Odyssey 2014:
The Speaker and Language Recognition Workshop
16-19 June 2014, Joensuu, Finland

Generative pairwise models for speaker recognition

Sandro Cumani, and Pietro Laface

{Sandro.Cumani, Pietro.Laface}@polito.it

Abstract

This paper proposes a simple model for speaker recognition based on i–vector pairs, and analyzes its similarity and differences with respect to the state–of–the–art Probabilistic Linear Discriminant Analysis (PLDA) and Pairwise Support Vector Machine (PSVM) models. Similar to the discriminative PSVM approach, we propose a generative model of i–vector pairs, rather than an usual i–vector based model. The model is based on two Gaussian distributions, one for the "same speakers" and the other for the "different speakers" i–vector pairs, and on the assumption that the i–vector pairs are independent. This independence assumption allows the distributions of the two classes to be independently estimated. The "Two–Gaussian" approach can be extended to the Heavy–Tailed distributions, still allowing a fast closed form solution to be obtained for testing i–vector pairs. We show that this model is closely related to PLDA and to PSVM models, and that tested on the female part of the tel–tel NIST SRE 2010 extended evaluation set, it is able to achieve comparable accuracy with respect to the other models, trained with different objective functions and training procedures.

1. Introduction

The current state–of–the–art in speaker recognition is based on a low–dimensional representation of a speech segment, the so–called i–vector [1, 2], in combination with Probabilistic Linear Discriminant Analysis (PLDA) generative models [3, 4, 5]. An i–vector is a compact representation of a speech segment, obtained from the statistics of a Gaussian Mixture Model (GMM) supervector [6] by a Maximum a Posteriori point estimate of a posterior distribution [2]. A PLDA classifier models the underlying distribution of the speaker and channel components of the i–vectors in a probabilistic framework. From these distributions it is possible to evaluate the likelihood ratio between the "same speaker" hypothesis and "different speaker" hypothesis for a pair of i–vectors. The same paradigm can be used to train discriminative systems where the observation patterns are pairs of i–vectors. In particular discriminative linear classifiers, based on Pairwise Support Vector Machine (PSVM) [7, 8] and on logistic regression [9] have been proposed, which have been shown to achieve state–of–the–art results on recent NIST evaluations [10, 11].

In this paper we propose a simple generative model for speaker recognition based on i–vector pairs. The model is based on two Gaussian distributions, one for the "same speaker" i–vector pairs and the other for the "different speaker" pairs, and on the assumption that the i–vector pairs are independent. We illustrate the structure of the precision matrices of the two distributions, and we detail how their parameters can be effectively estimated. Moreover, since the independence assumption allows the distributions of the two classes to be independently estimated, our "Two–Gaussian" model, referred to in the following as 2–GAU, can be easily extended to Heavy–Tailed distributions leading to the "Two–Heavy–Tailed" (2–HT) model.

We also show that the proposed model is closely related to PLDA and to PSVM models, and that tested on the female part of the tel–tel NIST SRE 2010 extended evaluation set, it is able to achieve comparable accuracy with respect to the other models, trained with different objective functions and training procedures. Although we do not claim that this simple model is more accurate than its state–of–the–art competitors, it has the merit of shedding some light on the pairwise classifiers, revealing a possible unifying framework, despite relevant variations about the model assumptions, the estimation procedures, and the objective functions that each model optimizes.

The paper is organized as follows: Section 2 and 3 briefly recall the PLDA and PSVM models, their parameters, and their objective functions. Section 4 presents the 2–GAU model and illustrates a very fast training procedure for estimating its parameters. Section 5 shows the similarity of the Gaussian PLDA and PSVM models with the 2–GAU model. Section 6 extends the 2–GAU model leading to the 2–HT model, and presents an effective approach for training this more complex model, together with considerations about its training and testing complexity. In Section 7 the similarities and differences of the classifiers are illustrated by using artificial uni–dimensional data. Section 8 is devoted to the illustration of the experimental results, and conclusions are drawn in Section 9.

2. Gaussian PLDA

The generative Gaussian PLDA models [12, 3] are among the best models for comparison of i–vectors. In this section we briefly recall the Gaussian PLDA framework, and also the "Two–covariance model" [4, 5], which provides a useful interpretation of the PSVM approach described in Section 3.

2.1. PLDA

The i–vector generation process is described in the PLDA approach by means of a latent variable probabilistic model where an i–vector $\boldsymbol{\phi}_i$ is represented as the sum of three factors, namely a speaker factor $\mathbf{y}$, an inter–session (channel) factor $\mathbf{x}_i$ and a residual noise $\boldsymbol{\epsilon}_i$ as:

$$\boldsymbol{\phi}_i = \mathbf{m} + \mathbf{U}\mathbf{y} + \mathbf{V}\mathbf{x}_i + \boldsymbol{\epsilon}_i \,. \tag{1}$$

Matrices $\mathbf{U}$ and $\mathbf{V}$ typically constrain the speaker and inter–session factors to be of lower dimension than the i–vectors space. PLDA estimates the distribution of the latent variables that maximize the likelihood of the observed i–vectors, assuming that i–vectors from the same speaker share the same speaker factor, i.e., the same value for latent variable $\mathbf{y}$ [3]. The simplest PLDA model assumes that all the hidden variables are Gaussian distributed, and that the noise term $\boldsymbol{\epsilon}_i$ has a full covariance matrix, so that the terms $\mathbf{V}\mathbf{x}_i$ and $\boldsymbol{\epsilon}_i$ in (1) can be merged. Thus,

Computational resources for this work were provided by HPC@POLITO (http://www.hpc.polito.it) Politecnico di Torino, Italy

an i–vector $\boldsymbol{\phi}_i$ is re–defined as:

$$\boldsymbol{\phi}_i = \mathbf{m} + \mathbf{U}\mathbf{y} + \boldsymbol{\epsilon}_i \,, \tag{2}$$

where the speaker factor $\mathbf{y}$ and the residual noise are distributed as:

$$\mathbf{y} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \quad \boldsymbol{\epsilon}_i \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Lambda}^{-1}) \,, \tag{3}$$

and $\boldsymbol{\Lambda}$ is the precision matrix of noise $\boldsymbol{\epsilon}_i$.

