Unifying Probabilistic Linear Discriminant Analysis Variants in Biometric Authentication

Aleksandr Sizov[1], Kong Aik Lee[2], Tomi Kinnunen[1]

[1] School of Computing, University of Eastern Finland, Finland
[2] Institute for Infocomm Research (I^{2}R), Singapore

Abstract. Probabilistic linear discriminant analysis (PLDA) is commonly used in biometric authentication. We review three PLDA variants — *standard*, *simplified* and *two-covariance* — and show how they are related. These clarifications are important because the variants were introduced in literature without argumenting their benefits. We analyse their predictive power, covariance structure and provide scalable algorithms for straightforward implementation of all the three variants. Experiments involve state-of-the-art speaker verification with i-vector features.

Keywords: PLDA, speaker and face recognition, i-vectors

1 Introduction

Biometric person authentication — recognizing persons from their physiological or behavioral traits — plays an increasingly important role in information security [1]. Face [2] and speaker [3] recognition are particularly attractive due to their unintrusiveness and low costs. Unfortunately, both involve prominent sample-to-sample variations that lead to decreased recognition accuracy; face images can be shot under differing lighting conditions or cameras and speech signals acquired using different microphones. Compensating for these nuisance variations is crucial for achieving robust recognition under varying conditions.

From various techniques studied, generative probabilistic models are among the top-performing ones for both face and speaker verification. A powerful, yet simple technique is *factor analysis* [4]. Given a feature vector that represents a single speech utterance or a face image, factor analysis captures the main correlations between its coordinates. A successful recent extension is the *probabilistic linear discriminant analysis* (PLDA) model [2, 5], where we split the total data variability into within-individual and between-individual variabilities, both residing on small-dimensional subspaces. Originally introduced in [2] for face recognition, PLDA has become a *de facto* standard in speaker recognition. We restrict our focus and experiments to speaker recognition but the general theory holds for arbitrary features.

Besides the original PLDA formulation [2], we are aware of two alternative variants that assume full covariance: *simplified* PLDA [6] and *two-covariance model* [7]. It is worth noting that the three models are related in terms of their

predictive power (degrees of freedom), covariance structure and computations. The main purpose of the current study is to provide a self-contained summary that elaborates the differences. The main benefit in doing so is that, instead of three different PLDA variants and learning algorithms, we show how to apply the *same* optimizer by merely modifying the latent subspace dimensions appropriately. As a further practical contribution, we provide an optimized open-source implementation[3].

2 Unified formulation of PLDA and its variants

We assume that the training set consists of K disjoint persons. For the i-th person we have n_i enrolment samples, each being represented by a single feature vector [4] $\boldsymbol{\phi}_{i\cdot}$. The PLDA models described below assume these vectors to be drawn from different generative processes.

2.1 Three types of a PLDA model

The first one is a **standard PLDA** as defined in the original study [2]:

$$\boldsymbol{\phi}_{ij} = \boldsymbol{\mu} + \mathbf{V}\mathbf{y}_i + \mathbf{U}\mathbf{x}_{ij} + \boldsymbol{\varepsilon}_{ij}\,, \tag{1}$$

$$\mathbf{y}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I})\,, \tag{2}$$

$$\mathbf{x}_{ij} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})\,, \tag{3}$$

$$\boldsymbol{\varepsilon}_{ij} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Lambda}^{-1})\,, \tag{4}$$

where $\boldsymbol{\phi} \in \mathbb{R}^{D\times 1}$, $\boldsymbol{\Lambda}$ is a diagonal precision matrix, $\boldsymbol{\mu}$ is a global mean, columns of the matrices $\mathbf{V} \in \mathbb{R}^{D\times P}$ and $\mathbf{U} \in \mathbb{R}^{D\times M}$ span the between- and within-individual subspaces. The second one is a **simplified PLDA** introduced in [6] and used in [9], [10], [11]:

$$\boldsymbol{\phi}_{ij} = \boldsymbol{\mu} + \mathbf{S}\mathbf{y}_i + \boldsymbol{\varepsilon}_{ij}\,, \tag{5}$$

$$\mathbf{y}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I})\,, \tag{6}$$

$$\boldsymbol{\varepsilon}_{ij} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Lambda}_f^{-1})\,, \tag{7}$$

where $\boldsymbol{\Lambda}_f$ is a full precision matrix instead of the diagonal matrix in the standard PLDA case and $\mathbf{S} \in \mathbb{R}^{D\times L}$. The third one is a **two-covariance model** introduced in [7] and used extensively in [12]:

$$\mathbf{y}_i \sim \mathcal{N}(\mathbf{y}_i|\boldsymbol{\mu}, \mathbf{B}^{-1})\,, \tag{8}$$

$$\boldsymbol{\phi}_{ij}|\mathbf{y}_i \sim \mathcal{N}(\boldsymbol{\phi}_{ij}|\mathbf{y}_i, \mathbf{W}^{-1})\,, \tag{9}$$

where both $\mathbf{B}$ and $\mathbf{W}$ are *full* precision matrices. Thus, unlike the two previous models, we no longer have any subspaces with reduced dimensionality.

