

Likelihood-ratio calibration using prior-weighted proper scoring rules

Niko Brümmer¹ and George Doddington²

¹AGNITIO Research, Somerset West, South Africa

²Walnut Creek, CA, USA

niko.brummer@gmail.com, george.doddington@comcast.net

Abstract

Prior-weighted logistic regression has become a standard tool for calibration in speaker recognition. Logistic regression is the optimization of the expected value of the logarithmic scoring rule. We generalize this via a parametric family of proper scoring rules. Our theoretical analysis shows how different members of this family induce different relative weightings over a spectrum of applications of which the decision thresholds range from low to high. Special attention is given to the interaction between prior weighting and proper scoring rule parameters. Experiments on NIST SRE'12 suggest that for applications with low false-alarm rate requirements, scoring rules tailored to emphasize higher score thresholds may give better accuracy than logistic regression.

Index Terms: speaker recognition, calibration, proper scoring rule

1. Introduction

We are concerned with the problem of speaker detection, where an automatic speaker recognizer is used to decide whether or not the voice of a designated target speaker is present in a given speech segment. Current speaker recognition algorithms output uncalibrated *scores*, which have to be processed through a calibration stage before cost effective decisions can be made. This paper deals with the problem of designing the calibration stage.

We implement the calibration stage in the form of a discriminative model, say \mathcal{M} , which outputs a posterior probability that the target speaker spoke the speech segment: $P(\text{tar}|s, \mathcal{M})$. The trial score, s , is computed by the speaker recognizer as a function of the target speaker and the speech segment. This posterior can be used in a straight-forward, standard way to make minimum-expected-cost Bayes decisions.

We shall take as the baseline discriminative model prior-weighted logistic regression [1], which has become a standard recipe for calibration in speaker recognition, with implementations available in the FoCal [2] and BOSARIS [3] toolkits.

Logistic regression is the minimization of the expected value of a special cost function, known as the *logarithmic proper scoring rule*. We are interested here in generalizing this recipe by modifying the cost function. Our motivation derives from [4], where it was demonstrated that a modified logistic regression that ignores scores below a suitable threshold can benefit applications with low false-alarm rate requirements. In this work we limit ourselves to cost functions which are *proper scoring rules* [5]. We expand on our previous work [6], to demonstrate theoretically and experimentally that we can tailor proper scoring rules to target the low false-alarm region.

2. Proper scoring rules

Given a database of supervised trials, the sum over trials of a proper scoring rule forms an objective function that can be used to simply evaluate the goodness of a recognizer with probabilistic output, and indeed also to facilitate discriminative training of such recognizers.

We restrict ourselves to *binary* proper scoring rules, a family of special cost functions of the form $C^*(q, h)$, which evaluates the goodness of the recognizer output q , for a trial where hypothesis h is true. We use the notation $h = \text{tar}$ for a trial where the target speaker spoke and $h = \text{non}$ for one where some other speaker spoke. In what follows it will be convenient to work with recognizer outputs in the form of posterior probabilities: $q = P(\text{tar}|s, \mathcal{M})$, where s is the uncalibrated trial score and \mathcal{M} is the calibration model. Later we show how to adapt this to recognizer outputs in likelihood-ratio form.

2.1. Definition

A binary proper scoring can be seen as a *model of an application* of a detector—or more generally, a mixture of such applications. An application, a , is represented by a cost function $C_a(d, h)$, which maps outcomes to real-valued consequences. The outcome is composed of the decision, $d \in \mathcal{D}_a$, and the true hypothesis, $h \in \{\text{tar}, \text{non}\}$. The sets from which decisions are chosen may differ between applications, for example $\mathcal{D}_a = \{\text{accept}, \text{reject}\}$, or $\mathcal{D}_a = \{\text{accept}, \text{reject}, \text{undecided}\}$, and so on. Our application model assumes that the detector output, q , is used to make a minimum-expected-cost Bayes decision:¹

$$d_a^*(q) \in \arg \min_d q C_a(d, \text{tar}) + (1 - q) C_a(d, \text{non})$$

The associated *proper scoring rule* is defined as the cost of this decision [7]:

$$C_a^*(q, h) = C_a(d_a^*(q), h) \quad (1)$$

A convex combination of proper scoring rules is still a proper scoring rule, in the sense that it can be derived via (1) from a suitably constructed cost function that represents a mixture of different applications [8].

