sigma_{\text{tot}}^{-1}) \\ &= \boldsymbol{S}^{-1}\Sigma_{\text{ac}}\Sigma_{\text{tot}}^{-1}(I - \Sigma_{\text{ac}}\Sigma_{\text{tot}}^{-1}) \qquad \text{(A.8)} \\ &= \boldsymbol{S}^{-1}(\Sigma_{\text{ac}} - \Sigma_{\text{ac}}\Sigma_{\text{tot}}^{-1}\Sigma_{\text{ac}})\Sigma_{\text{tot}}^{-1}, \qquad \text{(A.9)}\end{aligned}$$

where (A.8) comes from the symmetry of $\boldsymbol{P} = \Sigma_{\text{tot}}^{-1}\Sigma_{\text{ac}}\boldsymbol{S}^{-1}$. If $\boldsymbol{S}^{1/2}(\boldsymbol{P} + \boldsymbol{Q})\boldsymbol{S}^{1/2}$ is positive (semi)definite, then $\boldsymbol{S}^{-1/2}\boldsymbol{S}^{1/2}(\boldsymbol{P} + \boldsymbol{Q})\boldsymbol{S}^{1/2}\boldsymbol{S}^{-1/2} = \boldsymbol{P} + \boldsymbol{Q}$ is positive (semi)definite. Since this matrix is symmetric, it is enough to show that its eigenvalues are larger than, or equal to zero. For two matrices, $\boldsymbol{A}$ and $\boldsymbol{B}$, the eigenvalues of $\boldsymbol{AB}$ and $\boldsymbol{BA}$ are the same (Harville, 1997, Thm. 21.10.1). Therefore, the eigenvalues of $\boldsymbol{S}^{1/2}(\boldsymbol{P} + \boldsymbol{Q})\boldsymbol{S}^{1/2}$ are the same as for

$$\boldsymbol{S}(\boldsymbol{P} + \boldsymbol{Q}) = (\Sigma_{\text{ac}} - \Sigma_{\text{ac}}\Sigma_{\text{tot}}^{-1}\Sigma_{\text{ac}})\Sigma_{\text{tot}}^{-1/2}\Sigma_{\text{tot}}^{-1/2}, \qquad \text{(A.10)}$$

whose eigenvalues in turn are the same as for

$$\Sigma_{\text{tot}}^{-1/2}(\Sigma_{\text{ac}} - \Sigma_{\text{ac}}\Sigma_{\text{tot}}^{-1}\Sigma_{\text{ac}})\Sigma_{\text{tot}}^{-1/2}. \qquad \text{(A.11)}$$

The middle part is, as pointed out earlier, positive-definite if rank($\boldsymbol{V}$) = d, otherwise positive-semidefinite. Accordingly, Constraint 3 follows.

AppendixB. Derivation of formulas for weight-adjustment

In this section, we derive the formulas for the trial weights given in Eqs. (33) and (35).

AppendixB.1. Optimal weights for the target trials

From Eq. (26) we have

$$\Sigma_1\boldsymbol{\beta} = \mathbf{1}/(\mathbf{1}^T\Sigma_1^{-1}(\boldsymbol{\theta})\mathbf{1}). \qquad \text{(B.1)}$$

Consider the loss of one specific target trial of speaker A, $l(A_1, A_2)$, and its covariance with the loss of all other target trials. Let $\Sigma_1^{(A_1,A_2)}$ be the row in Σ_1, containing these covariances. Let the variance of the target trial losses be denoted $v_1(\boldsymbol{\theta})$. The covariances with the losses of the target trials from the other speakers are 0. The covariances with the losses of the other target trials of speaker A are either $v_1(\boldsymbol{\theta})c_a$ or $v_1(\boldsymbol{\theta})c_b$. Let the number of such trials be denoted n_a and n_b respectively. Assume that the weights for all target trials of speaker A are the same, β_A, then

$$\Sigma_1^{(A_1,A_2)}\boldsymbol{\beta} = (1 + n_a c_a + n_b c_b)v_1(\boldsymbol{\theta})\beta_A. \qquad \text{(B.2)}$$