[3] https://sites.google.com/site/fastplda/

[4] Traditionally, speech utterances have been represented as a sequence of acoustic feature vectors. In this paper we use the i-vector [8] representation that produces a fixed length vector from the variable length sequence. More on this in Section 4.

2.2 Exploring the structure of the models

All the latent variables in the standard PLDA formulation (1) have a Gaussian distribution. Thus, the distribution of the observed variables is also a Gaussian:

$$\boldsymbol{\phi}_{ij}|\mathbf{y}_i, \mathbf{x}_{ij} \sim \mathcal{N}(\boldsymbol{\phi}_{ij}|\boldsymbol{\mu} + \mathbf{V}\mathbf{y}_i + \mathbf{U}\mathbf{x}_{ij}, \boldsymbol{\Lambda}^{-1}), \tag{10}$$

and an integration of the channel latent variables $\{\mathbf{x}_{ij}\}$ leads to a closed-form result:

$$\boldsymbol{\phi}_{ij}|\mathbf{y}_i \sim \mathcal{N}(\boldsymbol{\phi}_{ij}|\boldsymbol{\mu} + \mathbf{V}\mathbf{y}_i, \mathbf{U}\mathbf{U}^\mathsf{T} + \boldsymbol{\Lambda}^{-1}). \tag{11}$$

We can now formulate (11) in a similar style as the two-covariance model:

$$\tilde{\mathbf{y}}_i \sim \mathcal{N}(\tilde{\mathbf{y}}_i|\boldsymbol{\mu}, \mathbf{V}\mathbf{V}^\mathsf{T}), \tag{12}$$

$$\boldsymbol{\phi}_{ij}|\tilde{\mathbf{y}}_i \sim \mathcal{N}(\boldsymbol{\phi}_{ij}|\tilde{\mathbf{y}}_i, \mathbf{U}\mathbf{U}^\mathsf{T} + \boldsymbol{\Lambda}^{-1}). \tag{13}$$

Comparing (12) with (8) and (13) with (9) reveals that the structure of a standard PLDA and a two-covariance model is the same and their only difference is in the covariance matrices. Let us call within- and between-individual covariance matrices of the n-th model as $\mathbf{W}_n^{-1}$ and $\mathbf{B}_n^{-1}$ (see Table 1), so that,

$$\mathbf{W}_3^{-1} = \mathbf{W}^{-1}, \tag{14}$$

$$\mathbf{B}_3^{-1} = \mathbf{B}^{-1}. \tag{15}$$

From (12) and (13) we conclude that

$$\mathbf{W}_1^{-1} = \mathbf{U}\mathbf{U}^\mathsf{T} + \boldsymbol{\Lambda}^{-1}, \tag{16}$$

$$\mathbf{B}_1^{-1} = \mathbf{V}\mathbf{V}^\mathsf{T}. \tag{17}$$

Applying the same analysis to the simplified PLDA leads to the following equations:

$$\mathbf{W}_2^{-1} = \boldsymbol{\Lambda}_f^{-1}, \tag{18}$$

$$\mathbf{B}_2^{-1} = \mathbf{S}\mathbf{S}^\mathsf{T}. \tag{19}$$

2.3 Calculating the degrees of freedom

We have seen that all the three models have the same structure, but their predictive powers differ because they have different number of independent parameters. It is a known fact that for a factor analysis model latent subspace has *rotational invariance* (see [4, Page 576]). If $\mathbf{R}$ is an arbitrary orthogonal matrix (that is, $\mathbf{R}\mathbf{R}^\mathsf{T} = \mathbf{R}^\mathsf{T}\mathbf{R} = \mathbf{I}$) then

$$\mathbf{B}_1^{-1} = \mathbf{V}\mathbf{V}^\mathsf{T} = \mathbf{V}(\mathbf{R}\mathbf{R}^\mathsf{T})\mathbf{V}^\mathsf{T} = (\mathbf{V}\mathbf{R})(\mathbf{V}\mathbf{R})^\mathsf{T}, \tag{20}$$

so that $\mathbf{V}$ and $\tilde{\mathbf{V}} = \mathbf{V}\mathbf{R}$ lead to the same covariance matrix and the same model. This ambiguity means that a particular solution is not unique.

In the two-covariance model both $\mathbf{W}_3^{-1}$ and $\mathbf{B}_3^{-1}$ are full and symmetric matrices so each of them has $D(D+1)/2$ degrees of freedom. In the case of standard PLDA, $\mathbf{W}_1^{-1} = \mathbf{U}\mathbf{U}^{\mathsf{T}} + \mathbf{\Lambda}^{-1}$ has $DM + D - M(M-1)/2$ degrees of freedom, where the second term is due to diagonal noise matrix and the last term is due to rotational invariance property. The same argument can be applied to the remaining matrices. Table 1 summarizes the degrees of freedom for each of the three models.