2.2. Two–covariance model

Further simplification of the PLDA model (2) is obtained assuming that the speaker and inter–session subspaces span the entire i–vector space. This simplified model, referred to as the Two–covariance model [4, 5], or 2–COV for short, accounts for two Gaussian–distributed components: the speaker component $\mathbf{y}$, and the inter–session variability component $\boldsymbol{\epsilon}_i$, which are combined to produce an i–vector as:

$$\boldsymbol{\phi}_i = \mathbf{m} + \mathbf{y} + \boldsymbol{\epsilon}_i \,, \tag{4}$$

where

$$\mathbf{y} \sim \mathcal{N}(\mathbf{0}, \mathbf{B}^{-1}) \quad \boldsymbol{\epsilon}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{W}^{-1}) \,, \tag{5}$$

and $\mathbf{B}^{-1}$ and $\mathbf{W}^{-1}$ are the between–speaker and within–speaker covariance matrix, respectively.

It has been shown in [7] that the 2–COV model log–likelihood ratio for an i–vector pair is a quadratic function, invariant to i–vector swapping, which can be formulated as:

$$\begin{aligned} s(\boldsymbol{\phi}_1, \boldsymbol{\phi}_2) \quad &= \quad \boldsymbol{\phi}_1^T \boldsymbol{\Lambda} \boldsymbol{\phi}_2 + \boldsymbol{\phi}_2^T \boldsymbol{\Lambda} \boldsymbol{\phi}_1 + \boldsymbol{\phi}_1^T \boldsymbol{\Gamma} \boldsymbol{\phi}_1 + \boldsymbol{\phi}_2^T \boldsymbol{\Gamma} \boldsymbol{\phi}_2 \\ &\quad + (\boldsymbol{\phi}_1 + \boldsymbol{\phi}_2)^T \mathbf{c} + k \,, \end{aligned} \tag{6}$$

where the within-speaker and between–speaker covariances are related to $\boldsymbol{\Lambda}, \boldsymbol{\Gamma}, \mathbf{c}$ and k according to:

$$\begin{aligned} &\boldsymbol{\Lambda} = \frac{1}{2}\mathbf{W}^T \tilde{\boldsymbol{\Lambda}} \mathbf{W} \qquad \boldsymbol{\Gamma} = \frac{1}{2}\mathbf{W}^T (\tilde{\boldsymbol{\Lambda}} - \tilde{\boldsymbol{\Gamma}}) \mathbf{W} \\ &\mathbf{c} = \mathbf{W}^T (\tilde{\boldsymbol{\Lambda}} - \tilde{\boldsymbol{\Gamma}}) \mathbf{B} \boldsymbol{\mu} \qquad k = \tilde{k} + \frac{1}{2}\left[(\mathbf{B}\boldsymbol{\mu})^T (\tilde{\boldsymbol{\Lambda}} - 2\tilde{\boldsymbol{\Gamma}}) \mathbf{B}\boldsymbol{\mu} \right] , \end{aligned} \tag{7}$$

with

$$\tilde{\boldsymbol{\Lambda}} = (\mathbf{B} + 2\mathbf{W})^{-1} \qquad \tilde{\boldsymbol{\Gamma}} = (\mathbf{B} + \mathbf{W})^{-1}$$

$$\tilde{k} = 2 \log |\tilde{\boldsymbol{\Gamma}}| - \log |\boldsymbol{B}| - \log |\tilde{\boldsymbol{\Lambda}}| + \boldsymbol{\mu}^T \mathbf{B} \boldsymbol{\mu} \,.$$

3. PSVM

A successful alternative to generative PLDA models has been presented in [7, 8], where a pairwise SVM model has been proposed, which is trained to discriminate between "same speaker" and "different speaker" pairs. This is in contrast with the usual "one-versus-all" framework, where an SVM model is created for each enrolled speaker, using as samples of the impostor class the utterances of a background cohort of speakers. This approach avoids the major weakness of "one-versus-all" SVM training, namely the scarcity of available samples for the target speakers.

In [7] it has been shown that the score of a second order Taylor expansion of an i–vector pair $\boldsymbol{\Phi} = (\boldsymbol{\phi}_1, \boldsymbol{\phi}_2)$ can be formulated as a function $s(\boldsymbol{\Phi})$, invariant to i–vector swapping, and that it leads to the same formulation of the 2–COV score (6). In particular, the second order Taylor expansion for $s(\boldsymbol{\Phi})$ around point $\hat{\boldsymbol{\Phi}} = \mathbf{0}$ is:

$$s(\boldsymbol{\Phi}) = s(\hat{\boldsymbol{\Phi}}) + (\boldsymbol{\Phi} \cdot \boldsymbol{\nabla} s|_{\hat{\boldsymbol{\Phi}}}) + \boldsymbol{\Phi}^T (\mathbf{H}(s)|_{\hat{\boldsymbol{\Phi}}}) \boldsymbol{\Phi} \,, \tag{8}$$

where $\boldsymbol{\nabla}$ is the vector of differential operators

$$\boldsymbol{\nabla} = \left(\frac{\partial}{\partial \Phi_1}, \ldots, \frac{\partial}{\partial \Phi_d} \right) \,,$$

d is the dimension of the i–vector pair, and $\mathbf{H}(s)$ is the Hessian of function $s(\boldsymbol{\Phi})$.

Defining:

$$s(\hat{\boldsymbol{\Phi}}) = k \,, \;\; \boldsymbol{\nabla} s|_{\hat{\boldsymbol{\Phi}}} = [\, \mathbf{c} \quad \mathbf{c} \,] \,, \;\; \mathbf{H}(s)|_{\hat{\boldsymbol{\Phi}}} = \begin{bmatrix} \boldsymbol{\Gamma} & \boldsymbol{\Lambda} \\ \boldsymbol{\Lambda} & \boldsymbol{\Gamma} \end{bmatrix} \,, \tag{9}$$

with a symmetric $\boldsymbol{\Lambda}$, we obtain the quadratic function of the i–vector pair:

$$\begin{aligned} s(\boldsymbol{\phi}_1, \boldsymbol{\phi}_2) \quad &= \quad \boldsymbol{\phi}_1^T \boldsymbol{\Lambda} \boldsymbol{\phi}_2 + \boldsymbol{\phi}_2^T \boldsymbol{\Lambda} \boldsymbol{\phi}_1 + \boldsymbol{\phi}_1^T \boldsymbol{\Gamma} \boldsymbol{\phi}_1 + \boldsymbol{\phi}_2^T \boldsymbol{\Gamma} \boldsymbol{\phi}_2 \\ &\quad + (\boldsymbol{\phi}_1 + \boldsymbol{\phi}_2)^T \mathbf{c} + k \,, \end{aligned} \tag{10}$$

which is identical to (6), but with matrices $\boldsymbol{\Lambda}$ and $\boldsymbol{\Gamma}$, and vector $\mathbf{c}$, estimated by means of a different objective function and training procedure. It is worth noting that the structure of (9) naturally arises from the symmetries of the problem (see Section III-E in [7]).