Notice that the elements in $\boldsymbol{\beta}$ which are weights for another speaker than speaker A are always multiplied with elements in $\Sigma_1^{(A_1,A_2)}$ that are 0, and therefore they are not present on the right hand side of Eq. (B.2). Setting Eq. (B.2) equal to the corresponding row in Eq. (B.1) gives

$$\begin{aligned}(1 + n_a c_a + n_b c_b)\beta_A &= 1/\left(v_1(\boldsymbol{\theta})\mathbf{1}^T\Sigma_1^{-1}(\boldsymbol{\theta})\mathbf{1}\right) \\ &= 1/(\mathbf{1}^T\boldsymbol{R}_1^{-1}\mathbf{1}) \\ &= k_1, \qquad \text{(B.3)}\end{aligned}$$

where $\boldsymbol{R}_1 = \Sigma_1(\boldsymbol{\theta})/v_1(\boldsymbol{\theta})$ is the correlation matrix which, according to our assumptions, does not depend on $\boldsymbol{\theta}$. Since we are using all possible target trials, the rows in Σ_1 corresponding to the other target trials of speaker A contains the same elements as $\Sigma_1^{(A_1,A_2)}$ but with a different order. These rows therefore also results in Eq. (B.3). Thus, an equal weight for all target trials of the same speaker gives a solution to Eq. (26) and since Σ_t is invertible, it is the only solution.

It remains to find n_a and n_b. There are $N_A(N_A - 1)/2 - 1$ *unique* target trials of speaker A, excluding the trial (A_1, A_2). n_a is the number of trials that include either A_1 or A_2 but not both, in total $n_a = 2(N_A - 2)$ trials. n_b is the remaining target trials of speaker A, except (A_1, A_2), in total

$$\begin{aligned}n_b &= N_A(N_A - 1)/2 - 2(N_A - 2) - 1 \\ &= (N_A - 2)(N_A - 3)/2. \qquad \text{(B.4)}\end{aligned}$$

AppendixB.2. Approximately optimal weights for the non-target trials

Consider the loss of one specific trial, $l(A_1, B_1)$ and its covariance with the loss of other non-target trials. We use a similar approach as for the target trials. However, the non-target trials where one speaker is different from A and B will complicate matters. The number of non-target trials involving the same speakers, A and B, where one utterance is either, A_1 or B_1, is $n_{-a} = N_A + N_B - 2$. The number of non-target trials involving the same speakers but not the utterance A_1 or B_1, are $n_{-b} = (N_A - 1)(N_B - 1)$. Now, assume that each non-target trial involving speaker A and B has same weight, β_{AB}, and similarly for the other *speaker pairs*. (It can be verified that this gives a solution.) Then from Eq. (26), we have

$$\begin{aligned}1/(\mathbf{1}^T\boldsymbol{R}_{(-1)}^{-1}\mathbf{1}) &= (1 + n_{-a}c_{-a} + n_{-b}c_{-b})\beta_{AB} \\ &+ c_{-c}\sum_{X\neq A,B}(\beta_{AX} + \beta_{BX})N_X \\ &+ c_{-d}\sum_{X\neq A,B}\left((N_A - 1)\beta_{AX} + (N_B - 1)\beta_{BX}\right)N_X, \qquad \text{(B.5)}\end{aligned}$$

where $\boldsymbol{R}_{-1} = \Sigma_{-1}/v_{-1}(\boldsymbol{\theta})$ is the correlation matrix and $v_{-1}(\boldsymbol{\theta})$ is the variance for the non-target trial losses. The number of unknown variables in this equation is equal to the number of speaker pairs and we have one such equation per speaker pair. The number of speaker pairs is, however, very large. Instead of solving this system of equations, we use the approximations:

$$\sum_{X\neq A,B}(\beta_{AX} + \beta_{BX})N_X \approx 2\sum_{X\neq A,B}\beta_{AB}N_X, \qquad \text{(B.6)}$$

and

$$\begin{aligned}&\sum_{X\neq A,B}\left((N_A - 1)\beta_{AX} + (N_B - 1)\beta_{BX}\right)N_X \\ &\qquad \approx \sum_{X\neq A,B}\left((N_A + N_B - 2)\beta_{AB}\right)N_X. \qquad \text{(B.7)}\end{aligned}$$