Table 1. Degrees of freedom for each model. Here, D is the dimensionality of the feature vectors, P and L are the number of basis vectors for between-individual subspaces of the corresponding models, M is a number of basis vectors for within-individual subspace, $\mathbf{B}_n$ and $\mathbf{W}_n$ are a between-individual and within-individual precision matrices for n-th model.

PLDA type	$\boldsymbol{\mu}$	$\mathbf{B}_n$	$\mathbf{W}_n$
$n=1$ (standard)	$D+$	$DP - \frac{P(P-1)}{2}+$	$DM + D - \frac{M(M-1)}{2}$
$n=2$ (simplified)	$D+$	$DL - \frac{L(L-1)}{2}+$	$\frac{D(D+1)}{2}$
$n=3$ (two-covariance)	$D+$	$\frac{D(D+1)}{2}+$	$\frac{D(D+1)}{2}$

Regarding the degrees of freedom, we conclude the following from Table 1:

1. When $L = D$ (factor loadings matrix is of full rank) the simplified PLDA is equivalent to the two-covariance model.
2. When $P = D$ and $M = D - 1$ the standard PLDA model is equivalent to the two-covariance model.
3. When $P = L$ and $M = D - 1$ the standard PLDA model is equivalent to the simplified PLDA.

To sum up, the standard PLDA is the most general model, and a two-covariance is the least general model.

2.4 Over-complete case

It is important to note that the above equations hold *only* when the dimensionality of the latent variables is less or equal to the dimensionality of the data. Otherwise, we have an over-complete basis for a latent variable subspace and we need an additional step before analysing the model. To this end, suppose that the matrix $\mathbf{V} \in \mathbb{R}^{D \times P}$ has more columns than D, then this matrix affects

generative process (12) only in the form $\mathbf{V}\mathbf{V}^\mathsf{T}$. When $P > D$, this $D \times D$ matrix has a rank D. As a symmetric positive-definite matrix, we may apply Cholesky decomposition to get,

$$\mathbf{V}\mathbf{V}^\mathsf{T} = \mathbf{L}\mathbf{L}^\mathsf{T}, \tag{21}$$

where $\mathbf{L} \in \mathbb{R}^{D \times D}$ is an upper triangular matrix. Without loss of generality, we can choose $\mathbf{V} = \mathbf{L}$ and transform an over-complete case to a complete one. The same argument holds for matrices $\mathbf{U}$ and $\mathbf{S}$.

2.5 Scoring

At verification stage we are given a pair of individual models: one created from the test features and the other from enrolment features of the claimed person and we need to decide whether these models belong to the same person. To do this in a PLDA approach we need to calculate a log-likelihood ratio between two hypothesis: both models share the same latent identity variable or they share a different identity variables. The scoring equations are the same for all models but due to lack of space we do not present them here. For an optimized scoring procedure please consult [13].

3 EM-algorithms

The original EM-algorithm proposed in [2] has a serious drawback: at the E-step we need to invert a matrix whose size grows linearly with the number of samples per individual. For large datasets this algorithm becomes highly impractical. A number of solutions for this problem have been introduced. In [14], the authors utilize a special matrix structure of PLDA model and manually derive equations for the required matrix inversions. In [15], the authors proposed a special change of variables that lead to a diagonalized versions of the required matrices. The most detailed derivations are given in [16]. Our version was based on [14] and accelerated in a similar style as in [16]. The algorithm 1 summarizes it and the details are presented in the appendix A.

Incomplete algorithm (only E-step) for the two-covariance model is given in [7]. Here we present complete solution in the form of short summary (see algorithm 2). The details are available in the appendix B.

Technical notes:

- The rank of matrix $\mathbf{V}$ is equal to the rank of $\mathbf{T}_\mathbf{y}$, which is just the number of individuals in the training set. So, this is an upper bound for the number of columns of matrix $\mathbf{V}$ that we should choose.
- If in the algorithm 1 we set matrix $\mathbf{U}$ to zero and do not constrain noise precision matrix $\mathbf{\Lambda}$ to be diagonal we get EM-algorithm for the simplified PLDA model [17].
- For the two-covariance model the number of individuals in the training set should be bigger than the dimensionality of feature vectors (i-vectors, in our case).