4. Generative Two–Gaussian model

The goal of PLDA, and of the other classifiers of the same family, is to model the distribution of the speaker and channel components of the ivectors. We propose, instead, to directly characterize the ivectors pairs $\boldsymbol{\Phi}_{ij} = \begin{bmatrix} \boldsymbol{\phi}_i \\ \boldsymbol{\phi}_j \end{bmatrix}$ by a simple generative Gaussian model.

Our main assumption is that the i–vector pairs, given their labels, are independently generated from the two Gaussian distributions:

$$\boldsymbol{\Phi}_{ij}^S \sim \mathcal{N}(\boldsymbol{\mu}_S, \boldsymbol{\Lambda}_S^{-1}) \qquad \boldsymbol{\Phi}_{ij}^D \sim \mathcal{N}(\boldsymbol{\mu}_D, \boldsymbol{\Lambda}_D^{-1}) \,, \tag{11}$$

where S and D refer to "same speaker" and "different speakers", respectively. This assumption is not accurate, because the complete set of i–vector pairs of a training dataset are, by definition, correlated. However, this working hypothesis allows obtaining a relevant simplification of the models.

The speaker verification log–likelihood ratio is simply computed as:

$$\log R = \log \mathcal{N}(\boldsymbol{\Phi}_{ij} | \boldsymbol{\mu}_S, \boldsymbol{\Lambda}_S^{-1}) - \log \mathcal{N}(\boldsymbol{\Phi}_{ij} | \boldsymbol{\mu}_D, \boldsymbol{\Lambda}_D^{-1}) \,. \tag{12}$$

A second assumption is that the two distributions have the same mean, i.e., $\boldsymbol{\mu}_S = \boldsymbol{\mu}_D$.

Recalling that the i–vector pair likelihoods must be invariant to i–vector swapping, the covariance matrices $\boldsymbol{\Lambda}_S^{-1}$ and $\boldsymbol{\Lambda}_D^{-1}$ must obey the following symmetry constraints:

$$\boldsymbol{\Lambda}_S^{-1} = \begin{bmatrix} \mathbf{A}_S & \mathbf{B}_S \\ \mathbf{B}_S & \mathbf{A}_S \end{bmatrix} \qquad \boldsymbol{\Lambda}_D^{-1} = \begin{bmatrix} \mathbf{A}_D & \mathbf{B}_D \\ \mathbf{B}_D & \mathbf{A}_D \end{bmatrix} \tag{13}$$

where, $\mathbf{A}_*$ and $\mathbf{B}_*$ are symmetric, and the mean of the distributions $\boldsymbol{\mu} = \boldsymbol{\mu}_S = \boldsymbol{\mu}_D$ are defined as $\boldsymbol{\mu} = \begin{bmatrix} \mathbf{m} \\ \mathbf{m} \end{bmatrix}$, $\mathbf{m}$ being a vector of the same dimensions of an i–vector. Although we expect that $\mathbf{B}_D \approx \mathbf{0}$, because the i–vectors should be uncorrelated, we do estimate this matrix from the data.

The equations for training $\mathbf{A}_D$ and $\mathbf{B}_D$ are illustrated in the next subsection, devoted to their estimation.

4.1. 2–GAU model training

The 2–GAU models are trained by maximizing the likelihood of the training pairs, under the i–vector pair independence assumption. Although the number of training pairs is huge, because it grows quadratically with the number of i–vectors, we here provide a fast solution that allows estimating the parameters of the 2–GAU model even with very large set of data. For the sake of clarity we assume that i–vectors have been centered, so that the mean of the two distributions is $\boldsymbol{\mu} = \mathbf{0}$, but $\boldsymbol{\mu}$ can be easily re–estimated extending the techniques detailed in the following.

Let δ_{ij}^S and δ_{ij}^D denote the indicator functions for the "same speaker" and "different speaker" pair classes, respectively:

$$\begin{aligned}\delta_{ij}^S &= \begin{cases} 1 & \text{if } \boldsymbol{\phi}_i, \boldsymbol{\phi}_j \text{ belong to "same speaker" class} \\ 0 & \text{otherwise} \end{cases} \\ \delta_{ij}^D &= 1 - \delta_{ij}^S \ . \end{aligned} \tag{14}$$

Maximum Likelihood estimate of the 2-GAU distribution parameters can be obtained in closed form [13] as:

$$\begin{aligned}\boldsymbol{\Lambda}_S^{-1} &= \frac{1}{N_S} \sum_{i,j} \delta_{ij}^S \boldsymbol{\Phi}_{ij} \boldsymbol{\Phi}_{ij}^T \\ \boldsymbol{\Lambda}_D^{-1} &= \frac{1}{N_D} \sum_{i,j} \delta_{ij}^D \boldsymbol{\Phi}_{ij} \boldsymbol{\Phi}_{ij}^T \end{aligned} \tag{15}$$

where N_S and N_D denote the number of "same–speaker" and "different speaker" pairs, respectively.
The "different speaker" covariance, $\boldsymbol{\Lambda}_D^{-1}$, can be alternatively computed as:

$$\boldsymbol{\Lambda}_D^{-1} = \frac{1}{N_D} \left(N_T \boldsymbol{\Lambda}_T^{-1} - N_S \boldsymbol{\Lambda}_S^{-1} \right) \ , \tag{16}$$

where

$$\boldsymbol{\Lambda}_T^{-1} = \frac{1}{N_T} \sum_{i,j} \boldsymbol{\Phi}_{ij} \boldsymbol{\Phi}_{ij}^T \ . \tag{17}$$

is the total covariance matrix of the pairs, and N_T is the number of pairs.