2.2. Canonical form

Although applications may be defined via a large variety of cost functions, all of this variety can be conveniently represented in a surprisingly simple form [8, 5, 9]. All binary proper scoring

¹We assume there is a rule—the details of which are unimportant here—to choose among multiple minimizers.

rules can be expressed in the form:²

$$\begin{aligned} C_w^*(q, \mathbf{tar}) &= k_0 \int_{\log \frac{q}{1-q}}^{\infty} (1 + e^{-t}) w(t) dt + k_1 \\ C_w^*(q, \mathbf{non}) &= k_0 \int_{-\infty}^{\log \frac{q}{1-q}} (1 + e^t) w(t) dt + k_2 \end{aligned} \quad (2)$$

where $k_0 > 0$, $w(t) \geq 0$ and $\int_0^1 w(t) dt = 1$. The constants k_0, k_1, k_2 don't play any useful role in the Bayes decision framework and may be set to $k_0 = 1$ and $k_1 = k_2 = 0$ without loss of generality [7].

The weighting distribution $w(t)$ can be very general, including smooth functions, step functions or even impulses. The impulse $w(t) = \delta(t - \theta)$ represents a single, *simple application*, with binary decisions in $\mathcal{D}_a = \{\mathbf{accept}, \mathbf{reject}\}$, a cost function with penalties $C_{\text{miss}} = 1 + e^{-\theta}$ and $C_{\text{fa}} = 1 + e^{\theta}$ and the Bayes decision threshold at $\log \frac{q}{1-q} = \theta$. A sum of impulses represents a discrete mixture of applications, while a smooth function represents a continuous mixture over a continuous spectrum of applications. The important point here is that all Bayes decision applications (or mixtures of applications) can be represented by (2) via a suitable choice of $w(t)$.

3. Objective function

Here we build an objective function, for training or evaluation, out of a proper scoring rule, C_w^* . The proper scoring rule models the cost of making a Bayes decision in a single detection trial. To turn this into an objective function, we take the *expected* cost over a whole supervised database. We form this expectation via two hypothesis conditional averages, weighted by a synthetic class prior (usually different from the database proportions), of the form $\pi = P(\mathbf{tar})$, $1 - \pi = P(\mathbf{non})$. The expected cost is:

$$\bar{C}_w^\pi = \frac{\pi}{T} \sum_{i \in \mathcal{T}} C_w^*(q_i, \mathbf{tar}) + \frac{1 - \pi}{N} \sum_{i \in \mathcal{N}} C_w^*(q_i, \mathbf{non}) \quad (3)$$

where q_i is the recognizer's posterior for trial i ; \mathcal{T} is a set of T target trial indices; and \mathcal{N} a set of N non-target indices.

3.1. Scoring likelihood-ratios

Now, we relieve the calibrator of the (implicit) responsibility to have a hypothesis prior and to produce posteriors. Instead, we require it to output *log-likelihood-ratios*, denoted ℓ_i . Since the prior is already fixed as a parameter of the objective (3), the recognizer's posterior is given by Bayes' rule as:³

$$q_i = \sigma(\ell_i + \tau) \quad (4)$$

where τ is prior log odds and σ is the logistic sigmoid:

$$\tau = \log \frac{\pi}{1 - \pi}, \quad \text{and} \quad \sigma(x) = \frac{1}{1 + e^{-x}} \quad (5)$$

²Regularity conditions apply to the cost functions. We have adapted the representation in [8], section 7.4.1, via a transformation to log odds domain.