These approximations are quite reasonable since, e.g., a larger N_A results in a smaller values of both β_{AB} and β_{AX}. This results in

$$(1 + n_{-a}c_{-a} + n_{-b}c_{-b} + n_{-c}c_{-c} + n_{-d}c_{-d})\beta_{AB} \approx k_{-1}, \qquad \text{(B.8)}$$

where,

$$k_{-1} = 1/(\mathbf{1}^T \boldsymbol{R}_{-1}^{-1}\mathbf{1}), \tag{B.9}$$

$$n_{-c} = 2\sum_{X\neq B,A} N_X, \tag{B.10}$$

and

$$n_{-d} = (N_A + N_B - 2)\sum_{X\neq B,A} N_X. \tag{B.11}$$

AppendixC. Initialization and calculation of gradients

AppendixC.1 states results given in previous studies. The gradients and initializations for Scr-4par, iV-elmnt and Scr-Def are then given in AppendixC.2, AppendixC.3 and AppendixC.4, respectively. In this section, $\mathbf{1}_{q\times r}$ denotes a matrix of dimension $q \times r$ whose all elements are equal to 1.

AppendixC.1. Results from previous studies

The results in this subsection are given in Cumani et al. (2011). Let the n training i-vectors be collected in a matrix, $\boldsymbol{\Psi} = [\boldsymbol{\omega}_1 \dots \boldsymbol{\omega}_n]$, and all the scores of the training data be collected in a matrix $\boldsymbol{S}$, i.e., $S_{ij} = s_{ij}$, where s_{ij} is given by Eq. (10). Then $\boldsymbol{S} = \boldsymbol{S}_\mathrm{P} + \boldsymbol{S}_\mathrm{Q} + \boldsymbol{S}_\mathrm{c} + \boldsymbol{S}_\mathrm{k}$, where

$$\begin{aligned}
\boldsymbol{S}_\mathrm{P} &= 2\boldsymbol{\Psi}^T\boldsymbol{P}\boldsymbol{\Psi},\\
\boldsymbol{S}_\mathrm{Q} &= \mathrm{diag}(\boldsymbol{\Psi}^T\boldsymbol{Q}\boldsymbol{\Psi})\mathbf{1}_{1\times n} + (\mathrm{diag}(\boldsymbol{\Psi}^T\boldsymbol{Q}\boldsymbol{\Psi})\mathbf{1}_{1\times n})^T,\\
\boldsymbol{S}_\mathrm{c} &= \boldsymbol{\Psi}^T\boldsymbol{c}\mathbf{1}_{1\times n} + (\boldsymbol{\Psi}^T\boldsymbol{c}\mathbf{1}_{1\times n})^T,\\
\boldsymbol{S}_\mathrm{k} &= k\mathbf{1}_{n\times n}.
\end{aligned} \tag{C.1}$$

The gradient of $\hat{L}(\boldsymbol{\gamma})$ in Eq. (18) is given by,

$$\nabla\hat{L}(\boldsymbol{\gamma}) = \begin{bmatrix} \nabla_P\hat{L}(\boldsymbol{\gamma})\\ \nabla_Q\hat{L}(\boldsymbol{\gamma})\\ \nabla_c\hat{L}(\boldsymbol{\gamma})\\ \nabla_k\hat{L}(\boldsymbol{\gamma})\end{bmatrix} = \begin{bmatrix} \mathrm{vec}(\boldsymbol{P}')\\ \mathrm{vec}(\boldsymbol{Q}')\\ \boldsymbol{c}'\\ k'\end{bmatrix}, \tag{C.2}$$

where

$$\boldsymbol{P}' = 2\boldsymbol{\Psi}\boldsymbol{G}\boldsymbol{\Psi}^T, \tag{C.3}$$

$$\boldsymbol{Q}' = 2\mathrm{vec}([\boldsymbol{\Psi}\circ(\mathbf{1}_{d\times n}\boldsymbol{G})]\boldsymbol{\Psi}^T), \tag{C.4}$$

$$\boldsymbol{c}' = 2[\boldsymbol{\Psi}\circ(\mathbf{1}_{d\times n}\boldsymbol{G})\boldsymbol{\Psi}]\mathbf{1}_{n\times 1}, \tag{C.5}$$