Direct computation of $\boldsymbol{\Lambda}_S^{-1}$ and $\boldsymbol{\Lambda}_D^{-1}$ by means of (15) entails a summation over all training pairs, which would have an overwhelming complexity of $O(N^2 d^2)$, where N and d are the number and the dimension of the i–vectors, respectively. However, by exploiting the block structure of (13), these covariances can be efficiently obtained. The "same–speaker" covariance, $\boldsymbol{\Lambda}_S^{-1}$, is compose of the blocks matrices $\mathbf{A}_S$ and $\mathbf{B}_S$ computed as:

$$\mathbf{A}_S = \frac{1}{N_S} \sum_{i,j} \delta_{ij}^S \boldsymbol{\phi}_i \boldsymbol{\phi}_i^T \quad \mathbf{B}_S = \frac{1}{N_S} \sum_{i,j} \delta_{ij}^S \boldsymbol{\phi}_i \boldsymbol{\phi}_j^T \ . \tag{18}$$

Since $\delta_{ij} = 1$ for all pairs belonging to speaker s, (18) can be rewritten, by substituting summations over the pairs with summations over the speakers, as:

$$\begin{aligned}\mathbf{A}_S &= \frac{1}{N_S} \sum_s \sum_{i|\boldsymbol{\phi}_i \in s} |s| \, \boldsymbol{\phi}_i \boldsymbol{\phi}_i^T \\ \mathbf{B}_S &= \frac{1}{N_S} \sum_s \left(\sum_{i|\boldsymbol{\phi}_i \in s} \boldsymbol{\phi}_i \right) \left(\sum_{j|\boldsymbol{\phi}_j \in s} \boldsymbol{\phi}_j \right)^T \ , \end{aligned} \tag{19}$$

where s denotes the set of i–vectors belonging to a speaker, and $|s|$ is its cardinality.

The total covariance matrix $\boldsymbol{\Lambda}_T^{-1}$ can be obtained from block matrices $\mathbf{A}_T$ and $\mathbf{B}_T$. By analogy with (19), we get:

$$\begin{aligned}\mathbf{A}_T &= \frac{1}{N_T} \sum_i N \boldsymbol{\phi}_i \boldsymbol{\phi}_i^T \\ \mathbf{B}_T &= \frac{1}{N_T} \left(\sum_i \boldsymbol{\phi}_i \right) \left(\sum_j \boldsymbol{\phi}_j \right)^T \ , \end{aligned} \tag{20}$$

and $\boldsymbol{\Lambda}_D^{-1}$ is obtained from (16).
Computing these statistics by using (19) and (20) has complexity $O(Nd^2)$, i.e., is linear with the number of i–vectors, thus very fast even for large datasets.
It is worth noting that these estimates are closely related to the i–vector within class and total covariances. However, since we maximize the likelihood of i–vector pairs, speakers providing more utterances have larger impact on the estimation of the covariances, as can be observed looking at matrices $\mathbf{A}_T$ and $\mathbf{B}_T$.

5. Relation to the PLDA and PSVM models

In the following we show that the 2–GAU model is closely related to the 2–COV and PSVM models.

5.1. Relation to PLDA

Let's recall that, if the noise term $\boldsymbol{\epsilon}_i$ has full covariance matrix, an i–vector is generated in the PLDA model according to (2), where i–vectors from the same speaker share the same value for the hidden variable $\mathbf{y}$. Thus, a pair of "same speaker" i–vectors, i.e., a pair sharing a single $\mathbf{y}$, is modeled as:

$$\boldsymbol{\Phi}_{ij}^S = \begin{bmatrix} \boldsymbol{\phi}_i \\ \boldsymbol{\phi}_j \end{bmatrix} = \begin{bmatrix} \mathbf{m} \\ \mathbf{m} \end{bmatrix} + \begin{bmatrix} \mathbf{U} \\ \mathbf{U} \end{bmatrix} \mathbf{y} + \begin{bmatrix} \boldsymbol{\epsilon}_i \\ \boldsymbol{\epsilon}_j \end{bmatrix} \ , \tag{21}$$

whereas, a "different speaker" i–vector pair is modeled as:

$$\boldsymbol{\Phi}_{ij}^D = \begin{bmatrix} \boldsymbol{\phi}_i \\ \boldsymbol{\phi}_j \end{bmatrix} = \begin{bmatrix} \mathbf{m} \\ \mathbf{m} \end{bmatrix} + \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{0} & \mathbf{U} \end{bmatrix} \begin{bmatrix} \mathbf{y}_i \\ \mathbf{y}_j \end{bmatrix} + \begin{bmatrix} \boldsymbol{\epsilon}_i \\ \boldsymbol{\epsilon}_j \end{bmatrix} . \tag{22}$$

Since each model is a linear combination of Gaussian–distributed variables, closed–form integration over the speaker variables is possible, which gives:

$$\boldsymbol{\Phi}_S \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Lambda}_S^{-1}) \quad , \boldsymbol{\Phi}_D \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Lambda}_D^{-1}), \tag{23}$$

where

$$\begin{aligned}\boldsymbol{\Lambda}_S^{-1} &= \begin{bmatrix} \mathbf{U}\mathbf{U}^T + \boldsymbol{\Lambda}^{-1} & \mathbf{U}\mathbf{U}^T \\ \mathbf{U}\mathbf{U}^T & \mathbf{U}\mathbf{U}^T + \boldsymbol{\Lambda}^{-1} \end{bmatrix} \\ \boldsymbol{\Lambda}_D^{-1} &= \begin{bmatrix} \mathbf{U}\mathbf{U}^T + \boldsymbol{\Lambda}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{U}\mathbf{U}^T + \boldsymbol{\Lambda}^{-1} \end{bmatrix} , \end{aligned} \tag{24}$$

and $\boldsymbol{\Lambda}$ is the noise precision matrix. Comparing (23) and (24) with (11) and (13), respectively, it can be observed that the PLDA model estimates a constrained solution for the covariance matrices of the 2–GAU model. However, the parameters of the PLDA model are estimated by maximizing the likelihood of the training i–vectors, whereas the parameters of the 2–GAU model are estimated by maximizing the likelihood of the training i–vector pairs. Although our original assumption - that the i–vector pairs, given their labels, are independent and identically distributed random variables - is not accurate, it will be shown in Section 8, devoted to the experiments, that it is does not affect the model accuracy.

