Pattern Recognition 47 (2014) 1509–1522

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Exploiting the relationships among several binary classifiers via data transformation

Kar-Ann Toh [a,*], Geok-Choo Tan [b]

[a] School of Electrical & Electronic Engineering, Yonsei University, Seoul 120-749, Korea
[b] Division of Mathematical Sciences, School of Physical & Mathematical Sciences, Nanyang Technological University, Singapore 639798, Singapore

ARTICLE INFO

Article history:
Received 16 May 2013
Received in revised form
18 September 2013
Accepted 25 September 2013
Available online 14 October 2013

Keywords:
Binary classification
Linear estimation methods
Deterministic methods
Area under the ROC curve

ABSTRACT

The structural resemblance among several existing classifiers has motivated us to investigate their underlying relationships. By exploring into the mapping solutions of these classifiers, we found that they can be linked by simple feature data scaling. In other words, the key to these relationships lies upon how the replica of feature data are being scaled. This finding leads us directly to an exploration of novel classifiers beyond existing settings. Based on an extensive empirical evaluation, we show that the proposed formulation facilitates a tuning capability beyond existing settings for classifier generalization.

1. Introduction

Pattern classification plays an important role in many decision processes such as biometric authentication, medical diagnosis, fault diagnosis, financial forecast, and data mining [13,20]. Although the field has been widely explored with many useful classification methods available, the quest towards a good generalization or predictivity performance given the limited amount of training data remains a topic of interest [8,42]. Among those existing methods which address the issue of classifier generalization, methods based on the *receiver operating characteristic* (ROC) performance has received considerable attention over recent years. A key reason to this attention could be attributed to the provision of an overview interpretation over all possible decision threshold values by the ROC curves. It has been argued that when the decision threshold and related decision parameters are not well defined, the ROC offers a relevant tool for assessment of the overall performance of a classifier (see e.g., [11,14,27,32]). While the ROC provides a range of performance values [1], the single valued area under the ROC curve (AUC) becomes a natural choice for overall classifier assessment [6,10]. It follows from [18] that AUC is linearly related to the average of the Bayes correct rates over all possible values of the classification threshold. This observation, perhaps, explains to a certain extend the relatively good generalization property of AUC based methods.

Capitalized on a linear parametric model with normalization of data and matching of a link-loss functional pair in optimization, the quadratic function has recently been shown to provide a relevant approximation to the step loss function in AUC optimization [41]. Particularly, the solution of the classifier's parameters with respect to AUC optimization can be expressed in closed-form. In view of the structural resemblance of this AUC solution with several well-known classifiers, this work investigates into the underlying data mapping of these classifiers and finds that they are related by simple feature data scaling. This finding leads us directly to an exploration of novel classifiers which adopt different scaling factors beyond existing frameworks.

We shall capitalize on the optimization tractability advantage of linear parametric models in developing the relationships among these classifiers. While noting that embedding of nonlinearities such as kernels and basis functions into linear regression models can widen the scope of applications, we shall explore a random projection network which has been shown to possess universal approximation capability [23] in this paper.

The main contributions of this paper are as follows: (i) Establishment of relationships among several binary classifiers via a framework of scaling and translational space thereby gaining insights into classifier learning from data perspective. (ii) Exploration of a random projection network for such classifier learning. (iii) Proposal of novel classifiers which exploit the established relationships. (iv) Extensive experiments to observe the impact of these classifiers on several performance measures. Since the projection network can be constructed using various activation functions and hidden layer structures such as echoed hidden

[*] Corresponding author. Tel.: + 82 2 2123 5864; fax: +82 2 312 4584.
E-mail addresses: katoh@ieee.org (K.-A. Toh), gctan@ntu.edu.sg (G.-C. Tan).

0031-3203/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.patcog.2013.09.030

neurons [24], this study serves as a benchmark for further generalization related explorations.

The paper is organized as follows. In the following Section 2, a brief review of classification based linear estimation methods is presented with several commonly adopted performance measures. Subsequently in Section 3, a data transformation perspective is presented and this view has been utilized to link up several binary classifiers. Capitalized on the established relationships, Section 4 proposes a set of novel classifiers for our experimentation. Section 5 shows the results of our empirical study and presents our observations. Finally some concluding remarks are given in Section 6.