$$k' = \mathbf{1}_{n\times 1}^T\boldsymbol{G}\mathbf{1}_{n\times 1}, \tag{C.6}$$

$$G_{ij} = \frac{\partial l_{ij}}{\partial s_{ij}}, \tag{C.7}$$

and

$$l_{ij} = l\left(t_{ij}, s_{ij}(\boldsymbol{\gamma}), \tau\right). \tag{C.8}$$

AppendixC.2. Scr-4Par

The derivative of $\hat{L}$ with respect to a_P, is

$$\begin{aligned}
\frac{\partial\hat{L}}{\partial a_\mathrm{P}} &= \sum_{ij}\frac{\partial l_{ij}}{\partial s_{ij}}\frac{\partial s_{ij}}{\partial a_\mathrm{P}} = \sum_{ij} G_{ij}S_{\mathrm{P}ij}\\
&= \mathbf{1}_{n\times 1}^T(\boldsymbol{G}\circ\boldsymbol{S}_\mathrm{P})\mathbf{1}_{n\times 1}.
\end{aligned} \tag{C.9}$$

The derivatives $\frac{\partial\hat{L}}{\partial a_\mathrm{Q}}$, $\frac{\partial\hat{L}}{\partial a_\mathrm{c}}$ and $\frac{\partial\hat{L}}{\partial a_\mathrm{k}}$ are calculated in the same way. Each of a_P, a_Q, a_c and a_k, are initialized to 1.

AppendixC.3. iV-elmnt

We collect the scalings of the i-vector elements in a diagonal matrix, $\boldsymbol{D}$, so that $\boldsymbol{\omega}$ is replaced by $\boldsymbol{D}\boldsymbol{\omega}$ in Eq. (10), i.e.,

$$\begin{aligned}
s_{ij} = {}& \boldsymbol{\omega}_i^T\boldsymbol{D}\boldsymbol{P}\boldsymbol{D}\boldsymbol{\omega}_j + \boldsymbol{\omega}_j^T\boldsymbol{D}\boldsymbol{P}\boldsymbol{D}\boldsymbol{\omega}_i\\
&+\boldsymbol{\omega}_i^T\boldsymbol{D}\boldsymbol{Q}\boldsymbol{D}\boldsymbol{\omega}_i + \boldsymbol{\omega}_j^T\boldsymbol{D}\boldsymbol{Q}\boldsymbol{D}\boldsymbol{\omega}_j\\
&+(\boldsymbol{\omega}_i + \boldsymbol{\omega}_j)^T\boldsymbol{D}\boldsymbol{c} + k.
\end{aligned} \tag{C.10}$$

Let the contribution to the gradient from terms including $\boldsymbol{P}$ be denoted $\nabla^{(P)}_{\mathrm{diag}(\boldsymbol{D})}\hat{L}$ and similarly for $\boldsymbol{Q}$ and $\boldsymbol{c}$. We will use matrix calculus (Minka, 2001) with the convention that the elements of the matrix derivative are laid out according to the transpose of the denominator. The contribution from $\boldsymbol{P}$ to the differential is

$$\mathrm{d}\hat{L} = \mathrm{tr}(\boldsymbol{P}'\mathrm{d}\boldsymbol{P}^T). \tag{C.11}$$

Replacing $\boldsymbol{P}$ with $\boldsymbol{D}\boldsymbol{P}\boldsymbol{D}$ we then get

$$\begin{aligned}
\mathrm{d}\hat{L} &= \mathrm{tr}\left(\boldsymbol{P}'\mathrm{d}(\boldsymbol{D}\boldsymbol{P}\boldsymbol{D})^T\right)\\
&= \mathrm{tr}\left(\boldsymbol{P}\boldsymbol{D}\boldsymbol{P}'\mathrm{d}\boldsymbol{D} + \boldsymbol{P}'\boldsymbol{D}\boldsymbol{P}\mathrm{d}\boldsymbol{D} + \boldsymbol{D}\boldsymbol{P}'\boldsymbol{D}\mathrm{d}\boldsymbol{P}\right),
\end{aligned} \tag{C.12}$$

i.e.,

$$\nabla^{(P)}_{\mathrm{diag}(\boldsymbol{D})}\hat{L} = \mathrm{diag}(\boldsymbol{P}\boldsymbol{D}\boldsymbol{P}' + \boldsymbol{P}'\boldsymbol{D}\boldsymbol{P}). \tag{C.13}$$