5.2. Relation to PSVM

The PSVM approach was introduced as a discriminatively trained model derived from the 2–COV in [7], where it was shown that the scoring functions of the PSVM and of the 2–COV models are formally equivalent. This equivalence has also been stated without reference to the 2–COV model, as it has been recalled in Section 3. The 2–GAU model allows providing a novel interpretation of the PSVM scoring function. Developing the log–likelihood ratio of the 2–GAU model (12), and recalling that $\boldsymbol{\mu} = \boldsymbol{\mu}_S = \boldsymbol{\mu}_D$, from (12) one gets:

$$\begin{aligned}\log R &= k - \frac{1}{2}(\boldsymbol{\Phi}_{ij} - \boldsymbol{\mu})^T \boldsymbol{\Lambda}_S (\boldsymbol{\Phi}_{ij} - \boldsymbol{\mu}) \\ &\quad + \frac{1}{2}(\boldsymbol{\Phi}_{ij} - \boldsymbol{\mu})^T \boldsymbol{\Lambda}_D (\boldsymbol{\Phi}_{ij} - \boldsymbol{\mu}) \\ &= k + \frac{1}{2}(\boldsymbol{\Phi}_{ij} - \boldsymbol{\mu})^T (\boldsymbol{\Lambda}_D - \boldsymbol{\Lambda}_S)(\boldsymbol{\Phi}_{ij} - \boldsymbol{\mu}) \\ &= \tilde{k} + \boldsymbol{\Phi}_{ij}^T \mathbf{c} + \frac{1}{2}\boldsymbol{\Phi}_{ij}^T \mathbf{H} \boldsymbol{\Phi}_{ij} \,, \end{aligned} \tag{25}$$

where $\mathbf{H} = \boldsymbol{\Lambda}_D - \boldsymbol{\Lambda}_S$, $\mathbf{c} = -\mathbf{H}\boldsymbol{\mu}$, and $\tilde{k}$ collects all the terms that do not depend on the i–vector pair $\boldsymbol{\Phi}_{ij}$. Comparing (25) with (8) we can observe that the two expression are formally equivalent. We can thus interpret the PSVM framework as a discriminative approach for estimating the difference of the precision matrices $\boldsymbol{\Lambda}_D - \boldsymbol{\Lambda}_S$, and the (shared) mean of the distributions of our 2–GAU model.

6. Two–Heavy–Tailed Model

The simplest PLDA model assumes a Gaussian distribution for the prior parameters. However, in [3] it has been shown that ML estimation of the PLDA parameters under a Gaussian assumption fails to produce accurate models for i-vectors that are not length–normalized. Thus, Heavy–Tailed distributions for the model priors have been proposed leading to the Heavy-Tailed PLDA model (HT–PLDA). A similar assumption for the prior distribution can be used to model i–vector pairs leading to the Two–Heavy–Tailed distribution model. In particular, for each i–vector pair, we define a hidden variable ν_{ij} that is assumed to be an i.i.d. random variable generated from a Gamma distribution depending on the pair label as:

$$\nu_{ij,S} \sim \Gamma\left(\frac{a_S}{2}, \frac{a_S}{2}\right) \quad \nu_{ij,D} \sim \Gamma\left(\frac{a_D}{2}, \frac{a_D}{2}\right) \,, \tag{26}$$

where S, D denote the "same speaker" and "different speaker" hypothesis, and a_S, a_D are the parameters of the two Gamma distributions, respectively. We also assume that the pairs are i.i.d. distributed, given the pair label and the hidden variables, according to the Gaussian distributions:

$$\begin{aligned}\boldsymbol{\Phi}_{ij}^S | \nu_{ij|S} &\sim \mathcal{N}(\boldsymbol{\mu}_S, \boldsymbol{\Lambda}_S^{-1} \nu_{ij|S}^{-1}) \\ \boldsymbol{\Phi}_{ij}^D | \nu_{ij|D} &\sim \mathcal{N}(\boldsymbol{\mu}_D, \boldsymbol{\Lambda}_D^{-1} \nu_{ij|D}^{-1}) \,. \end{aligned} \tag{27}$$

Integrating over the hidden variables, it follows that the pairs are distributed according to the Student's t–distributions [13]:

$$\boldsymbol{\Phi}_{ij}^S \sim \mathcal{T}(\boldsymbol{\mu}_S, \boldsymbol{\Lambda}_S, a_S) \quad \boldsymbol{\Phi}_{ij}^D \sim \mathcal{T}(\boldsymbol{\mu}_D, \boldsymbol{\Lambda}_D^{-1}, a_D) \,, \tag{28}$$

with a_S and a_D degrees of freedom, respectively. The likelihood ratio of an i–vector pair can then be computed as the ratio between two t–distributions. It is worth noting that, in contrast with the other models, the separation surfaces produced by the 2–HT model are not constrained to be quadratic, but can have more general shapes.

6.1. Relation with the HT–PLDA model

The 2–HT model has many similarities with the HT–PLDA model. In particular, we can show that the HT–PLDA model, with some additional constraints, formally corresponds to a slight simplification of the 2-HT model in (27). The HT–PLDA model assumes, as in (1), that an i–vector is generated according to:

$$\boldsymbol{\phi}_i = \mathbf{m} + \mathbf{U}\mathbf{y} + \mathbf{V}\mathbf{x}_i + \bar{\boldsymbol{\epsilon}}_i \,, \tag{29}$$

but with these distributions:

$$\begin{aligned} \mathbf{y} | u_1 &\sim \mathcal{N}(\mathbf{0}, \mathbf{I} u_1^{-1}) \,, & u_1 &\sim \Gamma(\frac{a_1}{2}, \frac{a_1}{2}) \\ \mathbf{x}_i | u_{2i} &\sim \mathcal{N}(\mathbf{0}, \mathbf{I} u_{2i}^{-1}) \,, & u_2 &\sim \Gamma(\frac{a_2}{2}, \frac{a_2}{2}) \\ \bar{\boldsymbol{\epsilon}}_i | \nu_i &\sim \mathcal{N}(\mathbf{0}, \bar{\boldsymbol{\Lambda}}^{-1} \nu_i^{-1}) \,, & \nu_i &\sim \Gamma(\frac{b}{2}, \frac{b}{2}) \,, \end{aligned} \tag{30}$$

where u_1, u_{2i} and ν_i are independently distributed hidden variables, and a_1, a_2 and b are the parameters of the prior distributions for u_1, u_{2i} and ν_i, respectively.