³In our discriminative calibration framework, defining ℓ_i as the ratio of two generative likelihoods adds no value. Instead (4) should be taken as the definition of ℓ_i .

After some manipulation, we can express the functions of q_i in (3) as functions of ℓ_i instead:

$$\begin{aligned} \pi C_w^*(q_i, \mathbf{tar}) &= \int_{\ell_i}^{\infty} (1 + e^{-t}) w_\pi(t) dt \\ (1 - \pi) C_w^*(q_i, \mathbf{non}) &= \int_{-\infty}^{\ell_i} (1 + e^t) w_\pi(t) dt \end{aligned} \quad (6)$$

where $w_\pi(t) = r_\tau(t) w(t + \tau)$ is a translated and modulated version of $w(t)$, where the modulation factor:

$$r_\tau(t) = \frac{1 + e^{t+\tau}}{(1 + e^t)(1 + e^\tau)} \quad (7)$$

is a raised and scaled sigmoid. If $w(t)$ is normalized, then $w_\pi(t)$ is normalizable in the sense: $0 < \int_{-\infty}^{\infty} w_\pi(t) dt < 1$.

3.2. Averaging over π

The integrals in (2) can be interpreted as averaging over all possible cost functions, where $w(t)$ determines the relative importance of different cost functions. In our construction of the objective function (3), we inherit this averaging over cost functions, but we seem to have fixed the prior at π . Since different applications of a speaker detector can be expected to have different priors, should we not also average over π ? We can, but it adds no generality, because this would effectively just replace $w_\pi(t)$ in (6) by some other normalized/normalizable weighting distribution, say $\Omega(t)$. We can now reformulate our objective function in its most general form as:

$$\bar{C}_\Omega = \frac{1}{T} \sum_{i \in \mathcal{T}} C_\Omega(\ell_i, \mathbf{tar}) + \frac{1}{N} \sum_{i \in \mathcal{N}} C_\Omega(\ell_i, \mathbf{non}) \quad (8)$$

where

$$\begin{aligned} C_\Omega(\ell_i, \mathbf{tar}) &= \int_{\ell_i}^{\infty} (1 + e^{-t}) \Omega(t) dt \\ C_\Omega(\ell_i, \mathbf{non}) &= \int_{-\infty}^{\ell_i} (1 + e^t) \Omega(t) dt \end{aligned} \quad (9)$$

We now have a general objective function, parametrized by $\Omega(t)$, which effectively determines the relative weighting over a mixture of different applications, each of which can have a different cost function and target prior. Notice that t is just the log-likelihood-ratio threshold. Simple *accept/reject* applications use a single threshold, while more complex applications (or mixtures) have multiple thresholds. Thus, the combination of application costs in (9) is accomplished via representing applications by their thresholds.

4. Practical recipe

Here we introduce a practical choice for the weighting distribution, $\Omega(t)$ and show how it is used for calibration. Since calibration involves optimization, smooth differentiable distributions are easier to work with. It is also desirable that the integrals have closed-form solutions. To this end, we adopt a 2-parameter (here α, β) family of proper scoring rules proposed in [9]. This family is effectively augmented by a third parameter, $\tau = \log \frac{\pi}{1 - \pi}$, via the prior-weighting in (3). This gives $\Omega = \Omega_{\alpha, \beta, \tau}$:

$$\Omega_{\alpha, \beta, \tau}(t) = \frac{r_\tau(t) w_{\alpha, \beta}(t + \tau)}{Z_{\alpha, \beta, \tau}} \quad (10)$$