The contribution from the terms with $\boldsymbol{Q}$ is calculated in the same way. For $\boldsymbol{c}$, we get

$$\begin{aligned}
\mathrm{d}\hat{L} &= \boldsymbol{c}'\mathrm{d}(\boldsymbol{D}\boldsymbol{c})^T\\
&= \boldsymbol{c}'((\mathrm{d}\boldsymbol{c}^T)\boldsymbol{D}^T + \boldsymbol{c}^T\mathrm{d}\boldsymbol{D}^T),
\end{aligned} \tag{C.14}$$

i.e.,

$$\nabla^{(c)}_{\mathrm{diag}(\boldsymbol{D})}\hat{L} = \mathrm{diag}(\boldsymbol{c}'\boldsymbol{c}^T) = \boldsymbol{c}'\circ\boldsymbol{c}. \tag{C.15}$$

Finally,

$$\nabla_{\mathrm{diag}(\boldsymbol{D})}\hat{L} = \nabla^{(P)}_{\mathrm{diag}(\boldsymbol{D})}\hat{L} + \nabla^{(Q)}_{\mathrm{diag}(\boldsymbol{D})}\hat{L} + \nabla^{(c)}_{\mathrm{diag}(\boldsymbol{D})}\hat{L}. \tag{C.16}$$

The derivative for k is calculated as in AppendixC.2. The scalings of the i-vector elements and k are initialized to 1.

AppendixC.4. Scr-Def

The gradients for $\boldsymbol{c}$ and k in Eq. (C.2) are used without modification. The contribution from $\boldsymbol{P}$ and $\boldsymbol{Q}$ to the differential is

$$\begin{aligned}
\mathrm{d}\hat{L} &= \mathrm{tr}\left(\boldsymbol{P}'\mathrm{d}\boldsymbol{P}^T + \boldsymbol{Q}'\mathrm{d}\boldsymbol{Q}^T\right)\\
&= \mathrm{tr}\left(\boldsymbol{P}'\mathrm{d}\boldsymbol{R}_A\boldsymbol{R}_A^T + \boldsymbol{P}'\mathrm{d}\boldsymbol{Q}_A\boldsymbol{Q}_A^T - \boldsymbol{Q}'\mathrm{d}\boldsymbol{Q}_A\boldsymbol{Q}_A^T\right).
\end{aligned} \tag{C.17}$$

Using the fact that $\boldsymbol{P}'$ is symmetric, we get for the first term

$$\begin{aligned}
\mathrm{tr}\left(\boldsymbol{P}'(\mathrm{d}\boldsymbol{R}_A\boldsymbol{R}_A^T)\right) &= \mathrm{tr}\left(\boldsymbol{P}'\left(\mathrm{d}\boldsymbol{R}_A\right)\boldsymbol{R}_A^T\right) + \mathrm{tr}\left(\boldsymbol{P}'\boldsymbol{R}_A\mathrm{d}\boldsymbol{R}_A^T\right)\\
&= \mathrm{tr}\left(\left(\left(\mathrm{d}\boldsymbol{R}_A\right)\boldsymbol{R}_A^T\right)^T\boldsymbol{P}'^T\right) + \mathrm{tr}\left(\boldsymbol{P}'\boldsymbol{R}_A\mathrm{d}\boldsymbol{R}_A^T\right)\\
&= \mathrm{tr}\left(2\boldsymbol{P}'\boldsymbol{R}_A\mathrm{d}\boldsymbol{R}_A^T\right).
\end{aligned} \tag{C.18}$$

The other terms are treated analogously, resulting in

$$\begin{bmatrix}\nabla_{\boldsymbol{R}_A}\hat{L}\\ \nabla_{\boldsymbol{Q}_A}\hat{L}\end{bmatrix} = \begin{bmatrix}2\mathrm{vec}(\boldsymbol{P}'\boldsymbol{R}_A)\\ 2\mathrm{vec}\left((\boldsymbol{P}' - \boldsymbol{Q}')\boldsymbol{Q}_A\right)\end{bmatrix}. \tag{C.19}$$

