Pattern Recognition 47 (2014) 3949–3958

Contents lists available at ScienceDirect

Pattern Recognition

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

Toward maximum-predictive-value classification

Eric Chalmers [a,*], Marcin Mizianty [a,b,c], Eric Parent [b], Yan Yuan [c], Edmond Lou [a]

[a] Electrical & Computer Engineering Department, University of Alberta, 9107 – 116 St, Edmonton, Alberta, Canada T6G 2V4
[b] Department of Physical Therapy, University of Alberta, 8205 – 114 St, Edmonton, Alberta, Canada T6G 2G4
[c] School of Public Health, University of Alberta, 11405 – 87 Ave, Edmonton, Alberta, Canada T6G 1C9

ARTICLE INFO

Article history:
Received 15 January 2014
Received in revised form
12 June 2014
Accepted 15 June 2014
Available online 24 June 2014

Keywords:
Classification
Nearest prototype
Precision
Predictive value

ABSTRACT

Methods for tackling classification problems usually maximize prediction accuracy. However some applications require maximum predictive value instead. That is, the designer hopes to predict one of the classes with maximum precision, and is less concerned about the others. Some techniques exist for fine-tuning a model's predictive value, but there seems to be a shortage of methods to generate maximum-predictive-value classifiers. We propose a method using a nearest-prototype-style classifier optimized by a genetic algorithm. We test its performance using 13 publicly available data sets from the life sciences. The method generally gives more effective high-predictive-value models than standard classification methods optimized for predictive value.

1. Introduction

1.1. Accuracy, predictive value, and true positive rate

In most classification problems, the goal is to generate class predictions with the highest possible accuracy. Designers develop models to predict class, and would like to be reasonably sure that those predictions are correct – whatever class they may be. There are a variety of learning methods available for training high-accuracy models (e.g. generalized linear model methods, decision trees, support vector machines) and data science provides some good guidelines and techniques for the successful application of these methods.

However in some cases high accuracy (defined as the fraction of all predictions which are correct) is not actually the goal. In some applications, the top priority may be to ensure that a certain class prediction is nearly never wrong. The example application which inspired this work is in an adolescent idiopathic scoliosis clinic. Idiopathic scoliosis is a spinal deformity affecting 2–3% of adolescents [1]. Patients are monitored closely (by x-ray examination) during their growth to check for "progression" (worsening) of the deformity. Most of these examinations show no change in the deformity since the previous examination – which is frustrating, because the x-ray exposed the child to harmful radiation but revealed no new information. Cumulative x-ray exposure is associated with increased cancer risk for these children [2], and a model to identify these "unchanged" cases using non-radiographic features could reduce unnecessary radiation exposure.

This situation presents an interesting classification problem. Since the current practice is to x-ray all patients at every visit, the number of unnecessary x-rays which the model can identify is not the major design consideration: any number of saved x-rays would be an improvement on the current situation, and this improvement would come at almost no cost if the model uses already-collected measurements as inputs. Instead the major consideration is the model's precision: when it predicts an "unchanged" case, it must (nearly) never be wrong. This is because omitting an x-ray when a change has occurred would mean a missed treatment opportunity. Physicians would only accept the model if its "unchanged" predictions could be trusted completely. In statistical terms, the model must have a high (100%, if possible) predictive value in predicting "unchanged" cases. The sensitivity to these cases should also be maximized, but only as a secondary objective to predictive value.

Standard modeling approaches are ineffective in this situation, as they generally seek to maximize predictive accuracy (by minimizing error probability, maximizing likelihood functions, etc.) This is inappropriate in situations like our scoliosis example. Instead the modeling process must sacrifice accuracy for predictive value. Maximizing predictive value can be difficult, and there seems to be a shortage of tools for doing it.

In this paper we will consider a requirement for maximum positive predictive value (PPV), though the -discussion applies equally to maximizing negative predictive value. The PPV (also called "precision")

* Correspondence to: 10105 – 112 Ave, Edmonton, Alberta, Canada T5G 0H1. Tel.: +1 780 337 4242; fax: +1 780 735 7972.
E-mail addresses: dchalmer@ualberta.ca (E. Chalmers), mizianty@ualberta.ca (M. Mizianty), eparent@ualberta.ca (E. Parent), yyuan@ualberta.ca (Y. Yuan), elou@ualberta.ca (E. Lou).

http://dx.doi.org/10.1016/j.patcog.2014.06.013
0031-3203/

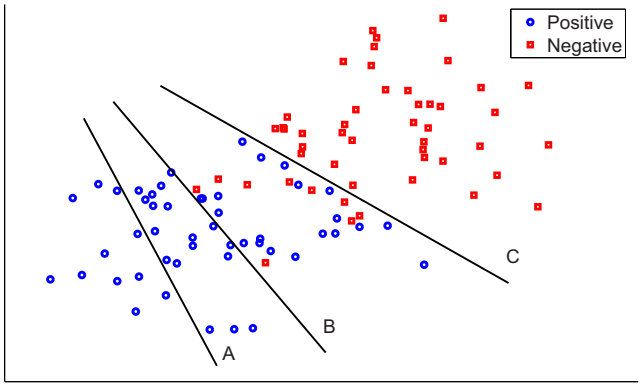

Fig. 1. Hypothetical distribution of positive and negative instances. Each of the three decision boundaries classifies points to its left as positive. Boundary A gives 100% PPV, boundary C gives 100% TPR, and boundary B gives maximum TPR while maintaining 100% PPV.