We can simplify the HT–PLDA model assuming that, for every speaker, $\nu_i = u_1$ and $u_{2i} = u_1$ regardless of the utterance, with $u \sim \Gamma(\frac{a}{2}, \frac{a}{2})$. This simplification violates the independence assumptions of PLDA because it makes the priors for the speaker and channel factors dependent. However, it allows us to integrate over the hidden variables and, since the terms $\mathbf{V}\mathbf{x}$ and $\bar{\boldsymbol{\Lambda}}^{-1}$ have the same precision matrix scaling factor, they can be merged into a single term $\boldsymbol{\epsilon}_i$ with $\boldsymbol{\Lambda}^{-1} = \bar{\boldsymbol{\Lambda}}^{-1} + \mathbf{V}\mathbf{V}^T$, leading to:

$$\boldsymbol{\phi}_i = \mathbf{m} + \mathbf{U}\mathbf{y} + \boldsymbol{\epsilon}_i \,, \tag{31}$$

where

$$\mathbf{y} \sim \mathcal{T}(\mathbf{0}, \mathbf{I}, a) \quad \boldsymbol{\epsilon}_i \sim \mathcal{T}(\mathbf{0}, \boldsymbol{\Lambda}^{-1}, a) \tag{32}$$

In analogy with (11), the distributions for "same speaker" and "different speaker" i–vector pairs can be written as:

$$\boldsymbol{\Phi}_{ij}^S \sim \mathcal{T}(\boldsymbol{\mu}_S, \boldsymbol{\Lambda}_S^{-1}, a) \quad \boldsymbol{\Phi}_{ij}^D \sim \mathcal{T}(\boldsymbol{\mu}_D, \boldsymbol{\Lambda}_D^{-1}, a) \,, \tag{33}$$

where $\boldsymbol{\Lambda}_S$ and $\boldsymbol{\Lambda}_D$ are given by (24).
Comparing (33) with (27) it easy verifying that the HT–PLDA model is formally equivalent to the proposed approach if we assume that $\nu_{ij,S} = \nu_{ij,D} = \nu_{ij} \sim \Gamma(\frac{a}{2}, \frac{a}{2})$ in (27).

Without the proposed simplification, the HT–PLDA model cannot be transformed in the 2–HT model (28), and the speaker verification log–likelihood ratio cannot be computed in closed form, thus making the HT–PLDA model much more expensive in testing than our proposed 2–HT model.

6.2. 2–HT model training

In contrast with the 2–GAU approach, the estimation of the 2–HT parameters does not have a closed form solution, thus we resort to Expectation-Maximization estimation.
We will illustrate only the estimation of the model parameters for the "same speaker" distribution because the same approach can be used for estimating the parameters of the "different speaker" class.

6.2.1. Expectation step

The expectation step requires computing the posterior distribution for the $\nu_{ij,S}$ hidden variables, given the observations. Since

the Gamma distribution is a conjugate prior for the scaling factor of the precision matrix of the i–vector conditional distribution, the posterior for $\nu_{ij,S}$ is again a Gamma distribution with parameters:

$$\nu_{ij,S,\mathbf{\Phi}_{ij}} \sim \Gamma\left(\frac{a_S+2d}{2}, \frac{a_S+\mathbf{\Phi}_{ij}^T\mathbf{\Lambda}_S\mathbf{\Phi}_{ij}}{2}\right) , \quad (34)$$

where d is the i–vector dimension ($2d$ is, thus, the i–vector pair dimension). The expectations necessary for the M–step are:

$$\begin{aligned}\mathbb{E}\left[\nu_{ij,S}\right] &= \frac{a_S+2d}{a_S+\mathbf{\Phi}_{ij}^T\mathbf{\Lambda}_S\mathbf{\Phi}_{ij}} \\ \mathbb{E}\left[\log\nu_{ij,S}\right] &= \psi(\frac{a_S+2d}{2}) - \log\frac{a_S+\mathbf{\Phi}_{ij}^T\mathbf{\Lambda}_S\mathbf{\Phi}_{ij}}{2}\end{aligned} \quad (35)$$

where $\psi(\cdot)$ denotes the digamma function.

6.2.2. Maximization step

The objective to be maximized is:

$$\sum_{i,j}\delta_{ij}^S\mathbb{E}\left[\log P(\mathbf{\Phi}_{ij},\nu_{ij,S})\right] , \quad (36)$$

which corresponds to solving the problem:

$$\begin{aligned}\underset{\mathbf{\Lambda}_S,a_S}{\operatorname{argmax}}\sum_{i,j}\delta_{ij}^S\Bigg[&\frac{1}{2}\log|\mathbf{\Lambda}_S| - \frac{1}{2}\mathbb{E}\left[\nu_{ij,S}\right]\mathbf{\Phi}_{ij}^T\mathbf{\Lambda}_S\mathbf{\Phi}_{ij} \\ &- \log\Gamma\left(\frac{a_S}{2}\right) + \frac{a_S}{2}\log\frac{a_S}{2} + \\ &\left(\frac{a_S}{2}-1\right)\mathbb{E}\left[\log\nu_{ij,S}\right] - \frac{a_S}{2}\mathbb{E}\left[\nu_{ij,S}\right]\Bigg] .\end{aligned} \quad (37)$$

The solution for $\mathbf{\Lambda}_S$ is given by:

$$\mathbf{\Lambda}_S^{-1} = \frac{1}{N_S}\sum_{i,j}\delta_{ij}^S\mathbb{E}\left[\nu_{ij,S}\right]\mathbf{\Phi}_{ij}\mathbf{\Phi}_{ij}^T , \quad (38)$$

The parameter a_S is obtained as the solution of the equation:

$$\log\frac{a_S}{2} - \psi\left(\frac{a_S}{2}\right) = \frac{1}{N_S}\sum_{ij}\delta_{ij}^S\left(\mathbb{E}\left[\nu_{ij,S}\right] - \mathbb{E}\left[\log\nu_{ij,S}\right] - 1\right) \quad (39)$$

which has no closed–form solution, thus it is solved numerically by using a line search algorithm.