Algorithm 1: Scalable PLDA learning algorithm

Input: $\mathbf{\Phi} = \{\boldsymbol{\phi}_{ij}\}_{i=1,j=1}^{K,n_i}$, where K is a total number of persons, and n_i is the number of samples for i-th person.
Output: Estimated matrices $\mathbf{V}$, $\mathbf{U}$ and $\mathbf{\Lambda}$.
Sort persons according to the number of samples $\{n_i\}$;
Find total number of samples N and center the data (eq. A.1 and A.2) ;
Compute data statistics $\{\mathbf{f}_i\}$ and $\mathbf{S}$ (eq. A.3 and A.4) ;
Initialize $\mathbf{V}$ and $\mathbf{U}$ with small random values, $\mathbf{\Lambda} \leftarrow N\mathbf{S}^{-1}$;
repeat
  E-step:
    Set $\mathbf{R} \leftarrow 0$;
    Compute auxiliary matrices $\mathbf{Q}$, $\mathbf{J}$ (eq. A.5 and A.6) ;
    for $i = 1$ *to* K **do**
      if $n_i \neq n_{i-1}$ **then** compute $\mathbf{M}_i$ (eq. A.7);
      else $\mathbf{M}_i \leftarrow \mathbf{M}_{i-1}$;
      Find $\mathbb{E}[\mathbf{y}_i]$ (eq. A.8) ;
      Update $\mathbf{R}_{\mathbf{yy}}$ (eq. A.13);
    Calculate $\mathbf{T}$, $\mathbf{R}_{\mathbf{yx}}$ and $\mathbf{R}_{\mathbf{xx}}$ (eq. A.12, A.14 and A.15) ;
  M-step:
    Find $\mathbf{V}$, $\mathbf{U}$, $\mathbf{\Lambda}$ (eq. A.16 and A.17) ;
  MD-step:
    Compute auxiliary matrices $\boldsymbol{\mathcal{Y}}$, $\mathbf{G}$, $\boldsymbol{\mathcal{X}}$ (eq. A.18, A.19 and A.20) ;
    Update $\mathbf{U}$, $\mathbf{V}$ (eq. A.21 and A.22) ;
until Convergence ;

Algorithm 2: Two-covariance model learning algorithm

Input: $\mathbf{\Phi} = \{\boldsymbol{\phi}_{ij}\}_{i=1,j=1}^{K,n_i}$, where K is a total number of persons, and n_i is a number of samples for i-th person.
Output: Estimated matrices $\boldsymbol{\mu}$, $\mathbf{B}$ and $\mathbf{W}$.
Sort persons according to the number of samples $\{n_i\}$;
Compute data statistics N, $\{\mathbf{f}_i\}$ and $\mathbf{S}$ (eq. B.1, B.2 and B.3) ;
Initialize $\boldsymbol{\mu}$, $\mathbf{B}$, $\mathbf{W}$;
repeat
  E-step:
    Set $\mathbf{T} \leftarrow 0$, $\mathbf{R} \leftarrow 0$, $\boldsymbol{\mathcal{Y}} \leftarrow 0$;
    for $i = 1$ *to* K **do**
      if $n_i \neq n_{i-1}$ **then** compute $\mathbf{L}_i$ (eq. B.4);
      else $\mathbf{L}_i \leftarrow \mathbf{L}_{i-1}$;
      Find $\mathbb{E}[\mathbf{y}_i]$ and $\mathbb{E}[\mathbf{y}_i\mathbf{y}_i^{\mathsf{T}}]$ (eq. B.5, B.6) ;
      Update $\mathbf{T}$, $\mathbf{R}$ and $\boldsymbol{\mathcal{Y}}$ (eq. B.8, B.9 and B.10);
  M-step:
    Find $\boldsymbol{\mu}$, $\mathbf{B}$ and $\mathbf{W}$ (eq. B.11, B.12 and B.13) ;
until Convergence ;

4 Experiments

4.1 System setup

In modern speaker and language recognition, a speech utterance can be represented using its *i-vector* [8]. Briefly, variable-duration feature sequences are first mapped to utterance-specific Gaussian mixture models (GMMs). The i-vector is a low-dimensional latent representation of the corresponding GMM mean supervector [18], typical dimensionality varying from 400 to 600. This is sufficiently low to robustly estimate a full within-individual variation covariance matrix [6].

Our i-vector system uses standard Mel-frequency cepstral coefficient (MFCC) features involving RASTA filter, delta and double delta coefficients, energy-based speech activity detector [19] and utterance level cepstral mean and variance normalization (CMVN), in this order. Gender-dependent universal background models (UBMs) were trained with data from NIST 2004, 2005 and 2006 data and gender-dependent i-vector extractors from NIST 2004, 2005, 2006, Fisher and Switchboard. For more details, see [20]. For the experiments we used only female subset which has 578 train speakers, 21216 train segments, 459 test speakers, 10524 target trails and 6061824 non-target trials.

The UBM has 1024 Gaussians and i-vector dimensionality is set to 600. The i-vectors are whitened and length-normalized [9]. Speaker verification accuracy is measured through the equal error rate (EER) corresponding to the operating point with equal false acceptance and false rejection rates.

4.2 Comparison of different PLDA configurations

We made a thorough comparison of different PLDA configurations. Since PLDA training uses random initialization, we made 10 independent runs for each tested configuration and averaged the EERs. Although usually PLDA models achieve the best performance when they are slightly under-trained, the number of iterations and relative increase in a log-likelihood at the optimal point are different for every configuration. That is why in this experiment we set the number of iterations to 50, that was more than enough for the convergence in all cases.