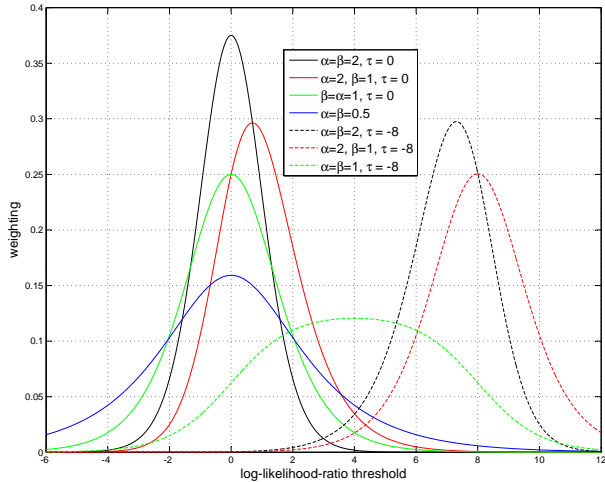


Figure 1: Threshold weightings, $\Omega_{\alpha, \beta, \tau}(t)$, of a few objective function parametrizations. (Solid green represents C_{lr} , see main text.)

where $\alpha, \beta > 0$; r_τ is defined in (7); $Z_{\alpha, \beta, \tau}$ ensures the distribution is normalized; and $w_{\alpha, \beta}$ is the beta distribution, transformed to log-odds domain:

$$w_{\alpha, \beta}(t) = \frac{\sigma(t)^\alpha \sigma(-t)^\beta}{B(\alpha, \beta)} \quad (11)$$

and $B(\alpha, \beta)$ is the beta function. When $\alpha, \beta \in \{\frac{1}{2}, 1, 1\frac{1}{2}, 2, \dots\}$, then using $w = w_{\alpha, \beta}$ in (2) gives closed-form solutions, which can be plugged into (3), with $\pi = \sigma(\tau)$ and $q_i = \sigma(\ell_i + \tau)$.

4.1. Interpretation

While this recipe is practically realized via (2) and (3), a theoretical interpretation is given by (10) and (8), because for this parametrization we have $\bar{C}_w^\pi = Z_{\alpha, \beta, \tau} \bar{C}_\Omega$. We exploit this in figure 1, where we plot (10) for several values of the parameters. Parameters α, β control left and right tails respectively—when they are increased tails become thinner and the distributions become narrower (higher modes). Location is indirectly controlled by τ , but this depends on α, β . For the case $\alpha = \beta = \frac{1}{2}$, the distribution is *invariant* to τ . For $\alpha = \beta = 1$, the mode shifts only as far as $-\frac{\tau}{2}$, while the distribution is considerably widened. However, when $\alpha = 2$, the mode shifts close to $-\tau$ and the widening is less dramatic. We show in our experiments below that when we target applications with high thresholds (low false-alarm rate), then the effective shifting allowed by $\alpha = 2$ leads to better performance than the halfway shifted and stretched version given by the baseline logistic regression solution with $\alpha = \beta = 1$.

4.2. Examples

Here we list some solutions for the integrals in (2), which we used in our experiments. The *boosting rule* [9], with $\alpha = \beta = \frac{1}{2}$:

$$\begin{aligned} C_{\frac{1}{2}, \frac{1}{2}}^*(q, \text{tar}) &= \frac{2}{\pi} \sqrt{\frac{1-q}{q}} \\ C_{\frac{1}{2}, \frac{1}{2}}^*(q, \text{non}) &= \frac{2}{\pi} \sqrt{\frac{q}{1-q}} \end{aligned} \quad (12)$$

The *logarithmic rule*, with $\alpha = \beta = 1$, forms the objective function for logistic regression and is our baseline:

$$\begin{aligned} C_{1,1}^*(q, \text{tar}) &= -\log(q) \\ C_{1,1}^*(q, \text{non}) &= -\log(1-q) \end{aligned} \quad (13)$$