2. Preliminaries

2.1. Prediction and classification

Given a learning set consisting of m examples $(\boldsymbol{x}_i, y_i)$, $i = 1, \ldots, m$, where $\boldsymbol{x}_i \in \mathbb{R}^d$ denotes the *ith* feature sample, and $y_i \in \{0, 1\}$ denotes the corresponding indicator or target label. The value y_i can be viewed as the class associated with $\boldsymbol{x}_i$. Based on the given feature sample as input, a learned predictor outputs a corresponding value related to target prediction.

For binary classification, our goal is to determine a predictor g and a threshold τ such that a correct class prediction can be obtained. An ideal classifier is such that $L(g(\boldsymbol{x}_i)) = y_i$ (see (1)) for all $i = 1, 2, \ldots, m$, which relates each $\boldsymbol{x}_i$ to its target label y_i. In this paper, we consider the predictor g to be a linear projection model. We shall determine a suitable $\hat{g}$, based on the given m examples, via some learning criteria such as the sum of squared errors, total error rate and the area under the ROC curve.

With this predictor $\hat{g}$, which can be normalized to within $[0, 1]$, we then seek to find a threshold τ which is a number between 0 and 1 and use it to determine the class of an unseen test data $\boldsymbol{x}_u$ in the following way:

$$L(\hat{g}(\boldsymbol{x}_u)) = \begin{cases} 1, & \hat{g}(\boldsymbol{x}_u) \geq \tau, \\ 0 & \text{else}. \end{cases} \tag{1}$$

2.2. Linear projection model

Suppose we have a projection model for predictor g which can be expressed in linear parametric form given by

$$g(\boldsymbol{\alpha}, \boldsymbol{x}) = \sum_{j=1}^{D} \alpha_j p_j(\boldsymbol{x}) = \boldsymbol{p}(\boldsymbol{x})^T \boldsymbol{\alpha}, \tag{2}$$

where each term $p_j(\boldsymbol{x})$ is an element of the column vector $\boldsymbol{p}(\boldsymbol{x})$ that maps the input vector $\boldsymbol{x} \in \mathbb{R}^d$ into a feature space $\mathbb{R}^D$ (for example, $p_j(\boldsymbol{x})$ can be a random projection model or a multivariate polynomial of some fixed order), and $\boldsymbol{\alpha} \in \mathbb{R}^D$ corresponds to a vector of weighting coefficients to be estimated. Here, we note that by incorporating a nonlinear mapping of $\boldsymbol{p} : \mathbb{R}^d \rightarrow \mathbb{R}^D$, the linear parametric model extends its capability to map nonlinear input-output spaces.

A good example of linear projection model is the random projection network given by

$$g(\boldsymbol{\alpha}, \boldsymbol{x}) = [\phi(\mathbf{w}_1^T \boldsymbol{x} + b_1), \ldots, \phi(\mathbf{w}_D^T \boldsymbol{x} + b_D)]\boldsymbol{\alpha} = \boldsymbol{\phi}(\mathbf{W}\boldsymbol{x} + \boldsymbol{b})^T \boldsymbol{\alpha}, \tag{3}$$

where ϕ is a nonlinear activation such as a sigmoid function, $\mathbf{W} \in \mathbb{R}^{D \times d}$ is a random weight matrix, $\boldsymbol{\alpha} \in \mathbb{R}^D$ is the parameter to be estimated, and $\boldsymbol{b} \in \mathbb{R}^D$ is a bias term. This random projection network is analogous to the single-hidden-layer feedforward network as seen in [22] except that we consider only a single output neuron in this work for binary classification applications.

Table 1
Some performance measures for binary classifiers.