is the fraction of a model's "positive" predictions which are actually correct [3]:

$$\mathrm{PPV} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} = \frac{\text{True positives}}{\text{Predicted positives}} \tag{1}$$

Here TP is the number of true positive and FP the number of false positive predictions produced by the model. In the scoliosis example, a true positive could be an "unchanged" case which is correctly identified, while a false positive would be a "change" case erroneously predicted as "unchanged". A PPV of 100% is achieved by eliminating all false positives. But there is a trade-off between PPV and true positive rate (TPR). TPR (also called "sensitivity" or "recall") is the fraction of all positives which are identified by the model:

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} = \frac{\text{True positives}}{\text{All positives}} \tag{2}$$

FN is the number of false negatives produced by the model (i.e. the number of "unchanged" cases not identified as such). There is a trade-off between PPV and TPR: maximizing PPV means being conservative with our 'positive' predictions, which reduces false positives but also increases false negatives (to the detriment of TPR). While the primary objective pursued in this paper is maximum PPV, TPR must be considered as well. To use the Scoliosis example, a model which identifies no-change cases with complete confidence (100% PPV) would be impractical if it identified only 1% of these cases. Fig. 1 shows three hypothetical linear classifiers which illustrate the PPV/TPR trade-off: Classifier "*A*" gives 100% PPV, "*C*" gives 100% TPR, and "*B*" gives maximum TPR while maintaining 100% PPV.

1.2. Existing techniques for increasing PPV

When working with a single feature the designer could manage the PPV–TPR trade-off using receiver operating characteristic (ROC) curve analysis. Selecting the threshold giving maximum TPR at zero FPR on the ROC curve is equivalent to maximizing TPR while maintaining 100% PPV.

But how should high-PPV predictions be made using multiple features? One option is to use a standard multivariate modeling technique, along with a feature selection scheme which selects the subset of the available features giving the best-PPV model. Sahiner et al. used a genetic algorithm (GA) to select features which minimized a diagnostic model's FPR in the high-TPR region of the ROC curve [4] (the opposite of the goal discussed in this paper). This custom feature selection pushes the modeling process toward higher predictive value models, but the core modeling technique itself is generally still designed to maximize accuracy. In effect, this approach enumerates several high-accuracy models and chooses the one with the best PPV post hoc.

Another option is to apply a cost matrix **C**, which quantifies how undesirable misclassifications are. $\mathbf{C}(i,j)$ is the cost incurred by predicting class i when the true class is j. Depending on the application, this "cost" may refer to a monetary cost, lost time, or some other measurement of undesirability. The optimal prediction is the class c which minimizes expected cost: $\Sigma_j P(j|\mathbf{X})\mathbf{C}(c,j)$. Ling et al. proposed a new decision tree splitting criterion for building decision trees which minimize cost [5]. Pendharkar et al. [6] and Chen et al. [7] used GAs to minimize cost in artificial neural network classifiers and nearest-prototype classifiers respectively. Cost sensitivity can also be induced by weighting or resampling training examples proportional to misclassification costs [8–10], or working costs into a boosting scheme [11,12]. But in our application specifying a cost for false-positives is somewhat awkward: we simply want no false-positives. This desire would translate into a cost matrix with an astronomically large penalty for predicting positive when the true class is negative. Many learning techniques – in the process of minimizing expected cost – would avoid predicting positive at all.

Elkan suggests it is better to train a high-accuracy classifier, and then make high-predictive-value predictions by considering the probability estimates it provides [13]. For example, we could train a multivariate logistic regression model, which computes a probability-like score for a given observation. By default the observation is predicted positive if this score is greater than 0.5, but we could increase PPV by increasing this threshold (using the ROC curve analysis described above). Zaugg et al. used this threshold moving technique to increase PPV [14], while Koh [15] and Pendharkar [16] used it to reduce misclassification cost. Note this approach changes the position of the model's decision boundary, but not its orientation. In some cases optimal high-PPV classification may require a completely different decision boundary. Fig. 2 illustrates this problem using data from the "ecoli" dataset [17] (with the "cytoplasm" class labeled positive, and all others labeled negative). This figure shows the decision boundary created by a standard logistic regression (A), and the boundary's new location after increasing the decision threshold to maximize PPV (B). Boundary A has imperfect PPV because it erroneously classifies some negative instances as positives. Boundary B has maximum PPV, but has lower TPR than the optimal maximum-PPV decision boundary (C).

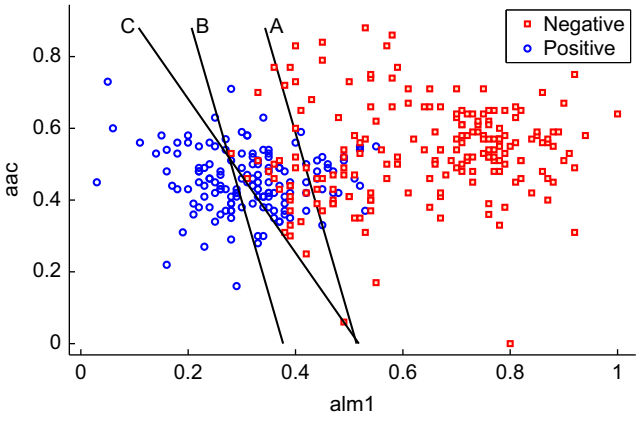

Fig. 2. A sample two-class classification problem taken from the "ecoli" dataset. A standard logistic regression produces decision boundary A. Increasing the decision threshold on class probability can maximize PPV by shifting the boundary to B, but can never produce optimal linear boundary C (the boundary with maximum TPR at 100% PPV).