

Safe-Level-SMOTE: Safe-Level-Synthetic Minority Over-Sampling TEchnique for Handling the Class Imbalanced Problem

Chumphol Bunkhumpornpat, Krung Sinapiromsaran, and Chidchanok Lursinsap

Department of Mathematics, Faculty of Science, Chulalongkorn University
chumphol@chiangmai.ac.th, Krung.S@chula.ac.th,
lchidcha@chula.ac.th

Abstract. The class imbalanced problem occurs in various disciplines when one of target classes has a tiny number of instances comparing to other classes. A typical classifier normally ignores or neglects to detect a minority class due to the small number of class instances. SMOTE is one of over-sampling techniques that remedies this situation. It generates minority instances within the overlapping regions. However, SMOTE randomly synthesizes the minority instances along a line joining a minority instance and its selected nearest neighbours, ignoring nearby majority instances. Our technique called Safe-Level-SMOTE carefully samples minority instances along the same line with different weight degree, called safe level. The safe level computes by using nearest neighbour minority instances. By synthesizing the minority instances more around larger safe level, we achieve a better accuracy performance than SMOTE and Borderline-SMOTE.

Keywords: Class Imbalanced Problem, Over-sampling, SMOTE, Safe Level.

1 Introduction

A dataset is considered to be imbalanced if one of target classes has a tiny number of instances comparing to other classes. In this paper, we consider only two-class case [5], [17]. The title of a smaller class is a minority class, and that of a bigger class is a majority class. The minority class includes a few positive instances, and the majority class includes a lot of negative instances.

In many real-world domains, analysts encounter many class imbalanced problems, such as the detection of unknown and known network intrusions [8], and the detection of oil spills in satellite radar images [13]. In these domains, standard classifiers need to accurately predict a minority class, which is important and rare, but the usual classifiers seldom predict this minority class.

Strategies for dealing with the class imbalanced problem can be grouped into two categories. One is to re-sample an original dataset [11], [14], [15], either by over-sampling a minority class or under-sampling a majority class until two classes are nearly balanced. The second is to use cost sensitive learning by assigning distinct costs to correctly classified instances or classifications errors [7], [9], [16].

Table 1. A confusion matrix for a two-class imbalanced problem

	Predicted Positive	Predicted Negative
Actual Positive	TP	FN
Actual Negative	FP	TN

The performance of classifiers is customarily evaluated by a confusion matrix as illustrated in Table 1. The rows of the table are the actual class label of an instance, and the columns of the table are the predicted class label of an instance. Typically, the class label of a minority class set as positive, and that of a majority class set as negative. *TP*, *FN*, *FP*, and *TN* are True Positive, False Negative, False Positive, and True Negative, respectively. From Table 1, the six performance measures on classification; *accuracy*, *precision*, *recall*, *F-value*, *TP rate*, and *FP rate*, are defined by formulae in (1)-(6).

$$\text{Accuracy} = (TP + TN) / (TP + FN + FP + TN) . \quad (1)$$

$$\text{Recall} = TP / (TP + FN) . \quad (2)$$

$$\text{Precision} = TP / (TP + FP) . \quad (3)$$

$$\text{F-value} = ((1 + \beta)^2 \cdot \text{Recall} \cdot \text{Precision}) / (\beta^2 \cdot \text{Recall} + \text{Precision}) . \quad (4)$$

$$\text{TP Rate} = TP / (TP + FN) . \quad (5)$$

$$\text{FP Rate} = FP / (TN + FP) . \quad (6)$$

The objective of a classifier needs to aim for high prediction performance on a minority class. Considering the definition of *accuracy*, if most instances in a minority class are misclassified and most instances in a majority class are correctly classified by a classifier, the *accuracy* is still high because the large number of negative instances influences the whole classification result on *accuracy*. Note that *precision* and *recall* are effective for this problem because they evaluate the classification rates by concentrating in a minority class. In addition, *F-value* [3] integrating *recall* and *precision*, is used instead of *recall* and *precision*. Its value is large when both *recall* and *precision* are large. The β parameter corresponding to relative importance of *precision* and *recall* is usually set to 1. Furthermore, ROC curve, The Receiver Operating Characteristic curve, is a standard technique for summarizing classifier performance over a range of tradeoffs between *TP rate*, benefits, and *FP rate*, costs. Moreover, AUC [2], Area under ROC, can also be applied to evaluate the performance of a classifier.

The content of this paper is organized as follows. Section 2 briefly describes related works for handling the class imbalanced problem. Section 3 describes the details of our over-sampling technique, Safe-Level-SMOTE. Section 4 shows the experimental results by comparing Safe-Level-SMOTE to SMOTE and Borderline-SMOTE. Section 5 summarizes the paper and points out our future works.