The regularization term is dealt with by adding $2(\boldsymbol{P} - \tilde{\boldsymbol{P}})$ to $\boldsymbol{P}'$ and $2(\boldsymbol{Q} - \tilde{\boldsymbol{Q}})$ to $\boldsymbol{Q}'$. For initialization, we use a model estimated by GT and calculate $\boldsymbol{R}_A$ and $\boldsymbol{Q}_A$ by means of eigendecomposition of $\boldsymbol{R}$ and $\boldsymbol{Q}$ respectively, e.g.,

$$\boldsymbol{Q}_A = \boldsymbol{E}(-\boldsymbol{Q})\boldsymbol{D}(-\boldsymbol{Q})^{\frac{1}{2}}, \tag{C.20}$$

where the columns of $\boldsymbol{E}(-\boldsymbol{Q})$ are the eigenvectors of $-\boldsymbol{Q}$ and $\boldsymbol{D}(-\boldsymbol{Q})$ is a diagonal matrix containing the corresponding eigenvalues.

References

Bishop, C.M., 2006. Pattern Recognition and Machine Learning. Springer.

Biswas, S., Rohdin, J., Shinoda, K., 2015. Autonomous selection of i-vectors for {PLDA} modelling in speaker verification. Speech Communication 72, 32 – 46.

Borgström, B., McCree, A., 2013. Discriminatively trained bayesian speaker comparison of i-vectors, in: ICASSP, pp. 7659–7662.

Bousquet, P.M., Bonastre, J.F., Matrouf, D., 2014. Exploring some limits of gaussian plda modeling for i-vector distributions, in: Odyssey, pp. 41–47.

Boyd, S., Vandenberghe, L., 2004. Convex Optimization. Cambridge University Press, New York, NY, USA.

Brümmer, N., 2010. EM for probabilistic LDA. URL: `https://sites.google.com/site/nikobrummer`.

Brümmer, N., 2010. Measuring, refining and calibrating speaker and language information extracted from speech. Ph.D. thesis. Stellenbosch: University of Stellenbosch.

Brümmer, N., Burget, L., Černocký, J., Glembek, O., Grézl, F., Karafiát, M., van Leeuwen, D.A., Matějka, P., Schwarz, P., Strasheim, A., 2007. Fusion of heterogeneous speaker recognition systems in the STBU submission for the NIST speaker recognition evaluation 2006. IEEE Transactions on Audio, Speech & Language Processing 15, 2072–2084.

Brümmer, N., Doddington, G., 2013. Likelihood-ratio calibration using prior-weighted proper scoring rules, in: INTERSPEECH, pp. 1976–1980.

Brümmer, N., du Preez, J., 2006. Application-independent evaluation of speaker detection. Computer Speech & Language 20, 230–275.

Brümmer, N., de Villiers, E., 2010. The speaker partitioning problem, in: Odyssey, pp. 194–201.

Brümmer, N., de Villiers, E., 2011. The bosaris toolkit user guide: Theory, algorithms and code for binary classifier score processing. URL: `https://sites.google.com/site/bosaristoolkit/home/bosaristoolkit_userguide.pdf`.

Burget, L., Brümmer, N., Reynolds, D., Kenny, P., Pelecanos, J., Vogt, R., Castaldo, F., Dehak, N., Dehak, R., Glembek, O., Karam, Z.N., Noecker, Jr., J., Na, E., Costin, C.C., Hubeika, V., Kajarekar, S., Scheffer, N., Černocký, J., 2008. Robust Speaker Recognition Over Varying Channels. Technical Report. URL: `http://www.fit.vutbr.cz/research/view_pub.php?id=8893`.

Burget, L., Plchot, O., Cumani, S., Glembek, O., Matějka, P., Brümmer, N., 2011. Discriminatively trained probabilistic linear discriminant analysis for speaker verification, in: ICASSP, pp. 4832–4835.

Cumani, S., Brümmer, N., Burget, L., Laface, P., 2011. Fast discriminative speaker verification in the i-vector space, in: ICASSP, pp. 4852–4855.

Cumani, S., Brümmer, N., Burget, L., Laface, P., Plchot, O., Vasilakakis, V., 2013. Pairwise discriminative speaker verification in the i-vector space. IEEE Transactions on Audio, Speech & Language Processing 21, 1217–1227.