The parametrization $\alpha = \beta = 1, \tau = 0$ gives (up to scaling), the objective C_{lr} , proposed in [2] for the evaluation of goodness of recognizers with likelihood-ratio output. The *Brier rule* [10], with $\alpha = \beta = 2$:

$$\begin{aligned} C_{2,2}^*(q, \text{tar}) &= 3(1-q)^2 \\ C_{2,2}^*(q, \text{non}) &= 3q^2 \end{aligned} \quad (14)$$

An *asymmetric rule*, with $\alpha = 2, \beta = 1$:

$$\begin{aligned} C_{2,1}^*(q, \text{tar}) &= 2(1-q) \\ C_{2,1}^*(q, \text{non}) &= -2\log(1-q) - 2q \end{aligned} \quad (15)$$

4.3. Calibration recipe

We assume that our recognizers output uncalibrated scores. Let the score for a trial i be denoted s_i . We then apply a parametric (affine) calibration transformation:

$$\ell_i = A s_i + B \quad (16)$$

The calibration parameters A, B need to be trained on a calibration training database. For training purposes, we choose and fix the parameters α, β, τ and then minimize the objective (3) w.r.t. A, B . In the special case $\alpha = \beta = 1$ such training is known as logistic regression. Since the objective and the calibration transformation are differentiable, one can obtain the gradient w.r.t. A, B by backpropagation and then use any of a variety of well-known unconstrained numerical optimization algorithms. For this work, we used BFGS [11].

5. Experiments

We performed calibration experiments on scores from a single speaker recognizer (an i-vector PLDA system), which was part of the ABC submission [12] to the NIST SRE'12 speaker recognition evaluation. Several calibration transformations of the form (16) were trained (separately for males and females) on a large development set with multiple microphone and telephone speech segments of 2019 speakers from SRE'04, '05, '06, '08 and '10. This gave about 120 million scores for calibration training. The different calibration transformations were obtained by using different objective function parameters, α, β, τ .

We tested the goodness of these calibrators on the NIST SRE'12 extended trial set [13], where we pooled males and females and all 5 common evaluation conditions, giving about 80 million trials.

Our evaluation criterion was C_{primary} , with $P_{\text{known}} = 0$, as defined in [13]. It should be noted that this criterion can *also* be interpreted as an example of our objective function (8), since $C_{\text{primary}} = \bar{C}_\Omega$, if we choose $\Omega = \Omega_{\text{primary}}$, defined as:

$$\Omega_{\text{primary}}(t) = \frac{1}{2} \sum_{i=1}^2 \sigma(\theta_i) \delta(t - \theta_i) \quad (17)$$

which is concentrated in two impulses⁴ at the log-likelihood-ratio threshold values of $\theta_1 = 4.59$ and $\theta_2 = 6.91$. This criterion was chosen by NIST to be calibration-sensitive, but only in the 'low false-alarm region' around these operating points.

⁴with *almost* equal weights, since $\sigma(\theta_i) \approx 1$

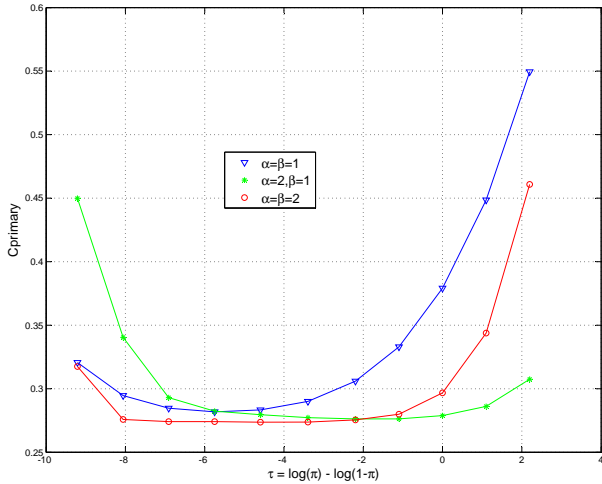


Figure 2: SRE'12: Comparison between proper scoring rules for calibration. More negative τ places more weight at more positive score thresholds. (Blue at $\tau = 0$ represents C_{lr} .)