Computing $\mathbf{\Lambda}_S^{-1}$ by means of (38) is very expensive because a single EM iteration has $O(N^2d^2)$ computational complexity. However, (38) can be rewritten, in matrix form, as:

$$\mathbf{\Lambda}_S^{-1} = \frac{1}{N_S}\mathbf{\Theta}(\mathbf{\Delta}_S\circ\mathbf{N})\mathbf{\Theta}^T , \quad (40)$$

where $\mathbf{\Theta}$ is the matrix of all i–vectors $\mathbf{\Theta} = [\boldsymbol{\phi}_1,\ldots,\boldsymbol{\phi}_N]$, $\mathbf{\Delta}_S$ is the matrix defined as $\mathbf{\Delta}_{S\,i,j} = \delta_{ij}^S$, $\mathbf{N}$ is matrix with elements $\mathbf{N}_{ij} = \mathbb{E}[\nu_{ij,S}]$ and $\circ$ denotes the Hadamard product, so that the complexity of an EM iteration reduces to $O(N^2d)$. Although the training time for the 2–HT model remains much higher compared to the 2–GAU training time, the scoring complexity of the 2–HT model is comparable to the one the Gaussian models (and PSVM), because given the parameters, the distributions in (28), and the corresponding likelihood ratios can be easily computed.

Directly modeling the distributions of the "same speaker" and "different speaker" i–vector pairs allows, thus, an easier extension of the model to more complex distributions without incurring in expensive or even intractable formulations. Different models can be devised to better capture the underlying distribution of the i–vector pairs, such as Mixture Models instead of single Gaussian or t–distributions. The effectiveness of these approaches, however, has not yet been experimentally validated.

7. Model comparison

An illustration of the similarities and differences of the models considered in this paper is summarized in Figures 1 ((a)–(i)). The figures show the contour levels[1] of the scoring functions of the 2–COV, 2–GAU, PSVM, and 2–HT models, respectively. Darker areas correspond to higher scores for the "same class" hypothesis. The dots represent training pairs of a set of data randomly generated from Heavy–Tailed distributions. The figures do not show the data, but pairs of data, identified by their associated label: white for "same class" and black for "different class". The PLDA model is equivalent to the 2–COV model because no subspace dimension reduction is possible for uni–dimensional data, thus it is not represented in these figures.

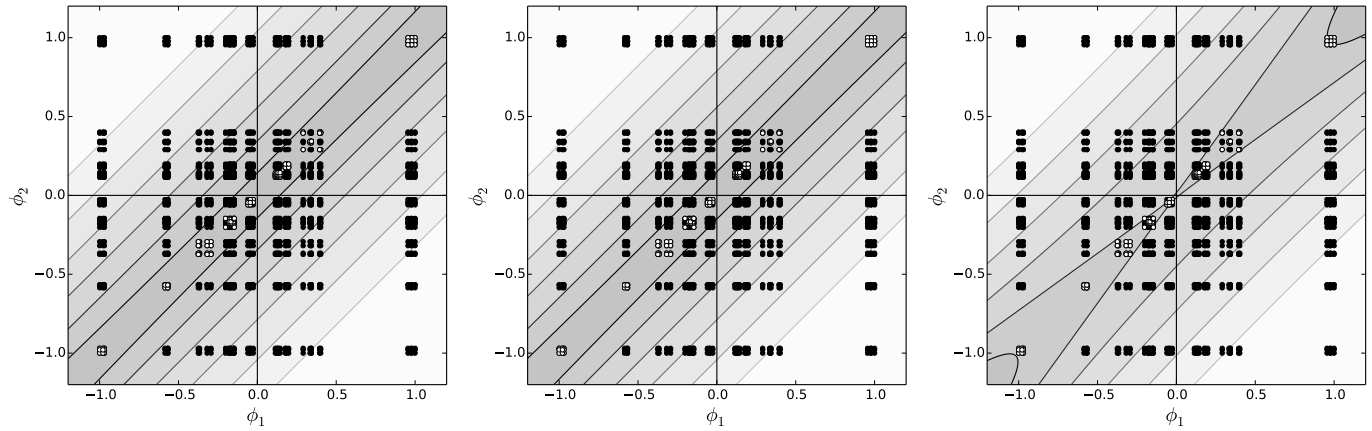
The contour plots clearly show that different separation regions are used by the classifiers. In particular, the 2–COV, 2–GAU, and PSVM classifiers have all quadratic shape separation regions, but they differ due to the different objective function that is optimized for each model. The shape of the separation regions for 2-HT model is not quadratic, and has, for this example, a sharper distribution of its log–likelihood ratio scores.

8. Experiments

The models have been tested on the female part of the tel-tel extended NIST 2010 evaluation trials [10] using a front–end based on 60-dimensional cepstral features. In particular, the i-vector extractor is based on a 2048-component full covariance gender-independent UBM, and on a gender-dependent $\mathbf{T}$ matrix. The UBM was trained on NIST SRE 2004, 2005 and 2006 data. The i–vector extractor has been trained from the same data, and in addition with Switchboard II Phases 2 and 3, Switchboard Cellular Parts 1 and 2, and Fisher datasets. For these experiments the dimension of the i-vectors was set to $d = 400$. All classifiers were trained with the complete set of data, excluding the Fisher dataset. The Gaussian PLDA system was implemented according to the framework illustrated in [3]. Since the 2–GAU and 2–HT models do not explicitly allow constraining the speaker space, Linear Discriminant Analysis (LDA) was used as an alternative approach for reducing the i–vector dimensions. The PLDA, 2–COV, and 2–GAU systems were trained with length–normalized i–vectors. Length–normalization was applied after mean removal and whitening of the i–vector covariance matrix. For the PLDA with low dimensional speaker subspace, covariance whitening was replaced by Within Class Covariance Normalization (WCCN)[2]. For the PSVM system, the i–vector were whitened by WCCN, but no length–normalization was applied, as we did in the experiments illustrated in[7]. Finally, no normalization was applied for the 2–HT system.