The averaged EERs are presented in Fig. 1. Here, we fix the number of columns of one subspace matrix and vary the other. Our training dataset has only 578 unique speakers that is why to compare standard and simplified PLDA to the two-covariance model we applied LDA to reduce the dimensionality to be 550.

The figures clearly show that for the 600-dimensional i-vectors channel subspace should be as large as possible whereas after LDA projection the channel variability is compensated and the best performance is achieved when matrix $\mathbf{U}$ is set to zero.

Another interesting finding is that usually deviations from the standard PLDA show better performance even when they are supposed to be theoretically equivalent. It could be the result of simpler EM-algorithms with less intermediate steps and matrices.

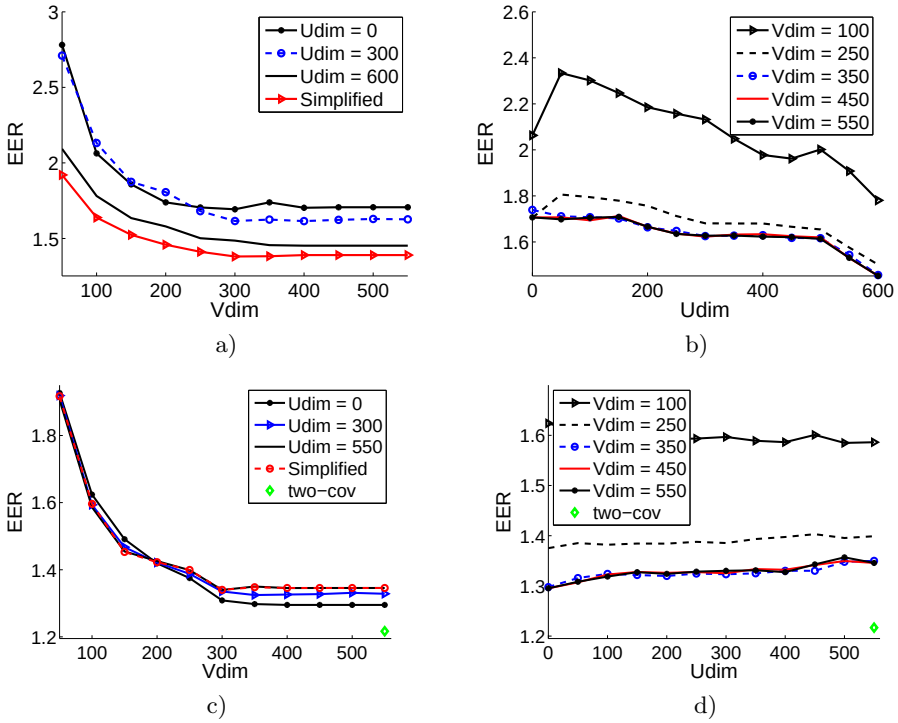

Fig. 1. Comparison of different configuration of the standard PLDA model with simplified and two-covariance models. Here, **Vdim** is a number of basis vectors for between-individual subspace (number of columns in matrix **V**), **Udim** is a number of basis vectors for within-individual subspace (number of columns in matrix **U**). Experiments a) and b) is done on the uncompressed i-vectors with 600 dimensions, c) and d) — on the LDA-projected 550-dimensional i-vectors.

5 Conclusion

We compared the standard, simplified and two-covariance PLDA variants. We have shown that the standard PLDA is the most general formulation and that, for certain configurations, it is equivalent to the other two models in terms of the predictive power. Our experimental results suggested that it is better to use the simplest possible model suited for the particular application. We presented the algorithms for all three models and shared their implementation online.

References

1. A.K. Jain, A. Ross, and S. Pankati. Biometrics: A tool for information security. *IEEE-TIFS*, 1(2):125–143, June 2006.
2. S.J.D. Prince and J.H. Elder. Probabilistic linear discriminant analysis for inferences about identity. In *IEEE 11th ICCV*, pages 1–8, Oct 2007.