Performance	Definition	$s=1$	$s=m^-/m^+$
True Positive Rate (Recall)	$tpr = \frac{tp}{m^+}$	–	–
True Negative Rate (specificity)	$tnr = \frac{tn}{m^-} = (1 - fpr)$	–	–
Accuracy	$\frac{tpr + s(1-fpr)}{1+s}$	$\frac{tpr+1-fpr}{2}$	$\frac{tp+tn}{m^+ + m^-}$
Precision	$\frac{tpr}{tpr + s \cdot fpr}$	$\frac{tpr}{tpr+fpr}$	$\frac{tp}{tp+fp}$
F-measure	$\frac{2tpr}{tpr + s \cdot fpr + 1}$	$\frac{2tpr}{tpr+fpr+1}$	$\frac{2tp}{tp+fp+m^+}$
AUC	$\frac{1}{m^+ m^-} \sum_{i=1}^{m^+} \sum_{j=1}^{m^-} u(\xi_{ij})$	–	–

Remark: see Section 2.4.3 for detailed definition of AUC.

It has been shown in [23] that a random projection network with arbitrarily number of hidden neurons and non-constant activation function possesses both approximation and classification capability. In view of its wide application potential, we shall adopt this random projection network ($g(\boldsymbol{\alpha}, \boldsymbol{x}) = \boldsymbol{p}(\boldsymbol{x})^T \boldsymbol{\alpha}$ where $\boldsymbol{p}(\boldsymbol{x}) = \boldsymbol{\phi}(\mathbf{W}\boldsymbol{x} + \boldsymbol{b})$) with a sigmoid function for $\boldsymbol{\phi}$.

2.3. Some common performance measures

In this subsection, we include a table of common performance measures related to our classification task. To simplify our notation, we shall indicate a superscript + or − on the variable $\boldsymbol{x}$ according to whether it belongs to class-1 (positive class) and class-0 (negative class), respectively. Among the m learning samples, suppose the samples $\boldsymbol{x}_i^+, i = 1, 2, \ldots, m^+$, belong to the positive class while samples $\boldsymbol{x}_j^-, j = 1, 2, \ldots, m^-$, belong to the negative class. In other words, m^+ is the number of $\boldsymbol{x}_i$ with $y_i = 1$ (ground truth for size of positive class) and m^- is the number of $\boldsymbol{x}_i$ with $y_i = 0$ (ground truth for size of negative class).

Suppose a predictor g is chosen, we can vary the threshold τ to classify the samples into positive class or negative class. Based on this set of learning examples, we calculate the following numbers for each threshold τ in (1):

$$\begin{aligned} tp &= \text{'number of } \boldsymbol{x}_i \text{ with } y_i = 1 \text{ and } L(g(\boldsymbol{x}_i)) = 1\text{' (true positive)} \\ fp &= \text{'number of } \boldsymbol{x}_i \text{ with } y_i = 0 \text{ but } L(g(\boldsymbol{x}_i)) = 1\text{' (false positive)} \\ fn &= \text{'number of } \boldsymbol{x}_i \text{ with } y_i = 1 \text{ but } L(g(\boldsymbol{x}_i)) = 0\text{' (false negative)} \\ tn &= \text{'number of } \boldsymbol{x}_i \text{ with } y_i = 0 \text{ and } L(g(\boldsymbol{x}_i)) = 0\text{' (true negative)} \end{aligned}$$

and compute the *true positive rate* (*tpr* or TPR) given by $tpr = tp/m^+$ which reflects the fraction of correctly classified learning examples from the positive class, and the *false positive rate* (*fpr* or FPR) given by $fpr = fp/m^-$ which reflects the fraction of wrongly classified learning examples from the negative class.

The above rates define several frequently used performance measures [15,28,30,33] such as those given in Table 1, where s is a skewing factor which is frequently taken to be the ratio between the size of negative examples and the size of positive examples (i.e., $s = m^-/m^+$). The true positive rate (tpr) is also termed *recall* and the true negative rate (tnr) is also called *specificity*. Recall can be treated as a measure of completeness while precision can be treated as a measure of fidelity. In documents retrieval, precision is the fraction of the documents retrieved that are relevant to the user's information need and recall is the fraction of the documents that are relevant to the query being successfully retrieved. The *F*-measure is the harmonic mean of precision and recall.

The tpr defined above can be plotted over fpr at different decision threshold values (τ) over a range, say from 0 to 1 in a