Cumani, S., Glembek, O., Brümmer, N., de Villiers, E., Laface, P., 2012. Gender independent discriminative speaker recognition in i-vector space, in: ICASSP, pp. 4361–4364.

Cumani, S., Laface, P., 2014. Training pairwise support vector machines with large scale datasets, in: ICASSP, pp. 1664–1668.

Dehak, N., Dehak, R., Kenny, P., Brümmer, N., Ouellet, P., Dumouchel, P., 2009. Support vector machines versus fast scoring in the low-dimensional total variability space for speaker verification, in: INTERSPEECH, pp. 1559–1562.

Dehak, N., Kenny, P., Dehak, R., Dumouchel, P., Ouellet, P., 2011. Front-end factor analysis for speaker verification. IEEE Transactions on Audio, Speech, and Language Processing 19, 788–798.

Garcia-Romero, D., Espy-Wilson, C., 2011. Analysis of i-vector length normalization in speaker recognition systems, in: INTERSPEECH, pp. 249–252.

Garcia-Romero, D., McCree, A., 2014. Supervised domain adaptation for i-vector based speaker recognition, in: ICASSP, pp. 4047–4051.

Harville, D., 1997. Matrix Algebra From A Statistician's Perspective. Springer-Verlag.

Hermansky, H., 1990. Perceptual linear predictive (PLP) analysis of speech. J. Acoust. Soc. Am. 57, 1738–1752.

Ioffe, S., 2006. Probabilistic linear discriminant analysis, in: ECCV (4), pp. 531–542.

Kay, S.M., 1993. Fundamentals of Statistical Signal Processing: Estimation Theory. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

Kenny, P., 2005. Joint factor analysis of speaker and session variability: Theory and algorithms, tech. report crim-06/08-13 URL: `http://www.crim.ca/perso/patrick.kenny/`.

Kenny, P., 2010. Bayesian speaker verfication with heavy tailed priors, in: Odyssey.

Kenny, P., Boulianne, G., Ouellet, P., Dumouchel, P., 2007. Joint factor analysis versus eigenchannels in speaker recognition. IEEE Transactions on Audio, Speech & Language Processing 15, 1435–1447.

Liu, D., Nocedal, J., 1989. On the limited memory BFGS method for large scale optimization. Mathematical Programming 45, 503–528.

Mak, M.W., Yu, H.B., 2010. Robust voice activity detection for interview speech in NIST speaker recognition evaluation, in: Proc. APSIPA ASC.

Matejka, P., Glembek, O., Castaldo, F., Alam, M., Plchot, O., Kenny, P., Burget, L., Cernocky, J., 2011. Full-covariance ubm and heavy-tailed PLDA in i-vector speaker verification, in: ICASSP, pp. 4828–4831.

Minka, T.P., 2001. Old and New Matrix Algebra Useful for Statistics. Technical Report. URL: `http://research.microsoft.com/en-us/um/people/minka/papers/matrix/minka-matrix.pdf`.

NIST, 2008. The NIST year 2008 speaker recognition evaluation plan. URL: `http://www.itl.nist.gov/iad/mig/tests/spk/2008/index.html`.

NIST, 2010. The NIST year 2010 speaker recognition evaluation plan. URL: `http://www.itl.nist.gov/iad/mig/tests/spk/2010/index.html`.

Pelecanos, J., Sridharan, S., 2001. Feature warping for robust speaker verification, in: Odyssey, pp. 213–218.

Petersen, K., Pedersen, M., 2012. The matrix cookbook. URL: `http://www.imm.dtu.dk/pubdb/views/edoc_download.php/3274/pdf/imm3274.pdf`. version 20121115.

Prince, S., Elder, J., 2007. Probabilistic linear discriminant analysis for inferences about identity, in: ICCV, pp. 1–8.

Rohdin, J., Biswas, S., Shinoda, K., 2014a. Constrained discriminative PLDA training for speaker verification, in: ICASSP, pp. 1689–1693.

Rohdin, J., Biswas, S., Shinoda, K., 2014b. Discriminative PLDA training application-specific loss functions for speaker verification, in: Odyssey, pp. 26–32.

Schmidt, M., 2012. minFunc.m. URL: `http://www.di.ens.fr/~mschmidt/Software/minFunc.html`.