3. T. Kinnunen and H. Li. An overview of text-independent speaker recognition: from features to supervectors. *Speech communication*, 52(1):12–40, 2010.
4. C.M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.
5. P. Li, Y. Fu, U. Mohammed, J. H Elder, and S.J.D. Prince. Probabilistic models for inference about identity. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(1):144–157, 2012.
6. P. Kenny. Bayesian speaker verification with heavy tailed priors. In *Proc. of the Odyssey Speak. and Lan. Recog. Workshop, Brno, Czech Republic*, 2010.
7. N. Brümmer and E. De Villiers. The speaker partitioning problem. In *Proc. of the Odyssey Speak. and Lan. Recog. Workshop, Brno, Czech Republic*, 2010.
8. 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*, 19(4):788–798, 2011.
9. D. Garcia-Romero and C. Espy-Wilson. Analysis of i-vector length normalization in speaker recognition systems. In *Interspeech*, pages 249–252, 2011.
10. B. Vesnicer, J.Z. Gros, N. Pavešić, and V. Štruc. Face recognition using simplified probabilistic linear discriminant analysis. *International Journal of Advanced Robotic Systems*, 9, 2012.
11. P. Rajan, T. Kinnunen, and V. Hautamäki. Effect of multicondition training on i-vector PLDA configurations for speaker recognition. In *Proc. Interspeech*, 2013.
12. J.A. Villalba and N. Brümmer. Towards fully bayesian speaker recognition: Integrating out the between-speaker covariance. In *Interspeech*, 2011.
13. P. Rajan, A. Afanasyev, V. Hautamki, and T. Kinnunen. From single to multiple enrollment i-vectors: Practical PLDA scoring variants for speaker verification. *Digital Signal Processing (to appear)*, 2014.
14. Y. Jiang, K.A. Lee, Z. Tang, B. Ma, A. Larcher, and H. Li. PLDA modeling in i-vector and supervector space for speaker verification. In *Interspeech*, 2012.
15. E.L. Shafey, C. McCool, R. Wallace, and S. Marcel. A scalable formulation of probabilistic linear discriminant analysis: applied to face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(7):1788–1794, 2013.
16. N. Brümmer. EM for probabilistic LDA. Technical report, Agnitio Research, Cape Town, 2010. `sites.google.com/site/nikobrummer/`.
17. T. Minka. Old and new matrix algebra useful for statistics. Technical report, MIT, 2000. `http://research.microsoft.com/en-us/um/people/minka/papers/matrix/`.
18. W.M. Campbell, D.E. Sturim, D.A. Reynolds, and A. Solomonoff. SVM based speaker verification using a GMM supervector kernel and NAP variability compensation. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 1, pages 97–100. IEEE, 2006.
19. T. Kinnunen and P. Rajan. A practical, self-adaptive voice activity detector for speaker verification with noisy telephone and microphone data. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2013.
20. R. Saeidi, K.A. Lee, T. Kinnunen, et al. I4U submission to NIST SRE 2012: A large-scale collaborative effort for noise-robust speaker verification. In *Proc. Interspeech*, 2013.
21. N. Brümmer. The EM algorithm and minimum divergence. Technical report, Agnitio Research, Cape Town, 2009. `sites.google.com/site/nikobrummer/`.
22. J. Luttinen and A. Ilin. Transformations in variational bayesian factor analysis to speed up learning. *Neurocomputing*, 73(79):1093 – 1102, 2010.

A EM-algorithm for standard/simplified PLDA

Suppose that we have K individuals in total and the i-th person has n_i enrolment samples $\{\boldsymbol{\phi}_{ij}\}_{j=1}^{n_i}$. It is more convenient to subtract the global mean from the data before learning the model. Let

$$\boldsymbol{\mu} = \frac{1}{N}\sum_{i,j}\boldsymbol{\phi}_{ij}\,, \tag{A.1}$$

where $N = \sum_{i=1}^{K} n_i$ is a global zero-order moment (total number of PLDA training vectors). We centralize the data

$$\boldsymbol{\varphi}_{ij} = \boldsymbol{\phi}_{ij} - \boldsymbol{\mu} \tag{A.2}$$

and define the first-order moment for the i-th person as

$$\mathbf{f}_i = \sum_{j=1}^{n_i}\boldsymbol{\varphi}_{ij}, \tag{A.3}$$

and the global second-order moment as

$$\mathbf{S} = \sum_{ij}\boldsymbol{\varphi}_{ij}\boldsymbol{\varphi}_{ij}^{\mathsf{T}}. \tag{A.4}$$

In the E-step we first pre-compute the following matrices:

$$\mathbf{Q} = (\mathbf{U}^{\mathsf{T}}\boldsymbol{\Lambda}\mathbf{U} + \mathbf{I})^{-1} \tag{A.5}$$

$$\mathbf{J} = \mathbf{U}^{\mathsf{T}}\boldsymbol{\Lambda}\mathbf{V} \tag{A.6}$$

$$\mathbf{M}_i = (n_i\mathbf{V}^{\mathsf{T}}\boldsymbol{\Lambda}(\mathbf{V} - \mathbf{U}\mathbf{Q}\mathbf{J}) + \mathbf{I})^{-1}\,, \tag{A.7}$$

where the matrices $\mathbf{U}$, $\mathbf{V}$ and $\boldsymbol{\Lambda}$ as defined in (1) and (4). After that we can easily find the first moments of the latent variables:

$$\mathbb{E}[\mathbf{y}_i] = \mathbf{M}_i(\mathbf{V} - \mathbf{U}\mathbf{Q}\mathbf{J})^{\mathsf{T}}\boldsymbol{\Lambda}\boldsymbol{f}_i \tag{A.8}$$

$$\mathbb{E}[\mathbf{x}_{ij}] = \mathbf{Q}(\mathbf{U}^{\mathsf{T}}\boldsymbol{\Lambda}\boldsymbol{\varphi}_{ij} - \mathbf{J}\mathbb{E}[\mathbf{y}_i]) \tag{A.9}$$