Our experimental results are summarized in figure 2, where we plotted C_{primary} obtained by using the calibrators formed by three representative proper scoring rules against the prior-weighting parameter, $\tau = \log \frac{\pi}{1-\pi}$. Our baseline (blue triangles) is logistic regression ($\alpha = \beta = 1$). The other two rules with $\alpha = 2$ (red circles, green asterisks), gave better performance in the sense of having lower, wider minima.

The boosting rule ($\alpha = \beta = \frac{1}{2}$) performed very poorly, at $C_{\text{primary}} = 0.65$, and is not shown on the graph. Its thick tails makes it vulnerable to outliers.

6. Discussion

The reason why different proper scoring rules perform differently is that the affine calibration transformation is limiting. It cannot satisfy the requirement of good calibration at all operating points simultaneously. We have to choose, via $\Omega(t)$, which operating points are important and which can be ignored. This is illustrated in figure 3, where we compare the log-likelihood-ratios produced for SRE'12 by two of the proper scoring rule calibrations (horizontal axis), against ideal, reference log-likelihood-ratios (vertical axis). Ideally these plots should lie along the identity transform (the black diagonal through the origin). But the affine calibration transformation can only try to fit these curves to the ideal straight line via scaling and shifting the curves along the horizontal axis. The logistic regression ($\alpha = \beta = 1$) places more weight at lower threshold values and indeed manages to be closer to the ideal there, to the detriment of calibration at the higher C_{primary} operating points. On the other hand, the Brier rule ($\alpha = \beta = 2$), which places more weight at the C_{primary} operating points, does better there, but pays for this by doing worse at lower threshold values.

6.1. The PAV reference

The ideal reference along the vertical axis of figure 3 is achieved via an optimization algorithm known as *pool adjacent violators* (PAV)[14, 8]. This algorithm finds the optimal log-likelihood-ratio value for every trial non-parametrically, subject only to the constraint that when sorted along the real line, the order of the

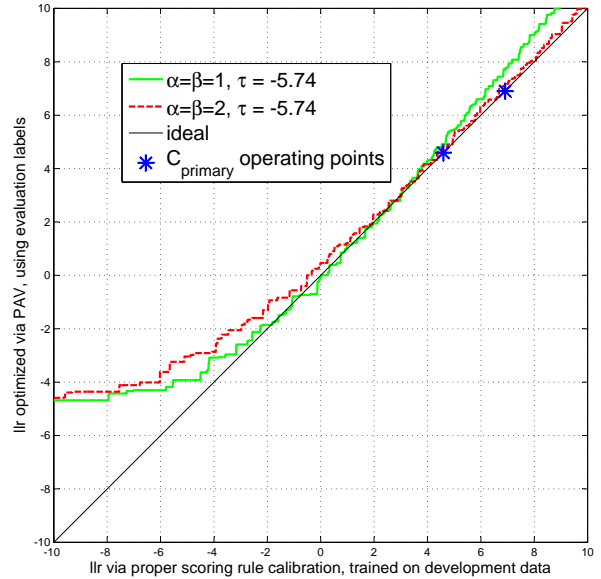


Figure 3: SRE'12: PAV reference vs proper scoring rule calibration

optimized values must be the same as the sorted order of the original uncalibrated scores. This is equivalent to constraining the calibration transformation to be monotonically rising. The PAV output is optimal in the sense that it simultaneously optimizes all binary proper scoring rules at all values of the prior weighting [8]. The PAV solution can be optimal everywhere because the non-parametric monotonicity constraint is less strict than the parametric affine transformation available to the proper scoring rule calibrators. It is important to note however that the true SRE'12 class labels were used to perform the PAV optimization, while the proper scoring rule calibrations were optimized on the independent development data. If the PAV had been optimized on the development data, it would no longer have this optimality on the evaluation data.⁵

6.2. Caveat

Conclusions from our experimental results should not be extrapolated without caution. The superiority demonstrated here of rules with $\alpha > 1$ may not hold for: (i) small calibration training databases; (ii) other recognizers; (iii) applications with thresholds in other operating regions.