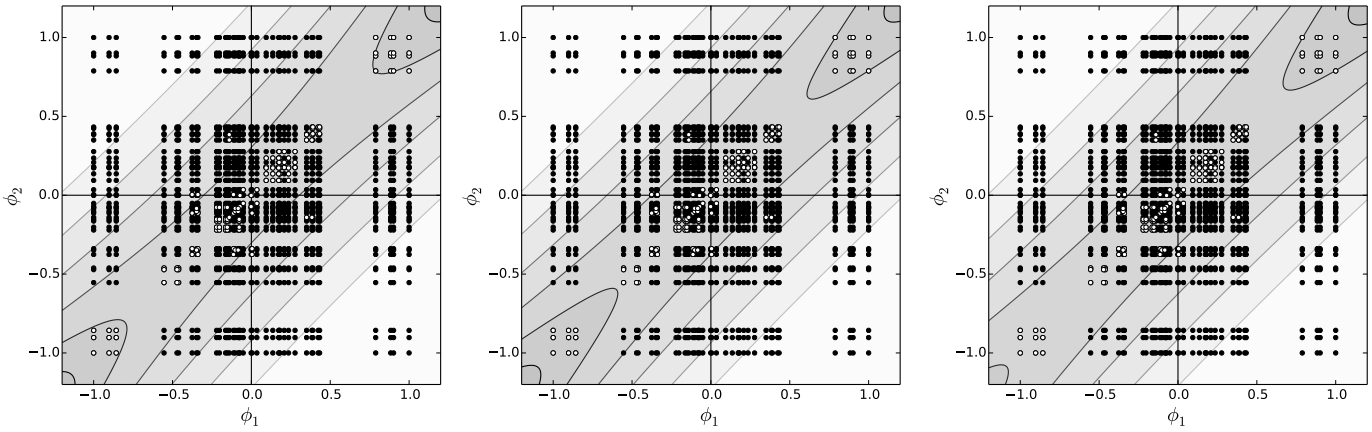
[1] A strictly monotone non–linear transformation of the scores has been performed to enhance the image quality

[2] The discussion of the appropriateness of the use of WCCN, rather than covariance whitening, as the i–vector pre-processing for PLDA models with low–dimensional speaker subspace compared to the noise space, is beyond the scope of this paper.



(a) 2–COV (b) 2–GAU (c) PSVM

(d) 2–HT (f) 2–HT



(g) 2–COV (h) 2–GAU (i) PSVM

Figure 1: Contour plots of the scoring functions of different classifiers for a set of pairs of uni–dimensional data. Data were randomly generated from Heavy–Tailed distributions. The first set of images refers to easily separable classes, whereas the second set refers to more noisy data. The "same class" and "different class" pairs are represented by white and black dots, respectively. Darker areas correspond to higher scores for the "same class" hypothesis.

Table 1: Comparison of the performance of PLDA, PSVM, 2–COV, 2–GAU, and 2–HT models on the female part of the tel-tel extended NIST 2010 evaluation trials. Systems marked by "*" pre–process i–vectors by using WCCN rather than covariance whitening. If relevant, the dimension of speaker or LDA subspace is given in parentheses.

System	Length Norm.	% EER	min DCF08	min DCF10
2–COV	Y	2.26	0.123	0.468
2–GAU	Y	2.32	0.121	0.451
PLDA* (U 120)	Y	2.00	0.100	0.339
PLDA (U 120)	Y	2.02	0.104	0.347
2–COV (LDA 120)	Y	2.08	0.105	0.347
2–GAU (LDA 120)	Y	2.08	0.105	0.341
PSVM*	N	2.39	0.110	0.320
2–HT (LDA 120)	N	2.27	0.109	0.328

Table 1 summarizes the performance of the evaluated models on the female part of the extended telephone condition in the NIST 2010 evaluation. The recognition accuracy is given in terms of Equal Error Rate (EER) and Minimum Detection Cost Functions defined by NIST for the 2008 (minDCF08) and 2010 (minDCF10) evaluations [10]. The scores were not normalized. The first two lines compare the performance of the 2–GAU model with the 2–COV model, corresponding to a PLDA model without speaker subspace dimensionality reduction. The two models give similar results, thus the 2–GAU model assumption about i–vector pair independence does not have significant impact on the performance. Constraining the speaker subspace by using a low–rank $\mathbf{U}$ or LDA shows significant improvement of the performance for both systems. Again, the 2–GAU classifier performs as well as the PLDA or 2–COV systems. Although not shown in the Table, these systems perform much worse without i–vector length–normalization. On the contrary, as shown in the last two rows, the PSVM and the 2–HT classifiers are able to achieve similar results without length–normalization.

9. Conclusions

A simple generative Gaussian model has been proposed using as its observations ivectors pairs rather than the i-vectors. We have highlighted the relations of this classifier with other pairwise classifiers, and we have shown that this simple model is able to achieve results that are comparable to the others approaches. The extension of this approach to the Two–Heavy–Tailed model has not given so far appreciable advantages, although it does not require i–vector pre–processing except LDA. Further developments are possible for the proposed approach, such as using Mixture Models, which are currently under test.

10. References

[1] N. Dehak, R. Dehak, P. Kenny, N. Brümmer, and P. Ouellet, "Support Vector Machines versus fast scoring in the low-dimensional total variability space for speaker verification," in *Proceedings of Interspeech 2009*, pp. 1559–1562, 2009.

[2] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front–end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[3] P. Kenny, "Bayesian speaker verification with heavy–tailed priors," in *Keynote presentation, Odyssey 2010, The Speaker and Language Recognition Workshop*, 2010. Available at `http://www.crim.ca/perso/patrick.kenny/kenny_Odyssey2010.pdf`.

[4] N.Brummer, "A farewell to SVM: Bayes factor speaker detection in supervector space," 2006. Available at `https://sites.google.com/site/nikobrummer/`.

[5] N. Brümmer and E. de Villiers, "The speaker partitioning problem," in *Proc. Odyssey 2010*, pp. 194–201, 2010.

[6] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian Mixture Models," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 31–44, 2000.

[7] S. Cumani, N. Brümmer, L. Burget, P. Laface, O. Plchot, and V. Vasilakakis, "Pairwise discriminative speaker verification in the i-vector space," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 6, pp. 1217–1227, 2013.

[8] S. Cumani, N. Brümmer, L. Burget, and P. Laface, "Fast discriminative speaker verification in the i–vector space," in *Proceedings of ICASSP 2011*, pp. 4852–4855, 2011.

[9] L. Burget, O. Plchot, S. Cumani, O. Glembek, P. Matějka, and N. Brümmer, "Discriminatively trained Probabilistic Linear Discriminant Analysis for speaker verification," in *Proceedings of ICASSP 2011*, pp. 4832–4835, 2011.

[10] "The NIST year 2010 speaker recognition evaluation plan." Available at `http://www.itl.nist.gov/iad/mig/tests/sre/2010/NIST_SRE10_evalplan.r6.pdf`.

[11] "The NIST year 2012 speaker recognition evaluation plan." Available at "`http://www.nist.gov/itl/iad/mig/upload/NIST_SRE12_evalplan-v17-r1.pdf`.

[12] S. J. D. Prince and J. H. Elder, "Probabilistic Linear Discriminant Analysis for inferences about identity," in *Proceedings of 11th International Conference on Computer Vision*, pp. 1–8, 2007.

[13] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.