Let us define $\mathbf{z}_{ij}^{\mathsf{T}} = \begin{bmatrix}\mathbf{y}_i^{\mathsf{T}} & \mathbf{x}_{ij}^{\mathsf{T}}\end{bmatrix}$. In the M-step, we need an aggregated second moment of the compound variables $\mathbf{z}_{ij}$:

$$\mathbf{R} = \sum_{ij}\mathbb{E}\left[\begin{pmatrix}\mathbf{y}_i\\ \mathbf{x}_{ij}\end{pmatrix}\begin{pmatrix}\mathbf{y}_i^{\mathsf{T}} & \mathbf{x}_{ij}^{\mathsf{T}}\end{pmatrix}\right] = \begin{bmatrix}\mathbf{R}_{\mathbf{yy}} & \mathbf{R}_{\mathbf{yx}}\\ \mathbf{R}_{yx}^{\mathsf{T}} & \mathbf{R}_{\mathbf{xx}}\end{bmatrix} \tag{A.10}$$

where

$$\mathbb{E}[\mathbf{z}_{ij}\mathbf{z}_{ij}^{\mathsf{T}}] = \begin{bmatrix}\mathbf{M}_i & -\mathbf{M}_i\mathbf{J}^{\mathsf{T}}\mathbf{Q}^{\mathsf{T}}\\ -\mathbf{Q}\mathbf{J}\mathbf{M}_i & \mathbf{Q} + \mathbf{Q}\mathbf{J}\mathbf{M}_i\mathbf{J}^{\mathsf{T}}\mathbf{Q}^{\mathsf{T}}\end{bmatrix} + \mathbb{E}[\mathbf{z}_{ij}]\mathbb{E}[\mathbf{z}_{ij}]^{\mathsf{T}} \tag{A.11}$$

$$\mathbf{T} = \sum_{ij} \mathbb{E}[\mathbf{z}_{ij}] \boldsymbol{f}_i^{\mathsf{T}} = \sum_{ij} \mathbb{E} \begin{bmatrix} \mathbf{y}_i \\ \mathbf{x}_{ij} \end{bmatrix} \boldsymbol{f}_i^{\mathsf{T}} = \begin{bmatrix} \mathbf{T_y} \\ \mathbf{T_x} \end{bmatrix} = \begin{bmatrix} \mathbf{T_y} \\ \mathbf{Q}(\mathbf{U}^{\mathsf{T}} \boldsymbol{\Lambda} \mathbf{S} - \mathbf{J}\mathbf{T_y}) \end{bmatrix} \tag{A.12}$$

$$\mathbf{R_{yy}} = \sum_i n_i (\mathbf{M}_i + \mathbb{E}[\mathbf{y}_i]\mathbb{E}[\mathbf{y}_i]^{\mathsf{T}}) \tag{A.13}$$

$$\mathbf{R_{yx}} = (\mathbf{T_y} \boldsymbol{\Lambda} \mathbf{U} - \mathbf{R_{yy}} \mathbf{J}^{\mathsf{T}}) \mathbf{Q} \tag{A.14}$$

$$\mathbf{R_{xx}} = \mathbf{Q}(\mathbf{U}^{\mathsf{T}} \boldsymbol{\Lambda} \mathbf{S} \boldsymbol{\Lambda} \mathbf{U} - \mathbf{U}^{\mathsf{T}} \boldsymbol{\Lambda} \mathbf{T}_{\mathbf{y}}^{\mathsf{T}} \mathbf{J}^{\mathsf{T}} - \mathbf{J}\mathbf{T_y} \boldsymbol{\Lambda} \mathbf{U} + \mathbf{J}\mathbf{R_{yy}} \mathbf{J}^{\mathsf{T}}) \mathbf{Q} + N\mathbf{Q} \tag{A.15}$$

At the M-step we update the matrices $\mathbf{V}$, $\mathbf{U}$ and $\boldsymbol{\Lambda}$ as following

$$\begin{bmatrix} \mathbf{V} & \mathbf{U} \end{bmatrix} = \mathbf{T}^{\mathsf{T}} \mathbf{R}^{-1} \tag{A.16}$$

$$\boldsymbol{\Lambda}^{-1} = \frac{1}{N} \mathrm{diag} \left\{ \mathbf{S} - \begin{bmatrix} \mathbf{V} & \mathbf{U} \end{bmatrix} \mathbf{T} \right\} \tag{A.17}$$