6.3. Generalization

The objective function family presented here could be more generally applied, not just for calibration. These objectives could be used for fusion of multiple recognizers, or indeed for more general discriminative training of speaker recognizers in the manner of [15]. However for more complex recognizers, the risk of overtraining is greater—and this risk may be compounded by more narrowly focussed objectives, such as the Brier rule. In contrast, the wider focus of the logarithmic rule has a regularizing effect, which combats overtraining.

⁵We *did* interpolate the PAV transform of the development scores to form a calibrator for the evaluation data. This gave C_{primary} very similar to the best proper scoring rule results shown in figure 2. The details are out of scope for this paper.

7. References

- [1] Niko Brümmer et al. “Fusion of Heterogeneous Speaker Recognition Systems in the STBU Submission for the NIST Speaker Recognition Evaluation 2006”, *IEEE Transactions on Audio, Speech, and Language Processing*, September 2007.
- [2] Niko Brümmer and Johan du Preez, “Application-Independent Evaluation of Speaker Detection”, *Computer Speech and Language*, 2006.
- [3] Niko Brümmer and Edward de Villiers, “The BOSARIS Toolkit: Theory, Algorithms and Code for Surviving the New DCF”, NIST SRE’11 Analysis Workshop, Atlanta, December 2011. Online: <http://sites.google.com/site/bosaristoolkit>.
- [4] George Doddington, “The Role of Score Calibration in Speaker Recognition”, *Interspeech*, Portland, September 2012.
- [5] Tilmann Gneiting and Adrian E. Raftery, “Strictly proper scoring rules, prediction, and estimation”, *Journal of the American Statistical Association*, vol. 102, no. 477, pp. 359–378, March 2007.
- [6] Niko Brümmer, “The Role of Proper Scoring Rules in Training and Evaluating Probabilistic Speaker and Language Recognizers”, Invited Talk, *Odyssey 2012: The Speaker and Language Recognition Workshop*, Singapore, 2012. Online: <http://www.odyssey2012.org/plenary.html>.
- [7] Morris H. DeGroot, *Optimal Statistical Decisions*, McGraw-Hill, 1970.
- [8] Niko Brümmer, “Measuring, refining and calibrating speaker and language information extracted from speech”, Ph.D. dissertation, University of Stellenbosch, December 2010.
- [9] Andreas Buja, Werner Stuetzle, and Yi Shen, “Loss functions for binary class probability estimation and classification: Structure and applications”, Technical report, Statistics Department, The Wharton School, University of Pennsylvania, November 2005.
- [10] Glenn W. Brier, “Verification of Forecasts Expressed in Terms of Probability”, *Monthly Weather Review*, vol.78, no.1, pp. 1–3, January 1950.
- [11] Jorge Nocedal and Stephen J. Wright, *Numerical Optimization*, 2nd edition, Springer 2006.
- [12] AGNITIO, BUT, CRIM, “ABC SRE’12 Presentation”, NIST SRE 2012 Workshop, Orlando, December 2012.
- [13] “The NIST Year 2012 Speaker Recognition Evaluation Plan” 2012. Online: http://www.nist.gov/itl/iad/mig/upload/NIST_SRE12_evalplan-v17-rl.pdf.
- [14] Bianca Zadrozny and Charles Elkan, “Transforming classifier scores into accurate multiclass probability estimates”, in *Proceedings of the Eighth International Conference on Knowledge Discovery and Data Mining*, Edmonton, Alberta, Canada, 2002, pp. 694–699.
- [15] Burget, L. et al., “Discriminatively Trained Probabilistic Linear Discriminant Analysis for Speaker Verification”, *IEEE ICASSP*, Prague, 2011.