To speed up convergence it is highly recommended to apply a so-called *minimum-divergence* (MD) step as well [21], [22]. During this step we assume that a prior for the latent variables $\{\mathbf{y}_i\}$ and $\{\mathbf{x}_{ij}\}$ could be in a non-standard Gaussian form, maximize w.r.t. its parameters and then find equivalent representation but with a standard prior. This step is very efficient against saddle-points. For MD-step we need a number of auxiliary matrices:

$$\boldsymbol{\mathcal{Y}} = \frac{1}{K} \sum_i \left(\mathbf{M}_i + \mathbb{E}[\mathbf{y}_i]\mathbb{E}[\mathbf{y}_i]^{\mathsf{T}} \right), \tag{A.18}$$

$$\mathbf{G} = \mathbf{R}_{\boldsymbol{yx}}^{\mathsf{T}} \mathbf{R}_{\boldsymbol{yy}}^{-1}, \tag{A.19}$$

$$\boldsymbol{\mathcal{X}} = \frac{1}{N} (\mathbf{R_{xx}} - \mathbf{G}\mathbf{R}_{\boldsymbol{yx}}). \tag{A.20}$$

After that it is enough to apply the following transformations:

$$\mathbf{U} \leftarrow \mathbf{U} \mathrm{chol}(\boldsymbol{\mathcal{X}}), \tag{A.21}$$

$$\mathbf{V} \leftarrow \mathbf{V} \mathrm{chol}(\boldsymbol{\mathcal{Y}}) + \mathbf{U}\mathbf{G}. \tag{A.22}$$

where $\mathrm{chol}(\boldsymbol{\mathcal{X}})$ is a Cholesky decomposition of the matrix $\boldsymbol{\mathcal{X}}$. The algorithm 1 presents a compact version of the derivations above.

B EM-algorithm for two-covariance model

As before we have K individuals in total and the i-th person has n_i enrolment samples $\{\boldsymbol{\phi}_{ij}\}_{j=1}^{n_i}$. Let's define a global zero-order moment:

$$N = \sum_{i=1}^{K} n_i, \tag{B.1}$$

the first-order moment for the i-th person as

$$\mathbf{f}_i = \sum_{j=1}^{n_i} \boldsymbol{\phi}_{ij}\,, \tag{B.2}$$

and the global second-order moment as

$$\mathbf{S} = \sum_{ij} \boldsymbol{\phi}_{ij}\boldsymbol{\phi}_{ij}^{\mathsf{T}}\,. \tag{B.3}$$

In the E-step we first pre-compute the following matrices

$$\mathbf{L}_i = \mathbf{B} + n_i\mathbf{W}\,, \tag{B.4}$$

where the matrices $\mathbf{B}$ and $\mathbf{W}$ are defined in (8) and (9). After that we can easily find the first and second moments of the latent variables:

$$\mathbb{E}[\mathbf{y}_i] = \mathbf{L}^{-1}\boldsymbol{\gamma}\,, \tag{B.5}$$

$$\mathbb{E}[\mathbf{y}_i\mathbf{y}_i^{\mathsf{T}}] = \mathbf{L}_i^{-1} + \mathbb{E}[\mathbf{y}_i]\mathbb{E}[\mathbf{y}_i]^{\mathsf{T}}, \tag{B.6}$$

where

$$\boldsymbol{\gamma} = \mathbf{B}\boldsymbol{\mu} + \mathbf{W}\mathbf{f}_i\,. \tag{B.7}$$

At the M-step we need to compute the following matrices

$$\mathbf{T} = \sum_i \mathbb{E}[\mathbf{y}_i]\mathbf{f}_i^{\mathsf{T}}\,, \tag{B.8}$$

$$\mathbf{R} = \sum_i n_i\mathbb{E}[\mathbf{y}_i\mathbf{y}_i^{\mathsf{T}}]\,, \tag{B.9}$$

$$\boldsymbol{\mathcal{Y}} = \sum_i n_i\mathbb{E}[\mathbf{y}_i]\,. \tag{B.10}$$

After that we update the parameters $\boldsymbol{\mu}$, $\mathbf{B}$ and $\mathbf{W}$ as follows

$$\boldsymbol{\mu} = \frac{1}{N}\boldsymbol{\mathcal{Y}}\,, \tag{B.11}$$

$$\mathbf{B}^{-1} = \frac{1}{N}(\mathbf{R} - (\boldsymbol{\mu}\boldsymbol{\mathcal{Y}}^{\mathsf{T}} + \boldsymbol{\mathcal{Y}}\boldsymbol{\mu}^{\mathsf{T}})) + \boldsymbol{\mu}\boldsymbol{\mu}^{\mathsf{T}}\,, \tag{B.12}$$

$$\mathbf{W}^{-1} = \frac{1}{N}(\mathbf{S} - (\mathbf{T} + \mathbf{T}^{\mathsf{T}}) + \mathbf{R})\,. \tag{B.13}$$

The algorithm 2 presents a compact version of the derivations above.